

13. Webové služby. K čemu slouží? Popis a vyhledávání služeb. Co je a k čemu slouží orchestrace a choreografie služeb. Technologie pro implementaci služeb.(A4M33AOS)

Webové služby. K čemu slouží?

Definice WS

Pracovní skupina Web Services Architectures konsorcia W3C zavedla následující pracovní definici: Základní myšlenka spočívá ve spojení dvou dnes široce rozšířených technologií. Těmito technologiemi jsou XML (univerzální jazyk pro popis dat) a HTTP (transportní protokol podporovaný většinou dnešních webových prohlížečů a serverů). Webová služba je potom aplikace identifikovatelná pomocí URI (angl. Uniform Resource Identifier) a její rozhraní a způsob komunikace lze definovat a poskytovat pomocí XML. Webová služba podporuje přímou interakci s jinými softwarovými agenty, kteří ke komunikaci používají zprávy ve formátu XML zasílané prostřednictvím internetových protokolů.

Vlastnosti WS

- *Nezávislost, samostatnost*
Pro využívání webových služeb není na klientské straně potřeba žádný speciální software, jak už bylo zmíněno, stačí pouze podpora XML a HTTP. Na straně serveru je nutný webový server, v případě implementace např. pomocí Java EE je to aplikační server implementující webový kontejner pro HTTP servlety (servlet je speciální program zpracovávající klientské požadavky na straně serveru – oproti appletu, který je určený k běhu na klientské straně).
- *Samo-popisnost*
Ani spotřebitel služby ani její poskytovatel nemají vlastní mechanismy pro určení významu zpráv. Definice a popis zprávy je posílán společně se samotnou zprávou.
- *Vyhledávání služeb v internetu*
Pro vyhledávání a následnou komunikaci WS slouží tyto standardy:
 - SOAP (Simple Object Access Protocol) – Někdy označovaný jako Service-Oriented Architecture Protocol.
 - WSDL (Web Service Description Language) – Popisný jazyk
 - UDDI (Universal Description, Discovery and Integration)

Mechanismus registrů pro vyhledávání webových služeb.

Webové služby lze stejně jako služby SOA sestavovat do vyšších celků. Pomocí WSDL a UDDI lze tento proces do značné míry automatizovat. Jednou ze základních vlastností SOA je volné propojení mezi službami. Aby byla tato vlastnost splněna, musí být spotřebitelé a poskytovatelé služeb navzájem co možná nejméně závislí. Tuto vlastnost umožňuje právě popis služeb. Každý spotřebitel služeb vytváří zprávy, které bude posílat cílové službě (poskytovateli služeb), právě na základě popisu cílové služby. Daný spotřebitel tímto docílí, že zpráva bude poskytovatelem přijatá a správně interpretována. Popis služby (angl. service description) určuje, jakým způsobem bude konzument nějaké služby komunikovat s poskytovatelem této služby. Specifikuje formát dotazu a odpovědi. Takovýto popis může také obsahovat seznam podmínek, které musí být splněny před/po komunikaci nebo

určujících úroveň QoS. Základním prostředkem pro vytváření popisu webových služeb (angl. web service description) je jazyk XML.

Více informací naleznete na <http://www.w3.org/XML>

RPC Web Service

RPC webové služby představují volání vzdálené funkce. Metoda je popsána jako operace ve WSDL. Parametry metody i odpověď jsou posílány jako XML zabalené v SOAP.



Většinou je implementováno jako mapování služby přímo na jazykově specifické volání funkce. Není tedy *loosely coupled*.

SOA Web Service

Webové služby se také používají k implementaci SOA, kde je základní jednotkou komunikace zpráva, nikoli operace. Tato skutečnost je často označována jako „message-oriented“ služba.

Na rozdíl od RPC webových služeb jsou SOA webové služby *loose coupled*, jelikož jsou zaměřeny na kontrakt poskytnutý WSDL, ne na implementační detaily.

RESTful Web services

RESTful webové služby jsou speciální podmnožinou webových služeb, které mají sadu předem definovaných operací. Tyto předem definované operace jsou přímo operace převzaté z protokolu HTTP, jedná se tedy o operace GET, POST, HEAD, atd. Každá z operací má vymezené svoje postavení vůči zdroji, který je identifikován pomocí URL. Následující tabulka shrnuje použití jednotlivých HTTP funkcí z pohledu REST služeb.

