# Enterprise Component Development

## Assignment 2

20th April 2012
Michael Dreeling

# Added Features

4 new service features have been added

- Add a **LicensingDeal** (A deal between a player and a company for one of their **Product**s)
- Add a **Product** to be used during a **LicensingDeal**
- Add a **Result** between two **Club**s
- Add some win/loss/point statistics to a **Club** using a **ClubStat**

All 4 services above are implemented in the UI using new Commands and have accompanying unit tests.

The accompanying GUI helpers were also added to the AbstractCommand class (i.e getProduct)

There is a triggered lazy load (below) in the AbstractCommand class when the company is printed alongside the Product List.

```java
/**
 * @author Michael Dreeling
 *         New method to return product
 * @return
 */
public Product getProduct() {
        Product result = null;
        boolean finished = false;
        List<Product> prds = leagueService.readProducts();

        for (Product p : prds) {

                System.out.println(p.getId() + ". " + p.getName() + " ("
                        + p.getCompany().getName()
                        + ")");
        }
```
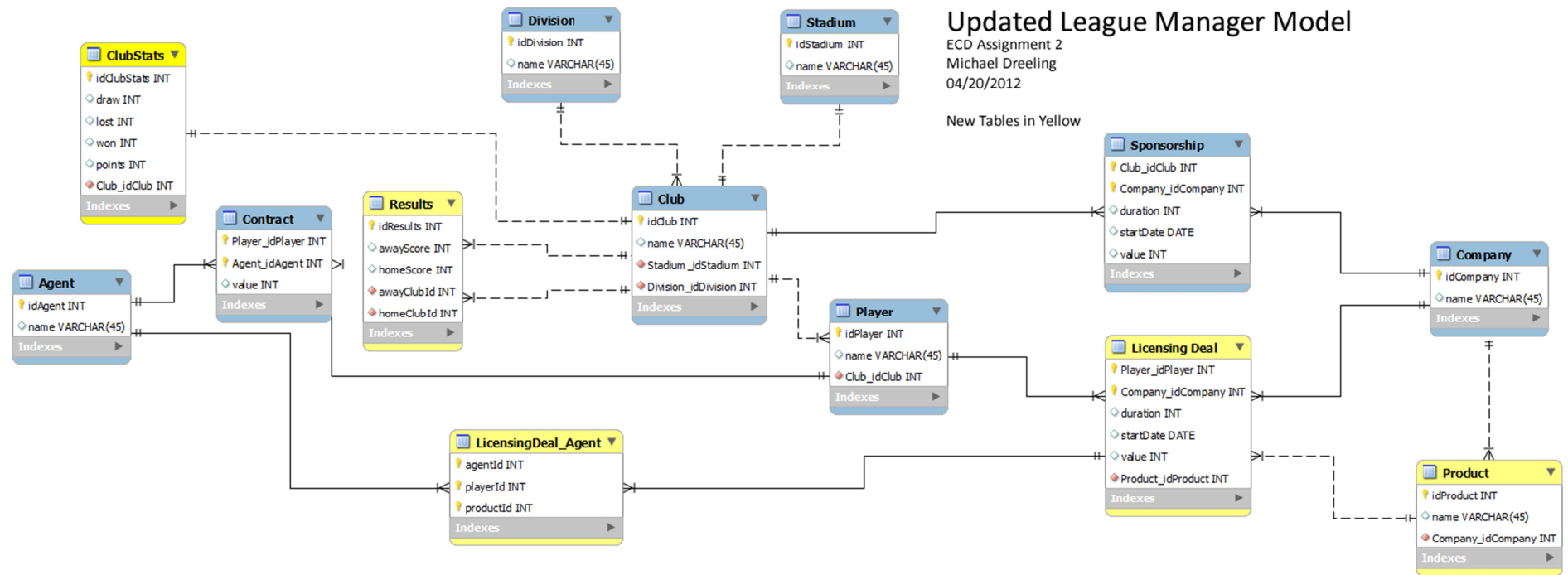
# Enhanced Domain Model

5 new domain objects have been added

- **ClubStats** - Stats for a Club
  - o One to One with Club
- **Result** - Results for a match
  - o Many to One with Clubs
- **LicensingDeal** - Product Licensing Deals for Players
  - o Many to One with Player
  - o Many to One with Products
  - o Many to Many with Agents (involved in the deal)
- **LicensingDeal_Agent** - Join Table for N-M relationship
- **Product** - Product involved in a licensing deal
  - o Many to One with Company

