



VEILLE INFORMATIQUE

7 FEVRIER 2025

MOHAMED EL HANI

SOMMAIRE :

Sommaire :.....	3
Intro	4
Démarche de veille	5
Analyse argumentée.....	6
Défis techniques et innovations	6
Enjeux économiques et stratégiques.....	8
Mes recommandations	10
Conclusion	11
Glossaire	12
Sources	15

INTRO

A l'heure actuelle le marché des systèmes d'exploitation pour les particuliers est dominé par Windows (75% des parts de marchés) et MacOS(15%) et une insignifiante part pour Linux et autre OS comme Chrome OS, avec aucun concurrent majeur depuis plusieurs décennies pour différentes raisons et ce malgré qu'il existe différent constructeur d'ordinateur, en effet on pourrait penser que puisque que ces constructeurs ont la main sur leur ordinateurs, ils pourraient profiter de cela pour un inclure un système d'exploitation personnalisé par rapport a la concurrence pour se démarquer et pouvoir récupérer plus d'information sur ses utilisateurs en leur offrant une expérience unique ce qui est le cas dans les marchés du mobiles, en effet un téléphone Samsung et un téléphone Xiaomi, bien que partageant le même système principale (Android) son totalement différent l'un de l'autre par rapport a la personnalisation de cet OS laissant, Pourquoi ce n'est pas le cas dans le marché de l'ordinateur, partageant le même principe que celui de l'ordinateur ? Vaut-il le coup pour une entreprise comme DELL ou ASUS d'investir dans un OS customisé pour attirer leurs clients et se démarquer des autres ?

Dans un marché des OS pour PC dominé par Windows et macOS, quelles stratégies techniques, économiques et réglementaires permettraient à une entreprise de créer un système d'exploitation viable ? Dans quelles conditions un tel projet pourrait-il devenir rentable ou répondre à des besoins de niche ?

Le but de cette veille est donc de comprendre les barrières et défis techniques dans la création d'un OS personnalisé pour une entreprise déjà installée et stable, de comprendre les enjeux et les avantages économiques et stratégiques et l'état du marché des systèmes d'exploitation en général, en utilisant de nombreux exemples et sources récentes et anciennes pour avoir une réponse concrète et une stratégie fiable avec des recommandations stratégiques.

DEMARCHE DE VEILLE

Pour mener à bien cette veille, différents outils ont été utilisés :

Des alertes automatisées via Google Alerts (custom OS, Linus Fork, OS market...) permettant d'être à jour sur les nouveautés.

Surveillance sur les réseaux sociaux, notamment Twitter (#OSDev, #Linux et @LinusTech), Reddit (r/osdev, r/programming) et Youtube (V2F)

Surveillance technologique, sur GitHub notamment avec suivi des projets open sources (Linux, Fuchsia)

Surveillance des communications entre connaisseurs, notamment sur le site OSdev.org (forum et wiki communautaire)

Tous ces outils ont permis une veille optimale avec plusieurs informations précises et à jour, sur plusieurs sources et fiables.

ANALYSE ARGUMENTEE

Défis techniques et innovations

Pour créer un OS capable de rivaliser avec Microsoft et Apple il va falloir être au niveau avec ces mastodontes sur presque tout les domaines, pour commencer il faut choisir un kernel ou noyau, la partie qui permet à l'ordinateur de gérer ses ressources et de permettre la communication entre les logiciels et matériels, le Choix d'un kernel et un choix primordial dans la création d'un OS maison, on peut sélectionner un kernel Open source comme celui de Linux ou Zircon (Google) ou créer un kernel maison comme ce que fait Windows et Apple, le choix le plus efficace pour une entreprise déjà stable et la création d'un kernel maison, puisque qu'il permet de facilement comprendre son fonctionnement et de pouvoir le modifier plus librement, cependant cela prend du temps et de l'argent à être créé avec un prix estimé de 1000000\$ à 1500000\$ par an et est une partie cruciale d'un développement d'OS puisque qu'un mauvais kernel peut conduire à de faible vente du à un mauvais OS (Windows Vista a échoué en partie à cause de son kernel mal optimisé).

