

ADSILLH
Outils de travail collaboratif

Alexis LAHOUE



ADSILLH 2016 S1

- 1 Introduction
- 2 Outils de Développement
 - Patch
 - Gestion du code source
- 3 Gestion de la Documentation
- 4 Outils de Communication
 - Notions générales
 - Communication synchrone
 - Communication asynchrone
- 5 Forges de développement
- 6 Fonctionnement de certains projets

Introduction

- 1 Introduction
- 2 Outils de Développement
- 3 Gestion de la Documentation
- 4 Outils de Communication
- 5 Forges de développement
- 6 Fonctionnement de certains projets

Alexis Lahouze

Vie professionnelle

- 2001-2003 : IUT Informatique Bayonne
- 2004-2011 : Capgemini : dev Java et adminsys
- 2011-2013 : F-Secure : adminsys
- 2013- : Sysnove : cofondateur



Alexis Lahouze

Vie professionnelle

- 2001-2003 : IUT Informatique Bayonne
- 2004-2011 : Capgemini : dev Java et adminsyst
- 2011-2013 : F-Secure : adminsyst
- 2013- : Sysnove : cofondateur



Alexis Lahouze

Vie professionnelle

- 2001-2003 : IUT Informatique Bayonne
- 2004-2011 : Capgemini : dev Java et adminsyst
- 2011-2013 : F-Secure : adminsyst
- 2013- : Sysnove : cofondateur



Alexis Lahouze

Vie professionnelle

- 2001-2003 : IUT Informatique Bayonne
- 2004-2011 : Capgemini : dev Java et adminsys
- 2011-2013 : F-Secure : adminsys
- 2013- : Sysnove : cofondateur



Alexis Lahouze

Contributions dans le libre

- ABUL
- RMLL
- Relecteur LinuxFR
- root lautre.net
- Diverses contributions à des projets

Alexis Lahouze

Contributions dans le libre

- ABUL
- RMLL
- Relecteur LinuxFR
- root lautre.net
- Diverses contributions à des projets

Alexis Lahouze

Contributions dans le libre

- ABUL
- RMLL
- Relecteur LinuxFR
- root lautre.net
- Diverses contributions à des projets

Alexis Lahouze

Contributions dans le libre

- ABUL
- RMLL
- Relecteur LinuxFR
- root lautre.net
- Diverses contributions à des projets

Alexis Lahouze

Contributions dans le libre

- ABUL
- RMLL
- Relecteur LinuxFR
- root lautre.net
- Diverses contributions à des projets

Objectif

Contribuer

- Comprendre les contraintes du travail en équipe. . .
- . . .et à distance ;
- Avoir un aperçu des outils utilisés dans les projets libres ;
- Savoir partager son travail ;
- Intégrer une communauté dans le cadre du projet tuteuré.

Objectif

Contribuer

- Comprendre les contraintes du travail en équipe...
- ...et à distance ;
- Avoir un aperçu des outils utilisés dans les projets libres ;
- Savoir partager son travail ;
- Intégrer une communauté dans le cadre du projet tuteuré.

Objectif

Contribuer

- Comprendre les contraintes du travail en équipe...
- ...et à distance ;
- Avoir un aperçu des outils utilisés dans les projets libres ;
- Savoir partager son travail ;
- Intégrer une communauté dans le cadre du projet tuteuré.

Objectif

Contribuer

- Comprendre les contraintes du travail en équipe...
- ...et à distance ;
- Avoir un aperçu des outils utilisés dans les projets libres ;
- Savoir partager son travail ;
- Intégrer une communauté dans le cadre du projet tuteuré.

Objectif

Contribuer

- Comprendre les contraintes du travail en équipe. . .
- . . .et à distance ;
- Avoir un aperçu des outils utilisés dans les projets libres ;
- Savoir partager son travail ;
- Intégrer une communauté dans le cadre du projet tuteuré.

Outils de Développement

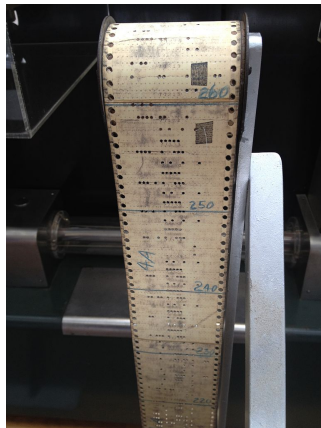
- 1 Introduction
- 2 Outils de Développement
 - Patch
 - Gestion du code source
- 3 Gestion de la Documentation
- 4 Outils de Communication
- 5 Forges de développement
- 6 Fonctionnement de certains projets

Patch

- 1 Introduction
- 2 Outils de Développement
 - Patch
 - Gestion du code source
- 3 Gestion de la Documentation
- 4 Outils de Communication
- 5 Forges de développement
- 6 Fonctionnement de certains projets

Premiers patches

FIGURE – Programme sur bande perforée patchée pour Harvard Mark I.



Source : Wikipedia

Principe de base

- **Généralité :**

- Mécanisme permettant de modifier (correction, amélioration) une ressource.

- **Programmation :**

- Fichier texte décrivant les différences entre deux fichiers texte ;
- Permet de stocker des incréments de modifications d'un fichier (révisions) ;
- Base de la gestion de versions.

Principe de base

- Généralité :
 - Mécanisme permettant de modifier (correction, amélioration) une ressource.
- Programmation :
 - Fichier texte décrivant les différences entre deux fichiers texte ;
 - Permet de stocker des incréments de modifications d'un fichier (révisions) ;
 - Base de la gestion de versions.

Principe de base

- Généralité :
 - Mécanisme permettant de modifier (correction, amélioration) une ressource.
- Programmation :
 - Fichier texte décrivant les différences entre deux fichiers texte ;
 - Permet de stocker des incréments de modifications d'un fichier (révisions) ;
 - Base de la gestion de versions.

Principe de base

- Généralité :
 - Mécanisme permettant de modifier (correction, amélioration) une ressource.
- Programmation :
 - Fichier texte décrivant les différences entre deux fichiers texte ;
 - Permet de stocker des incréments de modifications d'un fichier (révisions) ;
 - Base de la gestion de versions.

Principe de base

- Généralité :
 - Mécanisme permettant de modifier (correction, amélioration) une ressource.
- Programmation :
 - Fichier texte décrivant les différences entre deux fichiers texte ;
 - Permet de stocker des incréments de modifications d'un fichier (révisions) ;
 - Base de la gestion de versions.

Principe de base

- Généralité :
 - Mécanisme permettant de modifier (correction, amélioration) une ressource.
- Programmation :
 - Fichier texte décrivant les différences entre deux fichiers texte ;
 - Permet de stocker des incréments de modifications d'un fichier (révisions) ;
 - Base de la gestion de versions.

Ligne de commande : diff

- Commande de base permettant de comparer deux fichiers :

```
1 # diff original.txt modifie.txt
```

- Series de “morceaux” (hunks)

- Entête : plages de lignes impactées et action : `ww[,yy]aWW[,YY];`

- Possibilité d'avoir de la couleur avec `colordiff`.

Ligne de commande : diff

- Commande de base permettant de comparer deux fichiers :

```
1 # diff original.txt modifie.txt
```

- Series de “morceaux” (hunks)

- Entête : plages de lignes impactées et action : ww[,yy]aWW[,YY];

a add

a change 1,10 and 1,10

- Lignes supprimées préfixées par <;
- Lignes ajoutées préfixées par >;
- Séparateur lorsque qu'il y a les deux (change) : ---;
- Dans le cas de plusieurs fichiers (-r), la commande diff du fichier courant est ajoutée avant le premier morceau : diff -r v1/fichier.txt v2/fichier.txt;

- Possibilité d'avoir de la couleur avec colordiff.

Ligne de commande : diff

- Commande de base permettant de comparer deux fichiers :

```
1 # diff original.txt modifie.txt
```

- Series de “morceaux” (hunks)

- Entête : plages de lignes impactées et action : ww[,yy]aWW[,YY];

a add

d delete

c change (add and delete)

- Lignes supprimées préfixées par <;
- Lignes ajoutées préfixées par >;
- Séparateur lorsque qu'il y a les deux (change) : ---;
- Dans le cas de plusieurs fichiers (-r), la commande diff du fichier courant est ajoutée avant le premier morceau : diff -r v1/fichier.txt v2/fichier.txt;

- Possibilité d'avoir de la couleur avec colordiff.

Ligne de commande : diff

- Commande de base permettant de comparer deux fichiers :

```
1 # diff original.txt modifie.txt
```

- Series de “morceaux” (hunks)

- Entête : plages de lignes impactées et action : ww[,yy]aWW[,YY];

a add

d delete

c change (add and delete)

- Lignes supprimées préfixées par <;
- Lignes ajoutées préfixées par >;
- Séparateur lorsque qu'il y a les deux (change) : ---;
- Dans le cas de plusieurs fichiers (-r), la commande diff du fichier courant est ajoutée avant le premier morceau : diff -r v1/fichier.txt v2/fichier.txt;

- Possibilité d'avoir de la couleur avec colordiff.

Ligne de commande : diff

- Commande de base permettant de comparer deux fichiers :

```
1 # diff original.txt modifie.txt
```

- Series de “morceaux” (hunks)

- Entête : plages de lignes impactées et action : ww[,yy]aWW[,YY];

a add

d delete

c change (add and delete)

- Lignes supprimées préfixées par <;
- Lignes ajoutées préfixées par >;
- Séparateur lorsque qu'il y a les deux (change) : ---;
- Dans le cas de plusieurs fichiers (-r), la commande diff du fichier courant est ajoutée avant le premier morceau : diff -r v1/fichier.txt v2/fichier.txt;

- Possibilité d'avoir de la couleur avec colordiff.

Ligne de commande : diff

- Commande de base permettant de comparer deux fichiers :

```
1 # diff original.txt modifie.txt
```

- Series de “morceaux” (hunks)

- Entête : plages de lignes impactées et action : ww[,yy]aWW[,YY];

a add

d delete

c change (add and delete)

- Lignes supprimées préfixées par <;
- Lignes ajoutées préfixées par >;
- Séparateur lorsque qu'il y a les deux (change) : ---;
- Dans le cas de plusieurs fichiers (-r), la commande diff du fichier courant est ajoutée avant le premier morceau : diff -r v1/fichier.txt v2/fichier.txt;

- Possibilité d'avoir de la couleur avec colordiff.

Ligne de commande : diff

- Commande de base permettant de comparer deux fichiers :

```
1 # diff original.txt modifie.txt
```

- Series de “morceaux” (hunks)

- Entête : plages de lignes impactées et action : ww[,yy]aWW[,YY];

a add

d delete

c change (add and delete)

- Lignes supprimées préfixées par <;
- Lignes ajoutées préfixées par >;
- Séparateur lorsque qu'il y a les deux (change) : ---;
- Dans le cas de plusieurs fichiers (-r), la commande diff du fichier courant est ajoutée avant le premier morceau : diff -r v1/fichier.txt v2/fichier.txt;

- Possibilité d'avoir de la couleur avec colordiff.

Ligne de commande : diff

- Commande de base permettant de comparer deux fichiers :

```
1 # diff original.txt modifie.txt
```

- Series de “morceaux” (hunks)

- Entête : plages de lignes impactées et action : ww[,yy]aWW[,YY];

a add

d delete

c change (add and delete)

- Lignes supprimées préfixées par < ;
- Lignes ajoutées préfixées par > ;
- Séparateur lorsque qu'il y a les deux (change) : --- ;
- Dans le cas de plusieurs fichiers (-r), la commande diff du fichier courant est ajoutée avant le premier morceau : diff -r v1/fichier.txt v2/fichier.txt ;

- Possibilité d'avoir de la couleur avec colordiff.

Ligne de commande : diff

- Commande de base permettant de comparer deux fichiers :

```
1 # diff original.txt modifie.txt
```

- Series de “morceaux” (hunks)

- Entête : plages de lignes impactées et action : ww[,yy]aWW[,YY];

a add

d delete

c change (add and delete)

- Lignes supprimées préfixées par < ;

- Lignes ajoutées préfixées par > ;

- Séparateur lorsque qu'il y a les deux (change) : --- ;

- Dans le cas de plusieurs fichiers (-r), la commande diff du fichier courant est ajoutée avant le premier morceau : diff -r v1/fichier.txt v2/fichier.txt ;

- Possibilité d'avoir de la couleur avec colordiff.

Ligne de commande : diff

- Commande de base permettant de comparer deux fichiers :

```
1 # diff original.txt modifie.txt
```

- Series de “morceaux” (hunks)

- Entête : plages de lignes impactées et action : ww[,yy]aWW[,YY];

a add

d delete

c change (add and delete)

- Lignes supprimées préfixées par <;

- Lignes ajoutées préfixées par >;

- Séparateur lorsque qu'il y a les deux (change) : ---;

- Dans le cas de plusieurs fichiers (-r), la commande diff du fichier courant est ajoutée avant le premier morceau : diff -r v1/fichier.txt v2/fichier.txt;

- Possibilité d'avoir de la couleur avec colordiff.

Ligne de commande : diff

- Commande de base permettant de comparer deux fichiers :

```
1 # diff original.txt modifie.txt
```

- Series de “morceaux” (hunks)

- Entête : plages de lignes impactées et action : ww[,yy]aWW[,YY];

a add

d delete

c change (add and delete)

- Lignes supprimées préfixées par <;
- Lignes ajoutées préfixées par >;
- Séparateur lorsque qu'il y a les deux (change) : ---;
- Dans le cas de plusieurs fichiers (-r), la commande diff du fichier courant est ajoutée avant le premier morceau : diff -r v1/fichier.txt v2/fichier.txt;

- Possibilité d'avoir de la couleur avec colordiff.

Ligne de commande : diff

- Commande de base permettant de comparer deux fichiers :

```
1 # diff original.txt modifie.txt
```

- Series de “morceaux” (hunks)

- Entête : plages de lignes impactées et action : ww[,yy]aWW[,YY];

a add

d delete

c change (add and delete)

- Lignes supprimées préfixées par <;
- Lignes ajoutées préfixées par >;
- Séparateur lorsque qu'il y a les deux (change) : ---;
- Dans le cas de plusieurs fichiers (-r), la commande diff du fichier courant est ajoutée avant le premier morceau : diff -r v1/fichier.txt v2/fichier.txt;

- Possibilité d'avoir de la couleur avec colordiff.

Ligne de commande : diff

- Commande de base permettant de comparer deux fichiers :

```
1 # diff original.txt modifie.txt
```

- Series de “morceaux” (hunks)

- Entête : plages de lignes impactées et action : ww[,yy]aWW[,YY];

a add

d delete

c change (add and delete)

- Lignes supprimées préfixées par <;
- Lignes ajoutées préfixées par >;
- Séparateur lorsque qu'il y a les deux (change) : ---;
- Dans le cas de plusieurs fichiers (-r), la commande diff du fichier courant est ajoutée avant le premier morceau : diff -r v1/fichier.txt v2/fichier.txt;

- Possibilité d'avoir de la couleur avec colordiff.

Sortie normale de diff

original.txt

```

1 This part of the
2 document has stayed the
3 same from version to
4 version. It shouldn't
5 be shown if it doesn't
6 change. Otherwise, that
7 would not be helping to
8 compress the size of the
9 changes.
10
11 This paragraph contains
12 text that is outdated.
13 It will be deleted in the
14 near future.
15
16 It is important to spell
17 check this dokument. On
18 the other hand, a
19 misspelled word isn't
20 the end of the world.
21 Nothing in the rest of
22 this paragraph needs to
23 be changed. Things can
24 be added after it.
```

modifie.txt

```

1 This is an important
2 notice! It should
3 therefore be located at
4 the beginning of this
5 document!
6
7 This part of the
8 document has stayed the
9 same from version to
10 version. It shouldn't
11 be shown if it doesn't
12 change. Otherwise, that
13 would not be helping to
14 compress anything.
15
16 It is important to spell
17 check this document. On
18 the other hand, a
19 misspelled word isn't
20 the end of the world.
21 Nothing in the rest of
22 this paragraph needs to
23 be changed. Things can
24 be added after it.
25
26 This paragraph contains
27 important new additions
28 to this document.
```

diff original.txt modifie.txt

```

1 0a1,6
2 > This is an important
3 > notice! It should
4 > therefore be located at
5 > the beginning of this
6 > document!
7 >
8 8,14c14
9 < compress the size of the
10 < changes.
11 <
12 < This paragraph contains
13 < text that is outdated.
14 < It will be deleted in the
15 < near future.
16 ---
17 > compress anything.
18 17c17
19 < check this dokument. On
20 ---
21 > check this document. On
22 24a25,28
23 >
24 > This paragraph contains
25 > important new additions
26 > to this document.
```


Script ed

Format interprétable (presque) directement par la commande ed :

```
1  # printf "w\\nq\\n" >> v1v2.diff
2  # ed -s original.txt < v1v2.diff
```

Script ed

original.txt

```

1 This part of the
2 document has stayed the
3 same from version to
4 version. It shouldn't
5 be shown if it doesn't
6 change. Otherwise, that
7 would not be helping to
8 compress the size of the
9 changes.
10
11 This paragraph contains
12 text that is outdated.
13 It will be deleted in the
14 near future.
15
16 It is important to spell
17 check this document. On
18 the other hand, a
19 misspelled word isn't
20 the end of the world.
21 Nothing in the rest of
22 this paragraph needs to
23 be changed. Things can
24 be added after it.
```

modifie.txt

```

1 This is an important
2 notice! It should
3 therefore be located at
4 the beginning of this
5 document!
6
7 This part of the
8 document has stayed the
9 same from version to
10 version. It shouldn't
11 be shown if it doesn't
12 change. Otherwise, that
13 would not be helping to
14 compress anything.
15
16 It is important to spell
17 check this document. On
18 the other hand, a
19 misspelled word isn't
20 the end of the world.
21 Nothing in the rest of
22 this paragraph needs to
23 be changed. Things can
24 be added after it.
25
26 This paragraph contains
27 important new additions
28 to this document.
```

diff -e original.txt modifie.txt

```

1 24a
2
3 This paragraph contains
4 important new additions
5 to this document.
6 .
7 17c
8 check this document. On
9 .
10 8,14c
11 compress anything.
12 .
13 0a
14 This is an important
15 notice! It should
16 therefore be located at
17 the beginning of this
18 document!
19
20 .
```

Format contextuel

- BSD 2.8 (Juillet 1981) ;
- Noms de fichiers et timestamp ;
- Lignes contexte :
 - Lignes inchangées ;
 - Préfixées par `_` ;
 - 3 par défaut ;
 - Servent de référence pour valider la consistance du patch ;
 - Permettent parfois de trouver l'emplacement du hunk lorsque les numéros de lignes sont changés dans le fichier sur lequel on applique le patch (fuzzy matching) ;
 - Si lignes de contexte partagées entre deux parties modifiées, ces parties sont regroupées en un seul morceau.
- Pas d'actions explicites, juste les préfixes - (suppression), ! (modification) et + (ajout) ;

Format contextuel

- BSD 2.8 (Juillet 1981) ;
- Noms de fichiers et timestamp ;
- Lignes contexte :
 - Lignes inchangées ;
 - Préfixées par `_` ;
 - 3 par défaut ;
 - Servent de référence pour valider la consistance du patch ;
 - Permettent parfois de trouver l'emplacement du hunk lorsque les numéros de lignes sont changés dans le fichier sur lequel on applique le patch (fuzzy matching) ;
 - Si lignes de contexte partagées entre deux parties modifiées, ces parties sont regroupées en un seul morceau.
- Pas d'actions explicites, juste les préfixes - (suppression), ! (modification) et + (ajout) ;

Format contextuel

- BSD 2.8 (Juillet 1981) ;
- Noms de fichiers et timestamp ;
- Lignes contexte :
 - Lignes inchangées ;
 - Préfixées par `_` ;
 - 3 par défaut ;
 - Servent de référence pour valider la consistance du patch ;
 - Permettent parfois de trouver l'emplacement du hunk lorsque les numéros de lignes sont changés dans le fichier sur lequel on applique le patch (fuzzy matching) ;
 - Si lignes de contexte partagées entre deux parties modifiées, ces parties sont regroupées en un seul morceau.
- Pas d'actions explicites, juste les préfixes - (suppression), ! (modification) et + (ajout) ;

Format contextuel

- BSD 2.8 (Juillet 1981) ;
- Noms de fichiers et timestamp ;
- Lignes contexte :
 - Lignes inchangées ;
 - Préfixées par `_` ;
 - 3 par défaut ;
 - Servent de référence pour valider la consistance du patch ;
 - Permettent parfois de trouver l'emplacement du hunk lorsque les numéros de lignes sont changés dans le fichier sur lequel on applique le patch (fuzzy matching) ;
 - Si lignes de contexte partagées entre deux parties modifiées, ces parties sont regroupées en un seul morceau.
- Pas d'actions explicites, juste les préfixes - (suppression), ! (modification) et + (ajout) ;

Format contextuel

- BSD 2.8 (Juillet 1981) ;
- Noms de fichiers et timestamp ;
- Lignes contexte :
 - Lignes inchangées ;
 - Préfixées par `_` ;
 - 3 par défaut ;
 - Servent de référence pour valider la consistance du patch ;
 - Permettent parfois de trouver l'emplacement du hunk lorsque les numéros de lignes sont changés dans le fichier sur lequel on applique le patch (fuzzy matching) ;
 - Si lignes de contexte partagées entre deux parties modifiées, ces parties sont regroupées en un seul morceau.
- Pas d'actions explicites, juste les préfixes - (suppression), ! (modification) et + (ajout) ;

Format contextuel

- BSD 2.8 (Juillet 1981) ;
- Noms de fichiers et timestamp ;
- Lignes contexte :
 - Lignes inchangées ;
 - Préfixées par `_` ;
 - 3 par défaut ;
 - Servent de référence pour valider la consistance du patch ;
 - Permettent parfois de trouver l'emplacement du hunk lorsque les numéros de lignes sont changés dans le fichier sur lequel on applique le patch (fuzzy matching) ;
 - Si lignes de contexte partagées entre deux parties modifiées, ces parties sont regroupées en un seul morceau.
- Pas d'actions explicites, juste les préfixes - (suppression), ! (modification) et + (ajout) ;

Format contextuel

- BSD 2.8 (Juillet 1981) ;
- Noms de fichiers et timestamp ;
- Lignes contexte :
 - Lignes inchangées ;
 - Préfixées par `_` ;
 - 3 par défaut ;
 - Servent de référence pour valider la consistance du patch ;
 - Permettent parfois de trouver l'emplacement du hunk lorsque les numéros de lignes sont changés dans le fichier sur lequel on applique le patch (fuzzy matching) ;
 - Si lignes de contexte partagées entre deux parties modifiées, ces parties sont regroupées en un seul morceau.
- Pas d'actions explicites, juste les préfixes - (suppression), ! (modification) et + (ajout) ;

Format contextuel

- BSD 2.8 (Juillet 1981) ;
- Noms de fichiers et timestamp ;
- Lignes contexte :
 - Lignes inchangées ;
 - Préfixées par `_` ;
 - 3 par défaut ;
 - Servent de référence pour valider la consistance du patch ;
 - Permettent parfois de trouver l'emplacement du hunk lorsque les numéros de lignes sont changés dans le fichier sur lequel on applique le patch (fuzzy matching) ;
 - Si lignes de contexte partagées entre deux parties modifiées, ces parties sont regroupées en un seul morceau.
- Pas d'actions explicites, juste les préfixes - (suppression), ! (modification) et + (ajout) ;

Format contextuel

- BSD 2.8 (Juillet 1981) ;
- Noms de fichiers et timestamp ;
- Lignes contexte :
 - Lignes inchangées ;
 - Préfixées par `_` ;
 - 3 par défaut ;
 - Servent de référence pour valider la consistance du patch ;
 - Permettent parfois de trouver l'emplacement du hunk lorsque les numéros de lignes sont changés dans le fichier sur lequel on applique le patch (fuzzy matching) ;
 - Si lignes de contexte partagées entre deux parties modifiées, ces parties sont regroupées en un seul morceau.
- Pas d'actions explicites, juste les préfixes - (suppression), ! (modification) et + (ajout) ;

Format contextuel

- BSD 2.8 (Juillet 1981) ;
- Noms de fichiers et timestamp ;
- Lignes contexte :
 - Lignes inchangées ;
 - Préfixées par `_` ;
 - 3 par défaut ;
 - Servent de référence pour valider la consistance du patch ;
 - Permettent parfois de trouver l'emplacement du hunk lorsque les numéros de lignes sont changés dans le fichier sur lequel on applique le patch (fuzzy matching) ;
 - Si lignes de contexte partagées entre deux parties modifiées, ces parties sont regroupées en un seul morceau.
- Pas d'actions explicites, juste les préfixes - (suppression), ! (modification) et + (ajout) ;

Format contextuel

Format d'un morceau

- **Début : *****;**
 - Plage de lignes concernées fichier source, encadrée par *** ;
 - Contexte du début du morceau source, si pas au début du fichier ;
 - Lignes supprimées (préfixées par -), ou modifiées (préfixées par !) ;
 - Contexte de fin du morceau source, si pas en fin de fichier ;
 - Plage de lignes concernées fichier cible, encadrée par --- ;
 - Contexte du début du morceau cible, si pas au début du fichier ;
 - Lignes ajoutées (préfixées par +), ou modifiées (préfixées par !) ;
 - Contexte de fin du morceau cible, si pas en fin de fichier ;

Format contextuel

Format d'un morceau

- Début : *****;
- Plage de lignes concernées fichier source, encadrée par *** ;
 - Contexte du début du morceau source, si pas au début du fichier ;
 - Lignes supprimées (préfixées par -), ou modifiées (préfixées par !) ;
 - Contexte de fin du morceau source, si pas en fin de fichier ;
- Plage de lignes concernées fichier cible, encadrée par --- ;
 - Contexte du début du morceau cible, si pas au début du fichier ;
 - Lignes ajoutées (préfixées par +), ou modifiées (préfixées par !) ;
 - Contexte de fin du morceau cible, si pas en fin de fichier ;

Format contextuel

Format d'un morceau

- Début : *****;
- Plage de lignes concernées fichier source, encadrée par ***;
- Contexte du début du morceau source, si pas au début du fichier;
- Lignes supprimées (préfixées par -), ou modifiées (préfixées par !);
- Contexte de fin du morceau source, si pas en fin de fichier;
- Plage de lignes concernées fichier cible, encadrée par ---;
- Contexte du début du morceau cible, si pas au début du fichier;
- Lignes ajoutées (préfixées par +), ou modifiées (préfixées par !);
- Contexte de fin du morceau cible, si pas en fin de fichier;

Format contextuel

Format d'un morceau

- Début : *****;
- Plage de lignes concernées fichier source, encadrée par ***;
- Contexte du début du morceau source, si pas au début du fichier;
- Lignes supprimées (préfixées par -), ou modifiées (préfixées par !);
- Contexte de fin du morceau source, si pas en fin de fichier;
- Plage de lignes concernées fichier cible, encadrée par ---;
- Contexte du début du morceau cible, si pas au début du fichier;
- Lignes ajoutées (préfixées par +), ou modifiées (préfixées par !);
- Contexte de fin du morceau cible, si pas en fin de fichier;

Format contextuel

Format d'un morceau

- Début : *****;
- Plage de lignes concernées fichier source, encadrée par ***;
- Contexte du début du morceau source, si pas au début du fichier;
- Lignes supprimées (préfixées par -), ou modifiées (préfixées par !);
- Contexte de fin du morceau source, si pas en fin de fichier;
- Plage de lignes concernées fichier cible, encadrée par ---;
- Contexte du début du morceau cible, si pas au début du fichier;
- Lignes ajoutées (préfixées par +), ou modifiées (préfixées par !);
- Contexte de fin du morceau cible, si pas en fin de fichier;

Format contextuel

Format d'un morceau

- Début : *****;
- Plage de lignes concernées fichier source, encadrée par ***;
- Contexte du début du morceau source, si pas au début du fichier;
- Lignes supprimées (préfixées par -), ou modifiées (préfixées par !);
- Contexte de fin du morceau source, si pas en fin de fichier;
- Plage de lignes concernées fichier cible, encadrée par ---;
- Contexte du début du morceau cible, si pas au début du fichier;
- Lignes ajoutées (préfixées par +), ou modifiées (préfixées par !);
- Contexte de fin du morceau cible, si pas en fin de fichier;

Format contextuel

Format d'un morceau

- Début : *****;
- Plage de lignes concernées fichier source, encadrée par ***;
- Contexte du début du morceau source, si pas au début du fichier;
- Lignes supprimées (préfixées par -), ou modifiées (préfixées par !);
- Contexte de fin du morceau source, si pas en fin de fichier;
- Plage de lignes concernées fichier cible, encadrée par ---;
- Contexte du début du morceau cible, si pas au début du fichier;
- Lignes ajoutées (préfixées par +), ou modifiées (préfixées par !);
- Contexte de fin du morceau cible, si pas en fin de fichier;

Format contextuel

Format d'un morceau

- Début : *****;
- Plage de lignes concernées fichier source, encadrée par ***;
- Contexte du début du morceau source, si pas au début du fichier;
- Lignes supprimées (préfixées par -), ou modifiées (préfixées par !);
- Contexte de fin du morceau source, si pas en fin de fichier;
- Plage de lignes concernées fichier cible, encadrée par ---;
- Contexte du début du morceau cible, si pas au début du fichier;
- Lignes ajoutées (préfixées par +), ou modifiées (préfixées par !);
- Contexte de fin du morceau cible, si pas en fin de fichier;

Format contextuel

Format d'un morceau

- Début : *****;
- Plage de lignes concernées fichier source, encadrée par ***;
- Contexte du début du morceau source, si pas au début du fichier;
- Lignes supprimées (préfixées par -), ou modifiées (préfixées par !);
- Contexte de fin du morceau source, si pas en fin de fichier;
- Plage de lignes concernées fichier cible, encadrée par ---;
- Contexte du début du morceau cible, si pas au début du fichier;
- Lignes ajoutées (préfixées par +), ou modifiées (préfixées par !);
- Contexte de fin du morceau cible, si pas en fin de fichier;

Format contextuel

original.txt

```

1 This part of the
2 document has stayed the
3 same from version to
4 version. It shouldn't
5 be shown if it doesn't
6 change. Otherwise, that
7 would not be helping to
8 compress the size of the
9 changes.
10
11 This paragraph contains
12 text that is outdated.
13 It will be deleted in the
14 near future.
15
16 It is important to spell
17 check this dokument. On
18 the other hand, a
19 misspelled word isn't
20 the end of the world.
21 Nothing in the rest of
22 this paragraph needs to
23 be changed. Things can
24 be added after it.
```

modifie.txt

```

1 This is an important
2 notice! It should
3 therefore be located at
4 the beginning of this
5 document!
6
7 This part of the
8 document has stayed the
9 same from version to
10 version. It shouldn't
11 be shown if it doesn't
12 change. Otherwise, that
13 would not be helping to
14 compress anything.
15
16 It is important to spell
17 check this document. On
18 the other hand, a
19 misspelled word isn't
20 the end of the world.
21 Nothing in the rest of
22 this paragraph needs to
23 be changed. Things can
24 be added after it.
25
26 This paragraph contains
27 important new additions
28 to this document.
```

diff -c original.txt modifie.txt

```

1 *** original.txt      timestamp
2 --- modifie.txt timestamp
3 *****
4 *** 1,3 ***
5 --- 1,9 ---
6 + This is an important
7 + notice! It should
8 + therefore be located at
9 + the beginning of this
10 + document!
11 +
12 This part of the
13 document has stayed the
14 same from version to
15 *****
16 *** 5,20 ***
17 be shown if it doesn't
18 change. Otherwise, that
19 would not be helping to
20 ! compress the size of the
21 ! changes.
22 !
23 ! This paragraph contains
24 ! text that is outdated.
25 ! It will be deleted in the
26 ! near future.
27
28 It is important to spell
29 ! check this dokument. On
30 the other hand, a
31 misspelled word isn't
32 the end of the world.
33 --- 11,20 ---
```

Format unifié

- Format développé en août 1990, ajouté à GNU diff 1.15 en janvier 1991 ;
- Hérite des ajouts du format contextuel :
 - Informations sur les fichiers ;
 - Lignes de contexte.
- Moins verbeux que le format contextuel car moins de répétitions ;
- Utilisé pour les échanges entre développeurs (notamment l'envoi par email).

Format unifié

- Format développé en août 1990, ajouté à GNU diff 1.15 en janvier 1991 ;
- Hérite des ajouts du format contextuel :
 - Informations sur les fichiers ;
 - Lignes de contexte.
- Moins verbeux que le format contextuel car moins de répétitions ;
- Utilisé pour les échanges entre développeurs (notamment l'envoi par email).

Format unifié

- Format développé en août 1990, ajouté à GNU diff 1.15 en janvier 1991 ;
- Hérite des ajouts du format contextuel :
 - Informations sur les fichiers ;
 - Lignes de contexte.
- Moins verbeux que le format contextuel car moins de répétitions ;
- Utilisé pour les échanges entre développeurs (notamment l'envoi par email).

Format unifié

- Format développé en août 1990, ajouté à GNU diff 1.15 en janvier 1991 ;
- Hérite des ajouts du format contextuel :
 - Informations sur les fichiers ;
 - Lignes de contexte.
- Moins verbeux que le format contextuel car moins de répétitions ;
- Utilisé pour les échanges entre développeurs (notamment l'envoi par email).

Format unifié

- Format développé en août 1990, ajouté à GNU diff 1.15 en janvier 1991 ;
- Hérite des ajouts du format contextuel :
 - Informations sur les fichiers ;
 - Lignes de contexte.
- Moins verbeux que le format contextuel car moins de répétitions ;
- Utilisé pour les échanges entre développeurs (notamment l'envoi par email).

Format unifié

- Format développé en août 1990, ajouté à GNU diff 1.15 en janvier 1991 ;
- Hérite des ajouts du format contextuel :
 - Informations sur les fichiers ;
 - Lignes de contexte.
- Moins verbeux que le format contextuel car moins de répétitions ;
- Utilisé pour les échanges entre développeurs (notamment l'envoi par email).

