Contents

Chapter 1

Recommended Deployment Configurations: Client, Server Software, Server Hardware

The recommended server configuration is AWS Marketplace and Azure Marketplace with instance types noted on those screens. The one-click launcher deploys a fully preconfigured instance.

Graphistry Enterprise also ships as a Docker container that runs in Linux Nvidia GPU environments that are compatible with NVIDIA RAPIDS and the Nvidia Docker runtime. It is often run in the cloud as well.

Contents

- Overview
- Client
- Server Software: Cloud, OS, Docker, Avoiding Root Users
- Server Hardware: Capacity Planning

Overview

- **Client**: Chrome/Firefox from the last 3 years, WebGL enabled (1.0+), and 100KB/s download ability

- **Server**:
- Minimal: x86 Linux server with 4+ CPU cores, 16+ GB CPU RAM (3GB per concurrent user), 150GB+ disk, and 1+ Nvidia GPUs (Pascal onwards for NVIDIA RAPIDS) with 4+ GB RAM each (1+ GB per concurrent user)
- Recommended: Ubuntu 18.04 LTS, 4+ CPU cores, 64GB+ CPU RAM, 150GB+ disk, Nvidia Pascal or later (Volta, RTX, . . .) with 12+GB GPU RAM
- CUDA driver rated for NVIDIA RAPIDS
- Nvidia Docker runtime set as default runtime for docker-compose 1.20.0+ (yml file format 3.4+)

Client

A user's environment should support Graphistry if it supports Youtube, and even better, Netflix.

The Graphistry client runs in standard browser configurations:

- **Browser**: Chrome and Firefox from the last 3 years, and users regularly report success with other browsers

- **WebGL**: WebGL 1.0 is required. It is 7+ years old, so most client devices, including phones and tablets, support it: check browser settings for enabling. Graphistry runs fine on both integrated and discrete graphic cards, with especially large graphs working better on better GPUs.

- **Network**: 100KB+/s download speeds, and we recommend 1MB+/s if often using graphs with 100K+ nodes and edges.

- **Operating System**: All

Recommended: Chrome from last 2 years on a device from the last 4 years and a 1MB+/s network connection

Server Software: Cloud, OS, Docker, Avoiding Root Users

Graphistry can run both on-premises and in the cloud on Amazon EC2, Google GCP, and Microsoft Azure.

Cloud

Tested AWS Instances:

- P3.2 ***Recommended for testing and initial workloads***

Tested Azure Instances:

- NC6s_v2 ***Recommended for testing and initial workloads***
- NC6s_v3

See the hardware provisioning section to pick the right configuration for you.

Please contact for discussion of multi-GPU scenarios.

OS & Docker

Graphistry runs preconfigured with a point-and-click launch on Amazon Marketplace.

Graphistry regularly runs on:

- Ubuntu Xenial 16.04+ LTS ***Recommended***
- RedHat RHEL 7.4+

Both support Nvidia / Docker:

- CUDA driver rated for NVIDIA RAPIDS
- Nvidia Docker *native* runtime (for after Docker 19.03)
- docker-compose 1.20.0+ (yml file format 3.4+) with default runtime set as `nvidia` at time of launch

See Ubuntu 18.04 LTS manual configuration and RHEL 7.6 manual configuration for an example of setting up Nvidia for containerized use on Linux. Marketplace deployments come preconfigured with the latest working drivers and security patches.

User: Root vs. Not, Permissions

Installing Docker and Nvidia dependencies currently require root user permissions.

Graphistry can be installed and run as an unprivileged user as long as it has access to Docker and the Nvidia runtime.

Storage

We recommend using backed-up network attached storage for persisting visualizations and investigations. Data volumes are negligible in practice, e.g., < $10/mo on AWS S3.

Server Hardware: Capacity Planning

Graphistry utilization increases with the number of concurrent visualizations and the sizes of their datasets. Most teams will only have a few concurrent users and a few concurrent sessions per user. So, one primary server, and one spillover or dev server, gets a team far.

For teams doing single-purpose multi-year purchases, we generally recommend more GPUs and more memory: As Graphistry adds further scaling features, users will be able to upload more data and burst to more devices.

Network

A Graphistry server must support 1MB+/s per expected concurrent user. A moderately used team server may stream a few hundred GB / month.

- The server should allow http/https access by users and ssh by the administrator.
- TLS certificates can be installed (nginx)
- The Graphistry server may invoke various database connectors: Those systems should enable firewall and credential access that authorizes authenticated remote invocations from the Graphistry server.

GPUs & GPU RAM

Graphistry requires NVIDIA RAPIDS-compatible GPUs. The following GPUs, Pascal and later, are known to work with Graphistry:

- P100, V100, RTX
- ... Found both in DGX and DGX2

The GPU should provide 1+ GB of memory per concurrent user. At 4GB of GPU RAM is required, and 12GB+ is recommended.

CPU Cores & CPU RAM

CPU cores & CPU RAM should be provisioned in proportion to the number of GPUs and users:

- CPU Cores: We recommend 4-6 x86 CPU cores per GPU
- CPU RAM: We recommend 6 GB base memory and at least 16 GB total memory for a single GPU system. For balanced scaling, 3 GB per concurrent user or 3X the GPU RAM.

Multi-GPU, Multi-Node, and Multi-Tenancy

Graphistry virtualizes a single GPU for shared use by multiple users and can vertically scale to multiple CPUs+GPUs on the same node for additional users.

- Multitenancy via multiple GPUs: You can use more GPUs to handle more users and give more performance isolation between users. We recommend separating a few heavy users from many light users, and developers from non-developers.

- Acceleration via multiple GPUs: Graphistry is investigating how to achieve higher speeds via multi-GPU acceleration, but the current benefits are only for multitenancy.

HA

Graphistry resiliency typically comes in multiple forms:

- User separation: For larger deployments, we recommend separating developers (unpredictable), power users (many large graphs), and regular users (many small sessions).

- Physical resource isolation: Graphistry can run on the same device as other software as long as they respect Graphistry's CPU, GPU, and network resources. You can use Docker to limit Graphistry execution to specific CPU, GPU, and network resources. Check your other software to ensure that it can likewise be configured to not interfere with sibling workloads.

- Process isolation: You may run multiple Graphistry instances on the same node to increase resiliency between groups of users. This can be combined with same-node physical resource isolation. This increases resiliency up to hardware and driver failure. However, note that each Graphistry instance will consume 1.5GB+ of GPU RAM.

- Logical separation & replication: You may want to further tune software replication factors. Graphistry runs as multiple containerized shared services with distinct internal replication modes and automatic restarts. Depending on the service, a software failure may impact live sessions of cotenant users or prevent service for 3s-1min. Within a node, you may choose to either tune internal service replication or run multiple Graphistry instances.

- Safe upgrades: Due to Graphistry's use of version-tagged Docker images and project-namespaceable docker-compose orchestrations, upgrades can be performed through:

- New instances (e.g., DNS switch): recommended, especially for cloud

- Installation of a concurrent version Contact support staff for migration information.