
conduct Documentation

Release 0.0.1

Alexander Lenz

November 27, 2015

1	Introduction	3
2	Getting started	5
3	Build procedure	7
4	Conduct API	9
4.1	Build steps	9
4.2	Build chain	14
4.3	Colorizing stuff	14
4.4	Conduct configuration access	14
4.5	Custom logging	14
4.6	Build step and chain parameters	15
4.7	Utility stuff	17
5	Indices and tables	19
	Python Module Index	21

Contents:

Introduction

conduct is a build system that is intended to be used as a standalone script (conduct.py) or in conjunction with some load balancer (e.g. via Jenkins).

The idea is to checkout/install conduct to some system, give it a build chain config and the necessary parameters, and everything else shall be done automatically. That means:

- Bootstrapping of necessary tools
- Configuring necessary tools
- Building by using the given chain config and parameters
- Cleanup after the build
- Rollback the bootstrapped packages (if desired)

Basically, conduct's goals are:

- easy to configure the build process
- easy to use the build process
- easy to extend for special needs
- consistent logging
- automatic bootstrapping
- great selection of standard build steps

Getting started

Build procedure

Conduct API

4.1 Build steps

4.1.1 Base classes

class `conduct.buildsteps.base.BuildStep` (*name*, *paramValues*, *chain=None*)

Parameters

condition (*str*)

Precondition for this build step. Step will be skipped if not fulfilled. Default value: `''`.

description (*str*)

Build step description. Default value: `'Undescribed'`.

loglevel (one of [`'debug'`, `'info'`, `'warning'`, `'error'`])

Log level. Default value: `'info'`.

retries (*int*)

Number of retries to execute the build step. Default value: 0.

class `conduct.buildsteps.base.BuildStepMeta`

Bases: `type`

Meta class for merging parameters and outparameters within the inheritance tree.

4.1.2 Debian specific build steps

class `conduct.buildsteps.deb.Debootstrap` (*name*, *paramValues*, *chain=None*)

Bases: `BuildStep`

This build step bootstraps a basic debian system to the given directory.

Parameters

arch (*str*, **mandatory in setup**)

Desired architecture.

destdir (*str*, **mandatory in setup**)

Destination directory.

distribution (*str*, **mandatory in setup**)

Desired distribution.

includes (a list of str)

Packages to include. Default value: [].

Parameters inherited from the base classes: *condition*, *description*, *loglevel*, *retries*

class `conduct.buildsteps.deb.InstallDebPkg` (*name*, *paramValues*, *chain=None*)

Bases: *BuildStep*

Parameters**chrootdir (str)**

Chroot directory (if desired). Default value: ''.

depsonly (bool)

Install dependencies only. Default value: False.

pkg (str, mandatory in setup)

Package to install.

Parameters inherited from the base classes: *condition*, *description*, *loglevel*, *retries*

class `conduct.buildsteps.deb.PBuilderExecCmds` (*name*, *paramValues*, *chain=None*)

Bases: *BuildStep*

Execute commands inside a pbuilder jail.

Parameters**cmds (a list of str, mandatory in setup)**

List of commands to execute.

config (str)

Pbuilder config file (pbuilderrc). Default value: '/etc/pbuilderrc'.

save (bool)

Save jail after execution. Default value: False.

Parameters inherited from the base classes: *condition*, *description*, *loglevel*, *retries*

class `conduct.buildsteps.deb.Pdebuild` (*name*, *paramValues*, *chain=None*)

Bases: *BuildStep*

Build debian package via pdebuild.

Parameters**config (str)**

Pbuilder config file (pbuilderrc). Default value: '/etc/pbuilderrc'.

sourcedir (str, mandatory in setup)

Source directory.

Parameters inherited from the base classes: *condition*, *description*, *loglevel*, *retries*

4.1.3 Device file build steps

class `conduct.buildsteps.dev.DevMapper` (*name*, *paramValues*, *chain=None*)

Bases: *BuildStep*

This build step uses kpartx (devmapper) to map the partitions of the given device to own device files.

Parameters**dev (str, mandatory in setup)**

Path to the device file.

Parameters inherited from the base classes: *condition*, *description*, *loglevel*, *retries*

class `conduct.buildsteps.dev.Partitioning` (*name*, *paramValues*, *chain=None*)
Bases: *BuildStep*

Parameters

dev (*str*, mandatory in setup)
 Path to the device file.

partitions (a list of Referencer or int, mandatory in setup)
 List of partition sizes (in MB).

