# Anmerkungen

Zeitaufwand: 45h

## Konfiguration

Die persistence.xml enthält zwei verschiedene PUs. Die eine bietet die Konfiguration für eine Derby in-memory Datenbank. Diese wird von den Unit-Tests verwendet. Die andere konfiguriert einen PostgreSQL Treiber, der für die Konsolenanwendung hergenommen wird.

## Transaktionen

Das Programm verwendet den Bitronix JTA Manager. Es bietet ein Interface swt6.orm.persistence.PersistenceManager, das von swt6.orm.persistence.BitronixManager implementiert wird.

Über diesen Manager wird der EntityManager für die DAO Implementierungen im Paket swt6.orm.dao.bitronix, sowie alle Transaktionen in der Konsolenapplikation verwaltet.

## Testdaten

Für die Testdatengenerierung verwendet mein Programm DBSetup. Dieses muss über dbsetup.properties/dbsetup.test.properties eingestellt werden.

## Domainenmodell

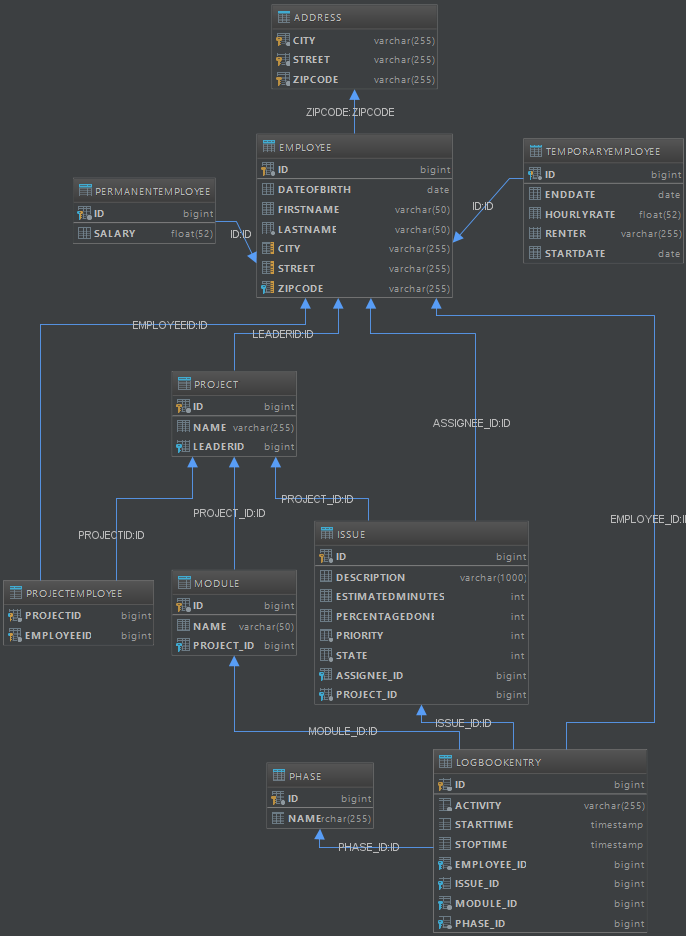
Um alle geforderten Queries vollständig richtig ausführen zu können, musste ich das Domainenmodell etwas verändern. Es wurde eine 1:N Beziehung zwischen LogbookEntry und Issue eingeführt. Mit dieser Beziehung wird jeder Arbeitszeiteintrag nicht nur einem konkreten Projekt, sondern auch einem konkreten Issue zugeordnet. Dadurch kann berechnet werden, wieviel ein Mitarbeiter an einem konkreten Issue gearbeitet hat. Ohne diese Beziehung könnte nur berechnet werden, wieviel Arbeit ein Arbeiter in ein konkretes Projekt gesteckt hat.   
Um inkonsistente Datensätze bestmöglichst zu vermeiden, könnte man es noch etwas anders modellieren: Anstatt der Beziehung von Logbookentry zu Module, könnte Issue (N:1) einem Module zugeordnet werden, und LogbookEntry (N:1) einem Issue. So ist der LogbookEntry über den Issue immer noch einem konkreten Modul zugeordnet, und damit auch gleichzeitig nur einem Projekt und einem Issue. Um die Angabe nicht zu weit zu verbiegen, habe ich jedoch die Variante oben implementiert.

## DAOs

Es wurden mehrere (nicht-triviale) Abfragen mit der Criteria-API implementiert, unter anderem im ProjectDao und LogbookEntryDao.

Alle DAOs bieten eine generische Abfrage swt6.orm.dao.BaseDao#findByPredicate. Mit einem Lambda Ausdruck kann so jeder beliebige Filter übergeben werden.

