



Systemy Operacyjne

dr inż. Andrzej Opaliński

AKADEMIA GÓRNICZO-HUTNICZA IM. STANISŁAWA STASZICA W KRAKOWIE
AGH UNIVERSITY OF SCIENCE AND TECHNOLOGY





Trochę o sobie

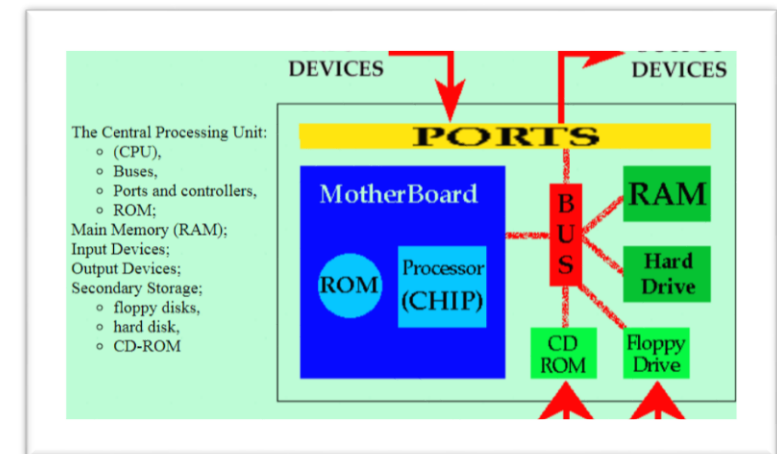
- dr inż. Andrzej Opaliński
- Pokój 413, budynek B5
- Email: andrzej.opalinski@agh.edu.pl
- WWW
 - <http://home.agh.edu.pl/~opal>
 - lub w google: „Andrzej Opaliński AGH” lub „opal agh”
 - strony przedmiotów – aktualności, materiały, harmonogram
- Tematyka dydaktyczno-naukowa
 - Sieci komputerowe
 - Systemy robotów autonomicznych
 - Współczesne modele oprogramowania
 - Przeszukiwania sieci WEB (crawling, scraping)
- Projekty dyplomowe

Po co ten wykład ?

- Kiedyś nie było go w programie
- Przygotowanie do sieci komputerowych (out of scope)
- Zawartość przedmiotu – W1, L1+L2
- Różny poziom zaawansowania uczestników studiów
- Ocena końcowa a cel przedmiotu

Jak działa komputer ?

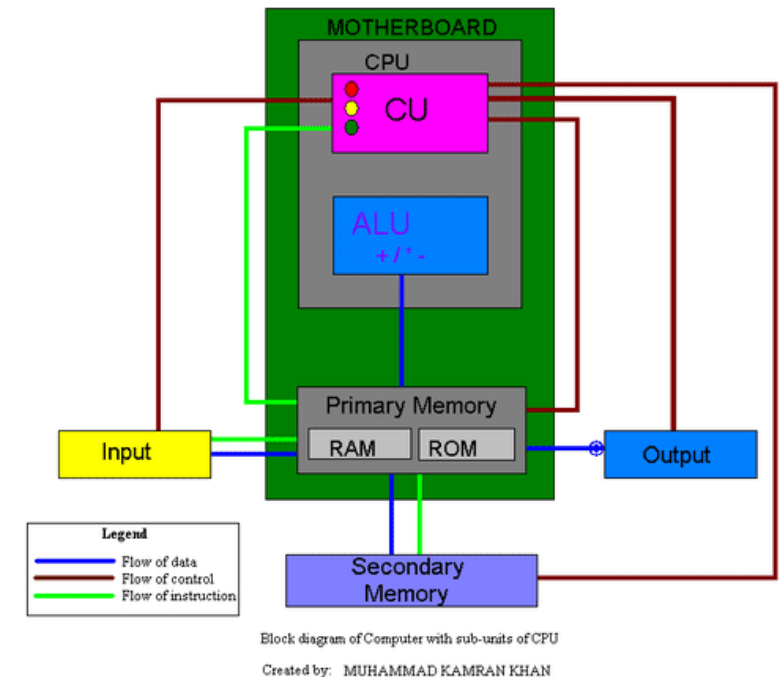
- Problemy dydaktyka – umieć a uczyć
- Wersja dla gimnazjum vs architektury komputerów
- Stara szkoła:



- Nowa wersja – prosto, ładnie, kolorowo
- Pomijamy:
 - Elektronikę – sygnały, obwody, układy logiczne
 - Kodowanie sygnałów – system binarny, kodowanie tekstu, multimediiów jako sygnał binarny
- 2 filmy – 9 minut...

Jak działa komputer - podsumowanie

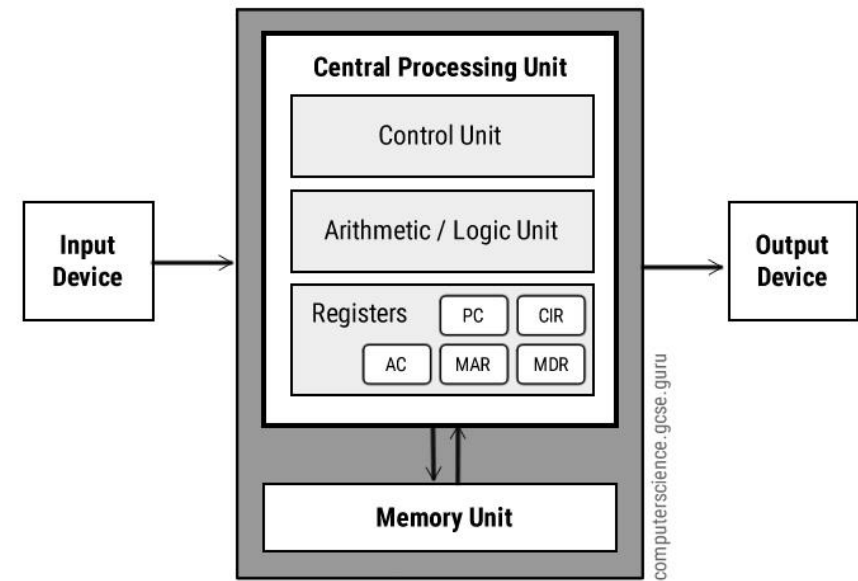
- Integracja systemów wejścia i wyjścia (przykłady)
- Zadania procesora
 - Wysokopoziomowo – wykonywanie instrukcji programu
- Uwzględniając I/O
 - Sterowniki - obsługa urządzeń
 - Kod maszynowy -> konwersja kodu programów (użytkowych, sterowników urządzeń) na operacje wykonywalne przez procesor
- Podstawowe funkcje
 - Operacje logiczno-arytmetyczne
 - Przesyłanie danych pomiędzy podzespołami komputera (pamięć, I/O)



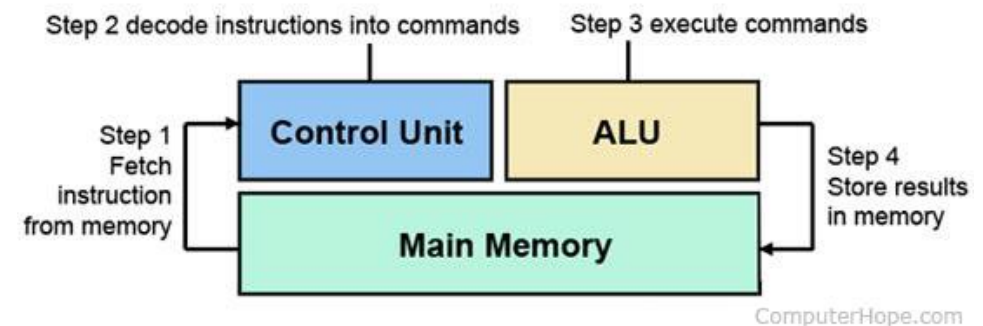
Processor

- Budowa

- Control Unit (układ sterujący)
(przetwarzanie rozkazów programów, sterowanie urządzeniami, obsługa przerwań)
- ALU (jednostka arytmetyczno-logiczna)
(dodawanie, odejmowanie, porównywanie liczb, AND, OR, XOR, NOT, >>, <<)
- Rejestry
(przechowywanie: danych, wyników, adresów, instrukcji)

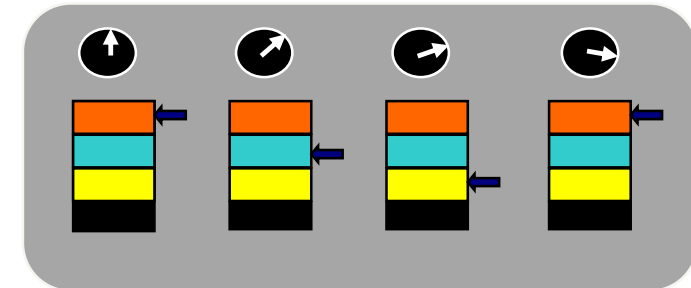


Machine Cycle

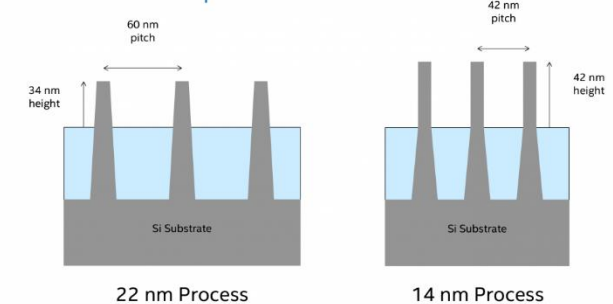


Współbieżność, wielowątkowość

- Wielozadaniowość
 - systemy jednozadaniowe – (do lat 90tych, MSDOS)
 - systemy wielozadaniowe – początek lat 90tych (Windows, Unix)
 - pierwotnie – pseudowspółbieżność - dzielenie czasu procesora pomiędzy wątki procesów
 - aktualnie – wielordzeniowe (wielowątkowe) procesory
- Pierwsze procesory wielordzeniowe:
 - Power4 (IBM) – 2001r
 - Opteron (AMD), Pentium Extreme (Intel) – 2005r
- Szybkość taktowania procesorów - aktualnie do 4,4GHz
- Proces technologiczny (mikroarchitektury)
 - 2006r – Core – 65nm,
 - 2011r – SandyBridge – 32nm
 - 2015r – Skylake - 14nm,
 - 2019r – 7nm (AMD, Zen2)



Transistor Fin Improvement



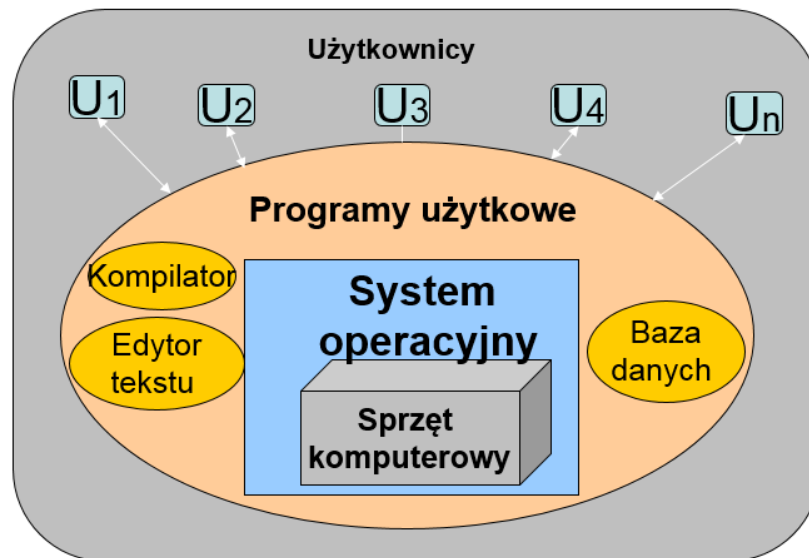
Taller and Thinner Fins for Increased Drive Current and Performance

System operacyjny - definicje

System operacyjny jest to zorganizowany zespół programów, które pośredniczą między sprzętem a użytkownikami, dostarczając użytkownikom zestawu środków ułatwiających projektowanie, kodowanie, uruchamianie i eksploatację programów oraz w tym samym czasie sterują przydziałem zasobów dla zapewnienia efektywnego działania. (Alan Shaw)

System operacyjny jest programem, który działa jako pośrednik między użytkownikiem komputera a sprzętem komputerowym. Zadaniem systemu operacyjnego jest tworzenie środowiska, w którym użytkownik może wykonywać programy w sposób wygodny i wydajny. (Abraham Silberschatz)

