

Environment Management and Multi-Machine Builds

Jay Alameda

Eclipse Parallel Tools User Workshop

18 September 2012

Environment and Build Management

- Consider this case:
 - Real synchronized project, with makefile
 - Would like to flip back and forth between compilers (eg Cray or PGI) or same compiler (openACC or straight CPU code) and stay with same code base
 - Currently: 1 version synchronized (cray compiler)
 - Need to login and do PGI builds manually/requires different modules

Environment and Build Management (2)

- Makefile excerpt:

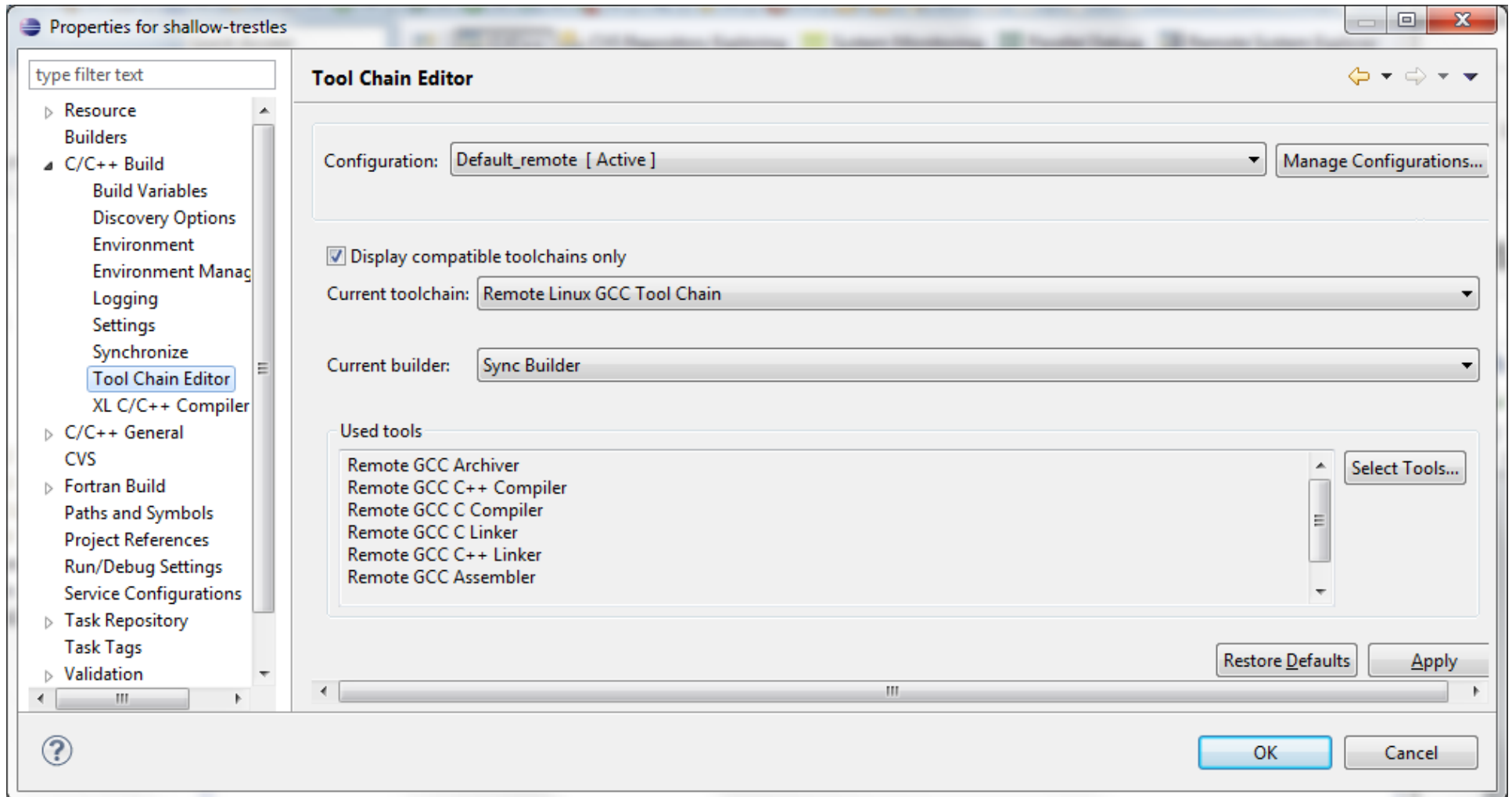
```
# cray ftn
#FFLAGS = -rm -s real64 -O3 -O fp3,cache3,scalar3,vector3 -h acc
# pgi ?
#FFLAGS = -acc -O2 -g -r8 -Minfo=acc,ccff
FFLAGS = -O2 -g -r8 -Minfo=acc,ccff
```

Side Note on synchronized projects: The sync. filtering probably needs tweaking because any large file beyond a few megabytes seems to break synchronization (making it take a very loooong time to synchronize). `core.*` should be added to the defaults and I'd like to see an option for excluding by size (ignore any file > 1M regardless of file type).

