



National Environmental  
Satellite, Data, and  
Information Service

National Centers for Environmental Information

# Enabling an Iterative, Open Science Transformation to the Geoverse with a Federated Knowledge Mesh

American Meteorological Society Winter Meeting  
January 30, 2024  
Charm City

**Ryan Berkheimer, Evan McQuinn, Brandon Shelton, Monica Youngman, Ken Casey, Nancy Ritchey, Rich Baldwin, Jeff Arnfield, Alan Hall, Daniel Lloyd, David Neufeld, Ken Tanaka, Pura Perez, Brent Hefner, Sarah Menassian, Charles Burris, Brian Meyer, Jessica Hung, John LaRocque, et al.** (a non-exhaustive list of a lot of great and forward looking people)

# Where To? Geoverse Semantics in Context

- A democratized 'system of systems' powered in large part by machine to machine communication
  - Systems may be 'smart agents'
- An ecosystem in which all users can both consume and contribute information
- A federated trust-based framework for realizing universally useful understanding

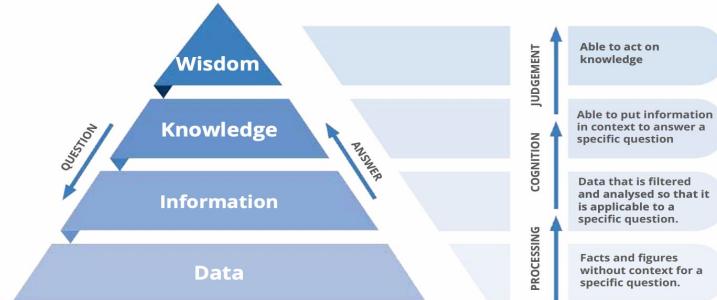
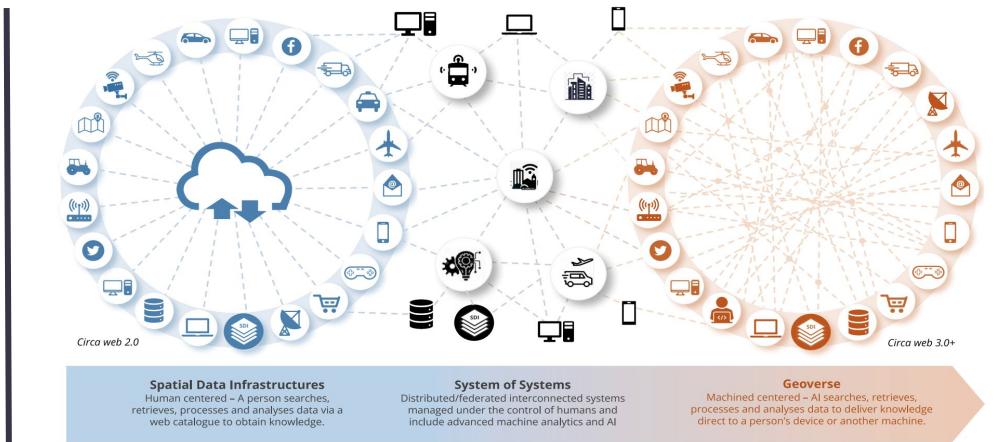
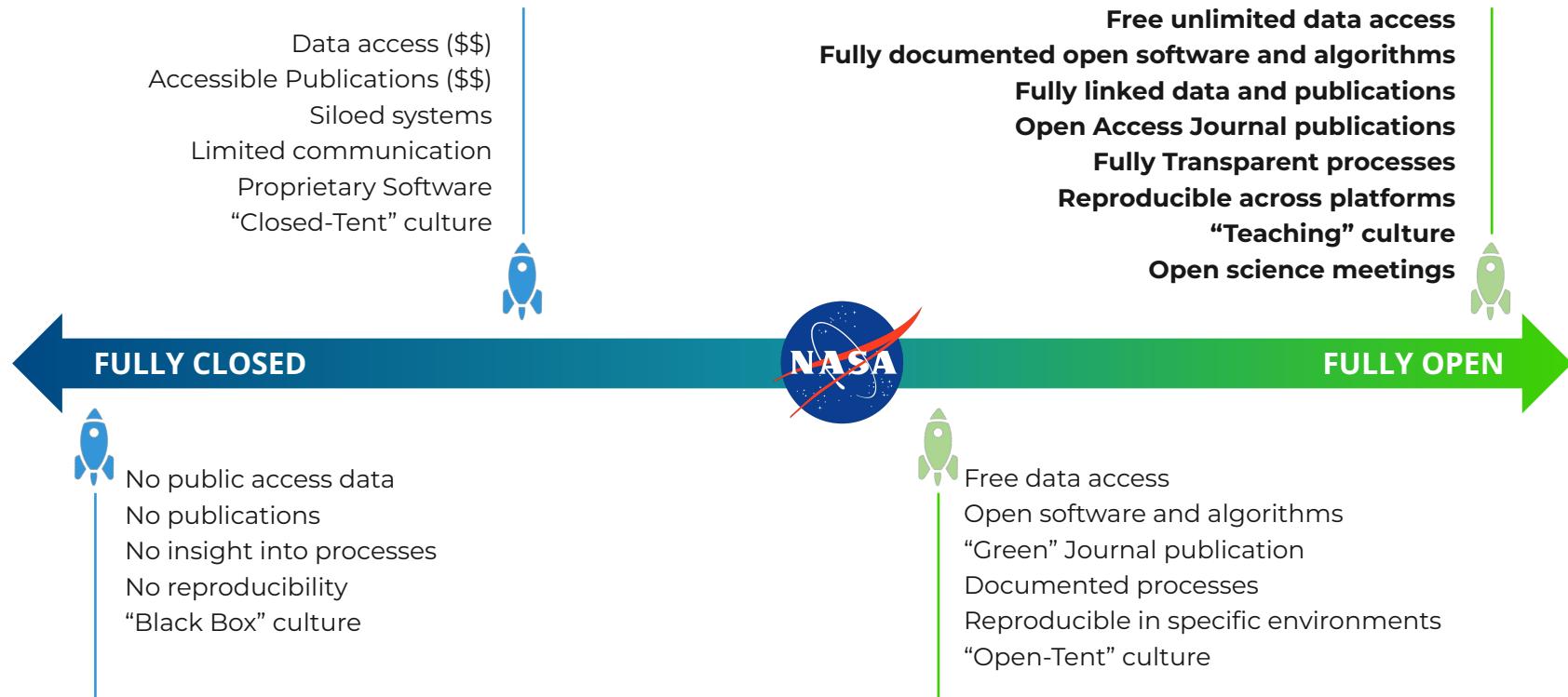


Figure 4. From data to information, knowledge and wisdom. Adapted from DIKW Model for knowledge management and data value extraction.

Strategic Pathway Elements	Spatial Data Infrastructures	Future Geospatial Information Ecosystem
Standards Governance and Policy	A commitment to assess, establish, and maintain a common standards framework	A commitment to adopt standard models for knowledge representation
Technology and Data Interoperability	Voluntary standards for data and technology	Anticipatory regulations, FAIR data and human-centered standards for knowledge creation
Compliance Testing and Certification	Regular assessments, training, government mandates, performance metrics, testing and certification	Regulatory sandboxes, proof of concepts and incentives for deploying and scaling 4IR technologies
Community of Practice	Sharing and leveraging data and technology standards	Inclusive and participatory roles in data vocabularies, digital twins, and other 4IR standards development

Figure 12. Standards Strategic Pathway – the Step Change

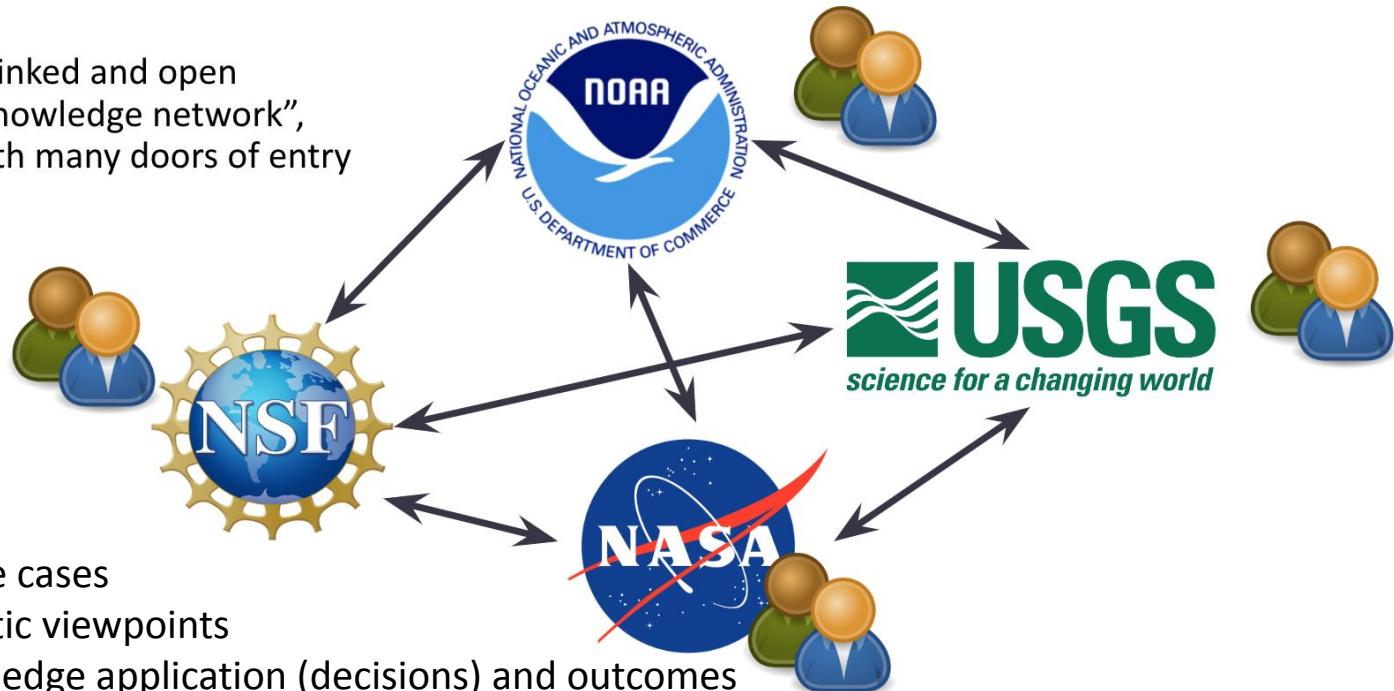
# What is “Open Science”?



Source: [C. Gentemann \(2022, NOAA EDM Workshop presentation\)](#)

# Toward an Open Ecosystem of Trusted Open Ecosystems

A linked and open  
“knowledge network”,  
with many doors of entry



More coverage in use cases

Providing more holistic viewpoints

Ability to track knowledge application (decisions) and outcomes



# What success means to constituents

## **Meg, a Coastal Resident and Environmental Enthusiast:**

Meg is a resident living on Hilton Head Island with a deep interest in the environment. She talks to the system through her Alexa to gain insights into local marine life, weather patterns, and pollution levels using real-time weather updates, satellite imagery from NASA, and environmental data from NOAA. She can report environmental concerns directly to the EPA with full provenance for investigation.

## **Bart, a Commercial Fisherman:**

Bart relies on accurate and timely information for successful operations in the Gulf. He uses an integrated ship dashboard to make predictions and get updates that help him make informed decisions about optimal fishing locations, avoiding adverse weather conditions, and ensuring sustainable practices. The system enables Bart by combining NOAA's fisheries data, NASA's satellite-based ocean temperature maps, and USGS coastal mapping. Bart sometimes asks the system to explain predictions and give alternative assessments. When needed, Bart further uses the system for direct communication with relevant federal agencies for licensing, regulatory updates, and emergency assistance.

## **Gertie, an Emergency Response Coordinator:**

Gertie requires immediate access to comprehensive information while monitoring wildfires in Colorado. A multi-modal system interface connects them to real-time weather updates, NASA's disaster monitoring, and USGS geological data. Gertie is enabled to coordinate evacuation plans with local and regional authorities and monitor emerging outcomes in real-time.

