Apache Apache Commons Collections 4.4 RELEASE NOTES

The Apache Commons Collections package contains types that extend and augment the Java Collections Framework.

Maintenance release.

Changes in this version include:

New features:

o COLLECTIONS-715: Implement Collection's removeIf(). Thanks to morningmemo, Gary Gregory.

o COLLECTIONS-719: Create a PropertiesFactory and SortedPropertiesFactory. Thanks to Gary Gregory.

o COLLECTIONS-719: Support Transformer for LazyList #52. Thanks to Stephan Windmüller, Bruno P. Kinoshita.

o COLLECTIONS-723: Make use of FunctionalInterface #48. Thanks to Eitan Adler, SOC, Bruno P. Kinoshita.

Fixed Bugs:

o COLLECTIONS-710: NullPointerExceptions in CompositeCollection, CompositeSet, and CompositeMap. Thanks to Yu Shi, Gary Gregory.

Changes:

o COLLECTIONS-718: Fix LRUMap exception message. Thanks to Eitan Adler.

o COLLECTIONS-716: Don't include email address in Exception messages. Thanks to Sebb.

For complete information on Apache Commons Collections, including instructions on how to submit bug reports,

patches, or suggestions for improvement, see the Apache Apache Commons Collections website:

https://commons.apache.org/proper/commons-collections/

Download page: https://commons.apache.org/proper/commons-collections/download\_pool.cgi

===============================================================================

Apache Apache Commons Collections 4.3 RELEASE NOTES

The Apache Commons Collections package contains types that extend and augment the Java Collections Framework.

Update from Java 7 to Java 8, bug fixes, and small changes.

Changes in this version include:

New features:

o COLLECTIONS-696: AbstractReferenceMap made easier for subclassing; PR #51. Thanks to Maxim Solodovnik.

Fixed Bugs:

o COLLECTIONS-691: Use boolean operator for boolean result. Thanks to Eitan Adler.

o COLLECTIONS-701: StackOverflowError in SetUniqueList.add() when it receives itself. Thanks to Shin Hong, Don Jeba.

o COLLECTIONS-703: The PassiveExpiringMap#put() method should return the previous record only if not expired. Thanks to Tomas Tulka.

o COLLECTIONS-706: Add SetUtils.unmodifiableSet(T... items) method. Thanks to Richard Eckart de Castilho, Gary Gregory.

Changes:

o COLLECTIONS-688: Update platform requirement from Java 7 to 8.

o COLLECTIONS-689: Link to Javadoc API broken. Thanks to Richard Walker.

o COLLECTIONS-692: Replace use of deprecated Class#newInstance() PR #49. Thanks to Gary Gregory, Eitan Adler.

Please NOTE: Clirr errors reported for this release are false positive, this is prooved by Japicmp report

And manual testing.

For complete information on Apache Commons Collections, including instructions on how to submit bug reports,

patches, or suggestions for improvement, see the Apache Apache Commons Collections website:

https://commons.apache.org/proper/commons-collections/

Download page: https://commons.apache.org/proper/commons-collections/download\_collections.cgi

===============================================================================

Apache Apache Commons Collections 4.2 RELEASE NOTES

The Apache Commons Collections package contains types that extend and augment the Java Collections Framework.

Update from Java 6 to Java 7, bug fixes, and small changes.

Changes in this version include:

New features:

o COLLECTIONS-681: Add test for MultiSetUtils Thanks to Stephan Fuhrmann.

o COLLECTIONS-658: Add Automatic-Module-Name MANIFEST entry for Java 9 compatibility.

o COLLECTIONS-589: Add null-safe MapUtils.size(Map<?, ?>) method. Thanks to Gary Gregory.

o COLLECTIONS-586: PatriciaTrie prefixMap clear throws NullPointerException. Thanks to Shailender Bathula, Gary Gregory.

o COLLECTIONS-654: Add class SortedProperties to sort keys.

o COLLECTIONS-668: Add CollectionUtils containsAny method for primitive array: org.apache.commons.collections4.CollectionUtils.containsAny(Collection<?>, T...). Thanks to Gary Gregory.

o COLLECTIONS-575: Synchronized queue wrapper in QueueUtils. Thanks to Guram Savinov, Grzegorz Rożniecki, Bruno P. Kinoshita, Gary Gregory.

o COLLECTIONS-670: Add org.apache.commons.collections4.IteratorUtils.first(Iterator). Thanks to Gary Gregory.

o COLLECTIONS-671: Add org.apache.commons.collections4.IterableUtils.first(Iterable). Thanks to Gary Gregory.

