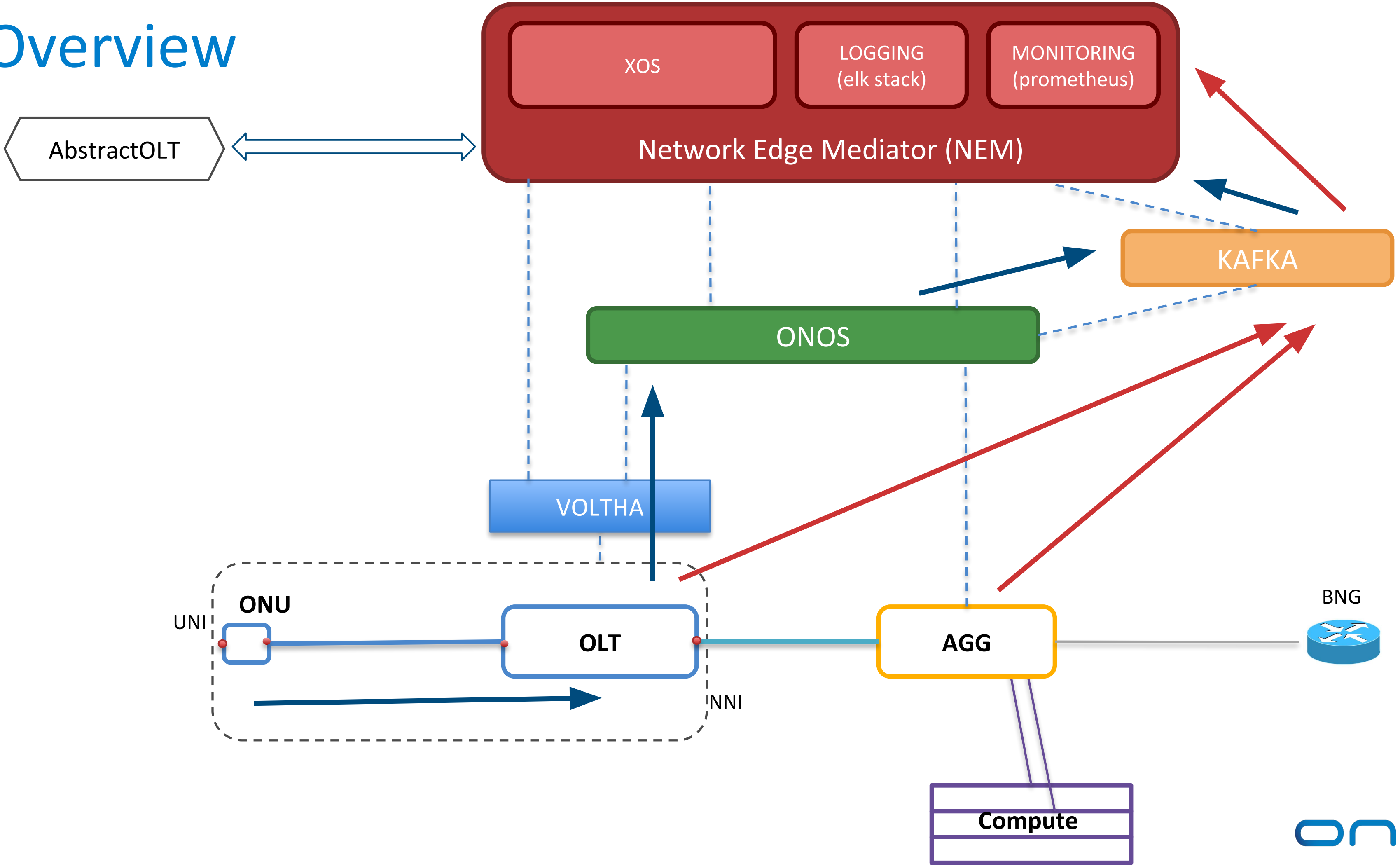
Open Networking Foundation

NEM: Overview and ISSU plans

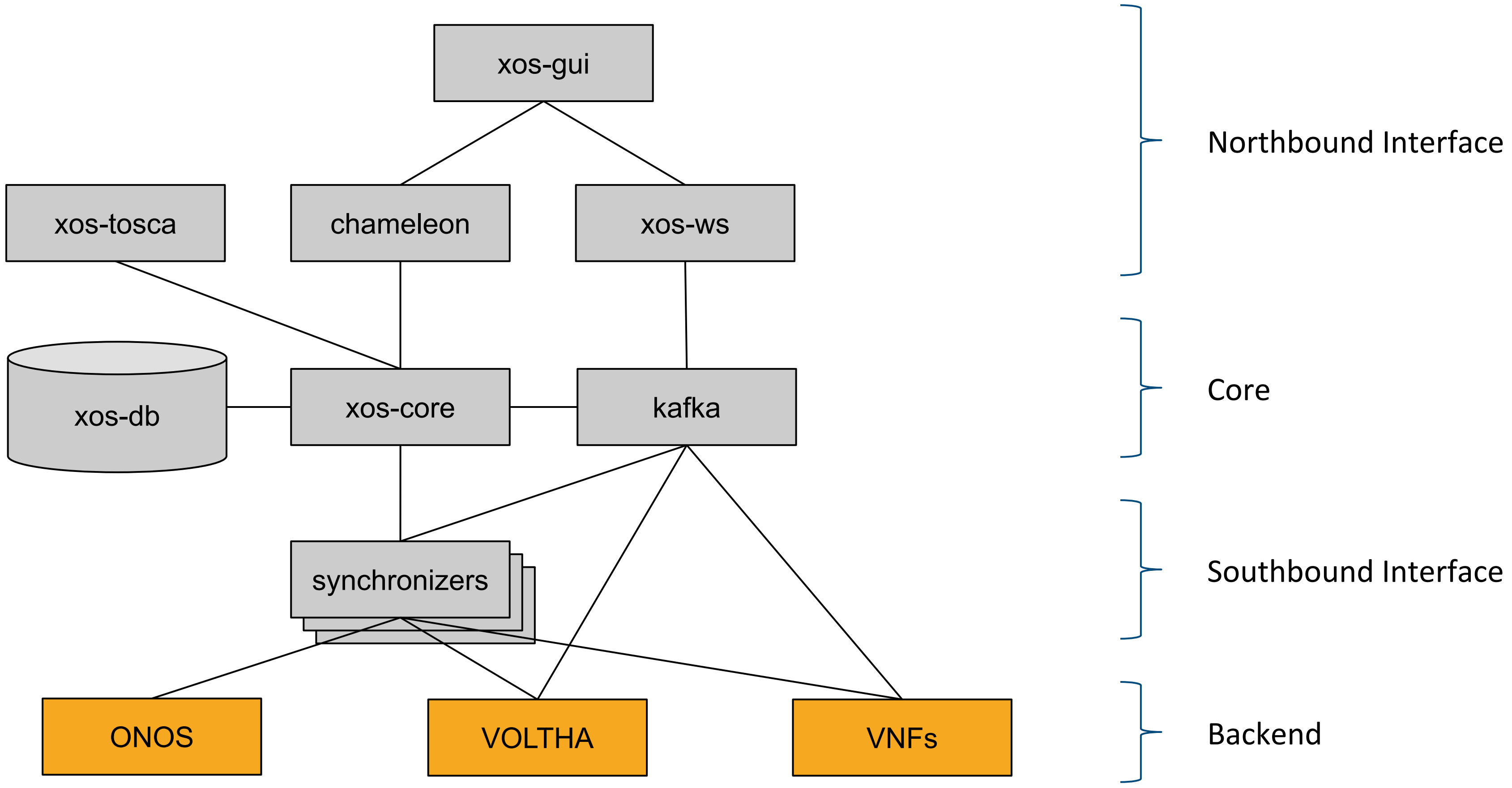