## **Holden, a Tourism and Recreation Planner:**

Holden aims to create enjoyable experiences for visitors to the Chesapeake Bay. As Holden crafts tailored experiences for his clients, he converses with the system to understand seasonal trends, identify ideal locations, and, shape events accordingly.

## **Elias, a Public Health Researcher:**

Elias is a public health researcher studying coastal communities and relies on diverse data sources. He uses a portal to access and fuse information from health, environmental quality, and satellite record databases to help identify potential health risks and patterns based on air and water quality. He uses context to understand ideal methods of engagement with local communities to provide evidence-based recommendations for improved public health outcomes.



# What success means to the workforce

## **Buffy, a NOAA Archivist:**

Buffy needs to efficiently manage and organize a vast number of heterogenous datasets in the Archive. Through her government portal Buffy accesses and catalogs data. She uses the marketplace to add quality matrix checking, link related datasets, or create new collections and views on the data. She develops, shares, and evolves standard templates and expressions for providers to use and assesses adoption.

## **Munir, a NOAA Data Manager:**

Munir's job is to coordinate marine data flows between providers and consumers. He uses the system API to help build, monitor, and communicate data pipelines, while ensuring data quality. He uses the API in his own software to achieve on-demand data retrieval in specific formats for generating comprehensive data reports. He contributes his techniques back to the marketplace as reusable templates for use by other data managers.

## **Marilyn, a NOAA Data Scientist:**

Marilyn uses the system daily to access diverse datasets for research and analysis to relate societal indicators to climate change. The system provides here facilitates seamless integration of datasets from NOAA, NASA, USGS, EPA, and NIH. She leverages user and system access data to quickly develop and test new models for improving user experience; the system automatically shares what she creates for reuse by others.

## **Aditi, a NOAA Physical Scientist:**

Aditi builds authoritative climate data records at NOAA. She leverages the system to explore what has been done before, what findings have been made, and what things might need further research. She leverages on-demand data retrieval for existing products in desired formats to try new modeling and simulation techniques. The data and products she produces are made available to the system.

*Not pictured: System Administrators; Data Operators; Policy Analysts; etc.*

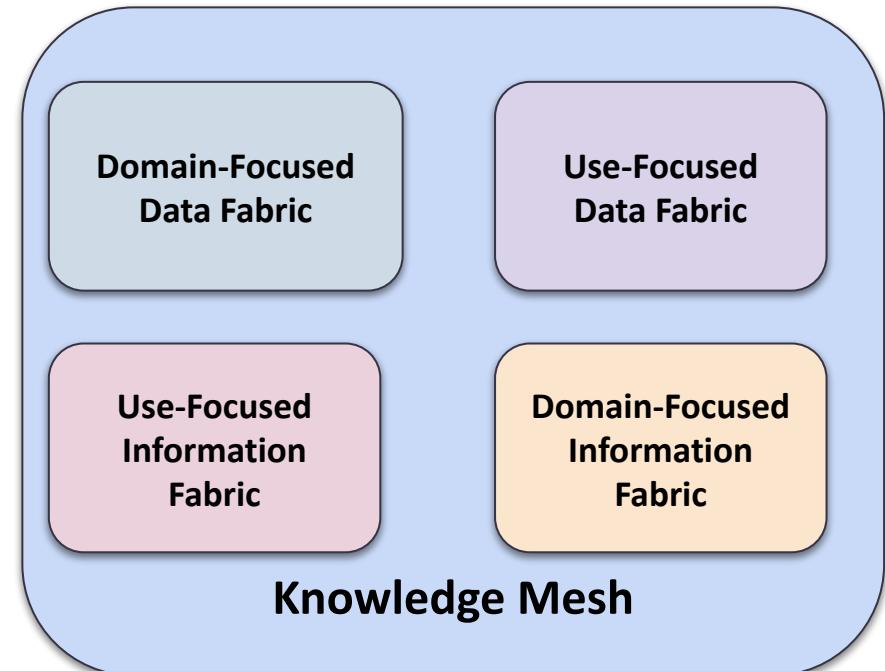


# Terminology - Mesh vs Fabric?

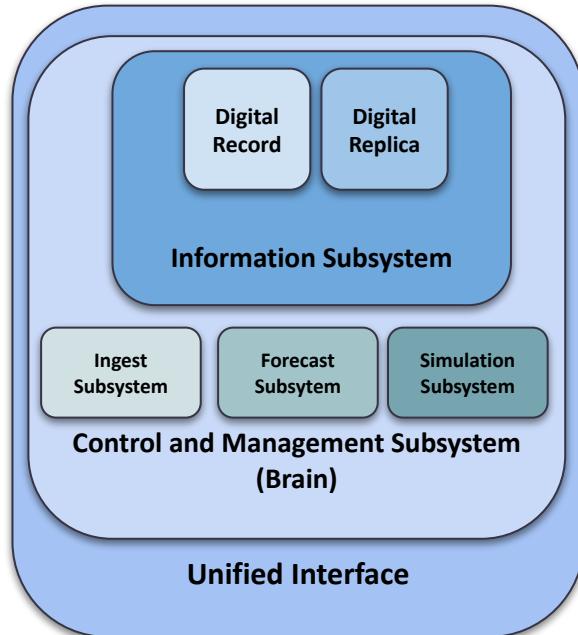
It's fuzzy, but very basically,

- **Fabric** - centralized governance, focused community, focused access patterns
- **Mesh** - decentralized control, community of communities, integrated access patterns

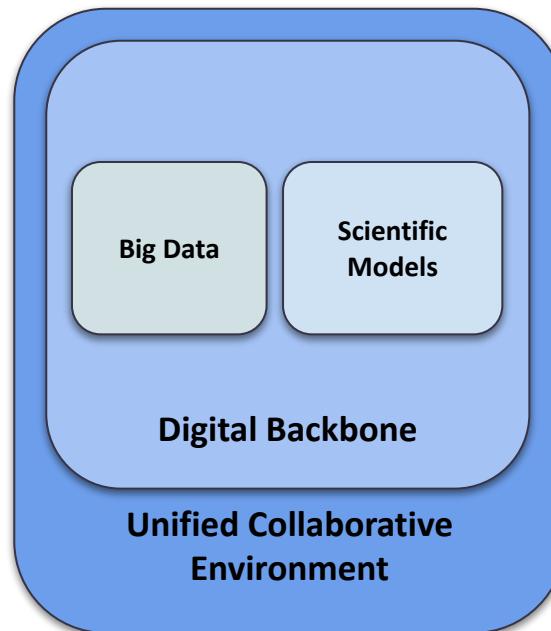
We need capabilities of both, and are seeing this need emerge in next-gen motifs.



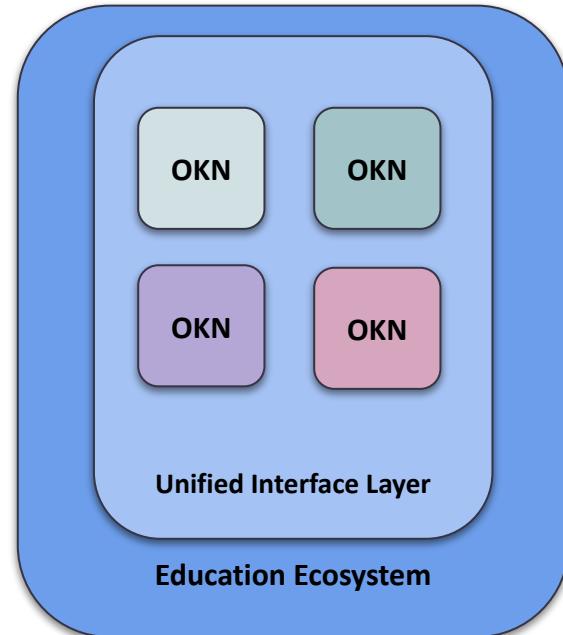
# Emerging Motifs in the Next-Gen Landscape



Earth System Digital Twin  
(ESDT)



European Digital Twins of the Ocean Ecosystem  
(DTO)

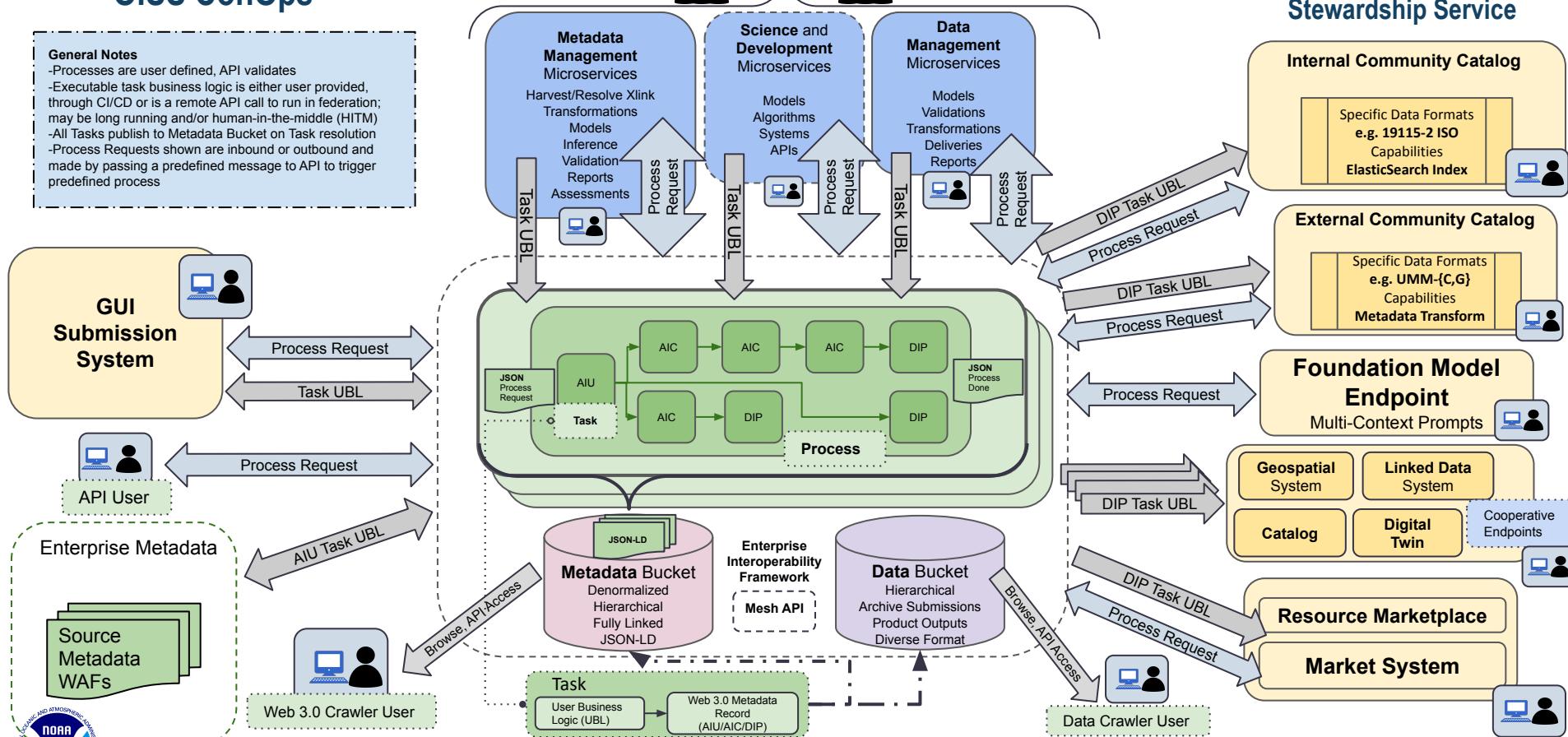


Prototype Open Knowledge Network  
(Proto-OKN)