Parameters inherited from the base classes: *condition*, *description*, *loglevel*, *retries*

4.1.4 File system build steps

class `conduct.buildsteps.fs.CopyPath` (*name*, *paramValues*, *chain=None*)
Bases: *BuildStep*

This build step copies a path to a given destination path. Shell wildcards are supported!

Parameters

destination (*str*, mandatory in setup)
 Destination path.

source (*str*, mandatory in setup)
 Source path.

Parameters inherited from the base classes: *condition*, *description*, *loglevel*, *retries*

class `conduct.buildsteps.fs.CreateFileSystem` (*name*, *paramValues*, *chain=None*)
Bases: *BuildStep*

This build step creates the desired file system on the given device. Used tool: mkfs

Parameters

dev (*str*, mandatory in setup)
 Path to the device file.

fstype (one of 'bfs', 'cramfs', 'ext2', 'ext3', 'ext4', 'fat', 'ntfs', 'vfat', mandatory in setup)
 Desired file system.

Parameters inherited from the base classes: *condition*, *description*, *loglevel*, *retries*

class `conduct.buildsteps.fs.MakeDirs` (*name*, *paramValues*, *chain=None*)
Bases: *BuildStep*

This build step create the desired directories.

Parameters

dirs (a list of *str*, mandatory in setup)
 List of directories.

removeoncleanup (*bool*)
 Remove the directories on clenup. Default value: True.

Parameters inherited from the base classes: *condition*, *description*, *loglevel*, *retries*

class `conduct.buildsteps.fs.Mount` (*name, paramValues, chain=None*)

Bases: *BuildStep*

This build step mounts given device to given mount point.

Parameters

dev (*str, mandatory in setup*)

Path to the device file.

mountpoint (*str, mandatory in setup*)

Path to the mount point.

Parameters inherited from the base classes: *condition, description, loglevel, retries*

class `conduct.buildsteps.fs.MovePath` (*name, paramValues, chain=None*)

Bases: *BuildStep*

This build step moves/renames a path to a given destination path. Shell wildcards are supported!

Parameters

destination (*str, mandatory in setup*)

Destination path.

source (*str, mandatory in setup*)

Source path.

Parameters inherited from the base classes: *condition, description, loglevel, retries*

class `conduct.buildsteps.fs.RmPath` (*name, paramValues, chain=None*)

Bases: *BuildStep*

Parameters

path (*str, mandatory in setup*)

Path to remove.

recursive (*bool*)

Remove recursive. Default value: True.

Parameters inherited from the base classes: *condition, description, loglevel, retries*

class `conduct.buildsteps.fs.TmpDir` (*name, paramValues, chain=None*)

Bases: *BuildStep*

Parameters

parentdir (*str*)

Path to parent directory. Default value: `' /tmp'`.

Parameters inherited from the base classes: *condition, description, loglevel, retries*

class `conduct.buildsteps.fs.WriteFile` (*name, paramValues, chain=None*)

Bases: *BuildStep*

Parameters

append (*bool*)

Append to the file (if existing). Default value: False.

content (*str, mandatory in setup*)

Content to write.

path (*str, mandatory in setup*)

Path to target file.

Parameters inherited from the base classes: *condition, description, loglevel, retries*

4.1.5 Generic build steps

class `conduct.buildsteps.generic.Calculation` (*name, paramValues, chain=None*)

Bases: *BuildStep*

Build step to do some calculation.

Parameters

formula (*str, mandatory in setup*)

Formula to calculate.

Parameters inherited from the base classes: *condition, description, loglevel, retries*

class `conduct.buildsteps.generic.Config` (*name, paramValues, chain=None*)

Bases: *BuildStep*

Build step to read given configuration file.

Parameters

format (*one of 'ini', 'py', 'auto'*)

Format of config file. Default value: 'auto'.

path (*str*)

Path to the configuration file. Default value: '.'.

Parameters inherited from the base classes: *condition, description, loglevel, retries*

class `conduct.buildsteps.generic.Map` (*name, paramValues, chain=None*)

Bases: *BuildStep*

Maps a value to another one. Supports regex matching and priority matching by order.

Parameters

input (*str, mandatory in setup*)

Input to map.

mapping (*OrderedDict, mandatory in setup*)

Mapping ordered dict.

