

Agile Software Development

Produced
by

Eamonn de Leastar (edeleastar@wit.ie)

Department of Computing, Maths & Physics
Waterford Institute of Technology

<http://www.wit.ie>

<http://elearning.wit.ie>



Waterford Institute of Technology
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRCE



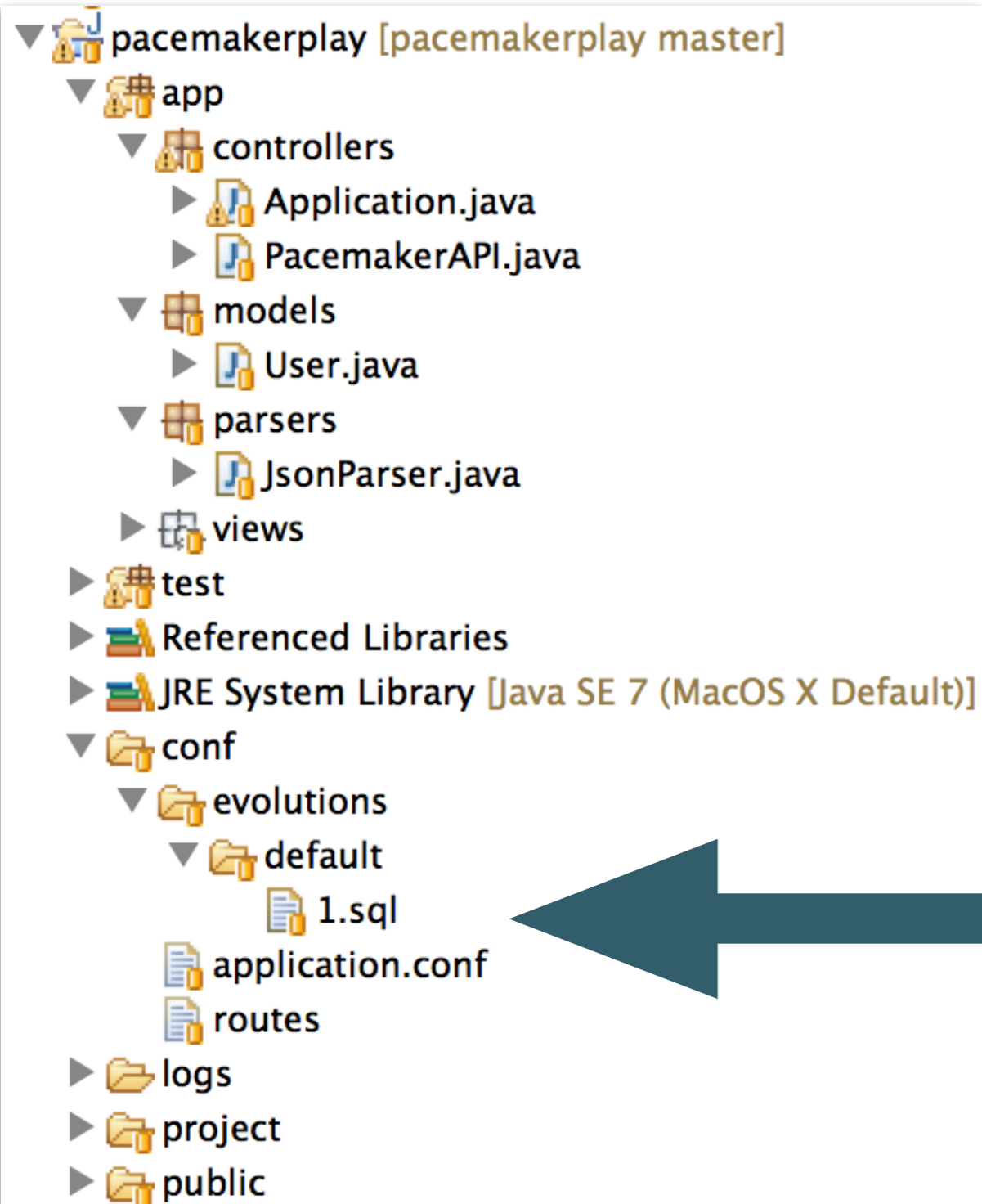
Database / Schema Evolution

Database Evolution

- Database evolution (sometimes called schema evolution) refers to the problem of evolving a database schema to adapt it to a change in the modeled reality.
- The problem is not limited to the modification of the schema, also affecting the data stored under the given schema

