

**Wydział Zarządzania AGH**

Katedra Informatyki Stosowanej



# System operacyjny GUI

Podstawy informatyki

# Systemy operacyjne i GUI

- ❑ Co to jest system operacyjny?
- ❑ Systemy operacyjne dla komputerów osobistych
- ❑ Graficzny interfejs użytkownika
- ❑ Windows jako przykład GUI

# System operacyjny

- ❑ **Jest zestawem programów umożliwiającym zarządzanie komputerem oraz danymi w pamięci operacyjnej i zewnętrznej.**

Podstawowymi zadaniami SO są:

- ❑ harmonogramowanie zadań,
- ❑ interpretowanie poleceń,
- ❑ sterowanie zadaniami i zasobami (przede wszystkim pamięcią operacyjną i zewnętrzną),
- ❑ ochrona zasobów,
- ❑ obsługa błędów,
- ❑ obsługa wejścia-wyjścia,
- ❑ udostępnianie interfejsu.

# Podstawowe cechy SO

- ❑ *przetwarzanie wsadowe*

zadania (zlecenia użytkownika) czekają w kolejce na przetworzenie: jedno z nich jest przetwarzane, przy czym zadanie, którego przetwarzanie zostało raz zainicjowane musi być przetwarzone do końca,

- ❑ *wielozadaniowość*

jest to sposób wykorzystania komputera, pozwalający wykonywać więcej niż jeden program w taki sposób, że żaden z nich nie musi być zakończony, zanim jakiś inny zostanie rozpoczęty lub wznowiony (pojęcia związane: podział czasu i podział pamięci),

# Podstawowe cechy SO

- ❑ *wielodostęp*

bezpośrednie wykorzystanie komputera przez co najmniej dwóch użytkowników jednocześnie (pozornie jednocześnie) w taki sposób, że żaden z nich nie musi być świadomy postępowania pozostałych użytkowników,

- ❑ *pamięć wirtualna*

mechanizm wykorzystania pamięci zewnętrznej polegający na przetrzymywaniu w niej części programu i wczytywaniu go do pamięci operacyjnej w razie potrzeby, co umożliwia wykonywanie programu o rozmiarze większym niż dostępna pamięć operacyjna.

# Podstawowe cechy SO

- ❑ *program*

sekwencja instrukcji zapisana w pliku dyskowym zamieniająca się w instancję programu (proces) w momencie załadowania do pamięci operacyjnej komputera,

- ❑ *proces (zadanie)*

to każda uruchomiona aplikacja; proces obejmuje kod programu, prywatną przestrzeń adresową oraz przynajmniej jeden wątek,

- ❑ *wątek*

składnik procesu (ciąg rozkazów), który jest wykonywany (*unit of execution*); wątek korzysta z zasobów oraz przestrzeni adresowej procesu, do którego należy.

# Podstawowe cechy SO

## ❑ *Zarządzanie procesami*

jest podstawowym zadaniem SO i obejmuje:

- alokowanie zasobów do procesów,
- organizację wymiany danych między procesami,
- ochrona zasobów procesów przed niepożądanym wpływem innych procesów,
- synchronizację współpracy procesów.

## ❑ Procesy nie działają w odosobnieniu:

1. współdziałają, by wykonać prace zleczone przez użytkownika,
2. współzawodniczą o korzystanie z ograniczonych zasobów (procesory, miejsce w pamięci operacyjnej, pliki).

## ❑ Współdziałanie i współzawodnictwo wymagają ustalonego sposobu komunikacji między procesami.

# Podstawowe cechy SO

- ❑ Zasoby systemowe dzielimy na podzielne (procesor, pliki do odczytu) i niepodzielne (jeden zasób - jeden proces); niepodzielność może wynikać:
  - natura fizyczna, która sprawia, że nie praktykuje się jego dzielenia (np. drukarka),
  - wykorzystanie współbieżne mogłoby powodować wzajemne zakłócanie czynności procesów (np. miejsce w pamięci operacyjnej).
- ❑ System operacyjny zapewnia (w odniesieniu do procesów):
  - **wzajemne wyłączenie**, by tylko jeden proces mógł korzystać w danym momencie z zasobów niepodzielnych,
  - **synchronizację**, by uzyskać zadowalającą współpracę (proces nie może przejść poza wyznaczone punkty, zanim drugi proces nie zakończy swego działania),
  - **niedopuszczenie do blokady** procesów.



