ERDC/ITL TR-20-2

Information Technology Laboratory

US Army Corps of Engineers®
Engineer Research and Development Center

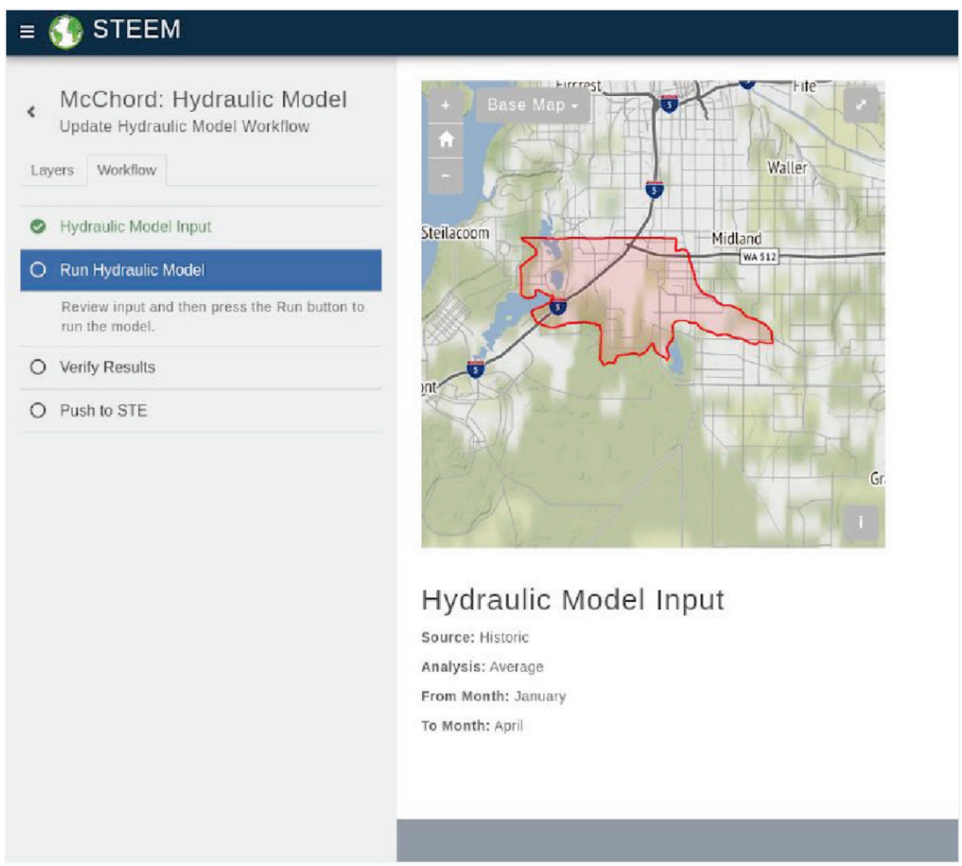
ERDC
ENGINEER RESEARCH & DEVELOPMENT CENTER

Engineered Resilient Systems RDA: Deployable Computational Capability for STE

STE Environmental Manager (STEEM) Demonstration Web Application

Jonathan Boone, Nathan Swain, Gage Larsen, and Michael Souffront

September 2020



STE Environmental Manager (STEEM) Demonstration Web Application

Jonathan L. Boone

Information Technology Laboratory
U.S. Army Engineer Research and Development Center
3909 Halls Ferry Road
Vicksburg, MS 39180-6199

Nathan Swain, Gage Larsen, and Michael Souffront

Aquaveo, LLC
3210 N. Canyon Road
Suite 300
Provo, Utah 84604

Final report

Approved for public release; distribution is unlimited.

Prepared for Information Technology Laboratory (ITL)
3909 Halls Ferry Road
Vicksburg, MS 39180-6199

Under Engineered Resilient System RDA, Deployable Computational Capability for STE,
FWIC:8f523J

Abstract

This report provides a summary of the development of the Synthetic Training Environment (STE) Environmental Manager (STEEM) demonstration web application. The purpose of this web application is twofold: (1) demonstrate a web application that enables non-technical users to prepare, run, and manage the physics-based models used by the STE to simulate realistic environmental conditions and (2) show how technologies developed by the Engineered Resilient Systems (ERS) Research and Development Area can be used to rapidly create applications to support U.S. Army Engineer Research and Development Center (ERDC) programs like the STE. A full build-out of STEEM would leverage the following ERS-developed technologies: data services, model development environment tools, model coupling/interface API, simulation workflow manager, and scenario generation tools.

Contents

Figures and Tables

Figures

Tables

Preface

This study was conducted for the Information Technology Laboratory (ITL) under the Engineered Resilient Systems (ERS) RDA FWIC 8f523J, “Deployable Computational Capability for STE.” The technical director for this work was Dr. Robert Wallace.

The work was performed by the Computer Aided Design/Building Information Modeling Technology Center of the Software Engineering and Informatics Division (SEID), U.S. Army Engineer Research and Development Center, Information Technology Laboratory (ERDC-ITL) and the Aquaveo, LLC team under contract by ERDC-ITL. At the time of publication, Mr. Edward L. Huell, Jr was the Chief of the CAD/BIM Technology Center, Mr. Quincy Alexander was the Acting Chief of the SEID, Ms. Patti S. Duett was the Deputy Director of ITL, and Dr. David A. Horner was the Director.

COL Teresa A. Schlosser was the Commander of ERDC, and Dr. David W. Pittman was the Director.

1 Introduction

1.1 Background

The Synthetic Training Environment (STE) is a system designed to provide a cognitive, collective, and virtual training capability. One of the main components of the STE is the ability to simulate the physical environment through terrain representation. The STE will also have the capability to use the results of physics-based environmental models to generate more realistic virtual training scenarios. For example, the STE may leverage a combination of weather, hydrologic, hydraulic, coastal, and mobility model output to more accurately simulate forecasted conditions for a particular mission or scenario.

The STE Environment Manager (STEEM) demonstration web application was developed to show how the physics-based models used by STE could be easily managed by users with limited to no experience with any of the models. It does this by distilling common modeling scenarios into simple, step-by-step workflows. For example, one workflow may involve selecting a forecasted storm event to use in the next STE session. STEEM performs all of the work needed to reparametrize and execute the underlying models and then prepares the output for consumption by the STE.

The development of the STEEM web application required the use of recent advancements in information technology (IT) and communication technologies, computing, and modeling to dynamically gather model inputs, run them, and manage them. Many of these advancements are the result of the research and development carried out by the Engineered Resilient Systems (ERS) Research and Development Area.

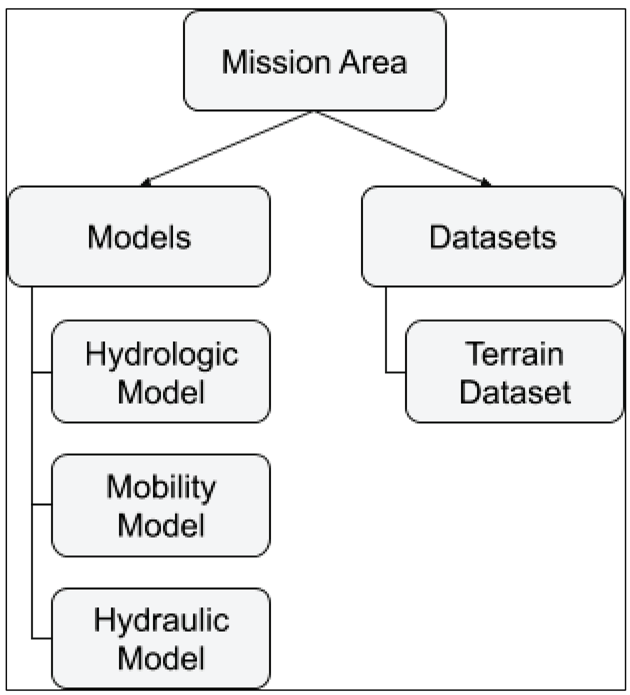
1.2 Objectives

The objective of the STEEM demonstration application is twofold: (1) demonstrate a web application that enables non-technical users to prepare, run, and manage the physics-based models used by the STE to simulate realistic environmental conditions and (2) show how technologies developed or improved by the Engineered Resilient Systems (ERS) Research and Development Area (Tethys Platform, Cesium 3D maps, Bokeh, and Panel) can be used to rapidly create applications to support U.S. Army Engineer Research and Development Center (ERDC) programs like the STE. This greatly simplified the design process, turning it into more of an integration exercise than a development activity. However, there were still parts of the implementation that were specific to STEEM that needed to be designed. The following sections describe some of the key components of STEEM and how they were designed.

