

# Wprowadzenie do systemu X Window

Witold Paluszyński  
Katedra Cybernetyki i Robotyki  
Politechnika Wroclawska  
<http://www.kcir.pwr.edu.pl/~witold/>

2001–2013



Ten utwór jest dostępny na licencji  
**Creative Commons Uznanie autorstwa-  
Na tych samych warunkach 3.0 Unported**

Utwór udostępniany na licencji Creative Commons: uznanie autorstwa, na tych samych warunkach. Udziela się zezwolenia do kopiowania, rozpowszechniania i/lub modyfikacji treści utworu zgodnie z zasadami w/w licencji opublikowanej przez Creative Commons. Licencja wymaga podania oryginalnego autora utworu, a dystrybucja materiałów pochodnych może odbywać się tylko na tych samych warunkach (nie można zastrzec, w jakikolwiek sposób ograniczyć, ani rozszerzyć praw do nich).

# Założenia i historia systemu X window

**System X Window** został opracowany na uniwersytecie M.I.T. w roku **1984** jako standard przemysłowy tworzenia systemów okienkowych, niezależnych od platformy sprzętowej i programowej, w środowisku sieciowym. W założeniach miał być zaimplementowany powyżej systemu operacyjnego komputera, i umożliwiać komunikację (wyświetlanie okienek i obrazków) między różnymi rodzajami komputerów, różnych producentów, z gdzie indziej napisanym oprogramowaniem, i być może pod kontrolą innego administratora systemu.

Cel sformułowany przez twórców systemu został osiągnięty. W roku **1987** wprowadzono wersję 11 systemu (nazywanego też X11), która jest jego bieżącą wersją. W tym czasie powstało kilka kolejnych wydań (*release*) rozszerzających funkcjonalności systemu. Najnowszym jest wydanie 7, tzw. X11R7, z roku 2005 (11 lat po R6). Jednak poza pewnymi rozszerzeniami zasadniczy protokół komunikacyjny systemu X Window pozostał w wersji 11 niezmienny, i programy uruchomione np. w roku 1987 mogą komunikować się z programami aktualnie kompilowanymi.

X Window nadzoruje obecnie *X Consortium* zrzeszające producentów, firmy, i organizacje zainteresowane użytkowaniem i rozwojem systemu.

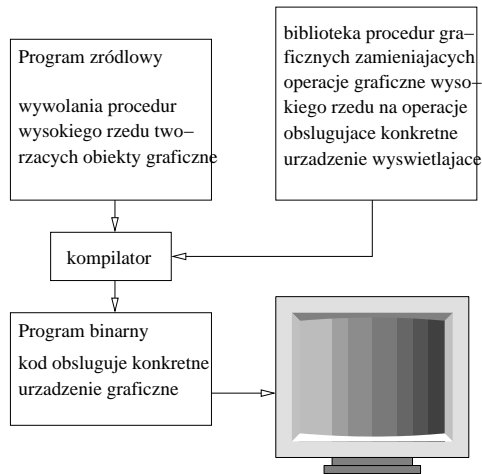
## System X Window — znaczenie społeczne

Od początku system X Window był projektem otwartym — zakładał całkowitą jawność opracowanej koncepcji i zamiar udostępnienia systemu zainteresowanym jednostkom. M.I.T. od początku udostępniał wszystkie opracowane programy w wersji źródłowej zarówno producentom oprogramowania, jak i użytkownikom.

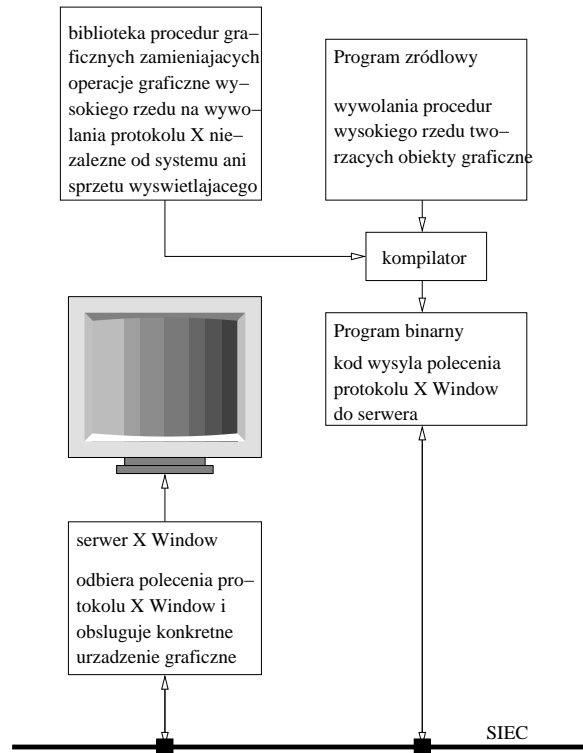
Między innymi dlatego system został od początku zaakceptowany przez międzynarodowe konsorcjum producentów sprzętu komputerowego jako standard systemu okienkowego, i koncepcja tego systemu jest najpowszechniej przyjętą koncepcją takiego systemu.

