

## 2. DESARROLLO DEL PROYECTO

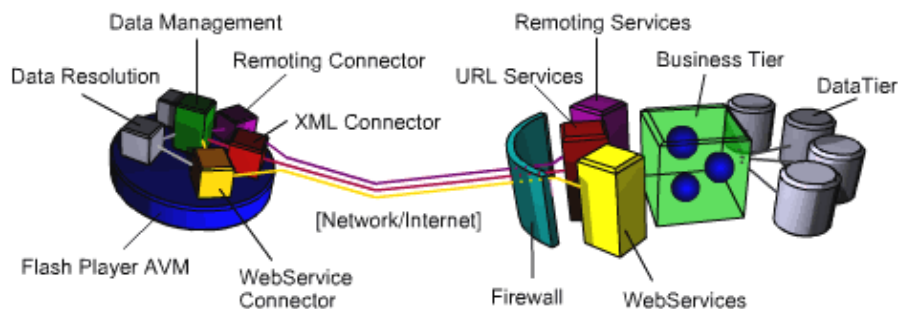
### 2.1 DEFINICION DEL DISEÑO DE INTERACCION DE ARQUITECTURAS

#### 2.1.1 Descripción de la arquitectura Macromedia Flash data integration

Macromedia Flash MX Professional 2004 integra los diferentes datos soportados por la aplicación (wsdl, bases de datos, XML), utilizando diferentes capas con un propósito específico para cada tipo de dato.

Ofrece entonces la capacidad para conectarse con Web services, servicios XML y servicios Remotos, utilizando las diferentes facilidades suministradas por los componentes del entorno de desarrollo a través de las APIs (repositorio de código) genéricas junto con la *capa de conexión de datos*.

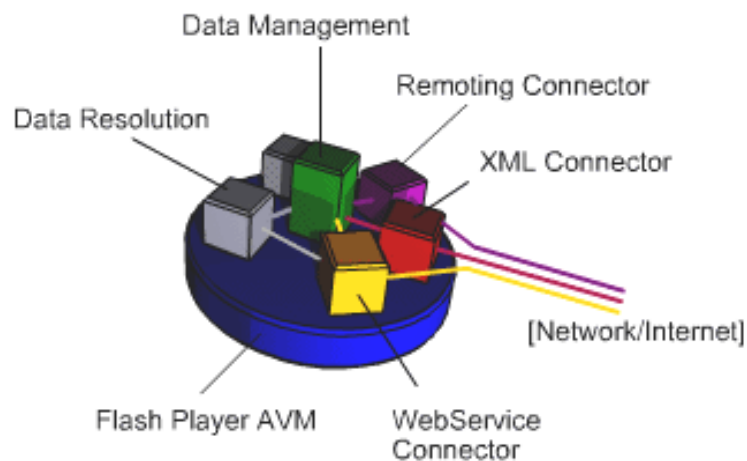
La *capa de manejo de datos* es utilizada para el manejo de los datos del cliente por medio del componente DataSet (Ver Figura 3), el cual permite la conexión de la Película Flash con bases de datos, además cuenta con la capacidad de interpretar actualizaciones en las base de datos XML usando la *capa de resolución de datos* y facilidad de integrar todas las capas y los servicios por medio de un ambiente visual ofrecido por la aplicación.



**Figura 3.** Conexión entre Flash data integration architecture con una fuente de datos remotos (Tomado de la página oficial de Macromedia Flash MX, <http://www.macromedia.com/la/software/flashremoting>)

### 2.1.1.1 Conectividad de los Datos

La capa de conectividad en Flash MX Professional 2004 ofrece la habilidad de conectarse a orígenes de datos externos y obtener datos al enviarlos. Esta funcionalidad le proporciona a los desarrolladores la posibilidad de conectarse a estos orígenes de datos a través del componente llamado WebService Conector (Ver Figura 4). Para conectarse a servicios Web SOAP, el componente XMLConnector ofrece la capacidad de conectarse a cualquier origen de datos externo que retorne un XML a través del protocolo http (tal como: JSP, ASP, Servlet, o ColdFusion), y Flash Remoting, para conectarse directamente a las aplicaciones por medio de un protocolo binario eficiente.

