You.areCurrent?(Hibernate 5)

Jason Porter

Senior Sofware Engineer

lightguardjp
jporter@redhat.com

Agenda

Spatial

Agenda

- Spatial
- Java 8

Agenda

- Spatial
- Java 8
- Misc Enhancements

Agenda

- Spatial
- Java 8
- Misc Enhancements
- Migration

HIBERNATE SPATIAL

What does it do

Native support for spatial database types

HIBERNATE

What does it do

Native support for spatial database types
Extends HQL/JPQL with spatial functions

HIBERNATE

Now part of Hibernate proper

HIBERNATE

Uses Geolatte now

HIBERNATE

GIS Functions

HIBERNATE

Disjoint
Touches
Crosses
Within
Overlaps
Contains
Intersects
Postgis &&

HIBERNATE

Java
JAVA 8

Java 8 Dependency

```
<dependency>
  <groupId>org.hibernate</groupId>
  <artifactId>hibernate-java8</artifactId>
  <version>${hibernate.version}</version>
</dependency>
```

HIBERNATE

Sorry, no

lambdas

HIBERNATE

Sorry, no

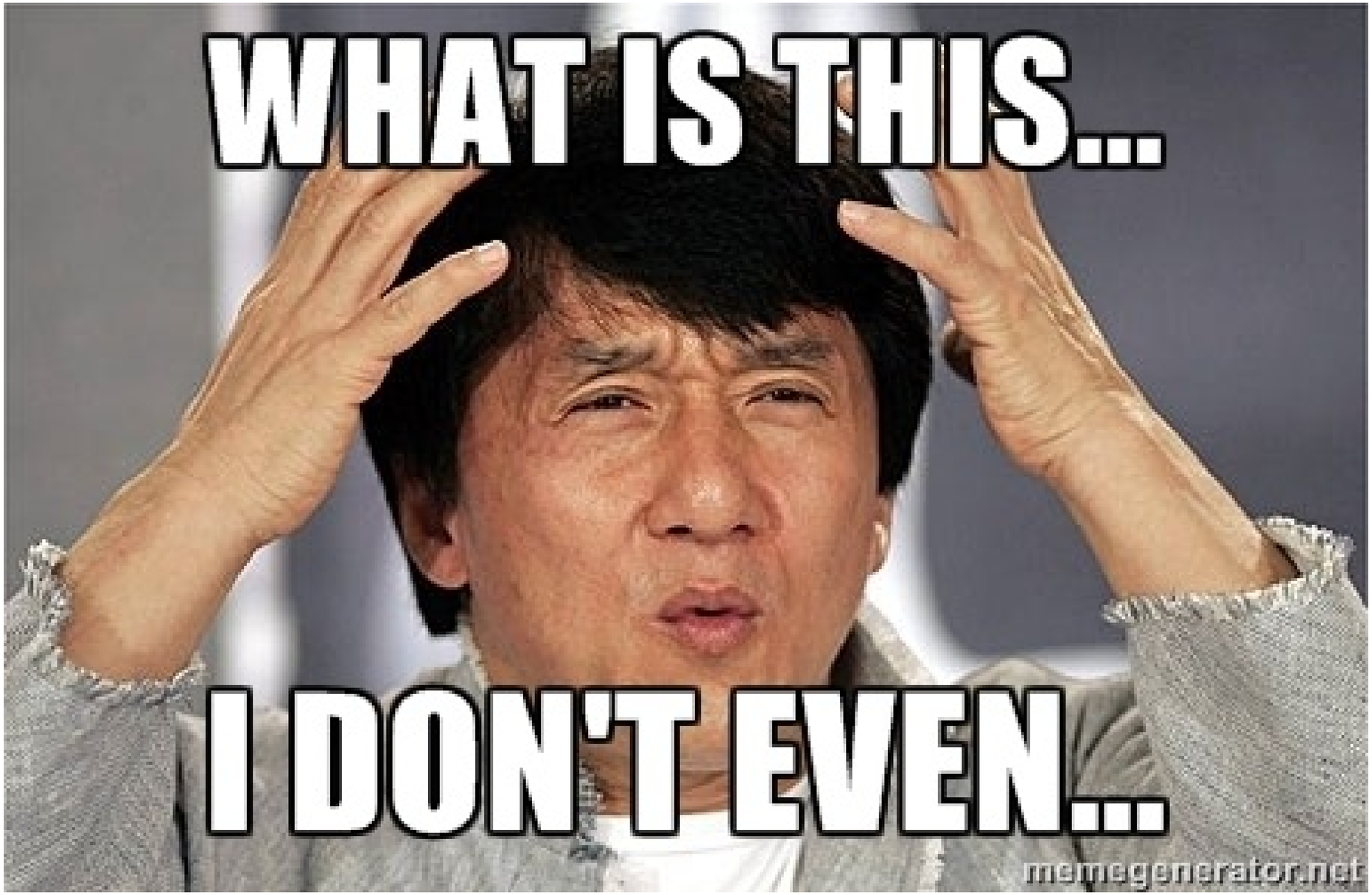
lambdas
streams

HIBERNATE

Sorry, no

lambdas
streams
other awesome language features

HIBERNATE


WHAT IS THIS...
I DON'T EVEN...
memegenerator.net

What we do have

JSR 310 support

HIBERNATE

What we do have

JSR 310 support
Java 5, 6, 7 - Generics (um, duh), blocks autoclose

HIBERNATE

Hibernate type (org.hibernate.type package)	JDBC type	Java type	BasicTypeRegistry key(s)
DurationType	BIGINT	java.time.Duration	Duration, java.time.Duration
InstantType	TIMESTAMP	java.time.Instant	Instant, java.time.Instant
LocalDateTimeType	TIMESTAMP	java.time.LocalDateTime	LocalDateTime, java.time.LocalDateTime
LocalDateType	DATE	java.time.LocalDate	LocalDate, java.time.LocalDate
LocalTimeType	TIME	java.time.LocalTime	LocalTime, java.time.LocalTime
OffsetDateTimeType	TIMESTAMP	java.time.OffsetDateTime	OffsetDateTime, java.time.OffsetDateTime
OffsetTimeType	TIME	java.time.OffsetTime	OffsetTime, java.time.OffsetTime
OffsetTimeType	TIMESTAMP	java.time.ZonedDateTime	ZonedDateTime, java.time.ZonedDateTime

