Template Guide

GR*i*NS Editor for RealONE
Windows-98/2000/XP
November, 2001

© 2001 Oratrix Development bv. All rights reserved.

GR*i*NS Editor for RealONE Template Guide for Windows-98/2000/XP. November, 2001.

The software described in this manual is furnished under license and may be used only in accordance with the terms of that license.

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form without the written permission of Oratrix Development bv.

GRiNS, *GRiNS Editor for Real*, *GRiNS Editor for IE, GRiNS Editor for 3Gwireless*, *GRiNS/Pro* and *GRiNS/SMIL* are trademarks of Oratrix Development.

RealAudio, RealVideo, RealText, RealPix, RealMedia and RealONE Player are trademarks of RealNetworks, Inc. Windows, Windows-2000, Windows-95, Windows-98 and InternetExplorer are registered trademarks of Microsoft Corporation.

Important Notices

This document is the GR*i*NS Editor for RealONE *Template Guide*. All of the information presented in this document has been verified, but incremental product updates may impact part of this guide.

This version of the GR*i*NS Editor for RealONE *Template Design Guide* has been produced for use as an off-line reference. Images and page layout have been optimized for printing on a 600-dpi (or greater) laser printer. For best reproducibility, the use of a color printer is recommended, although every effort has been made to make illustrations readable on other printers as well. If you wish to use it as an on-line reference via a PDF reader, we recommend that you increase the level of display magnification when viewing images.

The images used in this publication were taken from the GR*i*NS Editor for RealONE build 2.0-win32-93 for Windows 2000. While the look of other versions of GR*i*NS are slightly different because of adherence to common conventions on those other environments, the functionality described is similar for all versions of GR*i*NS. In order to reduce document size, only images from the Windows version have been included in this document.

We welcome your questions on GR*i*NS Editor for RealONE and comments on this documentation. Please submit all questions and comments to our support desk at *grins-support@oratrix.com*. We maintain a list server dedicated to sharing experiences among GR*i*NS Editor for RealONE users. See the on-line release notes that come with the software distribution for details of this listserver. Finally, if you wish to submit your own SMIL files as examples for other users, please send a request for submission to: *grins-examples@oratrix.com*.

Table of Contents

Important Notices ..... **iii**

GR*i*NS Editor for RealONE Distribution Roadmap ..... **vii**
Distribution Package Contents vii
Documentation vii
GR*i*NS and GR*i*NS/RealONE viii
GR*i*NS/RealONE Template Guide viii

Overview of GR*i*NS/RealONE Templates **1**
GR*i*NS/RealONE Template Collection 1
Accessing GR*i*NS/RealONE Templates 2

GR*i*NS/RealONE Slideshow Templates **3**
Basic Slideshow with Background Music 3
Slideshow with Background Music and Audio Captions 8
Slideshow with Background Music, plus Audio and Text Captions 11

GR*i*NS/RealONE Video Templates **13**
Video Show (240x180) 13
Video Show (320x240) 17

GR*i*NS/RealONE Adaptive Templates **23**
Adaptive Video Presentation 23

GR*i*NS/RealONE Empty Templates **27**
Empty RealONE Presentation 27

GR*i*NS Edition Comparison Information **31**
GR*i*NS Packaging Information 31

GR*i*NS/RealONE Quick Reference Information **37**
SMIL Compliance Information 37
Supported Media Table 37
RealONE Media Conversion 39
References and Links 39

Where Next? **41**

GR*i*NS Editor for RealONE Distribution Roadmap

Congratulations on selecting the GR*i*NS Editor for RealONE (GR*i*NS/RealONE) for creating SMIL-2.0 presentations. GR*i*NS/RealONE lets you harness the full power of the RealNetworks RealONE player in an easy and intuitive manner.

Distribution Package Contents

The GR*i*NS/RealONE distribution package consists of the following components:

- GR*i*NS Editor for RealONE *Quick Start Guide*: an overview of the installation instructions for GR*i*NS and a tour of the basics of the GR*i*NS Environment.
- *Templates*: a set of templates used in GR*i*NS/RealONE, which you may use to build your own presentations;
- GR*i*NS-*Icons*: a directory containing Icons used by GR*i*NS/RealONE;
- *Software:* depending on the distribution you downloaded, a GR*i*NS distribution for Windows-95/98/2000/XP or other platform.

Documentation

In order to reduce the size of the GR*i*NS distribution set, most documentation has been included in a separate distribution file. Theses manual, available via the documents link at *http://www.oratrix.com/*, consist of:

- GR*i*NS Editor for RealONE *Reference Manual*: A comprehensive description of the GR*i*NS/RealONE features and options;
- GR*i*NS Editor for RealONE *Template Guide*: A quick and easy description of the standard GR*i*NS/RealONE template set can get you started making RealONE presentations.
- GR*i*NS Editor for RealONE *Tutorials Guide*: A set of step-by-step instructions for creating a wide range of RealONE presentations.

The documentation set also contains an assets set that you can use to build tutorial presentations.

GR*i*NS and GR*i*NS/RealONE

GR*i*NS is available in several configurations: a *Pro* edition, intended for users who want full control over authoring and publishing to various SMIL-2.0 engines, and editions for particular streaming players (such as GR*i*NS/RealONE).

In the examples and text given in this document, we will use "GR*i*NS/Pro" to identify features available only in the *Pro* edition of GR*i*NS, and "GR*i*NS/RealONE" to identify features available in the RealONE edition of GR*i*NS. Unless otherwise marked, all features in GR*i*NS/RealONE are available in GR*i*NS/Pro.

GR*i*NS/RealONE Template Guide

After reading the GR*i*NS/RealONE *Quick Start Guide,* we encourage you to build some sample presentations with the GR*i*NS/RealONE Template set. The *Template Guide* has been written to help you on your way quickly. It is divided into seven sections:

1. *Overview of GRiNS/RealONE Templates*: a summary description of the available template types and their purpose.
2. *Slideshow Templates*: a description of how you can build simple and not-so-simple slideshows with GR*i*NS.
3. *Video Templates:* Got some footage that you are dying to get on the Web? We'll show you how RealONE can help add some spice.
4. *Adaptive Templates:* The Web is a complex place — different people, different bitrates, different languages. GR*i*NS can help manage this complexity without driving you crazy.
5. *Empty Templates:* Want to start from scratch? We'll show you how.
6. GR*i*NS *Version Comparison Information*: an overview of the differences in functionality offered in the various editions of GR*i*NS.
7. GR*i*NS *Editor Quick Reference*: an overview of the data types supported by the GR*i*NS/RealONE environment and other useful reference information for getting started with the GR*i*NS/RealONE Editor.
8. *Where Next?*: A set of pointers to GR*i*NS/RealONE resources and other hints to help you on your way to creating presentations for the RealONE Player.

Each of the sections has been written to be relatively stand-alone, but we suggest that you read them all to get a good overview of the system.

Overview of GR*i*NS/RealONE Templates

The GR*i*NS/RealONE comes with a basic set of design templates that can help you build presentations quickly. The RealNetworks RealONE player is based on the SMIL-2.0 language, developed by the World Wide Web consortium (W3C). The SMIL-2.0 specification is an imposing document that describes a wealth of features for making your media presentations more attractive, effective and interactive than was previously available. The GR*i*NS/RealONE template set gives you access to the basic facilities of the language, and the GR*i*NS Editor provides unprecedented support for manipulating all of SMIL-2's constructs.

GR*i*NS/RealONE Template Collection

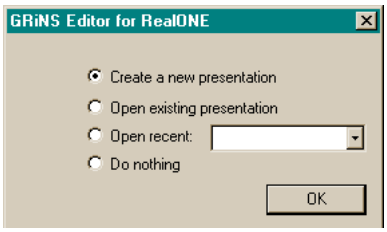
The basic set of templates delivered with the GR*i*NS/RealONE Editor contains four classes of templates:

- *Slideshow templates,* for creating presentations consisting mainly of images and (optional) music, spoken audio and text annotations;
- *Video templates*, for creating a sequence of one or more videos, with transitions and animations on video placement. The videos may also be accompanied with optional text and audio captions;
- *Adaptive templates*, for creating presentations that can "adapt" to the environment that the user has at the time of presentation delivery;
- *Empty templates*, for creating custom presentations without having to worry about the XML mumbo-jumbo required to format your presentation.

There are several types of templates in each class, each building upon the other. Over time, extra templates will be released by Oratrix or our GR*i*NS partners.

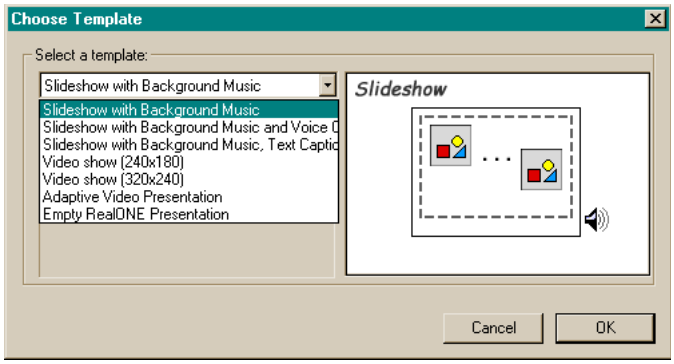
If you are interested in creating new templates, you may want to upgrade to GR*i*NS/Pro. GR*i*NS/Pro provides extra facilities for creating templates and for enhanced SMIL and RealONE editing support. GR*i*NS/Pro also allows you to create templates for other variants of SMIL, such as Microsoft's HTML+TIME in InternetExplorer 6, or the 3GPP variant of SMIL Basic. For upgrade information, see: *http://www.oratrix.com/*.

