HEINSOHN
BUSINESS TECHNOLOGY

Acceso a Datos con ADO.NET

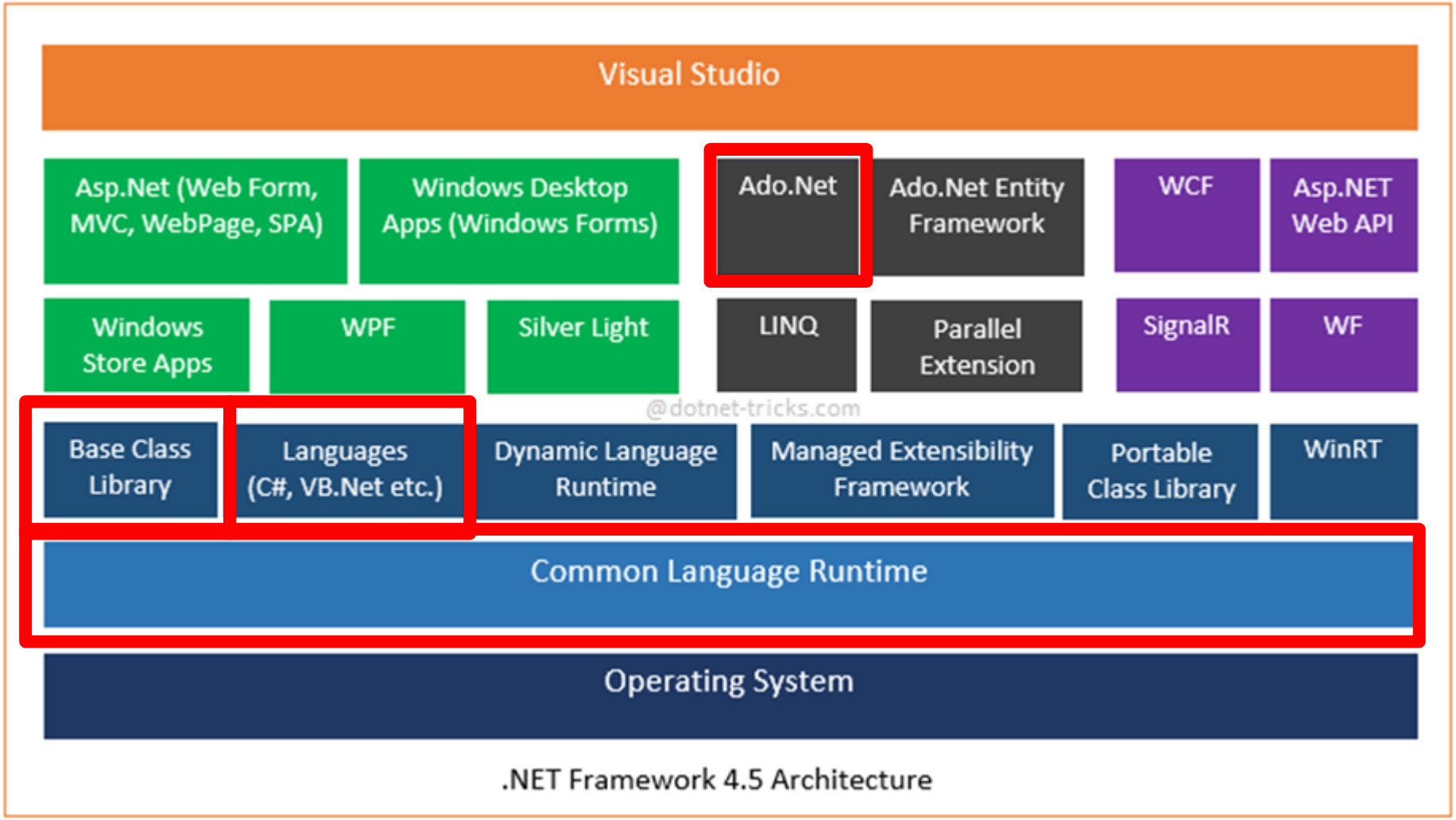
Linq - Entity Framework

Microsoft Partner

Gold Software Development
Gold Web Development
Gold Software Asset Management

www.heinsohn.com.co

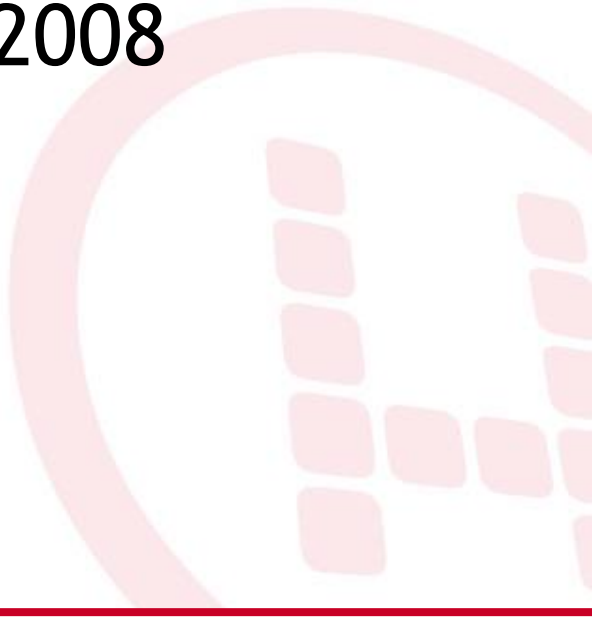
.Net Framework 4.5 Architecture



HEINSOHN
BUSINESS TECHNOLOGY

Que es LINQ?

