



Embedded Analytics:

The Future of Business Intelligence

Wayne W. Eckerson
April 2016



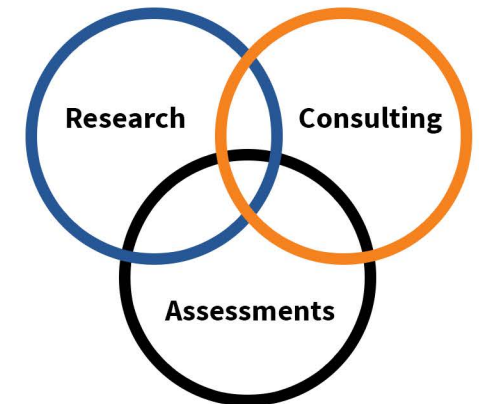
About the Author



Wayne W. Eckerson has been a thought leader in the business intelligence and analytics field since the early 1990s. He is a sought-after consultant, noted speaker, and expert educator who thinks critically, writes clearly, and presents persuasively about complex topics. Eckerson has conducted many groundbreaking research studies, chaired numerous conferences, and written two widely read books on performance dashboards and analytics. Eckerson is the founder and principal consultant of Eckerson Group, a research and consulting firm that helps business and analytics leaders use data and technology to drive better insights and actions.

About Eckerson Group

Eckerson Group is a research and consulting firm that helps business and analytics leaders use data and technology to drive better insights and actions. Through its reports and advisory services, the firm helps companies maximize their investment in data and analytics. Its researchers and consultants each have more than 20 years of experience in the field and are uniquely qualified to help business and technical leaders succeed with business intelligence, analytics, data management, data governance, performance management, and data science.



Report Sponsors

The research for this report and its accompanying report, **Which Embedded Analytics Product is Right For You?**, is sponsored by the following solution providers: Dundas Data Visualization, GoodData, Izenda, Logi Analytics, Looker, OpenText, Qlik, Sisense, and Zoomdata.

Executive Summary

Business intelligence (BI) empowers business users to access and analyze data in the form of reports, dashboards, and self-service analytics. Although most organizations already provide internal business users with BI tools to improve decision making, many now are embedding analytics into core business applications to broaden the reach and improve the timeliness of insights. By maintaining the business context for data and analysis, these enriched applications close the last mile of BI by helping business people turn insights into action.

Embedded analytics is not new, but the technology for integrating charts, reports, dashboards, and self-service tools has evolved considerably in the past 30 years. Formerly, only software vendors embedded analytical tools into applications, but now organizations in every industry are doing so to cement ties with customers and suppliers, define a “digital identity”, and even monetize data assets.

Consequently, many BI vendors now target the embedded analytics market as a high-growth area and have architected their products to make them easier to embed and manipulate from within other applications. These modern BI platforms are browser-based and support rich JavaScript and REST application programming interfaces as well as integrated security and deployment frameworks that work well in cloud-based, multi-tenant host environments.



