

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
1 February 2007 (01.02.2007)

PCT

(10) International Publication Number
WO 2007/013930 A1

(51) International Patent Classification:
G06F 9/44 (2006.01)

(21) International Application Number:
PCT/US2006/028063

(22) International Filing Date: 19 July 2006 (19.07.2006)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
11/186,338 21 July 2005 (21.07.2005) US

(71) Applicant (for all designated States except US): **COM-
PUTER ASSOCIATES THINK, INC.** [US/US]; One CA
Plaza, Islandia, NY 11749 (US).

(72) Inventors; and

(75) Inventors/Applicants (for US only): **SAUNDERS,
Roger, C.** [GB/GB]; 85B Oxford Road, Abingdon, Oxon
OX14 2AB (GB). **THAIN, William, R.** [GB/GB]; 77
Birrell Gardens, Murieston, Livingston, West Lothian
EH54 9LF (GB).

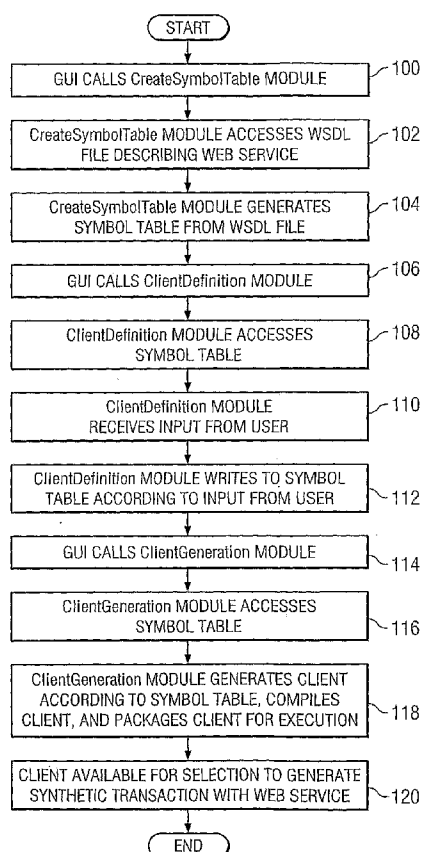
(74) Agent: **PEDERSEN, Luke, K.**; Baker Botts L.L.P., 2001
Ross Avenue, Suite 600, Dallas, TX 75201 (US).

(81) Designated States (unless otherwise indicated, for every
kind of national protection available): AE, AG, AL, AM,
AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN,
CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI,
GB, GD, GE, GH, GM, HN, HR, HU, ID, IL, IN, IS, JP,
KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT,
LU, LV, LY, MA, MD, MG, MK, MN, MW, MX, MZ, NA,
NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RS, RU, SC,
SD, SE, SG, SK, SL, SM, SY, TJ, TM, TN, TR, TT, TZ,
UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States (unless otherwise indicated, for every
kind of regional protection available): ARIPO (BW, GH,
GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM,
ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM),
European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI,
FR, GB, GR, HU, IE, IS, IT, LT, LU, LV, MC, NL, PL, PT,
RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA,
GN, GQ, GW, ML, MR, NE, SN, TD, TG).

[Continued on next page]

(54) Title: GENERATING ONE OR MORE CLIENTS FOR GENERATING ONE OR MORE SYNTHETIC TRANSACTIONS
WITH ONE OR MORE WEB SERVICE OPERATIONS



(57) Abstract: In one embodiment, a method for generating one or more synthetic transactions with one or more web service operations includes accessing a Web Services Description Language (WSDL) file describing a web service and, according to the WSDL file, generating a symbol table for describing a client for generating one or more synthetic transactions with the web service. The method' also includes receiving input from a user specifying one or more operations of the web service for invocation, an order for invoking the operations of the web service, and one or more values of one or more parameters of the operations. The method also includes incorporating the input from the user into the symbol table and generating the client according to the symbol table.

WO 2007/013930 A1

**Declarations under Rule 4.17:**

- as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))
- as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(iii))
- before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments

Published:

- with international search report

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

GENERATING ONE OR MORE CLIENTS FOR GENERATING ONE OR MORE
SYNTHETIC TRANSACTIONS WITH ONE OR MORE WEB SERVICE
OPERATIONS

TECHNICAL FIELD OF THE INVENTION

This invention relates in general to web services and more particularly to generating one or more clients for generating one or more synthetic transactions with one or more web service operations.

