Cost Savings and Business Benefits of Decentralized Cloud Storage

Taloflow Research

October 2021

In this paper, we evaluate the TCO, ROI, NPV, and payback period of digital-native companies switching from hyperscalers like AWS, Microsoft Azure, and Google Cloud to a decentralized cloud storage model.

Notice

Taloflow provides insights and tools to help engineering leaders make smarter and faster cloud infrastructure buying decisions. For more information, please visit www.taloflow.ai.

Disclosures

This particular study by Taloflow Inc. ("Taloflow") was performed with substantial input from Storj Labs, Inc. ("Storj"). Storj is a decentralized cloud storage provider and also a client of Taloflow's cost evaluation services.

Taloflow maintains editorial control over the study and its findings and does not accept changes that contradict Taloflow's findings or obscure the meaning of the study.

This analysis is not suitable for use as a competitive analysis for investment purposes. Taloflow strongly advises that readers use their estimates within the framework provided in the report to determine the appropriateness of an investment in or selection of Storj as a cloud storage provider or the STORJ token. Finally, Taloflow makes no assumptions about the potential Return on Investment that other organizations will receive when using decentralized services.

Table of Contents

Executive Summary	Page 1
Key Findings	Page 2
I. Methodology and Process	Page 5
II. Background	Page 6
III. Significant Conclusions	Page 7
IV. Analysis	Page 10
V. Results	Page 31
VI. Recap	Page 35

Executive Summary

Our analysis shows that decentralized cloud object storage may offer quantitative and qualitative advantages over centralized or distributed object storage.

Decentralized cloud object storage couples the inherent benefits of a tokenized incentive economy with decentralized storage technology. The result is a globally distributed and trustless architecture for cloud storage designed to be more efficient, private, and secure.

Storj, a company that offers a decentralized cloud storage solution called Storj DCS, requested that Taloflow explore the potential Return on Investment (ROI), the Total Cost of Ownership (TCO), Net Present Value (NPV), and Payback Period associated with switching from a centralized cloud storage provider, such as Amazon S3, Google Cloud Storage, and Microsoft Azure Blob Storage, to a decentralized model such as Storj DCS. The purpose of this study is to provide the reader with a framework to evaluate the potential costs and benefits of switching cloud storage providers with an emphasis on business and bottom-line decision-making rather than the technical differences between the platforms. This analysis focuses explicitly on switching to Storj DCS from one of the "Big 3" cloud providers, i.e., AWS, Google Cloud, and Microsoft Azure. This paper will not cover the differences in impact when switching to one of the centralized but low-cost "storage-only" alternatives that have recently emerged in the cloud storage market, such as Backblaze B2 and Wasabi.

This paper will cover decentralized storage in general. While Storj DCS is the leading option, some of the same generalizations about the potential benefits of decentralized storage may also fit any other decentralized providers, like Sia and Filecoin, subject to their service limitations. We note that no one provider is fully decentralized. That said, the expressed goal of the decentralized providers is to work towards a fully decentralized system in the long term.

There are many unique considerations should that should be taken into account when deciding to switch storage providers. We invite you to message our team at Taloflow (email: team@taloflow.ai) for more detail on the reports and summaries provided herein or help with your decision-making around cloud storage.

Key Findings

Top Quantified Benefits

Saving on usage fees

Depending on the overall volume of storage and access patterns, potential usage costs can be up to 70% lower with decentralized storage due to lower prices for egress, storing data, and READ and WRITE operations. However, use cases requiring heavy analysis of data tend to see little to no savings.

Native geo-redundancy

Decentralized architecture provides global redundancy as a core feature. Achieving the same level of redundancy with centralized storage is more expensive and time intensive.

Top Non-Quantified Benefits

There are other benefits to be considered that cannot be quantified using a simple formula. These are more anecdotal benefits than anything else.

Branding

Decentralized storage eliminates the risk of any party “peering in” to data that they shouldn’t have access to, offering some brands a point of differentiation.

Higher employee satisfaction

Some companies that have made the switch to a decentralized model feel that they are no longer locked in with one of the Big 3 providers and on the cutting edge of technology.

Simpler billing

Billing reports from decentralized providers tend to be more straightforward.

Reduced points of failure

By definition, decentralized storage has so many nodes that there are few paths to system failure.

Use cases that may benefit <u>the most</u> by switching to decentralized storage

HEAVY GRAPHICS SAAS

Use cases with long-term storage of images and infrequent downloads see improvements with decentralized storage due to competitive pricing and CDNs potentially being made unnecessary.

Example: Storing lot of thumbnails like Hootsuite

DATA SCIENCE WITH LOW STORAGE INTERACTION

This use case can see decreased costs while maintaining performance because of low storage costs and the ability to load data in parallel.

Example: Analyzing transaction data like Opendoor

NO GRAPHICS SAAS

This SaaS use case offers a text-heavy (e.g., serving documents and data) to its end users with storage either used as a data store, archive, or house state data. This use case always has databases to house customer or service data and is not likely to be multi-region.

Example: Workflow and project management tools like Jira

CUSTOMER-FACING AND CONTENT-DRIVEN

Application data tend to be database-driven, making engineering costs of switching high. Only in scenarios where storage volume is high is this switching justified.

Example: Serving media like Netflix, Reddit and Facebook

REAL-TIME DATA WORKFLOW

Decentralized storage costs make storing and then moving data more cost-effective. However, use cases relying on ETL or tiny file sizes do not see improvements with decentralized storage.

