**The AI Developer’s Dilemma: Proprietary AI vs. Open Source Ecosystem**

***Fundamental choices impacting integration and deployment at scale of GenAI into businesses***

[Gadi Singer](https://gadi-singer.medium.com/?source=post_page-----453ac735b760--------------------------------)

Sep 30, 2024

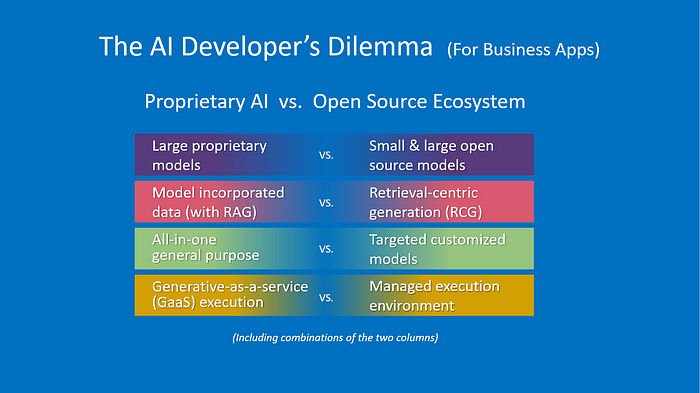
Before a company or a developer adopts generative artificial intelligence (GenAI), they often wonder how to get business value from the integration of AI into their business. With this in mind, a fundamental question arises: Which approach will deliver the best value on investment — a large all-encompassing proprietary model or an open source AI model that can be molded and fine-tuned for a company’s needs? AI adoption strategies fall within a wide spectrum, from accessing a cloud service from a large proprietary frontier model like [OpenAI’s GPT-4o](https://openai.com/index/hello-gpt-4o/) to building an internal solution in the company’s compute environment with an open source small model using indexed company data for a targeted set of tasks. Current AI solutions go well beyond the model itself, with a whole ecosystem of retrieval systems, agents, and other functional components such as AI accelerators, which are beneficial for both large and small models. Emergence of cross-industry collaborations like the [Open Platform for Enterprise AI (OPEA)](https://opea.dev/) further the promise of streamlining the access and structuring of end-to-end open source solutions.

This basic choice between the open source ecosystem and a proprietary setting impacts countless business and technical decisions, making it “the AI developer’s dilemma.” I believe that for most enterprise and other business deployments, it makes sense to initially use proprietary models to learn about AI’s potential and minimize early capital expenditure (CapEx). However, for broad sustained deployment, in many cases companies would use ecosystem-based open source targeted solutions, which allows for a cost-effective, adaptable strategy that aligns with evolving business needs and industry trends.

**GenAI Transition from Consumer to Business Deployment**

When GenAI burst onto the scene in late 2022 with Open AI’s GPT-3 and ChatGPT 3.5, it mainly garnered consumer interest. As businesses began investigating GenAI, two approaches to deploying GenAI quickly emerged in 2023 — using giant frontier models like ChatGPT vs. the newly introduced small, open source models originally inspired by Meta’s LLaMa model. By early 2024, two basic approaches have solidified, as shown in the columns in Figure 1. With the proprietary AI approach, the company relies on a large closed model to provide all the needed technology value. For example, taking GPT-4o as a proxy for the left column, AI developers would use OpenAI technology for the model, data, security, and compute. With the open source ecosystem AI approach, the company or developer may opt for the right-sized open source model, using corporate or private data, customized functionality, and the necessary compute and security.

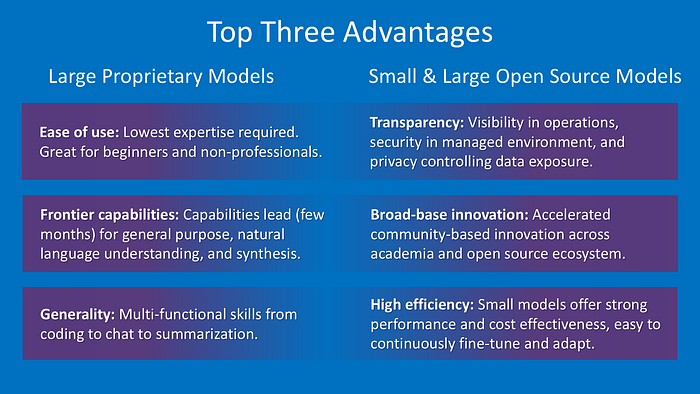
Both directions are valid and have advantages and disadvantages. It is not an absolute partition and developers can choose components from either approach, but taking either a proprietary or ecosystem-based open source AI path provides the company with a strategy with high internal consistency. While it is expected that both approaches will be broadly deployed, I believe that after an initial learning and transition period, most companies will follow the open source approach. Depending on the usage and setting, open source internal AI may provide significant benefits, including the ability to fine-tune the model and drive deployment using the company’s current infrastructure to run the model at the edge, on the client, in the data center, or as a dedicated service. With new AI fine-tuning tools, deep expertise is less of a barrier.



*Figure 1. Base approaches to the AI developer’s dilemma. Image credit: Intel Labs.*

Across all industries, AI developers are using GenAI for a variety of applications. An October 2023 [poll by Gartner](https://www.gartner.com/en/newsroom/press-releases/2023-10-03-gartner-poll-finds-55-percent-of-organizations-are-in-piloting-or-production-mode-with-generative-ai) found that 55% of organizations reported increasing investment in GenAI since early 2023, and many companies are in pilot or production mode for the growing technology. As of the time of the survey, companies were mainly investing in using GenAI for software development, followed closely by marketing and customer service functions. Clearly, the range of AI applications is growing rapidly.

**Large Proprietary Models vs. Small and Large Open Source Models**



*Figure 2: Advantages of large proprietary models, and small and large open source models. For business considerations, see Figure 7 for CapEx and OpEx aspects. Image credit: Intel Labs.*

In my blog [Survival of the Fittest: Compact Generative AI Models Are the Future for Cost-Effective AI at Scale](https://towardsdatascience.com/survival-of-the-fittest-compact-generative-ai-models-are-the-future-for-cost-effective-ai-at-scale-6bbdc138f618), I provide a detailed evaluation of large models vs. small models. In essence, following the introduction of [Meta’s LLaMa open source model in February 2023](https://ai.meta.com/blog/large-language-model-llama-meta-ai/), there has been a virtuous cycle of innovation and rapid improvement where the academia and broad-base ecosystem are creating highly effective models that are 10x to 100x smaller than the large frontier models. A crop of small models, which in 2024 were mostly less than 30 billion parameters, could [closely match](https://azure.microsoft.com/en-us/blog/introducing-phi-3-redefining-whats-possible-with-slms/) the capabilities of ChatGPT-style large models containing well over 100B parameters, especially when targeted for particular domains. While GenAI is already being deployed throughout industries for a wide range of business usages, the use of compact models is rising.

