



Viewpoint **Media Player** **Release Notes**

Version 3.0.12
November 2002

© 2003 Viewpoint Corporation. All Rights Reserved.

Viewpoint Media Player Release Notes

Viewpoint, the Viewpoint logo, Viewpoint Technology, Viewpoint Media Compressor, Viewpoint Media Publisher, Viewpoint FinalCheck, Viewpoint Scene Builder, and Viewpoint Media Player are registered trademarks or trademarks of Viewpoint Corporation in the United States and in other countries.

Companies, names, and data used in examples herein are fictitious unless otherwise noted. Information in this document is subject to change without notice.

Macromedia and Flash are trademarks or registered trademarks of Macromedia, Inc. in the United States and/or other countries. All other product and company names mentioned herein are the trademarks of their respective owners.

All other product and company names mentioned herein are the trademarks of their respective owners.

Disclaimer

Except as expressly provided otherwise in an agreement between you and Viewpoint, all information, software, and documentation is provided “as is,” without warranty of any kind. Viewpoint makes no warranties, express or implied, including without limitation the implied warranties of merchantability and fitness for a particular purpose regarding such information, software and documentation. Viewpoint does not warrant, guaranty, or make any representations regarding the use or the results of the software in terms of its correctness, accuracy, reliability, timeliness, suitability or otherwise. The entire risk as to the results of performance of the software is assumed by you.

In no event will Viewpoint be liable for any special, indirect, consequential, punitive, or exemplary damages or the loss of anticipated profits arising from the performance of the software or resulting from the loss of use, data or profits, whether in an action for breach of contract or warranty or tort (including negligence) arising out of or in connection with the information, technology, software and documentation.

The Web site and publications may contain technical inaccuracies or typographical errors. Viewpoint assumes no responsibility for and disclaims all liability for any such inaccuracy, error, or omission in the Web site and documentation and in any other referenced or linked documentation. Viewpoint may make changes to the information, software, Web site, documentation, prices, technical specifications, and product offerings in its sole discretion at any time and without notice.

Author: Caleb Hill

Contributors: Ari Berger, Barry Paul, Ben Guihaire, Carla Gray, Chris Earl, Derek Davies, Ed Peters, Nicolas Brun

Viewpoint Corporation

498 Seventh Avenue
Suite 1810
New York, NY 10018

Contents

Chapter 1: Introduction	4
New Features in this Release	4
Changed Behaviors in this Release.....	5
Windows Minimum System Requirements for this Release	8
Chapter 2: Introducing New Features, XML Tags, and Properties	9
New Animation Features	10
New HyperView Features.....	28
New Scene Features.....	29
New Scene Parameters Features.....	30
New Camera Features.....	31
New Interactors Features	32
New Hot Spot Features	34
New Instance Features	35
Chapter 3: General Feature Notes	37
AOL-Related Notes for Viewpoint Media Player 3.0.11 and 3.0.12	37
Setting the Viewpoint Media Player Cursor in Viewpoint Media Player 3.0.12 ..	37
Impact of Pop-up Stopper™ and AdAware™ on Viewpoint Media Player 3.0.12 .	
37	
Appendix A: Help, Resources, and Feedback	38
Viewpoint Developer Central: A Complete Resource.....	38
Download Viewpoint Applications, Guides, and Examples	38
Appendix B: Viewpoint Technology Content Creation Updates	42
Viewpoint Technology Content Checklist.....	42
Updating Existing Web Pages with MTS3Interface.....	42
Using One MTS3Interface.js File for a Web Site.....	43
Verifying the MTS3Interface Version in Existing Content.....	43

Chapter 1: Introduction

These release notes include the latest information on Viewpoint Media Player version 3.0.12 for the Windows operating system.

Note: The features introduced in Viewpoint Media Player 3.0.12 are the same features introduced in AOL Viewpoint Media Player release 3.0.11. ([Viewpoint Media Player release 3.0.10](#) is the latest release for the Mac operating system.)

This document has been organized into the following chapters:

- [Chapter 1: “Introduction”](#) — Overview of the new features in this Viewpoint Media Player release, changed behaviors and known issues in this release, and minimum system requirements.
- [Chapter 2: “Introducing New Features, XML Tags, and Properties”](#) — Learn about the new features and XML tags and properties that Viewpoint Media Player 3.0.12 supports.
- [Chapter 3: “General Feature Notes”](#) — Learn about workarounds and other tips for using Viewpoint Media Player 3.0.12.
- [Appendix A: “Help, Resources, and Feedback”](#) — Learn more about Viewpoint Technology, download free applications and documentation, and get technical support.
- [Appendix B: “Viewpoint Technology Content Creation Updates”](#) — Discover new content creation techniques and how to create content that runs optimally on the Microsoft Windows operating system and with Microsoft Internet Explorer or Netscape.

New Features in this Release

Viewpoint Media Player version 3.0.12 introduces support for the following new features, described in more detail in [Chapter 2: “Introducing New Features, XML Tags, and Properties”](#):

- [“Introducing SVG Basic Support”](#) — Vector graphic support directly through XML. In many cases, SVG can replace Flash in a Viewpoint Technology scene, saving valuable file size. Also, SVG XML can be generated on the fly, meaning that user interaction can generate the vector graphics automatically.
- [“New Features for Macromedia® Flash™ Animations”](#) — Viewpoint Media Player now supports Flash 4 scripting, which enables richer content creation. Also, the new Flash support enables you to easily construct user interfaces for Viewpoint Technology content.
- [“Introducing MTSCookie”](#) — The new Viewpoint cookie system allows better control over content, including a cross-domain cookie system that can limit the number of times content can play for an individual instance of Viewpoint Media Player and display alternate content.
- [“Introducing HyperView for AOL Rainman”](#) — Enables content beneath or covered by the Viewpoint Technology window to remain active, including ads and links.

- [“Example SVG Image in Viewpoint Technology Content”](#) — TalkNow is a new sound compression technology played directly through Viewpoint Media Player, ensuring that voice sound can be delivered efficiently with small files sizes while staying synched with the rest of the Viewpoint Technology content.
- [“Introducing Lens Flare Animations”](#) — Lens flares mimic the effect seen in photographs, video and film, or looking through a window, when light reflects or refracts off the lens or glass. Mimicking normal everyday visual experience increases the realism and cinematic effect of Viewpoint Technology content.

Changed Behaviors in this Release

This section contains a list of behaviors that have changed for this release of Viewpoint Media Player.

MTSScene Version Requirements

Each Viewpoint Technology scene must be assigned a Viewpoint Media Player version number via the <MTSScene Version> tag, which determines the version of the MTX/MTZ interpreter Viewpoint Media Player uses to process the scene. The value for version depends upon the version of Viewpoint Scene Builder in which the XML code is created.

Viewpoint Media Player release 3.0.11 and 3.0.12 share nearly all the same functionality ([“R3D Value Type”](#)). It is recommended that you set your minimum version requirement to Viewpoint Media Player 3.0.11 (“311”) to ensure users see new content created in 3.0.11 and 3.0.12:

```
<MTSScene Version="311" >
...
</MTSScene >
```

Changed Behaviors in Viewpoint Technology Content Via <MTSScene Version> Values

Some tags may alter their functionality and some tags may be ignored based on the version value assigned to this tag. The following table describes some of the changes between the previous public Viewpoint Media Player release for Windows, version 3.0.8, and the current build, version 3.0.12 (these same changes also apply to version 3.0.11 for AOL):

Content Behavior in 3.0.8	Content Behavior Changes in 3.0.12
Animations playing in reverse returned to the beginning of the animation when rewind, making it impossible for the animation to begin playing in reverse again.	Animations playing in reverse can be rewind to the end of the animation, allowing them to play from the end to the beginning, as specified.
Setting rewind, reset, or trigger to a parent animation also rewind, reset, or triggered the parent's children animations.	Setting rewind, reset, or trigger to a parent animation affects only the parent animation.

Content Behavior in 3.0.8

Interactors were inserted from bottom to top, meaning, they were executed from the last one defined to the first one.

For example, in Viewpoint Media Player 3.0.8, if two interactors were waiting for the same event, the last one would be executed first, then the first one (when `IgnoreHandled="1"` was specified).

When specifying two animations with the same name in a Viewpoint Technology scene, the animations were considered like one unique animation (this is still the case). However, in 3.0.8, this created inefficiencies in content due to the unnecessary computations this format required since the animation was added twice to the hierarchy (causing it to be executed twice).

Content Behavior Changes in 3.0.12

Interactors are parsed in the order they appear in the MTX code.

Two like-named animations are merged into the .mtx file, but the animation is inserted only once into the hierarchy (at the first position), allowing predeclaration.

Important: Regardless of the version value your scene is assigned, it is highly recommended that you use only unique names for each animation and interactor.

Viewpoint Media Player Version Reference Information

For more information on Viewpoint Media Player version numbers, auto-updating, and required settings for your web page's Viewpoint Technology content, see [Testing and Deploying Viewpoint Technology Content](#).

Pause Functionality

In Viewpoint Media Player 3.0.8, an animation set to pause could be triggered and remain paused. In Viewpoint Media Player 3.0.12, an animation set to pause cannot be triggered and remain paused; when the animation is triggered, the pause switches off (that is, the animation plays). To work around this functionality change, do not use `<MTSAction> <Trigger>` to start a paused animation; use `<MTSAction> <Start>`.

Note: For more information about how to use `<MTSAction> <Start>`, see the [Viewpoint XML Reference Guide](#).

XML Naming Conventions for Viewpoint Technology Properties

To make working with Viewpoint Technology XML properties easier, Viewpoint announces that as of Viewpoint Media Player release 3.0.12, Viewpoint Technology properties can now exceed their traditional 4-character length. As a result of this change in Viewpoint Technology property-naming conventions, an XML tag name and, if applicable, its corresponding property name can match. While not all related tags and properties introduced in Viewpoint Media Player version 3.0.12 match in spelling, many do.