Example: Event stream writing out to storage like NS1

Use cases that may benefit the least by switching to decentralized storage

DATA SCIENCE WITH HIGH STORAGE INTERACTION

Because of high fees for data transfer, decentralized storage providers may have higher costs.

Example: Analyzing data with tools like Redshift or Bigquery

DEV-HEAVY SOFTWARE DEVELOPMENT

Use cases with low storage utilization will see lower savings after switching but may fit when considering decentralized storage at the beginning of a project.

Example: Long-running R&D Projects / Skunkworks

I. Methodology and Process

About Taloflow

Taloflow is in the business of reducing the friction associated with making buying decisions for cloud services and developer tools. Providing an objective analysis is part of what we do. Whether that friction results from the complexity of the decision, the lack of information, or the magnitude of the hours needed to evaluate the situation, Taloflow has accumulated an extensive database of use cases, derived insights, and sophisticated methods to assist in a wide range of stack decisions. Taloflow is your partner to navigate the increasing depth and diversity of many segments, including cloud object storage.

Our Methodology and Process

The goal of Taloflow's proprietary method of structured comparison is to provide a framework for weighing an organization's objectives in a manner that can handle both quantitative and qualitative concerns. This creates a decision-making process that is transparent, objective, and efficient to all its constituents. As part of this process, Taloflow often calculates the traditional financial measures of Return on Investment (ROI), Total Cost of Ownership (TCO), Net Present Value (NPV), and Payback Period, but does so within the broadened context of four main elements:

- The quantified direct costs/benefits associated with using a particular product
- The unquantified direct costs/benefits associated with using a particular product
- The quantified indirect costs/benefits related to the organization as a whole
- The non-quantified impacts on the organization

Taloflow generally follows the process outlined below to derive the contexts and create the financial measures:

Use Case Analysis	→	We use the available data and industry experts to map potential use case groupings for the analysis.
Customer and Vendor Interviews	→	We have extensive discussions with customers and vendors, sometimes as "secret shoppers."
Benefits Mapping	→	We establish an initial set of potential benefits and a requirements document based on our customer interviews.
Financial Modeling	→	We prepare sophisticated financial models that use all available data.
Reporting	→	We prepare reports for specific decision-makers to have practical tools to communicate the decision process to the organization.

II. Background

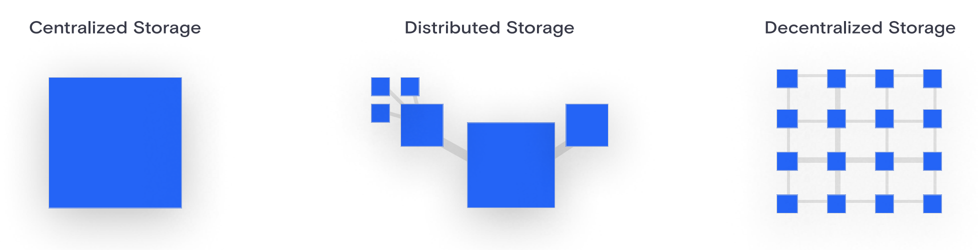
Over the next decade, it seems clear that we'll see cloud spending (IaaS and PaaS) continue to grow at a rapid clip thanks to the promise of immediately available and scalable infrastructure and the benefits of being able to run your business dynamically at the margin rather than with the long lead times and planning that characterize the history of computing. The hyperscalers, AWS, Google Cloud Platform (GCP), and Microsoft Azure are remarkable at driving adoption. Still, developers, DevOps, SysAdmins, and entire product teams realize limitations and even risks with the fixed, complex menu of cloud services they offer.

For example, there's limited price competition and business model innovation regarding their cloud storage offerings. There have been some tweaks around the edges of infrequent and archival storage offerings but no major price movements or changes in pricing schemes. AWS's cloud storage offering, Amazon S3, is rumored to be one of its most profitable products. If that is the case, that's saying a lot when AWS's operating margins can exceed 30%(1). There hasn't been a price reduction since 2014, either (save for archive tiers like Glacier). For Azure and GCP, things aren't much different.

Challengers have begun to offer technology and business model innovations that can provide considerable value over traditional hyperscaler cloud storage providers. The trifecta picking up the most steam in the cloud storage category comprises Backblaze, Wasabi, and Storj. However, the latter has a unique distinction with significant implications: decentralization.

Big 3 vs. Decentralized

Centralized vs. Distributed vs. Decentralized Storage



The Big 3 provide distributed cloud storage by spreading data to several locations within a region and by offering, at a price, the ability to replicate data across multiple regions. The technology overlay (security, maintenance, logging, billing) can be centralized or distributed. Governance, however, is strictly centralized. All decisions made as to implementation are corporate.

1. Bishop, T. (2021, February 3). Amazon web services posts Record $13.5B in *PROFITS* for 2020 in Andy JASSY'S AWS swan song. GeekWire.

Data is housed in facilities owned by the cloud service provider, with the software (and its maintenance) managed by that provider.

Decentralized storage (in the abstract) consists of data repositories constructed from files divided up and distributed across thousands of geographically diverse Storage Nodes. No single party - except the owner - has access to either coherent (complete and usable objects) or unencrypted data. At the extreme of decentralization, the software that manages this process is open source and hosted anywhere, relying on communication between Satellites and the Storage Nodes to manage the system. In practice, some centralized commercial enterprises are responsible for running the system, which keeps track of storage nodes and makes them functional as a storage complex, although the providers' expressed goal is to decentralize this role eventually.