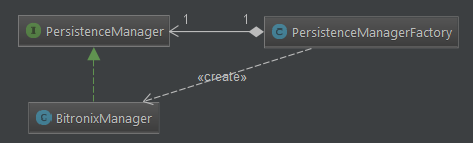
# Datenbank Aufbau



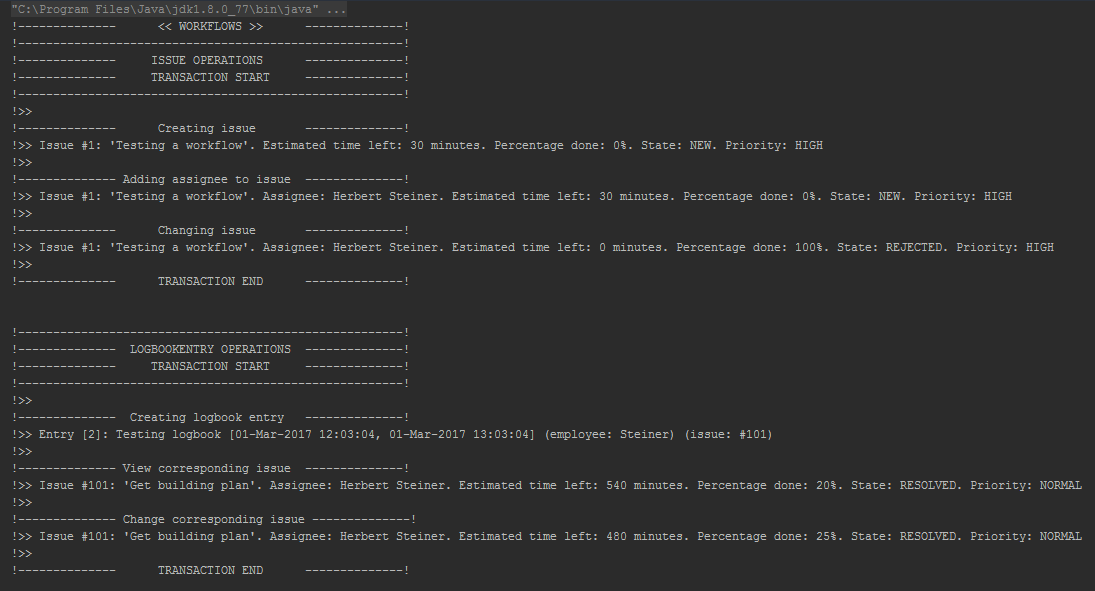
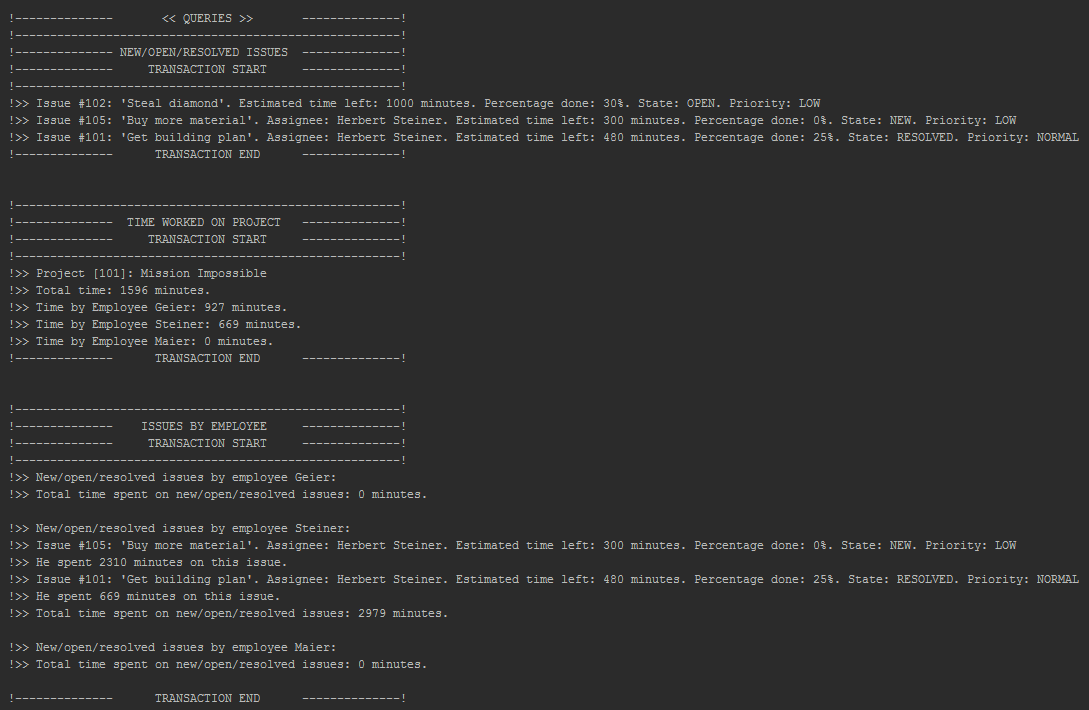
# DAO Klassendiagramm

# Domain-Klassen

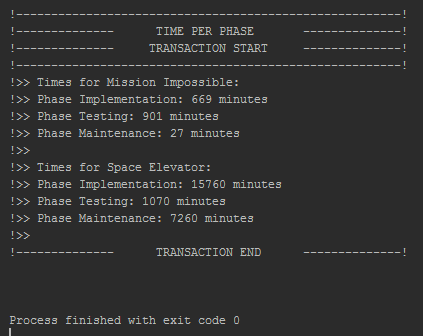
# Persistenz-Manager



# WorkLog Manager (Console App) - Testdurchlauf



cont…



# Tests

Für alle Tests swt6.orm.dao.test.DaoTestSuite ausführen.



