Catalog of Self Describing Digital Assets

—

Enterprise NeuroSystems Group
Central Intelligence Platform Team
Date: Jan 24, 2022

Capability

Central Intelligence Group should develop the capability of a catalog of self-describing assets

The catalog provides

- A REST interface for CRUD to a database of SDE (self-describing entries) and a database of opaque blobs
- Each SDE is a JSON record with two entries
  - “asset”: An URI for a digital asset (could be a URI for a blob in the catalog)
  - “metadata”: A metadata description
- Metadata description has following json structure “{“scope”: org-domain, “type”: string, “link”: blob}”
  - Link can have additional metadata details as desired by the publisher/context
  - <scope + type> defines the convention for the link and metadata description
  - Examples: {“owner”: slac.edu, “type”: “dataset”, “link”: ”None”}

Catalog Features

A Simple REST interface for a catalog

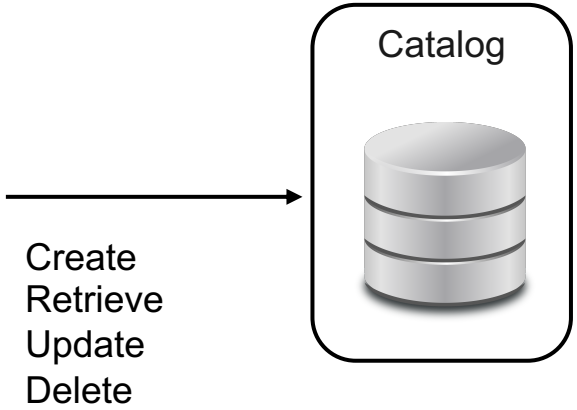
Component would support standard access control mechanisms

Component would be implemented in accordance with ‘Operate First’ principles

– Group will provide a container that can be deployed readily to obtain an instance

Guiding Principle

– Least number of additional specifications to support largest number of use-cases

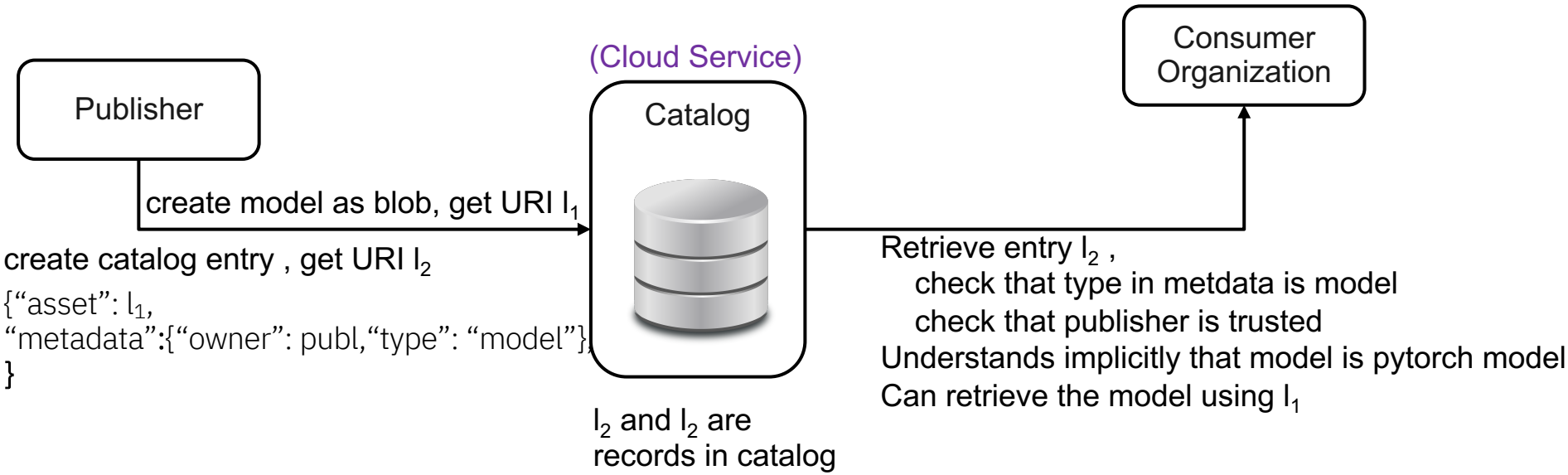


Usage Scenario 1

An organization creates an AI model and wants to share it with another organization

– Organizations have agreed upon their own acceptable vocabulary of types of model and type of acceptance