# Enhanced Class Diagram



Updated League Manager Model
ECD Assignment 2
Michael Dreeling
04/20/2012

New Tables in Yellow

**ClubStats**
- idClubStats INT
- draw INT
- lost INT
- won INT
- points INT
- Club_idClub INT
- Indexes

**Division**
- idDivision INT
- name VARCHAR(45)
- Indexes

**Stadium**
- idStadium INT
- name VARCHAR(45)
- Indexes

**Sponsorship**
- Club_idClub INT
- Company_idCompany INT
- duration INT
- startDate DATE
- value INT
- Indexes

**Contract**
- Player_idPlayer INT
- Agent_idAgent INT
- value INT
- Indexes

**Results**
- idResults INT
- awayScore INT
- homeScore INT
- awayClubId INT
- homeClubId INT
- Indexes

**Club**
- idClub INT
- name VARCHAR(45)
- Stadium_idStadium INT
- Division_idDivision INT
- Indexes

**Company**
- idCompany INT
- name VARCHAR(45)
- Indexes

**Agent**
- idAgent INT
- name VARCHAR(45)
- Indexes

**Player**
- idPlayer INT
- name VARCHAR(45)
- Club_idClub INT
- Indexes

**Licensing Deal**
- Player_idPlayer INT
- Company_idCompany INT
- duration INT
- startDate DATE
- value INT
- Product_idProduct INT
- Indexes

**LicensingDeal_Agent**
- agentId INT
- playerId INT
- productId INT
- Indexes

**Product**
- idProduct INT
- name VARCHAR(45)
- Company_idCompany INT
- Indexes

# Performance

To address performance and transaction propagation I performed the following

- Marked all read only methods as **readOnly**
- Added **REQUIRED** to all methods which change data

This was performed using one of the non-standard features below.

# Non Standard Features

I tried to use some features that I did not come across in the lecture notes. Some of them may not make sense in a real world scenario.
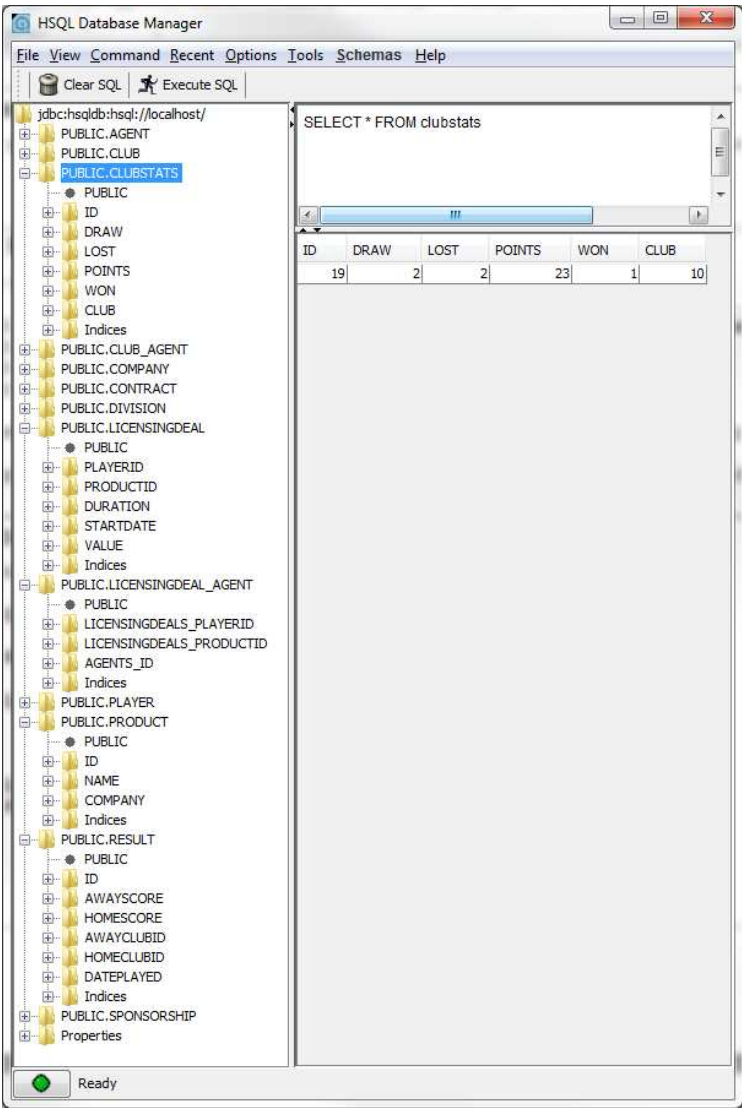
### Isolation Level Annotations

I have added

- isolation = Isolation.READ_UNCOMMITTED

  to the public List<Division> readDivisions() method. In this case I am trying to dirty-read the data (for some reason

### Spring Meta Annotations

I created 2 new annotations in the **doconnor.jpa.annotations.meta** package which are short hand for my 2 transaction types

