Workflow Architecture

Objectives

- **Why Workflow?**
- **Activities**
- **Workflows**
- **Messaging**
- **Correlation**
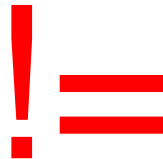
Business Processes

- **Business processes model real world activities**
  - Hiring new starter
  - Processing insurance claim
  - Commissioning new server
  - Stock tracking and ordering
- **Business processes have general requirements**
  - Auditable
  - Understandable by business users
  - Potentially long running
  - Non-linear flow, particularly when humans involved

Software Solutions

- **Software solves technical problems**
  - Reading and writing from databases
  - Sending data across networks
  - Performing complex calculations
  - Reacting to user input
- **Software has general requirements**
  - Maintainability
  - Performance
  - Re-use
  - Secure

Applying software to business processes

- **Mismatch between business process and technical solution**
  - Requires orchestration of technical details into business solution
- **Traditional approaches fail to meet business general requirements**
  - Requires large amounts of plumbing code and development process discipline

Applying Workflow to Business Problems

- **Workflow provides a framework to orchestrate technical solutions into a business solution**
  - Often associated with human involvement and work items
  - General model for all kinds of problems
- **Workflow can solve general business requirements**
  - Often graphical model allows audit and aids business user understanding
  - With persistence infrastructure can maintain state of long running process
  - Can model process flow based on external events

Windows Workflow Foundation

- **Windows Workflow Foundation (WF) is a framework for workflow based execution**
  - First introduced in .NET 3.0
  - Rewritten for .NET 4.0
  - Polished in .NET 4.5
- **Ships in a number of assemblies**
  - System.Activities
  - System.Runtime.DurableInstancing
  - System.ServiceModel.Activities

WF Concepts

- **Several core concepts in WF**
  - Activities
  - Workflows
  - Persistence
  - Messaging

Workflow and Services

- **Workflow powerful tool for building services**
  - declarative model provides clarity in processing
  - auto persistence infrastructure for long running stateful processes
- **Service composition simplified**
  - composing services can make it hard to understand the data flow between the different services
  - workflow's data flow and graphical designer make it simpler to understand how services are composed
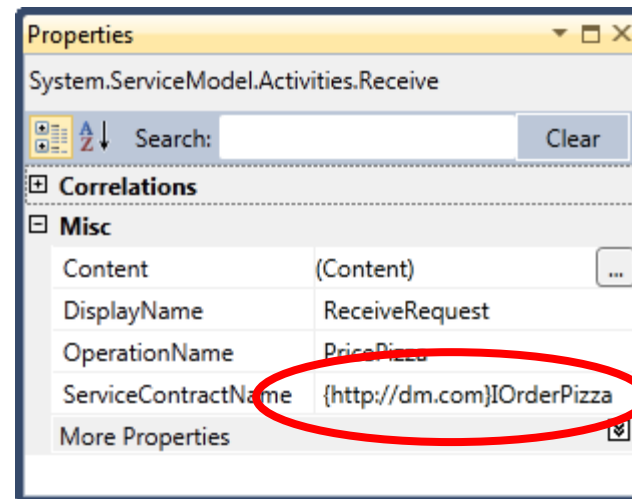
Creating Workflow Based Services

- **WF 4.0 now supports a fully declarative mode for authoring services**
  - services and endpoints can be configured in **XAML**
  - services defined in .xamlx file
  - contracts and endpoints still be declared in config

Service Activities

- **New activities for modelling service interaction**
- **Four new activities**
    - `Receive` – inbound one-way message
    - `ReceiveAndSendReply` – inbound request/response processing
    - `SendAndReceiveReply` – request/response outbound
    - `Send` – one-way outbound

