CAESAR
Space Weather archive prototype for
ASPIS

Marco Molinaro [INAF]
on behalf of

NODE 2000 Team
Marco Molinaro (INAF), Valerio Formato (INFN), Valeria di Felice (INFN), Dario Del Moro (UNITOV), Cristina Campi (UNIGE), Federico Benvenuto (UNIGE), Carmelo Magnafico (INAF), Rossana De Marco (INAF), Ermanno Pietropaolo (UNIAQ), Giancarlo de Gasperis (UNITOV), Luca di Fino (UNITOV), Gregoire Francisco (UNITOV), Igor Bertello (INAF), Andrea Tacchino (UNIGE), Alessandro Perfetti (INAF)

EB Team
Anna Milillo, Giuseppe Sindoni, Marco Giardino, Monica Laurenza, Dario Del Moro

Project Prime:

Project Partners:

ADASS XXXII
Oct 31 - Nov 4 2022

CAESAR for ASPIS

CAESAR

(Comprehensive spAce wEather Studies for the ASPIS prototype Realization)
aims to

- tackle all the relevant aspects of Space Weather science
- realize the prototype for **ASPIS**
  - (ASI SPace Weather InfraStructure)
 - the scientific data centre for Space Weather of the Italian Space Agency (**ASI**)

It involves a great part of the **Space Weather Italian community**
bringing together 10 Italian institutions as partners

- 3 national research institutes
  - **INAF, INGV, INFN**
- 7 universities
  - **Tor Vergata (Rome II), Perugia, Genoa, L'Aquila, Calabria, Catania, Trento**

CAESAR for ASPIS

CAESAR will adopt an approach to encompass the whole chain of phenomena from the Sun to the Earth up to planetary environments, selecting a number of well-observed "target Space Weather events" for detailed and comprehensive studies in order to showcase the proposed approach.
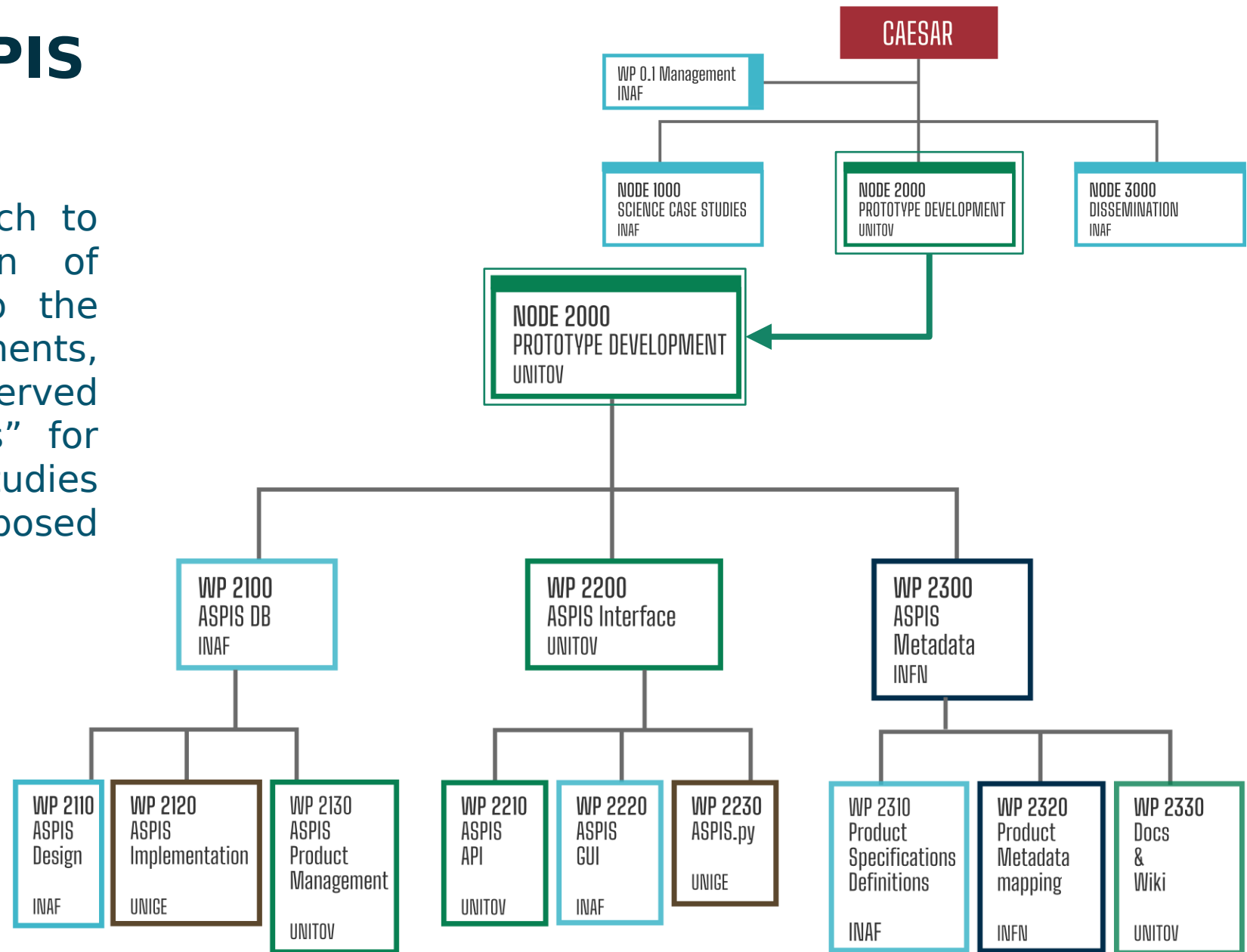
CAESAR NODE 2000 – ASPIS Prototype