In addition, open source models are mostly lagging [only six to 12 months behind](https://ark-invest.com/newsletter_item/1-openais-improved-chatgpt-should-delight-both-expert-and-novice-developers) the performance of proprietary models. Using the broad language benchmark MMLU, the improvement pace of the open source models is faster and the gap seems to be closing with proprietary models. For example, OpenAI’s [GPT-4o](https://openai.com/index/hello-gpt-4o/) came out this year on May 13 with major multimodal features while Microsoft’s small open source [Phi-3-vision](https://azure.microsoft.com/en-us/blog/new-models-added-to-the-phi-3-family-available-on-microsoft-azure/) was introduced just a week later on May 21. In [rudimentary comparisons](https://youtu.be/PZaNL6igONU?si=jCvhwvWBoZFnRG5X) done on visual recognition and understanding, the models showed some similar competencies, with several tests even favoring the Phi-3-vision model. [Initial evaluations of Meta’s Llama 3.2 open source release](https://ai.meta.com/blog/llama-3-2-connect-2024-vision-edge-mobile-devices/) suggest that its “vision models are competitive with leading foundation models, Claude 3 Haiku and GPT4o-mini on image recognition and a range of visual understanding tasks.”

Large models have incredible all-in-one versatility. Developers can choose from a variety of large commercially available proprietary GenAI models, including OpenAI’s GPT-4o multimodal model. Google’s [Gemini 1.5](https://deepmind.google/technologies/gemini/#introduction) natively multimodal model is available in four sizes: Nano for mobile device app development, Flash small model for specific tasks, Pro for a wide range of tasks, and Ultra for highly complex tasks. And Anthropic’s [Claude 3 Opus](https://www.anthropic.com/news/claude-3-family), rumored to have [approximately 2 trillion parameters](https://lifearchitect.substack.com/p/the-memo-special-edition-claude-3), has a 200K token context window, allowing users to upload large amounts of information. There’s also another category of out-of-the-box large GenAI models that businesses can use for employee productivity and creative development. [Microsoft 365 Copilot](https://blogs.microsoft.com/blog/2023/03/16/introducing-microsoft-365-copilot-your-copilot-for-work/) integrates the Microsoft 365 Apps suite, Microsoft Graph (content and context from emails, files, meetings, chats, calendars, and contacts), and GPT-4.

Most large and small open source models are often more transparent about application frameworks, tool ecosystem, training data, and evaluation platforms. Model architecture, hyperparameters, response quality, input modalities, context window size, and inference cost are partially or fully disclosed. These models often provide information on the dataset so that developers can determine if it meets copyright or quality expectations. This transparency allows developers to easily interchange models for future versions. Among the growing number of small commercially available open source models, Meta’s [Llama 3 and 3.1](https://ai.meta.com/blog/meta-llama-3-1/) are based on transformer architecture and available in 8B, 70B, and 405B parameters. Llama 3.2 multimodal model has 11B and 90B, with smaller versions at 1B and 3B parameters. Built in collaboration with NVIDIA, Mistral AI’s [Mistral NeMo](https://mistral.ai/news/mistral-nemo/) is a 12B model that features a large 128k context window while Microsoft’s [Phi-3](https://news.microsoft.com/source/features/ai/the-phi-3-small-language-models-with-big-potential/) (3.8B, 7B, and 14B) offers Transformer models for reasoning and language understanding tasks. Microsoft highlights Phi models as an example of “[the surprising power of small language models](https://www.microsoft.com/en-us/research/blog/phi-2-the-surprising-power-of-small-language-models/)” while investing heavily in OpenAI’s very large models. Microsoft’s diverse interest in GenAI indicates that it’s not a one-size-fits-all market.

**Model-Incorporated Data (with RAG) vs. Retrieval-Centric Generation (RCG)**

The next key question that AI developers need to address is where to find the data used during inference — within the model parametric memory or outside the model (accessible by retrieval). It might be hard to believe, but the first ChatGPT launched in November 2022 did not have any access to data outside the model. It was trained on September 21, 2022 and notoriously had no inclination of events and data past its training date. This major oversight was addressed in 2023 when retrieval plug-ins where added. Today, most models are coupled with a retrieval front-end with exceptions in cases where there is no expectation of accessing large or continuously updating information, such as dedicated programming models.

Current models have made significant progress on this issue by enhancing the solution platforms with a retrieval-augmented generation (RAG) front-end to allow for extracting information external to the model. An efficient and secure RAG is a requirement in enterprise GenAI deployment, as shown by Microsoft’s introduction of [GPT-RAG](https://github.com/Azure/GPT-RAG/) in late 2023. Furthermore, in the blog [Knowledge Retrieval Takes Center Stage](https://towardsdatascience.com/knowledge-retrieval-takes-center-stage-183be733c6e8), I cover how in the transition from consumer to business deployment for GenAI, solutions should be built primarily around information external to the model using retrieval-centric generation (RCG).

