

Podręcznik wyznaczania rozmiaru
funkcjonalnego oprogramowania
w Agencji Restrukturyzacji i
Modernizacji Rolnictwa

Spis treści

Wstęp	3
Zakres stosowania wymiarowania funkcjonalnego W ARiMR	5
Wstęp do metody punktów funkcyjnych IFPUG	5
Definicje	5
Wprowadzenie do punktów funkcyjnych	6
Założenia PF IFPUG	6

Metoda PF IFPUG – zasady	8
Identyfikacja Plików danych.....	9
Identyfikacja Elementarnych Procesów	10
Klasyfikacja Elementarnych Procesów	14
Wyliczanie rozmiaru funkcjonalnego elementów modelu IFPUG	17
Wyliczanie rozmiaru funkcjonalnego zmian.....	17
Wyliczanie rozmiaru zmian Plików Danych	18
Wyliczanie rozmiaru zmian Elementarnych procesów	19
Zasady stosowania metody PF IFPUG w ARiMR.....	19
Stosowanie PF IFPUG na potrzeby funkcjonalności biznesowej	19
Stosowanie PF IFPUG dla złożonych funkcjonalności	27
Wyznaczanie rozmiaru zmian funkcjonalności	30
Stosowanie PF IFPUG dla interfejsów i usług.....	32
Stosowanie PF IFPUG dla procesów wsadowych	34
Stosowanie PF IFPUG dla funkcjonalności przygotowania danych	34
Stosowanie PF IFPUG dla funkcjonalności bazowych	38
Zasady dokumentowania rozmiaru funkcjonalnego oprogramowania / zmian oprogramowania	41
Zasady tworzenia i utrzymywania modelu artefaktów PF IFPUG	41
Zasady prezentacji wyników Wymiarowania funkcjonalnego w dokumentacji ofertowej	45

WSTĘP

Niniejszy dokument (**Podręcznik**) ma na celu opisanie zakresu oraz zasad wyznaczania rozmiaru funkcjonalnego oprogramowania w Agencji Restrukturyzacji i Modernizacji Rolnictwa, a także zasad dokumentowania wyników wymiarowania funkcjonalnego. Rozmiar funkcjonalny oprogramowania w Agencji Restrukturyzacji i Modernizacji Rolnictwa bazuje na metodzie Punktów Funkcyjnych IFPUG (PF IFPUG). Metoda ta jest utrzymywana i rozwijana przez organizację *International Function Point Users Group* (www.ifpug.org).

Metodę PF IFPUG stosuje się w ARiMR zgodnie z zasadami opisanymi w dokumencie *Function Point Counting Practices Manual* w wersji 4.3.1 publikowanym przez organizację IFPUG między innymi w języku angielskim, a także publikacji organizacji NESMA (*Netherlands Software Metrics Association*) rozszerzającej zasady opisane w w/w dokumencie : *Function Point Analysis for Software Enhancement* (www.nesma.org, dokument dostępny w języku angielskim). Zasady opisane w publikacji organizacji NESMA mają zastosowanie dla szacowania zmian oprogramowania. Dokumenty te będą dalej nazywane **Dokumentami bazowymi**.

W rozdziale *Wstęp do metody punktów funkcyjnych IFPUG* Podręcznika zawarto tłumaczenia wybranych definicji oraz reguł pochodzących z wyżej wymienionych Dokumentów bazowych. Celem tego rozdziału jest wprowadzenie czytelnika w zagadnienie metody punktów funkcyjnych oraz zapewnienie kontekstu dla zasad opisanych w dalszych częściach Podręcznika. W przypadku stwierdzenia rozbieżności pomiędzy interpretacją pewnych definicji lub reguł zawartych w Dokumentach bazowych, a zapisami zawartymi we wskazanym wyżej rozdziale Podręcznika, należy przyjąć interpretację wynikającą z treści Dokumentów bazowych.

W rozdziale *Zasady stosowania metody PF IFPUG w ARiMR* Podręcznika zawarte są rozszerzenia i uszczegółowienia metody punktów funkcyjnych IFPUG, w stosunku do zasad opisanych w Dokumentach bazowych. Treści w tym rozdziale opierają się na zasadach opisanych w Dokumentach bazowych, w pewnych miejscach je rozszerzając, uszczegóławiając lub zmieniając w taki sposób, aby można je było stosować w sposób bardziej jednoznaczny przy wymiarowaniu funkcjonalności Aplikacji. Domyślnie wszystkie treści opisane w Dokumentach bazowych są stosowane w ARiMR, chyba, że nadpisują je treści zapisane w rozdziale *Zasady stosowania metody PF IFPUG w ARiMR* Podręcznika.

W rozdziale *Zasady dokumentowania rozmiaru funkcjonalnego oprogramowania / zmian oprogramowania* Podręcznika opisano zasady dokumentowania procesu wymiarowania funkcjonalności przeprowadzanego zgodnie z Podręcznikiem.

W przypadku stwierdzenia rozbieżności pomiędzy treścią Dokumentów bazowych, a zapisami zawartymi w rozdziałach *Zasady stosowania metody PF IFPUG w ARiMR* oraz *Zasady dokumentowania rozmiaru funkcjonalnego oprogramowania / zmian oprogramowania* Podręcznika, przyjmuje się interpretację zapisów wynikającą z Podręcznika.

Stosowanie zasad wymiarowania wynikających z Podręcznika wymaga znajomości Dokumentów bazowych.

Załączniki do niniejszego dokumentu:

- Załącznik nr 1A ARiMR_model_IFPUG_przyklad.eap
- Załącznik nr 1B ARiMR_model_IFPUG_zmiana_przyklad.eap
- Załącznik nr 2 ARiMR_raport_z_wymiarowania_IFPUG.xlsx

Źródła wykorzystane w opracowaniu:

- www.ifpug.org - Podręcznik Metody w wersji 4.3.1 i inne materiały dostępne dla członków organizacji,
- www.nesma.org – materiały publiczne,
- M. A. Parthasarathy, Practical Software Estimation, Addison-Wesley & Infosys 2007, ISBN 0-321-43910-4
- M. Mundsuh, C. Dekkers, The IT Measurement Compendium, Springer 2008, e-ISBN 978-3-540-68188-5
- Capers Jones, Estimating Software Costs, Bringing Realism To Estimating, Second Edition, The McGraw Hill Companies 2007, ISBN 13:978-0-07-148300-1, ISBN 10: 0-07-148300-4
- S. McConnell, Software Estimation: Demystifying the Black Art, Microsoftpress 2006, ISBN-13 978-0735605350

ZAKRES STOSOWANIA WYMIAROWANIA FUNKCJONALNEGO W ARIMR

Głównym celem wymiarowania funkcjonalnego w ARiMR jest wycena Modyfikacji.

Zakres Modyfikacji zdefiniowany jest za pomocą wymagań. Część tych wymagań odnosi się do pewnej funkcjonalności Aplikacji; zarówno istniejącej jak i potencjalnej (planowanej lub rozważanej do wytworzenia i wdrożenia). Znajomość rozmiaru tej funkcjonalności jest warunkiem wyceny Modyfikacji w oparciu o w/w wymagania.

Metodę PF IFPUG stosuje się do wyznaczania rozmiaru funkcjonalności, na podstawie jej opisu lub opisu zmian funkcjonalnych. Sam opis funkcjonalności, na podstawie którego wyznaczany jest rozmiar funkcjonalny, lub kontekst, w którym ten opis występuje, musi umożliwić identyfikację oraz zwymiarowanie wszystkich zidentyfikowanych elementów opisujących funkcjonalność zgodnie z przyjętą metodą.

WSTĘP DO METODY PUNKTÓW FUNKCYJNYCH IFPUG

DEFINICJE

Użytkownik funkcjonalny – osoba, system, urządzenie lub wyższa warstwa oprogramowania, która/y/e komunikuje się lub prowadzi interakcję z wymiarowanym oprogramowaniem.

Wymiarowanie – proces obejmujący wszystkie niezbędne czynności, do poprawnego określenia rozmiaru funkcjonalnego na podstawie opisu funkcjonalności oraz udokumentowania uzyskanych wyników wymiarowania zgodnie z niniejszym podręcznikiem

Stan stabilny – punkt, w procesie przetwarzania danych przez oprogramowanie, w którym cel Użytkownika funkcjonalnego został w pełni osiągnięty, nie pozostaje już nic do zrobienia, żeby go osiągnąć.

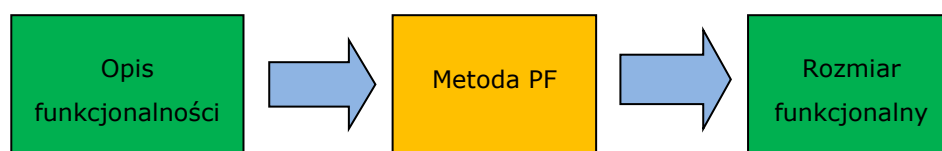
Przykład: Jeżeli Wymaganiem funkcjonalnym użytkownika jest wprowadzenie wniosku, to **stan stabilny** biznesu aplikacji zostanie osiągnięty, jeżeli wniosek zostanie w całości wprowadzony. Jeżeli system dostarcza użytkownikowi udogodnień, które pozwalają mu na przerwanie wprowadzania

wniosku z zachowaniem wprowadzonych danych, a następnie wznowienie wprowadzania, to z punktu widzenia wymagania użytkownika, w momencie przerywania wprowadzania wniosku biznesu aplikacji nie będzie pozostawiony w stanie stabilnym, ponieważ cel wymagania nie został osiągnięty. Podobnie wprowadzenie części wniosku, np. pojedynczej sekcji, nie jest wystarczające do realizacji celu użytkownika, po wprowadzeniu części wniosku, biznes aplikacji nie pozostaje w stanie stabilnym. Potrzebne są dodatkowe działania, które pozwolą na osiągnięcie celu użytkownika.

WPROWADZENIE DO PUNKTÓW FUNKCYJNYCH

Metody punktów funkcyjnych to zbiory reguł, których zastosowanie, pozwala wyznaczyć rozmiar funkcjonalny oprogramowania realizującego pewną funkcjonalność na podstawie opisu tej funkcjonalności. Oznacza to, iż na wejściu danej metody dostępny jest opis danej funkcjonalności, np. wynikający z określonego zbioru wymagań, a na wyjściu oczekiwana jest liczba punktów funkcyjnych wyznaczona na podstawie tego opisu.

Punkt funkcyjny jest abstrakcyjną miarą, definiowaną jako jednostka funkcjonalności dostarczanej przez mierzone oprogramowanie. Oznacza to, iż oprogramowanie o rozmiarze 100 punktów funkcyjnych dostarcza użytkownikowi 100 jednostek pewnej funkcjonalności.



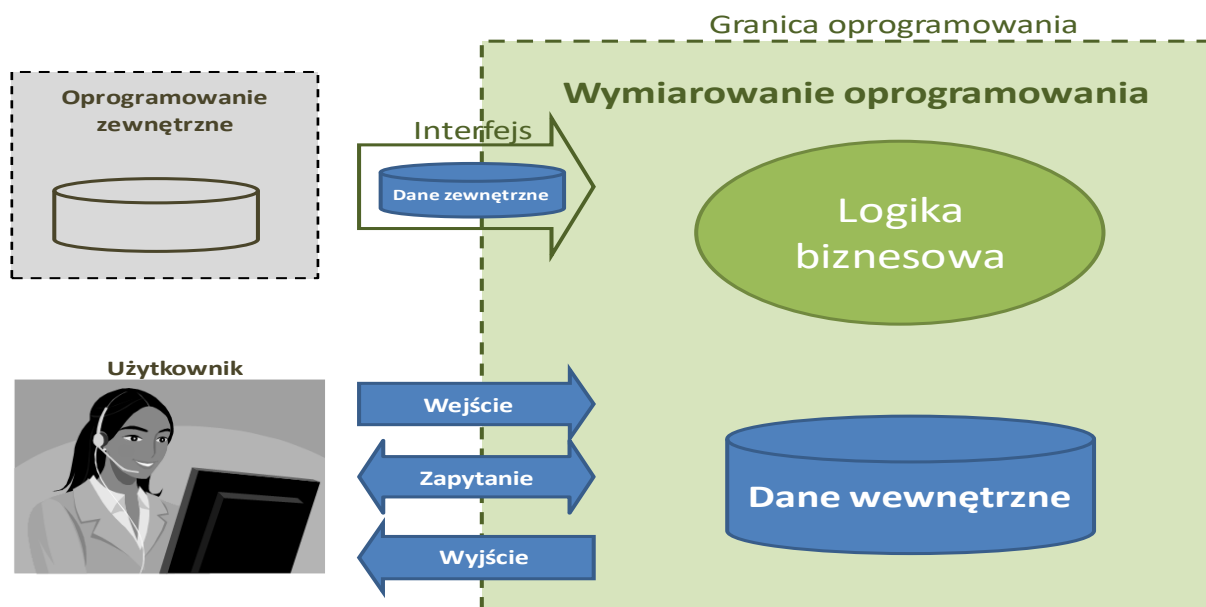
W przypadku zmian istniejącego oprogramowania, mówi się zarówno o rozmiarze funkcjonalnym samej zmiany, jak i rozmiarze funkcjonalnym oprogramowania po zmianie. W przypadku rozmiaru zmiany, liczba punktów funkcyjnych zmiany wynika z zmianą opisu zmian funkcjonalnych (zmian funkcjonalności). W przypadku rozmiaru oprogramowania po zmianie, liczba punktów funkcyjnych oprogramowania wynika z opisu funkcjonalności po zmianie i bez względu na rozmiar samej zmiany, może pozostać taka sama jak przed zmianą, może się zmniejszyć lub zwiększyć.

Przykład: W przypadku oprogramowania A o rozmiarze 100 PF zdefiniowano zmianę wprowadzającą nowe funkcje o rozmiarze 25 PF. Rozmiar zmiany to 25 PF a rozmiar oprogramowania A po zmianie to 125 PF. W przypadku oprogramowania B o rozmiarze 200 PF zdefiniowano zmianę modyfikującą istniejące funkcje - bez wprowadzania nowych. Rozmiar zmiany określono na 40 PF. Rozmiar oprogramowania B po zmianie to w dalszym ciągu 200 PF.

ZAŁOŻENIA PF IFPUG

Metoda Punktów Funkcyjnych IFPUG zakłada, iż rozmiar oprogramowania (wyrażony liczbą punktów funkcyjnych) może być wyznaczony w oparciu o analizę pięciu kluczowych elementów: funkcji wejścia, wyjścia, zapytań oraz wewnętrznego modelu danych i interfejsów do systemów zewnętrznych. Są to elementy z założenia wynikające z Wymagań funkcjonalnych użytkownika, a więc zrozumiałe z punktu widzenia Użytkownika funkcjonalnego. Elementy te **nie mogą** wynikać ze sposobu implementacji tych wymagań, czy użytej technologii. Jeżeli oprogramowanie ma realizować określoną funkcjonalność, to bez względu jak ono zostanie zrealizowane, będzie miało

zawsze ten sam rozmiar funkcjonalny. Decyzje dotyczące sposobu realizacji funkcjonalności nie wpływają na rozmiar funkcjonalny oprogramowania.



