



Systemy Agentowe - wprowadzenie

Mariusz.Matuszek@eti.pg.gda.pl

p.523 WETI PG

<http://www.eti.pg.gda.pl/~mrm>

materiały: <http://infinity.eti.pg.gda.pl/SA/>



Wykład – sprawy organizacyjne

- Obecność nieobowiązkowa, ale mile widziana
- W ciągu semestru możliwe krótkie sprawdziany z zagadnień poruszanych na wykładzie. Punkty z tych sprawdzianów NIE są liczone do podstawy zaliczenia przedmiotu i stanowią dodatkowe koło ratunkowe. Stosowane tylko w razie konieczności.



Laboratorium – sprawy organizacyjne

- (emailem) listy grup według terminów, zawierające:
 - Imię i nazwisko
 - Numer indeksu
 - Email kontaktowy
- Maksimum 18 osób w grupie!



Laboratorium – zaliczenie

- Zadania od tygodniowych indywidualnych, do nieco dłuższych, opracowywanych w podgrupach
- Oddawanie terminowe (2^N)
- Waga punktów z laboratorium: 50%
- Konieczne uzyskanie minimum połowy punktów z zadań aby zaliczyć laboratorium.



Wykład – zaliczenie

- Kolokwium sprawdzające wiedzę pod koniec semestru.
- Waga punktów z kolokwium: 50%
- Konieczne uzyskanie minimum połowy punktów z kolokwium aby zaliczyć wykład.



Zaliczenie przedmiotu

- Uzyskanie minimum 50% punktów z części laboratoryjnej ORAZ z wykładu.
- Osoby powtarzające przedmiot, które posiadają zaliczenie jego części, powinny zgłosić chęć ubiegania się o zwolnienie z odpowiedniego zaliczenia na początku semestru.



Zakres

Systemy Agentowe to rozproszone systemy informatyczne wykorzystujące agenty. Do ich tworzenia wykorzystywana jest wiedza z zakresu:

- Systemów rozproszonych
- Sztucznej inteligencji
- Bezpieczeństwa systemów informatycznych
- Socjologii (wzorce zachowań)



Do zobaczenia za tydzień



Wykład - tematyka


- Definicje agenta i środowiska agentowego
- Cechy zachowań agentów inteligentnych, umocowanie agentów w dziedzinie sztucznej inteligencji
- Wybrane przykłady zastosowań apl. agentowych
- Modele i architektura agentowa
- Wybrane architektury agentów
- Agent BDI – właściwości i zasada działania
- Właściwości algorytmów agentowych i wybrane grupy algorytmów
- Struktura i cykl życia aplikacji agentowych
- Środowiska wykonania i wytwarzania a.a.



Definicja agenta

- Wiele definicji, w zależności od rozpatrywanego zakresu zastosowań.
- Popularna definicja:

Jednostka obliczeniowa (program, robot), która może być widziana jako zdolna do postrzegania stanu środowiska w którym się znajduje, oraz oddziaływania na to środowisko, będąc jednocześnie autonomiczną w tym sensie, że przynajmniej niektóre z jej zachowań są zależne od nabytego przez nią doświadczenia (wiedzy).

A dark silhouette of a tree is positioned in the bottom right corner of the slide, partially overlapping the text area.

Działanie agenta

- Zakłada się, że działania agenta cechują:
 - Inteligencja (w znaczeniu racjonalności i elastyczności)
- Racjonalność i elastyczność uzyskiwane są przez
 - Wnioskowanie
 - Planowanie
 - Uczenie się
 - Rozwiązywanie problemów
 - Podejmowanie decyzji



Cechy wspólne agentów

- Autonomiczność (samodzielność)
- Proaktywność (zdolność do inicjowania czynności)
- Mobilność (zdolność do zmiany położenia – w systemach wykorzystujących agenty mobilne)

uwaga: agent nie jest wirusem – mobilność wymaga wsparcia ze strony *środowiska wykonania*.



Widok agenta w dziedzinie AI

- **Agent** jest czymś zdolnym do działań w ramach środowiska. W szerokim ujęciu agentem jest pies, robak, termostat, robot, człowiek, korporacja, państwo.
- Obiektem badań są sposoby **działań** agenta, czyli co czyni i jak podejmuje decyzje prowadzące do podjęcia czynności. Oceniamy i klasyfikujemy agenty według ich zachowania.
- **Agent inteligentny** to taki agent, który:
 - Działa odpowiednio do okoliczności i swojego celu
 - Adaptuje się do zmieniającego się środowiska i zmian celu
 - Uczy się na podstawie zdobywanego doświadczenia
 - Dokonuje właściwych wyborów, z uwzględnieniem własnych ograniczeń czasowych, przestrzennych i percepcyjnych (zazwyczaj nie widzi całego świata i nie ma nieskończonej pamięci ani czasu na reakcję)

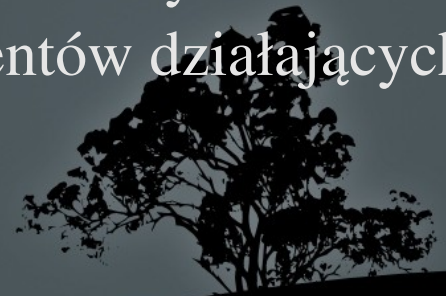


- **Agent obliczeniowy** (computational agent) to agent, którego proces decyzyjny może być wyrażony przy pomocy operacji matematycznych i logicznych, które mogą zostać zaimplementowane.

Głównym celem badań systemów agentowych jest próba zrozumienia reguł tworzących zachowania inteligentne w naturalnych i sztucznych systemach. Badania te obejmują:

- Analizę naturalnych i sztucznych agentów
- Formułowanie i weryfikowanie hipotez o tym, co jest potrzebne do wytworzenia inteligentnego agenta
- Projektowanie, implementacja i badania systemów obliczeniowych wykonujących zadania, wymagające jakiejś formy inteligencji.

Głównym celem inżynierskim jest projektowanie i synteza użytecznych inteligentnych artefaktów – agentów działających inteligentnie – przydatnych w aplikacjach.

