GIGAOM

GigaOm Perspective Brief



Restoring Data Is Not Recovering the Application

A CTO's Framework for Cloud-Native Backup and Application Recovery

Commissioned by Dreeze



Whit Walters, Field CTO, GigaOm



The Recovery Question

If your most important cloud application were destroyed tonight, how long would it be until customers could use it again? Not how long until the data is restored. How long until the application works?

Most teams answer the first question and assume it answers the second. A cloud-native application is not 10 or 12 VMs. It can span hundreds, sometimes more than a thousand, resources across the 200-plus services AWS and Azure each offer: compute, databases, queues, functions, identity policies, keys, and network configuration. The data is one layer. The application is all of it, wired together in a specific order.

Your backup tool protects the data. Your recovery time objective assumes the rest comes back with it. That gap is where recovery plans fail.

What You Are Actually Facing



Your backup tools were designed for a different estate. Most cloud backup products were built by VM-era vendors or retrofitted from VM-era products. They restore data and a few key resources. Everything else is rebuilt by hand: IAM roles, secrets, event triggers, queue subscriptions, the security group nobody documented.



Your RTO is a number, not a test result. If recovery depends on engineers reassembling an application from memory and from infrastructure-as-code that has drifted from production, the stated RTO is an aspiration, or it measures how fast the data comes back rather than how fast the business does. Most organizations find out which one they have during an incident.



You cannot protect what you have not mapped. Tags are inconsistent, IaC is incomplete, and applications span accounts and teams. The resources that look minor are the ones missing when the application will not start.



The threat changed; the tools did not. Inexpensive AI models put continuous, autonomous probing of cloud environments within reach of attackers who would otherwise be nowhere near nation-state capability. The insider threat is now as likely to be an over-permissioned agent with access to thousands of resources as a disgruntled employee. Both raise the odds that you will need to recover an entire application into an environment you cannot fully trust.

FIVE QUESTIONS EVERY CTO SHOULD BE ASKING RIGHT NOW

Use these as a vendor scorecard: a solution that cannot answer each one for your actual estate does not belong on your shortlist.

1. For our 10 most critical cloud applications, can we list every resource each one needs to run?
2. When did we last recover one of them end to end, and how long did it actually take?
3. Do our backups include IAM, keys and secrets, queues, functions, and network configuration, or only data?
4. After an attack, can we recover into a clean account from copies the attacker could not reach?
5. Who rebuilds the application if the engineers who know its dependencies are unavailable?

What Your Shortlist Should Require

Here is what separates adequate from effective in this space. This framework is based on production deployments and GigaOm's analysis of the data security platform market.

1. **Discovery that does not depend on tags** *(visibility).* The system should find applications and their dependencies in the live environment, not only from tags or IaC. Have it map one of your applications and compare the result with what your team believes is there.
2. **Coverage of the whole application** *(completeness).* Data, compute, and databases are table stakes. Ask about IAM, keys and secrets, Kubernetes, messaging, serverless functions, network configuration, and the niche resource types that matter only when they are missing.
3. **An isolated, immutable vault you control** *(integrity).* Data, configuration, and the recovery plans themselves belong in an air-gapped, immutable vault in your own accounts and regions, with the ability to recover into a clean account.
4. **Orchestrated, ordered recovery** *(velocity).* Recovery should automatically bring the application back in dependency order, keys and credentials included, rather than handing engineers a restored dataset and a checklist.
5. **Recovery you can prove** *(assurance).* Testing should be routine enough to run on a schedule and produce evidence for auditors. Each test also shows where your IaC has drifted from what production actually needs.

The Architecture Decision

Organize around the application, not the resource.

The business does not care whether a snapshot succeeded; it cares whether the application works, whether that is order processing, a claims system, or a customer portal. Protection policies, recovery plans, and reporting should be defined per application.

Separate recovery from the blast radius.

Recovery copies and plans that live in the same accounts and under the same credentials as production are exposed to the same attacker and the same misbehaving agent. Isolation has to be architectural, not procedural. The same holds for infrastructure-as-code: if recovery depends on rebuilding from a repository and that repository has been poisoned, the attacker controls the recovery too.

Keep the data in your control.

In regulated sectors, a design that moves copies into vendor-managed locations creates a new compliance question. Preferred architectures keep the vault and all backup data in your accounts and regions.

Use AI to discover, determinism to recover.

