

Collections GenericCollections

Page Guidelines

It is early days for this project. Many ideas need to be put forward and debated. For speed and efficiency, this page will/should list all questions and design ideas and decisions. Where ideas have not reached consensus, "(needs debate)" will be appended.

Purpose

The new generics version of Commons Collections is designed to update the apis for use with Java 5 and above. As such it will use the type safety provided by generics, and will use other new features as appropriate.

This is intended to be a complete rewrite, with every method and class considered as to it's benefit and function. New classes and concepts will be added.

If you know of any frustration that developers have with collections, this is your opportunity to make a change - possibly for the next decade

Backwards Compatibility

This release will be a "completely new" codebase and will offer very little binary compatibility with previous releases. All deprecated methods and classes will be removed, and return types will be altered where it is seen fit to do so (see [COLLECTIONS-218](#)).

Type safety

As stated above, generics will be used to ensure type safety of collection apis. This renders casting methods (`MapUtils.getDouble()`) useless, and they will be deleted.

However, classes and utilities that provide runtime type safety will probably stay in (needs debate).

Project base

There was consideration whether to build off the commons non-generic code, or the collections.sf.net generic code. The decision has been made to work directly off the apache code:

- the sf code attempted to maintain binary compatibility with the non-generic commons code. This release does not intend to have binary compatibility, and may be substantially different.
- since this is a rewrite, all generic conversion decisions will be reconsidered, and starting with an almost complete code base may actually hinder this.

Development

[Project Plans](#) outlines the direction of the development

[Development Guidelines](#) explains how the project is organised, how to get started, and most importantly, how to ensure that your contributions do not pose a burden to other developers

[Contributors](#) lists the people who are currently helping out with the project, and what they are working on