

Leapfin

**Buyer's Guide to
Finance Data Platforms
for Subscription
Companies**

Table of contents

Introduction	3
What is a finance data platform?	4
Benefits of a finance data platform	6
<hr/>	
From silos to systems signs you need a finance data platform	7
Key evaluation criteria to select the right finance data platform	9
Volume & scalability	9
Journal entry automatic generation use cases	11
Ease of integration	14
Dimensionality	16
Immutability & auditability	17
Accessibility	18
<hr/>	
Bottom Line	19
Get your checklist	19
About Leapfin	20

Introduction

Look inside a typical business and you'll see something akin to a mechanical Swiss watch: an array of different functions, spanning everything from order management to inventory and fulfillment, product and engineering, sales and marketing, and much more, synchronized in a delicate dance by production systems.

It's finance's job to report what happened — in standardized ways that investors can interpret — and assess and drive forward-looking changes to operations to move the business in the right direction. But production systems aren't designed for finance, so finance is often left to assemble, integrate, normalize, adjust, and summarize data into financial formats on its own.

For high transaction volume businesses with increasingly sprawling production systems, it's an impossible task.

That's why finance needs a data platform that turns operational data from production systems into finance-ready data.

But how do you know if (when) you've reached the point where you need a finance data platform?

And how do you evaluate the options to choose one that's appropriate for your business?

This buyer's guide answers those two questions.

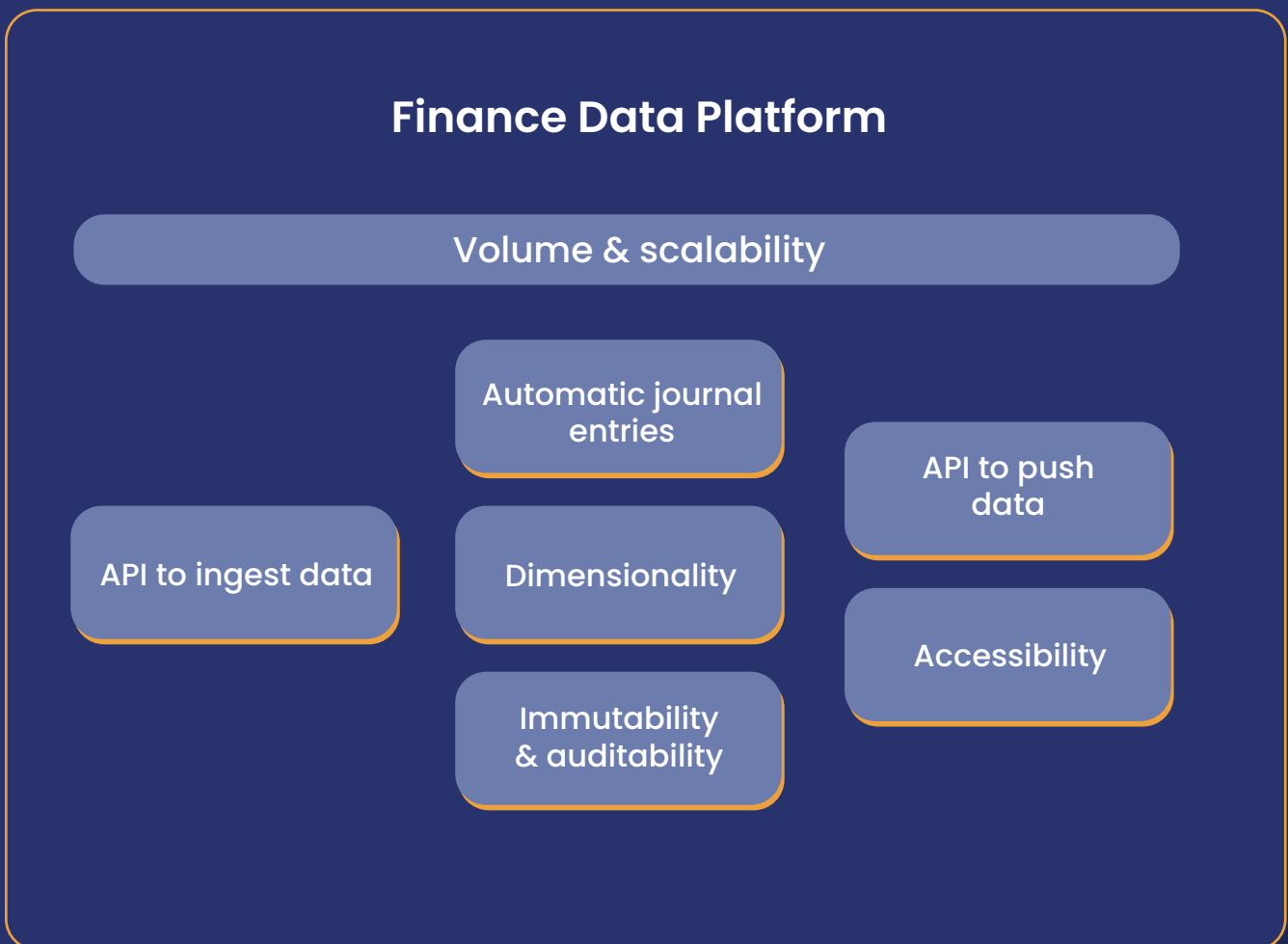
What is a finance data platform?

