

Time to Up-level Your Homegrown Analytics?



 insightsoftware

Contents

Introduction	3
5 Signs It's Time to Replace Your Homegrown Analytics	4
3 Steps to Up-leveling Your Analytics	9
Future Proofing: How to Avoid a Rip-and-Replace Scenario	18





1 Introduction

Does This Sound Familiar?

When you first added analytics functionality to your application, end users were thrilled. With the help of low-cost (or free) component libraries, you were able to quickly build simple charts, graphs, and dashboards, and customers were willing to pay more for these insights.

Fast-forward to today and your basic analytics functions are no longer cutting it. Maybe your end users have started demanding new features—filling your development queue with a ton of ad-

hoc improvement requests. Maybe your competitor is already offering more advanced capabilities and you're starting to feel it in customer churn. You're now forced to play catch up to the latest innovations.

Or perhaps the pain is coming from your development team: Your application's embedded dashboards are too difficult to manage, or the analytics won't scale the way they need it to for future growth. Either way, it's stealing your developers' time away from iterating on the core product.

What Could Bad BI Cost You?

If you built your original analytics in house, chances are your limited reporting will inevitably become antiquated. Continuing to let that insufficient solution languish could ultimately put your application at risk. Here's how:

- **Frustrated Users:** Bare-minimum analytics capabilities are no longer enough. End users want more than simple charts and basic data visualizations.
- **Frustrated Developers:** Advanced analytics capabilities aren't just about pleasing your end users. By adding sophisticated features such as embedded self-service analytics, 64 percent of companies have been able to reduce the number of ad hoc requests that created backlogs for their developers.
- **Dismal User Engagement:** Over 80 percent of application teams say they were able to increase the time spent in their applications just by adding or improving the embedded analytics.
- **Missed Revenue:** Embedding sophisticated analytics capabilities (not just basic dashboards and reporting) means companies are 84 percent more likely to charge more for the analytics in their applications.
- **Lost Ground to the Competition:** Seventy-eight percent of software vendors offering paid commercial applications charge more for their embedded analytics. For companies that don't charge for analytics, 55 percent said they can't do so because their competition already has a stronghold—and they need analytics either to keep up or catch up.

Updating and maintaining the code to your existing analytics may seem easier than searching for a different solution. However, delaying an inevitable decision could cost you much more than you think in the long run. By staying on the “build” track, you commit to staffing resources in development, support, and keeping up with modern feature requests from your customers.

We get it. The build vs. buy decision is never easy. But by following the advice in this ebook, you can make this intimidating process as simple as possible.

You'll Learn:

- Signs it's time to replace your homegrown analytics
- How to avoid a future rip-and-replace scenario
- How to future-proof your analytics

2 5 Signs It's Time to Replace Your Homegrown Analytics

Outdated analytics features make it impossible to keep your end users engaged, upsell your existing customers, or win new ones. And because it's a pain for your development team to manage, it affects the rest of your product—taking resources away from revenue-driving innovation elsewhere.

How do you know it's time to replace your embedded analytics? Look for these 5 signs:

1. Demand for New Capabilities

If your users are constantly clamoring for new capabilities, you're likely facing increased customer churn.

Basic dashboards and reports will only take you so far. Today, end users are demanding sophisticated features such as embedded self-service analytics, which empowers them to ask new questions and explore their data for unique answers without regular assistance from development or IT. They also want analytics to work with their other tools, supporting capabilities such as write-back (which lets them update information in the application's source systems without leaving the analytics interface) and workflow capabilities (which drive action by letting users kick off a workflow from your host application).

But if your outdated analytics won't support modern BI capabilities—or at least won't do it without a ton of pain, like you building it all from scratch—it may be time to find a third-party tool.



2. Painful Scaling

Embedded dashboards are not a “once and done” project. As your end users demand more features, your development team will struggle to keep up. It may be fine if you have 10 people using your product, but what if you have 100 or more users? If you continue to build analytics with components, you’ll be forced to research and develop each new capability, one at a time. This will unavoidably delay every product update. Every new feature takes exponentially more time and resources to deliver.

