Chapitre: 02
Architecture et Fonctionnement d'un Ordinateur

CHAPITRE 2 : Architecture et Fonctionnement d'un Ordinateur

Objectifs du chapitre

À la fin de ce chapitre, l'étudiant doit être capable de :

- Identifier les principaux composants matériels d'un ordinateur et expliquer leur rôle.
- Distinguer les différents types de mémoires (RAM, ROM, cache, registres) et de stockage (HDD, SSD, cloud).
- Comprendre le rôle du système d'exploitation et des logiciels d'application.
- Expliquer le cycle de traitement d'une instruction (fetch-decode-execute).
- Relier l'architecture informatique aux usages concrets en SNV (analyse de données, imagerie, bio-informatique).
- Décrire le flux d'information depuis l'entrée jusqu'à la sortie dans un ordinateur.

2. Introduction

Dans le chapitre précédent, nous avons défini l'informatique et le système informatique. Nous avons vu qu'un ordinateur est composé de **matériel** (hardware) et de **logiciel** (software). Ce chapitre approfondit cette distinction en détaillant l'architecture interne d'un ordinateur et la manière dont il exécute les programmes.
Pourquoi un étudiant en SNV doit-il comprendre cela ? Parce que :

- Les performances d'un ordinateur influencent la vitesse d'analyse des données.
- Connaître les limites de la mémoire permet de gérer des jeux de données volumineux (séquences, images).
- Comprendre le rôle du système d'exploitation aide à installer et utiliser correctement les logiciels scientifiques.
- Savoir comment fonctionne un ordinateur permet de mieux diagnostiquer les problèmes et d'optimiser son travail.

Ce chapitre répond à des questions simples : De quoi est fait un ordinateur ? Comment fonctionne-t-il ? Comment le matériel et le logiciel collaborent-ils ? Et surtout, comment tout cela s'applique-t-il en SNV ?

2.1 Les composants matériels (Hardware)

2.1.1 L'unité centrale de traitement (CPU)

Le **CPU** (Central Processing Unit) est le cerveau de l'ordinateur. Il exécute les instructions des programmes. Il est composé de deux parties principales :

- **L'unité de commande (UC)** : elle coordonne les opérations et contrôle le flux d'instructions.
- **L'unité arithmétique et logique (UAL)** : elle effectue les calculs (addition, soustraction, comparaisons logiques).

Le CPU contient également des **registres** (mémoires ultra-rapides) et une **horloge** qui cadence les opérations. Plus la fréquence de l'horloge est élevée, plus le CPU peut traiter d'instructions par seconde.

Le tableau suivant récapitule les composants du CPU et leur rôle.

Tableau 1 : Composants du CPU et leurs fonctions

Composant	Rôle	Exemple en SNV
Unité de commande (UC)	Lit et décode les instructions, contrôle les autres composants.	Ordonne le chargement d'une séquence d'ADN depuis le disque.
Unité arithmétique et logique (UAL)	Effectue les opérations arithmétiques et logiques.	Calcule la moyenne de plusieurs mesures de température.
Registres	Mémoires ultra-rapides internes au CPU.	Stockent temporairement les valeurs en cours de calcul.
Horloge	Cadence les opérations (fréquence en GHz).	Un CPU à 3 GHz peut exécuter 3 milliards de cycles par seconde.

Explication du tableau 1

- L'**UC** lit les instructions du programme et les traduit en signaux électriques.
- L'**UAL** exécute les calculs demandés : addition, multiplication, comparaison.
- Les **registres** sont de minuscules mémoires intégrées au processeur. Ils sont extrêmement rapides mais de très petite taille.
- L'**horloge** donne le rythme. Plus la fréquence est élevée, plus le CPU peut traiter d'instructions par seconde.

Pour visualiser l'organisation interne du CPU, observons la figure suivante.