En plus de cela, la création d'un OS est un système verrouillé par l'effet de réseau/club. En effet, pour qu'un OS puisse viser un grand public, il faut le rendre disponible sur tout hardware possible et compatible avec le plus de logiciels possibles, ou sinon créer des alternatives au logiciel soi-même (ce que fait MacOS). cependant la comptabilité Hardware est difficile puisque qu'elle une communication avec les fabricants (Intel, AMD, Nvidia) pour pouvoir avec un OS compatible avec les futures Hardware, ce que Windows possède déjà, ce qui rend la tâche de les concurrencer beaucoup plus difficile puisque que Windows sera compatible avec les futurs Hardware bien en avance et Apple est son propre constructeur d'ordinateur, leur permettant d'être libre et d'avoir le plein contrôle sur les compatibilités, en plus de cela les créateurs de logiciels privilégieront une compatibilité avec Windows et MacOS plutôt que les autres OS, créant une isolation pour les plus petits OS comme ce que subit Linux, constamment privé de plusieurs logiciels populaires chez les utilisateurs de Windows. En plus de cela, il faut rajouter les contraintes ergonomiques (interfaces UI/UX) et de sécurité (faille zero-day et maintenance constante pour éviter les failles de sécurité), mais le défi principal reste le budget et l'effectif nécessaire pour rattraper les décennies de retard par rapport aux concurrents actuels. En effet, un OS capable de rivaliser avec Windows et macOS a demandé plus de 300 ingénieurs, jusqu'à 1 milliard de dollars et presque 10 ans de développement à Google pour créer un Fuchsia, un OS supposé rivaliser avec Microsoft et Apple, mais ayant échoué et laissé à l'abandon aujourd'hui au profit de ChromeOS,

Cependant il est important de constater que malgré tout ces défis et contraintes il y a des avantages à arriver en retard sur le marché et des moyens de combattre ces barrières, en effet bien que Microsoft et Apple ont des décennies d'avance sur leurs OS, cette avance peut devenir une contrainte puisque que leurs kernel et fondations de code ont du mal à s'adapter aux nouvelles technologies, principalement l'ARM pour Windows et ont un kernel qui commence à être dépassé par les nouvelles technologies de kernel (Micro-Kernel et kernel hybride) qui sont plus efficaces et plus sécurisées que les kernels classiques toujours utilisés par Microsoft et Apple (Bien que Apple commence à réussir sa transition vers l'hybride), c'est notamment la technologie utilisée par Fuchsia et son kernel Zircon, un micro-kernel, sensé être plus efficace et sécurisé que ceux de Windows et MacOS, une entreprise stable qui veut commencer à créer un OS maison a tout intérêt à exploiter ces nouvelles technologies ARM et nouveaux kernels pour réussir à se démarquer de ces mastodontes.

Concernant les problèmes liés aux dépendances logicielles et compatibilité il existe trois solutions possibles, on peut essayer de fournir soit même des équivalents aux logiciels populaires, comme le fait l'OS chinois OpenKylin qui conçoit tout les logiciels d'outil pour être indépendant des Américains et qui bénéficie donc de subventions publiques pour pallier la barrière économique, cependant cette solution demande beaucoup d'investissement et de temps sans subvention publique, l'autre solution plus fiable et de contourner le problème avec des technologies récentes qui permettent d'améliorer la compatibilité avec les logiciels conçus pour Windows, comme WINE ou Proton utilisés par Steam OS, capable de faire tourner 80% des logiciels conçus pour Windows, cependant certains logiciels restent pas accessibles comme les logiciels Adobe et pour cela une dernière solution est possible, les partenariats pour pouvoir inclure des versions faites spécialement pour l'OS maison.

Bien qu'il existe de nombreuses barrières et défis à l'entrée, la création d'OS peut aussi être exploitée pour obtenir des avantages par rapport aux OS existants, surtout dans une ère où la technologie avance énormément et où l'obsolescence des codes préexistants de Windows se fait de plus en plus ressentir, notamment avec l'IA, se placer et exploiter les nouvelles technologies pour réduire les barrières et gagner de l'avance par rapport aux géants déjà présents.

