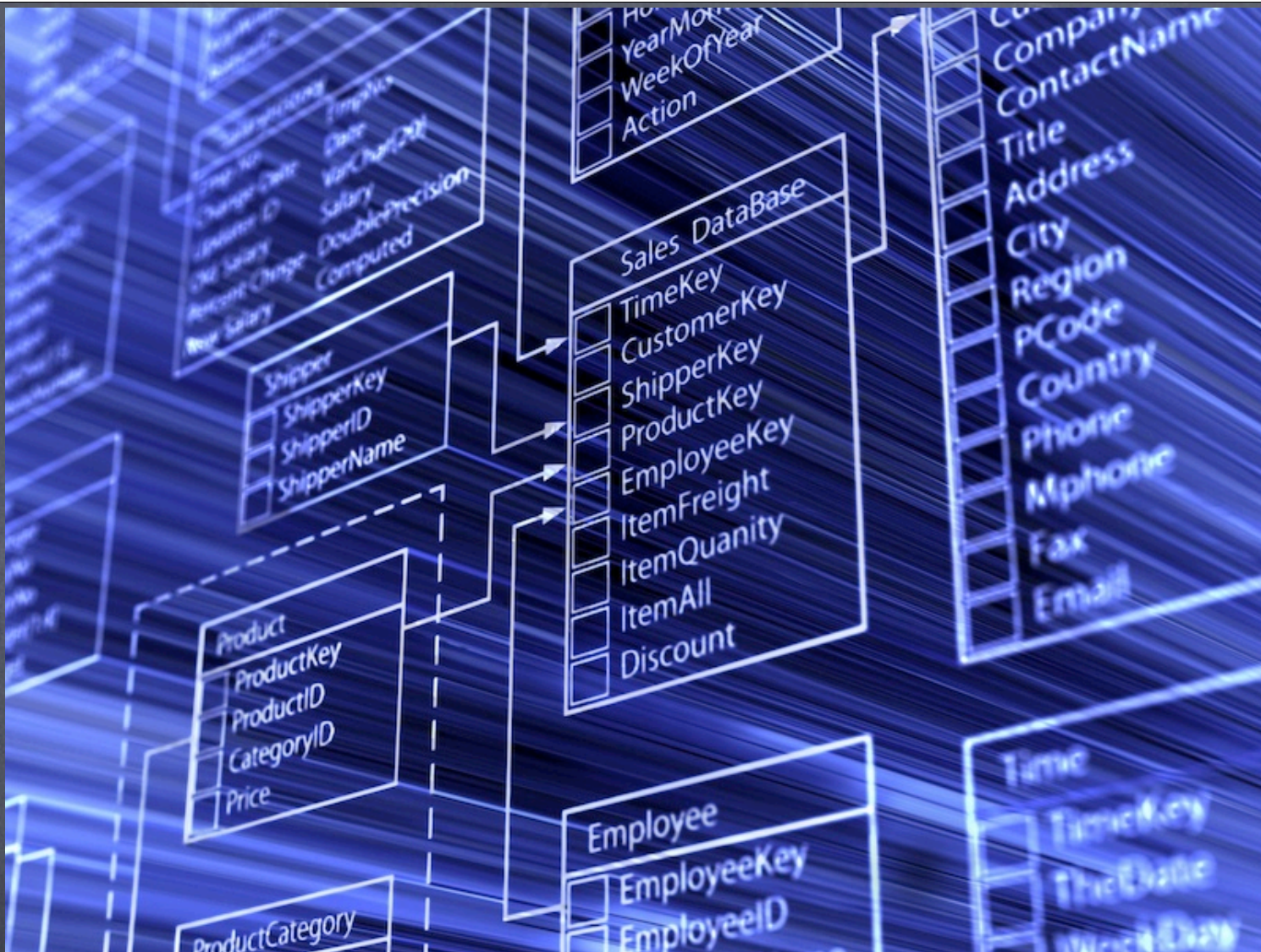




Databases

Practical Introduction to Programming for Scientists – Lecture 13

Databases

- Simple (embedded) Database
 - dbm, BerkeleyDB (bsddb), SQLite*
- Relational Database
 - MySQL, Oracle, DB2, SQLite*
- Object Oriented Database (OODB)
 - Zope, Databeans
- XML Database
 - Sedna, Oracle, BerkeleyDB-XML

Simple/Embedded Database

- Key/Value pairs
- Python
 - Shelve — Persistent dictionary
 - anydbm — Generic access to DBM-style databases
 - whichdb — Guess which DBM module created a database
 - dbm — Simple “database” interface
 - gdbm — GNU’s reinterpretation of dbm
 - dbhash — DBM-style interface to the BSD database library
 - bsddb — Interface to Berkeley DB library
 - dumbdbm — Portable DBM implementation

Relational Database

Tables (Relations)

First Name	Last Name	Program	Grade
John	Barnes	1	A
Frank	Smith	1	B+
Carol	Franklin	2	A-
Steve	Black	3	C+
Charles	Baker	1	B+

For many→many we use a 'junction table'.