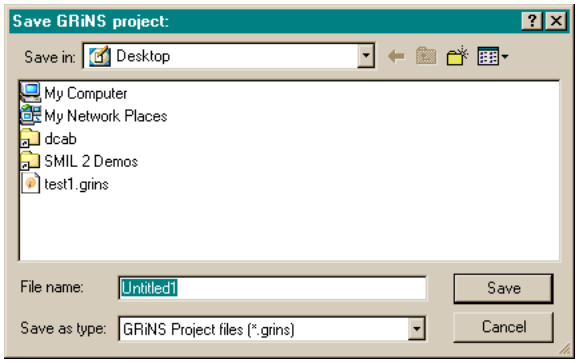
Accessing GR*i*NS/RealONE Templates



When you start the GR*i*NS/RealONE Editor, you will typically see a presentation file selection box. You can select a Template by choosing Create a new presentation, and then selecting Apply. (You can also select an existing presentation, or one of the recent files you were working on.) If you do not see this selection box, you can also open a Template via the *File->New* menu.

Whenever you select a New file, you get to choose from a list of available Templates. Each Template has a title, an identifying icon, and a short descriptive text. (Note that the list you see may not be the same as the one shown.)





After selecting a Template, you will be prompted for a location to save your work. The Template is saved as a *.grins* file by default. If you want to create a *.smil* file from the Template, you should use File->Publish once your presentation has been created. Saving the Template as a *.smil* file will cause all Template control information to be stripped — not a good idea.

GR*i*NS/RealONE Slideshow Templates

The GR*i*NS/RealONE Slideshow templates will help you create simple, smashing presentations with a minimum of effort. The presentations you create have all the power of Real's Slideshow product, but much more flexibility in assigning media types and interactive content.

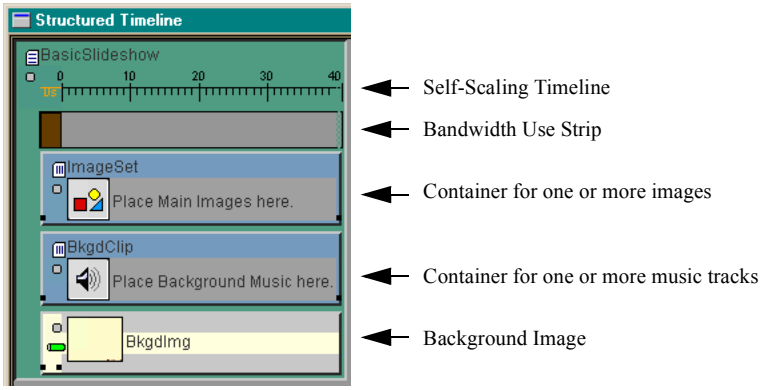
There are three Slideshow templates in the GR*i*NS/RealONE Template set:

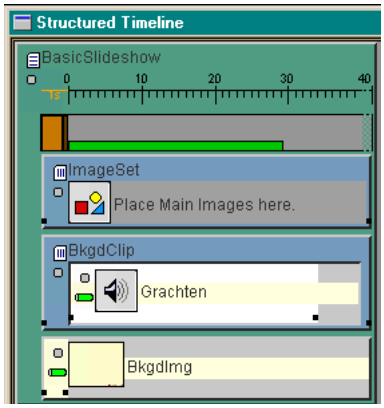
- Basic Slideshow with Background Music
- Basic Slideshow with Background Music and Audio Captions
- Basic Slideshow with Background Music and Audio plus Text Captions

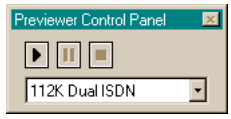
Before beginning to experiment with these templates, we strongly recommend that your read the *Quick Start Guide* for a introduction of GR*i*NS concepts.

Basic Slideshow with Background Music

The *Basic Slideshow with Background Music* template consists of two containers (or groups) to hold images and audio objects, plus one static background image. As with all other templates, there is a self-scaling Timeline, and a bandwidth usage strip. The empty version of the template is shown below. Look at the Bandwidth Strip: you'll see that the background image requires a short pre-roll. This is the time needed to load initial presentation assets.







With most templates, it is usually wise to put any timed media into the presentation first. For example, in the figure at left, we've added an audio object (*Grachten.mp3*), which is about 32 seconds long. Note that the bandwidth strip now shows the amount of bandwidth required *at the current preview bitrate setting* to present the presentation.

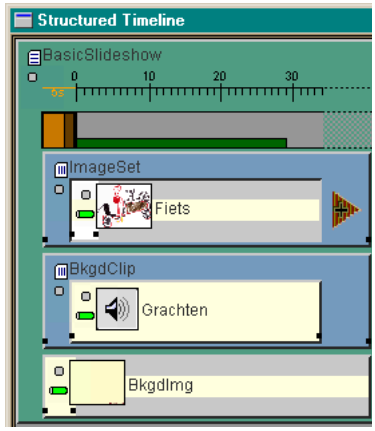
This bandwidth is determined by several factors, including information in the media encoding (if any) and the properties set on each object. The RealONE Player also determines default values. Note also that the pre-roll time has increased a bit.

By specifying longer (in duration) media objects first, the timeline scale can be optimized for use in visualizing the presentation. If we had first dropped an image onto the presentation, the time scale would have been different — not necessarily the timing of the presentation, just the time scale used by GR*i*NS to display the presentation structure.

A few of the important visual clues in the Structured Timeline view are:

- the drag handles on most media and structure objects, used to define start offsets and object duration,
- the green bandwidth pipes, which indicate that there is enough bandwidth available (at the current previewer setting) to deliver this object on time and without scheduling problems,
- the colored bar inside a media object that shows how the visual rendering of the object will be extended after the end of the object's *media duration* because of SMIL's timing and rendering fill rules,
- the use of a DropBox icon to indicate places in the template where the template designer expects you to add new media.

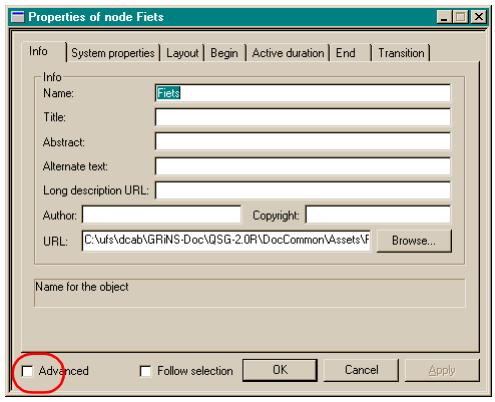
Many of these elements are brought together in the figure at right. Working from top to bottom, we see:



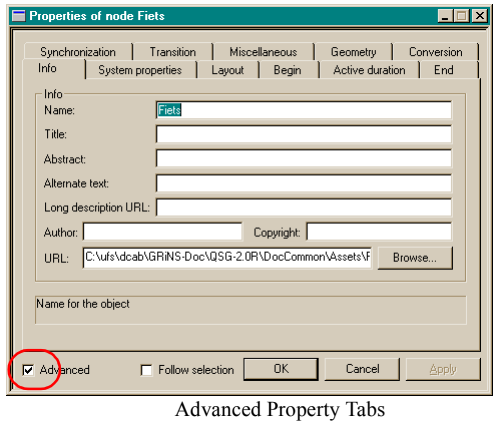
- the pre-roll time for this presentation, at the bitrate specified in the preview control panel, is 5 seconds. Since the images *Fiets* and *BkgdImg* need to be available at time 0s in the presentation, so their bandwidth needs are calculated into the pre-roll. The continuous media object *Grachten* can be streamed, meaning that we only need to wait a little bit of extra time before we can start rendering the show;
- the image *Fiets* can be presented without bandwidth problems (the green pipe). It has a duration of three seconds, and a fill behavior of freeze (meaning it stays rendered until the end of its context). In addition, the template author expects you to add more images, so a DropBox was placed inviting more objects;
- the audio object *Grachten* is about 32s long. Since it is the longest object, it determines the fill time of all of the other objects. Note that while there is no DropBox after *Grachten*, you can still add a media object that gets played after *Grachten* ends. The bandwidth analyzer doesn't expect any problems playing this object;
- the *BkgdImg* has a *defined duration of zero*. (Look closely at the timeline: the solid block of color that shows the media time takes up 0s of actual presentation time.) The fact that the object is visible is due to the fill="freeze" behavior of the media. Finally, we have enough time to load the object, since 5s of pre-roll is deemed acceptable.

When the template was created, the author determined reasonable defaults for most of the objects. For many presentations, you can simply drop in your media and publish.

You also have the option to override any presentation constraints by pulling up the Property sheet of an object. (This object can be either a media item or a structure group.) The properties are divided into *basic* properties and *advanced* properties. The basic properties available for image nodes is shown below at left, and the advanced properties are shown at right.

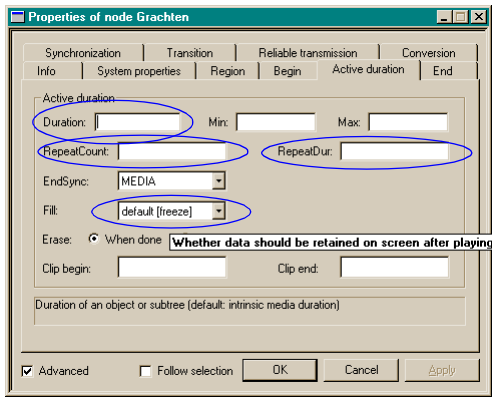


Basic Property Tabs



Advanced Property Tabs

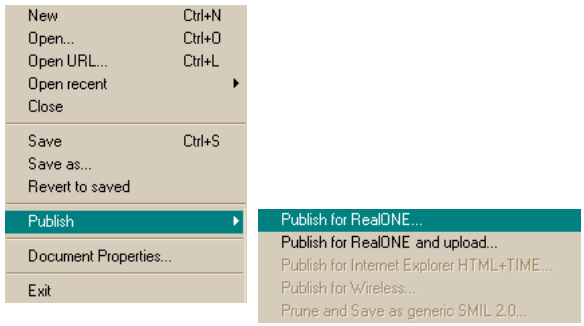
In order to effectively control all aspects of the RealONE presentation, you should take the time to read the *Real Production Guide* carefully.



