A gaming network including gaming machines and gaming services further includes a language translation service that provides systems and methods for providing translations for clients in the gaming network. The gaming services framework comprises a set of services, protocols, XML schemas, and methods for providing gaming system functionality in a distributed, network based architecture that includes gaming machines and servers. The systems and methods provide a service-oriented framework for gaming and property management based upon internetworking technology and web services concepts.
FIG. 2
Publish Availability of Language Translation Service

Discover Language Translation Service

Register With Language Translation Service

Process Language Translation Service Requests

FIG. 5A
GAMING NETWORK ENVIRONMENT HAVING A LANGUAGE TRANSLATION SERVICE

CROSS REFERENCE TO RELATED APPLICATIONS

[0001] This application is a continuation of U.S. patent application Ser. No. 10/857,433, entitled “GAMING NETWORK ENVIRONMENT HAVING A LANGUAGE TRANSLATION SERVICE”, filed May 27, 2004, which claims the priority benefit of U.S. Provisional Patent Application Ser. No. 60/473,733, entitled “LANGUAGE TRANSLATION SERVICE IN A SERVICE-ORIENTED GAMING NETWORK ENVIRONMENT”, filed May 27, 2003; and is related to U.S. patent application Ser. No. 10/788,903, entitled “A SERVICE-ORIENTED GAMING NETWORK ENVIRONMENT”, (Attorney Docket 1842.020US 1), filed on Feb. 26, 2004 and assigned to the same assignee as the present application; each of which are hereby incorporated by reference herein for all purposes.

FIELD

[0002] The present invention relates generally to software and hardware systems for gaming machines and gaming machine networks, and more particularly to providing a language translation service in a service-oriented gaming network environment.

LIMITED COPYRIGHT WAIVER

[0003] A portion of the disclosure of this patent document contains material to which the claim of copyright protection is made. The copyright owner has no objection to the facsimile reproduction by any person of the patent document or the patent disclosure, as it appears in the U.S. Patent and Trademark Office file or records, but reserves all other rights whatsoever.

BACKGROUND

[0004] Today’s gaming terminal typically comprises a computerized system controlling a video display or reel that provide wagering games such as video and mechanical slots, video card games (poker, blackjack, etc.), video keno, video bingo, video pachinko and other games typical in the gaming industry. In addition, support computing systems such as accounting, player tracking and other “back office” systems exist in order to provide support for a gaming environment.

[0005] In order to prevent players from becoming bored, new versions of wagering games, and alterations to existing games are constantly being developed. In the past, the game software and content for gaming terminals and back office systems have been developed using proprietary or closed hardware, operating systems, application development systems, and communications systems. Sometimes these systems are provided by a single vendor.

[0006] Additionally, gaming is popular in many different countries with each country having their own native language. Further, many visitors to a gaming establishment may come from countries where the visitor’s language is different from the native language used at the location of the gaming establishment. As a result, it is desirable to provide a gaming machine environment that uses the native language of the gaming machine user. Unfortunately, due to the proprietary and closed nature of existing architectures, it can be difficult to develop new games, and it is difficult to modify existing proprietary game architectures to include support for multiple languages. As a result, the cost and time associated with updating and adding new games or modifying existing games in gaming networks is relatively high.

[0007] In view of the above-mentioned problems and concerns, there is a need in the art for the present invention.

SUMMARY

[0008] The above-mentioned shortcomings, disadvantages and problems are addressed by the present invention, which will be understood by reading and studying the following specification.

[0009] One aspect of the systems and methods relates to providing a language translation service in a gaming network. The gaming network may comprise gaming machines, service providers, and other entities. The language translation service may provide for translations in one or more languages. The entities participating in the gaming network may implement a Gaming Services Framework using the World Wide Web and internetworking technology. The World Wide Web ("Web" from here on) is a networked information system comprising agents (clients, servers, and other programs) that exchange information. The Web and networking architecture is the set of rules that agents in the system follow, resulting in a shared information space that scales well and behaves predictably.

[0010] The Gaming Services Framework comprises a set of services, protocols, XML schemas, and methods for providing secure gaming system functionality in a distributed, network based architecture. It is intended to be a service-oriented framework for gaming and property management based upon internetworking technology and web services concepts. Specifically, it supports a loosely coupled architecture that consists of software components that semantically encapsulate discrete functionality (self-contained and perform a single function or a related group of functions—the component describes its own inputs and outputs in a way that other software can determine what it does, how to invoke its functionality, and what result to expect). These components are distributed and programmatically accessible (called by and exchange data with other software) over standard internetworking protocols (TCP/IP, HTTP, DNS, DHCP, etc.).

[0011] The present invention describes systems, methods, and computer-readable media of varying scope. In addition to the aspects and advantages of the present invention described in this summary, further aspects and advantages of the invention will become apparent by reference to the drawings and by reading the detailed description that follows.

