

Systemy operacyjne na urządzenia mobilne

P. Pasteczka K. Stefański Ł. Zubkowicz

16 stycznia 2009

Oczekiwania użytkownika

- Stabilność systemu,
- Szybkość działania,
- Możliwość obsługi wszystkich funkcji urządzenia,
- Wysokopoziomowy dostęp do funkcji systemu operacyjnego,
- Małe użycie zasobów systemowych,
- Kompatybilność aplikacji mobilnych z ich odpowiednikami na standardowym komputerze

Oczekiwania programisty

- Abstrakcyjna warstwa do obsługi funkcji telefonu

Ograniczenia techniczne

- ilość pamięci operacyjnej,
- ilość pamięci wewnętrznej,
- żywotność baterii,
- chłodzenie procesora

Krótką historia

- przed 1998 rokiem: firma PSION i jej system EPOC
- 24 czerwca 1998: utworzenie Symbian Ltd. na podstawie umowy o partnerstwie między firmami PSION, Nokia, Ericsson i Motorola
- 24 czerwca 2008: przejęcie firmy Symbian Ltd. przez firmę Nokia i ustanowienie Symbian Foundation

Kolejne odsłony systemu

- EPOC16, EPOC32, EPOC Release 4, EPOC Release 5
- Symbian OS v6.0, v6.1
- Symbian OS v7.0, v7.0s
- Symbian OS v8.0, v8.1
- Symbian OS v9.0, v9.1, v9.2, v.9.3, v9.4, v.9.5

Rynek smartfonów w listopadzie 2008 roku

- Symbian Ltd.: 46.6%
- Apple: 17.3%
- RIM: 15.3%
- Microsoft: 13.5%
- inni: 6.3%

Wspierane procesory: tylko rodzina ARM

- RISC
- 32 bit
- little-endian
- wspierają tryb chroniony
- energooszczędne

Wielozadaniowy system czasu rzeczywistego

- “wynik nie dostarczony na czas to zły wynik”
- gwarancja bezwzględnych ram czasowych wykonywanych czynności
- szeregowanie procesów na podstawie ich priorytetów

Stworzony w oparciu o mikrojądro

- mikrojądro: zarządzanie procesami i pamięcią, niezbędne funkcje systemowe
- sterowniki urządzeń: pisane jako rozszerzenia (DLL) do mikrojądra
- reszta: oparta o architekturę klient-serwer i wymianę komunikatów

Interfejs

- w pełni niezależny od systemu operacyjnego
- niedostarczany razem z systemem (tworzony przez producentów urządzeń)
- najpopularniejsze to S60, UIQ, MOAP, dawniej S80

Podstawowe serwery usług

- telefonia (ETEL)
- usługi komunikacyjno-sieciowe (ESOCK, C32)
- system plików
- usługi interfejsu użytkownika (UIKON)

Możliwości systemu

- GPRS, CDMA, 3G, HSDPA (UMTS), 802.11 (wifi)
- IPv6
- Bluetooth, IRDA, USB
- OpenGL ES, DVB-H, ISDB-T
- multimedia

Najnowsze możliwości systemu

- WiMax i inne szerokopasmowe technologie
- wbudowany SQLite
- wspieranie usług lokalizacyjnych (GPS)
- wieloprocessorowość na żądanie (SMP)
- wczytywanie kodu na żądanie

Instalacja oprogramowania

- aplikacje wymagają cyfrowego podpisania przed instalacją
- możliwość posiadania wirtualnej maszyny Java ME
- istnieje wiele innych interpreterów/wirtualnych maszyn (np. do języka Python)
- przykładowe programy: Mobile Web Server (port Apache), Nokia Internet Radio (SHOUTcast), PuTTY, SymTorrent

W skrócie

- Symbian OS Security Architecture, wymaga świadomego pisania programów
- programy antywirusowe

Symbian OS Security Architecture

- prewencja: instalowanie tylko podpisanych aplikacji, ograniczanie im uprawnień (firewall protection), istnienie pewnej przestrzeni systemu plików tylko dla systemu (data caging)
- detekcja: logowanie błędów, wsparcie dla programów antywirusowych
- reakcja: najczęściej polega na blokowaniu tego co nie działa w oczekiwaniu na aktualizację