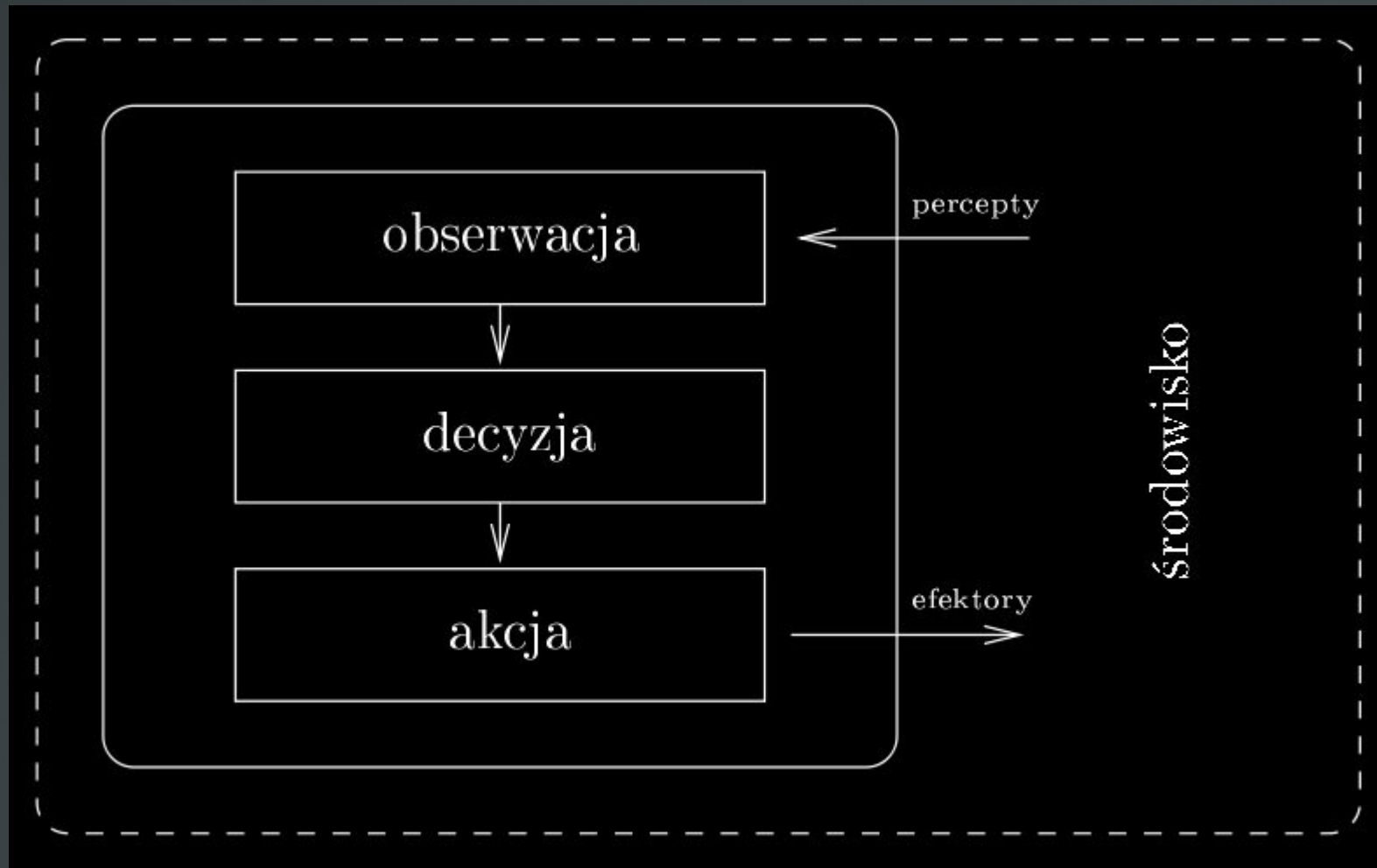


Definicja uogólniona

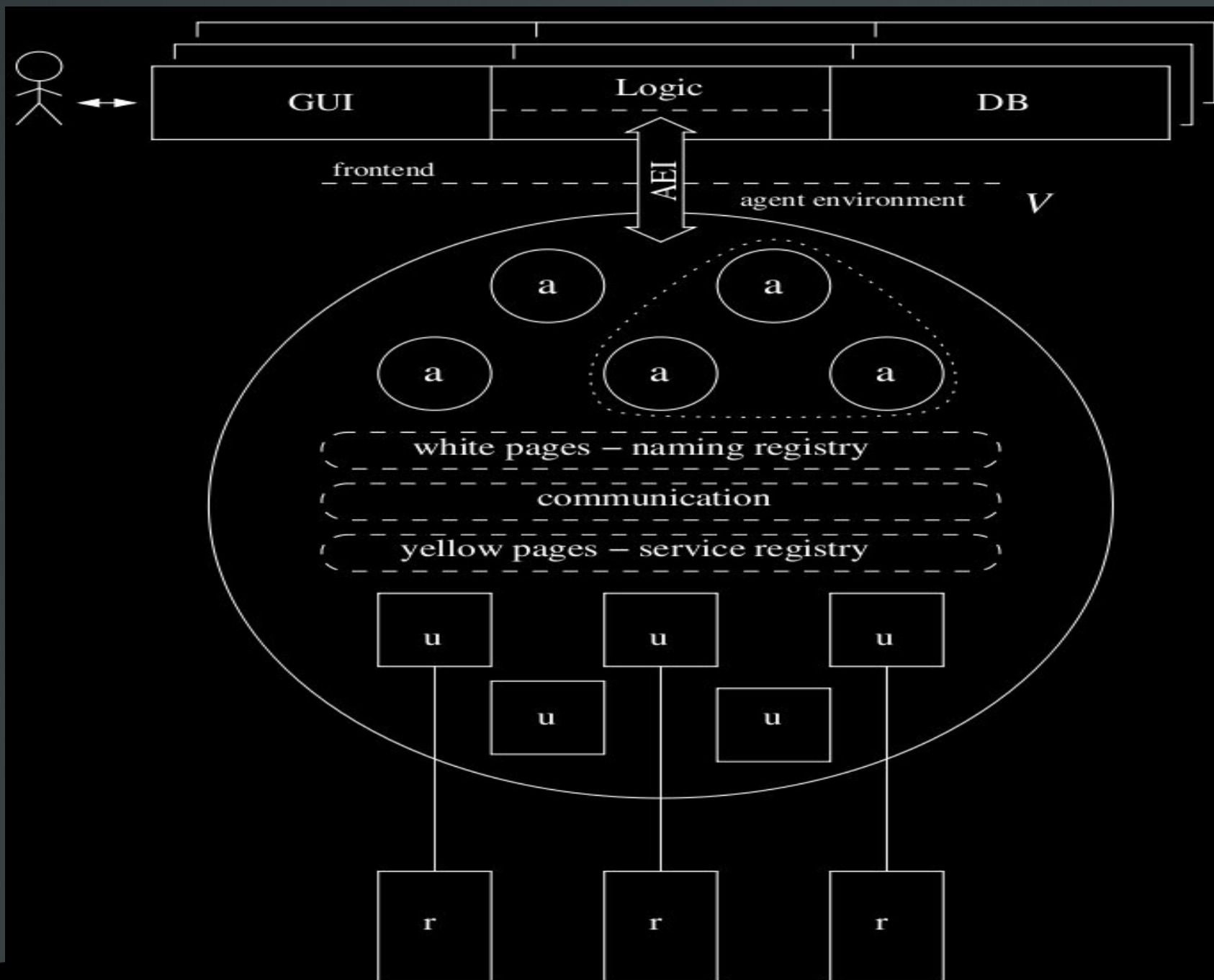
Agent inteligentny jest to system informatyczny (program, robot) osadzony w środowisku, posiadający zdolność do autonomicznego oraz inteligentnego decydowania o swoich akcjach w ramach tego środowiska, w celu spełnienia swojej misji.



