

Inżynieria oprogramowania II

Wykład 2:

“Proces tworzenia oprogramowania (na podstawie Unified Process)”

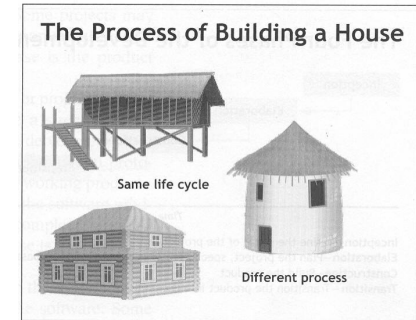
Marek Krętowski
pokój 206
e-mail: mkret@ii.pb.bialystok.pl
http://aragorn.pb.bialystok.pl/~mkret



Wersja 1.02

Analogia do budowanego domu

- Cykl życia domu:
 - pomysł (miejsce, styl, budżet, otoczenie, wyposażenie, ...)
 - planowanie (zakup gruntu, projektowanie, wybór wykonawcy, zapewnienie finansowania, ...)
 - realizacja (budowa)
 - zamieszkanie (przekazanie budynku właścicielowi, poprawki, przeprowadzka)
- Nowe potrzeby (garaż, większa kuchnia, ...) mogą spowodować ponowne przejście przez wymienione fazy
- Iteracje w ramach jednej fazy
- Proces definiuje sposób w jaki budynek powstaje (proces może być inny w różnych częściach świata, ale cykl życia jest ten sam)



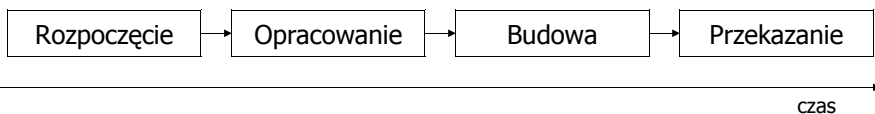
- Proces może być bardzo rygorystyczny (firma realizująca wiele budów równocześnie) lub mniej formalny (amator budujący domek na działce)
- Proces powinien być adaptowalny

IO2, wyk.2

Slajd 2 z 19

Cykl życia oprogramowania (ang. *software life cycle*)

- Obejmuje okres od pojawienia się pierwszych pomysłów dotyczących oprogramowania aż do rezygnacji z jego wykorzystania lub do wymiany na inne oprogramowanie
- Od momentu narodzin aż do śmierci programu obserwujemy serie różnego rodzaju zmian i modyfikacji (różnej wagi i znaczenia) stymulowanych przez potrzeby użytkownika
- Obejmuje początkowy cykl budowy (ang. *development cycle*) po którym następują kolejne cykle rozwoju (ang. *evolution cycles*)
- Niezależnie od aktualnego cyklu wyróżniane są 4 fazy:



IO2, wyk.2

Slajd 3 z 19

Cztery fazy cyklu życia

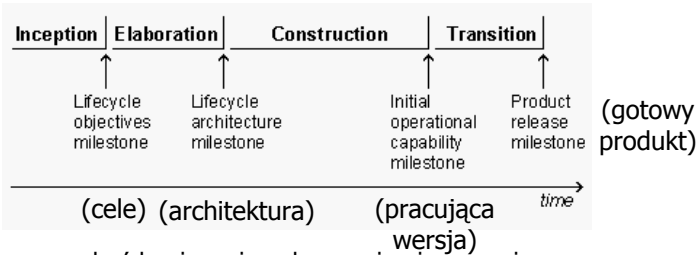
- **Rozpoczęcie**, inicjacja (ang. *inception - the concept*) - celem jest uzasadnienie (biznesowe) tworzenia nowego produktu lub modyfikowania istniejącego; główne zadania to opracowanie podstawowych wymagań dla projektu, przeprowadzenie wstępnej analizy ryzyka i ew. stworzenie prototypu pojęciowego
- **Opracowanie**, rozwinięcie - (ang. *elaboration - the design*) - analiza dziedziny problemu w celu opracowania odpowiedniej architektury produktu, przygotowanie planu pracy, który neutralizuje najistotniejsze zagrożenia; m.in. stworzenie modelu zachowania systemu i modelu dziedziny
- **Budowa** (ang. *construction*) - obejmuje zakodowanie (programowanie) produktu, stworzenie dokumentacji systemowej i użytkowników, opracowanie planu wdrożenia, zapewnienie jakości produktu
- **Przekazanie**, wdrożenie (ang. *transition*) - polega na udostępnieniu oprogramowania użytkownikowi; wiąże się zwykle z dopracowaniem szczegółów, usunięciem nie wykrytych wcześniej błędów, ...