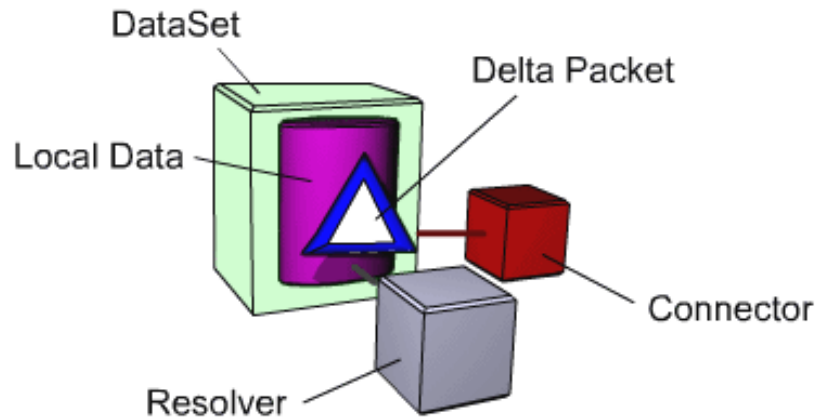


**Figura 4.** Flash Data Integration Architecture (Tomado de la página oficial de Macromedia Flash MX, <http://www.macromedia.com/la/software/flashremoting>)

### 2.1.1.2 Manejo de datos

La capa de Manejo de datos provee la funcionalidad para manejar los datos entre las aplicaciones Flash usando el componente DataSet (Ver Figura 5).

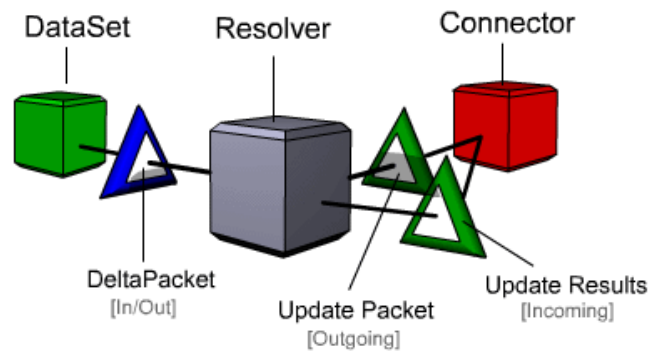
Este manejo de datos se refiere a la habilidad de realizar operaciones avanzadas en los datos del cache local, pudiendo realizar operaciones de organización de los datos, además los cambios que se realicen por medio de la Página Web y los componentes serán seguidos para generar un optimizado set de instrucciones (DeltaPacket) que son usados para actualizar base de datos externos.



**Figura 5.** Arquitectura de los componentes de manejo de datos de Macromedia Flash MX 2004 (Tomado de la página oficial de Macromedia Flash MX, <http://www.macromedia.com/la/software/flashremoting>).

### 2.1.1.3 Resolución de datos

La resolución de datos en Flash MX Professional 2004 ofrece la codificación y la decodificación de paquetes de datos que representan los cambios entre la aplicación Flash del cliente y las fuentes de datos externas en un formato que puede ser entendido por ambas partes (Ver Figura 5). Esta funcionalidad es suplida por los componentes que utiliza el desarrollador para resolver los datos.