# Model konstrukcji programu graficznego

Tradycyjny schemat programu graficznego



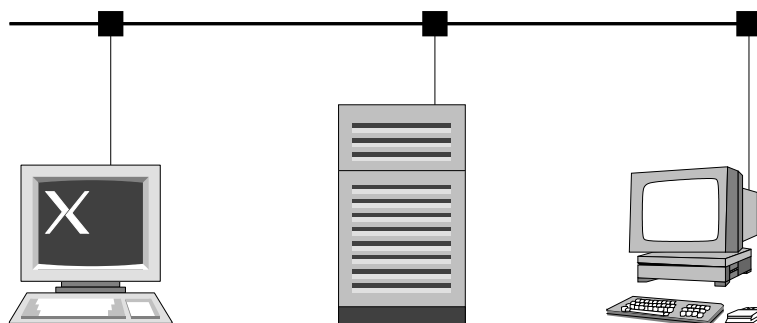
Model X Window programu graficznego



## Podstawowe pojęcia i architektura systemu X Window

Składniki systemu X Window:

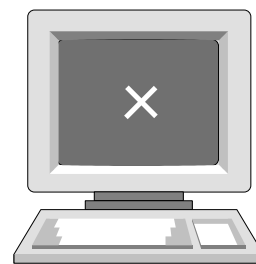
- **serwer ekranowy:** urządzenie użytkownika (ekran, klawiatura, mysz), i program obsługujący; najczęściej serwer jest uruchamiany na stacji roboczej albo komputerze typu PC, aczkolwiek istnieją sprzętowe realizacje serwerów X Window,
- **klienty:** programy wyświetlające; mogą być uruchamiane na tym samym komputerze co serwer X Window, albo gdziekolwiek w Internecie, w tym również z komputera o innej architekturze, procesorze, systemie operacyjnym,
- **protokół X** komunikacji klientów z serwerami.



# Serwer X Window

Rola serwera X Window — obsługa **zdarzeń serwera**:

- odbieranie sygnałów od myszy i z klawiatury oraz przekazywanie ich klientowi aktywnemu (ang. *focus*),
- odbieranie poleceń i zapytań klientów i ich realizacja.



Serwer X Window jest programem, który standardowo nazywa się X, zatem normalnie jego uruchomienie polega po prostu na wywołaniu tego programu.

Uruchomienie „gołego” serwera X Window jest niewystarczające, ponieważ pomimo iż jest on programem użytkownika, sam nie może być przez niego sterowany. **Co najmniej pierwszy klient musi być uruchomiony z zewnątrz.** Tradycyjnie, najważniejszym klientem, zwykle uruchamianym w domyślnej konfiguracji, był emulator terminala, np. **xterm**. Dzięki niemu, użytkownik serwera uzyskiwał zdolność uruchamiania dowolnych programów okienkowych (klientów X Window), konfiguracji systemu, i ogólnie pracy.

## Program xinit

Program **xinit** pozwala wygodnie uruchamiać serwer X Window. Wywołuje on serwer X, i od razu powoduje wykonanie skryptu (pliku) `~/.xinitrc`:

```
xedit -geometry -0-0 ~/.xinitrc &
xclock -geometry -0+0 &
xset s on s 300
xsetroot -solid '#006060'
xterm
```

Pierwsze dwa wiersze uruchamiają klienty okienkowe jako procesy tła (& na końcu). Programy te pracują trwale, do momentu ich jawnego zakończenia.

Kolejne dwa wiersze uruchamiają klienty X Window, które konfiguruje serwer, wydając mu pewne polecenia. Nie wyświetlają one okienek, i szybko kończą pracę, dlatego nie jest konieczne ich uruchamianie w tle.

Ostatni wiersz uruchamia znów „trwały” program okienkowy **xterm**. Jest on uruchamiany jako pierwszoplanowy, ponieważ po wykonaniu całego skryptu **xinit** zamyka serwer X i kończy pracę. Gdyby ostatni klient był uruchomiony w tle, cała sesja byłaby krótka i zostałaby automatycznie zamknięta. Wykonanie ostatniego klienta w pierwszym planie powoduje, że sesja trwa dopóki działa ten ostatni klient. Pełni on więc rolę sterującą dla serwera X Window. Zakończenie **xterm** przez użytkownika powoduje również zakończenie sesji okienkowej.

# Konwencje i mechanizmy systemu X Window

Zwróćmy uwagę na pewne wbudowane mechanizmy systemu X Window:

## Operacje myszą:

- lewy klawisz myszy wybiera i zaznacza (również 2-click, 3-click),
- środkowy wkleja zaznaczony tekst,
- prawy uruchamia menu kontekstowe (jeśli klient ma takie).