Format unifié

Entête concernant le fichier impacté

- Fichier source préfixé par --- et timestamp de modification ;
- Fichier cible préfixé par +++ et timestamp de modification.

Format unifié

Entête concernant le fichier impacté

- Fichier source préfixé par --- et timestamp de modification ;
- Fichier cible préfixé par +++ et timestamp de modification.

Format unifié

Format d'un morceau

- **Plages de lignes concernées :**

- Chaque plage contient la ligne de début et la ligne de fin ;
- La plage côté source est préfixée par - ;
- La plage côté cible est préfixée par + ;
- Encadrées par @@ ;
- Optionnellement suivie d'un entête de section.
- @@ -l,s +l,s @@ optional section heading

- Contexte du début du morceau, si pas au début du fichier ;
- Lignes supprimées (préfixées par -) ;
- Lignes ajoutées (préfixées par +) ;
- Lignes modifiées : ajout et suppression ;
- Contexte de fin du morceau, si pas en fin de fichier ;

Format unifié

Format d'un morceau

- Plages de lignes concernées :
 - Chaque plage contient la ligne de début et la ligne de fin ;
 - La plage côté source est préfixée par - ;
 - La plage côté cible est préfixée par + ;
 - Encadrées par @@ ;
 - Optionnellement suivie d'un entête de section.
 - @@ -l,s +l,s @@ optional section heading
- Contexte du début du morceau, si pas au début du fichier ;
- Lignes supprimées (préfixées par -) ;
- Lignes ajoutées (préfixées par +) ;
- Lignes modifiées : ajout et suppression ;
- Contexte de fin du morceau, si pas en fin de fichier ;

Format unifié

Format d'un morceau

- Plages de lignes concernées :
 - Chaque plage contient la ligne de début et la ligne de fin ;
 - La plage côté source est préfixée par - ;
 - La plage côté cible est préfixée par + ;
 - Encadrées par @@ ;
 - Optionnellement suivie d'un entête de section.
 - @@ -l,s +l,s @@ optional section heading
- Contexte du début du morceau, si pas au début du fichier ;
- Lignes supprimées (préfixées par -) ;
- Lignes ajoutées (préfixées par +) ;
- Lignes modifiées : ajout et suppression ;
- Contexte de fin du morceau, si pas en fin de fichier ;

Format unifié

Format d'un morceau

- Plages de lignes concernées :
 - Chaque plage contient la ligne de début et la ligne de fin ;
 - La plage côté source est préfixée par - ;
 - La plage côté cible est préfixée par + ;
 - Encadrées par @@;
 - Optionnellement suivie d'un entête de section.
 - @@ -l,s +l,s @@ optional section heading
- Contexte du début du morceau, si pas au début du fichier ;
- Lignes supprimées (préfixées par -) ;
- Lignes ajoutées (préfixées par +) ;
- Lignes modifiées : ajout et suppression ;
- Contexte de fin du morceau, si pas en fin de fichier ;

Format unifié

Format d'un morceau

- Plages de lignes concernées :
 - Chaque plage contient la ligne de début et la ligne de fin ;
 - La plage côté source est préfixée par - ;
 - La plage côté cible est préfixée par + ;
 - Encadrées par @@ ;
 - Optionnellement suivie d'un entête de section.
 - @@ -l,s +l,s @@ optional section heading
- Contexte du début du morceau, si pas au début du fichier ;
- Lignes supprimées (préfixées par -) ;
- Lignes ajoutées (préfixées par +) ;
- Lignes modifiées : ajout et suppression ;
- Contexte de fin du morceau, si pas en fin de fichier ;

Format unifié

Format d'un morceau

- Plages de lignes concernées :
 - Chaque plage contient la ligne de début et la ligne de fin ;
 - La plage côté source est préfixée par - ;
 - La plage côté cible est préfixée par + ;
 - Encadrées par @@ ;
 - Optionnellement suivie d'un entête de section.
 - @@ -l,s +l,s @@ optional section heading
- Contexte du début du morceau, si pas au début du fichier ;
- Lignes supprimées (préfixées par -) ;
- Lignes ajoutées (préfixées par +) ;
- Lignes modifiées : ajout et suppression ;
- Contexte de fin du morceau, si pas en fin de fichier ;

Format unifié

Format d'un morceau

- Plages de lignes concernées :
 - Chaque plage contient la ligne de début et la ligne de fin ;
 - La plage côté source est préfixée par - ;
 - La plage côté cible est préfixée par + ;
 - Encadrées par @@ ;
 - Optionnellement suivie d'un entête de section.
 - @@ -l,s +l,s @@ optional section heading
- Contexte du début du morceau, si pas au début du fichier ;
- Lignes supprimées (préfixées par -) ;
- Lignes ajoutées (préfixées par +) ;
- Lignes modifiées : ajout et suppression ;
- Contexte de fin du morceau, si pas en fin de fichier ;

Format unifié

Format d'un morceau

- Plages de lignes concernées :
 - Chaque plage contient la ligne de début et la ligne de fin ;
 - La plage côté source est préfixée par - ;
 - La plage côté cible est préfixée par + ;
 - Encadrées par @@ ;
 - Optionnellement suivie d'un entête de section.
 - @@ -l,s +l,s @@ optional section heading
- Contexte du début du morceau, si pas au début du fichier ;
- Lignes supprimées (préfixées par -) ;
- Lignes ajoutées (préfixées par +) ;
- Lignes modifiées : ajout et suppression ;
- Contexte de fin du morceau, si pas en fin de fichier ;

Format unifié

Format d'un morceau

- Plages de lignes concernées :
 - Chaque plage contient la ligne de début et la ligne de fin ;
 - La plage côté source est préfixée par - ;
 - La plage côté cible est préfixée par + ;
 - Encadrées par @@ ;
 - Optionnellement suivie d'un entête de section.
 - @@ -l,s +l,s @@ optional section heading
- Contexte du début du morceau, si pas au début du fichier ;
- Lignes supprimées (préfixées par -) ;
- Lignes ajoutées (préfixées par +) ;
- Lignes modifiées : ajout et suppression ;
- Contexte de fin du morceau, si pas en fin de fichier ;

Format unifié

Format d'un morceau

- Plages de lignes concernées :
 - Chaque plage contient la ligne de début et la ligne de fin ;
 - La plage côté source est préfixée par - ;
 - La plage côté cible est préfixée par + ;
 - Encadrées par @@ ;
 - Optionnellement suivie d'un entête de section.
 - @@ -l,s +l,s @@ optional section heading
- Contexte du début du morceau, si pas au début du fichier ;
- Lignes supprimées (préfixées par -) ;
- Lignes ajoutées (préfixées par +) ;
- Lignes modifiées : ajout et suppression ;
- Contexte de fin du morceau, si pas en fin de fichier ;

Format unifié

Format d'un morceau

- Plages de lignes concernées :
 - Chaque plage contient la ligne de début et la ligne de fin ;
 - La plage côté source est préfixée par - ;
 - La plage côté cible est préfixée par + ;
 - Encadrées par @@ ;
 - Optionnellement suivie d'un entête de section.
 - @@ -l,s +l,s @@ optional section heading
- Contexte du début du morceau, si pas au début du fichier ;
- Lignes supprimées (préfixées par -) ;
- Lignes ajoutées (préfixées par +) ;
- Lignes modifiées : ajout et suppression ;
- Contexte de fin du morceau, si pas en fin de fichier ;

Format unifié

Format d'un morceau

- Plages de lignes concernées :
 - Chaque plage contient la ligne de début et la ligne de fin ;
 - La plage côté source est préfixée par - ;
 - La plage côté cible est préfixée par + ;
 - Encadrées par @@ ;
 - Optionnellement suivie d'un entête de section.
 - @@ -l,s +l,s @@ optional section heading
- Contexte du début du morceau, si pas au début du fichier ;
- Lignes supprimées (préfixées par -) ;
- Lignes ajoutées (préfixées par +) ;
- Lignes modifiées : ajout et suppression ;
- Contexte de fin du morceau, si pas en fin de fichier ;

Format unifié

original.txt

```

1 This part of the
2 document has stayed the
3 same from version to
4 version. It shouldn't
5 be shown if it doesn't
6 change. Otherwise, that
7 would not be helping to
8 compress the size of the
9 changes.
10
11 This paragraph contains
12 text that is outdated.
13 It will be deleted in the
14 near future.
15
16 It is important to spell
17 check this document. On
18 the other hand, a
19 misspelled word isn't
20 the end of the world.
21 Nothing in the rest of
22 this paragraph needs to
23 be changed. Things can
24 be added after it.
```

modifie.txt

```

1 This is an important
2 notice! It should
3 therefore be located at
4 the beginning of this
5 document!
6
7 This part of the
8 document has stayed the
9 same from version to
10 version. It shouldn't
11 be shown if it doesn't
12 change. Otherwise, that
13 would not be helping to
14 compress anything.
15
16 It is important to spell
17 check this document. On
18 the other hand, a
19 misspelled word isn't
20 the end of the world.
21 Nothing in the rest of
22 this paragraph needs to
23 be changed. Things can
24 be added after it.
25
26 This paragraph contains
27 important new additions
28 to this document.
```

diff -u original.txt modifie.txt

```

1 --- original.txt    timestamp
2 +++ modifie.txt    timestamp
3 @@ -1,3 +1,9 @@
4 +This is an important
5 +notice! It should
6 +therefore be located at
7 +the beginning of this
8 +document!
9 +
10 This part of the
11 document has stayed the
12 same from version to
13 @@ -5,16 +11,10 @@
14 be shown if it doesn't
15 change. Otherwise, that
16 would not be helping to
17 -compress the size of the
18 -changes.
19 -
20 -This paragraph contains
21 -text that is outdated.
22 -It will be deleted in the
23 -near future.
24 +compress anything.
25
26 It is important to spell
27 -check this document. On
28 +check this document. On
29 the other hand, a
30 misspelled word isn't
31 the end of the world.
32 @@ -22,3 +22,7 @@
33 this paragraph needs to
```

Application d'un patch

Commande patch

- Première version publié en 1985 :
 - Sur newsgroup `mod.sources` (devenu `comp.sources.unix`);
 - Par Larry Wall (créateur de Perl).
- Version la plus utilisée : GNU.

Application d'un patch

Commande patch

- Première version publié en 1985 :
 - Sur newsgroup `mod.sources` (devenu `comp.sources.unix`);
 - Par Larry Wall (créateur de Perl).
- Version la plus utilisée : GNU.

Application d'un patch

Commande patch

- Première version publié en 1985 :
 - Sur newsgroup `mod.sources` (devenu `comp.sources.unix`);
 - Par Larry Wall (créateur de Perl).
- Version la plus utilisée : GNU.

Application d'un patch

Commande patch

- Première version publié en 1985 :
 - Sur newsgroup `mod.sources` (devenu `comp.sources.unix`);
 - Par Larry Wall (créateur de Perl).
- Version la plus utilisée : GNU.

Commande patch

Invocation

```
1 # patch < v1v2.diff
```

- Essaie de détecter le format parmi normal, script ed, contextuel ou unifié ;
- Se sert des informations de fichiers et de lignes dans le patch ;
- Si le patch n'est pas applicable directement, tente d'ignorer un certain nombre de lignes des contextes (fuzz factor, 2 par défaut) ;
- En cas de non application d'un morceau, il le met dans un fichier de rejet dont le nom est celui du fichier impacté suffixé par .rej.

Commande patch

Invocation

```
1 # patch < v1v2.diff
```

- Essaie de détecter le format parmi normal, script ed, contextuel ou unifié ;
- Se sert des informations de fichiers et de lignes dans le patch ;
- Si le patch n'est pas applicable directement, tente d'ignorer un certain nombre de lignes des contextes (fuzz factor, 2 par défaut) ;
- En cas de non application d'un morceau, il le met dans un fichier de rejet dont le nom est celui du fichier impacté suffixé par .rej.

Commande patch

Invocation

```
1 # patch < viv2.diff
```

- Essaie de détecter le format parmi normal, script ed, contextuel ou unifié ;
- Se sert des informations de fichiers et de lignes dans le patch ;
- Si le patch n'est pas applicable directement, tente d'ignorer un certain nombre de lignes des contextes (fuzz factor, 2 par défaut) ;
- En cas de non application d'un morceau, il le met dans un fichier de rejet dont le nom est celui du fichier impacté suffixé par .rej.

Commande patch

Invocation

```
1 # patch < viv2.diff
```

- Essaie de détecter le format parmi normal, script ed, contextuel ou unifié ;
- Se sert des informations de fichiers et de lignes dans le patch ;
- Si le patch n'est pas applicable directement, tente d'ignorer un certain nombre de lignes des contextes (fuzz factor, 2 par défaut) ;
- En cas de non application d'un morceau, il le met dans un fichier de rejet dont le nom est celui du fichier impacté suffixé par .rej.

Commande patch

Options utiles

- **Forcer le format du patch :**
 - Normal : `-n` ou `--normal` ;
 - Script ed : `-e` ou `--ed` ;
 - Contextuel : `-c` ou `--context` ;
 - Unifié : `-u` ou `--unified`.
- Répertoire de travail : `-d` ou `--directory` ;
- Tronquer le début des chemins : `-pnum` ou `--strip=num` : `patch -p1 < v1v2.diff`
- Application inverse d'un patch (reverse, ou rollback) : `-R` ou `--reverse` :
`patch -R < v1v2.diff` ;
- Suppression des fichiers vides après application du patch : `-E` ou `--remove-empty-files` ;
- Sauvegarde des fichiers originaux : `-b` ou `--backup` ;

Commande patch

Options utiles

- Forcer le format du patch :
 - Normal : `-n` ou `--normal` ;
 - Script ed : `-e` ou `--ed` ;
 - Contextuel : `-c` ou `--context` ;
 - Unifié : `-u` ou `--unified`.
- Répertoire de travail : `-d` ou `--directory` ;
- Tronquer le début des chemins : `-pnum` ou `--strip=num` : `patch -p1 < v1v2.diff`
- Application inverse d'un patch (reverse, ou rollback) : `-R` ou `--reverse` :
`patch -R < v1v2.diff` ;
- Suppression des fichiers vides après application du patch : `-E` ou `--remove-empty-files` ;
- Sauvegarde des fichiers originaux : `-b` ou `--backup` ;

Commande patch

Options utiles

- Forcer le format du patch :
 - Normal : `-n` ou `--normal` ;
 - Script ed : `-e` ou `--ed` ;
 - Contextuel : `-c` ou `--context` ;
 - Unifié : `-u` ou `--unified`.
- Répertoire de travail : `-d` ou `--directory` ;
- Tronquer le début des chemins : `-pnum` ou `--strip=num` : `patch -p1 < v1v2.diff`
- Application inverse d'un patch (reverse, ou rollback) : `-R` ou `--reverse` :
`patch -R < v1v2.diff` ;
- Suppression des fichiers vides après application du patch : `-E` ou `--remove-empty-files` ;
- Sauvegarde des fichiers originaux : `-b` ou `--backup` ;

Commande patch

Options utiles

- Forcer le format du patch :
 - Normal : `-n` ou `--normal` ;
 - Script ed : `-e` ou `--ed` ;
 - Contextuel : `-c` ou `--context` ;
 - Unifié : `-u` ou `--unified`.
- Répertoire de travail : `-d` ou `--directory` ;
- Tronquer le début des chemins : `-pnum` ou `--strip=num` : `patch -p1 < v1v2.diff`
- Application inverse d'un patch (reverse, ou rollback) : `-R` ou `--reverse` :
`patch -R < v1v2.diff` ;
- Suppression des fichiers vides après application du patch : `-E` ou `--remove-empty-files` ;
- Sauvegarde des fichiers originaux : `-b` ou `--backup` ;

Commande patch

Options utiles

- Forcer le format du patch :
 - Normal : `-n` ou `--normal` ;
 - Script ed : `-e` ou `--ed` ;
 - Contextuel : `-c` ou `--context` ;
 - Unifié : `-u` ou `--unified`.
- Répertoire de travail : `-d` ou `--directory` ;
- Tronquer le début des chemins : `-pnum` ou `--strip=num` : `patch -p1 < v1v2.diff`
- Application inverse d'un patch (reverse, ou rollback) : `-R` ou `--reverse` :
`patch -R < v1v2.diff` ;
- Suppression des fichiers vides après application du patch : `-E` ou `--remove-empty-files` ;
- Sauvegarde des fichiers originaux : `-b` ou `--backup` ;

Commande patch

Options utiles

- Forcer le format du patch :
 - Normal : `-n` ou `--normal` ;
 - Script ed : `-e` ou `--ed` ;
 - Contextuel : `-c` ou `--context` ;
 - Unifié : `-u` ou `--unified`.
- Répertoire de travail : `-d` ou `--directory` ;
- Tronquer le début des chemins : `-pnum` ou `--strip=num` : `patch -p1 < v1v2.diff`
- Application inverse d'un patch (reverse, ou rollback) : `-R` ou `--reverse` :
`patch -R < v1v2.diff` ;
- Suppression des fichiers vides après application du patch : `-E` ou `--remove-empty-files` ;
- Sauvegarde des fichiers originaux : `-b` ou `--backup` ;

Commande patch

Options utiles

- Forcer le format du patch :
 - Normal : `-n` ou `--normal` ;
 - Script ed : `-e` ou `--ed` ;
 - Contextuel : `-c` ou `--context` ;
 - Unifié : `-u` ou `--unified`.
- Répertoire de travail : `-d` ou `--directory` ;
- Tronquer le début des chemins : `-pnum` ou `--strip=num` : `patch -p1 < v1v2.diff`
- Application inverse d'un patch (reverse, ou rollback) : `-R` ou `--reverse` :
`patch -R < v1v2.diff` ;
- Suppression des fichiers vides après application du patch : `-E` ou `--remove-empty-files` ;
- Sauvegarde des fichiers originaux : `-b` ou `--backup` ;

Commande patch

Options utiles

- Forcer le format du patch :
 - Normal : `-n` ou `--normal` ;
 - Script ed : `-e` ou `--ed` ;
 - Contextuel : `-c` ou `--context` ;
 - Unifié : `-u` ou `--unified`.
- Répertoire de travail : `-d` ou `--directory` ;
- Tronquer le début des chemins : `-pnum` ou `--strip=num` : `patch -p1 < v1v2.diff`
- Application inverse d'un patch (reverse, ou rollback) : `-R` ou `--reverse` :
`patch -R < v1v2.diff` ;
- Suppression des fichiers vides après application du patch : `-E` ou `--remove-empty-files` ;
- Sauvegarde des fichiers originaux : `-b` ou `--backup` ;

Commande patch

Options utiles

- Forcer le format du patch :
 - Normal : `-n` ou `--normal` ;
 - Script ed : `-e` ou `--ed` ;
 - Contextuel : `-c` ou `--context` ;
 - Unifié : `-u` ou `--unified`.
- Répertoire de travail : `-d` ou `--directory` ;
- Tronquer le début des chemins : `-pnum` ou `--strip=num` : `patch -p1 < v1v2.diff`
- Application inverse d'un patch (reverse, ou rollback) : `-R` ou `--reverse` :
`patch -R < v1v2.diff` ;
- Suppression des fichiers vides après application du patch : `-E` ou `--remove-empty-files` ;
- Sauvegarde des fichiers originaux : `-b` ou `--backup` ;

Commande patch

Options utiles

- Forcer le format du patch :
 - Normal : `-n` ou `--normal` ;
 - Script ed : `-e` ou `--ed` ;
 - Contextuel : `-c` ou `--context` ;
 - Unifié : `-u` ou `--unified`.
- Répertoire de travail : `-d` ou `--directory` ;
- Tronquer le début des chemins : `-pnum` ou `--strip=num` : `patch -p1 < v1v2.diff`
- Application inverse d'un patch (reverse, ou rollback) : `-R` ou `--reverse` :
`patch -R < v1v2.diff` ;
- Suppression des fichiers vides après application du patch : `-E` ou `--remove-empty-files` ;
- Sauvegarde des fichiers originaux : `-b` ou `--backup` ;

Commande patch

Exemples d'invocation

v1v2.diff

```

1 diff -u v1/f1.txt v2/f1.txt
2 --- v1/f1.txt    2016-09-09 16:06:41.632285371 +0200
3 +++ v2/f1.txt    2016-09-09 16:07:12.913211867 +0200
4 @@ -1,3 +1,9 @@
5 +This is an important
6 +notice! It should
7 +therefore be located at
8 +the beginning of this
9 +document!
10 +
11 This part of the
12 document has stayed the
13 same from version to
14 @@ -5,16 +11,10 @@
15 be shown if it doesn't
16 change. Otherwise, that
17 would not be helping to
18 -compress the size of the
19 -changes.
20 -
21 -This paragraph contains
22 -text that is outdated.
23 -It will be deleted in the
24 -near future.
25 +compress anything.
26
27 It is important to spell
28 -check this document. On
29 +check this document. On
30 the other hand, a
31 misspelled word isn't
32 the end of the world.
```

```

1 @@ -22,3 +22,7 @@
2     this paragraph needs to
3     be changed. Things can
4     be added after it.
5 +
6 +This paragraph contains
7 +important new additions
8 +to this document.
```

Commande patch

Exemples d'invocation

Application réussie d'un patch

```
1 # patch --verbose -p1 < v1v2.diff
2 Hmm... Looks like a unified diff to me...
3 The text leading up to this was:
4 -----
5 |diff -du v1/f1.txt v2/f1.txt
6 |--- v1/f1.txt 2016-09-11 16:06:41.632285371 +0200
7 |+++ v2/f1.txt 2016-09-11 16:07:12.913211867 +0200
8 |-----
9 patching file f1.txt
10 Using Plan A...
11 Hunk #1 succeeded at 1.
12 Hunk #2 succeeded at 11.
13 Hunk #3 succeeded at 22.
14 done
```

Application échouée d'un patch

```
1 # patch --verbose -p1 < v1v2.diff
2 Hmm... Looks like a unified diff to me...
3 The text leading up to this was:
4 -----
5 |diff -du v1/f1.txt v2/f1.txt
6 |--- v1/f1.txt 2016-09-11 16:06:41.632285371 +0200
7 |+++ v2/f1.txt 2016-09-11 16:07:12.913211867 +0200
8 |-----
9 patching file f1.txt
10 Using Plan A...
11 Reversed (or previously applied) patch detected! Assume
    -R? [n] n
12 Apply anyway? [n] y
13 Hunk #1 FAILED at 1.
14 Hunk #2 FAILED at 5.
15 Hunk #3 FAILED at 22.
16 3 out of 3 hunks FAILED -- saving rejects to file
    f1.txt.rej
17 done
```

Commande patch

Exemples d'invocation

f1.txt.rej

```

1  --- f1.txt      2016-09-11 16:06:41.632285371 +0200
2  +++ f1.txt      2016-09-11 16:07:12.913211867 +0200
3  @@ -1,3 +1,9 @@
4  +This is an important
5  +notice! It should
6  +therefore be located at
7  +the beginning of this
8  +document!
9  +
10 This part of the
11 document has stayed the
12 same from version to
13 @@ -5,16 +11,10 @@
14 be shown if it doesn't
15 change. Otherwise, that
16 would not be helping to
17 -compress the size of the
18 -changes.
19 -
20 -This paragraph contains
21 -text that is outdated.
22 -It will be deleted in the
23 -near future.
24 +compress anything.
25
26 It is important to spell
27 -check this document. On
28 +check this document. On
29 the other hand, a
30 misspelled word isn't
31 the end of the world.
```

```

1  @@ -22,3 +22,7 @@
2  this paragraph needs to
3  be changed. Things can
4  be added after it.
5  +
6  +This paragraph contains
7  +important new additions
8  +to this document.
```


Commande patch

Exemples d'invocation

```
diff ../v1v2.diff f1.txt.rej
```

```
1 1,3c1,2
2 < diff -du v1/f1.txt v2/f1.txt
3 < --- v1/f1.txt 2016-09-11 16:06:41.632285371 +0200
4 < +++ v2/f1.txt 2016-09-11 16:07:12.913211867 +0200
5 ---
6 > --- f1.txt 2016-09-11 16:06:41.632285371 +0200
7 > +++ f1.txt 2016-09-11 16:07:12.913211867 +0200
```

Autres outils

Merges avec diff3

```
diff3 -m v3/f1.txt v1/f1.txt v2/f1.txt
```

```
1 This is an important
2 notice! It should
3 therefore be located at
4 the beginning of this
5 document!
6
7 This part of the
8 document has stayed the
9 same from version to
10 version. It shouldn't
11 be shown if it doesn't
12 change. Otherwise, that
13 would not be helping to
```

```
1 <<<<<< v3/f1.txt
2 compress the size of the
3 changes.
4
5 This paragraph contains
6 text that has been
7 modified on both sides.
8 It will be deleted in the
9 near future.
10 ||||| v1/f1.txt
11 compress the size of the
12 changes.
13
14 This paragraph contains
15 text that is outdated.
16 It will be deleted in the
17 near future.
18 =====
19 compress anything.
20 >>>>>> v2/f1.txt
```

```
1
2 It is important to spell
3 check this document. On
4 the other hand, a
5 misspelled word isn't
6 the end of the world.
7 Nothing in the rest of
8 this paragraph needs to
9 be changed. Things can
10 be added after it.
11
12 This paragraph contains
13 important new additions
14 to this document.
```

Autres outils

Quilt

- Nommé d'après patchwork quilt (courtepointe en patchwork) ;
- Permet de regrouper des ensembles de patches en un seul ;
- Initialement développé par Andrew Morton, développeur Kernel Linux ;
- https://stuff.mit.edu/afs/athena/system/i386_deb50/os/usr/share/doc/quilt/quilt.html ;
- Très utilisé dans les projets Linux et Debian ;
- <https://wiki.debian.org/UsingQuilt>.

Autres outils

Quilt

- Nommé d'après patchwork quilt (courtepointe en patchwork) ;
- Permet de regrouper des ensembles de patches en un seul ;
- Initialement développé par Andrew Morton, développeur Kernel Linux ;
- https://stuff.mit.edu/afs/athena/system/i386_deb50/os/usr/share/doc/quilt/quilt.html ;
- Très utilisé dans les projets Linux et Debian ;
- <https://wiki.debian.org/UsingQuilt>.

Autres outils

Quilt

- Nommé d'après patchwork quilt (courtepointe en patchwork) ;
- Permet de regrouper des ensembles de patches en un seul ;
- Initialement développé par Andrew Morton, développeur Kernel Linux ;
- https://stuff.mit.edu/afs/athena/system/i386_deb50/os/usr/share/doc/quilt/quilt.html ;
- Très utilisé dans les projets Linux et Debian ;
- <https://wiki.debian.org/UsingQuilt>.

Autres outils

Quilt

- Nommé d'après patchwork quilt (courtepointe en patchwork) ;
- Permet de regrouper des ensembles de patches en un seul ;
- Initialement développé par Andrew Morton, développeur Kernel Linux ;
- <https://stuff.mit.edu/afs/athena/system/i386-deb50/os/usr/share/doc/quilt/quilt.html> ;
- Très utilisé dans les projets Linux et Debian ;
- <https://wiki.debian.org/UsingQuilt>.

Autres outils

Quilt

- Nommé d'après patchwork quilt (courtepointe en patchwork) ;
- Permet de regrouper des ensembles de patches en un seul ;
- Initialement développé par Andrew Morton, développeur Kernel Linux ;
- https://stuff.mit.edu/afs/athena/system/i386_deb50/os/usr/share/doc/quilt/quilt.html ;
- Très utilisé dans les projets Linux et Debian ;
- <https://wiki.debian.org/UsingQuilt>.

Autres outils

Quilt

- Nommé d'après patchwork quilt (courtepointe en patchwork) ;
- Permet de regrouper des ensembles de patches en un seul ;
- Initialement développé par Andrew Morton, développeur Kernel Linux ;
- <https://stuff.mit.edu/afs/athena/system/i386-deb50/os/usr/share/doc/quilt/quilt.html> ;
- Très utilisé dans les projets Linux et Debian ;
- <https://wiki.debian.org/UsingQuilt>.

Gestion du code source

- 1 Introduction
- 2 Outils de Développement
 - Patch
 - **Gestion du code source**
- 3 Gestion de la Documentation
- 4 Outils de Communication
- 5 Forges de développement
- 6 Fonctionnement de certains projets

Systèmes de gestion de versions

- Local :
 - SCCS : remplacé par :
 - RCS : remplacé par :
- Client-serveur :
 - Tous ceux utilisés actuellement.
- Deux modes principaux :
 - Centralisé
 - Décentralisé

Systèmes de gestion de versions

- Local :
 - SCCS : remplacé par :
 - RCS : remplacé par :
- Client-serveur :
 - Tous ceux utilisés actuellement.
- Deux modes principaux :
 - Centralisé
 - Décentralisé

Systèmes de gestion de versions

- Local :
 - SCCS : remplacé par :
 - RCS : remplacé par :
- Client-serveur :
 - Tous ceux utilisés actuellement.
- Deux modes principaux :
 - Centralisé
 - Décentralisé

Systèmes de gestion de versions

- Local :
 - SCCS : remplacé par :
 - RCS : remplacé par :
- Client-serveur :
 - Tous ceux utilisés actuellement.
- Deux modes principaux :
 - Centralisé
 - Décentralisé

Systèmes de gestion de versions

- Local :
 - SCCS : remplacé par :
 - RCS : remplacé par :
- Client-serveur :
 - Tous ceux utilisés actuellement.
- Deux modes principaux :
 - Centralisé
 - Décentralisé

Systèmes de gestion de versions

- Local :
 - SCCS : remplacé par :
 - RCS : remplacé par :
- Client-serveur :
 - Tous ceux utilisés actuellement.
- Deux modes principaux :
 - Centralisé
 - Décentralisé

Systèmes de gestion de versions

- Local :
 - SCCS : remplacé par :
 - RCS : remplacé par :
- Client-serveur :
 - Tous ceux utilisés actuellement.
- Deux modes principaux :
 - Centralisé
 - Décentralisé

Systèmes de gestion de versions

- Local :
 - SCCS : remplacé par :
 - RCS : remplacé par :
- Client-serveur :
 - Tous ceux utilisés actuellement.
- Deux modes principaux :
 - Centralisé
 - Décentralisé

Systèmes de gestion de versions centralisés

- Principe :
 - Chaque modification (commit) est envoyée directement sur le serveur.
- Les plus connus :
 - CVS
 - Subversion
- Inconvénient : Nécessite une connexion pour envoyer ses modifications.

Systèmes de gestion de versions centralisés

- Principe :
 - Chaque modification (commit) est envoyée directement sur le serveur.
- Les plus connus :
 - CVS
 - Subversion
- Inconvénient : Nécessite une connexion pour envoyer ses modifications.

Systèmes de gestion de versions centralisés

- Principe :
 - Chaque modification (commit) est envoyée directement sur le serveur.
- Les plus connus :
 - CVS
 - Subversion
- Inconvénient : Nécessite une connexion pour envoyer ses modifications.

Systèmes de gestion de versions centralisés

- Principe :
 - Chaque modification (commit) est envoyée directement sur le serveur.
- Les plus connus :
 - CVS
 - Subversion
- Inconvénient : Nécessite une connexion pour envoyer ses modifications.

Systèmes de gestion de versions centralisés

- Principe :
 - Chaque modification (commit) est envoyée directement sur le serveur.
- Les plus connus :
 - CVS
 - Subversion
- Inconvénient : Nécessite une connexion pour envoyer ses modifications.

Systèmes de gestion de versions centralisés

- Principe :
 - Chaque modification (commit) est envoyée directement sur le serveur.
- Les plus connus :
 - CVS
 - Subversion
- Inconvénient : Nécessite une connexion pour envoyer ses modifications.

Systèmes de gestion de versions décentralisés

- Principe :

- Chaque modification (commit) est stockée localement.
- L'ensemble des modifications est envoyée à un dépôt distant (remote) ;
- Possibilité d'avoir plusieurs dépôts distants.

- Les plus connus :

- BitKeeper
- Git
- Mercurial
- GNU Bazaar (successeur de GNU Arch)

- Avantage : Pas besoin de connexion pour commiter ;

- Inconvénient : Workflow plus complexe.

Systèmes de gestion de versions décentralisés

- Principe :
 - Chaque modification (commit) est stockée localement.
 - L'ensemble des modifications est envoyée à un dépôt distant (remote) ;
 - Possibilité d'avoir plusieurs dépôts distants.
- Les plus connus :
 - BitKeeper
 - Git
 - Mercurial
 - GNU Bazaar (successeur de GNU Arch)
- Avantage : Pas besoin de connexion pour commiter ;
- Inconvénient : Workflow plus complexe.

Systèmes de gestion de versions décentralisés

- Principe :
 - Chaque modification (commit) est stockée localement.
 - L'ensemble des modifications est envoyée à un dépôt distant (remote) ;
 - Possibilité d'avoir plusieurs dépôts distants.
- Les plus connus :
 - BitKeeper
 - Git
 - Mercurial
 - GNU Bazaar (successeur de GNU Arch)
- Avantage : Pas besoin de connexion pour commiter ;
- Inconvénient : Workflow plus complexe.

Systèmes de gestion de versions décentralisés

- Principe :
 - Chaque modification (commit) est stockée localement.
 - L'ensemble des modifications est envoyée à un dépôt distant (remote) ;
 - Possibilité d'avoir plusieurs dépôts distants.
- Les plus connus :
 - BitKeeper
 - Git
 - Mercurial
 - GNU Bazaar (successeur de GNU Arch)
- Avantage : Pas besoin de connexion pour commiter ;
- Inconvénient : Workflow plus complexe.

Systèmes de gestion de versions décentralisés

- Principe :
 - Chaque modification (commit) est stockée localement.
 - L'ensemble des modifications est envoyée à un dépôt distant (remote) ;
 - Possibilité d'avoir plusieurs dépôts distants.
- Les plus connus :
 - BitKeeper
 - Git
 - Mercurial
 - GNU Bazaar (successeur de GNU Arch)
- Avantage : Pas besoin de connexion pour commiter ;
- Inconvénient : Workflow plus complexe.