5

BACKGROUND OF THE INVENTION

Code for interacting with a web service to measure and analyze performance of the web service is typically hand written by a programmer. Writing such code typically requires, on the part of the programmer, significant experience with and knowledge of programming and significant knowledge of a Web Services Description Language (WSDL) file describing the web service.

10

SUMMARY OF THE INVENTION

Particular embodiments of the present invention may reduce or eliminate problems and disadvantages associated with web services.

15

In one embodiment, a method for generating one or more synthetic transactions with one or more web service operations includes accessing a WSDL file describing a web service and, according to the WSDL file, generating a symbol table for describing a client for generating one or more synthetic transactions with the web service. The method also includes receiving input from a user specifying one or more operations of the web service for invocation, an order for invoking the operations of the web service, and one or more values of one or more parameters of the operations. The method also includes incorporating the input from the user into the symbol table and generating the client according to the symbol table.

20

Particular embodiments of the present invention provide one or more technical advantages. As an example, particular embodiments facilitate automatic generation of

25

clients for generating synthetic transactions with web services. Particular embodiments facilitate generation of clients for generating synthetic transactions with web services by users lacking significant experience with or knowledge of programming. Particular embodiments facilitate generation of clients for generating synthetic transactions with web services by users lacking significant knowledge of WSDL files describing the web services. Particular embodiments are encodeable in application server deployment files and deployable to applications servers, such as, for example, JBOSS application servers. Particular embodiments facilitate generation of clients for generating synthetic transactions with web services that are capable of invoking sequences of operations of web services.

Particular embodiments may provide all, some, or none of these technical advantages. Particular embodiments may provide one or more other technical advantages, one or more of which may be readily apparent to a person having ordinary skill in the art from the figures, descriptions, and claims herein.

BRIEF DESCRIPTION OF THE DRAWINGS

To provide a more complete understanding of the present invention and features and advantages thereof, reference is made to the following description, taken in conjunction with the accompanying drawings, in which:

FIGURE 1 illustrates an example system for generating one or more clients for generating one or more synthetic transactions with one or more web service operations;

FIGURE 2 illustrates an example web services distributed management (WSDM) tool in the system illustrated in FIGURE 1; and

FIGURE 3 illustrates an example method for generating one or more clients for generating one or more synthetic transactions with one or more web service operations.

DESCRIPTION OF EXAMPLE EMBODIMENTS

FIGURE 1 illustrates an example system 10 for generating one or more clients for generating one or more synthetic transactions with one or more web service operations. System 10 includes a web server 12, a third party server 14, and one or more user systems 16. A network 18 couples web server 12, third party server 14, and user systems 16 to

each other. The present invention contemplates any suitable network 18. As an example and not by way of limitation, network 18 may include a local area network (LAN), a metropolitan area network (MAN), a wide area network (WAN), a portion of the Internet, or another network 18 or a combination of two or more such networks 18.

5 Web server 12 includes a hardware, software, or embedded logic component or a combination of two or more such components hosting or otherwise supporting one or more websites 20. Web server 12 includes one or more websites 20. In particular embodiments, one or more applications servers support web server 12. Websites 20 hosted at web server 12 are accessible to one or more users at one or more user systems 16 via network 18.

10 Each website 20 includes one or more web pages 22. Reference to a website 20 encompasses one or more websites 20, one or more web pages 22, or both, where appropriate. Reference to a web page 22 encompasses one or more websites 20, one or more web pages 22, or both, where appropriate. In particular embodiments, a user at a user system 16 enters a particular uniform resource locator (URL) or other electronic

15 address to establish a connection to web server 12 and access a particular web page 22 associated with a particular website 20 hosted at web server 12. Server 12 communicates web page 22 to user system 16 via network 18, and user system 16 displays web page 22 to the user. In particular embodiments, to access another web page 22, the user at user system 16 enters another URL to access other web page 22. As an example and not by

20 way of limitation, the user may select an icon on web page 22 linking to other web page 22 and, in response, web server 12 may communicate other web page 22 to user system 16 for display to the user. The user at user system 16 may enter multiple URLs to navigate across multiple web pages 22, according to particular needs. In particular embodiments, one or more web pages 22 at web server 12 require authentication from a user before web server

25 12 will provide the user access to web pages 22.

