

MIXED VIEW DISPLAY METHOD AND DEVICE

[0001] The present application claims priority to Chinese Patent Application No. 201610172774.1, filed on March 24, 2016 and entitled "MIXED VIEW DISPLAY METHOD AND DEVICE", which is incorporated here by reference in its entirety.

5

TECHNICAL FIELD

[0002] The present disclosure relates to the field of webpage processing technologies, and in particular, to a mixed view display method and device.

BACKGROUND

[0003] With the rapid development of information technologies, people have increasingly high requirements for an application (App) on a mobile device. Due to limitation of the mobile device, the App on the mobile device changes from a native App to a hybrid App and then to a web-based App (web App) in response to technology update and market demand. The hybrid App means an App between the web App and the native App. A webpage view and a client view of a certain App can be both displayed by using the hybrid App. The webpage view is a Hypertext Markup Language version 5 (HTML5) page running on a browser. The client view is a page or a control for display that is written by using the iOS/Android development language. For example, a webpage view and a client view of the App Taobao can be displayed in a mixed way by using the hybrid App.

20 [0004] Currently, a subview can be added to the WebView (a client webpage container) of a mobile device to implement the hybrid App. To be specific, a webpage view and a client view of an App are displayed in a mixed way by adding the client view to the WebView. For example, the client view is added to webView.scrollView in the iOS system, or the client view is added to webView.addView in the Android system, to directly add the client view to the WebView. However, the webpage view relies on a browser engine for rendering, and the client view relies on compilation of an operating system for running. Because of the two different environments, a mixed

view is displayed inconsistently when a scroll change is made to the webpage view.

SUMMARY

5 [0005] In view of the earlier described problem, the present disclosure is proposed to provide a mixed view display method and device for resolving the problem or at least partially resolving the problem.

[0006] To achieve the objective, the present disclosure provides the following technical solutions:

10 [0007] According to one aspect, an implementation of the present disclosure provides a mixed view display method, and the method includes the following: loading a webpage view when a webpage view loading instruction is received, where the webpage view includes a placeholder element that is used to add a client view; obtaining location information of the placeholder element from a result of loading the webpage view; adding the client view to the location information; and displaying the webpage view that includes the added client view.

15 [0008] According to another aspect, an implementation of the present disclosure further provides a mixed view display device, and the device includes the following: a loading unit, configured to load a webpage view when a webpage view loading instruction is received, where the webpage view includes a placeholder element that is used to add a client view; an acquisition unit, configured to obtain location
20 information of the placeholder element from a result of loading the webpage view; an adding unit, configured to add the client view to the location information; and a display unit, configured to display the webpage view that includes the added client view.

25 [0009] The technical solutions provided in the implementations of the present disclosure have at least the following advantages:

[0010] The implementations of the present disclosure provide the mixed view display method and device. First, the webpage view is loaded when the webpage view loading instruction is received, where the webpage view includes the placeholder element that is used to add the client view. Then, the location information of the
30 placeholder element is obtained from the result of loading the webpage view, and the client view is added to the location information. Eventually, the webpage view that includes the added client view is displayed. In comparison to a subview being directly

added to a WebView to display a mixed view, in the implementations of the present disclosure, the location information of the placeholder element is first obtained from the result of loading the webpage view when the webpage view loading instruction is received. Then, the client view is added to a location of the placeholder element to
5 display the webpage view and the client view in a mixed way. The placeholder element in the implementations of the present disclosure can be rendered by using a browser engine. Therefore, the client view can scroll with the webpage view by adding the client view to the location corresponding to the placeholder element after the loading and can improve accuracy and consistency of displaying a mixed view.

10

BRIEF DESCRIPTION OF DRAWINGS

[0011] Various other advantages and benefits are clearly understood by a person of ordinary skill in the art by reading the detailed descriptions of the following implementations. The accompanying drawings are merely used to illustrate the implementations, and are not used to limit the present disclosure. In addition, the
15 same reference symbols are used to represent the same components in the drawings. In the accompanying drawings:

[0012] FIG. 1 is a flowchart illustrating a mixed view display method, according to an implementation of the present disclosure;

[0013] FIG. 2 is a flowchart illustrating another mixed view display method,
20 according to an implementation of the present disclosure;

[0014] FIG. 3 is a composition block diagram illustrating a mixed view display device, according to an implementation of the present disclosure;

[0015] FIG. 4 is a composition block diagram illustrating another mixed view display device, according to an implementation of the present disclosure; and

25 [0016] FIG. 5 is a schematic diagram illustrating an interface of a mixed view, according to an implementation of the present disclosure.

