



TEHNIČKO VELEUČILIŠTE U ZAGREBU
POLYTECHNICUM ZAGRABIENSE

Objektno orijentirani razvoj programa

ISVU: 130938/19881

Dr. sc. Danko Ivošević, dipl. ing.
Predavač

Akademska godina 2021./2022.
Ljetni semestar



Izvor: Ivar Jacobson: Applying UML in The Unified Process

UNIFIED PROCESS (UP)

Kako je nastao UP? (1)



Rational Unified Process 5.0
1998

Rational Objectory Process 4.1
1996-1997

Objectory Process 1.0-3.8
1987-1995

The Rational Approach

The Ericsson Approach !!! rane 80'

Uključuje inženjerstvo poslovnog procesa, rukovanje zahtjevima, rukovanje oblikovanjem i promjenama, funkcijsko ispitivanje, evaluaciju performansi, inženjerstvo podataka, oblikovanje sučelja

UML

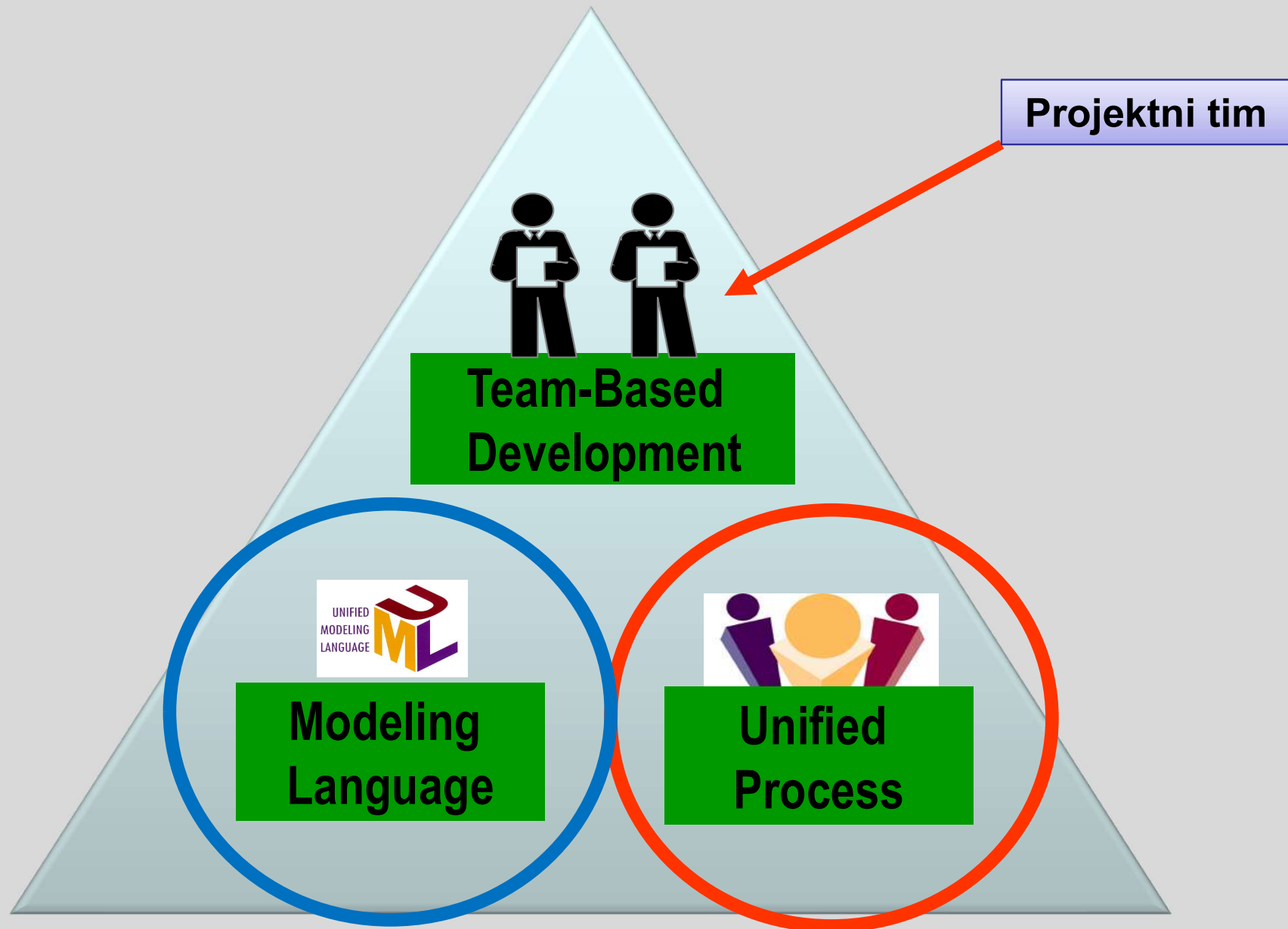
Kako je nastao UP? (2)