Bien que le marché des OS semble impossible et pas rentable à exploiter, il existe des enjeux économiques et stratégiques qui pourraient rendre le tout intéressant pour une entreprise.

Enjeux économiques et stratégiques

Bien que créer un OS soit difficile et possède beaucoup de barrières, les enjeux économiques et stratégiques sont très importants. Le budget pour un OS viable est entre 100M et 1Md, en prenant l'exemple de Fuchsia, rendre le projet économiquement viable. On peut utiliser un modèle de financement basé sur l'open source pour attirer le plus de personnes, mais faire payer une utilisation commerciale, des partenariats avec d'autres constructeurs d'ordinateurs ou des subventions publiques (surtout en UE et en Chine). Avec un OS de cette envergure, la stratégie d'implémentation se pose, soit concurrencer en face les géants Windows et Apple, stratégie très risquée mais qui peut rapporter gros si réussie, la stratégie adoptée par Google pour l'échec fuchsia, mais aussi pour la réussite. Chrome OS à la fois niche et général, ou bien stratégie plus stable, s'attaquer à une niche comme l'IoT, le gaming, l'éducation, la sécurité, L'IA ou autre, une niche qui pourrait avoir de l'importance dans le futur, comme un OS basé sur l'IA embarquée ou un OS sécurisé pour le vendre aux entreprises sensibles (banques, militaire) comme fait Qubes OS à l'heure actuelle, adulé par les experts en cybersécurité grâce à sa virtualisation.

En effet la meilleure solution semble être le niche et c'est ce que tout les constructeur d'os dont a l'heure actuelle, Steam OS pour le gaming, Harmony OS pour l'IoT chinois, Qubes OS pour la cybersécurité, cela permet de réduire les couts de l'OS et les risques liées en plus de pouvoir avoir accès a des licences moins couteuse du a leur niche et si possible de l'intégration optimale ce qui pousserait a un marketing basé sur l'optimisation pour cette niche spécialement, dans le monde actuelle ou Microsoft possède 75% des parts du marché de l'OS, l'exploitation d'une niche semble être la meilleurs solution, il est aussi possible de visé plusieurs niche a la fois, comme la fais Samsung avec Tizen qui visait le mobile et la télévision a la fois, puis c'est spécialisé sur la télévision, marché beaucoup plus intéressant. Un OS sur une niche réussie pourrait rapporter entre 20 et 50 millions de dollars par an (revenu de QNX, un OS qui a ciblé l'automobile et le médical), un investissement donc plus que rentable puisqu'il peut se rentabiliser en entre 2 et 20 ans.

De manière plus risquée, on peut aussi essayer de viser des futures niches, comme les OS post-quantiques ou avec IA. De cette manière, on pourra avoir une avance si cette niche évolue et devient de plus en plus importante, mais elle possède un risque si elle ne marche pas.

Il est aussi possible de se baser sur un point de vue de la souveraineté et, en effet, tenter de créer un OS conçu pour qu'un pays ne soit plus dépendant des géants américains. Cela permettrait d'obtenir des subventions publiques et de réduire les dépendances aux OS étrangers tout en obtenant une mainmise sur le marché de l'OS

et de ses dépendances dans ce pays. C'est ce qui se passe avec la Chine (iSoftStone pour l'entreprise et OpenKylin pour l'armée et Harmony OS pour l'IoT en général, tous deux avec des ambitions d'être installés sur tout le pays grâce au projet Xinchuang, qui a pour but de remplacer tous les Windows de Chine par des OS locaux) et la Russie (Astra Linux).

À titre personnel, une niche qui semble être une opportunité est la niche des NPU, des processeurs conçus pour et responsables des calculs intensifs des IA. Créer un OS capable d'exploiter à pleine puissance ces NPU pourrait être un grand investissement pour l'avenir, puisque les IA deviennent omniprésentes dans les OS et ordinateurs (Copilot, Siri) ; et la niche des processeurs ARM, qui commence à être exploitée par Apple, mais toujours pas par Windows, qui semble avoir du retard dû à leur code compliqué à transférer efficacement. Dans un monde toujours plus connecté, les OS basés sur l'IoT et les objets connectés comme Harmony OS ou Google Brillo sont toujours un investissement rentable.