One of the property tabs that is important to understand is *Active Duration*. This tab lets you define the value of various attributes associated with the duration of the object, and how it is rendered. (The advanced tab setting is shown; the attributes circled in blue are available in the basic setting.)

You should now populate the template with various sorts of media objects. You can preview the behavior of the presentation using the internal previewer (either the entire show or any individual object (or group of objects)).

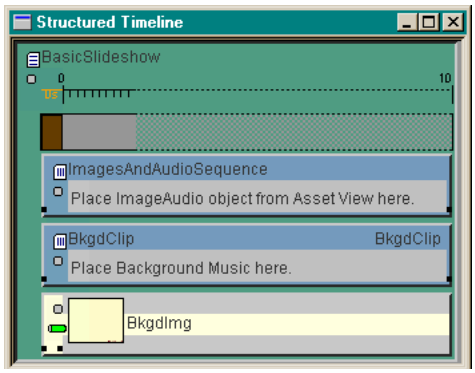
You should also use the File->Publish command and view the presentation in the RealONE Player, as shown below.



(Note: only those publishing options available in your configuration of GR*i*NS will be enabled. GR*i*NS/Pro will have all publishing options enabled.)

Slideshow with Background Music and Audio Captions

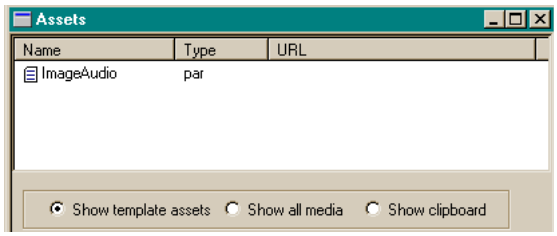
The second template in the Slideshow series is essentially the same as the first, with one important difference: it allows you to build up the presentation based on a collection of complex structure groups that are saved as GR*i*NS *Template Assets*.



Look at the empty view of the template at left. It has an appearance that is familiar from the first template, but the commentary on the container labelled *ImagesAndAudioSequence* says to place an ImageAudio object from the Asset View in the presentation.

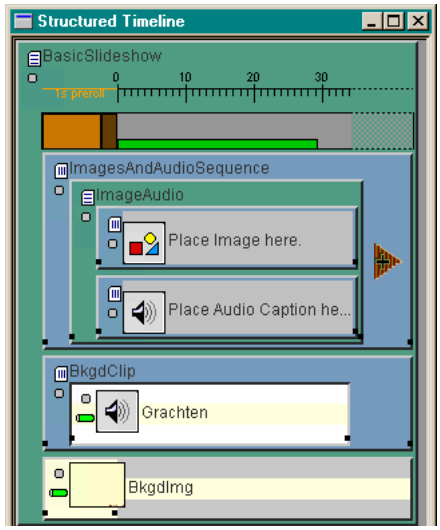
The GR*i*NS assets view is a utility container that allows you to manage assets in the presentation.

A view of the container defined for this template is shown at right. Under the button Show template assets, we see one asset named *ImageAudio*. It has been imported into the template, so it has no external URL. The type of group used to define the asset is a SMIL *par* container.

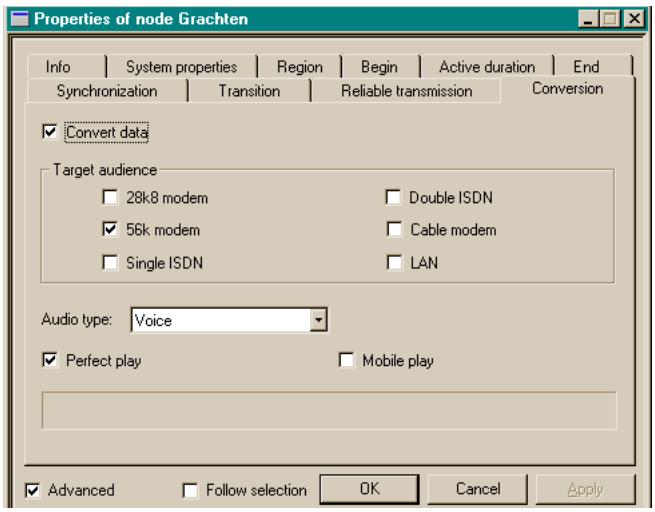


Using standard drag-and-drop techniques, grab the *ImageAudio* asset from the Assets view and drop it onto the *ImagesAndAudioSequence* container in the Structured Timeline view. As is shown on the following page, the ImageAudio asset provides a place to put one (or more) image(s), accompanied by some audio caption. Thus, if you have an audio caption that was made to accompany a single image, you can have one image and one audio. If, other the other hand, you want to have several images accompanied by a single audio caption, you would add more than one image to the Asset container.

In the view shown, we've added the background music and then dragged an ImageAudio asset into the *ImagesAndAudioSequence* container. The template author expects us to add multiple images with captions, thus the DropBox is places outside the asset. Of course, this template (like all GR*i*NS templates!) is intended to be enabling rather than confining. If you wish, you can place new objects anywhere in the template.

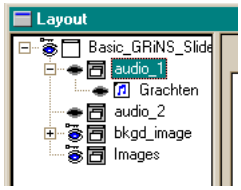


As with other GR*i*NS templates, you have the ability to control various aspects of the presentation via the Property sheets. Most of these properties involve setting values for the presentation, but some involve the conversion of media objects during publishing. In the figure at left, we see how the various conversion parameters to RealMedia can be set by GR*i*NS/RealONE. The values set in these tabs influence the bandwidth needs displayed in the bandwidth strip at the top of the Structured Timeline.

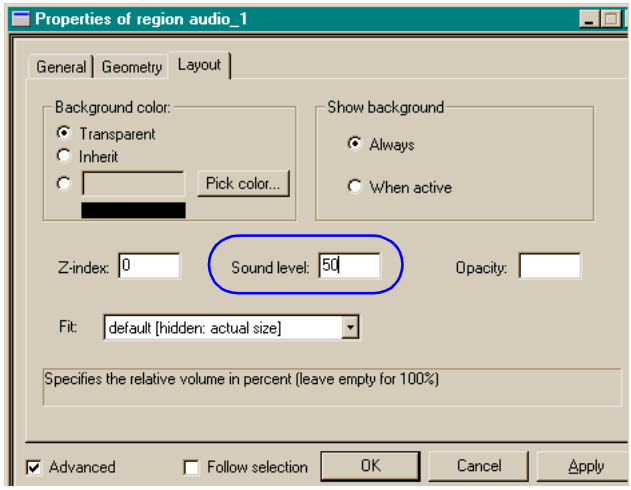


Another of the values than can be set within GR*i*NS involve various rendering attributes. These attributes are not properties of individual nodes or objects, but of the layout Region in which they are presented.

To view these, open the Layout view (Windows->Layout) and look at the Region Selector at the left side of the layout view. Select the region *audio_1*, as shown in the image at right, and then select Properties under the right mouse.



The Region property box has the values shown below.



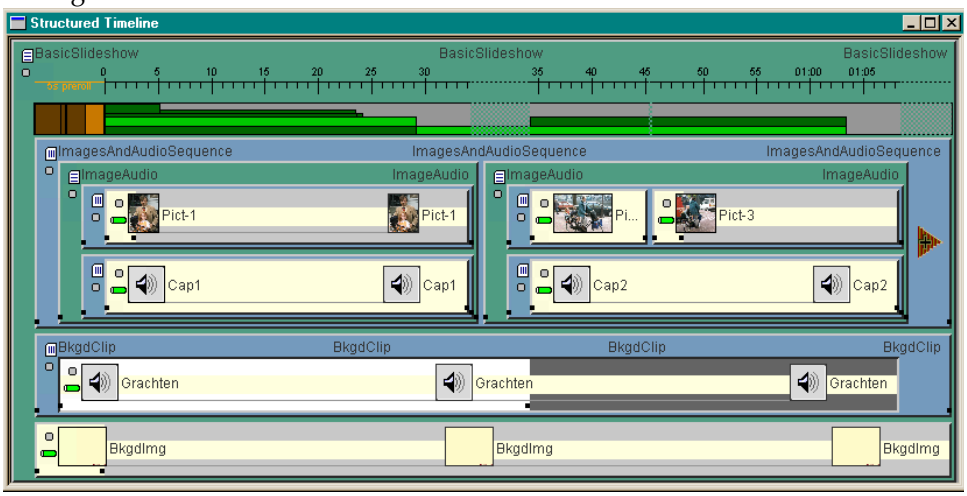
There are several types of SMIL related attributes in this tab, but the one that is of interest to us in this example is the Sound Level attribute. (See the circled area in the image.)

In applications with multiple audio regions, you can set the desired sound level for each region independently. The RealONE Player will produce a final result that mixes all of the audio. (Note that GR*i*NS/Pro allows you to animate these values so that the sound level increases or decreases at will during the presentation.)

Assume now that we've done the following:

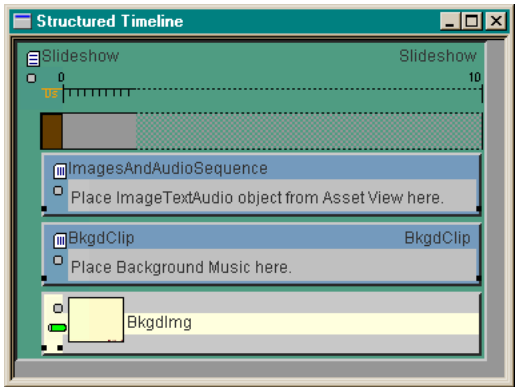
- added a background music node, and set it to repeat twice (via *ActiveDuration*)
- added two ImageAudio assets, filling one with one caption and one image, and the second with one caption and two images.