BRIEF DESCRIPTION OF THE DRAWINGS

[0012] FIG. 1 is a perspective view of an exemplary gaming machine incorporated in the present invention.

[0013] FIG. 2 is a block diagram providing an example of a service-oriented network for distributed management in a gaming environment.

[0014] FIG. 3 is a block diagram providing general description of service-oriented discovery and interaction.
FIG. 4 is a representation of a Gaming Services Protocol Stack according to embodiments of the invention.

FIGS. 5A and 5B are flow diagrams illustrating methods and message flow for a language translation service according to embodiments of the invention.

DETAILED DESCRIPTION

In the following detailed description of exemplary embodiments of the invention, reference is made to the accompanying drawings which form a part hereof, and in which is shown by way of illustration specific exemplary embodiments in which the invention may be practiced. These embodiments are described in sufficient detail to enable those skilled in the art to practice the invention, and it is to be understood that other embodiments may be utilized and that logical, mechanical, electrical and other changes may be made without departing from the scope of the present invention.

Some portions of the detailed descriptions which follow are presented in terms of algorithms and symbolic representations of operations on data bits within a computer memory. These algorithmic descriptions and representations are the ways used by those skilled in the data processing arts to most effectively convey the substance of their work to others skilled in the art. An algorithm is here, and generally, conceived to be a self-consistent sequence of steps leading to a desired result. The steps are those requiring physical manipulations of physical quantities. Usually, though not necessarily, these quantities take the form of electrical or magnetic signals capable of being stored, transferred, combined, compared, and otherwise manipulated. It has been proven convenient at times, principally for reasons of common usage, to refer to these signals as bits, values, elements, symbols, characters, terms, numbers, or the like. It should be borne in mind, however, that all of these and similar terms are to be associated with the appropriate physical quantities and are merely convenient labels applied to these quantities. Unless specifically stated otherwise as apparent from the following discussions, terms such as “processing” or “computing” or “calculating” or “determining” or “displaying” or the like, refer to the action and processes of a computer system, or similar computing device, that manipulates and transforms data represented as physical (e.g., electronic) quantities within the computer system’s registers and memories into other data similarly represented as physical quantities within the computer system’s memory or registers or other such information storage, transmission or display devices.

In the Figures, the same reference number is used throughout to refer to an identical component which appears in multiple Figures. Signals and connections may be referred to by the same reference number or label, and the actual meaning will be clear from its use in the context of the description.

The description of the various embodiments is to be construed as exemplary only and does not describe every possible instance of the invention. Numerous alternatives could be implemented, using combinations of current or future technologies, which would still fall within the scope of the claims. The present invention is directed to a language translation service in a service-oriented framework for gaming networks that allows for the interoperability of the software components (regardless of manufacturer, operating system, or application) reducing the dependence on a closed-system, single vendor solutions and allowing for variety in innovation and competition.

The following detailed description is, therefore, not to be taken in a limiting sense, and the scope of the present invention is defined only by the appended claims.

Operating Environment

FIG. 1 illustrates an exemplary gaming machine 10 in which embodiments of the invention may be implemented. In some embodiments, gaming machine 10 is operable to conduct a wagering game. These wagering games may include reel based games such as video or mechanical slot machine games, card based games such as video poker, video dice games (e.g. a Yahtzee® like dice game) or other types of wagering games typical in the gaming industry. If based in video, the gaming machine 10 includes a video display 12 such as a cathode ray tube (CRT), liquid crystal display (LCD), plasma, or other type of video display known in the art. A touch screen preferably overlays the display 12. In the illustrated embodiment, the gaming machine 10 is an “upright” version in which the display 12 is oriented vertically relative to a player. Alternatively, the gaming machine may be a “slant-top” version in which the display 12 is slanted at about a thirty-degree angle toward the player.

The gaming machine 10 includes a plurality of possible credit receiving mechanisms 14 for receiving credits to be used for placing wagers in the game. The credit receiving mechanisms 14 may, for example, include a coin acceptor, a bill acceptor, a ticket reader, and a card reader. The bill acceptor and the ticket reader may be combined into a single unit. The card reader may, for example, accept magnetic cards and smart (chip) cards coded with money or designating an account containing money.

In some embodiments, the gaming machine 10 includes a user interface comprising a plurality of push-buttons 16, the above-noted touch screen, and other possible devices. The plurality of push-buttons 16 may, for example, include one or more “bet” buttons for wagering, a “play” button for commencing play, a “collect” button for cashing out, a help button for viewing a help screen, a “pay table” button for viewing the pay table(s), and a “call attendant” button for calling an attendant. Additional game specific buttons may be provided to facilitate play of the specific game executed on the machine. The touch screen may define touch keys for implementing many of the same functions as the push-buttons. Additionally, in the case of video poker, the touch screen may implement a card identification function to indicate which cards a player desires to keep for the next round. Other possible user interface devices include a keyboard and a pointing device such as a mouse or trackball.