Mapping hundreds of resource types across clouds is a problem AI now makes tractable. Executing a recovery is no place for improvisation. Expect AI in discovery and deterministic, reviewable plans in execution.

What Manual Recovery Actually Risks

The runbook is out of date.

Manual recovery assumes documentation and IaC reflect production. They rarely do, and the gaps surface at the worst time.

The knowledge is in a few heads.

The engineers who know the dependencies are the same people you need everywhere else during an incident.

Untested recovery is unproven recovery.

When testing is expensive, it happens once a year at best, and the RTO you report to the board goes unverified.

Half-running is not recovered.

An application restored without its IAM roles, keys, or queue subscriptions may start but not work. Finding the missing pieces happens live, under incident pressure. In a regulated business, a half-working application can be worse than one that is down.



Where Dreeze Fits

Dreeze is a new entrant, launching publicly in November 2026. This assessment rests on a vendor briefing, a product demonstration, and documentation review rather than a scored evaluation.

Dreeze is built around the application rather than just the data or individual resources, organized into four functions: know, protect, orchestrate, and validate. It runs both halves of the job on its own engines: a backup engine with the tiering, compression, deduplication, and retention policies buyers expect, and a recovery engine built on top of it, whereas competitors that rebuild applications from configuration depend on another vendor's backups. Discovery combines deterministic logic with AI to identify applications and their dependencies across AWS and Azure without relying on tags or IaC alone, and presents them as dependency maps teams can review and edit. Backup plans are built per application, with data-sensitivity matching and cost estimates.

Coverage extends well past data: not only EC2, EBS, RDS, and VPC configuration, but also Kubernetes, IAM, keys and secrets, messaging queues, Lambda, and less common resource types. Data, configuration, and Dreeze's own recovery plans are held in an isolated, immutable, air-gapped vault in the customer's accounts and regions, never in Dreeze-managed locations, supporting recovery into a clean account after an attack.

Recovery runs as an ordered plan derived from the dependency map, including key handling, with custom steps for hybrid components. The same orchestration runs scheduled recovery tests in controlled environments, producing compliance evidence and exposing IaC drift.

The Gap Is Widening

Every service your teams adopt adds resource types your backup tool may not understand, and every agent deployed with broad permissions widens what can break at once. Cloud estates are getting more complex, and attacks on them are getting cheaper to run. Recovery designed for VMs will not catch up on its own.

Analyst's Take

Dreeze is aimed at the right gap. The backup market has spent a decade perfecting data protection, and in cloud-native estates that is no longer the hard part. Getting the application back is. Dreeze's application-first design, customer-controlled vault, and testable recovery address that directly.

It is also new. Dreeze is emerging from stealth with early design partners but no public customer track record, so its claims need validation in your environment. Two tests matter most: run its discovery against an application you know well and see what it finds that your team missed, then run a full recovery into a clean account and time it.

Start with your most critical application, not your whole estate. If the application comes back working, inside your stated RTO, you will have learned something your current backup tool cannot tell you.