# Podstawowe cechy SO

## ❑ *wielowątkowość*

oznacza zdolność SO do obsługi wielu wątków w ramach jednego procesu. Np. każde okno Eksploratora Windows stanowi odrębny wątek. Gdy użytkownik rozpocznie kopiowanie czy przenoszenie plików z jednego okna do drugiego, operacja ta jest obsługiwana przez okno docelowe. W tym czasie, nie czekając na zakończenie kopiowania, możemy w innym oknie wykonywać inne czynności.

- ❑ polega na dzieleniu czasu pracy procesora na krótkie okresy (rzędu milisekund) i przydzielaniu ich poszczególnym wątkom. Kiedy czas przeznaczony dla danego wątku mija, kontekst (stan) jego pracy jest zapamiętywany, a jego miejsce zajmuje kontekst następnego wątku (o najwyższym priorytecie spośród wątków będących w stanie gotowości). Przełączanie pomiędzy wątkami realizują tzw. *schedulery*.

Źródło: [old.wb.pb.edu.pl/Materialy/KOS/Andraka%20Dariusz/pinfo7b.pdf](http://old.wb.pb.edu.pl/Materialy/KOS/Andraka%20Dariusz/pinfo7b.pdf)

# Warstwowa budowa SO



Jądro (kernel) - część systemu operacyjnego, która przyjmuje kierowane do niego zlecenia od programów użytkowych oraz użytkownika komputera i wykonuje je przydzielając im zasoby komputera i urządzenia zewnętrzne

Powłoka (shell) - specjalny program interpretujący polecenia kierowane do jądra systemu

Programy użytkowe to programy do obsługi plików, komunikacji w sieci, prostej edycji tekstu i grafiki, programy diagnozujące pracę komputera, translatory języków programowania itp.

