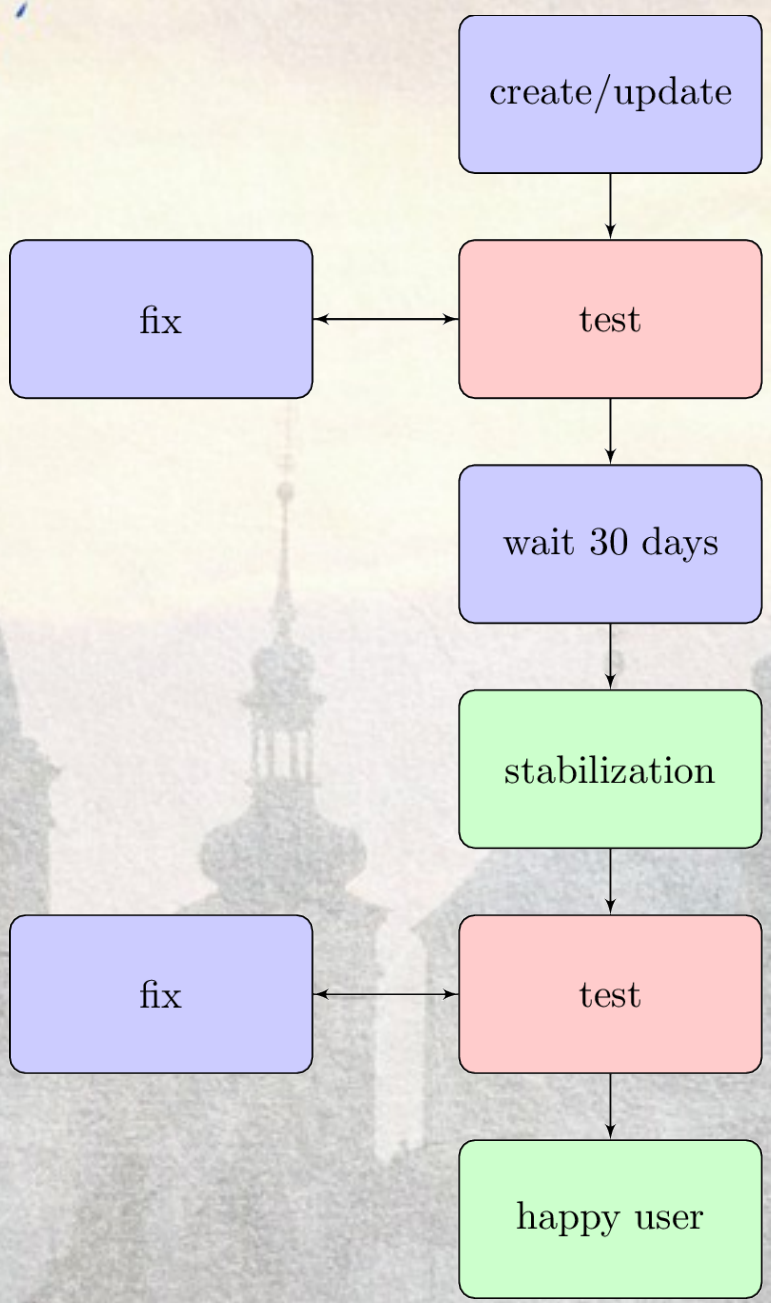
gentoo linux™
Still compiling...

gentoo@home

aka “they'd rather be compiling...”

package workflow



package testing

gentoo linux™
Still compiling...

```
for each config combination
    install deps
    install pkg / unit test
    test pkg / integration test
    remove pkg and deps
```

not for human

tree:~16M packages ~32M ebuilds

5M use flags (average ~2/ebuild)

12 C/C++ compilers (10M packages)

10 python (4M packages)

2 ruby (1.2M packages)

5 jdk (1.4M packages)

>100G tests

automation uses

- keywording
- stabilization
- q.a.
- binary package building
- benchmarks (see andrea's talk)
- pypy/clang/... testing
- integration tests
- release ?

other distros?

opensuse openqa (go see next door now!)
fedora autoqa
ubuntu checkbox/autotest/pbuilder

probably some scripts to recycle
but: need more resources / time / flexibility

solution?