DESCRIPTION OF IMPLEMENTATIONS

[0017] The following describes example implementations of the present disclosure in detail with reference to the accompanying drawings. Although the
30 accompanying drawings show the example implementations of the present disclosure,

it should be understood that the present disclosure can be implemented in various forms, and shall not be limited by the implementations described here. Instead, these implementations are used to provide a more thorough understanding of the present disclosure and to comprehensively convey the scope of the present disclosure to a person skilled in the art.

[0018] The present disclosure is described in detail in the following with reference to the accompanying drawings and implementations, to make advantages of the technical solutions of the present disclosure clearer.

[0019] An implementation of the present disclosure provides a mixed view display method. As shown in FIG. 1, the method includes the following steps:

[0020] 101. Load a webpage view when a webpage view loading instruction is received.

[0021] The webpage view includes a placeholder element that is used to add a client view. In the present implementation of the present disclosure, the webpage view loading instruction can be triggered by opening a specific page of an App, for example, selecting a repayment button of the Credit Pay in the Alipay or a purchase reservation button in the ZhaoCaiBao. It is not limited in the present implementation of the present disclosure.

[0022] It is worthwhile to note that the placeholder element included in the webpage view is an element that can change a size or a width/height of an object. In the present implementation of the present disclosure, the placeholder element can be a block-level element, or the placeholder element can be represented as a row-level element of a block-level element. It is not limited in the present implementation of the present disclosure. For example, the placeholder element is represented by using a block-level element `<div id="native-view"></div>`, or the placeholder element is represented by using a row-level element of a block-level element `<span id="native-view" style="display:block"></span>`. Because `style="display:block"` that is set in the row-level element means to change an object to a block-level element, the row-level element is actually represented as a block-level element, and the row-level element is used to change a corresponding size or width.

[0023] 102. Obtain location information of a placeholder element from a result of loading the webpage view.

[0024] In the present implementation of the present disclosure, because the webpage view is an HTML page, the to-be-displayed webpage view is loaded for

display by using a browser rendering engine. It is worthwhile to note that loading of different webpage views is internally monitored by different operating systems of terminal devices in the present disclosure. For example, whether the webpage view has been loaded is monitored by using the interface `webViewDidFinishLoad` (iOS) in the iOS (an operating system in an Apple mobile device); and whether the webpage view has been loaded is monitored by using the interface `onPageFinished` (Android) in the Android (Android operating system).

[0025] In the present implementation of the present disclosure, the `WebView` (a client webpage view container) executes the `JavaScript` (a scripting language) to obtain the location information of the placeholder element from the result of loading the to-be-displayed webpage view. There are different methods for obtaining a placeholder element based on different methods for marking a placeholder element. For example, script code for obtaining location information of a placeholder element is as follows:

```
15      (function(){
          var ele = document.getElementById('native-view');
          var xPoint = ele.getBoundingClientRect().left;
          var yPoint = ele.getBoundingClientRect().top + document.body.scrollTop;
          return JSON.stringify({x: xPoint, y: yPoint});
20      })();
```

[0026] In the code, 'native-view' indicates an identifier of a placeholder element, `xPoint` indicates left location distance information of the placeholder element relative to a browser window, and `yPoint` is used to obtain an upper location of the placeholder element relative to the browser window on a page. A location of the placeholder element can be determined from the result of loading the webpage view based on `x` and `y` in a coordinate axis.

[0027] 103. Add a client view to the location information.

[0028] The client view is a native view control that is written and compiled by using a corresponding operating system language of a mobile phone, for example, the iOS `UISwitch` (a switch control), the Android `TextView` (a text display), or a custom view control written by a user.

[0029] 104. Display the webpage view that includes the added client view.