1.3 Approach

The primary data that STEEM operates on are the models and datasets that are important to the STE to build the virtual training environment. Conceptually, the models and datasets are organized into Mission Areas (Figure 1).

Figure 1. STEEM modeling structure.

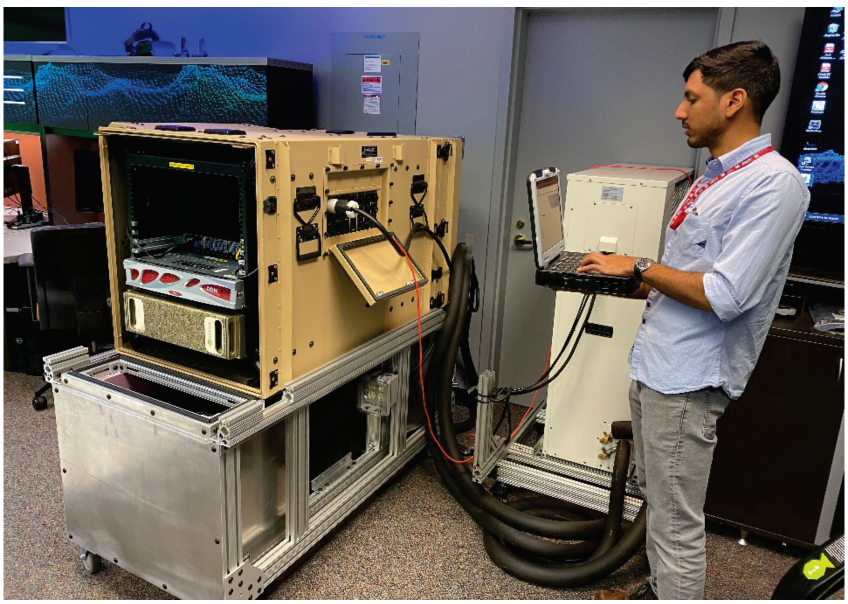


A Mission Area resource is a collection of other resources grouped by a geographic area. For example, one Mission Area could include coastal, hydrologic, mobility, meteorological, and terrain models. A Mission Area is represented visually in the application as a polygon that encompasses the combined extent of all models and datasets it contains.

For the demonstration version of the application, an existing resource data model was used to store metadata about each Mission Area in a Structured Query Language (SQL) database. This resource model can also store information about the model and dataset files associated with each Mission Area in the future. However, a full design for the file database that would store these files was not undertaken as part of the demonstration implementation of STEEM.

While the STEEM application could also be designed to work with data services, one application of STEEM is for it to be used offline on a portable supercomputer (Figure 2) in forward deployed locations. Any data that STEEM uses in this mode of operation would need to be available to it locally. Some refactoring of the framework used to develop STEEM would be required to support offline mode and was out of scope for the development of the demonstration version of the application.

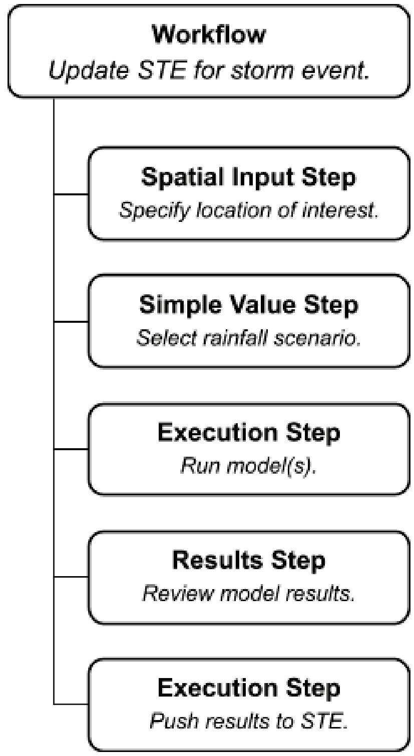
Figure 2. STEEM running on portable supercomputer.



1.4 Workflows

Workflows are another important component of the STEEM design. They are the primary means by which users operate on the models and datasets contained in the Mission Areas. Workflows divide common modeling and data manipulation tasks into simple steps (Figure 3). The input and output from each step in an instance of a workflow are saved for later review or reuse.

Figure 3. Illustration of how the different types of steps are used in a workflow.



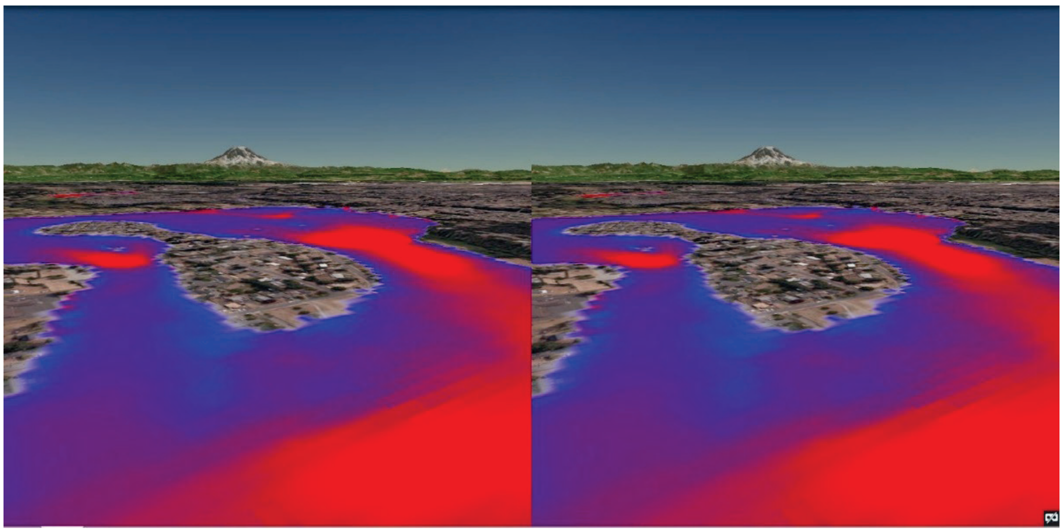
Workflow Steps are categorized into three types: Input Steps, Execution Steps, and Result Steps. The Input Steps include different types of steps for gathering input from users, including simple valued input (e.g. numbers, text, options), spatial input (e.g. points, lines, polygons), and tabular input (e.g. spreadsheet data). The Execution Steps are capable of executing models or scripts using local or remote computational resources. The Result Steps provide visualizations of different types of data including spatial data and tabular data.

A single workflow may operate on more than one model or dataset. For example, a workflow that generates the data needed by STE to simulate a rainfall event may need to run a hydrologic, hydraulic, coastal, and even a mobility model to properly characterize the environmental response. Instances of workflows are constrained to operate only on models and datasets of the Mission Area to which they belong.

1.5 Visualization

An important part of the design of STEEM is the visualization. The visualizations in the application need to be simple and intuitive to users who will have limited or no experience with the models and datasets. Since many of the models and datasets have a spatial component to them, various mapping tools were implemented. The Spatial Result Step allows data to be visualized either on a 2D OpenLayers-based map or on a 3D Cesium-based map. As the data would eventually be consumed by STE and used in a virtual environment, the built-in virtual reality mode was enabled on the 3D Cesium Result Steps, to allow users to preview the results data in virtual reality (VR) using Google Cardboard (Figure 4).

Figure 4. Screenshot of velocity data in VR mode on Cesium Globe Map.