- Language Integrated Query
- Make query a part of the language
- Component of .NET Framework 3.5
- Shipping from Visual Studio 2008



HEINSOHN
BUSINESS TECHNOLOGY

Query without LINQ

- Objects using loops and conditions

```
foreach(Customer c in customers)
  if (c.Region == "UK") ...
```

- Databases using SQL

```
SELECT * FROM Customers WHERE Region='UK'
```

- XML using XPath/XQuery

```
//Customers/Customer[@Region='UK']
```

HEINSOHN
BUSINESS TECHNOLOGY

ADO without LINQ

```
SqlConnection con = new SqlConnection(...);
con.Open();
SqlCommand cmd = new SqlCommand(
            @"SELECT * FROM Customers
                WHERE c.Region = @Region", con
            );
cmd.Parameters.AddWithValue("@Region", "UK");
DataReader dr = cmd.ExecuteReader();
while (dr.Read()) {
  string name = dr.GetString(dr.GetOrdinal("Name"));
  string phone = dr.GetString(dr.GetOrdinal("Phone"));
  DateTime date = dr.GetDateTime(3);
}
dr.Close();
con.Close();
```

Query with LINQ

C#

```
var myCustomers = from c in
customers
where c.Region == "UK"
select c;
```

HEINSOHN
BUSINESS TECHNOLOGY

More LINQ queries

C#

```
var goodCusts = (from c in db.Customers
    where c.PostCode.StartsWith("GY")
    orderby c.Sales descending
    select c).Skip(10).Take(10);
```