Note: Four-letter properties remain valid to ensure backward compatibility with existing Viewpoint Technology content. For content to work properly, ensure that you employ the correct property spelling in the MTX code.

Example for New Viewpoint Technology Property Names

The following example illustrates the difference between the property-naming convention of Viewpoint Media Player 3.0.12 and that of prior Viewpoint Media Player versions.

Introduced Prior to Viewpoint Media Player 3.0.12	Introduced in Viewpoint Media Player 3.0.12
---	---

Tag: <Rotation>
Property: rot_

Tag: <VETCamera Rotation>
Property: Rotation

Important: Use Viewpoint tags and properties exactly as they are spelled to ensure that your content works properly. For example, if you are using the rmod property and spell it rmode, Viewpoint Media Player does not recognize the property nor does it process its functionality.

Error Messages

Parse error messages no longer display when XML errors, such as missing closing tags, misspelled properties, etc., are present in .mtx files. There is no replacement error notification. To work around this change, use debugging tools, such as Viewpoint Scene Builder, XML Schema, etc. to detect and correct content mistakes.

Known Issues in this Release

This section contains a list of known issues for this release of Viewpoint Media Player.

R3D Value Type

When you animate a rotation of MTSCamera (using the rot_, romi, and roma properties) or MTSInstance (using the rot_ property) and you do not want keyframe interpolation (that is, tweening is off [Tween="0"]), then use the 3D value type instead of the R3D value type. Considering that the only difference between 3D and R3D is interpolation, using the 3D type instead of the R3D type has the same effect. However, if you use R3D in this instance, the results will be mathematically incorrect.

Note: This issue applies only to Viewpoint Media Player version 3.0.11.

Flash Textures

Content that uses an animated Macromedia® Flash™ movie as an invisible texture on the screen may use significant system resources if the SWFView animator is not turned off while the texture is invisible (that is, the object to which the texture is applied collapses or is set to invisible). To turn the animator off, simply set the runs property to '0'.

Keyframe Animations

Animations that contain only one (1) keyframe and are set to <Time> 0 </Time> do not work. To fix these animations, either add another keyframe or change the time to something other than "0", as illustrated in bold below:

```
<MTSTimeElem Type="Keyframe" Name="animation_reset" On="0"
AutoStop="1">
  <Time> 0.001 </Time>
  <MTSTimeElem Type="ActionAnimator" >
    <animation_reset_action1/>
```

```
</MTSTimeElem>  
</MTSTimeElem>
```

Netscape 7.0 Issues

Server Generated Content (JSP, ASP, and CGI)

In instances when a Viewpoint XML file (MTX/MTZ) does not have an .mtx, .mtz or .xml file extension and is smaller than 256 bytes, Viewpoint Media Player cannot load the file. Files served by AOL Rainman (with AOL SAP URLs) and dynamically generated content (CGI, JSP, and ASP) may be missing the extensions. To resolve this issue, ensure that all files contain an .mtx, .mtz, or .xml extension or are larger than 256 bytes.

File Paths

If you are viewing Viewpoint Technology content from your local hard drive using Netscape 7.0, the file will not display correctly if the file path has spaces. Remove spaces from the file path to view content correctly in Netscape 7.0.

Netscape 7.0 Installation

In some examples of Netscape installation, we have found that the Viewpoint Media Player version developed for the Netscape 4.x browsers incorrectly appears in the Plugins folder of the Netscape 7.0 browser. This mismatch can cause erratic Viewpoint Technology content behavior. To correct this issue, follow these steps:

- 1 Browse to the folder where you installed Netscape.
- 2 Select the Plugins folder.
- 3 If the folder contains the NPMetastream.dll file, delete it.

Windows Minimum System Requirements for this Release

- Microsoft® Windows® 95, Windows 98, Windows 2000, Windows Millennium Edition, Windows NT® 4.x, or Windows XP
- Netscape Navigator® 4.07 or later (not 6.0)
- Microsoft® Internet Explorer 4.x or later
- AOL® version 4, 5, 6, 7, or 8
- CompuServe 7
- 256 color display (24-bit recommended)
- 28.8 Kbps modem
- 5 MB available hard-disk space
- 32 MB RAM (64 MB recommended)
- Pentium® 166 MHz processor or faster (Pentium II recommended)

Chapter 2: Introducing New Features, XML Tags, and Properties

This chapter summarizes the new and enhanced features introduced in Viewpoint Media Player release 3.0.12, including all new tags and properties. Specifically, Viewpoint announces new or enhanced Viewpoint Media Player support of the following Viewpoint Technology XML categories:

- [“New Animation Features”](#) (<MTSTimeElem>) — Viewpoint Media Player now supports the following new animation features:
 - Scalable Vector Graphics (SVG) Basic (vector graphic format) — Initial support of a subset of SVG tags.
 - TalkNow — Enables sound streaming through Viewpoint Media Player.
 - Lens flares — Enables lighting effects for a scene.

Viewpoint Media Player includes enhanced support for the following existing animation features:

- QuickTime Pro object movies — Allows more efficient naming conventions for MTSObjectMovie images.
- Flash 4 (vector graphic authoring tool) — Supports more rendering and ActionScript functions and allows more effective scripted Flash-to-Viewpoint Media Player communication.
- Emboss — Includes new color and fill text tags and properties.
- Keyframe animations — Includes new trigger feature.
- [“New HyperView Features”](#) (<MTSBaseComponent>) — Viewpoint Media Player now supports HyperView for AOL Rainman content and includes enhanced support for non-AOL content.
- [“New Scene Features”](#) (<MTSScene>) — Viewpoint Media Player now supports new cookie sharing and camera switching features, and also supports enhanced set property functionality.
- [“New Scene Parameters Features”](#) (<MTSSceneParms>) — Viewpoint Media Player now supports "silent" content uploading to the Viewpoint Media Player cache.
- [“New Camera Features”](#) (introducing <VETCamera>) — Viewpoint Media Player now supports VETCamera, a new animatable camera model that can operate in conjunction with the existing MTSCamera model. VETCamera includes new matrix-copying properties.
- [“New Interactors Features”](#) (<MTSAction> and <MTSInteractor>) — Viewpoint Media Player now supports the special Viewpoint Technology function, ClearScene, which is called via the <MTSAction> tag. Viewpoint Media Player also includes enhanced support for the following existing interactor features:
 - String value concatenation and set property functionality for <MTSAction>
 - Animation event for <MTSInteractor>
 - Prioritization for <MTSPreload>

- [“New Hot Spot Features”](#) (<LayerData>) — Viewpoint Media Player now supports the transformation of hot spot text (in terms of size, orientation, skew, and position).
- [“New Instance Features”](#) (<MTSInstance>) — Viewpoint Media Player now supports an alternate method for transforming instances and using matrix coordinates.

New Animation Features

Animations modify scene elements over time and, in an .mtx file, are contained between the MTSTimeElem start and end tags. Specifically, animations load scene files and resources, manipulate cameras and objects (including textures), and enhance a scene's interactivity.

Viewpoint Media Player 3.0.12 introduces an exciting array of new features, such as SVG, lens flares, and new Flash commands, described below.

Note: For more information on these features, see the [Viewpoint XML Reference Guide](#).

Introducing SVG Basic Support

The SVG format is emerging through the cooperative efforts of the World Wide Web Consortium (W3C) and its members. Viewpoint has worked to ensure that Viewpoint Media Player 3.0.12 is SVG compatible. SVG is an industry-accepted vector-graphics XML format. Functionality-wise, it is somewhat similar to Flash, but it is a text format that you need only Windows Notepad to write.

SVG in Viewpoint Technology Content

SVG is a language for describing two-dimensional graphics in XML. SVG allows graphic objects in Viewpoint Technology content, such as primitive shapes and text. The SVG feature set for Viewpoint Technology content includes nested transformations, alpha masks, filter effects, template objects, and extensibility.

Advantages of Using SVG in Viewpoint Technology Scenes

One of the main advantages of using SVG in a Viewpoint Technology scene includes SVG's inherent compatibility with XML. For example, to modify an annotation, an arrow, and a tooltip for content that is already published with Flash requires a huge effort. With SVG, it is a matter of writing a couple of lines into the XML file.

Note: For more information about SVG, go to <http://www.w3.org/Graphics/SVG/Overview.htm>.

SVG Tags Supported by Viewpoint Media Player

Viewpoint Media Player 3.0.12 supports a subset of the entire suite of SVG features, including, but not limited to, those described in the following tables. In addition, the entire Viewpoint Technology XML <MTSTimeElem Type="SWFView"> tag suite now supports SVG.