Podstawowe informacje

- API systemu jest publicznie udokumentowane
- natywnym językiem jest zmodyfikowany C++
- narzędzia w ramach SDK skupione są głównie wokół UIQ i S60
- producenci telefonów także często dostarczają swoje własne narzędzia
- inne języki w których można coś zdziałać to OPL, Python, Visual Basic, Simkin, Perl, Java

Porównanie z typowymi praktykami

- konieczność używania “deskryptorów” i “stosu czyszczącego”
- skomplikowane zarządzanie pamięcią
- wiele niskopoziomowych szczegółów
- mocno specjalizowane modyfikacje do C++

Tworzenie oprogramowania

- Carbide.c++, IDE od firmy Nokia oparte na Eclipse
- Microsoft Visual Studio 2003, 2005 (z pluginem Carbide.vs)
- NS Basic/Symbian OS
- redFIVElabs Net60
- Borland IDE for Symbian OS
- Apple Xcode (z dodatkowym pluginem)

Wdrażanie oprogramowania

- kompilacja (np. cross-compiling poprzez GCC)
- paczkowanie do plików SIS
- podpisanie paczek w standardzie Symbian Signed

Zalety

- najpopularniejszy system, przez co istnieje sporo oprogramowania
- prężnie i od dawna rozwijany, dzięki temu nowoczesny i stale nabywający nowe możliwości
- mocno dostosowany do ograniczonych warunków sprzętowych
- udostępniający wiele możliwości pisania oprogramowania

Wady

- brak standardowego interfejsu, przez co trzeba utrzymywać różne wersje aplikacji
- wymóg posiadania certyfikatów, które sporo kosztują i długo się na nie czeka
- skomplikowane zwyczaje przy pisaniu oprogramowania, poza tym znajomość standardowego C++ nie wystarcza

Krótką historia

- czerwiec 2005: firma Google przejęła małą firmę Android Inc.
- grudzień 2006: nasiliły się spekulacje o wejściu firmy Google na rynek telefonów komórkowych
- wrzesień 2007: złożenie przez firmę Google kilku patentów w obszarze telefonii mobilnej
- 5 listopada 2007: założenie konsorcjum Open Handset Alliance i ujawnienie systemu Android
- 21 października 2008: uwolnienie źródeł systemu Android
- 9 grudnia 2008: powiększenie konsorcjum o 14 nowych członków (obecnie 48)

Ideologia systemu

- otwarty kod (większość na licencji Apache 2.0, reszta GPL2)
- możliwość łatwego zmieniania i dostosowywania systemu
- ułatwienie w tworzeniu aplikacji na urządzeniach mobilnych

Pierwszy telefon



HTC T-Mobile G1

System na bazie linuxa

- obsługa teoretycznie wszystkich procesorów (w praktyce tylko ARM)
- wieloprogramowość odziedziczona za darmo po linuxie
- linux to nie system czasu rzeczywistego
- jądro monolityczne

Konstrukcja systemu

- dostarczany wraz z oprogramowaniem wspierającym i gotowymi niektórymi aplikacjami
- udostępniający otwarte elementy pozwalające obsługiwać telefonię mobilną
- możliwość wymiany każdego standardowego elementu systemu na inny
- wirtualna maszyna Javy (Dalvik)
- brak implementacji szczegółów sprzętowych

Wyświetlanie interfejsu

- tradycyjne smartfonowe sposoby i możliwości
- grafika 2D i 3D w oparciu o OpenGL ES
- standard VGA

Niektóre możliwości systemu

- GSM/EDGE, CDMA, EV-DO, UMTS, 802.11 (wifi)
- Bluetooth, IrDA
- SQLite jako sposób organizacji pamięci masowej
- bardzo bogate multimedia (H.264, MP3, AAC, OGG, AMR, JPEG, PNG, GIF)
- usługi WWW w oparciu o silnik WebKit
- wsparcie dla różnych urządzeń (aparaty/kamerki, ekrany dotykowe, akcelerometry, GPS, układy GPU)

Instalacja oprogramowania

- sporo aplikacji dostępnych poprzez Android Market, pomimo małego rozpowszechnienia platformy
- możliwość skopiowania do urządzenia oprogramowania pod postacią pliku APK
- alternatywne menedżery pakietów (np. SAM, www.SlideME.org)