The CAESAR ASPIS prototype implementation will be hinged on three main pillars
- Database [WP2100]
- Interface [WP2200]
- Metadata mapping [WP2300]

The ASPIS DB will contain mainly proprietary/co-proprietary products, with their relative data policy in a homogeneous, standardised collection of resources.
Other important external data shall be accessed through links to existent archives.

74 (75) main products were identified in the proposal, spanning calibrated data, derived ones, models and tools; including near-real time data and tools.

Currently collected sample data and descriptions cover about a half of the above, for a, finer grained, current total of more than 100 among data collections, data models and tools.

CAESAR NODE 2000 – HL WP
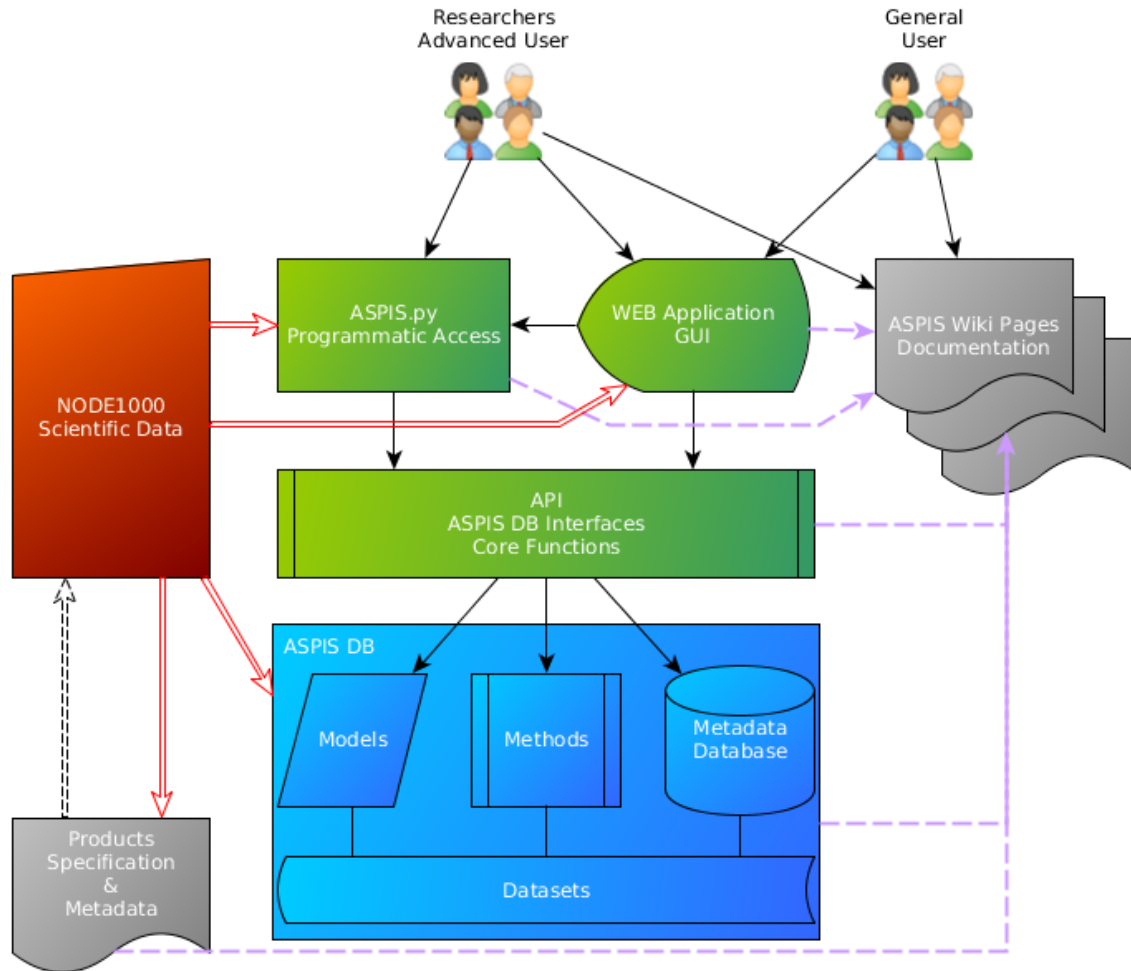
High Level Working Packages are responsible for

- **WP2100**
  - Designing the DB
    - Database in its general meaning
  - Implement it
    - Empty → preliminary→ final prototype
  - Ingest product data/models
  - Incorporate models/tools

- **WP2200**
  - Define the archive's API
  - Provide a GUI (available to all user)
  - Develop ASPIS.py (for advanced researchers)

- **WP2300**
  - Templating product descriptions
    - and collect them from NODE 1000
  - Map metadata content and formats
    - For internal/external usage
  - Document all the activities and processes

Project technical status – Product Specification

ProSpecT