Parameters inherited from the base classes: *condition, description, loglevel, retries*

class `conduct.buildsteps.generic.TriggerCleanup` (*name, paramValues, chain=None*)

Bases: *BuildStep*

Build step that triggers the cleanup of another step.

Parameters

step (*str, mandatory in setup*)

Step to clean up.

Parameters inherited from the base classes: *condition, description, loglevel, retries*

4.1.6 System call build steps

class `conduct.buildsteps.syscall.ChrootedSystemCall` (*name, paramValues, chain=None*)

Bases: *BuildStep*

Build step to execute given shell command in a chroot environment.

Parameters

chrootdir (str)

Chroot directory. Default value: ' . ' .

command (str, mandatory in setup)

command to execute.

Parameters inherited from the base classes: *condition*, *description*, *loglevel*, *retries*

class `conduct.buildsteps.syscall.SystemCall` (*name*, *paramValues*, *chain=None*)

Bases: *BuildStep*

Build step to execute given shell command.

Parameters

command (str, mandatory in setup)

command to execute.

workingdir (str)

Working directory for command execution. Default value: ' . ' .

Parameters inherited from the base classes: *condition*, *description*, *loglevel*, *retries*

4.2 Build chain

class `conduct.chain.Chain` (*name*, *paramValues*)

build ()

parameters

4.3 Colorizing stuff

Console coloring handlers.

`conduct.colors.colorcode` (*name*)

`conduct.colors.colorize` (*name*, *text*)

`conduct.colors.nocolor` ()

4.4 Conduct configuration access

4.5 Custom logging

class `conduct.loggers.ColoredConsoleHandler`

Bases: *StreamHandler*

A handler class that writes colored records to standard output.

emit (*record*)

```

class conduct.loggers.ConductLogger (*args, **kwargs)
    Bases: Logger

    getChild (suffix, ownDir=False)

    maxLogNameLength = 0

class conduct.loggers.ConsoleFormatter (fmt=None, datefmt=None, colorize=None)
    Bases: Formatter

    A lightweight formatter for the interactive console, with optional colored output.

    format (record)

    formatException (exc_info)

    formatTime (record, datefmt=None)

class conduct.loggers.LogfileFormatter (fmt=None, datefmt=None)
    Bases: Formatter

    The standard Formatter does not support milliseconds with an explicit timestamp format. It also doesn't show
    the full traceback for exceptions.

    extended_traceback = True

    formatException (ei)

    formatTime (record, datefmt=None)

class conduct.loggers.LogfileHandler (directory, filenameprefix, dayfmt='%Y-%m-%d')
    Bases: StreamHandler

    Logs to log files with a date stamp appended, and rollover on midnight.

    close ()

    doRollover ()

    emit (record)

    enable (enabled)

    filter (record)

    getChild (name)

class conduct.loggers.StreamHandler (stream=None)
    Bases: Handler

    Reimplemented from logging: remove cruft, remove bare excepts.

    emit (record)

    flush ()

conduct.loggers.format_extended_frame (frame)

conduct.loggers.format_extended_traceback (etype, value, tb)

```

4.6 Build step and chain parameters

```

class conduct.param.Dataholder (modelDict)

```

class `conduct.param.Parameter` (*type=<type 'str'>, description='Undescribed', default=None*)
Some kind of parameter.

No default means mandatory.

validate (*value*)

class `conduct.param.Referencer` (*fmt*)

evaluate (*chain, valType=<type 'str'>*)

`conduct.param.absolute_path` (*val=''*)
an absolute file path

`conduct.param.anytype` (*val=None*)
any value

`conduct.param.callableobj` (*obj*)

`conduct.param.convdoc` (*conv*)

class `conduct.param.dictof` (*keyconv, valconv*)

`conduct.param.expanded_path` (*val=''*)

class `conduct.param.floatrange` (*fr, to=None*)

`conduct.param.host` (*val=''*)
a host[:port] value

class `conduct.param.intrange` (*fr, to*)

`conduct.param.ipv4` (*val='0.0.0.0'*)
a IP v4 address

`conduct.param.limits` (*val=None*)
a tuple of lower and upper limit

class `conduct.param.listof` (*conv*)

`conduct.param.mailaddress` (*val=None*)
a valid mail address

class `conduct.param.none_or` (*conv*)

class `conduct.param.nonemptylistof` (*conv*)

