Today

- Object-Relational Mapping
- Data Mapper
- Active Record
- Prevalence
Persistence

Application
Data Access Layer

Persistent Storage

Ashwin J Mathew     Spring 2008
School of Information, UC Berkeley
Object-Relational Mapping

- Class Hierarchies
- Reference Graphs
- Instances

- Relational Tables
- Rows
ORM Inheritance Strategies

- **Single Table Inheritance**

  Represents an inheritance hierarchy of classes as a single table that has columns for all the fields of the various classes.

- **Class Table Inheritance**

  Represents an inheritance hierarchy of classes with one table for each class.

- **Concrete Table Inheritance**

  Represents an inheritance hierarchy of classes with one table per concrete class in the hierarchy.

  Martin Fowler, Patterns of Enterprise Application Architecture
Active Record

An object that wraps a row in a database table or view, encapsulates the database access, and adds domain logic on that data.

Martin Fowler
Patterns of Enterprise Application Architecture
Active Record in RoR

- Single table inheritance
- DB schema driven by RoR conventions
- Rapid prototyping and development
- Representations in DB schema and Data Access Layer closely tied
- Incapable of mapping to legacy databases
- Limited handling for cascading actions
Data Mapper

A layer of Mappers that moves data between objects and a database while keeping them independent of each other and the mapper itself.

Martin Fowler
Patterns of Enterprise Application Architecture

Ashwin J Mathew         Spring 2008
School of Information, UC Berkeley
Data Mapper in Hibernate

- Map to arbitrary DB schema
- Needs extra mapping metadata
- DB schema defined independent of application
- More time required to prototype and develop
- Flexible handling for cascading actions
ORM and Performance

- Object-Relational Mapping speeds development times
- Any form of Object-Relational Mapping carries runtime performance overheads
- Want speed? Use SQL and raw record sets
Using Prevalence

- Simple – direct persistence of object model
- High access speeds - in-memory!
- High degree of graph traversal
- Constraints
  - Memory consumption
  - Concurrency
  - Indexed search