The Long View of Business Intelligence

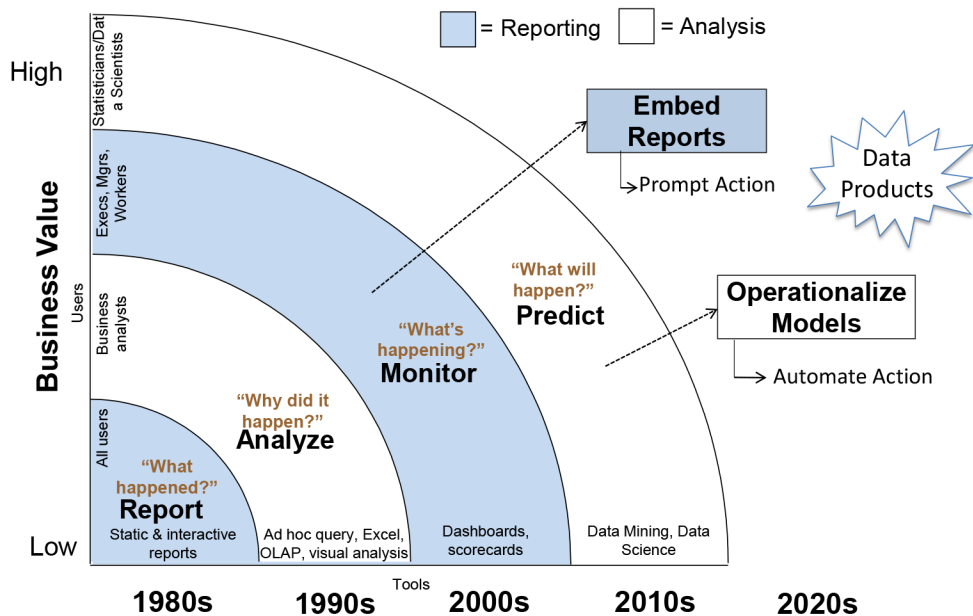
Business intelligence (BI) has long been associated with tools: business user tools for viewing and interacting with reports and dashboards, and business analyst tools for querying databases and visualizing, analyzing, and modeling the results. During the past 25 years, software vendors have shipped hundreds, if not thousands, of analytical tools that help businesspeople turn data into insights and action.

The era of BI tools may soon disappear.

But the era of BI tools may soon disappear. Or more accurately, the time when organizations purchased BI licenses for every knowledge worker may be drawing to an end. Rather than distribute analytical tools to employees, organizations will increasingly embed analytical output—reports, dashboards, insights, visualizations, and recommendations—into business applications that businesspeople both inside and outside the organization rely on to do their jobs every day. In short, the future of BI is embedded analytics. (See figure 1.)



Figure 1. The Evolution of BI—and Its Future



In the past 35 years, business intelligence (BI) has evolved in two alternating waves: a reporting wave geared to casual users, and an analysis wave geared to power users. Both waves are now converging in an embedded analytics market where organizations and software vendors embed reports, analysis and analytic functionality into applications that run the business. This convergence will help close the last mile of BI by turning insights into action.

This doesn't mean that organizations won't need to purchase BI tools. Instead, they will increasingly use them to build analytical capabilities inside core business applications in every department, including human resources, operations, finance, business development and marketing. At the same time, software vendors of all stripes—especially the new legion of cloud application vendors—are aggressively embedding analytical capabilities inside their applications as a key differentiator. Their goal is to increase the appeal and value of their applications, improve customer satisfaction and loyalty, increase application stickiness, and generate additional revenue. In some cases, embedded analytics can not only pay for itself, but return a handsome profit.

Of course, organizations will still need to distribute BI tools to their most advanced analytical users—business analysts, statisticians, and data scientists—who need a hands-on approach to data and upon whom organizations increasingly depend to create business models, predictions and forecasts as well as departmental reports and dashboards. In addition, casual users will need to consult cross-functional reports and dashboards that consolidate business activity and performance across or within departmental boundaries and alerts users to significant deviations.

Closing the Last Mile of BI

Context. The purpose of BI is to turn data into insights and action. Embedded analytics makes that possible by putting insights into the context of operational applications in which actions occur. With embedded analytics, business users no longer have to exit a business application to view results, analyze performance, and view recommended actions; they can do this inside the application itself.

For instance, when using a marketing automation tool enriched with embedded analytics, a marketer can not only execute marketing campaigns, but also view and analyze results and compare to industry benchmarks without ever leaving the application. This creates a seamless user experience that makes it easier and faster to design and execute real-time, data-driven marketing campaigns.

Embedding Predictions. Organizations can not only embed reports and dashboards into applications and portals; they can also embed syndicated data (e.g. demographics) and the output of predictive models, turning customer-facing applications into powerful tools for reducing costs, growing revenues, and increasing customer satisfaction. For instance, banks embed cross-sell models into branch applications or ATMs that prompt tellers to deliver customers personalized recommendations about other products that they might like. Manufacturers embed obsolescence models into factory automation software to predict and schedule when machines need maintenance to avoid a breakdown.

