

Gestion de Projet, Traitements de données...

Conception, Développement, Tests et Production


Gabriel CHANDESRIS

BIOLOGIE & INFORMATIQUE

`gabriel.ohandesris@laposte.net`

06 56 41 97 37

`http://gabriel.ohandesris.free.fr/`



08 Novembre 2010



Sommaire

Présentation Perso

Axes principaux

■Pack Bonus■

Questions



Présentation Perso

Présentation Perso

Parcours de formation

BTS Biochimie (2003)

Licence Bio-Informatique au CNAM (2005 – 2007)

IUP GBI à Evry - Genopole (2009)

Parcours professionnel

Laboratoires, Hôpitaux...

Recherche Publique

Recherche Privée / Industrie

Projets

Modélisation Biologique

Simulation Informatique

Un soupçon de 3D...



Parcours de formation

Présentation Perso

Parcours de formation

BTS Biochimie (2003)

Licence Bio-Informatique au CNAM (2005 – 2007)

IUP GBI à Evry - Genopole (2009)

Parcours professionnel

Projets



BTS Biochimie (2003)



BTS Biochimie (2003)

- ▶ Biochimie / chimie organique...
- ▶ Bactériologie / Microbiologie...
- ▶ Immunologie, contrôles qualité...
- ▶ Techniques classiques et Automates d'analyses...
- ▶ Autant de théorie que de pratique !!



Licence Bio-Informatique au CNAM (2005 – 2007)

Licence Bio-Informatique au CNAM (2005 – 2007)

- ▶ **Cours du soir**, Conservatoire National des Arts et Métiers.
- ▶ Remise à plat des concepts d'algorithmique et de programmation (Scheme – LISP, JAVA).
- ▶ Contacts professionnels (pas seulement universitaires).
- ▶ Axé outils de bio-informatique.





IUP GBI à Evry - Genopole (2009)



IUP GBI à Evry - Genopole (2009)

- ▶ ...
- ▶ Très (trop) universitaire.
- ▶ Ne pas hésiter à voir à côté (pure info. et pure biologie).
- ▶ Les stages sont des plus (++) non négligeables à considérer comme expérience pro. !!



Parcours professionnel

Présentation Perso

Parcours de formation

Parcours professionnel

Laboratoires, Hôpitaux...

Recherche Publique

Recherche Privée / Industrie

Projets



Laboratoires, Hôpitaux...



Laboratoires, Hôpitaux...

- ▶ Militaire, privé, public.
- ▶ Santé "générale" (public / semi-public > privé : "rareté").
- ▶ Gestion particulière, convention et contrôle...
- ▶ Techniques automatisées.



Recherche Publique

- ▶ CNRS : IBISC, Evry
- ▶ Professeurs de l'IUP GBI et du CNAM.
- ▶ ...





sanofi aventis

L'essentiel c'est la santé.



Recherche Privée / Industrie

- ▶ Sanofi-Aventis (Pharmacie) : études cliniques, génétiques, brevets...
- ▶ Dassault Systèmes (PLM, aéronautique, automobile...) :
 - ▶ Extension vers les Sciences de la Vie.
 - ▶ Projet BioIntelligence (socle + partenaires).
- ▶ EDD (traitement de données des entreprises et de la presse).



Projets

Présentation Perso

Parcours de formation

Parcours professionnel

Projets

Modélisation Biologique

Simulation Informatique

Un soupçon de 3D...



Modélisation Biologique

Select a defined Gene

BRGlyc01: Glucose=>G6P
BRGlyc02: Fructose=>F6P
BRGlyc03: G6P=>F6P
BRGlyc04: G6P=>F6P
BRGlyc05: F6P=>FPP
BRGlyc06: FPP=>DHAP+P
BRGlyc07: DHAP+PGAL=>
BRGlyc08: DHAP=>PGAL
BRGlyc09: PGAL=>DHAP
BRGlyc10: PGAL=>13PPGL
BRGlyc11: 13PPGLY=>PG
BRGlyc12: 13PPGLY=>3PG
BRGlyc13: 3PGLY=>13PPG
BRGlyc14: 3PGLY=>2PGLY
BRGlyc15: 2PGLY=>3PGLY
BRGlyc16: 2PGLY=>PePyr
BRGlyc17: PePyr+H2O=>2
BRGlyc18: PePyr+ADP=>F
BRGlyc19: Pyr+CoA=>CO
BRGlyc20: CO2+NAD=>CC
BRLY400: AG39=>AG40
BRLY401: AG40=>AG40Co
BRLY402: AG40CoA=>AG4
BRLY403: AG40CoA+FAD=

Delete Gene

Existent Pathways :

Select a metabolic way

BiochemicalReaction

Choose a type of Gene

InitialConcentration

BiochemicalReaction

BrainGene

BrainLobeGene

EmitterReceptor

StimulusDecision

Instinct

Create Gene

Change Gene

As new Gene

Parameters

☒ duplic ☒ delete ☒ actif

000 Max. age : 999

0 Mutation rate (%) : 25

Parameters

001

with

Bcoef : 001

=> to give at speed

Ccoef : 001

+ (and)

Dcoef : 001

* Achem : 000

* Bchem : 000

KM : 001

* Cchem : 000

* Dchem : 000

Gene Name :

List of Chemicals :

0 <NONE00> <NONE00>

Select a defined...

