**Currency Rate Notify**

**Overview:**

In Publish/Subscribe pattern the subscriber tag each message with a key, the publisher send the message to all subscribers that have asked to receive a message for this key.

The communication between publisher and subscribers is done throw messages.

**Solution Structure:**

1. Service

The service application conations the business logic code and the WCF contract implementations.

Subscribe Service:

To handle client subscription and unsubscribing scenarios.

Publish Service:

To handle sending an update to all subscribers.

Publisher Subscriber Hub:

Maintain the list of subscribers, Return subscribers by key, allow client to subscribe and unsubscribe.

1. Model

The Model applications conations all the message definition, WCF contract interfaces.

1. Host

The host application is just a console application that act as a hosting for the WCF services

1. Client

The client section include two application one application act as a publisher and the other application act as a subscriber, multiple instances of the subscriber could be opened to listen for same or different events.

**Current implementation issues**

* One or multiple of the communicating entities might be down.
* The message will be lost if the subscriber is not up and running.

**Further Enhancements**

* Rely on a durable storage to send the messages to and whenever the service is up and running the service will resume its work.
* Using a mechanism like MSMQ, which will grantee message delivery, enhance lose coupling and scalability.
* Hosting the WCF service on more mature environment like IIS instead of self-hosting.
* Configure the WCF with different endpoints to allow clients to communicate with different protocols.

**Others**

The solution is built using Visual Studio 2015, framework version 4.5.2