



**AKADEMIA GÓRNICZO-HUTNICZA IM. STANISŁAWA STASZICA W KRAKOWIE**

**WYDZIAŁ ELEKTROTECHNIKI, AUTOMATYKI,  
INFORMATYKI I INŻYNIERII BIOMEDYCZNEJ**

KATEDRA INFORMATYKI STOSOWANEJ

Praca dyplomowa magisterska

*Analiza możliwości adaptacji środowisk uruchomieniowych dla logiki  
biznesowej na platformach mobilnych.*

*Analysis of possible adaptation or business logic runtimes for mobile  
platforms.*

Autor: *Tomasz Landowski*  
Kierunek studiów: *Informatyka*  
Opiekun pracy: *dr hab. Grzegorz J. Nalepa*

Kraków, 2014

*Oświadczam, świadomy(-a) odpowiedzialności karnej za poświadczenie nieprawdy, że niniejszą pracę dyplomową wykonałem(-am) osobiście i samodzielnie i nie korzystałem(-am) ze źródeł innych niż wymienione w pracy.*

*Serdecznie dziękuję żonie i córce za cierpliwość.*



## Spis treści

<b>1. Wstęp</b>	7
1.1. Cel pracy	7
1.2. Struktura pracy	7
<b>2. Analiza problemu modelowania procesów biznesowych</b>	9
2.1. BPM	9
2.2. Projektowanie BPM	10
2.3. Wykonywanie BPM	10
2.4. Podsumowanie	10
<b>3. Analiza aplikacji na platformy mobilne</b>	11
3.1. Rodzaje aplikacji mobilnych	11
3.2. Tryby online/offline	12
3.3. Cechy specyficzne dla aplikacji mobilnych	12
3.4. Komunikaty Push	13
<b>4. BPEL „inteligentne” narzędzie integracyjne</b>	15
4.1. Cechy języka BPEL	15
4.2. Orkiestracja czy choreografia?	16
4.3. Silniki uruchomieniowe WS-BPEL	16
4.4. Podsumowanie	17
<b>5. BPEL vs Aplikacje mobilne</b>	19
5.1. Adaptacja środowisk uruchomieniowych na platformach mobilnych	19
5.2. Wykorzystanie procesów biznesowych w systemach back end’owych	20



# **1. Wstęp**

...

## **1.1. Cel pracy**

Celem poniższej pracy jest ...

## **1.2. Struktura pracy**

W rozdziale 2 ...





## 2. Analiza problemu modelowania procesów biznesowych

Modelowanie procesów biznesowych jest pojęciem bardzo ogólnym, nie zawiera w sobie żadnych szczegółów technicznych, określa jedynie pewne specyficzne podejście do rozwiązywania problemów informatycznych. W podejściu tym podczas analizy, główny nacisk kładziony jest na wyłonienie procesów występujących w analizowanym problemie. Proces jest tutaj rozumiany jako zbiór następujących w określonej kolejności operacji prowadzących do osiągnięcia konkretnego celu. Najczęściej modelowanie procesów biznesowych jest rozważane w kontekście działania konkretnego przedsiębiorstwa, w którym wyłonienie procesów oraz odpowiednie nimi zarządzanie ma kluczowe znaczenie dla osiągnięcia sukcesu.

### 2.1. BPM

W literaturze pojęcie modelowania oraz zarządzanie procesami biznesowymi w kontekście przedsiębiorstw występuje pod skrótem BPM, którego angielskie rozwinięcie to Business Process Management (pl. Zarządzanie Procesami Biznesowymi). Idea BPM jest bardzo popularna i szeroko rozwijana w środowisku IT, powstała nawet organizacja pod nazwą *European Association of Business Process Management* zajmująca się rozwojem i promocją BPM. Na stronie internetowej organizacji znaleźć możemy oficjalną definicję BPM, która mówi, że w skład Zarządzania Procesami Biznesowymi wychodzi:

- projektowanie,
- wykonywanie,
- dokumentacja,
- pomiar,
- monitorowanie,
- kontrola

zautomatyzowanych oraz niezautomatyzowanych procesów biznesowych.[1]

