



GoodData

GoodData Corporation

The Ultimate Guide to Going to Market with Embedded Analytics



Innovate. Differentiate. Optimize. Monetize.

Over the past few years, the number of SaaS solutions and providers has multiplied before our eyes in response to booming business demand. Gartner predicts that revenue from SaaS technologies will reach \$85 billion by the end of 2019 - a 17.8% increase from 2018 - and that the SaaS market is expected to grow from \$94.8 billion in 2019 to \$143.7 billion by 2022.

In this rapidly changing competitive environment, SaaS leaders are not only charged with keeping up with their peers but also with creating both business and customer value. This is especially true for software sales where, with the relatively low barrier to entry and the rapid speed of innovation, SaaS companies are constantly challenged to find new ways to augment and differentiate their existing products to capture and retain market share.

The good news is that, as a SaaS provider, you're already one step ahead of the game, because the resource that will fuel your next product and revenue upgrade is already available to you: your application's data. Analytics are the key to strengthening customer relationships, improving loyalty and retention, and driving ongoing revenue. In fact, Nucleus Research has found that analytics return \$9.01 for every dollar spent.



Are you ready to be a first mover?

Now is your opportunity to be a first mover in your space and begin monetizing your data. As a SaaS provider, you have a ready-made path in front of you that leads you to new monetizable products and features that will catapult your business ahead of the competition by integrating analytics into your offering.

GoodData is here to help you race down that path with the tools and resources you'll need to successfully go to market with an embedded analytics product. Our multi-tenant analytics distribution platform is custom-designed to help ISVs quickly integrate, deploy, and automate the delivery of targeted analytics to tens or thousands of customers in real time with:

- ▶ Seamless embedded analytics that can be packaged, tiered, and distributed at scale
- ▶ A highly engaging end user experience that quickly guides customers of all skill levels to insights that inform action
- ▶ Expertise and organizational support from implementation to launch and beyond

Ready to go to market? Let's get started.

Build your blueprint for analytic GTM success

If you're new to analytics, you may be unsure of whether your existing go-to-market (GTM) processes are transferable to a data project. GoodData offers go-to-market expertise that will help you draft your strategy and accelerate your path to launch and ROI by following these 7 steps to get started.

A photograph showing a person's hand holding a silver pen, writing on a whiteboard. The whiteboard has several colorful sticky notes pinned to it, including yellow, pink, and red ones. In the background, there's a wooden table and a glass of orange juice on a black saucer.

1. Kickoff with stakeholders

If you're thinking about your GTM strategy, you should have already chosen an analytics provider. You've had the build vs. buy discussion, kicked the tires of several solutions, and chosen a vendor that has the right infrastructure, capabilities, and SDKs to accelerate your time to market. Hopefully, you've also thought through some of the key requirements for success in embedded analytics, like multi-tenancy, security, automation, end user engagement, and cloud distribution. So now it's time to get your execs in a row.

The fastest way to kill your product update while it's still on the vine is neglecting to gather feedback from key stakeholders up front. Your first task when planning the launch of a new analytics offering is to decide who needs to be involved in defining the goals, constraints, users, functionality, structure, pricing, and support processes. Assigning categories of ownership, support, and approval is your key to preventing late-stage objections that could delay your time to market.

2. Set goals and parameters



Sure, everyone has their own objectives to meet, but what are the ultimate business goals for this project? Or for this launch?

Your job is to keep everyone focused on S.M.A.R.T goals as you move forward. Without tangible goals, it's nearly impossible to figure out if, and when, your product is truly ready to launch. Here are some tips to keep in mind when defining goals and constraints during kickoff.

You might not be able to have it all this time around. It's important to determine which functions are absolutely essential and which are nice-to-haves for your initial product launch in order to get the first iteration of your product to market quickly.

Must-have functionality we'll refer to as "table stakes." Stuff that's desirable, but not required, we'll call "delighters." Once you develop each list, refer to it when people suggest changes or fixes. This exercise is critical to determining the MVP-minimum viable product-for your launch. Doing so will ensure the team is working on critical functionality required for launch first, instead of worrying about the "shiny objects."

Focusing on your shortlist is the best way for you to test its performance in the market while retaining the flexibility to iterate in response to key learnings or changing market demands. You may also want to take this time to map out the MVP for following releases, so you'll have something to refer to and refine as you gather insights on what's working and what's not.

