Biodiversity Expedition: Virtualizing Lifemapper

Nadya Williams, UCSD, nadya@sdsc.edu
Aimee Stewart, KU, aimee.stewart@ku.edu
Phil Papadopoulos, UCSD phil@sdsc.edu

Introduction: the goal

Create a viable virtualization solution that can be easily adopted and reused by scientists at multiple institutions and projects.

Criteria

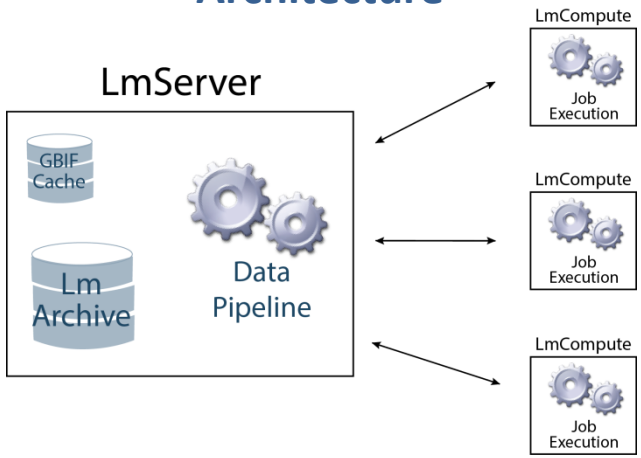
1. allows fast deployment of ready-made cluster images
2. reproduces the complete Lifemapper processing pipeline on demand at multiple sites and in different hosting environments
3. enables scientists to perform Lifemapper-facilitated data processing on restricted-use data, very large datasets, or other unique data.

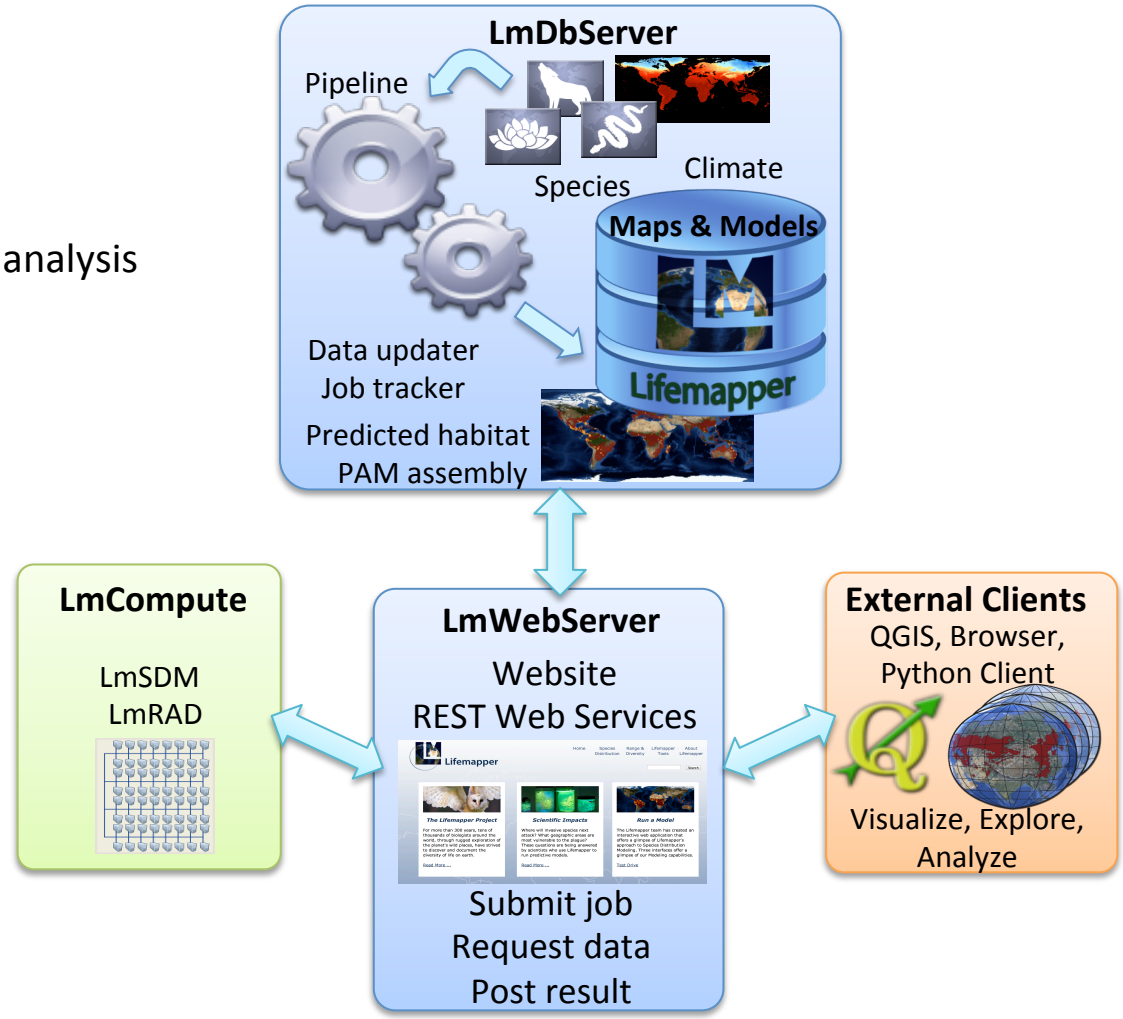
Lifemapper main components

What is Lifemapper ?

