

Amazon AppStream 2.0: PTC Creo Deployment Guide

Build an Amazon AppStream 2.0 environment to stream PTC Creo to your users

July 2020

<https://aws.amazon.com/appstream2/>

This paper is out of date and has been archived

This is in the process of being updated. Until then, for the latest technical content, contact your AWS Account Manager

Archived

Welcome

This guide describes how to deploy and stream PTC Creo for your company by using [Amazon AppStream 2.0](#), a fully managed, secure application streaming service that runs in the AWS Cloud.

What you'll accomplish:

- Provision an Amazon virtual private cloud ([Amazon VPC](#)) to provide an isolated virtual network infrastructure within the AWS Cloud. Your AppStream 2.0 resources will use this environment.
- Use the [AWS Management Console](#) to perform the basic administrative tasks required to build an AppStream 2.0 environment. Specifically, you'll:
 1. Create the AWS network resources
 2. Create and install Creo License Manager on an EC2 instance
 3. Install and configure Creo for streaming using an image builder.
 4. Provision a fleet of instances to stream your applications. The fleet will use g4dn instance type and adjust scaling policies to match the number of users that you want to be able to stream concurrently.
 5. Provision a stack to create a web portal from which users can stream your applications.
 6. Configure persistent storage that users can access across application streaming sessions.
 7. Create a user pool to manage users who access your streaming applications.

What you need before starting:

- **An AWS account:** You need an AWS account to use Amazon AppStream 2.0 and other AWS services, namely, VPC, EC2, and S3.
- **A current email address:** During the user configuration process for your Amazon AppStream 2.0 environment, AWS sends you two emails. You must use these emails to complete the process.

- **Skill level:** You do not need prior experience with AWS to complete these exercises. A basic understanding of desktop computing is helpful but not required.
- **An active Creo license.** This license is required for the Creo that you want to import into Amazon AppStream 2.0. Contact your PTC or PTC sales team for more information.
- **Creo Requirements:** The hardware and software requirements for Creo are available on the Creo [website](#).

To learn more about the number of vCPU cores, RAM memory specifications of these instance families, see [Amazon AppStream 2.0 pricing](#)

Creo is a market-leading suite of desktop CAD/CAM/CAE tools for 3D product design and development. Now a days with the new normal being running operations remotely, companies want to make it easy for their design engineers to access Creo from remote locations to perform their design activities securely. Streaming allows a fast and fluid experience at a fraction of a cost of running individual graphics workstations for individual users, without upfront investments or long-term commitments. Now you can stream Creo using Amazon AppStream 2.0 which runs Graphics g4dn instances that are powered by NVIDIA T4 Tensor Core GPU and NVIDIA Quadro technology.

Amazon AppStream 2.0 service delivers a workstation class experience for your users which is indistinguishable from their local graphics workstation while also reducing your operating costs. You can provide a Computer-aided Design (CAD) desktop application to your design engineers via just a web browser securely with low latency. Your CAD workload becomes accessible from anywhere in the world without the need to maintain individual desktop application or any other on-prem infrastructure. This also provides ability to scale to virtually unlimited number of additional global users. This deployment guide outlines all the steps necessary to create this end-to-end solution.

Table of Contents

| | |
|---|-----------|
| WELCOME | 2 |
| OVERVIEW OF SOLUTION | 5 |
| WALKTHROUGH | 5 |
| STEP 1: CREATE AWS NETWORK RESOURCES | 6 |
| STEP 2: CREATE EC2 INSTANCES TO HOST CREO LICENSE MANAGER & BASTION HOST | 7 |
| CONFIGURE AN EC2 INSTANCE FOR CREO LICENSE MANAGER | 7 |
| CONFIGURE AN EC2 INSTANCE FOR BASTION HOST TO ENABLE REMOTE LICENSE SERVER ADMINISTRATION | 7 |
| STEP 3: INSTALL AND CONFIGURE CREO LICENSE MANAGER | 8 |
| STEP 4: CREATE AN AMAZON APPSTREAM 2.0 IMAGE WITH CREO INSTALLED | 11 |
| STEP 5: CREATE AN AMAZON APPSTREAM 2.0 FLEET AND STACK | 13 |
| STEP 6: INVITE USERS TO ACCESS CREO THROUGH A WEB BROWSER | 14 |
| CLEANING UP | 16 |
| APPENDIX A. MANUALLY CREATE AND CONFIGURE NETWORK RESOURCES | 16 |
| APPSTREAM VPC REQUIREMENTS..... | 17 |
| ALLOCATE AN ELASTIC IP ADDRESS | 17 |
| CREATE A VPC BY USING THE VPC WIZARD | 17 |
| ADD A SECOND PRIVATE SUBNET | 19 |
| MODIFY THE SUBNET ROUTE TABLES | 19 |
| This paper is out of date and has been archived | |
| APPENDIX C. CLEAN UP YOUR APPSTREAM 2.0 RESOURCES | 22 |
| STOP AND DELETE YOUR IMAGE BUILD | 22 |
| REVOKE STACK PERMISSIONS FOR USERS IN THE USER POOL | 22 |
| DISASSOCIATE YOUR FLEETS FROM YOUR STACK AND DELETE YOUR STACK | 22 |
| STOP AND DELETE YOUR FLEET | 22 |

Overview of solution

In this deployment guide, we will walk you through steps that would be involved to build an Amazon AppStream 2.0 based Creo environment. Specifically,

- You'll install and configure Creo for streaming using an image builder. You'll also provision a fleet of instances to stream your applications
- The fleet will use the g4dn instance type and adhere to scaling policies to match the number of users that you want to be able to stream concurrently
- Provision a stack to create a web portal from which users can stream your applications
- Configure persistent storage that users can access across application streaming sessions
- Create a user pool to manage users who access your streaming applications

After completion of this exercise, your workload can be described as the figure below:

This paper is out of date and has been archived

This is in the process of being updated. Until then, for the latest technical content, contact your AWS Account Manager