Report Methodology

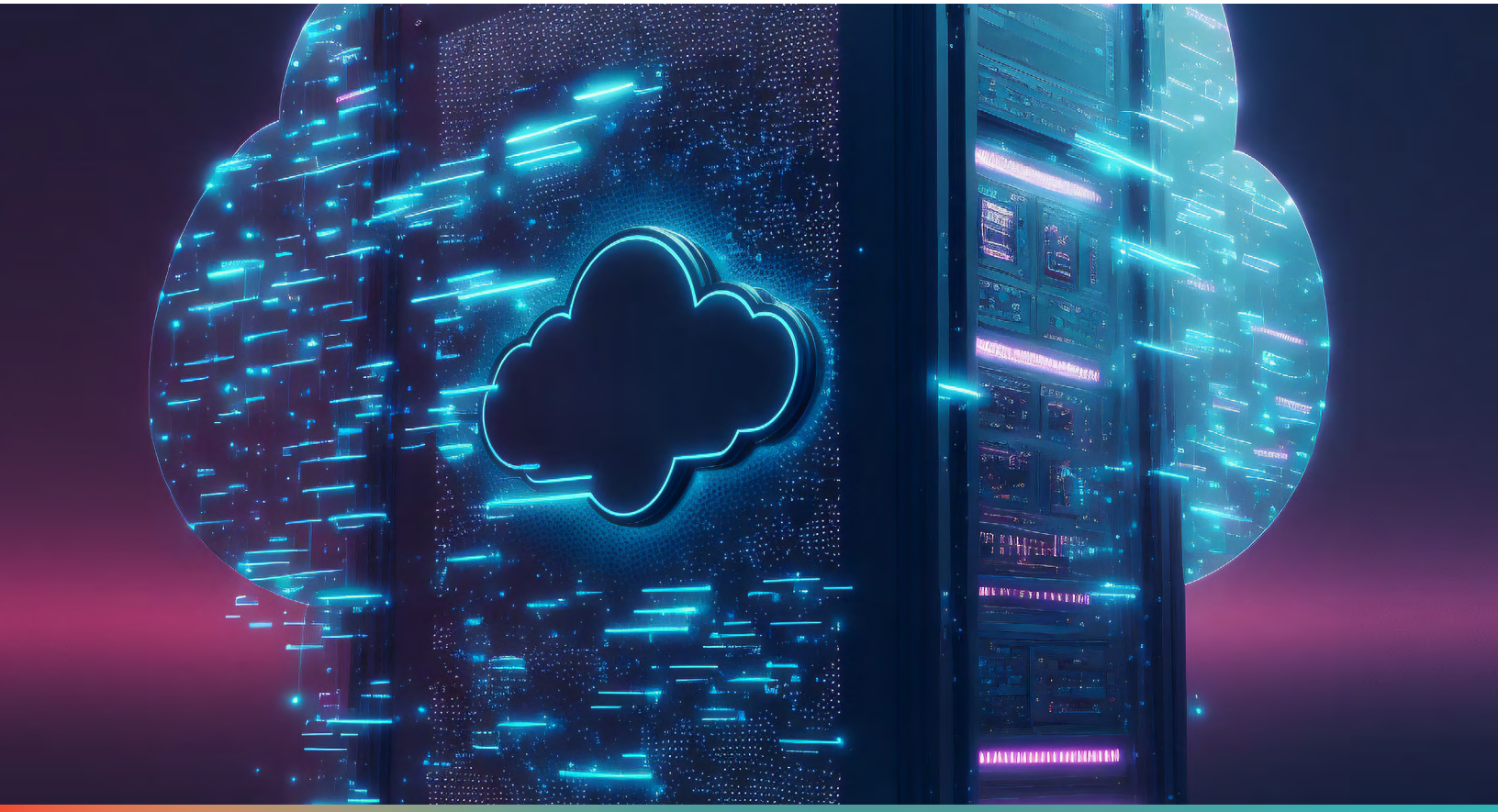
THE GIGAOM PERSPECTIVE BRIEF is an independent analyst framework for evaluating an emerging technology category, developed through vendor briefing, product demonstration, documentation review, and GigaOm's ongoing research and market coverage. It leads with the problem and evaluation criteria before the featured solution, so it stands on its own for decision-makers regardless of vendor selection.



ABOUT THE AUTHOR

Whit Walters is a Field CTO at GigaOm with 30 years of enterprise technology experience, including CTO roles at multiple startups, leadership positions within the Google Cloud partner ecosystem, and domain expertise in AI, machine learning, data architecture, and cloud security.

Copyright© Knowingly, Inc. 2026 "*GigaOm Perspective Brief: Restoring Data Is Not Recovering the Application*" is a trademark of Knowingly, Inc. For permission to reproduce this report, please contact sales@gigaom.com.



GIGAOM

GigaOm provides technical, operational, and business advice for IT's strategic digital enterprise and business initiatives. Enterprise business leaders, CIOs, and technology organizations partner with GigaOm for practical, actionable, strategic, and visionary advice for modernizing and transforming their business. GigaOm's advice empowers enterprises to successfully compete in an increasingly complicated business atmosphere that requires a solid understanding of constantly changing customer demands.

GigaOm works directly with enterprises both inside and outside of the IT organization to apply proven research and methodologies designed to avoid pitfalls and roadblocks while balancing risk and innovation. Research methodologies include but are not limited to adoption and benchmarking surveys, use cases, interviews, ROI/TCO, market landscapes, strategic trends, and technical benchmarks. Our analysts possess 20+ years of experience advising a spectrum of clients from early adopters to mainstream enterprises.

GigaOm's perspective is that of the unbiased enterprise practitioner. Through this perspective, GigaOm connects with engaged and loyal subscribers on a deep and meaningful level.