IO2, wyk.2

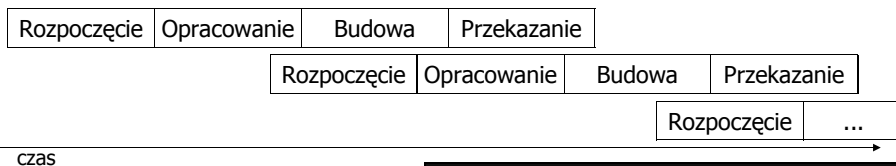
Slajd 4 z 19

Kamienie milowe

- Każda z faz powinna kończyć się tzw. kamieniami milowymi



- Cykle nie muszą być konieczne sekwencyjne i mogą się w pewnym zakresie nakładać



IO2, wyk.2

Slajd 5 z 19

Proces tworzenia oprogramowania (ang. *software engineering process*)

- Proces definiuje sposób postępowania, którego zastosowanie prowadzi do efektywnego tworzenia i/lub rozwoju oprogramowania
- Opiera się na opracowywaniu kolejnych generacji produktu (podejście ewolucyjne)
- Redukuje ryzyko niepowodzenia (wystąpienie możliwych do przewidzenia trudności) i umożliwia poprawę jakości
- Wychwytuje i dokumentuje najlepsze rozwiązania praktyczne gromadzone na podstawie wcześniejszych doświadczeń
- Promuje wspólną wizję (widzenie problemów) i kulturę pracy, poprzez zapewnienie zaleceń do stosowania odpowiednich narzędzi

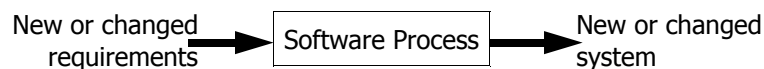
IO2, wyk.2

Slajd 6 z 19

Proces tworzenia oprogramowania (2)

- Proces definiuje w jaki sposób ludzie przeprowadzają kolejne etapy cyklu życia oprogramowania
- Cykl życia jest związany (należy do) produktu, natomiast proces opisuje czynności wykonywane przez ludzi w ramach cyklu
- Innymi słowy proces definiuje **kto** wykonuje **co**, **kiedy** i **jak** aby osiągnąć zamierzony cel (sukces przedsięwzięcia programistycznego)
- Stosowanie procesu zasadniczo poprawia rezultaty pracy zespołów wytwórczych