**Stos okienek** — Okienka wyświetlane są na ekranie w kolejności stosowej — te później wyświetlone mogą przykryć te wcześniejsze. Ten stos okienek jest stały i użytkownik nie może nim sterować. Jeśli jednak klient wyświetlający okienko na szczycie stosu zostanie zakończony, to odsłoni się okienko poprzedniego klienta, itd.

**Wybór okna/klienta aktywnego (focus)** — Okno w którym jest wskaźnik myszy jest aktywne. Ten klient otrzymuje znaki z klawiatury i zdarzenia myszy. Gdy wskaźnik myszy jest w tle ekranu, żadne okno nie jest aktywne.

Nie trzeba klikać na okienko, aby uczynić je aktywnym. Jednocześnie aktywność okienka nie jest związana z wyświetlaniem na szczycie stosu. Ten domyślny sposób wyboru aktywnego nazywa się: *focus-follows-mouse (FFM)*.

## Dynamiczna konfiguracja serwera X Window

Serwer X Window posiada wiele cech i funkcji, z których niektórymi można sterować w czasie pracy. Istnieje dedykowany klient `xset`, który wysyła pewne polecenia serwerowi, zgodne z argumentami wywołania, np.:

```
xset r off          # wylacza autorepeat na klawiaturze
xset s on s 300    # wygaszanie ekranu po 300 sekundach
```

`xset` pozwala również ustawiać serwerowi ważniejsze parametry, jak np. ścieżka dostępu do czcionek (patrz niżej), albo parametry sterowania zasilaniem monitora (są jednak one zależne od konkretnej implementacji serwera).

Program `xsetroot` pozwala ustawiać parametry wyświetlanego okna serwera, które ma postać tła ekranu X Window.

```
xsetroot -solid '#006060'
```

Istnieją jeszcze inne funkcje i parametry serwera, które można ustawiać w czasie pracy, jak np. autoryzacja wyświetlania albo zasoby, i służą do tego specjalne klienty. Będzie o nich mowa dalej.

# Klienty X Window

- Połączenie klientów z serwerem: o ile serwer normalnie komunikuje się z wieloma klientami jednocześnie, obsługując ich żądania, to każdy klient typowo wysyła dane do wyświetlania do jednego konkretnego serwera.
- **Zdarzenia klienta:** sygnały z klawiatury, od myszy, a także inne zdarzenia przekazywane klientom przez serwer, np. zdarzenie odślonięcia.
- Standardowe argumenty wywoływania klientów określają takie parametry jak:
  - adres serwera: `-display adres-ip:0.0`
  - geometrię: `-geometry szer× wys±xoff±yoff`
  - czcionki: `-fn '-misc-fixed-bold-r-normal-15-140-75-75-c-90-iso8859-2'`
  - kolory: `-fg yellow -bg blue -bd red`
  - inne: `-title xxx -iconic ...`
- Wiele klientów ma wiele parametrów konfiguracyjnych zwanych **zasobami**, o których niżej.
- Poza tym niektóre rozbudowane współczesne klienty X Window, jakimi są np. firefox, acroread, openoffice, itp. mają swoje własne systemy konfiguracji. Podobnie graficzne środowiska, zwane pulpitemi, będące konfiguracjami okienek, ustawień, konwencji, i protokołów komunikacji między klientami.

## Zestaw klientów X Window

### Zarządzanie systemem X Window

xinit, startx - uruchamianie serwera X  
xdm - zarządzanie uruchamianiem serwerów  
xset - konfigurowanie pracującego serwera  
xsetroot - ustawianie parametrów tła  
xhost - autoryzacja dostępu przez IP  
xauth - zarządzanie kodami autoryzacji  
xrdp - obsługa zasobów (ładowanie, wysw.)  
editres - edycja wartości zasobów klienta  
xmodmap - mapowanie klawiatury i myszy  
xrefresh - odświeżanie zawartości ekranu

### Informacje o systemie X Window

xdpyinfo - wysw.informacji o ekranie  
xwininfo - wyswietl.informacji o oknie  
xprop - wysw.własności okien i czcionek  
xlsfonts - wyswietlanie listy czcionek  
xfontsel - wybieranie czcionek  
xfd - wyswietlanie czcionek

### Użytkowe

xbiff - sygnalizacja poczty e-mail  
xcalc - kalkulator  
xclipboard - text clipboard  
xclock - okno z zegarem  
xedit - prosty edytor tekstowy  
xkill - kasowanie klienta przez wybór okna  
xload - wysw.obciążenia systemu  
xlock - zabezpieczenie klawiatury i ekranu  
xlogo - wysw.logo systemu X Window  
xterm - okienkowy emulator terminala

