NOAA AI-BASED 3D EARTH AND SPACE OBSERVING DIGITAL TWIN (EO-DT)

Contract # 1332KP22CNEEP0011

Prepared for NOAA, NESDIS Office of Systems Architecture and Engineering (SAE), Joint Venture Partnerships

This material is based upon work supported by the Dept of Commerce under Contract No. 1332KP22CNEEP0011. Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Dept of Commerce.

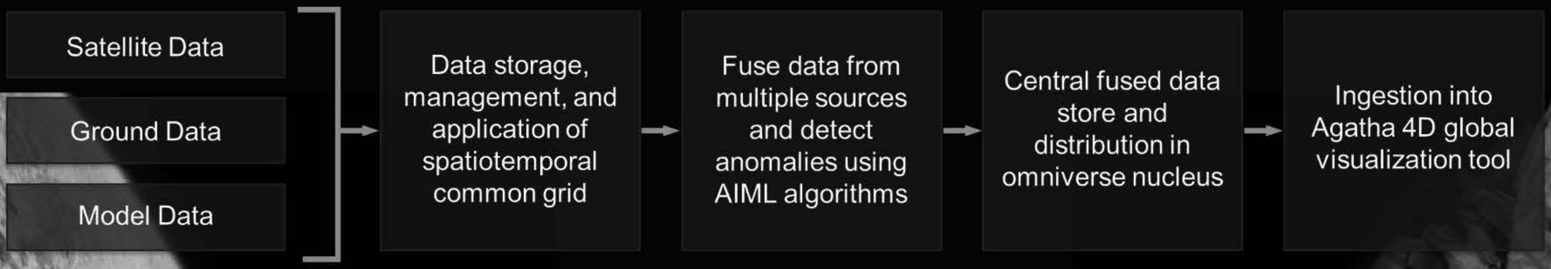
AI-Driven Earth and Space Observing Digital Twin

OR3D™

LOCKHEED MARTIN

NVIDIA. LOCKHEED MARTIN

AGATHA



Satellite Data

Ground Data

Model Data

Data storage, management, and application of spatiotemporal common grid

Fuse data from multiple sources and detect anomalies using AIML algorithms

Central fused data store and distribution in omniverse nucleus

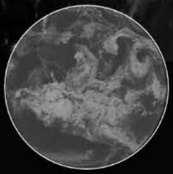
Ingestion into Agatha 4D global visualization tool

Lockheed Martin (LM) and NVIDIA are partnered together in a 2-year contract with NOAA NESDIS Joint Venture Partnerships for the AI-Driven Earth and Space Observing Digital Twin (EODT) prototype study and demonstration.

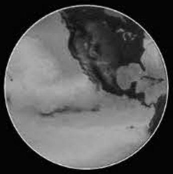
Our EODT provides NOAA a high-resolution, accurate, and near-real-time depiction of global conditions using space and ground-based observations and model output.

The EODT framework integrates three major components (OpenRosetta3D, Omniverse, Agatha). All software have high TRL and have been proven on past programs. We utilize AI/ML for data fusion and anomaly detection.

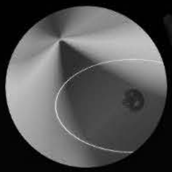
Our plug-and-play architecture enables users to ingest and build disparate data layers to provide current conditions and actionable insights.



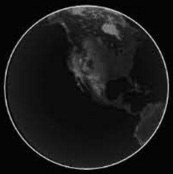
Atmosphere
(Temperature and
Moisture Profiles)



Ocean
(Sea Surface
Temperature)



Space Weather
(Solar Wind Bulk
Plasma)



Cryosphere
(Sea Ice Concentration)

Land and Hydrology
(Fire Products)

TABLE OF CONTENTS

Section **Page**

1 INTRODUCTION....1
1.1 DIGITAL TWINS AND HOW THEY CAN BENEFIT NOAA 1
1.2 DIGITAL TWIN OVERVIEW....3
1.2.1 OpenRosetta3D: AI-Enabled Data Processing and Orchestration Engine 4
1.2.2 Omniverse Nucleus: Digital Twin Distributed Collaboration Platform....4
1.2.3 Agatha: 4D Interactive User Visualization Platform....4
2 DIGITAL TWIN DATA....5
2.1 EO-DT DATA INPUT SERVICE....6
3 DIGITAL TWIN FRAMEWORK 7
3.1 NOAA PROCESSING ENGINE DATA PROCESSING PIPELINES 7
3.1.1 Static Data Processing Pipeline 8
3.1.2 Live Data Processing Pipeline....8
3.1.3 NOAA Processing Engine Algorithm Applications 9
3.1.4 NOAA Processing Engine: Data Processing, Data Fusion, and Anomaly Detection....10
3.2 NOAA PROCESSING ENGINE: TILING INTO UBER H3 17
3.3 NOAA PROCESSING ENGINE: FORMATTING INTO OPENUSD....18
3.4 NOAA PROCESSING ENGINE: AGGREGATION SCHEME 18
3.5 NOAA PROCESSING ENGINE FILE TRANSFER TO OMNIVERSE NUCLEUS....18
3.6 OMNIVERSE NUCLEUS....19
3.7 NOAA PROCESSING PIPELINE METRICS....19
3.8 OMNIVERSE NUCLEUS TO AGATHA FILE TRANSFER....20
3.9 AGATHA VISUALIZATION INTERFACE AND FEATURES....21
4 PROTOTYPE AND ESTIMATED OPERATIONAL COSTS....23
5 INTEROPERABILITY AND STANDARDIZATION WITH OTHER....26
5.1 FLEXIBILITY AND INTEROPERABILITY OF THE LM-NVIDIA EO-DT 27
5.2 INTEROPERABILITY WITH OTHER DIGITAL TWINS....30
5.3 FEDERATED SYSTEM OF DIGITAL TWINS 32
6 RECOMMENDATIONS AND LESSONS LEARNED 33
7 REFERENCES....39

LIST OF FIGURES

Figure **Page**

Figure 1-1 NASA AIST Definition of a Digital Twin. 2
Figure 1-2 National Academies Definitions of a Digital Twin. 2
Figure 1-3 NOAA's Vision for EO-DT. 3
Figure 1-4 Component Breakdown of LM-NVIDIA's AI-Based EO-DT Architecture. 4
Figure 3-1 Diagram Showing the Static Data Pipeline 7
Figure 3-2 Diagram Showing the Live Data Pipeline.. 9
Figure 3-3 Diagram Showing the Containerized Algorithm Pipeline. 10
Figure 3-4 SST Measurements. 11
Figure 3-5 SST Data Fusion, Anomaly Detection, # of Sensors, and Standard Deviation. 12
Figure 3-6 Insert Caption Here. 12
Figure 3-7 IGRA and MIIDAPS data. 13
Figure 3-8 GFS Model Output of Temperature at Four Vertical Levels. 13
Figure 3-9 GFS Model Output of Humidity at Four Vertical Levels. 14
Figure 3-10 Five Earth Panels. GFS (SP), GFS (NP), AMSR2 (NP), AMSR-2 (SP), Blended AMSR2/VIIRS (NP). 15
Figure 3-11 Fused Sea Ice Concentration from GFS and the blended VIIRS/AMSR2 product, Anomalous SIC, Number of Sensors, and standard deviation of SIC. 15
Figure 3-12 GOES (left) and VIIRS (right) Fire Data. 16
Figure 3-13 Solar Wind Magnetic Shear Derived from DSCOVR Observations. 17
Figure 3-14 Uber H3 Tiling System. 17
Figure 3-15 Highlighting Agatha's Features of the Global Interface.. 22
Figure 3-16 Highlighting Agatha's Features. 22
Figure 5-1 Recommended Digital Twin High-Level Architecture. 27
Figure 5-2 EO-DT Data Displayed in Three Separate Viewers a) Agatha, b) Earth-2, and c) Pixar USD to Demonstrate Interoperability with the OpenUSD File Format. 30
Figure 5-3 Example Federated System of Digital Twins. 32

LIST OF TABLES

Table **Page**

Table 2-1 EO-DT Input Variables and Sensors........5
Table 2-2 EO-DT Input Data Spatial and Temporal Resolution and Coverage........5
Table 2-3 EO-DT Data Sources for each Geophysical Variable Sensor........6
Table 3-1 SST........19
Table 3-2 Fire. 20
Table 3-3 SIC. 20
Table 3-4 ATMS. 20
Table 3-5 Space Weather. 20
Table 3-6 Standard and Custom Features Implemented in Agatha........21
Table 4-1 AWS Configuration Parameters of Interest. 23
Table 4-2 SST Hourly Live Processing Cost. 24
Table 4-3 2-week Data Set Processing Cost. 25
Table 6-1 High-Level Overview of Recommendations. 33
Table 6-2 Detailed Recommendations with References to Goals Provided in Section 1.2.2........34

APPENDICES

A .... See Attached

Study Authors

Principal Investigator

Dr. Lynn Montgomery, Staff AI Research Engineer, Lockheed Martin

Report Co-Authors

Brandon Ballance, Senior AI SW & Systems Architect, Lockheed Martin

Chandler Caldwell, Staff Project Engineer, Lockheed Martin

Gary Hoffmann, Systems Engineer Senior Staff, Lockheed Martin

Andrew Linenfelser, Senior Software Engineer, Lockheed Martin

Jared Lance, Software Engineer Principal, Lockheed Martin

John Stone, Distinguished Engineer, Scientific Visualization DevTech, NVIDIA

Thomas Kaye, Senior Solution Architect, NVIDIA

Special Thanks

(NOAA) Ramesh Rangachar, Beau Backus, Eric Maddy, Ryan Berkheimer, Kesha Hayes, Rob Redmon, Flavio Iturbide-Sanchez, Lynn Mayo, Jennifer Webster

(NASA) Sid Boukabara, Jaqueline Le Moigne, Benjamin Smith

(AWS) Anant Pradhan, John Kolman, Matthew Dowling, Dev Jodhun

1 INTRODUCTION

In May 2022, the National Oceanic and Atmospheric Administration (NOAA) National Environmental Satellite, Data, and Information Service (NESDIS) Joint Ventures released a broad agency announcement (BAA) expressing interest in exploring digital twin technologies. The goal of the BAA is for NESDIS to enhance their ability to process, monitor, quality control, consolidate, fuse, and assimilate environmental observations while streamlining satellite data ground processing and dissemination to users and applications. This Earth Observations Digital Twin (EO-DT) could serve as the next-generation ground enterprise system in NESDIS operations which will interface with the Earth system approach modeling effort NOAA is undertaking. The EO-DT must also have an agile, scalable framework to integrate with the rapidly expanding amount of data NOAA must handle from both ground and space-based observations along with model output. In addition, the digital twin will rely on responsible artificial intelligence (AI) and machine learning (ML) tools to process data efficiently and will be designed to serve as an entry point for a wide range of NESDIS operational environmental data users and applications.

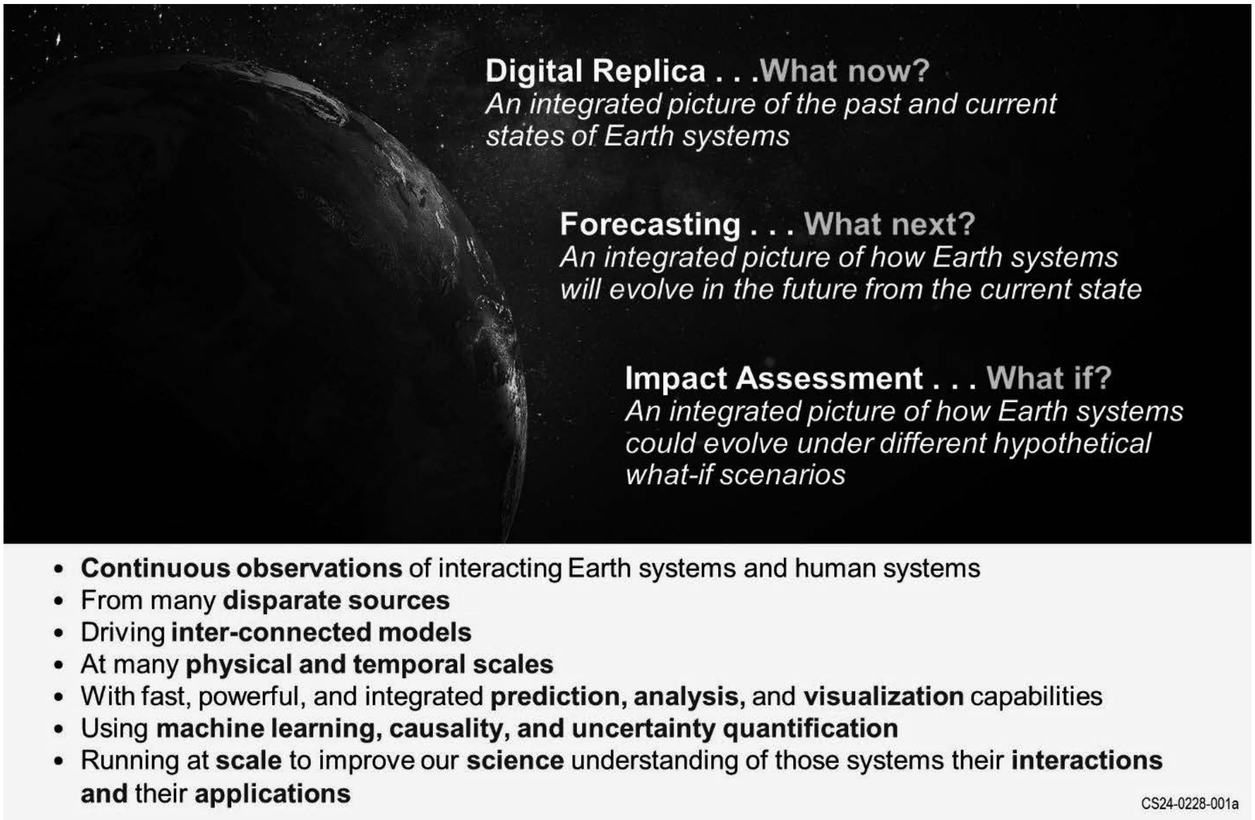
To explore digital twin technologies for this EO-DT use case, Lockheed Martin (LM) and NVIDIA partnered on a 2-year program to build an AI-Driven Earth and Space Observing Digital Twin prototype which ingests, analyzes, and visualizes geophysical data coming from five Earth system domains (atmosphere, ocean, cryosphere, land and hydrology, and space weather). The LM-NVIDIA EO-DT prototype demonstration and study goals included:

1. Provide NOAA with a functioning, scalable prototype that may serve as the foundation of next-generation ground enterprise system.
2. Determine cost estimates for maintaining a digital twin and scaling it to store large amounts of data.
3. Provide recommendations for standardization and interoperability with other digital twins.
4. Study how a digital twin can benefit NOAA as a research and development (R&D) product and an operational product.

In this report, we detail our study of building a functional digital twin prototype and our recommendations to NOAA based on our results. The technological assets we will deliver are outlined in Table A-1 in the Appendix.

1.1 DIGITAL TWINS AND HOW THEY CAN BENEFIT NOAA

Digital twin technology allows users to replicate, analyze, and simulate current, future, and past conditions. The National Academy of Science defines a digital twin as a set of virtual information constructs that mimic the structure, context, and behavior of a natural, engineered, or social system (or system of systems). It dynamically updates with data from its physical twin, has a predictive capability, and informs decisions that realize value (Figure 1-2, National Academies, 2024). Similarly, the National Aeronautics and Space Administration's (NASA) Advanced Information Systems Technology (AIST) team defines an Earth System Digital Twin (ESDT) as an interactive and integrated multidomain, multiscale, digital replica of the state and temporal evolution of Earth systems (Figure 1-1, Le Moigne & Smith, 2022). It dynamically integrates relevant Earth system models and simulations, other relevant models (e.g., related to the world's infrastructure), continuous and timely (including near real-time and direct readout) observations (e.g., space, air, ground, over/underwater, 'Internet of Things,' socioeconomic), long-time records, analytics, and AI tools. (Le Moigne & Smith, 2022)



Digital Replica . . .What now?
An integrated picture of the past and current states of Earth systems

Forecasting . . . What next?
An integrated picture of how Earth systems will evolve in the future from the current state

Impact Assessment . . . What if?
An integrated picture of how Earth systems could evolve under different hypothetical what-if scenarios

- **Continuous observations** of interacting Earth systems and human systems
- From many **disparate sources**
- Driving **inter-connected models**
- At many **physical and temporal scales**
- With fast, powerful, and integrated **prediction, analysis,** and **visualization** capabilities
- Using **machine learning, causality, and uncertainty quantification**
- Running at **scale** to improve our **science** understanding of those systems their **interactions and** their **applications**

CS24-0228-001a

Figure 1-1 NASA AIST Definition of a Digital Twin.
Graphic sourced from AIST ESDT Workshop Report.

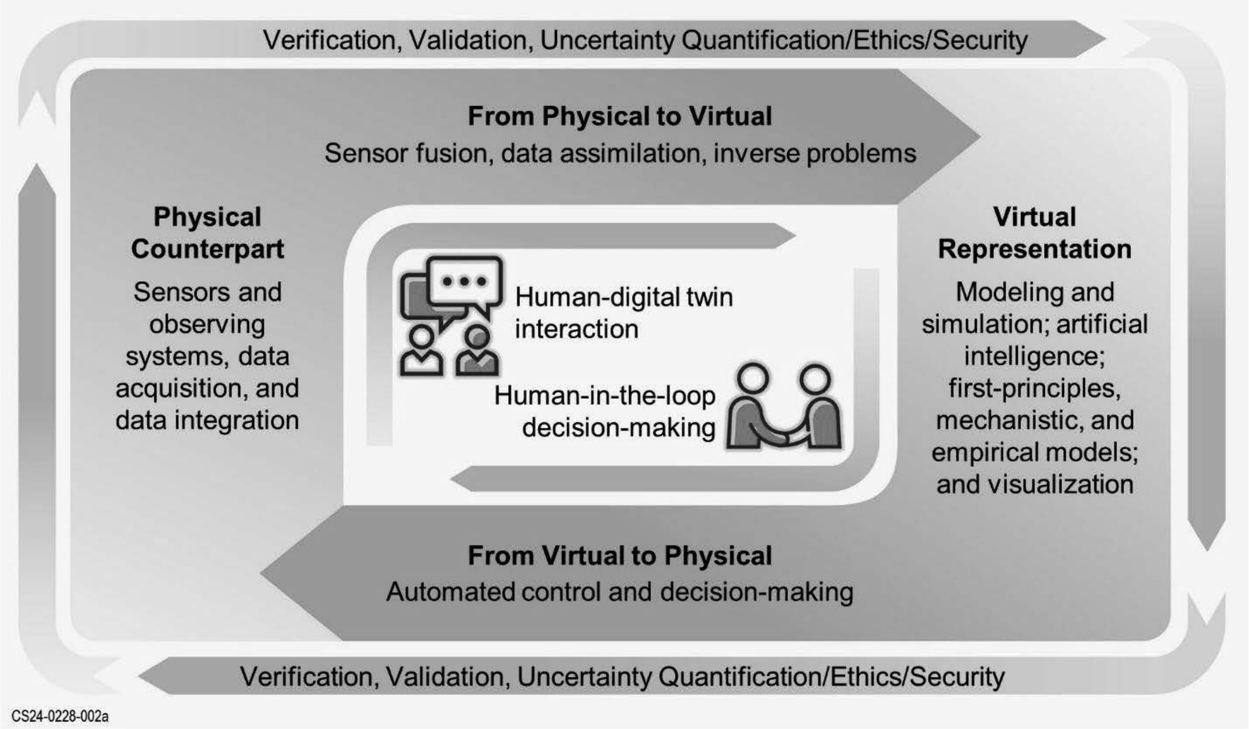

Verification, Validation, Uncertainty Quantification/Ethics/Security
From Physical to Virtual
Sensor fusion, data assimilation, inverse problems
Physical Counterpart
Sensors and observing systems, data acquisition, and data integration
Virtual Representation
Modeling and simulation; artificial intelligence; first-principles, mechanistic, and empirical models; and visualization
Human-digital twin interaction
Human-in-the-loop decision-making
From Virtual to Physical
Automated control and decision-making
Verification, Validation, Uncertainty Quantification/Ethics/Security
CS24-0228-002a

Figure 1-2 National Academies Definitions of a Digital Twin. *Graphic sourced from Foundational Research Gaps and Future Directions for Digital Twins.*

In our prototype, we strove to create a digital twin that supported NOAA NESDIS' mission from this BAA by exploring digital twin technologies. The EO-DT system NOAA NESDIS seeks is an integrated Earth system replica which monitors Earth's environment with multiscale, multivariable features that integrates a large set of observing systems and environment analyses systems (Le Moigne & Smith., 2022, Figure 1-3). It dynamically incorporates Earth system data and observations (primarily satellite and ground-based data) and relies on trustworthy and responsible AI tools, including ML and computer vision. Future EO-DT components include an Earth system model for prediction based on input of the monitoring EO-DT and an assessment component to cover 'what if' scenarios (Le Moigne & Smith., 2022, Figure 1-3). This prototype would benefit NOAA NESDIS by improving the ability to process, monitor, quality control, consolidate, fuse, and assimilate environmental observations. It would also streamline the processing and dissemination of satellite data to users and applications.