Systèmes de gestion de versions décentralisés

- Principe :
 - Chaque modification (commit) est stockée localement.
 - L'ensemble des modifications est envoyée à un dépôt distant (remote) ;
 - Possibilité d'avoir plusieurs dépôts distants.
- Les plus connus :
 - BitKeeper
 - Git
 - Mercurial
 - GNU Bazaar (successeur de GNU Arch)
- Avantage : Pas besoin de connexion pour commiter ;
- Inconvénient : Workflow plus complexe.

Systèmes de gestion de versions décentralisés

- Principe :
 - Chaque modification (commit) est stockée localement.
 - L'ensemble des modifications est envoyée à un dépôt distant (remote) ;
 - Possibilité d'avoir plusieurs dépôts distants.
- Les plus connus :
 - BitKeeper
 - Git
 - Mercurial
 - GNU Bazaar (successeur de GNU Arch)
- Avantage : Pas besoin de connexion pour commiter ;
- Inconvénient : Workflow plus complexe.

Systèmes de gestion de versions décentralisés

- Principe :
 - Chaque modification (commit) est stockée localement.
 - L'ensemble des modifications est envoyée à un dépôt distant (remote) ;
 - Possibilité d'avoir plusieurs dépôts distants.
- Les plus connus :
 - BitKeeper
 - Git
 - Mercurial
 - GNU Bazaar (successeur de GNU Arch)
- Avantage : Pas besoin de connexion pour commiter ;
- Inconvénient : Workflow plus complexe.

Systèmes de gestion de versions décentralisés

- Principe :
 - Chaque modification (commit) est stockée localement.
 - L'ensemble des modifications est envoyée à un dépôt distant (remote) ;
 - Possibilité d'avoir plusieurs dépôts distants.
- Les plus connus :
 - BitKeeper
 - Git
 - Mercurial
 - GNU Bazaar (successeur de GNU Arch)
- Avantage : Pas besoin de connexion pour commiter ;
- Inconvénient : Workflow plus complexe.

Systèmes de gestion de versions décentralisés

- Principe :
 - Chaque modification (commit) est stockée localement.
 - L'ensemble des modifications est envoyée à un dépôt distant (remote) ;
 - Possibilité d'avoir plusieurs dépôts distants.
- Les plus connus :
 - BitKeeper
 - Git
 - Mercurial
 - GNU Bazaar (successeur de GNU Arch)
- Avantage : Pas besoin de connexion pour commiter ;
- Inconvénient : Workflow plus complexe.

Systèmes de gestion de versions décentralisés

- Principe :
 - Chaque modification (commit) est stockée localement.
 - L'ensemble des modifications est envoyée à un dépôt distant (remote) ;
 - Possibilité d'avoir plusieurs dépôts distants.
- Les plus connus :
 - BitKeeper
 - Git
 - Mercurial
 - GNU Bazaar (successeur de GNU Arch)
- Avantage : Pas besoin de connexion pour commiter ;
- Inconvénient : Workflow plus complexe.

Lexique

Trunk version de développement principale ;

- Très instable ;
- Version à partir de laquelle on crée les branches et les tags ;
- Appelé aussi 'master' (Git).

Tags version figée (release) ;

- Ne doit pas être modifié directement. ...
- ...donc utilisation de branches de maintenance.

Branche version de développement secondaire ;

- Branche de maintenance de version (hotfix) ;
- Branche de fonctionnalité (feature).

Révision/commit modification atomique d'un ou plusieurs fichiers ;

Head dernière révision du dépôt ;

Objet élément versionnable (fichier, répertoire, lien symbolique, etc.).

Lexique

Trunk version de développement principale ;

- Très instable ;
- Version à partir de laquelle on crée les branches et les tags ;
- Appelé aussi 'master' (Git).

Tags version figée (release) ;

- Ne doit pas être modifié directement. ...
- ...donc utilisation de branches de maintenance.

Branche version de développement secondaire ;

- Branche de maintenance de version (hotfix) ;
- Branche de fonctionnalité (feature).

Révision/commit modification atomique d'un ou plusieurs fichiers ;

Head dernière révision du dépôt ;

Objet élément versionnable (fichier, répertoire, lien symbolique, etc.).

Lexique

Trunk version de développement principale ;

- Très instable ;
- Version à partir de laquelle on crée les branches et les tags ;
- Appelé aussi 'master' (Git).

Tags version figée (release) ;

- Ne doit pas être modifié directement. ...
- ...donc utilisation de branches de maintenance.

Branche version de développement secondaire ;

- Branche de maintenance de version (hotfix) ;
- Branche de fonctionnalité (feature).

Révision/commit modification atomique d'un ou plusieurs fichiers ;

Head dernière révision du dépôt ;

Objet élément versionnable (fichier, répertoire, lien symbolique, etc.).

Lexique

Trunk version de développement principale ;

- Très instable ;
- Version à partir de laquelle on crée les branches et les tags ;
- Appelé aussi 'master' (Git).

Tags version figée (release) ;

- Ne doit pas être modifié directement. ...
- ...donc utilisation de branches de maintenance.

Branche version de développement secondaire ;

- Branche de maintenance de version (hotfix) ;
- Branche de fonctionnalité (feature).

Révision/commit modification atomique d'un ou plusieurs fichiers ;

Head dernière révision du dépôt ;

Objet élément versionnable (fichier, répertoire, lien symbolique, etc.).

Lexique

Trunk version de développement principale ;

- Très instable ;
- Version à partir de laquelle on crée les branches et les tags ;
- Appelé aussi 'master' (Git).

Tags version figée (release) ;

- Ne doit pas être modifié directement. . .
- . . .donc utilisation de branches de maintenance.

Branche version de développement secondaire ;

- Branche de maintenance de version (hotfix) ;
- Branche de fonctionnalité (feature).

Révision/commit modification atomique d'un ou plusieurs fichiers ;

Head dernière révision du dépôt ;

Objet élément versionnable (fichier, répertoire, lien symbolique, etc.).

Lexique

Trunk version de développement principale ;

- Très instable ;
- Version à partir de laquelle on crée les branches et les tags ;
- Appelé aussi 'master' (Git).

Tags version figée (release) ;

- Ne doit pas être modifié directement. . .
- . . .donc utilisation de branches de maintenance.

Branche version de développement secondaire ;

- Branche de maintenance de version (hotfix) ;
- Branche de fonctionnalité (feature).

Révision/commit modification atomique d'un ou plusieurs fichiers ;

Head dernière révision du dépôt ;

Objet élément versionnable (fichier, répertoire, lien symbolique, etc.).

Lexique

Trunk version de développement principale ;

- Très instable ;
- Version à partir de laquelle on crée les branches et les tags ;
- Appelé aussi 'master' (Git).

Tags version figée (release) ;

- Ne doit pas être modifié directement. . .
- . . .donc utilisation de branches de maintenance.

Branche version de développement secondaire ;

- Branche de maintenance de version (hotfix) ;
- Branche de fonctionnalité (feature).

Révision/commit modification atomique d'un ou plusieurs fichiers ;

Head dernière révision du dépôt ;

Objet élément versionnable (fichier, répertoire, lien symbolique, etc.).

Lexique

Trunk version de développement principale ;

- Très instable ;
- Version à partir de laquelle on crée les branches et les tags ;
- Appelé aussi 'master' (Git).

Tags version figée (release) ;

- Ne doit pas être modifié directement. . .
- . . .donc utilisation de branches de maintenance.

Branche version de développement secondaire ;

- Branche de maintenance de version (hotfix) ;
- Branche de fonctionnalité (feature).

Révision/commit modification atomique d'un ou plusieurs fichiers ;

Head dernière révision du dépôt ;

Objet élément versionnable (fichier, répertoire, lien symbolique, etc.).

Lexique

Trunk version de développement principale ;

- Très instable ;
- Version à partir de laquelle on crée les branches et les tags ;
- Appelé aussi 'master' (Git).

Tags version figée (release) ;

- Ne doit pas être modifié directement. . .
- . . .donc utilisation de branches de maintenance.

Branche version de développement secondaire ;

- Branche de maintenance de version (hotfix) ;
- Branche de fonctionnalité (feature).

Révision/commit modification atomique d'un ou plusieurs fichiers ;

Head dernière révision du dépôt ;

Objet élément versionnable (fichier, répertoire, lien symbolique, etc.).

Lexique

Trunk version de développement principale ;

- Très instable ;
- Version à partir de laquelle on crée les branches et les tags ;
- Appelé aussi 'master' (Git).

Tags version figée (release) ;

- Ne doit pas être modifié directement. . .
- . . .donc utilisation de branches de maintenance.

Branche version de développement secondaire ;

- Branche de maintenance de version (hotfix) ;
- Branche de fonctionnalité (feature).

Révision/commit modification atomique d'un ou plusieurs fichiers ;

Head dernière révision du dépôt ;

Objet élément versionnable (fichier, répertoire, lien symbolique, etc.).

Lexique

Trunk version de développement principale ;

- Très instable ;
- Version à partir de laquelle on crée les branches et les tags ;
- Appelé aussi 'master' (Git).

Tags version figée (release) ;

- Ne doit pas être modifié directement. . .
- . . .donc utilisation de branches de maintenance.

Branche version de développement secondaire ;

- Branche de maintenance de version (hotfix) ;
- Branche de fonctionnalité (feature).

Révision/commit modification atomique d'un ou plusieurs fichiers ;

Head dernière révision du dépôt ;

Objet élément versionnable (fichier, répertoire, lien symbolique, etc.).

Lexique

Trunk version de développement principale ;

- Très instable ;
- Version à partir de laquelle on crée les branches et les tags ;
- Appelé aussi 'master' (Git).

Tags version figée (release) ;

- Ne doit pas être modifié directement. . .
- . . .donc utilisation de branches de maintenance.

Branche version de développement secondaire ;

- Branche de maintenance de version (hotfix) ;
- Branche de fonctionnalité (feature).

Révision/commit modification atomique d'un ou plusieurs fichiers ;

Head dernière révision du dépôt ;

Objet élément versionnable (fichier, répertoire, lien symbolique, etc.).

Lexique

Trunk version de développement principale ;

- Très instable ;
- Version à partir de laquelle on crée les branches et les tags ;
- Appelé aussi 'master' (Git).

Tags version figée (release) ;

- Ne doit pas être modifié directement. . .
- . . .donc utilisation de branches de maintenance.

Branche version de développement secondaire ;

- Branche de maintenance de version (hotfix) ;
- Branche de fonctionnalité (feature).

Révision/commit modification atomique d'un ou plusieurs fichiers ;

Head dernière révision du dépôt ;

Objet élément versionnable (fichier, répertoire, lien symbolique, etc.).

Utilisation générale

Que faut-il versionner ?

- Tout ce qui est susceptible d'être modifié au cours du temps :
 - Code source ;
 - Script de construction ;
 - Documentation.
- Mais pas ce qui est générable :
 - Code compilé ;
 - Documentation d'API ;
 - Etc. ;
 - Les différents systèmes possèdent un mécanisme d'ignore.

Utilisation générale

Que faut-il versionner ?

- Tout ce qui est susceptible d'être modifié au cours du temps :
 - Code source ;
 - Script de construction ;
 - Documentation.
- Mais pas ce qui est générable :
 - Code compilé ;
 - Documentation d'API ;
 - Etc. ;
 - Les différents systèmes possèdent un mécanisme d'ignore.

Utilisation générale

Que faut-il versionner ?

- Tout ce qui est susceptible d'être modifié au cours du temps :
 - Code source ;
 - Script de construction ;
 - Documentation.
- Mais pas ce qui est générable :
 - Code compilé ;
 - Documentation d'API ;
 - Etc. ;
 - Les différents systèmes possèdent un mécanisme d'ignore.

Utilisation générale

Que faut-il versionner ?

- Tout ce qui est susceptible d'être modifié au cours du temps :
 - Code source ;
 - Script de construction ;
 - Documentation.
- Mais pas ce qui est générable :
 - Code compilé ;
 - Documentation d'API ;
 - Etc. ;
 - Les différents systèmes possèdent un mécanisme d'ignore.

Utilisation générale

Que faut-il versionner ?

- Tout ce qui est susceptible d'être modifié au cours du temps :
 - Code source ;
 - Script de construction ;
 - Documentation.
- Mais pas ce qui est générable :
 - Code compilé ;
 - Documentation d'API ;
 - Etc. ;
 - Les différents systèmes possèdent un mécanisme d'ignore.

Utilisation générale

Que faut-il versionner ?

- Tout ce qui est susceptible d'être modifié au cours du temps :
 - Code source ;
 - Script de construction ;
 - Documentation.
- Mais pas ce qui est générable :
 - Code compilé ;
 - Documentation d'API ;
 - Etc. ;
 - Les différents systèmes possèdent un mécanisme d'ignore.

Utilisation générale

Que faut-il versionner ?

- Tout ce qui est susceptible d'être modifié au cours du temps :
 - Code source ;
 - Script de construction ;
 - Documentation.
- Mais pas ce qui est générable :
 - Code compilé ;
 - Documentation d'API ;
 - Etc. ;
 - Les différents systèmes possèdent un mécanisme d'ignore.

Utilisation générale

Que faut-il versionner ?

- Tout ce qui est susceptible d'être modifié au cours du temps :
 - Code source ;
 - Script de construction ;
 - Documentation.
- Mais pas ce qui est générable :
 - Code compilé ;
 - Documentation d'API ;
 - Etc. ;
 - Les différents systèmes possèdent un mécanisme d'ignore.

Utilisation générale

Que faut-il versionner ?

- Tout ce qui est susceptible d'être modifié au cours du temps :
 - Code source ;
 - Script de construction ;
 - Documentation.
- Mais pas ce qui est générable :
 - Code compilé ;
 - Documentation d'API ;
 - Etc. ;
 - Les différents systèmes possèdent un mécanisme d'ignore.

Utilisation générale

Quand faut-il commiter ?

- Le plus souvent possible ;
- Quand ça marche :
 - Compile sans erreur ;
 - Testé et validé.
- Un commit inclut une modification et une seule.
 - Éviter les commits monolithiques.

Utilisation générale

Quand faut-il commiter ?

- Le plus souvent possible ;
- Quand ça marche :
 - Compile sans erreur ;
 - Testé et validé.
- Un commit inclut une modification et une seule.
 - Éviter les commits monolithiques.

Utilisation générale

Quand faut-il commiter ?

- Le plus souvent possible ;
- Quand ça marche :
 - Compile sans erreur ;
 - Testé et validé.
- Un commit inclut une modification et une seule.
 - Éviter les commits monolithiques.

Utilisation générale

Quand faut-il commiter ?

- Le plus souvent possible ;
- Quand ça marche :
 - Compile sans erreur ;
 - Testé et validé.
- Un commit inclut une modification et une seule.
 - Éviter les commits monolithiques.

Utilisation générale

Quand faut-il commiter ?

- Le plus souvent possible ;
- Quand ça marche :
 - Compile sans erreur ;
 - Testé et validé.
- Un commit inclut une modification et une seule.
 - Éviter les commits monolythiques.

Utilisation générale

Quand faut-il commiter ?

- Le plus souvent possible ;
- Quand ça marche :
 - Compile sans erreur ;
 - Testé et validé.
- Un commit inclut une modification et une seule.
 - Éviter les commits monolythiques.

Subversion

Informations générales

- **Anciens développeurs de CVS ;**
- Première version en 2000...
- ...version 1.0.0 en 2004 ;
- Dernière version :1.9.4 (28 avril 2016) ;
- Un dépôt central, sur un serveur, sur les protocoles :
 - SVN (svnserve, TCP 3690) ;
 - HTTP (webdav, passe dans plus d'infrastructures réseau).
- Pas de notion de branche ou de tag, utilisation de l'arborescence ;
- Possibilité de checkout une sous-arborescence ;
- Numéros de versions séquentiels, globaux au dépôt ;
- Les répertoires sont versionnés (notamment les répertoires vides) ;
- <http://svnbook.red-bean.com/> (assez ancienne, mais toujours d'actualité).

Subversion

Informations générales

- Anciens développeurs de CVS ;
- Première version en 2000...
- ...version 1.0.0 en 2004 ;
- Dernière version :1.9.4 (28 avril 2016) ;
- Un dépôt central, sur un serveur, sur les protocoles :
 - SVN (svnserve, TCP 3690) ;
 - HTTP (webdav, passe dans plus d'infrastructures réseau).
- Pas de notion de branche ou de tag, utilisation de l'arborescence ;
- Possibilité de checkout une sous-arborescence ;
- Numéros de versions séquentiels, globaux au dépôt ;
- Les répertoires sont versionnés (notamment les répertoires vides) ;
- <http://svnbook.red-bean.com/> (assez ancienne, mais toujours d'actualité).

Subversion

Informations générales

- Anciens développeurs de CVS ;
- Première version en 2000...
- ...version 1.0.0 en 2004 ;
- Dernière version :1.9.4 (28 avril 2016) ;
- Un dépôt central, sur un serveur, sur les protocoles :
 - SVN (svnserve, TCP 3690) ;
 - HTTP (webdav, passe dans plus d'infrastructures réseau).
- Pas de notion de branche ou de tag, utilisation de l'arborescence ;
- Possibilité de checkout une sous-arborescence ;
- Numéros de versions séquentiels, globaux au dépôt ;
- Les répertoires sont versionnés (notamment les répertoires vides) ;
- <http://svnbook.red-bean.com/> (assez ancienne, mais toujours d'actualité).

Subversion

Informations générales

- Anciens développeurs de CVS ;
- Première version en 2000...
- ...version 1.0.0 en 2004 ;
- Dernière version :1.9.4 (28 avril 2016) ;
- Un dépôt central, sur un serveur, sur les protocoles :
 - SVN (svnserve, TCP 3690) ;
 - HTTP (webdav, passe dans plus d'infrastructures réseau).
- Pas de notion de branche ou de tag, utilisation de l'arborescence ;
- Possibilité de checkout une sous-arborescence ;
- Numéros de versions séquentiels, globaux au dépôt ;
- Les répertoires sont versionnés (notamment les répertoires vides) ;
- <http://svnbook.red-bean.com/> (assez ancienne, mais toujours d'actualité).

Subversion

Informations générales

- Anciens développeurs de CVS ;
- Première version en 2000...
- ...version 1.0.0 en 2004 ;
- Dernière version :1.9.4 (28 avril 2016) ;
- Un dépôt central, sur un serveur, sur les protocoles :
 - SVN (svnserve, TCP 3690) ;
 - HTTP (webdav, passe dans plus d'infrastructures réseau).
- Pas de notion de branche ou de tag, utilisation de l'arborescence ;
- Possibilité de checkout une sous-arborescence ;
- Numéros de versions séquentiels, globaux au dépôt ;
- Les répertoires sont versionnés (notamment les répertoires vides) ;
- <http://svnbook.red-bean.com/> (assez ancienne, mais toujours d'actualité).

Subversion

Informations générales

- Anciens développeurs de CVS ;
- Première version en 2000...
- ...version 1.0.0 en 2004 ;
- Dernière version :1.9.4 (28 avril 2016) ;
- Un dépôt central, sur un serveur, sur les protocoles :
 - SVN (svnserve, TCP 3690) ;
 - HTTP (webdav, passe dans plus d'infrastructures réseau).
- Pas de notion de branche ou de tag, utilisation de l'arborescence ;
- Possibilité de checkout une sous-arborescence ;
- Numéros de versions séquentiels, globaux au dépôt ;
- Les répertoires sont versionnés (notamment les répertoires vides) ;
- <http://svnbook.red-bean.com/> (assez ancienne, mais toujours d'actualité).

Subversion

Informations générales

- Anciens développeurs de CVS ;
- Première version en 2000...
- ...version 1.0.0 en 2004 ;
- Dernière version :1.9.4 (28 avril 2016) ;
- Un dépôt central, sur un serveur, sur les protocoles :
 - SVN (svnserve, TCP 3690) ;
 - HTTP (webdav, passe dans plus d'infrastructures réseau).
- Pas de notion de branche ou de tag, utilisation de l'arborescence ;
- Possibilité de checkout une sous-arborescence ;
- Numéros de versions séquentiels, globaux au dépôt ;
- Les répertoires sont versionnés (notamment les répertoires vides) ;
- <http://svnbook.red-bean.com/> (assez ancienne, mais toujours d'actualité).

Subversion

Informations générales

- Anciens développeurs de CVS ;
- Première version en 2000...
- ...version 1.0.0 en 2004 ;
- Dernière version :1.9.4 (28 avril 2016) ;
- Un dépôt central, sur un serveur, sur les protocoles :
 - SVN (svnserve, TCP 3690) ;
 - HTTP (webdav, passe dans plus d'infrastructures réseau).
- Pas de notion de branche ou de tag, utilisation de l'arborescence ;
- Possibilité de checkout une sous-arborescence ;
- Numéros de versions séquentiels, globaux au dépôt ;
- Les répertoires sont versionnés (notamment les répertoires vides) ;
- <http://svnbook.red-bean.com/> (assez ancienne, mais toujours d'actualité).

Subversion

Informations générales

- Anciens développeurs de CVS ;
- Première version en 2000...
- ...version 1.0.0 en 2004 ;
- Dernière version :1.9.4 (28 avril 2016) ;
- Un dépôt central, sur un serveur, sur les protocoles :
 - SVN (svnserve, TCP 3690) ;
 - HTTP (webdav, passe dans plus d'infrastructures réseau).
- Pas de notion de branche ou de tag, utilisation de l'arborescence ;
- Possibilité de checkout une sous-arborescence ;
- Numéros de versions séquentiels, globaux au dépôt ;
- Les répertoires sont versionnés (notamment les répertoires vides) ;
- <http://svnbook.red-bean.com/> (assez ancienne, mais toujours d'actualité).

Subversion

Informations générales

- Anciens développeurs de CVS ;
- Première version en 2000...
- ...version 1.0.0 en 2004 ;
- Dernière version :1.9.4 (28 avril 2016) ;
- Un dépôt central, sur un serveur, sur les protocoles :
 - SVN (svnserve, TCP 3690) ;
 - HTTP (webdav, passe dans plus d'infrastructures réseau).
- Pas de notion de branche ou de tag, utilisation de l'arborescence ;
- Possibilité de checkout une sous-arborescence ;
- Numéros de versions séquentiels, globaux au dépôt ;
- Les répertoires sont versionnés (notamment les répertoires vides) ;
- <http://svnbook.red-bean.com/> (assez ancienne, mais toujours d'actualité).

Subversion

Informations générales

- Anciens développeurs de CVS ;
- Première version en 2000...
- ...version 1.0.0 en 2004 ;
- Dernière version :1.9.4 (28 avril 2016) ;
- Un dépôt central, sur un serveur, sur les protocoles :
 - SVN (svnserve, TCP 3690) ;
 - HTTP (webdav, passe dans plus d'infrastructures réseau).
- Pas de notion de branche ou de tag, utilisation de l'arborescence ;
- Possibilité de checkout une sous-arborescence ;
- Numéros de versions séquentiels, globaux au dépôt ;
- Les répertoires sont versionnés (notamment les répertoires vides) ;
- <http://svnbook.red-bean.com/> (assez ancienne, mais toujours d'actualité).

Subversion

Informations générales

- Anciens développeurs de CVS ;
- Première version en 2000...
- ...version 1.0.0 en 2004 ;
- Dernière version :1.9.4 (28 avril 2016) ;
- Un dépôt central, sur un serveur, sur les protocoles :
 - SVN (svnserve, TCP 3690) ;
 - HTTP (webdav, passe dans plus d'infrastructures réseau).
- Pas de notion de branche ou de tag, utilisation de l'arborescence ;
- Possibilité de checkout une sous-arborescence ;
- Numéros de versions séquentiels, globaux au dépôt ;
- Les répertoires sont versionnés (notamment les répertoires vides) ;
- <http://svnbook.red-bean.com/> (assez ancienne, mais toujours d'actualité).

Subversion

Architecture d'un dépôt

- **trunk**
 - Développements courants ;
- **branches**
 - Développements conséquents ;
 - Maintenances de versions ;
- **tags**
 - Versions figées ;
 - Pas de modification dans les tags ;

Subversion

Architecture d'un dépôt

- trunk
 - Développements courants ;
- branches
 - Développements conséquents ;
 - Maintenances de versions ;
- tags
 - Versions figées ;
 - Pas de modification dans les tags ;

Subversion

Architecture d'un dépôt

- trunk
 - Développements courants ;
- branches
 - Développements conséquents ;
 - Maintenances de versions ;
- tags
 - Versions figées ;
 - Pas de modification dans les tags ;

Subversion

Architecture d'un dépôt

- trunk
 - Développements courants ;
- branches
 - Développements conséquents ;
 - Maintenances de versions ;
- tags
 - Versions figées ;
 - Pas de modification dans les tags ;

Subversion

Architecture d'un dépôt

- trunk
 - Développements courants ;
- branches
 - Développements conséquents ;
 - Maintenances de versions ;
- tags
 - Versions figées ;
 - Pas de modification dans les tags ;

Subversion

Architecture d'un dépôt

- trunk
 - Développements courants ;
- branches
 - Développements conséquents ;
 - Maintenances de versions ;
- tags
 - Versions figées ;
 - Pas de modification dans les tags ;

Subversion

Architecture d'un dépôt

- trunk
 - Développements courants ;
- branches
 - Développements conséquents ;
 - Maintenances de versions ;
- tags
 - Versions figées ;
 - Pas de modification dans les tags ;

Subversion

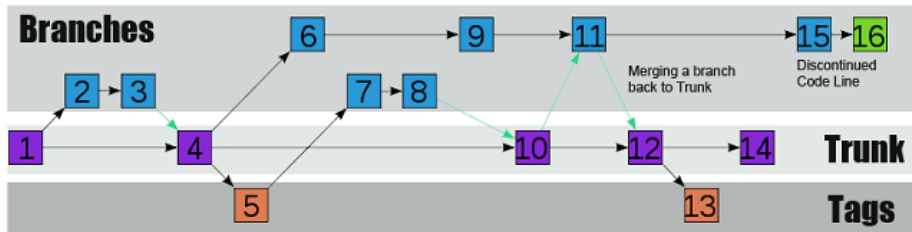
Architecture d'un dépôt

- trunk
 - Développements courants ;
- branches
 - Développements conséquents ;
 - Maintenances de versions ;
- tags
 - Versions figées ;
 - Pas de modification dans les tags ;

Subversion

Architecture d'un dépôt

FIGURE – Workflow de développement avec Subversion.



Source : <http://blogs.wandisco.com/2011/10/24/subversion-best-practices-repository-structure/>

Subversion

Workflow de développement

svn checkout créer sa copie locale ;

svn update mettre à jour sa copie locale avec les modifications des autres ;

svn add ajouter des fichiers ;

svn copy copier des fichiers ou une sous-arborescence ;

- Sert particulièrement à créer les branches et les tags.

svn delete supprimer des fichiers ;

svn move déplacer des fichiers ;

svn status voir l'état de la copie locale ;

svn diff voir les différences entre deux versions, ou de sa copie locale par rapport à la dernière mise à jour du dépôt ;

svn revert annuler une modification application inversée du diff d'un commit) ;

svn resolve effectuer une gestion des conflits sur un fichier ;

svn resolved marquer un conflit comme résolu ;

svn commit pousser ses modifications sur le dépôt ;

svn merge fusionner une branche dans une autre ;

svn switch changer de branche (changer le point d'origine dans l'arborescence distante) ;

svn info avoir les informations de sa copie locale.



Subversion

Workflow de développement

- `svn checkout` créer sa copie locale ;
- `svn update` mettre à jour sa copie locale avec les modifications des autres ;
 - `svn add` ajouter des fichiers ;
 - `svn copy` copier des fichiers ou une sous-arborescence ;
 - Sert particulièrement à créer les branches et les tags.
- `svn delete` supprimer des fichiers ;
- `svn move` déplacer des fichiers ;
- `svn status` voir l'état de la copie locale ;
- `svn diff` voir les différences entre deux versions, ou de sa copie locale par rapport à la dernière mise à jour du dépôt ;
- `svn revert` annuler une modification application inversée du diff d'un commit) ;
- `svn resolve` effectuer une gestion des conflits sur un fichier ;
- `svn resolved` marquer un conflit comme résolu ;
- `svn commit` pousser ses modifications sur le dépôt ;
- `svn merge` fusionner une branche dans une autre ;
- `svn switch` changer de branche (changer le point d'origine dans l'arborescence distante) ;
- `svn info` avoir les informations de sa copie locale.



Subversion

Workflow de développement

- `svn checkout` créer sa copie locale ;
- `svn update` mettre à jour sa copie locale avec les modifications des autres ;
- `svn add` ajouter des fichiers ;
- `svn copy` copier des fichiers ou une sous-arborescence ;
 - Sert particulièrement à créer les branches et les tags.
- `svn delete` supprimer des fichiers ;
- `svn move` déplacer des fichiers ;
- `svn status` voir l'état de la copie locale ;
- `svn diff` voir les différences entre deux versions, ou de sa copie locale par rapport à la dernière mise à jour du dépôt ;
- `svn revert` annuler une modification application inversée du diff d'un commit) ;
- `svn resolve` effectuer une gestion des conflits sur un fichier ;
- `svn resolved` marquer un conflit comme résolu ;
- `svn commit` pousser ses modifications sur le dépôt ;
- `svn merge` fusionner une branche dans une autre ;
- `svn switch` changer de branche (changer le point d'origine dans l'arborescence distante) ;
- `svn info` avoir les informations de sa copie locale.



Subversion

Workflow de développement

- `svn checkout` créer sa copie locale ;
- `svn update` mettre à jour sa copie locale avec les modifications des autres ;
- `svn add` ajouter des fichiers ;
- `svn copy` copier des fichiers ou une sous-arborescence ;
 - Sert particulièrement à créer les branches et les tags.
- `svn delete` supprimer des fichiers ;
- `svn move` déplacer des fichiers ;
- `svn status` voir l'état de la copie locale ;
- `svn diff` voir les différences entre deux versions, ou de sa copie locale par rapport à la dernière mise à jour du dépôt ;
- `svn revert` annuler une modification application inversée du diff d'un commit) ;
- `svn resolve` effectuer une gestion des conflits sur un fichier ;
- `svn resolved` marquer un conflit comme résolu ;
- `svn commit` pousser ses modifications sur le dépôt ;
- `svn merge` fusionner une branche dans une autre ;
- `svn switch` changer de branche (changer le point d'origine dans l'arborescence distante) ;
- `svn info` avoir les informations de sa copie locale.



Subversion

Workflow de développement

- `svn checkout` créer sa copie locale ;
- `svn update` mettre à jour sa copie locale avec les modifications des autres ;
- `svn add` ajouter des fichiers ;
- `svn copy` copier des fichiers ou une sous-arborescence ;
 - Sert particulièrement à créer les branches et les tags.
- `svn delete` supprimer des fichiers ;
- `svn move` déplacer des fichiers ;
- `svn status` voir l'état de la copie locale ;
- `svn diff` voir les différences entre deux versions, ou de sa copie locale par rapport à la dernière mise à jour du dépôt ;
- `svn revert` annuler une modification application inversée du diff d'un commit) ;
- `svn resolve` effectuer une gestion des conflits sur un fichier ;
- `svn resolved` marquer un conflit comme résolu ;
- `svn commit` pousser ses modifications sur le dépôt ;
- `svn merge` fusionner une branche dans une autre ;
- `svn switch` changer de branche (changer le point d'origine dans l'arborescence distante) ;
- `svn info` avoir les informations de sa copie locale.



Subversion

Workflow de développement

- `svn checkout` créer sa copie locale ;
- `svn update` mettre à jour sa copie locale avec les modifications des autres ;
- `svn add` ajouter des fichiers ;
- `svn copy` copier des fichiers ou une sous-arborescence ;
 - Sert particulièrement à créer les branches et les tags.
- `svn delete` supprimer des fichiers ;
- `svn move` déplacer des fichiers ;
- `svn status` voir l'état de la copie locale ;
- `svn diff` voir les différences entre deux versions, ou de sa copie locale par rapport à la dernière mise à jour du dépôt ;
- `svn revert` annuler une modification application inversée du diff d'un commit) ;
- `svn resolve` effectuer une gestion des conflits sur un fichier ;
- `svn resolved` marquer un conflit comme résolu ;
- `svn commit` pousser ses modifications sur le dépôt ;
- `svn merge` fusionner une branche dans une autre ;
- `svn switch` changer de branche (changer le point d'origine dans l'arborescence distante) ;
- `svn info` avoir les informations de sa copie locale.



Subversion

Workflow de développement

- `svn checkout` créer sa copie locale ;
- `svn update` mettre à jour sa copie locale avec les modifications des autres ;
- `svn add` ajouter des fichiers ;
- `svn copy` copier des fichiers ou une sous-arborescence ;
 - Sert particulièrement à créer les branches et les tags.
- `svn delete` supprimer des fichiers ;
- `svn move` déplacer des fichiers ;
- `svn status` voir l'état de la copie locale ;
- `svn diff` voir les différences entre deux versions, ou de sa copie locale par rapport à la dernière mise à jour du dépôt ;
- `svn revert` annuler une modification application inversée du diff d'un commit) ;
- `svn resolve` effectuer une gestion des conflits sur un fichier ;
- `svn resolved` marquer un conflit comme résolu ;
- `svn commit` pousser ses modifications sur le dépôt ;
- `svn merge` fusionner une branche dans une autre ;
- `svn switch` changer de branche (changer le point d'origine dans l'arborescence distante) ;
- `svn info` avoir les informations de sa copie locale.