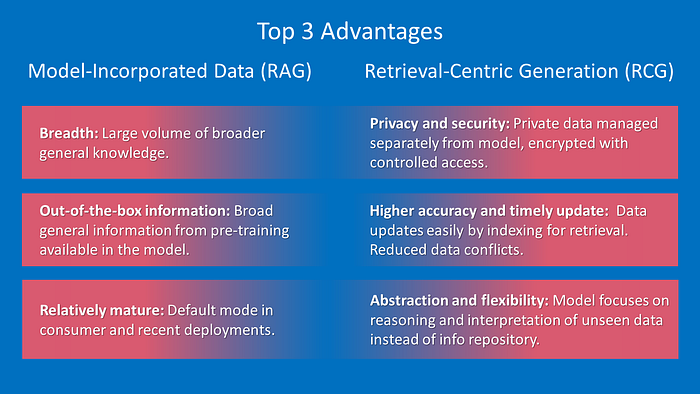
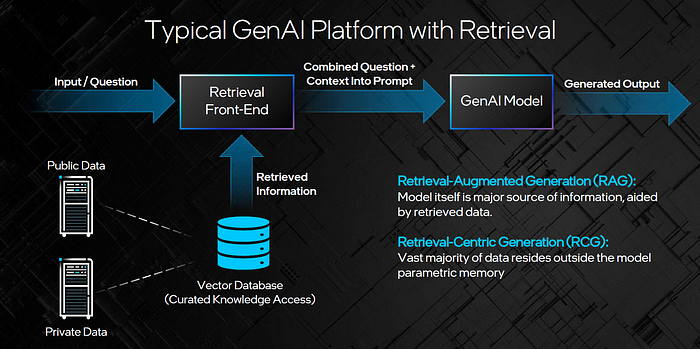


Figure 3. Advantage of RAG vs. RCG. Image credit: Intel Labs.

RCG models can be defined as a special case of RAG GenAI solutions designed for systems where the vast majority of data resides outside the model parametric memory and is mostly not seen in pre-training or fine-tuning. With RCG, the primary role of the GenAI model is to interpret rich retrieved information from a company’s indexed data corpus or other curated content. Rather than memorizing data, the model focuses on fine-tuning for targeted constructs, relationships, and functionality. The quality of data in generated output is expected to approach 100% accuracy and timeliness.



*Figure 4. How retrieval works in GenAI platforms. Image credit: Intel Labs.*

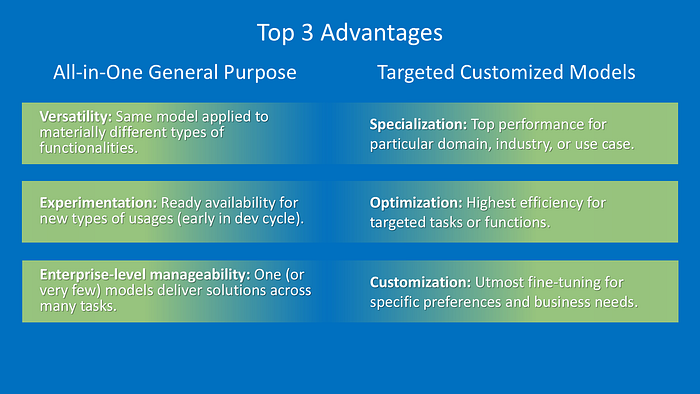
[OPEA](https://www.intel.com/content/www/us/en/developer/articles/news/introducing-the-open-platform-for-enterprise-ai.html) is a cross-ecosystem effort to ease the adoption and tuning of GenAI systems. Using this composable framework, developers can create and evaluate “open, multi-provider, robust, and composable GenAI solutions that harness the best innovation across the ecosystem.” OPEA is expected to simplify the implementation of enterprise-grade composite GenAI solutions, including RAG, agents, and memory systems.



*Figure 5. OPEA core principles for GenAI implementation.* *Image credit: OPEA.*

**All-in-One General Purpose vs. Targeted Customized Models**

Models like GPT-4o, Claude 3, and Gemini 1.5 are general purpose all-in-one foundation models. They are designed to perform a broad range of GenAI from coding to chat to summarization. The latest models have rapidly expanded to perform vision/image tasks, changing their function from just large language models to large multimodal models or vision language models (VLMs). Open source foundation models are headed in the same direction as integrated multimodalities.



*Figure 6. Advantages of general purpose vs. targeted customized models. Image credit: Intel Labs.*

However, rather than adopting the first wave of consumer-oriented GenAI models in this general-purpose form, most businesses are electing to use some form of specialization. When a healthcare company deploys GenAI technology, they would not use one general model for managing the supply chain, coding in the IT department, and deep medical analytics for managing patient care. Businesses deploy more specialized versions of the technology for each use case. There are several different ways that companies can build specialized GenAI solutions, including domain-specific models, targeted models, customized models, and optimized models.

*Domain-specific models* are specialized for a particular field of business or an area of interest. There are both proprietary and open source domain-specific models. For example, BloombergGPT, a 50B parameter proprietary large language model specialized for finance, [beats the larger GPT-3 175B parameter model](https://arxiv.org/pdf/2303.17564.pdf) on various financial benchmarks. However, small open source domain-specific models can provide an excellent alternative, as demonstrated by [FinGPT](https://arxiv.org/pdf/2306.06031.pdf), which provides accessible and transparent resources to develop FinLLMs. FinGPT 3.3 uses Llama 2 13B as a base model targeted for the financial sector. [In recent benchmarks](https://github.com/AI4Finance-Foundation/FinGPT), FinGPT surpassed BloombergGPT on a variety of tasks and beat GPT-4 handily on financial benchmark tasks like FPB, FiQA-SA, and TFNS. To understand the tremendous potential of this small open source model, it should be noted that FinGPT can be fine-tuned to incorporate new data for less than $300 per fine-tuning.

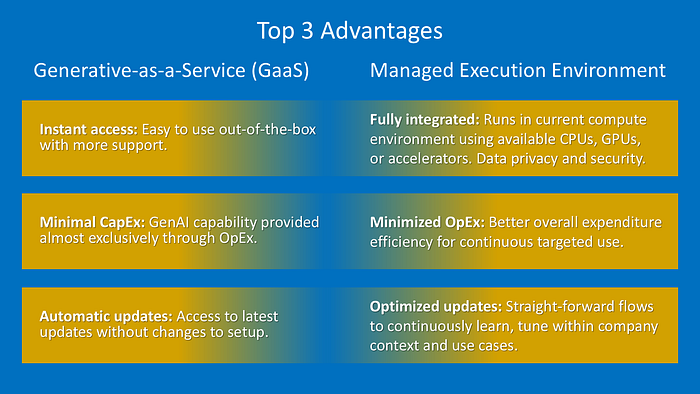
*Targeted models* specialize in a family of tasks or functions, such as separate targeted models for [coding](https://huggingface.co/docs/transformers/v4.39.0/en/model_doc/starcoder2), image generation, question answering, or sentiment analysis. A recent example of a targeted model is [SetFit](https://huggingface.co/blog/setfit) from Intel Labs, Hugging Face, and the UKP Lab. This few-shot text classification approach for fine-tuning Sentence Transformers is faster at inference and training, achieving high accuracy with a small number of labeled training data, such as only eight labeled examples per class on the Customer Reviews (CR) sentiment dataset. This small 355M parameter model can best the GPT-3 175B parameter model on the diverse RAFT benchmark.

