

Podniková sběrnice služeb

Jindřich Štumpf

Podniková sběrnice služeb ESB (Enterprise Service Bus) je relativně novou kategorií na trhu technologií integračního middlewaru. Poprvé se objevila v roce 2002 a dnes představuje velmi zdařilou implementaci servisně orientované integrace – jednoho ze směrů servisně orientované architektury SOA (Service Oriented Architecture).

SOA jako taková není, nemůže a nikdy ani nebude standardem. Její podoba závisí na konkrétní implementaci konkrétního produktu daného dodavatele softwaru. Každý z těchto dodavatelů pak nabízí v rámci příslušných standardů určitá vylepšení či rozšíření, čím se tyto „standards“ stávají vzájemně nekompatibilními a vytváří se tak prostor pro výzvy v integračních projektech. Příklon většiny dodavatelů softwaru ke konceptu ESB (resp. k produktům takto nazvaným) pak naznačuje, kudy se SOA bude dále ubírat.

Hlavním cílem ESB je spolehlivě realizovat a koordinovat interakci (mezi)podnikových aplikací a procesů. Mezi její hlavní přednosti patří schopnost technologicky podporovat podnikání v reálném čase (Real Time Enterprise), velká flexibilita při implementaci změn a schopnost inkrementálního a distribuovaného nasazení. Zároveň jde o nízkonákladovou (low-end) alternativu komplexních sad integračních brokerů, která sice nabízí o něco méně funkcionality, zato však jednodušeji a za méně peněz.

Java Business Integration

Koncepty podnikových sběrnic služeb (ESB) se vzájemně velmi liší. Některé mají blíže spíše k aplikačnímu serveru (Oracle, BEA), některé představují inteligentní vrstvy nad MOM produkty (Sonic, Fiorana), některé jsou čistě orientované na webové služby (Systinet, Cape Clear).

Další odlišností může být instanční prostředí pro provoz služeb, které u některých poskytovatelů představuje odlehčený kontejner (light-weight container). Není to nutnou podmínkou, ale kontejnery těchto ESB poskytovatelů jsou vesměs tvořeny v prostředí Java, které tak vloženým službám poskytuje hostitelské virtuální stroje JVM (Java Virtual Machine). Prostředí kontejneru nabízí vložené službě/službám/procesům veškeré nízkourovňové služby, které by jinak vývojáři museli pracně, zbytečně a opakovaně implementovat (např. ošetření transakcí, správa stavu, správa vláken, sdílení zdrojů).

V roce 2005 dokončila skupina dodavatelů (mezi jinými BEA, Borland, Cap Gemini, JBoss, Nokia, Oracle, SAP, Sonic Software, Sybase, Tibco, Vignette a další) vedená Sun Microsystems specifikaci JBI (Java Business Integration), která ESB formálně zastřešuje jako rozšíření J2EE.

Java Business Integration 1.0 je dobrým standardizačním prvkem na jinak nejednotném poli na Javě založených ESB. Představuje sadu specifikací pro tyto

odlehčené kontejnery služeb, poskytuje jmenné prostory pro adresování služeb odstíněné od fyzických adres, poskytuje správní rámec (typicky JMX) a používá normalizovaný XML formát pro interní asynchronní výměnu zpráv mezi službami.

Je však dobré si uvědomit, že JBI je ohraničeno jedním JVM a představuje tak spíše hub-and-spoke architekturu (hvězdicová topologie). Není tedy navrženo jako náhrada nebo konkurent těm ESB, které dokáží své služby a procesy geograficky rozprostřít přes více JVM a více fyzických strojů. JBI navíc umožňuje jednotlivým kontejnerům komunikaci přímo mezi sebou (mezi vstupními/výstupními body vložených služeb), což může znamenat hrozbu ztráty kontroly a přehledu nad celkovým tokem XML zpráv.

JBI je založena na řadě standardů pro Javu a webové služby včetně WSDL, XML, JAX-RPC (resp. WS-I Basic Profile 1.1) a může být implementována jak v J2SE tak J2EE. Přijetí JBI by mělo podpořit ekosystém vyměnitelných, interoperabilních a dodavatelsky neutrálních integračních komponent. To by mohlo být impulsem pro model, v kterém by se komponenty třetích stran mohly vcelku jednoduše zapojovat a orchestrovat v rámci ESB, a také impulsem ke vzniku trhu s těmito komponentami.

