AWS Technical Guide

# Run Semiconductor Design Workflows on AWS

# Run Semiconductor Design Workflows on AWS: AWS Technical Guide

# Table of Contents

This whitepaper is for historical reference only. Some content might be outdated and some links might not be available.

# Run Semiconductor Design Workflows on AWS

Publication date: **March 12, 2021** (*Document history*)

## Abstract

This implementation guide provides you with information and guidance to run production semiconductor workflows on AWS, from customer specification, to front-end design and verification, back-end fabrication, packaging, and assembly. Additionally, this guide shows you how to build secure chambers to quickly enable third-party collaboration, as well as leverage an analytics pipeline and artificial intelligence/machine learning (AI/ML) services to decrease time-to-market and increase return on investment (ROI). Customers that run semiconductor design workloads on AWS have designed everything from simple ASICs to large SOCs with tens of billions of transistors, at the most advanced process geometries. This guide describes the numerous AWS services involved with these workloads, including compute, storage, networking, and security. Finally, this paper provides guidance on hybrid flows and data transfer methods to enable a seamless hybrid environment between on-premises data centers and AWS.

# Migration methodology

When you begin the migration of your semiconductor design workflows to AWS, you will find there are many parallels with managing traditional deployments across multiple sites within your corporate network, whether these sites are distributed engineering locations, or represent an entire data center. Larger organizations in the semiconductor industry typically have multiple data centers that are geographically dispersed because of the distributed nature of their design teams. These organizations typically choose specific workloads to run in specific locations, or replicate and synchronize data to allow for multiple sites to take the load of large-scale, global workflows.

Geographically distributed teams may not lend itself to a simple, and straight-forward approach for migrating workflows to AWS. We encourage our customers to look at specific parts of their design flow or even a new project, when considering which workloads to migrate to AWS. Choosing the right tool or workflow will often require, for example, determining what data replication, caching, and license server management will be needed to run the flow on AWS.

Most of the same approaches and design decisions related to multiple data centers also apply to the cloud. With AWS, you can build one or more virtual data centers that mirror your existing on-premises electronic design automation (EDA) design environment and data center infrastructure. The foundational technologies that enable compute resources, storage servers, and user workstations are available with just a few keystrokes. This ability to rapidly create new semiconductor design and verification environments in just minutes is a major benefit of deploying on cloud. However, the real power of using the AWS Cloud for semiconductor design comes from the dynamic capabilities and enormous scale provided by AWS, resulting in faster time-to-results, reduced schedule risk, and more efficient utilization of valuable EDA software licenses.

# Companion Guide and GitHub Repository

## Companion guide

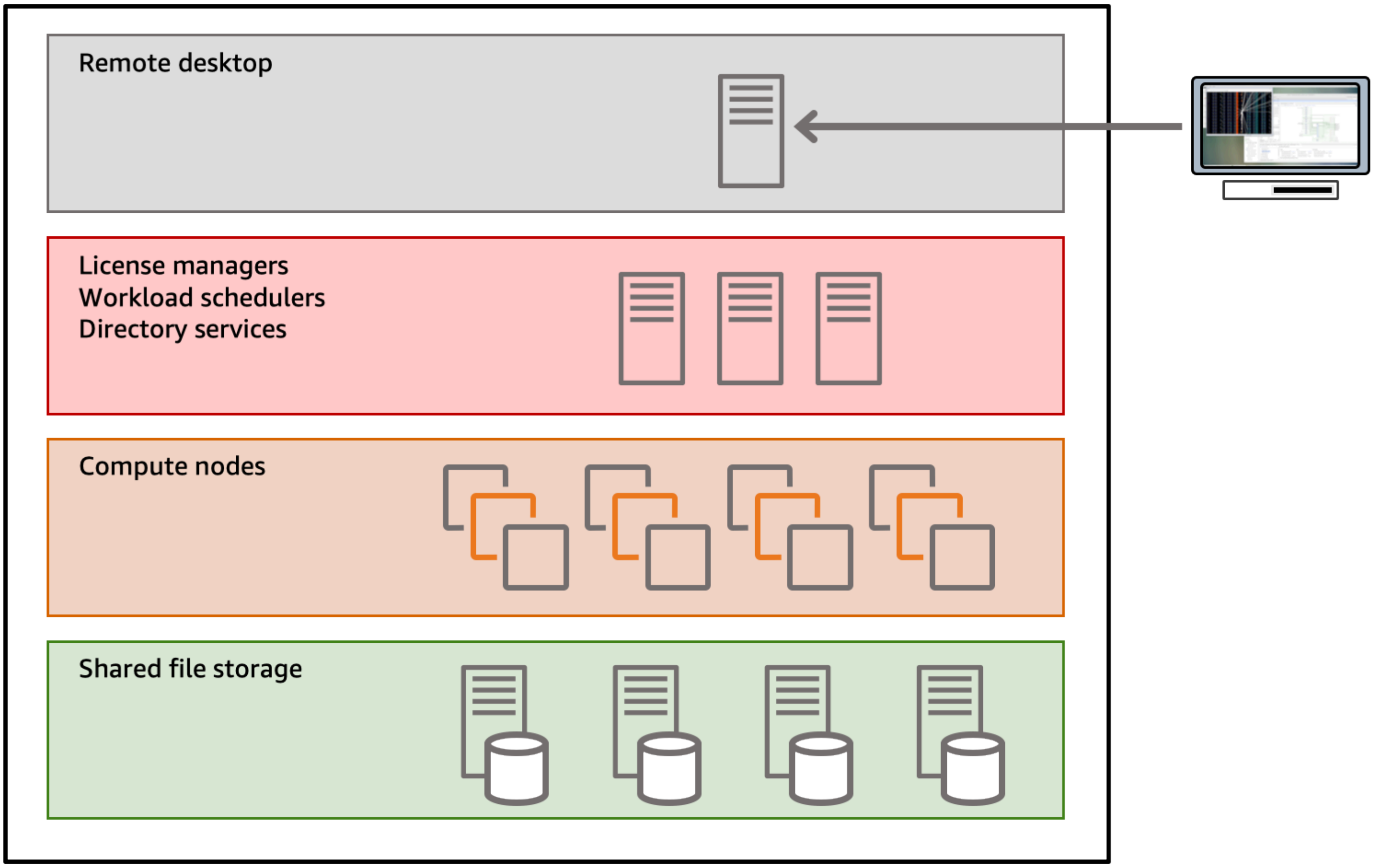
This implementation guide serves as a companion to the Semiconductor Design on AWS whitepaper. Although not necessary, you should consider reading the whitepaper first to ensure you have the building blocks for this guide.

## Supplemental GitHub repository

In addition to the whitepaper and this guide, refer to the Semiconductor Design on AWS GitHub repository for low-level commands for operating system tuning and tool optimization.

# Components of an on-premises semiconductor design environment

The following figure shows a simple view of a typical on-premises semiconductor and electronics design environment. This diagram has the necessary components to run the entire workflow. As you continue through this guide, make sure to reference this diagram as a comparison to the infrastructure that is launched on AWS. That is, all of the components in this diagram are launched and further optimized on AWS.



**Traditional on-premises environment**

# Migrating and running semiconductor workflows on AWS

Migrating workflows or even one tool to AWS can be a somewhat daunting task, particularly if you are new to AWS. As AWS has well over 200 services, and each service is extremely feature rich, this section narrows the scope and focuses on what you need to know for the semiconductor industry.

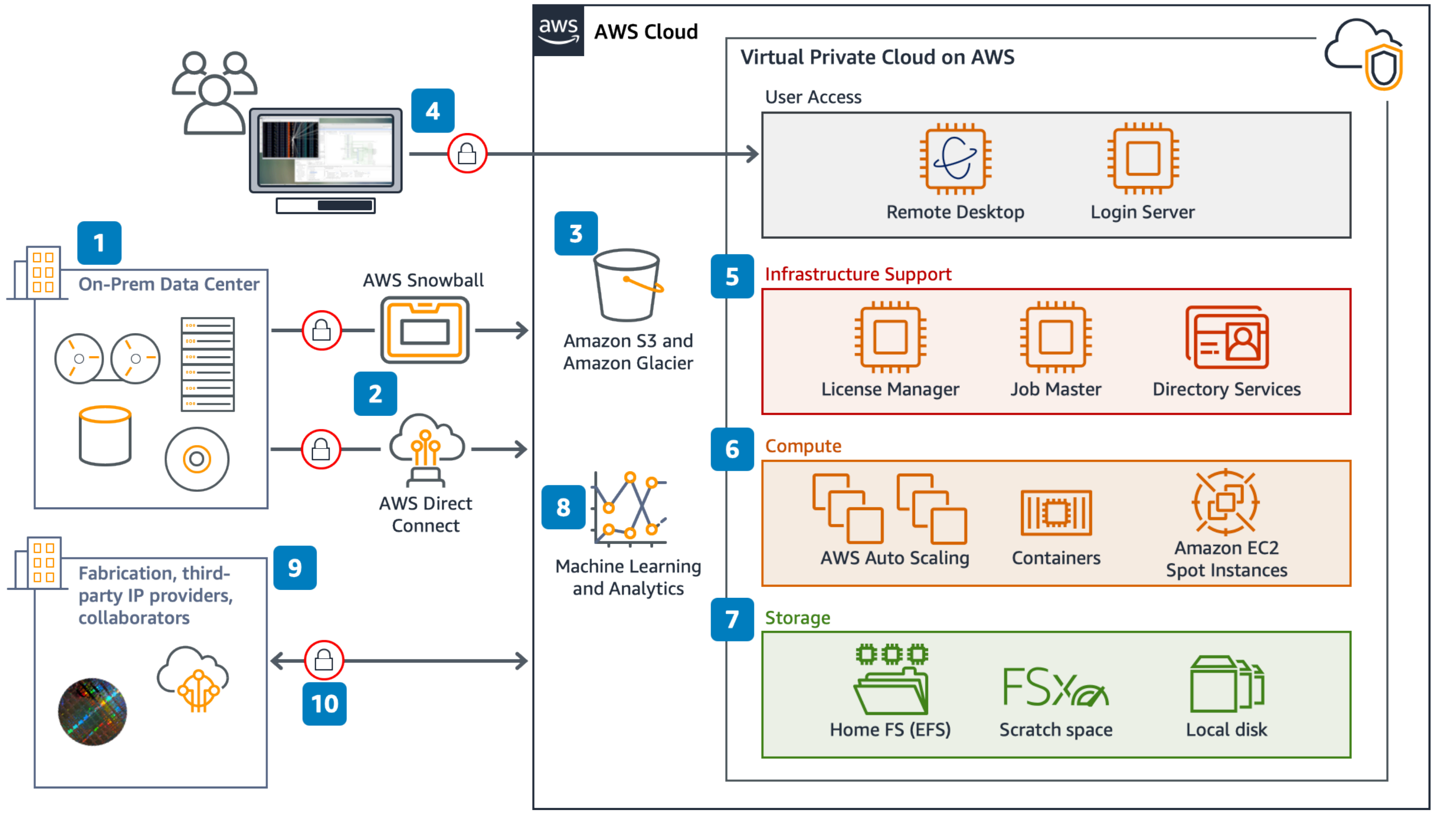
**Topics**

## Before you build

Before you start building out your environment on AWS, we strongly encourage you to read this implementation guide in its entirety. Seeing how the entire environment works together, understanding instance types and sizes, being aware of the storage options, and knowing what security measures will be needed, will allow you to configure an environment that will meet your engineer's requirements, as well as the requirements from IT and security teams.

# Semiconductor and Electronics on AWS - High Level

The following architecture diagram shows migration of workflows to AWS and a high-level architecture for running semiconductor design workflows. In this architecture diagram, the infrastructure that is running on AWS is similar to the previous on-premises environment diagram. This simple architecture helps you understand the high-level approach without the need for knowing the details about each of the services that is used.



### Semiconductor and Electronics Design on AWS - High Level

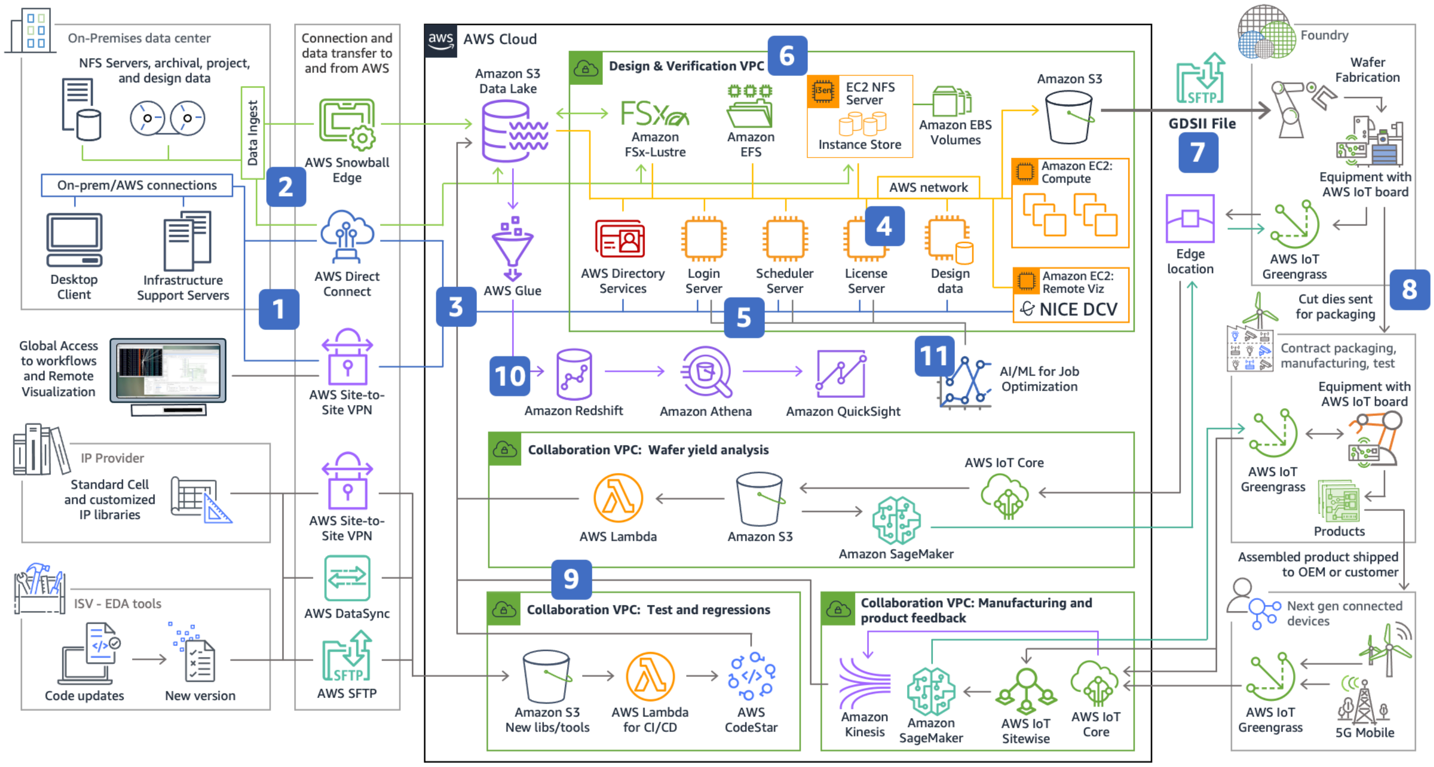
### Table 1 – Semiconductor and Electronics Design on AWS - High Level descriptions

| Callout | Description |
|---|---|
| 1 | Determine what data is needed for proof of concept or test. |
| 2 | Transfer data into AWS via **AWS Snowball Edge**, **AWS Direct Connect**, or using sever services. |

| Callout | Description |
|---|---|
| **3** | Transferred data is stored in **Amazon S3** buckets. You can access data stored in Amaz **Amazon EC2** instance or nearly any AWS service. |
| **4** | Users access their environment through a remote desktop session or command line (s |
| **5** | All of the infrastructure needed for semiconductor design workflows is available on A |
| **6** | AWS compute is flexible and robust, more than capable of running semiconductor de |
| **7** | Store tools and job data on Amazon EFS, Amazon FSx for Lustre, and local disk. Optio long-term data storage to Amazon S3. |
| **8** | Once your data is in AWS, you can leverage other services, such as data lakes, AI/ML, |
| **9** | Isolating environments leads to enhanced security and limits third parties to only the |
| **10** | Encryption is everywhere and can be enabled with your encryption keys. |

## Semiconductor and Electronics on AWS - Deep Dive

The following diagram is a complex architecture and gives a detailed guide for running your entire workflow, from customer specification, to silicon, to manufactured product. (For a PDF of this diagram, see Reference Architecture: Semiconductor and Electronics on AWS). The following table provide links to the respective section in this paper describing each component of this architecture along with the specific AWS services involved.



