Base Language - Feature #6936

OpenClient with null values at the DataObject (table) fields for native types

11/16/2022 04:59 AM - Constantin Asofiei

Status:	WIP		**Start date:**	
Priority:	High		**Due date:**	
Assignee:	Constantin Asofiei		**% Done:**	90%
Category:			**Estimated time:**	0.00 hour
Target version:				
billable:	No		**version:**	
vendor_id:	GCD			

Description

History

#1 - 11/16/2022 05:01 AM - Constantin Asofiei

- Subject changed from OE 1.6 vs OE 1.7 OpenClient differences with null values at the DataObject (table) fields to OE 11.6 vs OE 11.7 OpenClient differences with null values at the DataObject (table) fields

In OE, there seems to be a difference how null values are reported in a ProDataObject.

For example, doing something like this:

```
ProDataObject row = ... create row with a boolean field
System.out.println(row.get("some-boolean-field"));
```

in OE 11.6.3 will show 'false' instead of null.

Marian, can you please double-check in OE 11.7 or 12 how this behaves?

This behavior where a default value is returned instead of null can be seen for integer and int64 fields, oto.

#2 - 11/16/2022 05:38 AM - Constantin Asofiei

- Status changed from New to WIP

- Assignee set to Constantin Asofiei

- Priority changed from Normal to High

Marian, do you know in what conditions OE's JavaOpenClient would return null instead of a default value for integer,int64 and logical data-types?

I ask because I'm testing with the customer's OE jars, and I still get the same 'false' instead of 'null' for an unset boolean/logical field. And the customer's testing shows null... I'm trying to understand what configuration or conditions I'm missing in my tests.

#3 - 11/16/2022 06:22 AM - Marian Edu

Constantin Asofiei wrote:

> Marian, do you know in what conditions OE's JavaOpenClient would return null instead of a default value for integer,int64 and logical data-types?
>
> I ask because I'm testing with the customer's OE jars, and I still get the same 'false' instead of 'null' for an unset boolean/logical field. And the customer's testing shows null... I'm trying to understand what configuration or conditions I'm missing in my tests.

Constantin, you might want to check for any typos there... if the field exists (check metadata) it should return the default value for a new row. At least this is my experience with java open client.

#4 - 11/16/2022 06:33 AM - Constantin Asofiei

Marian Edu wrote:

> Constantin, you might want to check for any typos there... if the field exists (check metadata) it should return the default value for a new row. At least this is my experience with java open client.

I understand, but the customer really sees 'null' for a logical field, in a newly created row (something like row = dsTest.createProDataObject("tt1");, and after that calling row.get("logical-field");) at the JavaOpenClient code. It doesn't make any sense.

I don't see anything related to an 'initial' value for the OE generated Java proxy code.

#5 - 11/16/2022 11:17 AM - Constantin Asofiei

The issue in the end is described by https://community.progress.com/s/article/P133384 , calling RunTimeProperties.setDatasetNullInitials(true) will force the get() to return null for such cases. I'm adding this support to 6129b.

#7 - 11/16/2022 01:34 PM - Constantin Asofiei

- Subject changed from OE 11.6 vs OE 11.7 OpenClient differences with null values at the DataObject (table) fields to OpenClient with null values at the DataObject (table) fields for native types

- Status changed from WIP to Review

- % Done changed from 0 to 100

Added in 6129b/14324.

#8 - 11/16/2022 01:40 PM - Constantin Asofiei

- % Done changed from 100 to 90

- Status changed from Review to WIP

denormalized extent support is not done at this time.