# SimDAL proposals

## Observatoire de Paris / VO-Paris Data Centre

David Languignon, Franck Le Petit

Poona, India

October 25, 2011



# Table of Content

# Introduction

This presentation is organized as follows :

1. Recall of the Theory Group context and SimDM
2. The question of accessing theoretical data
3. Proposals for a theory access layer

This presentation is intended to :

1. Recall the basis behind the Theory original goals to settle new projects foundation
2. Recall SimDM usage
3. Discuss the issues relating to the data access question
4. Do standard proposals

# SimDM usage

Through the last 3 SimDM implementations made in Paris

- the model is rich and flexible enough to address many theoretical cases (3D+time, micro physics)
- SimDM releases all its potential by coupling itself with a **Semantic layer**

# SimDM usage

But,

- because it is highly abstract, the SimDM is very hard to query when implemented as a relational scheme
- Implementations using RDBMS are heavy as they require time, knowledge and maintenance
- even with the TAP views facility provided by projects like VO-URP*, a lot of joins are required
- no mechanism to access data results

**We need**

an **access layer** for theoretical data

*http://code.google.com/p/vo-urp/

# Part 1 Summary

- The current missions of the Theory Group
- The SimDM project and its usage
- The need for an access layer which provides **simple query services** on top of **complex data models**

# The needs

## Use cases for the access layer for theoretical data

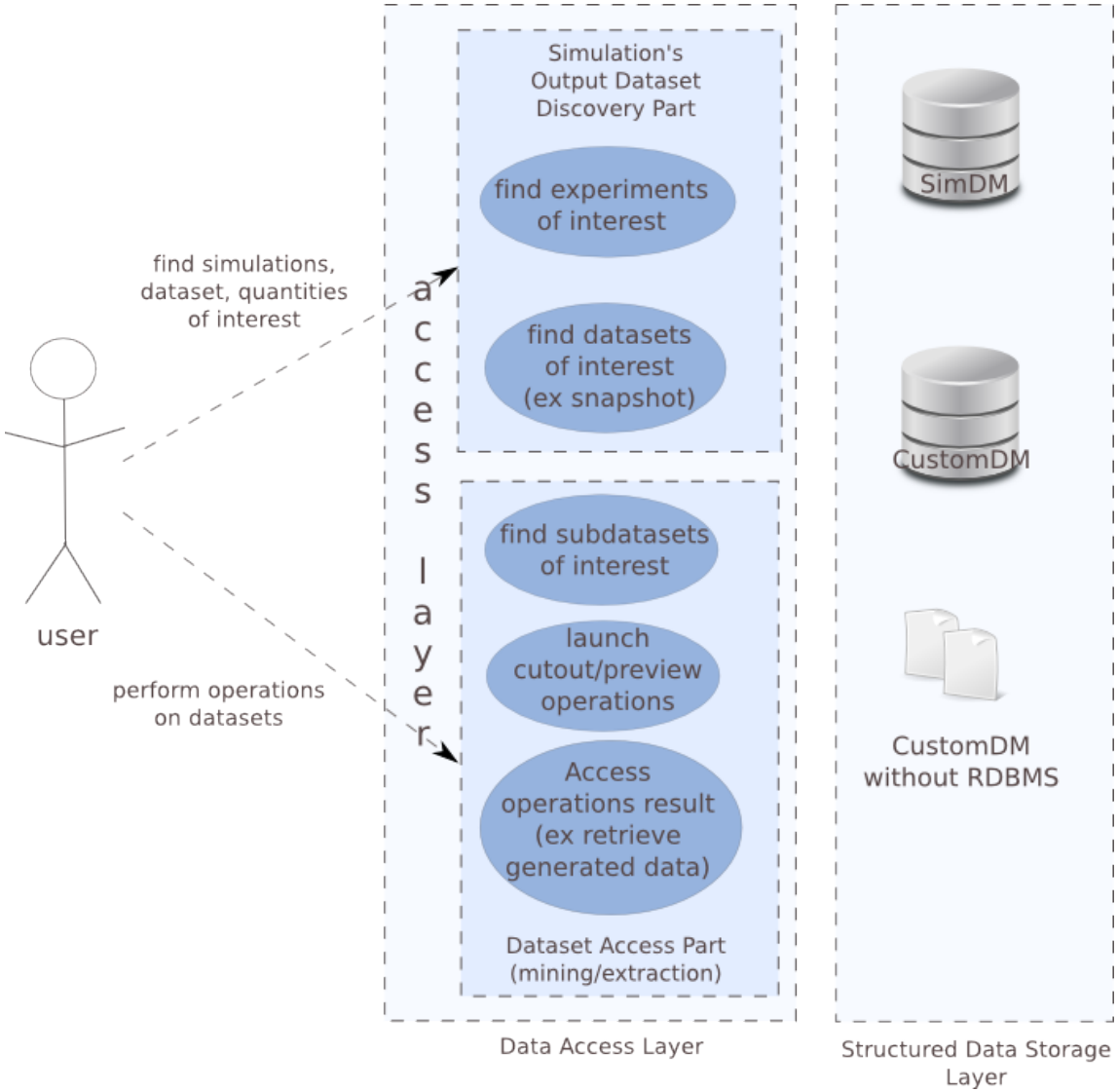
It is structured through the identified **data access use cases**, divided in 2 parts