## Semiconductor and Electronics Design on AWS - Deep Dive

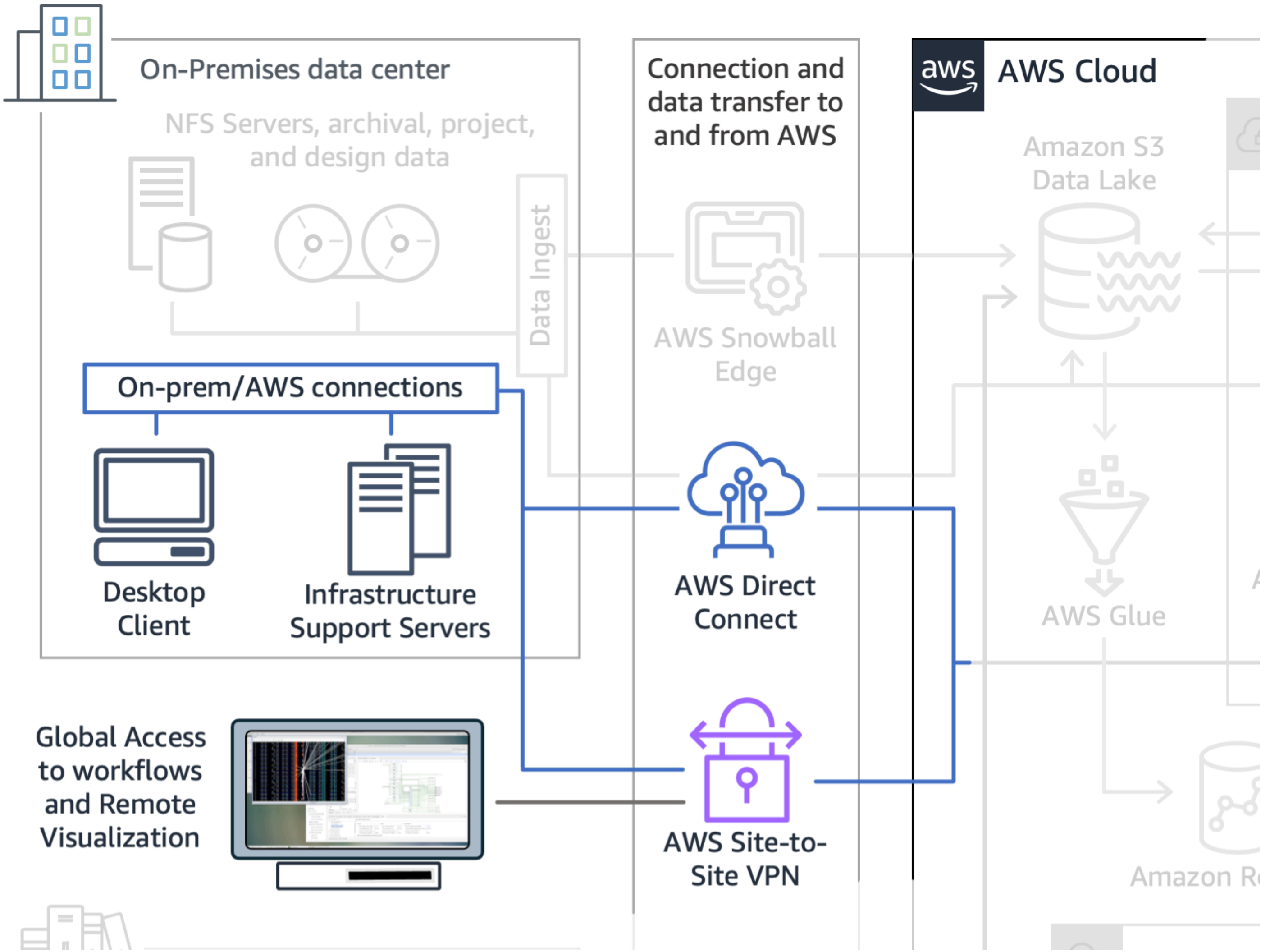
### Table 2 – Semiconductor and Electronics Design on AWS - Deep Dive desciptions

| Callout | See section |
|---|---|
| 1 | Connecting to AWS |
| 2 | Data requirements and transfer for proof of concept (POC) |
| 3 | User access and remote desktop |
| 4 | License server setup |
| 5 | Launch AWS services needed to run a POC |
| 6 | Launch and configure the entire semiconductor design workflow |
| 7 | Transfer GDSII file to foundry |
| 8 | Fabrication, packaging, and final product |

| Callout | See section |
|---|---|
| 9 | Enable secure collaboration chambers with third parties |
| 10 | AWS analytics pipeline and leveraging your data lake |
| 11 | AI/ML for workflow optimization |

## Connecting to AWS

When moving workflows to AWS, the first step is to determine how to best connect on-premises systems to AWS. The on-premises connections can range from simple thin clients, to entire legacy enterprise infrastructure that require many levels of authentication and user access. Minimally, we suggest that you use an AWS VPN connection to ensure a secure environment. That said, even though AWS VPN provides basic connectivity, using AWS Direct Connect can both solve security requirements and data transfer issues as well. That is, as you move data in to AWS, having a robust connection can establish a foundation for hybrid environments and allow for straightforward inbound and outbound data for the entire workflow.



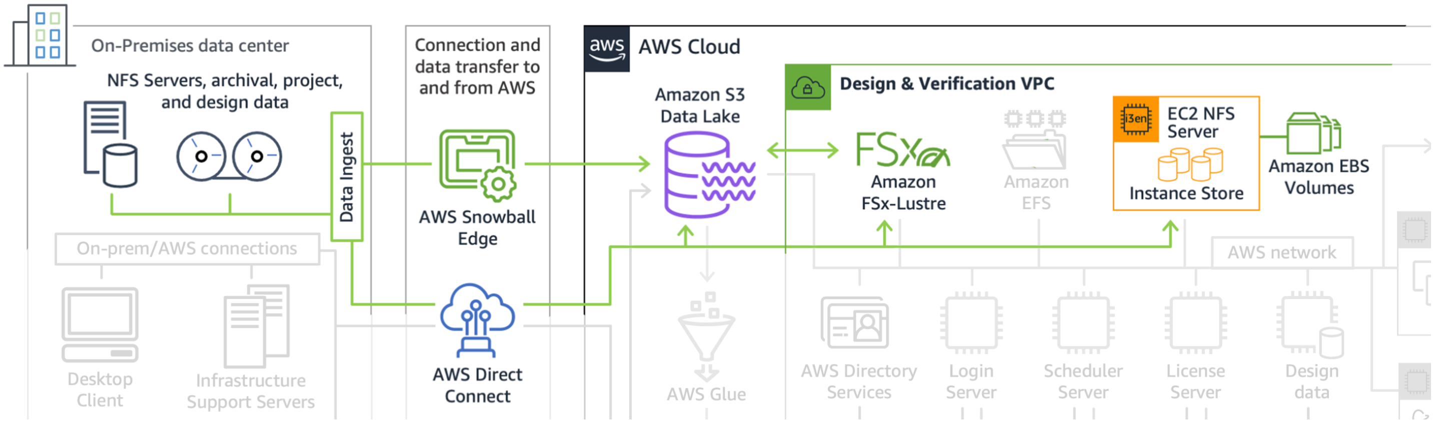
### Connecting to AWS

The preceding figure shows initial connectivity recommended to customers when first moving workflows. These connections are used for running workflows and data movement. Once you have a secure, robust connection to AWS you can start thinking about what data will be required to run your first workload.

## Data requirements and transfer for proof of concept (POC)

Once connections have been established with AWS, you should determine what data will be needed for a proof of concept (POC) and transfer that data to an Amazon Simple Storage Service (Amazon S3) bucket. The data movement shown in the following figure highlights two methods for transferring data into AWS. For customers that have a large amount of static data, we suggest using AWS Snowball Edge Edge for a one time transfer (supports up to 80 TB per Snowball Edge Edge device of usable storage). If you have a small amount of data that you want to use for initial testing or POC, then using an AWS Site-to-Site VPN connection or AWS Direct Connect is a viable and secure method of transfer. For a long term solution, we recommend using an AWS Direct Connect for both data transfer and enabling hybrid environments. Using AWS Direct Connect, you can establish private connectivity between AWS and your data center, office, or colocation environment, which in many cases can reduce your network costs, increase bandwidth throughput, and provide a more consistent network experience than internet-based connections.

Once the data has been transferred to an Amazon S3 bucket, you can quickly move data to an Amazon FSx for Lustre file system or to another instance for testing and POCs.



**Data requirements and transfer for proof of concept**

> **Note**
>
> There are additional services for data transfer that are beyond the scope of this paper, such as AWS DataSync, and partner solutions. For more information, see Migration & Transfer on AWS.

## Use of Amazon S3 data lake

In the preceding figure, the Amazon S3 bucket is shown as an Amazon S3 data lake. A data lake is a centralized repository that allows you to store all your structured and unstructured data at any scale. You can start with an Amazon S3 bucket, but we encourage you to consider converting the S3 bucket to an Amazon S3 data lake. This approach enables you to quickly begin analyzing data, both archival and new data. The data lake serves as the central data repository that can be leveraged for advanced analytics machine learning.

## Consider what data to move to Amazon S3

Prior to moving your flows to AWS, consider the processes and methods that will be in place as you move from initial experiments, to POC, and then full production.

We encourage customers to start with a relatively small amount of data; for example, only the data required to run a simulation job (hardware description language [HDL] compile and simulation workflow), or a subset of jobs. This approach allows you to quickly gain experience with AWS and build an understanding of how to build production-ready architectures on AWS, while also running an initial chip design workload. Optimization of your flows will come as your knowledge of AWS expands, and you are able to leverage more AWS services.

Data is gravity, and moving only the limited amount of data needed to run your tools to Amazon S3 allows for flexibly and agility when building and iterating your architecture on AWS. The S3 transfer speed to an Amazon Elastic Compute Cloud (Amazon EC2) instance is up to 25 Gbps per instance. With your data stored in Amazon S3, you can quickly experiment with different Amazon EC2 instance types, and experiment with different storage options as well.

## Dependencies

Semiconductor design environments often have complex dependencies that can hinder the process of moving workflows to AWS. AWS Solutions Architects can work with you to build an initial POC or even a complex architecture. However, it is often the designer's or the tool engineer's responsibility to identify and optimize any legacy on-premises dependencies. The initial POC process will require some effort to determine which dependencies, such as shared libraries, will need to be moved along with project data for a first test. There are tools available that help you build a list of run-time dependencies, and some of these tools yield a file manifest that expedites the process of moving data to AWS. For example, Altair/Ellexus has several tools that can be used to profile applications and determine dependencies. For more information, see Ellexus on the AWS Marketplace.

License sever dependencies and setup can be mitigated by using on-premises license servers while running simple tests on AWS that do not require frequent check outs. However, note that using an on-premises license server while running the job on AWS may adversely affect the run-time. This configuration may only be used temporarily, and for testing only. Having the license server in AWS, with your entire flow, will result in the fastest access to licenses and faster run times.

Dependencies can also include authentication methods (e.g., Active Directory, NIS, LDAP), shared file systems, cross organization collaboration, and globally distributed designs. Identifying and managing such dependencies is not unique to cloud migration; semiconductor design teams face similar challenges in any distributed environment.

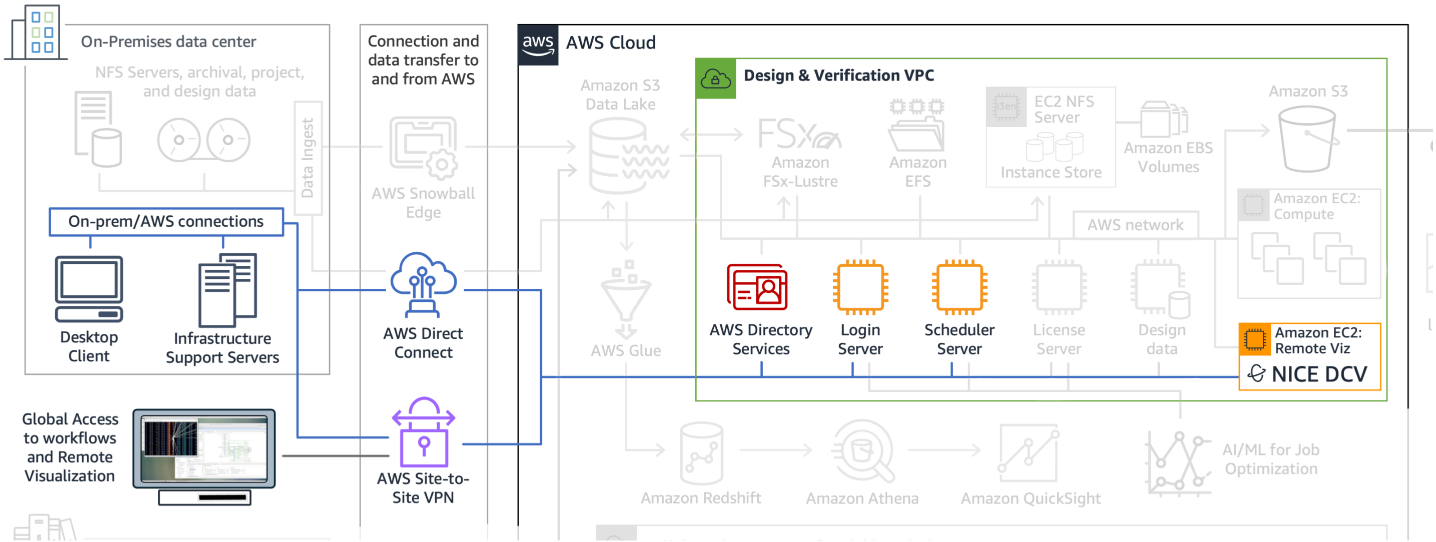
Rather than moving an existing workflow, you may avoid a considerable amount of work and launch a net-new semiconductor design and/or verification project on AWS, which should significantly reduce the number of legacy dependencies. For example, you might choose to focus only on one separable part of the flow, such as timing analysis, for your first POC. Or you might choose to test the complete chip design flow, and have the entire project and file dependencies moved to cloud for the duration of the project.

Data management and movement is especially important when migrating and running chip design flows on AWS. It is a critical initial step, that should not be underestimated. Without the data necessary to run even one tool, building out the environment becomes an academic exercise only.

## User access and remote desktops

Once users and engineers have the data they need to run flows, enabling access can be accomplished through multiple mechanisms. The same methods that are being used by your engineers to access on-premises systems can also be used for accessing AWS resources. Many engineers login via command line with ssh, or through a web UI for managing job submission and monitoring. These same methods are available on AWS.

The following figure shows connectivity from on-premises systems to AWS infrastructure. The additional launched services and resources are AWS Directory Service, login server, scheduling server, and a remote desktop using NICE DCV. The login and scheduler servers, remote desktop instance, testing, and POC should all use moderately sized instance types. For example, M5.2xlarge, T2, or T3 instance types should suffice for these applications. We provide additional details about which instances to use in the Amazon Elastic Compute Cloud (Amazon EC2) section.



**User access and remote desktops**

## Remote Desktops

Remote desktops are a cost effective way to bring the engineer to the data, rather than bring the data to the engineer. Remote desktops can be used to view waveforms or step through a simulation to identify and resolve register-transfer-level (RTL) regression errors, or it might be necessary to view a 2D or 3D graphical representation of results generated during signal integrity analysis. Some applications, such as printed circuit layout software, are inherently interactive and require a high quality, low latency user experience.

