

28 maja 2013 r.

X Window System

Autorzy:

Jerzy Jelinek
Paweł Korpowski

Spis treści

Wstęp	3
Architektura	3
Klienty-serwer	3
Sieciowa transparentność	3
Wady	3
Komunikacja.....	4
Uwierzytelnianie.....	5
Metody	5
Tunelowanie.....	6
Zarządzanie oknami	6
Okna.....	6
Menadżer okien.....	7
GUI.....	7
Przyszłość.....	9
Źródła.....	9

Wstęp

System X Window został opracowany na Uniwersytecie MIT w 1984 r. Jest to graficzny system komputerowy, którego zadaniem jest zapewnienie podstawowych operacji do tworzenia okien. System ten zajmuje się obsługą urządzeń wejściowych, tworzy abstrakcyjną warstwę sprzętową, umożliwia rysowanie prostych kształtów oraz kolorowanie pikseli i wczytywanie bitmap. Początkowo najpopularniejszą implementacją było XFree86, jednak od 2004 roku z powodów licencyjnych, większość dystrybucji wykorzystuje X.Org. Aplikacja ta działa w architekturze klient-serwer, dane wymieniane są przez X Window System Core Protocol. Aktualna wersja to X11R7.7, co ciekawe pierwsza wersja X11 powstała w 1987 r. i stworzone wtedy programy są zgodne z aktualnymi i nadal mogą się z nimi komunikować.

Architektura

Klienty-serwer

W X Window role klienta i serwera są odwrócone w stosunku do klasycznych systemów tego typu. Często jest to powodem komplikacji, zwłaszcza dla niezaawansowanych użytkowników.

Serwer X jest uruchamiany na lokalnym komputerze użytkownika, wyświetlającym grafikę i łączy się z różnymi, zdalnymi klientami, czyli programami uruchamianymi na innych komputerach.

- Serwer:
 - zajmuje się obsługą urządzeń wejściowych - klawiatury, myszy, tabletu
 - wyświetla okna otrzymywane od klientów
 - może być słabym komputerem łączącym się z mocnymi maszynami
- Klient:
 - otrzymuje dane z urządzeń wejściowych serwera
 - przesyła wyniki swojego działania do serwera (w postaci okien)

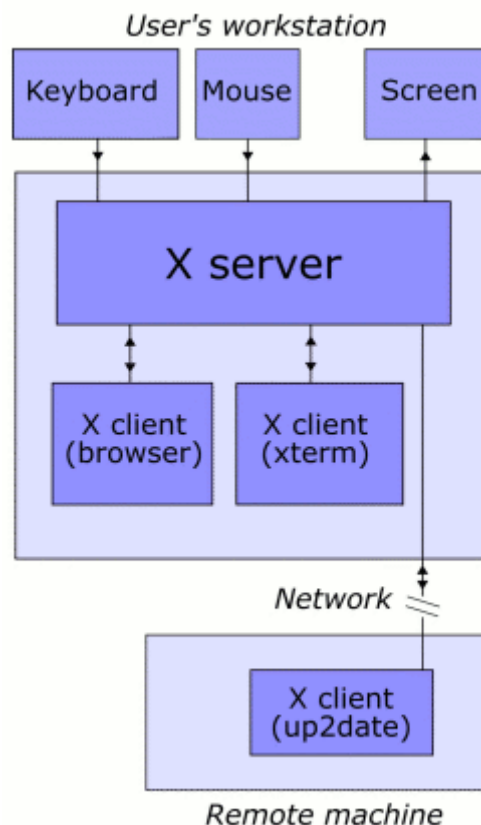
Sieciowa transparentność

Protokół z którego korzysta X jest przezroczysty sieciowo (sprawia wrażenie pojedynczego, zintegrowanego systemu) i powstał w czasach, gdy na jednym komputerze pracowało się za pomocą łączących się z nim "cienkich klientów" (klienci niezależni od serwera, np. korzystanie z systemu bankowego za pomocą przeglądarki www, operacje wykonywane są po stronie serwera). Klient i serwer mogą działać na tej samej bądź różnych maszynach (z różnymi architekturami i systemami operacyjnymi). Mogą łączyć się bezpiecznie poprzez Internet wykorzystując tunelowanie.

