

Microsoft®
Mediaroom™



Microsoft Mediaroom
Presentation Framework
**Building Differentiated
Experiences 2.0**

February, 2010

Learn more about the Microsoft Mediaroom™ platform at <http://www.microsoftmediaroom.com>

This is a preliminary document and may be changed substantially prior to final commercial release of the software described herein. The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

This document is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS DOCUMENT.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

©2010 Microsoft Corporation. All rights reserved. Microsoft, Mediaroom, and Visual Studio are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. All other trademarks are property of their respective owners.

Contents

Introduction	1
The Microsoft Mediaroom Presentation Framework	1
Building Microsoft Mediaroom PF Applications	2
Building with Web Services	4
The Microsoft Mediaroom Simulator	5
Applications and Layers	4
Physics-based controls	4
Parental Controls	5
Launching PF Applications	5
Protecting the Production Environments	6
Creating Exciting New Subscriber Experiences	7
Summary	8

Introduction

The Microsoft® Mediaroom™ Presentation Framework (PF), included with the Microsoft Mediaroom 2.0 client and Microsoft Mediaroom Application Development Kit (ADK) 3.0 releases, is an open platform that enables developers to build differentiated TV applications that extend and augment the Microsoft Mediaroom TV experience. This white paper provides an overview of the Microsoft Mediaroom PF and the kinds of applications that service providers and developers can quickly and cost-effectively deploy to create a new world of experience for subscribers. With Microsoft Mediaroom and the Microsoft Mediaroom PF, TVs can now be as connected as PCs to messaging, social networking, program fan sites, sports sites, and dozens of other services, taking the subscriber's experience of TV to whole new level.

The Microsoft Mediaroom Presentation Framework

The Microsoft Mediaroom PF represents the natural evolution of the IPTV application platform. It delivers the building blocks necessary for creating innovative, robust, and intuitive user experiences for TV. Service providers and third party partners can build exciting applications quickly and easily using Web services to incorporate streaming Web video, RSS and other data feeds, and many other elements. This content can be blended with the existing core user interface as well as broadcast, on-demand, or digitally recorded video streams. Service providers and third party developers can create applications that integrate seamlessly with the underlying content and interface, or they can create applications that overlay or replace video. Either way, a service provider can deploy Microsoft Mediaroom PF applications without having to update either Microsoft Mediaroom servers or clients. Indeed, other than a Microsoft Mediaroom-compliant set-top box, no special hardware or chipsets are required in a client device for a subscriber to experience a Microsoft Mediaroom PF-based application.



Figure 1 A Microsoft Mediaroom PF application rendering content from social and public sites

Microsoft Mediaroom PF applications can run on the servers where the pertinent data and content resides or they can make remote calls to external servers to deliver content to viewers. Figure 1, for example, shows a Microsoft Mediaroom PF application developed by Accedo Broadband that highlights publicly available Internet

content that can be pulled from a site such as Twitter in real time. Microsoft Mediaroom PF applications can mash up data from Microsoft Mediaroom clients and servers, the Internet, and private data sources—all in a single application environment—to enable a rich and differentiating experience for subscribers.

Building Microsoft Mediaroom PF Applications

Microsoft Mediaroom PF applications are essentially ASP.NET applications. As such, they can build upon your existing development and deployment processes and enable the development of new applications at “Web speed.” Developers can create client applications using the Microsoft Visual Studio® Designer and host the applications on Web servers running Internet Information Services. Because developers are using ASP.NET controls on the servers, there is no need to burn code into the client. Developers can also use all of the ASP.NET configuration, performance, debugging, and deployment features.

The following diagram shows the discrete elements of a Microsoft Mediaroom PF application:

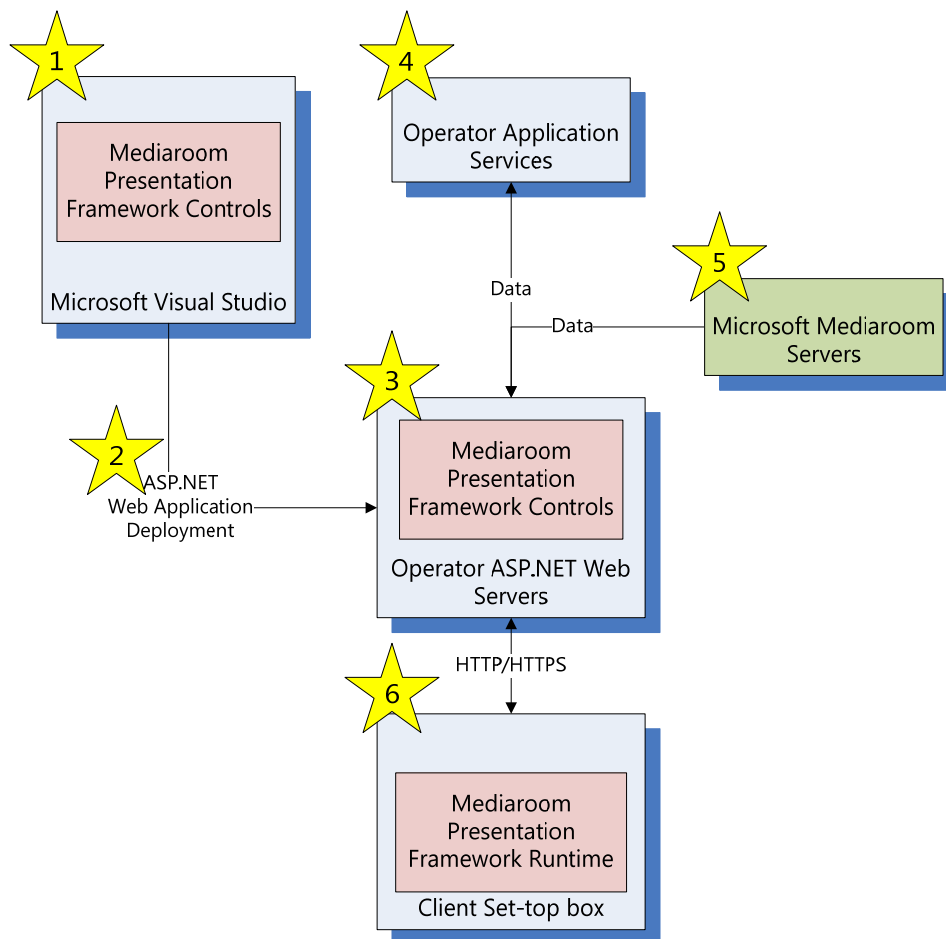


Figure 2 The building blocks of a Microsoft Mediaroom Presentation Framework Application

Developers can create Microsoft Mediaroom PF applications using Visual Studio, and they can do so in a manner that will be familiar to anyone who has developed standard ASP.NET applications. The Microsoft Mediaroom ADK installs the Microsoft Mediaroom PF controls inside the Visual Studio toolbox and a developer uses the controls to visually or programmatically build the application elements and connect to events, data sources, and

actions. Microsoft Mediaroom PF controls encapsulate code that enables them to bind to data sources, to serve as data sources themselves, to generate events, and to serve as the target of actions. For example, the data binding features enable a developer to create a Microsoft Mediaroom PF application that dynamically updates images from a source identified in an external Web site. The application could update selected text blocks at a specified frequency, and even update the information in a list based on the result of a Web service call. Developers can bind events to specific actions (clicked keys or pressed buttons), timers, and many other occurrences. Because the controls in the ADK rely on the same framework that renders the core Microsoft Mediaroom user interface, Microsoft Mediaroom PF applications can work easily with multiple video streams, transparent overlays, and animation, which gives viewers the same high levels of performance and responsiveness that they associate with the broader Microsoft Mediaroom experience.

