MicroStrategy

Guide to Making Every App Intelligent with Embedded Analytics

This guide spotlights the role of AI-powered embedded analytics to democratize data insights across entire organizations, regardless of technical proficiency.

1. Get Started with Embedded Analytics

In the evolving landscape of data-driven decision making, integrating AI-powered embedded analytics into your software or digital platform is more than just a technological upgrade — it's a competitive necessity. Our guide covers each phase of this integration, from setting clear goals that align with your organizational needs to navigating the decisions around product requirements.

We emphasize the importance of user experience, ensuring that the analytics feel like a natural extension of the existing platform, thereby enhancing user engagement without introducing unnecessary complexity. Development, testing, deployment and ongoing support are detailed step-by-step, featuring best practices to ensure a smooth, efficient implementation.

Table of Contents

1. Get Started with Embedded Analytics
2. Set the Foundation for Integration
3. Define Product Requirements
4. Evaluate Implementation Options
5. Design the User Experience
6. Develop and Test
7. Deploy and Support
8. Learn About MicroStrategy Embedded Analytics

2. Set the Foundation for Integration

Define Your Goals for a Winning Product Strategy

Product teams are looking to enrich and differentiate products with analytics. The ability to deploy Generative AI (GenAI) capabilities creates opportunities for competitive differentiation and revenue streams. Software vendors can extend their offerings with branded custom analytics and data workflows. Organizations can deploy custom workflows within existing internal applications to create actionable outcomes and drive more value from data investments.

Consider what business goals you want to solve

- Increase revenue of existing products and services as part of the “plain vanilla” service that every customer receives.
- Create a tiered offering whereby customers pay more for the higher-level plans that include AI-powered embedded analytics.
- Raise customer satisfaction and reduce churn by adding more value to existing services.
- Launch brand new data-driven services.
- Guide your organization’s workforce to make better data-informed decisions, improve productivity and gain a competitive advantage.

Identify key questions your analytics will answer

Consider what people will learn and what actions they can take based on the data in the embedded analytics. From a differentiation standpoint, consider how this is better or different than the options that people have today. Clearly describe the purpose of the new functionality and its benefits to your target audiences.

Determine the types of data you will expose

Start to scope the types of data your AI-powered analytics will expose. Your options for data sources range from comma-separated values (CSV) and text files to databases, data warehouses, data lakes and in-memory engines as well as streaming data integration pipelines. Often embedded analytics involve read-only use cases. If the user makes an action, such as an online banking customer clicking to set up a new savings account, this can then be recorded in a separate parameter table. In some use cases, you may want to allow write-back to the data source, such as a person updating their mailing address.

Table of Contents

1 Get Started with Embedded Analytics
2 Set the Foundation for Integration
3 Define Product Requirements
4 Evaluate Implementation Options
5 Design the User Experience
6 Develop and Test
7 Deploy and Support
8 Learn About MicroStrategy Embedded Analytics

Specify the levels of interactivity and data granularity to drill down or filter

From a usability perspective, you will want to consider the levels of interactivity and data granularity to drill down or filter. Your options to allow people to interact with their data include to hover over an object to reveal some additional text or to trigger an action, filter to see a subset, click to drill down to more granular underlying data, and using GenAI to ask natural language questions in a chat box. Customers increasingly expect to find analytics and reporting as part of their everyday services. Meanwhile, many product teams face limited development time, lengthy product roadmaps, and the management of a complex code base.

Find areas for personalization through AI-generated insights or recommendations

One important nuance to consider for GenAI is whether the user prompts and answers are persisted or not. A consumer may conflate the GenAI chat box with customer support chat and give an instruction for example to update their mailing address. Many GenAI models will respond along the lines of “Thank you, I have updated your address”. If the GenAI prompts and replies are not persisted, the GenAI model will forget that the user has a new address.

Consider adding natural language chat boxes

For competitive differentiation, it’s helpful to evaluate areas for personalization through AI-generated insights or recommendations. Consider embedding natural language chat boxes with answers generated by AI. When done well, this can be an elegant addition to your product with functionality that users find appealing and natural. Stay competitive with augmented analytics, guided machine learning (ML) workflows and AI integration.