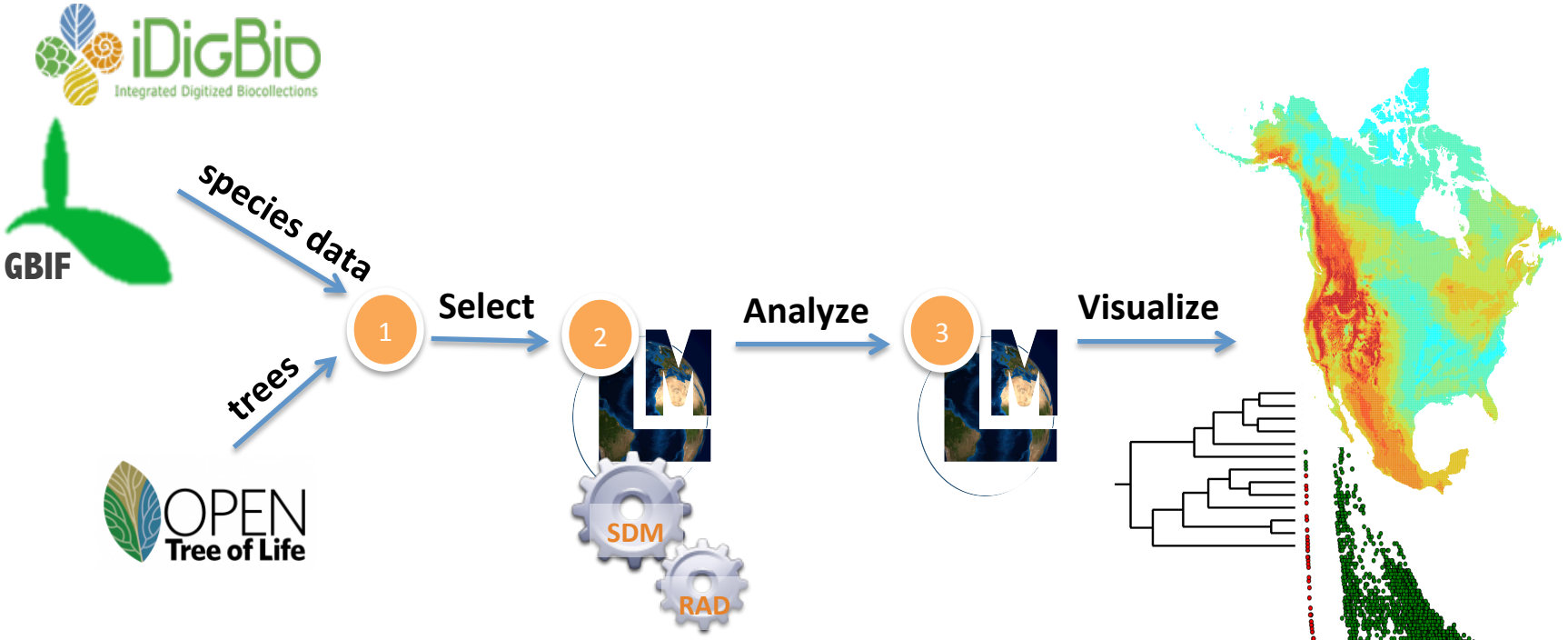
- ecological niche modeling
- multi-species range and diversity analysis
- visualization