### Graficzne

bitmap - tworzenie małych obrazków  
xmag - powiększenie fragmentu ekranu  
xwd - zrzut ekranu do pliku w spec.formacie  
xwd - wyswietlanie zrzutu ekranu z pliku  
xpr - drukowanie zrzutu ekranu na drukarce  
xdpr - bezpośr.drukowanie zrzutu ekranu

## Zarządca okien

- Zarządca okien (*window manager*) jest klientem systemu X Window o specjalnych prawach; może być tylko jeden zarządca okien dla danego serwera według zasady: kto pierwszy ten lepszy.
- Funkcje zarządcy okien: rozmieszczanie i rozmiar okien klientów na ekranie (zarządca może uznać — lub nie — geometrie podawane przez klientów), przesuwanie, zmiana rozmiaru okien, zmiana okien w ikony, i z powrotem, przenoszenie wyświetlanych okien na szczyt lub spód stosu (*raise/lower*), itp.
- Do interakcji z użytkownikiem zarządcy okien tworzą **dekoracje okien** (belki z przyciskami sterującymi i ramki okien też pełniące rolę przycisków), specjalne klienty-belki sterujące, oraz menu operacji związane z tłem ekranu.
- Różni zarządcy okien mają parametry konfigurujące ich zachowanie.
- Niektórzy zarządcy okien realizują minimalną funkcjonalność, i są niemal niewidoczni. Inni mają rozbudowaną funkcjonalność i narzucają wygląd i zachowanie okien klientów, oraz wygląd i organizację całego ekranu. Ci pierwsi zwykle pracują samodzielnie, natomiast drudzy często stanowią część bardziej rozbudowanego środowiska graficznego zwanego pulpitem.

## Zarządcy okien — wymagania

W początkowym okresie, gdy karty graficzne miały ograniczoną pamięć i możliwości, a monitory niewielkie rozmiary i pamięć, naturalna była tendencja do realizacji wszystkich operacji predefiniowanymi sekwencjami kliknięć myszy, oraz skrótami klawiaturowymi. Tworzenie dedykowanych przycisków dla tych operacji wydawało się wtedy marnowaniem cennego miejsca na ekranie (tzw. *screen real estate*).

Później, w miarę pojawiania się tanich monitorów i kart graficznych o większych możliwościach, środowiska graficzne były rozbudowywane. Towarzyszyli im zarządcy okien z bogatą konfiguracją ustawianą interakcyjnie, z systemu menu. Jednak nadal okazjonalnie konieczne było odpalenie minimalnej, niezawodnej sesji okienkowej (tzw. *failsafe session*), gdy uruchomienie pełnego środowiska takiego jak Gnome, było niemożliwe z jakiegoś powodu (zwykle błędu).

Ostatnio pojawia się znowu zapotrzebowanie na skromniejsze interfejsy użytkownika, dające się używać na 10-calowych netbookach, a nawet 7-calowych notepadach. Pojawiło się (i nadal pojawia) wiele implementacji zarządcy okien odpowiadających nowoczesnym standardom pulpitu graficznego, ale z interfejsem w stylu historycznego minimalizmu. Warto przynajmniej znać jakąś konfigurację tego typu.

## Zarządcy okien — podstawowe operacje (geometria)

Podstawowymi operacjami realizowanymi przez zarządców okien są przesuwanie i zmiana rozmiaru okienek, czyli ogólnie **zmiana geometrii** okienka.

Prawie wszyscy zarządcy okien realizują te operacje, ale na ogół nie są one konfigurowalne. Często do zmiany geometrii służą dedykowane kontrolki związane z okienkami, tzw. **dekoracje**, zwykle w belce nad oknem, i przyciski wbudowane w obramowania okienek.

Jednak minimalistyczni zarządcy okien czasem mają swoje predefiniowane operacje myszą albo skróty klawiaturowe. Trzeba dobrze znać danego zarządcę, aby posługiwać się nim sprawnie. W nagrodę nie tracimy miejsca na ekranie.

## Zarządcy okien — ikonizacja/ukrywanie

Wielu zarządców okien, chociaż nie wszyscy, umożliwia redukcję okien do ikonek (zwaną minimalizacją), albo pewien jej wariant zwany ukrywaniem okienek.

Konkretny mechanizm jest na ogół charakterystyczny dla zarządcy okien i jest elementem jego wyglądu i stylu. Niekiedy jednak pewne elementy są konfigurowalne, np. niektórzy zarządcy, którzy umieszczają ikony okien w dowolnym miejscu na ekranie, mają również opcję wyświetlenia pojemnika na ikony (*icon box*). Niektóre środowiska graficzne stosują pewien wariant pojemnika na ikony w postaci **belki icon**, która zwykle jest współpracującym z zarządcą okien klientem (ale bez żadnych specjalnych uprawnień).



