# Base Language - Bug #2499

## DST portion of the timezone offset is always included when it should be conditional

01/23/2015 01:46 PM - Greg Shah

| | | | |
|---|---|---|---|
| **Status:** | WIP | **Start date:** | 01/23/2015 |
| **Priority:** | Normal | **Due date:** | |
| **Assignee:** | | **% Done:** | 0% |
| **Category:** | | **Estimated time:** | 0.00 hour |
| **Target version:** | | | |
| **billable:** | No | **case_num:** | |
| **vendor_id:** | GCD | **version:** | |

**Description**

**Related issues:**

| | |
|---|---|
| Related to Base Language - Bug #2511: timestamp instance oddity when assignin... | **New** |

### History

**#1 - 01/23/2015 01:51 PM - Greg Shah**

TIMCO reported a problem with transactions starting at midnight. Constantin tracked the problem back to the TODAY function which was yielding next day values when called during the last hour of the day. Since there was a 1 hour offset and we are currently NOT in daylight savings time, the suspicion naturally turned towards timezone processing. Here is what he found:

More info related to this. The root cause is in date.getDefaultOffset() line 3138.

```
    public static int getDefaultOffset()
    {
       integer tzAttr = work.obtain().timezoneOffset;
       int     offset = 0;

       if (tzAttr.isUnknown())
       {
          // get the raw offset in millis and convert it to minutes
          TimeZone defTz = TimeZone.getDefault();
          // this is probably not the right calculation, but it seems that
          // in Progress the timezone also contains the DST
          offset = (defTz.getRawOffset() + defTz.getDSTSavings()) / (1000 * 60);
       }
       else
       {
          offset = tzAttr.intValue();
       }
  }
```

TimeZone.getDSTSavings() returns 3600 regardless if current date is within DST bracket or not, as long as the timezone uses DST (some China/Russia TZs don't use timezones); the correct API which tells if a date is within DST is TimeZone.inDaylightTime(Date), but I'm not sure how to use this for our purposes. I've discussed a little with Ovidiu and he says that, as getDefaultOffset() is used (only in calls for today or now function), we can use TimeZone.inDaylightTime(new Date()) to determine if we need the DST offset.

In any case, a final fix should include extensive tests of what happens in the edge cases, within and outside the interval when DST changes.

**#2 - 01/23/2015 01:57 PM - Greg Shah**

I took a quick look at this.  Outside the util package, date.getDefaultOffset() is used from persist/HQLPreprocessor and persist/pl/Functions.  We also have to look carefully at all the date, datetime and datetimetz classes since these may indirectly depend on date.getDefaultOffset() depending on the call path.

Once all usage points have been found, we need to write 4GL tests to confirm the behavior.  Based on that we can formulate a fix.

**#3 - 01/26/2015 02:42 PM - Constantin Asofiei**

Some ideas about automating the DST testing (in junits or date.main):

> Ovidiu Maxiniuc wrote:

Java 8 has some new APIs, http://docs.oracle.com/javase/8/docs/api/java/time/package-summary.html , among which:

```
    static Clock offset(Clock baseClock, Duration offsetDuration)
        Obtains a clock that returns instants from the specified clock with the specified duration added
```

But this will not work until we have Java8 :(.

> Constantin Asofiei wrote:

This link has some ideas about how to override the system clock:

> http://stackoverflow.com/questions/2001671/override-java-system-currenttimemillis-for-testing-time-sensitive-code

From these, jmock looks interesting.

**#4 - 01/28/2015 04:37 PM - Ovidiu Maxiniuc**

*- Status changed from New to WIP*

Most likely, the code inside getDefaultOffset should be fixed:

```
         offset = defTz.getRawOffset();
         if (defTz.inDaylightTime(new Date(System.currentTimeMillis())))
         {
            offset += defTz.getDSTSavings();
         }
         offset = offset / (1000 * 60);
```

The date.getDefaultOffset() calls from date, persist/HQLPreprocessor and persist/pl/Functions are meant for current location / timezone of local time.
So it should be safe to check the id the DST is active with the current instant.
Next, I am going to investigate the datetime and datetimetz from this point of view. They are more related to DST/timezone notion.

**#5 - 01/28/2015 04:52 PM - Greg Shah**

I'm good with this approach if you find the the datetime and datetimetz are OK too.