`conduct.param.nonemptystring` (*s*)
a non-empty string

class `conduct.param.oneof` (**vals*)

class `conduct.param.oneofdict` (*vals*)

class `conduct.param.referencer_or` (*conv*)

`conduct.param.relative_path` (*val=''*)
a relative path, may not use ../ tricks

`conduct.param.subdir` (*val=''*)
a relative subdir (a string NOT containing any path.sep)

class `conduct.param.tupleof` (**types*)

4.7 Utility stuff

class `conduct.util.AttrStringifier`

class `conduct.util.OrderedAttrDict` (**args, **kwargs*)

Bases: `OrderedDict`

`conduct.util.analyzeSystem()`

`conduct.util.chainNameToPath` (*name*)

`conduct.util.chainPathToName` (*path*)

`conduct.util.chrootedSystemCall` (*chrootDir, cmd, sh=True, mountPseudoFs=True, log=None*)

`conduct.util.ensureDirectory` (*dirpath*)

`conduct.util.getDefaultConfigPath()`

`conduct.util.importFromPath` (*import_name, prefixes=(), log=None*)

Imports an object based on a string.

The should be formatted like: `imp.path.to.mod.objname`

`conduct.util.loadChainConfig` (*chainName*)

`conduct.util.loadChainDefinition` (*chainName, app=None*)

`conduct.util.loadPyFile` (*path, ns=None*)

`conduct.util.logMultipleLines` (*strOrList, logFunc=None*)

`conduct.util.mount` (*dev, mountpoint, flags='', log=None*)

`conduct.util.systemCall` (*cmd, sh=True, log=None*)

`conduct.util.umount` (*mountpoint, flags='', log=None*)

Indices and tables

- `genindex`
- `modindex`
- `search`

C

- `conduct.buildsteps.base`, [9](#)
- `conduct.buildsteps.deb`, [9](#)
- `conduct.buildsteps.dev`, [10](#)
- `conduct.buildsteps.fs`, [11](#)
- `conduct.buildsteps.generic`, [13](#)
- `conduct.buildsteps.syscall`, [13](#)
- `conduct.chain`, [14](#)
- `conduct.colors`, [14](#)
- `conduct.loggers`, [14](#)
- `conduct.param`, [15](#)
- `conduct.util`, [17](#)

A

`absolute_path()` (in module `conduct.param`), 16
`analyzeSystem()` (in module `conduct.util`), 17
`anytype()` (in module `conduct.param`), 16
`AttrStringifier` (class in `conduct.util`), 17

B

`build()` (`conduct.chain.Chain` method), 14
`BuildStep` (class in `conduct.buildsteps.base`), 9
`BuildStepMeta` (class in `conduct.buildsteps.base`), 9

C

`Calculation` (class in `conduct.buildsteps.generic`), 13
`callableobj()` (in module `conduct.param`), 16
`Chain` (class in `conduct.chain`), 14
`chainNameToPath()` (in module `conduct.util`), 17
`chainPathToName()` (in module `conduct.util`), 17
`ChrootedSystemCall` (class in `conduct.buildsteps.syscall`), 13
`chrootedSystemCall()` (in module `conduct.util`), 17
`close()` (`conduct.loggers.LogfileHandler` method), 15
`colorcode()` (in module `conduct.colors`), 14
`ColoredConsoleHandler` (class in `conduct.loggers`), 14
`colorize()` (in module `conduct.colors`), 14
`conduct.buildsteps.base` (module), 9
`conduct.buildsteps.deb` (module), 9
`conduct.buildsteps.dev` (module), 10
`conduct.buildsteps.fs` (module), 11
`conduct.buildsteps.generic` (module), 13
`conduct.buildsteps.syscall` (module), 13
`conduct.chain` (module), 14
`conduct.colors` (module), 14
`conduct.loggers` (module), 14
`conduct.param` (module), 15
`conduct.util` (module), 17
`ConductLogger` (class in `conduct.loggers`), 14
`Config` (class in `conduct.buildsteps.generic`), 13
`ConsoleFormatter` (class in `conduct.loggers`), 15
`convdoc()` (in module `conduct.param`), 16
`CopyPath` (class in `conduct.buildsteps.fs`), 11