Architecture





Lifemapper Pipeline

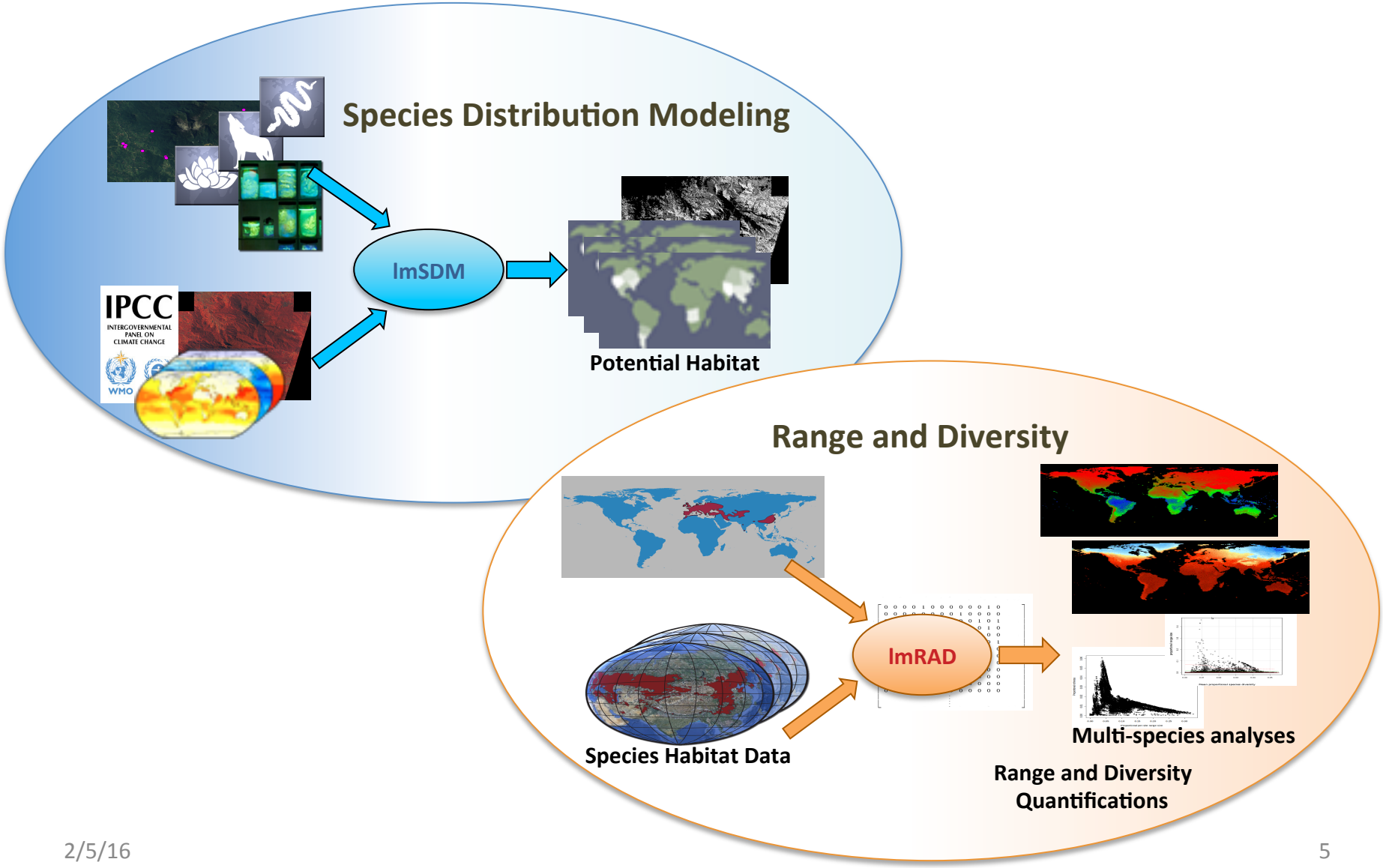


Multi-species data:

- measure macro-ecological shifts over time
- quantify and qualify biodiversity

Linked maps, trees and plots

Lifemapper Pipeline Tools



Lifemapper Virtualization

1. Software packaging as a Rocks roll
2. LmCompute virtualization
3. LmServer Virtualization
4. Using Different Virtualization Technologies

Software packaging as a Rocks Roll

Move Lifemapper installation to Rocks clusters + **Create a build process** = **Application Deployment**

1. Minimize cluster startup time:

- physical or virtual cluster
- efficient, programmable, configurable

2. Cluster state:

- known state
- known modular software stack

3. Cluster configuration:

- database,
- web server and
- job scheduler

1. Fast turn around

- from software updates to server availability

2. Full refactoring of Lifemapper software stack

- modularize

3. Rocks rolls:

- automate software build and install

Rocks

lifemapper-compute

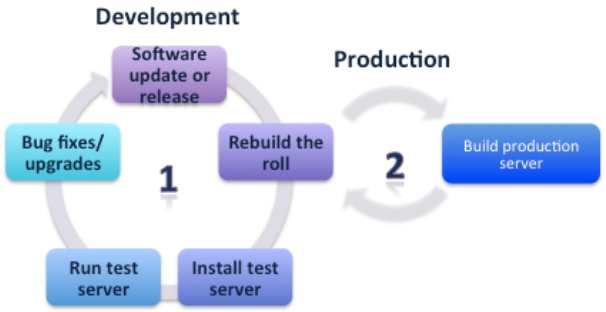
lifemapper-server

- Portable, can share
- Fast installation, configuration, update

LmCompute Virtualization

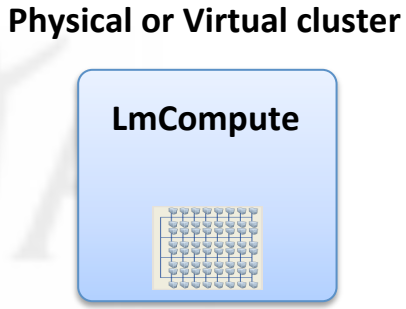
First step: separate the Lifemapper components and deploy LmCompute as a virtual cluster at SDSC.