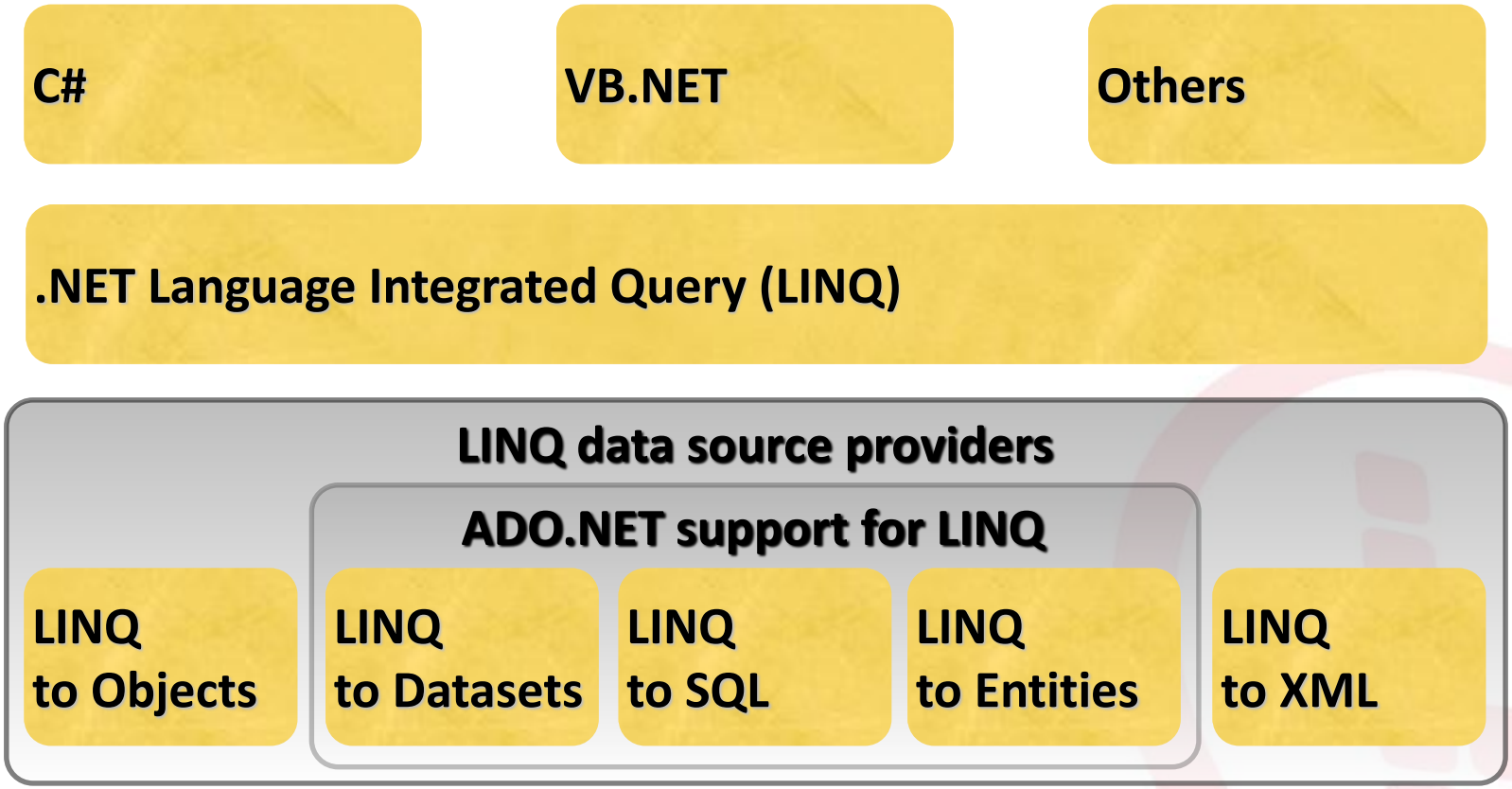
HEINSOHN
BUSINESS TECHNOLOGY

Advantages

- Unified data access
  Single syntax to learn and remember
- Strongly typed
  Catch errors during compilation
- IntelliSense
  Prompt for syntax and attributes
- Bindable result sets

HEINSOHN
BUSINESS TECHNOLOGY

Architecture



LINQ to Objects

C#

```
int[] nums = new int[] {0,4,2,6,3,8,3,1};
double average = nums.Take(6).Average();

var above = from n in nums
            where n > average
            select n;
```

HEINSOHN
BUSINESS TECHNOLOGY

LINQ to Objects (Lambda)

C#

```
int[] nums = new int[] {0,4,2,6,3,8,3,1};
double average = nums.Take(6).Average();

var above = nums.Where(X => X > average);
```

HEINSOHN
BUSINESS TECHNOLOGY

LINQ operators

Aggregate	Conversion	Ordering	Partitioning	Sets
Aggregate	Cast	OrderBy	Skip	Concat
Average	OfType	ThenBy	SkipWhile	Distinct
Count	ToArray	Descending	Take	Except
Max	ToDictionary	Reverse	TakeWhile	Intersect
Min	ToList			Union
Sum	ToLookup			
	ToSequence			

and many others

HEINSOHN
BUSINESS TECHNOLOGY

LINQ to SQL

- Object-relational mapping
  Records become strongly-typed objects
- Data context is the controller mechanism
- Facilitates update, delete & insert
- Translates LINQ queries behind the scenes
- Type, parameter and injection safe

HEINSOHN
BUSINESS TECHNOLOGY

Database mapping

- VS designer or SQLMetal command
- Map tables & fields to classes & properties
- Generates partial classes with attributes
- Each record becomes an object
- Data context represents the database
- Utilise tables, views or stored procedures

HEINSOHN
BUSINESS TECHNOLOGY

Modifying objects

- Update
  Set object properties
- Delete
  context.Table.DeleteOnSubmit(object)
- Insert
  context.Table.InsertOnSubmit(object)
- Commit changes back
  context.SubmitChanges()
  Transactional - all or nothing

Linq to SQL (Abstract)

- It only works with SQL Server Database.
- It has not support for complex type.
- It cannot generate database from model.
- It allows only one to one mapping between the entity classes and the relational tables /views.
- It allows you to query data using DataContext.
- It can be used for rapid application development only with SQL Server.

