Accordia Group

Entity Framework

ABOUT ACCORDIA

Technologies

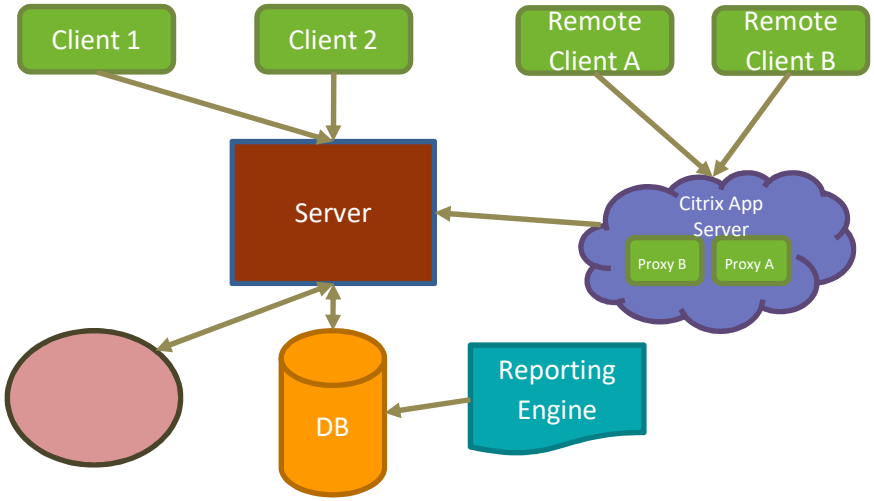
- .NET, C#, WCF, WPF, MEF, Entity Framework
- Java, Java EE, Hibernate
- Delphi
- MS Sql Server, Sybase Server
- HTML, CSS, JavaScript, Angular2 +
- Visual Studio, Eclipse, SQL Management Studio

Our area of expertise

- Financial instruments and transactions
- Analysis and performance of financial portfolios
- Financial derivatives modeling
- Risk management and hedging
- Communication with external systems
- Integration with standardized communication protocols for exchange of information
- Reporting, data analysis and presentation of large amounts of data

3-Tier Architecture

- 3-Tier Architecture with centralized business logic and access to external data systems.
- Role of the client application is to give the ability to the user to interact with the server application
- Easy to change, improve and maintain the system
- High degree of scalability and better performance