Powyższy rysunek przedstawia schemat wymiarowanego oprogramowania z punktu widzenia metody IFPUG. Wymiarowanie oprogramowanie udostępnia użytkownikowi funkcje wyjścia / wejścia / zapytań, posiada swoją własną logikę biznesową definiującą sposób przetwarzania danych, przechowuje także określone dane wewnętrzne. Wymiarowane oprogramowanie korzysta z oprogramowania zewnętrznego przy pomocy określonego interfejsu, przy pomocy tego interfejsu ma dostęp do danych zewnętrznych. Zgodnie z metodą punktów funkcyjnych rozmiar oprogramowania wyraża złożoność logiki biznesowej aplikacji, którą później musi zrealizować odpowiedni kod źródłowy. Im bardziej złożona logika biznesowa tym większy rozmiar oprogramowania. Z punktu widzenia metody PF IFPUG złożoność Logiki biznesowej oprogramowania jest określana przez dane wewnętrzne i dane zewnętrzne, które są przez tę logikę przetwarzane, a także funkcje wejścia, wyjścia oraz zapytania, które można przy pomocy tej logiki zrealizować. Zbiór w/w elementów oprogramowania, czyli zbiór funkcji wejścia, wyjścia, zapytań, modelu danych wewnętrznych i zewnętrznych (na rysunku elementy te zostały oznaczone kolorem niebieskim) jest w dalszej części określany jako **model IFPUG**. Model IFPUG jest odzwierciedleniem opisu funkcjonalności lub zmian funkcjonalnych w postaci określonych artefaktów definiowanych przez reguły metody PF IFPUG.

Metoda PF IFPUG wprowadza również pojęcie **granicy oprogramowania**. Granica oprogramowania wyznacza zakres szacowanego oprogramowania, jak i również służy do dookreślenia funkcji wejścia / wyjścia / zapytań jako funkcji przenoszących dane z zewnątrz do wewnątrz granicy lub z wewnątrz na zewnątrz granicy. Wewnętrzny model danych jest również traktowany jako mechanizm utrwalania danych wewnątrz granicy oprogramowania, podczas gdy interfejsy zewnętrzne są modelowane jako zbiory danych dostępne dla oprogramowania poza jego granicami.

Granica danego oprogramowania powinna być zgodnie z założeniami metody wyznaczona z punktu widzenia Użytkownika funkcjonalnego. Wewnątrz granicy oprogramowania MUSI w całości znajdować się logika biznesowa oprogramowania, która w całości zrealizuje daną funkcjonalność. Granica oprogramowania nie jest determinowana przez decyzje technologiczne, infrastrukturalne czy architektoniczne.

Przykład: Wymaganie polegające na wprowadzaniu danych z wniosku jest realizowane przy pomocy dwóch komponentów, jeden odpowiedzialny za dostarczenie interfejsu użytkownika, a drugi za przeprowadzenie kontroli wprowadzonych danych, jednakże z punktu widzenia Użytkownika funkcjonalnego, architektura aplikacji nie jest istotna, i nie ma możliwości zdefiniowania wymagań oddzielnie dla każdego z tych komponentów. W takim wypadku komponenty te nie mogą być rozdzielone granicą oprogramowania, muszą leżeć w obrębie tej samej granicy oprogramowania, stanowią dwa, z potencjalnie wielu elementów wymiarowanego oprogramowania.

W przypadku szacowania zmian oprogramowania należy mieć na uwadze fakt, iż granica oprogramowania jest czymś niezależnym od zmiany – powinna być zdefiniowana dla danego zmienianego oprogramowania jako logicznej całości a nie, dla zmienianego fragmentu.

METODA PF IFPUG – ZASADY

Metoda IFPUG definiuje następujące artefakty jako elementy modelu IFPUG:

- **Elementarny proces (EP – Elementary Process)** – minimalna jednostka funkcjonalności, która spełnia jednocześnie następujące kryteria:
 - jest zrozumiała z punktu widzenia Użytkownika funkcjonalnego,
 - w pełni realizuje transakcję,
 - w całości zawiera się w sobie, tzn. nie występują poza nią żadne dodatkowe czynności niezbędne do jej zainicjowania czy pełnego wypełnienia,
 - jej realizacja pozostawia biznes aplikacji w Stanie stabilnym, tzn. osiąga punkt, w którym Wymaganie funkcjonalne użytkownika zostało w całości wykonane, jego cel został w pełni osiągnięty, nie pozostaje już nic do zrobienia, żeby go osiągnąć.
- **Funkcja wejścia (EI – External Input)** – Elementarny proces przetwarzający lub kontrolujący informacje przychodzące z zewnątrz granicy oprogramowania.
- **Funkcja zapytania (EQ – External Inquiry)** – Elementarny proces przetwarzający lub kontrolujący informacje przesyłane na zewnątrz granicy oprogramowania.
- **Funkcja wyjścia (EO – External Output)** – Elementarny proces przetwarzający lub kontrolujący informacje przesyłane na zewnątrz granicy oprogramowania, obejmujący logikę przetwarzania, wykraczającą poza logikę Funkcji zapytania (EQ).
- **Wewnętrzny plik danych (ILF – Internal Logical File)** – rozpoznawana przez użytkownika grupa logicznie powiązanych danych, utrzymywana wewnątrz granicy wymiarowanego oprogramowania
- **Zewnętrzny plik danych (EIF – External Interface File)** – rozpoznawana przez użytkownika grupa logicznie powiązanych danych, odczytywana przez wymiarowane oprogramowanie, utrzymywana poza granicami wymiarowanego oprogramowania, np. przez oprogramowanie zewnętrzne.
- **Plik danych** – Wewnętrzny plik danych lub Zewnętrzny plik danych.

- **Odniesienie do Pliku danych (FTR – File Type Referenced)** – funkcja odczytu lub zapisu określonego Pliku danych wykonywana w ramach określonego Elementarnego procesu.
- **DET** (ang. Data Element Type) – unikalny, rozpoznawalny z punktu widzenia Użytkownika funkcjonalnego, niepowielony atrybut danych.
- **RET** (ang. Record Element Type) – rozpoznawalny, z punktu widzenia Użytkownika funkcjonalnego, spójny pod względem liczności atrybutów (DETów) w ramach Pliku danych, rekord danych.

Przykład: W przypadku, gdy oprogramowanie umożliwia rejestrowanie dokumentów kancelaryjnych to funkcja rejestracji nowego dokumentu będzie elementarnym procesem typu EI, funkcja prezentująca raport z zarejestrowanych dokumentów będzie elementarnym procesem typu EQ, funkcja prezentująca statystykę tempa rejestracji dokumentów będzie elementarnym procesem typu EO gdyż wylicza dodatkowe dane, nieprzechowywane w aplikacji; zbiór danych przechowujący dane zarejestrowanych dokumentów będzie plikiem danych typu ILF a zbiór danych przechowujący dane klientów organizacji (zarządzanym w ramach innego oprogramowania) będzie plikiem danych typu EIF wykorzystywanym celem identyfikacji nadawcy dokumentu.

W celu wyznaczenia rozmiaru funkcjonalnego oprogramowania konieczne jest określenie dla definiującego to oprogramowanie zbioru Wymagań funkcjonalnych użytkownika wszystkich wyżej zdefiniowanych elementów modelu IFPUG tj. Elementarnych procesów EI, EO i EQ oraz Plików danych ILF i EIF, czyli przygotowanie modelu IFPUG danego oprogramowania. Przygotowanie modelu IFPUG oprogramowania wchodzi w zakres Wymiarowania.

W celu wyznaczenia rozmiaru funkcjonalnego zmiany oprogramowania konieczne jest zidentyfikowanie elementów w istniejącym modelu IFPUG oprogramowania, które ulegają zmianie lub które są usuwane, oraz zidentyfikowanie nowych elementów modelu IFPUG, których utworzenie wynika z Wymagań funkcjonalnych użytkownika, określających zakres modyfikacji, czyli przygotowanie modelu IFPUG modyfikacji. Przygotowanie modelu IFPUG modyfikacji wchodzi w zakres Wymiarowania.

W przypadku Wymiarowania zmiany, kiedy nie istnieje model IFPUG, dla oprogramowania, które jest modyfikowane, konieczne jest w pierwszym kroku przygotowanie modelu IFPUG dla oprogramowania przed zmianą. Przygotowanie w/w modelu IFPUG dla oprogramowania przed zmianą, wchodzi w zakres Wymiarowania danej zmiany.

IDENTYFIKACJA PLIKÓW DANYCH

Identyfikacja Plików danych polega na identyfikacji grupy logicznie powiązanych z punktu widzenia opisywanej funkcjonalności danych. Przy czym wszelkie dane powiązane ze sobą relacją zależności, należy łączyć w pojedyncze Pliki danych.

Dane są powiązane relacją zależności, jeżeli pewne dane (dane zależne) zależą od innych danych (dane nadrzędne).

Mówimy, że dane są zależne od danych nadrzędnych, jeżeli:

- istnieją tylko wtedy, kiedy istnieją dane nadrzędne,
- usunięcie danych nadrzędnych oznacza zawsze usunięcie danych zależnych.

Przykład: Aplikacja do rejestracji dokumentów wpływających do organizacji opcjonalnie korzysta z systemu CRM celem identyfikacji nadawców. Dane podmiotów – nadawców z systemu CRM, z punktu widzenia szacowania aplikacji do rejestracji dokumentów, są modelowane jako plik danych EIF.

Dla każdego pliku danych należy określić jego unikalne atrybuty (DET – Data Element Type).

Atrybut pliku danych to każdy rozpoznawalny przez użytkownika, unikalny atrybut pliku danych, który jest zarządzany, zapisywany, aktualizowany lub odczytywany w ramach elementarnych procesów szacowanego oprogramowania. Atrybutami są również atrybuty związków danego Pliku danych z innymi Plikami danych, przy czym w zależności od nawigowalności tego związku, atrybut związku jest określany w jednym lub w obu powiązanych plikach danych.

Mając określone atrybuty poszczególnych plików danych należy zidentyfikować w ramach plików danych rekordy (RET – Record Element Type). Rekord grupuje w ramach pliku danych atrybuty które występują w liczności innej niż 1:1 do pliku danych (inaczej mówiąc są one w relacji kompozycji). Domyślnie każdy plik danych ma jeden rekord.

Przykład: Zbiór danych faktura to dane faktury plus lista pozycji faktury. Pozycja faktury jest elementem pliku danych Faktura i jest modelowana jako rekord w tym pliku danych. Pozycja faktury nie istnieje bez faktury, usunięcie faktury usuwa wszystkie powiązanie z nią pozycje faktury.

Przykład: Zbiór danych wniosek, posiada strukturę drzewiastą. Składa się z sekcji A, B, C i D, każda zawiera unikatowy zestaw atrybutów, sekcja A zawiera listę LA, która zawiera listę elementów posiadających unikatową strukturę, sekcja C również zawiera listę LB, przy czym elementy listy LB także zawierają listę elementów LLB. Cała opisana wyżej struktura to jeden Plik danych (ILF), który zawiera 4 rekordy (RET), jeden odpowiada wnioskowi i jego sekcjom (sekcje, które występują w relacji 1:1 do wniosku, należy traktować tylko jako element porządkujący atrybuty we wniosku, dlatego sekcji tych nie należy traktować jako odrębne RETy), kolejne elementom listy LA i LB, i ostatni najbardziej zagnieżdżonemu elementowi listy LLB.

Poziom zagnieżdżenia relacji wzajemnie ze sobą powiązanych danych, czy też cykliczność tych relacji nie ma wpływu na liczbę identyfikowanych Plików danych (ILF) czy też rekordów (RET).

Przykład: Zbiór danych składa się z danych osób, połączonych ze sobą relacją oznaczającą, że jedna osoba może reprezentować drugą, przy czym relacja ta nie posiada żadnych atrybutów tzn. jeżeli występuje między osobą A, a osobą B, to znaczy, że osoba A może reprezentować osobę B, a jeżeli nie występuje, to nie może. W rezultacie zbiór może tworzyć sieć powiązanych ze sobą osób. W takim wypadku zbiór taki identyfikujemy jako jeden Plik danych (ILF), posiadający jeden rekord (RET).

IDENTYFIKACJA ELEMENTARNYCH PROCESÓW

Elementarne procesy identyfikujemy jako minimalne, niezależne i odrębne aktywności użytkownika które w pełni realizują określony cel Użytkownika funkcjonalnego.

W ramach jednego Elementarnego procesu łączy się funkcje posiadające własną logikę biznesową, które są wariantami lub alternatywnymi przebiegami realizacji określonego celu użytkownika.

Różne warianty logiki biznesowej mogą wynikać m.in. z:

- różnych decyzji użytkownika,
- różnych wartości wprowadzonych danych (np. wprowadzona data jest wcześniejsza lub późniejsza od bieżącej, użytkownik podał dane kompletne lub niekompletne),
- różnych warunków uruchomienia Elementarnego procesu (np. różne uprawnienia użytkowników),
- różnych stanów systemu (np. występowania lub niewystępowania określonych danych utrwalonych w granicach oprogramowania),
- różnych sposobów porządkowania i prezentowania danych np. ukrywania niewymaganych pól, prezentacji w sekcjach.

Różne warianty logiki biznesowej mogą korzystać z różnych Plików danych lub różnych atrybutów.

Tabela 1. Różne formy logiki biznesowej – składniki Elementarnego procesu, które nie powinny być identyfikowane jako odrębne Elementarne procesy.

Lp.	Forma logiki biznesowej
1	Walidacja danych Przykład: Przy wprowadzaniu nowego wniosku sprawdzane jest czy wszystkie pola obowiązkowe są wypełnione.
2	Wykonywanie obliczeń matematycznych Przykład: Przy generowaniu raportu obliczana jest średnia wielkość deklarowanej powierzchni, na podstawie wszystkich zatwierdzonych wniosków.
3	Przekształcanie wartości Przykład: Na podstawie deklarowanej powierzchni, wyznaczana jest grupa, do której należy wniosek, gdzie przynależność do grupy jest definiowana przez przedział deklarowanej powierzchni.
4	Filtrowanie i selekcjonowanie danych wg określonych kryteriów Przykład: Na podstawie podanego przedziału dat wyznaczany jest ze wszystkich wniosków, zbiór wniosków, których data złożenia mieści się w przedziale.
5	Sprawdzenie czy spełnione są warunki Przykład: Oprogramowanie inaczej przetwarza wnioski, które zostały złożone w przyjętym okresie składania wniosków, a inaczej wniosku złożone po tym okresie. Oprogramowanie sprawdza, czy spełnione zostały warunki, określonego wariantu przetwarzania wniosku i uruchamia go.

	<p>Przykład: Oprogramowanie inaczej prezentuje dane użytkownikom, którzy pierwszy raz wykonują daną operację, niż użytkownikom, którzy wykonują ją po raz kolejny. Oprogramowanie sprawdza, czy użytkownik po raz pierwszy wykonuje daną operację, jeżeli tak to uruchamia odpowiedni wariant prezentacji danych.</p>
6	<p>Aktualizacja jednego lub więcej Plików wewnętrznych (ILF)</p> <p>Przykład: Oprogramowanie zapisuje wprowadzone dane wniosku.</p>
7	<p>Odwołanie się do jednego lub więcej Plików wewnętrznych (ILF) lub zewnętrznych (EIF)</p> <p>Przykład: Oprogramowanie wywołuje usługę, w celu sprawdzenia, czy w systemie zewnętrznym występuje podmiot posiadający określony nr NIP (EIF).</p>
8	<p>Pobranie danych lub informacji kontrolnych</p> <p>Przykład: W celu prezentacji listy wniosków, dane wniosków są odczytywane.</p>
9	<p>Utworzenie danych poprzez transformację istniejących danych</p> <p>Przykład: Znak sprawy jest tworzony przez system na podstawie typu dokumentu inicjującego sprawę i daty utworzenia sprawy</p>
10	<p>Zmiana zachowania aplikacji</p> <p>Przykład: Kontrola wniosku może być przeprowadzona albo w trybie zwykłym albo uproszczonym, tryb kontroli cechuje się specyficznym zestawem czynności kontrolnych. Proces kontroli wniosku zostanie zmieniony, jeżeli zmienimy typ kontroli wniosku, ze zwykłego na uproszczony.</p>
11	<p>Przygotowanie danych w celu ich prezentacji poza granicą oprogramowania</p> <p>Przykład: Lista prezentowanych wniosków jest odpowiednio formatowana i wyświetlana użytkownikowi</p>
12	<p>Umożliwienie przyjęcia danych spoza granicy oprogramowania</p> <p>Przykład: System wyświetla formatkę, która umożliwia użytkownikowi wprowadzanie i zapisanie danych wniosku</p>
13	<p>Sortowanie i porządkowanie danych</p> <p>Przykład: System sortuje wnioski wg daty złożenia wniosku</p> <p>Przykład: System grupuje wnioski wg właściwości miejscowej</p>

Przykład: System umożliwia rejestrację danych podmiotów, przy czym rejestrowane są zarówno podmioty jako osoby fizyczne, jak i prawne. Elementarny proces EI Rejestracja podmiotu obsługuje oba te przypadki a zbiór atrybutów tego procesu jest łącznym zbiorem wszystkich atrybutów osób prawnych i fizycznych.