 Web server 12 also includes a WSDM tool 24. WSDM tool 24 includes a hardware, software, or embedded logic component or a combination of two or more such components for monitoring web services provided through web pages 22 at web server 12. As an example and not by way of limitation, WSDM tool 24 may include functionality for

30 automatically discovering and cataloging web services added to web pages 22 at web

server 12, monitoring performance of operations of the web services and otherwise gathering information on the web services, and, in the event of one or more problems with one or more of the web services, notifying an administrator of the problems. An operation of a web service includes one or more operations of the web service. An administrator system 26 includes a hardware, software, or embedded logic component or a combination of two or more such components enabling one or more administrators to provide input to and receive output from web server 12 for purposes of operating web server 12. As an example and not by way of limitation, administrator system 26 may include a desktop computer system, a laptop computer system, a personal digital assistant (PDA), or another computer system or a combination of two or more such computer systems coupled to web server 12 via one or more links that may each include one or more wires in one or more circuit boards, one or more internal or external buses, one or more wireless links, one or more optical links, one or more wireline links, one or more LANs, one or more MANs, one or more WANs, one or more portions of the Internet, or a combination of two or more such links, according to particular needs. Administrator system 26 includes one or more administrator systems 26.

Third party server 14 includes a hardware, software, or embedded logic component or a combination of two or more such components hosting or otherwise supporting one or more web services. Third party server 14 includes one or more third party servers 14. In particular embodiments, a first entity operates web server 12 and a second entity independent of the first entity operates third party server 14. Example entities that run third party servers 14 hosting or otherwise supporting web services include MICROSOFT and AMAZON. In particular embodiments, web services reside at one or more application servers that possibly support one or more web servers. One or more WSDL files 28 at third party server 14 describe the web services. In particular embodiments, one WSDL file 28 describes one web service. A WSDL file 28 includes one or more WSDL files 28, and a web service includes one or more web services. The present invention contemplates any suitable web services providing any suitable functionalities. As an example and not by way of limitation, a web service may provide functionality for generating a map according to input from a user at a user system 16. A user at user system 16 may access a web page

22 at web server 12 providing access to the web service, enter an address into one or more text boxes or other components of web page 22 for providing input to the web service, and select an icon on web page 22 requesting a map of an area including the address. In response to the user selecting the icon, web server 12 may generate a request for the map and communicate the request to third party server 14. The request may include the address entered by the user and possibly other input from the user.

In response to the request, third party server 14 may access a WSDL file 28 at third party server 14 describing the web service for generating the requested map and, using WSDL file 28 and the input from the user, generate the requested map. Third party server 14 may then communicate the requested map to web server 12, and web server 12 may communicate the requested map to user system 16 for display to the user. In particular embodiments, instead of third party server 14 communicating the requested map to web server 12 for communication to user system 16, third party server 14 may communicate the requested map directly to user system 16 for display to the user. In particular embodiments, instead of third party server 14 generating the requested map in response to the request from web server 12 and communicating the requested map to web server 12 or user system 16, third party server 14 may communicate an instance of WSDL file 28 describing the web service for generating the requested map to web server 12. Web server 12 may then generate the requested map at web server 12 using WSDL file 28 and the input from the user and communicate the requested map to user system 16 for display to the user.

Although particular web services are illustrated and described, the present invention contemplates any suitable web services. Although particular interactions among particular components of system 10 for delivering particular web services to particular users are described and illustrated, the present invention contemplates any suitable interactions among any suitable components of system 10 delivering any suitable web services to any suitable users.

A user system 16 includes a hardware, software, or embedded logic component or a combination of two or more such components enabling one or more users to communicate with one or more of web server 12, third party 14, and one or more other user systems 16,

as described above. As an example and not by way of limitation, a user system 16 may include a desktop computer system, a laptop computer system, a personal digital assistant (PDA), or another computer system or a combination of two or more such computer systems coupled to web server 12 via one or more links that may each include one or more
5 wires in one or more circuit boards, one or more internal or external buses, one or more wireless links, one or more optical links, one or more wireline links, one or more LANs, one or more MANs, one or more WANs, one or more portions of the Internet, or a combination of two or more such links, according to particular needs. A user system 16 includes one or more user systems 16.