JBI komponenty jsou rozděleny do dvou typů, přičemž oba typy mohou hrát role jak poskytovatele, tak konzumenta služeb:

- **Service Engines (SE)**, které vykonávají provozní logiku a transformační služby pro ostatní komponenty a zároveň mohou tyto služby samy konzumovat. SE mohou integrovat na Javě založené aplikace a ostatní zdroje s programovacími rozhraními Java API.
- **Binding Components (BC)** poskytující JBI kontejneru konektivitu pro externí služby – komunikační protokoly nebo služby poskytované jinou aplikací. BC mohou integrovat aplikace a ostatní zdroje, které Java API nemají.

Komunikační protokoly

Z pohledu podporovaných komunikačních protokolů, pomocí nichž je možné se k ESB připojit, lze ESB rozdělit na:

- **Jednoprotokolové** – zaměřené pouze na podporu webových služeb (SOAP/HTTP/S). Tyto ESB se také označují jako Web Services Brokers (WSB). Jedním z nejznámějších dodavatelů WSB je firma Systinet (nyní součást společnosti Hewlett-Packard).
- **Víceprotokolové** – čím více komunikačních kanálů-protokolů ESB nabízí (SOAP/HTTP, XML/HTTP/S, XML/HTTP-D, JMS a další protokoly), tím dává integrátorům větší pružnost při zvažování konkrétní volby. Analytikové dávají mnohem větší šance na rozšíření právě víceprotokolovým ESB.

Ani víceprotokolová ESB však nepokryje všechny protokoly, které jsou při integraci podnikových aplikací a systémů potřebné. Dodavatelé ESB proto nabízejí různé specializované adaptéry. Na adaptéry je možné nahlížet jako na middleware pro middleware. Existují stovky různých adaptérů pro velké aplikační balíky (SAP, SIEBEL atd.), různé B2B adaptéry (EDI, SWIFT atp.), adaptéry pro transakční systémy (CICS, Tuxedo aj.), databázové adaptéry apod.

Komunikace bývá zpravidla řešena tak, že adaptér směrem k ESB nabízí některý ze jejích standardních protokolů (viz výše) a směrem k integrované aplikaci/systému jemu nativní komunikační protokol. Adaptéry představují velmi kvalitní, spolehlivý,

ale nutno dodat také velmi drahý middleware často řádově přesahující cenu samotné ESB.

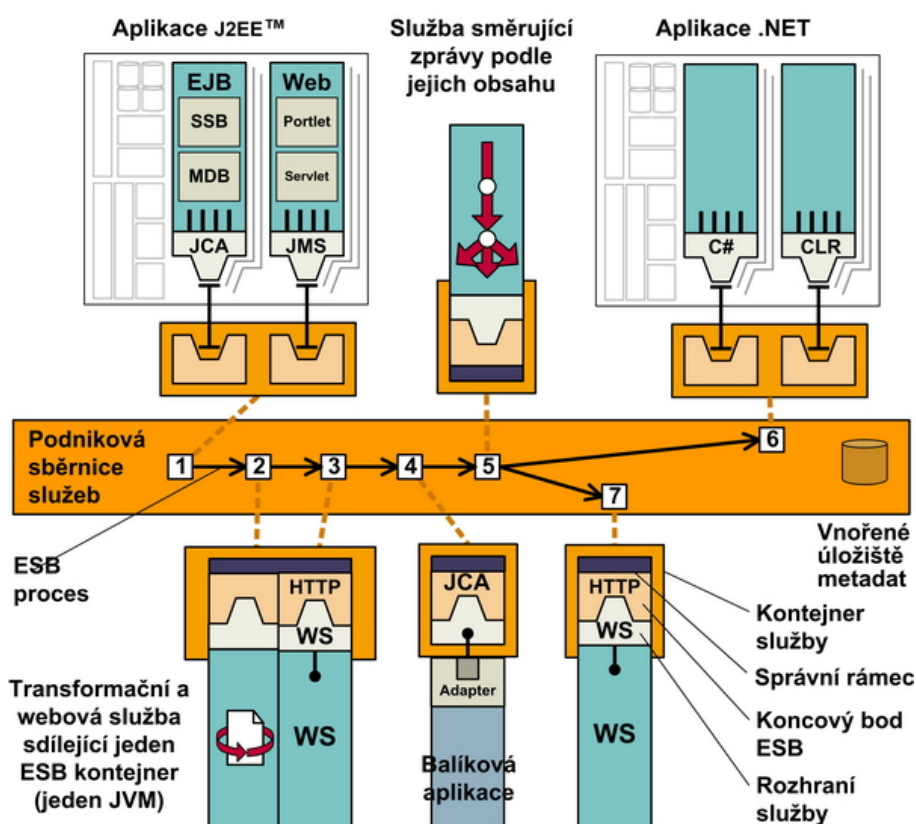
Dodavatelé ESB si zpravidla nevyrábějí tyto specializované adaptéry sami, ale využívají partnerství se třetí stranou. Asi nejužívanějším v této oblasti je partnerství se společností iWay Software, která má ve svém portfoliu více než 200 takových adaptérů.