Walkthrough

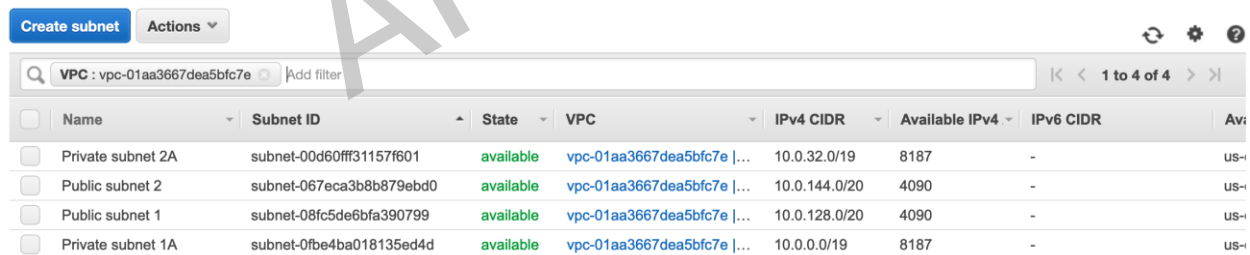
There are six main steps that we will go through summarized as below:

1. Create the AWS network resources
2. Create two EC2 instances, one will serve as a Creo license manager and another as Bastion host
3. Install and configure Creo License Manager
4. Create an Amazon AppStream 2.0 image with Creo installed
5. Create an Amazon AppStream 2.0 fleet and stack. This stack will be configured with persistent storage to S3.
6. Invite users to access Creo via Amazon AppStream 2.0

Step 1: Create AWS network resources

In this section, you will create an Amazon VPC and the other network resources. You can either create network resources using AWS console manually or can leverage AWS Cloud Formation template available at AWS QuickStart platform. This Quick Start reference deployment guide for VPC includes architectural considerations and configuration steps for deploying a modular Amazon Virtual Private Cloud (Amazon VPC) environment on the Amazon Web Services (AWS) Cloud. It also provides links for viewing and launching AWS CloudFormation templates that automate the deployment. In Step2 of this walkthrough we will add other infrastructure components and software layers on top of VPC to complete Creo AppStream 2.0 environment.

Follow the steps listed in the AWS QuickStart user deployment guide for VPC. After executing the CloudFormation Stack, you will have VPC constructs available that can be used to execute this walkthrough. If you set with parameter value for “Number of Availability Zones” set to 2, your subnets may look like the figure below. For this walkthrough you would be only be working with one public subnet and two private subnets in different Availability Zones.



| <input type="checkbox"/> | Name | Subnet ID | State | VPC | IPv4 CIDR | Available IPv4 | IPv6 CIDR | Avi |
|--------------------------|-------------------|--------------------------|-----------|----------------------------|---------------|----------------|-----------|-----|
| <input type="checkbox"/> | Private subnet 2A | subnet-00d60ff31157f601 | available | vpc-01aa3667dea5bfc7e ... | 10.0.32.0/19 | 8187 | - | us- |
| <input type="checkbox"/> | Public subnet 2 | subnet-067eca3b8b879ebd0 | available | vpc-01aa3667dea5bfc7e ... | 10.0.144.0/20 | 4090 | - | us- |
| <input type="checkbox"/> | Public subnet 1 | subnet-08fc5de6bfa390799 | available | vpc-01aa3667dea5bfc7e ... | 10.0.128.0/20 | 4090 | - | us- |
| <input type="checkbox"/> | Private subnet 1A | subnet-0fbc4ba018135ed4d | available | vpc-01aa3667dea5bfc7e ... | 10.0.0.0/19 | 8187 | - | us- |

To manually create and configure network resources, see *Appendix A*.

Step 2: Create EC2 instances to host Creo License Manager & Bastion Host

In this step, we will deploy an EC2 instance that will serve as Creo License Manager. Once the Creo application is installed and configured, your Amazon AppStream 2.0 streaming instances will contact this license server for license activation.

Configure an EC2 Instance for Creo License Manager

1. Open the AWS EC2 console and launch an instance
2. Select Microsoft Windows Server 2019 Base image
3. Choose the instance type as t2.medium and click Next: Configure Instance Details
4. Choose the following
 - a) Network – Quick-Start-VPC
 - b) Subnet – Private Subnet 1A
5. Leave the storage at its default values and select 'Review and Launch'
6. Select 'Create a new key pair' provide a name and click 'Download key pair'
7. Click 'Launch Instances' This is in the process of being updated. Until then, for the latest technical content, contact your AWS Account Manager
8. Now set the security group to allow traffic through certain ports
 - a) Select the instance that was created. On the Description tab, click the name of the security group that is associated with the instance.
 - b) Add Rule (This rule allows Creo applications to communicate with the Creo license server over TCP ports 7788 and 7799):
Type: Custom TCP
Port Range: 7788, 7799
Source: Custom: CIDR of your VPC (e.g. 10.0.0.0/16)

Configure an EC2 instance for Bastion host to enable remote license server administration

1. Repeat the processes 1-5, except choose the public subnet (Public Subnet1) and make sure that 'Enable for Auto-assign Public IP' is set to Enable.

2. Once you select 'Review and launch', select 'existing key pair' and choose the one that you created in step 5 above for the license manager EC2 instance
3. Now enable RDP connection to the bastion host
 - a) Select the instance that was created. On the Description tab, click the name of the security group that is associated with the instance.
 - b) Copy the name of the security group into your clipboard
 - c) Add rule for RDP connection to the bastion host

Type: RDP

Source: Custom: <Specify IP Address with CIDR block> (This would enable the AppStream environment creator/administrator to RDP into the bastion host from a particular IP address)
4. Enable RCP connection from the bastion host to the designated Creo license manger
 - a) Select the Creo License Manager server that you launched above. On the Description tab, click the name of the security group
 - b) Add rule for RDP connection from bastion host to license manager

Type: RDP This is in the process of being updated. Until then, for the latest technical content, contact your AWS Account Manager