There are multiple ways to deploy remote desktops on AWS. You have the option of using open-source software such as Virtual Network Computing (VNC), or commercial remote desktop solutions available from AWS partners. You can also make use of AWS solutions, including NICE desktop cloud visualization (NICE DCV).

### NICE DCV

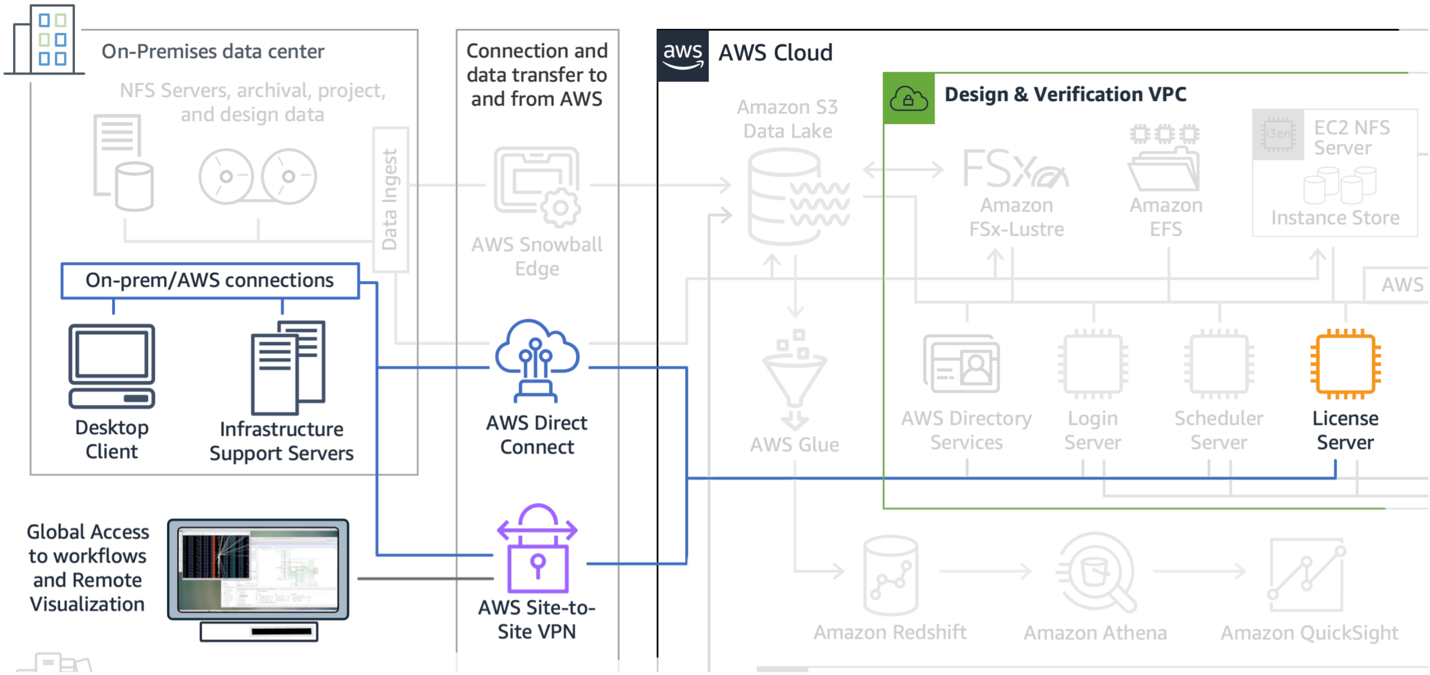
NICE Desktop Cloud Visualization (NICE DCV)is a remote visualization technology that enables users to securely connect to graphic-intensive 2D and 3D applications hosted on an Amazon EC2 instance. With NICE DCV, you can provide high-performance graphics processing to remote users by creating secure client sessions. NICE DCV was specifically designed for high performance technical applications, and is an excellent choice for semiconductor design workloads. For more information, see NICE DCV.

# License server setup

Application licensing is required for most semiconductor design workloads, both on-premises and on AWS. From a technical standpoint, managing and accessing EDA licenses can remain unchanged when migrating to AWS. Over time, you may choose to augment your existing license management using AWS services, for example using analytics and machine learning services on AWS to monitor and improve license utilization across projects, and across sites. You can also use features of AWS, such as AWS Auto Scaling, to improve license server reliability and reduce time to market.

Because of the connectivity options available, you can run your license servers on AWS using an Amazon EC2 instance or within your own data centers. By allowing connectivity through a VPN or AWS Direct Connect between cloud resources and on-premises license servers, you can seamlessly run workloads in any location without having to dedicate some of your valuable EDA licenses to your AWS computing environments, thereby fragmenting your license pools.

The following figure shows how to set up your on-premises license servers on AWS, so you can both run your existing on-premises flows using the license servers hosted on AWS, and also use the same servers when migrating your flows over to AWS. As with the other infrastructure support instances, a moderately sized instance (e.g., M5.2xlarge) will suffice for testing and POC environments.



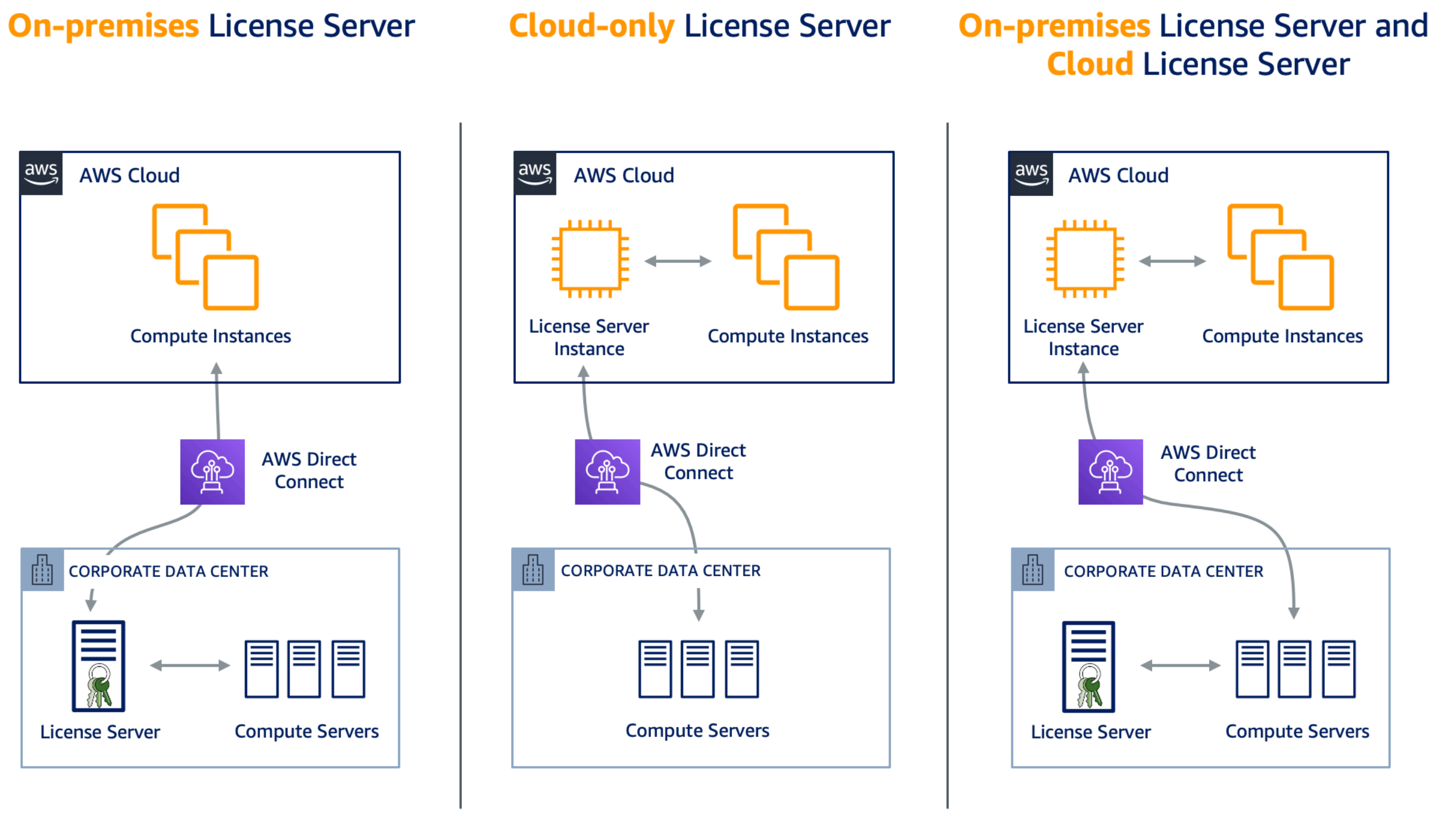
### License server setup

## License Server Access

On AWS, each Amazon EC2 instance launched is provided with a unique hostname and hardware address (MAC) using Amazon elastic network interfaces that cannot be cloned or spoofed. Therefore, traditional license server technologies (such as Flexera) work on AWS without any modification. The inability to clone license servers, which is prevented by AWS by not allowing the duplication of MAC addresses, also provides software vendors with the confidence that software can be deployed and used in a secure manner.

Although you can configure your EDA tools to use on-premises license servers while running in AWS, the latency may adversely affect the run time of the job. For initial tests and POCs, using an on-premises license server is a suitable solution. For production and for like-to-like comparisons, we recommend setting up license servers in AWS. This approach reduces the need to connect back in to on-premises license servers, and eliminates the latency penalty incurred at license check out.

The following figure shows three license server deployment scenarios – on premises only license server, cloud-only license server, and on-premises and cloud license servers. Your architecture depends on many factors, but these three options can provide guidance to build out a reliable and scalable licensing architecture. If you plan on using an on-premises license server while running on AWS, as many licensed applications are sensitive to network latency from client to server, a dedicated connection from your on-premises network to the nearest AWS Region using AWS Direct Connect can provide a reliable network connection with consistent latency.



**License server deployment scenarios**

## Improving License Server Reliability

License servers are critical components in all chip design environments. Hosting licenses in the AWS Cloud can provide improved reliability of license services with the use of a floating elastic network interface. The elastic network interface has a fixed, immutable MAC address that can be associated with software license keys.

The implementation of this high availability solution begins with the creation of a elastic network interface that is attached to a license server instance. Your license keys are associated with this elastic network interface. If a failure is detected on this instance, you, or your custom automation, can detach the elastic network interface and attach it to a standby license server. Because the elastic network interface maintains its IP and MAC addresses, network traffic begins flowing to the standby instance as soon as you attach the elastic network interface to the replacement instance.

This unique capability enables license administrators to provide a level of reliability that can be difficult to achieve using on-premises servers in a traditional datacenter.

## Working with Independent Software Vendors (ISVs)

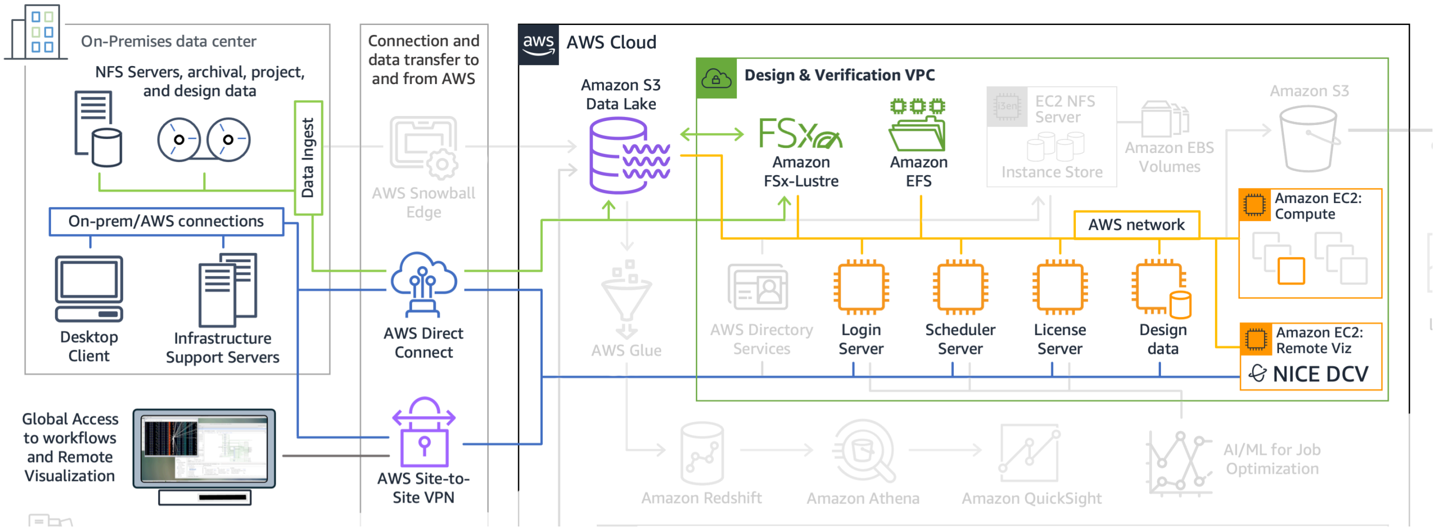
AWS works closely with thousands of independent software vendors (ISVs) that deliver solutions to customers on AWS using methods that may include software as a service (SaaS), platform as a service (PaaS), customer self-managed, and bring your own license (BYOL) models. In the semiconductor industry, AWS works closely with the major vendors of EDA software to help optimize performance, scalability, cost, and application security. AWS can assist ISVs and your organization with deployment best practices described in this guide.

AWS Partner Programs are designed to support the unique technical and business requirements of AWS Partner Network (APN) members by providing them with increased support from AWS, including access to AWS Partner team members who specialize in design and engineering applications. In addition, AWS has a growing number of APN Consulting Partners who can assist vendors and their customers with cloud migration.

# Launch AWS services needed to run a POC

The diagrams in preceding sections showed connectivity, data movement, user access, and license server access separately. The following figure brings those elements together, and adds Amazon EC2 instances used for the compute fleet and for design data. This environment is recommended for initial testing and POCs.

When launching services on AWS for your test or POC, you should be using repeatable mechanisms. For example, AWS CloudFormation (referred to as infrastructure as code) allows for a repeatable method for deploying infrastructure. The environment shown in following figure can be launched with an AWS CloudFormation template, or using the AWS Solutions Implementation which automates the process and allows you to launch this environment quickly and repeatedly. For more information, see the AWS Solutions Implementation: Scale-Out Computing on AWS section.



### Services needed to run a POC

Prior to building the environment shown in the preceding figure, you should be familiar with the compute, storage, file system, and network options recommended for semiconductor design workflows.

## Amazon Elastic Compute Cloud (Amazon EC2)

Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides secure, resizable compute capacity in the cloud. It is designed to make web-scale cloud computing easier for developers. Amazon EC2's simple web service interface allows you to obtain and configure capacity with minimal friction. It provides you with complete control of your computing resources and lets you run on Amazon's proven computing environment.

Amazon EC2 offers the broadest and deepest compute platform with choice of processor, storage, networking, operating system, and purchase model. We offer the fastest processors in the cloud and we are the only cloud with 400 Gbps ethernet networking. We have the most powerful GPU instances for machine learning training and graphics workloads, as well as the lowest cost-per-inference instances in the cloud.

Amazon EC2 instances use a hypervisor to divide resources on the server, so that each customer has separate CPU, memory, and storage resources for just that customer's instance. The hypervisor is not used to share resources between instances, except for the T* instance type family. In current-generation instances, for example M6g, R6g, C5, and Z1d, these instances use a specialized piece of hardware and a highly customized hypervisor based on KVM. This hypervisor system is called the AWS Nitro System.

Although there are older instance types (not using the AWS Nitro System) that can run your workflows, this guide focuses on current generation instance types that use the AWS Nitro System.