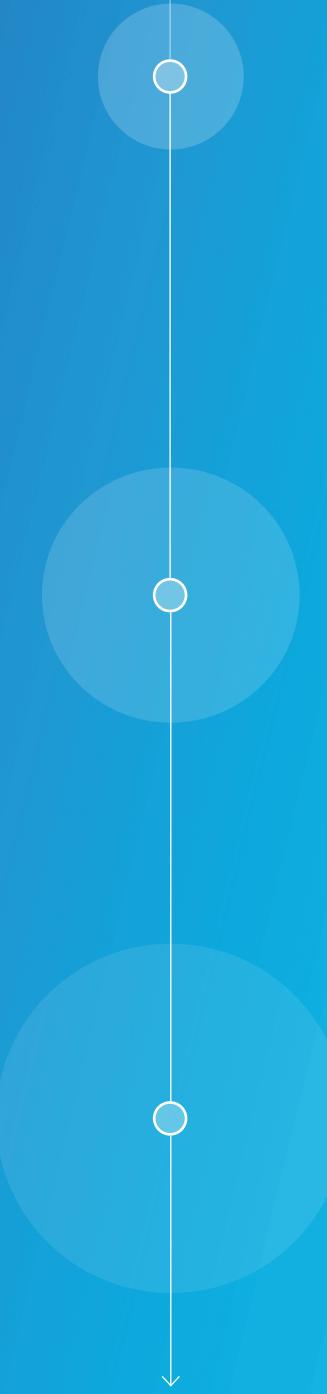


Table stakes

Required elements

Examples of non-negotiable required functions for your embedded analytics solution typically include the following.

Launch 1 (6 weeks)

- ▶ Single sign-on
- ▶ Embedded within core app
- ▶ Standard executive dashboard
- ▶ Ability to drill down into a report
- ▶ Ability to share via email

Launch 2 (12 weeks)

- ▶ Ability to create and save personal dashboards
- ▶ Ability to drill across into parent application
- ▶ Custom email alerts

Launch 3 (18 weeks)

- ▶ Ability to create ad hoc reports
- ▶ Pre-built advanced metrics

A close-up photograph of a person's hand holding a silver pen, poised to write on a white grid notebook. The notebook has handwritten text in blue ink, including the words "Drive" and "Maz".

Nice-to-haves

Examples of functionality that would be nice to include but may get pushed to a future release include the following.

Launch 1 (6 weeks)

- ▶ Dashboard filters
- ▶ Ability to save filtered views of data

Launch 2 (12 weeks)

- ▶ Ability to export to CSV and PDF

Launch 3 (18 weeks)

- ▶ Forecasting
- ▶ Customizable filters





3. Define your audience

Once you've made your MVP shortlist, it's time for a sanity check. You may think you know your users, but do you really? Take a step back and ask yourself: "Are the people who'll be accessing our product's new analytics capabilities the same people using our solution now? How does adding analytics change our users' requirements?"

Let's say that your SaaS product helps businesses manage customer service issues. On a daily basis, customer service representatives (CSRs) currently log into the system to create service tickets, answer questions, and find solutions to customer challenges. But now that you're adding analytics, there is a high likelihood that a whole new group of people are going to be accessing your solution, and each of those people brings their own unique requirements. For example, the VP of Support may want to look into the performance of his service team, or a C-suite user might be interested in seeing how service is impacting the business's NPS or customer satisfaction.

Choose your personas wisely

Look at key groups, both internal and external, who'll be tapping into your product for data, then pick the top 2-3 to focus on for your first iteration. Escalate personas by balancing the value delivered to the user, their ability to provide feedback, and the complexity of building for their requirements, then create a timeline for prioritizing the others. This exercise will help ensure that feature development is clearly mapped to real user needs.

It helps to begin with existing users, as they will offer the most immediate feedback and help you understand your levels of engagement. The more engaged your users (and customers) are, the greater lifetime value you can expect out of them.

Create a map of your buyer's needs

Once you've selected your personas, take a data-driven approach to understanding their needs and how they might turn into requirements.



Start with the basics:

Give them a name and a title. You're going to get to know these people well, so why not get personal?

► **Look for the pain:**

Outline key usage scenarios. What expectations will each persona have for your analytics?

► **List usability requirements:**

Determine what each persona needs to be able to use analytics effectively. What technical requirements should be prioritized?

► **Identify key questions:**

Define the questions they need to answer. What reports might they need to help them?

► **Define the data:**

Review the starting data universe. Is it easily accessible and how frequently does it change? Is any of it sensitive?

