



HAL
open science

Savoir lire une documentation : une compétence à développer chez les étudiants du BUT Informatique

Danielle Joulia

► To cite this version:

Danielle Joulia. Savoir lire une documentation : une compétence à développer chez les étudiants du BUT Informatique. Recherche et pratiques pédagogiques en langues. Cahiers de l'APLIUT, 2024, 43 (1), 19 p. hal-04788775

HAL Id: hal-04788775

<https://hal.science/hal-04788775v1>

Submitted on 18 Nov 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - NonCommercial - NoDerivatives 4.0 International License

Savoir lire une documentation : une compétence à développer chez les étudiants du BUT Informatique

*Knowing how to read documentation: a skill to be developed by students in
computing*

Danielle Joulia



Édition électronique

URL : <https://journals.openedition.org/apliut/11426>

ISSN : 2119-5242

Éditeur

APLIUT

Ce document vous est fourni par Université Fédérale Toulouse Midi-Pyrénées



Référence électronique

Danielle Joulia, « Savoir lire une documentation : une compétence à développer chez les étudiants du BUT Informatique », *Recherche et pratiques pédagogiques en langues* [En ligne], Vol. 43 N°1 | 2024, mis en ligne le 18 mai 2024, consulté le 18 novembre 2024. URL : <http://journals.openedition.org/apliut/11426>

Ce document a été généré automatiquement le 17 octobre 2024.



Le texte seul est utilisable sous licence CC BY-NC-ND 4.0. Les autres éléments (illustrations, fichiers annexes importés) sont « Tous droits réservés », sauf mention contraire.

Savoir lire une documentation : une compétence à développer chez les étudiants du BUT Informatique

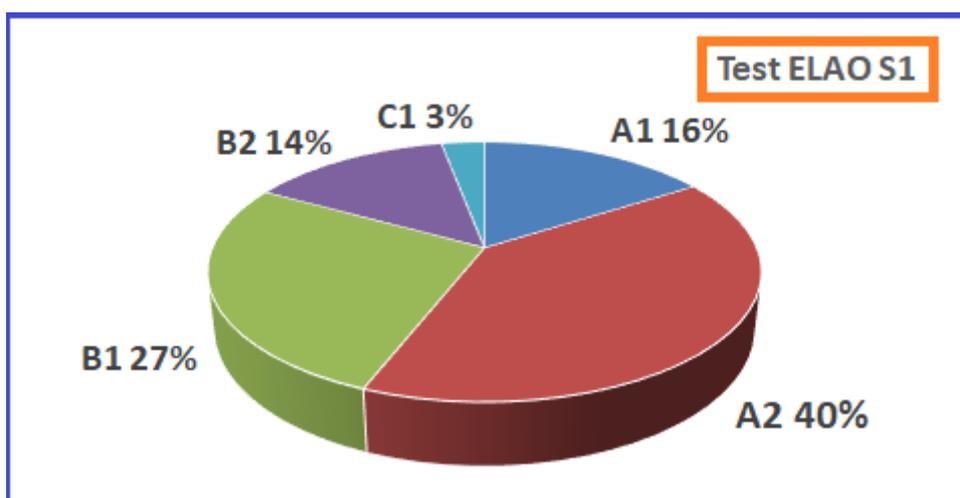
Knowing how to read documentation: a skill to be developed by students in computing

Danielle Joulia

Introduction

- 1 L'anglais est primordial dans les études d'informatique. Nos étudiants de Bachelor Universitaire de Technologie (BUT) Informatique sont dès le premier semestre, confrontés à l'anglais de spécialité lors des travaux pratiques du cœur de métier. Face à leur écran d'ordinateur qui affiche un contenu en anglais, ils doivent être capables de comprendre et de rechercher de manière efficace des informations pertinentes dans les documentations, notamment dans les manuels de langages de programmation. Pour une majorité d'étudiants, qui ont un niveau faible en anglais, 56 % de la promotion obtenant un score A1-A2 au test Elao¹ à leur entrée à l'Institut Universitaire de Technologie (IUT) (cf. Figure 1), leurs compétences linguistiques ne leur permettent pas de conduire une lecture efficiente de ces contenus spécialisés.

Figure 1. Résultats des étudiants du semestre 1 au test Elao à l'entrée en IUT



- 2 Les enseignants ne disposent pas de suffisamment d'heures pour relever le niveau linguistique de leurs étudiants, le volume horaire d'anglais, ayant subi une forte réduction lors du passage au BUT, de 190 heures en trois ans (150 heures pour le Diplôme Universitaire Technologique plus 40 heures pour la licence professionnelle) à 100 heures au niveau national dans le nouveau programme, auxquelles peuvent se rajouter 50 heures d'adaptation locale. L'objectif général de l'enseignement de la langue est de développer une compétence opératoire en anglais de communication et en anglais de spécialité pour préparer au mieux les apprenants à l'environnement linguistique professionnel qui les attend à leur sortie de l'IUT.
- 3 Le niveau des étudiants va de pair avec leur motivation pour l'anglais et une posture par rapport à la langue héritée du lycée. Un bon nombre d'entre eux font preuve de passivité face aux tâches à réaliser en anglais et sont résignés quant à leur capacité à progresser dans la matière (Joulia, 2011) alors que l'anglais est la *lingua franca* en informatique. Une des solutions pour les motiver est de donner du sens immédiatement à l'apprentissage de la langue et de montrer que l'anglais sert dans les études d'informatique, contrairement aux affirmations souvent entendues dans nos cours que l'avoir appris pendant sept années ne leur a servi à rien. Établir un lien avec le monde professionnel peut être un levier pour intéresser nos étudiants, pour les rendre actifs en cours tout en les préparant à la vie professionnelle.

1. Objectif didactique

- 4 Afin de proposer des tâches qui correspondent aux besoins des futurs informaticiens, nous avons mené une enquête auprès d'une centaine d'entreprises informatiques pour connaître leur utilisation de l'anglais. Vingt-neuf questionnaires nous ont été retournés. L'analyse des réponses, complétées par les programmeurs de l'entreprise, montrent que la compréhension de l'écrit est la première des compétences à travailler, elle représente 59 % des réponses (cf. Figure 2), et il s'agit en priorité de comprendre les messages d'erreur en anglais et les documentations de langages, de matériels et de logiciels (cf. Figure 3) (Joulia, 2014).

Figure 2. Résultats du questionnaire en termes de compétences requises

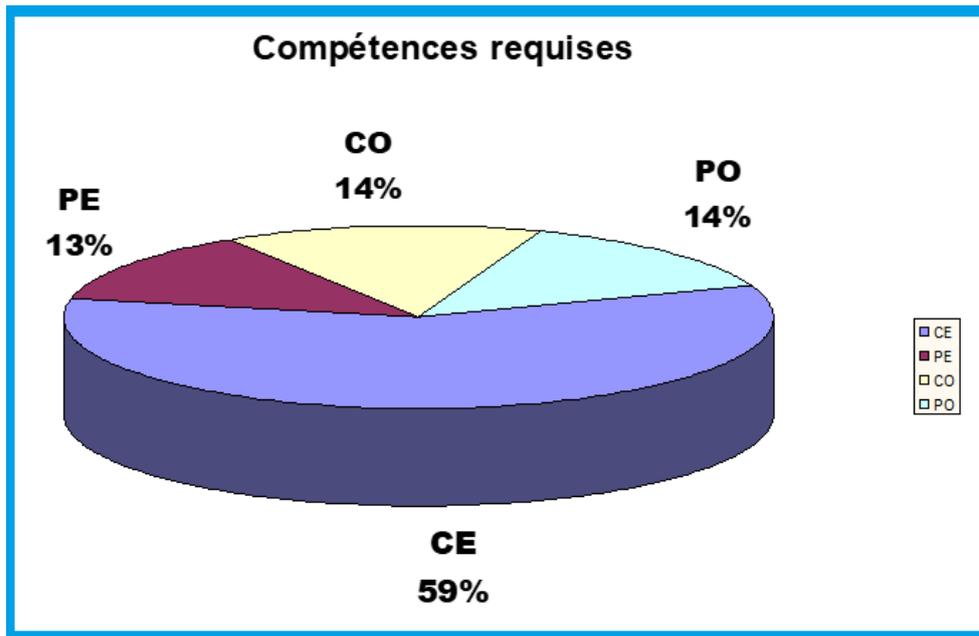
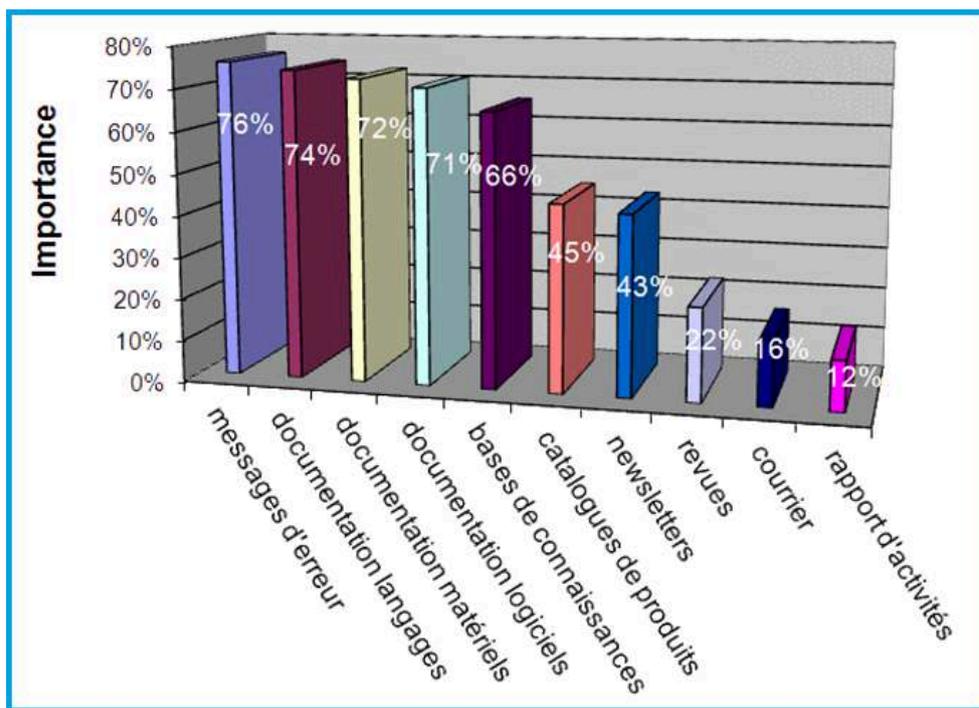