1. Created *lmcompute* roll
2. Setup *lmcompute* instances



Result:

- roll installs all the prerequisites and lmcompute software
- configures the cluster to use a specific lifemapper server
- reduces the cost of installing, configuring and replicating the LmCompute component
- drastically reduce the time spent on software build and configuration
- automated nearly all hands-on tasks
- building and testing a new compute resource became trivial



LmServer Virtualization

Challenge:

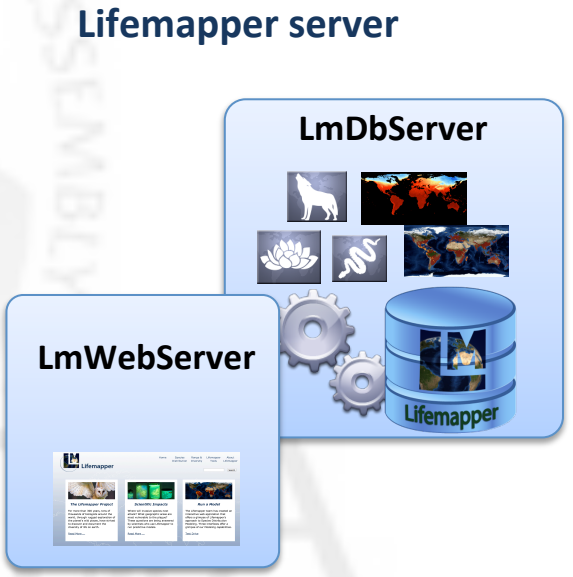
- Lifemapper project considers more efficient data storage and query,
- Need to experiment with different physical disks, dataset organizations and layouts, and file formats.
- Require a few instances of a portable and reproducible LmServer to test under various conditions.

Needs:

1. Portable Lifemapper server for UF to compute high quality species models using restricted satellite data.
2. Other data aggregators would benefit from their own install of Lifemapper to use with their specific data

Result:

- decouple webserver and dbserver from KU-specific implementation
- Lifemapper-server roll
- end-to-end build, install and configure process
- automates the entire lifecycle of application management
    1. fast software updates or rollback
    2. simple packaging and reliable robust deployment
    3. VM provisioning where building a virtual host is no different than building a physical host
    4. a hardened installation process
    5. full integration with the underlying cluster via customizable configuration files.
- automated Lifemapper server data and metadata seeding



Using Different Virtualization Technologies

Advantages for VBE

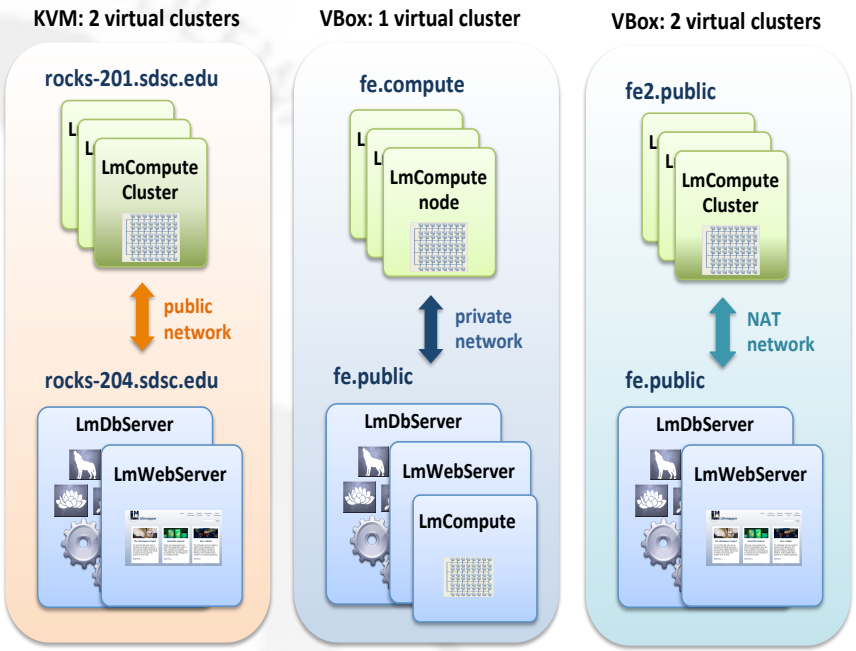
KVM

- larger instance sizes that are limited only be the hosting hardware specification
- long lasting instances used by multiple external clients
- dynamic input data
- multiple virtual clusters
- dynamically grow clusters based on computational needs.