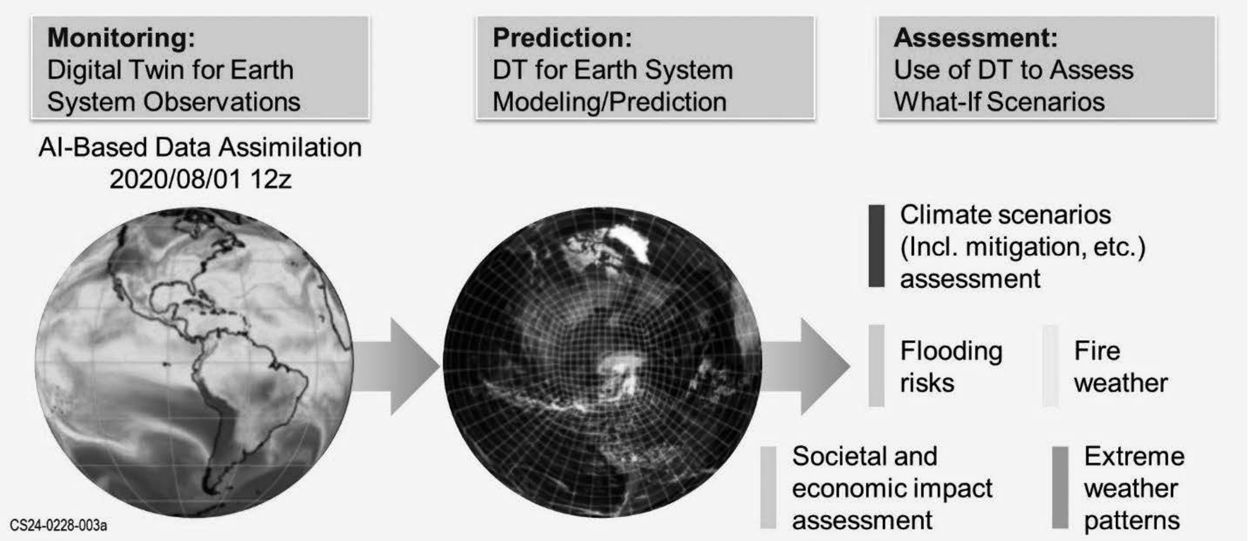

Monitoring:
Digital Twin for Earth System Observations
Prediction:
DT for Earth System Modeling/Prediction
Assessment:
Use of DT to Assess What-If Scenarios
AI-Based Data Assimilation
2020/08/01 12z
Climate scenarios (Incl. mitigation, etc.) assessment
Flooding risks
Fire weather
Societal and economic impact assessment
Extreme weather patterns
CS24-0228-003a

Figure 1-3 NOAA's Vision for EO-DT.
Graphic sourced from AIST ESDT Workshop Report.

1.2 DIGITAL TWIN OVERVIEW

The AI-Based EO-DT prototype that LM and NVIDIA built for NOAA NESDIS integrates mature software to ingest, process, and display geophysical and space weather data in an immersive digital environment. The agile, scalable EO-DT framework processed observations from secure ingest through product generation and data fusion to product distribution to end users. We composed our architecture with three functional components: backend data processing, real-time collaborative file sharing, and customizable scientific visualization (Figure 1-4). OpenRosetta3D™ (OR3D) formats, stores, and applies pre-existing AI/ML fusion and anomaly detection algorithms to data. NVIDIA's Omniverse connects multiple applications to a collaborative real-time environment. Agatha visualizes data from multiple sensors within an interactive 3D Earth and space platform. We designed our flexible architecture to incorporate data from additional and future sensors and enable interoperability with other initiatives through standard interfaces. Our prototype also allows users to view fused geophysical data by geographical area, temporal coverage, and vertical level, providing the foundation for the next-generation enterprise ground system.

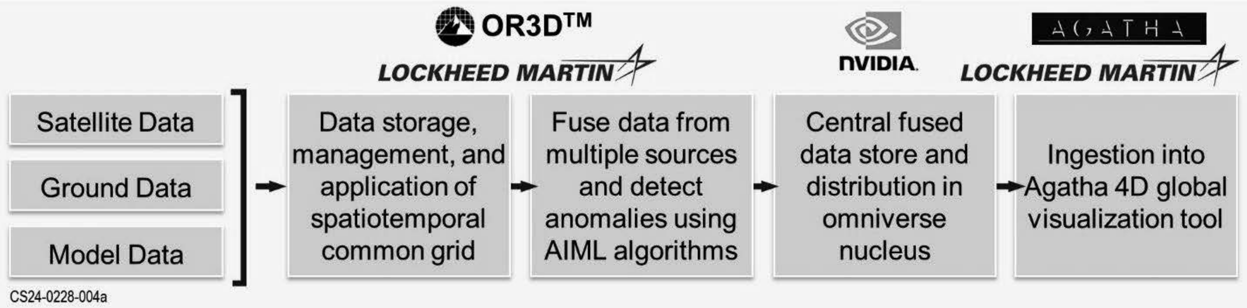

OR3D™
LOCKHEED MARTIN
NVIDIA
AGATHA
LOCKHEED MARTIN
Satellite Data
Ground Data
Model Data
Data storage, management, and application of spatiotemporal common grid
Fuse data from multiple sources and detect anomalies using AIML algorithms
Central fused data store and distribution in omniverse nucleus
Ingestion into Agatha 4D global visualization tool
CS24-0228-004a

Figure 1-4 Component Breakdown of LM-NVIDIA's AI-Based EO-DT Architecture.

1.2.1 OpenRosetta3D: AI-Enabled Data Processing and Orchestration Engine

OR3D is a backend Technology Readiness Level 9 architecture that supports disparate data processing and fusion. OR3D is cloud deployable and leverages common Amazon Web Services (AWS) features and services to orchestrate geospatial workflows for structured and unstructured data. Within our EO-DT architecture, OR3D's primary responsibilities are:

- Workflow orchestration and production management for large-scale data analytics.
- Processing using AWS.
- Formatting of geophysical data.

OR3D develops a customized workflow for each geophysical layer to implement the unique processing required for optimal visualization. Basic processing includes some combination of format interpretation, decoding of data streams, filtering, tiling geographic reprojection, and metadata mapping. Advanced processing includes developing, adapting, and implementing AI/ML algorithms to fuse data and detect anomalies for each geophysical variable.

1.2.2 Omniverse Nucleus: Digital Twin Distributed Collaboration Platform

NVIDIA Omniverse is a platform for developing digital twins in a distributed, collaborative environment. Developed to satisfy the collaboration needs of global, multidisciplinary organizations, Omniverse provides components to build and execute digital twins while satisfying the need for real-time interactivity and scalability. Omniverse has gained traction as the digital twin platform of choice across a broad range of domains, including media and entertainment, architecture and engineering, manufacturing, smart cities, robotics, autonomous vehicles, and multiple scientific use cases. For our EO-DT use case, the Omniverse Nucleus service subscribes to data layer updates made from OR3D and acts as a data server that can broadcast updates to a single instance of Agatha, multiple instances of Agatha, or to another application that has a connecter to the Omniverse Nucleus service.

1.2.3 Agatha: 4D Interactive User Visualization Platform

Agatha is an LM-developed composable data pipeline and visualization platform. It enables users to configure and control how that data is displayed on an interactive 4D canvas (including space and time). For geospatial data and our EO-DT use case, Agatha allows users to overlay multiple datasets directly onto the Earth or in orbit around the Earth. For each geophysical variable, we built custom visualization tools for an intuitive user interface and experience.

2 DIGITAL TWIN DATA

To prove out our prototype, we integrated a broad range of environmental variables. We selected datasets that allowed the use of a variety of geospatial and AI/ML tools to support specific NOAA operational use cases. Table 2-1 and Table 2-2 describe the geophysical parameters and data sources (NOAA satellites, instruments, ground-based measurements, and models) for each of the Earth system components we ingested for our demonstration along with details on spatial and temporal resolution and coverage.

For the atmospheric component, we processed 3D temperature and moisture profiles from Integrated Global Radiosonde Archive (IGRA) ground-based radiosonde point measurements**.** We used the AI/ML algorithm multi-instrument inversion and data assimilation preprocessing system-AI (MIIDAPS-AI) to derive temperature and moisture profiles from advanced technology microwave sounder (ATMS) satellite observations and Global Forecast System (GFS) model output (Section 3.1.4.2). For the ocean, we processed sea surface temperature (SST) using Advanced Clear-Sky Processor for Ocean global SST from the Joint Polar Satellite System (JPSS) Visible Infrared Imaging Radiometer Suite (VIIRS) and Geostationary Operational Environmental Satellites (GOES) Advanced Baseline Imager (ABI) along with model output from GFS. SST directly interacts with sea ice concentration (SIC) which we processed using derived products from Advanced Microwave Scanning Radiometer (AMSR-2) and a blended high-resolution VIIRS/AMSR-2 product, in addition to model output from GFS. For land, we demonstrated a fused fire product from GOES ABI Fire/Hot Spot Characterization and VIIRS Active Fires M- and Iband measurements. In addition, we used the Faraday cup and magnetometer on the Deep Space Climate Observatory (DSCOVR) to visualize the magnetic shear across the 3D magnetopause surface, using solar wind bulk plasma and the solar wind magnetic field.

Table 2-1 EO-DT Input Variables and Sensors.

Earth System Component	Variable	Data Sources
Atmosphere	Temperature and Moisture Profiles	IGRA data, ATMS (JPSS), and GFS
Ocean	SST	ABI (GOES), VIIRS (JPSS), and GFS
Cryosphere	SIC	AMSR-2/VIIRS (JPSS) and GFS
Land and Hydrology	Fire Product	ABI (GOES), VIIRS (JPSS)
Space Weather	Solar Wind Bulk Plasma and Magnetic Field	Faraday cup and magnetometer (DSCOVR)

Table 2-2 EO-DT Input Data Spatial and Temporal Resolution and Coverage.

Geophysical Variable	Sensor	Spatial Resolution	Spatial Coverage	Temporal Resolution
SST	ABI	2 km	Full disk	1 hour
SST	VIIRS	375m	Global	10 mins
SST	GFS	13 km	Global	1 hour
Temperature and moisture	ATMS	16 km	Global	1 hour
Temperature and moisture	GFS	13 km	Global	1 hour
Temperature and moisture	IGRA	N/A	Global	6 hour
SIC	AMSR2	10km	Both poles	1 day

Geophysical Variable	Sensor	Spatial Resolution	Spatial Coverage	Temporal Resolution
SIC	Blended VIIRS/AMSR2	1 km	Both poles	1 day
SIC	GFS	13 km	Both poles	1 hour
Fire product	ABI	2 km	Full disk	1–10 mins
Fire product	VIIRS	375m	Global	10 mins
Solar wind magnetic shear	Faraday cup and magnetometer	N/A	N/A	4–6 minutes

2.1 EO-DT DATA INPUT SERVICE

For each sensor, we pull 2 weeks of historical data at various times using our static data input service. Any missing data or shortened timelines are outlined in Appendix Section 1 Table A-2. Table 2-3 outlines data for the EO-DT that is pulled from various sources and stored in an AWS S3 bucket for later processing. Our data input service sources data directly from various NOAA S3 buckets or through HTTP requests that locate and download data from a variety of NOAA or NASA websites. Once the data is saved off, it is organized into directories by sensor, date, and time to prepare it for ingestion into our backend applications. Upon completion, the number of saved files per sensor is compared against the expected number of files to highlight if any are missing (e.g., for 1 hour of SST, we expect 2 GFS, 2 GOES, and 6 VIIRS).

Data can also be pulled on a real-time basis using the data input service in ‘live’ mode. For this prototype, the live input service focused on pulling down SST data only. When running in live mode, the service periodically checks for SST data within the previous hour to be uploaded to its respective NOAA storage location and downloads them to our S3 bucket. Once all 10 SST files (6 VIIRS, 2 GOES, 2 GFS) have been downloaded, a configuration file is generated which kicks off lambda events and all other downstream processing (detailed in Section 3.1.2).

Table 2-3 EO-DT Data Sources for each Geophysical Variable Sensor.

Geophysical Variable	Sensor	Data Source URL
SST	ABI	NOAA GOES Open Data Registry
SST	VIIRS	NOAA CoastWatch NPP VIIRS L3U Sea Surface Temperature Data
SST	GFS	NOAA GFS Open Data Registry
Temperature and moisture	ATMS	NOAA JPSS Open Data Registry
Temperature and moisture	GFS	NOAA GFS Open Data Registry
SIC	AMSR2	STAR NESDIS AMSR2 Daily Sea Ice Concentration
SIC	Blended VIIRS/AMSR2	NOAA PolarWatch Blended Daily Sea Ice Concentration
SIC	GFS	NOAA GFS Open Data Registry
Fire product	ABI	NOAA GFS Open Data Registry
Fire product	VIIRS	NASA VIIRS Fire Data Product
Solar wind magnetic shear	Faraday cup and magnetometer	NOAA DSCOVR Space Weather Data Portal

3 DIGITAL TWIN FRAMEWORK

For our prototype, we used the foundation of OR3D to build a custom processing engine (hereafter referred to as the NOAA processing engine). After data was organized into our S3 bucket through our data input service, it was ingested into the NOAA processing engine and converted into a common spatiotemporal grid and required data processing and fusion algorithms were applied. Data was then aggregated and tiled into the Uber H3 hierarchy and formatted as a series of OpenUSD layers for interactive visualization and analysis. The following sections provide more detail on each of the processing steps.

3.1 NOAA PROCESSING ENGINE DATA PROCESSING PIPELINES

We built two different methods of processing data to encompass our study goal of operational and R&D use cases for NOAA. These include:

- The static processing method, which consists of downloading the necessary data to a local EC2 server or Linux machine and running the NOAA processing engine docker container with the proper command line arguments (input folder for the downloaded data, output folder, etc.) (Figure 3-1) (Recommendation 1.1).
- The live processing method, which uses data upload events to invoke automatic processing of the data that was uploaded (Figure 3-2) (Recommendation 1.2).

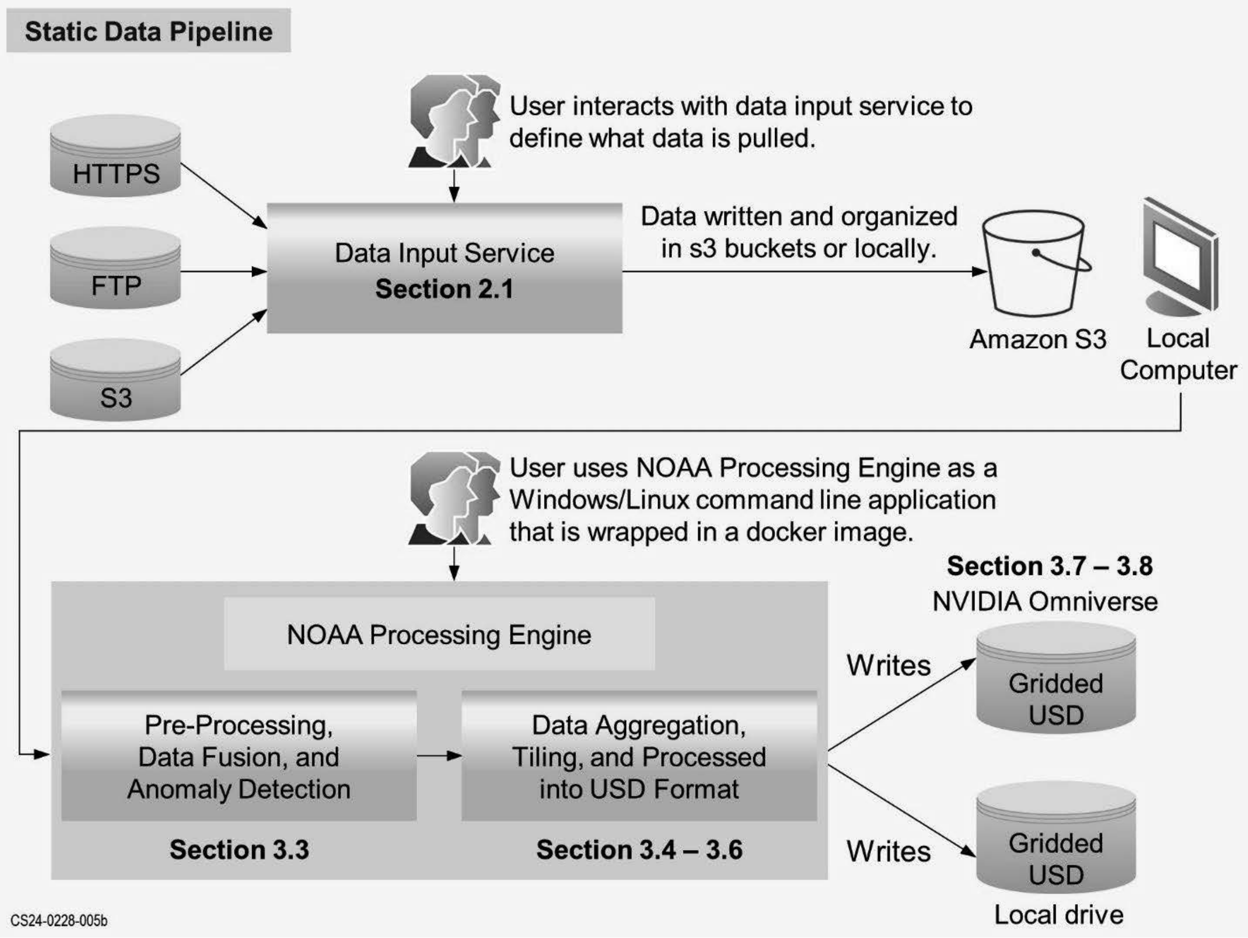

Static Data Pipeline
User interacts with data input service to define what data is pulled.
HTTPS
FTP
S3
Data Input Service
Section 2.1
Data written and organized in s3 buckets or locally.
Amazon S3
Local Computer
User uses NOAA Processing Engine as a Windows/Linux command line application that is wrapped in a docker image.
NOAA Processing Engine
Pre-Processing, Data Fusion, and Anomaly Detection
Section 3.3
Data Aggregation, Tiling, and Processed into USD Format
Section 3.4 – 3.6
Writes
Writes
Section 3.7 – 3.8
NVIDIA Omniverse
Gridded USD
Gridded USD
Local drive
CS24-0228-005b

Figure 3-1 Diagram Showing the Static Data Pipeline. *The data input service collects data, processes it through the NOAA processing engine, and writes it locally or to Omniverse Nucleus.*

We recommend using the static processing method for quick prototyping and R&D use cases and using the live processing method for processing data operationally using verified and highly tested pipelines (Recommendation 1.2).