System operacyjny jest warstwą oprogramowania operującą bezpośrednio na sprzęcie, której celem jest zarządzanie zasobami systemu komputerowego i stworzenie użytkownikowi środowiska łatwiejszego do zrozumienia i wykorzystania. (Andrew Tanenbaum)

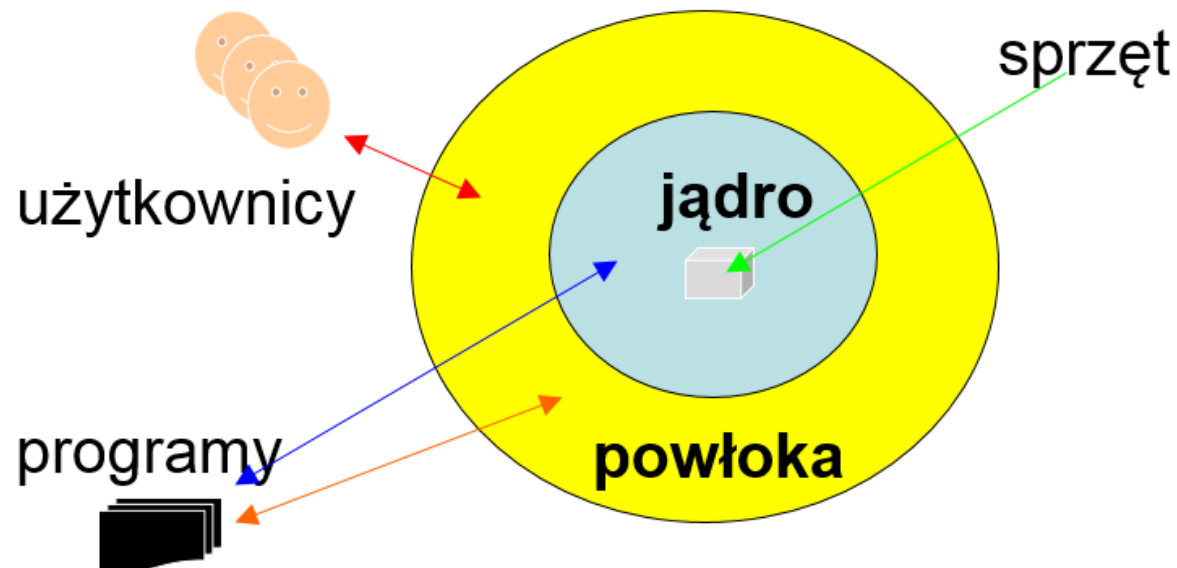


We wszystkich definicjach zderzają się z sobą dwie zasadnicze kwestie:



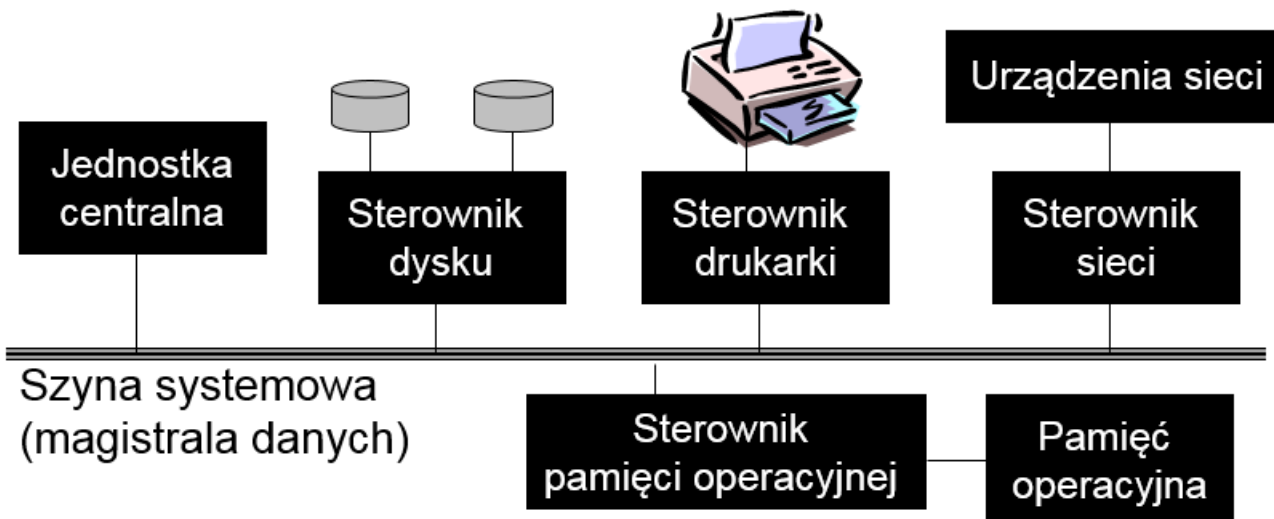
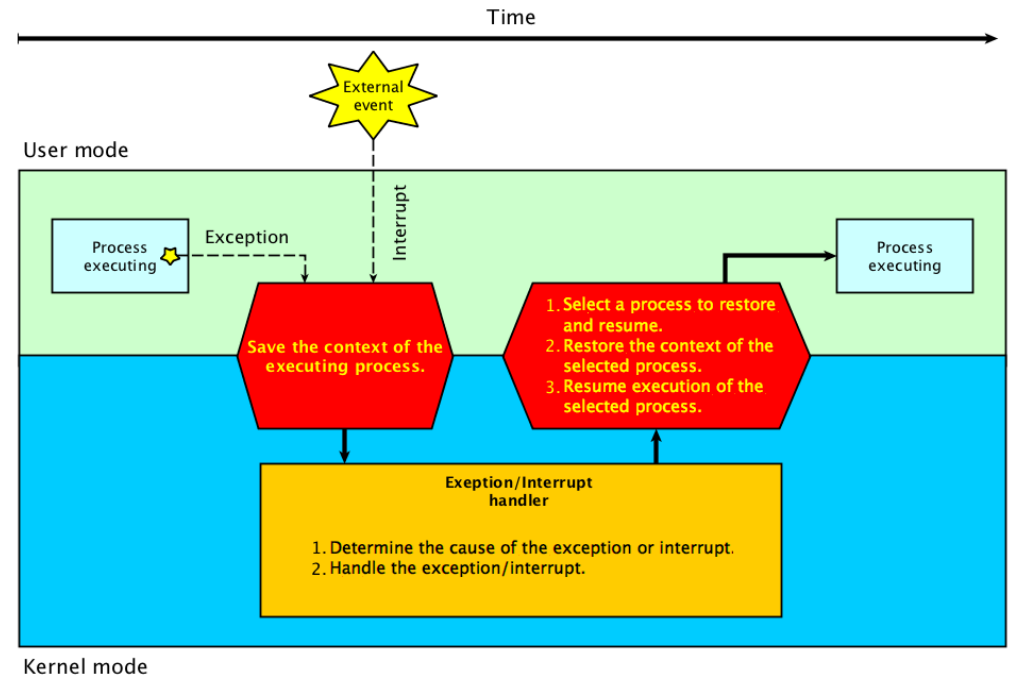
Budowa systemu operacyjnego

- jądro - komunikuje się z komputerem przez sterowniki urządzeń i wykonuje kolejkowanie zadań, obsługę pamięci
- powłoka - stanowi interpreter poleceń systemu (komunikacja z użytkownikiem)
- programy - polecenia systemowe nie zawarte w jądrze, programy narzędziowe, programy użytkowe



Jak działa system operacyjny ?

- Fizyczna budowa komputera
- Sterowanie przerwaniem
 - Od użytkownika
 - Od urządzeń
 - Błędy...



Zadania systemu operacyjnego

- Definiowanie interfejsu użytkownika:

- zbiór poleceń (linia komend),
 - system okienkowy - interakcję z systemem komputerowym.

- Udostępnianie systemu plików:

organizuje i ułatwia dostęp do informacji np. w postaci hierarchicznego systemu plików.

- Udostępnianie środowiska do wykonywania programów:

- struktury danych do wykonywania programu (procesy i przełączanie kontekstu).
 - mechanizmy komunikacji pomiędzy procesami (komunikaty, strumienie, pamięć)
 - mechanizmy synchronizacji procesów.

- Sterowanie urządzeniami wejścia-wyjścia:

- inicjalizacja pracy urządzeń zewnętrznych
 - pośredniczą w efektywnym przekazywaniu danych pomiędzy jednostką centralną a tymi urządzeniami.

- Obsługa podstawowej klasy błędów:

- błędy użytkowników (np. niedostępność zasobów, brak prawa dostępu)
 - programistów (np. błąd dzielenia przez 0, naruszenie ochrony pamięci)
 - systemu (np. brak komunikacji z urządzeniem)



Pytanie

- Czy komputer mógłby obyć się bez systemu operacyjnego ?
(rozumianego według poprzednich definicji)

- Czy komputer mógłby obyć się bez systemu operacyjnego ?
(rozumianego według poprzednich definicji)
- Odpowiedź: Tak, ale...

- Czy komputer mógłby obyć się bez systemu operacyjnego ?
(rozumianego według poprzednich definicji)
- Odpowiedź: Tak, ale...
 - Programy użytkowe musiałyby zawierać wszelkie procedury obsługi pamięci, urządzeń wejścia i wyjścia, dysków itd.

- Czy komputer mógłby obyć się bez systemu operacyjnego ?
(rozumianego według poprzednich definicji)
- Odpowiedź: Tak, ale...
 - Programy użytkowe musiałyby zawierać wszelkie procedury obsługi pamięci, urządzeń wejścia i wyjścia, dysków itd.
 - Programy musiałyby sprawdzać czy urządzenia są gotowe, czy nie są aktualnie wykorzystywane przez inne programy,

- Czy komputer mógłby obyć się bez systemu operacyjnego ?
(rozumianego według poprzednich definicji)
- Odpowiedź: Tak, ale...
 - Programy użytkowe musiałyby zawierać wszelkie procedury obsługi pamięci, urządzeń wejścia i wyjścia, dysków itd.
 - Programy musiałyby sprawdzać czy urządzenia są gotowe, czy nie są aktualnie wykorzystywane przez inne programy,
 - Programy musiałyby znać organizację danych na dysku, protokół komunikacji sieciowej itd.

- Czy komputer mógłby obyć się bez systemu operacyjnego ?
(rozumianego według poprzednich definicji)
- Odpowiedź: Tak, ale...
 - Programy użytkowe musiałyby zawierać wszelkie procedury obsługi pamięci, urządzeń wejścia i wyjścia, dysków itd.
 - Programy musiałyby sprawdzać czy urządzenia są gotowe, czy nie są aktualnie wykorzystywane przez inne programy,
 - Programy musiałyby znać organizację danych na dysku, protokół komunikacji sieciowej itd.
 - Po przeniesieniu na inny komputer, trzeba by w programach zmienić większość procedur obsługi sprzętu,

- Czy komputer mógłby obyć się bez systemu operacyjnego ?
(rozumianego według poprzednich definicji)
- Odpowiedź: Tak, ale...
 - Programy użytkowe musiałyby zawierać wszelkie procedury obsługi pamięci, urządzeń wejścia i wyjścia, dysków itd.
 - Programy musiałyby sprawdzać czy urządzenia są gotowe, czy nie są aktualnie wykorzystywane przez inne programy,
 - Programy musiałyby znać organizację danych na dysku, protokół komunikacji sieciowej itd.
 - Po przeniesieniu na inny komputer, trzeba by w programach zmienić większość procedur obsługi sprzętu,
 - Istniałyby ogromne problemy ze standaryzacją i kompatybilnością.

