A DATA MODEL FOR LEGACY CODE DESCRIPTOR IN DALIA

Frédéric Boone

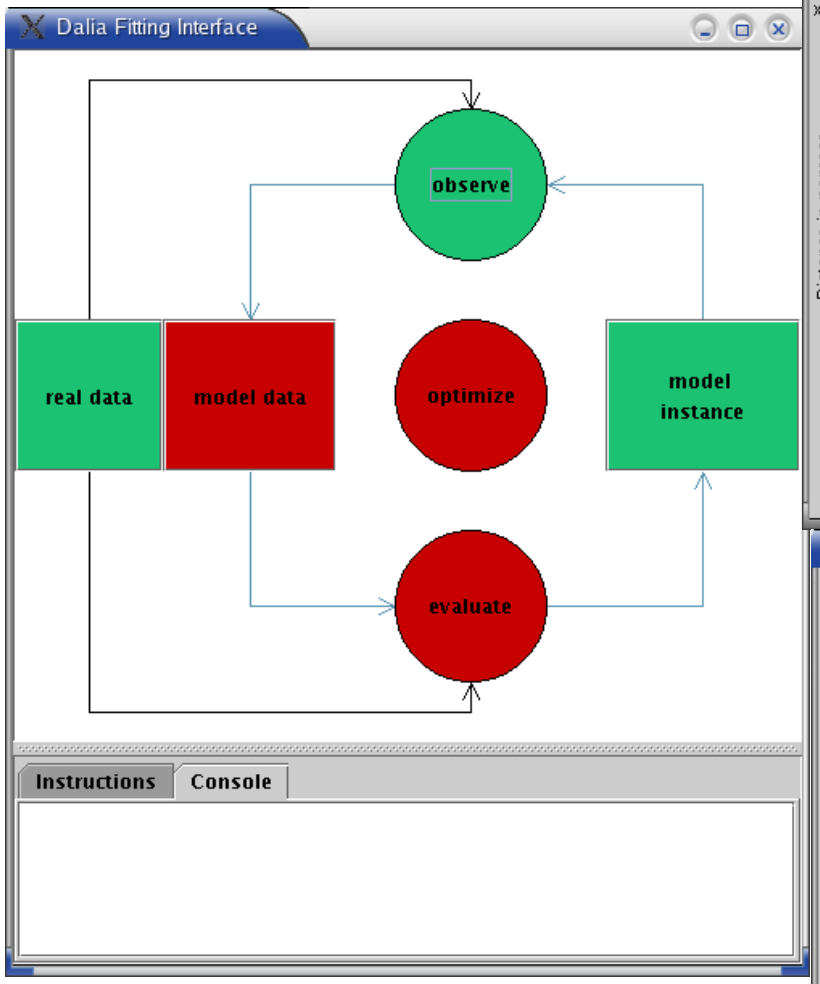
Mireille Louys

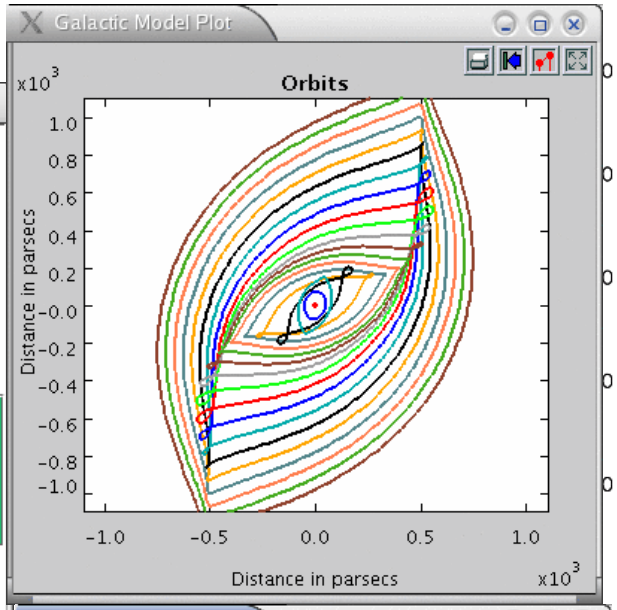
Marie-Lise Dubernet

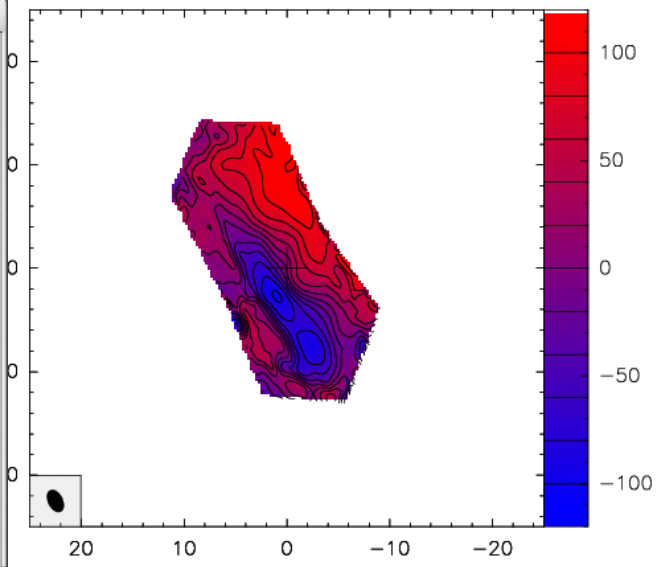
