

TIC : changement d'échelle et nouveaux défis

Roberto Di Cosmo

Professeur d'Informatique
Université Paris Diderot
40 ans de l'INRIA
Lille, 11 décembre 2007

Révolution?



Révolution?



⋮



⋮



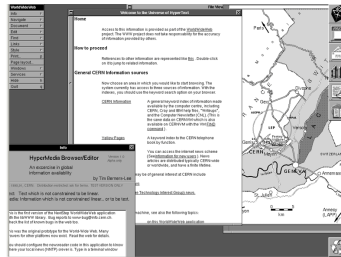
WorldWideWeb, 1989

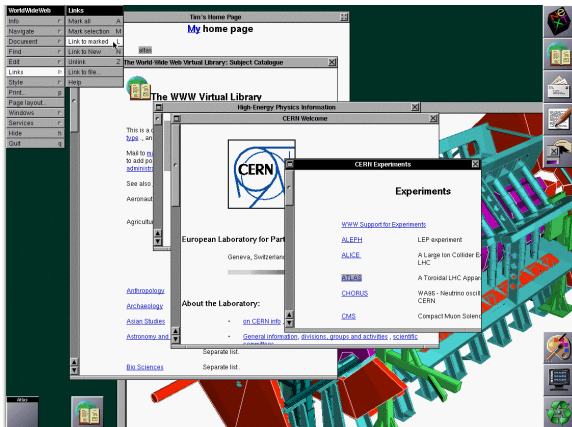
CERN (Suisse), T. Berners-Lee, R. Cailleau sur NeXTStep

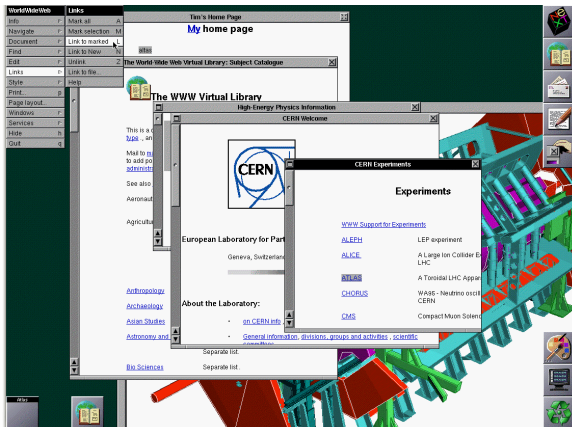
CERN (Suisse), T. Berners-Lee, R. Cailleau sur NeXTStep



CERN (Suisse), T. Berners-Lee, R. Cailleau sur NeXTStep

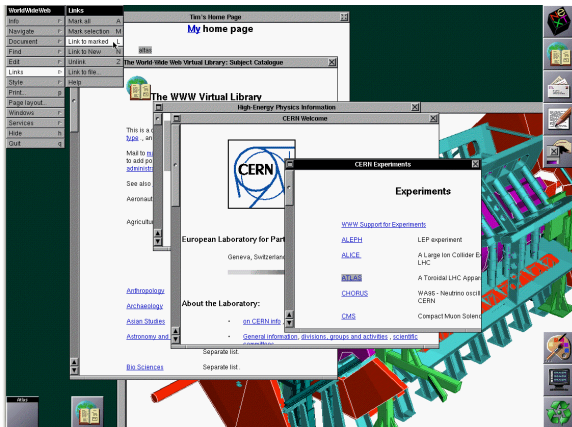






HTTP PUT était déjà prévu...

Web 2.0? Web 1.0!



HTTP PUT était déjà prévu...

Et encore ...

WebCrawler :

1992, US, avec *l'IndexingKit* de NeXTSTEP

Et encore ...

WebCrawler :

1992, US, avec *l'IndexingKit de NeXTSTEP*

Linux :

1990, Linus Torvalds, Finlande

Et encore ...

WebCrawler :

1992, US, avec l'*IndexingKit* de NeXTSTEP

Linux :

1990, Linus Torvalds, Finlande

TIC: le moteur de la croissance est né en Europe

L'essentiel de la technologie *logicielle* était déjà là il y a dix ans!

Et encore ...

