

Projektowanie interfejsu człowiek-komputer

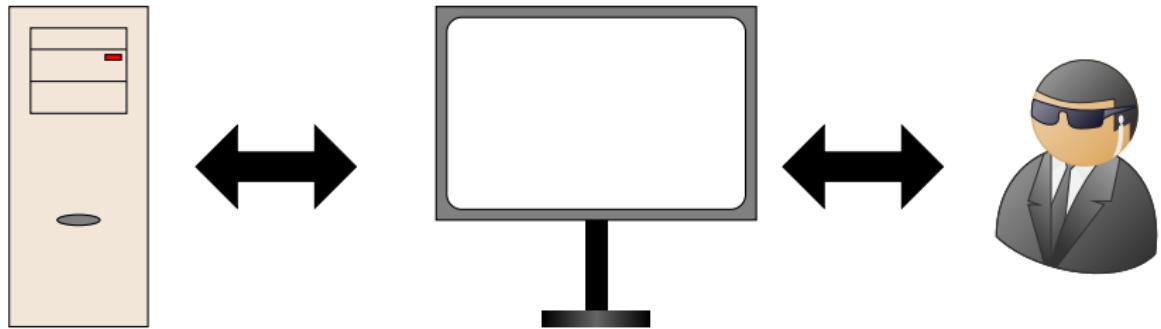
Jan Daciuk

¹Katedra Inteligentnych Systemów Interaktywnych, Wydział ETI, Politechnika Gdańsk

Multimedia i interfejsy

- ① Philip Kortum, *Usability Assessment: How to Measure the Usability of Products, Services, and Systems*, Users' Guides to Human Factors and Ergonomics Methods, Human Factors and Ergonomics Society, 2016.
- ② Jack Stuster, *Task Analysis: How to Develop an Understanding of Work*, Users' Guides to Human Factors and Ergonomics Methods, Human Factors and Ergonomics Society, 2019.
- ③ JoAnn T. Hackos, Janice C. Redish, *User and Task Analysis for Interface Design*, Wiley, 1998.
- ④ Mark Pearson, *Funkcjonalność stron internetowych*, Helion, 2002.
- ⑤ Jef Raskin, *The humane interface. New directions for designing interactive systems*, Addison-Wesley, 2000.
- ⑥ Serengul Smith-Atakan, *Human-Computer Interaction*, Thomson, 2006.

- ⑦ Jakob Nielsen, *Projektowanie funkcjonalnych serwisów internetowych*, Helion, 2003.
- ⑧ William M. Newman, Michael G. Lamming, *Interactive System Design*, Addison-Wesley, 1995.
- ⑨ Jenifer Tidwell, *Projektowanie interfejsów. Sprawdzone wzorce projektowe*, O'Reilly/Helion, 2012.



- Interfejs pośredniczy między człowiekiem a komputerem.
- Człowiek widzi wyłącznie interfejs
- Użyteczność interfejsu ma kluczowe znaczenie dla użytkownika

- Interfejs często decyduje o wyborze programu przez użytkownika.
- Interfejs decyduje o wygodzie, łatwości użytkowania, szybkości obsługi programu.
- Dobrze zaprojektowany interfejs systemu prowadzi do oszczędności przy jego użytkowaniu.
- Źle zaprojektowany interfejs utrudnia lub wręcz uniemożliwia skorzystanie z programu/systemu/urządzenia/strony internetowej.
- Świat jest pełen źle zaprojektowanych interfejsów.
- Umiejętność obsługi narzędzia do budowy interfejsu nie oznacza umiejętności projektowania interfejsu.
- Nowe rodzaje interfejsu wymagają nowych umiejętności, ale też znajomości ogólnych zasad.

Interfejs człowiek-komputer to nie tylko interfejs programu wykonywanego na komputerze biurowym. To także interfejs:

- stron WWW
- telefonów komórkowych
- tabletów
- automatów biletowych
- bankomatów
- aparatów fotograficznych
- pralek
- ...

Użytkowników możemy dzielić ze względu na:

- znajomość komputerów, znajomość dziedziny,częstość użycia
- wiek (interfejs dla dzieci, interfejs dla starszych użytkowników)
- sprawność fizyczną i umysłową
- język i kulturę

Różni użytkownicy mają różne wymagania dotyczące interfejsu.

Modele interfejsu

model **użytkownika** jak użytkownik widzi program/system; opis powinien być krótki, np. jednozdaniowy

model **programisty** jak interfejs ma działać

model **projektanta** model sugerowany użytkownikowi i zarys jego realizacji;
łączy model użytkownika z modelem programisty

Przyczyną wielu nieudanych projektów jest brak zgodności tych modeli, w szczególności modelu projektanta i modelu użytkownika.

Projekt programowego odtwarzacza muzycznego.

model użytkownika to się obsługuje jak odtwarzacz CD/DVD

model programisty program w C++, interfejs zrobiony za pomocą narzędzi dostępnych z Visual C++, funkcje podczepione do przycisków z obrazkami przycisków odtwarzacza CD/DVD, standardowe okno wyboru pliku

model projektanta okno z przyciskami przypominającymi przyciski odtwarzacza CD/DVD, obsługa za pomocą myszki, skróty, lista utworów dostępna pod klawiszem „wysuń płytke”

- Ocena pozwala stwierdzić, czy projekt idzie w dobrym kierunku i czy wykonaliśmy dobry interfejs.
- Oceny powinno się dokonywać często i jak najwcześniej. Wczesne wykrycie błędu zapobiega wielkim stratom.
- Ocena istniejącego już interfejsu może prowadzić do jego ulepszenia.
- Istnieją różne metody oceny, stosowane na różnych etapach projektu i badające różne aspekty interfejsu.
- Ostatecznym kryterium jest zadowolenie użytkownika.

Miary jakości:

- szybkość/wydajność przy wykonywania zadania (ilu ludzi do realizacji zadania);
- procent popełnianych błędów w trakcie wykonywania zadania;
- możliwość usunięcia skutków błędnych działań – wycofania się z błędnych decyzji;
- łatwość nauki (głównie czas);
- trwałość nauki;
- możliwość dostosowania interfejsu do własnych potrzeb;
- zadowolenie użytkownika.

Istnieją różne kryteria podziału metod oceny interfejsu:

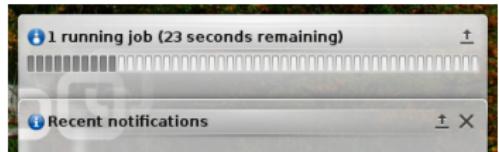
- ze względu na rolę osoby oceniającej:
 - ▶ obiektywne (np. napięcie psychiczne przez pomiar przewodności skóry, pomiar lub szacowanie czasu wykonania zadania na podstawie formalnego opisu),
 - ▶ subiektywne (np. poznanie, motywacja przez ankiety, wywiady);
- Ze względu na zakres:
 - ▶ ogólne, całościowe (np. ocena heurystyczna),
 - ▶ szczegółowe (np. ocena funkcjonalności/lista do sprawdzenia, GOMS);
- Ze względu na źródło wiedzy:
 - ▶ oparte na użytkowniku (np. obserwacja, wywiad),
 - ▶ oparte na specjalistie (np. ocena heurystyczna),
 - ▶ oparte na teorii (np. GOMS).

Metoda całościowa, wykonywana przez specjalistę lub zespół (najlepiej 3 do 5) specjalistów, wykorzystująca ich doświadczenie. Zasady:

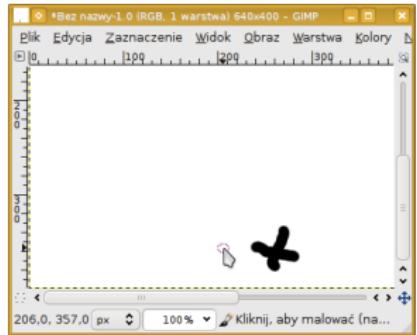
- ① Widoczny stan systemu.
- ② Używanie języka użytkownika.
- ③ Swoboda działań użytkownika.
- ④ Spójność i używanie standardów.
- ⑤ Zapobieganie błędom.
- ⑥ Minimalizowanie obciążenia pamięci.
- ⑦ Elastyczność i wydajność.
- ⑧ Prostota dialogu.
- ⑨ Dobre diagnozowanie błędów.
- ⑩ Pomoc i dokumentacja.

Użytkownik musi wiedzieć, co się dzieje w systemie, w jakim trybie jest interfejs.

- Stan systemu:



- Tryb systemu:



Należy używać takich pojęć, które pochodzą ze świata użytkownika, a nie ze świata programisty. Np. w pewnym znanym powszechnie systemie istnieje polecenie:

Polecenie

Mapuj dysk sieciowy

Wyraz „*mapować*” w języku polskim nie istnieje. Angielski wyraz „*to map*” należy przetłumaczyć na polski jako „*odwzorować*”. Jednak przetłumaczenie tekstu polecenia z polskiego na polski niewiele da:

Polecenie

Odwzoruj dysk sieciowy

bo nie wiadomo, o co chodzi. Lepiej by było napisać:

Polecenie

Podłącz dysk sieciowy

Użytkownik powinien mieć możliwość swobodnego **wyboru kolejności działań i sposobu wykonania** zadania. Powinien móc się **wycofać** z każdej czynności. Np. zamiast ostrzeżenia:

Ostrzeżenie

Czy na pewno chcesz usunąć plik poprawiny.jpg? (T/N)

lepiej jest dostarczyć funkcji anulowania usunięcia (odzyskania usuniętych danych).

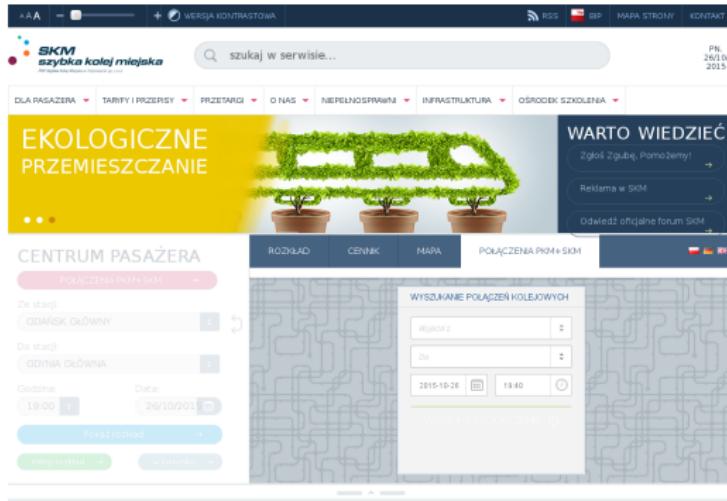
Zasada swobodnego wyboru jest nagminnie łamana przez administratorów/programistów stron internetowych, uniemożliwiających otwarcie strony, na którą wskazuje odnośnik w nowej karcie (zwykle środkowy przycisk/kółko myszy). Często wymusza się otwarcie strony albo zamiast bieżącej albo w nowym oknie.

