# 38. Języki orkiestracji (BPEL) i choreografii (CDL) procesów biznesowych

Opisując języki orkiestracji oraz choreografii procesów biznesowych konieczne jest wyróżnienie głównych różnić w tych dwóch podejściach, ponieważ mają one znaczący wpływ na konstrukcję oraz mechanizmy stosowane w ramach języków wykorzystywanych do opisu procesów biznesowych w ramach tych dwóch podejść.

## Orkiestracja

Zastosowanie orkiestracji w procesie modelowania procesów biznesowych wiąże się z opisem procesu z perspektywy zewnętrznego koordynatora, nazywanego projektantem orkiestracji. W tym modelu cały proces jest nadzorowany przez tego koordynatora i on zajmuje się wywoływaniem konkretnych podprocesów i aktywności w ramach zdefiniowanego procesu biznesowego. W tej interpretacji jedynie koordynator zdaje sobie sprawę z faktu istnienia procesu biznesowego oraz tego, że jest ok wykonywany. Aktywności (która może być realizowana, np. poprzez usługę webową) wykonywane w ramach procesu biznesowego nie zdają sobie sprawy z tego, że biorą udział w realizacji pewnego procesu, ani z tego w jakim celu są wywoływane, w związku z tym wywołanie w swoje wywołanie traktują jako pojedyncze żądanie nie wykonywane w ramach jakiegokolwiek kontekstu. Z tego względu cała komunikacja, przepływ kontroli oraz przepływ danych są realizowane poprzez centralnego koordynatora, który w ramach procesu dba o jego poprawne wykonanie i reaguje na potencjalne wyjątki występujące w trakcie jego wykonania, w związku z czym tylko on zna cały plan wykonania procesu biznesowego. Ze względu na to orkiestrację można uznać za scentralizowany modelu opisu procesów biznesowych, który zawiera dokładny opis wykorzystywanych aktywności oraz kolejność i wykonywania. Ten sposób opisu można porównać do sposobu w jaki dyrygent dyryguje orkiestrą – stąd odniesienie do orkiestracji w nazwie.

## Choreografia

Choreografia przedstawia zupełnie inny model opisu procesów biznesowych. W odróżnieniu od orkiestracji w przypadku choreografii nie istnieje żaden centralny element zarządzający wykonaniem, a poszczególne aktywności są wykonywane w modelu peer-to-peer. Ze względu na takich sposób opisu procesu każdy uczestnik procesu musi dostać jego opis oraz wszelka komunikacja pomiędzy uczestnikami musi być jawna, aby możliwe było śledzenie wykonania procesu przez wszystkich jego uczestników. Proces biznesowy zdefiniowany w ten sposób musi definiować rolę każdego uczestnika procesu względem pozostałych uczestników realizujących proces. W celu zapewnienia tej specyfikacji proces zdefiniowany w postaci choreografii powinien zawierać szczegółowy opis reakcji na komunikaty przesyłane w ramach procesu dla każdego procesu. Opis ten ma na celu zapewnienie odpowiedniej kolejności komunikacji oraz odpowiedniego czasu komunikacji. Ze względu na to proces opisany z wykorzystaniem choreografii zawiera globalny opis procesu wykonania, w którego realizację muszą się wpasować wszyscy jego uczestnicy. Podobnie jak w przypadku orkiestracji ma to swoje odniesienie w rzeczywistości. Tancerze tańczący zgodnie z pewną choreografią nie mają żadnego centralnego koordynatora tylko realizują wcześniej zdefiniowany globalny „plan” w postaci choreografii, reagując jednocześnie na zachowanie otaczających ich tancerzy.

## Definiowanie procesów biznesowych

Definiując procesy biznesowe możemy wyróżnić dwa główne typy procesów: *procesy abstrakcyjne* i *procesy wykonywalne*.

***Procesy abstrakcyjne*** pełnią funkcję opisową. Opisują one proces biznesowy bez wnikania w detale realizacji poszczególnych aktywności i podprocesów. Oznacza to, że proces abstrakcyjne proces biznesowy opisuje co ma zostać wykonane ale nie definiuje tego jak to ma zostać wykonane.