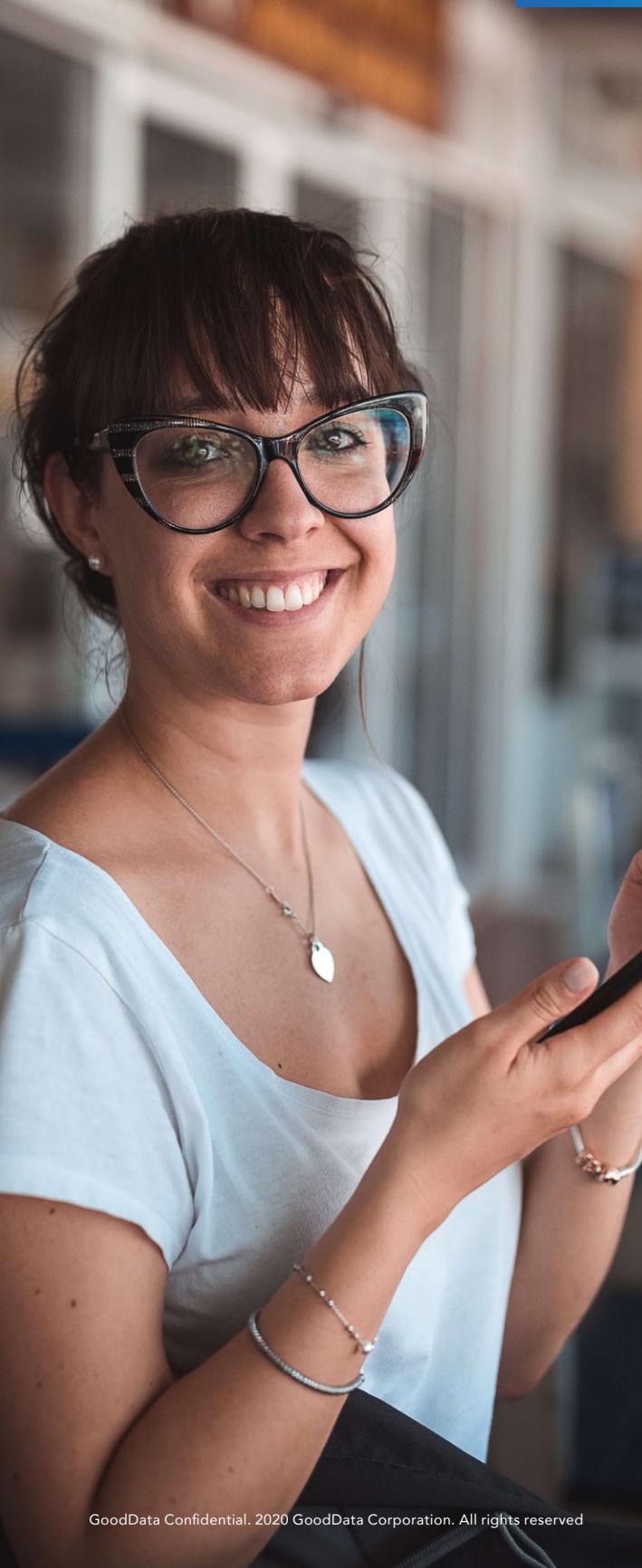
Create a feature wishlist:

Decide on a shortlist of “table stakes” features. If you can’t do it all, what are the must-haves?

 Sophie

Customer support representative

Core product user



Questions:

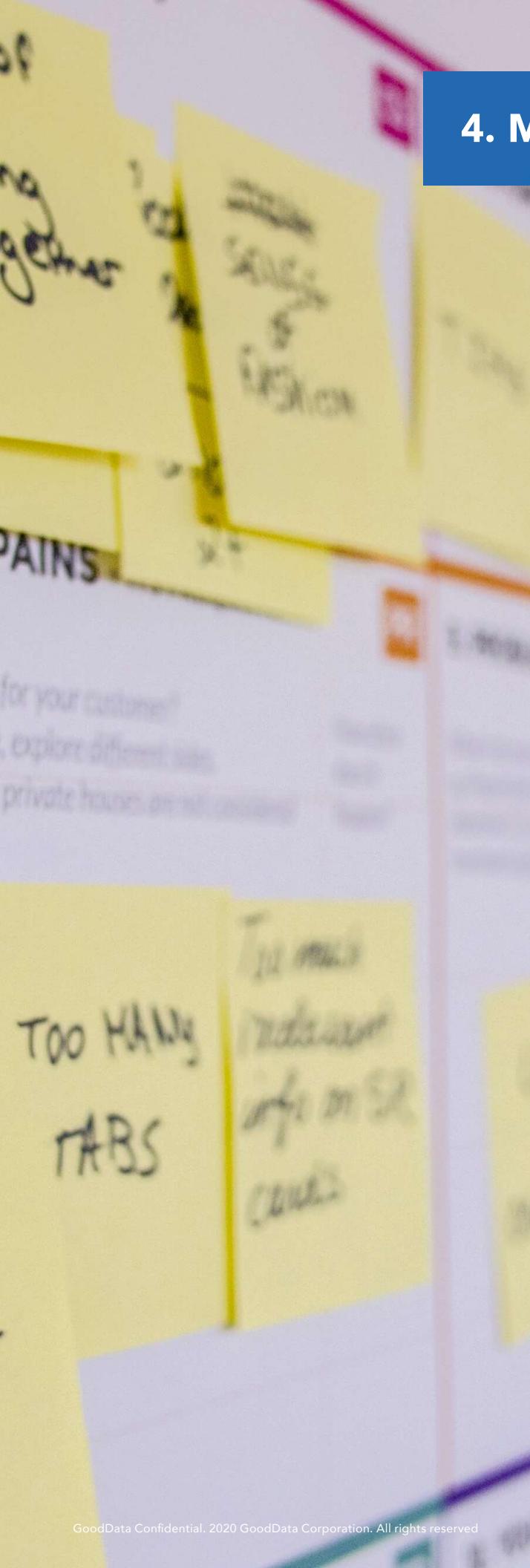
- ▶ What is my ticket close rate?
- ▶ How does my performance compare against my peers?
- ▶ How do customers rate my level of service?
- ▶ Do support calls vary by product or region?
- ▶ Call volume is too high to deliver my best service.
- ▶ I'm spending too much time answering basic questions.
- ▶ My boss doesn't know when I'm being overloaded.
- ▶ I never know if I'm going to get my bonus or not



Key usability requirements:

- ▶ Intuitive drilling to answer the "why"
- ▶ Deep integration with core application to link specific workflows with a given insight
- ▶ Easy sharing options

4. Map your requirements



WWSD? What would Sophie do? Answering this burning question is the next step in your GTM blueprint as you design workflows that will drive product requirements for each persona. Look to your key stakeholders and within your organization or trusted customers for SMEs who can give you honest feedback about whether or not you are on target.

- ▶ **Brainstorm their top five activities:**
Figure out where they spend their time. For example: repetitive calls and hang-ups; researching solutions; summarizing conversations; and closing tickets.
- ▶ **Name each key persona story:**
Give each task a short, descriptive title, like "Track personal support performance."
- ▶ **Build out workflows for each story:**
Choose 5-7 steps that you think the persona might take to complete these goals. For example: 1. Log in, 2. View trend chart, 3. Drill down into problem areas, etc.

Workflow

Now that you're feeling like you've become one with Sophie, it's time to shift gears and figure out how these workflows impact your analytics requirements. Go back through your workflow chart and assign analytical capabilities to each relevant step. Your goal is to make sure that not only are you meeting your user's needs, but that you're not wasting time solving nonexistent problems. You're also double-checking that you haven't left any gaps where a competitor could step in and differentiate.



5. Determine structure and pricing

Make decisions about product structure

In thinking about the structure of your offering, you'll need to decide if you want to take the most simple approach of going all in, offering the same level of access to every user, or if you want to give yourself the flexibility of a tiered product model, where functionality is offered at different levels to subscribers. Although the all-in approach is the simplest to employ, it limits your ability to monetize added functionality or recoup incremental investments. It also treats all customers as equal—including that one who taxes your resources the most.

The tiered product model—with levels like standard, pro, and enterprise—gives you both flexibility and the opportunity to incentivize your customers to upgrade. In fact, you may already have a model like this in place for your current SaaS solution.

Evaluate your cost of goods (COGs) and how that might impact your packaging strategy. If you have licensed your provider's analytic solution by user, then your strategy may be different than licensing by customer, regardless of user counts.

Finally, think about engagement. If you are creating tiers, what capabilities build engagement with your different personas? As you start ideating what might fall into your various tiers with your product team, ask yourself:

- ▶ Am I giving engaged customers a good reason to move from one tier to another?
- ▶ Do I have an ROI model and other measures of success in place?
- ▶ How many customers should we expect to upgrade vs. stay at entry level?
- ▶ How is our offering looking compared to competitive products?
- ▶ Have I fully baked in the costs I'll need to cover for each of these features?

Remember to balance your offerings and not expect customers to automatically pay for premium without having moved through the other tiers. You should also be sure to balance your revenue goals against your adoption goals, as some features may be more valuable to you as an adoption mechanism than as a line item.

Set boundaries for your product

Let's face it, you can't do everything. And as much as you want to surprise and delight every customer, you're going to need to resist the temptation to "yes" yourself to death by granting each (seemingly harmless) request.

