

IN-CORE Platform

Jong Lee, Ph.D.

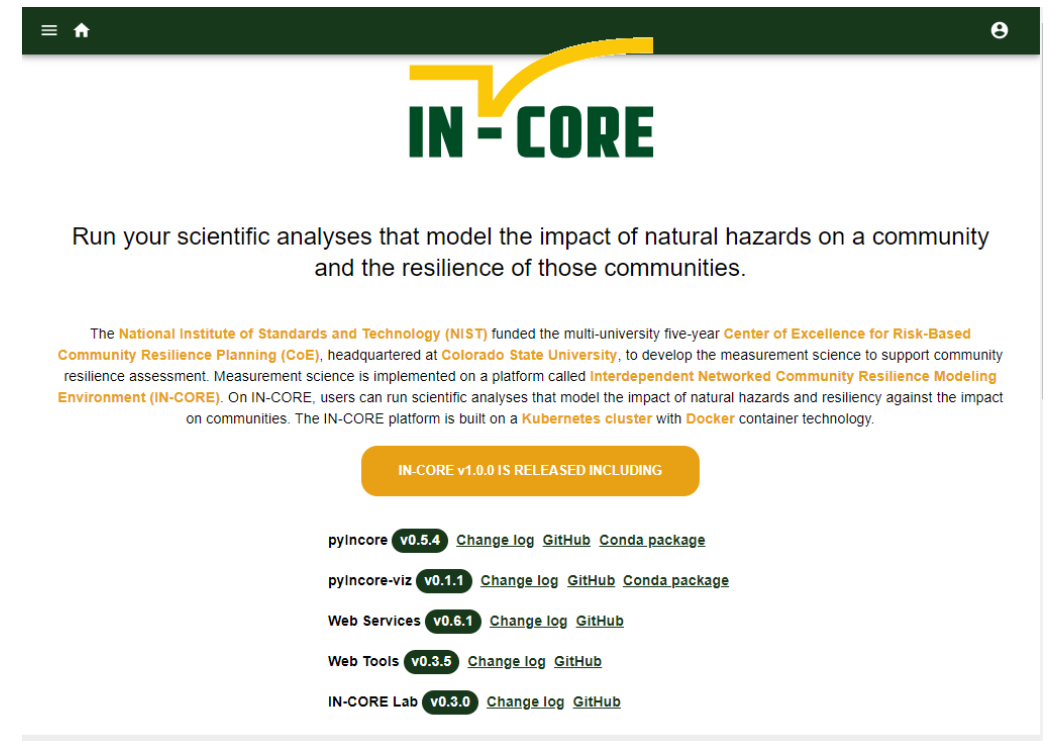
Co-PI, NIST-CoE Community Resilience

Deputy associate director, Software

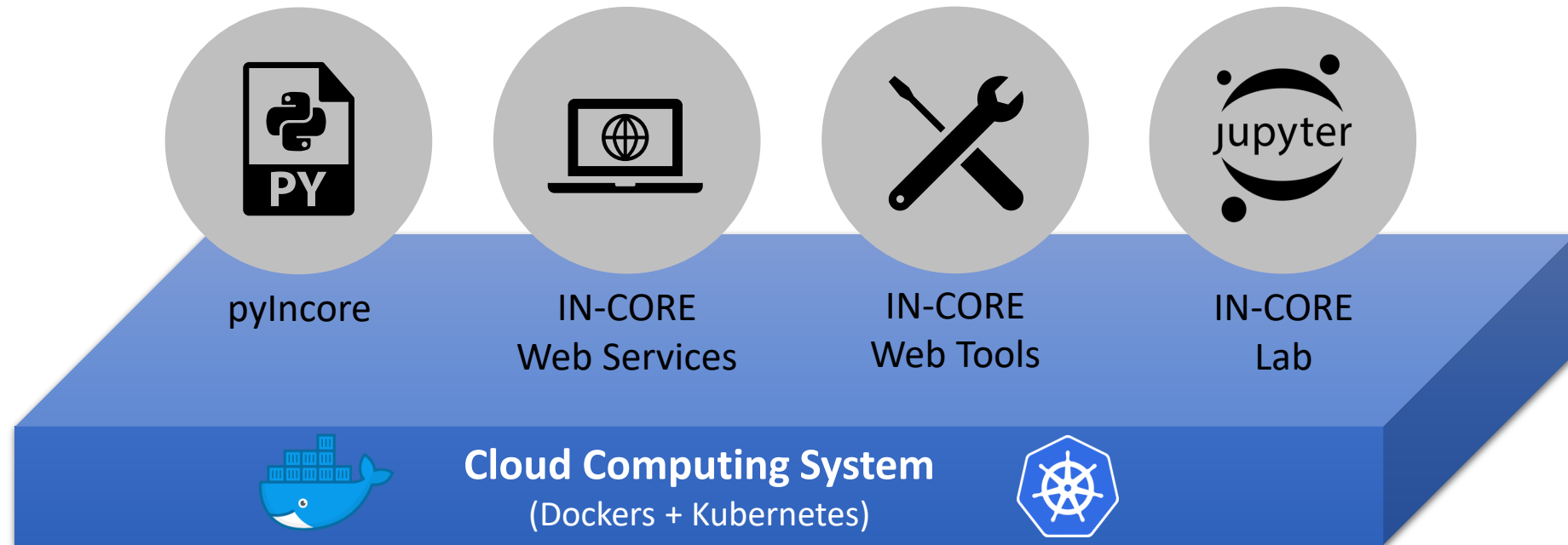
National Center for Supercomputing Applications

