

VELOSTRATA

Four Reasons You Should
Think Twice About Using
Agent-Based Migration Software

WHITE PAPER

Introduction

Cloud migration is becoming a key discussion topic among companies as they race to take advantage of the benefits offered by the public cloud. Whether it's migrating apps entirely to the public cloud or creating a hybrid cloud that combines on-premises systems with public cloud, enterprises are looking for solutions that can accelerate their migration – without additional risk or complexity.

The challenges with agents and replication

Despite the need for powerful cloud migration solutions, almost all the solutions on the market today are simply re-purposed disaster recovery tools. This means these tools were not designed or built for cloud migration and suffer from two major architectural drawbacks: agents and replication.

When we refer to agents, we're talking about an application that is installed on the source server or VM, the destination VMs, or both. In some cases, the agent might also be referred to as a 'client' (or even something else) but the intent remains the same: software that must be installed on source and/or destination VMs in order to power a cloud migration. Within the context of challenges companies will face, it does not matter if agents are installed manually or automatically (by the tool itself).

When we refer to replication, we're describing the strategy these tools take to migrate data to the cloud. In the case of replication, you'll be copying an exact replica of an entire data set to the cloud. Disaster events, by definition, aren't planned, so it was OK for a disaster recovery tool to slowly migrate full data sets to the cloud (where they would sit idly until needed). These data transfers were slow and hard to project, but this was OK for DR.

For cloud migration, however, it causes some major challenges because these are planned events with real deadlines. In some cases, companies are looking to end data center contracts or avoid software renewals by hitting a certain deadline with their cloud migration. With replication agents, they must wait for all data to be transferred into the cloud before testing can even begin. This has often led to lengthy wait times and resulted in project delays and missed deadlines.

Long story short, these re-purposed DR tools can sometimes get a cloud migration done, but there are some potential downsides to relying on a sub-optimal technology which we'll explore in this white paper.

Deployment

Agents are typically small pieces of software and organizations can usually deploy them using remote execution or software distribution tools. This is still burdensome on the cloud project (and IT), however, because one way or another it represents another group of steps that must be completed before any migration can take place. And this group of steps will require time to plan, time to execute, and potentially time to troubleshoot.

Security

In many cases with agents, IT must identify specific exceptions which must be made before the agents can work properly. For example, many agents will require specific ports on the VMs (or in firewalls) be opened to allow agent communication, and some require disabling anti-virus on systems altogether. The use of popular software packages such as Java or .NET in these agents may also increase a system's attack surface for cyber threats, especially if an outdated version is in use.

Testing

Testing is a crucial part of any cloud project, because enterprises need to pick the right cloud(s), evaluate the actual costs of running in the cloud, make sure applications work properly, fine tune performance, and essentially verify that applications in the cloud can properly serve their customers and end users. Unfortunately, agents do not make cloud testing easy or seamless.

For example, before you can test in-cloud performance, you'll have to wait for agents to replicate all source data into the cloud first. Essentially, you invest the duration of an entire data migration before you can even see if the cloud is a good fit. This becomes a huge drawback, because it takes up both time and energy, and can increase your labor costs for the project. And, in the event that a VM doesn't work properly in that cloud and you want to test another region or cloud vendor, you'll have to wait for another full data transfer.

Cutover

Once the workload's data set is transferred to the cloud, there is still the task of completing the migration. With agents, this will often involve the complex process of properly shutting down an application's services on the source system while keeping the agent running. This can add substantial delays to any migration project because it requires deep application expertise, and will often require IT getting additional people involved. Once the application is shut down properly, the system is turned off and then started up in the cloud. When the application is live in the cloud, steps will need to be taken to re-sync any final data changes between on-prem and the cloud to avoid data loss.

The potential impact on your migration

As we've outlined above, using re-purposed disaster recovery tools that rely on agents and replication can create some significant hurdles in a cloud migration. And while these tools can sometimes get the cloud migration done, there are often two major impacts on the project: labor and cost.