The figure below shows the Structured Timeline for these actions:



Note that the background of the BkdgClip object is now dark instead of light; this is an indication of the Repeat behavior of this object. Note also the time discontinuities in the timeline — these allow us to add structure annotation in a temporal view.

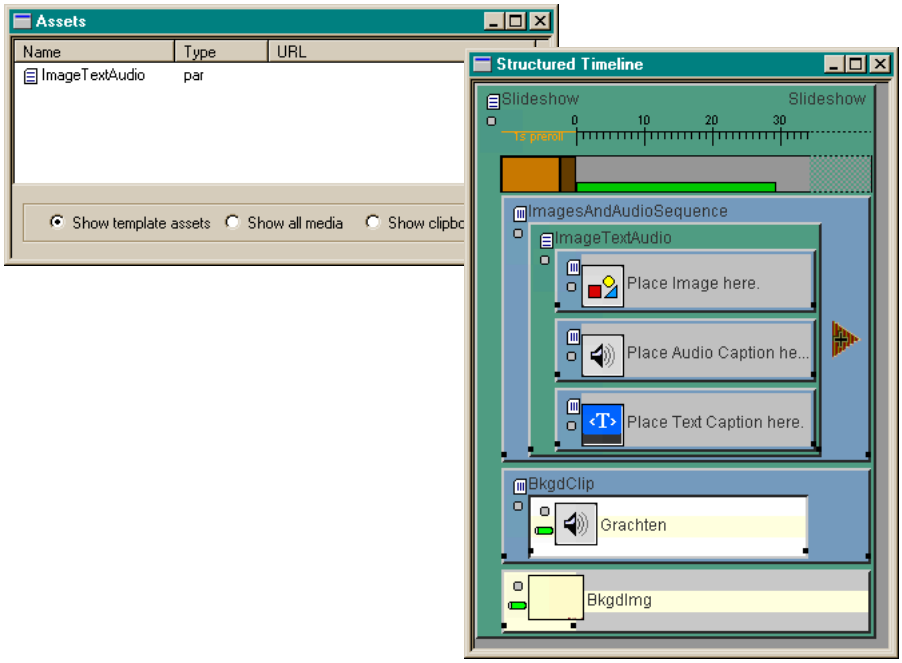
Slideshow with Background Music, plus Audio and Text Captions



The third of the Slideshow templates allows the addition of text captions into the presentation. The basic template is the same as we've seen in the second template, but the object in the Asset view now contains a more complex media asset: a combination of containers for images, audio captions and text captions. (You can potentially use several types

of text for captions, but using a RealText object give you extra control of text placement timing.

The image at left below is the Asset view (as we've seen before); the image at right shows the placement of the object into the template. You can use GR*i*NS to link into your favorite object editors. The Edit->Edit Content command will activate the content editor associated with the file type in your presentation.



GR*i*NS/RealONE Video Templates

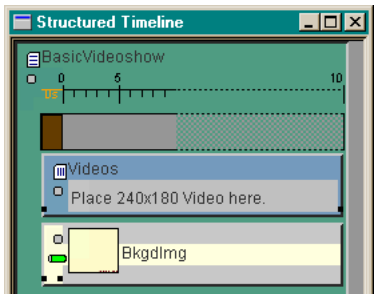
The GR*i*NS/RealONE Video templates will help you create video-based presentations with a minimum of effort.

There are two Video templates in the GR*i*NS/RealONE Template set:

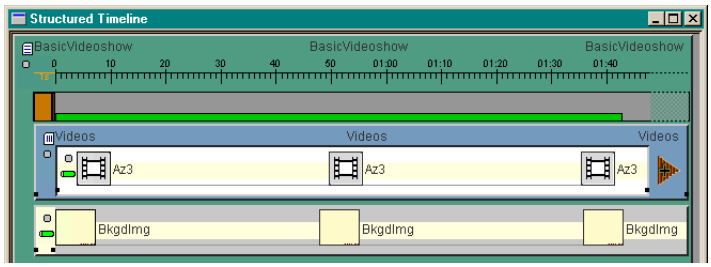
- Video Show (240x180)
- Video Show (320x240)

We suggest that you read through the descriptions of all of the Template to get a better idea of how GR*i*NS/RealONE works.

Video Show (240x180)



When many people think of streaming media, they think of small, postage-stamp sized video presentations. Luckily, with improved end-to-end bitrates and improved codecs, the size of the video on the Web doesn't need to be that small. But more importantly, if you want to partition a story from a set of video fragments — or sections of video combined with images and links — SMIL and RealONE (and GR*i*NS!) can help you add value to your video media.

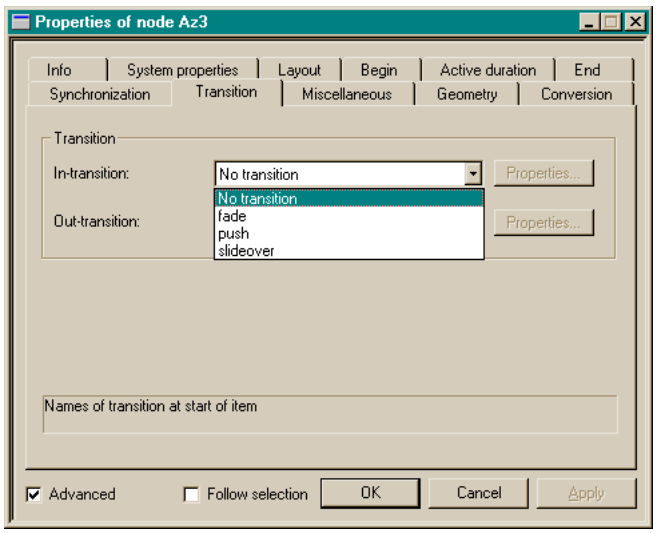


In order to get started with video, the Video 240x180 template gives you an easy framework for publishing your work. Just drop a video object into the Videos container, and hit preview, and you'll see a single video presentation. In the image above, we see the effect of adding a single video of just

over 100 seconds in length. The preview bitrate was set to 112K, and the encoding of the video to 28.8K.

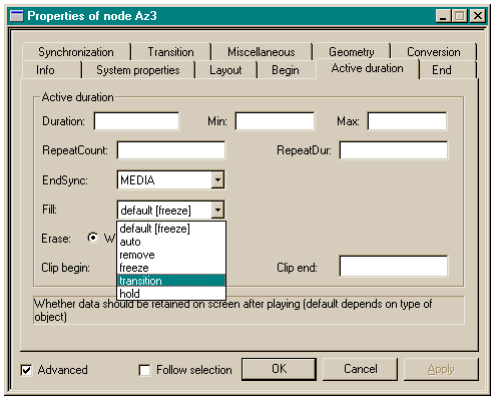
Having a video start and stop abruptly does not always give a very positive user experience. You can improve this by adding input and out transitions. There are basically two ways to add such transitions: you can alter the source data by using video editing software (such as Adobe's Premier), or you can use GR*i*NS to insert transitions in the SMIL presentation instead of the data. The main advantages of using GR*i*NS are:

- you don't have to alter the source data. If you use the same video clip several times, you can use different in-/out transitions with every instance.
- you don't violate the copyright by changing the source video.



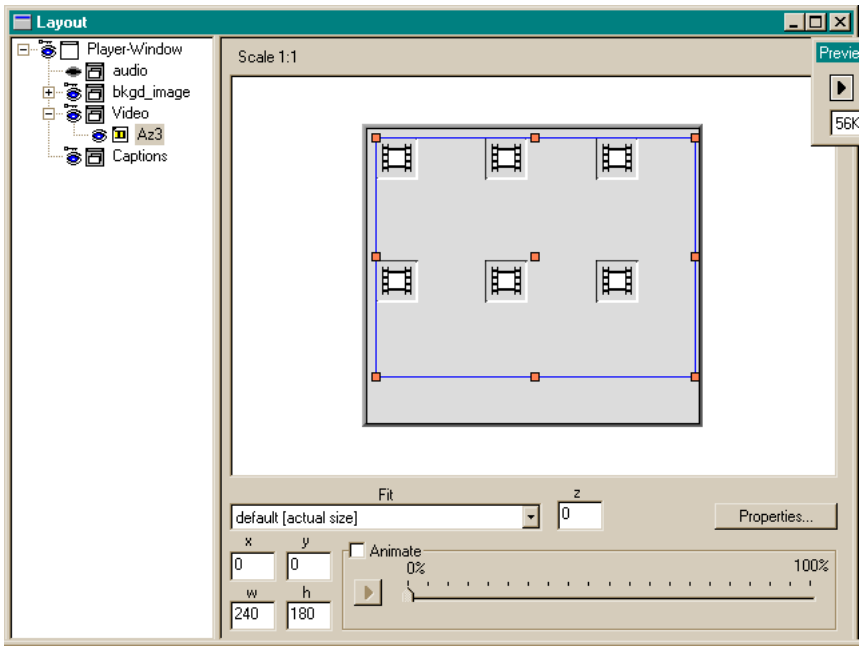
In order to set a transition, use the Properties sheet and select the Transitions tab, as shown. The *Video* template defines three transitions. Each has default properties (such as direction and duration), which you can change by selecting the (sub-)properties tab for each transition type.

Selecting an *input transition* is usually a one-step process. If you specify an output transition, however, you may also have to specify (in the ActiveDuration tab) that the object you want to transition out stays on the screen long enough for the transition to be visible. SMIL gives you two choices: you can set the fill attribute to either *transition* or *hold*.