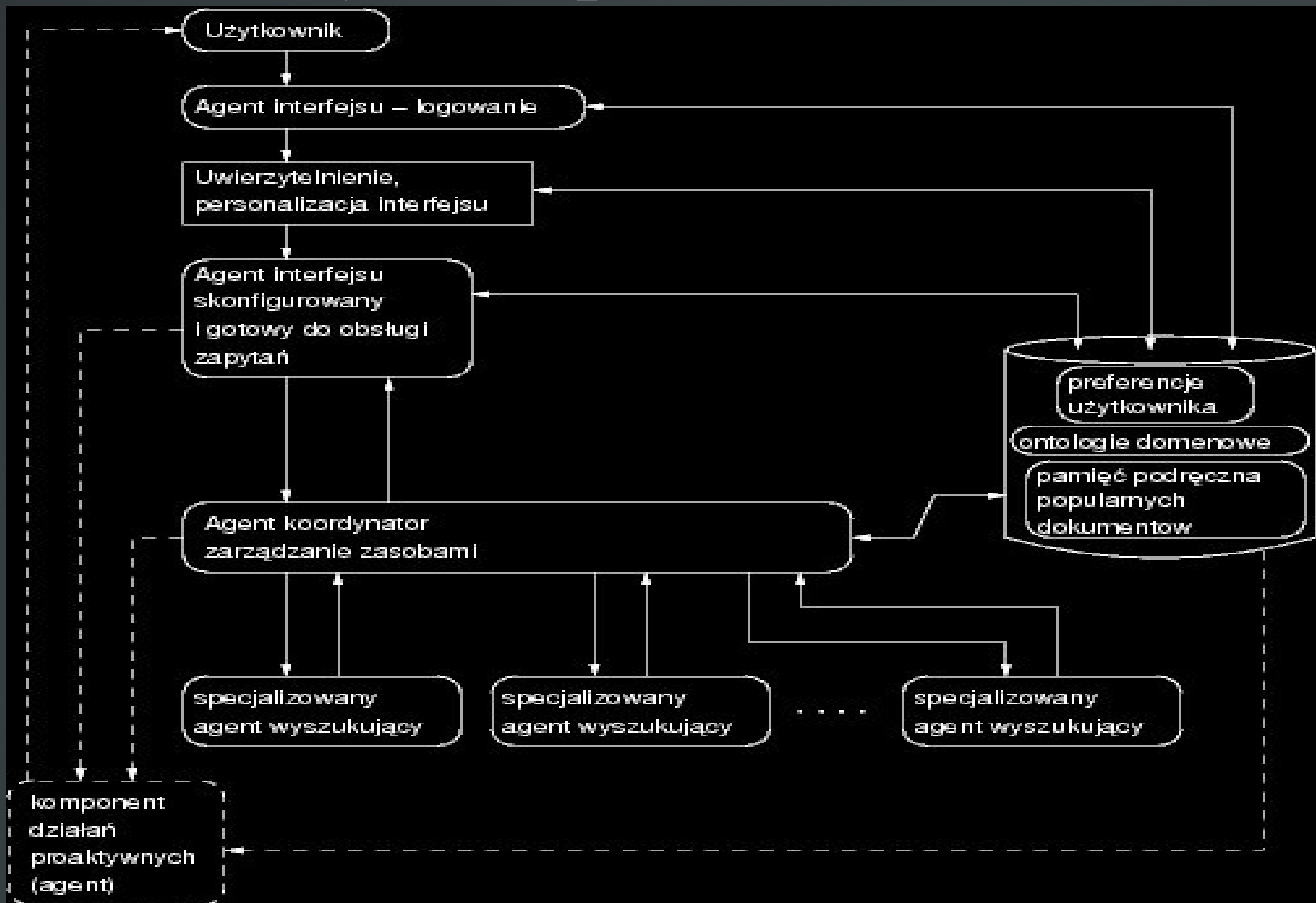
Architektura agenta



Aplikacja agentowa



Przykład aplikacji złożonej



Do zobaczenia za tydzień



Kartkówka:

Imię, nazwisko, nr indeksu

- Taksonomia środowisk agentowych
 - Definicja agenta



Taksonomia środowisk agentowych

- Agent w większości przypadków posiada zestaw funkcji poznawczych (sensorów) odbierających stan otoczenia jak też zestaw akcji (efektorów) pozwalających na dokonywanie zmian w środowisku.
- Zróżnicowanie cech środowiska, w którym osadzony jest agent, będzie miało zasadniczy wpływ na jego konstrukcję.
- Często spotykana klasyfikacja środowisk (cd..)



Dostępne – niedostępne

- Środowisko jest określane mianem dostępnego wtedy, gdy agent może uzyskać pełną, dokładną i aktualną informację o stanie całego środowiska.

Należy zauważyć, że większość złożonych środowisk osadzenia agenta jest niedostępna. W szczególności świat człowieka czy Internet są przykładami środowisk niedostępnych.



Deterministyczne – niedeterministyczne

- Środowisko jest deterministyczne wtedy, gdy dana akcja podejmowana przez agenta przynosi zawsze takie same rezultaty przy dowolnej liczbie powtórzeń – nie występuje niepewność co do stanu środowiska osiąganego w wyniku wykonanej przez agenta akcji.

(świat człowieka jest mocno niedeterministyczny)



Bezkontekstowe - kontekstowe

- Środowisko jest bezkontekstowe (epizodyczne) wtedy, gdy każda z akcji podejmowanych przez agenta nie jest powiązana z akcjami wykonywanymi wcześniej. Agent dokonuje wyboru akcji do wykonania wyłącznie w oparciu o bieżący stan środowiska (zadanie do wykonania) nie uwzględniając możliwych interakcji z akcjami (i ich wynikami) wcześniejszymi bądź przyszłymi.



Statyczne – dynamiczne

- Środowisko jest uznawane za statyczne wtedy, gdy nie zachodzą w nim zmiany inne niż te spowodowane działaniami agenta.
- W środowisku dynamicznym mają także miejsce zmiany niezależne od działań agenta, spowodowane procesami zachodzącymi w samym środowisku bądź nieskoordynowanymi działaniami innych agentów.



Dyskretne – ciągłe

- W środowisku dyskretnym istnieje ustalona, skończona liczba możliwych stanów przyjmowanych przez środowisko (postrzeganych przez agenta) oraz skończona liczba różnych akcji (i ich stanów wynikowych) możliwych do wykonania przez agenta.

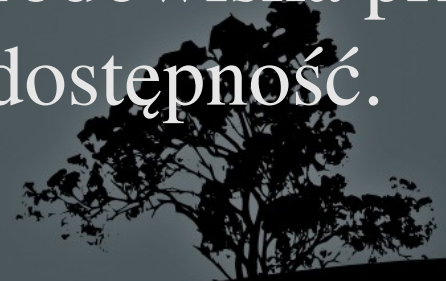
Przykładem takiego środowiska może być gra planszowa o ustalonych regułach.



podsumowanie

Najbardziej złożonym środowiskiem będzie śr.
niedostępne, niedeterministyczne, kontekstowe,
dynamiczne i ciągłe.

Należy zauważyć, że dla większości realnych
środowisk deterministyczność środowiska nie jest
istotnym uproszczeniem problemu, gdyż
decydującą rolę w pojmowaniu środowiska przez
agenta będzie odgrywać jego niedostępność.

