

Dépréciation VBScript : Délais et prochaines étapes



Naveen Shankar



22 mai 2024

Les options de scripting pour le développement web et l'automatisation des tâches se modernisent. Pour vous offrir les options les plus modernes et efficaces, nous remplaçons VBScript par des alternatives plus avancées telles que JavaScript et PowerShell. Découvrez ce que signifie la dépréciation VBScript pour vous et comment vous pouvez vous préparer.

Qu'est-ce que VBScript ?

Visual Basic Scripting Edition, communément appelé VBScript, est un langage de script léger introduit pour la première fois par Microsoft en 1996. Le langage a été disponible comme composant système sous Windows OS et a été largement utilisé pour automatiser des tâches et contrôler des applications sur des systèmes Windows. Il est souvent intégré dans des pages HTML pour ajouter une interactivité dynamique et des fonctionnalités aux pages web, et est couramment utilisé en conjonction avec des technologies Microsoft telles qu'Active Server Pages (ASP) et Windows Script Host (WSH). Cependant, avec l'avancement de la technologie, des options plus modernes et efficaces sont désormais disponibles.

Pourquoi VBScript est-il obsolète ?

La technologie a évolué au fil des années, donnant naissance à des langages de script plus puissants et polyvalents tels que JavaScript et PowerShell. Ces langages offrent des capacités plus larges et sont mieux adaptés aux tâches modernes de développement web et d'automatisation.

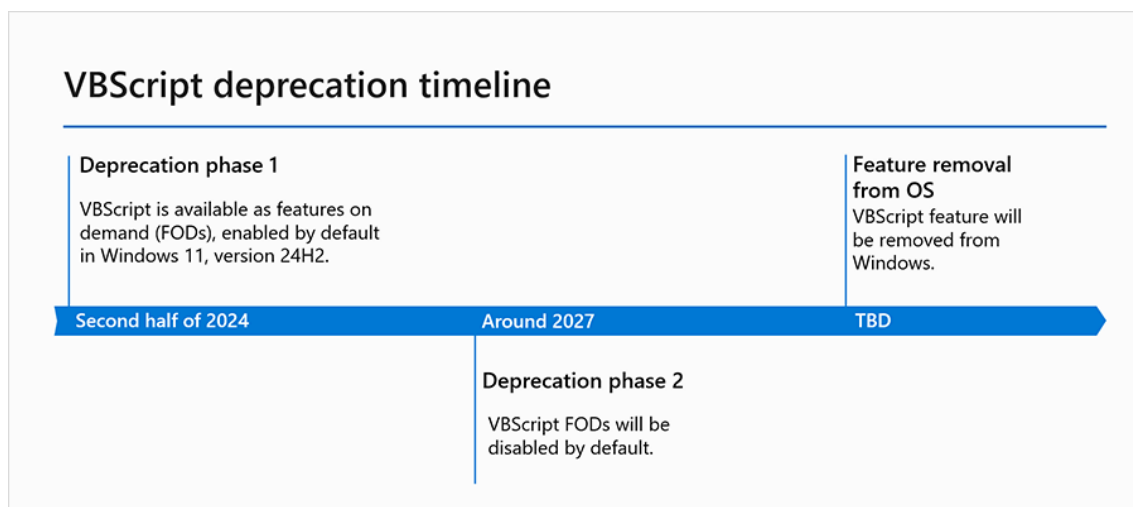
Conseil : La dépréciation est une étape du cycle de vie du produit où une fonctionnalité n'est plus en développement actif et peut être supprimée lors des futures versions d'un produit ou d'un service en ligne. C'est un processus progressif qui peut s'étendre sur quelques mois ou années. La fonctionnalité obsolète est généralement destinée à être remplacée par quelque chose de meilleur, plus avancé ou plus fonctionnel. La fonctionnalité continuera généralement à fonctionner et est entièrement supportée jusqu'à sa suppression officielle. Après suppression, la fonctionnalité ou la fonctionnalité ne fonctionnera plus. Supprimer un composant obsolète aide à réduire la complexité tout en vous gardant sécurisé et productif.