Even the smallest deviations from your product plan can have a far-reaching impact on your ability to serve the larger customer base. For example, let's say you have a customer who's eschewed a popular data source for a home-grown solution, and now they want you to deliver a custom build. They're willing to pay, but are you willing to do it?

Before you decide, think about the implications. Do you have the resources available, or will they need to be pulled off other projects that could offer a greater return? When they need support, will you have to provide specialized training to your service staff? Will the operations team resent you as they struggle to track all the unique costs associated with this customer-and potentially others? When your regional sales director comes to you with an attractive new deal that comes with some caveats, be prepared with your list of:

- ▶ **Features and functionality supported as part of the core product:**
Green light. Of course we'll do it, it's part of the product!
- ▶ **Features and functionality we'll support for an extra fee:**
Yellow light. Connecting to different data sources, or building custom charts, for example, will cost the customer more.
- ▶ **Features and functionality we won't support:**
Red light. Connecting to home-grown applications, for example, might not provide the ROI to justify the resources.

Get what you pay for: Charging for added functionality

Before we go any further, let's address a big question we often hear during GTM consultations with embedded analytics customers.

"Should we charge our customers for the analytics?"

The answer is unequivocally yes. If you deploy analytics structured for specific personas and their pain points, designed to solve business problems, then you're providing your customer with tremendous value that should be charged for. Giving away such rich functionality will degrade its value in the eyes of the customer.

Furthermore, it backs you into a corner should you want to upgrade your product by adding analytic richness or predictive or prescriptive analytics, all of which you'll want to charge for. Finally, assigning a financial value to the analytics you're providing forces you to look at those elements critically as a core component of your overall GTM. Even if you simply raise the overall price of your entire platform, you are declaring that the addition of analytics brings new value.

It's a good idea to think about price as you're defining functionality, because these two items are directly correlated. A change in one often necessitates a change in the other. Want to store more data? You're going to need to increase your price to cover those costs. Want to offer a basic tier free of charge? Then you're going to need to hold some functionality back if you ever expect users to upgrade.

Name your price: Choosing the right price and model

Now that you know you're going to charge customers for the analytics, the next questions are "How much will you charge? And how?" When setting your pricing, you should keep three core goals in mind:

1. Recovering your cost to provide the analytical capabilities and operation

2. Signaling the value of the added functionality while encouraging usage

3. Developing lasting value to cover investments in future product and development goals

The first goal is the easiest to tackle; you want to charge enough that you're, at minimum, breaking even. Losing money is a showstopper.

A little "back of the napkin" math should give you a sense of the minimum you'll need to make to recoup your costs.

Divide your ongoing monthly analytics cost by the number of customers you expect to sign up for it, then add in a percentage of the initial product development costs you need to chip away at (aim for 50% in year one). That should give you a rough idea of the minimum monthly fee you'll require per customer or user.

The second goal is a little harder. Does the number you came up with seem low to you, in comparison to what customers are paying for your existing capabilities? If so, you may want to consider a model that charges an incremental percentage increase of 10-15% for the added analytical functionality, instead of utilizing a fee-based model (more on that next). Or does it seem high? If that's the case, you're going to need to consider reducing your delivery costs by limiting some areas (storage, etc.) or develop a sales game plan to grow your customer count. Again, the goal is to find an attractive price that still signals value.

The third goal is going to require some experimentation to get right. This is where you go back through the formula and make adjustments to costs, recovery, and profits until you find the right fit. Price your product too low, and you'll gain customers but lose money. Too high, and you'll lose customers-and money with them. Look for that sweet spot above your current price and your future costs, but right on the edge of your competition.

Finally, let's take a look at the "How" part of the equation. How will you charge customers? Here are a few options to consider:

Option	Pros and cons	Example
A. Percentage increase of or over base	<p>PRO: If your product is relatively expensive, the upcharge could be fairly small, allowing you to deliver lots of customer value while making a healthy profit.</p> <p>CON: If your product is offered at a relatively low price point, the incremental cost to deliver analytics might seem high. (See Option B)</p>	<p>Financial SaaS Product X costs 1% of assets managed.</p> <p>Add analytics and increase the fee to 1.25%.</p>
B. Flat line item charge	<p>PRO: A logical choice that makes it easy to segment customers that are interested in adding analytics from those that aren't - without disrupting current billings.</p> <p>CON: A point of negotiation and potential contention during sales discussions that could result in you giving the value away.</p>	<p>Marketing Automation SaaS Product C costs \$1,000 per year for three users</p> <p>Add analytics and charge \$1,100 per year for three users.</p>