W środowisku IT zdarza się, że BPM traktowany jest jako osobna klasa systemów informatycznych, obok ERP, MES, CRM itd. Z punktu widzenia przedstawionej wyżej definicji trudno zgodzić się z takim

podejściem, można jednak zauważyć że BPM de facto może zostać wykorzystany do realizacji każdego z wymienionych rodzajów oprogramowania, lub posłużyć jako narzędzie integrujące wyżej wymienione klasy systemów, w celu stworzenia globalnego systemu zarządzania przedsiębiorstwem. [2]

## 2.2. Projektowanie BPM

Zdecydowanie jednym z najistotniejszych etapów w tworzeniu systemu opartego o procesy biznesowe jest etap projektowania. Jednym z najbardziej popularnych narzędzi do tego celu jest BPMN (*Business Process Modeling Notation*), która doczekała się już dwóch wersji. W kontekście niniejszej pracy magisterskiej notacja BPMN nie jest szczególnie ważna jednak została wspomniana ze względu na jej wykorzystanie w przykładach. Warto zwrócić uwagę na fakt, że BPMN nie jest w żadnym stopniu implementacją, jest to jedynie sposób wizualizacji procesu za pomocą diagramu w postaci obrazka.

## 2.3. Wykonywanie BPM

Jak zostało wspomniane na samym początku Modelowanie Procesów Biznesowych jest pojęciem ogólnym nie wskazującym konkretnej technologii, która ma posłużyć do wykonania procesów biznesowych. Z definicji można jednak wnioskować, że środowiskiem do wykonywania procesów biznesowych w przedsiębiorstwach na pewno nie powinna być prosta aplikacja desktopowa czy mobilna. Biorąc pod uwagę wachlarz zastosowań BPM nie trudno dojść do wniosku, że najbardziej odpowiednim środowiskiem dla procesów biznesowych są środowiska aplikacji rozproszonych. Powyższe stwierdzenie potwierdza przegląd istniejących systemów uruchomieniowych dla procesów biznesowych. W ogromnej większości są to aplikacje webowe.

## 2.4. Podsumowanie

Z punktu widzenia niniejszej pracy magisterskiej, głównym wnioskiem płynącym z powyższego opisu BPM jest skala problemów rozwiązywanych przez to podejście oraz ich usytuowanie w środowisku aplikacji rozproszonych.

### **3. Analiza aplikacji na platformy mobilne**

Wzrost popularności smartphonów oraz urządzeń przenośnych takich jak tablet wpłynął na powstanie specyficznej klasy systemów informatycznych zwanych aplikacjami mobilnymi. Aplikacje mobilne cechuje ukierunkowanie na rozwiązywanie wąskiego rodzaju problemów przy pomocy urządzenia dostępnego dla użytkownika niemal 24h na dobę. Szczególną popularność zyskały aplikacje mobilne ukierunkowane na użytkowników (TODO jak nazwać użytkownika typu powszechny Jan Kowalski). Coraz większą popularność zyskują również aplikacje biznesowe ukierunkowane np. na wymianę danych pomiędzy pracownikami przedsiębiorstwa w celu rozwiązania konkretnego zadania.

#### **3.1. Rodzaje aplikacji mobilnych**

Ze względu na rodzaj zastosowania aplikacje mobilne można sklasyfikować w następujący sposób:

- Samodzielne aplikacje – aplikacje wykorzystujące tylko i wyłącznie lokalne zasoby urządzenia, korzystające z lokalnej bazy danych bez połączenia z systemami zewnętrznym, często nie wymagające połączenia do internetu. Dobrym przykładem takiej aplikacji może być prosty notatki.
- Aplikacje klienckie – są to aplikacje bazujące na komunikacji z systemami zewnętrznymi przy pomocy jakiegokolwiek interfejsu komunikacyjnego, najczęściej połączenia HTTP. Zazwyczaj aplikacje te wykorzystują również lokalną bazę danych umożliwiając użytkownikom pracę z aplikacją w przypadku braku połączenia z systemem zewnętrznym.
- Internetowe – najczęściej strony www, aplikacje te nie wykorzystują lokalnych zasobów urządzeń mobilnych, ich rolą jest udostępnienie prostego interfejsu użytkownika systemu dla użytkowników aplikacji mobilnych.
- Gry – szczególny przypadek samodzielnych aplikacji ukierunkowanych głównie na wykorzystanie lokalnych zasobów urządzenia w celach rozrywkowych, szczególnie eksploatowana w tego typu aplikacjach jest karta graficzna urządzenia.