Przykład: System umożliwia weryfikację wniosku, poprzez wprowadzanie wyników weryfikacji, przy czym raz jako użytkownik, który ma węższe, a raz jako użytkownik, który ma szersze uprawnienia, co powoduje, że dla użytkownika, który posiada węższe uprawnienia nie są dostępne pewne opcje. Jeden Elementarny proces EI obejmuje zarówno weryfikację wykonywaną przez użytkownika o węższych uprawnieniach, jak i użytkownika o szerszych uprawnieniach.

Przykład: System umożliwia weryfikację wniosku, poprzez wprowadzanie wyników weryfikacji, przy czym jeżeli weryfikacja miała już wcześniej miejsce, to system wyświetla dodatkowo informacje dotyczące wcześniejszych weryfikacji. Zarówno weryfikacja wykonywana po raz pierwszy, jak i weryfikacja wykonywana za każdym następnym razem, to ten sam Elementarny proces EI.

Zidentyfikowane elementarne procesy należy sklasyfikować przypisując im jeden z trzech określonych przez metodę typów tj. EI, EO lub EQ.

Elementarny proces jest procesem typu EI, jeśli:

- 1) Zawiera logikę przetwarzania walidującą dane wprowadzane i utrwalane w granicach oprogramowania w ramach tego procesu, lub
- 2) Jego podstawowym zadaniem jest zarządzanie (dodawanie, usuwanie, modyfikacja) jednym lub więcej plikami danych w ramach procesu, lub
- 3) Jego podstawowym zadaniem jest modyfikacja zachowania oprogramowania po zakończeniu realizacji procesu (poprzez modyfikację określonych plików danych).

Przykład: Funkcja rejestracji dokumentu kancelaryjnego jest procesem EI, gdyż umożliwia wprowadzenie do oprogramowania danych rejestrowanego dokumentu.

Elementarny proces jest procesem typu EO, jeśli jego podstawowym zadaniem jest prezentacja danych użytkownikowi z wykorzystaniem przynajmniej jednego ze sposobów przetwarzania danych:

- 1) Obliczeń matematycznych.
- 2) Aktualizacji pliku lub plików ILF.
- 3) Utworzenia prezentowanych danych.
- 4) Modyfikacja zachowania oprogramowania po zakończeniu realizacji procesu (poprzez modyfikację określonych plików danych).

Przykład: Funkcja utworzenia raportu z zarejestrowanych dokumentów kancelaryjnych jest procesem typu EO, gdyż zawiera informacje o liczbie zarejestrowanych dokumentów z podziałem na stanowiska rejestracji.

Elementarny proces jest procesem typu EQ, jeśli jego zadaniem jest prezentacja informacji utrwalonych w granicach oprogramowania i nie zawiera przetwarzania danych wskazanego dla procesów typu EO.

Przykład: Funkcja wyszukiwania dokumentów kancelaryjnych jest procesem typu EQ, gdyż prezentuje wyłącznie dane dokumentów zapisane w bazie.

Dla każdego zidentyfikowanego i sklasyfikowanego elementarnego procesu należy następnie określić, z jakich plików danych ten proces korzysta (odczyt lub zapis) oraz jakie atrybuty są w ramach procesu wykorzystywane.

Jako atrybuty wykorzystywane przez Elementarny proces należy traktować:

- unikalne i rozpoznawalne przez użytkownika atrybuty danych przekraczające granicę systemu podczas realizacji procesu,
- jeden atrybut, jeśli elementarny proces może prezentować użytkownikowi komunikaty (proste błędy / potwierdzenia) związane z realizacją procesu – niezależnie od liczby tych komunikatów,

Nie należy traktować jako atrybutów elementarnych procesów:

- opisów na formatach, nagłówków kolumn, nazw pól,
- znaczników wersji, znaczników typów, znaczników czasowych nie mających znaczenia biznesowego (tj. wpływającego na procesy biznesowe użytkownika),
- numerowania stron, wierszy, liczb porządkowych w tabelach,
- elementów nawigacyjnych interfejsu użytkownika, które nie są jednocześnie atrybutami przekraczającymi granicę oprogramowania
- elementów wspomagających wprowadzanie danych na interfejsie użytkownika,
- atrybutów generowanych w ramach realizacji procesu, ale nie przekraczających granicy oprogramowania,
- atrybutów odczytywanych z plików danych nieprzekraczających granicy oprogramowania.

Przykład: Raport z liczby zarejestrowanych dokumentów kancelaryjnych jest tworzony na zadany dzień i prezentuje liczbę dokumentów zarejestrowanych w poszczególnych stanowiskach rejestracyjnych. Na raporcie jest też prezentowana wersja formularza raportu oraz godzina wygenerowania. Z punktu widzenia użytkownika te dwa ostatnie parametry są nieistotne - są to parametry techniczne dostawcy oprogramowania kancelaryjnego. Tym samym, do atrybutów procesu zalicza się: atrybut dnia, na jaki jest tworzony raport, liczbę raportów i identyfikację stanowiska. Dodatkowo mogą być również zliczone przycisk wykonania raportu na zadane parametry (jako realizacja akcji tego procesu) oraz ew. 1 DET dla komunikatu o błędnych parametrach.

KLASYFIKACJA ELEMENTARNYCH PROCESÓW

Klasyfikacja Elementarnych procesów zależy od celów jakim służą oraz logiki przetwarzania jaką w sobie zawierają. Zależności te określają poniższe tabele.

Tabela 2. Cele Elementarnych procesów w zależności od ich klasyfikacji.

Cel	EI	EO	EQ
Modyfikacja zachowania oprogramowania	Główny cel	Cel wspomagający	Niedozwolone
Zarządzanie plikami danych ILF	Główny cel	Cel wspomagający	Niedozwolone
Prezentacja danych użytkownikowi	Cel wspomagający	Główny cel	Główny cel

Tabela 3. Możliwe logiki przetwarzania w zależności od klasyfikacji Elementarnego procesu.

Logika przetwarzania	EI	EO	EQ
1. Przeprowadzane są walidacje atrybutów danych	T	T	T
2. Realizowane są obliczenia matematyczne na atrybutach danych	T	O*	N
3. Wartości są konwertowane pomiędzy różnymi systemami miar, jednostek lub z wykorzystaniem tablic konwersji	T	T	T

4. Dane są filtrowane i wybierane w oparciu o określone kryteria	T	T	T
5. Weryfikowane są reguły przetwarzania danych lub warunki biznesowe przeprowadzenia danej akcji	T	T	T
6. Jeden lub więcej plik ILF jest zaktualizowany	O*	O*	T
7. Jeden lub więcej plików ILF lub EIF jest odczytywany	T	T	O
8. Dane są odczytywane	T	T	O
9. Dane są tworzone	T	O*	N
10. Zachowanie oprogramowania jest zmienione po realizacji procesu	O*	O*	N
11. Dane są odczytywane i prezentowane poza granicą oprogramowania	T	O	O
12. Dane są wprowadzane w wyniku czego znajdują się w granicach oprogramowania (ale nie są utrwalane)	O	T	T
13. Dane są sortowane lub przetwarzane pod kątem sposobu prezentacji	T	T	T

- T – elementarny proces może realizować daną logikę przetwarzania
- N – elementarny proces nie może realizować danej logiki przetwarzania
- O – elementarny proces musi realizować daną logikę przetwarzania
- O* – elementarny proces musi realizować przynajmniej jedną z logik oznaczoną dla niego jako O* (w ramach kolumny)

Przykład:

Dla oprogramowania zdefiniowano elementarny proces EP1 Rejestracja przyjęcia dokumentu. Proces ten zakwalifikowano jako EI, gdyż jego głównym zadaniem jest zarządzanie (poprzez wprowadzanie) dokumentami kancelaryjnymi. Zgodnie z Tabelą 3 w procesie typu EI zawsze musi wystąpić logika nr 12 oraz musi wystąpić przynajmniej jedna z logik nr 6 i 10. Warunek ten jest spełniony, co wynika z tabeli poniżej.

Tabela 4. Lista kontrolna logik przetwarzania dla procesu EP1.

Logika przetwarzania	EI	Weryfikacja
1. Przeprowadzane są walidacje atrybutów danych	T	Tak - EP1 przeprowadza walidacje
2. Realizowane są obliczenia matematyczne na atrybutach danych	T	Nie - EP1 nie przeprowadza wyliczeń
3. Wartości są konwertowane pomiędzy różnymi systemami miar, jednostek lub z wykorzystaniem tablic konwersji	T	Nie - EP1 nie przeprowadza konwersji

4. Dane są filtrowane i wybierane w oparciu o określone kryteria	T	Nie - EP1 nie posiada filtracji i wybierania
5. Weryfikowane są reguły przetwarzania danych lub warunki biznesowe przeprowadzenia danej akcji	T	Tak - EP1 przeprowadza weryfikację reguł biznesowych
6. Jeden lub więcej plik ILF jest zaktualizowany	O*	Tak - aktualizowany jest zbiór ILF dokumentów kancelaryjnych
7. Jeden lub więcej plików ILF lub EIF jest odczytywany	T	Tak - odczytywany jest opcjonalnie EIF z danymi nadawców
8. Dane są odczytywane	T	Tak - dane są odczytywane
9. Dane są tworzone	T	Tak - tworzony jest nowy obiekt
10. Zachowanie oprogramowania jest zmienione po realizacji procesu	O*	Nie - zachowanie oprogramowania się nie zmienia
11. Dane są odczytywane i prezentowane poza granicą oprogramowania	T	Tak - część danych odczytanych z systemu jest prezentowana na formularz
12. Dane są wprowadzane w wyniku czego znajdują się w granicach oprogramowania (ale nie są utrwalane)	O	Tak - użytkownik wprowadza dane przez formularz rejestracji
13. Dane są sortowane lub przetwarzane pod kątem sposobu prezentacji	T	Nie - proces EP1 nie sortuje danych.

WYLICZANIE ROZMIARU FUNKCJONALNEGO ELEMENTÓW MODELU IFPUG

Rozmiar funkcjonalny plików danych jest wypadkową liczby atrybutów i rekordów w danym pliku. Rozmiar funkcjonalny elementarnych procesów jest wypadkową liczby atrybutów procesu i liczby odwołań do różnych Plików danych wykonywanych w ramach danego Elementarnego procesu. Dla każdego z typów artefaktów modelu jest zdefiniowana odpowiednia tabela wyliczeniowa.

ILF	DET 1 - 19	DET 20 - 50	DET > 50
RET 1	LOW (7)	LOW (7)	AVERAGE (10)
RET 2 - 5	LOW (7)	AVERAGE (10)	HIGH (15)
RET > 5	AVERAGE (10)	HIGH (15)	HIGH (15)
EIF	DET 1 - 19	DET 20 - 50	DET > 50
RET 1	LOW (5)	LOW (5)	AVERAGE (7)
RET 2 - 5	LOW (5)	AVERAGE (7)	HIGH (10)
RET > 5	AVERAGE (7)	HIGH (10)	HIGH (10)
EI	DET 1 - 4	DET 5 - 15	DET > 15
FTR < 2	LOW (3)	LOW (3)	AVERAGE (4)
FTR 2	LOW (3)	AVERAGE (4)	HIGH (6)
FTR > 2	AVERAGE (4)	HIGH (6)	HIGH (6)
EQ	DET 1 - 5	DET 6 - 19	DET > 19
FTR < 2	LOW (3)	LOW (3)	AVERAGE (4)
FTR 2	LOW (3)	AVERAGE (4)	HIGH (6)
FTR > 2	AVERAGE (4)	HIGH (6)	HIGH (6)
EO	DET 1 - 5	DET 6 - 19	DET > 19
FTR < 2	LOW (4)	LOW (4)	AVERAGE (5)
FTR 2 - 3	LOW (4)	AVERAGE (5)	HIGH (7)
FTR > 3	AVERAGE (5)	HIGH (7)	HIGH (7)

Suma rozmiarów funkcjonalnych wszystkich elementów modelu IFPUG jest rozmiarem oprogramowania w punktach funkcyjnych.

Przykład: Plik danych ILF o 25 atrybutach i dwóch rekordach ma złożoność 10 punktów funkcyjnych. Elementarny proces EQ o 3 atrybutach i jednym wykorzystywanym pliku FTR ma złożoność 3 punkty funkcyjne.

WYLICZANIE ROZMIARU FUNKCJONALNEGO ZMIAN

Do wyliczania rozmiaru funkcjonalnego zmian funkcjonalnych stosuje się metodę PF IFPUG z rozszerzeniem opracowanym przez organizację NESMA.

Ze zmianami funkcjonalnymi mamy do czynienia w przypadku zmian funkcjonalności, które skutkują:

- dodaniem nowych Plików danych lub Elementarnych procesów,
- usunięciem istniejących Plików danych lub Elementarnych procesów,
- modyfikacją istniejących Plików danych lub Elementarnych procesów.

Rozmiar funkcjonalny zmian polegających na dodaniu nowych Plików danych lub Elementarnych procesów równy jest rozmiarowi tych Plików danych lub Elementarnych procesów.

Rozmiar funkcjonalny zmian polegających na usunięciu istniejących Plików danych lub Elementarnych procesów równy jest iloczynowi współczynnika wpływu (wynoszącemu 0,4) i rozmiaru tych Plików danych lub Elementarnych procesów.

Powyższy współczynnik dla usuwanych elementów nie jest stosowany w przypadku wyłączenia całej aplikacji lub całego modułu z eksploatacji i utrzymania, dla takich przypadków stosuje się wycenę ekspercką.

W przypadku modyfikacji istniejącej funkcjonalności stosuje się dodatkowe reguły, większość z nich bazuje na opracowaniu organizacji NESMA wspomnianego wcześniej w dokumencie.

WYLICZANIE ROZMIARU ZMIAN PLIKÓW DANYCH

Plik danych jest uznawany za zmieniony, jeśli są do niego dodane atrybuty, odjęte atrybuty, zmieniony jest typ atrybutu lub zmieniona jest struktura rekordów. Inną możliwą zmianą Pliku danych jest zmiana jego typu (EIF / ILF).

Celem wyliczenia rozmiaru zmiany Pliku danych należy określić liczbę zmienianych atrybutów, a także liczbę atrybutów przed zmianą Pliku danych. Mając te dane wylicza się **procentowy wskaźnik zmiany**. Wskaźnik ten służy do wyliczenia **wskaźnika wpływu zmiany**. Następnie należy wyliczyć rozmiar funkcjonalny Pliku danych po zmianie. Iloczyn wskaźnika wpływu zmiany oraz rozmiaru Pliku danych po zmianie wyznacza rozmiar funkcjonalny zmiany Pliku danych.