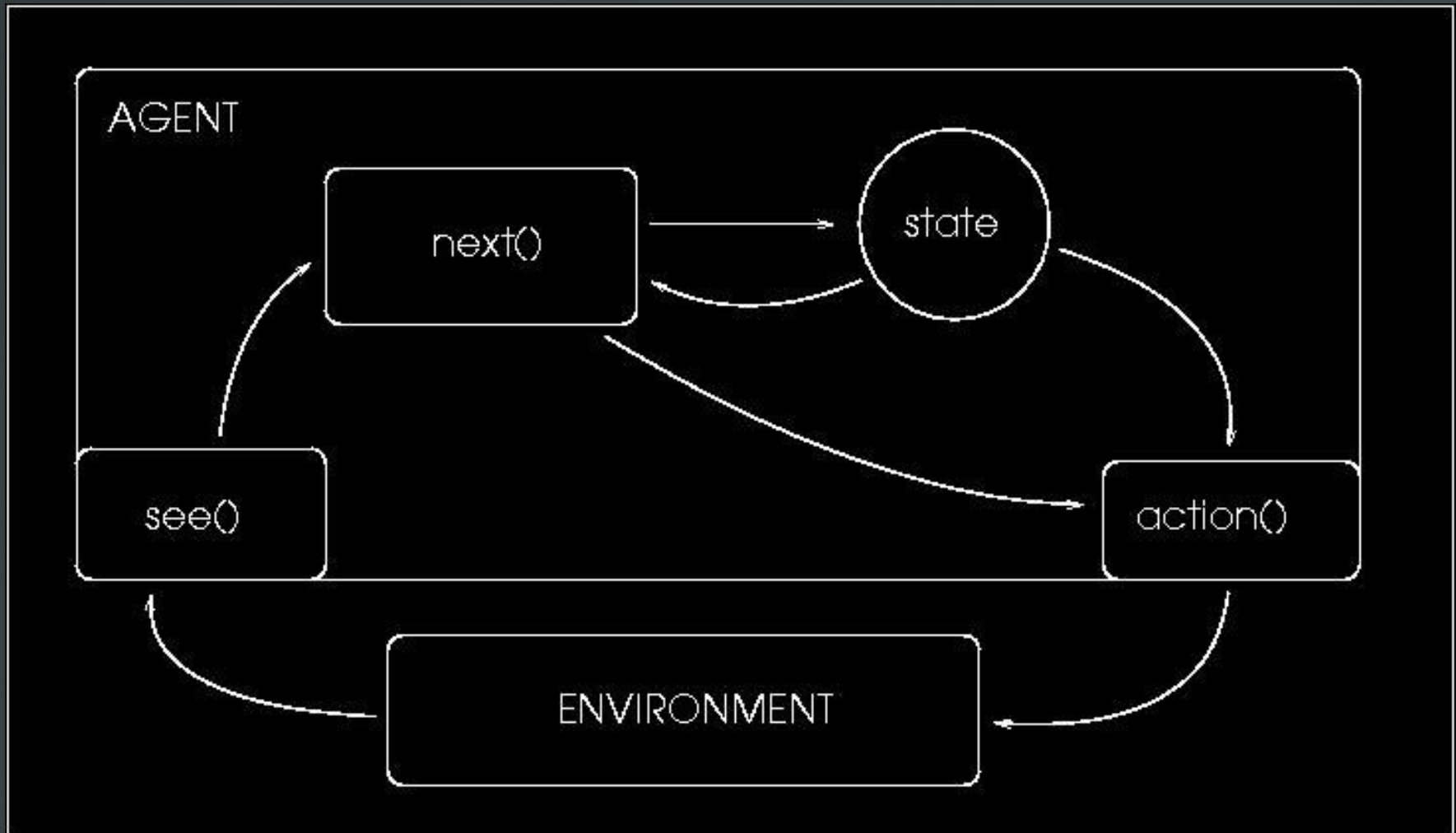


Architektury agentów inteligentnych

- Oparte na logice formalnej (np. BDI)
- Reaktywne (np. mikrotezy)
- Warstwowe
 - Pozioma
 - Pionowa jednoprzebiegowa
 - Pionowa dwuprzebiegowa