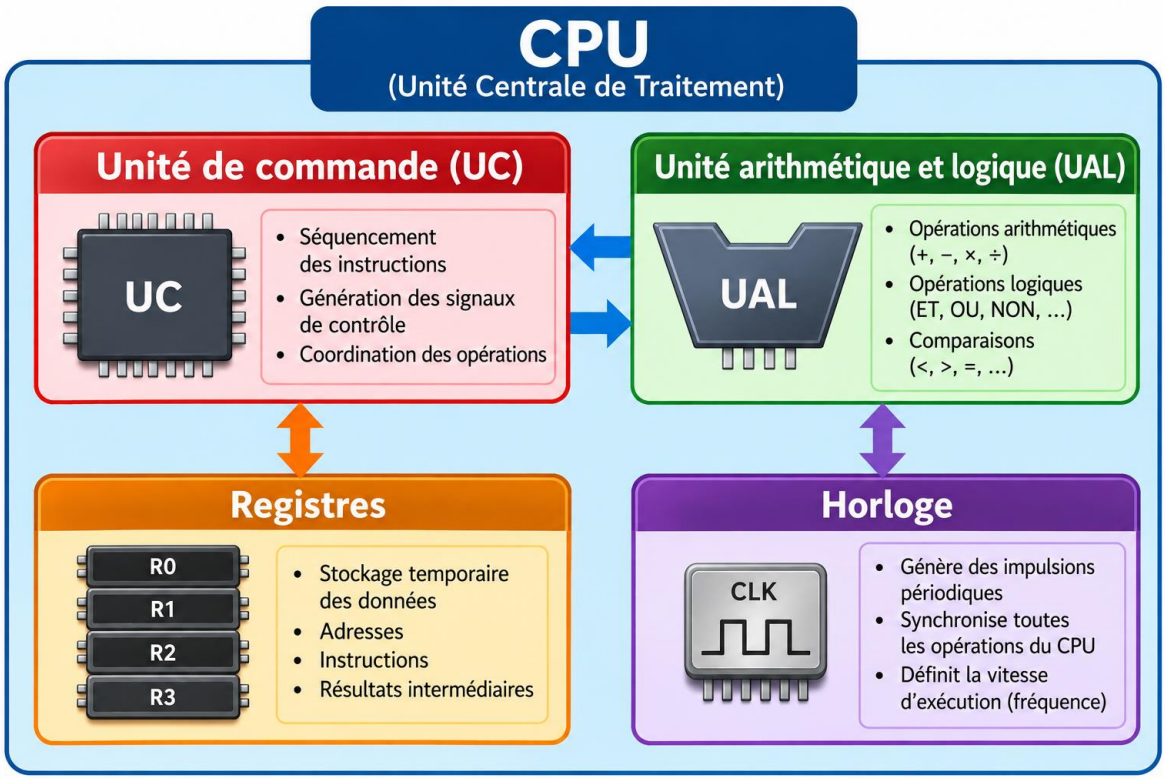

Figure 1 : Schéma simplifié du CPU

Explication de la figure 1

Le CPU communique en interne :

- L'**UC** envoie des signaux à l'**UAL** pour effectuer des calculs.
- Les **registres** servent de mémoire tampon pour les données et les instructions.
- L'**horloge** synchronise toutes les opérations.

Exemple concret en SNV :

Lorsque vous lancez un calcul de moyenne dans Excel, l'UC lit l'instruction, l'UAL effectue l'addition et la division, et le résultat est stocké temporairement dans un registre avant d'être affiché.

2.1.2 La mémoire

La mémoire est indispensable au fonctionnement de l'ordinateur. On distingue principalement :

- **La RAM** (Random Access Memory) : mémoire vive, temporaire, effacée à l'extinction.
- **La ROM** (Read Only Memory) : mémoire morte, permanente, contient les instructions de démarrage.
- **La mémoire cache** : mémoire très rapide, intégrée au CPU, qui accélère l'accès aux données fréquemment utilisées.
- **Les registres** : mémoires internes au CPU, encore plus rapides.

Le tableau suivant compare ces types de mémoires.

Tableau 2 : Comparaison des mémoires

Type	Volatilité	Vitesse	Taille typique	Rôle
RAM	Volatile (effacée à l'arrêt)	Rapide	4 à 32 Go	Stocke les programmes et données en cours d'utilisation.
ROM	Non volatile	Lente	Quelques Mo	Contient le BIOS, nécessaire au démarrage.
Cache	Volatile	Très rapide	Quelques Mo	Accélère l'accès aux données fréquemment utilisées.
Registres	Volatile	Ultra-rapide	Quelques octets	Stockage interne au CPU.

