# The Joint Effort for Data assimilation Integration (JEDI)


Y. Tremolet and T. Auligne

Joint Center for Satellite Data Assimilation (JCSDA)


NOAA Testbed and Proving Ground Workshop

Kansas City, MO April 10-11, 2018

(Presented by Jim Yoe)

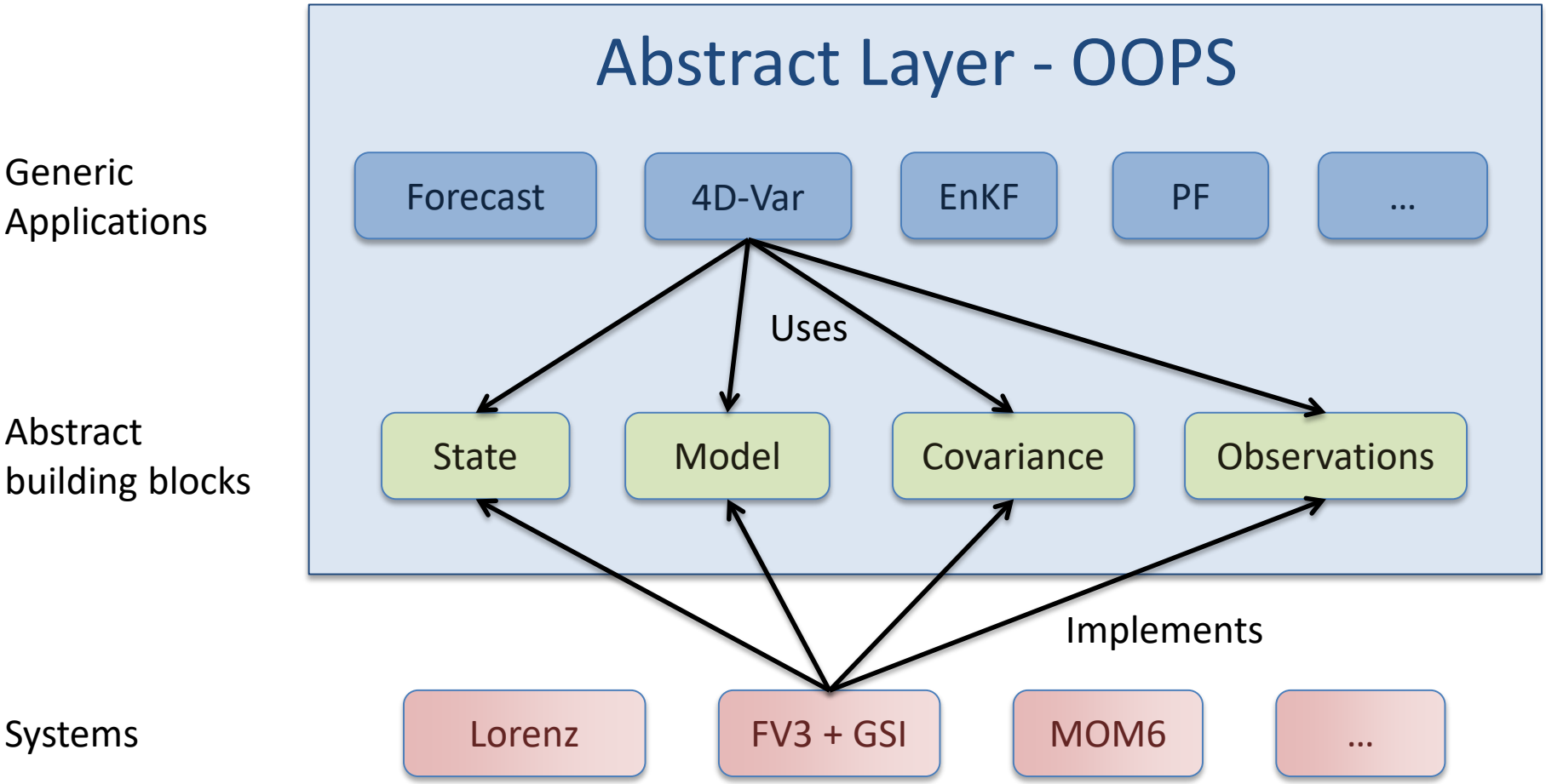
# JEDI: Motivations and Objectives

The Joint Effort for Data assimilation Integration (JEDI) is a collaborative development between JCSDA partners.

