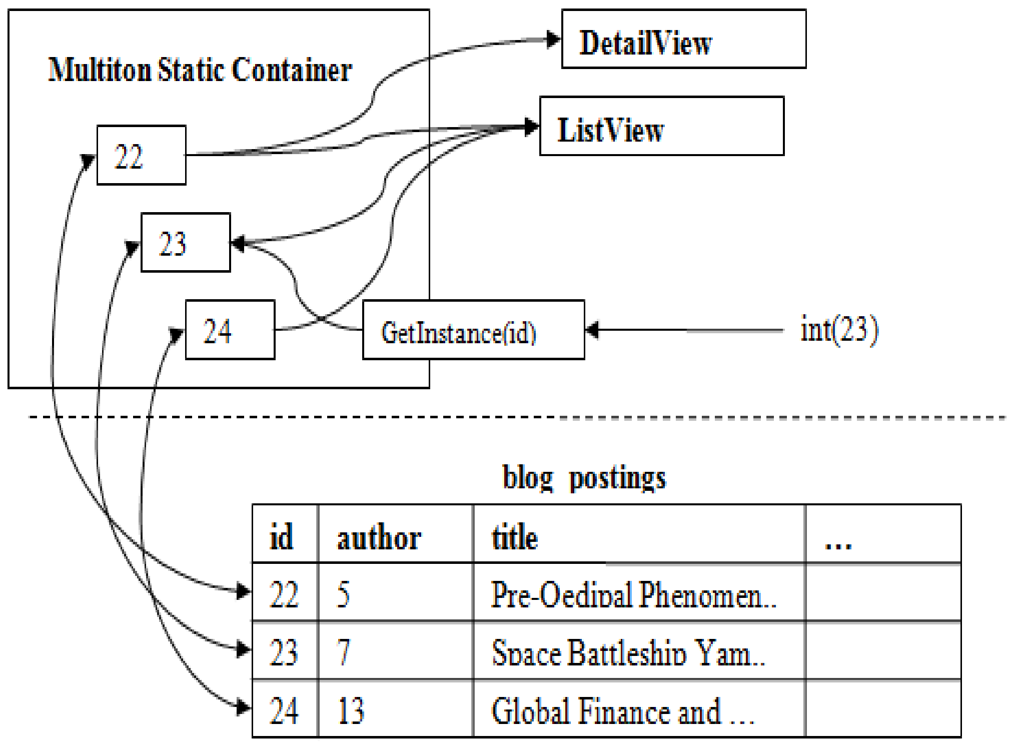
ORM

Object Relational Mapping

Omar El-Essiely

Object Relational Mapping Patterns

- Type System and Meta Mapping
- Data Source Architectural Patterns
- Object-Relational Structural Patterns
- Mapping Relationships



id	author	title	...
22	5	Pre-Oedipal Phenomen..	
23	7	Space Battleship Yam..	
24	13	Global Finance and ...	

Type System & Meta Mapping

Type System Mapping

Database type	Programming Type
NVarchar	String
Date	DateTime
Time	DateTime
Int	Int
bit	Boolean
Decimal	Decimal
Float	Float
Double	Double

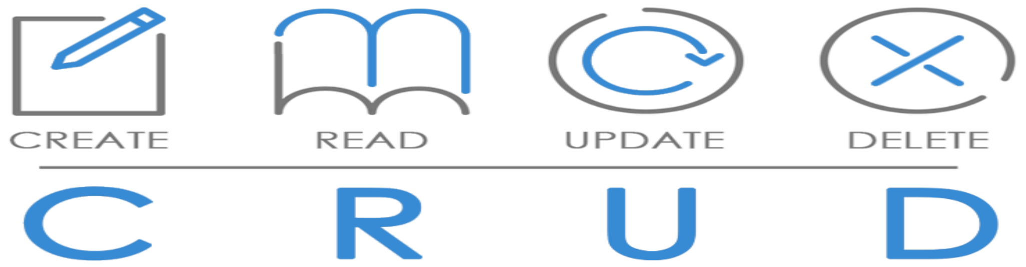
Meta Mapping

- Most ORMs use a meta data conversion look up table to convert between Relational Scheme and OOP
- Conversion table includes:
  - Types Mapping
  - Naming Convention
- The Result will be a mapping table between types