HIBERNATE

auto closable

Session / StatelessSession

HIBERNATE

auto closable

Session / StatelessSession
SessionFactor

HIBERNATE

auto closable

Session / StatelessSession
SessionFactor
ScrollableResults

HIBERNATE

```
try (Session session = sessionFactory.openSession ()) {
  Address address = Session.get (Address.class, id);
  address.getCity return ();
}
```

HIBERNATE

MISC ENHANCEMENTS

Second Level Cache

Fewer objects created (key)

HIBERNATE

Second Level Cache

Fewer objects created (key)
Cache by reference

HIBERNATE

```java
@Entity @Proxy( lazy = false )
@Immutable
@Cacheable @Cache( usage = CacheConcurrencyStrategy.READ_ONLY )
public class MyReferenceData {
    @Id
    private Integer id;
    private String name;
    private String theValue;
    ....
}
```

HIBERNATE

OSGi Improvements

HIBERNATE

bytecode enhancements

Smarter change detection

HIBERNATE

bytecode enhancements

Smarter change detection
Better gradle / maven support

HIBERNATE

bytecode enhancements

Smarter change detection
Better gradle / maven support
Lazy loaded properties

HIBERNATE

bytecode enhancements

Smarter change detection
Better gradle / maven support
Lazy loaded properties
Automatic bi-directional management

HIBERNATE

Maven

HIBERNATE

```
<plugin>
   <groupId>org.hibernate.orm.tooling</groupId>
   <artifactId>hibernate-enhance-maven-plugin</artifactId>
   <version>${currentHibernateVersion}</version>
   <executions>
       <execution>
           <configuration>
               <failOnError>true</failOnError>
               <enableLazyInitialization>true</enableLazyInitialization>
               <enableDirtyTracking>true</enableDirtyTracking>
               <enableAssociationManagement>true</enableAssociationManagement>
           </configuration>
           <goals>
               <goal>enhance</goal>
           </goals>
       </execution>
   </executions>
</plugin>
```