MES RECOMMANDATIONS

Mes recommandations pour une entreprise voulant essayer de s'installer dans le marché des OS sont de premièrement cibler une niche avec du potentiel dans le futur et où la concurrence n'est pas très forte, notamment dans L'IA embarquée, les NPU ou L'IoT, mais aussi dans le gaming ou la sécurité, qui sont des domaines fortement en demande mais sous-exploités par les géants (la seule concurrence étant Steam OS pour le gaming et Qubes OS pour la sécurité). Après la niche sélectionner, je recommande d'exploiter les nouvelles technologies ARM et micro-kernel que les géants n'ont pas encore réussi à parfaitement maîtriser pour pouvoir se démarquer et avoir de l'avance.

On peut aussi s'appuyer sur les projets Open Source comme Linux ou Zircon pour réduire massivement les coûts de production et attirer une communauté de développeurs qui connaissent déjà ces projets. En plus de cela, si possible, je conseillerais d'utiliser les subventions européennes ou chinoises pour avoir plus d'aide financière en n'utilisant que des technologies européennes ou chinoises, si applicable, pour avoir un OS souverain qui pourra, dans le futur, dépasser sa niche et concurrencer directement les géants dans ce territoire. Il peut aussi être intéressant d'investir et de financer les technologies utilisées pour permettre de rattraper Windows sur certains domaines, même si la technologie ne nous appartient pas entièrement, elle permettra aux autres concurrents de s'attaquer aussi aux géants (ce que Valve a fait avec Proton pour rendre Steam OS viable).



CONCLUSION

Développer un OS maison dans un marché dominé par Windows et MacOS est un défi de taille, mais pas impossible.

Pour survivre dans ce projet difficile, le plan se repose sur l'exploitation des nouvelles technologies (ARM, NPU, micro-kernel, etc.) et des niches technologiques (IA, gaming, sécurité IoT, etc.) en plus d'une adaptation aux enjeux politiques des pays du constructeur. Si ces conditions sont remplies alors une nouvelle OS pourrait avoir lieu et basculer le monde de l'informatique d'ici quelques années ou décennies, surtout dans un monde où l'IA et l'ARM prennent de plus en plus de place, les OS se diversifieront de plus en plus et se spécialiseront, jusqu'à atteindre un point où la concurrence redeviendra possible, jusqu'à atteindre un point où chaque OS sera une solution spécialisée dans un monde informatique fragmenté.

GLOSSAIRE

1. Système d'exploitation (OS)

Un logiciel qui gère les ressources matérielles et logicielles d'un ordinateur et permet aux applications de fonctionner. Exemples : Windows, macOS, Linux.

2. Kernel (Noyau)

La partie centrale du système d'exploitation qui gère la communication entre le matériel et les logiciels. Il existe plusieurs types de kernels : monolithique, micro-kernel et hybride.

3. Monolithique (Kernel)

Un type de noyau où toutes les fonctionnalités essentielles (gestion mémoire, fichiers, périphériques) sont intégrées dans un seul bloc de code. Exemples : Linux et Windows.

4. Micro-kernel

Un noyau minimaliste où seules les fonctions de base sont intégrées, et le reste fonctionne comme des services externes. Avantage : meilleure sécurité et modularité. Exemple : Zircon (Google Fuchsia).

5. Kernel hybride

Un mélange des approches monolithique et micro-kernel, optimisant performances et sécurité. Exemple : macOS utilise un noyau hybride.

6. Effet de réseau (ou effet club)

Un phénomène économique où la valeur d'un produit augmente avec le nombre d'utilisateurs. Exemples : Windows et macOS dominant car les développeurs créent en priorité des logiciels compatibles avec ces systèmes.

