SINGULAR development process

Hans Schönemann

Universität Kaiserslautern

Singular libraries

- Guideline for writing a libary (manual)
- each library must contain documentation for the library and each procedure (text with texinfo extensions)
- each procedure must have an example
- additional keyword (for index) and references possible via KEYWORD: and SEE ALSO:
- should have examples/tests in the tests suite

Singular kernel

- additional commands must have an description in the manual
- otherwise: follow the style guide: doxygen documention for the routines
- should have additional tests in the source code via assume (which are tested in the debug version of Singular)
- should have examples/tests in the tests suite

Singular test suite

- Tst/regress.cmd: expects a Singular binary either in Tst or given via -s
- single test(s): regress.cmd abc.tst def.tst
- lists of tests: regress.cmd abc.lst
- default tests:
  `Old/universal.lst Short/ok_s.lst Long/ok_l.lst Plura`
- each library function/kernel function should be covered
- each (fixed) bug will be added to the test suite

Singular trac server

- Singular trac server: bugs, etc.
- Singular forum: questions, mathematical problems etc.
- mailing list: questions to all singular developers
- move bug/improvements to trac
- assign anonymous tickets on Singular meetings: every month

Singular releases

- nightly builds (with tests) for several architectures
- official patch versions every month (3-1-1-3)
- new version with new kernel commands (3-1-1: Feb 2010)
- to be discussed and decided on Singular meetings: every month

Ressources

- svn (r/o: `http://www.singular.uni-kl.de/svn`)
- or use git (`http://www.berlios.de`, search for singular)
- or `ftp://www.mathematik.uni-kl.de/pub/Math/Singular/UNI`

Contributing

- svn
- or use git (`http://www.berlios.de`, search for singular)
- or send the patch via email
- or attach the patch to a trac ticket