

Move to the Planet of the APIs: Get the Most Out of Your Data Management Platform

Ask your current vendors if the products you're using hit these critical standards:



One product handles end-to-end data management, including storage, backup, back-end database/catalog, replication, and archiving.

One platform diminishes learning curves, reduces manual effort, and minimizes the opportunities for manual error.



All virtual and on-premises data is backed up without plug-ins, scripts, or work-arounds.

Using a patchwork of products for backups means that some data can slip through the cracks. Relying on vendor plug-ins, work-arounds, or product-specific scripts decreases the reliability of backups not only when your environment evolves, but also when you apply updates to the software you're using.



Backup data is stored in an immutable (unchangeable and thus secure) format.

If you're storing your data in a mutable format, or on the same system where the majority of your data lives, it's vulnerable to exactly the same attacks as your underlying data.



Critical data can be reliably validated nightly.

If you're performing periodic tableOvtop or smoke tests, you have no guarantee that critical backup data is valid. In the event of a breach, this could translate into having to revert to older, stale, or even incomplete backup data.



All backup data, whether on-premises or in the cloud, is globally searchable from your workstation.

The longer it takes to find the data you need to restore, the more productivity your company and users lose.

Move to the Planet of the APIs: Getting the Most Out of Your Data Management Platform

Ask your current vendors if the products you're using hit these critical standards:



A declarative policy engine that allows administrators to create service level agreement policies to define data protection frequency, retention, and archiving with no need to write any code.

Relying on manual intervention to create policies not only slows implementation, but also increases the possibility of errors. Using product-specific scripts leaves you vulnerable to script failure when your environment evolves or you apply updates to the underlying product.



Use the language of your choice to automate repetitive data management processes or incorporate data management functionality into your internal and external tools.

If you're using a product-specific scripting language for automation, you have to learn that language, and it's only good for automating the associated product. If you're using multiple products and each has its own scripting language, you're learning multiple scripting languages that can only result in a virtual duct tape suite of scripts. Any solution based on product-specific scripts is subject to failure when your environment evolves or you update the underlying software.



An API-first architecture compliant with the Open API Initiative (formerly known as Swagger), allowing the automation of any function available through the graphical user interface (GUI).

When your product exposes its full functionality through a language-agnostic API or set of APIs, you can use the language of your choice to automate data management into your company's processes as well as any internal or external tools that your company uses.