High Performance Optimization Navigator

From Research Software To Software For Research - Quality And Sustainable Provision

Thorsten Koch, Ambros Gleixner, Tim Hasler, Wolfgang Peters-Kottig, Yuji Shinano, Franziska Schlösser

HPO-NAVI: Sustainable Infrastructures for Archiving and Publishing High-Performance Optimization Software

Motivation and Goals

The complexity of sustainably developing research software is still largely underrated. How to guarantee visibility and fair attribution? Software utilized to produce the scientific output itself should be regarded as scientific output as well and should be made openly available.

- Foster **reuse** of scientific software.
- Make research software **available**
- **Exemplary implementation** of developed concepts for Ubiquity Generator (UG) and OPUS.

Long Term Use

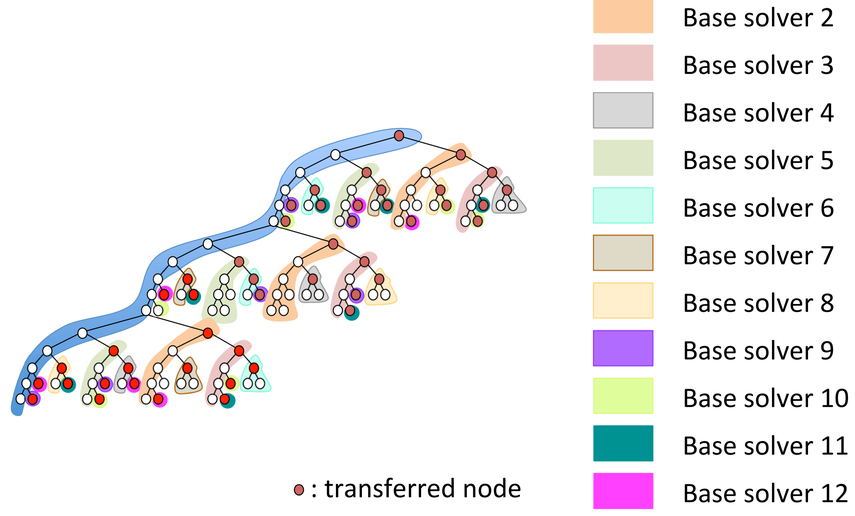
- **Repository Software**:
  OPUS will be improved
- **Define clear workflows**
  for long-term, sustainable software development
- **Deployment and accessibility**:
  Git server for development, archive server download of releases, code and meta information.
- **Software Management Plan**:
  template data management plan for research projects that want to use UG and document according to DFG's standards
- **Improved presentation and visibility**:
  develop a citation recommendation and metadata schemas (conforming to FORCE11 initiative)
- **Quality assurance**:
  define concepts and guidelines for qualitative evaluation of new features, code style and review, testing and continuous integration.
- **Documentation and dissemination of the project results**:
  best practice guide, workshops, and online tutorials.

Plans for UG

Exemplary implementation of goals for Ubiquity Generator (UG):

- **Documentation of the software**:
  description of API and interface of UG, an installation guide, description of the structure, documentation of the source code
- **Software Management Plan**:
  support the management of development of UG and include considerations for funding and licensing
- **Improved presentation and visibility**:
  prototypical extension of OPUS (publish software with landing page, resolvable via DOI, enhanced linkage and presentation in swMATH.org)
- **Digital preservation**:
  experimental ingest software code into OAIS-compliant digital preservation system EWIG and evaluation in the community.

What is UG?



A generic framework to parallelize branch-and-bound based solvers (e.g., MIP, MINLP) in distributed or shared memory computing environments.

