Technologies for Web Services Orchestration

Elena T. Ivanova[1]

Abstract – **Coordination of multiple services is considered. The paper surveys some aspects of main technologies for web services orchestration and focuses on two of them, BPEL and XPDL.**

Keywords – **Web services, Orchestration, BPEL, XPDL**

I. INTRODUCTION

Web service technologies sprang up from the need for different software applications to interoperate with each other. In general, Web services are modular applications or functions, which are generally independent and self-describing, that can be discovered and called across the Internet or an enterprise intranet. Web services are based on open standards: WSDL (Web Services Definition Language) for description, SOAP (Simple Object Access Protocol) for communication and UDDI (Universal Description, Discovery, and Integration) for register and discover services. Service orchestration is the next stage in developing and extending web services paradigm and addresses the need of composing several services in some business logic in order to achieve more complex and meaningful processes.

Term Orchestration refers to coordinating of multiple tasks. Orchestration of services means coordinating of several services (i.e. their relations and consequence of execution) so that the result is a complex service or a process composed by these services.

There is another set of technologies, relevant to composing of services – choreography. Choreography refers to global, multiparty, peer-to-peer collaborations where the component entities interact in long-lived stateful and coordinated way regardless of any programming model or supporting platform used. Choreography languages (examples are WSCI, WS-CDL, etc) are mainly descriptive and cannot be directly executed. These languages are necessary to be mapped to an orchestration language in order to be executed. Orchestration focuses on the behaviour of the composed services as a whole. Orchestration languages (e.g. BPLM, BPEL, XPDL, BPELJ, jPDL, etc) are executable languages and define a runtime environment for their execution.

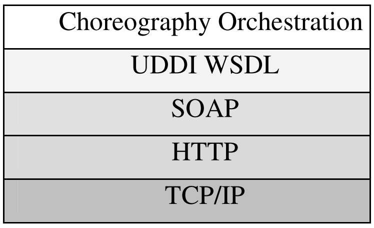

Fig. 1. Web services protocol stack

Choreography and orchestration languages are at the top of web services protocol stack (fig.1).

The paper surveys some aspects of main technologies for web services orchestration and focuses on two of them, BPEL and XPDL. Most of the software tools supporting orchestration are based on them.

II. LANGUAGES FOR ORCHESTRATION

There are several standardization bodies and organizations that aim at developing and adopting different orchestration technologies. Some of them are listed in the following Table I.

TABLE I

STANDARDIZATION BODIES FOR ORCHESTRATION TECHNOLOGIES

Standardization body	Short Description
BPMI (Business Process Management Initiative)	This is a non-profit organization that empowers companies of all sizes, across all industries, to develop and operate business processes that span multiple applications and business partners, behind the firewall and over the Internet. [1]
JSR (Java Specification Request)	This is an initiative of the Java Community Process (JCP) that standardizes how to automate business processes on a J2EE server. The JSR wants to standardize a set of XML meta tags that should be specified as meta data.[2]
OASIS (Organization for the Advancement of Structured Information Standards)	It is an international consortium working towards development, convergence, and adoption of common e-business standards.[3]
OMG (Object Management Group)	It is an open membership, not-for-profit consortium that produces and maintains computer industry specifications for interoperable enterprise applications.[4]
WfMC (The Workflow Management Coalition)	This is a consortium of about 300 organizations that defines a set of related standards based on an interesting reference model. The reference model describes the relation between a WFMS and other actors.[5]

There has been much effort in developing languages for web services orchestration during the last five-six years.

[1] Elena T. Ivanova is with Institute of Computer and Communication Systems – Bulgarian Academy of Sciences, Acad. G. Bonchev bl.2, 1113 Sofia, Bulgaria, e-mail: e_ivanova@hsh.iccs.bas.bg

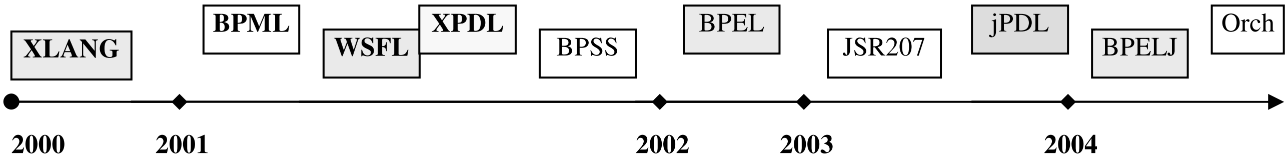

Fig. 2. Development of Orchestration Technologies

The chronological developments of these languages are tried to be represented on fig.2. Short descriptions of these languages are presented below in alphabetical order.