3. Security Inefficiencies

Old security is risky security. In addition to the risks it poses to your data, outdated security integration means that user management can quickly snowball for your development and IT teams. There’s no way to globally manage security with components, which means you’ll have to implement and maintain security separately and consistently for every component you use. You also have to ensure the components don’t have security risks themselves.





4. Time

This is the one that resonates with most software teams—you're out of time. Time spent building something outside of your core competency is time taken away from working on your core application. Updating and maintaining your homegrown analytics requires a significant amount of time. Every time you need to update or change your analytics, you have to refactor or recode each component one at a time.

"From a dev point of view, we were struggling to make it work," recalls Maurice Davidson, Senior Developer at Youmanage. "Every time we had to build out a custom report for a client, or when something went wrong, it would take forever to fix because we had to go back in and relearn the code base before we could make any changes. It was a lot of time spent relearning the system or playing around with it trying to make it work instead of adding value."

"Every time we had to build out a custom report for a client, or when something went wrong, it would take forever to fix because we had to go back in and relearn the code base before we could make any changes."

Maurice Davidson,
Senior Developer at Youmanage



5. Cost

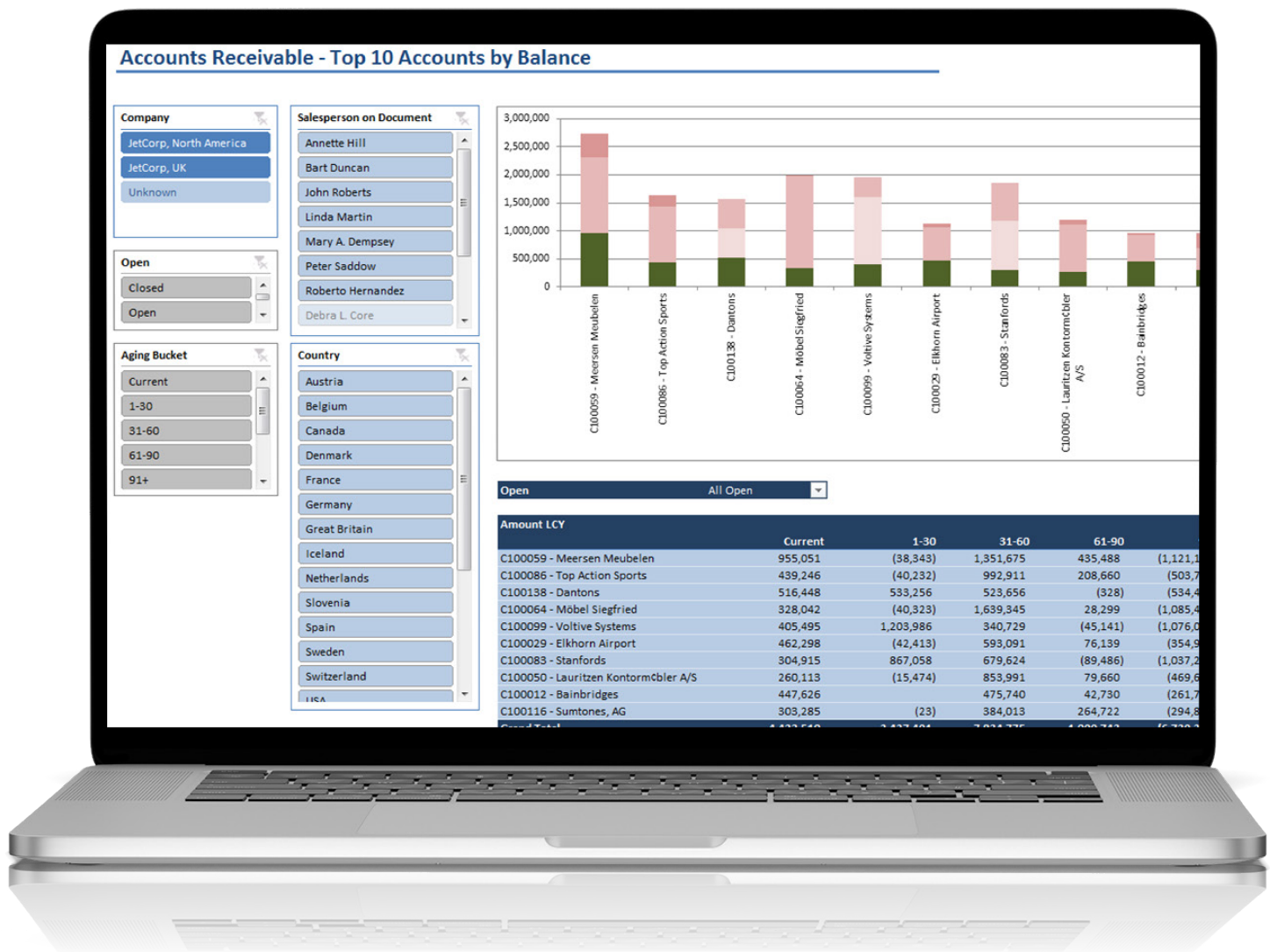
If you stay on the “build” track, cost will unavoidably become an issue over time. Since underlying technology is constantly evolving, the upkeep of your application’s analytics component adds to the investment. As your application hits the market, you will inevitably need to allocate time and resources to maintaining and adapting your software to ensure usability and stability.

