Provenance integration in R seasonal-to-decadal verification workflows

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Escola Tècnica Superior d'Enginyeria de Telecomunicacions de Barcelona

Author: Albert Puiggròs Figueras

Tutors: Victòria Agudetse Roures, Gladys Utrera

Abstract

In climate research, data integrity and reproducibility are critical. This report details the implementation of **SUNSET** (SUbseasoNal to decadal climate forecast post-processing and asSEssmenT suite), based on the **METACLIP** (METAdata for CLImate Products) framework, to manage **data provenance** in climate product verification. The goal is to create a system that tracks and manages data provenance, providing a clear and traceable history of data sources, transformations, and final products. This integration aims to enhance provenance across the entire SUNSET climate verification workflow

Objectives

1. Incorporate **METACLIP provenance framework into SUNSET** by embbeding all the provenance information in the R code.
2. Adapt SUNSET outputs to the **METACLIP web interpreter** (web-based interactive front-end for metadata visualization) to better display the provenance information produced by the workflow.

Data provenance

Data provenance documents the origin and transformations of data to ensure **transparency**, **reproducibility**, and **trustworthiness**. It is key for verifying data authenticity and consistency. W3C frameworks using **RDF**, **OWL**, and **PROV** languages help implement data provenance, improving data integration and search within the **Semantic Web**. **Ontologies** provide a common vocabulary and standardized data relationships.

STATE OF THE ART

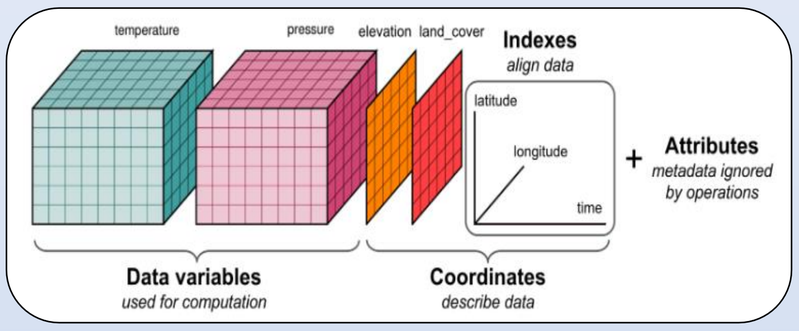


Figure 1: s2dv_cube structure

SUNSET

SUNSET, developed by the Barcelona Supercomputing Center (BSC), is a tool for **verifying climate forecasts on subseasonal to decadal scales**. Used by climate scientists, it evaluates models for climate services. It has a **modular structure** that allows customizable workflows. Users define operations in a **recipe** template. SUNSET processes data in objects that include observational and model data. Also, it integrates several R packages from BSC for climate data processing.

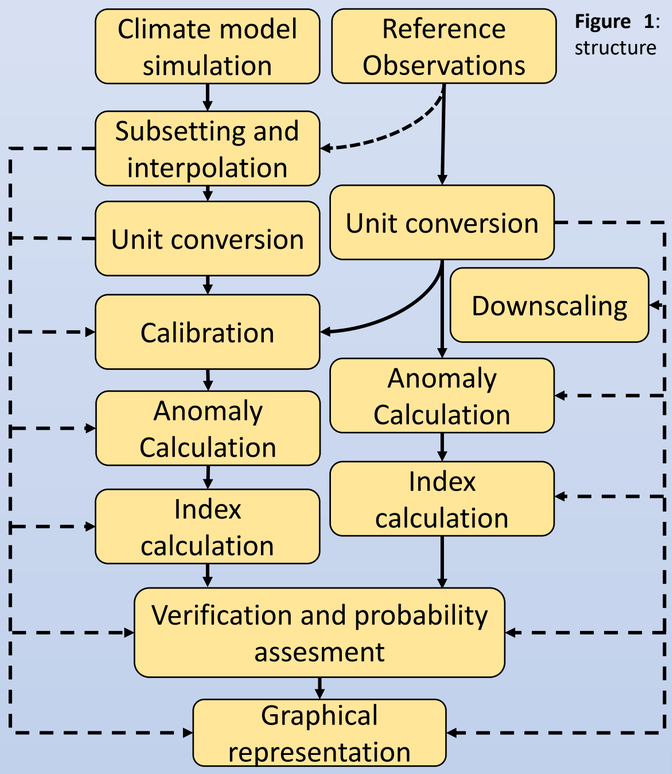


Figure 2: Example structure of SUNSET: Its modularized design allows users to dynamically execute the modules in diferent orders, enabling a wide range of post-processing and verification of possibilities.

