Building Enterprise AI Applications with Single-shot Data Retrieval

A handbook on how to create AI customer agent for enterprises

Madhukar Kumar

A little over two years ago, ChatGPT crashed into the tech world like a rogue wave hitting a tranquil beach—unexpected, a little overwhelming, and reshaping everything in its path. Soon after, much like how someone decided to put a camera in front of a radio and aired the first TV show when television was invented, people started playing with this new tech toy, asking it to respond like Shakespeare, generate poems and bedtime stories, and even argue the finer points of politics and possible ingredients of sentience.

But soon, the novelty wore off, and something more profound took shape. Turns out, AI wasn't just a parlor trick; it was embedding itself into everything we do, shifting from gimmick to game-changer affecting trillions of dollars in economy across the globe.

But, is this a bubble in tech and B2C only or are we seeing a similar wave play out in the world of large enterprises?

In my conversations with executives and industry leaders, I keep coming back to a simple but pressing question: What's really happening with AI in enterprises? Beyond the splashy headlines and tall promises, where is it actually making a difference? And, just as important, where is it falling short?

One insight has surfaced repeatedly: access to cutting edge LLMs is no longer the differentiator. Open-source models and cloud-based APIs have made generative AI available to anyone, from Wall Street executives to a farmer in a remote village with a mobile phone with no access to the internet. If you don't believe me, just dial 1-800-CHATGPT (use your flip phone if you must).

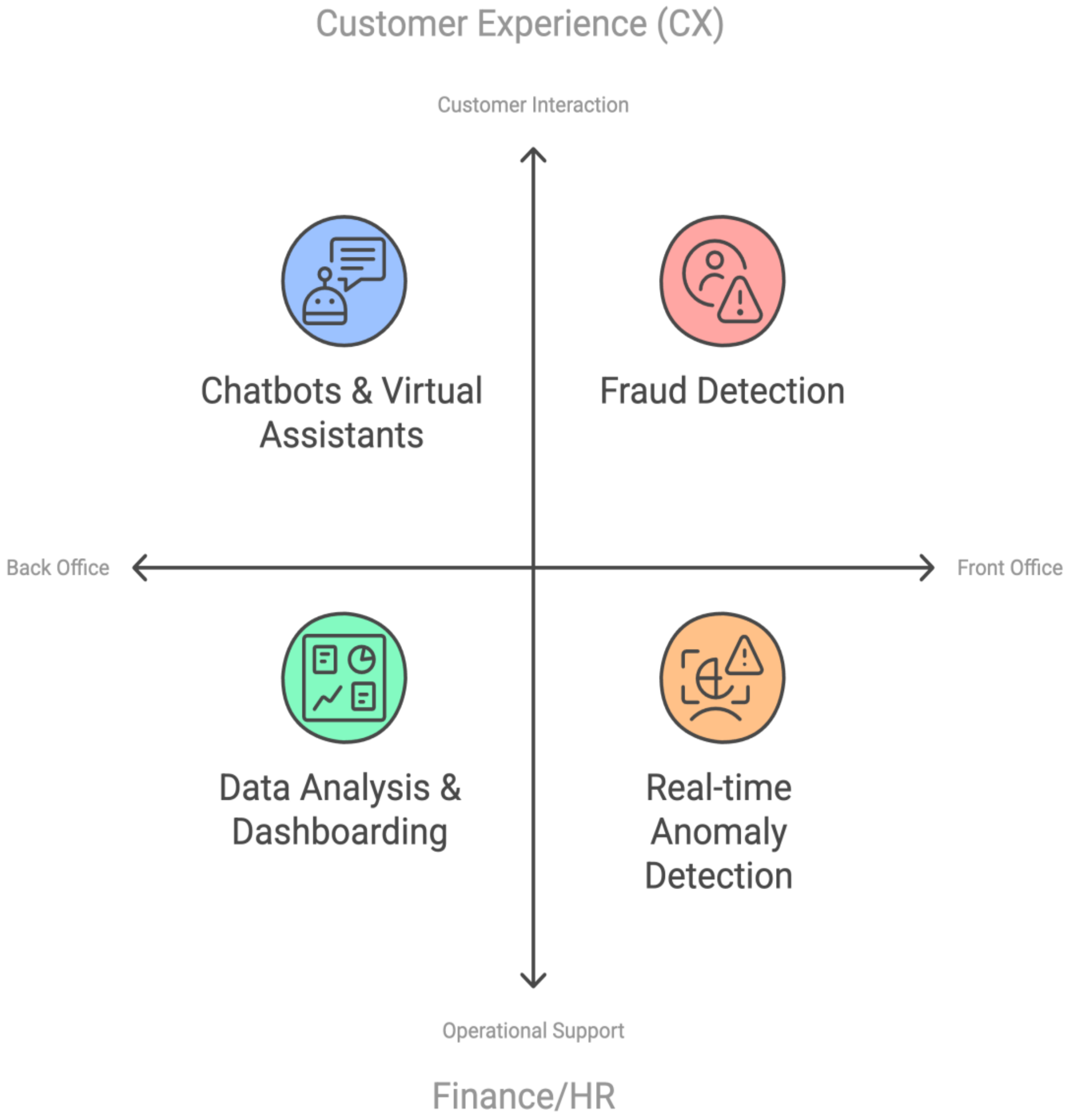
So, if access to "intelligence" isn't the moat, what is?

