

# Organizacja i struktura projektu FreeBSD

Wojciech A. Koszek  
[wkoszek@FreeBSD.org](mailto:wkoszek@FreeBSD.org)  
Częstochowa, 14.11.2008

# Plan prezentacji:

- Wprowadzenie do terminologii
- Krótki wstęp do świata BSD
- Opis “The FreeBSD Project”
- Społeczność, sposób i organizacja pracy
- Nowinki techniczne
- Podsumowanie

# BSD to...

- Berkley Software Distribution
  - niegdyś był to UNIX w formie znanej z książek i pism
- To w BSD powstawały najbardziej znaczące innowacje:
  - prototypy dot. abstrakcji danych, protokół TCP/IP, systemy plików, sposoby schedulingu
- BSD od zawsze było doskonałym przykładem współpracy świata akademickiego i przemysłu:
  - wiele prototypów sponsorowanych przez różnorodne korporacje powstaje w systemach BSD do dziś.

# BSD dziś...

- Świat BSD dziś to nie jak kiedyś jedyna “dystrybucja” oprogramowania. W międzyczasie dokonały się podziały i rezultatem są projekty: **FreeBSD [1993]** (**dowiesz się dalej, dlaczego FreeBSD jest ciekawe**), **NetBSD [1993]** (wieloplatformowy), OpenBSD [1995] (Hm.. Może kiedyś...) DragonFlyBSD (2003)

# Projekt FreeBSD ma na celu:

- rozwój systemu operacyjnego FreeBSD
  - zrzeszanie w jednym miejscu programistów, menadżerów, publicystów i innych osób zainteresowanych rozwojem FreeBSD
- celem wspólnym jest tworzenie i promocja profesjonalnego oprogramowania wysokiej jakości

# Sposób i organizacja pracy

# Ludzie:

- Programiści:
  - ogromna ilość profesjonalistów “z górnej półki”
- Administratorzy
- Wykładowcy
- Hobbyści, pasjonaci, studenci i reszta

# Podział pracy



# “FreeBSD contributor”

- ktoś, kto zgłosił błąd do bazy błędów (Problem Report Database).
  - “Couldn't you just fill a PR...?”
- ktoś, kto podesłał poprawkę do dokumentacji:
  - “teh” -> “the”
- ktoś, kto zaproponował poprawkę:
  - `char *p = “xyz” -> char *p = “MAGIC”`

# “FreeBSD commiter”

- osoba, która posiada dostęp do repozytorium (o tym za chwilę) i może wprowadzać zmiany do kodu źródłowego projektu:
  - wyróżniamy kilka poziomów uprzywilejowania: src, doc, ports
- ma możliwość korzystania z dóbr projektu:
  - maszyn testowych, klastra eksperymentalnego, zniżek na książki, konferencje, koszulki itd...
- zmianie statusu z “FreeBSD contributor” na “FreeBSD commiter” towarzyszy okres, w którym dala osoba posiada swojego “mentora”.

...cd....

- mentor odpowiedzialny jest za zapoznanie nowozaakceptowanego “commitera” ze sposobem pracy, wszystkimi zasobami
- jest po to by odpowiadać na ewentualne pytania
- jest programistą “pierwszego kontaktu”, z którym zaproponowane zmiany w źródłach projektu należy skoordynować

# Grupy trzymające władze:

- Core team:
  - zbiór 9 najbardziej respektowanych commiterów Projektu posiadających decydujący wpływ na kształt i architekturę projektu
- Security team
- Ports management team
- System administrator team
- Documentation team

# System FreeBSD to...

- ogromny projekt, w skład którego wchodzi:
  - kod źródłowy jądra systemu
  - kod źródłowy programów, skryptów oraz narzędzi do zbudowania np.: bootowalnego CD instalacyjnego
  - dokumentacja
  - system pakietów oprogramowania *PORTS*
  - regression tests
- Zarządzanie tak ogromną skarbinicą jest problematyczne

# Systemy kontroli wersji

- CVS
  - od zawsze!
  - historia wielu bardzo znaczących etapów rozwoju systemów operacyjnych trzema jest właśnie tutaj
- Subversion:
  - w tym roku dokonano migracji z CVS do Subversion:
    - długoterminowo ten krok ma usunąć problemy, które obecne są w CVS
- Perforce:
  - komercyjny, doskonały system kontroli wersji
  - używane we FreeBSD od bardzo dawna; wykorzystywany do najbardziej eksperymentalnych zmian

# Sposób wprowadzania zmian:

- Propozycja zmiany w postaci:
  - ustnej sugestii do programisty odpowiedzialnego za kod
  - łata na kod źródłowy
- Peer review:
  - w przypadku posiadania mentora, to on jako pierwszy dokonuje akceptacji/zwrotu zaproponowanej łatki
  - przejrzanie zmiany przez osobę posiadającą największą wiedzę dot. danej partii projektu
    - duża liczba doskonałych programistów

# Sposób wprowadzania zmian, cd:

- Każda zmiana trafia do repozytorium:
  - każdy “commit” widoczny w repozytorium wysyłany jest również na publicznie dostępną listę mailingową
    - każdy może komentować
  - zmiany posiadają dodatkowe nagłówki umieszczone w treści opisu zmiany:
    - Reviewed by:
    - Appoved by:
    - Security:
    - MFC itd..



# Komunikacja i rozwiązywanie konfliktów:

- Listy mailingowe:
  - doskonałe z powodu łatwego dostępu poprzez mail i WWW (świetny interfejs programu Mailman)
  - wiele tematycznych grup
  - wiele grup ogólnego przeznaczenia
- Rozwiązywanie problemów ma miejsce na publicznym forum.

# Aspekty techniczne:

- FreeBSD to głównie kod w języku C, potem C++
  - niskopoziomowy kod jądra w assemblerze
- Kod pisany jest w jednolitym stylu:
  - FreeBSD cechuje dbałość o ten aspekt tworzenia oprogramowania

# Dbłość o infrastrukturę:

- Wszelkie przewidywania sę z góry skazane na niepowodzenia
- Większość rzeczy projektowana jest w sposób "uogólniony", tak by możliwe było wykorzystywanie różnych podsystemów w bardzo uniwersalny sposób.
  - Przykłady: GEOM, Netgraph.

# Podsumowanie:

- FreeBSD == projekt wart Twojej uwagi
- FreeBSD.org <- dobry punkt startu
- lists.FreeBSD.org <- sposób na obserwację tego, co dzieję się we FreeBSD

# Koniec

Zapraszam do zadawania pytań

Wojciech A. Koszek  
<wkoszek@FreeBSD>