use community resources
enter volunteer computing

virtualization

the easiest way for clean base systems
we can compile on ms users machines
could be modular (i.e. lxc for other users)

infra@work

two clusters (100+400 cores)
nimbus
schedulers:
- condor
- cloud scheduler

tests

minimal gentoo VM + xen kernel
SL5/xen VM + gentoo prefix minimal

test: sci-physics/root-5.34.01 (27 use flags)
15 min for each compilation (-j4)

FEATURES=test and integration (stress -b)
result: 230 failures 54 success 10 cluster

volunteer computing

how to schedule
condor? another pbs?

→ BOINC it!

updating initial system

download stage each time?

regenerate base vms every day ? (too many vms?)

network read-only file system (cvmfs) LHC@home

binary packages

current approach: build pkg with default settings/use flags

→ not really useful in gentoo

typical binary distro: all use flags enabled, split binary packages

why not: binary for minimal pkg + deltas for a decent set of combo options?

binary packages

current approach: build pkg with default settings/use flags

→ not really useful in gentoo

typical binary distro: all use flags enabled, split binary packages

why not: binary for minimal pkg + deltas for a decent set of combo options?

gentoo@home

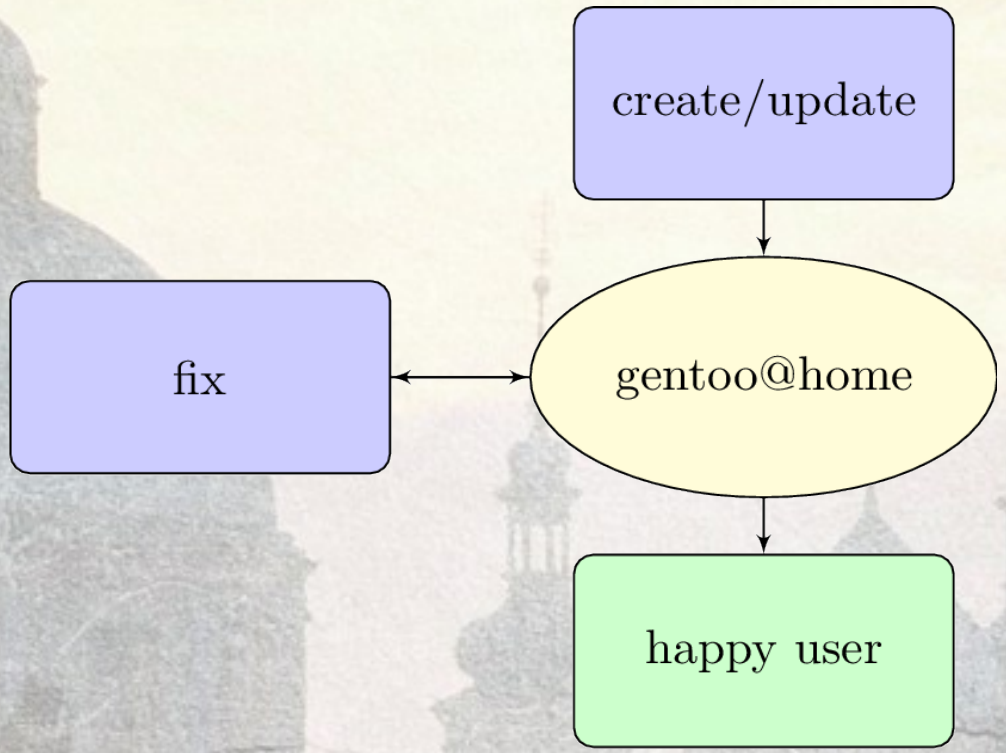
- start with minimal base vm
- lots of scripts to automate
- schedule to community via boinc
- update base via cvmfs
- too much infra work

some ideas

test scripts (one per package / ebuild?)
→ mirror tree for new script API and a post_install_test?

automation scripts (gatt/splat/tatt/...)
need more qa tools
lots of infra work (scheduler, bin pkg storage, tests database)

package workflow



now what?

still vaporware
gsoc 2012: no candidate
two gentoo users showed interest

interested? bicatali@gentoo.org