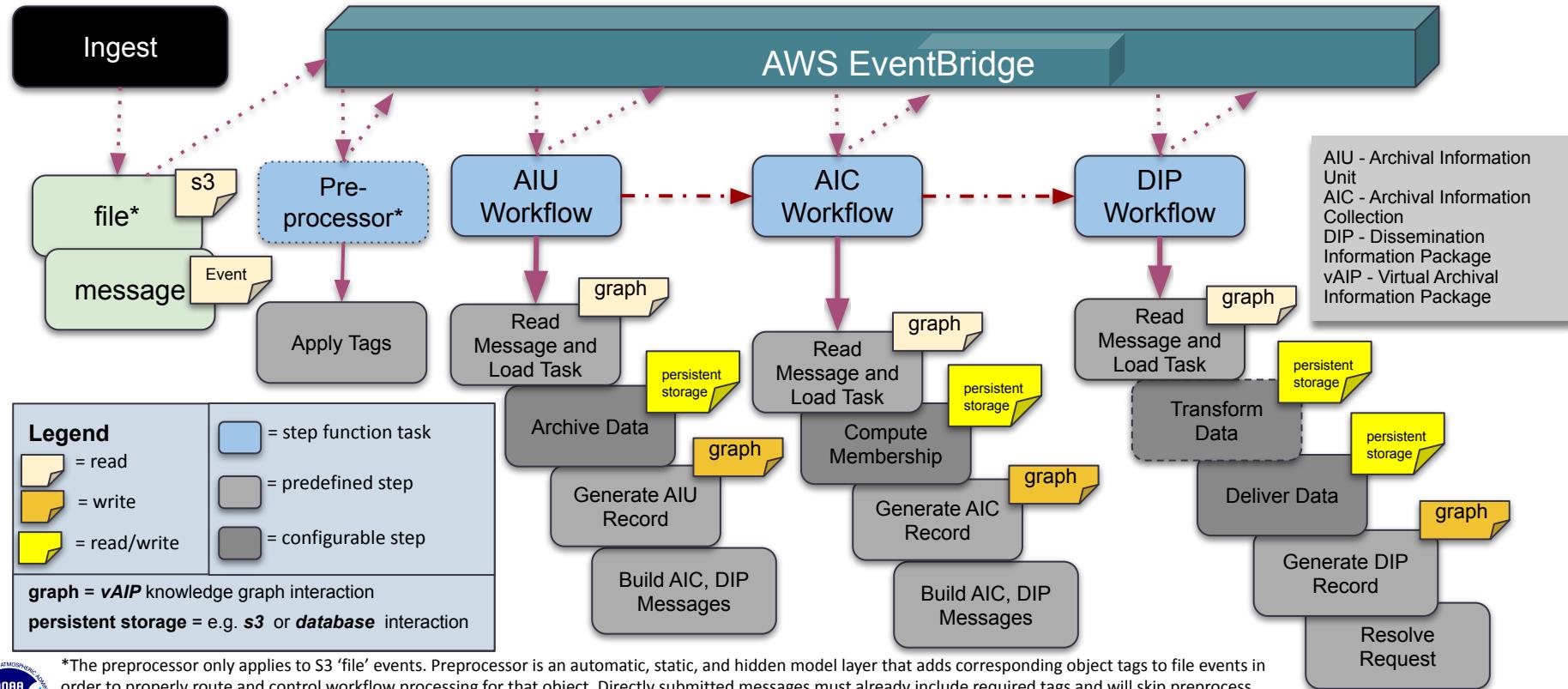


# OISS ConOps

## **Data Managers, Data Scientists, Physical Scientists, Operators, Researchers**

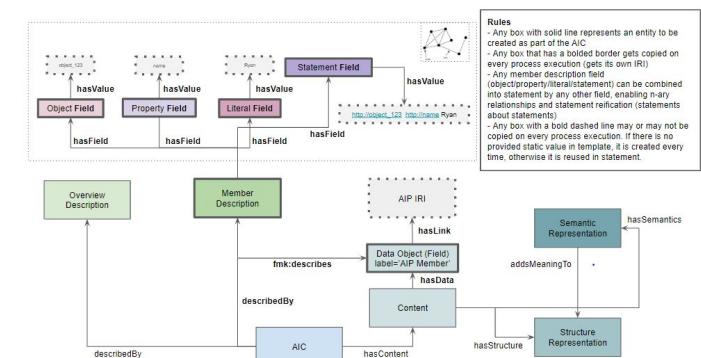
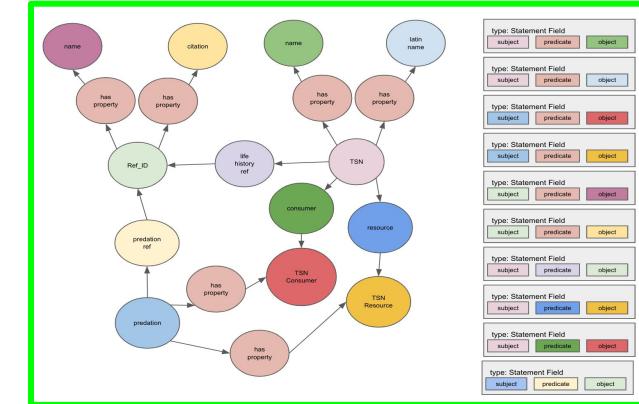
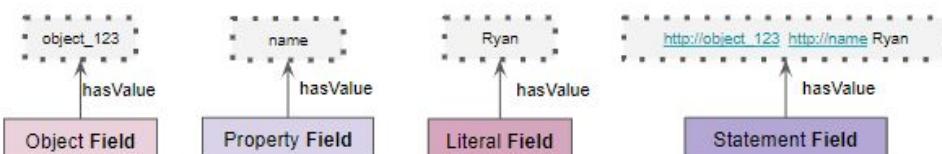
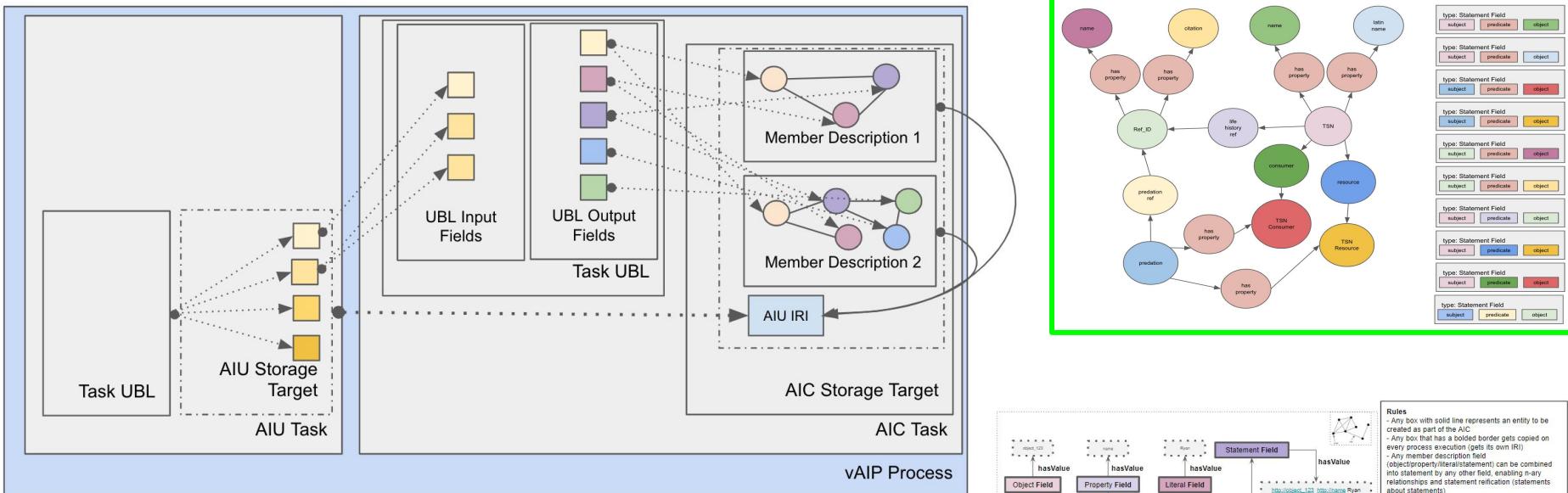


# Massive Real-time Concurrency, Small Footprint

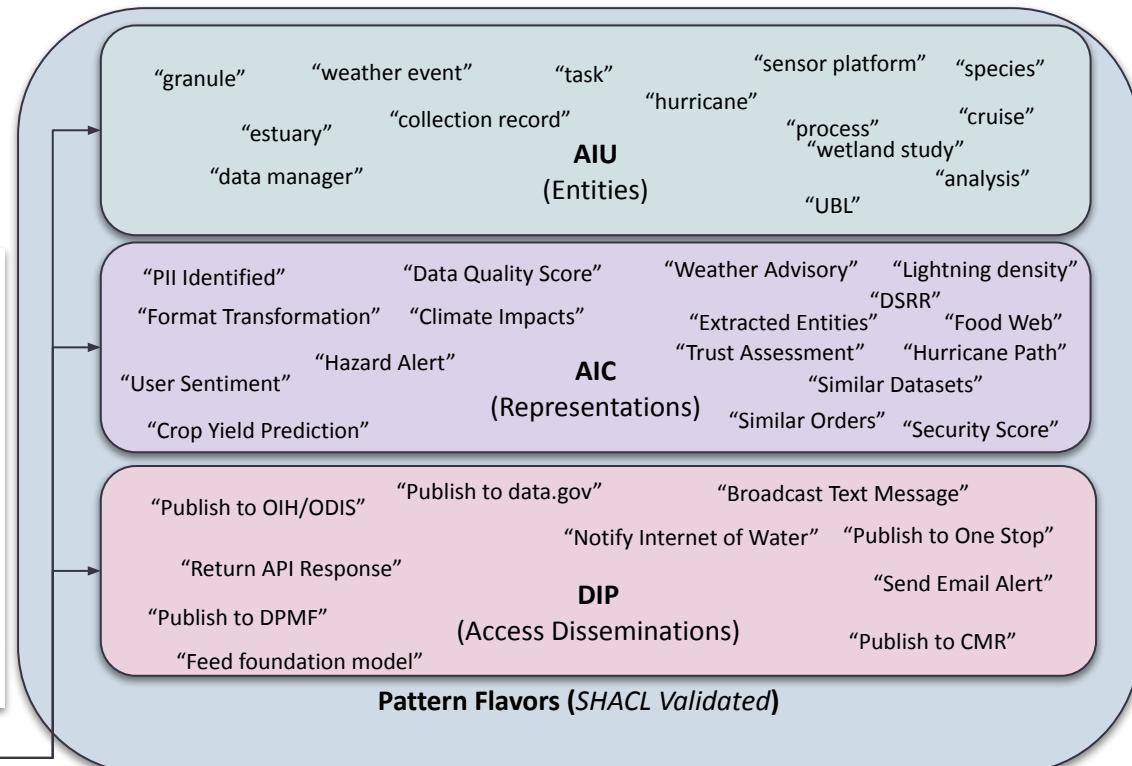
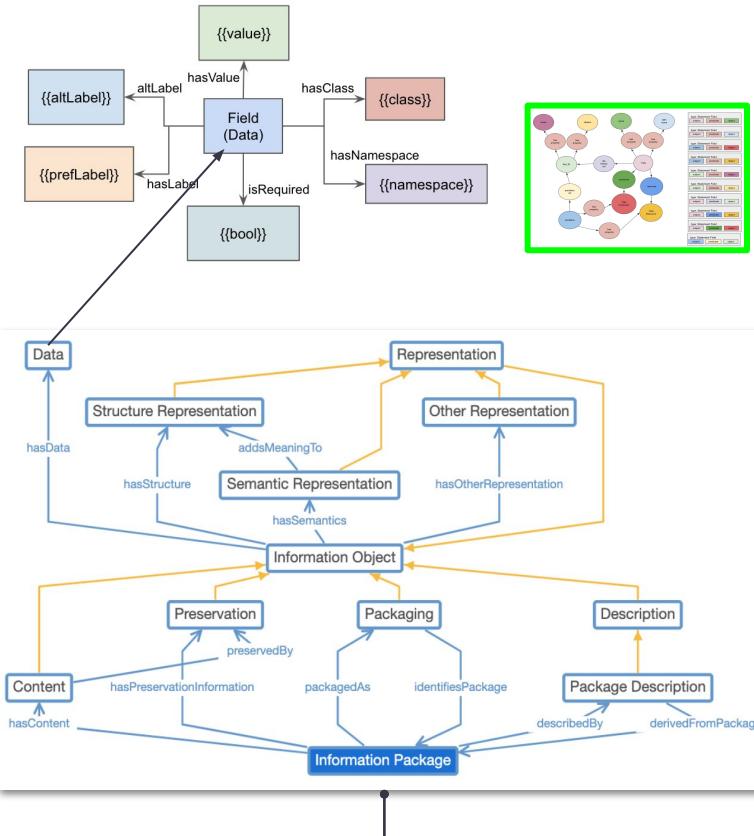


# Enormous Amounts of Trusted, High Quality, Fully Contextual, and Regularized Data

Links; texts; documents; input/output parameters for functions; video; audio; biological; ecological; etc.



