

Working with Data

Dan Wahlin



Agenda

- **Data Technology Overview**
- **Creating Model Classes**
- **Creating a DbContext Class**
- **Database Structure**
- **Creating Data Repository Classes**
- **Retrieving and Storing Financial Data**



Agenda

- **Data Technology Overview**
- **Creating Model Classes**
- **Creating a DbContext Class**
- **Database Structure**
- **Creating Data Repository Classes**
- **Retrieving and Storing Financial Data**

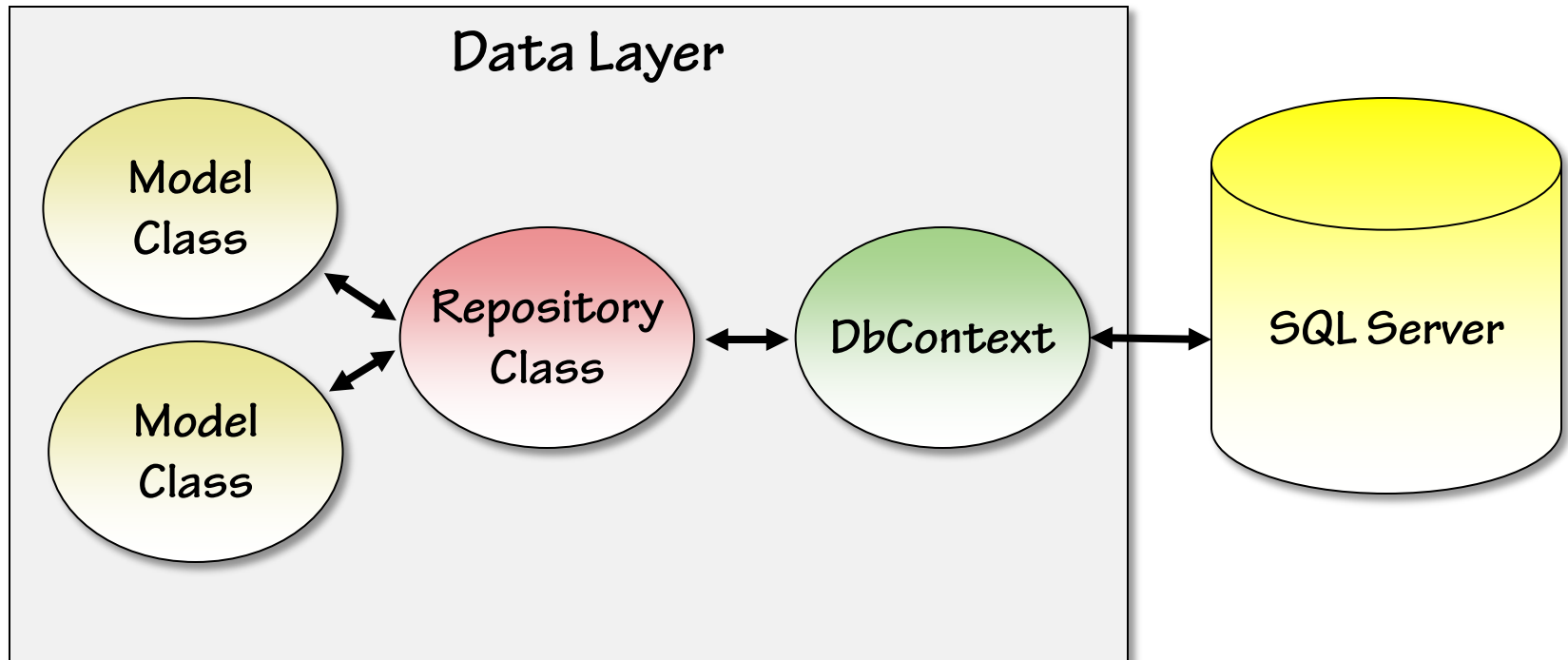


Data Technology Overview

- Financial data retrieved from Google REST service
- Data processed and stored in SQL Server
- Data operations performed using Entity Framework Code First
 - Model classes used to hold account/financial data
 - Repository pattern used for data retrieval classes



Model and Repository Classes



Agenda

- Data Technology Overview
- **Creating Model Classes**
- Creating a DbContext Class
- Database Structure
- Creating Data Repository Classes
- Retrieving and Storing Financial Data



