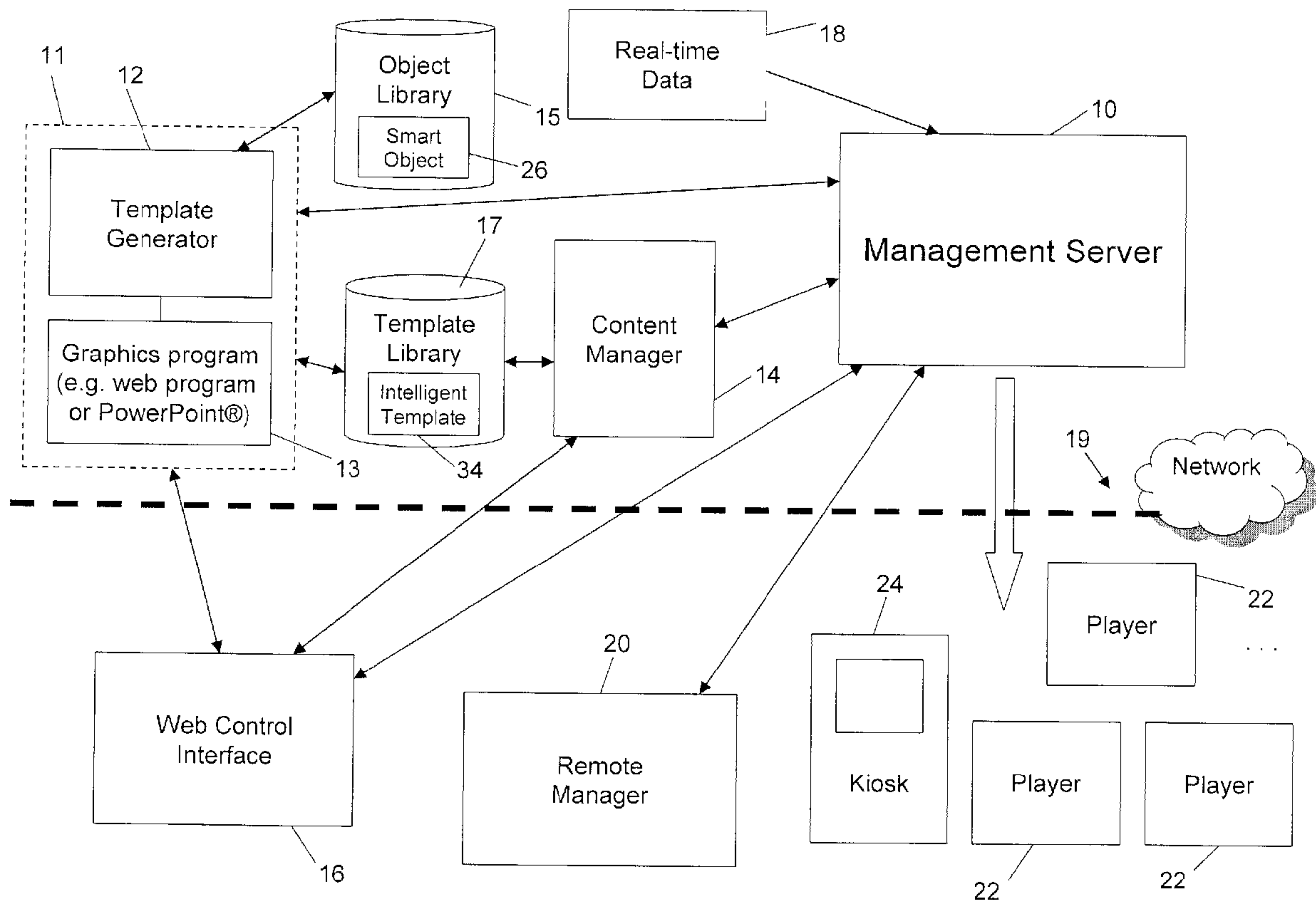




(22) **Date de dépôt/Filing Date:** 2008/04/11
 (41) **Mise à la disp. pub./Open to Public Insp.:** 2008/10/13
 (45) **Date de délivrance/Issue Date:** 2017/08/22
 (30) **Priorité/Priority:** 2007/04/13 (US60/911,572)

(51) **Cl.Int./Int.Cl. G06F 17/00** (2006.01),
G06F 3/14 (2006.01)
 (72) **Inventeur/Inventor:**
WILKINS, DAVID, CA
 (73) **Propriétaire/Owner:**
X20 MEDIA INC., CA
 (74) **Agent:** BLAKE, CASSELS & GRAYDON LLP

(54) **Titre : METHODE ASSURANT L'AUTOMATISATION DES APPLICATIONS DE SIGNALISATION NUMERIQUE AU MOYEN D'OBJETS A AUTOCONFIGURATION INTELLIGENTS ET DE FORMES DE REFERENCE INTELLIGENTES**
 (54) **Title: A METHOD FOR AUTOMATING DIGITAL SIGNAGE APPLICATIONS USING INTELLIGENT SELF-CONFIGURING OBJECTS AND SMART TEMPLATES**



(57) **Abrégé/Abstract:**

A digital signage content management system is provided that uses existing interfaces such as web interfaces and turns existing commercially available graphics programs such as web based tools or locally run programs such as Microsoft PowerPoint® into a

(57) Abrégé(suite)/Abstract(continued):

digital signage platform to facilitate developing and managing digital signage applications through the creation of smart objects and intelligent templates that are easy to create and easy to modify to suit different applications. This enables digital signage content to be professionally created without requiring custom programming for each and every stream of new and/or changing content. The smart objects and intelligent templates can also be used to provide content with changing elements in real-time.

1 **ABSTRACT**

2

3 A digital signage content management system is provided that uses existing interfaces such as
4 web interfaces and turns existing commercially available graphics programs such as web based
5 tools or locally run programs such as Microsoft PowerPoint® into a digital signage platform to
6 facilitate developing and managing digital signage applications through the creation of smart
7 objects and intelligent templates that are easy to create and easy to modify to suit different
8 applications. This enables digital signage content to be professionally created without requiring
9 custom programming for each and every stream of new and/or changing content. The smart
10 objects and intelligent templates can also be used to provide content with changing elements in
11 real-time.

1 **A METHOD FOR AUTOMATING DIGITAL SIGNAGE APPLICATIONS USING**
2 **INTELLIGENT SELF-CONFIGURING OBJECTS AND SMART TEMPLATES**

3
4 **TECHNICAL FIELD**

5 **[0001]** The present invention relates to digital signage applications and has particular utility
6 in automating such digital signage applications.

7 **BACKGROUND**

8 **[0002]** The market for public information displays has evolved considerably over the past
9 years to include full motion video content combined with images and text displayed on high-
10 resolution video graphics screens. It is becoming more and more common to see these kinds of
11 digital signage systems in shopping centers, hotels, university campuses, and corporate lobbies.
12 For the most part, the process of creating and managing content to be displayed on these screens
13 has involved using standard graphics and video production tools to produce pre-rendered video
14 clips that are then played back according to a predefined schedule or play list.

15 **[0003]** More recently, tools have become available that allow individual content elements
16 such as graphics, animations, and video to be dynamically composited and rendered into a video
17 stream in real-time, without requiring the need for pre-rendering all content into a single video
18 file. This allows independent elements or layers of content to be changed “on the fly” in response
19 to specific data conditions. In a typical example, a weather display could automatically show the
20 latest temperature and meteorological conditions throughout the day while the latest news
21 headlines scroll in a ticker at the bottom of the screen. As the weather changes or breaking news
22 becomes available, the content on the screen is automatically updated.

23 **[0004]** Whereas conventional digital signage systems have used DVD players or other
24 simple video playback systems to display pre-rendered content, real-time digital signage systems
25 utilize more advanced computer and video hardware and specialized software to dynamically
26 render content elements on demand. These real-time systems have the major advantage of being
27 able to instantly update screen content in response to manual or automatic triggers, unlike the
28 conventional video playback systems which require an entire video clip to be re-rendered every
29 time content needs to be updated. This greatly reduces production times and network bandwidth

1 required to distribute content for playback in multiple locations, resulting in content that is more
2 dynamic and visually appealing to the audience.

3 **[0005]** A major drawback of real-time digital signage tools is that they require a greater
4 amount of development effort to create video and graphics content and integrate this content with
5 real-time data sources. This translates into higher operating costs, making the return on
6 investment for this type of digital signage system less attractive for many applications, despite
7 the clear benefits to the audience in terms of more interesting and engaging content.

8 **[0006]** The process of creating real-time data-driven graphics for digital signage typically
9 requires 4 primary steps: 1) Creation of graphical and video elements by a graphic artist; 2)
10 Development of custom software applications or scripts by a software programmer to link
11 graphical elements to data sources; 3) Distribution of graphical elements and software
12 applications to final play-out locations using either a local or wide-area network, or a manual
13 distribution medium such as CD-ROM; and 4) Monitoring and updating of system elements on
14 an ongoing basis.

15 **[0007]** The currently available tools and systems provide a means for achieving each of these
16 steps, but for the most part require a level of specialized knowledge that the average user must
17 acquire through extensive training and hands-on experimentation.

18 BRIEF DESCRIPTION OF THE DRAWINGS

19 **[0008]** Embodiments of the invention will now be described by way of example only with
20 reference to the appended drawings wherein:

21 **[0009]** Figure 1 is schematic block diagram showing a digital signage content management
22 system.