Source: Paste the name of the security group that you copied in step 3b

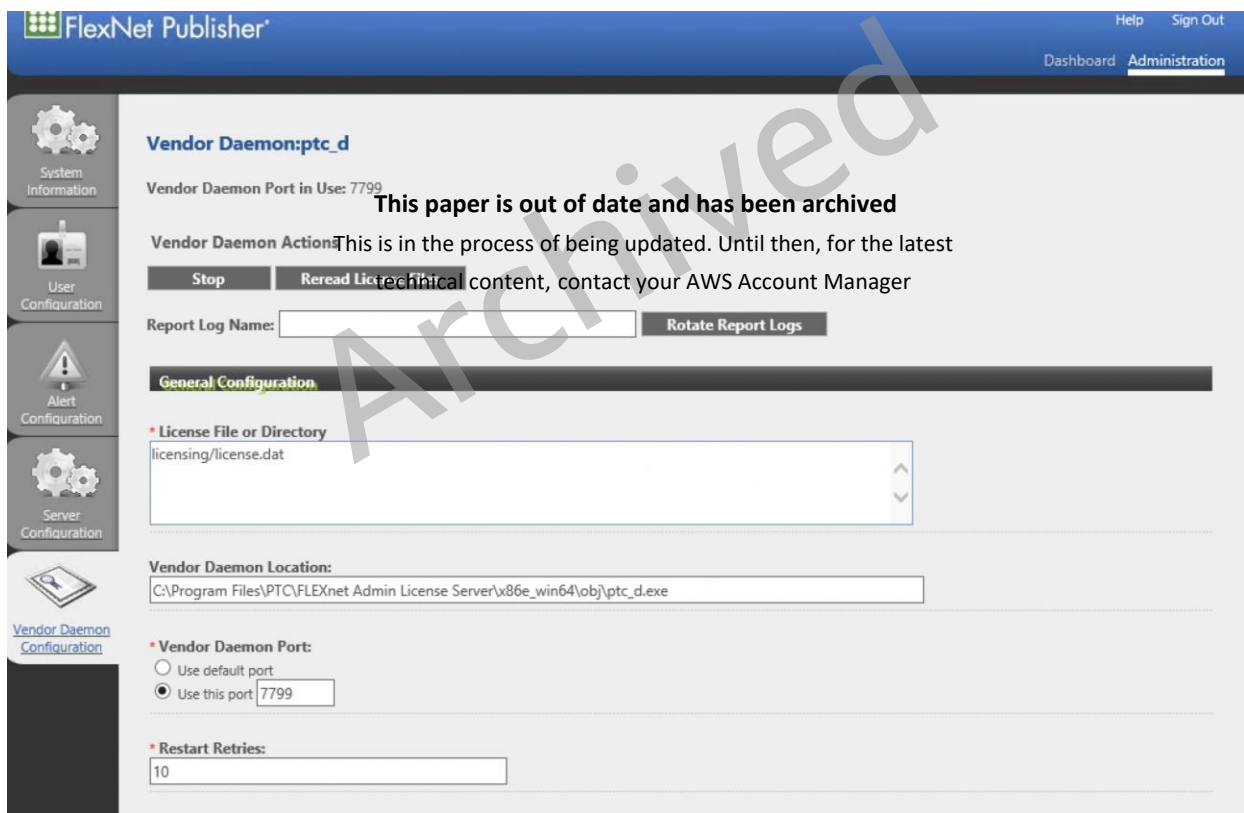
Step 3: Install and configure Creo License Manager

1. Connect to the Bastion host using Remote Desktop
2. Remote into the license server from the bastion host
3. After you connect to the designated license server, download the Creo install kit and license file
4. Install the license manager on this server.
5. You can then check the states of the license server by invoking the batch script from `..\FlexnetAdmin License Server\bin\ptcstatus.bat`
6. Then go to Start -> PTC -> FLEXnet Admin Web Interface, enter 'admin' as the value for both username and password (you will be prompted to change the password)

- Select 'Vendor Daemon Configuration' -> select the entry (in the screenshot below, its ptc_d.). First stop it



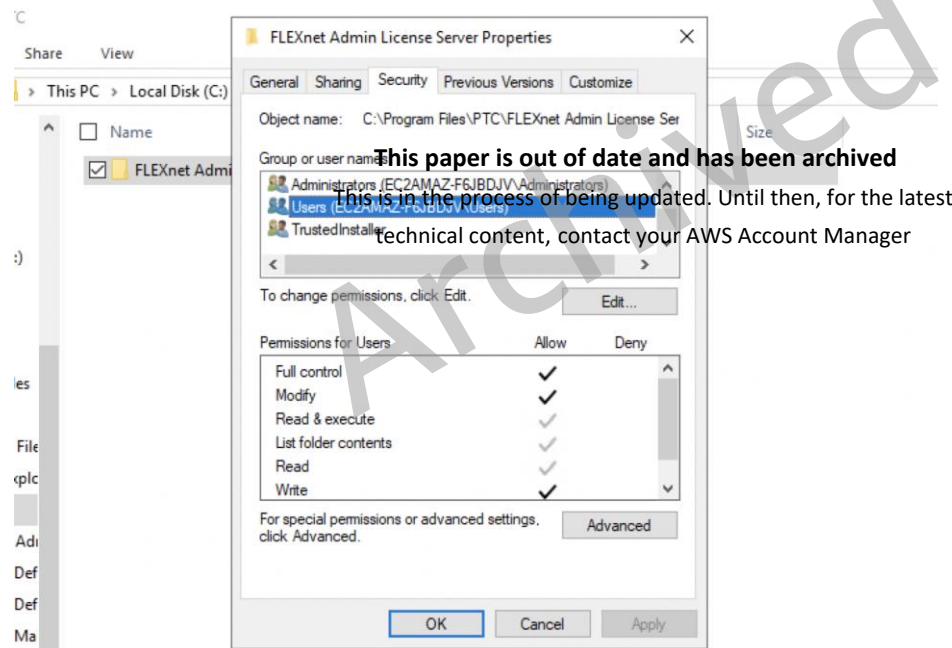
7. Now to enable proper communication between the license server and the AppStream instances on 7799 by configuring Vendor Daemon Port.



8. Then go to <Drive>:\Program Files\PTC\FLEXnet Admin License Server\logs and delete ptc_d file.
9. Enter the value for the Vendor Daemon log location to the above path, namely, <Drive>:\Program Files\PTC\FLEXnet Admin License Server\logs\ptc_d.log.