METACLIP

METACLIP (Metadata for Climate Products) manages and tracks climate data provenance using semantic web standards. It employs specialized ontologies to link climate data to its provenance, enabling users to trace data origins. Incorporating various well-known ontologies. It integrates with the R programming environment via **metaclipR.**

METACLIP ontology

Provides specialized **vocabularies** to describe climate data products

DEFINED ONTOLOGIES

- Datasource (**ds**): Source of the input data: dataset descriptions and transformations
- Calibration (**cal**): Bias correction and other statistical operations
- Verification (**veri**): Validation operations
- Graphical output (**go**): Graphical products

- Resource Data Framework → **RDF**
- World Ontology Language → **OWL**
- Ontology for Provenance → **PROV**

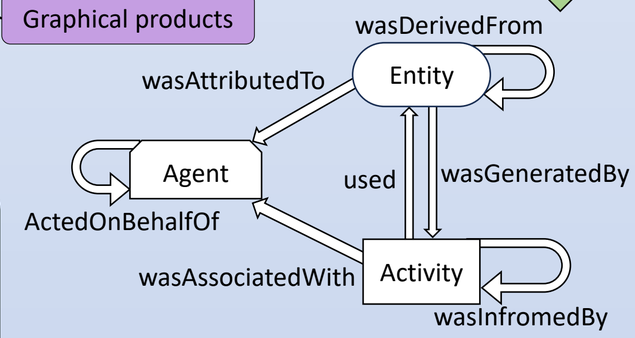


Figure 3: PROV-O core element representation

PROVENANCE IMPLEMENTATION

Dataset (s2dv_cube)

Recipe

Provenance in-house functions

We decided to develop **in-house functions for SUNSET provenance**. We created **SUNSET_PROV** functions to integrate seamlessly with SUNSET and comply with the METACLIP ontology. These functions allow for full provenance, adding complex relations and annotation properties. SUNSET_PROV functions are **modular**, work within SUNSET modules, and generate comprehensive provenance graphs, incorporating command calls and detailed dataset information.

Provenance data

The provenance record is stored in **JSON** format and embedded within the output images generated by the workflow. The provenance record can be also obtained directly as a *JSON* file without the need of generating plots.

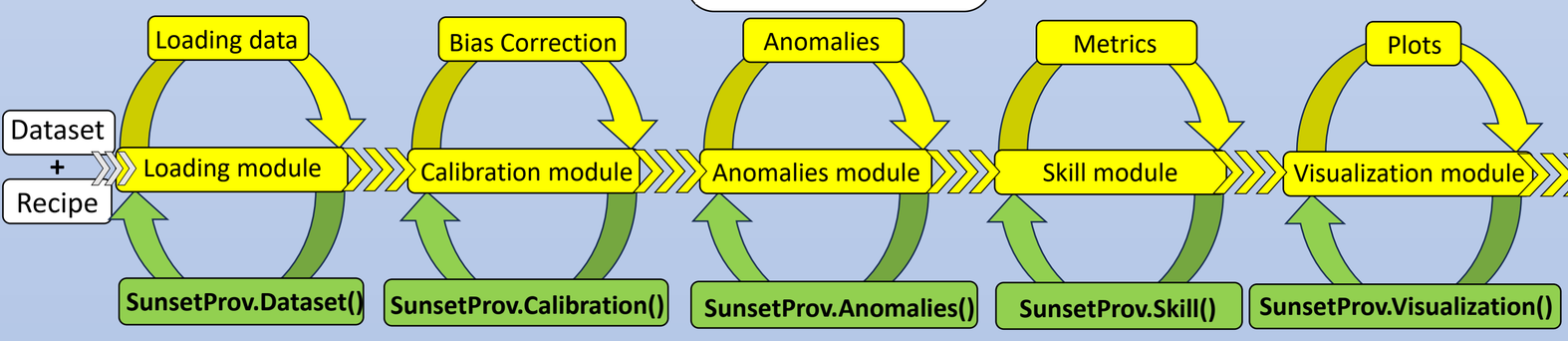
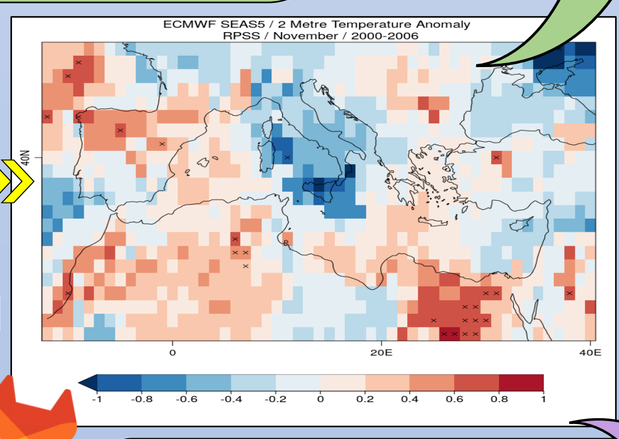


Figure 4: Schematic representation of unit test 1. It is a simple atomic execution of SUNSET, running the Calibration, Anomalies, Skill and Visualization modules. It loads seasonal hindcast and observational data from ECMWF (European Centre for Medium-Range Weather Forecasts) and ERA5 respectively.



METACLIP Interpreter

The **METACLIP interpreter** is a web tool designed to interactively visualize RDF representations in several JSON format.