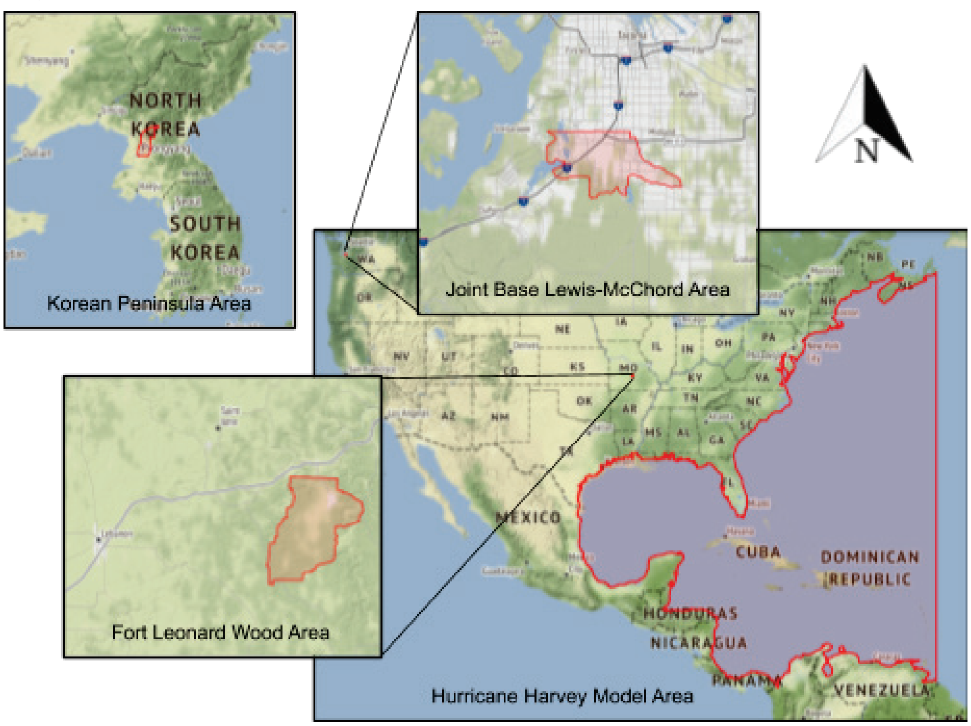
2 Demonstration Application

2.1 Background

A demonstration version of the STEEM application was developed. As a demonstration version, the implementation was minimal with much of the calculations pre-computed. The minimal implementation leverages a proprietary extension of Tethys Platform called ATCore so it could be developed as rapidly as possible. ATCore provides much of the workflow framework as discussed in the Design section.

The Adaptive Hydraulics (AdH) model was the model selected for the demonstration version of STEEM because there were already pre-existing models of several potential Mission Areas readily available. The Mission Areas included in the demonstration version of STEEM include Fort Leonard Wood, Joint Base Lewis-McChord, a region of the Korean Peninsula, and a simulation of the Atlantic coast of the United States during Hurricane Harvey (Figure 5).

Figure 5. The Mission Areas included in the demonstration version of STEEM.

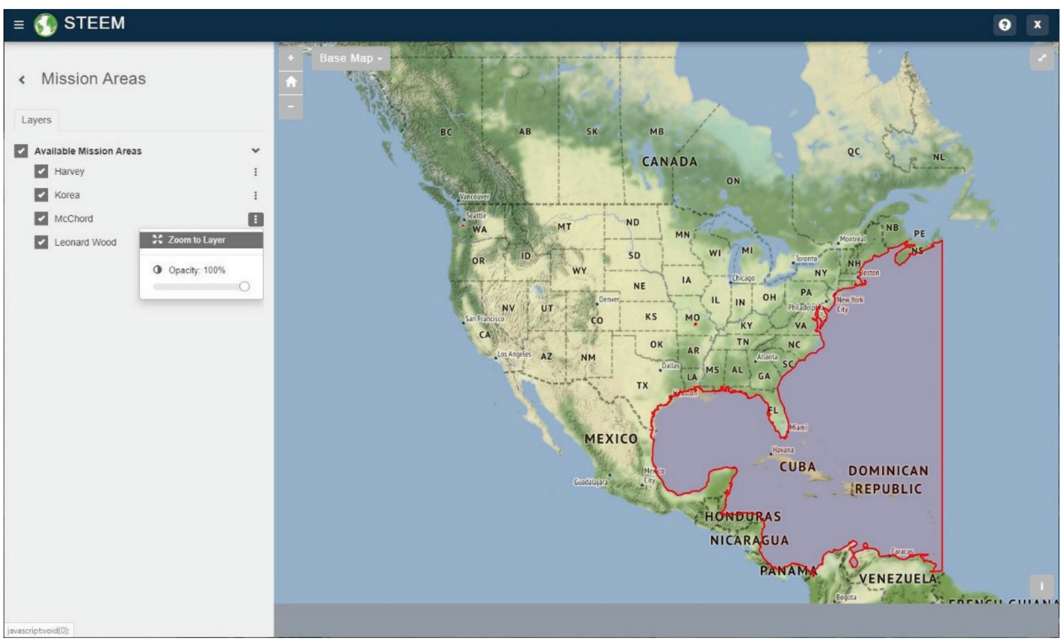


The demonstration version of the STEEM web application is composed of several views or web pages. These include the Select Mission Area view, the Mission Area Details view, the Workflow Views, and the Administration views. A brief description of each of these is provided in the following sections.

2.2 Select mission area

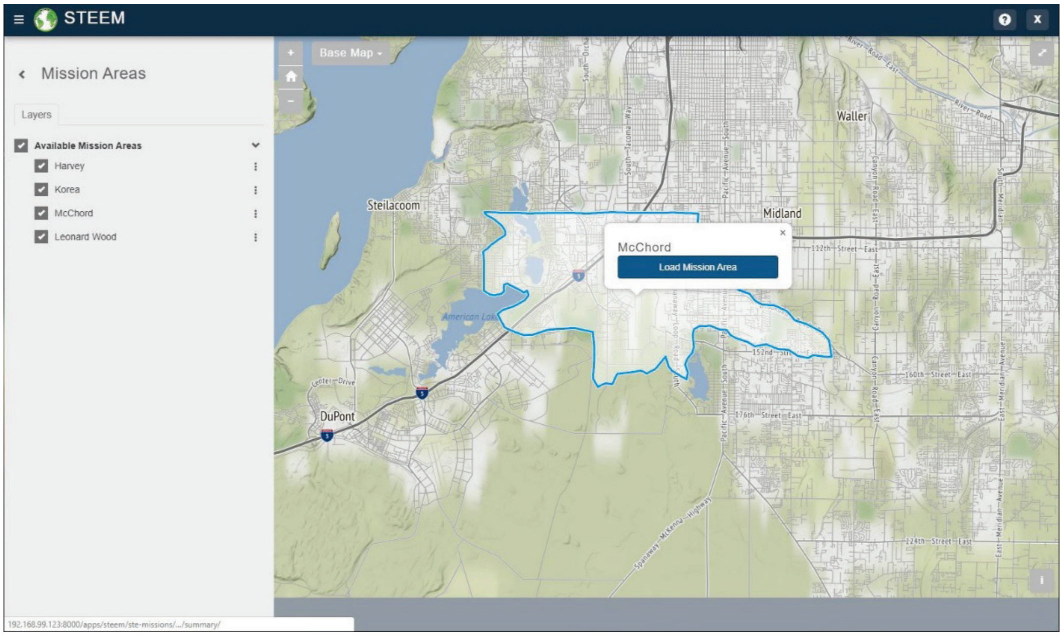
Upon launching the STEEM application, the user is presented with the Select Mission Area view. This view contains a map showing each of the available Mission Areas (Figure 6). The map is dynamic and interactive, which allows users to pan and zoom to the desired Mission Area. In addition, a list of the layers associated with each Mission Area is displayed in the left-hand menu under the tab labeled Layers. The checkbox on the left side of each layer list item controls the visibility of the Mission Area boundary layer on the map. The dot-menu on the right side of the layer list item contains various commands for manipulating the associated Mission Area layer.

Figure 6. Screenshot of the select mission area view of the STEEM web application.



To load a Mission Area, the user clicks anywhere within the Mission Area boundary to select it and presses the "Load Mission Area" button from the pop-up that appears (Figure 7). The "Zoom to Layer" option available in the dot menu for each Mission Area can be used to easily locate a particular Mission Area.