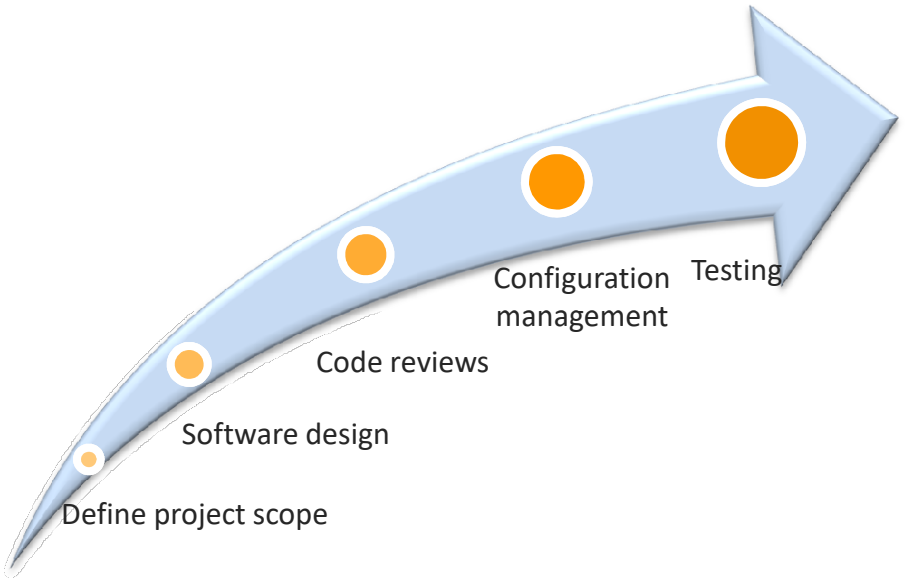
Grid

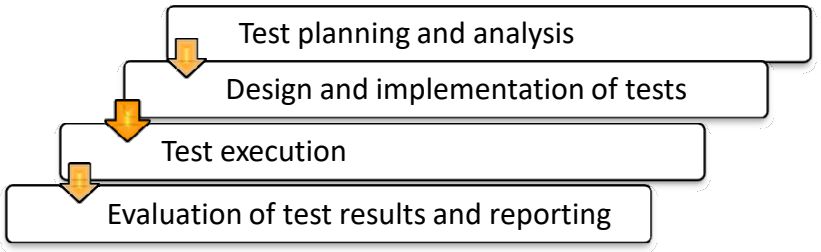
- Efficient presentation and processing large amount of data in tabular form
- Advanced functionalities od grouping, sorting, pivoting and exporting data
- Real-time data analysis
- Integration in hi performance financial applications
- User interface that is easy to use
- View management and ability to share views with individual users or groups

Trader	Fund	Type	End Price	Start Price	Quantity	P&L
⊟ Andy Davis	⊟ GEM	⊟ Bond Future	2,685.00	2,906.00	570	1,321.00
			832.00	1,291.00	653	265.00
		Bond Future	3,517.00	4,197.00	1223	1,586.00
		⊟ Currency	985.00	1,606.00	380	3,760.00
		Currency	985.00	1,606.00	380	3,760.00
	GEM		4,502.00	5,803.00	1603	5,346.00
	⊟ REDOAK	⊟ Equity	1,167.00	1,149.00	131	2,611.00
			1,470.00	1,408.00	1169	-38.00
		Equity	2,637.00	2,557.00	1300	2,573.00
		⊟ Metal Forward	1,642.00	1,584.00	1000	-1,265.00
			732.00	2,291.00	1653	3,214.00
		Metal Forward	2,374.00	3,875.00	2653	1,949.00
	REDOAK		5,011.00	6,432.00	3953	4,522.00
Andy Davis			9,513.00	12,235.00	5556	9,868.00
⊞ Anthony Carter			6,848.00	8,778.00	5463	7,029.00
⊞ Arron Johnson			4,599.00	5,660.00	864	6,553.00
⊞ Brad Murphy			4,346.00	4,404.00	1227	5,897.00
⊞ Matt Hollinger			9,316.00	7,031.00	2475	17,063.00
⊞ Max Lee			9,007.00	11,692.00	2377	424.00
⊞ Melvin Monroe			9,582.00	10,875.00	8264	9,980.00
⊞ Michael Young			1,471.00	1,358.00	1104	2,482.00
⊞ Tim Miller			2,488.00	3,356.00	4086	9,200.00
⊞ Toby Hall			903.00	855.00	145	7,346.00
⊞ Tom Lewis			1,150.00	1,022.00	220	3,498.00
⊞ Tony Hunter			1,261.00	1,157.00	199	-278.00
⊞ Travis Armstrong			2,608.00	2,337.00	417	5,707.00
⊞ Trevor Nixon			991.00	864.00	157	698.00
⊞ Troy Glass			884.00	760.00	2009	4,004.00
⊞ Vernon Wells			3,738.00	3,503.00	1345	5,618.00
⊞ Victor Curry			1,140.00	1,013.00	217	1,691.00
Total			**69,845.00**	**76,900.00**	**36125**	**96,780.00**