### Instance capabilities

Many of the Amazon EC2 instances have features that are specified in the name of the instance family and type. For example, for the instance family R5, there is also a variant that has local NVMe storage (Amazon EC2 instance store) that is named R5d. Similarly, the C6g is powered by the AWS Graviton2 Processor, and the C6gn is both powered by the AWS Graviton2 Processor and also has support for enhanced networking with up to 100 Gbps of network bandwidth. This guide includes notes when there are variants of recommended instances types.

### AWS Nitro System

The AWS Nitro System is the underlying platform for the next generation of EC2 instances that offloads many of the traditional virtualization functions to dedicated hardware and software to deliver high performance, high availability, and high security while also reducing virtualization overhead. The Nitro System is a rich collection of building blocks that can be assembled in many different ways, giving us the flexibility to design and rapidly deliver new EC2 instance types with an ever-broadening selection of compute, storage, memory, and networking options.

Launching Nitro-based instances requires specific drivers for networking and storage be installed and enabled before the instance can be launched. Many operating systems that can run design workflows have the necessary drivers already installed and configured. The recommended operating systems in this guide will already have the necessary drivers installed and configured.

## AWS Graviton powered instances

AWS Graviton processors are custom built by AWS using 64-bit Arm Neoverse cores to deliver the best price performance for your cloud workloads running on Amazon EC2. Many of the existing tools used for semiconductor design will run on AWS Graviton instances, and more are continually being enabled.

The Amazon EC2 instances M6g, C6g, R6g, and X2gd and their disk variants with local NVMe-based SSD storage deliver up to 40% better price/performance over comparable x86-based instances for a broad spectrum of workloads, including application servers, microservices, high-performance computing, CPU-based machine learning inference, electronic design automation, game applications, open-source databases, and in-memory caches. To optimize for price and performance, we suggest using AWS Graviton instances for tools and workloads that are compatible with the AWS Graviton processor. As of this writing, the primary EDA ISVs offer Arm versions of several of their tools. Contact your tool provider to learn which tools are available for AWS Graviton2-based instances.

# Choice of instance types for semiconductor design

Although AWS has many different types and sizes of instances, the instance types in the compute-optimized and memory-optimized categories are typically best suited for chip design workloads.

## Compute-Optimized instances

The compute-optimized instance family features instances that have the highest clock frequencies available on AWS, and typically enough memory to run some memory-intensive workloads.

The C6g, M6g, M5zn, Z1d, C5, and X2gd (and their disk variants) are appropriate for semiconductor design workloads. Typical EDA use cases for compute-optimized instance types include:

- Digital, analog, and mixed-signal simulations
- Physical synthesis
- Formal verification
- Regression tests
- IP characterization

## Memory-Optimized Instances

The memory-optimized instance family features instances that have a footprint to run the largest semiconductor design workloads.

You can choose from the R6g, Z1d, and R5 (and their disk variants) memory-optimized instances. Typical use cases for memory-optimized instance types:

- Place and route
- Timing and power analysis
- Physical verification
- Design rule checking (DRC)
- Batch mode RTL simulation (multithread optimized tools)

The following table provides detailed information for the instance types and the corresponding design tools or infrastructure use case recommended for chip design workflows.

**Note**

AWS uses vCPU to denote processors or symmetric multi-threading. This table uses physical cores.

**Table 3 – Instance types and corresponding design tools or infrastructure usage**

| Name** | Max Physical Cores | Max RAM in GiB and (GiB/core) | Local NVMe | Design Tool or Application |
|---|---|---|---|---|
| M6g | 64 | 256 (4) | Yes* | Formal verification |
| C6g | 64 | 128 (2) | Yes* | RTL Simulation Batch |
| M5zn | 24 | 192 (8) | No | RTL Simulation Interactive |
| Z1d | 24 | 384 (16) | Yes | RTL Gate Level Simulation |
| M5 | 48 | 384 (8) | Yes* | Synthesis/Compilation |
| | | | | Library Characterization |
| R6g | 64 | 512 (8) | Yes* | RTL Simulation Multi-Threaded |
| R5 | 48 | 768 (16) | Yes* | Extraction |
| | | | | DRC |
| | | | | Optical Proximity Correction |
| | | | | Library Characterization |
| M6g | 64 | 256 (4) | Yes* | Remote Desktop Sessions |
| M5 | 48 | 384 (16) | Yes* | |

| Name** | Max Physical Cores | Max RAM in GiB and (GiB/core) | Local<br>NVMe | Design Tool or Application |
|---|---|---|---|---|
| C6g | 64 | 128 (2) | Yes* | RTL Simulation Interactive<br>RTL Gate Level Simulation |
| C5 | 36 | 144 (4) | Yes* | RTL Simulation Interactive<br>RTL Gate Level Simulation |
| X2gd | 64 | 1,024 (16) | Yes* | Place & Route<br>Static Timing Analysis<br>Full Chip Simulation<br>Optical Proximity Correction |
| X1e | 64 | 3,904 (61) | Yes* | Place & Route<br>Static Timing Analysis<br>Full Chip Simulation |

** Supported on disk variant (e.g., M6gd, C5d, etc.)*

*** **g** - uses AWS Graviton Processors; **z** - higher clock frequency; **n** - enhanced networking of up to 100 Gbps*

## Hyper-Threading for EC2 instances with Intel Processor Technologies

Amazon EC2 instances with Intel processors support Intel Hyper-Threading Technology (HT Technology), which enables multiple threads to run concurrently on a single Intel CPU core. Each thread is represented as a virtual CPU (vCPU) on the instance. Each vCPU is a hyperthread of an Intel CPU core, except for T2 instances. To determine the physical cores, you divide the vCPU number by 2. If you determine that it has a negative impact on your application's performance, you can disable HT Technology when you launch an instance using CPU Options (which is an EC2 feature).

## CPU options (EC2 instance feature)

You can specify the following CPU options to optimize your instance:

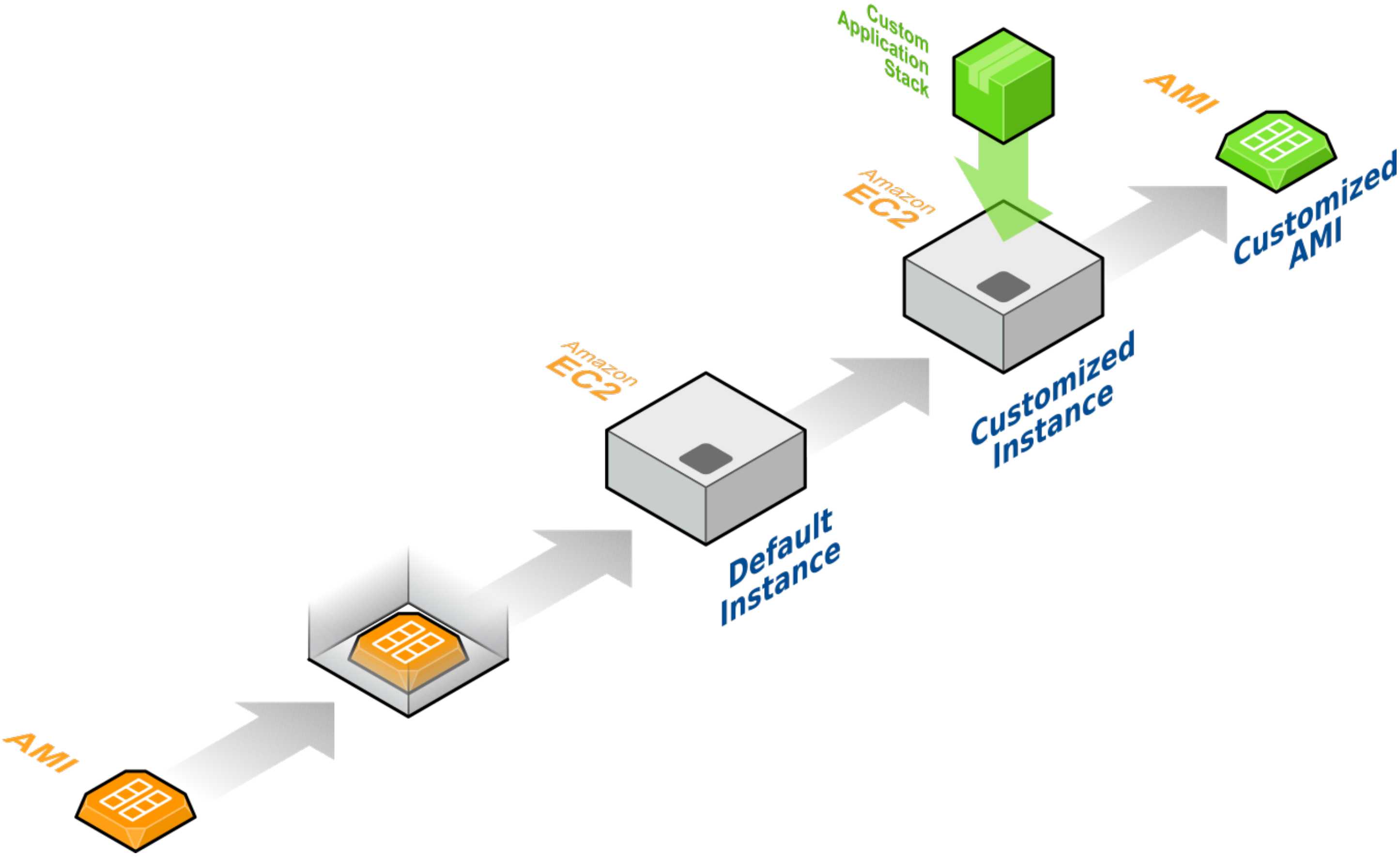
- **Number of CPU cores –** You can customize the number of CPU cores for the instance. This customization may optimize the licensing costs of your software with an instance that has sufficient amounts of RAM for memory-intensive workloads but fewer CPU cores.
- **Threads per core –** For AWS Graviton powered instances, there is one thread per core. For instances with Intel processors, you can disable Intel Hyper-Threading Technology by specifying a single thread per CPU core. This scenario applies to certain workloads, such as high performance computing (HPC) workloads.

You can only specify the CPU options during instance launch (for running instances, you can still disable multi-threading.) For details, see the Semiconductor Design on AWS GitHub repository. There is no additional or reduced charge for specifying CPU options. You are charged the same amount as instances that are launched with default CPU options. For more information and rules for specifying CPU options, see Optimizing CPU Options in the *Amazon Elastic Compute Cloud User Guide for Linux Instances*.

# AMI and operating system

AWS has built-in support for numerous operating systems. For semiconductor design, CentOS, Red Hat Enterprise Linux, and Amazon Linux 2 are used more than other operating systems. The operating system and the customizations that you have made in your on-premises environment are the baseline for building out your architecture on AWS. Before you can launch an EC2 instance, you must decide which Amazon Machine Image (AMI) to use. An AMI is used to boot EC2 instances, contains the OS, has any required OS and driver customizations, and may also include tools and applications. For EDA, one approach is to launch an instance from an existing AMI, customize the instance after launch, and then save this updated configuration as a custom AMI. Instances launched from this new custom AMI include the customizations that you made when you created the AMI.

The following figure shows the process of customizing an AMI.



**Use the Amazon provided AMI to build a customized AMI**

You can select the AMI from the AWS Management Console or from the AWS Marketplace, and then customize that instance by installing your EDA tools and configuring the environment to match what is running on-premises. After that, you can use the customized instance to create a new, customized AMI that you can then use to launch your entire EDA environment on AWS. Note also that the customized AMI that you created can be further customized at instance launch. For example, you can customize the AMI to add additional application software, load additional libraries, or apply patches, each time the customized AMI is launched onto an EC2 instance (this is done using the *User data* option at instance launch).

As of this writing, we recommend these OS levels for tools, infrastructure, and file system support:

- Amazon Linux 2 (verify certification with EDA tool vendors)
- CentOS 7.5 or newer
- Red Hat Enterprise Linux 7.5 or newer
- SUSE Linux Enterprise Server 12 Service Pack 4 or newer

These OS levels have the necessary drivers already included to support the current instance types, which include Nitro based instances. If you are not using one of these levels, you may need to perform extra steps to take advantage of the features of our current instances. Specifically, you may need to build and enable enhanced networking, and install and configure the NVMe drivers. For detailed information on AMI drivers, see the Semiconductor Design on AWS GitHub repository.

You can import your own on-premises image to use for your AMI. This process includes extra steps, but may result in time savings. Before importing an on-premises OS image, you first build a virtual machine (VM) image for your OS. AWS supports certain VM formats (for example, Linux VMs that use VMware ESX) that must be uploaded to an S3 bucket, and subsequently converted into an AMI. For detailed information and instructions, see VM Import/Export. The same operating system requirements mentioned previously are also applicable to imported images.

To verify that you can launch your AMI on a Nitro based instance, first launch the AMI on a Xen based instance type (e.g., C4), and then run the *NitroInstanceChecks* script found on the AWS Support Tools GitHub repository. This script analyzes your AMI and determines if it can run on a Nitro based instance. If it cannot, it displays recommended changes.

## Network

Amazon enhanced networking technology enables instances to communicate at up to 100 Gbps and 25 Gbps for current-generation instances, and up to 10 Gbps for previous-generation instances. In addition, enhanced networking reduces latency and network jitter. The recommended AMIs in the previous section include the required Elastic Network Adapter (ENA) module and have ENA support enabled. If you are unsure if your AMI or instance supports enhanced networking, see Enhanced Networking on Linux in the *Amazon Elastic Compute Cloud User Guide for Linux Instances*. This reference includes which instance types are currently supported, and if necessary, the steps required to enable support.

## Storage

For semiconductor design flows running at scale, storage can be the bottleneck that reduces job throughput. Traditionally, centralized filers serving network file systems (NFS) are commonly purchased from hardware vendors at significant costs in support of high throughout. However, these centralized filers can quickly become a bottleneck, resulting in increased job run times and correspondingly higher license costs. As the amount of data increases, the need to access that data across a fast-growing cluster means that the filers eventually run out of storage space, or become bandwidth constrained by either the network or storage tier.

The following sections provide information on currently available storage options recommended for semiconductor workflows:

- **Object storage** - Amazon S3
- **Block storage** - Amazon Elastic Block Store (Amazon EBS), and Amazon EC2 instance store (NVMe storage local to Amazon EC2 instances)
- **File storage** - Amazon Elastic File System (Amazon EFS), and Amazon FSx for Lustre

With the combination of these storage options, you can enable an elastic, cost-optimized, storage solution for your entire workflow that will eliminate storage bottlenecks typically found in chip design flows.

## Object Storage - Amazon S3

Amazon Simple Storage Service (Amazon S3) is an object storage service that offers industry-leading scalability, data availability, security, and performance. Amazon S3 provides easy-to-use management features so you can organize your data and configure finely-tuned access controls. Amazon S3 is designed for 99.999999999% (11 9's) of durability, and stores data for millions of applications for companies all around the world.

Amazon S3 has various features you can use to organize and manage your data in ways that support specific use cases, enable cost efficiencies, enforce security, and meet compliance requirements. Data is stored as objects within resources called “buckets”, and a single object can be up to 5 terabytes in size. Amazon S3 features include capabilities to append metadata tags to objects, move and store data across the Amazon S3 storage classes, configure and enforce data access controls, secure data against unauthorized users, run big data analytics, and monitor data at the object, bucket levels, and view storage usage and activity trends across your organization.

In particular, Amazon S3 storage classes can be used to define a data lifecycle strategy that can significantly reduce storage costs, without sacrificing access to critical data.

For semiconductor design workflows, we recommend Amazon S3 for your primary data storage solution. Today, EDA tools do not provide a built-in interface to object storage, so you will need to move data to a POSIX compliant file system before running jobs. This task can easily be performed as the file system is created, or when instances are being launched. Amazon S3 provides flexibility and agility to quickly move your data from object storage, to block storage, to file storage, and back to object storage. For example, you can quickly and efficiently copy data from Amazon S3 to Amazon EC2 instances and Amazon EBS storage to populate a high performance shared file system prior to launching a large batch regression test or timing analysis. Additionally, the same Amazon S3 bucket can be used to populate an Amazon FSx for Lustre file system that is used only once, or is persistent and used for large chip designs.