- Spójność oznacza, że takie same rzeczy są zrealizowane w ten sam sposób w różnych miejscach interfejsu.
- Używanie standardów oznacza, że interfejs jest obsługiwany w sposób znany użytkownikowi z innych programów/systemów.

Spójność i używanie standardów dotyczą takich rzeczy jak:

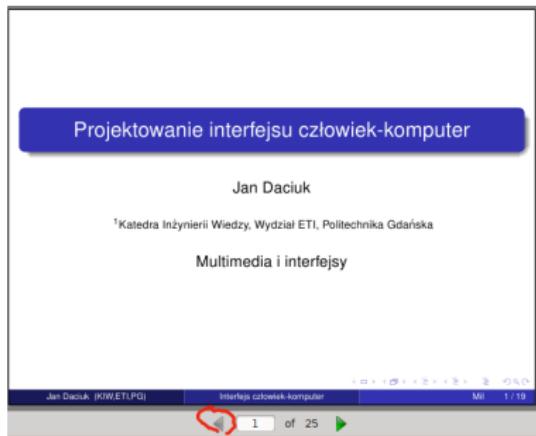
- sposób obsługi interfejsu
- barwy, czcionki
- menu, przyciski, skróty
- rozmieszczenie elementów na ekranie, wyrównywanie
- informacja zwrotna dla użytkownika

Spójność i używanie standardów



Dlaczego dwa rozkłady jazdy były obsługiwane w różny sposób? Z rozkładu „CENTRUM PASAŻERA” można było przejść do rozkładu „POŁĄCZENIA PKM+SKM” przyciskiem, w drugą stronę nie. W jednym rozkładzie była najpierw godzina, potem data; w drugim odwrotnie.

Zamiast wykrywać błędy po ich pojawienniu się i informować o nich użytkownika, lepiej jest im zapobiegać uniemożliwiając wykonanie działania prowadzącego do błędu, np. przez wyłączenie przycisku lub elementu menu, uniemożliwienie wprowadzenia znaku innego niż cyfra do pola liczby, wybór elementów z ograniczonego zbioru np. z menu itp.



Chodzi oczywiście o pamięć użytkownika, nie o pamięć komputera. Nie należy zmuszać użytkownika do pamiętania informacji np. o składni poleceń, funkcji przycisków, wprowadzonych przez użytkownika wcześniej danych, danych podanych wcześniej użytkownikowi itp. Można osiągnąć to np. przez:

- stosowanie menu,
- stosowanie list rozwijalnych,
- stosowanie przycisków radiowych itp.,
- zostawianie/wyświetlanie na ekranie wcześniej wprowadzonych danych,
- wyświetlanie skrótów obok pozycji menu,
- itp.

Doświadczony użytkownik powinien mieć możliwość jak najszybszego wykonania zadania. Osiąga się to przez stosowanie skrótów, np. w edytorze emacs kombinacja klawiszy:

`Ctrl-x r Ctrl-space x`

zapisuje bieżącą pozycję kurSORA w rejestrze x, zaś:

`Ctrl-x r j x`

ustawia kurSOR na pozycji zapisanej w rejestrze x.

Od czasów pierwszej przeglądarki internetowej NCSA Mosaic kombinacja Shift i kliknięcia lewym przyciskiem myszy w odnośnik oznacza zapisanie na dysku pliku, na który wskazuje odnośnik. Niestety M\$ IE wprowadził nowy „standard” i z nowoczesnych przeglądarek tylko kilka rzadko używanych stosuje standardowy skrót.

Dialog nie powinien zawierać informacji zbędnej lub rzadko potrzebnej. Dodatkowe informacje walczą o uwagę użytkownika i mogą przesłonić informacje bardziej istotne.

The screenshot shows the official website of the Szybka Kolej Miejska (SKM) in Gdańsk. At the top, there's a banner with a yellow and blue train and the text "Dajemy Ludziom swobodę przemieszczania się!". Below the banner, there's a "Centrum Pasażera" (Passenger Center) form where users can enter departure and arrival stations and travel times. To the right of the form is a map showing a route from Gdańsk to Gdynia. A sidebar on the right contains a "Warto wiedzieć" (It's worth knowing) section with links to forum discussions and a feedback form.

Ilu użytkowników – pasażerów SKM – potrzebuje na codzień mapy przystanków (na szczęście ta strona też została poprawiona)?

Dobre komunikaty o błędach powinny nie tylko w sposób dokładny i zrozumiały dla użytkownika informować o zaistniałym problemie, ale także podpowiadać możliwe rozwiązania. Zamiast:

Ostrzeżenie

Wystąpił nadmiar w funkcji ComputeAngle w wierszu 1378 pliku graphnew.cc. Kąt został wyzerowany.

lepiej użyć np.:

Ostrzeżenie

Podano niewłaściwą kombinację danych odległość i kierunek. Popraw dane i spróbuj ponownie.

Pomoc i dokumentacja

Nie dla wszystkich użytkowników program będzie intuicyjny w obsłudze.

Należy zapewnić:

- podpowiedzi („dymki”) do przycisków, elementów menu, pól
- podręcznik użytkownika i encyklopedię użytkownika z rozbudowanymi odnośnikami i skorowidzem.

Istnieją też inne formy pomocy:

- samouczek
- pomoc na telefon
- szkolenia
- fora internetowe

Przegląd poznawczy

Przegląd poznawczy (ang. *cognitive walkthrough*) bada wyznaczony fragment interfejsu. Najlepiej zacząć od najczęściej wykonywanych lub krytycznych dla systemu zadań.

- Metoda może być wykorzystana w dowolnej fazie projektowania systemu, a także na gotowym systemie.
- Wykonywana jest przez różne osoby, ale najlepiej jest, gdy wykonuje ją kilku specjalistów znających się na ocenie interfejsu.
- W jednej sesji wybierane jest pojedyncze zadanie do wykonania. Specjalista zna optymalny ciąg kroków.
- Każdy krok jest analizowany, specjalista odpowiada na cztery pytania.
- Po zakończeniu zadania, odpowiedzi są analizowane i sporządzane jest podsumowanie wskazujące możliwy sposób poprawy zauważonych niedociągnięć interfejsu.

W metodzie tej po każdym kroku odpowiada się na następujące pytania:

- ① Czy użytkownik pracuje nad właściwą częścią zadania?
- ② Czy użytkownik wie, że właściwe działanie jest dostępne?
- ③ Czy użytkownik skojarzy cel z właściwym działaniem?
- ④ Czy użytkownik będzie wiedział, że wykonane działanie przybliżyło go do celu?

Czy użytkownik pracuje nad właściwą częścią zadania?

- Użytkownik musi wiedzieć, jakie czynności musi wykonać do wykonania zadania.
 - ▶ Np. użytkownik nie wie, że zmiana czcionki tekstu wymaga zaznaczenia tekstu.
- Użytkownik musi wykonywać kroki zadania w określonej kolejności.
 - ▶ Np. użytkownik nie wie, że najpierw powinien zaznaczyć tekst, a dopiero później wybrać pożądaną czcionkę.

Wiedza użytkownika może wynikać:

- z doświadczenia z tym systemem (robił to we wcześniejszych krokach)
- z doświadczenia z innymi systemami
- z podpowiedzi systemu
- z logicznego ciągu czynności

Czy użytkownik wie, że właściwe działanie jest dostępne?

- z doświadczenia z tym systemem (robił to we wcześniejszych krokach)
- z doświadczenia z innymi systemami
- z powodu widoczności odpowiedniego elementu menu, przycisku itp. wskazującej jego aktywność
- dzięki podpowiedziom systemu

Czy użytkownik skojarzy cel z właściwym działaniem?

- z doświadczenia z tym systemem (robił to we wcześniejszych krokach)
- z doświadczenia z innymi systemami
- dzięki nazwie czynności będącej elementem menu, napisem na przycisku itp.
- dzięki podpowiedziom systemu
- drogą eliminacji — inne działania już próbował

Czy użytkownik będzie wiedział, że wykonane działanie przybliżyło go do celu?

Każde działanie użytkownika powinno być w jakiś sposób potwierdzone przez system. Może się to odbywać na różne sposoby:

- bezpośrednie pokazanie wyniku działania
- pokazanie paska postępu w przypadku działań wymagających dłuższego czasu
- wciśnięcie przycisku
- zmiana barwy elementu interfejsu
- sygnał dźwiękowy
- itp.

Istnieje uproszczona wersja przeglądu poznawczego, zwana czasem po angielsku *cognitive jogthrough*. W tej wersji cztery pytania zastąpione są tylko dwoma:

- ① Czy użytkownik będzie wiedział, co zrobić w tym kroku?
- ② Jeśli użytkownik podejmie właściwe działanie, czy będzie wiedział, że było to właściwe działanie?

Metoda wykorzystująca specjalistę, zorientowaną na szczegóły, która nie pozwala na ocenę całościową. Używana w dużych projektach w późnej fazie rozwoju systemu. Gwarantuje dużą spójność interfejsu.

- ① Interfejs jest dzielony na części. Każda jest oceniana oddzielnie.
- ② Na podstawie wymagań, standardów, podręczników stylu tworzona jest szczegółowa lista cech do sprawdzenia.
- ③ Osoba dokonująca oceny sprawdza przypisaną jej część interfejsu na zgodność z zaleceniami zapisanymi na liście. Osobą nie musi być specjalistą, ale specjalista lepiej wykona zadanie.
- ④ Wskazuje się zalecane zmiany na podstawie szczegółowych sprawozdań.