Explication du tableau 2

- La **RAM** est comme un bureau de travail : plus il est grand, plus vous pouvez étaler de documents en même temps. Si vous ouvrez trop de logiciels, l'ordinateur ralentit car la RAM est saturée.
- La **ROM** contient des informations fixes, comme le programme qui démarre l'ordinateur.
- Le **cache** est une mémoire intermédiaire entre le CPU et la RAM. Il évite au CPU d'attendre après la RAM.
- Les **registres** sont les plus rapides mais minuscules.

Pour comprendre la hiérarchie des mémoires, observons la figure suivante.

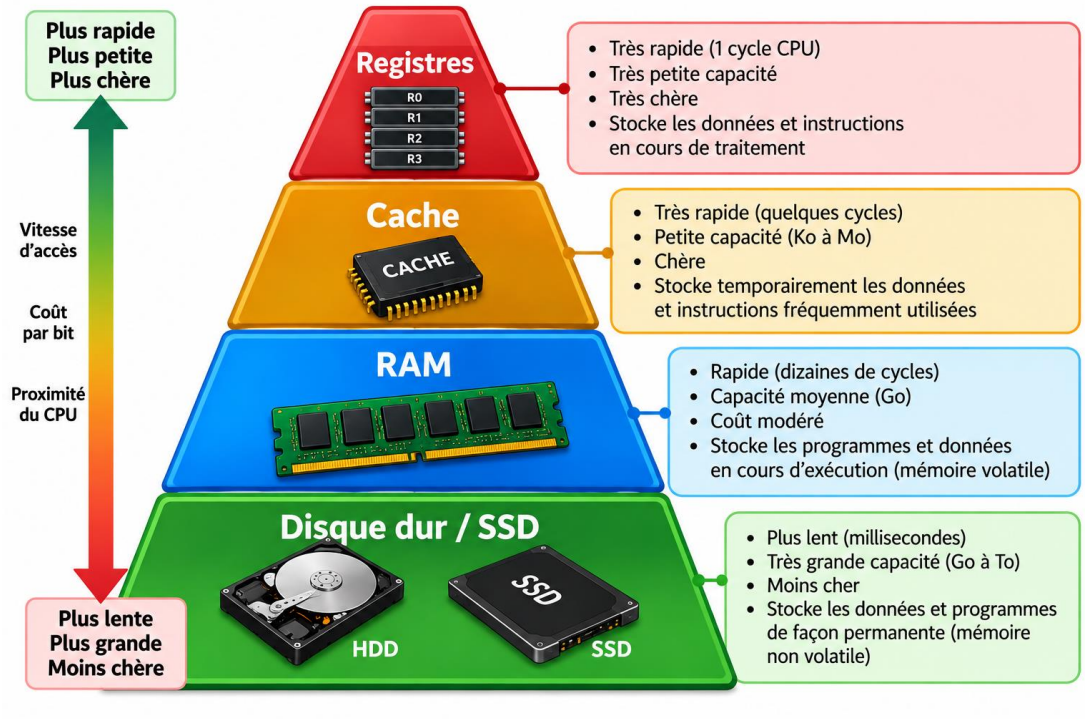

Figure 2 : Hiérarchie des mémoires

Explication de la figure 2

Plus on monte dans la pyramide, plus la mémoire est rapide mais petite et chère. Le CPU accède d'abord aux registres, puis au cache, puis à la RAM, et enfin au disque dur. En SNV, si vous travaillez sur un fichier de séquences volumineux, il sera chargé en RAM pour être traité rapidement.

Exemple concret en SNV :

Un étudiant ouvre un fichier Excel contenant 10 000 lignes de mesures. Le fichier est chargé depuis le disque dur vers la RAM. Le CPU peut alors effectuer les calculs rapidement. Si la RAM est insuffisante, l'ordinateur utilise le disque dur comme mémoire virtuelle, ce qui ralentit considérablement le traitement.