10. Using a Windows Explorer, navigate to <Drive>:\Program Files\PTC. Right-click FLEXnet Admin License Server folder -> Properties -> Security -> Edit. Give Full control for the 'Users' group



11. Then start the vendor daemon:ptc_d from step 9.

12. Open the ptc_d.log file and confirm if the its listening on port 7799

```
21:30:10 (ptc_d) (@ptc_d-SLOG@) === Network Info ===
21:30:10 (ptc_d) (@ptc_d-SLOG@) Listening port: 7799
21:30:10 (ptc_d) (@ptc_d-SLOG@) Daemon select timeout (in seconds): 1
21:30:10 (ptc_d) (@ptc_d-SLOG@)
```

13. Create an inbound and outbound firewall rule on the license server to allow communications on ports 7788 and 7799
14. Stop the vendor daemon:ptc_d process that was started in step 11. Open the license.dat file. Change the machine name identified in the 1st row to the IPv4 address of the license server. Then start back the vendor daemon:ptc_d process.

```

license - Notepad
File Edit Format View Help
SERVER 10.0.1.103| PTC_HOSTID=0A-A1-F1-FA-E6-2F 7788
DAEMON ptc_d "C:\Program Files\PTC\FLEXnet Admin License
INCREMENT PREPPMA_CUSTOMER ptc_d 38.0 13-jul-2020 10 \

```

15. Rerun the status check as in step 5.

```

C:\Users\Administrator>"C:\Program Files\PTC\FLEXnet Admin License Server\bin\ptcstatus.bat" -f
Displaying status for license file: C:\Program Files\PTC\FLEXnet Admin License Server\bin\..\licensing\license.dat
License Server is UP: C:\Program Files\PTC\FLEXnet Admin License Server\bin\..\licensing\license.dat
ptc_d FLEXnet Publisher version: 11.16
      License      In Use      Free
-----
PREPPMA_CUSTOMER      0         10
  Mecui_Advanced      0         10
  Meceng_Advanced      0         10
MECBASICENG_License    0         10
MECBASICUI_License     0         10
  MECEQRUN_EQ         0         10

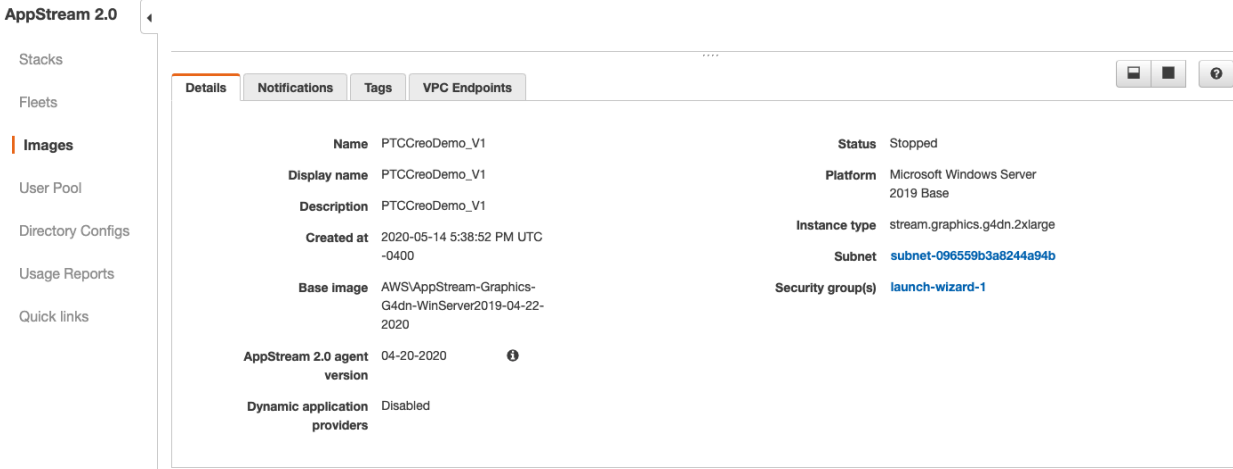
```

This paper is out of date and has been archived

This is in the process of being updated. Until then, for the latest

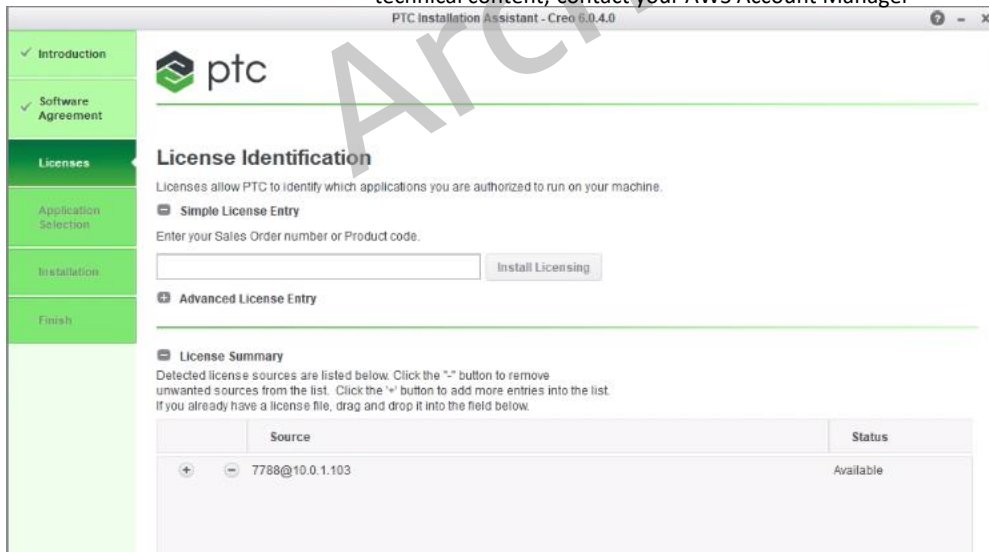
Step 4: Create an Amazon AppStream 2.0 image with Creo Installed

1. Open the Amazon AppStream 2.0 management console on AWS console and choose Images -> Image Builder, Launch Image Builder
2. Choose 'AppStream-Graphics-G4dn-Winserver2019<latest date> image
3. Deploy it in the VPC that you had created earlier and host it in the private subnet
 1. Select the security group attached to the license server EC2 instance (Step 2a-8b)



