

# Status of GENSER

- **GENSER\_2\_0\_0** is ready to use
- **Continuous** release scheme for separate generators (installed as soon as released by the authors). **GNU Make, no SCRAM**
- High priority and most medium priority generators are installed.
- Current integrators M. Kirsanov, O. Zenin (until 15.02.2007)

# New GENSER scheme

- **/afs/cern.ch/sw/lcg/external/MCGenerators**
- **The directory tree is**  
**generatorname/generatorversion/share**  
**generatorname/generatorversion/platform1**  
**generatorname/generatorversion/platform2**
- **generatorname in lowercase**
- **Generatorversion is arbitrary, there is no more GENSER1 tendency to use ”\_” as a separator. As before, ”.2” is added to versions with increased NMXHEP**

## New GENSER scheme (2)

- Under `/platform*` there are the directories `/include`, `/lib`, `[/bin]` and usually the file `[config.mk]` (if the GENSER build system is used, to easily see compilation flags)
- All other subdirectories, scripts, Makefiles, common for all platforms, are kept under `/share`
- The generator source tarball, from which the generator is built, is put in `MCGenerators/distribution`, the name convention is `generatorname-generatorversion-src.tgz`
- The generator tarball with binaries is prepared after build and put in `MCGenerators/distribution`

## New GENSER scheme (3)

- The generator tarball with binaries is prepared after build and put in MCGenerators/distribution
- In addition to what is put under platform\* the file README and directory [/examples] are in this tarball
- We included also /share in this tarball because some generators will not work without it (lhpdf). Problem for some users: big. Prepare for these generators the tarball of share?
- Not yet all generators are uniform. Latest developments of rules are applied now to pythia6.

# GENSER installation rules and tools

- Common to all generators is to have under `generatorname`, the files `README` and `lcg_configure` and the directory `/lcgscripts` with the installation scripts
- The usual way of compilation is "`lcg_configure`" followed by "`make`". Script `lcg_configure` runs script `configure` with `lcg` specific parameters. Variations for some generators (`sherpa`, `herwig++`)
- After that we do in GENSER additional copying and tarball preparation. Should we provide a script that installs generator in the same way as in MCGenerators? (CMS?)

# Generators in GENSER CVS: simu/GENSER2

- **CVS is not for normal users!**
- **Additional GENSER files are kept in CVS and tagged according to the version.**
- **There are 3 cases of keeping generators in GENSER CVS**

## Generators in GENSER CVS (2)

- **Case I** : everything, including source code is kept in **GENSER CVS**: pythia6, photos, tauola, hijing, stagen, toprex
- **Case II** : there is an author's tar file in **MCGenerators/tarFiles**, but the build system is provided or modified by **GENSER**: evtgenlhc, herwig, jimmy, lhapdf, alpgen
- **Case III** : the author's tar file and the author's build-install system are used. "lcg\_configure" runs author's configure with LCG-specific arguments. herwig++, thepeg, pythia8, sherpa

## Additional remarks

- Evtgenlhc 8.14 that is used now by LHCb is installed. Previous version, that was in GENSER1, is dropped. Installed from LHCb CVS
- alpgen is specific. The content of the author's tar file is under alpgen-author. GENSER build system builds 2 libraries. GENSER provides directory examples with a build system that can be used separately. Executable for one process is build directly from codes, not from libraries. Precompiled executables? gcc4 support?
- MC@NLO to be installed after obtaining some experience with alpgen
- Probably some generators will change the way they are kept in CVS



# CMT interface to generators

- `/afs/cern.ch/sw/lcg/app/releases/LCGCMT/LCGCMT_XY/LCG_GeneratorsInterfaces`, starting from `LCGCMT_49`
- The version number of the generators are NOT fixed in the interface.
- User needs to specify the version in "requirements"
- Example: `macro pythia6_native_version "409.2"`  
`use pythia6 v* LCG_GeneratorInterface`
- Example remove `pythia6_pdfdummy` from the lib. list:  
`macro_remove pythia6_linkopts "-lpythia6_pdfdummy"`

# Tests

- The package tests is intended for comparably quick testing (usually no histograms)
- Test codes write some numbers in output file, after all tests the comparison program compares with the **reference file**. If some number changes significantly, this is discussed with authors. If OK, new number goes in the reference file
- Most of old tests are adapted to GNU Make (not yet adapted: **isajet and pyquen: generators not yet installed**).
- **New tests**: sherpa W cross section, herwig++ Z event analysis, **pythia W at Tevatron, alpgen W at Tevatron**

# Conclusion

- **GENSER\_2\_0\_0 is ready for use**
- **Most of the structure and the rules are developed. Changes, additions on user requests and in case of problems**
- **Information on the WEB page (demonstration in the next talk)**
- **More on testing and validation in the next talk**