WebCrawler :

1992, US, avec l'*IndexingKit* de NeXTSTEP

Linux :

1990, Linus Torvalds, Finlande

TIC: le moteur de la croissance est né en Europe

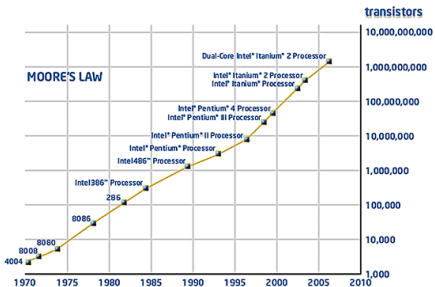
L'essentiel de la technologie *logicielle* était déjà là il y a dix ans!

Mais la carrosserie a été faite ailleurs

on y reviendra

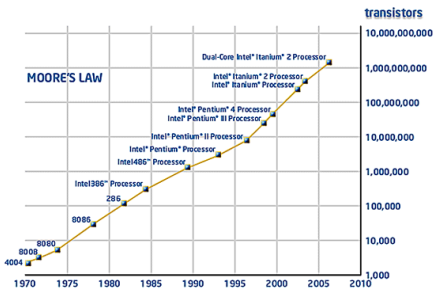
Ce qui a vraiment changé?

Transistors sur un chip



Ce qui a vraiment changé?

Transistors sur un chip

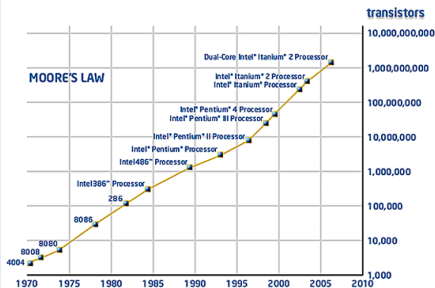


Capacité des disques

Année	2.5in	3.5in
1997	3Gb	10Gb
2007	250Gb	1000Gb

Ce qui a vraiment changé?

Transistors sur un chip



Capacité des disques

Année	2.5in	3.5in
1997	3Gb	10Gb
2007	250Gb	1000Gb

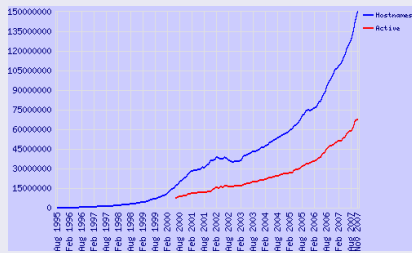
Débit du réseau

Année	ADSL Rx	ADSL Tx
1998	256Kb	128 Kb
2007	28Mb	1 Mb ^a

^aException importante!

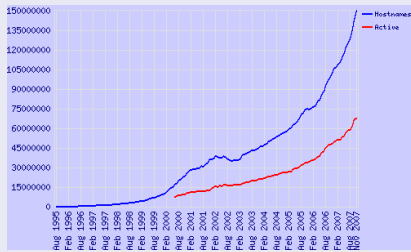
Ce qui a vraiment changé?

Sites web (Netcraft)

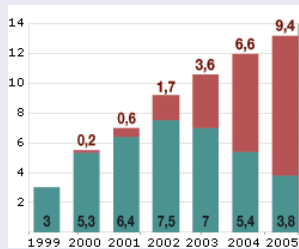


Ce qui a vraiment changé?

Sites web (Netcraft)

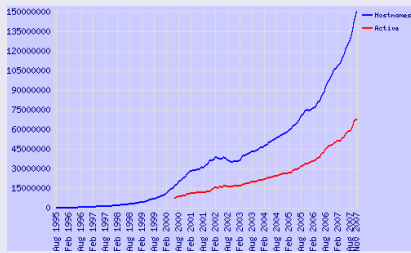


Foyers connectés (JdN)

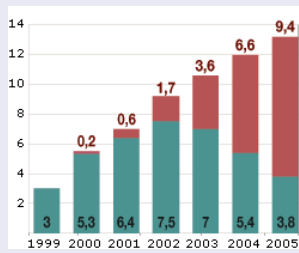


Ce qui a vraiment changé?