Vrstvy ESB

ESB kombinuje několik relativně samostatných technologií. Tou aplikačně nejspodnější vrstvou je messaging. Ten tvoří komunikační páteř, která fyzicky manipuluje se zprávami či dokumenty a je řízena pravidly definovanými ve vrstvách vyšších (směrování apod.).

Základní vrstvou ESB je framework umožňující vytvářet, distribuovat, spouštět, komponovat a spravovat služby. Předmětem zájmu je zde službami řízená výměna bezstavových dokumentů.

Nadstavbovými vrstvami jsou BPM (Business Process Management), která umožňuje výměnou stavových dokumentů vytvářet, spouštět, monitorovat a sledovat business procesy, a BAM (Business Activity Monitoring) doplněný o auditing a žurnál. Z pohledu ESB je tato vrstva službou umožňující generovat, zachytávat, zpracovávat, uchovávat a prezentovat události či metriky nižších vrstev.

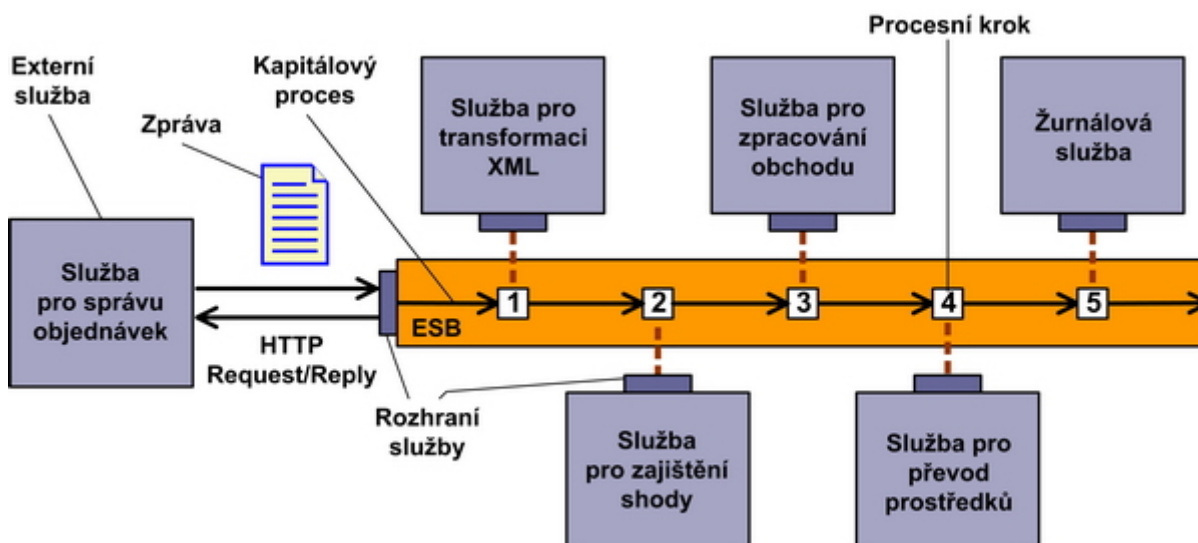


Obr. 1: Princip fungování podnikové sběrnice služeb

Jak funguje ESB

Podniková sběrnice služeb ESB spojuje a zprostředkovává všechny komunikace a interakce mezi službami. Zároveň dovoluje služby a procesy rychle měnit, snadno je připojovat, zviditelnit a řídit. Protože ESB má značný vliv na náklady spojené s vývojem a provozem daného podnikového systému, je účelné dobře porozumět její struktuře a funkcím.

Schéma na obr. 2 ukazuje příklad architektury SOA použité pro integraci a mediaci vícekanalového finančního kapitálového procesu. Jde o kombinaci již existujících (legacy) systémů opatřených rozhraním, díky němuž fungují jako služby, a nově vytvořených služeb.



Obr. 2: Příklad SOA aplikace pro finanční obchodování využívající ESB.

