

Quelques perspectives de l'Informatique

Roberto Di Cosmo

Professeur d'Informatique
Université Paris Diderot

28 Septembre 2007

Une tradition forte d'échanges entre Mathématique et Informatique

Mathématiques

Topologie

Informatique

Sémantique



Une tradition forte d'échanges entre Mathématique et Informatique

Mathématiques

Topologie

- espaces T_0

Informatique

Sémantique

Une tradition forte d'échanges entre Mathématique et Informatique

Mathématiques

Topologie

- espaces T_0

\Rightarrow

Informatique

Sémantique

- modèles du λ -calcul

Une tradition forte d'échanges entre Mathématique et Informatique

Mathématiques

Topologie

- espaces T_0

\Rightarrow

Informatique

Sémantique

- modèles du λ -calcul
- effectivité et calculabilité

Une tradition forte d'échanges entre Mathématique et Informatique

Mathématiques

Topologie

- espaces T_0
- réels effectifs

\Rightarrow

\Leftarrow

Informatique

Sémantique

- modèles du λ -calcul
- effectivité et calculabilité

Une tradition forte d'échanges entre Mathématique et Informatique

Mathématiques

Topologie

- espaces T_0
- réels effectifs

Catégories

- CCC

\Rightarrow

\Leftarrow

Informatique

Sémantique

- modèles du λ -calcul
- effectivité et calculabilité

Typage et calcul

Une tradition forte d'échanges entre Mathématique et Informatique

Mathématiques

Topologie

- espaces T_0
- réels effectifs

Catégories

- CCC

\Rightarrow

\Leftarrow

\Rightarrow

Informatique

Sémantique

- modèles du λ -calcul
- effectivité et calculabilité

Typage et calcul

- modèles de λ^T

Une tradition forte d'échanges entre Mathématique et Informatique

Mathématiques

Topologie

- espaces T_0
- réels effectifs

Catégories

- CCC

\Rightarrow

\Leftarrow

\Rightarrow

Informatique

Sémantique

- modèles du λ -calcul
- effectivité et calculabilité

Typage et calcul

- modèles de λ^T

- étude du typage

Une tradition forte d'échanges entre Mathématique et Informatique

Mathématiques

Topologie

- espaces T_0
- réels effectifs

Catégories

- CCC

Théorie des nombres

- $nfa(\langle N, 0, 1, +, \cdot, \uparrow \rangle)$

\Rightarrow

\Leftarrow

\Rightarrow

\Leftarrow

Informatique

Sémantique

- modèles du λ -calcul
- effectivité et calculabilité

Typage et calcul

- modèles de λ^τ
- étude du typage

Une tradition forte d'échanges entre Mathématique et Informatique

Mathématiques

Topologie

- espaces $T0$
- réels effectifs

Catégories

- CCC

Théorie des nombres

- $nfa(\langle N, 0, 1, +, \cdot, \uparrow \rangle)$

Algèbre

Homotopie

Logique

\Rightarrow

\Leftarrow

\Rightarrow

\Leftarrow

...

Informatique

Sémantique

- modèles du λ -calcul
- effectivité et calculabilité

Typage et calcul

- modèles de λ^τ

- étude du typage

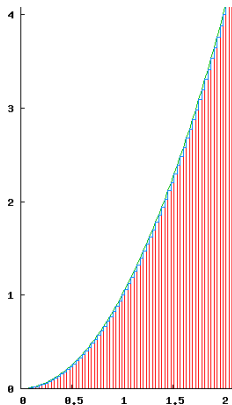
Langages formels

Parallélisme

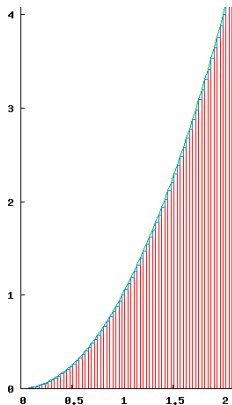
Preuve de programmes

...