CAESAR - Product Specification Template

CLEAR DATA   DOWNLOAD   UPLOAD

Welcome to the ASPIS/CAESAR metadata template form.
Please fill the form with all the relevant information regarding your products.

For instructions and help on this form, consider
reading the ProSpecT instructions document
viewing the video tutorial(s)
Full template explanation with data collection use case
Variant for software or numerical model use case(s)
or (if the above don't work)
contacting CAESAR NODE 2000

PRODUCT   CURATION   CONTENT & POLICY   DATA DESCRIPTION   FUNCTIONALITIES   >

Title *
TSST H-alpha Full Disk Images

Short name *
TSST-halpha

Type *
data

```
{
  "product": {
    "title": "UNITOV Catalogue of geoeffective CMEs",
    "shortname": "UNITOV-ICME",
    "type": "data",
    "identifier": "aspis:/unitov/icme",
    "altidentifier": [
      "doi:10.5281/zenodo.5516980"
    ],
    "status": "Active",
    "created": "2022-04-08T11:24:13+02:00",
    "updated": "2022-04-08T11:25:24+02:00"
  },
  "template": {
    "version": 1
  },
  "curation": {
    "publisher": "University of Rome Tor Vergata Dept. of Physics",
    "publisherID": "aspis:/unitov",
    "creator": [
      "Dario Del Moro",
      "Raffaello Foldes",
      "Gianluca Napoletano"
```

- JSONForms
- JSON metadata document
- VOResource-like
- Combine
  - Technical work requirements
  - Scientific user impact when collecting/curating metadata
  - Ease of use in prototype development