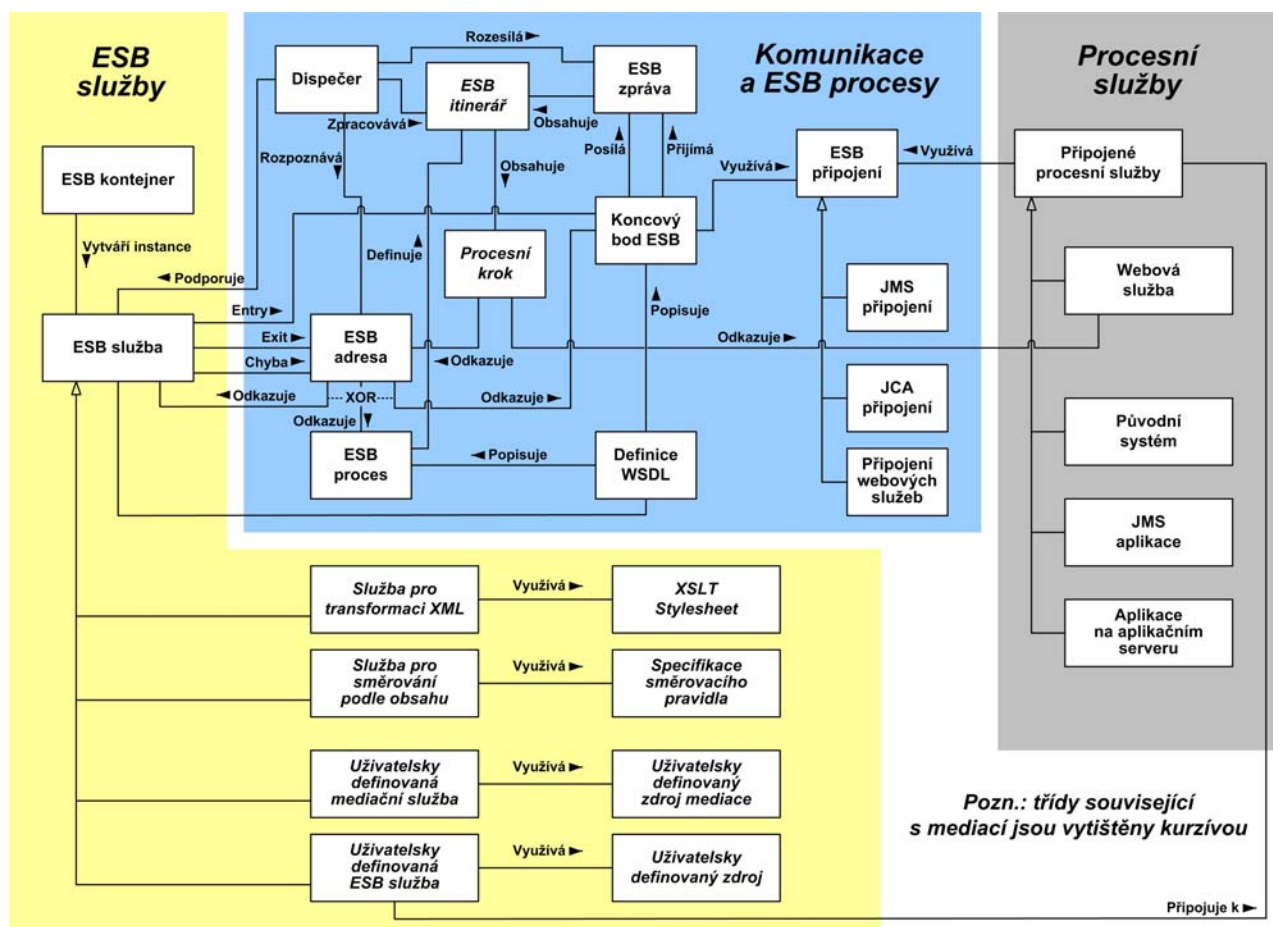
Ze schématu je patrné, že objednávka z burzy je nejprve zpracována službou pro správu objednávek. Poté prochází transformací dat, aby vyhověla požadavkům kapitálového systému, a ověřením, zda vyhovuje příslušným předpisům a vyhláškám (služba pro zajištění shody). Dále je zpracována službou pro zpracování obchodu a poté dojde k samotnému převodu finančních prostředků. Nakonec je objednávka zapsána jako dokončená transakce žurnálovou službou. Přitom nejde o jednu integrovanou aplikaci – celý proces je vykonáván jako sekvence heterogenních procesních kroků koordinovaných infrastrukturou ESB.