1. data discovery
2. data mining/extraction

**Whatever is the underlying provider's data format** (SimDM implementation, custom DM, flat files...)

# The needs

## Use cases for the access layer for theoretical data



# The needs

## Missions of the access layer for theoretical data

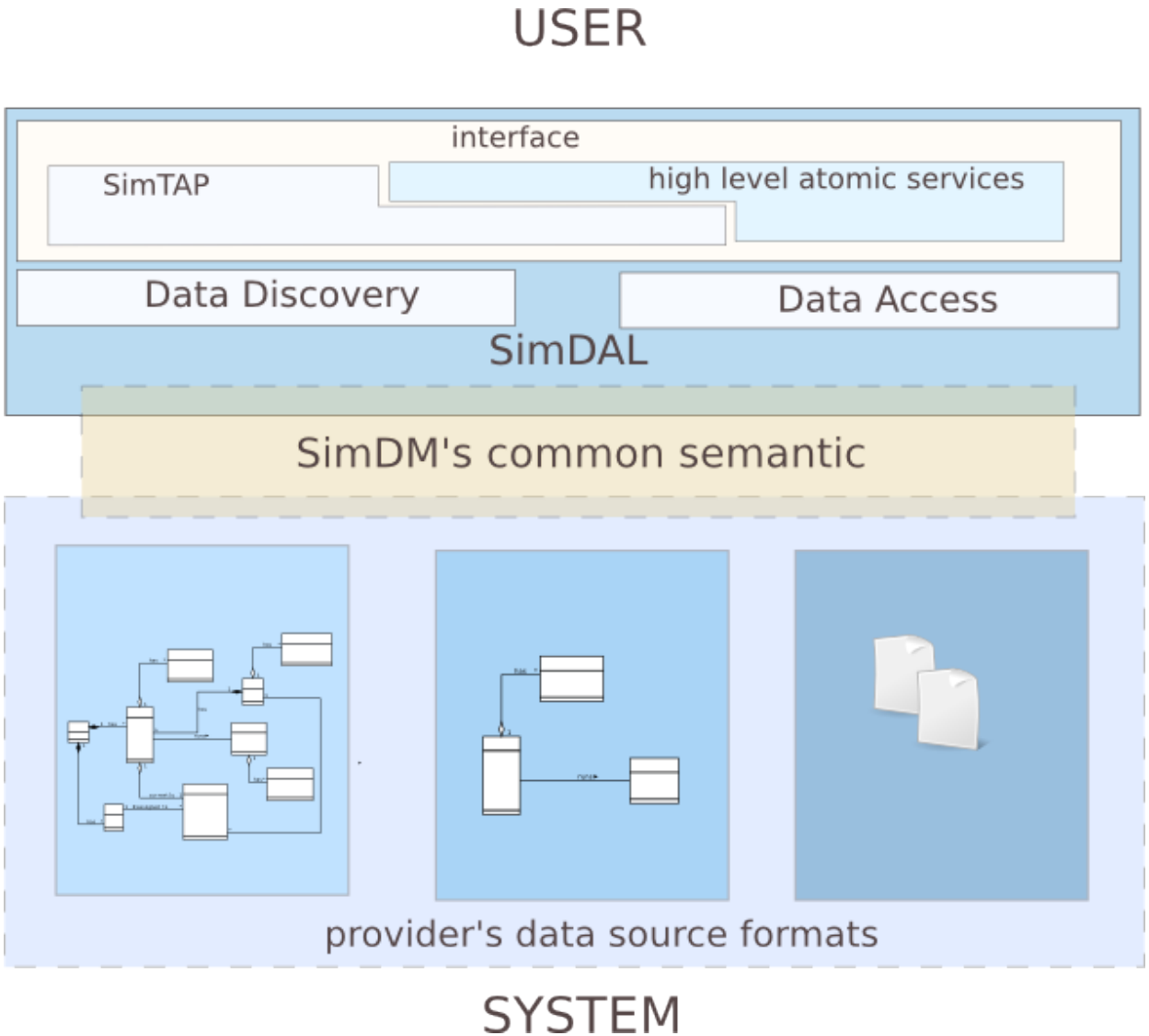
- **easy**, straightforward data discovery services
- ivoa DAL compliant
- **generic** enough to interface with all the provider's **data source formats**

