Eidgenössische
Technische Hochschule
Zürich

Ecole polytechnique fédérale de Zurich
Politecnico federale di Zurigo
Swiss Federal Institute of Technology Zurich

Institut für Informationssysteme, Dr. Markus Tresch

Uwe Röhm

Verteilte Objektverwaltung: Client, Server & Middleware
Lösungsblatt 2

Aufgabe 1: SQL Middleware

Teilaufgabe 1:

```
SELECT au_lname, au_fname
FROM   authors, titleauthor, titles
WHERE  titles.title    = 'Secrets of Silicon Valley'
  AND  titles.title_id = titleauthor.title_id
  AND  authors.au_id   = titleauthor.au_id

SELECT city
FROM   titles, stores, salesdetail
WHERE  titles.title        = 'Secrets of Silicon Valley'
  AND  titles.title_id     = salesdetail.title_id
  AND  salesdetail.stor_id = stores.stor_id

SELECT ename
FROM   scott.emp, scott.dept
WHERE  dept.dname  = 'RESEARCH'
  AND  dept.deptno = emp.deptno;

SELECT ename
FROM   scott.emp, scott.salgrade
WHERE  grade = '4'
  AND  losal < sal
  AND  hisal > sal;
```

Anfragen über beiden Schemata, wie die letzte Beispielanfrage, sind noch nicht beantwortenbar.

Teilaufgabe 2:

Siehe oben, jeweils mit `connect to oracle` oder `connect to sybase` ganz zu Anfang.

Teilaufgabe 3:

```
sp_addtabledef employees0,   "oracle..scott.emp"
sp_addtabledef departments0, "oracle..scott.dept"
sp_addtabledef salgrade0,    "oracle..scott.salgrade"

CREATE EXISTING TABLE employees0
(
   empno    int,
   ename    varchar(10) null,
   job      varchar(9)  null,
   mgr      int         null,
   hiredate datetime    null,
   sal      int         null,
   comm     int         null,
   deptno   int         null
)

CREATE EXISTING TABLE departments0
(
   deptno int         null,
   dname  varchar(14) null,
   loc    varchar(13) null
)

CREATE EXISTING TABLE salgrade0
(
   grade int null,
   losal int null,
   hisal int null
)
```

```
sp_addtabledef authors0,     "sybase...authors"
sp_addtabledef titles0,      "sybase...titles"
sp_addtabledef titleauthor0, "sybase...titleauthor"
sp_addtabledef stores0,      "sybase...stores"
sp_addtabledef sales0,       "sybase...sales"
sp_addtabledef salesdetail0, "sybase...salesdetail"
sp_addtabledef publisher0,   "sybase...publisher"
```

```
CREATE EXISTING TABLE authors0
(
   au_id      id,
   au_lname   varchar(40),
   au_fname   varchar(20),
   phone      char(12),
   address    varchar(12) null,
   city       varchar(20) null,
   state      varchar(2)  null,
   country    varchar(12) null,
   postalcode char(10)    null
)
```

```
CREATE EXISTING TABLE stores0
(
   stor_id      char(4),
   stor_name    varchar(40),
   stor_address varchar(40),
   city         varchar(20) null,
   state        char(2)     null,
   country      varchar(12) null,
   postalcode   char(10)    null,
   payterms     varchar(12) null
)
```

```
CREATE EXISTING TABLE titles0
(  au_id       id,
   title       varchar(80),
   type        char(12),
   pub_id      char(4)      null,
   price       money        null,
   advance     money        null,
   total_sales int          null,
   notes       varchar(200) null,
   pubdate     datetime,
   contract    bit
)
```

```
CREATE EXISTING TABLE sales0
(
   stor_id char(4),
   ord_num varchar(20),
   date    datetime
)
```

```
CREATE EXISTING TABLE titleauthor0
(
   au_id      id,
   title_id   tid,
   au_ord     tinyint null,
   royaltyper int     null
)
```

```
CREATE EXISTING TABLE salesdetail0
(
   stor_id  char(4),
   ord_num  varchar(20),
   title_id tid,
   qty      smallint,
   discount float
)
```

```
CREATE EXISTING TABLE publisher0
(
   pub_id   char(4),
   pub_name varchar(40),
   city     varchar(20),
   state    char(2)
)
```

Tip: Die Tabellen in Syabse können mit der Stored-Procedure `sp_help` *`tabellenname`*, die in Oracle mittels `describe` *`tabellenname`* abgefragt werden.

Teilaufgabe 4:

Die letzte Anfrage kann nun wie folgt gestellt werden:

```
SELECT au_lname
FROM   authors0, employees0, departments0
WHERE  dname = 'RESEARCH'
  AND  departments0.deptno = employees0.deptno
  AND  ename = au_lname
```

Teilaufgabe 5:

Das Hauptproblem ist, dass die Schemata der Angestelltendatenbank in Oracle und der Autorenwelt in Sybase sehr verschieden sind und nur wenige Berührungspunkte haben. So ist zum Beispiel die Angestelltennummer in Oracle („empno“) nicht mit der Autoren-ID in Sybase („au_id“) vergleichbar. Vielmehr muß man sich in der letzten Anfrage auf die Gleichheit der Namen verlassen.

Das Übernehmen eines Autoren als Angestellten birgt die gleichen Probleme. Man muss fast alle Attribute eines Angestellten erfinden und kann auch die Autoren-ID nicht als Angestelltennummer weiterverwenden. Es fehlt jeglicher Mechanismus zur Kontrolle der globalen Konsistenz. Jene könnte man immerhin nachprogrammieren...

Eine Möglichkeit ist noch das Anlegen einer zusätzlichen Tabelle zum Verwalten der Beziehungen zwischen Datensätzen beider Datenbanken. Jene könnte beispielsweise einer Autoren-ID eine äquivalente Personalnummer zuordnen.