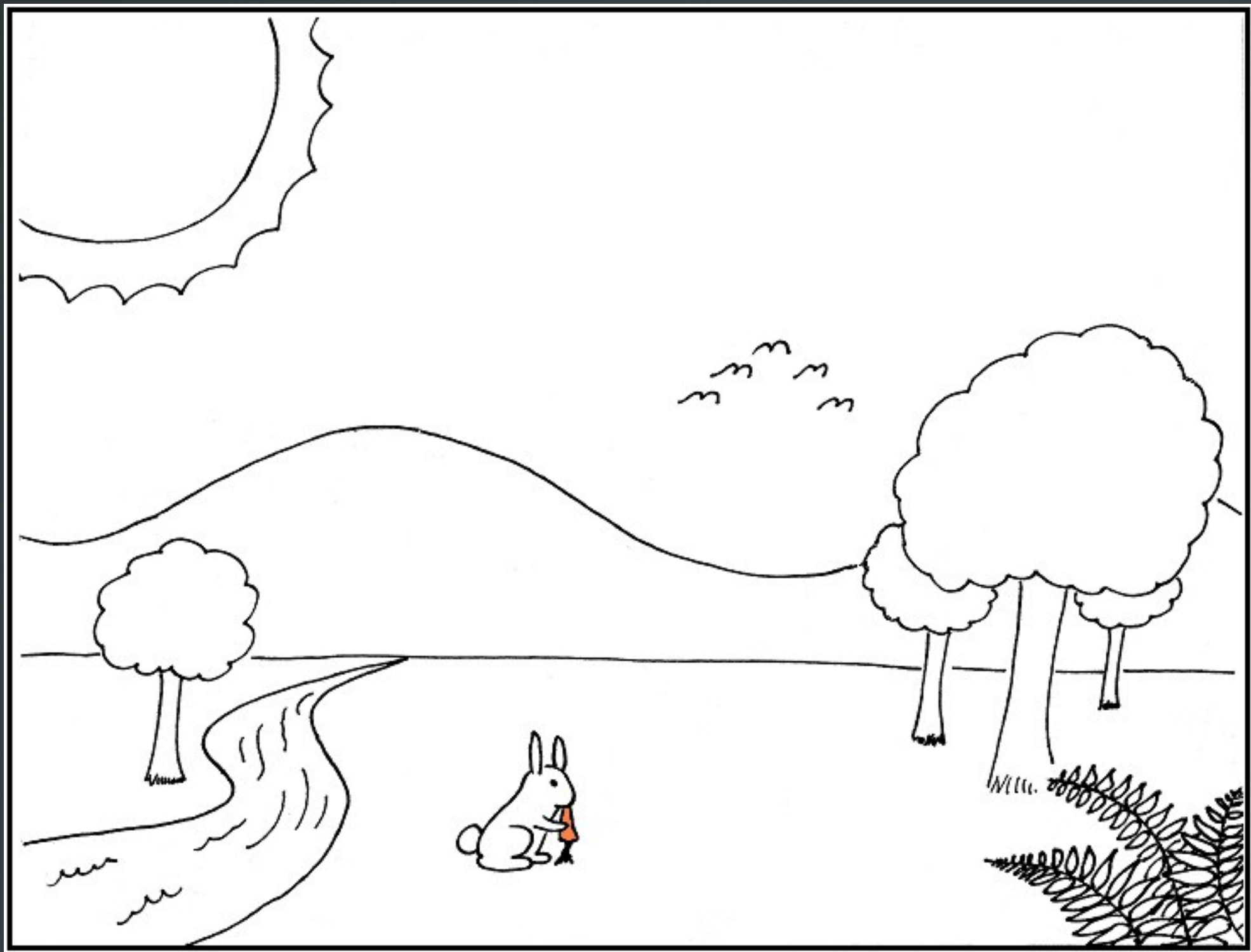
Architektura stanowa agenta

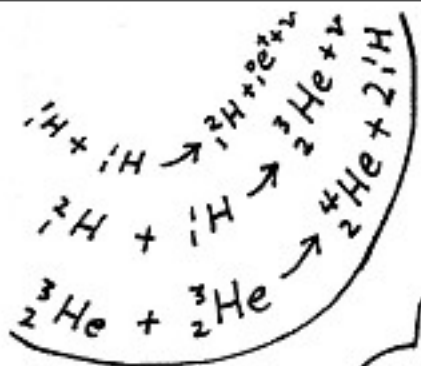


decyzje oparte o logikę formalną

- Pierwsze, najbardziej tradycyjne podejście
- Zakłada, że możliwe jest uzyskanie inteligentnego zachowania agenta poprzez przedstawienie cech jego otoczenia i zasad pożądanego zachowania w postaci symbolicznej (formuł logicznych), a następnie dokonywania syntaktycznych manipulacji tą reprezentacją (dedukcja, dowodzenie).







$$\nabla \cdot E = \frac{1}{\epsilon_0} \rho$$

$$\nabla \cdot B = 0$$

$$\nabla \times E + \frac{\partial B}{\partial t} = 0$$

$$F = G \frac{m_1 m_2}{r^2}$$

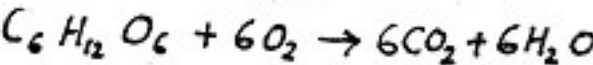
$$R_{\mu\nu} - \frac{1}{2} R g_{\mu\nu} = 8\pi G T_{\mu\nu}$$

$$\nabla_\mu \epsilon_0 \frac{\partial E}{\partial t} = \mu_0 J$$

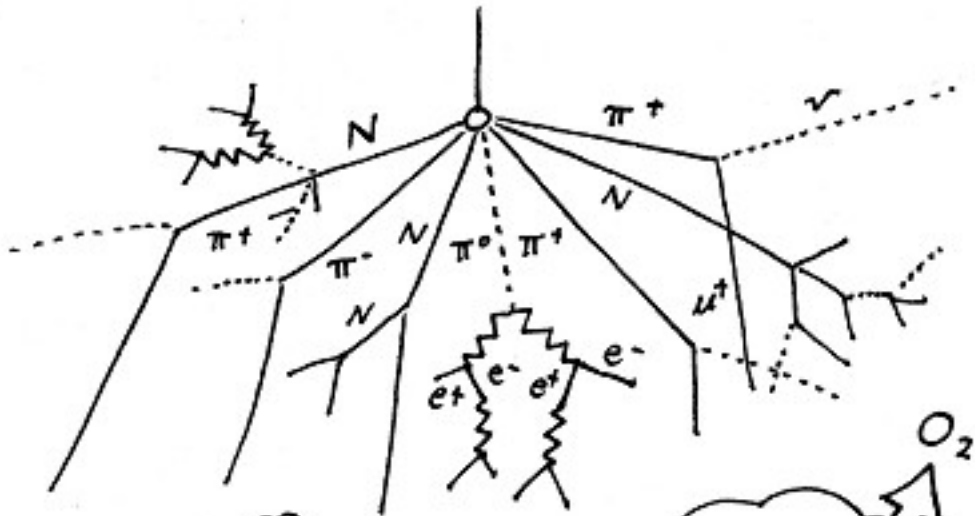
$$f(x) = a_0 + \sum_{n=1}^{\infty} (a_n \cos nx + b_n \sin nx)$$

$$\left[\frac{\hbar^2}{2m} \nabla^2 + V \right] \psi = i\hbar \frac{\partial}{\partial t} \psi$$

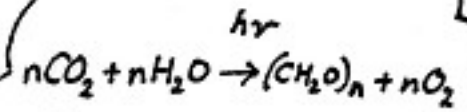
$$\frac{\partial}{\partial t} \mu_i + \sum_{j=1}^n \mu_j \frac{\partial \mu_i}{\partial x_j} = 0$$



$$\begin{aligned}
 f_1(x,y) &= \begin{bmatrix} 0.95 & 0.04 \\ -0.04 & 0.95 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0.15 \\ 0.26 \end{bmatrix} \\
 f_2(x,y) &= \begin{bmatrix} 0.15 & 0.26 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0.95 & 0.04 \\ -0.04 & 0.95 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0.15 \\ 0.26 \end{bmatrix}
 \end{aligned}$$



$$P + \frac{1}{2} \rho v^2 + \rho gh = C$$



CO₂

O₂



金吉博

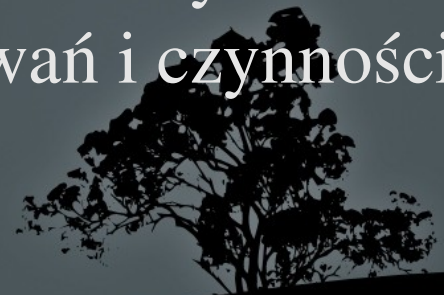
Zalety i wady

- Zalety
 - Bardzo eleganckie podejście
- Wady
 - Niepraktyczne
 - Brak gwarancji reakcji
 - O czasie
 - Właściwej
 - Trudno znaleźć mapowanie perceptów na język logiki formalnej



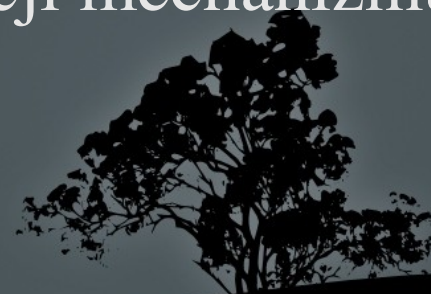
Reaktywne podejmowanie decyzji

- Antidotum na ułomności stosowania logiki formalnej:
 - Odrzucenie reprezentacji otoczenia i zachowań agenta w postaci symbolicznej
 - Rezygnacja z manipulacji syntaktycznych
 - Założenie, że inteligentne działanie nie istnieje w próżni, lecz jest wynikiem interakcji z otoczeniem
 - Założenie, że zachowanie inteligentne wyłania się z interakcji wielu prostych zachowań i czynności.



Architektura mikrotez (subsumption)

- Dwie cechy charakterystyczne
 - Implementacja procesu decyzyjnego agenta w formie zbioru zachowań zorientowanych na zrealizowanie konkretnej czynności – funkcji przetwarzających percepty w sposób ciągły i sygnalizujących możliwość wykonania skojarzonych z nimi akcji
 - Możliwość jednoczesnego sygnalizowania możliwości wykonania zachowania przez wiele funkcji → wymaga implementacji mechanizmu priorytetyzowania zachowań.



