

A thick black L-shaped frame surrounds the text. The top-left corner is a horizontal bar extending to the right, and the bottom-right corner is a vertical bar extending upwards. The text is centered within the open space of the frame.

AGENCY A MULTI- AGENTOVÉ SYSTÉMY

Inteligentní agenty

Obsah přednášky

- Agenty
- Úkolové prostředí
- Typy agentů



AGENTY

Agent (1)

- Def.: Ve větší či menší míře autonomní entita situovaná v adekvátním prostředí, využívající senzorů ke snímání parametrů prostředí a aktuátorů k provádění změn vedoucích k realizaci vlastních cílů.
- V kontextu umělé inteligence se očekává, že se agent bude odlišovat od běžného programu:
 - *Autonomním chováním*
 - *Vnímáním prostředí*
 - *Adaptací na změny*
 - ...
- Tato oblast UI se těší zájmu především od devadesátých let

Agent (2)

- Zjednodušený popis:

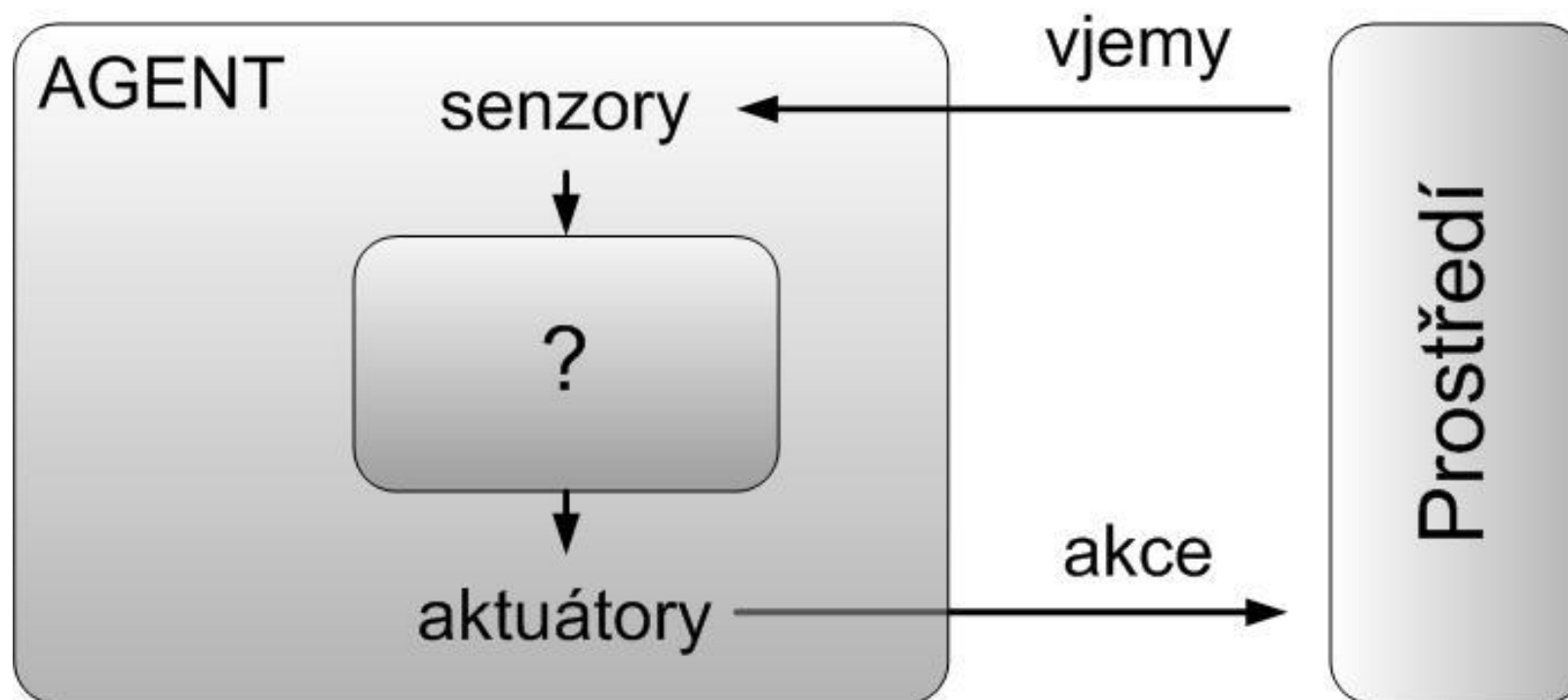
Agent je cokoliv, co vnímá své **prostředí** prostřednictvím **senzorů** a koná v tomto prostředí pomocí **aktuátorů**.