## Zarządcy okien — menu aplikacji

Niektórzy zarządcy okien umożliwiają uruchamianie nowych klientów przez kliknięcie myszą. Ta operacja jest na ogół realizowana przez **menu aplikacji** związane z oknem bazowym serwera (tłem, lub pulpitem), ponieważ nie ma ona nic wspólnego z żadnym z działających klientów.

Menu aplikacji można tworzyć i edytować albo za pośrednictwem pliku konfiguracyjnego, albo interakcyjnie, przez akcje zarządcy okien, albo dedykowany edytor konfiguracji (ta konfiguracja również jest zapisywana w pliku konfiguracyjnym, ale bez dokumentacji, niedostępnym bezpośrednio dla użytkownika).

## Zarządcy okien — wybór okna aktywnego (focus policy)

Przypomnijmy: domyślnym, wbudowanym w serwer X Window sposobem wyboru klienta aktywnego (*focus policy*) jest *focus-follows-mouse (FFM)*. Jednak operacje wykonywane przez serwer X Window na polecenie zarządcy okien umożliwiają implementację innego sposobu wyboru aktywnego klienta.

Ten sposób, zwany *click-to-focus (CTF)*, wymaga kliknięcia myszą na okno aby stało się ono aktywne. Wtedy, dodatkowo, jest ono wysuwane na szczyt stosu wyświetlanych okien (w istocie, w tym trybie operacja uaktywnienia okna, i przeniesienie go na szczyt stosu wyświetlanych jest jedną i tą samą operacją). Ponieważ tak zachowują się systemy Microsoft Windows i Mac OS, wiele zarządców okien systemu X Window implementuje to zachowanie jako domyślne.

Tryb *focus-follows-mouse (FFM)* jest chętniej stosowany z minimalistycznymi zarządcami okien, ponieważ umożliwia pracę również w oknie, które nie jest na szczycie stosu, i być może jest tylko częściowo wyświetlane. Co ciekawe, w tym trybie często przydatna okazuje się operacja przeniesienia okna na spód stosu (*lower*), która w trybie *click-to-focus (CTF)* nie ma w ogóle sensu.

Pewien wariant metody FFM, zwany *sloppy focus*, polega na tym, że klient nie przestaje być aktywny gdy wskaźnik myszy przesunie się z jego okna na tło.

## Zarządcy okien — porównanie

program	opis	geometria	ikonizacja	menu apl.	wyb.aktyw.
Historyczne					
olwm	Open Look WM	dekoracje	dekor.,pulpit	plik konf.	FFC/bez wys
twm	Tom's WM	dekoracje	dek.,pulp/pojem	plik konf.	FFM+CTF
Minimalistyczne					
tinywm	wzorcowy	predef.	?	-	FFM
aewm	Exiguous WM	predef.	predef.,ukryw.	-	FFM
evilwm	oparty na aewm	predef.	-	tylk.term	FFM
jwm	Joe's WM	dekor.	dekor.,belka	pl.konf.	FFM
lwm		spec.dek.	ukryw.,panel	-?	FFM
blackbox					
fluxbox					
openbox					
icewm					
Desktopowe					
dtwm	el.syst.CDE				CTF+FFM
xfwm4	el.syst.XFCE			-?	CTF+FFM
metacity	el.syst.GNOME			GNOME	CTF+FFM
compiz	el.syst.GNOME?				
enlightenment	rozbudowany				

### Własności wybranych zarządców okien

Spośród historycznych zarządców okien X11 ciekawy jest `twm` (Tom's window manager). Jest minimalny ale dalece konfigurowalny, i tradycyjnie był elementem bazowego pakietu X11. Jego konwencje są trochę nietypowe jak na dzisiejsze standardy, ale przez wielu użytkowników był traktowany jako opcja *fallback*, czyli: użyj gdy zawiedzie podstawowa konfiguracja. (Niestety, w wielu dystrybucjach Linuksa nie jest domyślnie instalowany i trzeba go doinstalować.) Pojawiły się też jego nowsze wersje: `ctwm` i `tctwm`.

-> Łatwa konfiguracja w pliku `~/ .twmrc`.

-> Root window menu pozwala uruchamiać dowolne programy.

Nowszym, minimalistycznym, ale mimo to funkcjonalnym zarządcą okien jest `evilwm`. Nie zapewnia ikonizacji i nie ma menu aplikacji. Pozwala przesuwając i zmieniać rozmiar, a także obniżać okna, klawiszami myszy +Alt, np. Alt-left realizuje przesunięcie, Alt-middle zmianę rozmiaru, a Alt-right przeniesienie na spód stosu (*lower*). Aktywne są również obramowania okien, ale w praktyce trzeba je nieco powiększyć od domyślnej szerokości 1 piksela.

Ma również szereg predefiniowanych skrótów klawiszowych z Ctrl-Alt, np. Ctrl-Alt-Return uruchamia emulator terminala, klawisze Ctrl-Alt-[hjk1] przesuwają okno, a Ctrl-Alt-= i Ctrl-Alt-x maksymalizują.