2.1.3 Les périphériques d'entrée/sortie

Les périphériques permettent à l'utilisateur de communiquer avec l'ordinateur. On distingue :

- **Périphériques d'entrée** : clavier, souris, scanner, capteurs, microscope numérique.
- **Périphériques de sortie** : écran, imprimante, haut-parleurs.
- **Périphériques d'entrée-sortie** : écran tactile, modem, clé USB.

Le tableau suivant présente des exemples de périphériques utilisés en SNV.

Tableau 3 : Périphériques courants en SNV

Type	Exemple	Utilisation en SNV
Entrée	Clavier, souris	Saisie de données, navigation dans les logiciels.
Entrée	Microscope numérique	Acquisition d'images de cellules.
Entrée	Capteur de température	Mesure automatique de la température d'une culture.
Sortie	Écran	Affichage des résultats, des graphiques.
Sortie	Imprimante	Impression de rapports, de posters.
Entrée-sortie	Clé USB	Transfert de données entre ordinateurs.
Entrée-sortie	Modem/routeur	Accès à Internet, aux bases de données en ligne.

Explication du tableau 3

- Les périphériques d'**entrée** permettent d'introduire des données dans l'ordinateur. En SNV, les capteurs et microscopes numériques sont très utilisés.
- Les périphériques de **sortie** restituent les résultats. L'écran est le plus courant.
- Les périphériques d'**entrée-sortie** permettent les deux sens : une clé USB sert à la fois à lire et à écrire des données.

Exemple concret en SNV :

Un laboratoire de microbiologie utilise un microscope numérique pour capturer des images de colonies bactériennes. Ces images sont transférées vers l'ordinateur via un câble USB (entrée-sortie). L'écran affiche l'image, et l'imprimante permet d'obtenir une version papier pour le rapport.

2.1.4 Le stockage

Le stockage permet de conserver les données de manière permanente. On distingue :

- **Disque dur (HDD)** : plateaux magnétiques, grande capacité, moins rapide.
- **SSD** (Solid State Drive) : mémoire flash, très rapide, plus cher.
- **Clé USB, carte SD** : stockage amovible.
- **Cloud** : stockage en ligne.

Le tableau suivant compare ces supports.

Tableau 4 : Comparaison des supports de stockage

Support	Capacité typique	Vitesse	Avantages	Inconvénients
HDD	500 Go – 4 To	Moyenne	Grande capacité, prix bas	Fragile, lent
SSD	128 Go – 2 To	Très rapide	Rapide, silencieux, résistant	Plus cher
Clé USB	8 – 256 Go	Moyenne	Portable	Capacité limitée
Cloud	Illimitée (selon abonnement)	Dépend du réseau	Accessible partout	Nécessite Internet

Explication du tableau 4

- Le **HDD** est adapté au stockage massif de données (archives, séquences).
- Le **SSD** est idéal pour le système d'exploitation et les logiciels : il accélère le démarrage.

- La **clé USB** est pratique pour transporter des fichiers.
- Le **cloud** permet de sauvegarder et partager des données, mais dépend de la connexion Internet.

Exemple en SNV :

Un laboratoire stocke des images de microscopie sur un disque dur externe de 4 To. Les fichiers de travail en cours sont sur le SSD pour un accès rapide.

2.1.5 La carte mère et les bus

La **carte mère** est le circuit principal qui relie tous les composants : CPU, mémoire, stockage, périphériques. Les **bus** sont les voies de communication qui transportent les données entre ces composants.

Le tableau suivant présente les principaux bus.

Tableau 5 : Types de bus

Bus	Rôle	Exemple
Bus de données	Transporte les données entre CPU, mémoire et périphériques.	Transfert d'une séquence d'ADN vers la RAM.
Bus d'adresses	Indique l'emplacement mémoire où lire ou écrire.	Localisation d'une variable dans la RAM.
Bus de contrôle	Transporte les signaux de commande (lecture, écriture).	Ordre d'écriture sur le disque dur.

Explication du tableau 5

Les bus sont comme des autoroutes : plus ils sont larges et rapides, plus les données circulent vite. La carte mère intègre ces bus et permet à tous les composants de communiquer.