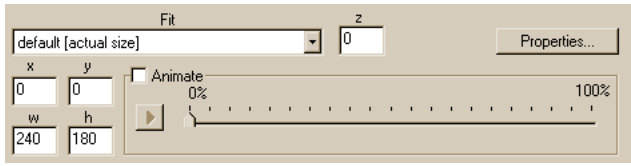
In order to set the value of the fill attribute to *transition*, open the ActiveDuration tab and pull-down the selection menu for fill type. Then select *transition*. You can also select an explicit duration for the video (which will override the intrinsic duration of the video itself. You can also select to show only part of a video by specifying a clip begin/clip end time (but doing this without using a Real server will sometimes cause problems.

Another option to add life to your video presentation is to animate its position. You specify this via the GR*i*NS Layout window, as shown below.



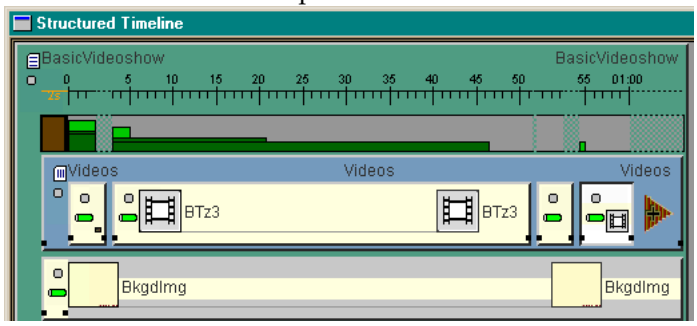
The layout window contains several sub-windows. At top-left is the Region selection window. This window allows you to select a top-level layout window, a region within that window or a media object in a region. When a key appears above the 'eye' icon, it means that region or object will be displayed in the main Layout view. The object being manipulated is indicated by the grey-back selection.

You can specify how the video object fits in the window by trying various values for the Fit attribute. This will allow the media to be shown at full size, or scaled to meet the window size. You can also enable animation and move the location of



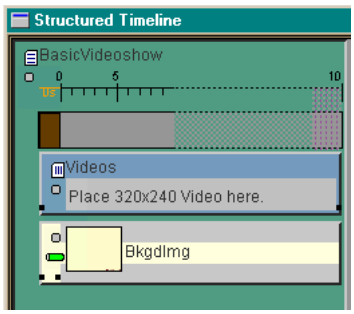
the video within the region over time. When you animate, you define what happens during the video's active duration, but not during its fill behavior.

In order to get familiar with the features for video, add two videos to a presentation and then test various settings for placement and alignment. A two-video structure view is shown below. Note that each video is preceded by a title slide, which is shown in the same region. Note also that since the last (and very short) video is highlighted, we see the impact of this video on the bandwidth in a highlight color in the bandwidth strip.



Video Show (320x240)

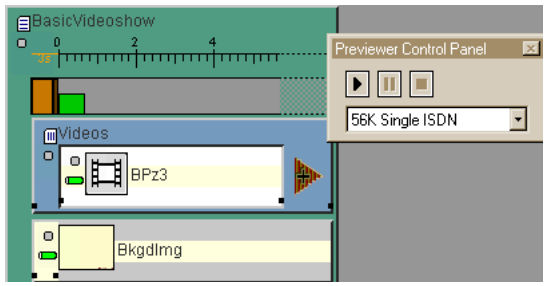
The template for the 320x240 video show is identical to that of the 240x180 show, except that the video is somewhat larger — and thus requires a larger background image.



In point of fact, depending on the setting of the fit attribute in the layout view (shown above), this template will take on a video of any size and either scale it to fit in the window, or clip off the parts that don't fit. Still, it is better for everyone concerned when the client doesn't have to scale the final video product.

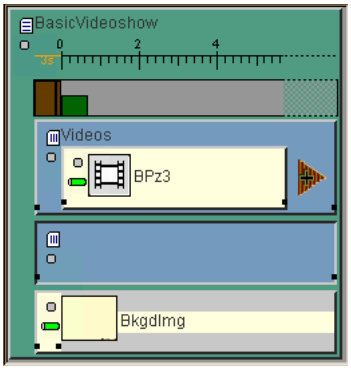
Since we already have shown how you can manipulate most of the basic template options, we'll use our discussion of this template to show how the basic presentation can be expanded.

First, take a piece of video and place it in the *Videos* container. Doing so yields the view at right.



Now suppose we'd like to add some captions to this presentation. There are several ways to do this. We'll look at two of these ways: adding a general captions container, and adding individual caption to each video object.

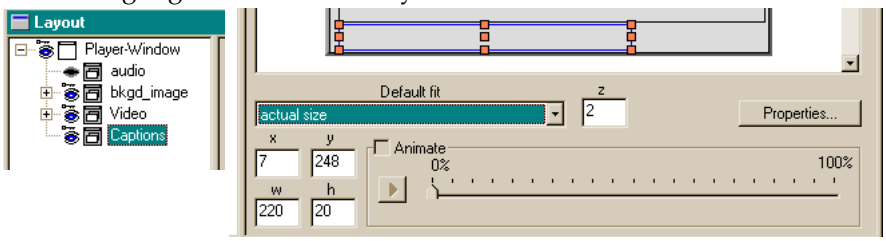
To create a separate captions container, go to the Containers toolbar. This should be in the toolbars area of your window, but if it isn't, you can activate it via the View->Toolbars->Containers command. (The toolbar is shown at left.) Select the blue container, with the tool tip *Sequential Group*.



Drag the *Sequential Group* icon onto the Structured Timeline, placing it below the *Videos* container. This will result in the view at left.

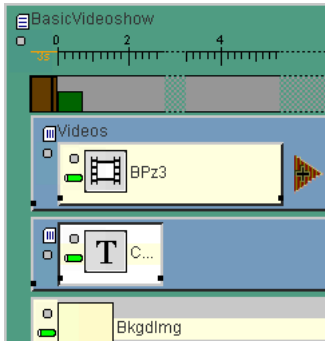
What we have done is to define a new container inside of the same parent par container as the *Videos* group. Since we want to insert a caption, we also could have just added a media object here, but planning ahead and using groups to hold media means that you can create multiple objects in a uniform manner.

Before adding any assets, we should also indicate where we want the caption to be played. Open the Layout view. You will see that the Template author was thinking ahead: there is a Captions region already defined. Select the region and note the highlighted area on the layout area.

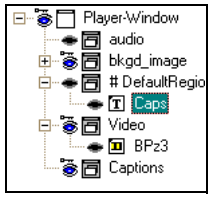


Now, place a text media caption into the presentation by dragging the text file holding the caption into the container we've just made. The resulting view is shown at right. What this view is telling us is:

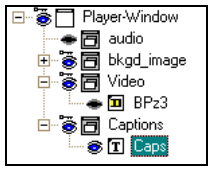
- we have a video container, a captions container and a background image that will run from a common timebase.
- The video is about 5 sec. long, the image is about 3 sec., and the background image is 0 seconds (but with a fill="freeze".)



before:

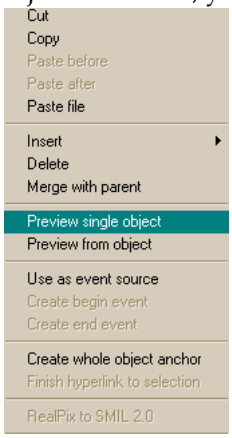


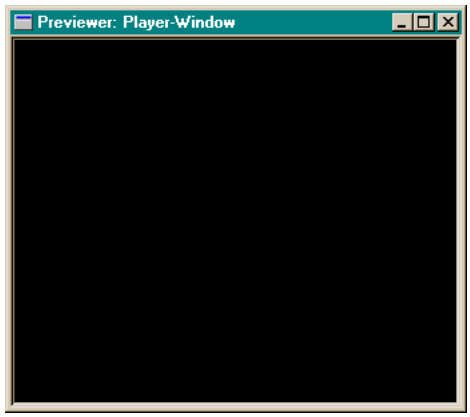
after:



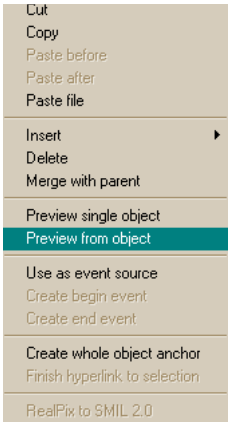
Once you've added the captions object, look at the Layout view again. In the left selection area, you'll see that the *Caps.txt* object was placed into a temporary default region (named *#DefaultRegion*). Since this is not where we want the object to be, we can grab the *Caps.txt* object and drag it to the *Captions* region. The before and after effects of making this move are shown at left. (Note that if the template author was really thinking ahead, the template would have said that *Captions* was the default regions for new text items!)

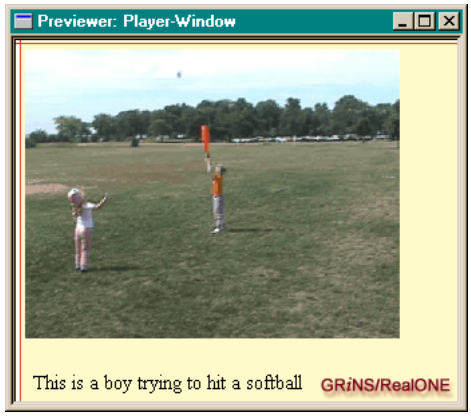
At this point, we've added a caption into the presentation. To make sure we have the correct data object, we should preview the show. If you do this via the Preview Single Object command, you get the view below:





We see a lot of black, but this is not an error. We are actually seeing a caption with a default text color of black against the default black background. (This is the behavior in GR*i*NS previewer; early versions of RealONE always inserted a white border around the text.)