“The problem has been recognized as a very pressing one by the database community for more than 12 years ... support for Schema Evolution, is a difficult problem involving complex mapping among schema versions, the tool support has been so far very limited. The recent theoretical advances on mapping composition and mapping invertibility, which represent the core problems underlying the schema evolution remains almost inaccessible to the large public”

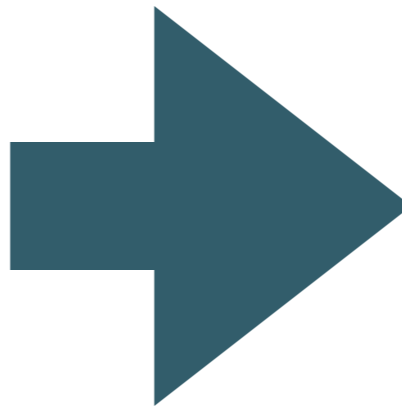
Database Evolution in Play



- Play monitors model classes and generates a new sql script if it detects a change in the model from the pre-existing script
- This script can also be manually updated and maintained

Evolution Script (1)

```
@SuppressWarnings("serial")
@Entity
@Table(name="my_user")
public class User extends Model
{
    @Id
    @GeneratedValue
    public Long id;
    public String firstname;
    public String lastname;
    public String email;
    public String password;
    ...
}
```



```
# --- Created by Ebean DDL
# --- !Ups

create table my_user (
    id                bigint not null,
    firstname         varchar(255),
    lastname          varchar(255),
    email             varchar(255),
    password          varchar(255),
    constraint pk_my_user primary key (id))
;

create sequence my_user_seq;

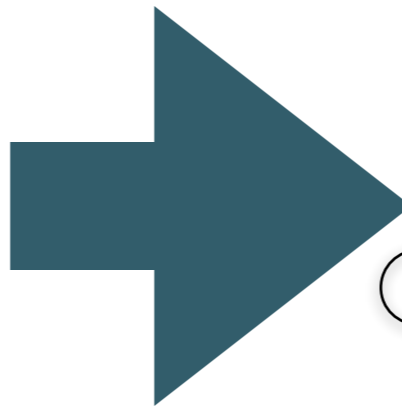
# --- !Downs

drop table if exists my_user cascade;

drop sequence if exists my_user_seq;
```

Evolution Script (2)

```
@SuppressWarnings("serial")
@Entity
@Table(name="my_user")
public class User extends Model
{
    @Id
    @GeneratedValue
    public Long id;
    public String firstname;
    public String lastname;
    public String email;
    public String password;
    public String nationality;
    ...
}
```



```
# --- Created by Ebean DDL

# --- !Ups

create table my_user (
    id                bigint not null,
    firstname         varchar(255),
    lastname          varchar(255),
    email             varchar(255),
    password          varchar(255),
    nationality        varchar(255),
    constraint pk_my_user primary key (id))
;

create sequence my_user_seq;

# --- !Downs

drop table if exists my_user cascade;

drop sequence if exists my_user_seq;
```

More interesting model

```
@Entity
@Table(name="my_user")
public class User extends Model
{
    @Id
    @GeneratedValue
    public Long id;
    public String firstname;
    public String lastname;
    public String email;
    public String password;

    @OneToMany(cascade=CascadeType.ALL)
    public List<Activity> activities = new ArrayList<Activity>();
    //...
}
```

```
@Entity
public class Activity extends Model
{
    @Id
    @GeneratedValue
    public Long id;
    public String type;
    public String location;
    public double distance;
    //...
}
```

application.conf

- Database URL
different for local/
heroku database
- Database Driver also
different!
- This implies the
syntax of the
evolution script may
differ depending on
which driver is loaded

```
#db.default.driver=org.postgresql.Driver
#db.default.url=${DATABASE_URL}

db.default.driver=org.h2.Driver
db.default.url="jdbc:h2:mem:play"
db.default.user=sa
db.default.password=""
```