Rem. Gene

Path Name :


Add Gene to Path

Create PathWay














Simulation Informatique

17-09-2010 15:16:09 127.0.0.1 0...50...100



LOGIN CONNECTION
IP address : 127.0.0.1
Login :
Password :



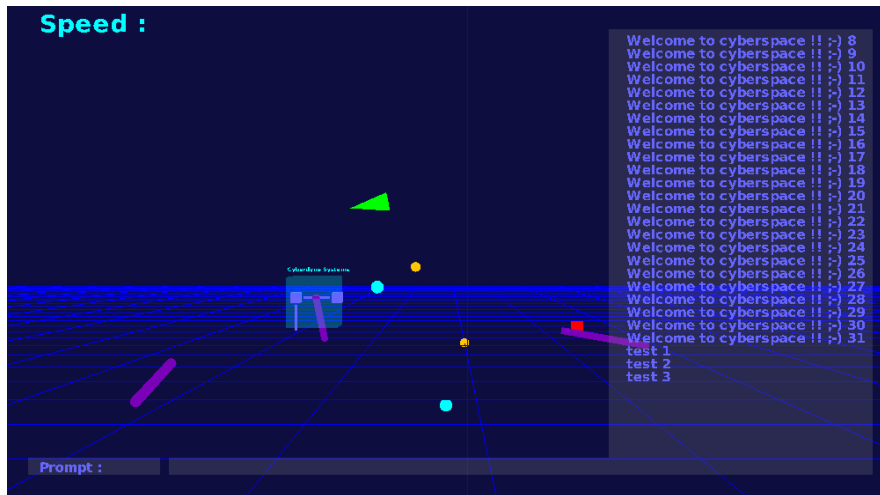
general info

Sample

Welcome to Terminal Urbs !!
Initiate DNS system.
Some default Servers...
Reading data about servers, users
Some random Servers to be added
Some User Agents to be added ?
Default connection is [127.0.0.1].



Un soupçon de 3D...





Axes principaux

Axes principaux

Gestion de projet

Conception (Réflexion et UML...) –
Développement et tests
Déploiement ([Pré-]Production)

Outils et Méthodes

Outils (IDE : Eclipse et équivalents, Qualité)
Méthodes (eXtreme Prog. , Agilité, Plannings)
Intérêts : Prototypage ET Industrialisation !

Traitements De Données

Formats De Données
Bases De Données
Autres...

Programmation

Impératif / Fonctionnel
Orienté Objet (POO)
Autres...



Axes principaux

Gestion de projet

Conception (Réflexion et UML...) –

Développement et tests

Déploiement ([Pré-]Production)

Outils et Méthodes

Traitements De Données

Programmation



Conception (Réflexion et UML...) – (1)

- ▶ Conception : établir un **cahier des charges** :
 - ▶ Rassembler les idées, concepts...
 - ▶ Organiser et planifier un ou plusieurs projets.
 - ▶ Papier, crayons ou équivalents.
- ▶ **Diagrammes UML** (*Unified Modeling Language*) :
 - ▶ Un dessin vaut parfois mieux qu'un long discours, avec (ou sans) explication(s).
 - ▶ Tableau(x) récapitulatif(s) d'exemples de données ou de nomenclature.
 - ▶ Différents aspects d'un même problème et de sa résolution.
 - ▶ **Classes, Cas d'utilisation, Séquence, Activités...**
- ▶ **Design Patterns** : *réfléchir à l'implémentation sans écrire une seule ligne de code !*
- ▶ **MVC** : *Modèle-Vue-Contrôleur*, et **API** : *Application Programming Interface* : réflexion sur la séparation des composants, ce qui interviendra lors du développement (modules, packages, GUI utilisateurS...).



Les diagrammes UML

Le diagramme de Classes

Le diagramme de Cas
d'Utilisation

Le diagramme de Séquences

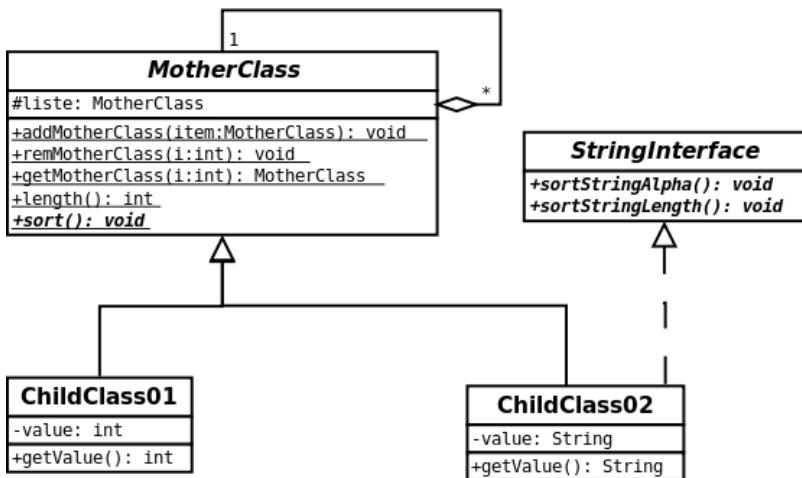
Le diagramme d'État et / ou
d'Activités

Autres Diagrammes UML : État-Transition ; Composants ;
Communication ; Structures composites...



