CLAIMS

What is claimed is:

1. A data synchronization method, comprising:

receiving, by an application, a payment service request;

determining, by the application, to-be-queried user account data of a user based upon the payment service request;

identifying, using the application, a data identifier associated with to-be-queried user account data;

identifying, using the application and the data identifier, cached user account data in a cache associated with the data identifier and a first version identifier corresponding to the cached user account data in the cache associated with the data identifier, wherein the first version identifier includes time information of a change from a previous version identifier corresponding to the cached user account data to the first version identifier corresponding to the cached user account data;

identifying a current version identifier of the application, wherein the application is associated with a database storing a plurality of pieces of database data, wherein each piece of the plurality of pieces of database data is associated with a version identifier that is the same as the current version identifier of the application; and

in response to determining that the first version identifier corresponding to the cached user account data in the cache associated with the data identifier does not match the current version identifier of the application:

obtaining, using the application, database data from the database associated with the data identifier; and

replacing, using the application, the cached user account data in the cache associated with the data identifier with the database data from the database associated with the data identifier.

2. The method according to claim 1, further comprising:

in response to determining that the first version identifier corresponding to the cached user account data in the cache associated with the data identifier matches the current version identifier of the application, obtaining, using the application, the cached user account data in the cache associated with the data identifier.

3. The method according to claim 1, wherein identifying the first version identifier corresponding to the cached user account data in the cache associated with the data identifier comprises:

obtaining, based on the data identifier, the cached user account data in the cache associated with the data identifier; and

identifying, from the cached user account data in the cache associated with the data identifier, the first version identifier.

4. The method according to claim 1, wherein identifying the first version identifier corresponding to the cached user account data in the cache associated with the data identifier comprises:

identifying, in the cache, an actual identifier that matches the data identifier; and

identifying, from the actual identifier, the first version identifier.

5. The method according to claim 1, wherein replacing the cached user account data in the cache associated with the data identifier with the database data from the database associated with the data identifier comprises:

invalidating the cached user account data in the cache associated with the data identifier; and

storing, in the cache, the database data from the database associated with the data identifier.

6. The method according to claim 1, further comprising:

in response to detecting a change to version information of the application, determining the current version identifier of the application based on changed version information.

7. The method according to claim 1, wherein the current version identifier of the application is determined based on current version information of the application.

8. A non-transitory, computer-readable medium storing one or more instructions executable by a computer system to perform operations comprising:

receiving, by an application, a payment service request;

determining, by the application, to-be-queried user account data of a user based upon the payment service request;

identifying, using the application, a data identifier associated with to-be-queried user

account data;

identifying, using the application and the data identifier, cached user account data in a cache associated with the data identifier and a first version identifier corresponding to the cached user account data in the cache associated with the data identifier, wherein the first version identifier includes time information of a change from a previous version identifier corresponding to the cached user account data to the first version identifier corresponding to the cached user account data;

identifying a current version identifier of the application, wherein the application is associated with a database storing a plurality of pieces of database data, wherein each piece of the plurality of pieces of database data is associated with a version identifier that is the same as the current version identifier of the application; and

in response to determining that the first version identifier corresponding to the cached user account data in the cache associated with the data identifier does not match the current version identifier of the application:

obtaining, using the application, database data from the database associated with the data identifier; and

replacing, using the application, the cached user account data in the cache associated with the data identifier with the database data from the database associated with the data identifier.

9. The non-transitory, computer-readable medium according to claim 8, wherein the operations comprise:

in response to determining that the first version identifier corresponding to the cached user account data in the cache associated with the data identifier matches the current version identifier of the application, obtaining, using the application, the cached user account data in the cache associated with the data identifier.

10. The non-transitory, computer-readable medium according to claim 8, wherein identifying the first version identifier corresponding to the cached user account data in the cache associated with the data identifier comprises:

obtaining, based on the data identifier, the cached user account data in the cache associated with the data identifier; and

identifying, from the cached user account data in the cache associated with the data identifier, the first version identifier.

11. The non-transitory, computer-readable medium according to claim 8, wherein identifying the first version identifier corresponding to the cached user account data in the cache associated with the data identifier comprises:

identifying, in the cache, an actual identifier that matches the data identifier; and

identifying, from the actual identifier, the first version identifier.

12. The non-transitory, computer-readable medium according to claim 8, wherein replacing the cached user account data in the cache associated with the data identifier with the database data from the database associated with the data identifier comprises:

invalidating the cached user account data in the cache associated with the data identifier; and

storing, in the cache, the database data from the database associated with the data identifier.

13. The non-transitory, computer-readable medium according to claim 8, wherein the operations comprise:

in response to detecting a change to version information of the application, determining the current version identifier of the application based on changed version information.

14. The non-transitory, computer-readable medium according to claim 8, wherein the current version identifier of the application is determined based on current version information of the application.

15. A computer-implemented system, comprising:

one or more computers; and

one or more computer memory devices interoperably coupled with the one or more computers and having tangible, non-transitory, machine-readable media storing one or more instructions that, when executed by the one or more computers, perform one or more operations comprising:

receiving, by an application, a payment service request;

determining, by the application, to-be-queried user account data of a user based upon the payment service request;

identifying, using the application, a data identifier associated with to-be-queried user account data;

identifying, using the application and the data identifier, cached user account data in a cache associated with the data identifier and a first version identifier corresponding to the cached user account data in the cache associated with the data identifier, wherein the first version identifier includes time information of a change from a previous version identifier corresponding to the cached user account data to the first version identifier corresponding to the cached user account data;

identifying a current version identifier of the application, wherein the application is associated with a database storing a plurality of pieces of database data, wherein each piece of the plurality of pieces of database data is associated with a version identifier that is the same as the current version identifier of the application; and

in response to determining that the first version identifier corresponding to the cached user account data in the cache associated with the data identifier does not match the current version identifier of the application:

obtaining, using the application, database data from the database associated with the data identifier; and

replacing, using the application, the cached user account data in the cache associated with the data identifier with the database data from the database associated with the data identifier.

16. The computer-implemented system according to claim 15, wherein the operations comprise:

in response to determining that the first version identifier corresponding to the cached user account data in the cache associated with the data identifier matches the current version identifier of the application, obtaining, using the application, the cached user account data in the cache associated with the data identifier.

17. The computer-implemented system according to claim 15, wherein identifying the first version identifier corresponding to the cached user account data in the cache associated with the data identifier comprises:

obtaining, based on the data identifier, the cached user account data in the cache associated with the data identifier; and

identifying, from the cached user account data in the cache associated with the data identifier, the first version identifier.

18. The computer-implemented system according to claim 15, wherein identifying the first version identifier corresponding to the cached user account data in the cache associated with the data identifier comprises:

identifying, in the cache, an actual identifier that matches the data identifier; and

identifying, from the actual identifier, the first version identifier.

19. The computer-implemented system according to claim 15, wherein replacing the cached user account data in the cache associated with the data identifier with the database data from the database associated with the data identifier comprises:

invalidating the cached user account data in the cache associated with the data identifier; and

storing, in the cache, the database data from the database associated with the data identifier.

20. The computer-implemented system according to claim 15, wherein the operations comprise:

in response to detecting a change to version information of the application, determining the current version identifier of the application based on changed version information.