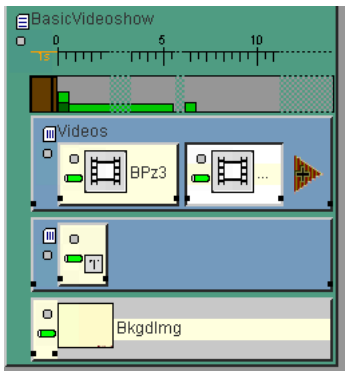
In order to see what the caption looks like in context, try the Preview From Object command in the Editor. He we see that the black caption is correctly placed against the presentation background.

(The entire caption file we used had the text:

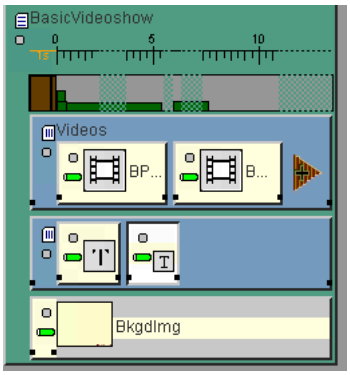
```
This is a boy trying to hit a softball thrown by his sister.
```

The text has been clipped because of the value of the Fit attribute for the region.)

Now, let's add a second video to the presentation. Drop the video after the first. This will give you the Structured Timeline view shown at right. The duration of the caption is unchanged, but the duration of the overall presentation has doubled.

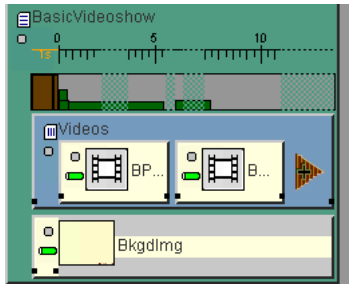


You should now add a second caption file:

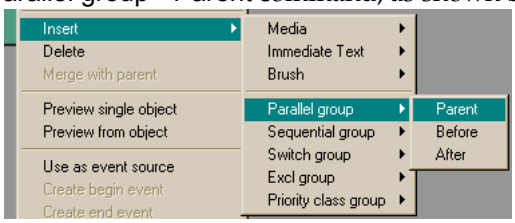


The second caption will start when the first one ends, rather than when the second video starts. Adjusting the start times of each of the captions to match that of the corresponding video is not hard: you use the drag handles at the bottom of each text object. However, if you later replace some of the videos, you will need to adjust the start times of the captions manually.

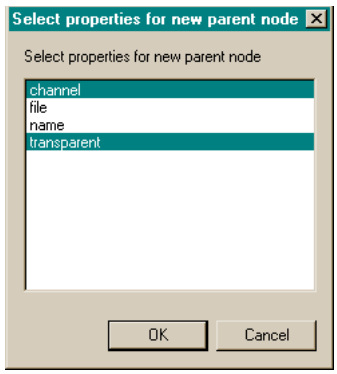
GR*i*NS can make the structure of your presentation more media-independent. Start by getting rid of all of the captions items by deleting the container in the Structured Timeline. This yields the view at right. What we will now do is to wrap the first video object within a parallel group, and add a caption object to that group.



Start by selecting the first video object, and then select the Insert->Parallel group->Parent command, as shown below.

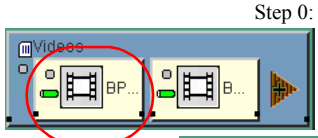


This creates a wrapper group around the one selected. When you create a new group in this manner, you get a special GR*i*NS dialog that lets you select which attributes of the child you would like to send to the parent. (See right.) You should un-select all of the options, since we don't need to actually set any parent values.

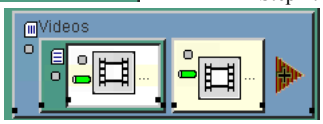


The set of images on the next page show the effect on the presentation of adding the wrapper, then adding a *Seq* group and then adding a caption.

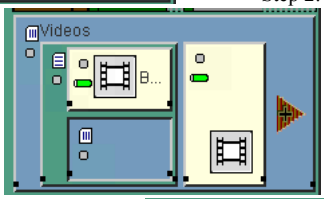
Step 0: Select the first video object.



Step 1: Add a *Parallel Parent* wrapper around the video



Step 2: Drag a SEQ group from the Container toolbar into the new PAR wrapper

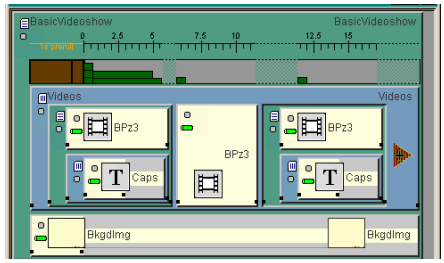


Step 3: Drag a captions object into the new SEQ group

(You could have dragged a captions object directly into the PAR group, but using a SEQ can give you some flexibility if all of your captions are not in one file.)

Once you've created the captions group and verified that the region association is correct, you can copy the entire PAR wrapper element and then paste it within the videos component using Paste Special->Within.

You now have the ability to add captions (or any other type of media) based on the composite structure containers you've created. The view at right shows a mix of individual videos and captioned components.



GR*i*NS/RealONE Adaptive Templates

The GR*i*NS/RealONE Adaptive templates will help you create presentations that can scale to the bandwidth available across a network connection. Unlike encoding-centric solutions, such as Real's SureStream technology, the GR*i*NS/RealONE templates provide a solution based on the SMIL switch construct. The advantage of using the switch mechanism is that you can make non-trivial substitutions of content: the issue is not always a different encoding of, say, video — sometimes you may want to use a whole different encoding for each audience.

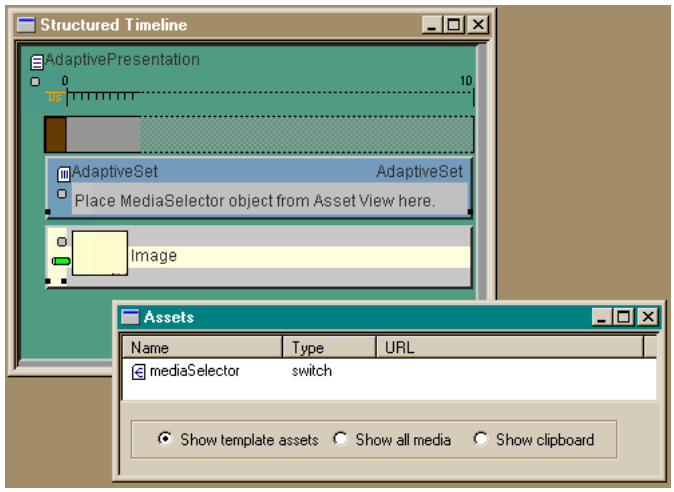
There is one Adaptive template in the GR*i*NS/RealONE Template set:

- Adaptive Video Presentation

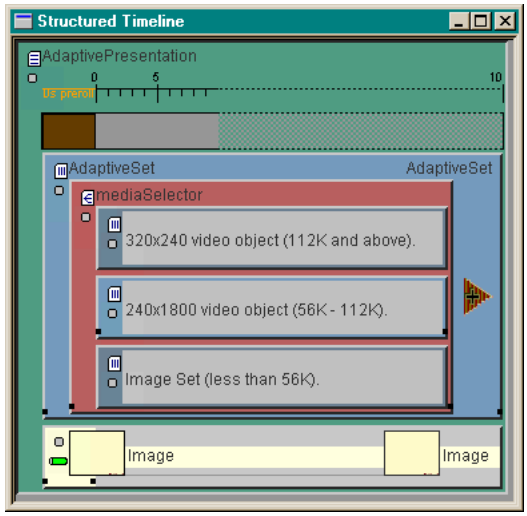
Only one Template has been included in the base set, because the techniques you learn by using this Template can be applied to many other presentations as well.

Adaptive Video Presentation

Select the Adaptive Video template from the template set. You will see the following view in the editor:



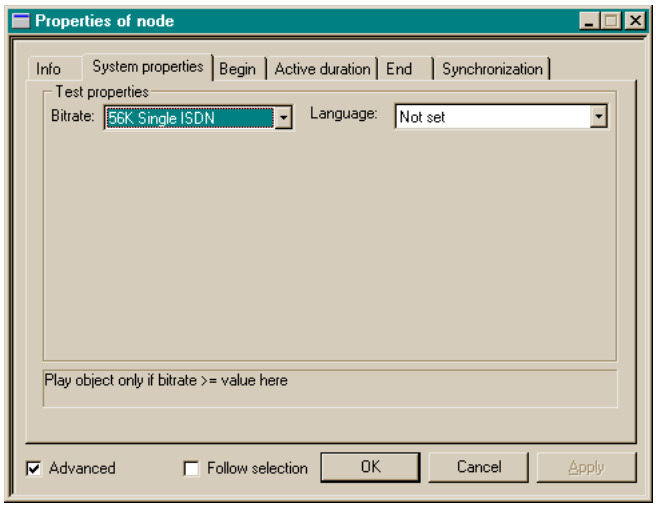
This view is similar to the more complex slideshow templates. You drag a complex structure container (defined by the template author) into the Structured Timeline. The view you now see is shown at right.



The *AdaptiveSet* object contains a red *mediaSelector* object, which contains:

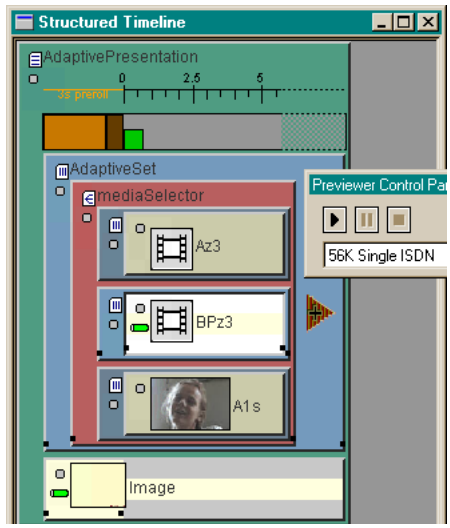
- a container for 320x240 video
- a container for 240x180 video
- a container for a set of images

The red container is the GR*i*NS representation for a SMIL *switch*.