101 Linq Samples

https://code.msdn.microsoft.com/101-LINQ-Samples-3fb9811b#content

LA MEJOR FORMA DE BUSCAR FÁCIL ACCESO A EJEMPLOS SOBRE LINQ

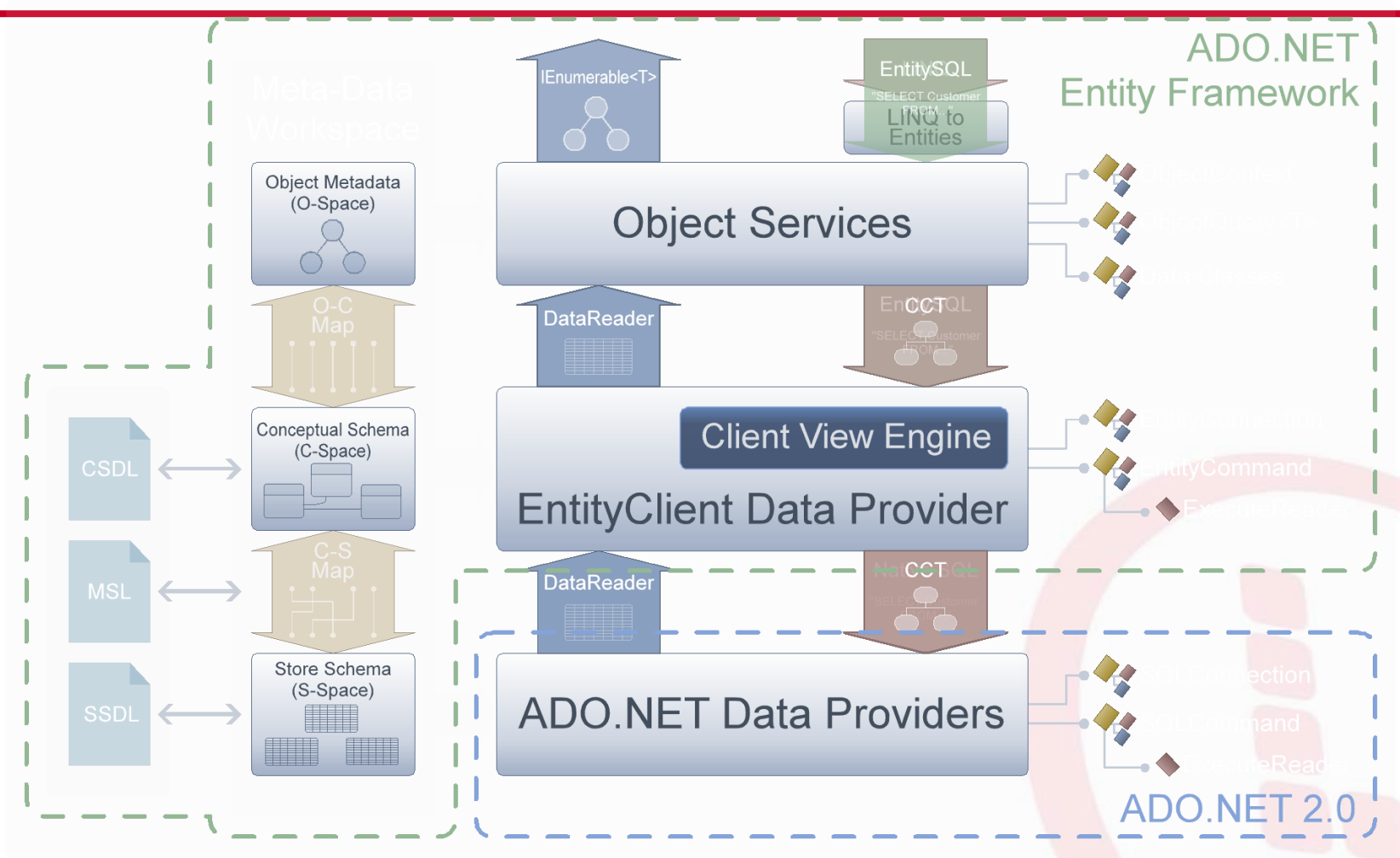
Lamba Expressions

http://linq101.nilzorblog.com/linq101-lambda.php

EJEMPLOS PRÁCTICOS CON LAMBDA

HEINSOHN
BUSINESS TECHNOLOGY

ADO.NET Entity Framework



ADO.NET Entity Framework

- It can works with various databases like Oracle, DB2, MYSQL, SQL Server etc.
- It generates an .edmx files initially. The relation is maintained using 3 different files .csdl, .msl and .ssdl
- It has support for complex type.
- It can generate database from model.
- It allows one-to-one, one-to-many & many-to-many mappings between the Entity classes and the relational tables /views
- It allows you to query data using EntitySQL, ObjectContext, DbContext.
- It can be used for rapid application development with RDBMS like SQL Server, Oracle, DB2 and MySQL etc.

HEINSOHN
BUSINESS TECHNOLOGY

The Problem
Programming Data is Hard

- Writing queries is difficult
  - No help from compiler
  - Results are untyped rectangular records
- Database Schemas optimized for storage concerns
  - Relational Tables contain flat, homogenous records
    - Implicit Logic Embedded in Application
  - Brittle, Hard to maintain
- Lack of common syntax across relational databases