**We propose**

to name this layer "Simulation Data Access layer", **SimDAL**, referring to the DAL standards family

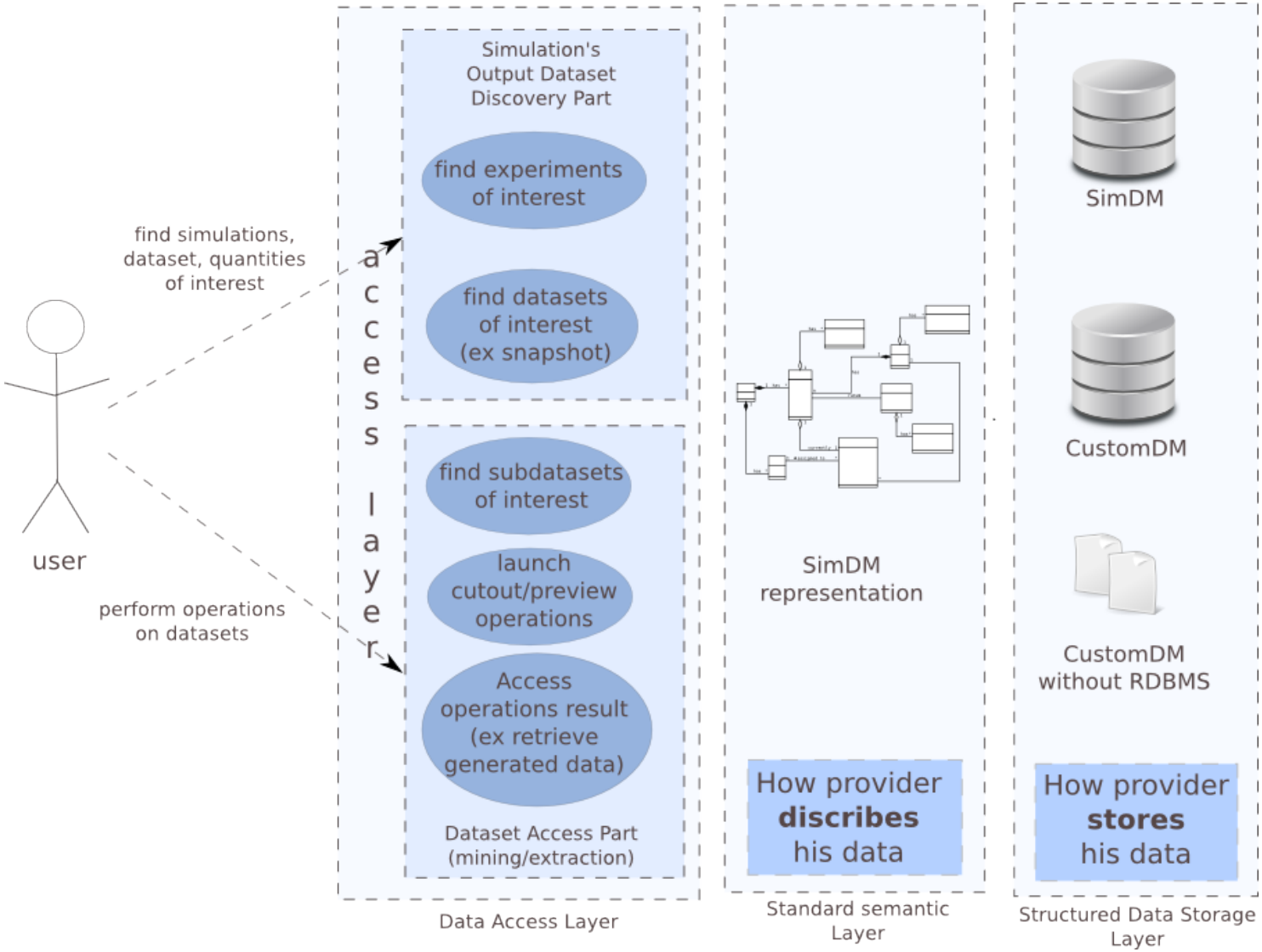
# The SimDAL proposal overview

## SimDAL architecture overview, level 0



# The SimDAL proposal overview

## SimDAL architecture overview, usecases level



# The SimDAL proposal overview

## Interface

We propose that the SimDAL services be accessible through 2 ways

- high level predefined atomic operations (ex cutOut)
- raw query (through an sql-like interface, next called SimTAP)
  - any provider data format will be exposed according to a common language : SimDM
  - allows application interoperability
  - allows custom user queries

# The SimDAL proposal overview

## Discovering Data

Basically it is about identifying a **dataset of interest** (output of a simulation) given a particular **protocol setting** (input of simulation).

It's done through the raw query interface.

# The SimDAL proposal overview

## Accessing Data

Based on the use cases we identified a set of core operations :

- preview
- cutOut
- download

SimDAL must provide standard services that

- provides user with a way to **identify subsets of interest** of a dataset
- provides user with a way to **extract subsets** of a dataset
- provides user with a way to **get dataset raw data** and understand/use it

# Part 2 Summary

- Access layer for theoretical data = data discovery + data access
- 1 straightforward access layer on top of multiple heterogeneous data storage configurations