The Last Mile. Critics of BI have said that by the time business users look at reports and dashboards, it's too late to take action and change outcomes. Embedded analytics addresses that challenge, turning BI from a reactive activity into a proactive one. It helps close the proverbial “last mile” of BI—turning insights into action that help an organization achieve its strategic goals and financial objectives.



Business Intelligence

On the surface, embedding analytics into applications is simple enough. But a closer look reveals unexpected complexities. And opportunities.

Business Intelligence

What is BI? First, BI is not a simple, monolithic, or clearly defined set of functionality. As figure 1 shows, there are two major categories of BI: reporting and analysis. Those two categories each divide into two subcategories, and within each subcategory, there are numerous permutations, each designed to deliver different types of insights to different users.

To add to the confusion, different people call different categories and subcategories of BI different things. For example, BI and analytics are sometimes used interchangeably. For the purposes of this report, we use BI to describe the process of turning data into insights and action—that is, helping businesses run more intelligently using data. We use “BI or analytic tools or platforms” to describe the complete range of products that support BI, and “analytics” to describe more advanced forms of BI, such as predictive modeling.

BI Users. Before organizations embark on an embedded BI strategy, they have to identify the analytical functionality they want to embed. The only way to do that is to first identify the types of business users they want to support and then ascertain their information requirements and decision-making styles.

As figure 1 indicates, casual users (or people who use information to do their jobs) generally want reporting functionality, while power users need analytics functionality. More specifically, casual users want interactive reports and dashboards tailored to their roles that enable them to service their own information needs. Power users want self-service discovery tools that enable them to query, combine, visualize, and analyze data. This persona profiling is particularly critical when embedding analytics in commercial applications or data products accessed by external organizations and users.



BI Functionality. There are five major categories of BI functionality that organizations embed inside applications:

- 1. Charts and components.** Individual report parts, such as a query, metric, chart, table, map, or graphic with a fixed or variable-driven connection to a predefined data source.
- 2. Interactive reports or dashboards.** A collection of interconnected report components (see above) and controls that provide casual users with a guided and role-based analytic experience, promoting self-service.
- 3. Self-service analytics.** Allows power users to connect to, explore, and combine data sets, create new metrics, dimensions, and groups, and self-publish reports, visualizations, and dashboards; promotes self-service for power users.
- 4. Predictive analytics.** These include self-service analytics tools that embed forecasting, regression, and other advanced algorithms as well as data science tools and languages (e.g. R) to create advanced analytical models or recommendations.
- 5. Transactions.** BI tools that execute transactions, update forms, and write data back to a database create hybrid applications that fuse analytical and transactional features.

Applications

Most organizations support a myriad of business applications that run on different systems and platforms. Many are packaged applications purchased from software vendors, while others are homegrown, or a combination. Some run on-premises; others, in the cloud outside the corporate firewall. Some can be accessed only by desktop machines attached to a corporate network, while others can be accessed via a Web browser or mobile device from anywhere in the world. In addition, some run on Unix or Linux, while others run on Windows servers or desktop machines. Some are designed using .NET technology, while others use JavaScript frameworks, such as AngularJS.

Before embedding analytical functionality, developers need to understand the nature of each host application and its infrastructure. This often defines the tools and skills required to execute the integration and whether they need outside help to deploy a blended application.

Embedded BI Functionality

Finally, organizations need to settle on a method to embed BI functionality into a host application. The way developers have embedded BI software has evolved significantly over the years (see next section). The advent of the Internet, Web browsers, and modern programming and packaging standards have greatly simplified the process. Organizations no longer have to deploy executable files into a desktop or client/server application and install it on users' desktop machines.

Today, developers can integrate disparate applications running on different servers and domains using application programming interfaces (APIs) that enable one application to communicate with another across a secure HTTP network (i.e., the secure Internet). Most BI vendors support modern APIs such as JavaScript and REST, which enable applications to dynamically access the entire slate of functions from a given BI tool or platform. Most also integrate with application security frameworks, easing access to BI functionality via single-sign-on privileges. (See **Which Embedded Analytics Product is Right For You?** for help selecting the right product to embed.)