- Obecně budeme předpokládat, že každý agent je schopen vnímat vlastní akce (ale ne vždy jejich následky)
- Autonomie (i částečná)
- Sociální kontext (někdy)

Agent (3)

- Analogie s agentem: **Člověk** má smysly (oči, uši) – tj. senzory, ale také ruce, nohy, ústa (aktuátory)
- **Robotický agent** může mít k dispozici kamery, infračervené čidla (senzory), soustavu motorů (aktuátory)
- **Softwarový agent** získává stisknutí kláves, síťové pakety nebo soubory jako senzorické vjemy a realizuje v prostředí akce zobrazováním údajů na obrazovku, zápisem do souborů a posíláním paketů

Agent a prostředí



Vjemy

- Pojem **vjem** používáme pro označení sensorických vstupů agenta v kterémkoliv okamžiku
- **Sekvence vjemů** je kompletním záznamem všeho co kdy agent vnímal
- Obecně lze říci, že *agent provádí v libovolném okamžiku výběr akcí závisející na celé sekvenci vjemů, které až do tohoto okamžiku zaznamenal*
- Jestliže jsme pro libovolnou sekvenci vjemů schopni specifikovat jakou akci agent zvolí, pak jsme víceméně řekli vše, co o takovém agentu mohlo být řečeno

Funkce a program

- Chování agenta je popsáno **funkcí agenta**, která přiřazuje sekvenci vjemů akci. Jedná se o matematickou abstrakci
- Funkce agenta je pak realizována **programem agenta**. Program agenta je jeho konkrétní implementací
- Program a funkce agenta nejsou totéž

Koncept racionality

- **Racionální agent** je takový agent, který má v rámci své funkce přiřazenu správnou akci pro každý vjem
- „Správnost“ akce se hodnotí podle toho, zda zvyšuje úspěšnost agenta při plnění jeho úkolu
- Je tedy zapotřebí měřit „výkonnost“ agenta

Měření výkonu agenta (1)

- Agent umístěný ve (svém) prostředí přijímá určitou posloupnost akcí podle vjemů, které se k němu dostávají
- V důsledku těchto akcí prochází prostředí různými stavy. Jsou-li tyto stavy žádoucí, pak agent konal správně
- Neexistuje obecný postup jak měřit výkonnost libovolného agenta – vždy záleží na konkrétním prostředí, konstrukci agenta a jeho funkci

Měření výkonu agenta (2)

- Objektivní způsob měření výkonu agenta by měl být součástí jeho návrhu
- *Obecně je vhodnější navrhovat měření výkonu agenta podle toho, co má být v prostředí vykonáno, namísto toho pokoušet se modelovat jak by se agent měl chovat*

Racionalita

- Racionalita agenta v libovolném okamžiku závisí na čtyřech věcech:
 1. *Měření výkonu, což definuje kritérium úspěšnosti agenta*
 2. *Předchozí znalost prostředí (znalosti vložené tvůrcem agenta)*
 3. *Akce, které je agent schopen vykonat*
 4. *Sekvence vjemů, které agent až do aktuálního okamžiku zachytil*

Definice racionálního agenta

- Pro každou možnou sekvenci vjemů vybere **racionální agent** takovou akci, u níž očekává maximalizaci výkonu. Očekávání je podloženo sekvencí vjemů a znalostmi, které má agent vestavěny.
- Posuzování racionality závisí na okolnostech, které nastávají v okamžiku výběru akce
- Pozor na perspektivu vnějšího pozorovatele - neobjektivní

Zlepšování racionality

- **Získávání informací** je přijímání takových akcí, které mají za cíl modifikovat budoucí vjemy (např. zkoumání prostředí)
- **Učení** – agent by měl být schopen přijímat na základě svých zkušeností z interakce s prostředím takové akce, které zvyšují jeho výkonnost, tj. adaptovat se na prostředí. Agent tak přizpůsobuje vložené znalosti.



ÚKLOVÉ PROSTŘEDÍ

Úkolové prostředí (1)

- Zjednodušeně řečeno jsou to „problémy“, pro které agenty představují „řešení“
- **Specifikace úkolového prostředí** musí obsahovat definice těchto parametrů:
 - *Výkon*
 - *Prostředí*
 - *Aktuátory*
 - *Senzory*

Úkolové prostředí (2)

- V literatuře* se tato specifikace označuje akronymem PEAS (Performance, Environment, Actuators, Sensors)
- Při návrhu agenta je zapotřebí specifikovat úkolové prostředí tak podrobně, jak jen je to možné