- SimDAL missions : **easy** to use, **DAL compliant**, **generic**
- SimDAL interface = 2 components : raw query + high level operations
- Discovery part : **find datasets of interest**
- Access part : **identify and extract subsets, then get raw data**

# SimDAL Architecture overview

## Data storage format/organization

Although the preferred way to store and organize data is SimDM, some providers may want to use an other one, fitted to there needs.

The solution proposed here is

1. map custom provider data format to SimDM (in terms of semantics)
2. provide a simple TAP-like interface (SimTAP) to query the provider's custom DM according to SimDM semantics

Note : Here, DM means "a way to store and organize data", it doesn't imply a relationnal DBMS (it can be simple plain text files).

# Interface/Data discovery

## Raw queries

To address the need to perform raw queries and hide SimDM complexity, we introduce SimTAP.

- it is a **light extension of the TAP standard**
- it is **easy to use and implement**
- it is intended to be plugged on top of a any DataModel
- it is part of SimDAL and use SimDM semantics

# Interface/Data discovery

## Why SimTAP

Why to overload TAP ?

- Entities in SimDM are very abstract in comparison to other VO DM : there is no valuable information in a DM entity queried alone.
- In a RDMBS implementation of SimDM, the simplest meaningful query requires a lot of joins among voluminous tables