23 **[0010]** Figure 2 is a schematic block diagram of a smart object.

24 **[0011]** Figure 3 is a schematic block diagram of an intelligent template.

1 [0012] Figure 4 is a screen shot showing a user interface for the template generator and
2 graphics program shown in Figure 1.

3 [0013] Figure 5 is another screen shot of the user interface of Figure 4.

4 [0014] Figure 6 is a screen shot showing a user interface for the content manager shown in
5 Figure 1.

6 [0015] Figure 7 is a screen shot showing a user interface for a template settings program
7 accessed through the content manager shown in Figure 6.

8 [0016] Figure 8 is a screen shot showing the user interface for the content manager when
9 initiating a content transfer.

10 [0017] Figure 9 is a screen shot showing a user interface for the remote manager shown in
11 Figure 1.

12 [0018] Figure 10 is a flow chart illustrating the creating, editing and use of smart objects,
13 intelligent templates and play lists using the system shown in Figure 1.

14 DETAILED DESCRIPTION OF THE DRAWINGS

15 [0019] following provides a digital signage content management system that uses existing
16 interfaces such as web interfaces and turns existing commercially available graphics programs
17 such as web based tools or locally run programs such as Microsoft PowerPoint® into a digital
18 signage platform to facilitate developing and managing digital signage applications. The
19 following enables digital signage content to be professionally created without requiring custom
20 programming for each and every stream of new and/or changing content.

21 [0020] Referring now to Figure 1, in the example shown, the management system comprises
22 an asset management server 10 for distributing new content to remote locations such as remote
23 content players 22 and kiosks 24, and for receiving continuous status updates from the remote
24 players 22, typically over a network 19. The server 10 can automatically send system alerts to
25 support staff by email if problems should arise. The server 10 obtains real time data 18, such as

21757682.1

1 news headlines, stock-market data, weather or custom content created by authors. The real time
2 data 18 can be obtained in various formats such as RSS, HTML, XML and SQL to name a few.

3 **[0021]** The management server 10 operates with various other elements and, in many
4 applications, various other entities to arrange and distribute the content to the players 22 and
5 kiosks 24 or any other entity that utilizes the content. It will be appreciated that each element
6 shown in Figure 1 may be run independently and locally or may be combined with any one or
7 more of the other elements at centralized locations. Also, as shown in Figure 1, the elements
8 may be accessed over the network 19 and thus provided on a web-based platform in addition to
9 or rather than a locally run instance. The management server 10 obtains play lists of one or more
10 intelligent templates 34 from a content manager 14, who accesses the intelligent templates 34
11 from a template library 17. The management server 10 is then able to combine the real-time data
12 18 with the respective templates 34 to continually and automatically update the content provided
13 through the templates 34.

14 **[0022]** The intelligent templates 34, which are explained in greater detail below, are created
15 by an authoring entity or program 11, which utilizes a template generator 12 an existing and
16 commercially available, or otherwise convenient or familiar graphics program 13. The template
17 generator 12 allows the user to create templates 34 using the familiar interface and functions
18 provided by the graphics program 13. One example of a particularly suitable graphics program
19 is Microsoft PowerPoint®. As can be seen, a web control interface 16 (e.g. a computer
20 connected to the program 11 over the Internet) can be used to provide a web-based development
21 environment. It will be appreciated that the template generator 12 and graphics program 13 may
22 also be run locally or be PC based. The template generator 12 allows users to work directly
23 inside the graphics program 13 such as PowerPoint® to develop high end digital signage
24 productions. Using smart objects 26 as shown in Figure 2 and intelligent templates as shown in
25 Figure 3, the template generator 12 can extend the capabilities of the graphics program 13 by
26 adding signage features, allowing users to author content that includes video, animations,
27 graphics, audio, live data and more.

1 [0023] The template generator 12 utilizes smart objects 26 to build the intelligent templates
2 34. In this way, certain properties and parameters defined for a smart object 26 can be inherited
3 by the intelligent templates 34 such that by modifying an object 26, a template 34 can be
4 modified. This allows standard objects 26 and templates 34 to be created that can change for
5 each and every instance and use of the object 26 and template 34 for different applications. The
6 smart objects 26 can be stored in an object library 15.

7 [0024] As can be seen in Figure 2, the smart object 26 includes a graphic layout 28, data
8 source 30 and behaviour logic 32 to provide conditions for updating content provided by the
9 object 26. As can be seen in Figure 3, the intelligent template 34 has data sources 36, scheduling
10 rules 38, behaviour logic 40 and a graphic layout 42.

11 [0025] The template generator 12 should be capable of dragging-and-dropping templates 34,
12 adding unlimited layers of objects 26 anywhere on the screen, including multiple video windows,
13 crawling data tickers, 3D animations etc. as well as enable the user to select from a library of
14 professional, customizable templates and reusable objects. The user can create templates 34
15 once then automatically update video clips, images and dynamic text elements based on actions
16 and business rules. The user can also link slide elements with information from databases,
17 websites, RSS feeds or any other data source. Custom control or behaviour logic 32, 40 can also
18 be defined using a scripting engine (not shown).

19 [0026] The content manager 14 allows the user to manage all aspects of control play-out
20 from a single interface. This includes building play lists, updating template 34 information,
21 defining play out schedules, and scheduling content delivery to any player 22 or group of players
22 on the network 19. Updating templates 34 can be automated using predefined rules, making
23 updating even complex play list content much easier and consistent, thus allowing non-skilled
24 users to create custom signage. Graphics can be scheduled to play out at specific intervals or
25 time-of-day; expiry dates can automatically delete slides after a specified time; templates 34,
26 play lists and other assets can be dragged onto individual players 22 for automatic distribution;
27 and shut down times for remote displays can be scheduled to preserve power and overall screen
28 life. A web control 16 may provided to enable the user to manage content with web pages. This

1 allows non-technical users to interact with the system to update specific areas without requiring
2 knowledge of or access to the authoring tools such as the template generator 12. It will be noted
3 that several separate web control interfaces 16 may be used (e.g. one for interacting with each
4 element in Figure 1) and one is shown in Figure 1 for clarity.

5 **[0027]** The players 22 and kiosks 24 are typically broadcast rendering engines that deliver
6 high quality output. The players 22 dynamically composite slide elements in real-time to
7 generate the final video output. In this way, individual content elements do not need to be re-
8 rendered whenever a change is required. The players 22 can use the same engine used by TV
9 networks etc. The players 22 can automatically download and store all content locally allowing
10 uninterrupted play out in the event of a network disruption. Inputs from cameras, cable/satellite
11 feeds and DVD players should be supported as well as any resolution output from analog
12 NTSC/PAL to DV1 to HD-SDI. The players 22 can also use an auto recovery feature to allow
13 the player 22 to self-diagnose and solve system errors without user intervention.

14 **[0028]** A remote network manager 20 may be used to manage the content and monitor
15 network status across multiple locations. The entire network 19 can be monitored and, should a
16 problem be detected, corrective action can be taken prior to being noticed by the users. From the
17 remote manager 20, content can be dragged-and-dropped from a local network to one or more
18 remote locations, network of players 22 can be browsed and viewed live from any location,
19 email alerts can be received when a problem occurs, detailed technical stats can be viewed for
20 any player 22, system and run logs can be viewed for any player 22, disk space and usage can be
21 managed for all players 22 automatically, and control commands can be sent to one or more
22 locations for automating tasks such as turning screens on or off.

23 **[0029]** The system shown in Figure 1 can be adapted for use in any environment. For
24 example, players 22 can be used in retail displays, public information displays, airports,
25 museums, in entertainment and hospitality, casinos, for corporate communications, university
26 displays, medical facility displays, multi-zone information channels, employee training functions
27 and self-service and information kiosks (e.g. kiosk 24).

1 **[0030]** The system introduces a concept of reusable “smart” components that include a
2 plurality of graphics or video elements, a data layer, and a behaviour layer. These self-contained
3 components can be used to generate a portion of a display, such as a weather or stock ticker, or
4 an entire full-screen video output comprising multiple elements, each with its own set of data
5 sources and individual behaviours.

6 **[0031]** The use of smart components greatly reduces the need for specialized training on the
7 part of the end user. Whereas in prior systems a user required a certain minimum level of
8 competency as a graphic artist or software developer, the introduction of smart components
9 allows users without any specialized knowledge to quickly and easily create complete video
10 graphics digital signage applications that combine real-time information sources with dynamic
11 display characteristics.

12 **[0032]** As shown in Figures 2 and 3, the system operates using two main components:
13 “Smart Objects” 26 and “Intelligent Templates” 34. Smart objects 26 form the building blocks
14 needed to create a digital signage display, and intelligent templates 34 dictate the layout and
15 production logic needed to generate the final video graphics output. Multiple smart objects 26
16 can be included in an intelligent template 34, and multiple templates 34 can be created from a
17 library of smart objects 26.

18 **[0033]** Smart objects 26 in this example, may include the following basic characteristics: 1)
19 An object 26 can contain an unlimited number of graphical elements, including text, images,
20 animations, and video; 2) Multiple objects 26 can be used simultaneously to form a composited
21 rich media final output; 3) Each object 26 is entirely self-contained, including all of the graphical
22 and video elements, data sources, and business rules needed to generate a final output; and 4)
23 Objects 26 can be self-configuring, allowing the output to be dynamically modified in response
24 to data triggers, without the need for user intervention. An example of this is a weather graphics
25 that automatically displays a cloud animation when it is cloudy or a sun animation when it is
26 sunny, or a financial graphic that shows a red downward pointing arrow when the stock market is
27 down or a green up arrow when the market is up.