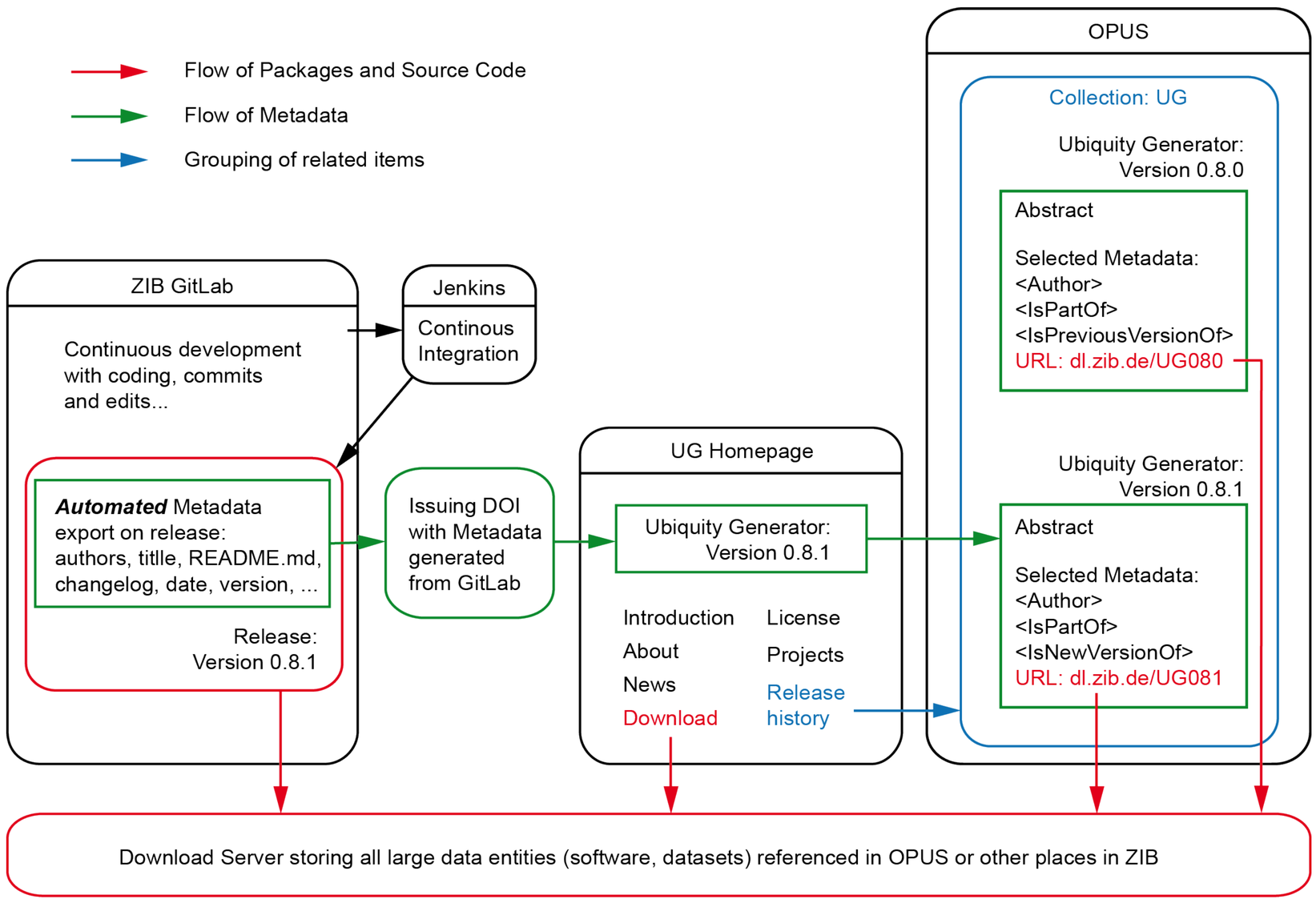
- Exploits powerful performance of state-of-the-art "base solvers", such as SCIP, Xpress, Gurobi, CPLEX, etc.,
- without the need for base solver parallelization.
- Deterministic mode for debugging.

Open Source Software

Open Source Software linked to the project:

- GitLab: **development environment**; harvest metadata (i.e. changelogs, dependencies, contributors etc.)
- Jenkins: **continuous integration**; automation of development process
- OPUS: **literature repository system**; store research literature, soon research software
- swMATH.org: **link between publications and software**; digital preservation strategy for source code. (experimental work)
- EWIG: **digital preservation system**; store code, data with additional metadata

Workflow



Funded by

Deutsche Forschungsgemeinschaft
German Research Foundation

DFG reference number: KO 3894/6-1 Project number: 391087700

Cooperations

Institute of Mathematics for Industry, Kyushu University
Oak Ridge National Laboratory, Managed by UT-Battelle for the Department of Energy
FICO, GAMS
Jülich Forschungszentrum
統計数理研究所, The Institute of Statistical Mathematics
Technische Universität Darmstadt
HLRS, High-Performance Computing Center | Stuttgart
Lawrence Livermore National Laboratory