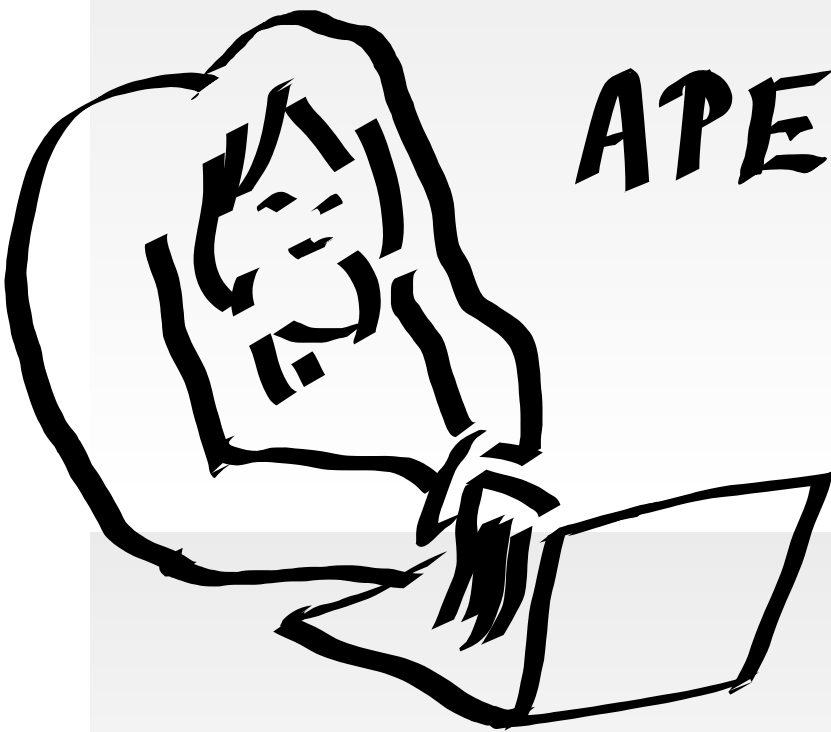


PIERRE
AUGER
OBSERVATORY



APE configuration and usage

Lukas Nellen

ICN-UNAM

lukas@nucleares.unam.mx

What is APE

- Ape is an install and configuration tool
 - Not a full package manager
 - No support for upgrades or removals
 - Erase and re-install model
- Multi-platform
- Also used by other experiments

Install Offline with Ape

- Main reference is the **offline download and install page** in the auger wiki

<https://www.auger.unam.mx/AugerWiki/DownloadAndInstall>

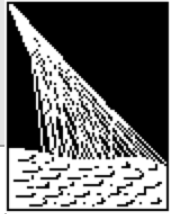
- Official, tested procedure
- List of released versions of offline

● Steps

- Prepare system: install pre-required packages
- Download ape
- Personal configure in `~/.aperc`
 - Site or multi-version: **APERC** env variable points to config file
 - Avoid using `--rc=` option, meant for testing
- Start install (and wait)
- Enjoy offline

- Consult the ape manual for details.

Download wiki page



PIERRE AUGER
OBSERVATORY

[DownloadAndInstall](#)

[FrontPage](#) » [DistributedComputing](#) » [OfflineSoftware](#) » [DownloadAndInstallV2](#) » [DownloadAndInstall](#)

Download and Install

For older releases, check the [DownloadAndInstallV2](#) page.

1. [Download and Install](#)

1. [Download the installer](#)
2. [Installation instructions](#)
 1. [Prerequisites](#)
 2. [Installing Offline using the installer](#)
 3. [Activating your environment](#)
 4. [Installing Offline from the SVN repository](#)
3. [Getting the databases](#)
4. [If you have problems](#)
 1. [Fresh linux installations](#)
 2. [The jobs variable in .aperc](#)
5. [More information on the installer](#)
6. [Old releases](#)

Current release:
v3r3p2 (October 2016)

Download the installer

Offline version	change log	installer	Release date	Notes
v3r3p0 Dornroesch	log	ape 0.99-v3r3p0_1	2016/06/20	This version is not 100% production ready.

The curious can find other ape releases in the [list of packaged ape versions](#).

⚠ OS X notes: The v3r3 series of releases uses Geant 4.9.4. This version of Geant 4 does not compile on OS X with recent XTools. This means that full Geant 4 simulations are currently not possible on the Mac. The rest of the externals and Offline work.