Fixed Bugs:

o COLLECTIONS-599: HashEntry array object naming data initialized with double the size during deserialization. Thanks to Tejas Patel, Saleem Akbar, Gary Gregory.

o COLLECTIONS-662: Unit tests MapUtilsTest and ListIteratorWrapperTest no longer fail on Java 9. Thanks to Vamsi Kavuri.

o COLLECTIONS-661: Intermittent test failures in Windows for HashSetValuedHashMap.

o COLLECTIONS-660: Uncomment test in AbstractMapTest regarding LRUMap equals.

o COLLECTIONS-656: Fix site build on Java 8.

o COLLECTIONS-653: Update Javadoc to Build on Java 1.8.

o COLLECTIONS-606: Build status, Coverage status and Maven central weren't in README.md Thanks to Vamsi Kavuri.

o COLLECTIONS-594: Web site spelling error: MultiValuedMapeList. Thanks to Javen O'Neal.

o COLLECTIONS-597: Correction of Javadoc for org.apache.commons.collections4.functors.CatchAndRethrowClosure. Thanks to Enrique.

o COLLECTIONS-678: The verification of unsupported iterator methods is not complete. Thanks to Oscar Luis Vera Pérez.

o COLLECTIONS-673: ListUtils.partition potential integer overflow. Thanks to John Mark, Stephan Fuhrmann.

Changes:

o COLLECTIONS-602: Improve efficiency of DefaultedMap.get. Thanks to John Mark.

o COLLECTIONS-603: Small improvements for generics, conditional statements, and warnings suppressions. Thanks to Artem Konovalov.

o COLLECTIONS-655: Update platform from Java 6 to Java 7.

o COLLECTIONS-666: org.apache.commons.collections4.ListUtils.union(List, List) should pre-allocate result list. Thanks to BELUGA BEHR.

o COLLECTIONS-669: Update org.apache.commons.collections4.CollectionUtils.addAll(Collection<C>, C[]) to addAll(Collection<C>, C...). Thanks to BELUGA BEHR, Gary Gregory.

For complete information on Apache Commons Collections, including instructions on how to submit bug reports,

patches, or suggestions for improvement, see the Apache Apache Commons Collections website:

https://commons.apache.org/proper/commons-collections/

Download page: https://commons.apache.org/proper/commons-collections/download\_collections.cgi

===============================================================================

Apache Commons Collections

Version 4.1

RELEASE NOTES

INTRODUCTION:

Apache Commons Collections is a project to develop and maintain collection

classes based on and inspired by the JDK collection framework.

This v4.1 release is a minor and security release, fixing several bugs present in

previous releases of the 4.X branch. Additionally, this release provides a mitigation

for a known remote code exploitation via the standard java object serialization

mechanism. Serialization support for unsafe classes in the functor package has been

completely removed (classes do not implement the Serializable interface anymore).

For more details, please refer to COLLECTIONS-580.

All users are strongly encouraged to updated to this release.

Compatibility

-------------

o Java 6 or later is now required (previous version 4.0 required Java 5).

o This release is source and binary compatible with the previous 4.0 release

of Commons Collections with the exception that some classes in the functor

package do not implement the Serializable interface anymore (see COLLECTIONS-580).

o some unit tests are disabled when executed with an IBM JDK 6 due to a faulty

java.util.TreeMap implementation.