Sites web (Netcraft)



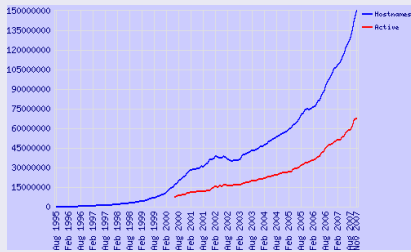
Foyers connectés (JdN)



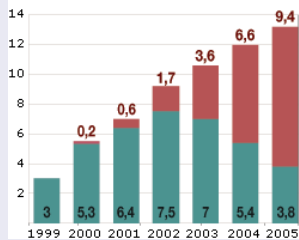
Deux ordres de grandeur en 10 ans!

Ce qui a vraiment changé?

Sites web (Netcraft)



Foyers connectés (JdN)



Deux ordres de grandeur en 10 ans!

Cela pose des défis éthiques, technologiques et organisationnels.

Quelques observations

L'échange *prime* sur la possession:

- peer to peer
- social networks
- logiciel libre
- open access

Quelques observations

L'échange *prime* sur la possession:

- peer to peer
- social networks
- logiciel libre
- open access

Test: combien de jours pouvez-vous résister sans réseau?

L'échange *prime* sur la possession:

- peer to peer
- social networks
- logiciel libre
- open access

Test: combien de jours pouvez-vous résister sans réseau?

L'utilité immédiate *prime* sur la réflexion:

- "polling" du Web, indexation
- plein d'exemples, voir MBA 101

En général, la refonte organisationnelle, économique et culturelle de la société, mais en particulier:

Privacy, anonymat, pseudonimat :

LinkedIn, FaceBook, Google, Flick'r, YouTube, blogs, e-administration etc.: c'est tellement attrayant et pratique qu'on accepte de

- tout dévoiler. . .
- remettre en cause des acquis fondamentaux (vérifiabilité du vote, etc.). . .

En général, la refonte organisationnelle, économique et culturelle de la société, mais en particulier:

Privacy, anonymat, pseudonimat :

LinkedIn, FaceBook, Google, Flick'r, YouTube, blogs, e-administration etc.: c'est tellement attrayant et pratique qu'on accepte de

- tout dévoiler. . .
- remettre en cause des acquis fondamentaux (vérifiabilité du vote, etc.). . .

confiance : à qui faire confiance dans une communauté de millions, voir milliards de personnes?

En général, la refonte organisationnelle, économique et culturelle de la société, mais en particulier:

Privacy, anonymat, pseudonimat :

LinkedIn, FaceBook, Google, Flick'r, YouTube, blogs, e-administration etc.: c'est tellement attrayant et pratique qu'on accepte de

- tout dévoiler. . .
- remettre en cause des acquis fondamentaux (vérifiabilité du vote, etc.). . .

confiance : à qui faire confiance dans une communauté de millions, voir milliards de personnes?

qualité : comment la concilier avec le changement d'échelle?
peut-on mesurer la *qualité* objectivement?
PageRank mesure-t-il la *qualité*?

En général, la refonte organisationnelle, économique et culturelle de la société, mais en particulier:

Privacy, anonymat, pseudonimat :

LinkedIn, FaceBook, Google, Flick'r, YouTube, blogs, e-administration etc.: c'est tellement attrayant et pratique qu'on accepte de

- tout dévoiler. . .
- remettre en cause des acquis fondamentaux (vérifiabilité du vote, etc.). . .

confiance : à qui faire confiance dans une communauté de millions, voir milliards de personnes?

qualité : comment la concilier avec le changement d'échelle?
peut-on mesurer la *qualité* objectivement?
PageRank mesure-t-il la *qualité*?

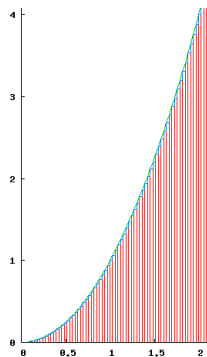
Les informaticiens auraient beaucoup à dire. . .

Juste un exemple: Scigen

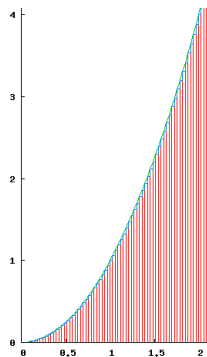
Il nous manque la machine inverse!