Figure 7. Mission area selection and loading.



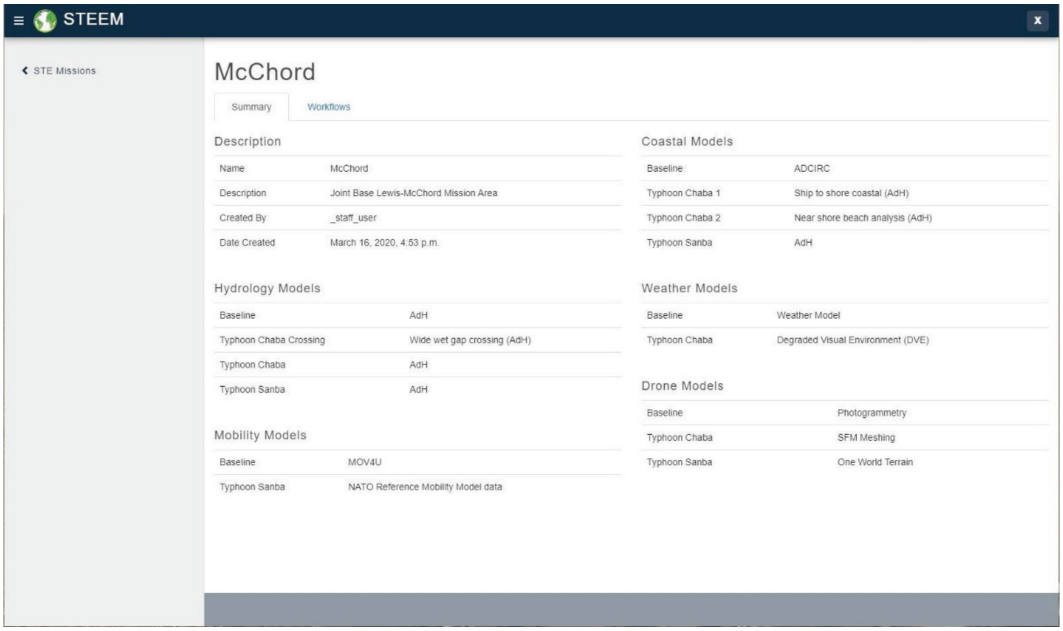
2.3 Mission area details

The Mission Area Details view contains details about a specific Mission Area. This information is grouped into two tabs: the Summary tab and Workflows tab. In the future, additional tabs could be added to provide more information about the Mission Area if desired. The Mission Area Details view is accessed by selecting a Mission Area on the Select Mission Area view and loading it. A brief description of each of the tabs follows.

2.3.1 Mission area summary

The Mission Area Summary tab contains summary information about the Mission Area. For the demonstration version of STEEM, the information in this tab shows metadata about the Mission Area and lists the models it contains, grouped by type (Figure 8). This list is hard-coded in the demonstration version of the application and will be the same for any Mission Area selected. In a full build of the application, this list would be dynamically generated from the models contained in the Mission Area database.

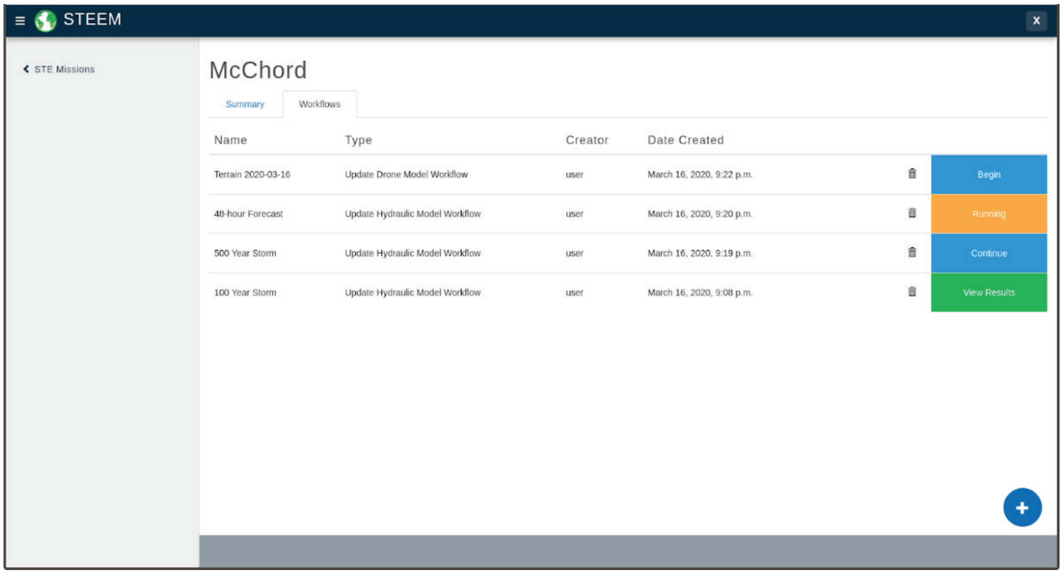
Figure 8. Screenshot of the mission area summary tab.



2.3.2 Mission area workflows

The Mission Area Workflows tab lists all workflows associated with that Mission Area (Figure 9). From this tab, users can view the status of workflows, create new workflows, continue existing workflows that are not yet complete, and review the results of past workflows. All data entered by users while completing workflows and the results of model runs are saved automatically, which allows users to complete workflows in multiple sessions and compare results of past workflows.

Figure 9. Screenshot of the mission area workflows tab.

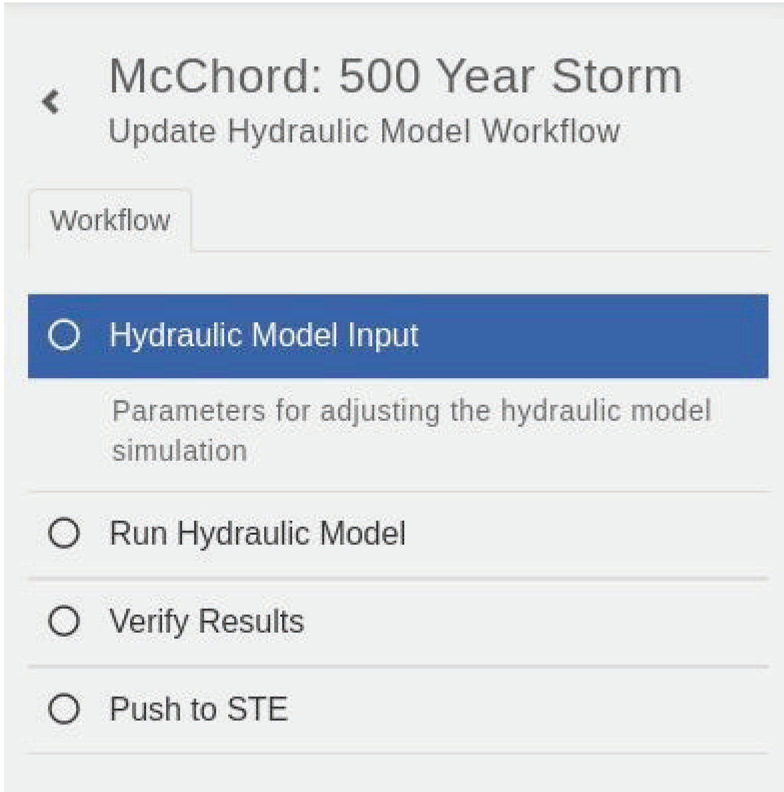


When creating new workflows, users provide a name for the workflow and select the type from a drop-down list. In the demonstration version of STEEM, the types of workflows available are: Update Hydraulic Model, Update Mobility Model, Update Coastal Model, Update Weather Model, and Update Drone Model. All but the Update Hydraulic Model workflows are placeholders included to illustrate the possibility of multiple workflows. None of the workflows shown are required in a full build-out, but are for illustration purposes only. Another important distinction is that a workflow is not constrained to use only one model as suggested by the names of these placeholder workflows, but could use multiple models if necessary to produce the data needed by the STE.

2.4 Update hydraulic model workflow

The Update Hydraulic Model workflow was created to demonstrate the types of workflows that are possible in STEEM. This workflow has four steps: Hydraulic Model Input, Run Hydraulic Model, Verify Results, and Push to STE (Figure 10). Each step is described in the following sections.