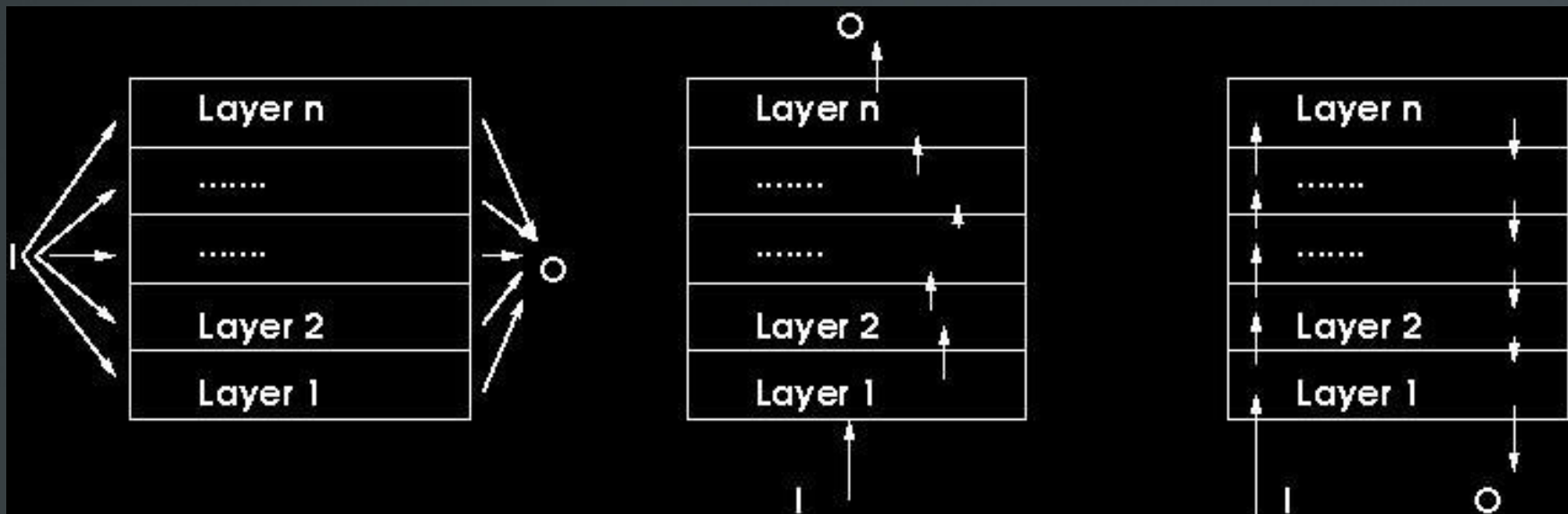
Architektury warstwowe

- Próba dekompozycji procesu decyzyjnego na specjalizowane podsystemy
- Podsystemy uporządkowane w hierarchiczne warstwy, pomiędzy którymi zachodzą interakcje
- Minimum dwie warstwy
 - Reaktywna
 - Proaktywna



c.d.

- Architektura pozioma i architektury pionowe



Do zobaczenia za tydzień



Systemy wieloagentowe – protokoły interakcji międzyagentowej

- Komunikacja
- Koordynacja
- Kooperacja („3 x K”)
- Sieć kontraktowa
- Systemy tablicowe



Protokoły z definicji, to

- Sformalizowane sekwencje komunikatów
- W przypadku, gdy agenty mają rozbieżne cele, protokoły mają na celu maksymalizację użyteczności agentów.
- W przypadku agentów mających cele zbieżne (kooperujących) lub rozwiązujących wspólny problem protokoły służą do zapewnienia globalnej koordynacji systemu bez ingerowania w autonomię agenta.



Protokoły (komunikacja) c.d.

Wspólny język i opis świata (bytów, pojęć)

ONTOLOGIA !



Zastosowania protokołów kooperacji

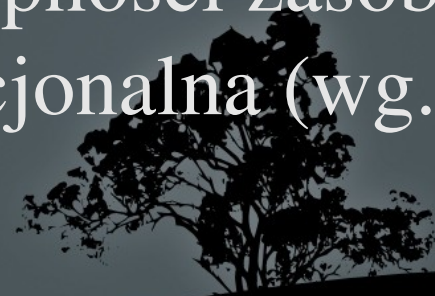
- Najważniejsze zastosowania protokołów kooperacji to:
 - Ustalenie wspólnego celu
 - Ustalenie wspólnego zadania
 - Unikanie sytuacji konfliktowych
 - Utrzymywanie wspólnej wiedzy, stanu, itp.



Strategia kooperacji (1)

Podstawową strategią, współdzieloną przez większość protokołów kooperacji jest dekompozycja zadania na podzadania. Podejście 'dziel i rządź' może pozwolić na użycie prostszych agentów, o mniejszych możliwościach. Pojawia się problem miejsca i rodzaju dekompozycji:

- Projektant \leftrightarrow wykonanie on-line (dekompozycja i planowanie hierarchiczne)
- Przestrzenna (wg. Rozkładu i dostępności zasobów i punktów decyzyjnych) \leftrightarrow funkcjonalna (wg. możliwości agentów).



Strategia kooperacji (2)

Po dekompozycji, zadania mogą być przydzielane według kryteriów uwzględniających:

- Unikanie przeciążania ważnych zasobów
- Możliwości (zdolności) agenta
- Stan wiedzy o zagadnieniu (agent z 'szeroką perspektywą' przydziela zadania innym agentom)
- Nakładanie się zadań
- Odległość semantyczną lub przestrzenną (koszty komunikacji i synchronizacji)
- Priorytetyzację zadań (i ewentualne dynamiczne zmiany przydziału zadania)



Mechanizmy przydziału zadań

Najczęściej stosowane:

- Mechanizmy rynkowe (zadania są dobierane do agenta przez uogólnioną zgodę lub wzajemny wybór)
- Sieć kontraktowa
- Agent planujący
- Struktura organizacyjna (stałe przypisanie zadań)



Sieć kontraktowa

Protokół interakcji kooperacyjnej nakierowanej na rozwiązywanie problemów/zadań. Modelowany według procesu stosowanego przez firmy (analogia do zamówień publicznych). Agent zgłaszający problem do rozwiązania zwany jest proponentem (w lit. ang. manager) a agent oferujący rozwiązanie zwany jest oferentem (ang. contractor).



Perspektywa proponenta (manager)

- Rozgłoszenie problemu do rozwiązania
- Odebranie i ocena zgłoszonych propozycji
- Zawarcie kontraktu z najlepszym oferentem
- Odbiór i wykorzystanie wyników



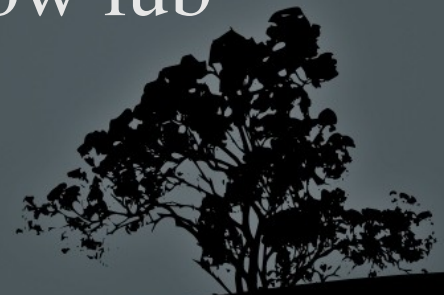
Perspektywa oferenta (contractor)

- Odebranie rozgłoszenia problemu
- Ocena własnych możliwości rozwiązania problemu
- Odpowiedź (rezygnacja albo złożenie oferty)
- Wykonanie zadania, jeśli kontrakt został przydzielony
- Odesłanie wyników



Sieć kontraktowa (2)

- Role oferenta i proponenta nie są znane z góry ani nie są przydzielane na stałe
- Możliwość dalszej dekompozycji zadania: oferent otrzymujący kontrakt na wykonanie złożonego zadania może dokonać jego dekompozycji i stać się proponentem propozycji dla podwykonawców
- Zapewnia łagodny upadek systemu (przez poszukiwanie innych wykonawców lub dekompozycję)



Systemy tablicowe

Metafora: "grupa specjalistów zgromadzona wokół pustej tablicy. Rozwiązywanie problemu rozpoczyna się gdy ktoś wpisze na tablicę problem i dane początkowe. Specjaliści obserwują tablicę i gdy mają coś do zaoferowania zapisują to na tablicy. Ta dodatkowa kontrybucja może pozwolić innemu specjalistcie na dodanie kolejnego fragmentu rozwiązania. Proces jest powtarzany aż do uzyskania rozwiązania problemu".



