

Apache CollectionUtils transformedCollection caveat

by Benny Bottema - Tuesday, November 23, 2010

<http://www.bennybottema.com/2010/11/23/apache-collectionutils-transformedcollection-caveat/>

I love the (now aging) Apache collections15 library, but here's a small ugly detail that is good to know.

The following call doesn't do what you would think it does – or what I thought it would do anyway (and the JavaDoc isn't clear enough):

```
Collection<Cube> ballsAsCubes = CollectionUtils.transformedCollection(  
myBalls, new BallToCubeTransformer());
```

This does **not** transform your collection of balls into a collection of cubes. It does transform balls you add afterwards but not what is already in there. So what happens is that when you pass in a populated list with balls, the `Collection<Cube>` will contain objects of type `Ball` in runtime! That's right, the `CollectionUtils` library suppresses any generics warnings so that you will get a nice runtime exception when you use it to, say, sort the list:

```
java.lang.ClassCastException: org.codemonkey.Ball  
  at org.codemonkey.CubeComparator.compare(CubeComparator.java:1)  
  at java.util.Collections$ReverseComparator2.compare(Collections.java:  
3361)  
  at java.util.Arrays.mergeSort(Arrays.java:1284)  
  at java.util.Arrays.sort(Arrays.java:1223)  
  at java.util.Collections.sort(Collections.java:159)
```

In fact, the `TransformedCollection` that comes out `CollectionUtils.transformedCollection()` is rather the odd one out, as you actually need to cast the list to the raw type (or `TransformedCollection`) in order to add a cube which is then transformed to a ball. either way the compiler will warn you about your foul language. From the `TransformedCollection` documentation:

above package comment:

```
// GenericsNote: Converted, but unfortunately very little type-  
safety could be achieved without breaking Collection interface.
```

in class comment:

```
/**  
 * Note: This class cannot support generics without breaking the Collection contract.  
 */
```

Fun isn't it.

What you should do instead is the following:

```
Collection<Cube> ballsAsCubes = CollectionUtils.collect(myBalls, new BallToCubeTransformer());
```

Almost the same... almost, but not quite.

PDF generated by Kalin's PDF Creation Station