IN-CORE Release

- Latest release: 2.8 (Released on December 15, 2021)
- Source code at GitHub
 - <https://github.com/IN-CORE>
 - Mozilla Public License v2.0 (MPL-2.0)
- Conda packages
 - <https://anaconda.org/IN-CORE>
- IN-CORE landing page
 - <https://incore.ncsa.illinois.edu/>



Architecture



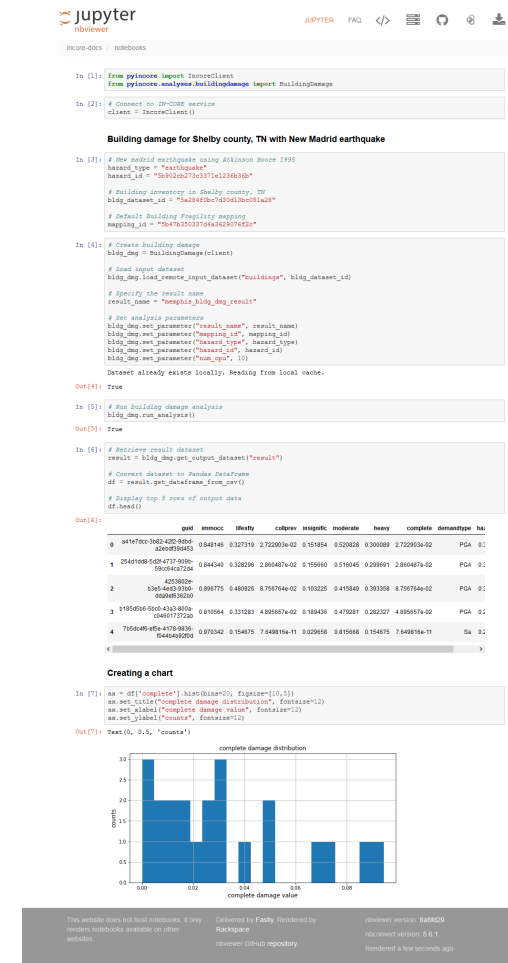
Containers on Kubernetes

- Container (Docker): A container image is a lightweight, stand-alone, executable package of a piece of software that includes everything needed to run it
- Kubernetes is a container management system
- The technology brings us
 - Automatic scaling corresponding to demands
 - Portability – deployable to different cloud
 - Streamline deployment from development and testing