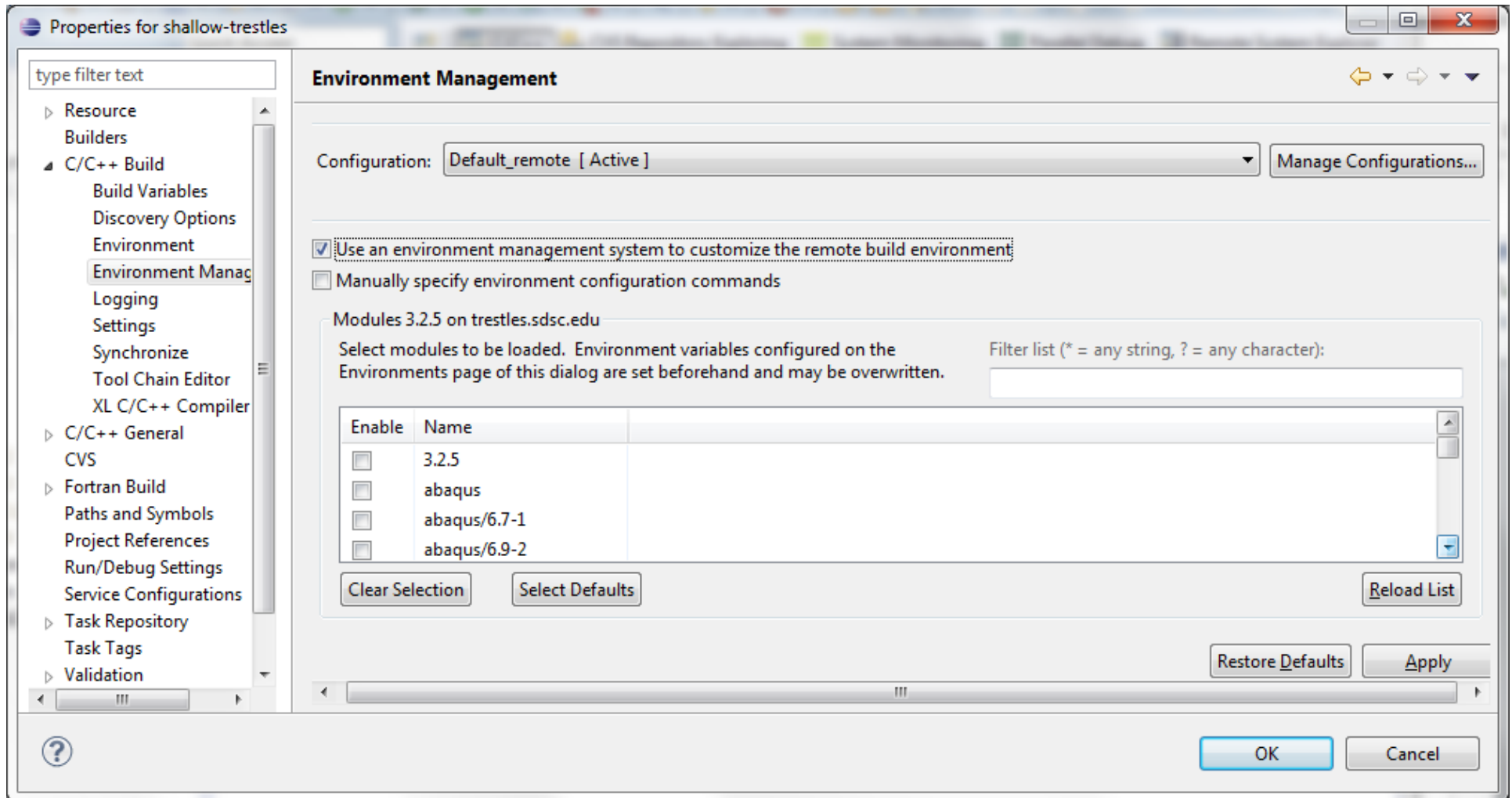
Environment Management

- Current situation:
 - Multiple mechanisms for managing env
 - Local vs remote versus synchronized projects
 - Toolchain contribution
 - MPI include paths *in wrapper script* (-I /path/to/include)
 - Modules
 - Importance of ordering modules!
- Mileage does vary

