

Budowa celów architektury korporacyjnej IT dla środowisk zintegrowanych w sektorze publicznym

1. Streszczenie

W chwili obecnej sektor publiczny w naszym kraju zaczyna budować i wdrażać u siebie architekturę korporacyjną. Przyczyną podjęcia działań w tym kierunku jest:

Duży wzrost złożoności Zintegrowanych Środowisk na co składa się:

- wiele komponentów (systemów),
- złożone procesy.
- analiza komunikacji międzysystemowej stała się krytyczna.

Wielu końcowych użytkowników (nowe role w organizacjach publicznych):

- Analityk Biznesowy, Architekt Rozwiązania, Architekt Edycji, Główni Architekci,
- brak możliwości dzielenia się wiedzą.

Brak narzędzi do centralnego zarządzania architekturą aplikacji (również na rynku):

- wiele projektów i edycji systemów, wiele diagramów w projektach, brak zarządzania zmianą w prowadzeniu projektów.

Brak formalnej specyfikacji zapisu architektury:

- wiele diagramów ale brak modeli architektury.

Prezentowany artykuł przedstawia rezultaty niektórych prac badawczych podjętych przez autorów w kilku sektorach: publicznym i telekomunikacji w Polsce w zakresie architektury dotyczących:

- **celów architektonicznych**, w tym: określanie macierzy celów architektonicznych, określanie celów projektowych (ustalenie mierników celów),
- **metodyki** w tym: metodyki budowy architektury w zakresie danych.

2. Architektura korporacyjna w sektorze publicznym

Przedstawiona tutaj koncepcja modelowania wymagań i architektury jest w pełni zgodna z TOGAF³. Ze względu na obszerność tematu artykuł ogranicza się do zdefiniowania celów : biznesowych, architektonicznych i projektowych w sektorze telekomunikacyjnym. Cele projektowe ujęte są w postaci wymagań na metodykę projektowania architektury w tym sektorze. Artykuł powstał jako rezultat pracy badawczej dla sektora publicznego i

¹ Management Systems Consulting, 93-134 Łódź, Poznańska 28/1. jerzy.roszkowski@neostrada.pl

² HomeCenter.pl, 43-300 Bielsko-Biała, ul. Leszczyńska 29,. marcin.cwajna@homecenter.pl

³ TOGAF –ang. *The Open Group Architecture Framework* szkielet dla architektury korporacyjnej, który zapewnia kompleksowe podejście do projektowania, planowania, implementacji oraz zarządzania informacyjną architekturą przedsiębiorstwa. Architektura jest modelowana typowo na czterech poziomach (domenach): Procesy biznesowe, Zastosowanie, Dane, Technologia. Grupa [The Open Group](http://www.opengroup.org) udostępnia specyfikację TOGAF bezpłatnie dla organizacji do ich własnego, niekomercyjnego wykorzystania.

telekomunikacyjnego i jako rezultat realizowanych tamże przez autorów projektów i wdrożeń w środowiskach zintegrowanych.

3. Cele architektoniczne

W wyniku badań cele architektoniczne podzielono na grupy, w których wyodrębniono cele szczegółowe. Całość została przedstawiona na rysunku 1 przy pomocy modelu celów zgodnie z metodyką ARIS⁴. Poniżej w tabelach podano przyjęte definicje poszczególnych celów w grupach.

A. Elastyczne środowisko IT

Osiągnięcie wysokiej elastyczności jest celem złożonym wynikającym z możliwości konfiguracyjnych środowiska i "łatwości" developmentu. Elastyczność określa podatność środowiska na zmiany. W tej grupie wyróżniono i zdefiniowano następujące podcele:

Tabela 1- Grupa podcelów: elastyczne środowisko IT

Sym- bol	Cel architektoniczny	Opis	Jak mierzyć?	Wskaźnik* 100% dla linii lotniczych (przykład punkt 5.)
A1	Wysoka konfigurowalność	Konfigurowalność jest cechą środowiska/systemu określającą procent funkcjonalności, którą można zmieniać w trybie konfiguracji wykorzystując do tego określoną ilość miejsc konfiguracji. Celem jest osiągnięcie wysokiej konfigurowalności.	Wskaźnik konfigurowalności jest wektorem o postaci [STR,SKP]	
A1.1	Minimalna ilość miejsc konfiguracji dla danej klasy zagadnień	Osiągnięcie pożądaných zmian konfiguracyjnych wymaga modyfikacji szeregu systemów poprzez wprowadzenie modyfikacji w jednym lub więcej miejscach. Celem jest ograniczenie miejsc wymagających konfiguracji do niezbędnego minimum. W szczególności parametry z danej klasy zagadnień powinny być konfigurowalne tylko raz, w wyniku czego byłby konfigurowalny jeden lub więcej system. Przeniesienie konfiguracji z miejsca konfiguracji do danego systemu powinno być automatyczne. Przykład: jeśli konfigurujemy produkt dokonujemy to tylko jeden raz w centralnym katalogu produktów w wyniku czego następuje konfiguracja n-systemów (CRM, Billing, EAI, OM, inne..) biorących udział w zamówieniach i obsłudze danego produktu.	l -liczba klas zagadnień, klasy zadane listą; lista klas zagadnień STR = Stopień redundancji miejsc konfiguracji = \sum_j liczba miejsc konfigurowalnych globalnie/liczba miejsc konfigurowalnych. Celem jest MIN(STR)	9,52
A1.2	Wysoka liczba konfigurowalnych parametrów	Celem jest osiągnięcie jak największej liczby parametrów konfigurowalnych. W pierwszej kolejności należy dążyć do zapewnienia konfigurowalności dla parametrów najczęściej modyfikowanych i tych dla których koszt zmian developerskich jest wysoki.	SKP =Stopień konfigurowalności parametrów = liczba parametrów konfigurowalnych w wybranym obszarze/liczba wszystkich parametrów w tym samym obszarze	25%

⁴ ARIS - ang. *Architecture of Integrated Information Systems*, metodologia architektury i narzędzia rozwijane przez firmę IDS Scheer

Sym-bol	Cel architektoniczny	Opis	Jak mierzyć?	Wskaźnik* 100% dla linii lotniczych (przykład punkt 5.)
A2	Niska złożoność developmentu	Inaczej "łatwy" development. Celem jest osiągnięcia stanu w którym realizacja pewnego zbioru wymagań biznesowych będzie miała minimalną złożoność (wyrażoną np. w Function Points). Development - zmiany w systemach realizowanych w trybie prac rozwojowych (nie będących zmianami konfiguracyjnymi).	Iloraz liczby elementów architektury do liczby elementów architektury charakteryzujących się możliwością modyfikacji za pomocą prostego developmentu.	23,81%
A3	Automatyzacja testów	Celem jest ułatwienie testowalności rozwiązania, umożliwienie przeprowadzenia większej liczby testów i wyeliminowanie tego czynnika jako ograniczenia/wąskiego gardła w rozwoju całego środowiska.	Wskaźnik automatyzacji testów = liczba testów przeprowadzonych automatycznie/liczba wszystkich testów	4,76%
A4	Wystarczająca skalowalność procesów	Systemy realizujące procesy powinny mieć wydajność odpowiadające bieżącym wymaganiom i być skalowalne zgodnie określonymi założeniami.	Iloraz liczby skalowalnych elementów architektury i liczby elementów w architekturze	80,95%