***Proces wykonywalny*** można traktować jak ukonkretnienie procesu abstrakcyjnego. Proces ten poza definicją tego co ma zostać wykonane definiuje sposób realizacji poszczególnych elementów procesu, ukonkretniając je poprzez definicje wykorzystania usług webowych, funkcji wbudowany w silnik wykonawczy realizujący proces itp., tak aby możliwe było wykonanie całego procesu w ramach jakiegoś środowiska wykonawczego.

## WS-BPEL

WS-BPEL (Web Service Business Process Execution Language) jest językiem opisu procesów biznesowych, opisywanych w postaci orkiestracji, pozwalającym na tworzenie zarówno procesów abstrakcyjnych jak I procesów wykonywalnych. Został on opracowany przez środowisko komercyjne i ma swoje korzenie w języka XLANG opracowanym przez Microsoft oraz WSFL opracowanym przez IBM. Język BPEL powstał w wyniku konsultacji i wspólnych pracy prowadzonych między innymi przez IBM, Microsoft oraz BEA Systems i został zatwierdzony jako standard OASIS. Warto wspomnieć, że nazwa WS-BPEL odnosi się do języka BPEL w wersji 2.0.

Język WS-BPEL bazuje na koncepcji *Programming In the large*, zgodnie z którą oprogramowanie jest wytwarzane przez duże grupy osób i zespoły projektowe, a elementy oprogramowania są wytwarzane jego moduły o dobrze zdefiniowanych interfejsach, które mogą być w łatwy sposób wymieniane i wykorzystywane do budowanie aplikacji.

Język WS-BPEL został stworzony na bazie języka XML, a jego gramatyka jest definiowana za pomocą schematu XML opisanego w plikach XSD. WS-BPEL bazuje głównie na języku opisu interfejsów usług webowych WSDL oraz na jego mechanizmach. Można go traktować jako behawioralne rozszerzenie WSDL wykorzystującą podejście opierające się na procesach abstrakcyjnych. Język ten wyraża powiązania pomiędzy wieloma wywołaniami usług poprzez definiowanie powiązań przepływu kontroli oraz przepływu danych pomiędzy nimi. Do zapewnienia tego wykorzystuje rozproszony równoległy model przetwarzania wykorzystujący zmienne.

Język ten zapewnia pełne wsparcie do wysyłania i odbierania komunikatów i komunikacji z usługami. Umożliwia programowanie strukturalne, manipulację zmiennymi zdefiniowanymi w ramach procesu oraz definiowanie ich zakresów. Pozwala również kompozycję oraz dekompozycję procesów, a także na wykorzystanie w ich ramach mechanizmów transakcji

Warte uwagi jest to, że język ten nie posiada żadnej standardowej notacji graficznej, ponieważ założenia projektowe języka nie zakładały nigdy stworzenia takiej notacji . Niektórzy mylnie uznają BPMN za notację dla BPEL. Błąd ten wynika z tego, że wiele komercyjnych narzędzi do definicji procesów biznesowych w BPEL wykorzystuje notację graficzną BPMN. Niestety, notacje te nie są w pełni kompatybilne i kod tworzony w ten sposób nie jest wysokiej jakości.

Głównymi standardami, na których opiera się język są standardy:

* *WSDL* (Web Service Description Language) – język opisu interfejsów usług webowych wykorzystywany w ramach języka
* *XPath* (XML Path Language) – język służący do wyszukiwania elementów w XML, który pozwala również na obliczanie wartości na podstawie danych z pliku XML.
* *XML Schema* – język do definiowania schematów i struktur danych zapisanych w postaci XML wykorzystywany do definiowania typów danych w ramach interfejsów usług webowych
* *XML* – uniwersalny język formalny przeznaczony do reprezentowania różnych danych w strukturalizowany sposób.

Głównym elementem języka zawierającym opis całego procesu jest element “*Process*”. W ramach tego elementu możemy wymienić kilka głównych elementów takich jak:

* **Acitivies** (aktywności) – są to podstawowe elementy definiujące zachowanie procesu. BPEL wyróżnia dwa główne grupy aktywności:
  + *Basic Activities (Aktywności proste)*
  + *Structured Activities (Aktywności strukturalne)*

Główna różnica pomiędzy tymi aktywnościami polega na ich charakterze. Aktywności strukturalne zawierają w sobie aktywności proste i definiują logikę biznesową, która tworzy powiązania pomiędzy nimi. Aktywności proste z kolei skupiają się na zrealizowaniu konkretnych zadań przed nimi stawianych i w żaden sposób nie definiują logiki biznesowej lub zachowania innych aktywności.

