Windows Communication Foundation: Hacia una Arquitectura Orientada a Servicios (SOA) más simple

Claudio Zamoszczyk*

Arquitectura orientada a Servicios (SOA)

SOA es un estilo arquitectónico de software, no una tecnología determinada. Como estilo **arquitectónico**, define a los servicios como unidades de partición para dar respuesta a los requerimientos del negocio dentro de una organización.

SOA proporciona una metodología y un marco de trabajo para mapear procesos de negocio a componentes de software en forma de servicios. Las aplicaciones ya no son más elementos aislados sino parte integrante del negocio.

Tecnológicamente, estos servicios tienen la particularidad de ser:

- encapsulados
- de bajo acoplamiento
- reutilizables
- sin información de estado (stateless)
- localizables
- independientes de una plataforma o lenguaje
- basados en estándares y tecnologías ya utilizadas (SOAP, UDDI, WSDL, http, entre otros)

Introducción a WCF (La respuesta de Microsoft a SOA)

Windows Communication Foundation o WCF es la nueva plataforma de comunicación y mensajería que forma parte del ya conocido entorno de trabajo .NET (en su versión 3.0)

La idea fundamental de WCF es la de permitir a las empresas desarrollar aplicaciones distribuidas basadas en la arquitectura orientada a servicios (SOA), donde éstas se puedan ejecutar, desde una máquina local hasta Internet, de manera simple y segura.

WCF		
Interoperabilidad	Productividad	Desarrollo Orientado a Servicios
- Soporte de las diferentes especificaciones basadas en Web Services (Seguridad, Transporte, etc.) - Compatible con tecnologías de programación distribuida de Microsoft	- Integración con Visual Studio 2005 - Unificación de tecnologías distribuidas ya existentes - API de fácil manejo	- Servicios alineados a la lógica de negocios independiente y reutilizables

* Docente de la Facultad de Ingeniería - UP.

La clave fundamental es la sencillez del API de desarrollo y su fácil utilización dentro de la plataforma .NET Framework; permitiendo utilizar cualquier medio de conectividad, ya sea por XML, HTTP SOAP, etc.

El modelo de programación de WCF plantea una unificación de Servicios Web, .NET Remoting, Transacciones Distribuidas y mensajería MQ en un solo marco orientado a servicios distribuidos. Esto no es nada nuevo desde el punto de vista de SOA y de las tecnologías relacionadas, sólo facilita su desarrollo en comparación con otras plataformas que brindan las mismas posibilidades como es el caso de IBM u Oracle. Permitiendo tener ciclos de desarrollo más cortos sin la necesidad de contar con expertos en tecnologías de integración.

Lo que hay que entender de primera mano es que WCF surge dentro del marco de la iniciativa Windows Vista, dejando de lado la posibilidad de desarrollar aplicaciones para otras plataformas, no así la conectividad y la interoperabilidad que continúan siendo independientes de la tecnología y la plataforma gracias a los Servicios Web (SOAP) y demás estándares.

Componentes básicos de WCF (DÓNDE, CÓMO y QUÉ)

En primer lugar, el desarrollo con WCF se basa en servicios. Un servicio es un programa que expone una colección de Endpoints (puntos de acceso). Cada Endpoint, es una puerta hacia el mundo exterior mediante una determinada tecnología de transporte.

Un Cliente es un programa que intercambia mensajes con uno o más Endpoints. Y un Cliente puede exponer también un Endpoints para recibir mensajes de un Servicio.

Un Endpoint de un Servicio está compuesto por una Address (Dirección), un Binding (Enlace) y un Contract (Contrato).

Definición de un Servicio

```
ServerHelloWorld.HelloWorldService
using System;
using System.Collections.Generic;
using System.Text;
using System.ServiceModel;
namespace ServerHelloWorld
{

    [ServiceContract]
    class HelloWorldService
    {


        [OperationContract]
        string SayHello()
        { return "Hello World!!!!";}

    }
}
```

La dirección de un Endpoint es una dirección de red donde reside dicho Endpoint. El Binding específica cómo se comunica dicho Endpoint con el resto del mundo, incluyendo aspectos como el protocolo de transporte (TCP, http, etc.), tipo de codificación (texto, binario), y requerimientos de seguridad (SSL, seguridad basada en mensajes SOAP, etc.).Por último el contrato especifica qué comunica dicho Endpoint y básicamente está compuesto por una colección de mensajes organizados internamente en operaciones que tienen un patrón de intercambio de mensajes (Message Exchange Patterns ó MEPs), como one-way (un sentido), dúplex y request/reply (petición/respuesta).