A *switch* is a selector container. Each of the objects inside the switch are evaluated by the RealONE Player in a top to bottom order. Each object can contain a *test attribute* setting. These settings are defined in the System Properties tab of the object's property sheet. An example of a system properties tab is shown. GR*i*NS/RealONE gives you access to the Bitrate and Language properties. GR*i*NS/Pro allows you to set all SMIL system test attributes.

Of of the powerful features of a switch is that only one of the candidate items are selected for presentation, based on the runtime characteristics of the environment. This mean, however, that the overall presentation timeline will change, depending on which element gets selected. The *structure* of the presentation doesn't change, just the *timeline*.

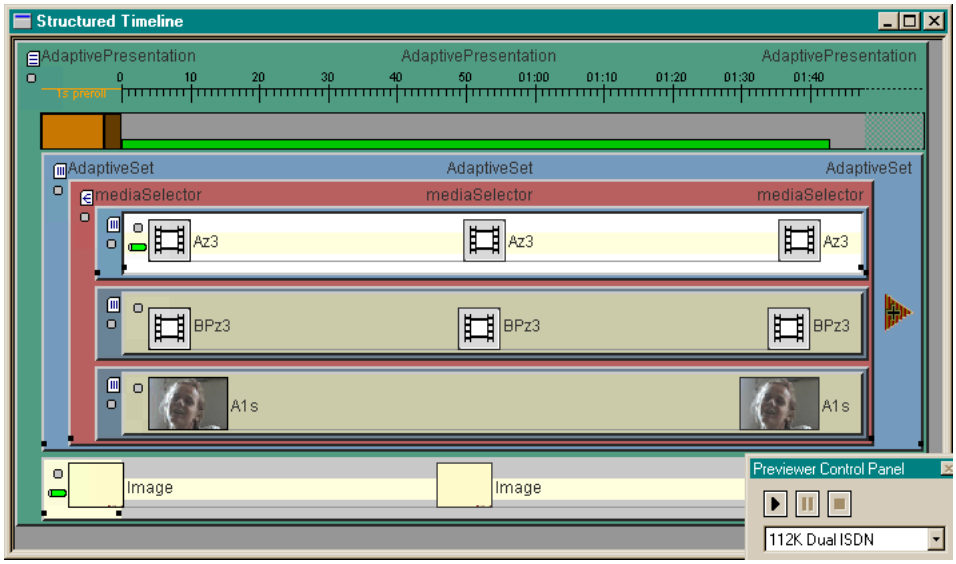


In the example at left, the top object is defined to play if the available bitrate is 112K or more. The middle object will play if the bitrate is between 56K Modem and 122K. The lower object will play if no other object has been selected. (It has this behavior because it has no bitrate set in its property tab: if none of the object above have been selected, then this one is activated by default.)

You should open the property tabs for each of the objects and look at the System Properties settings. Keep in mind that only the settings for the first-level children of the *switch* are used — the settings on the media objects don't matter.

The use of the *switch* node is a very powerful construct in GR*i*NS and SMIL. In the example above, we've done a fairly straightforward substitution of media objects, but in more complex presentations, you can substitute entire sub-presentations.

The switch element can be nested — that is, a switch can have a *switch* as a child. You should be careful to remember that it is the direct children of a *switch* that have their system properties evaluated, and not the *switch* itself.



The image above shows the same Structured Timeline view as on the previous page, but with 112K selected in the previewer control panel. GR*i*NS greys out the nodes that are not active in the switch, so that you get a better view of the presentation.

In some presentations, you may not want to substitute one element for another, but instead may only want to conditionally add a single object. You can do this without a switch by simply setting the system properties of any object. The player will then check to see if the requested condition is true — if so, it plays the object, and if not, it won't.

The GR*i*NS/RealONE editor was designed to allow manipulation of basic system test attributes. The GR*i*NS/Pro editor provides additional facilities to control more aspects of the presentation, including conditional captions and other accessibility features.

GR*i*NS/RealONE Empty Templates

The GR*i*NS/RealONE Empty templates will help you create presentations that can serve a wide range of needs. The Empty template contains all of the structure and background information you need to get started making custom presentations. You can then add structure groups, switches, animations and media content at will.

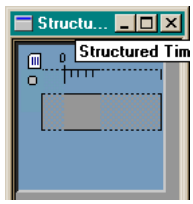
There is one Empty template in the GR*i*NS/RealONE Template set:

- Empty RealONE Presentation

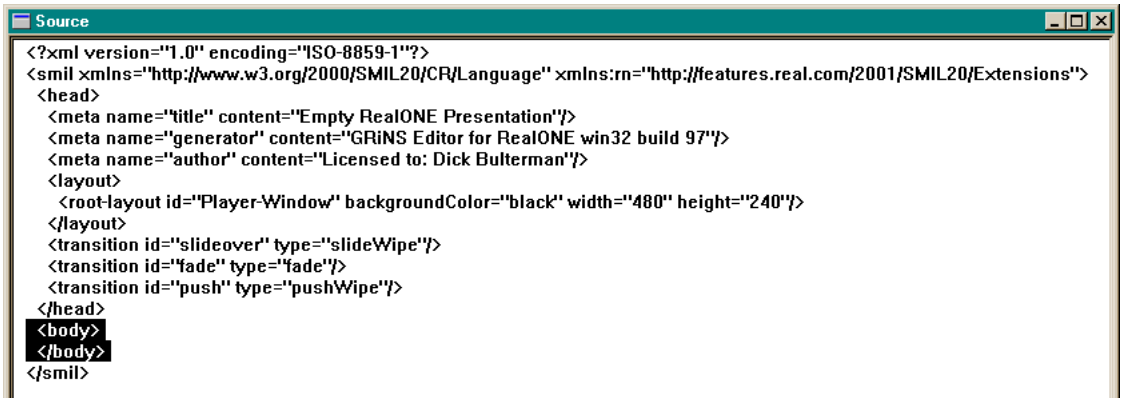
Only one Template has been included in the base set, because the techniques you learn by using this Template can be applied to many other presentations as well.

Empty RealONE Presentation

Open the Empty template from the template set. You see a Structured Timeline that does not contain any internal objects, just a general timeline and an empty bandwidth usage strip. While the empty template seems fully empty, it actually has some useful content.



You can see this content by opening the Source window.

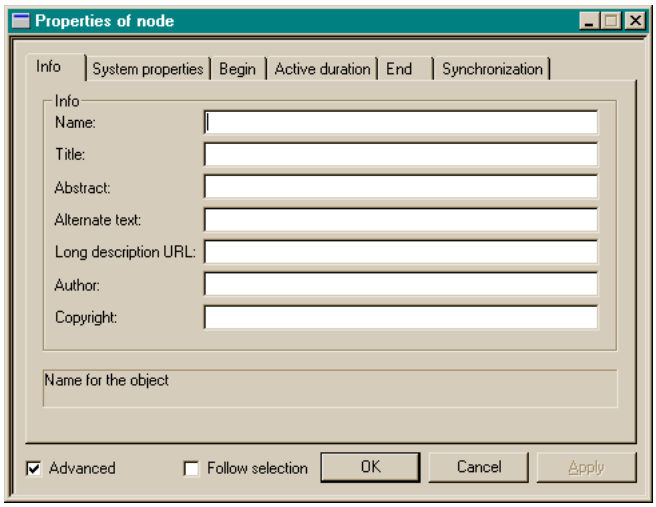


The empty document contains most of the XML boilerplate required to create a correct SMIL-2 application. It also hold some of the meta data that can be used to catalogue the presentation.

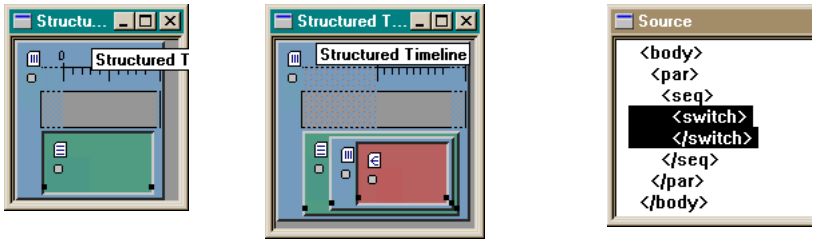
You can create the actual presentation by placing structure groups and media objects into the Structured Timeline. The groups can be entered via the Containers toolbar, shown at right. The color of the groups match their GR*i*NS visual equivalents:

- the green container is a SMIL par;
- the blue container is a SMIL seq;
- the red container is a SMIL switch;
- the dark purple is a SMIL excl; and
- the light purple is a SMIL priority class.

Whenever you add a container object to a presentation, you get the property sheet for that node. Use this to fill in properties such as the node name, or other useful information. An empty generic property sheet is shown below.



The figure below shows the effects of adding several container nodes to the Structured Timeline:



(a) (b) (c)

- (a) shows the impact of adding a PAR container;
- (b) shows the view after adding a nested SEQ and Switch; and
- (c) shows the SMIL source view for (b).

You can create any RealONE application using the empty template and the various GR*i*NS editing features. The advantages of using GR*i*NS over a text editor (which you could also use to create any RealONE application) are:

- incremental previewing via the GR*i*NS/RealONE preview engine;
- syntax and other error checking
- bandwidth monitoring and evaluation
- the ability to import and convert legacy RealPix
- the ability to open and extend any existing SMIL content and publish for the RealONE player
- the ability to visualize the structure and the temporal aspects of the presentation as it develops.