Nicolas Moreau

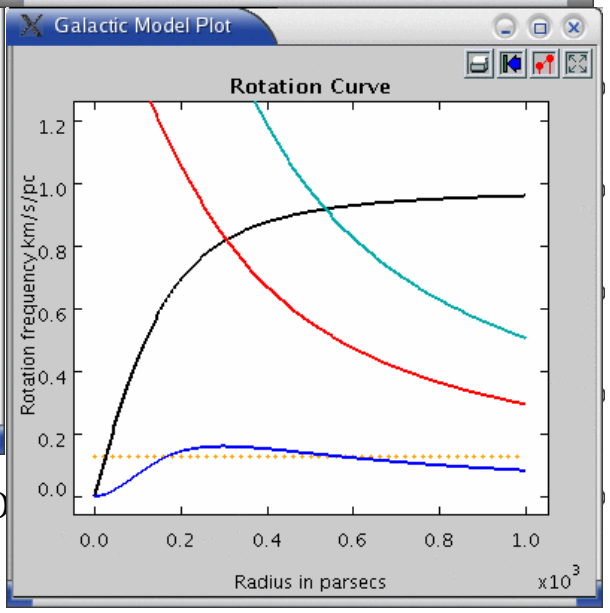
François Bonnarel

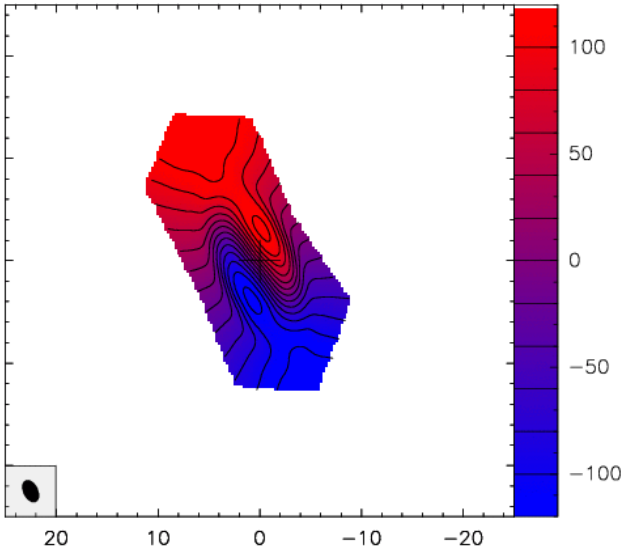
DALIA


Dalia Fitting Interface
observe
real data
model data
optimize
model instance
evaluate
Instructions
Console