An Operator Led Consortium

Overview

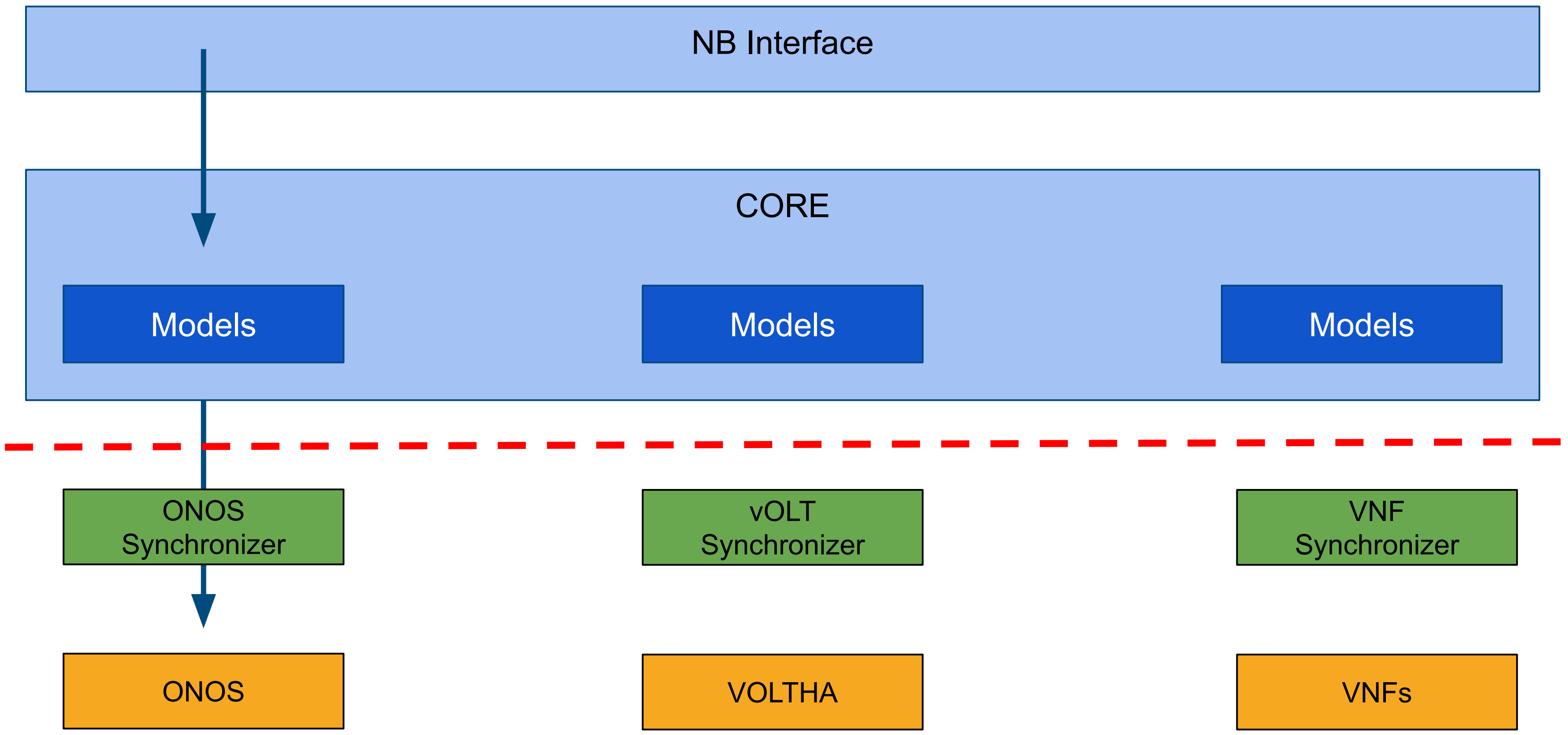


XOS
LOGGING (elk stack)
MONITORING (prometheus)
Network Edge Mediator (NEM)
AbstractOLT
KAFKA
ONOS
VOLTHA
ONU
UNI
OLT
NNI
AGG
BNG
Compute