> We consider Storj distributed for governance and decentralized for storage, whereas the Big 3 are centralized for governance and distributed for storage.

Who is Decentralized?

The Decentralized movement has a thriving community with Storj, Sia, Filecoin, and Filebase (reseller of decentralized storage), capturing increasing developer and product team mindshare. While Sia and Filecoin have fascinating offerings and promise of scale, the early state of these two projects makes anything close to an apples-to-apples (or close to) comparison to hyperscaler alternatives (like AWS, Microsoft Azure, and Google Cloud Platform) complex. However, Storj was the first to launch a decentralized cloud storage solution back in 2016 and already has a thriving community with well over 13,000 node operators supplying capacity for their decentralized service. Their success will be an essential bell-weather for Sia and Filecoin as well.

III. Significant Conclusions

Quantified Cost Benefits

Many of the use case scenarios we ran showed a significant quantified cost-benefit from using a decentralized storage provider over any Big 3. The basis for the cost-benefits follows.

Lower usage fees

In the selected use cases, the lower usage fees typically resulted in a total savings of approximately 70%. The main drivers are a lower cost of storing data, lower egress fees, and generally lower costs per READ and WRITE functions. If you isolate the storage fees from any data transfer fees or quality of service issues, we found no use case in which the storage fees for the decentralized providers would be more than that of the Big 3 providers.

Multi-region benefits can be significant

For example, if the use case requires a multi-regional approach, for things like data redundancy or lower latency, decentralized storage. This approach provides automatic global redundancy included in the storage/egress cost, whereas most centralized offerings require you to duplicate expenses - and administrative hours - to obtain true "geo-redundancy." As a result, automated redundancy savings - on storage costs alone - were often more than 60%. However, this did not include any data transfer costs or services like Amazon S3 Transfer Acceleration.

Concurrent downloads

It is possible to obtain parallel downloads with both centralized and decentralized services. Decentralized storage uses multipart downloads as a core, built-in part of its service since it reconstructs many parts of the file stored on many different nodes. As a consequence, it provides a level of concurrency as a matter of course. The Big 3 providers also offer concurrency. However, the difference is that it is not entirely automatic, and it can require some engineering resources. It's also not a geo-redundant concurrency as it is generally within a single region. There are usually no extra charges associated with the concurrency on decentralized storage, while additional transactional and potentially data transfer-related charges occur with concurrent downloads on the Big 3. For clients that want concurrent downloads from multiple regions, decentralized storage results in significant cost savings.

Non-Quantified Benefits

Clients referenced several non-quantified benefits from using decentralized storage. The following is a summary of the significant non-quantified benefits.

Employee satisfaction

Some companies noted that their developers do not want to be bound by the Big 3 providers. It made their employees feel more "cutting-edge," and they like the idea of decentralization as an approach.

Simple billing

Companies spend less time and effort going through billing data from decentralized storage providers because the billing dashboards and reports tend to be more straightforward.

Extensibility

How easy it is to add new features and have decentralized storage grow with your business. There are multiple scenarios in which a customer might choose to implement decentralized storage and later realize additional use cases and business opportunities.

Customer branding

There appears to be the potential for using decentralized storage as a branding proposition in that a company can make the claim it has no access to the data, that there is no centralized point of failure, and that no one can “peer in.”

Switching Costs

Incurred switching costs could significantly determine the ultimate TCO, ROI, NPV, or Payback Period for switching from the Big 3 providers to decentralized storage. The following is a summary of the significant switching costs.

Implementation and data migration

The cost of data migration can be high. Some providers, like Storj, offer extensive egress credits to reduce - and often eliminate - the cost of moving legacy data, which can be pretty substantial. The volume of data stored and the work required to filter, refine, or reorganize drive the migration costs.

Professional Services

Some companies, especially those with robust sets of legacy data, may require professional services (usually charged hourly) to get outside expertise or increase their development bandwidth.

Code changes

The cost to change code is dependent on system architecture. For example, most decentralized storage providers have an S3-compatible API which makes switching relatively easy. However, in some cases, this means giving up some of the benefits of decentralization as the API becomes a single entry point. In addition, to the extent that the processes are integrated heavily with the lower workings of S3 (e.g., using S3 events to trigger other events), more re-work may be necessary.

Architectural changes

In some cases, it may be necessary to rearchitect the storage layer to account for the subtle differences between the Big 3 provider storage mechanics and a decentralized storage provider. These would be use cases that incorporate prior decisions of localization, trust, or object hierarchy which might require modification in the decentralized environment. Implementation costs go up considerably if significant architectural changes are needed.

IV. Analysis

One of the critical components of our analysis is that the goals and objectives of an organization are part and parcel of the decision-making process. Typically, when working with our clients, we undergo lots of discussions and a detailed analysis of their historical billing to help determine what general use case they fit into as a first step. That informs the process of identifying the types of organizational objectives of companies.

The ROI, TCO, NPV, and Payback Period from one organization may not indicate the same measures in another, especially if their use cases vary widely. Therefore, the analysis presented here is for a frame of reference only.

We used a cloud cost database of over 30,000 cost and usage reports to create a set of use cases that represent how enterprises, startups, and other business structures use cloud storage. Our focus is on digital-native companies, meaning those founded after 2007 that rely almost entirely on cloud-based services for their technology stack.