1 **[0034]** The smart objects 26 are considerably powerful for the end user, since it not only
2 encompasses an object's graphical elements 28, but also the rules or behaviour logic 32 which
3 define how the graphical elements will respond to continuously changing inputs from the data
4 sources 30.

5 **[0035]** A typical example of using smart objects 26 involves retail displays installed in a
6 department store. A display could be configured to display a continuous loop of video, images,
7 and promotional text associated with the specials of that week. A smart object 26 within the
8 display layout can be designed to integrate with the department store's inventory management
9 system. If the inventory level for any of the items displayed on the screen falls below a minimum
10 threshold, the object automatically switches to an alternative set of specials on items for which
11 inventory is available. Without smart objects 26, this example would require custom software
12 development for each screen layout that is required. With smart objects 26, the rules are defined
13 once, and then reused again and again for any number of screen layouts. Also, the behaviour
14 logic 32 can be used to interrelate multiple objects 26 such that an event relevant to one object 26
15 triggers a change in another object 26. Using the above example, in a retail environment, a
16 change in the weather, e.g. it begins to rain can trigger a change in advertising for merchandise,
17 e.g. rain wear or umbrellas.

18 **[0036]** To use an object 26 to create a portion of a final video output, a user can simply drag
19 and drop the component from a browser window (organized in the template generator 12) onto
20 the workspace, or "canvas" (provided by the graphics program 13). The component's graphical
21 elements and layout, as well as any internal logic and business rules, are automatically added to
22 the canvas. By dragging and dropping multiple objects 26 onto the canvas, a user can create a
23 complete finished layout in a matter of seconds. This results in much more than a simple
24 graphics layout. When the layout is displayed on a video screen, each of the components
25 automatically configures itself and automatically displays live graphics and video information
26 based on its internal logic 32 and behaviour definitions.

27 **[0037]** Typical examples of smart objects include: 1) Weather objects showing real-time
28 weather conditions; 2) Sports tickers showing live sports results; 3) Headline tickers that

1 continuously scroll live news information; 4) Video windows that automatically play through a
2 loop of video content; and 5) Alert pop-ups that automatically appear in the event of a fire alarm
3 or weather warning.

4 **[0038]** Intelligent templates 34 typically include the following characteristics: 1) Layout
5 information defining where each individual object is located on the final output display; 2)
6 Dynamic parameters that can be changed by the user without requiring a re-edit of the template,
7 which can be as simple as a video filename that can be set by the user for a full screen video
8 template, or as complex as a drop list of branding options, each of which completely redefines
9 the entire template layout with a single click; 3) Rules defining how individual objects interact
10 with each other; 4) Scheduling information, defining where and when each template should be
11 displayed; 5) Expiry dates for content, allowing templates to be displayed only within a specified
12 validity period; and 6) Business rules dictating how a template should be reconfigured based on
13 dynamic data inputs, e.g., a single template which, when displayed in a certain location, displays
14 video content applicable to that audience demographic, but when displayed in a different
15 location, displays entirely different video content applicable to a different audience demographic.

16 **[0039]** Templates 34 should include everything necessary to generate a complete digital
17 signage output, including graphical elements, video components, multiple data inputs,
18 animations, business rules, and scheduling information 38.

19 **[0040]** Using the combination of smart objects 26 and intelligent templates 34, users can
20 build libraries of hundreds or thousands of reusable components, which can be stored in the
21 object library 15 and the template library 17. These libraries 15, 17 can be shared between users
22 in the same physical location or in multiple geographic locations, e.g. through a web control
23 interface 16. For many applications, generic default or otherwise existing objects 26 and
24 templates 34 can be used "as is" without modification. For other applications, users can select an
25 existing object 26 or template 34, modify the parameters of that object 26 or template 34, and
26 save it as a new component.

27 **[0041]** A typical scenario would involve multiple levels of users: 1) Advanced users would
28 create objects by combining graphics, data, and programming elements; 2) Intermediate users

21757682.1

1 would build new digital signage projects by drawing from existing library elements; and 3) Less
2 advanced users would manage the scheduling and custom information displayed by the objects
3 and templates on the final display output.

4 **[0042]** Turning now to Figure 4, an exemplary user interface (UI) for the template generator
5 12 and graphics program 13 is shown. In this example, the template generator 12 and graphics
6 program 13 are used side-by-side to facilitate easy drag and drop motions for placing objects 26
7 onto the “canvas” provided in this example by Microsoft PowerPoint®. The template generator
8 12 comprises a slide navigation window 50 for illustrating a list of the elements that comprise the
9 template 34 that is currently being edited using the template generator 12. These elements may
10 include primitives like text and images or complete objects. For example, each object can be
11 created and presented in a different slide and multiple slides can be used to build a template 34.
12 A properties window 52 is also provided for setting parameters of the template 34 that is being
13 created in the graphics program 13. A style browser 54 is also provided, which in this example
14 shows a list of object previews 56. The style browser 54 can also show predefined text styles,
15 image and video clip libraries and thumbnail views of all completed templates 34 in the template
16 library 17. As can be seen in Figure 4, a weather object 26a has been selected from the browser
17 54 and placed in the graphics program 13, e.g. by a drag and drop operation. The weather object
18 26a will carry with it, the data sources 30 and behaviour logic 32 required to update the content
19 provided through the object 26a. This in turn will change the template 34 and thus the final
20 output as the content is played.

21 **[0043]** Figure 5 illustrates a fully generated template 34, which includes the weather object
22 26a, a stock object 26b, a sport scores object 26c, a video display object 26d and a news ticker
23 object 26e. This allows the user to define a particular layout and combination of objects 26 for a
24 particular application. The template 34 itself has parameters, as will be explained below, that
25 then allows the user to use the same template 34 in multiple instances and in multiple locations
26 or applications to avoid re-developing standard layouts.

27 **[0044]** Figure 6 illustrates an exemplary UI 60 for the content manager 14. The content
28 manager UI 60 allows a user to import templates 34 from the template library 17 and configure

1 the templates 34 such that they play out in a particular way for that application. It can be seen
2 that six (6) templates 34 are defined for the play list in this example and a template settings
3 window 68 is loaded as a new template 34 is added. The template window 68 lists the
4 parameters so that the user can review these and decide whether or not modifications are
5 necessary and if so, make the necessary changes. A new template 34 may be selected from a file
6 explorer window 62 and thus it may be appreciated that more than one template library 17 may
7 exist. A network window 64 is also provided to give network information for the players 22
8 and/or kiosks 24 relevant to a particular network of content providers, which may or may not be
9 related to each other. A search tool 72 may be provided to enable the user to search for
10 templates 34 and the templates available at either a specified location in the file structure shown
11 in the window 62 or found in a search can be displayed. The play list properties can also be set
12 and modified using a properties tool 78. Figure 8 shows a transfer tool 100 that can be used in
13 the content manager UI 60 for selecting the remote destinations to which the content is provided
14 by sending a particular play list.

15 **[0045]** Turning now to Figure 7, a template settings UI 80 is shown which allows parameters
16 of a template 34 being created using the template generator 14, to be modified. A template 34
17 that has been selected is named in box 82 and a pointer 84 to the main template library folder is
18 also provided. A categories list 86 is provided, which lists all categories that have been defined
19 in the template library 17. The selected categories list 88 shows the categories that the particular
20 template 34 belongs to. The template parameters 90 are listed below, which is a list of
21 parameters that the user can modify when adding the template 34 to a play list in the content
22 manager UI 60. This list can be the same list identified by numeral 70 in Figure 6. For example,
23 the logo is listed, which describes the logo, indicates that it is an object having a certain name,
24 property and type and gives the default value. Parameters 90 can be added, removed, copied,
25 pasted and edited through this list. Template properties 92 are also given, which can be edited.
26 For example, the slide duration can be specified. It can therefore be seen that during creation of
27 a template 34, the parameters and properties can be modified to “intelligently” define how the
28 final output is updated and how it may change during a play out. This can drastically reduce
29 development times and allows greater flexibility in creating custom signage and custom content.

1 [0046] Figure 9 shows a remote manager UI 102, which can be used as an administrative tool
2 for managing and monitoring the output of each player 22 and kiosk 24 in a given network of
3 content providers. In this example, a list of players 104 is provided, which specifies which play
4 list is being played. A transfer status window 106 may also be provided to give the user
5 information regarding the transfer of the play list to the content providers. A preview pane 108
6 is provided that shows a highlighted one of the destinations that is selected from a collection of
7 previews 112. A status window 110 can also be used to provide status information pertaining to
8 the selected player 22. The remote manager UI 102 can be used by the remote manager 20
9 shown in Figure 1 or can be used directly by the management server 10 if they are separate and
10 remote entities. In this embodiment, the remote manager UI 102 is a PC-based application for
11 monitoring networks of players 22 etc. but may also be a web-based interface and thus should be
12 accessible anywhere.