XOS Architecture

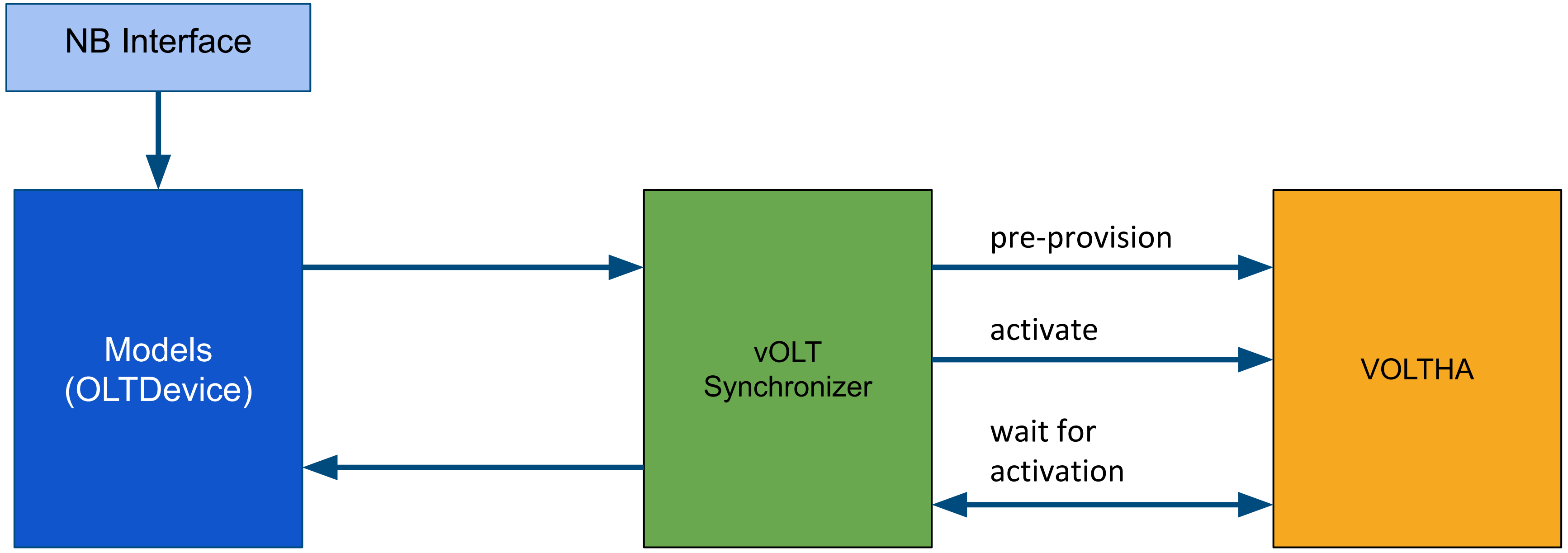


XOS Architecture



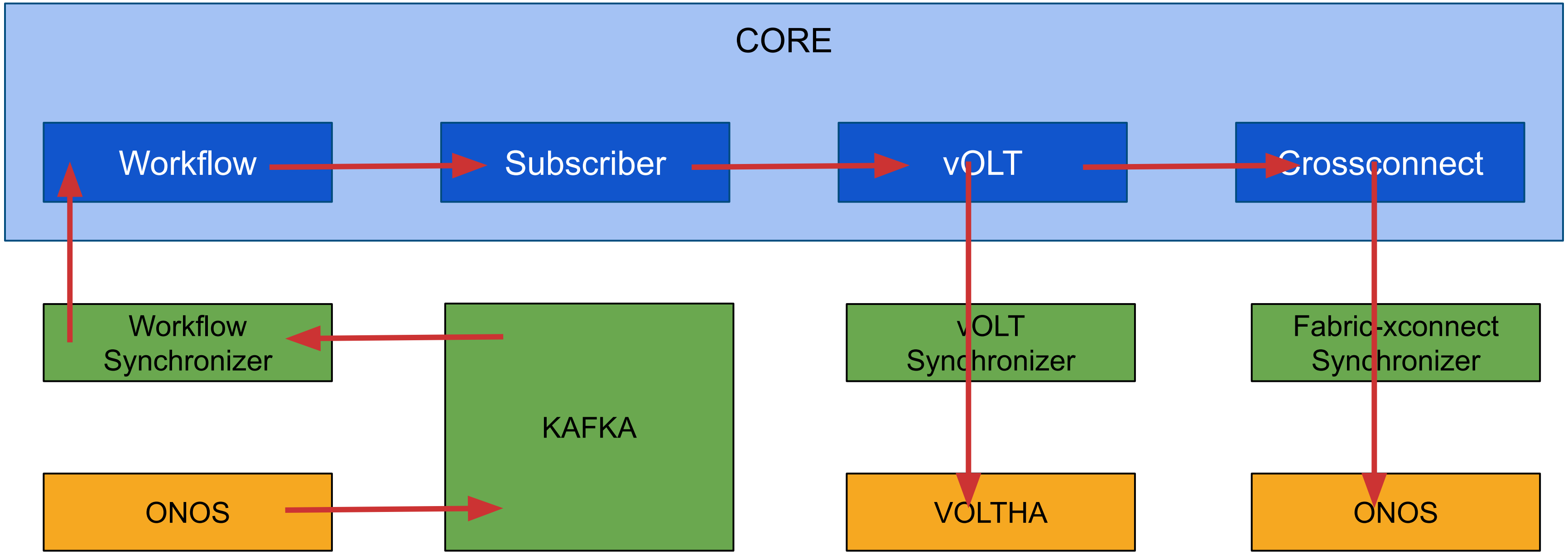
XOS in SEBA

An example operation, OLT provisioning.



XOS in SEBA

An example operation, Subscriber authentication.