# Interface/Data discovery

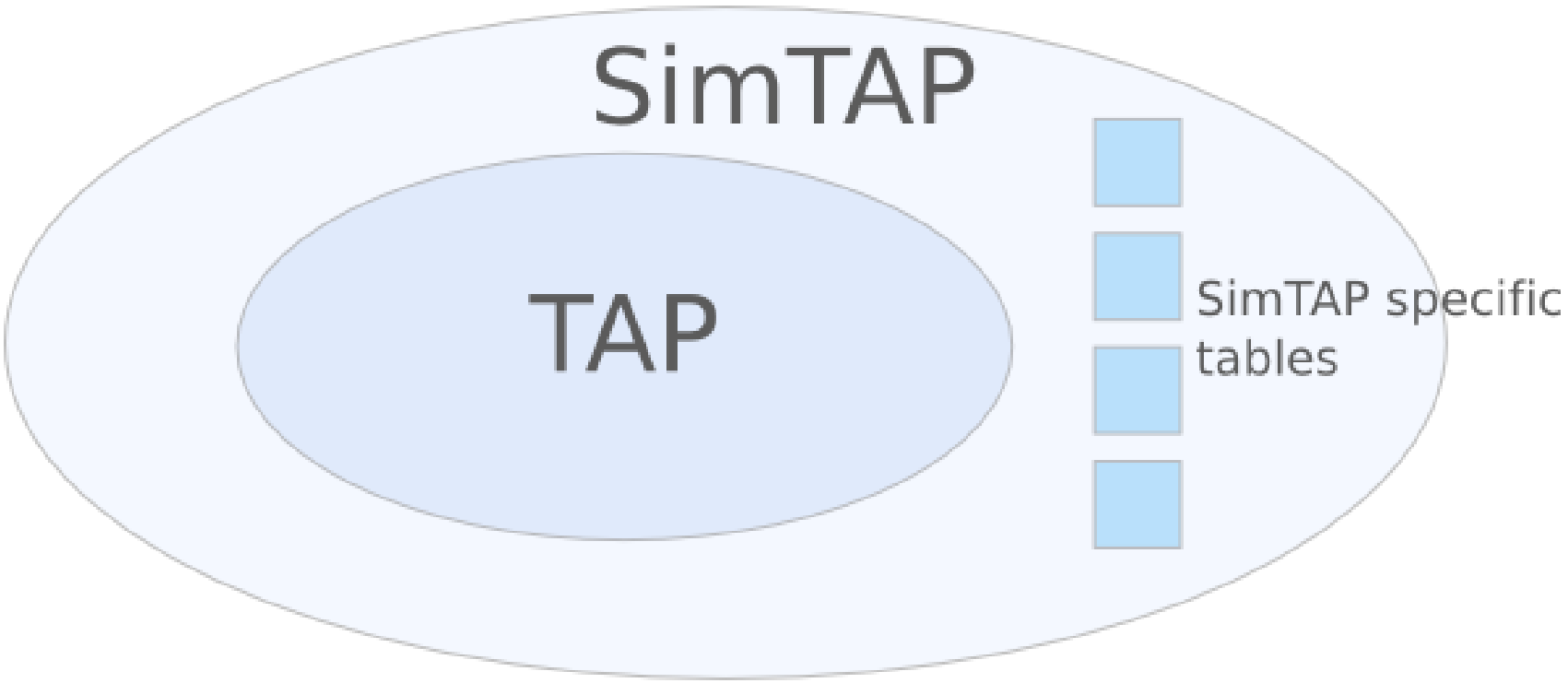
## SimTAP overview

The SimTAP definition is very simple, just a set of 2 requirements

- **must be compliant with the TAP standard**
- **must define 4 mandatory tables, whose 2 of them are freely named**

# Interface/Data discovery

## SimTAP overview



# Interface/Data discovery

## SimTAP origin

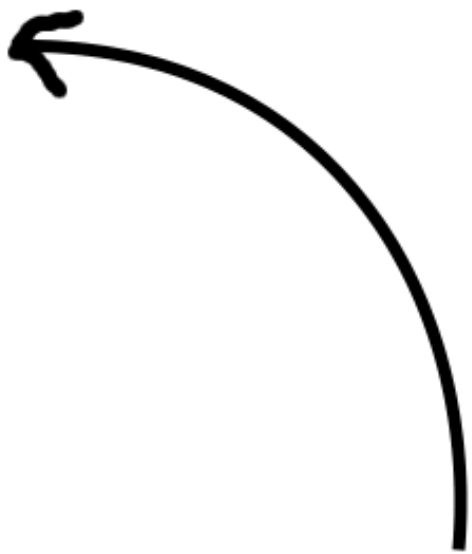
The idea behind the 4 tables addition of SimTAP comes from 2 very common use case in Simulation Access :

- to identify a simulation from a specific protocol setting
- to identify a sub-dataset from constraints on properties values of an objecttype