10 FIGURE 2 illustrates and example WSDM tool 24. WSDM tool 24 includes a synthetic transaction module 32 that includes a hardware, software, or embedded logic component or a combination of two or more such components for invoking operations of web services. In particular embodiments, to invoke an operation of a web service, synthetic transaction module 32 generates a client including one or more software
15 components for invoking the operation. In particular embodiments, a client is a JAVA class including one or more calls to one or more operations of a web service according to one or more values of one or more parameters. A client includes one or more clients. In particular embodiments, a client is a managed bean (MBean) that is compliant with Java Management Extensions (JMX) and deployable to an application server that is also
20 compliant with JMX (such as, for example, a JBOSS application server). Reference to a "synthetic transaction" encompasses invocation of one or more operations of a web service, where appropriate. In particular embodiments, a server-side observer does not trap a synthetic transaction with a web service before the synthetic transaction reaches the web service. In particular embodiments, to use synthetic transaction module 32 to invoke
25 operations of web services, a user at administrator system 26 need not have significant experience with or knowledge of programming. In particular embodiments, to use synthetic transaction module 32 to invoke operations of web services, a user at administrator system 26 need not have significant knowledge of WSDL files 28 describing the web services.

In particular embodiments, synthetic transaction module 32 resides in an application server deployment file, which enables deployment of synthetic transaction module 32 to an application server (such as, for example, a JBOSS application server). In particular embodiments, to facilitate a synthetic transaction, synthetic transaction module 32 executes three phases—CreateSymbolTable, ClientDefinition, and ClientGeneration—that communicate with each other via one or more serialized JAVA classes, one or more deserialized JAVA classes, or both. Reference to a “symbol table” encompasses JAVA classes used for communicating between or among such phases, where appropriate. In particular embodiments, a symbol table is a serialized tree of JAVA objects that represents a parsed WSDL file 28 and a sequence of operations of a web service described in WSDL file 28. In particular embodiments, a graphical user interface (GUI) controls initiation of each phase, which allows selection of a WSDL file 28 (or a symbol table) and facilitates scheduling of synthetic transactions with a web service described in the WSDL file 28 (or the symbol table) for execution.

In particular embodiments, during CreateSymbolTable, synthetic transaction module 32 parses a WSDL file 28. Such parsing includes generating code for calling a web service described in WSDL file 28 and constructing a symbol table according to WSDL file 28 for input to ClientDefinition. In particular embodiments, during ClientDefinition, a user at administrator system 26 provides input specifying one or more operations of the web service for invocation, an order for invoking the operations, and one or more values of one or more parameters of the operations. Synthetic transaction module 32 updates the symbol table according to the input from the user. In particular embodiments, during ClientGeneration, synthetic transaction module 32 generates code that implements the client described in the symbol table, compiles the client, and packages the client for execution. The client includes one or more calls to the operations specified by the user according to the values of the parameters specified by the user in the order specified by the user. After ClientGeneration, the client is available for selection to generate one or more synthetic transactions. In particular embodiments, a client generated during ClientGeneration is capable of invoking multiple operations of a web service in a specified order. In some such embodiments, the client is capable of using output from one

or more first operations earlier in the sequence to invoke one or more second operations later in the sequence. In some such embodiments, the client is capable of receiving, during the sequence, input from a user at administrator system 26 and using the input to invoke one or more operations in the sequence.

5 Synthetic transaction module 32 includes a CreateSymbolTable module 34, a ClientDefinition module 36, and a ClientGeneration module 38. In particular embodiments, CreateSymbolTable module 34, ClientDefinition module 36, and ClientGeneration module 38 are separate from each other. In particular embodiments, one or more components of synthetic transaction module 32 collectively include one or more
10 components of each of two or more of CreateSymbolTable module 34, ClientDefinition module 36, and ClientGeneration module 38. In particular embodiments, CreateSymbolTable module 34 and ClientGeneration module 38 have command line interfaces and a GUI controls CreateSymbolTable module 34, ClientDefinition module 36, and ClientGeneration module 38. In particular embodiments, input to synthetic transaction
15 module 32 includes a WSDL file 28 describing a web service for invocation. In particular embodiments, output from synthetic transaction module 32 includes a .sar file including a test client and generated stubs and associated beans for invoking one or more operations of the web service described in the WSDL file 28. The .sar file also includes a jboss-service.xml file to control loading onto a JBOSS application server. Synthetic transaction
20 module 32 uses a symbol table to pass information from one component of synthetic transaction module 32 to one or more other components of synthetic transaction module 32.

CreateSymbolTable module 34 handles CreateSymbolTable. In particular embodiments, CreateSymbolTable module 34 is an extension of an AXIS WSDL2Java
25 class. Synthetic transaction module 32 uses one or more facilities of AXIS to generate stubs and beans for calling operations of web services described in WSDL files 28. The extended code then processes an AXIS SymbolTable object produced by an AXIS parser to produce a WSDM symbol table, which synthetic transaction module 32 serializes to disk. To invoke CreateSymbolTable module 34, synthetic transaction module 32 specifies
30 a WSDL file 28 and one or more configuration parameters from either a GUI (which is

often the case) or a command line interface (which is often the case during testing and bulk loading of synthetic transaction module 32).