Metody	Použití v REST	Odesílaná data
GET	Získání konkrétní reprezentace zdroje.	Veškerá data jsou umístěna v URI konkrétního zdroje.
POST	Vytvoření nového objektu v systému.	Data jsou odesílána nejčastěji ve formě XML v těle požadavku.
PUT	Úprava existujícího objektu.	Stejně jako v případě POST.
DELETE	Vymazání objektu ze systému.	Objekt pro vymazání je jednoznačně určen svým URI.
HEAD	Získání dat, které popisují formát zdroje.	Stejně jako v případě GET.

Využití metod ve správném významu je zcela klíčové pro čisté REST služby. Existuje mnoho reálných implementací webových služeb, kde je právě tento princip porušen

Způsoby vytváření WS

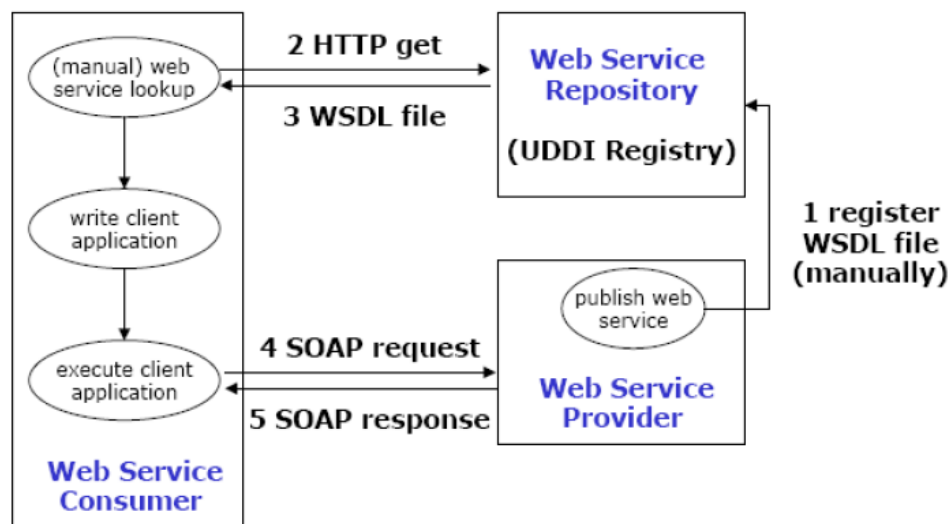
Bottom up

- Nejdříve je naimplementována služba v konkrétním jazyce, následně je vygenerováno WSDL
- Je považováno za jednodušší
- Riziko vzniku závislosti na programovacím jazyku či platformě

Top down

- Nejdříve je napsán WSDL dokument, následně je z něj vygenerován kód.
- Považováno za obtížnější.
- Výsledkem je čistší design.

Popis a vyhledávání služeb.



WSDL

Organizace W3C zavedla WSDL jako standard pro popis webových služeb.

V současné době je ve verzi 2.0. WSDL, potažmo popis služby, slouží k zodpovězení následujících otázek:

- Jaké funkce poskytuje daná služba?
- Kde je daná služba uložena?
- Jak může být s danou službou navázána komunikace?

WSDL je XML formát pro popis webových služeb. V kontextu WSDL je každá služba chápána jako množina koncových bodů (angl. service endpoints). V těchto bodech služba komunikuje se svým okolím pomocí zasílání zpráv (pro jednoduchost si lze koncový bod představit jako rozhraní služby). WSDL poskytuje formální definici koncových bodů. WSDL dokument se sestává ze dvou částí: abstraktního a konkrétního popisu koncového bodu.