Option	Pros and cons	Example
C. Transaction fee increase	<p>PRO: Great choice if your product pricing is already transaction-based (like an ordering system), allowing you to add a small surcharge into the existing per item cost.</p> <p>CON: Not so great if your transaction volume is low, which could make the upcharge very noticeable. (See Option B)</p>	<p>Logistics SaaS Product B charges a fee of \$0.05 per transaction.</p> <p>Add analytics and charge \$0.06 per transaction.</p>
D. Value-based fee	<p>PRO: Good option if you have a smaller volume of high-revenue customers with whom you can have a consultative relationship. Also eliminates COGs concerns.</p> <p>CON: Difficult to gauge and price point may vary by customer, increasing sales cycle time.</p>	<p>Operations SaaS Product D charges a variable fee based on set criteria.</p> <p>Base your product fees on how much value each of your products offer your customer. If they sell a product for \$1,000 and your data product contributes 10% of the value, price your product at \$100 for that customer.</p>

Want to see what one of these models looks like in action? Check out the example below to see how you can drive revenue with a flat line item charge for analytics.

6. Develop support requirements

Readiness planning is one of the most critical elements of your launch. You need to be sure that every department who'll be touched by this launch is prepared to support it, notably legal, finance, sales, and marketing.

Legal readiness:

Of all of the aspects of preparing your product and processes for launch, legal is the area that can cause the greatest delays or even damage, if not addressed properly. Involve your legal team early so that they can form a comprehensive understanding about the project and identify potential issues. Key areas of concern for your legal team will include:

- ▶ **Data usage:**

Whether you have customer personal data, credit card information, health-related data, or transaction history information, the legal team will need to understand the data you plan to store, transmit, and process.

If you are managing sensitive information, they'll be able to give advice on how to mitigate risk so that you don't open your company to legal actions, ensuring that your vendor's technology security aligns with your obligations.

- ▶ **Service-level agreements (SLAs):**
Your contracts probably already offer a minimum system uptime and issue response/resolution time guarantee, but what if these don't match what your analytics provider offers? You'll want to investigate their SLAs, disaster response protocols, and more. You may need to modify your contracts to promise "availability for core system functionality, excluding our non-transactional analytical capabilities" or something to this effect. Your legal team should be able to figure out the right approach.
- ▶ **Benchmarking capabilities:**
One of the most valuable things you can offer as a provider of embedded SaaS analytics is benchmarking data, culled from a cross-section of clients. But to do so, you'll have to be sure you've gotten explicit permission, been thorough in anonymizing the data, and given customers the ability to opt in or out.
- ▶ **Third parties:**
Finally, you're going to need to work with legal to update all your customer contracts with information related to all of the above, while also disclosing that a third-party analytics vendor will be processing your customers' data. Getting in front of these conversations will help ensure a smooth launch.

Finance readiness:

In addition to providing valuable insight on the pricing discussion (and making sure your vendor's terms are set up to meet your long-term needs), it's important to get your financial team's help with:

- ▶ **Usage monitoring:**
Can you track where your resources are being used, and if you're getting paid for them? You'll want to keep a tight rein on seats, data storage, professional services, and more. For example, if you're billing based on data volume, prepare finance to generate a monthly report that breaks out data usage by customer account.

► **Billing processes:**

You'll also want to enlist finance's help in determining what changes need to be made to your billing solution and processes to accurately charge customers for the new offering. It may be as simple as adding a new line item, or it may require you to create a new billing category. Make sure you consider the process for changing the structure of your agreement if your solution shifts.

Marketing readiness:

Corporate marketing will be involved in multiple highly visible aspects of your product launch, so it's good to get them on board early.

► **Product collateral:**

Obviously you'll want to promote your new features via sales presentations, brochures, email signatures, and more. Give your team the time they'll need to produce high-quality materials to support a successful product launch.

► **Website updates:**

Your website should show off your newly enhanced product and give customers a reason to upgrade. Start thinking through messaging, visuals, and demos early in the game, so that they're ready to go when you are.