Galactic Model Plot
Orbits
x10^3
Distance in parsecs
Distance in parsecs
x10^3



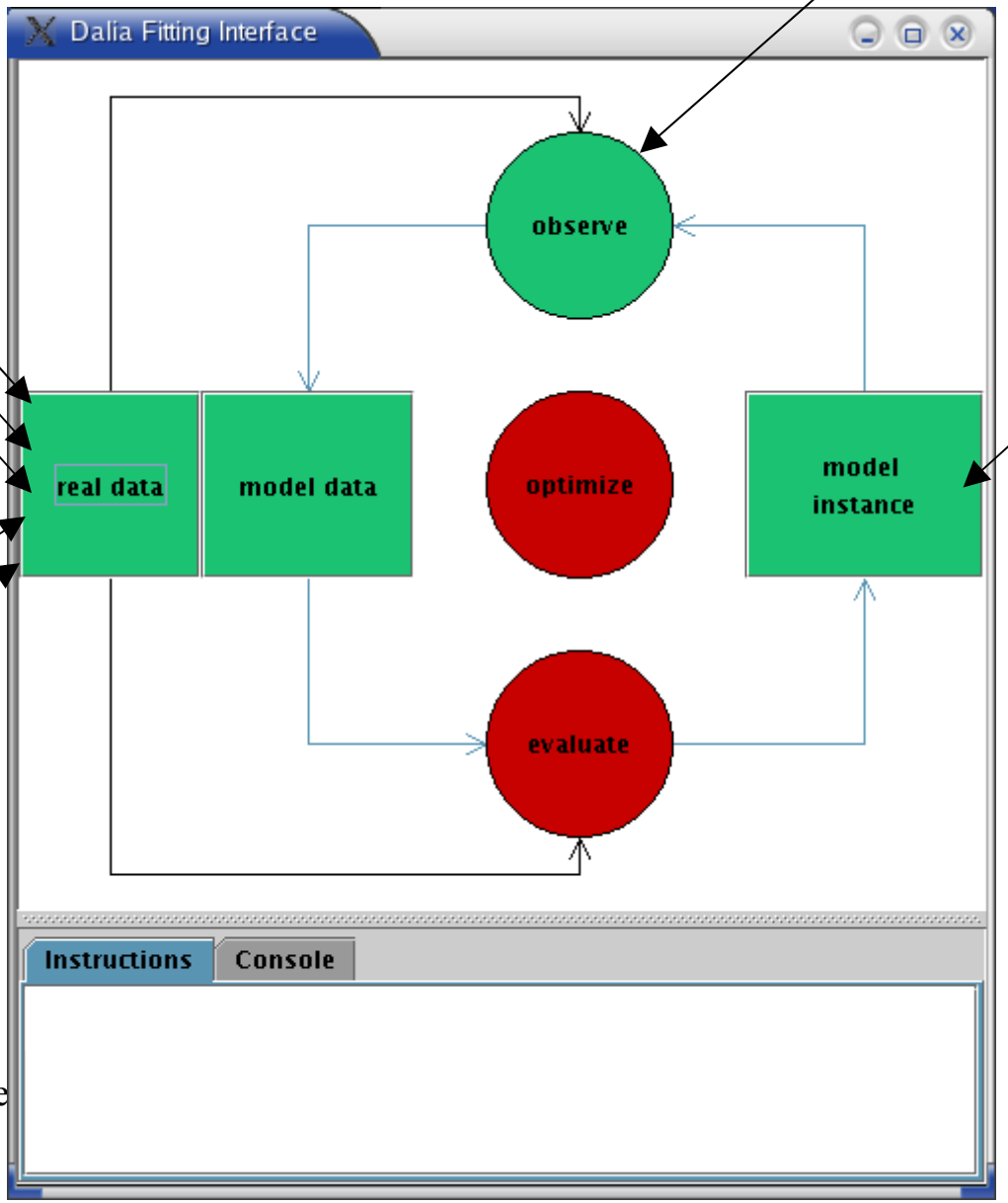

Galactic Model Plot
Rotation Curve
Rotation frequency km/s/pc
Radius in parsecs
x10^3



DALIA dans OV

Model code (algorithm) from archives

Old data from archives



Old optimal model instance from archives

New observations

Motivation

- Standardize description of numerical codes used to modelize sources
- Specify all the information required by a user to run the code: inputs, outputs
- Simulations vs Observations comparisons and interpretation

Code Example

- a code that simulates observations of a galactic disk
  - it takes some parameters as input
  - It produces a data cube (position, position, velocity) of flux density for a given molecular or atomic transition (e.g. Halpha, CO(1-0),...)
- To allow a user (or an application) to use the code (run it and read the outputs) the provider needs to describe the inputs and outputs

Different types of information required

- Inputs related to the *astrophysical source*

  inclination, size of disk etc...