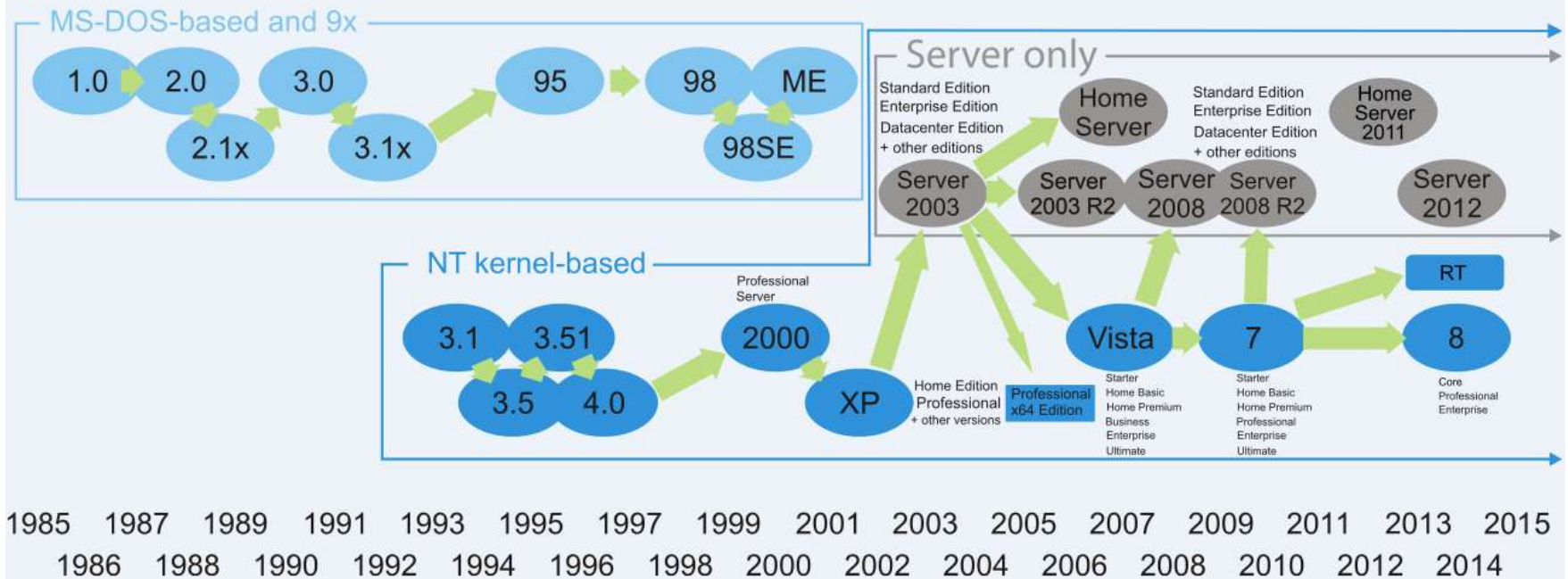
# SO dla komputerów osobistych

- ❑ jednoprogramowe i jednodostępne DOS (Disk Operating System): [MS-DOS](#), [PC-DOS](#), itp.,
- ❑ [UNIX](#), [LINUX](#) - uniwersalny system operacyjny pozwalający na wielodostęp i wieloprogramowość,
- ❑ [Windows](#), [OS/2](#), [MAC OS](#), itp. - systemy zorientowane graficznie, w których opcje dostępne dla użytkownika są zawarte w liście (menu) bądź reprezentowane przez odpowiednio dobrane piktogramy; systemy wieloprogramowe, jedno- lub wielodostępne.
- ❑ [Google Chrome OS](#) – oparty na Ubuntu, szybki i lekki - uruchamia się w przeciągu kilku sekund; bardzo prosty interfejs.

# Historia Windows

## Microsoft Windows

family tree



# Interfejs użytkownika

- **Interfejs użytkownika** (UI) jest granicą między komputerem a osobą wykorzystującą go do pewnych zadań. W tym ujęciu interfejs jest zarówno urządzeniem (urządzeniami) jak i sposobem jego (ich) wykorzystania. W przypadku komputera osobistego na standardowy interfejs użytkownika składa się ekran, klawiatura i mysz oraz programy służące do komunikowania się człowieka z komputerem przy pomocy tych urządzeń.

