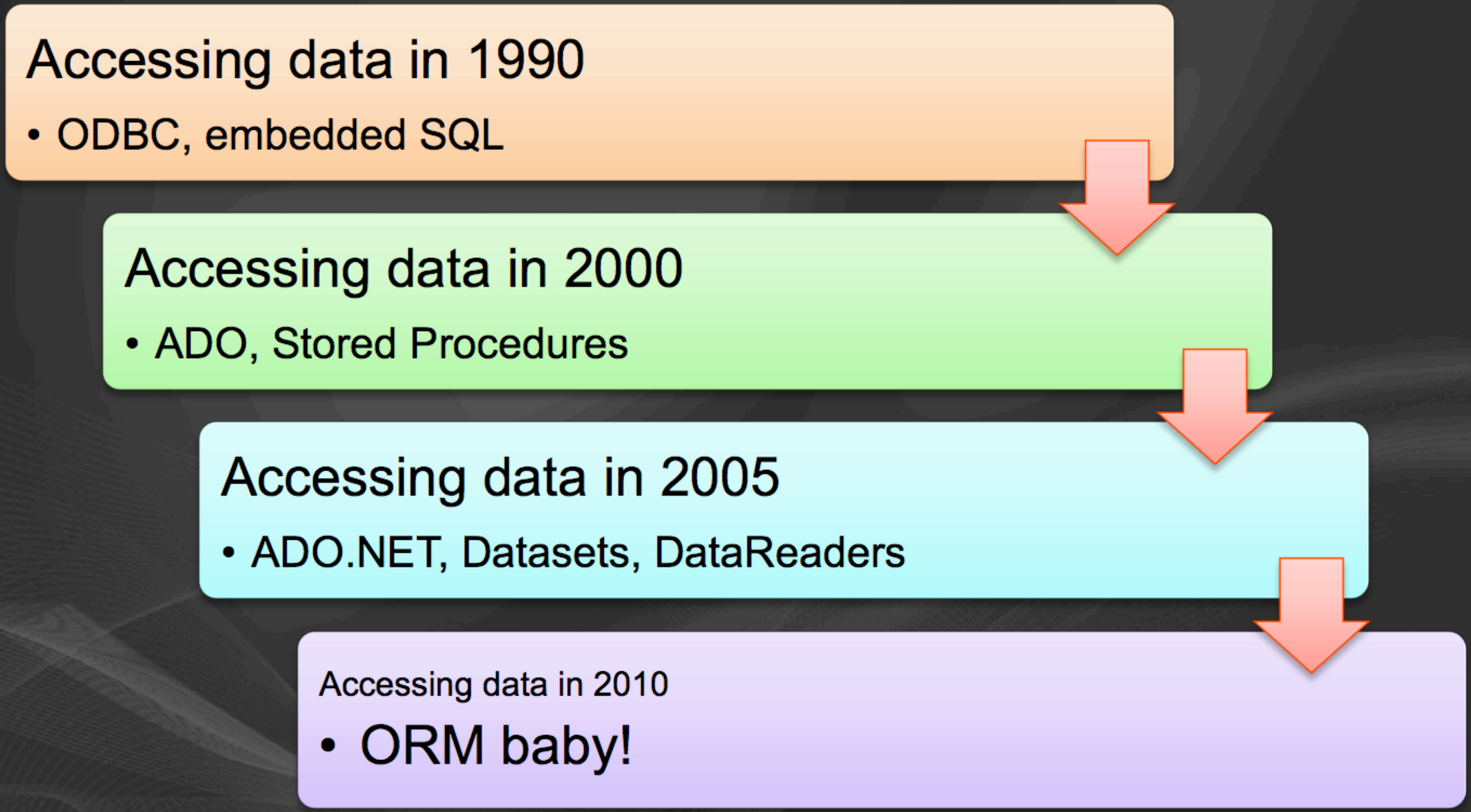
ORM, EDM, ESQL, Entity Framework, LINQ to SQL, LINQ to Entities - confused?"