# Interface/Data discovery

## SimTAP table example from SimDM

| experimentid | input_param_name | input_param_value |
|---|---|---|
| 14 | Baryton Density | 0.044 |
| 14 | Matter Density | 0.23 |
| 14 | Dark Energy Type | 2 |
| 15 | Baryton Density | 0.052 |
| 15 | Matter Density | 0.23 |
| 15 | Dark Energy Type | 3 |



| experimentid | baryton_density | matter_density | dark_enegy_type |
|---|---|---|---|
| 14 | 0.044 | 0.23 | 2 |
| 15 | 0.052 | 0.23 | 3 |

# Interface/Data discovery

## SimTAP details

- a table listing experiments against protocol setting values (call it T-EXP)
- a table listing datasets against the properties statistics on contained objecttype (call it T-DATASET)
- a table mapping a protocol ID to a T-EXP table (call it T-MAP-PROTO)
- a table mapping a set (experiment ID, objecttype skos) to a T-DATASET table (call it T-MAP-EXP-OT)

# Interface/Data discovery

## T-EXP details

| exp_publisherdid | proto_publisherdid | dark_energy_type | matter_density | radiation_density | ... |
|---|---|---|---|---|---|
| Boxlen648_n1024_sucdmw5 | Ram3 | 3 | 0.25 | 0 | |
| Boxlen162_n1024_sucdmw5 | Ram3 | 3 | 0.25 | 0 | |
| ... | ... | ... | ... | ... | |

# Interface/Data discovery

## T-DATASET details

| exp_publisherdid | objecttype_skos | Dataset_publisherdid | min_mass | max_mass | ... |
|---|---|---|---|---|---|
| Boxlen648_n1024_sucdmw5 | http://purl.org/astronomy/vocab/AstronomicalObjects/Halo | 3 | 12 | 2000 | |
| Boxlen162_n1024_sucdmw5 | http://purl.org/astronomy/vocab/AstronomicalObjects/Halo | 3 | 3400 | 2e12 | |
| ... | ... | ... | ... | ... | |

# Interface/Data discovery

## T-MAP-PROTO details

| protocol_publisherdid | table_name |
|---|---|
| Ram3 | simtap_exp_ram3 |
| Fof2 | exp_fof2 |
| ... | ... |

# Interface/Data discovery

## T-MAP-EXP-OT details

| exp_publisherdid | objecttype_skos | table_name |
|---|---|---|
| Boxlen648_n1024_sucdmw5 | http://purl.org/astronomy/vocab/AstronomicalObjects/Halo | simtap_648_halo |
| Boxlen162_n1024_sucdmw5 | http://purl.org/astronomy/vocab/AstronomicalObjects/Halo | b162_1024_halo |
| ... | ... | ... |

# Interface/Data discovery

## SimTAP summary

The aim of SimTAP is

- to encourage data providers to build "SimDM views" of their data model in the form of the 4 basic SimTAP mandatory tables
- to use that tables + TAP as the basis for the SimDAL standard services (high level operations + raw queries)
- to satisfy the major part (80% ?) of common user queries

# Data Access

There are basically 3 issues in the Data Access part of this SimDAL proposal

- To help the user to identify which dataset's subsets he is interested in : **preview**
- extracting that subsets, based on objecttype properties stats conditions : **cutOut**
- defining a standard way to get (**download**) this data.
  - download protocol (basic URI with UWS support)
  - data format (VOTABLE, FITS, HDF5 ?)

