

Top Reasons to Use Desktop Containers and Application Streaming

Desktop Containers and Application Streaming make it easy to isolate applications and deliver them to any device, anywhere, at any time.

1. Integrate your applications into the user's workflow. Let's face it, if you want your users to be productive, you need to adapt to the way they work. With Desktop Containers and OpenText™ Application Streaming you can deliver applications in a way that feels natural to your users. Use your existing software deployment tools to put applications where they expect them, deliver them through a web portal so they can be accessed from anywhere, integrate them into your favorite learning management system or share them via hyperlinks. OpenText™ Desktop Containers makes it easy to adapt application delivery to the way your users work.

2. Deliver applications to any device, anywhere at any time. In today's work from anywhere world, it is critical that users can access their applications and data, in a secure way, whether they are in the office, at home on their own PC, using a mobile device, or even on a simple web kiosk device. The combination of Desktop Containers and Application Streaming allows you to deliver applications to all these environments, ensuring users can work from anywhere.

3. Ensure users can securely access their data from the application. Without access to data, most applications are useless. Desktop Containers and Application Streaming includes features that help you deliver the data, securely. Use inbuilt storage controls to provide seamless access to corporate cloud storage via the T: drive, provide access to network shares when streaming and prevent opening or saving data from local drives when streaming.

4. Improve user satisfaction with integrated profile management. As users move from one place to another and one device to another there may be better ways to access the application. For instance, when in the office on a corporate PC, it is typically better to take advantage of the compute resources to run the application. However, when the same user goes home and is using their favorite Bring your own Device (BYOD) you may want to stream the app since you have no control of the local PC. It's critical that as the user moves from one place to the next that the application settings and configuration follow them. Desktop Containers features in-built application profile management that synchronizes application configuration and lets it follow the user for improved user satisfaction and higher effectiveness.

Allow your users to access their applications and data from anywhere on:

- Corporate owned-Windows devices
- Personal owned-Windows devices
- Mac OS devices
- iOS devices
- Android devices
- Any device with an HTML5 browser

- 5. Simplify access to internal resources.** When users are working from anywhere it can get confusing for them to access their resources. With the built-in ability to have the Desktop Container Server act as a secure tunnel, you can reduce the complexity of accessing corporate network resources without introducing security concerns. In many cases this eliminates the need for users starting a VPN session before they can be productive.
- 6. Track, expire and limit application access.** When you use Desktop Containers to deliver applications you can track where the applications are being executed and by whom, enforce basic licensing such as how many users or devices can be running or can have access to the application, and you can expire applications to provide applications for only a limited amount of time.
- 7. Require enhanced authentication.** Because applications are often sensitive and provide access to sensitive data, you may want to require enhanced, multi-factor authentication to access the applications you make available. With the included Advanced Authentication Limited Edition you can implement multifactor authentication right out of the box or you can integrate with Azure AD, ADF, the full version of Advanced Authentication, or other SAML v2 IDPs to ensure secure authentication
- 8. Simplify the software packaging process.** Software packaging is now easier than ever. With a library of pre-containerized applications and new installation monitoring capabilities you can quickly build and customize applications. Easily mix and match components such as browser versions, plugins, and then add in your own customization to deliver exactly the experience your users need. For advanced users you can even automate the packaging process so that as new application versions are released they are automatically converted to containers.
- 9. Extend the life of your applications.** Browsers and client operating systems are changing faster than ever, and often the data center and enterprise applications struggle to keep up. If you've got applications that require older browsers, unsupported plugins like Java or Flash, or even applications that don't seem to play nice on Windows 10 then Desktop Containers might be able to help you extend the life of those applications, reduce the need for expensive retraining and data center upgrades, and keep users productive and with the ability to restrict network access of those applications, you can protect the application from known exploits.
- 10. Eliminate application conflicts.** Because users are no longer using just the devices under corporate control, they are more likely than ever to run into application conflicts. Whether it's a student with a different version of Office than the one they are learning, applications that require different applications of the same browser, or conflicting versions of line of business applications, Desktop Containers creates isolated application environments for each application that allow you to run both applications, usually at the same time, letting the user get what they need done, when they need to, without having to worry about trying to resolve these conflicts.

Learn more at

www.microfocus.com/en-us/products/desktop-containers

www.microfocus.com/opentext

Connect with Us

[OpenText CEO Mark Barrenechea's blog](#)