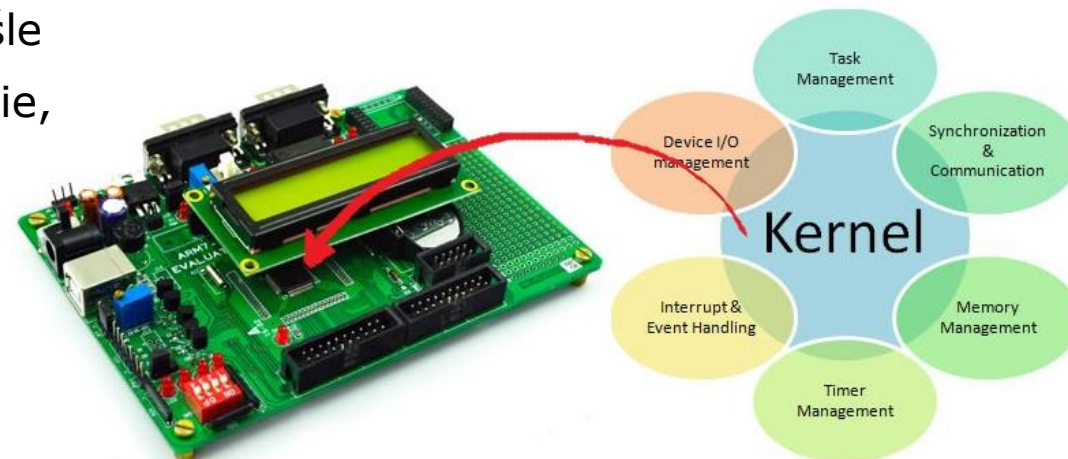
- Czy komputer mógłby obyć się bez systemu operacyjnego ?
(rozumianego według poprzednich definicji)
- Odpowiedź: Tak, ale...
 - Programy użytkowe musiałyby zawierać wszelkie procedury obsługi pamięci, urządzeń wejścia i wyjścia, dysków itd.
 - Programy musiałyby sprawdzać czy urządzenia są gotowe, czy nie są aktualnie wykorzystywane przez inne programy,
 - Programy musiałyby znać organizację danych na dysku, protokół komunikacji sieciowej itd.
 - Po przeniesieniu na inny komputer, trzeba by w programach zmienić większość procedur obsługi sprzętu,
 - Istniałyby ogromne problemy ze standaryzacją i kompatybilnością.
- Tego typu oprogramowanie nazywa się **FIRMWARE** – oprogramowanie zainstalowane na stałe w urządzeniu, zapewniające podstawowe procedury jego obsługi

Gdzie znajdziemy systemy operacyjne ?

- Komputery (serwery, desktopy, laptopy, terminale)
- Telefony, tablety
- IoT
 - Smart home (oświetlenie, ogrzewanie, monitoring)
 - Mini-komputery (Arduino, RaspberryPi, IntelGalileo)
 - Wearables (smartwatche, smartglasses, smartbands, smartclothes)
- Urządzenia – firmware
 - dedykowane do konkretnego modelu sprzętu/zastosowań
 - ograniczona interakcja z użytkownikiem i wsparcie różnych peryferiów I/O
 - Przykłady:
 - Sprzęt sieciowy – routery, switche, modemy
 - AGD – Telewizory, Lodówki, Pralki, Odkurzacze, Mikrofalówki
 - Urządzenia specjalistyczne – echosondy, oscyloskopy, urządzenia pomiarowe, itp..

Systemy czasu rzeczywistego

- Specyficzny rodzaj firmware, systemy wbudowane (ang. embedded)
- Stosowane tam, gdzie istnieją surowe wymagania na czas wykonania operacji lub przepływu danych
- Rygorystyczne systemy czasu rzeczywistego
 - Minimalizacja czasu operacji,
 - Wykluczenie konkurencyjnych programów i podział czasu
- Zastosowania
 - systemy nadzorowania eksperymentu, badań medycznych
 - sterowanie procesami w przemyśle
 - sterowniki wtrysku w samochodzie,
 - programator pralki,
 - sterowniki w raketach
 - itp.



Jaki system operacyjny ?

- Bezpieczeństwo
 - Stabilność
 - Łatwość obsługi
 - Dostępność oprogramowania
 - Cena
-
- Dla kogo ?
 - Do czego ?



Modele udostępniania OS

- Oprogramowanie własnościowe (nie tylko płatne)
 - Komercyjne/płatne
 - Darmowe
 - Freeware
 - Adware
 - Trial
- Oprogramowanie open source
 - Różne typy licencji
 - GNU
 - GPL
 - Apache
 - BSD
 - Wiele innych



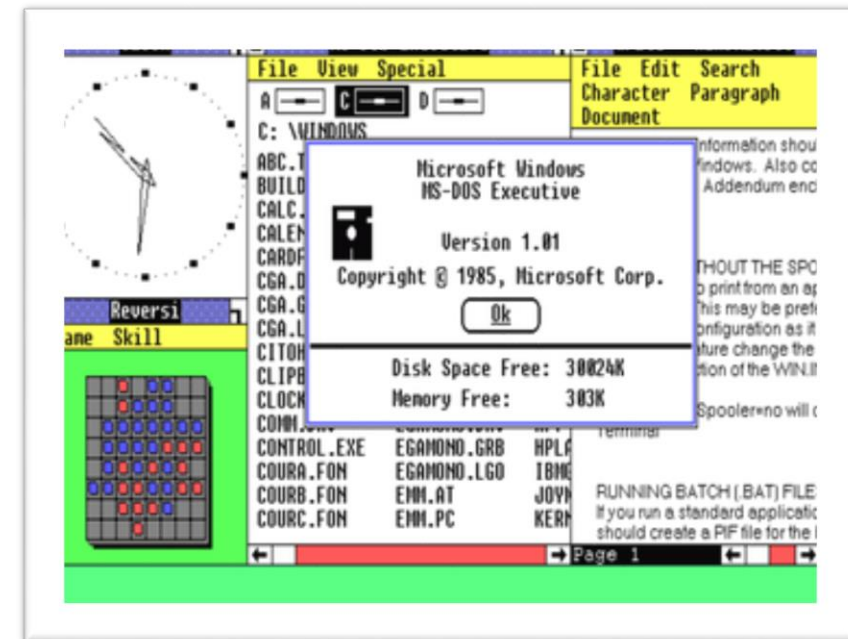
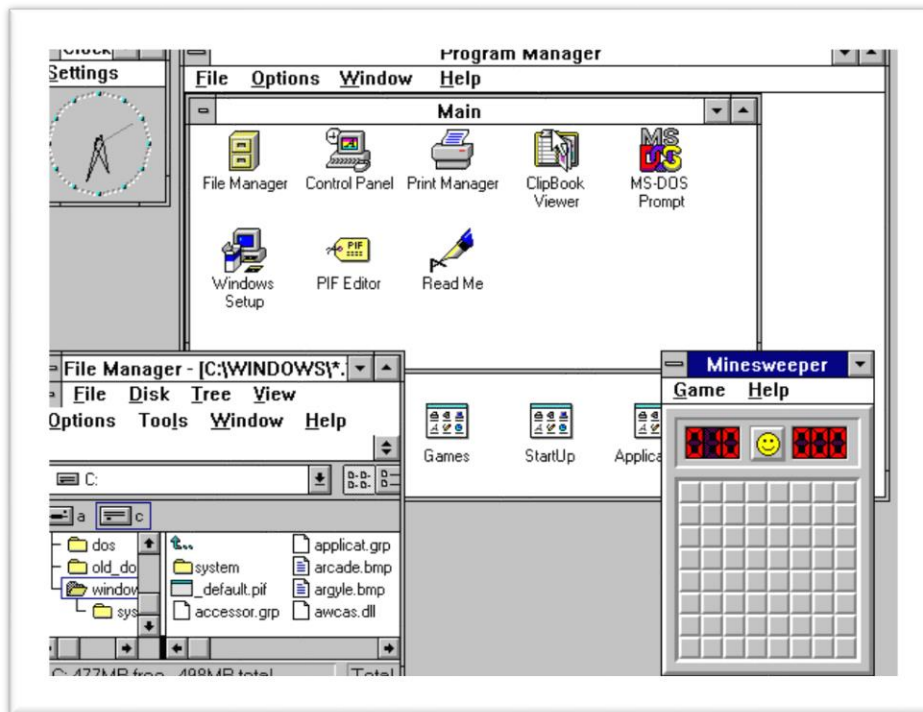
Systemy operacyjne

- Rodzina Microsoft
- Rodzina Apple
- Unixy
- Linuxy
- Android
- ChromeOS



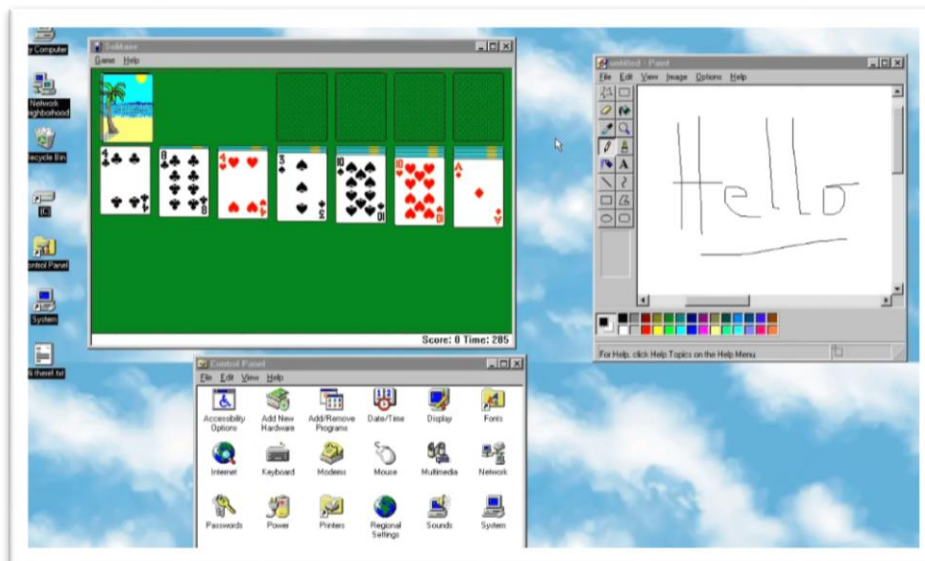
Systemy operacyjne Microsoftu

- Działają na serwerach (Windows Server), systemach wbudowanych (Windows CE), i desktopach
- 1984r - Pierwsza wersja – nakładka na MS DOS,
- 1985r – Windows 1.0
- 1992r - Windows 3.1



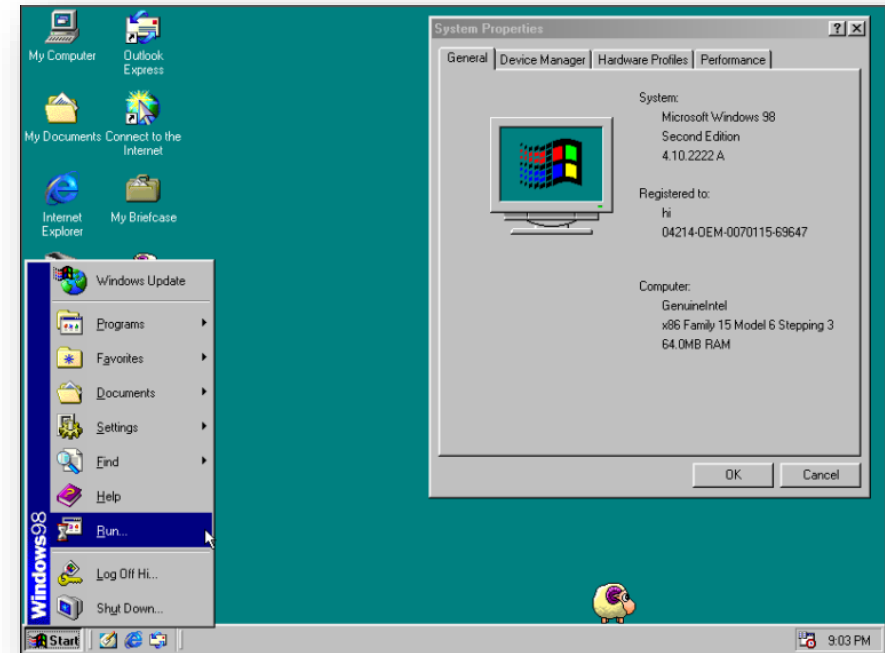
Systemy operacyjne Microsoftu

- 1993r – Windows NT 3.1
 - Pierwszy 32bitowy OS MS
- 1995 – Windows 95
 - Długie nazwy
 - Przycisk start
 - Pasek zadań
 - Ikona „Mój komputer”



