Questionnaire for Tool Developers - Workshop on Software Infrastructure for Reproducibility in Science

Here are some specific questions about the tool you're presenting. We would like to have these questions answered *before* the workshop to help us better structure the discussions on Friday.
* Required

Name of Presenter *

Ana Nelson

Information about the tool

Name *

Dexy

Brief Description *

Dexy is a free-form literate documentation tool for writing any kind of technical document incorporating code. Dexy helps you write correct documents, and to easily maintain them over time as your code changes. Dexy is a tool which allows you to specify a workflow involving many other types of tools (such as programming language interpreters and compilers, document markup converters, document templating systems, and custom filters implemented in a variety of programming languages).

Is it available for download?

- (•) Yes
- ( ) No

If yes, where can it be downloaded?

http://dexy.it

Select the reproducibility features that the tool support (select all that apply). In the next sections, please answer the questions that are relevant for the tool. *

- [ ] Capture -- capture of the steps carried out in an experiment
- [x] Representation -- creation of an executable specification of the experiment (e.g. workflow)
- [x] Portability -- ability for an experiment to be re-executed in an environment different from the one originally used
- [x] Document Linkage -- linkage between experiments and documents, creation of executable documents

- [x] Verification and Exploration -- ability to reproduce the original results and to vary original parameters / data
- [ ] Archival -- archival of experiment and data (e.g. repository)
- [ ] Longevity -- ability to maintain an experiment consistent and reproducible in a long-term basis
- [ ] Other:

Capture and Representation

How does the tool capture the steps of an experiment?
For example, some systems may require the user to port their experiments to these systems so that the capture can happen, while other tools may do this capture automatically.

Dexy can make use of existing scripts and workflow tools which have a scriptable interface. "Porting" to dexy involves writing a YAML-based configuration file to indicate how scripts depend on one another and how they should be executed.

How is the experiment represented?
E.g.: scientific workflow, script, etc.

Dexy will execute any script provided and specified by the user, in any programming language or using any system which can be scripted/automatically controlled.

Please, give any additional information on the tool about capture and representation that you would like to share.

Dexy is a free-form, general purpose project and document automation tool. As such, it can be used for representing, executing and documenting the workflow of the computational portion of a project. Dexy does not have a native way to represent or capture experiments, however it should be possible for dexy to interface with any such system or language which has a consistent API.

Portability

What type of portability does the tool support?
E.g.: portability among any operating systems, among Linux distros, etc.

Dexy runs on any platform supported by Python. Dexy calls filters which may be platform-specific. Depending on the scripts and filters used, a dexy project may run on other platforms with no modifications. In principle filters can be written to detect the operating system and make adjustments to allow for more transparent portability.

What are the requirements on the experiments themselves for portability to be possible in the tool?

For example, a system may require the experiment to be run on Linux so it can be portable to other environments

Experiments will be portable insofar as they use portable elements. An experiment using a Python script should be portable to most other systems, assuming platform-specific libraries are not used. An experiment using a bash script will only be portable to a linux environment. A project with documents written in a particular format will be portable to any system where a compatible document format compiler is available.

What does the tool preserve between source system (where the experiments are originally created) and target system (where the experiments will be reproduced)?

For example, a system may preserve input files and library dependencies.

A revision control system should be used to store and distribute the source files (input files, dexy configuration files) which dexy will operate on. The tool itself does not preserve anything, it generates a reproducible output based on the inputs.

Please, give any additional information on the tool about portability that you would like to share.

Dexy is very flexible and could be configured to work with binaries/libraries packaged along with input files in a repository, for example. It is also possible for dexy to run a tool which would generate a portability report, or report on versions of all tools used by dexy.

Document Linkage

Does the tool provide linkage between documents and experiments?

E.g.: executable paper.

- (•) Yes

- [ ] No

If yes, how does it work?

Typically, documents in dexy make use of a templating system in which dynamic elements based on output from other scripts and files can be incorporated using simple content tags. This can be followed by one or more filters which convert markup to other formats for publication or sharing. For example, markdown to HTML, or reStructuredText to LaTeX to PDF. Dexy is highly flexible and can support multiple templating systems and markup formats in a single project. Dexy supports several templating systems, and can be easily extended to support more. The most commonly used is Jinja.

Verification and Exploration

Does the tool provide support for a third party to repeat the execution of an experiment?

By repeating the execution, we mean with original data and parameters, outputting the same results.

- [x] Yes
- [ ] No

If yes, how does it work?

The third party runs the "dexy" command in the root directory of the source repository containing the input files.

Does the tool provide support for a third party to vary experiment parameters and data?

- [x] Yes
- [ ] No

If yes, how does it work?

The third party edits whichever input files control experiment parameters/data and re-runs dexy.

Please, give any additional information on the tool about verification and exploration that you

would like to share.

A project which has been properly configured to use dexy will have all dependencies captured, and should provide excellent transparency as to how all documents were generated, and to allow for exploration and modification of the project. However, it is possible for a project to be poorly configured, so using dexy in itself is not a guarantee that all source files are included, all software used is available, etc. A third-party build tool can verify that a dexy build does not fail (which would occur if, say, a specified data file was not available).

Archival and Longevity

Does the tool provide support for archival and longevity?

- ○ Yes
- ○ No

If yes, how does it work?

Please, give any additional information on the tool about archival and longevity that you would like to share.

Dexy runs on source files which are present in the file system. It is expected that source files will be stored in a version control system. Dexy is flexible and configurable, so it could integrate with archival systems.

Submit

Never submit passwords through Google Forms.