It’s important to note that targeted models are independent from domain-specific models. For example, a sentiment analysis solution like [SetFitABSA](https://huggingface.co/blog/setfit-absa) has targeted functionality and can be applied to various domains like industrial, entertainment, or hospitality. However, models that are both targeted and domain specialized can be more effective.

*Customized models* are further fine-tuned and refined to meet particular needs and preferences of companies, organizations, or individuals. By indexing particular content for retrieval, the resulting system becomes highly specific and effective on tasks related to this data (private or public). The open source field offers an array of options to customize the model. For example, Intel Labs used direct preference optimization (DPO) to improve on a Mistral 7B model to create the open source [Intel NeuralChat](https://huggingface.co/Intel/neural-chat-7b-v3-1). Developers also can fine-tune and customize models by using low-rank adaptation of large language ([LoRA](https://arxiv.org/abs/2106.09685)) models and its more memory-efficient version, [QLoRA](https://arxiv.org/abs/2305.14314).

*Optimization capabilities* are available for open source models. The objective of optimization is to retain the functionality and accuracy of a model while substantially reducing its execution footprint, which can significantly improve cost, latency, and optimal execution of an intended platform. Some techniques used for model optimization include distillation, pruning, compression, and quantization (to 8-bit and even 4-bit). Some methods like mixture of experts (MoE) and [speculative decoding](https://arxiv.org/pdf/2211.17192.pdf) can be considered as forms of execution optimization. For example, [GPT-4 is reportedly comprised](https://the-decoder.com/gpt-4-has-a-trillion-parameters/) of eight smaller MoE models with 220B parameters. The execution only activates parts of the model, allowing for much more economical inference.

**Generative-as-a-Service Cloud Execution vs. Managed Execution Environment for Inference**



*Figure 7. Advantages of GaaS vs. managed execution. Image credit: Intel Labs.*

Another key choice for developers to consider is the execution environment. If the company chooses a proprietary model direction, inference execution is done through API or query calls to an abstracted and obscured image of the model running in the cloud. The size of the model and other implementation details are insignificant, except when translated to availability and the cost charged by some key (per token, per query, or unlimited compute license). This approach, sometimes referred to as a [generative-as-a-service (GaaS)](https://www.forbes.com/sites/steveandriole/2023/07/26/llama-chatgpt-bard-co-pilot--all-the-rest--how-large-language-models-will-become-huge-cloud-services-with-massive-ecosystems/?sh=78764e1175b7) cloud offering, is the principle way for companies to consume very large proprietary models like GPT-4o, Gemini Ultra, and Claude 3. However, GaaS can also be offered for smaller models like Llama 3.2.

There are clear positive aspects to using GaaS for the outsourced intelligence approach. For example, the access is usually instantaneous and easy to use out-of-the-box, alleviating in-house development efforts. There is also the implied promise that when the models or their environment get upgraded, the AI solution developers have access to the latest updates without substantial effort or changes to their setup. Also, the costs are almost entirely operational expenditures (OpEx), which is preferred if the workload is initial or limited. For early-stage adoption and intermittent use, GaaS offers more support.

In contrast, when companies choose an internal intelligence approach, the model inference cycle is incorporated and managed within the compute environment and the existing business software setting. This is a viable solution for relatively small models (approximately 30B parameters or less in 2024) and potentially even medium models (50B to 70B parameters in 2024) on a client device, network, on-prem data center, or on-cloud cycles in an environment set with a service provider such as a virtual private cloud (VPC).

Models like Llama 3.1 8B or similar can run on the [developer’s local machine](https://www.forbes.com/sites/steveandriole/2023/07/26/llama-chatgpt-bard-co-pilot--all-the-rest--how-large-language-models-will-become-huge-cloud-services-with-massive-ecosystems/?sh=78764e1175b7) (Mac or PC). Using optimization techniques like [quantization](https://www.intel.com/content/www/us/en/developer/articles/case-study/q8-chat-efficient-generative-ai-experience-xeon.html#gs.36q4lk), the needed user experience can be achieved while operating within the local setting. Using a tool and framework like [Ollama](https://ollama.ai/), developers can manage inference execution locally. Inference cycles can be run on legacy GPUs, [Intel Xeon](https://www.intel.com/content/www/us/en/products/docs/processors/xeon-accelerated/ai-accelerators-product-brief.html), or [Intel Gaudi AI accelerators](https://www.intel.com/content/www/us/en/products/details/processors/ai-accelerators/gaudi-overview.html) in the company’s data center. If inference is run on the model at a service provider, it will be billed as infrastructure-as-a-service (IaaS), using the company’s own setting and execution choices.

When inference execution is done in the company compute environment (client, edge, on-prem, or IaaS), there is a higher requirement for CapEx for ownership of the computer equipment if it goes beyond adding a workload to existing hardware. While the comparison of OpEx vs. CapEx is complex and depends on many variables, CapEx is preferable when deployment requires broad, continuous, stable usage. This is especially true as smaller models and optimization technologies allow for running advanced open source models on mainstream devices and processors and even local notebooks/desktops.

Running inference in the company compute environment allows for tighter control over aspects of security and privacy. Reducing data movement and exposure can be valuable in preserving privacy. Furthermore, a retrieval-based AI solution run in a local setting can be supported with fine controls to address potential privacy concerns by giving user-controlled access to information. Security is frequently mentioned as one of the top concerns of companies deploying GenAI and [confidential computing](https://www.intel.com/content/dam/www/public/us/en/documents/solution-briefs/intro-to-confidential-computing-solution-brief.pdf) is a primary ask. Confidential computing protects data in use by computing in an attested hardware-based [Trusted Execution Environment (TEE)](https://www.intel.com/content/www/us/en/content-details/788130/what-is-a-trusted-execution-environment.html).