## Block storage - Amazon EBS

Amazon Elastic Block Store (Amazon EBS) is an easy to use, high-performance, block-storage service designed for use with Amazon EC2 for both throughput and transaction intensive workloads at any scale. You can choose from six different volume types to balance optimal price and performance. You can change volume types, tune performance, or increase volume size without disrupting your applications. Amazon EBS volumes are replicated within an Availability Zone and can easily scale to petabytes of data. Also, you can use Amazon EBS Snapshots with automated lifecycle policies to back up your volumes in Amazon S3, while ensuring geographic protection of your data and business continuity.

Amazon EBS volumes appear as local block storage that can be formatted with a file system on the instance itself. Amazon EBS volumes offer the consistent and low-latency performance required to run semiconductor workloads.

When selecting your instance type, you should select an instance that is Amazon EBS-optimized by default. The previously recommended instances are all EBS-optimized by default. If your application requires an instance that is not Amazon EBS optimized, see the EBS optimization section in the *Amazon Elastic Compute Cloud User Guide for Linux Instances*.

Additionally, there are several EBS volume types that you can choose from, with varying performance characteristics and pricing options. At the time of this writing, we recommend using Amazon EBS gp3 general purpose volumes for your default EBS volume type. The gp3 volume provides the performance needed for nearly any application, to include most file servers. For additional information about performance (to include throughput and IOPS), see Amazon EBS Volume Types in the *Amazon Elastic Compute Cloud User Guide for Linux Instances***.**

Although we recommend using AWS managed storage services, if you plan on building and maintaining your own NFS file servers on AWS, you need to use instance types and EBS volume types that are more suited for high performance throughput and lower latency. For example, the Amazon EC2 C6gn instance with 10 gp3 EBS volumes attached is capable of up to 160,000 IOPS, 4.75 GB/s (using 128 KiB I/O), and 100 Gbps network connectivity.

For more information about EBS-optimized instances, and to determine which instance meets your file system server requirements, see Amazon EBS-optimized instances in the *Amazon Elastic Compute Cloud User Guide for Linux Instances*.

### Amazon EC2 instance store

For use cases where the performance of Amazon EBS is not sufficient on a single instance, Amazon EC2 instances with Amazon EC2 instance store are available. An *instance store* provides temporary block-level storage for your instance. This storage is located on disks that are physically attached to the host computer, and the data on the instance store does not persist when you stop or terminate the instance. Additionally, hardware failures on the instance would likely result in data loss. For these reasons, instance store is recommended for temporary storage of information that changes frequently, such as buffers, caches, scratch data, and other temporary content, or for data that is replicated across a fleet of instances. You can replicate data off of the instance (for example, to Amazon S3), and increase durability by choosing an instance with multiple instance store volumes, and create a RAID set with one or more parity volumes.

Table 1 includes instances that are well-suited for chip design workloads requiring a significant amount of fast local storage, such as scratch data. The disk variants of these instances have Amazon EC2 instance store volumes that use NVMe based SSD storage devices. Each instance type has a different amount of instance store available, and increases with the size of the instance type. For more information about the NVMe volumes for each instance, see the *Instance Storage* column on the Amazon EC2 Instance Types page.

## File systems

Currently, semiconductor design flows require a POSIX compliant file system. This requirement has traditionally been met with NFS file servers, that are built with third-party vendors and expensive licensing. Building your environment on AWS allows you to choose from multiple managed services that can be used for the entire design workflow, and reduce expensive licensing costs.

### Amazon FSx for Lustre

Amazon FSx for Lustre is a fully managed service that provides cost-effective, high-performance, scalable storage for compute workloads. With Amazon FSx for Lustre, you can leverage the rich feature sets and fast performance of a widely-used open source file system, while avoiding time-consuming administrative tasks like hardware provisioning, software configuration, patching, and backups. Amazon FSx for Lustre provides cost-efficient capacity and high levels of reliability, and it integrates with other AWS services so that you can manage and use the file systems across your entire design workflow built on AWS.

For semiconductor design workflows, we recommend using FSx for Lustre for, at minimum, testing and POCs. Management of the file system is part of the FSx for Lustre service. This eliminates the time consuming management overhead that is normally associated with high performance file systems. FSx for Lustre offers sub-millisecond latencies, up to hundreds of gigabytes per second of throughput, and millions of IOPS. FSx for Lustre file systems can also be linked to Amazon S3 buckets, allowing you to populate file systems when they are created, and subsequently push data into S3 on an as needed basis.

Specifically, physical design workloads are particularly well suited for FSx for Lustre. This includes static timing analysis (STA), extraction, and design rule checking (DRC). Front-end workloads can run on FSx for Lustre, but may see a scaling limit for millions of small files and metadata heavy I/O. We encourage our customers to run their entire flow on FSx for Lustre for testing and verification. From there, you can optimize by tuning FSx for Lustre, or potentially use another file system. For information about tuning, see Amazon FSx for Lustre Performance in the *Amazon FSx for Lustre User Guide*.

## Amazon Elastic File System (Amazon EFS)

Amazon Elastic File System (Amazon EFS) provides a simple, scalable, fully managed elastic NFS file system for use with AWS services and on-premises resources. Amazon EFS is built to scale on demand to petabytes without disrupting applications, growing and shrinking automatically as you add and remove files, eliminating the need to provision and manage capacity to accommodate growth.

For semiconductor design workflows, Amazon EFS can be used for multiple applications. Customers use Amazon EFS for home directories, infrastructure support (installation and configuration files), and application binaries. We recommend using EFS for large sequential I/O, as large amounts of small files and metadata heavy I/O may not perform at the required throughput and IOPS.

## Traditional NFS file systems

For EDA workflow migration to AWS, you should start with migrating data to an environment that is similar to your on-premises environment. This allows you to migrate applications quickly without having to rearchitect your workflow. Both FSx for Lustre and EFS provide POSIX compliant file systems, and your workflows should be compatible with either. If you still require a more traditional NFS server, you can create a storage server by launching an Amazon EC2 instance, attaching the appropriate EBS volumes, and sharing the file system to your compute instances using NFS.

If the data is temporary or scratch data, you can use an instance with locally attached (Amazon EC2 instance store) NVMe volumes to optimize the backend storage. For example, you can use the i3en.24xlarge that has 8 NVMe volumes (60 TB total) and is capable of up to 16 GB/s of sequential throughput and 2M IOPS for local access (using 4K block sizes). The 100 Gbps network connection to the i3en.24xlarge then becomes the rate limiting factor, and not the backend storage. This configuration results in an NFS server capable of over 10 GB/s. If you want to preserve the data stored on the NVMe volumes, you can attach EBS volumes and rsync the data to EBS, or you can copy the data to an Amazon S3 bucket.

For workloads that require more performance in aggregate than can be provided by a single instance, you can build multiple NFS servers that are delegated to specific mount points. Typically, this means that you build servers for shared scratch, tools directories, and individual projects. By building servers this way, you can right size the server and the storage allocated according to the demands of a specific workload. When projects are finished, you can archive the data to a low cost, long term storage service like Amazon S3 Glacier, and terminate the storage servers.

## Cloud Storage Approaches

Cloud-optimized semiconductor design workflows use a combination of Amazon FSx for Lustre, Amazon EFS, Amazon EBS, Amazon EC2 instance store, and Amazon S3 to achieve extreme scalability at very low costs, without being bottlenecked by traditional storage systems.

To take advantage of a solution like this, your EDA organization and your supporting IT teams might need to untangle many years of legacy tools, file system sprawl, and large numbers of symbolic links to understand what data you need for specific projects (or job deck) and pre-package the data along with the job that requires it. The typical first step in this approach is to separate out the static data (for example, application binaries, compilers, and so on) from dynamically changing data and IP in order to build a front-end workflow that doesn't require any shared file systems. This step is important for optimized cloud migration, and also provides the benefit of increasing the scalability and reliability of your workflows.

By using this less NFS centric approach to manage EDA storage, operating system images can be regularly updated with static assets so that they're available when the instance is launched. Then, when a job is dispatched to the instance, it can be configured to first download the dynamic data from Amazon S3 to local or Amazon EBS storage before launching the application. When complete, results are uploaded back to Amazon S3 to be aggregated and processed when all jobs are finished. This method for decoupling compute from storage can provide substantial performance and reliability benefits, in particular for front-end register transfer language (RTL) batch regressions.

# Orchestration

Orchestration refers to the dynamic management of compute and storage, as well as the management of individual jobs being processed in a complex workflow (scheduling and monitoring), for example during RTL regression testing or IP characterization. For these and many other typical chip design workflows, the efficient use of compute and storage resources—as well as the efficient use of software licenses—depends on having a well-orchestrated, well-architected batch computing environment.

Chip design workflows gain new levels of flexibility in the cloud, making resource and job orchestration an important consideration for your workload. AWS provides a range of solutions for workload orchestration. Describing all possible methods of orchestration is beyond the scope of this document; however, it is important to note that the same orchestration methods and job scheduling software used in typical, legacy chip design environments can also be used on AWS. For example, commercial and open-source job scheduling software can be migrated to AWS, and be enhanced by the addition of automatic scaling (for dynamic resizing of EDA clusters in response to demand or other triggers), AWS CloudWatch (for monitoring the compute environment, for example CPU utilization and server health), and other AWS services to increase performance and security, while reducing costs.

## AWS Solutions Implementation: Scale-Out Computing on AWS

The AWS Solutions Implementation Scale-Out Computing on AWS helps customers easily deploy and operate a multiuser environment for computationally intensive workflows such as Computer-Aided Engineering (CAE) and EDA workflows. The solution features a large selection of compute resources, a fast network backbone, flexible storage and file system options, and budget and cost management directly integrated. This solution deploys a web user interface (UI) with cloud workstations, file management, and automation tools that enable you to create your own queues, scheduler resources, Amazon Machine Images (AMIs), and management functions for user and group permissions.

The services and recommendations that are covered in this guide can be launched and customized using Scale-Out Computing on AWS.

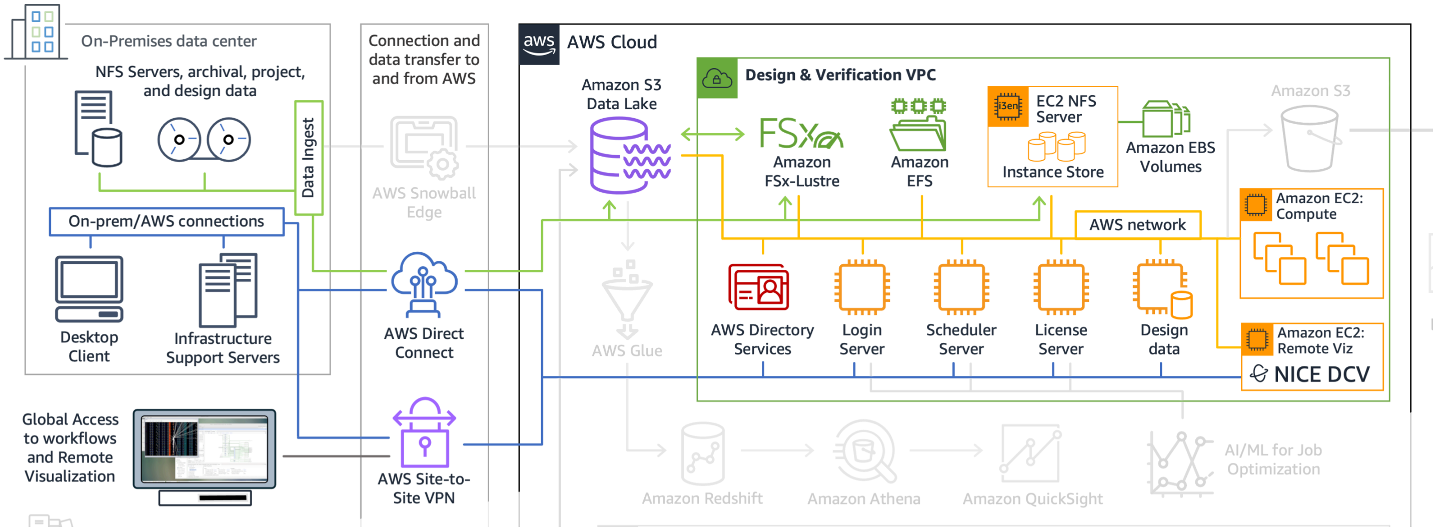
# Job scheduler integration

The semiconductor design workflow that you build on AWS can be a similar environment to the one you have in your on-premises data center. Many, if not all, of the same tools and applications running in your data center, as well as orchestration software, can also be run on AWS. Job schedulers, such as IBM Platform LSF, Altair PBS Pro, and Grid Engine (or their open source alternatives), are typically used in the semiconductor industry to manage compute resources, optimize license usage, and coordinate and prioritize jobs. When you migrate to AWS, you may choose to use these existing schedulers essentially unchanged, to minimize the impact on your end-user workflows and processes. Most of these job schedulers already have some form of cloud-optimized integration with AWS, allowing you to use the scheduler node to automatically launch cloud resources when there are jobs pending in the queue. Be sure to refer to the documentation of your specific job management tool for the steps to automate resource allocation and management on AWS.

## Launch and configure the entire semiconductor design workflow

With the combination of infrastructure already in place and the guidance from the previous section, you can now launch and configure the entire workflow, to include scaling out to 10,000s of cores. We recommend using the previously mentioned AWS Solutions Implementation Scale-Out Computing on AWS to launch your environment.

The following figure shows all of the necessary resources to run your entire semiconductor design workflow on AWS. The previous section provided the details and guidance for determining what compute, storage, file systems, and networking options are required. Using Scale-Out Computing on AWS, you can quickly (less than an hour) launch a turnkey solution that is capable of running jobs, monitoring queues, adding users, and many other features that are advantageous to chip design workflows.



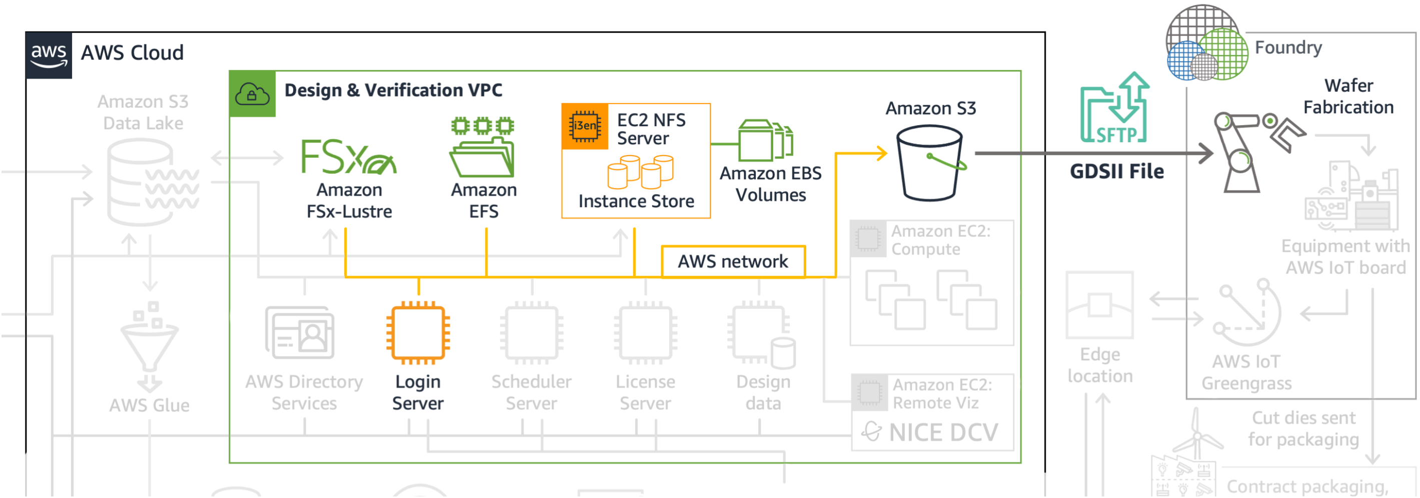
### Launch and configure the entire semiconductor design workflow

Building on the previously launched architecture used for your POC, the environment is now extended to handle additional storage options and scaling out compute resources to 10,000s of cores.