BEPLJ (BPEL for Java) [6] - It is a combination of BPEL and Java proposed by IBM and BEA that allows sections of Java code, called Java snippets, to be included in BPEL process definitions. Snippets are expressions or small blocks of Java code that can be used for things such as: loop conditions, branching conditions, variable initialization, Web service message preparation, logic of business functions etc. BPELJ introduces a few minor changes to BPEL as well as several extensions in order to fit BPEL and Java conveniently together. There is an effort to standardize BPELJ as part of JSR 207.

BPEL (Business Process Execution Language) [3] - This is a convergence of WSFL and XLANG, written by developers from BEA, IBM, SAP, Siebel and Microsoft. It is an XML-based language, depending on WSDL, XML-Schema and XPath, providing a language for the formal specification of business processes and interaction protocols. It supports both abstract and executable processes. In BPEL, a business process is composed of elements ("activities") that define activity behaviors, including the ability to invoke Web services and control flow, and to compensate when errors occur. The resulting business process is exposed as one or more Web services.

BPML (Business Process Modelling Language) [1] – This is a textual service-oriented process modelling language, based on XML, XML-NS, XML-S, XPath and WSDL. It provides an abstracted execution model for collaborative and transactional business processes. It is a strict superset of BPEL, although BPEL have been endorsed by for BPMI phase 2.0 instead of BPML. A process is viewed as a series of activities and an activity represents a component that performs a specific function. Activities can be composed into complex activities. Activities execute within a context which is transmitted from parent to child. In particular the context allows two activities to share properties. Properties constitute the data flow of BPML.

BPSS (Business Process Specification Schema) [7] - BPSS provides a standard framework (language) for business process specification. As such, it works with the ebXML Collaboration Protocol Profile (CPP) and Collaboration Protocol Agreement (CPA) specifications to bridge the gap between Business Process Modelling and the configuration of ebXML compliant e-commerce software.

jPDL (jBPM Process Definition Language) [8] - It is an XML based process execution language running in the Open Source JBoss jBPM (Java Business Process Management) workflow management system. Java actions can be triggered from the language.

JSR 207 (Java Specification Request) [2] is a draft JCP (Java Community Process) specification that is defining metadata, interfaces, and a runtime model that enables business processes to be easily and rapidly implemented using the Java language and deployed in J2EE containers. It will support tasks commonly encountered when programming business processes, e.g. parallel execution and asynchronous messaging.

Orch [9], [10] - This is a programming language and system for orchestrating distributed services. Orc has a strong theoretical foundation that supports modular composition and analysis of concurrent programs. The Orc model [9] assumes that basic services, like sequential computation and data manipulation, are implemented by primitive sites. Orc provides constructs to orchestrate the concurrent invocation of sites to achieve a goal – while managing time-outs, priorities, and failure of sites or communication.

XLANG [11] - It is an extension of WSDL, providing both the model of an orchestration of services and collaboration contracts between orchestrations. It has been standardized by Microsoft and has been used as a base by BPEL. XLANG, like BPML, were designed with an explicit -calculus theory foundation. WSFL and XLANG have converged to BPEL.

XPDL (XML Process Definition Language) [5] - It is an XML-based language from the Workflow Management Coalition (WfMC) for defining business processes. XPDL was based around a common set of functions for work distribution found in most workflow products. The XPDL's syntax is specified by an XML Schema.

WSFL (Web Services Flow Language) [6] - It is an XML based language, standardized by IBM, for the description of Web Services compositions as part of a business process definition. It relies and complements existing specifications like SOAP, WSDL, XMLP and UDDI. WSFL considers two types of Web Services compositions: the first type specifies an executable business process known as a flowModel; the second type specifies a business collaboration known as a globalModel.

Available orchestration technologies can be divided into three groups, according to their basic language: XML based, programming language based (mostly java based), and other based. This classification is represented on the following figure 3.