V tomto případě poskytuje sběrnice ESB konektivitu a směrování toku dat mezi službami, transformuje XML data, čímž umožňuje komunikaci služeb s rozdílnými rozhraními, a zajišťuje posloupnost vykonávání jednotlivých služeb tak, aby odpovídala danému podnikovému procesu.

Služby, komunikace a mediace

Jedním z průkopníků ESB je společnost Progress Software, která kromě produktové realizace (Sonic ESB) také formalizovala popis jejích funkcí pomocí schémat založených na standardizovaném modelovacím jazyku UML (Unified Modeling Language). UML Class schémata slouží k architektonické definici ESB, která je

užitečná pro porozumění principům ESB a diskuzi s podnikovými SOA architekty. Na obr. 3 je UML Class schéma s přehledem ESB struktur podporujících služby, komunikaci a mediaci (tj. zpracování zpráv na sběrnici).



Obr. 3: Přehled architektonických prvků pro služby, komunikaci a mediaci

ESB služby

ESB služby (levá část schématu na obr. 3) mohou být různých typů. Typické ESB služby pro mediaci jsou: transformace XML, směrování podle obsahu (Content-Based Routing) a uživatelsky definovaná mediace (User Defined Mediation). Služba transformace XML provádí datovou konverzi z jednoho XML dokumentu do druhého podle specifikace XSLT. Služba směrování podle obsahu řídí přenos zpráv ke koncovým bodům jiných služeb, přičemž se řídí nastavenými směrovacími pravidly. Služba uživatelsky definované mediace vykonává funkce jako žurnál nebo uživatelsky definované validace, které doplňují základní procesní logiku. Procesní (business) služby pro provádění základní procesní logiky spojuje uživatelsky definovaná ESB služba.

ESB služby vykonávají svou činnost v rámci ESB kontejneru. Tento kontejner slouží k vytvoření instancí ESB služeb a poskytuje rozhraní pro jejich správu. Umožňuje také správu komunikačních protokolů a kvůli lepší škálovatelnosti dovoluje vykonávat služby více vláknů. Standardně také umožňuje vykonávat distribuované procesy.

Připojené procesní služby

Připojené služby (pravá část schématu na obr. 3) vykonávají procesní logiku aplikací integrovaných pomocí ESB. Mezi typy procesních služeb patří webové služby, převzaté (legacy) systémy, aplikace založené na JMS (Java Message Service) a všechny druhy aplikací provozovaných na aplikačním serveru. Všechny typy procesních služeb mohou prostřednictvím ESB navzájem komunikovat a s ESB službami vykonávat mediaci.

Například služba pro správu objednávek na obr. 3 je připojená služba, která se spojuje pomocí HTTP připojení. Naproti tomu služba pro zajištění shody, služba pro zpracování obchodu a služba pro převod prostředků jsou procesní služby, které byly spojeny s ESB přímo jako uživatelsky definované ESB služby.

Komunikace a ESB procesy

Prostřední část schématu na obr. 3 znázorňuje klíčovou schopnost procesních i ESB služeb komunikovat a vykonávat ESB proces – sekvenci kroků aktivujících služby nebo jiné ESB procesy. Kapitálový proces na obr. 2 je pětikrokovým ESB procesem.

ESB služba zasílá a přijímá zprávy prostřednictvím odkazů na koncové vstupní a výstupní body ESB služeb či ESB procesů. Například ESB služba (jako je služba pro zajištění shody na obr. 2) může nastavit svůj výstup tak, aby odkazoval na vstupní koncový bod další služby (v našem případě Služby pro zpracování obchodu), která tyto zprávy přijme. Alternativně mohou koncové body ESB odkazovat na jiný ESB proces.

ESB proces je definován ESB itinerářem obsahujícím sekvenci procesních kroků. Každý procesní krok může vyvolat jednu (nebo několik) ESB služeb, jiných ESB procesů nebo externí webovou službu. ESB itinerář je doručován spolu s příslušející ESB zprávou, čímž vzniká kompletně distribuované a zároveň velmi spolehlivé prostředí pro zpracování ESB procesu. Neexistuje v něm žádný centrální engine pro provádění itineráře – každý ESB kontejner může samostatně zpracovat příslušný krok v ESB itineráři a směřovat následující krok. Tím je zajištěno jak efektivní vykonávání procesních kroků bez nutnosti komunikovat s centrálou, tak spolehlivost, protože zpracování může pokračovat, i když dojde k určitým výpadkům konektivity.