- Razvoj UP-a započeo je ranih 1980-ih u tvrtci Ericsson kao odgovor na potrebu za novom projektnom metodologijom koja će optimalno odgovarati različitim potrebama tvrtke za razvojem novih proizvoda u programskom inženjerstvu, ali i ne samo u njemu.
- Prema inačici UP-a koja je završena krajem 90-ih, određeno je da opseg i aktivnosti ove metodologije uključuju inženjerstvo poslovnog procesa, rukovanje zahtjevima, rukovanje oblikovanjem i promjenama, funkcijsko ispitivanje, vrednovanje performansi, inženjerstvo podataka te oblikovanje sučelja.
- Nakon 2003. godine razvoj UP-a nastavljen je u tvrtci Rational Software unutar IBM-a ➔ Rational Unified Process (RUP)

Što je UP?

- **Unificirani proces** (engl. *Unified Process*, UP) je metodologija razvoja programske potpore koja se temelji na oblikovanju pomoću modela (engl. *Model Based Design*, MBD), odnosno iterativnom razvoju, obraseima uporabe i usmjerenjem na arhitekturu sustava.

Tim + UML + UP = dobitna kombinacija



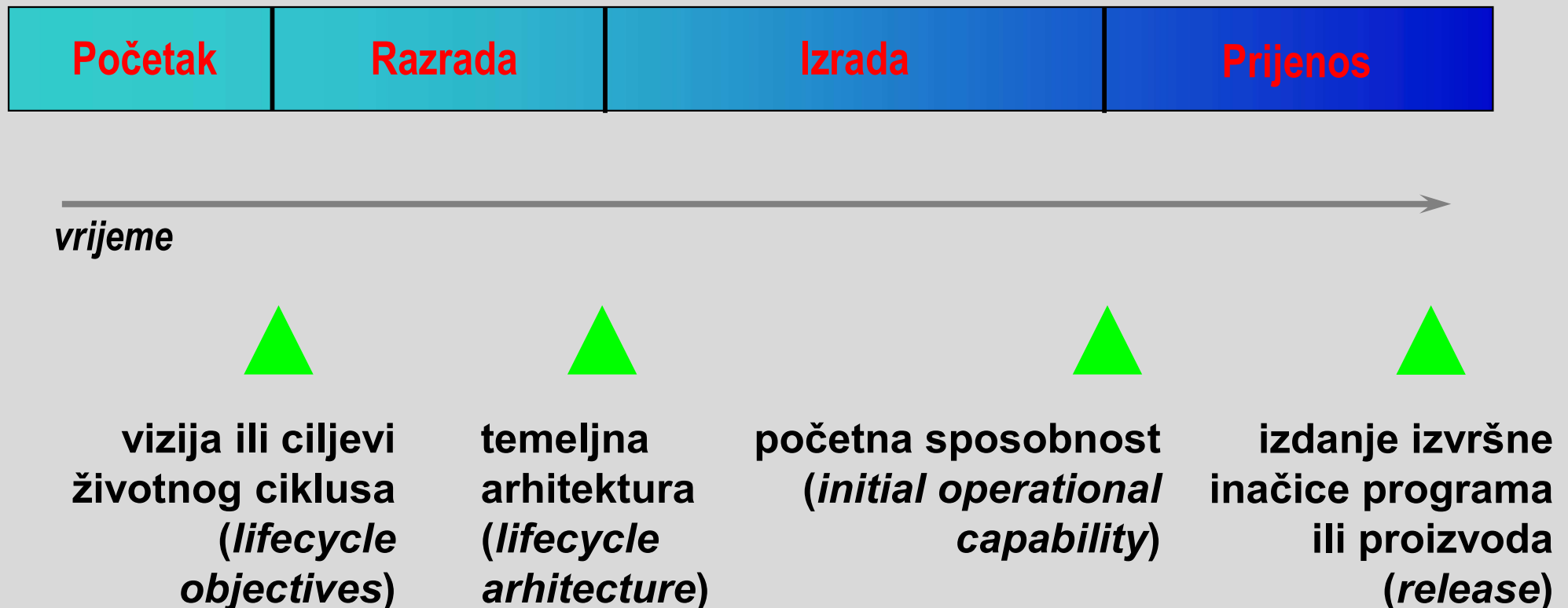
Prikaz UP-a

- UP se prikazuje kroz tri temeljna obilježja:
 1. Promiče **iteracije** na pojedinim **fazama** oblikovanja programske potpore
