

Vještačka inteligencija

Predavanje 8: Ekspertni sistemi

*„Postaviti pravo pitanje je pola znanja.“
~Roger Bacon*

Odgovorna nastavnica: Vanr. prof. dr Amila Akagić

Univerzitet u Sarajevu



Uvodne informacije

- This work is licensed under a Creative Commons 'Attribution-NonCommercial-ShareAlike 4.0 International' license. EN: <https://creativecommons.org/licenses/by-nc-sa/4.0/>



- Ovaj rad je licenciran pod međunarodnom licencom 'Imenovanje-Nekomercijalno-Dijeli pod istim uvjetima 4.0' od strane Creative Commons. HR: <https://creativecommons.org/licenses/by-nc-sa/4.0/deed.hr>

Najave

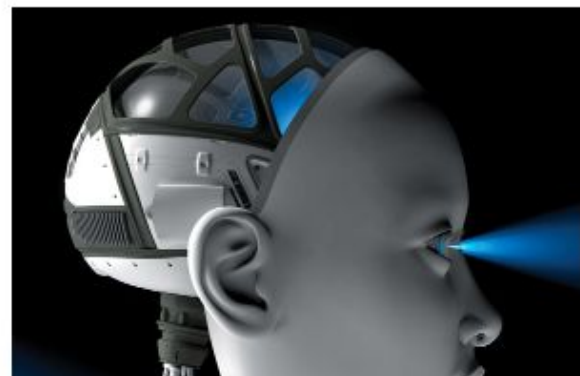
- ❑ Danas nema kviza. Naredni kviz je u sljedećoj sedmici (lab6).
- ❑ Snimljeno predavanje 7 biti će objavljeno preko vikenda (nalazit ce se u resursima).
- ❑ Ovo je već X sedmica nastave.
- ❑ Očekujte u XV sedmici II provjeru znanja.

Building an AI That Feels

AI systems with emotional intelligence could learn faster and be more helpful

By Mary Czerwinski, Javier Hernandez and Daniel McDuff

Our team hails from Microsoft's Human Understanding and Empathy group, where our mission is to imbue technology with emotional



About 1 billion people globally are affected by mental disorders; a scalable solution such as an AI therapist could be a huge boon.

<https://spectrum.ieee.org/artificial-intelligence/machine-learning/building-an-ai-that-feels>

Primjer virtuelnog asistena: Woebot



Podoblasti vještačke inteligencije



Metodi realizacije inteligentnih sistema

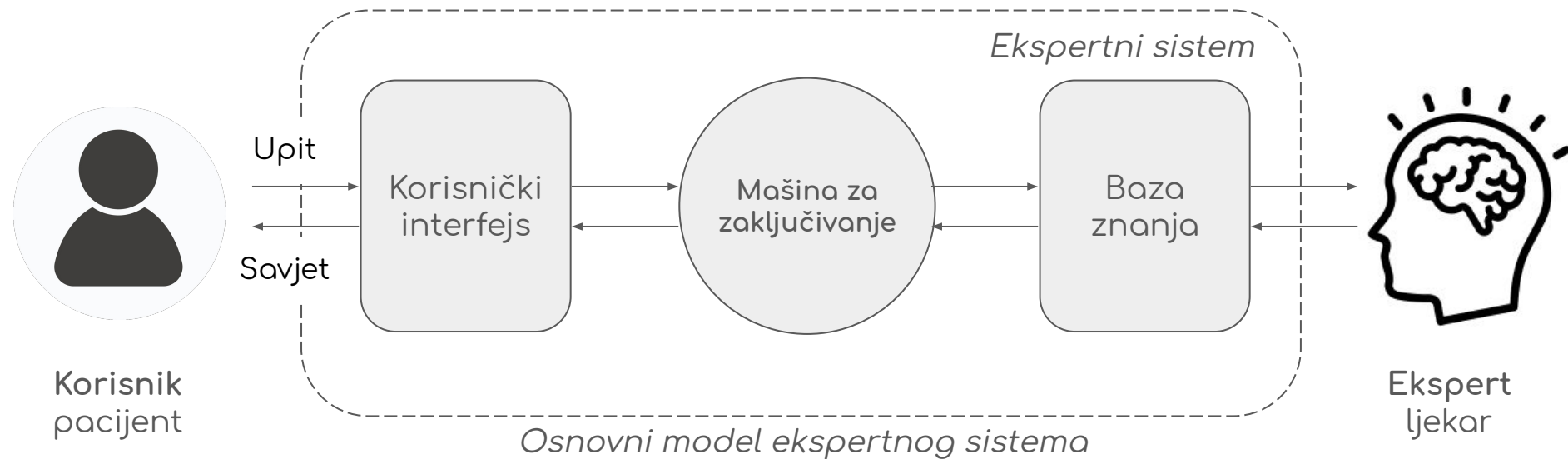
- ❑ Intelligentni sistemi mogu istovremeno da koriste jedan ili više metoda realizacije, u koje spadaju:
 - ❑ Ekspertni sistemi (*Expert Systems*),
 - ❑ Vještačke neuronske mreže (*Artificial Neural Networks*),
 - ❑ Genetički algoritmi (*Genetic Algorithms*, GA),
 - ❑ Fuzzy sistemi (*Fuzzy Systems*),
 - ❑ Zaključivanje na osnovu slučajeva (*Case Based Reasoning*, CBR),
 - ❑ Istraživanje podataka/otkrivanje znanja (*Data Mining/Knowledge Data Discovery*),
 - ❑ Intelligentni agenti (*Intelligent Software Agents*),
 - ❑ Komunikacija u prirodnom jeziku (*Language Technology/Natural Language Processing*).

Intelligentni agenti u e-trgovini

- ❑ Primjena u B2C (*Business-to-Consumer*):
 - ❑ posredovanje proizvoda (*Product Brokering*) - agent predlaže kupcu proizvode (sistem *Jango*);
 - ❑ trgovačko posredovanje (*Merchant Brokering*) - agent predlaže prodavca određenog proizvoda (sistem *BargainFinder*);
 - ❑ pregovaranje - agent određuje cijene ili druge elemente transakcija (aukcije i ugovori,...).
- ❑ Primjena u B2B (*Business-to-Business*) obuhvata npr. lance snabdjevanja (*Supply Chain Management*).
- ❑ Ostale primjene uključuju: mobilne agente, evolucione agente, data mining agente, itd. O njima će biti riječi kada budemo raditi Intelligentne agente.