Who Is Doing What, When, and How



IO2, wyk.2

Slajd 7 z 19

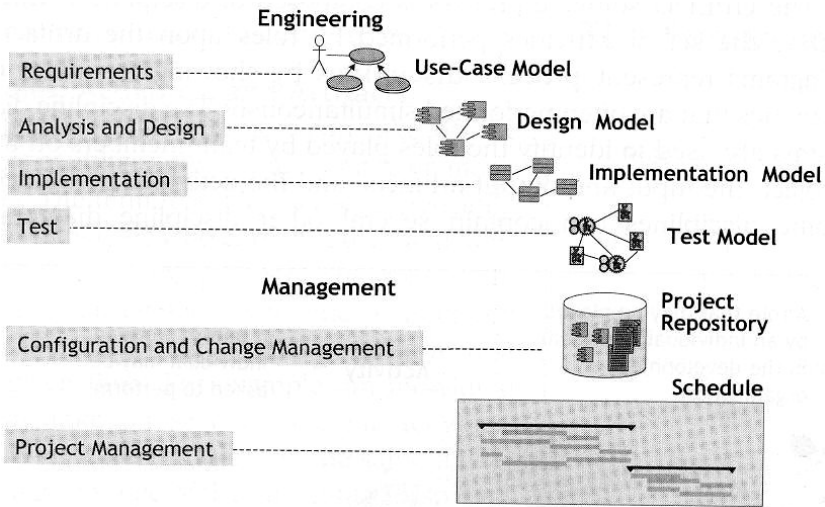
Dyscypliny (dziedziny) procesu

- Proces zawiera wszystkie czynności niezbędne do przełożenia (przetransformowania) potrzeb użytkownika w funkcjonujący produkt
- Wyróżnione są dwie grupy dyscyplin:
- inżynierskie (ang. *engineering disciplines*) - obejmują techniczne aspekty procesu takie jak opracowanie wymagań, na ich podstawie stworzenie projektu, implementację i testowanie kodu:
 - wymagania (przypadki użycia do opisanego zachowania systemu)
 - analiza i projektowanie (specyfikacja umożliwiająca implementację systemu)
 - implementacja (organizacja kodu, programowanie, integracja modułów)
 - testowanie (zapewnienie oczekiwanego poziomu niezawodności)
 - zarządcze (ang. *management disciplines*):
 - zarządzanie konfiguracją i zmianami
 - zarządzanie projektem (równoważenie sprzecznych celów, zarządzanie ryzykiem, pokonywanie ograniczeń)

IO2, wyk.2

Slajd 8 z 19

Dyscypliny procesu (2)



IO2, wyk.2

Slajd 9 z 19

UPEDU a RUP

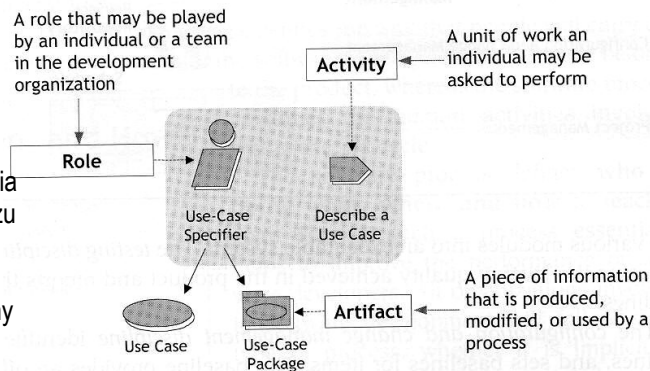
- Unified Process for EDUcation (UPEDU) jest procesem tworzenia oprogramowania powstałym na bazie Rational Unified Process(RUP)
- Stworzony został z myślą o wykorzystaniu w nauczaniu inżynierii oprogramowania
- Zawiera tylko najważniejsze elementy pełnego procesu, dzięki czemu możliwe jest skoncentrowanie się na najistotniejszych jego elementach, bez grzęźnięcia w drugorzędnych szczegółach
- UPEDU składa się z dyscyplin, z których każda definiuje zbiór czynności (działań) wykonywanych przez członków zespołu, w wyniku których tworzone są lub modyfikowane elementy projektu
- Dla poszczególnych dyscyplin istnieją diagramy (może być ich wiele), które pokazują role odgrywane przez członków zespołu, wejściowe i wyjściowe elementy oraz podejmowane działania

IO2, wyk.2

Slajd 10 z 19

Organizacja procesu - diagramy dyscyplin

- Diagramy wykorzystywane są do przedstawienia całościowego obrazu czynności wykonywanych w ramach dyscypliny (sekwencyjnie lub równolegle)
- Pojedynczy członek zespołu może równocześnie odgrywać różne role w różnych czynnościach pracując nad innymi elementami

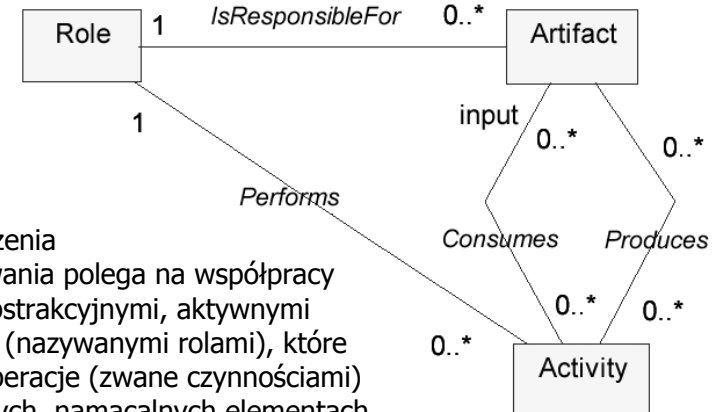


IO2, wyk.2

Slajd 11 z 19

Podstawowy model pojęciowy

Proces tworzenia oprogramowania polega na współpracy pomiędzy abstrakcyjnymi, aktywnymi jednostkami (nazywanymi rolami), które wykonują operacje (zwane czynnościami) na konkretnych, namacalnych elementach (zwanym artefaktami)



