CLAIMS

What is claimed:

1. A method of cross-domain two-way synchronization, comprising:

causing a web browser to load a first webpage from a first domain, wherein part of the first webpage is dynamic content that comprises one or more user interface elements;

causing the web browser to add a hidden or invisible iFrame into the first webpage, wherein the iFrame loads a second webpage from a second domain different from the first domain, wherein the second webpage comprises a set of one or more user interface elements that are identical to the one or more user interface elements contained in the first webpage, and wherein within the second webpage in the iFrame, a message listener is set up such that the message listener waits to receive messages from the dynamic content and only from that source;

causing the web browser to load a script file from within the iFrame;

subsequent to the adding of the iFrame and loading the script file within the iFrame, causing the web browser to bind one or more functions defined by the script file to the one or more user interface elements, the one or more functions including the web browser's Window.postMessage() functionality;

causing the web browser to, in response to the user interacting with the one or more user interface elements, use the one or more functions to send a message from the dynamic content to the message listener; wherein when the message is received at the message listener, the iFrame-contained second webpage makes a function call that causes a server-side code at the second domain to be executed to fulfill command or query that is requested by the user; and

causing the web browser to, after receiving a response from a server at the second domain using the web browser's Window.addEventListener() functionality, update the one or more user interface elements in both the first webpage and the second webpage in response to the received response.

2. The method of Claim 1, wherein the one or more user interface elements have an identical appearance as user interface elements of a webpage at the second domain.

3. The method of Claim 2, wherein the second webpage from the second domain is an online shopping cart page showing items that have been added to a cart.

4. The method of Claim 1, wherein the message sent to the second domain is structured so that the server at the second domain cannot determine that it was sent from within a webpage provided by the first domain.

5. The method of Claim 1, further comprising:

causing the web browser to load a third webpage, from the second domain, and allowing a user to interact with a user interface of the third webpage to create a persistent change to the user interface of the third webpage; and

causing the web browser to, when subsequently loading the first webpage, update the one or more user interface elements to reflect the persistent change that was made to the user interface of the third webpage.

6. The method of Claim 5, wherein the user interface of the third webpage is an online shopping cart, and wherein the persistent change to the user interface comprises an indication that one or more items have been added to the online shopping cart.

7. The method of Claim 5, further comprising:

causing the web browser to update the one or more user interface elements on the first webpage with a persistent change to those user interface elements, in response to user input; and

causing a web browser to, when subsequently loading the third webpage, update the user interface of the third webpage to reflect the persistent change that was made to the one or more user interface elements.

8. The method of Claim 5, wherein two identical sets of user interface elements are maintained in the user interface of the third webpage and the one or more user interface elements of the first webpage, respectively, and wherein a cross-domain user interface is maintained via synchronization between the two identical sets whenever a change is applied to one set and a webpage containing the other set is loaded.

9. The method of Claim 1, wherein the iFrame is hidden among other elements of the first webpage as a result of a size of or a location of the iFrame.

10. A system for cross-domain two-way synchronization, comprising:

one or more processors, and

non-transitory memory storing instructions that, when executed by the one or more processors, cause the one or more processors to:

cause a web browser to load a first webpage from a first domain, wherein part of the first webpage is dynamic content that comprises one or more user interface elements;

cause the web browser to add a hidden or invisible iFrame into the first webpage, wherein the iFrame loads a second webpage from a second domain different from the first domain, wherein the second webpage comprises a set of one or more user interface elements that are identical to the one or more user interface elements contained in the first webpage, and wherein within the second webpage in the iFrame, a message listener is set up such that the message listener waits to receive messages from the dynamic content and only from that source;

cause the web browser to load a script file from within the iFrame;

subsequent to the adding of the iFrame and loading the script file into the iFrame, cause the web browser to bind one or more functions defined by the script file to the one or more user interface elements, the one or more functions including the web browser's Window.postMessage() functionality;

cause the web browser to, in response to the user interacting with the one or more user interface elements, use the one or more functions to send a message from the dynamic content to the message listener; wherein when the message is received at the message listener, the iFrame-contained second webpage makes a function call that causes a server-side code at the second domain to be executed to fulfill command or query that is requested by the user; and

cause the web browser to, after receiving a response from a server at the second domain using the web browser's Window.addEventListener() functionality, update the one or more user interface elements in both the first webpage and the second webpage in response to the received response.

11. The system of Claim 10, wherein the one or more user interface elements have an identical appearance as user interface elements of a webpage at the second domain.

12. The system of Claim 11, wherein the second webpage from the second domain is an online shopping cart page showing items that have been added to a cart.

13. The system of Claim 10, wherein the message sent to the second domain is structured so that the server at the second domain cannot determine that it was sent from within a webpage provided by the first domain.

14. The system of Claim 10, wherein the instructions, when executed by the one or more processes, further cause the one or more processors to:

cause the web browser to load a third webpage, from the second domain, and allowing a user to interact with a user interface of the third webpage to create a persistent change to the user interface of the third webpage; and

cause the web browser to, when subsequently loading the first webpage, update the one or more user interface elements to reflect the persistent change that was made to the user interface of the third webpage.

15. The system of Claim 14, wherein the user interface of the third webpage is an online shopping cart, and wherein the persistent change to the user interface comprises an indication that one or more items have been added to the online shopping cart.

16. The system of Claim 14, wherein the instructions, when executed by the one or more processes, further cause the one or more processors to:

cause the web browser to update the one or more user interface elements on the first webpage with a persistent change to those user interface elements, in response to user input; and

cause a web browser to, when subsequently loading the third webpage, update the user interface of the third webpage to reflect the persistent change that was made to the one or more user interface elements.

17. The system of Claim 14, wherein two identical sets of user interface elements are maintained in the user interface of the third webpage and the one or more user interface elements of the first webpage, respectively, and wherein a cross-domain user interface is maintained via synchronization between the two identical sets whenever a change is applied to one set and a webpage containing the other set is loaded.

18. The system of Claim 10, wherein the iFrame is hidden among other elements of the first webpage as a result of a size of or a location of the iFrame.