

BuildNotes

Faster build times

- use a **minimal shell**: `export CONFIG_SHELL=/bin/sh`
- put the **source tree in an SSD**, if you have one.
- put the **build tree in a tmpfs**: `cd tmpfs-mountpoint && /path/to/configure $ARGS`
- **disable unneeded components**: `--without-berkeley-db --without-neon --without-serf --without-apxs --without-sasl --without-swig --without-ctypesgen --disable-javahl --without-kwallet --without-gnome-keyring --without-libmagic --disable-nls` (etc)
- don't build **tools**: run `make bin apache-mod` check-deps instead of `make`
- **reduce output**: pass `'-q'` to configure, `'-s'` to make
- enable configure's **cache**: pass `'-C'` to configure
- enable **concurrency**: pass `'-j'` or `'-j[number]'` to make
 - use `cpuset(1)/taskset(1)` to bind make to N-1 CPUs
- **dependencies**: build against installed dependencies rather than in-tree dependencies
- don't build both **static and shared** libraries: pass either `--disable-static` or `--disable-shared` to configure.

Faster check times

- **PARALLEL=1**
- **CLEANUP=""** (and empty `svn-test-work/` before the build)
- put `svn-test-work/` in a **tmpfs**
- **reduce output**: set `SET_LOG_LEVEL=WARN`, or disable verbose logging to **tests.log**
- set `$TMPDIR` to the **tmpfs** filesystem
- don't enable **SELinux** at boot.

dev@ threads about faster build/check times

https://mail-archives.apache.org/mod_mbox/subversion-dev/201204.mbox/%3C20120415053348.GB20857@tarsus.local2%3E

Suppress sqlite warning messages

- First, take a snapshot of all the 'native' compiler messages that come with the trunk:

```
make 1>stdout.report 2>stderr.constant; sort -u stderr.constant > stderr.unique;
grep -v -F sqlite stderr.unique
```

This suppresses the sqlite warnings and shows you only the current warning messages that are actually `~/trunk` related.

- Any subsequent compilations that are started with the command below will use the generated `stderr.unique` file to filter output and suppress every compiler message that is 'native' to the trunk, leaving just the messages that pertain to your code:

```
make 2>&1 >stdout.report | tee stderr.report | grep -v -F -f stderr.unique >&2 2>/dev/null
```