Sales readiness:

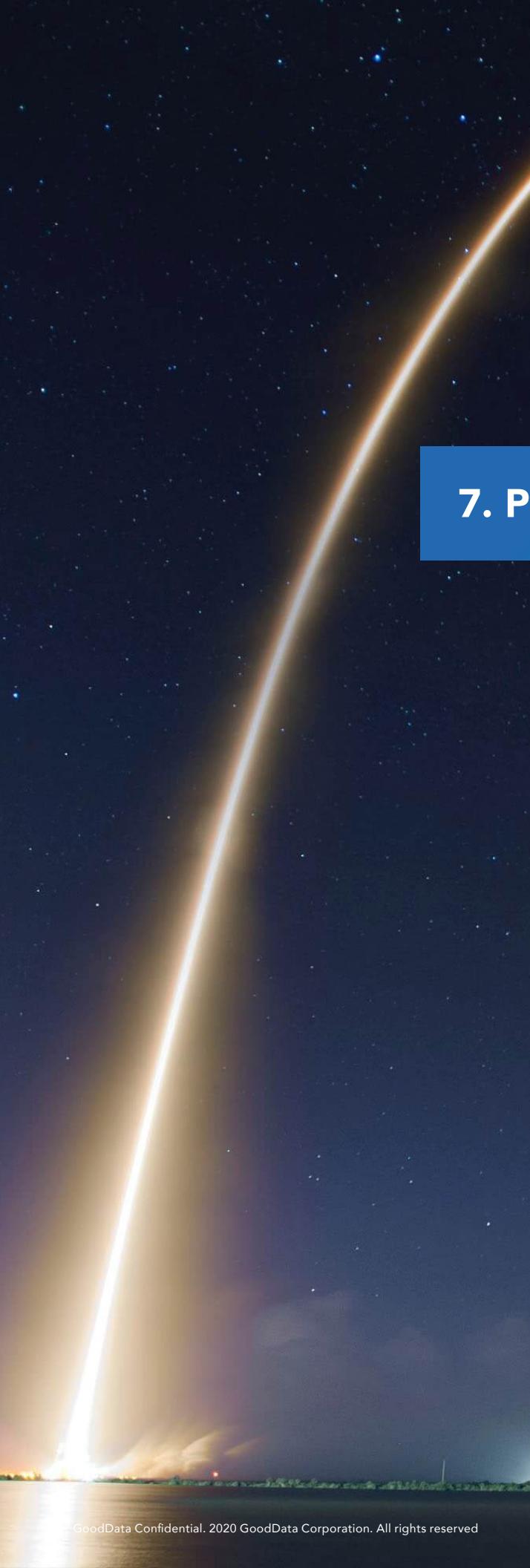
This is where the rubber hits the road. You've built it, but will they come? Sales needs to be trained and armed to do battle on your behalf:

► **Awareness training:**

At this stage, your goal is to prepare sales with a broad overview of the project, covering topics like who the users will be; what analytics you'll be delivering; how they'll be accessed; and what features you'll support (drill down, add filters, etc.). You'll also want to provide sales with a rough timeline so they can start seeding the market.

► **Feature and functionality training:**

Your objective isn't to turn your salespeople into experts but rather to be sure they can articulate core features and benefits and are ready to give a customer demo. You'll cover metrics, charts, dashboards, and UI during this session.



► **Process training:**

The final step before launch day, this is where you'll go over the details on how to turn a conversation into a sale. These details include ordering processes; service-level agreements; customization of dashboards, metrics, and charts; contracts; and support. You can also use this time to make sure they all have the right materials, including updated price sheets, new forms, and demo logins.

By now, your whole organization should be rallied and ready for launch. It's time to take it into the home stretch.

7. Prepare for launch

You're full of excitement and anticipation, but there's just a few more decisions to make until you're ready for launch! The suggested framework below will help take any guesswork out of the final stretch, so you can chart a clear path to success.

Get ready to rollout!

At GoodData, we suggest a staged rollout, which gives you the opportunity to test the product, get feedback, and make adjustments with a smaller customer sample before introducing the new product to your entire base.

Beta rollout

This is where you'll test the performance of not just the product but also the processes that support it. Better to experience a snag with three customers than 1,000, right?

► **Contracts:**

Were there any issues with the pricing or terms? How long did the process take?

► **Onboarding:**

Was it smooth? Were you able to provision the environment, implement single sign-on, and issue logins as planned?

► **Performance and scale:**

Did you stress-test the environment to ensure that it can handle customer load?

► **Data loading:**

Did you get customers' data loaded into the system without challenges?

► **Metrics:**

Do the metrics deliver on your personas' needs? Is user discovery intuitive? Is the data displayed in a useful way?

► **Training:**

Do customers understand how to use the new capabilities? How is adoption going?

► **Feedback:**

Are you receiving regular feedback on the new capabilities? How are you capturing that?

Make sure you pick the right customers for the beta phase. Don't just look for a good representative sample of your different customers' business sizes or industries; search for customers that will test the "edges" of your solution. For example, the customer with high analytical needs that requires a diverse range of metrics, or a customer with limited technical expertise who may be new to analytics and reporting. By combining both a high-needs and low-needs use case, you'll learn more than by choosing the "average" customer.

