

# Building End-to-End Multi-Client Service Oriented Applications

Module 09

Client Access

*moving to the client side*



# Highlights

- **Client Contracts & Entities**
- **WCF Proxies**
  - SOAP Headers
- **Dependency Injection**
- **Service Factory**

# Client-side Contracts

- **Already established client-side Entities**
  - Used to deserialize SOAP message into
- **Business service contracts work with business entities**
  - Need client-side service contracts
  - Will look just like business contracts, but return client entities
  - Can also have “Task-based” async operations

# WCF Proxies

- **Wrap access to the Channel Factory**
  - Implement client service contract
- **Client uses them like normal objects**
- **Will put thread's identity name into SOAP header**
  - Remember service pulled it out
  - Custom base class will hide all the code
- **Exported through MEF**
  - Already abstracted to interface
  - DI-requirements already met

# SOAP Header Usage

- **Proxy classes use “OutgoingMessageHeaders” collection**
  - Remember, services used “IncomingMessageHeaders”
- **Need to put current identity’s Name into header**
  - Will get it from current thread principal
  - In a desktop app, will be identity of desktop user
  - In a web app, will be identity established through the authentication scheme used (forms authentication will use user logged into app)
- **Want this passed with every call so service can pull it out later**
  - Don’t want to explicitly declare it in every operation
  - Don’t want client to have to do it manually
  - Answer: custom base class !

# Service Factory

- **Abstract factory pattern (yet again)**
- **Allows obtainment of a proxy on-demand**
  - Injection of proxies may not accommodate all clients
  - MVC controller-actions are stateless
  - WPF ViewModels are stateful
  - Proxy may be left open unnecessarily
- **Factory class injected instead**
- **Same exact pattern used in data repository factory and business engine factory**

# Summary

- **Nicely encapsulated client layer**
  - Included client-side contracts and entities
  - WCF service proxies and factory
  - All set up for DI
- **Unit tests confirmed DI access and connectivity to host**
- **Any client can now reference the client projects and access services**

**End of module**