And if you’re building with component libraries, you really have no guarantee that they will work together forever or have consistent versioning. Components are notorious for not being backwards compatible so a single upgrade may lead to regression issues.

In Short:

Does your application fail to offer deeply embedded analytics, empower users with self-service capabilities, or provide sophisticated capabilities that keep users in your application? Are you constantly hearing complaints from your engineering team about how difficult maintenance is, or facing prohibitive costs whenever you try to scale? **If yes, you’re being held back by outdated analytics.**





2 3 Steps to Up-leveling Your Analytics

Now that you've decided whether or not you're replacing your homegrown analytics, it's time to develop a plan of action.

Every Build to Buy Transition Will Have Three Phases:

1. Define your business needs
2. Map your business needs to requirements
3. Map your requirements to vendor capabilities

Phase 1:

Define Your Business Needs

Without a solid understanding of your business requirements, your new BI tool will inevitably falter and fail. Start this process by talking to your end users and your application team—including developers, engineers, product managers, and IT personnel.

Talk to users. One of the most common mistakes of any analytics project is not taking the time to talk to end users. Trust us: It's worth putting in the effort up front to thoroughly understand what your users need so you can translate that into your final product. Often, what you think your users need and what they actually need are not the same.

Talk to Your End Users to Determine:

- How do they prefer to work?
- In what context will they need the information?
- How urgent is that need?



Talk to developers. Next, talk to your development or IT team and find out what they need from an analytics platform. It's important to do this early in the process. All too often, companies ask users what they want, then map requirements and search for vendors—without asking their developers what they want or what's feasible with your application.

Talk to Your Application Developers to Determine:

- What do they like and dislike about the current analytics in your application?
- What do they need from an analytics platform?
- What can they do in terms of implementation and deployment? Just because you or your customers want something doesn't mean it's feasible.
- What considerations do you need to keep in mind for your current application architecture (security, data, deployment, etc.)? The best analytics tool won't force you to change everything else—which means you'll need a flexible platform that enables you to keep (rather than change) the investments you've already made.

Phase 2:

Map Business Needs to Requirements

Now that you know what your analytic application needs to look like from both the end-user and developer sides, consider what capabilities will help you meet those needs.

Start by exploring what current BI tools can do:

- What are the typical requirements for an analytic application?
- What common capabilities do most BI platforms have?
- Are there any unique capabilities you need to add—either through customization or with your new vendor's support?