Un but commun: maîtriser la complexité

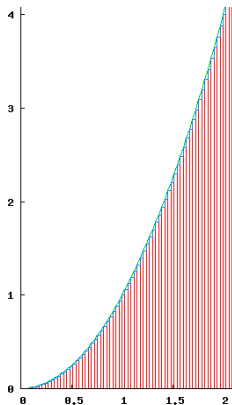


Un but commun: maîtriser la complexité



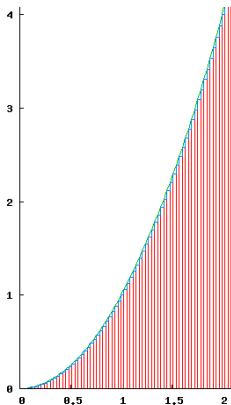
$$A = \int_0^2 x^2 dx$$

Un but commun: maîtriser la complexité



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

Un but commun: maîtriser la complexité

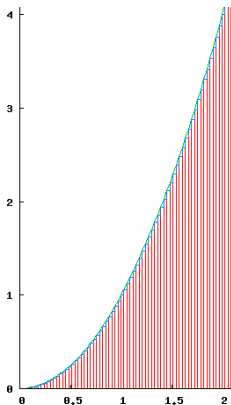


$$A = \int_0^2 x^2 dx = \frac{8}{3}$$

Pascal

```
type PSLLItem = ^TSLLItem;  
    TSLLItem = record  
        Next: pointer;  
        Data: integer;  
    end;  
function len(s: pointer): integer;  
var cur: PSLLItem;  
    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;
```

Un but commun: maîtriser la complexité



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

Pascal

```
type PSLLItem = ^TSLItem;  
    TSLItem = record  
        Next: pointer;  
        Data: integer;  
    end;  
function len(s: pointer): integer;  
var cur: PSLLItem;  
    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);;
```


Un nouveau domaine d'études : le Logiciel Libre

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

Un nouveau domaine d'études : le Logiciel Libre

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

Libre (anglais: free):
logiciel avec 4 droits

Un nouveau domaine d'études : le Logiciel Libre

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

Libre (anglais: free):
logiciel avec 4 droits

- Liberté d'**utiliser** le logiciel

Un nouveau domaine d'études : 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

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 nouveau domaine d'études : 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**)

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**)

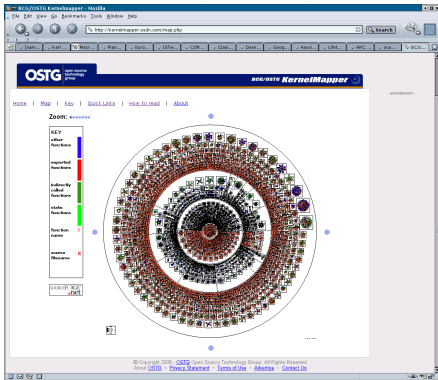
Il y a des **obligations** aussi, qui varient selon la licence: GPL/BSD/Mozilla/X, etc.

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

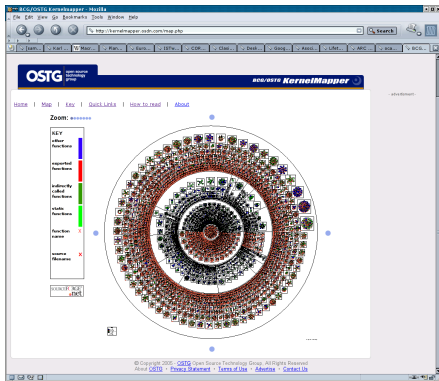
- pas d'architecte unique
- développement distribué
- cycle de développement très rapide
- interdépendances fortes
- disponibilité des sources pour des grandes masses de logiciels

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

Des logiciels complexes. . .



Des logiciels complexes. . .



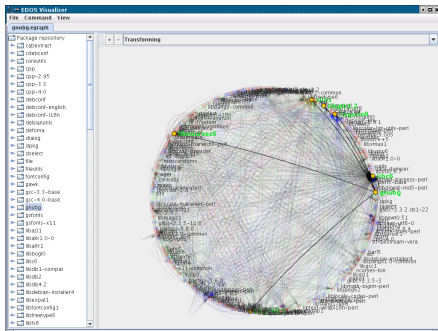
```
linux-2.6.16.20 > sloccount .
```

...

