Guava Release 14.0: Release Notes

- Release 14.0.1 was released on March 15, 2013 (it is identical to rc3).
- Release 14.0 was released on February 25, 2013 (it was incorrectly cut from rc2 instead of rc3).
- Release 14.0-rc3 was released on February 13, 2013.
- Release 14.0-rc2 was released on January 17, 2013.
- Release 14.0-rc1 was released on December 14, 2012.

(See ReleaseHistory.)

Full API Documentation

Using Guava in your project

This release will be identified in the Maven Central repository as com.google.guava:guava:14.0.1 and com.google.guava:guava-gwt:14.0.1

See UseGuavaInYourBuild for help integrating Guava into your build environment.

If you don't use managed dependencies, you can also just manually download JARs of the classes, sources and documentation from:

- guava-14.0.1.jar
- guava-gwt-14.0.1.jar (for GWT users)
- guava-14.0.1-javadoc.jar (Javadoc)
- guava-14.0.1-sources.jar (Source)

Issues resolved

64 issues are resolved in this release.

API Changes

Full JDiff Report of changes since release 13.0.1

To build a combined report of the API changes between release 14.0.1 and any older release, check out our docs tree and run `jdiff/jdiff.sh` with the previous release number as argument (example: `jdiff.sh 5.0`).

Significant API additions

common.collect

ForwardingBlockingDeque

RangeMap, ImmutableRangeMap, TreeRangeMap

RangeSet, ImmutableRangeSet, TreeRangeSet

common.io

BaseEncoding

ByteSink & ByteSource

CharSink & CharSource

FileWriteMode

common.reflect

ClassPath & ClassInfo

Invokable

Parameter

common.util.concurrent

AbstractListeningExecutorService

FutureFallback

ServiceManager & !Listener

Significant API changes

The static methods in Equivalences have been moved into Equivalence.

The static methods in DiscreteDomains have been moved into DiscreteDomain.

Stopwatch#elapsedMillis and #elapsedTime have been deprecated in favor of #elapsed

`Sets.cartesianProduct` now guarantees iteration in lexicographical order, instead of leaving order unspecified.

Promoted from `@Beta`

common.base

Ticker

ToStringHelper#omitNullValues

Optional#or #transform

common.collect

BoundType

Range

FluentIterable

ForwardingDeque, ForwardingNavigableMap, ForwardingNavigableSet

SortedMapDifference

Ordering.{lea,greate}stOf(Itera{ble,tor})

Ordering.max/min(Iterator)

Immutable{Set,List,}Multimap.inverse()

ImmutableMultimap.Builder.order{Value,Key}sBy

Sets.synchronizedNavigableSet

Maps.filter{Values,Keys,Entries}

Maps.synchronizedNavigableMap

Maps.transform{Values,Keys,Entries}

common.math

IntMath & LongMath

common.net

HttpHeaders

common.primitives

Unsigned{Long,Int}s

common.util.concurrent

FutureCallback

Futures (many methods)

AsyncFunction

Atomics

AtomicDouble

AtomicDoubleArray & AtomicLongMap

ExecutionError & UncheckedExecutionException

Non-API changes

RateLimiter getRate/setRate are no longer synchronized on the current instance

`Sets.cartesianProduct`

Optimizations

`LongMath.binomial` is significantly faster.

`LongMath.sqrt` is significantly faster.

`ImmutableBiMap` takes ~35% less memory.

`HashBiMap` takes ~40% less memory.

`Ordering.{lea,greate}stOf(elements, k)` still run in /O(n)/ time, but they perform only one pass and use only /O(k)/ extra memory. (Accordingly, they now have `Iterator` overloads.)

`ImmutableSet` now uses an improved murmur-based rehashing function to improve dispersion.