## Zarządcy okien — skróty klawiaturowe

Tradycyjnie zarządcy okien realizowali poniższe skróty klawiaturowe, co pozwalało na pewne przyspieszenie pracy na małych ekranach:

ALT + F3	- lower window
ALT + F4	- close window
ALT + F5	- restore window
ALT + F7	- move window
ALT + F8	- resize window
ALT + F9	- minimize window
ALT + F10	- maximize window

Obecnie większość zarządców okien predefiniuje tylko część z tych konwencji (np. Alt-F7, Alt-F9), ale pozwala zdefiniować dowolne inne.

## Praca z systemem X Window w środowisku sieciowym

Gdy mamy wiele serwerów i wielu klientów, musimy wskazać klientowi, na który serwer ma wysyłać żądania wyświetlania, za pomocą tzw. specyfikacji ekranu:

```
[adresIPkomputera]:numerserwera[.numerekranu]
```

Specyfikację tę można podać klientom za pomocą argumentu wywołania `-display`:

```
xcalc -display 156.17.9.1:0
```

albo za pomocą zmiennej środowiskowej `$DISPLAY`:

```
setenv DISPLAY lab103-2.ict.pwr.wroc.pl:0.0
xterm &
```

W specyfikacji serwera brak numeru ekranu oznacza domyślny numer 0, a brak adresu IP komputera oznacza adres IP lokalnego komputera. Zamiast adresu lokalnego komputera można też podać nazwę specjalną "unix". Natomiast numer serwera jest obowiązkowy i nie może być domyślny. Zatem najprostsza specyfikacja ekranu ma postać `:0` i jest równoważna `unix:0` oraz `localhost:0`.

Uruchamianie klientów na innym (zdalnym) komputerze niż serwer, przez telnet, rsh lub ssh:

```
ssh diablo.ict.pwr.wroc.pl xterm -display ja.t16.ds.pwr.wroc.pl:0
```

```
ssh diablo.ict.pwr.wroc.pl
setenv DISPLAY ja.t16.ds.pwr.wroc.pl:0
xterm &
```

## Zezwolenie na wyświetlanie na serwerze

System X Window posiada mechanizm pozwalający serwerowi kontrolować, od których klientów będzie on przyjmował żądania. Istnieją dwa mechanizmy kontroli dostępu. Prostszy (starszy i bardziej prymitywny) mechanizm oparty jest na adresie IP klienta. Można zadać listę adresów, z których serwer będzie przyjmował żądania zgłaszających się klientów. Można również całkowicie wyłączyć kontrolę dostępu do serwera.

Np.:

```
xhost
xhost +diablo.ict.pwr.wroc.pl
xhost
xhost -diablo.ict.pwr.wroc.pl
xhost
```

albo

```
xhost +
```

(Ostatnie wywołanie całkowicie wyłącza kontrolę dostępu.)

## Autoryzacja wyświetlania na serwerze

Nowszy i skuteczniejszy mechanizm kontroli dostępu do serwera opiera się na wykorzystaniu kodów autoryzacji. Serwer tworzy w chwili rozpoczęcia sesji użytkownika kod autoryzacyjny sesji i zapisuje go na pliku `~/.Xauthority`. Kod autoryzacyjny można przesłać na komputer, z którego uruchamiane będą klienty, które normalnie szukają tego kodu w pliku zadanym zmienną środowiskową `XAUTHORITY`.

Wyświetlenie na ekranie wszystkich kodów autoryzacji z domyślnego pliku:

```
xauth list
```

Specyfikacja serwera, jaką podaje się programowi `xauth` musi zgadzać się ze specyfikacją podaną w chwili generowania danego kodu. Czyli jeśli np. kod wygenerowany został na komputerze `diablo.ict.pwr.wroc.pl` dla serwera X zapisanego jako `:0`, to nie możemy tego kodu wykorzystać na innym komputerze, ponieważ tam specyfikacja serwera `:0` byłaby źle zinterpretowana przez klientów. Można jednak podawać specyfikacje poprzedzone adresem IP serwera X w formacie:

```
adresIPkomputera/[adresIPkomputera]:numerserwera[numerekranu]
```

## Manipulowanie kodami autoryzacji

Skuteczne posługiwanie się systemem X Window w sieci, gdy niektóre klienty są wywoływane ze zdalnych komputerów, wymaga jawnego przekazywania kodów autoryzacyjnych. Na przykład, **odczytanie kodu autoryzacji** domyślnego serwera, oraz **przesłanie go i zapisanie na zdalnym komputerze**, pozwala zdalnym klientom autoryzować się przy wyświetlaniu na lokalnym serwerze:

```
xauth extract - $DISPLAY | ssh diablo.ict.pwr.wroc.pl xauth merge -
```

W razie potrzeby można zażądać by lokalny serwer (:0) **wygenerował nowy kod autoryzacji** sesji:

```
xauth generate :0 .
```

W powyższym wywołaniu kropka jest skrótem domyślnego protokołu autoryzacji, tzw. MIT-MAGIC-COOKIE-1

Oba mechanizmy kontroli dostępu do serwera współpracują w ten sposób, że kody autoryzacji `xauth` są sprawdzane tylko wtedy, gdy dany klient nie ma zezwolenia na wyświetlanie `xhost`.

