

Optymalizacja programów Open-Source

Inne zagadnienia

Krzysztof Lichota
lichota@mimuw.edu.pl

„Oszukiwanie” użytkownika

„Oszukiwanie” użytkownika

- Jeśli nie jesteśmy w stanie wykonać jakiejś czynności w programie szybko, możemy sprawić, że użytkownikowi będzie się wydawało, że czynność zostaje wykonana szybko albo że w ogóle coś się dzieje
- Można wykorzystać specyficzne cechy ludzkiego sposobu myślenia, by użytkownikowi nie wydawało się, że program nic nie robi, a on czeka
- Najprostsze spostrzeżenie – jak nam się nudzi, to czas płynie wolno

Informowanie użytkownika

- Użytkownik ma tendencję do myślenia, że jak coś się dzieje w interfejsie, to coś się dzieje w programie („kartka lata między folderami, to informacja jest przesyłana”)
- Jak użytkownik nic nie widzi na ekranie, to zaczyna się nudzić i wydaje mu się, że czas płynie wolno – dodatkowo nie wie, czy jego polecenie zostało wykonane, więc zaczyna się irytować

Informowanie użytkownika (2)

- Wnioski:
 - Użytkownikowi trzeba pokazać coś tak szybko, jak to możliwe, nawet jeśli nie jest w pełni użyteczne (vide uruchomienie Windows XP)
 - Jeśli uruchamianie programu trwa długo, to należy pokazać użytkownikowi „splash screen”
 - Jeśli operacja w programie trwa długo, to należy wyświetlić pasek postępu lub wykonywać operację w tle
 - We wszystkich wypadkach, jeżeli to możliwe, należy informować użytkownika o fazie wykonania operacji (powinna się zmieniać w miarę szybko), jeśli to niemożliwe, to trzeba sprawić wrażenie ruchu (animacja, miganie, itp.)

Częściowe wyniki

- Jeśli mamy możliwość zaprezentowania częściowo przetworzonych wyników (np. częściowo wyświetlonej strony WWW, pierwszych wyników zapytania SQL, itp. to należy to zrobić, nawet jeśli potem trzeba będzie coś poprawić (np. układ strony)
- Należy pamiętać, żeby dać użytkownikowi sygnał, kiedy operacja naprawdę się zakończy, żeby nie czekał na więcej!

Częściowe wyniki (2)

- W przypadku przetwarzania obrazków można zastosować „progresywne” wyświetlanie (najpierw wyświetlamy w niskiej rozdzielczości, potem doczytujemy szczegóły)
- Największe znaczenie ma to, gdy użytkownik na podstawie części wyniku decyduje czy dalej wykonywać operację (np. zastosowanie filtra obrazu)
- Można też zastosować techniki omawiane przy starcie programu – wyświetlić wyniki z cache poprzedniego wykonania, wczytywać z wyprzedzeniem, itp.

Zyskiwanie na czasie przez rozwijanie

- W przypadku gdy operacja zajmuje około sekundy, a nie chcemy by użytkownik miał wrażenie, że jest opóźnienie można zastosować sztuczkę z „rozwijaniem” listy bądź menu, co pozwala uzyskać kilka ułamków sekund na wypełnienie listy lub menu

Optymalizacja zużycia energii

Optymalizacja zużycia energii

- W przypadku urządzeń przenośnych ważnym wyznacznikiem działania programu jest zużycie energii
- Procesory, dyski i inne urządzenia są w stanie automatycznie przełączyć się na niższy pobór energii jeśli nie mają nic do roboty
- Programy można napisać w taki sposób, by współgrały z mechanizmami zarządzania energią, często wyższe zużycie energii jest wynikiem nieprzemyślanego projektowania i ułatwiania sobie implementacji, więc można to poprawić

Ogólne zasady optymalizacji zużycia energii

- Staramy się wykonywać operacje rzadko i przeplatać je maksymalnie długimi okresami bezczynności
- Jeśli coś robimy, to staramy się wykonać maksymalnie dużo w jednym rzucie, w szczególności dotyczy to korzystania z dysku, które wymaga rozkręcenia talerzy
- Zamieniamy przepytывanie (polling) na sterowanie oparte na zdarzeniach (np. ustawiamy timer, czekamy na powiadomienie z programu)

Ogólne zasady optymalizacji zużycia energii (2)

- Zużycie energii może być też pośrednie przez wywołanie innego programu, więc na to też trzeba zwrócić uwagę!
- Unikamy używania bez potrzeby zaawansowanych funkcji sprzętu (np. grafiki 3D, operacji zmiennoprzecinkowych) – procesory mają możliwość wyłączenia części rdzeni
- Absolutnie zabroniony jest spinning (wirowanie)
- Każde zmniejszenie zużycia procesora przez program umożliwia obniżenie częstotliwości CPU, co przekłada się na niższe zużycie energii i ograniczenie chłodzenia (np. wyłączenie wentylatora)

PowerTOP

- PowerTOP to narzędzie stworzone przez Intel'a do analizy zużycia energii przez procesor spowodowanego przez programy
- Pokazuje ile czasu procesor spędza w poszczególnych poziomach zasilania
- Pokazuje liczbę obudzeń na sekundę
- Pokazuje, które programy powodują obudzenie
- Przykłady znalezionych problemów:
<http://www.lesswatts.org/projects/powertop/known.php>

Zmniejszanie głośności

Zmniejszanie głośności

- W niektórych programach (np. odtwarzaczach filmów, programach do prezentacji) kluczowe jest, by komputer zachowywał się cicho
- Dużym źródłem hałasu są wentylatory procesora i karty graficznej – można je wyciszyć stosując techniki ograniczania poboru energii
- Kolejnym źródłem hałasu jest dysk – trzeba wczytać wszystkie potrzebne dane od razu albo wczytywać dane powoli ale jednostajnie, ale unikając zatrzymania i ponownego rozkręcenia talerzy, co jest głośne

Zmniejszanie głośności

- W przypadku dysku należy unikać również przesunięć głowicy (wydaje charakterystyczne tyknięcia), najlepiej zgrupować dane w jednym miejscu na dysku i nie używać małych plików
- Jeśli wczytujemy dane z płyty CD lub DVD, należy ograniczyć maksymalną szybkość wczytywania w napędzie (im szybciej kręci się płyta tym głośniejszy dźwięk), należy dane wczytywać jednostajnie z niską szybkością, żeby unikać również rozkręcania i zatrzymywania płyty

Bibliografia

- <http://developer.gnome.org/projects/gup/hig/2.0/feedback>
- <http://www.lesswatts.org/projects/power-top/>
- <http://www.lesswatts.org/projects/index.php>