Příklad PEAS popisu

Typ agenta	Měření výkonu	Prostředí	Aktuátory	Senzory
Řidič taxi	Bezpečnost, rychlost, dodržování zákonů, pohodlí, maximalizace zisku	Cesty, jiná vozidla, chodci, zákazníci	Řízení volantem, plyn, brzda, signalizace, klakson, display	Kamery, sonar, tachometr, GPS, taxametr, akcelerometr, snímání motoru, klávesnice

Další příklady agentů a prostředí (1)

Typ agenta	Měření výkonu	Prostředí	Aktuátory	Senzory
System pro analýzu satelitních snímků	Správná kategorizace snímků	Downlink ze satelitu	Zobrazení kategorizace	Pole barevných pixelů
Robot pro třídění součástek	Procento správně roztríděných dílů	Běžící pás se součástkami, koše	Mechanická paže	Kamera, senzory natočení paže

Další příklady agentů a prostředí (2)

Typ agenta	Měření výkonu	Prostředí	Aktuátory	Senzory
Řídící prvek rafinérie	Maximalizace čistoty, zisk, bezpečnost	Rafinérie, operátoři	Ventily, pumpy, topná tělesa, displeje	Teplota, tlak, chemické senzory
Interaktivní výuka jazyků (tutor)	Maximalizace bodů studenta v testu	Množina studentů, testovací agentura	Zobrazování cvičení, návrhy, opravy	Vstup z klávesnice

Poznámka o prostředí

- „Virtuální“ (softwarové) prostředí hraje stejnou roli jako „reálné“
- Nezáleží na rozdělení prostředí na „opravdové“ a „umělé“, důležitá je složitost vazeb mezi chováním agenta, sekvencí vjemů z prostředí a měřením výkonu
- Některá „reálná“ prostředí jsou celkem jednoduchá – robot třídící součástky
- Některé „virtuální“ prostředí jsou velmi komplexní – simulátory pro výcvik pilotů

Vlastnosti úkolových prostředí (1)

- Pozorovatelnost (relevantních aspektů)
 - *Plně pozorovatelné – např. šachy*
 - *Částečně pozorovatelné – např. taxi*
- Determinismus
 - *Deterministické (další stav prostředí je jednoznačně určen)*
 - *Stochastické (pravděpodobnostní) – např. řízení vozidla*
- Dělení zkušeností
 - *Epizodické (atomické dělení, výběr akcí pouze v rámci 1 epizody) – např. rozpoznání zmetků na pásu*
 - *Sekvenční* (aktuální rozhodnutí ovlivňuje rozhodování v budoucnu) – např. šachy, taxi*

Vlastnosti úkolových prostředí (2)

- Stabilita prostředí
 - *Statické prostředí – např. křižovky*
 - *Dynamické prostředí – např. taxi*
- Kontinuita
 - *Diskrétní – např. šachy*
 - *Kontinuální – např. taxi*
- Četnost agentů
 - *Single-agent*
 - *Multiagent (konkurenční, kooperativní)*

Vlastnosti úkolových prostředí (3)

Úkolové prostředí	Pozorovatelnost	Determinismus	Zkušenosti	Stabilita	Kontinuita	Agenty
Taxi	Část	Prav	Sekv	Dyn	Kon	Multi
Analýza snímků	Plně	Det	Epiz	Semi-dyn	Kon	Sin
Třídění součástí	Část	Prav	Epiz	Dyn	Kon	Sin
Rafinérie	Část	Prav	Sekv	Dyn	Kon	Sin
Výuka jazyků	Část	Prav	Sekv	Dyn	Disk	Multi

The image features two thick black L-shaped brackets. One is positioned on the left side, with its vertical bar extending downwards and its horizontal bar extending to the right. The other is on the right side, with its vertical bar extending upwards and its horizontal bar extending to the left. These brackets frame the central text.