**Figura 5.** Arquitectura de los componentes de resolución de datos de Macromedia Flash MX 2004 (Tomado de la página oficial de Macromedia Flash MX, [http://www.macromedia.com/ la/software/flashremoting](http://www.macromedia.com/la/software/flashremoting))

### 2.1.2 Descripción de la arquitectura Microsoft .Net

La plataforma .Net Provee las bases para la nueva generación de software. Utiliza los Servicios Web como un medio para poder interoperar entre distintas tecnologías. Permite conectar distintos sistemas operativos, dispositivos físicos, información y usuarios. Les da a los desarrolladores las herramientas y tecnologías para hacer rápidamente soluciones de negocios que involucran distintas aplicaciones, dispositivos físicos y organizaciones (Ver Figura 6).

## Que es la Plataforma .NET



**Figura 6.** Diagrama Global de la Plataforma .Net (Tomado de la página oficial de Microsoft.  
<http://msdn.microsoft.net>).

La idea central detrás de la plataforma .NET es la de servicio. Más concretamente *software como servicio* y de cómo construir, instalar, consumir, integrar o agregar estos servicios para que puedan ser accedidos mediante Internet. El usuario de Internet puede con un explorador de Internet no solamente acceder a contenido como texto, imágenes o sonido, también puede hacer uso de servicios Web. Estos son los bloques de construcción o componentes sobre los cuales se basa el modelo de computación distribuida en Internet. La plataforma .NET permite usar Internet y su capacidad de distribución para que los usuarios accedan desde cualquier dispositivo, en cualquier sistema operativo y lugar, a la funcionalidad que los servicios Web proveen.

### 2.1.2.1 Componentes de la plataforma .NET.

La plataforma .NET no es un solo producto. Es un conjunto de productos. Desde sistemas operativos como Windows XP, servidores de aplicaciones como SQL Server 2000, productos de oficina como Office XP, herramientas de desarrollo

como Visual Studio .NET hasta servicios Web provistos por Microsoft como .NET Passport.

Tanto la invocación de los servicios como su ejecución pueden ser hechas en cualquier dispositivo y sistema operativo, y accedido desde Internet. Los sitios se comunican entre sí y acceden a servicios y contenidos sin la intervención humana.

**'Smart Clients' (Clientes Inteligentes):** Son dispositivos muy variados. Lo que los hace 'Smart' o inteligentes es su capacidad para hacer uso de servicios Web.

**PCs:** Las computadoras personales.

**NoteBooks:** Las computadoras portátiles.

Y muchos otros dispositivos en desarrollo. Además:

**Servidores:** Proveen de la infraestructura para implementar el modelo de computación distribuida en Internet. Son sistemas operativos y de aplicación.

**Sistemas Operativos:** Windows 2000: Server, Advance Server y Datacenter, Windows Server 2003: Standard, Enterprise, Datacenter y Web Server.

#### **Servidores .NET Corporativos:**

- **Microsoft Application Center 2000:** Para instalar y administrar aplicaciones Web altamente disponibles y escalables.
- **Microsoft BizTalk Server 2000:** Para construir procesos de negocios basados en XML a través de distintas aplicaciones y organizaciones.
- **Microsoft Commerce Server 2000:** Para construir rápidamente soluciones de e-commerce escalables.
- **Microsoft Content Management Server 2001:** Para administrar contenido para sitios Web de e-bussines dinámicos.
- **Microsoft Exchange Server 2000:** Para permitir enviar mensajes y trabajar en forma colaborativa en cualquier momento y lugar.

- **Microsoft Host Integration Server 2000:** Para acceder a datos y aplicaciones en mainframes.
- **Microsoft SQL Server 2000:** Para almacenar, recuperar y analizar datos en formato XML.
- **Microsoft SharePoint Portal Server 2001:** Para encontrar, compartir y publicar información de negocios.
- **Microsoft Internet Security and Acceleration Server 2000:** Para conectividad a Internet rápida y segura.
- **Microsoft Mobile Information 2001 Server:** Para soportar aplicaciones en dispositivos móviles, como por ejemplo celulares.