Software quality control

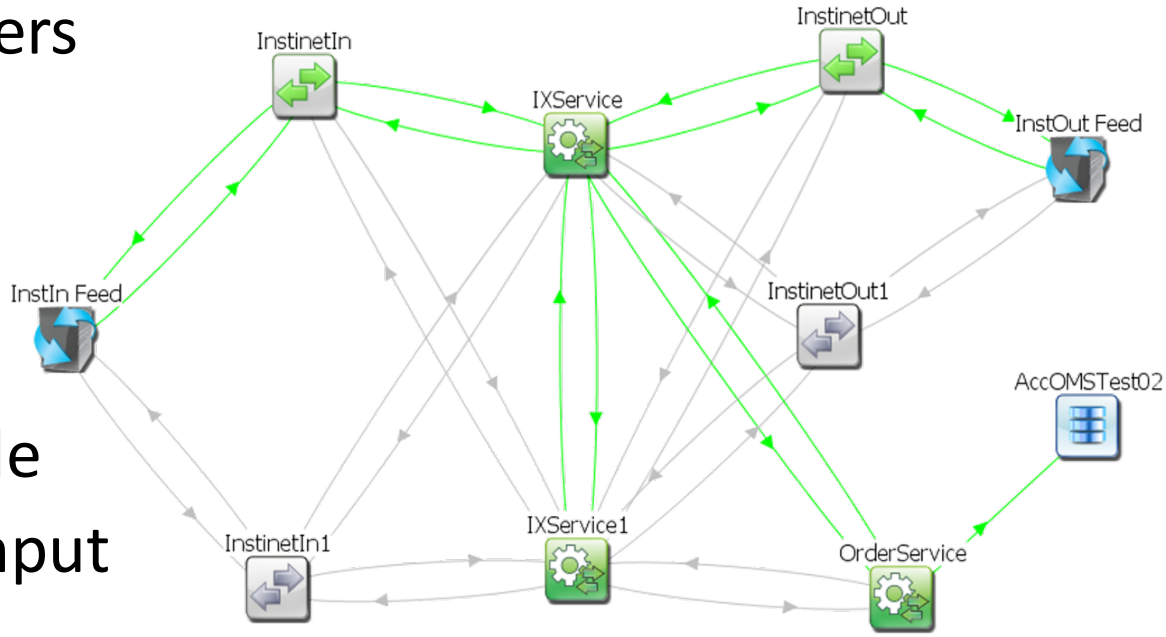
- Quality control ensures that developed software meets desired quality measures.
- Good software is reliable, efficient and satisfies user needs
- Testing is one of the major components in software quality assurance





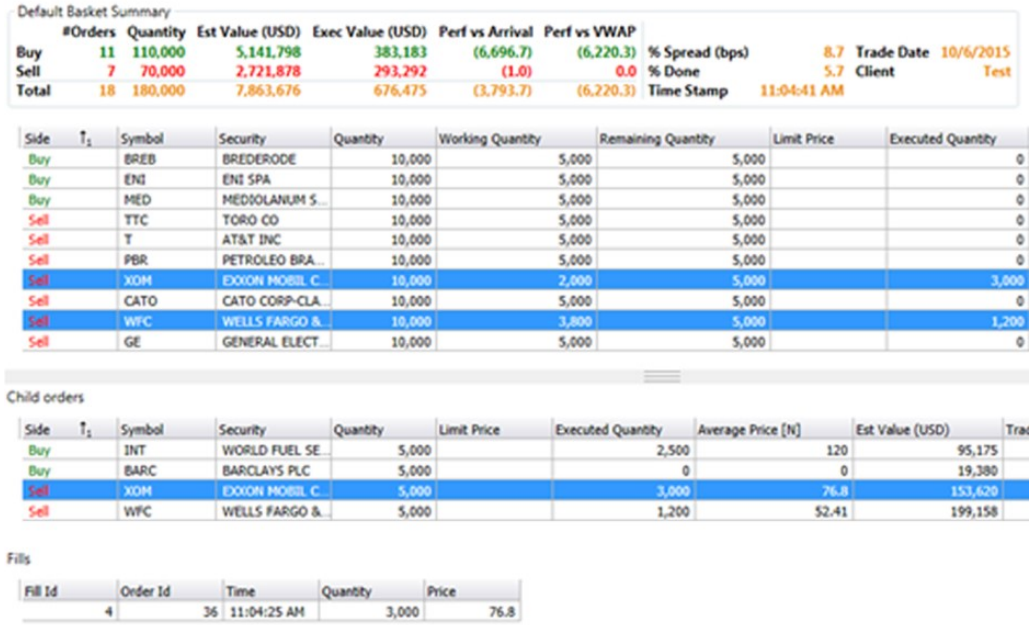
Hub

- Connecting traders and brokers
- System and message flow monitoring
- Fully configurable
- Massive throughput
- Fix financial protocol support
- Zero failure



Trade execution support

- Tracking of orders arriving into the system
- Internal and external (hub) data sources
- Order grouping by different criteria
- Advance algorithms for optimal order execution
  - Assessment od daily trading distribution based on historical data
  - Best price trading (maximize profit)
  - Trading of small amount in specific intervals (avoid negative market impact)
- Sending orders to brokers for execution
- Monitoring order execution process