Figure 3. Résultats du questionnaire en termes de textes à comprendre



- 5 Notre objectif didactique fut donc de se focaliser sur cette compétence prioritaire, de développer le savoir lire de ces documentations en anglais, de rendre nos étudiants opérationnels en situation de lecture professionnelle afin qu'ils soient capables de mieux traiter l'information écrite et de rechercher de manière efficace les informations requises.
- 6 Afin de cerner les compétences sollicitées lors d'une lecture professionnelle, nous avons assisté aux travaux pratiques (TP) en informatique (TP de programmation et de

systèmes et réseaux) pour observer ce qu'il s'agissait de faire avec les documentations mises à disposition des étudiants dès le premier semestre. Les collègues informaticiens donnent aux étudiants une série d'exercices à réaliser sur ordinateur : établir la connexion avec le système d'exploitation Unix, accéder à un fichier et à un répertoire, changer de répertoire, changer les droits d'accès d'un fichier, supprimer un fichier, etc., opérations qui sont effectuées en saisissant des instructions de code informatique au clavier. Les étudiants, s'initiant à Unix, ont recours en ligne au manuel du langage de commandes dudit système pour savoir quelle commande utiliser pour coder ces opérations et comment l'utiliser.

- 7 Les objectifs de lecture varient et font appel à des stratégies de lecture différentes. Le lecteur peut balayer le texte (« scanning »), c'est-à-dire le lire sélectivement en diagonale pour localiser une information particulière, stratégie la moins coûteuse, posant peu de problèmes au lecteur non natif, qui reste à un niveau superficiel, n'ayant qu'à repérer dans le texte le mot qui correspond à celui qu'il recherche. Il peut aussi le survoler (« skimming »), en d'autres termes le parcourir pour identifier les idées principales ou les phrases-clés et en retirer une impression générale. Il peut également lire avec davantage d'attention (« receptive reading ») pour comprendre le sens du texte, ce qu'a voulu dire l'auteur sans le dire, ce qui demande un traitement approfondi pour saisir l'implicite², élaborer des inférences, tout n'étant pas dit dans le texte car l'auteur suppose que les informations manquantes sont connues du lecteur. Les différentes sections d'une page du manuel d'Unix pour une commande donnée appellent ces modes de lecture.
- 8 Nous avons assisté à des séances de TP d'informatique dans le but d'observer comment les étudiants abordaient les documentations en ligne et les avons interrogés sur leur compréhension des textes en anglais et sur les méthodes auxquelles ils avaient recours pour comprendre leur contenu. Ces observations nous ont permis de constater qu'ils adoptaient des stratégies qui n'étaient pas adaptées à leurs objectifs de lecture, en effectuant une lecture linéaire, souvent exhaustive du texte à l'écran, et peu efficace. Ils éprouvaient des difficultés au niveau du décodage linguistique (de la reconnaissance des mots), dû à un déficit lexical, et de la syntaxe, sur lequel ils se focalisaient. Ils n'utilisaient pas la structure du texte pour guider leur lecture, qu'ils ne connaissaient pas car propre au genre des documentations des langages de programmation qu'ils découvraient.
- 9 Pour plus de la moitié d'entre eux (soit 95 sur une cohorte de 168), notamment les étudiants faibles en anglais, qui ont souvent une appréhension face à un texte en anglais, un manque de confiance en soi et un faible sentiment d'efficacité personnelle (Bandura, 2003), ils renonçaient en choisissant la solution de facilité : soit ils se passaient de la documentation et essayaient de coder par essai et erreur, soit ils faisaient appel aux pairs ou à l'enseignant pour avoir les informations, soit ils consultaient la documentation en version française quand ils avaient compris qu'elle existait en ligne et comment y accéder. La difficulté n'était pas seulement linguistique (lacunes en anglais) mais aussi conceptuelle, en abordant de nouveaux concepts informatiques, aidés par une documentation, qui ne s'adressait pas à des débutants. Le cumul des difficultés rendait la tâche des apprenants plus complexe. Savoir lire ces documentations impliquait de savoir mobiliser les bonnes stratégies de lecture, les travailler pour les améliorer chez nos étudiants devint notre objectif didactique.

2. Lire et comprendre : fondements théoriques

- 10 La lecture-compréhension, en langue maternelle et en langue étrangère, a fait l'objet de nombreuses recherches (van Dijk et Kintsch, 1983 ; Kintsch, 1998 ; Carrell, 1990 ; Taillefer, 1996 ; Fayol et Gaonac'h, 2003 ; Gaonac'h, 2003 ; Huntley et Peñate, 2003 ; Nassaji, 2003), nous renvoyons le lecteur à notre thèse (Joulia, 2011) pour une étude exhaustive de la question. Dans le cadre de cet article, nous mettrons l'accent sur les causes qui peuvent expliquer les difficultés de nos étudiants lors de la lecture des documentations informatiques. La lecture-compréhension en langue étrangère représente une situation coûteuse sur le plan cognitif, notamment pour des apprenants qui ne maîtrisent pas suffisamment la langue et qui se retrouvent en surcharge mentale face à un texte en anglais. Lire et comprendre font appel à deux types de processus : appelés de bas niveau et de haut niveau. Les processus de bas niveau, de décodage linguistique, ne sont pas automatisés chez les étudiants faibles en anglais et consomment beaucoup de ressources mentales, contrairement à ce qui se passe en langue maternelle où le décodage est automatique. Les processus de haut niveau, qui impliquent l'utilisation du contexte (le repérage et la sélection des informations importantes), la convocation des connaissances discursives, l'anticipation, la prédiction, la formulation d'hypothèses, la production d'inférences, auxquels se rajoutent les processus métacognitifs de planification, de régulation et d'évaluation de sa compréhension (Dhieb-Henia, 2003 ; Goigoux, 2003 ; Rémond, 2003 ; Hamon et Renard, 2006 ; Cohen, 2009) ne sont pas mobilisés par les étudiants qui n'ont pas automatisé les processus de bas niveau : trop de ressources cognitives sont allouées aux traitements linguistiques de telle façon qu'ils ne disposent plus de ressources suffisantes pour faire appel à ces processus de haut niveau dont ils font normalement usage lors de la lecture en langue maternelle³. En surcharge cognitive, ils ne peuvent mener une lecture efficiente. Nos ressources mentales étant limitées, on ne peut se focaliser sur plusieurs tâches à la fois, il est nécessaire d'automatiser une partie des traitements, qui deviennent alors peu coûteux et laissent suffisamment de ressources pour se concentrer sur d'autres processus⁴.
- 11 Améliorer la compréhension de nos étudiants suppose donc, d'une part de faciliter l'automatisation du décodage linguistique, en élargissant leur champ lexical, à la fois, général et de spécialité, et en consolidant les structures syntaxiques de l'anglais général récurrentes en anglais informatique, et d'autre part de favoriser le transfert de la langue maternelle à la langue étrangère des processus de haut niveau. Ces derniers s'appuient sur des connaissances discursives et référentielles. Les premières seront travaillées en cours d'anglais, pour les secondes, nos collègues informaticiens assureront l'enseignement des notions informatiques utiles à la compréhension des documentations. Seule une expertise linguistique, discursive et référentielle permet d'aboutir à une compréhension approfondie, en accord avec les trois types de compétences définis par Moirand pour qualifier la compétence de lecture (1979, p. 22) :
- une compétence linguistique : connaissance des « modèles syntactico-sémantiques de la langue » ;
 - une compétence discursive : « connaissance des différents types d'écrits (leur organisation rhétorique) et de leurs dimensions pragmatiques » ;
 - une compétence référentielle : « connaissance des références extra-linguistiques des textes », en d'autres termes une familiarité avec les concepts abordés dans le texte.