As you can see below, a finance data platform **connects to all the operational production systems** your business runs on, **normalizes, combines, and enriches that data** into financial records, and **makes those financial records available to reporting and insight systems.**



As the diagram below shows, to do that, any finance data platform must:

- Manage high transaction volumes
- Ingest operational data easily from many different sources
- Handle many dimensions
- Observe accounting rules like immutability
- Use accounting rules to turn source data into properly formatted journal entries that finance systems and teams can understand and rely on
- Make data easily accessible for downstream systems, reporting, and auditing



Benefits of a finance data platform

Controllers

Accurate revenue reporting

Eliminate estimates in financial reporting and adjustments after month-end. Disaggregate and report revenue and trends by country, product, platform, or any other dimension.

No month-end surprises

Report your numbers every day (or even intra-day), eliminating surprises so you can fix any concerns immediately. Track refunds, revenue stream trends, and changes over time without reporting bottlenecks.

Simpler audits

Giving auditors access to the data platform lets them perform sample testing, like tracing payments back to the processor, without burdening finance with data requests.

Finance teams focused on charting the best course for the business

More data driven decision making. Instead of using averages or aggregates to make broad changes, like customer acquisition spend, analyze revenue and profitability on a customer-by-customer basis to identify cohorts worth investing in and others to avoid.

CFOs

Better substantiation of initiatives to investors, so you can tell your operational story more confidently, increase investor trust, and lower your cost of capital.

Business systems / Finance Ops

Cost and risk avoidance of trying to build an internal system without sufficient dedicated product, engineering, and data team resources who understand accounting and finance.

An easier-to-use system with a proper front-end user interface.

From silos to systems: signs you need a finance data platform

Subscription-based companies share several common traits that create exceptional data challenges for their finance teams:

Traits and challenges for digital distribution businesses

Volume

Trait

Digital distribution companies often operate at far higher transaction volumes than similarly sized companies did a decade or two ago.

It's common for a company with only a few hundred employees to have tens of thousands, hundreds of thousands, or even millions of transactions per month.

Challenge

Data at scale is hard to move and standardize. As companies grow to tens of thousands of monthly transactions, handling data becomes challenging.

The ducktaping that finance teams can do at a small size – by downloading data, manipulating it in Excel, and uploading results to the GL or other systems for further analysis – becomes unworkable.

Indicators to look for

Many MB Manual CSV downloads

Excel files that take minutes to open (if they open at all)

Restating financials

Adjustments to prior periods getting larger rather than smaller over time.

Transaction-level data in the ERP leading to threatened huge overage charges and/or unacceptable software performance problems.

Engineering and IT proposes using a data warehouse, but doesn't fully understand complex accounting business logic, like immutable journal entries, locking closed periods, dunning rules, and auditability.¹

¹ Efforts to solve the challenge with an internal build should be evaluated against the same criteria presented in this guide. Additionally, consider if the business is prepared to dedicate product, engineering, and data teams to manage an internal product and process.

Speed

Trait

Digitally-native companies are able to change internal operations at incredible speed to adjust to their markets. Whether international expansion, new offerings, or pricing changes, the number of dimensions along which the business can experiment and iterate is much greater than in most traditional companies.

Internal technology

Digitally-native companies often have core technology competencies around their order management or fulfillment that are tied to their core offering. Unlike third party, off-the-shelf software, these systems may be less well documented, changes may not be as well communicated, and data structures less stable.

Distribution

As companies scale to multiple regions, countries, and currencies, they need to adopt many different payment processors with local presence and expertise

Challenge

Operational changes are not so easy to reflect in accounting systems, due to the number of production systems that need to be connected and rigid ERP data structures that make accounting complex.

Having a mix of internally built and best of breed production systems across payment processing, order management, fulfillment, and financial reporting creates data silos and integration maintenance headaches, because each system makes data available to other programs differently, with periodic changes that aren't well documented or communicated.