Eric Nelson
Developer & Platform Group
Microsoft Ltd
eric.nelson@microsoft.com
http://geekswithblogs.net/IUpdateable
http://twitter.com/ericnel

Data Access over the years



Accessing data in 1990

- ODBC, embedded SQL

Accessing data in 2000

- ADO, Stored Procedures

Accessing data in 2005

- ADO.NET, Datasets, DataReaders

Accessing data in 2010

- ORM baby!

Object Relational Mapping

- What is it?
  - an ***Abstraction***
  - technique for working with relational tables as if they were objects in memory
  - Intention is to hide away the complexity of the underlying tables and give a uniform way of working with data
- Why use it?
  - Developer ***productivity***
  - Retain ***database independence***
- Note:
  - Using an ORM does not mean you must use the ORM!

ORM on Windows

- ***Many ORMs*** out there
  - LLBLGen Pro http://www.llblgen.com/
  - Nhibernate http://www.hibernate.org/343.html
  - EntitySpaces http://www.entityspaces.net/Portal/Default.aspx
  - Open Access http://www.telerik.com/products/orm.aspx
  - DevForce http://www.ideablade.com/
  - XPO http://www.devexpress.com/Products/NET/ORM/
  - Lightspeed http://www.mindscape.co.nz/products/LightSpeed/default.aspx
  - Plus many, many more
- ***No clear "winner"*** = relatively little adoption of ORM
  - Developers waiting on Microsoft
  - Of 31 .NET ORMs in 2003, 9 lasted to 2008

It was a long wait...2002 to 2008

- Typed Datasets (cough) – shipped ☺
- ObjectSpaces “v1” – never shipped ☹
- ObjectSpaces “v2” – never shipped ☹
- Microsoft Business Framework – never shipped ☹
- WinFS – never shipped ☹
- ***LINQ to SQL*** - shipped November 2007 ☺
  - Visual Studio 2008 & .NET Framework 3.5
- ***LINQ to Entities*** - shipped August 2008 ☺
  - Visual Studio 2008 SP1 & .NET Framework 3.5 SP1

Note: ***LINQ to Entities*** is the most visible part of the ***ADO.NET Entity Framework***

The future of ORM on Windows

- ***Entity Framework*** and ***LINQ to Entities*** is our ***strategic technology***
  - Entity Framework v2 in VS2010 and .NET Framework 4.0
    - Best of LINQ to SQL moves into LINQ to Entities
  - Microsoft is using it
    - Data Services - shipping
    - Reporting Services
    - Microsoft “M”
- ***Partners supporting*** it
  - Database Vendors – IBM,OpenLink, Data Direct, Devart etc
  - ORM vendors supporting it
    - DevForce now target Entity Framework, replacing their own
    - LLBLGen v3 will target Entity Framework as well as their own
- It is ***not just*** about ORM

LINQ

- What is LINQ?
  - LINQ is ***not about databases***, it is ***about objects***
  - LINQ = Language-Integrated Query
    - Extensions to programming languages to query objects in memory
- Why use it?
  - Productivity!!! LINQ rocks
  - www.linqpad.net
- We ship
  - LINQ to Objects
  - LINQ to XML
  - LINQ to Datasets
  - LINQ to SQL – just SQL Server
  - LINQ to Entities – added in .NET Framework 3.5 SP1
- Others make LINQ to Flickr etc