ClientDefinition module 36 handles ClientDefinition. In particular embodiments, to invoke a GUI associated with ClientDefinition module 36, synthetic transaction module 5 32 specifies a symbol table edited by a previous invocation of ClientDefinition module 36 or processed by CreateSymbolTable module 34. The GUI enables a user at administrator system 26 to edit a possibly empty sequence of operations. In particular embodiments such editing includes selecting from operations defined in the symbol table and editing one or more values of one or more parameters of one or more operations in the sequence. The 10 user specifies values of parameters for each operation in the sequence before synthetic transaction module 32 proceeds to ClientGeneration. In particular embodiments, one or more parameters of one or more subsequent operations in the sequence references one or more results of one or more previous operations in the sequence. A deletion or a reordering of one or more portions of the sequence causes synthetic transaction module 32 15 to check whether such references are still valid before synthetic transaction module 32 will permit any actions relying on such references.

ClientGeneration module 38 handles ClientGeneration. In particular embodiments, to invoke ClientGeneration module 38, synthetic transaction module 32 specifies a symbol table processed by ClientDefinition module 36. ClientGeneration module 38 then uses 20 information in a WSDMClient object (which includes both an object describing a client for generating a synthetic transaction and a symbol table) and information in a sequence of WSDMOperations (each of which include an object representing an operation of a web service and one or more parameters of the operation) to write a JAVA source file for calling the operations selected by the user during ClientDefinition. ClientGeneration 25 module 38 also writes a jboss-service.xml file and an MBean interface for the JAVA source file. ClientGeneration module 38 compiles the JAVA source file and, if the compilation is successful, generates a .sar file including the JAVA source file, the jboss-service.xml file, and the MBean interface. If one or more errors occur during the compilation, synthetic transaction module 32 returns information on the errors to one or 30 more programs calling synthetic transaction module 32. In particular embodiments,

ClientGeneration module 38 is invocable from a command line for testing, but normally invoked from ClientDefinition module 36.

FIGURE 3 illustrates an example method for generating one or more clients for generating one or more synthetic transactions with one or more web service operations.

5 The method begins at step 100, where a GUI calls CreateSymbolTable module 34. In particular embodiments, the GUI calls CreateSymbolTable module 34 in response to input from a user at administrator system 26. At step 102, CreateSymbolTable module 34 accesses a WSDL file 28 describing a web service. At step 104, CreateSymbolTable module 34 generates a symbol table from WSDL file 28. At step 106, the GUI calls
10 ClientDefinition module 36. At step 108, ClientDefinition module 36 accesses the symbol table. At step 110, ClientDefinition module 36 receives input from the user at administrator system 26. At step 112, ClientDefinition module 36 writes to the symbol table according to the input from the user at administrator system 26. At step 114, the GUI calls ClientGeneration module 38. At step 116, ClientGeneration module 38 accesses the
15 symbol table. At step 118, ClientGeneration module 38 generates a client according to the symbol table, compiles the client, and packages the client for execution. At step 120, the client is available for selection to generate one or more synthetic transactions with the web service described in WSDL file 28, at which point the method ends.

20 Although particular steps of the method illustrated in FIGURE 3 are illustrated and described as occurring in a particular order, the present invention contemplates any suitable steps of the method illustrated in FIGURE 3 occurring in any suitable order.

Particular embodiments have been used to describe the present invention, and a person having ordinary skill in the art may comprehend one or more changes, substitutions, variations, alterations, or modifications to the particular embodiments used
25 to describe the present invention. The present invention encompasses all such changes, substitutions, variations, alterations, and modifications within the scope of the appended claims.

WHAT IS CLAIMED IS:

1. A system for generating one or more clients for generating one or more synthetic transactions with one or more web service operations, the system comprising:

a first module operable to:

5 access a Web Services Description Language (WSDL) file describing a web service; and

according to the WSDL file, generate a symbol table for describing a client for generating one or more synthetic transactions with the web service;

a second module operable to:

10 receive input from a user indicating:

one or more operations of the web service for invocation;

an order for invoking the operations of the web service; and

one or more values of one or more parameters of the operations;

access the symbol table; and

15 incorporate the input from the user into the symbol table; and

a third module operable to:

access the symbol table; and

generate the client according to the symbol table.