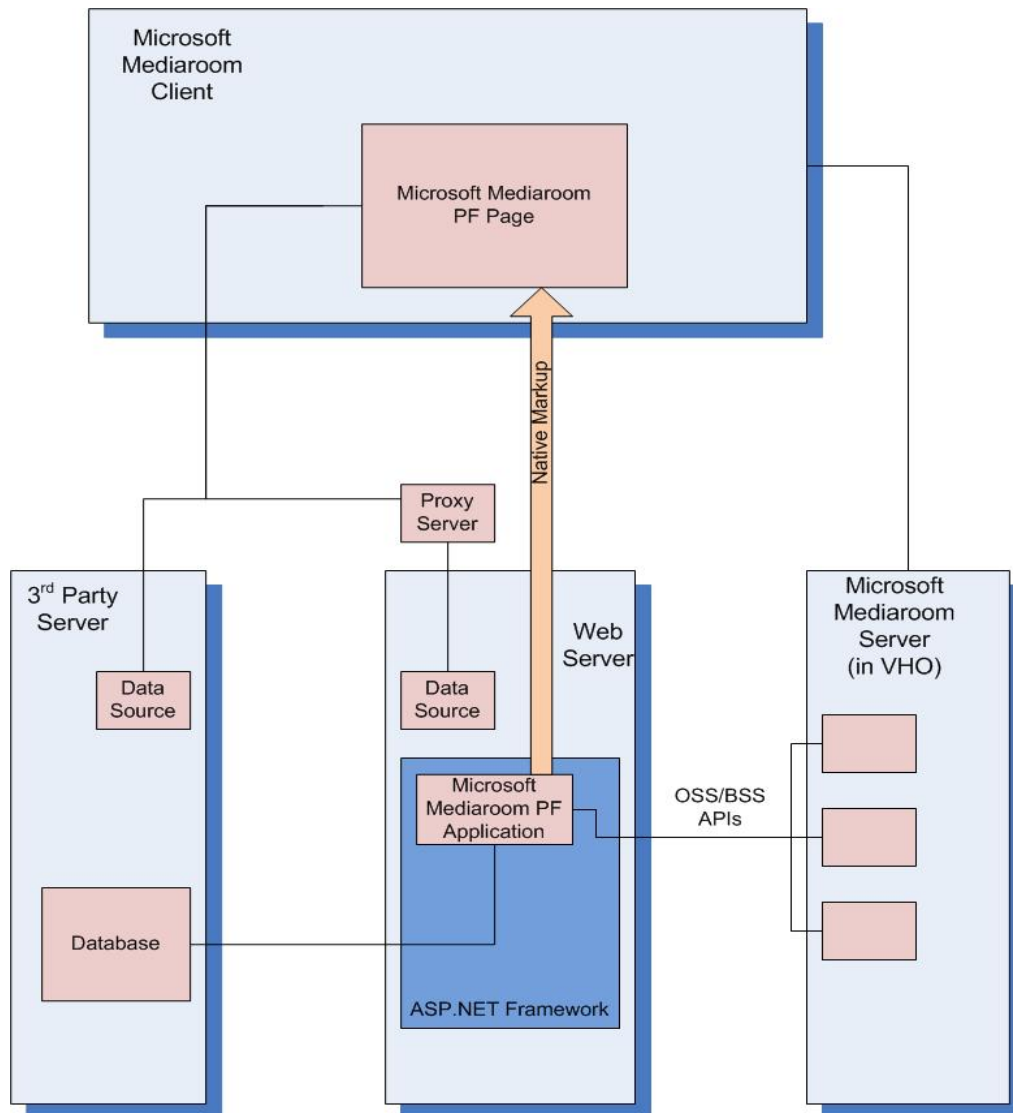


Figure 3 The Microsoft Mediaroom PF application architecture at a high level

ASP.NET helps developers place the business logic of a Microsoft Mediaroom PF application on the server. Due to the data binding and event-generation abilities of the ADK controls, the ASP.NET model also makes it possible to eliminate the need for custom code on the client and still produce applications with a native look and feel.

This code separation leads to simpler application development and a more stable client. The same business logic can also power other versions of the application interface—for example, on Web browsers or mobile phones.

Building with Web Services

The Microsoft Mediaroom PF also makes extensive use of Web services. This not only creates additional flexibility in application development but, because of the inherent reusability of Web services, can further lower the cost of developing applications. Using Web services you could, for example, create an application bound to a particular TV show that takes advantage of the show's fan Web site. As features on the Web site are developed or changed—such as exclusively available footage, cast interviews, polls, and contests—the features automatically become a part of the TV application.

The TV Dashboard from ES3, shown in Figure 4 below, shows another example of the use of Web Services in a Microsoft Mediaroom PF application. The TV Dashboard is a simple overlay application that can feature information and graphics pulled in real time from any Web-based news or information feed. As that content is updated at its source, it is updated in the application.



Figure 4 The TV Dashboard application designed by ES3

Applications and Layers

The layers feature of the Microsoft Mediaroom PF enables multiple applications and pages to run and appear on screen at the same time. Key events may propagate through layers. This enables, for example, an application in the focused layer to process key events while other applications continue to run in background layers. Layers also allow separation of disparate functions into multiple applications. For example, one application can enable a user to browse through video on demand (VOD) titles, and another application can appear without displacing the first to remind the user of a live event. When creating a launch point for an application (for example, a trigger or a navigate action) a developer can choose a layer in which to run the application. This facilitates dialog boxes opening in layers above the primary application.

