Storing/retrieving BYTEA and BLOB with Java/Scala: "java.sql.SQLFeatureNotSupportedException"

Asked 10 years, 7 months ago Modified 10 years, 7 months ago Viewed 4k times

2

The situation is as follows: I'm trying to store/retrieve byte[] arrays and BLOBS in a PostgreSQL database.

I'm using Scala Play with Anorm, but for dealing with BYTEA and BLOB's I've regressed to using plain old java.sql.Connection API and roughly followed this description:

http://jdbc.postgresql.org/documentation/80/binary-data.html

My code looks as follows:

```
def saveSmallImage(id: String, img: SmallImage): Unit = {
  DB.withConnection { implicit conn =>

  val ps = conn.prepareStatement(
    "INSERT INTO SmallImage(id, mimeType, image) VALUES (?, ?, ?)")
  ps.setString(1, id)
  ps.setString(2, img.mimeType)
  ps.setBinaryStream(3, img.binaryStream, img.length)
  ps.executeUpdate()
  ps.close()
}
}
```

where 'image' column is of type BYTEA, img.binaryStream is of type InputStream.

I'm getting this error:

```
java.sql.SQLFeatureNotSupportedException: Method
org.postgresql.jdbc4.Jdbc4PreparedStatement.
```

```
  setBinaryStream(int, InputStream, long)
is not yet implemented.
```

I have the following dependency in my build.sbt:

```
"postgresql" % "postgresql" % "9.1-901-1.jdbc4"
```

(this corresponds to groupId, artifactId, version in maven POM-files). This seems to be the most recent one. The Postgres Manual tells me, that this feature has been present since PostgreSQL version 7.2.

What's going on? Is PostgreSQL/JDBC somehow deprecated and no longer maintained? Should I take another version to make it work? Or is SQL completely dead, and I should switch to some other database? Is it somehow a generally bad idea to use Postgres with Java? Could it somehow matter that my postgres version is 9.3 but I'm using JDBC 9.1-x?

I would be very thankful if anyone tells me what I'm doing wrong and how to make it work.

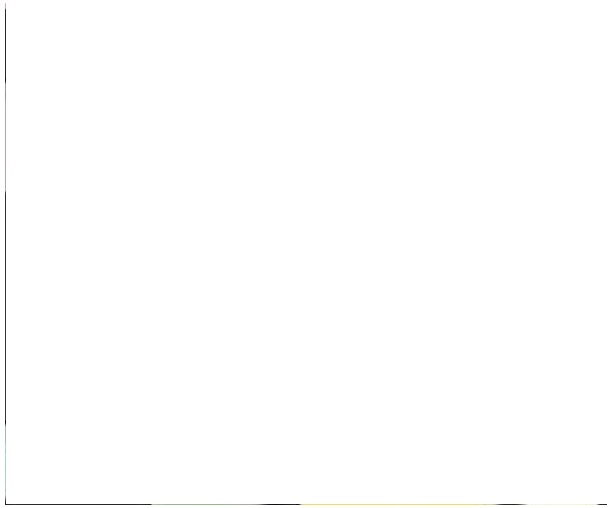
PS: ... Tried to git-clone pgjdbc. It requires ant 1.4.1 to build. This ant version is ~8 years old and no longer available at the website of the Apache foundation. Isn't it kind of eerie?

java postgresql scala jdbc

edited Jun 25, 2014 at 21:30

asked Jun 25, 2014 at 20:36

Andrey Tyukin
44.9k 4 64 101



2 Answers

Sorted by: Highest score (default)

2

This seems to be more up-to-date one, no?

"org.postgresql" % "postgresql" % "9.3-1100-jdbc41"

Share Edit Follow

answered Jun 25, 2014 at 21:35

Ashalynd
12.6k 2 35 38

1 Yes, this works too, this is much better than my proposal! I've just searched for the wrong groupId, under groupId="postgresql" there is only deprecated stuff that does not work... But "org.postgresql" contains the newest stuff, and now everything works fine. Thank you. – Andrey Tyukin Jun 25, 2014 at 21:55

The solution turned out to be rather trivial:

2

- Download the most recent version of the JDBC driver directly from the project page:

http://jdbc.postgresql.org/download.html

- create 'lib' directory in the root folder of the play project, just put the jar there.
- remove the dependency from the build.sbt script.

So, the problem was that the most recent stuff in Maven Repositories is outdated and malfunctioning: it's just a distribution problem. Make an exception, manage this dependency manually, then it should work.

EDIT: the other answer is much better.

Share Edit Follow

edited Jun 25, 2014 at 21:53

answered Jun 25, 2014 at 21:46

Andrey Tyukin
44.9k 4 64 101