Evolution of Embedded BI

The embedded BI market is not new; it’s been around as long as there has been BI software. But the way BI has been embedded into applications has changed immensely since the mid-1990s when Crystal Reports was the de facto report tool that every software vendor, including Microsoft, bundled into its commercial software. (See table 1.)

BI Features. The level of embedded BI functionality has evolved from static reports in the 1990s to interactive reports and dashboards in the 2000s to self-service, predictive, and blended analytics today. In the 1990s, BI tools had their own look and feel and were a distinct module within most application packages. By the 2000s, BI tools could be configured to adopt the look and feel of the host application. Today, developers can create completely custom front ends or applications that run against the BI product’s application server.

Data. Moreover, BI tools can now access a much broader range of data than relational databases, which were the predominant data source in the 1990s. In the 2000s, BI tools could routinely query OLAP and XML sources, and today, many support a bevy of cloud applications, such as Salesforce or Zendesk, and big data sources, including Hadoop, NoSQL, server logs, event streams, search indexes, and cloud file systems, such as Amazon S3 and Google File System, among others.

Table 1. Evolution of Embedded BI

Decade	BI User Features	Data Sources	Platform	BI Software	Client Code	APIs	Developers	Pricing
1990s	Static Reports	RDBMS, files	Desktop	Desktop BI tools	Windows, Unix	Code-specific libraries	Independent software vendors (ISVs)	User-based
2000s	Interactive reports, OLAP, dashboards	OLAP and XML	Web	Web BI tools	ActiveX, JVM, Flash, Silverlight	SOAP, iFrames	ISVs and internal developers	Server-based
2010s	Self-service, predictive, and blended analytics	Cloud apps, big data, NoSQL, streams, search	Cloud	BI platforms	AngularJS, Ember, js jQuery, AJAX, etc.	REST, JavaScript,	ISVs and internal developers	Value-based

Platform. BI software has moved from the desktop to the Web and now to the cloud, where organizations can rent the software on a monthly (or sometimes hourly) basis. Web and cloud-enabled BI applications run on separate servers and thus share no code or libraries with the host applications.

This eliminates versioning considerations and allows administrators to tune performance separately.

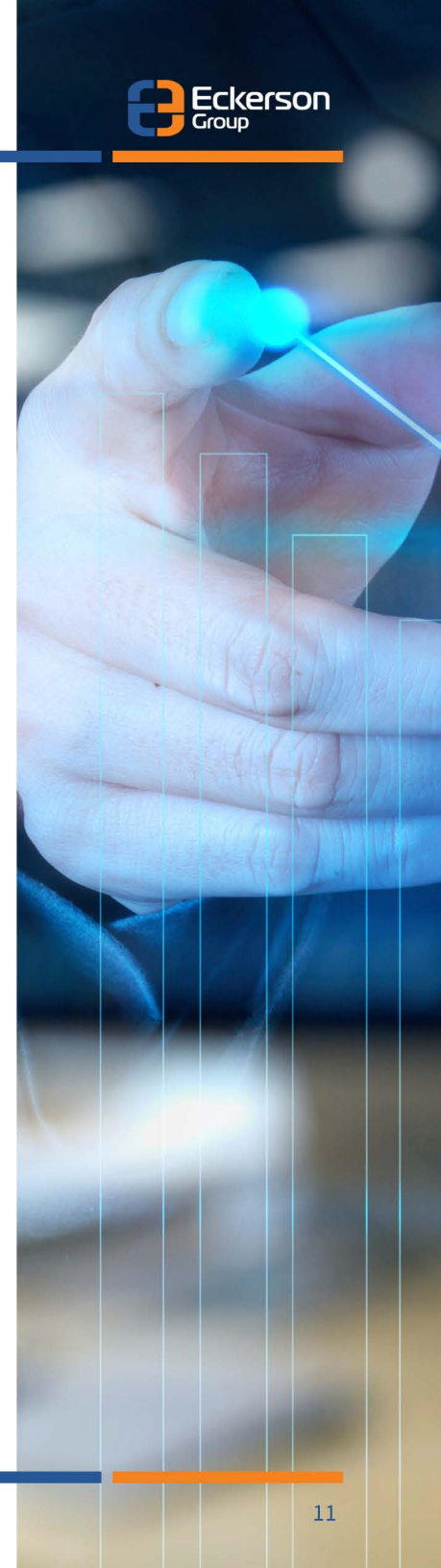
Cloud. With the cloud, BI products need to work within a multi-tenant environment that allows a host application to provision, configure, administer, and secure BI software for each supported tenant and their specific data. The cloud also gives BI vendors the opportunity to offer their products as software-as-a-service (SaaS) or platform-as-a-service (PaaS) applications that support multiple customers or divisions on a single instance of the software, achieving economies of scale and balancing central governance with local flexibility. Other cloud advantages include automatic software updates, up-to-date security, elastic scalability, and predictable licensing.