Major changes since 4.0

-----------------------

o Added interface MultiSet (package multiset)

o Added interfaces MultiValuedMap, ListValuedMap and SetValuedMap (package multimap)

o Added fluent API for Iterable instances (class FluentIterable)

Changes

-------

o [COLLECTIONS-580]: Serialization support for unsafe classes in the functor package

has been removed as this can be exploited for remote code execution

attacks. Classes considered to be unsafe are: CloneTransformer,

ForClosure, InstantiateFactory, InstantiateTransformer, InvokerTransformer,

PrototypeCloneFactory, PrototypeSerializationFactory, WhileClosure.

o [COLLECTIONS-551]: Deprecated various method in "CollectionUtils" in favor of similar

methods in the newly introduced "IterableUtils".

o [COLLECTIONS-570]: All constructors and static factory methods will now throw a "NullPointerException"

if a required input argument is null. Previously sometimes a "IllegalArgumentException"

was used.

o [COLLECTIONS-571]: Deprecated methods "synchronizedCollection(Collection)" and

"unmodifiableCollection(Collection)" in class "CollectionUtils", the corresponding

methods in "java.util.Collections" should be used instead.

o [COLLECTIONS-562]: Upgraded minimum java requirement to Java 6 (up from Java 5).

New features

------------

o [COLLECTIONS-508]: Added new interfaces "MultiValuedMap", "ListValuedMap" and "SetValuedMap"

as a replacement for "MultiMap". Decorators and implementations reside in

the "multimap" package and a "MultiMapUtils" class has been added.

The existing interface "MultiMap" as well as the concrete implementation

"MultiValueMap" has been deprecated.

o [COLLECTIONS-567]: Added new MultiSet interface which is intended to be a replacement for

the Bag interface. The main difference is that a MultiSet is fully compatible

to the Collection contract.

o [COLLECTIONS-572]: Added set operations to "SetUtils": union, difference, intersection and disjunction.

The operations return a view of the result that is backed by the input sets.

o [COLLECTIONS-557]: Added support to specify the initial size of a "LRUMap". Thanks to Philippe Mouawad.

o [COLLECTIONS-565]: Added decorators for "NavigableSet" interface.

o [COLLECTIONS-464]

+COLLECTIONS-442]: Added new class "FluentIterable" to support a fluent API for manipulating

Iterable instances. Additionally various supporting methods have been

added to "IterableUtils" and "IteratorUtils".

o [COLLECTIONS-464]: Added new "ZippingIterator" and factory methods "IteratorUtils#zippingIterator(...)".

o [COLLECTIONS-464]: Added new decorator "SkippingIterator" and factory methods "IteratorUtils#skippingIterator(...)".

o [COLLECTIONS-556]: Added method "SetUtils#newIdentityHashSet()" which returns a new identity HashSet

using reference-equality instead of object-equality.

o [COLLECTIONS-395]: Added method "LRUMap#get(Object, boolean)" that allows to query the map

without affecting the least recently used order. Thanks to David Hawthorne.

o [COLLECTIONS-427]: Added "toString(...)" methods to newly created "IterableUtils" and existing "IteratorUtils"

to get a string representation of an Iterable/Iterator instance similar to "Arrays#toString(...)".

Thanks to Gonçalo Marques.

o [COLLECTIONS-539]: Changed scope of "CircularFifoQueue#isAtFullCapacity()" to public. Thanks to Guram Savinov.

o [COLLECTIONS-511]: Added new methods "IterableUtils#partition(...)" to partition an input collection

into separate output collections based on evaluation of one or more predicates.

Thanks to Nathan Blomquist, Brent Worden.

o [COLLECTIONS-529]: Added methods "removeAll(...)" and "retainAll(...)" to "CollectionUtils" that perform

equality checks using the provided "Equator" object instead of "Object#equals()".

Thanks to Alexander Muthmann, Dipanjan Laha.

o [COLLECTIONS-503]: Added new transformer "IfTransformer" and factory methods "TransformerUtils#ifTransformer(...)"

which replace "TransformerUtils#switchTransformer(Predicate, Transformer, Transformer)".

Thanks to Josh Cain.

o [COLLECTIONS-471]: Added new decorator "BoundedIterator" and factory methods "IteratorUtils#boundedIterator(...)".

Thanks to Radford Tam.

Fixed Bugs

----------

o [COLLECTIONS-580]: Added validation when de-serializing a "MultiValueMap#ReflectionFactory":

only Collection classes are allowed, otherwise an UnsupportedOperationException

will be thrown during de-serialization.

o [COLLECTIONS-576]: Subclasses of MultiKey did not re-calculate their hashcode after de-serialization.