## Tunelowanie komunikacji X Window przez połączenia ssh

W wielu sytuacjach połączenie klienta z serwerem X Window (port TCP 6000 dla serwera :0, 6001 dla :1, itd.) z różnych względów nie jest możliwe. Na przykład: serwer może znajdować się w sieci lokalnej za **zaporą ogniową** (ang. *firewall*), która nie przepuszcza do wewnątrz sieci żadnych połączeń, albo może znajdować się w sieci z filtrowanymi adresami, gdzie komputery w sieci wewnętrznej mają prywatne adresy w ogóle niewidoczne w Internecie.

W takich sytuacjach można często **przetunelować** połączenia X Window przez połączenie ssh. Jeżeli możemy wykonać połączenie ssh z komputera, posiadającego serwer X Window, w takiej sieci lokalnej, do jakiegoś serwera internetowego, to program ssh może utworzyć tunel przekazujący pakiety na wskazane porty w ramach swojego połączenia. W ten sposób możemy utworzyć na zdalnym serwerze internetowym wirtualny serwer X Window, o adresie wewnątrz tego serwera. Jednak komunikacja kierowana przez klienty X Window na ten adres będzie przekazywana do bezpośrednio niedostępnego serwera X Window.

## Komunikacja ze zdalnym zarządcą ekranów xdm

Praca w sieci zwykle polega na podłączaniu się do innych komputerów (np. ssh) i uruchamianiu na nich klientów z wyświetlaniem na lokalny serwer. W efekcie mamy środowisko graficzne lokalnego komputera, i wybrane zdalne klienty.

W systemie X Window istnieje mechanizm, zwany **zarządcą ekranów** (*display manager*), służący do automatycznego uruchamiania serwerów X na grupie ekranów graficznych, będących pod jego kontrolą. Zarządca ekranów obsługuje też proces logowania się i tworzenia sesji na serwerze X, ponieważ użytkownik nie ma możliwości rozpoczęcia pracy na uruchomionym przez system serwerze.

Zarządca ekranów może też być skonfigurowany do wysyłania klienta logowania na serwer X Window, który o to poprosi (protokołem XDMCP). Pozwala to uruchomić serwer, który zamiast wyświetlać okna lokalnych klientów, inicjuje logowanie się użytkownika do zdalnego komputera w trybie graficznym.

Poniższe wywołanie uruchamia na lokalnym komputerze drugi serwer (:1), który poprosi zdalnego zarządcę ekranów o przysłanie klienta logowania:

```
X -query -once adres.ip.zarzadcy.ekranow :1
```

## Serwery-klienty X Window

Możliwość uruchamiania niestandardowych konfiguracji serwerów X Window jest atrakcyjna, jednak utrudniona przez konieczność obsługi surowego ekranu graficznego. Istnieją programy, które są klientami, tworzą okienko graficzne, a następnie inicjują wirtualny serwer X Window. Serwer jest wirtualny w takim sensie, że nie obsługuje sprzętu wyświetlającego, karty graficznej, klawiatury ani myszy, tylko jako ekran wykorzystuje swoje okienko, a z klawiaturą i myszą komunikuje się jak zwykły klient.

Natomiast od strony sieciowej taki program jest normalnym serwerem, i może zarówno przyjmować żądania wyświetlania od dowolnych klientów, jak komunikować się z zarządcami ekranów protokołem XDMCP.

Przykładami takich serwerów-klientów są Xnest i Xephyr. Uruchamiając je podajemy zarówno argumenty charakterystyczne dla klienta jak i serwera, np.:

```
Xephyr -screen 1024x768 -display :0 -query adres.ip.zarzadcy.ekranow :1
```

Powyższe wywołanie tworzy okienko klienta o zadanej geometrii na serwerze :0, oraz inicjuje serwer :1 który łączy się ze zdalnym zarządcą ekranów.

# Zasoby

Zasoby (ang. *resources*) są zmiennymi, określonymi dla konkretnych aplikacji (klientów), które pozwalają ustawić na stałe pewne opcjonalne parametry klientów, bez ich każdorazowego jawnego podawania (np. w wierszu wywołania). Przykładami zasobów mogą być wielkość okienka edytora, flagi określające zachowanie się klienta, pożądane kolory, itp.

