

IDs, or a premium package of **1000** authentication IDs. Other sizes of packages may be preset. A client application **15** may also be prompted by the authentication ID provider to enter the number of authentication IDs in the pool of authentication IDs.

[**0106**] Alternatively, the billing module **970** may charge based upon use of an authentication ID. In such a scenario, the metering module **950** tracks and records usage of the pool of authentication IDs. The information collected by the metering module **950** is stored in the repository **1401**, or another central repository which may be accessed by components of the gateway module **900**.

[**0107**] FIG. **15** shows a method for providing a pool of authentication IDs (**1500**) for use in web services communication. The method begins with the login module **1400** receiving a request for a pool of authentication IDs from a client application **15**. Specifically, the login services module **1400** receives client application credentials from the client application interface unit **310** which receives the request over a secured channel (**1501**). The client application credentials are passed to the authentication module **520** to authenticate the client application **15** (**1502**). The authentication module **520** may reference the information repository **1401** when authenticating the client application **15**. If the client application **15** is not authentic (**1502**), i.e., the client application credentials are incorrect, then the request will be rejected (**1505**). If the client application **15** is authentic (**1502**), then the request is passed to the authentication ID provider **940**. The authentication ID provider **940** creates and assigns a pool of authentication IDs and sends the pool of authentication IDs to the client application interface unit **310** to send to the client application **15** (**1503**) over a secured channel. The pool of authentication IDs may be parameters passed as parameters by the client application **15** when invoking client calls **701**. The authentication IDs may be created and assigned by code in the authentication ID provider **940**. Alternatively, a bank of authentication IDs may be stored in the information repository **1401** to be assigned by the authentication ID provider **940**. The client application **15** is sent the pool of authentication IDs (**1504**) and the method is done (**1506**). The client application **15** may now use the pool of authentication IDs. Other steps may be added to the method (**1500**), such as billing users using client applications **15** or web service providers **20**, for the authentication IDs or the use of authentication IDs. As described above, client applications **15** may be billed based upon the amount of authentication IDs in the pool of authentication IDs. Furthermore, the usage of the authentication IDs may be tracked and metered for billing client applications **15** on a per use basis.

[**0108**] The assignment of the pool of authentication IDs may be registered in the information repository **1401**. Alternatively, the assignment of the pool of authentication IDs may be registered with the authentication ID validator **945**. The registration of the pool of authentication IDs may be in the form of a file containing the assigned pool of authentication IDs, their status, such as used and not used, and client application credentials information, such as the user name and password. Other information may be added to the registration file as desired. The registration file may be referenced by the authentication ID validator **945** when the client application **15** uses each authentication ID.

[**0109**] One aspect of the gateway module **900** relates to the field of dynamic application programming interface (API) and type reflection as well as dynamic modification of these API and types used. Another aspect of the gateway module **900** relates to the authentication of a client application **15** to gain access to the web services **25** that are hosted by the remote server. Another aspect of the gateway module **900** pertains to the field of web services hosting, monitoring and administration. Preferably, the gateway module **900** is independent from the platforms on which the web services **25** are hosted. Furthermore, the gateway module **900** is independent from the protocol used to access the web services **25**. A further aspect of the gateway module **900** pertains to the field of billing and authorization of web services **25**.

[**0110**] FIG. **16** shows an example of enhanced web service deployment environment **1601** using the gateway module **300**. The environment **1601** comprises client applications **15**, an internal web services management system **1600**, and remote web services **25**. The internal web services management system **1600** comprises a repository **1610**, an authentication and authorization module **1620**, a deployment and registration module **1630**, a logging and metering module **1640**, an administrator console **1650**, a reporting module **1660**, a gateway module **300**, and local web services **25**. Components may be added or removed from the internal web services management system **1600**.

[**0111**] FIG. **16** shows licensed web service client applications **15** interacting with the web service gateway module **1650** through a single entry point, the gateway module **300**. Communication from a client application **15**, such as calls for WSDL or SOAP messages, are intercepted by the gateway module **300**. The communication are authenticated and authorized using the authentication and registration module **1620**. As the SOAP calls are being made, they are logged for auditing, metering and billing purposes by the logging and metering module **1640**. The metering module **1640** records events that take place that can later be used for analysis. A client of the logging and metering module **1640** is the administrator console **1680** that enables the administrator of the web services management system **1600** to track how the system **1600** has been used. Statistics may also be generated and stored in the repository **1610** in order to help the administrator make business decisions such as offering new services, adjusting prices, and other business decisions relating to the packaging of web services.

[**0112**] The gateway module **300** delegates the authorized requests to the actual web service **25**, which may be either hosted locally or on remote systems. The reporting module **1660** generates reports on the usage for billing and auditing purposes. The deployment and registration module **1630** registers local and remote web services **25** with the gateway module **300**. There is also an administrator console **1650** that is used to perform administration tasks, including provisioning users, setting or modifying access rights, generating reports, and other tasks desired for administration purposes. Finally, the repository **1610** is used to store information desired and accessible by components of the internal web services management system **1600**. The repository **1610** may be a database. The components of the internal web services management system **1600** may be software code or code embedded in hardware.