Keep the number of customers relatively low for the beta, so you can spend more time with each. Doing so will also help ensure you've got the right resources available to address any potential challenges as they emerge.

Phased general rollout

To keep the rollout as manageable as possible, you should consider dividing it into four stages, grouping customers by region, product line, or randomly. Unless there are special circumstances (prior commitments or unusually challenging customers), make it a point not to group customers by size, revenue, or maturity, which can cause unintended financial or support impacts.

Once you have your segments dialed-in, your account representatives will need to make sure they are fully prepped on:

- **Launch timeline**
- **Expected downtime**
- **Contract changes**
- **New functionality**
- **Analytics access**
- **Support requests**

Shortly before launch, do one more spot check to make sure everything and everyone is still aligned, so there are no surprises.

Define what success looks like

When your rollout is complete, how will you know if you've succeeded? This may be one of the most important questions you ask (and answer) during your GTM planning process. Common launch metrics or KPIs might include:

- ▶ [Time to onboard a new customer](#)
- ▶ [Customers onboarded to date](#)
- ▶ [Adoption rate vs. expectations](#)
- ▶ [Customer issue rate](#)
- ▶ [Customer usage rates](#)

Analyzing your own analytics (usage, adoption, etc) is a great way to make sure all your key stakeholders are in the loop and have a full understanding of where you're winning and what the opportunities for improvement are.

You can also set up tripwires to notify you when something is off so you can take corrective action. Tripwires allow you to set acceptable boundaries related to your KPIs that set off alarms when they are exceeded, helping you hold your product up to the standards you know your customers expect.

Whether it's a KPI or a tripwire, make sure you've got a centralized "launch control" team established that can take fast action. Include stakeholders from each of the departments represented at your kickoff, establishing ownership and investment right up front. Set follow-up points where the group should come together to survey progress and readiness-and where the launch leader decides if the team is ready to move forward or will need to delay until the identified issues can be corrected.

Choose a qualified partner

The final and perhaps most important element to creating and delivering a successful embedded analytics product is choosing the right partner. GoodData offers an industry-leading business intelligence platform and deep go-to-market expertise. The GoodData platform helps ISVs quickly create new revenue streams and increase customer retention by embedding advanced self-service analytics directly into existing SaaS solutions.

GoodData delivers the market's most complete end-to-end multi-tenant analytics distribution platform, coupled with a hyper-intuitive user interface that your customers will love. With fully responsive embedded analytics customized to meet your (and your customers') brand and business needs, GoodData guides users to fast discovery and adoption using interactive recommendations.



Additional resources

[Request a custom demo](#) to talk to our experts, or start directly for free. It's easy to do so without the risk of a substantial upfront investment. With [GoodData Free](#), you can get started with our self-service analytics platform today at no cost.

The Free pricing tier offers the same benefits that you expect from a modern analytics platform, such as the ability to:

- ▶ Create dashboards simply by dragging and dropping
- ▶ Discover insights to fuel your business
- ▶ Embed dashboards into any web and mobile application
- ▶ Integrate with data warehouses like Redshift, Snowflake, and BigQuery or upload CSVs

When you sign up for GoodData Free, you get five workspaces and 100 MB/workspace for an unlimited number of users. You can continue to use GoodData Free for as long as you like, and our support team is available for whatever you need. If at any point you'd like to take your analytics to the next level, our team can guide you through the process of transitioning to our Growth or Enterprise tiers.

If you'd like to discover more about embedded analytics and the GoodData platform, we have a number of additional resources available.

Learn more

Visit GoodData's [embedded analytics](#) website to learn more about different types of embedded analytics, solutions, benefits, and additional customer success stories.

Technical paper

For technical professionals, [this paper](#) is designed to give you an overview of exactly how GoodData is able to deliver powerful analytics to massive audiences while still being the most cost-effective platform on the market. It explores how our modular platform provides the tools, runtimes, and storage for data ingestion, preparation, transformation, analytic queries, data visualization, and application integration.

Platform trial

For data engineers who'd like to see the platform in action, take a look at [our platform trial](#). With the trial, you can learn how to build and deliver powerful analytics, including creating reusable, context-aware metrics for business users.

GoodData.UI and Live Examples

For UI developers looking to explore more in-depth information on how the GoodData platform can be customized, take a look at [GoodData.UI](#). By referencing our JavaScript library, you can take a look at how you can easily customize visualizations and build applications with our ready-made and custom React components.

Get more information

Have a question or want more information that we didn't cover here? Our team is happy to [schedule a call](#).