Smaller, open source models can run within a company’s most secure application setting. For example, a model running on Xeon can be fully executed within a TEE with limited overhead. As shown in Figure 8, encrypted data remains protected while not in compute. The model is checked for provenance and integrity to protect against tampering. The actual execution is protected from any breach, including by the operating system or other applications, preventing viewing or alteration by untrusted entities.

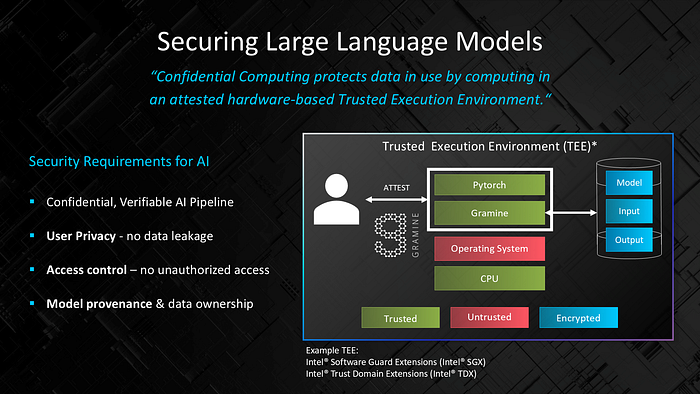


Figure 8. Security requirements for GenAI. Image credit: Intel Labs.

**Summary**

Generative AI is a transformative technology now under evaluation or active adoption by most companies across all industries and sectors. As AI developers consider their options for the best solution, one of the most important questions they need to address is whether to use external proprietary models or rely on the open source ecosystem. One path is to rely on a large proprietary black-box GaaS solution using RAG, such as GPT-4o or Gemini Ultra. The other path uses a more adaptive and integrative approach — small, selected, and exchanged as needed from a large open source model pool, mainly utilizing company information, customized and optimized based on particular needs, and executed within the existing infrastructure of the company. As mentioned, there could be a combination of choices within these two base strategies.

I believe that as numerous AI solution developers face this essential dilemma, most will eventually (after a learning period) choose to embed open source GenAI models in their internal compute environment, data, and business setting. They will ride the incredible advancement of the open source and broad ecosystem virtuous cycle of AI innovation, while maintaining control over their costs and destiny.

Let’s give AI the final word in solving the AI developer’s dilemma. In a [staged AI debate](https://pub.aimind.so/gpt-4-debates-open-orca-2-13b-with-surprising-results-b4ada53845ba), OpenAI’s GPT-4 argued with Microsoft’s open source Orca 2 13B on the merits of using proprietary vs. open source GenAI for future development. Using GPT-4 Turbo as the judge, open source GenAI won the debate. The [winning argument](https://youtu.be/JuwJLeVlB-w?t=774)? Orca 2 called for a “more distributed, open, collaborative future of AI development that leverages worldwide talent and aims for collective advancements. This model promises to accelerate innovation and democratize access to AI, and ensure ethical and transparent practices through community governance.”

**Learn More: GenAI Series**

[Knowledge Retrieval Takes Center Stage: GenAI Architecture Shifting from RAG Toward Interpretive Retrieval-Centric Generation (RCG) Models](https://towardsdatascience.com/knowledge-retrieval-takes-center-stage-183be733c6e8)

[Survival of the Fittest: Compact Generative AI Models Are the Future for Cost-Effective AI at Scale](https://towardsdatascience.com/survival-of-the-fittest-compact-generative-ai-models-are-the-future-for-cost-effective-ai-at-scale-6bbdc138f618)

[Have Machines Just Made an Evolutionary Leap to Speak in Human Language?](https://towardsdatascience.com/have-machines-just-made-an-evolutionary-leap-to-speak-in-human-language-319237593aa4)

**El Dilema del Desarrollador de IA: IA Propietaria vs. Ecosistema de Código Abierto**

**Elecciones fundamentales que afectan la integración y el despliegue a gran escala de GenAI en las empresas**

**Gadi Singer - 30 de septiembre de 2024**

Antes de que una empresa o un desarrollador adopte la inteligencia artificial generativa (GenAI), suele preguntarse cómo obtener valor empresarial de la integración de la IA en sus operaciones. Surge así una pregunta clave: ¿qué enfoque proporcionará el mejor retorno de inversión: un modelo propietario amplio y completo o un modelo de IA de código abierto que pueda adaptarse y afinarse a las necesidades específicas de la empresa?

Las estrategias de adopción de IA abarcan un amplio espectro: desde el acceso a un servicio en la nube basado en modelos avanzados y propietarios, como el GPT-4o de OpenAI, hasta la construcción de soluciones internas en el entorno de la empresa con modelos pequeños de código abierto que utilizan datos indexados de la compañía para tareas específicas. Las soluciones de IA actuales van más allá del modelo mismo e incluyen un ecosistema completo de sistemas de recuperación, agentes y otros componentes funcionales, como aceleradores de IA, útiles tanto para modelos grandes como pequeños. La aparición de colaboraciones intersectoriales, como la Plataforma Abierta para la IA Empresarial (OPEA), promete además agilizar el acceso y la estructuración de soluciones de código abierto de extremo a extremo.

Esta elección básica entre el ecosistema de código abierto y un entorno propietario afecta innumerables decisiones comerciales y técnicas, convirtiéndose en lo que se conoce como “el dilema del desarrollador de IA.” Creo que, para la mayoría de las implementaciones empresariales, inicialmente tiene sentido usar modelos propietarios para aprender sobre el potencial de la IA y minimizar los gastos de capital iniciales (CapEx). Sin embargo, para un despliegue amplio y sostenido, en muchos casos las empresas optarían por soluciones de código abierto basadas en un ecosistema que permitan una estrategia adaptable y rentable alineada con las necesidades comerciales y las tendencias de la industria.