A processor controls operation of the gaming machine 10. In response to receiving a wager and a command to initiate play, the processor randomly selects a game outcome from a plurality of possible outcomes and causes the display 12 to depict indicia representative of the selected game outcome. In the case of slots for example mechanical or simulated slot reels are rotated and stopped to place symbols on the reels in visual association with one or more pay lines. If the selected outcome is one of the winning
outcomes defined by a pay table, the processor awards the player with a number of credits associated with the winning outcome.

[0026] FIG. 2 illustrates an example of a Gaming Service Network 210 comprising a customer data center 218 and a customer property 216. The data center 218 and customer property 216 are connected via a network 220. In some embodiments, network 220 is a public network such as the Internet. However, in alternative embodiments, private networks, including corporate intranets or extranets may be used to connect a data center 218 with one or more properties 216.

[0027] In some embodiments, the Customer Corporate Data Center 218 contains the bulk of the network servers supporting gaming properties owned by the corporation. Major elements of the gaming service network include Auth server 232, Gaming Management Server 236, and Progressive Server 238. In some embodiments, Auth Server 32 provides authentication, authorization and content integrity for client devices attempting to interact with other servers and services in the architecture.

[0028] In some embodiments, the Gaming Management Server 236 includes the following services: Boot Service, Name Service, Time Service, Game Management Service, Game Update Service, Event Management Service, Accounting Service, and Discovery Service.

[0029] In some embodiments, the Progressive Server 238 hosts a value-add service that allows a gaming machine to participate within a progressive gaming offering. Any value-add service can be added or substituted for this server/service. A progressive gaming offering is provided as an example. Other value-add services can be distributed on existing servers or reside on a newly added server.

[0030] The Customer Property 16 contains gaming machines 10, which in some embodiments allow remote updates and configuration through a network interface on the gaming machine. In some embodiments, a Boot Server 234 contains a DHCP service that facilitates the distribution of IP addressing to the gaming machines 10. It should be noted that any device capable of supporting a wagering game could be substituted for gaming machine 10. For example, a personal or laptop computer executing a wagering game may participate in the gaming network using the services described below.

[0031] As noted above, various services may be located throughout the gaming network. In some embodiments of the invention, a set of core operational services may include one or more of the following services:

[0032] Boot Service Provides dynamic IP addressing to devices upon boot (start-up). Typically supported by Dynamic Host Configuration Protocol (DHCP).

[0033] Discovery Service Provides the address information of the server containing the service when prompted by the requestor as well as the service description, binding and location on the server.

[0034] Authentication Service Contains the master Authentication Database. Authenticates the service user before allowing the use of services in the Gaming Services Framework.

[0035] Authorization Service Contains the master Authorization Database. Authorizes the use of services in the Gaming Services Framework by a service requestor.

[0036] Gaming Management Service Provides the ability to configure and monitor gaming machines and other services from a central location.

[0037] Name Service Provides name resolution service to enable machines in a gaming network to refer to each other by name instead of an IP Address. In some embodiments the name service is implemented in part using the Domain Naming System (DNS) protocol.

[0038] Time Service Provides global synchronization of time in the gaming network. This may be implemented by running the Network Time Protocol (NTP) client software on gaming machines.

[0039] In addition to or instead of the core services described above, some embodiments of the invention include one or more of the following services referred to as Basic Gaming Services:

[0040] Accounting Service Provides logging of transaction records for billing and general tracking purposes.

[0041] Event Management Service Logs events occurring at client and server machines.

[0042] Game Update Service Provides dynamic distribution of new or updated game content to gaming machines.

[0043] Message Director Service This service uses a software-configurable message routing application to facilitate the reliable exchange of data messages among multiple application processes within one or more gaming systems.

[0044] Content Integrity Service This service provides the ability to verify the integrity of software components running in the gaming network. This includes the verification of software versions running on gaming machines, peripherals, services as well the detection of tampering or modification of the software.

[0045] As noted above, a gaming service network may include Value Add Services. These services include participation services and player services. Examples of participation services that may be included in various embodiments of the invention include the following:

[0046] Progressive Service Provides functionality for a gaming machine to participate within a single progressive or multiple progressives.

[0047] Wide Area Disruption Progressive Service This service takes over the processing of wide area progressives at each gaming site in the event that there is no connection with a central system or the connection with the central system is temporarily disabled.

[0048] Mobile Gaming Device GPS Service This service processes the GPS location of gaming machines compared with coordinates of a gaming jurisdiction. Example: players can ride a bus and begin gambling on the bus when the bus crosses into the gaming jurisdiction.
Examples of Player Services that may be included in various embodiments of the invention include:

Player Tracking Service This service provides the operator and player with standard player tracking applications such as monitoring card in/card out transactions to track play and award player points for play, providing targeted promotional compensation to specific players, publishing account status to the player or operator, providing temporary gaming machine locking in order to hold the machine for the player for short periods of time, and providing operators and players an interface and capability for Responsible Gaming Initiatives.