W skrócie

- linuks: poziom uprzywilejowany i zwykłego użytkownika, dobrze przetestowane jądro
- standardowo programy uruchamiane na wirtualnej maszynie
- ograniczony dostęp do możliwości sprzętowych telefonu

Podstawowe informacje

- oficjalnie nie jest wspierane tworzenie natywnych aplikacji
- jedynym oficjalnie wspieranym językiem jest implementacja Javy zwana Apache Harmony
- bogata dokumentacja (dostępnych jest wiele tutoriali i przykładowych aplikacji)
- SDK dostępne na wszystkie popularne platformy: Linux, Win, Mac (niestety nie jest open-source)

Software Development Kit

- biblioteki
- emulator (na bazie QEMU)
- debugger, profiler
- dokumentacja (razem z przykładowym kodem i tutorialami)
- Anroid Development Tools (ADT, plugin do Eclipse)
- wymagania: JDK, Ant, Python

Zalety

- oparty na linuxsie, dziedziczący po nim wiele istotnych mechanizmów
- otwarty kod, dzięki czemu wiadomo w jaki sposób wszystko działa i można to zmieniać
- niemalże kompletny system, wymagający tylko łatwego dostosowania do konkretnego sprzętu
- wspieranie wielu nowoczesnych technologii
- swoboda w instalowaniu oprogramowania
- łatwość tworzenia oprogramowania (język wysokiego poziomu, dużo narzędzi, dobra dokumentacja)

Wady

- brak wsparcia dla natywnego kodu
- mniejsza kontrola nad urządzeniem z poziomu Javy
- niezgodność oficjalnego języka z ogólnie przyjętymi standardami (Java SE/ME)
- mało rzeczywistych urządzeń działających pod kontrolą tego systemu

Windows Mobile

- System operacyjny firmy Microsoft na urządzenia mobilne
- Przez urządzenia mobilne rozumiemy telefony komórkowe i PDA
- Zaprojektowany, by przypominał desktopowy Windows
- Szeroka gama oprogramowania użytkowego dostarczanego przez Microsoft (Office Mobile, Outlook Mobile, Internet Explorer Mobile, Windows Media Player Mobile, ...)
- Wsparcie dla deweloperów

Kolejne wydania

- Pocket PC 2000
- Pocket PC 2002
- Windows Mobile 2003
- Windows Mobile 2003 SE
- Windows Mobile 5
- Windows Mobile 6
- Windows Mobile 6.1

Architektura

- Oparty na systemie Windows CE (Embedded Compact, wydany na architektury x86, MIPS, ARM, Hitachi SuperH)
- Wielozadaniowy system czasu rzeczywistego
- Priorytetyzowanie zadań
- Dziedziczenie priorytetów w celu zapobiegania zjawisku inwersji priorytetów zadań (wątek o niskim priorytecie blokuje zasób potrzebny wątkowi o wysokim priorytecie)
- Sterowniki urządzeń najczęściej wstawione na stałe do pamięci ROM

Platforma sprzętowa

- Windows Mobile 6.1 (oparty na Windows CE 5.2) jest dostępny w dwóch, nie całkiem zgodnych ze sobą wersjach
- Wersja dla smartfonów (rozumianych jako telefon komórkowy BEZ dotykowego ekranu)
 - Windows Mobile Standard
- Wersja dla PDA
 - Windows Mobile Classic (dla palmtopów bez funkcji telefonu)
 - Windows Mobile Professional (dla palmtopów z funkcją telefonu)

Niektóre możliwości systemu

- AJAX, JavaScript
- Bluetooth, IrDA
- WiFi (WPA), VoIP
- Pulpit zdalny (tylko w wersji dla PDA)
- Praca w rozdzielczości do 800x480
- wsparcie dla GPS
- DirectShow

