ToolDog — generating tool descriptors from the ELIXIR tool registry

Kenzo-Hugo Hillion\(^1\), Ivan Kuzmin\(^2\), Hedi Peterson\(^2\), Jon Ison\(^3\), Hervé Ménager\(^1\)

\(^1\)Bioinformatics and Biostatistics HUB, C3BI, Institut Pasteur, France. Email: kehillio@pasteur.fr
\(^2\)Institute of Computer Science, University of Tartu, Estonia.
\(^3\)Center for Biological Sequence Analysis Department of Systems Biology, Technical University of Denmark, Denmark.

During the last years, integration of various tools has been eased by the use of workbench systems such as Galaxy \(^1\) or frameworks using Common Workflow Language (CWL) \(^2\). Still, it remains time consuming and not straightforward to adapt resources to such environments. ToolDog (Tool DescriptiOn Generator) is the main component of the Workbench Integration Enabler service of the ELIXIR bio.tools registry \(^3\), that guides the integration of tools into workbench environments.

Why ToolDog?

Taking the example of the widely used workbench environment Galaxy, a consequence of the difficulty to integrate tools is their incomplete description. For instance, an analysis of two Galaxy servers (524 tools from the main Galaxy instance and 310 tools from the Institut Pasteur instance) showed that for many tools, useful properties such as citation information are missing.

How does it work?

ToolDog takes advantages of the information from the ELIXIR Tools registry to guide the integration of tools in workbench environments:

1. Get the information from the registry using the tool ID and its version number through the bio.tools REST API.
2. Build tool descriptors from basic metadata, including:
   - **Galaxy specific**: Map parameter types from EDAM terms in bio.tools to Galaxy datatypes.
   - **Python tool specific**: Infer the execution layer description from the inspection of the code (command line parameters parsing code).

Conclusion & Perspectives

ToolDog helps users generate well-described and well-documented tool descriptions for bioinformatics workbench systems using the ELIXIR Tools registry.

Future work includes:
- The addition of code analysis modules for non-python based tools (e.g. Java, C).
- Model that can be extracted in JSON for bio.tools, Galaxy XML or CWL Tool.
- Build appropriate translator/generator objects to jump from one model to another.


Technical details

ToolDog is developed in Python3. Several steps are based on the following python libraries:
- **galaxyxml** for the generation of Galaxy XML tool descriptions (https://github.com/erasche/galaxyxml).
- **cwlgen** for the generation of CWL tool descriptions (https://github.com/common-workflow-language/python-cwlgen/).
- **argparse2tool** for command line parameters parsing (https://github.com/erasche/argparse2tool) of python based tools.