## Transfer GDSII file to foundry

With your workflow built on AWS, you can now do final sign-off and tape-out. You can transfer the GDSII file to the foundry in multiple ways, including legacy methods such as SFTP. Work with your foundry to enable secure and reliable methods of transferring the GDSII file over to the fab.

The following figure shows transferring the GDSII file over to the foundry using AWS Transfer for SFTP. The AWS Transfer Family provides fully managed support for file transfers directly into and out of Amazon S3 or Amazon EFS.

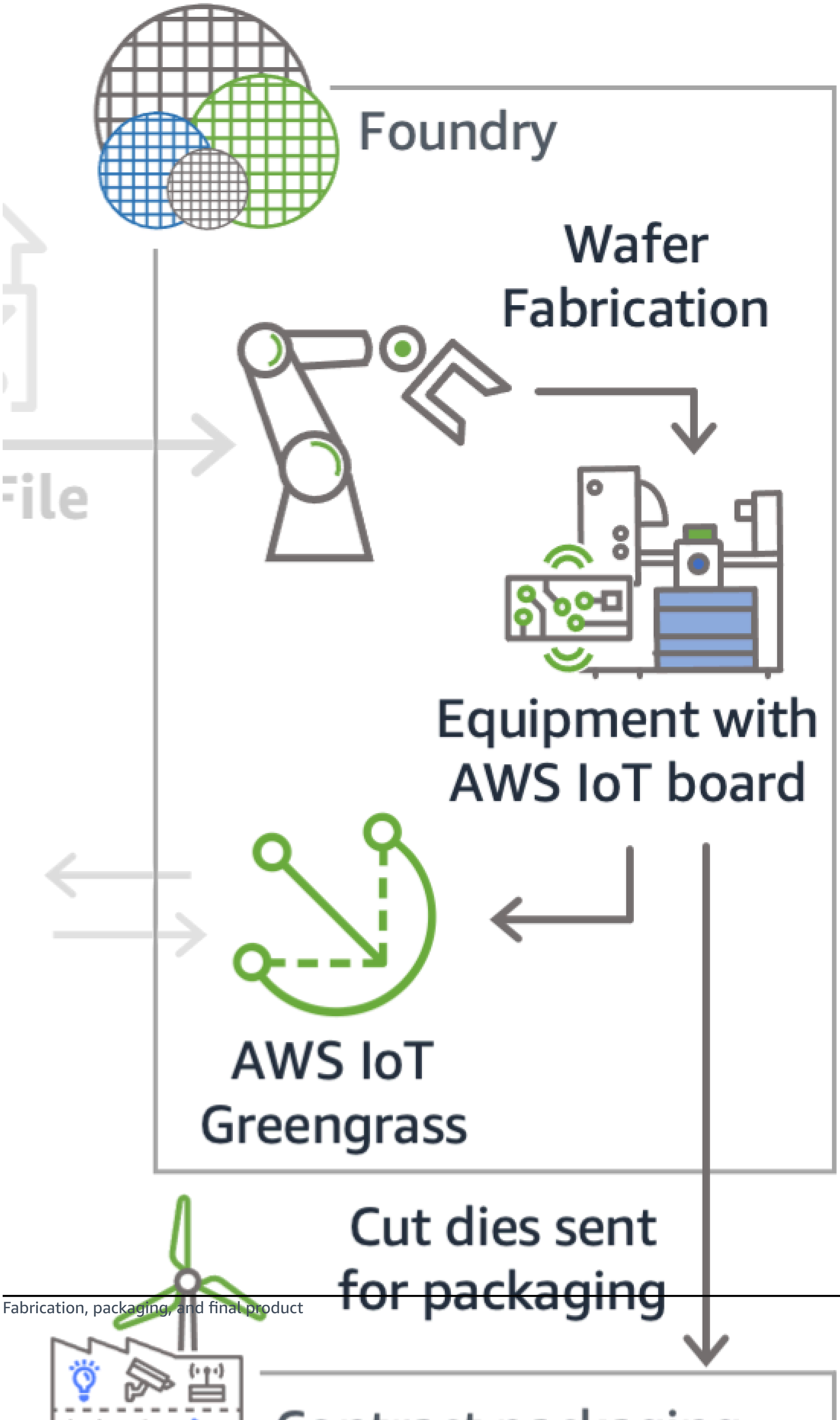


**Transfer GDSII file to foundry**

## Fabrication, packaging, and final product

Once the GDSII file is sent to the foundry, this initiates a process resulting in the delivery of the IC or device to the OEM or end user.

The following figure shows the process of fabricating the wafers, sending the cut dies for packaging, and sending the assembled product to the OEM or customer. Normally, the chip designer has little or no visibility for these processes. By introducing collaboration across the entire industry, time-to-market (TTM) can be reduced while increasing return on investment (ROI). The product of industry wide collaboration is comprehensive data collection of the entire workflow, from customer specification to products in the field. The data collected can then be analyzed using an analytics pipeline, entirely built on AWS. This pipeline results in insights and actions that can dramatically influence the way your teams design semiconductors.


Foundry
Wafer Fabrication
ile
Equipment with AWS IoT board
AWS IoT Greengrass
Cut dies sent for packaging

**Fabrication, packaging, and final product**

# Enable secure collaboration chambers with third parties

Across the entire semiconductor industry, the need for collaboration is part of the design process, fabrication, and product manufacturing. AWS allows you to securely collaborate with third-party IP providers, EDA tool vendors, foundries, and contract manufacturers. For example, you might have a requirement to work with a third-party IP provider or contract engineering team to create or validate a portion of your system-on-chip (SoC). Using AWS for collaboration makes it possible to segregate roles and data, lock down the environment to only authorized users, and monitor activity in the environment.

When trying to create similar collaborative environments in your on-premises data center, you might have the ability to isolate users and groups through existing network policies; however, you are still allowing external access to your internal infrastructure, and the collaboration environment is not scalable. On AWS, you can set up completely separate, secure, and scalable environments that allow you to isolate access to just what is needed for the collaborative effort. This approach can be accomplished in several ways on AWS, but typically starts with a separate Amazon Virtual Private Cloud (Amazon VPC) with specific security settings for the level of security and access required. For additional details on VPC settings, see the Security section.

This section includes three collaboration examples specific to the semiconductor industry:

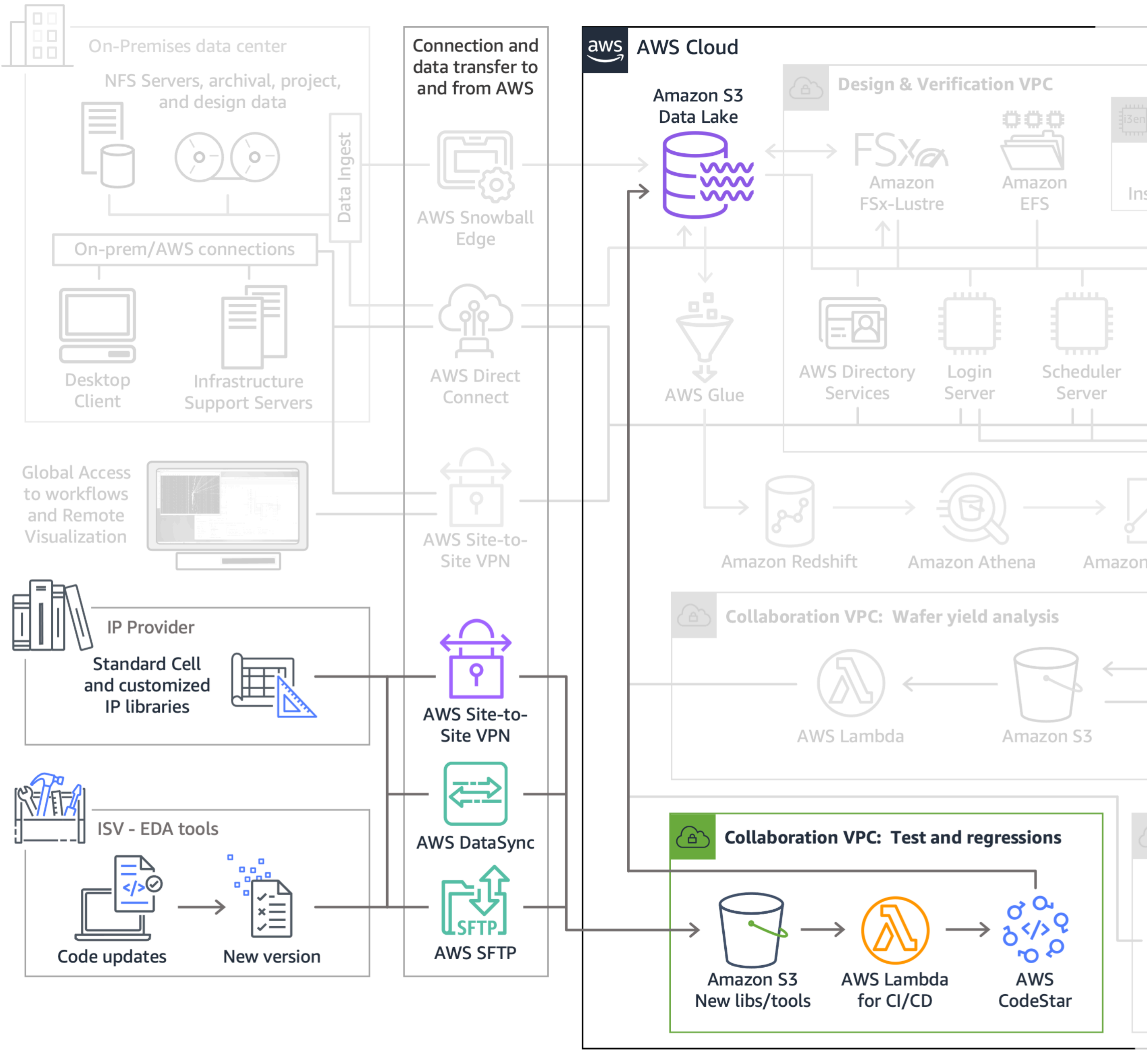
- Collaboration with IP providers and EDA tool vendors (ISVs)
- Collaboration with foundry
- Collaboration with packaging and contract manufactures

Each of these examples leverages a separate VPC to ensure a secure, isolated chamber that enables fine-grained control that restricts the environment to only the data and applications necessary for that specific project.

## Collaboration with IP providers and EDA tool vendors (ISVs)

From customer specifications to silicon, tool vendors and IP providers are a critical part of the entire workflow. Acquiring the latest version of tools and libraries is a manual process, that remains largely unmonitored and untracked.

The following figure shows a collaboration VPC for test and regressions.

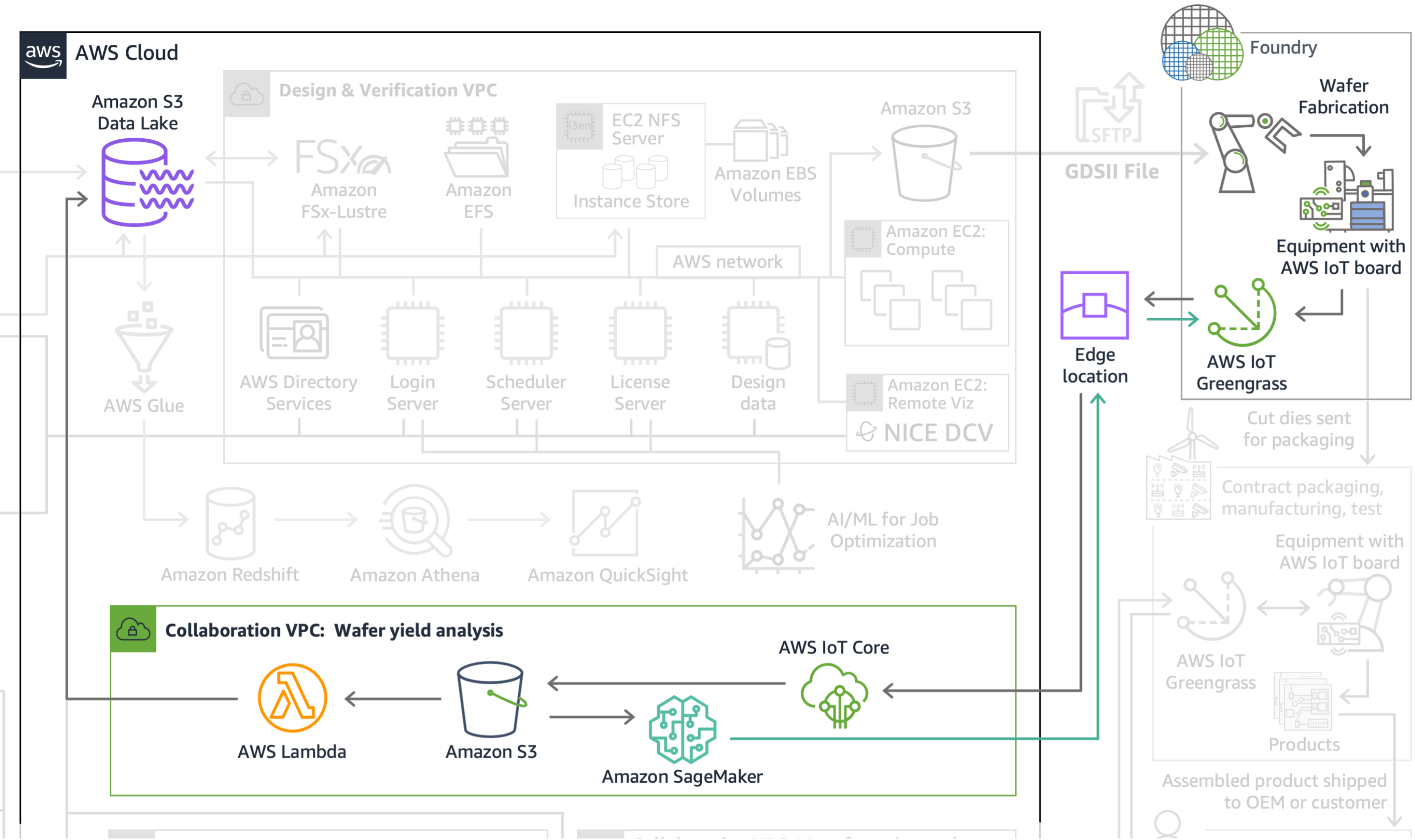


## Collaboration with IP providers and EDA tool vendors (ISVs)

In this figure, the collaboration VPC is set up to allow for inbound transfers from both the tool and IP providers. You can allow inbound transfers using any of the AWS Transfer Family services. This diagram shows AWS Site-to-Site VPN, AWS DataSync, and AWS Transfer for SFTP because these options are typically seen in the semiconductor industry. Once the tool or library is transferred to the Amazon S3 bucket that is in the test and regression VPC, this transfer triggers an AWS Lambda function that starts the continuous integration/continuous deployment (CI/CD). One potential example of this workflow is automating IP characterization. When an IP provider sends a new library, characterization is automatically triggered. Regardless of the specific use case (regressions, IP char, software build, and so on), the output data and results are captured and sent to your data lake. This approach ensures data is in the same place for your entire design environment.

## Collaboration with foundry

After sending your GDSII file to the foundry, the wafer fabrication process has traditionally been obfuscated from the chip design teams. Launching a separate VPC to enable collaboration with just your foundry can result in robust analytics, a reduction in time-to-market, and increased ROI. The following figure shows the wafer yield analysis from collaboration with your foundry.