Installation instructions

Once you have downloaded the ape installer, unpack it

```
tar xjvf (name of the installer).tar.bz2
```

Prerequisites

Some packages are either too time-consuming to install from source or not version-critical. Those packages have to be installed from the distribution.

Wiki

[FrontPage](#)
[RecentChanges](#)
[FindPage](#)
[HelpContents](#)
[OfflineSoftware](#)
[OfflineDeveloperToDo](#)
[LukasNellenToDo](#)
[MeetingMinutes](#)
[DownloadAndInstall](#)

Page

[Edit \(Text\)](#)
[Edit \(GUI\)](#)
[Info](#)
[Unsubscribe](#)
[Add Link](#)
[Attachments](#)

More Actions:

User

[LukasNellen](#)
[UserPreferences](#)

[Logout](#)

Ape download page



Auger Package Environment

 [Login](#) [Preferences](#) [Help/Guide](#) [About Trac](#)

	Wiki	Timeline	Roadmap	Source via trac	Source via gitweb	View Tickets	Search	Downloads
--	----------------------	--------------------------	-------------------------	---------------------------------	-----------------------------------	------------------------------	------------------------	---------------------------

Downloads

Here is a list of available downloads:

ID	File	Description	Size	Uploaded	Downloads	Uploader	Component	Version	Architecture	Platform	Type
80	ape-1.00-augerdev-20180521.tar.bz2	ape augerdev-20180521 for auger trunk development	135.7 KB	05/21/18 22:16:03 (4 months ago)	56	lukas	Distribution	0.9...beta	any	any	source
79	ape-1.00dev-augerdev-20180420.tar.bz2	ape augerdev-20180420 for auger trunk development	135.7 KB	04/20/18 08:40:01 (5 months ago)	27	lukas	Distribution	0.9...beta	any	any	source
78	ape-0.99-cdas-v6r0p1.tar.bz2	Ape for Auger CDAS v6r0p1. uses c++98	134.3 KB	10/26/17 09:30:41 (11 months ago)	82	lukas	Distribution	0.99	any	any	source
77	ape-0.99-cdas-v6r0.tar.bz2	Ape for Auger CDAS v6r0. uses c++98	134.2 KB	10/17/17 23:21:17 (12 months ago)	34	lukas	Distribution	0.99	any	any	source
76	ape-0.99-cdas-v5r9-prod-a2.tar.bz2	Ape for Auger CDAS v5r9, 2nd ape patch. uses c++98	134.2 KB	10/10/17 08:47:05 (12 months ago)	47	lukas	Distribution	0.99	any	any	source
75	ape-hawc-2.06.00.tar.bz2	HAWC Release for AERIE v2.06.00	120.0 KB	08/31/17 15:06:28 (13 months ago)	439	sybenzvi	Distribution	hawc-2.06.00	any	any	source
74	ape-0.99-cdas-v5r8p1.tar.bz2	Ape for Auger CDAS v5r8p1	134.8 KB	02/14/17 13:34:29 (20 months ago)	141	lukas	Distribution	0.99	any	any	source
73	ape-0.99-cdas-v5r8.tar.bz2	Ape for Auger CDAS v5r8	134.8 KB	02/08/17 14:11:48 (20 months ago)	72	lukas	Distribution	0.99	any	any	source
72	ape-0.99-v3r3p2a.tar.bz2	For Auger Offline v3r3p2. 17.01 update of externals	137.3 KB	01/25/17 22:57:15 (20 months ago)	519	lukas	Distribution	0.99-v3r3p2	any	any	source
71	ape-0.99-v3r3p2.tar.bz2	For Auger Offline v3r3p2	124.4 KB	10/11/16 08:34:29 (2 years ago)	262	lukas	Distribution	0.99-v3r3p2	any	any	source

Ape Manual

[Ape documentation »](#)

[next](#) | [modules](#) | [index](#)



Table Of Contents

User manual and API
documentation for the **Auger
Package Environment (ape)**
[Indices and tables](#)

Next topic

[Ape user guide](#)

This Page

[Show Source](#)

Quick search

Go

Enter search terms or a module,
class or function name.

User manual and API documentation for the Auger Package Environment (ape)

This is the documentation for release 1.00dev of `ape`. Please consult the [ape trac pages](#) for the documentation of other releases.

The `Auger Package Environment (ape)` is a tool to install the software suite of the [Pierre Auger Observatory](#). It installs the [Offline software](#).