B. Proste środowisko IT

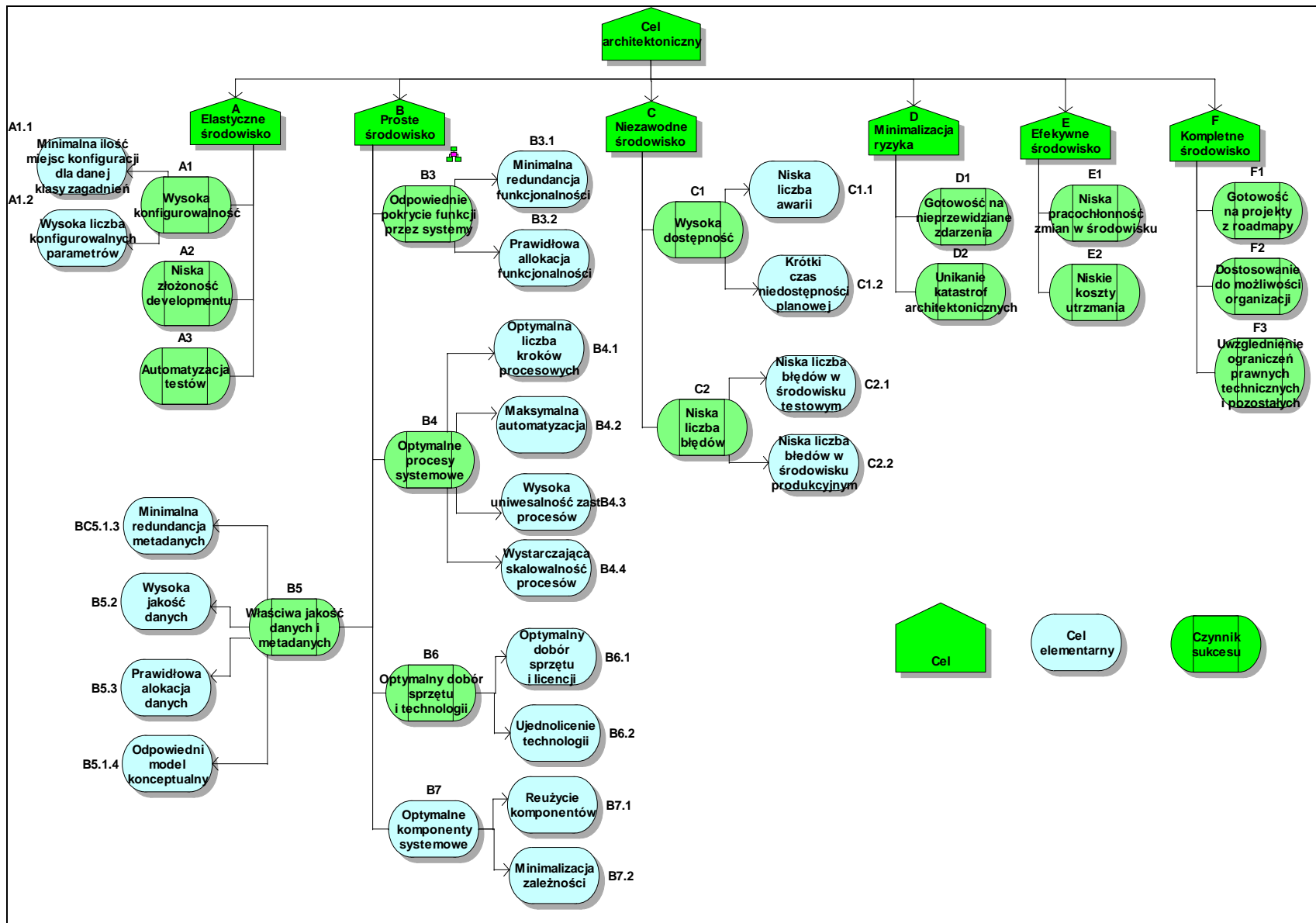
Cel złożony grupujący cele określające cechy bądź stan środowiska w zakresie funkcjonalności, procesów, danych i komponentów systemowych. W tej grupie wyróżniono i zdefiniowano następujące podcele:

Tabela 2- Grupa podcelów: proste środowisko IT

Sym-bol	Cel architektoniczny	Opis	Jak mierzyć?	Wskaźnik* 100% dla linii lotniczych (przykład punkt 5.)
B3	Odpowiednie pokrycie funkcji przez systemy	Celem jest odpowiednia, zgodna z najlepszą wiedzą i założeniami rozwoju środowiska alokacja funkcji do systemów.	Wektor postaci [WRF, WPA]	
B3.1	Minimalna redundancja funkcjonalności	Redundancja funkcjonalności jest liczba powtórzeń użycia danego komponentu lub funkcji w zdefiniowanym obszarze przy realizacji tej samej funkcjonalności. Określona jest tylko dla funkcji, które mogą być obsługiwane w sposób globalny.	WRF = Wskaźnik redundancji funkcjonalnej = łączna liczba funkcji/(Σ (liczby powtórzeń dla wszystkich funkcji*miara krotności funkcji))	47,17%
B3.2	Prawidłowa alokacja funkcjonalności	Do alokacji funkcjonalności do systemów przyjmuje się Application Map (AM) jako referencje. Zatem prawidłowa alokacja oznacza zgodność z AM. Strategia architektoniczna może stanowić inaczej i w takich przypadkach jest nadrzędna względem AM.	WPA = Wskaźnik poprawności alokacji = funkcjonalność poprawnie zaalokowana do systemów/funkcjonalność zaalokowana do systemów	52,00%

Sym- bol	Cel architektoniczny	Opis	Jak mierzyć?	Wskaźnik* 100% dla linii lotniczych (przykład punkt 5.)
B4	Optymalne procesy systemowe	Celem jest optymalizacja w zakresie projektowania procesów systemowych - maksymalna automatyzacja, uniwersalność i odpowiednie (pod kątem czasu trwania, oczekiwania i wykorzystania zasobów) dobranie liczby kroków procesów systemowych.		
B4.1	Optymalna liczba kroków procesowych	Optymalizacja procesu polegająca na takim doborze i podziale kroków procesowych zapewniająca minimalny czas trwania, minimalny czas oczekiwania, maksymalne wykorzystanie zasobów	Określona jest wektorem: [Min(T_w), Min(T_o), Max(WWZ)] , gdzie Min(T _w) - to minimalny łączny czas wykonania , Min(T _o) - to minimalny czas oczekiwania, Max(WWZ)- to wskaźnik wykorzystania przydzielonych zasobów	[1260, 0.5, 40%]
B4.2	Maksymalna automatyzacja	Celem jest realizowanie funkcjonalności automatycznych, czyli bez użycia kroków manualnych (o ile wymagania biznesowe nie stanowią inaczej).	WA= Współczynnik automatyzacji = liczba kroków wykonanych automatycznie / liczba wszystkich kroków.	47%
B4.3	Wysoka uniwersalność zastosowania procesów	Celem jest wdrażanie rozwiązań zapewniających ponowne używanie wdrożonych procesów. Inaczej pisząc procesy powinny być generyczne z możliwością reużycia.	LPG = liczba procesów generycznych, SR = stopień reużycia procesów generycznych SR= Σ (krotności reużycia)*miara krotności reużycia/LPG Potrzebna definicja procesu generycznego	35%
B5	Właściwa jakość metadanych i danych	Osiągana poprzez wdrożenie procesu zarządzania danymi na którego składają się: -Zarządzanie danymi referencyjnymi -Standardy danych -Zarządzanie metadanymi -Jakość i certyfikacja danych -Zapewnienie właściwych danych -Bezpieczeństwo danych		
B5.1	Minimalna redundancja danych i metadanych	Inaczej dublowanie się informacji	Wskaźnik redundancji danych = WRD= liczba danych zdublowanych/ liczba wszystkich danych; Wartość wskaźnika WRD	9,00%