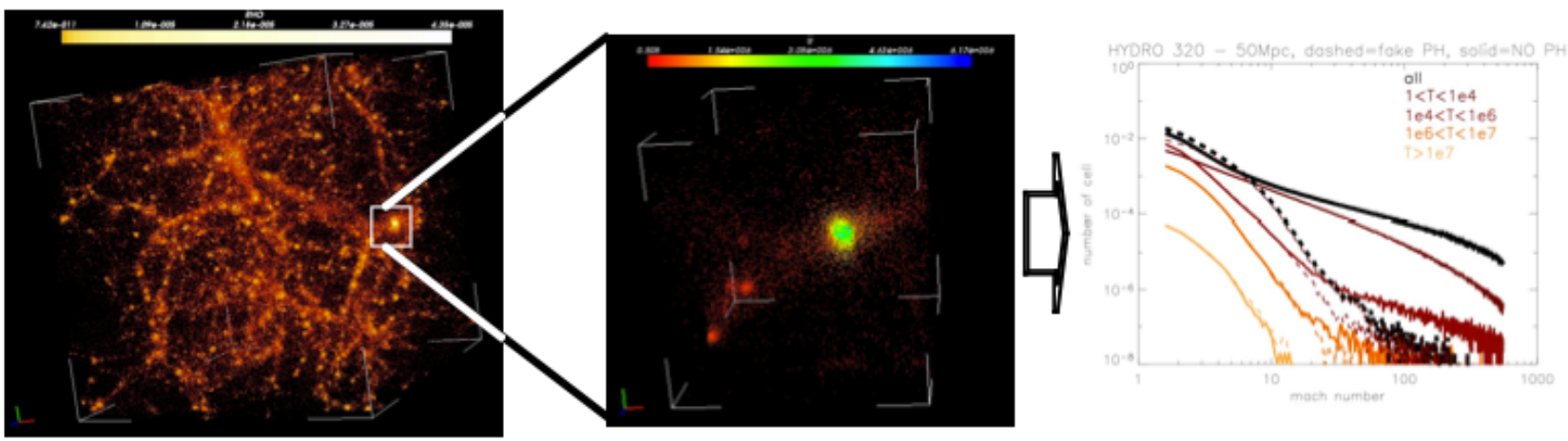
# Data Access

## preview a dataset : preview

The most difficult and interesting issue to address

- How to **preview** interactively a **very large amount of data** ?
- Focusing on which point of view ? Which parameter ? How many dimensions ?

intuitive solution :



# Data Access

## extracting subsets : cutOut

- This operation is very easy thanks to one of the SimTAP's tables.
- The interface has already been defined by Gheller/Lemson

| Operation name: | **Cutout** | | |
|---|---|---|---|
| Input parameter | UTYPE | Required? | |
| EXPERIMENT | SimDB.Experiment.PublisherDID | REQ | ID of the simulation |
| SNAPSHOT | SimDB.Snapshot.PublisherDID | REQ | ID of the snapshot subject to the cutout |
| PROPERTY | SimDB.RepresentationObject.Property | OPT | ID of the quantities to be extracted (if more than one, comma separated list) |
| PARAM | | OPT | IDs of the parameters that define the cutout region (e.g. x, y, z for a geometric cutout) |
| MINVAL | | OPT | minimum value of PARAM (MINVAL/MAXVAL defines the range) |
| MAXVAL | | OPT | maximum value of PARAM (MINVAL/MAXVAL defines the range) |

# Data Access

## extracting subsets : cutOut

Example URL :

http://example.org/simdap/sync?REQUEST=Cutout&EXPERIMENT=my_favourite_simulation&SNAPSHOT=snap0001.h5

Returns the whole snap0001.h5 dataset in the standardized format

http://example.org/simdap/sync?REQUEST=Cutout&EXPERIMENT=my_favourite_simulation&SNAPSHOT=snap0001.h5&PROPERTY=temperature,density&PARAM=xpos,ypos,zpos&MINVAL=0.3,0.5,0.3&MAXVAL=0.8,1.0,0.8

Returns a sub volume with temperature and density from snap0001.h5 dataset. The subvolume has coordinates between 0.3 and 0.8 in x and z and between 0.5 and 1.0 in y

