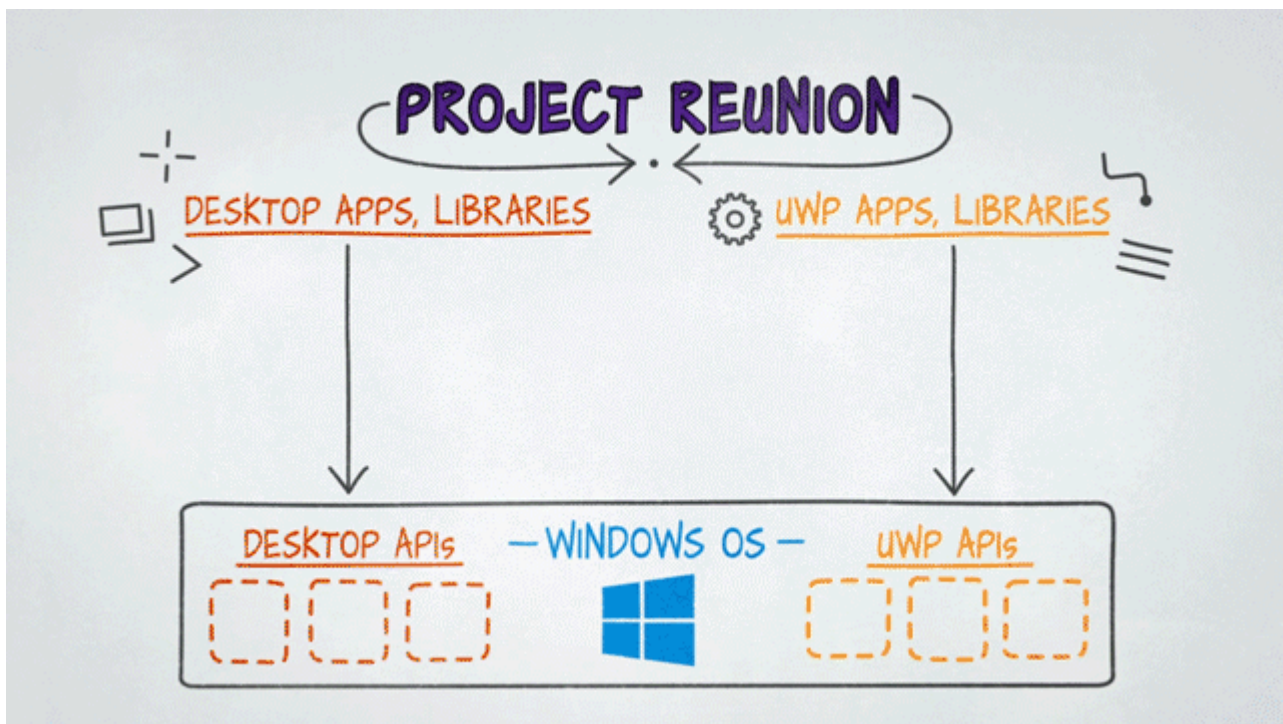


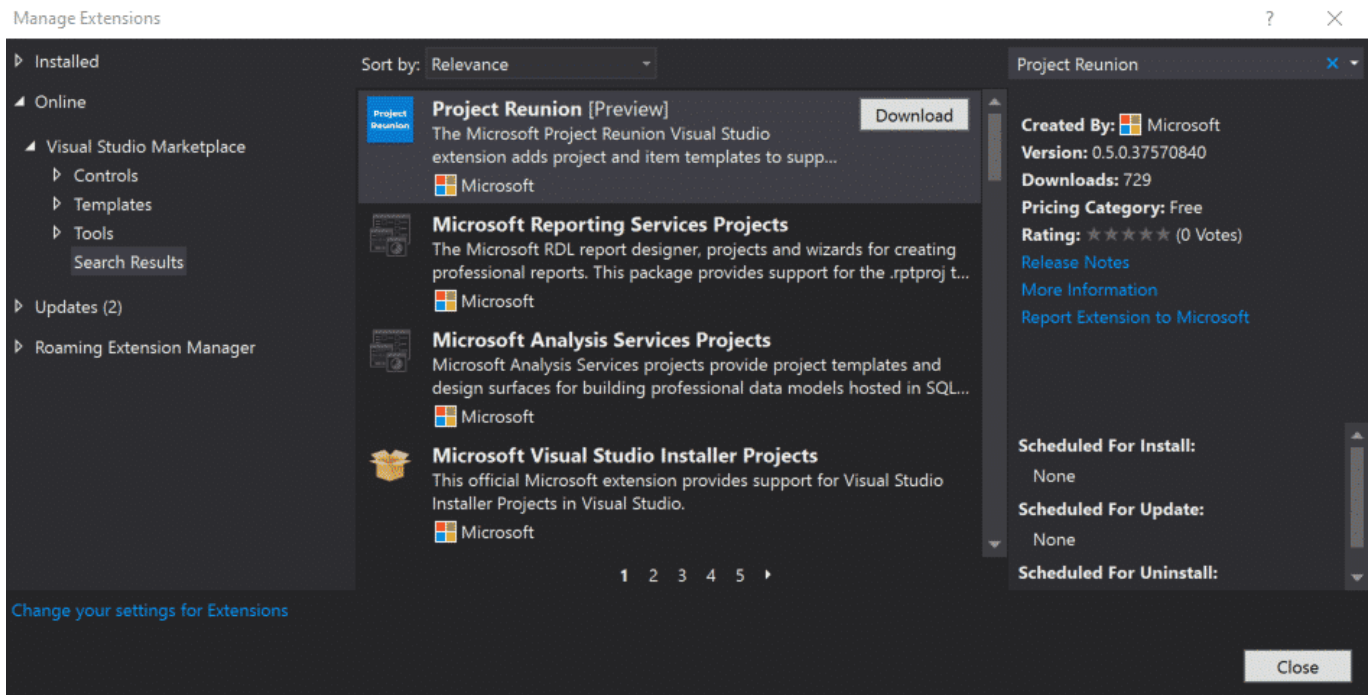
Project Reunion : quelles avancées pour Microsoft ?

Project Reunion a-t-il atteint un premier stade de maturité ? Certains de ses composants sont en tout cas désormais suffisamment stables pour un usage en production, a décrété Microsoft à l'occasion de la sortie de la [version 0.5](#).

Voilà près d'un an que le groupe américain a [officialisé](#) cette initiative. Son objectif : unifier l'accès aux fonctionnalités de la plate-forme Windows pour les développeurs. Son incarnation : un substrat qui regroupe des API découplées du système d'exploitation.



Ce substrat parvient aux utilisateurs finaux par l'intermédiaire du Windows Store. Il prend en charge toutes les versions de Windows 10 depuis la 1809 (October 2018 Update). Pour développer des applications qui y font appel, on installe une extension dans Visual Studio*.