TYPY AGENTŮ

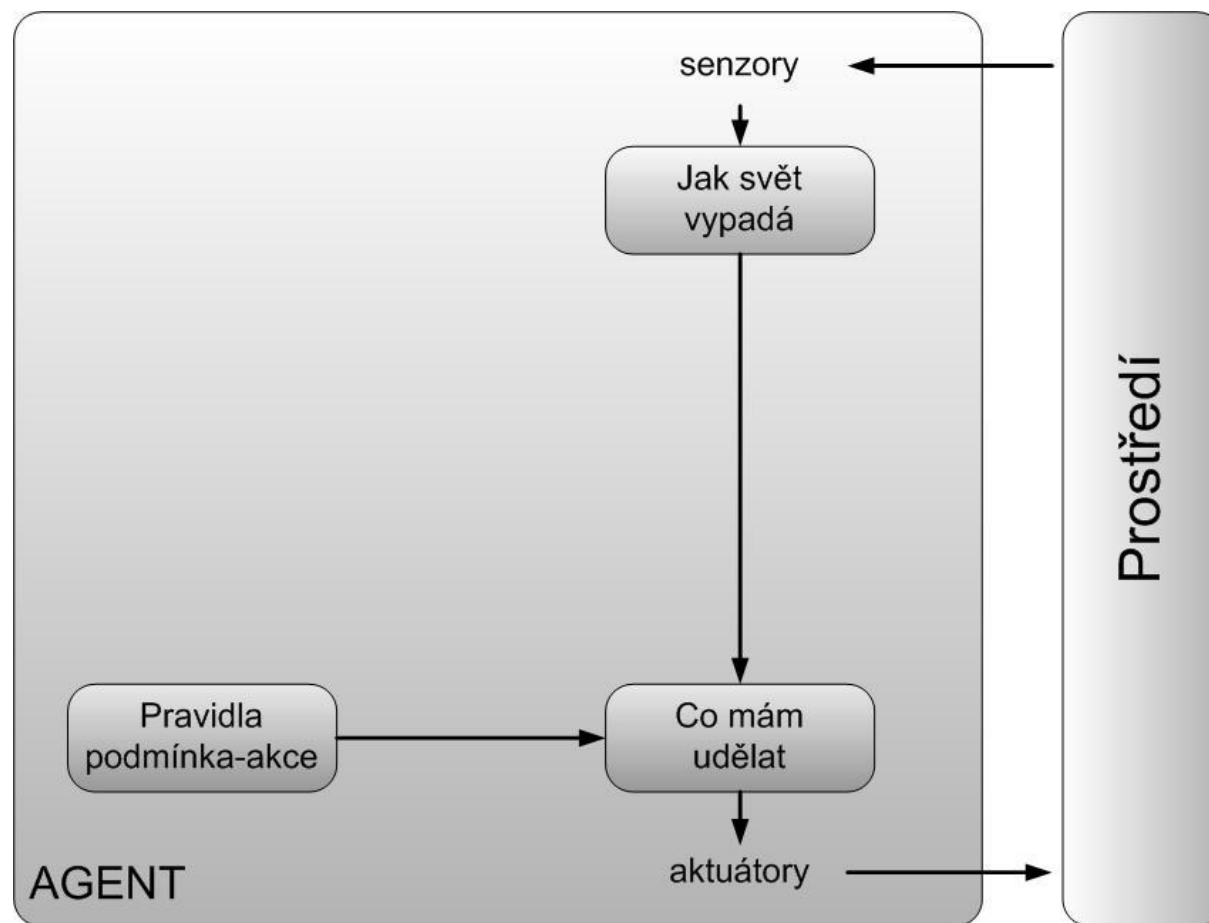
Ryze reaktivní agenty

- Jsou nejjednodušším typem agentů
- Volí akci na základě aktuálního vjemu, ignorují sekvenci dřívějších vjemů
- Propojení mezi vjemy a akcemi je reprezentováno **pravidly podmínka-akce**

KDYŽ(vjem)→PAK(akce)

- Koncept pracuje správně tehdy, jestliže se agent je schopen rozhodovat správně pouze na základě aktuálního vjemu – tedy pouze pokud je prostředí plně pozorovatelné