VirtualBox

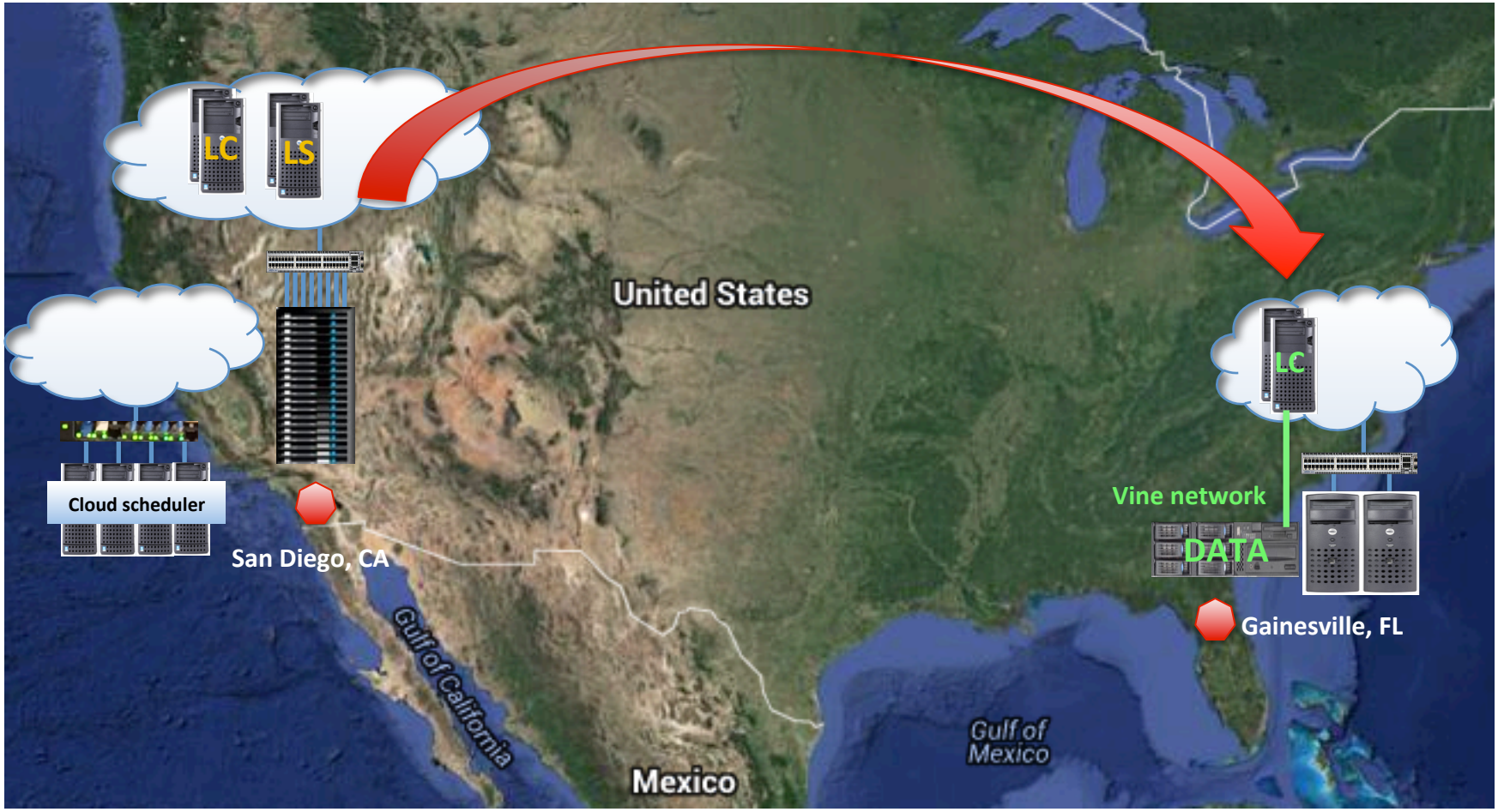
- Can have special-purpose instance
- Can have short-lived instance
- Pre-defined unique input data
- Intended for field work (with no network connection) and teaching tool
- instantiation of virtual cluster ready-made images can be accomplished in very few steps.