Ekspertni sistemi (ES)

- ❑ Ekspertni sistemi su računarski bazirani informacijski sistemi koji koriste znanje eksperta za donošenje odluka i pokretanja akcija u svrhu rješavanja problema.
- ❑ MYCIN: prvi pravi poznati ekspertni sistem, korišten za medicinsku dijagnozu, razvijen je na Univerzitetu Stanford početkom 1980 godine.



Zaključivanje (*inference*)

- ❑ Zaključivanje (*inference*) je proces izvođenja logičkih zaključaka na osnovu premisa za koje se zna ili pretpostavlja da su istinite.
- ❑ Može biti:
 - ❑ Induktivno (od pojedinačnog ka općem),
 - ❑ Deduktivno (od općeg ka pojedinačnom),
 - ❑ Abduktivno (od konkretnog ka konkretnom).
- ❑ Zaključivanje u uslovima neizvesnosti proučava se u okviru matematičkih disciplina verovatnoće i statistike. Postoje različite forme zaključivanja (logike), a osnovne su logika sudova i logika predikata.
- ❑ Bajesova formula je jedan od pristupa zaključivanju na osnovu vjerovatnoća.
- ❑ Na primjer: možemo je koristiti za izračunavanje vjerovatnoće neke dijagnoze d ukoliko imamo listu simptoma s.

Ekspertni sistemi (ES)

“**Ekspertni sistem** je računarski sistem koji oponaša ili djeluje u svim pogledima kao ljudski ekspert (stručnjak), s mogućnostima donošenja odluka ljudskog stručnjaka.”

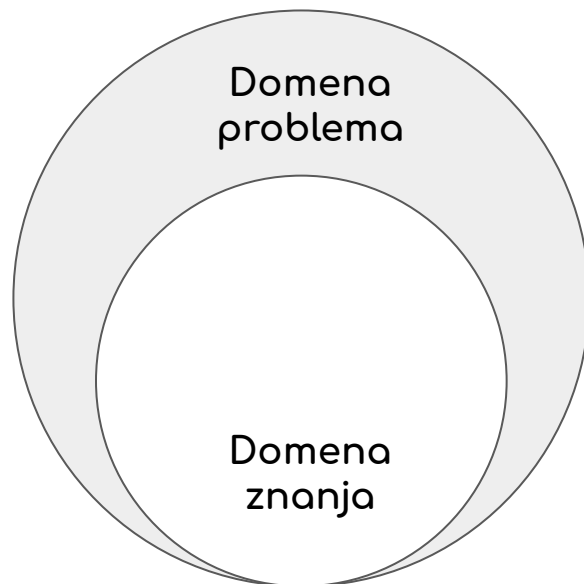
Professor Edward Feigenbaum
Stanford University

- ❑ Polje istraživanja: Računarske nauke / Vještačka inteligencija
- ❑ Dobitnik nagrada:
 - ❑ ACM Turing Award (1994)
 - ❑ Computer Pioneer Award
 - ❑ AAAI Fellow (1990)
 - ❑ ACM Fellow (2003)
- ❑ *Doktorska disertacija 1960 (superviziran od strane: Herbert A. Simon, dobitnik Nobelove nagrade za ekonomiju 1978. i Turingove nagrade 1975. godine.)*



(1936-)

Domena problema i domena znanja



- ❑ Znanje stručnjaka specifično je za jednu domenu problema - medicinu, finansije, nauku, inženjerstvo itd.
- ❑ Znanje stručnjaka o rješavanju određenih problema naziva se domena znanja.
- ❑ Domena znanja je uvijek podskup domene problema.

Historijat

- ❑ Prve poznate ekspertne sisteme DENDRAL i MYCIN pokrenuo je otac ekspertnih sistema Edward Feigenbaum na Stanfordu 1960.
 - ❑ DENDRAL (Dendritic Algorithm) (Stanford 1965) - ekspertni sistem za pomoć u identifikaciji molekula u organskoj hemiji na osnovu spektrograma mase; Napisan u Lisp jeziku;
 - ❑ MYCIN (Stanford 1980) – prvi pravi poznati ekspertni sistem – korišten za dijagnozu bakterijskih infekcija i propisivanje antibiotskih terapija (≈ 600 pravila, 69% tačnost - bolje nego ljekari!);
 - ❑ Na osnovu ovom primjera kasnije su razvijeni ekspertni sistemi KEE i CADUCEUS.

Historijat

- ❑ Drugi poznati ekspertni sistemi uključuju:
 - ❑ **XCON** = eXpert **CON**figurer (DEC ili Digital Equipment Corporation 1970) – prva poznata aplikacija koja je korištena za namjenu sistemske konfiguracije (≈ 2500 pravila).
 - ❑ **Loan Probe** (Peat Marwick 1984) - daje preporuke o izdavanju kredita u najkraćem mogućem roku koristeći minimalnu količinu potrebnih informacija (više od 8500 pravila);
 - ❑ **ExperTAX** (Coopers & Lybrand 1990) – ekspertni sistem koji je pomagao revizorima i poreznim ekspertima u planiranju, otkrivanju problema i alternativa (≈ 3000 pravila).

Ekspertni sistemi (ES)

- ❑ Primjena ekspertnih sistema u praksi:
 - ❑ obračun poreza,
 - ❑ kreditnih analiza,
 - ❑ održavanje opreme,
 - ❑ dijagnoze grešaka i slično.
- ❑ Koristi se kao osnovni alat za poboljšanje produktivnosti i kvaliteta.
- ❑ Fundamentalni elementi nekog ES su:
 - ❑ Ekspert
 - ❑ Ekspertiza

Ekspert

- ❑ Osoba koja posjeduje specijalna znanja, iskustva, vještine i metode za savjetovanje i rješavanje problema.
 - ❑ Predviđa znanje neophodno za izvršavanje korisnih zadataka u sistemu.
 - ❑ Zna koje su činjenice važne i razumije značenje veza između njih.
- ❑ Tipični eksperti posjeduje nekoliko osnovnih osobina, od kojih su najspecifičnija sljedeća:
 - ❑ Prvo, mogu riješiti problem i postići bolje rezultate od prosječnih ljudi u istim i/ili sličnim oblastima
 - ❑ Drugo, eksperti se biraju u zavisnosti od problema koji se rješava.

Ekspert

- ❑ Neki zadaci koje eksperti izvršavaju:
 - ❑ prepoznaju i formulišu problem
 - ❑ brzo i korektno rješavaju problem
 - ❑ objasne postupak rješenja problema
 - ❑ uče iz iskustva
 - ❑ rekonstruišu znanje u svrhu rješenja problema
 - ❑ mogu da prekinu postojeća pravila ako je neophodno
 - ❑ mogu odrediti prioritete i slično.