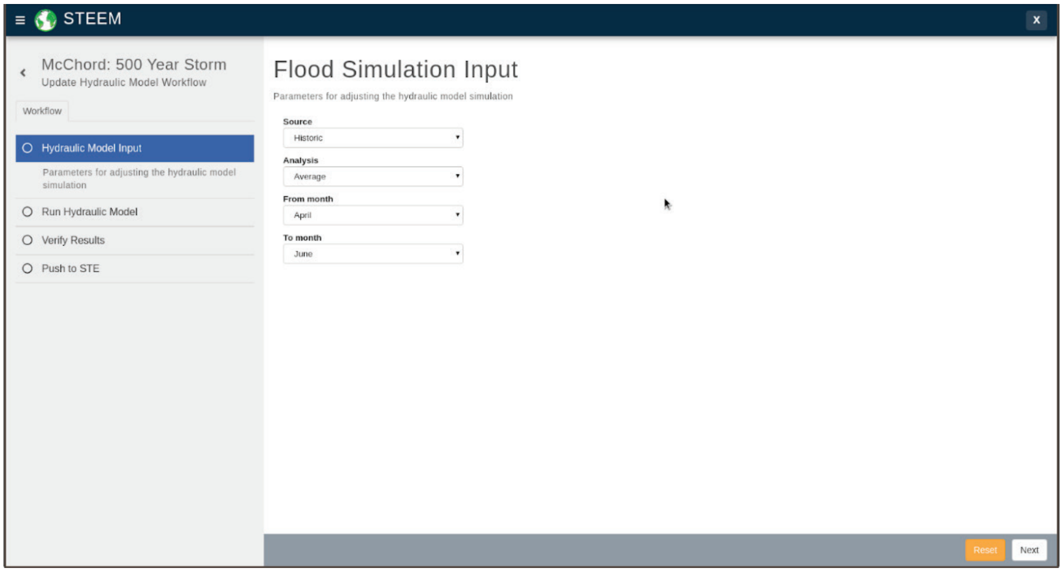
Figure 10. The update hydraulic model workflow steps with the first step active.



2.4.1 Hydraulic model input

The Hydraulic Model Input step is a Form Input Step that prompts the user to define a rainfall scenario for the hydraulic model. The different options are grouped into three different sources or categories: Historic, Forecast, or Existing Scenario. Once a source is selected, other options appear to allow users to further refine their selection. For example, after selecting the Historic source, additional options appear to allow the user to select the type of analysis and the beginning and ending month of the historic analysis (Figure 11). Similarly, users select from a list of scenarios (e.g. 1 yr, 100 yr, 500 yr) when using the Existing Scenario source or they select from a list of available forecasts (e.g. 24 hr, 48 hr) when using the Forecast source.

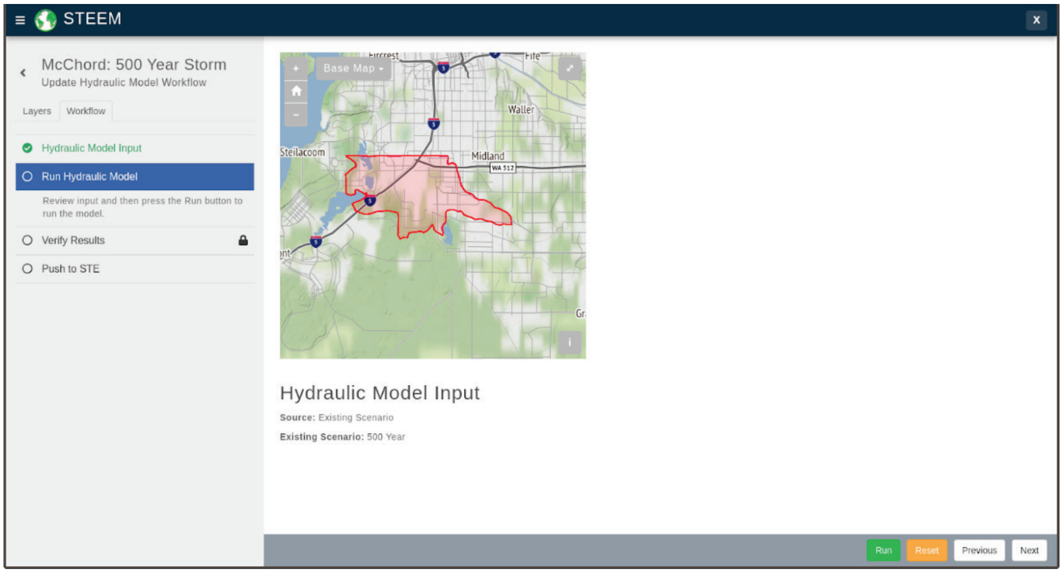
Figure 11. Screenshot of the hydraulic model input step showing the historic source selected.



2.4.2 Run hydraulic model

The Run Hydraulic Model step is an Execution Step. It initially provides a summary of the input provided by the user in all previous steps (Figure 12). This summary view is automatically generated for all steps preceding an Execution Step. After reviewing the summary of input, the user can submit the job to modify and run the model by pressing the “Run” button at the bottom of the page.

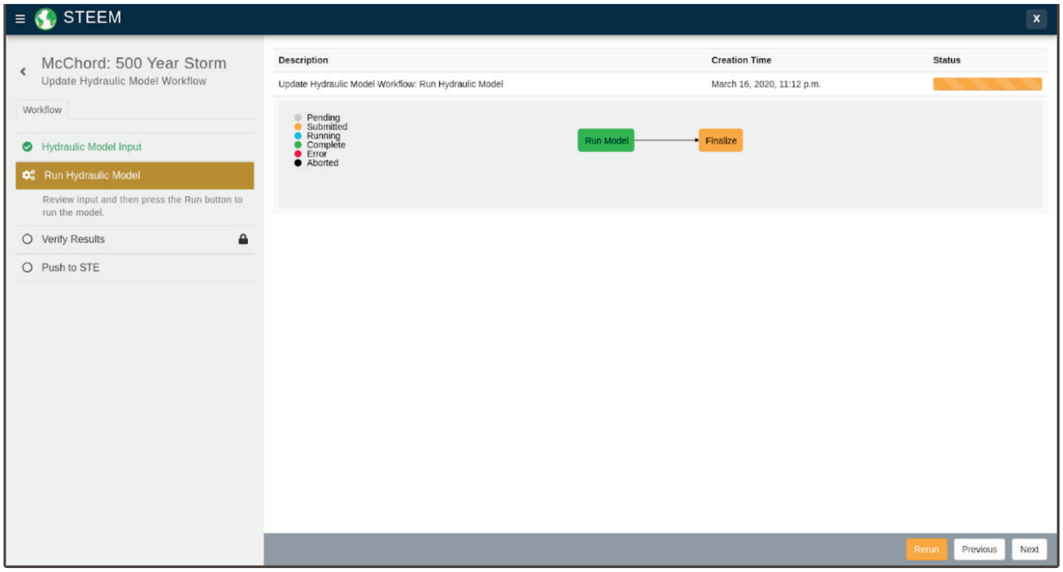
Figure 12. Screenshot of the input review page on the run hydraulic model step.



Once the job has been submitted, the step will display the status of the running job (Figure 13). It is not necessary for users to monitor the jobs that are running. This would not be practical because some of the models can take hours to run, even on High Performance Computing (HPC) resources. The application can be safely closed and revisited at a later time to check on the status of the running job. All results will be saved upon completion of the job for later review.

The demonstration version of the Update Hydraulic Model workflow does not currently modify the ADH model or run to allow for quick demonstrations. However, it does execute a simple script when the job is run. This script matches the options selected by the user to pre-computed ADH output for the Mission Area. The initial implementation only includes pre-computed output for the McChord and Leonard Wood Mission Areas for the 100 and 500 yr existing scenario options. The job will still run for other Mission Areas and other meteorological scenarios, but will default to showing the output for the 100 yr scenario.

Figure 13. Screenshot of the job status page on the run hydraulic model step.