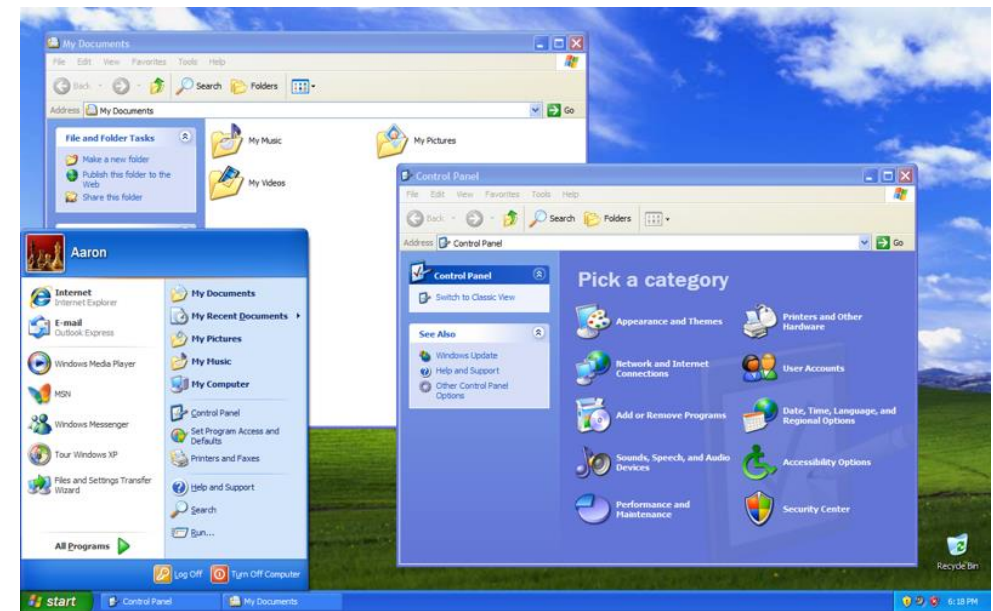
Systemy operacyjne Microsoftu

- 1996 – Windows CE
 - System dla urządzeń przenośnych lub system embedded
 - Jądro – 1MB pamięci / ROM
- 1998 – Windows 98
 - Obsługa USB, FAT32
 - Usługa Windows Update
- 1999 – Windows SE (Second Edition)
 - Nowy Internet Explorer
 - Obsługa napędów DVD
 - Wielu użytkowników systemu
- 2000 – Windows ME (Millenium)
 - Dodany pakier Windows Media (Player, Movie Maker)
- Wysoka niestabilność, słaby odbiór



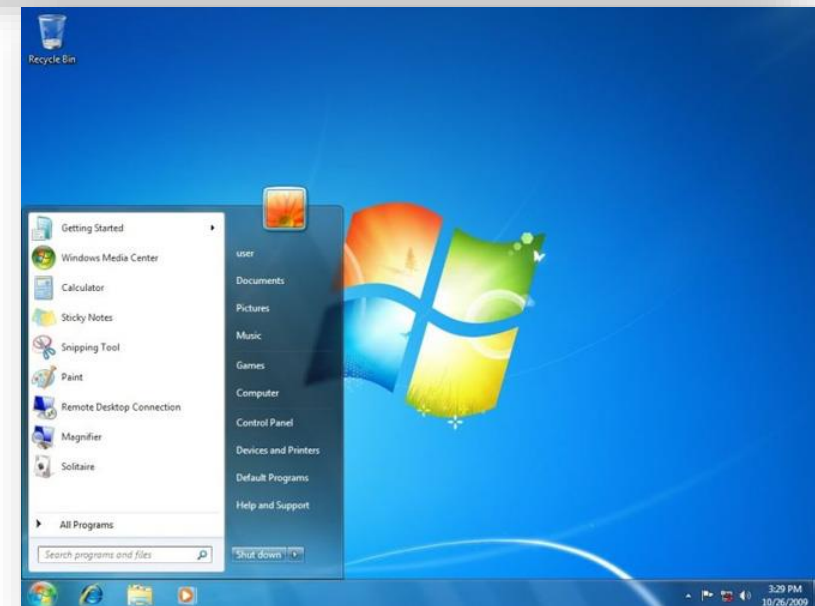
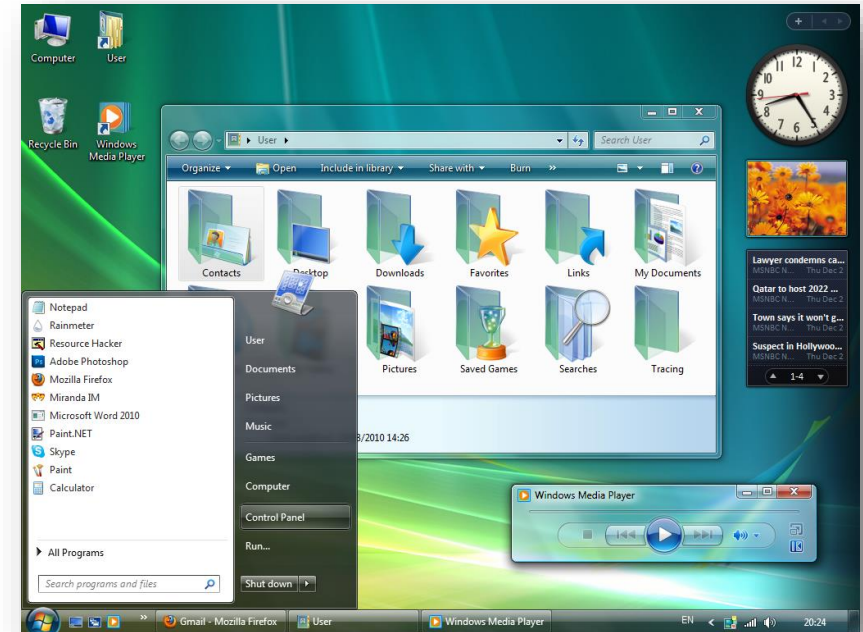
Systemy operacyjne Microsoftu

- 2000r – Windows 2000
 - Windows 2000 Professional – wersja na komputery osobiste
 - Windows 2000 Server – wersja na serwery
 - Obsługa NTFS 3.0 i Active Directory
- 2001 – Windows XP
 - nowy interfejs graficzny Luna
 - Zintegrowana zapora sieciowa
 - Antypiracja aktywacja systemu
- 2003 – Windows Server 2003
 - Serwerowa wersja Windowsa XP
 - IIS v6.0
 - .NET v.1.0



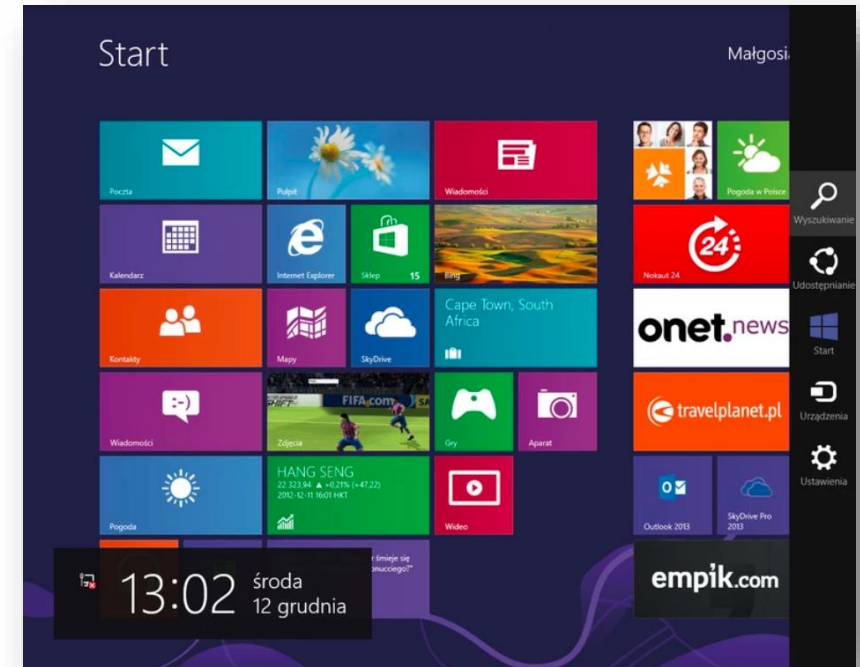
Systemy operacyjne Microsoftu

- 2007r – Windows Vista
 - Interfejs graficzny Aero
 - Kontrola użytkownika
 - Nagrywanie płyt DVD
 - Słabe recenzje systemu – błędy, wysokie wymagania sprzętowe
- 2009r – Windows 7
 - Pozbawiony odtwarzacza WMPlayer
 - W ciągu roku liczba instalacji przekroczyła całkowitą liczbę instalacji Visty
 - Biblioteki – wirtualne foldery
 - Poprawiona stabilność systemu



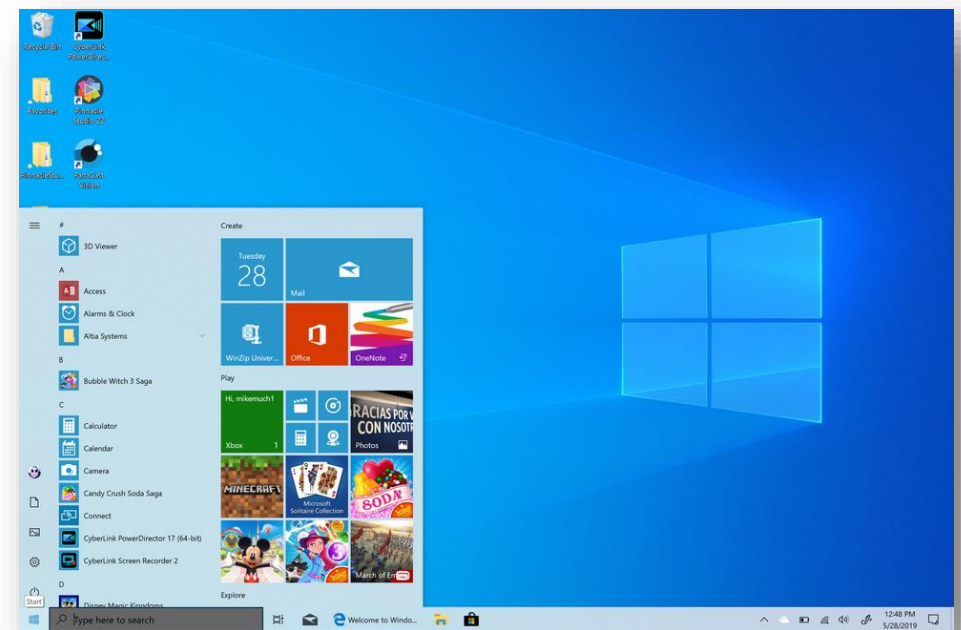
Systemy operacyjne Microsoftu

- 2012r – Windows 8
 - Interfejs ModernUI
 - Wprowadzenie Windows Store
 - Ekran Start + Kafelki (krytyka użytkowników desktopowych)
 - Obsługa NFC
 - Wersja RT dla architektury ARM
- 2013r – Windows 8.1
 - Darmowa aktualizacja systemu Windows 8
 - Udoskonalona wyszukiwarka
 - Domyślnie zainstalowany Skype
 - Personalizacja ekranu startowego
 - Powrót paska start i menu kontekstowego



Systemy operacyjne Microsoftu

- 2015r – Windows 10
 - Dostosowanie interfejsu do urządzeń bez ekranu dotykowego
 - Przywrócenie menu Start
 - Dostępny dla laptopów, dekstopów, tabletów, smartfonów, Xbox One
 - Tryb S – wyższa ochrona, instalacja jedynie aplikacji z Windows Store
 - Nowa przeglądarka – Microsoft Edge



Systemy operacyjne Apple

- Rodziny systemów
 - macOS – systemy dla komputerów Macintosh (wcześniej OS X lub Max OS X)
 - iOS – systemy dla urządzeń przenośnych (iPhone, iPod Touch, iPad)
 - tvOS – system do odtwarzania materiałów video
 - watchOS – system operacyjny smartwatchy Apple



- System dla komputerów Macintosh
 - oparty o jądro UNIXa
 - Pierwsza wersja 1991r (serwer), 2001r (desktop)
 - Do 2006 Macintoshe jedynie z procesorami PowerPC
 - 2006r – Mac OS X 10.4 – dla Mac’ów w architekturze x86
 - 2008r – Mac OS X 10.6 – ostatnia wersja wspierająca PowerPC
 - Dedykowany dla: iBook, PowerBook, Power Mac, MacBook, Mac Pro, iMac



- System przeznaczony dla urządzeń mobilnych
 - Obecna nazwa od 2010 roku, wcześniej jako iPhoneOS
 - Bazuje na systemie Mac OS X 10.5
 - Oprogramowanie dostępne jedynie z repozytorium AppStore (brak oprogramowania OpenSource, wysokie bezpieczeństwo)
 - Dedykowany dla: iPhone, iPod Touch, iPad



Apple - tvOS

- System przeznaczony dla urządzenia Apple TV digital media player
 - Apple TV – odtwarzacz multimedialny
 - odbiera dane z aplikacji iTunes lub urządzeń z macOS lub iOS
 - transmituje na ekran monitora z rozdzielczością 4K
 - tvOS
 - Wprowadzony w 2015r
 - Oparty na iOS



Apple – WatchOS

- System przeznaczony dla smartzegarka Apple Watch
 - Pierwsza wersja - kwiecień 2015 rok
 - Szósta wersja – sierpień 2019 rok
 - Rozpoznawanie gestów i siły dotyku
 - Oferuje HealthKit – środowisko fitness