Subversion

Workflow de développement

- `svn checkout` créer sa copie locale ;
- `svn update` mettre à jour sa copie locale avec les modifications des autres ;
- `svn add` ajouter des fichiers ;
- `svn copy` copier des fichiers ou une sous-arborescence ;
 - Sert particulièrement à créer les branches et les tags.
- `svn delete` supprimer des fichiers ;
- `svn move` déplacer des fichiers ;
- `svn status` voir l'état de la copie locale ;
- `svn diff` voir les différences entre deux versions, ou de sa copie locale par rapport à la dernière mise à jour du dépôt ;
- `svn revert` annuler une modification application inversée du diff d'un commit) ;
- `svn resolve` effectuer une gestion des conflits sur un fichier ;
- `svn resolved` marquer un conflit comme résolu ;
- `svn commit` pousser ses modifications sur le dépôt ;
- `svn merge` fusionner une branche dans une autre ;
- `svn switch` changer de branche (changer le point d'origine dans l'arborescence distante) ;
- `svn info` avoir les informations de sa copie locale.



Subversion

Workflow de développement

- `svn checkout` créer sa copie locale ;
- `svn update` mettre à jour sa copie locale avec les modifications des autres ;
- `svn add` ajouter des fichiers ;
- `svn copy` copier des fichiers ou une sous-arborescence ;
 - Sert particulièrement à créer les branches et les tags.
- `svn delete` supprimer des fichiers ;
- `svn move` déplacer des fichiers ;
- `svn status` voir l'état de la copie locale ;
- `svn diff` voir les différences entre deux versions, ou de sa copie locale par rapport à la dernière mise à jour du dépôt ;
- `svn revert` annuler une modification application inversée du diff d'un commit) ;
- `svn resolve` effectuer une gestion des conflits sur un fichier ;
- `svn resolved` marquer un conflit comme résolu ;
- `svn commit` pousser ses modifications sur le dépôt ;
- `svn merge` fusionner une branche dans une autre ;
- `svn switch` changer de branche (changer le point d'origine dans l'arborescence distante) ;
- `svn info` avoir les informations de sa copie locale.



Subversion

Workflow de développement

- `svn checkout` créer sa copie locale ;
- `svn update` mettre à jour sa copie locale avec les modifications des autres ;
- `svn add` ajouter des fichiers ;
- `svn copy` copier des fichiers ou une sous-arborescence ;
 - Sert particulièrement à créer les branches et les tags.
- `svn delete` supprimer des fichiers ;
- `svn move` déplacer des fichiers ;
- `svn status` voir l'état de la copie locale ;
- `svn diff` voir les différences entre deux versions, ou de sa copie locale par rapport à la dernière mise à jour du dépôt ;
- `svn revert` annuler une modification application inversée du diff d'un commit) ;
- `svn resolve` effectuer une gestion des conflits sur un fichier ;
- `svn resolved` marquer un conflit comme résolu ;
- `svn commit` pousser ses modifications sur le dépôt ;
- `svn merge` fusionner une branche dans une autre ;
- `svn switch` changer de branche (changer le point d'origine dans l'arborescence distante) ;
- `svn info` avoir les informations de sa copie locale.



Subversion

Workflow de développement

- `svn checkout` créer sa copie locale ;
- `svn update` mettre à jour sa copie locale avec les modifications des autres ;
- `svn add` ajouter des fichiers ;
- `svn copy` copier des fichiers ou une sous-arborescence ;
 - Sert particulièrement à créer les branches et les tags.
- `svn delete` supprimer des fichiers ;
- `svn move` déplacer des fichiers ;
- `svn status` voir l'état de la copie locale ;
- `svn diff` voir les différences entre deux versions, ou de sa copie locale par rapport à la dernière mise à jour du dépôt ;
- `svn revert` annuler une modification application inversée du diff d'un commit) ;
- `svn resolve` effectuer une gestion des conflits sur un fichier ;
- `svn resolved` marquer un conflit comme résolu ;
- `svn commit` pousser ses modifications sur le dépôt ;
- `svn merge` fusionner une branche dans une autre ;
- `svn switch` changer de branche (changer le point d'origine dans l'arborescence distante) ;
- `svn info` avoir les informations de sa copie locale.



Subversion

Workflow de développement

- `svn checkout` créer sa copie locale ;
- `svn update` mettre à jour sa copie locale avec les modifications des autres ;
- `svn add` ajouter des fichiers ;
- `svn copy` copier des fichiers ou une sous-arborescence ;
 - Sert particulièrement à créer les branches et les tags.
- `svn delete` supprimer des fichiers ;
- `svn move` déplacer des fichiers ;
- `svn status` voir l'état de la copie locale ;
- `svn diff` voir les différences entre deux versions, ou de sa copie locale par rapport à la dernière mise à jour du dépôt ;
- `svn revert` annuler une modification application inversée du diff d'un commit) ;
- `svn resolve` effectuer une gestion des conflits sur un fichier ;
- `svn resolved` marquer un conflit comme résolu ;
- `svn commit` pousser ses modifications sur le dépôt ;
- `svn merge` fusionner une branche dans une autre ;
- `svn switch` changer de branche (changer le point d'origine dans l'arborescence distante) ;
- `svn info` avoir les informations de sa copie locale.



Subversion

Workflow de développement

- `svn checkout` créer sa copie locale ;
- `svn update` mettre à jour sa copie locale avec les modifications des autres ;
- `svn add` ajouter des fichiers ;
- `svn copy` copier des fichiers ou une sous-arborescence ;
 - Sert particulièrement à créer les branches et les tags.
- `svn delete` supprimer des fichiers ;
- `svn move` déplacer des fichiers ;
- `svn status` voir l'état de la copie locale ;
- `svn diff` voir les différences entre deux versions, ou de sa copie locale par rapport à la dernière mise à jour du dépôt ;
- `svn revert` annuler une modification application inversée du diff d'un commit) ;
- `svn resolve` effectuer une gestion des conflits sur un fichier ;
- `svn resolved` marquer un conflit comme résolu ;
- `svn commit` pousser ses modifications sur le dépôt ;
- `svn merge` fusionner une branche dans une autre ;
- `svn switch` changer de branche (changer le point d'origine dans l'arborescence distante) ;
- `svn info` avoir les informations de sa copie locale.



Subversion

Workflow de développement

- `svn checkout` créer sa copie locale ;
- `svn update` mettre à jour sa copie locale avec les modifications des autres ;
- `svn add` ajouter des fichiers ;
- `svn copy` copier des fichiers ou une sous-arborescence ;
 - Sert particulièrement à créer les branches et les tags.
- `svn delete` supprimer des fichiers ;
- `svn move` déplacer des fichiers ;
- `svn status` voir l'état de la copie locale ;
- `svn diff` voir les différences entre deux versions, ou de sa copie locale par rapport à la dernière mise à jour du dépôt ;
- `svn revert` annuler une modification application inversée du diff d'un commit) ;
- `svn resolve` effectuer une gestion des conflits sur un fichier ;
- `svn resolved` marquer un conflit comme résolu ;
- `svn commit` pousser ses modifications sur le dépôt ;
- `svn merge` fusionner une branche dans une autre ;
- `svn switch` changer de branche (changer le point d'origine dans l'arborescence distante) ;
- `svn info` avoir les informations de sa copie locale.



Subversion

Workflow de développement

- `svn checkout` créer sa copie locale ;
- `svn update` mettre à jour sa copie locale avec les modifications des autres ;
- `svn add` ajouter des fichiers ;
- `svn copy` copier des fichiers ou une sous-arborescence ;
 - Sert particulièrement à créer les branches et les tags.
- `svn delete` supprimer des fichiers ;
- `svn move` déplacer des fichiers ;
- `svn status` voir l'état de la copie locale ;
- `svn diff` voir les différences entre deux versions, ou de sa copie locale par rapport à la dernière mise à jour du dépôt ;
- `svn revert` annuler une modification application inversée du diff d'un commit) ;
- `svn resolve` effectuer une gestion des conflits sur un fichier ;
- `svn resolved` marquer un conflit comme résolu ;
- `svn commit` pousser ses modifications sur le dépôt ;
- `svn merge` fusionner une branche dans une autre ;
- `svn switch` changer de branche (changer le point d'origine dans l'arborescence distante) ;
- `svn info` avoir les informations de sa copie locale.



svn status

- Affiche l'état de la copie locale ;
- Permet d'avoir les modifications sur le serveur avec l'option `--show-updates` ou `-u` ;

Exemple d'invocation (honteusement copié sur <http://svnbook.red-bean.com/en/1.8/svn.ref.svn.c.status.html>)

```
1 # svn status -u wc
2   M          965    wc/bar.c
3   *          965    wc/foo.c
4   A +        965    wc/qax.c
5 Status against revision: 981
```

svn status

- Affiche l'état de la copie locale ;
- Permet d'avoir les modifications sur le serveur avec l'option `--show-updates` ou `-u` ;

Exemple d'invocation (honteusement copié sur <http://svnbook.red-bean.com/en/1.8/svn.ref.svn.c.status.html>)

```
1 # svn status -u wc
2   M          965   wc/bar.c
3   *          965   wc/foo.c
4   A +        965   wc/qax.c
5 Status against revision: 981
```

svn status – format

- Première colonne : changements sur l'objet :

- objet inchangé ;

- A objet ajouté (svn add) ;

- D objet supprimé (svn delete) ;

- M objet modifié ;

- C objet en conflit (après svn update) ;

- I objet ignoré ;

- ? objet inconnu (non ajouté) ;

- ! objet manquant (supprimé du système de fichiers mais pas avec svn delete)

- ~ objet ayant changé de type (par ex. : fichier -> répertoire)

svn status – format

- Première colonne : changements sur l'objet :

- └─ objet inchangé ;

- A objet ajouté (svn add) ;

- D objet supprimé (svn delete) ;

- M objet modifié ;

- C objet en conflit (après svn update) ;

- I objet ignoré ;

- ? objet inconnu (non ajouté) ;

- ! objet manquant (supprimé du système de fichiers mais pas avec svn delete)

- ~ objet ayant changé de type (par ex. : fichier -> répertoire)

svn status – format

- Première colonne : changements sur l'objet :

- └ objet inchangé ;

- A** objet ajouté (svn add) ;

- D objet supprimé (svn delete) ;

- M objet modifié ;

- C objet en conflit (après svn update) ;

- I objet ignoré ;

- ? objet inconnu (non ajouté) ;

- ! objet manquant (supprimé du système de fichiers mais pas avec svn delete)

- ~ objet ayant changé de type (par ex. : fichier -> répertoire)

svn status – format

- Première colonne : changements sur l'objet :

- └ objet inchangé ;

- A objet ajouté (svn add) ;

- D objet supprimé (svn delete) ;

- M objet modifié ;

- C objet en conflit (après svn update) ;

- I objet ignoré ;

- ? objet inconnu (non ajouté) ;

- ! objet manquant (supprimé du système de fichiers mais pas avec svn delete)

- ~ objet ayant changé de type (par ex. : fichier -> répertoire)

svn status – format

- Première colonne : changements sur l'objet :

- └ objet inchangé ;

- A objet ajouté (svn add) ;

- D objet supprimé (svn delete) ;

- M objet modifié ;

- C objet en conflit (après svn update) ;

- I objet ignoré ;

- ? objet inconnu (non ajouté) ;

- ! objet manquant (supprimé du système de fichiers mais pas avec svn delete)

- ~ objet ayant changé de type (par ex. : fichier -> répertoire)

svn status – format

- Première colonne : changements sur l'objet :

- objet inchangé ;

- A objet ajouté (svn add) ;

- D objet supprimé (svn delete) ;

- M objet modifié ;

- C objet en conflit (après svn update) ;

- I objet ignoré ;

- ? objet inconnu (non ajouté) ;

- ! objet manquant (supprimé du système de fichiers mais pas avec svn delete)

- ~ objet ayant changé de type (par ex. : fichier -> répertoire)

svn status – format

- Première colonne : changements sur l'objet :

- objet inchangé ;

- A objet ajouté (svn add) ;

- D objet supprimé (svn delete) ;

- M objet modifié ;

- C objet en conflit (après svn update) ;

- I objet ignoré ;

- ? objet inconnu (non ajouté) ;

- ! objet manquant (supprimé du système de fichiers mais pas avec svn delete)

- ~ objet ayant changé de type (par ex. : fichier -> répertoire)

svn status – format

- Première colonne : changements sur l'objet :

- objet inchangé ;

- A objet ajouté (svn add) ;

- D objet supprimé (svn delete) ;

- M objet modifié ;

- C objet en conflit (après svn update) ;

- I objet ignoré ;

- ? objet inconnu (non ajouté) ;

- ! objet manquant (supprimé du système de fichiers mais pas avec svn delete)

- ~ objet ayant changé de type (par ex. : fichier -> répertoire)

svn status – format

- Première colonne : changements sur l'objet :
 - └ objet inchangé ;
 - A objet ajouté (svn add) ;
 - D objet supprimé (svn delete) ;
 - M objet modifié ;
 - C objet en conflit (après svn update) ;
 - I objet ignoré ;
 - ? objet inconnu (non ajouté) ;
 - ! objet manquant (supprimé du système de fichiers mais pas avec svn delete)
 - ~ objet ayant changé de type (par ex. : fichier -> répertoire)

svn status – format

- Première colonne : changements sur l'objet :
 - └ objet inchangé ;
 - A objet ajouté (svn add) ;
 - D objet supprimé (svn delete) ;
 - M objet modifié ;
 - C objet en conflit (après svn update) ;
 - I objet ignoré ;
 - ? objet inconnu (non ajouté) ;
 - ! objet manquant (supprimé du système de fichiers mais pas avec svn delete)
 - ~ objet ayant changé de type (par ex. : fichier -> répertoire)

svn status – format

- **Seconde colonne : changements sur les propriétés de l'objet :**
 - propriétés inchangées ;
 - M propriétés modifiées ;
 - C propriétés en conflit (modifiées localement et sur le dépôt distant).
- Troisième colonne : L si copie locale verrouillée ;
- Quatrième colonne : + si ajout avec historique ;
- Cinquième colonne : S si l'objet a changé de branche ;
- Sixième colonne : informations de verrouillage distant ;
- Septième colonne : informations de conflit d'arbre ;
- Huitième colonne : toujours une espace ;
- Neuvième colonne : * si l'objet est périmé (mis à jour sur le dépôt distant) ;
- Révision locale (si --show-updates ou --verbose passé en paramètre) ;
- Dernière révision commitée (si --verbose passé en paramètre) ;
- Toujours en dernier : chemin de l'objet concerné, peut contenir des espaces.

svn status – format

- Seconde colonne : changements sur les propriétés de l'objet :
 - └ propriétés inchangées ;
 - M propriétés modifiées ;
 - C propriétés en conflit (modifiées localement et sur le dépôt distant).
- Troisième colonne : L si copie locale verrouillée ;
- Quatrième colonne : + si ajout avec historique ;
- Cinquième colonne : S si l'objet a changé de branche ;
- Sixième colonne : informations de verrouillage distant ;
- Septième colonne : informations de conflit d'arbre ;
- Huitième colonne : toujours une espace ;
- Neuvième colonne : * si l'objet est périmé (mis à jour sur le dépôt distant) ;
- Révision locale (si --show-updates ou --verbose passé en paramètre) ;
- Dernière révision commitée (si --verbose passé en paramètre) ;
- Toujours en dernier : chemin de l'objet concerné, peut contenir des espaces.

svn status – format

- Seconde colonne : changements sur les propriétés de l'objet :
 - L propriétés inchangées ;
 - M propriétés modifiées ;
 - C propriétés en conflit (modifiées localement et sur le dépôt distant).
- Troisième colonne : L si copie locale verrouillée ;
- Quatrième colonne : + si ajout avec historique ;
- Cinquième colonne : S si l'objet a changé de branche ;
- Sixième colonne : informations de verrouillage distant ;
- Septième colonne : informations de conflit d'arbre ;
- Huitième colonne : toujours une espace ;
- Neuvième colonne : * si l'objet est périmé (mis à jour sur le dépôt distant) ;
- Révision locale (si --show-updates ou --verbose passé en paramètre) ;
- Dernière révision commitée (si --verbose passé en paramètre) ;
- Toujours en dernier : chemin de l'objet concerné, peut contenir des espaces.

svn status – format

- Seconde colonne : changements sur les propriétés de l'objet :
 - L propriétés inchangées ;
 - M propriétés modifiées ;
 - C propriétés en conflit (modifiées localement et sur le dépôt distant).
- Troisième colonne : L si copie locale verrouillée ;
- Quatrième colonne : + si ajout avec historique ;
- Cinquième colonne : S si l'objet a changé de branche ;
- Sixième colonne : informations de verrouillage distant ;
- Septième colonne : informations de conflit d'arbre ;
- Huitième colonne : toujours une espace ;
- Neuvième colonne : * si l'objet est périmé (mis à jour sur le dépôt distant) ;
- Révision locale (si --show-updates ou --verbose passé en paramètre) ;
- Dernière révision commitée (si --verbose passé en paramètre) ;
- Toujours en dernier : chemin de l'objet concerné, peut contenir des espaces.

svn status – format

- Seconde colonne : changements sur les propriétés de l'objet :
 - L propriétés inchangées ;
 - M propriétés modifiées ;
 - C propriétés en conflit (modifiées localement et sur le dépôt distant).
- Troisième colonne : L si copie locale verrouillée ;
- Quatrième colonne : + si ajout avec historique ;
- Cinquième colonne : S si l'objet a changé de branche ;
- Sixième colonne : informations de verrouillage distant ;
- Septième colonne : informations de conflit d'arbre ;
- Huitième colonne : toujours une espace ;
- Neuvième colonne : * si l'objet est périmé (mis à jour sur le dépôt distant) ;
- Révision locale (si --show-updates ou --verbose passé en paramètre) ;
- Dernière révision commitée (si --verbose passé en paramètre) ;
- Toujours en dernier : chemin de l'objet concerné, peut contenir des espaces.

svn status – format

- Seconde colonne : changements sur les propriétés de l'objet :
 - L propriétés inchangées ;
 - M propriétés modifiées ;
 - C propriétés en conflit (modifiées localement et sur le dépôt distant).
- Troisième colonne : L si copie locale verrouillée ;
- Quatrième colonne : + si ajout avec historique ;
- Cinquième colonne : S si l'objet a changé de branche ;
- Sixième colonne : informations de verrouillage distant ;
- Septième colonne : informations de conflit d'arbre ;
- Huitième colonne : toujours une espace ;
- Neuvième colonne : * si l'objet est périmé (mis à jour sur le dépôt distant) ;
- Révision locale (si --show-updates ou --verbose passé en paramètre) ;
- Dernière révision commitée (si --verbose passé en paramètre) ;
- Toujours en dernier : chemin de l'objet concerné, peut contenir des espaces.

svn status – format

- Seconde colonne : changements sur les propriétés de l'objet :
 - L propriétés inchangées ;
 - M propriétés modifiées ;
 - C propriétés en conflit (modifiées localement et sur le dépôt distant).
- Troisième colonne : L si copie locale verrouillée ;
- Quatrième colonne : + si ajout avec historique ;
- Cinquième colonne : S si l'objet a changé de branche ;
- Sixième colonne : informations de verrouillage distant ;
- Septième colonne : informations de conflit d'arbre ;
- Huitième colonne : toujours une espace ;
- Neuvième colonne : * si l'objet est périmé (mis à jour sur le dépôt distant) ;
- Révision locale (si --show-updates ou --verbose passé en paramètre) ;
- Dernière révision commitée (si --verbose passé en paramètre) ;
- Toujours en dernier : chemin de l'objet concerné, peut contenir des espaces.

svn status – format

- Seconde colonne : changements sur les propriétés de l'objet :
 - L propriétés inchangées ;
 - M propriétés modifiées ;
 - C propriétés en conflit (modifiées localement et sur le dépôt distant).
- Troisième colonne : L si copie locale verrouillée ;
- Quatrième colonne : + si ajout avec historique ;
- Cinquième colonne : S si l'objet a changé de branche ;
- Sixième colonne : informations de verrouillage distant ;
- Septième colonne : informations de conflit d'arbre ;
- Huitième colonne : toujours une espace ;
- Neuvième colonne : * si l'objet est périmé (mis à jour sur le dépôt distant) ;
- Révision locale (si --show-updates ou --verbose passé en paramètre) ;
- Dernière révision commitée (si --verbose passé en paramètre) ;
- Toujours en dernier : chemin de l'objet concerné, peut contenir des espaces.

svn status – format

- Seconde colonne : changements sur les propriétés de l'objet :
 - L propriétés inchangées ;
 - M propriétés modifiées ;
 - C propriétés en conflit (modifiées localement et sur le dépôt distant).
- Troisième colonne : L si copie locale verrouillée ;
- Quatrième colonne : + si ajout avec historique ;
- Cinquième colonne : S si l'objet a changé de branche ;
- Sixième colonne : informations de verrouillage distant ;
- Septième colonne : informations de conflit d'arbre ;
- Huitième colonne : toujours une espace ;
- Neuvième colonne : * si l'objet est périmé (mis à jour sur le dépôt distant) ;
- Révision locale (si --show-updates ou --verbose passé en paramètre) ;
- Dernière révision commitée (si --verbose passé en paramètre) ;
- Toujours en dernier : chemin de l'objet concerné, peut contenir des espaces.

svn status – format

- Seconde colonne : changements sur les propriétés de l'objet :
 - L propriétés inchangées ;
 - M propriétés modifiées ;
 - C propriétés en conflit (modifiées localement et sur le dépôt distant).
- Troisième colonne : L si copie locale verrouillée ;
- Quatrième colonne : + si ajout avec historique ;
- Cinquième colonne : S si l'objet a changé de branche ;
- Sixième colonne : informations de verrouillage distant ;
- Septième colonne : informations de conflit d'arbre ;
- Huitième colonne : toujours une espace ;
- Neuvième colonne : * si l'objet est périmé (mis à jour sur le dépôt distant) ;
- Révision locale (si --show-updates ou --verbose passé en paramètre) ;
- Dernière révision committée (si --verbose passé en paramètre) ;
- Toujours en dernier : chemin de l'objet concerné, peut contenir des espaces.

svn status – format

- Seconde colonne : changements sur les propriétés de l'objet :
 - L propriétés inchangées ;
 - M propriétés modifiées ;
 - C propriétés en conflit (modifiées localement et sur le dépôt distant).
- Troisième colonne : L si copie locale verrouillée ;
- Quatrième colonne : + si ajout avec historique ;
- Cinquième colonne : S si l'objet a changé de branche ;
- Sixième colonne : informations de verrouillage distant ;
- Septième colonne : informations de conflit d'arbre ;
- Huitième colonne : toujours une espace ;
- Neuvième colonne : * si l'objet est périmé (mis à jour sur le dépôt distant) ;
- Révision locale (si --show-updates ou --verbose passé en paramètre) ;
- Dernière révision commitée (si --verbose passé en paramètre) ;
- Toujours en dernier : chemin de l'objet concerné, peut contenir des espaces.

svn status – format

- Seconde colonne : changements sur les propriétés de l'objet :
 - L propriétés inchangées ;
 - M propriétés modifiées ;
 - C propriétés en conflit (modifiées localement et sur le dépôt distant).
- Troisième colonne : L si copie locale verrouillée ;
- Quatrième colonne : + si ajout avec historique ;
- Cinquième colonne : S si l'objet a changé de branche ;
- Sixième colonne : informations de verrouillage distant ;
- Septième colonne : informations de conflit d'arbre ;
- Huitième colonne : toujours une espace ;
- Neuvième colonne : * si l'objet est périmé (mis à jour sur le dépôt distant) ;
- Révision locale (si --show-updates ou --verbose passé en paramètre) ;
- Dernière révision commitée (si --verbose passé en paramètre) ;
- Toujours en dernier : chemin de l'objet concerné, peut contenir des espaces.

svn status – format

- Seconde colonne : changements sur les propriétés de l'objet :
 - L propriétés inchangées ;
 - M propriétés modifiées ;
 - C propriétés en conflit (modifiées localement et sur le dépôt distant).
- Troisième colonne : L si copie locale verrouillée ;
- Quatrième colonne : + si ajout avec historique ;
- Cinquième colonne : S si l'objet a changé de branche ;
- Sixième colonne : informations de verrouillage distant ;
- Septième colonne : informations de conflit d'arbre ;
- Huitième colonne : toujours une espace ;
- Neuvième colonne : * si l'objet est périmé (mis à jour sur le dépôt distant) ;
- Révision locale (si --show-updates ou --verbose passé en paramètre) ;
- Dernière révision commitée (si --verbose passé en paramètre) ;
- Toujours en dernier : chemin de l'objet concerné, peut contenir des espaces.

svn status – format

- Seconde colonne : changements sur les propriétés de l'objet :
 - L propriétés inchangées ;
 - M propriétés modifiées ;
 - C propriétés en conflit (modifiées localement et sur le dépôt distant).
- Troisième colonne : L si copie locale verrouillée ;
- Quatrième colonne : + si ajout avec historique ;
- Cinquième colonne : S si l'objet a changé de branche ;
- Sixième colonne : informations de verrouillage distant ;
- Septième colonne : informations de conflit d'arbre ;
- Huitième colonne : toujours une espace ;
- Neuvième colonne : * si l'objet est périmé (mis à jour sur le dépôt distant) ;
- Révision locale (si --show-updates ou --verbose passé en paramètre) ;
- Dernière révision commitée (si --verbose passé en paramètre) ;
- Toujours en dernier : chemin de l'objet concerné, peut contenir des espaces.

svn info

Exemple d'invocation (honteusement copié sur <http://svnbook.red-bean.com/en/1.8/svn.ref.svn.c.info.html>)

```
1 # svn info foo.c
2 Path: foo.c
3 Name: foo.c
4 Working Copy Root Path: /home/sally/projects/test
5 URL: http://svn.red-bean.com/repos/test/foo.c
6 Repository Root: http://svn.red-bean.com/repos/test
7 Repository UUID: 5e7d134a-54fb-0310-bd04-b611643e5c25
8 Revision: 4417
9 Node Kind: file
10 Schedule: normal
11 Last Changed Author: sally
12 Last Changed Rev: 20
13 Last Changed Date: 2003-01-13 16:43:13 -0600 (Mon, 13 Jan 2003)
14 Text Last Updated: 2003-01-16 21:18:16 -0600 (Thu, 16 Jan 2003)
15 Properties Last Updated: 2003-01-13 21:50:19 -0600 (Mon, 13 Jan 2003)
16 Checksum: d6aeb60b0662ccceb6bce4bac344cb66
```

Propriétés

- Métadonnées des objets ;
- Clé-valeur ;
- Personnalisées ou réservées à subversion (préfixées par svn:)
- Commandes :

svn proplist <objet> affiche les propriétés d'un objet ;

svn propedit <nom> <objet> ouvre un éditeur pour définir la valeur d'une propriété d'un objet ;

svn propset <nom> <valeur> <objet> définit la valeur d'une propriété d'un objet ;

svn propdel <nom> <objet> supprime la propriété de l'objet.

- Quelques propriétés connues :

svn:ignore liste des objets non versionnés ;

svn:eol-style type de fin de ligne (unix, dos) ;

svn:author auteur de la révision ;

svn:date date de la révision ;

svn:log message de révision.

Propriétés

- Métadonnées des objets ;
- Clé-valeur ;
- Personnalisées ou réservées à subversion (préfixées par svn:)
- Commandes :

svn proplist <objet> affiche les propriétés d'un objet ;

svn propedit <nom> <objet> ouvre un éditeur pour définir la valeur d'une propriété d'un objet ;

svn propset <nom> <valeur> <objet> définit la valeur d'une propriété d'un objet ;

svn propdel <nom> <objet> supprime la propriété de l'objet.

- Quelques propriétés connues :

svn:ignore liste des objets non versionnés ;

svn:eol-style type de fin de ligne (unix, dos) ;

svn:author auteur de la révision ;

svn:date date de la révision ;

svn:log message de révision.

Propriétés

- Métadonnées des objets ;
- Clé-valeur ;
- Personnalisées ou réservées à subversion (préfixées par `svn:`)
- Commandes :

`svn proplist <objet>` affiche les propriétés d'un objet ;

`svn propedit <nom> <objet>` ouvre un éditeur pour définir la valeur d'une propriété d'un objet ;

`svn propset <nom> <valeur> <objet>` définit la valeur d'une propriété d'un objet ;

`svn propdel <nom> <objet>` supprime la propriété de l'objet.

- Quelques propriétés connues :

`svn:ignore` liste des objets non versionnés ;

`svn:eol-style` type de fin de ligne (unix, dos) ;

`svn:author` auteur de la révision ;

`svn:date` date de la révision ;

`svn:log` message de révision.

Propriétés

- Métadonnées des objets ;
- Clé-valeur ;
- Personnalisées ou réservées à subversion (préfixées par `svn:`)
- Commandes :

`svn proplist <objet>` affiche les propriétés d'un objet ;

`svn propedit <nom> <objet>` ouvre un éditeur pour définir la valeur d'une propriété d'un objet ;

`svn propset <nom> <valeur> <objet>` définit la valeur d'une propriété d'un objet ;

`svn propdel <nom> <objet>` supprime la propriété de l'objet.

- Quelques propriétés connues :

`svn:ignore` liste des objets non versionnés ;

`svn:eol-style` type de fin de ligne (unix, dos) ;

`svn:author` auteur de la révision ;

`svn:date` date de la révision ;

`svn:log` message de révision.

Propriétés

- Métadonnées des objets ;
- Clé-valeur ;
- Personnalisées ou réservées à subversion (préfixées par `svn:`)
- Commandes :

`svn proplist <objet>` affiche les propriétés d'un objet ;

`svn propedit <nom> <objet>` ouvre un éditeur pour définir la valeur d'une propriété d'un objet ;

`svn propset <nom> <valeur> <objet>` définit la valeur d'une propriété d'un objet ;

`svn propdel <nom> <objet>` supprime la propriété de l'objet.

- Quelques propriétés connues :

`svn:ignore` liste des objets non versionnés ;

`svn:eol-style` type de fin de ligne (unix, dos) ;

`svn:author` auteur de la révision ;

`svn:date` date de la révision ;

`svn:log` message de révision.

Propriétés

- Métadonnées des objets ;
- Clé-valeur ;
- Personnalisées ou réservées à subversion (préfixées par `svn:`)
- Commandes :

`svn proplist <objet>` affiche les propriétés d'un objet ;

`svn propedit <nom> <objet>` ouvre un éditeur pour définir la valeur d'une propriété d'un objet ;

`svn propset <nom> <valeur> <objet>` définit la valeur d'une propriété d'un objet ;

`svn propdel <nom> <objet>` supprime la propriété de l'objet.

- Quelques propriétés connues :

`svn:ignore` liste des objets non versionnés ;

`svn:eol-style` type de fin de ligne (unix, dos) ;

`svn:author` auteur de la révision ;

`svn:date` date de la révision ;

`svn:log` message de révision.

Propriétés

- Métadonnées des objets ;
- Clé-valeur ;
- Personnalisées ou réservées à subversion (préfixées par `svn:`)
- Commandes :

`svn proplist <objet>` affiche les propriétés d'un objet ;

`svn propedit <nom> <objet>` ouvre un éditeur pour définir la valeur d'une propriété d'un objet ;

`svn propset <nom> <valeur> <objet>` définit la valeur d'une propriété d'un objet ;

`svn propdel <nom> <objet>` supprime la propriété de l'objet.

- Quelques propriétés connues :

`svn:ignore` liste des objets non versionnés ;

`svn:eol-style` type de fin de ligne (unix, dos) ;

`svn:author` auteur de la révision ;

`svn:date` date de la révision ;

`svn:log` message de révision.

Propriétés

- Métadonnées des objets ;
- Clé-valeur ;
- Personnalisées ou réservées à subversion (préfixées par `svn:`)
- Commandes :

`svn proplist <objet>` affiche les propriétés d'un objet ;

`svn propedit <nom> <objet>` ouvre un éditeur pour définir la valeur d'une propriété d'un objet ;

`svn propset <nom> <valeur> <objet>` définit la valeur d'une propriété d'un objet ;

`svn propdel <nom> <objet>` supprime la propriété de l'objet.

- Quelques propriétés connues :

`svn:ignore` liste des objets non versionnés ;

`svn:eol-style` type de fin de ligne (unix, dos) ;

`svn:author` auteur de la révision ;

`svn:date` date de la révision ;

`svn:log` message de révision.

Propriétés

- Métadonnées des objets ;
- Clé-valeur ;
- Personnalisées ou réservées à subversion (préfixées par `svn:`)
- Commandes :

`svn proplist <objet>` affiche les propriétés d'un objet ;

`svn propedit <nom> <objet>` ouvre un éditeur pour définir la valeur d'une propriété d'un objet ;

`svn propset <nom> <valeur> <objet>` définit la valeur d'une propriété d'un objet ;

`svn propdel <nom> <objet>` supprime la propriété de l'objet.

- Quelques propriétés connues :

`svn:ignore` liste des objets non versionnés ;

`svn:eol-style` type de fin de ligne (unix, dos) ;

`svn:author` auteur de la révision ;

`svn:date` date de la révision ;

`svn:log` message de révision.

Propriétés

- Métadonnées des objets ;
- Clé-valeur ;
- Personnalisées ou réservées à subversion (préfixées par `svn:`)
- Commandes :

`svn proplist <objet>` affiche les propriétés d'un objet ;

`svn propedit <nom> <objet>` ouvre un éditeur pour définir la valeur d'une propriété d'un objet ;

`svn propset <nom> <valeur> <objet>` définit la valeur d'une propriété d'un objet ;

`svn propdel <nom> <objet>` supprime la propriété de l'objet.

- Quelques propriétés connues :

`svn:ignore` liste des objets non versionnés ;

`svn:eol-style` type de fin de ligne (unix, dos) ;

`svn:author` auteur de la révision ;

`svn:date` date de la révision ;

`svn:log` message de révision.

Propriétés

- Métadonnées des objets ;
- Clé-valeur ;
- Personnalisées ou réservées à subversion (préfixées par `svn:`)
- Commandes :

`svn proplist <objet>` affiche les propriétés d'un objet ;

`svn propedit <nom> <objet>` ouvre un éditeur pour définir la valeur d'une propriété d'un objet ;

`svn propset <nom> <valeur> <objet>` définit la valeur d'une propriété d'un objet ;

`svn propdel <nom> <objet>` supprime la propriété de l'objet.