Registración y publicación del servicio

```
ServerHelloWorld.Program                                   Main(string[] args)
using System;
using System.Collections.Generic;
using System.Text;
using System.ServiceModel;
namespace ServerHelloWorld
{
    class Program
    {
        static void Main(string[] args)
        {

            Uri uri = new Uri("http://localhost/HelloWorldService");
            ServiceHost serviceHost = new ServiceHost(typeof(HelloWorldService), uri);

            serviceHost.AddServiceEndpoint(typeof(HelloWorldService), new BasicHttpBinding(), uri);

            serviceHost.Open();

            Console.Write("Service is ready :)");


            Console.ReadKey();
            serviceHost.Close();

        }
    }
}
```

Desarrollo con WCF

Para que los desarrolladores puedan crear aplicaciones utilizando WCF deben contar primordialmente con el sistema operativo Windows Vista (ya que inicialmente fue pensado para éste), aunque también puede utilizarse con Windows XP y Windows 2003 Server. Luego para la compilación y ejecución es necesario contar con el .NET Framework 3.0. Y por último, desde el punto de vista de la codificación, es necesario contar con Microsoft Visual Studio 2005 con las extensiones correspondientes a la plataforma .NET 3.0.

¿Por qué usar Windows Communication Foundation?

Cuando hay que decidir qué tecnología usar para determinado sistema distribuido nos encontramos con varias opciones: Servicios Web XML con extensiones WSE, Microsoft Queue (MSMQ), el ambiente de ejecución Enterprise Services/COM+, y .NET Remoting. Escoger entre una u otra depende de qué se necesite más -seguridad,

interoperabilidad, performance, transacciones. Y escoger una de las primeras tecnologías mencionadas conlleva a sacrificar seguridad, performance, u otros aspectos del desarrollo. Estos inconvenientes nos plantean el porqué de la utilidad de WCF:

Cuando sea necesario crear aplicaciones conectadas: Actualmente la conectividad es la primera consideración en la arquitectura de un Sistema Distribuido. La orientación a servicios ayuda a lograrlo. WCF se diseñó desde sus inicios con la orientación a servicios en la mente.

Modelo de programación unificado: Con frecuencia los desarrolladores de aplicaciones tienen que escoger entre múltiples tecnologías, cada una con un único modelo de programación; una tecnología para llamar a procedimientos remotos, otra para queuing, otra para componentes transaccionales y otras para lograr interoperatividad entre plataformas. WCF provee un solo modelo de programación para la comunicación.

Requisitos amplios de interoperabilidad: La interoperatividad entre plataformas se ha vuelto una necesidad obligatoria de todas las aplicaciones conectadas. WFC usa los protocolos estándares de la industria que permiten a éste interoperar con cualquier plataforma que soporte estos protocolos.

Mirada al futuro

Es evidente que la idea fundamental de WFC es poder desarrollar aplicaciones SOA de manera simple, rápida y segura. Planteando un modelo de desarrollo práctico en contrapartida a otras implementaciones de SOA sobre Java, como es el caso de IBM y Oracle, que cuentan con entornos de desarrollo más complicados y más difíciles de utilizar. Donde el tiempo de aprendizaje e implementación es mucho más costoso.

Lo que sí queda en claro es que las soluciones SOA utilizando Java se basan en herramientas y tecnologías abiertas y permiten su desarrollo y ejecución en cualquier entorno, no sólo dentro del mundo Microsoft Windows.

Aún falta mucho tiempo para poder observar los frutos del desarrollo basado en SOA, pero sin lugar a duda WFC ayudará a que esto sea logrado con éxito.