U každé ESB služby může dojít ke změně jejího vstupního a výstupního bodu bez toho, aby došlo ke změně jejich implementace.

Koncový bod ESB umožňuje ESB službám zasílat a dostávat zprávy se specifikovanou QoS (Quality-of-Service) – například „best effort“ nebo „guaranteed, once-and-only-once“. Připojení rozpozná, který podpůrný kanál zpráv je použitý koncovým bodem ESB pro propojení s procesní nebo ESB službou. Tento kanál zpráv může používat nativní ESB propojení nebo může pro externí systémy využít webovou službu, propojení JMS nebo adaptér JCA (J2EE Connector Architecture).

ESB propojení využívá uživatelsky konfigurovatelnou asynchronní a synchronní sémantiku včetně „publish-and-subscribe“, „send-and forget“ a „request-reply“. ESB propojení mezi službou pro správu objednávek a kapitálovým procesem v příkladu na obr. 2 je request-reply webové služby. Pokud procesní nebo ESB služba potřebuje změnit způsob, kterým se připojuje, může toho dosáhnout, aniž by bylo nutné měnit její implementaci. V případě, že služby vyžadují odlišné typy propojení, poskytuje ESB mediaci mezi odlišnými protokoly.

Třída ESB zpráv spolu se zasílací a přijímací vazbou mezi koncovými body vytvářejí základní strukturu k tomu, aby jedna služba mohla poslat zprávu druhé ESB službě prostřednictvím dispečera ESB služeb. Dispečer spravuje zasílání a přijímání ESB zpráv mezi ESB službami a externími koncovými body – ve schématu na obr. 3 realizuje šipky mezi jednotlivými procesními kroky. Tento dispečer také spravuje ESB procesy.

Pro ESB služby a ESB procesy mohou být vytvářeny standardizované definice WSDL (Web Services Definition Language). To dovoluje použití standardních nástrojů a integraci s dalšími komponentami infrastruktury SOA, jako je registr UDDI. Například v obr. 2 je celý kapitálový proces vystaven jako webová služba.

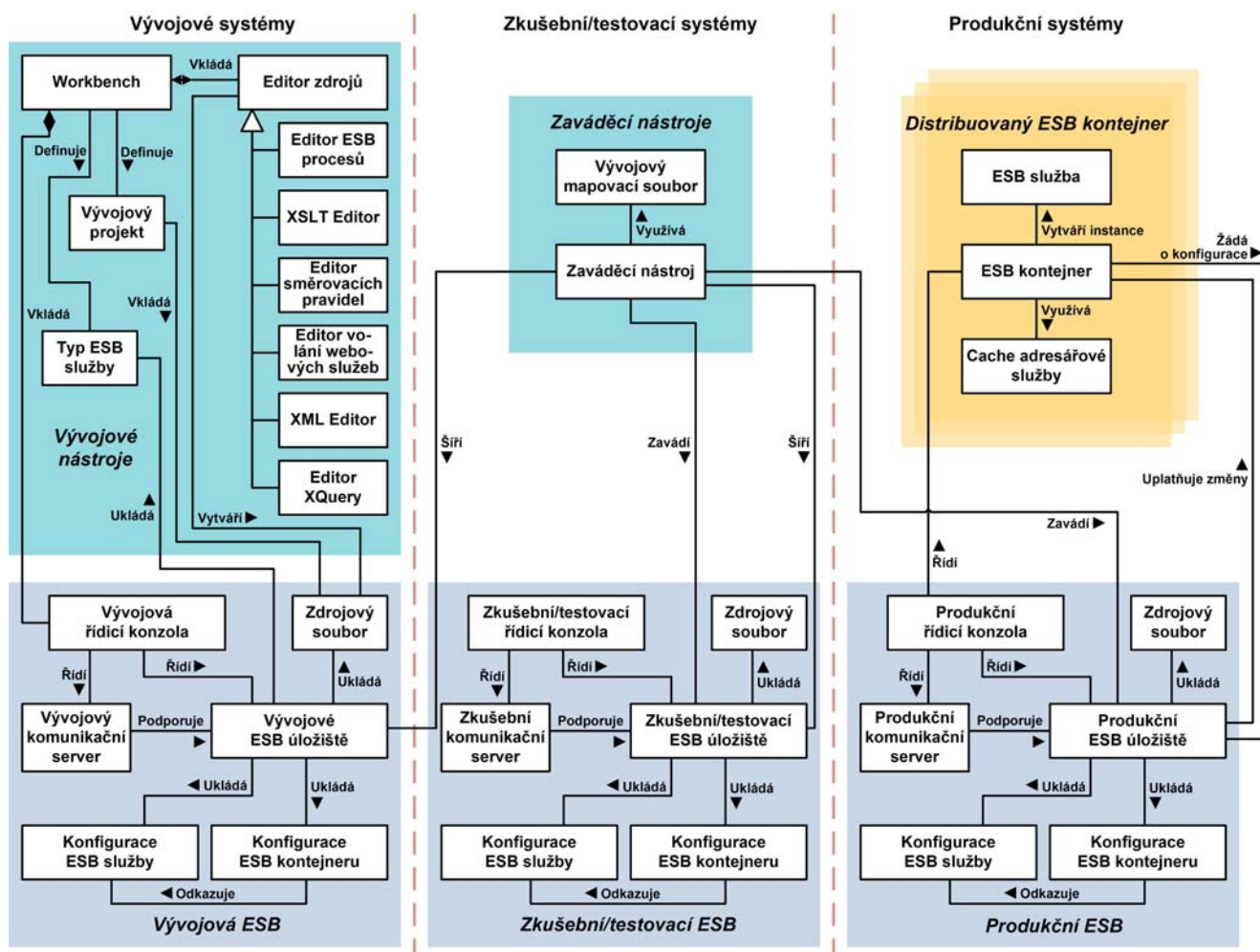
Takový způsob implementace ESB služeb, mediace a propojení s procesními službami má následující přednosti:

- akceschopnost podniku díky bohaté podpoře mediačních služeb včetně XML transformace a zabudované podpory distribuovaných ESB procesů,
- možnost širokého využití mnoha druhů procesních služeb a komunikačních protokolů; integrace heterogenních původních elementů i nových prvků webových služeb,
- distribuované vykonávání procesů, výkonnost a spolehlivost i ve vysoce distribuovaných nasazeních.

Životní cyklus ESB

Konkrétní implementace sběrnic ESB obvykle disponují bohatou podporou celého životního cyklu služeb umožňujících vývoj, zavádění a řízení. Obr. 4 ukazuje UML Class diagram s příkladem některých klíčových struktur této podpory.

Diagram je rozdělen do tří částí, které znázorňují migraci ESB služeb od vývojových systémů přes systémy pro zkoušení a testování až k distribuovanému produkčnímu prostředí. Důležitým požadavkem je, aby praktická realizace umožňovala tento přechod uskutečnit co možná nejjednodušší a bez chyb.



Obr. 4: Prvky Sonic ESB pro vývoj, zavádění a správu ostrého provozu ESB

Vývojové systémy

Vývojové systémy jsou znázorněny v levé části obr. 4. Horní část této části zobrazuje vývojářské nástroje, zatímco spodní část označená *Vývojová ESB* definuje základní komponenty ESB infrastruktury, které jsou využívány v každé fázi životního cyklu ESB od vývoje přes zkoušení a testování až k ostrému provozu v produkčním prostředí.

Samotné pracovní prostředí vývojáře (Workbench) poskytuje sadu nástrojů pro vývoj a zavádění ESB včetně tvorby na míru šitých implementací příslušného typu ESB služby. Takovým typem zákaznické ESB služby by mohla být například žurnálová služba z obr. 2 znázorňujícího příklad SOA aplikace použité pro integraci a mediaci vícekanálového finančního kapitálového procesu.

Pomocí sady editorů zdrojů (Resource Editors) a v souladu s vývojářskými prostředími třetích stran a standardizovanými vývojářskými prostředími pak mohou vývojáři vytvářet soubory zdrojů (Resource Files), které jsou uchovávány v úložišti ESB (konfiguračním úložišti).

Workbench obsahuje i konfigurační soubory potřebné pro projekt vývoje specifické služby využitelné v kompozitní nebo ESB aplikaci. V příkladě obchodování by mezi ně patřily soubory popisující ESB proces a jeho WSDL soubor, které lze využít

k vystavení procesu jako webové služby. Kromě jiných zdrojových souborů by mohl také obsahovat implementaci žurnálové služby.

Čím více editorů mají vývojáři k dispozici, tím snáze mohou zvládnout návrh a vývoj ESB. Editor ESB procesu (ESB Process Editor) může pomoci při vizuální tvorbě a editaci definic podnikového procesu se vzájemným provázáním různých ESB služeb. Editor XSLT slouží pro vytváření XSLT Stylesheetu definujícího XML transformaci a editor směrovacích pravidel (Routing Rules Editor) je určen pro tvorbu konfiguračního souboru využívaného službami směřujícími zprávy podle obsahu.