IO2, wyk.2

Slajd 12 z 19

Role, czynności i artefakty

- **Rola** (ang. role) - jest odgrywana przez pojedynczą osobę lub niewielką grupę; definiuje zakres odpowiedzialności poszczególnych osób; rola określonego członka zespołu zmienia się w czasie projektu; jest sterowana przez aktualne potrzeby
- **Czynność**, działania (ang. activity) - najmniejsza część pracy wykonywana w ramach procesu przez rolę. Technicznie określana jest jako "operacja", czyli jest w zasadzie niepodzielna (możliwe są pewne opcjonalne kroki, np. "myślenie, wykonywanie, przeglądanie"). Czynność musi mieć jasno zdefiniowany cel, wyrażony zwykle w terminach tworzenia lub zmiany elementów (artefaktów). Odpowiadająca zwykle od kilku godzin do kilku osobo-dni.
- **Artefakt** (ang. artifact) - dowolna grupa informacji lub fizyczny byt wykorzystywany (przetwarzany) przez czynności procesu. Przykłady: modele, plany, kod, programy wykonywalne, dokumenty, bazy danych, ...

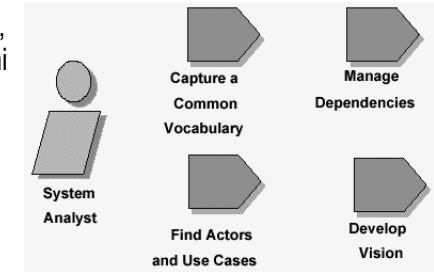
IO2, wyk.2

Slajd 13 z 19

Czynności i artefakty

- Podział pracy na czynności ułatwia monitorowanie, gdyż każda czynność jest odrębnym zadaniem, które może być zaplanowane i ocenione

- Artefakty są elementami tworzonymi, modyfikowanymi i wykorzystywanymi przez proces; pracownicy mogą otrzymywać określone artefakty na wejściu czynności oraz rezultatem wykonania czynności są również artefakty

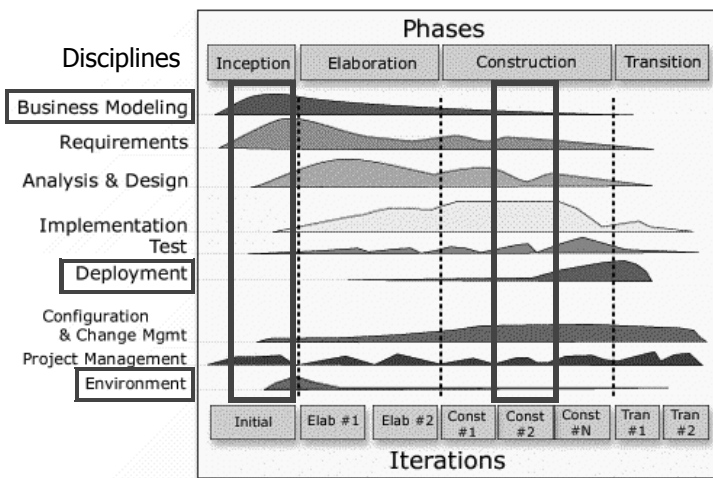


- Zwykle artefakty mają określone w miarę szczegółowe zalecenia (ang. guidelines), dzięki czemu mogą być opisane w sposób wyczerpujący; ponadto mogą składać się z innych elementów (np. model projektowy zawierający klasy)

IO2, wyk.2

Slajd 14 z 19

(Rational) Unified Process (for Education)



Iteracja to konstrukcja zarządcza, polegająca na tworzeniu kolejnych przybliżeń systemu

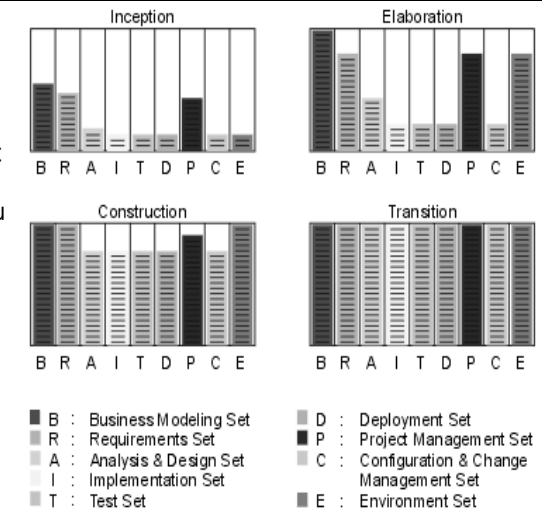
Waga dyscyplin w każdej iteracji zmienia się