{ Simple Walkthough }

demo

LINQ to SQL vs LINQ to Entities

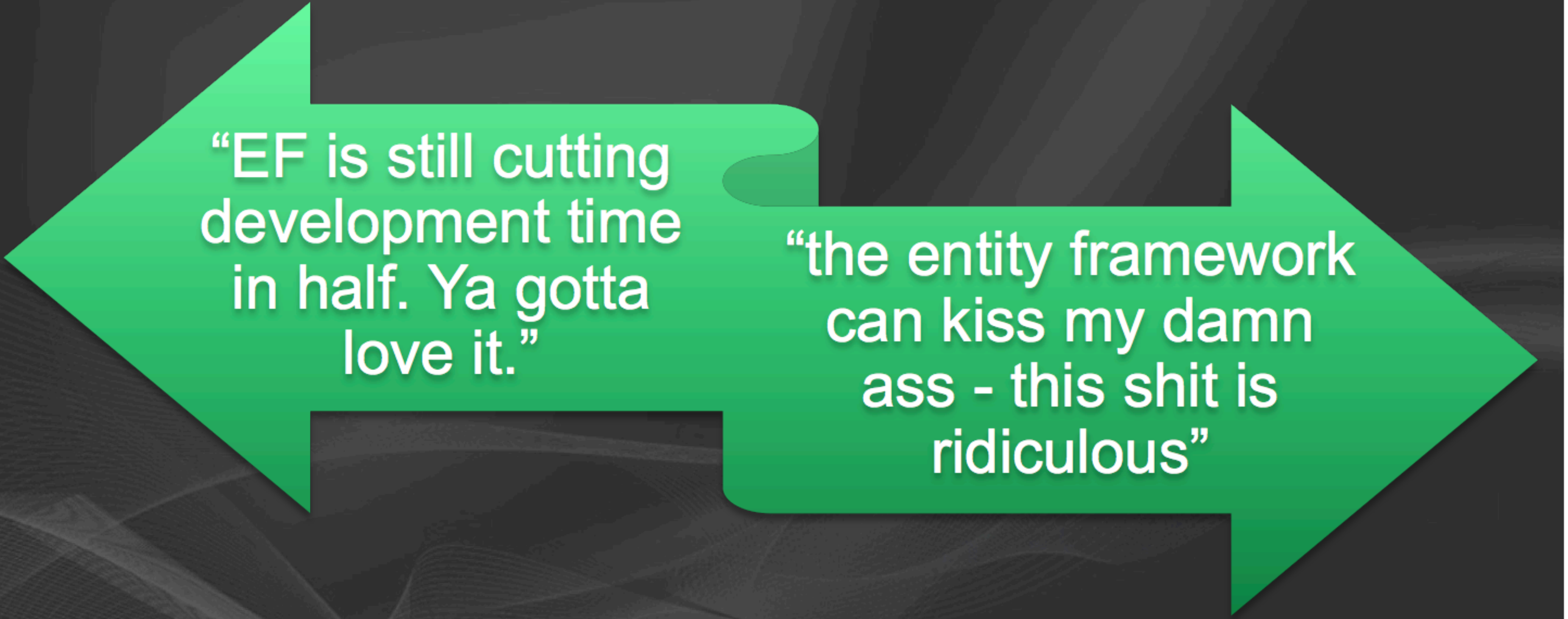
	LINQ to SQL	**LINQ to Entities**
Database Support	SQL Server	SQL Server, DB2, Oracle, Sybase, MySQL ...
Object Relational Mapping Capabilities	Simple	Complex
Status	Not strategic ☺	Strategic
Annoying	Rarely	Often ☺

Quick Overview

- Entity Framework is ***not just about ORM***
- Entity Framework v1 ***is being adopted***
  - Typically on applications expected to “live” a long time
- Entity Framework is ***only .NET 3.5 SP1*** and above
- Entity Framework is probably not the best choice for a “short lived” SQL Server applications
- Entity Framework v1 has “warts”
  - Designer
    - It is annoying – buggy, clunky
    - Exposes subset of the functionality
    - Does not support model first
  - N-tier story
  - Stored Procedure Support
  - Foreign Keys
  - PoCo
  - Guids
  - SQL 2008 New Types

Early adopter feedback

Courtesy of twitter ☺



“EF is still cutting development time in half. Ya gotta love it.”

“the entity framework can kiss my damn ass - this shit is ridiculous”