Overlay Analysis

We took the dimensions developed through our research and mapped those against the requirements and the use cases. The tables in the following pages offer a generalized view of how the various offerings related to object storage interact with the needs and priorities of the use cases we covered.

Overlay analysis

Dimension	Big 3 / Distributed	Decentralized
Geo-redundancy	Multi-regional redundancy (a.k.a. “geo-redundancy”) is available for an extra cost. It can be implemented with one data center in each region or multiple data centers (zones) within multiple geographic regions.	Geo-redundancy is automatic all over the world
Latency	The latency to a specific region is very low, while geographic latency requires the cost of Acceleration-like products, CDN, or elastic and reduce type products.	The interregional latency is generally equal to or better than that of the Big 3 providers out-of-the-box. In addition, latency to a specific region is usually comparable to that of a Big 3 provider, if not slightly slower.
Redundancy	You generally get some intra-regional duplication but pay for inter-regional.	You get extensive, inter-regional redundancy for no additional charge.
Uptime/Accessibility	If the Big 3 provider goes down in your region, you can generally not access data. However, if you pay for redundancy, your data can apply a failover routing. There is excellent uptime for storage, and most issues in the past have been primarily linked to accessibility.	If you use legacy APIs, you might have an issue if this provider has uptime issues. However, you can access the network’s nodes directly at any time and would therefore have almost no downtime exposure. As a result, uptime for Storj is generally excellent, although it is a bit early to make a call here based on its limited history in the market.

Dimension	Big 3 / Distributed	Decentralized
Concurrency/ Multipart	Concurrency is possible using multipart requests. It can be significantly increased with engineering.	There is some level of concurrency out-of-the-box with no dev work. The maximum concurrency possible may require non-S3 compatible access to take full advantage of the platform. If you choose to download directly from node providers, you can also achieve substantial concurrency that way.
Throughput	Throughput can be very high. For example, Amazon S3 automatically scales to achieve at least 3,500 PUT/ COPY/POST/DELETE and 5,500 GET/HEAD requests per second per prefix in a bucket, with no limit on the number of prefixes.	Throughput can be more variable as it depends on the node traffic and the number of concurrent downloads.
Price	The Big 3 providers generally price higher than decentralized providers.	These offerings generally price lower than the Big 3 providers.
Ransomware	If the Big 3 providers are exposed to a ransomware attack that originates against the provider, that is a real risk. Implementing best practices would help avoid and manage attacks on company data.	It would be challenging to have a ransomware attack at the provider level since the nodes are distributed across the network. However, best practices also need to be employed to avoid or manage attacks on company data.

Overlay analysis

Dimension	Big 3 / Distributed	Decentralized
Capacity	Currently, there appears to be sufficient capacity within the Big 3 providers to handle almost any conceivable storage demand. However, how that manifests going forward remains to be seen.	The current decentralized providers rely on nodes that have available capacity. There is currently excess capacity in the systems, and they appear to be growing, but how that matches demand in the future has not yet been established.
Connectedness	The Big 3 providers are the broadest in the scope of service offerings, so naturally, there are fewer concerns about interconnectedness. You can also easily integrate bucket events into a workflow.	The more interconnected the plan is, the higher the start-up costs because the more work related to "connecting" things has to occur. This work can be costly on engineering resources, especially for non-available connectors. Integration lists and marketplaces are growing fast but are still relatively limited compared to the Big 3 providers. For example, there is no streamlined integration for bucket events triggering a workflow at present.
Data Business Risk	If the provider is no longer financially viable, you will either have to move your data somewhere else or not be able to access it. Centralization is also easier to subpoena or be made available to regulators.	In theory, the stored data is always there as there is no central point of failure. However, in practice, this is much less clear until the file manifests are also decentralized. Then, it's possible to set up a system and distribute keys in such a way to make data seizure not possible.

Overlay analysis

Dimension	Big 3 / Distributed	Decentralized
Integration	The Big 3 providers offer a comprehensive list of services, such as backup or CDN, and have a robust, well-used integration with their object storage.	Some integrations are available. However, the list is more limited than those of the Big 3 providers.
Security	There are broader threat surfaces between multiple regions with the Big 3. A very high level of security can be obtained, but it requires good policy maintenance, testing tools, and someone who knows what they are doing. With the right experts in tow, you can create very fine-tuned security in the Big 3 providers.	There are very narrow threat surfaces, and simplified key-based security can be very effective if best practices are maintained concerning the keys. However, there is still prominent exposure to developer error. Fine-grained levels of security nuances are possible but require customer wrapping of security within its framework. We believe this may be just as complicated as the Big 3 providers in the long run. There is also no role-based or IAM-based security out of the box.
Durability	Offers 11 9s of durability by using erasure encoding (e.g, Reed Solomon) to ensure data integrity.	Offers 11 9s of durability by using erasure encoding (e.g, Reed Solomon) to ensure data integrity.