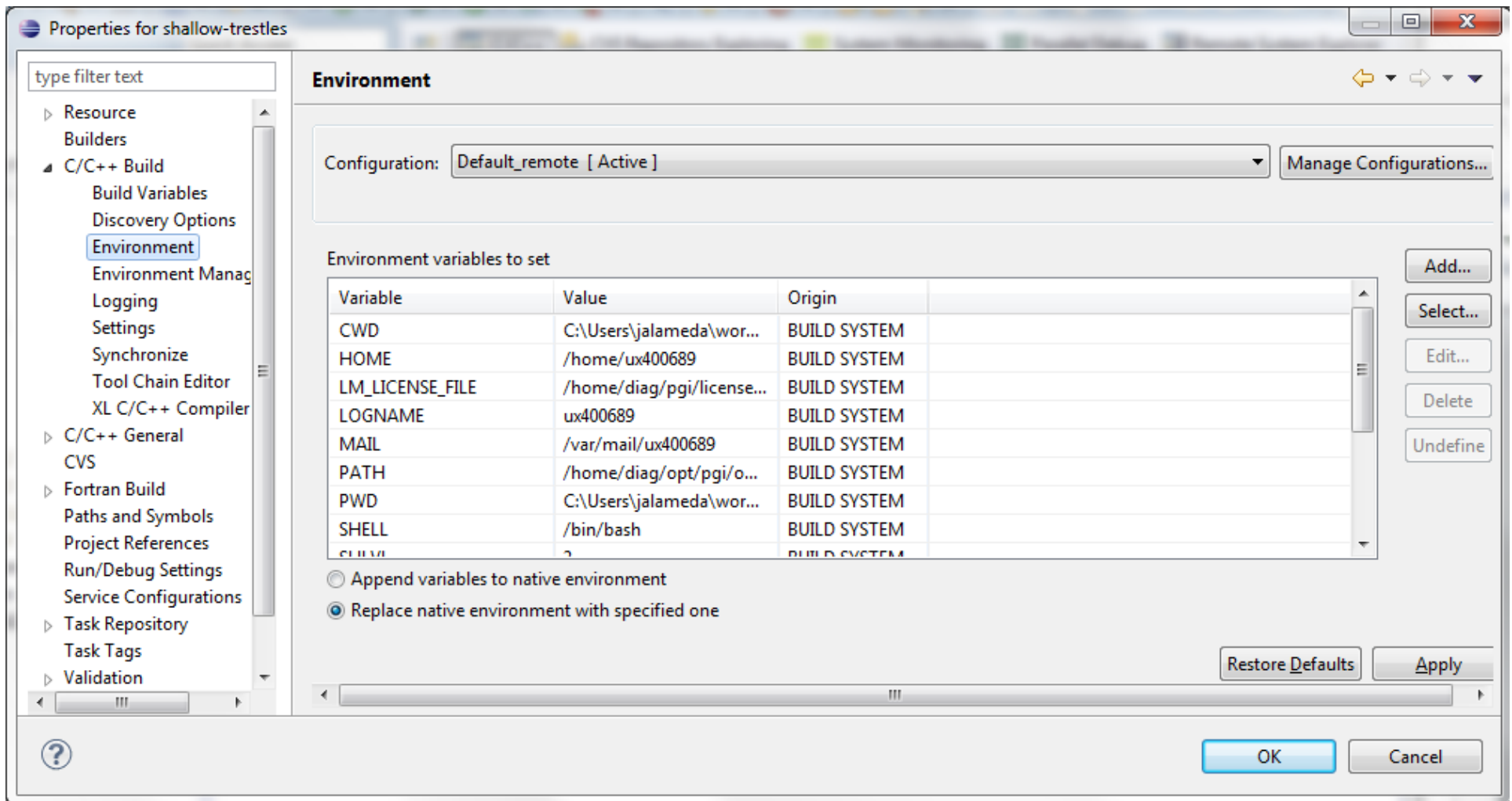
Toolchain interface



Modules...



Environment variables (populated by?)



Managing Multiple Machine Builds

- Using CDT Build Configurations
- How to switch machines
- What works, what doesn't work
- How modules confounds the situation
- What abstraction makes sense to you