Systemy tablicowe - charakterystyka

- Niezależność wiedzy eksperckiej
 - Specjaliści są autonomiczni i działają niezależnie od konfiguracji innych zgromadzonych przy tablicy ekspertów (źródeł wiedzy)
- Różnorodność implementacji – wewnętrzne szczegóły implementacji ekspertów są ukryte.
- Brak ograniczeń formy zapisów na tablicy
- (ale) Wspólny język przedstawiania wiedzy na tablicy.
Wybór między ogólnością a precyzją.



Systemy tablicowe – charakterystyka(2)

- System zorientowany na zdarzenia
 - Zdarzenia na tablicy
 - Dodanie, usunięcie, zmiana zapisu
 - Zdarzenia zewnętrzne
 - Eksperci wpisują na tablicę zaistnienie zdarzeń, które rozpoznali / zinterpretowali
- Konieczność zachowania kontroli
 - specjalizowany komponent (rodzaj moderatora dopuszczający do głosu kolejnych ekspertów)
- Inkrementalne wypracowywanie rozwiązań



Negocjacje międzyagentowe

Sytuacja, gdy w systemach agentowych poszczególne agenty mają różne cele często prowadzi do klasy interakcji zwanych negocjacjami.

Negocjacja jest to proces, w którym agenty – każdy posiadający indywidualny cel lub zadanie - wypracowują wspólną decyzję.



W procesie negocjacji agenty wpierw komunikują swoje stanowiska, które mogą być rozbieżne. Następnie próbują wypracować porozumienie przez czynienie ustępstw albo szukanie alternatyw.

Podstawowymi cechami negocjacji są:

1. język negocjacji,
2. protokół wymiany komunikatów,
3. proces decyzyjny agenta, ustalający jego stanowisko, ewentualne ustępstwa i kryteria porozumienia.



Cechy idealnych negocjacji

Wydajność – agent nie powinien marnować zasobów podczas osiągania porozumienia.

Stabilność – osiągnięte porozumienie powinno być wiążące dla agentów. Przyjęta strategia nie powinna zachęcać do odejścia od porozumienia.

Prostota – mechanizm negocjacji powinien mieć niskie zapotrzebowania (obliczenia, pasmo).

Rozproszenie – proces decyzyjny nie powinien być scentralizowany.

Symetria – nie faworyzujący żadnego agenta.



Przykład zorientowany zadaniowo

Cel: pobrać z sieci wymagane dokumenty (lista dokumentów) minimalnym kosztem.

Jeśli niektóre dokumenty się powtarzają, agenty mogą zaoszczędzić pobierając dokument raz.

(przez koszt możemy rozumieć spożytkowanie dowolnego zasobu, np. pasma, tokenów...)



1. Każdy agent rozgłasza nazwy potrzebnych dokumentów.
2. Dokumenty zidentyfikowane jako wspólne przypisywane są jednemu z agentów (na przykład rzutem moneta)
3. Każdy agent pobiera swoje dokumenty płacąc za nie.
4. Agenty udostępniają sobie dokumenty wspólne.

Algorytm jest prosty, symetryczny, rozproszony i wydajny. A czy jest stabilny?



Zagadnienie stabilności negocjacji

- Istnieją trzy domeny, w których występuje zagadnienie stabilności negocjacji
 - Domena zysku,
 - Domena wspólnego stanu,
 - Domena zadaniowa.
- Poruszany już problem podziału pracy przy pobieraniu wielu dokumentów należy do domeny zadaniowej.



Domena zadaniowa (zorientowana zadaniowo) występuje, gdy agenty mają zbiór zadań do wykonania, wszystkie zasoby niezbędne do wykonania zadań są dostępne oraz agenty mogą wykonać swoje zadania samodzielnie, bez pomocy albo przeszkód ze strony innych agentów. Jednakże, agenty mogą osiągnąć korzyści współdzieląc zadania.

Rozpatrując stabilność algorytmu wspólnego pobierania dokumentów należy rozważyć strategię działania agenta.



W omawianym przypadku optymalną strategią postępowania agenta jest rozgłoszenie prawdziwego zbioru potrzebnych dokumentów, niezależnie od strategii, jaką w tej kwestii obiorą pozostałe agenty i od dokumentów potrzebnych innym agentom. Ponieważ w takiej sytuacji nie ma zachęty do odejścia od tej deklaracji (i wynikających z niej zobowiązań), algorytm jest stabilny.

Strategie negocjacji opracowuje się dla konkretnego przypadku (aplikacji) i środowiska działania agentów. Brak ogólnych recept na 'najlepszą strategię'. Są jednak dwa ogólne podejścia...

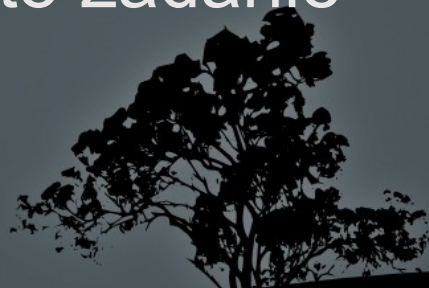


Każde z podejść czyni pewne założenia co do rodzaju negocjujących agentów. Podejście pierwsze stosuje się do agentów logicznych wykorzystujących akty mowy wyrażone przez logikę formalną. Przykład:

$$\forall x (x \neq y)$$

$$\neg (Precommit_a y x \emptyset) \wedge (Goal y Eventually (Achieves y \emptyset)) \wedge (Willing y \emptyset) \\ \Leftrightarrow (Intend y Eventually (Achieves y \emptyset))$$

Należy rozumieć: agent y podejmuje się zadania jeżeli (1) nie zobowiązał się wobec innego agenta x do przejęcia i wykonania zadania, (2) sam ma na celu osiągnąć to zadanie i (3) chce osiągnąć to zadanie indywidualnie.



Podstawą podejścia drugiego jest założenie o ekonomicznej racjonalności agentów. Dodatkowo, zbiór agentów musi być nieduży, agenty muszą posługiwać się wspólnym językiem, wspólnymi regułami abstrakcji (opisu problemu) i muszą wypracować wspólne rozwiązanie (*deal*).

Wspólne rozwiązanie ma postać planu działania spełniającego wszystkie cele agentów.

Użyteczność (zysk) rozwiązania dla agenta to różnica między wartością celu dla agenta a kosztem jego osiągnięcia. Każdy agent próbuje zmaksymalizować swój zysk.