Default Basket Summary

	#Orders	Quantity	Est Value (USD)	Exec Value (USD)	Perf vs Arrival	Perf vs VWAP				
Buy	11	110,000	5,141,798	383,183	(6,696.7)	(6,220.3)	% Spread (bps)	8.7	Trade Date	10/6/2015
Sell	7	70,000	2,721,878	293,292	(1.0)	0.0	% Done	5.7	Client	Test
Total	18	180,000	7,863,676	676,475	(3,793.7)	(6,220.3)	Time Stamp	11:04:41 AM		

Side	Symbol	Security	Quantity	Working Quantity	Remaining Quantity	Limit Price	Executed Quantity
Buy	BREB	BREDERODE	10,000	5,000	5,000		0
Buy	ENI	ENI SPA	10,000	5,000	5,000		0
Buy	MED	MEDIOLANUM S...	10,000	5,000	5,000		0
Sell	TTC	TORO CO	10,000	5,000	5,000		0
Sell	T	AT&T INC	10,000	5,000	5,000		0
Sell	PBR	PETROLEO BRA...	10,000	5,000	5,000		0
Sell	XOM	EXXON MOBIL C...	10,000	2,000	5,000		3,000
Sell	CATO	CATO CORP-CLA...	10,000	5,000	5,000		0
Sell	WFC	WELLS FARGO &...	10,000	3,800	5,000		1,200
Sell	GE	GENERAL ELECT...	10,000	5,000	5,000		0

Child orders

Side	Symbol	Security	Quantity	Limit Price	Executed Quantity	Average Price [N]	Est Value (USD)	Tra
Buy	INT	WORLD FUEL SE...	5,000		2,500	120	95,175	
Buy	BARC	BARCLAYS PLC	5,000		0	0	19,380	
Sell	XOM	EXXON MOBIL C...	5,000		3,000	76.8	153,620	
Sell	WFC	WELLS FARGO &...	5,000		1,200	52.41	199,158	

Fills

Fill Id	Order Id	Time	Quantity	Price
4	36	11:04:25 AM	3,000	76.8

.NET Framework Essentials

THINGS TO KNOW

.NET Framework

- Namespaces
  - System.Generic.Collections
  - System.Security.Cryptography
  - System.Xml.Serialization
  - System.Runtime.Serialization
  - System.Linq
- Assemblies
  - System.Core.dll
  - System.Security.dll

.NET Framework Frameworks

- WCF (Windows Communication Foundation)
- WPF (Windows Presentation Foundation)
- MEF (Managed Extensibility Framework)
  - .NET Framework 4
  - Attributed Programming Model
  - Parts, Imports, and Exports
  - System.ComponentModel Namespaces
- EF (Entity Framework)
  - EF (or EF 3.5) included in .NET 3.5 SP1, O/RM support using the Database First workflow.
  - System.Data.Entity, System.ComponentModel
  - Version 6 NuGet package