We welcome your comments on GR*i*NS/RealONE and the GR*i*NS template set. Please contact:

grins-support@oratrix.com

GR*i*NS Edition Comparison Information

GR*i*NS Packaging Information

The GR*i*NS Editor for RealONE is one member of a family of editing products for the SMIL-2.0 standard. This section provides a feature differentiation and compatibility matrix for the GR*i*NS/RealONE and GR*i*NS/Pro editions.

The features of each product are subject to change without notice.

GRiNS Feature Table	**GR*i*NS / Pro**	**GR*i*NS / RealONE**
Standard Features		
Structured Timeline Editing	☑	☑
Layout Editor	☑	☑
Animation Editor	☑	☑
Transition Selector/Editor	☑	☑
Source View/Editor (see below)	☑	☑
Full GUI-based editing	☑	☑
Design from Templates	☑	☑
Create custom presentations	☑	☑
Import existing presentation	☑	☑
Publish to popular players	☑	☑
Create par/seq/excl/switch groups	☑	☑
Integrate audio/video/text/images	☑	☑
Drag&Drop editing	☑	☑
Infinite Undo Stack	☑	☑
Create (timed) links	☑	☑
Insert events	☑	☑
Preview Presentation	☑	☑
Preview Individual Node/Group	☑	☑
Preview from any Node/Group	☑	☑
Model Network performance	☑	☑
Preview for different bitrates	☑	☑

GRiNS Feature Table	GR*i*NS / Pro	GR*i*NS / RealONE
New Presentations:		
From Templates	☑	☑
Custom Document	☑	☑
Create New Template	☑	--
Import:		
From Existing SMIL-2	☑	☑
From Existing SMIL-1	☑	--
From Existing RP-6/7/8	☑	☑
Publish:		
To SMIL-2 (Generic SMIL-2)	☑	--
To RealONE	☑	☑
To SMIL-1	☑	--
To RP-6/7/8	☑	--
To HTML+Time	☑	--
Publish & Prune to Profile		
Based on Bitrate	☑	--
Based on Language	☑	--
Based on Captions	☑	--
Based on Any Test Attribute	☑	--
Convert		
RealPix to SMIL 2.0	☑	☑
SMIL 2.0 to RealPix	☑	--
Media Objects to RM	☑	☑
Media Objects to WMP	☑	--
Content		
Link to Installed 3rd Party Editors	☑	☑
Embedded image trimmer	☑	--
TWAIN object import	☑	--
Toolbars		
Standard Configuration	☑	☑
Customize Configuration	☑	--

GRiNS Feature Table	GR*i*NS / Pro	GR*i*NS / RealONE
UI Features		
Show Asset Thumbnails	☑	☑
Customize Asset Thumbnails	☑	--
Show Playable in Switch	☑	☑
Customize 'Show Playable'	☑	--
Structured Timeline View		
Standard Timeline View	☑	☑
Customized Timeline View	☑	--
Optimized Structured Timeline	☑	--
Source View		
View Source	☑	☑
Edit Source	☑	--
Animation Editor		
Child Animations on Nodes	☑	☑
Animations on Regions	☑	☑
Full Animation Elements	☑	--
Exclusive Nodes		
Create EXCL	☑	☑
Create PRIORITY CLASS	☑	☑
Transitions		
Use Standard GR*i*NS set	☑	☑
Edit Standard GR*i*NS set	☑	☑
Define new transitions	☑	--

GRiNS Feature Table	GR*i*NS / Pro	GR*i*NS / RealONE
Switch Support		
Specify Bitrate	☑	☑
Specify Language	☑	☑
Specify Captions	☑	--
Specify Audio Desc	☑	--
Specify Subtitle/Overdub	☑	--
Specify Screen/Depth	☑	--
Specify CPU / OS	☑	--
Specify Player/Req/Comp	☑	--
Custom Tests		
Play Custom Tests	☑	--
Define Custom Tests	☑	--
Tools		
Summary Bandwidth Usage	☑	☑
Detailed Bandwidth Strip	☑	☑
Align/Distribute Nodes in Layout	☑	--
Layout Support		
Define root-layout via UI	☑	☑
Edit root-layout numerically	☑	--
Define mulltiple topLayout via UI	☑	--
Edit topLayout numerically	☑	--
Define regions via UI	☑	☑
Edit regions numerically	☑	--
Define reg hierarchy via UI	☑	☑
Edit reg hier. numerically	☑	--
Define sub-regions via UI	☑	☑
Edit sub-region numerically	☑	--
Select regPoints via UI	☑	☑
Define regPoint numerically	☑	--

GRiNS Feature Table	GR*i*NS / Pro	GR*i*NS / RealONE
Timing Specification		
Specify Duration	☑	☑
Specify EndSync	☑	☑
Specify Fill	☑	☑
Specify repeatCount	☑	☑
Specify repeatDuration	☑	☑
Specify Min	☑	--
Specify Max	☑	--
Begin/End Specification		
Delay / NodeID / Indefinite	☑	☑
AccessKey / WallClock	☑	--
Event	☑	☑
X-Path Based Events	☑	☑
Restart Semantics	☑	--
Media		
via Drag&Drop	☑	☑
via local name	☑	☑
via URL	☑	☑
ClipBegin / ClipEnd	☑	☑
Event Sensitivity	☑	☑
Erase	☑	--
Sychnronization Behavior		
Specify Sync Default	☑	--
Specify Sync Behavior	☑	--
Specify Sync Master	--	--
Productivity Enhancements		
Specify Default Layouts for Groups	☑	☑
Specify Default Timing for Groups	☑	--
Specify Default Fill for Groups	☑	--
Specify Default Transitions	☑	--
Merge Nodes with Parents	☑	--

GRiNS Feature Table	GR*i*NS / Pro	GR*i*NS / RealONE
Linking		
Create whole-node anchors	☑	☑
Create partial-node anchors (X/Y)	☑	--
Create timed anchors	☑	--
Create external anchors	☑	☑
Create internal anchors	☑	--
Create Link to RealONE Browser Window	☑	☑
Create Link to RealONE Context Window	☑	☑
Events		
Create short-cut events	☑	☑
Create/edit begin events	☑	☑
Create/edit end events	☑	☑

NOTE:

1. The available versions of the GR*i*NS/RealONE products are updated regularly. Please consult the following Web URL for the most recent version of the GR*i*NS product matrix at:
   http://www.oratrix.com/

GR*i*NS/RealONE Quick Reference Information

SMIL Compliance Information

The GR*i*NS/RealONE supports the entire SMIL 2.0 specification, although not all features are previewed during editing. Documents that make use of these constructs are parsed correctly, but the features are ignored during rendering.

Supported Media Table

The following chart gives a listing of the media types supported by the Win32 versions of GR*i*NS Editor for RealONE:

MIME type	Extensions	Windows 98SE/2K/XP
audio/basic	au	yes
audio/x-aiff	aiff, aifc, aif	yes
audio/x-wav	wav	yes
image/jpeg	jpeg, jpg	yes
image/png	png	yes (2)
image/tiff	tiff, tif	yes
image/x-portable-anymap	pnm	no
image/x-portable-bitmap	pbm	no
image/x-portable-graymap	pgm	no
image/x-portable-pixmap	ppm	no
image/x-rgb	rgb	yes
image/x-xbitmap	xbm	no

MIME type	Extensions	Windows 98SE/2K/XP
	bmp	yes
	ras	yes
	tga	yes
video/mpeg	mpeg, mpg	yes
video/quicktime	qt	yes
video/x-msvideo	avi	yes
video/x-sgi-movie	mov	no
text/html (5)	html, htm	no
text/plain	txt	yes

Notes

1. Uncompressed WAV only.
2. Support seems to be somewhat buggy.
3. Not all encodings supported.
4. Linux information provided for planning purposes only.
5. HTML data is rendered by the GRiNS Preview Player, but not converted to RealText.

Each of these formats is converted to the appropriate RealSystem datatype. For highest quality rending of a final presentation, we recommend converting some datatypes to RealMedia before inserting them in a presentation, if possible.

RealONE Media Conversion

The following chart describes the levels of support provided in the GR*i*NS/RealONE version for the listed RealMedia data types used in the RealNetworks RealONE Player:

RealMedia	Extension	Importable	Auto-Generated
RealAudio	ra, rm	yes	yes
RealVideo	rm	yes	yes
RealText	rt	yes	no(1)
RealPix	rp	yes	no

Notes

1. GR*i*NS/RealONE provides support for the automatic generation of simple RealText documents from immediate strings in the Editor, but it does not at present provide full RealText editing facilities. This is expected in a future release.
2. GR*i*NS/RealONE can import and convert most existing RealPix files. Limited support is available for creating RealPix from SMIL-2.0 constructs for users wishing backward compatibility.

References and Links

Please see the Links section of the GR*i*NS/RealONE Web site (*www.oratrix.com/GRiNS*).

Where Next?

The GR*i*NS Editor for RealONE *Template Guide* provides you with a general overview of GR*i*NS and a particular introduction to the templates available with the base distribution of GR*i*NS/RealONE.

You should also consult the RealONE IQ Production Guide, available from RealNetworks, Inc. This guide provides an overview of the facilities supported by the RealONE player. It also provides a comprehensive introduction to the concepts of streaming media and details on the SMIL-2.0 language.

The GR*i*NS Web site (*www.oratrix.com/*) provides a set of *release notes* and *issues reports* for each version of the GR*i*NS Editor and Player releases. This site also contains a publicly-available version of the GR*i*NS Frequently-Asked Questions list and errata to this and other GR*i*NS publications. If you purchased GR*i*NS, you will have had the opportunity to enroll for our automatic notification service for the version(s) of GR*i*NS you acquired.

We recommend that you use the Check for GRiNS update button in the Help menu. This version will poll the GR*i*NS Web site to see if an updated version of GR*i*NS exists.

We welcome your comments, criticisms, compliments and suggestions. You can reach us at: *grins-documentation@oratrix.com*.

We hope you find working with GR*i*NS a productive experience!