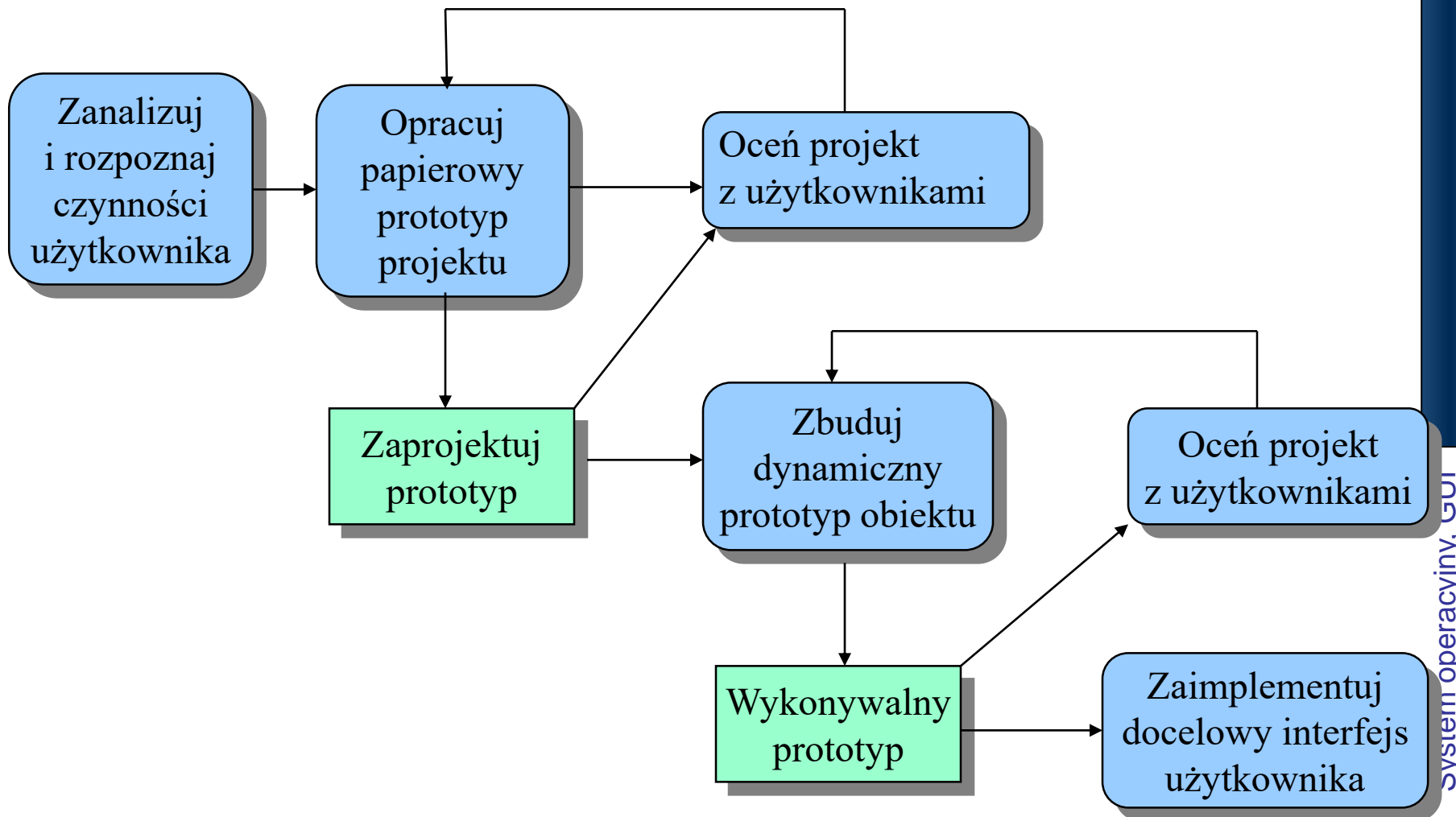
Następne 5 slajdów

Źródło: *Ian Sommerville, Inżynieria oprogramowania, WNT 2005.*

# Interfejs użytkownika

- ❑ Dobry projekt interfejsu użytkownika jest niezbędnym warunkiem prawidłowego użytkowania systemu.
- ❑ Interfejs trudny w użyciu z pewnością doprowadzi do wielu pomyłek użytkowników.
- ❑ W najgorszym wypadku użytkownicy odmówią używania systemu oprogramowania niezależnie od jego funkcjonalności.
- ❑ Jeśli informacja jest przedstawiona w sposób zagmatwany, użytkownicy na pewno źle zrozumieją znaczenie systemu.
- ❑ Mogą wykonać ciągi poleceń, które uszkodzą dane lub doprowadzą do awarii systemu.

# Projektowanie interfejsu użytkownika



# Projektowanie interfejsu użytkownika

<b>Zasada</b>	<b>Opis</b>
❑ Zbliżenie do użytkownika	Interfejs powinien posługiwać się pojęciami i kategoriami wziętymi z doświadczeń osób, które najczęściej będą korzystać z systemu.
❑ Spójność	Interfejs powinien być spójny, tzn. podobne operacje powinny być wykonywane w ten sam sposób.
❑ Minimum niespodzianek	Użytkownicy nie powinni być zaskakiwani zachowaniem systemu.
❑ Możliwość wycofania	Interfejs powinien obejmować mechanizmy, które umożliwiają użytkownikom wycofanie się z błędów.
❑ Porady dla użytkownika	Interfejs powinien przekazywać znaczące informacje zwrotne, gdy dochodzi do błędów; powinien też oferować pomoc, której treść zależy od kontekstu.
❑ Rozróżnianie użytkowników	interfejs powinien oferować udogodnienia do interakcji dostosowane do różnych rodzajów użytkowników systemu.



# Projektowanie interfejsu użytkownika

- Spójny interfejs użytkownika musi integrować interakcję użytkownika i prezentację informacji.
- Rodzaje interakcji:
  - działanie bezpośrednie,
  - wybór z menu,
  - wypełnianie formularza,
  - język poleceń,
  - język naturalny.