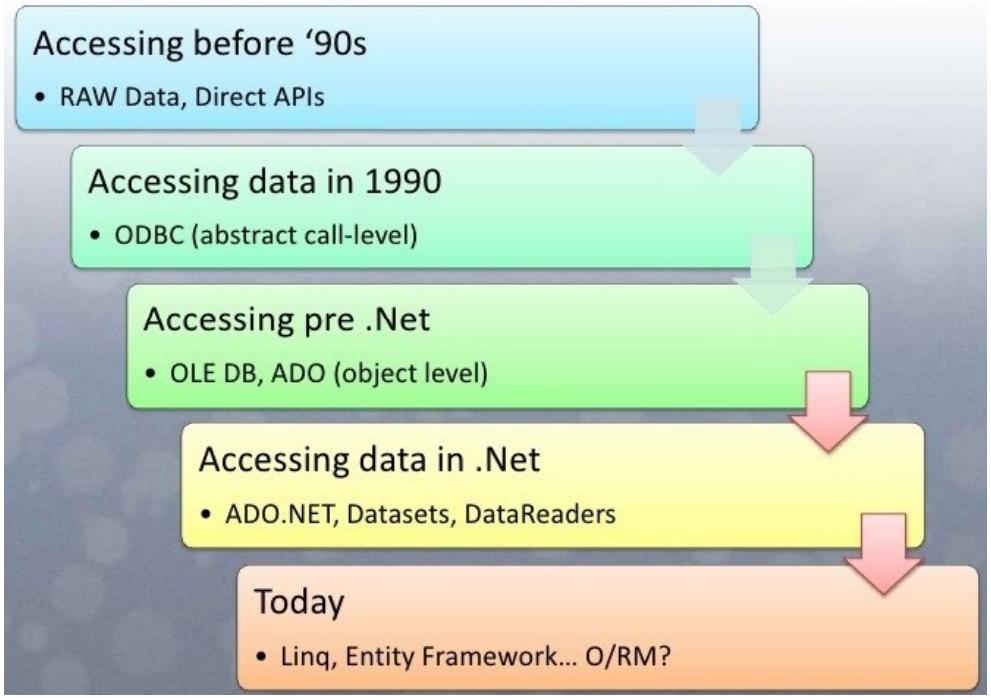
Architecture

ENTITY FRAMEWORK DEFINITION

What is Entity Framework

- Object-relational mapper
- Works with relational data using domain-specific objects
- Eliminates the need for most of the data-access code

What was before ORM



- A lot more code for data access layer
- Remap the data from database according to the classes objects
- Take the data out of the classes push it back up again to persist

Entity Framework Version History

Release	Summary
EF 6.1.2	EF6.1.2 is mostly about bug fixes.
EF 6.1.1	Fixes
EF 6.1	Minor release with few new features
EF 6.0.2	The 6.0.2 patch release is limited to fixing issues that were introduced in the EF6
EF 6.0.1	The 6.0.1 patch release is limited to fixing issues introduced in the EF6 release
EF 6.0	This release can be used in VS 2013, VS 2012 and VS 2010 to write applications that target .NET 4.0 and .NET 4.5.
EF 5	This release can be used in VS 2010 and VS 2012 to write applications that target .NET 4.0 and .NET 4.5.
EF 4.3.1	This patch release included some bug fixes to the EF 4.3 release
EF 4.3	The EF 4.3 release included the Code First Migrations feature
EF 4.2	This release includes bug fixes to the EF 4.1.1 release.
EF 4.1.1	Upgrade
EF 4.1	The EF 4.1 release was the first to be published on NuGet.
EF 4	This release was included in .NET Framework 4 and Visual Studio 2010
EF (or EF 3.5)	The initial release of Entity Framework was included in .NET 3.5 SP1 and VS 2008 SP1 and provided basic O/RM support using the Database First workflow.

Entity Data Model

- Conceptual Model
  - Entities
  - Relationships
- Mapping
- Storage Model
  - Tables
  - Views
  - Relationships, keys