- Quelques propriétés connues :

`svn:ignore` liste des objets non versionnés ;

`svn:eol-style` type de fin de ligne (unix, dos) ;

`svn:author` auteur de la révision ;

`svn:date` date de la révision ;

`svn:log` message de révision.

Propriétés

- Métadonnées des objets ;
- Clé-valeur ;
- Personnalisées ou réservées à subversion (préfixées par `svn:`)
- Commandes :

`svn proplist <objet>` affiche les propriétés d'un objet ;

`svn propedit <nom> <objet>` ouvre un éditeur pour définir la valeur d'une propriété d'un objet ;

`svn propset <nom> <valeur> <objet>` définit la valeur d'une propriété d'un objet ;

`svn propdel <nom> <objet>` supprime la propriété de l'objet.

- Quelques propriétés connues :

`svn:ignore` liste des objets non versionnés ;

`svn:eol-style` type de fin de ligne (unix, dos) ;

`svn:author` auteur de la révision ;

`svn:date` date de la révision ;

`svn:log` message de révision.

Propriétés

- Métadonnées des objets ;
- Clé-valeur ;
- Personnalisées ou réservées à subversion (préfixées par `svn:`)
- Commandes :

`svn proplist <objet>` affiche les propriétés d'un objet ;

`svn propedit <nom> <objet>` ouvre un éditeur pour définir la valeur d'une propriété d'un objet ;

`svn propset <nom> <valeur> <objet>` définit la valeur d'une propriété d'un objet ;

`svn propdel <nom> <objet>` supprime la propriété de l'objet.

- Quelques propriétés connues :

`svn:ignore` liste des objets non versionnés ;

`svn:eol-style` type de fin de ligne (unix, dos) ;

`svn:author` auteur de la révision ;

`svn:date` date de la révision ;

`svn:log` message de révision.

Propriétés

- Métadonnées des objets ;
- Clé-valeur ;
- Personnalisées ou réservées à subversion (préfixées par `svn:`)
- Commandes :

`svn proplist <objet>` affiche les propriétés d'un objet ;

`svn propedit <nom> <objet>` ouvre un éditeur pour définir la valeur d'une propriété d'un objet ;

`svn propset <nom> <valeur> <objet>` définit la valeur d'une propriété d'un objet ;

`svn propdel <nom> <objet>` supprime la propriété de l'objet.

- Quelques propriétés connues :

`svn:ignore` liste des objets non versionnés ;

`svn:eol-style` type de fin de ligne (unix, dos) ;

`svn:author` auteur de la révision ;

`svn:date` date de la révision ;

`svn:log` message de révision.

svn:ignore

- Contient la liste des patterns d'objets ignorés ;
- S'applique à un répertoire.

Exemple

```
1  *~  
2  *.sw*  
3  *.pyc
```

svn:ignore

- Contient la liste des patterns d'objets ignorés ;
- S'applique à un répertoire.

Exemple

```
1  *~  
2  .* , SW*  
3  *.pyc
```

Subversion

Outils annexes

TortoiseSVN Intégration dans l'explorateur de fichiers Windows ;

eSVN Interface graphique dédiée, multiplateforme ;

kdesvn Interface graphique dédiée, Linux uniquement ;

Différents IDE La plupart des IDE ont une intégration de subversion ;

Forges Outils web intégrés pour le développement.

Etc. Modules d'éditeurs de texte, intégration gestionnaires de fichiers, ...

Subversion

Outils annexes

TortoiseSVN Intégration dans l'explorateur de fichiers Windows ;

eSVN Interface graphique dédiée, multiplateforme ;

kdesvn Interface graphique dédiée, Linux uniquement ;

Différents IDE La plupart des IDE ont une intégration de subversion ;

Forges Outils web intégrés pour le développement.

Etc. Modules d'éditeurs de texte, intégration gestionnaires de fichiers, ...

Subversion

Outils annexes

TortoiseSVN Intégration dans l'explorateur de fichiers Windows ;

eSVN Interface graphique dédiée, multiplateforme ;

kdesvn Interface graphique dédiée, Linux uniquement ;

Différents IDE La plupart des IDE ont une intégration de subversion ;

Forges Outils web intégrés pour le développement.

Etc. Modules d'éditeurs de texte, intégration gestionnaires de fichiers, ...

Subversion

Outils annexes

TortoiseSVN Intégration dans l'explorateur de fichiers Windows ;

eSVN Interface graphique dédiée, multiplateforme ;

kdesvn Interface graphique dédiée, Linux uniquement ;

Différents IDE La plupart des IDE ont une intégration de subversion ;

Forges Outils web intégrés pour le développement.

Etc. Modules d'éditeurs de texte, intégration gestionnaires de fichiers, ...

Subversion

Outils annexes

TortoiseSVN Intégration dans l'explorateur de fichiers Windows ;

eSVN Interface graphique dédiée, multiplateforme ;

kdesvn Interface graphique dédiée, Linux uniquement ;

Différents IDE La plupart des IDE ont une intégration de subversion ;

Forges Outils web intégrés pour le développement.

Etc. Modules d'éditeurs de texte, intégration gestionnaires de fichiers, ...

Subversion

Outils annexes

TortoiseSVN Intégration dans l'explorateur de fichiers Windows ;

eSVN Interface graphique dédiée, multiplateforme ;

kdesvn Interface graphique dédiée, Linux uniquement ;

Différents IDE La plupart des IDE ont une intégration de subversion ;

Forges Outils web intégrés pour le développement.

Etc. Modules d'éditeurs de texte, intégration gestionnaires de fichiers, ...

Subversion

Outils annexes

TortoiseSVN Intégration dans l'explorateur de fichiers Windows ;

eSVN Interface graphique dédiée, multiplateforme ;

kdesvn Interface graphique dédiée, Linux uniquement ;

Différents IDE La plupart des IDE ont une intégration de subversion ;

Forges Outils web intégrés pour le développement.

Etc. Modules d'éditeurs de texte, intégration gestionnaires de fichiers, ...

Informations générales

- Initialement écrit par Linus Torvalds pour remplacer BitKeeper ;
- Première version le 7 avril 2005 ;
- Dernière version le 2 septembre 2016 (2.10.0) ;
- Possibilité d'avoir plusieurs dépôts distants (remote) ;
- Accessibles via les protocoles :
 - SSH (authentification par clé)
 - HTTP (clonage anonyme)
 - git (en pratique peu utilisé)
- La copie locale est un clone (complet ou incomplet) du dépôt distant ;
- Notions de tags et de branches ;
- Pas de séquence, utilisation de hash pour identifier les commits ;
- Les répertoires ne sont pas versionnés, donc absence de répertoires vides ;
- Possibilité d'étendre la commande (`git flow` par exemple) ;
- <https://git-scm.com/book/fr/v2/>.

Informations générales

- Initialement écrit par Linus Torvalds pour remplacer BitKeeper ;
- Première version le 7 avril 2005 ;
- Dernière version le 2 septembre 2016 (2.10.0) ;
- Possibilité d'avoir plusieurs dépôts distants (remote) ;
- Accessibles via les protocoles :
 - SSH (authentification par clé)
 - HTTP (clonage anonyme)
 - git (en pratique peu utilisé)
- La copie locale est un clone (complet ou incomplet) du dépôt distant ;
- Notions de tags et de branches ;
- Pas de séquence, utilisation de hash pour identifier les commits ;
- Les répertoires ne sont pas versionnés, donc absence de répertoires vides ;
- Possibilité d'étendre la commande (`git flow` par exemple) ;
- <https://git-scm.com/book/fr/v2/>.

Informations générales

- Initialement écrit par Linus Torvalds pour remplacer BitKeeper ;
- Première version le 7 avril 2005 ;
- Dernière version le 2 septembre 2016 (2.10.0) ;
- Possibilité d'avoir plusieurs dépôts distants (remote) ;
- Accessibles via les protocoles :
 - SSH (authentification par clé)
 - HTTP (clonage anonyme)
 - git (en pratique peu utilisé)
- La copie locale est un clone (complet ou incomplet) du dépôt distant ;
- Notions de tags et de branches ;
- Pas de séquence, utilisation de hash pour identifier les commits ;
- Les répertoires ne sont pas versionnés, donc absence de répertoires vides ;
- Possibilité d'étendre la commande (`git flow` par exemple) ;
- <https://git-scm.com/book/fr/v2/>.

Informations générales

- Initialement écrit par Linus Torvalds pour remplacer BitKeeper ;
- Première version le 7 avril 2005 ;
- Dernière version le 2 septembre 2016 (2.10.0) ;
- Possibilité d'avoir plusieurs dépôts distants (remote) ;
- Accessibles via les protocoles :
 - SSH (authentification par clé)
 - HTTP (clonage anonyme)
 - git (en pratique peu utilisé)
- La copie locale est un clone (complet ou incomplet) du dépôt distant ;
- Notions de tags et de branches ;
- Pas de séquence, utilisation de hash pour identifier les commits ;
- Les répertoires ne sont pas versionnés, donc absence de répertoires vides ;
- Possibilité d'étendre la commande (`git flow` par exemple) ;
- <https://git-scm.com/book/fr/v2/>.

Informations générales

- Initialement écrit par Linus Torvalds pour remplacer BitKeeper ;
- Première version le 7 avril 2005 ;
- Dernière version le 2 septembre 2016 (2.10.0) ;
- Possibilité d'avoir plusieurs dépôts distants (remote) ;
- Accessibles via les protocoles :
 - SSH (authentification par clé)
 - HTTP (clonage anonyme)
 - git (en pratique peu utilisé)
- La copie locale est un clone (complet ou incomplet) du dépôt distant ;
- Notions de tags et de branches ;
- Pas de séquence, utilisation de hash pour identifier les commits ;
- Les répertoires ne sont pas versionnés, donc absence de répertoires vides ;
- Possibilité d'étendre la commande (`git flow` par exemple) ;
- <https://git-scm.com/book/fr/v2/>.

Informations générales

- Initialement écrit par Linus Torvalds pour remplacer BitKeeper ;
- Première version le 7 avril 2005 ;
- Dernière version le 2 septembre 2016 (2.10.0) ;
- Possibilité d'avoir plusieurs dépôts distants (remote) ;
- Accessibles via les protocoles :
 - SSH (authentification par clé)
 - HTTP (clonage anonyme)
 - git (en pratique peu utilisé)
- La copie locale est un clone (complet ou incomplet) du dépôt distant ;
- Notions de tags et de branches ;
- Pas de séquence, utilisation de hash pour identifier les commits ;
- Les répertoires ne sont pas versionnés, donc absence de répertoires vides ;
- Possibilité d'étendre la commande (`git flow` par exemple) ;
- <https://git-scm.com/book/fr/v2/>.

Informations générales

- Initialement écrit par Linus Torvalds pour remplacer BitKeeper ;
- Première version le 7 avril 2005 ;
- Dernière version le 2 septembre 2016 (2.10.0) ;
- Possibilité d'avoir plusieurs dépôts distants (remote) ;
- Accessibles via les protocoles :
 - SSH (authentification par clé)
 - HTTP (clonage anonyme)
 - git (en pratique peu utilisé)
- La copie locale est un clone (complet ou incomplet) du dépôt distant ;
- Notions de tags et de branches ;
- Pas de séquence, utilisation de hash pour identifier les commits ;
- Les répertoires ne sont pas versionnés, donc absence de répertoires vides ;
- Possibilité d'étendre la commande (`git flow` par exemple) ;
- <https://git-scm.com/book/fr/v2/>.

Informations générales

- Initialement écrit par Linus Torvalds pour remplacer BitKeeper ;
- Première version le 7 avril 2005 ;
- Dernière version le 2 septembre 2016 (2.10.0) ;
- Possibilité d'avoir plusieurs dépôts distants (remote) ;
- Accessibles via les protocoles :
 - SSH (authentification par clé)
 - HTTP (clonage anonyme)
 - git (en pratique peu utilisé)
- La copie locale est un clone (complet ou incomplet) du dépôt distant ;
- Notions de tags et de branches ;
- Pas de séquence, utilisation de hash pour identifier les commits ;
- Les répertoires ne sont pas versionnés, donc absence de répertoires vides ;
- Possibilité d'étendre la commande (`git flow` par exemple) ;
- <https://git-scm.com/book/fr/v2/>.

Informations générales

- Initialement écrit par Linus Torvalds pour remplacer BitKeeper ;
- Première version le 7 avril 2005 ;
- Dernière version le 2 septembre 2016 (2.10.0) ;
- Possibilité d'avoir plusieurs dépôts distants (remote) ;
- Accessibles via les protocoles :
 - SSH (authentification par clé)
 - HTTP (clonage anonyme)
 - git (en pratique peu utilisé)
- La copie locale est un clone (complet ou incomplet) du dépôt distant ;
- Notions de tags et de branches ;
- Pas de séquence, utilisation de hash pour identifier les commits ;
- Les répertoires ne sont pas versionnés, donc absence de répertoires vides ;
- Possibilité d'étendre la commande (`git flow` par exemple) ;
- <https://git-scm.com/book/fr/v2/>.

Informations générales

- Initialement écrit par Linus Torvalds pour remplacer BitKeeper ;
- Première version le 7 avril 2005 ;
- Dernière version le 2 septembre 2016 (2.10.0) ;
- Possibilité d'avoir plusieurs dépôts distants (remote) ;
- Accessibles via les protocoles :
 - SSH (authentification par clé)
 - HTTP (clonage anonyme)
 - git (en pratique peu utilisé)
- La copie locale est un clone (complet ou incomplet) du dépôt distant ;
- Notions de tags et de branches ;
- Pas de séquence, utilisation de hash pour identifier les commits ;
- Les répertoires ne sont pas versionnés, donc absence de répertoires vides ;
- Possibilité d'étendre la commande (`git flow` par exemple) ;
- <https://git-scm.com/book/fr/v2/>.

Informations générales

- Initialement écrit par Linus Torvalds pour remplacer BitKeeper ;
- Première version le 7 avril 2005 ;
- Dernière version le 2 septembre 2016 (2.10.0) ;
- Possibilité d'avoir plusieurs dépôts distants (remote) ;
- Accessibles via les protocoles :
 - SSH (authentification par clé)
 - HTTP (clonage anonyme)
 - git (en pratique peu utilisé)
- La copie locale est un clone (complet ou incomplet) du dépôt distant ;
- Notions de tags et de branches ;
- Pas de séquence, utilisation de hash pour identifier les commits ;
- Les répertoires ne sont pas versionnés, donc absence de répertoires vides ;
- Possibilité d'étendre la commande (`git flow` par exemple) ;
- <https://git-scm.com/book/fr/v2/>.

Informations générales

- Initialement écrit par Linus Torvalds pour remplacer BitKeeper ;
- Première version le 7 avril 2005 ;
- Dernière version le 2 septembre 2016 (2.10.0) ;
- Possibilité d'avoir plusieurs dépôts distants (remote) ;
- Accessibles via les protocoles :
 - SSH (authentification par clé)
 - HTTP (clonage anonyme)
 - git (en pratique peu utilisé)
- La copie locale est un clone (complet ou incomplet) du dépôt distant ;
- Notions de tags et de branches ;
- Pas de séquence, utilisation de hash pour identifier les commits ;
- Les répertoires ne sont pas versionnés, donc absence de répertoires vides ;
- Possibilité d'étendre la commande (`git flow` par exemple) ;
- <https://git-scm.com/book/fr/v2/>.

Informations générales

- Initialement écrit par Linus Torvalds pour remplacer BitKeeper ;
 - Première version le 7 avril 2005 ;
 - Dernière version le 2 septembre 2016 (2.10.0) ;
 - Possibilité d'avoir plusieurs dépôts distants (remote) ;
 - Accessibles via les protocoles :
 - SSH (authentification par clé)
 - HTTP (clonage anonyme)
 - git (en pratique peu utilisé)
 - La copie locale est un clone (complet ou incomplet) du dépôt distant ;
 - Notions de tags et de branches ;
 - Pas de séquence, utilisation de hash pour identifier les commits ;
 - Les répertoires ne sont pas versionnés, donc absence de répertoires vides ;
 - Possibilité d'étendre la commande (`git flow` par exemple) ;
- <https://git-scm.com/book/fr/v2/>.

Informations générales

- Initialement écrit par Linus Torvalds pour remplacer BitKeeper ;
- Première version le 7 avril 2005 ;
- Dernière version le 2 septembre 2016 (2.10.0) ;
- Possibilité d'avoir plusieurs dépôts distants (remote) ;
- Accessibles via les protocoles :
 - SSH (authentification par clé)
 - HTTP (clonage anonyme)
 - git (en pratique peu utilisé)
- La copie locale est un clone (complet ou incomplet) du dépôt distant ;
- Notions de tags et de branches ;
- Pas de séquence, utilisation de hash pour identifier les commits ;
- Les répertoires ne sont pas versionnés, donc absence de répertoires vides ;
- Possibilité d'étendre la commande (`git flow` par exemple) ;
- <https://git-scm.com/book/fr/v2/>.

Git

Terminologie

blob contenu d'un objet versionné (fichier) ;

commit révision, hash et commit parent ;

branche version active du projet ;

tag étiquette sur un point donné de l'historique, permet de figer des releases ;

merge fusion entre deux branches ;

rebase changement de la base de la branche par une autre branche ;

staging aussi appelé cache, contient les objets et états servant au commit ;

stash pile dans laquelle le développeur peut stocker ses modifications sans les commiter, souvent avant un pull pour fusionner ses modifications a posteriori.

Git

Terminologie

blob contenu d'un objet versionné (fichier) ;

commit révision, hash et commit parent ;

branche version active du projet ;

tag étiquette sur un point donné de l'historique, permet de figer des releases ;

merge fusion entre deux branches ;

rebase changement de la base de la branche par une autre branche ;

staging aussi appelé cache, contient les objets et états servant au commit ;

stash pile dans laquelle le développeur peut stocker ses modifications sans les commiter, souvent avant un pull pour fusionner ses modifications a posteriori.

Git

Terminologie

blob contenu d'un objet versionné (fichier) ;

commit révision, hash et commit parent ;

branche version active du projet ;

tag étiquette sur un point donné de l'historique, permet de figer des releases ;

merge fusion entre deux branches ;

rebase changement de la base de la branche par une autre branche ;

staging aussi appelé cache, contient les objets et états servant au commit ;

stash pile dans laquelle le développeur peut stocker ses modifications sans les commiter, souvent avant un pull pour fusionner ses modifications a posteriori.

Git

Terminologie

blob contenu d'un objet versionné (fichier) ;

commit révision, hash et commit parent ;

branche version active du projet ;

tag étiquette sur un point donné de l'historique, permet de figer des releases ;

merge fusion entre deux branches ;

rebase changement de la base de la branche par une autre branche ;

staging aussi appelé cache, contient les objets et états servant au commit ;

stash pile dans laquelle le développeur peut stocker ses modifications sans les commiter, souvent avant un pull pour fusionner ses modifications a posteriori.

Git

Terminologie

blob contenu d'un objet versionné (fichier) ;

commit révision, hash et commit parent ;

branche version active du projet ;

tag étiquette sur un point donné de l'historique, permet de figer des releases ;

merge fusion entre deux branches ;

rebase changement de la base de la branche par une autre branche ;

staging aussi appelé cache, contient les objets et états servant au commit ;

stash pile dans laquelle le développeur peut stocker ses modifications sans les commiter, souvent avant un pull pour fusionner ses modifications a posteriori.

Git

Terminologie

blob contenu d'un objet versionné (fichier) ;

commit révision, hash et commit parent ;

branche version active du projet ;

tag étiquette sur un point donné de l'historique, permet de figer des releases ;

merge fusion entre deux branches ;

rebase changement de la base de la branche par une autre branche ;

staging aussi appelé cache, contient les objets et états servant au commit ;

stash pile dans laquelle le développeur peut stocker ses modifications sans les commiter, souvent avant un pull pour fusionner ses modifications a posteriori.

Git

Terminologie

blob contenu d'un objet versionné (fichier) ;

commit révision, hash et commit parent ;

branche version active du projet ;

tag étiquette sur un point donné de l'historique, permet de figer des releases ;

merge fusion entre deux branches ;

rebase changement de la base de la branche par une autre branche ;

staging aussi appelé cache, contient les objets et états servant au commit ;

stash pile dans laquelle le développeur peut stocker ses modifications sans les commiter, souvent avant un pull pour fusionner ses modifications a posteriori.

Git

Terminologie

blob contenu d'un objet versionné (fichier) ;

commit révision, hash et commit parent ;

branche version active du projet ;

tag étiquette sur un point donné de l'historique, permet de figer des releases ;

merge fusion entre deux branches ;

rebase changement de la base de la branche par une autre branche ;

staging aussi appelé cache, contient les objets et états servant au commit ;

stash pile dans laquelle le développeur peut stocker ses modifications sans les commiter, souvent avant un pull pour fusionner ses modifications a posteriori.

Git

Workflow

`git clone` clone un dépôt distant en local ;

`édition` modifie les fichiers ;

`git stash` sauvegarde les modifications dans stash ;

`git pull` récupère les révisions distantes ;

`git stash pop` réapplique les modifications sur la copie de travail à jour ;

`git mergetool` lance l'outil de gestion des conflits (lors d'un pull, ou d'un merge) ;

`git add` ajoute un fichier modifié ou non versionné dans le cache ;

`git rm` supprime un fichier et le met comme tel dans le cache ;

`git commit` crée une révision locale d'après le cache ;

`git push` pousse les révisions locales ;

Git

Workflow

`git clone` clone un dépôt distant en local ;

`édition` modifie les fichiers ;

`git stash` sauvegarde les modifications dans stash ;

`git pull` récupère les révisions distantes ;

`git stash pop` réapplique les modifications sur la copie de travail à jour ;

`git mergetool` lance l'outil de gestion des conflits (lors d'un pull, ou d'un merge) ;

`git add` ajoute un fichier modifié ou non versionné dans le cache ;

`git rm` supprime un fichier et le met comme tel dans le cache ;

`git commit` crée une révision locale d'après le cache ;

`git push` pousse les révisions locales ;

Git

Workflow

`git clone` clone un dépôt distant en local ;

`édition` modifie les fichiers ;

`git stash` sauvegarde les modifications dans stash ;

`git pull` récupère les révisions distantes ;

`git stash pop` réapplique les modifications sur la copie de travail à jour ;

`git mergetool` lance l'outil de gestion des conflits (lors d'un pull, ou d'un merge) ;

`git add` ajoute un fichier modifié ou non versionné dans le cache ;

`git rm` supprime un fichier et le met comme tel dans le cache ;

`git commit` crée une révision locale d'après le cache ;

`git push` pousse les révisions locales ;

Git

Workflow

`git clone` clone un dépôt distant en local ;

`édition` modifie les fichiers ;

`git stash` sauvegarde les modifications dans stash ;

`git pull` récupère les révisions distantes ;

`git stash pop` réapplique les modifications sur la copie de travail à jour ;

`git mergetool` lance l'outil de gestion des conflits (lors d'un pull, ou d'un merge) ;

`git add` ajoute un fichier modifié ou non versionné dans le cache ;

`git rm` supprime un fichier et le met comme tel dans le cache ;

`git commit` crée une révision locale d'après le cache ;

`git push` pousse les révisions locales ;

Git

Workflow

`git clone` clone un dépôt distant en local ;

`édition` modifie les fichiers ;

`git stash` sauvegarde les modifications dans stash ;

`git pull` récupère les révisions distantes ;

`git stash pop` réapplique les modifications sur la copie de travail à jour ;

`git mergetool` lance l'outil de gestion des conflits (lors d'un pull, ou d'un merge) ;

`git add` ajoute un fichier modifié ou non versionné dans le cache ;

`git rm` supprime un fichier et le met comme tel dans le cache ;

`git commit` crée une révision locale d'après le cache ;

`git push` pousse les révisions locales ;

Git

Workflow

`git clone` clone un dépôt distant en local ;

`édition` modifie les fichiers ;

`git stash` sauvegarde les modifications dans stash ;

`git pull` récupère les révisions distantes ;

`git stash pop` réapplique les modifications sur la copie de travail à jour ;

`git mergetool` lance l'outil de gestion des conflits (lors d'un pull, ou d'un merge) ;

`git add` ajoute un fichier modifié ou non versionné dans le cache ;

`git rm` supprime un fichier et le met comme tel dans le cache ;

`git commit` crée une révision locale d'après le cache ;

`git push` pousse les révisions locales ;

Git

Workflow

`git clone` clone un dépôt distant en local ;

`édition` modifie les fichiers ;

`git stash` sauvegarde les modifications dans stash ;

`git pull` récupère les révisions distantes ;

`git stash pop` réapplique les modifications sur la copie de travail à jour ;

`git mergetool` lance l'outil de gestion des conflits (lors d'un pull, ou d'un merge) ;

`git add` ajoute un fichier modifié ou non versionné dans le cache ;

`git rm` supprime un fichier et le met comme tel dans le cache ;

`git commit` crée une révision locale d'après le cache ;

`git push` pousse les révisions locales ;

Git

Workflow

`git clone` clone un dépôt distant en local ;

`édition` modifie les fichiers ;

`git stash` sauvegarde les modifications dans stash ;

`git pull` récupère les révisions distantes ;

`git stash pop` réapplique les modifications sur la copie de travail à jour ;

`git mergetool` lance l'outil de gestion des conflits (lors d'un pull, ou d'un merge) ;

`git add` ajoute un fichier modifié ou non versionné dans le cache ;

`git rm` supprime un fichier et le met comme tel dans le cache ;

`git commit` crée une révision locale d'après le cache ;

`git push` pousse les révisions locales ;

Git

Workflow

`git clone` clone un dépôt distant en local ;

`édition` modifie les fichiers ;

`git stash` sauvegarde les modifications dans stash ;

`git pull` récupère les révisions distantes ;

`git stash pop` réapplique les modifications sur la copie de travail à jour ;

`git mergetool` lance l'outil de gestion des conflits (lors d'un pull, ou d'un merge) ;

`git add` ajoute un fichier modifié ou non versionné dans le cache ;

`git rm` supprime un fichier et le met comme tel dans le cache ;

`git commit` crée une révision locale d'après le cache ;

`git push` pousse les révisions locales ;

Git

Workflow

`git clone` clone un dépôt distant en local ;

`édition` modifie les fichiers ;

`git stash` sauvegarde les modifications dans stash ;

`git pull` récupère les révisions distantes ;

`git stash pop` réapplique les modifications sur la copie de travail à jour ;

`git mergetool` lance l'outil de gestion des conflits (lors d'un pull, ou d'un merge) ;

`git add` ajoute un fichier modifié ou non versionné dans le cache ;

`git rm` supprime un fichier et le met comme tel dans le cache ;

`git commit` crée une révision locale d'après le cache ;

`git push` pousse les révisions locales ;

Git

Workflow – suite

`git branch` manipule les branches ;

`git checkout` change de branche, ou restaure un fichier de travail ;

`git reset` réinitialise HEAD à un état différent ;

`git merge` fusionne une branche dans la branche courante ;

`git mergetool` lance l'outil de gestion des conflits si nécessaire ;

`git rebase` reconstruit l'historique de la branche par rapport à une autre branche ;

`git tag` manipule les tags, possibilité de le signer avec une clé PGP.

Git

Workflow – suite

`git branch` manipule les branches ;

`git checkout` change de branche, ou restaure un fichier de travail ;

`git reset` réinitialise HEAD à un état différent ;

`git merge` fusionne une branche dans la branche courante ;

`git mergetool` lance l'outil de gestion des conflits si nécessaire ;

`git rebase` reconstruit l'historique de la branche par rapport à une autre branche ;

`git tag` manipule les tags, possibilité de le signer avec une clé PGP.

Git

Workflow – suite

`git branch` manipule les branches ;

`git checkout` change de branche, ou restaure un fichier de travail ;

`git reset` réinitialise HEAD à un état différent ;

`git merge` fusionne une branche dans la branche courante ;

`git mergetool` lance l'outil de gestion des conflits si nécessaire ;

`git rebase` reconstruit l'historique de la branche par rapport à une autre branche ;

`git tag` manipule les tags, possibilité de le signer avec une clé PGP.

Git

Workflow – suite

`git branch` manipule les branches ;

`git checkout` change de branche, ou restaure un fichier de travail ;

`git reset` réinitialise HEAD à un état différent ;

`git merge` fusionne une branche dans la branche courante ;

`git mergetool` lance l'outil de gestion des conflits si nécessaire ;

`git rebase` reconstruit l'historique de la branche par rapport à une autre branche ;

`git tag` manipule les tags, possibilité de le signer avec une clé PGP.

Git

Workflow – suite

`git branch` manipule les branches ;

`git checkout` change de branche, ou restaure un fichier de travail ;

`git reset` réinitialise HEAD à un état différent ;

`git merge` fusionne une branche dans la branche courante ;

`git mergetool` lance l'outil de gestion des conflits si nécessaire ;

`git rebase` reconstruit l'historique de la branche par rapport à une autre branche ;

`git tag` manipule les tags, possibilité de le signer avec une clé PGP.

Git

Workflow – suite

`git branch` manipule les branches ;

`git checkout` change de branche, ou restaure un fichier de travail ;

`git reset` réinitialise HEAD à un état différent ;

`git merge` fusionne une branche dans la branche courante ;

`git mergetool` lance l'outil de gestion des conflits si nécessaire ;

`git rebase` reconstruit l'historique de la branche par rapport à une autre branche ;

`git tag` manipule les tags, possibilité de le signer avec une clé PGP.

Git

Workflow – suite

`git branch` manipule les branches ;

`git checkout` change de branche, ou restaure un fichier de travail ;

`git reset` réinitialise HEAD à un état différent ;

`git merge` fusionne une branche dans la branche courante ;

`git mergetool` lance l'outil de gestion des conflits si nécessaire ;

`git rebase` reconstruit l'historique de la branche par rapport à une autre branche ;

`git tag` manipule les tags, possibilité de le signer avec une clé PGP.

Autres commandes utiles

`git blame` affiche le commit et l'auteur de chaque ligne d'un fichier ;

`git diff` affiche les différences sur la copie locale, ou entre deux révisions ;

- Permet de générer un fichier patch, avec l'option `--patch` ou `-p` ou encore `-u` ;

`git submodule` gère les sous-modules ;

`git help <command>` RTFM ;-) Attention, très (trop) complet.

Autres commandes utiles

`git blame` affiche le commit et l'auteur de chaque ligne d'un fichier ;

`git diff` affiche les différences sur la copie locale, ou entre deux révisions ;

- Permet de générer un fichier patch, avec l'option `--patch` ou `-p` ou encore `-u` ;

`git submodule` gère les sous-modules ;

`git help <command>` RTFM ;-) Attention, très (trop) complet.

Autres commandes utiles

`git blame` affiche le commit et l'auteur de chaque ligne d'un fichier ;

`git diff` affiche les différences sur la copie locale, ou entre deux révisions ;

- Permet de générer un fichier patch, avec l'option `--patch` ou `-p` ou encore `-u` ;

`git submodule` gère les sous-modules ;

`git help <command>` RTFM ;-) Attention, très (trop) complet.

Autres commandes utiles

`git blame` affiche le commit et l'auteur de chaque ligne d'un fichier ;

`git diff` affiche les différences sur la copie locale, ou entre deux révisions ;

- Permet de générer un fichier patch, avec l'option `--patch` ou `-p` ou encore `-u` ;

`git submodule` gère les sous-modules ;

`git help <command>` RTFM ;-) Attention, très (trop) complet.

Autres commandes utiles

`git blame` affiche le commit et l'auteur de chaque ligne d'un fichier ;

`git diff` affiche les différences sur la copie locale, ou entre deux révisions ;

- Permet de générer un fichier patch, avec l'option `--patch` ou `-p` ou encore `-u` ;

`git submodule` gère les sous-modules ;

`git help <command>` RTFM ;) Attention, très (trop) complet.

Autres commandes utiles

`git blame` affiche le commit et l'auteur de chaque ligne d'un fichier ;

`git diff` affiche les différences sur la copie locale, ou entre deux révisions ;

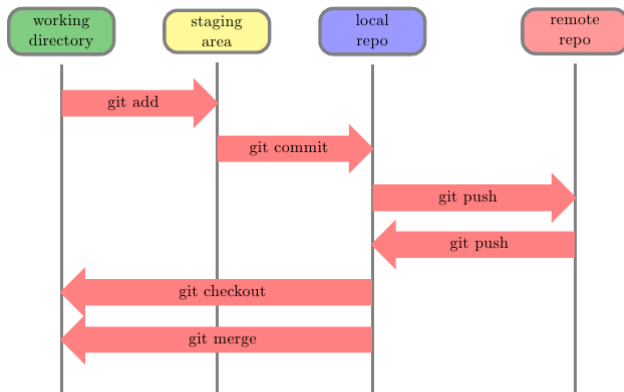
- Permet de générer un fichier patch, avec l'option `--patch` ou `-p` ou encore `-u` ;

`git submodule` gère les sous-modules ;

`git help <command>` RTFM ;-) Attention, très (trop) complet.

Workflow simple

FIGURE – Workflow de développement local avec Git.