# Projektowanie interfejsu użytkownika

<b>Sposób interakcji</b>	<b>Główne zalety</b>	<b>Główne wady</b>	<b>Przykład zastosowań</b>
Bezpośrednie działanie	Szybka i intuicyjna interakcja Łatwe do nauczenia	Może być trudna do implementacji Odpowiednie jedynie wówczas, gdy istnieje graficzne wyobrażenie czynności i obiektów	Gry wideo Systemy CAD
Wybór z menu	Umożliwia uniknięcie błędów użytkownika Wymaga mało pisania	Zbyt powolny dla doświadczonych użytkowników Może być skomplikowany, gdy opcji menu jest dużo	Większość systemów ogólnego
Wypełnianie formularza	Proste wprowadzenie danych Łatwe do nauczenia	Zajmuje duży obszar ekranu	Zarządzanie magazynem Przetwarzanie kredytów dla ludności
Język poleceń	Solidny i elastyczny	Trudny do nauczenia Małe możliwości obsługi błędów	Systemy operacyjne Systemy wydobywania informacji bibliotecznych
Język naturalny	Dostępny dla przypadkowych użytkowników Łatwy do rozszerzenia	Wymaga więcej pisania Systemy rozpoznające język naturalny są zawodne	Systemy rozkładów jazdy Systemy określania informacji WWW

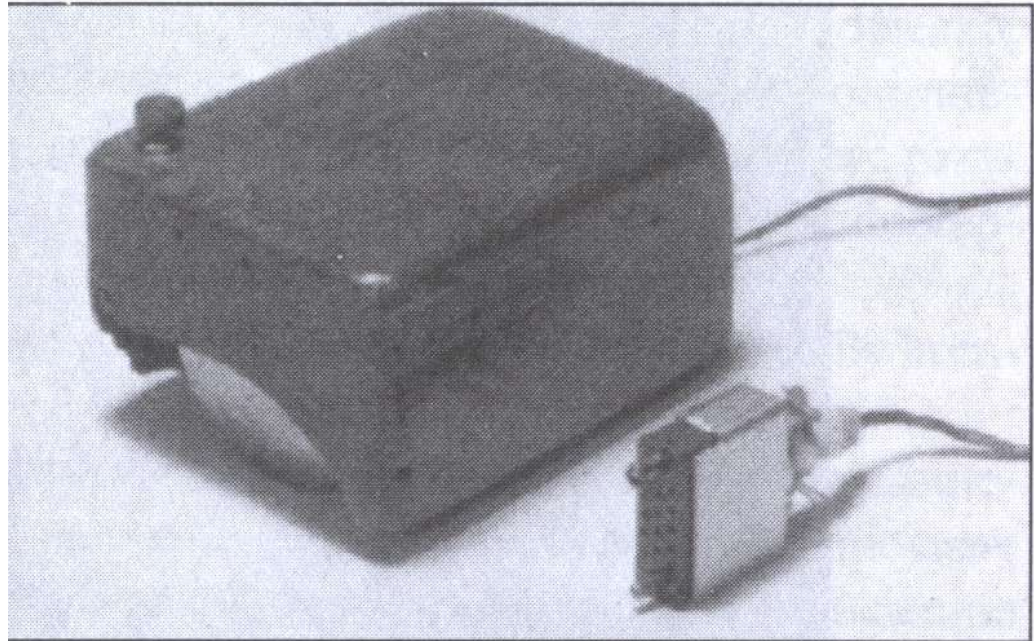
# Graficzny interfejs użytkownika

Cechy wyróżniające GUI:

- ❑ jednolity interfejs użytkownika,
- ❑ wyświetlanie w kolorze i wysokiej rozdzielczości,
- ❑ zasada: to co widzisz na ekranie, dostaniesz naprawdę (ang. WYSIWYG - What you see is what you get),
- ❑ bezpośrednie sterowanie.

# Elementy interfejsu

- ❑ **pulpit**
- ❑ **okno**
- ❑ **ikona**
- ❑ **menu**
- ❑ **wskazywanie**



Drewniana mysz Engelbarta

# GUI – jednolity wygląd i obsługa

## ❑ Było:

We wczesnych latach osiemdziesiątych powszechne było wymyślanie dla każdej nowej aplikacji unikalnego interfejsu użytkownika (Lotus 1-2-3, WP, dBase). Stąd każda nowa aplikacja wymagała od użytkownika nauczenia się jej od podstaw. Wyuczone sposoby obsługi jednej aplikacji były zupełnie nieprzydatne w przypadku innej. Użytkownicy PC poprzestawali na nauczeniu się i używaniu dwóch aplikacji.