### 3.2. Tryby online/offline

Najbardziej docenianym rodzajem aplikacji mobilnych z punktu widzenia biznesu są aplikacje klienckie. W przypadku tych aplikacji szalenie istotny jest problem połączenia z systemami zewnętrznymi. Wyróżniamy tutaj dwa tryby pracy aplikacji:

- Tryb online – urządzenie udostępnia przynajmniej jeden typ komunikacji bezprzewodowej (Wi-fi, Bluetooth, łącza podczerwieni (IrDa), GPRS). W trybie online aplikacje mobilne mają bezpośredni dostęp do zewnętrznych i zdalnych źródeł danych, innych urządzeń mobilnych lub stacjonarnych systemów komputerowych.
- Tryb offline – urządzenie ma bezpośredni dostęp tylko do lokalnie przechowywanych informacji. Dane te mogą być jednak synchronizowane z innymi urządzeniami w czasie krótkich sesji komunikacyjnych. Wymiana danych podczas synchronizacji może następować w obu kierunkach.

### 3.3. Cechy specyficzne dla aplikacji mobilnych

Opisując aplikacje mobilne nie można zapomnieć o cechach specyficznych, które są szalenie ważne z punktu widzenia projektanta i programisty aplikacji mobilnych. To właśnie dzięki tym cechom aplikacje mobilne posiadają tę a nie inną specyfikę:

- Ograniczone zasoby sprzętowe – urządzenie przenośne pod względem zasobów sprzętowych nigdy nie zastąpi komputerów stacjonarnych a tym bardziej urządzeń serwerowych. Właśnie ze względu na tę cechę aplikacje mobilne ukierunkowane są na rozwiązywanie małych - jednostkowych problemów. Projektant aplikacji mobilnych powinien szczególnie zwrócić uwagę na ten aspekt w przypadku projektowania interfejsu do komunikacji z systemami zewnętrznymi, aplikacje mobilne powinny wykorzystywać lekkie interfejsy komunikacyjne np. REST, aby jak najmniej obciążać kartę sieciową urządzenia.
- Uproszczony interfejs użytkownika dostosowany do ekranów dotykowych – cecha ta jest bardzo ważna głównie dla projektanta interfejsu użytkownika, ale ma również znaczący wpływ na rodzaj zadań rozwiązywanych za pomocą tych aplikacji.
- Przerwy w działaniu aplikacji – aplikacje mobilne w przeciwieństwie do innych rodzajów systemów są szczególnie narażone na przerwy w działaniu, najprostszym przykładem może być rozmowa telefoniczna przychodząca w trakcie pracy z aplikacją. Najczęściej w takich przypadkach aplikacje przechodzą w tryb uśpienia i nie ma pewności czy praca zostanie wznowiona. Programiści powinni pamiętać o zapamiętywaniu poszczególnych stanów aplikacji aby użytkownik nie tracił wykonanej pracy.

- Dostęp do GEO lokalizacji – aplikacje mobilne oprócz ograniczeń posiadają również wiele cech dodatkowych wyróżniających je wśród innych systemów, jedną z nich jest dostęp do GEO lokalizacji. Funkcjonalność ta bardzo szeroko wykorzystywana jest przez systemy zwane Context Aware Middleware.
- Dostęp do aparatu/kamery – jest to kolejna cecha rozszerzająca możliwości aplikacji mobilnych, może być wykorzystana np. jako narzędzie do skanowania kodów kreskowych, albo sposób dokumentacji pracy wykonanej przez użytkownika systemów mobilnych.
- Specyficzna architektura dla różnych systemów operacyjnych – bardzo wiele cech aplikacji mobilnych jest wspólna w przypadku wszystkich systemów operacyjnych dostępnych na rynku, trzeba mieć jednak świadomość, że jest również wiele cech specyficznych dla różnych platform mobilnych.

### 3.4. Komunikaty Push

Ze względu na zmienne warunki w dostępie do mediów komunikacyjnych, aplikacje mobilne wykorzystują w głównej mierze komunikację pull, (czyli pobieranie informacji na żądanie) z systemami zewnętrznymi. Najpopularniejsze platformy mobilne jak Android, iOS oraz Windows Phone udostępniają również alternatywny sposób komunikacji - push.

