(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
12 September 2008 (12.09.2008)

PCT

(10) International Publication Number
WO 2008/109581 A1

(51) International Patent Classification:
*G06F 9/44* (2006.01)

(21) International Application Number: PCT/US2008/055756

(22) International Filing Date: 4 March 2008 (04.03.2008)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
11/682,639 6 March 2007 (06.03.2007) US

(63) Related by continuation (CON) or continuation-in-part (CIP) to earlier application:
US 11/682,639 (CON)
Filed on 6 March 2007 (06.03.2007)

(71) Applicant *(for all designated States except US)*: **CLEARSPRING TECHNOLOGIES, INC.** [US/US]; 8000 Westpark Dr., Suite 625, McLean, Virginia 22102 (US).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **ALLEN, Stewart, O.** [US/US]; 1486 Waterfront Road, Reston, Virginia 20197 (US). **COSBY, Scott, F.** [US/US]; 11 Hayward Place, Rye, New York 10580 (US). **DEBUTTS, Hasseltine, R.** [US/US]; 1220 Delaplane Grade Road, Upperville, Virginia 20184 (US). **FATH, John, A.** [US/US]; 850 N. Randolph St., Apt. #706, Arlington, Virginia 22203 (US). **KEESAN, Matthew, J.** [US/US]; 2821 13th St., NW, Apt. #A, Washington, DC 20009 (US). **RADFAR, Hooman** [US/US]; 900 N. Stuart St., Apt. #1508, Arlington, Virginia 22203 (US). **REVERTE, Carlos, F.** [US/US]; 9221 Kendale Blvd., Miami, Florida 33176 (US).

(74) Agents: **BENNETT, Daniel, M.** et al.; Cooley Godward Kronish LLP, Attn: Patent Group, 777 6th Street, NW, Suite 1100, Washington, DC 20001 (US).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, SV, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US (patent), UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MT, NL, NO, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

*[Continued on next page]*

# (54) Title: METHOD AND APPARATUS FOR WIDGET AND WIDGET-CONTAINER DISTRIBUTION CONTROL BASED ON CONTENT RULES

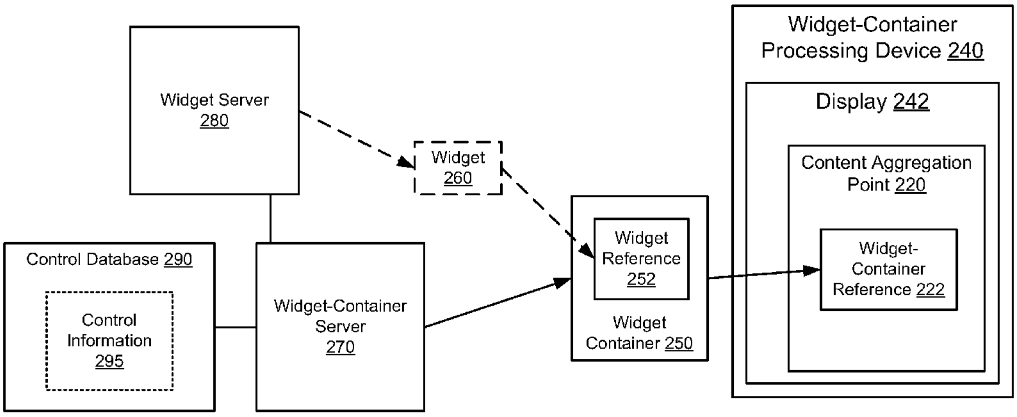

Widget Server 280
Widget 260
Control Database 290
Control Information 295
Widget-Container Server 270
Widget Reference 252
Widget Container 250
Widget-Container Processing Device 240
Display 242
Content Aggregation Point 220
Widget-Container Reference 222

FIG. 2

(57) **Abstract:** A method includes receiving an instruction used to define a widget-container containing a service module associated with a widget. The widget-container is a procedural software framework configured to be executed at a widget-container processing device in response to a reference to the widget-container being accessed from a content aggregation point. The widget-container has a portion configured for receiving the widget. The method also includes determining whether processing of the widget within the portion of the widget-container is restricted and/or allowed.

**Declaration under Rule 4.17:**

— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*

**Published:**

— *with international search report*

— *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments*

# METHOD AND APPARATUS FOR WIDGET AND WIDGET-CONTAINER DISTRIBUTION CONTROL BASED ON CONTENT RULES

## *Cross Reference to Related Application*

[1001] This application is a continuation of and claims priority to U.S. Patent Application No. 11/682,639, entitled "Method and Apparatus for Widget and Widget-Container Distribution Control Based on Content Rules," filed March 6, 2007, the disclosure of which is hereby incorporated by reference in its entirety.

## *Background*

[1002] The disclosed invention relates generally to control of distribution of content over a network, and in particular, to control of processing and/or distribution of static, media, and/or software objects in a procedural software framework.

[1003] The world wide web is a platform that has been used to exchange various forms of content including videos, text, music, etc. Often this content is distributed to users and/or computers in an ad-hoc fashion, for example, using e-mail or as files embedded in a web-page. Recently, various forms of "viral" distribution or replication of content have been developed that allow users to more easily spread content to other users than previously known ad-hoc methods. Although these viral distribution methods are more convenient than distributing content in an ad-hoc fashion, they have many shortcomings. For example, they do not provide for methods to control undesirable distribution of content based on criteria defined by, for example, an owner of the content. In other words, the spreading of content using ad-hoc methods and/or forms of viral spreading cannot be controlled in a useful way. Thus, there is a need for an apparatus and method for restricting or permitting the distribution of content.

## *Summary of the Invention*

[1004] A method includes receiving an instruction used to define a widget-container containing a service module associated with a widget. The widget-container is a procedural software framework configured to be executed at a widget-container processing device in response to a reference to the widget-container being accessed from a content aggregation point. The widget-container has a portion configured for receiving the widget. The method also includes determining whether processing of the widget within the portion of the widget-container is restricted and/or allowed.

## *Brief Description of the Drawings*

[1005] The present invention is described with reference to the accompanying drawings. In the drawings, identical or like reference numbers indicate identical or functionally similar elements.

[1006] FIG. 1 is a schematic block diagram of a widget-container host configured to control, configure, and coordinate the sending of widget-containers and/or components of a widget-container to widget-container processing devices based on control information, according to an embodiment of the invention.

[1007] FIG. 2 is a schematic diagram illustrating a widget-container associated with a content aggregation point being displayed at a widget-container processing device, according to an embodiment of the invention.

[1008] FIG. 3 is a table that illustrates a set of control rules that are based on content aggregation point identifiers, according to an embodiment of the invention.

[1009] FIG. 4 illustrates a set of category-based control tables that can be used in a control determination, according to an embodiment of the invention.

[1010] FIG. 5 is a flowchart that illustrates a control determination performed after a widget-container has been sent to a widget-container processing device, according to an embodiment of the invention.

[1011] FIG. 6 is a flowchart that illustrates a method for restricting or allowing processing of a component associated with a widget-container before sending the widget-container for display in a content aggregation point, according to an embodiment of the invention.

## *Detailed Description*

[1012] A widget-container (also can be referred to as a container) is a procedural software framework that can be configured to contain a widget and/or contain at least one service module that can be associated with the widget and/or widget-container. As a procedural software framework, the widget-container can be a series of instructions that are executable (e.g., interpretable) by, for example, a processor of, for example, a computer or mobile phone. The widget and/or service module is "contained" in the widget-container when a widget and/or service module is either referenced in a widget-container or actually integrated into the procedural software framework of the widget-container. In some embodiments, the widget-container includes a portion that is configured for processing of the widget within the widget-container. The widget and/or service module when being contained in the widget-container can be referred to as being wrapped or containerized in the widget-container. In some embodiments, the widget and/or the service module (or references to the widget and/or service module) can be referred to as components or portions of the widget-container.

[1013] The widget-container is a portable framework that can be referenced in (e.g., embedded in, referenced using an embed or object tag) and/or accessed from/using a content aggregation point (e.g., web-page, wireless access protocol (WAP) page). For example, when a request for the widget-container (and a widget associated with the widget-container) is received in response to a reference to the widget-container being accessed from a content aggregation point, the widget-container can be served for execution within the content aggregation point in response to the request. The widget can be any type of object such as a static data object (e.g., text or textual document), media object (e.g., video, mp3, or image), and/or software object (e.g., javascript, Flash program) that can be contained (e.g., integrated or referenced) and/or processed in the widget-container. Components of the widget-container are processed in the widget-container when they are, for example, executed, rendered, manipulated, and/or displayed.

[1014] Placing, sending (e.g., sending and instance of, serving and instance of), and/or processing of a portion associated with a widget-container (e.g., entire widget-container) can be restricted or allowed (e.g., permitted) based on control information (e.g., restriction/blocking information, allowance/permission information) and based on parameter values/identifiers, for example, included in a request for a widget-container. This determination can be referred to generically as a control determination, or as a restriction determination or an allowance determination. The parameters/identifiers can be, for example, extracted from a reference associated with a widget-container and/or associated with a content aggregation point where a reference to the widget-container has been placed.