- LMReadOnly (has REQUIRED and readOnly)

- LMWrite (has REQUIRED and a default 30 second timeout)

Service Specification



HSQL Working screen shot

---

Service Specification

# Sample Outputs

## Add Club Statistics

```
   --> addClubStat
Hibernate: select club0_.id as id6_, club0_.name as name6_, club0_.capacity as capacity6_, club0_.ticket_price as ticket4_6_, club0_.title as title6_ from Club
club0_
Hibernate: select clubstats0_.id as id2_1_, clubstats0_.club as club2_1_, clubstats0_.draw as draw2_1_, clubstats0_.lost as lost2_1_, clubstats0_.points as
points2_1_, clubstats0_.won as won2_1_, club1_.id as id6_0_, club1_.name as name6_0_, club1_.capacity as capacity6_0_, club1_.ticket_price as ticket4_6_0_,
club1_.title as title6_0_ from ClubStats clubstats0_ left outer join Club club1_ on clubstats0_.club=club1_.id where clubstats0_.club=?
Hibernate: select clubstats0_.id as id2_1_, clubstats0_.club as club2_1_, clubstats0_.draw as draw2_1_, clubstats0_.lost as lost2_1_, clubstats0_.points as
points2_1_, clubstats0_.won as won2_1_, club1_.id as id6_0_, club1_.name as name6_0_, club1_.capacity as capacity6_0_, club1_.ticket_price as ticket4_6_0_,
club1_.title as title6_0_ from ClubStats clubstats0_ left outer join Club club1_ on clubstats0_.club=club1_.id where clubstats0_.club=?
Hibernate: select clubstats0_.id as id2_1_, clubstats0_.club as club2_1_, clubstats0_.draw as draw2_1_, clubstats0_.lost as lost2_1_, clubstats0_.points as
points2_1_, clubstats0_.won as won2_1_, club1_.id as id6_0_, club1_.name as name6_0_, club1_.capacity as capacity6_0_, club1_.ticket_price as ticket4_6_0_,
club1_.title as title6_0_ from ClubStats clubstats0_ left outer join Club club1_ on clubstats0_.club=club1_.id where clubstats0_.club=?
1 c1
2 c2
3 Liverpool
Which Club ?3
won?1
lost? 2
draw?2
points?23
Hibernate: insert into ClubStats (id, club, draw, lost, points, won) values (null, ?, ?, ?, ?, ?)
Hibernate: call identity()
   -->
```

## Add Licensing Deal

```
League Manager.
Type 'help' for a list of commands.
  --> addLicensingDeal
Hibernate: select product0_.id as id7_, product0_.company as company7_, product0_.name as name7_ from Product product0_
Hibernate: select company0_.id as id3_0_, company0_.name as name3_0_ from Company company0_ where company0_.id=?
Gillette
1. ProGlide (Gillette)
Which product is being licensed? 1
Hibernate: select player0_.id as id8_, player0_.club as club8_, player0_.name as name8_ from Player player0_
Hibernate: select club0_.id as id6_1_, club0_.name as name6_1_, club0_.capacity as capacity6_1_, club0_.ticket_price as ticket4_6_1_, club0_.title as title6_1_,
clubstats1_.id as id2_0_, clubstats1_.club as club2_0_, clubstats1_.draw as draw2_0_, clubstats1_.lost as lost2_0_, clubstats1_.points as points2_0_,
clubstats1_.won as won2_0_ from Club club0_ left outer join ClubStats clubstats1_ on club0_.id=clubstats1_.club where club0_.id=?
Hibernate: select contract0_.agentId as agentId10_4_, contract0_.playerId as playerId10_4_, contract0_.player as player10_4_, contract0_.value as value10_4_,
agent1_.id as id0_0_, agent1_.name as name0_0_, player2_.id as id8_1_, player2_.club as club8_1_, player2_.name as name8_1_, club3_.id as id6_2_, club3_.name as
name6_2_, club3_.capacity as capacity6_2_, club3_.ticket_price as ticket4_6_2_, club3_.title as title6_2_, clubstats4_.id as id2_3_, clubstats4_.club as club2_3_,
clubstats4_.draw as draw2_3_, clubstats4_.lost as lost2_3_, clubstats4_.points as points2_3_, clubstats4_.won as won2_3_ from Contract contract0
1   p4
2   p6
4   p1
5   p2
6   p3
7   p3
8   p5
9   p3
10  p5
11  p1
12  p2
21  Steven
Player ID?12
Duration (years)? 2
Value (millions)? 234
Adding licensing deal for player 12 and product ProGlide for 2 years @ 234.0 million
Hibernate: insert into LicensingDeal (duration, startDate, value, playerId, productId) values (?, ?, ?, ?, ?)
Persisted licensing deal.
  -->
```