The SVG Document

Description	An SVG document fragment consists of any number of SVG elements contained within an <svg> element. An SVG document fragment can stand by itself as a self-contained file or resource, in which case the SVG document fragment is an SVG document, or it can be embedded inline as a fragment within a parent XML document.
Tag	<svg>
Attributes	<p>width: For outermost <svg> elements, the intrinsic width of the SVG document fragment. For embedded <svg> elements, the width of the rectangular region into which the <svg> element is placed.</p> <p>height: For outermost <svg> elements, the intrinsic height of the SVG document fragment. For embedded <svg> elements, the height of the rectangular region into which the <svg> element is placed.</p>
Default Value	100% — If the attribute is not specified.
Other Value(s)	<p>(-) — A negative value results in an error</p> <p>0 — Disables rendering of the element.</p>
Usage Dependencies	Must be introduced in an .mtx file with <MTSTimeElem Type="SWFView">.
Remarks	An SVG document fragment can range from an empty fragment (that is, no content inside of the <svg> element) to a very simple SVG document fragment containing a single SVG shape/graphic element, such as <rect>, to a complex, deeply nested collection of subelements, such as <transform>.
Sample Code	<pre><MTSTimeElem Type="SWFView" On="1" > <svg width="200" height="200" > <line x1="10" y1="70" x2="100" y2="170" style=" stroke: green; stroke-width: 2;" /> </svg> </MTSTimeElem></pre>

Grouping

Description	The <g> element is a container element for grouping together related graphics elements.
Tag	<g>
Attributes	transform: The value of the transform attribute is a <transform-list>, defined as a list of transform definitions applied in the order specified. The individual transform types, including matrix, translate, scale, and rotate are separated by whitespace and/or a comma.
Default Transform Type Values	0 — Defined as a real number for the x, y, and z axes.
Usage Dependencies	Must be introduced as a subelement of <svg>.
Remarks	The transform attribute is applied to an element before processing any other coordinate or length values supplied for that element.
Sample Code	<pre> <MTSTimeElem Type="SWFView" On="1" > <svg width="512" height="512"> <g transform="translate(0,100)"> <text y="50" fill="rgb(75,195,24)" font-size="48" font- family="times"> Viewpoint </text> <animateTransform begin="0" dur="3" repeatCount="2" values="0,0; 20,0; 330,0; 300,0; 0,0" keyTimes="0; 0.1; 0.5; 0.6; 1"/> </g> </svg> </MTSTimeElem> </pre>

Shapes

Description	Shapes is a graphics classification, not a tag itself, that serves as an umbrella term to describe the features listed below as subelements. Mathematically, these shape elements are equivalent to a <path> element that would construct the same shape.
Tag	None.
Elements	<rect>, <circle>, <ellipse>, <line>, <path>

Attributes	<p>fill: The operation of coloring the interior of a shape or the interior of the character graphics in a text string.</p> <p>fill-opacity: Specifies the opacity of the painting operation used to paint the interior of the current object.</p> <p>stroke: Colors the outline of the specified graphical element.</p> <p>stroke-width: The width in pixels of the stroke on the current object.</p> <p>stroke-opacity: Specifies the opacity of the painting operation used to stroke the current object.</p> <p>transform: See the definition in the </g> tag table.</p>
Usage Dependencies	All shape tags must be introduced as a subelement of <svg>.
Remarks	The basic shapes may be stroked, filled, and used as clip paths. All the properties available for <path> elements also apply to the basic shapes.

Rectangle

Description	Defines a rectangle. Rounded rectangles can be achieved by setting appropriate values for the attributes, rx and ry.
Tag	<rect>
Attributes	<p>x: The x-axis coordinate of the side of the rectangle.</p> <p>y: The y-axis coordinate of the side of the rectangle.</p> <p>width: The width in pixels of the rectangle.</p> <p>height: The height in pixels of the rectangle.</p> <p>rx: For rounded rectangles, the x-axis radius of the ellipse used to round off the corners of the rectangle.</p> <p>ry: For rounded rectangles, the y-axis radius of the ellipse used to round off the corners of the rectangle.</p>
Default Attribute Value	0 — If the attributes x, y, width, and height are not specified.
Other Value(s)	The rx and ry values are automatically set to match each other when either of the two is not specified.
Usage Dependencies	Must be introduced as a subelement of <svg>.
Sample Code	<pre><MTSTimeElem Type="SWFView" Name="barry" On="1" > <svg width="512" height="512" > <rect x="50" y="70" width="335" height="320" style="fill: blue; fill- opacity: 0.5; stroke: green; stroke- width: 2" /> </svg> </MTSTimeElem></pre>

Circle

Description	Defines a circle based on a center point and a radius.
Tag	<circle>
Attributes	<p>cx: The x-axis coordinate of the center of the circle.</p> <p>cy: The y-axis coordinate of the center of the circle.</p> <p>r: The radius of the circle.</p>
Default Value	0 — If not specified.
Usage Dependencies	Must be introduced as a subelement of <svg>.
Sample Code	<pre><MTSTimeElem Type="SWFView" Name="barry" On="1" > <svg width="512" height="512" > <circle cx="200" cy="100" r="50" stroke="blue" stroke-width="2" /> </svg> </MTSTimeElem></pre>

Ellipse

Description	Defines an ellipse.
Tag	<ellipse/>
Attributes	<p>cx: The x-axis coordinate of the center of the ellipse.</p> <p>cy: The y-axis coordinate of the center of the ellipse.</p> <p>rx: The x-axis radius of the ellipse.</p> <p>ry: The y-axis radius of the ellipse.</p>
Default Value	0 — If not specified.
Other Value(s)	(-) — Negative values result in an error.
Usage Dependencies	Must be introduced as a subelement of <svg>.
Sample Code	<pre><MTSTimeElem Type="SWFView" Name="barry" On="1" > <svg width="512" height="512" > <ellipse cx="0" cy="0" rx="150" ry="250" fill="red" /> </svg> </MTSTimeElem></pre>

Line

Description	Defines a line segment that starts at one point and ends at another.
Tag	<line/>
Attributes	x1 : The x-axis coordinate of the start of the line. y1 : The y-axis coordinate of the start of the line. x2 : The x-axis coordinate of the end of the line. y2 : The y-axis coordinate of the end of the line.
Default Value	0 — If not specified.
Usage Dependencies	Must be introduced as a subelement of <svg>.
Remarks	Because the <line> element is a single line, and thus, is geometrically one-dimensional, it has no interior and cannot be filled.
Sample Code	<pre><MTSTimeElem Type="SWFView" On="1" > <svg width="200" height="200" > <line x1="10" y1="70" x2="100" y2="170" style=" stroke: green; stroke-width: 2;" /> </svg> </MTSTimeElem></pre>

Path

Description	Represents the outline of a shape, which can be filled, stroked, used as a clipping path, or any combination of the three.
Tag	<path/>
Attributes	d : Specifies the outline definition of a shape.
Usage Dependencies	Must be introduced as a subelement of <svg> and used as one expression: <path d=""/>.
Remarks	Path data animation is only possible when each path data specification within an animation specification has exactly the same list of path data commands as the d attribute.
Sample Code	<pre><MTSTimeElem Type="SWFView" On="1" > <svg width="200" height="200" > <path style=" stroke: green; stroke-width: 2;" d="M99.333,51.666c0,26.326- 21.341,47.667- 47.666,47.667s4,77.992,4,51.666C4,25.341,25.3 41,4,51.667,4s47.666,21.341,47.666,47.666z"/> </svg> </MTSTimeElem></pre>

Text

Description	Defines a graphics element consisting of text.
Tag	<text/>
Attributes	<p>font-family: Describes the name of the font family of a font and takes the same values as the font-family property.</p> <p>font-size: Same syntax and semantics as the descriptor within a font-face rule.</p> <p>font-style: The style of a font.</p> <p>font-weight: The weight of a face relative to others in the same font family.</p> <p>transform: See the definition of this tag in the <g/> table.</p>
Usage Dependencies	Must be introduced as a subelement of <svg>.
Sample Code	<pre><MTSTimeElem Type="SWFView" On="1" > <svg width="512" height="512"> <text y="50" fill="rgb(75,195,24)" font-size="48" font-family="times" font-style="italic"> Viewpoint </text> </svg> </MTSTimeElem></pre>

Animation

Description	Animates a single attribute or property over time.
Tag	<animate/>
Attributes	<p>begin: Defines when the element should begin (that is, become active).</p> <p>dur: Specifies the simple duration.</p> <p>repeatCount: Specifies how many times the animation repeats.</p> <p>values: A list of one or more values separated by a semicolon.</p> <p>keyTimes: A list of time values used to control the pacing of the animation separated by a semicolon..</p> <p>attributeName: Specifies the name of the target attribute in the parent, and includes the values: "fill-color" and "stroke-color".</p>
Default Attribute Value	For more information on the values of these attributes, see http://www.w3.org/TR/SVG/animate.html - AnimationElements

Usage Dependencies Must be introduced as a subelement of <svg>.

Sample Code

```
<MTSTimeElem Type="SWFView" On="1" >
  <svg width="512" height="512">
    <g transform="translate(0,100)">
      <text y="50"
fill="rgb(75,195,24)" font-size="48" font-
family="times" font-style="italic">
Viewpoint </text>
      <animateTransform begin="0"
dur="3" repeatCount="2"
values="0,0; 20,0; 330,0;
300,0; 0,0"
keyTimes="0; 0.1; 0.5; 0.6;
1"/>
    </g>
  </svg>
</MTSTimeElem>
```

Transform Animation

Description Modifies one of SVG's transformation attributes over time, such as the <transform> attribute.

Tag <animateTransform/>

Attributes

begin: Defines when the element should begin (that is, become active).

dur: Specifies the simple duration.

repeatCount: Specifies how many times the animation repeats.

values: A list of one or more values separated by a semicolon.

keyTimes: A list of time values used to control the pacing of the animation separated by a semicolon.

transform: See the definition of this tag in the <g> table.

type: Indicates the type of transformation whose values change over time, including the values: "translate", "scale", and "rotate".

additive: Controls whether or not the animation is additive ("sum") or is replaced ("replace").

attributeName: Must always be expressed equal to transform.

Default Value For more information on the values of these attributes, see <http://www.w3.org/TR/SVG/animate.html> - AnimationElements

Usage Dependencies Must be introduced as a subelement of <svg>.