7. Compatibilité logicielle

La capacité d'un OS à exécuter des logiciels développés pour d'autres systèmes. Linux, par exemple, rencontre des difficultés de compatibilité avec de nombreux logiciels Windows.

8. Open Source

Un modèle de développement où le code source d'un logiciel est accessible à tous, permettant à une communauté de développeurs de l'améliorer. Exemples : Linux, Android, Fuchsia.

9. WINE

Un logiciel permettant d'exécuter des applications Windows sur Linux sans émulation complète du système.

10. Proton

Une version avancée de WINE développée par Steam pour exécuter des jeux Windows sous Linux avec de meilleures performances.

11. Virtualisation

Une technologie permettant d'exécuter plusieurs systèmes d'exploitation sur une même machine en créant des environnements isolés (machines virtuelles). Exemple : QubesOS.

12. Zero-Day (faille de sécurité)

Une vulnérabilité dans un logiciel qui n'a pas encore été corrigée et peut être exploitée par des cybercriminels.

13. Interface UI/UX

UI (User Interface) : L'aspect visuel et interactif d'un logiciel.

UX (User Experience) : L'expérience globale de l'utilisateur en termes de facilité d'utilisation et d'ergonomie.

14. IoT (Internet of Things)

L'Internet des objets : interconnexion d'appareils intelligents (montres connectées, voitures autonomes, etc.) qui communiquent via Internet. Exemples d'OS spécialisés : HarmonyOS, Brillo (Google).

15. ARM (Architecture ARM)

Un type d'architecture de processeur optimisé pour la faible consommation énergétique, utilisé dans les smartphones, tablettes et certains ordinateurs récents (ex : Mac M1/M2).

16. NPU (Neural Processing Unit)

Un processeur spécialisé dans l'exécution des tâches d'intelligence artificielle (IA), comme la reconnaissance vocale ou l'analyse d'images.

17. OS souverain

Un système d'exploitation conçu pour réduire la dépendance d'un pays aux logiciels étrangers. Exemples : OpenKylin (Chine), Astra Linux (Russie).

18. Subventions publiques

Aides financières accordées par un gouvernement pour soutenir un projet. Dans le domaine des OS, elles peuvent servir à financer des alternatives locales à Windows et macOS.

19. SteamOS

Un système d'exploitation basé sur Linux, optimisé pour les jeux vidéo et utilisé par la console Steam Deck.

20. QubesOS

Un système d'exploitation axé sur la cybersécurité qui utilise la virtualisation pour isoler chaque application et réduire les risques de piratage.

21. Fuchsia (Google)

Un OS expérimental développé par Google, basé sur un micro-kernel Zircon, qui n'a pas réussi à s'imposer face à Android et ChromeOS.

22. ChromeOS

Un système d'exploitation léger basé sur Linux et conçu pour fonctionner principalement avec des applications web.

23. Post-quantique (OS ou cryptographie)

Un domaine de recherche visant à développer des technologies résistantes aux futurs ordinateurs quantiques, qui pourraient casser les systèmes de sécurité actuels.