Conception (Réflexion et UML...) – (4)

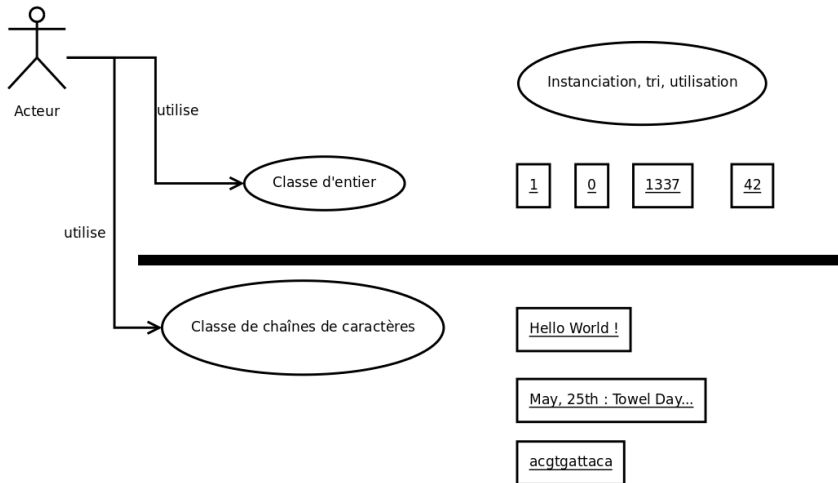
Le diagramme de Classes





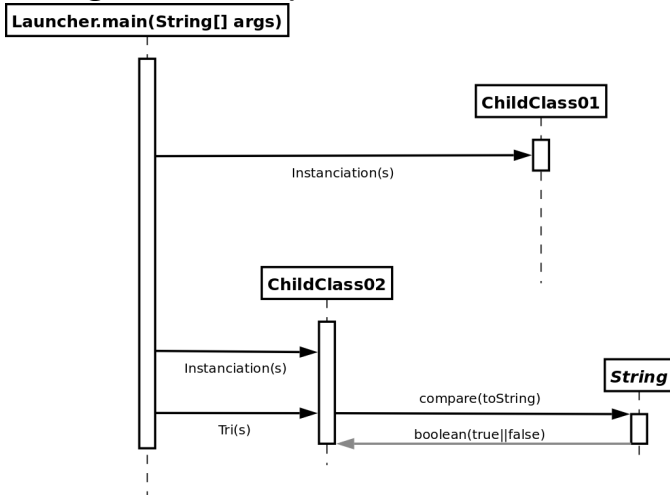
Conception (Réflexion et UML...) – (5)

Le diagramme de Cas d'Utilisation



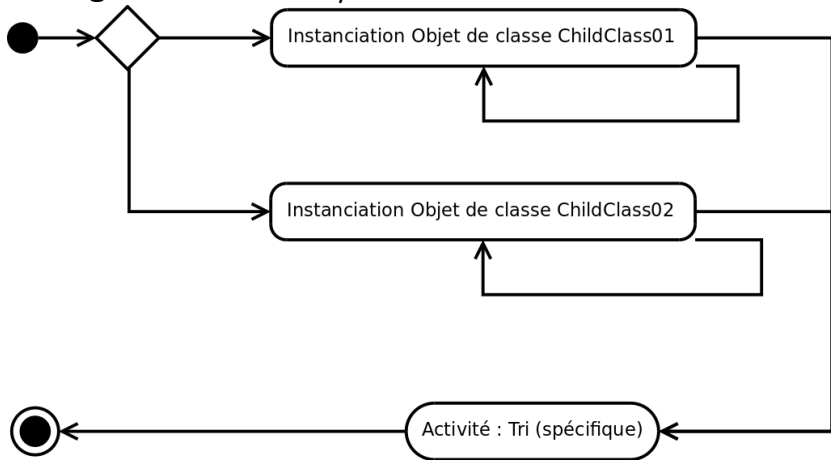


Le diagramme de Séquences





Le diagramme d'État et / ou d'Activités





Les Design-Pattern's

Regroupements

- ▶ Interfaces (cf. Structures)
- ▶ Responsabilité (cf. Comportement)
- ▶ Construction (cf. Création)
- ▶ Opérations
- ▶ Extensions

Création

- ▶ Abstractions et héritage
- ▶ Factorisation,
- ▶ Builder, Prototype...

Structure

- ▶ Granularité des classes et instances (Flyweight),
- ▶ Accès aux instances (Bridge, Façade...),
- ▶ Adapter...

Comportements

- ▶ Itération,
- ▶ Médiation,
- ▶ Chaînes / Responsibility...