Plan de dépréciation VBScript

Compte tenu du déclin de l'utilisation de VBScript au profit de technologies web plus modernes, nous avons élaboré un plan de dépréciation progressive pour VBScript. Passons en revue la chronologie de ces changements.

En octobre 2023, nous [avons annoncé](#) notre engagement à offrir les meilleures et les plus efficaces expériences en dépréциant progressivement VBScript. À partir de la nouvelle version du système d'exploitation prévue pour la fin de l'année, VBScript sera disponible en tant que fonctionnalités à la demande (FOD). Cette fonctionnalité sera complètement retirée des futures versions du système Windows, à mesure que nous passerons aux expériences PowerShell plus efficaces.

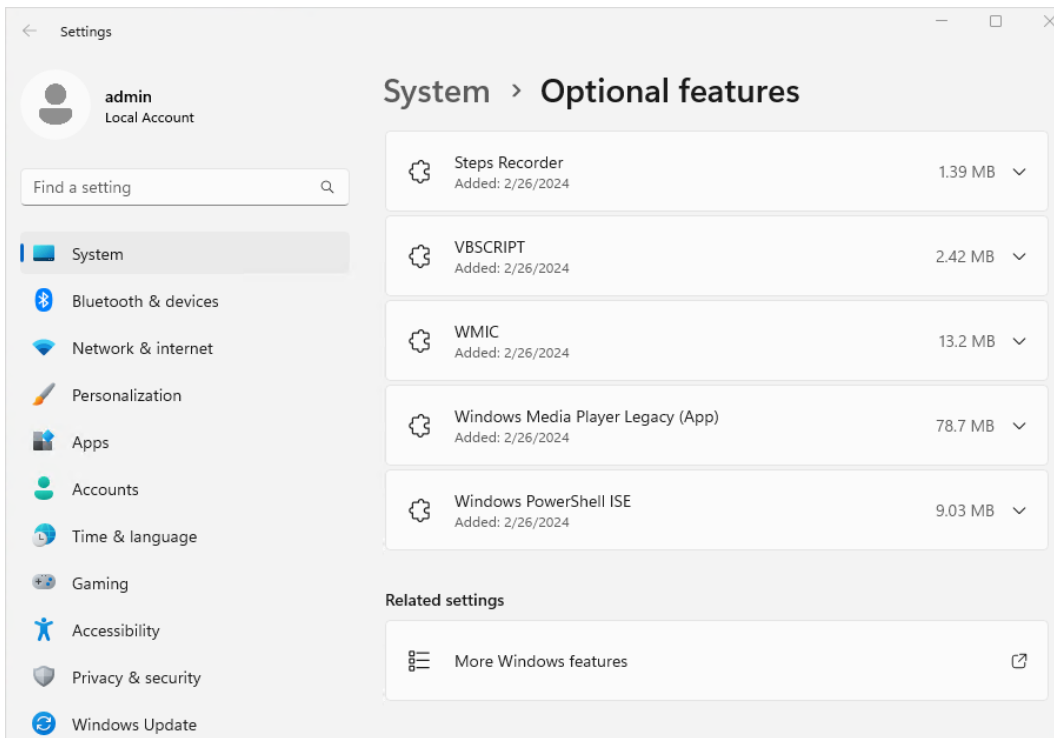
La dépréciation se fera en trois phases.



Une chronologie visuelle des dates importantes pour les phases de dépréciation de VBScript.

Phase 1

Dans la première phase, les FOD VBScript seront préinstallés par défaut dans tous les Windows 11, version 24H2 et activés. Cela aide à ne pas perturber vos expériences si vous dépendez de VBScript pendant que vous migrez vos dépendances (applications, processus, etc.) hors de VBScript. Vous pouvez voir les FOD VBScript activés par défaut dans **Start > Paramètres > Système > Fonctionnalités optionnelles**.