[1015] The control information can be, for example, a control rule, a control condition, and/or a control instruction that can be associated with any portion associated with the widget-container. In some embodiments, for example, control information can be based on a metadata parameter value (e.g., tracking parameter values) associated with a portion associated with a widget-container and stored in a database such as a metadata database. More details regarding metadata parameter values as related to a widget-container are set forth in co-pending Application Serial No. 11/537,375, “Method and Apparatus for Widget Container/Widget Tracking and Metadata Manipulation,” which is incorporated herein by reference in its entirety.

[1016] A control determination (e.g., restriction determination, allowance determination) can be performed at various times. For example, a control determination can be performed before or in response to a widget-container being placed in a content aggregation point, before the widget-container is sent to a widget-container processing device, or after the widget-container is sent to a widget-container processing device. For example, a widget can be prevented from processing in a widget-container before the widget-container is sent to a widget-container processing device based on, for example, content included in a content aggregation point where the widget is to be processed. In some embodiments, a control determination can prevent (e.g., block) a widget-container from being placed in one or more specified content aggregation points.

[1017] A content aggregation point can be, for example, a desktop, a start page, a wireless application protocol (WAP) gallery, a gallery, a portal, a processor readable-vehicle (e.g., web-page, web site), and/or a directory. WAP galleries, web galleries, and so forth are types of content aggregation points that can be referred to as content distribution points because they can be vehicles from which content such as widgets and/or widget-containers can be distributed. A content distribution point can be a type of processor-readable vehicle. More details regarding processor-readable vehicles are set forth in co-pending Application Serial No. 11/537,362, “Method and Apparatus for Widget-Container Hosting and Generation,” which is incorporated herein by reference in its entirety.

[1018] The widget-container can be defined (e.g., dynamically generated) and/or modified to control processing and/or sending associated with the widget-container in accordance with a control determination (e.g., restriction determination, allowance determination). Modifying the widget-container, for example, can enable control of the processing and/or sending of portions of a widget-container such as a widget and/or a service module even when these components are served from separate devices/locations. Statistics related to control determinations can be collected and/or processed and proactive control determinations can be performed based on the placement (e.g., parentage, tracking) of widget-containers. The statistics can be based on tracking parameters associated with a metadata database.

[1019] Although many of the embodiments within the detailed description are described with respect to portions associated with widget-containers (e.g., widget associated with a widget-container, service module associated with a widget-container, widget-container framework, entire widget-container containing a widget(s)and/or a service module(s)) a person of skill in the art, having the benefit of this disclosure, can recognize that the methods and apparatus associated with the embodiments can be applied to widgets that are not associated with a widget-container.

[1020] Referring now to the drawings, FIG. 1 is a schematic block diagram of a widget-container host 100 configured to control, configure, and/or coordinate the sending of widget-containers and/or components of a widget-container to widget-container processing devices 150 based on control information 130, according to an embodiment of the invention. A widget-container is a procedural software framework that can contain (e.g., include a reference to) one or more components such as a widget and/or a service module.

[1021] The widget-container can be executed on various platforms (e.g., mobile phone, personal computer, personal digital assistant (PDA)) and instances of references to the widget-container can be included in and/or spread to a variety of locations such as content aggregation points (e.g., web sites, WAP site) that can be content aggregation points executed (e.g., read) using the widget-container processing devices 150. The widget-container can be configured to include a portion configured for processing one or more widgets when the widget-container is executed at a widget-container processing device 150. The service module (or reference to the service module) contained in the widget-container can be a pre-defined and/or customizable (e.g., user-defined) function related to a variety of functions (e.g., tracking, placing) related to the widget-container and/or its components (e.g., widget). More details related to widget-container spreading and components are set forth in the above-identified co-pending application entitled, “Method and Apparatus for Widget-Container Hosting and Generation.”

[1022] The service module and/or widget can be wrapped in the widget-container, for example, at the time that the widget-container is first defined, after the widget-container has been generated before serving, dynamically when the widget-container is being served, and/or after the widget-container has been served. In some embodiments, the widget-container can be produced dynamically and/or statically for storage, for example, using a hardware and/or software generation engine (not shown) that can be included in the widget-container host 100 and/or a widget-container creation device 120. In some embodiments, the widget-container can be dynamically modified using dynamic injection (i.e., injecting data into the widget-container just before/when the widget-container is served). More details related to dynamic widget-container generation are set forth in the above-identified co-pending application entitled, "Method and Apparatus for Widget-Container Hosting and Generation."

[1023] Each of the widget-container processing devices 150 (also can be referred to as a processing device) can be any type of device that is configured to process the widget-container and a widget and/or service module that can be contained in the widget-container. Each of the widget-container processing devices 150 can be, for example, a computer, a mobile phone, a PDA, a server, and/or any other processor-based device. The widget-container can be configured such that the widget-container can be processed by each of these widget-container processing devices 150 even though the platforms (e.g., hardware, architecture, software, operating system, runtime libraries, programming languages) of the respective widget-container processing devices 150 may be different. The network 140 can be any type of network such as a local area network (LAN) and/or a wide area network (WAN) implemented as a wired and/or wireless network with one or more segments in a variety of environments such as, for example, an office complex.

[1024] The widget-container host 100 is configured to use the control information 130 to determine whether or not to allow any portion associated with a widget-container to be sent to and/or processed at one or more of the widget-container processing devices 150. For example, the widget-container host 100 can receive a request for a widget-container from a widget-container processing device 150. Based on a parameter value/identifier included in the request and based on the control information 130, sending a portion associated with a widget-container that is stored at the widget-container host 100 and/or dynamically generated at the widget-container host 100 can be restricted or allowed. In some embodiments, the parameter value/identifier can be retrieved from, for example, a metadata database (not shown) based on a value included in the request. In some embodiments, for example, the widget-container host 100 can be configured to determine whether or not one or more service modules should be contained in a widget-container based on the control information 130.

[1025] The block diagram also illustrates a widget-server 110 in communication with the widget-container processing devices 150 via network 140. Assuming that sending of a widget is permitted, the widget server 110 can be configured to store and send the widget referenced within a widget-container. For example, in response to a request for the widget, if allowed based on the control information 130, the widget server 110 can send (e.g., serve) the widget to one or more of the widget-container processing devices 150. In some embodiments, the request for the widget can be defined based on a reference to a widget being accessed from an instance of the widget-container at a widget-container processing device 150. The widget can then be sent from the widget server 110 to the widget-container processing device 150 and processed within the instance of the widget-container.

[1026] The widget-container host 100 can be configured to control sending and/or processing of any portion (e.g., widget) of a widget-container by modifying the widget-container in a variety of ways. For example, if processing of a widget within an instance of a widget-container is restricted based on the control information 130, a reference to the widget that is normally included in the widget-container can be removed when the instance of the widget-container is generated and/or served (e.g., sent). In other words, the widget-container can be modified such that processing of the widget is prevented. Because the widget server 110, for example, may not have access to the control information 130 and/or may not be configured to process the control information 130, the sending of the widget from the widget server 110 can be controlled by modifying the widget-container associated with the widget.

[1027] In some embodiments, the widget-container host 100 can modify a widget-container such that a reference to a portion that is typically included in the widget-container is not included in the widget-container when the widget-container is generated or can be removed after the widget-container is generated. In some embodiments, the widget-container host 100 can also include a null reference in place of a valid reference to a portion associated with the widget-container (e.g., widget). This can effectively prevent the portion from being sent and/or processed. The widget-container host 100, in some embodiments, can send an indicator to, for example, a widget-container processing device 150 to trigger the widget-container processing device 150 to stop processing a portion associated with a widget-container. In some embodiments, the widget-container host 100 can send an indicator to, for example, the widget server 110 to prevent the widget server 110 from sending (e.g., block streaming) a widget for processing within a widget-container.

[1028] A control determination can be based on parameters and/or identifiers included in a request for a widget-container that can be, for example, extracted from a reference associated with a widget-container as well as based on control information 130 related to the widget-container stored in a memory (e.g., database). For example, an identifier associated with a placement of a widget-container can be received at the widget-container host 100 in a request for the widget-container and used by the widget-container host 100 with reference to the control information 130 to make a control determination. The request can be defined by any of the widget-container processing devices 150. The widget-container 100 can then generate, modify, and/or send the widget-container based on a result of the control determination.

[1029] The control information 130 can be any type of information that can be used to determine whether or not to allow sending and/or processing of portions of a widget-container associated with one or more of the widget-container processing devices 150. For example, the control information 130 can include or be an instruction that specifies whether or not a particular widget, service module, and/or widget-container should be sent to a particular type of widget-container processing device 150. The control information can include information from a whitelist database (e.g., third party whitelist database, widget-container specific whitelist database) or blacklist database (e.g., third party blacklist database, widget-container specific blacklist database) associated with any portion associated with a widget-container.