The simple fact is that it is easy for agents (and replication technologies) to end up accruing a lot of additional labor throughout the migration, which leads to a higher overall project cost. We've worked with a lot of customers who have attempted migrations with that type of solution before, and they shared with us some of the additional tasks that agents required, estimates of the resulting hidden labor costs associated with those tasks, and some additional risk(s) to their migration project that they considered as a result of that additional labor:

Task	Description	Hours per server	Additional Risk
Manual Application Assessment	With agents, you'll need to have a deep understanding of how each application you plan to migrate operates. This is due to the agent requiring specific shut down procedures before you can properly migrate it. Furthermore, for the final data sync, you'll need to properly shut down all running applications but keep the (migration) agent running. Typically, this will mean bringing your application team into the migration mix as well.	4 hours	High. Manual intervention introduces a lot of places for potential errors or data inconsistencies to take place, further delaying migration.
Agent install and configuration	Before you can tackle anything migration-related, you'll need to install (or deploy) the agent to the source and/or destination VMs and get them configured properly. This might also involve disabling A/V software and/or opening firewall ports prior to agent installation.	1 hour	Moderate. Security posture changes any time A/V is disabled and/or new agents installed that communicate between sites.
Pre-migration delta Sync	The agent will need to perform a full scan of the source VM's hard drive.	1 hour	High. Consistency errors are likely when dealing with large, complex file systems spread across multiple systems
Isolated replica configuration for testing	Once a replica has been built from the source VM, it must be isolated in the network to avoid conflicts so that testing can commence.	.5 hours	Moderate. Risk of accidentally interfering with existing domain name resolution and/or data.
Post-migration delta sync	Once the destination VM is running the application, another hard drive scan has to take place on the source and destination to properly capture the data delta and apply appropriate changes	1 hour	Moderate. Risk of not properly synchronizing all data between last transfer and newest changes.
Post-migration agent uninstalls	Once the migration is complete, uninstall the agent from the destination VM, re-enable A/V, close previously opened firewall ports.	1 hour	Moderate. Possible corruption of other entities, and/or troubleshooting that may occur re-enabling A/V.
Total		8.5 Hours	

Figure 1: Additional Labor Estimates with Agents

These were the most common hidden labor tasks when using a replication-based, agent-based migration tool that our customers identified. Now, to be clear, every migration project will require labor- that's just a fact of migration. What we've outlined above represents the hidden labor that you'll have to factor into your migration project in addition to the standard labor that would be required with any migration solution.

As outlined above, these additional tasks add, on average, a total of 8.5 additional hours per machine to any migration project. For every 1,000 machines that a company migrates to the cloud with this type of technology, they've added an additional 8,500

personnel hours to the project. With conservative estimates of \$65/hour for in-house IT staff or \$125/hour for consultants, that's a staggering \$552,500 to \$1,062,500 of added labor costs.

Agents all but guarantee the additional labor above, but there are even more labor costs that can be incurred. Since these re-purposed tools were built for DR, they might be missing capabilities that are essential to a seamless cloud migration. In the table below, we've outlined some additional issues that some of our customers had experienced, their likelihood of occurring, and their estimated additional time per server when it did occur during their migrations:

Task	Description	Estimated duration per server	Likelihood
Manual inventory, monitoring, scheduling, job management	Large-scale migration planning and execution more challenging due to other solutions having limited UI, alarms, events, progress info, ETA reporting, integrations, etc.	.5 hours	High
Pre-migration testing	No way to test (or iterate) quickly. The need to potentially repeat tests with different configuration changes means needing to cleanup and re-launch every time, and potentially uploading data changes each time as well.	1.5 hours	Moderate
Cut over	Manually coordinate monitoring, shutdown, and final replication and re-launch of the server. Even more challenging with multi-tier applications, and without tools like runbook automation.	1 hour	High
Cleanup	Need to cleanup interim AMIs and EBS snapshots after a migration is completed	.25 hours	High
Stateful Rollback	When something goes awry during the migration and a server must be returned on-premises, often with manual, complicated procedures	2 hours	Low
Error Handling	If troubleshooting is required with a tool that has limited UI and reporting it will require significant manual intervention	2 hours	Low

Figure 2: Additional Labor Estimates with Agents Due to Missing Capabilities

Error handling, stateful rollback, and pre-migration testing (for complex apps) won't occur for every single server, so we're taking likelihood into account when we do our final math. Given that, and based on our customer's experiences with those types of tools, we can estimate, on average, an additional 3.5 hours' worth of manual labor in addition to your standard migration project timeline. That means for every 1,000 machines an enterprise migrates to the cloud with this type of migration tool, they've added an additional 3,500 personnel hours to their migration project. Using the same hourly rates as before, that's \$227,500 to \$437,500 of hidden labor costs.