- Inputs related to *instrumental effects*

  psf, etc ... **--> New data model for parameter sets**

- Inputs related to the *algorithm* used in the code

  number of particles

- Inputs related to general *knowledge in physics*

  frequency(ies) of the transition(s) considered (Halpha or CO etc...) for the emission

  --> Point to Atomic Line data model (e.g. Dubernet et al)

- Output dataset(s)

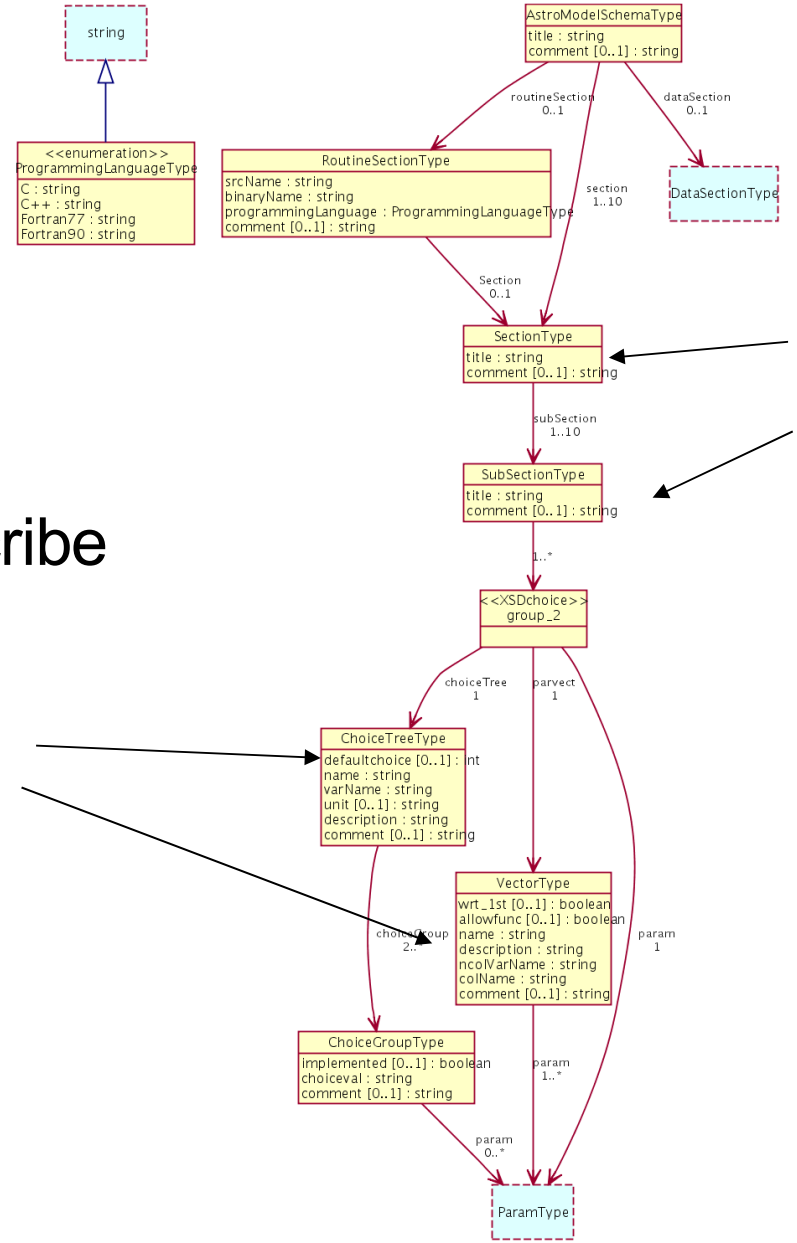
  number of axes, nature of axes, ...