			dla maksimum funkcji WP=WP(WRD), gdzie WP - wydajność przetwarzania	
B5.2	Kompletność danych	Definicja kompletności : 1. Opis każdej danej powinien dać się skompletować do postaci informacji tj jest może być przedstawiony w postaci czwórki piątki: (N, D, DD, JM, T), gdzie N- nazwa danej, D- definicja danej, DD - Dziedzina danej, JM - jednostka miary, T - czas. 2. Dane powinny zawierać wszystkie informacje niezbędne dla danego obszaru biznesowego 3. Dane spełniają wszystkie wymagane reguły biznesowe w przedmiotowym obszarze 4. Dane są wystarczająco dokładne i prawdziwe.	Wskaźnik niekompletności danych = liczba danych niekompletnych/liczba wszystkich danych	0,50%
B5.3	Prawidłowa alokacja danych	Prawidłowa alokacja oznacza zgodnie z modelem funkcjonalnym w sektorze (np. w Telekomach : modele eTOM i TAM) o ile strategię nie mówią inaczej. Celem jest wdrażanie rozwiązań zapewniających odpowiednie, zgodne z długoterminowymi założeniami dla środowiska zapewnienie masterowości danych oraz procesów dystrybucji tych danych.	PAD = Liczba grup danych prawidłowo zaalokowanych w Hurtowni Danych lub w systemach architektury/Liczba grup danych w elementach architektury	99%
B5.4	Odpowiedni model konceptualny danych (spójność, zakres biznesowy)	1. Celem jest zapewnienie takiego modelu który będzie zawierał dane spójne (spełnienie co 3 reguł normalizacji Codd'a) . 2. Zakres biznesowy i granulacja taka jaką wymagają funkcje realizowane przez system.	OMK =Liczba grup danych w architekturze spełniających 3 reg. Codd'a/liczba wszystkich grup danych	100%
B6	Optymalny dobór sprzętu i technologii	Celem jest wybór komponentów optymalnych kosztowo w zakresie wdrożenia i utrzymania. Optymalny dobór oznacza również niską awaryjność.		
B6.1	Optymalny dobór sprzętu i licencji	Celem jest wybór komponentów optymalnych kosztowo w zakresie wdrożenia i utrzymania. Optymalny dobór oznacza również niską awaryjność.	ODS = Przeciętna liczba nieuszkodzonych elementów arch. technicznej w ciągu 1 roku/liczba elementów architektury techn.	99%
B6.2	Ujednolicenie technologii	Celem jest ujednolicenie technologii w zakresie sprzętu i software.	UT = liczba spójnych technologicznie rozwiązań El. Architektury/ogólna liczba El. architektury	80%
B7	Minimalizacja zależności	Celem jest minimalizacja zależności. Minimalizacja zależności ma prowadzić do zmniejszenia ilości relacji w środowisku tylko do niezbędnych.		Niska



Rys 1. Model celów architektonicznych w środowisku zintegrowanym architektury korporacyjnej IT. Źródło: opracowanie własne.

C. Niezawodne środowisko IT

Celem jest osiągnięcie stanu, w którym środowisko wspiera procesy biznesowe zgodnie z założoną dostępnością i bezbłędnie. Jednocześnie powinien być zapewniony dostateczny poziom bezpieczeństwa ciągłości rozwoju i utrzymania. W tej grupie wyróżniono i zdefiniowano następujące podcele:

Tabela 3- Grupa podcelów: niezawodne środowisko IT

Symbol	Cel architektoniczny	Opis	Jak mierzyć?	Wskaźnik* 100% dla linii lotniczych (przykład punkt 5.)
C1	Wysoka dostępność	Dostępność oznacza zdolność do realizacji procesów biznesowych. Celem jest osiągnięcie wysokiej dostępności i co z tego wynika unikanie wszelkich zdarzeń mających na dostępność wpływ negatywny. Dotyczy to zarówno zdarzeń nieplanowanych (awarii) i planowanych (wyłączenie w trakcie wdrożenia).		
C1.1	Niska awaryjność	Celem minimalizacja liczby i skutków awarii.	liczba, skutki awarii Definicja awarii i skutek awarii	23 awarie w skali roku. Awaria = ITIL incydent
C1.2	Krótki czas niedostępności planowej	Celem jest skrócenie czasu wyłączenia środowiska/komponentu środowiska w trakcie wdrażania funkcjonalności na produkcję.	Wskaźnik dostępności = czas niedostępności w trakcie wdrożeń/całkowity czas pracy	1%
C2	Niska liczba błędów	Celem jest jak najniższa liczba poważnych błędów. Błędy nie są traktowane tak samo, większa waga przypisywana jest do błędów krytycznych, mniejsza do niekrytycznych. Nie każdy błąd musi powodować awarię czy zmniejszenie dostępności, ale każdy ma wpływ na koszty i czas TTM.	WB=WB_T+WB_P gdzie WB - współczynnik liczby błędów	
C2.1	Niska liczba błędów na środowisku testowym	Celem jest jak najniższa liczba błędów na środowisku testowym. Znaczenie ma waga błędu.	WB_T=Współczynnik ważony liczby błędów TESTOWANIA= $\sum w_i/LB$, gdzie LB – ogólna liczba błędów, $w_i = 1$ dla błędu gdzie krytycznego $w_i = 1/2$ dla błędu niekrytycznego, i – wskaźnik błędu	poniżej 2,00%
C2.2	Niska ilość błędów na środowisku produkcyjnym	Celem jest jak najniższa liczba błędów na środowisku produkcyjnym. Znaczenie ma waga błędu.	WB_P=Współczynnik ważony liczby błędów produkcyjnych= $\sum w_i/LB$, gdzie LB – ogólna liczba błędów, $w_i = 1$ dla błędu gdzie krytycznego $w_i = 1/2$ dla błędu niekrytycznego, i – wskaźnik błędu	poniżej 0,01%

D. Minimalizacja ryzyka

Celem jest minimalizacja ryzyka. W tej grupie wyróżniono i zdefiniowano następujące podcele:

Tabela 4 - Grupa podcelów: minimalizacja ryzyka

<i>Symbol</i>	<i>Cel architektoniczny</i>	<i>Opis</i>	<i>Jak mierzyć?</i>	<i>Wskaźnik* 100% dla linii lotniczych (przykład punkt 5.)</i>
D1	Gotowość na nieprzewidziane zdarzenia	Rozwój architektury powinien być prowadzony w taki sposób żeby oprócz sprawnej i efektywnej realizacji bezpośrednich zadań biznesowych i konkretnych projektów z roadmapy środowisko IT było przygotowane na zdarzenia nieprzewidziane. Przykład takiego zdarzenia: połączenie z nowym operatorem. Może to oznaczać odpowiednią dekompozycje funkcjonalności, zachowanie odpowiedniej elastyczności procesów, czy inne cechy środowiska nie ujęte pozostałymi celami architektonicznymi.	niemierzalne - konieczne określenie miary	Na podstawie doświadczeń przy incydentach (niezidentyfikowanych awariach) - maksymalny czas odtworzenia do działania produkcyjnego nie przekroczył 3 godz
D2	Unikanie "Katastrof architektonicznych"	Katastrofa architektoniczna to naturalny rozwój systemu w czasie którego popełniono błąd prowadzący do braku możliwości dalszego rozwoju systemu/architektury (np. przypadek starego OV). Wynikiem takiego zdarzenia może być powołanie projektu, którego jedynym bądź głównym celem będzie zapewnienie ciągłości biznesowej. Celem jest unikanie takich katastrof.	Prawdopodobieństw o katastrofy = Sumaryczna pracochłonność projektów, które nie dają wartości dodanej dla biznesu/Łączna pracochłonność wszystkich projektów	0,80%

E. Efektywne środowisko IT

Celem jest osiągnięcie wymagań biznesowych przy minimalizacji całkowitego kosztu związanego ze środowiskiem wspierającym/realizującym te wymagania. Na koszt ten składa się koszt wdrożenia i utrzymania.

W tej grupie wyróżniono i zdefiniowano następujące podcele:

Tabela 5 - Grupa podcelów: efektywne środowisko

<i>Symbol</i>	<i>Cel architektoniczny</i>	<i>Opis</i>	<i>Jak mierzyć?</i>	<i>Wskaźnik* 100% dla linii lotniczych (przykład punkt 5.)</i>
E1	Niska pracochłonność zmian deweloperskich w środowisku	Celem jest osiągnięcie stanu w którym koszt realizacji zmian deweloperskich (o określonej złożoności) jest minimalny.	WSP = Wartość prac deweloperskich (w zakresie niezbędnych zmian wynikających ze zmian w strategii lub uwarunkowań biznesowych/wartość wszystkich prac	47,92%

			developerskich	
E2	Niskie koszty utrzymania	Celem jest osiągnięcie niskiego kosztu utrzymania środowiska.	koszty utrzymania na jednostkę funkcjonalności	3840 PLN

F. Kompletnie środowisko IT

Kompletność środowiska określa stopień i gotowość do realizacji wymagań. Wymaganiami są bezpośrednie wymagania zgłaszane w projektach, roadmapa projektów, możliwości organizacji i ograniczenia. Celem jest stworzenie środowiska umożliwiającego kompletną realizację wymagań. Unikamy sytuacji w której ograniczenia środowiska uniemożliwiają realizację niektórych wymagań.

W tej grupie wyróżniono i zdefiniowano następujące podcele:

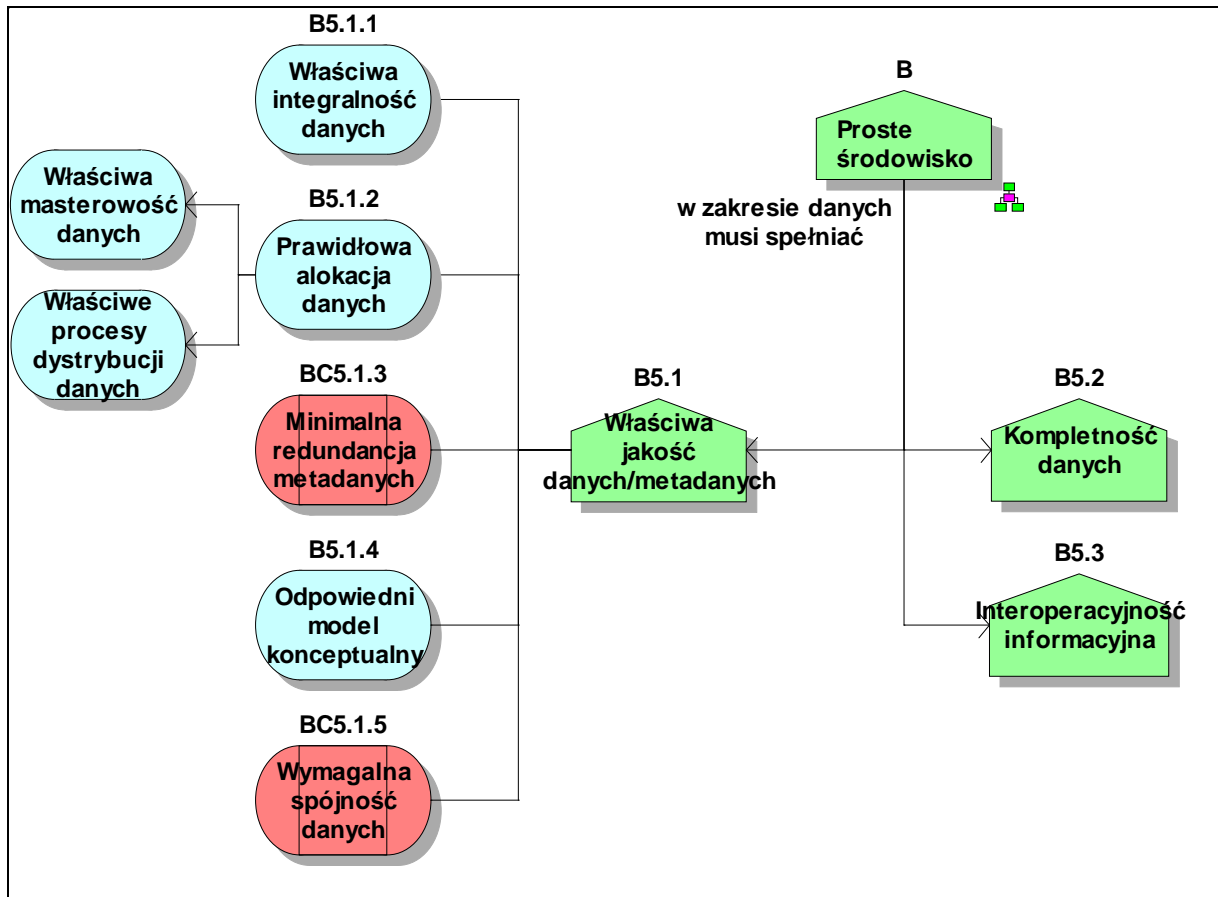
Tabela 6 - Grupa podcelów: kompletne środowisko IT