Développement et tests

1. Documentation (besoins, cahier des charges, notes, diagrammes...).
2. Implémentation (documentation interne).
3. Tests (unitaires, élaborés...).
4. Du *Back Office* au *Front Office* : rapports de bugs, re-implémentation, nouveaux besoins...
5. *Industrialisation* (utilisation courante, conditions logicielles et / ou matérielles...).
6. (Pré-)Production et mise en service...

Plus de détails dans le slide suivant et la partie suivante (*Outils et Méthodes*).



Déploiement ([Pré-]Production)

Partie finale du cycle en V ou Y (retour vers une re-définition ou ajout de besoins, fonctionnalités... pour une correction de bug ou une version suivante!).

Industrialisation

- ▶ Modifier l'implémentation pour une utilisation courante
- ▶ Enlever (ou rendre inaccessibles) les parties de tests.
 - ▶ les parties de tests,
 - ▶ le code source (propriétaire ; libre, ouvert...).
- ▶ Facile à (re-)installer (version via CVS / SVN – **subversion**).

(Pré-)Production

- ▶ Tests en grandeur / taille réelle.
- ▶ Mise en fonctionnement courant (*Front Office*).
- ▶ Distribution auprès de l'utilisateur (installation).
- ▶ Accessibilité auprès de l'utilisateur (service web...).



Outils et Méthodes

Axes principaux

Gestion de projet

Outils et Méthodes

Outils (IDE : Eclipse et équivalents, Qualité)

Méthodes (eXtreme Prog. , Agilité, Plannings)

Intérêts : Prototypage ET Industrialisation !

Traitements De Données

Programmation



Outils (IDE : Eclipse et équivalents, Qualité)– 1

- ▶ **Environnements de Développements Intégrés** : outils d'aides.
 1. Conception, développement.
 2. Modèles de classes et packages.
 3. Reconnaissance de langage(s) et mise en valeurs de mots-clés.
 4. Signaux d'avertissements, d'erreurs...
- ▶ Modules complémentaires (pour Eclipse : Subclipse, EPIC, C / C++, PyDev, XML...) et autres IDE (xCode, Visual Studio...).
- ▶ Gestion du temps, projet(s), versions (SVN), Intégration de méthodes de gestion de projets.



► "Démarche qualité"

1. Écrire et décrire ce que l'on fait et réalise.
2. Validation progressive (points de blocage, points réalisés, erreurs connues...).
3. Des normes ISO existent (9000-9100 pour la démarche qualité, 9100-9999 pour les langages, 27000 pour la sécurité des informations...).

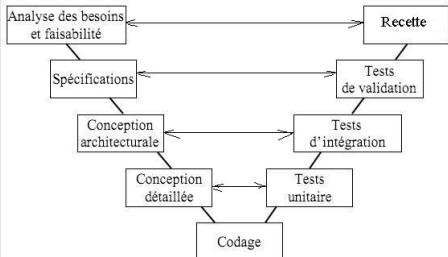
- Beaucoup de documentation supplémentaire mais ajoute la vérification.
- *Outil facultatif / optionnel.*



Méthodes (eXtreme Prog. , Agilité, Plannings)– 1

Quelques indications...

- ▶ eXtreme Programming :
 - ▶ Itérations,
 - ▶ alternance documentation et implémentation.
- ▶ Méthode Scrum.
 - ▶ Cycle de développement (sur une semaine / un mois).
 - ▶ Travail en équipe (répartition dans le temps et entre membres).
- ▶ Méthodes "agiles" (adaptation).
- ▶ **Versions successives.**

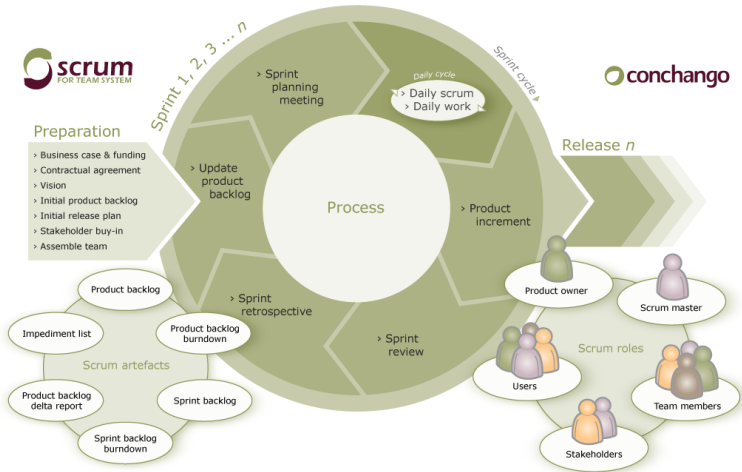


Le cycle en V, un classique...

Voir notamment [http://fr.wikipedia.org/wiki/Scrum_\(methode\)](http://fr.wikipedia.org/wiki/Scrum_(methode)) ainsi que les articles ET références liés.



Méthodes (eXtreme Prog. , Agilité, Plannings)– 2



Le déroulement de SCRUM : étapes du cycle.



Intérêts : Prototypage ET Industrialisation !