3.1.1 Static Data Processing Pipeline

For quick prototyping and R&D use cases, we recommend our static data processing pipeline which allows the user to control what data is processed. This can be performed for analysis of a large historical dataset or for a single file. The static data pipeline provides the flexibility to process data coming from a local user directory or an AWS S3 bucket rather than being streamed in near real-time for operational purposes using the data input service (Figure 3-1) (Recommendation 1.2). Using the NOAA processing engine, a user can set up command line arguments including the path to the data to be processed, the output folder, and some optional flags depending on the data type being processed. The static data pipeline also allows for easier integration of containerized algorithms for testing (Section 3.1.3). A full example of the static data pipeline and a list of all command line options and their usages are documents in Appendix Section 5.1 and 5.2.

3.1.2 Live Data Processing Pipeline

To study how our digital twin prototype would be used operationally, we built a live data processing pipeline as part of our NOAA processing engine (Figure 3-2). First, a user interacts with a configuration file which is input into our data input service that determines what data a user wants processed and defines NOAA processing engine parameters like the resolution, tiling system, etc. In this implementation, data streams in from our data input service as if it was coming in on a near real-time basis from a sensor. For the current implementation of the live data processing pipeline, we focused on one example of processing hourly timesteps of SST data. Each timestep consists of six VIIRS, two GOES (East and West), and two GFS files; one for the previous and one for the current timestep.

When a file is uploaded from our data input service into an S3 bucket, our AWS Lambda function (detailed in Appendix Section 6), which is an event driven python script, determines if the file is a data file or a configuration file. If a data file is uploaded, its path is collected and stored in a DynamoDB database which acts as a temporary storage location while other files are collected The configuration file should only be uploaded when a full timestep of data is collected and signals that processing of the data should begin. Our data input service automatically generates and uploads the configuration file once all files from 1 hour of data have been uploaded.

Once the configuration file is uploaded, the AWS Lambda then acquires all the file paths stored in the DynamoDB, parses the configuration file for user-defined options, and creates a script that will be passed to an AWS EC2 server for it to perform. This script consists of downloading all the files locally to the EC2 server, pulling the most updated version of the NOAA processing engine container, and setting up the command line arguments with the options specified in the configuration file. After this is setup, the container runs on the EC2 which processes the data that was downloaded and outputs tiled USD files (Sections 3.3–3.6) to the Omniverse server (Sections 3.7–3.8). The EC2s are created and destroyed on demand to save on processing costs.

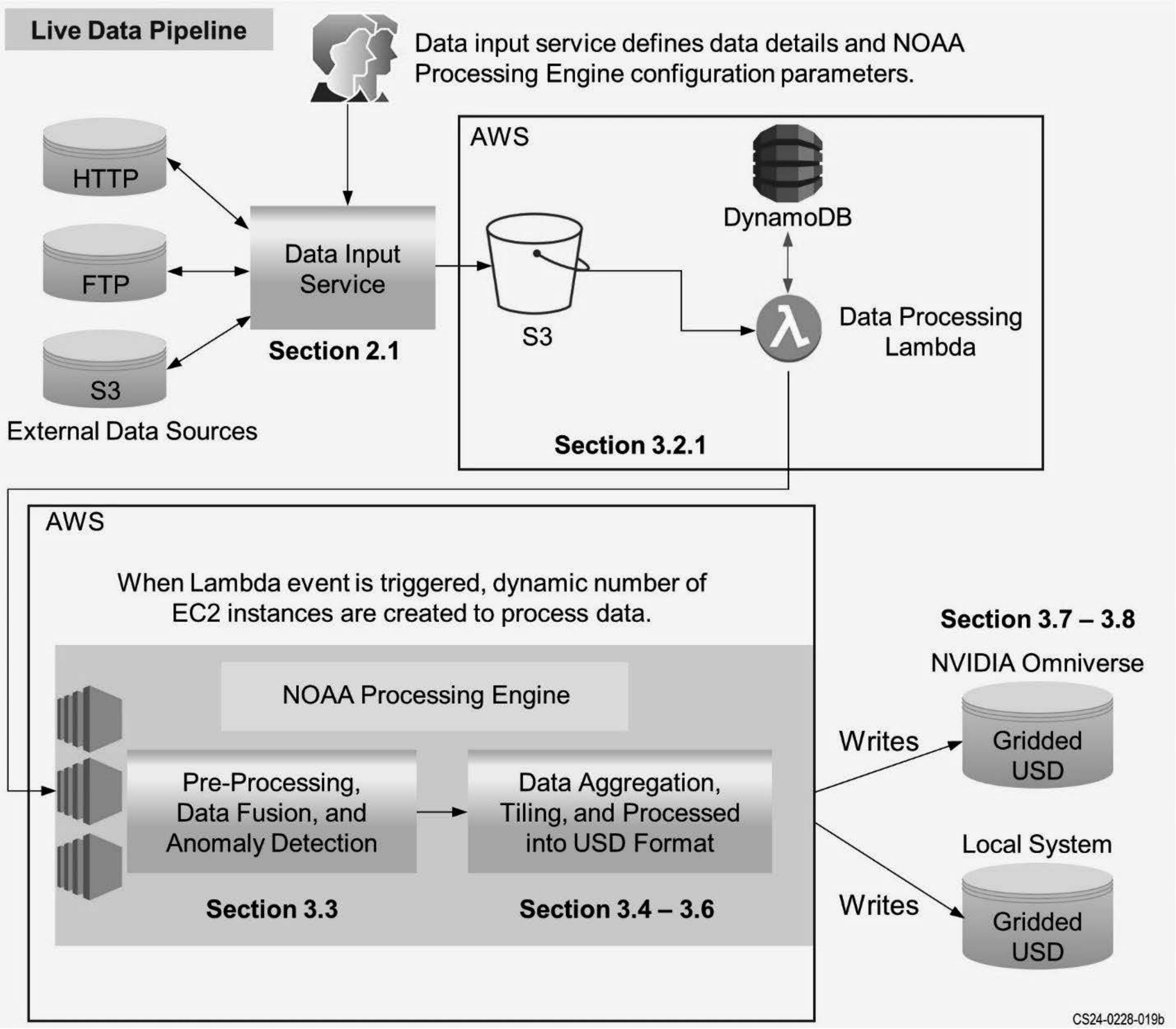

Live Data Pipeline
Data input service defines data details and NOAA Processing Engine configuration parameters.
HTTP
FTP
S3
External Data Sources
Data Input Service
Section 2.1
AWS
DynamoDB
S3
Data Processing Lambda
Section 3.2.1
AWS
When Lambda event is triggered, dynamic number of EC2 instances are created to process data.
NOAA Processing Engine
Pre-Processing, Data Fusion, and Anomaly Detection
Data Aggregation, Tiling, and Processed into USD Format
Section 3.3
Section 3.4 – 3.6
Section 3.7 – 3.8
NVIDIA Omniverse
Writes
Gridded USD
Local System
Writes
Gridded USD
CS24-0228-019b

Figure 3-2 Diagram Showing the Live Data Pipeline.
The data input service collects data, processes it through the NOAA processing engine using multiple AWS EC2s, and writes it locally or to Omniverse Nucleus.

3.1.3 NOAA Processing Engine Algorithm Applications

Depending on the use case, algorithms can be applied within or outside of the NOAA processing engine. If NOAA-provided data processing algorithms are provided as individual containers in the case of MIIDAPS-AI (Section 3.1.4.2) **for non-operational use**, **we recommend those containers are run as a preprocessing step outside of the NOAA processing engine (Recommendation 1.2)**. Inputs are ingested from an S3 bucket or local directory which already includes the necessary files from pulling from the data input service (Section 2.1). Once processed, outputs from the algorithm container are then ingested from an AWS S3 bucket or the local filesystem on which the NOAA processing engine is run (Figure 3-3). **If containerized algorithms are desired for operational use, we recommend they can be added directly into the NOAA processing engine pipeline (Recommendation 1.2)**. As an example of a potential operational process that needs to be completed on a near real-time basis, we built our data fusion and anomaly detection algorithms directly into the NOAA processing engine that are run as part of a mode passed into the application.

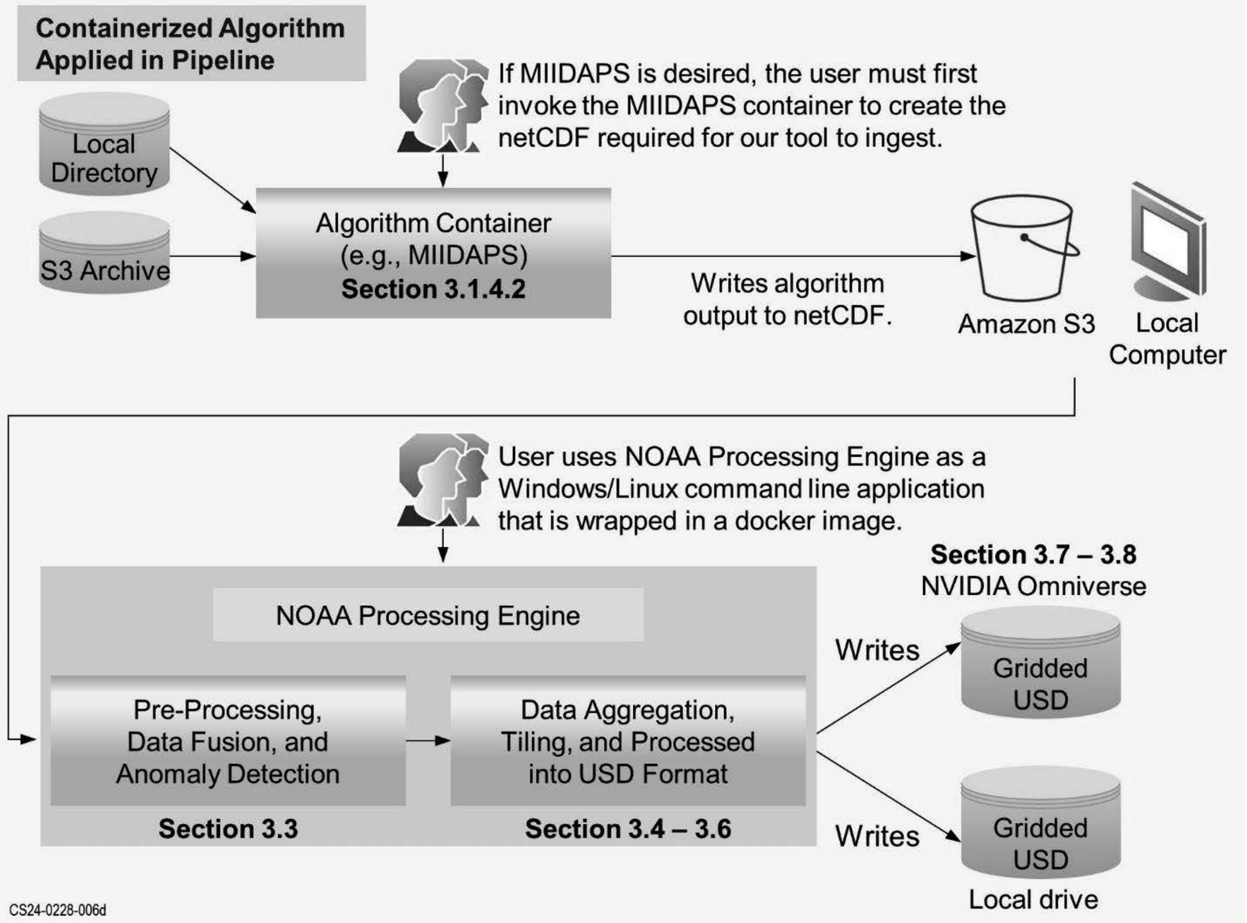

Containerized Algorithm Applied in Pipeline
If MIIDAPS is desired, the user must first invoke the MIIDAPS container to create the netCDF required for our tool to ingest.
Local Directory
S3 Archive
Algorithm Container (e.g., MIIDAPS) Section 3.1.4.2
Writes algorithm output to netCDF.
Amazon S3
Local Computer
User uses NOAA Processing Engine as a Windows/Linux command line application that is wrapped in a docker image.
Section 3.7 – 3.8 NVIDIA Omniverse
NOAA Processing Engine
Pre-Processing, Data Fusion, and Anomaly Detection
Section 3.3
Data Aggregation, Tiling, and Processed into USD Format
Section 3.4 – 3.6
Writes
Gridded USD
Writes
Gridded USD
Local drive
CS24-0228-006d

Figure 3-3 Diagram Showing the Containerized Algorithm Pipeline.
The data input service collects data, processes it through an algorithm prior to being passed to the NOAA processing engine, and writes it locally or to Omniverse Nucleus.

3.1.4 NOAA Processing Engine: Data Processing, Data Fusion, and Anomaly Detection

Once the data is ingested, regardless of the pipeline used, the data is preprocessed into a common spatiotemporal grid. Our processing linearly interpolates all 2D data products to a global 10 km Earth centered Earth fixed (ECEF) common grid, although this spatial resolution is configurable. Different data sources are also fused together at this resolution defined by the BAA as an ideal global resolution. In addition, our pipeline detects both sensor and physical anomalies in each Earth system domain dataset. Our system mainly uses quality flags provided in metadata to filter out sensor anomalies (e.g., non-clear-sky anomalies). To ensure the EO-DT prototype included technical design considerations required by a complete implementation, the NOAA processing engine pipeline records data source attributes and observational and analytical metadata in the final OpenUSD output. The metadata is used to optimize visualizations in our user interface and can be inspected or queried quantitatively by the user. The metadata recording mechanism permits future incorporation and linkage to more extensive scientific metadata and associated schemas, in-depth data provenance, and knowledge graph resource description framework (RDF) entities, relationships, and triplestores.

3.1.4.1 Sea Surface Temperature

To study how to best represent SST in our EO-DT prototype, we use global observations from geostationary and polar orbiting satellite and models including GOES East, GOES West, VIIRS, and GFS (Figure 3-4). To fuse this data, we integrated a data fusion algorithm directly into the NOAA processing engine which linearly interpolates data from each of the data sources into a single new fused layer (Figure 3-5). The fusion algorithm does not track which individual sensors are integrated into the fusion layer, rather it is defined by a list of files that is processed by the back end. To further demonstrate data fusion, we created a metric in the SST output metadata which provides a count of the number of sensors that see an observation of SST in the pixel (Figure 3-5). This gives users a confidence interval based on how many sensors verified the same measurement. The standard deviation of the fused measurement is also provided to give additional confidence in using a measurement since they can see the variance between sensor measurements. We do not validate thermodynamic consistency beyond these metrics but they should give the user enough context to determine if the fused data product is viable for their use case. To detect short-term anomalies, the fused layer is subtracted from the previous hour of the GFS data (Figure 3-5). This provides an hourly measurement of SST change, the use case of which depends on a scientist's interest. For example, a change of 1° K in an hour may be deemed normal while a change of >5° K would highlight an area requiring further investigation.

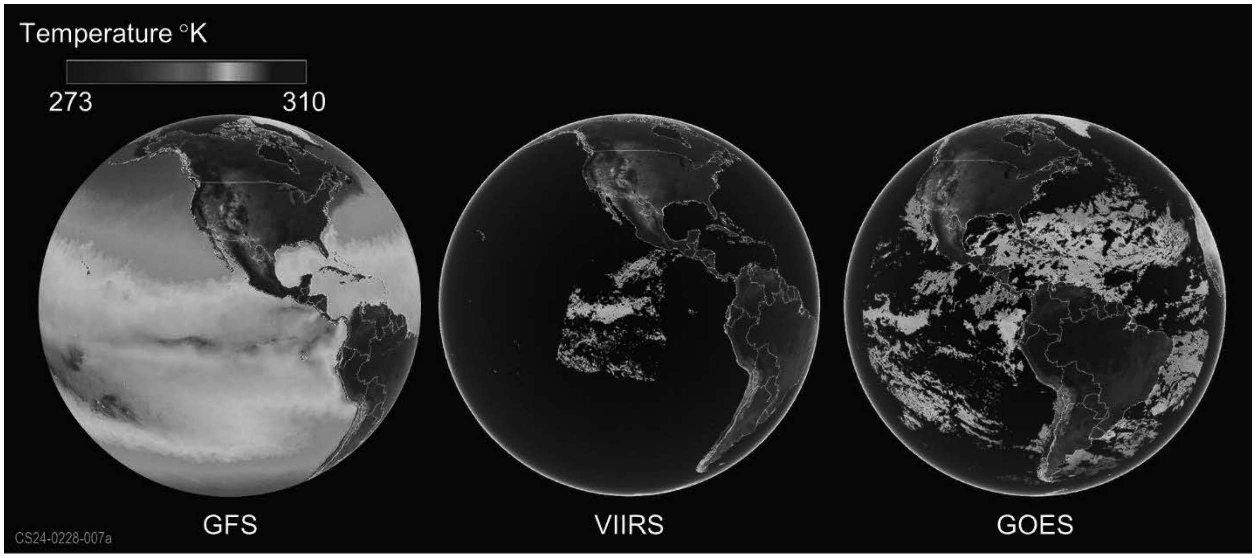


Figure 3-4 SST Measurements. *From 1. GFS, 2. VIIRS, and 3. GOES visualized in Agatha.*

We initially built and studied a convolutional autoencoder for anomaly detection. It was trained on GFS data as truth, which would take in SST data from sensors, reconstruct that data, and use absolute error between the reconstructed data and the input data, per pixel, to determine any spikes or wells in temperature (Figure 3-6). However, this solution proved to be overengineered for the problem (especially with globally available GFS data) and we reverted to a simpler methodology. In addition, anomalies in higher resolution data did not present well since we trained the autoencoder on relatively low resolution GFS data. We recommend using high-resolution data if a similar approach is pursued for a future digital twin to make sure that sensor anomalies can be detected. Although we did not implement this approach for SST anomaly detection, we did show that both AI-based and numerical methods can easily be applied in our plug-and-play framework.