[0030] In the present implementation of the present disclosure, first, the location information of the placeholder element is obtained from the result of loading the

webpage view when the webpage view loading instruction is received; then, the client view is correspondingly added to a location of the placeholder element, to display the webpage view and the client view in a mixed way. The placeholder element in the present implementation of the present disclosure can be rendered by using a browser engine. Therefore, the client view can scroll with the webpage view by adding the client view to the location corresponding to the placeholder element after the loading. The present disclosure can improve accuracy and consistency of displaying a mixed view.

[0031] The present implementation of the present disclosure provides the mixed view display method. First, the webpage view is loaded when the webpage view loading instruction is received, where the webpage view includes the placeholder element that is used to add the client view. Then, the location information of the placeholder element is obtained from the result of loading the webpage view, and the client view is added to the location information. Eventually, the webpage view that includes the added client view is displayed. In comparison to a subview being directly added to a WebView to display a mixed view, in the implementations of the present disclosure, the location information of the placeholder element is first obtained from the result of loading the webpage view when the webpage view loading instruction is received. Then, the client view is added to a location of the placeholder element to display the webpage view and the client view in a mixed way. The placeholder element in the implementations of the present disclosure can be rendered by using a browser engine. Therefore, the client view can scroll with the webpage view by adding the client view to the location corresponding to the placeholder element after the loading. The present disclosure can improve accuracy and consistency of displaying a mixed view.

[0032] An implementation of the present disclosure provides another mixed view display method. As shown in FIG. 2, the method includes the following steps:

[0033] 201. Add a placeholder element to a webpage view.

[0034] Corresponding identification information is set for each placeholder element, and the identification information is used to uniquely identify the placeholder element. The identification information corresponding to each placeholder element can be set by using class or id. For example, in a placeholder element represented by using a block-level element `<div class="native-view1"></div>`, identification information of the placeholder element is

set by using class, that is, the identification information of the placeholder element is native-view1. In a placeholder element represented by using a row-level element of a block-level element `<span id = "native-view2" style="display: block"></span>`, identification information of the placeholder element is set by using id, that is, the
5 identification information of the placeholder element is native-view2.

[0035] It is worthwhile to note that the placeholder element included in the webpage view is an element that can change a size or a width/height of an object. In the present implementation of the present disclosure, the placeholder element can be a block-level element, or the placeholder element can be represented as a row-level
10 element of a block-level element. It is not limited in the present implementation of the present disclosure. In the present implementation of the present disclosure, the placeholder element is set to a block-level element, or the placeholder element is represented as a row-level element of a block-level element. A width/height of the placeholder element can be set in a later step. To be specific, a size of a client view
15 can be adjusted by using the placeholder element. Therefore, the present implementation of the present disclosure can improve accuracy and consistency of displaying a client view.

[0036] 202. Set a client view corresponding to each placeholder element.

[0037] The client view is a native view control that is written and compiled by
20 using a corresponding operating system language of a mobile phone, for example, the iOS UISwitch (a switch control), the Android TextView (a text display), or a custom view control written by a user. It is not limited in the present implementation of the present disclosure. It is worthwhile to note that, because the webpage view may include multiple client views and one client view can be added to each placeholder
25 element, client views corresponding to the placeholder elements further need to be set after the multiple placeholder elements are added to the webpage view. In the present implementation of the present disclosure, the client views corresponding to the placeholder elements are set so that the multiple client views can be added to location information corresponding to the placeholder elements in a later step.

[0038] 203. Load the webpage view when a webpage view loading instruction is
30 received.

[0039] The webpage view includes the placeholder element that is used to add the client view. In the present implementation of the present disclosure, the webpage view loading instruction can be triggered by opening a specific page of an App, for

example, selecting a repayment button of the Credit Pay in the Alipay or a purchase reservation button in the ZhaoCaiBao. It is not limited in the present implementation of the present disclosure.

5 **[0040]** 204. Determine whether location information of the placeholder element can be obtained from a result of loading the webpage view.

10 **[0041]** It is worthwhile to note that webpage view loading is internally listened in different operating systems of terminal devices in the present disclosure. For example, whether the webpage view has been loaded is learned by using the interface `webViewDidFinishLoad` (iOS) in the iOS (an operating system in an Apple mobile device); and whether the webpage view has been loaded is learned by using the interface `onPageFinished` (Android) in the Android (Android operating system). The location information of the placeholder element is a location of a page element in a viewable area, and the left and top locations of the placeholder element are located relative to a browser window.