IO2, wyk.2

Slajd 15 z 19

Charakterystyka procesu: iteracyjność

- W iteracyjnym procesie pewne elementy (artefakty) są rozwijane w różnych iteracjach
- Z punktu widzenia zarządzania procesem szczególnie istotne jest jasne określenie mechanizmów służących do szacowania postępu prac; ponadto ważna jest umiejętność stwierdzenia, że zadanie zostało już wykonane (*milestones* oraz *deadlines*)
- **Iteracja** - wyróżniona sekwencja działań wykonywana w określonym celu, posiadająca ustalony plan oraz kryteria oceny, skutkująca zmianą określonych elementów (artefaktów)



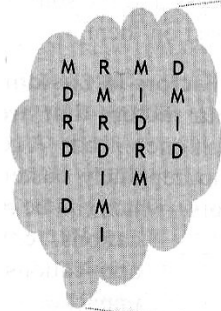
Ewolucja artefaktów w procesie wytwórczym

IO2, wyk.2

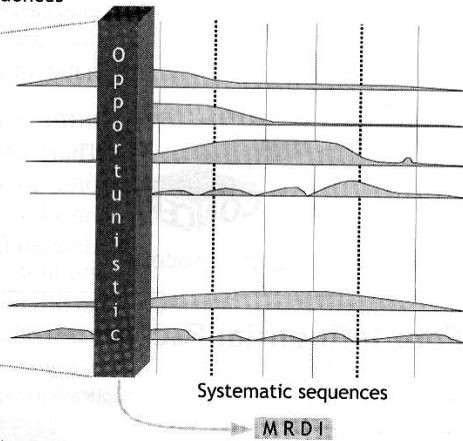
Slajd 16 z 19

Porządkowanie działań

Possible opportunistic sequences



M : Management artifacts
R : Requirements artifacts
D : Design artifacts
I : Implementation artifacts



IO2, wyk.2

Slajd 17 z 19

Charakterystyka UP (2)

- **Proces architekurocentryczny** - nacisk na możliwie wczesne opracowanie architektury systemu; dobra architektura:
 - ułatwia równoległe tworzenie oprogr.
 - wpływa na minimalizowanie powtarzania tych samych działań,
 - zwiększenie prawdopodobieństwa ponownego użycia i łatwość pielęgnacji (oprogramowanie komponentowe)
- **Ułatwia kontrolę jakości i zarządzania ryzykiem** - wykorzystanie do kontroli obiektowych miar i kryteriów jakości; dotyczy wszystkich działań i każdego uczestnika przedsięwzięcia
- **Proces konfigurowalny** - ponieważ nie istnieje metoda optymalna w każdym zastosowaniu UP jest elastyczny; można go dostosować zarówno do małych zespołów jak i wielkich firm informatycznych;
 - RUP zawiera informacje jak dostosowywać proces do konkretnych przedsięwzięć
- **Ukierunkowany na przypadki użycia** - kładzie nacisk na określenie pożądanego zachowania systemu;
 - przypadki użycia i scenariusze służą do uporządkowania procesu - od rozpoznawania wymagań do testowania systemu

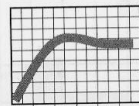
IO2, wyk.2

Slajd 18 z 19

Istotne czynniki podczas dostosowywania procesu

Czas (okres wyk.),
technologia,
planowane wersje

Product Life Cycle



Analiza i projektowanie
Zarządzanie konfiguracją i zmianami

Rodzaj aplikacji (biznesowa, czasu rzeczywistego, krytyczna, gra)

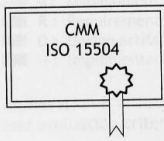
Application Domain



Wymagania Testowanie

Wszystkie dyscypliny

Process



Wzorce elementów,
wykorzystywane narzędzia,
standardowe architektury, ...

People and Organization



Analiza i projektowanie
Implementacja
Zarządzanie projektem

Rozmiar i struktura firmy (projektu),
kompetencje, doświadczenie,
istniejące systemy

IO2, wyk.2

Slajd 19 z 19