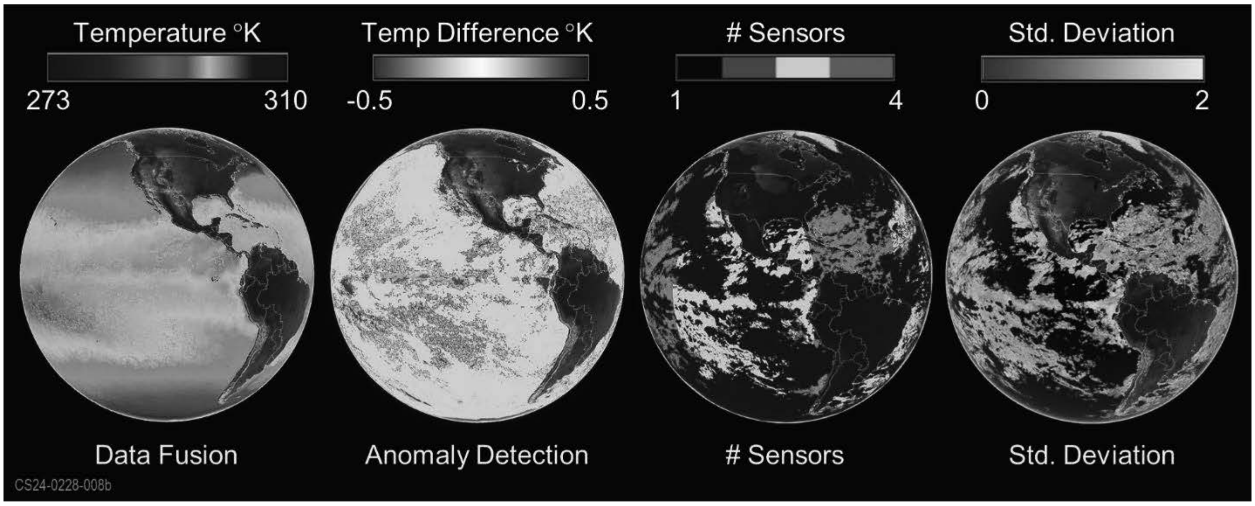


Figure 3-5 SST Data Fusion, Anomaly Detection, # of Sensors, and Standard Deviation.

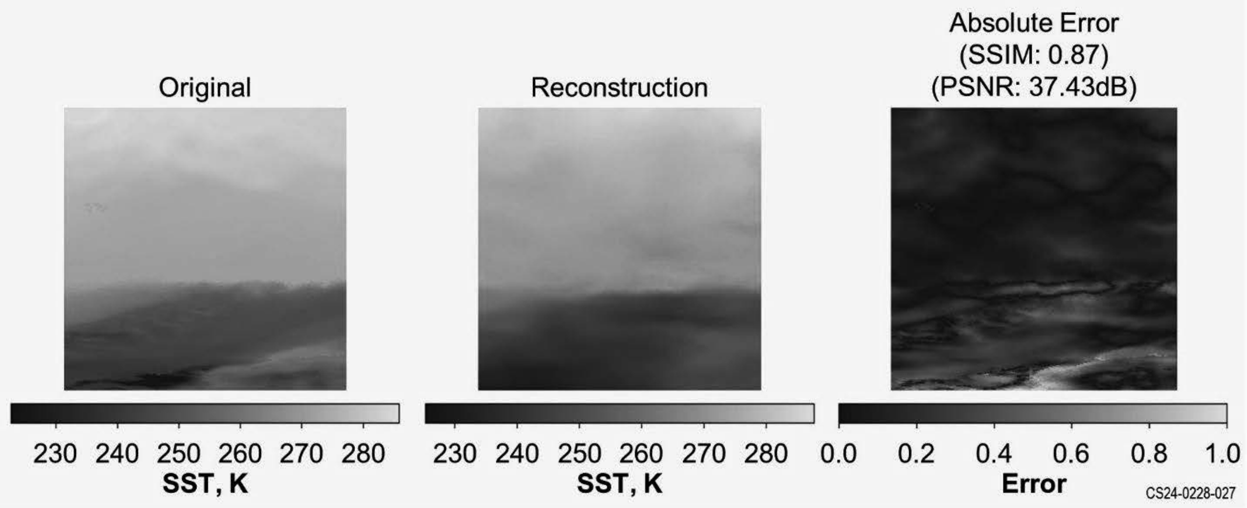


Figure 3-6 Preliminary Results from a Convolutional Autoencoder Reconstruction and Absolute Error for Sea Surface Temperature.

3.1.4.2 3D Temperature and Moisture Profiles

To investigate 3D temperature and moisture profiles in out EO-DT prototype, we explored ground- and satellite-based observations along with GFS output. Our ground-based observations come from the IGRA which consists of radiosonde and pilot balloon observations from more than 2800 globally distributed stations with data including pressure, temperature, geopotential height, relative humidity, dew point depression, wind direction and speed, and elapsed time since launch (Durre et al., 2018). These observations were preprocessed by NOAA to 20 evenly spaced vertical pressure levels (1000 mb to 50 mb at 50 mb spacing) and provided in NetCDF format for temperature and moisture profiles (Figure 3-7).

In addition to ground-based observations, we ingested and processed observations from ATMS combined with GFS model output in a NOAA developed AI/ML algorithm, MIIDAPS-AI, which infers atmospheric parameters including temperature and moisture profiles, and is orders of magnitude faster than traditional remote sensing algorithms while using far fewer resources. MIIDAPS-AI has been successfully applied to infrared, microwave, polar and geo sounders and imagers and can also be used for satellite observation preprocessing for data assimilation, data fusion, and more (Maddy & Boukabara, 2021). For our demonstration, we containerized this algorithm (Appendix Section 4) and implemented it into our pipeline (Section 3.1.4, Figure 3-3). We processed and display MIIDAPS-AI output at four meteorologically relevant pressure levels (1000 mb, 750 mb, 500 mb, 250 mb) on a 2D surface (Figure 3-7). For direct comparison, we also integrated GFS model output of temperature and specific humidity at the same four pressure levels at an hourly interval (Figure 3-8 and Figure 3-9).

We did not implement data fusion for 3D temperature and moisture profiles as GFS is an input to MIIDAPS-AI and IGRA data was irregularly available at different times and sites. However, we highly recommend as part of a future study to determine how to best fuse gridded and point data into a product on non-regular spatial and temporal scales.

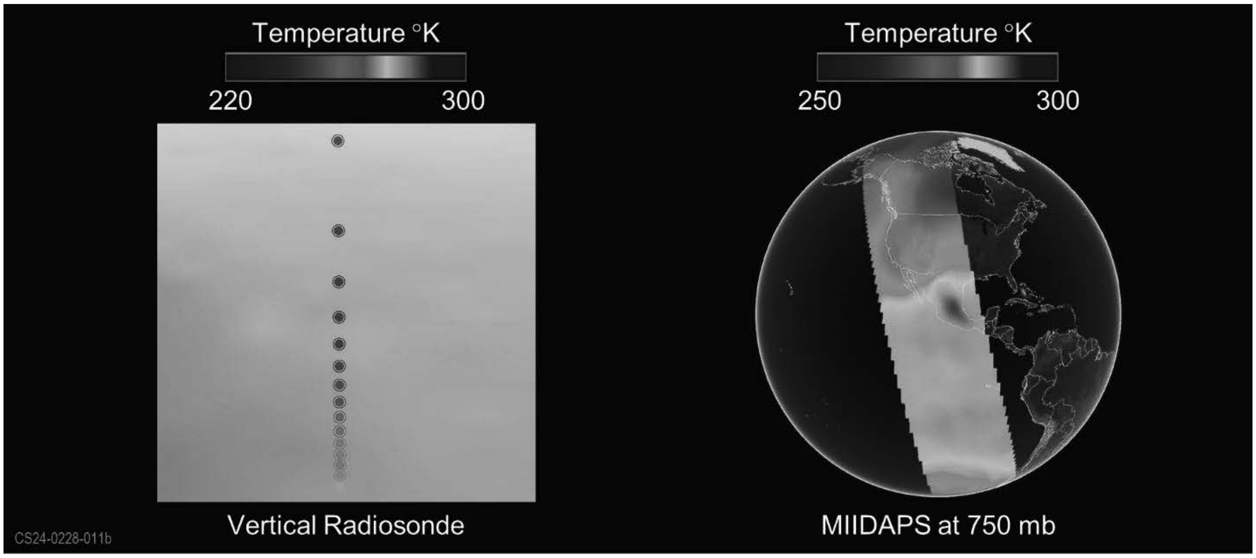


Figure 3-7 IGRA data and MIIDAPS output. *1. 3D temperature profile from 20 vertical levels of IGRA data and 2. MIIDAPS output visualized in Agatha.*

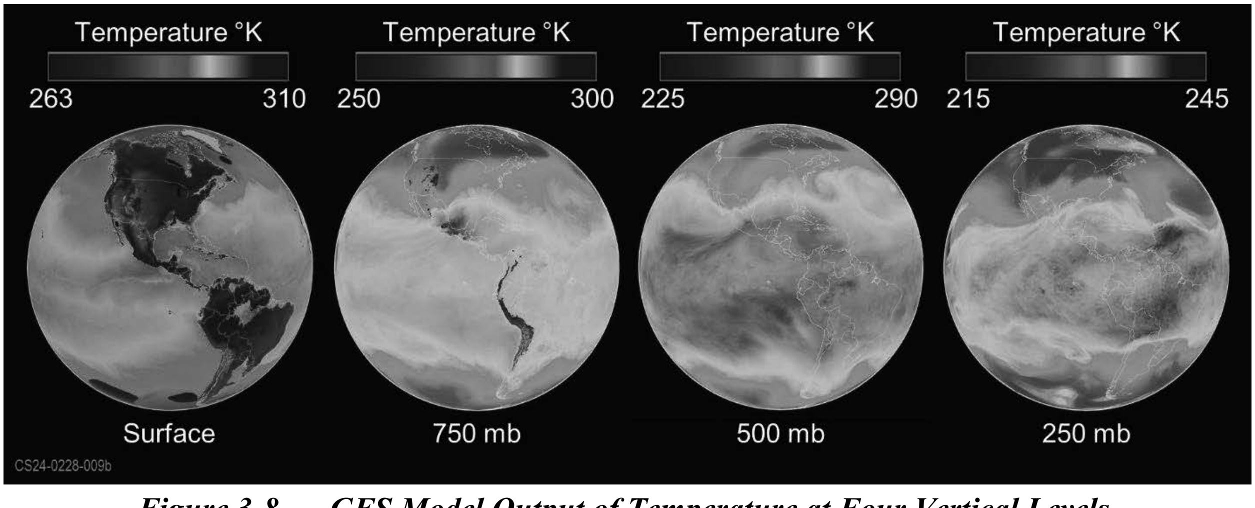


Figure 3-8 GFS Model Output of Temperature at Four Vertical Levels. *1. 1000 mb, 2. 750 mb, 3. 500 mb, 4. 250 mb visualized in Agatha.*

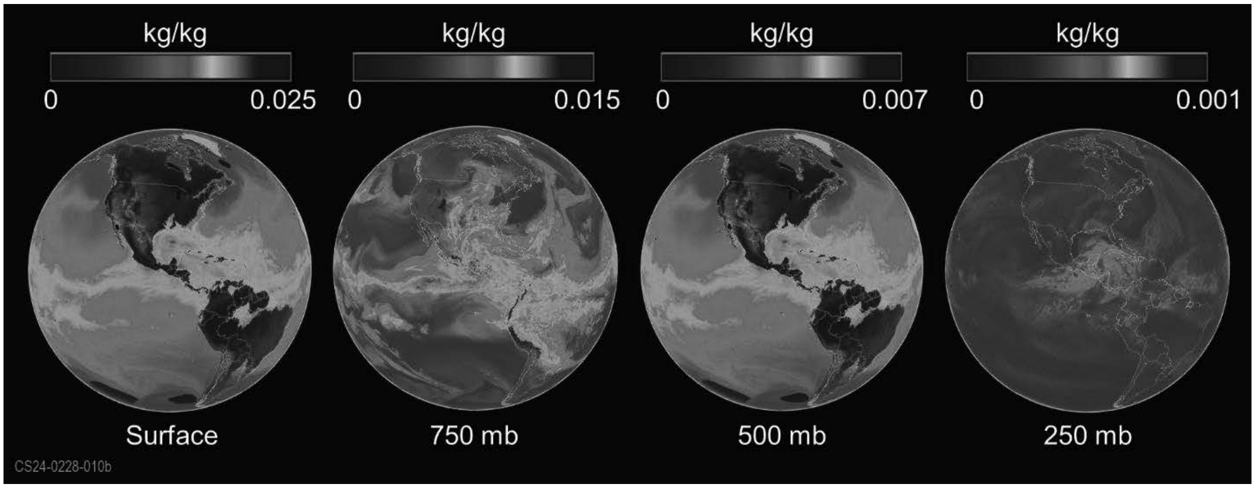


Figure 3-9 GFS Model Output of Humidity at Four Vertical Levels.
1. 1000 mb, 2. 750 mb 3. 500 mb, 4. 250 mb visualized in Agatha.

3.1.4.3 Sea Ice Concentration

Our EO-DT prototype demonstrates SIC at both the north pole (NP) and south pole (SP) using daily global GFS model output, AMSR-2 satellite observations, and a high-resolution blended AMSR2/VIIRS product (Figure 3-10). GFS data is available hourly, though changes in SIC are largely relevant daily, thus the GFS model output layer is representative of the 0th and 18th hours averaged together. Our NOAA processing pipeline fused data using the blended AMSR2/VIIRS product and GFS at both poles. Similar to SST, we created a metric in the SIC output metadata which provides a count of the number of sensors that see an observation of SIC in the pixel and a standard deviation between the fused data source inputs (Figure 3-11). Our approach for anomaly detection included taking the absolute difference between our fused daily SIC product and the median extent from 1981–2010 for that day of the year (Figure 3-11), following similar methods as are employed for the products displayed on the National Snow and Ice Data Center’s website for SIC. The median extent product was only available in the Arctic so our anomaly product is limited to the NP.

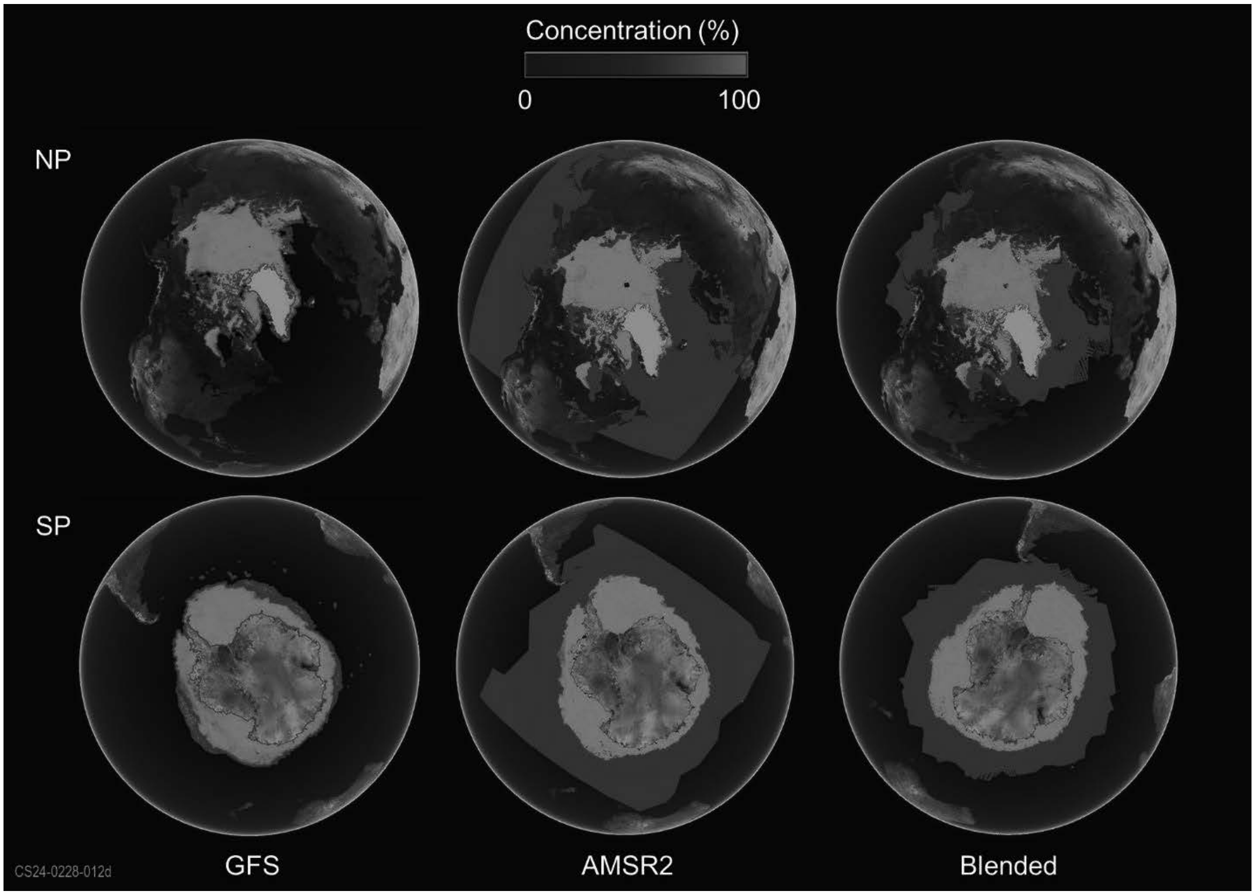


Figure 3-10 Sea Ice Concentration Data.
GFS (SP), GFS (NP), AMSR2 (NP), AMSR-2 (SP), Blended AMSR2/VIIRS (NP).