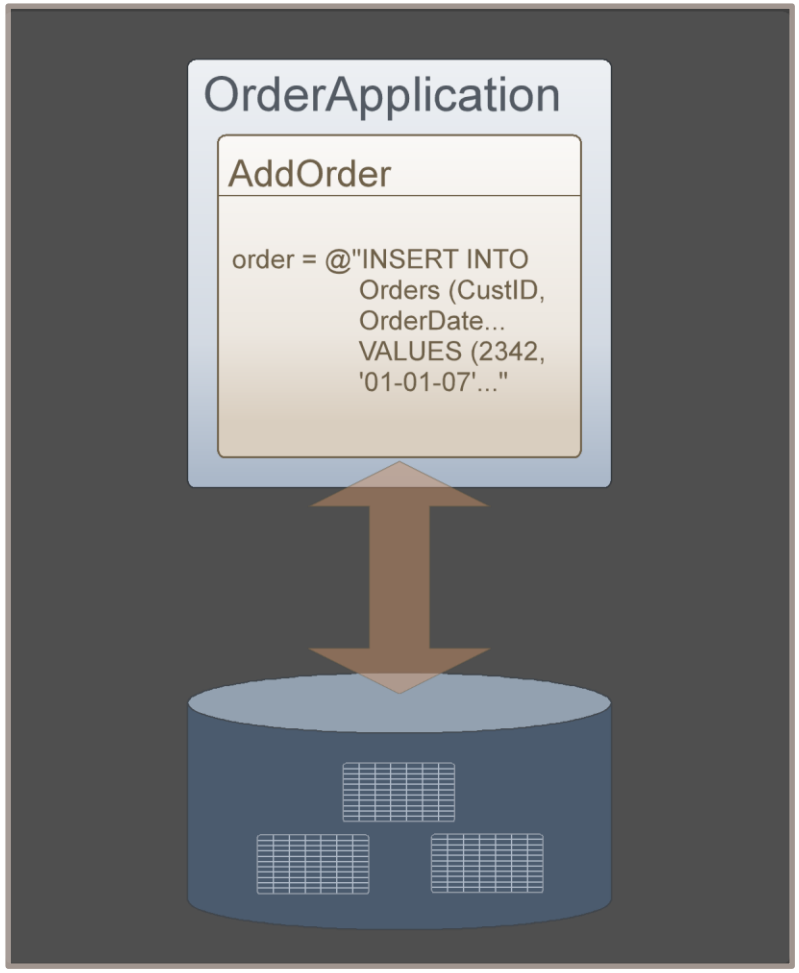
HEINSOHN
BUSINESS TECHNOLOGY

The Opportunity
Increase Developer Productivity

- Rapid Development
  - Strongly typed queries
  - Strongly typed results with Business Logic
- Lower TCO
  - Work with an explicit data model
    - Types, Inheritance, Relationships, Complex Properties,…
  - Decouple application from storage schema
- Better Portability
  - Common query language across disparate sources

HEINSOHN
BUSINESS TECHNOLOGY

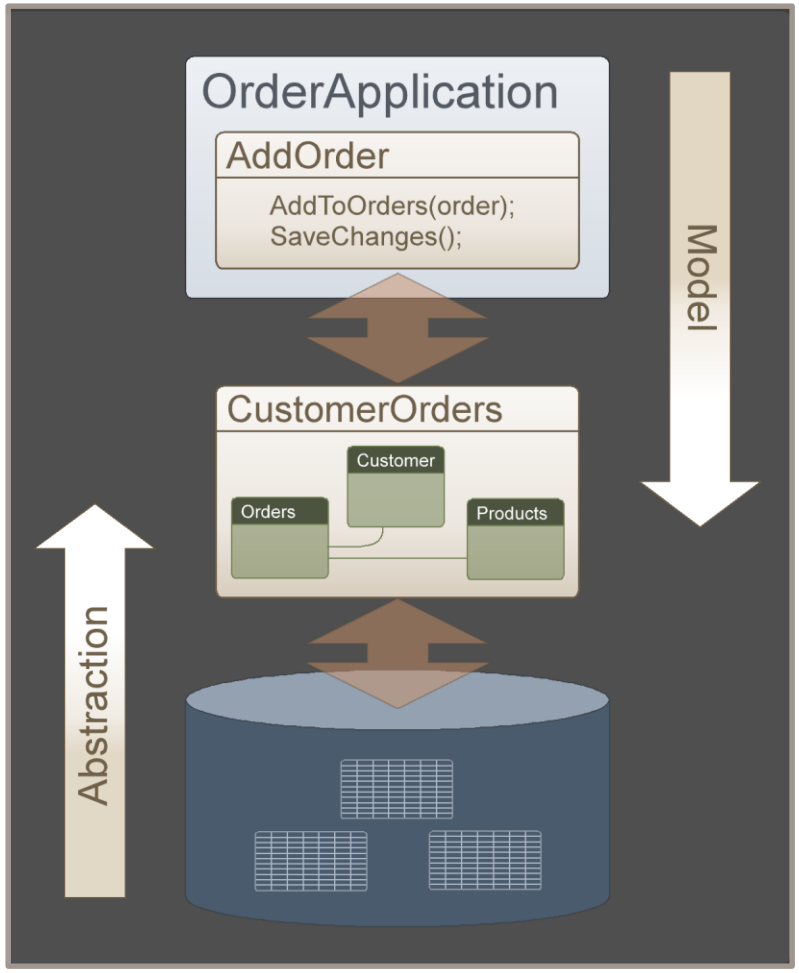
Where's Your Data Model?