**Transición de GenAI del Consumo al Despliegue Empresarial**

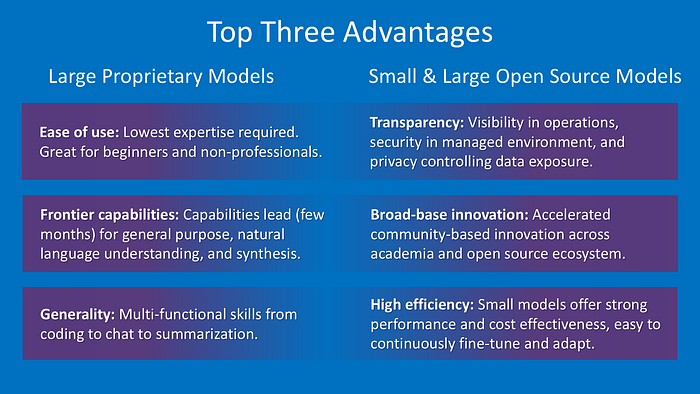
Cuando GenAI irrumpió en la escena a finales de 2022 con el GPT-3 de OpenAI y ChatGPT 3.5, generó principalmente interés entre los consumidores. A medida que las empresas empezaron a investigar sobre GenAI, en 2023 surgieron rápidamente dos enfoques para su implementación: el uso de modelos avanzados gigantes como ChatGPT frente a los nuevos modelos pequeños y de código abierto inspirados inicialmente en el modelo LLaMa de Meta. Para principios de 2024, estos dos enfoques básicos se habían consolidado. Con el enfoque de IA propietaria, la empresa depende de un modelo cerrado y grande para proporcionar todo el valor tecnológico necesario, como sería el caso del modelo GPT-4o de OpenAI.

En el enfoque de IA de código abierto, la empresa o desarrollador puede optar por un modelo de código abierto adecuado al tamaño necesario, utilizando datos corporativos o privados, funcionalidad personalizada y la seguridad y potencia de cómputo necesarias. Ambos caminos tienen ventajas y desventajas; no son excluyentes, y los desarrolladores pueden combinar componentes de ambos enfoques. Sin embargo, elegir una de estas rutas ofrece una estrategia empresarial con alta consistencia interna. Aunque se espera que ambos enfoques se desplieguen ampliamente, considero que, tras un periodo inicial de aprendizaje y transición, la mayoría de las empresas seguirán el enfoque de código abierto. Dependiendo del uso y el entorno, una IA interna de código abierto podría aportar importantes beneficios, incluida la capacidad de afinar el modelo y desplegarlo utilizando la infraestructura actual de la empresa, ejecutándolo en el cliente, en el centro de datos o como servicio dedicado. Con las nuevas herramientas de ajuste de IA, la experiencia técnica profunda ya no es una barrera tan significativa.

**Figura 1. Enfoques base del dilema del desarrollador de IA. Crédito de imagen: Intel Labs.**

A través de todas las industrias, los desarrolladores de IA están utilizando GenAI para diversas aplicaciones. Una encuesta de octubre de 2023 de Gartner mostró que el 55% de las organizaciones informaron haber incrementado su inversión en GenAI desde principios de 2023, y muchas están en fase piloto o de producción con esta tecnología en crecimiento. En el momento de la encuesta, las empresas invertían principalmente en el uso de GenAI para el desarrollo de software, seguido de cerca por las funciones de marketing y atención al cliente. Claramente, el rango de aplicaciones de IA está creciendo rápidamente.

**Modelos Propietarios Grandes vs. Modelos Abiertos Pequeños y Grandes**



**Figura 2: Ventajas de los modelos propietarios grandes y los modelos abiertos pequeños y grandes. Para consideraciones empresariales, consulte la Figura 7 para aspectos de CapEx y OpEx. Crédito de imagen: Intel Labs.**

En mi blog *La supervivencia del más apto: Los modelos de IA generativa compactos son el futuro para una IA rentable a gran escala*, proporciono una evaluación detallada de los modelos grandes frente a los pequeños. En esencia, tras la introducción del modelo LLaMa de código abierto de Meta en febrero de 2023, ha habido un ciclo virtuoso de innovación y mejora rápida en el que la academia y el ecosistema amplio están creando modelos altamente efectivos que son de 10 a 100 veces más pequeños que los modelos avanzados. Los modelos compactos, que en 2024 tenían en su mayoría menos de 30 mil millones de parámetros, lograron igualar las capacidades de los modelos grandes estilo ChatGPT, que contenían más de 100 mil millones de parámetros, especialmente cuando se enfocan en dominios específicos.

Además, los modelos de código abierto suelen estar solo de seis a doce meses detrás en rendimiento respecto a los modelos propietarios. Utilizando el amplio punto de referencia de lenguaje MMLU, la mejora en el ritmo de los modelos de código abierto es más rápida y la brecha parece estar cerrándose. Por ejemplo, el GPT-4o de OpenAI fue lanzado en mayo de 2023 con funciones multimodales importantes, mientras que el pequeño modelo de código abierto Phi-3-vision de Microsoft se presentó solo una semana después, mostrando competencias similares en pruebas iniciales de reconocimiento visual y comprensión.

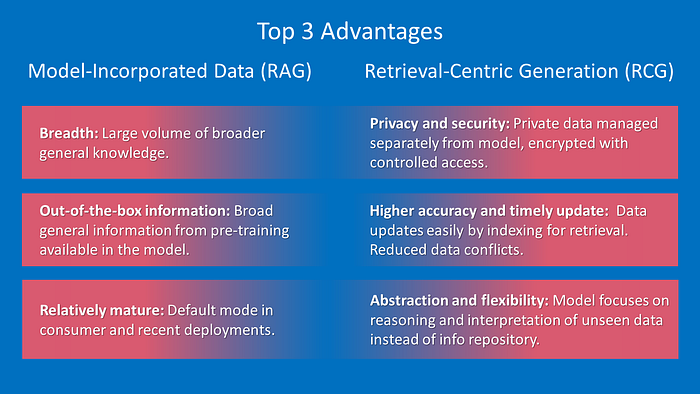
Los modelos grandes ofrecen una versatilidad completa. Los desarrolladores pueden elegir entre una variedad de modelos de GenAI propietarios comercialmente disponibles, incluidos el modelo multimodal GPT-4o de OpenAI, el modelo Gemini 1.5 de Google, disponible en cuatro tamaños, y el modelo Claude 3 Opus de Anthropic, que se rumorea tiene aproximadamente 2 billones de parámetros y una ventana de contexto de 200K tokens, permitiendo cargar grandes cantidades de información. También existe otra categoría de modelos grandes de GenAI para productividad y desarrollo creativo de empleados, como el Microsoft 365 Copilot, que integra la suite de aplicaciones de Microsoft 365.