Table of Contents

1 Get Started with Embedded Analytics

2 Set the Foundation for Integration

3 Define Product Requirements

4 Evaluate Implementation Options

5 Design the User Experience

6 Develop and Test

7 Deploy and Support

8 Learn About MicroStrategy Embedded Analytics

3. Define Product Requirements

AI Prompt Engineering

For incorporating Generative AI into your embedded analytics, in most cases you will want to choose a closed language model that does not use the information shared in your application for training the large language models (LLMs) available to people outside of your organization and your customers. In addition, you may likely want to restrict the scope of the language model's outputs to prevent people from asking extraneous questions or generating inappropriate content. It's helpful to provide guidance with pre-configured common or standard questions as well as built in actions so it is simple to get started and easy to take the next step.

Architecture and Containerization

Your organization likely already has a preferred architecture in terms of choice of cloud provider, a hybrid model or on-premises. If your organization is considering a change, for example to move to work with multiple cloud providers, it's helpful to ensure that your embedded analytics and AI support both current and future deployment choices without requiring a major overhaul or replacement.

Containerization is increasingly popular for AI-powered embedded analytics. Containers allow your software developers to deploy applications across multiple environments without rewriting the code. Embedded containers are lightweight components that package code and dependencies to enable applications to run on their host systems with low memory and power consumption.

Benefits of containerization for embedded analytics and AI can include improved developer productivity, lower testing costs, increased product quality, reduced time to market and fewer quality assurance (QA) problems. When working with a vendor for AI-powered embedded analytics, it's helpful to ask them about their current support and roadmap for containerization.

Automation and Operational Efficiency

Automation impacts the time and personnel needed to develop, test and launch your AI-powered embedded analytics and how you will ease deployment, upgrades and day-to-day operations. For adding AI-powered embedded analytics into your existing application, you will want to leverage your current and planned tech stack; developer, security and operations (DevSecOps) processes; and continuous integration and development (CI/CD) pipelines.

Table of Contents

1 Get Started with Embedded Analytics
2 Set the Foundation for Integration
3 Define Product Requirements
4 Evaluate Implementation Options
5 Design the User Experience
6 Develop and Test
7 Deploy and Support
8 Learn About MicroStrategy Embedded Analytics

Compliance and Trust

Ensure that the AI-powered embedded analytics or GenAI solution complies with relevant data privacy regulations such as the General Data Protection Regulation (GDPR) and AI Act in Europe, the California Consumer Privacy Act (CCPA) in the United States, or similar laws in other jurisdictions. This includes obtaining consent for data collection and processing, protecting user personally identifiable information (PII), implementing appropriate data encryption and security measures, and providing users with control over their data.

Depending on your industry, you may also require compliance for financial regulations, e.g., Sarbanes-Oxley Act or the Payment Card Industry Data Security Standard, or for U.S. healthcare the Health Insurance Portability and Accountability Act (HIPAA). One of the key concerns with AI is trust. AI hallucinations may result in incorrect or incomplete answers. As well, AI can raise issues of systemic bias. These are important considerations for choosing a GenAI-powered analytics provider and educating your internal users or customers about what to expect.

Data Governance and Quality Management

Clarify roles and responsibilities for data governance within the organization, including data stewards, data custodians and data governance committees. Assign accountability for data governance activities such as data quality management, metadata management and compliance oversight. Consider how you will measure and assure data quality.

Consider metadata management processes to catalog and document the metadata associated with embedded analytics datasets, including data definitions, data lineage and data usage policies. When possible, maintain a centralized metadata repository to facilitate data discovery, understanding and governance.

Explainability and Transparency

Plan to describe the data used to train models and the reasoning behind AI-generated insights.

You may also want to explain to users the concept of a GenAI hallucination for why the AI model may on occasion produce an incorrect or incomplete answer.

Monitoring and Management Tools

You will want your embedded AI-powered analytics to provide integrations to your existing or planned DevSecOps monitoring tools to measure app adoption with a 360-degree management view and real-time insights on usage patterns, error rates and commonly used pathways that people explore and benefit from your product.