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

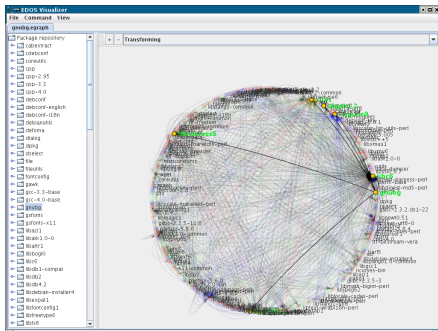
Data generated using David A. Wheeler's 'SLOccount'.

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: ...

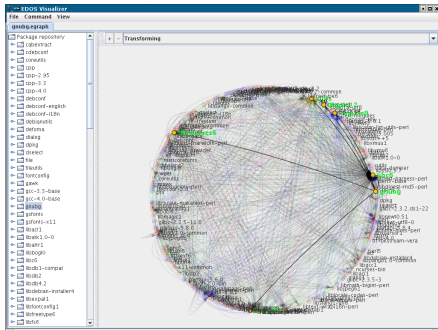
Des interdépendances complexes...



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!

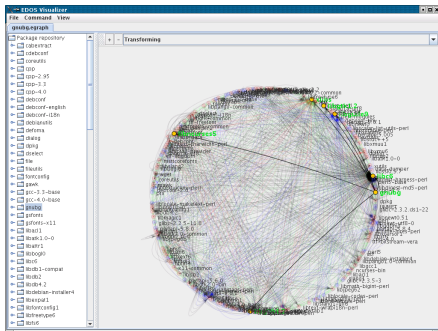
Des interdépendances complexes...



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?

Ces problèmes sont au coeur des projets **EDOS**, et **MANCOOSI**.

Installer libc6 dans:

Package: libc6
Version: 2.2.5-11.8

Package: libc6
Version: 2.3.5-3

Package: libc6
Version: 2.3.2.ds1-22
Depends: libdb1-compat

Package: libdb1-compat
Version: 2.1.3-8
Depends: libc6 (>= 2.3.5-1)

Package: libdb1-compat
Version: 2.1.3-7
Depends: libc6 (>= 2.2.5-13)

Installer libc6 dans:

Package: libc6
Version: 2.2.5-11.8

Package: libc6
Version: 2.3.5-3

Package: libc6
Version: 2.3.2.ds1-22
Depends: libdb1-compat

Package: libdb1-compat
Version: 2.1.3-8
Depends: libc6 (>= 2.3.5-1)

Package: libdb1-compat
Version: 2.1.3-7
Depends: libc6 (>= 2.2.5-13)

peut devenir

Installer libc6 dans:

Package: libc6
Version: 2.2.5-11.8

Package: libc6
Version: 2.3.5-3

Package: libc6
Version: 2.3.2.ds1-22
Depends: libdb1-compat

Package: libdb1-compat
Version: 2.1.3-8
Depends: libc6 (>= 2.3.5-1)

Package: libdb1-compat
Version: 2.1.3-7
Depends: libc6 (>= 2.2.5-13)

peut devenir

```
l2.3.2.ds1-22libc6
^
¬(l2.3.2.ds1-22libc6 ∧ l2.2.5-11.8libc6)
^
¬(l2.3.2.ds1-22libc6 ∧ l2.3.5-3libc6)
^
¬(l2.3.5-3libc6 ∧ l2.2.5-11.8libc6)
^
¬(l2.1.3-7libdb1-compat ∧ l2.1.3-8libdb1-compat)
^
l2.3.2.ds1-22libc6 →
l2.1.3-7libdb1-compat ∨ l2.1.3-8libdb1-compat)
^
l2.1.3-7libdb1-compat →
(l2.3.2.ds1-22libc6 ∨ l2.3.5-3libc6)
^
l2.1.3-8libdb1-compat → l2.3.5-3libc6
```

Installer libc6 dans:

Package: libc6
Version: 2.2.5-11.8

Package: libc6
Version: 2.3.5-3

Package: libc6
Version: 2.3.2.ds1-22
Depends: libdb1-compat

Package: libdb1-compat
Version: 2.1.3-8
Depends: libc6 (>= 2.3.5-1)