BI Software. BI software itself has evolved from packaged tools in the 1990s to Web and open source software in the 2000s, then to BI platforms and cloud-based tools today. With the advent of the Web and HTML pages, BI vendors struggled to achieve the interactivity and performance of desktop and client/server applications. Client-side code evolved from ActiveX and Java virtual machines in the early 2000s to client-side plug-ins, such as Flash and Silverlight, in the mid-2000s. Today, many BI products make heavy use of client-side scripting languages, such as AngularJS, React.js, and Ember.js, and other technologies, such as jQuery, AJAX, and Node.js, Bootstrap to create a more dynamic, responsive Web and mobile user experience.

BI Platforms are infinitely customizable and extensible and can be 'beheaded'

APIs. Although customers can still purchase off-the-shelf BI tools, many BI vendors are shipping BI platforms, which come with a rich set of APIs and cloud-based deployment features that make them ideal for embedding into other applications. With BI tools (including open source), what you see is what you get—you must embed the entire tool, including the graphical user interface (GUI). But BI platforms are infinitely customizable and extensible and can be “beheaded.” That is, customers can ditch the BI platform’s packaged GUI and write their own that exploits the BI platform’s analytic functionality.

APIs have evolved from client/server programming libraries (e.g., COM) to Web-based interfaces such as SOAP, iFrames, and client-side plug-ins, to modern standards such as JavaScript and REST. Many BI vendors, especially the sponsors of this report, supply JavaScript APIs that make it easy to create new charts and graphics, modify existing ones, and





embed BI visualizations into other Web applications. In contrast, REST APIs are generally used to access back-end administrative functions, such as publishing, provisioning, scheduling, and user administration.

Developers. In the past, commercial software developers did most of the embedding, using tools such as Microsoft Visual Basic and Visual Studio, to insert static reports into applications. Today, enterprise IT developers have followed suit, embedding analytic functionality to meet the needs of business users. In short, embedded analytics has become pervasive.

As executives recognize the value of data, they now want to monetize it by developing data-rich applications for customers and suppliers.

Although software vendors still comprise the lion's share of organizations implementing embedded BI, corporate developers are catching up. As executives recognize the value of their data assets, they now want to monetize them by developing data-rich applications for customers data-rich applications for internal and external customers. This typically involves embedding reports, dashboards, self-service analytics, and predictive models into tiers of outward-facing applications.

Pricing. Pricing has evolved from the days when software vendors bundled a free copy of Crystal Reports with their software for up to five users or a single data source. In the 2000s, embedded BI software adopted enterprise pricing based on the size of the BI server, given the broad deployments of many embedded BI applications.