Virtual cluster scenarios for VBE

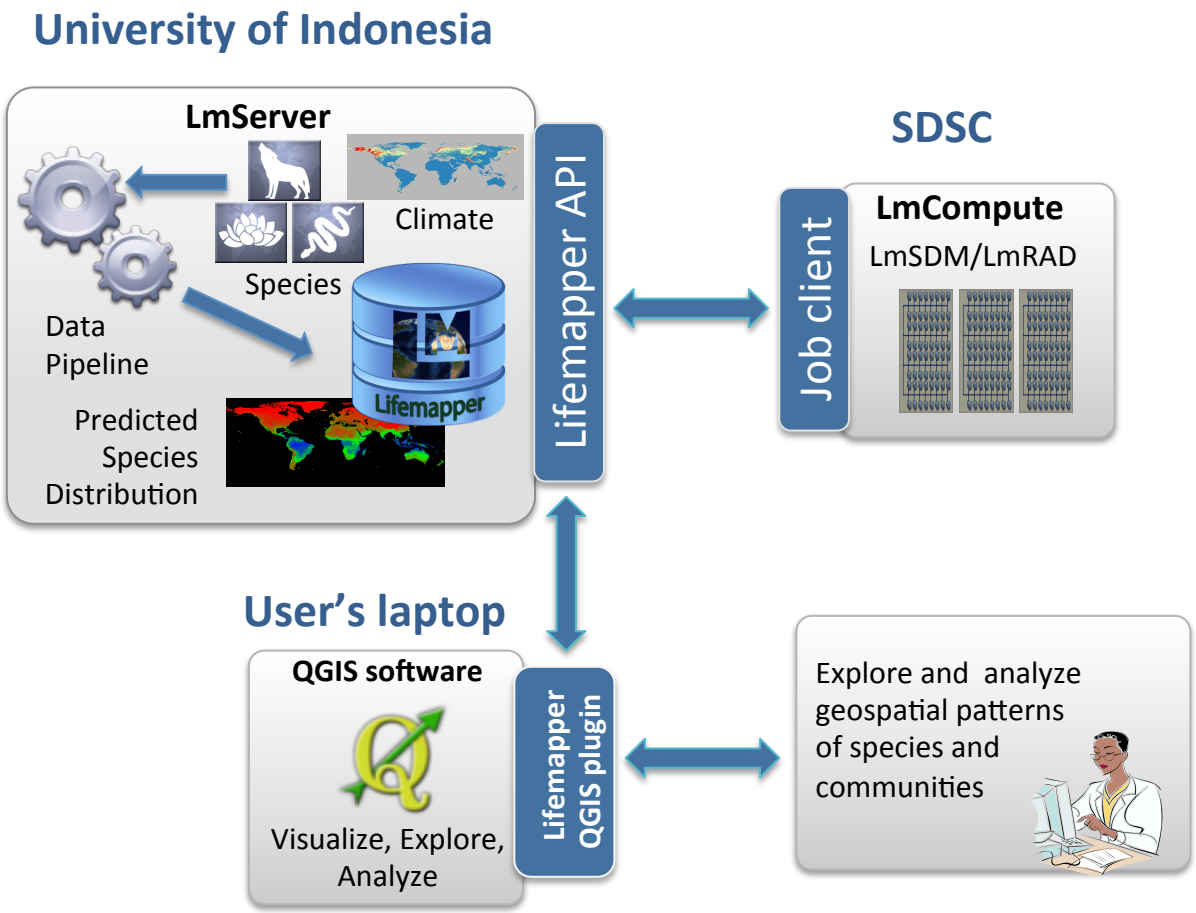


Distributed Computing & Geographically Restricted Data Resource

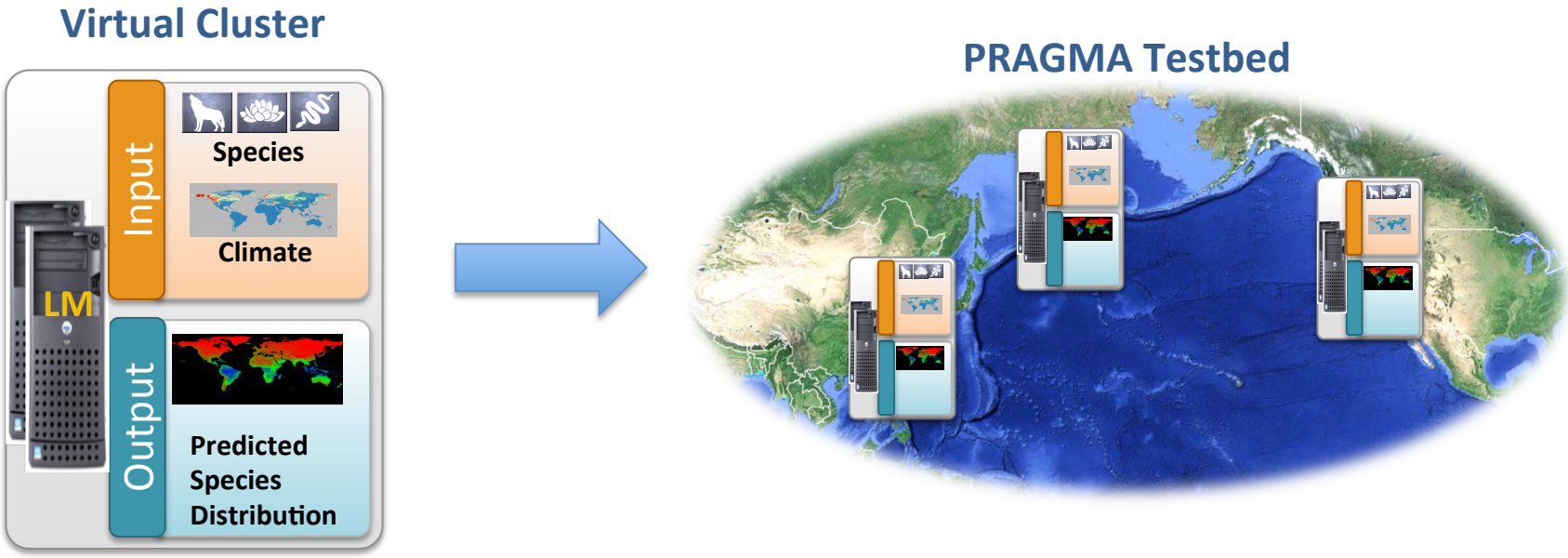
The infrastructure needed to make it work