The answer, as it turns out, is as old as business itself: knowledge. Or, more specifically, the data that fuels AI.

History offers a clear parallel. During the internet boom, software ate the world, and companies that leaned into digital transformation—think Netflix versus Blockbuster—emerged as dominant forces. We can see this play out in the rise of the next wave of trillion-dollar enterprises will be those that master data—structuring, managing, and wielding it with precision to supercharge AI models, and of course those who provide the picks and shovels aka - infrastructure.

Before we get into the hard part—untangling enterprise data infrastructure—let's take a look at where AI is already making an impact. Not surprisingly, my observation is that enterprise AI use cases start in the back office before they evolve into front-line, customer-facing innovations.

AI Use Cases in Enterprises



Back office use cases:

1. Customer Service & Support

By far the most common use case I keep hearing about is the use of AI chatbots and virtual assistants being used for customer interactions, helping support agents resolve customer issues efficiently. If we are talking about agentic applications then this often includes intelligent ticket triage, automated routing and performing repetitive tasks, ensuring faster response times at lower costs. Additionally, sentiment analysis on incoming tickets, emails, or chat messages allows enterprises to gauge customer emotions and adjust responses accordingly. (PS: Stay tuned till the end if you want to see how to build this example use case).

2. Sales & Marketing Intelligence

AI-driven content generation and outreach have now made their way into a majority of all enterprises. The more sophisticated versions of this include personalized product recommendations and targeted marketing campaigns that some companies are also using for customer churn prediction, allowing proactive retention strategies to be implemented before customers leave. Candidly, this use case overlaps between back office and front office but I see a lot more AI being used in the background versus what the end customer actually ends up seeing for example, sentiment analysis from User Generated Content (UGC).

3. Knowledge Management & Document Processing

Almost all enterprises I have spoken to are leveraging AI for searching and synthesizing information over vast document repositories, making contracts, manuals, and documentation more accessible. This is where the use of semantic search and vectors stand out and the response from the search is then used to generate documents summarization, classification, and sometimes tagging. I have seen the use of privately deployed commercial models to open source models based applications to off the shelf AI search tools being used for this use case.

4. Forecasting & Business Planning

Another use case that is quite common in my experience is using AI tools to do data analysis and building dashboards and charts for decision making though this is not a solved problem yet and involves wrangling with structured data (a whole different animal).

5. HR & Talent Management

Though not as common as other use cases I have spoken with companies who are also using AI-driven tools for candidate screening and resume matching to streamline recruitment processes. Some companies are also exploring personalized learning and upskilling programs with the help of AI.

Front office use cases:

When it comes to using AI that may be directly front-office facing, the most common use cases are around security and risk mitigation.

6. Fraud Detection & Risk Analysis

Real-time anomaly detection in financial transactions and insurance claims helps prevent fraud before it escalates. This was already a standard practice for financial and security companies but now predictive models are finding their way for credit risk assessment, underwriting, and compliance. AI-powered identity verification and document validation are other areas finding ground in this category of use case.

7. Real-Time Anomaly & Outlier Detection

Most security companies or companies with high sensitivity for anomalies (for example transportation or credit card companies) monitor streaming data and are now looking at using LLMs to match against their vast corpus of data to make quick and often agentic decisions.

