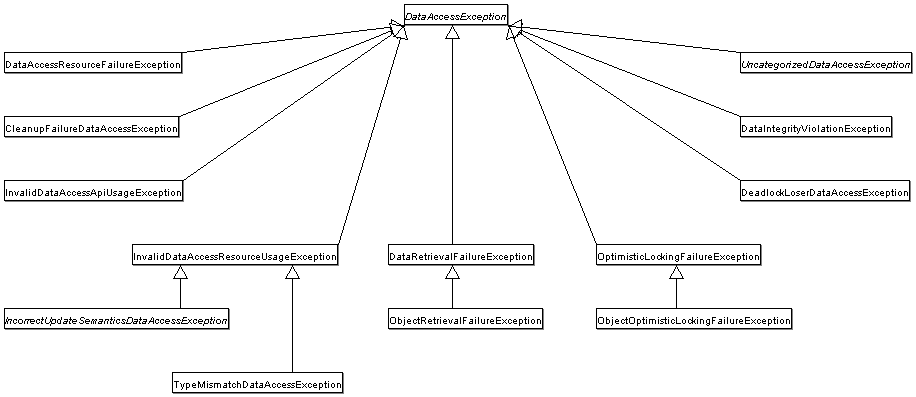
## 13. DAO support

### 13.1 Introduction

DAO层主要目的性就是能够迅速地进行数据访问像JDBC、Hibernate、JPA等。

### 13.2 Consistent exception hierarchy

以DataAccessException为根节点的异常树



## 13.3 Annotations used for configuring DAO or Repository classes

通过@Repository接口我们通过组件查询，能够在不XML标准格式下，找到DAOs和repositories ，JDBC repositories 需要JDBC DataSource，JPA repositories 需要EntityManager，最为容易的方式是使用下面注解@Autowired,, @Inject, @Resource or @PersistenceContext

*@Repository*

**public** **class** JpaMovieFinder **implements** MovieFinder {

*@PersistenceContext*

**private** EntityManager entityManager;

*// ...*

}

如果你用Hibernate api方式的话

*@Repository*

**public** **class** HibernateMovieFinder **implements** MovieFinder {

**private** SessionFactory sessionFactory;

*@Autowired*

**public** **void** setSessionFactory(SessionFactory sessionFactory) {

**this**.sessionFactory = sessionFactory;

}

*// ...*

}

如果你用标准JDBC的方式的话

*@Repository*

**public** **class** JdbcMovieFinder **implements** MovieFinder {

**private** JdbcTemplate jdbcTemplate;

*@Autowired*

**public** **void** init(DataSource dataSource) {

**this**.jdbcTemplate = **new** JdbcTemplate(dataSource);

}

*// ...*

}