- 12 Ces trois compétences seront développées en prenant comme support didactique le manuel du langage de commandes d'Unix.

3. Approche didactique

- 13 Nous avons opté pour une approche didactique fondée sur le genre, en choisissant des textes de spécialité, extraits du manuel d'Unix. Le niveau de spécialisation y est élevé, ce manuel étant rédigé par des informaticiens experts à destination de leurs pairs, appartenant à une communauté de discours (Swales, 1990), il ne s'agit pas d'un tutoriel pour débutants. Le manque de connaissances sur les genres les plus utilisés dans un domaine de spécialité constituent l'un des points faibles des étudiants qui abordent la lecture en langue étrangère (Magno E Silva, 2002), aussi est-il important de les y exposer afin qu'ils développent une aisance par rapport à ces textes et qu'ils acquièrent ce que Parkinson (2000, p. 371) qualifie de « *scientific literacy* » un ensemble de stratégies, attitudes, conceptions qu'appliquent les experts à la communication scientifique. Ils permettent de traiter des aspects linguistiques, discursifs, référentiels d'un genre, et ainsi d'en découvrir les conventions à ces différents niveaux, à savoir un ensemble de régularités lexicales, syntaxiques, rhétoriques, structurelles et thématiques. Les connaître favorise des stratégies de haut niveau qui améliorent la compréhension (Lemarié et al., 2012) et que des lecteurs experts, membres d'une communauté de discours, mobilisent habituellement face aux textes professionnels. Le genre de ces documentations sera à la fois objet d'apprentissage et moyen de mettre en œuvre des processus de haut niveau, lors d'une mise en situation qui requiert de lire et comprendre ces documentations.
- 14 Les textes sélectionnés pour la séquence pédagogique sont de type informatif à visée explicative, ils obéissent à une organisation textuelle spécifique avec différentes sections qui se suivent dans un ordre canonique. L'organisation rhétorique y est logique, du général vers le particulier. Ils présentent des structures syntaxiques récurrentes qui sont des phraséologies propres au domaine de spécialité. Cette restriction syntaxique va de pair avec une restriction lexicale (qui se traduit par des répétitions fréquentes de mots-clés) et sémantique (en raison des thèmes abordés limités à la programmation et fortement récurrents), dénotant une économie linguistique, qui facilite le traitement de l'information écrite. Ce sont des textes représentatifs du genre des documentations des langages informatiques. Ils vont servir à travailler sur les stratégies de lecture globale, sélective ou détaillée, sur des processus de haut niveau tels que la formulation d'hypothèses et la vérification d'hypothèses, l'anticipation, la prédiction, la production d'inférences, l'utilisation du contexte (le repérage de mots-clés) ainsi que sur les processus de bas niveau, de décodage linguistique, du lexique et de la syntaxe de l'anglais informatique de la programmation.

4. Séquence pédagogique

- 15 La séquence pédagogique est constituée de trois textes authentiques. Le premier, non spécialisé, est une notice d'utilisation d'un graveur de CD (Iomega, 2000), destinée à l'utilisateur lambda sans compétence en informatique. Nous avons choisi la version en néerlandais afin de contrer la tendance naturelle des étudiants pour la lecture mot à

mot. Les deux autres textes, en anglais de spécialité, sont extraits du manuel du langage de commandes du système d'exploitation Unix, qui s'adresse à des informaticiens confirmés. Les étudiants commencent par une approche globale du premier texte pour aller progressivement vers une lecture détaillée avec le troisième texte. La séquence pédagogique se répartit sur deux séances d'une heure trente. Il est préférable qu'elle ait lieu avant que les étudiants commencent leur TP de systèmes et réseaux et aient accès à cette documentation, soit en septembre, durant la troisième ou quatrième semaine de cours.

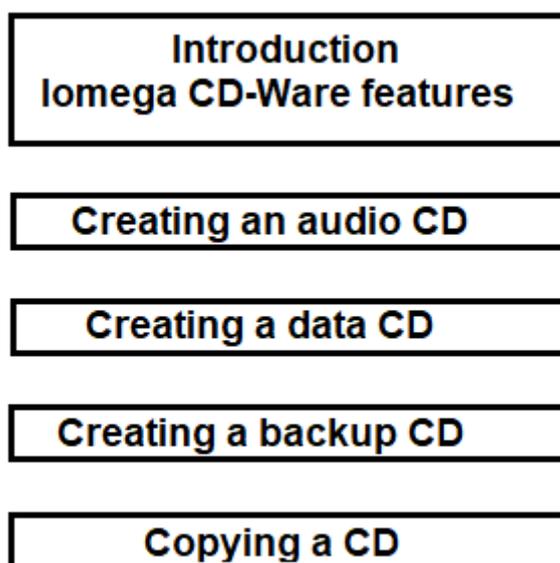
- 16 Elle s'inscrit dans le bloc de compétence 3 : administrer des systèmes informatiques communicants complexes du BUT Informatique, incluant les compétences suivantes : installer, configurer, mettre à disposition, maintenir en conditions opérationnelles des infrastructures, des services et des réseaux et optimiser le système informatique d'une organisation. Utiliser les fonctionnalités de base d'un système multitâches/multiutilisateurs ainsi qu'installer et configurer un système d'exploitation constituent les apprentissages critiques à développer (changements majeurs donnant lieu à une réorganisation cognitive chez l'apprenant). La séquence pédagogique se situe en parallèle avec la ressource 1.04 du semestre 1, d'introduction aux systèmes d'exploitation et à leur fonctionnement, dont l'objectif est de comprendre le rôle, les composants et le fonctionnement d'un système d'exploitation. Le savoir de référence ciblé plus particulièrement est la gestion des fichiers (types et droits). La situation d'évaluation finale qui s'intègre dans la Situation d'Apprentissage et d'Évaluation (SAÉ) 1.03, consiste à installer sur un ordinateur plusieurs systèmes d'exploitation, dont Unix/Linux, en d'autres termes à configurer une machine multi-systèmes. Une notice d'installation et d'utilisation devra être fournie par l'étudiant⁵.

4.1. Premier texte : CDWARE

- 17 Bien que le néerlandais soit une langue inconnue pour les étudiants, un certain nombre de mots transparents ainsi que sa structure permettent d'en deviner le thème principal. La tâche consiste à trouver le genre, le type de texte, le thème et la cible (lectorat) de ce texte (cf. Annexe 1), à le décomposer en parties et en donnant un titre en anglais à chaque partie trouvée. Il est utile de rappeler ce qu'est un genre⁶ et un type de texte⁷, notions vues au collège et oubliées.
- 18 Nous leur avons conseillé d'approcher ce texte comme ils le feraient d'une image, de prêter attention à sa forme, à sa structure, en le survolant. L'organisation en est aisément repérable, les parties sont facilement identifiables grâce à la mise en forme matérielle (numérotation, titres en gras, sauts de ligne). Portant sur un domaine qu'ils connaissent, ils peuvent rapidement reconstruire l'essentiel du contenu en utilisant les mots transparents (« *köpieren, klik, dubbelklik, data, backup, maken, solutions* ») et les mots internationaux, relatifs à la technologie, qui s'exporte avec le mot la désignant (« *CD-R, CD-RW, discs, audio* »), pour la plupart en gras et qui sont aussi des mots-clés. La structure textuelle leur permet de reconnaître le format d'une notice technique et de donner le type de texte (explicatif et injonctif) et le genre (manuel d'utilisation) auquel il appartient. Les étudiants sont familiers de ce type de documentation, destinée aux utilisateurs d'appareils ou de logiciels, qu'ils ont d'ailleurs tendance à ne pas lire sauf en cas d'incapacité à faire sans. Une fois la correction effectuée, le schéma de l'organisation textuelle (cf. Figure 4) est projeté avec une question subsidiaire : était-il

possible de répondre aux questions en trouvant un passage-clé qui résume le texte ? Quelques étudiants trouvent le passage, figurant dans l'introduction, comme cela est souvent le cas dans les documentations, ce qui permet en le repérant de savoir rapidement si le texte considéré intéresse le lecteur ou pas, sans perdre le temps de le parcourir ou d'en lire l'intégralité.