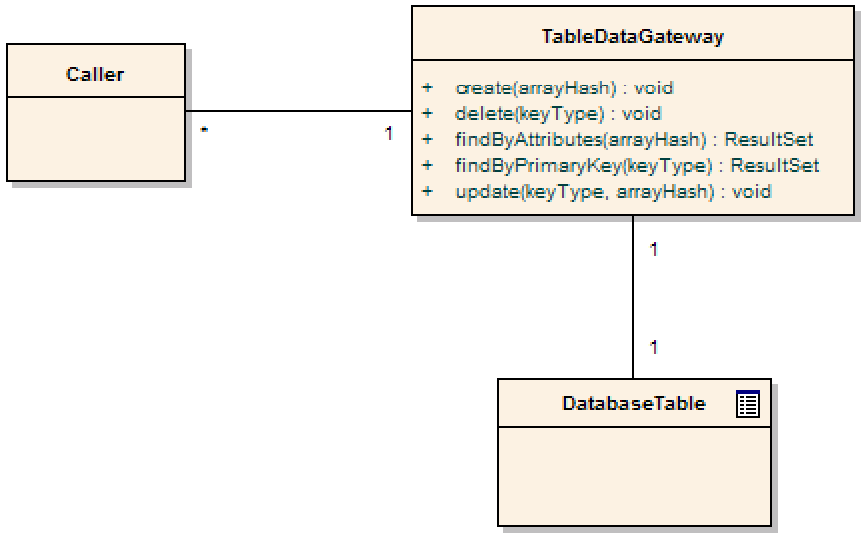
Property	Column
Order.orderID	Order.OrderID
Order.dateOrdered	Order.DateOrdered
Order.dateFulfilled	Order.DateFulfilled
Order.getTotalTax()	Order.Tax
Order.subtotalBeforeTax	Order.SubtotalBeforeTax
Order.shipTo.personID	Order.ShipToContactID
Order.billTo.personID	Order.BillToContactID
Order.lastUpdate	Order.LastUpdate

Data Source Architectural Patterns

Table Gateway



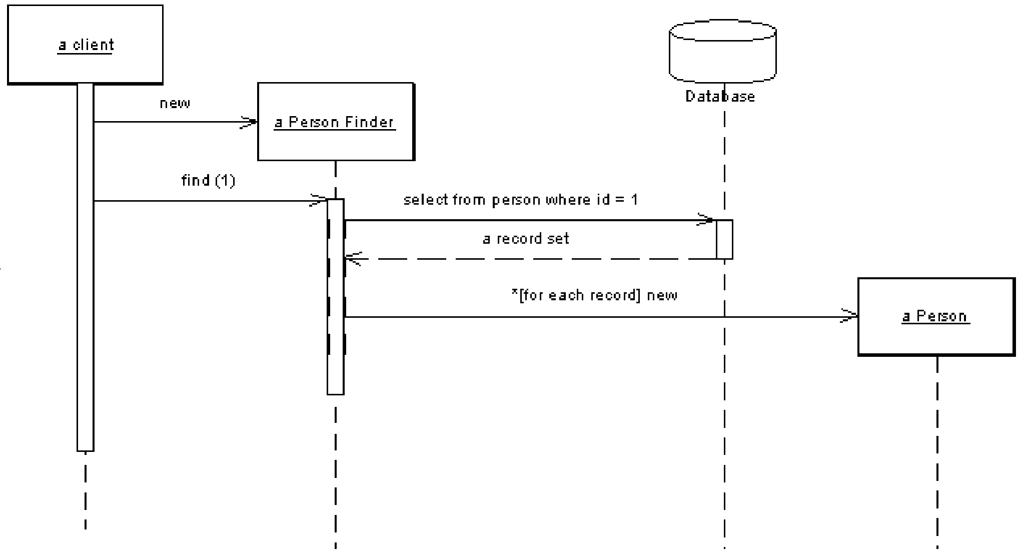
- A Table Data Gateway holds all the SQL for accessing a single table or view: selects, inserts, updates, and deletes. Other code calls its methods for all interaction with the database.
- Table Gateway deals with a predefined …