2.5 Verify results

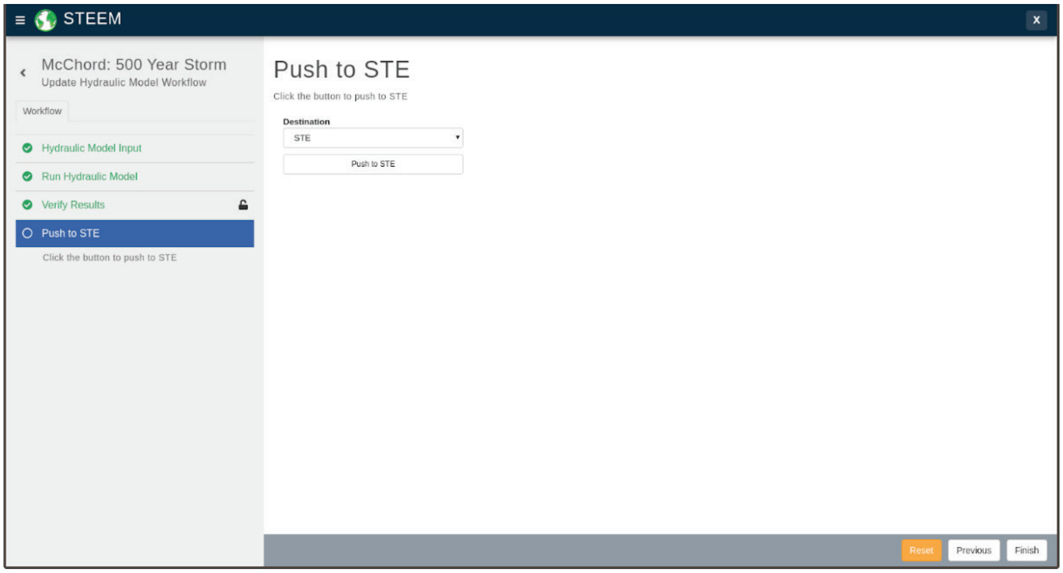
The Verify Results step is a Result Step that provides several tabs with visualizations of the different outputs for the user to review before proceeding (Figure 14). The results that are shown in the Update Hydraulic Model workflow include depth, velocity, and a summary table from the AdH output. The depth and velocity output are visualized on Cesium Globe maps with 3D terrain and VR mode enabled to provide a more intuitive visual for non-technical users.

Figure 14. Screenshot of the verify results step showing the depth results tab.

2.6 Push to STE

The Push to STE step is currently a placeholder step because there is currently no STE to which the data can be pushed. It is currently implemented using a Form Input Step (Figure 15), but will likely be implemented as another Execution Step in a full build-out of STEEM. This step is meant to collect the output from the AdH model that is needed by the STE, reformat the data to a format that the STE can recognize, and push the data to a place where the STE can use it for rendering the virtual environment.

Figure 15. Screenshot of the verify results step showing the depth results tab.

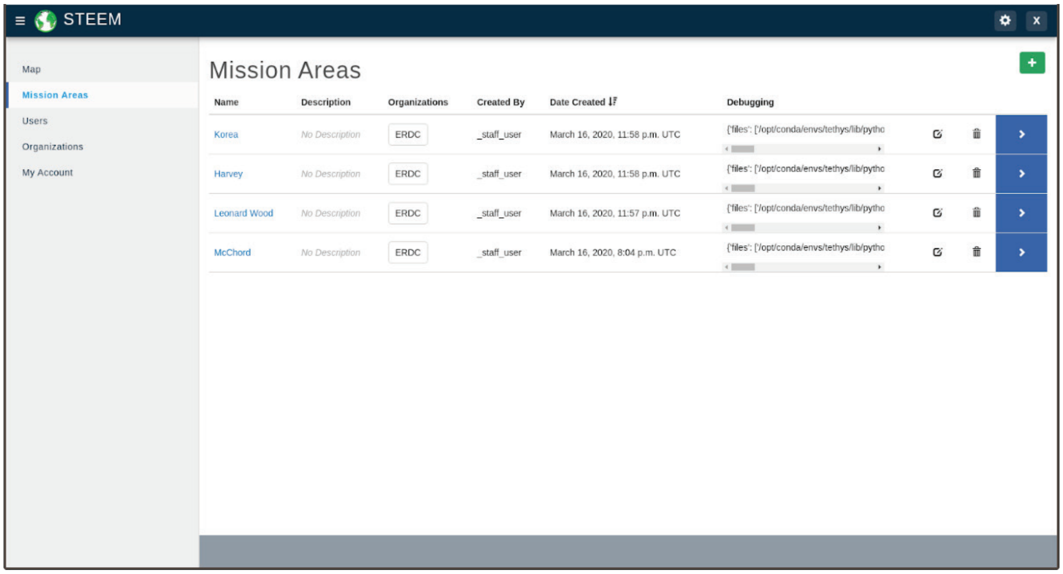


2.7 Administration

The STEEM web application includes several administration pages that can be used to manage the Mission Areas, Organizations, and User accounts. The Mission Areas administration page (Figure 16) contains a list of each Mission Area loaded into the application. From this page administrators can create, edit, and delete Mission Areas.

Similarly, the Organizations and Users administration pages are used to add and remove Organizations and User accounts, respectively. Organizations are the primary mechanism of controlling access to Mission Areas. Users can only access Mission Areas that are assigned to the organization(s) to which they belong.

Figure 16. Screenshot of the mission areas administration page for STEEM.



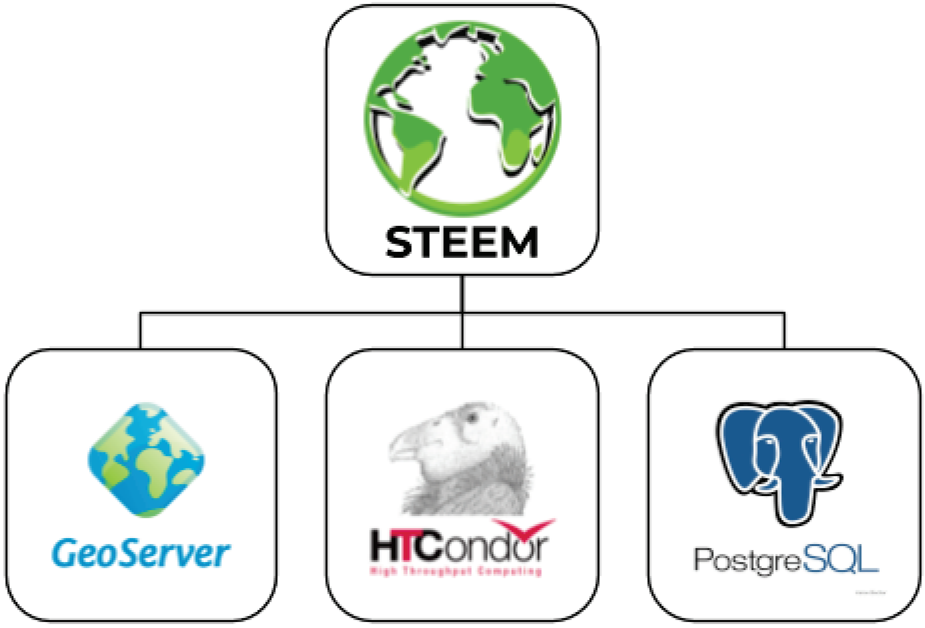
2.8 Deployment

The demonstration STEEM application was deployed to two virtual machines on the Research and Development Enterprise (RDE) Cloud Computing Environment (CCE) system and on an NVIDIA DGX-1 system that had been setup at the HPC for testing. The DGX-1 system deployment was an important test for deployment because that system will eventually be used in the portable supercomputer technology that is currently being developed for the STE. The following sections describe the methodology employed to deploy STEEM on these systems.

2.9 Docker

The STEEM web application has four primary software components: PostgreSQL database, GeoServer, HTCondor, and the STEEM web application (Figure 17). The PostgreSQL database is used by STEEM to store the workflow data and user account information. The GeoServer component hosts the spatial visualization services, such as spatial results from model runs. The HTCondor component handles job scheduling and contains many of the dependencies needed by the execution jobs. The STEEM component contains the web server and all of the application logic.