Game Theme Location Service This service provides location information to clients regarding specific games, game themes or vendor brands. The service may publish the information by casino, by area, by city, by state, by region, by country, or by continent depending on the input parameters provided. An example would be to publish where all of the progressive games of a particular theme (e.g., “Monopoly Money”) are located in a particular hotel (e.g., the Reno Hilton) in Reno, Nev.

Personalization Service This service provides the gaming player with a more personalized gaming environment. Example: the player could choose to see text in Chinese, could choose to be reminded of dinner reservation time, could customize machine graphics, or could have a portion of his coin in go to his football club’s progressive.

Cashless Transaction Service This service provides the ability for a player to transfer funds between financial institutions, in-house accounts and gaming machines.

Bonusing Service This service provides the ability for casinos to set up bonus games for a specific gaming machine, carousel of machines or one or more game themes.

Game Service This service is a server-side process that provides the outcome of game play. This service may be used to enable Internet/online gaming.

Advertising Service This service allows the operator to display advertising information to players in multimedia format as well as simple audio and graphic formats.

Property Service This is a group of services that provides the ability for the property management company to integrate with gaming systems. It can provide interaction with functions such as hotel and restaurant reservations.

Language Translation Service This service provides a translation method for players on a networked gaming machine. It may provide translations for one or more languages for the game itself, some of the additional features found on the machine, or the entire feature set of the gaming machine.

Additional details on a language translation service according to embodiments of the invention are provided below.

It should be noted that with the distributed architecture of the Gaming Service Network 210, the above-described services that reside on network servers are not limited to location and can reside anywhere the network supports. For example, it is desirable to consider security and network latency when locating services.

FIG. 3 is a block diagram of a Gaming Services Framework 300 according to various embodiments of the invention. In some embodiments, the Gaming Services Framework 300 includes a set of protocols, XML schemas, and methods for providing gaming system functionality in a distributed, network-based architecture such as the network described above in FIG. 2. In order to participate in such network-based architectures, the participating machines are interconnected via public or private networks that may be wired or wireless networks. Further, devices performing service communication support a common services protocol stack such as the Gaming Services Protocol Stack that is further described below.

The Gaming Services Framework 300 provides for the interaction of several logical elements as depicted in FIG. 3. Logical elements represent the fundamental entities that interact to implement a service. In some embodiments, these logical elements include Service Requestor 302, Service Provider 304, and Discovery Agency 306. In general terms, the roles these elements play are as defined in Web Services Architecture—W3C Working (Draft 14 Nov. 2002 and later versions). Further details on these elements are provided below.

Logical elements may reside in a number of different physical devices as part of delivering any service. For example, a Service Provider 304 will typically reside in a slot accounting or player tracking system and the Service Requestor 302 will typically reside in a gaming machine. However, there may be scenarios where it would be advantageous or appropriate for the logical elements to reside in other physical devices. For example, in alternative embodiments a Service Requestor 302 may reside in a slot accounting system.

Service Provider 304 comprises a platform that hosts access to a service 314. A service provider may also be referred to as a service execution environment or a service container. Its role in the client-server message exchange patterns is that of a server.

Service Requestor 302 comprises an application that is looking for and invoking or initiating an interaction with a service such as that provided by service provider 304. Its role in the client-server message exchange patterns is that of a client 312.

Discovery Agency 306 comprises a searchable set of service descriptions where service providers 304 publish their service description(s) 324 and service location(s) 326. The service discovery agency 306 can be centralized or distributed. A discovery agency 306 can support both patterns where service descriptions 322 are sent to discovery agency 306 and patterns where the discovery agency 306 actively inspects public service providers 304 for service descriptions 322. Service requesters 302 may find services and obtain binding information (in the service descriptions 324) during development of static binding, or during execution for dynamic binding. In some embodiments, for
example in statically bound service requesters, the service discovery agent may be an optional role in the framework architecture, as a service provider 304 can send the service description 322 directly to service requester 302. Likewise, service requestors 302 can obtain a service description 324 from other sources besides a discovery agency 306, such as a local file system, FTP site, URL, or WSDL document.

FIG. 4 provides a block diagram of a Gaming Services Protocol Stack 400 according to embodiments of the invention. In some embodiments, the protocol stack includes core layers that define basic services communication and transport, and are typically implemented uniformly. Higher layers that define strategic aspects of gaming processes are also described below. FIG. 4 illustrates both the widely implemented core layers and in addition illustrates the higher gaming services oriented layers of the protocol stack.