Contract Inference

- **Contract is inferred from receives**
  - Service contract is string entered in property window
  - Operations inferred from receive operation names and parameters
  - Care must be taken to ensure service contract name exactly matches for all operations intended to be on same contract
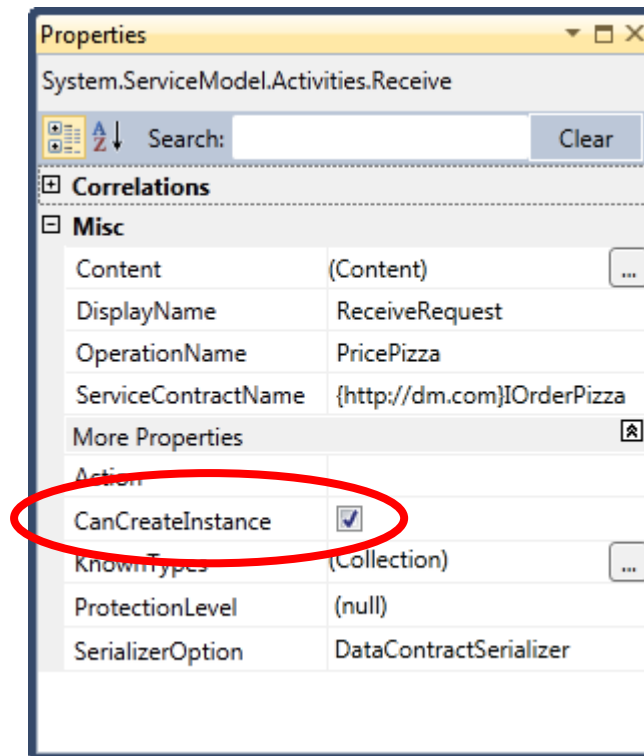
Importing Service Contracts

- **In 4.5 can import Service Contract**
  - Generates activities
  - Right click on project and import
- **Can Update Service Contract**
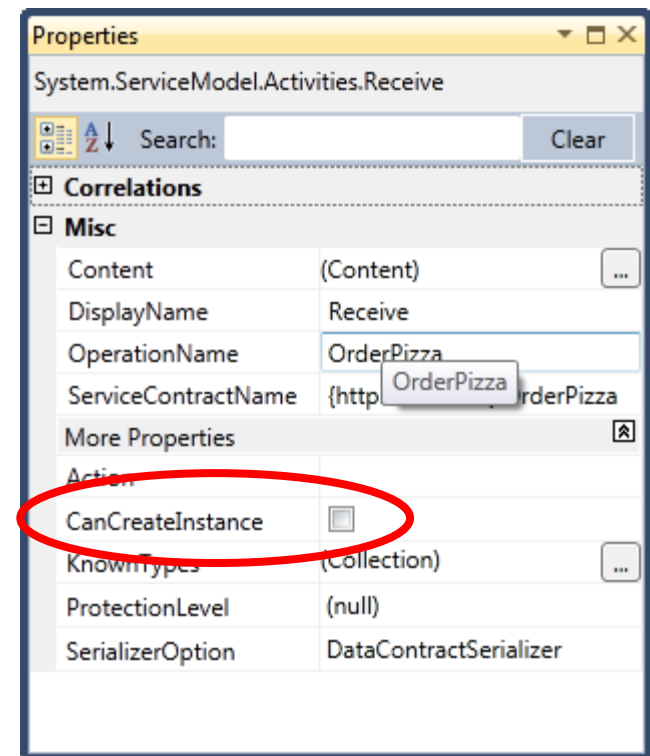 - Right click on contract in Service Contracts folder in project

CanCreateInstance

- **A receive often states that a message received creates a new instance of the workflow**

Long Running Execution

- **Services that run for extended periods have particular requirements**
  - state must be maintained: persistence
  - further requests must be directed to already running instance of workflow: correlation
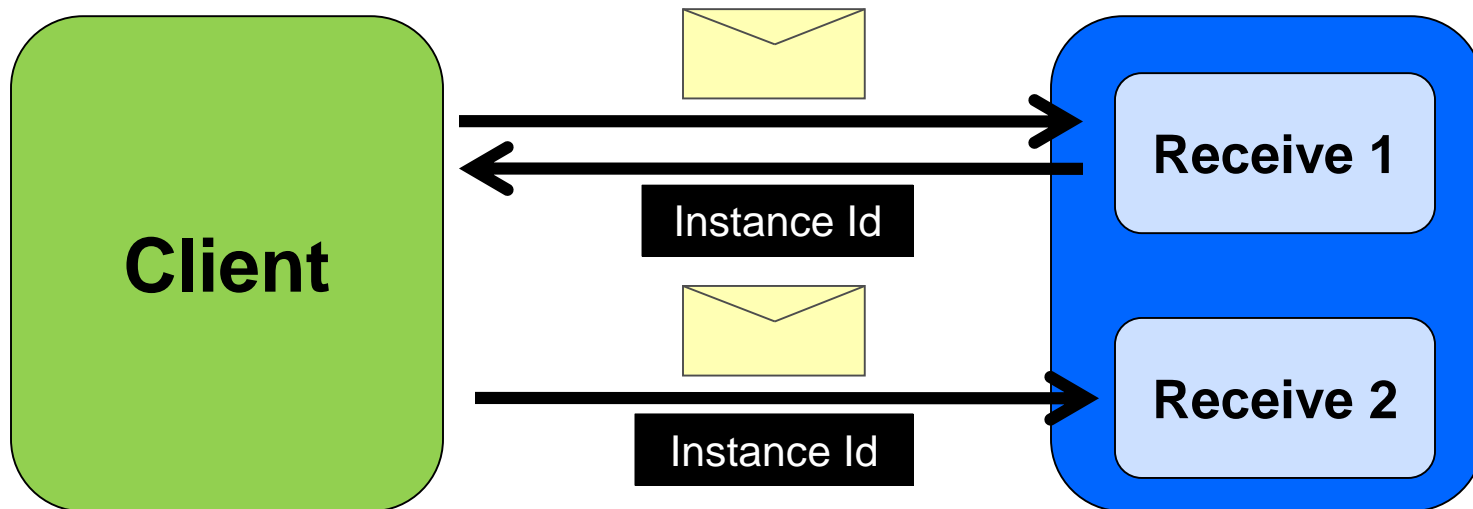- **Need non-creating receives**