Wady

Kiedy klient i serwer działają na jednej maszynie, to ścieżka komunikacji staje się niepotrzebnie długa. Klient wysyła sygnał, serwer odbiera go, przesyła do sterowników, te

korzystając ze sprzętu przetwarzają go, zwracają rezultat serwerowi, a ten przesyła go do klienta. Stanowi to problem, kiedy przetwarzana jest zaawansowana grafika 3D. W 1998 r. rozszerzono X o DRI (Direct Rendering Infrastructure), czyli technologie pozwalającą na bezpośrednie renderowanie grafiki pomijając serwer X. Znacząco podniosło to wydajność, zwłaszcza aplikacji 3D. W 2007 r. pojawiło się DRI2. Obecnie Linuxy korzystają z dwóch niezwiązanych ze sobą sterowników - DDX do wolnych operacji serwerowych X (czyli np. obsługi okien) oraz DRI zajmujący się przetwarzaniem grafiki. Pojawia się pytanie, czy serwer X jest w ogóle potrzebny, skoro wydajniejsze jest renderowanie grafiki bez niego. Kolejną wadą jest konieczność uruchamiania serwera na prawach roota.



Rysunek 1 Schemat działania X Window

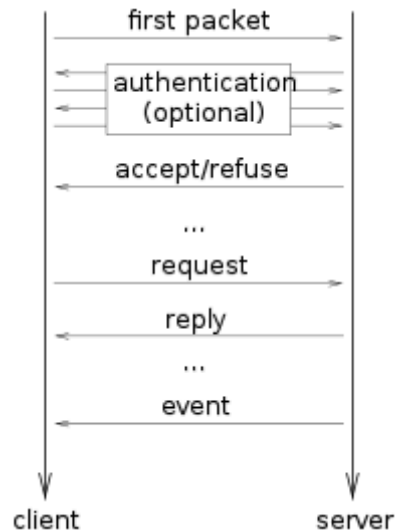
Komunikacja

W X11 komunikacja polega na wymianie pakietów przez otwarty kanał komunikacyjny. Połączenie jest nawiązywane przez klienta, który wysyła pierwszy pakiet zawierający kolejność bajtów jaka ma być używana podczas komunikacji oraz informacje na temat wersji protokołu i rodzaju uwierzytelniania jakiego klient oczekuje od serwera. Serwer odpowiada poprzez wysłanie pakietu z prośbą o dalsze uwierzytelnianie lub odmową dostępu. Jeśli połączenie jest zaakceptowane, pakiet akceptacji zawiera dane dla klienta, które będą wykorzystane w kolejnej interakcji z serwerem. Po ustanowieniu połączenia, między klientem a serwerem mogą zostać przesyłane cztery typy pakietów:

- Request - klient prosi o informacje z serwera albo prosi serwer o wykonanie akcji.
- Reply - serwer odpowiada na zapytanie klienta (nie wszystkie zapytania wymagają odpowiedzi).

- Event - serwer informuje klienta o wydarzeniu takim jak naciśnięcie klawisza, kliknięcie, przesunięcie okna, etc.
- Error - jeśli zapytanie jest niepoprawne serwer wysyła pakiet błędu. Ponieważ zapytania są kolejgowane, pakiet błędu może być wysłany z opóźnieniem.

Pakiety *request* i *reply* mają zmienną długość, natomiast pakiety *event* i *error* stałą, wynoszącą 32 bajty.



Rysunek 2 Pakiety przesyłane pomiędzy klientem, a serwerem

Uwierzytelnianie

Metody

X Window Serwer posiada pięć standardowych rodzajów mechanizmów kontroli, które klient może wykorzystać podczas łączenia. Można je pogrupować w 3 kategorie:

- dostęp oparty o hosta
- dostęp oparty o ciasteczka
- dostęp oparty o użytkownika

Host:

Metoda polegająca na określeniu, które komputery są uprawnione do łączenia się z serwerem X. Program xhost daje możliwość wyświetlenia oraz modyfikowania listy komputerów z dostępem do serwera. Nieprawidłowe użycie xhost może doprowadzić do tego, że każdy komputer w sieci będzie miał dostęp do naszego serwera X.