13 [0047] Turning now to Figure 10, a flow chart is shown that illustrates the creating,
14 modification, access and use of the objects 26, templates 34 and play lists generated therefrom.
15 At 200, an object 26 is created. This can be done at any time and then stored in the object library
16 15 at 202. In this example, an object 26 is accessed for two different reasons, namely to edit the
17 object 26 or its properties and parameters, and to add the object 26 to a template 34. At 204, the
18 object 26 is accessed from the object library 15 and edited at 206. This may then generate a new
19 distinct object 26 or may simply overwrite the existing object 26 at 208. In either case, the new
20 or modified object is stored in the object library 15. At 210, the object (and likely other objects
21 26) are accessed and a template 34 is created using those selected objects 26 at 212. The newly
22 created template 34 is then stored in the template library 17 at 214. Similar to the objects 26, the
23 templates 34 may be accessed for modification purposes, to create new templates 34 from
24 existing ones, and to generate play lists. At 216, the template 34 is accessed and edited at 218.
25 The template 34 is either modified and overwritten or a new template 34 created at 220 and the
26 new or modified template 34 is stored in the template library 17. At 222, a play list has been
27 generated or is in the process of being generated and the user wishes to add the template 34 to
28 the play list. In this case, the template 34 is accessed from the template library 17 at 224. The
29 play list, thus created then forms a set of content that is provided to the players 22 and/or kiosks

1 24 at 226, which store and queue up the content at 228. The content is then played at 230 by
2 executing the play list. It can be seen that this process can be repeated to obtain (e.g. download
3 or receive) new play lists.

4 **[0048]** It can therefore be seen that the above-described system can be used to provide a
5 digital signage content management system that uses existing interfaces such as web interfaces
6 and turns existing commercially available graphics programs such as web based tools or locally
7 run programs such as Microsoft PowerPoint® into a digital signage platform to facilitate
8 developing and managing digital signage applications. The following enables digital signage
9 content to be professionally created without requiring custom programming for each and every
10 stream of new and/or changing content.

11 **[0049]** Although the invention has been described with reference to certain specific
12 embodiments, various modifications thereof will be apparent to those skilled in the art.

Claims:

1. A method for generating content for a digital display comprising:
 - enabling a plurality of objects to be authored using a commercially available graphics program by enabling, for each object, an object layout to be defined for presenting data within said object using an interface provided by said commercially available graphics program, one or more data sources to be identified for obtaining said data, and behavior logic to be created for responding to changing inputs from said one or more data sources, said behavior logic being executable to self-configure a respective object to dynamically modify its output in response to at least one data trigger, without user intervention, said behavior logic interrelating one object with at least one other object to have an event related to said one object trigger a change in said at least one other object;
 - enabling said plurality of objects to be stored in a library to enable said objects to be available to multiple applications;
 - enabling a template to be authored using said commercially available graphics program by enabling a template layout to be defined using said interface;
 - providing access to said library;
 - enabling a plurality of selected objects to be obtained from said library and added to said template using said interface;
 - enabling respective data sources for said plurality of selected objects to be defined;
 - enabling scheduling rules to be defined for displaying said plurality of selected objects within said template;
 - generating said template by inheriting said data sources and said behavior logic of said plurality of selected objects to individually control said plurality of selected objects within said template;
 - enabling said template to be stored such that said template may be used in multiple applications; and
 - enabling said template to be used by a content provider for generating a digital display output by executing said behavior logic for said plurality of objects.

2. The method according to claim 1, wherein said graphics program is Microsoft PowerPoint® or a web-based graphics program.
3. The method according to claim 2, wherein each of said plurality of objects object comprises a data structure in a Microsoft PowerPoint® format or a format applicable to said web-based graphics program.
4. The method according to claim 1 wherein said template is stored in a template library.
5. The method according to claim 1, further comprising enabling at least one of said layout of said objects, said behavior logic, said scheduling rules and said data sources to be modified.
6. The method according to claim 1 wherein said object layout, said one or more data sources and said behavior logic are stored as a set of modifiable parameters.
7. The method according to claim 1, wherein said data is provided by said data source in real-time and said plurality of objects are object is configured to be updated in real-time.
8. A computer readable medium comprising computer executable instructions for generating content for a digital display, the computer executable instructions comprising instructions for:
 - enabling a plurality of objects to be authored using a commercially available graphics program by enabling, for each object, an object layout to be defined for presenting data within said object using an interface provided by said commercially available graphics program, one or more data sources to be identified for obtaining said data, and behavior logic to be created for responding to changing inputs from said one or more data sources, said behavior logic being executable to self-configure a respective object to dynamically modify its output in response to at least one data trigger, without user intervention, said behavior logic interrelating one object with at least one other object to have an event related to said one object trigger a change in said at least one other object;

enabling said plurality of objects to be stored in a library to enable said objects to be available to multiple applications;

enabling a template to be authored using said commercially available graphics program by enabling a template layout to be defined using said interface;

providing access to said library;

enabling a plurality of selected objects to be obtained from said library and added to said template using said interface;

enabling respective data sources for said plurality of selected objects to be defined;

enabling scheduling rules to be defined for displaying said plurality of selected objects within said template;

generating said template by inheriting said data sources and said behavior logic of said plurality of selected objects to individually control said plurality of selected objects within said template;

enabling said template to be stored such that said template may be used in multiple applications; and

enabling said template to be used by a content provider for generating a digital display output by executing said behavior logic for said plurality of objects.

9. The computer readable medium according to claim 8, wherein said graphics program is Microsoft PowerPoint® or a web-based graphics program.
10. The computer readable medium according to claim 9, wherein each of said plurality of objects comprises a data structure in a Microsoft PowerPoint® format or a format applicable to said web-based graphics program.
11. The computer readable medium according to claim 8 wherein said template is stored in a template library.
12. The computer readable medium according to claim 8, further comprising enabling at least one of said layout of said objects, said behavior logic, said scheduling rules and said data sources to be modified.

13. The computer readable medium according to claim 8, wherein said object layout, said one or more data sources and said behavior logic are stored as a set of modifiable parameters.
14. The computer readable medium according to claim 8, wherein said data is provided by said data source in real-time and said plurality of objects are configured to be updated in real-time.
15. A method for generating objects for a digital display comprising:
- enabling an object layout to be defined for presenting data within said object using an interface provided by an existing available graphics program;
 - enabling one or more data sources to be identified for obtaining said data;
 - enabling behavior logic to be created for responding to changing inputs from said one or more data sources, said behavior logic being executable to self-configure a respective object to dynamically modify its output in response to at least one data trigger, without user intervention, said behavior logic interrelating said object with at least one other object to have an event related to said object trigger a change in said at least one other object;
 - enabling said object to be stored in a library of objects with said at least one other object to enable said objects to be available to multiple applications; and
 - enabling said object to be used with said at least one other object in digital display output by:
 - enabling said object to be added, with at least one other object from said library of objects, to a reusable template that inherits said data sources and said behavior logic to individually control said object in conjunction with said at least one other object within said template; and
 - executing said behavior logic for said object, during use of the template in providing the digital display output.
16. The method of claim 15, further comprising enabling at least one of said layout of said objects, said behavior logic, scheduling rules and said data sources to be modified.

17. The method of claim 15, wherein each object from said library of objects comprises a data structure in a format applicable to a web-based graphics program.
18. The method of claim 15, wherein said object layout, said one or more data sources and said behavior logic are stored as a set of modifiable parameters.
19. The method of claim 15, wherein said data is provided by said data source in real-time and said object is configured to be updated in real-time.
20. A computer readable medium comprising computer executable instructions for generating objects for a digital display, comprising instructions for:
 - enabling an object layout to be defined for presenting data within said object using an interface provided by an existing available graphics program;
 - enabling one or more data sources to be identified for obtaining said data;
 - enabling behavior logic to be created for responding to changing inputs from said one or more data sources, said behavior logic being executable to self-configure a respective object to dynamically modify its output in response to at least one data trigger, without user intervention, said behavior logic interrelating said object with at least one other object to have an event related to said object trigger a change in said at least one other object;
 - enabling said object to be stored in a library of objects with said at least one other object to enable said objects to be available to multiple applications; and
 - enabling said object to be used with said at least one other object in digital display output by:
 - enabling said object to be added, with at least one other object from said library of objects, to a reusable template that inherits said data sources and said behavior logic to individually control said object in conjunction with said at least one other object within said template; and
 - executing said behavior logic for said object, during use of the template in providing the digital display output.

21. The computer readable medium of claim 20, further comprising enabling at least one of said layout of said objects, said behavior logic, scheduling rules and said data sources to be modified.
22. The computer readable medium of claim 20, wherein each object from said library of objects comprises a data structure in a format applicable to a web-based graphics program.
23. The computer readable medium of claim 20, wherein said object layout, said one or more data sources and said behavior logic are stored as a set of modifiable parameters.
24. The computer readable medium of claim 20, wherein said data is provided by said data source in real-time and said object is configured to be updated in real-time.