Pour quoi se compliquer la vie ?

Les devises Shadok



Attitudes réflexives

Les devises Shadok



EN ESSAYANT CONTINUUELLEMENT
ON FINIT PAR RÉUSSIR. DONC:
PLUS ÇA RATE, PLUS ON A
DE CHANCES QUE ÇA MARCHÉ.

Sciences expérimentales

Les devises Shadok



Observation réfléchie

Les devises Shadok



LA NOTION DE PASSOIRE
EST INDÉPENDANTE DE
LA NOTION DE TROU.

De la notion au concept

Les devises Shadok



Expérimentations

La devise Shadok du mois



IL VAUT MIEUX MOBILISER
SON INTELLIGENCE SUR DES
BÊTISES QUE MOBILISER
SA BÊTISE SUR DES CHOSSES
INTELLIGENTES.

Métacognition

- Faisabilité / Prototype : preuve de concept (en début de projet).
- Industrialiser : faire fonctionner couramment.
- Pouvoir faire évoluer le projet sans tout refaire.
- L'effet (besoin) n'est pas indépendant de la faisabilité de la cause (réponse).
- **Projets mal conçus donc non maintenables, jetés...**
- Éviter le coin de table (projet et douleur).



Traitements De Données

Axes principaux

Gestion de projet

Outils et Méthodes

Traitements De Données

Formats De Données

Bases De Données

Autres...

Programmation



Formats De Données

Formats de fichiers !!

1. Fichiers "plats" : texte brut ;
tabulaire, structuration
spécifique, scripts...
 - ▶ FASTA, EMBL, GenBank...
2. Fichiers XML :
semi-structuration, définition
(DTD, XSD).
 - ▶ UniprotKB...
3. Fichiers binaires (issus de
compilation, instances
sérialisées...).
4. Répertoires, liens symboliques
(via le système d'exploitation)...
5. Bases De Données (selon
SGBD).

Extraction de données, conversions...

- ▶ *Parsers* pré-existants (BioPerl, BioPython...)
- ▶ Moteurs d'extractions
génériques (pour le XML si
définition, XPath).
- ▶ SGBD : MySQL, Postgre,
Oracle → particularismes.
- ▶ Construire son propre système
d'extraction de données (autant
voire plus de temps à utiliser).



- ▶ **Intérêt** : structurer et trouver facilement une donnée selon certains critères, autrement que par un *find* ou un *grep* ou tout autre équivalent...
- ▶ **Optimisation** du stockage de données en volumétrie et en liens (XML très verbeux et redondant, fichiers plats faiblement reliés entre eux...), mais peut être moins efficace (rare).
- ▶ Couplage à une ou plusieurs interfaces utilisateurs / programmes via des connecteurs spécifiques (PHP, Python, Perl, Java...).
- ▶ **Entités-Relations** et SQL : intérêt et utilité de UML.
 - ▶ Bien définir les entités et leurs liens.
 - ▶ Éviter les confusions à venir sur l'utilisation de(s) la base(s) de données (schéma, table, colonne).



Autres...

- ▶ Choix d'implémentation : quantité de données, répartition, utilisation, codage, rapidité, enregistrement, mémoire...
- ▶ **Berkeley** : couples (clef; valeur) et structures similaires.
- ▶ **SGBDR** : le plus classique (SQL).
- ▶ **SGBDO** : du Relationnel à l'Objet : lourd et peu utilisé (sérialisation d'instances d'objets).
- ▶ **ORM** – Object-Relationnal-Mapping : relier une Programmation Objet et un SGBDR sans écrire du SQL !!



Programmation

Axes principaux

Gestion de projet

Outils et Méthodes

Traitements De Données

Programmation

Impératif / Fonctionnel

Orienté Objet (POO)

Autres...



Impératif / Fonctionnel

- ▶ **Intérêt** : Chaînes de traitement ETL (Extraction, Traitement, Chargement)
- ▶ **Langages** : Perl, Python, Shell et dérivés (bash, ksh...).
- ▶ L'objectif est ici la rapidité de traitement dans la conversion des données, la Programmation Orientée Objet est *facultative* dans ce cas.
- ▶ Aide d'outils adaptés : *ImageMagick* pour les images, *gs* pour les PDF, *libxml...*).
- ▶ Utilisation de l'ordinateur comme d'un automate avec une grosse mémoire (événements, base de données...).



Orienté Objet (POO)

- ▶ **Intérêt** : Représentation des données et leur manipulation.
- ▶ **Langages** : Java, C++, PHP5...
- ▶ Indispensable pour bien conceptualiser !!
- ▶ Code compréhensible humainement, maintenable, factorisable...
- ▶ Structures communes (héritage, interfaces, Design-Patterns).
- ▶ **Représentation, Classification, Manipulation !!**
- ▶ Comme pour représentation ER : *si on ne comprend pas bien, ne pas faire...*



1. *Trigger* de Bases De Données

- ▶ **Intérêt** : fonctions construites en BDD pour la manipulation des données.
- ▶ **Langages** : SQL, C...