Exemple concret en SNV :

Lorsqu'un logiciel de bio-informatique charge une séquence de 10 000 bases, les données transitent par le bus de données depuis le disque dur vers la RAM, puis vers le CPU pour analyse.

2.2 Les composants logiciels (Software)

2.2.1 Le système d'exploitation

Le **système d'exploitation (SE)** est le logiciel principal qui gère le matériel et permet aux applications de fonctionner. Il assure :

- La gestion des fichiers et des dossiers.
- La gestion des périphériques.
- La gestion des utilisateurs et des droits.
- L'exécution des programmes.

Le tableau suivant présente les fonctions du système d'exploitation.

Tableau 6 : Fonctions du système d'exploitation

Fonction	Description	Exemple en SNV
Gestion des processus	Coordonne l'exécution des programmes.	Lancer Excel et ImageJ en même temps.
Gestion de la mémoire	Alloue la RAM aux programmes.	Charger une image de microscope en mémoire.
Gestion des fichiers	Organise le stockage sur le disque.	Enregistrer un rapport dans un dossier.
Gestion des périphériques	Pilote les imprimantes, scanners, etc.	Imprimer un graphique.
Interface utilisateur	Fournit un environnement de travail (bureau, fenêtres).	Utiliser Windows ou Linux.

Explication du tableau 6

Le SE fait le lien entre le matériel et les logiciels. Sans lui, l'ordinateur ne pourrait pas démarrer. Les principaux SE sont Windows, Linux et macOS. En SNV, Linux est souvent utilisé pour les serveurs de calcul et la bio-informatique.

2.2.2 Les logiciels d'application

Les logiciels d'application sont des programmes conçus pour des tâches spécifiques. Le tableau suivant en présente quelques-uns utilisés en SNV.

Tableau 7 : Logiciels d'application en SNV

Domaine	Logiciel	Utilisation
Bureautique	Word, Excel, PowerPoint	Rédaction, calcul, présentation.
Statistiques	R, GraphPad Prism	Analyses statistiques, graphiques.
Bio-informatique	BLAST, SerialCloner	Comparaison de séquences, clonage.
Imagerie	ImageJ / FIJI	Analyse d'images microscopiques.
Gestion bibliographique	Zotero, Mendeley	Gestion des références.

Explication du tableau 7

Ces logiciels ne font pas partie du système d'exploitation. Ils sont installés par l'utilisateur selon ses besoins. En SNV, la maîtrise de ces outils est indispensable pour analyser et communiquer les résultats scientifiques.

2.3 Fonctionnement d'un ordinateur : le cycle d'exécution

2.3.1 Le cycle fetch-decode-execute

Pour exécuter une instruction, le CPU suit un cycle en trois étapes :

1. **Fetch (charger)** : l'instruction est lue depuis la mémoire.
2. **Decode (décoder)** : l'instruction est traduite en signaux compréhensibles.
3. **Execute (exécuter)** : l'opération est réalisée.

La figure suivante illustre ce cycle.

