

Nightly build and daily runs

The HARMONIE testbed



Ulf Andræ SMHI

Thanks to Xioahua Yang, Tomas Wilhelmsson

ALADIN 19 workshop/ HIRLAM ASM
Utrecht 2009

Something about the meaning of words

- **Wine**

Something about the meaning of words

- **Wine**

- Red, White
- Cabernet Sauvignon, Chardonnay, Gewürztraminer, Tempranillo
- Rioja, Bordeaux, Bourgogne, Mosel, Madeira
- Vintages

Something about the meaning of words

- **Wine**

- Red, White
- Cabernet Sauvignon, Chardonnay, Gewürztraminer, Tempranillo
- Rioja, Bordeaux, Bourgogne, Mosel, Madeira
- Vintages

- **Harmonie**

(Baigorri Reserva Tempranillo 2004)

Something about the meaning of words

• **Wine**

- Red, White
- Cabernet Sauvignon, Chardonnay, Gewürztraminer, Tempranillo
- Rioja, Bordeaux, Bourgogne, Mosel, Madeira
- Vintages

(Baigorri Reserva Tempranillo 2004)

• **Harmonie**

- Hydrostatic, Non-Hydrostatic
- SURFEX, not-SURFEX
- ALADIN, ALARO, AROME, HIRALD
- 3DVAR, CANARI, downscaling
- IFS, HIRLAM, ARPEGE, ALADIN boundaries

Motivation for a testbed

- Lots of different combinations to play with. All meaningful switches should work!
- Need for a technical and meteorological evaluation
- Continuity between cycles and people
- Realistic full scale tests of the whole system. From compilation to monitoring
- **Repetitive tasks are boring and I'm lazy....**

History

- No existing testbed procedure in HIRLAM (it doesn't mean its not evaluated)
- First tests in HARMONIE were just scattered efforts...
- Mitraillette has been available in ALADIN for a long time
 - Works differently, mainly adiabatic tests
 - A change is accepted if the norms are unchanged or changes can be explained.
 - No general methodology to analyze the experiments (to my knowledge)

The solution, Harmonie testbed.pl

- Test meaningful combinations of changes in the experiment setup (sms/config_exp.h)
- Use only existing tools, i.e. run mSMS through mSMS
- Avoid duplicated work when possible.
 - Compile only once
 - Reuse climate files, observations and boundaries when applicable
 - Make changes at one place only