Core Layers of the Gaming Services Protocol Stack 400

In some embodiments, the gaming services framework utilizes common Internet protocols, which may include web services protocols. Although not specifically tied to any transport protocol, it is desirable to build the gaming services on ubiquitous Internet connectivity and infrastructure to ensure nearly universal reach and support. In some embodiments, gaming services will take advantage of Ethernet 405 or 406, Transmission Control Protocol (TCP) 408, Internet Protocol (IP) 407, User Datagram Protocol (UDP) 409, HyperText Transfer Protocol (HTTP) 410, HyperText Transfer Protocol Secure/Secure Socket Layer (HTTPS/SSL) 411, Lightweight Directory Access Protocol (LDAP) 412, Domain Naming System (DNS) 413, and Dynamic Host Configuration Protocol (DHCP) 414 layers in the protocol stack 400. Those of skill in the art will appreciate that other protocol layers performing equivalent functionality may be substituted for those described above and are within the scope of the present invention.

In some embodiments, service request and response data are formatted using Extensible Markup Language (XML). XML is a widely accepted format for exchanging data and its corresponding semantics. XML is a fundamental building block used in layers above the Common Internet Protocols. In some embodiments, the Gaming Services Protocol Stack 400 incorporates this protocol in accordance with the World Wide Web Consortium (W3C) XML Working Group’s XML specification. However, those of skill in the art will appreciate that other data exchange formats may be substituted for XML, and such formats are within the scope of the present invention.

In some embodiments of the invention, the gaming service protocol stack 400 utilizes the Simple Object Access Protocol (SOAP). SOAP is a protocol for messaging and RPC (Remote Procedure Call) style communication between applications. SOAP is based on XML and uses common Internet transport protocols like HTTP to carry data. SOAP may be used to convey a model of an envelope request and response messages encoded in XML. SOAP messaging may be used to exchange any kind of XML information. SOAP is used in some embodiments as the basic standard for carrying service requests/responses between service users and providers. SOAP has been submitted to the World Wide Web Consortium (W3C) standards body as recommendation documents (versions 1.1 and 1.2) and will likely emerge as “XML Protocol (XP).”

Higher Layers of the Gaming Services Protocol Stack 400

In some embodiments, the gaming services protocol stack includes a Web Services Description Language (WSDL) and a Universal Description, Discovery, and Integration (UDDI). WSDL 417 comprises a description of how to connect to a particular service. In some embodiments, WSDL 417 is based on XML. A WSDL 417 description abstracts a particular service’s various connection and messaging protocols into a high-level bundle and forms an element of the UDDI 418 directory’s information. WSDL 417 is similar to CORBA or COM IDL in that WSDL 417 describes programmatic interfaces. WSDL 417 is typically independent of the underlying service implementation language or component model, and focuses on an abstract description. The Gaming Services Protocol Stack 400 incorporates this description in accordance with the World Wide Web Consortium (W3C) Web Services Description Language (WSDL) 1.1—W3C Note 15 Mar. 2001 and later versions.

In some embodiments, UDDI 418 represents a set of protocols and a public directory for the registration and real-time lookup of services. UDDI 418 enables an entity such as a company to publish a description of available services to the registry, thereby announcing itself as a service provider. Service users can send requests conforming to the UDDI 418 schema as SOAP messages to the service registry to discover a provider for services. Some embodiments of the present invention may utilize UDDI Version 3, released in July of 2002 and later versions. Further development of UDDI 418 is managed under the auspices of the OASIS (Organization for the Advancement of Structured Information Standards) UDDI Specifications technical committee.

Returning to FIG. 3, the service requesters and service providers use the above-described protocol stack to perform service interactions with one another. The service interactions include publish 330, discover (find) 332, and interact 334.

Publish interaction 330 provides a mechanism for a service to be made accessible by other entities in the gaming network environment. In order to be accessible, a service needs to publish its description such that the requester can subsequently find it. Where it is published can vary depending upon the requirements of the application. A service description 322 can be published using a variety of mechanisms known in the art. The various mechanisms used by the varying embodiments of the invention provide different capabilities depending on how dynamic the application using the service is intended to be. The service description may be published to multiple service registries using several different mechanisms. The simplest case is a direct publish. A direct publish means the service provider sends the service description directly to the service requester. In this case the service requester may maintain a local copy of the service description 322.

Another means of publishing service descriptions utilized in alternative embodiments of the invention is through a UDDI registry. There are several types of UDDI...
registries known in the art that may be used depending on
the scope of the domain of Web services published to it. When
publishing a Web service description to a UDDI registry, it is desirable to consider the business context and
taxonomies in order for the service to be found by its
potential service consumers. Examples of UDDI registries
used in the gaming service architecture of various embo-
iments of the invention are Internal Enterprise Application
UDDI registry, Portal UDDI registry, and Partner Catalog
UDDI registry.

[0078] An Internal Enterprise Application UDDI registry
may be used in some embodiments for gaming services
intended for use within an organization for internal enter-
prise applications integration. For example, all services that
provide gaming and gaming management to devices within
a casino or casino organization may be published to an
Internal Enterprise Application UDDI registry.