Konzola pro řízení vývoje poskytuje správný rámec pro konfiguraci SOA infrastruktury pro ESB služby, ESB kontejnery a podporující prvky multiprotokolového komunikačního serveru. Řídící konzola umožňuje konfiguraci nových instancí ESB služeb a ESB kontejnerů a dovoluje tak snadný přístup „configure-and-go“ ke správě SOA infrastruktury. Použitím této konzoly by například ESB propojení mezi všemi službami v obchodovacím procesu na obr. 2 mohlo být nastaveno na stav „once-and-only-once“ jako prevence ztráty jakékoli zprávy.

Zkušební a testovací systémy

Zkušební a testovací systémy spolu s nástroji pro zavádění zobrazuje střední část obr. 4. Nástroj pro zavádění (Deployment Tool) slouží k realizaci přechodu konfigurací z vývojového prostředí přes zkušební a testovací prostředí až do provozního prostředí.

Vybrané zdrojové soubory a konfigurace jsou převáděny vytvářením dočasných archivních souborů a využitím souboru pro mapování zavádění (Deployment Map File) umožňujícího mapování parametrů služeb a messagingu tak, aby v cílovém prostředí, kde se nastavení kvality služeb, bezpečnostní pravidla a ESB adresy mohou lišit, pracovaly správně. Schéma ukazuje postup a zavádění konfigurací z vývojové adresářové služby do adresářové služby pro zkoušení a testování.

Produkční systémy

Zaváděcí nástroje jsou také využity k postupu a nasazování ESB konfigurací ze zkušební a testovací adresářové služby do produkčního ESB úložiště. Produkční systémy jsou znázorněny v pravé části obr. 4.

Řídící konzola slouží k umisťování ESB kontejnerů a jejich konfigurovaných služeb do různých distribuovaných systémů, v nichž jsou zavedena příslušná pravidla interakce, transportní vazby a bezpečnostní pravidla pro komunikační vrstvu.

ESB poskytuje vysoce spolehlivou podporu distribuovaných systémů. Pro poskytování lokálních kopií konfiguračních souborů využívaných ESB kontejnery a ESB službami vykonávanými v distribuovaných systémech se využívá cache ESB úložiště. Distribuované systémy tak mohou být zároveň vysoce výkonné i spolehlivé a navíc mohou být v nepřetržitém provozu i v okamžiku, kdy centrální systémová adresářová služba není k dispozici nebo je nedostupná. Stejně prostředí je k dispozici také v prostředích pro zkoušení a testování, resp. pro vývoj (není znázorněno ve schématu).

Provoz více různých distribuovaných systémů v ESB umožňuje škálovat kapacitu ESB služeb na více strojů a míst. Každý distribuovaný systém sdílí stejnou konfiguraci pro reporting selhání a posílání stavových hodnot zpět na řídicí konzoli.

Pokud by mezi kterýmikoli dvěma kroky uvedeného procesu finančního obchodování bylo potřeba nárazově zvýšit výkonnost aplikace, dala by se transformační služba replikovat na více exekučních strojů a tyto stroje by se zapojovaly při špičkových zatíženích.

Administrátor používá ESB služby a ESB procesy k automatickému ošetření chyb a upozornění ve stylu SOA, využívá tak stále stejnou ESB infrastrukturu. Například v případě chyby v procesním kroku zajišťovaném službou pro zajištění shody obchodního procesu s předpisy a vyhláškami (viz obr. 2), lze koncový bod této služby pro normální chybu konfigurovat tak, aby směřoval chybu zpět k žadateli s využitím sémantiky reply-to. Alternativně by mohl být definován separátní proces pro výjimku, na jehož vstupní koncový bod by byla chyba poslána.

Výhody popsaného přístupu

Výhody popsaného přístupu k implementaci vývoje, zavádění a řízení ESB spočívají zejména v tom, že:

- bohatá sada vývojových a řídicích nástrojů umožňuje snadné nasazení způsobem „configure-and-go“,
- vývojové nástroje poskytují podporu životního cyklu ESB a umožňují automatizované mapování konfigurací a pohyb implementačních prostředků z vývojového prostředí přes zkušební a testovací prostředí do produkčního prostředí,
- distribuované prováděcí prostředí včetně vytváření lokálních kopií metadat umožňuje provoz vysoce výkonných, škálovatelných a nepřetržitě dostupných systémů.