Maîtriser la complexité

Maîtriser la complexité

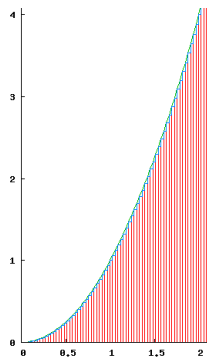


Maîtriser la complexité



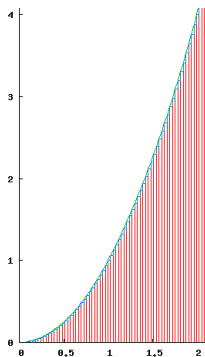
$$A = \int_0^2 x^2 dx$$

Maîtriser la complexité



$$A = \int_0^2 x^2 dx = \frac{8}{3}$$

Maîtriser la complexité

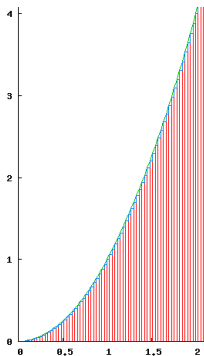


$$A = \int_0^2 x^2 dx = \frac{8}{3}$$

Pascal

```
type pointer = ^cell;  
cell = record  
    Next: pointer; Data: integer;  
end;  
function len(s: pointer): integer;  
var cur: pointer; tmp: integer;  
begin  
    if s = nil then begin len := 0; Exit; end;  
    tmp := 1; cur := s;  
    while (cur.Next <> nil) do  
        begin  
            tmp := tmp + 1; cur := cur.Next;  
        end;  
    len := tmp;  
end;
```

Maîtriser la complexité



$$A = \int_0^2 x^2 dx = \frac{8}{3}$$

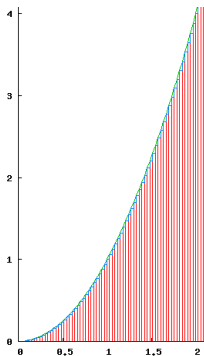
Pascal

```
type pointer = ^cell;  
cell = record  
    Next: pointer; Data: integer;  
end;  
function len(s: pointer): integer;  
var cur: pointer; tmp: integer;  
begin  
    if s = nil then begin len := 0; Exit; end;  
    tmp := 1; cur := s;  
    while (cur^.Next <> nil) do  
        begin  
            tmp := tmp + 1; cur := cur^.Next;  
        end;  
    len := tmp;  
end;
```

OCaml

```
type 'a list = Nil | Cons of 'a * 'a list;;  
let rec len = function Nil -> 0  
                  | Cons (_, r) -> 1+(len r);;
```

Maîtriser la complexité



$$A = \int_0^2 x^2 dx = \frac{8}{3}$$

Pascal

```
type pointer = ^cell;  
cell = record  
    Next: pointer; Data: integer;  
end;  
function len(s: pointer): integer;  
var cur: pointer; tmp: integer;  
begin  
    if s = nil then begin len := 0; Exit; end;  
    tmp := 1; cur := s;  
    while (cur^.Next <> nil) do  
        begin  
            tmp := tmp + 1; cur := cur^.Next;  
        end;  
    len := tmp;  
end;
```

OCaml

```
type 'a list = Nil | Cons of 'a * 'a list;;  
let rec len = function Nil -> 0  
                | Cons (_, r) -> 1+(len r);;
```

La recherche à long terme est essentielle.