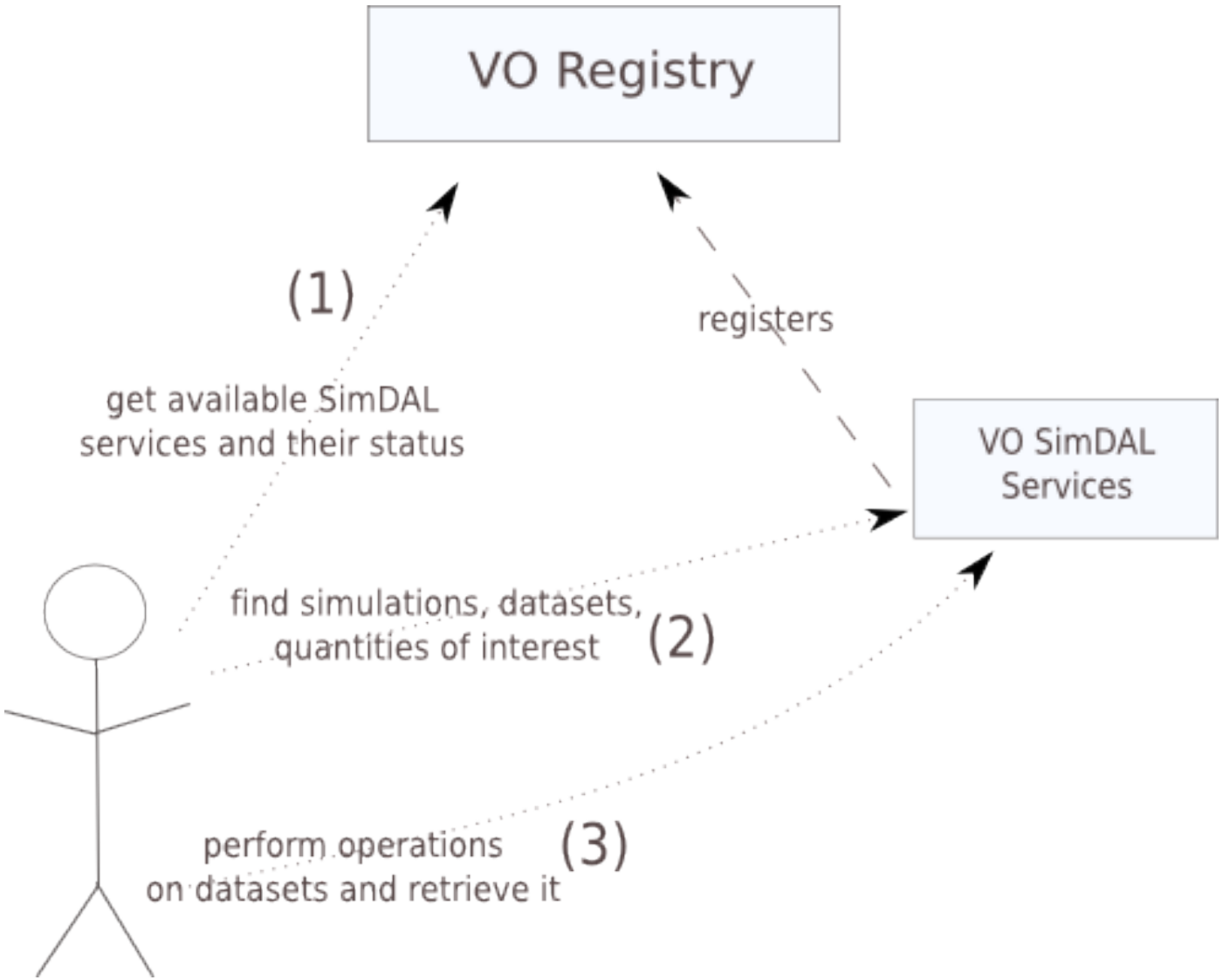
# Data Access

## Standard data format : standard way to transfer/exchange simulation data

- define a new format (C. Geller proposal)
  - + : perfect fitting of the requirements
  - - : reinvent the wheel ?
- make a robust and trusted format mandatory (fits, hdf5...)
  - + : large dataset are well handled (partial loading in memory)
  - + : most simulations already in this format
  - + : quality libraries/tools available
  - - : why one instead of the other ?
- define an "envelop format" interfacing one of fits,hdf5...
  - - : non uniformity in the description of data/metadata
  - + : generic/customizable

# SimDAL integration in VO environment

## Workflow



# Remarks

This approach assumes that the relevant SimDAL service can be found by user through a VO registry query.

- done through querying registry against SimDM related skos concepts
- TAG mechanism needed in registries
  - TAG-TYPE:SimDM, TAG-NAME:OBJECT-TYPE, TAG-VALUE:¡SKOS-URL¿
- Key feature of StandardsRegExt does not fit well (need to copy the whole skos list in voresource spec)
- should a TAG mechanism added to registries ?

# Conclusion

We have proposed a SimDAL specification which

- is **easy to understand and use** for user and provider
- integrates itself with **SimDM as well as custom DMs**
- is strongly integrated in the IVOA's VO environment
- **use trusted standards** instead of reinventing the wheel (reuse of existing libs etc...)