Overlay analysis

Dimension	Big 3 / Distributed	Decentralized
Privacy	You are granting some level of visibility about your business activities to the Big 3 providers. This information can show up in logs and other areas if not carefully watched. It depends to some extent on developer implementation to ensure that no personal or company information is exposed through unencrypted data or exposed APIs.	All data is default encrypted with user-controlled access keys giving only users - and whom they designate - access to data. Since no single node has the entire picture (or even knows it's your data in the first place), there is no possible snooping visibility on any of your activities. It depends to some extent on developer implementation to ensure that no encryption keys are leaked or that no data is exposed through APIs.
Management	The Big 3 providers generally offer a robust web console. There are many automatic features like life-cycle management. However, billing management can be time-intensive, and there are not many tools to assist in paring down unused files, so some custom functions are usually necessary. A fully documented robust management API that can implement most if not all of the required management functions is available.	In general, the consoles of the decentralized providers are not as robust or well-tuned as those of the Big 3 providers. However, the command line access tends to be comparable. Similar to the Big 3, these vendors lack tooling to assist in paring down unused files. APIs are available for most providers for most of their functionality.

Overlay analysis

Dimension	Big 3 / Distributed	Decentralized
Learning Curve/ Training	Many developers are already familiar with most of the Big 3 provider offerings. So it's pretty easy to hire people who have completed dedicated training programs or have work experience with their services. However, nuances can be tricky, and intense deep-dives into the details of how data is stored and retrieved may require a "guru" of sorts to execute a low-cost, high throughput, or large file setup.	Decentralized services will be new to most developers. However, an S3 compatibility layer should reduce the costs and friction tied to the learning curve. It is unclear whether or not the developers will have to deep-dive into the nuances to get going. It may be less complex to address these nuances for some very low latency or high concurrency applications. The company will have to train up a "guru" or work with the provider directly as it will be tough to find one on the open market.
Use Case Suitability	Since the product offerings are so comprehensive, generally, you can get a product offering that is at least suitable for your use case, although it might not be the best provider for that use case. Also, if you have any compliance certificate requirements, it will usually be easier to meet using one of the Big 3 providers.	Some care needs to be taken at the onset as all use cases cannot have their requirements met at the current time with decentralized storage, but in some cases, it might be the best possible provider. It can also be more challenging to meet some compliance requirements with decentralized storage.

Attributes and Requirements

The use case cards below summarize the cluster groups related to object storage, including the main attributes or improvements considered in a company's decision to switch object storage providers. We mainly used anecdotal evidence to name and describe the mathematically derived groups. The next step involved us asking our clients and using our database to prioritize the objectives of each use case, including the level of pain associated with any issue that arose.

All stack decisions involve trade-offs. Thus, having a requirement framework that establishes priorities between the requirements and threshold requirements is very useful. The next step is to prepare a requirements analysis for each use case groupings, determining priority, relative weights, and potential ways of ranking or analyzing the various dimensions by using the information above.

HEAVY GRAPHICS SAAS

Use Case Description

This SaaS use case offers a heavy graphics-based service (e.g., catalogs, documents, image previews). We differentiated it from the other content providers because it generally does not have an upload or volume-based usage pattern but rather an API-driven service. The service needs to manipulate the materials rather than display them, implying a CDN, a database, and not much archival. Everything needs to be hot. It is almost always a multi-region setup with storage embedded into serverless technologies for scheduling and probably a sizable CDN bill to go alongside storage.

Attributes

- Ability to handle large file sizes efficiently
- Cheap long-term storage
- Multipart, or very parallel downloads/uploads

Requirements

- Efficient ample low-cost, and low-touch file storage are highly valued.
- Connection to transcoding is highly valued.
- Hot archival is moderately valued.

DATA SCIENCE WITH HIGH STORAGE INTERACTION

Use Case Description

This use case is a data analysis-heavy company with a SaaS-like implementation or machine learning (ML) integrated tightly into production. This use case stores what is most likely raw or data files that go through ETL and get read into an ML pipeline for analysis. It probably uses storage as a data store with well-organized files. This use case is somewhat likely to be using parquet or some other columnar structure.

Attributes

- Easiest Interconnectedness of data
- Wide range of interconnected services
- Cost-effective archiving solutions

Requirements

- Inexpensive data flow between application/analysis required.
- Fast data flow between applications/analysis is highly valued.

DATA SCIENCE WITH LOW STORAGE INTERACTION

Use Case Description

This use case is heavy on data analysis with a SaaS-like implementation or has a lot of offline and ad-hoc data science. If we suppose the storage service is a data store, this use case has a data lake or different database platform to house data for analysis with large and well-organized files (i.e., client or divisional representations of block data). There is some need for archiving older data, and this use case likely only requires running the platform in one region. The loads are presumably large when the storage service gets a data upload and needs multiple upload streams.

Attributes

- Lower storage cost
- Minimize the cost of pushing to 3rd party analytics
- Faster application load time from storage into processing instances.

Requirements

- Low-cost storage is highly valued.
- Concurrency on load for faster throughput is highly valued.
- The ability to connect to 3rd parties is moderately valued.

REAL-TIME DATA WORKFLOW

Use Case Description

This use case is for a process-oriented platform that receives a lot of real-time data and then either displays or analyzes the data. The real-time files tend to be in an event stream (e.g., workflow managed via a pipeline like Kafka with ETL) writing to storage and are usually small in size. There is some abstraction layer pushing to other data repositories as necessary, with storage being transactional.

Attributes

- Low-cost archival storage
- Large pipe capacity in
- Multi-Regional

Requirements

- Low-cost archival storage is highly valued.
- Inexpensive egress is required.
- Fast upload speeds are required.

DEV-HEAVY SOFTWARE DEVELOPMENT

Use Case Description

This use case pertains to companies producing code for 3rd parties or in a pure R&D context. Storage is an integral part of the development process, but the levels of stored data are moderate. Flexible access to data is a big deal to run tests or tests and stress-test the system.

