Hawaii Interop 2013

The PDL description editor

Carlo Maria Zwölf,

Laboratoire d'Etude du Rayonnement
et de la Matière en Astrophysique

Renaud Savalle.

PDL: A QUICK OVERVIEW

- Parameter Description Language (PDL) is intended to be a lingua franca of parameters:
  - Describes params in a sufficient detail to allow workflow tools to check if parameters can be “piped” between services
    - Physical Properties (Nature, Meaning, unit, precision,...)
    - Computing (Numerical Type, UCD, SKOS concept)
  - Also has capabilities do describe constraints on parameters
    - Physical constraints
    - Arbitrary (including mathematical) constraints
- **Not** a description of parameters “values” (cf. UWS).

PDL: A QUICK OVERVIEW

- Parameter Description Language (PDL) is intended to be a lingua franca of parameters:
  - Describes params in a sufficient detail to allow workflow tools to check if parameters can be “piped” between services
    - Physical Properties (Nature, Meaning, unit, precision,...)
    - Computing (Numerical Type, UCD, SKOS concept)
  - Also has capabilities do describe constraints on parameters
    - Physical constraints
    - Arbitrary (including mathematical) constraints
- **Not** a description of parameters “values” (cf. UWS).

PDL Uses				
Generic software components can be 'configured' by a PDL description for creating quickly fully interoperable new services				
Server exposing services as web services	User Interface (for interaction with PDL services)	Auto Generation of checking algorithms from description	Workflow plugin (for WF interaction with PDL services)	A priori computation of interoperability graphs

PDL NEED COMES FROM SCIENTIFIC SERVICE

Scientific real use case : Service for broadening computations

- Initial level $I \in \mathbb{N}$
- Final level $F \in \mathbb{N}$
- Temperature T in Kelvin
- Electron density ρ in cm^{-3}

Constraints:

- $I < F$
- $\frac{9\rho^{1/6}}{100T^{1/2}} < 1.$

PDL NEED COMES FROM SCIENTIFIC SERVICE

Scientific real use case : Service for broadening computations

- Initial level $I \in \mathbb{N}$
- Final level $F \in \mathbb{N}$
- Temperature T in Kelvin
- Electron density ρ in cm^{-3}

Constraints:

- $I < F$
- $\frac{9\rho^{1/6}}{100T^{1/2}} < 1.$

- Existing solutions (Wadl, WSDL) for describing services does not fit the scientific needs:
  - There is no description of algorithms, physics and utility behind a given service (one has to know *a priori* the service for using it)
  - There is no description about the physical meaning of parameters and units
  - Descriptions are only in a computer science sense.
  - Interoperability is understood only in a basic computer science way.

Software components based on PDL

Since parameters and constraints are finely described with fine grained granularity:

- Generic software elements could be automatically “configured” by a specific PDL description instance:
  - Services containers
  - Graphical User Interfaces
  - Workflow Plugins
- Checking algorithms and interoperability checker between service are automatically generated from descriptions