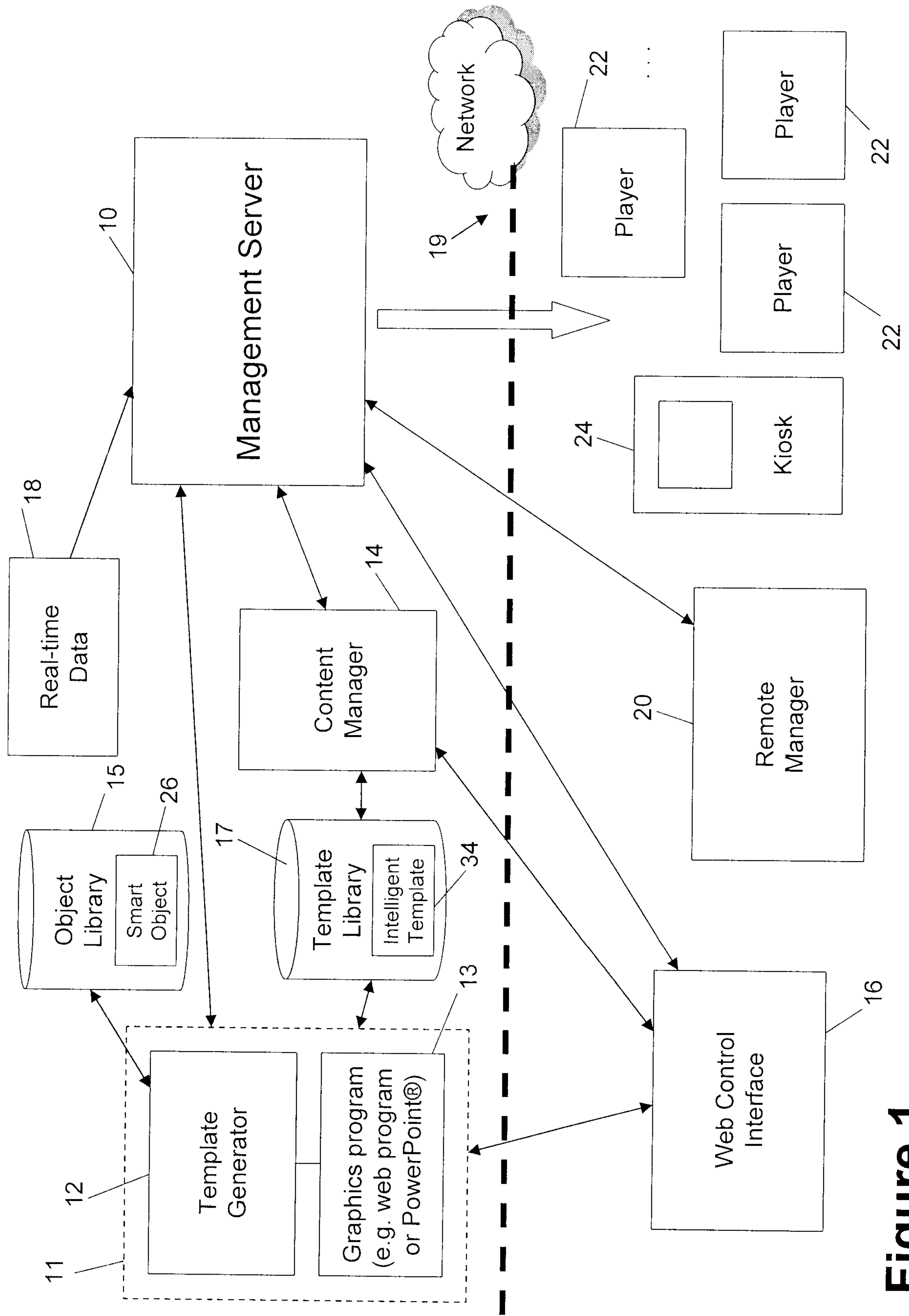


Figure 1

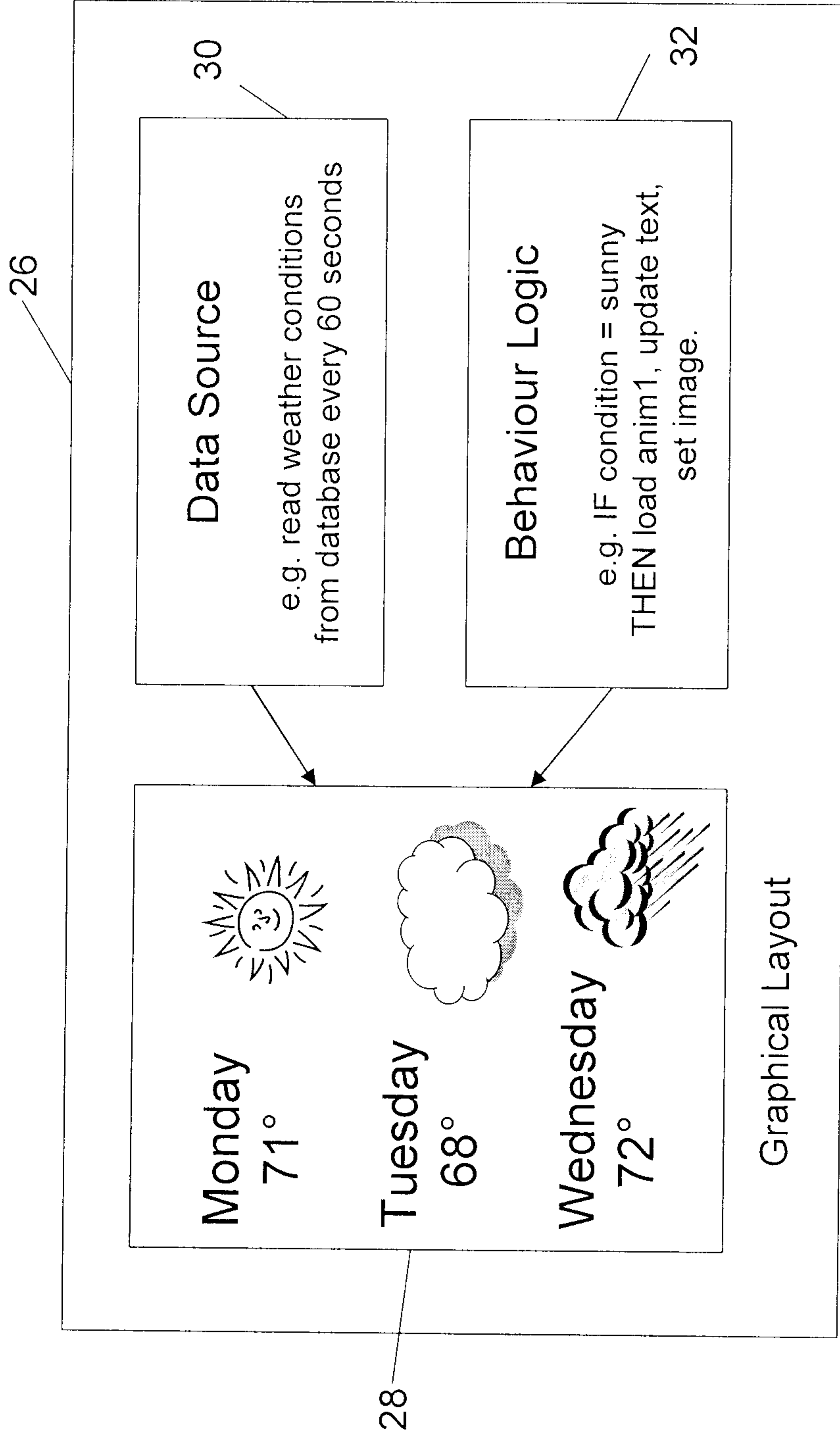