Thanks to Stephan Roch.

o [COLLECTIONS-566]: "IteratorUtils#collate(...)" methods did not use natural ordering when a

null comparator was provided.

o [COLLECTIONS-558]: Changed return type of "ListOrderedSet#remove(int)" from Object to the generic

type parameter. Thanks to Felix Rabe.

o [COLLECTIONS-555]: Added clarification to javadoc of "TreeBag#add(Object)" wrt null arguments.

Thanks to M Kim.

o [COLLECTIONS-427]: Reverted performance improvement for "SetUniqueList#retainAll(Collection)"

introduced in 4.0. Added clarifying javadoc wrt runtime complexity instead.

o [COLLECTIONS-426]: Reverted performance improvement for "ListOrderedSet#retainAll(Collection)"

introduced in 4.0. Added clarifying javadoc wrt runtime complexity instead.

o [COLLECTIONS-530]: Added a Builder for "PredicatedCollection". Elements added to the builder

that fail the predicate will not throw an IllegalArgumentException. The builder

supports creating predicated lists, bags, sets and queues. Thanks to Erik.

o [COLLECTIONS-545]: Documented runtime complexity of "CollectionUtils#removeAll(Collection, Collection).

Thanks to Oswaldo Olivo.

o [COLLECTIONS-543]: "AbstractCollectionDecorator" doesn't forward equals and hashCode anymore.

o [COLLECTIONS-544]: Documented runtime complexity of "CollectionUtils#retainAll(Collection, Collection).

Thanks to Oswaldo Olivo.

o [COLLECTIONS-542]: "AbstractHashedMap" still inherits from "AbstractMap", contrary to what

the class javadoc stated. The inheritance will now be removed in v5.0.

o [COLLECTIONS-525]: The map returned by "PatriciaTrie#prefixMap()" did not contain all keys

that are prefixed by the given search key in some rare cases. Thanks to Zigler Zhang.

o [COLLECTIONS-537]: Harmonized signature of factory methods for functor-related classes which take

a collection as input with their array counterparts. Thanks to Frank Jakop.

o [COLLECTIONS-540]: Added overloaded method "CollectionUtils#get(Enumeration, int)" and simplified

code for "CollectionUtils#get(Object, int)". Thanks to Daniel Stewart, Issam El Atif.

o [COLLECTIONS-536]: Improved check for null input in "MapUtils#putAll(Map, Object[])". Thanks to Tagir Valeev.

o [COLLECTIONS-534]: Added clarifying javadoc wrt runtime complexity of "CollectionBag#retainAll".

Thanks to Oswaldo Olivo.

o [COLLECTIONS-531]: Use correct type bounds in

"CollectionUtils#isEqualCollection(Collection, Collection, Equator)" to

prevent a "ClassCastException" at runtime for invalid inputs. Thanks to Dipanjan Laha.

o [COLLECTIONS-523]: Removed unneeded private method in "PassiveExpiringMap". Thanks to Thiago Andrade.

o [COLLECTIONS-516]: Added clarification to the javadoc of "MapUtils#toProperties(Map)" in case

of null keys/values. Thanks to Cyrille Artho.

o [COLLECTIONS-524]: "ListOrderedSet#listOrderedSet(List)" did not remove duplicates from the

input list as advertised in the javadoc. Thanks to J Goodfellow.

o [COLLECTIONS-521]: "MultiKeyMap" was throwing a "NullPointerException" for various operations

if two key arguments have been used and the second was "null". Thanks to Maxime Nay.

o [COLLECTIONS-522]: Updated code example for "PredicatedList". Thanks to Erik.

o [COLLECTIONS-512]: "TransformingComparator" and "FixedOrderComparator" did not comply with

the contract of "Object#equals". Thanks to Cyrille Artho.

o [COLLECTIONS-510]: Fix compilation errors when using source level 1.8 and a recent java 8 compiler.

Thanks to Hollis Waite.

o [COLLECTIONS-509]: Clarified javadoc of "CollectionBag" wrt changes from the original Bag interface.

o [COLLECTIONS-507]: Removed wrong type bounds for "ComparatorUtils#chainedComparator(...)". Thanks to Gerson.

o [COLLECTIONS-506]: Added javadoc clarification to class "CollectionUtils" that input objects which

override "Object#equals(Object)" must also maintain the general contract of

"Object#hashCode()" as various utility methods take advantage of sets/maps/bags.