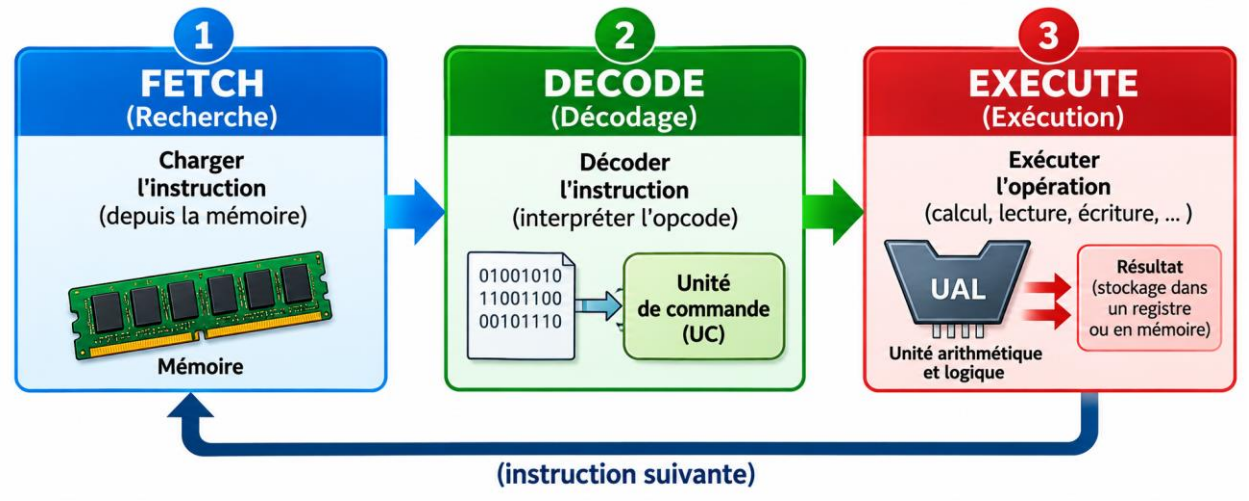

Figure 3 : Cycle fetch-decode-execute

Explication de la figure 3

Le CPU répète ce cycle des millions de fois par seconde. Par exemple, pour calculer une moyenne :

- **Fetch** : l'instruction « additionner » est chargée.
- **Decode** : le CPU comprend qu'il doit additionner deux nombres.
- **Execute** : l'UAL effectue l'addition.
- Le résultat est stocké, puis le cycle recommence pour l'instruction suivante.

2.3.2 Exemple concret : ouvrir un fichier Excel et calculer une moyenne

1. **Démarrage** : le système d'exploitation charge Excel depuis le disque dur vers la RAM.
2. **Saisie** : vous tapez des données au clavier (périphérique d'entrée).
3. **Traitement** : le CPU exécute les instructions de calcul.
4. **Affichage** : le résultat s'affiche à l'écran (périphérique de sortie).
5. **Enregistrement** : vous sauvegardez le fichier sur le disque dur (stockage).

2.4 Synthèse : architecture matérielle et logicielle

La figure suivante présente une vue d'ensemble de l'architecture d'un ordinateur.

