

Question

[Emmanuel Vanlaar](#) · May 24, 2020

Python binding type conversions in Class Queries

Hi,

I was playing around with the python binding for caché (2018.1.4) and I ran into some problems.

When executing the class query "List" from SYS.Database the pythonbind interface throws an exception, that seems to be caused by a mismatch of the defined SQL datatypes for this query and what is actually returned (or the lack of type conversion in the pythonbind interface).

Is this a known issue ? Are there solutions/workarounds for this,
Or is this a bug ?

The details:

When executing "List" from SYS.Database there is an error for every datarow fetched.
When executing "FreeSpace" from SYS.Database there is an error for some datarows.

The errors are (same location, but different error code):

```
file=intersys/pythonbind3.c line=3516 err=22005 message=[Cache ODBC][State :  
22005][Native Code 22005]  
[libcacheodbciw.so]  
ERROR #388: Unknown error, code 22005
```

and

```
file=intersys/pythonbind3.c line=3516 err=1007 message=[Cache ODBC][State :  
01S07][Native Code 1007]  
[libcacheodbciw.so]  
ERROR #388: Unknown error, code 1007
```

When I look at the definition for the FreeSpace query of SYS.Database I see that field 'AvailableNum' is defined as %Integer, however the actual returned value seems to be a float (for some rows). When executing this query in cache terminal this becomes clear;

The value for AvailableNum of the CACHE database for example is returned as '9.9':

```
%SYS> Do ##class(%ResultSet).RunQuery("SYS.Database", "FreeSpace", "")  
DatabaseName:Directory:MaxSize:Size:ExpansionSize:Available:%  
Free:DiskFreeSpace:Status:SizeInt:AvailableNum:DiskFreeSpaceNum:ReadOnly:  
CACHESYS:/usr/cache/mgr/:Unlimited:170MB:System  
Default:112MB:65.88:6.23GB:Mounted/RW:170:112:6379:0:  
CACHE:/usr/cache/mgr/cache/:Unlimited:11MB:System  
Default:9.9MB:90:6.23GB:Mounted/RW:11:9.9:6379:0:  
CACHEAUDIT:/usr/cache/mgr/cacheaudit/:Unlimited:1MB:System  
Default:0.34MB:34:6.23GB:Mounted/RW:1:0.34:6379:0:  
CACHELIB:/usr/cache/mgr/cachelib/:Unlimited:500MB:System  
Default:25MB:5:6.23GB:Mounted/R:500:25:6379:1:  
CACHETEMP:/usr/cache/mgr/cachetemp/:Unlimited:11MB:System  
Default:9.6MB:87.27:6.23GB:Mounted/RW:11:9.6:6379:0:  
DOCBOOK:/usr/cache/mgr/docbook/:Unlimited:200MB:System
```

```

Default:2.6MB:1.3:6.23GB:Mounted/RW:200:2.6:6379:0:
SAMPLES:/usr/cache/mgr/samples/:Unlimited:102MB:System
Default:9.8MB:9.6:6.23GB:Mounted/RW:102:9.8:6379:0:
USER:/usr/cache/mgr/user/:Unlimited:1MB:System
Default:0.43MB:43:6.23GB:Mounted/RW:1:0.43:6379:0:
%SYS>

```

In case of the 'List' query, the definition for 'List' of SYS.Database specifies %Boolean as type for the field 'Encrypted' (which becomes SQLINTEGER in pythonbind) , however this is actually returned as 'No'/'Yes', which also throws an error in the pythonbind interface.

Note that when executing these queries via JDBC they seem to return different results;

Here the 'Encrypted' values are actually '0'

```
call SYS.Database_List(,)
```

Directory	MaxSize	Size	Status	Resource	Encrypted
StateInt Mirrored SFN					
-----	-----	----	-----	-----	-----
/usr/cache/mgr/	Unlimited	170	Mounted/RW	%DB_CACHESYS	0
Mounted/RW 0 0					
/usr/cache/mgr/cache/	Unlimited	11	Mounted/RW	%DB_CACHE	0
Mounted/RW 0 3					
/usr/cache/mgr/cacheaudit/	Unlimited	1	Mounted/RW	%DB_CACHEAUDIT	0
Mounted/RW 0 4					
/usr/cache/mgr/cachelib/	Unlimited	500	Mounted/R	%DB_CACHELIB	0
Mounted/R 0 1					
/usr/cache/mgr/cachetemp/	Unlimited	11	Mounted/RW	%DB_CACHETEMP	0
Mounted/RW 0 2					
/usr/cache/mgr/docbook/	Unlimited	200	Mounted/RW	%DB_DOCBOOK	0
Mounted/RW 0 6					
/usr/cache/mgr/samples/	Unlimited	102	Mounted/RW	%DB_SAMPLES	0
Mounted/RW 0 7					
/usr/cache/mgr/user/	Unlimited	1	Mounted/RW	%DB_USER	0
Mounted/RW 0 5					