Today, software pricing reflects the influence of the cloud and software-as-a-service offerings. Subscription-based pricing is mostly monthly or annual, but some hourly subscriptions are available on public cloud sites. And some vendors offer pay-per-use or a pure utility pricing model. Some embedded BI vendors are moving to value-based pricing, which means they try to align with a customer's business model for monetizing the software. Generally, their goal is to minimize up-front costs until the customer can monetize the software, then provide predictable pricing that aligns with expected value. This can be defined by software sales, number of users, data sources, application functionality, or query activity.

Additionally, pricing should facilitate a customer's external pricing strategy. Software vendors that embed analytic functionality can either provide the software at no extra charge to their customers in an effort to "sweeten" the value of the core application, or they can offer it as an add-on product for an extra charge. A third option is a tiered pricing strategy in which the software vendor allocates analytic functionality across several pricing tiers, such as Free, Basic, and Premium. This requires an analytic product with granular access control and APIs that make it easy to provision functionality based on pricing tier.

Build or Buy? That Is the Question

A major question is whether developers should build or buy analytical functionality to embed in their applications.

Suppose you are an application developer who wants to include a few charts in your application or provide a complete catalog of reports or dashboards that will allow customers to analyze their activity from any perspective. Or maybe you want to give users the ability to explore the application's data in an ad hoc fashion, and perhaps mash it up with their own local data, and subsequently, create their own reports and dashboards.

The question a developer must ask in all these scenarios is: "Should I build the analytical functionality myself or buy it from a established BI vendor?"

Scale and Scope. Certainly, if developers only need to embed one or two static charts into a Web site or build a single report or dashboard, the odds are that they will build it themselves. But what starts as a quick side project often explodes into a full-time job with a never-ending set of requests for more data, more reports, better dashboards, and self-service discovery features. Before long, many developers opt to purchase a commercial BI tool and adapt it to their environment.

The decision to Build or Buy BI software to embed in an Application (or suite of applications) is a multi-faceted one and requires a thorough evaluation

The decision to build or buy BI software to embed in an application (or suite of applications) is a multi-faceted one and requires a thorough evaluation. (To conduct a self-assessment, see Eckerson Group's online tool **Embedded Analytics: The Future of Business Intelligence**. The decision depends on business and user requirements, developer skills and resources, the culture of the organization and its competitive environment, the functionality of available BI products, and finally the budget.

Ultimately, there is no right or wrong answer. Some organizations have a build-only mentality, while others commit to using only packaged software. Most fall somewhere in between and both build and buy. Some start by building analytical functionality and then convert to a packaged BI tool. Others buy an analytical tool and then heavily customize it to support key features and requirements. Every organization needs to weigh the pros and cons of each approach before making a decision.

Why Build?

There are several reasons why an organization might want to roll out its own analytic functionality. Control is perhaps the biggest factor: organizations get exactly what they want and nothing more or less. When they add analytic functionality and tightly integrate it within the workflow of an application, they can deliver exactly the user experience they want, assuming they have developers inhouse with the skills to develop and maintain analytic functionality. Or the analytic tool may lack key features that have to be custom built, minimizing the cost benefit of embedding a packaged tool. Or maybe the organization wants to avoid licensing and royalty costs, or doesn't want to shuttle support calls to a third party.

Why Buy?

Despite these advantages, many organizations choose to embed third-party BI software into their applications, usually after they have tried the “build” route without success. The primary reason is that BI is usually not a core competency of most organizations, and they don't want to divert developer resources toward building, maintaining, and supporting analytic functionality and tools. It's easier to outsource these capabilities to a trusted BI specialist who can keep up with the pace of user requests for the latest developments and has more thoroughly designed, vetted, and tested its products for a wider range of users in a variety of environments. Embedding commercial BI products into an application enables development teams to focus on what they do best: building the features and functions of the core application.

Best of Breed. Moreover, a buy approach gives organizations mature, best-of-breed BI software and accelerates deployment, both of which are critical in competitive environments where customers expect top-notch BI functionality from an application supplier. In addition, the BI vendor invests its own time and resources to continuously enrich the BI software, ensuring that customers always have the most modern BI capabilities. The right BI vendor can work on your behalf, promoting your (commercial) product in the marketplace and serving as a channel for additional sales. Finally, organizations buy analytic products not just to embed in external applications but for internal consumption as well.

