CLAIMS:

1. A module for tracking information shared between a first business application and a second business application, wherein the first and second business applications comprise disparate business applications, the module comprising:

a processing unit; and

a non-transitory media readable by the processing unit, the media storing instructions that when executed by the processing unit cause the processing unit to:

receive a first-message from the first business application;

retrieve local-first primary keys L1-Keys associated with the first-message, wherein each of the local-first primary keys L1-Keys comprise unique identifiers associated with records in the first message;

for each of the retrieved local-first primary keys L1-Keys that has been assigned a global key GKey,

select local-second primary key L2-Keys and semantic keys SKeys associated with the assigned global key GKey,

identify from the selected local-second primary keys L2-Keys a specific local-second primary key L2-Key and a specific semantic key SKey that have been paired with the local-first primary key L1-Key, and

retrieve system identities associated with the local-first primary key L1-Key, the specific local-second primary key L2-Key and the specific semantic key SKey,

wherein each of the local-second primary keys L2-Keys comprise unique identifiers associated with records in a second message associated with the second business application;

for each of the retrieved local-first primary keys L1-Keys that has not been assigned a global key GKey,

generate and assign a global key GKey and a semantic key SKey to the local-first primary key L1-Key to form a (Gkey, L1-Key, SKey) entry,

receive a local-second primary key L2-Key generated for the (GKey, L1-Key, SKey) entry and associate the local-second primary key L2-Key to the (GKey, L1-Key, SKey) entry, and

retrieve system identities associated with the local-first primary key L1-Key and the local-second primary key L2-Key;

append all the retrieved local-first primary keys L1-Keys, their associated global keys GKeys and semantic keys SKeys, and the respective retrieved system identities to the first-message, wherein the semantic keys SKeys are made available to and identifiable by the first and second business applications;

provide the appended first-message to an integration logic module, whereby the integration logic module is configured to process the appended first-message and provide the processed first-message to the second business application;

receive a response from the second business application, wherein the response is stored in a delivery log;

determine, from the response, a delivery status of the appended first-message.

2. The module according to claim 1 wherein the instructions to determine the delivery status of the appended first-message comprises:

instructions for directing the processing unit to:

receive the semantic keys SKeys from the first business application, wherein the semantic keys SKeys were obtained by the first business applications from the module and whereby the semantic keys SKeys are associated with the local-first primary key L1-Keys of the first message;

identify local-first primary keys L1-Keys associated with the received semantic keys SKeys, and

validate the delivery status of the appended first-message when all the identified local-first primary keys L1-Keys are found in the response.

3. The module according to claim 1 wherein the module further comprises:

instructions for directing the processing unit to:

assign timestamps to all instructions executed by the processing unit; and

determine from the delivery log, using a semantic key SKey associated with the appended first-message, a delivery status of the appended first-message.

4. The module according to claim 3 wherein the instructions to determine from the delivery log, using the semantic key SKey associated with the appended first-message, the delivery status of the appended first-message comprises:

instructions for directing the processing unit to:

retrieve (GKey, L1-Key, L2-Key) entries associated with the SKey from the delivery log;

arrange the retrieved (GKey, L1-Key, L2-Key) entries according to their timestamps; and

validate, using the timestamps, the delivery status of the appended first-message.

5. The module according to claim 1 wherein the module further comprises:

instructions for directing the processing unit to:

receive, from the integration logic module, errors, messages associated with the errors, and a queue indicator linked with each of the messages;

rectify the errors associated with the messages; and

return each rectified message to its original queue location in the integration logic module based on the queue indicator linked with the message.

6. The module according to claim 1 wherein the local-second primary key L2-Key generated for the (GKey, L1-Key) entry is generated by the module based on a Global Unique Identifier (GUID), a random number, a primary key generation algorithm in combination with a number range, or a primary key generation algorithm in combination with a defined logic.

7. The module according to claim 1 wherein the local-second primary key L2-Key generated for the (GKey, L1-Key) entry is generated as part of a response by the second business application in response to an Application Programming Interface (API) call from the first business application, whereby the generated response is provided to the first business application.

8. The module according to claim 1 wherein the local-second primary key L2-Key generated for the (GKey, L1-Key) entry is generated by the module based on a bundle of documents received from the second business application, the bundle of documents being sent by the second business application to the first business application in response to a continuous API call from the first business application, whereby the bundle of documents comprise documents having a same document-type as the first-message and having a similar time-stamp as a first API that was sent from the first business application to the second business application.

9. The module according to claim 1 wherein the local-second primary key L2-Key generated for the (GKey, L1-Key) entry is generated by the module based on a bundle of documents received from the second business application, the bundle of documents being sent by the second business application to the first business application in response to a continuous API call from the first business application, whereby the bundle of documents comprise documents having a same document-type as the first-message and having fields that match predefined requirements.