Develop a unified data assimilation system:

- From toy models to Earth system coupled models
- Unified observation (forward) operators (UFO)
- For research and operations (including R2O/O2R)
- Share as much as possible without imposing one approach

# Abstract Design: separation of concerns




Abstract interfaces are the most important aspect of the design

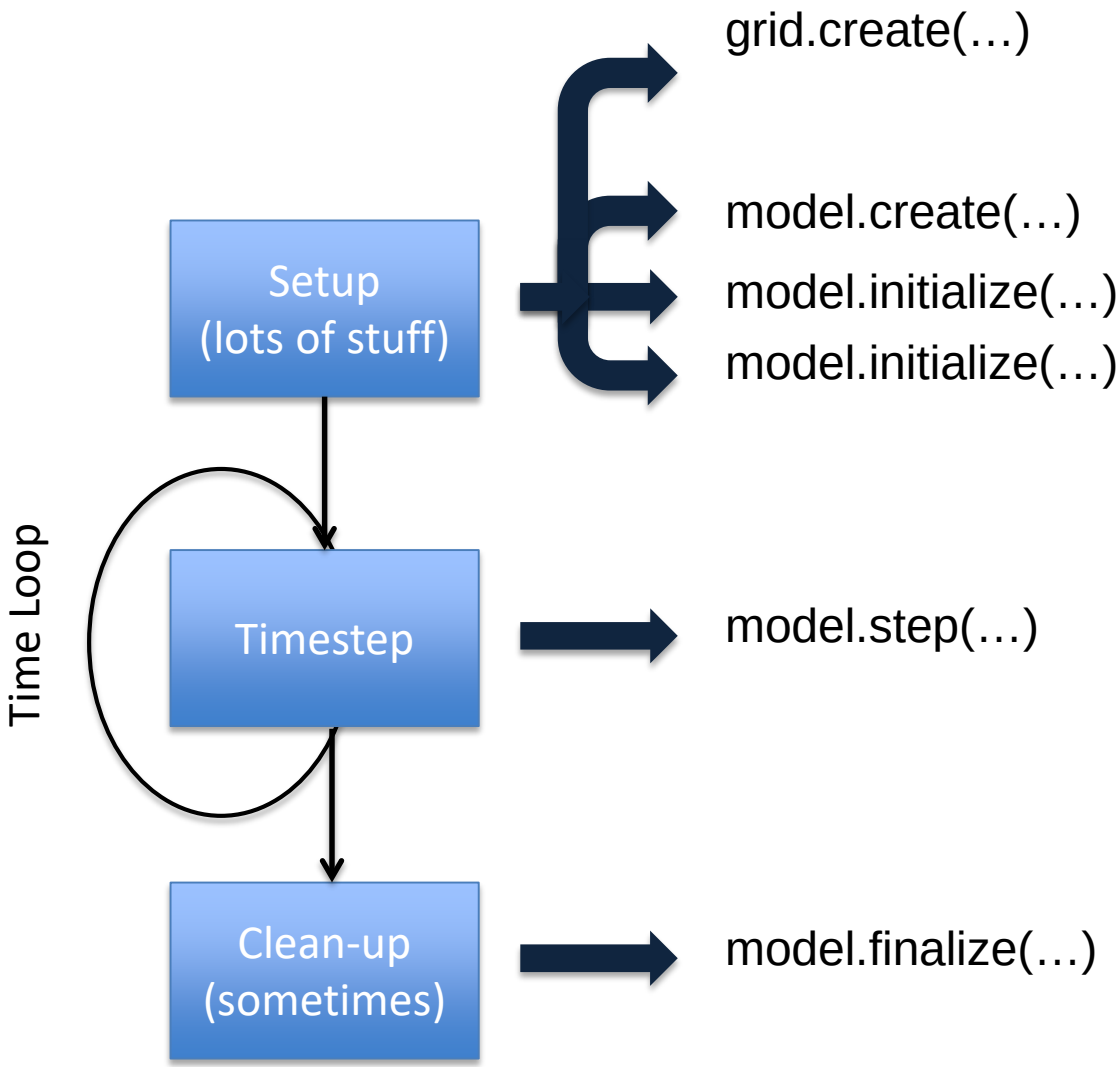
# Joint Effort for Data assimilation Integration