Figure 17. The four software components of the STEEM web application.



The Docker containerization system was used to streamline the deployment of all of STEEM software components. A Docker container is a standard unit of software that packages up code and all its dependencies so the application runs quickly and reliably from one computing environment to another.[1] Container images are smaller than virtual machine images and can be run on most Linux distributions. Using Docker makes the deployment process a two-step process: (1) download images for each software component and (2) run new containers from each image.

Custom Docker images were developed for the STEEM and HTCondor software components. Community-developed Docker images from Docker Hub were used for the PostgreSQL and GeoServer components. A list of each of the Docker images needed to run STEEM and the name of the image that is used to download it is shown in Table 1.

1 "What is a Container?". Docker.com. Web. 2020 Mar.16. https://www.docker.com/resources/what-container.

Table 1. Software components and Docker image names.

Software Component	Docker Image Name
STEEM Web Application	docker.aquaveo.com/tethys/steem/tethysapp-steem
STEEM HTCondor	docker.aquaveo.com/tethys/steem/steem-condor
PostgreSQL	mdillon/postgis
GeoServer	tethysplatform/geoserver

2.9.1 Docker compose

A Docker deployment with multiple Docker containers requires a small amount of additional work to properly configure the containers to connect to one another. Although it is not hard to do this manually, it is fairly tedious. Docker compose is a tool that streamlines the process of defining and running multi-container Docker applications.[1] Using Compose, the developer uses a YAML file to configure each container required by the application. Then, the application and all of the containers it depends on can be started with a single command.

A Docker Compose YAML file was developed for STEEM to simplify the deployment. Instructions for how to deploy STEEM using Docker Compose are included in a README in the source code and can also be found in Appendix A. The Docker Compose configuration file is included in the source code for the application and the latest version at the time of writing is included in Appendix B for reference.

[1] "Overview of Docker Compose". Docker.com.Web.2020 Mar. 16. https://docs,.docker.com/compose/.

3 Future Work

3.1 Overview

As previously mentioned, the STEEM application that has been developed is a demonstration application. A full build-out of the application or the build-out of an application with similar functionality would require some additional work. Some work that needs to be completed depends on the implementation details of the STE. However, many capabilities demonstrated in STEEM could easily be applied to other ERDC applications. This section contains recommendations for future work on STEEM.

3.1.1 Offline safe

The current implementation of the STEEM application is not equipped to be used in an offline environment. It depends on many external resources for proper functioning. This includes JavaScript dependencies, Cascading Style Sheets (CSS) dependencies, and basemap services. For the application to function properly in an off-grid portable supercomputer, the JavaScript and CSS dependencies will need to be hosted by the application. The basemap services will also need to be hosted by a local map server and the application will need to be reconfigured to use the local map services. Additionally, any dataset services that the workflows depend on will need to be made available locally as well.

3.2 Additional step types

A full build-out of the STEEM web application may require the development of additional workflow step types and improvements to some existing step types. The following sections discuss these recommendations.

3.2.1 User Interface Toolkit (UIT) plus execution step

The demonstration application uses the HTCondor Execution Step to submit jobs to the HTCondor scheduler, which handles scheduling and orchestration of the jobs on the system. It is recommended that an additional UIT Plus Execution Step be developed to make it possible for the application to submit jobs directly to the HPC using the UIT Plus REST API.

The development of the UIT Plus Execution Step was begun during the development of STEEM, but not completed. The HTCondor Execution Step was prioritized because the target deployment was the portable supercomputer system. On this system, the application would be deployed on the same machine that would perform the computation and the HTCondor Execution Step lended itself better to that environment.

3.2.2 Additional results steps

There is currently no Result Step that supports plotting data, but this is a baseline capability that is anticipated will be needed by future workflows. The framework STEEM is built on supports several options for plotting that could be used to implement a Plot Result Step including Highcharts, Bokeh, and Plotly.

The Report Result Step would provide a summary of all results tabs on a Result view. It would automatically compile the results and present them in a format suitable for printing or sharing. This view would also provide an option for downloading the report as a PDF.

3.2.3 Spatial result step improvements

There are a few improvements that could be made to the Spatial Result Step. One feature that was mentioned during demonstrations of STEEM was a legend for the layers being displayed. Another related feature would be symbology controls that would allow the user to change the symbology of the rendered layers.

The current implementation of the Spatial Result Step supports rendering spatial results on either an OpenLayers 2D map or a Cesium 3D map. For the Cesium 3D map renderer, the built-in VR mode is a good first pass, but is suboptimal for VR previews. It only supports Google Cardboard and seems to only work on Android devices. It also does not pick up on device orientation or rotate the view as the user looks around. There are extensions for Cesium that would add support for more capable VR systems like Oculus that would be more appropriate for the VR preview capability.[1]

[1] https://github.com/jsanchezfr/cesium-oculus-plugin.

Additionally, the Cesium map renderer for the Spatial Result Step currently only supports GeoJSON or Web Map Service (WMS) service layers, while the OpenLayers renderer supports many other layers and services. If it is anticipated that other formats would be needed (e.g. ArcGIS REST, KML, CZML, WFS, WCS), these would need to be added to the Cesium renderer implementation to be able to render those data sources on a 3D Spatial Result Step.

3.3 File data model

The Mission Area File Data Model needs to be fully implemented. While STEEM includes database tables that store metadata about the models and Mission Areas, there is no official data schema for organizing the model and dataset files contained in Mission Areas. The ideal would be to adapt an existing data model to the needs of STEEM. Otherwise, a data model would need to be designed and developed. One recommended technology to investigate would be object storage.

3.4 Mission area initialization and management

The Mission Area initialization and management processes need to be designed and implemented. In the demonstration version of STEEM, the user uploads a file containing GeoJSON that defines the Mission Area boundary when setting up a new Mission Area. There is currently no interface for adding models or datasets to a Mission Area nor pages for managing existing models and datasets. It is recommended that additional tabs or views be added to the Mission Area Details view for managing models and datasets.

Many of the models and datasets will be too large to efficiently upload through an HTML form. One design recommendation is to allow two options for adding models: (1) upload an archive containing model files and (2) select a directory containing model files that has been placed in the Mission Area database manually or through a server-side process. Better still would be to implement automatic detection of models that are added server-side.

3.5 Mission area details views

Additional design is needed for the Mission Area Details views after the Mission Area Data Model has been fully implemented. For example, the information in the Mission Area Summary tab is currently hard-coded for the demonstration. Once a Mission Area file database is in place, this view should display a list of all models and datasets contained in the Mission Area. However, this view could also be simplified or show different information completely. The Mission Area Details view could also contain additional tabs of information such as a tab for each type of model that is included in the Mission Area or additional tabs with metadata about the Mission Area.

3.6 Define workflows

The initial set of STE workflows need to be identified, designed, and implemented. Most of the workflows listed in the demonstration version of STEEM were placeholders. If the Update Hydraulic Model workflow is identified as one of the initial workflows to implement, it will need to be fully implemented. This would involve developing the logic that modifies the model files, executes the model, and post-processes the results. The process for implementing other workflows will be similar, with the addition of building out the steps for the workflow.

3.7 ATCore workflows

The workflow framework used by STEEM is provided by a Tethys Platform extension called ATCore that is developed by Aquaveo. The decision was made to use ATCore to make it possible to develop the demonstration application in the constrained time frame allotted for development of the demonstration application.

Aquaveo has granted ERDC an unofficial license to use ATCore for the STEEM demonstration application and it is included in the Docker image deliverables. However, for a full build out of the STEEM application or development of additional applications that use the workflow framework it is recommended that ATCore be officially licensed or that it be purchased by ERDC and released under an open-source license.

3.8 Summary

Below is a summary of the recommended future tasks for STEEM development:

- Implement an offline safe mode hosting dependencies, basemaps, and datasets locally.
- Add additional step types: UIT Plus Execution Step, Plot Result Step, and Report Result Step
- Make improvements to the Spatial Result Step: add a legend and implement a better VR preview for Cesium 3D maps.
- Design and implement a file data model and database for the Mission Area model and dataset files.
- Design and implement Mission Area initialization and model/database management views and workflows.
- Design and implement the Mission Area Details views.
- Identify the initial STE workflow(s), then design and implement them.
- Officially license ATCore or purchase and release it under an open-source license.

