

# Data Mapper Object

- Data mappers map data from a Data Access Object to Data Models
- Data Access Object
  - not covered too much in this talk
  - provides access to a data source like a database, salesforce, as400, REST, via a predictable interface
- Concerned with retrieving or persisting a single Data Model (or collection)

# Data Mapper Object (con't)

- Not particularly smart
  - will work to retrieve data based on parameters
  - will populate a Data Model generally without validation
    - \*\* caveat: sometimes a hybrid is created when working with remote sources to introduce Domain Logic Models here... but that's another talk
- Works directly with the Data Model
  - Creates new instances
  - Understands properties
  - Determines what to persist

# Data Mapper Object (con't)

- Harddrive example
  - Can take a Harddrive Data Model and save it in the proper format to MySQL
    - Remember, the object has the data, but perhaps the database is normalized
  - Has a method to find a harddrive by serial number
    - Internally, determines what db field is the serial number, runs the query, and then populates the model with the full properties
  - Build collection
    - findAll() - parameter where size is > 250gb

# Why is Data Mapper Important?

- Data Models are dumb
  - Don't care what their data is or where it came from
- Domain Logic Models only care about decisions
  - They don't care about the source
- Something needs to retrieve and persist
- Data Mapper is still not your "public" interface
  - (that's a service) - can swap out Data Mappers

# Summarize Data Mapper Model

- Works directly with Data Model and Data Access Objects
- Retrieves and Persists Data Models
- Understands properties of Data Models
  - Can map data source to properties
  - Can determine retrieval method based on properties