Sym bol	Cel architektoniczny	Opis	Jak mierzyć?	Wskaźnik* 100% dla linii lotniczych (przykład punkt 5.)
F1	Gotowość na projekty z roadmapy	Celem jest dążenie do takiej postaci środowiska, która umożliwi i ułatwi (w rozumieniu kosztów i czasu rozwoju), w chwili obecnej i w przyszłości, wdrażanie kolejnych zmian, przewidzianych w roadmapie projektów.	GPR= Liczba projektów w Roadmapie które mogą być wykonane bez modyfikacji architektury/ liczba wszystkich projektów z Roadmapy	80% gotowości
F2	Dostosowanie do możliwości organizacji	Możliwości organizacji, które nie mogą być w skończonym czasie zmodyfikowane powinny być brane pod uwagę jako sztywne ograniczenia dla projektów wpływających na architekturę. Spełnienie tego ograniczenia staje się więc jednym z celów architektonicznych, podobnym do spełnienia wymagań biznesowych.		
F3	Uwzględnienie ograniczeń prawnych, technicznych i pozostałych	Celem jest uwzględnienie ograniczeń mogących wpływać na środowisko. W szczególności należy wziąć pod uwagę ograniczenia prawne i techniczne.	OP= Liczba elementów architektury, które w przypadku zmiany otoczenia prawnego, technicznego nie spowoduje konieczności modyfikacji architektury/ogólna liczba elementów architektury	Na poziomie 80%

4. Cele architektoniczne w zakresie danych w sektorze publicznym

Budowa architektury danych jest ściśle związana z celami architektonicznymi w tym zakresie i wymaganiami jakie ma ona spełniać w środowisku. Poniżej na rysunku przedstawiono diagram celów architektonicznych w zakresie danych (wg standardu Aris). Ze względu na wieloaspektowość dekompozycji celów dekompozycja ta nie może być wszędzie

rozłączna. Cele szczegółowe które występują kilkakrotnie w drzewie nazwano „celami wspólnymi” (common goals). Cele architektoniczne w zakresie danych są jednym z aspektów wymagań na proste środowisko. Kompletność i jakość danych umożliwia posługiwanie się nimi w przetwarzaniu, natomiast zapewnienie interoperacyjności związane jest z budową złożonych komunikatów jednoznacznie zrozumiałych przez nadawcę (system wytwarzający) i odbiorcę (system odbierający) .



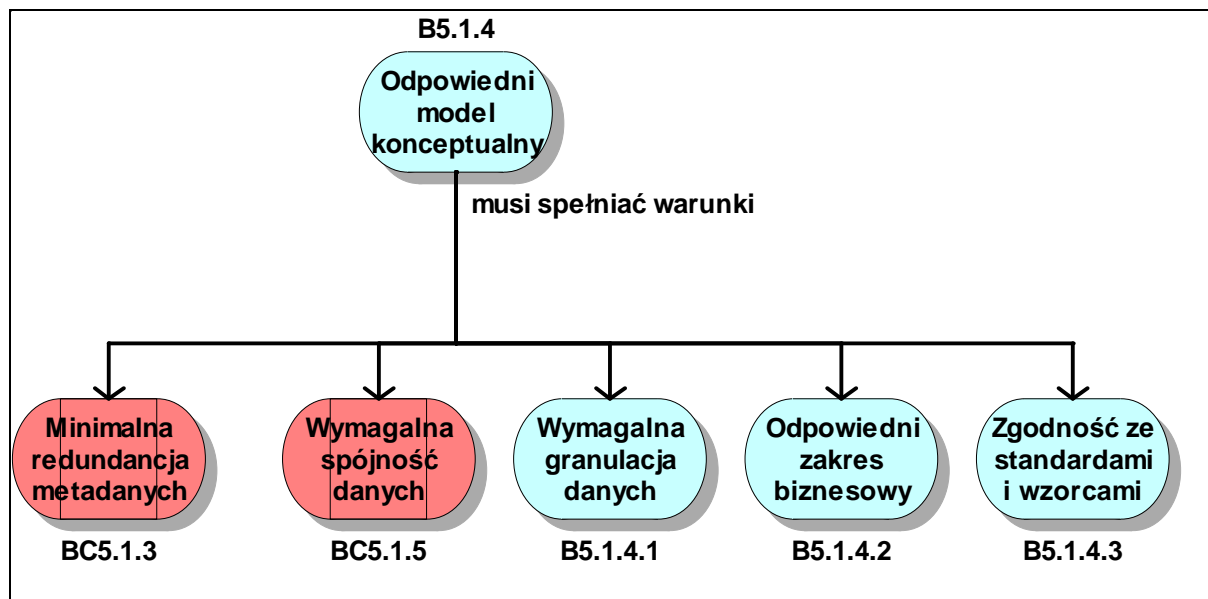
Rys.2. Cele architektoniczne w zakresie danych .Source: author's study

5. Wymagania w zakresie modelowania danych

Na poniższym rysunku przedstawiono wymagania na **model konceptualny danych**
 Wymagalna granulacja – (inaczej ziarnistość danych) oznacza wymaganą powtarzalną długość okresu czasowego w którym dane są gromadzone

Odpowiedni zakres biznesowy - oznacza zakres modelu danych powinien obejmować zakres biznesowy niezbędny dla realizacji celu biznesowego.

Zgodność ze standardami i wzorcami – oznacza zgodność z UML (diagram klas, oraz normalizację danych do 3 formy normalnej Codd'a).



Rys 3. Wymagania na modele danych. Źródło: opracowanie własne.

6. Przykład i ocena architektury IT dla jednostki sektora publicznego

Poniżej na przykładzie architektury IT linii lotniczych (jednoosobowa spółka skarbu państwa) pokazano jak wykaz celów i mierników podany w punkcie 3 może posłużyć do oceny tej architektury w tym do określenia jej mocnych i słabych punktów oraz wskazania kierunków dalszego rozwoju.

6.1. Ogólny opis architektury IT linii lotniczej

Głównym elementem architektury jest System Rezerwacyjny NAV, w którym zaszyte są główne części procesu sprzedażowego. Klienci rezerwują, odwołują rejsy. Wokół systemu NAV została zbudowana hurtownia danych, która przetwarza dane z systemu NAV i tymi danymi zasila zarówno inne systemy, jak i bezpośrednio, przy użyciu komponentów Business Objects udostępnia przetworzone dane użytkownikom (szczególnie dotyczy to finansów).

Dane do hurtowni danych są generowane codziennie przez System rezerwacyjny na konto FTP z którego moduł **BulkLoadMapper** ładuje dane do Bazy danych. Moduł ten służy także do kontroli poprawności danych jak i do kontroli integracji danych (sprawdzanie powtórzeń danych i kontynuacji danych). System navitaire, jest zasilany danymi z potwierdzeń płatności oraz z modułu służącego do administracyjnego anulowania rejsów.