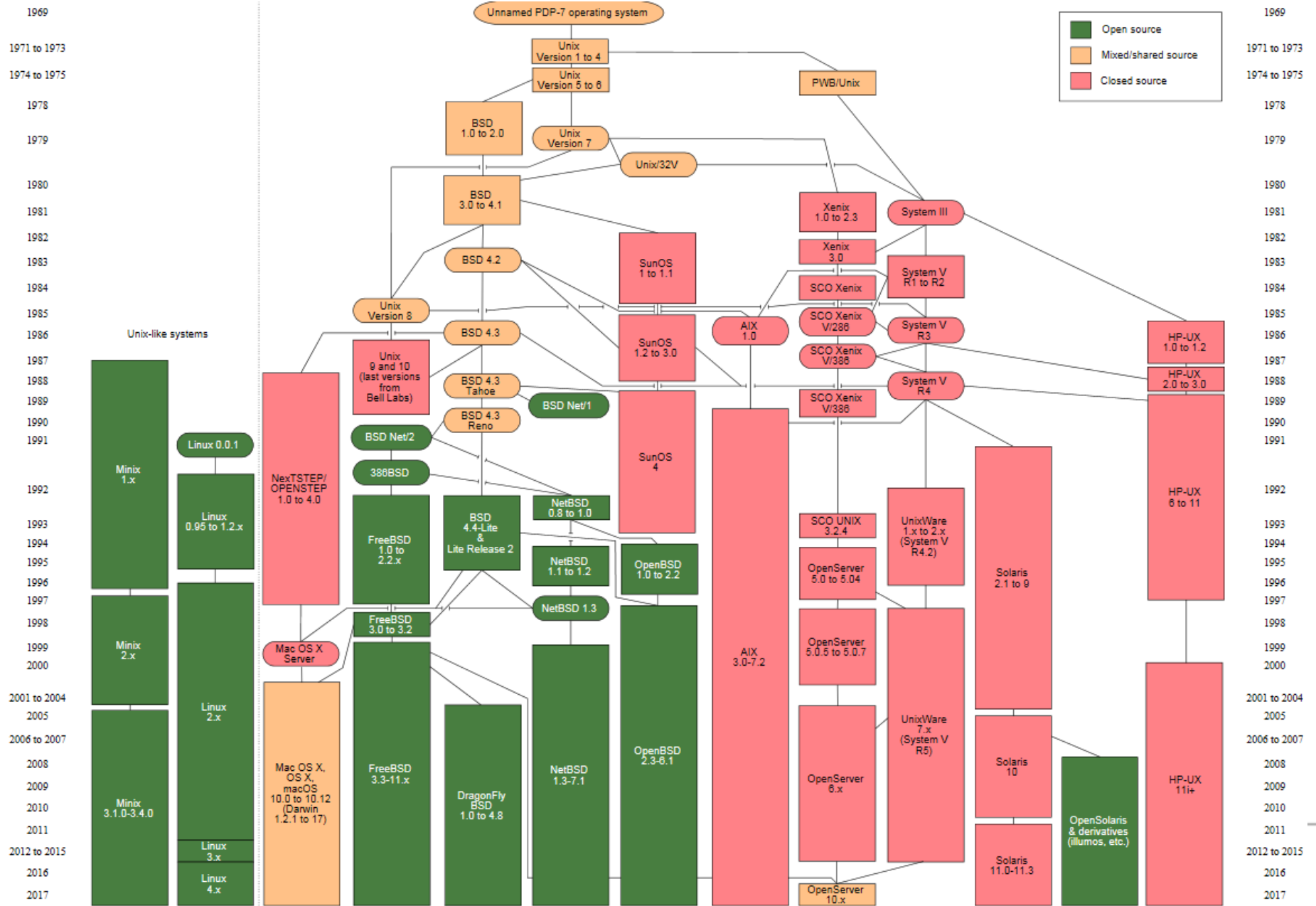
Systemy UNIX

- UNIX – system operacyjny
 - Rozwijany od 1969 roku w Bell Labs (Unix System Laboratories)
 - Ogromna popularność w latach 70tych – ewolucja wielu odmian i implementacji
 - Główne cechy
 - Hierarchiczny system plików
 - Składniki systemu jako pliki (także urządzenia)
 - UNIX obecnie
 - Nazwa zastrzeżona dla The Open Group
 - Synonim grupy systemów
 - Wywodzących się z pierwotnego pnia UNIXa
 - Zgodnych z normami POSIX lub Single UNIX Specification
 - Skomplikowana sytuacja prawna (prawa do nazwy, kodu źródłowego, praw autorskich, patentów)

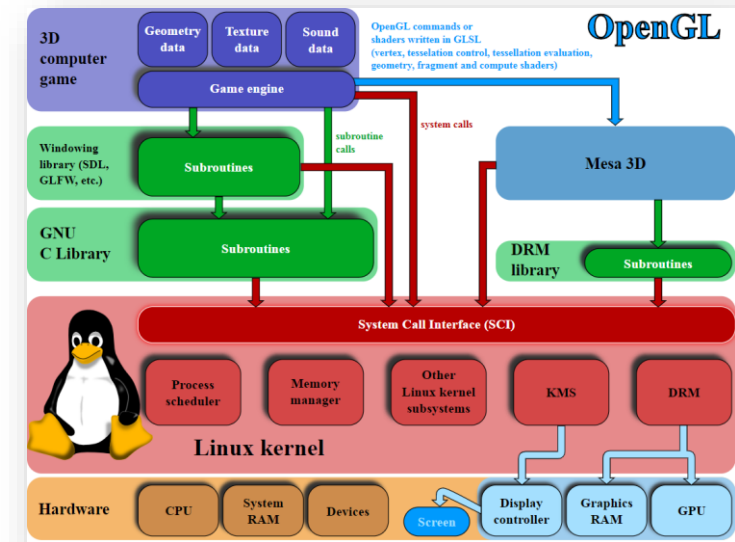




Systemy UNIX - historia



- Rodzina unixo-podobnych OS opartych na jądrze Linux
- „jądro” (def)
 - niskopoziomowe oprogramowanie systemowe, którego funkcją jest podstawowa obsługa sprzętu, dysków i systemu plików, wielozadaniowości, równoważenia obciążenia, sieci oraz zabezpieczeń.
 - jądro nie jest kompletnym systemem operacyjnym (w typowym rozumieniu tego pojęcia).
- Jądro Linux (ang. Linux Kernel)
 - Jedna z najważniejszych, WOLNA część wielu unixopodobnych systemów operacyjnych
 - Kod źródłowy objęty licencją GNU GPL – oprogramowanie wolne i otwarte
 - Początkowo opracowane przez Linusa Torvaldsa (1991r), aktualnie rozwijane przez The Linux Foundation
 - Napisane w języku C z elementami Asemblera
 - Obsługuje: wielozadaniowość, wielowątkowość, pamięć wirtualną, obsługę TCP/IP, i wiele innych elementów standardu POSIX i ANSI
 - W październiku 2008 liczba linii kodu jądra przekroczyła 10mln

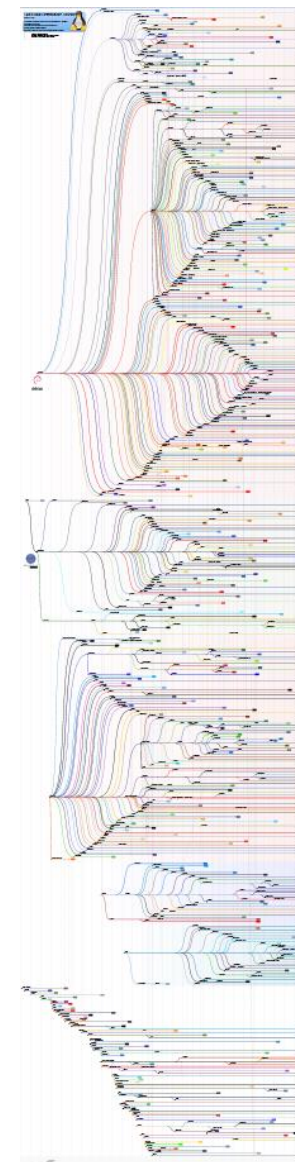
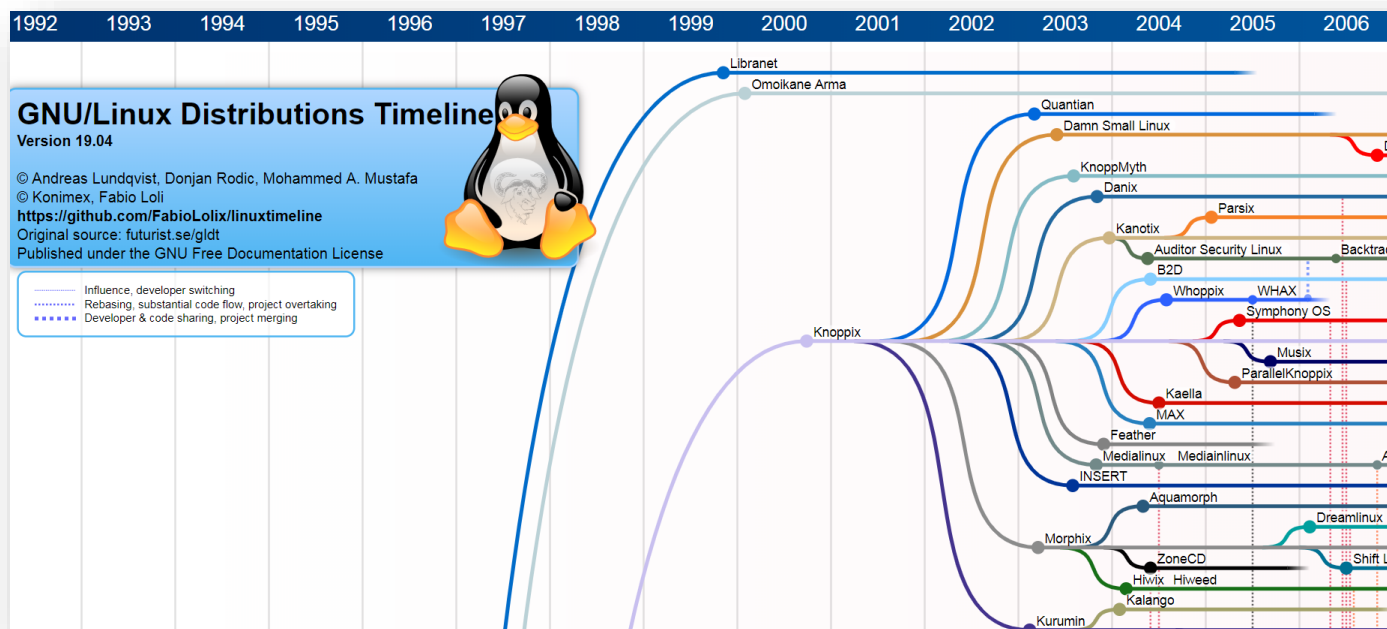


Systemy Linux

- Dystrybucja Linuxa - pełny system czyli zbiór programów (dobranych do konkretnych wymagań) zbudowanych z użyciem jądra Linux
- Wspiera kilkadziesiąt architektur sprzętowych różnych producentów (ARM, HP, IBM, Intel, Motorola, PowerPC, Sparc)
- Cecha wspólna dystrybucji Linuxa – kod źródłowy może być dowolnie wykorzystywany, modyfikowany i rozpowszechniany
- Zastosowania:
 - Środowiska serwerowe (IBM, Oracle, Dell, MS, HP, RedHat, Novell)
 - Superkomputery
 - Desktopy/laptopy
 - Systemy wbudowane – telefony, routery, telewizory