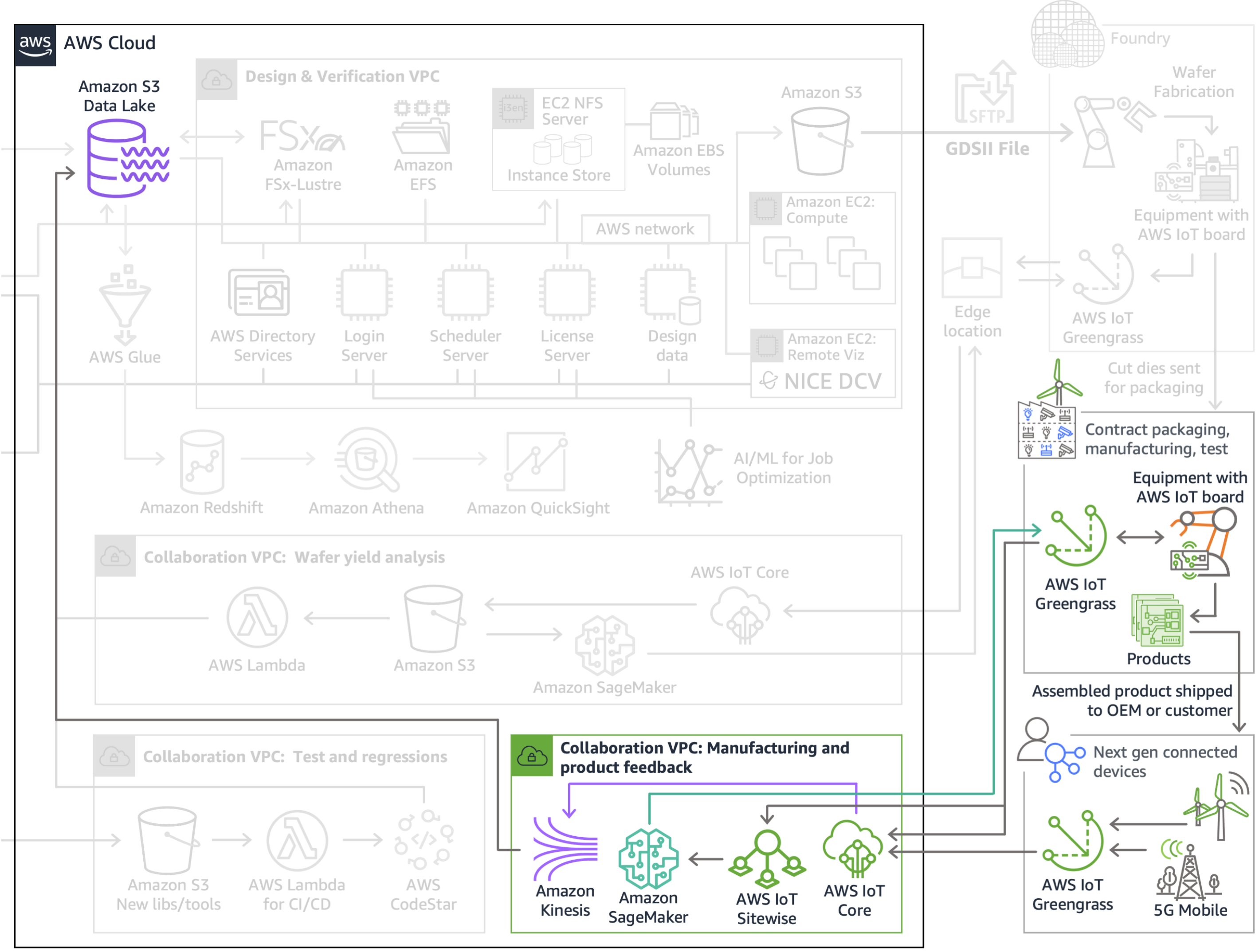
### Collaboration with foundry - wafer yield analysis

As shown in the preceding figure, collaboration with your foundry starts with data collection from an AWS IoT board that is installed in the on-premises foundry equipment. The IoT board sends data to AWS IoT Greengrass. Using an Edge location, the data is sent to AWS IoT Core located inside the collaboration VPC. In this diagram, the data is used for wafer yield analysis, which should lead to increased yields at the foundry, and help determine if design changes would result in less defects. AWS IoT Greengrass makes it easy to perform machine learning inference locally on devices (located in the foundry), using models that are created, trained, and optimized in the cloud. IoT AWS IoT Greengrass gives you the flexibility to use machine learning models trained in Amazon SageMaker AI or to bring your own pre-trained model stored in Amazon S3.

Similar to the collaborative efforts with the IP providers and EDA tool vendors, the resulting wafer data is sent to the same data lake used for the entire semiconductor design workflow.

## Collaboration with packaging and contract manufacturers

Similar to the way collaboration is enabled with the foundry, you can also enable collaboration with your packaging and contract manufacturers, as well as the devices in the field. The following figure shows the workflow for collaboration with packaging and contract manufacturers.

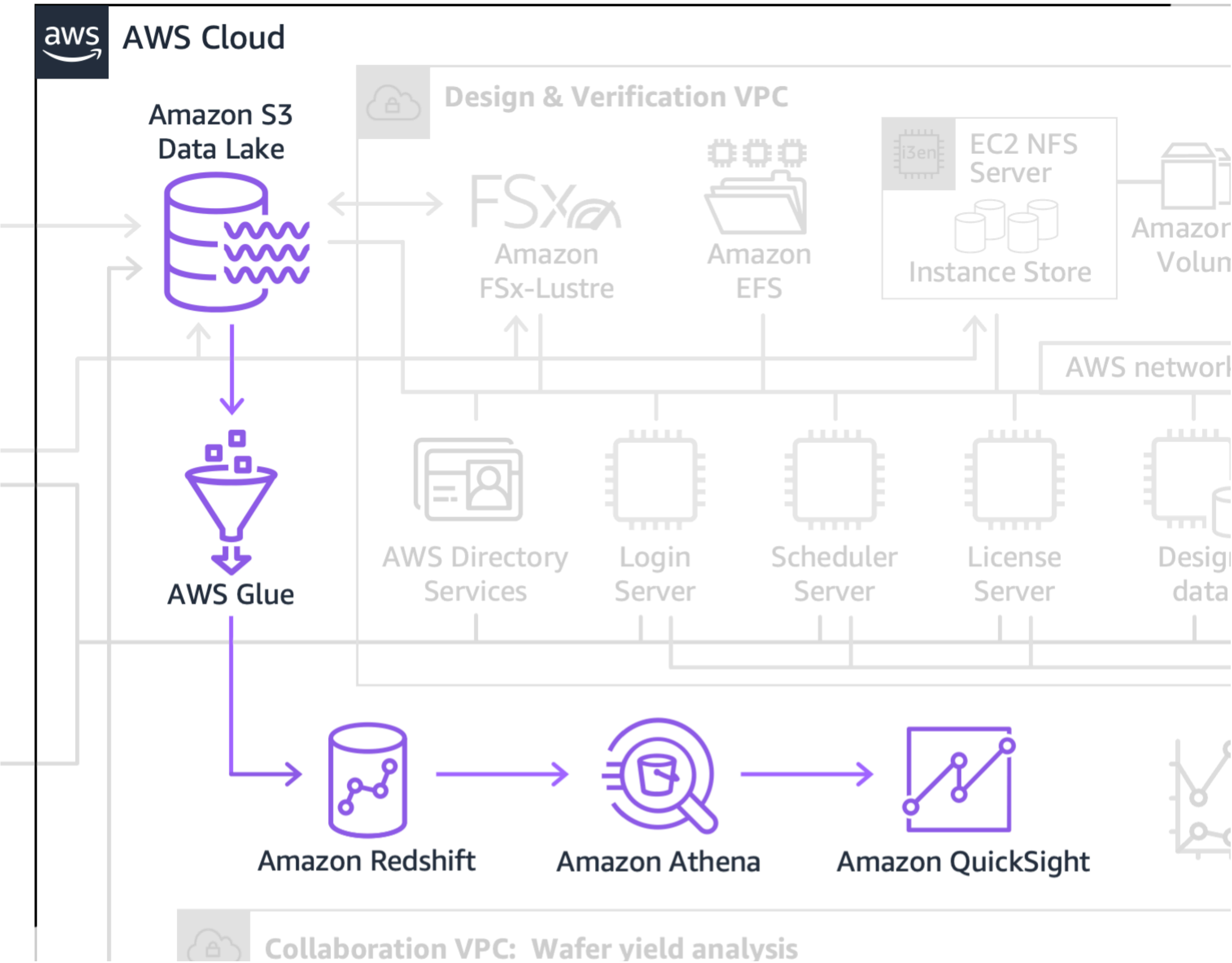


### Collaboration with packaging and contract manufacturers

In the preceding figure, data is sent to both AWS IoT Core and AWS IoT SiteWise using an AWS IoT board that is installed in the on-premises manufacturing equipment. AWS IoT SiteWise makes it easy to collect, store, organize, and monitor data from industrial equipment at scale to help you make better, data-driven decisions. From there, machine learning models trained in Amazon SageMaker AI provide real-time inference on the manufacturing floor. Additionally, all incoming data is sent to Amazon Kinesis to stream data to the same data lake that is used throughout the entire environment.

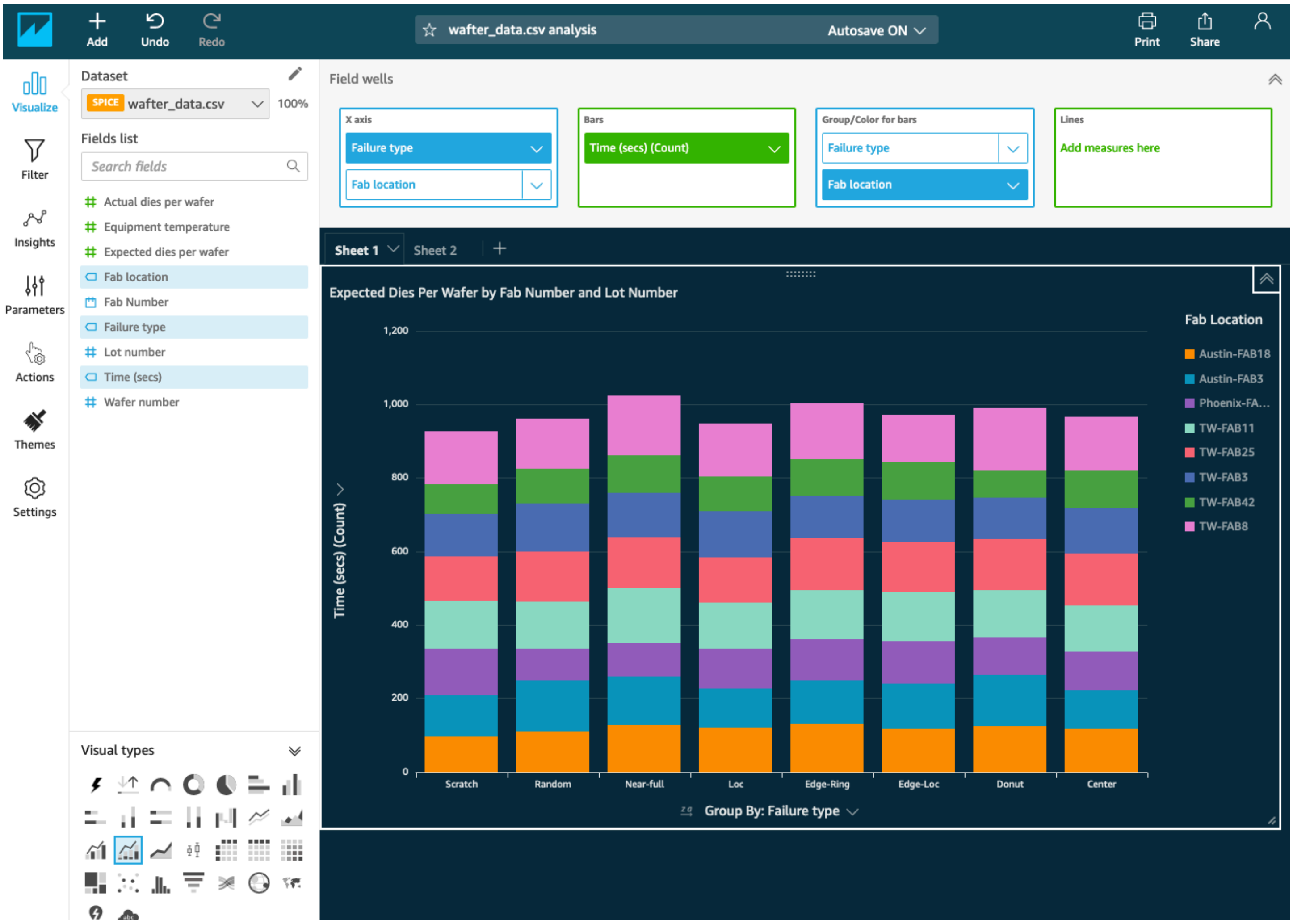
## AWS analytics pipeline and leveraging your data lake

This guide has emphasized moving data in to your data lake that is built on an Amazon S3 bucket. Next, you learn about what you can do with the data you have collected. Archival data can be an invaluable source when performing analytics and model training for AI/ML. The archival data provides historical comparison to accurately determine trends, and provide helpful insights. For more information, see the AI/ML for workflow optimization section.



## Analytics pipeline and leveraging your data lake

Starting with all of the data that you have collected, both current and archival, you use several AWS managed services, to include visualizing your data with an Amazon QuickSight dashboard. For example, a QuickSight dashboard can help you analyze wafer data that is sent over from the foundry and provide valuable insights into future designs. The following figure shows a QuickSight dashboard analyzing wafer defect data.



X axis: Failure type, Fab location

Bars: Time (secs) (Count)

Group/Color for bars: Failure type, Fab location

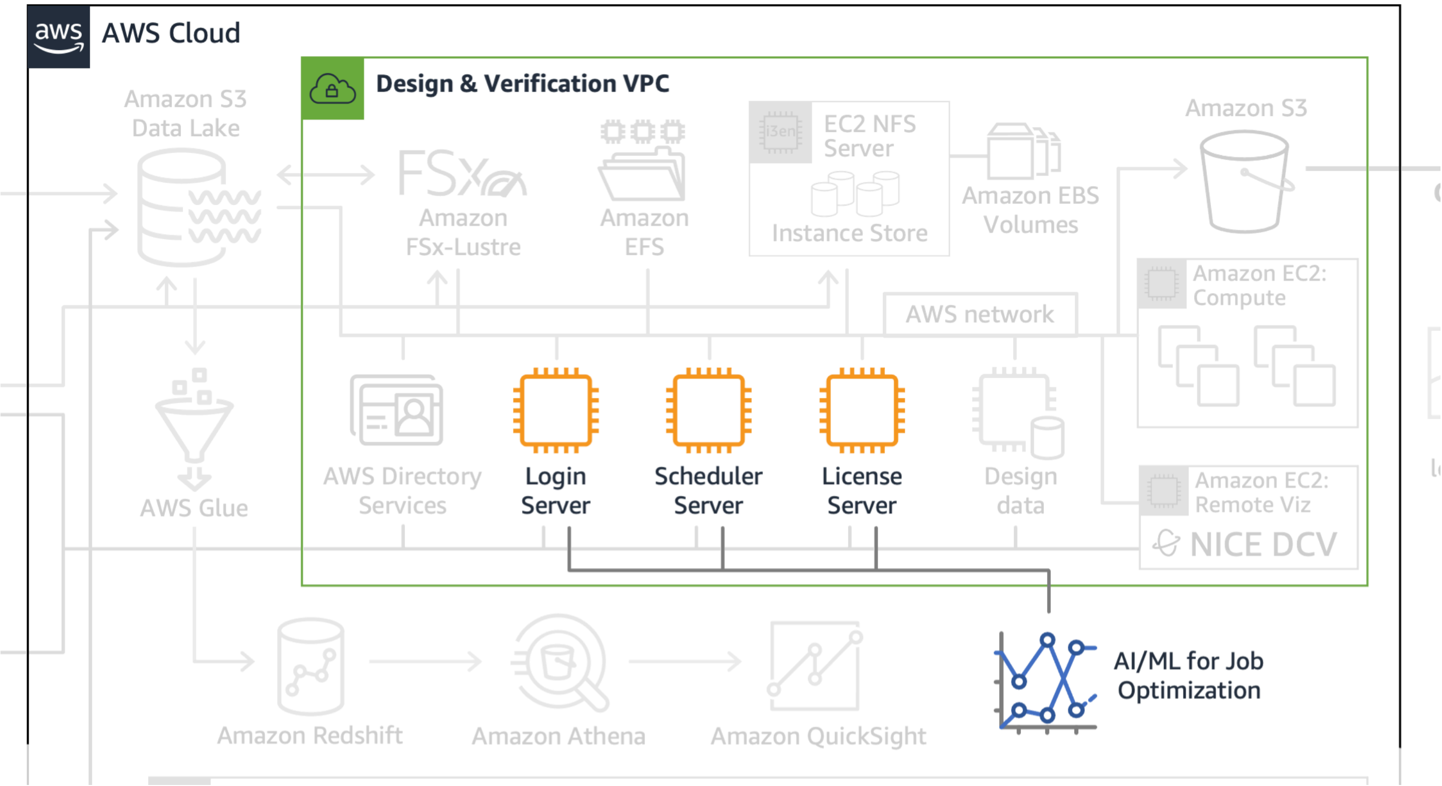
Lines: Add measures here

**QuickSight dashboard showing wafer defect types and foundry location**

In this example, the dashboard shows the defect type and the location of the fab where the defect occurred. From this data, it appears that defects are occurring about the same rate and type across all of the fabs. If, for example, there was an increase in defects at a specific fab when fabrication started on your design, this may indicate a problem with the PDK or another design issue. With this insight, you can make decisions about designs in-flight, that could result in reduced re-spins and increased fabrication yields.