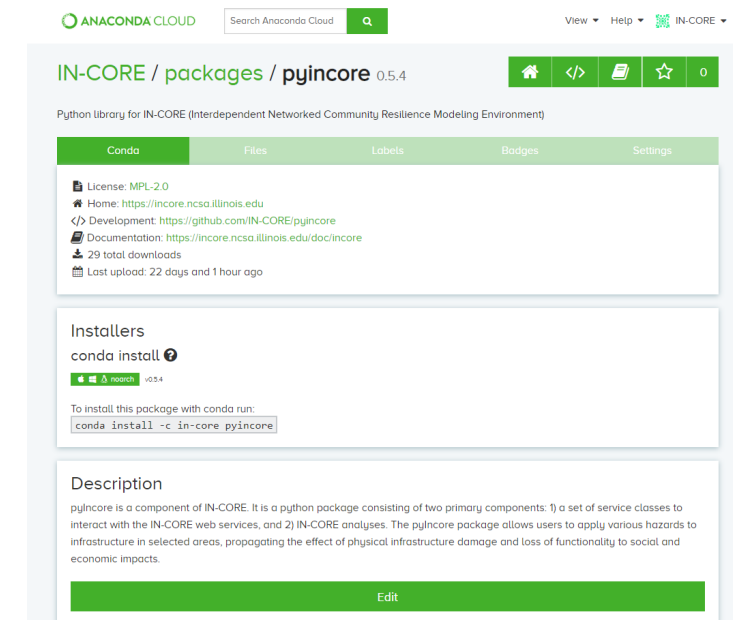
pyIncore, pyIncore-viz , pyIncore-data

- Python library (modules) for IN-CORE
- Three components
 - Interact with IN-CORE web services
 - Base classes for analysis and datasets
 - Analyses
- pyIncore-viz
 - Visualization methods and utilities
- pyIncore-data
 - Data utility methods for preparing IN-CORE compatible data
- How to install
 - `conda install -c in-core pyincore`
 - `conda install -c in-core pyincore-viz`
- Documentation is available
 - <https://incore.ncsa.illinois.edu/doc/incore/index.html>



pyIncore Resources

- pyIncore
 - GitHub: <https://github.com/IN-CORE/pyincore>
 - Anaconda: <https://anaconda.org/IN-CORE/pyincore>
 - General documentation: <https://incore.ncsa.illinois.edu/doc/incore/pyincore.html>
 - Technical reference documentation: <https://incore.ncsa.illinois.edu/doc/pyincore/>
- pyIncore-viz
 - More capability will come in future release
 - GitHub: <https://github.com/IN-CORE/pyincore-viz>
 - Anaconda: <https://anaconda.org/IN-CORE/pyincore-viz>



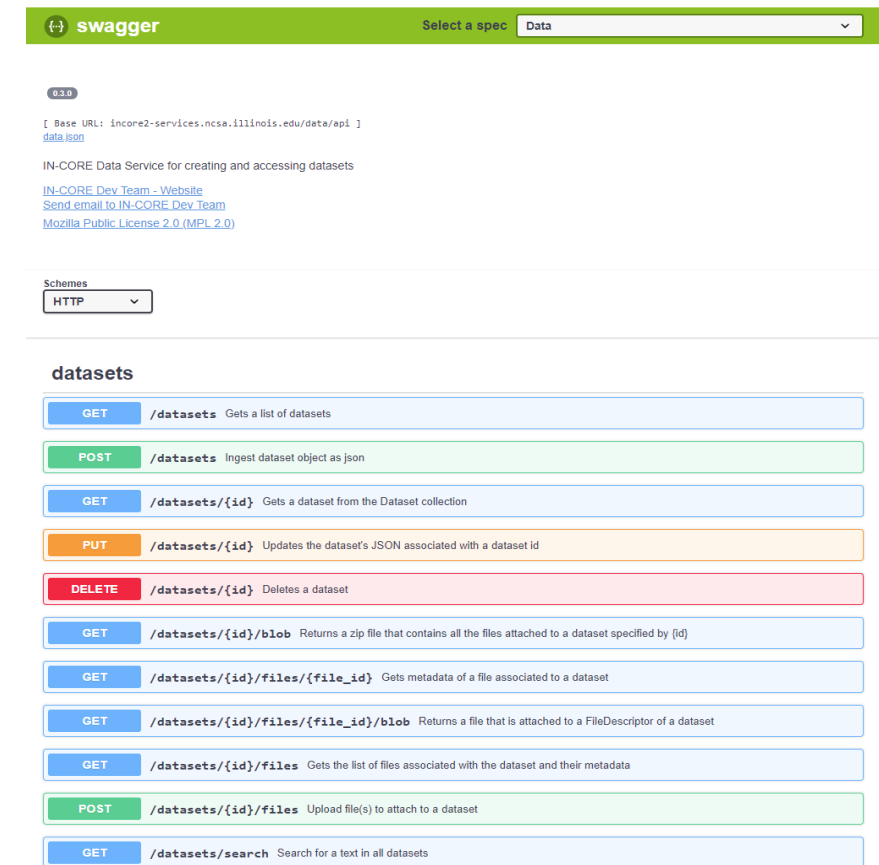
The screenshot shows the Anaconda Cloud interface for the 'pyincore' package. The header includes the Anaconda Cloud logo, a search bar, and navigation links for 'View', 'Help', and 'IN-CORE'. The main content area displays the package name 'IN-CORE / packages / pyincore 0.5.4' with icons for home, code, documentation, and settings. Below this, it states 'Python library for IN-CORE (Interdependent Networked Community Resilience Modeling Environment)'. A tabbed interface shows 'Conda' as the active tab, displaying package details: License (MPL-2.0), Home (https://incore.ncsa.illinois.edu), Development (https://github.com/IN-CORE/pyincore), Documentation (https://incore.ncsa.illinois.edu/doc/incore), 29 total downloads, and last upload (22 days and 1 hour ago). The 'Installers' section shows 'conda install' with a version 'v0.5.4' and a button to 'Install'. Below this, a 'Description' section explains that pyincore is a component of IN-CORE, consisting of service classes to interact with IN-CORE web services and IN-CORE analyses, allowing users to apply various hazards to infrastructure in selected areas. An 'Edit' button is located at the bottom of the description.

