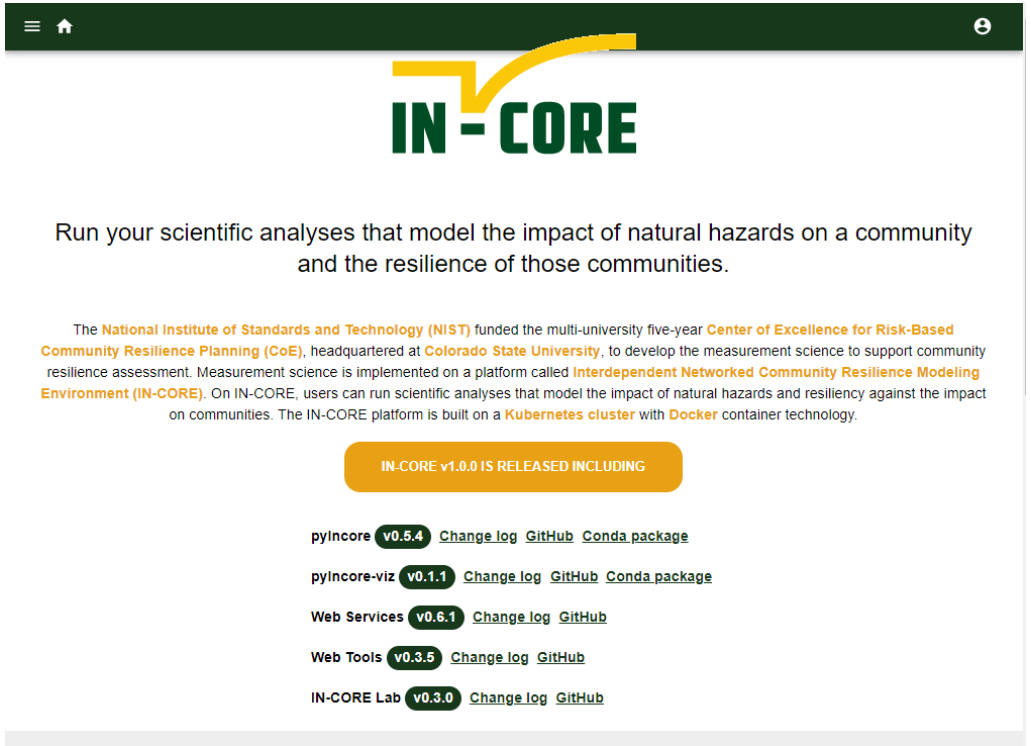
# IN-CORE Platform

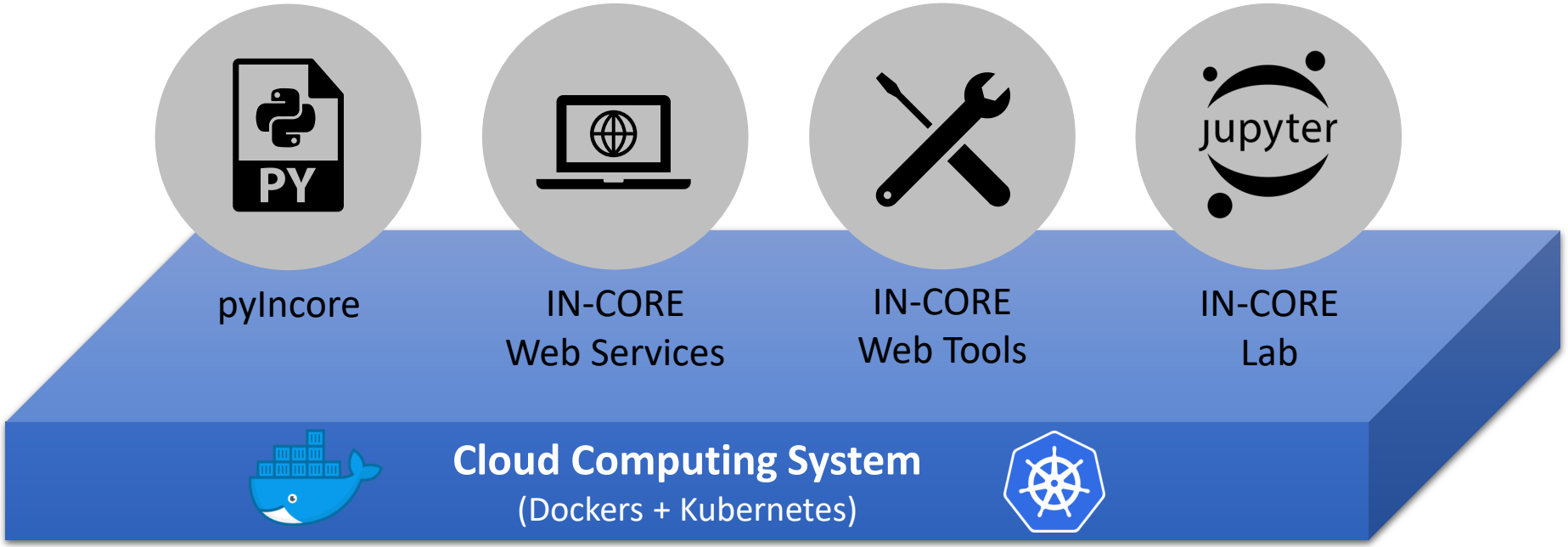
Jong Lee, Ph.D.
Co-PI, NIST-CoE Community Resilience
Deputy associate director, Software
National Center for Supercomputing Applications