15 **[0042]** 205a. If yes, obtain the location information of the placeholder element.

[0043] In the present implementation of the present disclosure, the location information of the placeholder element is obtained if the location information of the placeholder element can be obtained from the result of loading the to-be-displayed webpage view.

20 **[0044]** 206a. Add, to the location information, a client view corresponding to the placeholder element.

[0045] In the present implementation of the present disclosure, the placeholder element is represented as a block-level element. After step 206a, the method further includes the following: configuring status information of the placeholder element as status information corresponding to the client view. The status information of the placeholder element can be width/height information, resolution information, etc. of a status element. It is not limited in the present implementation of the present disclosure. It is worthwhile to note that, the placeholder element is represented as a block-level element, which can be as follows: The placeholder element is a block-level element, or the placeholder element is represented as a row-level element of a block-level element, that is, all elements that can change the status information of the placeholder element. It is not limited in the present implementation of the present disclosure.

25 **[0046]** In the present implementation of the present disclosure, the configuring status information of the placeholder element as status information corresponding to

30

the client view includes the following: setting the width/height information of the placeholder element to width/height information corresponding to the client view. For example, the width/height information of the element can be set by using the following code:

```
5         document.getElementById('native-view').style.height='400px';  
        document.getElementById('native-view').style.width='100px';
```

[0047] 400 is an actual height of the client view, and 100 is an actual width of the client view. In the present implementation of the present disclosure, the width/height information of the placeholder element is set to the width/height information corresponding to the client view so that the added client view is fully displayed in the to-be-displayed webpage view. Further, the width/height information of the placeholder element is set to the width/height information corresponding to the client view, to improve accuracy of displaying a client view and further improve accuracy and consistency of displaying a mixed view.

15 **[0048]** In the present implementation of the present disclosure, the adding a client view to the location information includes the following: obtaining location information corresponding to each placeholder element from the result of loading the webpage view; and adding, to the location information, a client view corresponding to the placeholder element. In the present implementation of the present disclosure, if
20 there are multiple placeholder elements in the webpage view, the location information corresponding to each placeholder element is obtained from the loading result after the webpage view has been loaded, and then, the client view corresponding to the placeholder element is added to the location information of each placeholder element, to add the multiple client views to the webpage view.

25 **[0049]** 207a. Display the webpage view that includes the added client view.

[0050] 205b. If no, output error prompt information of loading the webpage view.

[0051] Step 205b is in parallel with step 205a. The error prompt information of loading the webpage view is output if the location information of the placeholder element cannot be obtained from the result of loading the webpage view, to instruct
30 reloading of the webpage view. It is worthwhile to note that the location information of the placeholder element cannot be obtained from the result of loading the webpage view if a terminal device is disconnected from a network, or an error occurs in the webpage view loading process, etc.

[0052] The implementations of the present disclosure can be applied to, but is not

limited to, a scenario shown in FIG. 5. The scenario includes the following content:
First, a webpage view is loaded when a webpage view loading instruction is received,
that is, a webpage element in a dashed-line box of the webpage view in FIG. 5 is
loaded. Then, whether location information of a placeholder element can be obtained
5 from a result of loading the webpage view is determined, that is, whether location
information of a placeholder element "Contact us" can be obtained from the result of
loading the webpage view. If yes, the location information of the placeholder element
"Contact us" is obtained, and a client view is added to the location information
obtained from the WebView, that is, the client view is added to the WebView at a
10 corresponding location of the "Contact us" dashed-line box in FIG. 5, to display the
webpage view and the client view in a mixed way. If no, error prompt information of
loading the webpage view is output, to instruct reloading of the webpage view. The
placeholder element in the present implementation of the present disclosure can be
rendered by using a browser engine. Therefore, the client view is added to a location
15 corresponding to the placeholder element after the loading so that the client view can
scroll with the webpage view, that is, the client view in the "Contact us" dashed-line
box in FIG. 5 can scroll with the webpage view in a webpage view dashed-line box.
The placeholder element is added to the webpage view, to display a mixed view in a
flow layout, so that the present disclosure can improve accuracy and consistency of
20 displaying a mixed view.