Abstraktní popis

Abstraktní popis udržuje integritu popisu služby. Obsahuje informace charakterizující rozhraní služby bez ohledu na technologie, kterými je (bude) služba implementována, operační systém, ve kterém bude služba pracovat, a způsobu komunikace. Abstraktní popis obsahuje následující tři základní oddíly: interface, operation a message. Oddíl interface poskytuje abstraktní pohled na rozhraní služby – v rámci tohoto oddílu jsou vyjmenovány poskytované operace (pro jednoduchost si lze tyto operace představit jako veřejné metody objektů v klasickém objektově orientovaném přístupu). Každá operace má samozřejmě vstupní a výstupní parametry. Jelikož webové služby komunikují výlučně jen pomocí zpráv, jsou tyto parametry v podstatě zprávami. Oddíl operation tedy zahrnuje prvky input a output, které tyto zprávy specifikují.

Konkrétní popis

Tento popis slouží k navázání logiky popsané v abstraktním popisu na reálnou implementaci a k navázání komunikace na konkrétní protokol. Konkrétní popis se skládá ze tří základních oddílů: binding, endpoint a service. Oddíl binding popisuje požadavky služby pro navázání konkrétního spojení. Jinými slovy binding reprezentuje nějakou technologii, kterou služba využije pro komunikaci. Webové služby využívají SOAP, ale koncept webových služeb umožňuje použití i jiných technologií. Binding může být aplikován přímo na jednotlivé operation nebo na celé interface. Oddíl service seskupuje prvky endpoint (v tomto smyslu lze chápat jako port). Každý endpoint určuje fyzickou adresu, na které je služba přístupná.

```
<?xml version="1.0" encoding="UTF-8"?>
<description targetNamespace="http://example.com/bank"
xmlns="http://www.w3.org/2006/01/wsdl"
xmlns:ns1="http://example.com/bank">
<interface name="ns1:Bank">
...
<operation name="withdrawFunds">
<input messageLabel=""/>
<output messageLabel=""/>
</operation>
</interface>
<binding name="ns1:BankSOAPBinding">
...
```

```

<operation ref="withdrawFunds">
...
</operation>
</binding>
<service name="ns1:BankService" interface="tns:Bank">
<endpoint binding="ns1:BankSOAPBinding">
...
</endpoint>
</service>
</description>

```

Více informací na stránce <http://www.w3.org/TR/wsdl20>. (W3C: Web Services Description Language (WSDL) Version 2.0 Part 1: Core Language)

SOAP

Jelikož veškerá komunikace mezi službami je založená na posílání zpráv, musely být zavedeny takové standardy, aby služby mezi sebou komunikovaly jednotným způsobem. Takovýmto standardem se stal SOAP. SOAP je protokol umožňující spotřebiteli služeb komunikovat s jejich poskytovatelem. Tento protokol je nezávislý na typu sítě, podporuje zprávy ve formátu XML a v současné době je ve specifikaci 1.2 od organizace W3C.

Vlastnosti SOAP:

- SOAP je vyvíjen se zaměřením na jeho jednoduchost a snadnou rozšiřitelnost
- SOAP je nezávislý na transportních protokolech. HTTP je jen jedním z podporovaných protokolů. SOAP zprávy lze posílat třeba i emailem.
- SOAP je bezstavový protokol
- SOAP je nezávislý na operačním systému

Struktura zprávy v SOAP

Každá zpráva dodržující podmínky kladené SOAP je v podstatě *balíček* (*obálka*, angl. *envelope*). Tento balíček obsahuje hlavičku (angl. *head*) a *tělo* (angl. *body*). Hlavička se skládá z několika bloků, které obsahují metainformace. Hlavička je nepovinná (tzn. může být vynechána). Metainformace v sobě ukrývají část komunikační logiky a obecně umožňují zavádět nová rozšíření. Typicky hlavička obsahuje nutné informace pro všechny služby, které mohou zprávu obdržet. Cílová služba potom na základě těchto informací rozhodne o způsobu zpracování zprávy. Tělo obsahuje samotná data (ve formátu XML). Tělo může také obsahovat sekci pro *chyby* (angl. *faults*), která obsahuje logiku pro zpracování výjimek. Většinou jsou v této části uloženy jednoduché zprávy, které slouží k odeslání informací o chybě při výskytu výjimky. SOAP zahrnuje také prostředky pro posílání dat, která jsou těžko popsitelná pomocí XML (binární data, např. obrázky). Takováto data se posílají jako *přílohy* (angl. *attachments*).