IN-CORE Web Services

- RESTful Web Service Technology
- Database: MongoDB
- Authentication service
- Data service
 - Storing/managing datasets
- Hazard service
 - Storing hazard definitions
 - Getting hazard value by location
 - Earthquake
 - Tsunami
 - Tornado
 - Hurricane wind field
- DFR3 service
 - Storing/managing fragility curve sets, damage functions, repair, recovery, restoration
 - Matching inventory to fragility curve set
- Geospatial Viz service
 - Generating geospatial map/layer images
- Semantic service
 - Storing/managing definition of datasets
 - Coming to next release
- Space service
 - Creating content spaces
 - Access control

IN-CORE Web Services

- How to use IN-CORE Web Services
 - Need to have a user account managed by NCSA identity management system
 - For authentication
 - For authorization (access control)
 - Various ways
 - RESTful web service clients
 - Web browser
 - pyIncore
 - IN-CORE Web Tools (browsing only)
- Technical reference documentation is available



The image shows a Swagger UI interface for the IN-CORE Data Service. At the top, there's a green header with the Swagger logo and a dropdown menu labeled "Select a spec" with "Data" selected. Below the header, the base URL is displayed as "incore2-services.ncsa.illinois.edu/data/api". The interface lists several API endpoints under the "datasets" group, each with a colored button indicating the HTTP method (GET, POST, PUT, DELETE) and a brief description of the endpoint's function.

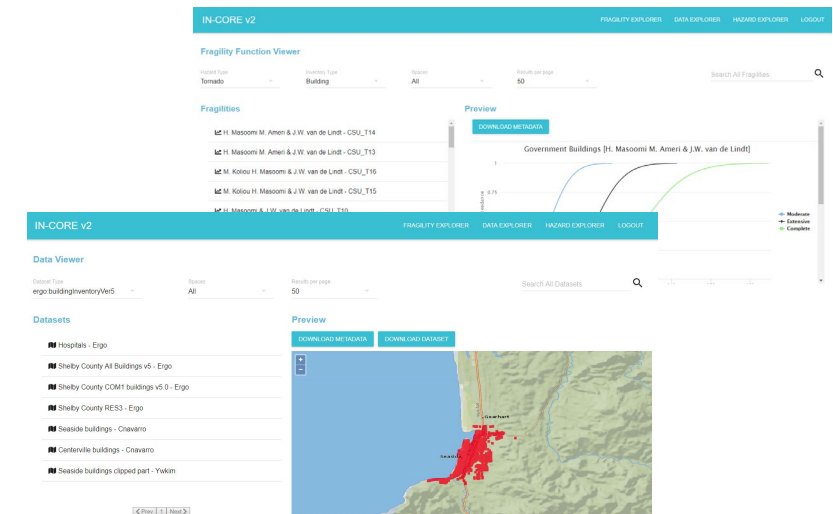
Method	Endpoint	Description
GET	/datasets	Gets a list of datasets
POST	/datasets	Ingest dataset object as json
GET	/datasets/{id}	Gets a dataset from the Dataset collection
PUT	/datasets/{id}	Updates the dataset's JSON associated with a dataset id
DELETE	/datasets/{id}	Deletes a dataset
GET	/datasets/{id}/blob	Returns a zip file that contains all the files attached to a dataset specified by {id}
GET	/datasets/{id}/files/{file_id}	Gets metadata of a file associated to a dataset
GET	/datasets/{id}/files/{file_id}/blob	Returns a file that is attached to a FileDescriptor of a dataset
GET	/datasets/{id}/files	Gets the list of files associated with the dataset and their metadata
POST	/datasets/{id}/files	Upload file(s) to attach to a dataset
GET	/datasets/search	Search for a text in all datasets