 2. Temelji se na **obrascima uporabe** (*use case*, tj. predlošku scenarija)
 3. U fokusu UP-a je **arhitektura** sustava

Faze u životnom ciklusu UP procesa

- Kao i svaka druga projektna metodologija, UP određuje faze životnog ciklusa (engl. *lifecycle*) procesa i dokumente koji se moraju izraditi završetkom izvođenja svake faze.
- **Faze životnog ciklusa UP-a su:**
 1. **Početak** (*inception*), u kojemu se definira doseg projekta, razvoj modela poslovnog procesa, specifikacija početnih zahtjeva,
 2. **Razrada, elaboracija** (*elaboration*), kada se definiraju plan projekta, specifikacija značajki i temelji arhitekture sustava,
 3. **Izgradnja, izrada** (*construction*), koja se sastoji od oblikovanja, programiranja i ispitivanja,
 4. **Prijenos** (*transition*), pri čemu se završni proizvod prenosi korisnicima i postavlja u radnu okolinu.

Glavne **ključne točke** i pridruženi dokumenti ("Milestones")



Faze i pridružene iteracije



Svaka faza ima barem jednu iteraciju. Stoga, u kontekstu UP-a, iteracija je niz ili slijed aktivnosti u okviru prihvaćenog plana i kriterija evaluacije. **Ishod svake iteracije je dokument, a na kraju i jedna izvršna inačica (tj. izdanje) programa ili sustava.**