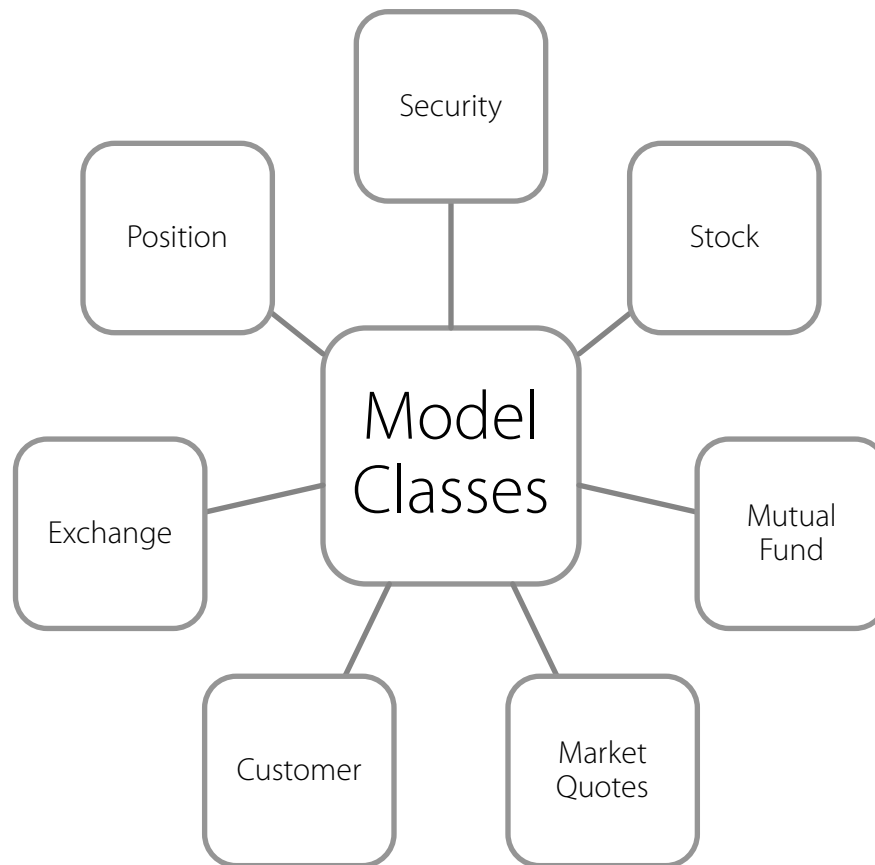
Purpose of Model Classes

- **Model class usage:**
 - Act as data containers from data coming from a data store or being sent to a data store
 - Can be used to generate table structures in a relational database



Model Classes

- Plain Old CLR Objects (POCO) used to hold data



Creating Model Classes

- Defining an abstract Security model class:

```
public abstract class Security
{
    public int Id { get; set; }
    public decimal Change { get; set; }
    public decimal PercentChange { get; set; }
    public decimal Last { get; set; }
    public decimal Shares { get; set; }
    public string Symbol { get; set; }
    public System.DateTime RetrievalDateTime { get; set; }
    public string Company { get; set; }
}
```



Creating Model Classes (cont..)

- Creating a Stock class that derives from Security:

```
public class Stock : Security
{
    public decimal DayHigh { get; set; }
    public decimal DayLow { get; set; }
    public decimal Dividend { get; set; }
    public decimal Open { get; set; }
    public decimal Volume { get; set; }
    //Code removed for brevity
}
```



Agenda

- Data Technology Overview
- Creating Model Classes
- **Creating a DbContext Class**
- Generating a Database
- Creating Data Repository Classes
- Retrieving and Storing Financial Data



Creating a DbContext Class

- Database interactions are performed by a DbContext class
- Defines types to work with using DbSet<T>:

```
public DbSet<Exchange> Exchanges { get; set; }
```

- Optionally defines relationships, table mappings, and more



AccountAtAGlance Class

```
public class AccountAtAGlance : DbContext
{
    public DbSet<BrokerageAccount> BrokerageAccounts { get; set; }
    public DbSet<Customer> Customers { get; set; }
    public DbSet<Exchange> Exchanges { get; set; }
    public DbSet<MarketIndex> MarketIndexes { get; set; }
    public DbSet<Order> Orders { get; set; }
}
```



Agenda

- Data Technology Overview
- Creating Model Classes
- Creating a DbContext Class
- **Generating a Database**
- Creating Data Repository Classes
- Retrieving and Storing Financial Data