EF Layers

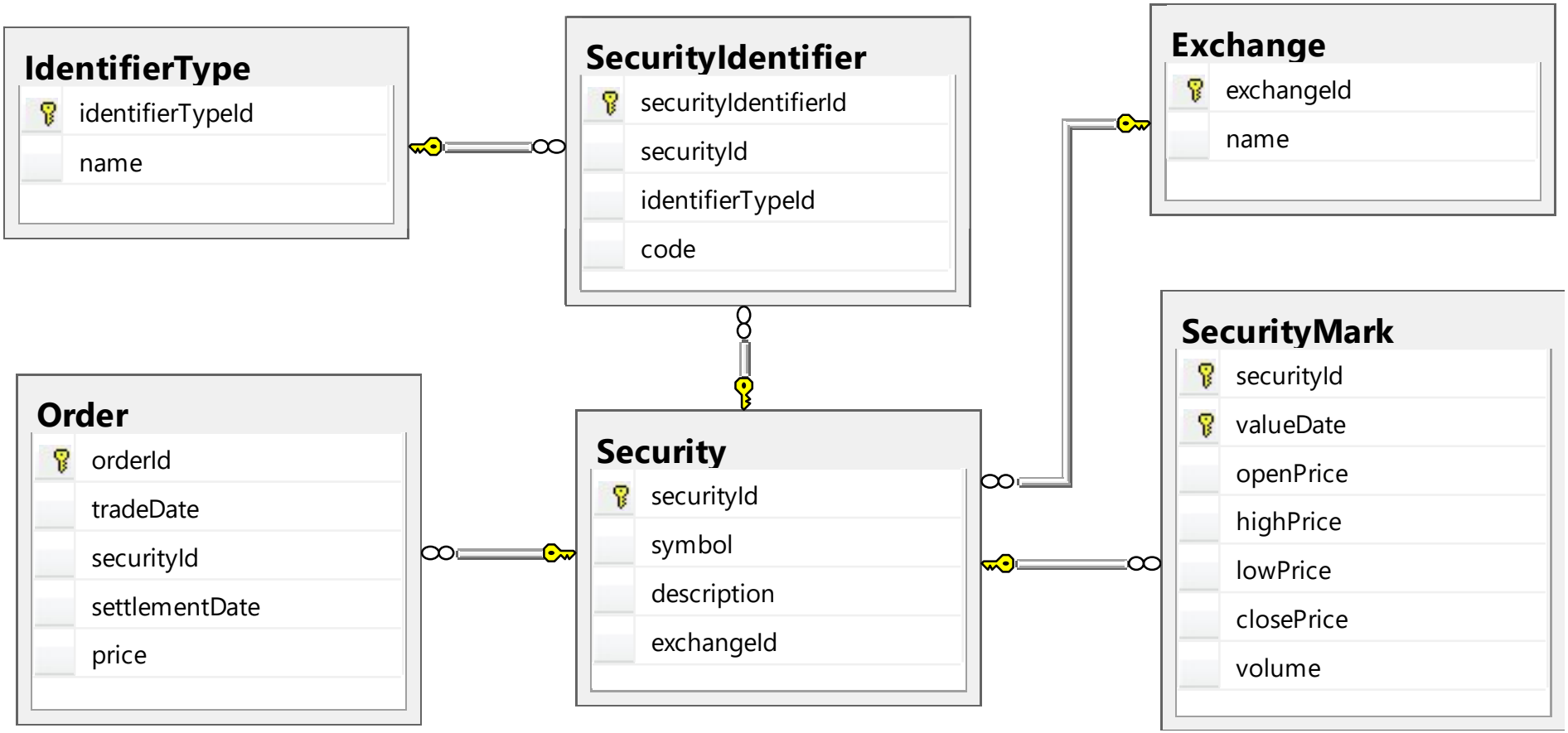
- LINQ to Entities and Entity SQL
  - Query language is used to write queries against the object model
  - Returns entities, which are defined in the conceptual model
- Object Service
  - Entry point for accessing data from/to the database
  - Responsible for materialization
- Entity Client Data Provider:
  - Converts L2E or Entity SQL queries into a SQL
- ADO.Net Data Provider