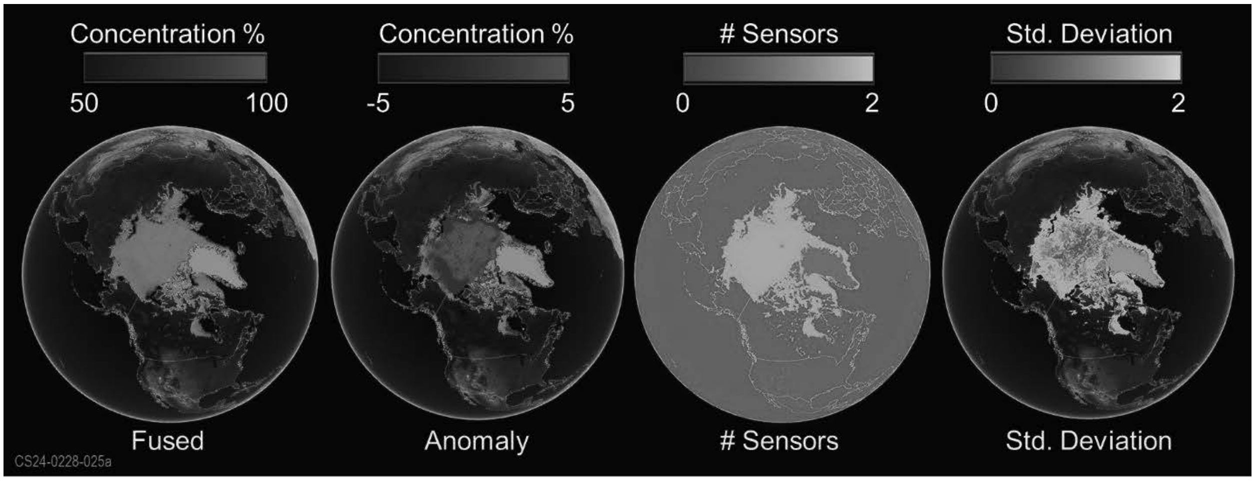


Figure 3-11 Fused Sea Ice Concentration from GFS and the blended VIIRS/AMSR2 product, Anomalous SIC, Number of Sensors, and standard deviation of SIC.

3.1.4.4 Fire Product

In our EO-DT prototype, we studied fire products using GOES ABI and VIIRS (Figure 3-12). We mapped values to confidence levels of fire which are outlined in Appendix Section 3.2 in Tables A-4 and A-5. These sensors represent geostationary and polar orbiting fire products to capture the best possible spatial and temporal resolution. Our NOAA processing engine filtered based on data quality flags to remove sensor anomalies based on the instrumentation for GOES and VIIRS. Our prototype does not include fire anomaly detections outside metadata flags as that is an ongoing area of research. To provide a relative simple solution for anomalies, we recommend to filter fires on known areas of biomass burning, flares, etc. through a trusted global database.

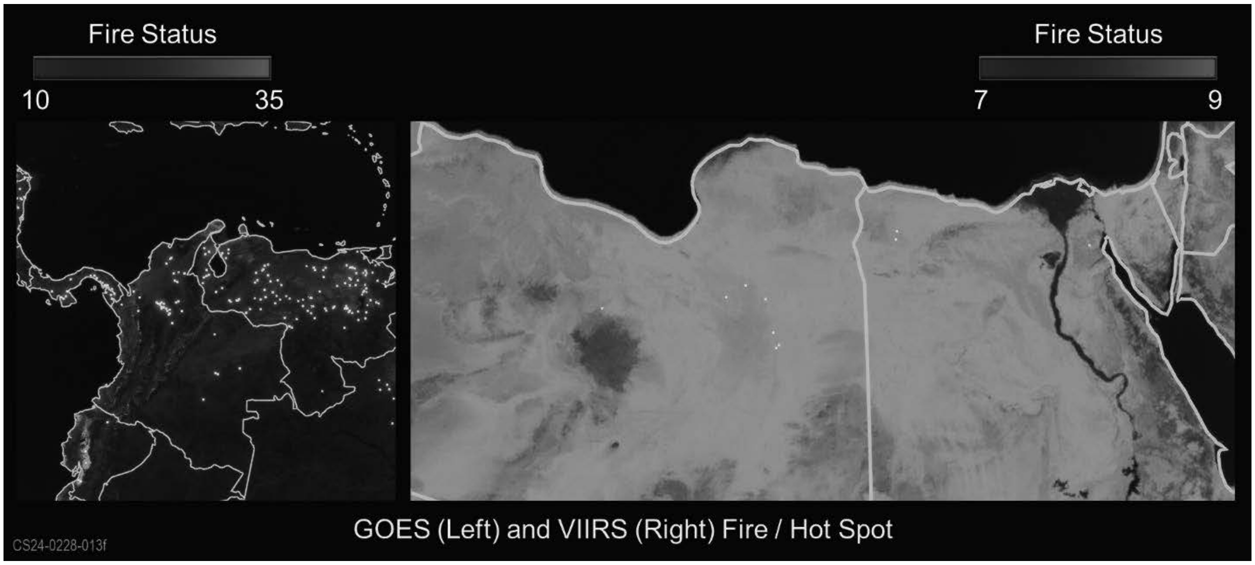


Figure 3-12 GOES (left) and VIIRS (right) Fire Data.

3.1.4.5 Space Weather

To demonstrate space weather in our EO-DT, we used the Faraday cup and magnetometer on the Deep Space Climate Observatory (DSCOVR) to derive the magnetic shear across the 3D magnetopause surface, using solar wind bulk plasma and the solar wind magnetic field (Trattner et al., 2021) (Figure 3-13). This is useful for predicting where magnetic reconnection is likely to occur, which transports solar wind plasma and energy into the magnetosphere, conditioning the magnetotail for the occurrence of geomagnetic substorms. Similar to MIIDAPS-AI we preprocessed this data prior to ingesting into the NOAA processing engine, though instead of a container we used a set of interactive data language scripts (detailed in Appendix Section 3.1). We output this data in a similar ECEF grid to the other geophysical variables but through feedback in the program we would recommend looking into Geocentric Solar Magnetospheric (GSM) coordinates which would be more appropriate in the future. **We found integrating space weather into the same digital twin framework that dealt with surface or tropospheric Earth system domains difficult due to the magnitude of scale difference and juxtaposing ideal reference coordinate systems**. **From this we learned that a unified single digital twin may not be the best way to represent all Earth system domains especially for space weather which may require its own individual digital twin framework** (**Recommendation 2.4**). We did not perform data fusion or anomaly detection for space weather for our EO-DT prototype.

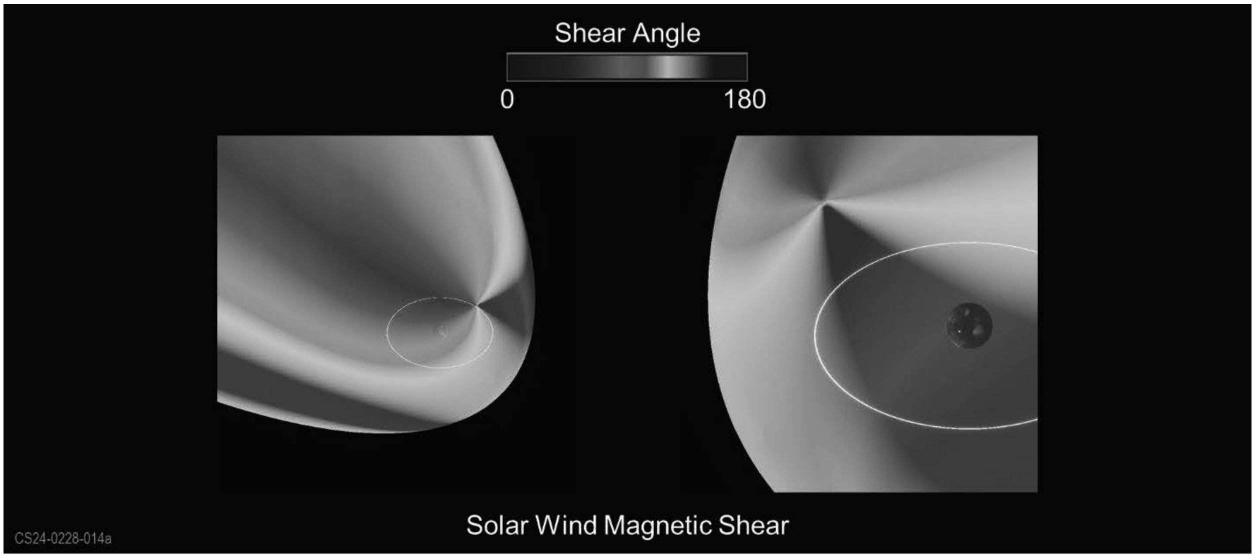


Figure 3-13 Solar Wind Magnetic Shear Derived from DSCOVR Observations.

3.2 NOAA PROCESSING ENGINE: TILING INTO UBER H3

Once data is processed and fused and the NOAA processing engine detects anomalies, it is tiled into the Uber H3 tiling scheme which provides uniform, global data resolution, ensuring precision in spatial analysis (Figure 3-14). We chose this tiling scheme since it is better represented at the poles than other tiling systems. Uber H3's hierarchical structure supports multi-resolution analysis for both detailed and broad perspectives, streamlining geospatial data processes for efficiency and accuracy. Each tile has a persistent, georeferenced index. This unique identifier allows for swift and precise reconstruction of cells based solely on this value. Its persistent nature means it is accessible to both the backend and frontend, enabling quick, collaborative reconstructions of the tiles. The benefit of the NOAA processing engine's plug-in based architecture is that it can facilitate flexible handling and interoperability with multiple tiling systems, as might be needed to incorporate data from other agencies (e.g., NASA). To prove this out in out prototype, in addition to Uber H3, we also tested and verified that Google S2 worked as an alternative and could integrate others that conform to the globally consistent and hierarchal structure.

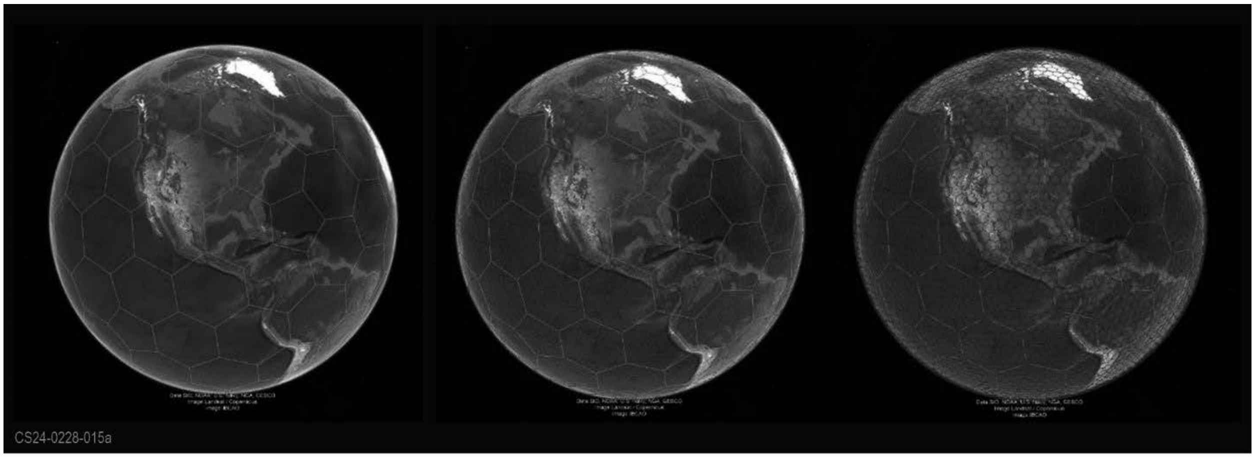


Figure 3-14 ***Uber H3 Tiling System.*** *Image Shows Multiple Resolutions of the Hexagonal Tile System with more Granular Resolutions from left to right.*

3.3 NOAA PROCESSING ENGINE: FORMATTING INTO OPENUSD

A key challenge that our EO-DT must address is the aggregation and composition of multiple observational data layers that together create a holistic digital replica of the state of Earth. Our EO-DT uses OpenUSD (Universal Scene Description framework) to compose and store 3-D geospatial information and support arbitrary domain-specific metadata, extensibility, and interoperability among diverse simulation and collaboration workflows and applications. OpenUSD allows EO-DT to represent the entire Earth using a hierarchy of USD files, each containing groups of aggregated tiles (e.g., using the Uber H3 tiling system at particular resolution levels), permitting observational data to be efficiently fetched on demand and cached according to the needs of the Agatha viewer or other client applications. The EO-DT observational data is represented geometrically using USD points and triangle mesh primitives, with measured quantities stored as attributes on the geometric primitives themselves or encoded into quantized texture maps stored in PNG files referenced by the USD files. Since the key EO-DT observations are encoded in standard USD geometric primitives and texture maps, a broad range of 3-D viewers, editors, design and simulation tools can directly load and operate on the data. To enable general tools to correctly display EO-DT observation layers, the NOAA processing engine assigns appropriate default color maps to the quantized PNG texture maps so that no domain-specific knowledge of the data is required for correct display in a 3-D viewer. OpenUSD supports XML-like extensibility by allowing user-defined attributes to be assigned to data objects, and EO-DT uses these mechanisms to encode metadata about observations so that it is embedded and thereby inseparable, but does not necessarily have to be processed or interpreted by software tools or workflows that do not require it. OpenUSD supports the definition of formal schemas for attributes and metadata extensions to help ensure interoperability and correctness.

3.4 NOAA PROCESSING ENGINE: AGGREGATION SCHEME

Next, the NOAA processing engine pipeline post-processes the tile hierarchy into aggregate groups of tiles together. This helps to achieve balance between granularity of file-level access, caching, transmission, and undesirable overheads that can arise from excessively-fine-grained approaches. By aggregating groupings of spatially neighboring tiles together into the same OpenUSD file, the NOAA processing engine cloud service I/O performance is improved, performance of staging and updates to observation data into the Omniverse Nucleus service are improved, and client-side OpenUSD parsing overheads are collectively reduced, while still avoiding excess data transmission during on demand streaming to remote clients. The tiles are represented in OpenUSD files as ‘prims’ which retain their attributes and metadata and can still be directly individually referred to and accessed by their own Uber H3 indices as they were prior to aggregation into OpenUSD files.

3.5 NOAA PROCESSING ENGINE FILE TRANSFER TO OMNIVERSE NUCLEUS

At the point when the NOAA processing engine has generated OpenUSD files containing aggregated tiles of EO-DT observational data, the set of USD and PNG files covering the observational data layer for a given time sample must be transferred to the Omniverse Nucleus service instance that serves all clients. To perform this bulk data transfer efficiently, the NOAA processing engine makes use of the Omniverse client library and its associated APIs. The combination of the NOAA processing engine USD output and the Omniverse client library are collectively referred to as the EO-DT Omniverse ‘connector’ between the NOAA processing engine and the Omniverse Nucleus service. To permit out-of-band debugging, testing, and maintenance, the Omniverse Connect software development kit includes a standalone command line ‘omni client’ tool that performs the same bulk transfer operations that the NOAA processing engine implements internally.

3.6 OMNIVERSE NUCLEUS

A key element of constructing digital twins is the requirement for a data store that:

- Maintains the state of the twin
- Ensures data integrity, user authentication, and security
- Enables multi-user and multi-site collaboration
- Acts as an arbiter of competing operations on the twin data

The Omniverse Nucleus service is a high-performance object store that serves in this role for EO-DT. EO-DT’s Nucleus service instance uses a cloud-based deployment that permits a natural and high-performance coupling with the NOAA processing engine pipeline. This brings in new observational data and remotely located Agatha users on the Internet at large. To permit scalability to large numbers of remote users and help overcome sources of communication latency such as firewalls, VPNs, and long-haul networks, Omniverse provides a Nucleus Cache service. Nucleus caches can be placed in performance-advantageous locations in a local or regional network and be chained hierarchically, thereby greatly reducing the amount of read-mostly EO-DT data sent to remote users, particularly at shared offices or other sites supporting multiple client users. The reduction of EO-DT data egress from AWS EC2 to remote users reduces operational costs accordingly. Nucleus supports versioning of files, concurrent editing of versions of the same files, publish-subscribe update notifications, and live-update sessions for clients. The versioning and collaboration components of the Omniverse Nucleus service design make it particularly well suited as a hub for connecting the constituent services that make up a digital twin to the clients that interact with it. While Nucleus is optimized for storage and transmission of OpenUSD, raster images (PNG, GeoTIFF, and similar), and volumetric grids (VDB and similar), it supports any file or data format, including NetCDF or HDF5 as examples relevant for EO-DT. Refer to Appendix Section 7 for Omniverse Nucleus setup instructions.

3.7 NOAA PROCESSING PIPELINE METRICS

As part of our study, we provide metrics on the amount of data and resources available to process using the full static NOAA processing engine pipeline (Section 3.1.1). The full pipeline for one time step consists of downloading files from S3 using the data input service, processing raw data, applying data fusion and anomaly detection algorithms, tiling into the UberH3 format, aggregating USD files, and writing final output files to Omniverse Nucleus for each data example from each Earth system domain. **For the entire two week window across all domains we process over 7 TB of raw data down to ~130 GB in USD file format.**

Table 3-1 SST.

Sensor	Files per Hour	Data Size Ingested	Time to Process (Workstation)	Time to Process (AWS EC2 Instance)
GOES East and West	2	36M	00:01:28	00:00:33
VIIRS	6	3.1M	00:00:09	00:00:06
GFS	1	331M	00:00:16	00:00:14
One Time Step	10	1.4G	00:04:09	00:02:15

Table 3-2 Fire.

Sensor	Files per Hour	Data Size Ingested	Time to Process (Workstation)	Time to Process (AWS EC2 Instance)
GOES East and West	4	1.7 M	00:01:37	00:00:22
VIIRS	6	1.2 M	00:00:11	00:00:06
One Time Step	10	14 M	00:07:34	00:02:04

Table 3-3 SIC.

Sensor	Files per Day	Data Size Ingested	Time to Process (Workstation)	Time to Process (AWS EC2 Instance)
AMSR-2	1	27 M	00:00:06	00:00:04
Blended VIIRS/AMSR-2	2	318M	00:05:18	00:01:18
GFS	1	1 GB	00:00:25	00:00:11
One Time Step	4	1.4 GB	00:05:55	00:01:37

Table 3-4 ATMS.

Sensor	Files per Hour	Data Size Ingested	Time to Process (Workstation)	Time to Process (AWS EC2 Instance)
IGRA	N/A	37 KB	00:00:01	00:00:01
MIIDAPS	1	27 M	00:00:35	00:00:11
GFS	1	13 GB	00:03:15	00:02:09
One Time Step	4	13.28 GB	00:04:12	00:02:50

Table 3-5 Space Weather.