Procentowy wskaźnik zmiany: $(\text{liczba zmienionych atrybutów} / \text{liczba atrybutów przed zmianą}) * 100\%$

Wskaźnik wpływu wyznacza się w oparciu o poniższą tabelę:

Procentowy wskaźnik zmiany	$p \leq 30\%$	$30\% < p \leq 60\%$	$60\% < p \leq 100\%$	$p > 100\%$
Wskaźnik wpływu zmiany	0.25	0.50	0.75	1.00

W przypadku, gdy zmianie ulegnie jedynie typ pliku danych lub struktura rekordów lub jedno i drugie to wskaźnik wpływu dla tej modyfikacji ma wartość 0.40.

W przypadku, gdy jednocześnie uległy zmianie atrybuty, to jako wskaźnik wpływu, przyjmuje się wartość większą spośród wskaźnika wynikającego z modyfikacji typu lub struktury rekordów oraz wskaźnika wynikającego ze zmian atrybutów.

W przypadku, gdy dwa pliki danych są łączone w jeden to traktuje się to jako usunięcie dwóch plików i dodanie jednego nowego. W przypadku, gdy istniejący plik jest dzielony, to traktuje się to jako usunięcie jednego pliku i dodanie dwóch nowych.

Przykład: Plik danych ILF A posiada 15 atrybutów i jeden rekord. W ramach modyfikacji jest dodawanych kolejnych pięć a dodatkowo dla jednego istniejącego jest zmieniany typ danych. Złożoność pliku przed modyfikacją to 7 PF. Procentowy wskaźnik zmiany to $(6/15) \cdot 100\% = 40\%$. Wskaźnik wpływu to 0.50. Złożoność pliku po modyfikacji to 10 PF. Złożoność zmiany to $10 \text{ PF} \cdot 50\% = 5 \text{ PF}$.

WYLICZANIE ROZMIARU ZMIAN ELEMENTARNYCH PROCESÓW

Elementarny proces jest uznawany za zmieniony, jeśli zachodzi jeden z poniższych warunków:

- dodano, usunięto lub zmodyfikowano atrybut Elementarnego procesu,
- w ramach Elementarnego procesu: zmodyfikowano zakres odwołania do Pliku danych, dodano nowe odwołanie do Pliku danych, usunięto odwołanie do Pliku danych,
- zmieniły się reguły przetwarzania danych w ramach Elementarnego procesu.

Celem wyliczenia rozmiaru zmiany Elementarnego procesu należy:

- 1) określić liczbę zmienianych atrybutów (przy czym przez zmieniane atrybuty rozumiemy dodane lub usunięte w wyniku zmiany atrybuty Elementarnego procesu) oraz liczbę atrybutów Elementarnego procesu przed zmianą i na ich podstawie wyliczyć wskaźnik zmiany atrybutów zgodnie ze wzorem:

Procentowy wskaźnik zmiany atrybutów: $(\text{liczba zmienionych atrybutów} / \text{liczba atrybutów przed zmianą}) \cdot 100\%$

- 2) określić liczbę zmienianych plików (przy czym przez zmieniane pliki rozumiemy Pliki danych, z których korzysta Elementarny proces i które ulegają zmianie lub Pliki danych, z których zaczął korzystać Elementarny proces lub Pliki danych, z których przestał korzystać Elementarny proces) oraz liczbę plików przed zmianą, (tj. liczbę Plików danych, z których korzystał Elementarny proces przed zmianą) i na ich podstawie wyliczyć wskaźnik zmiany plików zgodnie ze wzorem:

Procentowy wskaźnik zmiany plików: $(\text{liczba zmienionych plików} / \text{liczba plików przed zmianą}) \cdot 100\%$

- 3) na podstawie wskaźnika zmiany atrybutów oraz wskaźnika zmiany plików wyznaczyć wartość wskaźnika wpływu,
- 4) określić rozmiar Elementarnego procesu po zmianie,
- 5) wyliczyć rozmiar zmiany Elementarnego procesu, jako iloczyn wskaźnika wpływu zmiany oraz rozmiaru Elementarnego procesu po zmianie.

ZASADY STOSOWANIA METODY PF IFPUG W ARIMR

STOSOWANIE PF IFPUG NA POTRZEBY FUNKCJONALNOŚCI BIZNESOWEJ

ID reguły	IFPUG001: Identyfikacja Plików danych - wyłączenia
Opis reguły	<p>Z punktu widzenia wymiarowanej funkcjonalności Plików danych (czy to w całości czy w części) nie stanowią:</p> <ol style="list-style-type: none"> 1) Dane, które nie wynikają z opisu danej funkcjonalności. 2) Dane, które nie są zmieniane w wyniku wykonania funkcjonalności danego oprogramowania, w tym między innymi: <ol style="list-style-type: none"> a) domyślne wartości, b) definicje zakresów wartości, c) kryteria poprawności atrybutów, d) definicje formatów, e) szablony lub style definiujące wygląd interfejsu użytkownika, o ile system nie pozwala na ich edycję, f) szablony generowanych dokumentów lub raportów, o ile system nie pozwala na ich edycję. 3) Dane złączeniowe, dane niezbędne do przechowywania powiązań pomiędzy obiektami np. tabel złączeniowych, indeksów, widoków itp. 4) Zbiory tymczasowe (robocze, kontrolne, częściowe, pośrednie itp.) tworzone tylko na czas wykonania danej funkcjonalności. 5) Dane redundantne, powielane w celu optymalizacji czy zapewnienia wydajności danej funkcjonalności (np. przechowywanie tych samych danych w różnych formatach).
Przykłady	<ol style="list-style-type: none"> A. Przy realizacji funkcjonalności rejestracji dokumentu system korzysta z danych, które zawierają informacje o miejscu przechowywania słowników i szablonów dokumentów (np. adres i nazwa bazy danych, nazwa tabeli). Struktury te nie występują w opisie funkcjonalności oprogramowania. Przy wymiarowaniu struktury te nie są identyfikowane jako Pliki danych, ani części Plików danych. B. Funkcjonalność generowania decyzji korzysta z szablonów dokumentów. Oprogramowanie nie posiada funkcjonalności, która pozwalałaby zmienić te szablony. Szablony te nie są traktowane jako Pliki danych, ani jako część Plików danych. C. Funkcjonalność rejestracji wniosku korzysta z atrybutu data złożenia wniosku. Istnieje reguła, która określa, w jakim przedziale powinna się mieścić data złożenia wniosku, aby wniosek mógł zostać zaakceptowany. System nie umożliwia zmiany akceptowalnego zakresu dat. Dane akceptowalnego zakresu dat nie są traktowane jako Plik danych, ani jako część Pliku danych. D. Oprogramowanie korzysta z danych pozwalających na wyznaczenie domyślnych wartości, z jakimi inicjalizowana jest sprawa – domyślnie kolejny z sekwencji, data utworzenia – domyślnie bieżąca, typ sprawy – domyślny dla typu dokumentu inicjującego. Danych tych nie można zmienić przy pomocy

	<p>funkcjonalności oprogramowania. Danych tych nie traktuje się jako Pliku danych, ani jako części Pliku danych.</p> <p>E. Funkcjonalność korzysta z Plików danych, które odpowiadają obiektom klasy A i klasy B, pomiędzy którymi występuje relacja wiele do wielu. Funkcjonalność została zrealizowana z użyciem relacyjnej bazy danych. Do przechowywania danych klas i związku między nimi potrzebne są 3 tabele: jedna przechowuje dane obiektów klasy A, druga dane obiektów klasy B, a trzecia jest tabelą złączeniową przechowującą związki pomiędzy nimi. Jako Pliki danych, z których korzysta funkcjonalność uznaje się wyłącznie dane odpowiadające klasie A i dane odpowiadające klasie B (2 Pliki danych), tablicy złączeniowej nie traktuje się jako odrębnego Pliku danych. Natomiast atrybuty wskazujące związek pomiędzy klasami A i B są traktowane jako atrybuty (DET) każdego z tych Plików danych.</p> <p>F. Funkcjonalność naliczania w celu określenia kwoty płatności wyznacza pewne wartości pośrednie. Jeżeli wartości te, nie są prezentowane użytkownikowi oraz nie są przechowywane w systemie w celu wykorzystania ich przez inne, aktualnie występujące funkcjonalności, to nie traktuje się jako Plików danych.</p> <p>G. Funkcjonalność obejmuje prezentację danych, których przygotowanie jest złożone obliczeniowo. Przy jej realizacji podjęto decyzję, że w celu optymalizacji wydajności, przetworzone dane są buforowane, w celu ich prezentacji przy kolejnym uruchomieniu funkcjonalności. Operacja ta jest przeźroczysta dla użytkownika funkcjonalności, z jego punktu widzenia jego wymagań, nie ma znaczenia czy dane są buforowane czy nie. Dane te są redundantne, tzn. powstają na podstawie innych danych. Danych tych nie traktuje się jako odrębnych Plików danych.</p>
--	---

ID reguły	IFPUG002: Zależność pomiędzy ILF, a EIF
Opis reguły	Jeśli dane wewnętrzne aplikacji A będące elementem pliku ILF, są udostępniane aplikacji B, jako EIF, to do oszacowania rozmiaru pliku EIF, brane są pod uwagę tylko te atrybuty, które są udostępniane aplikacji B przez aplikację A. W takim wypadku rozmiar ILF aplikacji A, nie wpływa na rozmiar pliku EIF aplikacji B.
Przykład	<p>Aplikacja wprowadzania i kontroli wniosków przetwarza plik danych Wniosek składający się z 45 atrybutów. Aplikacja wprowadzania i kontroli udostępnia aplikacji naliczania 34 z 45 atrybutów wniosku.</p> <p>Dla aplikacji wprowadzania i kontroli zostanie zidentyfikowany plik ILF o liczbie DET: 45. Dla aplikacji naliczania zostanie zidentyfikowany plik EIF o liczbie DET: 34.</p>

ID reguły	IFPUG003: Pliki danych – niezależność od sposobu realizacji
-----------	---

Opis reguły	Jeżeli postać, sposób przechowywania czy formatowanie danych nie wynika z opisu danej funkcjonalności, to nie mają one znaczenia przy identyfikacji Plików danych.
Przykład	<p>Mimo, że oprogramowanie przechowuje załączniki mapowe w postaci plików graficznych oraz wpisów w bazie danych, to identyfikowany jest jeden Plik danych reprezentujący załączniki mapowe jako całość, ponieważ opis funkcjonalności nie odnosi się do technicznych składowych załączników mapowych.</p> <p>W celu optymalizacji funkcjonalności oprogramowanie, w zależności od metody wprowadzania, raz zapisuje dane podmiotu w pliku XML, a innym razem w relacyjnej bazie danych. Opis funkcjonalności nie wskazuje na konieczność przechowywania tych danych w dwóch różnych postaciach. W takim wypadku identyfikowany jest jeden Plik danych reprezentujący dane Podmiotu.</p>

ID reguły	IFPUG005: Definiowanie zbioru wartości atrybutu w Elementarnych procesach
Opis reguły	<p>Jeżeli Elementarny proces umożliwia określenie zbioru wartości pojedynczego atrybutu, zliczany dla tego atrybutu 1 DET, bez względu na to, z ilu i jakich kontrolek oraz wartości korzysta się do określenia tego zbioru np. wielowyboru ze słownika, wyboru wartości z kilku połączonych słowników, określenia początku i końca przedziału wartości itp.</p> <p>Wybór, przy pomocy interfejsu użytkownika, jednego lub więcej rekordów z wyselekcjonowanego zbioru zaprezentowanego, np. w postaci listy, należy zliczać jako 1 DET danego EP.</p>
Przykład	<p>Jeżeli Elementarny proces umożliwia wyszukiwanie dokumentów poprzez podanie zakresu daty rejestracji: wartość wartości „od – do”, to jest to zliczane jako 1 DET, gdyż odnosi się on do pojedynczego atrybutu: daty rejestracji.</p> <p>Jeżeli oprogramowanie umożliwia wybór pojedynczego dokumentu z listy dokumentów, to należy taką możliwość wyboru zliczyć jako pojedynczy atrybut EP, w którym ta możliwość występuje (1 DET), gdyż dotyczy pojedynczego atrybutu wskazującego na konkretny dokument (identyfikator dokumentu)</p> <p>Jeżeli Elementarny proces umożliwia dla atrybutu „typ dokumentu” wybór z listy wartości będących połączeniem wartości słowników „typy dokumentów zwykłych” oraz „typy dokumentów specjalnych”, to zliczane jest to jako 1 DET, gdyż dotyczy pojedynczego atrybutu „typ dokumentu”.</p> <p>Jeżeli Elementarny proces umożliwia dla atrybutu „gmina” wybór gminy ze słownika hierarchicznego, którego wartości można zawęzić poprzez wybór województwa oraz powiatu, a także poprzez ręczne wpisanie nazwy gminy i wyszukanie pasujących wartości w słowniku, to zliczany jest to jako 1 DET, gdyż dotyczy pojedynczego atrybutu „gmina”.</p>

ID reguły	IFPUG006: Atrybuty Elementarnego procesu obejmującego modyfikację danych.
Opis reguły	W przypadku aktualizacji danych, dane tego samego typu, które wielokrotnie przekraczają granicę oprogramowania np. wyświetlenie dotychczasowych wartości, zmiana i przesłanie nowych wartości, wyświetlenie zmienionych wartości, zliczane są tylko raz, bez względu na to ile razy przekraczają granicę oprogramowania.
Przykład	Elementarny proces edycji danych podmiotu prezentuje dane istniejące w systemie i umożliwia ich aktualizację. System prezentuje 7 atrybutów podmiotu, z czego 5 można edytować. Liczba DET dla procesu to 8 – 7 unikalnych atrybutów przekracza granicę systemu oraz jedna akcja potwierdzenia zmiany danych podmiotu.

ID reguły	IFPUG007: Wymiarowanie procesu usuwania danych
Opis reguły	Jeżeli Elementarny proces obejmuje usuwanie danych, to zlicza się jedynie atrybuty, których przekroczenie granicy oprogramowania jest niezbędne do skutecznego usunięcia tych danych, a nie wszystkie atrybuty usuwanych danych.
Przykład	Oprogramowanie udostępnia funkcję usuwania dokumentów. Dokument zidentyfikowany jako plik ILF Dokument posiada 21 atrybutów. Jako potwierdzenie usunięcia dokumentu system prezentuje nazwę usuwanego dokumentu. Proces EI Usuwanie dokumenty będzie miał 3 atrybuty: wskazanie (id) usuwanego dokumentu, komunikat potwierdzenia, prezentowaną nazwę usuwanego dokumentu.

ID reguły	IFPUG008: Unikalność atrybutów Elementarnych procesów
Opis reguły	Unikalny atrybut Elementarnego procesu jest dla danego procesu zliczany raz - niezależnie od tego ile razy przekracza granicę oprogramowania w czasie realizacji tego procesu.
Przykład	W funkcji wyszukiwania dokumentów jednym z parametrów wyszukiwania jest data wpływu. Data ta jest prezentowana również na liście wyszukanych dokumentów. Atrybut ten jest zliczany jako 1 DET.