# OISS Motifs



# Motif Search Spaces

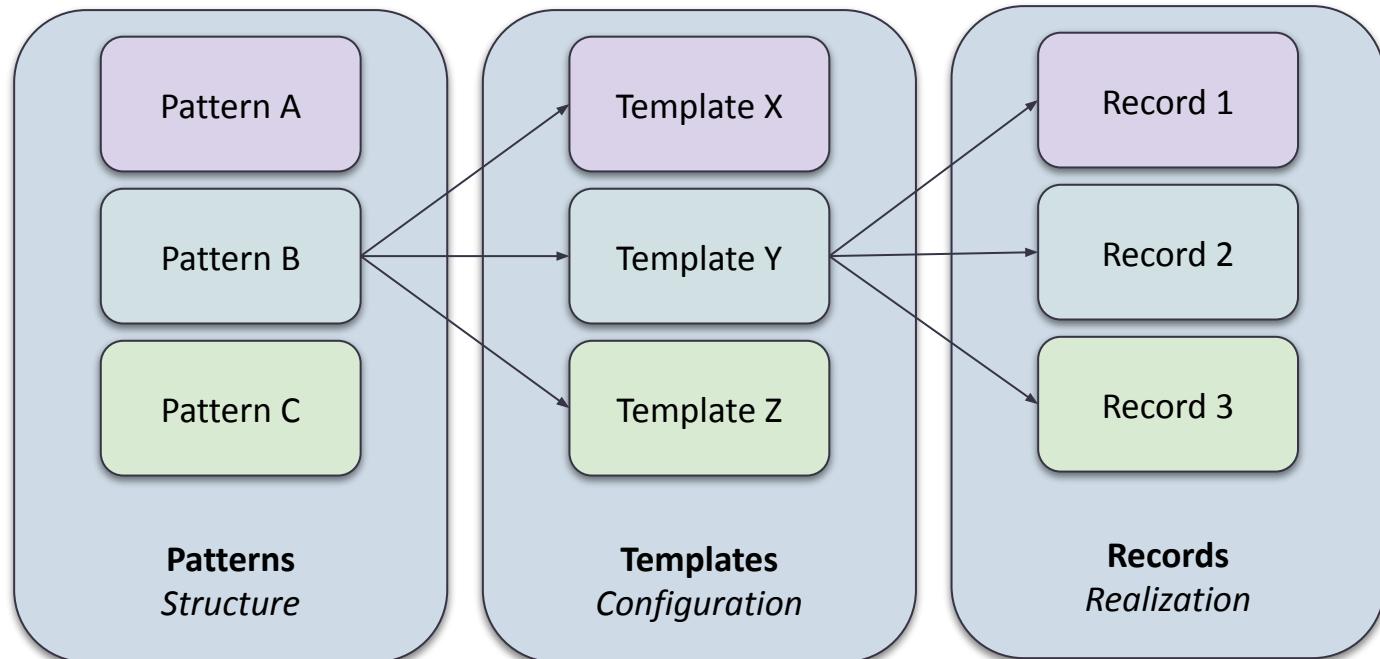
Motifs (subgraphs) are enablers for:

- Stable API interfaces
- Optimized searches
- Feature models
- Visualizations

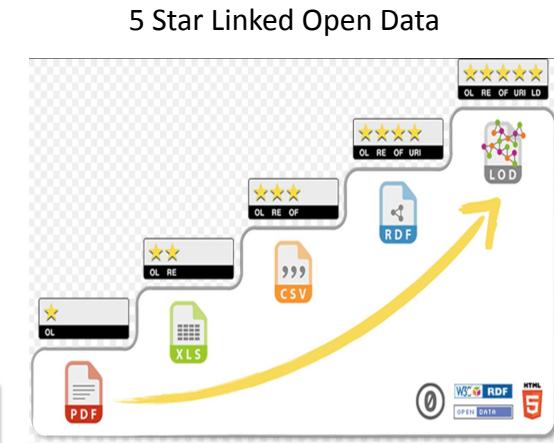
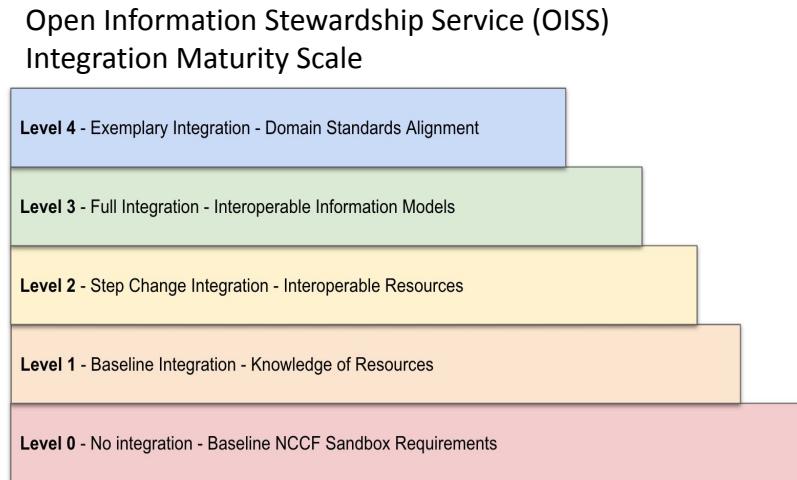
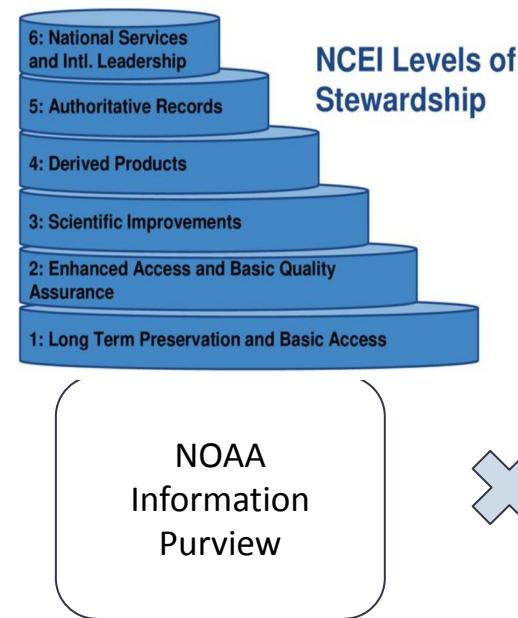
All OISS **explicit motifs** are **OAIS Information Packages** Reference model defined user patterns.

Each explicit motif has different SHACL characteristics - for example, **AICs support n-ary relationships** in their member descriptions.

These are the motifs that are defined and known - there are other motifs emergent in the system data due to its linked data nature