Un exemple significatif: le Logiciel Libre

Un exemple significatif: le Logiciel Libre

Gratuit (anglais: free):
logiciel non payant (aujourd'hui)

Un exemple significatif: le Logiciel Libre

Gratuit (anglais: free):
logiciel non payant (aujourd'hui)

Libre (anglais: free):
logiciel avec 4 droits

Un exemple significatif: le Logiciel Libre

- Gratuit** (anglais: free):
logiciel non payant (aujourd'hui)
- Libre** (anglais: free):
logiciel avec 4 droits
- Liberté d'**utiliser** le logiciel

Un exemple significatif: le Logiciel Libre

Gratuit (anglais: free):
logiciel non payant (aujourd'hui)

Libre (anglais: free):
logiciel avec 4 droits

- Liberté d'**utiliser** le logiciel
- Liberté d'**étudier** les sources du logiciel et de l'**adapter** à ses besoins

Un exemple significatif: le Logiciel Libre

Gratuit (anglais: free):

logiciel non payant (aujourd'hui)

Libre (anglais: free):

logiciel avec 4 droits

- Liberté d'**utiliser** le logiciel
- Liberté d'**étudier** les sources du logiciel et de l'**adapter** à ses besoins
- Liberté de **distribuer** des copies

Un exemple significatif: le Logiciel Libre

Gratuit (anglais: free):
logiciel non payant (aujourd'hui)

Libre (anglais: free):
logiciel avec 4 droits

- Liberté d'**utiliser** le logiciel
- Liberté d'**étudier** les sources du logiciel et de l'**adapter** à ses besoins
- Liberté de **distribuer** des copies
- Liberté de **distribuer** les sources (même **modifiées**)

Un exemple significatif: le Logiciel Libre

Gratuit (anglais: free):

logiciel non payant (aujourd'hui)

Libre (anglais: free):

logiciel avec 4 droits

- Liberté d'**utiliser** le logiciel
- Liberté d'**étudier** les sources du logiciel et de l'**adapter** à ses besoins
- Liberté de **distribuer** des copies
- Liberté de **distribuer** les sources (même **modifiées**)

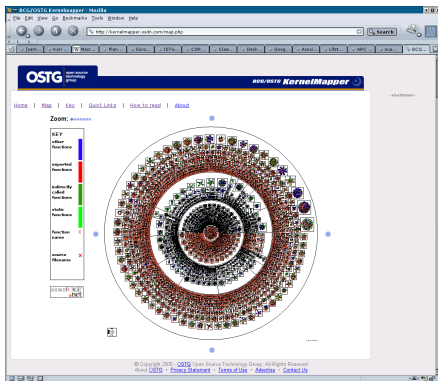
Il y a des **obligations** aussi, qui varient selon la licence: GPL/BSD/Mozilla/X, etc.

Un logiciel libre *n'est pas* un logiciel comme les autres:

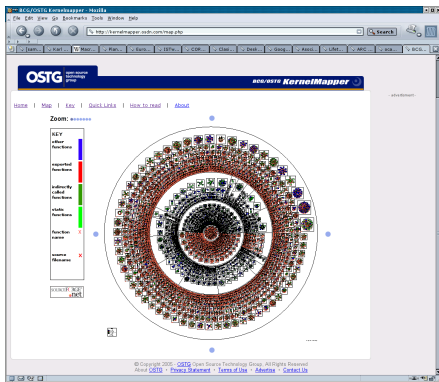
- pas d'architecte unique
- développement distribué
- cycle de développement très rapide
- interdépendances fortes
- disponibilité des sources pour des grandes masses de logiciels

SourceForge.net: [123,736](#) projets, [1,342,153](#) utilisateurs

Des logiciels complexes. . .



Des logiciels complexes...



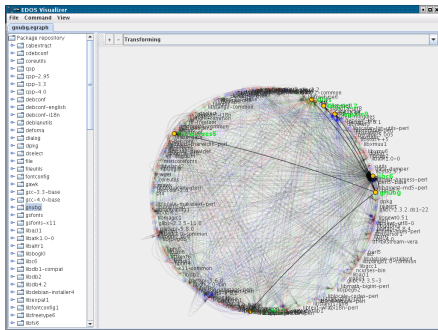
```
linux-2.6.16.20 > sloccount .
```

...

Total Physical Source Lines of Code (SLOC) = 4,827,227

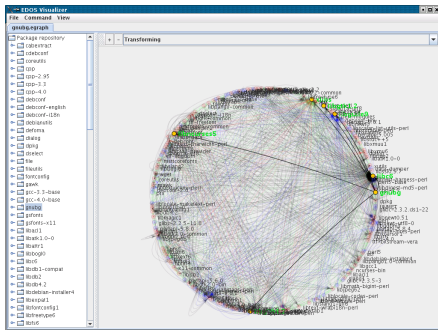
Data generated using David A. Wheeler's 'SLOccount'.