8. Personalization and Recommendation Systems

Although this use case has been around in many sectors like retail and ad tech, with LLMs now this has started to find its way into a number of marketing use case. In this category we see companies use AI to analyze customer data (browsing behavior, purchase history, preferences) to tailor the user experience within the session vs running batch analytics later and then sending recommendations asynchronously.

9. Voice Assistants & Search

A number of companies have also started to augment or replace the search functionalities on the publicly available web properties to surface data that may have been erstwhile inaccessible. In addition, we have also seen some companies now use AI voice assistants to front end some of these information synthesis use cases in an attempt to help their customers with self-help before opening support tickets.

Barriers to AI

Trying to build AI in large enterprises is like attempting to retrofit a car or a plane while it's still moving —everything is in motion, interconnected, and built on decades-old infrastructure that wasn't designed for the new demands being placed upon it. More so, when implementing AI often the fragmented data infrastructures get exposed. Operational data resides in transactional databases, while analytical insights require data warehouses, and AI workloads need vector stores. Managing these separate systems comes with its own sets of complex problems:

If you have ever worked in a large enterprise you probably already know that the biggest barriers to adopt any new technology or build something are primarily:

- Dysfunctional processes brought about by the sheer number of departments and employees
- Data silos
- Technology fragmentation

Given the focus on data as a differentiator for AI apps we let's focus on this as the biggest challenge and opportunity.

Duplication & Latency: Moving and syncing data between OLTP, OLAP, and vector indexes adds overhead and delays.

Real-time Context Issues: AI models need fresh data, but fragmented systems make real-time inference almost impossible if not difficult.

Scalability Challenges: AI models require synthesis over massive datasets, straining traditional database architectures.

Security & Governance Risks: Broad access requirements for AI can conflict with enterprise security policies.

Given the fact that no LLMs have been trained on bespoke private enterprise data, it is moot that the vast majority of data needs to be searched and curated before handing over the context data to LLMs for either information synthesis or to take actions. This strategy is often referred to as Retrieval Augmented Generation (RAG) and primarily there are two options to manage the data for this process which we look at next.

Tackling Data Silos: The Two Options

I. Ensemble Databases

Ensemble databases involve integrating multiple database systems, such as transactional databases, analytical data warehouses, and vector stores, to support AI workloads. This approach allows enterprises to retain their existing infrastructure while introducing AI capabilities.

Pros:

Minimal disruption to existing infrastructure.

Lower upfront cost by adding a vector-only database to the existing data landscape.

Cons:

Not scalable for external applications requiring real-time insights.

Performance bottlenecks due to multiple data retrieval steps.

Long-term costs increase due to inefficiencies in data movement.

One could argue that this approach, though simpler on the surface, adds significant technical debt and leads to higher costs for a number of reasons:

Data movement: Once you add a vector-only data, you now need to create and maintain pipelines to create embeddings (vectors) and store them along with meta data. When retrieving the data, you now have to also worry about getting the actual data chunks from another database. Worse, some vector-databases do not have data immediately available once you add them to the index. This leads to a bigger issue - accuracy.

Accuracy: If the data searched and curated is not current and/or fresh, the responses from the LLMs will be based on point-in-time data which could be disastrous in use cases like anomaly detection, real-time financial decisions or anything to do with streaming audio and video data.

Latency: Given that different kinds of data, structured, semi-structured and unstructured, sit in different databases, it becomes impossible to now retrieve data without multiple round-trips which introduces significant latency in the overall applications. This leads to not only low to no adoption but also accuracy issues we see above.

In order to enable AI to be able to get accurate and latest data in the least amount of time aka single-shot retrieval requires first setting up a data layer that becomes the interface to all AI for all enterprise data. This brings us to option 2 below.