The services provided by `ape` include:

- Package download.
- Unpacking, building, and installing of packages.
- Setting up of a build environment to compile, test, and execute software dependent on the packages installed with `ape`.
- Status and other reports.

Dependencies between packages are handled during all the processes.

For more information, consult the documentation listed below, especially the [Ape user guide](#). We also have a [PDF version](#) of the documentation available.

The development of `ape` is hosted at the [ape trac pages](#).

- [Ape user guide](#)
 - [Getting started](#)
 - [Details of the `ape` command](#)
 - [Troubleshooting](#)
 - [The `ape` configuration files](#)
 - [Customizing packages](#)
- [Extending `ape`](#)
 - [Simple packages](#)
 - [Customizing `ApeTools.Build.Package`](#)
- [Code Documentation](#)
 - [Support utilities: the `ApeTools` package and sub-modules](#)
 - [Specialized build support for individual packages: `ApePackages`](#)
 - [The main script](#)

Indices and tables

- [Index](#)
- [Module Index](#)
- [Search Page](#)

[Ape documentation »](#)

[next](#) | [modules](#) | [index](#)

Ape configuration

● INI style configuration ~/.aperc

```
[DEFAULT]
# Installation area
base = %(home)s/auger/software/ApeInstalled
# Build area (e.g., local disk or RAM disk)
# Only set
##build = /tmp/ape-build-%(user)s

[ape]
jobs = 8 # number of cores HT threads (-2)
mirrors = mx us

# place for source files
##distfiles = %(home)s/distfiles
```

Installation with Ape

- Ape provides several install targets
 - offline
 - externals (for building offline separately from source)
 - cdas
 - cdasuser
 - cdasexternals
- Install command, run in ape's source directory:
 - `./ape install offline`
 - or
 - `./ape install externals`

Some install times

Platform	Cores	GHz	Notes	Ext	Offl
Macbook Pro '13	2+2 / 12	2.3	SSD, no G4, clang 3.7 equiv	23 min	9 min
Debian testing	6+6 / 24	3.2	Ramdisk, gcc 5.4	26:40	8:31
Debian testing	6+6 / 24	3.2	Ramdisk, no G4 gcc 5.4	20:18	9 min
Ubuntu 16.04	6 / 12	3.2	Virt, gcc 5.3	38:30	16:15
Centos 6	6 / 12	3.2	Virt, gcc 4.4.7	31:30	13:44

Tested up to gcc 8.2

Ape testing in containers

- More than 10 different distributions/releases
 - Extra containers for v3r3 testing
- Less space, easier scripting than VM
- Custom for testing, could be basis for distribution
 - f: fedora
 - c: **centos**, c-?t7: **centos, with devtoolset 7**
 - d: **debian**, d-tx: extended compiler set in testing
 - u: **ubuntu**, m: mint
 - a: arch

lukasellen/augerdev	a-latest	9de1b41d88e2	3 days ago	1.32GB
lukasellen/augerdev	c-6	4c93cb202833	3 days ago	418MB
lukasellen/augerdev	c-6t7	a795ecc980de	3 days ago	719MB
lukasellen/augerdev	c-7	8dfe6f654f48	3 days ago	448MB
lukasellen/augerdev	c-7t7	ad26b6ff6bf7	3 days ago	760MB
lukasellen/augerdev	d-stable	176c87db28ea	3 days ago	604MB
lukasellen/augerdev	d-testing	da9bd93398a9	3 days ago	719MB
lukasellen/augerdev	d-tx	ccb155ec00e1	3 days ago	945MB
lukasellen/augerdev	f-28	c034bbc3a844	3 days ago	630MB
lukasellen/augerdev	f-29	be45ed825313	3 days ago	726MB
lukasellen/augerdev	u-14.04	7757003d31f5	3 days ago	461MB
lukasellen/augerdev	u-16.04	6243ed692085	3 days ago	542MB
lukasellen/augerdev	u-18.04	7049f394906e	3 days ago	574MB

Activating the Ape Environment

- Ape installs packages in separate hierarchy
 - Isolation from other software
 - Separation of releases
- Have to activate to set PATH variables etc with one of

```
eval ` /path/to/ape sh offline`  
eval ` /path/to/ape sh externals`  
eval ` /path/to/ape csh offline`  
eval ` /path/to/ape csh offline`
```

Depending on the package you want to activate and the shell you use

- Place command in **startup files**, e.g. `~/.bashrc` or **define alias** for easy invocation or use some **setup helper**