Design approach

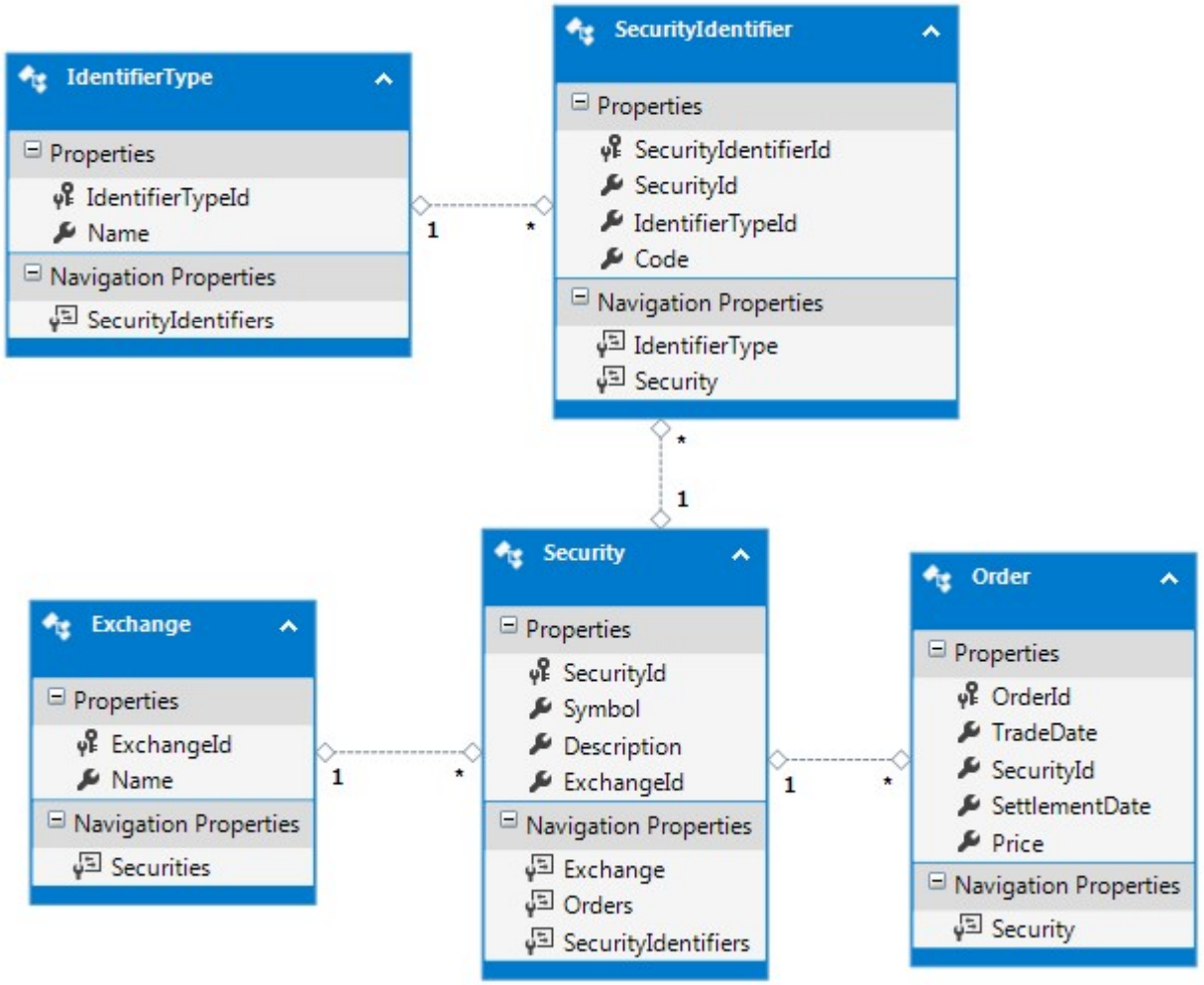
- Database first
  - DB designed by DBAs, developed separately
  - Let EF create entities for you
  - You can always update model from database
  - Easy
- Code first
  - Full control over the code
  - Hardcore programmers don't like designers
  - Works well on large projects with complex database
- Model first

DATABASE FIRST APPROACH

Database TradeDB scheme



ADO.NET Entity Data Model



ADO.NET Entity Data Model 2

- T4 Text Templates (mixture of text blocks and control logic that can generate a text file)
- Context
  - T4 generated
  - DbSets of entities
  - Derived from DbContext
- Diagram
  - Shapes
  - Connectors
- Entities

L2E vs Entity SQL

```
public void QueryByLinqToEntities()
{
      using (var context = new TradeDBEntities())
      {
          foreach (var item in context.Security.Where(x => x.Symbol.StartsWith("Ab"))
              .Where(x => x.Orders.Any(y => y.Price > 100)).OrderByDescending(x => x.SecurityId)
              .Select(x => new { x.Symbol, x.Exchange.Name })
              .Union(context.Security.Where(x => x.Exchange.Name == "NYSE")
              .OrderByDescending(x => x.SecurityId).Select(x => new { x.Symbol, x.Exchange.Name })))

              Console.WriteLine("{0} - {1}", item.Symbol, item.Name);
      }
}

public void QueryByEntitySQL()
{
      using (var context = new TradeDBEntities())
      {
          var query = (from s in context.Security
                       join o in context.Order on s.SecurityId equals o.SecurityId
                       where s.Symbol.StartsWith("Ab") && o.Price > 100 orderby s.SecurityId descending
                       select new { s.Symbol, s.Exchange.Name }).Union
                        (from s in context.Security join e in context.Exchange on s.ExchangeId equals e.ExchangeId
                       where e.Name == "NYSE" orderby s.SecurityId descending
                       select new { s.Symbol, s.Exchange.Name });

          foreach (var item in query)
              Console.WriteLine("{0} - {1}", item.Symbol, item.Name);
      }
}
```

