

CLAIMS

1. A computer-implemented method executed by one or more processors, the method comprising:

receiving, by the one or more processors, a request for a service;
determining, by the one or more processors, a page identifier associated with the service, wherein the page identifier is an identifier of a first jump page;
retrieving, by the one or more processors, a first jump route from a local page routing table based on the page identifier, wherein the first jump route provides an online route for online retrieving of the first jump page from a first server;
in response to determining the first jump route, sending a query to a second server, the query inquiring whether the first jump route is a latest jump route associated with the service;
receiving, from the second server, a first message indicating that the first jump route is the latest jump route; and
in response to receiving the first message, loading, by the one or more processors, the first jump page based on the first jump route.

2. The method of claim 1, further comprising:

receiving, by the one or more processors and from the second server, a second message including a second jump route as the latest jump page route associated with the page identifier;
in response to receiving the second message, updating, by the one or more processors, the local page routing table to include the second jump route; and
loading a second jump page based on the second jump route.

3. The method of claim 2, wherein the second jump page is loaded from the first server and based on the second jump page.

4. The method of claim 2, wherein the second server is configured to transmit the second jump route based on a new service associated with the page identifier and deployed by the first server.

5. The method of claim 1, wherein the one or more processors are associated with a client computing system and the local page routing table is specific to the client computing system.
6. The method of claim 1, wherein loading the first jump page comprises:
determining, by the one or more processors, that page data of the first jump page is stored in a page database that is in communication with the one or more processors, and in response, retrieving the page data from the page database.
7. The method of claim 6, further comprising:
receiving, by the one or more processors, the page data of the first jump page from the first server; and
updating, by the one or more processors, the page database to include the page data of the first jump page.
8. The method of claim 1, wherein loading the first jump page comprises:
determining, by the one or more processors, that a local page database lacks page data of the first jump page, and in response, requesting the page data from a server, wherein the local page database is configured to store page data of the first jump page.
9. A non-transitory, computer-readable medium storing one or more instructions executable by one or more processors to perform the method of any one of claims 1 to 8.
10. A computer-implemented system, comprising:
one or more processors; and
one or more computer memory devices interoperably coupled with the one or more processors and having tangible, non-transitory, machine-readable media storing one or more instructions that, when executed by the one or more processors, perform the method of any one of claims 1 to 8.