## ❑ Jest:

Jednolity interfejs użytkownika w postaci GUI określa standard porozumiewania się z komputerem w celu osiągnięcia określonych efektów. Używanie rozwijalnych menu połączone z systemem pomocy pozwala użytkownikowi na badanie programu i odkrywanie dostępnych poleceń bez konieczności czytania dokumentacji.

# GUI – jednolity wygląd i obsługa

- ❑ Co więcej, ponieważ wszystkie aplikacje używają tej samej struktury, nauczenie się jednej pozwala na korzystanie z innych z równym powodzeniem. Staje się to istotne, jeśli wyobrazimy sobie kilka aplikacji, każdą w swoim oknie, działających w tym samym czasie i na tym samym ekranie.
- ❑ Istnienie jednolitego interfejsu użytkownika jest w dzisiejszym świecie nie tyle udogodnieniem ile koniecznością. Dobrze zaprojektowane GUI są, w połączeniu z intuicją użytkownika, potężnym narzędziem, i to w stopniu nieosiągalnym przez jakikolwiek tradycyjny interfejs terminalowy.

# GUI – kolor i rozdzielczość

## ❑ Było:

Interfejs oparty na znakach; przykład systemu rezerwacji miejsc w samolotach.

## ❑ Jest:

W świecie grafiki ekran dodatkowo pokazuje obrazy, będące modelami świata rzeczywistego. GUI przedstawia ten model w formie bezpośrednio zrozumiałej przez użytkownika (tzw. rzeczywistość wirtualna). Zamiast przekształcać go na cyfry i litery, próbuje przedstawiać świat takim, jaki jest on naprawdę.

# GUI – kolor i rozdzielczość

- ❑ W środowisku graficznym wymagania stawiane przed komputerami są bardzo duże. Pamięć ekranu musi wynosić ponad **XXX** MB, a komputer musi mieć wystarczającą moc, by manipulować tą ilością informacji wielokrotnie w ciągu sekundy.
- ❑ GUI wymaga operowania wielką ilością informacji, wielokrotnie częściej, w bardziej złożony sposób, a przy tym - w sposób niezauważalny dla użytkownika. Jedynym sposobem na podołanie takim wymaganiom było dostarczenie każdemu użytkownikowi oddzielnego komputera.



# GUI – WYSIWYG

- GUI jako przykład rzeczywistości wirtualnej jest często określany mianem WYSIWYG. Termin ten najczęściej odnosi się do procesorów tekstu zapewniających zgodność tego, co widzimy na ekranie z tym, co otrzymujemy po wydrukowaniu. Jednakże można ten termin rozumieć znacznie szerzej:  
**to co widzisz, jest w miarę dokładnym modelem rzeczywistości.**

# GUI – bezpośrednie sterowanie

- Dawniej użytkownik sterował przedstawionym światem za pomocą słownych komend zorganizowanych w hierarchiczne struktury (przykład: zmiana szerokości kolumny w dawnych arkuszach kalkulacyjnych), teraz steruje częściami świata wirtualnego bezpośrednio. Koncepcja prosta i intuicyjnie zrozumiała, a jej zastosowanie w różnych aplikacjach daje bardzo dobre rezultaty.
- Omówione przykłady ilustrują jedną z technik bezpośredniego sterowania zwaną *przeciągnij i zostaw* (drag and drop), polegającą na wybraniu obiektu oraz przeciągnięciu go do nowego miejsca. Innym przykładem może być kasowanie obiektu przez wrzucenie go do kosza na śmieci czy pomalowanie obiektu po naciśnięciu obrazka z puszką farby.

# GUI – bezpośrednie sterowanie

- Każdy obiekt w świecie wirtualnym, podobnie jak w świecie rzeczywistym, ma określone właściwości (cechy); wiele z nich jest widocznych, ale by poznać cechy pośrednie stosowane *podwójne kliknięcie* jest technika inspekcji. Inspekcja umożliwia natychmiastowy wgląd w cechy obiektu i ich modyfikację.