Ciasteczka:

Metoda jest oparta o wybór pliku ciasteczka i przekazaniu go do serwera podczas jego uruchamiania. Dostęp do serwera X uzyskują ci klienci, którzy posiadają odpowiedni plik ciasteczka. Ciasteczka są tworzone poprzez oddzielny program i standardowo przechowywane w pliku `.Xauthority`, w katalogu użytkownika. Plik ten wykorzystywany jest przez klienta do

uwierzytelniania się na serwerze X. Chcąc uzyskać dostęp z innego komputera w sieci, ciasteczko musi zostać skopiowane na ten komputer (np. wykorzystując scp). Dwoma systemami wykorzystującymi tę metodę są MIT-MAGIC-COOKIE-1 i XDM-AUTHORIZATION-1. W pierwszym klient wysyła plik ciasteczka, kiedy jest proszony o uwierzytelnienie, w drugim oprócz pliku ciasteczka w .Xauthority przechowywany jest też tajny klucz. Klient tworzy ciąg znaków poprzez połączenie aktualnego czasu, identyfikatora zależnego od transportu i ciasteczka. Następnie ciąg jest szyfrowany i wysyłany na serwer. Aplikacją do obsługi pliku .Xauthority jest xauth.

Użytkownik:

Metoda pozwala na dostęp do serwera konkretnym użytkownikom. Klient łącząc się z serwerem musi udowodnić, że posiada prawa do korzystania z niego. Dwie metody oparte na uwierzytelnianiu użytkowników to: SUN-DES-1 i MIT-KERBEROS-5. Pierwszy system oparty jest o mechanizm bezpiecznego, zdalnego wywoływania procedury, i był rozwijany w SunOS. Drugi to protokół uwierzytelniania Kerberos.

Tunelowanie

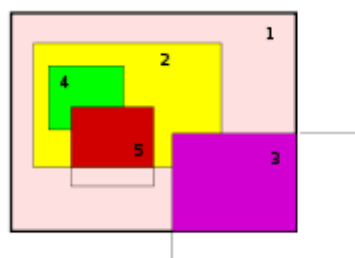
Połączenie między serwerem a klientem może być chronione poprzez bezpieczne protokoły takie jak SSL, SSH.

Zarządzanie oknami

Okna

To, co w GUI nazywane jest oknem, w X jest top-level window. Termin okno jest używany także dla okien leżących w innych oknach (podokien). Wszystkie graficzne elementy takie jak guziki, menu, ikony są realizowane przy użyciu okien.

Okna mogą być tworzone tylko jako podokna rodzica, tworzy to strukturę drzewiastą. X server automatycznie tworzy korzeń drzewa (root window).



Rysunek 3 Okna w X

Serwer X przechowuje wszystkie informacje dotyczące okien, czcionek etc. Klient zna identyfikatory tych obiektów - integery. Klient może tworzyć okna z konkretnym identyfikatorem. Identyfikatory są unikalne nie tylko w obrębie klienta, ale także serwera. Żadne dwa okna nie mogą mieć tego samego identyfikatora, nawet jeśli są stworzone przez różnych klientów.

Menadżer okien

Jest to program odpowiadający za wyświetlanie i położenie okien w graficznym interfejsie użytkownika w X Window System, umożliwia pracę z wieloma programami w tym samym czasie.

Różnice pomiędzy menadżerami okien występują w:

- konfiguracji wyglądu oraz funkcjonalności
 - menu tekstowych
 - programów lub zmieniania ich ustawień
 - paneli i innych graficznych metod uruchamiania programów
 - liczby pulpitów oraz wirtualnych pulpitów (pulpity, które są większe niż fizyczny rozmiar monitora) i metod przełączania się pomiędzy nimi
- ilością zajmowanej pamięci i innych zasobów systemowych
- stopnia integracji ze środowiskiem

Przykłady menadżerów okien wykorzystywanych w X Window System :

- Blackbox
- Compiz
- Fluxbox
- IceWM
- KWin (dawniej KWM, domyślny menadżer okien KDE)
- Metacity (menadżer okien dla GNOME 2.x)
- Openbox
- Window Maker

GUI

Definicje za Wikipedią:

- GUI - graficzny interfejs użytkownika, nazywany też środowiskiem graficznym – ogólne określenie sposobu prezentacji informacji przez komputer oraz interakcji z użytkownikiem, polegającego na rysowaniu i obsłudze widżetów.
- Widżet - Podstawowy element graficznego interfejsu użytkownika (np. okno, pole edycji, suwak, przycisk).

Graficzne interfejsy użytkownika wykorzystują system X Window oraz menadżery okien. Różnica pomiędzy menadżerem okien, a GUI jest taka, że menadżer zapewnia jedynie zarządzanie oknami (fragmentami ekranu), natomiast środowisko graficzne dostarcza nie tylko menadżera okien, ale też szereg podstawowych aplikacji łatwych w obsłudze nawet dla początkującego użytkownika.