[0053] The present implementation of the present disclosure provides the other
mixed view display method. First, the webpage view is loaded when the webpage
view loading instruction is received, where the webpage view includes the
placeholder element that is used to add the client view. Then, the location information
25 of the placeholder element is obtained from the result of loading the webpage view,
and the client view is added to the location information. Eventually, the webpage view
that includes the added client view is displayed. In a current method, a subview is
directly added to a WebView to display a mixed view. In comparison with the earlier
described current method, in the implementations of the present disclosure, first, the
30 location information of the placeholder element is obtained from the result of loading
the webpage view when the webpage view loading instruction is received; then, the
client view is added to a location of the placeholder element, to display the webpage
view and the client view in a mixed way. The placeholder element in the
implementations of the present disclosure can be rendered by using a browser engine.

Therefore, the client view can scroll with the webpage view by adding the client view to the location corresponding to the placeholder element after the loading. The present disclosure can improve accuracy and consistency of displaying a mixed view.

[0054] Further, an implementation of the present disclosure provides a mixed view display device. As shown in FIG. 3, the device includes a loading unit 31, an acquisition unit 32, an adding unit 33, and a display unit 34.

[0055] The loading unit 31 loads a webpage view when a webpage view loading instruction is received, where the webpage view includes a placeholder element that is used to add a client view.

[0056] The acquisition unit 32 obtains location information of the placeholder element from a result of loading the webpage view.

[0057] The adding unit 33 adds the client view to the location information.

[0058] The display unit 34 displays the webpage view that includes the added client view.

[0059] It is worthwhile to note that reference can be made to a corresponding description of the method shown in FIG. 1 for other corresponding descriptions of the functional units involved in the mixed view display device provided in the present implementation of the present disclosure. Details are omitted here for simplicity. However, it is worthwhile to note that the device in the present implementation can correspondingly implement all content in the previous method implementations.

[0060] The present implementation of the present disclosure provides the mixed view display device. First, the webpage view is loaded when the webpage view loading instruction is received, where the webpage view includes the placeholder element that is used to add the client view. Then, the location information of the placeholder element is obtained from the result of loading the webpage view, and the client view is added to the location information. Eventually, the webpage view that includes the added client view is displayed. In a current method, a subview is directly added to a WebView to display a mixed view. In comparison with the earlier described current method, in the implementations of the present disclosure, first, the location information of the placeholder element is obtained from the result of loading the webpage view when the webpage view loading instruction is received; then, the client view is added to a location of the placeholder element, to display the webpage view and the client view in a mixed way. The placeholder element in the implementations of the present disclosure can be rendered by using a browser engine.

Therefore, the client view can scroll with the webpage view by adding the client view to the location corresponding to the placeholder element after the loading. The present disclosure can improve accuracy and consistency of displaying a mixed view.

[0061] Further, an implementation of the present disclosure provides another mixed view display device. As shown in FIG. 4, the device includes a loading unit 41, an acquisition unit 42, an adding unit 43, and a display unit 44.

[0062] The loading unit 41 loads a webpage view when a webpage view loading instruction is received, where the webpage view includes a placeholder element that is used to add a client view.

[0063] The acquisition unit 42 obtains location information of the placeholder element from a result of loading the webpage view.

[0064] The adding unit 43 adds the client view to the location information.

[0065] The display unit 44 displays the webpage view that includes the added client view.

[0066] In the present implementation of the present disclosure, the device further includes a setting unit 45.

[0067] The adding unit 43 adds the placeholder element to the webpage view, where corresponding identification information is set for each placeholder element.

[0068] The setting unit 45 sets a client view corresponding to each placeholder element.

[0069] In the present implementation of the present disclosure, the placeholder element is represented as a block-level element. The device further includes: a configuration unit 46, configured to configure status information of the placeholder element as status information corresponding to the client view.

[0070] The configuration unit 46 sets width/height information of the placeholder element to width/height information corresponding to the client view.

[0071] In the present implementation of the present disclosure, the adding unit 43 includes the following: an acquisition module 431, configured to obtain location information corresponding to each placeholder element from the result of loading the webpage view; and an adding module 432, configured to add, to the location information, a client view corresponding to the placeholder element.

[0072] In the present implementation of the present disclosure, the device further includes the following: an output unit 47, configured to output error prompt information of loading the webpage view, if the location information of the

placeholder element is not obtained from the result of loading the webpage view.