Ryze reaktivní agent



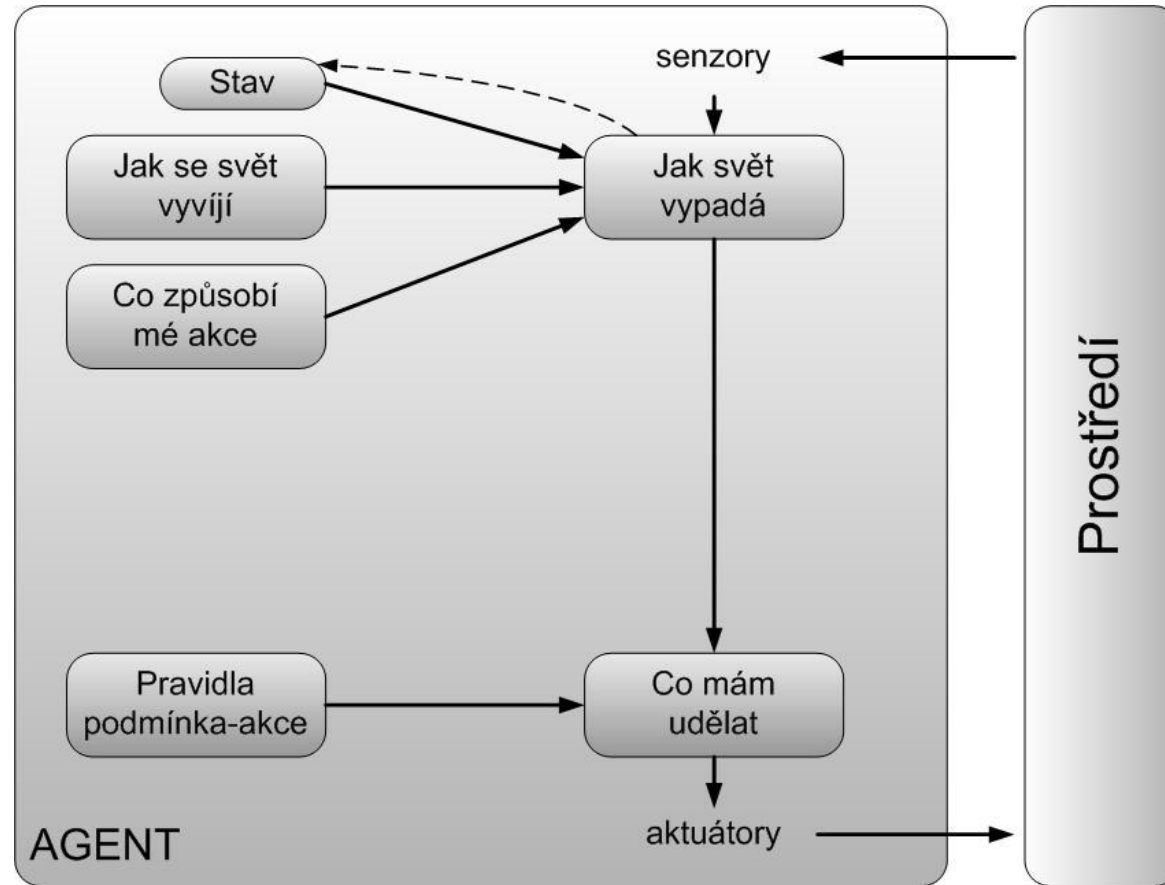
Agenty s vnitřní reprezentací prostředí (1)

- Způsobem jak ošetřit omezenou pozorovatelnost prostředí je zavést **agenta s vnitřní reprezentací světa**
- Takovýto agent si nějakým způsobem uchovává reprezentaci **vnitřního stavu**, která závisí na minulých vjemech a tudíž odráží i některé nepozorovatelné vjemy aktuálního stavu (rozdíly mezi stavy)

Agenty s vnitřní reprezentací prostředí (2)

- Aktualizace vnitřního stavu agenta vyžaduje dva druhy informací:
 - *Jak se svět vyvíjí nezávisle na agentu (část, kterou agent není schopen sledovat)*
 - *Jak akce agenta ovlivňují svět*
- Znalost toho, jak okolní svět pracuje se nazývá **model světa (prostředí)**
- Agenty pracující s takovým modelem světa se nazývají agenty s vnitřní reprezentací prostředí

