



Moshe Rubin <moshe.rubin@gmail.com>

I have a fix for data selection not working on a zoomed chart

1 message

Moshe Rubin <moshe.rubin@gmail.com>

Thu, Jun 30, 2016 at 8:10 PM

To: David Gilbert <david.gilbert@object-refinery.com>

Bcc: Moshe Rubin <moshe.rubin@gmail.com>, Yosh Mantinband <yoshm@renalsense.com>

Hi Dave,

You may remember a recent correspondence we had regarding getting JFreeChart-FSE to build without test failures. I'm happy to report that I have now modified my client's application to use JFreeChart-FSE in order to support data selection ("lassoing"), and am very excited with what data selection provides the app.

I wanted to report a problem I encountered with JFreeChart-FSE and the fix I would like to propose.

The Problem

Data selection works fine when the chart is unzoomed, with my app's selectionChanged() callback being invoked following data point selection. When the chart is zoomed, however, the selectionChanged() callback is not invoked.

Playing with zooming shows that data selection always fails following zooming using the mouse (i.e., mouse button down + drag). When zooming using the mouse wheel, data selection works for the first few zoom-in mouse wheel clicks. Once the chart has been enlarged a few times, selection fails.

Reproducing the Problem

I can easily reproduce the problem with a minimally modified SelectionDemo1.java file, will I am attaching to this email. All code source changes are prefaced with a comment of the form "/*MR-0n*/" (n is 1,2,3,...).

The Technical Explanation

Examining the FSE code shows that the reason for SelectionChanged() not being called following zooming is because the hash value of the DatasetSelectionExtension<XYCursor> object registered in function createDemoPanel() is modified because of zooming. This causes the statement "registeredExtensions.get(dataset)" in DatasetExtensionManager.getExtension() to return null:

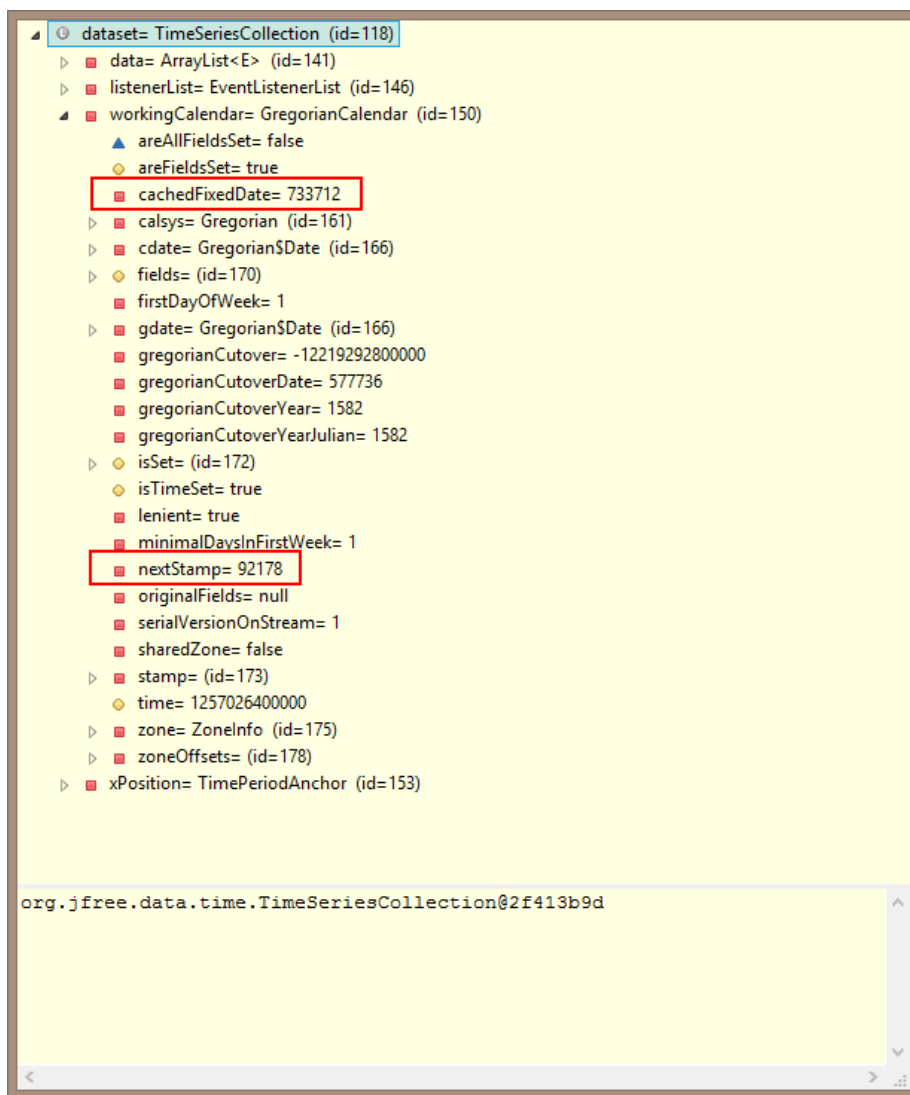
```
public <T extends DatasetExtension> T getExtension(Dataset dataset,
    Class<T> interfaceClass) {
    if (interfaceClass.isAssignableFrom(dataset.getClass())) {
        //the dataset supports the interface
        return interfaceClass.cast(dataset);
    } else {
        List<DatasetExtension> extensionList
            = registeredExtensions.get(dataset);
        if (extensionList != null) {
            for (DatasetExtension extension : extensionList) {
                if (interfaceClass.isAssignableFrom(extension.getClass())) {
                    //the dataset does not support the extension but
                    //a matching helper object is registered for the dataset
                    return interfaceClass.cast(extension);
                }
            }
        }
    }
    return null;
}
```