Correlation

- **3.5 supported "context correlation"**
  - workflow instance id passed back to client
  - client sends workflow instance id in each request
  - handled automatically by ContextChannel
  - client needed to store and restore context between client restarts
- **WCF 4.0 supports context correlation and data correlation**
  - data correlation uses data in the message to map to the appropriate workflow instance –e.g. Invoice Number
  - same model as BizTalk

Context Correlation

- **ContextChannel manages the workflow instance id**
  - stored inside a message header or cookie
- **Must use special bindings containing the context channel**
  - WSHttpContextBinding
  - NetTcpContextBinding
  - BasicHttpContextBinding

Client

Instance Id

Instance Id

Receive 1

Receive 2

Managing the client context

- **Long running services require restartable clients**
- **Client can obtain and save context from channel property**
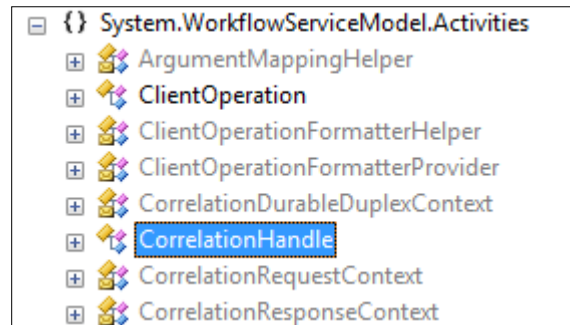
```
IChannel chan = (IChannel)client.InnerChannel;

IContextManager ctxMgr = chan.GetProperty<IContextManager>();

IDictionary<string, string> ctx = ctxMgr.GetContext();

WriteContextToStore(ctx);
```

- **Client can read and restore context using channel property**

```
IChannel chan = (IChannel)client.InnerChannel;

IContextManager ctxMgr = chan.GetProperty<IContextManager>();

IDictionary<string, string> ctx = GetContextFromStore();

ctxMgr.SetContext(ctx);
```
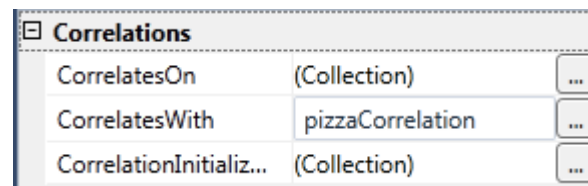
Data Correlation