Source : <http://tex.stackexchange.com/questions/70320/workflow-diagram>

git status

Exemple

```
1  On branch master
2  Your branch is up-to-date with 'origin/master'.
3  Changes to be committed:
4    (use "git reset HEAD <file>..." to unstage)
5
6      modified:   mailcap
7
8  Changes not staged for commit:
9    (use "git add <file>..." to update what will be committed)
10   (use "git checkout -- <file>..." to discard changes in working directory)
11
12      modified:   caffrc
13      modified:   config/i3pystatus/config.py
14      modified:   dotfilesrc
15      modified:   mutt-profiles/alexis@lahouze.org/profile
16      modified:   mutt-profiles/alexis@sysnove.fr/profile
17      modified:   offlineimaprc
18      modified:   services/dunst/log/run
19      modified:   tmux.conf
20      modified:   vimrc
21      modified:   zshrc
22
23  Untracked files:
24    (use "git add <file>..." to include in what will be committed)
25
26      config/flexget/
27      imapnotify.js
```

Sous-modules

- Référence de dépôts git dans des sous-répertoires du dépôt ;
- Ne se mettent pas à jour automatiquement ;
- Enregistré dans `.gitmodules` ;
- Workflow :
 - `git submodule add`
 - `git commit -m "..."`
 - `git submodule update`
- <https://git-scm.com/book/fr/v1/Utilitaires-Git-Sous-modules>

Sous-modules

- Référence de dépôts git dans des sous-répertoires du dépôt ;
- Ne se mettent pas à jour automatiquement ;
- Enregistré dans `.gitmodules` ;
- Workflow :
 - `git submodule add`
 - `git commit -m "..."`
 - `git submodule update`
- <https://git-scm.com/book/fr/v1/Utilitaires-Git-Sous-modules>

Sous-modules

- Référence de dépôts git dans des sous-répertoires du dépôt ;
- Ne se mettent pas à jour automatiquement ;
- Enregistré dans `.gitmodules` ;
- Workflow :
 - `git submodule add`
 - `git commit -m "..."`
 - `git submodule update`
- <https://git-scm.com/book/fr/v1/Utilitaires-Git-Sous-modules>

Sous-modules

- Référence de dépôts git dans des sous-répertoires du dépôt ;
- Ne se mettent pas à jour automatiquement ;
- Enregistré dans `.gitmodules` ;
- Workflow :
 - `git submodule add`
 - `git commit -m "..."`
 - `git submodule update`
- <https://git-scm.com/book/fr/v1/Utilitaires-Git-Sous-modules>

Sous-modules

- Référence de dépôts git dans des sous-répertoires du dépôt ;
- Ne se mettent pas à jour automatiquement ;
- Enregistré dans `.gitmodules` ;
- Workflow :
 - `git submodule add`
 - `git commit -m "..."`
 - `git submodule update`
- <https://git-scm.com/book/fr/v1/Utilitaires-Git-Sous-modules>

Sous-modules

- Référence de dépôts git dans des sous-répertoires du dépôt ;
- Ne se mettent pas à jour automatiquement ;
- Enregistré dans `.gitmodules` ;
- Workflow :
 - `git submodule add`
 - `git commit -m "..."`
 - `git submodule update`
- <https://git-scm.com/book/fr/v1/Utilitaires-Git-Sous-modules>

Sous-modules

- Référence de dépôts git dans des sous-répertoires du dépôt ;
 - Ne se mettent pas à jour automatiquement ;
 - Enregistré dans `.gitmodules` ;
 - Workflow :
 - `git submodule add`
 - `git commit -m "..."`
 - `git submodule update`
- <https://git-scm.com/book/fr/v1/Utilitaires-Git-Sous-modules>

Sous-modules

- Référence de dépôts git dans des sous-répertoires du dépôt ;
- Ne se mettent pas à jour automatiquement ;
- Enregistré dans `.gitmodules` ;
- Workflow :
 - `git submodule add`
 - `git commit -m "..."`
 - `git submodule update`
- <https://git-scm.com/book/fr/v1/Utilitaires-Git-Sous-modules>

Git Flow

- Décrit par Vincent Driesse sur <http://nvie.com/posts/a-successful-git-branching-model/>
- Process de développement et de publication standardisé ;
- Commande `git flow` (<https://github.com/nvie/gitflow>).
- Antisèche : <http://danielkummer.github.io/git-flow-cheatsheet/>

Git Flow

- Décrit par Vincent Driesse sur <http://nvie.com/posts/a-successful-git-branching-model/>
- Process de développement et de publication standardisé ;
- Commande `git flow` (<https://github.com/nvie/gitflow>).
- Antisèche : <http://danielkummer.github.io/git-flow-cheatsheet/>

Git Flow

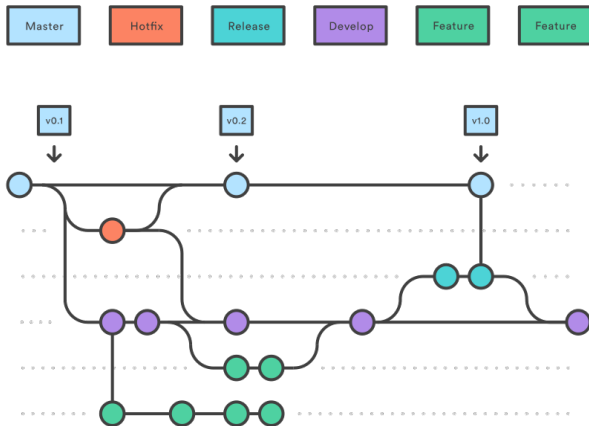
- Décrit par Vincent Driesse sur <http://nvie.com/posts/a-successful-git-branching-model/>
- Process de développement et de publication standardisé ;
- Commande `git flow` (<https://github.com/nvie/gitflow>).
- Antisèche : <http://danielkummer.github.io/git-flow-cheatsheet/>

Git Flow

- Décrit par Vincent Driesse sur <http://nvie.com/posts/a-successful-git-branching-model/>
- Process de développement et de publication standardisé ;
- Commande `git flow` (<https://github.com/nvie/gitflow>).
- Antisèche : <http://danielkummer.github.io/git-flow-cheatsheet/>

Git Flow

FIGURE – Workflow complet avec git flow.



Source : <https://www.atlassian.com/git/tutorials/comparing-workflows/gitflow-workflow>

git svn

- Possibilité pour Git de se connecter à un dépôt Subversion ;
- Permet d'effectuer des commits locaux ;
- Ne permet pas de décentraliser le développement.
- <https://git-scm.com/book/fr/v2/Git-et-les-autres-systèmes-Git-comme-client#Git-et-Subversion>

git svn

- Possibilité pour Git de se connecter à un dépôt Subversion ;
- Permet d'effectuer des commits locaux ;
- Ne permet pas de décentraliser le développement.
- <https://git-scm.com/book/fr/v2/Git-et-les-autres-systèmes-Git-comme-client#Git-et-Subversion>

git svn

- Possibilité pour Git de se connecter à un dépôt Subversion ;
 - Permet d'effectuer des commits locaux ;
 - Ne permet pas de décentraliser le développement.
- <https://git-scm.com/book/fr/v2/Git-et-les-autres-systèmes-Git-comme-client#Git-et-Subversion>

git svn

- Possibilité pour Git de se connecter à un dépôt Subversion ;
- Permet d'effectuer des commits locaux ;
- Ne permet pas de décentraliser le développement.
- <https://git-scm.com/book/fr/v2/Git-et-les-autres-systèmes-Git-comme-client#Git-et-Subversion>

git svn – utilisation de base

`git svn clone` équivalent à `svn checkout` ;
`git add`, `git commit` travailler localement ;
`git svn rebase` équivalent à `svn update` ;
`git svn dcommit` équivalent à `svn commit` pour chaque commit local ;
`git svn branch` manipule les branches et les tags (`--tag` ou `--t`) ;
 `git svn log` affiche l'historique ;
`git svn blame` affiche l'auteur et la révision de chaque ligne d'un fichier ;
`git svn reset` permet de revenir à une révision spécifique ;
 `git svn help` RTFM;-)

git svn – utilisation de base

`git svn clone` équivalent à `svn checkout` ;
`git add`, `git commit` travailler localement ;
`git svn rebase` équivalent à `svn update` ;
`git svn dcommit` équivalent à `svn commit` pour chaque commit local ;
`git svn branch` manipule les branches et les tags (`--tag` ou `--t`) ;
 `git svn log` affiche l'historique ;
`git svn blame` affiche l'auteur et la révision de chaque ligne d'un fichier ;
`git svn reset` permet de revenir à une révision spécifique ;
 `git svn help` RTFM;-)

git svn – utilisation de base

`git svn clone` équivalent à `svn checkout` ;
`git add`, `git commit` travailler localement ;
`git svn rebase` équivalent à `svn update` ;
`git svn dcommit` équivalent à `svn commit` pour chaque commit local ;
`git svn branch` manipule les branches et les tags (`--tag` ou `--t`) ;
 `git svn log` affiche l'historique ;
`git svn blame` affiche l'auteur et la révision de chaque ligne d'un fichier ;
`git svn reset` permet de revenir à une révision spécifique ;
 `git svn help` RTFM;-)

git svn – utilisation de base

`git svn clone` équivalent à `svn checkout` ;
`git add`, `git commit` travailler localement ;
`git svn rebase` équivalent à `svn update` ;
`git svn dcommit` équivalent à `svn commit` pour chaque commit local ;
`git svn branch` manipule les branches et les tags (`--tag` ou `--t`) ;
 `git svn log` affiche l'historique ;
`git svn blame` affiche l'auteur et la révision de chaque ligne d'un fichier ;
`git svn reset` permet de revenir à une révision spécifique ;
 `git svn help` RTFM;-)

git svn – utilisation de base

`git svn clone` équivalent à `svn checkout` ;
`git add`, `git commit` travailler localement ;
`git svn rebase` équivalent à `svn update` ;
`git svn dcommit` équivalent à `svn commit` pour chaque commit local ;
`git svn branch` manipule les branches et les tags (`--tag` ou `--t`) ;
`git svn log` affiche l'historique ;
`git svn blame` affiche l'auteur et la révision de chaque ligne d'un fichier ;
`git svn reset` permet de revenir à une révision spécifique ;
`git svn help` RTFM;-)

git svn – utilisation de base

`git svn clone` équivalent à `svn checkout` ;
`git add`, `git commit` travailler localement ;
`git svn rebase` équivalent à `svn update` ;
`git svn dcommit` équivalent à `svn commit` pour chaque commit local ;
`git svn branch` manipule les branches et les tags (`--tag` ou `--t`) ;
 `git svn log` affiche l'historique ;
`git svn blame` affiche l'auteur et la révision de chaque ligne d'un fichier ;
`git svn reset` permet de revenir à une révision spécifique ;
 `git svn help` RTFM;-)

git svn – utilisation de base

`git svn clone` équivalent à `svn checkout` ;
`git add`, `git commit` travailler localement ;
`git svn rebase` équivalent à `svn update` ;
`git svn dcommit` équivalent à `svn commit` pour chaque commit local ;
`git svn branch` manipule les branches et les tags (`--tag` ou `--t`) ;
 `git svn log` affiche l'historique ;
`git svn blame` affiche l'auteur et la révision de chaque ligne d'un fichier ;
`git svn reset` permet de revenir à une révision spécifique ;
 `git svn help` RTFM;-)

git svn – utilisation de base

`git svn clone` équivalent à `svn checkout` ;
`git add`, `git commit` travailler localement ;
`git svn rebase` équivalent à `svn update` ;
`git svn dcommit` équivalent à `svn commit` pour chaque commit local ;
`git svn branch` manipule les branches et les tags (`--tag` ou `--t`) ;
 `git svn log` affiche l'historique ;
`git svn blame` affiche l'auteur et la révision de chaque ligne d'un fichier ;
`git svn reset` permet de revenir à une révision spécifique ;
 `git svn help` RTFM;-)

git svn – utilisation de base

`git svn clone` équivalent à `svn checkout` ;
`git add`, `git commit` travailler localement ;
`git svn rebase` équivalent à `svn update` ;
`git svn dcommit` équivalent à `svn commit` pour chaque commit local ;
`git svn branch` manipule les branches et les tags (`--tag` ou `--t`) ;
 `git svn log` affiche l'historique ;
`git svn blame` affiche l'auteur et la révision de chaque ligne d'un fichier ;
`git svn reset` permet de revenir à une révision spécifique ;
 `git svn help` RTFM ;-)

Gestion de la Documentation

- 1 Introduction
- 2 Outils de Développement
- 3 Gestion de la Documentation**
- 4 Outils de Communication
- 5 Forges de développement
- 6 Fonctionnement de certains projets

Outils

Wiki Utilisé surtout pour la documentation technique ;

Pad Édition collaborative de texte (Etherpad) :

- Utilisés souvent pour des travaux ponctuels ;
- <https://framapad.org> ;
- <https://pad.ilico.org> :

Cloud Synchronisation, partage de fichiers, parfois édition en ligne (NextCloud, Cozy cloud) :

- Autohébergés la plupart du temps ;
- <https://framadrive.org> ;
- Google Drive, MS Onedrive, etc. (non libres).

Outils

Wiki Utilisé surtout pour la documentation technique ;

Pad Édition collaborative de texte (Etherpad) :

- Utilisés souvent pour des travaux ponctuels ;
- <https://framapad.org> ;
- <https://pad.ilico.org> :

Cloud Synchronisation, partage de fichiers, parfois édition en ligne (NextCloud, Cozy cloud) :

- Autohébergés la plupart du temps ;
- <https://framadrive.org> ;
- Google Drive, MS Onedrive, etc. (non libres).

Outils

Wiki Utilisé surtout pour la documentation technique ;

Pad Édition collaborative de texte (Etherpad) :

- Utilisés souvent pour des travaux ponctuels ;
- <https://framapad.org> ;
- <https://pad.ilico.org> :

Cloud Synchronisation, partage de fichiers, parfois édition en ligne (NextCloud, Cozy cloud) :

- Autohébergés la plupart du temps ;
- <https://framadrive.org> ;
- Google Drive, MS Onedrive, etc. (non libres).

Outils

Wiki Utilisé surtout pour la documentation technique ;

Pad Édition collaborative de texte (Etherpad) :

- Utilisés souvent pour des travaux ponctuels ;
- <https://framapad.org> ;
- <https://pad.ilico.org> :

Cloud Synchronisation, partage de fichiers, parfois édition en ligne (NextCloud, Cozy cloud) :

- Autohébergés la plupart du temps ;
- <https://framadrive.org> ;
- Google Drive, MS Onedrive, etc. (non libres).

Outils

Wiki Utilisé surtout pour la documentation technique ;

Pad Édition collaborative de texte (Etherpad) :

- Utilisés souvent pour des travaux ponctuels ;
- <https://framapad.org> ;
- <https://pad.ilico.org> :

Cloud Synchronisation, partage de fichiers, parfois édition en ligne (NextCloud, Cozy cloud) :

- Autohébergés la plupart du temps ;
- <https://framadrive.org> ;
- Google Drive, MS Onedrive, etc. (non libres).

Outils

Wiki Utilisé surtout pour la documentation technique ;

Pad Édition collaborative de texte (Etherpad) :

- Utilisés souvent pour des travaux ponctuels ;
- <https://framapad.org> ;
- <https://pad.ilico.org> :

Cloud Synchronisation, partage de fichiers, parfois édition en ligne (NextCloud, Cozy cloud) :

- Autohébergés la plupart du temps ;
- <https://framadrive.org> ;
- Google Drive, MS Onedrive, etc. (non libres).

Outils

Wiki Utilisé surtout pour la documentation technique ;

Pad Édition collaborative de texte (Etherpad) :

- Utilisés souvent pour des travaux ponctuels ;
- <https://framapad.org> ;
- <https://pad.ilico.org> :

Cloud Synchronisation, partage de fichiers, parfois édition en ligne (NextCloud, Cozy cloud) :

- Autohébergés la plupart du temps ;
- <https://framadrive.org> ;
- Google Drive, MS Onedrive, etc. (non libres).

Outils

Wiki Utilisé surtout pour la documentation technique ;

Pad Édition collaborative de texte (Etherpad) :

- Utilisés souvent pour des travaux ponctuels ;
- <https://framapad.org> ;
- <https://pad.ilico.org> :

Cloud Synchronisation, partage de fichiers, parfois édition en ligne (NextCloud, Cozy cloud) :

- Autohébergés la plupart du temps ;
- <https://framadrive.org> ;
- Google Drive, MS Onedrive, etc. (non libres).

Outils

Wiki Utilisé surtout pour la documentation technique ;

Pad Édition collaborative de texte (Etherpad) :

- Utilisés souvent pour des travaux ponctuels ;
- <https://framapad.org> ;
- <https://pad.ilico.org> :

Cloud Synchronisation, partage de fichiers, parfois édition en ligne (NextCloud, Cozy cloud) :

- Autohébergés la plupart du temps ;
- <https://framadrive.org> ;
- Google Drive, MS Onedrive, etc. (non libres).

Outils de Communication

- 1 Introduction
- 2 Outils de Développement
- 3 Gestion de la Documentation
- 4 Outils de Communication**
 - Notions générales
 - Communication synchrone
 - Communication asynchrone
- 5 Forges de développement
- 6 Fonctionnement de certains projets

Notions générales

- 1 Introduction
- 2 Outils de Développement
- 3 Gestion de la Documentation
- 4 Outils de Communication**
 - **Notions générales**
 - Communication synchrone
 - Communication asynchrone
- 5 Forges de développement
- 6 Fonctionnement de certains projets

Netiquette

- Règles de bonne conduite sur les Internets
 - Network Etiquette
 - RFC1855 : <http://fgouget.free.fr/netiquette/rfc1855-fr.html>
 - Communications un à un (mail, messagerie instantanée)
 - Communications un à plusieurs (listes de diffusion, forums)
 - Forme
 - Écrire en entier (pas de langage SMS) ;
 - Écrire en minuscules, ÉCRIRE EN MAJUSCULES REVIENT À CRIER ;
 - Réponses sous questions ;
 - Ne pas dépasser 65 caractères dans un mail ;
 - Etc.
- Fond
 - Politesse
 - Respect

Netiquette

- Règles de bonne conduite sur les Internets
- Network Etiquette
- RFC1855 : <http://fgouget.free.fr/netiquette/rfc1855-fr.html>
- Communications un à un (mail, messagerie instantanée)
- Communications un à plusieurs (listes de diffusion, forums)
- Forme
 - Écrire en entier (pas de langage SMS) ;
 - Écrire en minuscules, ÉCRIRE EN MAJUSCULES REVIENT À CRIER ;
 - Réponses sous questions ;
 - Ne pas dépasser 65 caractères dans un mail ;
 - Etc.
- Fond
 - Politesse
 - Respect

Netiquette

- Règles de bonne conduite sur les Internets
- Network Etiquette
- RFC1855 : <http://fgouget.free.fr/netiquette/rfc1855-fr.html>
- Communications un à un (mail, messagerie instantanée)
- Communications un à plusieurs (listes de diffusion, forums)
- Forme
 - Écrire en entier (pas de langage SMS) ;
 - Écrire en minuscules, ÉCRIRE EN MAJUSCULES REVIENT À CRIER ;
 - Réponses sous questions ;
 - Ne pas dépasser 65 caractères dans un mail ;
 - Etc.
- Fond
 - Politesse
 - Respect

Netiquette

- Règles de bonne conduite sur les Internets
- Network Etiquette
- RFC1855 : <http://fgouget.free.fr/netiquette/rfc1855-fr.html>
- Communications un à un (mail, messagerie instantanée)
- Communications un à plusieurs (listes de diffusion, forums)
- Forme
 - Écrire en entier (pas de langage SMS) ;
 - Écrire en minuscules, ÉCRIRE EN MAJUSCULES REVIENT À CRIER ;
 - Réponses sous questions ;
 - Ne pas dépasser 65 caractères dans un mail ;
 - Etc.
- Fond
 - Politesse
 - Respect

Netiquette

- Règles de bonne conduite sur les Internets
- Network Etiquette
- RFC1855 : <http://fgouget.free.fr/netiquette/rfc1855-fr.html>
- Communications un à un (mail, messagerie instantanée)
- Communications un à plusieurs (listes de diffusion, forums)
- Forme
 - Écrire en entier (pas de langage SMS) ;
 - Écrire en minuscules, ÉCRIRE EN MAJUSCULES REVIENT À CRIER ;
 - Réponses sous questions ;
 - Ne pas dépasser 65 caractères dans un mail ;
 - Etc.
- Fond
 - Politesse
 - Respect

Netiquette

- Règles de bonne conduite sur les Internets
- Network Etiquette
- RFC1855 : <http://fgouget.free.fr/netiquette/rfc1855-fr.html>
- Communications un à un (mail, messagerie instantanée)
- Communications un à plusieurs (listes de diffusion, forums)
- Forme
 - Écrire en entier (pas de langage SMS) ;
 - Écrire en minuscules, ÉCRIRE EN MAJUSCULES REVIENT À CRIER ;
 - Réponses sous questions ;
 - Ne pas dépasser 65 caractères dans un mail ;
 - Etc.
- Fond
 - Politesse
 - Respect

Netiquette

- Règles de bonne conduite sur les Internets
- Network Etiquette
- RFC1855 : <http://fgouget.free.fr/netiquette/rfc1855-fr.html>
- Communications un à un (mail, messagerie instantanée)
- Communications un à plusieurs (listes de diffusion, forums)
- Forme
 - Écrire en entier (pas de langage SMS) ;
 - Écrire en minuscules, ÉCRIRE EN MAJUSCULES REVIENT À CRIER ;
 - Réponses sous questions ;
 - Ne pas dépasser 65 caractères dans un mail ;
 - Etc.
- Fond
 - Politesse
 - Respect

Netiquette

- Règles de bonne conduite sur les Internets
- Network Etiquette
- RFC1855 : <http://fgouget.free.fr/netiquette/rfc1855-fr.html>
- Communications un à un (mail, messagerie instantanée)
- Communications un à plusieurs (listes de diffusion, forums)
- Forme
 - Écrire en entier (pas de langage SMS) ;
 - Écrire en minuscules, ÉCRIRE EN MAJUSCULES REVIENT À CRIER ;
 - Réponses sous questions ;
 - Ne pas dépasser 65 caractères dans un mail ;
 - Etc.
- Fond
 - Politesse
 - Respect

Netiquette

- Règles de bonne conduite sur les Internets
- Network Etiquette
- RFC1855 : <http://fgouget.free.fr/netiquette/rfc1855-fr.html>
- Communications un à un (mail, messagerie instantanée)
- Communications un à plusieurs (listes de diffusion, forums)
- Forme
 - Écrire en entier (pas de langage SMS) ;
 - Écrire en minuscules, ÉCRIRE EN MAJUSCULES REVIENT À CRIER ;
 - Réponses sous questions ;
 - Ne pas dépasser 65 caractères dans un mail ;
 - Etc.
- Fond
 - Politesse
 - Respect

Netiquette

- Règles de bonne conduite sur les Internets
- Network Etiquette
- RFC1855 : <http://fgouget.free.fr/netiquette/rfc1855-fr.html>
- Communications un à un (mail, messagerie instantanée)
- Communications un à plusieurs (listes de diffusion, forums)
- Forme
 - Écrire en entier (pas de langage SMS) ;
 - Écrire en minuscules, ÉCRIRE EN MAJUSCULES REVIENT À CRIER ;
 - Réponses sous questions ;
 - Ne pas dépasser 65 caractères dans un mail ;
 - Etc.
- Fond
 - Politesse
 - Respect

Netiquette

- Règles de bonne conduite sur les Internets
- Network Etiquette
- RFC1855 : <http://fgouget.free.fr/netiquette/rfc1855-fr.html>
- Communications un à un (mail, messagerie instantanée)
- Communications un à plusieurs (listes de diffusion, forums)
- Forme
 - Écrire en entier (pas de langage SMS) ;
 - Écrire en minuscules, ÉCRIRE EN MAJUSCULES REVIENT À CRIER ;
 - Réponses sous questions ;
 - Ne pas dépasser 65 caractères dans un mail ;
 - Etc.
- Fond
 - Politesse
 - Respect

Netiquette

- Règles de bonne conduite sur les Internets
- Network Etiquette
- RFC1855 : <http://fgouget.free.fr/netiquette/rfc1855-fr.html>
- Communications un à un (mail, messagerie instantanée)
- Communications un à plusieurs (listes de diffusion, forums)
- Forme
 - Écrire en entier (pas de langage SMS) ;
 - Écrire en minuscules, ÉCRIRE EN MAJUSCULES REVIENT À CRIER ;
 - Réponses sous questions ;
 - Ne pas dépasser 65 caractères dans un mail ;
 - Etc.
- Fond
 - Politesse
 - Respect

Netiquette

- Règles de bonne conduite sur les Internets
- Network Etiquette
- RFC1855 : <http://fgouget.free.fr/netiquette/rfc1855-fr.html>
- Communications un à un (mail, messagerie instantanée)
- Communications un à plusieurs (listes de diffusion, forums)
- Forme
 - Écrire en entier (pas de langage SMS) ;
 - Écrire en minuscules, ÉCRIRE EN MAJUSCULES REVIENT À CRIER ;
 - Réponses sous questions ;
 - Ne pas dépasser 65 caractères dans un mail ;
 - Etc.
- Fond
 - Politesse
 - Respect

Netiquette

- Règles de bonne conduite sur les Internets
- Network Etiquette
- RFC1855 : <http://fgouget.free.fr/netiquette/rfc1855-fr.html>
- Communications un à un (mail, messagerie instantanée)
- Communications un à plusieurs (listes de diffusion, forums)
- Forme
 - Écrire en entier (pas de langage SMS) ;
 - Écrire en minuscules, ÉCRIRE EN MAJUSCULES REVIENT À CRIER ;
 - Réponses sous questions ;
 - Ne pas dépasser 65 caractères dans un mail ;
 - Etc.
- Fond
 - Politesse
 - Respect

Types de communication

- Interne :

- Entre les membres du projets, et les contributeurs ;
- Liés à l'évolution du projet.

- Externe/publique :

- Nouvelles du projet (versions, évolutions importantes).

Types de communication

- Interne :
 - Entre les membres du projets, et les contributeurs ;
 - Liés à l'évolution du projet.
- Externe/publique :
 - Nouvelles du projet (versions, évolutions importantes).

Types de communication

- Interne :
 - Entre les membres du projets, et les contributeurs ;
 - Liés à l'évolution du projet.
- Externe/publique :
 - Nouvelles du projet (versions, évolutions importantes).

Types de communication

- Interne :
 - Entre les membres du projets, et les contributeurs ;
 - Liés à l'évolution du projet.
- Externe/publique :
 - Nouvelles du projet (versions, évolutions importantes).

Types de communication

- Interne :
 - Entre les membres du projets, et les contributeurs ;
 - Liés à l'évolution du projet.
- Externe/publique :
 - Nouvelles du projet (versions, évolutions importantes).

Modes de communication

- Synchrones (messagerie instantanée)

- IRC
- ICQ
- Aol Instant Messaging
- MSN Messenger
- XMPP
- VoIP

- Asynchrones

- Usenet/Newsgroups
- Mail
- Forums

Modes de communication

- Synchrones (messagerie instantanée)

- IRC
- ICQ
- Aol Instant Messaging
- MSN Messenger
- XMPP
- VoIP

- Asynchrones

- Usenet/Newsgroups
- Mail
- Forums

Modes de communication

- Synchrones (messagerie instantanée)

- IRC
- ICQ
- Aol Instant Messaging
- MSN Messenger
- XMPP
- VoIP

- Asynchrones

- Usenet/Newsgroups
- Mail
- Forums

Modes de communication

- Synchrones (messagerie instantanée)

- IRC
- ICQ
- Aol Instant Messaging
- MSN Messenger
- XMPP
- VoIP

- Asynchrones

- Usenet/Newsgroups
- Mail
- Forums

Modes de communication

- Synchrones (messagerie instantanée)

- IRC
- ICQ
- Aol Instant Messaging
- MSN Messenger
- XMPP
- VoIP

- Asynchrones

- Usenet/Newsgroups
- Mail
- Forums

Modes de communication

- Synchrones (messagerie instantanée)

- IRC
- ICQ
- Aol Instant Messaging
- MSN Messenger
- XMPP
- VoIP

- Asynchrones

- Usenet/Newsgroups
- Mail
- Forums

Modes de communication

- Synchrones (messagerie instantanée)

- IRC
- ICQ
- Aol Instant Messaging
- MSN Messenger
- XMPP
- VoIP

- Asynchrones

- Usenet/Newsgroups
- Mail
- Forums

Modes de communication

- Synchrones (messagerie instantanée)

- IRC
- ICQ
- Aol Instant Messaging
- MSN Messenger
- XMPP
- VoIP

- Asynchrones

- Usenet/Newsgroups
- Mail
- Forums

Modes de communication

- Synchrones (messagerie instantanée)

- IRC
- ICQ
- Aol Instant Messaging
- MSN Messenger
- XMPP
- VoIP

- Asynchrones

- Usenet/Newsgroups
- Mail
- Forums

Modes de communication

- Synchrones (messagerie instantanée)

- IRC
- ICQ
- Aol Instant Messaging
- MSN Messenger
- XMPP
- VoIP

- Asynchrones

- Usenet/Newsgroups
- Mail
- Forums

Modes de communication

- Synchrones (messagerie instantanée)

- IRC
- ICQ
- Aol Instant Messaging
- MSN Messenger
- XMPP
- VoIP

- Asynchrones

- Usenet/Newsgroups
- Mail
- Forums

Communication synchrone

- 1 Introduction
- 2 Outils de Développement
- 3 Gestion de la Documentation
- 4 Outils de Communication**
 - Notions générales
 - Communication synchrone**
 - Communication asynchrone
- 5 Forges de développement
- 6 Fonctionnement de certains projets

Internet Relay Chat

Informations générales

- RFC 1459 (août 1988), puis RFC 2810 à 2813 (2000) ;
- Client-Serveur ;
- Réseaux (plusieurs serveurs connectés entre eux) ;
- Discussion orientée salon de discussion (canal, channel, chan) ;
- #canal@reseau ;
- utilisateur@reseau ;
- Modes utilisateur : Opérateur du réseau (o), invisible (i), ...
- Modes : Opérateur (o), voice (v), Secret (s), ...
- Canal et pseudos volatiles.

Internet Relay Chat

Informations générales

- RFC 1459 (août 1988), puis RFC 2810 à 2813 (2000) ;
- Client-Serveur ;
- Réseaux (plusieurs serveurs connectés entre eux) ;
- Discussion orientée salon de discussion (canal, channel, chan) ;
- #canal@reseau ;
- utilisateur@reseau ;
- Modes utilisateur : Opérateur du réseau (o), invisible (i), ...
- Modes : Opérateur (o), voice (v), Secret (s), ...
- Canal et pseudos volatiles.

Internet Relay Chat

Informations générales

- RFC 1459 (août 1988), puis RFC 2810 à 2813 (2000) ;
- Client-Serveur ;
- Réseaux (plusieurs serveurs connectés entre eux) ;
- Discussion orientée salon de discussion (canal, channel, chan) ;
- #canal@reseau ;
- utilisateur@reseau ;
- Modes utilisateur : Opérateur du réseau (o), invisible (i), ...
- Modes : Opérateur (o), voice (v), Secret (s), ...
- Canal et pseudos volatiles.

Internet Relay Chat

Informations générales

- RFC 1459 (août 1988), puis RFC 2810 à 2813 (2000) ;
- Client-Serveur ;
- Réseaux (plusieurs serveurs connectés entre eux) ;
- Discussion orientée salon de discussion (canal, channel, chan) ;
- #canal@reseau ;
- utilisateur@reseau ;
- Modes utilisateur : Opérateur du réseau (o), invisible (i), ...
- Modes : Opérateur (o), voice (v), Secret (s), ...
- Canal et pseudos volatiles.

Internet Relay Chat

Informations générales

- RFC 1459 (août 1988), puis RFC 2810 à 2813 (2000) ;
- Client-Serveur ;
- Réseaux (plusieurs serveurs connectés entre eux) ;
- Discussion orientée salon de discussion (canal, channel, chan) ;
- #canal@reseau ;
- utilisateur@reseau ;
- Modes utilisateur : Opérateur du réseau (o), invisible (i), ...
- Modes : Opérateur (o), voice (v), Secret (s), ...
- Canal et pseudos volatiles.

Internet Relay Chat

Informations générales

- RFC 1459 (août 1988), puis RFC 2810 à 2813 (2000) ;
- Client-Serveur ;
- Réseaux (plusieurs serveurs connectés entre eux) ;
- Discussion orientée salon de discussion (canal, channel, chan) ;
- #canal@reseau ;
- utilisateur@reseau ;
- Modes utilisateur : Opérateur du réseau (o), invisible (i), ...
- Modes : Opérateur (o), voice (v), Secret (s), ...
- Canal et pseudos volatiles.

Internet Relay Chat

Informations générales

- RFC 1459 (août 1988), puis RFC 2810 à 2813 (2000) ;
- Client-Serveur ;
- Réseaux (plusieurs serveurs connectés entre eux) ;
- Discussion orientée salon de discussion (canal, channel, chan) ;
- #canal@reseau ;
- utilisateur@reseau ;
- Modes utilisateur : Opérateur du réseau (o), invisible (i), ...
- Modes : Opérateur (o), voice (v), Secret (s), ...
- Canal et pseudos volatiles.

Internet Relay Chat

Informations générales

- RFC 1459 (août 1988), puis RFC 2810 à 2813 (2000) ;
- Client-Serveur ;
- Réseaux (plusieurs serveurs connectés entre eux) ;
- Discussion orientée salon de discussion (canal, channel, chan) ;
- #canal@reseau ;
- utilisateur@reseau ;
- Modes utilisateur : Opérateur du réseau (o), invisible (i), ...
- Modes : Opérateur (o), voice (v), Secret (s), ...
- Canal et pseudos volatiles.

Internet Relay Chat

Informations générales

- RFC 1459 (août 1988), puis RFC 2810 à 2813 (2000) ;
- Client-Serveur ;
- Réseaux (plusieurs serveurs connectés entre eux) ;
- Discussion orientée salon de discussion (canal, channel, chan) ;
- #canal@reseau ;
- utilisateur@reseau ;
- Modes utilisateur : Opérateur du réseau (o), invisible (i), ...
- Modes : Opérateur (o), voice (v), Secret (s), ...
- Canal et pseudos volatiles.