2. XSLT

- ▶ **Intérêt** : Manipulation directe des données dans leur(s) format(s).
- ▶ **Langages** : XML.

3. Documentation : Wiki, \LaTeX , Code Source...

- ▶ **Intérêts** : relecture du code, reprise du projet, maintenabilité du projet et du code source...
- ▶ **Langages** : \TeX / \LaTeX , RST, JavaDoc et assimilés,
- ▶ **Indispensable !!**



■Pack Bonus■

■Pack Bonus■

Information !!

InterWeb et Librairies

E-Mail et NewsGroup / Forum

Veille Info / Techno

Hackez !! (bidouille)

Maître devenu tu es, Padawan tu restes...

Participation à des projets

Biologie ++ Informatique

Prospective

Réformes (législatives, sociales)...

Investissements Industriels

Initiative(s)...



Information !!

■Pack Bonus■

Information !!

InterWeb et Librairies

E-Mail et NewsGroup / Forum

Veille Info / Techno

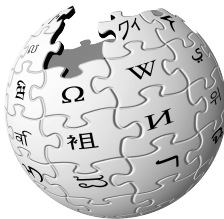
Hackez !! (bidouille)

Prospective



InterWeb et Librairies

- ▶ Librairie Réelles ET Logicielles *explorer / tester*.
- ▶ *En BioInfo* BioPerl, BioPython, BioJava, BioC++...
- ▶ *[http ://www.dmoz.org/Science/Biology/Bioinformatics/Software/](http://www.dmoz.org/Science/Biology/Bioinformatics/Software/)*
- ▶ Encyclopédies : périmées dès la publication : *approfondir*.



*Un point de départ, ne pas s'en contenter,
y participer éventuellement.*



E-Mail et NewsGroup / Forum



UseNet et NewsGroups : posez des questions, répondez-y aussi !!

La liste bioinfo : <http://listes.sfbi.fr/www/info/bioinfo>

Groupes UseNet et Google Groups (Listes Python et Perl).

- ▶ <http://www.usenet-fr.net/> : fr.usenet.usages, fr.sci.biologie...
- ▶ sci.bio.technology, bionet.celegans, alt.bio.ethics, alt.bio.hackers...
- ▶ <http://news.lacave.net/servers/reader/list>



- ▶ Faire l'effort de s'informer, même de façon passive (e-mail, newsgroup, RSS, journaux...) : cas précédents.
- ▶ Méthodologie active (exemples) :
 - ▶ maintenir une bibliographie, si possible à jour, étendue (sources primaires et secondaires).
 - ▶ tenir un journal ou un blog (*DLFP*,
<http://duvernoisevelyne.blog.rhonealpesjob.com/>,
<http://www.biologeeek.com/>)...
- ▶ Vitae \neq Mortae : maintenir son Curriculum à jour.
- ▶ Informer (journal, blog : accès public ou restreint).



- ▶ Faire l'effort de s'informer, même de façon passive (e-mail, newsgroup, RSS, journaux...) : cas précédents.
- ▶ Méthodologie active (exemples) :
 - ▶ maintenir une bibliographie, si possible à jour, étendue (sources primaires et secondaires).
 - ▶ tenir un journal ou un blog (*DLFP*,
<http://duvernoisevelyne.blog.rhonealpesjob.com/>,
<http://www.biologeeek.com/>)...
- ▶ Vitae \neq Mortae : maintenir son Curriculum à jour.
- ▶ Informer (journal, blog : accès public ou restreint).



- ▶ Faire l'effort de s'informer, même de façon passive (e-mail, newsgroup, RSS, journaux...) : cas précédents.
- ▶ Méthodologie active (exemples) :
 - ▶ maintenir une bibliographie, si possible à jour, étendue (sources primaires et secondaires).
 - ▶ tenir un journal ou un blog (*DLFP*,
<http://duvernoisevelyne.blog.rhonealpesjob.com/>,
<http://www.biologeeek.com/>)...
- ▶ Vitae \neq Mortae : maintenir son Curriculum à jour.
- ▶ Informer (journal, blog : accès public ou restreint).



Hackez !! (bidouille)

■Pack Bonus■

Information !!

Hackez !! (bidouille)

Maître devenu tu es, Padawan tu restes...

Participation à des projets

Biologie ++ Informatique

Prospective



Maître devenu tu es, Padawan tu restes...

Maître devenu tu es, Padawan tu restes...

- ▶ Diplôme : reconnaissance ultime ?
 - ▶ Pas forcément demandé dans le privé (expérience ET période d'essai).
 - ▶ Un projet de trois ans vaut bien un doctorat, un diplôme d'ingénieur, une compétence acquise...
- ▶ Apprentissage permanent (curiosité, formations diplomantes ou non...).
- ▶ *L'attitude n'est pas un substitut à la compétence.*





Participation à des projets



Respect des normes et standards.

Participation à des projets

- ▶ HackerSpaces.
 - ▶ Lieux de partages (information, électronique...).
 - ▶ Do It Yourself (prototypage, manuels...).
 - ▶ DIY Bio (essor actuel : matériel, éthique...).
 - ▶ **/tmp/lab** <http://www.tmplab.org/>
- ▶ Projets et initiatives personnels.
- ▶ Logiciels Libres // Open Source.
- ▶ "Biologie Libre // Open Source".