**#6 - 01/29/2015 02:44 PM - Ovidiu Maxiniuc**

*- File om_upd20150129b.zip added*

This is most likely an RC.
I only done the 'theoretical' checking on datetime and datetimetz. I am preparing now a testing setup.

**#7 - 01/29/2015 05:51 PM - Greg Shah**

Code Review om_upd20150129b.zip

There are no changes in the datetime.java that was in the update zip. Otherwise, I'm OK with the changes.

**#8 - 02/02/2015 05:34 PM - Ovidiu Maxiniuc**

*- File om_upd20150203a.zip added*

I am putting this update to test.
It passed my tests regarding the DST. Yet, there is an issue regarding the assignment of a datetime-tz to datetime variable that I discovered and wanted fixed but since this is beyond the subject of this issue, I will defer to another task, allowing to commit this update.

**#9 - 02/03/2015 07:44 AM - Greg Shah**

Code Review om_upd20150203a.zip

I'm fine with the changes. If it passes testing, you can check it in.

Please add a new task for the remaining known datetime issue. Add it into the Base Language project and include a testcase that shows the problem. Assign it to milestone 12.

**#10 - 02/03/2015 11:21 AM - Ovidiu Maxiniuc**

The om_upd20150203a.zip update has:

- 3 fails in CTRL+C
- 2 fails in gc_tests
- 6 fails in tc_tests

Except for a couple of cases, the other are 'old acquaintances' that usually disappear at the next run.
I am re-running the runtime regression testing only.

**#11 - 02/04/2015 01:13 PM - Ovidiu Maxiniuc**

Greg Shah wrote:

> I'm fine with the changes.  If it passes testing, you can check it in.

The main passed the regression testing.
The CTRL+C still has the same 3 fails. In fact there is only one issue here, the client is killed in test_5_ctrlc_11_session3.html at step 9, most likely after entering the credentials for authentication. As that instance is dead, the other two cannot receive the POST event they require for advancing. Since this is a very common issue in later test runs and not related to current issue, I deem the update passed the runtime test and I am going to commit and distribute.

> Please add a new task for the remaining known datetime issue.  Add it into the Base Language project and include a testcase that shows the problem.  Assign it to milestone 12.

I created the new issue (#2511) and linked it to this one.

**#12 - 02/04/2015 01:31 PM - Greg Shah**

Great!

Next steps:

1. Check in and distribute om_upd20150203a.zip.

2. Prepare a version of the fix that has the date.java changes merged with the H063 version of date.java.  This can't be provided back to TIMCO on the H064 version without pulling in more files and unneeded risk.  That is the version we need to send to them.

**#13 - 02/04/2015 02:31 PM - Ovidiu Maxiniuc**

> 1. Check in and distribute om_upd20150203a.zip.

The update was committed to bzr renvo 10742 and distributed by mail.

**#14 - 02/04/2015 02:44 PM - Ovidiu Maxiniuc**

*- File om_upd20150204a.zip added*

Greg Shah wrote:

> 2. Prepare a version of the fix that has the date.java changes merged with the H063 version of date.java. This can't be provided back to TIMCO on the H064 version without pulling in more files and unneeded risk. That is the version we need to send to them.

Is this the file we need for Timco ?

**#15 - 02/04/2015 02:45 PM - Greg Shah**

Yes, exactly.

**#16 - 02/05/2015 06:00 PM - Ovidiu Maxiniuc**

I believe I have good news.
The testing of 0204a finished with following result:
gso_ctrlc_tests: 2 fails: once the client got killed and the other was waiting for posting event.
tc_tests: 2 fails: tc_job_002 and tc_item_master_104, step 45 because of a line switch 55580 with 55582.
So I think it passed the test.

**#17 - 02/05/2015 06:02 PM - Greg Shah**

I agree. Well done.

## Files

| | | | |
|---|---|---|---|
| om_upd20150129b.zip | 65.8 KB | 01/29/2015 | Ovidiu Maxiniuc |
| om_upd20150203a.zip | 68.1 KB | 02/02/2015 | Ovidiu Maxiniuc |
| om_upd20150204a.zip | 37.4 KB | 02/04/2015 | Ovidiu Maxiniuc |