Wokół hurtowni danych osadzone są systemy (komponenty), które na podstawie danych z systemu rezerwacyjnego umożliwiają:

- Rozliczenia Agentów, którzy sprzedają rejsy firmy
- Moduł Call Center, który zasilany jest danymi do IVR
- Moduł, który umożliwi rozliczenia personelu pokładowego
- Raporty dla Finansów i Księgowości (także VAT)
- Raporty sprzedażowe

- Revenue Management
- Moduł raportowy (także do wysyłania SMSów)

Aby w ramach powyższych modułów zapewnić poprawność danych i łatwą ich interpretację, dodatkowo Hurtownia danych jest ładowana „Kursami Walut”, Informacjami z wpływów na rachunki Bankowe (co z resztą z jednej strony jest podstawą do obliczania Cash Flow a z drugiej strony po przetworzeniu zasila system rezerwacyjny „w drugą stronę” zmieniając statusy rezerwacji), informacjami z usług płatności (np. w hipermarketach) jak BillBird czy Bibit (płatności kartami).

Dodatkowo, dane z systemu rezerwacyjnego po przetworzeniu zasilają firmę ubezpieczeniową, a moduł konfiguracyjny umożliwia stosowną parametryzację rozliczeń pomiędzy linią lotniczą a dostawcą produktu ubezpieczeniowego.

Hurtownia danych zasilana jest także danymi z modułu Netline Crew (w którym są zawarte informacje o personelu pokładowym i godzinach wylatanych w ramach danych rejsów).

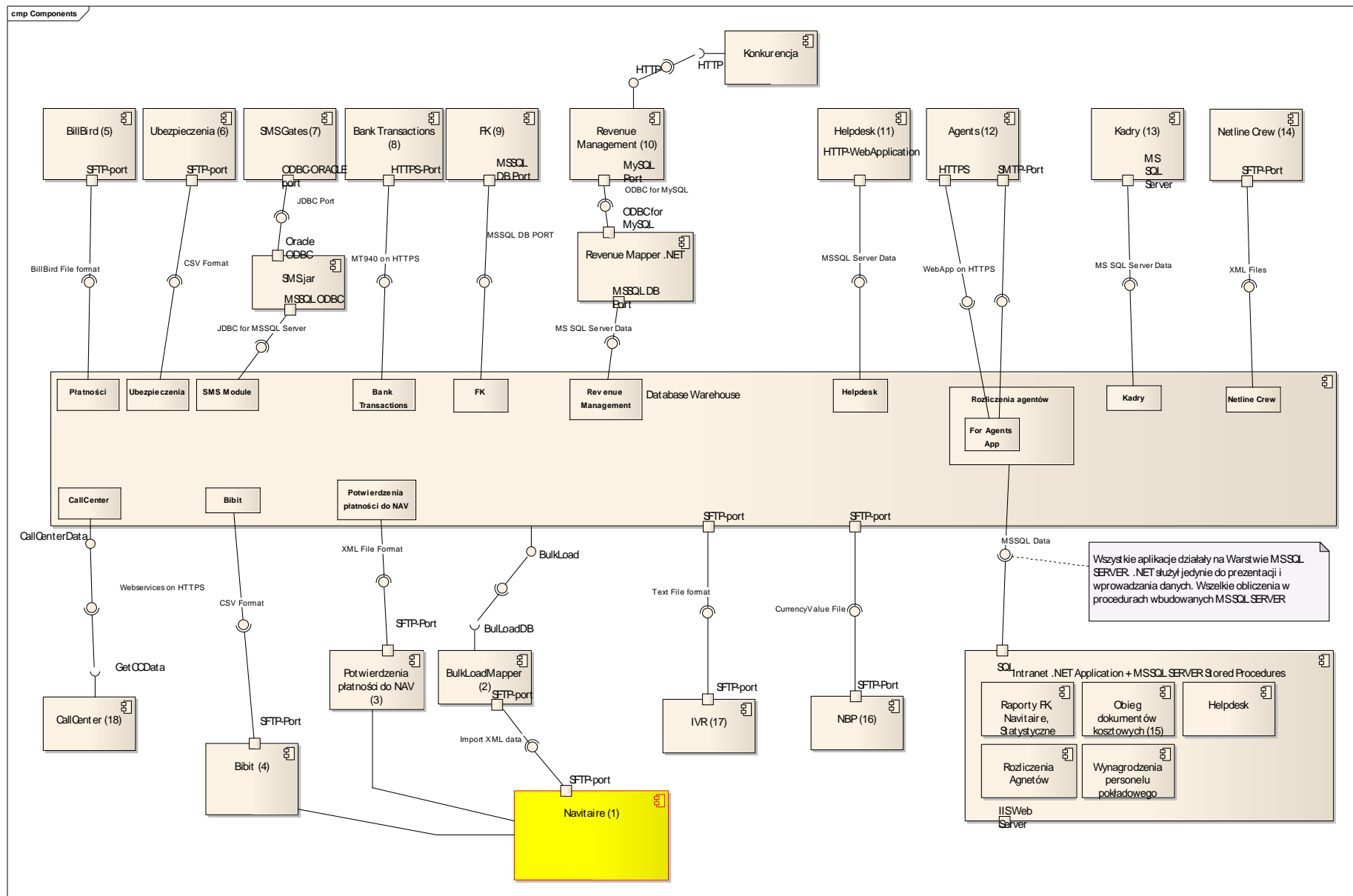
6.2. Zestawienie komponentów architektury IT linii lotniczej

Poniżej w tabeli 7 przedstawiono wykaz komponentów architektury IT linii lotniczej uzupełniający opis przedstawiony w punkcie 6.1.

Tabela 7 – Wykaz komponentów architektury przedstawionej na rysunku 4

Lp	Component	Opis elementu	Relacja
1.	System Rezerwacyjny – Navitaire(1)	Główny system, które dane są podstawą systemów informatycznych w firmie	Komponent zasila hurtownię danych oraz jest zasilany potwierdzeniami płatności (bibit, billbird), oraz anulacjami rejsów
2	BulkLoadMapper (2)	Komponent, który zasila hurtownię danych.	Zasila hurtownię danych danymi w formacie XML generowanymi automatycznie z systemu rezerwacyjnego
3.	Potwierdzenia płatności do NAV (3)	Moduły, pobierające dane z usług płatności.	Zasila NAV zmieniając statusy rezerwacji na „zapłacone”
4	Bibit (4)	Komponent, który zasila NAV potwierdzeniami obciążania Kart Kredytowych	Interfejs: Weryfikacja w NAV płatności i klientów, ładowanie hurtowni danych transakcjami
5	BillBird (5)	Pobiera dane z systemu BillBird i zasila transakcjami hurtownię. Następnie moduł potwierżeń płatności zasila NAV.	Billbird zasila hurtownię transakcjami
6	Ubezpieczenia (6)	Zasilany przetworzonymi transakcjami z systemu rezerwacyjnego moduł, pozwalający na obliczenie należnych ubezpieczalni kwot.	Zasila firmę ubezpieczeniową danymi do rozliczeń
7	SMS Gates (7)	Moduł wysyłający SMSy	Zasilany jest przez moduł SMS.jar, który pobiera dane z modułu raportowego (różne raporty)
8	BanTransactions (8)	Moduł, zasilający hurtownię transakcjami płatności za rezerwacje w formie przelewu.	Zasila hurtownię danych z formatu MT940 (konwerter)
9	FK (9)	System Finansowo-Księgowy	Zasilany jest z hurtowni danymi finansowymi i rejestrami VAT
10	Revenue Management (10)	System pozwalający na zarządzanie wpływami, pozwalający na modyfikację taryf po wnioskowaniu na danych konkurencji	Zasila hurtownię danych
11	Helpdesk (11)	Moduł do obsługi zgłoszeń, powiązany jest z modułem raportowym i modułem nadawania uprawnień jak i workflow.	Niezależny pod względem danych, obudowany modułem raportowym i modułem nadawania uprawnień jak i workflow.
12	Agents (12)	Moduł do obliczania zarobków Agentów. Wysyłający cyklicznie raporty generowane	Zasilany z systemu NAV i bazy parametryzującej wewnętrznej.