In the UK, card payments dominate, while in Germany open invoicing – where a consumer buys a product and pays for it later – is popular. In Asia, payment systems are often integrated directly into the major e-commerce platforms, like Alibaba or WeChat.

Indicators to look for

Managers lack the visibility they need. By the time they get month end data, the problem that started to emerge in week two of the previous month has been festering for a month. Because finance relies on summary data, when the business asks to summarize along different dimensions, a lot of manual work and estimation is required.

Finance leadership that can't confidently analyze or predict the effects of operational changes or explain the dynamics to investors.

Finance processes become a blocker to international expansion.

Key evaluation criteria to select the right finance data platform



| Volume & scalability

Given that one of the primary drivers for finance teams to consider a finance data platform is outgrowing manual Excel work, choosing a system that can operate effectively at scale is a foundational requirement.

Manage high data volumes of 1 million + transactions/month

If you're dealing with a small number of monthly transactions, Excel is a reasonably good tool for merging production data from various systems, calculating adjustments, and generating summary journal entries.

While you'll still need to rerun the process manually every time finance or leadership want to cut the data differently, with small volumes this is a relatively light burden. But as your volumes exceed 10, 20, or 30,000 transactions a month, the challenge of using spreadsheets to work with this data will start to become untenable. They'll start to be very slow, and eventually won't open at all.

As you're looking for a better solution, don't solve for the volume you have today, but for the volume you expect to grow into over time. The last thing you want is to implement something that won't last.

Process data as fast as needed: daily, hourly, or sub-hourly

For most businesses used to measuring finance metrics monthly with a multiday (or even multi week) close process, just shortening the effort of the close process is an improvement, even without generating data intra-month.

But once the manual constraint of work is lifted, most businesses will discover an appetite for accurate transaction-level finance data that's almost continuous.

Automatically flag data hygiene issues; frequency & response time of any issues should be reported

When operating at high volume and speed, failed or incomplete data syncs from integrated systems, like payments, can cause big problems with accounting accuracy and reconciliation downstream and can be hard to pinpoint and address, so the system should make it easy to identify when such issues happen.

Low downtime

Look for a system that's highly reliable — no more than .05% in the last 12 months (99.999% or "Five Nines" reliability — so you always have confidence in the data reported to you.)

System downtime scheduled & communicated in advance

In addition to minimizing downtime, knowing when it's coming in advance is critical for instilling confidence.

Journal entry automatic generation use cases

The core functional value of a finance data platform is the ability to create journal entries automatically for every transaction, and allowing you to summarize them across any dimension you wish. But, of course, there are a lot of nuances to accurate recording of journal entries, so look at the list below to make sure you're considering the accounting rules critical to your business.

Double-entry bookkeeping

This requirement sounds almost redundant or tautological, but many engineering and product teams in house aren't familiar with accounting practices, so might miss this absolutely critical aspect.

Recalculate journal entries when existing transaction updated

Additions or modifications to an underlying order should always record and trigger appropriate updates to accounting events.

Replace original journal entries with recalculated journal entries, unless original falls in locked period

Additions or modifications to an underlying order should always record and trigger appropriate updates to accounting events. This ensures SOX compliance and traceability during audits. It's non-intuitive to nonaccountants that once a period is locked, it cannot be changed, so this design requirement is easy to miss.

Generate transaction-level journal entries for revenue recognition

High data volumes often get in the way of timely revenue recording. Automatic journal entry creation ensures a smoother end of month closing process.

This is even more complex when an order has two different items that need to be recognized differently, like annual or multi-year subscriptions, overlay services (recognized on delivery), and so forth.. Treating each item as a separate transaction makes this easier to calculate correctly.

Handling subscriptions in aggregate is extremely complex. Proper accounting would require a schedule of all possible combinations of start, change, and end dates and manually amortizing revenue for each situation. The number of combinations is prohibitively high.

Aggregate transaction level journal entries & post summary journal entries to general ledger

ERPs are not designed to perform well with high transaction volumes, and customers who try to store high transaction volumes in them will pay a huge price premium and performance penalty. It's much more cost effective and time-efficient to maintain transaction-level journal entries in a finance data platform, and aggregate to the summary level for posting to the general ledger or ad hoc analysis.

Standalone Selling Price (SSP) allocations for correct accounting of product bundles and discounts

Automatically adjust selling price in situations like bundle level discounts or buy-one-get-one-free, where net invoice amounts need to be proportionally allocated to individual items for accurate accounting.

Handle credit issuance and application accurately

Credits are complex because for accounting purposes it's necessary to distinguish between and track both issuance and application separately, while relating the two.