Figure 4. Schéma de l'organisation textuelle du texte CDWARE



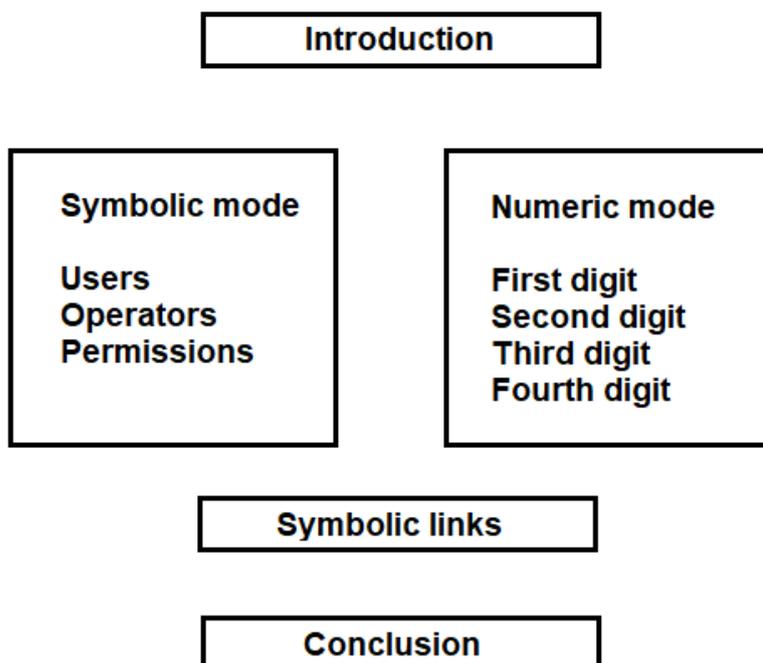
- 19 L'objectif de cette première phase était de leur montrer qu'il est possible de tirer du sens d'un texte, dont on ne maîtrise pas la langue, sans tout lire, en ayant recours à une approche textuelle différente, globale, de survol du texte et de prise d'indices. La preuve est faite qu'en produisant des inférences grâce à leurs connaissances antérieures sur le sujet et sur le format des notices techniques, en acceptant un certain degré d'incertitude et en ne visant pas une lecture exhaustive, tout dépendant de l'objectif de lecture fixé, ils parviennent à répondre aux questions. Il existe initialement une certaine résistance⁸ à approcher globalement le texte chez les étudiants, qui disent ne pas pouvoir faire l'exercice, ne parlant pas le néerlandais, ou se mettant à le lire de la première ligne à la dernière en tentant de comprendre toutes les phrases, comme nous avons pu le constater lors de nos observations en T.P. d'informatique, et ainsi en ne parvenant pas à réaliser la tâche dans le temps imparti (une dizaine de minutes). Toutefois, cela ne concerne qu'une poignée d'étudiants qui n'ont pas suivi les consignes de l'enseignante et ont voulu *lire* tout le texte avant de commencer à répondre à la première question. La majorité réalise la tâche sans trop de difficulté, trouvant le genre (**notice*), le type (**explicative, *instructive*), le thème (**CD-Rom*), la cible (*CD users*) et le nombre de parties (**6* ou *5*) avec leur titre (**copy CD*). Leurs erreurs portent surtout sur la forme et c'est l'occasion d'introduire du vocabulaire : *user guide/manual, explanatory, injunctive, CD burner/recorder* et de mettre l'accent sur la syntaxe utilisée pour intituler une procédure : *how to copy a CD* ou *copying a CD*, l'emploi erroné de la seule base verbale étant récurrent chez nos étudiants.

4.2. Deuxième texte : COPY

- 20 L'activité se poursuit avec un deuxième texte, en anglais, extrait du manuel d'Unix, en adoptant la même démarche globale. Il s'agit de la commande « *copy* » qui copie des groupes de fichiers (cf. Annexe 2). Cette commande est l'occasion d'introduire le format des pages du manuel en signalant que les étudiants auront sous peu à le consulter en ligne lors de leurs TP d'informatique. Tout comme pour la notice en néerlandais, ils ne peuvent parvenir à une compréhension complète de la commande, ignorant le sens de mots qui appartiennent au vocabulaire semi-technique et technique tels que « *mode, flag, n-flag, r-flag, recursively, verbose* ».
- 21 De même que pour le premier texte, il faut leur laisser peu de temps (dix minutes) pour l'activité (trouver le genre, le type de texte, le thème, le lecteur cible, découper le texte en parties et nommer en anglais les parties) afin d'éviter qu'ils se mettent à tout lire de façon linéaire. Il arrive toutefois que des étudiants n'aient quasiment rien répondu quand la correction commence parce qu'ils n'ont pas eu le temps de lire tout le texte avant de répondre aux questions. Il n'est alors pas inutile de leur rappeler qu'ils ont utilisé une mauvaise stratégie de lecture, peu efficace et inadaptée à l'exercice.
- 22 Lors de la correction, nous interrogeons les étudiants et projetons le texte avec les titres des sections, que nous avons initialement effacés du texte original. Il est important de faire expliciter à quoi sert chaque section, de demander lesquelles sont les plus consultées/lues pendant les TP d'informatique (« *name, synopsis, description, options* » en fonction de l'objectif de lecture).
- 23 Chaque unité textuelle assure une fonction communicative particulière. La section « *Name* » donne le nom de la commande et indique par une courte phrase-clé de résumé à quoi sert la commande, en fournissant ainsi un aperçu rapide de sa fonction. La commande est ensuite expliquée plus longuement dans la partie « *Description* ». La section « *Synopsis* » montre le modèle de la syntaxe : comment on doit écrire la commande (en code informatique). Vient ensuite la section des paramètres (nommés « *Options* » dans le manuel d'Unix), présentés succinctement, elle décrit l'action effectuée par les paramètres (ces derniers modifient la commande, la façon dont elle va être exécutée). Une section de « *Notes* » apporte une information qui n'est pas essentielle mais qui peut être utile. Enfin la dernière ligne correspond à une commande similaire qu'il peut être intéressant de consulter quand on copie des fichiers (la commande « *cp* »), d'où le titre de la section : « *See also* ».
- 24 Nous signalons que ce format en différentes sections est récurrent dans les pages du manuel d'Unix et se retrouve pour toutes les commandes du système d'exploitation, ainsi que peu ou prou dans les autres documentations des langages informatiques, avec les mêmes intitulés ou pas, certains étant bien spécifiques à Unix (tels que « *synopsis* » et « *options* »). Nous précisons également qu'un quart des mots de ce type de documentations sont des mots transparents (Joulia, 2011), facilitant le décodage linguistique (source chez les étudiants de surcharge cognitive qui est par là même allégée) et qu'il est donc plus facile de lire une documentation technique qu'un texte d'anglais général. Notre but est de rassurer les étudiants qui ont peu confiance en eux en anglais.