[1030] The control information 130 can be stored in, for example, the memory 105 of the widget-container host 100 or a separate database (e.g., metadata database) that can be accessed by the widget-container host 100 via a network such as network 140. The control information 130 can be associated with or include metadata parameters and metadata parameter values (e.g., tracking parameter values) associated with a widget-container and/or a component of the widget-container (e.g., widget or service module). The metadata parameters and values can be used to, for example, dynamically customize or track the widget-container and/or the component of the widget-container. The control determination (e.g., restriction determination, allowance determination) can include a restriction or allowance of a customizable feature of any portion associated with the widget-container. For example, a determination can be based on a tracking history of a particular viewer of any portion associated with a widget-container, a location where any portion associated with a widget-container is viewed, a placement history of any portion associated with a widget-container.

[1031] In some embodiments, any portion associated with the widget-container can be stored and retrieved from the memory 105 of the widget-container host. The memory 105 can be, for example, a database included in the widget-container host 100. In some embodiments, any portion associated with the widget-container can be stored in and accessed from a memory device such as a database that is a remote database and/or a distributed database that can be accessed by the widget-container host 100 via network 140 or via a separate network (not shown). In some embodiments, a widget-container served from the widget-container host 100 can contain a reference to one or more service modules that are served from, for example, a service module server (not shown).

[1032] Also, as shown in FIG. 1, a widget-container creation device 120 can be connected to the container host 100 via network 140. The widget-container creation device 120 can be used to generate, define, and/or modify a widget-container. The widget-container creation device 120 can be, for example, a personal computer, a mobile device (e.g., mobile phone and/or PDA), and/or a server (e.g., back-end server, network server). In some embodiments, the widget-container creation device 120 can be configured to automatically produce instructions used to generate and/or modify widgets containers, for example, when a widget is received. In many embodiments, the widget-container creation device 120 can also store widgets and function as a widget server. Each of the widget-container processing devices 150 can also be configured to function as a widget-container creation device 120 and vice versa.

[1033] The widget-container creation device 120 can trigger the generation and/or modification of the widget-container at the widget-container host 100. In some embodiments, the widget-container creation device 120 can send an instruction from, for example, a processor (not shown) to cause the widget-container host 100 to generate and/or modify a widget-container. In some embodiments, widget-container creation device 120 can trigger the generation and/or modification of the widget-container at the widget-container host 100.

[1034] Although in this embodiment, the widget-container host 100 is a centralized control/storage mechanism that manages requests received from container processing devices 150 and manages the sending of widgets containers (e.g., prompts a separate device (not shown) to send a widget-container stored in a separate database (not shown)), the functionality of the widget-container host 100 can be decentralized and/or distributed. For example, a hierarchy of widget-container hosts 100 can be arranged, for example, with management components and/or slave components that control, configure, and coordinate the sending of widget-containers to the widget-container processing devices 150.

[1035] FIG. 2 is a schematic diagram illustrating a widget-container 250 associated with a content aggregation point 220 being displayed on a display 242 at a widget-container processing device 240, according to an embodiment of the invention. The widget-container server 270 can be configured to determine whether sending/processing of the widget 260 within the widget-container 250 and/or sending/execution of the widget-container 250 are allowed based on control information 295 stored in a control database 290 (also can be referred to as a restriction database or permission database).

[1036] As shown in FIG. 2, the widget-container 250 is served from the widget-container server 270 to the widget-container processing device 240 in response to a widget-container reference 222 being accessed from the content aggregation point 220. The widget-container reference 222 is defined to trigger sending of the widget-container 250 from the widget-container server 270 for display in the content aggregation point 220, for example, when accessed. In some embodiments, the widget-container 250 can be dynamically generated based on an instruction used to define the widget-container 250 or retrieved (e.g., from a memory (not shown)) before the widget-container 250 is served (e.g., sent) to the widget-container processing device 240. The generating or retrieving of the widget-container 250 can be triggered in response to a request for the widget-container 250.

[1037] After the widget-container 250 has been received at the widget-container processing device 240, a widget 260 is served from a widget-server 280 for processing (e.g., execution) within the widget container 250 in response to a widget reference 252 being accessed from the widget-container 250. The widget 260 is served from the widget server 280 in response to a request for the widget 260 defined based on the widget reference 252. In some embodiments, the widget 260 can be requested, received, and/or processed within the widget-container 250 after the widget-container 250 is received at the content aggregation point 220.

[1038] The content aggregation point 220 can be, for example, a web-page, and the widget-container reference 222 can be, for example, an embed tag included in the content aggregation point 220. In some embodiments, the widget-container 250 is a Flash object and the content aggregation point 220 is a Flash-enabled web browser or Flash-enabled web browser page.

[1039] Although FIG. 2 illustrates a single instance of the widget-container 250/widget 260 being sent to content aggregation point 220, multiple instances of the widget-container 250/widget 260 can be synchronously or asynchronously generated (e.g., dynamically generated) and/or distributed to multiple widget-container processing devices and/or to multiple content aggregation points that each include a reference to the widget-container 250.

[1040] The widget-container server 270 can be configured to determine whether processing of the widget 260 within the widget-container 250 and/or sending/execution of the widget-container 250 are allowed based on control information 295. Although, in this embodiment, the control information 295 is stored in and accessed by the widget-container server 270 from a control database 290, in some embodiments, the control information 295 can be stored in a memory (not shown) of the widget-container server 270 or in a set of distributed memory devices (not shown) such as distributed databases. In some embodiments, the control information 295 can be generated and/or collected by the widget-container server 270 when needed. In some embodiments, the control database 290 can be a third-party database that includes control information 295 such as a white-list or a black-list.

[1041] The control information 295 can be defined by the owner of widget 260 and/or the owner/creator of the widget-container 250. Although the widget 260 and the widget-container 250 can be served from two different devices—the widget server 280 and widget-container server 270, respectively—the owner of each can be the same entity. The owner of the widget 260, for example, can define a control rule or control condition based on content (e.g., images, text) associated with content aggregation point 220 (e.g., web-page, WAP page). The owner of widget 260 may restrict (or allow) processing of widget 260 within the content aggregation point 220 because the content of content aggregation point 220 may be viewed as conflicting or incompatible with the content of widget 260.

[1042] In some embodiments, the control information 295 can be based on a content rule associated with a content aggregation point. More details regarding content rules associated with content aggregation points are set forth in co-pending Application Serial No. unassigned, filed on March 6, 2007, and entitled, "Method and Apparatus for Widget-Container Platform Adaptation and Distribution" which is incorporated herein by reference in its entirety.

[1043] The control information 295 can be related to any type of information that can be used to determine whether or not to allow processing of widget 260 within the widget-container 250 and/or whether or not to allow sending of the widget-container 250 to the widget-container processing device 240 for execution within the content aggregation point 220. For example, the control information 295 can be configured such that a control determination (e.g., restriction determination, allowance determination) can be based on an identifier (e.g., uniform resource locator (URL), uniform resource identifier (URI)) associated with the content aggregation point 220 where the widget-container reference 222 is placed. The control information 295 can include, for example, a rule that indicates whether processing of widget 260 in content aggregation point 220 is allowed or restricted.

[1044] In some embodiments, a control determination can be based on a location of the widget-container processing device 240. For example, an indicator (e.g., Internet Protocol (IP) address or portion of an IP address) of the location of the widget-container processing device 240 can be used to determine whether or not to allow sending a portion associated with the widget-container 270 to the widget-container processing device 240.

[1045] In some embodiments, a control determination can be performed in response to the widget-container 250 being placed (e.g., widget-container reference 222 to the widget-container 250 being placed) in content aggregation point 220. The result of the control determination can be stored (e.g., at the widget-container host 100) and associated with a placement identifier uniquely associated with the placement of the widget-container 250 in the content aggregation point 220. If the unique placement identifier is received in a request for an instance of the widget-container 250, the result of the control determination can be immediately implemented without performing the control determination a second time. If the widget-container reference 222 is uniquely associated with the placement of the widget-container 250 in the content aggregation point 220, the widget-container reference 222 can be used as the trigger for implementing the stored control determination result.

[1046] Although not illustrated in this embodiment to simplify the discussion, the widget-container 250 can also contain a service module integrated into the widget-container 250 (e.g., integrated during dynamic generation) or served from, for example, a service module server (not shown). A control determination can also be based on or related to the service module. The service module can be a pre-defined function associated with the widget 260. For example, a service module can be a metadata searching/retrieval function, a polling/categorizing function, a widget-container deployment function (e.g., using a placement service module), a transaction service function (e.g., service module for facilitating a web purchase, service module used for signing a user up for a web service, etc.), a security function (e.g., security firewall function), and/or a widget-container tracking function.