Lifemapper on PRAGMA Testbed



Lifemapper and RDA



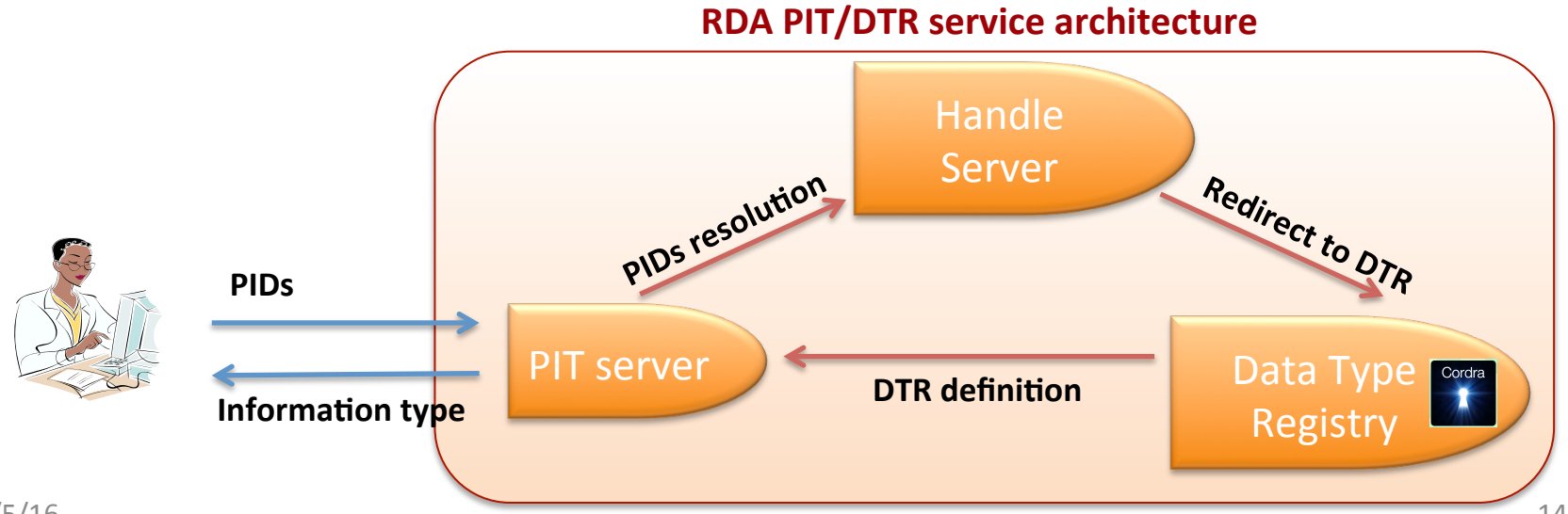
Data Enabling PRAGMA testbed

1. Can we provide for Scientific Data
   - Validation
   - Access
   - Exploration
2. Can we determine trustworthiness of projection data objects through detecting the mutation of Lifemapper VM instances?