Consider Today and Tomorrow

It's not just about what capabilities you need now. Give careful consideration to how you'll keep up with user demands in the future. Choosing a scalable analytics platform with extensible elements and a rapid development environment will help you build a great solution for your current needs—and ensure you won't have to rip it out in a few years.

Have a Commercial Application?

Don't skip this step. Commercial software and SaaS providers need to go beyond what features their customers want now and consider which analytics capabilities will set your product apart and drive more revenue from customers. Features like basic dashboards and interactive reports may have been enough to justify charging a premium five or 10 years ago, and you'll likely still need to offer those for free. But you should also consider more sophisticated capabilities that will help you win customers now and into the future.



#1 Mistake: Failing to Validate Your Requirements and Capabilities

There's no sense doing all this work in a vacuum. Once you feel like you understand your business requirements and what capabilities you need to meet those requirements, validate your plan with your users and developers.

- **Validating With Users:** Present mockups to your end users and ask them: "If this was the solution—with this look, and these capabilities—what would you think?"
- **Validating With Developers:** A solid mockup isn't as important for your developers, but you should still have a conversation with them. If you skip this step and go right to vendor selection, you risk choosing a solution your developers simply cannot build.

Which advanced capabilities will make your application stand out? Consider features such as:

- Write-back, so users can update the application's source systems without leaving the application
- Workflow integration, which allows users to take action or kick off a process from their analytics interface
- Mobile responsiveness, so your application's dashboards instantly look great on any device
- Embedded self-service, so end users can explore their data for unique answers without help from your team
- Tailored experiences for each user based on their roles and rights

Phase 3:

Map Your Requirements to Vendor Capabilities

By now, you've laid some invaluable groundwork:

- You understand your business requirements
- You know which capabilities you need to offer to meet those requirements
- You've validated those requirements and capabilities with your end users and development team

Now it's time to find your perfect analytics match. Most vendor evaluation processes share a few common steps:

Establish Your Timeline

Identify the steps you'll take to reach your goals. These steps may vary based on your company and your timeline, but in general, start by asking yourself, "When do I want to..."

- Begin the selection process?
- Schedule detailed vendor demos?
- Finish a proof of concept?
- Make my final decision?
- Start development and QA?
- Release a beta solution?
- Release the final product?

Expert Tip: Listen to your instincts. Vague answers about roadmaps and elusive ideas about how to meet your custom requirements won't cut it. If you're talking to a vendor and you're unclear on something, keep pushing until you have a concrete understanding of what you're getting into—and don't be afraid to refocus on other solutions if needed.

Assemble Your Team

Determine the stakeholders who need to be involved. You've likely already consulted them as you've determined your business requirements and desired capabilities. Consider who is going to care about embedded analytics internally (your executive team, product management, and lead developers) and externally (your key customers and customer advisory board). Don't forget to involve your development operations team in these assessments.

Research Potential Vendors

Assign a point person to research potential vendors and evaluate whether their functionality matches your requirements. For embedded analytics, many companies start with independent industry resources such as the Dresner Advisory Services Embedded Business Intelligence Market Study.

Narrow down your list to the top two or three vendors by attending free demonstrations, holding calls with sales teams, and participating in free trials.

Complete Technical Evaluations

Begin a structured evaluation process with each of the vendors on your shortlist. This is where you'll define a proof of concept and establish clear-cut guidelines for what you want to accomplish (within a reasonable timeframe).

Always implement the proof of concept in a technical environment that's as close to your production environment as possible. This means it should be connected to your data sources, integrated with your security, and embedded in your application. If you host a SaaS application in the cloud, don't evaluate desktop tools or run analysis off a cleansed spreadsheet (unless that's what you expect your users to do).

Look for Support

Your application team and developers should have access to extensive, accurate resources whenever they need them. Be wary of vendors who say they have documentation, communities, and professional support resources, but fail to show you what that support looks like before you buy. And if you're planning to rely on any open-source analytics or components, know that relying on the market's online community can sometimes create support challenges. Look for a solution with plenty of documentation and experts who are always ready and able to help with your specific issue.