Des interdépendances complexes...



Package: gnubg
Version: 0.14.3+20060923-4
Depends: gnubg-data,
ttf-bitstream-vera, libartsc0
(>= 1.5.0-1), ..., libgl1-mesa-glx
| libgl1, ...
Conflicts: ...

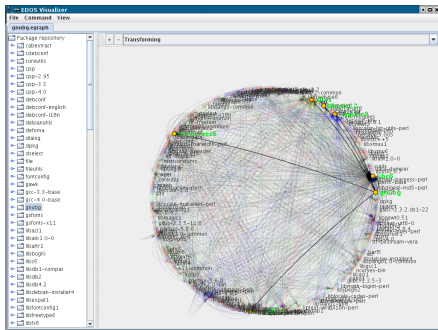
Des interdépendances complexes...



Package: gnubg
Version: 0.14.3+20060923-4
Depends: gnubg-data,
ttf-bitstream-vera, libartsc0
(>= 1.5.0-1), ..., libgl1-mesa-glx
| libgl1, ...
Conflicts: ...

Cela change tous les jours! Comment s'y retrouver?

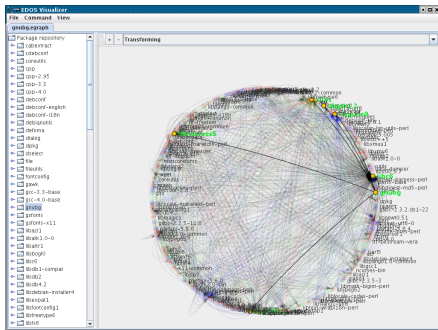
Des interdépendances complexes...



Package: gnubg
Version: 0.14.3+20060923-4
Depends: gnubg-data,
ttf-bitstream-vera, libartsc0
(>= 1.5.0-1), ..., libgl1-mesa-glx
| libgl1, ...
Conflicts: ...

Cela change *tous les jours!* Comment s'y retrouver?
Vous le faite tous les jour. . .

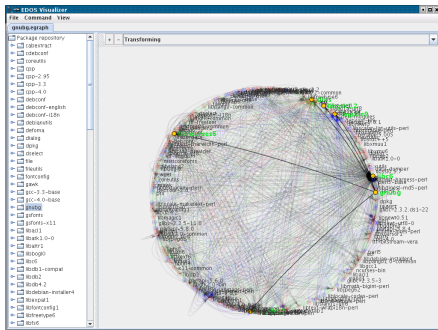
Des interdépendances complexes...



Package: gnuob
Version: 0.14.3+20060923-4
Depends: gnuob-data,
ttf-bitstream-vera, libartsc0
(>= 1.5.0-1), ..., libgl1-mesa-glx
| libgl1, ...
Conflicts: ...

Cela change *tous les jours!* Comment s'y retrouver?
Vous le faite tous les jour...
pourtant ... c'est un problème au moins NP-complet.

Des interdépendances complexes...



Package: gnuobj-graph
Version: 0.14.3+20060923-4
Depends: gnuobj-graph-data,
tft-bitstream-vera, libartsc0
(>= 1.5.0-1), ..., libgl1-mesa-glx
| libgl1, ...
Conflicts: ...

Cela change *tous les jours!* Comment s'y retrouver?
Vous le faites tous les jours...
pourtant... c'est un problème au moins NP-complet.

Cela préfigure les systèmes complexes de demain.

Le logiciel libre: un cas d'école

- existe depuis très longtemps. . .
- a réussi de succès remarquables. . .
- évolue depuis longtemps grâce aux outils collaboratifs. . .

On a beaucoup à apprendre, *si on n'est pas naïfs*. . .

Mettre une licence libre sur un logiciel propriétaire existant

- un logiciel libre sans communauté active a peu de valeur¹
- il s'agit d'un *nouveau* métier

¹à quelques exceptions près. . .