Instalacja oprogramowania

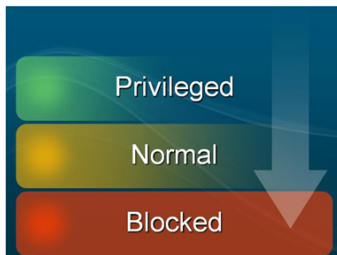
- Pliki instalacyjne *.cab
- Instalacja pomocą mechanizmu ActiveSync (synchronizacja z komputerem pc również multimediami, wiadomości, organizatora, poczty - także z serwerem MS Exchange)
- Instalacja przez skopiowanie pliku bezpośrednio na telefon i uruchomienie instalatora
- Instalacja przez pobranie pliku za pośrednictwem przeglądarki (instalator sam się uruchamia)
- Aplikacje muszą być cyfrowo podpisane przez producenta telefonu bądź operatora

Bezpieczeństwo

- Szyfrowanie danych
- Dostępne antywirusy (Symantec, Kaspersky Lab)

Uprawnienia

- Brak pojęcia aktualnego użytkownika (nie ma logowania, jest po prostu użytkownik korzystający z systemu)
- 3 poziomy uprawnień dla aplikacji



Uprawnienia

- Poziom uprzywilejowany - aplikacja może wszystko - pisać wszędzie po rejestrze, wywoływać wszystkie funkcje systemowe, modyfikować pliki systemowe, instalować certyfikaty
- Poziom normalny - najczęściej używany, aplikacja ma ograniczony dostęp do funkcji systemowych, nie może pisać po chronionej części rejestru, modyfikować plików systemowych ani instalować certyfikatów
- Poziom zablokowany - aplikacja na tym poziomie nie ma prawa w ogóle się uruchomić

Certyfikaty

- Uprzywilejowany - pozwala na uruchomienie programu na poziomie uprzywilejowanym
- Normalny - pozwala na uruchomienie programu z normalnymi uprawnieniami
- Certyfikatami zarządza operator usług telekomunikacyjnych

Polityka zabezpieczeń

- Ochrona wyłączona - każdy program można uruchomić (domyślnie w emulatorach)
- Ochrona jednopoziomowa - uruchomiony program ma dostęp do wszystkiego; program uruchamia się jeśli jest podpisany dowolnym certyfikatem bądź użytkownik się zgodzi (domyślnie w wersji dla PDA)
- Ochrona dwupoziomowa - rozróżnienie certyfikatów, jeśli program nie ma certyfikatu, użytkownik jest pytany, czy program może się uruchomić (domyślnie w wersji dla Smartfonów)
- Blokada Mobile2Market - jak poprzednio z tą różnicą, że nie ma możliwości uruchomienia niepodpisanego programu

Programowanie

- Dobre wsparcie techniczne (MSDN)
- Microsoft .NET Compact Framework
- Kod natywny
- Dostępne interpretery języków skryptowych (np. PythonCE)
- Brak stosu czyszczącego, nie używa się deskryptorów

Środowisko

- Zintegrowane środowisko programistyczne dostarczane razem z SDK i dokumentacją (Microsoft Visual Studio 2005+)
- Wygodny debugger
 - Debuggowanie przy użyciu emulatora
 - Debuggowanie przy użyciu podłączonego urządzenia

Microsoft .NET Compact Framework

- Okrojona wersja środowiska .NET (część pakietów zrealizowana w inny sposób)
- Języki C# oraz Visual Basic .NET
- Szybkie i wygodnie programowanie złożonych aplikacji
- Garbage collector
- Zamiast deskryptorów, obiektowy mechanizm operacji na napisach

Kod natywny

- Język C++
- Siła programowania na niższym poziomie, większa wydajność
- Nieco zmodyfikowane środowisko Win32
- Zarządzanie pamięcią za pomocą **new** i **delete**
- Zamiast deskryptorów, biblioteki standardowe udostępniające zaawansowane operacje na napisach

Przechowywanie danych

- Rejestr pozwalający przechowywać dane dotyczące aplikacji
- Wbudowany (w pamięci ROM) system zarządzania bazą danych MS SQL Server Mobile Edition
- Dane o kontaktach, zadaniach i spotkaniach dostępne jako jeden obiektowy typ danych (Pocket Outlook Object Model)

Gry

- Game Api dające pełną kontrolę nad sterownikiem karty graficznej
- Od Windowsa w wersji 5.0 wsparcie dla Direct3D Mobile

A gdzie moja Java...?