[0073] It is worthwhile to note that reference can be made to a corresponding description of the method shown in FIG. 2 for other corresponding descriptions of the functional units involved in the mixed view display device provided in the present implementation of the present disclosure. Details are omitted here for simplicity. However, it is worthwhile to note that the device in the present implementation can correspondingly implement all content in the previous method implementations.

[0074] The present implementation of the present disclosure provides another mixed view display device. First, the webpage view is loaded when the webpage view loading instruction is received, where the webpage view includes the placeholder element that is used to add the client view. Then, the location information of the placeholder element is obtained from the result of loading the webpage view, and the client view is added to the location information. Eventually, the webpage view that includes the added client view is displayed. In a current method, a subview is directly added to a WebView to display a mixed view. In comparison with the earlier described current method, in the implementations of the present disclosure, first, the location information of the placeholder element is obtained from the result of loading the webpage view when the webpage view loading instruction is received; then, the client view is added to a location of the placeholder element, to display the webpage view and the client view in a mixed way. The placeholder element in the implementations of the present disclosure can be rendered by using a browser engine. Therefore, the client view can scroll with the webpage view by adding the client view to the location corresponding to the placeholder element after the loading. The present disclosure can improve accuracy and consistency of displaying a mixed view.

[0075] The mixed view display device includes a processor and a memory. The loading unit, the acquisition unit, the adding unit, the display unit, the setting unit, the configuration unit, the output unit, etc., are stored in the memory as program units. The processor executes the program units stored in the memory to implement corresponding functions.

[0076] The processor includes a kernel. The kernel invokes the corresponding program units from the memory. There can be one or more kernels. Accuracy of displaying a mixed view is improved by adjusting a kernel parameter.

[0077] The memory can include a non-persistent memory, a random access memory (RAM), a nonvolatile memory, etc. in a computer readable medium, for

example, a read-only memory (ROM) or a flash memory (flash RAM). The memory includes at least one storage chip.

[0078] The present application further provides a computer program product. When a data processing device executes the computer program product, program code
5 for initialization includes the following method steps: loading a to-be-displayed webpage view when a webpage view display location switch instruction is received, where the to-be-displayed webpage view includes a placeholder element that is used to add a client view; obtaining location information of the placeholder element from a result of loading the to-be-displayed webpage view; adding the client view to the
10 location information; and displaying the webpage view that includes the added client view.

[0079] A person skilled in the art should understand that the implementations of the present application can be provided as a method, a system, or a computer program product. Therefore, the present application can use hardware only implementations,
15 software only implementations, or implementations with a combination of software and hardware. In addition, the present application can use a form of a computer program product that is implemented on one or more computer-usable storage media (including but not limited to a magnetic disk storage, a CD-ROM, and an optical memory) that include computer-usable program code.

[0080] The present application is described with reference to the flowcharts and/or block diagrams of the method, the device, and the computer program product based on the implementations of the present application. It should be understood that computer program instructions can be used to implement each process and/or each block in the flowcharts and/or the block diagrams, and a combination of a process
20 and/or a block in the flowcharts and/or the block diagrams. These computer program instructions can be provided for a general-purpose computer, a dedicated computer, an embedded processor, or a processor of another programmable data processing device to generate a machine, so that the instructions executed by the computer or the processor of the another programmable data processing device generate a device for
25 implementing a specific function in one or more processes in the flowcharts and/or in one or more blocks in the block diagrams.

[0081] These computer program instructions can be stored in a computer readable memory that can instruct the computer or another programmable data processing device to work in a specific way so that the instructions stored in the computer
30

readable memory generate an artifact that includes an instruction device. The instruction device implements a specific function in one or more processes in the flowcharts and/or in one or more blocks in the block diagrams.

5 [0082] These computer program instructions can be loaded to the computer or another programmable data processing device so that a series of operation steps are performed on the computer or the other programmable device for computer-implemented processing. Therefore, the instructions executed on the computer or the other programmable device provide steps for implementing a specific function in one or more processes in the flowcharts and/or in one or more blocks in
10 the block diagrams.

[0083] In typical configuration, a computing device includes one or more central processing units (CPU), an input/output interface, a network interface, and a memory.