[1047] The widget 260 can be, for example, a static data object, a software object, and/or a media object. Within each of these categories, the widget can be a specific content type. For example, the widget 260 can be an processed (e.g., played) or a displayed media object such as a video file (i.e., a video content file), an mp3 file (i.e., audio content file), or an image file (i.e., image content file). The widget 260 can be a static data object that is, for example, displayed within the widget-container 250. The static data object can be, for example, a textual object such as a set of letters, a textual document, or an html based textual document. The static data object can include, for example, links to other widgets (not shown). The widget 260 can also be a software object such as an executable application or set of instructions that is executed within the procedural software framework of the widget-container 250. The results of the executable application can be included and/or displayed within the procedural software framework of the widget-container 250. The widget-container 250 can be configured so that the widget 260 can be processed within a portion of the widget-container 250 when the widget-container 250 is displayed on the display 242.

[1048] The procedural software framework that makes up the widget-container 250 can be programmed in any number of programming languages that are compiled (e.g., C++) and/or in programming languages that are interpreted (e.g., Java). An executable portion (of the procedural software framework) that manages and controls service module(s) (not shown) and/or the widget 260 referenced (e.g., contained) in the widget-container 250 can be referred to as the core of the widget-container 250. Metadata parameter values can be passed to the widget-container 250 and used by, for example, the core of the widget-container 250 to modify, for example, attributes of the widget-container 250 and/or widget 260 (e.g., x, y location of the widget 260, size of the widget 260, color, themes, styles, etc.).

[1049] FIG. 3 is a table 300 that illustrates a set of control rules (e.g., restriction information) that are based on content aggregation point identifiers 340, according to an embodiment of the invention. The table 300 illustrates that execution of widget-container A (column 310) is allowed in content aggregation points with identifiers URI1 and URI2. Based on an indicator of the content aggregation point identifier 340 received in, for example, a request for widget-container A, table 300 can be used in a control determination to determine that processing of widget W is restricted in content aggregation point URI2, but allowed in content aggregation point URI1.

[1050] The identifier of the content aggregation point 340 can be received at, for example, a widget-container host for use in a control determination at the widget-container host. A request for the widget-container 310, for example, can include the identifier of the content aggregation point 340 or a separate identifier such as a placement identifier 330 that can be used to look-up the identifier of the content aggregation point 340. In some embodiments, the identifier of the content aggregation point 340 can be retrieved from a metadata database (not shown) that can be used to associate a widget-container reference with the identifier of the content aggregation point 340.

[1051] Although, a control determination can be based on one or more content aggregation point identifiers 340, in some embodiments, a control determination can be performed based on a variety of parameters that can be, for example, stored in a metadata parameter database. For example, a control determination can be based on parentage information in a metadata database (not shown) or a placement identifier 330 associated with a particular placement of a reference to a widget container 310 in a content aggregation point 345 associated with the content aggregation point identifiers 340.

[1052] For example, in some embodiments, the placement identifier 330 is included in a reference to a widget-container 310 placed in a content aggregation point 345 and can be used in a control determination. When a reference to widget-container A is accessed from a content aggregation point 345, a request for widget-container A can be defined to indicate the placement identifier 330 uniquely associated with the particular placement of widget-container A and included in the reference. If the widget-container A being requested is the widget-container placed in the content aggregation point X, the request can include the placement identifier P1 from the reference. The placement identifier P1 can then be used to determine based on table 300 that both widget processing and widget-container execution are permitted.

[1053] In some embodiments, the control information in table 300 can be used in conjunction with a default rule that processing of a widget and/or sending/execution of a widget-container is restricted unless allowed. In this situation, the determination can be referred to as an allowance determination. Conversely, a set of control rules can be defined such that processing of a widget and/or sending/execution of a widget-container is allowed unless restricted based on a control rule in a control database. In this situation, the determination can be referred to as a restriction determination.

[1054] Referring back to FIG. 2, widget-container 250 can also include a tracking kernel (not shown) configured to perform a tracking function. The tracking kernel, which can also be referred to as a tracking module, can be embedded/integrated in (e.g., programmed within) the procedural software framework of the widget-container. The tracking kernel, in some embodiments, may not be displayed on the display 242. The tracking kernel can be used to collect tracking parameter values related to user-triggered interactions with the widget-container 250 such as basic mouse movement and clock events.

[1055] The tracking kernel can also be used to collect tracking parameter values related to placement information such as, for example, what web-page (e.g., content aggregation point 220) a widget-container is referenced on and/or placed on, where on the page it was placed, and/or size of the widget-container 250 (e.g., dimensional and memory). This information can be used in a control determination. For example, a tracking parameter value(s) can be used to determine whether a reference to a widget-container will trigger a portion associated with the widget-container to be processed in a portion of a web-page, such as processor readable vehicle 220, that is not allowed based on the control information 295. The tracking parameter value can be used to determine whether the widget-container should be processed on the web-page at all based on the control information 295. Certain user-triggered interactions based on a tracking parameter value can also trigger restriction (or allowance) of a portion associated with a widget-container from being processed and/or executed.

[1056] In some embodiments, a control rule can be a complex set of rules based on one or more variables and/or an algorithm. For example, a control rule can be based on portions of multiple widget-containers placed within a particular content aggregation point that are defined as mutually exclusive portions of widget-containers (e.g., should not be displayed on the same content aggregation point, should not be displayed on a specified content aggregation point). In some embodiments, a control determination based on a complex set of rules can be performed based on tracking parameter values (e.g., parentage information) collected and/or stored in, for example, a metadata database (not shown) or the control database 290.

[1057] In some embodiments, a notification can be sent to an entity associated with a component of the widget-container 250 that indicates that processing of a particular component of the widget-container 250 such as the widget 260 has been terminated. The notification can be generated and sent by, for example, the widget-container server 270. The notification can include statistical data that indicates, for example, one or more content aggregation points, locations, and/or processing devices where the processing of the widget 260 has been terminated or permitted. The statistical data can be based on tracking parameter values and/or parentage information associated with the widget 260 of the widget-container. For example, control results for a portion associated with a widget-container with a particular genealogical lineage can be compiled and sent in a notification. In some embodiments, statistical data based on control information can be stored in, for example, the control database 290 where the statistical data can be retrieved. In some embodiments, historical termination data can be used to update, for example, control information 295 in the control database 290.

[1058] FIG. 4 illustrates a set of category-based control tables 400 and 490 that can be used in a control determination, according to an embodiment of the invention. Table 400 illustrates that each widget 420 of the widget-containers 410 has been associated with a control category 450 and a control indicator 440. Table 400 is a set of rules that indicates whether widget 420 should be processed within a content aggregation point that has been classified as a certain control category 430. For example, table 400 illustrates widget W can be processed in a content aggregation point with a control category 430 of A, but not a control category 430 of B. In some embodiments, table 400 can also be used as a set of rules to determine whether or not widget-containers 410 or service modules (not shown) should be sent to or processed within a content aggregation point that has been classified as a certain control category 430.

[1059] Table 490 illustrates a set of control categories 450 associated with content aggregation points identifiers 460. The control categories 450 can be determined based on, for example, an assessment (e.g., content scan) of each of the content aggregation points where a widget-container 410 has been placed. In some embodiments, a regular (e.g., nightly) or random assessment of content of content aggregation points, for example, can be used to update table 490. An assessment of content can be based on, for example, a scanned image of a content aggregation point. The assessment of the content of a content aggregation point can be used to update control information associated with any portion associated with a widget-container. In some embodiments, content associated with a particular content aggregation point can be assessed in response to a widget-container reference associated with a widget-container being placed in the particular content aggregation point. In other words, placement of a widget-container (e.g., placement of a reference to a widget container) in a content aggregation point can trigger updating of table 490.

[1060] If table 400 is a table stored at, for example, a widget-container server and table 490 is a table stored at a control database, the widget-container server can make a control determination by cross-referencing the control information in table 400 with the control information in table 490 accessed from the control database. The widget-container server can cross-reference the information based on a determination of, for example, a content aggregation point identifier where any of the widget-containers 410 have been placed. For example, if widget-container M has been placed in content aggregation point URI3, a widget-container server can determine using tables 400 and 490 that widget W can be processed within widget-container M at URI3 because content category A is not restricted.

[1061] Default rules can be used in conjunction with tables 400 and 490. For example, if widget-container P were placed in URI3, a default rule may prevent processing of widget Q within the widget-container P at URI3 because an existing rule in table 400 does not cover that situation. Although the tables 400 and 490 were illustrated as two separate blocks of information, in some embodiments, the tables 400 and 490 can be combined or portions of either of the tables 400 and/or 490 included as separate blocks of information in, for example, separate databases.