Dystrybucje systemu Linux



Popular Linux Distributions

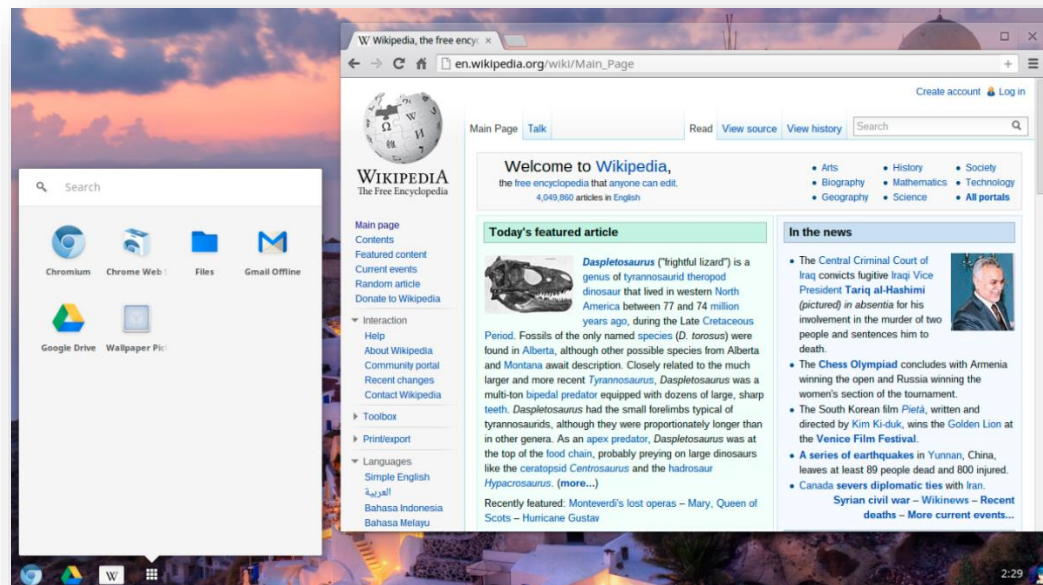


- System operacyjny z jądrem Linux dla urządzeń mobilnych (telefony, smartfony, tablety, netbooki)
 - Początkowo rozwijany przez firmę Android Inc.
 - Zakupiony przez Google (2005r)
 - Aktualnie (od 2007r) rozwijany przez sojusz Open Handset Alliance (sojusz biznesowy 78 firm, m.in.: Google, HTC, Dell, Motorola, Samsung, LG, T-Mobile, Nvidia)
 - Aplikacje dostępne przez GooglePlay (wcześniej Android Market)
 - Aktualna wersja 10 (Android Q) – sierpień 2019



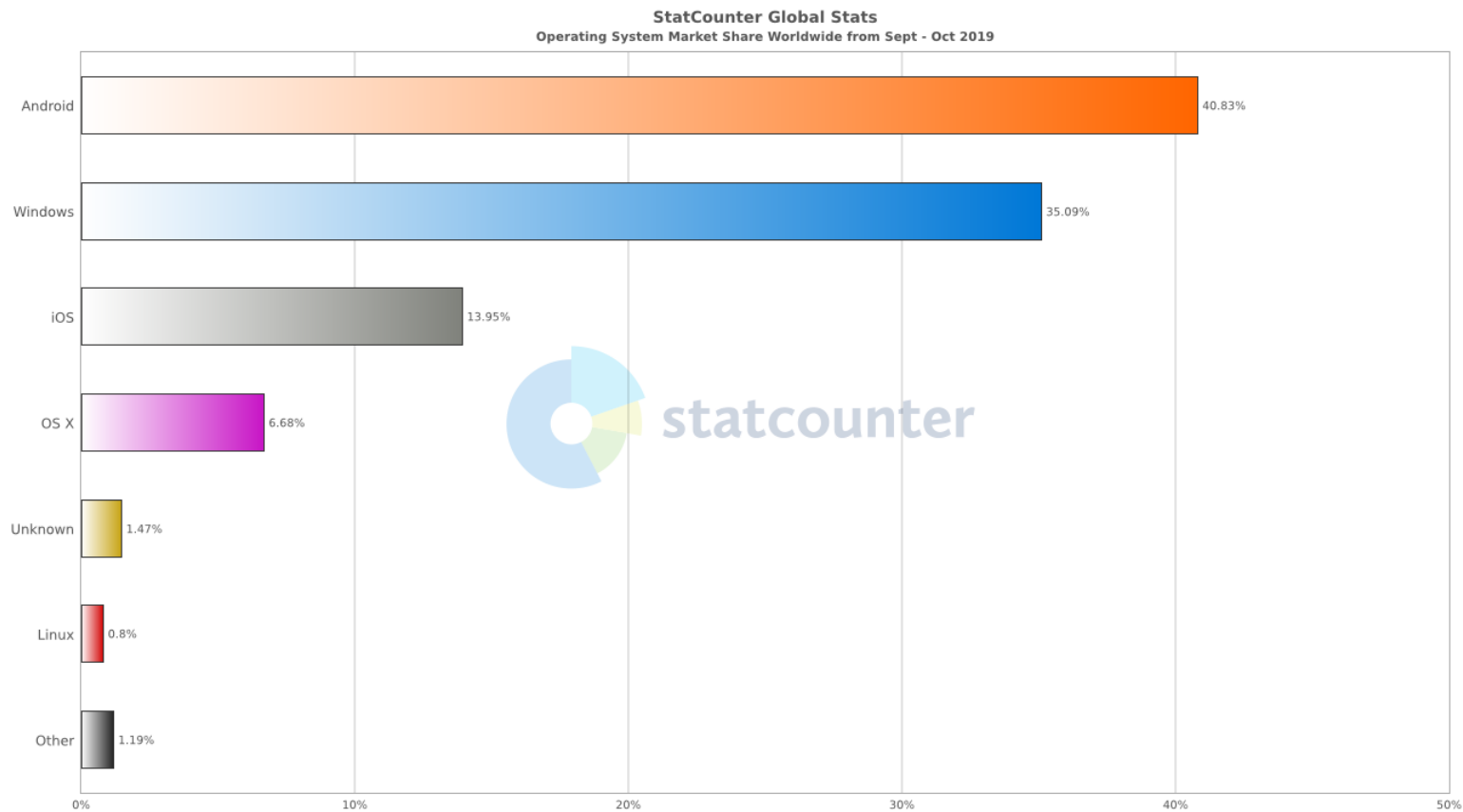
ChromeOS / Chromium OS

- Dystrybucje Linuxa tworzone przez Google
 - Chromium OS – otwarty i darmowy (open-source) system operacyjny
 - Chrome OS – wersja Chromium OS instalowana jako OEM na tabletach ChromeBook
- Wykorzystują Google Chrome jako interfejs użytkownika – dedykowane raczej do zastosowań webowych
- Kod źródłowy opublikowany w 2009 roku



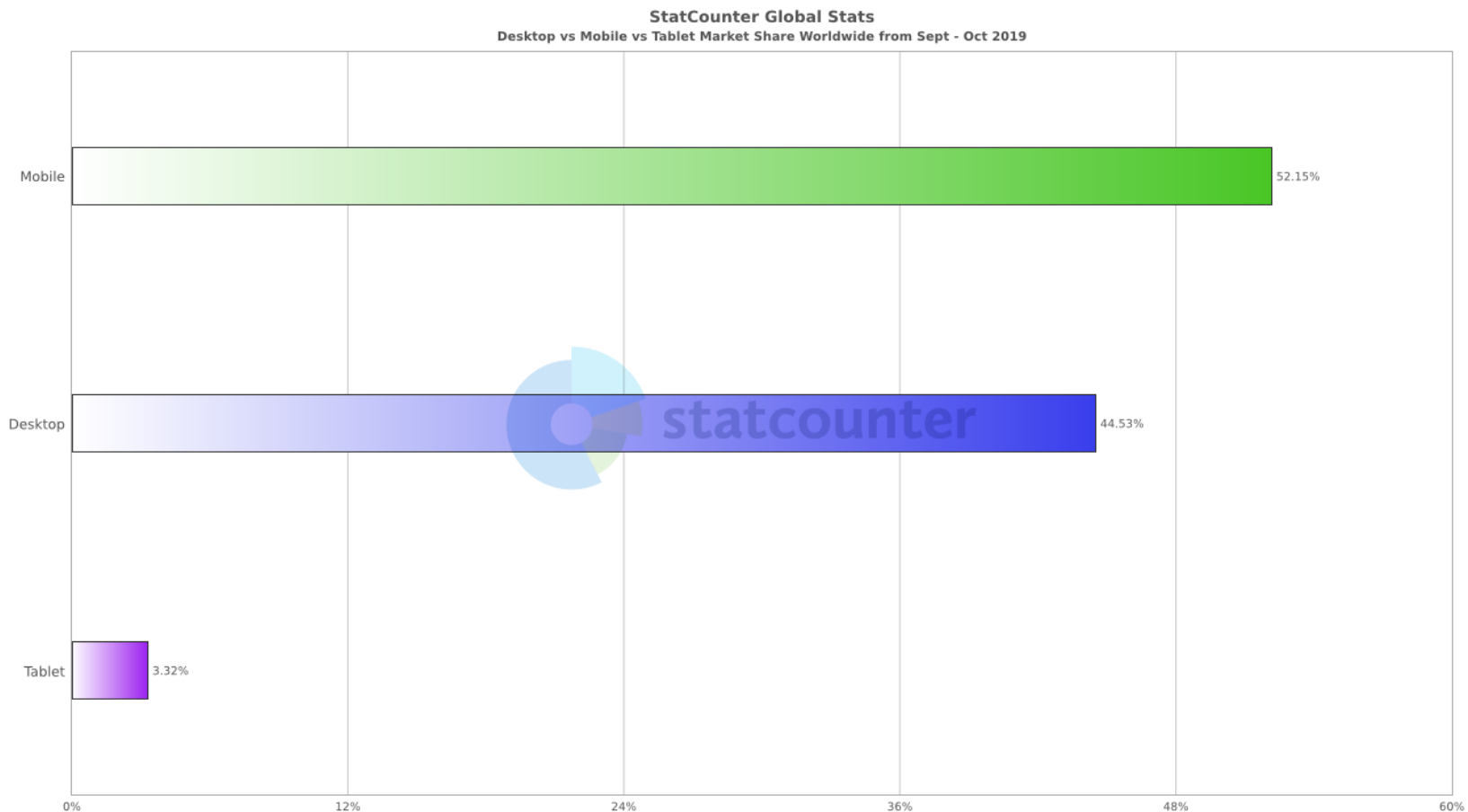
Rynek systemów operacyjnych

- Źródło: <https://gs.statcounter.com/>
- Dane z września 2019
- Porównanie globalne (wszystkie urządzenia)



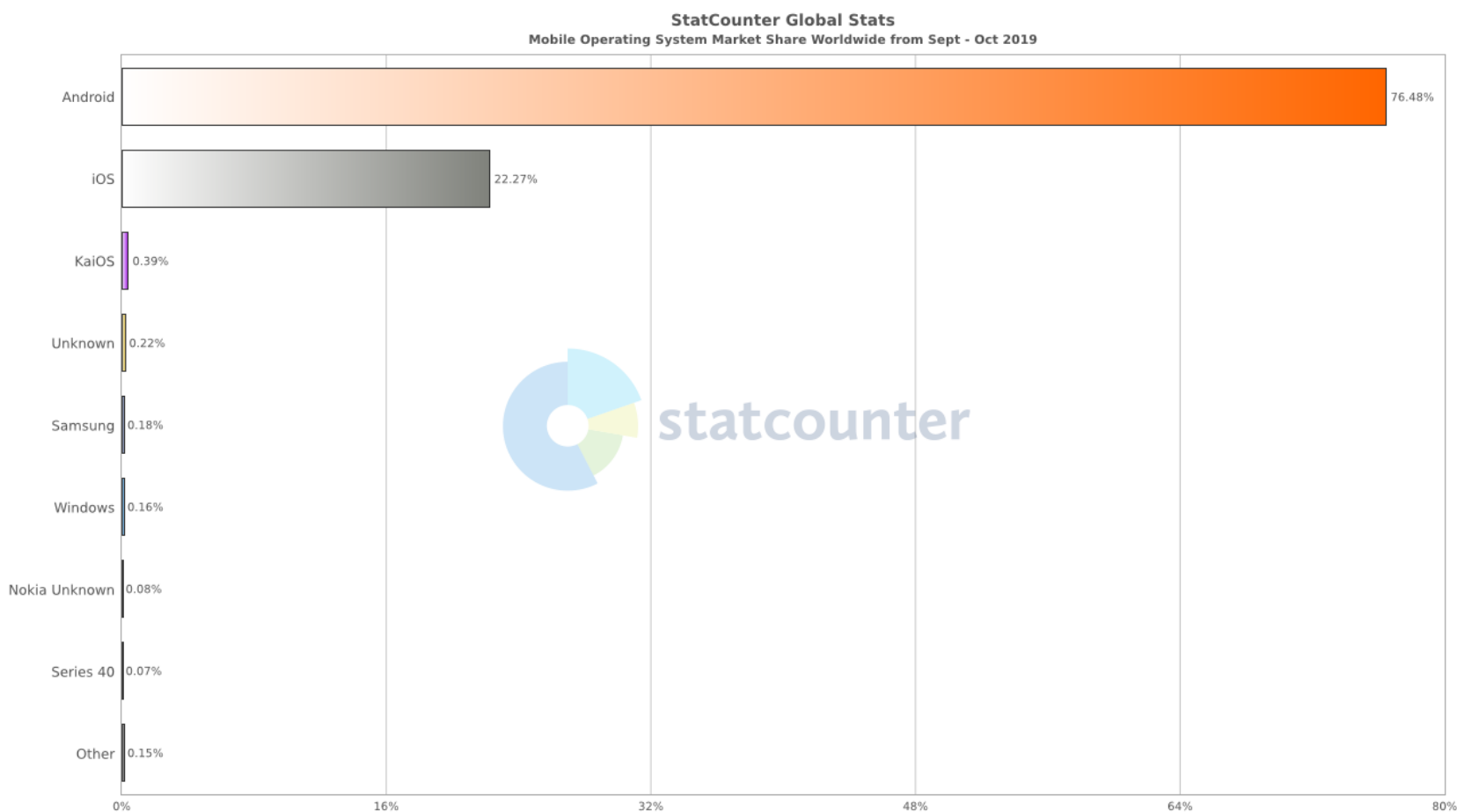
Rynek systemów operacyjnych

- Podział rynku między typy urządzeń (wrzesień-październik 2019)