ID reguły	IFPUG009: Identyfikacja Plików danych – wyłączenia dotyczące funkcjonalności biznesowej
Opis reguły	<p>Z punktu widzenia funkcjonalności biznesowej, dane stanowiące jej parametryzację, służące autentykacji i autoryzacji użytkowników, a także służące monitorowaniu systemu:</p> <ul style="list-style-type: none"> nie są identyfikowane jako Pliki danych,

	<ul style="list-style-type: none"> nie są zliczane odwołania do nich, jako odwołania do Plików danych przy wymiarowaniu Elementarnych procesów. <p>Wynika stąd, że przy wymiarowaniu funkcjonalności biznesowej, nie identyfikuje się Plików danych reprezentujących m.in.:</p> <ul style="list-style-type: none"> dane uwierzytelniające użytkownika, uprawnienia użytkownika, logi systemowe, parametry biznesowe, słowniki, szablony dokumentów, symbolizację warstw GIS, reguły przetwarzania danych, <p>oraz w ramach Elementarnych procesów nie zlicza się jako funkcji dostępu do Plików danych m.in.:</p> <ul style="list-style-type: none"> operacji uwierzytelniania oraz autoryzacji użytkowników, odczytu i zapisu do logów systemowych, odczytu parametrów biznesowych, odczytu zawartości słowników, odczytu szablonów dokumentów, odczytu symbolizacji warstw GIS, odczytu reguł przetwarzania danych, <p>Powyższe dane mogą być identyfikowane jako Pliki danych z punktu widzenia wymiarowania funkcjonalności bazowej, o ile zdefiniowano wymagania dotyczące funkcjonalności bazowej.</p>
Przykład	<p>A. Wymaganie definiuje funkcjonalność rejestracji dokumentów, która wykorzystuje słownik prosty „Typy dokumentów”. Słownik ten jest elementem parametryzacji. Słownik ten nie jest identyfikowany jako Plik danych, nie jest uwzględniany w wymiarowaniu tej funkcjonalności. Przy wymiarowaniu Elementarnego procesu reprezentującego rejestrację dokumentu nie zlicza się jako dostępu do Pliku danych (FTR) odczytu tego słownika.</p> <p>B. Wymaganie definiuje funkcjonalność generowania decyzji zgodnie z określonym szablonem. Szablon decyzji jest elementem parametryzacji. Szablon ten nie jest nie jest identyfikowany jako Plik danych, nie jest uwzględniany w wymiarowaniu tej funkcjonalności. Przy wymiarowaniu Elementarnego procesu reprezentującego generowanie decyzji nie zlicza się jako dostępu do Pliku danych (FTR) odczytu tego szablonu.</p> <p>C. Wymaganie definiuje funkcjonalność wprowadzania lub modyfikacji danych dokumentów. Funkcjonalność ta odnotowuje każdy przypadek modyfikacji danych dokumentu w standardowym logu systemowym dla celów audytowych. Log systemowy nie jest identyfikowany jako Plik danych. Przy wymiarowaniu Elementarnego procesu reprezentującego</p>

	<p>wprowadzanie lub modyfikację danych dokumentu nie zlicza się jako dostępu do Pliku danych (FTR) zapisu do logu systemowego.</p> <p>D. Wymaganie definiuje funkcjonalność zarządzania słownikami prostymi (słowników o ustalonej, stałej strukturze). Funkcjonalność umożliwia utworzenie nowego słownika, a także zarządzanie zawartością istniejących słowników. Słownik prosty jest identyfikowany jako jeden Plik danych (nie ma tu znaczenia liczba słowników). Przy wymiarowaniu Elementarnych procesów służących definiowaniu nowych słowników, zliczane są dostępy do słownika prostego, jako dostępu do Pliku danych.</p> <p>E. Wymaganie opisuje funkcjonalność zarządzania słownikiem jednostek organizacyjnych oraz instrumentów finansowych. Słownik jednostek organizacyjnych definiuje zależności pomiędzy jednostkami organizacyjnymi. Słownik instrumentów opisany jest przy pomocy specyficznego zbioru atrybutów, które określają między innymi rodzaj i wielkość pomocy finansowej. W przypadku jednostek organizacyjnych funkcjonalność ma umożliwiać definiowanie, usuwanie i modyfikację jednostek oraz zależności między nimi. W przypadku instrumentów finansowych funkcjonalność ma umożliwiać tworzenie i wersjonowanie nowych instrumentów. Z punktu widzenia funkcjonalności bazowej zostaną zidentyfikowane dwa odrębne Pliki danych, jeden reprezentujący jednostki organizacyjne, drugi instrumenty finansowe. Przy wymiarowaniu Elementarnych procesów służących do zarządzania tymi słownikami, zliczane są odwołania do jednostek organizacyjnych jak i instrumentów finansowych.</p> <p>F. Wymaganie opisuje funkcjonalność przeglądu logu systemowego. Z punktu widzenia tej funkcjonalności bazowej identyfikowany jest Plik danych reprezentujący log systemowy. Przy wymiarowaniu Elementarnego procesu przeglądu logu systemowego zliczane są odwołania do logu systemowego.</p>
--	---

ID reguły	IFPUG010: Wymiarowanie zmiany zakresu parametryzacji funkcjonalności biznesowej
Opis reguły	<p>Zmiana zakresu parametryzacji funkcjonalności biznesowej czyli:</p> <ul style="list-style-type: none"> • dodanie lub usunięcie parametrów biznesowych, • dodanie lub usunięcie słownika, • dodanie lub usunięcie szablonu dokumentu, • zmiana zakresu szablonu dokumentu, <p>wymiarowane są jako zmiany Elementarnych Procesów funkcjonalności biznesowej, wynikające z wprowadzenia parametryzacji.</p>
Przykład	

ID reguły	IFPUG012: Interfejs użytkownika - enumeracje
Opis reguły	Jeśli wartość atrybutu w procesie jest nadawana poprzez wybór wartości ze stałej listy (enumeratora), to taki atrybut, razem z jego wyborem, jest zliczany jako 1 DET.
Przykład	Atrybut wprowadzenia miesiąca ważności karty kredytowej jest modelowany jako 1 DET w elementarnym procesie zarówno, gdy edycja jest zrealizowana jako wprowadzenie numeru miesiąca, jak i wybór miesiąca z rozwijalnej listy.

ID reguły	IFPUG013: Interfejs użytkownika - akcje
Opis reguły	<p>Jako atrybuty elementarnych procesów zlicza się możliwe akcje uruchamiające inne elementarne procesy lub finalizujące dany elementarny proces. Zlicza się zawsze jeden DET dla danej akcji niezależnie od liczby sposobów jej wywołania. Jako DET nie zlicza się akcji związanych z nawigowaniem po interfejsie, krokach kreatora, dostępu do pomocy, akcji anulowania, o ile nie wiąże się z ona z elementarnym procesem realizującym to anulowanie a jest tylko wycofaniem transakcji / zamknięciem okna.</p> <p>W przypadku zmian wizualnych /układów kontrolek interfejsu użytkownika jako zmienione traktuje się wszystkie atrybuty wynikające z możliwości uruchamiania akcji w ramach danego procesu elementarnego.</p> <p>Metodą IFPUG nie szacuje się konstrukcji ani modyfikacji struktury menu aplikacji.</p>
Przykład	W ramach prezentacji listy zarejestrowanych w systemie dokumentów (EQ Wyszukanie dokumentów) możliwe jest uruchomienie EQ Przegląd dokumentu poprzez dwuklik na liście dokumentów lub wybranie opcji „Przeglądaj” dla wybranego dokumentu. Dla tej akcji liczony jest jeden atrybut elementarnego procesu.

ID reguły	IFPUG014: Identyfikacja atrybutów Elementarnych Procesów – niezależność od atrybutów Plików danych
Opis reguły	<p>Atrybutami Elementarnych procesów (DET) są jedynie atrybuty danych przekraczające granicę oprogramowania.</p> <p>Liczba atrybutów Elementarnych procesów jest niezależna od liczby atrybutów odczytywanych lub zapisywanych z/do Plików danych, z których korzysta Elementarny proces.</p> <p>Jeżeli w ramach Elementarnego procesu wprowadzany jest jeden atrybut danych (DET), to bez względu na to w ilu atrybutach Plików danych, czy w ilu Plikach danych zostanie on zapisany w ramach tego Elementarnego procesu zliczany jest tylko jeden atrybut (DET) dla danego Elementarnego procesu.</p>

	<p>Jeżeli w ramach Elementarnego procesu prezentowane są użytkownikowi obiekty o jednorodnej strukturze, to bez względu na to z ilu atrybutów Plików danych, czy z ilu Plików danych zostały odczytane dane tych obiektów, jako atrybuty Elementarnego procesu zliczane są tylko atrybuty obiektu prezentowane użytkownikowi.</p> <p>Jeżeli w ramach Elementarnego procesu prezentowany jest pojedynczy atrybut, to bez względu na to z ilu atrybutów Plików danych, czy z ilu Plików danych zostały odczytane dane potrzebne do wyznaczenia wartości tego atrybutu, zliczany jest tylko jeden atrybut (DET) dla danego Elementarnego procesu.</p>
Przykład	<p>A. Funkcjonalność umożliwia wprowadzanie identyfikatora dokumentu. System wyodrębnia z identyfikatora sygnaturę typu dokumentu. W zależności od typu dokumentu zapisuje identyfikator albo w Pliku danych reprezentującym wnioski, albo w Pliku danych reprezentującym decyzje. Typ dokumentu oraz identyfikator są zapisywane jako oddzielne atrybuty w obydwu Plikach danych. Elementarny proces odpowiedzialny za wprowadzanie identyfikatora posiada tylko jeden atrybut, bo tylko identyfikator dokumentu przekracza granicę oprogramowania.</p> <p>B. Wyniki wyszukiwania dokumentów mają postać tabeli zawierającej 3 kolumny: numer dokumentu, typ dokumentu oraz datę utworzenia. W celu wyszukania dokumentów system odczytuje dane z wielu Plików danych reprezentujących różne typy dokumentów. Numer dokumentu, typ dokumentu, a także data utworzenia odczytywane są więc z wielu atrybutów i wielu Plików danych. Elementarny proces odpowiedzialny za wyszukiwanie dokumentów będzie posiadał tylko 3 atrybuty (DETy) prezentowane użytkownikowi, bez względu na to ile odczytanych zostanie Plików danych oraz atrybutów w procesie wyszukiwania.</p> <p>C. Funkcjonalność wyznacza i prezentuje użytkownikowi sumę kwot przyznanych płatności dla danego beneficjenta, wynikającą z różnych decyzji. System musi odczytać dane z 3 Plików danych, reprezentujących różne typy decyzji, ponadto kwoty przyznanych płatności na decyzjach rozbite są na składniki, zapisane w różnych atrybutach. Elementarny proces odpowiedzialny za wyznaczenie i prezentację sumy kwot przyznanych płatności będzie posiadał pojedynczy atrybut (DET) reprezentujący sumę kwot przyznanych płatności, bez względu na to z ilu Plików danych oraz atrybutów będzie odczytywał dane potrzebne do wyznaczenia tej sumy.</p>

ID reguły	IFPUG101: Dodatkowy poziom złożoności Elementarnych procesów
Opis reguły	Dla bardzo złożonych Elementarnych procesów definiuje się nowy poziom złożoności o nazwie „COMPLEX” zgodnie z Tabela 5. Rozszerzona tabela wyliczania złożoności Elementarnych procesów
Przykład	Elementarny proces EI F1, w ramach którego granicę systemu przekracza 66 atrybutów i który korzysta z 3 plików danych ma złożoność 9 PF (jest typu Complex).

Tabela 5. Rozszerzona tabela wyliczania złożoności Elementarnych procesów

EI	DET 1 – 4	DET 5 – 15	DET 16 - 60	DET > 60
FTR < 2	LOW (3)	LOW (3)	AVERAGE (4)	HIGH (6)
FTR 2	LOW (3)	AVERAGE (4)	HIGH (6)	COMPLEX (9)
FTR > 2	AVERAGE (4)	HIGH (6)	HIGH (6)	COMPLEX (9)
EQ	DET 1 – 5	DET 6 – 19	DET 20 – 80	DET > 80
FTR < 2	LOW (3)	LOW (3)	AVERAGE (4)	HIGH (6)
FTR 2	LOW (3)	AVERAGE (4)	HIGH (6)	COMPLEX (9)
FTR > 2	AVERAGE (4)	HIGH (6)	HIGH (6)	COMPLEX (9)
EO	DET 1 - 5	DET 6 - 19	DET 20 – 80	DET > 80
FTR < 2	LOW (4)	LOW (4)	AVERAGE (5)	HIGH (7)
FTR 2 - 3	LOW (4)	AVERAGE (5)	HIGH (7)	COMPLEX (10)
FTR > 3	AVERAGE (5)	HIGH (7)	HIGH (7)	COMPLEX (10)

ID reguły	IFPUG301: Dekompozycja złożonych Elementarnych Procesów
Opis reguły	<p>Jeżeli w ramach Elementarnego Procesu (dalej Złożonego EP):</p> <ul style="list-style-type: none"> • Użytkownikowi funkcjonalnemu dostarczana jest informacja w postaci wielu Niezależnych grup atrybutów (DETów), • zidentyfikowano więcej niż 80 atrybutów (DET), • zidentyfikowano więcej niż 6 odwołań do Plików danych (FTR), <p>to funkcjonalność związana z wyznaczeniem wartości i dostarczeniem Użytkownikowi funkcjonalnemu pojedynczej Niezależnej grupy atrybutów w ramach tego Złożonego EP identyfikowana jest jako odrębny Elementarny proces (dalej Składowy EP) zgodnie z zasadami określonymi poniżej.</p>

	<p>Jeżeli atrybuty Złożonego EP spełniają poniższe warunki, to stanowią tę samą Niezależną grupę atrybutów (nie mogą należeć do odrębnych Niezależnych grup atrybutów):</p> <ul style="list-style-type: none"> • jednocześnie przekraczają granice oprogramowania w kierunku z oprogramowania na zewnątrz (wyjściowe jednocześnie są prezentowane lub kolejność prezentacji nie ma znaczenia z punktu widzenia wymagań) <p>oraz</p> <ul style="list-style-type: none"> • występuje między nimi dowolna z następujących relacji <ul style="list-style-type: none"> ○ opisują te same obiekty, ○ opisują obiekty tego samego typu, ○ opisują bezpośrednio powiązane ze sobą obiekty, ○ wartość jednego z atrybutów jest wynikiem przekształcenia wartości jednego lub wielu innych atrybutów Złożonego EP. <p>Przekształcenie wartości może być wynikiem rachunku logicznego lub arytmetycznego, może być wykonane z użyciem stałych wynikających z treści wymagania.</p> <p>Atrybuty Złożonego EP mogą należeć tylko do jednej Niezależnej grupy atrybutów.</p> <p>Przy identyfikacji atrybutów (DET) Złożonego EP, wyłączone są atrybuty wchodzące w skład Niezależnej grupy atrybutów, dla której zidentyfikowano odrębny Składowy EP.</p> <p>Przy identyfikacji odwołań do Plików danych (FTR) Złożonego EP, wyłączone są odwołania do Plików danych niezbędne jedynie do wyznaczenia wartości Niezależnej grupy atrybutów, dla której zidentyfikowano odrębny Składowy EP.</p> <p>Przy identyfikacji atrybutów (DET) Złożonego EP, wyłączone są atrybuty wejściowe, niezbędne jedynie do wyznaczenia wartości Niezależnej grupy atrybutów, dla której zidentyfikowano odrębny Składowy EP.</p> <p>Jeżeli po takiej operacji wyłączenia, Złożony EP nadal spełnia warunki wymienione na wstępie, to dozwolone jest wyodrębnienie kolejnego Składowego EP, na tych samych zasadach.</p> <p>Przy wymiarowaniu Składowych EP należy uwzględnić wszystkie odwołania do Plików danych niezbędne do wyznaczenia wartości atrybutów z Niezależnego podzbioru atrybutów. Odwołania te mogą się powtarzać, o ile wynika to z logiki aplikacji, w wielu Składowych EP wyodrębnionych z tego samego Złożonego EP, a także w samym Złożonym EP i Składowych EP z niego wyodrębnionych.</p> <p>Przy wymiarowaniu Składowych EP należy uwzględnić wszystkie atrybuty wejściowe Złożonego EP, które są niezbędne do wyznaczenia wartości atrybutów z Niezależnego podzbioru atrybutów. Atrybuty te mogą się powtarzać, o ile wynika to z logiki aplikacji, w wielu Składowych EP wyodrębnionych z tego samego Złożonego EP, a także w samym Złożonym EP i Składowych EP z niego wyodrębnionych.</p>
--	--