Figure 5: RPSS 8-metre temperature anomaly. Out put of unit test 1

"SUNSET_PROV"

Functions from the **SUNSET_PROV** set can be classified into three types. First, the "**Prov()**" functions generate provenance graphs specific to each SUNSET module. The "**SunsetProv()**" structure the "Prov()" functions for each module and the operation or transformation taking place within it. Additional functions were developed as tools to support provenance generation and conversion.

Unit tests

Unit tests is designed to either test specific SUNSET_PROV functions or cases for a dataset loading or certain transformations, verifying the structure of the generated provenance graph at each step of the workflow, including node names, node positions, node classes, and other aspects.

GitLab repository: Code, examples and unit tests

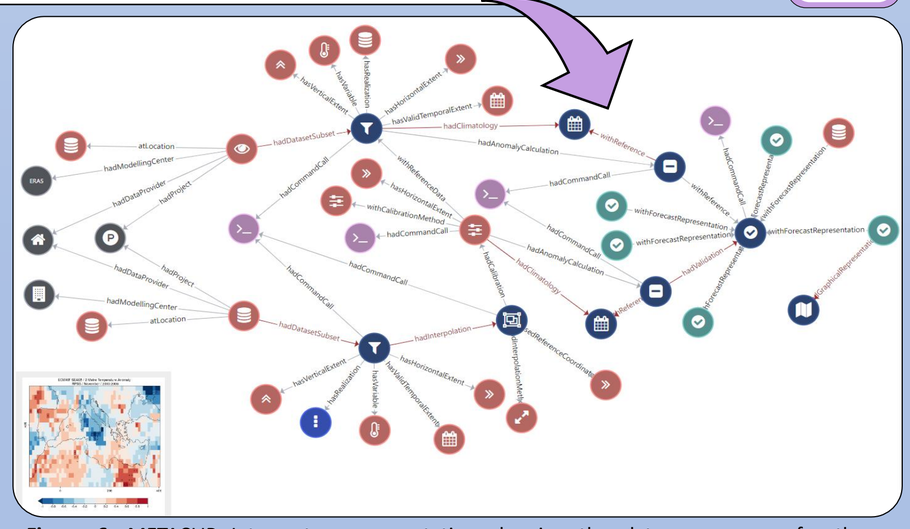


Figure 6: METACLIP Interpreter representation showing the data provenance for the execution of unit test 1.

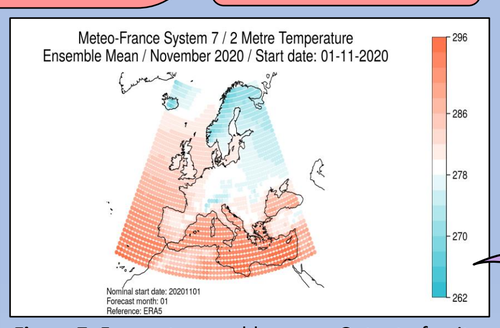


Figure 7: Forecast ensemble mean. Output of unit test 5

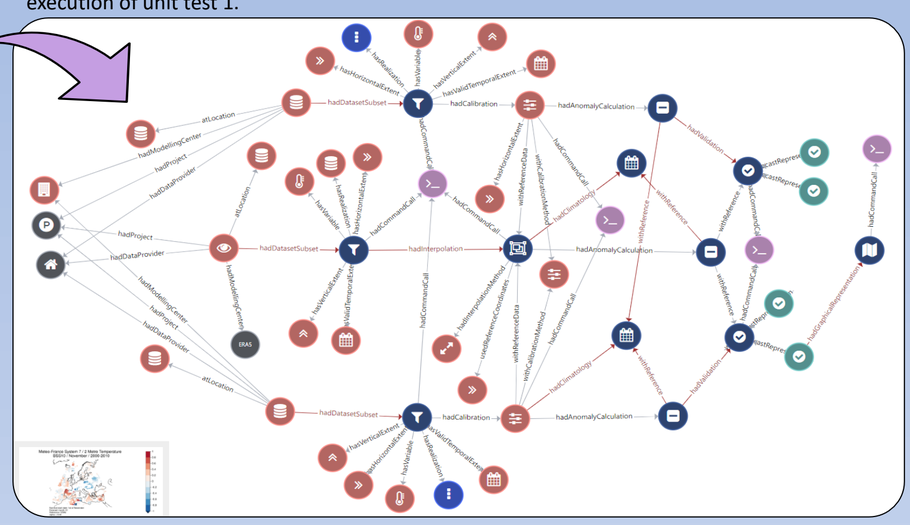


Figure 8: METACLIP Interpreter representation showing the data provenance for the execution of unit test 5.

SUNSET MODULE

Figure 9: Structure of the provenance functions developed (SUNSET_PROV), showing how "SunsetProv()" functions integrate in SUNSET and structure de "Prov() executions".

ONTOLOGY EXPANSION

We contacted METACLIP developers via **GitHub** and made a pull request to include a new "**cal:Downscaling**" class in the Calibration ontology.

PULL REQUEST

GitHub