[1062] In some embodiments, control categories can be, for example, location-related control categories. For example, a portion associated with a widget-container may be permitted to be processed (or prevented from processing) in only certain locations based on a control category within a control rule. The location control can be defined based on, for example, an Internet Protocol (IP) address associated with a request for the widget-container.

[1063] Referring back to FIG. 2, the widget-container server 270 can be configured to make a control determination with respect to the widget 260 and/or widget-container 250 after the widget-container 250 is served (e.g., sent). For example, when the widget-container server 270 determines, after the widget-container 250 has been sent to the widget-container processing device 240, that sending/execution of the widget-container 250 and/or processing of the widget 260 is restricted (or allowed), the widget-container server 270 can send an indicator to the widget-container processing device 240 to stop (or start/continue/allow) execution of the widget-container 250 within the content aggregation point 220 or processing of the widget 260 within the widget-container 250. The widget-container server 270 can passively permit processing if processing of the widget 260 within the widget-container 250 has already commenced and is allowed.

[1064] FIG. 5 is a flowchart that illustrates a control determination performed after a widget-container has been sent to a widget-container processing device, according to an embodiment of the invention. The flowchart illustrates that a widget-container is produced (e.g., generated) with a valid reference to a component of the widget-container 500. The valid reference can be a reference to a component of the widget-container such as a widget and/or a reference to a service module associated with the widget. The widget-container can be produced in response to a request for the widget-container defined in response to a reference to the widget-container being accessed from a content aggregation point (e.g., web-page, WAP page) executed/displayed at the widget-container processing device. The widget-container produced at 500 can be an instance that is, for example, generated dynamically with customized features for the content aggregation point based on a metadata parameter value. After the widget-container has been produced, the widget-container is sent to the widget-container processing device such as a cell phone at 510.

[1065] After the widget-container has been sent, a control determination (e.g., restriction determination, allowance determination) is performed at 520 to determine whether processing/execution of the component of the widget-container is restricted or allowed. The control determination can be based on control information that is retrieved from a memory such as a control database. The control determination can be based on, for example, an identifier associated with the content aggregation point where the widget-container is to be executed and/or displayed. The control determination can be based on the content of the content aggregation point or the physical location where the content aggregation point is being accessed. The control determination can also be performed based on a credential or location identifier associated with an entity that is requesting the widget-container.

[1066] In response to a control determination being made at 520, an indicator to terminate processing/execution of the component of the widget-container at the widget-container processing device is sent at 530. The indicator can be sent to, for example, a widget server to cause the widget server to stop streaming a video widget that is being processed within the widget-container. The indicator can be sent directly to the widget-container processing device and can be configured to cause the widget-container processing device to stop processing/executing the component of the widget-container.

[1067] At 540, a notification to an entity associated with the component of the widget-container that processing/execution of that component has been terminated. The notification can include statistical data that indicates, for example, other content aggregation points, locations, and/or processing devices where the processing/execution of the component has been terminated. The statistical data can be based on tracking and/or parentage information associated with any portion associated with the widget-container. In some embodiments, historical termination data can be used to update, for example, a control database.

[1068] Although the flowchart shown in FIG. 5 is directed towards components of widget-containers, in some embodiments, the flowchart can be modified to apply to any portion associated with the widget-container including the entire widget-container itself. In some embodiments, the flowchart can be modified such that the blocks are in any order. For example, a notification can be sent to an entity before an indicator to cause termination is generated and/or sent.

[1069] Referring back to FIG. 2, the widget-container server 270 can be configured to make a control determination with respect to the widget 260 and/or widget-container 250 before the widget-container 250 is served (e.g., sent). For example, when the widget-container server 270 determines that sending of the widget-container 250 is restricted (or allowed) before the widget-container 250 has been sent, the widget-container server 270 can be configured to deny (or grant) a request for the widget-container 250 and/or can halt (or continue) processing with respect to the widget-container 250.

[1070] If the widget-container server 270 determines, before the widget-container 250 has been sent, that processing of the widget 260 is restricted, the widget-container server 270 can still send the widget-container 270. However, the widget-container 270 can be modified, for example, by removing the widget reference 252, including a null (e.g., empty) reference (not shown), and/or a re-directing reference (e.g., to a notice that the widget 260 will not be processed) to prevent a valid request for the widget 260 from being defined and sent to the widget server 280. These techniques will effectively prevent the widget 260 from being sent to and processed in the widget-container 250 at the widget-container processing device 240.

[1071] In contrast, if the widget-container server 270 determines that processing of the widget 260 within the widget-container 250 is allowed before the widget-container 250 has been sent, the widget-container server 270 can send the widget-container 250 with a widget reference 252 that is valid. The valid widget reference 252 or null widget reference 252 can be dynamically injected into the widget-container 250 when the widget-container 250 is generated, for example, based on an instruction in response to a request for the widget-container 250. Likewise, a valid widget reference 252 can be removed from the widget-container 250 when the widget-container 250 is dynamically generated.

[1072] Whether a control determination is made before or after the widget-container 250 is served, the widget-container server 270 can send an indicator to the widget server 280 to cause the widget server 280 to prevent sending or halt (e.g., block) sending, if necessary, of the widget 260 to the widget-container processing device 240. The widget-container server 270 can cause the widget server 280 to halt sending if sending and/or processing of the widget 260 within the widget-container 250 has already commenced.

[1073] In some embodiments, the widget-container processing device 240 and/or widget-container 250 can be configured to wait for an indicator from the widget-container server 270 before processing the widget 260. Likewise, in some embodiments, the widget server 260 can be configured to wait for an indicator from the widget-container server 270 before sending the widget 260 for processing within the widget-container 250. In other words, the default mode of the widget-container processing device 240, widget server 260, and/or widget-container 250 can be to process and/or send the widget 260 when triggered by the widget-container server 270.

[1074] FIG. 6 is a flowchart that illustrates a method for restricting or allowing processing of a component associated with a widget-container before sending the widget-container for display in a content aggregation point, according to an embodiment of the invention. The flowchart illustrates that an instruction defining a widget-container associated with a widget is received at 600. The instruction can be used to dynamically produce the widget-container when a request for the widget-container is received.

[1075] A reference uniquely defined to trigger sending of an instance of the widget-container for execution within a content aggregation point is received at 610. In some embodiments, the reference can be received and stored in a database and associated with customizable parameter values (e.g., metadata parameter values) associated with the placement of the reference to the widget-container in the content aggregation point so that the customizable parameters values can be used when producing the widget-container for sending to the content aggregation point.

[1076] An indicator that the reference to the widget-container has been accessed at the content aggregation point is received at 620, an identifier associated with the content aggregation point is received at 630 in response to the reference being accessed, and a control determination is performed at 640 to determine whether processing of any component (include the widget referenced above) of the widget-container is restricted or allowed. For example, the control determination can be based on control information that is retrieved from a memory such as a control database. The control determination can be based on, for example, an identifier associated with the content aggregation point where the widget-container is to be executed and/or displayed.

[1077] If the processing of the component of the widget-container is allowed, an instance of the widget-container is produced with a valid reference to the component of the widget-container at 650 based on the instruction. In other words, the widget-container is produced without modifying the instruction or the widget-container with respect to the reference to the component. The instance of the widget-container with the valid reference is then sent to the widget-container processing device that requested the widget-container at 670. In some embodiments, if the widget-container is a static object stored in a memory (e.g., not dynamically generated based on the instruction), the widget-container can be retrieved from the memory and sent to the widget-container processing device. In other words, the widget-container will not be modified.

[1078] If the processing of a component of the widget-container is restricted, the widget-container can be modified and/or produced at 660 based on the control determination. The widget-container can be produced based on the instruction with some modification to address the control determination. For example, the widget-container can be produced with a null reference to the component of the widget-container (or a re-directing reference) such that the component of the widget-container is not requested from, for example, a server of the component when the widget-container is executed a widget-container processing device. The modified instance of the widget-container is then sent to the widget-container processing device that requested the widget-container at 670.

[1079] In some embodiments, if the widget-container is a static object stored in a memory (e.g., not dynamically generated based on the instruction), the widget-container can be retrieved from the memory and modified to, for example, remove the reference to the component before the widget-container is sent to the widget-container processing device.

[1080] Some embodiments of the invention relate to a storage product with a processor-readable medium having instructions or code thereon for performing various processor-implemented operations. The media and code may be those specially designed and constructed for the purposes of the invention, or they may be of the kind well known and available to those having skill in the software arts. Examples of processor-readable media include, but are not limited to: magnetic storage media such as hard disks, floppy disks, and magnetic tape; optical storage media such as Compact Disc/Digital Video Discs ("CD/DVDs"), Compact Disc-Read Only Memories ("CD-ROMs"), and holographic devices; magneto-optical storage media such as floppy or optical disks; carrier wave signals; and hardware devices that are specially configured to store and execute program code, such as Application-Specific Integrated Circuits ("ASICs"), Programmable Logic Devices ("PLDs"), and ROM and RAM devices. Examples of code include, but are not limited to, micro-code or micro-instructions, machine instructions, such as produced by a compiler, and files containing higher-level instructions that are executed by a processor using an interpreter. For example, an embodiment of the invention may be implemented using Java, C++, or other object-oriented programming language and development tools. Additional examples of code include, but are not limited to, control signals, encrypted code, and compressed code.