Participation à des projets



Respect des normes et standards.

Participation à des projets

► **HackerSpaces.**

- Lieux de partages (information, électronique...).
- Do It Yourself (prototypage, manuels...).
- DIY Bio (essor actuel : matériel, éthique...).
- **/tmp/lab** [http ://www.tmplab.org/](http://www.tmplab.org/)

► Projets et initiatives personnels.

► Logiciels Libres // Open Source.

► "Biologie Libre // Open Source".



Participation à des projets



Respect des normes et standards.

Participation à des projets

► **HackerSpaces.**

- Lieux de partages (information, électronique...).
- Do It Yourself (prototypage, manuels...).
- DIY Bio (essor actuel : matériel, éthique...).
- **/tmp/lab** [http ://www.tmplab.org/](http://www.tmplab.org/)

► Projets et initiatives personnels.

- Logiciels Libres // Open Source.
- "Biologie Libre // Open Source".



Participation à des projets



Respect des normes et standards.

Participation à des projets

- ▶ HackerSpaces.
 - ▶ Lieux de partages (information, électronique...).
 - ▶ Do It Yourself (prototypage, manuels...).
 - ▶ DIY Bio (essor actuel : matériel, éthique...).
 - ▶ **/tmp/lab** [http ://www.tmplab.org/](http://www.tmplab.org/)
- ▶ Projets et initiatives personnels.
- ▶ Logiciels Libres // Open Source.
- ▶ "Biologie Libre // Open Source".



Biologie ++ Informatique

- ▶ ■Homologies■ entre les deux (systèmes complexes, fonctionnement...).
- ▶ Compétences d'analyse, de conception.
- ▶ Aspects scientifiques >> purement informatique...
- ▶ Construction mutuelle (limitations techniques et usages).
- ▶ (*Steve Jobs*) : s'il démarrait Apple maintenant ce serait dans les biotech's...





Biologie ++ Informatique

- ▶ ■Homologies■ entre les deux (systèmes complexes, fonctionnement...).
- ▶ Compétences d'analyse, de conception.
- ▶ Aspects scientifiques >> purement informatique...
- ▶ Construction mutuelle (limitations techniques et usages).
- ▶ (*Steve Jobs*) : s'il démarrait Apple maintenant ce serait dans les biotech's...





Biologie ++ Informatique

Biologie ++ Informatique

- ▶ ■Homologies■ entre les deux (systèmes complexes, fonctionnement...).
- ▶ Compétences d'analyse, de conception.
- ▶ Aspects scientifiques >> purement informatique...
- ▶ Construction mutuelle (limitations techniques et usages).
- ▶ (*Steve Jobs*) : s'il démarrait Apple maintenant ce serait dans les biotech's...





Biologie ++ Informatique

- ▶ ■Homologies■ entre les deux (systèmes complexes, fonctionnement...).
- ▶ Compétences d'analyse, de conception.
- ▶ Aspects scientifiques >> purement informatique...
- ▶ Construction mutuelle (limitations techniques et usages).
- ▶ (*Steve Jobs*) : s'il démarrait Apple maintenant ce serait dans les biotech's...





Prospective

■Pack Bonus■

Information !!

Hackez !! (bidouille)

Prospective

Réformes (législatives, sociales)...

Investissements Industriels

Initiative(s)...



Réformes (législatives, sociales)...

Réformes (législatives, sociales)...

- ▶ Université et recherche française, financements (voir : Angleterre, Allemagne).
- ▶ Liens public-privé (adéquation des formations et des compétences).
 - ▶ Construction mutuelle (limitations techniques et usages).
 - ▶ Soutiens mutuels (formations, ressources).
- ▶ Apprentissage personnel (autodidactes) et "privatisation de l'enseignement".



Omnes Docet Ubique
Enseigner à Tous et en
Tous Lieux
(Abbé Grégoire 1794)



Investissements Industriels

- ▶ **Besoin précis / ponctuels de recherche.**
 - ▶ Faisabilité, preuve de concept.
 - ▶ Marge de proposition / recherche fondamentale.
- ▶ Besoins à plus long terme.
 - ▶ Propositions industriel \Leftrightarrow (chercheur || équipe).
 - ▶ Gestion en projets.
 - ▶ Financement direct.
- ▶ Formation dans les entreprises.
- ▶ Formation dans les centre de recherches.



Investissements Industriels

- ▶ Besoin précis / ponctuels de recherche.
 - ▶ Faisabilité, preuve de concept.
 - ▶ Marge de proposition / recherche fondamentale.
- ▶ **Besoins à plus long terme.**
 - ▶ Propositions industriel \Leftrightarrow (chercheur || équipe).
 - ▶ Gestion en projets.
 - ▶ Financement direct.
- ▶ Formation dans les entreprises.
- ▶ Formation dans les centre de recherches.



