Provenance Data Harvest and Scientific Results Reproducibility

BIBI RAJU
Pacific Northwest National Laboratory
ESGF 2017, San Francisco, CA
Provenance Environment (ProvEn)

- **ProvEn** is a provenance management platform consisting of loosely coupled components supporting the disclosure, storage, and access to provenance information.

- **Describe Anything Provenance Interface API (DAPI)**
  - ProvEn’s provenance disclosure library. Scientific workflow applications instrumented with DAPI can produce and disclose their provenance data.

- **Provenance Cluster**
  - ProvEn’s scalable approach for collecting concurrent provenance data streams from DAPI sources.

- **Hybrid Store**
  - ProvEn combines system level metrics (Metric Store) with the traditional disclosed provenance (Semantic Store) to create an extended provenance view.
ProvEn’s New Features

▶ **HArvester Provenance Interface (HAPI)**  *
  - ProvEn’s harvester library that is capable of extracting already existing file based information produced by applications
  - HAPI uses scruffy provenance content as basis for messages:
    - Tabular data
    - Parameter list
    - Large objects
  - Uses schemas, identifiers, and references to other content to support enrichment.

▶ **Interfaces**  *
  - Developed alpha release portal tied together with Jupyter notebook, Swagger, and REST API and SPARQL endpoint offering a wide range of client side access to provenance
Harvester Provenance Interface (HAPI)

- Extract existing information produced by applications
- Transform the information to HAPI syntax inspired by W3C CSV on the web recommendations
- Pre-stage provenance information into provenance messages
- Write provenance messages into ProvEn store
- Use the retrieved provenance information for
  - scientific results reproducibility
  - scientific results explanations
  - comparing two simulations
- HAPI is a generic format and can be applied to harvest provenance from relational database tables as well as other scientific applications that log provenance related information
- Supports alignment to community vocabularies.
  - Uses W3C PROV for traceability
**Use Case: Energy Exascale Earth System Model (E3SM)**

- **Focus:** Recovering enough information to re-execute a given simulation in the future

- **Steps**
  - Run a simulation
  - Crawl through simulation data to extract relevant pieces of information
  - Run harvester to store the extracted information in ProvEn database
  - Retrieve captured information from ProvEn database
  - Use the retrieved information to re-execute the simulation with the same set of initial conditions, input parameters and settings on the same machine
Why is Reproducibility Possible in E3SM?

E3SM provides a systematic way initializing a directory tree, configuration files, file-based input settings, and run scripts that serve as a base line for any E3SM simulation run.

Simulation code uses configuration control (github) and versioning to manage changes source code, scripts, and new software releases.

Input files, configuration settings, and scripts were human readable and were easily decipherable.

Example Artifacts
- git hash of the E3SM code
- Machine and compiler details
- Input parameters
- simulation compset and resolution
- configuration XML files

```
Simulation name: Try1.Run1.ne4_ne4
Compset and resolution: FC5AV1C-L
ne4_ne4
ACME Github hashkey: v1.0.0-beta.1-8397-g0af35b6
Machine: edison
Compiler: intel
ACME Script Name: run_acme.template.csh.2017-08-04_15:15:44_PDT
```

SUMMARY of cprnc:
A total number of 305 fields were compared
of which 0 had non-zero differences
and 0 had differences in fill patterns
A total number of 16 fields could not be analyzed
A total number of 0 fields on file 1 were not found on file2.
diff_test: the two files seem to be IDENTICAL
### ProvEn Interfaces

- **Developed alpha release ProvEn portal** that allows visualization of the captured provenance data and Swagger interface for client side access to provenance
- **Jupyter notebook interfaced with ProvEn Portal** to support desktop analysis
- **REST interfaces** allow any HTTP enabled client to access time series or semantic information

HAPI Message Harvested, Transformed to JSON-LD

<table>
<thead>
<tr>
<th><strong>ProvEn Environment (ProvEn) REST Services</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>REST based services providing access to ProvEn's hybrid (Semantic + Time-Series) data repository</td>
</tr>
</tbody>
</table>

#### Repository

- **GET** `/repository/concepts/{domain}` (pattern)
- **GET** `/repository/concepts/{type}/{domain}`
- **POST** `/repository/influxql`
- **GET** `/repository/messages/{domain}`
- **GET** `/repository/messages/{domain}/messageName`
- **GET** `/repository/messages/{domain}/conceptsId`
- **POST** `/repository/spark`
- **GET** `/repository/statements`
- **DELETE** `/repository/statements/{domain}`
- **GET** `/repository/statements/{domain}`

Jupyter notebook can be used to query provenance and metrics
Impact to ESGF

- ProvEn helps in the ESGF domain to maintain
detailed history information about the steps followed during data publishing, processing and movement
provenance of data products and of the workflows that derive these products and their executions
- Capture provenance in various projects (e.g. CMIP6) for reproducibility
- Extract provenance from projects that already capture provenance
- ProvEn repository could be hosted by those who lack a provenance solution.
- ProvEn is open source (MIT license)
Acknowledgements

- Eric Stephan, Todd Elsethagen - Pacific Northwest National Laboratory
- Project Acknowledgements
  - Integrated End-to-end Performance Prediction and Diagnosis for Extreme Scientific Workflows (IPPD) Project. IPPD is funded by the U. S. Department of Energy Awards FWP-66406 and DEC0012630
  - Energy Exascale Earth System Model (E3SM) project funded by the Office of Biological and Environmental Research (BER) in the U.S. Department of Energy (DOE) Office of Science.