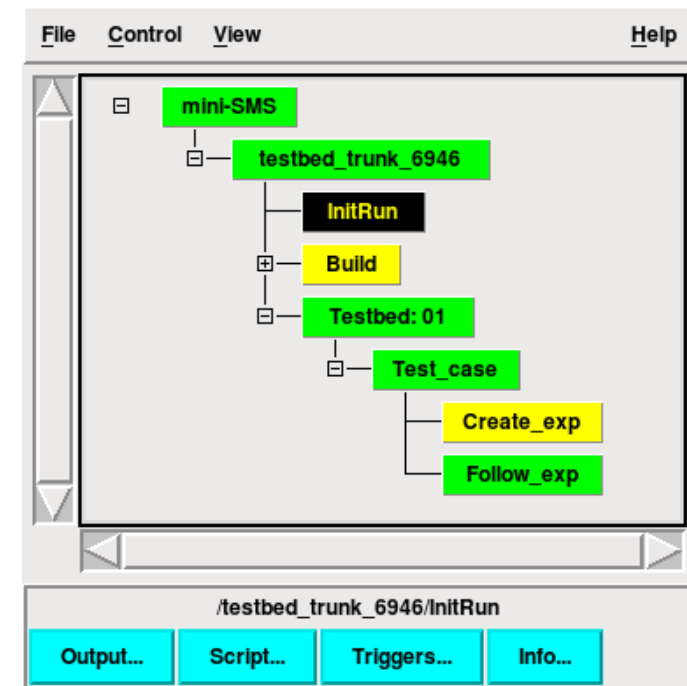


:> Harmonie testbed

- Setup the testbed experiment and build the binaries
- Define the tests as deviations to the default setup.

```
# ALADIN_NH from ALADIN
'ALADIN_NH_BD_ALD' => {
  'ANAATMO'    => 'none',
  'ANASURF'    => 'none',
  'DYNAMICS'   => 'nh',
  'CLIMDIR'    => '$HM_DATA/../../.$ENV{EXP}.'/climate/arome_domain',
  'BDCLIM'     => '$HM_DATA/../../.$ENV{EXP}.'/climate/default',
  'DOMAIN'     => 'SWEDEN_SOUTH',
  'HOST_MODEL' => 'ald',
  'BDDIR'      => '/TESTBED/archive/@YYYY@/@MM@/@DD@/@HH@/ICMSHHARM+0@LLL@',
  'DFI'        => 'no',
  'BDSTRATEGY' => 'available',
},
```

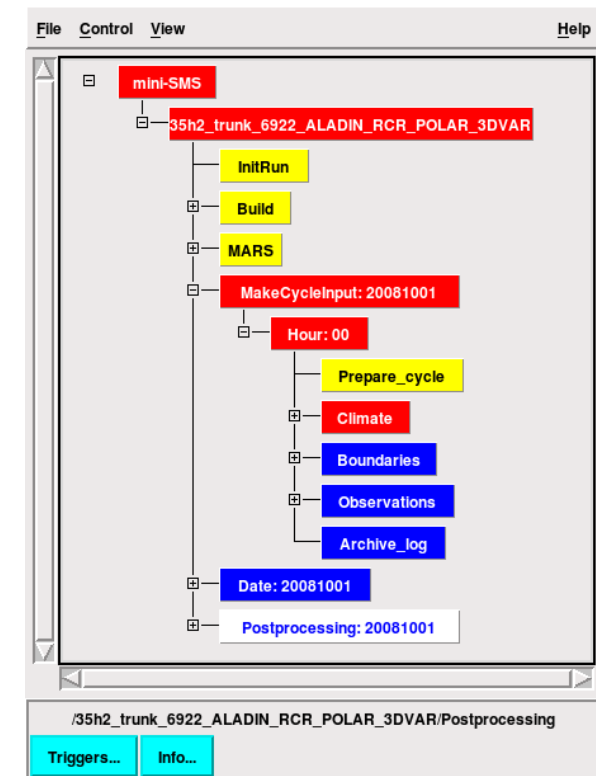
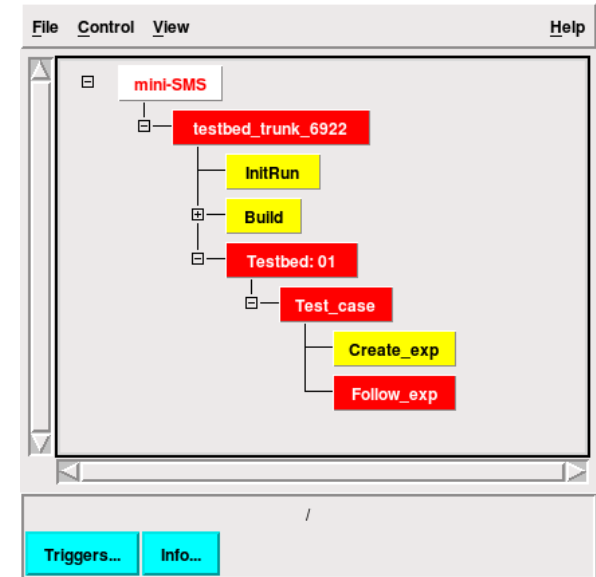
- Create and launch a new experiment like
35h1_trunk_6946_ALADIN_NH_BD_ALD



- Follow the child experiment. Launch next experiment when the last has finished.
- All changes are done in the testbed experiment.
- Loop over the defined configurations: (TESTBED_LIST in sms/config_exp.h)