  --> Point to Characterisation, Spectrum data models

Parameter Model Requirements

- To **structure** the parameter set (Sections, Subsections...)
  --> The code provider may structure the parameters according to their physical meaning to help the user to catch the physics behind
- Allow for **dynamical** description
  - Hierarchy (a parameter determines a subset of parameters)
  - Tables (several parameters can be vectors of variable length)
- Allow **information on the numerical code** (program) to be included (native language, variable names used in the code)
  --> help user/application to implement protocol for data exchange with code (generate pieces of codes in the native language)

The UML Schema

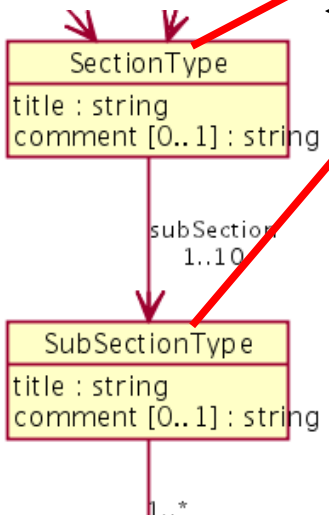


Objects to structure the description of parameter sets

Objects to describe dynamical parameter sets

Structured parameter set

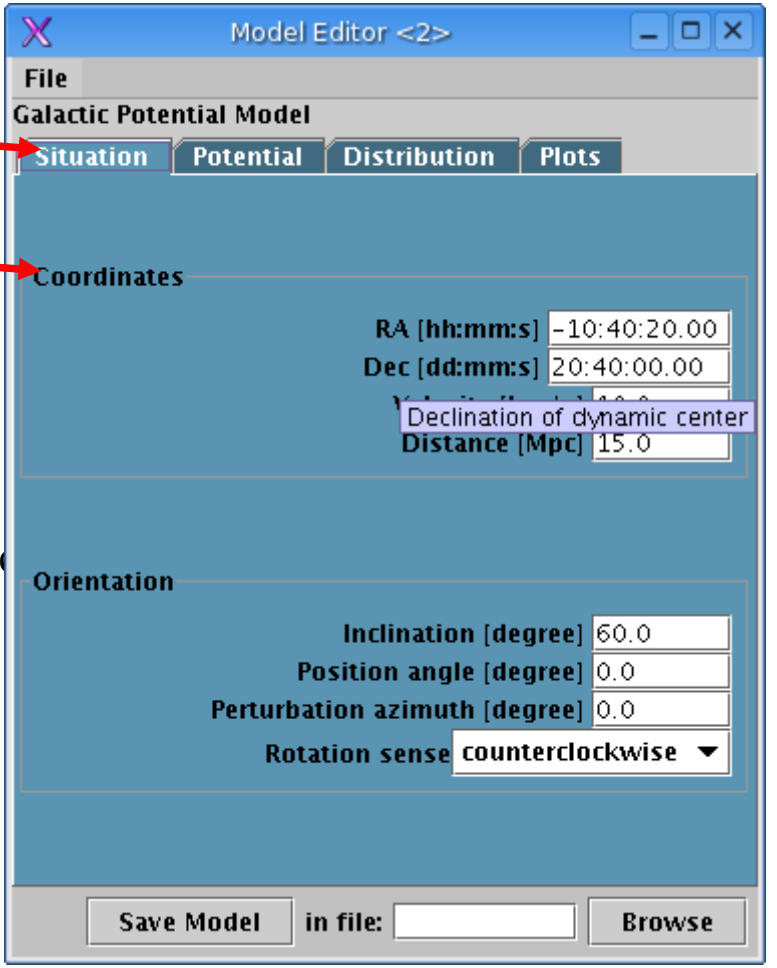
UML



XML INSTANCE

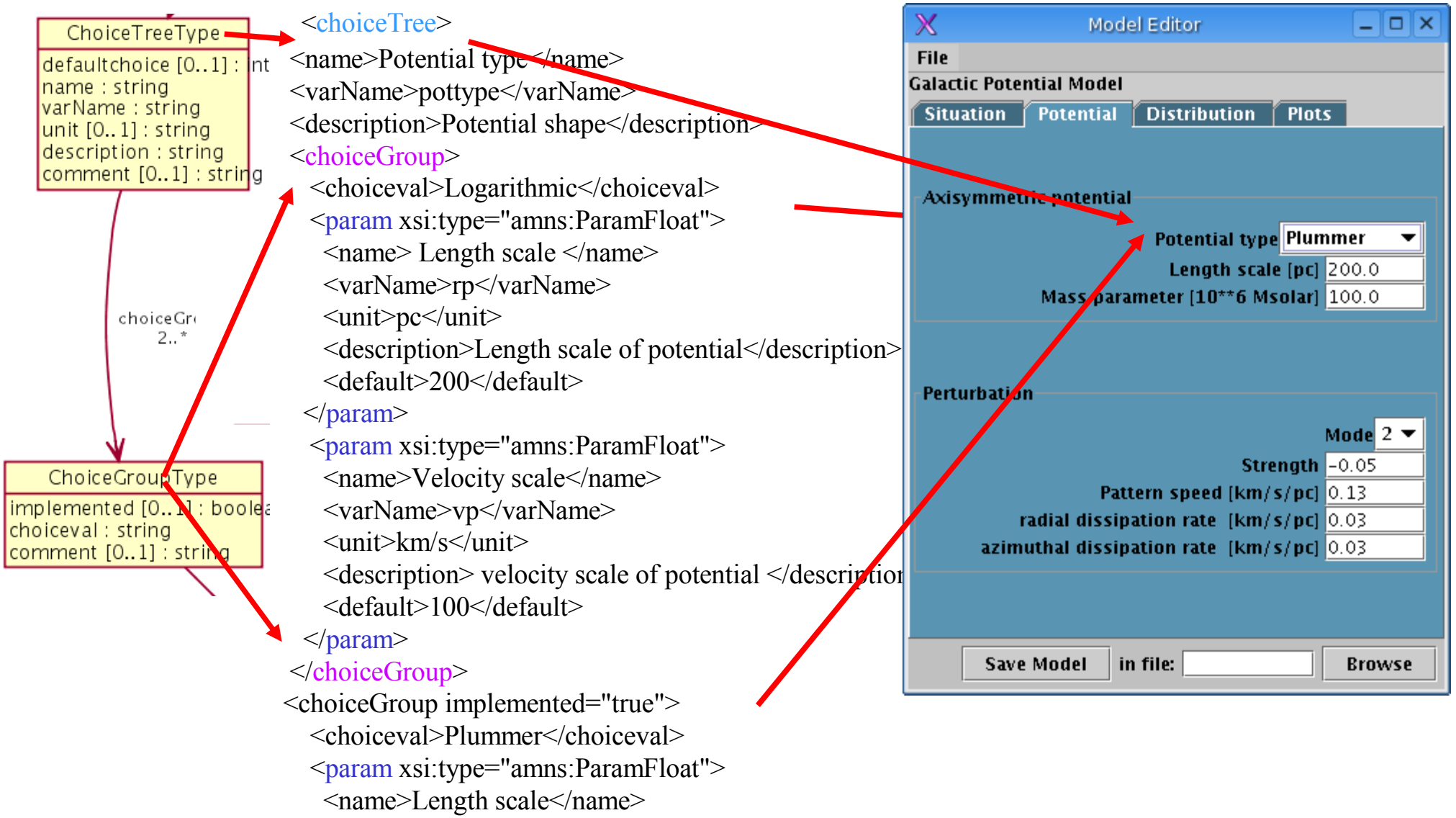
```
<section>
 <title>Situation</title>
   <subSection>
    <title>Coordinates</title>

  <param xsi:type="amns:ParamCoor">
    <name>RA</name>
    <varName>ra</varName>
    <unit>hh:mm:s</unit>
    <description>Right Ascension of dynami
    <default>-10:40:20.00</default>
  < /param>

  <param xsi:type="amns:ParamCoor">
    <name>Dec</name>
    <varName>dec</varName>
    <unit>dd:mm:s</unit>
    <description>Declination of dynamic center</description>
    <default>20:40:00.00</default>
  </param>
```