Table of Contents

1 Get Started with Embedded Analytics

2 Set the Foundation for Integration

3 Define Product Requirements

4 Evaluate Implementation Options

5 Design the User Experience

6 Develop and Test

7 Deploy and Support

8 Learn About MicroStrategy Embedded Analytics

Performance and Scalability

One of the most important considerations for performance of AI-powered embedded analytics is the loading speed in an environment when you scale to many concurrent users. This is one reason why you'll see a lot of attention placed on in-memory capabilities, direct query optimization and push-down data blending.

As people, our expectations from consumer applications and consumer devices are that content and services are available in less than 1 second, ideally less than 0.2 seconds (200 milliseconds) for a seamless experience. This has popularized caching commonly requested content, blending data into a single integrated data store, or storing data in-memory. These are all ways to reduce or avoid the delay of an API request to pull data from an external database while the user waits impatiently looking at their screen.

Security and Accessibility

As part of a robust DevSecOps framework, you should endeavor to deliver security at scale with governance layers at every stage of development and deployment. This starts with multi-factor authentication and single-sign on integration (SSO) and extends to role-based security filters to control access and protect data integration. Avoid shortcuts such as hardcoding usernames and passwords. Standard authentication options include Active Directory, OAuth 2.0 or Lightweight Direct Access Protocol (LDAP).

Table of Contents

1 Get Started with Embedded Analytics

2 Set the Foundation for Integration

3 Define Product Requirements

4 Evaluate Implementation Options

5 Design the User Experience

6 Develop and Test

7 Deploy and Support

8 Learn About MicroStrategy Embedded Analytics

4. Evaluate Implementation Options

Buy or Build: Weighing Your Options

Buy or Build

Evaluate your options to work with a best-of-breed partner or build internally.

API

REST API's meet most needs while GraphQL serves more specialized environments.

Library

Your prototype web application may start with coding from an open-source visual component library to publish in an iframe before upgrading to an analytics visualization platform.

SDK

Peer reviews help confirm that your developers will be happy with the vendor's software development kit and community forums.

Cloud

Embedding from an AI-powered analytics cloud service can offer benefits to get started quickly with advanced features while tying costs to usage.

Several considerations help inform your buy or build decision. Your organization's product culture likely trends toward either buying best-of-breed third-party services or developing with internal resources. A build process gives your organization the most control over software development and intellectual property ownership. It can mitigate against the costs of purchasing per user software licenses if you anticipate scaling to millions of users.

For a small number of basic visualizations that will appear in a web application, you may consider hardcoding from a visual component library such as D3.js or Material UI that support your chosen software programming language publishing as an iframe. Short for inline frames, iframes are commonly used to embed content into a webpage.

An approach to hard code to iframes can get you started for simple charts or a product prototype but become impractical and hurt performance for agile, commercial applications compared to a best-of-breed analytics and visualization platform that supports agile programming, container components and greater interactivity. These concerns raise the risk of project failure from a build approach.

Table of Contents

1 Get Started with Embedded Analytics

2 Set the Foundation for Integration

3 Define Product Requirements

4 Evaluate Implementation Options

5 Design the User Experience

6 Develop and Test

7 Deploy and Support

8 Learn About MicroStrategy Embedded Analytics

Where a buy approach works well is when you want to go to market quickly and benefit from the expertise of a best of breed provider. The new generation of AI-powered embedded analytics appear seamlessly in your apps and deliver powerful, intuitive analytics. Your choice of an AI-powered embedded analytics partner should support a diverse range of databases and cloud services, to give you options for transforming and modeling data.

REST APIs provide a flexible and scalable way to build web services that can be easily consumed by various clients, including web browsers, mobile devices, and other applications. REST APIs are stateless, meaning that the server and client don't know each other's state. Each request contains all the information the server needs to understand it. This allows the server and client to understand any message sent, even if they haven't seen previous messages.