	<p>Składowy EP jest klasyfikowany jako EO.</p> <p>Nie można wyodrębnić Składowego EP jedynie w oparciu o część Niezależnego podzbioru atrybutów.</p>
Przykład	<p>Przykładem Złożonych EP są złożone, przekrojowe raporty zawierające części/sekcje które wyznaczają się niezależnie. Wygenerowanie części raportu, której dane spełniają kryteria Niezależnej grupy atrybutów identyfikowane jest jako Składowy EP.</p> <p>Przykładem Złożonych EP są procesy mające na celu złożone sprawdzenia danych, polegające na dostarczeniu wielu różnorodnych informacji związanych z danymi np. kontrola administracyjna. Wynik poszczególnych sprawdzeń (reguł) o ile stanowią one element procesu spełniającego kryteria Złożonego EP, może być traktowany jako Niezależna grupa atrybutów i jego uzyskanie może być identyfikowane jako Składowy EP.</p> <p>Jeżeli w ramach EP prezentowana jest tabela, której kolumny zawierają atrybuty dokumentów i powiązanych z nimi spraw, to stanowią one tę samą Niezależną grupę atrybutów, ponieważ opisują obiekty tego samego typu (kolejne wiersze tabeli) lub obiekty ze sobą powiązane (dokumenty dołączone do spraw).</p>

WYZNACZANIE ROZMIARU ZMIAN FUNKCJONALNOŚCI

ID reguły	IFPUG110: Wyliczanie współczynnika wpływu
Opis reguły	Przy wyznaczaniu współczynnika wpływu, stosowanego przy wyliczaniu rozmiaru funkcjonalnego zmian funkcjonalności, na podstawie procentowego wskaźnika zmiany atrybutów oraz procentowego wskaźnika zmiany plików stosuje się wartości z Tabela 6. Wyznaczanie współczynnika wpływu.
Przykład	<p>Przykład: Elementarny proces E1 korzysta z pliku danych ILF1 i wykorzystuje 10 atrybutów. Ma złożoność 3 PF. W wyniku zmiany, proces korzysta z dodatkowego pliku ILF2 oraz wykorzystuje kolejne dwa atrybuty. Procentowy wskaźnik zmiany atrybutów to $2/10 * 100\% = 20\%$. Procentowy wskaźnik zmiany plików to $1/1 * 100\% = 100\%$. Wskaźnik wpływu wynosi, zatem 0.5. Złożoność procesu po zmianie to 4 PF. Złożoność zmiany to $4 PF * 0.5 = 2PF$.</p>

Tabela 6. Wyznaczanie współczynnika wpływu

Procentowy wskaźnik zmiany atrybutu	pa ≤ 60 %	60 % < pa ≤ 100 %	pa > 100%
Procentowy wskaźnik zmiany plików			
pp ≤ 60 %	0.25	0.50	0.75
60% < pp ≤ 100 %	0.50	0.75	1.00
pp > 100%	0.75	1.00	1.25

ID reguły	IFPUG111: Wyliczanie rozmiaru zmian reguł przetwarzania danych
Opis reguły	<p>W przypadku, gdy zmiana dotyczy reguły walidacji danych w ramach Elementarnego procesu, to przez zmieniane atrybuty rozumie się, atrybuty Elementarnego procesu, których reguły walidacji uległy zmianie. Analogicznie, w przypadku odwołań do Plików danych, przez zmieniane pliki rozumiemy odwołania do Plików danych, dla których reguły walidacji uległy zmianie.</p> <p>W przypadku, gdy zmiana dotyczy reguły przetwarzania danych w ramach Elementarnego procesu, które prowadzą do uzyskania określonego rezultatu w postaci danych to przez zmienione atrybuty rozumie się, atrybuty Elementarnego procesu, które są wynikiem przetwarzania zgodnie ze daną regułą. Analogicznie w przypadku odwołań do Plików danych, przez zmieniane pliki rozumiemy odwołania do Plików danych w celu zapisania wyniku przetwarzania danych zgodnie ze zmienianą regułą.</p> <p>Jeżeli reguły przetwarzania są sterowane za pomocą parametrów biznesowych, które są zrozumiałe dla użytkownika, to zmiana reguł przetwarzania polegająca wyłącznie na zmianie wartości tych parametrów nie jest traktowana ani jako zmiana Elementarnych procesów, ani jako zmiana Plików danych. Zmiana taka nie jest traktowana jako zmiana funkcjonalności, jej rozmiar funkcjonalny jest równy 0.</p>
Przykłady	<p>Przykład: Zmiana funkcjonalności polega na zmianie wzoru, zgodnie z którym ma być wyliczana wysokość sankcji. Wysokość sankcji jest prezentowana po jej wyznaczeniu, jest zatem atrybutem Elementarnego procesu, który polega na wyznaczeniu wysokości sankcji. Wysokość sankcji zarówno przed zmianą jak i po zmianie wyznaczana jest na podstawie tych samych danych wniosku oraz danych wnioskodawcy. W przypadku tej zmiany zmianie ulega atrybut wysokość sankcji: liczba zmienianych atrybutów wynosi 1, natomiast nie zmieniają się odwołania do plików danych mające na celu pobranie danych z wniosku oraz wnioskodawcy: liczba zmienianych plików wynosi 0.</p>

	<p>Przykład: Zmiana funkcjonalności polega na zmianie wzoru, zgodnie z którym ma być wyliczana wysokość sankcji. Wysokość sankcji jest wyznaczana na podstawie wprowadzanej przez użytkownika daty, która jest atrybutem Elementarnego procesu, który polega na wyznaczeniu wysokości sankcji. Wielkość sankcji jest zapisywana. W przypadku tej zmiany, nie ulegają zmianie atrybuty Elementarnego Procesu: liczba zmienianych atrybutów wynosi 0, natomiast zmianie ulega odwołanie do Pliku danych, w którym zapisywana jest sankcja: liczba zmienianych plików wynosi 1.</p> <p>Przykład: Jeżeli w trakcie wprowadzania wniosku, sprawdzane jest, czy wiek beneficjenta przekracza określony parametrem biznesowy próg, np. 60 lat, to zmiana wartości tego parametru np. na 65 lat, która skutkuje zmianą reguły walidacji, nie jest traktowana jako zmiana Elementarnego procesu związanego z wprowadzaniem wniosku.</p>
--	--

ID reguły	IFPUG112: Dzielenie i łączenie Elementarnych procesów
Opis reguły	Jeżeli w wyniku zmiany, jeden Elementarny proces jest dzielony na wiele Elementarnych procesów lub odwrotnie, wiele Elementarnych procesów jest łączonych w jeden Elementarny proces, to zmiany te należy wymiarować, tak jak usunięcie starych (przed podzieleniem/połączeniem) i dodawanie nowych (po podzieleniu/połączeniu) Elementarnych procesów.

ID reguły	IFPUG113: Zmiana, a zastąpienie Elementarnego procesu
Opis reguły	Jeżeli rozmiar funkcjonalny Elementarnego procesu, jest większy niż rozmiar funkcjonalny usunięcia Elementarnego procesu przed zmianą i dodaniu nowego Elementarnego procesu po zmianie to przyjmujemy, że zmiana polega na zastąpieniu Elementarnego procesu przed zmianą, Elementarnym procesem po zmianie.

STOSOWANIE PF IFPUG DLA INTERFEJSÓW I USŁUG

ID reguły	IFPUG201: Szacowanie interfejsów i usług
Opis reguły	Wszelkie interfejsy oraz usługi udostępniane na potrzeby komunikacji pomiędzy modułami Aplikacji lub na potrzeby systemów zewnętrznych są modelowane jako Elementarne procesy. Zasady identyfikacji tych procesów abstrahują od implementacji np. nie ma znaczenia czy są realizowane w postaci widoków bazodanowych czy udostępniane na szynie usług.

	<p>Jeżeli Elementarny proces, wywołuje usługę lub interfejs udostępniany przez inny moduł oprogramowania, oprogramowanie zewnętrzne lub udostępnioną na szynie danych (bez względu na to, czy sama aplikacja dostarcza tej usługi na szynę danych, czy nie), to przyjmuje się, że proces korzysta z EIF. Przy wymiarowaniu tego EIF uwzględnia się atrybuty i rekordy zwracane przez usługę. Jeżeli usługa zwraca jedynie status wykonania, to liczony jest 1 DET.</p> <p>Jeżeli funkcjonalność oprogramowania realizowana jest jako formularz (interfejs użytkownika), który wywołuje określoną usługę oraz zarówno formularz jak i usługa znajdują się wewnątrz granicy oprogramowania, a do komunikacji między nimi nie służy żaden zewnętrzny komponent (znajdujący się poza granicą oprogramowania komponent) np. zewnętrzna względem modułu szyna danych, to formularz i usługę traktuje się jako tę samą funkcjonalność, nie definiuje się dla nich odrębnych Elementarnych procesów.</p>
Przykład	<ol style="list-style-type: none"> 1. Aplikacja Web umożliwia wprowadzanie danych za pomocą formularza: F1. Dane z formularza są zapisywane w systemie za pomocą usługi U1 zachowującej formularz elektroniczny. Formularz F1 i usługa U1 znajdują się wewnątrz tej samej granicy oprogramowania. Usługa U1 nie jest udostępniana innym systemom na szynie danych, jej implementacja jest wynikiem decyzji dotyczącej technologii realizacji funkcjonalności wprowadzania danych. Dla oszacowania tej funkcjonalności należy zdefiniować jeden proces EI Wprowadzanie danych. Proces ten obejmuje wszystkie atrybuty EI oraz korzysta z wszystkich Plików danych, z których korzysta usługa U1. Nie jest definiowany odrębny Elementarny proces reprezentujący usługę U1. 2. Aplikacja A udostępnia na szynie danych usługę, która udostępnia nazwę oraz nr NIP podmiotu. Podmiot zawiera dodatkowo dane dotyczące roku utworzenia. Z usługi tej korzysta aplikacja B do generowania raportu zawierającego dane podmiotu. Dla aplikacji A: <ul style="list-style-type: none"> • identyfikowany jest wewnętrzny plik danych (ILF) Dane podmiotu, który będzie posiadał 3 DETy: nazwę, NIP oraz rok utworzenia, • identyfikowany jest elementarny proces (EQ) Udostępnianie danych podmiotu, który będzie korzystał z w/w pliku (ILF) Dane podmiotu Dla aplikacji B: <ul style="list-style-type: none"> • identyfikowany jest zewnętrzny plik danych (EIF) Dane podmiotu, który będzie posiadał 2 DETy: nazwę, NIP • identyfikowany jest Elementarny proces generowania raportu (EO), który będzie korzystał z w/w EIF Dane podmiotu.

STOSOWANIE PF IFPUG DLA PROCESÓW WSADOWYCH

ID reguły	IFPUG501: Szacowanie procesów wsadowych
Opis reguły	<p>W przypadku szacowania procesów wsadowych, w ramach których wynik działania tego procesu (rozumiany jako cel biznesowy) nie przekracza granicy systemu przyjmuje się, iż wszystkie atrybuty aktualizowane jako element realizacji celu biznesowego, traktuje się jako atrybuty Elementarnego procesu reprezentującego proces wsadowy.</p> <p>Elementarne procesy odpowiadające procesom wsadowym klasyfikuje się jako EI.</p>
Przykład	<p>Proces wsadowy, P1 ma za zadanie ustawić jako „nowy” status wszystkich obiektów klasy A, które na koniec dnia roboczego nie mają statusu „zatwierdzone”. Jako efekt działania procesu ma być zaprezentowana liczba zmodyfikowanych obiektów. Dla każdego aktualizowanego obiektu proces ma utworzyć operację klasy B zawierającą informacje o zmianie statusu zawierającą alfanumeryczną kopię danych obiektu A. Zmiana statusu danych obiektu A na „nowy” powoduje ustalenie niektórym jego atrybutom wartości domyślnych.</p> <p>Z punktu widzenia szacowania, w powyższym procesie EI jedynym atrybutem przekraczającym granicę systemu jest liczba zmodyfikowanych obiektów. Jednakże uwzględniając regułę IFPUG501 jako atrybuty procesu należy uznać również resetowane atrybuty obiektu A oraz atrybut jego statusu, jak i atrybuty obiektu tworzonej operacji.</p>

STOSOWANIE PF IFPUG DLA FUNKCJONALNOŚCI PRZYGOTOWANIA DANYCH

ID reguły	IFPUG601: Przygotowanie danych: Identyfikacja Plików danych
Opis reguły	<p>Dane, które mają być utworzone w wyniku funkcjonalności przygotowania danych, identyfikuje się jako Wewnętrzne pliki danych (ILF).</p> <p>Dane źródłowe, które są przekształcane w celu przygotowania danych, identyfikuje się jako Zewnętrzne pliki danych (EIF).</p>
Przykłady	<p>A. Wymaganie dotyczy migracji danych podmiotów z systemu A do systemu B. Aby utworzyć kompletne dane podmiotu w systemie B należy je pobrać zarówno z danych podmiotów, jak i wniosków o płatność składanych przez te podmioty w systemie A. Podmiot w systemie B identyfikowany jest jako pojedynczy ILF, Podmiot oraz wniosek o płatność w systemie A identyfikowane są jako dwa pliki EIF.</p> <p>B. Wymaganie dotyczy migracji danych podmiotów z systemu A do systemu B. Dane podmiotu w systemie B są pobierane albo z danych podmiotu w systemie A albo z danych wniosków o rejestrację tych podmiotów w systemie A.</p>

	<p>Podmiot w systemie B identyfikowany jest jako pojedynczy ILF, Podmiot oraz wniosek o rejestrację w systemie A identyfikowane są jako dwa pliki EIF.</p> <p>C. Wymaganie dotyczy migracji danych podmiotu i wniosków o rejestrację z systemu A do systemu B. Dane podmiotu w systemie B pobierane są z danych wniosków o rejestrację tych podmiotów w systemie A. Dane wniosków o rejestrację w systemie B pobierane są także z danych wniosków o rejestrację w systemie A. Podmiot w systemie B oraz wniosek o rejestrację w systemie B identyfikowane są jako dwa odrębne ILF, wniosek o rejestrację w systemie A identyfikowany jest jako pojedynczy EIF</p> <p>D. Wymaganie dotyczy inicjalizacji danych słownika jednostek terytorialnych. Słownik jednostek terytorialnych identyfikowany jest jako pojedynczy ILF.</p>
--	---