Ekspertiza

- ❑ Ekspertiza (vještina) je opsežno znanje eksperta upotrijebljeno za rješavanje specifičnog zadatka.
- ❑ Ona proizilazi iz konstantnog usavršavanja eksperta putem **treninga**, svakodnevnog **čitanja** i znatnog **iskustva** koje se stiče kroz praksu.
- ❑ Pod tim se podrazumijeva **eksplicitno znanje**, stečeno putem teoretskih znanja iz knjiga i/ili časova, i **implicitnog znanja** stečenog kroz svakodnevnu praksu.

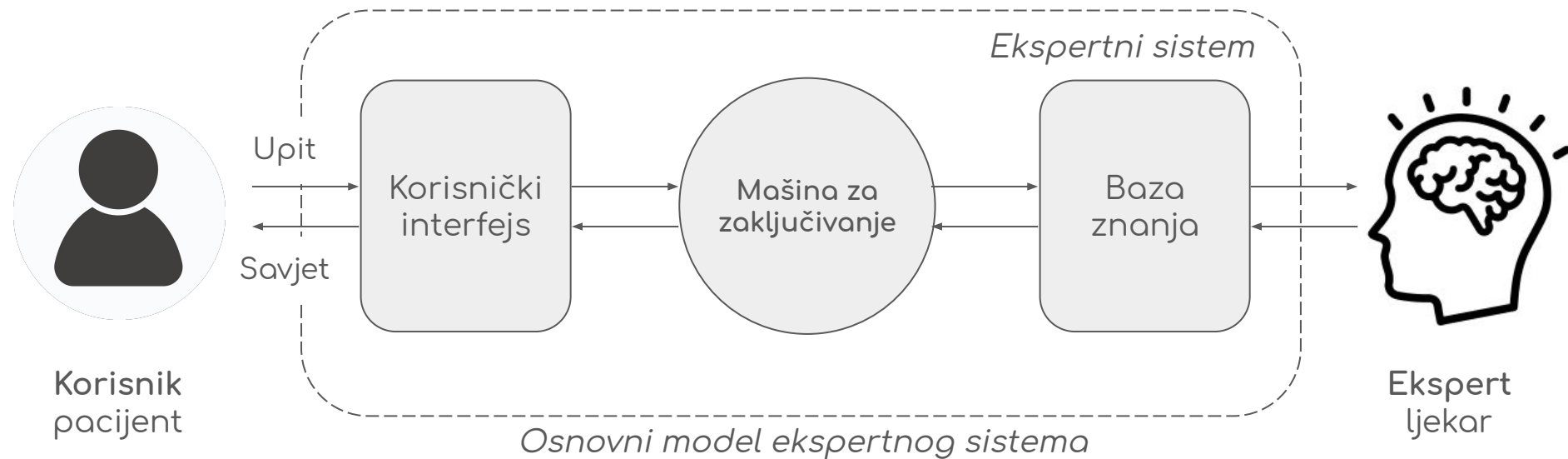
Ekspertiza

- ❑ Ekspertiza često ima sljedeće karakteristike:
 - ❑ ekspertiza je obično povezana sa visokim stepenom inteligencije
 - ❑ ekspertiza je obično povezana sa velikom količinom znanja
 - ❑ eksperti uče iz prethodnih uspjeha i grešaka
 - ❑ znanje eskperta je dobro organizirano i lako pretraživo, posebno u kritičnim momentima kada je neophodna brza odluka i zaključak.

Osobine ekspertnih sistema

- ❑ *Ekspertiza*: najvažniji dio ekspertnog sistema: eksperti
- ❑ *Simboličko zaključivanje*: simboličko zaključivanje
Znanje mora biti predstavljeno simbolički, tj. osnovni tok zaključivanja u sistemu mora biti jasno i precizno predstavljeno.
- ❑ *Znanje*: Baza znanja sadrži kompleksno znanje koje nije lako pronaći među ne-ekspertima.
- ❑ *Učenje*: sposobnost objašnjenja logičke odluke i detaljna interpretacija pojedinih procesa zaključivanja.
Većina današnjih ekspertnih sistema dizajnirana je tako da sadrži osobine **samostalnog učenja**, tj. putem konstantnog ažuriranja postojeće baze znanja, sistem poboljšava i optimizira procese u sistemu pri donošenju zaključaka.

Osnovni model ekspertnog sistema



Mogu sadržavati i sljedeće komponente:

- podsystem za prikupljanje podataka znanja
- radno okruženje
- podsystem za obrazloženje izvedenih zaključaka
- podsystem za filtriranje znanja

Baza znanja

- ❑ Najčešći način predstavljanja znanja u bazi znanja su **produkcionalna pravila**, **logički izrazi** i **semantike mreže**.
- ❑ **Produkcionalna pravila** (*production rules*) su jednostavna pravila koja imaju sljedeći oblik:

IF si umoran THEN se odmori

- ❑ Koriste se jer su razumljiva ljudima. Njihova struktura odgovara strukturi logičkog mišljenja i konstrukcijama prirodnog jezika (ako... onda...).
- ❑ Omogućava i jednostavno predstavljanje neizvjesnost i realizaciju probabilističkog zaključivanja.
- ❑ Logički izrazi koriste se za predstavljanje znanja u ekspertnim sistemima:

H:- B1, ..., Bn

H je zaključak, a B1,..., Bn su elementi složenog uslova tvrdnje.