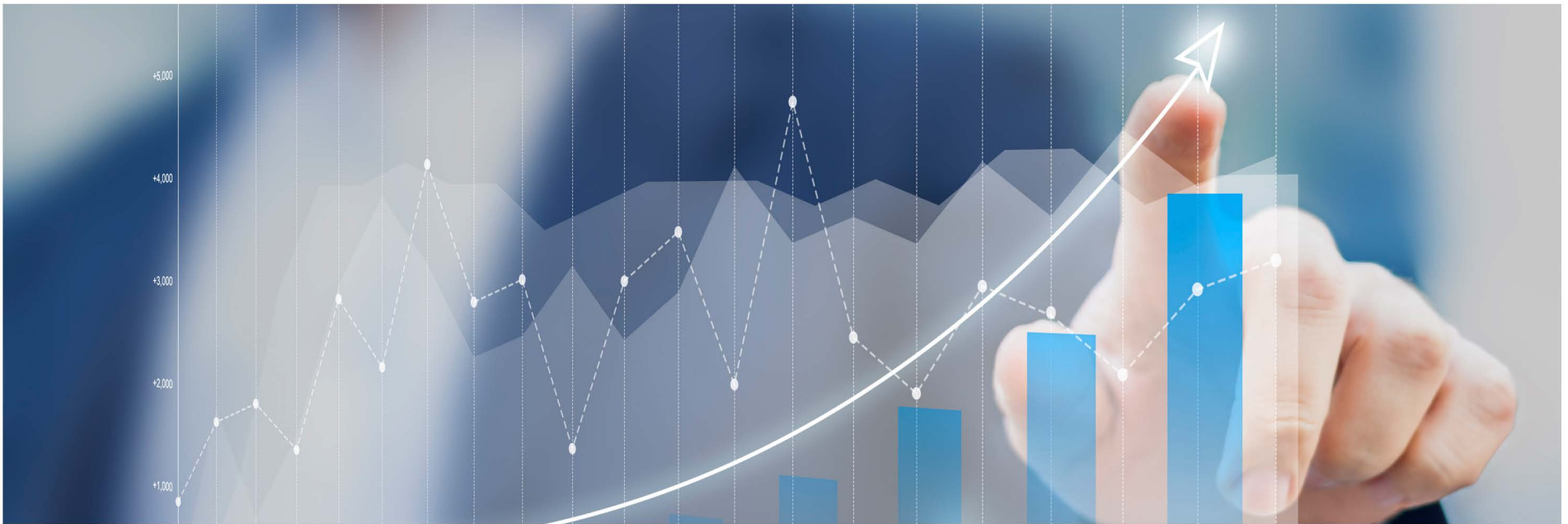


Why Build and Buy?

Often, the decision to build or buy BI software is not clear-cut. Consequently, many organizations do both. The main driver for a hybrid strategy is that no analytical tool meets 100% of requirements. Thus, developers not only have to configure the software to work in their environment, but they also have to code new functionality, ideally in conjunction with the BI vendor (who could later define automation templates or insert it into their commercial code base and compensate the customer for its contribution). Customization might entail writing new code or scripts, accessing new data sources, adding new visualizations or calculations, or building write-back functionality to the application.

Another build-and-buy option is to build a custom report or dashboard using open source or licensed component libraries. Many developers use Highcharts, ESRI Maps, and other JavaScript visualization libraries to create custom charts and graphics and embed them in other applications.

Organizations might also extend commercial BI software with custom visualizations, data connections, public data sets, or administrative functions. Many BI vendors offer open and standard APIs access and software development kits.