Sample Code

```
<MTSTimeElem Type="SWFView" On="1" >
  <svg width="512" height="512">
    <g transform="translate(0,100)">
      <text y="50"
fill="rgb(75,195,24)" font-size="48" font-
family="times" font-style="italic">
Viewpoint </text>
      <animateTransform additive="sum"
begin="0s" dur="60s"
attributeName="transform" fill="freeze"
type="translate" values="2325,3500;
2050,3100;" />
    </g>
  </svg>
</MTSTimeElem>
```

Color Animation

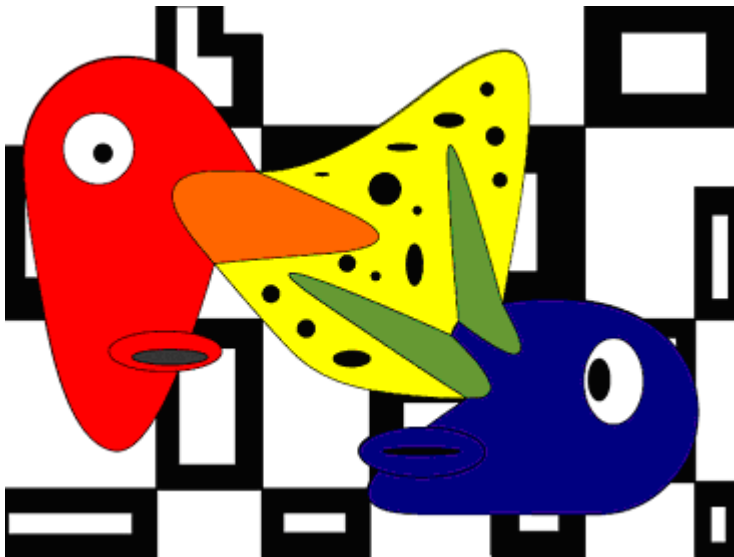
Description	Specifies a color transformation over time.
Tag	<animateColor/>
Attributes	<p>begin: Defines when the element should begin (that is, become active).</p> <p>dur: Specifies the simple duration.</p> <p>repeatCount: Specifies how many times the animation repeats.</p> <p>values: A list of one or more values separated by a semicolon.</p> <p>keyTimes: A semicolon-separated list of time values used to control the pacing of the animation.</p> <p>fill: The operation of coloring the interior of a shape or the interior of the character graphics in a text string.</p> <p>stroke: Colors the outline of the specified graphical element.</p> <p>attributeType: Does not affect a Viewpoint Media Player scene, but is required for color animation content to work.</p>
Default Value	For more information on the values of these attributes, see http://www.w3.org/TR/SVG/animate.html - AnimationElements
Usage Dependencies	Must be introduced as a subelement of <svg>.

Sample Code

```
<MTSTimeElem Type="SWFView" On="1" >
  <svg width="512" height="512">
    <text y="50" fill="rgb(75,195,24)"
font-size="48" font-family="times" font-
style="italic">Viewpoint
    <animateColor begin="0" dur="3"
repeatCount="10" attributeName="fill"
attributeType="CSS" fill="freeze"
values="#FF0000; #FFFFFF; #FF0000; #FFFFFF;
#FF0000" />
    </text>
  </svg>
</MTSTimeElem>
```

Example SVG Image in Viewpoint Technology Content

The following graphic illustrates an example of SVG using vector objects that display more crisply than a raster image.

**Introducing Viewpoint TalkNow (Viewpoint Media Player Sound Streaming)**

Viewpoint introduces TalkNow, which enables sound to be streamed from Viewpoint Media Player. With TalkNow, Viewpoint Technology scenes can be enhanced with spoken content, thereby bringing content and sounds together for a compelling presentation. Create TalkNow files by pairing an object with speech to deliver a completely synchronized speaking animation. Once created, broadcast this content with Viewpoint Media Player.

At 6 kilobits per second, TalkNow delivers streamed spoken content in the lowest bandwidth available on the market, making it six to ten times smaller than MP3. With the streaming, static bit-rate, your speech file plays progressively over a low bandwidth, reducing the chances of a skip.

Creating TalkNow Files

Creating TalkNow files is easy! To create TalkNow files and add them to a Viewpoint Technology scene, follow the instructions in Viewpoint's Enhance Your Scenes with TalkNow quick-reference guide.

Tag and Property Descriptions

Viewpoint Media Player 3.0.12 introduces <MTSTimeElem Type="VMPSpeech" Path>, which sets the location of the Viewpoint TalkNow sound file (always converted to an .mts file extension). <VMPSpeech Path> cannot be animated via a keyframe animation, but can be changed in the MTX code.

Note: In Viewpoint Technology scenes, TalkNow is activated via <VMPSpeech> tags, described in the [Viewpoint XML Reference Guide](#).

Introducing Lens Flare Animations

A lens flare is a lighting effect that occurs when internal refractions and refractions of incident light within a photographic camera create bright patterns within the image. Lens flares are considered both undesirable and desirable by photographers, depending on the image being shot.

In case of computer graphic reproduction of lens flares, adding this lighting effect to computer generated images or animations can help overcome the lifelessness of computer-rendered scenes. The lens flare effect can bring a more "natural" feel to a Viewpoint Technology scene.

Tag and Property Descriptions

The following table lists the MTSLensFlares attributes and describes how to use them in MTX code.

MTSTimeElem - Type="LensFlare"

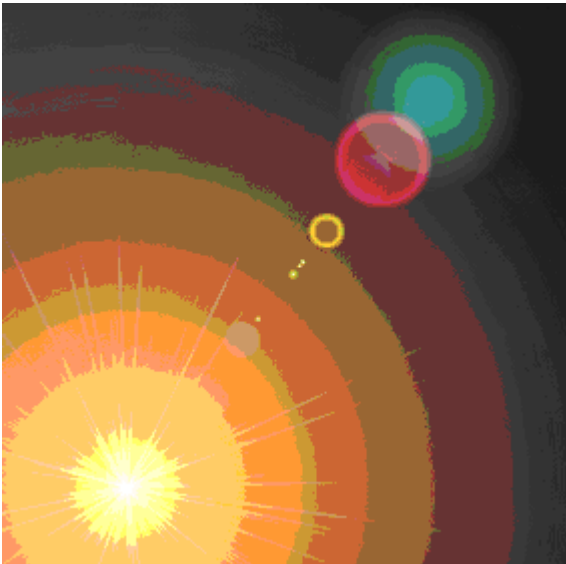
Description	Creates the effect of a lens flare in a scene using either 2D or 3D coordinates. A lens flare recreates the effect of an artifact of a bright light shining directly into the lens assembly of a camera.
Tag	<MTSTimeElem Type="MTSLensFlares">
Attributes	Name, loc2, loc3, vno3, vio2, lmap, zmin, zmax, halo, refl, stco, haco, lico, soco, inco, ouco, rtco, Brit, liin, lisc, hain, hasc, rein, resc, stin, stsc, stro, dlco, dsgl, digl, dogl, drtt, postAnimator
Usage Dependencies	None.
Remarks	MTSLensFlares wait to display in a scene until after the object to which they are attached downloads. Avoid high CPU use by reducing MTSLensFlares scales, such as the lisc tag. You can compensate for lost color intensity via reduced scales by increasing the values of MTSLensFlares intensity tags.

Sample Code
(includes all
MTSLensFlares tags)

```
<!--All MTSLensFlares attributes used as tags.  
-->  
<MTSTimeElem Type="MTSLensFlares"  
Name="myLensFlare1" loc2="0.000000 0.000000"  
loc3="0.000000 0.000000 0.000000"  
vno3="0.000000 0.000000 0.000000"  
vio2="0.000000 1.000000" lmap="0" zmin="-  
1.000000 -1.000000" zmax="-1.000000 -  
1.000000" halo="1" strk="1" refl="4"  
stco="0.999954 0.999954 0.999954"  
haco="0.531243 0.019760 0.019760"  
lico="0.869398 0.881544 0.223621"  
soco="0.855238 0.843748 0.249989"  
inco="0.815747 0.593744 0.052674"  
ouco="0.703120 0.332021 0.085435"  
rtco="0.000000 0.000000 0.000000" Brit="1.37"  
liin="1" lisc="1.49" hain="0.05"  
hasc="1.49" rein="0.12" resc="1.11"  
stin="0.68" stsc="0.92" stro="0" dlco="1"  
dsgl="1" digl="1" dogl="1" drtt="1"  
PostAnimator="1" />
```

Example MTSLensFlares Image

The following graphic illustrates an example of MTSLensFlares, which adds a more realistic effect to Viewpoint Technology content.



New Features for Macromedia® Flash™ Animations

Flash technology is commonly accepted for displaying interactive graphics on the Internet. It is the technology of choice for displaying animated advertisement banners, product presentations, online computer games, etc. Flash is vector-based, like SVG, and provides a high-level of file compression and high, resolution-independent image quality.

Viewpoint Media Player 3.0.12 introduces support for many new Flash features, described below, and also introduces a new property, `tmap`, which multiplies the speed of a time-based or a `SWFView` animation by an `mts_real` value.

Flash Rendering in Viewpoint Media Player

Viewpoint Media Player now supports the rendering capabilities of Flash 5, including these features:

- Vector-based, bitmap-based and text drawing
- Dynamic affine transforms: scaling, rotation, skewing of both vector and non-vector based shapes
- Shape morphing (tweening) of vector shapes (control point animation)
- Support for embedded fonts
- Transparency and translucency over the browser, with full 8-bit alpha channel
- Keyframe animations
- Sprite animations
- Morphing shapes (tweening)
- Hue and tint
- JPEG images
- Layer masks

ActionScript Functionality in Viewpoint Media Player

Also, ActionScript functionality is enabled to support the following:

- Play a submovie (movie clip).
- Jump to a frame in the main movie or submovie.
- Invoke a JavaScript function.
- Communicate commands from the Flash file to the Viewpoint Media Player window using the fscommand.