[0079] A Portal UDDI registry may be used in some
embodiments for gaming services that are published by a
company for external partners to find and use. Aportal
UDDI registry typically runs in the service provider’s envi-
nronment outside of a firewall or in a DMZ (de-militarized
zone) between firewalls. This kind of private UDDI registry
generally contains only those service descriptions that a
company wishes to provide to service requestors from
external partners through a network. For example, these
services may be used to provide online gaming to customers
connecting through the World-Wide Web.

[0080] A Partner Catalog UDDI registry may be used in
some embodiments for gaming services to be used by a
particular company. The Partner Catalog UDDI registry can
be thought of as a rolodex like UDDI registry. A Partner
Catalog UDDI registry is typically located on a computer or
gaming machine behind a firewall. This kind of private
UDDI registry typically contains approved, tested, and valid
service descriptions from legitimate (e.g., authorized) busi-
ness partners. The business context and metadata for these
services can be targeted to the specific requester. In some
embodiments, this type of registry may be used for inter-
casino services as well as interactions between casinos and
other types of organizations such as regulators and financial
institutions. It is desirable that an appropriate authoriza-
tion and qualification procedure be in place to ensure that only
approved services are published to service repositories.

[0081] In the discover interactions 332 (also referred to as
find interactions), the service requestor retrieves a service
description directly or queries the registry for the type of
service required. It then processes the description in order
eto be able to bind and invoke it.

[0082] As with publishing service descriptions, acquiring
service descriptions may vary depending on how the service
description is published and how dynamic the service appli-
cation is meant to be. In some embodiments, service request-
ors may find Web services during two different phases of an
application lifecycle—design time and run time. At design
time, service requesters search for web service descriptions
by the type of interface they support. At run time, service
requesters search for a web service based on how they
communicate or qualities of service advertised.

[0083] With the direct publish approach noted above, the
service requestor may cache the service description at design
time for use at runtime. The service description may be
statically represented in the program logic, stored in a file,
or in a simple, local service description repository.

[0084] Service requestors can retrieve a service descrip-
tion at design time or runtime from a Web page (URL), a
service description repository, a simple service registry or a
UDDI registry. The look-up mechanism typically supports a
query mechanism that provides a find by type of interface
capability (for example, based on a WSDL template), the
binding information (i.e., protocols), properties (such as
QoS parameters), the types of intermediaries required, the
taxonomy of the service, business information, etc.

[0085] The various types of UDDI registries, including
those described above, have implications on the number of
runtime binding services that can choose from, policy for choos-
ing one among many, or the amount of pre screening that
will be done by the requestor before invoking the service.
Service selection can be based on binding support, historical
performance, quality of service classification, proximity, or
load balancing. It is desirable that an appropriate authoriza-
tion and qualification procedure be in place to ensure that only
approved services are published to service repositories.

[0086] Once a service description is acquired, the service
requestor will need to process it in order to invoke the
service. In some embodiments, the service requestor uses the
service description to generate SOAP requests or programming
language specific proxies to the service. The genera-
tion of such requests can be done at design time or at runtime
to format an invocation to the service. Various tools can be
used at design time or runtime to generate programming
language bindings from interface descriptions, such as
WSDL documents. These bindings present an API (Applier
Program Interface) to the application program and encapsulate
the details of the messaging from the application.

[0087] After a service has been published 330 and discover-
ed 332, the service may be invoked so that a service
requestor and service provider may interact 334. In the
interact operation 334, the service requestor invokes or
initiates an interaction with the service at runtime using the
binding details in the service description 322 to locate,
contact, and invoke the service. Examples of service inter-
actions 334 include: single message one way, broadcast from
requester to many services, a multi message conversation, or
a business process. Any of these types of interactions can be
synchronous or asynchronous requests.

[0088] In some embodiments of the invention, security
mechanisms may be used to secure the Gaming Services
Framework 300. Securing the Gaming Services Framework
typically involves providing facilities for ensuring the integ-
rity and confidentiality of the messages and for ensuring that
a service acts only on requests in messages that express the
claims required by policies. Examples of such mechanisms
used in various embodiments of the invention include IPsec
and SSL/TLS, which provide network and transport layer
security between two endpoints. However, when data is
received and forwarded on by an intermediary beyond the
transport layer both the integrity of data and any security
information that flows with it maybe lost. This forces any
upstream message processors to rely on the security evalua-
tions made by previous intermediaries and to completely
trust their handling of the content of messages. Thus it is
It is desirable to include security mechanisms that provide end-to-end security. It is also desirable that such mechanisms be able to leverage both transport and application layer security mechanisms to provide a comprehensive suite of security capabilities.