For complex data requirements and frequent schema changes, or a unified interface to access data from multiple services in a microservices architecture, you may want to consider GraphQL as an alternative to REST. The client-driven approach by GraphQL, compared to the REST API server-driven architecture, can also be a good choice to help optimize mobile applications for speed and bandwidth by combining multiple requests into a single network query.

To secure your APIs, you will need API keys or OAuth tokens. API keys can be a good choice for read-only data. OAuth tokens provide more options for authorization but can be more complex to implement.

When you decide on embedding best-of-breed external analytics and GenAI application, you should expect that they give your developers a robust, well documented software development kit.

A good SDK has comprehensive documentation, is easy to use, and supports platform compatibility with Python and TypeScript and other popular choices by developers. Your developers will appreciate that the vendor supports developer community forums to ask for help and collaboration. Gartner Peer Insights and G2 are useful for peer reviews.

Advantages of Cloud Services for Embedded Analytics

Cloud integrations support cloud-native services and shift more of the burden of software development and maintenance to your partner. Choosing a cloud service for embedded analytics can be advantageous when you need to integrate advanced AI-powered analytics capabilities quickly and cost-effectively into your application, reduce development and maintenance overhead, and prefer to budget on a subscription model that scales with usage versus the higher up-front investments required for a build approach.

Table of Contents

1 Get Started with Embedded Analytics
2 Set the Foundation for Integration
3 Define Product Requirements
4 Evaluate Implementation Options
5 Design the User Experience
6 Develop and Test
7 Deploy and Support
8 Learn About MicroStrategy Embedded Analytics

5. Design the User Experience

Chat Bots

Consider incorporating GenAI-powered natural language search bars for data exploration and automated report generation based on user queries.

Data Visualizations

Integrate charts and graphs seamlessly within your application's user interface.

Navigation

Ensure intuitive navigation and clear data representation.

Roles and Permissions

Plan how to control data access and show role-specific workflows.

Consider incorporating GenAI-powered natural language search bars for data exploration and automated report generation based on user queries. It can be helpful to include some sample questions.

Part of the GenAI experience is to give people the option to ask follow-up questions. Advanced AI-powered embedded analytics allow a user to not only ask natural language questions and click to see relevant data but also to click to initiate built-in actions tied to role-specific workflows.

For data visualizations, your goal is to integrate charts, graphs, or other analytics elements seamlessly within your application's user interface (UI). To ensure intuitive navigation and clear data representation, the host application, embedded analytics, and GenAI should present a coherent, seamless experience. It's important to clearly label data sources and document data lineage.

Commonly embedded analytics objects start with charts and tabular data to display as a data table, and key performance indicators (KPIs) to track what's happening to an aggregated measure such as sales or the number of subscribers. These KPIs can be expressed as dynamic text, such as "Profits grew by 6% last quarter".

Steve Krug offers worthwhile tips for usability in his book Don't Make Me Think. The goal is that the embedded analytics tell a data-driven story that feels natural in your application, so that the user does not need to stop their mental flow to try to figure out what to click or what question to ask.

Plan user roles and permissions to control access. It's preferable for users to inherit software permissions from the host application.

In their O'Reilly book "Embedded Analytics," Donald Farmer and Jim Horbury note that "From the point of view of an IT team ... embedding your analytics in another application that is already secure has a simple but significant benefit. There is only one environment and one login to be managed; the analytics application does not add an additional layer of complexity."

Table of Contents

1. Get Started with Embedded Analytics
2. Set the Foundation for Integration
3. Define Product Requirements
4. Evaluate Implementation Options
5. Design the User Experience
6. Develop and Test
7. Deploy and Support
8. Learn About MicroStrategy Embedded Analytics

6. Develop and Test

To integrate Generative AI into your application, involve developers familiar with APIs, your chosen framework, and Generative AI concepts. Follow best practices and documentation while developing the integration. Thoroughly test the functionality and user experience from a customer perspective, including Generative AI-powered features.

Development Plan Best Practices

You will want to translate your product requirements documentation (PRD) into a project plan that includes:

- Source the data elements. Document data sources and any assumptions made about the data.
- Create analytics and visualizations that solve real-world needs.
- Implement a GenAI model that correctly answers relevant natural language questions while minimizing inappropriate content.
- Integrate with your organization's DevSecOps tools and choice of Docker, Kubernetes or other container architecture.
- Support an ever-larger number of concurrent users through horizontal scaling of adding more nodes to the system along with vertical scaling.

