Callable and Querybable Objects over NoSQL

Context and Problematic

Treasure Rush Game

- Aspect of the game: a maze where heroes go through different shared rooms and compete in order to collect Treasures.
- Handling concurrency and critical sections using « Synchronized »
  - How can we easily port a shared memory in a distributed settings?
  - How can we query shared objects?

State of the art

CRESON: Callable and Replicated Objects over NoSQL

- Synchronization framework:
  - No Further action required from the developer to guarantee synchronization
  - Using annotations @Shared
- Based on Infinispan:
  - Distributed Key-value Store
  - Remote Server
  - Fluent programmatic configuration API
- Remote callable objects:
  - No need for ORM and serialization to send objects over the network

Solution Design

Indexing:

- Accelerating the fetch of objects from the store
- Using a specific data structure (B-Tree) to reduce the query response time
- Annotating indexed classes using Hibernate Search

Querying:

- Implementing an API:
  - based on Java Objects
  - supporting Icke query Language
- Querying Creson Objects in a simple way

```java
Query q = qf.createQuery("from org.example.Room room" + 
  " where room.treasure = 0");
```