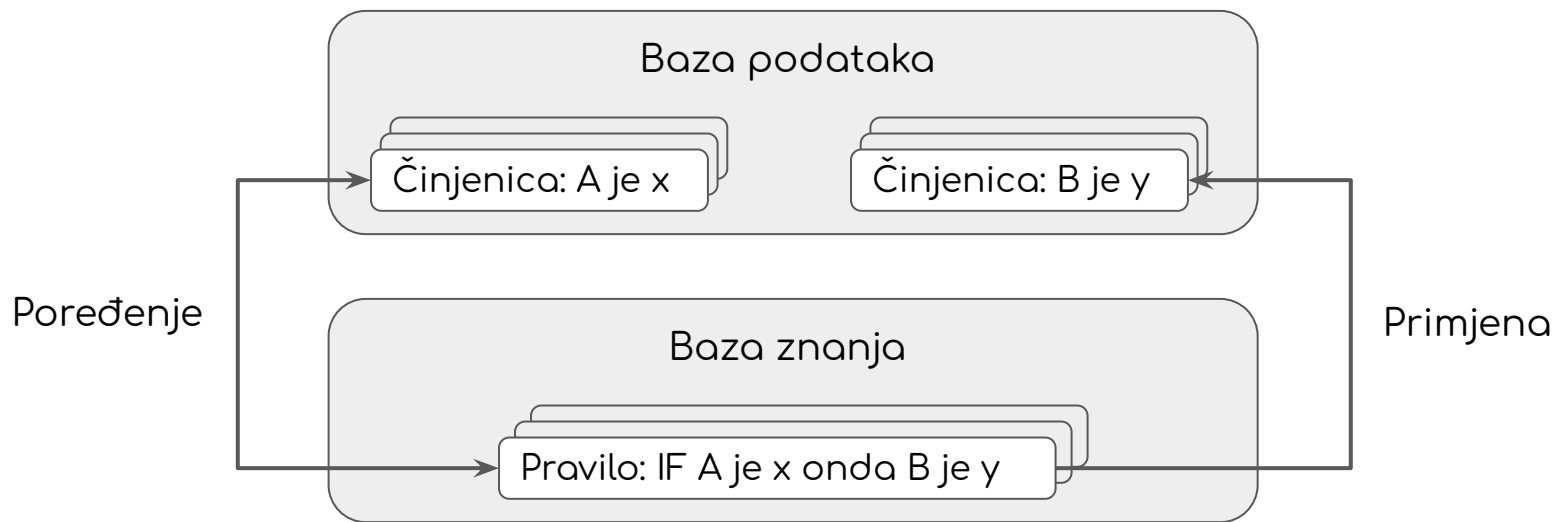
Mašina za zaključivanje

- ❑ U literaturi se jos naziva i mehanizam zaključivanje (od *Inference Engine*).
- ❑ Komponenta ekspertnog sistema koja realizuje operacije izvođenja zaključaka na osnovu trenutnog stanja baze podataka, odnosno radnog prostora ekspertnog sistema.

Predstavljanje znanja	Metod zaključivanja
Logička pravila	Princip rezolucije (<i>resolution principle</i>)
Produkciona pravila	Unaprijed (<i>forward</i>) i unazad (<i>backward</i>)
Sematičke mreže	Nasljeđivanje i posebne metode
Pamćenje primjera (slučajeva)	Analogno, koristi koncept sličnosti

Mašina za zaključivanje

- ❑ Ekspertni sistemi zasnovani na pravilima, znanje predstavljaju skupom **produkcionihi pravila** i **skupom činjenica**.
- ❑ Mehanizam zaključivanja poredi uslove pravila baze znanja (IF) sa činjenicama i aktivira pravila za koje su uslovi zadovoljeni (THEN), tj. izvršava predviđenu akciju, koja se može sastojati u promijeni činjenica.



Mašina za zaključivanje

□ Primjer:

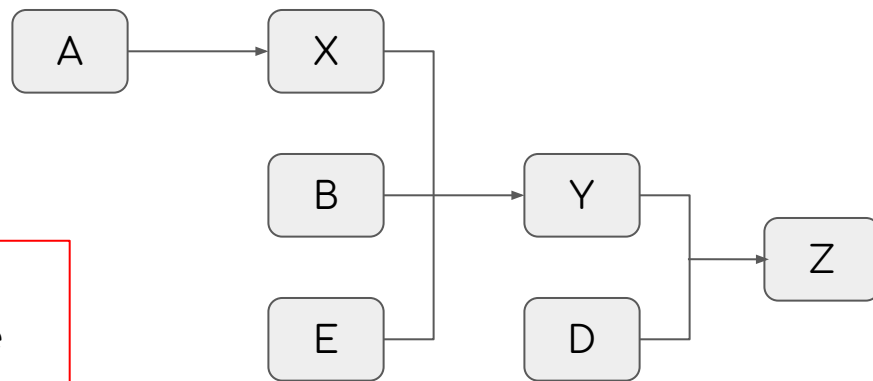
Pravilo 1: IF je Y tačno
AND je D tačno
THEN je Z tačno

Pravilo 2: IF je X tačno
AND je B tačno
AND je E tačno
THNE je Y tačno

Pravilo 3: IF je A tačno
THEN je X tačno

Drugo se
aktivira

Prvo se
aktivira



Dva osnovna načina izbora pravila koje treba primijeniti su **metod zaključivanja unaprijed** (*forward chaining*) i **zaključivanja unazad** (*backward chaining*).

Metod zaključivanja unaprijed

- ❑ Traži se **uslov pravila** koji odgovara **činjenicama iz baze podataka** i aktivira se odgovarajuće pravilo.
- ❑ Pravila je moguće kompaktnije napisati:

Pravilo 1: $Y \ \& \ D \rightarrow Z$

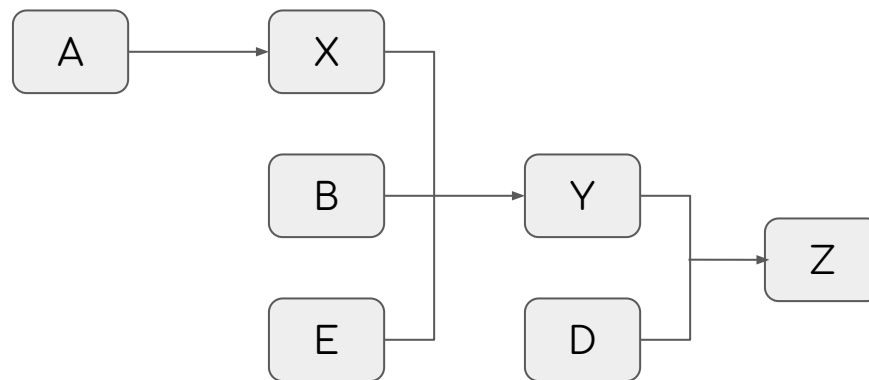
Pravilo 2: $X \ \& \ B \ \& \ E \rightarrow Y$