DALIA GUI



Hierarchy in a dynamical parameter set

UML XML INSTANCE DALIA GUI

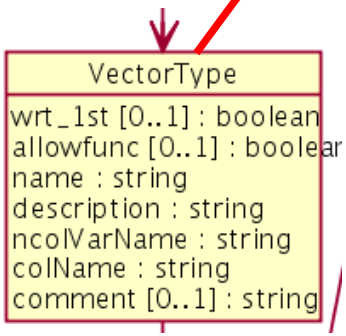


Dynamical table of parameters

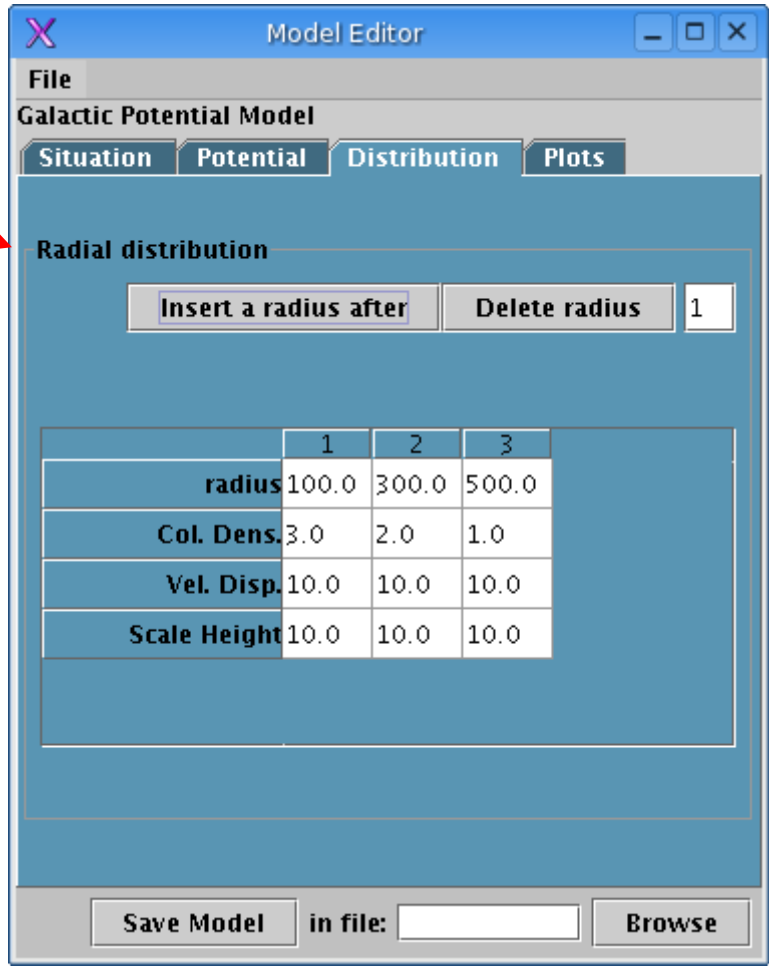
UML

XML INSTANCE

DALIA GUI




```
<parvect>
<name>Radial distribution</name>
<description>Radial distribution of matter</description>
<ncolVarName>nur</ncolVarName>
<colName>radius</colName>
<param xsi:type="amns:ParamFloat">
      <name>radius</name>
      <varName>uradii</varName>
      <unit>pc</unit>
      <description>Radius in pc</description>
      <default>100.</default>
 </param>
 <param xsi:type="amns:ParamFloat">
      <name>Col. Dens.</name>
      <varName>udens</varName>
      <unit>10^22 cm-2</unit>
      <description>Comlumn density</description>
      <default>1.</default>
   </param>
 <param xsi:type="amns:ParamFloat">
      <name>Vel. Disp.</name>
      <varName>udisp</varName>
      <unit>km/s </unit>
      <description>Velocity dispersion </description>
      <default>10.</default>
  </param>
 <parvect>
```



Perspectives

- Publish numerical models for data analysis through a Registry
- Share « instances » of models in the sense of an instance of parameter set:
  - e.g. result of a model fit = values of the best fit parameters
    --> need to define a data model for instances of parameter sets

    ==> share theoretical knowledge at the same level as the observations