Full & partial refunds, credits, disputes, chargebacks

Refunds, credits, disputes, and chargebacks, are complex, but it's important to track the status of each to ensure proper refunds or revenue are recognized.

Manage subscription amendments, like mid-cycle up/downgrades

Handling subscriptions in aggregate is extremely complex. Proper accounting would require a schedule of all possible combinations of start, change, and end dates and manually amortizing revenue for each situation. The number of combinations is prohibitively high.

Instead, managing amortization at an individual transaction level ensures accuracy and maximizes flexibility for the product team and customers to up/down grade.

Downgraded subscriptions often result in issuing a credit for an unused period, so handling partial credits or refunds (see above) is a related requirement.

Dunning

Dunning is a complex workflow because your payment service provider will try to process payment repeatedly over a period of time. At any given moment, some payment requests will be paid, some open, and some have failed out of the dunning process. Your finance data platform should be able to automatically debit the payment in transit account (for example, your Stripe wallet account) and credit accounts receivable, when the payment status changes.

Multi-currency calculations

Exchange rate gains and losses are an important expense account to track. Make sure your system handles this with an integration to an FX provider with daily market rate updates, but also allows using a specific rate you determine. This includes handling local (or presentment) and settlement currencies, converting each transaction to a single reporting currency or reporting in base currencies.

Differential updates

Most production or operational systems track the latest state of a given balance, overriding historical values. That's a problem for finance, which has to close the books and then track changes in balances from month to month.

For example, if a customer has a credit on their account for \$100 at the end of September, that needs to be recorded on the books accordingly. If they spend \$20 against that credit in October, the value of the credit in the production system will update to \$80, but the finance system should record a \$20 debit against the credit account.

Similarly, if an invoice payment fails on September 30 and clears on October 1, there should be an entry for an unpaid invoice for September and a second entry for the payment for October. But a production system would lose the September status and any ability to identify that the invoice had been unpaid until October.

Ease of integration

A big challenge for finance teams is managing the multitude of data sources and streams that converge to calculate journal entries and the three financial statements. Even apart from the volume, it's really challenging to manage the complexity of normalizing across many different data formats, monitoring and correcting for changes in source system exports, and the time and manual, error prone process of downloading many CSVs and working with them in a perfectly consistent way. A great finance data platform handles all of this messy work automatically and in a standard and flexible way, saving your team a huge amount of work and risk.

Internal or best of breed billing systems

Pre-built integrations with internally built billing systems or vendors like Recurly, Stripe Billing, Chargify, or Zuora speed up implementation and remove the burden of maintenance as APIs or data formats change at any given billing system.

Payment service providers

Unlike billing systems, where a given company only has one or two, payment system provider (PSP) proliferation is a necessary cost of international expansion and customer preference. As a consequence, many companies have a large number of different payment service providers, each making data available in different formats with different quirks.

Make sure you pick a finance data platform that has pre-built integrations with the widest number of PSPs to both serve your immediate needs as well as minimize the headache of future expansion. Look for integrations with Adyen, Affirm, Amex, Authorize.net, BlueSnap, Braintree, Checkout, ChasePaymentTech, Discover, PayPal API, PayPaly SFTP, and Stripe.

Also give preference for an API that makes it easy to connect other payment systems providers that might not have pre-built connections. That will derisk

AppStore & Google Play

To handle mobile transactions like apps and games, the platform app stores manage billing and payments. Their lack of transparency can make integration challenging for non-experts.

Order management system

Some organizations like to track inventory and the entire Order-to-Cash journey. To do so, your finance data platform should be able to integrate with the right internal data source to understand when an order is placed, shipped, received, etc, and trigger appropriate journal entries at each step. (The OMS may be internally built or a best of breed vendor; your financial data platform should work with whichever system you have.)

ERP

Your CRM can use revenue recognition data to present tailored offers to particular customers in renewal cycles. A direct integration is necessary to automate this process.

Other integrations

A finance data platform should be flexible enough to accommodate other integrations apart from those mentioned above, like bespoke proprietary systems with an API that allows the finance data platform to regularly collect and store financial data without manual input.

| Dimensionality

A critical component of an effective financial data model is its flexibility to accommodate the ever-changing nature of your business. ERPs, as a counterpoint, generally don't accommodate structural changes to the GL very easily. But it's hard to predict the nature of your business in six months, let alone a few years in the future, so ensuring your systems are flexible for what's to come may be essential.

Unified, extensible data structure (data model or schema)

Your financial data model should process data from internal and external sources and normalize it into one standard structure, allowing data from different sources to be compared. For example, payments processed by Stripe can be compared to payments processed by Braintree.