Komunikacja push odbywa się jednak zawsze za pośrednictwem serwisów udostępnianych przez platformy mobilne do tego celu. Komunikacja ta posiada wiele ograniczeń a już na pewno najważniejszym z nich jest brak pewności że komunikat zostanie w ogóle dostarczony. Komunikacja push powinna być zatem wykorzystywana jedynie do notyfikacji użytkownika o jakimś zdarzeniu a nie do przesyłania informacji kluczowych dla działania aplikacji.



## 4. BPEL „inteligentne” narzędzie integracyjne

BPEL (*Business Process Execution Language*) a właściwie pełna nazwa WS-BPEL, której rozwinięcie to *Web Services Business Process Execution Language* jest językiem opartym o składnię XML, służącym do wykonywania procesów biznesowych w środowisku usług sieciowych. Język BPEL jest ustandaryzowany przez organizację OASIS (*Organization for the Advancement of Structured Information Standards*).

Organizacja OASIS definiuje język BPEL jako: *Umożliwiający użytkownikom opis aktywności procesów biznesowych jako usługi sieciowe oraz definicję w jaki sposób usługi te mogą być połączone między sobą w celu wykonania zadania.* [3]

Jak argumentuje OASIS, język BPEL powstał z konieczności stworzenia dodatkowej warstwy integracyjnej, która wykorzysta potencjał dostarczony przez usługi sieciowe oraz procesy biznesowe. Warto tutaj wspomnieć o historii powstania BPEL. Microsoft i IBM dostrzegając potrzebę stworzenia technologii pozwalającej definiować przepływ wywołań usług sieciowych stworzyli osobne, ale bardzo podobne języki, służące do tego celu, nazwane WSFL (*Web Service Flow Language*) oraz Xlang. W miarę wzrostu popularności BPM zdecydowali się połączyć siły aby stworzyć wspólny standard (opisanych za pomocą WSDL. Zarówno Microsoft jak i IBM (zresztą nie tylko oni) dostrzegli, że architektura SOA (*Service Oriented Architecture*) wymaga wprowadzenia czegoś co pozwoli składać klocki jakimi są usługi sieciowe w jedną zgrabną całość i właściwie to było genezą powstania BPEL'a.

Dotychczasowy model komunikacji udostępniany przez usługi sieciowe był niewystarczający, głównie z powodu braku zachowania stanu podczas prostej komunikacji żądanie-odpowiedź. Komunikacja ponadto była głównie jednokierunkowa. Model komunikacji dla biznesu wymagał natomiast sekwencyjnej wymiany wiadomości pomiędzy wieloma węzłami, operacje wykonywane na węzłach mogły trwać bardzo długo, dlatego istnieje również konieczność obsługi tego typu procesów przy pomocy zapamiętywania stanu, oraz asynchronicznego wywoływania serwisów. [4]

### 4.1. Cechy języka BPEL

- Definiowane procesy biznesowe komunikują się z usługami sieciowymi przy pomocy WSDL 1.1 i same są usługami sieciowymi opisanymi przez WSDL 1.1.
- Do komunikacji pomiędzy usługami wykorzystywany jest protokół SOAP.



- Procesy zdefiniowane są w języku bazującym na XML.
- BPEL dostarcza funkcji do wykonywania prostych manipulacji na danych.
- Posiada wsparcie do identyfikowania instancji procesów.
- Posiada wsparcie dla podstawowego cyklu życia procesu.
- Wspiera transakcyjność przy wykorzystaniu podstawowych technik takich jak akcje kompensacyjne.

## 4.2. Orkiestracja czy choreografia?

BPEL jest językiem orkiestracji usług sieciowych, a nie choreografii. Podstawowa różnica polega na tym, że w choreografii usługi współpracują ze sobą bezpośrednio. W orkiestracji usługa komunikuje się wyłącznie z menedżerem orkiestracji (w tym przypadku: maszyna BPEL), który wie, gdzie i jak przekazać dalej komunikaty. Nie musi tym samym znać innych usług biorących udział w procesie.