4.3. Troisième texte : CHMOD

- 25 Un troisième texte, extrait du même manuel, est remis aux étudiants. Il porte sur la commande « *chmod* », qui concerne les droits d'accès aux fichiers. Cette commande permet de les modifier⁹, de coder quel(s) utilisateur(s) dispose(nt) de quel(s) droit(s) sur un fichier donné. Elle est étudiée en TP de systèmes et réseaux au premier semestre et est source de difficultés, à la fois sur le plan conceptuel et linguistique. Les étudiants abordent ce texte en ayant déjà pris connaissance du format textuel avec la commande « *copy* », aussi la première étape qui consiste à trouver le genre, type, thème et lecteur cible est-elle réalisée sans poser de problème. Progressivement ils sont amenés à prêter attention à des points de détail, à conduire une lecture détaillée d'une partie du texte (la section « *Description* »), essentielle pour comprendre la commande « *chmod* ». Cette lecture requiert d'utiliser des stratégies de haut niveau, à savoir être capable de repérer et de sélectionner des informations clés d'un passage textuel pour en donner un titre. Nous projetons le texte avec les neuf parties de la description numérotées (chaque numéro doit être remplacé par un titre, cf. Annexe 3) et nous soulignons que le titre peut être un mot-clé présent dans le paragraphe considéré. Il s'agit donc de repérer quelle est l'idée essentielle d'un passage (provenant d'une phrase-clé) et de trouver le mot qui tiendra lieu de titre. Dans un texte bien rédigé, la phrase-clé se situe en général en début de paragraphe, ce qui est le cas dans ce texte. Elle introduit l'idée principale qui va être développée dans le paragraphe ou résume un point important et comporte le mot-clé. Les phrases-clés étant difficilement repérées par les apprenants (Taillefer, 2004), nous leur précisons qu'elles se trouvent au début des paragraphes numérotés et sont susceptibles de fournir les titres recherchés.
- 26 La plupart des titres à trouver sont des mots-clés de la description qui apparaissent dans les paragraphes. Soit les étudiants sélectionnent sans aucune aide le mot-clé qui va servir de titre, soit ils ont besoin d'être guidés pour le repérer, en leur demandant de rechercher quel est le mot qui est répété souvent et/ou qui est utilisé pour la première fois dans le texte au début d'un paragraphe donné. La correction s'effectue avec la projection du plan de la section « *Description* » (cf. Annexe 4), qui permet de visualiser comment les idées sont reliées entre elles de façon cohérente. Être capable de se représenter (hiérarchiquement) les idées contenues dans un texte ainsi que de saisir les relations qu'elles entretiennent est une stratégie de lecture primordiale à acquérir chez les étudiants. Le plan explicite l'ordonnancement hiérarchique de la description en détaillant les deux types de codage (représentation) de la commande. Les étudiants interrogés lors de la correction doivent donner, outre le titre du paragraphe, les différents éléments énumérés dans le paragraphe (par exemple pour « *operators* » mentionner les trois opérateurs : +, -, =).
- 27 Le plan est complété par la projection du schéma (cf. Figure 5) qui met en évidence l'organisation textuelle, le découpage du texte en cinq pavés, identifiés par leur contenu (introduction, représentation symbolique / numérique, liens symboliques, conclusion). Les deux représentations (symbolique et numérique) de « *chmod* » sont placées côte à côte, afin d'indiquer que l'une ou l'autre est utilisée pour coder les droits d'accès, tous les étudiants ne connaissant pas le sens de « *either* » (à la première ligne de la description).

Figure 5. Schéma de la section « Description » du texte « *chmod* »

- 28 Enfin, nous projetons le graphe (cf. Annexe 5) qui détaille les informations jusqu'au dernier niveau de profondeur, pour la représentation symbolique et numérique. Cette dernière n'est pas très bien expliquée (pour un débutant en informatique) et nécessite de faire de grandes inférences pour saisir comment l'utiliser, une partie des étudiants ne l'ayant toujours pas comprise, même après avoir suivi les TP d'informatique sur « *chmod* ». Le graphe est apprécié des étudiants qui visualisent ainsi les deux représentations jusqu'au code utilisé (« *ugo*, + =, *rwXstugo* » et « 4, 2, 1 »). Le graphe montre un exemple d'organisation rhétorique logique, qui part du général pour aller vers le particulier, en décrivant de plus en plus la commande, en annonçant d'abord qu'il y a deux façons pour coder cette commande et en expliquant ensuite comment chacune fonctionne. Notre objectif est de faire prendre conscience aux étudiants de l'existence d'une organisation textuelle et de les aider à la repérer quand ils abordent un texte, ayant ainsi recours à une stratégie de haut niveau qui facilitera leur compréhension.
- 29 L'activité sur ce texte se poursuit avec un exercice d'application¹⁰, similaire à ce qui est donné en TP d'informatique, où les étudiants consultent cette documentation pour coder. Il permet de vérifier s'ils ont compris et s'ils savent utiliser la commande, notamment avec la représentation numérique. Pour les aider, nous leur conseillons de s'inspirer des exemples à la fin du texte. En général, la plupart parviennent à coder avec les lettres (représentation symbolique) mais l'utilisation des chiffres pose problème à un bon nombre d'entre eux et leur code est faux. La correction de cet exercice participe à la compréhension de la représentation numérique. Nous réutilisons le graphe en entourant au tableau les permissions mentionnées dans l'énoncé (« *read*, *write*, *execute* ») et en y faisant correspondre les valeurs affectées à ces permissions (4, 2, 1). Nous terminons l'activité en leur signalant que la représentation numérique

n'étant pas bien explicitée, il est difficile de la comprendre sans élaborer des inférences, qu'ils ne peuvent produire car ils sont au début de leur apprentissage informatique. Cela est l'occasion de souligner que comprendre un texte implique deux composantes fondamentales : d'une part le lecteur, avec ses connaissances (et les opérations cognitives qu'il met en œuvre lors de la lecture) et d'autre part le texte, dont la forme et le contenu peuvent varier. Bien qu'ils aient compris tous les mots du paragraphe sur la représentation numérique, cette compréhension (superficielle) ne leur permet pas d'en saisir le fonctionnement. Nous insistons sur l'implicite dans les textes, pas toujours pris en compte par les étudiants, qui à l'entrée à l'IUT s'estiment bons en compréhension de l'écrit, s'évaluant à un niveau B2, voire C1, dans le tableau du Cadre européen commun de référence pour les langues (CECRL).

- 30 Nous concluons la séquence pédagogique par un travail lexical en leur demandant de traduire en français cinq mots du texte¹¹, en contexte, ce qui permet de faire un rappel sur les mots de liaison, notamment « *since* », systématiquement traduit par « depuis » (dont l'incohérence de la phrase ainsi produite en français¹² n'interpelle pas les étudiants), sur les faux-amis (« *assumed, currently* »), bien qu'il y ait beaucoup de mots transparents en anglais de spécialité, et les néologismes de sens (« *to set* » : allouer, définir, affecter, paramétrer) fréquents en anglais informatique. Nous faisons remarquer la répétition de mots-clés plutôt que l'utilisation d'anaphoriques pour ne pas générer d'ambiguïté, la répétition de phrases ou de structures syntaxiques¹³, qui sont autant de phraséologies propres au domaine de spécialité, et que ces récurrences facilitent la lecture (le décodage linguistique). Il est conseillé aux étudiants de se constituer un glossaire anglais-français, notamment pour ceux qui ont des lacunes lexicales, mais peu le font. Lorsque le dictionnaire de poche *Computers Basic Facts*¹⁴ était encore édité, chaque étudiant en recevait un en début d'année et devait l'avoir systématiquement en cours pour les activités de classe. Lorsqu'il a cessé d'être édité, nous l'avons remplacé par un dictionnaire en ligne, traduit en français, mis à disposition sur Moodle, accompagné de quiz (appariement de termes et de définitions) pour un travail en autonomie mais les statistiques d'utilisation montrent qu'il est peu consulté.
- 31 La mise en œuvre de cette séquence, qui correspond aux premiers cours de langues du semestre 1 rompt avec les leçons d'anglais du secondaire de par son contenu et son approche. D'abord étonnés, les étudiants se « prêtent au jeu », même s'ils n'arrivent pas tous à réaliser les exercices. C'est un premier succès de les avoir réconciliés avec la matière, qui apparaît autrement à leurs yeux, ayant l'impression d'avoir davantage fait de l'informatique que de l'anglais durant la séance. C'est aussi un prérequis pour les amener plus loin dans l'apprentissage de la langue de spécialité.

Conclusion

- 32 Nous avons présenté une approche didactique, centrée sur les besoins des apprenants, qui leur permet d'aborder ces documentations de façon plus efficace. Cette séquence pédagogique les aide à se rendre compte que leurs stratégies de lecture sont inadaptées à la tâche prescrite et vise à les modifier, à provoquer un « (r)éveil stratégique »¹⁵ (Taillefer, 2004, p. 80) afin d'améliorer leur saisie du texte. Pour qu'ils deviennent autonomes face à ces textes de spécialité, qu'ils sachent agir en situation de lecture-compréhension, il est nécessaire qu'ils changent de posture et cela ne peut

s'envisager que sur le long terme, l'étape de déconstruction des conduites de lecture antérieures étant longue et laborieuse et la mobilisation des stratégies de haut niveau (cognitives et métacognitives) en langue étrangère nécessitant d'avoir automatisé au préalable les processus de bas niveau. Aussi, est-il important, tout au long du cursus, en introduisant de nouveaux textes, de plus en plus complexes, de commencer par cette approche globale dans le but d'installer chez les étudiants des habitudes de lecture qu'ils pourront ensuite reproduire d'eux-mêmes face à tout document écrit, dans un contexte scolaire, extra-scolaire et professionnel. Il est aussi essentiel qu'ils soient exposés le plus possible à la langue étrangère et l'utilisent pour développer des automatismes. La réduction des heures d'anglais en IUT ne favorise guère la progression vers un niveau seuil mais les occasions d'apprentissage informel (Toffoli, Sockett et Kusyk, 2023), en particulier chez les étudiants d'informatique, rivaux à leur écran d'ordinateur, peut combler ce déficit d'heures et contribuer au développement des compétences en L2.