**Model Space Interfaces**

Observation Space Interfaces

Infrastructure and working practices

# Model Design: $x_t=M(x_0)$



# Model Design: Post-processors

- Between model "steps" OOPS calls post-processors
  - OOPS manages when post-processing should be called
  - Post-processing moved away from model code (**separation of concerns**)
  - Adding a post-processor is just adding it to a list

- A post-processor can be anything that
  - Is called (regularly) during model integration
  - Does not modify the State

- OOPS relies on post-processors for isolating data assimilation from the model (**separation of concerns**)
  - Computing simulated observations H(x)
  - Jc-DFI, …

# Joint Effort for Data assimilation Integration

Model Space Interfaces

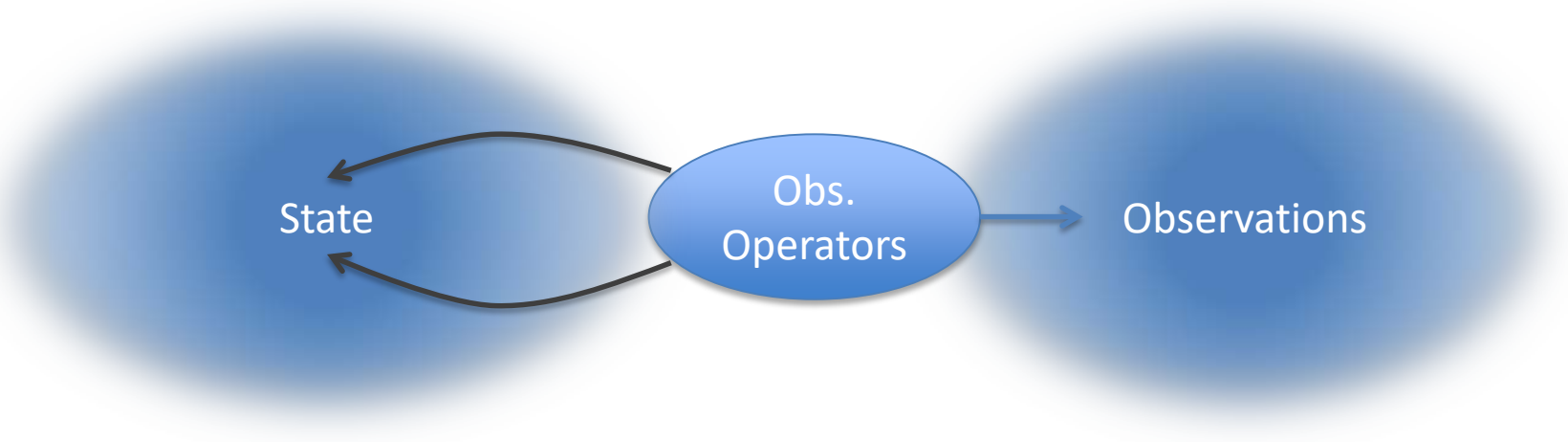
**Observation Space Interfaces**

Infrastructure and working practices

# Observation Space Objectives

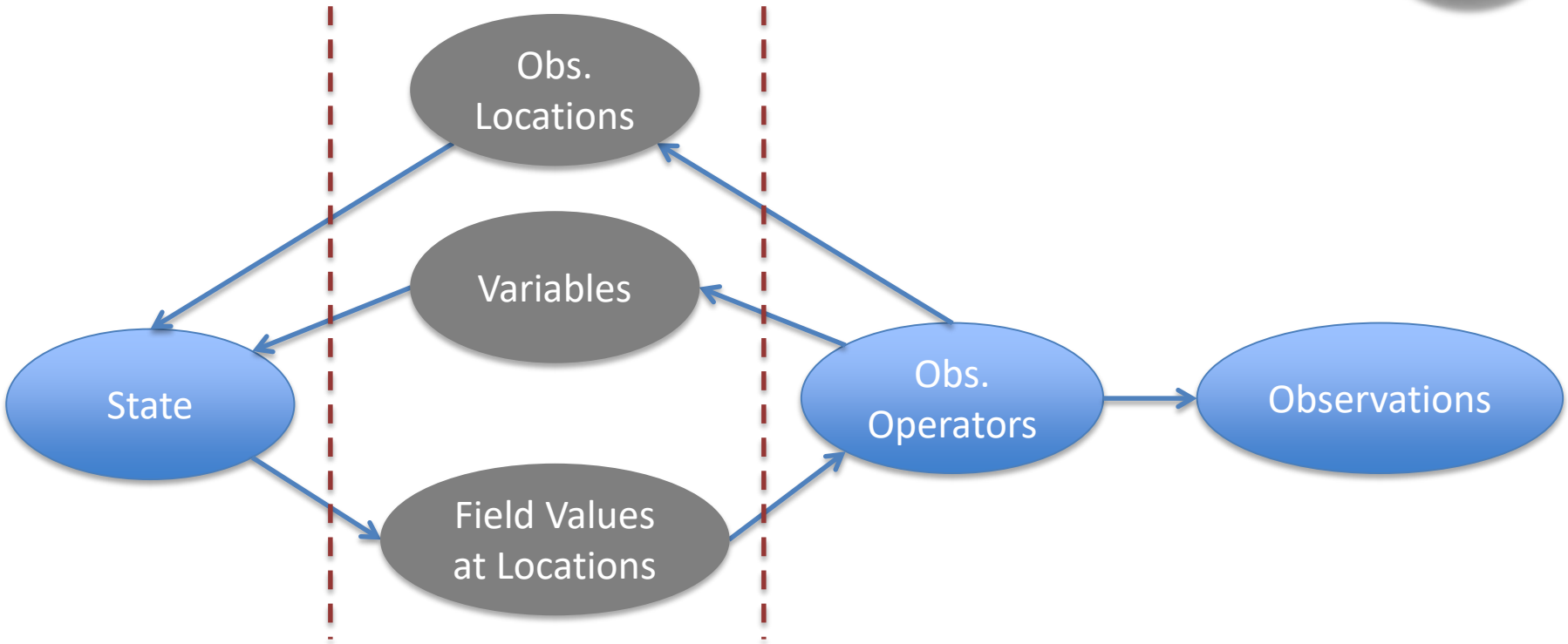
- Share observation operators between JCSDA partners and reduce duplication
  - Include instruments science teams
- Faster use of new observing platforms
  - Regular satellite missions are expensive
  - Cube-sat have short expected life time
- Unified Forward Operator (UFO)
  - Build a community app-store of observation operators (“op-store”)

# Observation Operators




- In most existing systems, observation operators directly access state/model data
- Observation operators, and as a result DA systems, are very model specific

# UFO: the interface advantage




- JEDI/UFO introduces standard interfaces between the model and observation worlds
- Observation operators are independent of the model and can easily be shared, exchanged, compared

# UFO Observation "filters"

- JEDI/UFO calls abstract "observation filters" before and after the actual observation operator
- Filters can be written once and used with many observation types
- Observation filters are generic and have access to
  - Observation values and metadata
  - Simulated observation value (post-filter)
  - Their own private data
- Examples:
  - Quality control (background check, buddy check, cloud detection…)
  - Thinning
  - Saving linearization trajectory or Jacobians

# Interface for Observation Data Access (IODA)

Interface to isolate science code from data storage

Three levels:

- Long term storage (historic database)
- Files on disk (one DA cycle)
- In memory handling of observations (hardware specific?)