Lista cech do sprawdzenia może zawierać np.:

- co najwyżej dwa poziomy intensywności (np. wytłuszczenie)
- nie więcej niż 4 wielkości czcionek (większa=ważniejsza)
- nie więcej niż 3 kroje czcionek
- nie więcej niż 4 barwy; skojarzenia barw
- migotanie tylko w wyjątkowych sytuacjach
- łagodne dźwięki do potwierdzania, ostre – ostrzegania
- szersze zamiast głębsze menu; grupowanie w menu
- grupowanie i wyrównywanie w formularzach, pokazywanie struktury pól
- w pasku głównego menu, pierwszy element to File/Plik, ostatni to Help/Pomoc
- ostatni element menu File/Plik to Quit/Exit/Wyjście
- skrót Ctrl-Q powoduje wyjście z programu

Metody opisu interfejsu

- opisowe

- ▶ gramatyka wielostronna (multiparty grammar)
- ▶ drzewo menu
- ▶ UAN (**U**ser **A**ction **N**otation)
- ▶ specyfikacja obiektu
- ▶ direct manipulation sketch

- szacunkowe

- ▶ GOMS (**G**oals, **O**perators, **M**ethods, **S**election rules)
- ▶ TAG (**T**ask **A**ction **G**rammar)
- ▶ diagram stanów: STD (**S**tate **T**ransition **D**iagram), RTD (**R**ecursive **T**ransition **D**iagram), ATD (**A**ugmented **T**ransition **D**iagram), IOG (**I**nteraction **O**bject **G**raph **S**tate **D**iagrams)

- **Goals** (cele) – to, co użytkownik zamierza osiągnąć; rozbijane na podcele.
- **Operators** (operatorzy) – (niepodzielne) działania do osiągnięcia celu niezależnie od kontekstu; czas przybliżany stałą, rozkładem prawdopodobieństwa lub funkcją parametrów.
- **Methods** (metody) – ciągi operatorów i/lub wywołań podcelów do osiągnięcia celu.
- **Selection rules** (reguły wyboru) – w przypadku istnienia więcej niż jednej metody do osiągnięcia tego samego celu rozstrzygają wybór w zależności od warunków zadania.

Metoda GOMS pozwala na szacowanie czasu wykonania (wszystkie wersje) oraz czasu uczenia się i spójności (z wyjątkiem KLM).

Czas wykonania uzyskuje się mnożąc czas wykonania operatorów w poszczególnych metodach przez liczbę ich wywołań. Czas ten odzwierciedla sytuację, w której zadanie jest wykonywane przez doświadczonego użytkownika.

Czas uczenia się zawiera czas wykonania oraz czysty czas uczenia się. Z kolei czysty czas uczenia się składa się z czasu uczenia metod (ok. 17 s na każde zdanie języka GOMS) i czas uczenia się faktów do przechowania w pamięci długoterminowej (ok. 7 s na pojedynczy fakt).

Spójność można mierzyć licząc długość (liczbę zdań) opisu w języku GOMS. Jest to miara względna – może służyć do porównywania różnych wersji interfejsu.

- **CMN-GOMS** – oryginalny GOMS (Card, Moran i Newell, 1980) – pełny GOMS, operatory verify-... po niektórych operacjach
- **KLM** – Keystroke-Level Model (Card, Moran i Newell, 1980) – bardzo uproszczony CMN-GOMS z 6 operatorami (w tym **M** – „przygotuj się do...” zamiast verify-...), brakiem metod i reguł wyboru: po prostu lista operatorów do wykonania danego zadania
- **NGOMSL** – Natural GOMS Language (Kieras 1988) – sformalizowana analiza GOMS
- **CPM-GOMS** – Cognitive-Perceptual-Motor analysis of activity (Gray, John i Atwood, 1993) – do analizy zadań z dużą liczbą równoległości, możliwość szacowania czasu podobna do PERT
- **GOMSL** – GOMS Language (Kieras, 1999) – następca NGOMSL, model wykonywalny za pomocą narzędzia GLEAN3

Przykład **metody** w NGOMSL. Kolumna *czas* określa czas wykonania operatora. Kolumna *wykonań* wypełniana jest dla określonego zadania.

	Wykonań	Czas
Method for goal: Highlight arbitrary text	1	
Step 1. Determine position of beginning of text.	1	1.20
Step 2. Move cursor to beginning of text.	1	1.10
Step 3. Click mouse button.	1	0.20
Step 4. Determine position of end of text. (already known)	1	0.00
Step 5. Move cursor to end of text.	1	1.10
Step 6. Shift-click mouse button.	1	0.48
Step 7. Verify that correct text is highlighted.	1	1.20
Step 8. Return with goal accomplished.	1	

Przykład **reguły wyboru** w NGOMSL. W tym konkretnym zadaniu tekst jest dowolny, wybrany jest więc podcel: *podświetl (wyróżnij) dowolny tekst.*

	Wykonań	Czas
Selection rule set for goal: Highlight text		1
If text-is word, then accomplish goal: Highlight word.		
If text-is arbitrary, then accomplish goal: Highlight arbitrary text.	1	
Return with goal accomplished.	1	

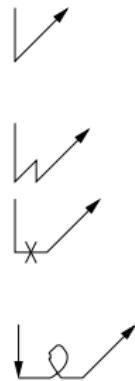
User Action Notation (UAN)

Elementy opisu:

M	– przycisk myszy	K	– wejście znakowe (klawiatura)
v	– w dół (wciśnięcie)	K"znaki"	– podanie „znaków”
^	– w górę (zwolnienie)	K(zmienna)	– podanie wartości zmiennej
[]	– kontekst obiektu	K(regexp)	– określenie za pomocą wyr. reg.
~	– przesuń kursor	piktogram'	– wybrany piktogram
¬	– negacja	ramka >~	– ramka podąża za kursorem
=	– równe	~ [x, y]*	– dowolna liczba powt.
*	– różne	forall()	– dla każdego

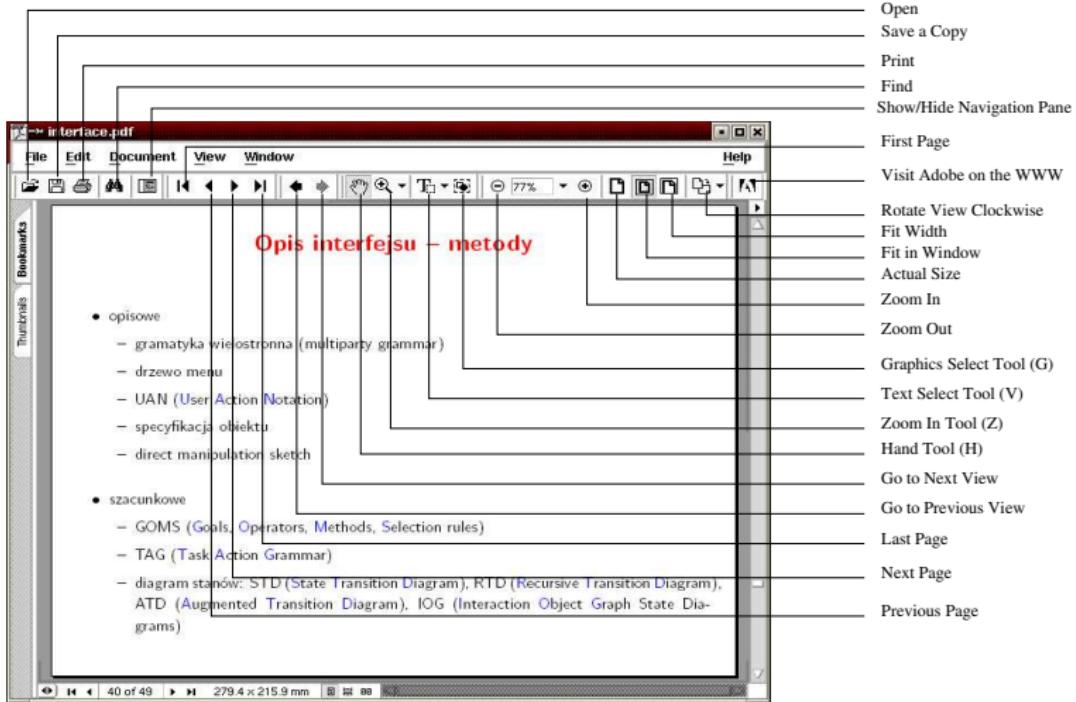
Przykład: usunięcie pliku.

działania użytkownika	odpowiedź interfejsu	stan interfejsu
~[piktogram pliku']Mv	piktogram pliku'! forall(piktogram pliku * piktogram pliku'): piktogram pliku-!	wybrany = plik
~[x,y]*	obrys(piktogram pliku') >~	
~[piktogram kosza]	obrys(piktogram pliku') >~ piktogram kosza!	
M^	usuń(piktogram pliku') piktogram kosza!!	wybrany = null



- kliknięcie/wybranie
lub dotknięcie
- podwójne kliknięcie
- wybranie z
samopowtarzaniem
- przeciągnięcie





Drzewo menu

Alt-F File

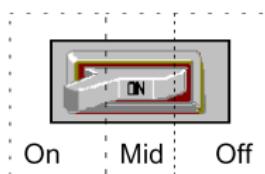
- N** New... Ctrl+N
- O** Open... Ctrl+O
- C** Close Ctrl+w
- Close all
- S** Save Ctrl+S
- A** Save As...
- L** Save All
- P** Print
- X** Exit

Alt-E Edit

- N** Undo Ctrl+Z
- E** Redo Ctrl+4
- U** Cut Ctrl+X
- C** Copy Ctrl+C
- P** Paste Ctrl+V
- ...

Łatwe w użyciu, szybko można je przekształcić w prototyp i zrealizować.
Nadają się tylko do opisu menu...