While in the cache terminal they are 'No':

```

%SYS>Do ##class(%ResultSet).RunQuery("SYS.Database","List","")
Directory:MaxSize:Size:Status:Resource:Encrypted:StateInt:Mirrored:SFN:
/usr/cache/mgr/:Unlimited:170:Mounted/RW:%DB_CACHESYS:No:Mounted/RW:0:0:
/usr/cache/mgr/cache/:Unlimited:11:Mounted/RW:%DB_CACHE:No:Mounted/RW:0:3:
/usr/cache/mgr/cacheaudit/:Unlimited:1:Mounted/RW:%DB_CACHEAUDIT:No:Mounted/RW:0:4:
/usr/cache/mgr/cachelib/:Unlimited:500:Mounted/R:%DB_CACHELIB:No:Mounted/R:0:1:
/usr/cache/mgr/cachetemp/:Unlimited:11:Mounted/RW:%DB_CACHETEMP:No:Mounted/RW:0:2:
/usr/cache/mgr/docbook/:Unlimited:200:Mounted/RW:%DB_DOCBOOK:No:Mounted/RW:0:6:
/usr/cache/mgr/samples/:Unlimited:102:Mounted/RW:%DB_SAMPLES:No:Mounted/RW:0:7:
/usr/cache/mgr/user/:Unlimited:1:Mounted/RW:%DB_USER:No:Mounted/RW:0:5:
%SYS>

```

Simple Python test script (output from my system is below the script):

```

#!/bin/python3
import codecs, sys, traceback
import intersys.pythonbind3
from enum import Enum

```

```
url="localhost[1972]:%SYS"
```

```
conn = intersys.pythonbind3.connection( )
```

```
conn.connectnow(url, "system", "*****", None)
```

```
class SqlType(Enum):
```

```
    UNKNOWN_TYPE = 0
```

```
    CHAR = 1
```

```
    NUMERIC = 2
```

```
    DECIMAL = 3
```

```
    INTEGER = 4
```

```
    SMALLINT = 5
```

```
    FLOAT = 6
```

```
    REAL = 7
```

```
    DOUBLE = 8
```

```
    DATETIME = 9
```

```
    VARCHAR = 12
```

```
    TIME = 10
```

```
    TIMESTAMP = 11
```

```
    LONGVARCHAR = (-1)
```

```
    BINARY = (-2)
```

```
    VARBINARY = (-3)
```

```
    LONGVARBINARY = (-4)
```

```
    BIGINT = (-5)
```

```
    TINYINT = (-6)
```

```
    BIT = (-7)
```

```
    GUID = (-11)
```

```
def dofetech(query):
```

```
    try:
```

```
        return query.fetch([None])
```

```
    except Exception as err:
```

```
        # force a non-empty result on error
```

```
        return [ str(err) ]
```

```
def runfetech(query):
```

```
    sqlcode = query.execute()
```

```
    names = [ query.colname(i+1) for i in range(query.numcols()) ]
```

```
    types = [ (SqlType(query.colsqltype(i+1))).name for i in range(query.numcols()) ]
```

```
    print(names)
```

```
    print(types)
```

```
    datalist = dofetech(query)
```

```
    while datalist:
```

```
        i = 1
```

```
        print(datalist)
```

```
        datalist = dofetech(query)
```

```
database = intersys.pythonbind3.database(conn)
```

```
query = intersys.pythonbind3.query(database)
```

```
ClassQueries=[ "CompactLocalList", "FreeSpace", "List" ]
```

```
for q in ClassQueries:
```

```
    print(" h h%S" % q)
```

```
    print("-----")
```

```
    sqlcode = query.prepareclass("SYS.Database", q)
```

```
    runfetech(query)
```

And the output:

CompactLocalList

```
-----
['Directory', 'SFN', 'ReadOnly', 'Resource', 'BlockSize', 'Collation']
['VARCHAR', 'INTEGER', 'INTEGER', 'VARCHAR', 'INTEGER', 'INTEGER']
['/usr/cache/mgr/', 0, 0, '0', 8, 5]
['/usr/cache/mgr/cache/', 3, 0, '0', 8, 5]
['/usr/cache/mgr/cacheaudit/', 4, 0, '0', 8, 5]
['/usr/cache/mgr/cachelib/', 1, 1, '0', 8, 5]
['/usr/cache/mgr/cachetemp/', 2, 0, '0', 8, 5]
['/usr/cache/mgr/docbook/', 6, 0, '0', 8, 5]
['/usr/cache/mgr/samples/', 7, 0, '0', 8, 5]
['/usr/cache/mgr/user/', 5, 0, '0', 8, 5]
```