# Enabling Iterative Transformation at NOAA

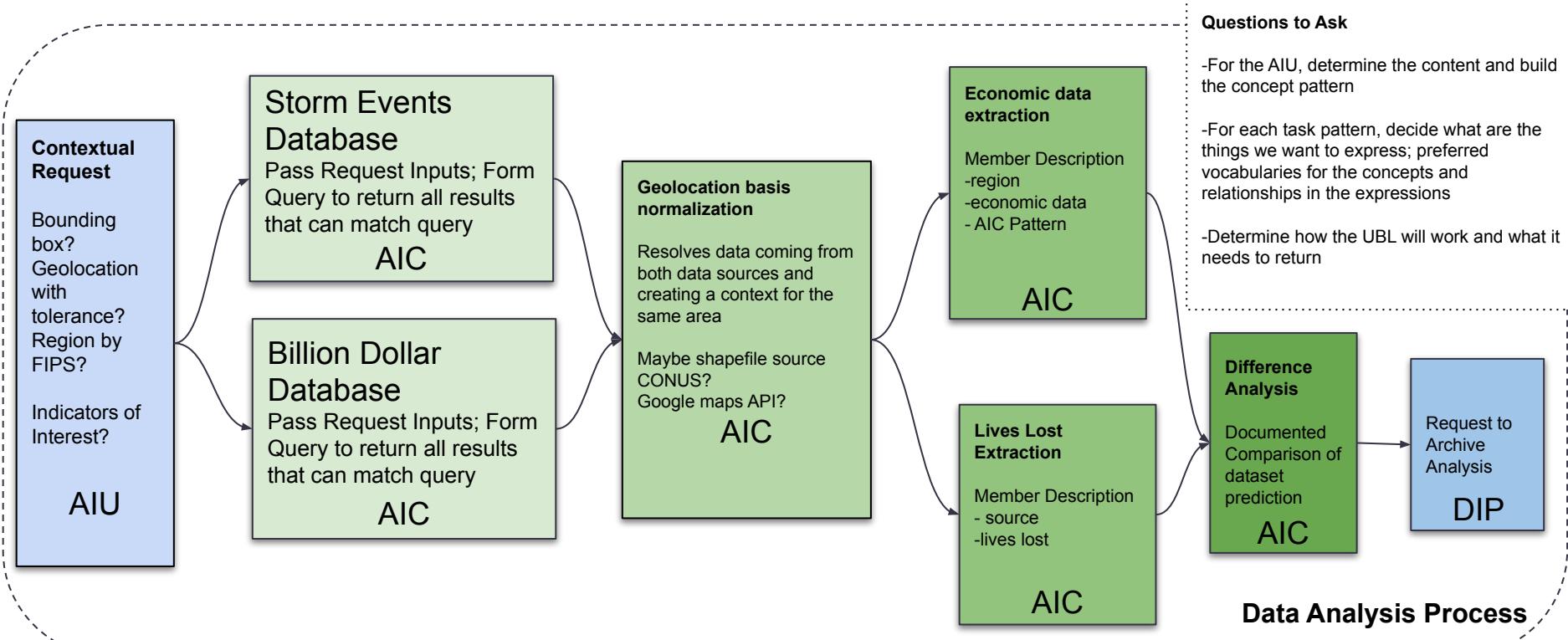


## Recursive Stepwise Integration

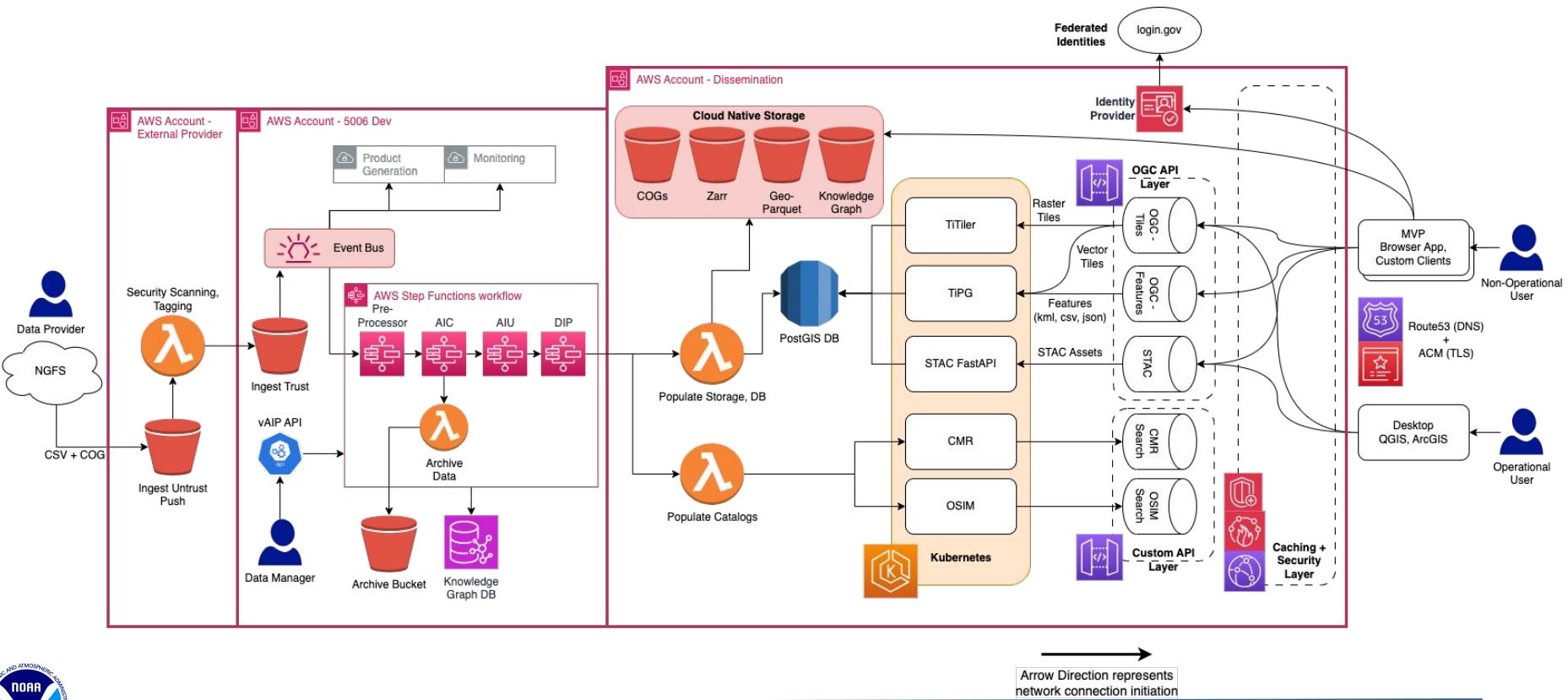


## Iterative Increases to Openness

# OAR-NCEI Societal Data Partnership

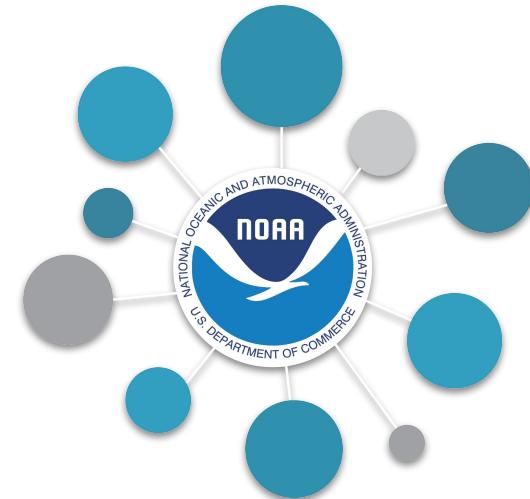


## Multi-transform and Multi-access for Wildland Fire Storefront



# Current Activities

- **Seeding the mesh** with early adopter processes and resources
  - Lots of data streams and system integrations being onboarded or planned - Nexrad, CCOR, OISST, Faraday Cup, Extended Continental Shelf, Passive Acoustic Data, IGRA, World Ocean Database, etc.
- Developing and delivering **educational materials** for mesh adoption and maturation
  - Jupyter notebooks; trainings; 'teaching to fish' with the API and using the JSON-LD files and various data formats that are produced
- Identifying **beneficial partnerships and integration pathways**
  - E.g. across NOAA LOs; with NASA AIST and CNES in Digital Twin contexts; with NSF in OKN contexts; with OIH/ODIS in specific Ocean datastream contexts
- Developing **access processes** using **advanced AI/ML**
  - E.g. use of Retrieval Augmented Generation (RAG) to build out **conversational discovery** and access Processes; automated metadata normalization
- Building out the **marketplace**
  - Working with users to build out system resources for sharing



# Backups



# Mesh Highlight Summary

- Provides a full capability **metadata layer as an enterprise orchestration service**
  - Supports all **Digital Twin Earth Framework requirements** as specified by the DTE-FS
  - Maps as an AIST ESDT Architecture
  - Satisfies NARA requirements, intrinsic and largely automated support for PARR 2.0, FAIR, CARE, TRUST
  - As **cheap** as possible - cost optimized at **full capability**
  - As **flexible** as possible - any process, entity, expression, or access pattern; any user code in any compute context; agile reference architecture for re-implementation
    - Allows evolutionary growth in all exposed aspects
  - As **simple** as possible - no message adapters, pip install, small purposeful API
- Emphasizes **facts and traceability**
  - Everything is data; process is data; full provenance at the task level published in real-time on every process event
- Enables **definition and contextualization of anything** (entities, transformations, disseminations)
  - Inbound and outbound processes are the same; capturing use information; semantic records of function call inputs and outputs for model training; only running on request; API definition of everything
  - Provides a Motif basis for everything in larger connected context - **Motifs are critical** for enabling analyses
- Enables easy and automatable **intra-agency** and **inter-agency** interoperability
  - Process lets us connect with (and trace) federated APIs; supports interoperability with legal and regulatory requirements; private data
  - Open participation
- Supports **exploitation by Next-Gen capabilities** - multimodal foundation models, virtual reality environments, 'Spatial Computing'
- Guards against future migrations (**assumes eventual sunset up front**) - everything is data, no 'code-lock'



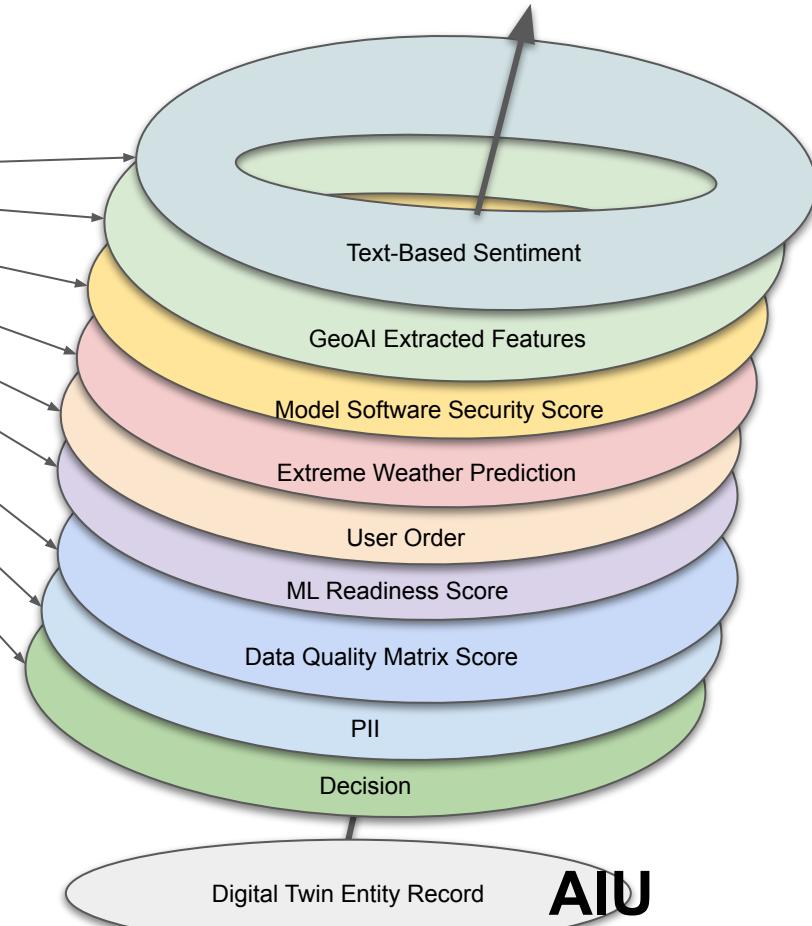
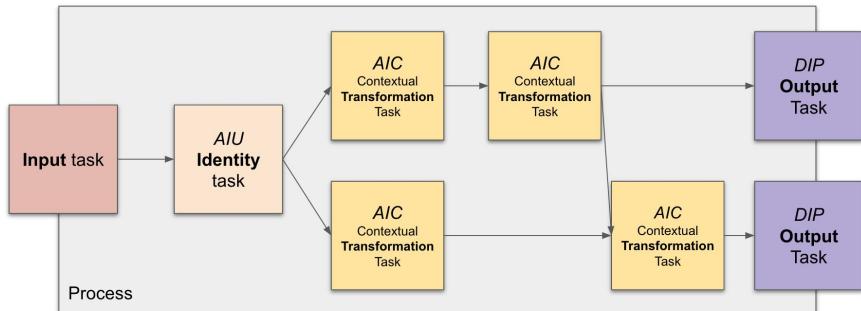
# Linked Independent Information Models

**DIP**

- Structured KOS
- FAIR
  - Findable
  - Accessible
  - Interoperable
  - Reusable
- CARE
  - Fine-grain access rights
  - Legal Interoperability
- **Knowledge Graph** held
- Independent spaces
- API Discovery/Document Browsing

**AIC**

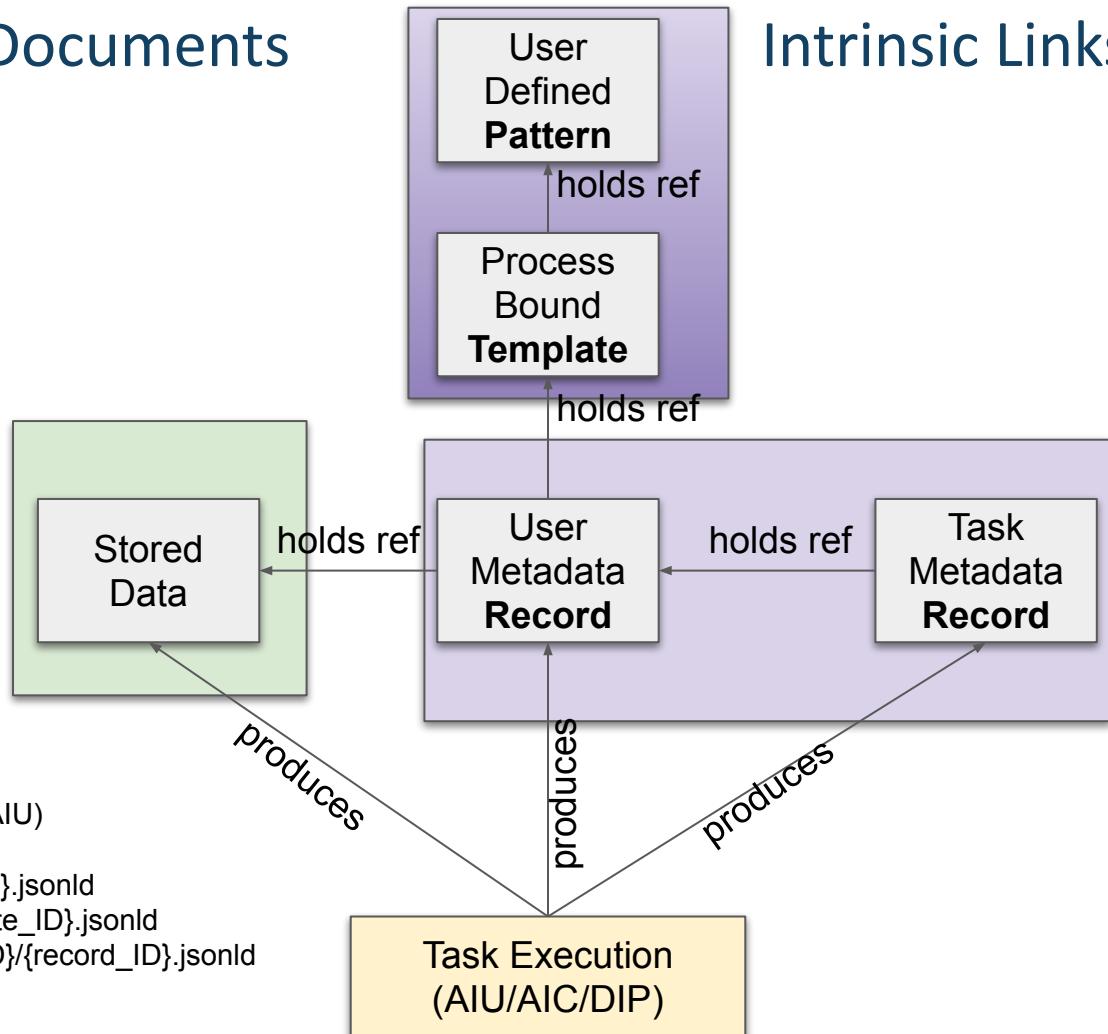
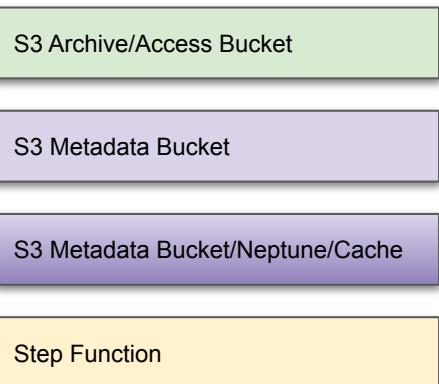
The **AIC** provides controlled access to the 'full graph' of semantically encoded data, information, and knowledge, allowing users authorized by associated Access Rights policy to define AICs that implement Patterns using one or more of any preexisting, well-defined ontologies available on the semantic web (e.g. [SSN](#), [SOSO](#), [Schema.org](#), [SensorML](#), [Dublin Core](#), [GCMD](#), etc.), and/or make use of newly minted, user defined ontologies that are constructed and published automatically on AIC Pattern deployment for system-wide discovery and reuse.