# 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

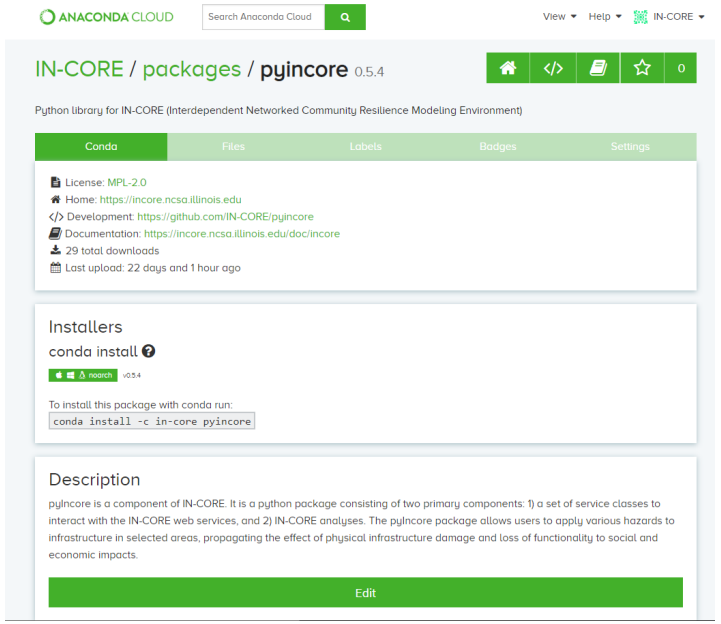
- Container (Docker): A container image is a lightweight, stand-alone, executable package of a piece of software that includes everything needed to run it
- Kubernetes is a container management system
- The technology brings us
  - Automatic scaling corresponding to demands
  - Portability – deployable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