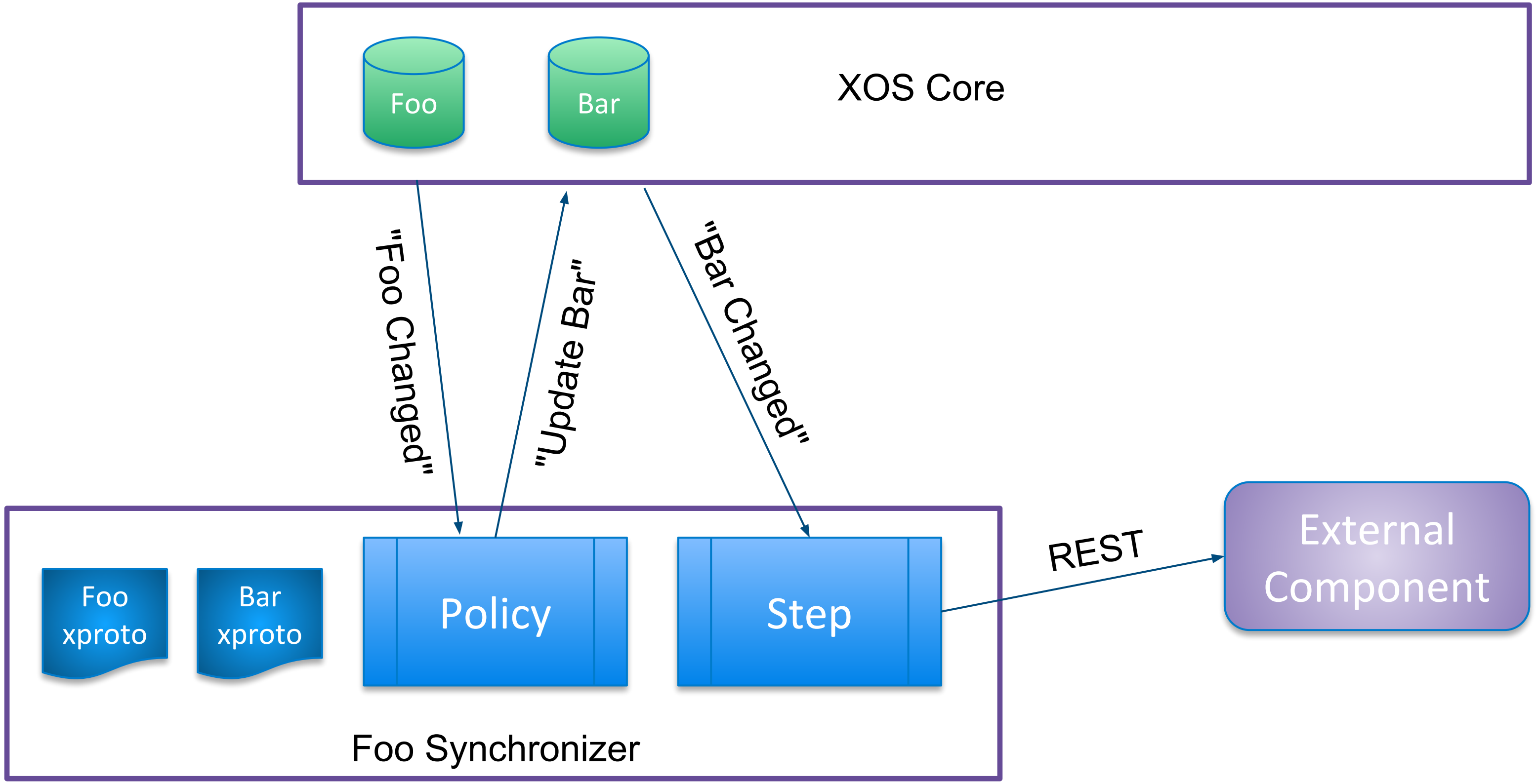
XOS: The Synchronizer Framework

The synchronizer framework allows XOS to be extended in service-specific ways.

- Service-specific models
- Service-specific business logic
- Abstractions and logic that span multiple services

XOS supports diverse heterogeneous services. Different kinds services naturally need different models and logic.

Synchronizers specify models, and implement policies and steps



Types of Steps

- XOS -> External Component
  - Sync Step
  - Delete Step
- External Component -> XOS
  - Pull Step
  - Event Step
- XOS -> XOS
  - Model Policy

Synchronizers: moving to a library

The synchronizer framework was refactored as a python library.