[0084] The memory may include a non-persistent memory, a random access memory (RAM), a nonvolatile memory, etc. in a computer readable medium, for
15 example, a read-only memory (ROM) or a flash memory (flash RAM). The memory is an example of the computer readable medium.

[0085] The computer readable medium includes persistent, non-persistent, movable, and unmovable media that can implement information storage by using any method or technology. Information can be a computer readable instruction, a data
20 structure, a program module, or other data. A computer storage medium includes but is not limited to a phase-change random access machine (PRAM), a static random access memory (SRAM), a dynamic random access memory (DRAM), a random access memory (RAM) of another type, a read-only memory (ROM), an electrically erasable programmable read-only memory (EEPROM), a flash memory or another
25 memory technology, a compact disc read-only memory (CD-ROM), a digital versatile disc (DVD), or another optical storage, a cassette, a cassette magnetic disk storage, or another magnetic storage device or any other non-transmission medium. The computer storage medium can be configured to store information that can be accessed by the computing device. Based on the definition in the present specification, the
30 computer readable medium does not include transitory computer-readable media (transitory media), for example, a modulated data signal and carrier.

[0086] The previous descriptions are merely the implementations of the present application, but are not intended to limit the present application. For a person skilled in the art, the present application can have various modifications and changes. Any

modifications, equivalent replacements, improvements, etc. made without departing from the spirit and principle of the present application shall fall within the scope of the claims in the present application.

CLAIMS

1. A mixed view display method, comprising:

loading code for a webpage view when a webpage view loading instruction is received, wherein the code for the webpage view comprises a placeholder element over which a client view controls that are written using a development language associated with a mobile operating system of a computing device are added when generating mixed views, wherein the webpage view is rendered by a browser application executing on the computing device (101, 203);

in response to loading the code for the webpage view, obtaining location information indicating a current position of the placeholder element within a browser window of the browser application from a result of loading the webpage view (102);

generating a mixed view, comprising adding the client view controls written using the development language associated with the mobile operating system of the computing device to the webpage view at the current position of the placeholder element within the browser window of the browser application (103); and

displaying the mixed view, including the webpage view and the added client view (104, 207a).

2. The method according to claim 1, wherein before loading the webpage view, the method further comprises:

adding the placeholder element to the webpage view, wherein corresponding identification information is set for each placeholder element (201); and

setting a client view corresponding to each placeholder element (202).

3. The method according to claim 2, wherein the placeholder element is represented as a block-level element, and after adding the client view to the location information, the method further comprises:

configuring status information of the placeholder element as status information corresponding to the client view.

4. The method according to claim 3, wherein the configuring status information of the placeholder element as status information corresponding to the client view comprises setting

width/height information of the placeholder element to width/height information corresponding to the client view.

5. The method according to claim 2, wherein adding the client view to the location information comprises:

obtaining location information corresponding to each placeholder element from the result of loading the webpage view (205a); and

adding, to the location information, a client view corresponding to the placeholder element (206a).

6. The method according to claim 1, wherein the method further comprises outputting error prompt information of loading the webpage view if the location information of the placeholder element is not obtained from the result of loading the webpage view (205b).

7. A mixed view display device, comprising:

a loading unit (31), configured to load code for a webpage view when a webpage view loading instruction is received, wherein the code for the webpage view comprises a placeholder element over which a client view controls that are written using a development language associated with a mobile operating system are added when generating mixed views, wherein the webpage view is rendered by a browser application;

an acquisition unit (32), configured to obtain location information indicating a current position of the placeholder element within a browser window of the browser application from a result of loading the webpage view;

an adding unit (33), configured to add the client view controls written using the development language associated with the mobile operating system to the webpage view at the current position of the placeholder element within the browser window of the browser application; and

a display unit (34), configured to display the webpage view that comprises the added client view.

8. The device according to claim 7, wherein the device further comprises a setting unit, and the adding unit adds the placeholder element to the webpage view, wherein corresponding

identification information is set for each placeholder element; and
the setting unit sets a client view corresponding to each placeholder element.

9. The device according to claim 8, wherein the placeholder element is represented as a block-level element, and the device further comprises a configuration unit, configured to configure status information of the placeholder element as status information corresponding to the client view.

10. The device according to claim 9, wherein the configuration unit sets width/height information of the placeholder element to width/height information corresponding to the client view.