Rynek systemów operacyjnych

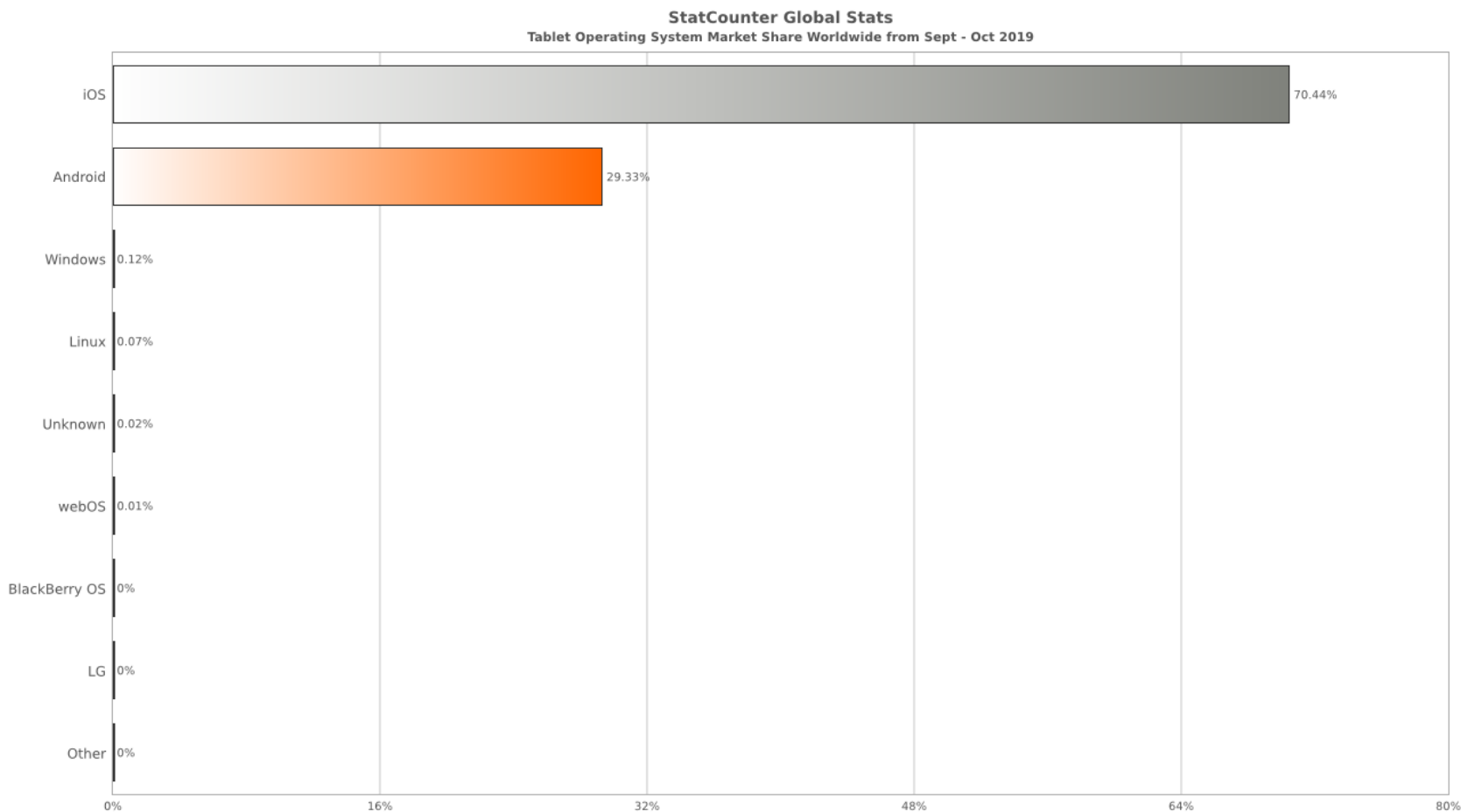
- Telefony komórkowe (wrzesień-październik 2019)





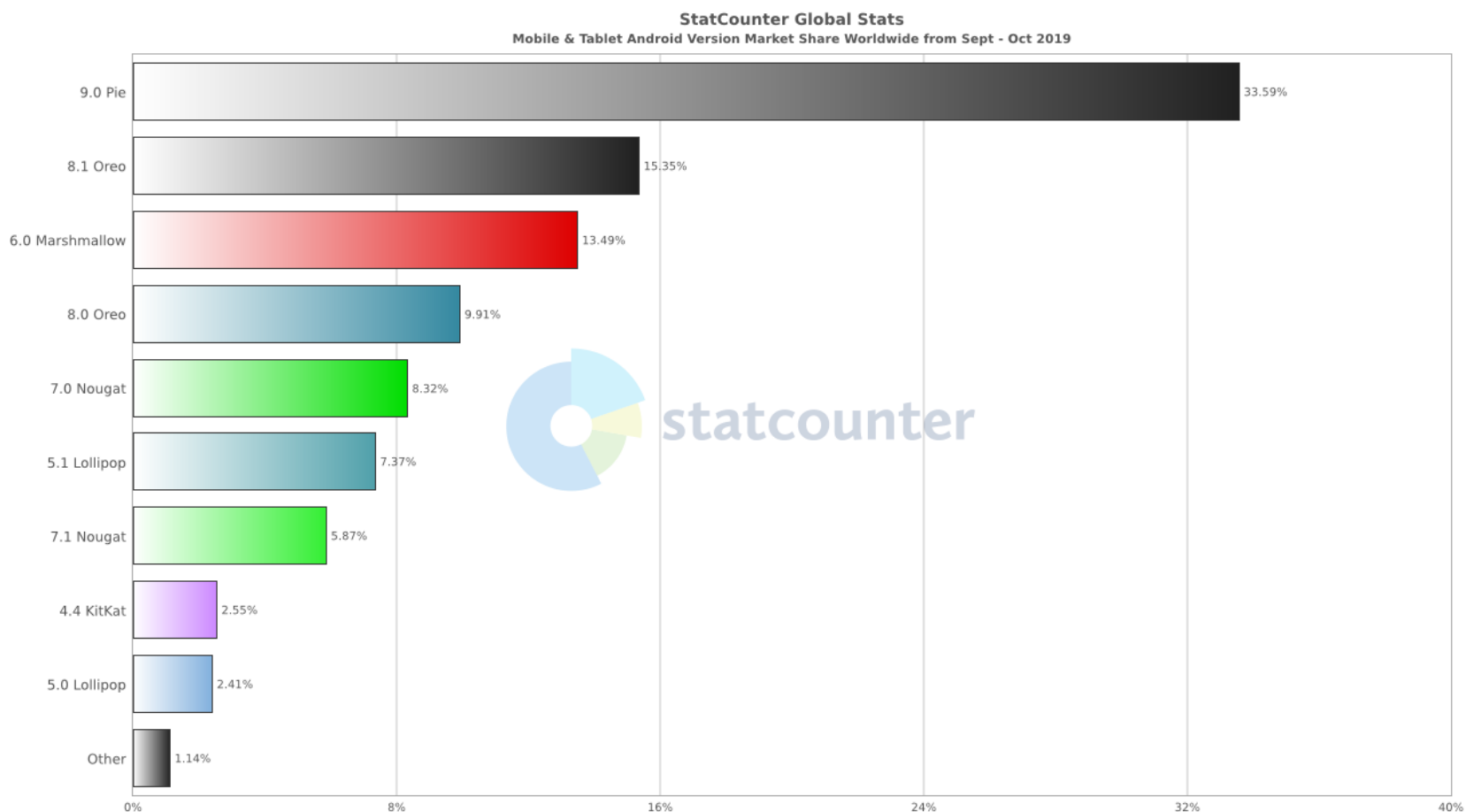
Rynek systemów operacyjnych

- Tablety (wrzesień-październik 2019)



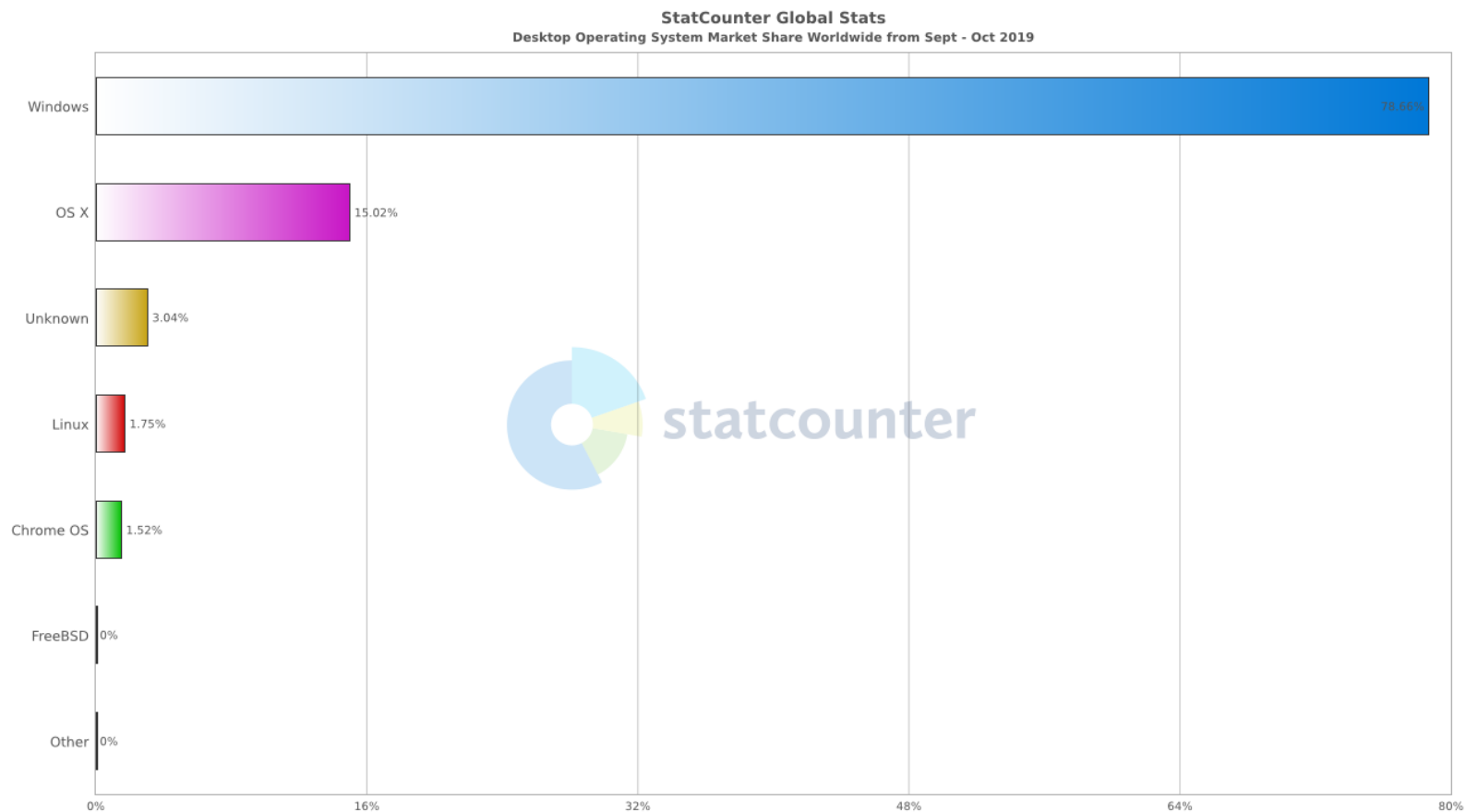
Rynek systemów operacyjnych

- Telefony+tablety – wersje Androida (wrzesień-październik 2019)