In particular, the following ActionScript commands are now supported:

- ActionNone — Functions as a Noop.
- ActionNextFrame — Goes to the next frame.
- ActionPrevFrame — Goes to the previous frame.
- ActionPlay — Starts playing at the current frame (ActionScript: Play).
- ActionStop — Stops playing the movie at the current frame (ActionScript: Stop).
- ActionWaitForFrame — Waits for the specified frame. Otherwise, skips the specified number of actions (ActionScript: ifFrameLoaded).
- ActionSetTarget — Sets context of action (Actionscript: Telltarget, GotoAndPlay, and others).
- ActionGotoLabel — Goes to frame associated with the specified label (ActionScript: GotoAndPlay).
- ActionGotoFrame — Instructs Viewpoint Media Player to go to the specified frame in the current movie.
- ActionBranchIfTrue[ActionIf] — Expresses conditional test and branch.

- ActionBranchAlways[ActionJump] — Expresses unconditional branch.
- ActionCall[Frame] — Call a subroutine.

Specific Flash Commands in Viewpoint Media Player

Also, the following Flash commands, described in detail below, are now enabled in Viewpoint Media Player 3.0.12:

- fscommand()
- ifFrameLoaded()
- tellTarget()
- gotoAndPlay()
- gotoAndStop()
- play()
- stop()
- nextFrame()
- prevFrame()

Tag Descriptions

The following tables describe how to script Flash-to-Viewpoint Media Player communication and how scripting behavior is changed for Flash movies embedded in a Viewpoint Technology scene.

fscommand()

Description	<p>Inside a Viewpoint Technology scene, allows the Flash movie to communicate with Viewpoint Media Player, but only when the Flash feature is in the foreground of the scene. All communication from Flash to Viewpoint Media Player is affected through fscommand(). To date, communication from Viewpoint Media Player to Flash is not possible.</p> <p>Important: The fscommand() function adapted to a Viewpoint Technology scene DOES NOT function the same as the Flash fscommand() that communicates between the Macromedia Flash Player and a Web browser.</p>
Syntax	fscommand(command, arguments)
Usage Dependencies	Must be introduced in an .mtx file with <MTSTimeElem Type="SWFView">.
Remarks	Works with pre- and post-animators, not as a texture.
Reference	<p>Read more about fscommand() at:</p> <p>http://www.macromedia.com/support.</p>
Sample Code	<pre>on (release) { fscommand("MTSSetProperty", "Target=MTSInstance.Cube1::clps; Value="1"); }</pre>

ifFrameLoaded()

Description	Checks whether the contents of a specific frame are available locally.
Syntax	<pre>ifFrameLoaded(expression) { Statements }</pre>
Remarks	Functions the same in Viewpoint Media Player as in Flash.
Reference	Read more about ifFrameLoaded() at: http://www.macromedia.com/support/flash/action_scripts/actions/iframeloaded.html .
Sample Code	<pre>ifFrameLoaded("start"){ GotoAndPlay("start"); }</pre>

tellTarget()

Description	Functions the same in Viewpoint Media Player as in Flash.
Syntax	<pre>TellTarget(target) { Statements }</pre>
Reference	Read more about tellTarget() at: http://www.macromedia.com/support/flash/action_scripts/actions/telltarget.html .
Sample Code	<pre>On (rollOver) { TellTarget("item1") { play(); }; }</pre>

gotoAndPlay()

Description	Functions the same in Viewpoint Media Player as in Flash.
Syntax	<pre>gotoAndPlay(frameNumber) gotoAndPlay(frameLabel)</pre>
Reference	Read more about gotoAndPlay() at: http://www.macromedia.com/support/flash/action_scripts/objects/movieclipgotoandplay.html .
Sample Code	<pre>gotoAndPlay(10);</pre>

gotoAndStop()

Description	Functions the same in Viewpoint Media Player as in Flash.
Syntax	<code>gotoAndStop(frameNumber)</code> <code>gotoAndStop(frameLabel)</code>
Reference	Read more about <code>gotoAndStop()</code> at: http://www.macromedia.com/support/flash/action_scripts/objects/movieclipgotoandstop.html .
Sample Code	<code>gotoAndStop(10) ;</code>

play()

Description	Functions the same in Viewpoint Media Player as in Flash.
Syntax	<code>play()</code>
Reference	Read more about <code>play()</code> at: http://www.macromedia.com/support/flash/action_scripts/actions/play.html .
Sample Code	<code>play() ;</code>

stop()

Description	Functions the same in Viewpoint Media Player as in Flash.
Syntax	<code>stop()</code>
Reference	Read more about <code>stop()</code> at: http://www.macromedia.com/support/flash/action_scripts/actions/stop.html .
Sample Code	<code>stop() ;</code>

nextFrame()

Description	Functions the same in Viewpoint Media Player as in Flash.
Syntax	<code>nextFrame()</code>
Reference	Read more about <code>nextFrame()</code> at: http://www.macromedia.com/support/flash/action_scripts/actions/nextframe.html .
Sample Code	<code>prevFrame() ;</code>

prevFrame()

Description	Functions the same in Viewpoint Media Player as in Flash.
Syntax	prevFrame()
Reference	Read more about prevFrame() at: http://www.macromedia.com/support/flash/action_scripts/actions/prevframe.html .
Sample Code	prevFrame();

New Coloring and Fill Features for Emboss Animations

Viewpoint Media Player already supports Emboss, an effect that allows you to add various combinations of highlights and shadows to a layer, creating effects such as 3D text. Viewpoint Media Player 3.0.12 supports a wider array of Emboss functionality that enables better text coloring and filling methods and more control when applying embossing to an alpha map or a texture.

Tag and Property Descriptions

Viewpoint Media Player 3.0.12 introduces the following Emboss tags and properties:

Tag	Description
<MTSTimeElem Type="Emboss"> <Text alph>	Determines how embossed text is rendered over an area that contains an alpha.
<MTSTimeElem Type="Emboss"> <Text clea>	Determines whether or not the texture clears each time an Emboss animation is updated.
<MTSTimeElem Type="Emboss"> <Text coll>	Replaces <Text ColorA> in setting the shading colors for the left side of the embossed text in RGB (red, green, blue) values.
<MTSTimeElem Type="Emboss"> <Text col2>	Replaces <Text ColorB> in setting the shading colors for the right side of the embossed text in RGB (red, green, blue) values.
<MTSTimeElem Type="Emboss"> <Text col3>	Fills the core color of embossed text in RGB (red, green, and blue) values.
<MTSTimeElem Type="Emboss"> <Text fill>	Determines whether the text core color, set by col3, is drawn or not in an Emboss animation.

Example Emboss Image Using New Features

The following graphic illustrates an example of the new Emboss features in Viewpoint Technology.



New Naming Convention Features for MTSObjectMovie Animations

Viewpoint Media Player already supports multiple textures (individual images) extracted from or simulating a QuickTime Pro object movie. Viewpoint Media Player 3.0.12 supports a more robust file naming functionality that allows for sequential numbering of object movie images and a more precise reference to an object movie image.

Tag and Property Descriptions

Viewpoint Media Player 3.0.12 introduces the following MTSObjectMovie tags and properties:

Tag	Description
<code><MTSTimeElem Type="MTSObjectMovie" BaseName></code>	Refers to the image name of a QuickTime object movie, devoid of any other descriptors, such as sequential numbering (similar to a ZoomView image).
<code><MTSTimeElem Type="MTSObjectMovie" Format></code>	Sets the sequential numbering convention used to name a sequence of QuickTime object movie images by specifying the first image number and the maximum number of digits the number can contain.

New Trigger Feature for Keyframe Animations

Viewpoint Media Player 3.0.12 introduces `<MTSTimeElem Type="Keyframe"> <Timeline OnlyOnce>`, a tag that triggers a keyframe animation at a specified time only one time until the time changes to a different keyframe.

For example, when a keyframe animation is set to `On="1"` and `Paus="1"`, if HyperView is initialized by setting the time of this keyframe animation to `2: 1 2 [0] [1]`, the action triggers at

Time=2 and sets `MTSBaseComponent::smode=1`. After leaving HyperView via some specified event, HyperView reinitializes because the keyframe is triggered again at Time=2.

Since the keyframe animation is paused, it continues to get triggered at Time=2 over and over. To avoid this, set the `OnlyOnce` property to: `1 2 [0] [1]`, thereby triggering the keyframe animation at Time=2 only once until the time has changed to a different keyframe.

New Animation Property: FromXMLversion

Viewpoint Media Player 3.0.12 supports a new read-only property, `FromXMLversion`, that gets and displays the MTX version number from a scene. `FromXMLversion` specifies the MTX version from which a scene element or "atom" originates. Use in conjunction only with a `GetProperty` action.

This property helps you work more efficiently within Viewpoint Technology scenes that have multiple version numbers. For example, consider loading a number of secondary .mtx files comprising multiple version numbers into a master .mtx file. If each .mtx file contains an object with a Flash texture, you could use `FromXMLversion` to create an onclick action for each object to return the version number for each object's MTX version. Thus, one could examine how Flash gets displayed in Viewpoint Technology within a number of different MTX versions.

New HyperView Features

Viewpoint HyperView technology lets you "color outside the lines." Your Viewpoint Technology-enabled website can burst beyond the edges of the current window or frame to let your message pop. HyperView frees animations from the constraints of a browser window, allowing such effects as a car animation taking a drive across the bottom of a web viewer's desktop before it parks itself back in its original window.

Note: HyperView can be used with any rich media content played in Viewpoint Media Player: ZoomView, 3D, Flash, object movies, etc.

Viewpoint Media Player 3.0.12 introduces two new HyperView tags, `<DrawAnywhere>`, which enables AOL content covered by `DrawAnywhere` to remain active, and `<HyperView>`, which enables a Viewpoint Technology scene to initiate in HyperView mode.