Physics-based controls

To enable rich, advanced user experiences, the latest release of the Microsoft Mediaroom PF includes a set of smart, data-bound controls with physics-based animations. A physics-based grid and physics-based table are the primary controls used to organize various templates for use with data binding. These controls enable developers

to create one-dimensional lists, two-dimensional grids, and tables that group different lists of data. These controls use intelligent resource management so that a physics-based control uses fewer resources than a traditional control for the same task. The templates for the grid and table also allow finer control than traditional templates. Multiple types of data can use different templates, so lists displaying information about books or CDs, for example, can use different templates than lists displaying information about DVDs. Developers can also customize animations used to move between items and scroll through the grid or table.

Parental Controls

Microsoft Mediaroom PF applications integrate with the parental controls system. Before a user tunes in to a video or loads a page, an application can require that the user unlock the box using the appropriate PIN (Purchase, Adult, or Parent Control). An application can perform different actions depending on the success or failure of the PIN entry. For example, an application can load the intended content upon successful entry and load alternate content upon failure to enter the correct PIN. Microsoft Mediaroom PF applications use the same native PIN dialog as the rest of the client.

The Microsoft Mediaroom Simulator

The Microsoft Mediaroom simulator enables application developers to simulate a client environment in order to test and tune Microsoft Mediaroom PF applications. Using the simulator, which can run on the same PC that a developer uses to create a Microsoft Mediaroom PF application, a developer can quickly preview and test the behavior of Microsoft Mediaroom PF applications in a what-you-see-is-what-you-get (WYSIWYG) environment. The simulator also offers an online mode that enables it to act as a PC client to server-based deployment of Microsoft Mediaroom.

Developers who need a more full-featured performance testing environment for their Microsoft Mediaroom PF applications can use the Microsoft Mediaroom Personal Server. The Microsoft Mediaroom Personal Server consists of a single-seat deployment of Microsoft Mediaroom, all the key components of which are deployed on a single specially-configured PC. By connecting a TV to the supplied set-top box, a developer can test a Microsoft Mediaroom PF application in a true Microsoft Mediaroom environment. More information on the Microsoft Mediaroom Personal Server is available on the Mediaroom Backstage Web site.

Launching PF Applications

Service providers can configure Microsoft Mediaroom PF applications to launch from the Guide, service menus, handheld remotes, triggers tied to the video stream, or on-screen buttons. The application is identified through its URL, and when it is launched the Microsoft Mediaroom PF runtime uses either HTTP or HTTPS to connect to the server hosting the application. The Microsoft Mediaroom PF runtime retrieves the markup describing the application and the starts the execution process.

Applications can also launch in response to streamed content, timers, and other specified triggers. Unlike traditional enhanced television (ETV) systems that look for triggers embedded in the content stream itself, the Microsoft Mediaroom PF decouples the triggers from the video stream. The Microsoft Mediaroom PF relies on a client-resident trigger engine that monitors content and criteria in real time and looks for conditions that developers or service provider have defined as triggers (see Figure 5 below). Those conditions can include closed caption (CC) and timed text (TT) strings, stream time, wall-clock time, and program metadata attributes

(including program title and description). Using this more flexible approach to triggers, a service provider could, for example, launch a specific Microsoft Mediaroom PF application to run any time a subscriber tunes into a show on channel 35 that contains the word “lost” in its title. As soon as the trigger engine encounters a condition (or conditions) registered as a trigger, it fires the events defined by the developer.

