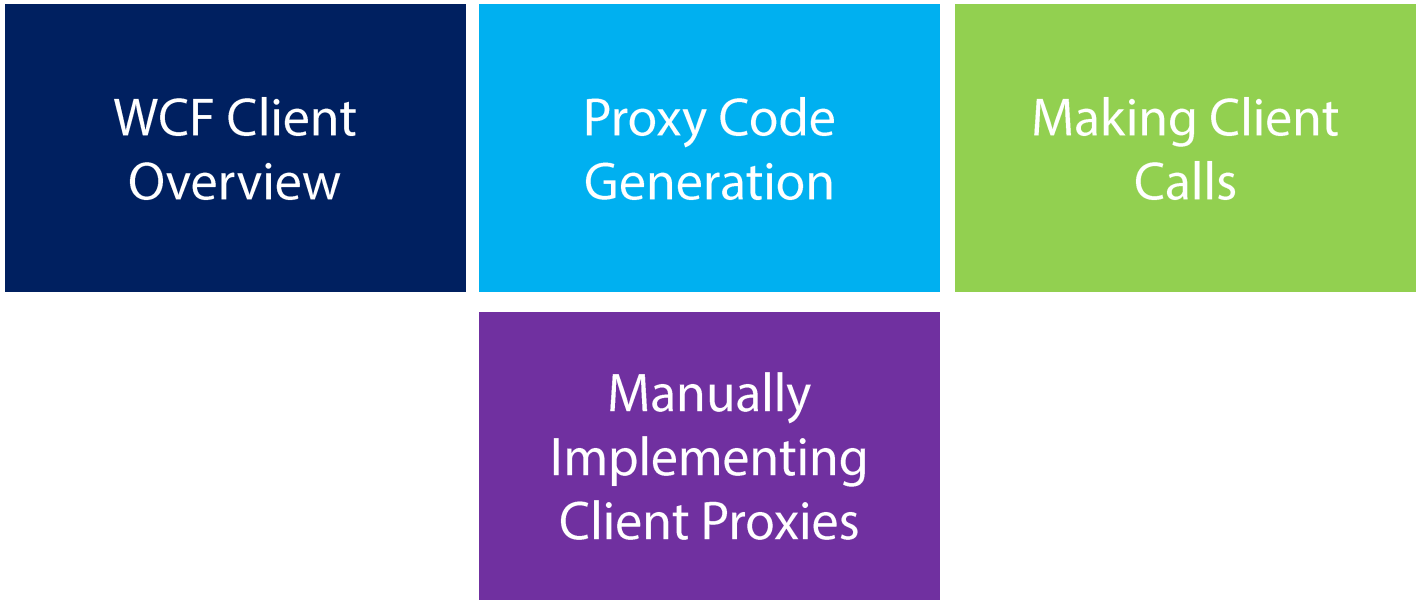
Implementing Clients

Brian Noyes
CTO, Solliance (www.solliance.net)
brian.noyes@solliance.net, @briannoyes

Outline


WCF Client Overview
Proxy Code Generation
Making Client Calls
Manually Implementing Client Proxies

WCF Client Overview

- **To consume WCF services from .NET clients, you need a client proxy**
- **Proxies can be code generated by Visual Studio based on metadata exposed by the service**
  - Service must enable metadata either through WSDL or WS-MetadataExchange endpoint
- **Proxies can also be hand-coded for more control over the code in the proxy**
  - Encapsulate repeating patterns of usage, leverage WCF extensibility features
- **Make calls through proxy instance methods**
  - Opens connection to the service
  - Dispatches the call through SOAP messages
  - Gets a response message back – completes method

Proxy Code Generation

- **Add Service Reference…**
- **Enter service address + Go, or Discover if service is auto-hosted in same solution**
- **Enter code generation child namespace**
- **Configure Advanced settings**
  - Async methods options
  - Types for collection and dictionary parameters and return types
  - Whether to use referenced assemblies in code generation process

Proxy Code Generation

- **Visual Studio generates**
  - Client side compatible service contract
  - Client proxy class
  - Data Contracts for parameters and return types if not already present in referenced assemblies
  - Client endpoint configuration compatible with service configuration
    - May need to tweak

Summary

- Add Service Reference makes implementing a client proxy simple
- Generated proxies have both synchronous and asynchronous methods
- Generated client configuration may need to be tweaked to work with service
- Manually implemented proxies may make sense to have more control over the proxy implementation