- Znaczna część urządzeń z systemem Windows Mobile ma zainstalowaną domyślnie wirtualną maszynę Javy (J2ME) pochodzącą od jakiejś firmy trzeciej
- Jeśli nie ma, to można sobie doinstalować (np. Esmertec Jbed)

Zalety

- Interfejs zbliżony do desktopowych systemów Windows
- Szczególnie menu “Start”, do którego większość ludzi jest przyzwyczajona
- Wsparcie dla aplikacji niemicrosoftowych
- Natywne api zbliżone do standardowego Win32
- Łatwe pisanie aplikacji
- Zdalny dostęp do komputera PC

Wady

- Zasobożerny
- Niezgodne ze sobą wersje systemu dla PDA i telefonów, co powoduje, że trzeba często dostosowywać kod źródłowy do konkretnej wersji
- Drogie środowisko programistyczne (Visual Studio)...
- i nie bardzo są alternatywy wobec niego...
- W Windows Mobile 5 dla Smartfonów nie było pakietu Office (w wersji 6 już jest)

Podpisy cyfrowe

- W Symbianie tak, restrykcyjnie sprawdzane
- W Windowsie tak, w zależności od konfiguracji różny poziom restrykcji
- W Androidzie tak, jednak służą jedynie weryfikacji tożsamości dewelopera, certyfikat nie musi być zatwierdzony przez zaufany urząd certyfikacji

Wirusy

- Na Symbiana dostępne antywirusy (Symantec, Kaspersky Lab, ...)
- Na Windowsa też (Symantec, Kaspersky Lab, ...)
- Na Androida dostępny antywirus VirusGuard, jednak jako że liczba modeli telefonów, na których aktualnie pracuje ten system nie jest zbyt duża (1?), nie bardzo pisze się na niego wirusy...

Uprawnienia

- Na Symbianie 3 poziomy uprawnień, w zależności od posiadanego certyfikatu
- Na Windowsie również 3 poziomy uprawnień, w zależności od posiadanego certyfikatu
- Na Androidzie uprawnienia przydzielane podczas instalacji programu, na podstawie dołączonego pliku manifest (użytkownik decyduje, czy przydzielić dany przywilej)

Programowanie

- W Symbianie skomplikowane programowanie aplikacji natywnych, ew. J2ME
- W Windowsie do wyboru programowanie aplikacji natywnych lub .NET Compact Framework, oba zbliżone do programowania aplikacji desktopowych
- W Androidzie programowanie z wykorzystaniem Android API (wariant Javy)

Środowiska

- Na Symbiana dostępna szeroka gama środowisk programistycznych
- Na Windowsa w zasadzie tylko drogie Visual Studio, ew. Embedded C++
- Na Androida - można pobrać Android SDK i plugin do Eclipse oraz narzędzia

Interaktywność

- Wszystkie trzy systemy wieloprocessowe
- Możliwość minimalizacji programów, korzystania z wielu naraz
- Wszystkie mają wsparcie dla ekranu dotykowego
- Dodatkowo Windows w wersji dla telefonów bez dotykowego ekranu, ze specjalnym interfejsem nastawionym na sterowanie jedną ręką

Pytania

Czy są jakieś pytania?

Bibliografia

- <http://en.wikipedia.org/wiki/Symbian>
- <http://dsonline.computer.org> (Operatng Systems/Archives/Mobile Operating Systems)
- <http://www.symbian.com>
- [http://en.wikipedia.org/wiki/Android_\(google\)](http://en.wikipedia.org/wiki/Android_(google))
- <http://www.android.com/>
- <http://code.google.com/android/documentation.html>
- <http://www.slideme.org/>

Bibliografia

- <http://msdn.microsoft.com/en-us/windowsmobile>
- <http://msdn.microsoft.com/en-us/library/aa454908.aspx>
- <http://technet.microsoft.com/en-us/library/cc512651.aspx>
- http://en.wikipedia.org/wiki/Windows_Mobile
- http://en.wikipedia.org/wiki/Windows_CE
- http://en.wikipedia.org/wiki/Windows_CE_5.0
- http://searchmobilecomputing.techtarget.com/news/article/0,289142,sid40_gci1196410,00.html