Two environments:

- Plotting, analyzing, verifying on workstation
- DA and other HPC applications (MPI, threads, GPUs…)

Goal: one interface, possibly several implementations?

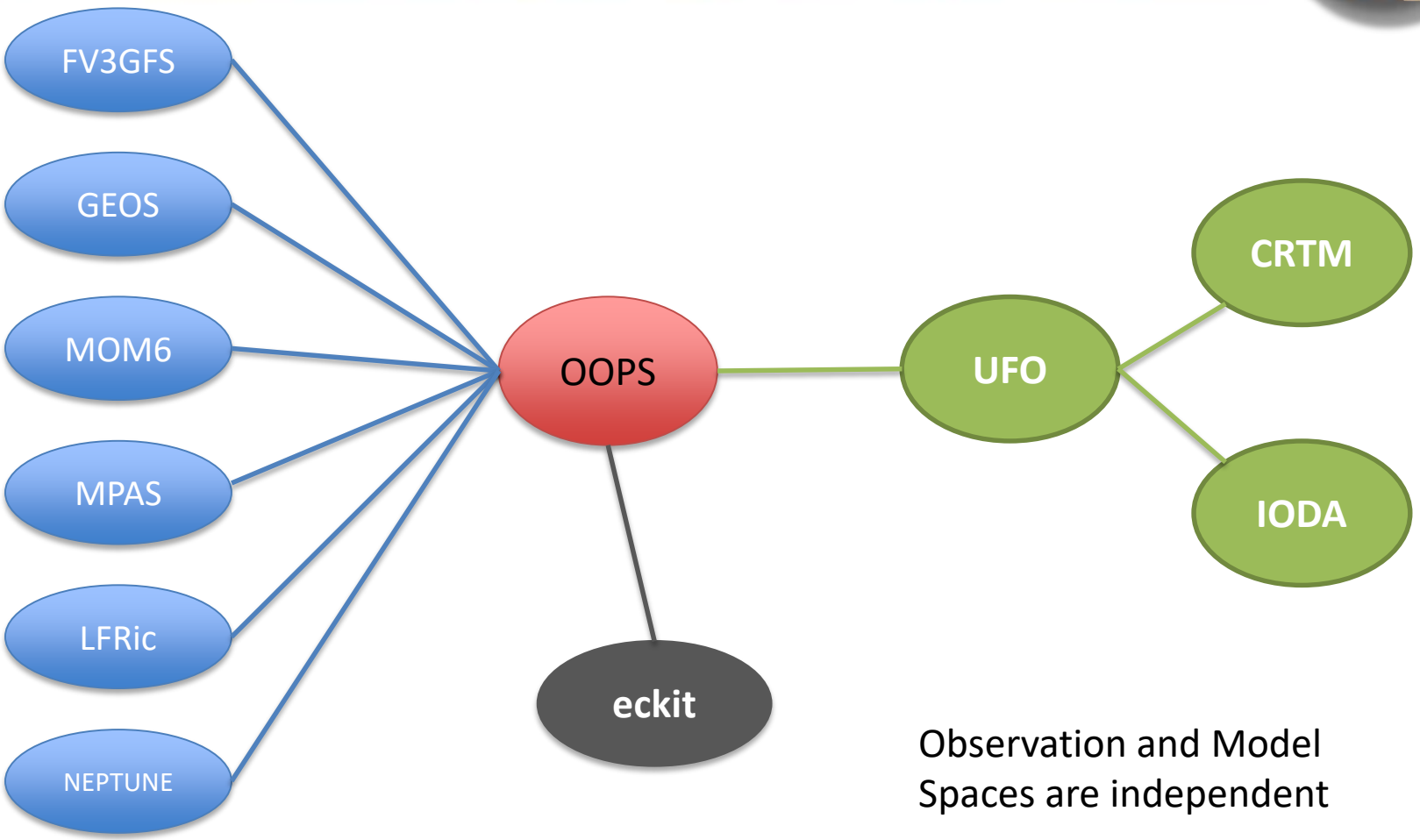
# Joint Effort for Data assimilation Integration