Different Evolutions!

```
create table activity (  
  id                bigint not null,  
  user_id           bigint not null,  
  type              varchar(255),  
  location          varchar(255),  
  distance          float,  
  constraint pk_activity primary key (id))  
;  
  
create table my_user (  
  id                bigint not null,  
  firstname         varchar(255),  
  lastname         varchar(255),  
  email            varchar(255),  
  password         varchar(255),  
  constraint pk_my_user primary key (id))  
;  
  
create sequence activity_seq;  
create sequence my_user_seq;  
  
alter table activity add constraint fk_activity_my_user_1  
foreign key (user_id) references my_user (id);  
create index ix_activity_my_user_1 on activity (user_id);  
  
drop table if exists activity cascade;  
drop table if exists my_user cascade;  
drop sequence if exists activity_seq;  
drop sequence if exists my_user_seq;
```

Postgres

```
create table activity (  
  id                bigint not null,  
  user_id           bigint not null,  
  type              varchar(255),  
  location          varchar(255),  
  distance          double,  
  constraint pk_activity primary key (id))  
;  
  
create table my_user (  
  id                bigint not null,  
  firstname         varchar(255),  
  lastname          varchar(255),  
  email            varchar(255),  
  password         varchar(255),  
  constraint pk_my_user primary key (id))  
;  
  
create sequence activity_seq;  
create sequence my_user_seq;  
  
alter table activity add constraint fk_activity_my_user_1  
foreign key (user_id) references my_user (id) on delete  
restrict on update restrict;  
create index ix_activity_my_user_1 on activity (user_id);  
  
SET REFERENTIAL_INTEGRITY FALSE;  
drop table if exists activity;  
drop table if exists my_user;  
SET REFERENTIAL_INTEGRITY TRUE;  
drop sequence if exists activity_seq;  
drop sequence if exists my_user_seq;
```

MySql

Switching Drivers

- This will not run locally
- `${DATABASE_URL}` is only valid inside the Heroku environment
- ==> Evolution will not be generated unless:
 - use postgres database locally
 - connect to postgres in heroku

```
db.default.driver=org.postgresql.Driver  
db.default.url=${DATABASE_URL}
```

```
#db.default.driver=org.h2.Driver  
#db.default.url="jdbc:h2:mem:play"  
#db.default.user=sa  
#db.default.password=""
```


Connecting to Postgress on Heroku


Dynos

web target/universal/stage/bin/pacemakerplay -Dhttp.port=\$PORT

1X ▾ 1

Add-ons

 Heroku Postgres Dev :: Gold

 Get Add-ons

limitless-river-5468 :: pink 

Connection Settings : Default 

Host	ec2-54-227-242-149.compute-1.amazonaws.com
Database	d36j7gchn8nuh6
User	cywohlrwnycunw
Port	5432
Password	Show

limitless-river-5468 :: pink



Connection Settings : Default



Host ec2-54-227-242-149.compute-1.amazonaws.com
Database d36j7gchn8nuh6
User cywohlrwnycunw
Port 5432
Password [Show](#)

Connection Settings

Default

URL

PSQL

PG Restore

Active Record

Sequel

Data Mapper

Django

JDBC URL

JDBC Properties

PHP

Statistics

Plan Dev
Status available
Data Size 6.6 MB
Tables 4
PG Version 9.2.5
Created November 06, 2013 12:48

limitless-river-5468 :: pink



Connection Settings : JDBC URL



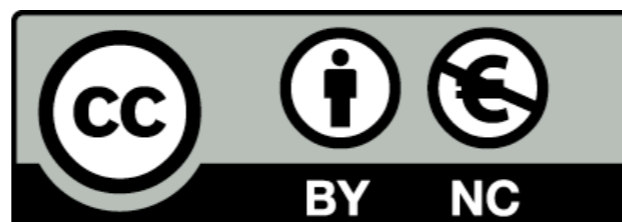
```
String url = "jdbc:postgresql://ec2-54-227-242-149.compute-1.amazonaws.com:5432/d36j7gchn8nuh6?user=cywohlrwnycunw&password=X2-VqariG0cbludR74bVWqRzQK&ssl=true";  
Connection conn = DriverManager.getConnection(url);
```

Statistics

Evolve Locally - Deploy Remotely

- Evolve the database locally
- Commit the generated sql script to git
- push to heroku
- This will trigger remote evolution

```
db.default.driver=org.postgresql.Driver  
db.default.url="jdbc:postgresql://ec2-54-227-242-149.compute-1.amazonaws.com:543....."
```



Except where otherwise noted, this content is licensed under a Creative Commons Attribution-NonCommercial 3.0 License.

For more information, please see <http://creativecommons.org/licenses/by-nc/3.0/>



Waterford Institute of Technology
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRCE