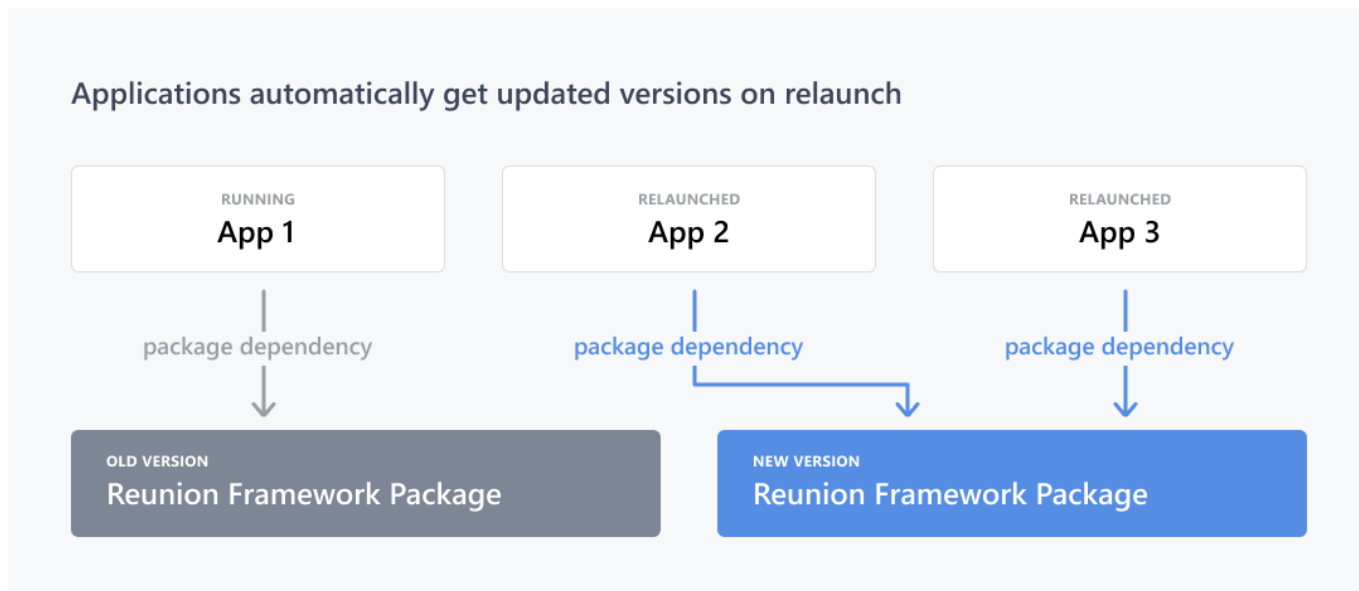


Pour le moment, Project Reunion regroupe trois briques stables : WinUI 3, MRT Core (version allégée du gestionnaire de ressources Windows) et DWriteCore (implémentation de DirectWrite pour le rendu de texte).

Deux cibles pour ces différentes briques : les applications de bureau (C#/.NET 5, C++/WinRT) et celles « de plate-forme universelle » (UWP). En l'état, Project Reunion n'est utilisable en prod qu'avec les premières, à condition qu'elles soient empaquetées en MSIX. Pour les secondes, Microsoft prie les développeurs de rester sur la *preview*.

Exploiter Project Reunion requiert au minimum Visual Studio 2019 16.9. On utilisera si possible la préversion de Visual Studio 16.10 pour pouvoir exploiter pleinement WinUI (par exemple, le rechargement à chaud).

Toutes les ressources du *framework* s'exécutent dans le contexte des applications qui y font appel. Elles ont l'avantage de pouvoir être mises à jour plus rapidement qu'en suivant le cycle de vie de Windows.



Project Reunion : deux feuilles de route

Parmi les fonctionnalités stabilisées de la v0.5, on aura noté, outre MRT Core :

- La prise en charge d'Arm64
- Le glisser-déplacer vers et depuis les applications
- Le support des pointeurs et des barres de titre personnalisés
- Des améliorations pour WebView2 (HiDPI, redimensionnement et déplacement de fenêtres...)
- Des mises à jour pour TabView (gestion des onglets) et le mode sombre

Comme la prise en charge des UWP, le multifenêtrage pour les applications de bureau reste en aperçu. La prise en charge des applications de bureau non empaquetées n'est pas encore effective. Même chose pour les principales méthodes de saisie manuelle (InkCanvas, InkPresenter, Handwriting View) ou encore les îles XAML (destinées à intégrer des composants UWP dans les applications de bureau).

La stabilisation du multifenêtrage sur les applications MSIX ne devrait pas intervenir avant la v1, prévue pour le 4^e trimestre. Entretemps, avec la v0.8, on devrait pouvoir expérimenter des fonctionnalités de gestion du cycle de vie (activation, instanciation multiple, redémarrage et récupération...).



La feuille de route n'est pas la même pour les applications non empaquetées (WPF, WinForms, Win32, ligne de commande).



* Project Reunion peut aussi s'intégrer dans des projets existants, par l'intermédiaire d'un paquet NuGet. Il y a toutefois des limites à cette approche. Dont l'impossibilité d'utiliser WinUI 3 et l'incompatibilité des projets

WPF.

Illustration principale © Thomas Hawk via Visualhunt