Attributes

- Cost of storage
- Cost of connecting to 3rd Parties
- Cost of loading applications and tests
- Cost of long-term storage

Requirements

- Low-cost archival storage is highly valued.
- Project accountability or tagging are moderately valued.
- Role-based permissions are moderately valued.

CUSTOMER-FACING AND CONTENT-DRIVEN

Use Case Description

This use case is for the customer-facing sites that are content-driven with a mixture of text, graphics, and searches and no heavy upload and storage requirements. Most content is delivered "hot" (not much archival) and leverages a Content-Delivery Network (CDN) and a database to do so across multiple regions. This kind of storage use case is akin to having a deployment vehicle for static assets. If it is archival, it is for old history. In many cases, some logging data gets stored.

Attributes

- Low latency content delivery either directly or through CDN
- Moderate price sensitivity
- Ability to archive effectively
- A cost-effective way to house operational/logging data
- Better visibility on how storage costs relate to customers
- Better throughput in the system overall

Requirements

- Must be multi-regional
- Low latency of multi-regional content delivery is highly valued.
- Tracking product profitability with tagging is desired (nice-to-have).

NO GRAPHICS SAAS

Use Case Description

This SaaS use case offers a text-heavy (e.g., serving documents and data) to its end users with storage either used as a data store, archive, or house state data. This use case always has databases to house customer or service data and is not likely to be multi-region.

Attributes

- Cost of storage
- Low latency
- High availability and durability

Requirements

- Low-cost storage is highly valued.
- Low latency, multi-regional is highly valued.
- Trackability is moderately valued.

Quantified Costs

Using our pricing models and examples for each kind of use case from our database, we estimated quantifiable comparison costs for each use case between traditional distributed object storage vendors (like the "Big 3" providers AWS, Microsoft Azure, and Google Cloud Platform) and decentralized object storage vendors (like Storj).

Please note that we used the rate card prices as of July 1, 2021. The two main categories of costs we considered quantifiable are (1) Start-up/Switching Costs and (2) Maintenance Costs.

Additional start-up/switching costs

Use case-driven factors, such as the amount of data, complexity, interconnectedness, dispersion within the organization, and frequency, were all considered when determining start-up costs for each category. When determining ROI and other comparative measures, start-up costs apply to a new vendor and should not be included for the existing vendor when generating the comparison.

Data migration

Data migration involves the cost of moving data from one provider to another. Usually, this means a per-GB egress charge, as well as potential transaction costs. In addition, there may be associated processing, consulting, and other expenses. Data migration costs are primarily a function of the volume of data. However, in some cases, it is also influenced by the type of migration connections available by the provider. We estimated the data migration costs using the stated transfer rates and the size of the existing data.

For the data migration analysis, we used the heavily discounted transfer rates from AWS to Storj. We observed that each decentralized provider has a plan that provides discounts to the main distributed provider. We also applied for an egress credit which seems to be standard as well.

Training and learning curve

As with any new environment, there are costs associated with training developers and a general learning curve that reduces the team’s efficiency for some time. However, we did not find these significant for any use cases and estimated them at $0.00.

Code modifications

Because API calls and other storage-specific elements are often hardcoded into various platforms, it is sometimes necessary to modify existing source code to incorporate a new object storage provider. We found that some of the clients we evaluated did have code modification requirements while others did not. Specific modifications involved concurrency alterations and endpoint management. We estimated the cost of the modification time by gathering estimates of the labor time from customer interviews and our general knowledge. We then multiplied the number of hours of labor by the average labor rate for developers in San Francisco, California.

Environment duplication

Often, moving object storage involves duplicating the environment temporarily to ensure a clean turnover with minimal service interruptions. However, we did find that some of the use cases involved duplication of production environments. We estimated the duplication by assigning a typical monthly storage bill as a "duplicate" to some use cases.

Design and engineering time

In addition to time spent modifying code, some companies hire consultants to architect the overall system and point out potential data flow problems. We found this to be more organizationally driven than use case-driven. Therefore, we did not estimate these amounts.

Software and license

The move of storage providers can impact specific licenses (such as those purchased through AWS directly). However, we did not find these to be significant and therefore estimated them at $0.00.

Downtime and general disruption

Downtime and its incumbent opportunity costs are sometimes unavoidable. We estimated this for the use cases using a percentage of the revenue method wherever it was appropriate.

Ongoing costs

Ongoing costs are a function of price and quantity. Both of these have potential changes during the time it takes to make a decision. Therefore, our analysis used the price models we developed over the last several years, which considered both the rates and quantities associated with each storage provider and calculated the usage costs of the services and the related maintenance costs.