Wśród aktywności prostych możemy wyróżnić:

* + *Invoke* – wywoływanie web service’ów
  + *Receive* – aktywność blokująca, która oczekuje na otrzymanie pewnej wiadomości. Aktywność ta będzie blokowała dalsze wykonanie procesu do momentu jej otrzymania.
  + *Reply* – aktywnośc wykorzystywana do wysłania odpowiedzi na wiadomość uzyskaną w ramach aktywności receive
  + *Assign* – aktywność mająca na celu przypisanie wartości zmiennym zdefiniowanym w ramach procesu
  + *Throw* – aktywnośc wykorzystywana do informowania o wystąpieniu wyjątku podczas wykonania procesu
  + *Validate* – aktywność wykorzystywana do walidacji danych zapisanych w zmiennych względem schematów XML
  + *Pick* – aktywność ta czeka na wystąpienie dokładnie jednego zdefiniowanego zdarzenia ze zbioru zdarzeń i wykonuje powiązaną z tym zdarzeniem aktywność

Wśród strukturalnynych aktywności możemy wyróżnić

* + *Squance* – grupuje aktywności do wykonania sekwencyjnego
  + *Flow* – grupuje aktywności do wykonania równoległego
  + *If-else*– definiuje strukturę zawierającą warunek logiczny
  + *While* – zwykła pętla while w ramach procesu
  + *RepeatUntil* – analogicznie do while, z tym że ma konstrukcję, która wymusza co najmniej jednokrotne wykonania ciała pętli
  + *ForEach* – standardowy foreach
* **Partner links** (linki partnerskie)

Są to elementy WS-BPEL, które pozwalają na definiowanie interakcji z innymi partnerami biznesowymi. Linki partnerskie specyfikują interfejs WSDL, który powinien zostać wykorzystany w ramach komunikacji z partnerem biznesowym. Ważna uwagą jest to, że jeden partner biznesowy może występować w wielu linkach partnerskich. W związku z tym każdy link partnerski jest opisany przez typ (identyfikujący partnera) oraz role (identyfikującą rodzaj komunikacji z partnerem). Linki partnerskie są głównym mechanizmem pozwalającym na wywoływanie usług webowych w ramach WS-BPEL.

* **Variables** (zmienne)

Zmienne są wykorzystywane do przechowywaniu stanu w jakim znajduje się process w trakcie wykonania. Dane w WS-BPEL są danymi typizowanymi. Wartości zmiennych wykorzystywanych w procesach mogą pochodzić z dwóch źródeł: z wymiany komunikatów z partnerami biznesowymi lub z prywatnych danych wykorzystywanych w ramach procesu. Wszystkie zmienne definiowane w ramach procesu muszą być definiowane jako proste typy zdefiniowane w ramach XML lub jako typy złożone opisane schematem XML. Język udostępnia również narzędzia do manipulowania zmiennymi oraz wykonywania prostych operacji z ich wykorzystaniem. Jest to możliwe dzięki mechanizmowi pozwalającymi na implementację języków, które mogą być bezpośrednio wykonywane w ramach niektórych typów aktywności. Domyślnymi językiem wykorzystywanym do tych zadań jest XPath 1.0.

Zmienne mogą być deklarowane bezpośrednio w ramach procesu, wtedy są widoczne dla całego procesu i dla wszystkich akcji w nim wykonywanych. Możliwe jest zdefiniowanie również zakresów widoczności (scope) zmiennych w ramach procesu.

* **Foult Handlers**

Ze względu na powszechne podczas wykonywania ryzyko wystąpienia błędu BPEL definiuje mechanizmy obsługi błędów. Mechanizm taki może być dołączony do zdefiniowanego zakresu (scope), procesu lub do samego wywołania pewnej usługi. Mechanizm obsługi błędów jest uruchamiany wraz z rozpoczęciem wykonywania zakresu procesu w ramach, którego jest zdefiniowany i jest wywoływany w przypadku wystąpienia błędu lub po prostu pomijany. Obsługa błędów jest definiowana podobnie jak to ma miejsce w językach programowania takich jak Java lub C#. Definiujemy rodzaj błędu po czym w znacznikach *Catach* lub *catchAll* definiujemy sposób obsługi błędu poprzez wskazanie aktywności, która ma zostać uruchomiona w przypadku wystąpienia błędu.