ID reguły	IFPUG602: Przygotowanie danych: Identyfikacja Elementarnych procesów
Opis reguły	<p>Dla każdego ILF reprezentującego dane, które mają być utworzone oraz każdego unikatowego procesu polegającego na utworzeniu kompletnych danych dla danego ILF na podstawie rozłącznego zbioru danych źródłowych identyfikuje się jeden odrębny Elementarny proces typu EI.</p> <p>Konsekwencją tej zasady są wymienione niżej podreguły.</p> <ol style="list-style-type: none"> 1. Dla każdego ILF reprezentującego dane, które mają być utworzone identyfikuje się co najmniej jeden Elementarny proces typu EI, 2. W przypadku funkcjonalności generowania danych, dla każdego ILF reprezentującego dane, które mają być utworzone, identyfikuje się dokładnie jeden Elementarny proces typu EO,
Przykłady	<p>A. Wymaganie dotyczy migracji danych podmiotów z systemu A do systemu B. Aby utworzyć kompletne dane podmiotu w systemie B należy je pobrać zarówno z danych podmiotów, jak i wniosków o płatność składanych przez te podmioty w systemie A. Dla procesu przekształcenia danych wniosku oraz podmiotu z systemu A na dane podmiotu w systemie B (zidentyfikowanego jako pojedynczy ILF) identyfikowany jest jeden Elementarny proces typu EI.</p> <p>B. Wymaganie dotyczy migracji danych podmiotów z systemu A do systemu B. Dane podmiotu w systemie B są pobierane albo z danych podmiotu w systemie A albo z danych wniosków o rejestrację tych podmiotów w systemie A. Dane podmiotu oraz dane wniosku o rejestrację w systemie A to rozłączne zbiory danych. Dane każdego z nich są wystarczające do przygotowania kompletnych danych podmiotu w systemie B.</p>

	<p>Dla procesu przekształcenia danych podmiotu z systemu A na dane podmiotu w systemie B oraz dla procesu przekształcenia danych wniosku o rejestrację z podmiotu A na dane podmiotu w systemie B identyfikowane są dwa odrębne Elementarne procesy typu EI.</p> <p>C. Wymaganie dotyczy migracji danych podmiotu i wniosków o rejestrację z systemu A do systemu B. Dane podmiotu w systemie B pobierane są z danych wniosków o rejestrację tych podmiotów w systemie A. Dane wniosków o rejestrację w systemie B pobierane są także z danych wniosków o rejestrację w systemie A.</p> <p>D. Dla procesu przekształcenia danych wniosku o rejestrację z systemu A na dane podmiotu w systemie B oraz dla procesu przekształcenia danych wniosku o rejestrację z podmiotu A na dane wniosku o rejestrację w systemie B identyfikowane są dwa odrębne Elementarne procesy typu EI.</p> <p>E. Wymaganie dotyczy inicjalizacji danych słownika jednostek terytorialnych.</p> <p>Dla procesu inicjalizacji danych słownika jednostek terytorialnych, identyfikowany jest jeden Elementarny proces typu EI.</p>
--	--

ID reguły	IFPUG603: Przygotowanie danych: Liczba FTR dla Elementarnych procesów
Opis reguły	<p>Liczbę FTR dla Elementarnego procesu określa się wyłącznie na podstawie liczby EIF reprezentujących dane źródłowe używane przez ten proces oraz liczby ILF reprezentujących dane przygotowane w ramach danego Elementarnego procesu (która zgodnie z IFPUG602 zawsze wynosi 1). Natomiast przy określaniu liczby FTR dla Elementarnego procesu nie są brane pod uwagę ewentualne dane pośrednie lub dane przygotowane w ramach innych Elementarnych procesów.</p> <p>Konsekwencją tej zasady są wymienione niżej podreguły.</p> <ol style="list-style-type: none"> 1. Liczba FTR to liczba EIF reprezentujących dane źródłowe używane przez ten proces powiększone o 1. 2. W przypadku funkcjonalności generowania danych, liczba FTR dla Elementarnych procesów zawsze wynosi 1.
Przykłady	<p>A. Wymaganie dotyczy migracji danych podmiotów z systemu A do systemu B. Aby utworzyć kompletne dane podmiotu w systemie B należy je pobrać zarówno z danych podmiotów, jak i wniosków o płatność składanych przez te podmioty w systemie A. Dla Elementarnego procesu reprezentującego przekształcenie danych wniosku oraz podmiotu z systemu A na dane podmiotu w systemie B liczba FTR wynosi 3.</p> <p>B. Wymaganie dotyczy migracji danych podmiotów z systemu A do systemu B. Dane podmiotu w systemie B są pobierane albo z danych podmiotu w systemie A albo z danych wniosków o rejestrację tych podmiotów w systemie A.</p>

	<p>Dane podmiotu oraz dane wniosku o rejestrację w systemie A to rozłączne zbiory danych. Dane każdego z nich są wystarczające do przygotowania kompletnych danych podmiotu w systemie B. Zarówno dla Elementarnego procesu przekształcenia danych podmiotu z systemu A na dane podmiotu w systemie B jak i dla procesu przekształcenia danych wniosku o rejestrację z podmiotu A na dane podmiotu w systemie B liczba FTR wynosi 2.</p> <p>C. Wymaganie dotyczy migracji danych podmiotu i wniosków o rejestrację z systemu A do systemu B. Dane podmiotu w systemie B pobierane są z danych wniosków o rejestrację tych podmiotów w systemie A. Dane wniosków o rejestrację w systemie B pobierane są także z danych wniosków o rejestrację w systemie A. Zarówno dla Elementarnego procesu przekształcenia danych wniosku o rejestrację z systemu A na dane podmiotu w systemie B jak i dla Elementarnego procesu przekształcenia danych wniosku o rejestrację z podmiotu A na dane wniosku o rejestrację w systemie B liczba FTR wynosi 2.</p> <p>D. Wymaganie dotyczy inicjalizacji danych słownika jednostek terytorialnych. Dla Elementarnego procesu inicjalizacji danych słownika jednostek terytorialnych, liczba FTR wynosi 1.</p>
--	---

ID reguły	IFPUG604: Przygotowanie danych: Liczba DET dla Elementarnych procesów
Opis reguły	Jako liczbę DET, która przekracza granice oprogramowania dla Elementarnego procesu przyjmuje się liczbę DET pliku ILF reprezentującego dane przygotowane w ramach danego Elementarnego procesu.
Przykłady	<p>A. Wymaganie dotyczy migracji danych podmiotów z systemu A do systemu B. Aby utworzyć kompletne dane podmiotu w systemie B należy je pobrać zarówno z danych podmiotów, jak i wniosków o płatność składanych przez te podmioty w systemie A. Dane podmiotu w systemie B składają się z 12 atrybutów. Dla Elementarnego procesu reprezentującego przekształcenie danych wniosku oraz podmiotu z systemu A na dane podmiotu w systemie B liczba DET przekraczających wynosi 12.</p> <p>B. Wymaganie dotyczy inicjalizacji danych słownika jednostek terytorialnych. Dane słownika jednostek terytorialnych opisane są przy pomocy 3 atrybutów. Dla Elementarnego procesu inicjalizacji danych słownika jednostek terytorialnych, liczba DET wynosi 3.</p>

STOSOWANIE PF IFPUG DLA FUNKCJONALNOŚCI BAZOWYCH

ID reguły	IFPUG701: Funkcjonalność bazowa: Odrębność funkcjonalności bazowej od funkcjonalności biznesowej
Opis reguły	<p>Funkcjonalność bazowa definiowana jest na innym poziomie abstrakcji i z użyciem innych pojęć niż funkcjonalność biznesowa.</p> <p>Jeżeli te same dane są przetwarzane zarówno w kontekście funkcjonalności bazowej jak i biznesowej, to są definiowane są odrębnie Pliki danych dla funkcjonalności bazowej i biznesowej.</p> <p>Te same Pliki danych nie mogą być współdzielone przez funkcjonalność bazową oraz funkcjonalność biznesową. Elementarne procesy zidentyfikowane dla funkcjonalności bazowej nie mogą korzystać z Plików danych zidentyfikowanych dla funkcjonalności biznesowej i odwrotnie.</p>
Przykłady	<p>A. W ramach realizacji wymagania należy opracować funkcjonalność stronicowania wyników wyszukiwania. W aplikacji występuje funkcjonalność wyszukiwania podmiotów oraz wniosków. Funkcjonalność stronicowania jest funkcjonalnością bazową, może być użyta przez dowolną funkcjonalność wyszukiwania. W wyniku wymagania zostanie utworzona funkcjonalność stronicowania, która zostanie zwymiarowana niezależnie od funkcjonalności biznesowej wyszukiwania podmiotów oraz wniosków. W wyniku wymagania zostanie także zmodyfikowana funkcjonalność wyszukiwania podmiotów oraz wniosków, polegająca na użyciu nowego mechanizmu prezentacji wyników wyszukiwania. Zmiana tej funkcjonalności także zostanie zwymiarowana niezależnie od funkcjonalności bazowej stronicowania.</p>

ID reguły	IFPUG702: Funkcjonalność bazowa: Funkcjonalność wywoływana
Opis reguły	<p>Jeżeli funkcjonalność bazowa jest wywoływana przez funkcjonalność biznesową, to Użytkownikiem funkcjonalnym jest warstwa funkcjonalności biznesowej, a dane, które przekraczają granicę oprogramowania, to parametry wejściowe i wyjściowe funkcjonalności bazowej.</p> <p>W tym przypadku przyjmuje się, że funkcjonalność bazowa ma co najmniej jeden parametr wejściowy. Jeżeli do funkcji nie jest przekazywana żadna wartość, to przyjmuje się, że parametrem wejściowym jest sygnał uruchomienia danej funkcjonalności.</p>
Przykłady	<p>A. Celem wymagania jest udostępnienie mechanizmu SSO pomiędzy aplikacjami A i B w oparciu o mechanizm uwierzytelnienia aplikacji A. Użytkownicy w aplikacji B identyfikowani są za pomocą loginu, a uwierzytelniani przy pomocy hasła.</p>

	<p>Konieczne jest zbudowanie lub modyfikacja funkcji uwierzytelniania w aplikacji B, która będzie korzystała z mechanizmu uwierzytelniania aplikacji A (np. usługi udostępnianej przez aplikację A). Z funkcjonalnością uwierzytelniania aplikacji B wiąże się Elementarny proces uwierzytelniania, który może być wywoływany przez dowolną funkcjonalność, która będzie potrzebowała potwierdzenia autentyczności danego użytkownika, a także zewnętrzny plik danych reprezentujący mechanizm uwierzytelniania aplikacji A (EIF). Dane które przekraczają w tym wypadku granice oprogramowania, to login i hasło (2 DET), które są parametrami, z którymi musi być wywołana funkcja uwierzytelniania, oraz potwierdzenie autentykacji zwracane przez tę funkcję (1 DET). Elementarny proces uwierzytelniania, będzie korzystał z zewnętrznego mechanizmu (EIF), nie utrwała i dodatkowo nie przetwarza danych, zwraca dane bezpośrednio odczytane z zewnętrznej usługi aplikacji A. W rezultacie Elementarny proces odpowiadający nowej funkcji uwierzytelniania to EQ, które posiada 3 DET oraz korzysta z 1 FTR. Jego rozmiar funkcjonalny wynosi LOW(3). Funkcjonalność uwierzytelniania wymiarowana jest tylko raz, niezależnie od tego ile razy i przez jakie funkcjonalności biznesowe jest wywoływana w systemie.</p> <p>B. Wymaganie wprowadza nowy mechanizm rozkładania zadań generowania załączników graficznych na serwerach wydruku. W tym przypadku, funkcjonalność biznesowa generowania załącznika graficznego wywołuje bazową funkcjonalność, którą celem jest skierowanie zadania generowania na odpowiedni serwer wydruku (dodanie go do kolejki zadań odpowiedniego serwera wydruku). Zakładamy, że jedynym kryterium rozkładania zadań na serwerach jest wielkość kolejki zadań na poszczególnych serwerach wydruku. Parametrem wejściowym funkcjonalności jest identyfikator zadania generowania załącznika graficznego (1 DET). Z funkcjonalnością wiąże się zatem Elementarny proces rozkładania zadań na serwerach wydruku. Proces ten to EI, bowiem w wyniku jego wykonania zostanie dodane zadanie do kolejki Sewera wydruku (nastąpi modyfikacja kolejki). Elementarny proces nie korzysta z innych Plików danych (1 FTR) . Z funkcjonalnością wiąże się, wspomniany wyżej Plik danych (ILF) reprezentujący kolejkę zadań na serwerze wydruku.</p>
--	---

ID reguły	IFPUG703: Funkcjonalność bazowa: Funkcjonalność określająca sposób interakcji i przetwarzania danych
O pis reguły	<p>Jeżeli funkcjonalność bazowa określa sposób interakcji i przetwarzania danych funkcjonalności biznesowej, to Użytkownikiem funkcjonalnym jest Użytkownik funkcjonalny funkcjonalności biznesowej.</p> <p>W tym przypadku dane jakie przekraczają granice oprogramowania, to:</p>

	<ul style="list-style-type: none"> • dane przekraczające granicę oprogramowania wyłącznie na potrzeby danej funkcjonalności bazowej (mające sens tylko w kontekście danej funkcjonalności bazowej, nie mające znaczenia w przypadku funkcjonalności biznesowej), np. informacje wyświetlane na potrzeby określonego typu interakcji • dane przekraczające granicę oprogramowania przekształcane przez funkcjonalność bazową, np. formatowane, szyfrowane, itp. • dane służące do ograniczenia, podzielenia lub zmiany uporządkowania zbiorów danych, które przekraczają granicę oprogramowania, np. kryteria sortowania i filtrowania <p>W przypadku tej funkcjonalności odwołania do Plików danych, to:</p> <ul style="list-style-type: none"> • odwołania do Plików danych, wyłącznie na potrzeby danej funkcjonalności bazowej (mające sens tylko w kontekście danej funkcjonalności bazowej, nie mające znaczenia w przypadku funkcjonalności bazowej), • odwołania do Plików danych, które są modyfikowane przez funkcjonalność bazową.
Przykłady	<p>A. W ramach realizacji wymagania należy opracować funkcjonalność stronicowania wyników wyszukiwania. W aplikacji występuje funkcjonalność wyszukiwania podmiotów oraz wniosków. W systemie zdefiniowany jest parametr systemowy, reprezentujący domyślną wielkość strony.</p> <p>Funkcjonalność stronicowania będzie modyfikowała sposób interakcji i przetwarzania danych funkcjonalności wyszukiwania podmiotów i wniosków.</p> <p>Użytkownikiem funkcjonalnym stronicowania jest użytkownik funkcjonalny wyszukiwania podmiotów i wniosków. Z funkcjonalnością stronicowania wiąże się Elementarny proces stronicowania typu EO (dane po przekształceniu są wysyłane z systemu poza granicę oprogramowania) oraz Plik danych reprezentujący dowolne wyniki wyszukiwania. Parametr systemowy jest przechowywany w Pliku danych reprezentującym wszystkie proste parametry systemowe (zakładamy, że istnieje funkcjonalność bazowa modyfikacji parametrów), zatem przy okazji funkcjonalności stronicowania nie jest dodatkowo identyfikowany.</p> <p>Wyłącznie na potrzeby stronicowania granicę oprogramowania przekraczają dane dotyczące: wielkości strony, numeru strony jaka ma być wyświetlona, liczby stron, bieżącego numeru strony, akcji poprzedni i następny (6 DET).</p> <p>Zbiór danych, który przekracza granice oprogramowania i jest ograniczany do określonej strony przez funkcjonalność bazową stronicowania to wyniki wyszukiwania. Kryterium ograniczającym zbiór wyników jest numer kolejny strony oraz wielkość strony atrybuty, które zostały już uwzględnione wyżej.</p>

	W sumie Elementarny proces posiada więc 6 atrybutów (DET). Funkcjonalność bazowa odwołuje się do wyników wyszukiwania oraz do parametru, który określa domyślną wielkość strony (w sumie 2 FTR). W rezultacie rozmiar Elementarnego procesu reprezentującego wynosi funkcjonalność stronicowania wynosi AVERAGE (5).
--	--

ZASADY DOKUMENTOWANIA ROZMIARU FUNKcjONALNEGO OPROGRAMOWANIA / ZMIAN OPROGRAMOWANIA

ZASADY TWORZENIA I UTRZYMYWANIA MODELU ARTEFAKTÓW PF IFPUG

Każda aplikacja, zdefiniowana przez granice oprogramowania, posiada własny model IFPUG w ramach tego projektu. Pojedynczy model IFPUG powinien być utrzymywany jako jeden projekt EA. Struktura tego modelu powinna odzwierciedlać logiczny model funkcji i danych aplikacji.