Viewpoint Media Player 3.0.12 also introduces the Viewpoint Technology special function, `ClearScene`, which clears a Viewpoint Technology scene when called.

Note: For more information on these features, see the [Viewpoint XML Reference Guide](#).

Introducing HyperView for AOL Rainman

Similar to `DrawAnywhere` for HTML, `HyperView` for AOL Rainman (`<MTSBaseComponent DrawAnywhere>`) allows content beneath or covered by the Viewpoint Technology HyperView window to remain active. As a result, ads and links stay active while the Viewpoint Technology window is open. Outside of AOL forms, `<MTSBaseComponent DrawAnywhere>` enables HyperView.

New MTSBaseComponent Feature for Non-AOL HyperView

Viewpoint Media Player now supports the newly created <HyperView> tag with which a developer can initiate a Viewpoint Technology scene in HyperView mode. Prior to Viewpoint Media Player 3.0.12, HyperView had to be initiated via the smod property, which could be set to activate HyperView after the Viewpoint Technology scene finished loading.

Introducing Special Viewpoint Technology Functions

Special Viewpoint Technology functions are functions built into Viewpoint Media Player that need only to be called into action to perform their specified functions. Viewpoint Media Player 3.0.12 introduces the ClearScene function, a special function linked to MTSBaseComponent, which clears a scene when called. Use the ClearScene function to clear a Viewpoint Technology scene from Viewpoint Media Player, leaving Viewpoint Media Player in the same state as when it was instantiated.

To call ClearScene, use <MTSAction> <VETDispatchCall Function>, which follows an object-oriented model that is similar to JavaScript or other object-oriented programming languages. For example, the syntax looks like:

```
<VETDispatchCall Function="MTSBaseComponent::ClearScene()" />
```

For more information about <VETDispatchCall Function>, see [“Specify a Special Function Call”](#).

Note: To date, ClearScene is the only special Viewpoint Technology function available.

New Scene Features

MTSScene is the root element of an .mtx file and follows the XML header information. This means that all other XML elements in a Viewpoint Technology scene are contained between MTSScene opening and closing tags. Tags in MTSScene allow you to set the Viewpoint Media Player version for which the scene is created, as well as a custom content version number.

Viewpoint Media Player 3.0.12 introduces new MTSScene tags and properties, explained below in detail, including:

- MTSCookie tag suite
- MTSSetProperty tag InitOnly
- MTSScene related property: camera

Note: For more information on these features, see the [Viewpoint XML Reference Guide](#).

Introducing MTSCookie

Just like browser cookies, the MTSCookie tag suite allows you to save variables or properties between runs of Viewpoint Media Player, allowing content to share its cookie information with other content, while providing security controls to prevent unauthorized content from reading the cookies.

Tag and Property Descriptions

Viewpoint Media Player 3.0.12 introduces the following MTSCookie tags:

Tag	Description
<code><MTSScene> <MTSCookie></code>	Saves variables or properties between various active Viewpoint Media Player instances. In addition, provides so that all VET content sharing the same broadcast key file also shares its cookies and furnishes security controls to prevent Viewpoint Technology content that does not share broadcast key files from accessing those cookie values.
<code><MTSScene> <MTSCookie> <MTSCookieTime></code>	Sets the lifetime of MTSCookie.
<code><MTSScene> <MTSCookie> <MTSCookieTime Category></code>	Groups cookies together. Also, determines the MTSCookie category for which you set the lifetime of the cookie.
<code><MTSScene> <MTSCookie> <MTSCookieTime ExpireNow></code>	Expires MTSCookie values immediately, rendering them unavailable.
<code><MTSScene> <MTSCookie> <MTSCookieTime Lifetime></code>	Sets the length of time (in seconds) that MTSCookie is available.

New Set Property Feature

The MTSScene tag `<MTSSetProperty>` sets the value for a property that you define and also assigns values to Viewpoint Media Player-defined properties. Although you can use this tag in any scene where you are creating and setting your own property, you would most likely use it with a cookie property.

The property becomes attached to the scene via the `<MTSPProperty>` tag as soon as the scene loads. However, you can change the value of your property via `<MTSSetProperty>`.

Viewpoint Media Player 3.0.12 introduces `<MTSScene> <MTSSetProperty InitOnly>`, which sets a property only if it does not already exist in the scene or as a cookie. When set to `Init="1"`, it ensures that once the value is set, it cannot be changed by `<MTSSetProperty>`.

New MTSScene Property: Camera

The Camera property switches a scene's camera from one camera to another, and is especially useful when employed with the new VETCamera model, described below.

New Scene Parameters Features

The `<MTSSceneParms>` element is where global settings for a Viewpoint Technology scene are defined, such as rendering mode, shadowing, procedural lightmaps, and anti-aliasing.

Note: For more information on these features, see the [Viewpoint XML Reference Guide](#).

Introducing Cache Content Updating

Viewpoint Media Player 3.0.12 introduces a new function, `<MTSSceneParms FileLifetime>`, that updates cache content. Content can be updated via several different settings: in seconds, minutes, or days. For example, there are 86,400 seconds in a day, so `Lifetime="86400"` means Viewpoint Media Player asks the server only if the file has changed once per day. Another example, `Lifetime="7D"` means that Viewpoint Media Player asks the server for updates once a week.

When to Use

Use these features if the content changes frequently, or if the content might change and it is important that the user sees the newest version. However, realize that using this tag actually can delay how fast content can be viewed; viewing content out of cache can be slightly faster than checking the server (higher load on the server).

Note: Can be specified in HTML as well.

Tag and Property Descriptions

Viewpoint Media Player 3.0.12 introduces the following cache-updating `MTSSceneParms` tags and properties:

Tag	Description
<code><MTSSceneParms FileLifetime></code>	Specifies how often Viewpoint Media Player checks for newer versions of Viewpoint Technology content on the server.
<code><MTSSceneParms ForceNoProxy></code>	Gets content changes directly from the server by bypassing proxy servers that might cache outdated content.

New Camera Features

Viewpoint Media Player supports a camera model, `<MTSCamera>`, which determines how a scene's camera records an action in a scene based on user interaction. `<MTSCamera>` allows identical objects to display differently in each scene because of camera transformations alone, even if the only part of each scene changing is the camera.

Effective Viewpoint Media Player 3.0.12, Viewpoint introduces a new camera model, `<VETCamera>`, which determines how the camera views a scene through animation.

Note: For more information on these features, see the [Viewpoint XML Reference Guide](#).

Introducing VETCamera

`<MTSCamera>` reacts to mouse interaction while the new `<VETCamera>` tags enable the camera to view a scene via camera animation. This difference offers you more flexibility in directing a Viewpoint Technology scene. In addition, the `VETCamera` model behaves exactly like a regular 3D instance in a scene, meaning it can be inserted anywhere in the instance hierarchy, having parents and children, and its location (the matrix) can be set in the same manner as an instance.

While the <VETCamera> model can be used as a child of an instance (or another camera) and can have any number of children, it cannot be the parent or child of <MTSCamera> because <MTSCamera> supports only one 3D instance of parent and child. The new <VETCamera> matrix properties allow an instance's matrix to be copied from one to another.

Note: <VETCamera> can be referred to as "MTSCamera" (for example, MTSCamera::SOMETHING); however, the <MTSCamera> tag refers to the interactive camera model.

Tag and Property Descriptions

Viewpoint Media Player 3.0.12 introduces the following VETCamera tags and properties:

Tag	Description
<VETCamera Name>	Specifies the VETCamera name.
<VETCamera Rotation>	Sets the rotation of an entire VETCamera scene either around the x, y, and z axes by the number of specified radians or as a quaternion.
<VETCamera Scale>	Determines the size of all geometries in a VETCamera scene and allows you to alter the scale along each or all of the coordinate axes (x, y, z).
<VETCamera Shear>	Skews the camera in a VETCamera scene globally along any axis not set to 0 (zero).
<VETCamera Translation>	Specifies the VETCamera position along the x, y, and z axes, thus shifting the camera's apparent position in the Viewpoint Media Player window.
<VETCamera> Related Property Matrix	Specifies the position, rotation, scale, and shear of the camera in space, subject to the camera's hierarchy.
<VETCamera> Related Property WorldMatrix	Specifies the position, rotation, scale, and shear of the camera in space, independent of the camera's hierarchy.
<VETCamera> Related Property InvMatrix	Specifies the inverse position, rotation, scale, and shear of the camera in space, subject to the camera's hierarchy.

New Interactors Features

The Viewpoint Technology interactor system allows scripting of system and user events (such as OnClick) that trigger actions altering a scene. The Viewpoint Technology interactor system adds some of the power of JavaScript with the ease of XML scripting.

Viewpoint Technology interactors are expressed within several elements, including MTSAction, MTSHandle, MTSInteractor, and MTSPreload. Viewpoint Media Player 3.0.12 supports new features within these elements, such as the <MTSAction> <VETDispatchCall Function> tag, which calls the MTSBaseComponent special function, ClearScene, to clear a

Viewpoint Technology scene. Viewpoint Media Player also includes enhanced support for the following existing interactor features:

- String value concatenation and set property functionality for <MTSAction>, see [“New MTSAction String Value Concatenation Feature”](#) and [“New MTSAction Set Property Feature”](#).
- Animation controls for <MTSInteractor>, see [“New MTSInteractor Animation Control Feature”](#).
- Prioritization for <MTSPreload>, see [“New MTSPreload Prioritization Feature”](#).

Note: For more information on these features, see the [Viewpoint XML Reference Guide](#).