Pravilo 3: $A \rightarrow X$

Pravilo 4: $C \rightarrow L$

Pravilo 5: $L \ \& \ M \rightarrow L$

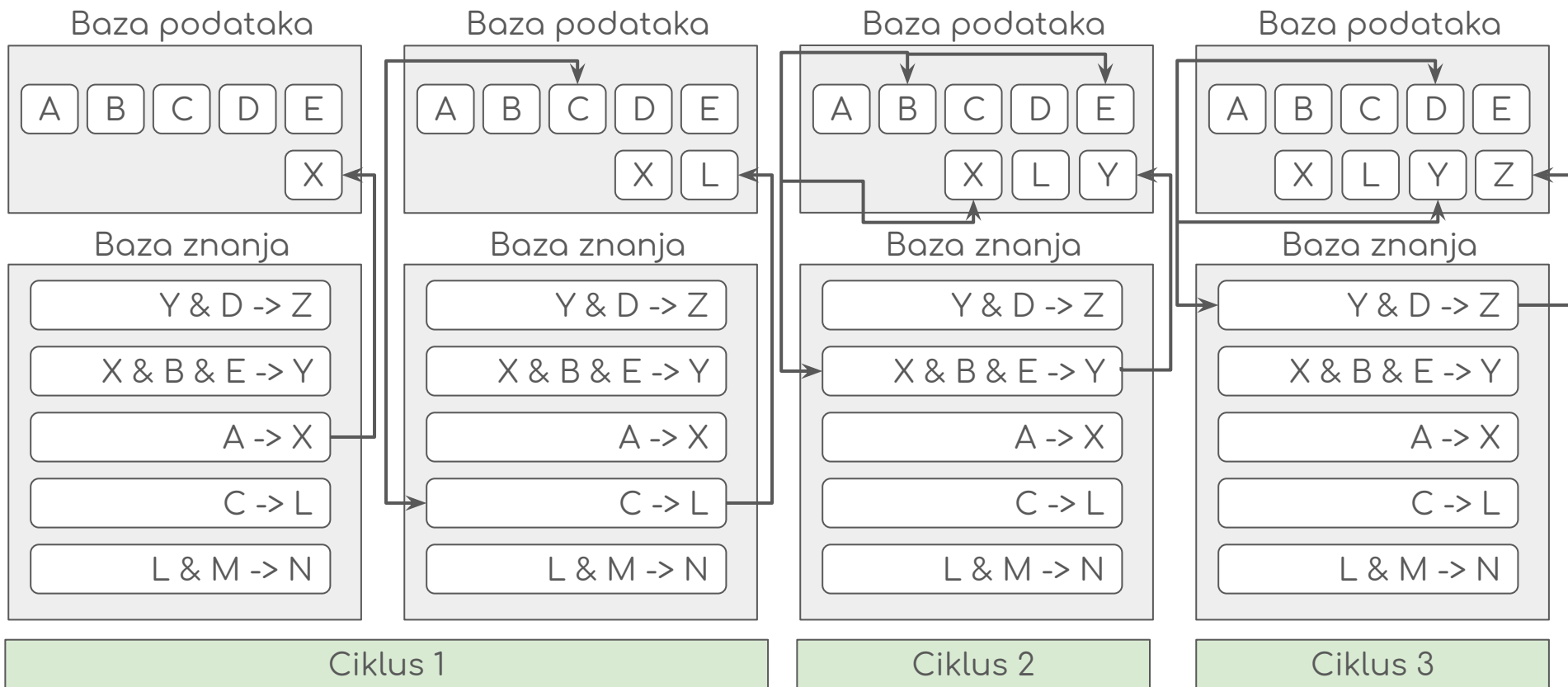
-> implikacija, razdvaja uslov i zaključak



Proces počinje od poznatih činjenica i nastavlja se u skladu sa njihovim sadržajem.

Proces završava ako nema više pravila koja treba primijeniti.

