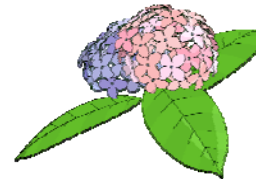


Seasar Conference 2008 Spring



早分かりS2Dao

Seasarプロジェクトコミッタ

(株)エルテックス

大中浩行

- はじめに
- S2Daoとは？
- S2Daoに必要なもの
- S2Daoを動かしてみる
 - 挿入
 - 検索
 - 更新
 - 複雑な検索(2WaySQL)
 - 削除

- 大中浩行
- Seasarプロジェクトコミッタ
- (株)エルテックス SI事業部
- <http://www.fieldnotes.jp/d/>
- Twitter: @setoazusa
- Working Effectivelry With Legacy Code読書会 主催
 - 5/31(土)に第1回を開催します。
 - <http://fieldnotes.sytes.net/~azusa/wiki/?LegacyCode>
 - <http://groups.google.co.jp/group/legacy-code>



S2Daoとは？

■ S2Daoとは？



S2Daoとは？

1. 実装コードを書かずに、
2. SQLを直接書ける
3. O/Rマツパ

- 冗長なJDBCのコードの組み立てから解放されます！
- DaoはJavaのインターフェースだけ(実装クラス不要)

```
@S2Dao(bean=Employee.class)
public interface EmployeeDao{
    public List<Employee> getAllEmployees();
}
```

■ Daoが使うSQLファイルを、ツールから直接実行可能(2-WaySQL)

– **SELECT * FROM EMPLOYEE**

WHERE empno = /*empno*/7369

- ResultSetからJavaBeanへの詰め替えも、
- PreparedStatementへのパラメータ設定も、
- S2Daoがやってくれます。



■ 必要なもの

- Daoのインターフェース
- テーブル、ResultSetに対応するJavaオブジェクト(JavaBean/Entity/Dto)
- SQLファイル
- Seasarの設定
 - Seasar2.4のSMART Deployに従えば設定レス

■動かしてみる

■ Employeeテーブル

カラム名	型	PK
EMPNO	numeric(4)	○
ENAME	varchar(10)	
JOB	varchar(9)	
MGR	numeric(7,2)	
HIREDATE	Date	
SAL	numeric(7,2)	
COMM	numeric(7,2)	
DEPTNO	numeric(2)	
TIMESTAMP	timestamp	



```
@Bean(table = "EMPLOYEE")
public class Employee {
    @Column("EMPNO")
    public Integer empno;
    @Column("EMPNO")
    public String ename;
    @Column("JOB")
    public String job;
    @Column("MGR")
    public Integer mgr;
    @Column("HIREDATE")
    public Date hiredate;
    @Column("SAL")
    public BigDecimal sal;
    @Column("COMM")
    public BigDecimal comm;
    @Column("DEPTNO")
    public Integer deptno;
    @Column("TIMESTAMP")
    public Timestamp timestamp;
    public String toString() {
        return org.apache.commons.lang.builder.ToStringBuilder.reflectionToString(this);
    }
}
```

```
@S2Dao(bean = Employee.class)
public interface EmployeeDao {

}
```

```
public int insert(Employee employee);
```

//全検索

```
public List<Employee> getAllEmployees();
```

//主キーで検索

```
@Query("EMPNO = ?")
```

```
public Employee getEmp(int empno);
```




public int update (Employee employee) ;

```
@Arguments ( { "job", "deptno" } )  
public List<Employee>  
    getEmployeeByJobAndDeptno  
        (String job, Integer deptno);
```

```
SELECT * FROM EMPLOYEE
/*BEGIN*/WHERE
  /*IF "aaa".equals(job)*/
    JOB = /*job*/' CLERK'
/*END*/
/*IF deptno != null*/
  AND DEPTNO =/*deptno*/20
/*END*/
/*END*/
```

- ファイル名は「Daoのクラス名_メソッド名.sql」
 - この場合
EmployeeDao_getEmployeeByJobAndDeptno.sql
 - Eclipseの場合はソースファイルと同じ場所に置く
 - Maven2の場合はsrc/main/resources

```
public int delete(Employee employee);
```

```
//もしくは
```

```
@Query("EMPNO = ?")
```

```
public int deleteByQuery(int empno);
```

- S2Daoは、
 1. 実装コードを書かずに、→Daoはインターフェースのみ
 2. SQLを直接書ける →2WaySQLで自由自在
 3. O/Rマツパ → 詰め替え、設定手間いらず

■ ホームページ

– <http://s2dao.seasar.org/ja/index.html>

■ 質問はSeasar-Userメーリングリストまで

– <https://ml.seasar.org/mailman/listinfo/seasar-user>

■ 開発に関する議論はseasar-s2dao-devメーリングリストまで

– <https://ml.seasar.org/mailman/listinfo/seasar-s2dao-dev>



おわり

ありがとうございました。