Registr UDDI

UDDI (The Universal Description, Discovery and Integration Service) poskytuje mechanismus, přes který mohou klienti dynamicky hledat požadované webové služby. Tímto způsobem by aplikace měly být schopny se kontaktovat na služby poskytované externími partnery. Registr UDDI se podobá službě „CORBA trader“, nebo na něj lze pohlížet jako na „DNS service“ pro aplikace využívající webových služeb. Registr UDDI má dva druhy klientů: ty, kteří chtějí nějakou službu poskytovat a ty, kteří chtějí službu využívat.

Registr UDDI obsahuje tzv. „White pages“, které obsahují informaci jako jméno, adresu, číslo telefonu, nebo další kontaktní informace. Sděluje programátorovi, jak se má registrovat při používání služby. Další část UDDI představují tzv. „Yellow pages“, které obsahují specifikaci, jak může aplikace najít danou webovou službu. Třetí část se nazývá „Green pages“, kde jsou uvedeny informace nezbytné pro vyvolání služby. Jsou zde uvedena URL služeb, jména, argumenty apod.

Co je a k čemu slouží orchestrace a choreografie služeb.

Kompozice služeb

K vymezení pojmů orchestrace a choreografie v rámci kompozitních služeb si musíme nejprve ujasnit, co chápeme pod pojmem kompozitní služba. Na kompozitní službu se můžeme dívat jako na sadu služeb, které spolu spolupracují za účelem vykonání určitého procesu, který definuje interakční workflow. SOA poskytuje nezbytnou podporu pro konsolidaci různých služeb do jedné kompozitní služby odpovídající souhrnnému procesu. V literatuře se zmiňuje několik termínů pro popis různých stylů spolupráce mezi službami, nejvýznamnější jsou orchestrace a choreografie. Tyto dva termíny představují rozdílné vzory interakce mezi službami.

Orchestrace

Wikipedia definuje pojem orchestrace následovně. Orchestrace popisuje automatickou koordinaci a řízení komplexních počítačových systémů, middleware a služeb. Orchestrace zabezpečuje koordinaci procesů a výměnu informací pomocí webových služeb. Tato definice se zaměřuje hlavně na technologický pohled, ale je málo zaměřená na vymezení pojmu orchestrace v rámci webových služeb. Šafář již vymezuje tento pojem podrobněji. Orchestrace je centrální proces (může se jednat o další webovou službu) přebírající kontrolu nad službami, které jsou do procesu zapojeny, a koordinuje spouštění jednotlivých operací. Zúčastněné služby nevědí, a ani nemusí vědět, že jsou účastníky nějakého vyššího procesu. Dle Peltze orchestrace zahrnuje pořadí vykonávání interakcí webových služeb, popisuje tok vykonatelného procesu a může zahrnovat jak interní, tak externí webové služby. Při orchestraci je proces vždy řízen jednou stranou. Interakce při orchestraci nastávají na úrovni zpráv. Zahrnují byznys logiku, pořadí vykonávání úkolů a mohou pokrýt aplikace a organizování k definování dlouhotrvajícího, transakčního a vícestupňového procesního modelu. Dle orchestrace popisuje interakce, které má poskytovatel jedné služby s ostatními poskytovateli při poskytování své vlastní služby. Z tohoto důvodu se interakce v orchestraci

zaměřuje na jednoho poskytovatele. Interakce mohou být přímo vykonávány tímto poskytovatelem. Typickým příkladem *orchestration description language* je BPEL4WS a BPML. Orchestrace může být vykonávána tzv. orchestračním enginem, stejně jako byznys procesy mohou být vykonávány ve workflow enginech. Oracle ve svých materiálech popisuje orchestraci následovně. Samotná koordinace volání webových služeb (tzv. WS Orchestration) se skládá z několika základních operací, jako jsou například – synchronní nebo asynchronní volání webových služeb, XML data transformace, řízení běhu procesu (podmínky, cykly, výjimky) nebo notifikační události.