Primary quantity units for usage costs

1. Amount of data transfer — *Also called bandwidth, egress or download*
   - Size ranges of file transfers
   - Frequency of access
2. Amount of transactions — *Like the number of API requests (e.g., ListObject)*
3. Amount of stored data
4. Amount of archival
5. File size
6. Regions
   - Number of unique regions
   - Region distribution
7. Access patterns
   - Number of WRITES
   - Number of READS

Other considerations

- Storage pricing
- Per API request pricing
- "Per Segment" fees (unique to Storj)
- Differences in per-region pricing
- Tiers or classes of storage available
- Rate tranches or fair-use policies

Calculating maintenance costs

1. Versioning — *saves on recovery time and helps with the ability to delete old files*
2. Automated life-cycling — *saves money deleting old files*
3. Console user experience
4. Developer-hours in optimization
5. Detailed billing information/tracing available
6. Technical support quality and accessibility

   If it takes a week to get an answer from providers via email support, you may be inclined to pay more for better support or endure impacts on your business.
7. Financial Planning and Analysis
8. Reports and alerting

Other costs

- Compression-related charges
- Support charges (e.g., 10% support charge on overall object storage bill)
- Payment terms, which may increase the float time (we'll do a time value of money calculation)

Pricing Comparison

Dimension	Big 3 / Distributed	Decentralized
Storage Price Per GB for Standard Storage	It tends to be higher with the Big 3 storage providers.	+ It tends to be lower than the Big 3 storage providers.
Geo-redundancy	Multiple regions or zones adds more cost in most cases.	+ You get it for free.
Security/ Compressions	Some things like compression are done on your CPU time. However, encryption can be done at the bucket level, so this is cost-free. Generally, security protocols and access control are not charged.	+ Data is automatically encrypted and compressed for storage, so no CPU time is incurred. There are no charges for security protocols.
Transaction Fees	Transaction fees apply to every READ, WRITE, and LIST request above a low minimum threshold.	+ These tend to be lower or non-existent in some categories.
Support Plans	Support tends to be expensive and can reach 10% of overall spend.	+ Usually free.
Egress Fees (OUT OF PLATFORM)	It tends to be very high because they don't want you to move your data.	− It tends to be equal to or lower than Big 3.

Pricing Comparison

Dimension	Big 3 / Distributed	Decentralized
Egress Fees Within Platform	It tends to be $0 or very low.	Egress (outbound) fees do generally apply.
Monitoring, Alerting	These tend to have built-in alerting systems.	Currently does not have robust monitoring built-in.
Life Cycle Support/ Versioning	Easily applicable life-cycle policies and versioning.	Life-cycle and versioning must be coded manually.
Billing Management	These tend to be part of complicated bills that require more person-hours to manage or the purchasing of a separate billing management system.	These tend to lack detailed billing information.

Monthly savings

We estimated the monthly savings associated with switching to a decentralized provider, including:

- Impact of data movement between integrated services and a stand-alone provider. To the extent that data is moved between applications, there may be new data transfer charges.
- Costs related to daily usage that are not rate card-driven, like the UI. Some costs associated with use, such as log monitoring, specialized dashboards, are not part of the service package.
- Maintenance costs, including extensibility, licensing, monitoring and optimization. The general maintenance costs include the labor to change and maintain as well as monitor costs.
- Costs related to hiring security personnel to implement or ensure compliance with security protocols.
- The cost of hiring developers to implement connections with legacy providers where necessary.
- Allocation for required support plans in many cases.
- Monthly service billing for the storage provider.

In each case, we assumed today's prices. Future analysis might look at the relative change in pricing as different providers obtain increased economies of scale.

Risks and probabilities

Next, we created a risk-weighting for various aspects of the comparison. Risk, as we are using it here, can be described as a measure of two elements:

1. What is the probability of the user incurring the cost or reaping the benefit?
2. What is the value of the realized benefit given the risk?

Some things are not single point occurrences but can be a range of results over time, so this analysis can get quite complicated to implement without enough experience and an established roadmap.

Risk ultimately involves:

- Finding a level of risk preference,
- Calculating the expected risk weight values to benefit, and
- Discounting those benefits over time to reflect the time value of money and cost of capital.

Cost of capital

The opportunity cost of investing in projects of similar risk.

Risk preferences

Risk preferences are tricky because the value to a company is use case-dependent. For example, in the same industry, one company may view security as necessary but not pay more to avoid an extreme tail probability. At the same time, another may want its distribution to have no tails and be more confident of an outcome. These decisions frequently relate to product branding, corporate culture, and market demands.

Market risk

Another measure of risk relates to market risk and establishing a reasonable discount rate to use on the cash flows over time. In this case, we are adjusting for market risk, not the asymptomatic risk above.

The following sections detail the risks that our clients raised and their impact on their decision to use a decentralized storage provider.

Uncertainty of use case fit

This uncertainty relates to the cost of moving your data and then ultimately not being able to use the service as intended. For example, you may have forgotten to evaluate a specific need, or the vendor has not met your presented requirements. Additionally, roadmap changes of the vendor or the user can impact use case fit, including lack of backward or forward compatibility.

How it is addressed

We use risk-weighting to assess the appropriateness of a recommendation.

Uncertainity of actually getting lower costs

This is use case-driven and can eliminate all potential savings.

Most of the providers in the decentralized space are new and have positioned themselves as low-cost. There is evidence that their model will provide long-term cost benefits as it uses the excess capacity available to maintain its cost-base.

The typical risks associated with a purchase decision are tied to forecasting a company's actual usage and growth rates. For example, will the ex-post fail to meet the growth targets if growth drives the Payback Period?

How it is addressed

This risk is addressed with sensitivity analysis, like “what happens if the forecast is off?”.

Uncertainty of reliability, accessibility, and durability

Not having access to data can be costly.

This uncertainty relates to whether or not having decentralized storage reduces the likelihood of not being able to access your data or having it corrupted when accessed. So again, we believe that the differential here, if any, is minimal. But it is perceived as a risk by our clients.

How it is addressed

We did not make an allowance for this risk because it is mainly a cost issue. For example, the same service levels can be achieved on the Big 3 providers, while the architecture differs.

