My installation process from svn.

```
jperaltac@baco:~/Projects/openbabel$ sudo find / -iname "abinitformat.so"
jperaltac@baco:~/Projects/openbabel$ svn st
?       test/files/prueba.xyz
?       test/files/Au.lpmd
A       src/formats/lpmdformat.cpp
jperaltac@baco:~/Projects/openbabel$ svn up
At revision 5011.
jperaltac@baco:~/Projects/openbabel$
```

Compiling

```
jperaltac@baco:~/Projects/openbabel$ export BABEL_LIBDIR=/home/jperaltac/local/lib/openbabel/
jperaltac@baco:~/Projects/openbabel$
```

Then

```
jperaltac@baco:~/Projects/openbabel$ cd ~/buildob/
jperaltac@baco:~/buildob$ ls
jperaltac@baco:~/buildob$ cmake -DCMAKE_BUILD_TYPE=Debug
-DCMAKE_INSTALL_PREFIX=/home/jperaltac/local/ ~/Projects/openbabel/
-- The C compiler identification is GNU
-- The CXX compiler identification is GNU
-- Check for working C compiler: /usr/bin/gcc
-- Check for working C compiler: /usr/bin/gcc -- works
-- Detecting C compiler ABI info
-- Detecting C compiler ABI info - done
-- Check for working CXX compiler: /usr/bin/c++
-- Check for working CXX compiler: /usr/bin/c++ -- works
-- Detecting CXX compiler ABI info
-- Detecting CXX compiler ABI info - done
-- Using included inchi library.
-- Could NOT find LibXml2 (missing:  LIBXML2_LIBRARIES LIBXML2_INCLUDE_DIR)
CMake Warning at CMakeLists.txt:152 (message):
  libxml2 not found - disabling CML support!


-- Found ZLIB: /usr/lib/x86_64-linux-gnu/libz.so (found version "1.2.3.4")
-- Could NOT find wxWidgets (missing:  wxWidgets_FOUND)
-- Looking for conio.h
-- Looking for conio.h - not found
-- Looking for sys/time.h
-- Looking for sys/time.h - found
-- Looking for time.h
-- Looking for time.h - found
-- Looking for strings.h
-- Looking for strings.h - found
-- Looking for rpc/xdr.h
-- Looking for rpc/xdr.h - found
```

```
-- Looking for regex.h
-- Looking for regex.h - found
-- Looking for C++ include sstream
-- Looking for C++ include sstream - found
-- Looking for rint
-- Looking for rint - not found.
-- Looking for snprintf
-- Looking for snprintf - found
-- Looking for sranddev
-- Looking for sranddev - not found.
-- Looking for strcasecmp
-- Looking for strcasecmp - found
-- Looking for strncasecmp
-- Looking for strncasecmp - found
-- Looking for dlopen in dl
-- Looking for dlopen in dl - found
-- Looking for sys/types.h
-- Looking for sys/types.h - found
-- Looking for stdint.h
-- Looking for stdint.h - found
-- Looking for stddef.h
-- Looking for stddef.h - found
-- Check size of clock_t
-- Check size of clock_t - done
-- Performing Test SCANDIR_NEEDS_CONST
-- Performing Test SCANDIR_NEEDS_CONST - Failed
-- Performing Test HAVE_GCC_VISIBILITY
-- Performing Test HAVE_GCC_VISIBILITY - Success
-- Could NOT find Eigen3 (missing:  EIGEN3_INCLUDE_DIR EIGEN3_VERSION_OK) (Required
is at least version "2.91.0")
-- Could NOT find Eigen2 (missing:  EIGEN2_INCLUDE_DIR EIGEN2_VERSION_OK) (Required
is at least version "2.0.0")
-- checking for module 'cairo'
--   package 'cairo' not found
-- Could NOT find Cairo. PNG output will NOT be supported.
-- Looking for getopt
-- Looking for getopt - found
-- Attempting to build the GUI
--    wxWidgets not found => GUI will not be built
-- Found PythonInterp: /usr/bin/python (found version "2.7.3")
-- Configuring done
-- Generating done
-- Build files have been written to: /home/jperaltac/buildob
```

