

# FrontPage

## Welcome to the Apache Commons Wiki

bl o c k e d U RL	This is the <a href="#">Apache Wiki</a> for the <a href="#">Apache Commons</a> project and is maintained by the <a href="#">Commons</a> community. To edit pages, visit login near the top right corner of any page to create a user profile or to login. Then send a brief request ( <b>including your wiki username</b> ) to the <a href="#">Commons dev@ list</a> asking to be added to the Contributors Group. You can then edit the pages in this Wiki. Notifications of all changes you make will be sent to the dev@commons mailing list, so we will be aware of your changes and we will happily correct any small mistakes that you might make.
--	--

We're a [Commons](#) community, dedicated to creating reusable library components in Java. Apache Commons mainly uses [Subversion](#) as its version control system, however some components now use [Git](#), and Commons components are available as read-only mirrors on [Github](#).

Welcome: [CommonsEtiquette](#) | [CommonsResources](#) | [ArticlesAndTutorials](#)

Developers: [GettingInvolved](#) | [UsingGIT](#) | [UsingSVN](#) | [CodeStyle](#) | [UseClirrWithShade](#)

Committers: [CommonsPeople](#) | [ComponentPlans](#) | [CommonsCommitters](#) | [CommonsOsgi](#) | [UsingNexus](#) | [CommonsGroupids](#) | [Maven3Plan](#)

Bug Reports: [Jira](#)

Security Reports: [Security](#)

## Components

On this wiki:

- [BeanUtils](#) - Utilities for manipulating java beans, including copying properties, cloning, invoking methods and type conversion.
- [Betwixt](#) - Services for mapping JavaBeans to XML documents, and vice versa.
- [Chain](#) - A "Chain of Responsibility" pattern implementation for organizing complex processing flows.
- [CLI](#) - Command Line Interface library for processing command-line options and arguments.
- [Codec](#) - General encoding/decoding algorithms. Includes some phonetic encoders, Hex, Base64, and a URL encoder.
- [Collections](#) - Builds upon the Java Collections Framework of List, Set and Map to provide many more implementations, new collections and abstract base classes.
- [Configuration](#) - Tools to assist in the reading of configuration/preferences files in various formats
- [Compress](#) - Defines an API for working with archives and compression formats including but not limited to: tar,gz, ar, cpio, zip and bzip2 files
- [Daemon](#) - Run (java) applications as windows service or UNIX daemon.
- [DBCP](#) - Database connection pooling services.
- [Digester](#) - An XML-to-Java-object mapping utility commonly used for parsing XML configuration files.
- [Email](#) - Simplify email sending with Java. Built on top of the Java Mail API.
- [Exec](#) - A Java library to execute external processes from within a Java application
- [FileUpload](#) - Makes it easy to add robust, high-performance, file upload capability to your servlets and web applications
- [Functor](#) - Supports functional programming in Java
- [HttpClient](#) - A framework for working with the client-side of the HTTP protocol.
- [IO](#) - Utility classes, stream implementations, file filters and endian classes to assist in input/output operations.
- [JCI](#) - Java Compiler Interface
- [JEXL](#) - A velocity-esque expression interpreter
- [Jelly](#) - An XML-based scripting language
- [XPath](#) - Facilities for applying xpath expressions to graphs of objects of all kinds: Java beans, Maps, Servlet contexts, DOM etc.
- [Lang](#) - Common utilities that should be provided by the JDK, including String handling, Object and Date helpers, nested exceptions and enumerated types.
- [Launcher](#) - A cross platform Java application launcher which eliminates the need for a batch or shell script to launch a Java class.
- [Logging](#) - A thin, modular bridging API with out-of-the-box support for the most well known logging systems.
- [Math](#) - Lightweight, self-contained mathematics and statistics components.
- [Modeler](#) - Mechanisms to create Model MBeans compatible with the Java Management Extensions (JMX) specification.
- [Net](#) - A collection of classes implementing various network protocols such as FTP, NNTP, SMTP, Telnet.
- [Pool](#) - A generic object pooling interface, a toolkit for creating modular object pools and several general purpose pool implementations.
- [Proxy](#) - A Java library for dynamic proxying
- [SCXML](#) - A Java implementation of State Chart XML (SCXML), currently a W3C Working Draft.
- [Transaction](#) - Utility classes commonly used in transactional programming
- [Validator](#) - Components for user input validation
- [VFS](#) - Extensible framework to access various filesystems through a unified api with implementations for e.g. sftp, smb, http, zip, ...

On the commons site:

- <http://commons.apache.org/components.html>

# Sandbox

[TheSandbox](#) is an open workspace for Apache committers. This is also divided into components.

On this wiki:

- [ClassScan](#) - Read access to the ClassPath of a ClassLoader and metadata about the classes available from a ClassLoader.
  - [CLI2](#) - An experimental rewrite of the Commons CLI API
  - [CommonsCsv](#) - Proposed new component
  - [Id](#) - Generators for identifiers
  - [ME](#) - Provides JaveME helpers, and implementations of JavaSE functionality that JavaME lacks but programmers want
  - [Metadata](#) - Proposed new component which provides a class metadata API similar to that of JDK5
- 

## Dormant

Sandbox components that are currently inactive. On this wiki:

- [CommonsConvert](#) - Various conversion ideas extracted from [BeanUtils](#)
- [FeedParser](#) - RSS and Atom parsing infrastructure
- [Resources](#) - A lightweight framework for defining and looking up internationalized message strings.