Reaktivní agent s reprezentací prostředí



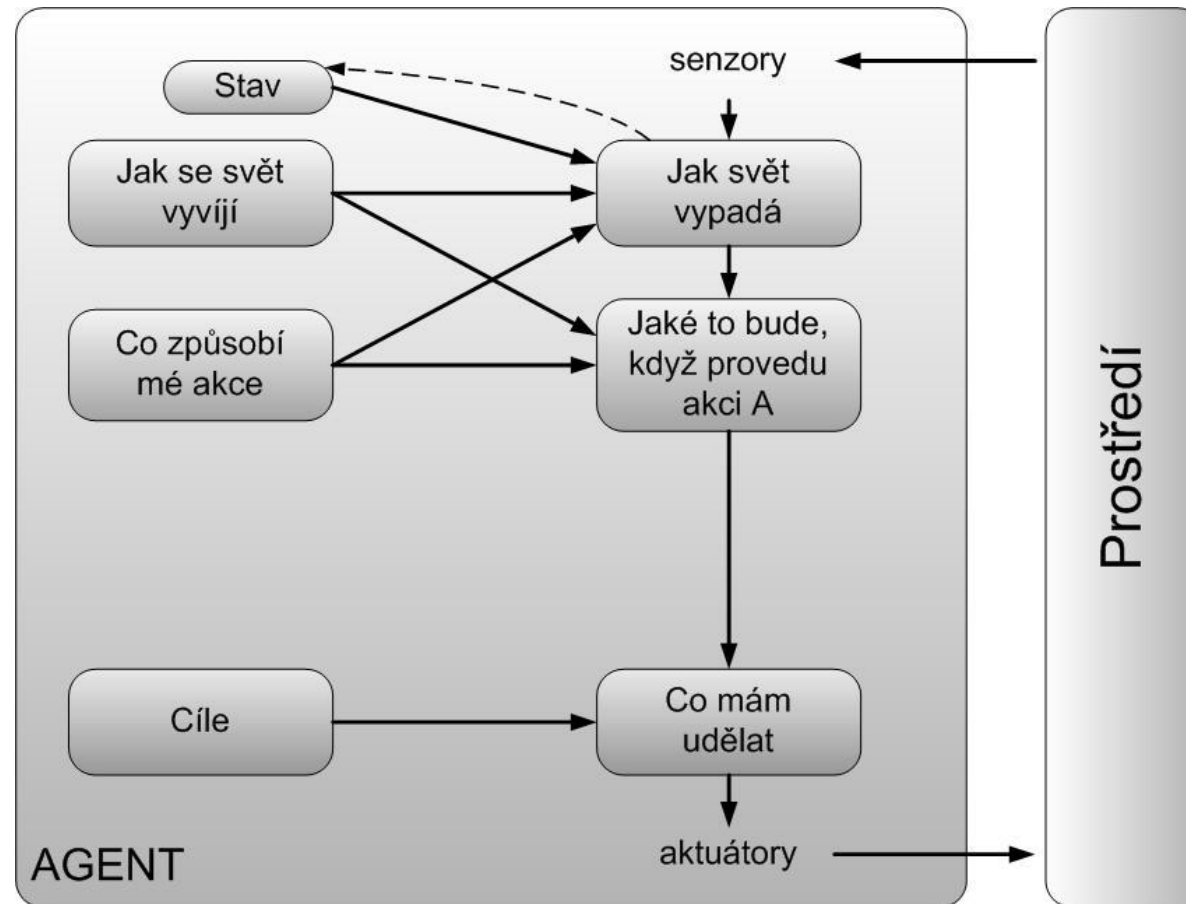
Agenty řízené cílem (1)

- Znalost o aktuálním stavu prostředí nemusí být dostatečná k tomu, aby se agent správně rozhodl co dělat
- Příklad – taxi na křižovatce se má rozhodnout zda zatočí doleva, doprava, či pojedou rovně
- Agent potřebuje nějakou formu **cíle**, informace o tom, čeho se snaží dosáhnout, aby fungoval racionálně

Agenty řízené cílem (2)

- Dosažení cíle bývá někdy přímočaré. Tj. cílový stav navazuje bezprostředně na akci
- V případě, že tomu tak není, musí agent vyhledat sekvenci akcí vedoucí k dosažení cíle. Takový postup se nazývá **prohledávání** (search) a **plánování** (planning)
- Tyto dvě činnosti jsou oblastmi UI, kterým je samostatně věnována velká pozornost

Agent řízený cílem s vnitřní reprezentací prostředí



Agenty řízené užitečností (1)

- Pokud neusilujeme pouze o dosažení cíle, ale záleží nám i na kvalitě rozhodnutí, které k němu vedou, mluvíme o **agentech řízených užitečností**
- Tyto agenty mají definovanu **odměňovací funkci** (utility function), která rozlišuje mezi horším a lepším řešením vzniklé situace
- Je-li některý stav prostředí preferován před jiným, má pro agenta vyšší **užitečnost**

Agenty řízené užitečností (2)

- **Odměňovací funkce** přiřazuje stavům (nebo sekvencím stavů) reálné číslo, které popisuje asociovaný stupeň odměny
- Agent se snaží maximalizovat odměnu, optimalizuje tedy svůj rozhodovací proces v souladu s definovanou odměňovací funkcí
- Je tím vyřešeno rozhodování v situacích, kdy pouhá definice cíle nepostačuje