Korzyści płynące z tego, że BPEL jest warstwa orkiestracji to [5]:

- podejście integracyjne do komunikacji pomiędzy aplikacjami, które uniezależnia aplikacje od siebie,
- umożliwia sterowanie przepływem, zarządzanie bezpieczeństwem oraz niezawodność komunikacji,
- co najważniejsze sposób zarządzanie i monitorowania procesów jest scentralizowany.

## 4.3. Silniki uruchomieniowe WS-BPEL

WS-BPEL dostarcza jedynie opisu w jaki sposób proces ma przebiegać oraz z jakimi usługami ma się komunikować, potrafi również zdefiniować proste operacje proceduralne jak tworzenie zmiennych, operacje przypisania, warunki czy pętle. Język sam w sobie jednak nie ma zbyt wielkiej wartości bez odpowiedniego środowiska uruchomieniowego. Procesy BPEL można traktować jako skrypty interpretowane, wykonywane przez maszyny procesów biznesowych w środowisku zgodnym ze standardem BPEL.

Najbardziej popularne środowiska uruchomieniowe BPEL to:

- ActiveVOS
- Apache ODE
- ExpressBPEL BPM

- BizTalk Server
- Open ESB
- Oracle BPEL Process Manager
- OW2 Orchestra
- Parasoft BPEL Maestro
- Petals BPEL Engine
- WebSphere Process Server

Do celów niniejszej pracy magisterskiej zostanie wykorzystany Apache ODE (*Apache Orchestration Director Engine*). Jest to narzędzie rozwijane przez społeczność Open Source, zgodne ze standardem WS-BPEL. Silnik Apache ODE został zaimplementowany przy wykorzystaniu języka Java, jest on w rzeczywistości standardową aplikacją webową uruchamianą na serwerach aplikacyjnych Java np. na serwerze Tomcat.

## 4.4. Podsumowanie

Należy pamiętać, że język BPEL nie jest jedynym słusznym rozwiązaniem służącym do implementacji procesów biznesowych. Bardzo często zdarza się, że projektanci systemów IT, którzy preferują podejście top-down design (pl. projektowanie od góry do dołu), zaczynając projektować system IT przy pomocy notacji BPMN. Następnie zaś próbują przełożyć modele procesów biznesowych na kod wykonywalny w postaci BPEL. Nie jest to jednak słuszne podejście. Procesy BPMN zawierają często bardzo wiele szczegółów opisujących zachowanie samych operacji jednostkowych, natomiast pomijają szczegóły techniczne dotyczące usług sieciowych. Zazwyczaj okazuje się, że usługa sieciowa która została by wygenerowana z diagramu BPMN w ogóle nie istnieje, a jej stworzenie może być mało sensowne.

Niestety nazwa BPEL jest w tym przypadku nie do końca trafna, ponieważ sugeruje, że BPEL jest językiem służącym do implementacji procesów biznesowych. W rzeczywistości jednak BPEL należy traktować jako narzędzie integracyjne. Narzędzie to ma służyć do definicji przebiegu komunikacji pomiędzy usługami sieciowymi, przy pomocy idei procesów biznesowych. Jest bardzo ważnym aby BPEL był stosowany właśnie w takich przypadkach, pozwoli to uniknąć poważnych błędów projektowych wynikających z zastosowania nieodpowiednich narzędzi.



## **5. BPEL vs Aplikacje mobilne**

W rozdziałach poprzednich sporo uwagi zostało skupione na temacie modelowania procesów biznesowych. Został również przybliżony temat aplikacji mobilnych, nadszedł czas aby odnieść się do tematu niniejszej pracy magisterskiej i spojrzeć na zestawienie tych dwóch technologii.

W postawionym problemie jak zwykle kluczowym jest odpowiednie podejście do tematu, uświadomienie sobie ograniczeń i odpowiednie dopasowanie technologii. Ograniczeń jest tutaj bardzo wiele głównie od strony aplikacji mobilnych.

Przykładów na wykorzystanie modelowania biznesowego w środowisku aplikacji mobilnych można wymyślić bardzo wiele, tym bardziej w przypadku aplikacji klienckich przeznaczonych dla biznesu. W tym ostatnim, środowisko mimo, że ukierunkowane na niewielkie aplikacje jest środowiskiem rozproszonym, w którym do tej pory procesy biznesowe sprawdzały się doskonale.