**Servicios Web basados en XML:** Son los bloques de construcción de la tercera generación de Internet. Algunas de sus características son:

- **Permiten a las aplicaciones compartir datos:** Son componentes. Es decir, unidades de código discretas, cada una haciendo una tarea en particular.
- **Están basados en el lenguaje universal de intercambio de datos de Internet:** Pueden ser llamados desde distintos sistemas operativos, plataformas de hardware y lenguajes de programación.

#### **2.1.2.2 Herramientas de desarrollo**

Visual Studio .NET y el .NET Framework. Ambos permiten al desarrollador hacer servicios Web basados en XML además de otro tipo de aplicaciones. El .NET Framework viene incorporado directamente en la nueva línea de sistemas operativos Windows .NET. Para los dispositivos móviles se llama .NET Compact Framework.

Es claro entonces que el objetivo de la plataforma .NET es simplificar el desarrollo de aplicaciones Web. Provee las herramientas y tecnologías para transformar a Internet en una plataforma de computación distribuida en gran escala. Esta plataforma además soporta los estándares sobre los cuales se basan los servicios Web.

### 2.1.2.3 El .NET Framework

Es un conjunto de servicios de programación diseñados para simplificar el desarrollo de aplicaciones en el entorno altamente distribuido de Internet. El *.NET Framework* se instala como un componente aparte en Windows 2000, mientras que Windows XP y las futuras versiones de Windows lo incorporan directamente al sistema operativo, como por ejemplo Windows Server 2003 o Windows .NET CE. El *.NET Compact Framework* permite hacer uso de los servicios Web en dispositivos móviles. Debido a que es un subconjunto del .NET Framework, comparte el mismo modelo de programación y herramientas de desarrollo de aplicaciones (Ver Figura 7) haciendo posible que los desarrolladores transfieran sus conocimientos existentes al desarrollo de aplicaciones móviles.



**Figura 7.** Componentes del .Net Framework (Tomado de la página oficial de Microsoft.  
<http://msdn.microsoft.net>)



Los componentes del .NET Framework proveen los "ladrillos" necesarios para construir las aplicaciones Web, los servicios Web y cualquier otra aplicación dentro de Visual Studio .NET.

#### 2.1.2.4 Componentes de la arquitectura:



**Figura 8.** Elementos que Componen el .Net Framework (Tomado de la página oficial de Microsoft. <http://msdn.microsoft.net>).

El **Common Language Runtime** provee lo que se llama código administrado, es decir, un entorno que provee servicios automáticos al código que se ejecuta. Los servicios son variados:

**Cargador de Clases:** Permite cargar en memoria las clases.

**Compilador MSIL a nativo:** Transforma código intermedio de alto nivel independiente del hardware que lo ejecuta a código de máquina propio del dispositivo que lo ejecuta.

**Administrador de Código:** Coordina toda la operación de los distintos subsistemas del Common Language Runtime.

**Recolector de Basura:** Elimina de memoria objetos no utilizados.

**Motor de Seguridad:** Administra la seguridad del código que se ejecuta.

**Motor de Depuración:** Permite hacer un seguimiento de la ejecución del código aún cuando se utilicen lenguajes distintos.

**Verificador de Tipos:** Controla que las variables de la aplicación usen el área de memoria que tienen asignado.

**Administrador de Excepciones:** Maneja los errores que se producen durante la ejecución del código.

**Soporte de multiproceso (threads):** Permite ejecutar código en forma paralela.

**Empaquetador de COM:** Coordina la comunicación con los componentes COM para que puedan ser usados por el .NET Framework.

**Soporte de la Biblioteca de Clases Base:** Interfaz con las clases base del .NET Framework.

### 2.1.3 Descripción de la relación e interacción entre las arquitecturas



**Figura 9:** Interacción de los elementos de la arquitectura.

Para la integración de ambas arquitecturas, Macromedia Flash Data Integration y la arquitectura Microsoft .Net, se cuenta con varios elementos que permiten realizar el transporte e interpretación de datos entre las capas de ambas arquitecturas (Ver Figura 9).

