

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 first service;
 - determining, by the one or more processors, a first jump page associated with the first service, wherein the first jump page is associated with a page identifier;
 - retrieving, based on the page identifier and by the one or more processors, a first jump route from a page routing table, wherein the first jump route is associated with the first service, and wherein the first jump route includes the page identifier and represents a route for retrieving the first jump page;
 - in response to retrieving the first jump route, loading, by the one or more processors, the first jump page based on the first jump route;
 - receiving, by the one or more processors, a second jump route from a jump route server, wherein the second jump route includes the page identifier and represents a route for retrieving a second jump page, the second jump route being associated with a deployed second service that is different from the first service; and
 - updating, by the one or more processors, the page routing table based on the second jump route.
2. The method of claim 1, further comprising
 - loading, by the one or more processors, the second jump page based on the second jump route.
3. The method of claim 1, wherein the second jump route is received in response to sending a first query to the jump route server, wherein the jump route server is configured to transmit the second jump route based on the deployed second service and in response to the first query.
4. The method of claim 1, wherein the one or more processors are associated with a client computing system.

5. 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 of the first jump page from a page data server, wherein the local page database is configured to store the page data of the first jump page.
6. The method of claim 5, further comprising:
receiving, by the one or more processors, the page data of the first jump page from the page data server; and
updating, by the one or more processors, the local page database to include the page data of the first jump page.
7. 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 local page database, and in response, retrieving the page data of the first jump page from the local page database.
8. 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 7.
9. 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 7.