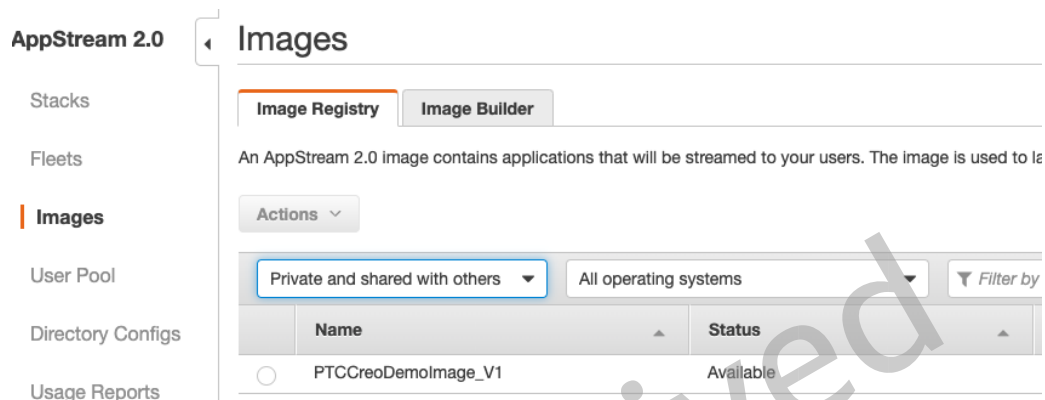
4. Once the image is launched, click 'Connect' once the status changes from pending to running. This will open a new browser tab and display the desktop of the image
5. Navigate to Control Panel, Systems, and Security and Windows Firewall and then create an inbound and outbound firewall to allow communications on ports 7788 and 7799
6. Now install Creo. ~~Not that during install enter the source as 7788@<IPv4 address of the License Server>, as shown below~~

This is in the process of being updated. Until then, for the latest technical content, contact your AWS Account Manager



7. Open the Creo Parametrics to make sure it's working properly
8. Click Image Assistant from the desktop. Set the Launch Path to the path to the parametrics.exe

9. Then click on the Test tab -> Switch users > Test user. This process will simulate a test user and you should be able to click on Image Assistant from the desktop and click on Parametric. Once you verify that application starts, you can close the application
10. Click Switch user -> Administrator.
11. Follow the wizard to the next step to optimize and finally create the image
12. You will see that the image will be created and the status marked as available after some time.



This paper is out of date and has been archived

This is in the process of being updated. Until then, for the latest

technical content, contact your AWS Account Manager

Step 5: Create an Amazon AppStream 2.0 Fleet and Stack

1. In the Amazon AppStream 2.0 management console, choose Fleets and create Fleet
2. Provide a name to the fleet -> select the image that was created in Step 5(13), deploy in the 2 private subnets and in the VPC that was created for AppStream. Note that you can select other criteria, such as, whether it should be on-demand or Always-on fleet type and the minimum number and maximum number of instances, disconnect timeout variables etc

| | | | |
|-------------------------|--|---------------------------------|--|
| Name | PTCCreoDemoFleet_V1 | Maximum session duration | 960 minutes |
| Display Name | PTCCreoDemoFleet_V1 | Disconnect timeout | 15 minutes |
| Status | Stopped | Idle disconnect timeout | 15 minutes |
| Image | Me (420581810454)\PTCCreoDemoImage_V1 | VPC | vpc-0cfd34b9e3d0ed0aa (AppStream2 VPC) |
| Stack | PTCCreoDemoStack_V1 | Subnets | subnet-037eabe9cc0ef4315 (10.0.2.0/24) subnet-096559b3a8244a94b (10.0.1.0/24) |
| Desired Capacity | 1 | Security group(s) | launch-wizard-1 |
| Instance Type | stream.graphics.g4dn.2xlarge | Created At | 2020-05-14 7:03:35 PM UTC -0400 |
| Fleet Type | On-Demand | Default Internet Access | Disabled |
| IAM role | None | | |

3. Then choose Stacks -> Create Stack. Enter a name for the stack
4. Once the stack is created, select it and choose Actions -> Associate Fleet. Associate the stack with the fleet created in step 5(2). Make sure you have enabled access to S3 bucket
5. Finally, confirm that Nvidia drive is properly installed on this instance. You can check the Quadro licensing is in place by running the following powershell command after installing the drivers

& "C:\Program Files\NVIDIA Corporation\NVSMI\nvidia-smi" -q

This paper is out of date and has been archived
This is in the process of being updated. Until then, for the latest technical content, contact your AWS Account Manager