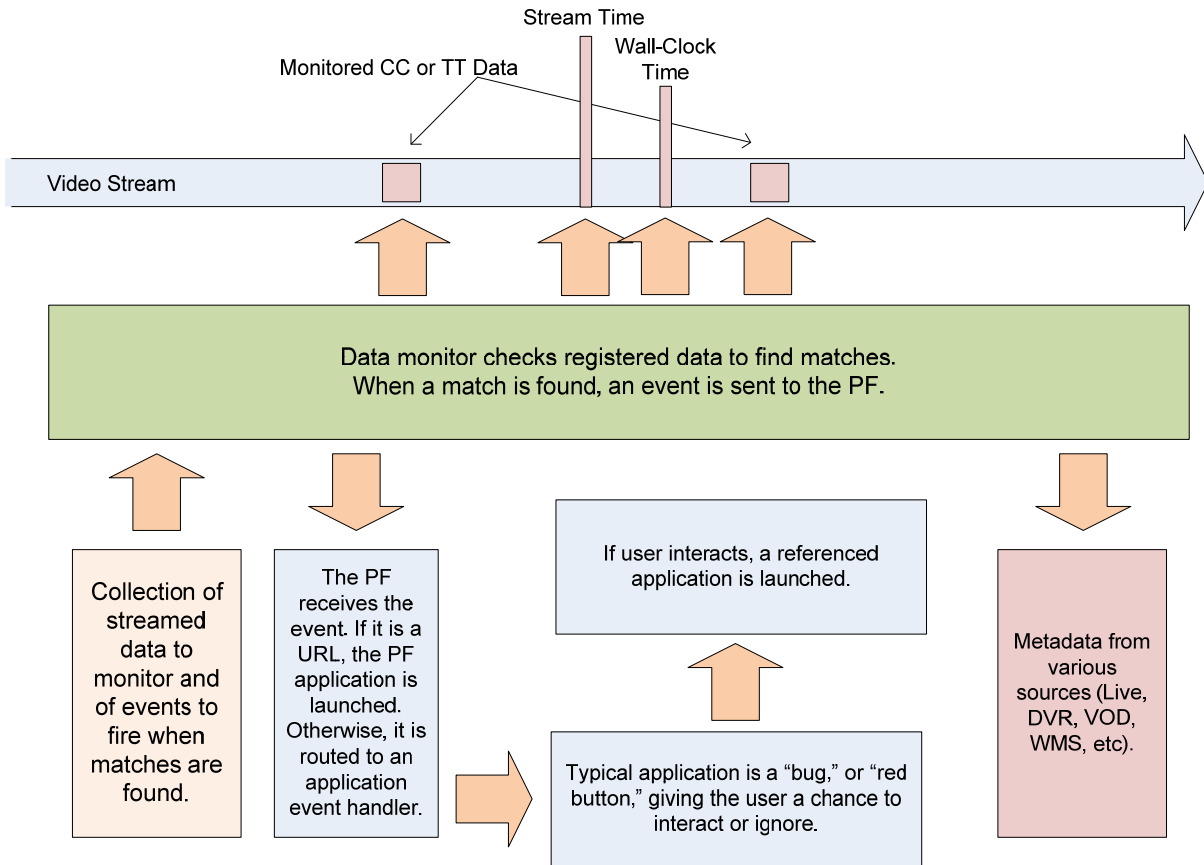


Figure 5 Extending the capabilities of triggers through the Microsoft Mediaroom PF

Protecting the Production Environments

The deployment paradigm for Microsoft Mediaroom PF applications assumes that new applications and features may be released very quickly. Because these applications will be running within a television systems environment, however, Microsoft has taken steps to ensure the reliability of operator-delivered video despite potential bugs in third party applications. To this end, the client has a “seatbelt” feature that limits the complexity of applications, thus avoiding runaway applications from consuming too many resources and degrading the stability or performance of the client.

The seatbelt mechanism assigns a weight to the various pieces of an application (controls, images, XML data, and so on) and prevents a page’s total weight from crossing a defined limit. The seatbelt mechanism also limits the total weight of all active Microsoft Mediaroom PF pages on the client. Operators set the limits based either on their own stability testing or on Microsoft-recommended values. The simulator has diagnostic features that allow developers to test and debug their applications with respect to the operator-defined seatbelt limits.

Creating Exciting New Subscriber Experiences

Ultimately, the purpose of the Microsoft Mediaroom PF is to enable the easy and rapid development of high performance applications that take advantage of the potential of a truly connected, personalized TV service. Service providers and third party developers can build whole new subscriber experiences with greater opportunities for personalization (for instance, building an application that presents VOD and DVR content according to an individual subscriber’s genre preferences) and for creating mash-ups using content from multiple new media sources. Consider the following examples of Microsoft Mediaroom PF applications deployed by AT&T and Deutsche Telekom:



Figure 6 AT&T U-Verse “News Multiview”



Figure 7 Deutsche Telekom “Liga total!”



Figure 9 AT&T U-Verse "Santa Tracker"

Microsoft Mediaroom PF applications can create exciting, connected experiences for subscribers, all delivered within a context circumscribed by parental controls and subscriber account parameters.

Summary

The Microsoft Mediaroom PF enables TV service providers and third party developers to create entirely new kinds of connected, interactive TV experiences. Developers with knowledge of ASP.NET and the Microsoft Visual Studio development tools will quickly become comfortable with developing Microsoft Mediaroom PF applications. The ADK provides prebuilt controls that developers can use to bring new applications to market quickly, and the Framework's reliance on Web services means that you can build upon your existing library of Web services, take advantage of Web services that other providers have built, and reduce the cost of bringing compelling new experiences to subscribers.

For more information about the Microsoft Mediaroom Presentation Framework and how Microsoft can help you deploy a successful consumer TV service, contact your Microsoft representative today.