Deployment Plan for Tyche

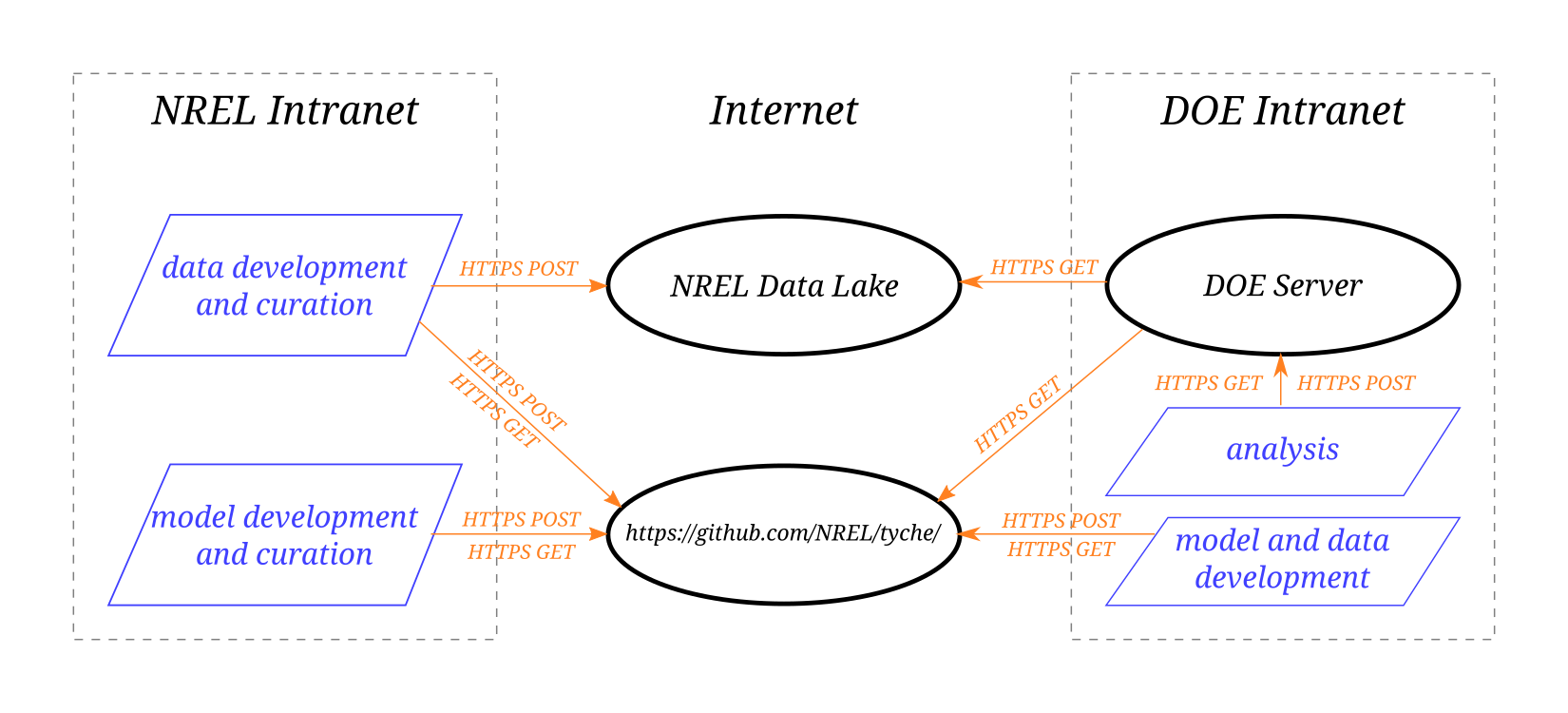
Version 0.1 Draft

20 November 2020

# Objectives

1. Securely house all potentially sensitive data within on DOE servers within the DOE intranet.
2. Minimize the deployment and maintenance burden at DOE.
3. Assure the quality of software and data updates.
4. Enable DOE personnel and contractors to contribute technology models and data.

# Components and Activities



Deployment of services and activities.

## Activities

Analysts at DOE will connect to Tyche server within the DOE intranet using their web browsers to run and analyze scenarios using Tyche. The server will have the capability to record scenarios for sharing within DOE, but that data will never leave the DOE intranet.

Analysts developing data and technology models at DOE, NREL, and elsewhere can post that data and software to a branch of the GitHub Software Repository. Those contributions will be reviewed, vetted, and tested before they are pushed to the NREL Data Lake (in the case of datasets) or to the production branch of the GitHub repository (in the case of technology models).

NREL will perform quality assurance and periodically update the production version of the data and software, both of which can be fetched by DOE on a regular basis.

## Components

### DOE Server

The DOE server for Tyche resides within the DOE intranet. It fetches software updates from the GitHub Software Repository and fetches data updates from the NREL Data Lake. (Because data volumes are small, it could perform these automatically on a daily or weekly basis during off hours.) It runs a [Quart HTTP server](https://pgjones.gitlab.io/quart/) within an [Anaconda](https://www.anaconda.com/) environment. Requirements for this server are as follows:

1. Linux (preferred) or Windows.
2. Four to 16 CPU cores and at least 32 GB of memory.
3. An up-to-date installation of the [Anaconda](https://www.anaconda.com/) software package manager.
4. Installation of the Tyche Conda environment within Anaconda. (This will install the correct version of Python and the other required software, so those need not be installed individually.) See the attachment <conda-environment.yml>.
5. Running a shell script for the Quart HTTP server.
6. Open outgoing HTTPS ports for GET requests to the NREL Data Lake and GitHub.com.
7. An open HTTP incoming port from client web browsers withing the DOE intranet.
8. A folder on disk that is regularly backed up.

### NREL Data Lake

The NREL Data Lake, which is housed on Amazon Web Services (AWS), contains all of the non-sensitive data, such as the parameters for technology models and the results of expert elicitations. NREL curates the data that is pushed to the data lake.

### GitHub Software Repository

The Tyche software resides on the NREL GitHub software repository <<https://github.com/NREL/tyche/>>. The production branch contains the latest deployable version of the software. Other branches contain work in progress, contributions from DOE and its subcontractors, and the development (pre-release) version of the software.

# Security Considerations

1. NREL has authority to operate (ATO) with non-sensitive software and data on its Data Lake and on GitHub.com.
2. Sensitive data (in the form of scenario definitions and results) may reside on the DOE server and on the laptops of DOE users.
3. The Tyche service only makes HTTPS GET requests outside of the DOE intranet, and these only consist of fetching non-sensitive datasets and technology models. Thus, the firewall for the Tyche server should be configured at follows:
   1. Block all incoming traffic from outside the DOE intranet.
   2. Allow incoming HTTP traffic from inside the DOE intranet.
   3. Allow outgoing HTTPS traffic to NREL Data Lake and GitHub.com.
   4. Block all other outgoing traffic.
4. Ideally, the Tyche software (and its library dependencies) and its updates should undergo a security audit.