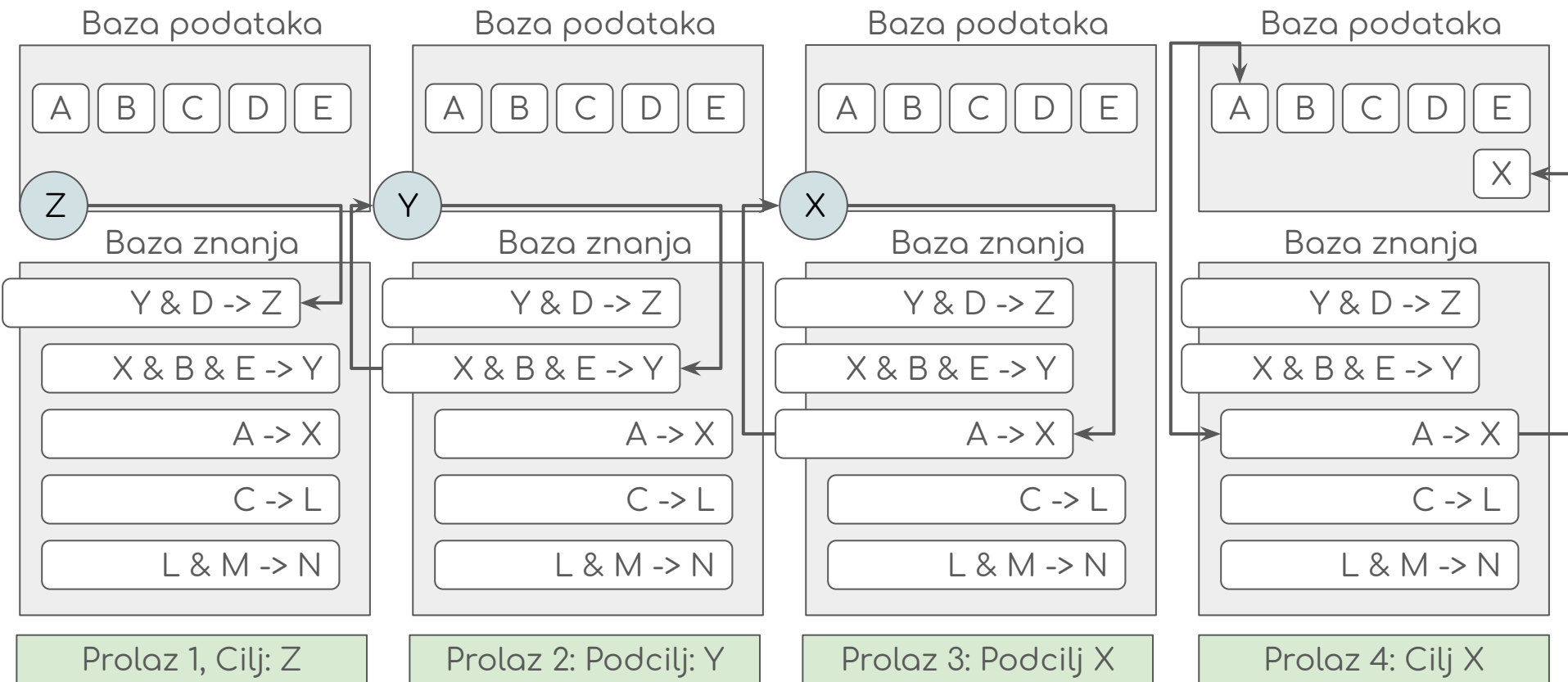
Metod zaključivanja unaprijed



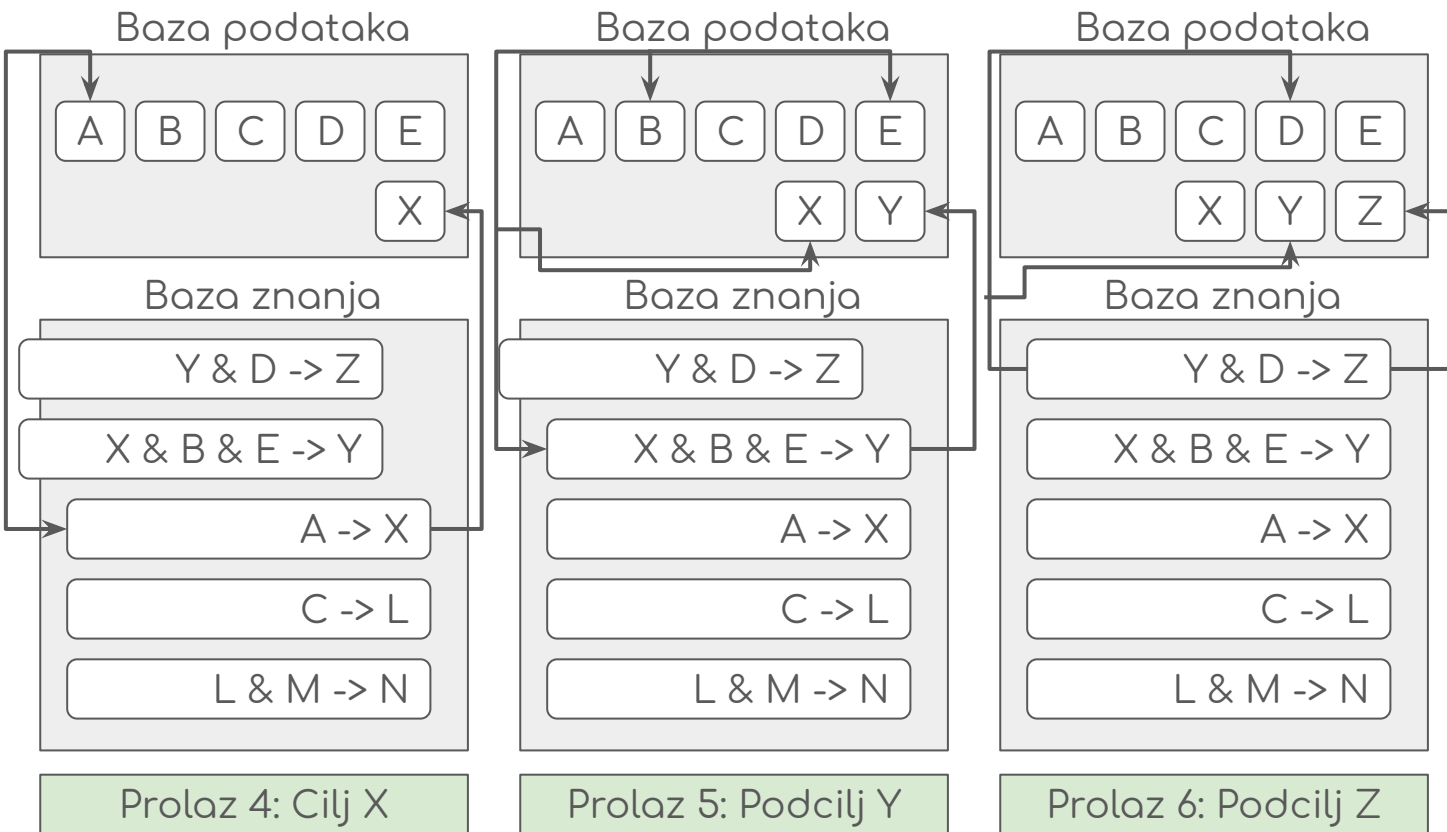
Metod zaključivanja unazad

- ❑ Pretpostavi se zaključak, a mehanizam zaključivanja pokušava da prikupi činjenice koje ga potvrđuju (hipoteza).
- ❑ Mehanizam zaključivanja pretražuje zaključke pravila, u kojima traži pretpostavljenu činjenicu.
- ❑ Ako postoji i ako je primjenjivo u odnosu na činjenice iz baze podataka, pravilo se primijeni i ciljna tvrdnja je dokazana.
 - ❑ U suprotnom, pravilo se smješta u stek i postavlja se novi (pod)cilj da se dokaže istinitost uslova ovog pravila.

Metod zaključivanja unazad



Metod zaključivanja unazad



...

U procesu zaključivanja su upotrebljena samo tri pravila, dok je zaključivanje unaprijed koristilo četiri.

Kada koristiti koji metod?

- ❑ Kad ekspert treba da zaključuje o prethodno prikupljenim informacijama, koristi zaključivanje unaprijed.
- ❑ Kad postavlja hipoteze i traži činjenice koje bi ih potvrdile ili opovrgle, koristi zaključivanje unazad.
- ❑ Ekspertni sistem DENDRAL je za određivanje molekularne strukture na osnovu spektrograma mase (raspoložive informacije) koristio zaključivanje unaprijed.
- ❑ Dijagnostički ekspertni sistemi, kao što je MYCIN, koriste zaključivanje unazad.