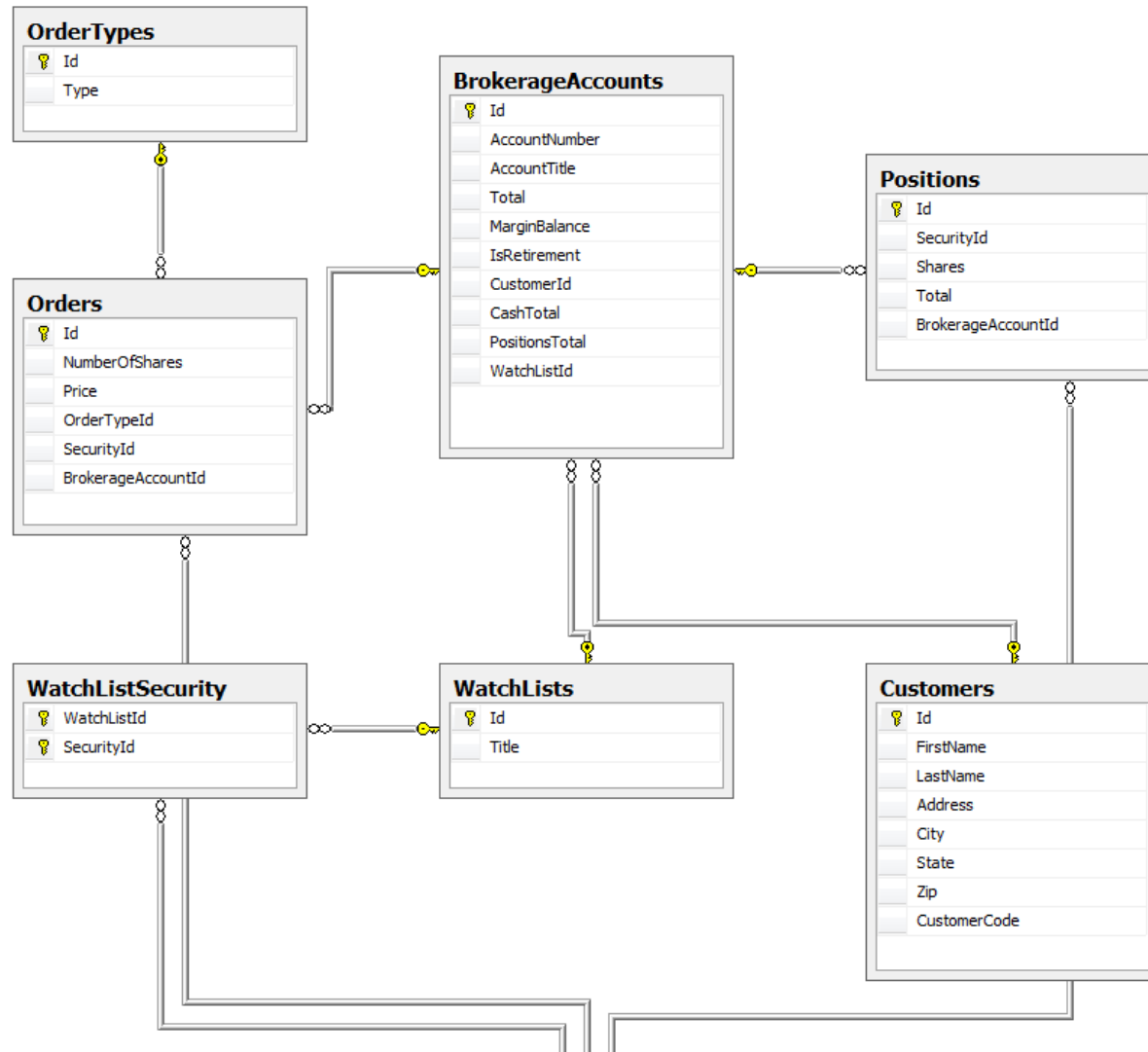


Account at a Glance Database

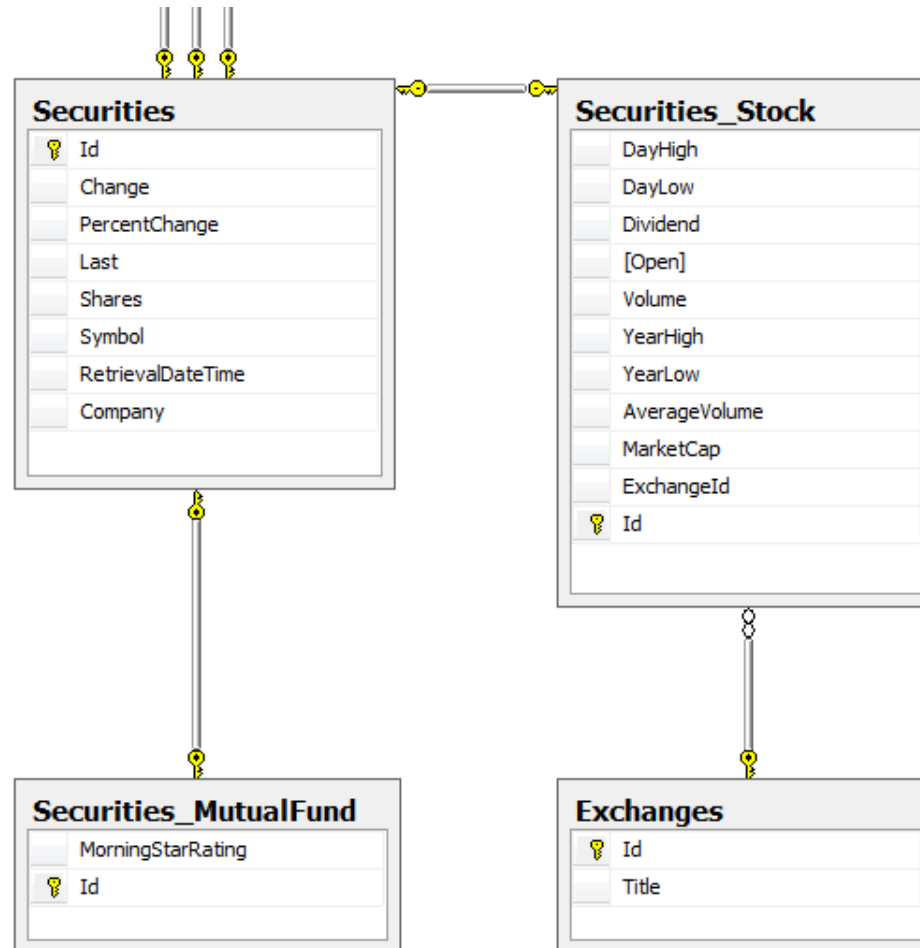
- Application relies on SQL Server 2008
- Data/Model layers originally created using *Entity Framework Model First* with POCO classes
- Database generated from the Model classes
- Model/Data layers converted to *Entity Framework Code First*



Database Diagram



Database Diagram (cont..)



Agenda

- Data Technology Overview
- Creating Model Classes
- Creating a DbContext Class
- Database Structure
- **Creating Data Repository Classes**
- Retrieving and Storing Financial Data



Repository Pattern

- Repository classes created to handle CRUD operations
- Classes follow the Repository Pattern

```
public class SecurityRepository :  
    RepositoryBase<AccountAtAGlance>, ISecurityRepository  
{  
    public Security GetSecurity(string symbol) {...}  
    public List<TickerQuote> GetSecurityTickerQuotes() {...}  
    public OperationStatus UpdateSecurities() {...}  
    public OperationStatus InsertSecurityData() {...}  
}
```



Repository Class Example

```
public class AccountRepository :  
    RepositoryBase<AccountAtAGlance>, IAccountRepository  
{  
    public Customer GetCustomer(string custId)  
    {  
        using (var context = DataContext)  
        {  
            return context.Customers  
                .Include("BrokerageAccounts")  
                .Where(c => c.CustomerCode ==  
                    custId).SingleOrDefault();  
        }  
    }  
}
```