La interacción de las dos arquitecturas consiste en la implementación de: Clases en lenguajes como actionscript y c#, Webservices creados en .Net, Almacenamiento de datos y Uso de Componentes de Conexión existentes en Macromedia Flash. Estos últimos pueden ser usados por la aplicación flash para la manipulación tanto del contenido textual, comportamiento animado y la transaccionalidad de datos de la película flash, que una vez creada permite ser controlada y actualizada de manera constante y simple por medio de un administrador de contenido cada vez que sea necesario.

Se utiliza una base de datos para el almacenamiento del contenido e información del comportamiento de la película flash.

La conexión entre la base de datos y la película es realizada por medio de Web services, que serán creados y estarán en funcionamiento, los cuales son una serie de objetos que pueden ser llamados para realizar acciones específicas y que pueden ser invocados por diferentes aplicaciones clientes servidor para un uso específico

#### **2.1.3.1 Framework (Invocado por la película flash):**

El framework consiste en una serie de clases desarrolladas en lenguajes como actionscript y c#, la cual se utilizará para diseñar y poner en funcionamiento un solo paquete de herramientas para la ejecución. Por otra parte, al diseñar el paquete de herramientas como framework, se permitirá generar una colección de las clases y métodos para ser utilizadas en una variedad de aplicaciones o películas Flash. El framework debe describir el comportamiento de las

aplicaciones Flash para un uso diversificado. Por lo tanto, debe tener la posibilidad de modificar los requisitos particulares para ajustarse a la aplicación.

### **2.1.3.2 Webservices**

El servicio Web será una entidad programable que proporcionará un elemento de funcionalidad determinado, como lógica de aplicación, al que se puede tener acceso desde diversos sistemas potencialmente distintos, mediante estándares de Internet muy extendidos, como XML y HTTP. Los servicios Web dependen en gran medida de la aceptación de XML y otros estándares de Internet para crear una infraestructura que posibilite el funcionamiento conjunto de aplicaciones que proveen un servicio específico.

Un servicio Web puede ser utilizado internamente por una aplicación o bien ser expuesto de forma externa en Internet por varias aplicaciones. Dado que a través de una interfaz estándar es posible el acceso a un servicio Web, éste permite el funcionamiento de una serie de sistemas heterogéneos como un conjunto integrado de servicios.

### **2.1.3.3 XML Serializado**

Este objeto XML recibido por la aplicación Flash contiene los datos ingresados en la base de datos por medio del administrador de contenido, utilizando el Webservice como intermediario para comunicar las dos arquitecturas (Flash y .Net).

El XML es serializado desde .Net por medio de sus clases de manipulación de objetos XML y acceso a Bases de Datos, permitiendo que el objeto XML pueda ser utilizado por diferentes arquitecturas con un estándar internacional como lo es el XML.

#### **2.1.3.4 Administrador de Contenido**

El administrador será una aplicación Web desarrollada en Microsoft ASP .Net, la cual permitirá administrar, como su nombre lo indica, el contenido y las acciones de la aplicación, de igual manera tendrá la posibilidad de administrar la información ya almacenada anteriormente en la base de datos para que se actualice automáticamente en la aplicación Flash, una vez sea requerida.

El administrador generará registros en la base de datos de Contenido con información correspondiente a las diferentes películas. Una vez se haya ingresado la información de la aplicación, el usuario final podrá tener acceso a la configuración y la presentación de una manera simple y funcional.

#### **2.1.3.5 Webservice conector en Aplicación Flash**

Este componente será usado por la aplicación flash para comunicarse directamente con el Webservice y así obtener información consultada sobre la base de datos por medio de la arquitectura .Net.

Cada vez que sea necesario actualizar la información del contenido de la película, se deberá hacer una solicitud por parte de este componente hacia el Webservice, pidiendo la ejecución de cualquiera de sus métodos Web (Métodos para ser usados como Servicio) para así poder interactuar directamente sobre los datos almacenados en la Base de Datos.