**AIU**

# AIU/AIC/DIP Metadata Documents

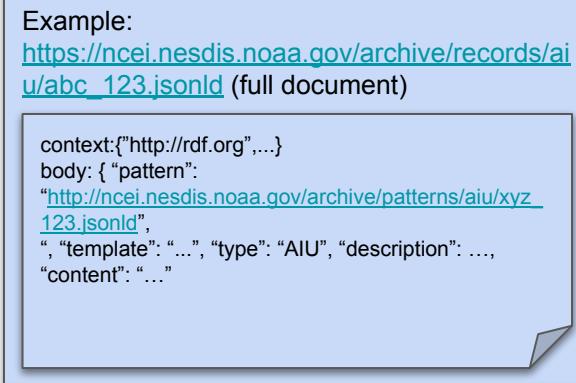
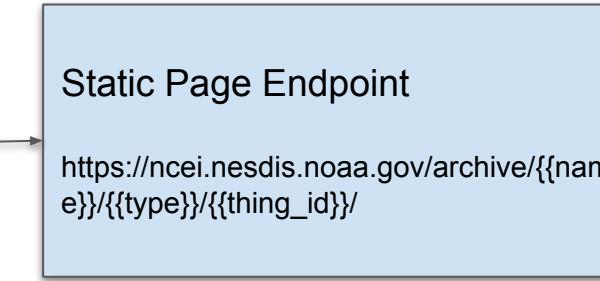
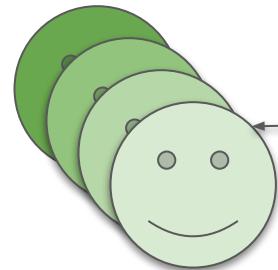
## Intrinsic Links



Metadata Bucket Hierarchy (example layout for AIU)

`s3://patterns/storage/aiu/{pattern_ID}/{pattern_ID}.jsonld`  
`s3://templates/storage/aiu/{template_ID}/{template_ID}.jsonld`  
`s3://records/storage/aiu/{template_ID}/{record_ID}/{record_ID}.jsonld`

# Direct, Immediate Metadata Access



## S3 Metadata Bucket

### Namespace = Core

#### Reference Model Classes

Small and immutable-per-reference-model surface. Recursive classes provide ability to compose inherently interoperable structures

### Namespace = Patterns

#### Unvalued Schema Individuals

Potential for ontologically rich augmentation via overloaded labeling and tagging

### Namespace = Templates

#### Partially Valued Schema Individuals

Serves as a fast-query inference layer at data-stream resolution of descriptions and representation networks

### Namespace = Records

#### Fully Valued Schema Individuals

Mesh layer, high write velocity, fully featured inferencing, rich membership content

# Creating a Pattern

```
pattern = ontology.IdentityStoragePattern(session, "Granule")
```

Create a new pattern (here, AIU)

```
reference = pattern.add_reference("Archive PID", field={'title': 'Value', 'is_link': False})  
semantic = reference.add_semantic_representation("Description")  
structure = reference.add_structure_representation("Format")  
semantic.node.addsMeaningTo.append(structure.node)
```

Add AIP-required information objects^

- Content
- Descriptive Information
- Packaging
- PDI
- Reference
- Fixity
- Access Rights
- Context\*
- Provenance\*

```
checksum = pattern.add_fixity("Checksum", field={'title': 'Value', 'is_link': False})  
checksum.add_extra_representation("Purpose")  
checksum_semantic = checksum.add_semantic_representation("Definition")  
checksum_structure = checksum.add_structure_representation("Format")  
checksum_semantic.node.addsMeaningTo.append(checksum_structure.node)
```

```
file_content = pattern.set_content("File", field={'is_link': True})  
file_semantic = file_content.add_semantic_representation("File Schema")  
file_structure = file_content.add_structure_representation("File Format")  
file_semantic.node.addsMeaningTo.append(file_structure.node)  
file_content.add_extra_representation("Decoding Software")  
file_content.add_extra_representation("Visualization Software")
```

Add representation IOs to each IO

- structure
- semantic
- 'extra' (other)

```
desc = pattern.add_description("Product Description")  
desc.add_semantic_representation("Product Description Semantic Rep")  
desc.add_structure_representation("Product Description Structure Rep")
```

```
ar1 = pattern.add_access_rights("Distribution License")  
ar1.add_semantic_representation("Distribution License Semantic Rep")  
ar1.add_structure_representation("Distribution License Structure Rep")
```

Notes - Can add any number of Fields (DOs) per IO  
Fields are semantic (classed and namespaced) links or values

\*Automatically satisfied, but user can extend

^Can add greater or equal to the Information Package Subtype (AIU/AIC/DIP) Requirements



# Creating a Process (Pattern to Workflow)

```
process = ProcessTemplate(session, purpose="E2E Granule Process", utility="Test process template")
process.namespace = process.namespace

## Initialize the IdentityStorageTemplate using a premade pattern
granule_pattern = granule(session)
aiu_template = IdentityStorageTemplate(session, source_pattern=granule_pattern)

task = IdentityTaskTemplate(session, target=aiu_template, purpose="E2E Granule AIU Task", utility="Test task template")
process.set_aiu_task(task)
UBLTemplate(session, namespace=task.namespace)
task.set_UBL(UBL)

# This sets the AWS ARN for the step function executing this task (the aiu-workflow step function)
task.set_packaging_field(packaging="Resource Identifier", field="Value", value="arn:aws:states:us-east-1:560271376700:function:create-uuid-lambda")

## Set the AWS ARN for the UBL step function that gets executed in the AIU Task (archive-data)
## This is required in order for the AIU Workflow to call its UBL
UBLTemplate(session, namespace=task.namespace)
task.set_packaging_field(packaging="Resource Identifier", field="Value", value="arn:aws:states:us-east-1:560271376700:function:archive-file-lambda")

## Now we add all the input fields to the UBL
# The inputs below here are all "hardcoded" at the UBLTemplate Level
archive_file_lambda_arn = UBLTemplate.add_input("Archive File ARN", value="arn:aws:lambda:us-east-1:560271376700:function:create-uuid-lambda", is_required=False, value="arn:aws:lambda:us-east-1:560271376700:function:create-uuid-lambda")
destination_prefix = UBLTemplate.add_input("Destination Prefix", value="copy")
destination_key = UBLTemplate.add_input("Destination Key", is_required=False, value="")
destination_bucket = UBLTemplate.add_input("Destination Bucket", value="nccf-dev-archive-us-east-1-560271376700")
algorithm = UBLTemplate.add_input("Checksum Algorithm", value="MD5")
retention = UBLTemplate.add_input("Retention Time", value=20)
UBLTemplate.add_input("Archive Status", is_required=False, value="")
```

The diagram illustrates the flow of code from a template to a workflow, with annotations explaining each step:

- Create a Process (template)**: Points to the first line of code: `process = ProcessTemplate(session, purpose="E2E Granule Process", utility="Test process template")`
- Add Tasks (AIU/AIC/DIP)**: Points to the line: `aiu_template = IdentityStorageTemplate(session, source_pattern=granule_pattern)`
- For each Task, specify what we're storing (Pattern), and how to produce it (UBL - user business logic)**: Points to the line: `task = IdentityTaskTemplate(session, target=aiu_template, purpose="E2E Granule AIU Task", utility="Test task template")`
- For UBL, specify the 'field map' - what goes in, what comes out**: Points to the line: `task.set_packaging_field(packaging="Resource Identifier", field="Value", value="arn:aws:states:us-east-1:560271376700:function:create-uuid-lambda")`
- For a Task Pattern, optionally specify static values**: Points to the line: `UBLTemplate(session, namespace=task.namespace)`
- Connect Task Internally (UBL key to key matching)**: Points to the line: `task.set_packaging_field(packaging="Resource Identifier", field="Value", value="arn:aws:states:us-east-1:560271376700:function:archive-file-lambda")`
- Connect Tasks externally (Pattern to UBL matching)**: Points to the line: `UBLTemplate(session, namespace=task.namespace)`



# Dynamic Resource Analysis During Creation

```
1 ?xml version="1.0"?>
2 <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
3   xmlns:xs="http://www.w3.org/2001/XMLSchema#"
4   xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
5   xmlns:owl="http://www.w3.org/2002/07/owl#"
6   xmlns:base="https://ncei.noaa.gov/vaip/pattern/storage/aiu/6b587130-829e"
7   xmlns="https://ncei.noaa.gov/vaip/pattern/storage/aiu/6b587130-829e-42"
8   xmlns:core="https://ncei.noaa.gov/vaip/core#"
9   xmlns:core2="http://www.w3.org/2004/02/skos/core#">
10
11
12 <owl:Ontology rdf:about="https://ncei.noaa.gov/vaip/pattern/storage/aiu/6b587130-829e-42">
13
14 <core:ArchivalInformationUnit rdf:about="https://ncei.noaa.gov/vaip/pattern/storage/aiu/6b587130-829e-42">
15   <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#NamedIndividual"/>
16   <core:hasContentInformation rdf:resource="https://ncei.noaa.gov/vaip/pattern/storage/aiu/6b587130-829e-42">
17   <core2:prefLabel rdf:datatype="http://www.w3.org/2001/XMLSchema#string">weather data
18 </core:ArchivalInformationUnit>
19
20 <core:ContentInformationObject rdf:about="https://ncei.noaa.gov/vaip/pattern/storage/aiu/6b587130-829e-42">
21   <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#NamedIndividual"/>
22   <core:hasDataObject rdf:resource="https://ncei.noaa.gov/vaip/pattern/storage/aiu/6b587130-829e-42">
23   <core:hasDataObject rdf:resource="https://ncei.noaa.gov/vaip/pattern/storage/aiu/6b587130-829e-42">
24   <core2:prefLabel rdf:datatype="http://www.w3.org/2001/XMLSchema#string">extreme weather data
25   <core2:altLabel rdf:datatype="http://www.w3.org/2001/XMLSchema#string">forecasts
26   <core2:altLabel rdf:datatype="http://www.w3.org/2001/XMLSchema#string">model forecasts
27   <core2:altLabel rdf:datatype="http://www.w3.org/2001/XMLSchema#string">data
28 </core:ContentInformationObject>
29
30 <core:DigitalObject rdf:about="https://ncei.noaa.gov/vaip/pattern/storage/aiu/6b587130-829e-42">
31   <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#NamedIndividual"/>
32   <core2:prefLabel rdf:datatype="http://www.w3.org/2001/XMLSchema#string">event
33   <core:hasLink rdf:datatype="http://www.w3.org/2001/XMLSchema#string">{{d2dfab}}
34 </core:DigitalObject>
35
36 <core:PhysicalObject rdf:about="https://ncei.noaa.gov/vaip/pattern/storage/aiu/6b587130-829e-42">
37   <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#NamedIndividual"/>
38   <core2:prefLabel rdf:datatype="http://www.w3.org/2001/XMLSchema#string">event
39   <core:hasValue rdf:datatype="http://www.w3.org/2001/XMLSchema#string">{{e0f98}}
40 </core:PhysicalObject>
```