Have a Robust Testing Plan

Allow proper timing to thoroughly test the functionality and user experience from a customer perspective, including Generative AI-powered features.

People need to be able to go back and forth between the host service and embedded analytics without having to click a browser back button and reload the host content.

It's important to test your visualizations on desktop, mobile and tablet devices.

Your embedded analytics should provide easy, intuitive interactions on screens of different sizes, with a sufficiently fast response time to not disrupt the user workflow.

Table of Contents

1 Get Started with Embedded Analytics
2 Set the Foundation for Integration
3 Define Product Requirements
4 Evaluate Implementation Options
5 Design the User Experience
6 Develop and Test
7 Deploy and Support
8 Learn About MicroStrategy Embedded Analytics

7. Deploy and Support

The launch of your AI-powered embedded analytics should be a smooth, exciting event for your organization. It's helpful to plan for internal training, marketing and support readiness. You may want to consider a phased rollout starting with a beta release to a limited audience or for internal use only.

Product Launch Process Steps

- Develop a comprehensive launch strategy that includes internal training, marketing and customer support readiness.
- Clearly communicate the benefits and changes to both stakeholders and customers.
- Establish a responsive support system to address customer issues and feedback.
- Define and track key performance indicators to measure the success of the embedded analytics integration.

Clearly communicate the benefits and changes to both stakeholders and customers. This includes not only the features and functionality but also the data-driven outcomes and benefits that people can expect. It's also helpful to create a feedback loop to gather user input and continuously improve the product.

Establish a responsive support system to address customer issues and feedback. This may involve creating new support materials, such as FAQs or tutorials, and training customer support staff on the new analytics features including the new GenAI functionality.

Choose a launch strategy that aligns with your infrastructure, security needs and AI integration considerations.

Define and track key performance indicators (KPIs) to measure the success of the embedded analytics integration. These could include user adoption rates, engagement metrics, customer satisfaction scores, and the impact on revenue or other business goals.

MicroStrategy's Role and Trustworthiness

"We're experiencing the start of an exciting third generation of business intelligence powered by generative AI," notes Brett Sheppard in the MicroStrategy blog Join the Third Generation of Business Intelligence.

MicroStrategy's modern, open architecture, and comprehensive set of APIs enable organizations to embed analytics into any application, platform, or product.

This allows organizations to deliver modern consumer-grade, AI-powered experiences with powerful, intuitive analytics that are seamlessly integrated into the apps and products that employees, customers and partners use daily.

Table of Contents

1. Get Started with Embedded Analytics
2. Set the Foundation for Integration
3. Define Product Requirements
4. Evaluate Implementation Options
5. Design the User Experience
6. Develop and Test
7. Deploy and Support
8. Learn About MicroStrategy Embedded Analytics

8. Learn About MicroStrategy Embedded Analytics

Hundreds of global companies and public sector agencies trust MicroStrategy as an AI-powered embedded analytics partner to deliver customized analytics experiences on top of a secure layer of trusted metadata with options to add natural language discovery, predictive analytics and geospatial analysis.

While other BI and AI vendors have been acquired, merged, sold, or divested, MicroStrategy is the world's largest independent provider of AI-powered pervasive analytics at scale.

Unlike the previous generation of embedded analytics, MicroStrategy makes it possible for you to integrate controls and actions so they feel like part of your apps and systems. MicroStrategy offers several choices for pricing models including white labeling and revenue sharing.

» **Learn more about how to make every app intelligent with MicroStrategy AI-powered embedded analytics at** microstrategy.com/embedded-analytics.

Table of Contents

1 Get Started with Embedded Analytics
2 Set the Foundation for Integration
3 Define Product Requirements
4 Evaluate Implementation Options
5 Design the User Experience
6 Develop and Test
7 Deploy and Support
8 Learn About MicroStrategy Embedded Analytics