



RDA-PRAGMA Sprint Biodiversity Model Replay



Quan (Gabriel) Zhou^[1], Aimee Stewart^[2], Nadya Williams^[3], Jason Haga^[4], Beth Plale^[1]
Indiana University Bloomington^[1] / Kansas University^[2] / SDSC^[3] / AIST^[4]

Introduction

In this RDA-PRAGMA sprint, we installed a couple tools emerging from the Research Data Alliance that give us the means to assign and track datasets and VM rolls by their unique PIDs. We use Lifemapper from PRAGMA cloud as our application case. With this simple functionality, plus some data management modeling in support of versioning of datasets and VM rolls [1], we provide functionality to users:

- Assign PIDs to biodiversity data objects and VM instance for long-term findability and accessibility;
- Create data information types from RDA PIT/DTR output to improve data interoperability;
- Help users to determine whether to use an analysis output against mutation of data source and software/hardware stack.

RDA PIT/DTR Service

Identity and interoperability of PIDs resolved among different systems and providers is needed. RDA PIT working group allows providers agree on a common API [2], register their information types in a common data type registry [3] and agree on some core types, such as Lifemapper projection set and VM instance with useful metadata units.

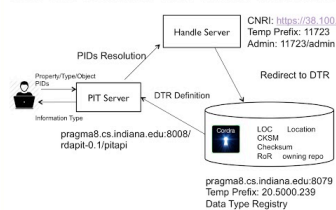


Fig. RDA DTR/PIT Service @ PRAGMA IU

Data Validation Service

To valid trustworthiness of Lifemapper SDM projection set, we consider the mutation of input occurrence set and LM VM instance such as possible updates.

Challenges Overcome:

- Input datasets and generated products of Lifemapper are embedded within the Lifemapper VM;
- Generated data products must be exposed and persisted outside the Lifemapper VM.

Design Features:

- Use landing page with metadata repo/file archive to persist generated Lifemapper data products;
- Validate data object against latest VM instance and compare checksum to determine its trustworthiness.

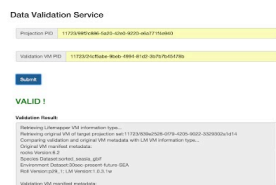


Fig. One sample Lifemapper projection set validation output

Framework

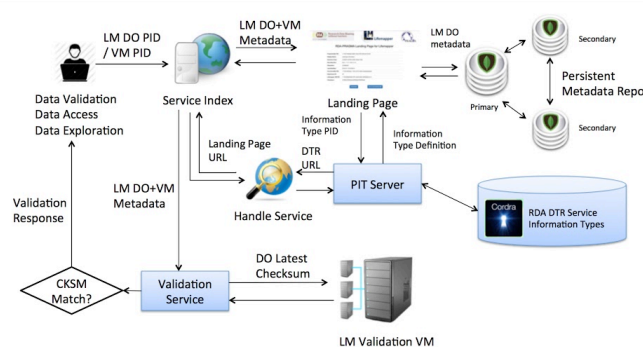


Fig. User workflow for biodiversity data object exploration, access and validation

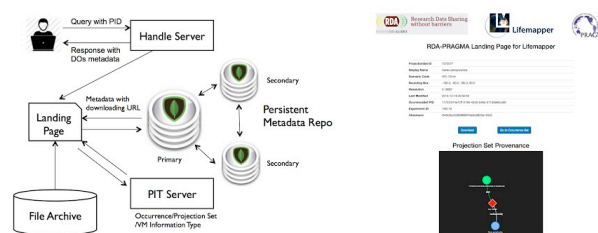


Fig. Landing Page Framework and one sample landing page of Lifemapper Projection Set

Future Work

- Pull input datasets out of the Lifemapper VM and make the Lifemapper VM automatically responsive to changes to the input data sources such as GBIF and iDigBio;
- Provide functionality for users to automatically bootstrap Lifemapper VM instance in order to determine data object trustworthiness against the mutation of VM versions and physical environment settings;

REFERENCES

- [1] Code Base Github URL: <https://github.com/Gabriel-Zhou/LMLandingpage>
- [2] RDA Persistent Information Type WG: <https://rd-alliance.org/groups/pid-information-types-wg.html>
- [3] RDA Data Type Registries WG: <https://rd-alliance.org/groups/data-type-registries-wg.html>

Acknowledgement

Funded in part through a grant from the National Science Foundation, grant #OCI-1234983.