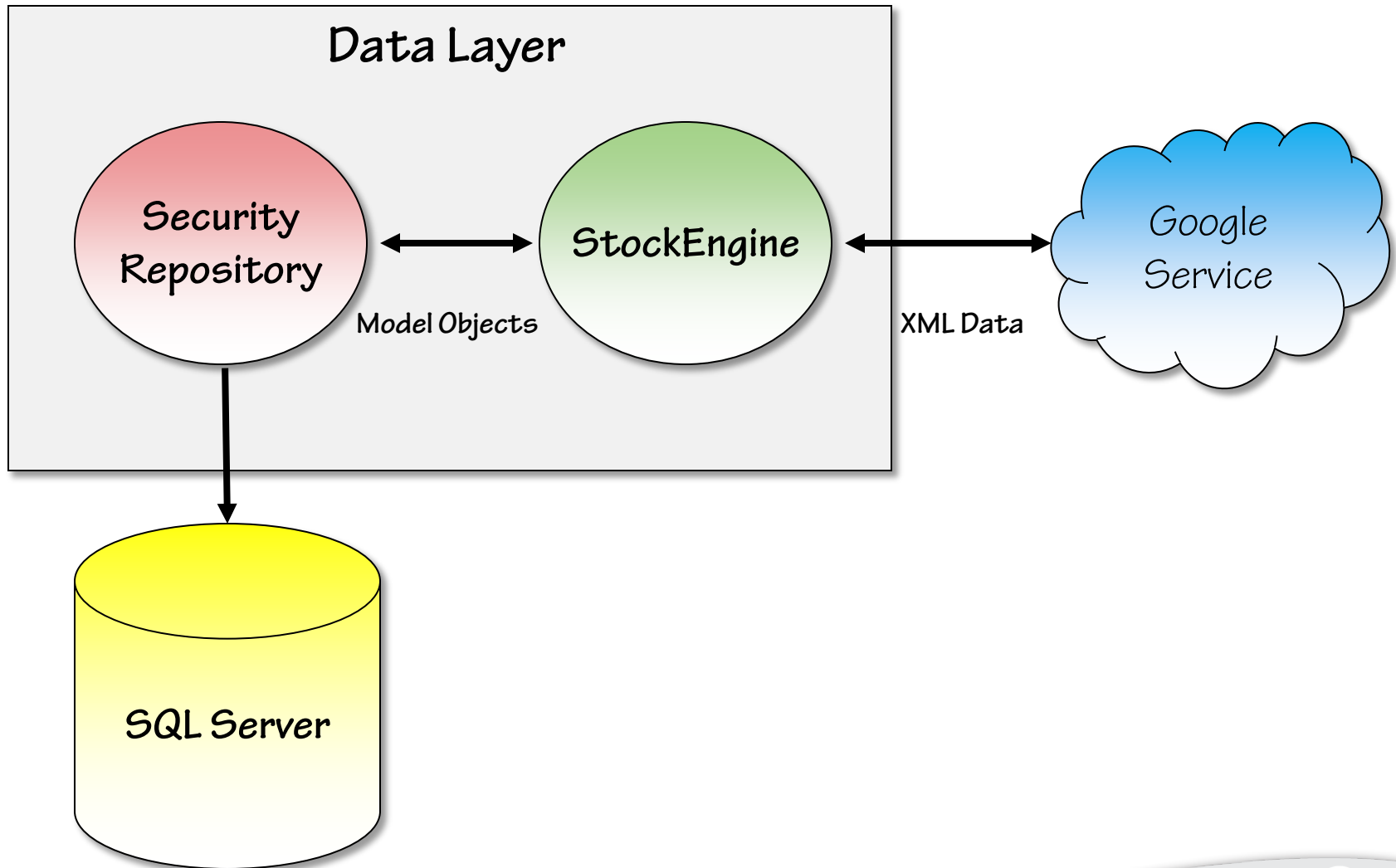


Agenda

- Data Technology Overview
- Creating Model Classes
- Creating a DbContext Class
- Database Structure
- Creating Data Repository Classes
- Retrieving and Storing Financial Data



Retrieving and Storing Data



Retrieving Financial Data

- **Security and market quote data pulled "live" from a Google service:**

<http://www.google.com/ig/api?stock=msft>

- **StockEngine class retrieves and processes data:**
 - XML data deserialized to MarketIndex, Stock, and MutualFund objects
 - LINQ to XML used for deserialization



Storing Financial Data

- **Model object data returned from StockEngine class stored in the database**
- **SecurityRepository class processes and updates database**
 - Calls StockEngine to retrieve and deserialize XML data
 - Provides select, insert, update, and delete functionality
 - Delayed quotes can be pulled from the database rather than calling the Google service every time



Summary

- **Model classes created to hold data**
- **Entity Framework Code First used for data operations**
- **Repository classes created to perform CRUD operations**
- **LINQ to XML used to parse financial data**

