

Social Mining & Big Data Analytics



FLARE: a flexible workflow language for research Infrastructure

*Leonardo Candela,
Fosca Giannotti,
Valerio Grossi,
Paolo Manghi,
Roberto Trasarti*



ISTI CNR, Pisa Italy






Research E-infrastructure

Systems of systems, patchworks of tools, services and data sources, evolving over time to address the needs of the scientific process.

Scientists implement their processes by **hybrid workflows** whose steps include:


- *Use of web applications*
 - *Download and use of software libraries or tools*
 - *Use of workflow execution engines*
 - *Other..*
- 



Use case

Repeating scientific workflows in Research e-Infrastructures

A scientist runs her experiments within an e-Infrastructure. She uses a variety of tools and services integrated by the e-Infrastructure to produce research data and methods. Once her experiment is concluded, the scientists has identified the *hybrid workflow steps* she went through and would like to pin them down, for her and others to repeat the experiment.



Workflow Languages and RI Hybrid Workflows

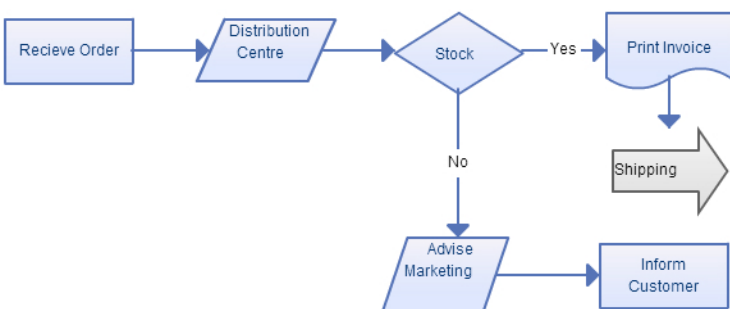
WF Business Languages

Workflows represent a set of logical steps to be interpreted by a human

Research Infrastructures:

pros: scientists may describe and share hybrid workflows

cons: scientists are not provided with tools for hybrid workflows execution



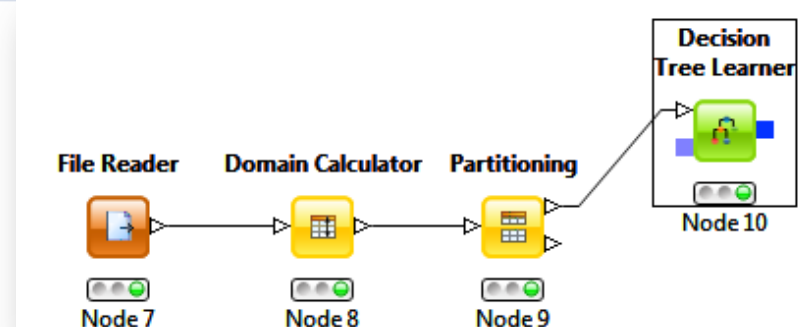
WF Execution Languages

Workflows represent a set of logical steps to be interpreted by a machine (e.g. BPEL, Taverna)

Research Infrastructures:

pros: scientists are provided with tools to create and execute workflows

cons: such workflows cannot be hybrid, must be made of *uniform and interpretable* steps





FLARE

A Flexible workflow Language for REsearch

Addresses the problem of supporting **sharing** and **repeatability** of **hybrid workflows** in **highly-heterogeneous** e-Infrastructures


Lays in between *business process modeling languages*, and *workflow execution languages*





FLARE Steps

FLARE steps model typical Research e-Infrastructure steps, which may include:

- **Tools** (*to be downloaded*): the execution of the step requires the user to download and execute the tool on its own premises;
 - **Web-accessible services** (SOAP or REST): the execution of the step requires a call to the service that is operated by a provider;
 - **Web-accessible applications** (tools accessible via user interfaces from the web): the execution of the step requires accessing the web user interface;
 - **Executable workflows**: the execution of the step requires invoking the respective workflow execution engine.
 - **Scientific process workflows**: indeed workflows can be obtained by combining, i.e. nesting, other workflows.
- 



FLARE: Language Operators

Data management operators

- Data reader: to provide input to execution operator
- Data manipulation (and formatting): to prepare the input to an execution operator
- Data writer: to specify where to store the output of an execution or data manipulation operator

Step execution operators: An external algorithm is executed specifying the parameter and the data source (if required).


Workflow control: define the execution flow of the process, i.e conditional, loops, variable, etc.






Data Managment

The data management, in particular the readers and writers, operators in FLARE are classified by different types of data sources:

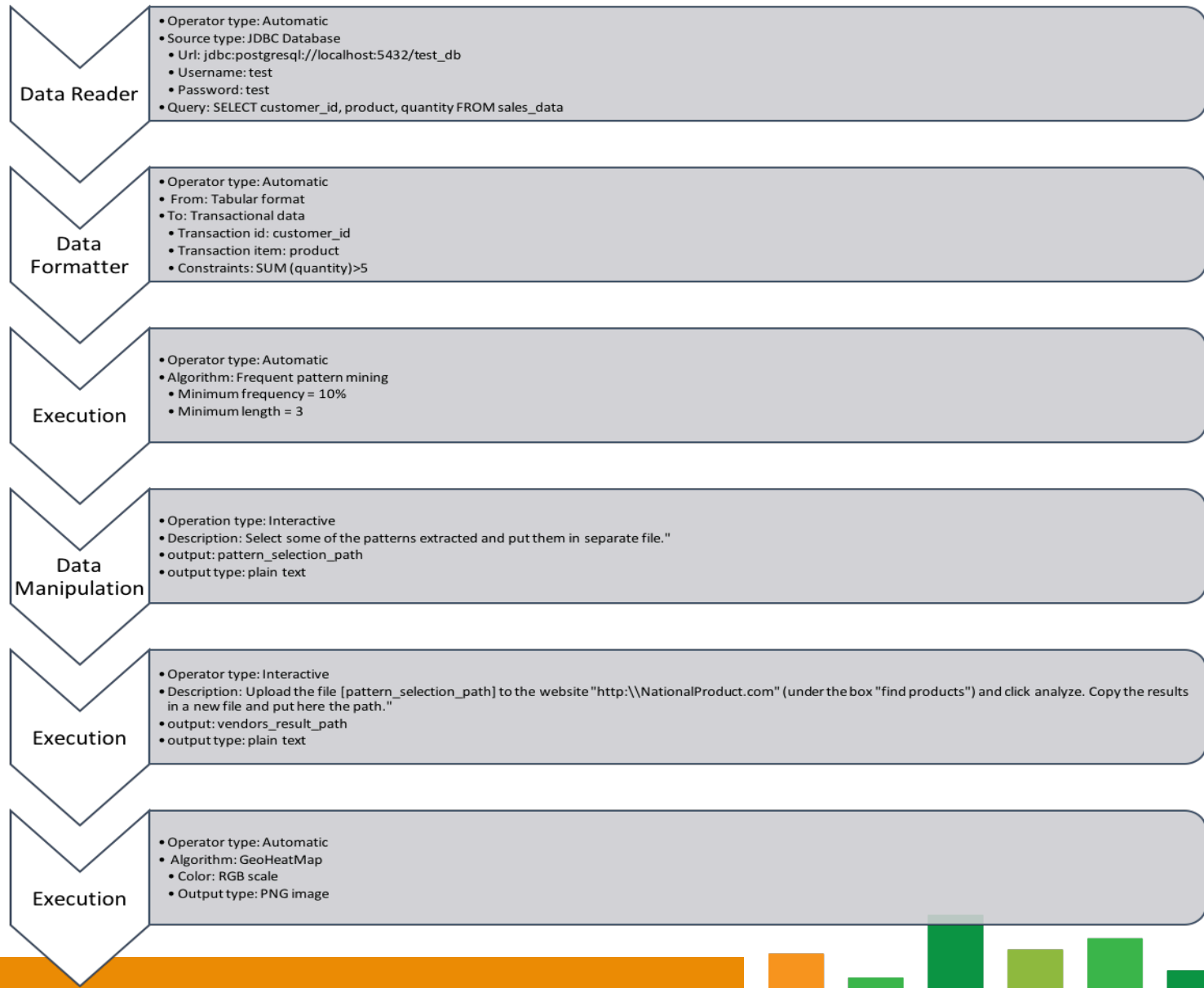
- **JDBC Database:** using the JDBC interface is possible to link a generic database
 - **Registered Database:** the infrastructure give the possibility to register a database, in that case the operator specify only its alias to access to it
 - **Workspace object:** the Research infrastructure might be equipped with several services offering access to stored objects, such as files and data streams
 - **External file:** using FTP (File Transfer Protocol)
 - **Interactive insertion:** This operator allow the user to upload manually the data to be used in several formats, e.g. plain text, tabular, XML, etc.
- 



Step execution

- ***Internal execution*** (automated) the operator requires the name of the “method” (and relative parameters) to be executed by the infrastructure
 - ***External execution*** (manual): the operator requires the **external link** to a service and a description on how to interact with it; examples are **web-services and web applications** (i.e. portals for interactive sessions)
- 

Example of FLARE workflow



Snippets of code (1)

***city** = interactive Insertion {Interactive,
Description="Select a city to be analyzed"}*

***city_data** = Data Reader {Automatic, Registered
Database, alias="dataRepository", query="Select * from
GPS_data where city='"+city+"'"}*

Snippets of code (2)


```
city_geometry = city + "_geometry"
```

External Execution {Interactive, Description="Go to <http://kdd.isti.cnr.it/uma2/?city=city> to see the statistics generated by the Urban Mobility Atlas. Select from the toolbar 'in' and 'systematic' to see the traffic generated by commuters entering in "+ city + ". Do the same selecting 'out' and 'systematic'. Determine the areas you are interested in (e.g. the one with higher volume of traffic) and create a set of (postgres) geometry in a file representing them. Upload the file as "+city_geometry}



FLARE: workflow execution

Workflow descriptions can be created and executed by the RI via GUIs

- **Creation:** the GUI allows scientists to select the steps, complete their descriptions, and organize them into pipelines
 - **Execution:** the GUI, given a workflows, drives the scientist through its execution, by automatically executing the internal steps and guiding the scientists at the execution of external steps
- 



Future Work

The scenario presented is a starting point on which building a workflow language that allows the representation and reproducibility of a scientific process in a research e-infrastructure.

FLARE will be the scientific process workflow language of **SoBigData.eu** Research Infrastructure, which builds on the **D4Science.org** e-infrastructure platform.

The idea is to extend an existing language such as R or Knime to reach the flexibility proposed.