Language Translation Service

In general, the various embodiments of the invention implement a language translation service that provides functionality for a client application to obtain translated data for one or more specified languages. For example, a game application may let the user choose a language like Spanish, Chinese or Japanese to view and play a game. Another example is a player service application shows the player’s preferred language for the player’s credits, bonus, hotel or dinner reservations, or any player account activities. A client application registers with the language translation service. Once registered, this client application can request one or more translated files from Language Translation Service according to client application name, version number and device type. In some embodiments, each translated file contains name attributes and translated values for one language in XML structure. These translated files typically contain commonly used strings that can be stored at the client side. Client applications only need to request them occasionally. For real time or less frequent used data, the client may request them upon a request by a user. In this case, a simple XML string translation request to the service can be done. This is desirable because in some embodiments, network traffic may be reduced. Those of skill in the art will appreciate the name attributes and translated values may be maintained in alternative manners instead of or in addition to XML. For example, other file structures besides XML may be used in alternative embodiments of the invention. Further, a database such as a relational database may be used to store name attributes and translated values.

FIGS. 5A and 5B are flow diagrams illustrating methods for providing a language translation service in a gaming network according to embodiments of the invention. The methods may be performed within an operating environment such as that described above with reference to FIGS. 1-4. The methods to be performed by the operating environment constitute computer programs made up of computer-executable instructions. Describing the methods by reference to a flow diagram enables one skilled in the art to develop such programs including such instructions to carry out the methods on suitable computers (the processor of the computer executing the instructions from machine-readable media such as RAM, ROM, CD-ROM, DVD-ROM, flash memory etc.). The methods illustrated in FIGS. 5A and 5B are inclusive of the acts performed by an operating environment executing an exemplary embodiment of the invention.

FIG. 5A is a flow diagram illustrating a method for providing a language translation service in a service-oriented gaming network. In the detailed description of the method below, particular program method names may be provided for particular embodiments of the invention. It should be noted that such names are convenient labels for the method and are exemplary in nature. The present invention is not limited to any functionality that may be implied by the name.

The method begins by publishing the availability of a language translation service on a gaming network (block 510). In some embodiments, the service is registered by sending a description (e.g. in WSDL) of the service to the discovery agency. The discovery agency adds the service description to its UDDI repository. At this point the language translation service is available for discovery by interested parties.

Next, in some embodiments, a client/service requester makes UDDI calls to the discovery agency to find a language translation service (block 512). The discovery agency returns the service description and location information to the requester.

Next, a client/service requester registers with the service provider (block 514). In some embodiments, this is accomplished by invoking a languageTranslationService-Register method on the language translation service. In some embodiments, this method call is a SOAP call and includes parameters that identify the client and provide authentication information to the language translation service provider. The language translation service provider may verify that the client is authorized to receive translated data or other language translation services before successfully registering the client. When the client is done using the service, it may invoke the Language Translation Service Deregister method on the language translation service.

Finally, a client (e.g. a gaming machine, a service requester or a service provider) can invoke the language translation service to process a request (block 516). In some embodiments, the following methods of the language translation service may be invoked (the methods may be implemented as SOAP calls):

languageTranslationServiceGetFile The client makes this call with parameters of languages, application name, version number and device type to the service to get translated file(s) or data sets. Each file contains name/value pairs for one language.

languageTranslationServiceGetString The client makes this call with parameters of a language and a string containing name attributes to be translated. Language translation service returns the translated string.

In some embodiments, the name attribute string and the translated string are XML strings.

FIG. 5B illustrates a method according to an embodiment of the invention for providing a language translation service to a client in a gaming machine network. In particular, FIG. 5B illustrates an exemplary usage scenario involving an exemplary message sequence 500 that describes how a client such as gaming machine 501 and a language translation service interact between themselves and other components of a gaming network such as discovery service 503, authorization database 504 and localizing database 505. Message sequence 500 is but one example of a message sequence. Those of skill in the art will appreciate that other message sequences for other types of requests are within the scope of the invention. Additional information for each message is provided below as defined by the reference number in FIG. 5B.

At 521 the language translation service 502 is deployed and saves its binding information to the Discovery Service 503 (UDDI Registry).
At 522 the Discovery Service 503 authenticates the language translation service 502 with an Authentication/Authorization Database 504 (e.g. using LDAP, RADIUS, etc.).

At 523 the Authentication/Authorization Database 504 successfully authenticates the language translation service 502 (e.g. using LDAP, RADIUS, etc.).

At 524 the Discovery Service 503 returns a bindingDetail information element to the language translation service 502 (UDDI). The language translation service 502 is now ready to accept requests for translation related services from clients such as gaming machine, game server or other components of a gaming network.

At 525 a client such as Gaming Machine 501 communicates with the Discovery Service 503 to find the location of a language translation service (e.g. using UDDI).

At 526 the Discovery Service 503 returns with a list of possible language translation services (e.g. using UDDI).

At 527 the client Gaming Machine 501 chooses one of the available language translation services (using some suitable algorithm) and requests the binding information of that instance of the chosen language translation service 502 (e.g. using UDDI).

At 528 the Discovery Service 503 returns the binding information to the client Gaming Machine 501 (e.g. using UDDI).

At 529 the client Gaming Machine 501 registers with the language translation service 502 (e.g. using SOAP).

At 530 the language translation service 502 authenticates the client Gaming Machine 501 with the Authentication/Authorization Database 504 (e.g. using LDAP, RADIUS, etc.).