Project technical status – Database Design
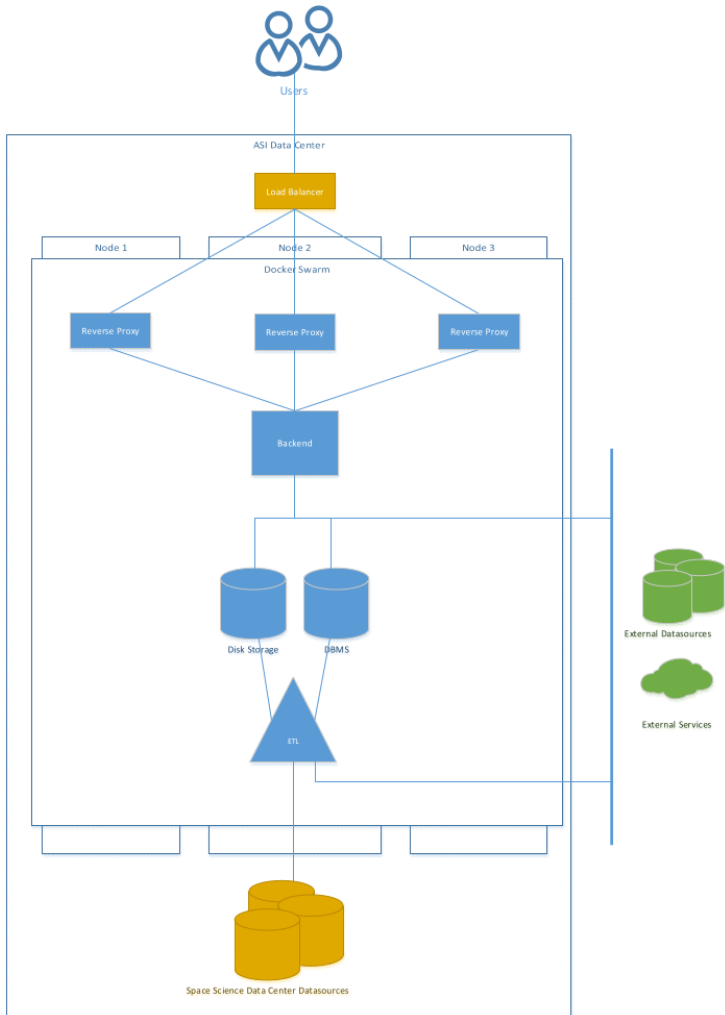
Scientific, functional and system requirements have been developed, starting from

- Products' specifications and samples
- Requirements on event/observational phenomena representations
- Technical solutions

A high-level architecture has been prepared and the design of the CAESAR ASPIS database, including the preliminary version of the database schema, empty and ready for the preliminary ingestion.

Evaluation on submitted data formats and metadata is ongoing to allow the transformation layer to populate the DB and attach models and tools.

Data & Metadata, Formats & Communities

- Homogenising submitted data and metadata always involves a critical balance
  - Lessen work to be done
  - Homogenise formats (dataset format, metadata content, …)
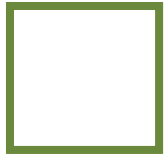- Find new ways to "standardise" following community needs

some examples ⬇ in CAESAR's case

- Date format and frame: from Sun to Martian surface
  - "interface" compromise
- Product/resource description: what spatial coverage means… ← **SPASE 2.5.0**
- **CDF** → Dataset formats
  - what do single groups use
  - What do the community would find acceptable

Interfacing and interoperability

CAESAR lives within the Space Weather community, that is attached to Solar Physics, Planetary Science and Heliophysics in general.

On the ASPIS side, GUI and internal solutions will be driven by project identified requirements, but the goal is also to have this archive connect, interoperate at global level.

We investigated (and still doing so) previous and existing approaches of a variegated community: HELIO, SOLARNET, EuroPlaNet, VSO, HAPI, SunPy, …

Now that we are at a first implementation step we wil have to understand:

- What frameworks/standards do we follow? …and why?

- How do we attach to them (ASPIS.py w.r.t. SunPy & HAPI)?