Specify a Special Function Call

Viewpoint Media Player 3.0.12 introduces <MTSAction> <VETDispatchCall Function>, which is a new tag that allows object-oriented function calls to be specified. To date, only one function call, ClearScene, is available in Viewpoint Media Player. VETDispatchCall follows an object-oriented model that is similar to JavaScript or other object-oriented programming languages. Thus, functions can be directly related to media resource files.

Previously, properties were assigned to run functions, such as trigger (trgr="1", etc.).

Note: Call the function ClearScene with no parameters on MTSBaseComponent.

New MTSAction String Value Concatenation Feature

Viewpoint Media Player 3.0.12 introduces the Append attribute for the <MTSAssignProperty>, <MTSCopyProperty>, and <MTSSetProperty> tags, which allows you to add strings together rather than overwrite them. Append is especially useful with calling URLs ("pinging").

Also, use Append to concatenate hot spot text strings. For instance, hot spot text reading, "Lid is", can be appended to new text, such as "open" or "closed", giving a new message: "Lid is open/closed".

New MTSAction Set Property Feature

In previous Viewpoint Media Player versions, you could use the <MTSAssignProperty> and <MTSSetProperty> tags in the .mtx file for your scene to manage property settings and definition. Viewpoint Media Player 3.0.12 introduces <MTSInitProperty> to set a value property that you define. It is primarily used in conjunction with Viewpoint MTSCookies.

Note that the differences between using MTSAssignProperty, MTSSetProperty, and MTSInitProperty are subtle but important. The difference lies in how you want user interaction to be recorded in the user's Viewpoint cookies. For instance, if you want to record the number of clicks a user makes on an object per session, MTSAssignProperty records new clicks only if the user has previously clicked and the event(s) have been recorded, MTSSetProperty either adds the clicks to a previous session or sets the clicks for the first time, and MTSInitProperty records the clicks only if the user has never interacted via clicks before.

Also, each tag handles a property that does not exist differently. Comparing the three different tags:

- MTSAssignProperty sets a property if it exists, but does nothing if it does not.
- MTSSetProperty sets the property if it exists and attaches and sets the property if it does not.
- MTSInitProperty does nothing if the property exists, but attaches and sets the property if it does not.

New MTSInteractor Animation Control Feature

Viewpoint Media Player 3.0.12 introduces a new event-type associated with Viewpoint animations, MTSAnimationLoop, which is activated when an animation enters a new loop. To find out how many loops an animation has entered, request the property lpmn of this event. Expressed in MTX code, its syntax looks like:

```
<MTSInteractor> <MTSHandle Event="MTSAnimationLoop">
```

The MTSAnimationLoop event now works for every Viewpoint animation except for those whose durations are unknown. It is called via the <MTSHandle Event> tag, which includes dozens of events. In particular, several animation events already exist, including MTSAnimBegin and MTSAnimEnd.

New MTSPreload Prioritization Feature

The MTSPreload functionality was introduced in Viewpoint Media Player version 3.0.8 as a means to allow a preloader file for a Viewpoint Technology web application to be specified in the MTX code. The MTSPreLoad element can be used to load files in the background to provide a faster and better user experience.

Viewpoint Media Player 3.0.12 introduces a new MTSPreload attribute, Priority, which specifies the order in which preloaders are loaded. This feature allows you more control about what resource files are uploaded first.

Tip: When building Viewpoint Technology scenes of various .mtx files, place your preloader in either the first or second .mtx file to ensure that it is processed quickly.

New Interactor Property: FromXMLVersion

This property, described above, functions the same for interactors as it does for animations.

New Hot Spot Features

Hot spots, also known as widgets, are a great way to annotate a scene. Hot spots function when a user mouses over a hot spot geometry, text, or a texture display. For example, one can highlight the features of a car by adding hot spots with text or a Flash texture to the car surface.

A subelement of MTSInstance, <LayerData> and its attributes enable you to customize hot spots. Viewpoint Media Player 3.0.12 introduces the <Transform> subelement to <LayerData>; it sets the size, orientation, skew, and position of hot spot text.

Note: For more information on these features, see the [Viewpoint XML Reference Guide](#).

Introducing LayerData Text Transform

A complex subelement with four nested attributes, `<LayerData> <Transform>` gives developers more flexibility and control over hot spot text rendering. Certain settings for its attributes may render text unreadable (`<Shear>` and `<Rotate>`) or cause it to display outside the scene window (`<Scale>` and `<Position>`).

Tag and Property Descriptions

Viewpoint Media Player 3.0.12 introduces the following `<LayerData> <Transform>` tags and properties:

Tag	Description
<code><LayerData> <Transform> <Position></code>	When nested within a <code><LayerData> <Transform></code> tag, <code><Position></code> translates the text of the hot spot along the specified vector.
<code><LayerData> <Transform> <Rotate></code>	When nested within a <code><LayerData> <Transform></code> tag, <code><Rotate></code> sets the rotation of the text in a hot spot around the x, y, and z axes by the number of degrees specified.
<code><LayerData> <Transform> <Scale></code>	When nested within a <code><LayerData> <Transform></code> tag, <code><Scale></code> determines the size of the text in a hot spot and allows you to alter the scale along each or all of the coordinate axes (x, y, z).
<code><LayerData> <Transform> <Shear></code>	When nested within a <code><LayerData> <Transform></code> tag, <code><Shear></code> skews text of a hot spot along any plane not set to 0 (zero).

New Instance Features

An `MTSInstance` element contains the definition of a 3D object in a Viewpoint Technology scene, by specifying transformations, geometry, attributes, materials, texture, and lightmaps. The instance element generally contains subelements `Transform`, `MTSGeometry`, `MTSMaterial`, and `MTSTextureMap`. An instance element may also contain hot spot text defined by a `Layer2D` geometry and a `LayerData` subelement.

A subelement of `<MTSInstance>`, `<Transform>` and its attributes enable customization of instances. Viewpoint Media Player 3.0.12 introduces a new set of transform-like attributes to `<MTSInstance>`, which no longer require nesting in the `<Transform>` tag. These attributes set the size, orientation, skew, and position of instances.

Note: For more information on these features, see the [Viewpoint XML Reference Guide](#).

Introducing MTSInstance Transform and Matrix Features

A complex subelement with four nested attributes, `<MTSInstance>` `<Transform>` has given developers flexibility and control over instance rendering. Viewpoint Media Player 3.0.12 introduces a new way to express these functions without nesting them in a `<Transform>` tag. The new matrix properties allow an instance's matrix to be copied from one to another.

Tag and Property Descriptions

Viewpoint Media Player 3.0.12 introduces the following `<MTSInstance>` tags and properties:

Tag	Description
<code><MTSInstance Translation></code>	Specifies the camera's position along the x, y, and z axes, thus shifting the camera's viewable position in the Viewpoint Media Player window.
<code><MTSInstance Rotation></code>	Determines the instances rotation either around the x, y, and z axes by the number of specified radians or as a quaternion.
<code><MTSInstance Scale></code>	Alters the scale of an instance along each or all of the coordinate axes (x, y, z).
<code><MTSInstance Shear></code>	Skews the instance along any axis not set to 0 (zero).
<code><MTSInstance></code> Related Property Matrix	Specifies the position, rotation, scale, and shear of the instance in space, subject to the instance's hierarchy.
<code><MTSInstance></code> Related Property WorldMatrix	Specifies the position, rotation, scale, and shear of the instance in space, independent of the instance's hierarchy.
<code><MTSInstance></code> Related Property InvMatrix	Specifies the position, rotation, scale, and shear of the instance in space, independent of the instance's hierarchy.

Chapter 3: General Feature Notes

This chapter addresses several topics, ranging from specific Viewpoint Technology feature behaviors in Viewpoint Media Player 3.0.12, such as the curs tag, to the effect of ad-stopper utilities on Viewpoint Technology content, providing valuable and time-saving workarounds, tips, and troubleshooting methods for Viewpoint Media Player release 3.0.12.

AOL-Related Notes for Viewpoint Media Player 3.0.11 and 3.0.12

The AOL-specific Viewpoint Media Player release, 3.0.11, supports compressed .mtx files (the .mtz file format).

Also, for reference, the following code sample displays the appropriate format for linking to a web page from within Viewpoint Technology AOL Rainman content.

```
<MTSInteractor Name="GoToAdTarget" >  
  <MTSHandle Event="MouseLeftClick" Action="MTSVBScript"  
    Target='aol.GoToKeyword  
"aol://4344:1723.Sharks.14683563.673275171"' />  
</MTSInteractor>
```

Setting the Viewpoint Media Player Cursor in Viewpoint Media Player 3.0.12

When setting a Viewpoint Media Player-defined cursor, choose between two XML tag options, curs and crsr. For the best results in Viewpoint Media Player 3.0.12, we recommend you use the curs property to set a Viewpoint Media Player-defined cursor. Use the crsr property to set the default cursor (value="0").

The tag, curs, sets the cursor in a scene to a Viewpoint Media Player-defined cursor based on a string value. For example:

```
<OnEnter Action="MTSAssignProperty" Target="MTSScene::curs"  
Value="MTS_Point" />
```

The tag, crsr, sets the cursor in a scene to the default cursor based on an integer ID ("0") or a Viewpoint Media Player-defined cursor based on a string value. For example:

```
<!--With an integer: -->  
<OnEnter Action="MTSAssignProperty" Target="MTSScene::crsr"  
Value="0" />  
<!--With a string value: -->  
<OnEnter Action="MTSAssignProperty" Target="MTSScene::crsr"  
Value="MTSCursor::MTS_Point" />
```

Note: Using an integer ID other than "0" can yield differing results for different operating systems.