10. A method for tracking information shared between a first business application and a second business application, wherein the first and second business applications comprise disparate business applications, the method comprising:

receiving, using a module, a first-message from the first business application;

retrieving, using the module, local-first primary keys L1-Keys associated with the first-message, wherein each of the local-first primary keys L1-Keys comprise unique identifiers associated with records in the first message;

for each of the retrieved local-first primary keys L1-Keys that has been assigned a global key GKey,

select local-second primary keys L2-Keys and semantic keys SKeys associated with the assigned global key GKey,

identify from the selected local-second primary keys L2-Keys a specific local send primary key L2-Key and a specific semantic key SKey that have been paired with the local-first primary key L1-Key, and

retrieve system identities associated with the local-first primary key L1-Key, the specific local-second primary key L2-Key and the specific semantic key SKey,

wherein each of the local-second primary keys L2-Keys comprise unique identifiers associated with records in a second message associated with the second business application;

for each of the retrieved local-first primary keys L1-Keys that has not been assigned a global key GKey,

generate and assign a global key GKey and a semantic key SKey to the local-first primary key L1-Key to form a (GKey, L1-Key, SKey) entry,

receive a local-second primary key L2-Key generated for the (GKey, L1-Key, SKey) entry and associate the local-second primary key L2-Key to the (GKey, L1-Key, SKey) entry, and

retrieve system identities associated with the local-first primary key L1-Key and the local-second primary key L2-Key;

appending, using the module, all the retrieved local-first primary keys L1-Keys, their associated global keys GKeys and semantic keys SKeys, and the respective retrieved system identities to the first-message, wherein the semantic keys SKeys are made available to and identifiable by the first and second business applications;

providing, using the module, the appended first-message to an integration logic module;

receiving, using the module, a response from the second business application, wherein the response is stored in a delivery log;

determining, using the module, from the response, a delivery status of the appended first-message.

11. The method according to claim 11 wherein the step of determining the delivery status of the appended first-message comprises:

receiving, using the module, the semantic keys SKeys from the first business application, wherein the semantic keys SKeys were obtained by the first business applications from the module and whereby the semantic keys SKeys are associated with the local-first primary key L1-Keys of the first message;

identifying, using the module, local-first primary keys L1-Keys associated with the received semantic keys SKeys; and

validating, using the module, the delivery status of the appended first-message when all the identified local-first primary keys L1-Keys are found in the response.

12. The method according to claim 10 wherein the method further comprises the steps of:

assigning, using the module, timestamps to all instructions executed by the processing unit; and

determining, using the module, from the delivery log, using a semantic key SKey associated with the appended first-message, a delivery status of the appended first-message.

13. The method according to claim 12 wherein the step of determining from the delivery log, using the semantic key SKey associated with the appended first-message, the delivery status of the appended first-message comprises the steps of:

retrieving, using the module, (GKey, L1-Key, L2-Key) entries associated with the SKey from the delivery log;

arranging, using the module, the retrieved (GKey, L1-Key, L2-Key) entries according to their timestamps; and

validating, using the module, based on the timestamps, the delivery status of the appended first-message.

14. The method according to claim 10 wherein the method further comprises the steps of:

receiving, using the module, from the integration logic module, errors, messages associated with the errors, and a queue indicator linked with each of the messages;

rectifying, using the module, the errors associated with the messages; and

returning, using the module, each rectified message to its original queue location in the integration logic module based on the queue indicator linked with the message.

15. The method according to claim 10 wherein the local-second primary key L2-Key generated for the (GKey, L1-Key) entry is generated by the module based on a Global Unique Identifier (GUID), a random number, a primary key generation algorithm in combination with a number range, or a primary key generation algorithm in combination with a defined logic.

16. The method according to claim 10 wherein the local-second primary key L2-Key generated for the (GKey, L1-Key) entry is generated as part of a response by the second business application in response to an Application Programming Interface (API) call from the first business application, whereby the generated response is provided to the first business application.

17. The method according to claim 10 wherein the local-second primary key L2-Key generated for the (GKey, L1-Key) entry is generated by the module based on a bundle of documents received from the second business application, the bundle of documents being sent by the second business application to the first business application in response to a continuous API call from the first business application, whereby the bundle of documents comprise documents having a same document-type as the first-message and having a similar time-stamp as a first API that was sent from the first business application to the second business application.

18. The method according to claim 10 wherein the local-second primary key L2-Key generated for the (GKey, L1-Key) entry is generated by the module based on a bundle of documents received from the second business application, the bundle of documents being sent by the second business application to the first business application in response to a continuous API call from the first business application, whereby the bundle of documents comprise documents having a same document-type as the first-message and having fields that match predefined requirements.