Advanced Configuration

- Ape allows for extended customisation
 - See manual
 - Voids warranties
 - Not tested
 - No support
 - Only for experts
- Change dependencies
 - Suppress the installation of a package
 - Change version of package used
- Modify build options for packages

Ape for Developers

- If you work on the offline source, you need the externals to be able to build

`./ape install externals`

- You might have to install a developer release

`ape-...dev-YYYYMMDD.tar.bz2`

- NB: Developer ape **does NOT provide offline**

- In rare cases, you might have to use ape from the git repository

`git clone git://devel.auger.unam.mx/ape.git`

- You risk getting untested externals

Only do this if you know what you're doing

- Follow instructions on how to install offline from SVN or git

Externals updating

- All externals compiled with C++11 option
 - Consider: C++ 14
- All releases: new auger user
- CDAS update
- Need to look into root6
 - Last root5 commit (v5-34-38 / 39): March 2018

```
commit e68a976e0bfe845b0d3ba0fc550c43c7944c56ea (origin/v5-34-00-patches, v5-34-00-patches)
```

```
Author: Axel Naumann <Axel.Naumann@cern.ch>
```

```
Date: Mon Mar 12 15:46:18 2018 +0100
```

```
Update ROOT version files to v5.34/39.
```

```
commit 250a360349e60dc2dc187ecaa63fc9f5f81a9d74 (tag: v5-34-38)
```

```
Author: Axel Naumann <Axel.Naumann@cern.ch>
```

```
Date: Mon Mar 12 13:41:02 2018 +0100
```

```
Update ROOT version files to v5.34/38.
```

Setup tool example

<https://github.com/lukasnellen/environment-setup>

Setup hook

- Hook file, include in your setup, e.g, `~/.bashrc`
- Invoke: `setup auger`

```
# Simple setup script invoker for bash with completion
# Author: Lukas Nellen
#
# Setup scripts are in $HOME/setup.d
# They get sourced to modify the current shell's environment
#
# Special case if $PYENV set to point to a directory with python virtual envs:
#   setup py "the-env"
# will activate the python environment "the-env" in ${PYENV}
#
# Configure automatic name completion

_setup_complete () {
    local cur="$2"
    if [ ${COMP_CWORD} -gt 1 -a -f "${HOME}/.setup.d/${COMP_WORDS[1]}.compgen" ]
    then
        COMPREPLY=( $(bash "${HOME}/.setup.d/${COMP_WORDS[1]}.compgen" "${cur}" ) )
    else
        COMPREPLY=( $(builtin cd $HOME/.setup.d && \
            compgen -f -X "[~.]*" -- "${cur}" ) )
    fi
}

function setup () {
    script=$1
    shift
    source $HOME/.setup.d/${script} "$@"
}

complete -o nospace -F _setup_complete setup
```

Auger setup script

```
# Lukas's local auger environment

export SOFTWAREBASE=$HOME/auger/software/
eclipse/Offline-Install
export DISTBASE=$HOME/auger/software/eclipse/Ape

cfg_level=3
case "${1}" in
    # ape config only
    -a) cfg_level=0
        shift
        ;;
    # CDAS externals
    -c) cfg_level=1
        shift
        ;;
    # Offline externals
    -x) cfg_level=2
        shift
        ;;
    # Offline test install
    -t) export SOFTWAREBASE=$HOME/auger/
software/eclipse/Offline/00-Install
        shift
        ;;
esac

export APERC=$HOME/.aperc.auger${1}
if [ ! -e "${APERC}" ]; then
    echo === Ape rc file ${APERC} does not exist
    # this is sourced in a function, return not
    exit
    return
fi

#level 1: cdas externals (only)
if [ "${cfg_level}" -eq 1 ]
then
    eval ` $DISTBASE/ape sh cdasexternals `
    return
fi

#level 2: externals
if [ "${cfg_level}" -ge 2 ]
then
    eval ` $DISTBASE/ape sh externals `
fi

#level2: full offline
if [ "${cfg_level}" -ge 3 ]
then
    eval ` $SOFTWAREBASE/bin/auger-offline-config --
env-sh `
fi
```

```
# -*- shell-script -*- (completion)
```

```
builtin cd $HOME
list=$(for i in .aperc.auger*; do [ ${i:${#i}-1} != "~" ] && echo $
{i/.aperc.auger/}; done )
echo $(compgen -W "-a -c -x -t ${list}" -- "${1}" )
```

Other setup helpers

- Python virtual env
- root
- HAWC
- CORSIKA
- ...