Define a Proof of Concept and Establish Clear-Cut Guidelines





3 Future Proofing: How to Avoid a Rip-and-Replace Scenario

Now you have a plan of attack and a good idea of who your dream analytics partner will be. To avoid a rip-and-replace scenario in a year or two—follow these best practices.



Ask for Help

If you're on a tight deadline, look for third-party support, such as professional services, from your vendor. They can help train your people, determine your architecture, come up with a data strategy, and/or build a custom feature. All in all, this support will help you get up and running faster than other solutions.



Consider Existing Skill Sets

We've said it before, but it's worth repeating: Make sure your development team can actually use your chosen BI solution. Every platform is a little different, so involve developers early and get them into the weeds as soon as possible. Do they have the skills to support an intricate development environment?

If not, you may need to find a different solution or compensate for any missing skills with new team members or a third-party support team.



Don't Forget Data Architecture

In BI, data is everything. Consider where your data lives and make sure it's accessible for the new tool. Also think about what will happen to your data if you need to rip and replace again in the future.

Most successful data strategies come from a partnership between you, your BI vendor, and your data consultant (if you choose to have one) or internal data expert. A solid data strategy means your data is stored in a location that's accessible for reporting, and that your developers understand the data model and know how to query it to build reports.

In some cases, you may be able to bolt your BI solution onto your current data model. But in others—such as when you're replacing a proprietary solution that ingests your data with it—you will need to re-architect your backend for the data layer. If you don't already have a data warehouse, building one before you tackle the BI switch will make the transition significantly easier.



Involve Users Early and Often

We've all seen what happens when you don't follow this practice: poor user adoption. Your solution could be amazing, but you can't expect your users to touch it if you don't vet it with them first. Ask yourself what it will take for your users to learn the new tool. Are you building something that matches well with their skill sets? What resources, training, or documentation will you need to ensure their success?

A common complaint we hear from customers is, "Our users hate this." But that's rarely the real case. Once you dig in and talk to your users, you may discover they find the tool difficult to use or wish it had single sign-on—or any number of other pain points that can easily be addressed if you ask users for feedback from the beginning (as we discussed in Chapter 2).



Plan Your BI Roadmap

Your analytics timeline doesn't stop at launch. It also includes iterating on future versions. Think far ahead and establish a well-defined process for what and when you need to deliver each of your business requirements and capabilities. The right solution should support the latest capabilities and give you the flexibility to meet challenges you don't even know you have yet. This is what we call "future-proofing."

So how do you future proof your embedded analytics? Look for:

- Robust customization options (so you're not locked into your vendor's rigid experience)
- Ability to connect to a slew of different data sources (not just the ones you have today)
- Scalability (extensible elements and customization options that allow you to create whatever you need)
- A rapid development environment and high-performance data store

Logi Embedded Analytics: Purpose-Built for Software Teams

Product teams need intuitive analytics and data visualization capabilities in their applications, purpose-built for every user's unique role and skills. Logi's embedded analytics solutions, by insightsoftware, empower you to design and deploy analytics into the fabric of your organization and products. These analytics integrate with your existing workflows and security models providing a seamless experience where anyone can analyze data, share insights, and make informed decisions.

Learn more at insightsoftware.com/logi-analytics/.

About insightsoftware

insightsoftware is a leading provider of reporting, analytics, and performance management solutions. Over 30,000 organizations worldwide rely on us to support business needs in the areas of accounting, finance, operations, supply chain, tax, budgeting, planning, HR, and disclosure management. We enable the Office of the CFO to connect to and make sense of their data in real time so they can proactively drive greater financial intelligence across their organization. Our best-in-class solutions provide customers with increased productivity, visibility, accuracy, and compliance.



insightsoftware

US +1 919 872 7800

UK +44 (0) 845 467 4448

AU +61 2 8985 7777

insightsoftware.com