Model Space Interfaces

Observation Space Interfaces

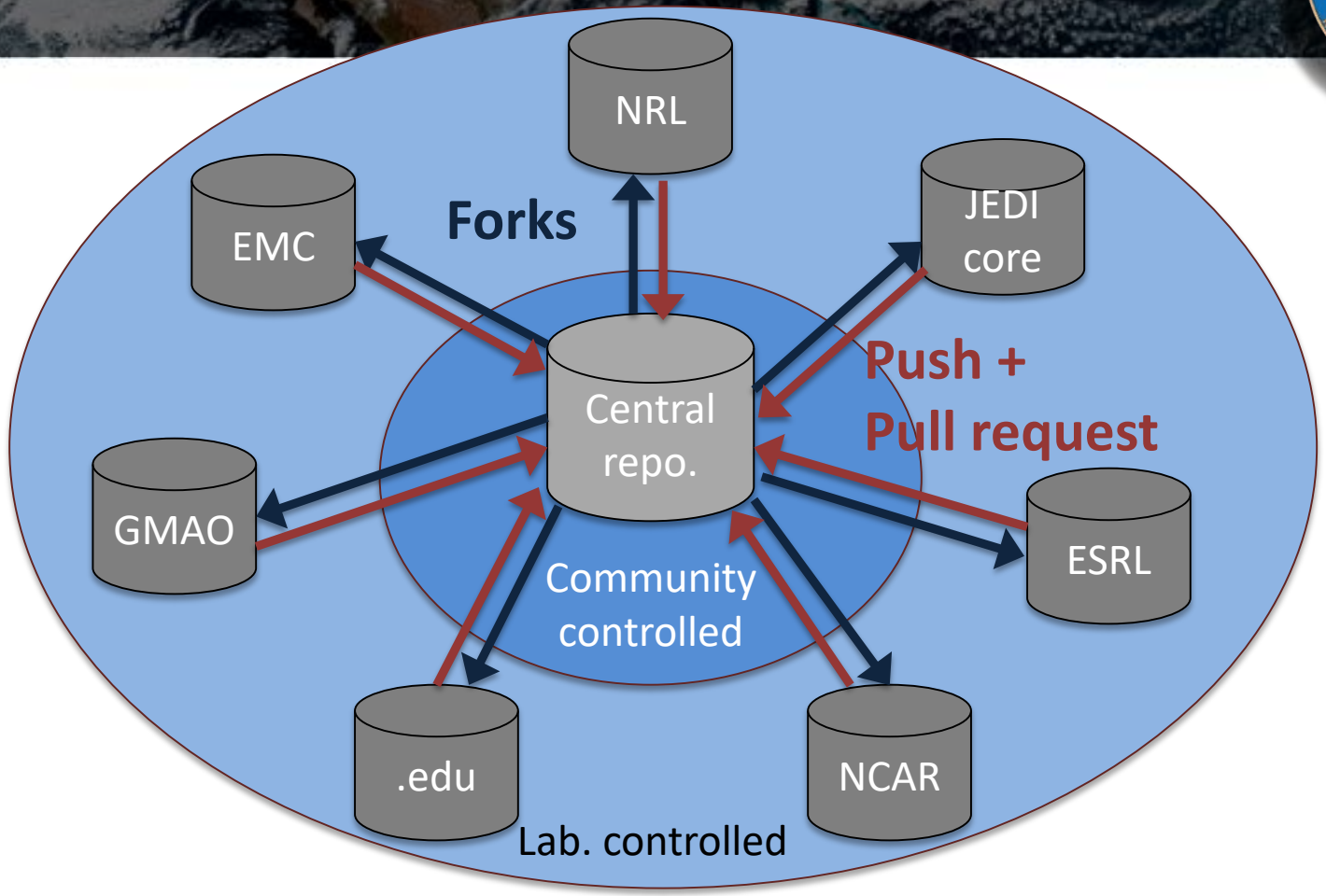
**Infrastructure and working practices**