Sensor	Files per Hour	Data Size Ingested	Time to Process (Workstation)	Time to Process (AWS EC2 Instance)
DSCOVR	10	2.7 KB	00:00:01	00:00:01
One Time Step	10	2.7 KB	00:00:01	00:00:01

3.8 OMNIVERSE NUCLEUS TO AGATHA FILE TRANSFER

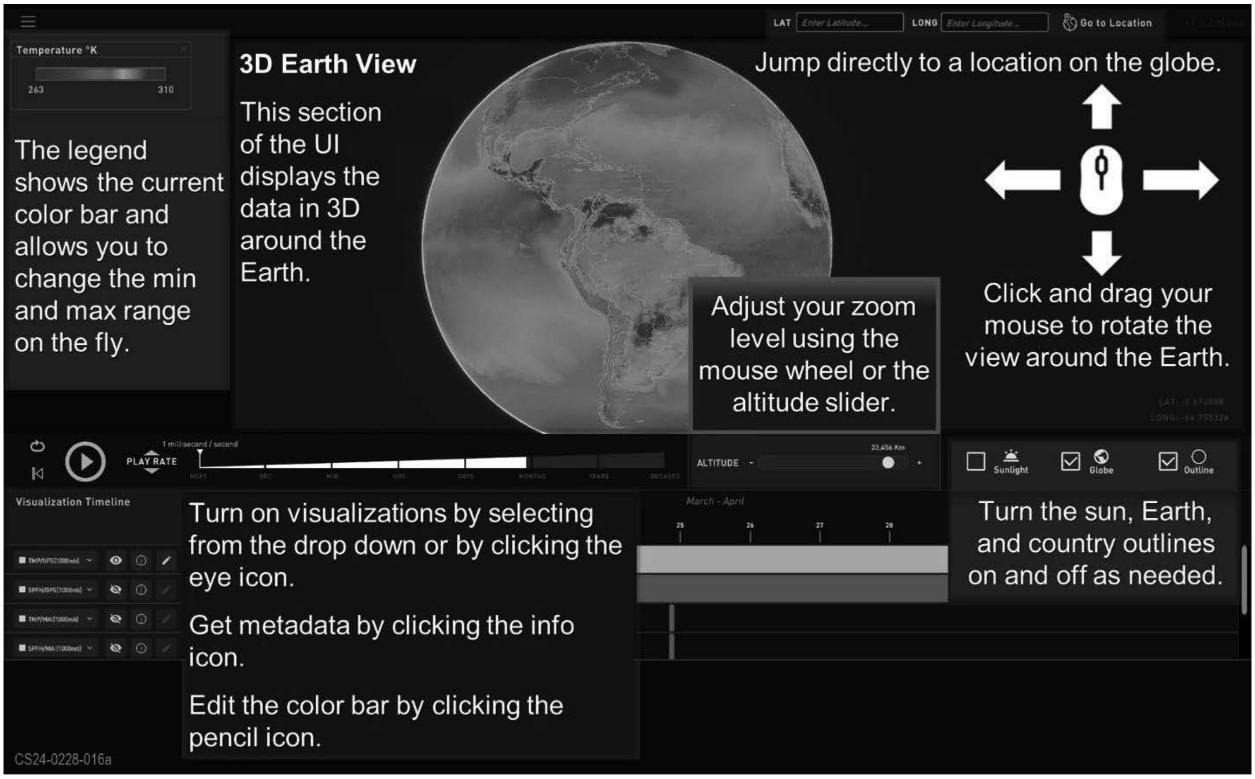
After final OpenUSD and NetCDF files are output to the Omniverse Nucleus service, our frontend, Agatha, must retrieve them. Agatha provides a detailed display of user-selected EO-DT observational data layers and metadata. To achieve and maintain its high-interactivity, Agatha gathers EO-DT observational data on demand in a streaming manner from the connected Omniverse Nucleus service. Agatha uses the view frustum, incident angle of tiles visible on the Earth's surface, and user preferences to determine the specific EO-DT tiles that must be fetched and their required resolution level. By virtue of the Uber H3 tile indexing mechanism, Agatha can fetch only those OpenUSD and PNG files containing the relevant tiles at the required resolution. Like the NOAA processing engine, Agatha directly incorporates the Omniverse client library and therefore can directly transfer files from Omniverse Nucleus with high-performance. The combination of Agatha's incorporation of OpenUSD and the Omniverse client library are collectively referred to as Agatha's Omniverse 'connector,' a mirror image of what is implemented in the NOAA processing engine.

3.9 AGATHA VISUALIZATION INTERFACE AND FEATURES

Agatha is the visualization frontend of the EO-DT which provides users with an intuitive and high-resolution interface to interact with their data. This component is a Windows-based application that runs natively on a laptop or desktop and requires a graphics processing unit (GPU) to run. For our study we used both standard Agatha features along with custom features built specifically for this program outlined in Table 3-6 and shown in Figure 3-15 (Recommendation 3.2) and Figure 3-16 (Recommendation 3.2). One example of a custom feature we implemented was NOAA themed colormaps (e.g., topo, grid3d, icing, HRRR Reflectivity, etc.) which can be dynamically changed when running the application. Many features of Agatha are easily editable by the configuration file and parameters within, described in Appendix Section 8. **To build a digital twin, we recommend using a flexible visualization tool where features can easily be added and customized for various user skill levels and use cases (Recommendation 3.2).**

Table 3-6 Standard and Custom Features Implemented in Agatha.

Feature	Type	Description
Globe interface	Standard	Can be panned around using the cursor.
Zoom	Standard	Move closer and farther away from the Earth’s surface.
Timeline interface	Standard	Allows user to step through individual time steps of each data type and a start, pause, and advance feature which allows the data to play on a loop.
Sun, globe, and country outlines	Standard	Toggle on and off the sun reflection, background globe imagery, and country outlines on demand.
Layers toggle	Standard	Toggle on and off different layers of data.
Location search	Custom	Automatically place the center screen point to a certain location on Earth.
Dynamic color bar values	Custom	Change the minimum and maximum color bar values on the fly to view different ranges.
Dynamic color bar gradient	Custom	Change the color map of the color bar using preset ranges and insert custom color maps through the configuration file.
Metadata viewer	Custom	View metadata on files including time ranges, value ranges, source filenames, time intervals, domains, etc.
Sample values	Custom	Clicking on the globe will let user sample point values including their latitude, longitude, value, and time stamp.



3D Earth View

This section of the UI displays the data in 3D around the Earth.

The legend shows the current color bar and allows you to change the min and max range on the fly.

Jump directly to a location on the globe.

Click and drag your mouse to rotate the view around the Earth.

Adjust your zoom level using the mouse wheel or the altitude slider.

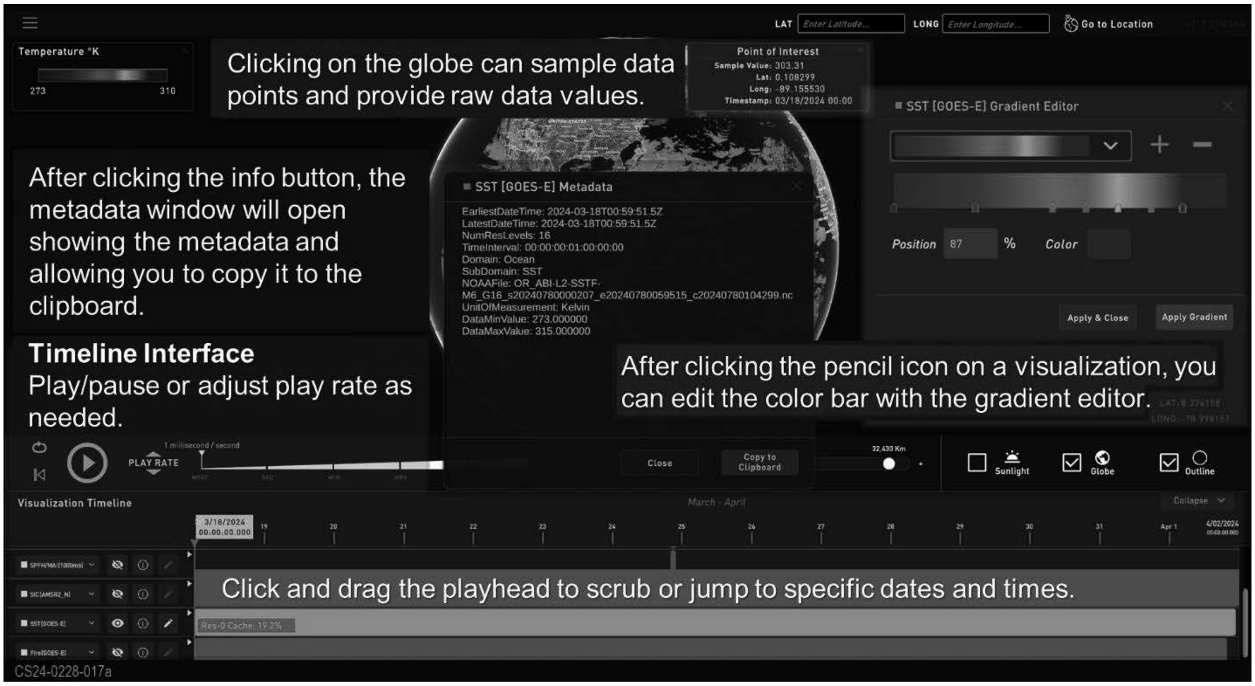
Turn on visualizations by selecting from the drop down or by clicking the eye icon.

Get metadata by clicking the info icon.

Edit the color bar by clicking the pencil icon.

Turn the sun, Earth, and country outlines on and off as needed.

Figure 3-15 Highlighting Agatha's Features of the Global Interface. *Image includes zoom, sun, globe, and country outlines, dynamic color bar values from the legend, location search, and toggling on/off layers.*



Clicking on the globe can sample data points and provide raw data values.

Point of Interest
Sample Value: 303.31
Lat: 0.108299
Long: -89.155530
Timestamp: 03/18/2024 00:00

After clicking the info button, the metadata window will open showing the metadata and allowing you to copy it to the clipboard.

SST [GOES-E] Metadata

```
EarliestDateTime: 2024-03-18T00:59:51.5Z
LatestDateTime: 2024-03-18T00:59:51.5Z
NumResLevels: 16
TimeInterval: 00:00:00:01:00:00:00
Domain: Ocean
SubDomain: SST
NOAAFile: OR_ABI-L2-SSTF-
M6_G16_s20240780000207_e20240780059515_c20240780104299.nc
UnitOfMeasurement: Kelvin
DataMinValue: 273.000000
DataMaxValue: 315.000000
```

Timeline Interface
Play/pause or adjust play rate as needed.

After clicking the pencil icon on a visualization, you can edit the color bar with the gradient editor.

SST [GOES-E] Gradient Editor
Position 87 % Color

Click and drag the playhead to scrub or jump to specific dates and times.

Figure 3-16 Highlighting Agatha's Features. *Image includes sample values, timeline interface, metadata viewer, and dynamic color bar gradient.*

4 PROTOTYPE AND ESTIMATED OPERATIONAL COSTS

During prototype development, cloud resource use was sporadic, making it impossible to precisely extrapolate operational costs from the development phase. To estimate a reasonable cost for an operational prototype, we used the AWS Pricing Calculator.

For estimating prototype operational costs, we gathered input parameters for two operating modes:

1. Continuous processing of hourly SST data
2. Processing a 2-week window of data from all domains for this prototype

Both operating modes assume a simple implementation of the prototype—particularly regarding Amazon EC2, where OR3D and Omniverse Nucleus each run on a single EC2. More advanced deployments could be used operationally. For example, Nucleus functions could be split across multiple EC2s. Doing so allows for scaling Nucleus functionality (e.g., storage/processing and data egress) independently. OR3D could also use Amazon Fargate to optimize processing loads.

Configuring the AWS Pricing Calculator allows the user to control cost-driving variables for each service. Table 4-1 shows examples of services and parameters of interest for this prototype.

Table 4-1 AWS Configuration Parameters of Interest.

AWS Pricing Calculator Service	Configuration Parameters of Interest
Amazon Simple Storage Service (Amazon S3)	• Storage size • Data transfer frequency, volume, and type
Amazon Elastic Compute Cloud (Amazon EC2)	• Workload frequency • Central processing unit • Memory • Network bandwidth • Utilization • Data transfer requirements
Amazon Lambda	• Request frequency • Memory • Ephemeral storage
Amazon DynamoDB	• Storage size • Average item size • Write type and rate • Peak activity duration

Table 4-2 displays the estimated costs for continuously processing SST data. We eliminated a $2 per hour fee for each region with dedicated EC2s from the calculations, assuming NOAA would already have EC2s operating in the region. Use of the AWS Lambda service fell within the free tier threshold.

Table 4-2 SST Hourly Live Processing Cost.

Prototype Component	Service	Upfront	Monthly	Annual	Configuration Summary
Data transfer from data source	S3 Standard	$0	$24	$284	• S3 standard storage (1 TB per month) • S3 standard average object size (50 MB) • PUT, COPY, POST, LIST requests to S3 standard • (20000) • GET, SELECT, and all other requests from S3 Standard • (20000)
OR3D	AWS Lambda	$0	$0	$0	• Invoke Mode (Buffered) • Architecture (x86) • Architecture (x86) • Number of requests (10000 per month) • Amount of ephemeral storage allocated (512 MB)
OR3D	DynamoDB provisioned capacity	$180	$12	$328	• Table class (standard) • Average item size (all attributes) (1 byte) • Write reserved capacity term • (1 year) • Read reserved capacity term • (1 year) • Data storage size (1 GB)
OR3D	Amazon EC2	$0	$69	$826	• Tenancy (dedicated instances) • Operating system (Linux) • Workload (consistent, number of instances: 1) • Advance EC2 instance • (c6i.16xlarge) • Pricing strategy (on demand utilization: 2 hours/day) • Enable monitoring (disabled) • EBS storage amount (512 GB) • DT inbound: not selected (0 TB per month) • DT outbound: not selected (365 GB per month) • DT intra-region: (0 TB per month)

Prototype Component	Service	Upfront	Monthly	Annual	Configuration Summary
Omniverse Nucleus	Amazon EC2	$0	$418	$5,016	• Tenancy (dedicated instances) • Operating system (Linux) • Workload (consistent, number of instances: 1) • Advance EC2 instance • (r7a.xlarge) • Pricing strategy (on demand utilization: 24 hours/day) • Enable monitoring (disabled) • EBS storage amount (1 TB) • DT inbound: not selected (0 TB per month) • DT outbound: not selected (1 TB per month) • DT intra-region: (0 TB per month)
Total		**$180**	**$523**	**$6,454**	

Table 4-3 shows estimated costs to process a single 2-week time window of all prototype variables.

Table 4-3 2-week Data Set Processing Cost.

Prototype Component	Service	Cost	Configuration Summary
Data transfer from data source	S3 Standard	$83	• S3 standard storage (7 TB per month) • PUT, COPY, POST, LIST requests to S3 standard (35000) • GET, SELECT, and all other requests from S3 standard (35000) • S3 standard average object size (200 MB)
OR3D	Amazon EC2	$129	• Tenancy (dedicated instances) • Operating system (Linux) • Workload (consistent, number of instances: 1) • Advance EC2 instance (c6i.16xlarge) • Pricing strategy (on demand utilization: 82 hours/month) • Enable monitoring (disabled) • EBS storage amount (512 GB) • DT inbound: not selected (0 TB per month) • DT outbound: not selected (365 GB per month) • DT intra-region: (0 TB per month)

Prototype Component	Service	Cost	Configuration Summary
Omniverse Nucleus	Amazon EC2	$418	• Tenancy (dedicated instances) • Operating system (Linux) • Workload (consistent, number of instances: 1) • Advance EC2 instance (r7a.xlarge) • Pricing strategy (on demand utilization: 100% utilized/month) • Enable monitoring (disabled) • EBS storage amount (1 TB) • DT inbound: not selected (0 TB per month) • DT outbound: Internet (1 TB per month) • DT intra-region: (0 TB per month)
Total		**$630**	

The highest cost services were associated with always-on infrastructure used for the prototype. This was due Omniverse Nucleus' need to be continuously available for data egress to Agatha and for receiving newly processed data from OR3D during development, which prevented on demand or scheduled utilization approaches that would increase operating efficiency. As the volume of users increase and the volume of outgoing data from the Nucleus EC2 increases commensurately, costs increase linearly with the volume of outbound data (can range from $50 to $90 per TB). This was not an issue during the prototype as there was only one Agatha user. Other prototype component costs are not directly impacted by number of users.

We recommend investigating AWS cost optimization strategies (Recommendation 2.5) such as:

- **Using caching servers on NOAA networks accessible by scientists would reduce the volume of data egress traffic from the Omniverse Nucleus EC2 to Agatha. The cost efficiency improvements provided by a caching service would depend on the number users on the networks with cached data.**
- **Adding the ability for Nucleus EC2 to 'spin down' when there are no active Agatha users. This also requires modifying O3RD to intermittently push updates to the Nucleus EC2.**
- **Using AWS Fargate to optimize EC2 processing.**

5 INTEROPERABILITY AND STANDARDIZATION WITH OTHER Digital Twins

One main goal of our study was to provide recommendations for standardization and interoperability with other digital twins being developed by academia, industry, and other Government agencies. To do this, we searched literature and reports and attended conferences, workshops, and seminars to understand the current state of other digital twins being developed for various purposes. We found that most other digital twin programs focusing on Earth science have similar goals to the NOAA EO-DT prototype objectives and are planning development initiatives on multi-year timelines. We also found that although the concepts are similar, there is a disconnect between the programs at a foundational level on requirements, standards, and formats which could slow future opportunities for interoperability. In this section, we focus on recommendations for current development to remain agile as requirements are defined and outline recommended collaboration between programs.

5.1 FLEXIBILITY AND INTEROPERABILITY OF THE LM-NVIDIA EO-DT

To achieve the flexibility and interoperability needed to interact with other digital twins, our study used multiple file formats, data types, algorithm interactions, tiling systems, and visualization tools. From our study, **we recommend a flexible digital twin architecture which has the following components: a data archive, a common data file formatter, a data input service, a service for containerized algorithm processing which outputs to a common processed file and a tiled processed file (Figure 5-1) (Recommendation 1.1).** Tiled processed output should be easily composable, enabling flexible aggregation of data layers and hierarchies. These files should be accessible by both interactive visualization tools and user interfaces, and in an accessible user repository. This general schema provides the flexibility within both large and small scale digital twins to represent simple and complex processes. Of course there will be additional specific work for each individual initiative, however this skeleton framework should be easily adapted to a variety of use cases. More specifically, the NASA ESDT defined their architecture framework with an observational data repository, ingest subsystem, digital twin information subsystem (including a digital replica, digital twin record, and external repositories for source data), nominal forecast subsystem, impact assessment subsystem, control and monitor subsystem, and user interface (Le Moigne et al., 2023). This framework is broader than our prototype which only maps to their ingest subsystem, digital twin information subsystem, and user interface, though the overlaps that do occur are in agreement.

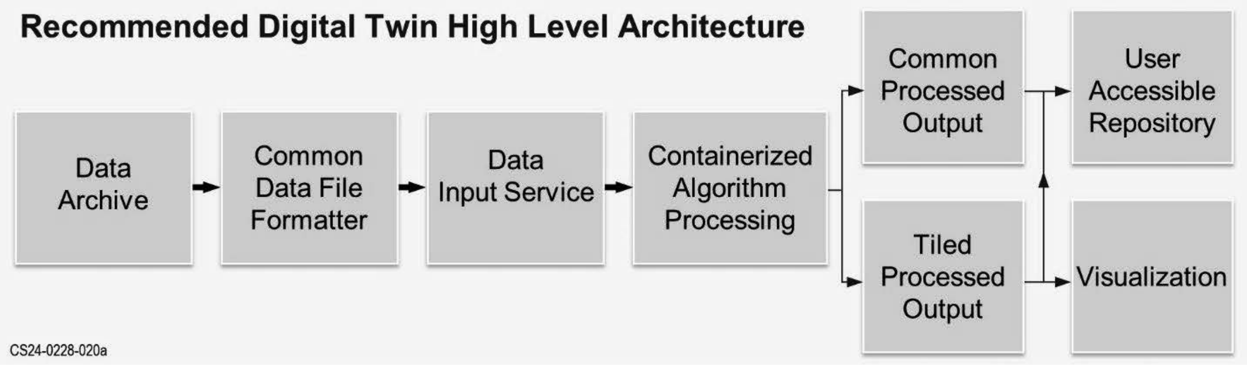

Recommended Digital Twin High Level Architecture
Data Archive
Common Data File Formatter
Data Input Service
Containerized Algorithm Processing
Common Processed Output
Tiled Processed Output
User Accessible Repository
Visualization
CS24-0228-020a

Figure 5-1 Recommended Digital Twin High-Level Architecture.