IN-CORE Web Services Resources

- GitHub:
 - <https://github.com/IN-CORE/incore-services>
- Technical reference documentation:
 - <https://incore.ncsa.illinois.edu/doc/api/>

IN-CORE Web Tools

- Lightweight web applications for IN-CORE Web Services
- Allows users to browse, search, and preview data from the service
- Data browser
 - Client to data service
- Fragility browser
 - Client to DFR3 service
 - Currently it shows fragilities
- Hazard browser
 - Client to hazard service
- Login with your account credential to access tools at
 - <https://incore.ncsa.illinois.edu>

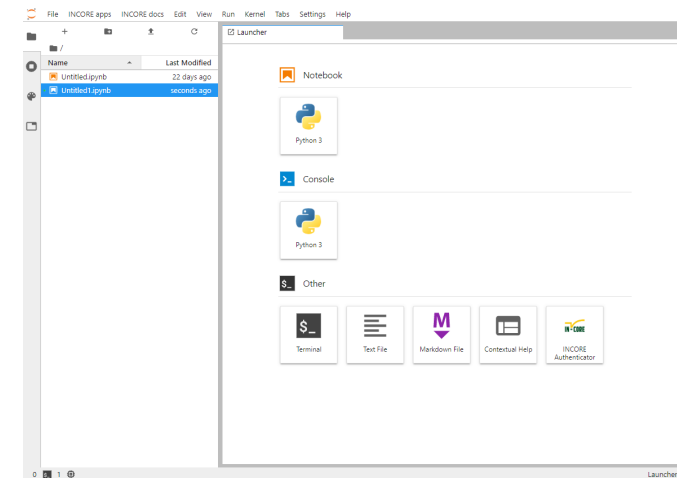


IN-CORE Web Tools Resources

- Access at
 - <https://incore.ncsa.illinois.edu>
- GitHub:
 - <https://github.com/IN-CORE/incore-ui>
- General documentation:
 - <https://incore.ncsa.illinois.edu/doc/incore/webtools.html>

IN-CORE Lab

- Customized JupyterLab
- Integrated environments for developing algorithms
 - Menu items to access documentations, IN-CORE Web Tools
 - Authentication (single-sign-on)
- Two ways to use IN-CORE Lab
 - Locally (a docker image will be available)
 - Online (JupyterHub at NCSA)
- Online version:
 - pyIncore is installed with all dependent libraries
 - Includes popular python libraries such as Pandas, GeoPandas, Matplotlib, etc.
 - Account and allocation policy are under development for public access



IN-CORE Workshop

Workshop Schedule

- Session 1: Overview of IN-CORE & Intro to Jupyter Notebook (60 min)
 - Overview of IN-CORE
 - Intro to Jupyter Notebook and preparing workshop material
- Break (5 min)
- Session 2: Hazard (60 min)
 - Earthquake, Tornado
- Break (5 min)
- Session 3: Building Damage Analysis (65 min)
 - Fragility, Inventory, Damage analysis
- Break (5 min)
- Session 4: Damage analysis on lifeline (60 min)
 - Water facility, Electric Power facility, Pipeline, Roadway
- Closing (5 min)

Preparing Workshop

- In general,
 - Presentation with Jupyter Notebook
 - Hands-on exercise with the presented Jupyter Notebook
 - Breakout sessions: 8 – 10 people with TA(s) (NCSA developers or CoE members)
- Next presentation about Jupyter Notebook will have exercise to acquire the workshop material
- NOTE:
 - IF you don't have IN-CORE account, please try the material after the workshop.
 - We will approve your account after the workshop if you signup the account today or during the workshop.

Support

- Email: incore-dev@lists.illinois.edu
- Documentation: tutorials, tips, and FAQ
- Slack channel will be available soon