		na podstawie umów z Agentami (3000 agentów w Polsce i zagranicą)	
13	Kadry (13)	System Kadry – płace – umożliwiający obsługę (pisma, urlopy delegacje itp.) i drukowanie rozliczeń, PIT itp.	Zasilany przez moduł raportowy obliczający wynagrodzenia personelu na przykładzie lotów (Netline Crew) oraz Systemu rezerwacyjnego.
14	Ntline Crew(14)	System operacyjny dla załóg lotów	Zasila hurtownię danych
15	Obieg dok. kosztowych (15)	System umożliwiający kwalifikację kosztów w obszary działania spółki oraz księgujący na odpowiednie konta koszty	Zasilany przez użytkowników (koszty, plany kont obszary), wykorzystujący moduł raportowy i nadawania uprawnień, ładujący system FK.
16	NBP (16)	Kursy Walut (dzienne)	Moduł zasila hurtownię danych (wszystkie dostępne kursy walut)
17	IVR (17)	Moduł zasilający usługę IVR w Call Center	Pobiera dane z systemu Rezerwacyjnego za pośrednictwem modułu raportowego hurtowni danych.
18	Call Center	Moduł zasilający Call Center w dane pominięte w systemie rezerwacyjnym (np. reklamacje)	Moduł ładujący dane z systemu NAV do Call Center.



Rys 4. Przykładowa architektura środowiska zintegrowanego IT linii lotniczych (jednoosobowa spółka skarbu państwa). Źródło: opracowanie własne.

6.2. Cele biznesowe

Poniższa tabela przedstawia najważniejsze cele biznesowe stawiane przed środowiskiem IT

Tabela 8 – Strategiczne cele biznesowe środowiska IT linii lotniczej

ID	Cel biznesowy
1	Osiągnięcie odpowiedniej pozycji (dla klientów z Polski) rynku lotniczego typu LOW COST (konkurencja to WIZZAir, RyanAir, Norwegian, Na niektórych kierunkach – Lufthansa).
2	Osiągnięcie cen, które spowodują wzrost sprzedaży.
3	Zrównoważenie realizacji oraz sprzedaży pomiędzy sezonami LATO-ZIMA.
4	Rezygnacja z głównych lotnisk (migracja na lotniska małe, podmiejskie).

Ewentualne zmiany (dodanie nowych, rezygnacja z obecnych) celów biznesowych będą miały wpływ na cele architektoniczne a to może powodować zmianę architektury docelowej lub położyć nacisk na inne aspekty w ramach obecnej architektury.

6.3. Mapowanie niektórych celów biznesowych na niektóre cele architektoniczne

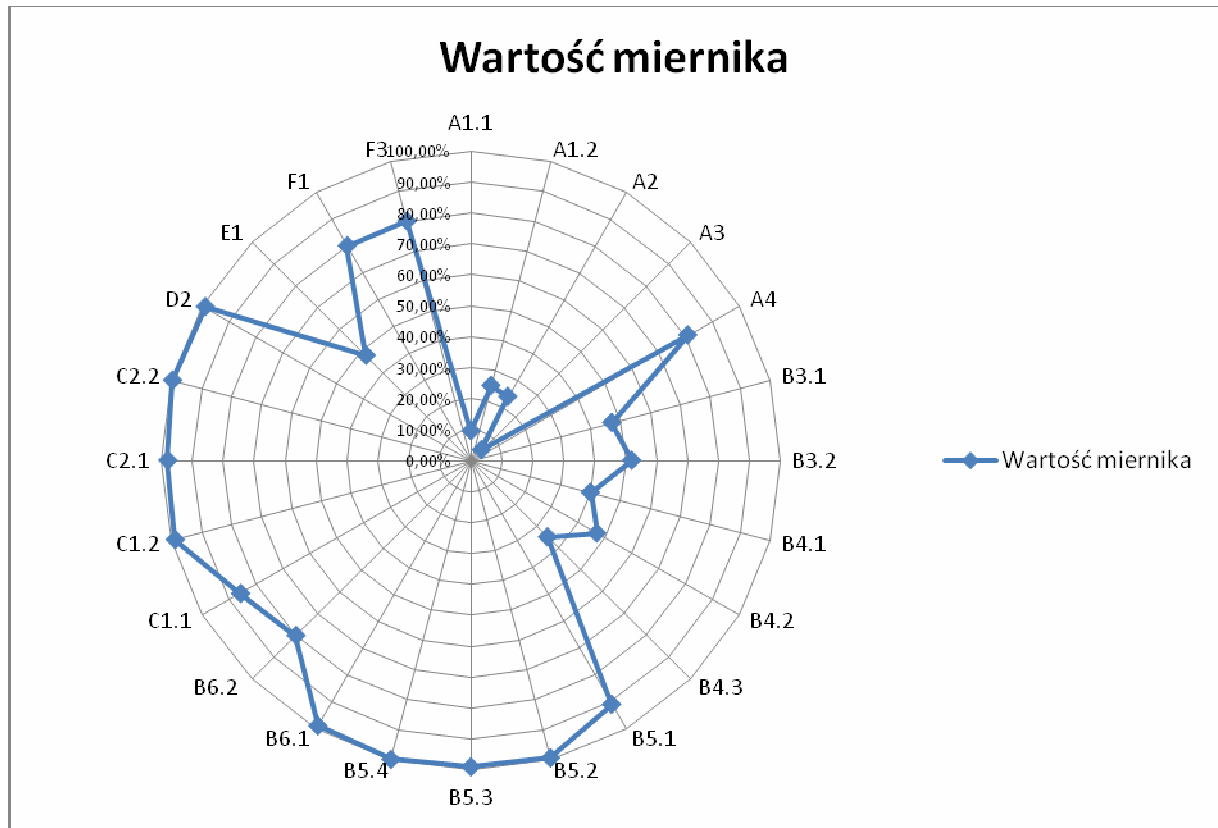
Tabela przedstawia stopień wpływu realizacji celu architektonicznego na osiągnięcie danego celu biznesowego (X – wpływ duży, x – wpływ mały). Przykład interpretacji: Osiągnięcie celów architektonicznych [A], [C], [E] będzie miało pozytywny wpływ na realizację celu biznesowego [1].