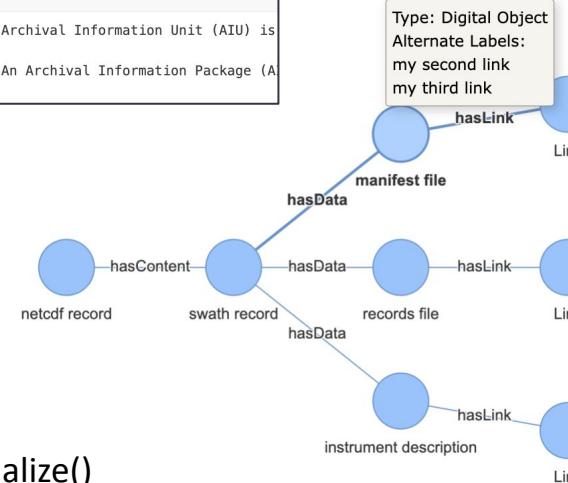
resource.serialize()

```
1 print(granule.example(ancestors=True))
✓ 0.0s
Archival Information Unit(ancestor 0): An example of an AIU might be a granule; a file containing
Archival Information Package(ancestor 1): An example of an archival information package might be
Information Package(ancestor 2): An information package might be a SIP that represents a process
```

resource.example()

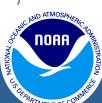
```
1 print(granule.definition(ancestors=True, limit=1))
✓ 0.0s
Archival Information Unit(ancestor 0): An Archival Information Unit (AIU) is
Archival Information Package(ancestor 1): An Archival Information Package (AIP)
```

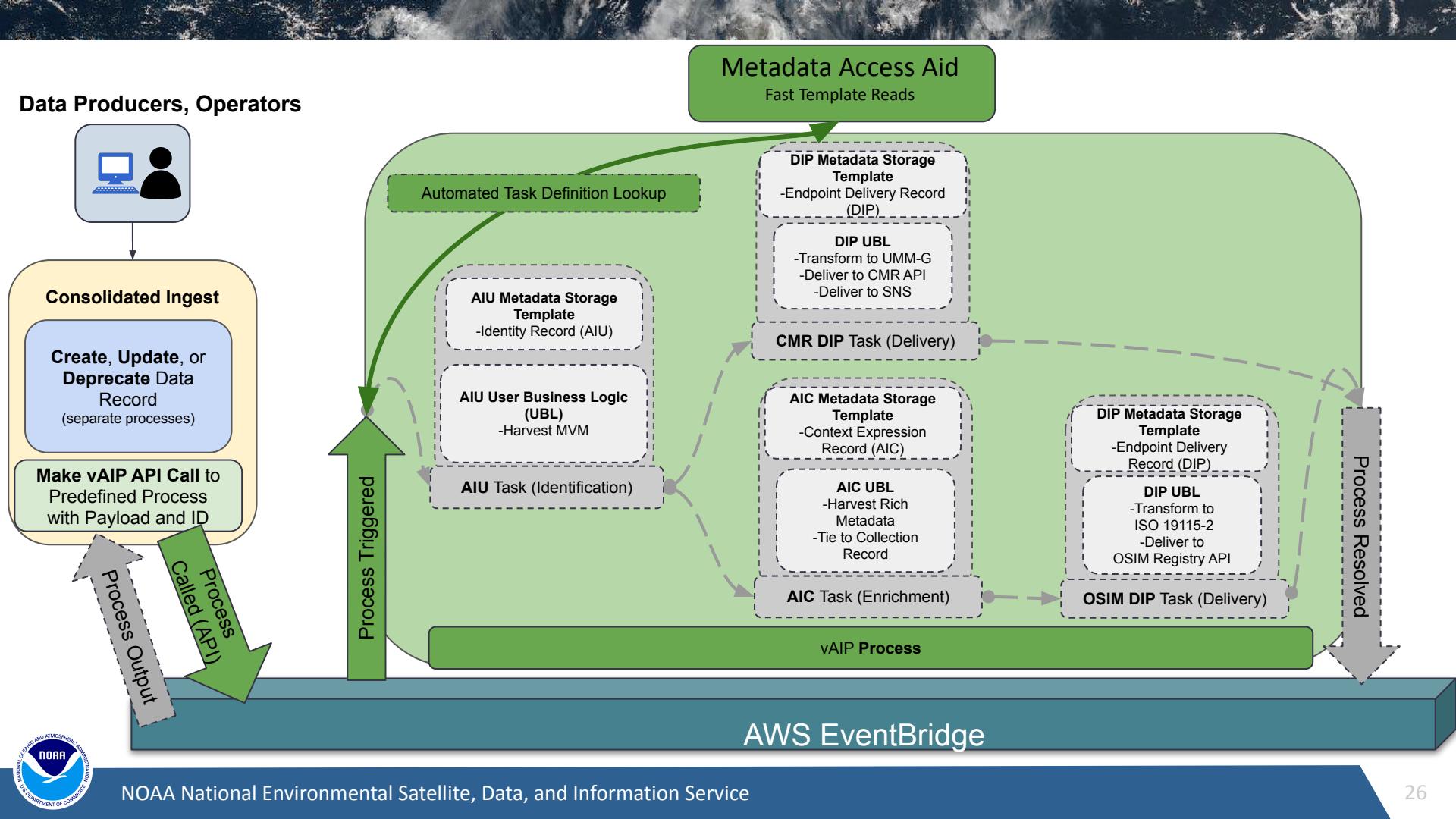
resource.definition()

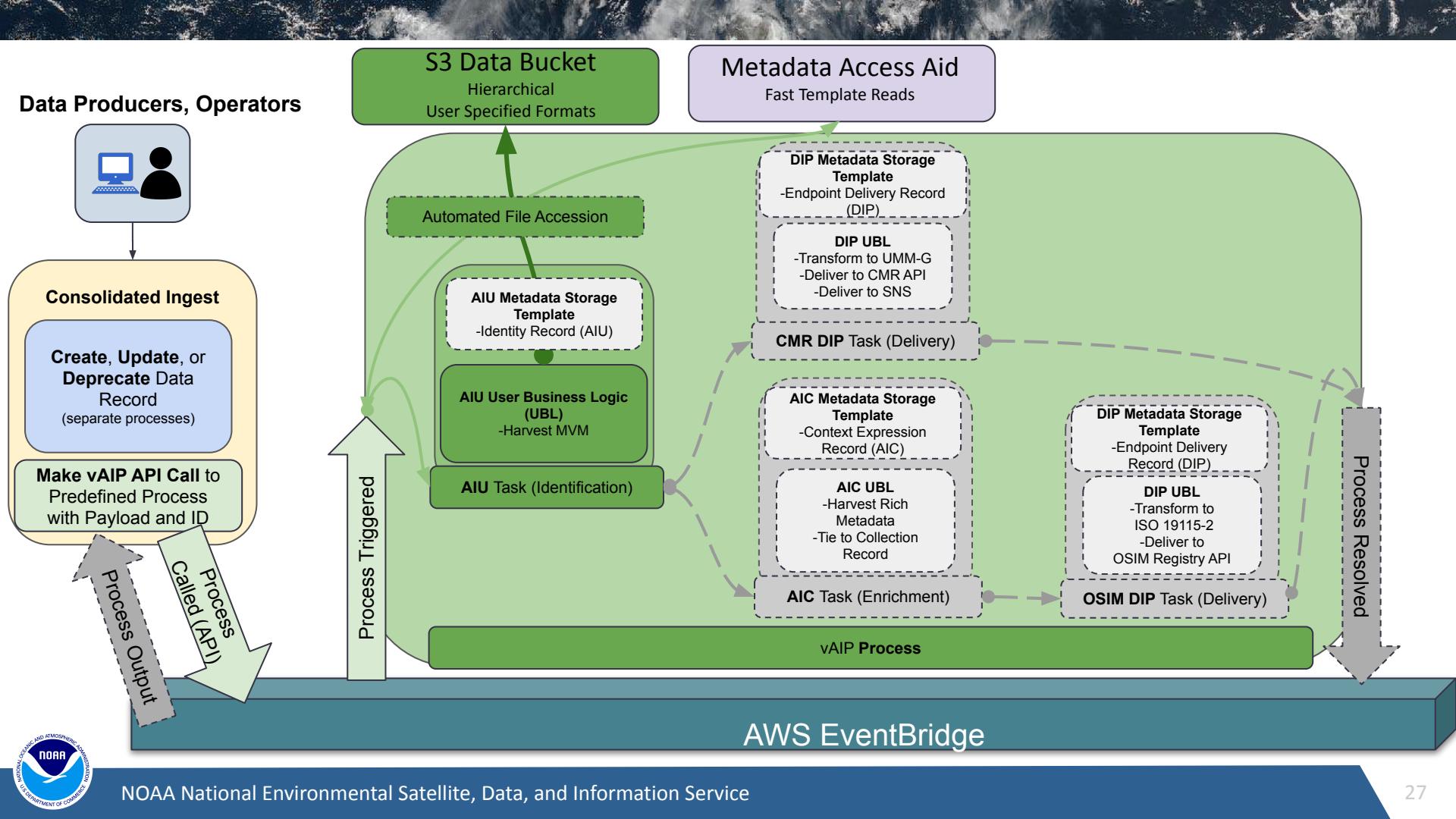


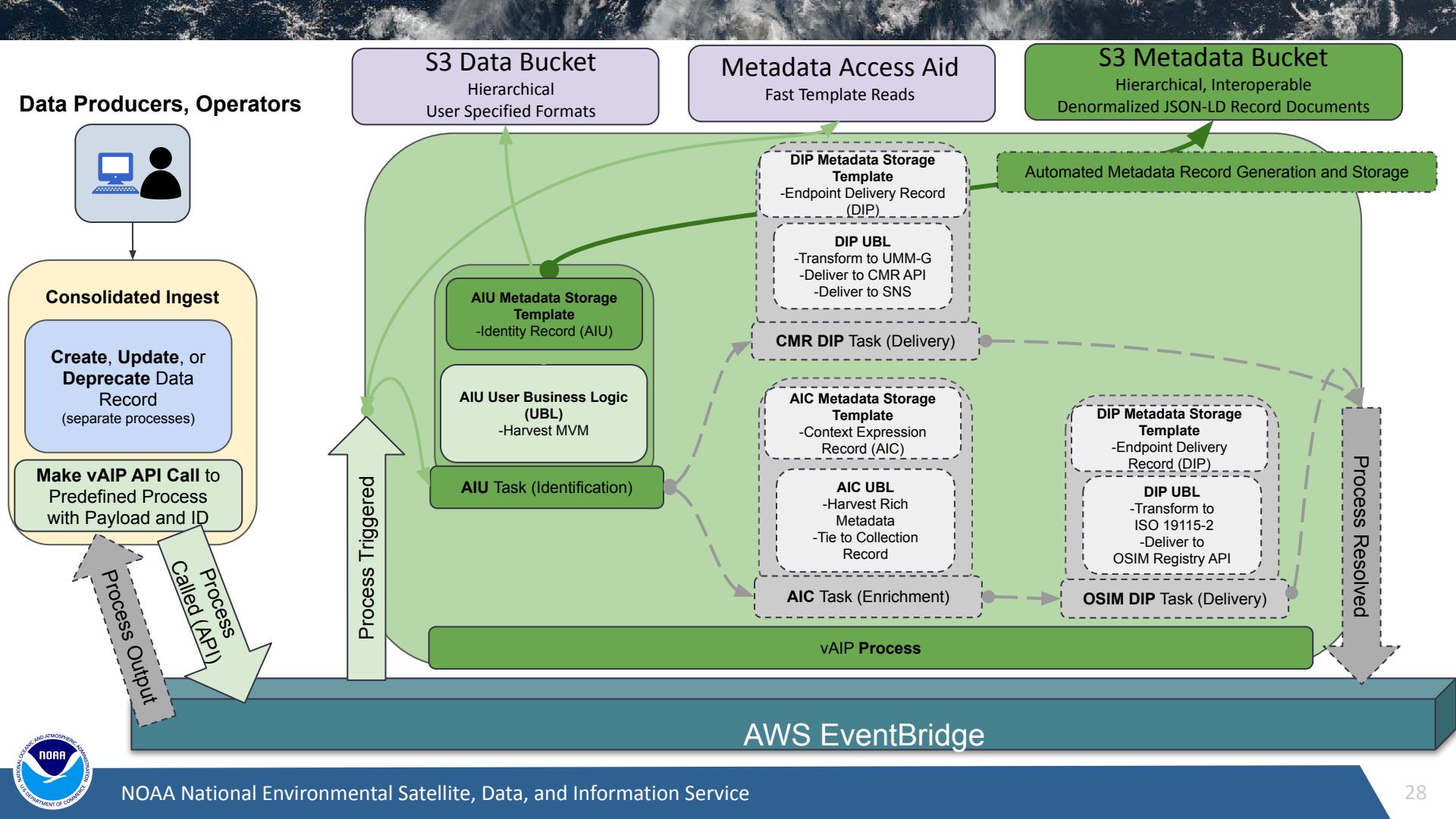
resource.visualize()