  • An AI model is a Neural Network exported in pytorch

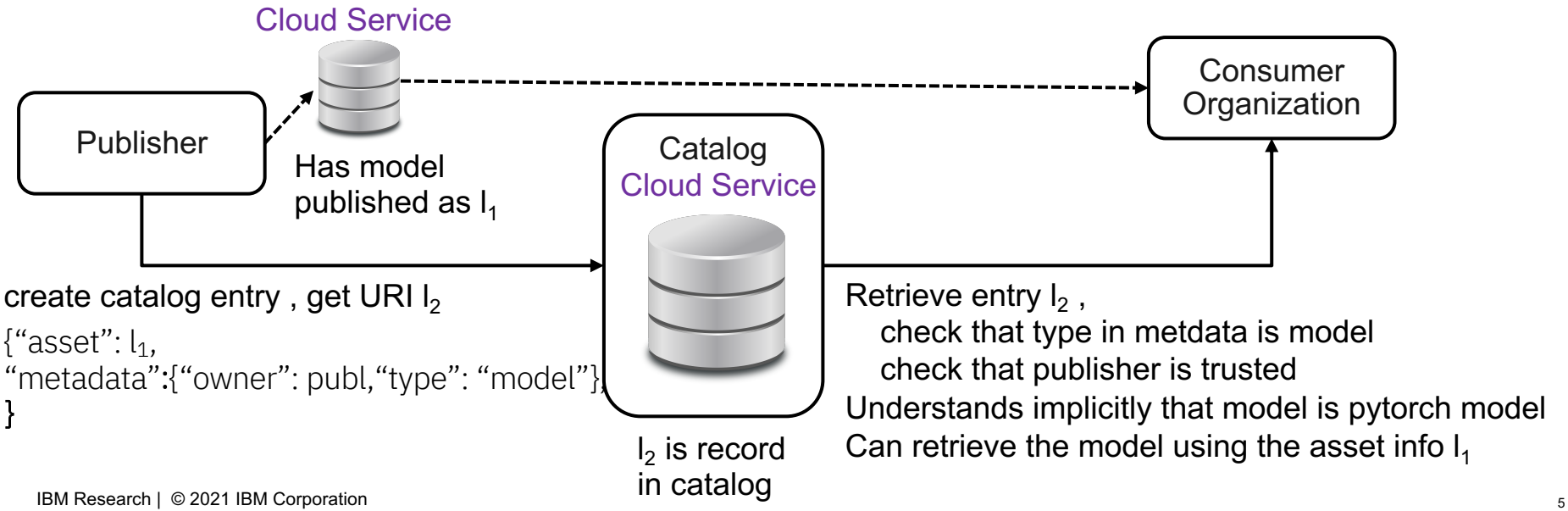


Usage Scenario 2

An organization creates an AI model and wants to share it with another organization

– Organizations have agreed upon their own acceptable vocabulary of types of model and type of acceptance
  - An AI model is a Neural Network exported in pytorch

– Organization has its own repository of assets

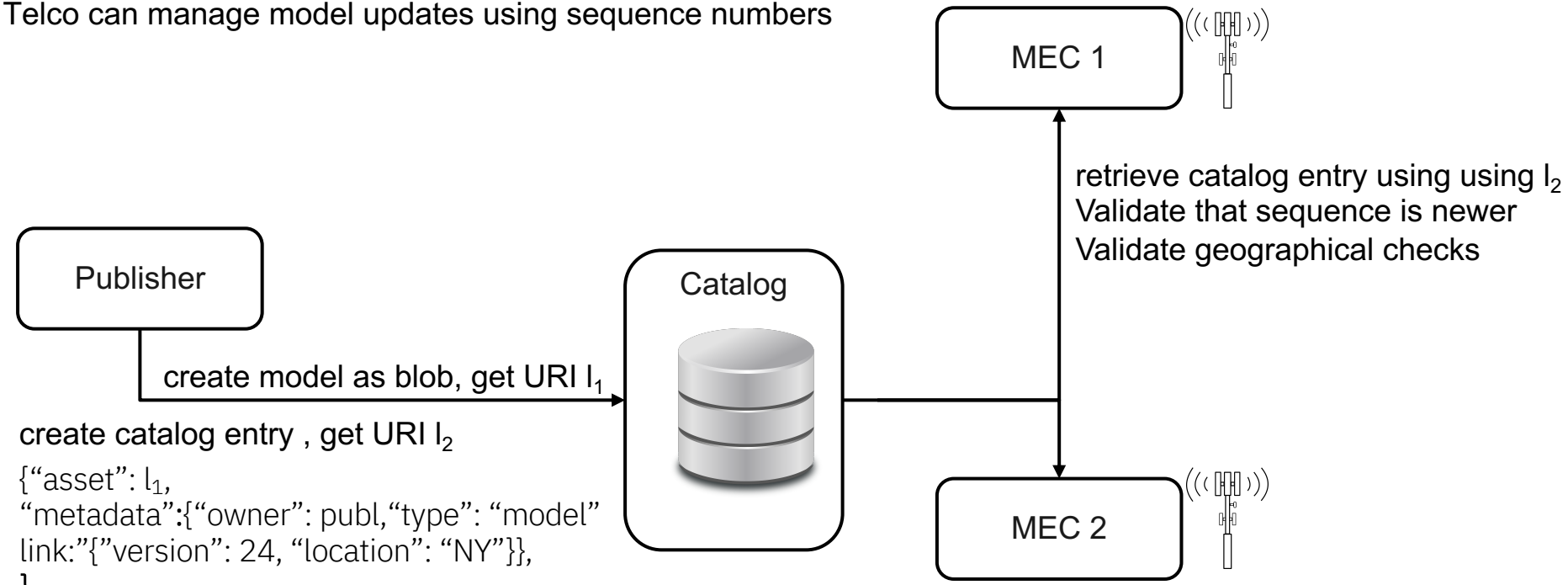


Usage Scenario 3

A Telco publishes a model for Visual Analytics, it is to be used by MEC servers at base of its towers

– Telco adds additional information in metadata link (e.g. a sequence no, or a geographic boundary check}

Telco can manage model updates using sequence numbers



Suggestion from Team: Break it into more Telco-soecific use-cases

Usage Scenario 4

Federated Learning with several organizations and a Model Fusion Service

Each site publishes their models with round number in the catalog, along with environment for training

Fusion server picks up the entries with different rounds and merges them, republishing with catalog

- Fusion server could call transformations on retrieved model if it is in a different environment/architecture
- Metadata link would contain round no, link to merged model and version of models used from each
- Each site will remove their model once the version is merged