Investissements Industriels

- ▶ Besoin précis / ponctuels de recherche.
 - ▶ Faisabilité, preuve de concept.
 - ▶ Marge de proposition / recherche fondamentale.
- ▶ Besoins à plus long terme.
 - ▶ Propositions industriel \Leftrightarrow (chercheur || équipe).
 - ▶ Gestion en projets.
 - ▶ Financement direct.
- ▶ **Formation** dans les entreprises.
- ▶ Formation dans les centre de recherches.



Initiative(s)...

- ▶ **En guise de conclusion...**
- ▶ ETL + Bases De Données : toujours utile (quelque soit le domaine) : conception et développement permanent utile (base existante ou *de novo*).
- ▶ Choix d'implémentation, changements...
- ▶ *Le présent n'est pas certain, l'avenir l'est moins* : initiative de groupe ou personnelle (profiter des deux).
- ▶ Profitez de vos connaissances et de l'information dont vous disposez ! Mais ne vous limitez pas à cela...

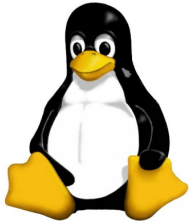


Questions

Questions ??



Le Libre et l'OpenSource



Le Libre et l'OpenSource

- ▶ Libre / Open Source : 4 libertés fondamentales.
 1. La liberté d'utilisation, pour tous les usages [limitations éthiques].
 2. La liberté d'étudier, pour adapter à ses besoins.
 3. La liberté de redistribuer des copies (ou assimilé : cultures, lignées).
 4. La liberté d'améliorer et de publier ces améliorations pour la communauté.
- ▶ Licences libres (GPL, Creative Commons...), des droits de réutilisation et re-publication.
- ▶ Open Source : liberté de moyens (le logiciel est un outil).
- ▶ **Intérêt** : cadre formel (et non plus informel) de modificationS au sein de la communauté.



Outils pratiques Unix / Linux / ...

Outils pratiques Unix / Linux / ...

- ▶ Commandes et outils classiques : *ed* / *sed* ; *grep* ; *find* ; ... (se faire un memento – aide-mémoire).
- ▶ Éditeurs de textes : *vi* ; *emacs* ; *nano* / *pico* / *jedit* ; *geany* ; *Notepad++* ; ...
- ▶ Émulateurs : *Wine* ; *qemu* ; *Virtual Box* ; ...
- ▶ Outils en ligne de commande : *mc* (*Midnight Commander*) ; *mutt* ; *screen* / *byobu* ; ...
- ▶ IDE / Environnements de développement : *Eclipse* (et ses nombreux plug-ins / modules) ; éditeur de texte favori ++ compilateur(s) ; ...
- ▶ Outils de version : *diff*, *patch*, *CVS*, *SVN* (*subversion*)...
- ▶ Traitement d'images, construction de diagrammes : *ImageMagick*, *GIMP*, *Dia*, *ArgoUML*, *Umbrello*...



Recherche d'emploi et code du travail

Recherche d'emploi et code du travail

- ▶ **Contrats Public** : Vacataires (recrutement direct, temporaire) et permanent (fonctionnaire).
- ▶ **Contrats Privé** : CDD / Intérim (court, primes, conditions), CDI (période d'essai).
- ▶ **Militaire** : contrats de 3 à 5 ans, conditions particulières.
- ▶ Recherche d'emploi : NO Bisounours !
- ▶ APEC, Pôle-Emploi (ex-ANPE) : dépôt de CV gratuit / recherche de CV gratuite : conséquences (budget, recrutement...).
- ▶ Monster : dépôt de CV gratuit / recherche de CV à tarifs variables : différents types de recrutement.
- ▶ Candidature spontanée : faible taux de réponse, il faut correspondre aux attentes en cours (cf. listes spécialisées).



Recherche d'emploi et code du travail

Recherche d'emploi et code du travail

- ▶ Contrats Public : Vacataires (recrutement direct, temporaire) et permanent (fonctionnaire).
- ▶ Contrats Privé : CDD / Intérim (court, primes, conditions), CDI (période d'essai).
- ▶ Militaire : contrats de 3 à 5 ans, conditions particulières.
- ▶ **Recherche d'emploi** : NO Bisounours !
- ▶ **APEC**, Pôle-Emploi (ex-ANPE) : dépôt de CV gratuit / recherche de CV gratuite : conséquences (budget, recrutement...).
- ▶ **Monster** : dépôt de CV gratuit / recherche de CV à tarifs variables : différents types de recrutement.
- ▶ **Candidature** spontanée : faible taux de réponse, il faut correspondre aux attentes en cours (cf. listes spécialisées).



"Social Patterns"

"Social Patterns"

- ▶ **Entretiens** : naturel mais neutre, pas obligé de répondre à toutes les questions, faire le tri et s'entraîner.
- ▶ **Au boulot** : bien faire et efficace, prendre des pauses (mentales, physiques...), préparer au stress si besoin.
- ▶ Savoir dire "NON" : mettre des limites, connaître les siennes, répartir vie pro. / vie privée...
- ▶ *Mettre de l'argent de côté*, même un minimum : toujours utile, préparer l'avenir, logement, changer de boulot...



...

Questions ??