Pliki danych są modelowane jako zbiór klas powiązanych ze sobą relacją kompozycji. Rekord nadrzędny Pliku danych jest modelowany przy pomocy klasy ze stereotypem «ILF» lub «EIF» (w zależności od typu Pliku danych), nazwa klasy powinna odzwierciedlać nazwę zidentyfikowanego Pliku danych. Pozostałe rekordy Plików danych są modelowane jako klasy w relacji kompozycji z klasą główną. Nazwy tych klas składają się z dwóch części: nazwy Pliku danych, nazwy rekordu oddzielonych znakiem „_”: <nazwa klas głównej>_<nazwa rekordu>. Relacje pomiędzy Plikami danych są modelowane, jako asocjacje klas ze wskazaniem nawigowalności. Atrybuty Pliku danych (DETy) są modelowane jako atrybuty klas, stanowiących danych Plik danych, bez określania ich typów oraz opisów chyba, że opis jest niezbędny by prawidłowo zidentyfikować dany atrybut. W przypadku Plików danych zawierających duże ilości DETów, atrybuty klas mogą reprezentować grupy atrybutów, lecz w takim przypadku, określana jest liczba DETów reprezentowana przez dany atrybut klasy poprzez określenie liczebności atrybutu (pola „Lower bound” i „Upper bound” opisu atrybutu). Atrybuty związków pomiędzy plikami danych nie są modelowane dodatkowo – są odzwierciedlone poprzez asocjacje klas reprezentujących pliki danych i ich rekordy.

Elementarne procesy są modelowane jako przypadki użycia ze stereotypami «EI», «EQ» oraz «EO». Wykorzystanie Plików danych przez Elementarne procesy jest modelowane jako relacja «dependency» z przypadku użycia reprezentującego Elementarny proces do klasy nadrzędnej danego Pliku danych. Atrybuty wykorzystywane przez Elementarne procesy modelowane są jako atrybuty klasy ze stereotypem «EP», powiązanej z przypadkiem użycia modelującym dany Elementarny proces. Związek pomiędzy przypadkiem użycia modelującym Elementarny proces a klasą modelującą atrybuty Elementarnego procesu modelowany jest przy pomocy relacji «dependency» z przypadku użycia reprezentującego Elementarny proces do klasy modelującej atrybuty, nazwa przypadku użycia i powiązanej z nim klasy musi być taka sama. Jeżeli atrybut wykorzystywany przez Elementarny proces jest także atrybutem Pliku danych, z którego korzysta Elementarny proces, to nazwa tego atrybutu musi być taka sama jak nazwa atrybutu Pliku danych oraz być poprzedzona nazwą klasy Pliku danych.

W przypadku wymiarowania zmian model IFPUG musi zawierać:

- elementy dodawane,

- elementy usuwane,
- elementy zmieniane, przy czym klasy modelujące Pliki danych i zbiory atrybutów Procesów elementarnych, powinny występować w modelu w wersji sprzed zmiany i po zmianie.

Model IFPUG dla zmian, musi posiadać określoną wersję i posiadać informację, na podstawie jakiej wersji modelu IFPUG, obrazującego oprogramowania przed zmianą, został wykonany.

Dodawane elementy modelu (klasy, przypadki użycia, atrybuty, relacje «dependency») oznaczane są tagiem (zakładka „tagged value”) „DODANE”

Usunięte elementy modelu oznaczane są tagiem „USUNIETE”.

Zmienione elementy modelu sprzed zmiany oznaczane są tagiem „PRZED ZMIANA”, a po zmianie tagiem „PO ZMIANIE”, przy czym przypadki użycia reprezentujące zmienione Elementarne procesy oznaczane są tagiem „ZMIENIONE”.

Relację pomiędzy klasą reprezentującą element zmieniany (przed zmianą), a zmieniony (po zmianie) oznaczamy relacją «trace».

Pliki danych (tylko klasa nadrzędna) oraz Elementarne procesy, elementy zmienione (w wersji po zmianie) oznaczana jest liczba dodawanych, zmienianych, usuwanych lub wykorzystywanych przez zmienioną logikę biznesową atrybutów tagiem „ZMIANA_ATRYBUTY”.

Atrybuty Elementarnych procesów po zmianie, które wykorzystywane są przez zmienioną logikę biznesową oznaczane są tagiem „LOGIKA”.

Każdy Elementarny proces i Plik danych posiada swój unikalny niezmienny identyfikator postaci xy[-zzzz] gdzie:

- x - kod Systemu Informatycznego ARiMR,
- y - kod podsystemu Systemu Informatycznego ARiMR,
- zzzz – (opcjonalnie, o ile zdefiniowano) unikalny numer elementu modelu.

Identyfikator umieszczany jest w polu „Alias”.

Klasy i przypadki użycia reprezentujące elementy usuwane powinny być oznaczone kolorem czerwonym.

Klasy i przypadki użycia reprezentujące elementy zmieniane przed zmianą powinny być oznaczone kolorem szarym.

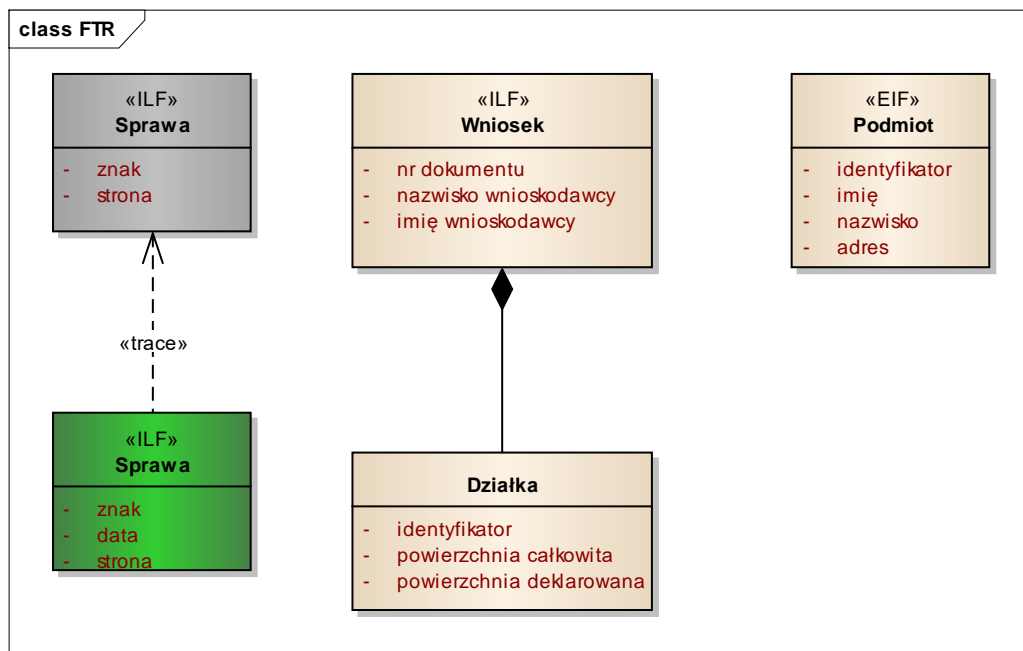
Klasy i przypadki użycia reprezentujące elementy nowe i zmienione po zmianie powinny być oznaczone kolorem zielonym.

Klasy i przypadki użycia reprezentujące elementy nie zmienione, powinny być oznaczone kolorem żółtym.

Przykładu modelu IFPUG załączono do Podręcznika:

- Załącznik nr 1A ARiMR_model_IFPUG_przyklad.eap
- Załącznik nr 1B ARiMR_model_IFPUG_zmiana_przyklad.eap

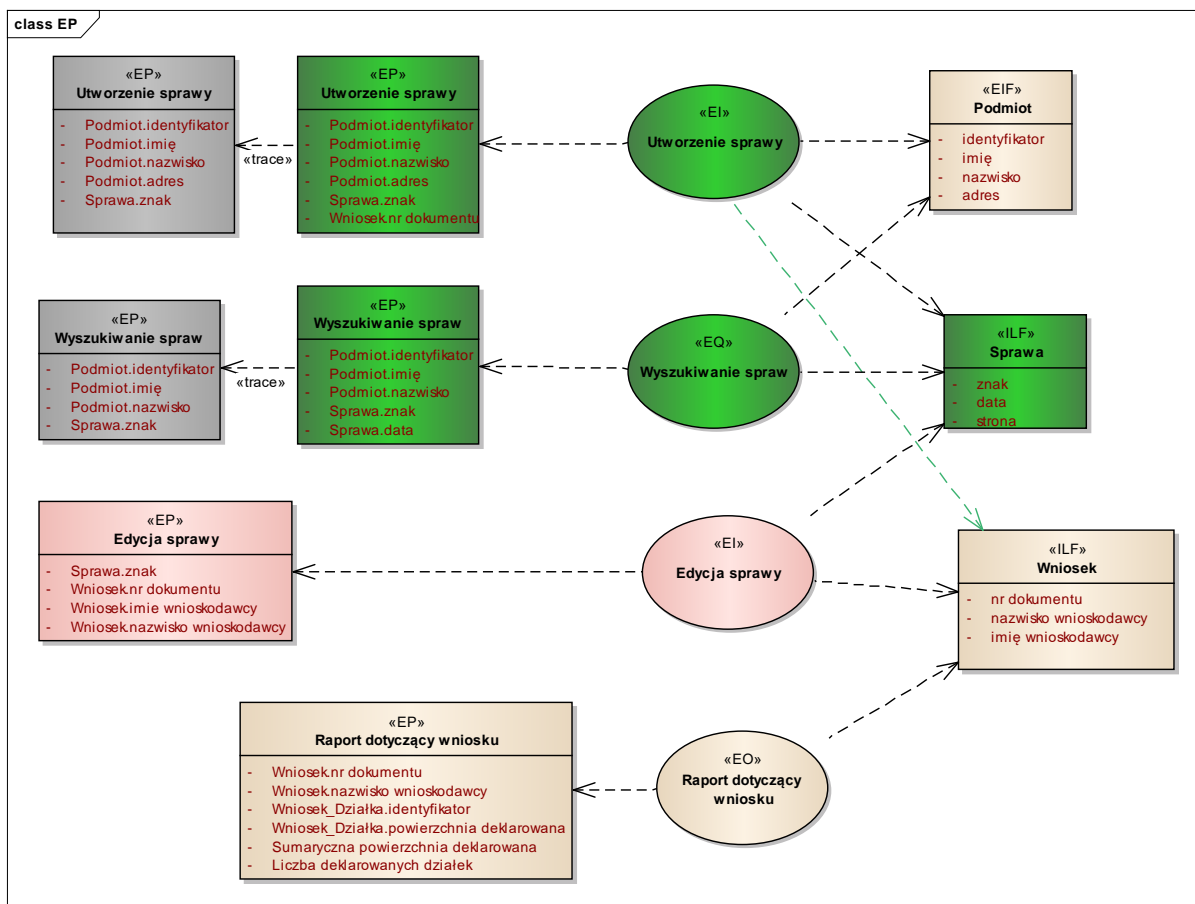
Przykład:



Rysunek 1 Modelowanie Plików danych

przedstawia fragment modelu IFPUG pewnej zmiany, w którym (od lewej):

- jeden zidentyfikowany ILF „Sprawa”, który uległ zmianie, przed zmianą posiadał 2 DETy, po zmianie posiada 3 DETy, dodany został DET „data”,
- jeden zidentyfikowany ILF „Wniosek”, który nie uległ zmianie, który posiada 2 RETy (Wniosek, Działka) i 6 DETów (nr dokumentu, nazwisko wnioskodawcy, imię wnioskodawcy, identyfikator działki, powierzchnię całkowitą działki, powierzchnię deklarowaną działki),
- jeden zidentyfikowany EIF „Podmiot” posiadający jednego RETa, oraz 4 DETy (identyfikator, imię, nazwisko, adres).



Rysunek 2 Modelowanie Elementarnych procesów

przedstawia fragment modelu IFPUG pewnej zmiany, w którym:

- Funkcja wejściowa (EI) „Utworzenie sprawy” uległa zmianie. Funkcja ta korzysta z 3 Plików danych (FTR): „Podmiot”, „Sprawa”, „Wniosek”. W wyniku zmiany funkcja zaczęła korzystać z Pliku danych „Wniosek”. Korzysta też z Pliku danych „Sprawa”, która sama uległa zmianie. Po zmianie funkcja posiada 6 DETów, 4 z nich pochodzi z pliku Podmiot (pierwszy człon nazwy atrybutu), 1 z pliku „Sprawa” i 1 z pliku „Wniosek”. W wyniku zmiany Elementarnego procesu „Utworzenie sprawy” dodany został 1 DET „Wniosek.nr dokumentu”.
- Funkcja zapytania (EQ) „Wyszukanie sprawy” uległa zmianie. Funkcja ta korzysta z 2 Plików danych: „Podmiot”, „Sprawa”. Po zmianie funkcja posiada 5 DETów: 3 DETy pochodzą z pliku „Podmiot”, a 2 z pliku „Sprawa”, w wyniku zmiany doszedł 1 DET o nazwie „Sprawa.data”.
- Funkcja wejściowa (EI) „Edycja sprawy” została usunięta.
- Funkcja wyjściowa (EO) „Raport dotyczący wniosku” nie uległa zmianie. Funkcja korzysta z jednego Pliku danych: „Wniosek”. Funkcja posiada 6 DETów, z czego 4 pochodzą z Wniosku, a 2 reprezentują dane, które są wyznaczane w trakcie Elementarnego procesu i prezentowane użytkownikowi.

Za utrzymanie, aktualizację, walidację modelu artefaktów PF IFPUG odpowiada **Strona przeprowadzająca wymiarowanie**, czyli zespół ekspertów, reprezentujący Podmiot, który na

podstawie dostępnej dokumentacji dotyczącej oprogramowania lub projektu np. umowy, dokumentacji analitycznej, odpowiada za przeprowadzenie Wymiarowania.

ZASADY PREZENTACJI WYNIKÓW WYMIAROWANIA FUNKCJONALNEGO W DOKUMENTACJI OFERTOWEJ

W ramach przedstawienia wyników oszacowania w ofercie Wykonawca jest zobowiązany wskazać:

- dodawane elementy modelu IFPUG,
- usuwane elementy modelu IFPUG,
- modyfikowane elementy modelu IFPUG.

Dla każdego wskazywanego elementu modelu musi być podany jego numer, nazwa, wymaganie funkcjonalne, z którego wynika dany element, typ oraz ostateczny rozmiar funkcjonalny zmiany (w przypadku modyfikowanych elementów – rozmiar po uwzględnieniu współczynników wpływu).

Na żądanie ARiMR Wykonawca jest zobowiązany przekazać model IFPUG w postaci projektów EA przed zmianą i po zmianie tj. z modelem IFPUG reprezentującym stan oprogramowania przed wprowadzeniem Modyfikacji oraz po Modyfikacji.

Szablon raportu prezentującego wyniki wymiarowania funkcjonalności załączono do Podręcznika:

- *Podręcznik_Załącznik nr 2 Raport_z_wymiarowania_funkcjonalności.xlsx*

Wyniki wymiarowania funkcjonalnego muszą być zgodne z modelem artefaktów PF IFPUG.