- 33 La lecture-compréhension de ces documentations correspond à un besoin à court, moyen et long terme, constituant une compétence nécessaire lors des TP d'informatique durant les trois années du BUT ainsi qu'au cours de leur vie professionnelle. Celle-ci sera jalonnée d'un recours à ces sources d'informations sans cesse renouvelées du fait de l'évolution rapide de l'informatique. Les prendre comme support didactique en cours d'anglais est motivant pour des étudiants qui ont un passé d'apprenants d'anglais en échec et fait sens par rapport à leurs TP et à leur futur métier. Répondant à un besoin perçu, ils ont tendance à s'engager plus facilement dans la construction de la compétence (Loisy, 2024). Ce nouveau besoin, professionnel, contribue à corriger leur représentation de l'anglais dans le secondaire et la posture défaitiste souvent adoptée par ces étudiants faibles en anglais. L'introduction des SAÉ dans le programme du BUT, avec participation et évaluation de la langue étrangère, renforce la motivation pour notre matière et favorise chez les étudiants une posture professionnelle correspondant aux attendus du diplôme.

BIBLIOGRAPHIE

- Adam, J.-M. (1999). *Linguistique textuelle, des genres de discours aux textes*. Nathan.
- Bandura, A. (2003). *Auto-efficacité : le sentiment d'efficacité personnelle*. De Boeck.
- Bronckart, J.-P. (1996). L'acquisition des discours, le point de vue de l'interactionnisme socio-discursif. Dans S. Moirand (dir.). *Le discours : enjeux et perspectives. Le Français dans le Monde, Recherches et Applications*, n° spécial, 55-64.
- Carrell, P. (1990). Culture et contexte dans la lecture en langue étrangère. Dans D. Gaonac'h (dir.). *Acquisition et utilisation d'une langue étrangère. Le Français dans le Monde, Recherches et Applications*, n° spécial, 16-29.
- Cohen, R. (2009). *Explorations in second language reading*. TESOL.

- Coirier, P., Gaonac'h, D. et Passerault, J.-M. (1996). *Psycholinguistique textuelle*. Colin/Masson.
- Dhieb-Henia, N. (2003). Evaluating the effectiveness of metacognitive strategy training for reading research articles in an ESP context. *ESP*, 22(4), 387-417. [https://doi.org/10.1016/S0889-4906\(03\)00017-6](https://doi.org/10.1016/S0889-4906(03)00017-6)
- Fayol, M. et Gaonac'h, D. (2003). La compréhension, une approche de psychologie cognitive. Dans D. Gaonac'h et M. Fayol (dir.), *Aider les élèves à comprendre, du texte au multimédia* (p. 5-72). Hachette.
- Gaonac'h, D. (2003). Comprendre en langue étrangère. Dans D. Gaonac'h et M. Fayol (dir.), *Aider les élèves à comprendre, du texte au multimédia* (p. 137-154). Hachette.
- Goigoux, R. (2003). Enseigner la compréhension : l'importance de l'autorégulation. Dans D. Gaonac'h et M. Fayol (dir.), *Aider les élèves à comprendre, du texte au multimédia* (p. 182-204). Hachette.
- Hamon, L. et Renard, C. (2006). Analyse de « Aider les élèves à comprendre, du texte au multimédia » de Daniel Gaonac'h et Michel Fayol (dir.). *ALSIC*, 9, 245-258. <https://doi.org/10.4000/alsic.264>
- Huntley, S. et Peñate, M. (2003). Specific reading at an advanced level: linguistic or strategic competence? *Ibérica*, 5, 87-100.
- Jouliá, D. (2011). *Faciliter la lecture-compréhension de documentations informatiques en anglais : de l'analyse à l'expérimentation didactique assistée par ordinateur*. [Thèse de doctorat non publiée. Université de Toulouse 2 Le Mirail].
- Jouliá, D. (2014). Une approche professionnalisante pour des étudiants de DUT Informatique. *Recherche et pratiques pédagogiques en langues de spécialité*, XXXIII(1), 91-108. <https://doi.org/10.4000/apliut.4190>
- Kintsch, W. (1998). *Comprehension: A paradigm for cognition*. CUP.
- Lemarié, J., Lorch Jr. R. F. et Péry-Woodley, M.-P. (2012). Understanding how headings influence text processing. *Discours*, 10. <https://doi.org/10.4000/discours.8600>
- Loisy, C. (2024). Compétences en langues dans les cursus universitaires : les valoriser et caractériser leurs situations développementales. *Recherche et pratiques pédagogiques en langues*, 43(1). <https://doi.org/10.4000/apliut.11448>
- Magno E Silva, W. (2002). *Étude linguistique d'un corpus de manuels d'informatique : application à la lecture en langue étrangère*. [Thèse de doctorat non publiée. Université de Toulouse 2 Le Mirail].
- Moirand, S. (1979). *Situations d'écrit* (1^{re} éd). CLÉ International.
- Nassaji, H. (2003). Higher-level and lower-level text processing skills in advanced ESL reading comprehension. *The Modern Language Journal*, 87(2), 261-276. <https://doi.org/10.1111/1540-4781.00189>
- Parkinson, J. (2000). Acquiring scientific literacy through content and genre: a theme-based language course for science students. *ESP*, 19, 369-387. [10.1016/S0889-4906\(99\)00012-5](https://doi.org/10.1016/S0889-4906(99)00012-5)
- Rémond, M. (2003). Enseigner la compréhension : les entraînements métacognitifs. Dans D. Gaonac'h et M. Fayol (dir.), *Aider les élèves à comprendre, du texte au multimédia* (p. 205-232). Hachette.
- Swales, J. (1990). *Genre analysis: English in academic and research settings* (1^{re} éd). CUP.

Taillefer, G. (2004). Lire en anglais à l'université : étude de cas d'étudiants en Maîtrise de Sciences économiques confrontés à l'article scientifique. *Les Cahiers de l'APLIUT*, XXIII (3), 76-90. <https://doi.org/10.4000/apliut.3274>

Taillefer, G. (1996). L2 reading ability: Further insight into the short-circuit hypothesis. *The Modern Language Journal*, 80 (4), 461-477. <https://doi.org/10.1111/j.1540-4781.1996.tb05466.x>

Toffoli, D., Sockett, G. et Kusyk, M. (2023). *Language learning and leisure: Informal language learning in the digital age*. De Gruyter Mouton.

van Dijk, T. et Kintsch, W. (1983). *Strategies of discourse comprehension*. Academic Press.

ANNEXES

Annexe 1 : notice d'utilisation d'un graveur de CD

Aan de slag met CD-Ware

Met de Iomega® CD-RW-drive kunt u CD-R-discs (Compact Disc-Recordable) en CD-RW-discs (Compact Disc-Rewritable) maken.

Met de CD-Ware-software kunt u CD-R-discs en CD-RW-discs formatteren. Bovendien kunt u er gegevens naar toe schrijven.

- **CD-R-discs** — U kunt bestanden van de CD-R-disc verwijderen. De vrijgekomen ruimte kan echter niet meer worden gebruikt. Het is dus niet mogelijk om bestanden op CD-R-discs te overschrijven. CD-R-discs zijn bij uitstek geschikt voor audio-opnamen.

- **CD-RW-discs** — Deze discs kunnen wel 1000 keer opnieuw worden geformatteerd. Bestanden op een CD-RW-disc kunnen worden overschreven. Deze discs zijn geschikt voor de opslag van gegevens en voor het maken van backups. De meeste computers, maar ook sommige cd-rom-stations zijn niet in staat om CD-RW-discs te lezen.