II: AI-Optimized Databases

Instead of adding a vector only data base first, let's examine what AI applications need for a single-shot retrieval:

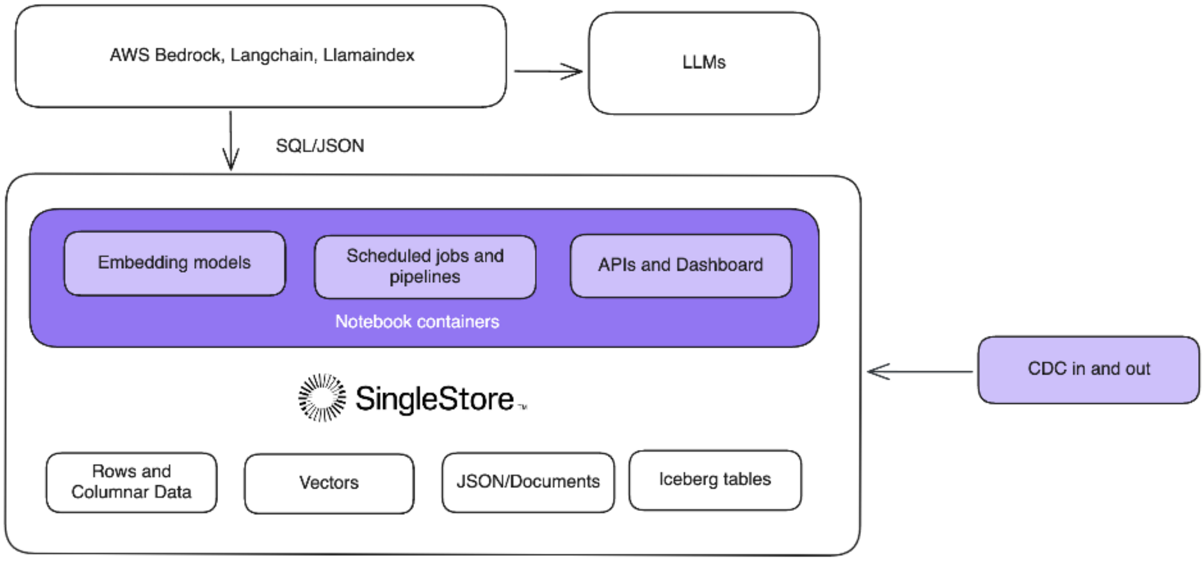
- Ability to run queries with accurate responses across petabytes of data in a few milliseconds (in order to match the evolving millseconds real-time LLMs inference).
- Support for all major data types and interfaces, for example SQL, JSON, Geo-spatial, Key-Value, Vectors
- Ability to search with vectors and exact keyword matches.
- Ability to retrieve and reason with real-time streaming data from events like Kafka or Kinesis etc.
- Ability to write custom jobs local to data, for example, frequent embedding pipelines etc.
- Finally, an ability to do CDC in and out to not just other databases but also applications like a CRM and ERP systems.

Does a database like this exist?

Yes, I can think of two. First, there is SAP Hana. However, as most enterprises know it is a memory-only database so expensive to run retrieval over petabytes of data.

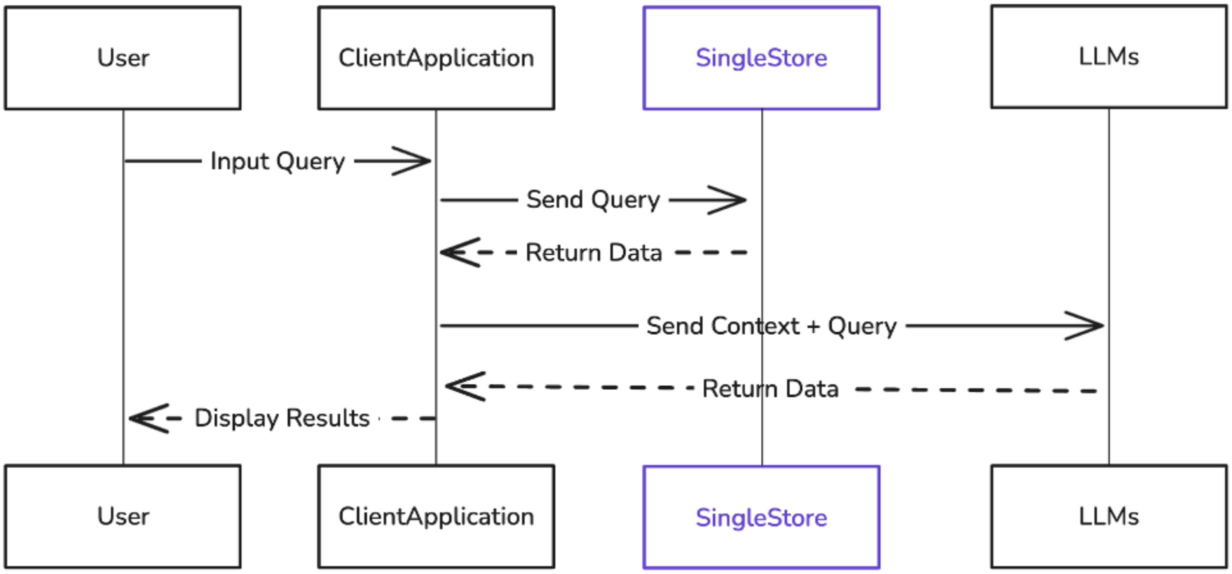
The second option is SingleStore. It is a Hybrid Transactional and Analytics Processing (HTAP) database and is memory-first instead of memory-only. In addition, it has support for all datatypes and interfaces except for Graph but that is a separate article altogether. Here are some key capabilities that make SingleStore the ideal choice for enabling single-shot retrieval for enterprise AI use cases:

- Multi-model support: Relational, vector, and time-series workloads in one system.
- Metadata & knowledge graphs: AI-ready indexing and cataloging.
- Extensibility with custom functions: Ability to create Web Assembly (WASM) functions as UDF for ML and predictive AI use cases.
- Real-time processing: Ability to process Kafka streams and process within milliseconds
- CDC in and out: Built-in pipelines for bringing in data from all major data sources, including Apache Iceberg and enterprise applications like SAP and Salesforce.
- Platform for API-driven AI applications: Built-in container services with Python Jupyter notebook interface to enable scheduled jobs, exposing API end points and creating persistent materialized view dashboards
- Seamless ecosystem integration: Supports AI frameworks with connectors from OpenAI, Langchain, Llamaindex, and NextJS-based ORMs like Drizzle. Additionally, enterprises can integrate SingleStore with AWS Bedrock Agents to create AI-powered applications.



High-Level Setup Steps for AI-Powered Data Layer

What will we be building?
Our goal is to be able to get the user query, get the appropriate context from our data layer using vector, exact keyword match, do matches, joins and aggregates across different data types then return this as a content to the LLM which then responds back with the context to the client app.
Our client app could also have input/output validation and Guardrails to add enterprise level features in the application.



1. **Set Up SingleStore**: Deploy a SingleStore cluster on AWS, Azure, or on-premise. Create a schema to fit your needs for transactional, analytical, and vector workloads.

2. **Create Data Pipelines**

   1. Kafka Integration:
   2. Use built-in SingleStore pipelines to ingest streaming data from Kafka.
   3. **CRM & ERP Integration**: Set up CDC pipelines to bring data from Salesforce, SAP, or other enterprise applications. SingleStore recently acquired BryteFlow that makes this into a no code task.
   4. **S3 Bucket Ingestion**: Load structured and unstructured data from cloud storage, for example csv files or text only files that represent knowledge base articles

3. **Vectorize Data with Python Jobs**

   Use SingleStore's built-in Jupyter based containerized jobs to schedule Python jobs. These jobs could be put on a schedule and can be used to create embeddings while bringing them in

4. **Enable Real-Time Data Layer for AI Agents**:

   - Expose SQL and vector search endpoints for AI models through Python Notebooks that act like Lambda functions.
   - Provide function calling interfaces to allow AI agents to retrieve structured, semi-structured, and real-time data efficiently.

By following these steps, enterprises can build a scalable, AI-optimized data platform that ensures accurate and low-latency retrieval for AI applications.

Conclusion

The enterprise AI landscape is rapidly evolving, and the true differentiator isn't access to AI models—it's the ability to effectively manage and utilize enterprise data. While many organizations are still grappling with fragmented data infrastructures and the challenges of implementing AI at scale, the path forward is clear: successful AI implementation requires a robust, unified data foundation that can handle diverse data types and provide single-shot retrieval capabilities.

In our next section, we'll look at a detailed, step-by-step guide on building an AI-powered customer support agent using AWS Bedrock Agents and SingleStore. We'll cover everything from initial setup to deployment, showing you how to create a practical, production-ready AI application that leverages enterprise data effectively.