- Developer benefits
    - Compliant with python best-practices
    - Developer friendly (IDEs)
- Community benefits
    - Ease of re-use promotes adoption
- Operational benefits
    - De-layering of containers -> Smaller containers

Migrations: principles

Anytime a model evolves actions needs to be take, mainly:

- Bring the database schema up to date
- Make sure data are kept in a consistent state

Best practices:

- Migrations are treated as code
- Migrations can be executed both ways

Migrations: example

Model v1:

string firstName

string lastName

Model v1.1:

string firstName

string lastName

string fullName

Model v2.0:

string fullName

A field is added (autogenerated)

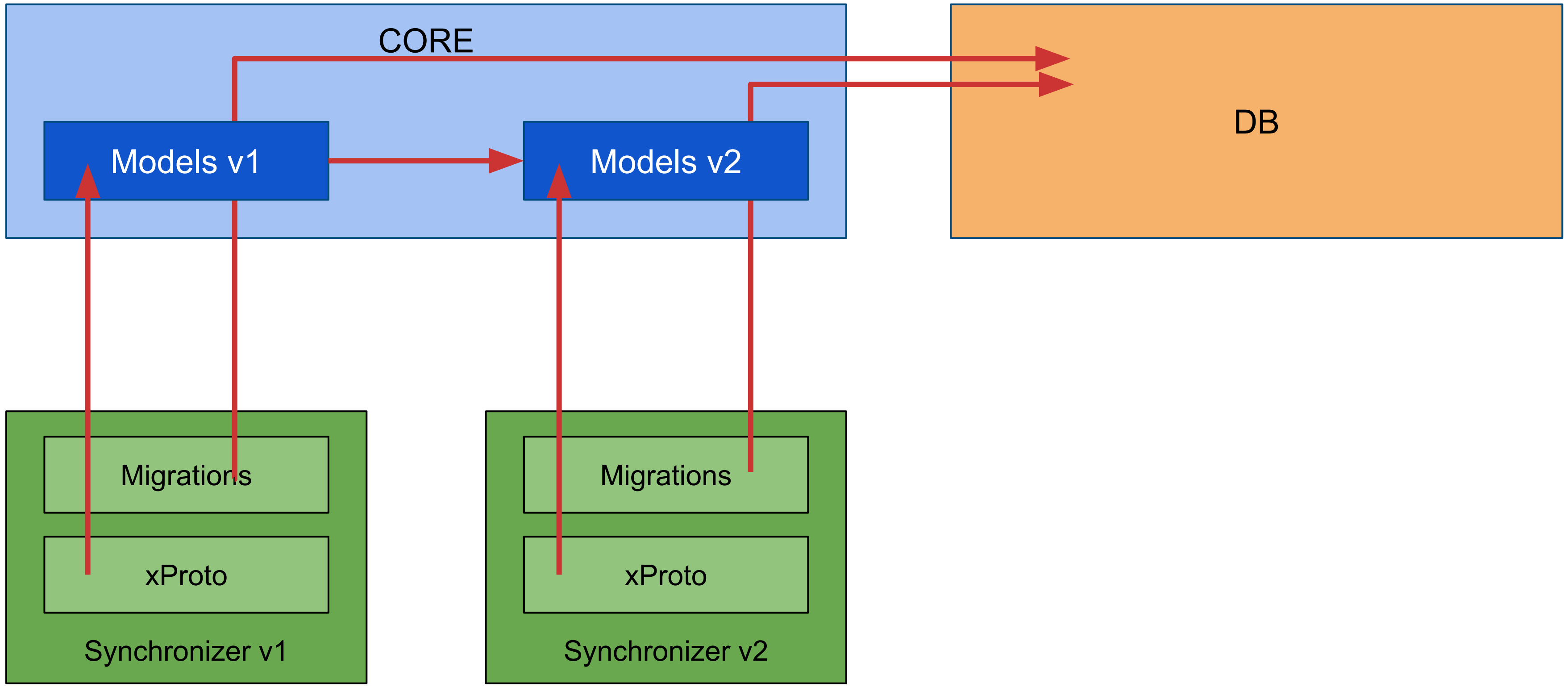
Data are changed (custom logic)

Migrations: XOS

xos-migrate: https://guide.opencord.org/xos/dev/xosmigrate.html

- Generate standard migrations base on xProto changes
- Allow developers to extend migrations with custom logic

Migrations: XOS


CORE
Models v1
Models v2
DB
Migrations
xProto
Synchronizer v1
Migrations
xProto
Synchronizer v2