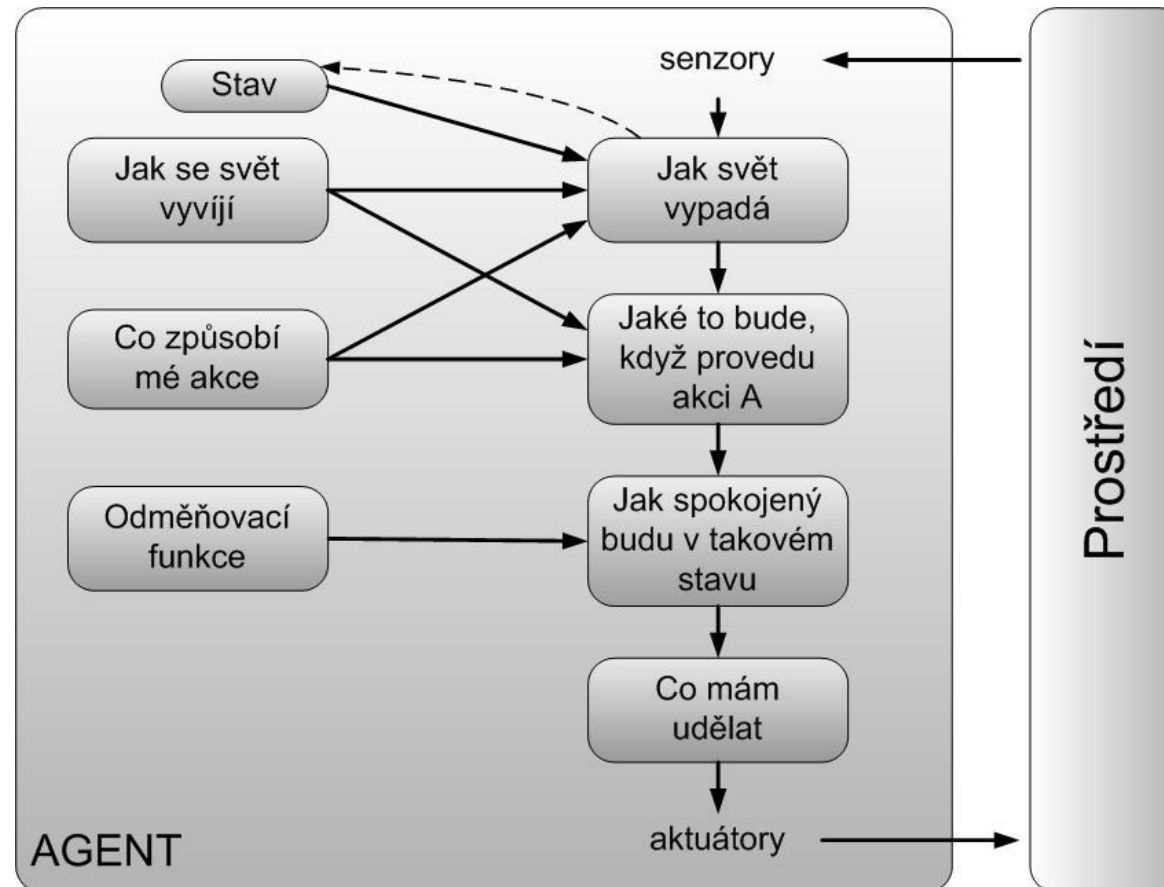
Agenty řízené užitečností (3)

- Racionální agent se obecně má chovat tak, jako by měl definovanou odměňovací funkci, jejíž hodnotu by se snažil maximalizovat
- Agent, který má explicitně definovanou odměňovací funkci je schopen činit racionální rozhodnutí a činí tak obecně použitelným algoritmem, nezávislým na konkrétní podobě odměňovací funkce

Agenty řízené užitečností (4)

- „Globální“ definice racionality – navrhovat funkce racionálního agenta tak, aby měl co nejvyšší výkon - je redukována na „lokální“ plnění omezení, což může být vyjádřeno jednoduchým programem
- Agenty založené na odměňovací funkci jsou schopny pracovat i s neurčitostí v částečně pozorovatelných prostředích