Gramatyka bezkontekstowa, w której symbole niekońcowe należą do trzech kategorii, w zależności od tego, kto inicjuje działanie: człowiek/ użytkownik (etykietowane **H:** lub **U:**), komputer (etykietowane **C:**) lub mieszane (brak oznaczenia). Nie ogranicza się wyłącznie do opisu poleceń wierszowych – symbole końcowe mogą być np. zapisane w UAN:



```
<TurnOn> ::= <C:(switch = OFF)> <H:~[Off]> <H:Mv> <C:display(OffActive)>
             <H:~[Mid]> <C:display(MidActive)>
             <H:~[On]> <C:display(OnActive)>
             <H:M^> <C: display(OnStatic)> <C: switch := ON>
```

Analiza zadania jest metodą doświadczalną, która może doprowadzić do uzyskania modelu zadania i sposobów jego wykonywania przez ludzi. Analiza zadania jest metodą badania, jak ludzie wykonują zadania, wykorzystując przy tym:

- teorię zadań,
- techniki zbierania danych,
- metody analizy zadania,
- reprezentację pozwalającą tworzyć modele zadania.

Możliwe podejścia:

- ① analiza i modelowanie ludzkiej wiedzy,
- ② badanie zdolności,
- ③ hierarchiczna analiza zadania.

Analiza zadania stara się rozpoznać wiedzę na temat:

- celów, podcelów i podzadań
- planów
- strategii
- procedur
- obiektów
- działań

a także ustalić, które elementy są typowe, a które najważniejsze.

Są stosowane różne techniki zbierania wiedzy:

- ankiety i wywiady - dostarczają wiedzy ogólnej, reguł, zasad
- techniki obserwacyjne - dostarczają wiedzy szczegółowej, kontekstowej
- protokoły - dostarczają wiedzy zarówno ogólnej, jak i szczegółowej, bardzo subiektywne i czasochłonne
- techniki doświadczalne: siatka zespołu, porządkowanie kart, ocena liczbową, liczenie częstości: dostarczają wiedzy na temat obiektów i działań
- inne techniki: konkurs wiedzy, dyskusja w grupie, pytania wielokrotnego wyboru itp.

Modele pojęciowe

Model pojęciowy to sposób w jaki użytkownik postrzega i rozumie interfejs.

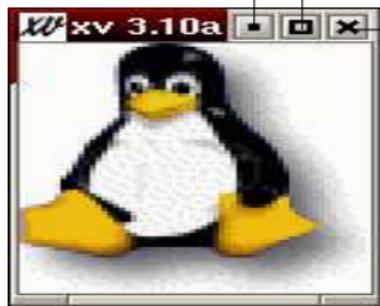
- Rodzaje modeli:

- ▶ obiekt-działanie
- ▶ odwzorowanie
- ▶ diagram przejść stanów
- ▶ przenośnia
- ▶ analogia

- Rodzaje modeli:

- ▶ wykonywalne
- ▶ opisowe

Obiekt – działanie

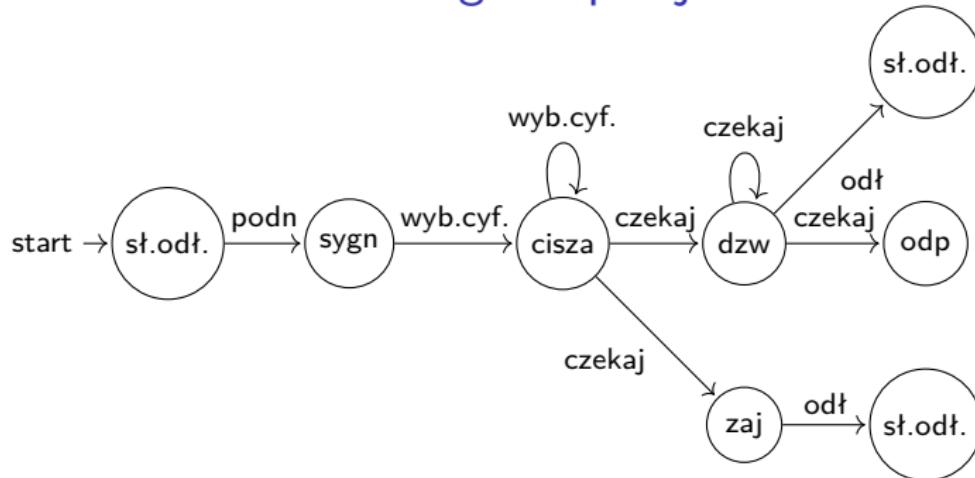


- M1v[^] – minimize
- M1v[^] – maximize
- M2v[^] – maximize vertically
- M3v[^] – maximize horizontally
- M1v[^] – close

Najprostszy model pojęciowy. Może być traktowany jako ścieżka w diagramie przejść stanów.

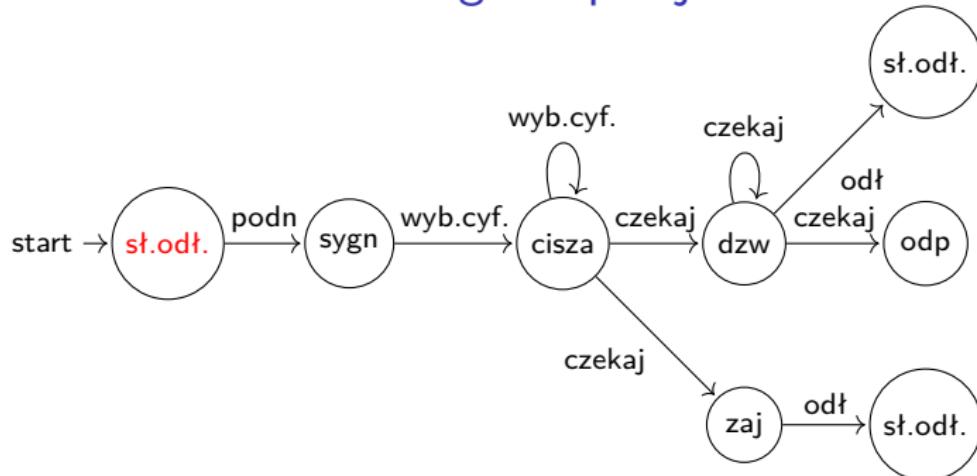
- Zamiar: usuń plik.
- Wykonanie:
 - ① napisz `rm`;
 - ② napisz nazwę pliku;
 - ③ wcisnij ENTER.

Diagram przejść stanów



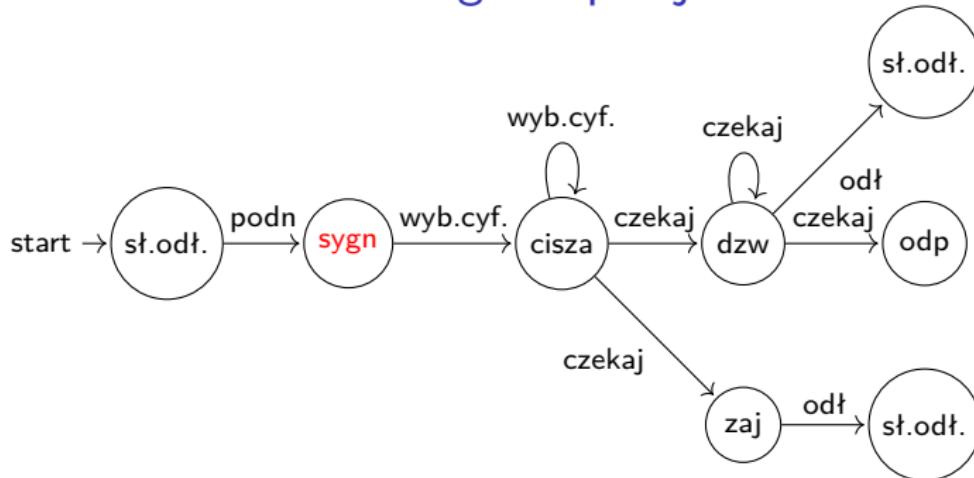
sł.odł słuchawka odłożona;
podn podnieś słuchawkę;
sygn ciągły sygnał;
wyb.cyf wybierz cyfrę;
dzw dzwonek;
odł odłóż słuchawkę;
odp telefon odpowiada.

Diagram przejść stanów



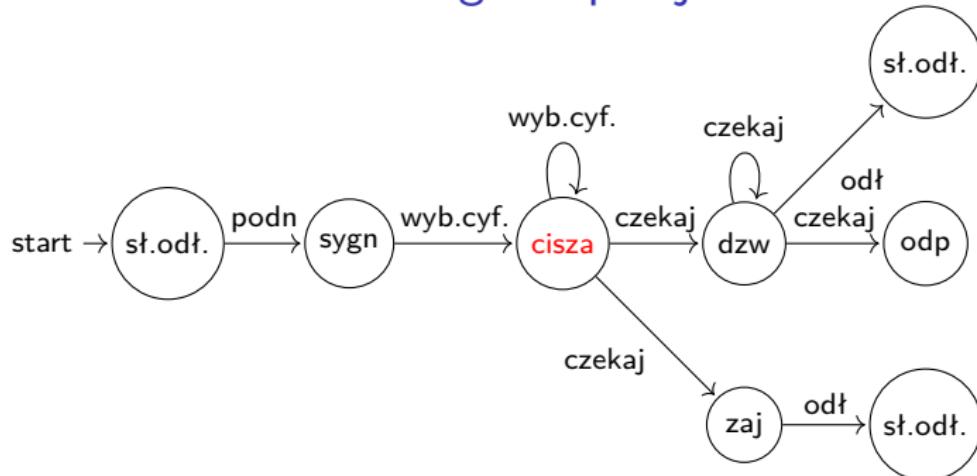
sł.odł słuchawka odłożona;
podn podnieś słuchawkę;
sygn ciągły sygnał;
wyb.cyf wybierz cyfrę;
dzw dzwonek;
odł odłóż słuchawkę;
odp telefon odpowiada.