Figure 2

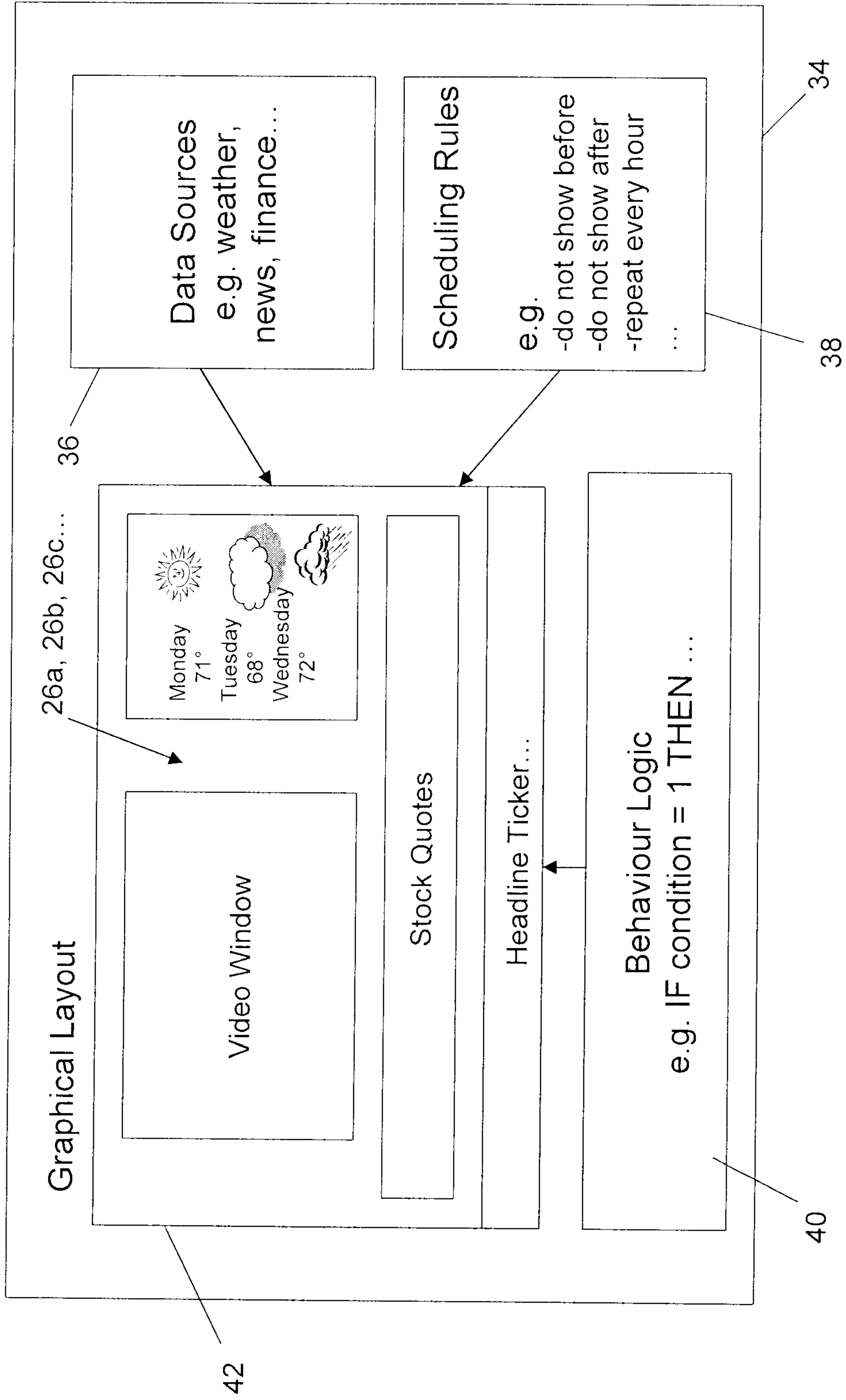
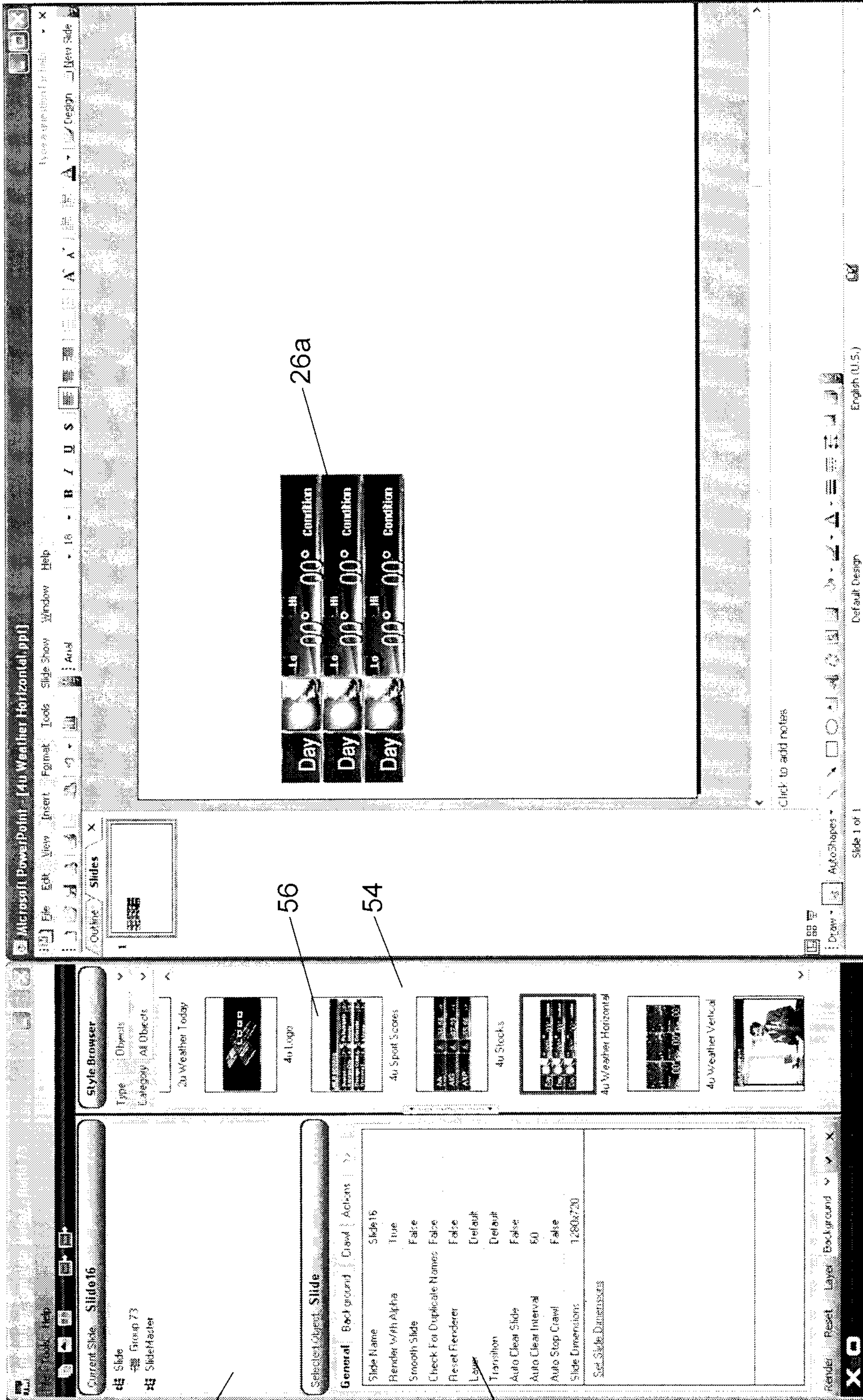