20 2. The system of Claim 1, wherein the first, second, and third modules reside in an application server deployment file deployable to an application server.

3. The system of Claim 2, wherein the application server is a JBOSS application server.

25 4. The system of Claim 1, wherein output from a first one of the operations of the web service is usable as input to a second one of the operations of the web service.

5. The system of Claim 1, wherein:
the first module comprises a CreateSymbolTable module;
the second module comprises a ClientDefinition module;
the third module comprises a ClientGeneration module;

5

6. The system of Claim 1, wherein:
a synthetic transaction module comprises the first, second, and third modules; and
a web services distributed management (WSDM) tool comprises the synthetic
transaction module.

10

7. The system of Claim 1, wherein the WSDL file comprises a WSDL file
from a third party server providing web services.

15

8. The system of Claim 1, wherein the symbol table comprises a serialized
tree of JAVA objects representing a parsed version of the WSDL file and capable of
representing a sequence of the operations of the web service for invocation.

20

9. A method for generating one or more clients for generating one or more
synthetic transactions with one or more web service operations, the method comprising:

accessing a Web Services Description Language (WSDL) file describing a web
service;

according to the WSDL file, generating a symbol table for describing a client for
generating one or more synthetic transactions with the web service;

receiving input from a user indicating:

25

one or more operations of the web service for invocation;

an order for invoking the operations of the web service; and

one or more values of one or more parameters of the operations;

incorporating the input from the user into the symbol table; and

generating the client according to the symbol table;

10. The method of Claim 9, executable from an application server deployment file deployable to an application server.

11. The method of Claim 10, wherein the application server is a JBOSS application server.

12. The method of Claim 9, wherein output from a first one of the operations of the web service is usable as input to a second one of the operations of the web service.

14. The method of Claim 9, executable by a synthetic transaction module in a web services distributed management (WSDM) tool.

15. The method of Claim 9, wherein the WSDL file comprises a WSDL file from a third party server providing web services.

16. The method of Claim 9, wherein the symbol table comprises a serialized tree of JAVA objects representing a parsed version of the WSDL file and capable of representing a sequence of the operations of the web service for invocation.

17. Logic for generating one or more clients for generating one or more synthetic transactions with one or more web service operations, the logic encoded in one or more media for execution and when executed operable to:

access a Web Services Description Language (WSDL) file describing a web
5 service;

according to the WSDL file, generate a symbol table for describing a client for generating one or more synthetic transactions with the web service;

receive input from a user indicating:

one or more operations of the web service for invocation;

10 an order for invoking the operations of the web service; and

one or more values of one or more parameters of the operations;

incorporate the input from the user into the symbol table; and

generate the client according to the symbol table;

15 18. The logic of Claim 17, encoded in an application server deployment file deployable to an application server.

19. The logic of Claim 18, wherein the application server is a JBOSS application server.

20

20. The logic of Claim 17, wherein output from a first one of the operations of the web service is usable as input to a second one of the operations of the web service.

21. The logic of Claim 17, provided by a synthetic transaction module in a web
25 services distributed management (WSDM) tool.

22. The logic of Claim 17, wherein the WSDL file comprises a WSDL file from a third party server providing web services.

23. The logic of Claim 17, wherein the symbol table comprises a serialized tree of JAVA objects representing a parsed version of the WSDL file and capable of representing a sequence of the operations of the web service for invocation.

5 24. A system for generating one or more clients for generating one or more synthetic transactions with one or more web service operations, the system comprising:

means for accessing a Web Services Description Language (WSDL) file describing a web service;

10 means for, according to the WSDL file, generating a symbol table for describing a client for generating one or more synthetic transactions with the web service;

means for receiving input from a user indicating:

one or more operations of the web service for invocation;

an order for invoking the operations of the web service; and

one or more values of one or more parameters of the operations;

15 means for incorporating the input from the user into the symbol table;

means for generating the client according to the symbol table; and

means for making the client available for generating one or more synthetic transactions with the web service.