Diagram przejść stanów



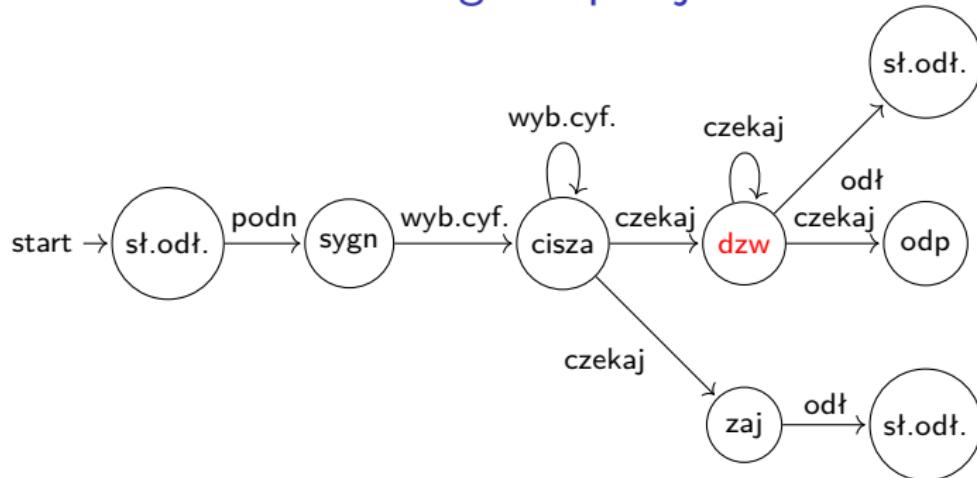
sł.odł słuchawka odłożona;
podn podnieś słuchawkę;
sygn ciągły sygnał;
wyb.cyf wybierz cyfrę;
dzw dzwonek;
odł odłóż słuchawkę;
odp telefon odpowiada.

Diagram przejść stanów



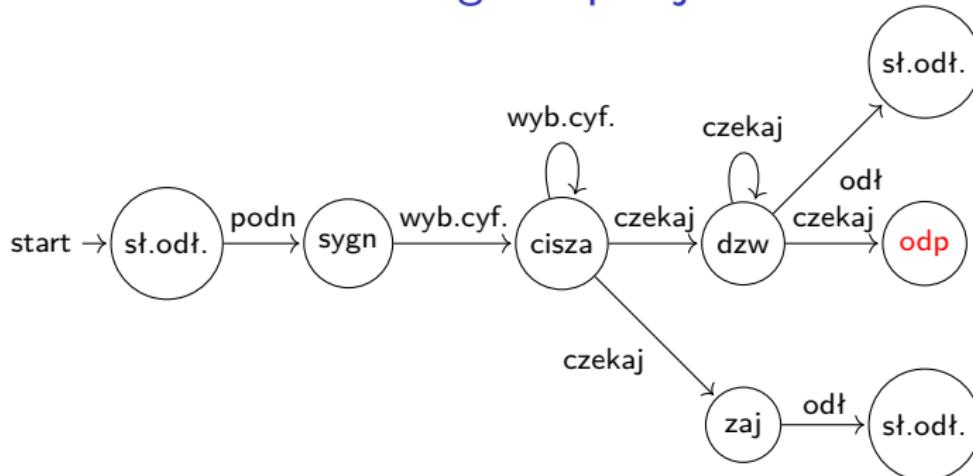
sł.odł słuchawka odłożona;
podn podnieś słuchawkę;
sygn ciągły sygnał;
wyb.cyf wybierz cyfrę;
dzw dzwonek;
odł odłóż słuchawkę;
odp telefon odpowiada.

Diagram przejść stanów



sł.odł słuchawka odłożona;
podn podnieś słuchawkę;
sygn ciągły sygnał;
wyb.cyf wybierz cyfrę;
dzw dzwonek;
odł odłóż słuchawkę;
odp telefon odpowiada.

Diagram przejść stanów



sł.odł słuchawka odłożona;
podn podnieś słuchawkę;
sygn ciągły sygnał;
wyb.cyf wybierz cyfrę;
dzw dzwonek;
odł odłóż słuchawkę;
odp telefon odpowiada.

Analogia

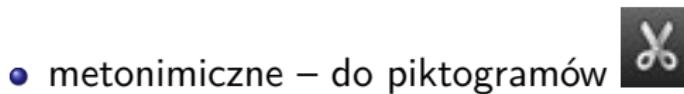


Rodzaje przenośni:

- strukturalne – **model pojęciowy**
- orientacyjne – do formantów (widgetów)



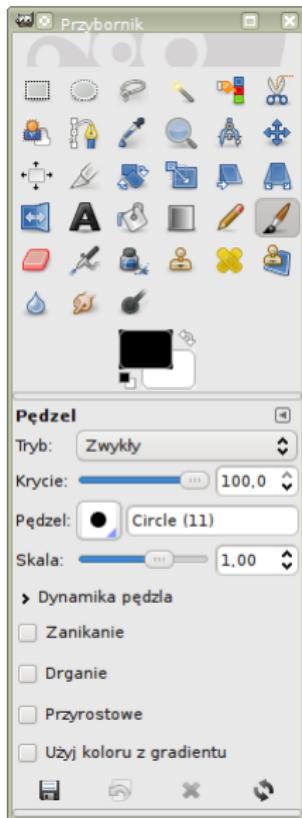
- ontologiczne – do sterowania



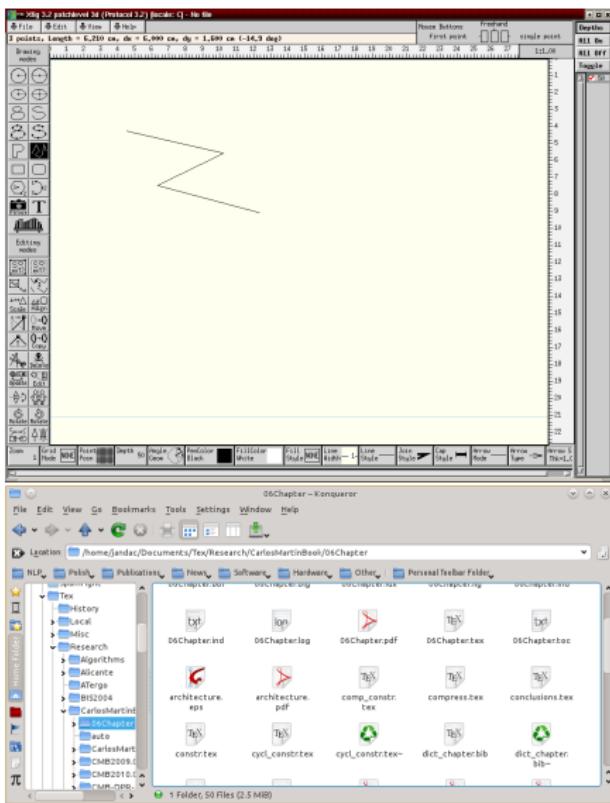
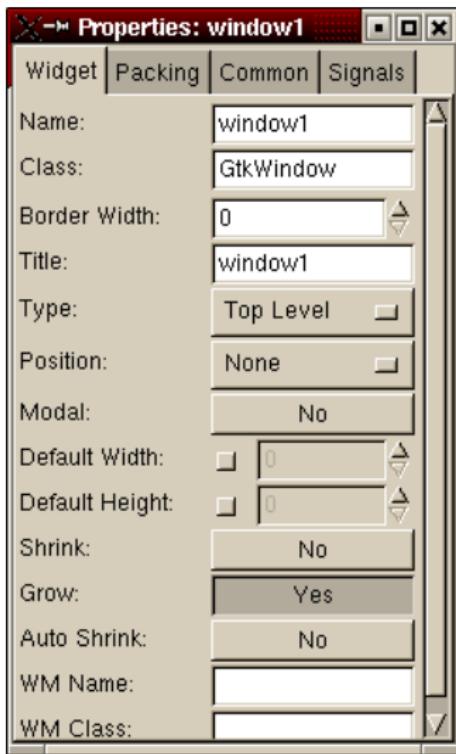
- metonimiczne – do pictograms

Pierwsze trzy rodzaje przenośni tłumaczone są na angielski jako *metaphor*, ostatnia – jako *metonymy*.