- Applications Today…
  - Implicitly Contain the Data Model
  - Logic and Model Intertwined
  - Conceptual Mismatch
  - Often encapsulate in a "Data Access Layer"

HEINSOHN
BUSINESS TECHNOLOGY

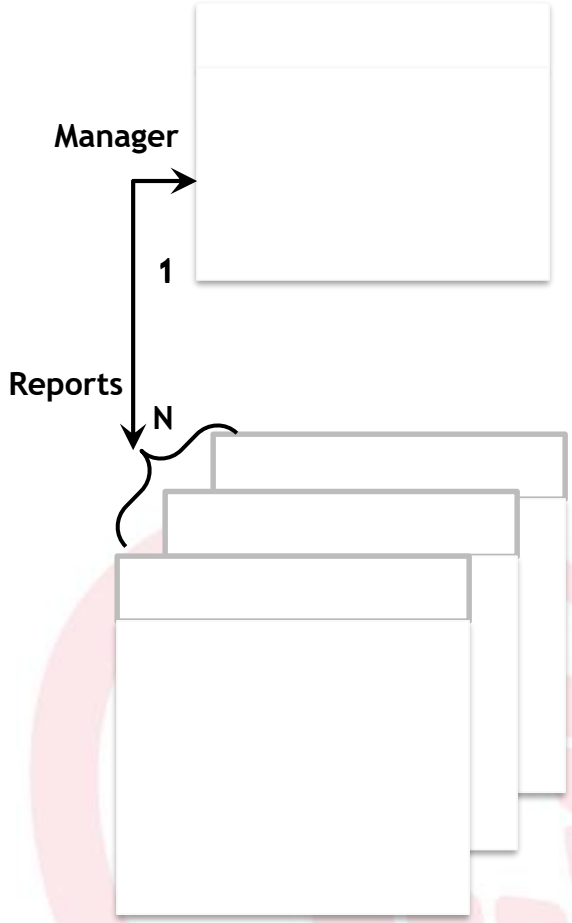
Where's Your Data Model?



- Applications Today…
  - Implicitly Contain the Data Model
  - Logic and Model Intertwined
  - Conceptual Mismatch
  - Often encapsulate in a "Data Access Layer"
- The Need…
  - Applications work with a well Defined Model
  - Storage Schema Abstraction
    - Declarative mapping between application and storage models
    - No brittle, hard-coded mapping

