Store and Restore .NET Objects as XML

Paul D. Sheriff
BUSINESS SOLUTIONS ARCHITECT, FAIRWAY TECHNOLOGIES, INC.
www.fairwaytech.com psheriff@fairwaytech.com

Module Goals

- Serialize .NET object into XML
- Store to disk
- Deserialize XML into .NET object
- Control XML with attributes
- Using the DataContractSerializer
- Using the BinaryFormatter

XML Serialization

Use XmlSerializer class

Serialize to a stream
(memory, string, file, etc.)

Save to disk

Deserialize from file back to a
.NET object

Demo

Serialize and deserialize

Control Serialization Using Attributes

XML Serialization Attributes

XmlRoot("name", Namespace)

XmlAttribute("name")

XmlElement("name")

XmlArrayAttribute

Demo

Using attributes to control serialization

Demo

Nested objects and [XmlArray]

Extension Methods

XML Extension Methods

Create generic Serialize<T>

Create generic Deserialize<T>

Demo

Extension methods

Other Serialization Methods

DataContractSerializer

~10% faster than XmlSerializer

Must mark properties to serialize

Can serialize private properties

Does not support XML attributes

Order of elements matter when deserializing

Demo

DataContractSerializer

BinaryFormatter

Serializes private properties

Little to no control over serialization

Not portable to other systems other than .NET

Demo

BinaryFormatter

Summary

- Serialization is easy to accomplish
- Great for storing an object
- XmlSerialization allows full control over XML
- DataContactSerializer is good for Web Services
- BinaryFormatter should only be used within .NET applications

Coming up in the next module...

Download data and store as XML

Detect changes in database data