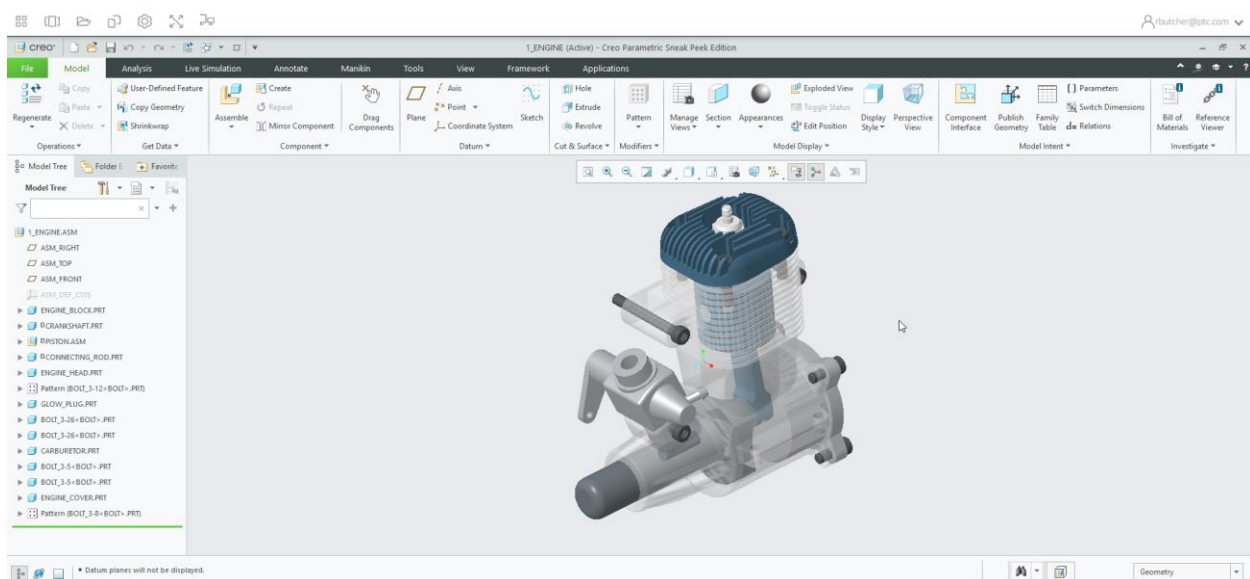
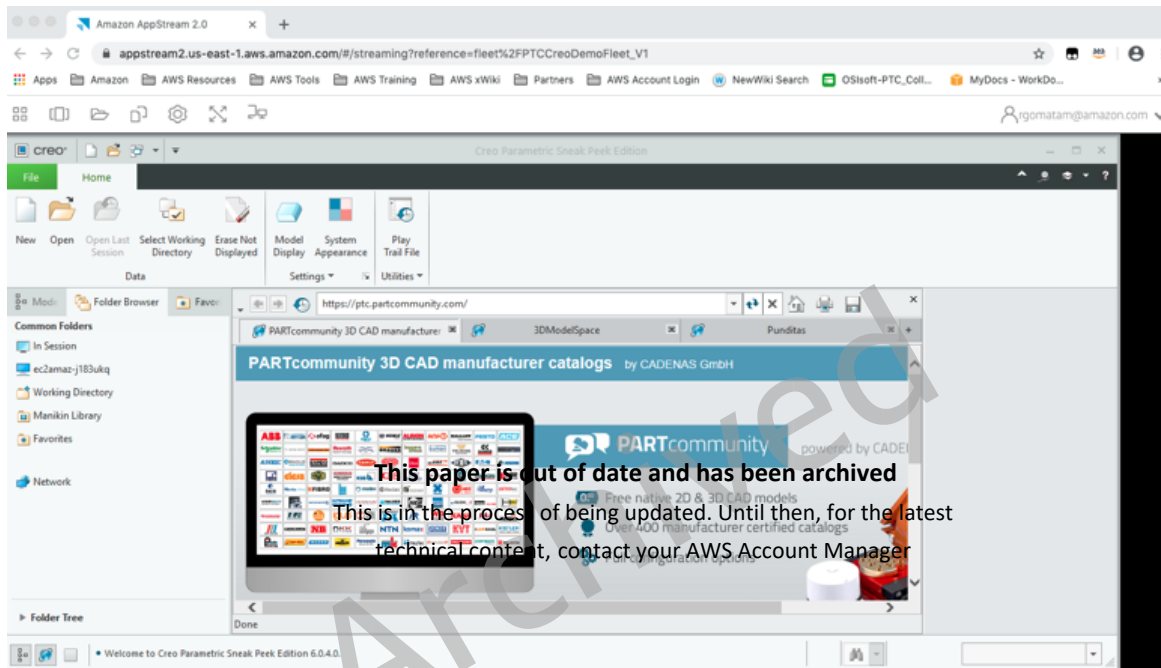
Confirm you see the following in the results. Look for the GRID Licensed Product to show status as "Licensed" and "Quadro Virtual Datacenter Workstation" as they License Type.

```
Host VGPU Mode      : N/A
GRID Licensed Product
Product Name       : Quadro Virtual Data Center Workstation
License Status     : Licensed
IBMNPU
```

Step 6: Invite users to access Creo through a web browser

1. Choose User Pools, Create User to create users
2. Enter the name and email address of each user
3. Select the user just created, and choose Actions, Assign Stack to provide access to the stack created in step 6(2). Your user will receive an email invitation to set up an account and use the web portal to access Creo that you have included in your stack.

4. Open the link for the Amazon AppStream 2.0 web portal shared through the email invitation. At this point you would be prompted to sign in with the temporary password and set a new password.
5. Launch Creo Parametric and you can create the CAD design and save it in your home folder, which is your own private folder in AWS S3 by navigating to D:\PhotonUser\MyFiles\Home Folder\



Cleaning up

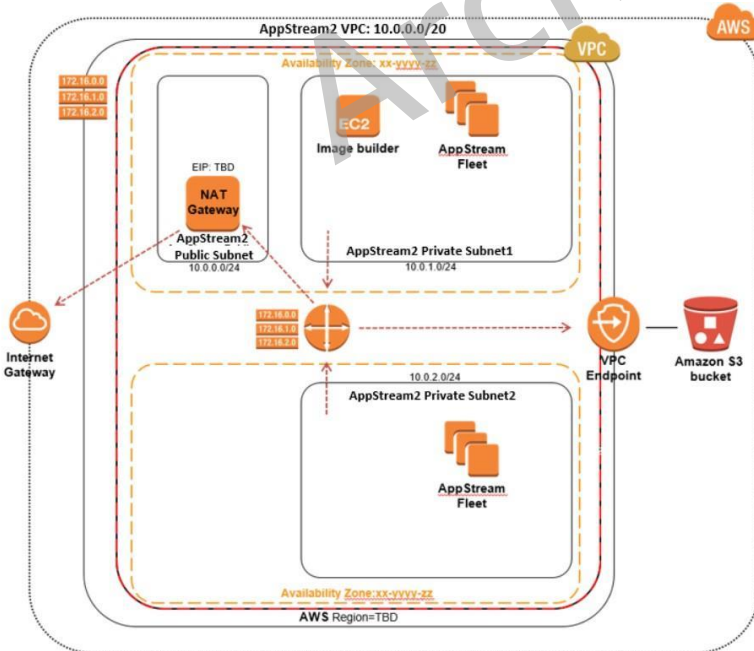
To avoid incurring future charges, delete the resources. Please delete the resources in the following order (Appendix B).