SOURCES

<https://wiki.osdev.org/Kernel>

<https://wiki.osdev.org/Microkernel>

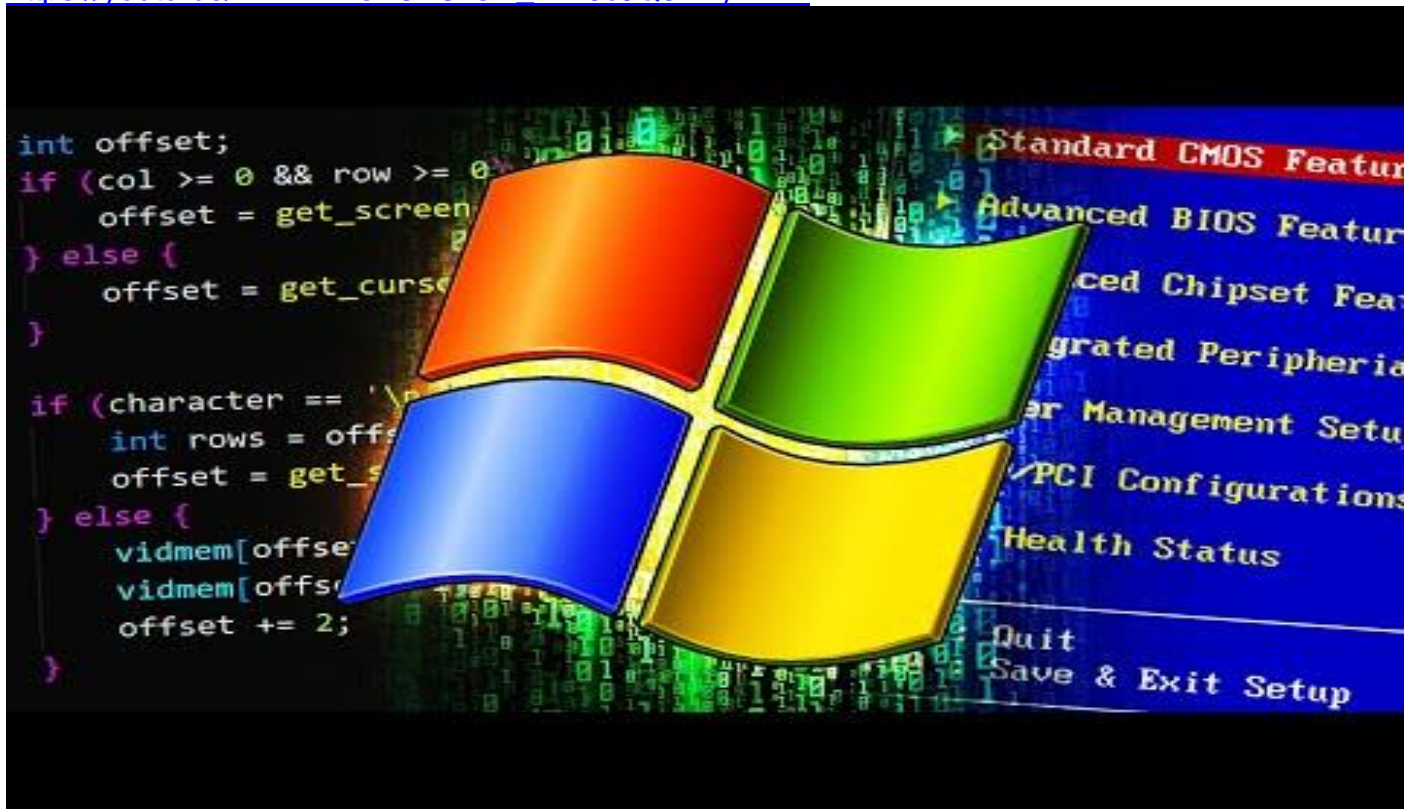
<https://forum.osdev.org/>

[https://en.wikipedia.org/wiki/Fuchsia_\(operating_system\)](https://en.wikipedia.org/wiki/Fuchsia_(operating_system))

<https://en.wikipedia.org/wiki/SteamOS>

<https://en.wikipedia.org/wiki/Unix-like>

<https://youtu.be/ELTwwTsR5w8?si=vxF0cJQ3R1yxZY7>



<https://www.comptoir-hardware.com/actus/modding-et-systemes/47696-linux-et-macos-sapent-inlassablement-les-parts-de-marche-de-windows.html>

<https://fuchsia.dev/fuchsia-src/development>

<https://www.01net.com/actualites/pour-reduire-sa-dependance-a-windows-et-macos-la-chine-travaille-sur-son-propre-systeme-dexploitation.html>

<https://www.journaldugeek.com/2022/07/07/kylin-los-chinois-qui-reve-de-remplacer-windows-et-macos/>

<https://www.canalys.com/newsroom/mainland-china-pc-market-q3-2024>

<https://www.gubes-os.org/doc/>

<https://wiki.archlinux.org/title/Kernel>

<https://gs.statcounter.com/os-market-share>

https://fr.wikipedia.org/wiki/Astra_Linux

https://wiki.osdev.org/ARM_Overview