Avoiding agents, replication altogether

The numbers above show how labor costs can accrue rapidly with the wrong solution. Velostrata, unlike other tools on the market today, was designed with one goal in mind: to support cloud

migration, especially for enterprise applications which can't afford downtime or data loss. Velostrata was architected to avoid agents and replication altogether, which means that Velostrata is also able to sidestep the challenges we referred earlier and which we'll cover in more detail now.

Deployment

Velostrata migration and workload mobility software leverages a streaming-based approach, and does not require agents at all, resulting in faster, easier deployment, and faster time to cloud. Velostrata's goal is to migrate enterprise workloads in minutes, while providing high availability and optimized performance between your data center and the cloud.

To accomplish this, Velostrata relies on a virtual appliance on-prem and additional virtual appliances in your cloud(s). These virtual appliances are the backbone which efficiently stream data back and forth between on-prem and your cloud(s) using state of the art WAN optimizations.

For management, IT managers use a vCenter plug-in (something familiar to existing IT staff who use VMware) to begin cloud operations with just a few clicks, though customers can also use PowerShell modules or our REST API to integrate with 3rd party solutions as well.

These two pieces make Velostrata easy to deploy and to manage, and eliminate the burden of having to install, troubleshoot, and then uninstall agents across countless source and/or destination systems during your cloud migration.

Note: A full discussion of our architecture and deployment would be out of scope for this white paper, but if you'd like a more detailed look, please refer to another one of our white papers, [Understanding Velostrata Technology and How It Works](#).

Security

Since there are no agents, there are no added security risks: no connections are needed to the workload's network, no ports need to be opened, no holes made in the firewall, no disabling A/V software, and no outdated agent software that can be exploited by attackers.

Testing

Velostrata provides built-in pre-migration testing capabilities, whereby IT simply selects a VM, right-clicks, and selects a testing option. It's simple, and it also lets IT choose between two testing strategies: sequential testing or pipelining.

Sequential testing means creating a dedicated cloud instance to perform key tests like performance tuning and cost analysis. With Velostrata, IT does this using our test-clone capability, which creates a replica of the application and its data directly in the cloud, but while the live workload (and data) remain on-prem and untouched. This gives IT the power to run real tests in the cloud with actual data, but without interrupting production systems or end users at all. When you're done with testing, just delete the test clone with a few clicks.

The second approach to cloud testing, often referred to as pipelining, is where the production app begins running in the cloud while data remains synchronized on-prem. With the application running in the cloud, you can see exactly how it performs with actual end user loads and exactly how much it costs. Depending on those results, you can also use Velostrata to shift the size or type of that application's cloud instance to meet your performance or cost requirements. Plus, with Velostrata, there's always the option to rollback on-premises while maintaining all the data changes that have already taken place.

No matter which testing strategy you employ, Velostrata gets you testing in minutes and with just a few clicks. This helps shave a significant amount of time (and energy) off any cloud migration

project, and gives IT the capabilities they need to properly architect their ideal cloud from day one, which are the foundations needed for a successful cloud migration project.

Cutover

Our architecture was designed to keep on-premises data synchronized with the cloud, down to a 30-second RPO, so there is no data loss whatsoever. And, since we don't rely on agents, there is no complex application quiesce to perform before cutover. This makes cutover both simple and seamless. Simply use our built-in detach operation to sever the tie on-premises and let this workload (and its data) run solely in the cloud.

Velostrata keeps cloud migrations on track, under budget

As you can see, not all migration tools are created equally, and many suffer from non-ideal technologies like agents and replication, or end up with limited capabilities because they were designed for DR and then re-purposed for migration. As we showed, this can result in significant additional labor and cost to your migration project.

In fact, if you add up both of the previously cited calculations (hidden labor with agent usage/install + potential troubleshooting), Velostrata could save customers up to twelve hours, on average, of additional labor per server. For a 1,000 server migration, this would yield savings between \$780,000 and 1,500,000 depending on the hourly rate of your staff or consultants.

Velostrata, unlike re-purposed DR tools that use agents and replication, were designed from the ground up to support cloud migration at speed and scale. Velostrata relies on a technology that is agent-less, and which intelligently streams data to and from the cloud when needed. These crucial architectural differences make Velostrata capable of easily avoiding the challenges we've outlined in this white paper, and thus, reducing the time, labor, and complexity of a cloud migration.