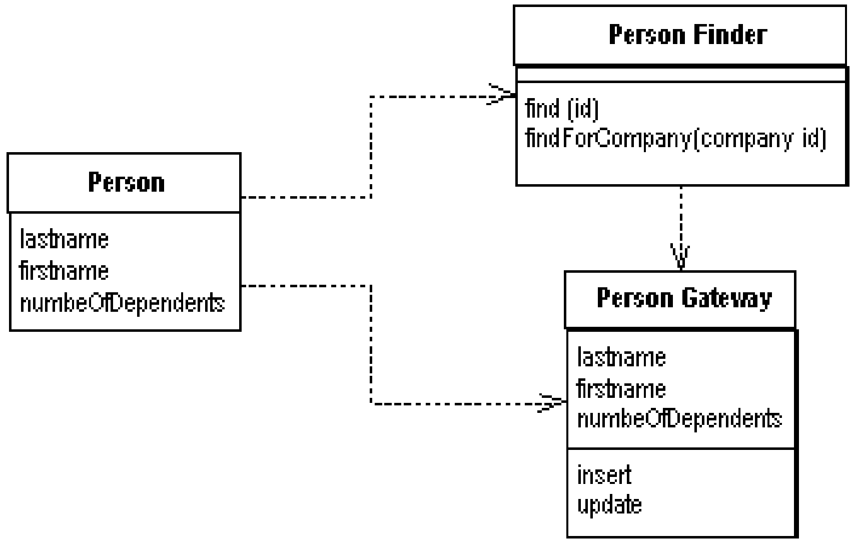

Caller
TableDataGateway
+ create(arrayHash) : void
+ delete(keyType) : void
+ findByAttributes(arrayHash) : ResultSet
+ findByPrimaryKey(keyType) : ResultSet
+ update(keyType, arrayHash) : void
*
1
1
1
DatabaseTable

Client Is aware of the internal implementation details of the object and no object mapping exists.

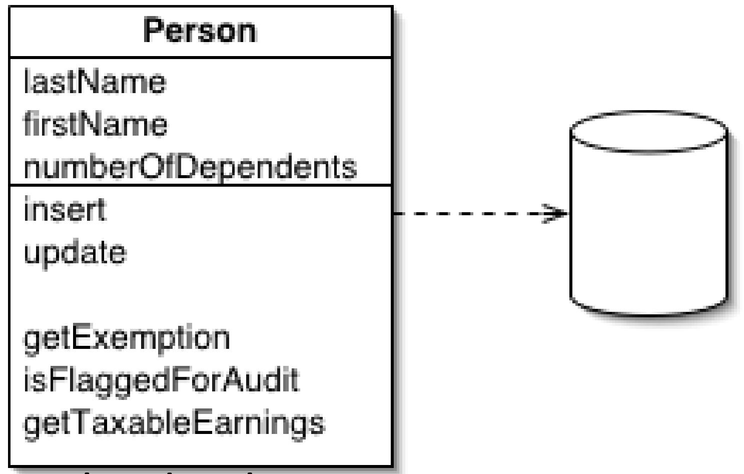
Row Data Gateway

- A Row Data Gateway gives you objects that look exactly like the record in your record structure but can be accessed with the regular mechanisms of your programming language.
- All details of data source access are hidden behind this interface.
- It is totally separated:
    - The access code
    - The data manipulation code
    - The presentation entities