HIBERNATE

Gradle

HIBERNATE

```
ext {
    hibernateVersion = 'hibernate-version-you-want'
}

buildscript {
    dependencies
    classpath "org.hibernate: hibernate-gradle-plugin: $ hibernateVersion"
}

hibernate
  enhance {
      // Any configuration goes here
  }
}
```

HIBERNATE

Lazy loaded Properties

HIBERNATE

```java
@Entity
public class Customer {
  @Id
  private Integer id;

  private String name;

  @Basic(Fetch = FetchType.LAZY)
  private UUID accountsPayableXrefId;

  @LobBasic(Fetch = FetchType.LAZY)
  @LazyGroup("Lobs")
  private Blob image;
  ...
}
```

HIBERNATE

Bi-Directional Association Management

Currently

```
Order order = new Order();
LineItem lineItem = new LineItem();
order.getLineItems().add(lineItem).
lineItem.setOrder(order);

lineItem.getOrder.getName();
```

HIBERNATE

Bi-Directional Association Management

With bytecode enhancement

```
Order order = new Order();
LineItem lineItem = new LineItem();
order.getLineItems().add(lineItem).

// Normally this would throw NPE
lineItem.getOrder.getName();
```

HIBERNATE

Perf Improvements

sorry no numbers

HIBERNATE

Improvements to GenerateId.AUTO

JPA number support

HIBERNATE

Improvements to GenerateId.AUTO

JPA number support
UUID

HIBERNATE

Improvements to GenerateId.AUTO

JPA number support
UUID
new org.hibernate.boot.model.IdGeneratorStrategyInterpreter extension

HIBERNATE

MIGRATION SMALL LIST

https://github.com/hibernate/hibernate-orm/blob/5.0/migration-guide.adoc

HIBERNATE

Bootstrap

HIBERNATE

```
StandardServiceRegistry standardRegistry = new StandardServiceRegistryBuilder()
 .configure("org/hibernate/example/MyCfg.xml")
 .build();

Metadata metadata = new MetadataSources(standardRegistry)
  .addAnnotatedClass(MyEntity.class)
  .addAnnotatedClassName("org.hibernate.example.Customer")
  .addResource("org/hibernate/example/Order.hbm.xml")
  .addResource("org/hibernate/example/Product.orm.xml")
  .getMetadataBuilder()
  .applyImplicitNamingStrategy(ImplicitNamingStrategyJpaCompliantImpl.INSTANCE)
  .build();

SessionFactory sessionFactory = metadata.getSessionFactoryBuilder()
  .applyBeanManager(getBeanManagerFromSomewhere())
  //CDI BeanManager to the SessionFactory being built
  .build();
```

HIBERNATE

Naming strategies

HIBERNATE

ImplicitNamingStrategy

HIBERNATE

ImplicitNamingStrategy
PhysicalNamingStrategy

HIBERNATE

Type handling

HIBERNATE

Minor package updates

HIBERNATE

Minor package updates

SqlTypeDescriptor no longer auto-registers

HIBERNATE

Minor package updates

SqlTypeDescriptor no longer auto-registers "big_integer" from java.sql.Types.NUMBERIC to java.sql.Types.BIGINT

HIBERNATE

Minor package updates

SqlTypeDescriptor no longer auto-registers "big_integer" from java.sql.Types.NUMBERIC to java.sql.Types.BIGINT

UUID may require explicit `@Column(length16)`

HIBERNATE

Minor package updates

SqlTypeDescriptor no longer auto-registers "big_integer" from java.sql.Types.NUMBERIC to java.sql.Types.BIGINT

UUID may require explicit `@Column(length16)`

EnumType mappings in hbm.xml may need tweaking for name-mapping

HIBERNATE

Hibernate OGM

HIBERNATE

Hibernate Search

HIBERNATE