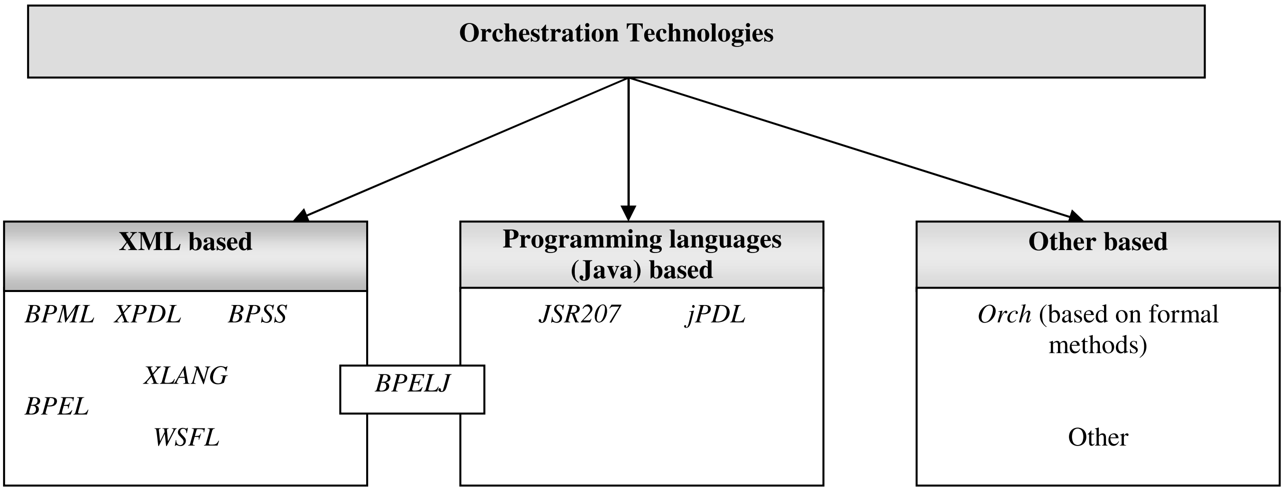
Orchestration Technologies
XML based
BPML
XPDL
BPSS
XLANG
BPEL
WSFL
BPELJ
Programming languages (Java) based
JSR207
jPDL
Other based
Orch (based on formal methods)
Other

Fig. 3. Types of Orchestration Technologies

III. Requirements for Web Services Orchestration

In order for web services to be composed into a complex service or business process, there are a set of requirements which should be addressed. Several works and papers deal with such requirements [12, 13]. The main of them can be summarized as follows:

- ability to invoke services in an asynchronous manner – the system has to enable services to be invoked concurrently, not only sequentially;
- ability to manage transaction and compensation – this is crucial for long-running compound services – in case of a failure in a single component service, the system has to provide adequate compensation;
- exception handling – how the system will behave when some error occurs;
- security and reliable messaging – security is a key point in all the layers (see fig.1) in web service technologies;
- support the separation of abstract process logic and concrete web services used – it is necessary for building dynamic and flexible processes.

Almost all available languages for web services orchestration have mechanisms respecting the aforementioned requirements.

IV. BPEL and XPDL – main features

This section is devoted to the most used orchestration languages – BPEL and XPDL.

A. BPEL

BPEL is:

- is an XML-based language designed to enable task-sharing for a distributed computing;
- is written by developers from BEA Systems, IBM, and Microsoft;
- combines and replaces IBM's WebServices Flow Lan–guage (WSFL) and Microsoft's XLANG specification;
- depends on WSDL, XML Schema, and XPath;
- supports both abstract processes and executable processes:
  - Abstract processes are useful for specifying expected protocols and publicly visible behaviors without too much detail;
  - Executable processes contain enough detail to fully specify execution.
- BPEL is essentially a layer on top of WSDL, with WSDL defining the specific operations allowed and BPEL defining how the operations can be sequenced.

In BPEL, a business process is composed of elements ("activities") that define activity behaviors, including the ability to invoke Web services and control flow, and to compensate when errors occur. The resulting business process is exposed as one or more Web services. The BPEL elements are listed below.