Package: libdb1-compat
Version: 2.1.3-7
Depends: libc6 (>= 2.2.5-13)

peut devenir

```
I2.3.2.ds1-22libc6
^
¬(I2.3.2.ds1-22libc6 ∧ I2.2.5-11.8libc6)
^
¬(I2.3.2.ds1-22libc6 ∧ I2.3.5-3libc6)
^
¬(I2.3.5-3libc6 ∧ I2.2.5-11.8libc6)
^
¬(I2.1.3-7libdb1-compat ∧ I2.1.3-8libdb1-compat)
^
I2.3.2.ds1-22libc6 →
(I2.1.3-7libdb1-compat ∨ I2.1.3-8libdb1-compat)
^
I2.1.3-7libdb1-compat →
(I2.3.2.ds1-22libc6 ∨ I2.3.5-3libc6)
^
I2.1.3-8libdb1-compat → I2.3.5-3libc6
```

Mais il faut aller beaucoup plus loin:

- comment trouver les *meilleures* solutions?
- comment garantir la réversibilité des mises à jours?

Installer libc6 dans:

Package: libc6
Version: 2.2.5-11.8

Package: libc6
Version: 2.3.5-3

Package: libc6
Version: 2.3.2.ds1-22
Depends: libdb1-compat

Package: libdb1-compat
Version: 2.1.3-8
Depends: libc6 (>= 2.3.5-1)

Package: libdb1-compat
Version: 2.1.3-7
Depends: libc6 (>= 2.2.5-13)

peut devenir

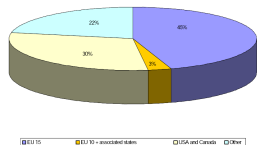
$$\begin{array}{l} I_{libc6}^{2.3.2.ds1-22} \\ \wedge \\ \neg(I_{libc6}^{2.3.2.ds1-22} \wedge I_{libc6}^{2.2.5-11.8}) \\ \wedge \\ \neg(I_{libc6}^{2.3.2.ds1-22} \wedge I_{libc6}^{2.3.5-3}) \\ \wedge \\ \neg(I_{libc6}^{2.3.5-3} \wedge I_{libc6}^{2.2.5-11.8}) \\ \wedge \\ \neg(I_{libdb1-compat}^{2.1.3-7} \wedge I_{libdb1-compat}^{2.1.3-8}) \\ \wedge \\ I_{libc6}^{2.3.2.ds1-22} \rightarrow \\ (I_{libdb1-compat}^{2.1.3-7} \vee I_{libdb1-compat}^{2.1.3-8}) \\ \wedge \\ I_{libdb1-compat}^{2.1.3-7} \rightarrow \\ (I_{libc6}^{2.3.2.ds1-22} \vee I_{libc6}^{2.3.5-3}) \\ \wedge \\ I_{libdb1-compat}^{2.1.3-8} \rightarrow I_{libc6}^{2.3.5-3} \end{array}$$

Mais il faut aller beaucoup plus loin:

- comment trouver les *meilleures* solutions?
- comment garantir la réversibilité des mises à jours?

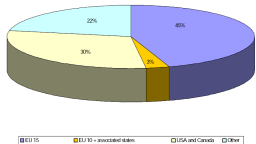
Ces problèmes intéressants, et bien d'autres, ont un lien fort avec la réalité économique...

Figure 21: Geographic distribution of leadership in development



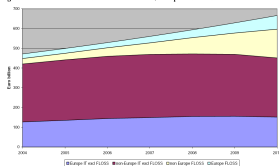
Copyright © 2006 MERIT. Source: db.debian.org

Figure 21: Geographic distribution of leadership in development



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 (€ans 667 billion in 2010).

Nos défis:

scientifique : *résoudre* les problèmes nouveaux

Nos défis:

scientifique : *résoudre* les problèmes nouveaux

pédagogique : *former* les ingénieurs de demain

Nos défis:

scientifique : *résoudre* les problèmes nouveaux

pédagogique : *former* les ingénieurs de demain

économique : *retenir* les succès créateurs de valeur

Nos défis:

scientifique : *résoudre* les problèmes nouveaux

pédagogique : *former* les ingénieurs de demain

économique : *retenir* les succès créateurs de valeur

La Fondation, ainsi que des initiatives comme le GT Ouverture,
ont un rôle majeur à jouer pour relever ces défis.