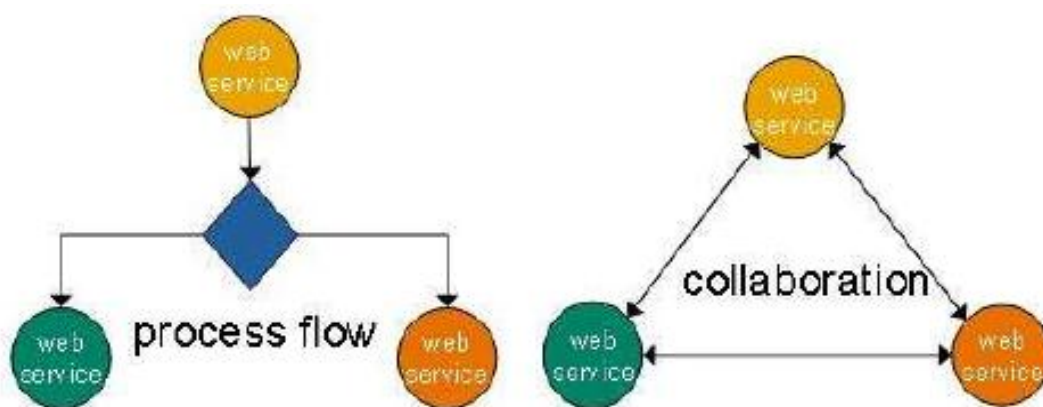
Choreografie

Choreografie popisuje interakce, které mají mezi sebou navzájem dvě a více aplikací při dosahování společného cíle a vztahy mezi těmito interakcemi. Proto musí být logika, která vykonává choreografii distribuována poskytovatelem aplikačních služeb. Typickým příkladem choreography description language je web-services choreography model 1.

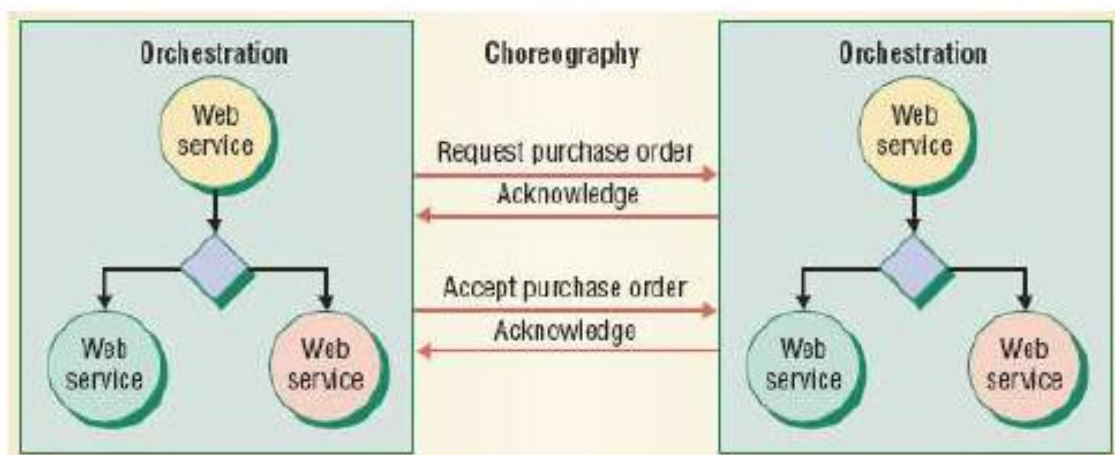
Popisy choreografie mohou sloužit k rozdílným cílům. Mohou být použity jako standard byznys procesu, ve kterém mohou poskytovatelé aplikační služby označit části, které mohou vyplnit. Potom mohou poskytovatelé tyto popisy použít jako základ ke startu implementace jejich služeb. Alternativně, popisy choreografie mohou být vykonávány choreografistickými enginy. Peltz uvádí, že choreografie sleduje sekvenci zpráv zahrnující několik účastníků a zdrojů, dále je spojena s veřejnou výměnou zpráv a ne se specifickými vykonavatelnými procesy. Choreografie je ve skutečnosti více spolupracující, každý účastník zahrnutý do procesu popisuje svojí úlohu, kterou hraje v interakci (nejedná se o řízení z pohledu jedné strany). Šafář zmiňuje, že choreografie nevyužívá centrálního koordinátora. Každá zúčastněná služba přesně ví, kdy se má spustit a s kým má komunikovat. Všichni účastníci choreografie musí vědět o procesu, jeho operacích, zprávách a na časování výměn zpráv.