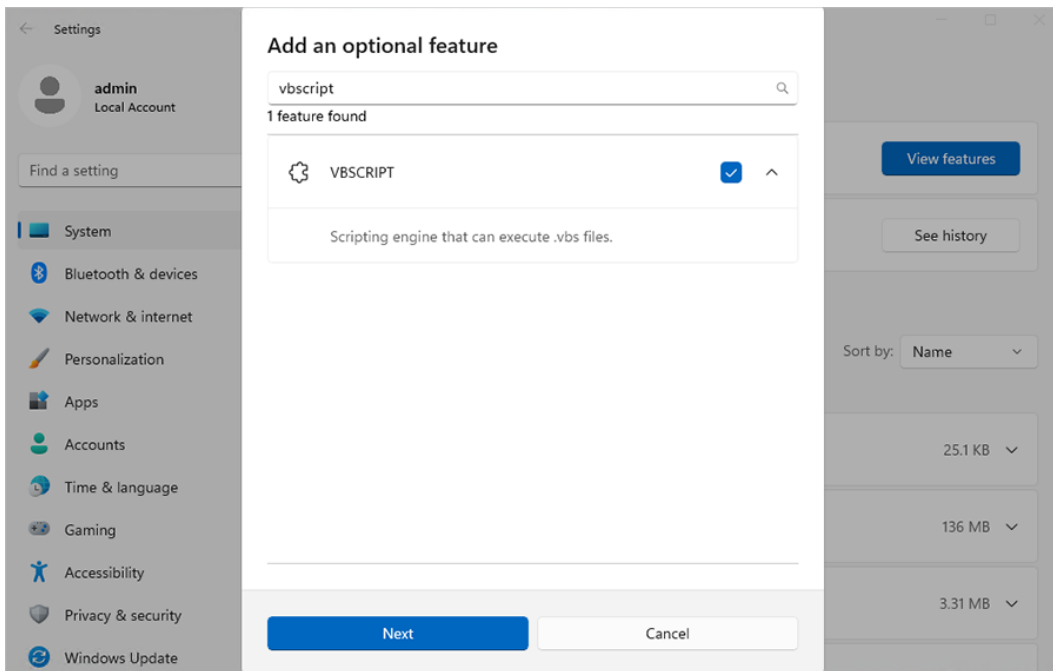
Capture d'écran des paramètres système de Windows montre VBScript installé dans la section Fonctionnalités optionnelles.

Phase 2

Vers 2027, les FODs VBScript ne seront plus activés par défaut. Cela signifie que si vous dépendez encore de VBScript à ce moment-là, vous devrez activer les FOD pour éviter que vos applications et processus ne rencontrent des problèmes.

Suivez ces étapes si vous souhaitez continuer à utiliser les FOD VBScript :

1. Va dans **Start > Paramètres > Système > Fonctionnalités optionnelles**.
2. Sélectionnez **Voir les fonctionnalités** à côté de l'option « Ajouter une fonctionnalité optionnelle » en haut.
3. Tapez « VBSCRIPT » dans la boîte de recherche et sélectionnez la case à cocher à côté du résultat.
4. Pour activer la fonction désactivée, appuyez sur **Suivant**.



Capture d'écran d'une boîte de

dialogue pour ajouter une fonctionnalité optionnelle avec une case à cocher à côté de VBScript.

Phase 3

VBScript sera retiré du service et supprimé des futures versions de Windows. Cela signifie que toutes les bibliothèques de liens dynamiques (.dll fichiers) de VBScript seront supprimées. En conséquence, les projets qui dépendent de VBScript cesseront de fonctionner. D'ici là, nous nous attendons à ce que vous ayez adopté des alternatives suggérées.