Figure 3



13

Figure 4

12

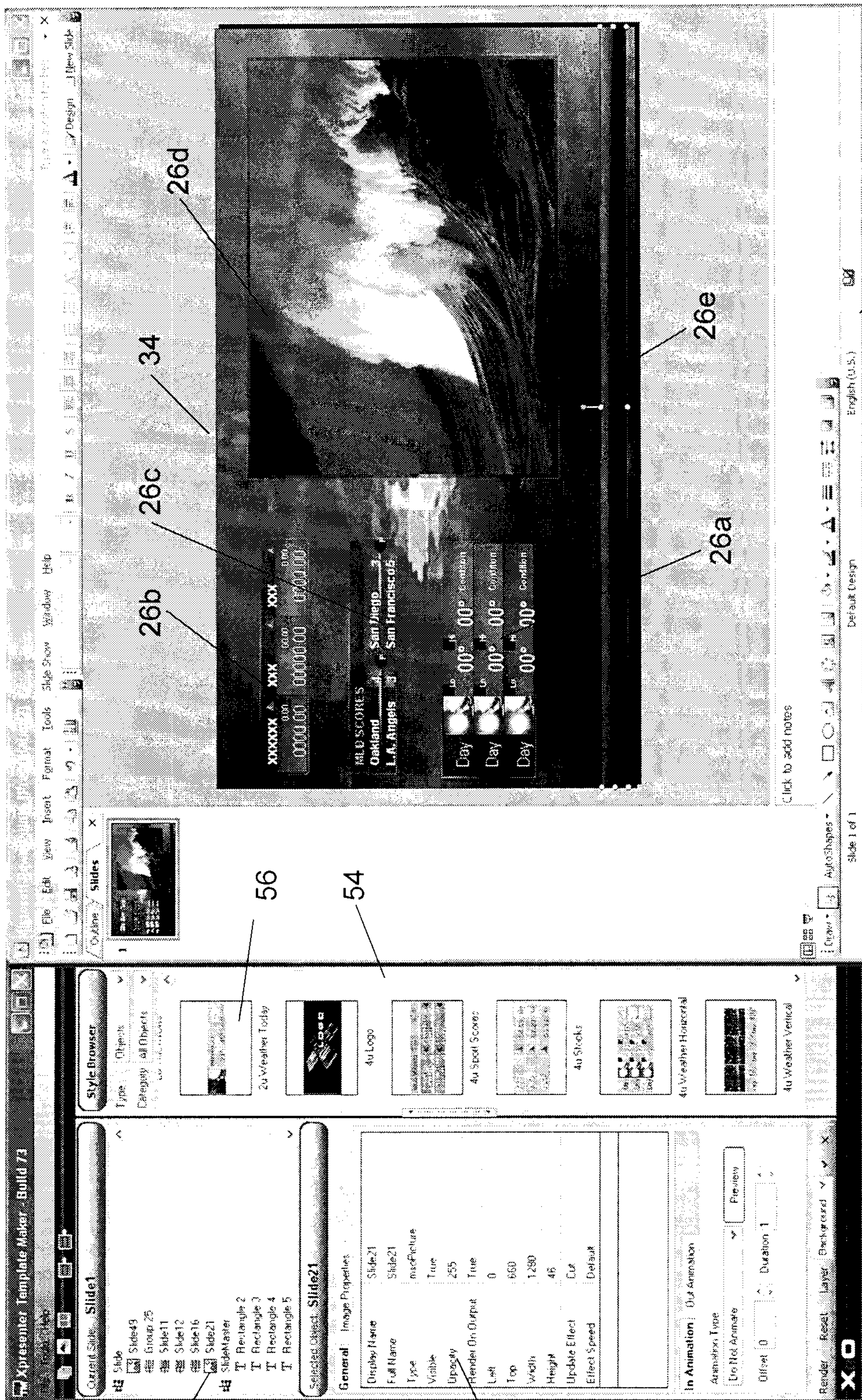


Figure 5

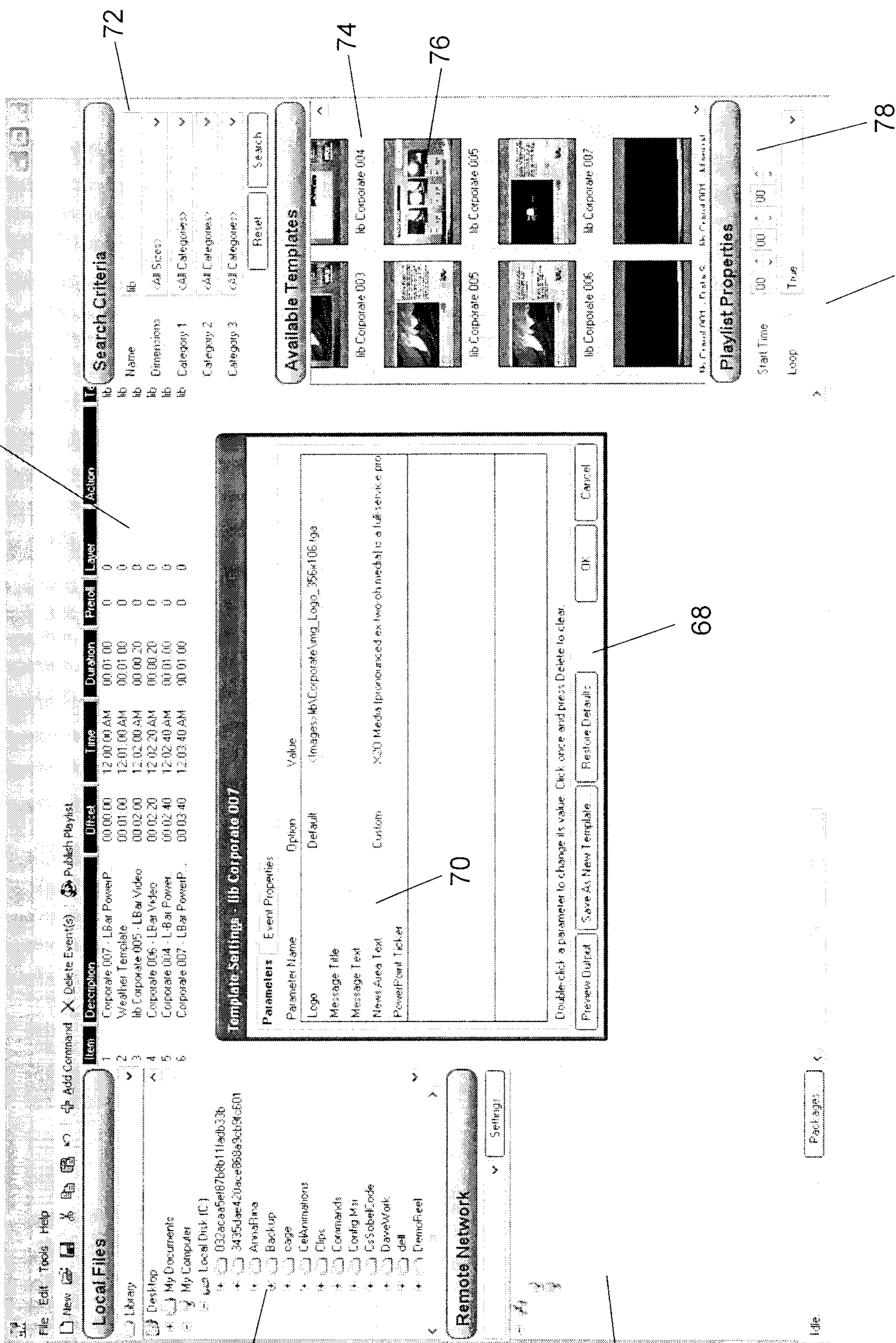


Figure 6

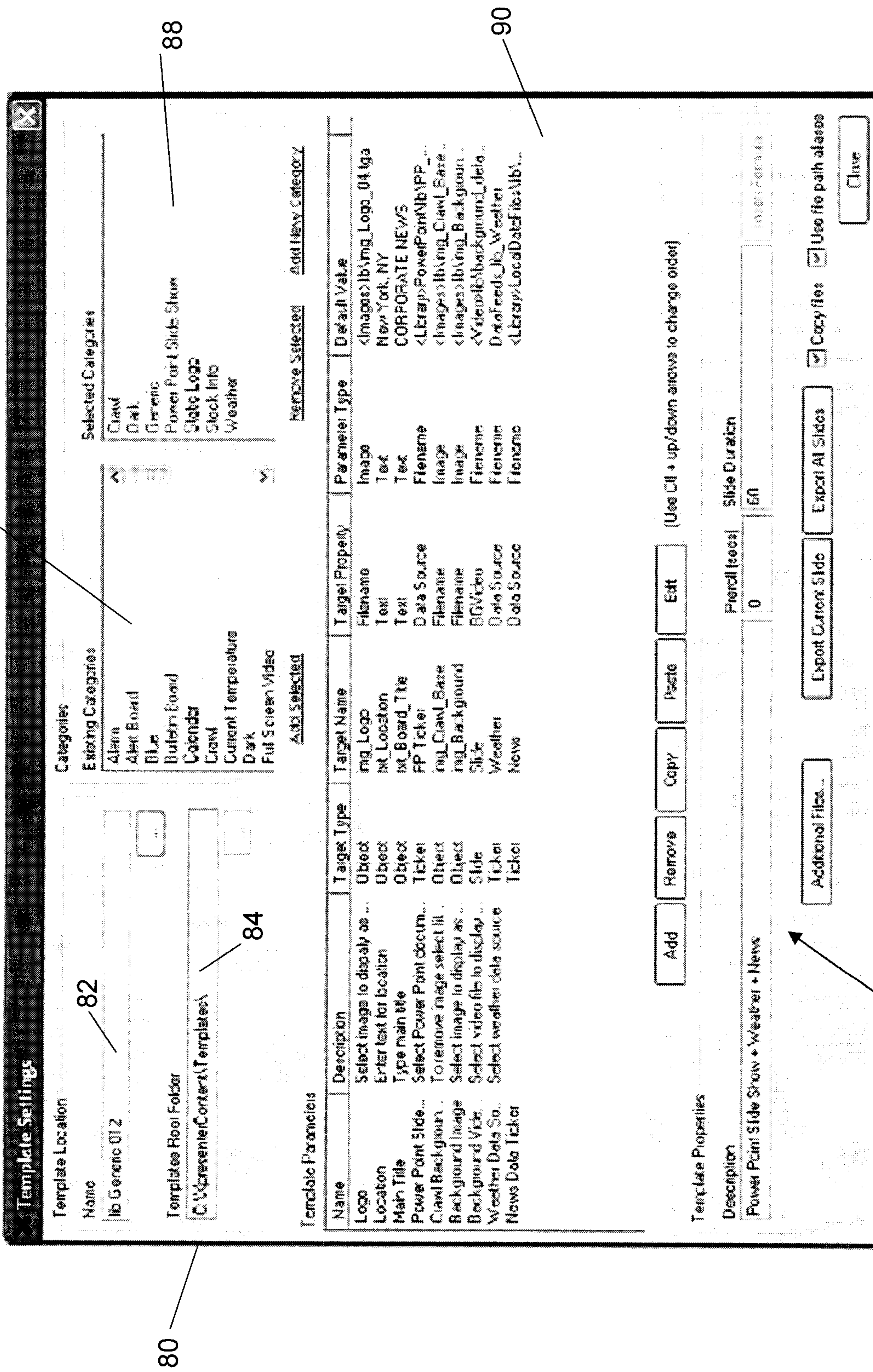


Figure 7