Společné a odlišné rysy

Obrázek č. 1 a obrázek č. 2 ilustrují rozdíl mezi orchestrací a choreografií. Obrázky vyjadřují skutečnost, že se orchestrace zaměřuje na vykonatelný proces, který může spolupracovat s externími a interními službami. Orchestrace reprezentuje řízení z pohledu jedné strany (služby), zatímco choreografie je více spolupracující, umožňuje každé straně (služby) popsat její úlohu v interakci. Choreografie se zaměřuje na vztahy (spolupráci) mezi službami.

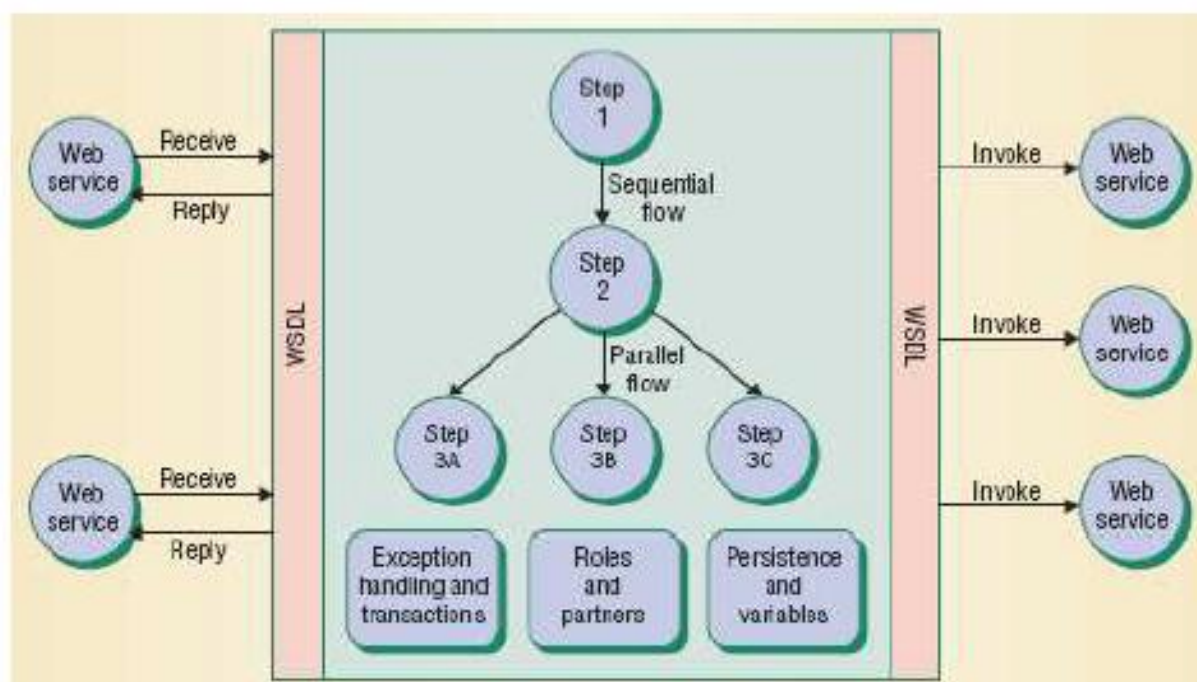


Obrázek č.1 Orchestrace vs Choreografie (Peltz 2003a)



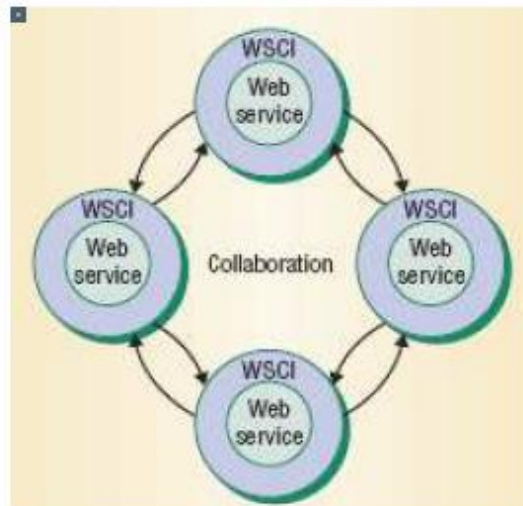
Obrázek č. 2 Orchestrace vs Choreografie (Pelz 2003b)

Obrázek č. 3 ilustruje orchestraci, jež je vyjádřena tokem procesu v BPELu. Specifikace umožňuje podporu strukturovaných aktivit určených k podpoře celého toku procesu, stejně tak jako základní aktivity, které zahrnují vazby s externími službami na daný proces.