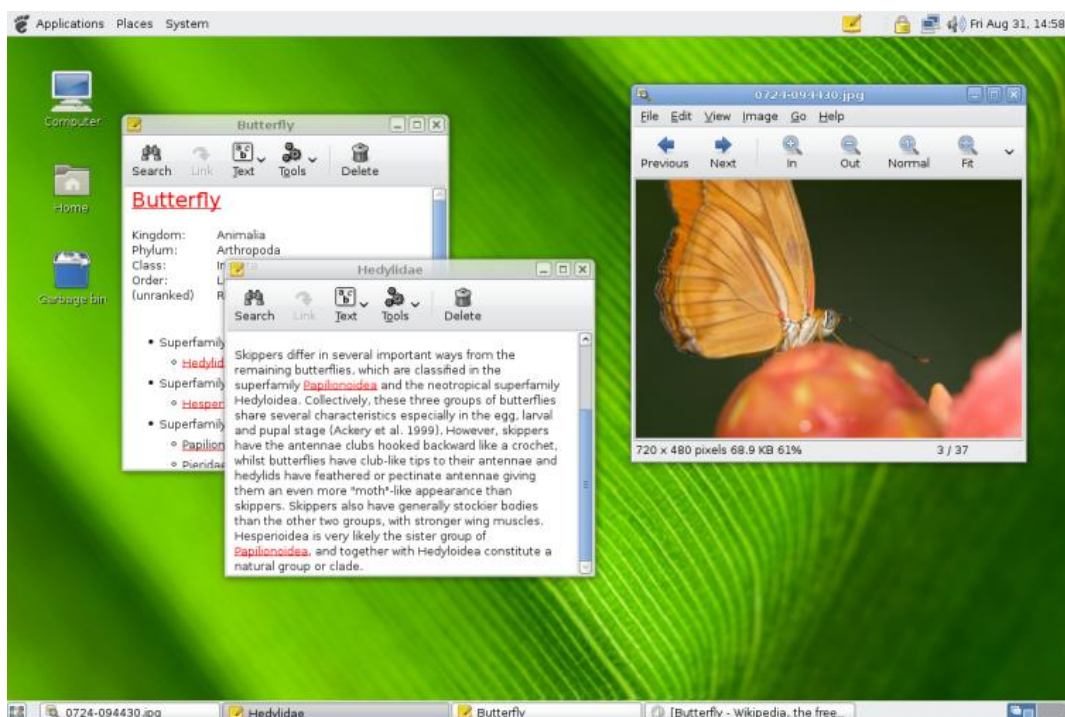
Najpopularniejsze graficzne interfejsy użytkownika to:

- KDE
 - powstało w 1996 r.
 - korzysta z biblioteki graficznej Qt (do 97 r. nie było to wolne oprogramowanie)
 - zapewnia dużą konfigurowalność aplikacji
 - menadżerem okien jest KWin



Rysunek 4 Wygląd KDE

- GNOME
 - projekt rozpoczął się w 1997 r., pierwsze wydanie w 1999 r.
 - alternatywa dla KDE, na licencji GPL
 - kładzie nacisk na prostotę obsługi
 - menadżer okien to Mutter (od wersji 3.0)



Rysunek 5 GNOME

Przyszłość

Jako następcę X Window postrzega się system Weyland. Pretenduje on do miana prostszego zamiennika dla X, łatwiejszego w rozwijaniu i utrzymaniu. Jest to protokół systemu okien stworzony w języku C dla systemu Linux. Dzięki niemu ułatwione jest wyświetlanie grafiki poprzez połączenie menadżera kompozycji oraz systemu okien w jedną aplikację. Projekt ten został zapoczątkowany przez Kristiana Høgsberga w 2008 roku. Zostało zapowiedziane, że GNOME i KDE zostaną przeniesione na Weyland. Projekt jest w początkowej fazie rozwoju i ma jeszcze kilka ograniczeń:

- Utrata transparentności sieciowej
- Brak natywnego OpenGL
- Zamknięte sterowniki do grafik AMD i NVIDIA nie wspierają protokołu Wayland

Źródła

http://sequoia.ict.pwr.wroc.pl/~witold/unixintro/xwinintro_s.pdf

http://en.wikipedia.org/wiki/X_Window_System_protocols_and_architecture

http://en.wikipedia.org/wiki/X_Window_System

<http://wayland.freedesktop.org/>

<http://zsm-e.cal.pl/presentations/so/12.3.pdf>

<http://linux4u.w.interia.pl/pdf/X%20Window%20System.pdf>

<http://www.cyberciti.biz/tips/running-x-window-graphical-application-over-ssh-session.html>