Przenośnie strukturalne

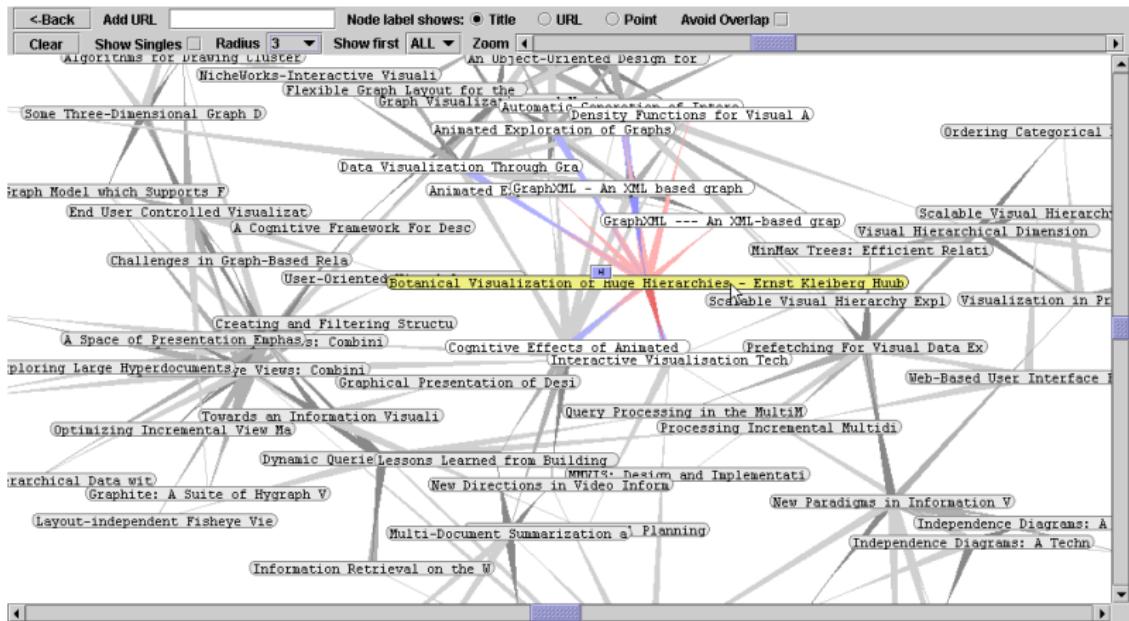


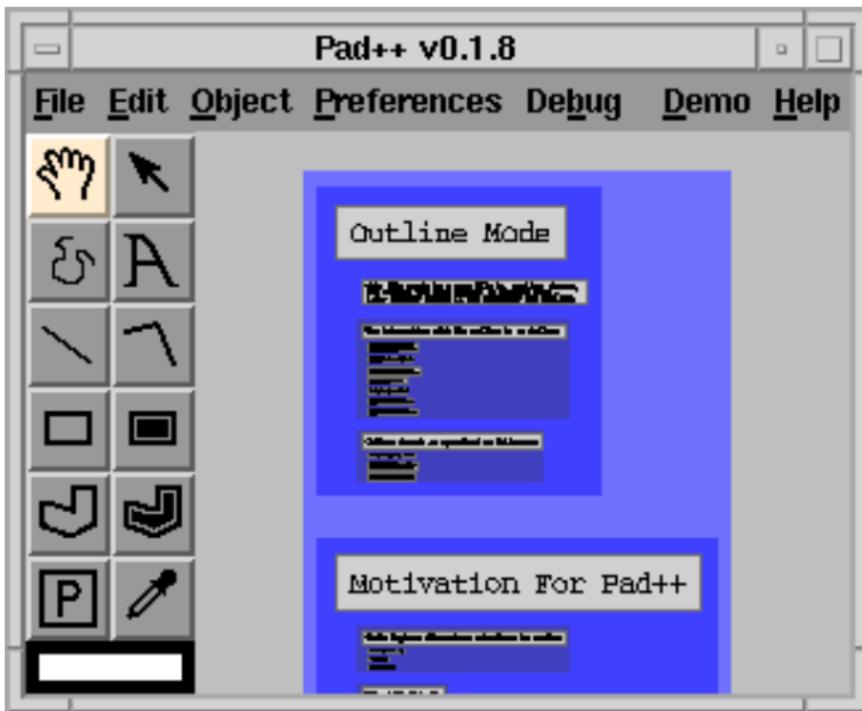
The screenshot displays three windows side-by-side. The top window, 'Dokument_1', is a word processor with a toolbar, font and size dropdowns, and a rich text editor. It shows a single paragraph of text. The bottom-left window, 'Zeszyt1 gnumeric - Gnumeric', is a spreadsheet with a toolbar, font dropdown, and a grid of cells labeled A1 through L9. The bottom-right window, 'Zeszyt2 gnumeric', is another spreadsheet with a similar interface and grid.

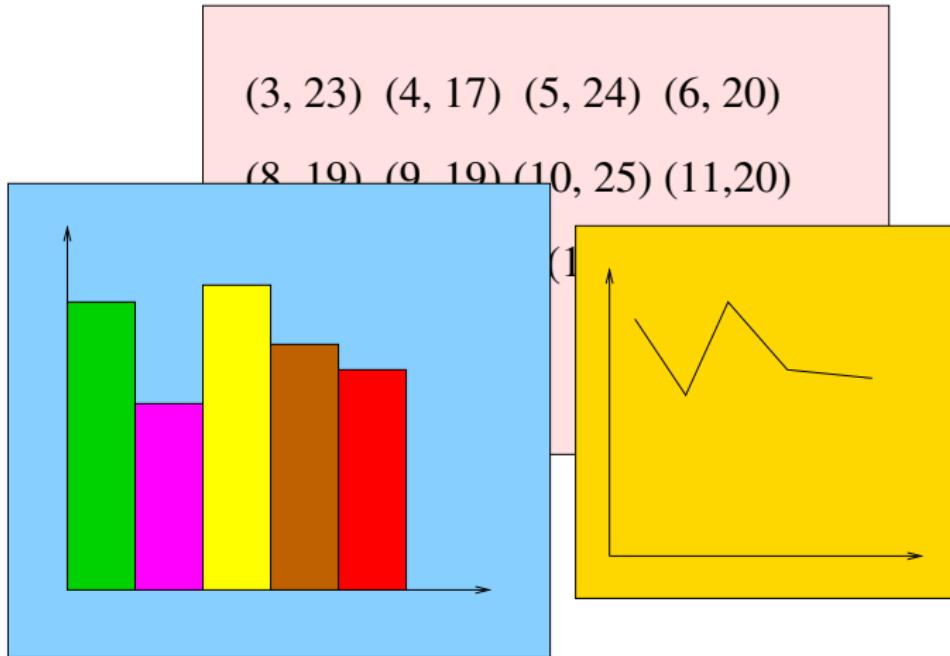




Co się dzieje, kiedy nie wszystko w interfejsie działa w sposób analogiczny do rzeczywistości naśladowanej przez przenośnięę?







Loupe demo

Where are the Pirates of the Caribbean?



Loupe demo

Where are the Pirates of the Caribbean?





Display



ZUI – magiczna lupa

- A**
 Akcent Portal
 Akufen Search
 Amazon Shopping
 AOL Classifieds
 AOL Classifieds
B
 Akzent Portal
 Akufen Search
 Amazon Shopping
 AOL Classifieds
 AOL Classifieds
C
 Akzent Portal
 Akufen Search
 AOL Classifieds
 AOL Classifieds
 AOL Classifieds
 AOL Classifieds
 AOL Classifieds
D
 Akzent Portal
 Akufen Search
 AOL Classifieds
 AOL Classifieds
 AOL Classifieds
 AOL Classifieds
 AOL Classifieds
 AOL Classifieds
 AOL Classifieds
E
 Akzent Portal
 Akufen Search
 AOL Classifieds
 AOL Classifieds
 AOL Classifieds
F
 Akzent Portal
 AOL Classifieds
 AOL Classifieds
G
 AOL Classifieds
H
 AOL Classifieds
I
 AOL Classifieds
J
 AOL Classifieds
K
 AOL Classifieds
L
 AOL Classifieds
M
 AOL Classifieds
N
 AOL Classifieds
O
 AOL Classifieds
P
 AOL Classifieds
R
 AOL Classifieds
S
 AOL Classifieds
T
 AOL Classifieds
U
 AOL Classifieds
W
 AOL Classifieds

- A**
 AOL Classifieds
 AOL Classifieds
B
 AOL Classifieds
 AOL Classifieds
C
 AOL Classifieds
D
 AOL Classifieds
E
 AOL Classifieds
 AOL Classifieds
 AOL Classifieds
 AOL Classifieds
 AOL Classifieds
 AOL Classifieds
F
 AOL Classifieds
 AOL Classifieds
 AOL Classifieds
 AOL Classifieds
 AOL Classifieds
 AOL Classifieds
G
 AOL Classifieds
H
 AOL Classifieds
 AOL Classifieds
 AOL Classifieds
 AOL Classifieds
 AOL Classifieds
I
 AOL Classifieds
J
 AOL Classifieds
K
 AOL Classifieds
L
 AOL Classifieds
M
 AOL Classifieds
N
 AOL Classifieds
O
 AOL Classifieds
P
 AOL Classifieds
R
 AOL Classifieds
S
 AOL Classifieds
T
 AOL Classifieds
U
 AOL Classifieds
W
 AOL Classifieds

- Metoda opisuje jak odbywa się dialog człowieka z komputerem; nie jest bezpośrednio związana z przenośnią ani z konkretnym urządzeniem.
- Większość metod metod może być użyta z dowolną przenośnią i z prawie dowolnym urządzeniem. Przykłady:
 - ▶ Unix – metoda wiersza poleceń z przenośnią „pliki i segregatory” i urządzeniem „klawiatura”;
 - ▶ Gry przygodowe – metoda wiersza poleceń z przenośnią „rzeczywistość” i urządzeniem „klawiatura”.
- Możliwe jest stosowanie kilku metod jednocześnie.

```
gs -sDEVICE=lpjet4 -sOutputFile=foo.lj4 -sPAPERSIZE=a4  
-dNOPAUSE
```

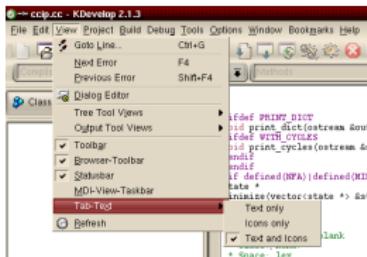
- Użytkownik pisze wiersz tekstu, naciska wybrany klawisz (np. Enter lub Return). Komputer interpretuje wiersz jako polecenia i wyświetla wyniki.
- Użytkownik musi dokładnie pamiętać składnię poleceń. Nazwy poleceń na ogół kojarzą się ze słowami z języka naturalnego.
- Zalety: bardzo szybka i wydajna, gdy używana przez fachowców, dokładna, elastyczna.
- Wady: olbrzymie wymagania stawiane ludzkiej pamięci, wymaga długiego szkolenia i częstego używania, nędzna obsługa błędów.

Metoda – bezpośredni wybór



- Każda funkcja ma przyporządkowany klawisz (lub kombinację klawiszy).
- Obecnie bardzo często używana jako metoda pomocnicza dla innych metod – skróty klawiszowe.
- Zalety: jeszcze szybsza niż polecenia wierszowe, użyteczna przy ograniczonej liczbie funkcji. Wady: obciąża pamięć jeszcze więcej niż metoda poleceń wierszowych.

- Rodzaje:



paskowe i rozwijalne



pasek narzędzi

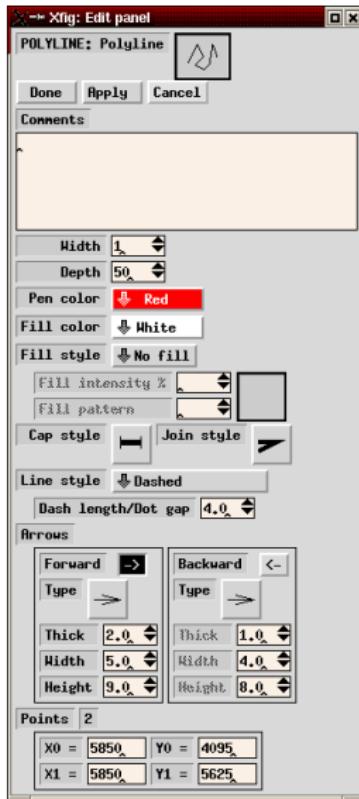


skrzynka narzędziowa

- Zalety: minimalne obciążenie pamięci, łatwość tworzenia (gotowe narzędzia), znane użytkownikom, niepotrzebna klawiatura.
- Wady: zajmuje miejsce, nie nadaje się do wprowadzania danych, nie nadaje się do komunikacji inicjowanej przez użytkownika, czasami wolne.



- Zarządza widokami: rozdziela, skaluje, nakłada, ogranicza.
- Rozdziela wątki.
- Dwa style zarządzania wieloma oknami: SDI (**S**ingle **D**ocument **I**nterface) i MDI (**M**ultiple **D**ocument **I**nterface) – obecnie najczęściej jako karty.