# Code and repositories




Observation and Model Spaces are independent

# Collaborating: Repositories




Permission to fork repository are very easy to obtain

Contributing code is very controlled:

- Pushing a branch requires write permission on central repository
- Pull request triggers code review and approval for merging to higher level branch

# Governance and code reviews

**Governance** is about management keeping in control and deciding what features should be in the system

**Code reviews** are about quality of the code

Two different levels of control

- Good code can stay outside of central repository (stability of interfaces is important)
- A desired feature that does not satisfy quality requirements cannot be accepted as is

Testing is a pre-requirement for code reviewing

Different people and different pace: **Separation of concerns**…

# Infrastructure, working practices

- Project methodology inspired by Agile/SCRUM
  - Adapted to distributed teams and part time members
- Collaborative environment
  - Easy access to up-to-date source code (github)
  - Easy exchange of information (Confluence, zenhub)
- Flexible build system (ecbuild-based)
- Coding norms
- Documentation, tutorials, JEDI Academy

# Infrastructure, working practices

- Continuous Integration, Testing framework
  - Toolbox for writing tests
  - Automated running of tests (on push to repositories)

- Effort on portability
  - Automatically run tests with several compilers
  - JEDI available in containers (Docker, singularity)

- Enforce software quality (correctness, coding norms, efficiency)

- Change in working practices take time…

# Code Sprints

- Gather 8-10 people in a room for 2 weeks
  - NICAS B Matrix (Aug 2017)
  - Observation Operators (Nov. 2017)
- Efficient use of time, especially for part time contributors
- Involve people from all partner institutions in project
- Very motivating (before, during, after)

# Summary

- JEDI is critical to next-generation DA development (hence to NGGPS)
- Provides scalability/reusability to support multiple applications, users, and contributors
- Builds off successful example (Object Oriented Prediction System – OOPS)
- Coding began in August, 2017
  - Successful components already exist
    - See SOCA example in JCSDA Round-Up Presentation

# Thanks to all involved in JEDI (2017)

Amal El Akkraoui, Anna Shlyaeva, Ben Johnson, Ben Ruston, Benjamin Ménétrier, BJ Jung, Bob Oehmke, Bryan Flynt, Bryan Karpowicz, Clara Draper, Dan Holdaway, Dom Heinzeller, François Vandenberghe, Gael Descombes, Guillaume Vernières, Jeff Whitaker, Jing Guo, John Michalakes, Julie Schramm, Mariusz Pagowski, Mark Miesch, Mark Potts, Ming Hu, Rahul Mahajan, Ricardo Todling, Scott Gregory, Soyoung Ha, Steve Herbener, Steve Sandbach, Tom Auligné, Will McCarty, Xin Zhang, Yannick Trémolet.

# Joint Center for Satellite Data Assimilation

A multi-agency research center created to improve the use of satellite data for analyzing and predicting the weather, the ocean, the climate and the environment.

Collaborative organization funded by

- NOAA/NWS
- NOAA/NESDIS
- NOAA/OAR
- NASA/GMAO
- US Navy
- US Air Force

Organized by projects:

- CRTM (Community Radiative Transfer Model)
- JEDI (Joint Effort for Data assimilation Integration)
- SOCA (Sea-ice Ocean Coupled Assimilation)
- NIO (New and Improved Observations)
- IOS (Impact of Observing Systems)