

CLAIMS

1. A computer-implemented system for interfacing a set of one or more communication networks and a second communication network, the system comprising:
 - a memory storing instructions; and
 - at least one processor configured to execute the instructions for:
 - receiving a first message from a first communication network among the set of communication networks, the first message comprising a first unknown message header and a first unknown message body;
 - parsing the first unknown message header to determine a first variable corresponding to the identity of the first communication network and a second variable corresponding to a first destination;
 - identifying, based on a preconfigured data structure and the first variable, that the first unknown message header and the first unknown message body are in a predetermined format;
 - deserializing the first unknown message body into its underlying data structure based on the predetermined format;
 - converting the first unknown message body into a standardized format for the second communication network through an accelerated conversion process;
 - reserializing the standardized message body for transmission; and
 - transmitting the first standardized message body to the first destination based on the second variable,
 - wherein the first unknown message body comprises one or more variables, and
 - wherein converting the first unknown message body into a standardized format comprises:
 - determining an interface map between the predetermined format and the standardized format of the second communication network based on the preconfigured data structure; and
 - assigning values of the variables of the first unknown message body to corresponding variables of the standardized format based on the interface map.

2. The computer-implemented system of claim 1, the instructions further comprising:

- receiving a second message from the second communication network, the second message comprising a second standardized message body and a second destination;
- identifying a second communication protocol of the second message and a third communication network among the set of communication networks;
- determining that the third communication network can receive messages in the predetermined format;
- converting the second standardized message into the predetermined format through the accelerated conversion process;
- serializing the converted second message for transmission; and
- transmitting the second converted message to the second destination.

3. The computer-implemented system of claim 2, wherein converting the second standardized message comprises appending a second message header to the second converted message, the second message header comprising a third variable corresponding to the second destination.

4. The computer-implemented system of claim 1, wherein the first message is received using at least one of a Representational State Transfer (REST) API, a RESTful API, a file transfer API, a database-to-database API, a Simple Object Access Protocol (SOAP) API, or a Remote Procedure Call (RPC) API.

5. The computer-implemented system of claim 1, wherein a first communication protocol of the first message comprises at least one of JavaScript Object Notation (JSON), YAML Ain't Markup Language (YAML), Extensible Markup Language (XML), Tom's Obvious Minimal Language (TOML), CoffeeScript Object Notation (CSON), or MessagePack.

6. The computer-implemented system of claim 1, wherein the preconfigured data structure matches one or more known communication networks with one or more communication protocols, the one or more communication protocols including a predetermined communication protocol corresponding to the predetermined format.

7. The computer-implemented system of claim 1, wherein individual communication networks of the set of communication networks are non-overlapping with the second communication network and among themselves.

8. The computer-implemented system of claim 1, wherein each element in the predetermined format has one-to-one correspondence to its counterpart in the standardized format.

9. A computer-implemented method for interfacing a set of one or more communication networks and a second communication network, the method comprising:

- receiving a first message from a first communication network among the set of communication networks, the first message comprising a first unknown message header and a first unknown message body;

- parsing the first unknown message header to determine a first variable corresponding to the identity of the first communication network and a second variable corresponding to a first destination;

- identifying, based on a preconfigured data structure and the first variable, that the first unknown message header and the first unknown message body are in a predetermined format;

- deserializing the first unknown message body into its underlying data structure based on the predetermined format;

- converting the first unknown message body into a standardized format for the second communication network through an accelerated conversion process;

- reserializing the standardized message body for transmission; and

transmitting the first standardized message body to the first destination based on the second variable,

wherein the first unknown message body comprises one or more variables, and

wherein converting the first unknown message body into a standardized format comprises:

determining an interface map between the predetermined format and the standardized format of the second communication network based on the preconfigured data structure; and

assigning values of the variables of the first unknown message body to corresponding variables of the standardized format based on the interface map.

10. The computer-implemented method of claim 9, the method further comprising: receiving a second message from the second communication network, the second message comprising a second standardized message body and a second destination;

identifying a second communication protocol of the second message and a third communication network among the set of communication networks;

determining that the third communication network can receive messages in the predetermined format;

converting the second standardized message body into the predetermined format through the accelerated conversion process;

serializing the converted second message for transmission; and

transmitting the second converted message body to the second destination.

11. The computer-implemented method of claim 10, wherein transmitting the second standardized message body to the second destination comprises:

appending a second message header to the second converted message body to create a second converted message, the second converted message header comprising a third variable corresponding to the second destination; and

transmitting the second converted message to the second destination using the second communication network.

12. The computer-implemented method of claim 9, wherein the first message is received using at least one of a Representational State Transfer (REST) API, a RESTful API, a file transfer API, a database-to-database API, a Simple Object Access Protocol (SOAP) API, or a Remote Procedure Call (RPC) API.

13. The computer-implemented method of claim 9, wherein a first communication protocol of the first message comprises at least one of JavaScript Object Notation (JSON), YAML Ain't Markup Language (YAML), Extensible Markup Language (XML), Tom's Obvious Minimal Language (TOML), CoffeeScript Object Notation (CSON), or MessagePack.

14. The computer-implemented method of claim 9, wherein the preconfigured data structure matches one or more known communication networks with one or more communication protocols, the one or more communication protocols including a predetermined communication protocol corresponding to the predetermined format.

15. The computer-implemented method of claim 9, wherein individual communication networks of the set of communication networks are non-overlapping with the second communication network and among themselves.

16. The computer-implemented method of claim 9, wherein each element in the predetermined format has one-to-one correspondence to its counterpart in the standardized format.

17. A computer-implemented system for interfacing a set of one or more communication networks and a second communication network, the system comprising:
an inbound endpoint layer configured to receive an inbound message from a first communication network among the set of communication networks, the inbound message comprising an unknown inbound message header and an unknown inbound message body, the inbound endpoint layer further comprising an inbound parser

configured to parse the unknown inbound message header to determine a first portion corresponding to an identity of the first communication network and a second portion corresponding to a first destination located within the second communication network;

an inbound converting layer configured to convert the unknown inbound message body into a standardized format based on the first portion and a preconfigured data structure of known communication networks through an accelerated conversion process;

a routing layer configured to transmit the standardized inbound message body to the first destination;

an outbound endpoint layer configured to receive an outbound message from the second communication network, the outbound message comprising a standardized outbound message body and a second destination located within a third communication network among the set of communication networks; and

an outbound converting layer configured to convert the standardized outbound message body into a predetermined format based on the second destination and the preconfigured data structure,

wherein the routing layer is further configured to transmit the predetermined format outbound message body to the second destination, and

wherein the outbound converting layer is further configured to convert the standardized outbound message body by:

identifying the third communication network corresponding to the second destination based on the preconfigured data structure; and

appending an outbound message header to the predetermined format outbound message body to create a predetermined format outbound message, the outbound message header comprising a first data field corresponding to the second destination.