Primjer: Kupovina računara

Pravilo 1: IF namjena = obrada teksta
AND nacin upotrebe = na putovanju
THEN tezina = lagan

Pravilo 2: IF namjena = obrada teksta
AND nacin upotrebe = u kancelariji
THEN tezina = nije bitna

Pravilo 3: IF budzet <= 2000
AND budzet > 1000
AND tezina = lagan
THEN model = Dell Latitude

Pravilo 4: IF budzet < 1000
AND tezina = nije bitna
THEN model = Toshiba Satellite

Primjer: Kupovina računara

A) Zaključivanje unaprijed

- a) Osnovna namjena?
 - i) Obrada teksta
 - ii) Komunikacije
 - iii) Multimedija

Odgovor: 1

- b) Način upotrebe?
 - i) U kancelariji
 - ii) Na putovanju

Odgovor: 2

- c) Koliki je budžet?
 - i) <1000
 - ii) 1000..2000
 - iii) > 2000

Odgovor: 2

Sistem preporučuje Dell Latitude
Obrazloženje: lagan, pogodan za obradu teksta i putovanja, zadovoljava budžet

A) Zaključivanje unazad

- a) Koliki je budžet?
 - i) <1000
 - ii) 1000..2000
 - iii) > 2000

Odgovor: 2

- b) Osnovna namjena?
 - i) Obrada teksta
 - ii) Komunikacije
 - iii) Multimedija

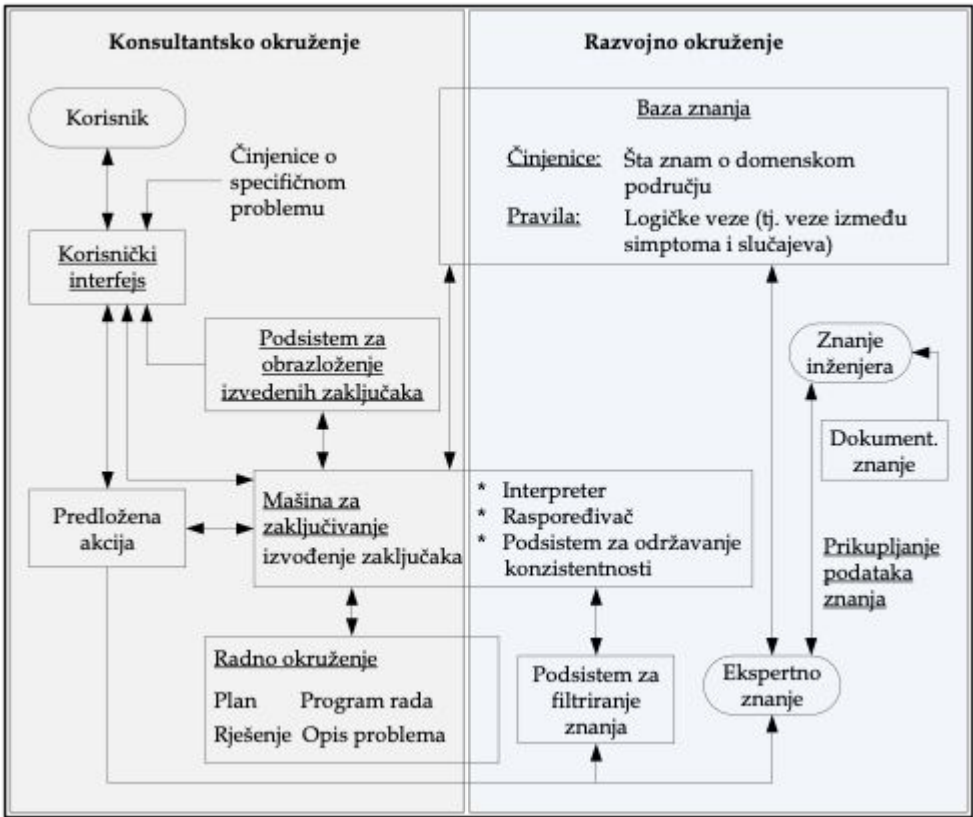
Odgovor: 1

- c) Način upotrebe?
 - i) U kancelariji
 - ii) Na putovanju

Odgovor: 2

Sistem preporučuje Dell Latitude
Obrazloženje: lagan, pogodan za obradu teksta i putovanja, zadovoljava budžet

Prošireni model ekspertnog sistema



Vrste ekspertnih sistema

- ❑ Ekspertni sistem na bazi pravila – *znanje se predstavlja kao serija pravila*
- ❑ Ekspertni sistem bazi okvira (frame) – *koriste relacione strukture za predstavljanje složenih koncepata; pristup objektno orijetnisanom programiranju (slot = set atributa i aspekt = znanje i/ili proceduralna informacija)*
- ❑ Hibridni ekspertni sistem – *kombinacija više tipova ES-a*
- ❑ Sistem na bazi modela – *model kao potencijalni prototip budućeg sistema; npr. za simulaciju strukture i funkcionisanje sistema;*
- ❑ Ekspertni sistemi opšte namjene (*off-the-shelf*), koji su namijenjeni korištenju u široj oblasti primjene;
- ❑ Ekspertni sistemi posebne namjene (*custom-made*), koji su izraženi prema posebnim zahtjevima krajnjih korisnika;
- ❑ Ekspertni sistemi u realnom vremenu – *strikno graničeni na sisteme koji su vremenski zahtjevni u pogledu resursa i očekivanih akcija.*

Alati za razvoj ekspertnih sistema

- ❑ Softverski alati za razvoj ekspertnih sistema su:
 - ❑ programski jezici opšte namene, kao što su C/C++, F#, Prolog i LISP;
 - ❑ ljuske ekspertnih sistema, npr. Exsys CORVID, CLIPS, Jess i Experise2Go;
 - ❑ gotova rešenja za određene šire oblasti primjene, kao što je oblas osiguranja, medicine ili planiranje (Haley, ILOG, LPA VisiRule).
- ❑ Programski jezici opšte namene su algoritamski jezici kao C/C++, Java i C#, Python, funkcionalni jezici kao LISP, Erlang ili F#, kao i jezici za logiko programiranje, kao što je Prolog.