- Partners / Partner Links – the different parties that interact with the business process
- Variables – Data variables used by the process; Persistent for long running interactions; Defined in WSDL types and messages
- Correlation Sets – Set of properties shared by all messages in a correlated group
- FaultHandlers – Activities that must be performed in response to a fault;
- CompensationHandlers - allowing the process designer to implement compensation actions for certain irreversible errors in business (wrapper for a compensation activity);
- EventHandlers – invoked concurrently if the corres–ponding event occurs;
- Activities (Flow Logic) – the actions that are being carried out within a business process. There are two types of activities:
  - basic activities (<receive> <reply> <invoke> <assign> <throw> <terminate> <wait> <empty>);
  - structured activities (<sequence> <switch> <while> <pick> <flow> <scope> <compensate>).

It is very hard to itemize all the software tools for service orchestration based on BPEL. The list is quite long. Some of these products are mentioned in the following Table II.

TABLE II
SOFTWARE TOOLS BASED ON BPEL

ActiveBPEL Engine	Bexee
ActiveBPEL Designer	Biztalk Server
ADONIS	Cape Clear Orchestrator
Apache Agila	iGrafx BPEL
BEA WebLogic	MidOffice
Oracle BPEL Process Manager	PXE
Parasoft BPEL Maestro	SAP NetWeaver Exchange Infrastructure

B. XPDL

It is an XML-based language from the Workflow Management Coalition (WfMC) for defining business processes. Whereas BPML and other business process languages are geared to Web services, the foundation of XPDL was based around a common set of functions for work distribution found in most workflow products. XPDL uses an XML-based syntax, specified by an XML schema.

The main elements of the language are [5]:

- Package – the container holding the other elements;
- Application – is used to specify the applications/tools invoked by the workflow processes defined in a package;
- Workflow-Process – is used to define workflow processes or parts of workflow processes; it is composed of elements of type Activity and Transition;
- Activity – the basic building block of a workflow process definition. There are three types of activities: Route, Implementation, and BlockActivity. Activities of type Route are dummy activities just used for routing purposes. Activities of type BlockActivity are used to execute sets of smaller activities. Element ActivitySet refers to a self contained set of activities and transitions. A BlockActivity executes such an ActivitySet. Activities of type Imple–mentation are steps in the process which are implemented by manual procedures (No), implemented by one of more applications (Tool), or implemented by another workflow process (Subflow).
- Transition – elements of type Activity are connected through elements of type Transition.
- Participant – is used to specify the participants in the workflow, i.e., the entities that can execute work. There are 6 types of participants: ResourceSet, Resource, Role, OrganizationalUnit, Human; System.
- DataField
- DataType.

The last two elements are used to specify workflow relevant data. Data is used to make decisions or to refer to data outside of the workflow, and is passed between activities and subflows.

Several orchestration software tools are listed in the Table III. The number of available orchestration products, based on XPDL, is quite less than these, based on BPEL.

TABLE III
SOFTWARE TOOLS BASED ON XPDL

Aspose.Workflow	ObjectWeb BONITA
Agentflow	OFBiz Workflow Engine
Enhydra JaWE	Open Business Engine
Enhydra Shark	Vignette Process Workflow Modeler
Newgen	wfmOpen

V. CONCLUDING REMARKS

Service orchestration is a natural growth and extension of web services paradigm. It refers to composing several services in a complex process. The paper has reviewed available technologies for web services orchestration and extracted some key features of theirs. The attention was paid on two of these technologies, BPEL and XPDL, because of their wide acceptance.

VI. REFERENCES

[1] http://www.bpmi.org

[2] http://www.jcp.org/en/jsr/

[3] http://www.oasis-open.org/home/index.php

[4] http://www.omg.org

[5] http://wfmc.org/

[6] http://www.ibm.com

[7] http:// www.ebxml.org

[8] http:// jbpm.org

[9] http://www.cs.utexas.edu/users/wcook/projects/orc/

[10] Misra J.and W. Cook, 2004, *Computation Orchestration: A Basis for Wide-Area Computing,* Lecture Notes for NATO summer school, held at Marktoberdorf in August 2004

[11] http://www.microsoft.com

[12] Peltz C., Web ser vices orchestration, available at http://devresource.hp.com/drc/technical_ white_papers/WSOrch/WSOrchestration.pdf

[13] Yushi C., L. E. Wah and D. K. Limbu, Web Services Composition - An Overview of Standards, Singapore Institute of Manufacturing Technology

This work is partly supported by European Commission, project VISP №FP6-027178, and National Scientific Fund of Bulgaria, project № ВУ-МИ-108/2005.