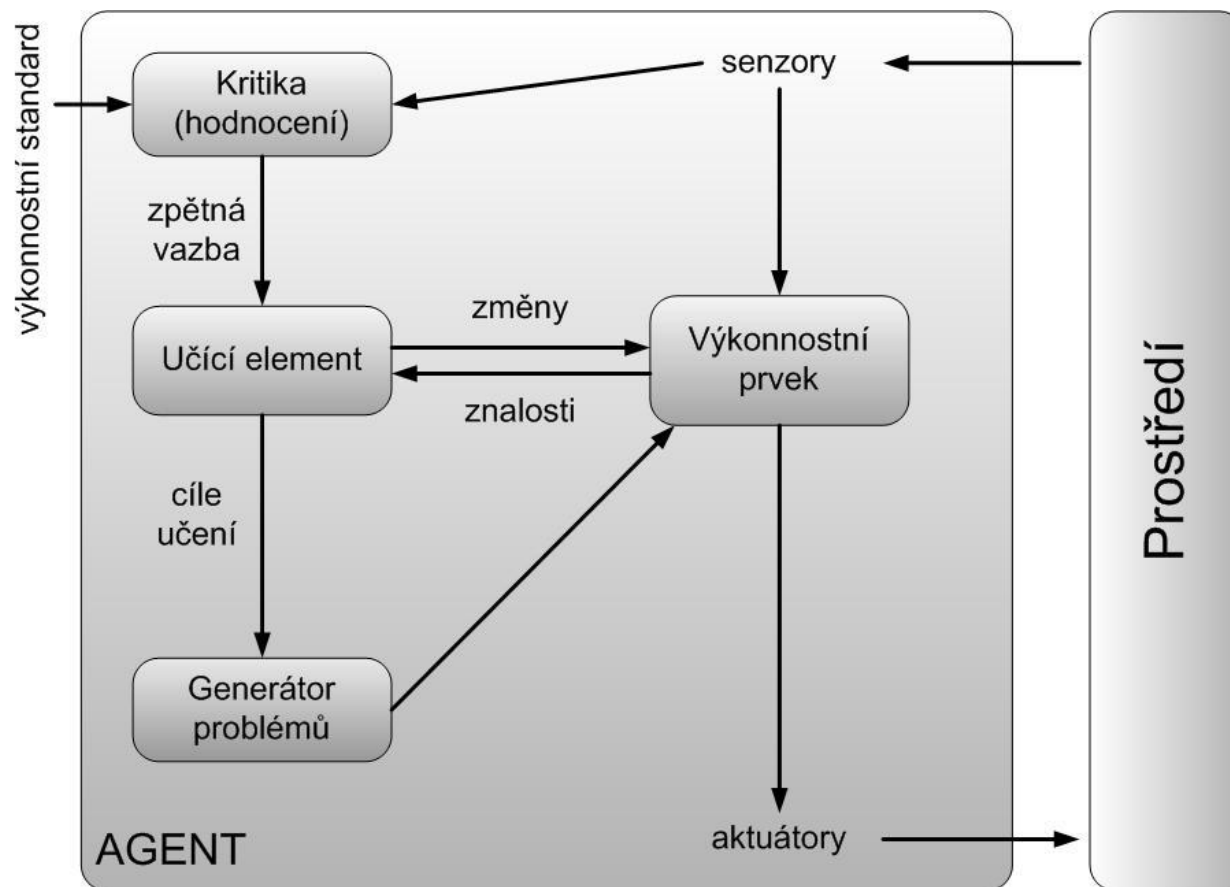
Agent řízený užitečností s vnitřní reprezentací prostředí



Učící se agenty (1)

- **Učení** umožňuje agentu adaptovat se v předem neznámém prostředí a optimalizovat výkon
- Stává se tak kompetentnějším než by mu normálně umožnil soubor znalostí vložených při jeho tvorbě
- **Učící element** je zodpovědný za zlepšení výkonu, **výkonnostní prvek** za jeho realizaci. Tyto dva pojmy je třeba rozlišovat

Učící se agent



Učící se agenty (2)

- Výkonnostní prvek sleduje vjemy a přijímá odpovídající akce
- Učící element využívá **hodnocení (hodnotící element)** k tomu, aby posoudil, jak dobře agent pracuje, a přizpůsobil tomu funkci výkonnostního prvku
- Podoba učícího elementu závisí do značné míry na podobně výkonnostního prvku

Učící se agenty (3)

- Otázka není: „Jak se tohle zvládnou naučit?“
- Otázkou je: „Jaký druh výkonnostního prvku bude agent potřebovat, aby to mohl provést když ví jak?“
- Učící element může být navržen ke každé části agenta

Učící se agenty (4)

- **Generátor problémů** je část agenta odpovědná za navrhování akcí, které vedou k novým, informativním zkušenostem
- Výkonnostní prvek by sám o sobě vykonával pouze akce, které jsou tím nejlepším z toho, co zná
- Je tak odstraněno riziko sub-optimálních řešení

The image features two thick black L-shaped brackets. One is positioned in the top-left corner, and the other is in the bottom-right corner. They are oriented towards each other, framing the central text.

APLIKAČNÍ OBLASTI

Aplikace multiagentových systémů (1)

- Sociologie
- Ekonomie (simulace tržních subjektů)
- Psychologie (Skinner, 1953) – redukce psychologie chování organismů na čistě převedení vstupu/výstupu na podnět/reakci Tzv. **behavioralismus**
- Vojenství
- Teorie řízení

Aplikace multiagentových systémů (2)

- Robotika
- Internetové služby
- Vyhledávání
- Biologie (napodobování živých organismů)
- Simulace a modelování
- Distribuované řešení problémů



DĚKUJI ZA POZORNOST

Použité zdroje

- [1] Russell, S.; Norvig, P.: Artificial Intelligence – A Modern Approach (Prentice Hall, 2003, 2nd Edition)
- [2] Kubík, A.: Inteligentní agenty – tvorba aplikačního software na bázi multiagentových systémů (Computer Press, 2004)