ALADIN_RCR_POLAR_3DVAR **ALADIN_3DVAR**
AROME **ALARO** **HIRALD** **ALADIN_NH**
ALADIN_SURFEX **ALARO_NH** **ALARO_SURFEX**
HIRALD_NH **HIRALD_SURFEX** **ALADIN_RCR**
AROME_RCR **ALADIN_NH_BD_ALD**

The bold ones have been successfully tested at ECMWF and SMHI.for cy35h1.2



Local implementation

- ECMWF data considered as the default option
- Local deviations can easily be defined
- Errors can be treated as fatal or non fatal
- Graphical window ON/OFF

```
'default' => {
  'LL'           => $ll,
  'BINDIR'      => '$HM_DATA/../../.$ENV{EXP}.'/bin',
  'BUILD'       => 'no',
  'MAKEGRIB'    => 'yes',
  'POSTP'       => 'yes',
  'PPTYPE'      => "'pp md zz'",
},

'gimle' => {
  'BUILD'       => 'no',
  'BINDIR'      => '/gmkpack/bin',
  'BDDIR'       => '@YYYY@/@MM@/',
  'OBDIR'       => '/data/obs',
  'MAKEGRIB'    => 'yes',
  'POSTP'       => 'yes',
  'HOST_MODEL'  => 'hir',
  'CLIMATE'     => '/some/old/climate/files/'
  'CREATE_CLIMATE' => 'no'
  'ARCHIVE_MARS' => 'yes',
  'DTG'         => '2009051400',
  'LL'         => '24',
},

'nrlx019.dyn.smhi.se' => {
  'BUILD'       => 'no',
  'BUILD_ROOTPACK' => 'no',
  'BINDIR'      => '$HM_DATA/../../.$ENV{EXP}.'/bin',
  'BDDIR'       => '/local_disk/local/data/ECMWF/',
  'OBDIR'       => '/local_disk/local/data/obs/',
  'MAKEGRIB'    => 'yes',
},
```

Experiences

- Heavily used for cy35h1
- Efficient to pinpoint new or remaining problems
- To clever solutions sometimes hits back
- The brave ones (where are you?) could use the testbed to check your latest changes.
 - Does my changes in SURFEX work for “all” physics options?
 - Does the changes in the postprocessing work for
- Define a serie of CIS experiments?

Disappointments

- Does a configuration work just because it doesn't fail?
- We are missing a good way to analyze the output
- Are there other useful systems around?
(Bitten?)
- What about the meteorological quality?
 - Doesn't say anything about the standard
 - We have a online verification but no online diagnostics
 - We could define configurations to be used as tests for meteorological evaluation

Nightly builds and daily runs

The way others look upon it:

“The best practices of today's software companies emphasize **frequent integrations** via builds from the top. Such builds integrate changes bit by bit, as testers and developers check them in. **Builds can be nightly or continuous.** This practice assures continual integration and avoids the big bang that lots of colliding changes can cause when the project team delays integration until the release cycle's end.”

Nightly builds

- Build a new rootpack base on an update of the IFS part of the repository
- Rootpack is tagged with the id of the repository changeset
- In schedule at SMHI and at ECMWF
- At ECMWF the rootpack is even built on user demand.

Daily runs

- Trigger a Testbed run based on an update of any part of the repository
- Not everyday, but perhaps weekly?
- Not yet in continuous usage

Coming changes

- Basic monitoring (norms, obs usage, increments, verification)
- Testbed could be an technical and meteorological tool during phasing (under discussion)
- More configurations added like, postprocessing only, more domains, longer periods, ...

Conclusions

- The testbed is valuable tool to test the available meaningful configurations in HARMONIE
- It has been useful for the technical testing of the last cycle (35h1.2) and will be useful for technical and meteorological evaluation of future cycles.
- Documentation
<https://hirlam.org/trac/wiki/HarmonieSystemDocumentation/>
 - -> Evaluation/HarmonieTestbed
- Nightly build script
<https://svn.hirlam.org/trunk/contrib/util/TestHarmonie>

The bed is made for you
Go home and sleep well!