1. Delete the AppStream stack
2. Stop and then delete the fleet
3. Delete the EC2 instances for Creo License Manager server and Bastion host
4. Delete the cloud formation template
5. Finally, delete the AppStream 2.0 image

Appendix A. Manually create and configure network resources

Step 2 of this guide described how to use a CloudFormation template to automatically create and configure the necessary network resources for your AppStream 2.0 environment. To manually create and configure network resources, follow the steps in this appendix. At the end of this appendix, the topology of your VPC should look similar to the following diagram:

This paper is out of date and has been archived
This is in the process of being updated. Until then, for the latest technical content, contact your AWS Account Manager



Note: The CIDR block assignments for the private subnets might be reversed depending on the availability zones used by the VPC wizard.

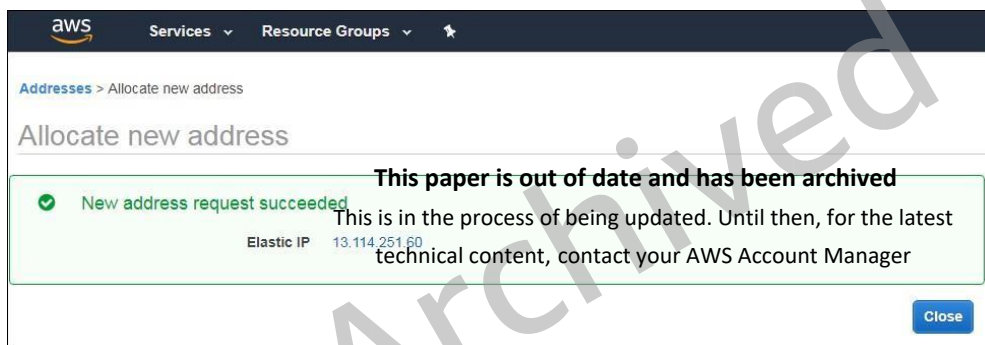
AppStream VPC requirements

At a minimum, AppStream 2.0 requires a VPC that includes one public subnet and two private subnets. A public subnet has direct access to the internet through an internet gateway. A private subnet requires a Network Address Translation (NAT) gateway or NAT instance to access the internet.

Allocate an Elastic IP address

Before you create your VPC, you must allocate an Elastic IP address in your AppStream 2.0 region. An Elastic IP address enables your streaming instances to be accessible through an internet gateway.

1. Open the Amazon EC2 console at <https://console.aws.amazon.com/ec2>.
2. In the navigation pane, under **Network & Security**, choose **Elastic IPs**.
3. Choose **Allocate New Address** and then choose **Allocate**.
4. Note the Elastic IP address and then choose **Close**.



Create a VPC by using the VPC Wizard

The easiest way to start building your VPC environment is to use the VPC Wizard. The wizard guides you through the process of creating a public subnet, private subnet, NAT gateway, and internet gateway, with the correct route table configurations.

1. Open the Amazon VPC console at <https://console.aws.amazon.com/vpc/>.
2. In the navigation pane, choose **VPC Dashboard**, **Start VPC Wizard**.
3. In **Step 1: Select a VPC Configuration**, choose **VPC with Public and Private Subnets**, **Select**.

Services ▾ Resource Groups ▾ ☆

Step 1: Select a VPC Configuration

VPC with a Single Public Subnet

VPC with Public and Private Subnets

VPC with Public and Private Subnets and Hardware VPN Access

VPC with a Private Subnet Only and Hardware VPN Access

In addition to containing a public subnet, this configuration adds a private subnet whose instances are not addressable from the Internet. Instances in the private subnet can establish outbound connections to the Internet via the public subnet using Network Address Translation (NAT).

Creates:

A /16 network with two /24 subnets. Public subnet instances use Elastic IPs to access the Internet. Private subnet instances access the Internet via Network Address Translation (NAT). (Hourly charges for NAT devices apply.)

[Select](#)

Internet, S3, DynamoDB, SNS, SQS, etc.

Amazon Virtual Private Cloud

Public Subnet NAT Private Subnet

- In **Step 2: VPC with Public and Private Subnets**, type the following information and then choose **Create VPC**.

| Option | Value |
|----------------------------|--|
| IPv4 CIDR block | 10.0.0.0/20 |
| IPv6 CIDR block | Accept the default value: <i>No IPv6 CIDR Block</i> |
| VPC Name | AppStream2 VPC |
| Public subnet's IPv4 CIDR | Accept the default value: 10.0.0.0/24 |
| Availability Zone | Accept the default value: <i>No Preference</i> |
| Public subnet name | AppStream2 Public Subnet |
| Private subnet's IPv4 CIDR | Accept the default value: 10.0.1.0/24 |
| Availability Zone | Accept the default value: <i>No Preference</i> |
| Private subnet name | AppStream2 Private Subnet1 |
| Elastic IP Allocation ID | Click the text box and select the value that corresponds to the Elastic IP address you created. This address is assigned to the NAT Gateway |
| Service endpoints | Choose Add Endpoint . |
| Service | Select the entry in the list that ends with "S3" (the com.amazonaws.ss-rrrr-x.s3 service that corresponds to the region in which the VPC is being created). Note: This is not the default value |
| Subnet | Select Private subnet |
| Policy | Accept the default value: Full Access |
| Enable DNS hostnames | Accept the default value: Yes |
| Hardware tenancy | Accept the default value: Default |

- After a few minutes, when a message in the VPC dashboard notifies you that the VPC is created, choose **OK**.

Add a second private subnet

1. In the navigation pane, choose **Subnets**.
2. Select the subnet with the name **AppStream2 Private Subnet1**. On the **Summary** tab, below the list of subnets, make a note of the Availability Zone for this subnet.