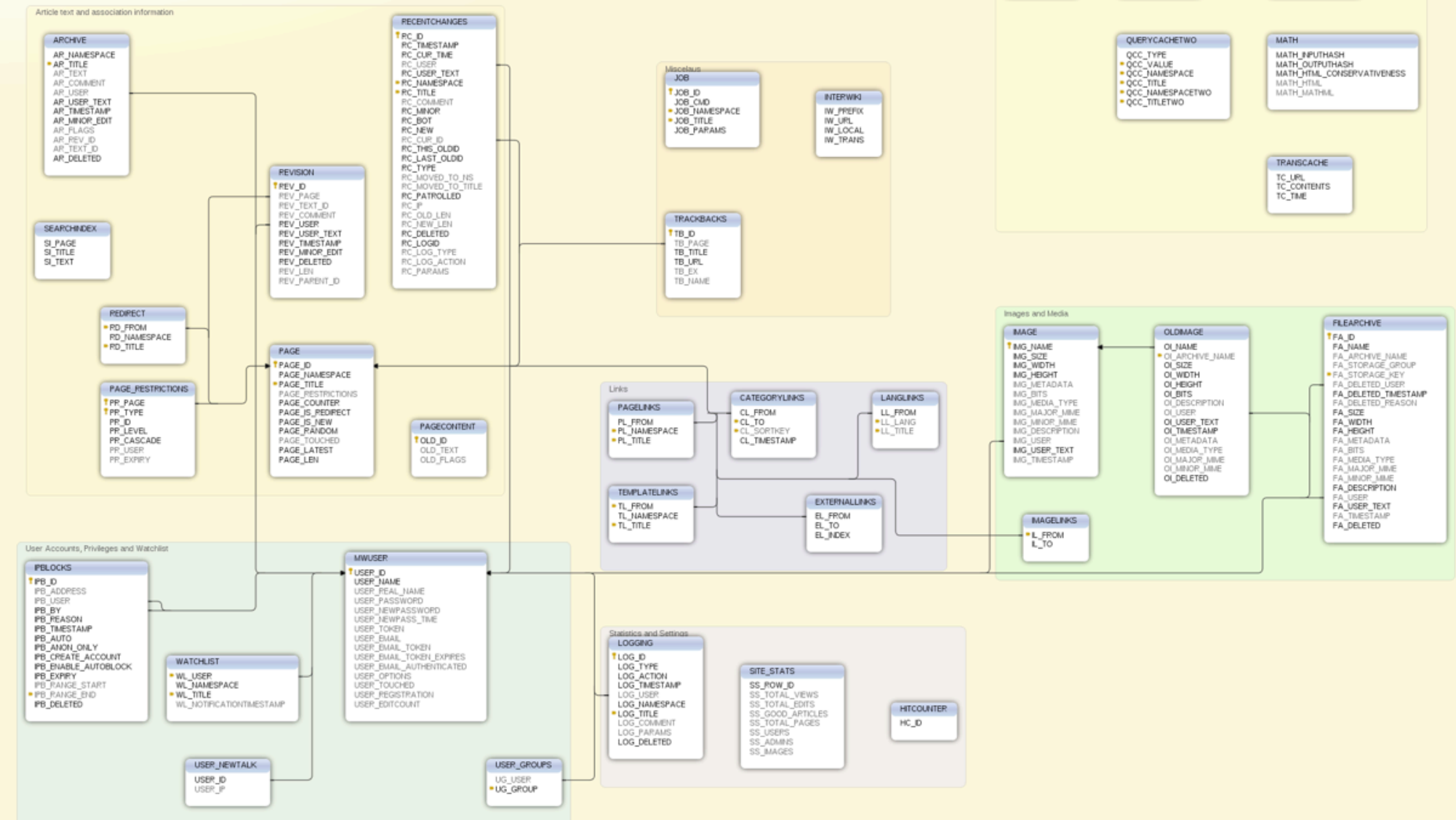
1 to 1



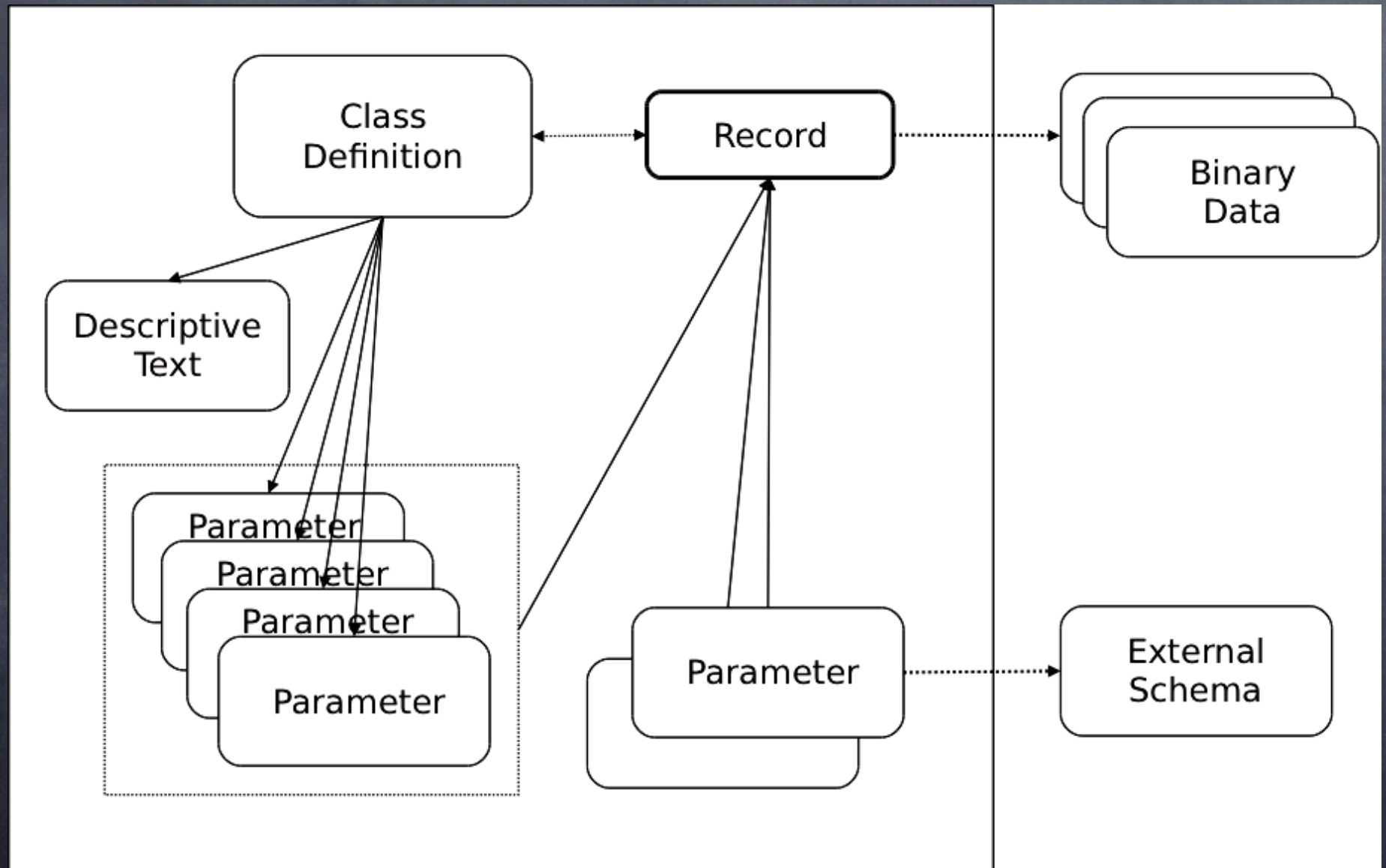
id	Name	Institution
1	SCBMB	BCM
2	Biochemistry	BCM
3	Biochemistry	Rice

Database Schema

MediaWiki database schema
Move the mouse cursor over the table header
and columns to view the comments
generated using DbSchema



OODB



ACID

- Atomicity
 - Transactions are 'all or none', no partial modifications
- Consistency
 - Contents of the database are always valid and self-consistent
- Isolation
 - Readers will not see partially completed transactions of writers
- Durability
 - Completed transactions can survive any possible system crash

SQLite3

- `import sqlite3`
- `conn=sqlite.connect(path)` # Create a database 'connection'
- `curs=conn.cursor()` # Create a cursor object
- `curs.execute(SQLstring)` # Execute a query
- `conn.commit()` # After making any changes
- `curs.close()` # close the cursor when you're done

SQLite3

Python type	SQLite type
<u>None</u>	NULL
<u>int</u>	INTEGER
<u>long</u>	INTEGER
<u>float</u>	REAL
<u>str</u> (UTF8-encoded)	TEXT
<u>unicode</u>	TEXT
<u>buffer</u>	BLOB

SQLite3

- create table
- insert
- select
- update
- drop