`CreateFileSystem` (class in `conduct.buildsteps.fs`), 11

D

`Dataholder` (class in `conduct.param`), 15
`Debootstrap` (class in `conduct.buildsteps.deb`), 9
`DevMapper` (class in `conduct.buildsteps.dev`), 10
`dictof` (class in `conduct.param`), 16
`doRollover()` (`conduct.loggers.LogfileHandler` method), 15

E

`emit()` (`conduct.loggers.ColoredConsoleHandler` method), 14
`emit()` (`conduct.loggers.LogfileHandler` method), 15
`emit()` (`conduct.loggers.StreamHandler` method), 15
`enable()` (`conduct.loggers.LogfileHandler` method), 15
`ensureDirectory()` (in module `conduct.util`), 17
`evaluate()` (`conduct.param.Referencer` method), 16
`expanded_path()` (in module `conduct.param`), 16
`extended_traceback` (`conduct.loggers.LogfileFormatter` attribute), 15

F

`filter()` (`conduct.loggers.LogfileHandler` method), 15
`floatrange` (class in `conduct.param`), 16
`flush()` (`conduct.loggers.StreamHandler` method), 15
`format()` (`conduct.loggers.ConsoleFormatter` method), 15
`format_extended_frame()` (in module `conduct.loggers`), 15
`format_extended_traceback()` (in module `conduct.loggers`), 15
`formatException()` (`conduct.loggers.ConsoleFormatter` method), 15
`formatException()` (`conduct.loggers.LogfileFormatter` method), 15
`formatTime()` (`conduct.loggers.ConsoleFormatter` method), 15
`formatTime()` (`conduct.loggers.LogfileFormatter` method), 15

G

`getChild()` (conduct.loggers.ConductLogger method), 15
`getChild()` (conduct.loggers.LogfileHandler method), 15
`getDefaultConfigPath()` (in module conduct.util), 17

H

`host()` (in module conduct.param), 16

I

`importFromPath()` (in module conduct.util), 17
`InstallDebPkg` (class in conduct.buildsteps.deb), 10
`inrange` (class in conduct.param), 16
`ipv4()` (in module conduct.param), 16

L

`limits()` (in module conduct.param), 16
`listof` (class in conduct.param), 16
`loadChainConfig()` (in module conduct.util), 17
`loadChainDefinition()` (in module conduct.util), 17
`loadPyFile()` (in module conduct.util), 17
`LogfileFormatter` (class in conduct.loggers), 15
`LogfileHandler` (class in conduct.loggers), 15
`logMultipleLines()` (in module conduct.util), 17

M

`mailaddress()` (in module conduct.param), 16
`MakeDirs` (class in conduct.buildsteps.fs), 11
`Map` (class in conduct.buildsteps.generic), 13
`maxLogNameLength` (conduct.loggers.ConductLogger attribute), 15
`Mount` (class in conduct.buildsteps.fs), 11
`mount()` (in module conduct.util), 17
`MovePath` (class in conduct.buildsteps.fs), 12

N

`nocolor()` (in module conduct.colors), 14
`none_or` (class in conduct.param), 16
`nonemptylistof` (class in conduct.param), 16
`nonemptystring()` (in module conduct.param), 16

O

`oneof` (class in conduct.param), 16
`oneofdict` (class in conduct.param), 16
`OrderedAttrDict` (class in conduct.util), 17

P

`Parameter` (class in conduct.param), 15
`parameters` (conduct.chain.Chain attribute), 14
`Partitioning` (class in conduct.buildsteps.dev), 11
`PBuilderExecCmds` (class in conduct.buildsteps.deb), 10
`Pdebuild` (class in conduct.buildsteps.deb), 10

R

`Referencer` (class in conduct.param), 16
`referencer_or` (class in conduct.param), 16
`relative_path()` (in module conduct.param), 16
`RmPath` (class in conduct.buildsteps.fs), 12

S

`StreamHandler` (class in conduct.loggers), 15
`subdir()` (in module conduct.param), 16
`SystemCall` (class in conduct.buildsteps.syscall), 14
`systemCall()` (in module conduct.util), 17

T

`TmpDir` (class in conduct.buildsteps.fs), 12
`TriggerCleanup` (class in conduct.buildsteps.generic), 13
`tupleof` (class in conduct.param), 16

U

`umount()` (in module conduct.util), 17

V

`validate()` (conduct.param.Parameter method), 16

W

`WriteFile` (class in conduct.buildsteps.fs), 12