Compiling

make ; make instal

Work finw without error message nor suspissious warnings. Then :

```
jperaltac@baco:~/buildob$ babel
No output file or format spec!
Open Babel 2.3.2 -- Sep 14 2012 -- 17:07:39
Usage: babel [-i<input-type>] <name> [-o<output-type>] <name>
Try  -H option for more information.
jperaltac@baco:~/buildob$ babel -H
Open Babel converts chemical structures from one file format to another

Usage: babel <input spec> <output spec> [Options]

Each spec can be a file whose extension decides the format.
Optionally the format can be specified by preceding the file by
-i<format-type> e.g. -icml, for input and -o<format-type> for output

See below for available format-types, which are the same as the
file extensions and are case independent.
If no input or output file is given stdin or stdout are used instead.

More than one input file can be specified and their names can contain
wildcard chars (* and ?).The molecules are aggregated in the output file.

Conversion options
-f <#> Start import at molecule # specified
-l <#> End import at molecule # specified
-e Continue with next object after error, if possible
-z Compress the output with gzip
-k Attempt to translate keywords
-H Outputs this help text
-V Outputs version number
-L <category> Lists plugin classes of this category, e.g. <formats>
   Use just -L for a list of plugin categories.
   Use -L <ID> e.g. -L sdf for details of a format or other plugin.
-m Produces multiple output files, to allow:
   Splitting: e.g.        babel infile.mol new.smi -m
     puts each molecule into new1.smi new2.smi etc
   Batch conversion: e.g. babel *.mol -osmi -m
     converts each input file to a .smi file
Segmentation fault (core dumped)
jperaltac@baco:~/buildob$
```

Looks like somethings is wrong here, I don't know why the segmentation fault!. Any hint now with this ?

And my environments variables :

```
jperaltac@baco:~/buildob$ env | grep bab
BABEL_LIBDIR=/home/jperaltac/local/lib/openbabel/
OLDPWD=/home/jperaltac/Projects/openbabel
```

jperaltac@baco:~/buildob$

Finallym the valgrind output :

```
==1485== Stack overflow in thread 1: can't grow stack to 0x7fe801ff8
==1485==
==1485== Process terminating with default action of signal 11 (SIGSEGV)
==1485==  Access not within mapped region at address 0x7FE801FF8
==1485==    at 0x40790A: std::string* std::__uninitialized_copy<false>::__uninit_copy<std::string*,
std::string*>(std::string*, std::string*, std::string*) (stl_uninitialized.h:70)
==1485==  If you believe this happened as a result of a stack
==1485==  overflow in your program's main thread (unlikely but
==1485==  possible), you can try to increase the size of the
==1485==  main thread stack using the --main-stacksize= flag.
==1485==  The main thread stack size used in this run was 8388608.
==1485== Stack overflow in thread 1: can't grow stack to 0x7fe801fe8
==1485==
==1485== Process terminating with default action of signal 11 (SIGSEGV)
==1485==  Access not within mapped region at address 0x7FE801FE8
==1485==    at 0x4A255A0: _vgnU_freeres (in /usr/lib/valgrind/vgpreload_core-amd64-linux.so)
==1485==  If you believe this happened as a result of a stack
==1485==  overflow in your program's main thread (unlikely but
==1485==  possible), you can try to increase the size of the
==1485==  main thread stack using the --main-stacksize= flag.
==1485==  The main thread stack size used in this run was 8388608.
==1485==
==1485== HEAP SUMMARY:
==1485==     in use at exit: 1,813,622 bytes in 28,223 blocks
==1485==   total heap usage: 288,894 allocs, 260,671 frees, 866,039,001 bytes allocated
==1485==
==1485== LEAK SUMMARY:
==1485==    definitely lost: 0 bytes in 0 blocks
==1485==    indirectly lost: 0 bytes in 0 blocks
==1485==      possibly lost: 1,758,118 bytes in 26,375 blocks
==1485==    still reachable: 55,504 bytes in 1,848 blocks
==1485==         suppressed: 0 bytes in 0 blocks
==1485== Rerun with --leak-check=full to see details of leaked memory
==1485==
==1485== For counts of detected and suppressed errors, rerun with: -v
==1485== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 2 from 2)
Segmentation fault (core dumped)
```