- **Need a way to specify what data in the message indicates the destination workflow instance**
  - specified data must be unique for the lifetime of the workflow instance, e.g. Invoice Number
- **CorrelationHandle**
  - used to identify data to be used for a correlation
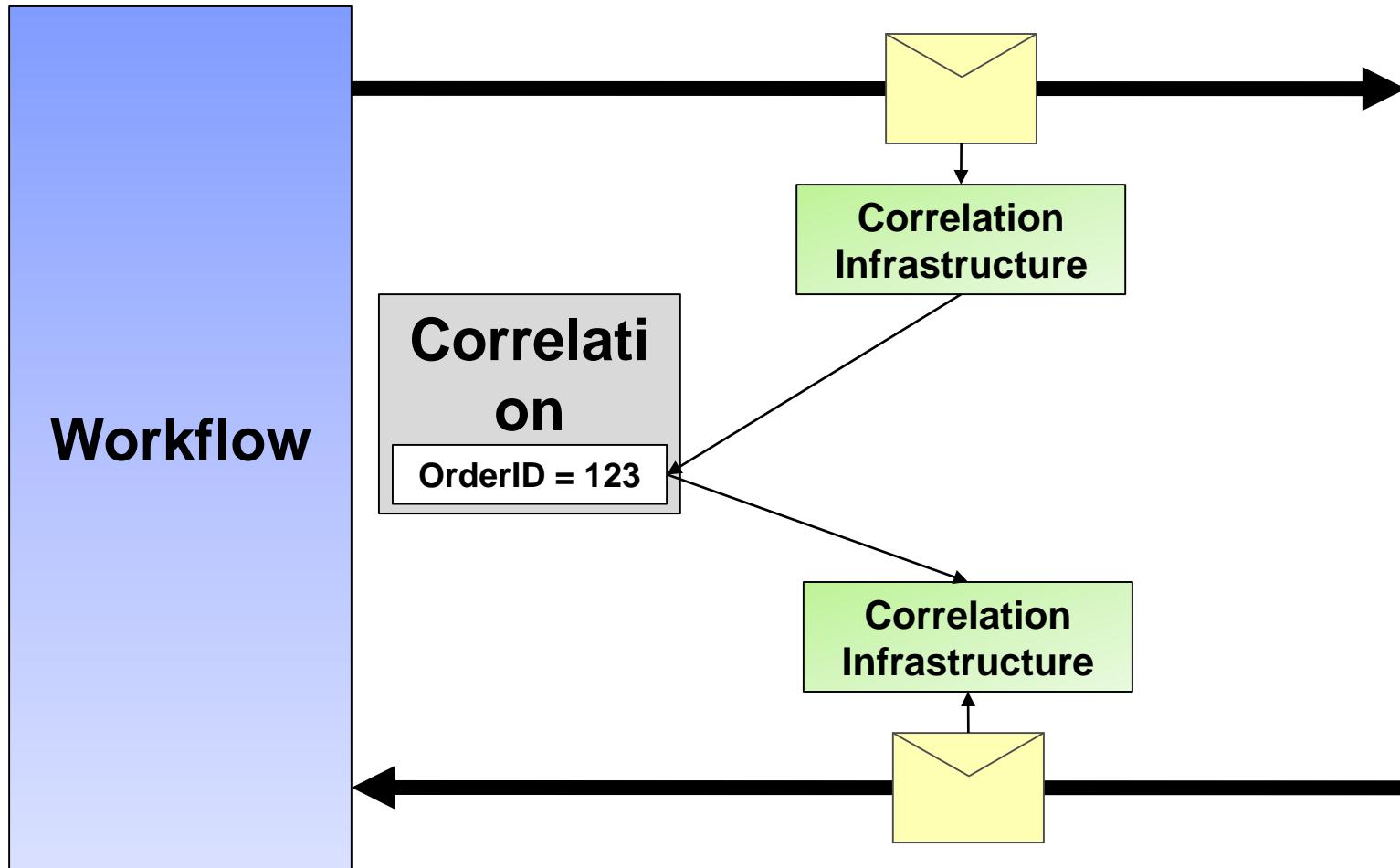 - single workflow instance may have a number of different correlations

Setting Up Data Correlation

1. **Create variable of type `CorrelationHandle`**
2. **Initialize correlation via `CorrelationInitializers` in receive or send**
   - specify XPath statements to the correlation data in the request or response message
3. **Specify same `CorrelationHandle` in one or more receive activties with `CorrelatesWith`**
   - specify XPath statements to the correlation data in this message using `CorrelatesOn`

Correlations		
CorrelatesOn	(Collection)	...
CorrelatesWith	pizzaCorrelation	...
CorrelationInitializ...	(Collection)	...

Data Correlation Example

Workflow

Correlation

OrderID = 123

Correlation Infrastructure

Correlation Infrastructure

Summary

- **Workflow bridges the technical world to the business world**
- **Includes a lot of out-of-the-box functionality**
- **Highly extensible**
- **Powerful tool for building services**
- **Activities generated by importing service contracts and WSDL**
- **Correlation essential for long running execution**