[1081] In conclusion, the present invention is related to an apparatus and methods for controlling processing of and/or distribution of static, media, and/or software objects in a procedural software framework. Those skilled in the art can readily recognize that numerous variations and substitutions may be made in the invention, its use and its configuration to achieve substantially the same results as achieved by the embodiments described herein. Many variations, modifications and alternative constructions fall within the scope and spirit of the disclosed invention as expressed in the claims.

What is claimed is:

1. A method, comprising:

receiving an instruction used to define a widget-container containing a service module associated with a widget, the widget-container being a procedural software framework configured to be executed at a widget-container processing device in response to a reference to the widget-container being accessed from a content aggregation point, the widget-container having a portion configured for receiving the widget; and

determining whether processing of the widget within the portion of the widget-container is at least one of restricted or allowed.

2. The method of claim 1, further comprising:

producing the widget-container based on the instruction, the producing includes incorporating a valid reference to the widget in the widget-container in response to the processing of the widget being allowed; and

sending the widget-container to the widget-container processing device in response to the reference to the widget-container being accessed.

3. The method of claim 2, wherein the producing is performed at a first device, the widget-container processing device is configured to receive the widget from a second device when processing the widget within the widget-container.

4. The method of claim 1, wherein the service module is a pre-defined function associated with the widget-container, the widget is at least one of a static data object, a media object, or a software object.

5. The method of claim 1, further comprising:

producing the widget-container based on the instruction, the producing includes incorporating at a first time a reference to the widget in the widget-container;

sending at a second time the widget-container to a widget-container processing device in response to the reference to the widget-container being accessed, the second time being after the first time, the determining includes determining at a third time that processing of the widget is at least one of restricted or allowed, the third time being after the second time; and sending, in response to the determining, to the widget-container processing device an indicator configured to stop processing of the widget within the widget-container.

6. The method of claim 5, wherein the determining includes determining before the reference is accessed.

7. The method of claim 1, wherein the content aggregation point is a first content aggregation point, the reference is a first reference derived from a second reference to the widget-container included in a second content aggregation point, the determining includes determining after the first reference is included in the first content aggregation point.

8. The method of claim 1, wherein the determining includes determining based on an identifier associated with the content aggregation point in response to the reference being placed in the processor readable vehicle.

9. The method of claim 1, further comprising:

modifying the widget-container in response to the processing of the widget being at least one of restricted or allowed.

10. The method of claim 1, wherein the determining includes determining based on at least one of a geographic location associated with the widget-container processing device, a stored image of the content aggregation point, content included in the content aggregation point, or a preference from a control database.

11. A processor-readable medium comprising code representing instructions to cause a processor to:

receive a first reference to a widget-container placed in a first content aggregation point, the first reference being derived from a second reference to the widget-container placed in a second content aggregation point, the widget-container being a procedural software framework configured to be executed at a widget-container processing device in response to the first reference to the widget-container being accessed at the widget-container processor device, the widget-container having a portion configured for receiving a component; and determine whether processing of the component within the widget-container is at least one of restricted or allowed in response to the first reference to the widget-container being accessed.

12. The processor-readable medium of claim 11, wherein the first reference is uniquely defined to trigger sending of the widget-container for execution within a location of the first content aggregation point.

13. The processor-readable medium of claim 11, further comprising instructions to cause a processor to:

retrieve an identifier associated with the content aggregation point in response to the first reference being accessed at the widget-container processing device, the code to determine including code to determine based on the identifier.

14. The processor-readable medium of claim 11, further comprising instructions to cause a processor to:

generate the widget-container based on an instruction defining the widget-container in response to the first reference being accessed;

incorporate at a first time a valid reference to the component in the widget-container in response to an indicator that processing of the component is allowed; and

trigger at a second time sending the widget-container to the widget-container processing device, the second time being after the first time.

15. The processor-readable medium of claim 11, wherein the determine includes code to determine based on an identifier associated with the first content aggregation point in response to the first reference being placed in the first processor readable vehicle.

16. The processor-readable medium of claim 11, further comprising instructions to cause a processor to:

trigger sending a notification to an entity associated with the component, the notification including an indicator of whether processing of the component is at least one of restricted or allowed.

17. A method, comprising:

receiving a request for a widget-container from a widget-container processing device in response to a reference to the widget-container being accessed from a content aggregation point at the widget-container processing device;

determining in response to the request whether processing of a portion associated with the widget-container is allowed based on a preference associated with the portion associated with the widget-container and an identifier associated with the content aggregation point; and

sending in response to the request the widget-container to the widget-container processing device.

18. The method of claim 17, wherein the sending includes sending after the determining.

19. The method of claim 17, wherein the determining includes determining after the sending.

20. The method of claim 17, wherein the widget-container is a procedural software framework configured to be executed at the widget-container processing device,

the method further comprising:

modifying the procedural software framework of the widget-container before the sending and in response to the determining.

21. The method of claim 17, wherein the preference is associated at a first time with the portion associated with the widget-container, the reference to the widget-container is associated with the content aggregation point at a second time, the second time being after the first time, the determining includes determining after the second time.

22. The method of claim 17, wherein the reference is uniquely defined to trigger sending of the widget-container for execution within a location of the first content aggregation point, the reference is from a plurality of references to the widget-container, each reference from the plurality of references is derived from a separate reference from the plurality of references.

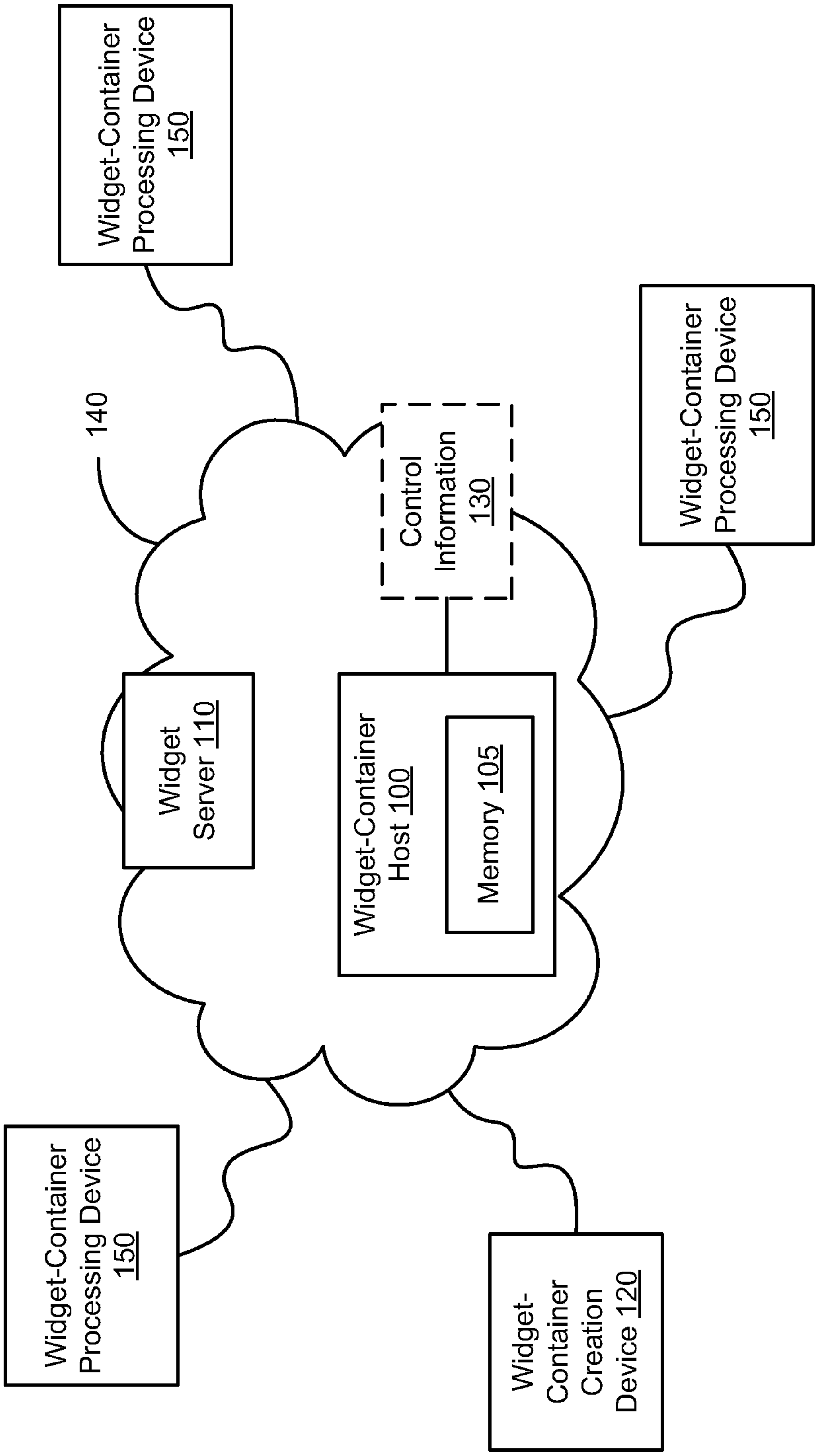

Widget-Container Processing Device 150
140
Control Information 130
Widget-Container Processing Device 150
Widget Server 110
Widget-Container Host 100
Memory 105
Widget-Container Processing Device 150
Widget-Container Creation Device 120