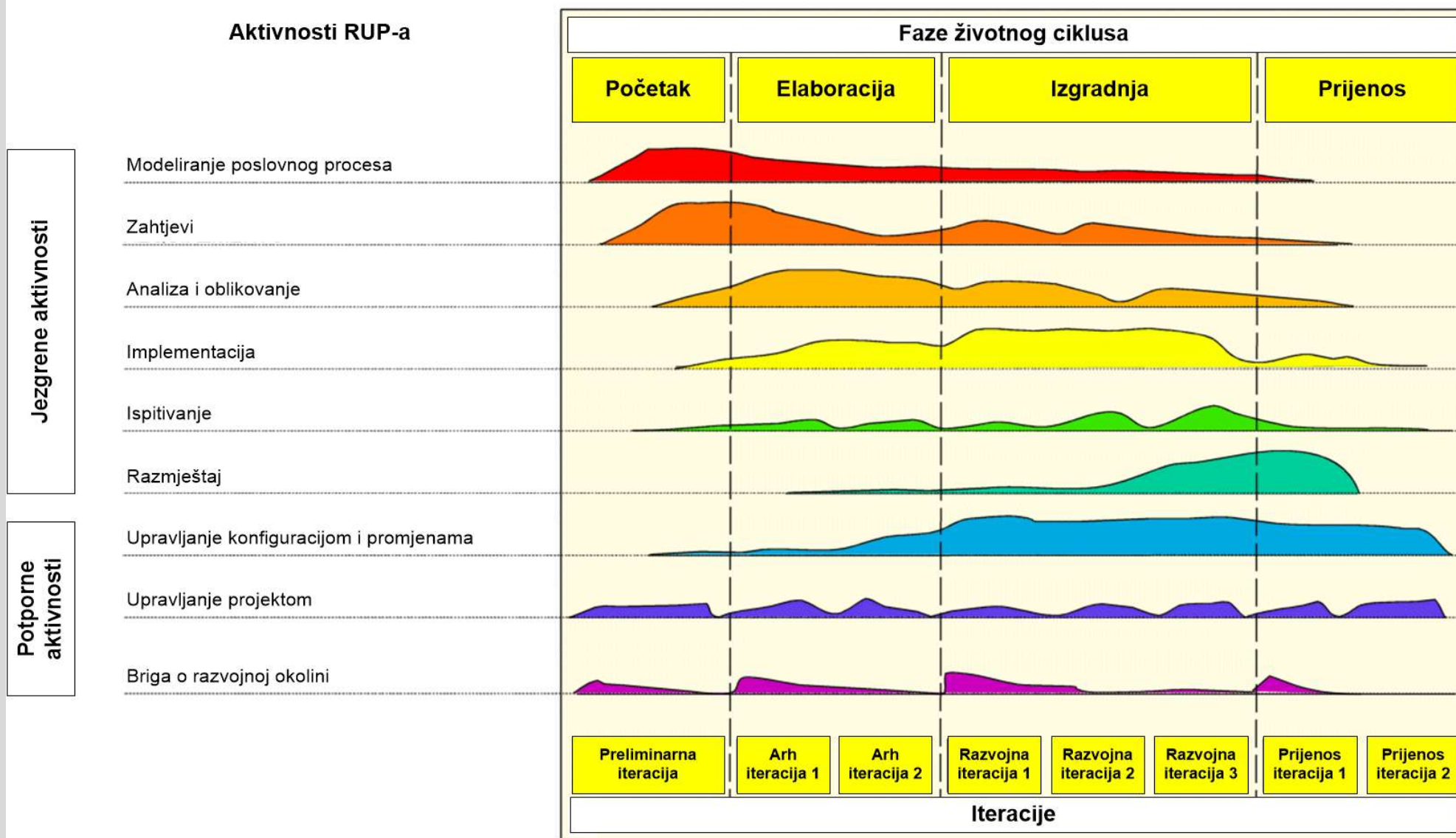
Iteracije i **aktivnosti**

- Osim faza životnog ciklusa, UP definira i niz aktivnosti koje se mogu odvijati u svim fazama, ali obično s različitim intenzitetom.
- **Jezgrene aktivnosti (core workflows)** UP-a su:
 1. Modeliranje poslovnog procesa (*business modelling*),
 2. Zahtjevi (*requirements*),
 3. Analiza i oblikovanje (*analysis and design*),
 4. Implementacija (*implementation*),
 5. Ispitivanje (*test*),
 6. Razmještaj (*deployment*)

Iteracije i **potporne aktivnosti**

- Vrlo često se dodaju dodatne, **potporne aktivnosti** (*support workflows*) vezane uz organizaciju i provedbu životnog ciklusa, kao što su: **upravljanje konfiguracijom i promjenama** (*configuration and change management*), **upravljanje projektom** (*project management*), te **briga o razvojnoj okolini** (*environment*; svodi se na ispravnu interakciju s dionicima projekta).
- Ove dodatne aktivnosti se zbog svoje namjene zajedno nazivaju potporne aktivnosti.