Internet Relay Chat

Réseaux connus

- **IRCnet** : <http://www.ircnet.org>
- DALnet : <http://www.dal.net>
- EFnet : <http://www.efnet.org>
- Undernet : <http://www.undernet.org>
- QuakeNet : <http://www.quakenet.org>
- Freenode : <http://www.freenode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #aquilenet
 - #adsillh
 - #abul
- Geeknode : <http://www.geeknode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #fdn
 - #sysnove
- OFTC (Debian) : <http://www.oftc.net>
 - #debian-fr

Internet Relay Chat

Réseaux connus

- IRCnet : <http://www.ircnet.org>
- DALnet : <http://www.dal.net>
- EFnet : <http://www.efnet.org>
- Undernet : <http://www.undernet.org>
- QuakeNet : <http://www.quakenet.org>
- Freenode : <http://www.freenode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #aquilenet
 - #adsillh
 - #abul
- Geeknode : <http://www.geeknode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #fdn
 - #sysnove
- OFTC (Debian) : <http://www.oftc.net>
 - #debian-fr

Internet Relay Chat

Réseaux connus

- IRCnet : <http://www.ircnet.org>
- DALnet : <http://www.dal.net>
- EFnet : <http://www.efnet.org>
- Undernet : <http://www.undernet.org>
- QuakeNet : <http://www.quakenet.org>
- Freenode : <http://www.freenode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #aquilenet
 - #adsillh
 - #abul
- Geeknode : <http://www.geeknode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #fdn
 - #sysnove
- OFTC (Debian) : <http://www.oftc.net>
 - #debian-fr

Internet Relay Chat

Réseaux connus

- IRCnet : <http://www.ircnet.org>
- DALnet : <http://www.dal.net>
- EFnet : <http://www.efnet.org>
- Undernet : <http://www.undernet.org>
- QuakeNet : <http://www.quakenet.org>
- Freenode : <http://www.freenode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #aquilenet
 - #adsillh
 - #abul
- Geeknode : <http://www.geeknode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #fdn
 - #sysnove
- OFTC (Debian) : <http://www.oftc.net>
 - #debian-fr

Internet Relay Chat

Réseaux connus

- IRCnet : <http://www.ircnet.org>
- DALnet : <http://www.dal.net>
- EFnet : <http://www.efnet.org>
- Undernet : <http://www.undernet.org>
- QuakeNet : <http://www.quakenet.org>
- Freenode : <http://www.freenode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #aquilenet
 - #adsillh
 - #abul
- Geeknode : <http://www.geeknode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #fdn
 - #sysnove
- OFTC (Debian) : <http://www.oftc.net>
 - #debian-fr

Internet Relay Chat

Réseaux connus

- IRCnet : <http://www.ircnet.org>
- DALnet : <http://www.dal.net>
- EFnet : <http://www.efnet.org>
- Undernet : <http://www.undernet.org>
- QuakeNet : <http://www.quakenet.org>
- Freenode : <http://www.freenode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #aquilenet
 - #adsillh
 - #abul
- Geeknode : <http://www.geeknode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #fdn
 - #sysnove
- OFTC (Debian) : <http://www.oftc.net>
 - #debian-fr

Internet Relay Chat

Réseaux connus

- IRCnet : <http://www.ircnet.org>
- DALnet : <http://www.dal.net>
- EFnet : <http://www.efnet.org>
- Undernet : <http://www.undernet.org>
- QuakeNet : <http://www.quakenet.org>
- Freenode : <http://www.freenode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #aquilenet
 - #adsillh
 - #abul
- Geeknode : <http://www.geeknode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #fdn
 - #sysnove
- OFTC (Debian) : <http://www.oftc.net>
 - #debian-fr

Internet Relay Chat

Réseaux connus

- IRCnet : <http://www.ircnet.org>
- DALnet : <http://www.dal.net>
- EFnet : <http://www.efnet.org>
- Undernet : <http://www.undernet.org>
- QuakeNet : <http://www.quakenet.org>
- Freenode : <http://www.freenode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #aquilenet
 - #adsillh
 - #abul
- Geeknode : <http://www.geeknode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #fdn
 - #sysnove
- OFTC (Debian) : <http://www.oftc.net>
 - #debian-fr

Internet Relay Chat

Réseaux connus

- IRCnet : <http://www.ircnet.org>
- DALnet : <http://www.dal.net>
- EFnet : <http://www.efnet.org>
- Undernet : <http://www.undernet.org>
- QuakeNet : <http://www.quakenet.org>
- Freenode : <http://www.freenode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #aquilenet
 - #adsillh
 - #abul
- Geeknode : <http://www.geeknode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #fdn
 - #sysnove
- OFTC (Debian) : <http://www.oftc.net>
 - #debian-fr

Internet Relay Chat

Réseaux connus

- IRCnet : <http://www.ircnet.org>
- DALnet : <http://www.dal.net>
- EFnet : <http://www.efnet.org>
- Undernet : <http://www.undernet.org>
- QuakeNet : <http://www.quakenet.org>
- Freenode : <http://www.freenode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #aquilenet
 - #adsillh
 - #abul
- Geeknode : <http://www.geeknode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #fdn
 - #sysnove
- OFTC (Debian) : <http://www.oftc.net>
 - #debian-fr

Internet Relay Chat

Réseaux connus

- IRCnet : <http://www.ircnet.org>
- DALnet : <http://www.dal.net>
- EFnet : <http://www.efnet.org>
- Undernet : <http://www.undernet.org>
- QuakeNet : <http://www.quakenet.org>
- Freenode : <http://www.freenode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #aquilenet
 - #adsillh
 - #abul
- Geeknode : <http://www.geeknode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #fdn
 - #sysnove
- OFTC (Debian) : <http://www.oftc.net>
 - #debian-fr

Internet Relay Chat

Réseaux connus

- IRCnet : <http://www.ircnet.org>
- DALnet : <http://www.dal.net>
- EFnet : <http://www.efnet.org>
- Undernet : <http://www.undernet.org>
- QuakeNet : <http://www.quakenet.org>
- Freenode : <http://www.freenode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #aquilenet
 - #adsillh
 - #abul
- Geeknode : <http://www.geeknode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #fdn
 - #sysnove
- OFTC (Debian) : <http://www.oftc.net>
 - #debian-fr

Internet Relay Chat

Réseaux connus

- IRCnet : <http://www.ircnet.org>
- DALnet : <http://www.dal.net>
- EFnet : <http://www.efnet.org>
- Undernet : <http://www.undernet.org>
- QuakeNet : <http://www.quakenet.org>
- Freenode : <http://www.freenode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #aquilenet
 - #adsillh
 - #abul
- Geeknode : <http://www.geeknode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #fdn
 - #sysnove
- OFTC (Debian) : <http://www.oftc.net>
 - #debian-fr

Internet Relay Chat

Réseaux connus

- IRCnet : <http://www.ircnet.org>
- DALnet : <http://www.dal.net>
- EFnet : <http://www.efnet.org>
- Undernet : <http://www.undernet.org>
- QuakeNet : <http://www.quakenet.org>
- Freenode : <http://www.freenode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #aquilenet
 - #adsillh
 - #abul
- Geeknode : <http://www.geeknode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #fdn
 - #sysnove
- OFTC (Debian) : <http://www.oftc.net>
 - #debian-fr

Internet Relay Chat

Réseaux connus

- IRCnet : <http://www.ircnet.org>
- DALnet : <http://www.dal.net>
- EFnet : <http://www.efnet.org>
- Undernet : <http://www.undernet.org>
- QuakeNet : <http://www.quakenet.org>
- Freenode : <http://www.freenode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #aquilenet
 - #adsillh
 - #abul
- Geeknode : <http://www.geeknode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #fdn
 - #sysnove
- OFTC (Debian) : <http://www.oftc.net>
 - #debian-fr

Internet Relay Chat

Réseaux connus

- IRCnet : <http://www.ircnet.org>
- DALnet : <http://www.dal.net>
- EFnet : <http://www.efnet.org>
- Undernet : <http://www.undernet.org>
- QuakeNet : <http://www.quakenet.org>
- Freenode : <http://www.freenode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #aquilenet
 - #adsillh
 - #abul
- Geeknode : <http://www.geeknode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #fdn
 - #sysnove
- OFTC (Debian) : <http://www.oftc.net>
 - #debian-fr

Internet Relay Chat

Réseaux connus

- IRCnet : <http://www.ircnet.org>
- DALnet : <http://www.dal.net>
- EFnet : <http://www.efnet.org>
- Undernet : <http://www.undernet.org>
- QuakeNet : <http://www.quakenet.org>
- Freenode : <http://www.freenode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #aquilenet
 - #adsillh
 - #abul
- Geeknode : <http://www.geeknode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #fdn
 - #sysnove
- OFTC (Debian) : <http://www.oftc.net>
 - #debian-fr

Internet Relay Chat

Réseaux connus

- IRCnet : <http://www.ircnet.org>
- DALnet : <http://www.dal.net>
- EFnet : <http://www.efnet.org>
- Undernet : <http://www.undernet.org>
- QuakeNet : <http://www.quakenet.org>
- Freenode : <http://www.freenode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #aquilenet
 - #adsillh
 - #abul
- Geeknode : <http://www.geeknode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #fdn
 - #sysnove
- OFTC (Debian) : <http://www.oftc.net>
 - #debian-fr

Internet Relay Chat

Réseaux connus

- IRCnet : <http://www.ircnet.org>
- DALnet : <http://www.dal.net>
- EFnet : <http://www.efnet.org>
- Undernet : <http://www.undernet.org>
- QuakeNet : <http://www.quakenet.org>
- Freenode : <http://www.freenode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #aquilenet
 - #adsillh
 - #abul
- Geeknode : <http://www.geeknode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #fdn
 - #sysnove
- OFTC (Debian) : <http://www.oftc.net>
 - #debian-fr

Internet Relay Chat

Réseaux connus

- IRCnet : <http://www.ircnet.org>
- DALnet : <http://www.dal.net>
- EFnet : <http://www.efnet.org>
- Undernet : <http://www.undernet.org>
- QuakeNet : <http://www.quakenet.org>
- Freenode : <http://www.freenode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #aquilenet
 - #adsillh
 - #abul
- Geeknode : <http://www.geeknode.org>
 - Alexis Lahouze : xals
 - Samuel Thibault : youpi
 - Cyril Roelandt : Steap
 - #fdn
 - #sysnove
- OFTC (Debian) : <http://www.oftc.net>
 - #debian-fr

Internet Relay Chat

Bots

- À l'origine créés pour éviter une prise de contrôle par quelqu'un d'autre (takeover) d'un canal ou d'un pseudo en cas d'absence ;
- Permettent de transférer des informations sur le canal ;
- Peuvent réagir à des mots (commandes) ;
- La plupart des réseaux ont leur propre bot de gestion :
 - Freenode : ChanServ et NickServ
 - Geeknode : C
- Quelques logiciels de bot connus :
 - Supybot ;
 - Hubot ;
 - Eggdrop ;
 - Etc.

Internet Relay Chat

Bots

- À l'origine créés pour éviter une prise de contrôle par quelqu'un d'autre (takeover) d'un canal ou d'un pseudo en cas d'absence ;
- Permettent de transférer des informations sur le canal ;
- Peuvent réagir à des mots (commandes) ;
- La plupart des réseaux ont leur propre bot de gestion :
 - Freenode : ChanServ et NickServ
 - Geeknode : C
- Quelques logiciels de bot connus :
 - Supybot ;
 - Hubot ;
 - Eggdrop ;
 - Etc.

Internet Relay Chat

Bots

- À l'origine créés pour éviter une prise de contrôle par quelqu'un d'autre (takeover) d'un canal ou d'un pseudo en cas d'absence ;
- Permettent de transférer des informations sur le canal ;
- Peuvent réagir à des mots (commandes) ;
- La plupart des réseaux ont leur propre bot de gestion :
 - Freenode : ChanServ et NickServ
 - Geeknode : C
- Quelques logiciels de bot connus :
 - Supybot ;
 - Hubot ;
 - Eggdrop ;
 - Etc.

Internet Relay Chat

Bots

- À l'origine créés pour éviter une prise de contrôle par quelqu'un d'autre (takeover) d'un canal ou d'un pseudo en cas d'absence ;
- Permettent de transférer des informations sur le canal ;
- Peuvent réagir à des mots (commandes) ;
- La plupart des réseaux ont leur propre bot de gestion :
 - Freenode : ChanServ et NickServ
 - Geeknode : C
- Quelques logiciels de bot connus :
 - Supybot ;
 - Hubot ;
 - Eggdrop ;
 - Etc.

Internet Relay Chat

Bots

- À l'origine créés pour éviter une prise de contrôle par quelqu'un d'autre (takeover) d'un canal ou d'un pseudo en cas d'absence ;
- Permettent de transférer des informations sur le canal ;
- Peuvent réagir à des mots (commandes) ;
- La plupart des réseaux ont leur propre bot de gestion :
 - Freenode : ChanServ et NickServ
 - Geeknode : C
- Quelques logiciels de bot connus :
 - Supybot ;
 - Hubot ;
 - Eggdrop ;
 - Etc.

Internet Relay Chat

Bots

- À l'origine créés pour éviter une prise de contrôle par quelqu'un d'autre (takeover) d'un canal ou d'un pseudo en cas d'absence ;
- Permettent de transférer des informations sur le canal ;
- Peuvent réagir à des mots (commandes) ;
- La plupart des réseaux ont leur propre bot de gestion :
 - Freenode : ChanServ et NickServ
 - Geeknode : C
- Quelques logiciels de bot connus :
 - Supybot ;
 - Hubot ;
 - Eggdrop ;
 - Etc.

Internet Relay Chat

Bots

- À l'origine créés pour éviter une prise de contrôle par quelqu'un d'autre (takeover) d'un canal ou d'un pseudo en cas d'absence ;
- Permettent de transférer des informations sur le canal ;
- Peuvent réagir à des mots (commandes) ;
- La plupart des réseaux ont leur propre bot de gestion :
 - Freenode : ChanServ et NickServ
 - Geeknode : C
- Quelques logiciels de bot connus :
 - Supybot ;
 - Hubot ;
 - Eggdrop ;
 - Etc.

Internet Relay Chat

Bots

- À l'origine créés pour éviter une prise de contrôle par quelqu'un d'autre (takeover) d'un canal ou d'un pseudo en cas d'absence ;
- Permettent de transférer des informations sur le canal ;
- Peuvent réagir à des mots (commandes) ;
- La plupart des réseaux ont leur propre bot de gestion :
 - Freenode : ChanServ et NickServ
 - Geeknode : C
- Quelques logiciels de bot connus :
 - Supybot ;
 - Hubot ;
 - Eggdrop ;
 - Etc.

Internet Relay Chat

Bots

- À l'origine créés pour éviter une prise de contrôle par quelqu'un d'autre (takeover) d'un canal ou d'un pseudo en cas d'absence ;
- Permettent de transférer des informations sur le canal ;
- Peuvent réagir à des mots (commandes) ;
- La plupart des réseaux ont leur propre bot de gestion :
 - Freenode : ChanServ et NickServ
 - Geeknode : C
- Quelques logiciels de bot connus :
 - Supybot ;
 - Hubot ;
 - Eggdrop ;
 - Etc.

Internet Relay Chat

Bots

- À l'origine créés pour éviter une prise de contrôle par quelqu'un d'autre (takeover) d'un canal ou d'un pseudo en cas d'absence ;
- Permettent de transférer des informations sur le canal ;
- Peuvent réagir à des mots (commandes) ;
- La plupart des réseaux ont leur propre bot de gestion :
 - Freenode : ChanServ et NickServ
 - Geeknode : C
- Quelques logiciels de bot connus :
 - Supybot ;
 - Hubot ;
 - Eggdrop ;
 - Etc.

Internet Relay Chat

Bots

- À l'origine créés pour éviter une prise de contrôle par quelqu'un d'autre (takeover) d'un canal ou d'un pseudo en cas d'absence ;
- Permettent de transférer des informations sur le canal ;
- Peuvent réagir à des mots (commandes) ;
- La plupart des réseaux ont leur propre bot de gestion :
 - Freenode : ChanServ et NickServ
 - Geeknode : C
- Quelques logiciels de bot connus :
 - Supybot ;
 - Hubot ;
 - Eggdrop ;
 - Etc.

Internet Relay Chat

Clients les plus connus

- Console :

- BitchX (semble maintenu car derniers commits dans la journée, mais dernière release le 14 novembre 2014)
- Irssi (toujours actif, dernière release le 21 septembre dernier)
- Weechat (très actif)

- Graphiques :

- mIRC (Windows, développement toujours actif) ;
- kvIRC (multiplateforme, anciennement KDE) ;
- konversation (KDE) ;
- XChat (GTK) ;
- Chatzilla (extension Firefox) ;
- Pidgin (GTK) ;
- Passerelles XMPP <-> IRC.

Internet Relay Chat

Clients les plus connus

- Console :

- BitchX (semble maintenu car derniers commits dans la journée, mais dernière release le 14 novembre 2014)
- Irssi (toujours actif, dernière release le 21 septembre dernier)
- Weechat (très actif)

- Graphiques :

- mIRC (Windows, développement toujours actif) ;
- kvIRC (multiplateforme, anciennement KDE) ;
- konversation (KDE) ;
- XChat (GTK) ;
- Chatzilla (extension Firefox) ;
- Pidgin (GTK) ;
- Passerelles XMPP <-> IRC.

Internet Relay Chat

Clients les plus connus

- Console :

- BitchX (semble maintenu car derniers commits dans la journée, mais dernière release le 14 novembre 2014)
- Irssi (toujours actif, dernière release le 21 septembre dernier)
- Weechat (très actif)

- Graphiques :

- mIRC (Windows, développement toujours actif) ;
- kvIRC (multiplateforme, anciennement KDE) ;
- konversation (KDE) ;
- XChat (GTK) ;
- Chatzilla (extension Firefox) ;
- Pidgin (GTK) ;
- Passerelles XMPP <-> IRC.

Internet Relay Chat

Clients les plus connus

- Console :

- BitchX (semble maintenu car derniers commits dans la journée, mais dernière release le 14 novembre 2014)
- Irssi (toujours actif, dernière release le 21 septembre dernier)
- Weechat (très actif)

- Graphiques :

- mIRC (Windows, développement toujours actif) ;
- kvIRC (multiplateforme, anciennement KDE) ;
- konversation (KDE) ;
- XChat (GTK) ;
- Chatzilla (extension Firefox) ;
- Pidgin (GTK) ;
- Passerelles XMPP <-> IRC.

Internet Relay Chat

Clients les plus connus

- Console :

- BitchX (semble maintenu car derniers commits dans la journée, mais dernière release le 14 novembre 2014)
- Irssi (toujours actif, dernière release le 21 septembre dernier)
- Weechat (très actif)

- Graphiques :

- mIRC (Windows, développement toujours actif) ;
- kvIRC (multiplateforme, anciennement KDE) ;
- konversation (KDE) ;
- XChat (GTK) ;
- Chatzilla (extension Firefox) ;
- Pidgin (GTK) ;
- Passerelles XMPP <-> IRC.

Internet Relay Chat

Clients les plus connus

- Console :

- BitchX (semble maintenu car derniers commits dans la journée, mais dernière release le 14 novembre 2014)
- Irssi (toujours actif, dernière release le 21 septembre dernier)
- Weechat (très actif)

- Graphiques :

- mIRC (Windows, développement toujours actif) ;
- kvIRC (multiplateforme, anciennement KDE) ;
- konversation (KDE) ;
- XChat (GTK) ;
- Chatzilla (extension Firefox) ;
- Pidgin (GTK) ;
- Passerelles XMPP <-> IRC.

Internet Relay Chat

Clients les plus connus

- Console :

- BitchX (semble maintenu car derniers commits dans la journée, mais dernière release le 14 novembre 2014)
- Irssi (toujours actif, dernière release le 21 septembre dernier)
- Weechat (très actif)

- Graphiques :

- mIRC (Windows, développement toujours actif) ;
- kvIRC (multiplateforme, anciennement KDE) ;
- konversation (KDE) ;
- XChat (GTK) ;
- Chatzilla (extension Firefox) ;
- Pidgin (GTK) ;
- Passerelles XMPP <-> IRC.

Internet Relay Chat

Clients les plus connus

- Console :

- BitchX (semble maintenu car derniers commits dans la journée, mais dernière release le 14 novembre 2014)
- Irssi (toujours actif, dernière release le 21 septembre dernier)
- Weechat (très actif)

- Graphiques :

- mIRC (Windows, développement toujours actif) ;
- kvIRC (multiplateforme, anciennement KDE) ;
- konversation (KDE) ;
- XChat (GTK) ;
- Chatzilla (extension Firefox) ;
- Pidgin (GTK) ;
- Passerelles XMPP <-> IRC.

Internet Relay Chat

Clients les plus connus

- Console :

- BitchX (semble maintenu car derniers commits dans la journée, mais dernière release le 14 novembre 2014)
- Irssi (toujours actif, dernière release le 21 septembre dernier)
- Weechat (très actif)

- Graphiques :

- mIRC (Windows, développement toujours actif) ;
- kvIRC (multiplateforme, anciennement KDE) ;
- konversation (KDE) ;
- XChat (GTK) ;
- Chatzilla (extension Firefox) ;
- Pidgin (GTK) ;
- Passerelles XMPP <-> IRC.

Internet Relay Chat

Clients les plus connus

- Console :

- BitchX (semble maintenu car derniers commits dans la journée, mais dernière release le 14 novembre 2014)
- Irssi (toujours actif, dernière release le 21 septembre dernier)
- Weechat (très actif)

- Graphiques :

- mIRC (Windows, développement toujours actif) ;
- kvIRC (multiplateforme, anciennement KDE) ;
- konversation (KDE) ;
- XChat (GTK) ;
- Chatzilla (extension Firefox) ;
- Pidgin (GTK) ;
- Passerelles XMPP <-> IRC.

XMPP (Jabber)

Informations générales

- Créé en 1998, standardisé IETF en 2002 (RFC 3920 à 3923) ;
- Client-Serveur-Serveur (interconnexion des domaines) ;
- Orienté discussion un à un (comme ICQ) ;
- Liste de contacts ;
- Chiffrement bout en bout avec OTR et PGP.

XMPP (Jabber)

Informations générales

- Créé en 1998, standardisé IETF en 2002 (RFC 3920 à 3923) ;
- Client-Serveur-Serveur (interconnexion des domaines) ;
- Orienté discussion un à un (comme ICQ) ;
- Liste de contacts ;
- Chiffrement bout en bout avec OTR et PGP.

XMPP (Jabber)

Informations générales

- Créé en 1998, standardisé IETF en 2002 (RFC 3920 à 3923) ;
- Client-Serveur-Serveur (interconnexion des domaines) ;
- Orienté discussion un à un (comme ICQ) ;
- Liste de contacts ;
- Chiffrement bout en bout avec OTR et PGP.

XMPP (Jabber)

Informations générales

- Créé en 1998, standardisé IETF en 2002 (RFC 3920 à 3923) ;
- Client-Serveur-Serveur (interconnexion des domaines) ;
- Orienté discussion un à un (comme ICQ) ;
- Liste de contacts ;
- Chiffrement bout en bout avec OTR et PGP.

XMPP (Jabber)

Informations générales

- Créé en 1998, standardisé IETF en 2002 (RFC 3920 à 3923) ;
- Client-Serveur-Serveur (interconnexion des domaines) ;
- Orienté discussion un à un (comme ICQ) ;
- Liste de contacts ;
- Chiffrement bout en bout avec OTR et PGP.

XMPP (Jabber)

Logiciels Clients

- **Pidgin (multiplateforme)**
- Gajim (multiplateforme)
- Psi (multiplateforme)
- Xabber (Android)
- Conversations (Android)
- Kopete (Linux, multiprotocole)
- Miranda IM (Windows, multiprotocole)
- Jappix (Web/Ajax)
- Bitlbee (Linux, serveur IRC, client multiprotocole)
- Etc.

XMPP (Jabber)

Logiciels Clients

- Pidgin (multiplateforme)
- Gajim (multiplateforme)
- Psi (multiplateforme)
- Xabber (Android)
- Conversations (Android)
- Kopete (Linux, multiprotocole)
- Miranda IM (Windows, multiprotocole)
- Jappix (Web/Ajax)
- Bitlbee (Linux, serveur IRC, client multiprotocole)
- Etc.

XMPP (Jabber)

Logiciels Clients

- Pidgin (multiplateforme)
- Gajim (multiplateforme)
- Psi (multiplateforme)
- Xabber (Android)
- Conversations (Android)
- Kopete (Linux, multiprotocole)
- Miranda IM (Windows, multiprotocole)
- Jappix (Web/Ajax)
- Bitlbee (Linux, serveur IRC, client multiprotocole)
- Etc.

XMPP (Jabber)

Logiciels Clients

- Pidgin (multiplateforme)
- Gajim (multiplateforme)
- Psi (multiplateforme)
- Xabber (Android)
- Conversations (Android)
- Kopete (Linux, multiprotocole)
- Miranda IM (Windows, multiprotocole)
- Jappix (Web/Ajax)
- Bitlbee (Linux, serveur IRC, client multiprotocole)
- Etc.

XMPP (Jabber)

Logiciels Clients

- Pidgin (multiplateforme)
- Gajim (multiplateforme)
- Psi (multiplateforme)
- Xabber (Android)
- Conversations (Android)
- Kopete (Linux, multiprotocole)
- Miranda IM (Windows, multiprotocole)
- Jappix (Web/Ajax)
- Bitlbee (Linux, serveur IRC, client multiprotocole)
- Etc.

XMPP (Jabber)

Logiciels Clients

- Pidgin (multiplateforme)
- Gajim (multiplateforme)
- Psi (multiplateforme)
- Xabber (Android)
- Conversations (Android)
- Kopete (Linux, multiprotocole)
- Miranda IM (Windows, multiprotocole)
- Jappix (Web/Ajax)
- Bitlbee (Linux, serveur IRC, client multiprotocole)
- Etc.

XMPP (Jabber)

Logiciels Clients

- Pidgin (multiplateforme)
- Gajim (multiplateforme)
- Psi (multiplateforme)
- Xabber (Android)
- Conversations (Android)
- Kopete (Linux, multiprotocole)
- Miranda IM (Windows, multiprotocole)
- Jappix (Web/Ajax)
- Bitlbee (Linux, serveur IRC, client multiprotocole)
- Etc.

XMPP (Jabber)

Logiciels Clients

- Pidgin (multiplateforme)
- Gajim (multiplateforme)
- Psi (multiplateforme)
- Xabber (Android)
- Conversations (Android)
- Kopete (Linux, multiprotocole)
- Miranda IM (Windows, multiprotocole)
- Jappix (Web/Ajax)
- Bitlbee (Linux, serveur IRC, client multiprotocole)
- Etc.

XMPP (Jabber)

Logiciels Clients

- Pidgin (multiplateforme)
- Gajim (multiplateforme)
- Psi (multiplateforme)
- Xabber (Android)
- Conversations (Android)
- Kopete (Linux, multiprotocole)
- Miranda IM (Windows, multiprotocole)
- Jappix (Web/Ajax)
- Bitlbee (Linux, serveur IRC, client multiprotocole)
- Etc.

XMPP (Jabber)

Logiciels Clients

- Pidgin (multiplateforme)
- Gajim (multiplateforme)
- Psi (multiplateforme)
- Xabber (Android)
- Conversations (Android)
- Kopete (Linux, multiprotocole)
- Miranda IM (Windows, multiprotocole)
- Jappix (Web/Ajax)
- Bitlbee (Linux, serveur IRC, client multiprotocole)
- Etc.

XMPP (Jabber)

Logiciels Serveurs

- **Jabberd14 (C) ;**
- Jabberd2 (C) ;
- eJabberd (erlang) ;
- Prosody (Lua) ;
- Tigase (Java) ;
- Etc.

XMPP (Jabber)

Logiciels Serveurs

- Jabberd14 (C) ;
- Jabberd2 (C) ;
- eJabberd (erlang) ;
- Prosody (Lua) ;
- Tigase (Java) ;
- Etc.

XMPP (Jabber)

Logiciels Serveurs

- Jabberd14 (C) ;
- Jabberd2 (C) ;
- eJabberd (erlang) ;
- Prosody (Lua) ;
- Tigase (Java) ;
- Etc.

XMPP (Jabber)

Logiciels Serveurs

- Jabberd14 (C) ;
- Jabberd2 (C) ;
- eJabberd (erlang) ;
- Prosody (Lua) ;
- Tigase (Java) ;
- Etc.

XMPP (Jabber)

Logiciels Serveurs

- Jabberd14 (C) ;
- Jabberd2 (C) ;
- eJabberd (erlang) ;
- Prosody (Lua) ;
- Tigase (Java) ;
- Etc.

XMPP (Jabber)

Logiciels Serveurs

- Jabberd14 (C) ;
- Jabberd2 (C) ;
- eJabberd (erlang) ;
- Prosody (Lua) ;
- Tigase (Java) ;
- Etc.

XMPP (Jabber)

Services XMPP connus

- GTalk (Google)
- im.apinc.org (FR)
- jabber.im

XMPP (Jabber)

Services XMPP connus

- GTalk (Google)
- im.apinc.org (FR)
- jabber.im

XMPP (Jabber)

Services XMPP connus

- GTalk (Google)
- im.apinc.org (FR)
- jabber.im

Autres systèmes

- La plupart propriétaires ;
 - Slack ;
 - Notion d'équipes ;
 - Possibilité de connecter des applications externes ;
 - Utilisé par les startups.
 - Gitter (github) ;
 - Orienté projet.
 - Certains réseaux sociaux ;
 - Etc.

Autres systèmes

- La plupart propriétaires ;
- Slack ;
 - Notion d'équipes ;
 - Possibilité de connecter des applications externes ;
 - Utilisé par les startups.
- Gitter (github) ;
 - Orienté projet.
- Certains réseaux sociaux ;
- Etc.

Autres systèmes

- La plupart propriétaires ;
- Slack ;
 - Notion d'équipes ;
 - Possibilité de connecter des applications externes ;
 - Utilisé par les startups.
- Gitter (github) ;
 - Orienté projet.
- Certains réseaux sociaux ;
- Etc.

Autres systèmes

- La plupart propriétaires ;
- Slack ;
 - Notion d'équipes ;
 - Possibilité de connecter des applications externes ;
 - Utilisé par les startups.
- Gitter (github) ;
 - Orienté projet.
- Certains réseaux sociaux ;
- Etc.

Autres systèmes

- La plupart propriétaires ;
- Slack ;
 - Notion d'équipes ;
 - Possibilité de connecter des applications externes ;
 - Utilisé par les startups.
- Gitter (github) ;
 - Orienté projet.
- Certains réseaux sociaux ;
- Etc.

Autres systèmes

- La plupart propriétaires ;
- Slack ;
 - Notion d'équipes ;
 - Possibilité de connecter des applications externes ;
 - Utilisé par les startups.
- Gitter (github) ;
 - Orienté projet.
- Certains réseaux sociaux ;
- Etc.

Autres systèmes

- La plupart propriétaires ;
- Slack ;
 - Notion d'équipes ;
 - Possibilité de connecter des applications externes ;
 - Utilisé par les startups.
- Gitter (github) ;
 - Orienté projet.
- Certains réseaux sociaux ;
- Etc.

Communication asynchrone

- 1 Introduction
- 2 Outils de Développement
- 3 Gestion de la Documentation
- 4 Outils de Communication**
 - Notions générales
 - Communication synchrone
 - Communication asynchrone**
- 5 Forges de développement
- 6 Fonctionnement de certains projets

Notion de base

- On sait quand ça part, mais on ne sait pas quand ça arrive.

Usenet/Newsgroups

- Inventé en 1979 ;
- Client-serveur ;
- Protocole UUCP (Unix to Unix Copy Protocol) puis NNTP (Network News Transfer Protocol) ;
- Réseau de forums ;
- Génération, stockage et récupération d'articles ;
- Très proche du mail ;
- Notion de groupes et d'abonnements à ces groupes.

Usenet/Newsgroups

- Inventé en 1979 ;
- Client-serveur ;
- Protocole UUCP (Unix to Unix Copy Protocol) puis NNTP (Network News Transfer Protocol) ;
- Réseau de forums ;
- Génération, stockage et récupération d'articles ;
- Très proche du mail ;
- Notion de groupes et d'abonnements à ces groupes.

Usenet/Newsgroups

- Inventé en 1979 ;
- Client-serveur ;
- Protocole UUCP (Unix to Unix Copy Protocol) puis NNTP (Network News Transfer Protocol) ;
- Réseau de forums ;
- Génération, stockage et récupération d'articles ;
- Très proche du mail ;
- Notion de groupes et d'abonnements à ces groupes.

Usenet/Newsgroups

- Inventé en 1979 ;
- Client-serveur ;
- Protocole UUCP (Unix to Unix Copy Protocol) puis NNTP (Network News Transfer Protocol) ;
- Réseau de forums ;
- Génération, stockage et récupération d'articles ;
- Très proche du mail ;
- Notion de groupes et d'abonnements à ces groupes.

Usenet/Newsgroups

- Inventé en 1979 ;
- Client-serveur ;
- Protocole UUCP (Unix to Unix Copy Protocol) puis NNTP (Network News Transfer Protocol) ;
- Réseau de forums ;
- Génération, stockage et récupération d'articles ;
- Très proche du mail ;
- Notion de groupes et d'abonnements à ces groupes.

Usenet/Newsgroups

- Inventé en 1979 ;
- Client-serveur ;
- Protocole UUCP (Unix to Unix Copy Protocol) puis NNTP (Network News Transfer Protocol) ;
- Réseau de forums ;
- Génération, stockage et récupération d'articles ;
- Très proche du mail ;
- Notion de groupes et d'abonnements à ces groupes.

Usenet/Newsgroups

- Inventé en 1979 ;
- Client-serveur ;
- Protocole UUCP (Unix to Unix Copy Protocol) puis NNTP (Network News Transfer Protocol) ;
- Réseau de forums ;
- Génération, stockage et récupération d'articles ;
- Très proche du mail ;
- Notion de groupes et d'abonnements à ces groupes.