Tabela 9 – Mapowanie celów biznesowych na cele architektoniczne dla linii lotniczych.

Cele architektoniczne \ Cele biznesowe		Osiągnięcie odpowiedniej pozycji (dla klientów z Polski) rynku lotniczego typu LOW COST	Osiągnięcie cen, które spowodują wzrost sprzedaży	Zrównoważenie realizacji oraz sprzedaży pomiędzy sezonami LATO-ZIMA	Rezygnacja z głównych lotnisk (migracja na lotniska małe, podmiejskie)
		[1]	[2]	[3]	[4]
Wysoka konfigurowalność	[A]	X	X		X
Proste środowisko IT	[B]		x	X	
Niezawodne środowisko IT	[C]	x			
Minimalizacja ryzyka	[D]		X		
Efektywne środowisko IT	[E]	X		X	
Kompletne środowisko IT	[F]			x	x

6.4. Ocena architektury środowiska zintegrowanego IT linii lotniczych

Poniżej pokazano wykres radarowy wyliczonych mierników (podanych w ostatniej kolumnie tabel od 1- do 7 dla architektury IT linii lotniczych.



Rys.5. Wykres radarowy wartości mierników architektury dla architektury IT linii lotniczych. W rysunku powyższym dla wskaźników (nx) B5.1 B5.2, C1.1, C1.2, C2.1, C2.2, D2 oraz E1, **zostały odwrócone wskaźniki**, do postaci 100%-nx% aby uzyskać stan wskaźnika 100% obrazującego najlepsze możliwe rozwiązanie architektoniczne.

W zakresie mierników typu A oraz E (niezbędna jest wspólna interpretacja) (tj. *konfigurowalności architektury*) wyraźnie widać, iż jest to słaby jej punkt. Miernik pokazuje, że w przypadku pojawienia się nowych potrzeb biznesowych, zmiany będą czasochłonne i wymagana będzie ingerencja zarówno programistyczna o dużym stopniu złożoności. Dalej należy wyciągnąć wniosek, iż pociągnie to za sobą wielowarstwowe zmiany, które wymagać będą testów integracyjnych. Jest to słaby element architektury. Na wykresie widzimy, że architektura daje na ponad 80% możliwość wyskalowania procesów, jednak dużym kosztem.

Mierniki typu B (odpowiednie pokrycie funkcji przez systemy oraz redundancja danych) wskazują na wyższą niż przeciętną koncepcję rozwiązania (oscylacja w okolicy 70%). Optymalizacja powinna polegać na przeanalizowaniu procesów biznesowych i ich optymalizacji tym nie mniej spełnia ona założenia. Z tego miernika może wynikać, iż architektura podczas procesu projektowania nie uwzględniała wszystkich celów biznesowych i część z niej powstało ad-hoc. Dodatkowo, należy także wyciągnąć wniosek, iż hurtownia danych i jej redundancja została zaprojektowana wzorcowo.

Mierniki typu C oraz D. Tu zdecydowanie widać mocno przemyślaną koncepcję, która duży nacisk (98%) kieruje na poziom dostępności, niezawodności środowiska oraz jego wydajność. W tym zakresie architektura jest wzorcowa.

Miernik F pokazuje, że architektura ponadprzeciętnie jest przygotowana na nowe projekty i że z punktu widzenia danych i ich redundancji, łatwe będzie zarówno pokrycie funkcjonalne jak i ew. rozwój.

Ogólnie, można postawić tezę, że architektura jest przygotowana ponadprzeciętnie 67,66% (zakładając przeciętność architektury na poziomie osiągnięcia średniego wskaźnika = 30% - dla większości niedojrzałych architektur korporacyjnych) i została przygotowana przez zespół projektantów stawiających nacisk bardziej na SLA, czy też dostępność systemu oraz przez specjalistów od integracji danych i hurtowni danych.

7. Podsumowanie

Definiowanie celów architektonicznych i biznesowych jest podstawowym i pierwszym krokiem w budowaniu architektury korporacyjnej. W TOGAF ADM jest ustalane to w kroku "architecture vision". W pracy przedstawiono wyniki badań w tym zakresie przeprowadzonych przez autorów w kilku sektorach: telekomunikacyjnym, publicznym. Cele architektoniczne mogą posłużyć do oceny aktualnego stanu architektury IT i wytyczenia jej kierunków rozwoju. W artykule pokazano przykładową skróconą ocenę (ze względu na dopuszczalną objętość pracy) architektury IT linii lotniczych. Kolejnym krokiem po zdefiniowaniu celów architektonicznych jest budowa repozytorium wytycznych i zasad, określonych dla powyższych. Wytyczne i zasady służą kontroli realizacji celów podczas iteracyjnego rozwoju architektury IT. Dalszych pogłębionych analiz i badań wymaga przełożenie celów biznesowych (w sektorze) na cele architektoniczne.

Tematy i prace poruszone w artykule cieszą się dużym zainteresowaniem i w związku ze zgłoszonym zapotrzebowaniem będą dalej rozwijane przez autorów w sektorze przemysłu.

Literatura

- [1]. Jacob M.,Jonkers H., (2004) *Quantitative Analysis of Enterprise Architectures*, ArchiMate Deliverable D3.5.1 b, Telematica Institut,
- [2]. Lankhorst M., (2005) *Enterprise Architecture at Work , Modeling, Communication and Analysis*, Springer –Verlag
- [3]. Richardson G., Jackson B., Dickson G.,(1990) *A Principles-Based Enterprise Architecture*, MIS Quartely, vol 14, no.4.
- [4] Roszkowski J., (2004), „*Analiza i projektowanie strukturalne (eng. analysis and design)*”, Helion, ISBN: 83-7361-397-8
- [5]. Roszkowski J. (2007) *The Formal Approach to Definition of the Requirements for the Needs of Computer Application Selection and Implementation*, Proc. of the 16th International Conference on Systems Science ICSS'07
- [7]. Roszkowski J.,Kobyliński A., (October 1-2 2009), “*Quality Management Reference Models for Business Intelligence – Class Systems*”, The 8th International Conference on Perspectives in Business Informatics Research, Kristianstad University College, Sweden
- [8]. Roszkowski. J., (2010), “*The simulation of processes in the integrated computer environment in the TELCO sector*” , Proc. of the 7 Symposium „ Modeling and Computer Simulation”, Łódź, College of Computer Science

- [9]. Scheer A. W. (2000), *Business Process Modeling*, Springer-Verlag, Berlin, Heidelberg, New York, ISBN: 3-540-51480-5
- [10]. Scheer A.W., (1994), *Business Process Engineering*, Springer-Verlag, Berlin, Heidelberg, New York,
- [11]. TOGAF™ Version 8.1.1 Enterprise Edition, (2007), ISBN: 9789087531737 Van Haren Publishing
- [12]. Wegemann A., (2003) *On the systemic Enterprise Architecture Methodology*, Proc. of the International Conference on Enterprise Information Systems, Angers, France