Iteracije i aktivnosti



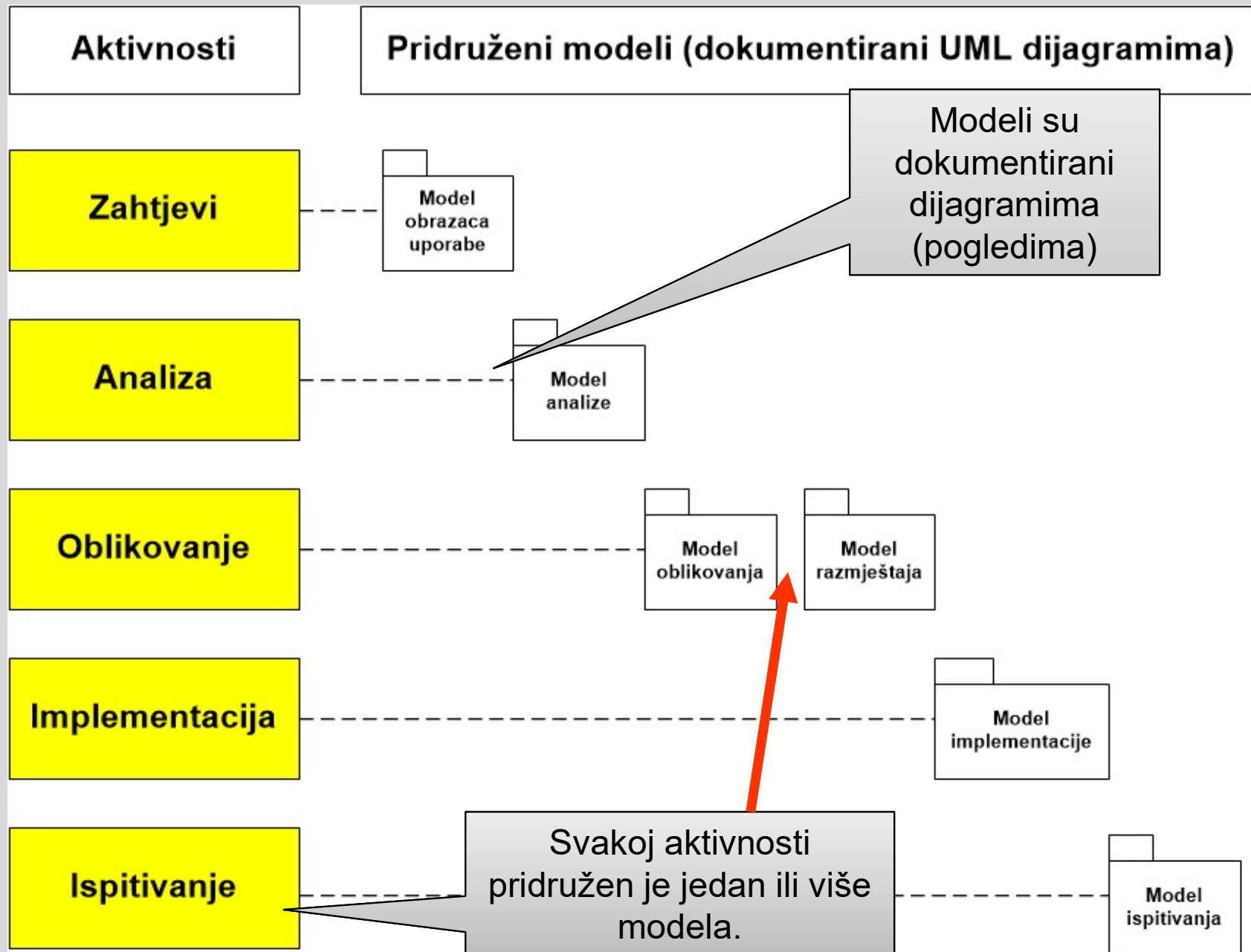
Vođenje projekata u modelno-usmjerenom razvoju

- **Ovisno o fazi u kojoj se nalazi razvoj proizvoda, bit će potrebno dodijeliti više ljudi (projektnih resursa) na određene aktivnosti, a manje na druge i obrnuto.**
- **Određivanjem prioriteta i raspoređivanjem resursa bavi se voditelj projekta (*project manager*).**
- Osim što se u svojoj strukturi temelji na iteracijama u svakoj pojedinoj fazi, UP intenzivno koristi obrasce uporabe (*use cases*) i usmjeruje se na arhitekturu sustava, odnosno na strukturu programske potpore koja se izrađuje.

Aktivnosti i pridruženi modeli

- Za opis pojedinih aktivnosti (tijeka rada) koriste se odgovarajući modeli (tj. UML dijagrami).
- Modeli su dokumentirani s jednim ili, u praksi češće, više dijagrama programske potpore. Dijagrami moraju biti definirani UML standardom.
 - Stoga, prema UP-u, oblikovana arhitektura sustava sadrži skup UML dijagrama kojima su opisani različiti pogledi (*views*) u modele sustava (*models*).
- Svakoj aktivnosti pridružen je jedan ili više modela za opis, koji su dokumentirani s jednim ili više dijagrama (tj. pogleda).