Gebruik de Iomega CD-RW-drive voor het volgende:

- audio-cd maken
- data-cd maken
- backup-cd maken
- cd kopiëren

Inhoud Solutions-cd

CD-Ware — Het softwarepakket CD-Ware bevat alle noodzakelijke systeemsoftware voor het gebruik van de Iomega CD-RW-drive. Het pakket bevat ook de software waarmee u cd's kunt maken. Dit zijn de belangrijkste programma's:

Toast (alleen voor Mac) — Hiermee maakt u audio-cd's die u thuis en in de auto kunt afspelen of data-cd's om gegevens op te slaan, bijvoorbeeld bestanden en mappen van de vaste schijf.

Adaptec Easy CD Creator (alleen pc) — Hiermee maakt u audio-cd's die u thuis en in de auto kunt afspelen of data-cd's om gegevens op te slaan, bijvoorbeeld bestanden en mappen van de vaste schijf.

Adaptec DirectCD (alleen pc) — Hiermee kopieert u bestanden rechtstreeks naar de Iomega CD-RW-drive. Dit gaat net zo als het kopiëren van en naar de vaste schijf. U kunt bestanden rechtstreeks op de CD-RW-discs opslaan vanuit programma's zoals Microsoft Word. Het is ook mogelijk om in Windows Verkenner bestanden naar de CD-RW-disc te slepen en hier neer te zetten.

Iomega Quik Sync 2-software (Mac en pc) — Telkens wanneer u een bestand opslaat op de vaste schijf of op een netwerkdisk, kan er door Quik Sync 2 automatisch een kopie van dat bestand worden op de CD-RW-discs.

1

Een audio-cd maken

1. Dubbelklik op het pictogram **Cd maken** dat zich op het bureaublad bevindt.
2. Klik op **Audio**.
3. Klik op **Audio-cd**.
4. Plaats de muziek-cd in de Iomega CD-RW-drive.
5. Sleep de audiotracks vanuit het bovenste venster naar het onderste venster in de indeling.
6. Als u tracks vanaf meerdere cd's kopieert, moet u eerst de eerste cd uit de drive verwijderen voordat u de volgende cd plaatst. Herhaal stap 5.
7. Klik op de rode knop **Cd maken** in de bovenste werkbalk.
8. Klik op **OK**.
9. Plaats een lege CD-R-disc in de Iomega CD-RW-drive wanneer hierom wordt gevraagd.
10. Volg de aanwijzingen op het scherm als u de cd's wilt kopiëren.
11. Klik op **OK** als de procedure is voltooid.

3

Een backup-cd maken

1. Dubbelklik op het pictogram **Cd maken** dat zich op het bureaublad bevindt.
2. Klik op **Data**.
3. Klik op **Direct CD**.
4. Klik op **Volgende**.
5. Selecteer de driveletter van de Iomega CD-RW.
6. Plaats een lege CD-R of een geformatteerde CD-RW-disc in de Iomega CD-RW-drive.
7. Wacht 10-20 seconden tot het systeem de disc herkent.
8. Klik op **Volgende**.
9. Klik op **Volgende**.
10. Typ een etiket voor de disc en klik op **Voltoelen**.
11. Klik op **OK** als de procedure is voltooid.

2

Een data-cd (gegevens-cd) maken

1. Dubbelklik op het pictogram **Cd maken** dat zich op het bureaublad bevindt.
2. Klik op **Data**.
3. Klik op **Data-cd**.
4. Zoek in het bovenste venster naar de bestanden die u wilt kopiëren.
5. Sleep de gewenste bestanden vanuit het bovenste naar het onderste venster in de indeling.
6. Klik op de rode knop **Cd maken** in de bovenste werkbalk.
7. Klik op **OK**.
8. Plaats een lege CD-R- of CD-RW-disc in de Iomega CD-RW-drive wanneer hierom wordt gevraagd.
9. Klik op **OK** als de procedure is voltooid.

4

Een cd kopiëren

1. Plaats de cd die u wilt kopiëren in de Iomega CD-RW-drive.
2. Dubbelklik op het pictogram **Cd maken** dat zich op het bureaublad bevindt.
3. Klik op **CD Copier**.
4. Klik op de lijst **Kopiëren vanaf**.
5. Selecteer de letter van de Iomega CD-RW-drive.
6. Klik op de lijst **Opnemen naar**.
7. Selecteer de letter van de Iomega CD-RW-drive.
8. Klik op de knop **Kopiëren** rechtsonder in het venster.
9. Volg de aanwijzingen op het scherm.

Annexe 2 : la commande « copy » extraite du manuel d'Unix.

NAME

copy - copy groups of files

SYNOPSIS

copy [option]... source ... dest

DESCRIPTION

The `copy` command copies the contents of directories to another directory. It is possible to copy whole file systems since directories are made when needed. If files, directories, or special files do not exist at the destination, then they are created with the same modes and flags as the source. In addition, the super-user may set the `user` and `group` ID. The owner and mode are not changed if the destination file exists. Note that there may be more than one source directory. If so, the effect is the same as if the `copy` command had been issued for each source directory with the same destination directory for each copy.

OPTIONS

All of the options must be given as separate arguments, and they may appear in any order even after the other arguments. The arguments are:

- a Asks the user before attempting a copy. If the response does not begin with a "y", then a copy is not done. This option also sets the `ad` option.
- l Uses links instead whenever they can be used. Otherwise a copy is done. Note that links are never done for special files or directories.
- n Requires the destination file to be new. If not, then the copy command does not change the destination file. The `-n` flag is meaningless for directories. For special files an `-n` flag is assumed (that is, the destination of a special file must not exist).
- o If set then every file copied has its owner and group set to those of source. If not set, then the file's owner is the user who invoked the program.
- m If set, then every file copied has its modification time and access time set to that of the source. If not set, then the modification time is set to the time of the copy.
- r If set, then every directory is recursively examined as it is encountered. If not set, then any directories that are found are ignored.
- sd Asks the user whether an `-r` flag applies when a directory is discovered. If the answer does not begin with a "y" then the directory is ignored.
- v If the verbose option is set, messages are printed that reveal what the program is doing.
- source This may be a file, directory or special file. It must exist. If it is not a directory, then the results of the command are the same as for the `cp` command.
- dest The destination must be either a file or directory that is different from the source. If `source` and `destination` are anything but directories, then `copy` acts just like a `cp` command. If both are directories, then `copy` copies each file into the destination directory according to the flags that have been set.

NOTES

Special device files can be copied. When they are copied, any data associated with the specified device is not copied.

SEE ALSO

`cp` - copy files

NB : les titres en gras des sections sont effacés dans la version distribuée aux étudiants.

Annexe 3 : la commande « chmod » extraite du manuel d'Unix.

NAME

chmod - change file access permissions

SYNOPSIS

chmod [OPTION]... MODE[MODE]... FILE...

chmod [OPTION]... OCTAL-MODE FILE...

DESCRIPTION

① `chmod` changes the permissions of each given file according to mode, which can be either a symbolic representation of changes to make, or an octal number representing the bit pattern for the new permissions.

② The format of a symbolic mode is '[ugoa...][[+-][rwxXstugo...]]...[.].'. Multiple symbolic operations can be given, separated by commas.

③ A combination of the letters 'ugoa' controls which users' access to the file will be changed: the user who owns it (u), other users in the file's group (g), other users not in the file's group (o), or all users (a). If none of these are given, the effect is as if 'a' were given, but bits that are set in the umask are not affected.

④ The operator '+' causes the permissions selected to be added to the existing permissions of each file; '-' causes them to be removed; and '=' causes them to be the only permissions that the file has.

⑤ The letters 'rwxXstugo' select the new permissions for the affected users: read (r), write (w), execute (or access for directories) (x), execute only if the file is a directory or already has execute permission for some user (X), set user or group ID on execution (s), sticky (t), the permissions that the user who owns the file currently has for it (u), the permissions that other users in the file's group have for it (g), and the permissions that other users not in the file's group have for it (o).

⑥ A numeric mode is from one to four octal digits (0-7), derived by adding up the bits with values 4, 2, and 1. Any omitted digits are assumed to be leading zeros. The first digit selects the set user ID (4) and set group ID (2) and sticky (1) attributes. The second digit selects permissions for the user who owns the file: read (4), write (2), and execute (1); the third selects permissions for other users in the file's group, with the same values; and the fourth for other users not in the file's group, with the same values.

⑦ `chmod` never changes the permissions of symbolic links; the `chmod` system call cannot change their permissions. This is not a problem since the permissions of symbolic links are never used. However, for each symbolic link listed on the command line, `chmod` changes the permissions of the pointed-to file. In contrast, `chmod` ignores symbolic links encountered during recursive directory traversals.

⑧ Only the owner of a file or directory (or the super user) may change the file's or directory's mode.