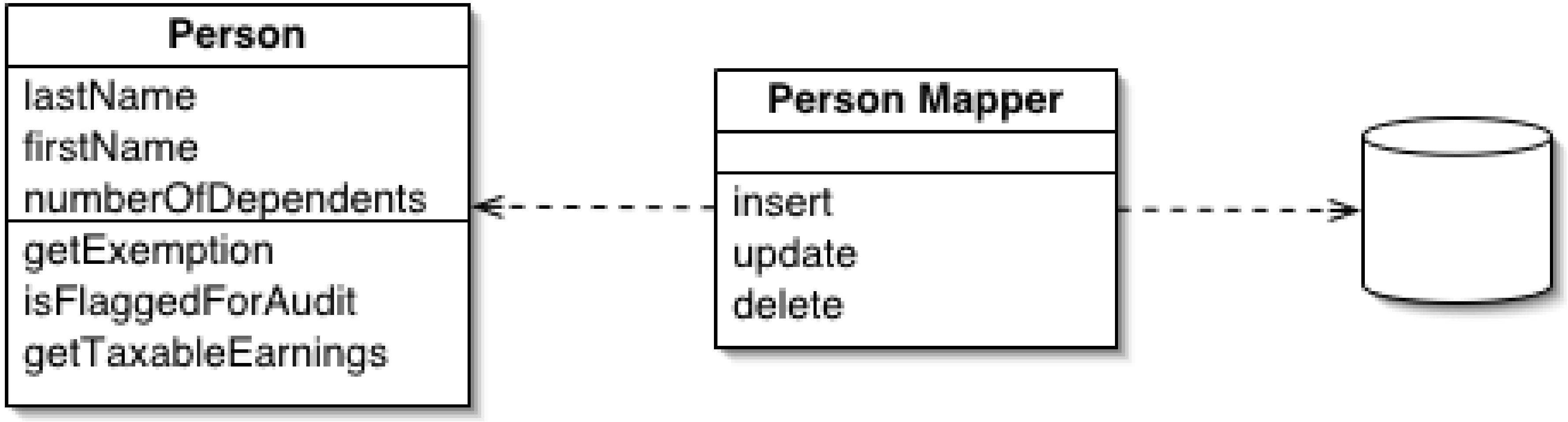


Active Record Pattern



- An object carries both data and access logic.
- This way all people know how to read and write their data to and from the database.
- The Active Record class typically has methods that do the following:
  - Construct an instance of the Active Record from a SQL result set row
  - Construct a new instance for later insertion into the table
  - Static finder methods to wrap commonly used SQL queries and return Active Record objects
  - Update the database and insert into it the data in the Active Record
  - Get and set the fields
  - Implement some pieces of business logic
- Active Record Pattern is very suitable for very simple Business logic that only contains CRUD Operations.

Data Mapper Pattern



Data Mapper Pattern is very suitable for cases where the Database and the domain entities evolve separately.

Identity Field

Saves a database key field in an object to maintain identity between an in-memory object and a database row.

There Are Two Main Methods to Represent an Identity Field:

- Metadata (Configuration or Attribute based)
- Same Name (A Key should always name as ID)

Creating Keys

- Database Created
  - The Database creates the key and usually sets an auto increment identifier.
- GUID
  - (Globally Unique Identifier) is a number generated on one machine that's guaranteed to be unique across all machines in space and time.
- Business Defined Key
  - A Key is known to be unique e.g. SSN.

Mapping Relationships

Mapping Relationships

OOP doesn’t have the concept of Foreign Keys instead it is created as associations.

Relationships have two points of view:

First Category (Cardinality):

- One to One Relationship: Object Reference – Object Reference
- One to Many Relationship: Object Reference – Collection
- Many to Many Relationship: Collection – Collection

Second Category (Direction):

- Uni-directional Relationship: A root object references the other one
- Bi-directional Relationship: Both Objects reference each other

Any questions?