Projets VBA utilisant VBScript

Visual Basic for Applications (VBA) permet aux utilisateurs d'automatiser des tâches répétitives et de personnaliser les fonctionnalités au sein de la suite Microsoft Office. Cela inclut Excel, Word, PowerPoint, Access et certaines autres applications. Avec VBA, vous avez pu écrire des scripts (macros) pour manipuler des données, créer des formulaires personnalisés, automatiser des rapports, interagir avec d'autres applications et effectuer diverses autres tâches pour rationaliser les flux de travail et améliorer la productivité.

Actuellement, VBScript peut être utilisé en VBA pour deux scénarios :

- Scénario 1 : Appeler un script .vbs directement depuis VBA.
- Scénario 2 : Utiliser VBScript comme référence typelib (comme l'expression régulière VBScript) dans VBA.

Vous pouvez continuer à utiliser les solutions existantes si vos solutions VBA comportent les scénarios ci-dessus, car la Phase 1 ne vous affectera pas. Mais les phases futures vous affecteront, alors soyez attentif aux nouveaux développements.

Si vous constatez une erreur d'exécution ou une erreur de compilation lors de l'exécution des projets VBA, vérifiez que les FODs VBScript ne sont pas désactivés par le paramètre administrateur.

Prochaines étapes si mon application ou mon site web dépend de VBScript

Considérons deux solutions modernes pour remplacer VBScript, à savoir PowerShell et JavaScript.

Migrer vers PowerShell

Nous recommandons de migrer vers PowerShell si vous :

- Ayez des sites web ou des applications dépendants de VBScript pour automatiser les tâches.
- Utilisez les actions personnalisées VBScript comme fonctionnalité dans les paquets d'installateur. Pendant le processus de configuration, ces actions personnalisées peuvent utiliser la session d'installation et effectuer des tâches complexes. Ces actions personnalisées peuvent cesser de fonctionner après dépréciation.

Dans ces cas, nous vous recommandons de migrer vers PowerShell. Apprenez comment faire cela à [la suite Convertir VBScript en Windows PowerShell](#).

Migrer vers JavaScript

Comme la fonctionnalité de VBScript est actuellement limitée aux navigateurs antérieurs à Internet Explorer 11, nous recommandons de migrer vos pages web vers JavaScript avant la phase 2. JavaScript offre une compatibilité multi-navigateurs, fonctionnant parfaitement sur des navigateurs modernes tels que Microsoft Edge, Mozilla Firefox, Google Chrome et Apple Safari. Notamment, ces navigateurs n'ont jamais mis en place le support de VBScript.

Si votre page web fonctionne correctement sur ces navigateurs modernes, alors VBScript n'est pas impliqué, et sa dépréciation ne vous affectera pas.

Commencez à planifier votre migration et restez en contact !

Nous sommes là pour vous aider avec les dernières mises à niveau concernant la dépréciation de VBScript et pour fournir un support supplémentaire lors de la migration de vos dépendances lors des phases 1, 2 et 3.

Y a-t-il des considérations ou des scénarios particuliers dont vous aimeriez parler ? Merci de nous laisser un commentaire ci-dessous.

En attendant, consultez des ressources supplémentaires liées à la fonctionnalité et à la dépréciation de VBScript :

<https://techcommunity.microsoft.com/blog/windows-itpro-blog/vbscript-deprecation-timelines-and-next-steps/4148301>

- [Création de scripts avec Visual Basic](#)
- [VBScript n'est plus pris en charge en mode périphérique IE11 \(Windows\)](#)
- [PowerShell Documentation - PowerShell](#)
- [À propos des scripts - PowerShell](#)
- [Objet expression régulière \(RegExp\)](#)

Continue la conversation. Trouvez les meilleures pratiques. Ajoutez en favori la [communauté Windows Tech](#), puis suivez-nous [@MSWindowsITPro](#) sur X/Twitter. Vous cherchez du soutien ? Consultez [Windows on Microsoft Q&R](#).