La mayoría de los modelos de código abierto, tanto grandes como pequeños, suelen ser más transparentes sobre los marcos de aplicación, el ecosistema de herramientas, los datos de entrenamiento y las plataformas de evaluación. Esta transparencia permite a los desarrolladores intercambiar fácilmente modelos para futuras versiones.

**Datos Incorporados en el Modelo (con RAG) vs. Generación Centrada en Recuperación (RCG)**

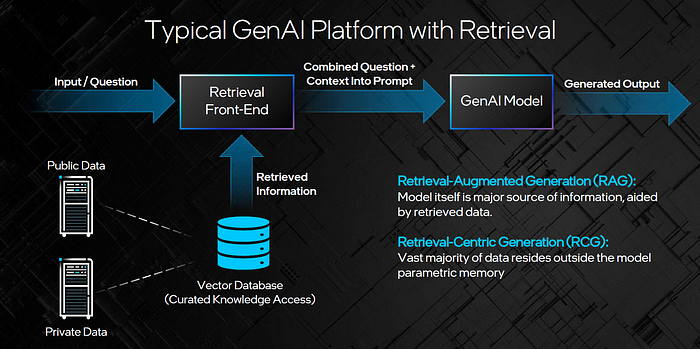
La próxima pregunta clave para los desarrolladores de IA es dónde encontrar los datos utilizados durante la inferencia: dentro de la memoria paramétrica del modelo o fuera de él (accesible mediante recuperación). Aunque resulte difícil de creer, el primer ChatGPT lanzado en noviembre de 2022 no tenía acceso a datos externos al modelo. Este importante problema fue abordado en 2023 con la adición de complementos de recuperación. Hoy en día, la mayoría de los modelos están acoplados a un front-end de recuperación, salvo en casos donde no se espera acceder a gran cantidad de información actualizada constantemente.

Los modelos actuales han progresado significativamente en este aspecto, al mejorar las plataformas de solución con un front-end de generación aumentada por recuperación (RAG), lo que permite extraer información externa al modelo. Un RAG eficiente y seguro es un requisito en el despliegue empresarial de GenAI, como se evidencia en la introducción de GPT-RAG por Microsoft a finales de 2023.



**Figura 3. Ventajas de RAG frente a RCG. Crédito de imagen: Intel Labs.**

Los modelos RCG pueden definirse como un caso especial de soluciones GenAI de RAG diseñadas para sistemas donde la gran mayoría de los datos reside fuera de la memoria paramétrica del modelo y no es visto en preentrenamiento ni en ajuste fino.



**Figura 4. Cómo funciona la recuperación en plataformas de GenAI. Crédito de imagen: Intel Labs.**

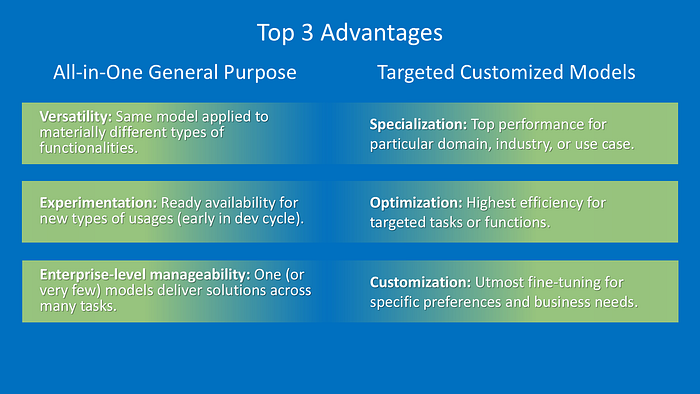
OPEA es un esfuerzo de colaboración inter-ecosistemas para facilitar la adopción y ajuste de sistemas de GenAI. Utilizando este marco componible, los desarrolladores pueden crear y evaluar “soluciones GenAI abiertas, de múltiples proveedores, robustas y componibles que aprovechen la mejor innovación de todo el ecosistema.” Se espera que OPEA simplifique la implementación de soluciones GenAI compuestas de nivel empresarial, que incluyan RAG, agentes y sistemas de memoria.



**Figura 5. Principios básicos de OPEA para la implementación de GenAI. Crédito de imagen: OPEA.**

**Modelos Todo-en-Uno de Uso General vs. Modelos Personalizados y Específicos**

Modelos como GPT-4o, Claude 3 y Gemini 1.5 son modelos de base de propósito general. Están diseñados para realizar una amplia gama de tareas de GenAI, desde programación hasta chat y resumen de textos. Los modelos más recientes han expandido rápidamente su capacidad para realizar tareas visuales, transformándose de modelos de lenguaje grandes a modelos multimodales o modelos de lenguaje visual (VLMs). Los modelos de código abierto también están avanzando hacia la integración de multimodalidades.



**Figura 6. Ventajas de modelos de uso general vs. modelos personalizados específicos. Crédito de imagen: Intel Labs.**

Sin embargo, en lugar de adoptar estos modelos GenAI de propósito general, la mayoría de las empresas eligen una forma de especialización. Por ejemplo, una empresa de atención médica no utilizaría un modelo general para gestionar la cadena de suministro, programar en el departamento de TI y realizar análisis médicos profundos para la atención al paciente. Las empresas implementan versiones especializadas de la tecnología para cada caso de uso. Existen varias formas en que las empresas pueden construir soluciones GenAI especializadas, incluyendo modelos específicos de dominio, modelos dirigidos, modelos personalizados y modelos optimizados.

Los **modelos específicos de dominio** se especializan en un campo de negocios o área de interés particular. Existen tanto modelos de dominio específicos propietarios como de código abierto. Por ejemplo, BloombergGPT, un modelo propietario de lenguaje grande de 50 mil millones de parámetros especializado en finanzas, supera al modelo más grande GPT-3 de 175 mil millones de parámetros en varios puntos de referencia financieros. Sin embargo, los modelos de código abierto de menor tamaño también pueden ser una excelente alternativa. FinGPT, un modelo basado en Llama 2 de 13 mil millones de parámetros, orientado al sector financiero, ha demostrado superar a BloombergGPT y a GPT-4 en varias tareas de referencia financiera.