Thanks to Anthony Communier.

For complete information on Apache Commons Collections, including instructions on how to submit bug reports,

patches, or suggestions for improvement, see the Apache Apache Commons Collections website:

https://commons.apache.org/proper/commons-collections/

===============================================================================

Apache Commons Collections

Version 3.2.2

RELEASE NOTES

INTRODUCTION:

Commons collections is a project to develop and maintain collection classes

based on and inspired by the JDK collection framework.

This release is JDK1.3 compatible, and does not use JDK1.5 generics.

This v3.2.2 release is a bugfix release, fixing several bugs present in the previous

releases of the 3.2 branch. Additionally, this release provides a mitigation for a

known remote code exploitation via the standard java object serialization mechanism.

By default, serialization support for unsafe classes in the functor package is

disabled and will result in an exception when either trying to serialize or de-serialize

an instance of these classes. For more details, please refer to COLLECTIONS-580.

All users are strongly encouraged to updated to this release.

Changes in this version include:

CHANGES

=======

o COLLECTIONS-580: Serialization support for unsafe classes in the functor package is

disabled by default as this can be exploited for remote code execution

attacks. To re-enable the feature the system property

"org.apache.commons.collections.enableUnsafeSerialization" needs to be

set to "true".

Classes considered to be unsafe are: CloneTransformer, ForClosure,

InstantiateFactory, InstantiateTransformer, InvokerTransformer,

PrototypeCloneFactory, PrototypeSerializationFactory, WhileClosure.

BUGFIXES

========

o COLLECTIONS-538: "ExtendedProperties" will now use a privileged action to access the

"file.separator" system property. In case the class does not have

permission to read system properties, the "File#separator" field will

be used instead. Thanks to Trejkaz.

o COLLECTIONS-447: Tree traversal with a TreeListIterator will not be affected anymore by

the removal of an element directly after a call to previous(). Thanks to Jeffrey Barnes.

o COLLECTIONS-444: SetUniqueList.set(int, Object) now works correctly if the object to be inserted

is already placed at the given position. Thanks to Thomas Vahrst, John Vasileff.

o COLLECTIONS-350: Removed debug output in "MapUtils#getNumber(Map)". Thanks to Michael Akerman.

o COLLECTIONS-335: Fixed cache assignment for "TreeBidiMap#entrySet". Thanks to sebb.

o COLLECTIONS-334: Synchronized access to lock in "StaticBucketMap#size()". Thanks to sebb.

o COLLECTIONS-307: "SetUniqueList#subList()#contains(Object)" will now correctly check the subList

rather than the parent list. Thanks to Christian Semrau.

o COLLECTIONS-304: "SetUniqueList#set(int, Object)" will now correctly enforce the uniqueness constraint.

Thanks to Rafa? Figas,Bjorn Townsend.

o COLLECTIONS-294: "CaseInsensitiveMap" will now convert input strings to lower-case in a

locale-independent manner. Thanks to Benjamin Bentmann.

o COLLECTIONS-266: "MultiKey" will now be correctly serialized/de-serialized. Thanks to Joerg Schaible.

o COLLECTIONS-261: "Flat3Map#remove(Object)" will now return the correct value mapped to the removed key

if the size of the map is less or equal 3. Thanks to ori.

o COLLECTIONS-249: "SetUniqueList.addAll(int, Collection)" now correctly add the collection at the

provided index. Thanks to Joe Kelly.

o COLLECTIONS-228: "MultiValueMap#put(Object, Object)" and "MultiValueMap#putAll(Object, Collection)"

now correctly return if the map has changed by this operation.

o COLLECTIONS-219: "CollectionUtils#removeAll" wrongly called "ListUtils#retainAll". Thanks to Tom Leccese.

o COLLECTIONS-217: Calling "setValue(Object)" on any Entry returned by a "Flat3Map" will now

correctly set the value for the current entry. Thanks to Matt Bishop.

For complete information on Apache Commons Collections, including instructions on how to submit bug reports,

patches, or suggestions for improvement, see the Apache Commons Collections website:

https://commons.apache.org/collections/