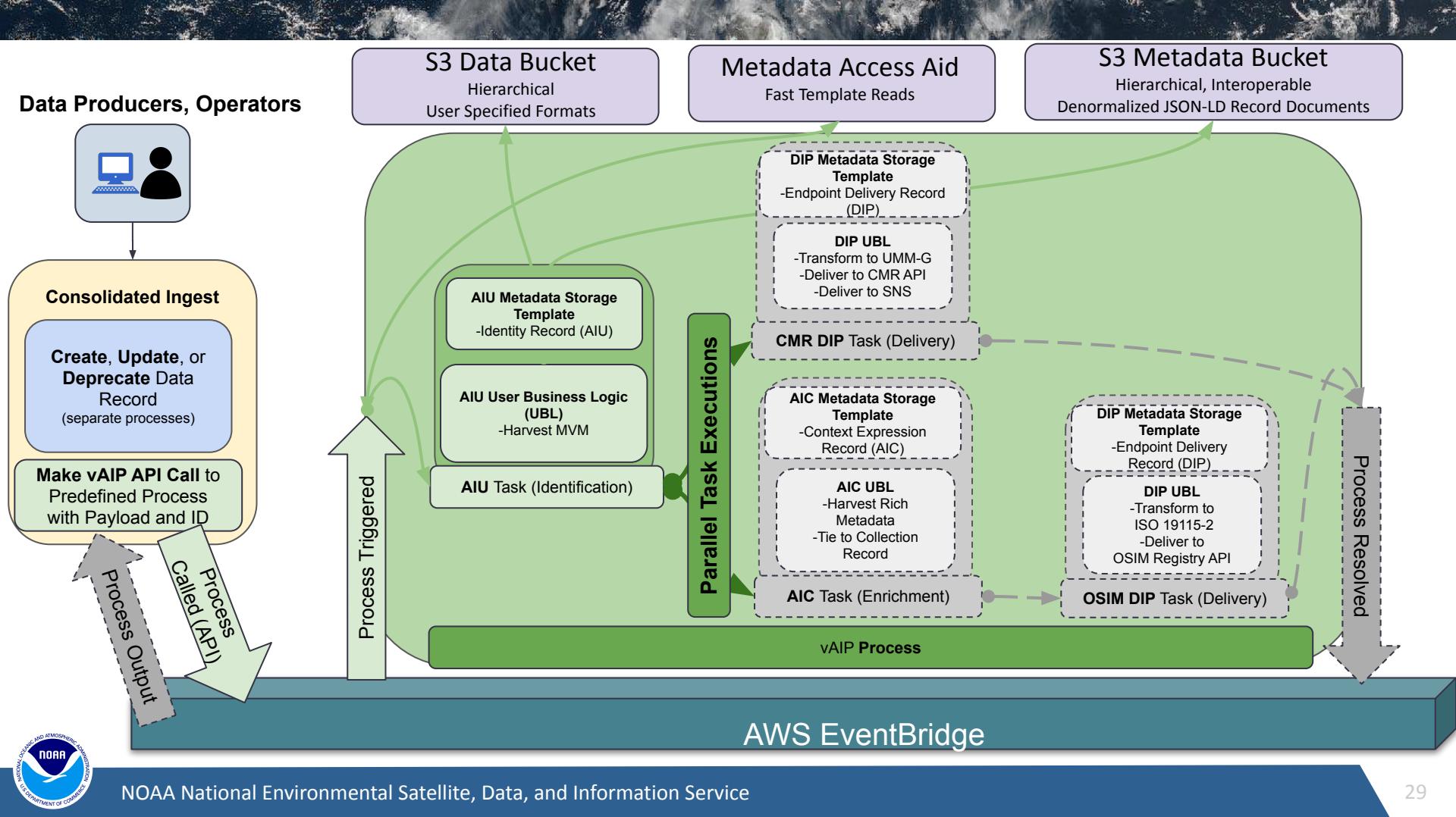
*note: expanded and improved analytic capability is on the backlog as a higher order API library*

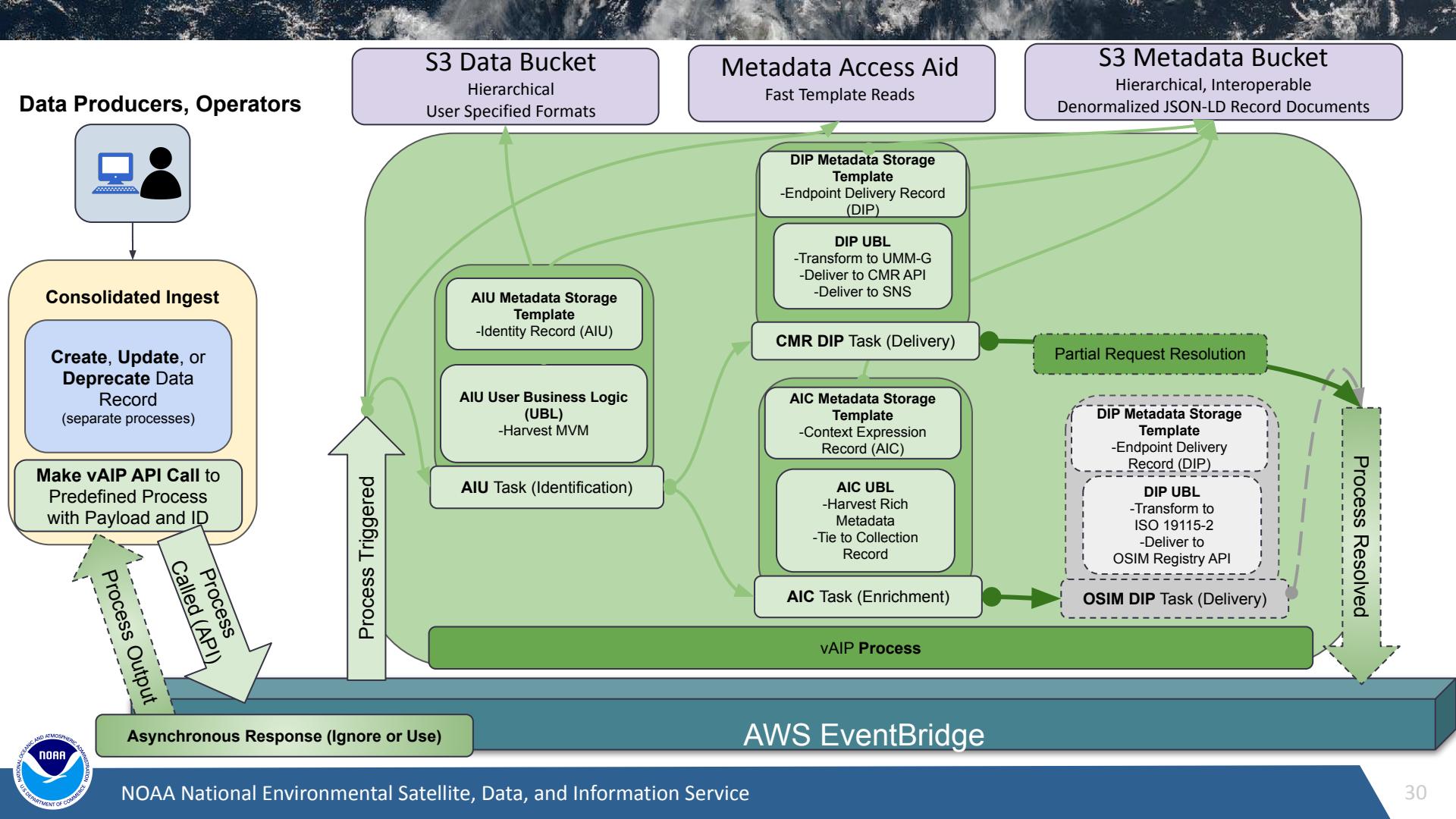


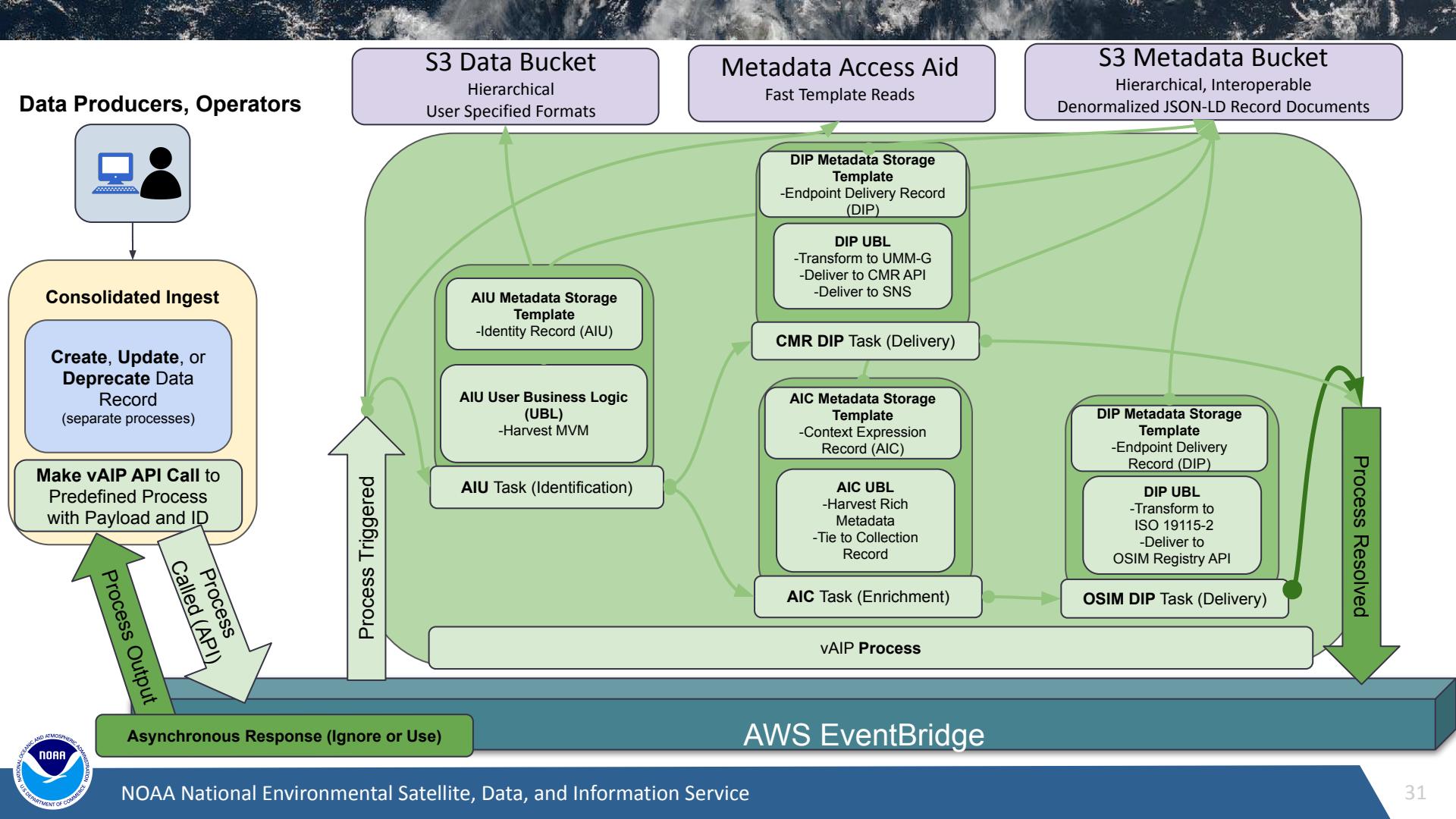












# Common Service - Ordering System

## Access Order Broker - Superset of Order Fulfillment Systems