Uncertainty of benefiting from security cost/benefits

If an attack occurs, it can be costly and can effectively shut down a business.

A basic level of ransomware protection should be implemented for any type of storage provider, so we don't think it's essential to this decision.

With decentralized storage, the risk for breaches that target the storage provider instead of the storage user is reduced. If the use case demands that the tail probability of a provider breach be reduced, then decentralization adds value. First, however, the company must follow its internal protocols to place a value on it. For example:

1. Does it pay to go beyond "best practices" if your competition is not paying for higher security?

2. Do users have a more straightforward product implementation because of lower security?

How it is addressed

In this analysis, we have viewed this as a threshold function, not assigning it value but using it as a necessary condition.

Non-Quantified Direct Cost/Benefit

Using the risk measures described in the section above, we evaluated the following difficult to quantify cost/benefits the client expected to receive.

Less risk of developer mistakes on privacy

Because the data is always encrypted, developers cannot put up code that exposes data in transmission or at rest through a lack of encryption.

Irrevocable simple tamper free control

This benefit reduces the need for hard-to-find security professionals because many of the security components are built into the platform, which reduces the risk of error in certain circumstances.

Reduced surface area for attacks

This benefit includes ransomware attacks wherein an attacker manages to infiltrate or compromise a Big 3 provider. However, this risk is not reduced for attacks where a developer or user inappropriately exposed a key.

Quantified Organizational Benefits

Aside from lower organizational costs, there are few quantifiable organizational benefits identified for cloud storage. These would be benefits that are not a direct cost or occur directly with product use. Instead, they are secondary benefits that change the organization's opportunities or operations in a positive/negative way that results in structural changes to the organization that has a quantifiable effect or new revenue opportunities.

V. Results

We took seven of our clients that illustrated each use case type, and ran our "Detailed Analysis" on their data. The cards below summarize the results for these seven use cases.

HEAVY GRAPHICS SAAS **5/5**

Findings

Because the business is heavily involved with either long-term storage of images and customer media downloads, the decentralized pricing structure and potentially making a CDN unnecessary make this an attractive use case.

Main drivers

- Lower storage costs
- Lower transaction costs
- Cheaper egress to the Internet
- Cheaper concurrency

Impact

- **TCO** 65.10% lower
- **ROI** 102.78% higher
- **PAYBACK** 3 months

NO GRAPHICS SAAS **4/5**

Findings

Because these types of use cases have limited storage external to a database, the lower storage and egress costs associated with the non-database data can make switching that data justifiable.

Main drivers

- Lower storage costs
- Reduced cost of redundancy
- Cheaper concurrency

Impact

- **TCO** 55.93% lower
- **ROI** 47.20% higher
- **PAYBACK** 11 months

DATA SCIENCE WITH LOW STORAGE INTERACTION **4/5**

Findings

Because of the cost savings associated with decentralized storage, and the ability to load data in parallel, decentralization is an excellent opportunity for this use case.

Main drivers

- Lower storage costs
- Reduced cost of redundancy
- The ability to connect to 3rd parties is moderately valued.

Impact

TCO 55.34% lower

ROI 51.98% higher

PAYBACK 10 months

REAL-TIME DATA WORKFLOW **4/5**

Findings

For those real-time workflow situations where data needs to be collected, then moved, there was the benefit of decentralization. However, there was no real benefit for those use cases that were using heavy ETL or had tiny file sizes.

Main drivers

- Lower storage costs
- Lower transaction costs
- Reduced cost of redundancy

Impact

TCO 56.12% lower

ROI 48.03% higher

PAYBACK 11 months

CUSTOMER-FACING AND CONTENT-DRIVEN 3/5

Findings

This use case benefited from decentralized storage for non-transactional data. However, application data tends to be database-driven, so some expense may be realized getting data in and out of databases. Thus, there are savings on storage, but engineering costs can drive negative returns in this case unless storage volume is high.

Main drivers

- ✓ Lower storage costs
- ✓ Reduced cost of redundancy if used appropriately

Impact

- **TCO** 14.623% lower
- **ROI** 14.21% higher
- **PAYBACK** 35 months

DEV-HEAVY SOFTWARE DEVELOPMENT 2/5

Findings

This use case could benefit from decentralized storage, but the storage costs were not significant enough to justify switching in many cases. The example provided here is a development company that spends enough on storage each month to justify the engineering costs, albeit over three years.

Main drivers

- ✓ Lower storage costs
- − The general level of data utilization.

Impact

- **TCO** 31.67% lower
- **ROI** 12.30% higher
- **PAYBACK** 38 months

DATA SCIENCE WITH HIGH STORAGE INTERACTION **1/5**

Findings

Because the DATA TRANSFER IN/OUT related to tightly coupled services is not a good use case for decentralized storage.

Main drivers

- The complication of engineering to migrate.
- Data transfers between connected services.

Impact

TCO 14.95% higher

ROI 27.86% lower

VI. Recap

For the correct use case, decentralized storage may make a lot of sense and produce a significant positive impact. The most extensive impact areas are related to the multi-regional benefits, security benefits, and loss of a single point of failure.

We noted that no one provider is fully decentralized but that the direction is towards that end. Some unique considerations should be taken into account when deciding to switch. Feel free to reach out to us (email: team@taloflow.ai) for more detail on the reports and summaries we provided herein or help with your storage journey.