## Add Result

```
  --> addResult
Match date? (dd/mm/yyyy)28/03/2012
Choose home clubHibernate: select club0_.id as id6_, club0_.name as name6_, club0_.capacity as capacity6_, club0_.ticket_price as ticket4_6_, club0_.title as
title6_ from Club club0_
Hibernate: select clubstats0_.id as id2_1_, clubstats0_.club as club2_1_, clubstats0_.draw as draw2_1_, clubstats0_.lost as lost2_1_, clubstats0_.points as
points2_1_, clubstats0_.won as won2_1_, club1_.id as id6_0_, club1_.name as name6_0_, club1_.capacity as capacity6_0_, club1_.ticket_price as ticket4_6_0_,
club1_.title as title6_0_ from ClubStats clubstats0_ left outer join Club club1_ on clubstats0_.club=club1_.id where clubstats0_.club=?
Hibernate: select clubstats0_.id as id2_1_, clubstats0_.club as club2_1_, clubstats0_.draw as draw2_1_, clubstats0_.lost as lost2_1_, clubstats0_.points as
points2_1_, clubstats0_.won as won2_1_, club1_.id as id6_0_, club1_.name as name6_0_, club1_.capacity as capacity6_0_, club1_.ticket_price as ticket4_6_0_,
club1_.title as title6_0_ from ClubStats clubstats0_ left outer join Club club1_ on clubstats0_.club=club1_.id where clubstats0_.club=?
Hibernate: select clubstats0_.id as id2_1_, clubstats0_.club as club2_1_, clubstats0_.draw as draw2_1_, clubstats0_.lost as lost2_1_, clubstats0_.points as
points2_1_, clubstats0_.won as won2_1_, club1_.id as id6_0_, club1_.name as name6_0_, club1_.capacity as capacity6_0_, club1_.ticket_price as ticket4_6_0_,
club1_.title as title6_0_ from ClubStats clubstats0_ left outer join Club club1_ on clubstats0_.club=club1_.id where clubstats0_.club=?
1 c1
2 c2
3 Liverpool
Which Club ?3
Choose away clubHibernate: select club0_.id as id6_, club0_.name as name6_, club0_.capacity as capacity6_, club0_.ticket_price as ticket4_6_, club0_.title as
title6_ from Club club0_
Hibernate: select clubstats0_.id as id2_1_, clubstats0_.club as club2_1_, clubstats0_.draw as draw2_1_, clubstats0_.lost as lost2_1_, clubstats0_.points as
points2_1_, clubstats0_.won as won2_1_, club1_.id as id6_0_, club1_.name as name6_0_, club1_.capacity as capacity6_0_, club1_.ticket_price as ticket4_6_0_,
club1_.title as title6_0_ from ClubStats clubstats0_ left outer join Club club1_ on clubstats0_.club=club1_.id where clubstats0_.club=?
Hibernate: select clubstats0_.id as id2_1_, clubstats0_.club as club2_1_, clubstats0_.draw as draw2_1_, clubstats0_.lost as lost2_1_, clubstats0_.points as
points2_1_, clubstats0_.won as won2_1_, club1_.id as id6_0_, club1_.name as name6_0_, club1_.capacity as capacity6_0_, club1_.ticket_price as ticket4_6_0_,
club1_.title as title6_0_ from ClubStats clubstats0_ left outer join Club club1_ on clubstats0_.club=club1_.id where clubstats0_.club=?
Hibernate: select clubstats0_.id as id2_1_, clubstats0_.club as club2_1_, clubstats0_.draw as draw2_1_, clubstats0_.lost as lost2_1_, clubstats0_.points as
points2_1_, clubstats0_.won as won2_1_, club1_.id as id6_0_, club1_.name as name6_0_, club1_.capacity as capacity6_0_, club1_.ticket_price as ticket4_6_0_,
club1_.title as title6_0_ from ClubStats clubstats0_ left outer join Club club1_ on clubstats0_.club=club1_.id where clubstats0_.club=?
1 c1
2 c2
3 Liverpool
Which Club ?2
Score? (Format is 'Home-Away')2-0
Hibernate: insert into Result (id, awayclubId, awayScore, datePlayed, homeclubId, homeScore) values (null, ?, ?, ?, ?, ?)
Hibernate: call identity()
  -->
```

## Add Product

```
  --> addProduct
Product Name?Axe
Hibernate: select company0_.id as id3_, company0_.name as name3_ from Company company0_
1 Company1
2 Company2
3 Candy
4 Gillette
Which Company ? 4
Hibernate: insert into Product (id, company, name) values (null, ?, ?)
Hibernate: call identity()
  -->
```