Mails

Informations générales

- **Prémises en 1965 ;**
- RFC 561 en 1971, puis RFC 680, 724 et enfin 733 en 1977 : entêtes et protocole ;
- RFC 822 en 1982 : format des messages ;
- RFC 2045 à 2049 : MIME ;
- RFC 2822 en 2001 : remplace la 822 ;
- RFC 5322 en 2008 : étend la 2822 ;
- Client-Serveur-Serveur ;
- Chiffage avec PGP (clé asymétrique), ou S/MIME (certificat numérique) ;
- Organisation par sujets (threads) :
 - Entête In-Reply-To.

Mails

Informations générales

- Prémisses en 1965 ;
- RFC 561 en 1971, puis RFC 680, 724 et enfin 733 en 1977 : entêtes et protocole ;
- RFC 822 en 1982 : format des messages ;
- RFC 2045 à 2049 : MIME ;
- RFC 2822 en 2001 : remplace la 822 ;
- RFC 5322 en 2008 : étend la 2822 ;
- Client-Serveur-Serveur ;
- Chiffage avec PGP (clé asymétrique), ou S/MIME (certificat numérique) ;
- Organisation par sujets (threads) :
 - Entête In-Reply-To.

Mails

Informations générales

- Prémisses en 1965 ;
- RFC 561 en 1971, puis RFC 680, 724 et enfin 733 en 1977 : entêtes et protocole ;
- RFC 822 en 1982 : format des messages ;
- RFC 2045 à 2049 : MIME ;
- RFC 2822 en 2001 : remplace la 822 ;
- RFC 5322 en 2008 : étend la 2822 ;
- Client-Serveur-Serveur ;
- Chiffage avec PGP (clé asymétrique), ou S/MIME (certificat numérique) ;
- Organisation par sujets (threads) :
 - Entête In-Reply-To.

Mails

Informations générales

- Prémisses en 1965 ;
- RFC 561 en 1971, puis RFC 680, 724 et enfin 733 en 1977 : entêtes et protocole ;
- RFC 822 en 1982 : format des messages ;
- RFC 2045 à 2049 : MIME ;
- RFC 2822 en 2001 : remplace la 822 ;
- RFC 5322 en 2008 : étend la 2822 ;
- Client-Serveur-Serveur ;
- Chiffage avec PGP (clé asymétrique), ou S/MIME (certificat numérique) ;
- Organisation par sujets (threads) :
 - Entête In-Reply-To.

Mails

Informations générales

- Prémisses en 1965 ;
- RFC 561 en 1971, puis RFC 680, 724 et enfin 733 en 1977 : entêtes et protocole ;
- RFC 822 en 1982 : format des messages ;
- RFC 2045 à 2049 : MIME ;
- RFC 2822 en 2001 : remplace la 822 ;
- RFC 5322 en 2008 : étend la 2822 ;
- Client-Serveur-Serveur ;
- Chiffage avec PGP (clé asymétrique), ou S/MIME (certificat numérique) ;
- Organisation par sujets (threads) :
 - Entête In-Reply-To.

Mails

Informations générales

- Prémisses en 1965 ;
- RFC 561 en 1971, puis RFC 680, 724 et enfin 733 en 1977 : entêtes et protocole ;
- RFC 822 en 1982 : format des messages ;
- RFC 2045 à 2049 : MIME ;
- RFC 2822 en 2001 : remplace la 822 ;
- RFC 5322 en 2008 : étend la 2822 ;
- Client-Serveur-Serveur ;
- Chiffage avec PGP (clé asymétrique), ou S/MIME (certificat numérique) ;
- Organisation par sujets (threads) :
 - Entête In-Reply-To.

Mails

Informations générales

- Prémisses en 1965 ;
- RFC 561 en 1971, puis RFC 680, 724 et enfin 733 en 1977 : entêtes et protocole ;
- RFC 822 en 1982 : format des messages ;
- RFC 2045 à 2049 : MIME ;
- RFC 2822 en 2001 : remplace la 822 ;
- RFC 5322 en 2008 : étend la 2822 ;
- Client-Serveur-Serveur ;
- Chiffage avec PGP (clé asymétrique), ou S/MIME (certificat numérique) ;
- Organisation par sujets (threads) :
 - Entête In-Reply-To.

Mails

Informations générales

- Prémisses en 1965 ;
- RFC 561 en 1971, puis RFC 680, 724 et enfin 733 en 1977 : entêtes et protocole ;
- RFC 822 en 1982 : format des messages ;
- RFC 2045 à 2049 : MIME ;
- RFC 2822 en 2001 : remplace la 822 ;
- RFC 5322 en 2008 : étend la 2822 ;
- Client-Serveur-Serveur ;
- Chiffrage avec PGP (clé asymétrique), ou S/MIME (certificat numérique) ;
- Organisation par sujets (threads) :
 - Entête In-Reply-To.

Mails

Informations générales

- Prémisses en 1965 ;
- RFC 561 en 1971, puis RFC 680, 724 et enfin 733 en 1977 : entêtes et protocole ;
- RFC 822 en 1982 : format des messages ;
- RFC 2045 à 2049 : MIME ;
- RFC 2822 en 2001 : remplace la 822 ;
- RFC 5322 en 2008 : étend la 2822 ;
- Client-Serveur-Serveur ;
- Chiffage avec PGP (clé asymétrique), ou S/MIME (certificat numérique) ;
- Organisation par sujets (threads) :
 - Entête In-Reply-To.

Mails

Informations générales

- Prémisses en 1965 ;
- RFC 561 en 1971, puis RFC 680, 724 et enfin 733 en 1977 : entêtes et protocole ;
- RFC 822 en 1982 : format des messages ;
- RFC 2045 à 2049 : MIME ;
- RFC 2822 en 2001 : remplace la 822 ;
- RFC 5322 en 2008 : étend la 2822 ;
- Client-Serveur-Serveur ;
- Chiffage avec PGP (clé asymétrique), ou S/MIME (certificat numérique) ;
- Organisation par sujets (threads) :
 - Entête In-Reply-To.

Mails

Mailing-lists

- **Mails gérés par un robot de liste, ou gestionnaire de listes ;**
 - Le gestionnaire permet de gérer les abonnements ;
 - Adresse de destination unique ;
 - Redirection à tous les abonnés ;
 - Utilisées pour communiquer entre les membres de l'équipe ;
 - Utilisées pour communiquer avec l'extérieur (newsletters) ;
 - Souvent couplées à une interface web de gestion ;
 - Archives ;
 - Aggrégateurs de listes publiques :
 - <http://www.gmane.org> (listes de diffusion et newsgroups) ;
 - Moteurs connus :
 - Sympa
 - Mailman
 - Majordomo

Mails

Mailing-lists

- Mails gérés par un robot de liste, ou gestionnaire de listes ;
- Le gestionnaire permet de gérer les abonnements ;
- Adresse de destination unique ;
- Redirection à tous les abonnés ;
- Utilisées pour communiquer entre les membres de l'équipe ;
- Utilisées pour communiquer avec l'extérieur (newsletters) ;
- Souvent couplées à une interface web de gestion ;
- Archives ;
- Aggrégateurs de listes publiques :
 - <http://www.gmane.org> (listes de diffusion et newsgroups) ;
- Moteurs connus :
 - Sympa
 - Mailman
 - Majordomo

Mails

Mailing-lists

- Mails gérés par un robot de liste, ou gestionnaire de listes ;
- Le gestionnaire permet de gérer les abonnements ;
- Adresse de destination unique ;
- Redirection à tous les abonnés ;
- Utilisées pour communiquer entre les membres de l'équipe ;
- Utilisées pour communiquer avec l'extérieur (newsletters) ;
- Souvent couplées à une interface web de gestion ;
- Archives ;
- Aggrégateurs de listes publiques :
 - <http://www.gmane.org> (listes de diffusion et newsgroups) ;
- Moteurs connus :
 - Sympa
 - Mailman
 - Majordomo

Mails

Mailing-lists

- Mails gérés par un robot de liste, ou gestionnaire de listes ;
- Le gestionnaire permet de gérer les abonnements ;
- Adresse de destination unique ;
- Redirection à tous les abonnés ;
- Utilisées pour communiquer entre les membres de l'équipe ;
- Utilisées pour communiquer avec l'extérieur (newsletters) ;
- Souvent couplées à une interface web de gestion ;
- Archives ;
- Aggrégateurs de listes publiques :
 - <http://www.gmane.org> (listes de diffusion et newsgroups) ;
- Moteurs connus :
 - Sympa
 - Mailman
 - Majordomo

Mails

Mailing-lists

- Mails gérés par un robot de liste, ou gestionnaire de listes ;
- Le gestionnaire permet de gérer les abonnements ;
- Adresse de destination unique ;
- Redirection à tous les abonnés ;
- Utilisées pour communiquer entre les membres de l'équipe ;
- Utilisées pour communiquer avec l'extérieur (newsletters) ;
- Souvent couplées à une interface web de gestion ;
- Archives ;
- Aggrégateurs de listes publiques :
 - <http://www.gmane.org> (listes de diffusion et newsgroups) ;
- Moteurs connus :
 - Sympa
 - Mailman
 - Majordomo

Mails

Mailing-lists

- Mails gérés par un robot de liste, ou gestionnaire de listes ;
- Le gestionnaire permet de gérer les abonnements ;
- Adresse de destination unique ;
- Redirection à tous les abonnés ;
- Utilisées pour communiquer entre les membres de l'équipe ;
- Utilisées pour communiquer avec l'extérieur (newsletters) ;
- Souvent couplées à une interface web de gestion ;
- Archives ;
- Aggrégateurs de listes publiques :
 - <http://www.gmane.org> (listes de diffusion et newsgroups) ;
- Moteurs connus :
 - Sympa
 - Mailman
 - Majordomo

Mails

Mailing-lists

- Mails gérés par un robot de liste, ou gestionnaire de listes ;
- Le gestionnaire permet de gérer les abonnements ;
- Adresse de destination unique ;
- Redirection à tous les abonnés ;
- Utilisées pour communiquer entre les membres de l'équipe ;
- Utilisées pour communiquer avec l'extérieur (newsletters) ;
- Souvent couplées à une interface web de gestion ;
- Archives ;
- Aggrégateurs de listes publiques :
 - <http://www.gmane.org> (listes de diffusion et newsgroups) ;
- Moteurs connus :
 - Sympa
 - Mailman
 - Majordomo

Mails

Mailing-lists

- Mails gérés par un robot de liste, ou gestionnaire de listes ;
- Le gestionnaire permet de gérer les abonnements ;
- Adresse de destination unique ;
- Redirection à tous les abonnés ;
- Utilisées pour communiquer entre les membres de l'équipe ;
- Utilisées pour communiquer avec l'extérieur (newsletters) ;
- Souvent couplées à une interface web de gestion ;
- Archives ;
- Aggrégateurs de listes publiques :
 - <http://www.gmane.org> (listes de diffusion et newsgroups) ;
- Moteurs connus :
 - Sympa
 - Mailman
 - Majordomo

Mails

Mailing-lists

- Mails gérés par un robot de liste, ou gestionnaire de listes ;
- Le gestionnaire permet de gérer les abonnements ;
- Adresse de destination unique ;
- Redirection à tous les abonnés ;
- Utilisées pour communiquer entre les membres de l'équipe ;
- Utilisées pour communiquer avec l'extérieur (newsletters) ;
- Souvent couplées à une interface web de gestion ;
- Archives ;
- Aggrégateurs de listes publiques :
 - <http://www.gmane.org> (listes de diffusion et newsgroups) ;
- Moteurs connus :
 - Sympa
 - Mailman
 - Majordomo

Mails

Mailing-lists

- Mails gérés par un robot de liste, ou gestionnaire de listes ;
- Le gestionnaire permet de gérer les abonnements ;
- Adresse de destination unique ;
- Redirection à tous les abonnés ;
- Utilisées pour communiquer entre les membres de l'équipe ;
- Utilisées pour communiquer avec l'extérieur (newsletters) ;
- Souvent couplées à une interface web de gestion ;
- Archives ;
- Aggrégateurs de listes publiques :
 - <http://www.gmane.org> (listes de diffusion et newsgroups) ;
- Moteurs connus :
 - Sympa
 - Mailman
 - Majordomo

Mails

Mailing-lists

- Mails gérés par un robot de liste, ou gestionnaire de listes ;
- Le gestionnaire permet de gérer les abonnements ;
- Adresse de destination unique ;
- Redirection à tous les abonnés ;
- Utilisées pour communiquer entre les membres de l'équipe ;
- Utilisées pour communiquer avec l'extérieur (newsletters) ;
- Souvent couplées à une interface web de gestion ;
- Archives ;
- Agrégateurs de listes publiques :
 - <http://www.gmane.org> (listes de diffusion et newsgroups) ;
- Moteurs connus :
 - Sympa
 - Mailman
 - Majordomo

Mails

Mailing-lists

- Mails gérés par un robot de liste, ou gestionnaire de listes ;
- Le gestionnaire permet de gérer les abonnements ;
- Adresse de destination unique ;
- Redirection à tous les abonnés ;
- Utilisées pour communiquer entre les membres de l'équipe ;
- Utilisées pour communiquer avec l'extérieur (newsletters) ;
- Souvent couplées à une interface web de gestion ;
- Archives ;
- Aggrégateurs de listes publiques :
 - <http://www.gmane.org> (listes de diffusion et newsgroups) ;
- Moteurs connus :
 - Sympa
 - Mailman
 - Majordomo

Mails

Mailing-lists

- Mails gérés par un robot de liste, ou gestionnaire de listes ;
- Le gestionnaire permet de gérer les abonnements ;
- Adresse de destination unique ;
- Redirection à tous les abonnés ;
- Utilisées pour communiquer entre les membres de l'équipe ;
- Utilisées pour communiquer avec l'extérieur (newsletters) ;
- Souvent couplées à une interface web de gestion ;
- Archives ;
- Aggrégateurs de listes publiques :
 - <http://www.gmane.org> (listes de diffusion et newsgroups) ;
- Moteurs connus :
 - Sympa
 - Mailman
 - Majordomo

Mails

Mailing-lists

- Mails gérés par un robot de liste, ou gestionnaire de listes ;
- Le gestionnaire permet de gérer les abonnements ;
- Adresse de destination unique ;
- Redirection à tous les abonnés ;
- Utilisées pour communiquer entre les membres de l'équipe ;
- Utilisées pour communiquer avec l'extérieur (newsletters) ;
- Souvent couplées à une interface web de gestion ;
- Archives ;
- Aggrégateurs de listes publiques :
 - <http://www.gmane.org> (listes de diffusion et newsgroups) ;
- Moteurs connus :
 - Sympa
 - Mailman
 - Majordomo

Mails

Mailing-lists, entêtes spécifiques

RFC 4021 ;

List-ID Identifiant de la liste, permet une classification ;

List-Owner Adresse du propriétaire de la liste ;

List-Post Adresse d'envoi de la liste ;

List-Subscribe Adresse d'abonnement à la liste ;

List-Unsubscribe Adresse de désabonnement de la liste ;

List-Help Adresse pour obtenir de l'aide sur la liste ;

List-Archive Adresse de l'archive de la liste.

Mails

Mailing-lists, entêtes spécifiques

RFC 4021 ;

List-ID Identifiant de la liste, permet une classification ;

List-Owner Adresse du propriétaire de la liste ;

List-Post Adresse d'envoi de la liste ;

List-Subscribe Adresse d'abonnement à la liste ;

List-Unsubscribe Adresse de désabonnement de la liste ;

List-Help Adresse pour obtenir de l'aide sur la liste ;

List-Archive Adresse de l'archive de la liste.

Mails

Mailing-lists, entêtes spécifiques

RFC 4021 ;

List-ID Identifiant de la liste, permet une classification ;

List-Owner Adresse du propriétaire de la liste ;

List-Post Adresse d'envoi de la liste ;

List-Subscribe Adresse d'abonnement à la liste ;

List-Unsubscribe Adresse de désabonnement de la liste ;

List-Help Adresse pour obtenir de l'aide sur la liste ;

List-Archive Adresse de l'archive de la liste.

Mails

Mailing-lists, entêtes spécifiques

RFC 4021 ;

List-ID Identifiant de la liste, permet une classification ;

List-Owner Adresse du propriétaire de la liste ;

List-Post Adresse d'envoi de la liste ;

List-Subscribe Adresse d'abonnement à la liste ;

List-Unsubscribe Adresse de désabonnement de la liste ;

List-Help Adresse pour obtenir de l'aide sur la liste ;

List-Archive Adresse de l'archive de la liste.

Mails

Mailing-lists, entêtes spécifiques

RFC 4021 ;

List-ID Identifiant de la liste, permet une classification ;

List-Owner Adresse du propriétaire de la liste ;

List-Post Adresse d'envoi de la liste ;

List-Subscribe Adresse d'abonnement à la liste ;

List-Unsubscribe Adresse de désabonnement de la liste ;

List-Help Adresse pour obtenir de l'aide sur la liste ;

List-Archive Adresse de l'archive de la liste.

Mails

Mailing-lists, entêtes spécifiques

RFC 4021 ;

List-ID Identifiant de la liste, permet une classification ;

List-Owner Adresse du propriétaire de la liste ;

List-Post Adresse d'envoi de la liste ;

List-Subscribe Adresse d'abonnement à la liste ;

List-Unsubscribe Adresse de désabonnement de la liste ;

List-Help Adresse pour obtenir de l'aide sur la liste ;

List-Archive Adresse de l'archive de la liste.

Mails

Mailing-lists, entêtes spécifiques

RFC 4021 ;

List-ID Identifiant de la liste, permet une classification ;

List-Owner Adresse du propriétaire de la liste ;

List-Post Adresse d'envoi de la liste ;

List-Subscribe Adresse d'abonnement à la liste ;

List-Unsubscribe Adresse de désabonnement de la liste ;

List-Help Adresse pour obtenir de l'aide sur la liste ;

List-Archive Adresse de l'archive de la liste.

Mails

Mailing-lists, entêtes spécifiques

RFC 4021 ;

List-ID Identifiant de la liste, permet une classification ;

List-Owner Adresse du propriétaire de la liste ;

List-Post Adresse d'envoi de la liste ;

List-Subscribe Adresse d'abonnement à la liste ;

List-Unsubscribe Adresse de désabonnement de la liste ;

List-Help Adresse pour obtenir de l'aide sur la liste ;

List-Archive Adresse de l'archive de la liste.

Forum

- Site web ;
 - Hierarchie : Catégories, Sujets, Messages ;
 - Souvent des passerelles mails (envoi et gestion des réponses) :
 - Catégories → mailing list ;
 - Sujets → Usenet ;
 - Messages → mail
 - <https://www.hardware.fr>

Forum

- Site web ;
 - Hierarchie : Catégories, Sujets, Messages ;
 - Souvent des passerelles mails (envoi et gestion des réponses) :
 - Catégories → mailing list ;
 - Sujets → Usenet ;
 - Messages → mail
 - <https://www.hardware.fr>

Forum

- Site web ;
- Hierarchie : Catégories, Sujets, Messages ;
- Souvent des passerelles mails (envoi et gestion des réponses) :
 - `mailto:contact@domaine.com`
 - `reply-to: contact@domaine.com`
 - `reply-to: contact@domaine.com`
- <https://www.hardware.fr>

Forum

- Site web ;
- Hierarchie : Catégories, Sujets, Messages ;
- Souvent des passerelles mails (envoi et gestion des réponses) :
 - `mailto:contact@domaine.com`
 - `reply-to: contact@domaine.com`
 - `reply-to: contact@domaine.com`
- <https://www.hardware.fr>

Forum

- Site web ;
- Hierarchie : Catégories, Sujets, Messages ;
- Souvent des passerelles mails (envoi et gestion des réponses) :
 - catégorie => mailing list ;
 - sujet => thread ;
 - message => mail.
- <https://www.hardware.fr>

Forum

- Site web ;
- Hierarchie : Catégories, Sujets, Messages ;
- Souvent des passerelles mails (envoi et gestion des réponses) :
 - catégorie => mailing list ;
 - sujet => thread ;
 - message => mail.
- <https://www.hardware.fr>

Forum

- Site web ;
- Hierarchie : Catégories, Sujets, Messages ;
- Souvent des passerelles mails (envoi et gestion des réponses) :
 - catégorie => mailing list ;
 - sujet => thread ;
 - message => mail.
- <https://www.hardware.fr>

Forum

- Site web ;
- Hierarchie : Catégories, Sujets, Messages ;
- Souvent des passerelles mails (envoi et gestion des réponses) :
 - catégorie => mailing list ;
 - sujet => thread ;
 - message => mail.
- <https://www.hardware.fr>

Forum

- Site web ;
- Hierarchie : Catégories, Sujets, Messages ;
- Souvent des passerelles mails (envoi et gestion des réponses) :
 - catégorie => mailing list ;
 - sujet => thread ;
 - message => mail.
- <https://www.hardware.fr>

Forum

- Site web ;
- Hierarchie : Catégories, Sujets, Messages ;
- Souvent des passerelles mails (envoi et gestion des réponses) :
 - catégorie => mailing list ;
 - sujet => thread ;
 - message => mail.

• <https://www.hardware.fr>

Forum

- Site web ;
- Hierarchie : Catégories, Sujets, Messages ;
- Souvent des passerelles mails (envoi et gestion des réponses) :
 - catégorie => mailing list ;
 - sujet => thread ;
 - message => mail.

● <https://www.hardware.fr>

Forum

- Site web ;
- Hierarchie : Catégories, Sujets, Messages ;
- Souvent des passerelles mails (envoi et gestion des réponses) :
 - catégorie => mailing list ;
 - sujet => thread ;
 - message => mail.
- <https://www.hardware.fr>

Bugtrackers

- **Site web, mailing lists**
- Permet de recenser et suivre les tâches du projet
- Anomalies (bugs) et fonctionnalités
- Moteurs connus :
 - Bugzilla ;
 - Mantis ;
 - Flyspray ;
 - Debian BTS (emails, chaque bug est une liste de diffusion) ;
 - Forges.

Bugtrackers

- Site web, mailing lists
- Permet de recenser et suivre les tâches du projet
- Anomalies (bugs) et fonctionnalités
- Moteurs connus :
 - Bugzilla ;
 - Mantis ;
 - Flyspray ;
 - Debian BTS (emails, chaque bug est une liste de diffusion) ;
 - Forges.

Bugtrackers

- Site web, mailing lists
- Permet de recenser et suivre les tâches du projet
- Anomalies (bugs) et fonctionnalités
- Moteurs connus :
 - Bugzilla ;
 - Mantis ;
 - Flyspray ;
 - Debian BTS (emails, chaque bug est une liste de diffusion) ;
 - Forges.

Bugtrackers

- Site web, mailing lists
- Permet de recenser et suivre les tâches du projet
- Anomalies (bugs) et fonctionnalités
- Moteurs connus :
 - Bugzilla ;
 - Mantis ;
 - Flyspray ;
 - Debian BTS (emails, chaque bug est une liste de diffusion) ;
 - Forges.

Bugtrackers

- Site web, mailing lists
- Permet de recenser et suivre les tâches du projet
- Anomalies (bugs) et fonctionnalités
- Moteurs connus :
 - Bugzilla ;
 - Mantis ;
 - Flyspray ;
 - Debian BTS (emails, chaque bug est une liste de diffusion) ;
 - Forges.

Bugtrackers

- Site web, mailing lists
- Permet de recenser et suivre les tâches du projet
- Anomalies (bugs) et fonctionnalités
- Moteurs connus :
 - Bugzilla ;
 - Mantis ;
 - Flyspray ;
 - Debian BTS (emails, chaque bug est une liste de diffusion) ;
 - Forges.

Bugtrackers

- Site web, mailing lists
- Permet de recenser et suivre les tâches du projet
- Anomalies (bugs) et fonctionnalités
- Moteurs connus :
 - Bugzilla ;
 - Mantis ;
 - Flyspray ;
 - Debian BTS (emails, chaque bug est une liste de diffusion) ;
 - Forges.

Bugtrackers

- Site web, mailing lists
- Permet de recenser et suivre les tâches du projet
- Anomalies (bugs) et fonctionnalités
- Moteurs connus :
 - Bugzilla ;
 - Mantis ;
 - Flyspray ;
 - Debian BTS (emails, chaque bug est une liste de diffusion) ;
 - Forges.

Forges de développement

- 1 Introduction
- 2 Outils de Développement
- 3 Gestion de la Documentation
- 4 Outils de Communication
- 5 Forges de développement**
- 6 Fonctionnement de certains projets

Informations générales

- Plateformes Web dédiées au développement ;
- Intégration de différents outils :
 - Site web dédié au projet ;
 - Gestionnaire de version ;
 - Bugtracker ;
 - Gestionnaire de documentation (wiki) ;
 - Listes de discussion, forums.

Informations générales

- Plateformes Web dédiées au développement ;
- Intégration de différents outils :
 - Site web dédié au projet ;
 - Gestionnaire de version ;
 - Bugtracker ;
 - Gestionnaire de documentation (wiki) ;
 - Listes de discussion, forums.

Informations générales

- Plateformes Web dédiées au développement ;
- Intégration de différents outils :
 - Site web dédié au projet ;
 - Gestionnaire de version ;
 - Bugtracker ;
 - Gestionnaire de documentation (wiki) ;
 - Listes de discussion, forums.

Informations générales

- Plateformes Web dédiées au développement ;
- Intégration de différents outils :
 - Site web dédié au projet ;
 - Gestionnaire de version ;
 - Bugtracker ;
 - Gestionnaire de documentation (wiki) ;
 - Listes de discussion, forums.

Informations générales

- Plateformes Web dédiées au développement ;
- Intégration de différents outils :
 - Site web dédié au projet ;
 - Gestionnaire de version ;
 - Bugtracker ;
 - Gestionnaire de documentation (wiki) ;
 - Listes de discussion, forums.

Informations générales

- Plateformes Web dédiées au développement ;
- Intégration de différents outils :
 - Site web dédié au projet ;
 - Gestionnaire de version ;
 - Bugtracker ;
 - Gestionnaire de documentation (wiki) ;
 - Listes de discussion, forums.

Informations générales

- Plateformes Web dédiées au développement ;
- Intégration de différents outils :
 - Site web dédié au projet ;
 - Gestionnaire de version ;
 - Bugtracker ;
 - Gestionnaire de documentation (wiki) ;
 - Listes de discussion, forums.

Logiciels (autohébergement)

- **SourceForge et forks :**

- FusionForge (ex-GForge) ;
- Savane ;
- Berlios ;
- TeamForge (propriétaire).

- Trac ;

- Redmine/Chiliproject ;

- Gitlab (double licence communauté et entreprise) ;

- Gogs ;

- Phabricator ;

- Github (propriétaire).

Logiciels (autohébergement)

- SourceForge et forks :
 - FusionForge (ex-GForge) ;
 - Savane ;
 - Berlios ;
 - TeamForge (propriétaire).
- Trac ;
- Redmine/Chiliproject ;
- Gitlab (double licence communauté et entreprise) ;
- Gogs ;
- Phabricator ;
- Github (propriétaire).

Logiciels (autohébergement)

- SourceForge et forks :
 - FusionForge (ex-GForge) ;
 - Savane ;
 - Berlios ;
 - TeamForge (propriétaire).
- Trac ;
- Redmine/Chiliproject ;
- Gitlab (double licence communauté et entreprise) ;
- Gogs ;
- Phabricator ;
- Github (propriétaire).

Logiciels (autohébergement)

- SourceForge et forks :
 - FusionForge (ex-GForge) ;
 - Savane ;
 - Berlios ;
 - TeamForge (propriétaire).
- Trac ;
- Redmine/Chiliproject ;
- Gitlab (double licence communauté et entreprise) ;
- Gogs ;
- Phabricator ;
- Github (propriétaire).

Logiciels (autohébergement)

- SourceForge et forks :
 - FusionForge (ex-GForge) ;
 - Savane ;
 - Berlios ;
 - TeamForge (propriétaire).
- Trac ;
- Redmine/Chiliproject ;
- Gitlab (double licence communauté et entreprise) ;
- Gogs ;
- Phabricator ;
- Github (propriétaire).

Logiciels (autohébergement)

- SourceForge et forks :
 - FusionForge (ex-GForge) ;
 - Savane ;
 - Berlios ;
 - TeamForge (propriétaire).
- Trac ;
- Redmine/Chiliproject ;
- Gitlab (double licence communauté et entreprise) ;
- Gogs ;
- Phabricator ;
- Github (propriétaire).

Logiciels (autohébergement)

- SourceForge et forks :
 - FusionForge (ex-GForge) ;
 - Savane ;
 - Berlios ;
 - TeamForge (propriétaire).
- Trac ;
- Redmine/Chiliproject ;
- Gitlab (double licence communauté et entreprise) ;
- Gogs ;
- Phabricator ;
- Github (propriétaire).

Logiciels (autohébergement)

- SourceForge et forks :
 - FusionForge (ex-GForge) ;
 - Savane ;
 - Berlios ;
 - TeamForge (propriétaire).
- Trac ;
- Redmine/Chiliproject ;
- Gitlab (double licence communauté et entreprise) ;
- Gogs ;
- Phabricator ;
- Github (propriétaire).

Logiciels (autohébergement)

- SourceForge et forks :
 - FusionForge (ex-GForge) ;
 - Savane ;
 - Berlios ;
 - TeamForge (propriétaire).
- Trac ;
- Redmine/Chiliproject ;
- Gitlab (double licence communauté et entreprise) ;
- Gogs ;
- Phabricator ;
- Github (propriétaire).

Logiciels (autohébergement)

- SourceForge et forks :
 - FusionForge (ex-GForge) ;
 - Savane ;
 - Berlios ;
 - TeamForge (propriétaire).
- Trac ;
- Redmine/Chiliproject ;
- Gitlab (double licence communauté et entreprise) ;
- Gogs ;
- Phabricator ;
- Github (propriétaire).

Plateforme publiques

- <http://sourceforge.net> : CVS, SVN, Git;
- <http://savannah.gnu.org> : GNU Arch, GNU Bazaar, CVS, Git, Mercurial, Subversion;
- <http://gna.org> : CVS, GNU Arch, Subversion;
- <http://github.com> : Git;
- <http://gitlab.com> (ex Gitorious) : Git.
- <http://framagit.org> (Gitlab)

Plateforme publiques

- <http://sourceforge.net> : CVS, SVN, Git;
- <http://savannah.gnu.org> : GNU Arch, GNU Bazaar, CVS, Git, Mercurial, Subversion;
- <http://gna.org> : CVS, GNU Arch, Subversion;
- <http://github.com> : Git;
- <http://gitlab.com> (ex Gitorious) : Git.
- <http://framagit.org> (Gitlab)

Plateforme publiques

- <http://sourceforge.net> : CVS, SVN, Git;
- <http://savannah.gnu.org> : GNU Arch, GNU Bazaar, CVS, Git, Mercurial, Subversion;
- <http://gna.org> : CVS, GNU Arch, Subversion;
- <http://github.com> : Git;
- <http://gitlab.com> (ex Gitorious) : Git.
- <http://framagit.org> (Gitlab)

Plateforme publiques

- <http://sourceforge.net> : CVS, SVN, Git;
- <http://savannah.gnu.org> : GNU Arch, GNU Bazaar, CVS, Git, Mercurial, Subversion;
- <http://gna.org> : CVS, GNU Arch, Subversion;
- <http://github.com> : Git;
- <http://gitlab.com> (ex Gitorious) : Git.
- <http://framagit.org> (Gitlab)

Plateforme publiques

- <http://sourceforge.net> : CVS, SVN, Git ;
- <http://savannah.gnu.org> : GNU Arch, GNU Bazaar, CVS, Git, Mercurial, Subversion ;
- <http://gna.org> : CVS, GNU Arch, Subversion ;
- <http://github.com> : Git ;
- <http://gitlab.com> (ex Gitorious) : Git.
- <http://framagit.org> (Gitlab)

Plateforme publiques

- <http://sourceforge.net> : CVS, SVN, Git;
- <http://savannah.gnu.org> : GNU Arch, GNU Bazaar, CVS, Git, Mercurial, Subversion;
- <http://gna.org> : CVS, GNU Arch, Subversion;
- <http://github.com> : Git;
- <http://gitlab.com> (ex Gitorious) : Git.
- <http://framagit.org> (Gitlab)

Fonctionnement de certains projets

- 1 Introduction
- 2 Outils de Développement
- 3 Gestion de la Documentation
- 4 Outils de Communication
- 5 Forges de développement
- 6 Fonctionnement de certains projets**

Debian

Type Distribution GNU/Linux ;

Site web <https://www.debian.org> ;

IRC OFTC : #debian et #debian-fr ;

Bugtracker <https://www.debian.org/Bugs>, mail: request@bugs.debian.org ;

Mailing-lists <https://lists.debian.org/> :

debian-announce Informations capitales et changements importants dans le projet, modérée ;

debian-backports Discussion sur les backports, non modérée ;

debian-user Aide pour les utilisateurs de Debian, volume important, non modérée ;

debian-user-french Aide francophone pour les utilisateurs de Debian ;
Liste complète sur
<https://www.debian.org/MailingLists/subscribe>.

Forge <https://alioth.debian.org> (GForge)

Contribution <https://www.debian.org/devel/join/>



Linux

Type Noyau du système d'exploitation ;

Site web <https://www.kernel.org> ;

Patches <https://patchwork.kernel.org> ;

Bugtracker <https://bugzilla.kernel.org/> ;

Mailing-lists <http://vger.kernel.org/> ;

Wikis <https://www.wiki.kernel.org/> ;

Git <https://git.kernel.org>.

KDE

Type Gestionnaire de bureaux, applications bureautiques ;

Site Web <https://www.kde.org> ;

Wiki utilisateurs <https://userbase.kde.org> ;

Wiki admin <https://techbase.kde.org> ;

Wiki communauté <https://community.kde.org> ;

Forum <https://forum.kde.org/> ;

Mailing lists <https://www.kde.org/support/maillinglists/>

Bugtracker <https://bugs.kde.org/> ;

Git <https://community.kde.org/Infrastructure/Git> ;

Contribuer https://community.kde.org/Get_Involved.