1/2

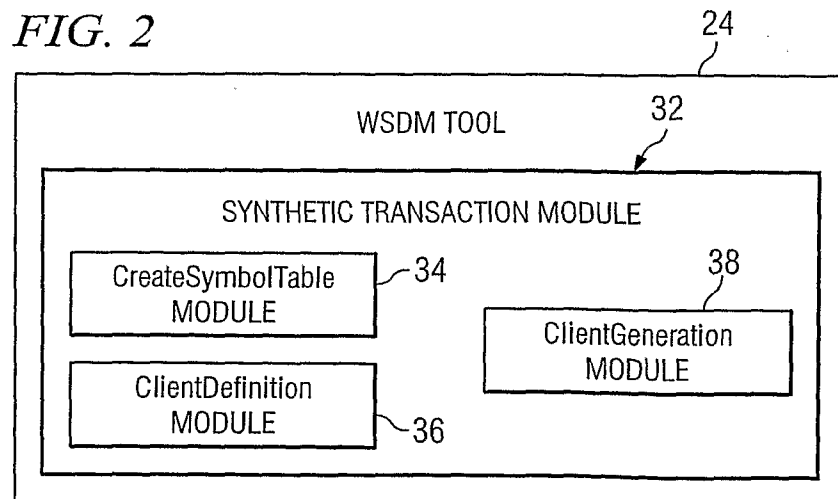
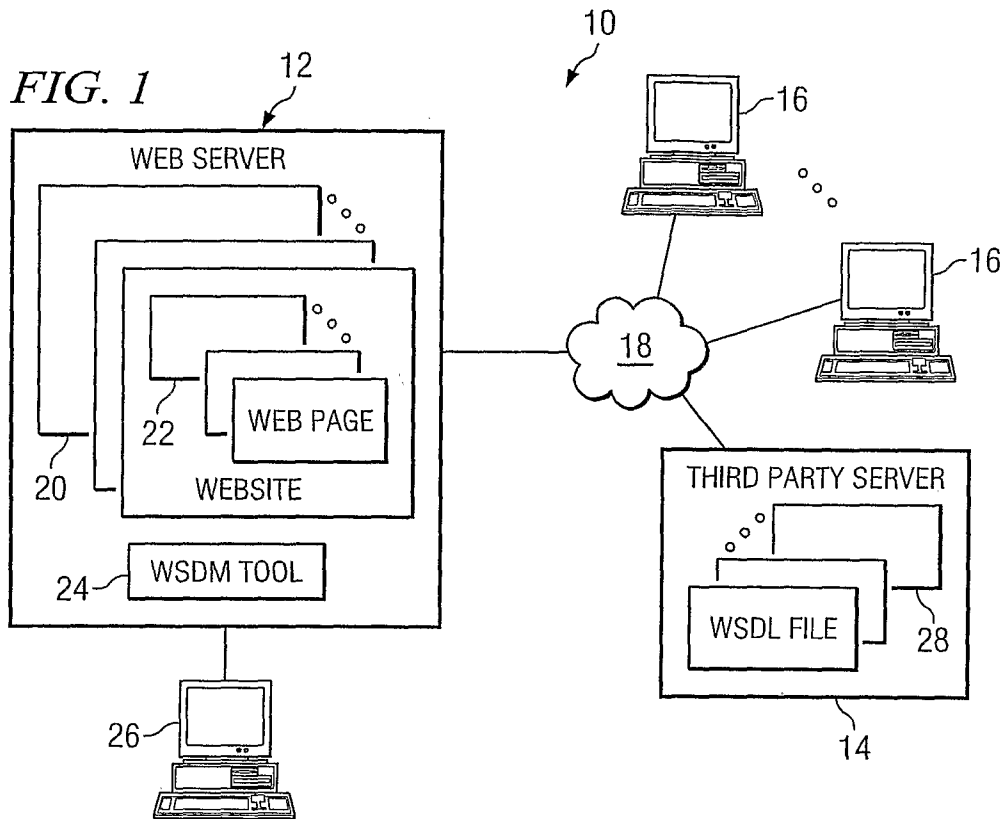
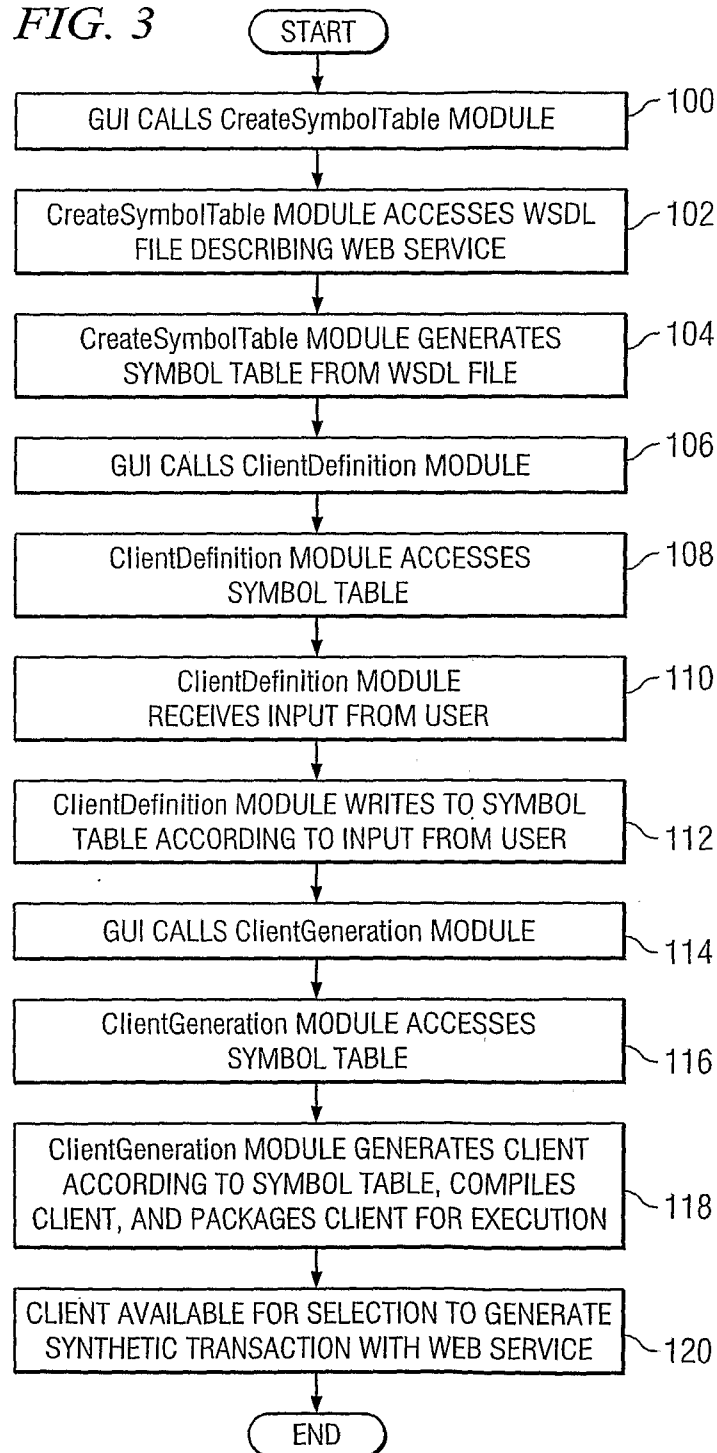


