

Attila User's Manual

Jean-Paul Chaput
System Administrator
Pierre & Marie Curie University, LIP6
ASIM Department

Attila User's Manual
by Jean-Paul Chaput

Published September 2002

Table of Contents

I. *Alliance - attila* User’s Manual..... v
 attila.....7

I. *Alliance* - attila User's Manual

attila

Name

attila — A Tool Installer

Synopsis

```
attila [-h] [-S] [-U] [-F] [-A]
[--help] [--ssh] [--user] [--asim]
[--prefix=INSTALL_DIR] [--builddir=BUILD_DIR]
[--tool=name1] [--tool=name2...]
[-c {configure_arg1} [configure_arg2 ...]]
[-m {make_arg1} [make_arg2...]]
```

Description

attila automates the process of compiling and installing one or more *Alliance* tools. The tool can be installed either in the user's account (during the development stage) or in the *Alliance* system wide tree (for instance */asim/alliance*) when a new version is made available to all.

attila proceed with the following steps :

1. Checks if the sources of tools are present in the user's account. If not, check them out from the *Alliance* CVS tree. Note that you must have access to it.
2. In case of **--asim** or **--full** installations, attila will fork itself on one Linux computer (currently *bip*) and on one Solaris computer (*beny*). As to connect on thoses computer it will uses *rsh* so you must setup your *~/.rhosts* to access them whitouth passwords. You also can uses *ssh* (but the procedure to allow automatic login is more complicated).
3. Run *autostuff* for the tool in the *~/alliance/src* directory.
4. Run *configure* in the build directory (see below).
5. Install the tool in the local install directory (see below) or in the system-wide *Alliance* directory rooted under */asim/alliance*.

After an **--asim install:** the build directory tree of the tool will be removed to avoid messing with further local installations.

Directory Structure

attila relies on the following tree structure : (all paths below are given relative to the user's home directory)

- *~/alliance/src* where the tools sources are to be found.
- *~/alliance/Linux/build/\$TOOL* : the top directory under which the tool will be compiled for Linux. This is where the *configure* script will be run.
- *~/alliance/Linux/install* : the top of the install tree when the tool is compiled locally for Linux. Under this directory you will found (at least) : *./bin*, *./lib* and *./include*.

- `~/alliance/Solaris/build/$TOOL` : the tool's build directory for Solaris.
- `~/alliance/Solaris/install` : top of the local install tree for Solaris.

CVS checkout

If the sources of the requested tool(s) are not found under `~/alliance/src/` `attila` will try to check them out. So, as says above, you must have access rights to the *Alliance* CVS tree.

In addition to the tool(s) sources, it will also checks for the minimal set of files needed for configure to run. As for now :

- `autostuff`
- `alliance.m4`
- `motif.m4`
- `Makefile.am`

Guessing CVSROOT

The root of the CVS tree will be set according to the following rules :

1. Uses the user's environment variable `CVSROOT` if sets.
2. Uses the `attila` default value sets in `attila.conf` (variable `ATTILA_CVSROOT`).

Guessing ALLIANCE_TOP

The root of the *Alliance* installed distribution tree will be set according to the following rules :

1. Uses the user's environment variable `ALLIANCE_TOP` if sets.
2. Uses the `attila` default value sets in `attila` itself (variable `ATTILA_ALLIANCE_TOP`).

ALLIANCE_TOP: is set in `attila` itself because its value is a prerequisite to load the configuration file `attila.conf` which is in the directory `$ALLIANCE_TOP/etc/`.

Arguments

`attila` accepts the followings arguments :

- `-h, --help` : print help.

- **-S, --ssh** : uses `ssh` instead of `rsh` to connect to the remote computers (in case of **--asim** or **--full**).
- **-U, --user** : perform a local installation.
- **-F, --full** : install for all available architectures (currently Linux and Solaris).
- **-A, --asim** : install in the system-wide directory (`/asim/alliance`).
- **--prefix=INSTALL_DIR** : override the default installation directory.
- **--builddir=BUILD_DIR** : override the default building directory.
- **--tool=name1** : name of the tool to be installed.
- **-c- configure_arg1** : all arguments following **-c-** (until a **-m-** is encountered) are passed *as is* to the subsequent call to `configure`. For example, if you want to first use your local libraries, give **--enable-devel**.
- **-m- make_arg1** : all arguments following **-m-** are passed *as is* to the subsequent call to `make`. If no **-m-** argument is given, then **install** is assumed. If you want to completely uninstall a tool and clean its build directory you can pass **uninstall clean**.

Configuration file **attila.conf**

The configuration file is located in `$ALLIANCE_TOP/etc`. This file is to be read by the `sh shell`. It sets up the following variables :

- **LINUX_TARGET** : the computer where to compile for the Linux architecture (default `bip`).
- **LINUX_CC** : the name or full path to the C compiler for Linux system (default `gcc3`).
- **LINUX_CXX** : the name or full path to the C++ compiler for Linux system (default `g++3`).
- **SOLARIS_TARGET** : the computer where to compile for the Solaris architecture (default `beny`).
- **SOLARIS_CC** : the name or full path to the C compiler for Solaris system (default `/usr/local/gcc-3.0.4/bin/g++3`).
- **SOLARIS_CXX** : the name or full path to the C++ compiler for Solaris system (default `/usr/local/gcc-3.0.4/bin/g++3`).
- **ATTILA_CVSROOT** : the root of the *Alliance* CVS tree (default `/users/outil/alliance/cvsroot`).
- **CVS_STARTUP_FILES** : the minimal set of files needed to run `configure`.

Examples

Compile & install `nero` tool on the local computer (must be either a Linux or a Solaris one) :

```
$ attila --tool=nero
```

Compile & install `nero` tool for all architectures (currently only Linux and Solaris are supported) :

```
$ attila --full --tool=nero
```

Compile & install `nero` tool in the system-wide directory (a new version for everyone to use) :

```
$ attila --asim --tool=nero
```

Compile & install `mbk` then `genlib` (the order is significant) in a row for a local install on the current computer :

```
$ attila --tool=mbk --tool=genlib
```

Compile `nero` tool and link it against the locally installed libraries (if any).

```
$ attila --tool=nero -c- --enable-devel
```

Remove `poire` tool from the system-wide tree. (`poire` is the old name of `nero`).

```
$ attila --asim --tool=poire -m- uninstall
```