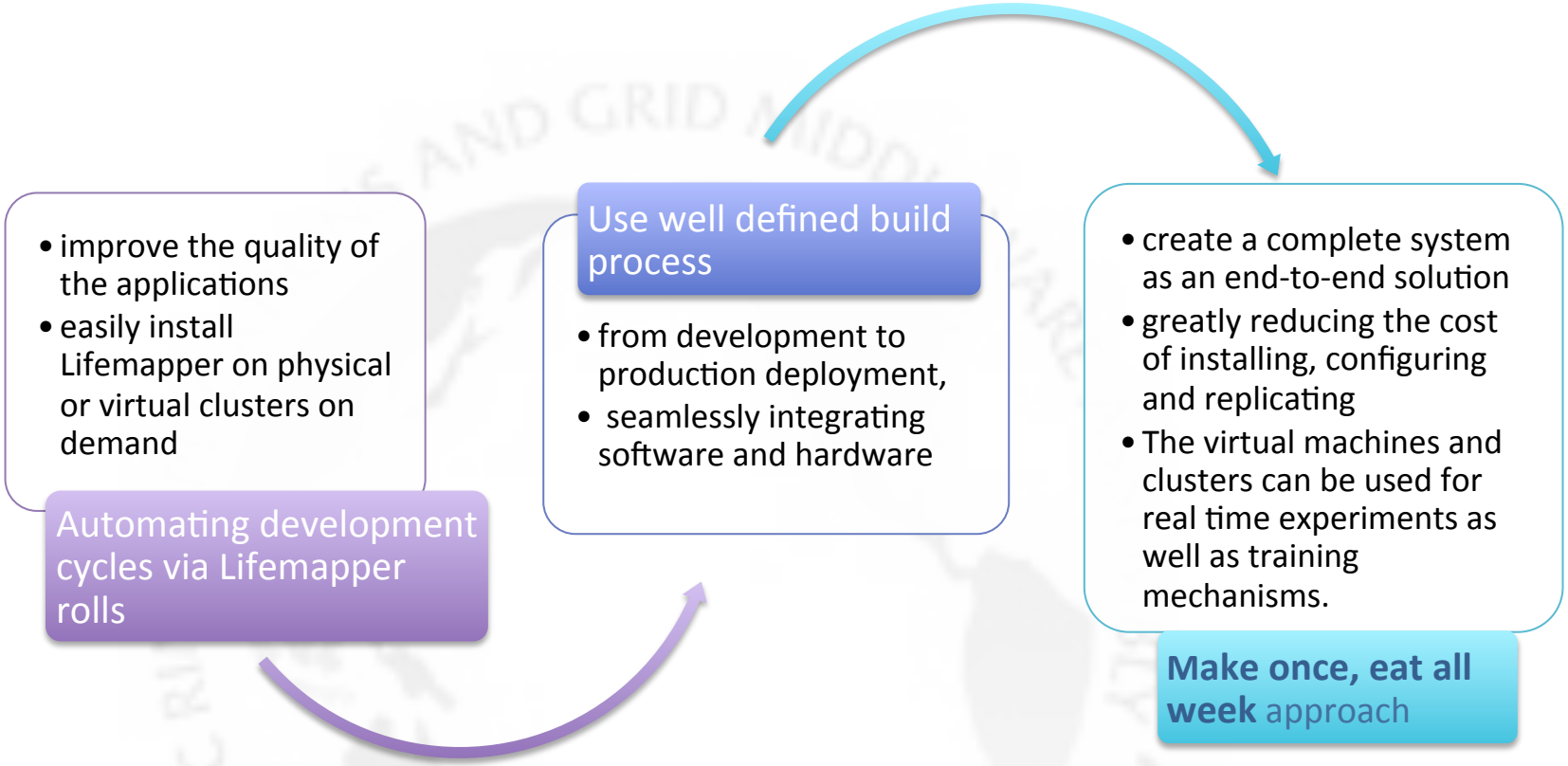
Lifemapper and RDA (cont'd)

Use Lifemapper Virtual Machine on PRAGMA testbed as a test case :

- Evaluate two RDA services, **Persistent Information Type (PIT)** and **Data Type Registry (DTR)** and link results back to RDA
- Assign **Persistent Identifiers** and **Information Types** to
  - LM biodiversity objects
  - VM instances



Summary


• improve the quality of the applications
• easily install Lifemapper on physical or virtual clusters on demand
Automating development cycles via Lifemapper rolls
Use well defined build process
• from development to production deployment,
• seamlessly integrating software and hardware
• create a complete system as an end-to-end solution
• greatly reducing the cost of installing, configuring and replicating
• The virtual machines and clusters can be used for real time experiments as well as training mechanisms.
Make once, eat all week approach

Related Links

Rocks clusters

https://github.com/rocksclusters

Lifemapper code

https://github.com/lifemapper

Lifemapper rolls

https://github.com/pragmagrid/lifemapper-compute

https://github.com/pragmagrid/lifemapper-server

Lifemapper in VirtualBox

https://github.com/pragmagrid/cloud/tree/master/VirtualBox

Cloud Scheduler

http://fiji.rocksclusters.org/cloud-scheduler

RDA-PRAGMA landing page Lifemapper

http://pragma8.cs.indiana.edu:9002

Future Work

- Lifemapper code modularization
  - Simplify data initialization and population
  - Formalize requirements for fully described data allowing easy use of different input datasets (iDigBio, GBIF, BISON, individual scientist's dataset) and switching among them.
  - Extend the pipeline to enable multi-species pattern analyses on the instance populated with data for Mt. Kinabalu
  - Create new modules to enable batch processing, editing pipeline workflows, spatial queries and archive subsets for dynamic microecological analysis.
- Create infrastructure bridging Indonesia and other PRAGMA sites
  - Setup a dedicated server in Indonesia (done)
  - Set up pipeline between Indonesia and other sites (ex: UFL with restricted satellite data)
- Lifemapper in the field:
  - Laptop installation of both components in single VC using mounted data
  - Identify optimal setup (memory/disk) to allow working with different datasets, crucial for virtual cluster on a laptop.
- Build on advances in overlay network in the PRAGMA ENT (iPOP & ViNe):
  - Incorporate different networking scenarios in the Lifemapper virtual infrastructure for accessing specialized data

Acknowledgements

This work is funded in part by National Science Foundation and USGS grants

PRAGMA

US NSF 1234953

Lifemapper

USGS BISON G14AC00285

US NSF BIO/ABI 1356732

US NSF BIO/ABI 1458422

Rocks

US NSF OCI-1032778

US NSF OCI-0721623

iDigBio

US NSF EF-1115210