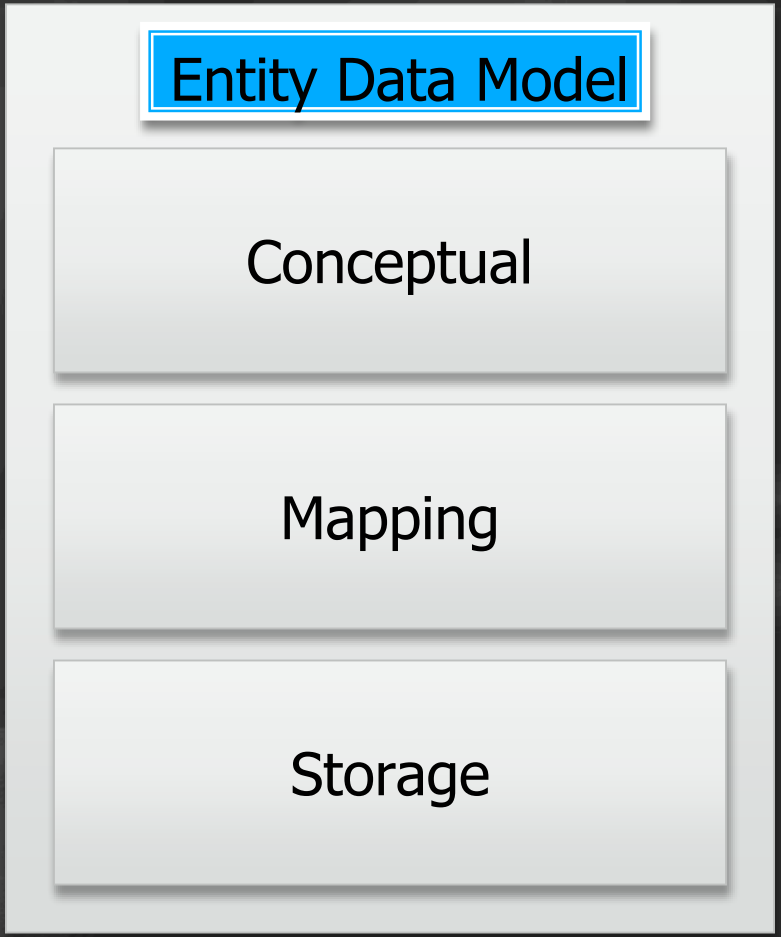
Entity Framework

ADO.NET Entity Framework

- What is it?
  - Tools and services to ***create an Entity Data Model***
    - EDM gives ORM to SQL Server, Oracle, DB2 etc
  - Tools and services for ***consuming an Entity Data Model***
- Why use it?
  - Productivity
  - Complex mapping between entities and database tables
  - Works great with ADO.NET Data Services
- Notes
  - Strategic but just released...

Entity Data Model

- Application model
  - Mapped to a persistence store
- Comprised of three layers:
  - Conceptual (CSDL)
  - Mapping (MSL)
  - Storage (SSDL)
- Database agnostic
- Comprised of:
  - Entities
  - Associations
  - Functions