# AI/ML for workflow optimization

Using AWS artificial intelligence/machine learning (AI/ML) services, you can easily train models and do real-time inference in the cloud or on-premises. In the semiconductor industry, an example is optimizing job queues and license usage. The following figure shows a simple workflow for optimization using AI/ML.

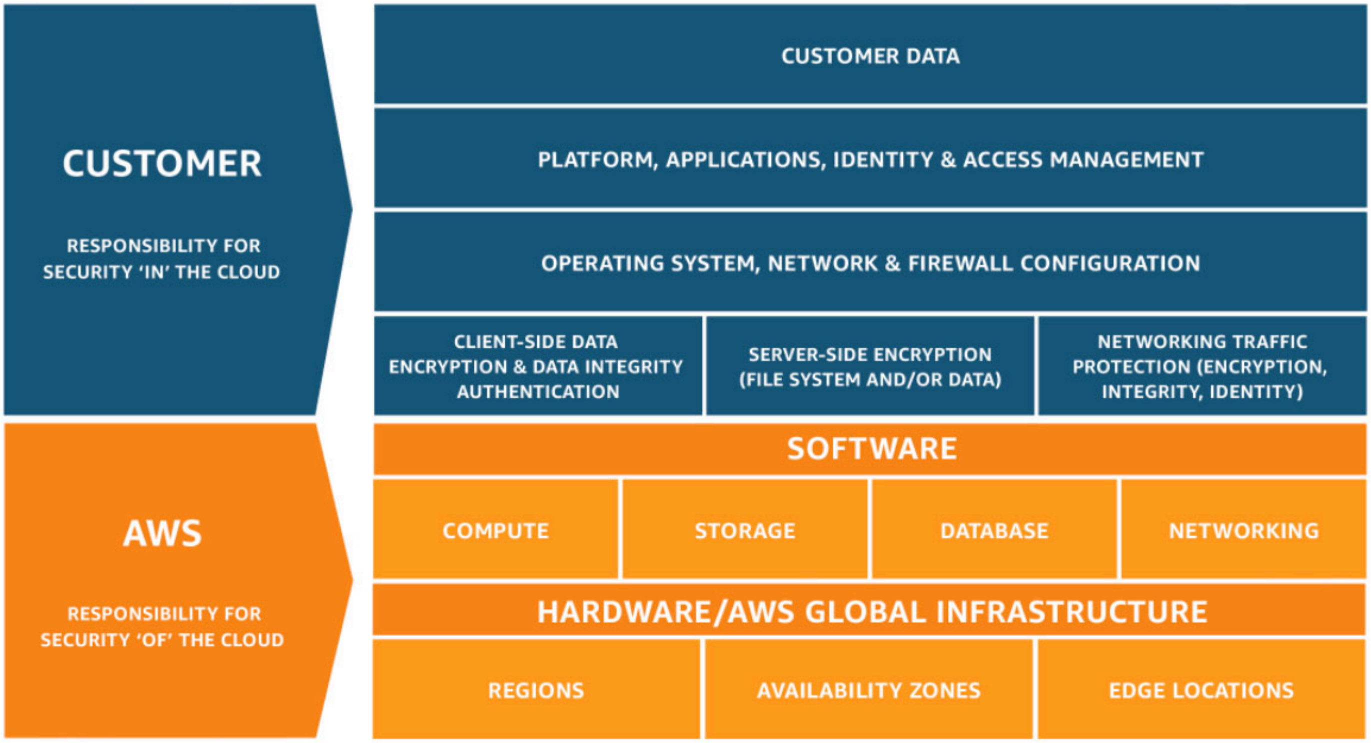


## AI/ML for workflow optimization

This example uses multiple data sources for model training. For example, from the login server, you can collect when users are active, what data (file system level) they are using, and any other relevant user activities. When the user submits a job to the scheduler server, you can scan the submit script and collect job and runtime configurations. These configurations include system resources requested (compute, memory, and so on), file system usage, expected job runtime, which tools will be used, and so on. You also have data coming in from the license server(s), so you know exactly which licenses were used for which job and for how long. With this data, you can build a model that predicts if the license will be fully utilized. As the model is trained over time (or historical data is used), inference is performed to alter the job runtimes to only the time needed to complete the job, thereby reducing or eliminating unused license time. This inference further results in cost savings and license optimization.

# Security

Security and Compliance is a shared responsibility between AWS and the customer. This shared model can help relieve the customer's operational burden as AWS operates, manages and controls the components from the host operating system and virtualization layer down to the physical security of the facilities in which the service operates. The customer assumes responsibility and management of the guest operating system (including updates and security patches), other associated application software as well as the configuration of the AWS provided security group firewall. Customers should carefully consider the services they choose as their responsibilities vary depending on the services used, the integration of those services into their IT environment, and applicable laws and regulations. The nature of this shared responsibility also provides the flexibility and customer control that permits the deployment. As shown in the following chart, this differentiation of responsibility is commonly referred to as Security "of" the Cloud versus Security "in" the Cloud. For more information, see Shared Responsibility Model.


CUSTOMER
RESPONSIBILITY FOR SECURITY 'IN' THE CLOUD
CUSTOMER DATA
PLATFORM, APPLICATIONS, IDENTITY & ACCESS MANAGEMENT
OPERATING SYSTEM, NETWORK & FIREWALL CONFIGURATION
CLIENT-SIDE DATA ENCRYPTION & DATA INTEGRITY AUTHENTICATION
SERVER-SIDE ENCRYPTION (FILE SYSTEM AND/OR DATA)
NETWORKING TRAFFIC PROTECTION (ENCRYPTION, INTEGRITY, IDENTITY)
AWS
RESPONSIBILITY FOR SECURITY 'OF' THE CLOUD
SOFTWARE
COMPUTE
STORAGE
DATABASE
NETWORKING
HARDWARE/AWS GLOBAL INFRASTRUCTURE
REGIONS
AVAILABILITY ZONES
EDGE LOCATIONS

### Security "of" the Cloud versus Security "in" the Cloud

AWS offers a wide array of tools and configurations that enable your organization to protect your data and IP in ways that are difficult to achieve with traditional on-premises environments. The following sections outline a few of the ways you can protect users, data, and network connections.

# User authentication

When managing users and access to compute nodes, you can adapt the technologies that you use today to work in the same way on AWS. Many organizations already have existing LDAP, Microsoft Active Directory, or Network Information System (NIS) services that they use for authentication. Almost all of these services provide replication and functionality to support multiple data centers. With the appropriate network and VPN setup in place, you can manage these systems on AWS using the same methods and configurations as you do for any remote data center configuration.

If your organization wants to run an isolated directory on the cloud, you have a number of options to choose from. If you want to use a managed solution, AWS Directory Service for Microsoft Active Directory (Standard) is a popular choice. AWS Managed Microsoft AD (Standard Edition) is a managed Microsoft Active Directory (AD) that is optimized for small and midsize businesses (SMBs).

# Network

AWS employs a number of technologies that allow you to isolate components from each other and control access to the network, including Amazon VPC and security groups.

## Amazon VPC

Amazon Virtual Private Cloud (Amazon VPC) is a service that lets you launch AWS resources in a logically isolated virtual network that you define. 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 use both IPv4 and IPv6 for most resources in your virtual private cloud, helping to ensure secure and easy access to resources and applications.

You can easily customize the network configuration for your Amazon VPC. For example, you can create a public-facing subnet for your FTP and Bastion servers that has access to the internet. Then, you can place your design and engineering systems in a private subnet with no internet access. You can leverage multiple layers of security, including security groups and network access control lists, to help control access to EC2 instances in each subnet.

Additionally, you can create a hardware virtual private network (VPN) connection between your corporate data center and your VPC and leverage the AWS Cloud as an extension of your organization's data center.

## Security groups

Amazon VPC provides advanced security features such as security groups and network access control lists to enable inbound and outbound filtering at the instance level and subnet level, respectively. A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. When you launch an instance in a VPC, you can assign the instance to up to five security groups.

Network access control lists (ACLs) control inbound and outbound traffic for your subnets. In most cases, security groups can meet your needs. However, you can also use network ACLs if you want an additional layer of security for your VPC. For more information, see Security in Amazon Virtual Private Cloud in the *Amazon Virtual Private Cloud User Guide*.

You can create a flow log on your Amazon VPC or subnet to capture the traffic that flows to and from the network interfaces in your VPC or subnet. You can also create a flow log on an individual network interface. Flow logs are published to Amazon CloudWatch Logs. For more details, see VPC Flow Logs.

# Data storage and transfer

AWS offers many ways to protect data, both in transit and at rest. Many third-party storage vendors also offer additional encryption and security technologies in their own implementations of storage in the AWS Cloud.

## AWS Key Management Service (KMS)

AWS Key Management Service (AWS KMS) makes it easy for you to create and manage cryptographic keys and control their use across a wide range of AWS services and in your applications. AWS KMS is a secure and resilient service that uses hardware security modules that have been validated under FIPS 140-2, or are in the process of being validated, to protect your keys. AWS KMS is integrated with AWS CloudTrail to provide you with logs of all key usage to help meet your regulatory and compliance needs.

## Amazon EBS Encryption

Use Amazon EBS encryption as a straight-forward encryption solution for your EBS resources associated with your EC2 instances. With Amazon EBS encryption, you aren't required to build, maintain, and secure your own key management infrastructure. Amazon EBS encryption uses AWS

Key Management Service (AWS KMS) keys (formerly CMKs) when creating encrypted volumes and snapshots.

Encryption operations occur on the servers that host EC2 instances, ensuring the security of both data-at-rest and data-in-transit between an instance and its attached EBS storage.

You can attach both encrypted and unencrypted volumes to an instance simultaneously.

## EC2 Instance Store Encryption

The data on NVMe instance store volumes is encrypted using an XTS-AES-256 cipher implemented on a hardware module on the instance. The encryption keys are generated using the hardware module and are unique to each NVMe instance storage device. All encryption keys are destroyed when the instance is stopped or terminated and cannot be recovered. You cannot disable this encryption and you cannot provide your own encryption key. For more information, see Data protection in Amazon EC2 in the *Amazon Elastic Compute Cloud User Guide for Linux Instances*.

## Amazon S3 Encryption

Data protection refers to protecting data while in-transit (as it travels to and from Amazon S3) and at rest (while it is stored on disks in Amazon S3 data centers). You can protect data in transit using Secure Socket Layer/Transport Layer Security (SSL/TLS) or client-side encryption. You have the following options for protecting data at rest in Amazon S3:

- **Server-Side Encryption** – Request Amazon S3 to encrypt your object before saving it on disks in its data centers and then decrypt it when you download the objects.
- **Client-Side Encryption** – Encrypt data client-side and upload the encrypted data to Amazon S3. In this case, you manage the encryption process, the encryption keys, and related tools.

For more information about encryption on Amazon S3, see Protecting data using encryption in the *Amazon Simple Storage Service User Guide.*

# Management and Governance

Management and Governance on AWS provides the services needed to implement the security methods described in the preceding section, as well as many other security options.

For a detailed list and description of each service, see Management and Governance on AWS.

# Cost optimization

The intention of this guide is to focus on the enablement and building of your workflows on AWS. However, pricing is an important component when deciding to move production from on-premises to AWS. In a like-for-like comparison, we encourage you to first analyze the results of on-premises jobs as compared to the jobs run on AWS. That is, when using nearly the same resources, did the job runs on AWS meet or exceed your expectations of the job runs performed on-premises? The next steps are to perform workload and cost optimization on your entire workflow. These steps can mean choosing different instance types, storage options, serverless architectures, and of course leveraging the breadth of services that AWS has to offer.

## Amazon EC2 Spot Instances

Amazon EC2 Spot Instances let you take advantage of unused EC2 capacity in the AWS Cloud. Spot Instances are available at up to a 90% discount compared to On-Demand prices. You can use Spot Instances for various stateless, fault-tolerant, or flexible applications such as big data, containerized workloads, CI/CD, web servers, high-performance computing (HPC), and test & development workloads.

In the semiconductor industry, you can also use EC2 Spot instances to reduce cost. One specific usage is for library characterization. As the workloads run on EC2 Spot instances need to be fault-tolerant, if a library characterization job were to be interrupted, the job can be immediately restarted, and the time lost is minimal. The cost savings when using EC2 Spot instances far outweighs the time lost due to interruptions. For information, see the **Savings over On-Demand** and the **Frequency of Interruption columns in the** Spot Instance Advisor.

# Building your environment on AWS

The best way to get started building your environment on AWS is to use the AWS Solutions Implementation Scale-Out Computing on AWS to launch your environment. AWS has many resources available to use on your own, but we suggest working with an AWS Solutions Architect to launch a test environment using Scale-Out Computing on AWS.

# Conclusion

Semiconductor design teams worldwide are benefiting from the agility, performance, and security of cloud for their most critical semiconductor design workflows. AWS can significantly accelerate product development lifecycles and reduce time to market by providing near infinite compute and storage. In addition, AWS enables secure, fast-to-deploy environments for design collaboration. Other advantages include access to a wide range of analytics capabilities, including AI services (for example, yield/failure analysis), and EDA workflow optimization.

# Additional Resources

See the following additional resources to help you get started running your semiconductor workloads on AWS.

- For an overview of semiconductor, see the Semiconductor Design on AWS whitepaper.
- For a comprehensive list of resources, see Semiconductor and Electronics on AWS Resources
- For infrastructure setup and automation, see AWS Solutions Implementation Scale-Out Computing on AWS.
- For addition information for how AWS can help run your workflows, see Semiconductor and Electronics on AWS.
- For a hands-on workshop, contact your Account Manager or Solutions Architect. AWS offers hands-on workshops that go through the process of launching your environment on AWS and running a test workload in the same day.

# Contributors

Contributors to this document include:

- Mark Duffield, Worldwide Tech Leader, Semiconductors, Amazon Web Services
- Matt Morris, Senior HPC Solutions Architect, Amazon Web Services
- David Pellerin, Principal Business Development for Infotech/Semiconductor, Amazon Web Services
- Nafea Bshara, VP/Distinguished Engineer, Amazon Web Services

# Document history

To be notified about updates to this whitepaper, subscribe to the RSS feed.

| Change | Description | Date |
|---|---|---|
| Minor update | Fix non-inclusive language. | April 6, 2022 |
| Initial release | Guide first published. | March 12, 2021 |

# Notices

Customers are responsible for making their own independent assessment of the information in this document. This document: (a) is for informational purposes only, (b) represents current AWS product offerings and practices, which are subject to change without notice, and (c) does not create any commitments or assurances from AWS and its affiliates, suppliers or licensors. AWS products or services are provided “as is” without warranties, representations, or conditions of any kind, whether express or implied. The responsibilities and liabilities of AWS to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.