Obrázek č.3 (Peltz 2003b)

Obrázek č. 4 vyjadřuje schéma choreografie. Specifikace se zaměřuje jenom na pozorovatelné chování mezi webovými službami, nikoli však definicí vykonatelného byznys procesu.



Obrázek č.4 (Peltz 2003b)

Business Process Execution Language

Business Process Execution Language (BPEL) je jazyk pro popis chování byznys procesů. Základ toho jazyka tvoří WSFL a XLANG. WSFL je zkratka pro Web Services Flow Language, jazyk vytvořený IBM pro popis kompozice webových služeb. XLANG je rozšíření WSDL. Zjednodušeně se dá říct, že popis služby v XLANG se skládá z popisu dané služby ve WSDL a popisu chování služby jako části byznys procesu. Varianta BPEL pro webové služby se označuje BPEL4WS (někdy také WS-BPEL). Tento jazyk rozšiřuje základní interakční schéma mezi službami o podporu byznys transakcí.

Web Services Choreography Description Language

Web Services Choreography Description Language (WS-CDL) patří do rodiny standardů, která se označuje WS-*. Tyto standardy se zabývají vzájemnou spoluprací služeb. Kromě WS-CDL sem patří ještě např. WS-Policy, WS-Security, WS-Coordination. WS-CDL je postavena na XML (a souvisejících technologiích, např. XPath) a podporuje WSDL.

Choreografie popisuje interakce mezi službami, závislosti mezi těmito interakcemi, včetně řízení toku závislostí (tzn. nějaká interakce musí proběhnout dříve než jiná), časových omezení atd. WS-CDL slouží k vytváření předpisů definujících podmínky a omezení vztahujících se na komunikaci pomocí zpráv v rámci spolupráce nějaké skupiny služeb se záměrem dosáhnout určitý cíl. Tyto předpisy popisují chování celé skupiny služeb z globálního pohledu. Každý účastník spolupráce potom na základě takového předpisu vytváří dílčí řešení cílového problému. Celkové řešení je vytvořeno kombinací těchto dílčích řešení. Výhodou tohoto globálního přístupu je, že zatímco způsob řešení v dílčích (lokálních) systémech se může měnit, globální předpisy zůstávají stejné a součinnost všech zúčastněných je zachována.

Technologie pro implementaci služeb.

Nevím, co se tím přesně myslí, podle mne všechny technologie jsou již dostatečně popsány v předchozích kapitolách.

Bibliografie

- Kadlec, J. (2010). *REST a webové služby v jazyce Java*. Načteno z http://is.muni.cz/th/143417/fi_m/143417_kadlec_jiri_diplomova_prace.pdf
- Knap, P. (2007). *Orchestrace a choreografie služeb*. Načteno z <http://www.cssi.cz/cssi/system/files/all/00knap.pdf>
- Richta, K. (nedatováno). *Standardy pro Webové služby WSDL, UDDI1*. Načteno z <http://www.ksi.mff.cuni.cz/~richta/publications/Richta-MD-2003>
- Vokřínek, J. (nedatováno). Slidy k přednáškám 2 a 3.
- Weiss, P., & Rychlý, M. (2007). *ARCHITEKTURA ORIENTO VANÁ NA SLUŽBY, NÁVRH ORIENTO VANÝ NA SLUŽBY, WEBOVÉ SLUŽBY*. Načteno z http://www.fit.vutbr.cz/research/view_project.php?file=%2Fproj%2F378%2FSOA_SOAD_WS.pdf&id=378
- Wikipedia. (nedatováno). *Wikipedia - Web Services*. Získáno 2011, z http://en.wikipedia.org/wiki/Web_service