FreeSpace

```
-----
['DatabaseName', 'Directory', 'MaxSize', 'Size', 'ExpansionSize', 'Available',
'Free', 'DiskFreeSpace', 'Status', 'SizeInt', 'AvailableNum', 'DiskFreeSpaceNum',
'ReadOnly']
['VARCHAR', 'VARCHAR', 'VARCHAR', 'VARCHAR', 'VARCHAR', 'VARCHAR', 'DOUBLE',
'VARCHAR', 'VARCHAR', 'INTEGER', 'INTEGER', 'INTEGER', 'INTEGER']
['CACHESYS', '/usr/cache/mgr/', 'Unlimited', '170MB', 'System Default', '112MB',
65.88, '6.23GB', 'Mounted/RW', 170, 112, 6379, 0]
['file=intersys/pythonbind3.c line=3516 err=1007 message=[Cache ODBC][State :
01S07][Native Code 1007]\n[libcacheodbcw.so]\nERROR #388: Unknown error, code
1007\n']
['file=intersys/pythonbind3.c line=3516 err=1007 message=[Cache ODBC][State :
01S07][Native Code 1007]\n[libcacheodbcw.so]\nERROR #388: Unknown error, code
1007\n']
['CACHELIB', '/usr/cache/mgr/cachelib/', 'Unlimited', '500MB', 'System Default',
'25MB', 5.0, '6.23GB', 'Mounted/R', 500, 25, 6379, 1]
['file=intersys/pythonbind3.c line=3516 err=1007 message=[Cache ODBC][State :
01S07][Native Code 1007]\n[libcacheodbcw.so]\nERROR #388: Unknown error, code
1007\n']
['file=intersys/pythonbind3.c line=3516 err=1007 message=[Cache ODBC][State :
01S07][Native Code 1007]\n[libcacheodbcw.so]\nERROR #388: Unknown error, code
1007\n']
['file=intersys/pythonbind3.c line=3516 err=1007 message=[Cache ODBC][State :
01S07][Native Code 1007]\n[libcacheodbcw.so]\nERROR #388: Unknown error, code
1007\n']
['file=intersys/pythonbind3.c line=3516 err=1007 message=[Cache ODBC][State :
01S07][Native Code 1007]\n[libcacheodbcw.so]\nERROR #388: Unknown error, code
1007\n']
```

List

```
-----
['Directory', 'MaxSize', 'Size', 'Status', 'Resource', 'Encrypted', 'StateInt',
'Mirrored', 'SFN']
['VARCHAR', 'VARCHAR', 'INTEGER', 'VARCHAR', 'VARCHAR', 'INTEGER', 'VARCHAR',
'INTEGER', 'VARCHAR']
['file=intersys/pythonbind3.c line=3516 err=22005 message=[Cache ODBC][State :
22005][Native Code 22005]\n[libcacheodbcw.so]\nERROR #388: Unknown error, code
22005\n']
['file=intersys/pythonbind3.c line=3516 err=22005 message=[Cache ODBC][State :
22005][Native Code 22005]\n[libcacheodbcw.so]\nERROR #388: Unknown error, code
22005\n']
['file=intersys/pythonbind3.c line=3516 err=22005 message=[Cache ODBC][State :
```

```
22005][Native Code 22005]\n[libcacheodbcw.so]\nERROR #388: Unknown error, code
22005\n']
['file=intersys/pythonbind3.c line=3516 err=22005 message=[Cache ODBC][State :
22005][Native Code 22005]\n[libcacheodbcw.so]\nERROR #388: Unknown error, code
22005\n']
['file=intersys/pythonbind3.c line=3516 err=22005 message=[Cache ODBC][State :
22005][Native Code 22005]\n[libcacheodbcw.so]\nERROR #388: Unknown error, code
22005\n']
['file=intersys/pythonbind3.c line=3516 err=22005 message=[Cache ODBC][State :
22005][Native Code 22005]\n[libcacheodbcw.so]\nERROR #388: Unknown error, code
22005\n']
['file=intersys/pythonbind3.c line=3516 err=22005 message=[Cache ODBC][State :
22005][Native Code 22005]\n[libcacheodbcw.so]\nERROR #388: Unknown error, code
22005\n']
```

[#Python](#) [#Caché](#)

Source URL: <https://community.intersystems.com/post/python-binding-type-conversions-class-queries>