We ingested data coming from ground and space-based assets along with model data which cover both point and gridded raster data types. For some use cases, we built custom algorithms to take point data and transform it into a raster if it was efficient and fit the use case. It was then projected into an ECEF coordinate system. From our study, **we recommend that the NOAA EO-DT has a method to ingest both point and raster data independently since data types vary so widely from sensors and models (Recommendation 2.1).** To further standardize data, **we recommend NOAA chooses a single coordinate system depending on the type of digital twin and use case to re-grid and standardize all data into a common spatial grid across all similar Earth system domain sensors (Recommendation 2.1)**. ECEF works for most Earth centered measurements like SST, SIC, temperature, moisture, and fire, however we ran into many issues with using ECEF for space weather data. **We recommend using the GSM coordinate system to best integrate space weather data such as solar wind magnetic shear (Recommendation 2.4).**

In addition to data types and coordinate system standardization, we studied a wide variety of data formats (e.g.,.nc,.txt,.tiff, etc.). With NOAA's current data system architecture, we found NetCDF4 files to be the most common and did not see a large performance decrease even with a non-cloud-optimized file format with the amount of data we ingested, though we did not stress test beyond our 2-week window for our prototype dataset. This may impact processing time on an operational scale with terabytes of data incoming. Although our backend can ingest any of these data formats, we found that standardizing to one data format made ingestion more efficient. Modern NetCDF4 supported by a suite of Open Geospatial Consortium (OGC) standards, and is built upon HDF5. HDF5, and thus NetCDF, are expected to benefit from ongoing technical evolution and advances that will permit direct and highly efficient GPU-accelerated I/O within AI data processing pipelines. Further, anticipated work within NVIDIA on open source submissions to HDF5 for optimized access to object stores will benefit digital twin projects using AWS S3 and Omniverse Nucleus. A 2020 NASA study on cloud-optimized file formats included evaluation of HDF5, NetCDF, GeoTIFF, nascent cloud-optimized variants, and other popular file formats against a variety of criteria relevant to EO-DT and digital twins generally. **We recommend using file formats that support flexible incorporation of arbitrary metadata (e.g., XML, OGC NetCDF, OpenUSD, and similar) to ensure complete data provenance continuity, to permit incorporation of RDF knowledge graph tags, and to provide visualization tools and user interfaces to exploit the use of metadata to improve performance and user experience (Recommendation 2.2).** We expect that accelerated AI data processing I/O performance, extensibility, and several of the evaluation criteria in the NASA report will remain highly relevant for implementing digital twins of Earth. **We recommend choosing one common observational data format and requiring users to provide data in that standard format prior to the digital twin ingesting it or using a common data file formatter (Recommendation 2.1).**

A major issue we ran into was non-standard metadata. This included flags, units, scaling, and data naming conventions. Each instrument and model has their own standards which makes ingesting, comparing, fusing, or analyzing each different sensor data type difficult because you have to refer to individual documentation for each. **We would recommend NOAA either provide more data standardization for all NOAA data products, or provide a preprocessing step prior to digital twin ingestion through a data template or with a configuration file (Recommendation 2.1)**. Having a data template or configuration file that users interact with to define or map their particular data to common required parameters (e.g., variable names, datetime, units, flags, EPSG projection) prior to ingestion will be essential for a wide variety of users to interact easily with the EO-DT.

We also recommend using RDF tags in observational databases to make the immense amount of data coming into NOAA's system from satellites, ground-based observations, and model output easily usable and ingestible into a digital twin architecture (Recommendation 2.3). Regardless, the digital twin will only run as well as the data provided to it so these initial data preprocessing and ingestion steps are vital to creating a successful system.

Once the data is successfully preprocessed, ingested, and put into a common spatiotemporal grid, any algorithms can be applied. **We recommend that NOAA's digital twin prototype has built in algorithms for data fusion and anomaly detection using simple methodologies like linear, bilinear, or cubic interpolation options and subtracting from a previous timestep to show how data is changing on a short time scale (Recommendation 2.1)**. There could also be cached datasets to see how data is changing on a longer time scale. These simple algorithms would provide users with the ability to do quick comparisons with low processing costs, as opposed to large-scale custom algorithms for each user. In addition, our system provides a plug-and-play algorithm system which we applied the MIIDAPS algorithm to as a preprocessing step (Figure 3-3). **We recommend that a standardized template is used for algorithms defined for the digital twin so inputs and outputs can plug in and be integrated more seamlessly (Recommendation 2.1).** Providing customized algorithm support for each user, especially on complex containerized algorithms with multiple different data types would require additional development.

In our study, we tested both Uber H3 and Google S2 hierarchical tiling systems which are two of the most popular tiling systems. We chose the Uber H3 tiling system since it is well represented at the poles with its hexagonal tiling scheme, compared to Google S2's rectangular scheme. Its persistent, georeferenced index identifier that is accessible to both the backend and frontend allows for swift and precise reconstruction of cells and tiles, and easy incorporation of tiles into on-disk file formats such as NetCDF and OpenUSD. Uber H3 provides extensive online documentation allowing for easier development compared to Google S2. In addition to tiling systems, we also looked into different tile formats including the Cesium OGC 3-D Tiles Community Standard and OpenUSD, which are both compatible with our frontend visualization. Tiled processed outputs were written in OpenUSD format, enabling interoperability with a large ecosystem of other software tools. The data aggregation flexibility OpenUSD provides and standards being developed within the Alliance for OpenUSD permit aggregation and composition of EO-DT data with a wide variety of other complementary geospatial, architectural, and built environment data types and sources (Cozzi, 2023; OpenUSD, 2023). OpenUSD's support for flexible incorporation of schemas and metadata has led to widespread use for synthetic data generation for AI training, where semantic labeling is required. Ongoing development of a standard semantic schema for OpenUSD is anticipated to benefit a broad range of cases by specifying a high-performance implementation approach but without defining specific semantics, taxonomies, or ontologies. A recent implementation of interactive taxonomy computation, label resolution and display has been developed for incorporation into Pixar's USDView, achieving near-instantaneous response, even for deep OpenUSD prim hierarchies. We anticipate the ongoing work to standardize encoding semantic information in OpenUSD will directly benefit EO-DT and similar geospatial use cases that could leverage knowledge graphs and semantic labels in a wide variety of uses.

We also studied how best to visualize data and tested our output with multiple viewers including Agatha, Pixar USDView, and NVIDIA Omniverse/Earth2 (Figure 5-2). In our prototype we quantized values in the USD and PNG files per time step, thereby dramatically reducing data storage and transmission requirements for visualization by end users, with commensurate reductions in data egress operational costs. The underlying observational data is quantized and the quantized numeric values directly reference a color map, with a default color scale encoded in the files on-disk and subsequently transmitted for viewing by Agatha. We note that any choice of a single quantization approach could potentially result in unacceptable loss of precision for some data, or for accuracy-sensitive uses not envisioned. Therefore, **we recommend that aggressive quantization and compression techniques be used where and when possible, but that it will remain necessary for visualization software to also be able to perform its own color transfer functions on full-precision floating point data as well** (i.e., sending the raw data information from the back end or from a cloud service that responds to full-precision quantitative data analysis queries, and not a quantized or loss-compressed output) **(Recommendation 2.6).** While we presently use a single scheme, in a production implementation, there should be options to use 8-bit or 16-bit quantized representations, as well as 32-bit or 64-bit floating point, depending on the particular data and use case combination being served. **In the end, we recommend taking a**

quantization or compression approach based on the precision and cost driven by the individual use case (Recommendation 2.6).

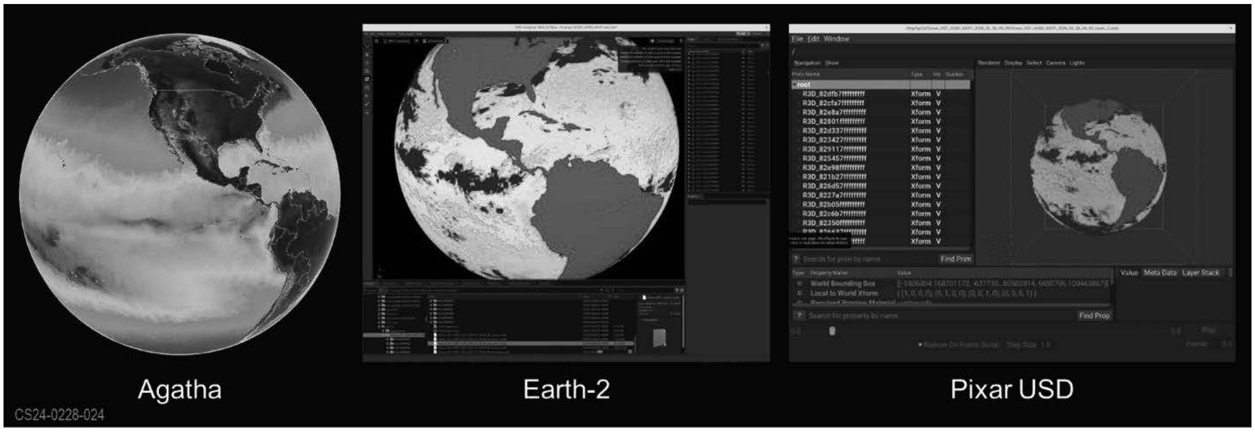


Figure 5-2 EO-DT Data Displayed in Three Separate Viewers a) Agatha, b) Earth-2, and c) Pixar USD to Demonstrate Interoperability with the OpenUSD File Format.

Lastly, we learned that working with 3D point data (e.g., IGRA) in our visualization was especially difficult since Agatha mainly performs well showing surface level data. In Agatha, a custom feature was added to have a 3D zoom to show 20 vertical levels but we view this as a preliminary feature that needs more development and user feedback to be useful by intended users. **Features like temperature profiles at such a high vertical resolution may be best to visualize by a more standard view in a 2D skew-T plot instead for practical applications (Recommendation 3.2).** For our fusion algorithm, Agatha does not know which sensor measurements go into fused and anomalous data so a potential future task which would benefit users is to track that in metadata so they can dynamically turn on and off sensors in those algorithms from the frontend. Finally, Agatha does require a GPU to run the desktop application we provide as part of our prototype for high quality performance. For the most flexible platform for users **we recommend further looking into web-based applications or streaming to remove the resource requirement for digital twin visualization (Recommendation 3.2)**.

5.2 INTEROPERABILITY WITH OTHER DIGITAL TWINS

As part of our study, we looked at the development of other digital twins such as NASA ESDT, Digital Twins of the Ocean (DITTO), Destination Earth, and NVIDIA's Earth-2. We also investigated the Digital Twin Earth Framework Specification (DTE-FS) which is useful in guiding construction of a model-based framework that enables implementation of a multi-viewpoint, evolutionary, and communally managed DTE that is represented as a syntactically, schematically, semantically, and legally interoperable system of systems (Berkheimer, 2022; Berkheimer, 2023). With multiple digital twin programs occurring and being developed in parallel, there needs to be a central authority to define standards so they can be adapted to a common platform. Our EO-DT prototype mainly focused on building a flexible platform to provide recommendations in alignment with the DTE-FS, but it was often difficult to make architectural decisions on implementation when universal standards are not set to transition to an operational digital twin. There are also many open source frameworks already in use for data processing like OpenEO, funded by the European Space Agency and to potentially be used in DestinE, and data/metadata sharing (e.g., Ocean Data and Information System (ODIS) used by DITTO) which should be considered to provide common standards and requirements to be adopted. **We recommend coordination**

between digital twin programs to adopt common standards for interoperability since the goals of each program align fairly directly (Recommendation 3.1).

NASA's ESDT is led by NASA AIST and the goals are to:

- Develop information system frameworks that provide continuous and accurate representations of systems as they change over time.
- Mirror various Earth science systems and use the combination of data analytics, AI, Digital Thread and state-of-the-art models to help predict the Earth's response to various phenomena.
- Provide the tools to conduct 'what if' investigations that can result in actionable predictions (Le Moigne and Smith, 2022).

The ESDT program is similar to a digital twin focusing specifically on the oceans, DITTO. DITTO has a goal of establishing and advancing a high-performance computing digital framework to access, manipulate (e.g., using AI/ML), analyze, visualize, and effectively use marine data and model output. (Bahurel et al., 2023). Similar to ESDT, the DITTO platform will also enable users to address 'what if' questions based on shared data, models and knowledge. Both programs are slated to be expanded using a bottom up approach with smaller scale studies and prototypes to eventually be built into a larger scale effort over the next five to ten years.

Conversely, the European Commission's Destination Earth (DestinE) and NVIDIA's Earth-2 are taking a top down approach to have a full digital replica of the Earth. DestinE plans to support tackling complex environmental challenges to:

- Monitor and simulate the Earth's system developments (land, marine, atmosphere, biosphere) and human interventions.
- Anticipate environmental disasters and resultant socioeconomic crises to save lives and avoid large economic downturns.
- Enable the development and testing of scenarios for ever more sustainable development.

Earth-2 is an internally developed digital twin platform that permits coupling of state-of-the-art accelerated, AI-augmented, high-resolution climate and weather simulations together with powerful interactive visualizations. Both of these programs are being developed at a rapid pace planning for a fully integrated platform using AI/ML and GPU-accelerated processing.

All of these digital twin efforts aligns with common goals to ingest large amounts of data to understand both large and small scale Earth processes for historical, current, and future environmental conditions. They also each intend to show how differing conditions can impact the Earth's processes through simulations. Each digital twin model relies upon scalability and may use AI/ML for processing or applying algorithms and likely will need cloud infrastructure to support this massive endeavor. **To reap the benefit of interoperability between all of these digital twins beyond common standard so they can share data and output with one another, we recommend workshops between the programs to plan in advance how they can interact (Recommendation 3.1).** This is similar to Recommendation #5 in the Foundational Research Gaps and Future Directions for Digital Twins report stating that "Agencies should collaboratively and in a coordinated fashion provide cross-disciplinary workshops and venues to foster identification of those aspects of digital twin R&D that would benefit from a common approach and which specific research topics are shared." (National Academies, 2024). This recommendation goes hand in hand with all ongoing programs adherence to the Findability, Accessibility, Interoperability, and Reusability (FAIR) principles (Wilkinson et al., 2016; Brönner et al., 2023; Wearing et al., 2024; Berkheimer, 2024). Communication during the development stage of these programs will be vital to constructing them in the most flexible way possible and can provide insight into other parts of the development processes where they may also align.

5.3 FEDERATED SYSTEM OF DIGITAL TWINS

For this study, we also wanted to look at how best to architect an operational digital twin system. Digital twin systems have been used for many different purposes including in the manufacturing, healthcare, spacecraft, and other disciplines (Soori et al., 2023; Sun et al., 2023; Pinello et al., 2024). The Earth science use case includes combining multiple different Earth systems which are individually complex (Henrikson et al., 2022; Barros, 2024; Brocca et al., 2024). **This leads us to recommend a federated system of digital twins rather than a one fits all to capture domain and process scale models (Recommendation 1.3).** Domain scale digital twins would capture Earth system domains that the EO-DT already encapsulates while process scale digital twins could capture even finer grain processes that build the foundation of smaller scale processes derived from observations (Figure 5-3). For example, a domain scale digital twin of the ocean would encompass process scales of biogeochemical interactions, carbon sources and sinks, and impacts of SIC. Some of these smaller scale processes would overlap with other domain scale digital twins since SIC also impacts the cryosphere. Depending on the ideal granularity of the system, interconnected domain and process scale digital twins could interoperate to represent coupled systems and interact with each other. **For the entire global system we do not recommend a single digital twin as the processes are too complex and the amount of processing needed is too large to realistically integrate all systems. Even if processing unlimited, the software complexity and difficulty of creating effective user interfaces for diverse digital twin use cases are barriers to a completely unified digital twin approach (Recommendation 1.3).**

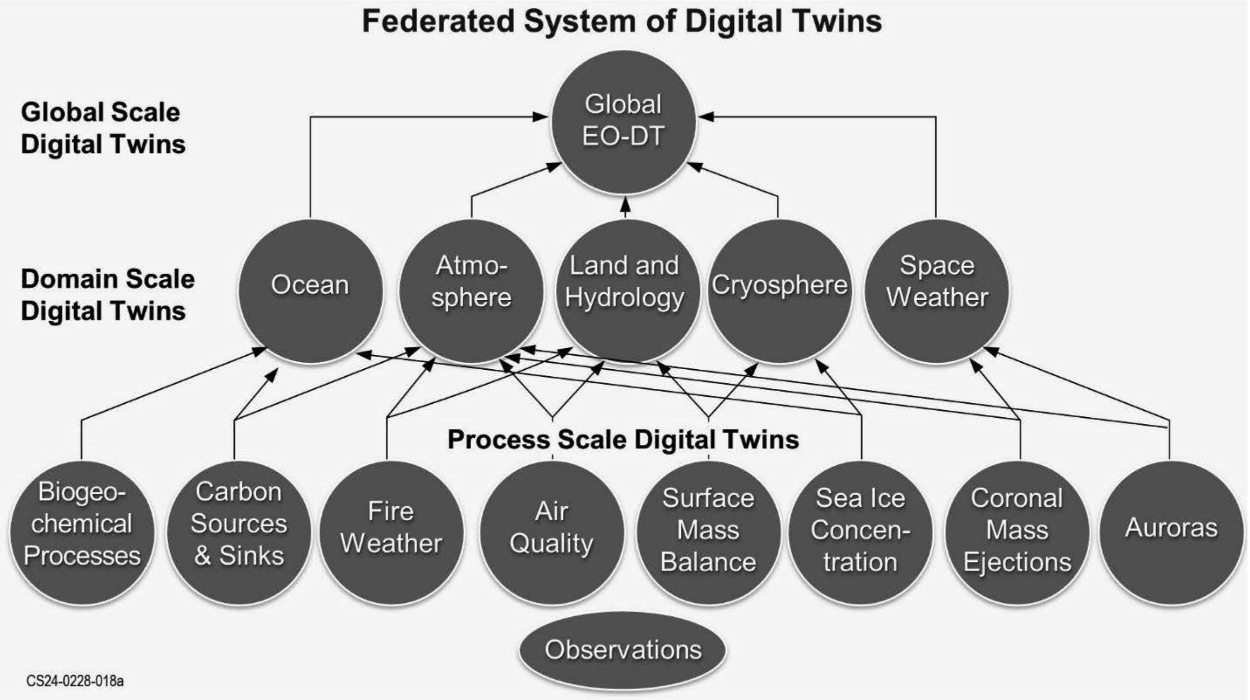


Figure 5-3 Example Federated System of Digital Twins. *Flow diagram includes global, domain, and process scale digital twins included with a foundation of observations.*

A federated system of digital twins would require integration and interoperability. Broadly, it is difficult to design a system that would work for every use case or persona covering all Earth system domains. The main considerations would be transferring raw data and derived products from algorithm/model output between the digital twins at each level which would require the common data formats we mention in Section 5.1. This federated system would require data calls to be made both hierarchically and laterally (Figure 5-3). An example for a specific use case may be for a cryospheric scientist interested in how the Arctic will change during the summer. This persona requires knowledge about multiple different complex processes including surface mass balance, SIC change, shallow and deep ocean current interactions, albedo changes, etc. Each of these processes could represent their own digital twin, although some observations would be shared depending on how the processes are driven. Often processes will impact one another (e.g., albedo drives snow melt) so an output of one process digital twin may be the input to another. This is similar in the domain scale digital twins where impacts to the cryosphere will impact the ocean (e.g., outlet glaciers releasing meltwater may cool down temperatures significantly).