- Zalety: grupuje powiązane ze sobą dane, małe obciążenie pamięci, podobne do formularza – znajome pojęcie, istnieją gotowe narzędzia.
- Wady: zabiera miejsce na ekranie (przesłania inne obiekty), kolejność nie jest oczywista, wymaga sterowania kursorem, czasami wolne.

- Użytkownik w naturalny sposób posługuje się obiekttami pojawiającymi się na ekranie tak jakby były one obiekttami świata rzeczywistego. Mysz jest przedłużeniem ludzkiej ręki do świata komputerowego.
- Zalety: metoda lubiana przez większość użytkowników, łatwo rozszerza się na pozorną rzeczywistość przy użyciu specjalnych rękawic, hełmów, urządzeń śledzących ruch itp.
- Wady: realizacja może być wolna, trudniej jest dostosować metodę do potrzeb wymagających użytkowników, zawodzi, gdy zawodzi przenośnia, kiedy zbyt wiele dzieje się w sposób „magiczny” (np. wyrzucanie dyskietki napędu przez przeniesienie piktogramu dyskietki na piktogram kosza na śmieci).

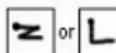
- **W**indows, **I**cons, **M**enu, **P**ointer.
- Najpopularniejsza metoda, łącząca w sobie kilka innych.
- Istnieje w wielu wersjach, także wyłącznie tekstowych.
- Wymaga szkolenia, zawodzi w przypadku użytkowników z poważnymi wadami wzroku. Zastosowana w czystej formie przeszkadza fachowcom w wykonywaniu prostych zadań.

Navigation

History Forward



Close



Double Size



Start over image

History Backward



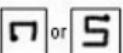
Up a Directory



Half Size



View Source



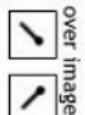
Reload



Scroll Up (200px)



Hide Image



View Cookies
View <META> Info



Forced Reload



Scroll Down (200px)



Links Functions

Homepage



Tabbed Browsing



Drag at/over link

New Document



Duplicate Tab



Link in new tab



Link in new window

Duplicate Window



Next Tab



Horizontal Stack



Finish with

Minimize Window



Previous Tab



Open every link
dragged over (window)



Open every link
dragged over (tab)

Maximize/Restore

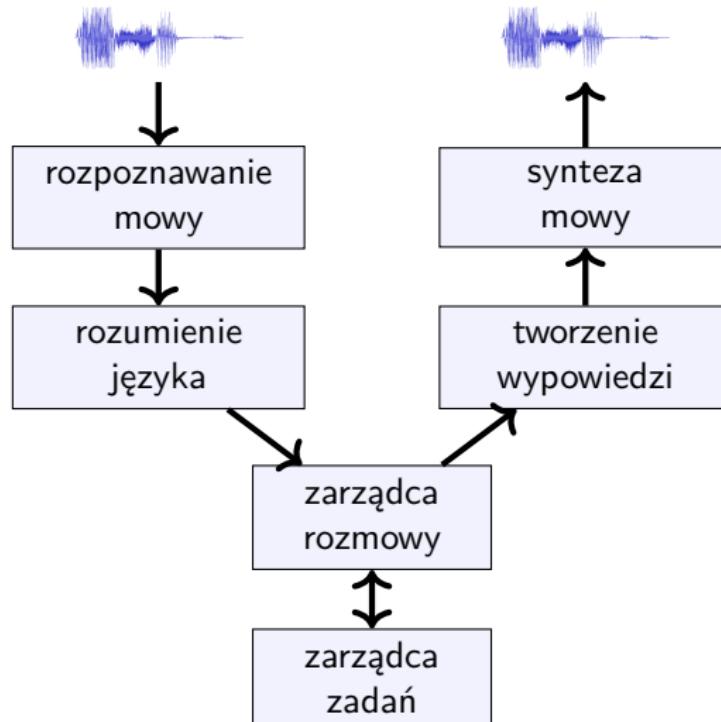


New Tab

- Użyteczne w sytuacjach takich jak np. robienie notatek na zebraniach, rozpoznawanie adresów na kopertach itp.
- Dwa sposoby rozpoznawania:
 - ▶ dynamiczny - bierze pod uwagę szybkość pisania, bada zależność położenia pisaka od czasu (dScript z NICI Nijmegen, Crosspad z IBM);
 - ▶ statyczny - danymi do rozpoznawania jest obraz (HWAI z SUNY Buffalo).
- Każdy ze sposobów ma fazy segmentacji (podział na część zawierającą pismo i nie, podział na wiersze, wyrazy i znaki), rozpoznawania znaków, filtrowania wyników (odrzucania mało prawdopodobnych).
- Skuteczność zależy od wielu czynników: stylu pisania, jakości uczenia (danych i algorytmu), wielkości słownika itp.;

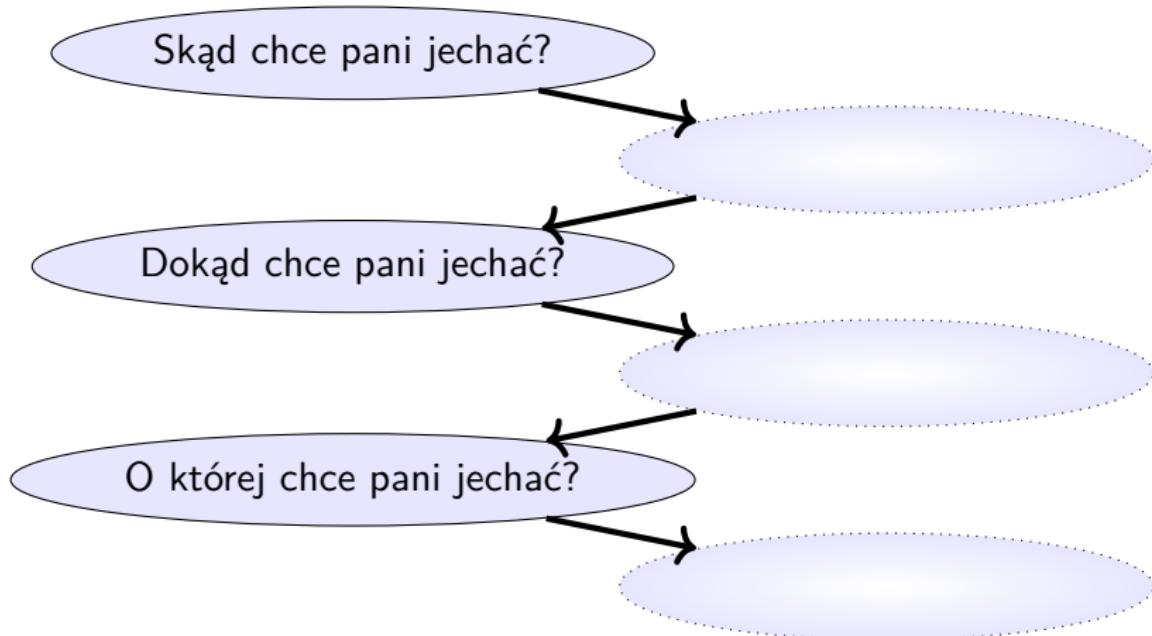
- Tekst pisany lub mowa: użytkownik i komputer rozmawiają w języku naturalnym
- Zastosowania:
 - ▶ mowa – gdy użytkownik nie może skorzystać z klawiatury, drążka sterowego itp. lub gdy jest to niewygodne, gdy ma zajęty wzrok lub kanał wzrokowy nie jest dostępny (telefonia); także tam, gdzie stosuje się tekst pisany
 - ▶ tekst pisany – gdy inne metody komunikacji są zbyt skomplikowane, np. awatary reklamujące wyroby dostępne na stronach internetowych, sexboty itp.
- Metoda gwałtownie zyskująca na popularności (Siri...)

Architektura systemów dialogowych



Możliwe są różne sposoby realizacji zarządcy rozmowy:

- oparte na automatach skończonych,
- oparte na szablonach / regułach produkcji
- oparte na modelach BDI (belief-desire-intention)



Zarządcą oparty na automatach skończonych – system z jednostronną inicjatywą, lub inaczej z inicjatywą systemu.

Szczelina	Pytanie
z_przystanku	„Skąd chce pani jechać?”
do_przystanku	„Dokąd chce pani jechać?”
o_godzinie	„O której chce pani jechać?”

Zarządca oparty na szablonach/ramach/regułach produkcji. Posiada szablony/ramki ze szczelinami na określone informacje. Jeśli nie uzyskał informacji na dany temat od użytkownika, zadaje dodatkowe pytanie.

Log in / create account

Article Discussion about the content page [Edit]

Read Edit View history Search

Hypertext

From Wikipedia, the free encyclopedia

"Metatext" redirects here. For the literary concept, see [Metafiction](#).

Hypertext is text displayed on a computer or other electronic device with references ([hyperlinks](#)) to other text that the reader can immediately access, usually by a mouse click or keypress sequence. Apart from running text, hypertext may contain tables, images and other presentational devices. Hypertext is the underlying concept defining the structure of the [World Wide Web](#).^[1] It is an easy-to-use and flexible format to share information over the [Internet](#).

Contents [hide]

- 1 Etymology
- 2 Types and uses of hypertext
- 3 History
- 4 Implementations
- 5 Academic conferences
- 6 Hypertext fiction
- 6.1 Critics and theorists
- 7 See also
- 8 References
- 9 Further reading
- 10 External links

Etymology

The prefix **hyper-** (comes from the Greek prefix "ὑπέρ-" and means "over" or "beyond") signifies the overcoming of the old linear constraints of written text. The term "hypertext" is often used where the term "[hypermedia](#)" might seem appropriate. In 1992, author [Ted Nelson](#) – who coined both terms in 1963 – wrote:

By now the word "hypertext" has become generally accepted for branching and responding text, but the corresponding word "hypermedia", meaning complexes of branching and responding graphics, movies and sound – as well as text – is much less used. Instead they use the strange term "interactive multimedia": this is four syllables longer, and does not express the idea of extending hypertext. — Nelson, [Literary Machines](#), 1992

Types and uses of hypertext

Hypertext documents can either be static (prepared and stored in advance) or dynamic (continually changing in response to user input). Static hypertext can be used to cross-reference collections of data in documents, [software applications](#), or books on CDs. A well-constructed system can also incorporate other user-interface conventions, such as menus and command lines. Hypertext can develop very complex and dynamic systems of linking and cross-referencing. The most famous implementation of hypertext is the [World Wide Web](#), first deployed in 1992.