Wypracowując rozwiązanie agenty tworzą i poddają dyskusji *zbiór negocjacyjny* zawierający wszystkie rozwiązania mające dodatni zysk dla każdego z agentów. Formalnie, domena zorientowana zadaniowo staje się krotką:

$$\langle T, A, c \rangle$$

Gdzie T jest zbiorem zadań, A jest zbiorem agentów a $c(X)$ jest funkcją określającą koszt wykonania zadania X .




Porozumienie (deal) jest redystrybucją zadań.
Użyteczność (zyskowność) porozumienia d dla agenta k określa wzór:

$$U_k(d) = c(T_k) - c(d_k)$$

Porozumienie sprzeczne D występuje, gdy agenci nie mogą wypracować porozumienia.

Porozumienie d jest indywidualnie racjonalne gdy $d > D$. Porozumienie d jest optymalne pareto gdy nie występuje porozumienie $d' > d$. Zbiór wszystkich porozumień indywidualnie racjonalnych i pareto optymalnych tworzy zbiór negocjacji NS. Możliwe są trzy sytuacje:



- Konflikt: zbiór NS jest pusty.
- Kompromis: indywidualnie agent preferuje wykonanie zadania samodzielnie, ale ponieważ 'nie jest sam' (uczestniczy w negocjacjach) realizuje wypracowane porozumienie.
- Kooperacja: gdy wszystkie elementy zbioru NS są preferowane przez każdego z agentów do wykonania wspólnego.



Modele koordynacji agentów w środowiskach sieciowych

Agenty mobilne posiadają kilka zalet wobec tradycyjnego modelu aplikacji sieciowej:

- Oszczędność pasma przez przemieszczanie się do potrzebnych zasobów.
- Posiadają wbudowany kod potrzebny do pracy z lokalnymi zasobami – nie wymagają więc pośrednictwa zdalnego serwera / zewnętrznych usług.
- Mogą kontynuować pracę także przy zaniku komunikacji sieciowej (interakcje lokalne – potrzeba koordynacji działań).
- Współpracują w sposób naturalny z systemami mobilnymi.



Taksonomia modeli koordynacji

		Temporal	
		coupled	uncoupled
Spatial	coupled	direct	blackboard based
	uncoupled	meeting oriented	PageSpace



Model bezpośredni (direct)

- Komunikacja (zazwyczaj) punkt-punkt
- Adresaci nazwani explicite
- Wzajemna zgoda w kwestiach:
 - Protokół komunikacji
 - Formaty danych
- Zazwyczaj implikuje sprzężenie temporalne – komunikacja podlega synchronizacji
- Niezbyt dobry w rozległych aplikacjach internetowych



Model spotkań (meetings)

- Interakcja w ramach spotkania bez jawnego nazywania partnerów
- Znane a priori punkty spotkań
- Punkty spotkań otwarte stale lub wedle potrzeb

Punkty spotkań można traktować jako 'obiekty synchronizujące' a komunikację porównać do modelu 'event generator / listener'

- Agenty muszą posiadać przynajmniej wspólną wiedzę o nazwach punktów spotkań



Model tablicowy (blackboard)

- Współdzielone przestrzenie danych używane jako repozytoria komunikatów (zapis/odczyt)
- Interakcje nie są temporalnie sprzężone
- Ponieważ agenty muszą być zgodne co do wspólnych identyfikatorów wiadomości, komunikacja jest sprzężona przestrzennie
- Najczęściej tablice są lokalne dla węzła środowiska (unikanie problemów skalowalności)



Model przestrzenny (PageSpace)

- Przestrzenie (repozytoria) krotek
- Dostęp przez mechanizmy asocjacyjne
(dopasowanie do niepełnego wzorca)
- Rozwinięcie modelu tablicowego
- Pełna rozłączność temporalna i przestrzenna



Etyka maszyn

- Trzy (+0) prawa robotyki:
 - (Robot nie może szkodzić ludzkości lub, przez niepodjęcie akcji, zezwolić na powstanie szkody)
 - Robot nie może uszkodzić człowieka lub, przez niepodjęcie akcji, zezwolić na powstanie szkody
 - Robot musi słuchać poleceń człowieka, za wyjątkiem sytuacji będących w sprzeczności z w/w zasadami
 - Robot musi chronić własny byt, za wyjątkiem sytuacji będących w sprzeczności z w/w zasadami



W ujęciu agentowym, prawa robotyki można zaadaptować do następujących reguł:

1. Agent nie powinien szkodzić misji czynnościami lub zaniechaniami
2. Z wyjątkiem konfliktu z 1. agent nie powinien szkodzić uczestnikom misji
3. Z wyjątkiem konfliktu z w/w agent nie powinien szkodzić sobie
4. Z wyjątkiem konfliktu z w/w agent powinien racjonalnie postępować w kierunku osiągnięcia celów misji
5. Z wyjątkiem konfliktu z w/w agent powinien postępować zgodnie z przyjętą konwencją (wzorcem zachowań)
6. Z wyjątkiem konfliktu z w/w agent powinien racjonalnie postępować w kierunku osiągnięcia własnych celów
7. Z wyjątkiem konfliktu z w/w agent powinien działać wydajnie



Systemy rozproszone są podatne na zakleszczenia (ang. deadlocks).

- Zastosowanie siedmiu reguł postępowania eliminuje zagrożenie zakleszczenia, ponieważ sytuacja taka stoi w sprzeczności z zasadą nr 6



Zagadnienia współpracy międzyagentowej

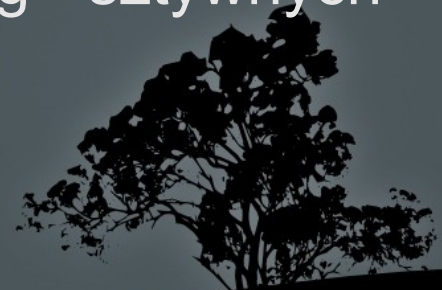
W jaki sposób osiągnąć zadowalającą pracę
systemu wieloagentowego.

- Agenty w systemie wieloagentowym mogą być zaprojektowane przez różnych projektantów, posiadać różne cele. Interakcje takich agentów bardziej przypominają *grę* wymagającą *strategii* niż klasyczną współpracę znaną z systemów rozproszonych. Strategia prowadzi do osiągnięcia najbardziej preferowanego przez agenta wyniku.



- ponieważ agenty działają *autonomicznie*, podejmują *decyzje* o akcjach w czasie działania, a nie podążają za logiką zaszytą 'na sztywno' w kodzie;
- powyższe implikuje konieczność posiadania przez agenty zdolności do dynamicznego koordynowania swoich akcji i kooperowania z innymi agentami.

Na zasadzie kontrastu: w klasycznych systemach rozproszonych kooperacja i koordynacja zachodzi według sztywnych reguł opracowanych w fazie projektu systemu.



Współpraca osiągana jest przez:

- współdzielenie zadań
- współdzielenie informacji
- dynamiczną koordynację działań agentów



Dziękuję za uwagę





kartkówka

1. Wymień pożądane cechy idealnych negocjacji
2. Zilustruj znane architektury wewnętrzne agentów, krótko opisz proces decyzyjny każdej z architektur
3. Opisz zasadę działania „filtrów cząsteczkowych”