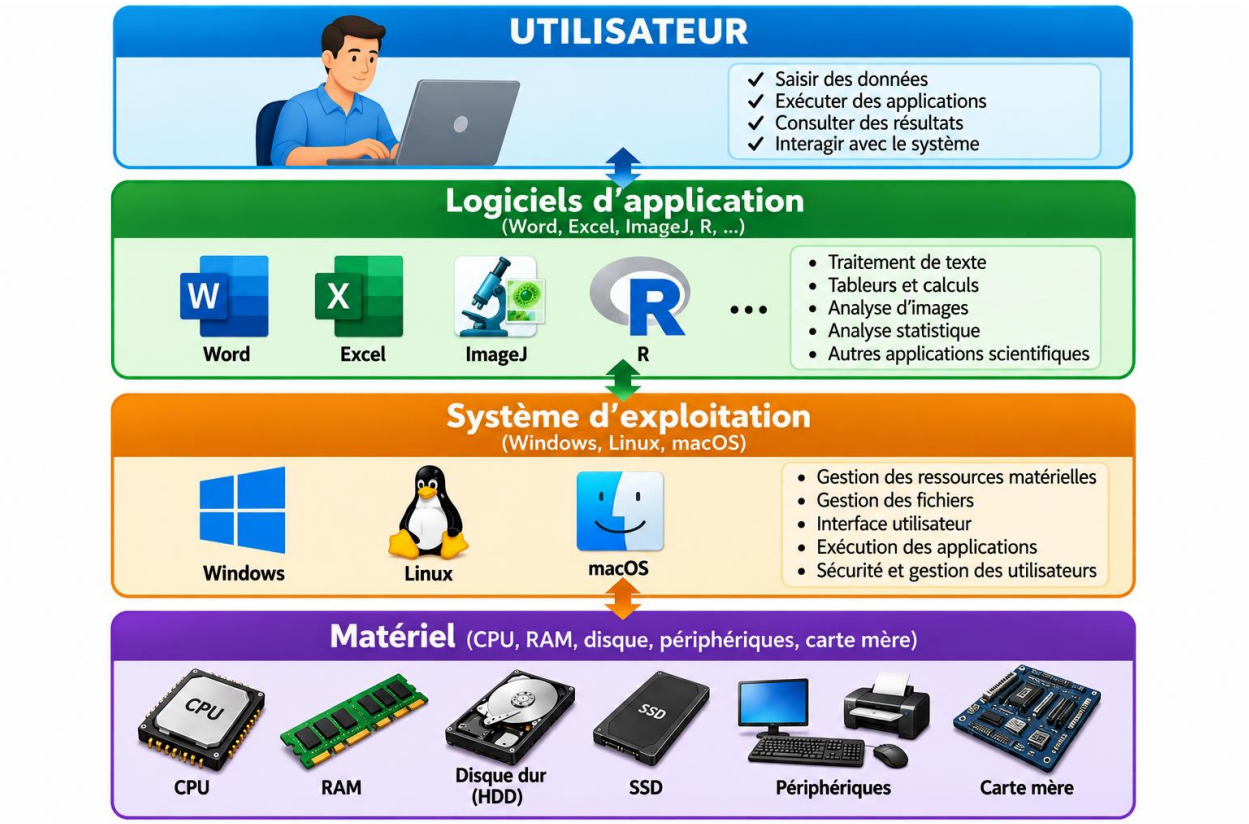

Figure 4 : Vue d'ensemble de l'architecture d'un ordinateur

Explication de la figure 4

L'utilisateur interagit avec les logiciels d'application. Ces logiciels reposent sur le système d'exploitation, qui à son tour pilote le matériel. Chaque couche dépend de la précédente. En SNV, cette architecture permet d'utiliser des outils spécialisés sans se soucier des détails électroniques.

2.5 Applications en SNV

Exemple 1 : Acquisition et analyse d'images microscopiques

- **Matériel** : microscope numérique connecté à un ordinateur.
- **Logiciel** : ImageJ.
- **Processus** : l'image est capturée, transférée en RAM, traitée par le CPU, affichée à l'écran, puis stockée sur le disque.

Exemple 2 : Analyse statistique avec R

- **Matériel** : ordinateur avec 8 Go de RAM.
- **Logiciel** : R et RStudio.
- **Processus** : importation d'un fichier CSV, calcul de moyennes, création de graphiques, exportation des résultats.

Exemple 3 : Comparaison de séquences avec BLAST

- **Matériel** : ordinateur connecté à Internet.
- **Logiciel** : navigateur web.
- **Processus** : envoi d'une séquence, traitement sur un serveur distant, réception des résultats.

2.6 Conclusion

Ce chapitre a détaillé l'architecture et le fonctionnement d'un ordinateur :

- Le **matériel** comprend le CPU, la mémoire, les périphériques, le stockage et la carte mère.
- Le **logiciel** se compose du système d'exploitation et des applications.
- Le **cycle fetch-decode-execute** explique comment le CPU traite les instructions.
- En SNV, cette architecture permet d'utiliser des outils puissants pour l'analyse de données, l'imagerie et la bio-informatique.

Comprendre ces bases est essentiel pour utiliser efficacement les logiciels spécialisés et pour interpréter les performances des machines.

📌 À retenir

- **CPU** = UC + UAL + registres + horloge.
- **RAM** = mémoire temporaire ; **ROM** = mémoire permanente.
- **Périphériques** : entrée, sortie, entrée-sortie.
- **Stockage** : HDD, SSD, USB, cloud.
- **Système d'exploitation** : gère le matériel et les applications.
- **Cycle fetch-decode-execute** : charger, décoder, exécuter.

❓ Questions de révision

1. Quels sont les composants principaux du CPU ?
2. Quelle est la différence entre RAM et ROM ?
3. Citez trois périphériques d'entrée utilisés en SNV.
4. Quels sont les avantages d'un SSD par rapport à un HDD ?
5. Quel est le rôle du système d'exploitation ?
6. Décrivez le cycle fetch-decode-execute.
7. Donnez un exemple concret d'utilisation de l'architecture informatique en SNV.
8. Expliquez pourquoi la RAM est essentielle pour le traitement de données volumineuses en bio-informatique.