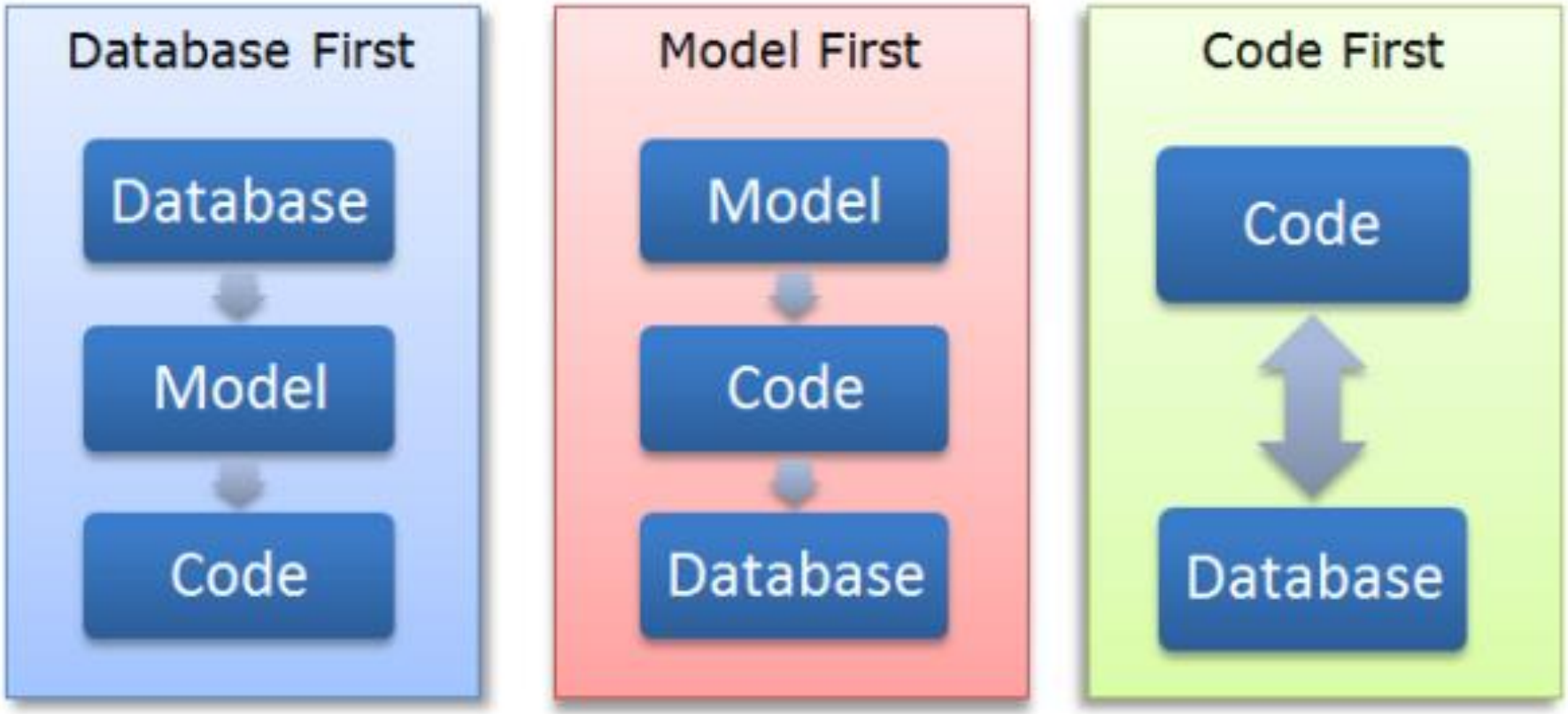
The Microsoft Entity Data Model

- An extended relational model with Entity-Relationship Model concepts
  - Entity Types
    - Strong type with Identity
    - Inheritance
    - Scalar/Complex properties
  - EntitySets
    - Hold instances of Entity Types
      - Similar to relational tables
    - Can have multiple Entitysets of the same EntityTypes
  - Relationships ("Associations")
    - Named relationships between Entities
    - 0..1:*, 0..1:0..1, 1:1, 1:M, M:N
    - Navigation may be exposed as NavigationProperties on EntityTypes
  - AssociationSets
    - Contains instances of associations
    - May be queried directly
  - EntityContainers
    - Contains EntitySets, AssociationSets



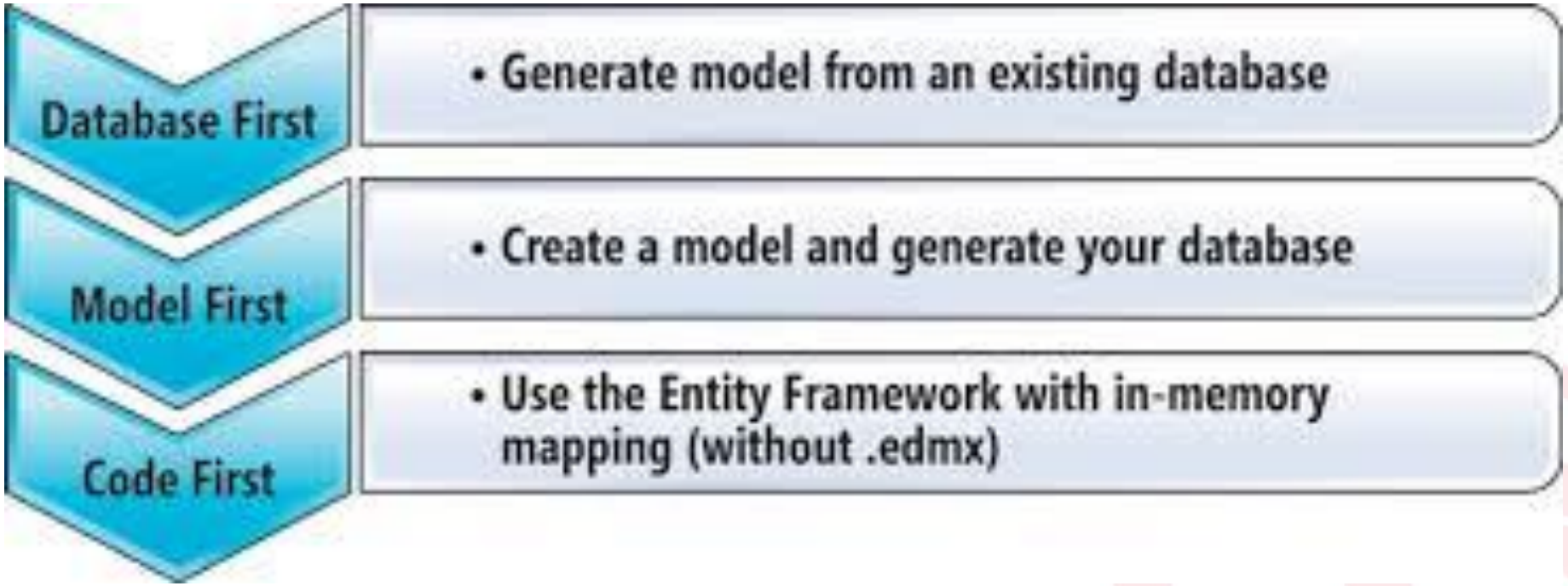
HEINSOHN
BUSINESS TECHNOLOGY

Entity Framework Approaches



HEINSOHN
BUSINESS TECHNOLOGY

Entity Framework Approaches



HEINSOHN
BUSINESS TECHNOLOGY

Entity Framework Approaches

DATABASE FIRST

Model the database artifacts (tables, views, primary keys, foreign keys, etc.)

Allows you to use legacy database in your ORM application

Low level control over the database

Entity Data Model can be updated whenever database schema changes.

Database-first approach supports stored procedure, view, etc.