Impact of Pop-up Stopper™ and AdAware™ on Viewpoint Media Player 3.0.12

In Windows 98, Pop-up Stopper on Netscape Navigator prevents additional Viewpoint Technology windows from opening. Pop-up Stopper does not specifically target Viewpoint Technology content, but it affects Viewpoint Technology content because much of Viewpoint Technology content launches a new browser window in which the content is presented.

To date, AdAware does not affect Viewpoint Technology content, even when conducting a full system scan. AdAware seems to target cookies and similar functions.

Appendix A: Help, Resources, and Feedback

Viewpoint Developer Central: A Complete Resource

[Viewpoint Developer Central](#) is a complete resource for Viewpoint Technology content developers. At this website, you can access Viewpoint applications, user guides, downloadable example files, support, production tips, and techniques – to name just a few of the offerings there.

Access to Viewpoint Developer Central to:

- Get Assistance For questions about using Viewpoint Technology, click **Forums** under **Support** in the left navigation bar.
- Get Examples Click **Examples & Tips** in the left navigation bar.
- Subscribe to the Viewpoint Developer Newsletter Learn new production tips and techniques for creating 3D and rich media content for the web with Viewpoint Technology. Click **Newsletter** in the left navigation bar.
- Give Feedback About Viewpoint Applications Viewpoint Corporation values your feedback. Direct your comments and suggestions to the Viewpoint Forums.

You can also visit the [Viewpoint Corporation website](#) for company news, links to websites featuring Viewpoint Technology, and more.

Download Viewpoint Applications, Guides, and Examples

Viewpoint Developer Central is updated continuously with the latest versions of applications, user guides, and examples. Find links to the following in the left navigation bar.

Viewpoint Applications

You can download Viewpoint applications free of charge. The following applications are available for download:

- Viewpoint Media Player The web browser plug-in necessary to view Viewpoint content with Netscape Navigator or Internet Explorer.
- Viewpoint Scene Builder An essential application for assembling a scene and publishing it in .html/.mtx/.mts format.
- Viewpoint Media Publisher An application enabling you to quickly create Viewpoint Technology web applications from Viewpoint media files (.mtx/.mtz) by embedding them in web (html) pages or running transformations on .mtx (xml) files through built-in XSLT support.
- Viewpoint Stream Tuning Studio An application for reducing .mts file sizes, enabling optimized 3D scenes rendered on a web page to stream quickly and retain visual integrity.
- Viewpoint Control Panel A utility for checking, installing, and removing individual Viewpoint Media Player components.

User Guides

For more information on Viewpoint Technology and related utilities, check out these documents on [Viewpoint Developer Central](#) :

See this document	To learn about
Viewpoint Technology Feature Overview	What Viewpoint Technology is and how you can use rich media components-including 2D photos and drawings, 3D models, animation, ZoomView images, Macromedia Flash movies, video, text, and audio-to create rich media content.
Using 3D in Viewpoint Technology	Describes how to create a basic 3D Viewpoint Technology web application with simple interactivity and Flash effects. You'll learn how to take content created in a 3D modeling application, enhance it with Viewpoint Technology, and publish it to a web page.
Viewpoint Scene Builder User Guide	Offers step-by-step procedures for using Viewpoint Scene Builder to edit and assemble the components of a Viewpoint Technology scene before publishing it in the Viewpoint media file format.
Using High-Resolution 2D Graphics in Viewpoint Technology	This guide is a comprehensive reference for ZoomView, a technology that permits the deployment of large, print quality images online. The guide explains the three authoring paths to create ZoomView images, which include Adobe® Photoshop® 7, ZoomView Builder, and ZoomView Hosted Solutions. It also describes how to add advanced features to your ZoomView scenes by making changes to XML code in your .mtx file.
Viewpoint Stream Tuning Studio Guide	Explains how to use Viewpoint Stream Tuning Studio to optimize uncompressed .mts files from Viewpoint Scene Builder and other applications that export .mts files for Viewpoint Technology scenes.
Viewpoint Media Publisher Guide	New features of Viewpoint Media Publisher provide a fast and convenient path to the web for Viewpoint Technology content. Viewpoint Media Publisher enables you to quickly create Viewpoint Technology web applications from Viewpoint Media Files (.mtx/.mtz) by embedding them in web (html) pages or running transformations on .mtx (xml) files. This guide includes step-by-step instructions for creating Viewpoint Technology-enabled web pages, customizing html templates, customizing .mtx files with XSLT, and publishing Viewpoint content.
Using Vector Graphics in Viewpoint Technology	Describes how to integrate Macromedia® Flash™ and Scalable Vector Graphics (SVG) with other rich media, such as animated 3D objects, high-resolution photographs, object movies, and panoramas into Viewpoint Technology scenes.

[Viewpoint Technology:
Technical Overview](#)

Describes the architecture of a Viewpoint Technology scene, including the key components that comprise it: objects, animations, events, and properties. You'll learn how to combine these components and organize them in the Viewpoint Technology scene .mtx description file.

[Deployment for System
Administrators](#)

Steps that Administrators can take to set up a web server with Viewpoint Technology content.

[Testing and Deploying
Viewpoint Technology
Content](#)

Describes how to test and prepare your Viewpoint Technology content for deployment. Details the MTS3Interface, the auto-installer, and their usage, making it an invaluable resource for Viewpoint content producers.

[Viewpoint XML Reference
Guide](#)

A powerful reference tool containing the most up-to-date information and examples for the XML tags and properties that control a Viewpoint Technology scene. Viewpoint Technology uses XML scripted in an .mtx file to orchestrate the elements of a scene, including animation, interactivity, and loading of files, as well as ZoomView and HyperView functionality.

Appendix B: Viewpoint Technology Content Creation Updates

Viewpoint Technology Content Checklist

Viewpoint Technology 3.0.12 content has certain features and authoring requirements for display on the Windows platform:

- **Install the latest version of Viewpoint Media Player (version 3.0.12).**

To ensure new Viewpoint Technology content displays as expected, install the latest version of Viewpoint Media Player from Viewpoint Developer Central:

<http://developer.viewpoint.com/developerzone/5-1.html>.

- **Use the latest version of MTS3Interface.js (currently version 4.0.5.4).**

To ensure new Viewpoint Technology content uses the correct version of MTS3Interface, you can publish it with the newest release of Viewpoint Media Publisher. Viewpoint Media Publisher contains the version of MTS3Interface that supports content viewed with Viewpoint Media Player 3.0.12 for Windows.

To find out more about using MTS3Interface, download the [Viewpoint Technology Deployment Guide](#) from Viewpoint Developer Central.

- **Correct DIV and Layer tags for Netscape 4.x.**

This involves adding sniffer script available on web developer resource sites to detect the web browser type and version.

Important: Do not create sniffer scripts that refer to variables (such as isIE or isNN) set in MTS3Interface.js, as this JavaScript file is modified periodically to accommodate new functionality, and variables contained in it may change.

- **Update required settings for your web page's Viewpoint Technology content.**

The type of rich media content you include in your web page dictates the components you must specify in your constructor call. See the Viewpoint Experience Technology Deployment Guide for information on how to refer to these requirements for your content.

- **Create content to avoid the opening of multiple browser windows containing Viewpoint Technology content.**

For best performance, only one browser window containing Viewpoint Technology content should be open at a time.

Updating Existing Web Pages with MTS3Interface

MTS3Interface version 4.0.5.4 or higher is required for content you want to view with Viewpoint Media Player 3.0.12. You can quickly update any existing web pages (.html) that use the object/embed tag to use the MTS3Interface.js file and the constructor tag. The MTS3Interface dynamically builds an object embed and assigns a variable to it. It also facilitates interaction between Viewpoint Media Player and the web page, bi-directionally, across a variety of browsers and operating systems.

If you are already using MTS3Interface, simply replace the old file. If you are using an object embed style that does not refer to MTS3Interface, you need to update your HTML. Simply replace the old object/embed with a Viewpoint Media Player constructor line, and refer to the MTS3Interface in the head of your document. For step-by-step instructions, refer to the [Viewpoint Technology Deployment Guide](#).

Example for the New Naming Convention

To change JavaScript calls between the HTML page and Viewpoint Media Player to the new naming convention:

- 1 With the name of your object variable, "vmp", by default:
Change: onclick="triggeranimation('anim1')
To: onclick="vmp.TriggerAnim('anim1')
- 2 Remove all previous functions which are now unused from the HTML.

Using One MTS3Interface.js File for a Web Site

Using one MTS3Interface.js file for a website allows you to easily take advantage of updates and improvements to this file's support for Viewpoint Technology scenes. You need only to replace one file to update.

This also benefits users viewing your Viewpoint Technology content: The MTS3Interface.js file is cached after a user views the first Viewpoint Technology scene on your site, thereby speeding up all other pages viewed.

Verifying the MTS3Interface Version in Existing Content

When experiencing problems with Viewpoint Technology 3.0.12 content, check which version of MTS3Interface the content uses. This method is applicable if the content:

- Scripts in/out of the Viewpoint Technology scene with JavaScript
- Uses layers
- Runs in Netscape 7 (or on a Mac or Compuserve)

To check what MTS3Interface a piece of content is using:

- 1 Type the following command into the browser address line while the web page displays:

```
javascript:alert(VET_IfVer)
```

- 2 Look for the displayed value of the VET_IfVer variable, which is in all of the MTS3Interface files from version 4 onwards.

If a lower number than 04.00.05.04 displays, this could be the problem for why the content is not performing as expected.

If the VET_IfVer command gives an error and an error display box appears:

- 1 Click No in the error message box that displays asking if you want to debug.
- 2 Then, use the command:

```
javascript:alert(_ver)
```

Note: If this process still does not get the version of MTS3Interface used by Viewpoint Technology content, one of two possibilities explains why: 1) you are using a very outdated MTS3Interface version; or 2) you are not using the MTS3Interface at all.