Mechanizm zasobów pełni podobną funkcję do plików konfiguracyjnych klientów, jednak zamiast wpisywać preferencje użytkownika do plików na wszystkich komputerach, z których uruchamiane mają być klienci, zasoby są wpisywane do serwera (dynamicznie), skąd są dostępne dla wszystkich klientów. Przy uruchamianiu serwera lub sesji wartości zasobów są normalnie ładowane do serwera z pliku `.Xresources` lub `.Xdefaults`.

Zasoby określane są w systemie hierarchicznym, gdzie pierwszym elementem hierarchii jest umowna nazwa klienta, a ostatnim nazwa konkretnego zasobu. Możliwe są poziomy pośrednie w określaniu zasobów. Możliwe jest również określanie całych **klas** zasobów obejmujących szereg **instancji** zasobów należących do danej klasy (klasę identyfikuje nazwa rozpoczynająca się wielką literą). Pozwala to np. na łatwe określanie wartości dla wielu zasobów, i od nich pojedynczych wyjątków.

- Przykłady specyfikacji zasobów:

```
emacs.font: -adobe-courier--r---18-----*
emacs.cursorColor: black
emacs.pointerColor: red
emacs.geometry: 80x50
emacs.foreground: light goldenrod yellow
emacs*Background: #5b7686
```

```
Ghostview.pageMedia: A4
```

```
Fig.latexfonts: true
Fig.specialtext: true
```

- Przykłady ładowania zasobów z pliku do serwera:

```
xrdb -load $HOME/.Xresources
```

```
echo 'xterm*scrollBar: True' | xrdb -merge
```



# Czcionki

System X Window ma rozbudowany system czcionek, którego elementami są: system nazw czcionek, którymi mogą posługiwać się klienci, i ścieżka katalogów dyskowych dostępnych dla serwera, gdzie może on odnajdować pliki z konkretnymi czcionkami. Serwer czyta opisy wszystkich czcionek na startcie, ale można zmienić te opisy lub całą ścieżkę poleceniem `xset fp . . .`. Można posługiwać się nazwami czcionek uogólnionymi (z \*) oraz aliasami. Baza aliasów czcionek znajduje się w pliku `/usr/lib/X11/fonts/fonts.dir`

```
xfontsel -sample "Ala ma kota" &
xfontsel -sample "kość chrzęść woń toń śniedz świtez trójkąt wąwóz" &

xfd -fn -dt-application-bold-r-normal-serif-18-180-75-75-m-110-iso8859-2

xlsfonts -fn '*-iso8859-2'

mkdir ~/FONTS
echo 'polska -dt-*-bold-r-*-18-*-m-*-iso8859-2' > ~/FONTS/fonts.alias
xset fp+ ~/FONTS/
xset fp rehash
xterm -fn polska &
```

- 1= foundry
- 2 = font family
- 3 = weight
- 4 = slant
- 5 = set width
- 6 = pixels
- 7 = points (in tenths)
- 8 = horizontal resolution in d.p.i.
- 9 = vertical resolution in d.p.i.
- 10 = spacing
- 11 = average width (in tenths of a pixel)
- 12 = character set

## Kolory

Kolory, podobnie jak czcionki, są elementami systemu pod nadzorem serwera, do których klienci mogą odwoływać się w swoich żądaniach wyświetlenia. Specyfikacja koloru jest możliwa numerycznie, np. w systemie RGB, albo przez słowne określenia kolorów, zdefiniowane w bazie danych kolorów w pliku `/usr/lib/X11/rgb.txt` (na Solarisie: `/usr/openwin/lib/X11/rgb.txt`). Baza ta zawiera takie określenia jak: „lavender blush”, „pale violet red”, albo „deep sky blue”.

Można również określać kolory w systemie RGB cyframi szesnastkowymi używając od jednej do czterech cyfr na kolor, np. czerwony kolor to `#F00` jak również `#F00000000000` (ten sam czerwony) oraz `#FF0000` („bardziej” czerwony). Nowszy model specyfikacji kolorów pozwala określać je w różnych **przestrzeniach kolorów** używając notacji typu: `rgb:ffff/0/0`.

Najprostszy sposób posługiwania się kolorami jest przez opcje klienta: `-bg` (background) `-fg` (foreground) `-bd` (border) albo przez ustawienia jego zasobów.

```
xterm -bg 'light blue' -fg '#fff' -bd plum -cr rgb:f/f/0
```

Pomimo tego wirtualnego bogactwa kolorów, istnieją serwery X Window z ograniczoną liczbą kolorów możliwych do wyświetlenia w danym momencie, z tzw. mapą kolorów. Nie mogą one zrealizować wszystkich żądań klientów o wyświetlanie kolorów. Mogą również zmieniać mapy kolorów przy zmianie aktywnego klienta. Polecenie `xdpyinfo` podaje informacje o możliwościach wyświetlania serwera.