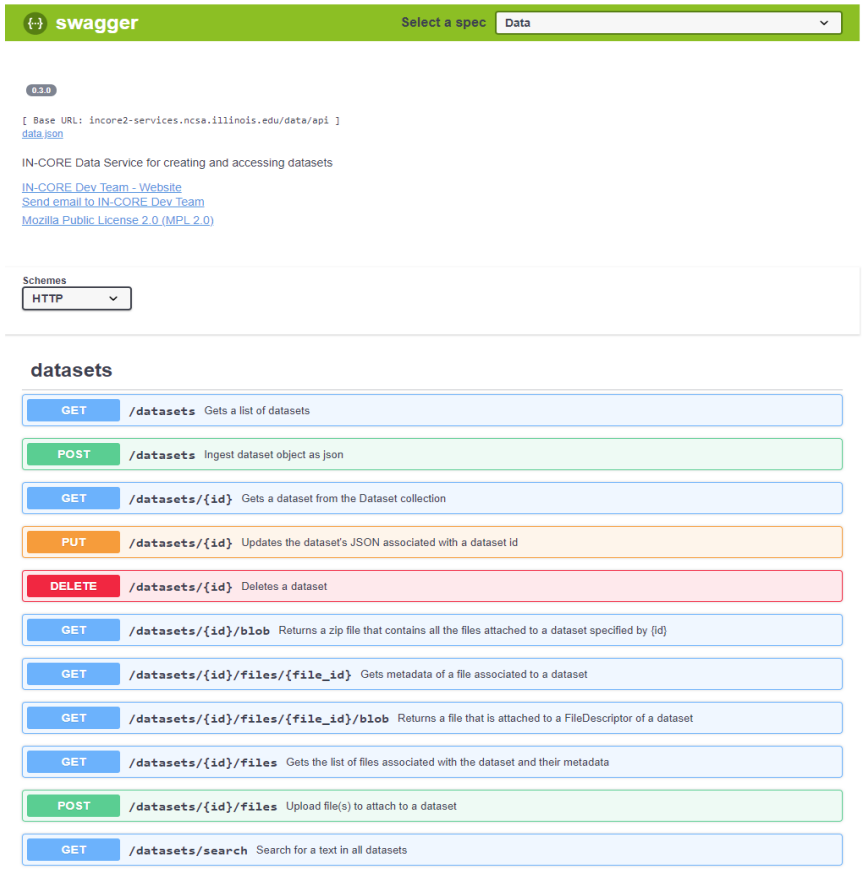


# IN-CORE Web Services

- 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

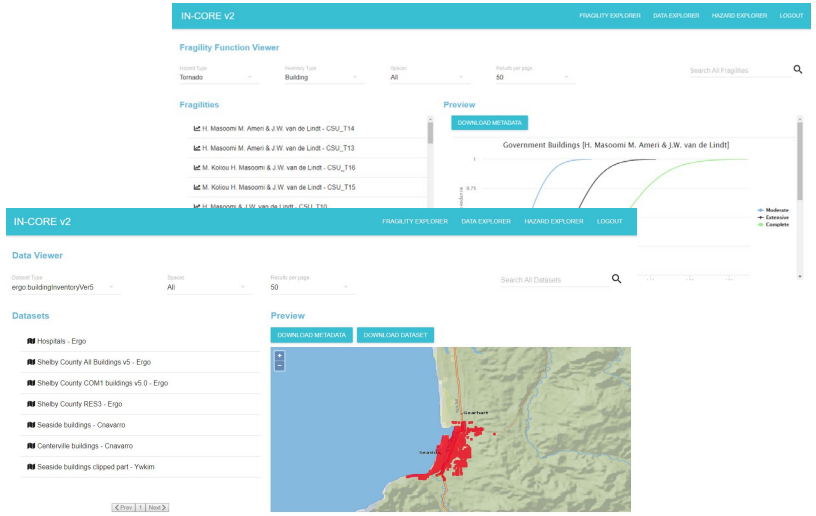


# IN-CORE Web Services Resources

- 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

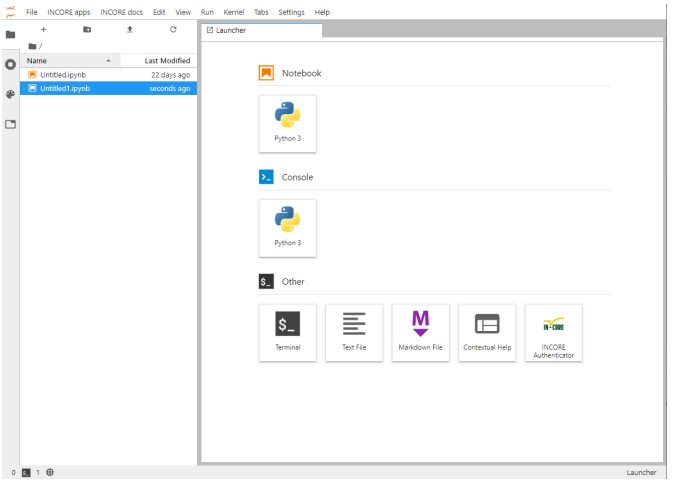


# IN-CORE Web Tools Resources

- 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