FIG. 1

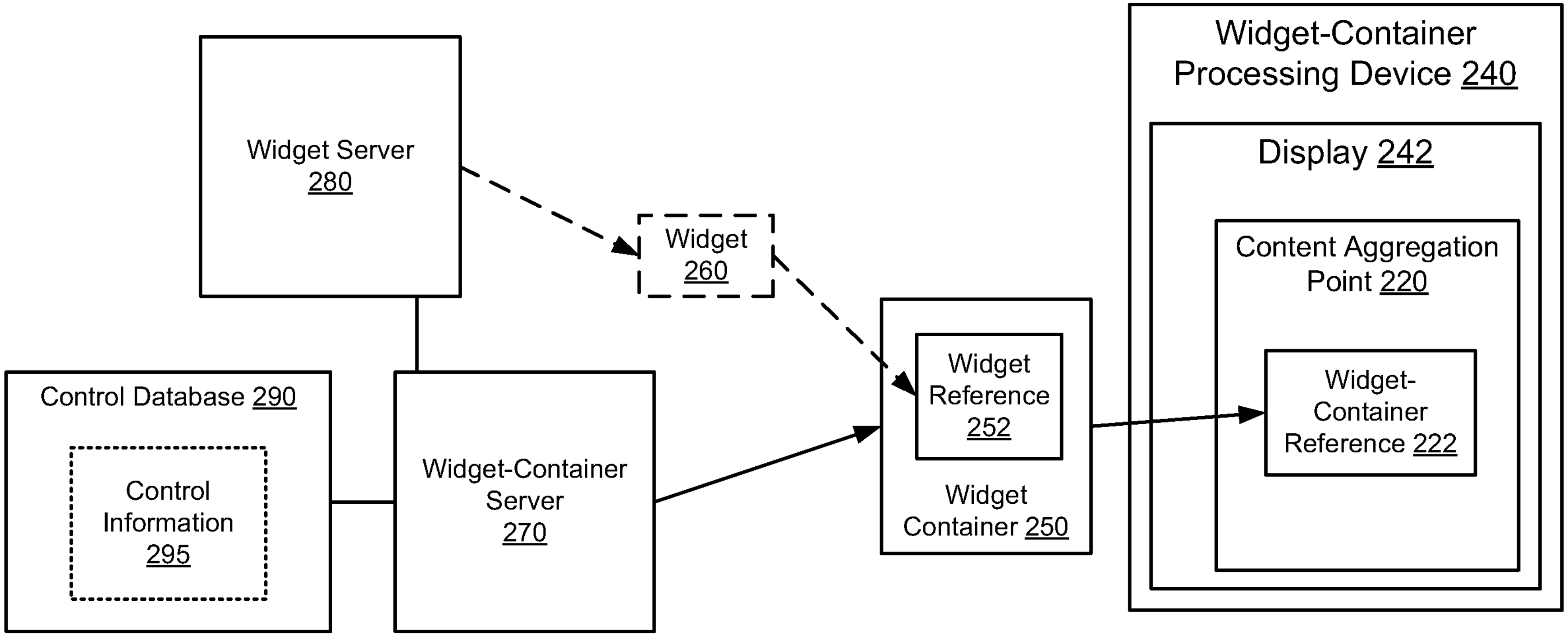
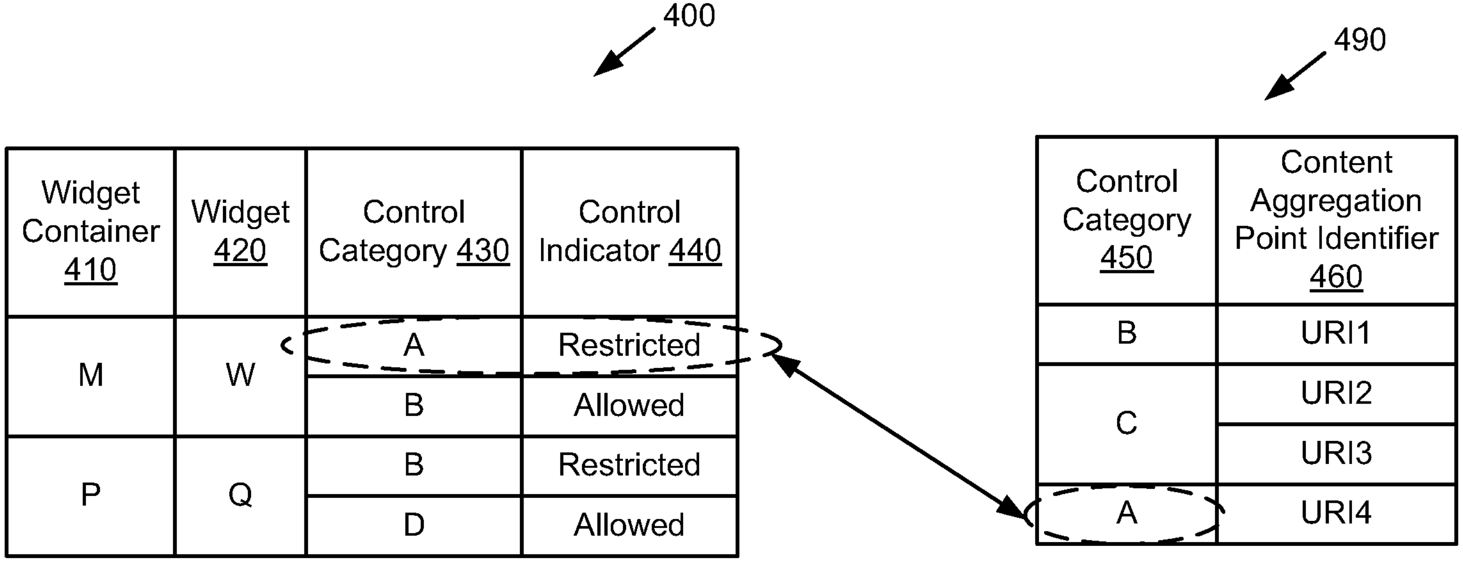

Widget Server 280
Widget 260
Widget-Container Processing Device 240
Display 242
Content Aggregation Point 220
Widget-Container Reference 222
Widget Reference 252
Widget Container 250
Control Database 290
Control Information 295
Widget-Container Server 270

FIG. 2

300

| Widget Container 310 | Widget 320 | Placement identifier 330 | Content Aggregation Point 345 | Content Aggregation Point Identifier 340 | Widget Rendering Allowed 350 | Widget-Container Allowed 360 |
|---|---|---|---|---|---|---|
| A | W | P1 | X | URI1 | Yes | Yes |
|  |  | P2 | Y | URI2 | No | Yes |

FIG. 3



| Widget Container 410 | Widget 420 | Control Category 430 | Control Indicator 440 |
|---|---|---|---|
| M | W | A | Restricted |
| | | B | Allowed |
| P | Q | B | Restricted |
| | | D | Allowed |

| Control Category 450 | Content Aggregation Point Identifier 460 |
|---|---|
| B | URI1 |
| C | URI2 |
| | URI3 |
| A | URI4 |