To ensure accounting accuracy, easier reconciliation, normalizing silo'ed/disperate financial data systems becomes important for better reporting as well.

Easily create & add new fields post-implementation, such as new product lines, GL codes, bundles, etc.

As your business grows and gets more complex, the data structure needs to be easily extended. For example, you might start by offering discounts on a single subscription product. Later, you might have multiple subscriptions and offer discounts on the bundle as well

You might respond to an economic downturn by shifting from growth at all costs to profitable growth, which necessitates studying transactions by customer segments more closely to identify which segments have the highest contribution margins. Each of these evolutions must be accommodated by the financial system.

A finance data platform built on an extensible graph database is the ideal technical architecture to facilitate this.

| Immutability & auditability

Every accountant knows that once you close the books the numbers are final, and any adjustments get reflected in the following period. But this is not how many other operational systems work, so it's not the most common design structure if you're using an internally maintained finance data warehouse, for example.

Locking closed periods

A finance data warehouse should be designed to allow locking periods when they are closed to prevent any further transactions from having an impact on a given period. Once closed, no changes should be made to an accounting period to ensure validity, accuracy, and reliability of those numbers.

Audit trail

Financial records should all have a clear and complete history of changes to make auditing results painless and instill confidence with auditors in your results and controls.

Traceability

It should be easy to identify all financial records that result in a GL entry. For example, if \$500,000 in revenue is booked in the ERP, it should be easy to identify all the transactions that led to that revenue.

Easy audit sampling

Auditors should be able to easily query with a search interface.

Accessibility

A critical function of a finance data warehouse is making finance data easy to use downstream – for closing and auditing the books, financial analysis, and decision making.

Track balance & changes in each financial account defined by Chart of Accounts (COA)

To ensure accuracy and reliability of numbers within the COA.

Easily modify COA; apply modifications to all historical & future data

It's common to add or split the COA when new product lines are introduced. For example, if you add different subscription tiers to your business models, accounting may need to split the revenue and deferred revenue accounts accordingly

A well designed finance system should have a centralized COA used across the entire process. Centralization ensures consistency and is easy to update. But it also requires systems that make the structure easy to adjust when needed.

Present revenue reporting aligned to COA

Balance sheet reporting is usually done at the COA level, so it is important that your finance data platform allows you to aggregate in an aligned way.

Regular reporting such as A/R aging reports, tax rates, discounts, & ad hoc reporting

Reporting requirements are fairly common for every organization. A strong finance data platform makes it easy to create whatever reports are required.

Search functionality for reporting reports, tax rates, discounts, & ad hoc reporting

Finance and accounting teams often need to run ad hoc reports and analysis. For example, if there's an invoicepayments discrepancy, or to report on the refund rate for a given product line.

Manually-driven accounting organizations usually struggle to merge data and search across spreadsheets for every request.

Transaction search allows finance and accounting teams to quickly and easily capture exactly the data they need for ad hoc analysis.

Bottom Line

The need for accounting and finance teams to get better control of their data and make it available for reporting, analysis, and action, has never been greater than today, and is only growing as business gets more complex, more global, and more dynamic. Regardless of the particular system you chose, keep in mind that to do the job, any finance data platform must deliver powerful functionality in each of the six main areas: volume & scalability, journal entry generation use cases relevant to your business, ease of integration, dimensionality, immutability and auditability, and visibility.

If even one of these elements is missing, your solution won't really be solving the problem you have. It won't provide you with control, and it won't provide your business with value.

Get your checklist

We've made the criteria above into a handy spreadsheet you can use as you consider your options. In the spreadsheet, you can assign your preferred weights to the criteria most important to you, and score each vendor (or internally proposed solution) against those criteria to get a weighted average vendor score.

[Download checklist](#)

Checklist

Checklist text

Checklist text

Checklist text

About Leapfin

Leapfin turns production data into finance data in real time for the world's leading digital brands.

The Leapfin Finance Data Platform unifies billing, ordering, payment, CRM, and other production data with intelligent accounting automation to create a GAAP-compliant, immutable Leapfin Financial Record for every transaction.

The Leapfin Finance Data Platform provides reliable and fast financial reporting to controllers, transaction-level detail for analysis to finance, and visibility to real-time revenue to decision makers.

Canva, Reddit, SeatGeek, Medium, Vimeo, and many other fast-growing, high-volume digital brands trust Leapfin to be their finance data system of record.

To learn more about Leapfin, visit leapfin.com and follow us on LinkedIn or join the conversation on Twitter.