At 531 the Authentication/Authorization Database 504 successfully authenticates the client Gaming Machine 501 (e.g. using LDAP, RADIUS, etc.).

At 532 the language translation service 502 returns a successful response to the client Gaming Machine 501 (e.g. using SOAP).

At 533 the client Gaming Machine 501 calls for an application, version and device type specific data with multiple languages from language translation service 502. The input parameters can include languages to translate to, client application name, version number and device type (e.g. using SOAP).

At 534 the language translation service 502 calls the Localizing Database 505 for the specified application name, version number, device type and language. The Localizing Database 505 is a repository containing translated data for different languages. In some embodiments, it categorizes this data by application name, version number, device type and so on.

At 535 the Localizing Database 505 returns a result set of translated data for the requested application, version, device type and languages.

At 536 the language translation service 502 builds one or more files from the data returned by the Localizing Database 505.

At 537 the language translation service 502 returns multiple translated files that are specific to an application, version and device type (e.g. using SOAP). Translated files typically differ in the language used. Alternatively, the service can return just one file with different language xml sections inside.

Conclusion

Systems and methods providing a language translation service in a service-oriented gaming network environment have been disclosed. Although specific embodiments have been illustrated and described herein, it will be appreciated by those of ordinary skill in the art that any arrangement which is calculated to achieve the same purpose may be substituted for the specific embodiments shown. This application is intended to cover any adaptations or variations of the present invention.

The terminology used in this application is meant to include all of these environments. It is to be understood that the above description is intended to be illustrative, and not restrictive. Many other embodiments will be apparent to those of skill in the art upon reviewing the above description. Therefore, it is manifestly intended that this invention be limited only by the following claims and equivalents thereof.

What is claimed is:

1. A method for providing a language translation service in a gaming network including gaming machines, the method comprising:
   publishing an availability of the language translation service on the gaming network;
   receiving a discovery request for the language translation service;
   registering by a gaming client with the language translation service; and
   processing one or more service requests between the gaming client and the language translation service, said service requests conforming to an internetworking protocol.

2. The method of claim 1, wherein the language translation service comprises a web service.

3. The method of claim 2, wherein the service request is formatted according to a service description language.

4. The method of claim 3, wherein the service description language is a Web Services Description Language (WSDL).

5. The method of claim 2, wherein the language translation service is registered in a UDDI registry.

6. The method of claim 1, wherein the gaming client comprises a gaming machine.

7. The method of claim 1, wherein the gaming client comprises a service provider.

8. The method of claim 1, wherein the service request comprises a request by the gaming client for a translation of a text string from the language translation service.

9. The method of claim 8, wherein the text string comprises an XML (extensible Markup Language) formatted text string.

10. The method of claim 1, wherein the service request comprises a request by the gaming client for a set of translations from the language translation service.
11. The method of claim 10, wherein the language translation service provides the set of translations in a file.
12. The method of claim 11, wherein the file conforms to an XML format.
13. The method of claim 1, further comprising authenticating the gaming client to determine if the gaming client is authorized to receive language translations.
14. A gaming network system, the gaming network system comprising:
   a gaming client communicably coupled to the gaming network; and
   a language translation service communicably coupled to the gaming network and operable to:
   publish an availability of the language translation service on the gaming network;
   register a gaming client with the language translation service; and
   process one or more service requests between the gaming client and the language translation service, said service requests conforming to an internetworking protocol.
15. The gaming network system of claim 14, wherein the language translation service comprises a web service.
16. The gaming network system of claim 14, wherein the service request is formatted according to a service description language.
17. The gaming network system of claim 16, wherein the service description language is a Web Services Description Language (WSDL).
18. The gaming network system of claim 14, wherein the language translation service is registered in a UDDI registry.
19. The gaming network system of claim 14, wherein the gaming client comprises a gaming machine.
20. The gaming network system of claim 14, wherein the gaming client comprises a service provider in the gaming network.
21. The gaming network system of claim 14, wherein the service request comprises a request by the gaming client for a translation of a text string from the language translation service.
22. The gaming network system of claim 21, wherein the text string comprises an XML (eXtensible Markup Language) formatted text string.
23. The gaming network system of claim 14, wherein the service request comprises a request by the gaming client for a set of translations from the language translation service.
24. The gaming network system of claim 23, wherein the language translation service provides the set of translations in a file.
25. The gaming network system of claim 24, wherein the file conforms to an XML format.
26. The gaming network system of claim 14, further comprising an authentication service operable to authenticate the gaming client to determine if the gaming client is authorized to receive language translations.
27. The gaming network system of claim 26, wherein the authentication service includes a RADIUS authentication service.
28. The gaming network system of claim 26, further comprising a localizing database operable to store a set of one or more translations.
29. The gaming network system of claim 28, wherein the set of one or more translations are organized by one or more fields selected from the group consisting of application name, version number, device type, language type.

* * * * *