* **Compensation handlers**

W przypadku wytąpienia błędu w transakcji wykonywanej w ramach procesu WS-BPEL definiuje mechanizm pozwalający na kompensację wykonywanej transakcji. W ramach tego znacznika można zdefiniować sposób reakcji w przypadku wystąpienia błędu poprzez zdefiniowanie aktywności, które mają zostać wywołane takiej sytuacji.

* **Event handlers**

Mechanizmy pozwalające na obsługę zdarzeń w WS-BPEL mają na celu przede wszystkim umożliwienie wywoływanie usług asynchronicznych, które dają odpowiedź dopiero po jakimś czasie. Czekanie na takie usługi czasami nie jest konieczne do dalszej realizacji procesu, w związku z czym mechanizm pozwala na zdefiniowanie tego na jakie zdarzenia ma oczekiwać mechanizm oraz jak ma na nie zareagować. Mechanizmy do obsługi zdarzeń mogą być definiowane w kontekście całego procesu lub pojedynczego zdefiniowanego zakresu.

## WS-CDL

WS-CDL (Web Service Choreography Description Language) jest językiem do specyfikacji procesów poprzez wykorzystanie opisu w postaci choreografii. Oznacza to, że definiuje on powiązania peer-to-peer pomiędzy elementami. Ze względu na to w ramach tego opisu definiowany jest zachowanie usług w perspektywie globalnej oraz to w jaki sposób usługi powinny ze sobą współpracować. Sprawia to, że nie ma możliwości, tak jak w BPEL, szczegółowego definiowania realizacji procesu oraz przechowywania jego stanu, np. w postaci zmiennych. Pomimo braku tych możliwości wynikających z charakteru choreografii WS-CDL stara się dostarczyć mechanizmy, które mimo wszystko umożliwią wykorzystanie takich informacji oraz dostęp do nich.

Opisując język WS-CDL jako jego główne cech charakterystyczne można wskazać to, że opisuje on globalną wymianę komunikatów pomiędzy uczestnikami komunikacji bez przyjmowania żadnego specyficznego punktu widzenia. Poza tym WS-CDL zapewnia zestaw reaktywnych reguł, które mogą zostać wykorzystane przez każdego z uczestników komunikacji do obliczenia stanu komunikacji.

Pomimo tego, że podstawowy opis WS-CDL oraz samych języków opierających się na choreografii nie wspomina o tym, proces zamodelowany za pomocą choreografii może być świadomy jakie dane są przekazywane w ramach wiadomości. W przypadku WS-CDL informacji tych dostarczają m.in. tokeny zawarte w wiadomościach. Wiadomości przesyłane w ramach WS-CDL z wykorzystaniem tokenów zawierają następujące informacje:

* **informationType**: alias wskazujący na plik WSDL lub XSD definiujący typ danych
* **token**: zmienna powiązana z procesem choreografii
* **tokenLocator**: wyrażanie wskazujące na daną wyrażone za pomocą XPath

WS-CDL definiując komunikację w ramach choreografii definiują ją poprzez :

* ***Role*** – specyfikuje operacje zaimplementowane przez uczestnika, które zostaną wykonane w ramach jego roli w choreografii. W Odróżnieniu od WS-BPEL wykorzystanie opisów usług w formacie WSDL nie jest tutaj obligatoryjne.
* ***Powiązania*** – analizując proces dowolny proces choreografii można zauważyć, że może on zostać podzielony na zbiór relacji binarnych. Powiązania mają na celu wyrażenie właśnie tych binarnych reflacji pomiędzy uczestnikami.
* ***Uczestników*** –uczestnik implementuje jedną lub więcej roli w ramach choreografii.

PS. O WS-CDL jest ogólnie mało konkretnych informacji, bo mało kto tego tak naprawdę używa (to też jest w sumie jakaś wiedza, która może się przydać na egzaminie). Jak coś jeszcze znajdę to dorzucę, ale jest to wątpliwe. Ze swojej strony na egzaminie wchodząc głębiej w temat WS-CDL starałbym go się porównać do WS-BPEL, co de facto sprowadzałoby się do porównania orkiestracji z choreografią.