FIG. 4

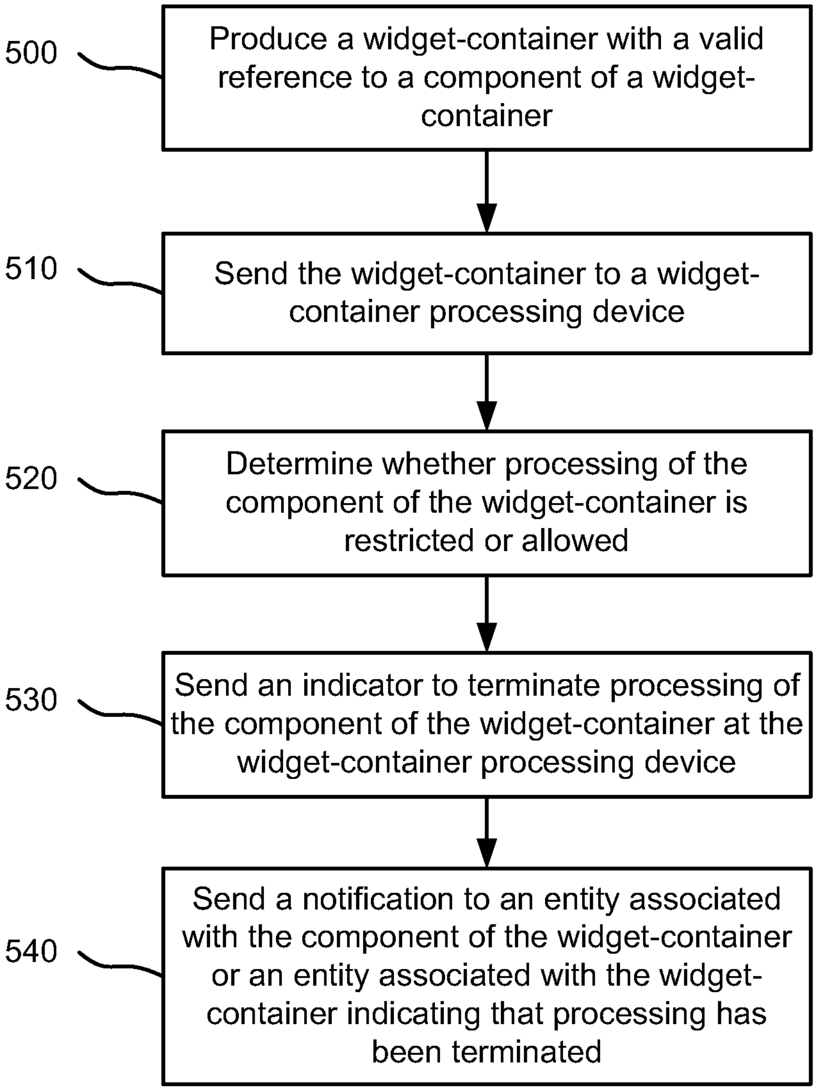

500
Produce a widget-container with a valid reference to a component of a widget-container
510
Send the widget-container to a widget-container processing device
520
Determine whether processing of the component of the widget-container is restricted or allowed
530
Send an indicator to terminate processing of the component of the widget-container at the widget-container processing device
540
Send a notification to an entity associated with the component of the widget-container or an entity associated with the widget-container indicating that processing has been terminated

FIG. 5

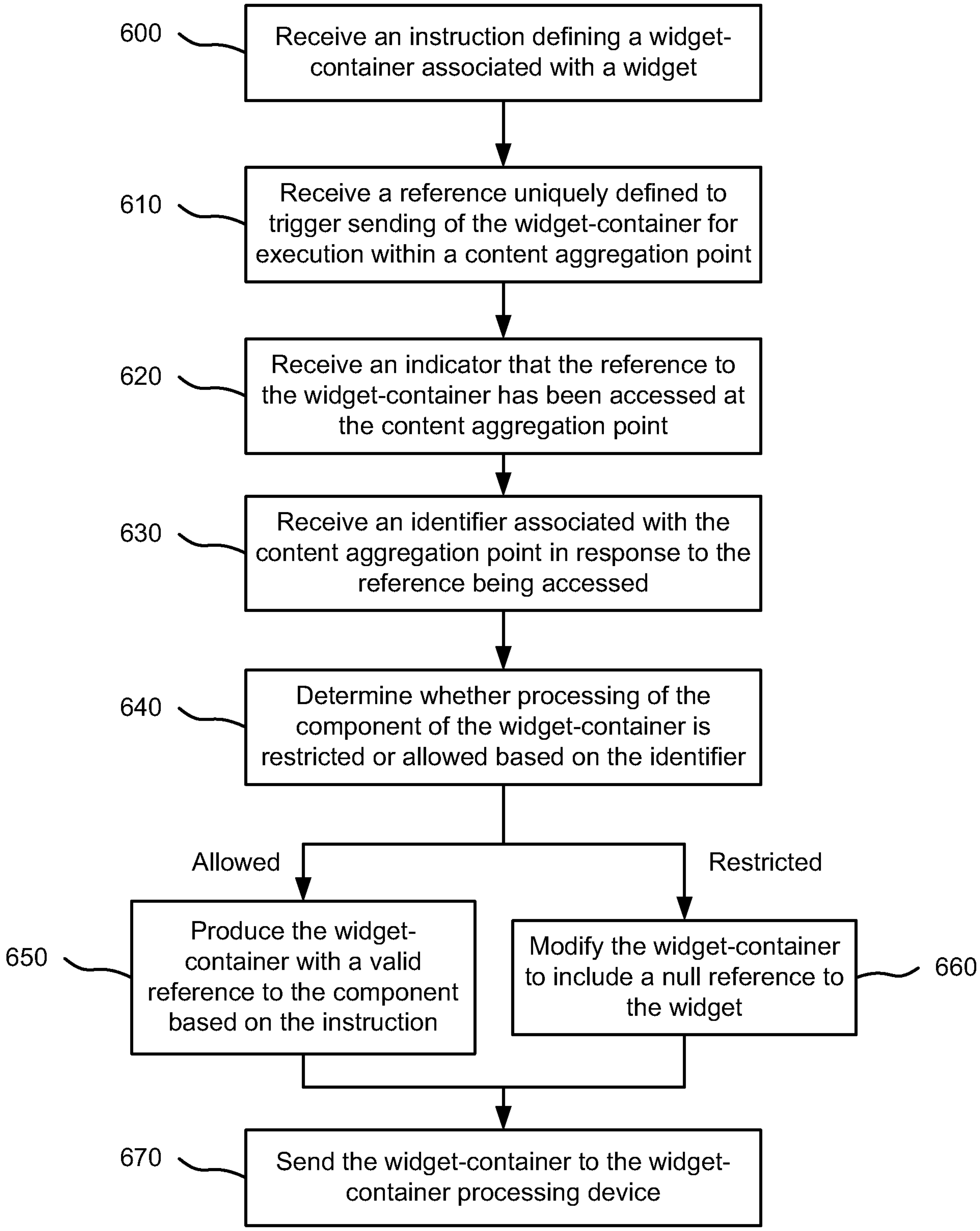

600
Receive an instruction defining a widget-container associated with a widget
610
Receive a reference uniquely defined to trigger sending of the widget-container for execution within a content aggregation point
620
Receive an indicator that the reference to the widget-container has been accessed at the content aggregation point
630
Receive an identifier associated with the content aggregation point in response to the reference being accessed
640
Determine whether processing of the component of the widget-container is restricted or allowed based on the identifier
Allowed
Restricted
650
Produce the widget-container with a valid reference to the component based on the instruction
660
Modify the widget-container to include a null reference to the widget
670
Send the widget-container to the widget-container processing device

FIG. 6

# INTERNATIONAL SEARCH REPORT

International application No.
PCT/US 08/55756

A. CLASSIFICATION OF SUBJECT MATTER
IPC(8) - G06F 9/44 (2008.04)
USPC - 717/108
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
USPC: 717/108

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
USPC: 707/100-104.1; 717/106, 108, 116; 709/203, 223 (view text search terms below)

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
pubWEST(PGPB,USPT,EPAB,JPAB; PLUR=YES); DialogPRO(Engineering); Google Scholar; Text search terms: interoperat, widget, container, content, aggregat, point, modif, content, data, compat, point, device, server, PC, laptop, PDA, mobile, phone, transform, module, compl, rule, trigger, event, cause, initiate, launch, alert, alarm

C. DOCUMENTS CONSIDERED TO BE RELEVANT

| Category* | Citation of document, with indication, where appropriate, of the relevant passages | Relevant to claim No. |
|---|---|---|
| X | US 2004/0107125 A1 (GUHEEN, et al.) 03 June 2004 (03.06.2004) entire document, especially Abstract, FIGS.15, 17 and para [0114], [0141], [0149]-[0150], [0174]-[0175], [0393], [0416], [0442], [0602], [0610]-[0611], [0784], [0808], [0874], [1865], [1869], [2004], [2189], [2330], [2399], [2495], [2920], [3046], [3117], [3260], [3346] and [3949] | 1-22 |
| A | US 2006/0178918 A1 (MIKURAK) 10 August 2006 (10.08.2006) entire document | 1-22 |

□ Further documents are listed in the continuation of Box C. □

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search
28 July 2008 (28.07.2008)

Date of mailing of the international search report
01 AUG 2008

Name and mailing address of the ISA/US
Mail Stop PCT, Attn: ISA/US, Commissioner for Patents
P.O. Box 1450, Alexandria, Virginia 22313-1450
Facsimile No. 571-273-3201

Authorized officer:
Lee W. Young
PCT Helpdesk: 571-272-4300
PCT OSP: 571-272-7774