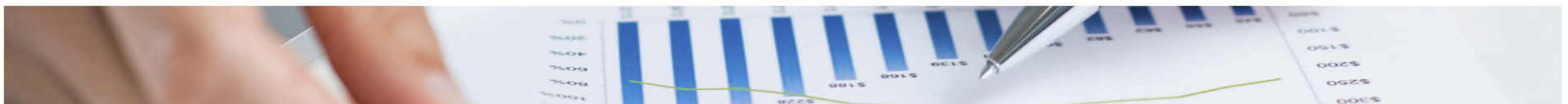


Key Success Factors

If embedded BI is the future, organizations need to understand the pitfalls and obstacles before they dive in. Below is a list of 10 key success factors for embedding analytics into an application, divided into business and technical considerations. Most apply whether an organization decides to build or buy BI functionality.

Business Considerations

- 1. Identify business goals.** An embedded BI strategy needs to align with your organization’s goals for the application. Should it increase revenues? Reduce costs? Improve customer satisfaction and loyalty? Streamline processes?
- 2. Understand the role of BI in your application.** Are you building a data-driven application in which BI plays a central role and is key to its success? Or is BI a nice-to-have or seldom-used feature that customers can live without?
- 3. Understand your competitive environment.** To what degree have your competitors implemented embedded BI? If you’re a software vendor, your competitors are easy to identify: they make competing products. If you are an internal developer, however, your competitors are other applications that compete for the time and attention of your target users.
- 4. Understand business requirements.** It’s critical that you identify your target users, what information they need, what they plan to do with it, and how those needs might change.
- 5. Develop a support strategy.** Identify and train individuals who will support the BI functionality and create suitable documentation and videos. Consider outsourcing support and documentation to a BI vendor or partner. Consider how you will offer new functionality to your analytic application over time.



Technical Considerations

- 1. Find appropriate software.** A commercial BI tool should have at least 85% of the functionality you require, and software developer kits to extend or customize the rest. This places a premium on finding a BI platform that is easily extensible beyond current requirements and can address multiple use cases.
- 2. Use rich APIs.** Make sure the software has a rich set of well-documented APIs that make it easy to call any required BI functionality from within your application.
- 3. Align security.** Make sure your BI tool doesn't expose sensitive data or force users to log on every time they want to run a report. It should integrate seamlessly with multi-tenant cloud software so you can provision, administer, monitor, and manage the BI environment from a unified administrative console.
- 4. Find a suitable partner.** If you buy BI software, select a vendor that is willing to partner closely. The vendor should have plenty of experience scoping and executing embedded BI projects and provide sufficient training and support for in-house developers.
- 5. Identify suitable pricing.** Make sure a commercial BI vendor offers flexible pricing that minimizes payment until you can monetize the software. The pricing should also be predictable, especially in high-volume environments.

This is a high-level list of key success factors for embedding analytics into homegrown or third-party applications. Before embarking on an embedded analytics strategy, make sure you have an experienced partner who can help guide you through the process.



Future of BI?

Embedded analytics is the future of BI. It presents users with data and insights in the context of the core business applications and processes they manage and use. Rather than shifting applications to view and analyze relevant data, embedded BI gives business users actionable insights within the context of their primary applications. This helps close the proverbial “last mile” of BI.

Today, organizations can embed a variety of BI functionality, ranging from individual charts and graphics to entire reports, dashboards, and self-service discovery environments. The next wave of embedded BI applications supports self-service and predictive analytics and weaves operational and analytical processes together in a single blended or immersive application.

Before embarking on an embedded analytics strategy, organizations should ask whether they will build or buy BI functionality to embed in an application. There are pros and cons to each, but given the maturity of commercial BI products and BI vendors’ current focus on the embedded market, it makes more sense than ever to embed third-party BI products into applications.

Of course, embedding a commercial BI tool isn’t easy. Even when a product meets a majority of business requirements, there is usually a good deal of planning, configuration, and coding required to get the exact look, feel, and functionality your organization requires. Thus, it’s imperative to find a BI vendor that knows how to scope and manage your embedded BI project and will work closely with you to deliver a successful implementation in the minimal amount of time.



Need help with your business analytics or data management and governance strategy?

Want to learn about the latest business analytics and big data tools and trends?

Check out **Eckerson Group** research and consulting services