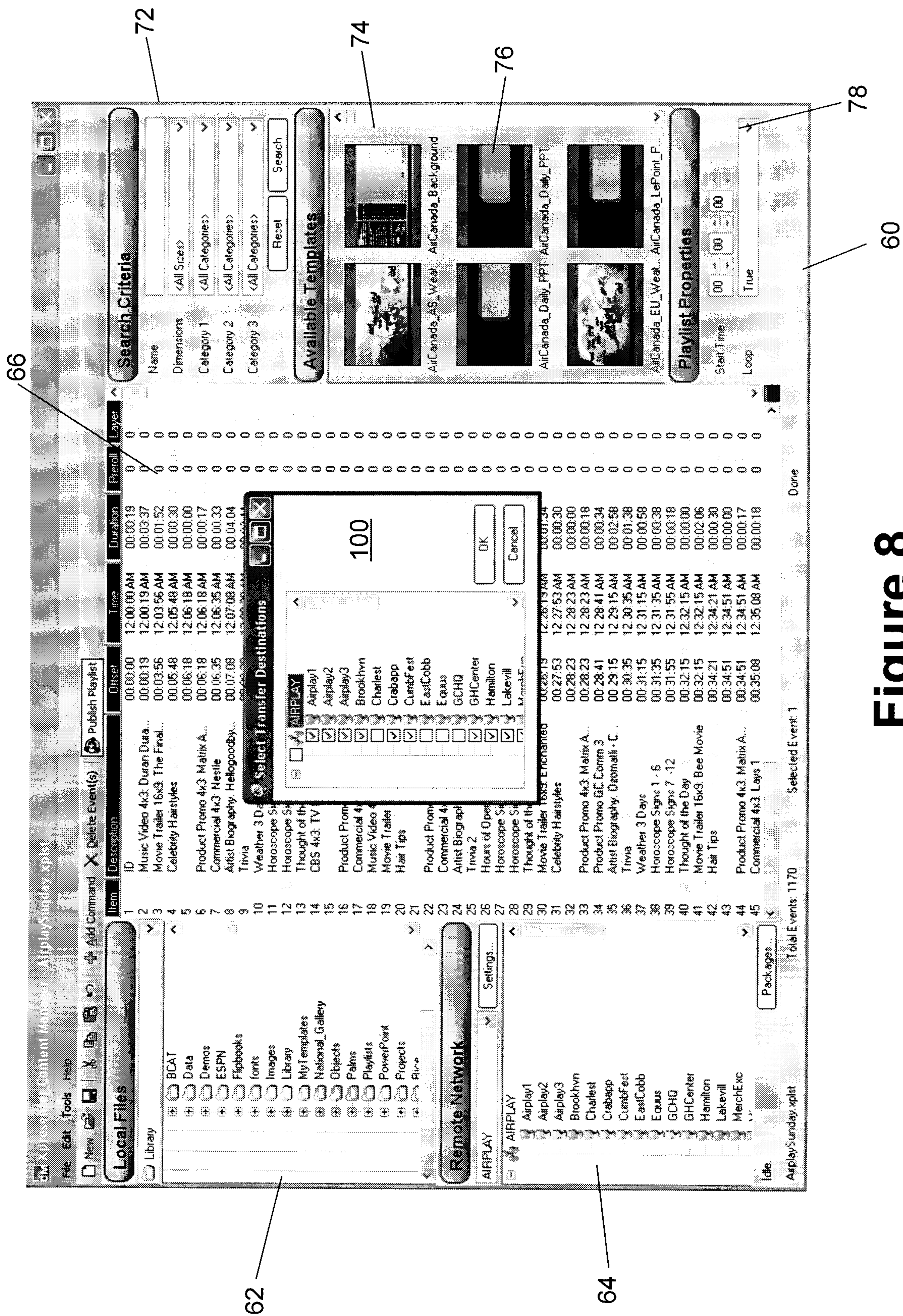
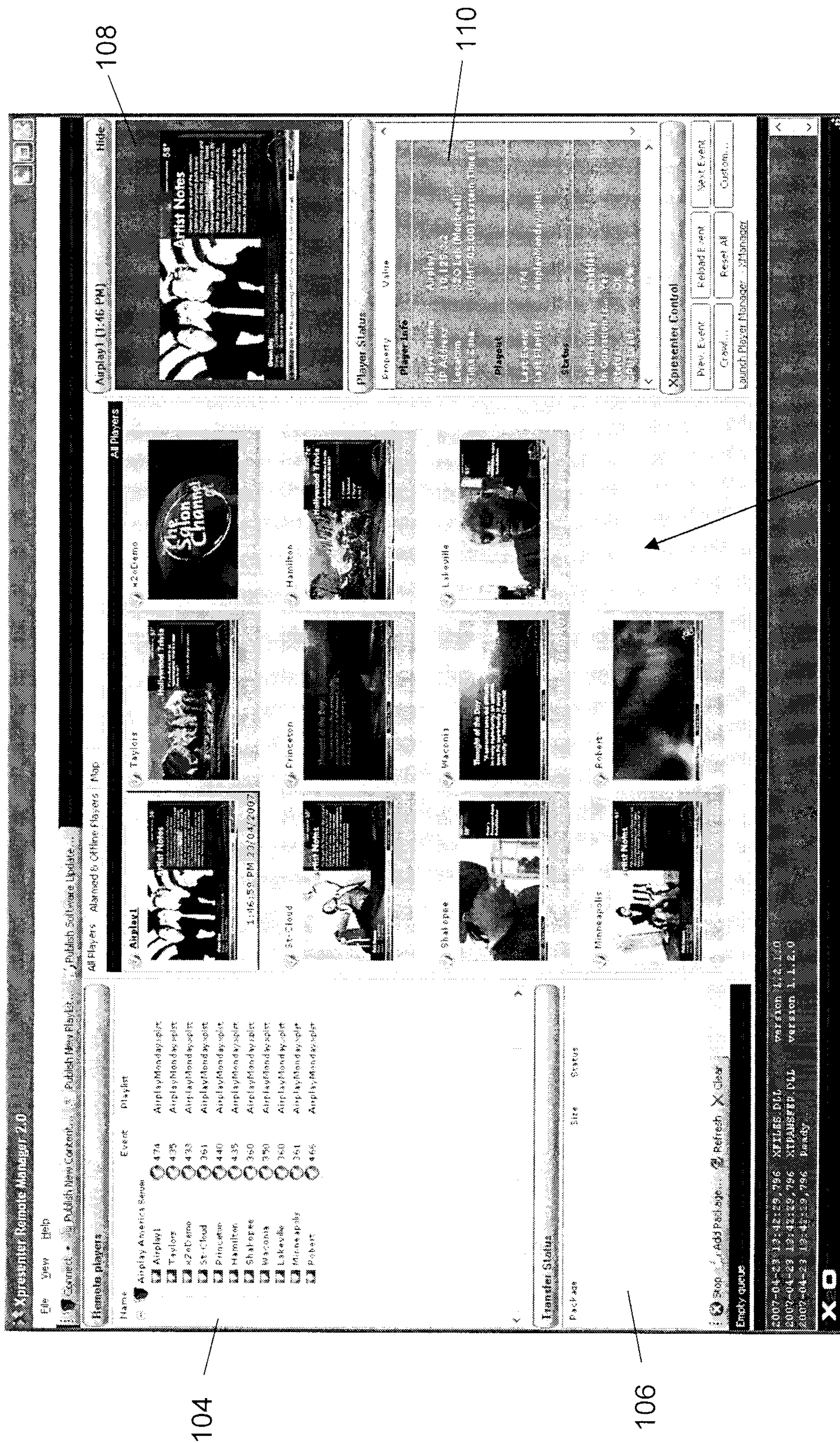


Figure 8



104

106

102

112

108

110

Figure 9

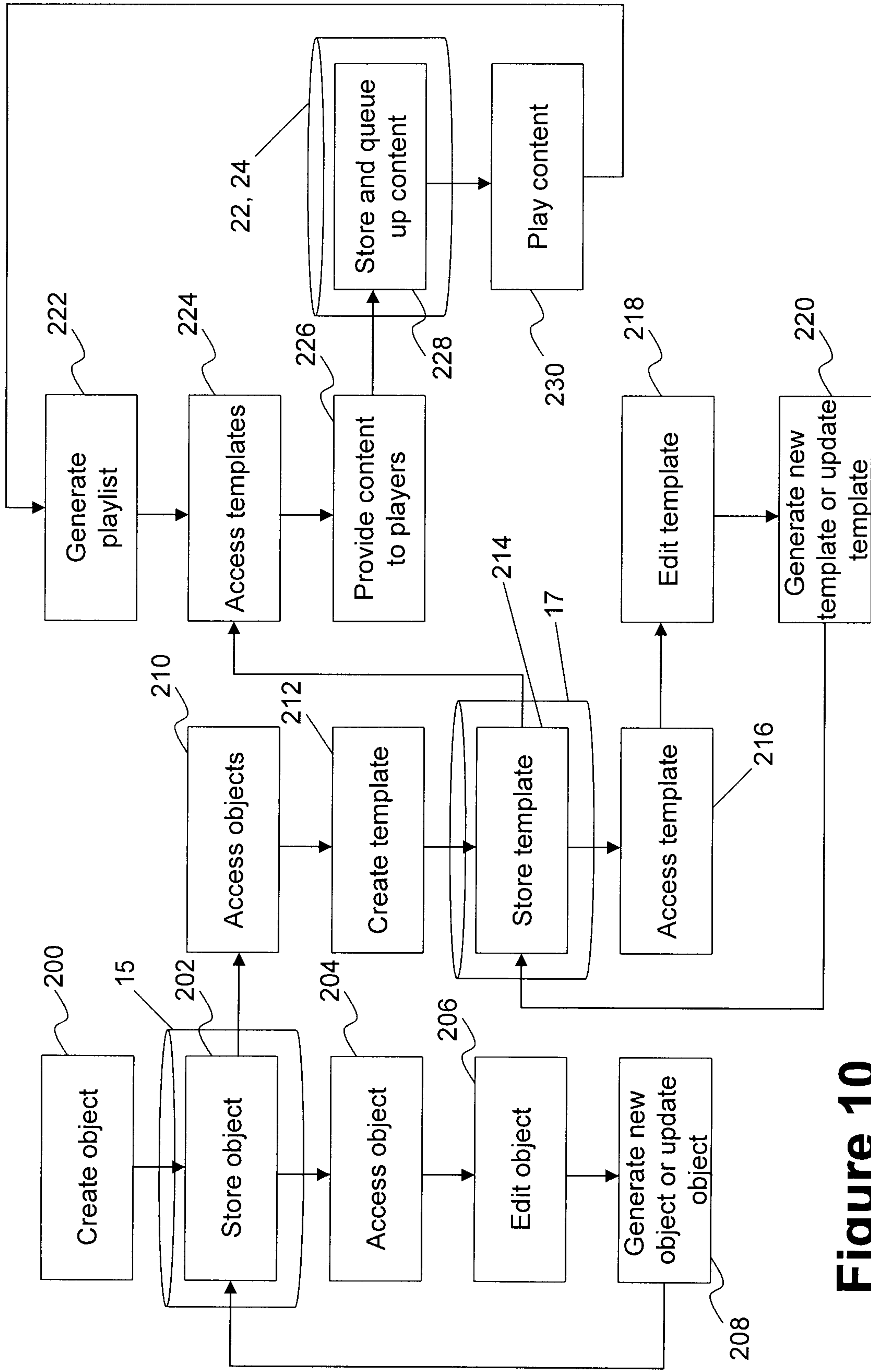


Figure 10