We also recommend to use Observing System Simulation Experiments (OSSEs) to study how particular large or small scale processes could be integrated into a digital twin infrastructure (Recommendation 1.4). OSSEs are used to simulate and assess the impacts of new observing systems planned for the future or the impacts of adopting new techniques for exploiting data or for forecasting. (Boukabara et al., 2018) This is similar to the work already being done in the NASA ESDT program in which they are exploring analytic collaborative frameworks toward ESDT, AI and ML-based Surrogate Modeling for ESDT, ESDT Infrastructure, and ESDT Prototypes through the AIST-21 Solicitation Awards (Le Moigne, 2022). Each of these NASA-funded programs will provide guidance into exploiting data for subdomain processes and functionality. Most vitally, similar to this program, we recommend that any OSSE or explorative program brings back lessons learned to the larger community.

6 RECOMMENDATIONS AND LESSONS LEARNED

The goal of this BAA was to study an integrated Earth system replica of the Earth environment with multiscale, multi-variables features, and integrating a large set of observing systems and environment analyses systems.

We highlight the goals of this program referenced from Section 1 and a high-level summary in Table 6-1. More in-depth recommendations which are tied to each goal by a reference number were mentioned earlier in the text are detailed into Table 6-2.

Table 6-1 High-Level Overview of Recommendations.

Goal	Quick Summary
1. Provide NOAA with a functioning, scalable prototype that may serve as the foundation of next-generation ground enterprise system.	While developing our prototype, our main challenges included handling non-standardized data and metadata, creating a user interface that represents all Earth domains well, and determining a stable way to integrate custom algorithms.
2. Determine cost estimates for maintaining a digital twin and scaling it to store large amounts of data.	Operational: $523/month—Continuous processing of hourly SST data (one geophysical variable). Development: $630/month—Processing a 2-week window of data from all domains for this prototype.

Goal	Quick Summary
3. Provide recommendations for standardization and interoperability with other digital twins.	Multiple efforts happening concurrently should be aligned to support architecture and data standardization decisions in the early stages which will provide easier interoperability and integration in the future.
4. Study how a digital twin can benefit NOAA as an R&D product and an operational product.	We recommend using a lighter weight implementation for developing and testing prototype software compared to building out a scalable operational product which impacts both architecture choices and costs.

Table 6-2 Detailed Recommendations with References to Goals Provided in Section 1.2.2.

Rec	Digital Twin Architecture and Costs
1.1	**Lesson Learned (1, 4):** Certain architectural components of a digital twin are non-negotiable and understanding inputs and outputs of each part of the system is vital to creating a successful platform. We found the necessary components of a digital twin to be: a data archive, a common data file formatter, a data input service, a service for containerized algorithm processing which outputs to a common processed file ands well as a tiled processed file (Figure 5-1). (Section 5.1)
1.2	**Recommendation (1, 4):** We recommend using the static processing method for quick prototyping and R&D use cases and using the live processing method for processing data operationally using verified and highly tested pipelines. Our static data pipeline allows for more flexibility with users to control the amount and range of data being processed (e.g., historical events) versus wanting a near real-time operational pipeline with the live processing method. There is also flexibility in being able to pre- and post-process data more easily as able to do more piecewise development including running containers outside of the NOAA processing engine (Section 3.1.3). The live processing pipeline is mainly meant for scalability and integrated, tested algorithm implementation on cloud computing resources. (Figure 3-1 and Figure 3-2, Sections 3.1.1 and 3.1.2) For operational use, AWS offers a dynamic compute resource service called Fargate. Instead of needing to allocate an EC2 beforehand, we could leverage Fargate in the future and have it determine the best compute resource to use for a given set of data which could also improve cost efficiency.

Rec	Digital Twin Architecture and Costs
1.3	**Lesson Learned (1, 3)**: For the entire global system we found using a single digital twin was too simplistic of a framework since processes are too complex and the amount of processing needed is too large to realistically integrate all systems. **Recommendation (1, 3)**: The EO-DT should use a federated system of digital twins rather than a one fits all to capture domain and process scale models. Our EO-DT looked geophysical variables in different Earth domains that change on varying temporal scales from minutes to hours to days (Table 2-2). Each system domain includes a multitude of large and small scale subprocesses which are individually complex (Figure 5-1). In addition, from conferences and literature we found many instances of smaller field and domain-specific digital twins which would be more easily integrated into a federated system rather than a complex single system (Section 5.2 and 5.3). A system of digital twins that feed into one another would provide a more comprehensive view of the whole Earth system. More specifically, we highly recommend space weather has its own digital twin due to issues with its native non-Earth-centric coordinate system (Section 3.1.4.5). Regardless of the challenge of processing and incoming massive amounts of data into one singular digital twin, the software complexity and difficulty of creating effective user interfaces for diverse digital twin use cases are barriers to a completely unified digital twin approach (Section 5.3).
1.4	**Recommendation (4):** OSSEs can be used to study how particular large or small scale processes could be integrated into a digital twin infrastructure. (Section 5.3) In a complex system such as the Earth, there is an enormous amount of overlap in geophysical processes. To build a digital twin of the system, smaller scale simulation experiments should be used to determine how these processes of different Earth system domains can be integrated into this federated system of models. OSSE's provide the ideal methodology to do small scale integration and provide a sandbox for testing prior to implementation.

<table>
<tr><th>Rec</th><th>Standardization, Data, and Formatting</th></tr>
<tr><td><b>2.1</b></td><td><b>Recommendation (1, 3, 4):</b> We would recommend either NOAA provide more data standardization for all data products, or provide a preprocessing step prior to digital twin ingestion through a data template. (Section 5.1)

Each data type we ingested, either from satellite and ground observations or model output, in raster or point data, required an understanding of variables, formats, units, and metadata. This was a challenge when it came to platform flexibility. It will be necessary to either require users to conform to a common projection and data format or build in tools to dynamically standardize different incoming data types to ingest different types of observation, sensor, and model data for different use cases. In addition, making sure standardized variable names or having a map will be necessary (e.g., ‘lat’ = ‘latitude’, ‘long’, ‘lon’ = ‘longitude’). An AI/ML technique like natural language processing may be useful in building something dynamic to understand user intent and incoming metadata in the future.

This data standardization will also directly impact standard or custom AI/ML algorithms which are integrated into the platform. To seamlessly integrate custom use cases inputs and outputs must be in a standard format and data type which is expected by the algorithm. Even for more simple methodologies applied for data fusion and anomaly detection such as interpolation or subtraction data must be in a particular format to get a consistently correct output. If these are built into the platform as a standard algorithm, there may be a benefit to knowing how to pre-process rather than having to do a custom change each time for individual users fusion or anomaly algorithms. This general standardization would also benefit interoperability with other digital twins so data could be easily integrated.</td></tr>
<tr><td><b>2.2</b></td><td><b>Recommendation (1, 3)</b>: We recommend using file formats that support flexible incorporation of arbitrary metadata (e.g., XML, OGC NetCDF, OpenUSD, and similar) to ensure complete data provenance continuity, to permit incorporation of RDF knowledge graph tags, and to provide visualization tools and user interfaces to exploit the use of metadata to improve performance and user experience. (Section 5.1)</td></tr>
<tr><td><b>2.3</b></td><td><b>Recommendation (1, 3)</b>: We also recommend using RDF tags in observational databases to make the immense amount of data coming into NOAA’s system from satellites, ground-based observations, and model output easily usable and ingestible into a digital twin architecture. (Section 3.1.4 and Section 5.1)</td></tr>
</table>

Rec	Standardization, Data, and Formatting
2.4	**Lesson Learned (1):** Integrating space weather data into the same digital twin framework that dealt with surface or tropospheric Earth system domains was extremely difficult due to the magnitude of scale difference and juxtaposing ideal reference coordinate systems. **Recommendation (1):** We recommend using the GSM coordinate system instead to best integrate space weather data such as solar wind magnetic shear. (Section 3.1.4.5) Space weather brought a unique set of challenges to our EO-DT program. Raw data was difficult to work with as it was not in a common coordinate projection and the solar wind magnetic shear product we demonstrated was a derived product from multiple sensors on DSCOVR. We had to adapt both our back- and frontend systems to work with a new data format and had to study how to best show off-Earth data when it was orders of magnitude farther away than our upper-tropospheric 3D temperature and moisture profiles. This brought a lot of discussion of if it would be better to have a separate digital twin for space weather data which could eventually be tracked at a high level in another global digital twin to study interconnected processes. (Recommendation 1.3)
2.5	**Recommendation (2, 4):** The EO-DT AWS architecture should minimize utilization (cost) by implementing caching systems on NOAA networks and using on demand services as much as possible. (Section 4) During prototype development most AWS services were in an ‘always-on’ configuration to maximize availability for the team to integrate and test. An operational EO-DT could be made available on an as-needed basis, or at a minimum have scheduled data updates.
2.6	**Lesson Learned (1, 3, 4):** Quantization can cause loss of data precision but is important and necessary for compression. **Recommendation (1, 3, 4):** We recommend taking a quantization or compression approach based on the precision and cost driven by the individual use case. (Section 5.1) In our prototype we quantized values in the USD and PNG files per time step, thereby dramatically reducing data storage and transmission requirements for visualization by end users, with commensurate reductions in data egress operational costs. The underlying observational data is quantized and the quantized numeric values directly reference a color map, with a default color scale encoded in the files on-disk and subsequently transmitted for viewing by Agatha. We note that any choice of a single quantization approach could potentially result in unacceptable loss of precision for some data, or for accuracy-sensitive uses not envisioned. Therefore, we recommend that aggressive quantization and compression techniques be used where and when possible, but that it will remain necessary for visualization software to also be able to perform its own color transfer functions on full-precision floating point data as well (i.e., sending the raw data information from the back end or from a cloud service that responds to full-precision quantitative data analysis queries, and not a quantized or loss-compressed output).

Rec	Digital Twin Interoperability
3.1	**Recommendation (3):** To reap the benefit of interoperability between digital twins we recommend coordination between these digital twin programs to adopt common standards for interoperability since the goals of each program align fairly directly. (Section 5.2) By connecting programs like EO-DT, ESDT, Earth-2, DITTO, and DestinE who have common goals of creating digital replicas and simulating future conditions, efforts can be combined to create better solutions and any duplicate work can be cut. Coordination between the programs will allow for accelerated development and creative problem solving by bringing together developers with different backgrounds. Even if there is not direct coordination in sharing development or algorithms, at a minimum standard data files and formats should be agreed upon from the outset so integration and interoperability will be a smooth transition in the future. This recommendation goes hand in hand with Recommendation #5 in the Foundational Research Gaps and Future Directions for Digital Twins report stating "Agencies should collaboratively and in a coordinated fashion provide cross-disciplinary workshops and venues to foster identification of those aspects of digital twin R&D that would benefit from a common approach and which specific research topics are shared." (National Academies, 2024).
3.2	**Lesson Learned (1, 3):** From interacting with scientists, software developers, students, and the general community at conferences and demonstrations we found there are a wide variety of users and use cases for the EO-DT. **Recommendation (1, 3):** Using an interactive, intuitive user platform is key to the adoption and long term use of a digital twin system by the community. (Section 3.9, Figure 3-15 and Figure 3-16). When demonstrating the EO-DT at conferences and talks in the community, we received quite a bit of feedback regarding features implemented in our prototype. During the program we tried to incorporate as many as possible within scope including custom features like dynamic color bar ranges and colormaps/gradients so users could look at data of interest for their individual use cases. Users can also import or create custom colormaps to fit their data and present it to others in their field. We also added in a feature where a user can copy and paste metadata to share it quickly and easily with others. Understanding users use cases across all fields who will use it will drive what other custom features will need to be implemented in the user interface. An example of this is that radiosonde data may be best viewed on a 2D skew-T plot rather than on a 3D globe as it is difficult to view. We found that the easier to use, the more users who will want to engage with the digital twin platform. Further, we found the higher the performance requirements, the more difficult it is for all users to access so we recommend looking at a web-based application in parallel with a high compute approach. Both solutions are important for different users and use cases.

7 REFERENCES

Bahurel, P., Brönner, U., Buttigieg, P.-L., Chai, F., Chassignet, E., & Devey, C. (2023). *DITTO Programme White Paper.*

Barros, A. P. (2024). Digital Twin Earth: The next-generation Earth Information System. *Frontiers in Science*, *2*, 1383659.

Berkheimer, R. (2022). *DTE-FS: Describing A Model Based System Specification for a Digital Twin Earth Framework. 2022*, IN42B-0330. AGU Fall Meeting Abstracts.

Berkheimer, R. (2024, April 23). *Webinar: Enabling an Iterative Open Science Transformation to the Geoverse at NOAA with a Federated Knowledge Mesh with Ryan Berkheimer*. CI Compass.

Boukabara, S.-A., Ide, K., Shahroudi, N., Zhou, Y., Zhu, T., Li, R., Cucurull, L., Atlas, R., Casey, S. P. F., & Hoffman, R. N. (2018). Community Global Observing System Simulation Experiment (OSSE) Package (CGOP): Perfect Observations Simulation Validation. *Journal of Atmospheric and Oceanic Technology*, *35*(1), 207–226.

Brocca, L., Barbetta, S., Camici, S., Ciabatta, L., Dari, J., Filippucci, P., Massari, C., Modanesi, S., Tarpanelli, A., Bonaccorsi, B., Mosaffa, H., Wagner, W., Vreugdenhil, M., Quast, R., Alfieri, L., Gabellani, S., Avanzi, F., Rains, D., Miralles, D. G., … Fernandez, D. (2024). A Digital Twin of the terrestrial water cycle: A glimpse into the future through high-resolution Earth observations. *Frontiers in Science*, *1*, 1190191.

Brönner, U., Sonnewald, M., & Visbeck, M. (2023). *Digital Twins of the Ocean can foster a sustainable blue economy in a protected marine environment.*

Committee on Foundational Research Gaps and Future Directions for Digital Twins, Board on Mathematical Sciences and Analytics, Committee on Applied and Theoretical Statistics, Computer Science and Telecommunications Board, Board on Life Sciences, Board on Atmospheric Sciences and Climate, Division on Engineering and Physical Sciences, Division on Earth and Life Studies, National Academy of Engineering, & National Academies of Sciences, Engineering, and Medicine. (2024). *Foundational Research Gaps and Future Directions for Digital Twins* (p. 26894). National Academies Press.

Cozzi, P. (2023, August 1). Cesium Joins the Alliance for OpenUSD. *Cesium Joins the Alliance for OpenUSD*.

Destination Earth. (n.d.). Destination Earth. Retrieved August 22, 2024.

Durre, I., Yin, X., Vose, R. S., Applequist, S., & Arnfield, J. (2018). Enhancing the Data Coverage in the Integrated Global Radiosonde Archive. *Journal of Atmospheric and Oceanic Technology*, *35*(9), 1753–1770.

Earth System Digital Twins. (n.d.). NASA Earth Science and Technology Office. Retrieved August 22, 2024.

Henriksen, H., Schneider, R., Koch, J., Ondracek, M., Troldborg, L., Seidenfaden, I., Kragh, S., Bøgh, E., & Stisen, S. (2022). A New Digital Twin for Climate Change Adaptation, Water Management, and Disaster Risk Reduction (HIP Digital Twin). *Water*, *15*(1), 25.

Le Moigne, J. (2022, May 17). *Earth System Digital Twins (ESDT) Technology for NASA Earth Science*.

Le Moigne, J., & Smith, B. (2022). *Earth Systems Digital Twin (ESDT) Workshop Report*.

Maddy, E. S., & Boukabara, S. A. (2021). MIIDAPS-AI: An Explainable Machine-Learning Algorithm for Infrared and Microwave Remote Sensing and Data Assimilation Preprocessing - Application to LEO and GEO Sensors. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *14*, 8566–8576.

Moigne, J. L., Little, M. M., Morris, R. A., Oza, N. C., Ranson, K. J., Riris, H., Rogers, L. J., & Smith, B. D. (2023, October 1). *Earth System Digital Twin (ESDT) Architecture Framework*.

NVIDIA Earth 2 Platform. (n.d.). NVIDIA. Retrieved August 22, 2024, from

OpenUSD, A. for. (2023, December 12). Alliance for OpenUSD Unveils Roadmap for Core USD Specification and Ecosystem Collaboration. *The Alliance for OpenUSD (AOUSD)*.

Pinello, L., Brancato, L., Giglio, M., Cadini, F., & De Luca, G. F. (2024). Enhancing Planetary Exploration through Digital Twins: A Tool for Virtual Prototyping and HUMS Design. *Aerospace*, *11*(1), 73.

Soori, M., Arezoo, B., & Dastres, R. (2023). Digital twin for smart manufacturing, A review. *Sustainable Manufacturing and Service Economics*, *2*, 100017.

Sun, T., He, X., & Li, Z. (2023). Digital twin in healthcare: Recent updates and challenges. *DIGITAL HEALTH*, *9*, 205520762211496.

Trattner, K. J., Petrinec, S. M., & Fuselier, S. A. (2021). The Location of Magnetic Reconnection at Earth’s Magnetopause. *Space Science Reviews*, *217*(3), 41.

Wearing, M., Malina, E., & Fernandez, D. (2024). *Earth Observation Based Digital Twin Components of the Earth System*. 20503. EGU General Assembly Conference Abstracts.

Wilkinson, M. D., Dumontier, M., Aalbersberg, Ij. J., Appleton, G., Axton, M., Baak, A., Blomberg, N., Boiten, J.-W., da Silva Santos, L. B., Bourne, P. E., Bouwman, J., Brookes, A. J., Clark, T., Crosas, M., Dillo, I., Dumon, O., Edmunds, S., Evelo, C. T., Finkers, R., … Mons, B. (2016). The FAIR Guiding Principles for scientific data management and stewardship. *Scientific Data*, *3*(1), 160018.