Entity Data Model

Conceptual

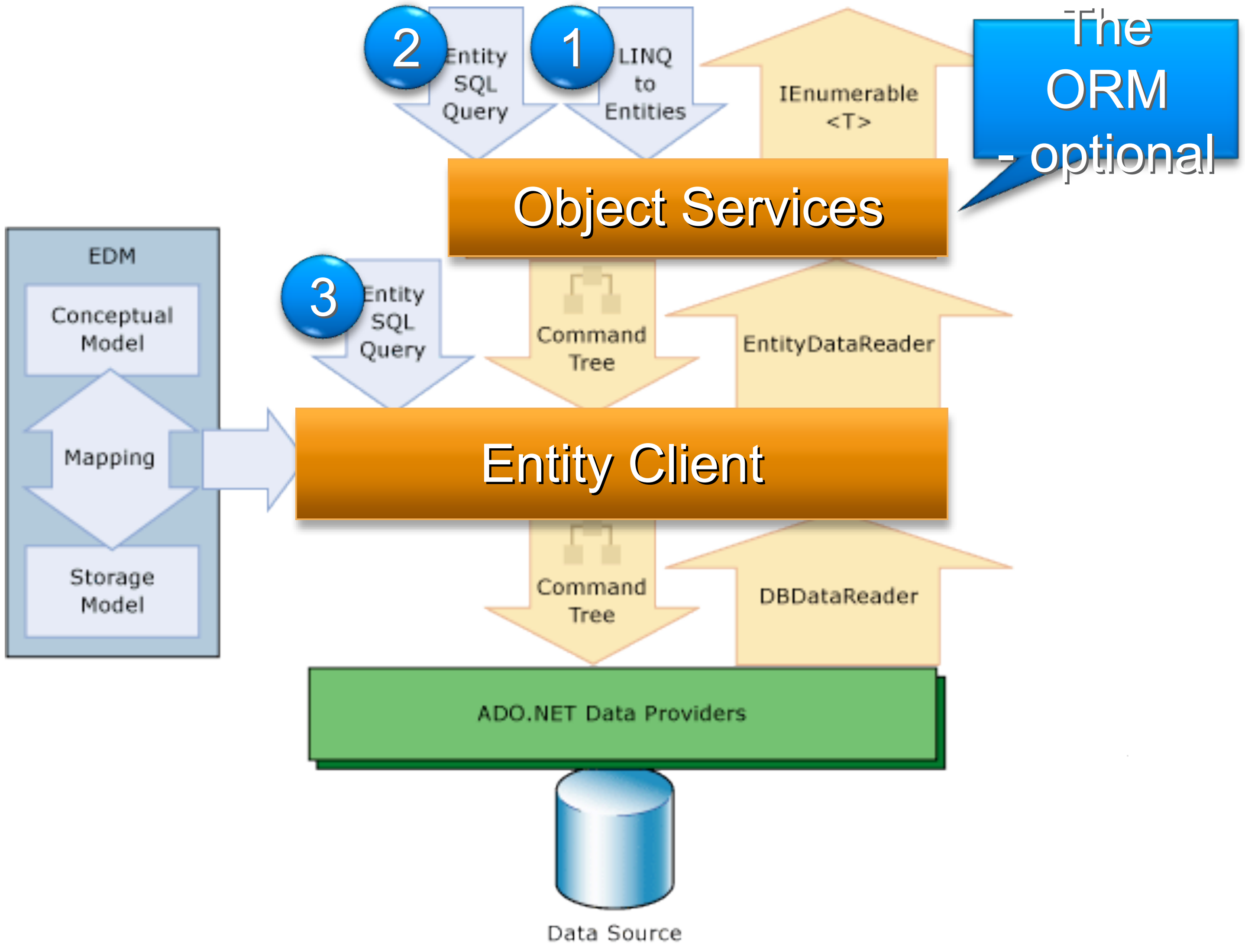
Mapping

Storage

{ A look at mapping }

demo

Consuming the EDM



Entity Client

- Familiar ADO.NET object model:
  - EntityCommand
  - EntityConnection
  - EntityDataReader
  - EntityParameter
  - EntityTransaction
- Text-based results
- Read-only
- Uses Entity SQL

Object Services

- Queries materialized as Objects
  - ObjectContext
  - ObjectQuery<T>
- Built on top of Entity Client
- Two query options:
  - Entity SQL
  - LINQ
- Runtime services:
  - Unit of work
  - Identity tracking
  - Eager/explicit loading

{ Consuming an EDM}

demo

Summary

ORM is an abstraction layer about data

- Productivity
- Database Independence

Microsoft has two ORMs

- LINQ to SQL – only SQL Server
- Entity Framework (includes LINQ to Entities) – many RDBMS, strategic

ADO.NET Entity Framework. Includes all the other bits ☺

- EDM, Linq to Entities, ESQL, Object Services
- Part of .NET Framework 3.5 SP1

Entity Data Model – mapping from conceptual to store

- CSDL = Conceptual Schema Definition Language
- SSDL = Store Schema Definition Language
- MSDL = Mapping Schema Definition Language

Two query languages

- ESQL = Entity SQL. Read only query language similar to SQL
- LINQ to Entities = Language Integrated Query to the EDM

The future?

Entity Framework v2	• .NET Framework 4.0 • http://blogs.msdn.com/efdesign
Entity Framework more widely adopted	• Across MS Products and Technologies • Reporting Services vNext
Rough edges removed	• Better ORM capabilities in LINQ to Entities • POCO support, N-Tier, Model First
Technologies work better together	• e.g. ASP.NET Dynamic Data and Entity Framework
Better tooling	• Less xml editing during development

Conclusion

1990 to 2007 - Tables

- Data Access technology remained reasonably static
- Procedural, API access
- Surfaced the database schema
- No clear ORM choice

2008 – Objects

- LINQ – excellent addition to the .NET languages
- Entity Framework and the EDM – strategic investment
- Productivity leap

Resources

- http://geekswithblogs.net/IUpdateable
- Or
- http://iupdateable.com

Microsoft®

Your potential. Our passion.™