CODE FIRST APPROACH

Attributes

- Attribute class (System namespace, mscorlib.dll assembly)
- Attributed Programming Model or .NET Aspect Oriented Programming
- Attribute Usage (AttributeUsageAttribute)
- Positional parameters (obligatory) and Named parameters (optional)
- Attribute parameters are restricted to constant values: simple types, string, System.Type, enums, arrays of any of the previous types

Code – First Approach

- Entity Framework introduced the Code-First approach with Entity Framework 4.1.
- EF API will create the database based on domain classes and configuration
- Workflow:
  - Create or modify domain classes
  - Configure these domain classes using Fluent-API or data annotation attributes
  - Create or update the database schema

- Use Nugget to install EF in projects
- Code - First requires a context class which should be derived from DbContext class
- Context class exposes DbSet properties
- Database initialization strategies
  - CreateDatabaseIfNotExists: This is the default initializer
  - DropCreateDatabaseIfModelChanges: This initializer drops an existing database and creates a new database, if your model classes (entity classes) have been changed
  - DropCreateDatabaseAlways
  - Custom DB Initializer
- Ability to disable initializer

Data Annotation Attributes

- System.ComponentModel.DataAnnotations
- - Key
- - MaxLength / MinLength
- System.ComponentModel.DataAnnotations.Schema
- - Table
- - Column
- - DatabaseGenerated (Identity, Computed, None)
- - ForeignKey
- - NotMapped

Fluent - API

- Entity Framework Fluent API is used to configure domain classes to override conventions.
  - need to override the OnModelCreating() method of DbContext
  - entity configuration methods configure entity to table and map relationships
  - property configuration methods: Configures property to column mappings
  - methods: HasKey(), HasMany(), HasRequired(), HasOptional(), HasDatabaseGeneratedOption(), HasColumnName(),
- Use Data Annotation attributes and Fluent API at the same time
- EF gives precedence to Fluent API over Data Annotations attributes

Inheritance Strategy

- Inheritance Strategy in Entity Framework 6:
  - Table per Hierarchy (TPH)
  - Table per Type (TPT)
  - Table per Concrete Class (TPC)
- Default conventions
- Override default conventions with data annotation attributes or Fluent-API

COURSE FOR INTERNS

Course topics

- Advanced C# Language Features
- Introduction to Business logic
- Reflection and Attribute Based Programming
- Managed Extensibility Framework
- Multi-threaded Programming
- XML Serialization
- ADO.NET Entity Framework
- Windows Communication Foundation
- Windows Presentation Foundation

Course schedule

week	day	Task 1	Task 2
1st	1	2. Advanced C# Language Features	3. Financial Terms and Definitions
	2	2. Task: Expressions Model	3. Primary and Derived Instruments
	3	4. Reflection and Attribute Based Programming	3. Swaps
	4	4. Task: Build plug-in framework	3. Task: Bonds
	5	5. MEF	3. Task: Repos
2nd	6	5. Task: Plug-in pattern example	3. Trading Terms and Definitions
	7	6. Multithreading	3. Trades and Positions
	8	6. Task: Singleton pattern example	3. Trade Matching
	9	7. XML	3. P&L
	10	7. Task: Persistence of singleton cache	3. Risk
3rd	11	8. ADO.NET entity framework	
	12	8. Task: db schema, scripts	
	13	8. Task: entity model, T4	
	14	9. WCF	
	15	9. Task: server, querying methods	
4th	16	9. Task: create, update methods	
	17	10. WPF: container ctrls, trees, properties, data binding, MVVM, events, D&D, commands, content and item ctrls	
	18	10. WPF: templates, virtualization, resources, user ctrls, styles, triggers, custom ctrls	
	19	10. Task: security form, new, edit	
	20	10. Task: security list, search	

Contact

Učitelj Tasina 38, Regionalna kancelarija Niš

018 519 903

www.accordia-group.com

office@accordia.co.rs

Dejan Paunović

Dejan.Paunovic@accordia.co.rs

www.linkedin.com/in/dejanpaunovic

Ivan Anđelković

Ivan.Andjelkovic@accordia.co.rs

THANK YOU