### **5.1. Adaptacja środowisk uruchomieniowych na platformach mobilnych**

Jednym z możliwych sposobów podejścia do tematu może być próba bezpośredniej adaptacji środowisk uruchomieniowych dla procesów biznesowych na platformach mobilnych. Bezpośrednia adaptacja w tym przypadku oznacza próbę uruchomienia jakiegoś lekkiego środowiska uruchomieniowego na smartphonie. Po krótkim zastanowieniu można powiedzieć, że pomysł może być realny w realizacji - zasoby sprzętowe smartphonów są coraz większe, języki programowania wykorzystywane do tworzenia aplikacji mobilnych są identyczne jak języki do tworzenia środowisk uruchomieniowych (Android - język Java, Windows Phone - język C#). Gdy jednak kontynuujemy przemyślenia bardzo szybko natrafiamy na mur. Jak wspomniano w rozdziale 3, komunikacja aplikacji mobilnych odbywa się przede wszystkim metodą pull, głównie ze względu na zmienne warunki dostępu do mediów komunikacyjnych. W jaki sposób zatem środowisko uruchomieniowe miałyby odbierać komunikację z zewnątrz? Jest przecież jeszcze metoda push, dostarczana przez dostawców platform mobilnych. Skorzystanie z niej jest jednak kiepskim pomysłem, głównie ze względu na brak gwarancji dostarczenia komunikatu. Podejście adaptacyjne zatem nie jest dobrym pomysłem.

## 5.2. Wykorzystanie procesów biznesowych w systemach back end'owych

Jak wspomniano wyżej najwięcej zastosowań dla procesów biznesowych w kontekście aplikacji mobilnych można dostrzec w aplikacjach klienckich. Należy się zatem zastanowić w jaki sposób działają typowe aplikacje klienckie. Z powodu ograniczeń opisanych w poprzednim akapicie komunikacja zazwyczaj nie odbywa się na zasadzie Peer-to-Peer, użytkownicy aplikacji klienckich jednak w jakiś sposób wymieniają między sobą informację. Jest to możliwe dzięki aplikacjom centralnym, zazwyczaj webowym, zwanych systemami back'endowymi. Aplikacje mobilne w takich przypadkach komunikują się jedynie z systemem back end'owym nie posiadając informacji osobie nawzajem. W tego rodzaju sposobie komunikacji można dostrzec bardzo wiele cech wspólnych z warstwą orkiestracji opisaną w rozdziale 4. Można zatem wyciągnąć wniosek, że procesy biznesowe w kontekście aplikacji mobilnych zostaną najefektywniej wykorzystane właśnie po stronie systemu back end'owego.

Podejście to można nazwać podejściem integracyjnym, w przypadku skorzystania z języka BPEL można sobie wyobrazić, że zrealizowany w ten sposób system back end'owy oprócz obsługi żądań aplikacji mobilnych mógłby komunikować się z innymi systemami przy wykorzystaniu usług sieciowych. Stworzony w ten sposób proces mógłby kontrolować przebieg komunikacji między aplikacjami mobilnymi traktując je jednocześnie na równi z innymi systemami. Niestety kolejny raz mimo sensownego pomysłu występują problemy w realizacji, mało tego ograniczenie kolejny raz jest takie samo. Proces BPEL jak opisano w rozdziale 4, do komunikacji poszczególnych węzłów wykorzystuje usługi sieciowe udostępniające odpowiednie interfejsy WSDL. Aplikacje mobilne nie są w stanie jednak udostępnić takich usług.

W tym przypadku istnieje sposób na poradzenie sobie z tym problemem, można stworzyć pewnego rodzaju adapter w postaci aplikacji webowej będącej warstwą pośrednią między procesami BPEL a aplikacjami mobilnymi. Taka warstwa pośrednia miałaby za zadanie z jednej strony udostępnienie usług sieciowych dla procesów biznesowych a z drugiej strony realizowała by komunikację z aplikacjami mobilnymi w celu przekazania żądań pochodzących od procesów biznesowych.

Opisanemu powyżej sposobowi wykorzystania procesów biznesowych w aplikacjach mobilnych została poświęcona niniejsza praca, zostanie on dokładniej opisany w kolejnych rozdziałach.