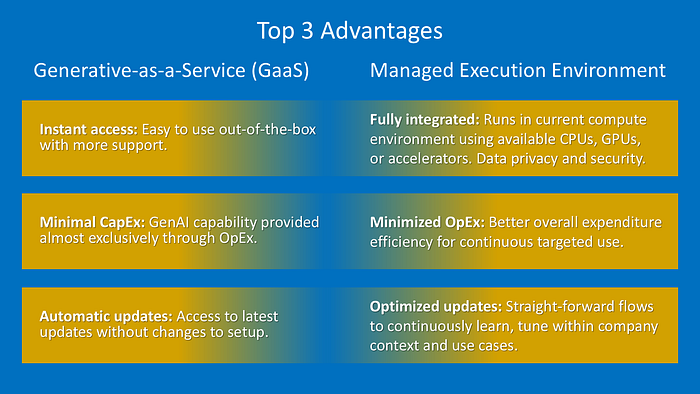
**Modelos dirigidos** se especializan en una familia de tareas o funciones, como modelos para programación, generación de imágenes, respuesta a preguntas o análisis de sentimientos. Un ejemplo reciente es SetFit, una solución de clasificación de texto de pocas muestras que utiliza transformadores de oraciones, optimizando la precisión con un número reducido de datos etiquetados. Este modelo compacto de 355 millones de parámetros puede superar al modelo GPT-3 en diversos puntos de referencia.

Es importante destacar que los **modelos dirigidos** son independientes de los **modelos específicos de dominio**. Por ejemplo, una solución de análisis de sentimientos puede aplicarse en múltiples dominios, como industrial, entretenimiento u hospitalidad. Sin embargo, los modelos que son tanto específicos de dominio como dirigidos pueden ser aún más efectivos.

Los **modelos personalizados** son ajustados para cumplir con las necesidades particulares de empresas u organizaciones. Al indexar contenido específico, el sistema resultante se vuelve altamente efectivo en tareas relacionadas con estos datos. En el campo de código abierto, hay diversas opciones para personalizar modelos. Por ejemplo, Intel Labs utilizó la optimización de preferencias directas (DPO) para mejorar el modelo Mistral 7B y crear NeuralChat de Intel.

Las capacidades de **optimización** están disponibles en modelos de código abierto, reteniendo funcionalidad y precisión mientras se reduce el costo de ejecución. Entre las técnicas de optimización se incluyen destilación, poda, compresión y cuantización (a 8 bits e incluso a 4 bits). Algunas técnicas, como la mezcla de expertos (MoE) y la decodificación especulativa, se consideran formas de optimización de ejecución.

**Ejecución de Generative-as-a-Service en la Nube vs. Entorno de Ejecución Gestionado para Inferencia**

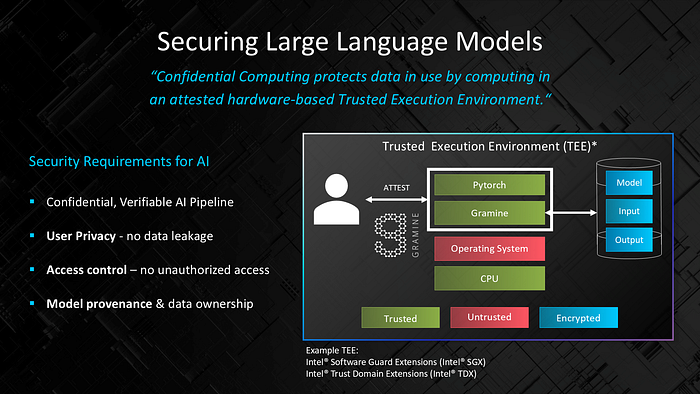


**Figura 7. Ventajas de GaaS vs. ejecución gestionada. Crédito de imagen: Intel Labs.**

Otra decisión clave para los desarrolladores es el entorno de ejecución. Si la empresa elige un modelo propietario, la ejecución de inferencia se realiza mediante API o consultas a un modelo en la nube. Este enfoque, conocido como generative-as-a-service (GaaS), es la forma principal en que las empresas consumen grandes modelos propietarios. También puede ofrecerse GaaS para modelos más pequeños como Llama 3.2.

Existen claros beneficios al usar GaaS. Por ejemplo, el acceso es generalmente instantáneo y fácil de usar, eliminando esfuerzos de desarrollo interno. Los costos suelen ser casi en su totalidad de gastos operativos (OpEx), lo cual es ideal para cargas iniciales o uso intermitente.

En contraste, si las empresas eligen una inteligencia interna, el ciclo de inferencia se incorpora en el entorno de computación propio. Este es un enfoque viable para modelos relativamente pequeños (aproximadamente 30 mil millones de parámetros o menos en 2024) y hasta modelos medianos en dispositivos cliente, redes, centros de datos o en entornos de nube privada. Modelos como Llama 3.1 8B pueden ejecutarse en máquinas locales, y el uso de técnicas de optimización como la cuantización permite la ejecución local en dispositivos convencionales.



**Figura 8. Requisitos de seguridad para GenAI. Crédito de imagen: Intel Labs.**

Ejecutar la inferencia en el entorno computacional propio permite un control más estricto sobre aspectos de seguridad y privacidad. Reducir el movimiento de datos puede ser útil para preservar la privacidad. Además, una solución de IA basada en recuperación en un entorno local puede contar con controles específicos para abordar preocupaciones de privacidad. La **computación confidencial** es una prioridad, protegiendo los datos en uso al computar en un Entorno de Ejecución Confiable (TEE) basado en hardware.

**Resumen**

La IA generativa es una tecnología transformadora que ahora se está evaluando o adoptando activamente en múltiples sectores. Al considerar sus opciones, una de las preguntas más importantes es si usar modelos propietarios externos o depender del ecosistema de código abierto. Un camino es confiar en una solución GaaS como GPT-4o o Gemini Ultra. El otro camino utiliza un enfoque más adaptativo e integrador — modelos pequeños, seleccionados y cambiados según sea necesario, basados en la información de la empresa y ejecutados en la infraestructura existente. La mayoría de los desarrolladores eventualmente optarán por integrar modelos GenAI de código abierto en sus entornos internos, beneficiándose del avance del ciclo de innovación del ecosistema de código abierto.