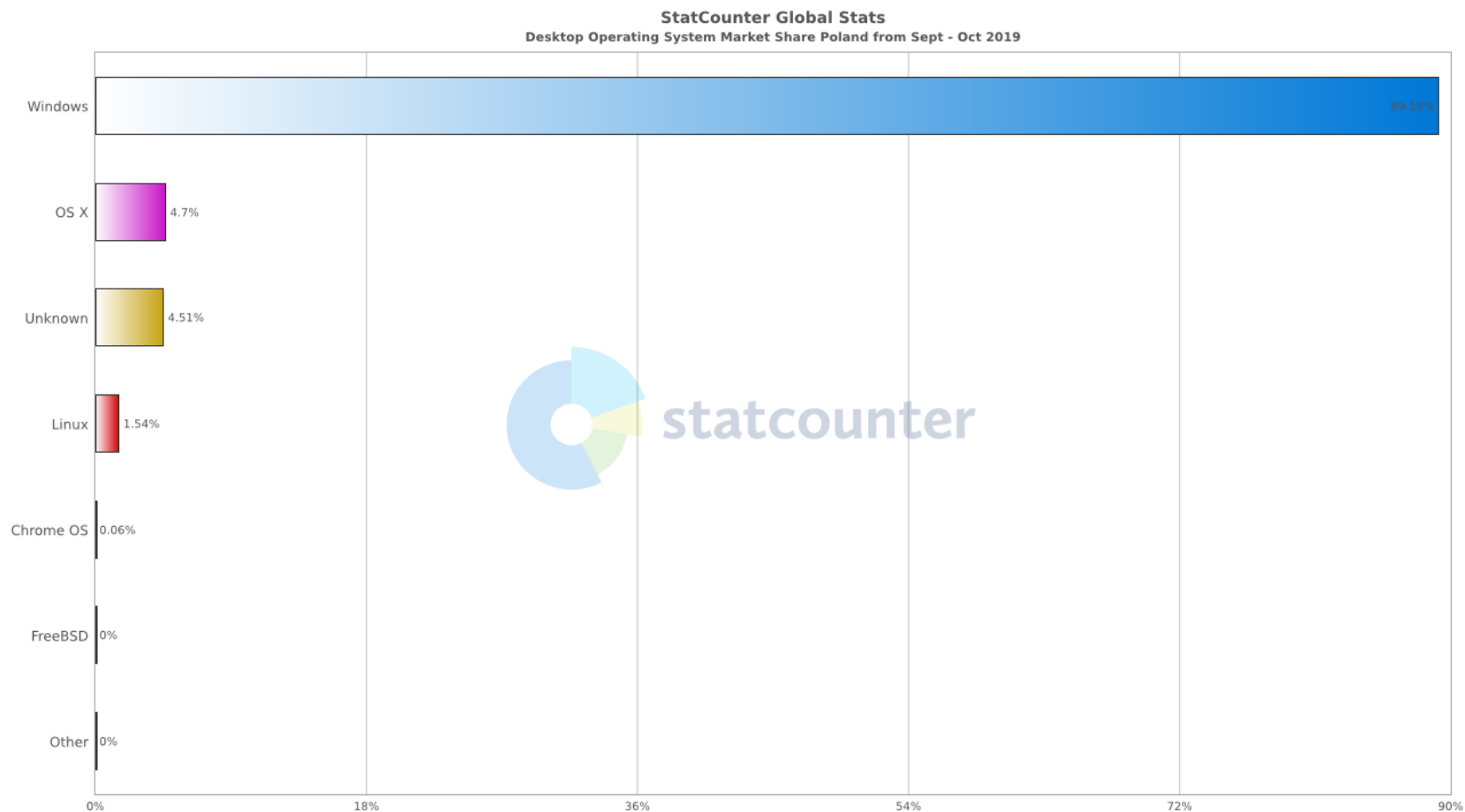
Rynek systemów operacyjnych

- Desktopy (wrzesień-październik 2019)



Rynek systemów operacyjnych

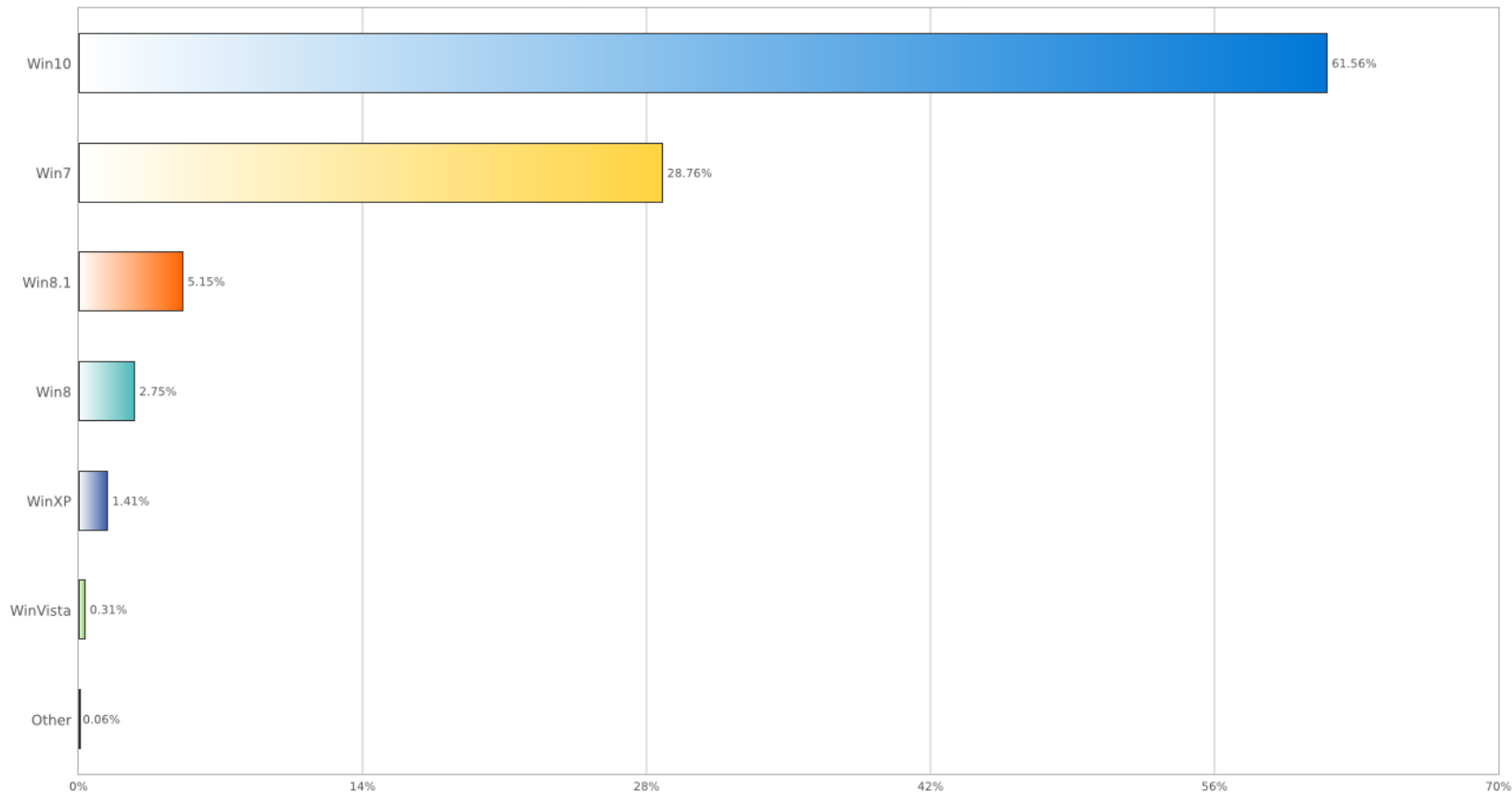
- Desktopy – Polska (wrzesień-październik 2019)



Rynek systemów operacyjnych

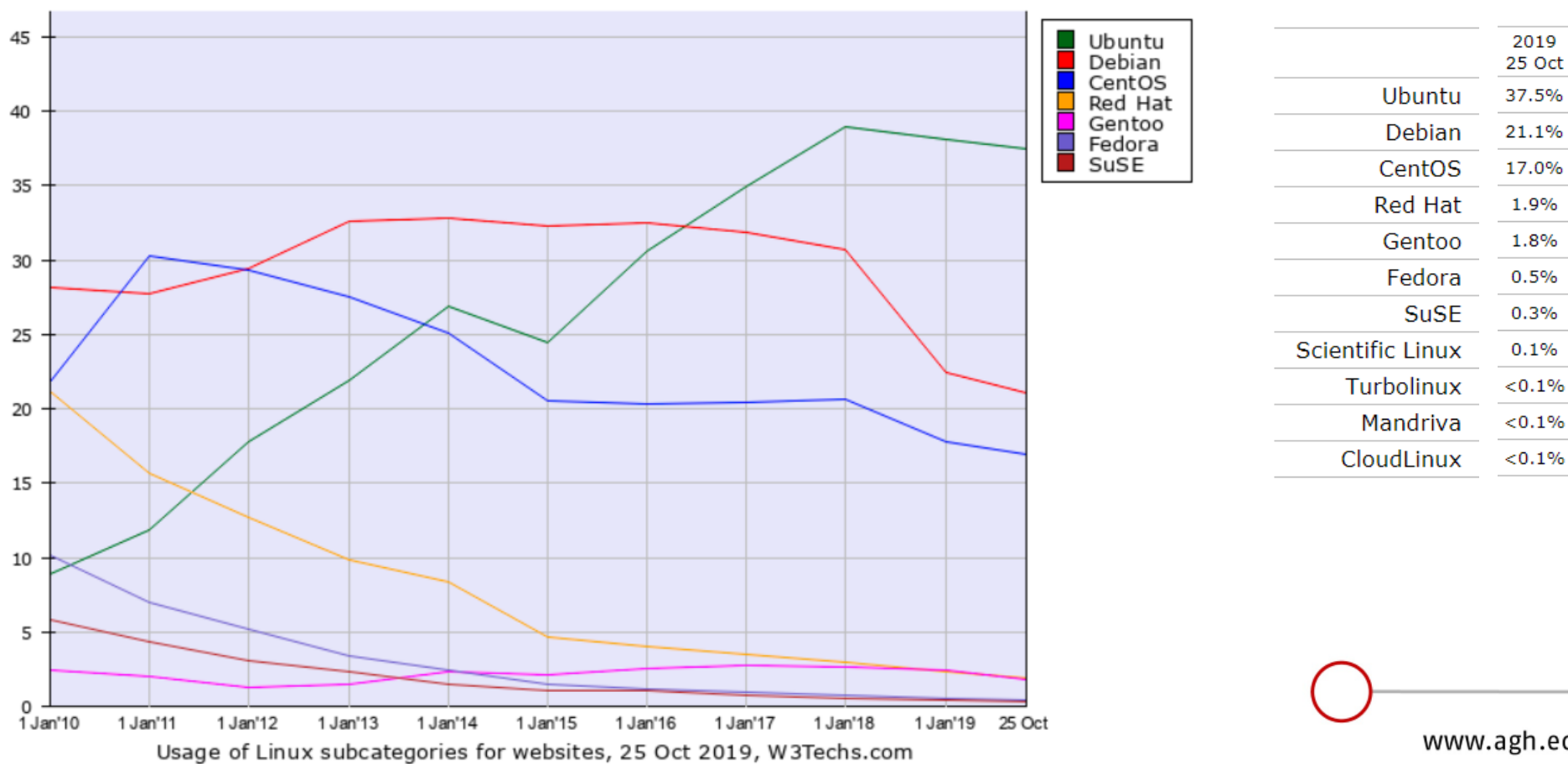
- Desktopy – wersje systemów z rodziny Microsoft

StatCounter Global Stats
Desktop Windows Version Market Share Worldwide from Sept - Oct 2019



Rynek systemów operacyjnych

- Desktopy – wersje systemów z rodziny Linux (październik 2019)
- Źródło w3techs.com



Literatura, bibliografia

- Silberschatz A., Galvin P.B.: Podstawy systemów operacyjnych, WNT 2006.
- Nemeth E., Snyder G., Hein T., Whaley B.: Unix i Linux. Przewodnik administratora systemów, Helion 2011
- Stevens R.W.: Programowanie w środowisku systemu UNIX, WNT 2014.
- Flynt C.: Skrypty powłoki systemu Linux – receptury. Helion 2018
- Sosna Ł.: Linux Komendy i polecenia, Helion 2014