Primjeri u Python-u

```
from random import choice
from experta import *

class Light(Fact):
    """Info about the traffic light."""
    pass

class RobotCrossStreet(KnowledgeEngine):
    @Rule(Light(color='green'))
    def green_light(self):
        print("Walk")

    @Rule(Light(color='red'))
    def red_light(self):
        print("Don't walk")

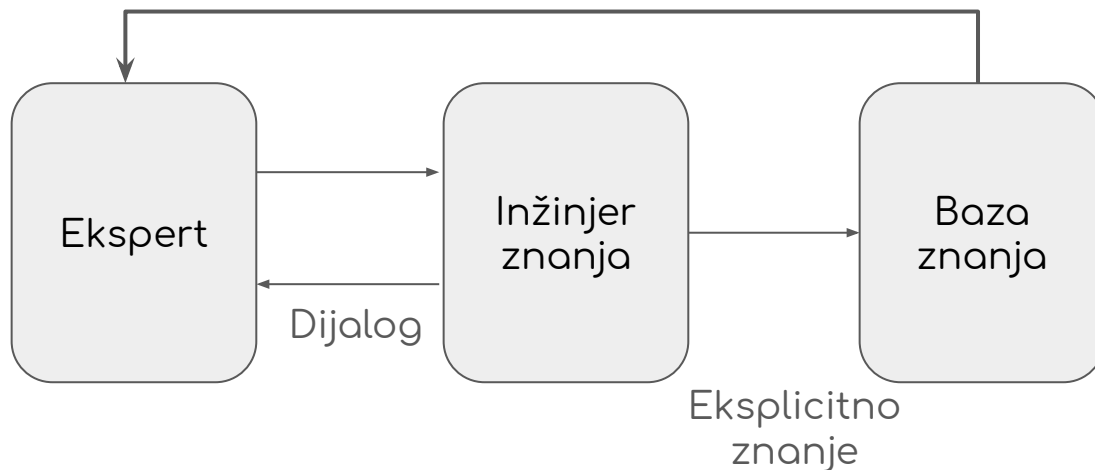
    @Rule(AS.light << Light(color=L('yellow') | L('blinking-yellow'))))
    def cautious(self, light):
        print("Be cautious because light is", light["color"])
```

Primjeri u Python-u

```
>>> engine = RobotCrossStreet()
>>> engine.reset()
>>> engine.declare(Light(color=choice(['green', 'yellow', 'blinking-yellow', 'red'])))
>>> engine.run()
Be cautious because light is blinking-yellow
```

Inženjering znanja

- ❑ Proces izgradnje ekspertskog sistema:
 - ❑ Inženjer znanja uspostavlja **dijalog** sa ekspertom (čovjekom) kako bi pribavio znanje.
 - ❑ Inženjer znanja to znanje eksplicitno kodira u bazi znanja.
 - ❑ Ekspert procjenjuje ekspertni sistem i daje kritiku inženjeru znanja.



Reprezentacija znanja

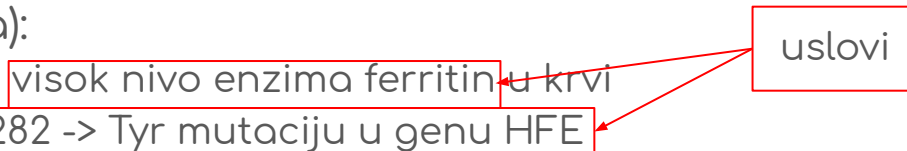
- ❑ Znanje ekspertskog sistema može se predstaviti na više načina, uključujući pravila IF-THEN:

IF si umoran THEN se odmori

Pravila - još jedan način reprezentacija znanja

- ❑ Reprezentacija znanja preko pravila je veoma jednostavna:
 - ❑ **IF-THEN** izrazi:
 - ❑ IF uslov THEN zaključak
 - ❑ **Uslov/akcija** izrazi:
 - ❑ IF uslov THEN akcija
- ❑ Primjer 1:
 - ❑ (IF) Ako je crveno svjetlo na semaforu (AND) i ste stali na semaforu, (THEN) onda možete skrenuti desno.
- ❑ Primjer 2 (biomedicina):
 - ❑ (IF) Ako pacijent ima visok nivo enzima ferritin u krvi (AND) i ako ima Cys282 -> Tyr mutaciju u genu HFE (THEN) onda pacijent ima oboljenje Hemohromatoza.
- ❑ Uslovi (*conditions*): premisa, nezavisna varijabla.
- ❑ Zaključak (*conclusion*): akcija, rezultat, konsekvensa, cilj, ishod, zavisna varijabla.

uslovi



Pravila

❑ Uslov ('IF'):

- ❑ Uslovi su serije testova. Predstavljaju i zamjenu za “stablo odlučivanja” (*decision tree*).
- ❑ Uslovi se kombinuju na način da se primijeni neka logička operacija, npr. AND, OR, itd.

❑ Posljedica ('THEN'):

- ❑ Može biti jedan ili više zaključaka (*conclusions*).
- ❑ Može biti akcija ili više akcija (*actions*).
- ❑ Dodatno pored zaključka ili akcije, može postojati i distribucija vjerovatnoće koja daje informaciju o podršci ili izvjesnosti date posljedice.
 - ❑ Kako donositi zaključke u situaciji kada podaci nose neku vrstu neizvjesnosti (da li će izvršiti ili ne, koliko su “tačni”, itd.)? Postoje posebni sistemi koji se nazivaju “rasuđivanje (rezonovanje) u slučaju neizvjesnosti”.

Posljedice (konsekvence)

- ❑ Posljedice se koriste za:
 - ❑ Dovođenje u relaciju:
 - ❑ IF je rezervoar prazan
THEN auto se ne može pokrenuti.
 - ❑ Davanje preporuke:
 - ❑ IF je jesen
AND je oblačno je
AND prognoza je kiša
THEN savjet je da ponesete kišobran.
 - ❑ Davanje direktive:
 - ❑ IF se auto ne može pokrenuti
AND ako je rezervoar prazan
THEN akcija je “napuni rezervoar”.

Posljedice (konsekvence)

- ❑ Posljedice se koriste za:

- ❑ Donošenje strategija:

- ❑ IF se auto ne može pokrenuti

- THEN provjeri rezervoar;

- Korak1 je završen.

- IF Korak1 završen

- AND rezervoar je pun

- THEN akcija je provjeri bateriju

- Korak2 je završen

- ❑ Heruristiku (model predikcije: klasifikacija/regresija)

- ❑ IF je prosuta tečnost

- AND ako je $Ph < 6$

- AND i miriše na sirće

- THEN je prosuta tečnost kisela.

Propoziciona pravila

- ❑ Primjeri relacija: $<$, $>$, $=$, $'$
- ❑ **Propoziciona pravila:** Pravila koja porede vrijednost atributa sa konstantom (primjer: da li je temperatura $> 28?$):
 - ❑ Imaju istu ekspresivnu snagu kao **propoziciona logika**.
 - ❑ Primjer:
 - ❑ IF je kreditni rejting visok
AND plata $> 15,000$ KM godišnje
AND druge vrijednosti $> 80,000$ KM
AND istorija plaćanja u skladu sa obrascom
THEN odobri kredit do $20,000$ KM i postavi pozajmnicu u kategoriju 'B'.
- ❑ Šta ako problem uključuje zavisnosti među atributima?
 - ❑ Ovakav problem se ne može definisati propozicionim pravilima.
 - ❑ Potrebno je koristiti neku drugu vrstu pravila.