Aktivnosti i pridruženi modeli

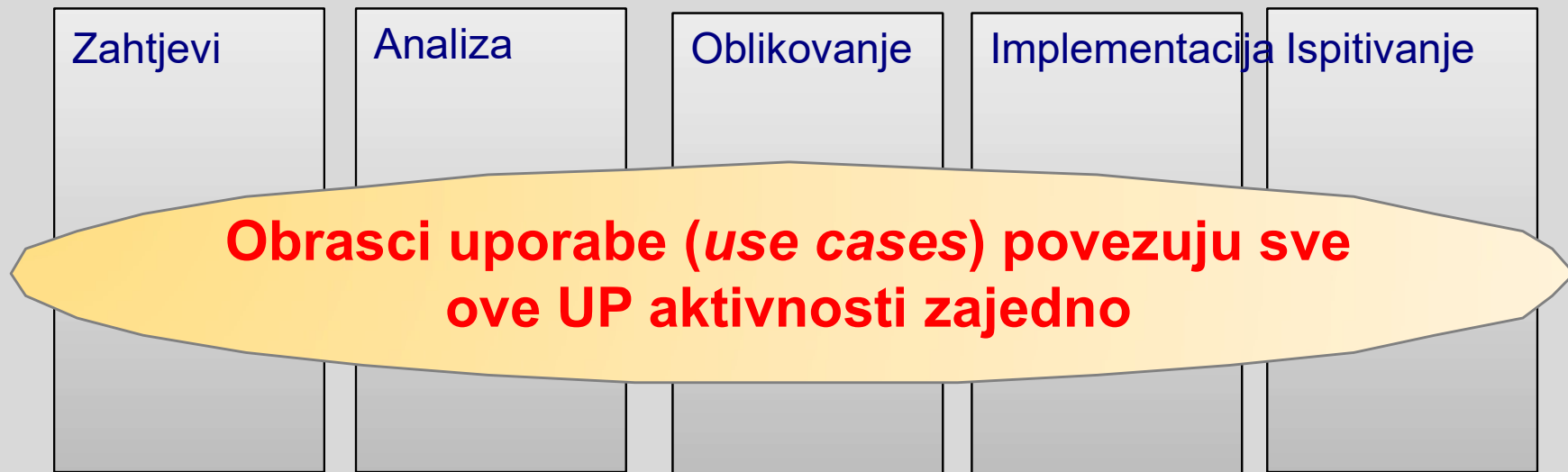


Perspektive opisa

- UP je najčešće opisan kroz tri perspektive koje se ogledavaju i u korištenim UML dijagramima:
 1. Dinamička perspektiva, koja pokazuje slijed faza procesa kroz vrijeme,
 2. Statička perspektiva, koja pokazuje aktivnosti u pojedinim fazama procesa, i
 3. Praktična perspektiva, koja sugerira aktivnosti kroz iskustvo i dobru praksu.

Što znači: Temelji se na obrascima uporabe?

Aktivnosti:



Obrasci uporabe kao pokretači iteracija

- **Obrasci uporabe su vrlo važni u modelno-usmjerenom razvoju**, jer povezuju i pokreću brojne aktivnosti u životnom ciklusu oblikovanja programske potpore, kao što su:
 1. Stvaranje i validacija arhitekture sustava,
 2. Definicija ispitnih slučajeva, scenarija i procedura,
 3. Planiranje pojedinih iteracija,
 4. Stvaranje korisničke dokumentacije,
 5. Razmještaj sustava (engl. *deployment*)
- Obrasci uporabe pomažu u sinkroniziranju sadržaja različitih modela.

Koja projektna metodologija je najbolja?

- **Ne postoji univerzalno optimalan proces oblikovanja programske potpore!**
- Svaki pristup ili metodologija imaju svoje komparativne prednosti i nedostatke.
- Stoga se tijekom razvoja UP-a vodilo računa o fleksibilnosti i budućim proširenjima, jer se time omogućuju razne strategije životnog ciklusa projekta.
- Na primjer, moguće je odabrati koje artefakte treba proizvesti, definirati nužne aktivnosti i potrebne resurse (ukupni trošak, potrebno vrijeme, nužni programeri, vještine i znanja, programska potpora, sklopovlje, itd.), te modelirati koncepte sustava.

UP ZAKLJUČCI

- U središtu UP procesa su obrasci uporabe sustava koji semantički povezuju sve aktivnosti.
- UP definira i međusobno povezuje dinamičke faze i statičke aktivnosti procesa (pomoću prikaza iteracija i aktivnosti).
- Za opis pojedinih aktivnosti koriste se odgovarajući modeli.
- **Modeli** su dokumentirani s jednim ili više dijagrama (pogleda na sustav).
- **Dijagrami** su definirani UML standardom.
- Oblikovana arhitektura sustava sadrži **skup pogleda u modele** (tj. **skup dijagrama**).

REFERENCE I LITERATURA

- Predavanja ovog predmeta
- I. Sommerville, Software Engineering, 9th ed., Addison-Wesley, Harlow, England, 2011.
- R. Pressman, Software Engineering: A Practitioner's Approach (7th Edition), McGraw-Hill Science/Engineering/Math, 2009.
- T. C. Lethbridge and R. Laganière, Object-Oriented Software Engineering: Practical Software Development Using UML and Java, 2nd ed., McGraw-Hill Publishing Company, London, 2004.
- Nastavni materijali kolegija Oblikovanje programske potpore, Fakultet elektrotehnike i računarstva, Sveučilište u Zagrebu.