4 Conclusions and Recommendations

A demonstration implementation of the STEEM web application was successfully developed and deployed. The development was accomplished efficiently because it leveraged many existing technologies that had been developed or improved through the ERS Research and Development Area. Among the technologies used by STEEM that have been developed or improved by ERS are Tethys Platform, Cesium 3D maps, Bokeh, and Panel.

The STEEM web application provides a promising interface for managing modeling and data resources used by the Synthetic Training Environment. Users with no prior knowledge of the environmental modeling techniques can use STEEM to generate virtual scenarios based on physics-based simulations that can be readily consumed by the STE.

More broadly, STEEM demonstrates a general pattern of how web-based workflows can be used to simplify common modeling exercises. This pattern is general enough to be applied to any modeling or computational exercise and is not limited to environmental applications.

As part of the demonstration, STEEM was deployed on two different ERDC systems including two virtual machines on the RDE Cloud Computing Environment (CCE) and on an NVIDIA DGX-1 machine. The latter is the system being a potential component for the portable supercomputer technology currently under development and demonstrated that STEEM would be compatible with that system. Docker container technology was used to deploy STEEM, which grants it the ability to be easily deployed on any Linux-based system.

As a demonstration application, there is quite a bit of work that could be done for a full build-out of STEEM. However, even without a full implementation of STEEM, the generalized web-based workflow pattern should be investigated as a potential tool to be used in other projects. Projects that could take advantage of this pattern would include any project that uses numerical modeling or other computations in need of a simple interface for non-technical users.

Appendix A: STEEM Deployment Instructions

1. Create the `db`, `keys`, and `workspaces` directories that will be used by the STEEM deployment:

```
$ sudo mkdir -p /var/lib/steem/db

$ sudo mkdir -p /var/lib/steem/keys
```

NOTES:

- Different locations can be used for these directories if the Docker Compose script is also updated to match.
- These directories persist the data used by STEEM and should be included in a backup for production versions of the STEEM application.

2. Generate the SSH keys needed by STEEM and rename them to what STEEM expects:

```
$ cd /var/lib/steem/keys

$ ssh-keygen -N steem -f condorkey
```

3. Run Docker Compose command in the directory containing the Docker Compose YAML file:

```
$ docker-compose up
```

Appendix B: STEEM Docker Compose Configuration

```
version: "3.2"
services: web:
    image: docker.aquaveo.com/tethys/steem/tethysapp-
    steem:latest
    restart: always
    depends_on:
      - "db"
      - "geoserver"
    networks:
      - "internal"
      - "external"
    ports:
      - "80:80"
    environment:
      ALLOWED_HOSTS: "127.0.0.1"
      ASGI_P
      ROCESS
      ES: 1
      APP_DB
      _HOST:
      "db"
      APP_DB
      _PORT:
      "5432"
      APP_DB_USERNAME: "tethys_super"
      APP_DB_PASSWORD: "pass"
      TETHYS_PUBLIC_HOST: "localhost"
      TETHYS_DB_HOST: "db"
      TETHYS_DB_USERNAME: "tethys_default"
      TETHYS_DB_PASSWORD: "pass"
      TETHYS_DB_SUPERUSER:
      "tethys_super"
      TETHYS_DB_SUPERUSER_PA
      SS: "pass"
      POSTGRES_PASSWORD:
      "pass" TETHYS_DB_PORT:
```

```
"5432"
TETHYS_GS_USERNAME:
"admin"
TETHYS_GS_PASSWORD: "geoserver"
TETHYS_GS_PROTOCOL: 'http'
TETHYS_GS_HOST: "geoserver"
TETHYS_GS_PORT: "8181"
TETHYS_GS_HOST_PUB: "localhost"
TETHYS_GS_PORT_PUB: "8181"
```

```
      TETHYS_GS_PROTOCOL_PUB: "http"
      TETHYS_CLUSTER_IP:
      "condor"
      TETHYS_CLUSTER_USERNAME:
      "condor"
      TETHYS_CLUSTER_PKEY_PASS
      WORD: "steem"
      CONDOR_PRIVATE_KEY: "/usr/lib/tethys/keys/condorkey"
    volumes:
      - type: bind
        source: /var/lib/steem/workspaces
        target: /var/lib/tethys_persist/workspaces
      - type: bind
        source: /var/lib/steem/keys
        target: /tmp/keys
  db:
    image: mdillon/postgis:latest
    restart: always
    networks:
      - "internal"
    ports:
      - "5432:5432"
    environment:
      POSTGRES_USER: "postgres"
      POSTGRES_PASSWORD: "pass"
    volumes:
      - type: bind
        source: /var/lib/steem/db
        target: /var/lib/postgresql/data
  geoserver:
    image: ciwater/geoserver:latest
    restart: always
    networks:
      - "internal"
      - "external"
    ports:
      - "8181:8181"
      - "8081:8081"
      - "8082:8082"
```

```
      - "8083:8083"
      - "8084:8084"
```

```
        Environment:
        ENABLED_NODES: '4'
        REST_NODES: '1'
        MAX_MEMORY: '1024'
        MIN_MEMORY: '512'
        NUM_CORES: '8'
        MAX_TIMEOUT: '60'
    condor:
      image: docker.aquaveo.com/tethys/steem/steem-
      condor:latest
      restart: always
      networks:
        - "internal"
      ports:
        - "22:22"
      volumes:
        - type: bind
          source: /var/lib/steem/keys
          target: /var/lib/condor/keys
networks:
      internal: true
    external:
```

Acronyms and Abbreviations

ADH	Adaptive Hydraulics Model
API	Application Programming Interface
CCE	Cloud computing Environment
CSS	Cascading Style Sheets
DoD	Department of Defense
ERDC	U.S. Army Engineer Research and Development Center
ERS	Engineered Resilient Systems
HPC	High Performance Computing
HTML	Hypertext Markup Language
IT	Information technology
ITL	Information Technology Laboratory
RDE	Research and Development Enterprise
SEID	Software Engineering and Informatics Division
SQL	Structured Query Language
STE	Synthetic Training Environment
STEEM	STE Environmental Manager
3D	Three-dimensional
2D	Two-dimensional
VR	Virtual reality

UIT	User Interface Toolkit
U.S	United States
WMS	Web Map Service
YAML	YAML Ain't Markup Language

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
September 2020	Final	

4. TITLE AND SUBTITLE

STE Environmental Manager (STEEM) Demonstration Web Application

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Jonathan Boone, Nathan Swain, Gage Larsen, and Michael Souffront

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Information Technology Laboratory
U.S. Army Engineer Research and Development Center
3909 Halls Ferry Road, Vicksburg, MS 39180-6199

Aquaveo, LLC
3210 N. Canyon Road
Suite 300, Provo, Utah 84604

8. PERFORMING ORGANIZATION REPORT NUMBER

ERDC/ITL TR-20-2

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Information Technology Laboratory (ITL)
3909 Halls Ferry Road
Vicksburg, MS 39180-6199

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

Engineered Resilient System RDA, Deployable Computational Capability for STE, FWIC:8f523J

14. ABSTRACT

This report provides a summary of the development of the Synthetic Training Environment (STE) Environmental Manager (STEEM) demonstration web application. The purpose of this web application is twofold: (1) demonstrate a web application that enables non-technical users to prepare, run, and manage the physics-based models used by the STE to simulate realistic environmental conditions and (2) show how technologies developed by the Engineered Resilient Systems (ERS) Research and Development Area can be used to rapidly create applications to support U.S. Army Engineer Research and Development Center (ERDC) programs like the STE. A full build-out of STEEM would leverage the following ERS-developed technologies: data services, model development environment tools, model coupling/interface API, simulation workflow manager, and scenario generation tools.

15. SUBJECT TERMS

Environmental management – Computer simulation
Environmental conditions
Web applications
Synthetic training devices
Systems engineering

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			19b. TELEPHONE NUMBER *(include area code)*
Unclassified	Unclassified	Unclassified	SAR	42	