Mettre une licence libre sur un logiciel propriétaire existant

- un logiciel libre sans communauté active a peu de valeur¹
- il s'agit d'un *nouveau* métier

Jouer le free-rider sur un logiciel libre existant

- maintenir des patches coûte cher: les modifications doivent être acceptées par la communauté. . .

¹à quelques exceptions près. . .

Ce qu'on commence à comprendre

Des chercheurs s'y intéressent;

Les analyses de Martin Michlmayr (ex Debian leader)

Cathedral phase

Ce qu'on commence à comprendre

Des chercheurs s'y intéressent;

Les analyses de Martin Michlmayr (ex Debian leader)

Cathedral phase

Transition phase

Ce qu'on commence à comprendre

Des chercheurs s'y intéressent;

Les analyses de Martin Michlmayr (ex Debian leader)

Cathedral phase

Transition phase

Bazaar phase

Ce qu'on commence à comprendre

Des chercheurs s'y intéressent;

Les analyses de Martin Michlmayr (ex Debian leader)

Cathedral phase

Transition phase

Bazaar phase

Original "idea"

Project Author

Core developers

Unix philosophy

Ce qu'on commence à comprendre

Des chercheurs s'y intéressent;

Les analyses de Martin Michlmayr (ex Debian leader)

Cathedral phase

Transition phase

Bazaar phase

Original "idea"

Project Author

Core developers

Unix philosophy

"Interest"

⇒ Prototype ⇒

Modular design

Des chercheurs s'y intéressent;

Les analyses de Martin Michlmayr (ex Debian leader)

Cathedral phase	Transition phase	Bazaar phase
Original "idea"	"Interest"	Distributed development
Project Author	⇒ Prototype	⇒ Community
Core developers	Modular design	Parallel maintenance
Unix philosophy		Peer reviews

Des chercheurs s'y intéressent;

Les analyses de Martin Michlmayr (ex Debian leader)

Cathedral phase	Transition phase	Bazaar phase
Original "idea"	"Interest"	Distributed development
Project Author	⇒ Prototype	⇒ Community
Core developers	Modular design	Parallel maintenance
Unix philosophy		Peer reviews

Et dans tous cela il y a des groupes d'acteurs fort différents.

- enlever les verrous qui ne servent pas (BAB du collaboratif)
- rendre faciles d'usages ceux qui sont utiles (controle d'accès)
- ne pas hésiter à réécrire (refactoring)

Wikipedia :

- modularité extrême d'une encyclopédie
- modularité des communautés
- ouverte à l'origine, émergence du besoin du contrôle

Wikipedia :

- modularité extrême d'une encyclopédie
- modularité des communautés
- ouverte à l'origine, émergence du besoin du contrôle

Linux kernel :

- modularité de l'architecture (plus de la moitié sont des pilotes)
- crise de croissance (contrôle centralisé strict)
- solution via un cercle de lieutenants

Test: comment réalisez-vous vos documents?

Conclusion

“le succès est dans le *design* modulaire”:

Conclusion

“le succès est dans le *design* modulaire”:

- technologique

“le succès est dans le *design* modulaire” :

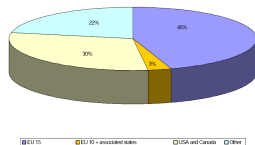
- technologique
- des communautés

“le succès est dans le *design* modulaire” :

- technologique
- des communautés
- économique

On a du chemin à faire!

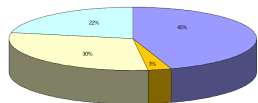
Figure 21: Geographic distribution of leadership in development



Copyright © 2006 MERIT. Source: db.debian.org

On a du chemin à faire!

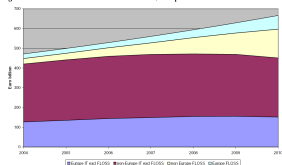
Figure 21: Geographic distribution of leadership in development



■ EU 15 ■ EU 10 + associated states □ USA and Canada □ Other

Copyright © 2006 MERIT. Source: db.debian.org

Figure 47: FLOSS-related and IT services revenue, Europe and world



Copyright © 2006 MERIT. MERIT estimates and projections based on sources including Gartner (IT services market size), IDC (Linux server and PC sales), GIGIX (software investment ratios). Shares add up to the total (Euro 602 billion in 2010).