The screenshot shows the AWS console interface for subnets. At the top, there is a search bar and a list of subnets. The selected subnet, 'AppStream2 Private Subnet1', is highlighted. Below the list, the 'Summary' tab is active, displaying the following details:

- Subnet ID:** subnet-c175b38b | AppStream2 Private Subnet1
- Availability Zone:** us-east-1d
- IPv4 CIDR:** 10.0.1.0/24
- Route table:** rtb-831e9aff | AppStream2 Private Route Table
- IPv6 CIDR:** (empty)
- Network ACL:** acl-c0a140ba
- State:** available
- Default subnet:** no
- VPC:** vpc-d6ff74ad | AppStream2 VPC
- Auto-assign Public IP:** no
- Available IPs:** 250

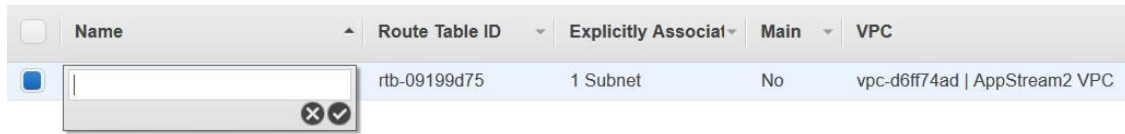
3. At the top of the same page, choose **Create Subnet**. Enter the following information in the **Create Subnet** dialog box and then choose **Yes, Create**.

| Option | Value |
|--------------------------|--|
| Name tag | <i>AppStream2 Private Subnet2</i> |
| VPC | <i>Select the VPC with the name AppStream2 VPC.</i> |
| Availability Zone | <i>Select an Availability Zone other than the one you are using for AppStream2 Private Subnet1. Selecting a different Availability Zone increases fault tolerance.</i> |
| IPv4 CIDR block | <i>10.0.2.0/24 (This is a subset of the CIDR block for your VPC.)</i> |

Modify the subnet route tables

1. In the navigation pane, choose **Subnets**, and then select the subnet with the name **AppStream2 Public Subnet**.
2. On the **Route Table** tab, note the ID of the route table (similar to *rtb-XXXXXXXX*).

- In the navigation pane, choose **Route Tables** and select the route table with the ID that you noted in the previous step.
- For **Name**, open the empty field, type *AppStream2 Public Route Table*, and then select the check mark to save your changes.



- Make sure that **AppStream2 Public Route Table** is still selected. On the **Routes** tab, verify that the route table includes the following two routes:

| Destination | Target |
|-------------|------------------------|
| 10.0.0.0/20 | Local |
| 0.0.0.0/0 | <i>igw - XXXXXXXXX</i> |

These two routes function as follows for all resources within a subnet that is associated with the route table:

- Local: All traffic from the resources destined for IPv4 addresses within the 10.0.0.0/20 CIDR block is routed locally within the VPC.
- Outbound: Traffic destined for all other IPv4 addresses is routed to the internet gateway (identified by *igw-XXXXXXX*) that was created by the VPC Wizard.

To modify the route table, choose **Edit** and make the needed changes.

- In the navigation pane, choose **Subnets** and select the subnet named **AppStream2 Private Subnet1**.
- On the **Route Table** tab, note the ID of the route table (similar to *rtb-XXXXXXX*).
- In the navigation pane, choose **Route Tables** and select the route table with the ID you noted in the previous step.
- For **Name**, open the empty field, type *AppStream2 Private Route Table*, and then select the check mark to save your changes.

| | | | | | |
|-------------------------------------|--------------------------------|--------------|-----------|-----|-------------------------------|
| <input type="checkbox"/> | AppStream2 Public Route Table | rtb-09199d75 | 1 Subnet | No | vpc-d6ff74ad AppStream2 VPC |
| <input checked="" type="checkbox"/> | AppStream2 Private Route Table | rtb-831e9aff | 0 Subnets | Yes | vpc-d6ff74ad AppStream2 VPC |
| <input type="checkbox"/> | | rtb-c7d654ba | 0 Subnets | Yes | vpc-c98ab7b1 |

10. Make sure that **AppStream2 Private Route Table** is still selected, and on the **Routes** tab, verify that the route table includes the following routes:

| Destination | Target |
|---|--------------------------|
| 10.0.0.0/20 | local |
| 0.0.0.0/0 | nat-XXXXXXXXXXXXXXXXXXXX |
| pl-YYYYYYYY (com.amazonaws.<region>-<#>.s3) | vpce-ZZZZZZ |

These three routes function as follows for all resources within a subnet that is associated with the route table:

- Local: All traffic from the resources destined for IPv4 addresses within the 10.0.0.0/20 CIDR block is routed locally within the VPC.
 - Storage: Traffic destined for S3 buckets is routed to the S3 endpoint (identified by *vpce-ZZZZZZ*).
 - Outbound: Traffic destined for all other IPv4 addresses is routed to the NAT gateway (identified by *nat-XXXXXXXX*).

To modify the route table, choose **Edit** and make the needed changes.

11. In the navigation pane, choose **Subnets** and select the subnet with the name **AppStream2 Private Subnet2**.
12. On the **Routes** tab, verify that the route table is the one named AppStream2 Private Route Table. If the route table is different, choose **Edit** and select this route table.

Appendix C. Clean up your AppStream 2.0 resources

Although you can continue to use this AppStream 2.0 environment, keep in mind that you pay for your running resources. For more information, see [Amazon AppStream 2.0 Pricing](#).

Cleaning up the resources that you created frees up resources and helps you avoid unintended charges to your account.

Stop and delete your image builder

1. Open the Amazon AppStream 2.0 console at <https://console.aws.amazon.com/appstream2> .
2. In the navigation pane, choose **Images, Image Builder**.
3. Confirm whether the image builder that you created in Step 3 in this guide is in a stopped state. If not, select the image builder and choose **Actions, Stop**. If you created multiple image builders, repeat this step for each image builder that you created.
4. After the image builder has stopped, choose **Actions, Delete**. Repeat this step for each image builder that you created.

This paper is out of date and has been archived

This is in the process of being updated. Until then, for the latest

technical content, contact your AWS Account Manager

Revoke stack permissions

1. In the navigation pane, choose **User Pool**.
2. Select the user you created in Step 9 in this guide and choose **Actions, Unassign stack**. This action revokes the stack permissions for the user.

Disassociate your fleets from your stack and delete your stack

1. In the navigation pane, choose **Stacks**.
2. Select the stack you created and choose **Actions, Dissociate Fleet**. This action dissociates the fleet from the stack.
3. To delete the stack, choose **Actions, Delete**.

Stop and delete your fleet

1. In the navigation pane, choose **Fleets**.
2. Confirm whether the fleet that you created in Step 6 in this guide is in a stopped state. If not, select the fleet and choose **Actions, Stop**.

3. After the fleet has stopped, choose **Actions, Delete**.

This paper is out of date and has been archived

This is in the process of being updated. Until then, for the latest technical content, contact your AWS Account Manager

Archived