Drilling down further shows that the Calendar member variable *workingCalendar* has some of its member variables changed due to zooming. Here is *workingCalendar* before zooming:

```
dataset= TimeSeriesCollection (id=118)
├─ data= ArrayList<E> (id=141)
├─ listenerList= EventListenerList (id=146)
└─ workingCalendar= GregorianCalendar (id=150)
    ├─ areAllFieldsSet= false
    ├─ areFieldsSet= true
    │   └─ cachedFixedDate= 733924
    ├─ calsys= Gregorian (id=161)
    ├─ cdate= GregorianCalendar (id=166)
    ├─ fields= (id=170)
    │   └─ firstDayOfWeek= 1
    ├─ gdate= GregorianCalendar (id=166)
    ├─ gregorianCutover= -12219292800000
    ├─ gregorianCutoverDate= 577736
    ├─ gregorianCutoverYear= 1582
    ├─ gregorianCutoverYearJulian= 1582
    ├─ isSet= (id=172)
    │   └─ isTimeSet= true
    │       └─ lenient= true
    │           └─ minimalDaysInFirstWeek= 1
    │               └─ nextStamp= 90344
    │                   └─ originalFields= null
    │                       └─ serialVersionOnStream= 1
    │                           └─ sharedZone= false
    │                               └─ stamp= (id=173)
    │                                   └─ time= 1275339600000
    │                                       └─ zone= ZoneInfo (id=175)
    │                                           └─ zoneOffsets= (id=178)
    └─ xPosition= TimePeriodAnchor (id=153)
        ├─ name (TimePeriodAnchor)= "TimePeriodAnchor.START" (id=179)
        └─ name (Enum)= "START" (id=182)
            └─ ordinal= 0
```

org.jfree.data.time.TimeSeriesCollection@3a7d85c1

Here it is after zooming:



The hash returned by the *Calendar* class takes these into account, causing the *HashMap* `get()` to fail and return null, preventing `selectionChanged()` from being invoked.

The Proposed Fix

I believe *workCalendar* is unsafe to include in the hash of a *TimeseriesCollection* object should be removed. I would like to propose removing it from *TimeSeriesCollection.hashCode()*:

```
/**
 * Returns a hash code value for the object.
 *
 * @return The hashcode
 */
@Override
public int hashCode() {
    int result;
    result = this.data.hashCode();
    // Moshe Rubin (30 June 2016)
    //
    // Several fields in <workingCalendar> change following zooming of a chart,
    // preventing the XYDatasetSelectionExtension object from being found in
    // the HashMap. We therefore remove <workingCalendar> from being a part
    // of the hash calculation.
    //result = 29 * result + (this.workingCalendar != null
    //    ? this.workingCalendar.hashCode() : 0);
    result = 29 * result + (this.xPosition != null
        ? this.xPosition.hashCode() : 0);
}
```

```
        return result;  
    }
```

When I made the fix locally and rebuilt FSE, data selection works perfectly on both zoomed and non-zoomed charts.

Can you confirm my analysis and proposed fix?

Best regards,

Moshe

 **SelectionDemo1.java**
10K