- Wykonywane w celu:

- ▶ przekazania pomysłów użytkownikowi;
- ▶ zebraniu opinii od użytkowników;
- ▶ testowania pomysłów;
- ▶ uzyskania punktu wyjścia do realizacji.

- Rodzaje:

- ▶ jednorazowy;
- ▶ scenariusz;
- ▶ makietka;
- ▶ normalny;
- ▶ formalny.

- Wykonywany w bardzo wczesnej fazie projektowania.
- Wykonywany z użyciem tektury, kredek, kleju i taśmy klejącej, nożyczek itp.
- Może być użyty tylko do sprawdzenia ogólnych pomysłów – trudno testować w ten sposób szczegółowo.
- Wyrzucany po użyciu – nie nadaje się do wykorzystania w toku dalszych prac.

- Wykonywany w późnej fazie opracowywania systemu.
- Jest prawie wierną imitacją (przyszłego) rzeczywistego systemu, ale może używać innego sprzętu, obrazów ekranu zamiast rzeczywistych wyników dla niektórych zestawów danych itp.
- Może być testowany przez rzeczywistych użytkowników w rzeczywistych warunkach.
- Testuje szczegóły.
- Poszczególne części mogą być użyte w realizacji.

- Używany w każdej fazie opracowywania systemu.
- Jest to system w trakcie opracowywania; pewne części już działają, a inne jeszcze nie.
- Może używać prototypowego sprzętu.
- Zorientowany zarówno na ogólne pomysły, jak i na szczegóły.
- Może być w większości wykorzystany do dalszych prac nad systemem.

- Używa formalnej specyfikacji (np. GOMS, UAN, UML itp.) do opisania systemu.
- Używany do sprawdzania zupełności i spójności.
- Chociaż opisuje interfejs, w niczym go nie przypomina.
- Nie może być testowany przez rzeczywistego użytkownika.

- Używany we wczesnej fazie projektowania.
- Używa opisu językiem naturalnym.
- Opisywane są zadania i sposób ich wykonania krok po kroku.
- Użytkownicy mogą go czytać we własnym tempie; łatwo go komentować.
- Nie może być testowany w rzeczywistych warunkach; użytkownicy muszą sobie wyobrażać rzeczywiste działanie.
- Może stanowić zaczątek podręcznika użytkownika.

- Pomoc, szkolenia, podręczniki i obsługa po sprzedaży stanowią także część interfejsu; często się o niej zapomina.
- Ta część interfejsu powinna być rozwijana równolegle do innych.
- Podręcznik może stanowić doskonały prototyp – scenariusz.
- Obsługa po sprzedaży może pozwolić określić części systemu wymagające poprawy.

- **Podręcznik użytkownika** (ang. *user's guide*).

- ▶ Opisuje krok po kroku zadania do wykonania (np. jak wydrukować dokument), a nie elementy interfejsu (np. wykaz elementów okna dialogowego wydruku). Podaje przykłady użycia.
- ▶ Powinien być pisany językiem użytkownika przez specjalistę od pisania podręczników.
- ▶ Nie powinien zawierać szczegółów technicznych, historii itp. - zwiększą niepotrzebnie objętość. Im dłuższy podręcznik, tym trudniej coś w nim znaleźć.
- ▶ Powinien zawierać spis treści i skorowidz, a także zrzuty ekranów. Skorowidz powinien zawierać także inne możliwe nazwy czynności.

- **Encyklopedia użytkownika** (ang. *reference manual*).
 - ▶ Zawiera opis wszystkich elementów interfejsu (menu, okna dialogowe, itp.).
- **Ściąga** (ang. *quick reference*).
 - ▶ Zawiera zbiór najważniejszych / najczęściej używanych poleceń systemu. Mieści się na ogół na pojedynczej kartce papieru.
- **Samouczek** (ang. *tutorial*).
 - ▶ Zawiera dokładny opis krok po kroku przykładowych podstawowych operacji w systemie.

- Użytkownik coraz rzadziej ma czas na czytanie podręczników.
- Pomoc powinna być w możliwie największym stopniu zależna od kontekstu. Jeżeli użytkownik odwołuje się do pomocy w oknie dialogowym lub po pojawienniu się komunikatu o błędzie, widocznie potrzebuje dodatkowej informacji na ten temat.
- Informacje powinny być powiązane ze sobą za pomocą odnośników. W ten sposób unika się powtarzania tych samych informacji, a można umieścić ich więcej.
- Należy szczególną uwagę zwrócić na system poruszania się po systemie pomocy. Użytkownik nie powinien się w nim gubić.
- System pomocy także powinien mieć spis treści i skorowidz.
- Krótkie opisy w „dymkach” pozwalają użytkownikowi szybko zorientować się w znaczeniu elementów interfejsu, np. pictogramów czy przycisków.

Inne formy pomocy

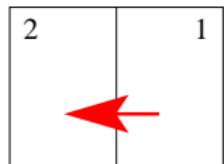
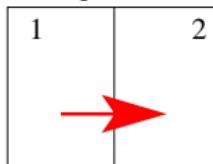
- Szkolenia pozwalają na bezpośredni kontakt z użytkownikiem, przyjrzenie się jego pracy i popełnianym błędom.
- Pomoc telefoniczna pozwala zorientować się, które części interfejsu sprawiają użytkownikom najwięcej kłopotów. Może być to bodziec do zmian w systemie i w podręcznikach oraz systemie pomocy.
- Listy i grupy dyskusyjne pozwalają zorganizować społeczność użytkowników, tak aby mogli pomagać sobie nawzajem. Mogą też być źródłem inspiracji dla nowych funkcji systemu w jego przyszłych wersjach.

- Postrzeganie.
 - ▶ Zmysły ludzkie mają swoje cechy i swoje ograniczenia
- Zróżnicowanie kulturowe.
 - ▶ Wszyscy jesteśmy równi, ale nie tacy sami
- Środowisko pracy.
 - ▶ Gdzie program/system ma być wykorzystywany
- Osobowość.
 - ▶ Inaczej dla młodych, inaczej dla wcześniej urodzonych
- Użytkownicy specjalnej troski.
 - ▶ Użytkownicy upośledzeni potrzebują specjalnego traktowania

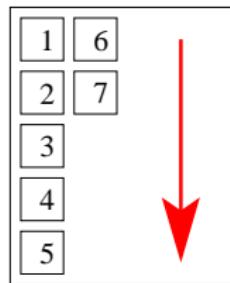
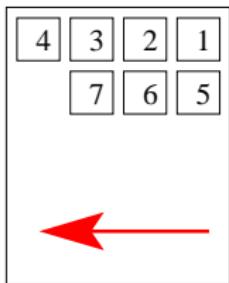
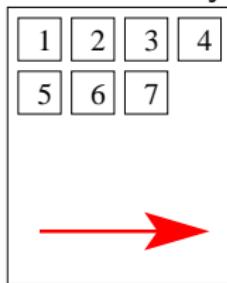
- Zestaw znaków

- ▶ angielski: abcdef,
- ▶ polski: aąbcćdeeęf,
- ▶ grecki: $\alpha\beta\gamma\delta\dots$

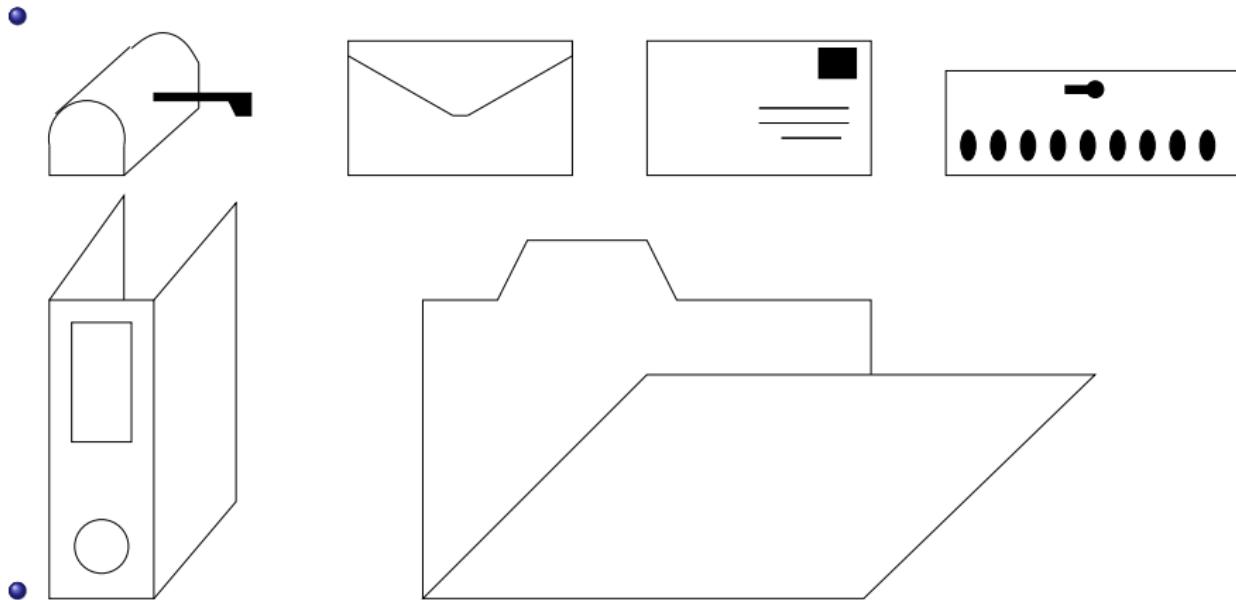
- Kolejność stron



- Układ strony



- miejsce symbolu waluty: 20 zł, \$20;
- czas i data: 22:31, 10:31 PM; 27/02/2007, 2007/02/27, 02/27/2007;
- separator części ułamkowej: 1.234,75 1,234.75.
- system miar: metryczny – 1cm, imperialny – 1"



Zróżnicowanie kulturowe – symbole



R 125



R 126



R 127



R 128



W 314



W 315



W 316



W 317