FIG. 3



INTERNATIONAL SEARCH REPORT

International application No

PCT/US2006/028063

A. CLASSIFICATION OF SUBJECT MATTER

INV. G06F9/44

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

EPO-Internal, INSPEC, COMPENDEX, IBM-TDB

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2005/038708 A1 (WU LISA [US]) 17 February 2005 (2005-02-17) paragraphs [0017] - [0019] paragraphs [0022], [0023] paragraphs [0034] - [0040] -----	1-12, 14-24
A	SAHU M: "XML DEVELOPMENT IN JAVA IT'S ALL IN THE BEANS" WEB TECHNIQUES, MILLER FREEMAN, US, vol. 4, no. 6, June 1999 (1999-06), pages 51-55, XP001130850 ISSN: 1086-556X page 51, left-hand column, lines 1-29 page 51, right-hand column, lines 8-16 page 54, middle column, line 41 - right-hand column, line 35 ----- -/--	1-12, 14-24

☒ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search

4 December 2006

Date of mailing of the international search report

13/12/2006

Name and mailing address of the ISA/

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,
Fax: (+31-70) 340-3016

Authorized officer

de Man, Ronald

INTERNATIONAL SEARCH REPORT

International application No

PCT/US2006/028063

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	EP 1 523 151 A2 (IBM [US]) 13 April 2005 (2005-04-13) paragraph [0010] -----	1-12, 14-24
A	GLASS G: "The Web services (r)evolution: Part 4. Web Services Description Language (WSDL)" INTERNET CITATION, [Online] 1 February 2001 (2001-02-01), pages 1-11, XP002247117 Retrieved from the Internet: URL:ftp://www6.software.ibm.com/software/d eveloper/library/ws-peer4.pdf> [retrieved on 2003-07-10] page 3, lines 6-29 -----	1-12, 14-24

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/US2006/028063

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2005038708	A1	17-02-2005	NONE
EP 1523151	A2	13-04-2005	NONE