MODEL FIRST

Creating a model using the Entity Framework designer tools

Designer can generate DDL statements to create the database

An easy-to-use UI allows quickly create new entities

Separation of code and database in a declarative format

Dropping and re-creating the database is not a choice

This uses an .EDMX file to store model and mapping information.

CODE FIRST

The primary focus is getting the code and logic

Define entity model with classes and mappings in code..

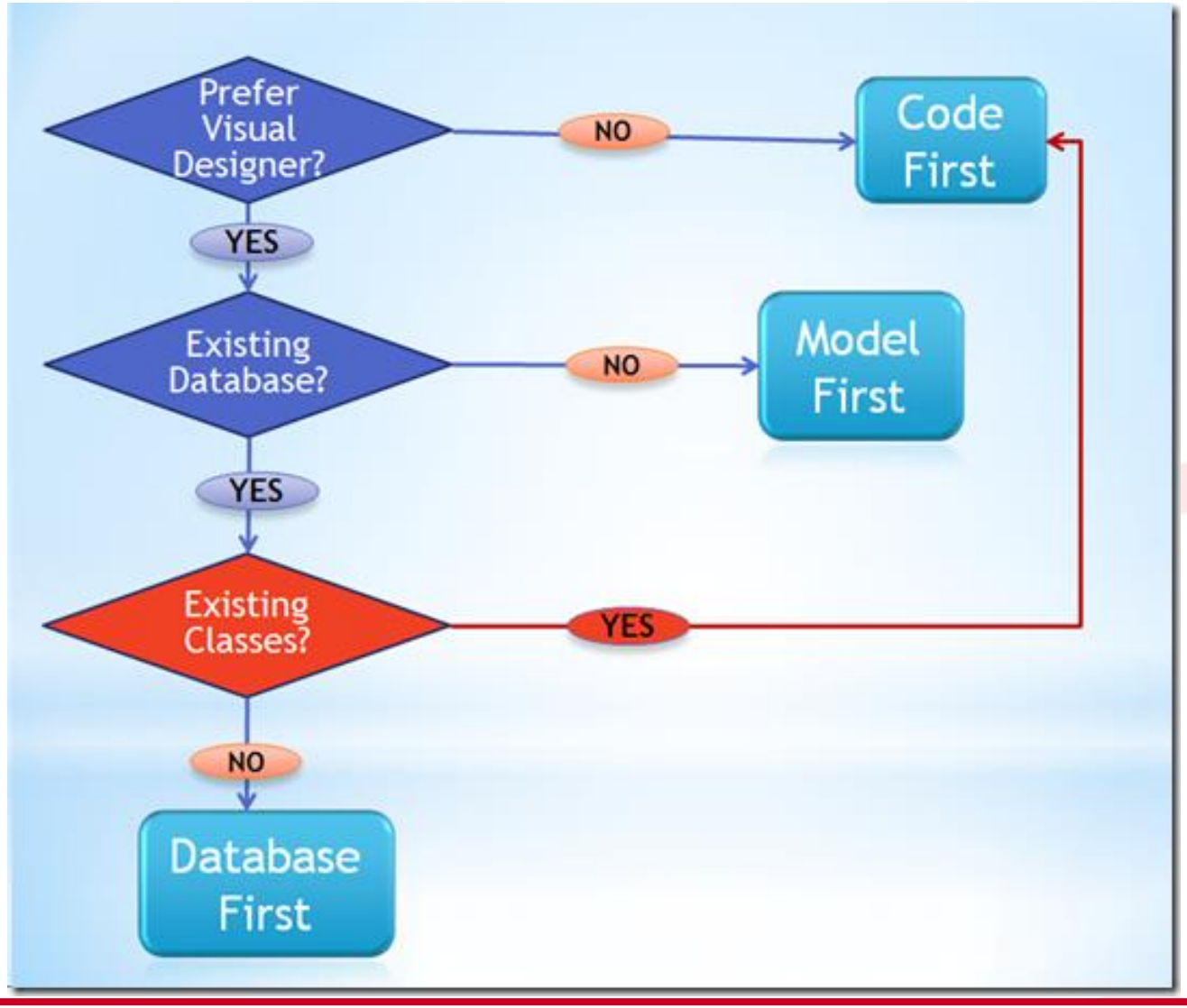
Entity Framework create (and recreate) the database for you.

Does not use Entity Framework visual design tool, neither to design entities

There is no XML file – no EDMX file - to represent model and storage schemas

HEINSOHN
BUSINESS TECHNOLOGY

Selecting Approach



HEINSOHN
BUSINESS TECHNOLOGY

DEMO

- Crear el Entity Model de Nortwind
- Mostrar los registros de la tabla de clientes
- Filtrar un cliente
- Crear un cliente
- Crear un pedido

https://msdn.microsoft.com/en-us/library/aa697427(VS.80).aspx?tduid=(6b7b599b8c322c79ee0d32babf11d3ed)(256380)(2459594)(TnL5HPStwNw-buFSaah3HjmnFMBD1TtcMA)()

HEINSOHN
BUSINESS TECHNOLOGY