OPTIONS

Change the mode of each FILE to MODE.

-c, --changes

like verbose but report only when a change is made

-f, --silent, --quiet

suppress most error messages

-v, --verbose

output a diagnostic for every file processed

-R, --recursive

change files and directories recursively

Each MODE is one or more of the letters 'ugoa', one of the symbols '+-=' and one or more of the letters 'rwxXstugo'.

EXAMPLES

`chmod u=rwx,g=rX,o=r FILE`

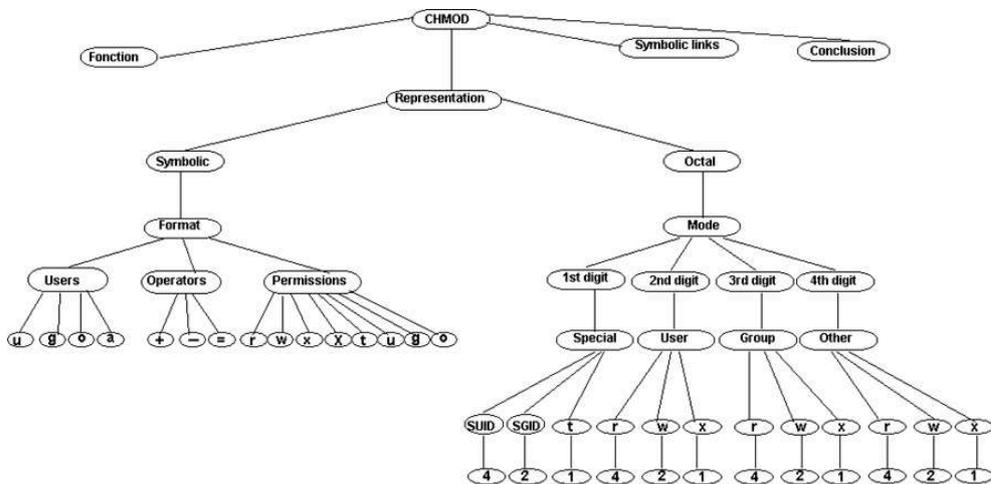
`chmod 751 FILE`

The two statements are equivalent.

The permissions of the file are `rwx-r-x-`.

Annexe 4 : plan de la section « Description » du texte « *chmod* »**CHMOD****Description**

- 1 Introduction**
chmod function
chmod representation
- 2 Symbolic mode**
- 3 Format**
- 4 Users**
- 5 Operators**
- 6 Permissions**
- 7 Numeric mode**
First digit
Second digit
Third digit
Fourth digit
- 8 Symbolic links**
- 9 Restriction**

Annexe 5 : graphe de la section « Description » du texte « *chmod* »**NOTES**

1. Le test Elao évalue les connaissances grammaticales, lexicales (vocabulaire actif et passif) ainsi que la compréhension de l'oral et attribue à l'étudiant un score entre A0 et C2 (<https://www.elao-test.com/>).

2. Par exemple, dans le texte CHMOD (cf. Annexe 3), à la fin du paragraphe 5, pour la phrase « *'=' causes them to be the only permissions that the file has* », il faut comprendre que l'opérateur '=' retire les droits d'accès par ailleurs donnés au fichier et n'octroie que les droits spécifiés après '='. La traduction française officielle du manuel est plus explicite : « l'opérateur '=' autorise

exclusivement l'accès indiqué ». Cet opérateur est souvent source d'erreurs en début d'apprentissage de la commande « chmod » chez les étudiants du semestre 1.

3. Leur compétence de lecture en langue maternelle n'a cependant pas été testée.

4. Par exemple, quand on apprend à conduire une voiture, nous nous focalisons d'abord sur ce qui se passe à l'intérieur de la voiture, sur quelle pédale appuyer, comment débrayer et passer une vitesse ; une fois ces opérations automatisées, on peut davantage se concentrer sur ce qui se passe à l'extérieur de la voiture, voire anticiper les prochaines actions, pour dépasser, éviter un obstacle, etc., tout en écoutant de la musique, activité qui ne requiert pas particulièrement de traitement coûteux (attentionnels) sur le plan cognitif.

5. D'après le programme national du BUT Informatique : <https://www.enseignementsup-recherche.gouv.fr/sites/default/files/annexe-2-licence-professionnelle-bachelor-universitaire-de-technologie-informatique-29016.pdf>

6. Défini comme des « formes communicatives historiquement construites par diverses formations sociales, en fonction de leurs intérêts et de leur objectifs propres » (Bronckart, 1996, p. 56), par exemple, le roman, l'article scientifique, le manuel technique, la recette de cuisine...

7. Défini comme une séquence élémentaire prototypique (type narratif, descriptif, explicatif, injonctif, argumentatif, dialogal) selon Adam (1999).

8. Plusieurs séances sont nécessaires pour lever cette résistance. La contrainte temporelle convaincant les plus « rebelles », quand ils apprennent qu'au partiel d'anglais ils n'auront que trente minutes pour répondre aux questions de compréhension du texte et faire les autres exercices requis.

9. On peut accorder ou supprimer des droits de lecture, d'écriture ou d'exécution à différents types d'utilisateur.

10. 1) « Write in symbolic and numeric modes the command that gives all users the permissions to read and execute a file and which adds the permission to write for the owner ».

2) « Write in numeric mode the command that gives all users the permission to read a file ».

11. « either (l.1), set (l.7), currently (l.12), assumed (l.16), since (l. 21) ».

12. Ceci n'est pas un problème **depuis que** [au lieu de « puisque »] les droits des liens symboliques ne sont **jamais** utilisés.

13. § 4 : « other users in the file's group (g), other users not in the file's group (o) ».

§ 5 : « causes the permissions selected to be added to the existing permissions of each file; '-' causes them to be removed; and '=' causes them to be the only permissions that the file has ».

§ 6 : « the permissions that the user who owns the file currently has for it (u), the permissions that other users in the file's group have for it (g), and the permissions that other users not in the file's group have for it (o) ».

14. Voir <https://www.amazon.com/Computers-Basic-Facts-Collins-Gem/dp/0004708423>.

15. Mise entre parenthèses par nous.

RÉSUMÉS

L'anglais est primordial dans les études d'informatique. Dès le premier semestre, les étudiants sont confrontés à l'anglais de spécialité lors des travaux pratiques du cœur de métier. Face à leur écran d'ordinateur qui affiche un contenu en anglais, ils doivent être capables de comprendre et de rechercher de manière efficace des informations pertinentes dans les documentations,

notamment dans les manuels de langage de programmation. Pour la plupart, qui ont un niveau en anglais de faux-débutant (A1-A2), leurs compétences linguistiques ne leur permettent pas de conduire une lecture efficace de ces contenus spécialisés. Notre objectif est de les aider à développer des stratégies de lecture : à partir d'extraits de manuels techniques, nous les guidons vers une approche globale de ces textes authentiques, et ce faisant, ils découvrent le genre textuel de ces documentations informatiques, leur organisation textuelle et rhétorique, leurs régularités syntaxiques et lexicales qui en facilitent la compréhension.

English plays an essential role in computer science studies. Right from the first semester, students are faced with English for specific purposes in their practical work. In front of a computer screen displaying content in English, they need to be able to understand and efficiently search for relevant information in documentation, particularly in programming language manuals. For most of them, who have a pre-intermediate level of English (A1-A2), their language skills do not enable them to read this specialised content efficiently. Our aim is to help them develop reading strategies to overcome this. Using extracts from technical manuals, we guide them towards a global approach to discover the genre of these authentic documents, the rhetorical organisation, the syntax and lexical regularities, all of which facilitate comprehension.

INDEX

Keywords : English for computer science, genre-based approach, global reading, reading comprehension, skill-based approach

Mots-clés : anglais de spécialité informatique, approche fondée sur le genre, approche globale des textes, approche par compétences, compréhension de l'écrit

AUTEUR

DANIELLE JOULIA

Danielle Joulia est maître de conférences en anglais de spécialité au département Informatique de l'IUT Paul Sabatier, Université Toulouse III. Ses axes de recherche portent sur la didactique de l'anglais de spécialité scientifique et technique, en lien avec ses enseignements (compréhension écrite de documentations techniques en anglais, analyse de corpus spécialisés, conception et utilisation de technologies éducatives ciblées sur les besoins des étudiants d'informatique). Elle s'intéresse également aux dispositifs d'internationalisation (EMILE), qu'elle aborde en termes de charge cognitive pour l'apprenant. Elle est membre du LAIRDIL (URU 7415).
danielle.joulia[at]iut-tlse3.fr