zeenea

5 TECHNOLOGICAL BREAKTHROUGHS OF A NEXT-GEN DATA CATALOG

"Developing any technological product requires making a number of critical architectural and design choices early on. In this piece, I will expand on some of the choices we made at Zeenea which, we feel, set the ground for a next generation data catalog - automated, smart and simple."

- Guillaume Bodet, CPTO

zeenea

Be Data Fluent

www.zeenea.com - info@zeenea.com

SUMMARY

INTRODUCTION *1*

TECHNOLOGICAL BREAKTHROUGH #1
A MULTI-TENANT CLOUD ARCHITECTURE *7*

TECHNOLOGICAL BREAKTHROUGH #2
A PROPRIETARY, DISTRIBUTED, OPEN AND UNIVERSAL CONNECTIVITY *13*

TECHNOLOGICAL BREAKTHROUGH #3
A PLATFORM ARCHITECTURE *16*

TECHNOLOGICAL BREAKTHROUGH #4
A KNOWLEDGE GRAPH AT THE CORE *21*

TECHNOLOGICAL BREAKTHROUGH #5
A MULTI-DIMENSIONAL SEARCH ENGINE *23*

TAKE AWAY *27*

INTRODUCTION

Developing any technological product requires making a number of critical architectural and design choices early on that will inevitably have an impact on its ability to meet market and user expectations. Software architecture is one of the main levers of execution for technology companies.

When we founded Zeenea, our goal was to build a world leading data catalog. It still is. To be fair, this is probably the goal for anyone looking to offer a new technology product.

We started this adventure with a common vision, a solid financial situation and an accumulated 50 years of experience in innovation and data - an ideal set of circumstances to start a great project.

In this piece, I will expand on some of the choices we made early on which, we feel, **set the ground for a next generation data catalog -automated, smart and simple.**

Of course, these choices were not made arbitrarily. On the contrary, their purpose was to meet on one hand, what we see as the inherent objective of a data catalog (those that involve data exploration - we will come back to this) and on the other hand, the challenges of enterprise-wide data catalog implementation.

To place these choices in the wider context, let's first look at the data cataloging lay of the land as it stands today.

METADATA MANAGEMENT: A CHANGING DISCIPLINE

Metadata management, which is at the heart of the capabilities of a data catalog, is nothing new. For a long time, it was limited to data mapping for IT teams or niche audit trails aimed at ensuring regulatory compliance in specific fields (banking, finance, insurance and health for instance).

A couple of recent developments have turned the discipline on its head:

1. **Influenced by the giant corporations from the US, organizations have become aware of the value their data holds.** Many have launched ambitious digital transformation programs to leverage this data in order to strengthen or maintain their market position. They strive to achieve this, firstly, by optimizing their operational processes through synthetic indicators, with the use of quantitative analysis and predictive models, and, secondly, by developing new products/services to help them monetize their data.

2. **Digital transformation has itself disrupted the information landscape in all sectors of activity by multiplying the volume and variety of data available** (big data platforms, IoT data, web profiling, social network data along with the exponential growth of Open Data, all of which organizations seek to exploit). As a result of digital transformation, the number of data users interested in the informational legacy has grown considerably. The pool of stakeholders is no longer limited to IT or compliance teams but also includes new data experts (data scientists and data analysts) alongside the traditional divisions (product, marketing, finance, human resources, client relations, logistics, etc.)

In this new context, metadata management has lost its tactical importance and endorsed a larger strategic dimension.

Invariably, these digital transformations have led to a multiplication of the number of usable data sets, as well as the number of people involved in their consumption and production. These data sets, however, often remain opaque and the potential they hold cannot be leveraged because of a dearth in available resources for identifying, localizing and understanding them.

The solution to this issue isn't new: it involves building a registry of the available data sets, documenting them with the help of the metadata, and providing the data consumers with a data catalog that enables them to search and view the information they need.

The metadata provides the information that is crucial to the use and understanding of the data: they can cover some or all of the following:

- **Technical information:** storage systems, location, permissions, formats, types, etc.
- **Statistical information:** volume, distribution, number of null, minimal and maximal values, etc.
- **Information on the life cycle of the data:** age, update frequency, quality, origin, lineage, impact, retention, etc.
- **Semantic information:** definitions, relation with business concepts, classifications, etc.
- **Organizational information:** individuals or services in charge of any given data set, or people who know what it contains.
- **Usage information:** services, products, type of usage, relationship with other data sets, etc.
- **Compliance information:** PII, sensitive information, level of confidentiality, etc.
- And more generally any **type of information that is considered important to understand and use the data.**

THE LIMITS OF A DECLARATIVE (TRADITIONAL) APPROACH TO METADATA MANAGEMENT

For Zeenea, traditional metadata management and data cataloging solutions are not designed to properly fulfill their roles in a rapidly evolving digital landscape. The reason for this: these solutions were built to cater for a small number of experts for which the parameters were relatively narrow. As a result, these solutions focus on a declarative approach (each data set has to be recorded and described manually) and have limited exploration and search capabilities. This actually makes sense when the stakeholders understand their perimeter enough to find what they want without needing any deeper exploration.

For us, this approach is obsolete and doesn't address the digital challenges organizations are facing today. **The exponential number of data sets, their fast-paced life cycles and the growing number of users needing to exploit them completely disqualifies this purely declarative approach.**

OUR BOTTOM-UP APPROACH TO DATA CATALOGING, GUIDED BY STRONG PRODUCT VALUES

Unconvinced by the efficacy of the more traditional solutions on the market, we opted for a bottom-up approach to the building of the data catalog: the physical elements from the catalog (data sets, dashboards, models, treatments, etc), along with the metadata, are collected and synchronized automatically in the catalog. The other layers of the catalog (semantic and business layers) are modeled, and algorithms can then attach them to the physical layers.

Enacting this bottom-up approach drives both our mission and our architectural choices.

The other pillar centers around our product values, and they are essential to software development. In an ocean of possible solutions to any given problem, they are able to pick out those that are best adapted to a stable and long lasting value system. It is these product values that will justify the choice made in the long term. For Zeenea, these choices are as follows:

Simplicity: the product must be simple to deploy, easy to configure, easy to use. This is much easier said than done and could constitute an entire article in and of itself…

Flexibility: the product must be adaptable to any context - freedom has to be the norm in both the modeling and the data catalog enterprise deployment trajectory.

Deep Tech: algorithmic solutions (be they basic or more complex), with the aid of machine learning or semantic analysis for example, are preferable to the declarative approaches referenced above which are a throwback from IT management.

Below are the 5 essential choices we made for the Zeenea architecture, which we feel represent a technology breakthrough in the data cataloging field:

- **A multi-tenant Cloud Architecture**
- **An open and universal proprietary connectivity**
- **A platform architecture**
- **A knowledge graph at the core**
- **A multidimensional search engine**

5 TECHNOLOGICAL BREAKTHROUGHS OF A NEXT-GEN DATA CATALOG

TECHNOLOGICAL BREAKTHROUGH #1

A MULTI-TENANT CLOUD ARCHITECTURE

This is probably the most structuring choice in the initial phases of the development of a solution. Should the solution be purely cloud based? On-premise? Both? Since it would be incredibly difficult if not impossible to change later on, this choice is crucial. The refactoring costs would be huge and the impact would be felt across all operational aspects - development processes, version management, operations costs, pricing, business model, etc. Software vendors that have managed the switch from On-prem to the Cloud are few and far between, and their success involved rewriting their solution from scratch. No migrations from the Cloud to On-prem spring to mind.

Opting for a Cloud solution therefore has major consequences.

CHOOSING A CLOUD ARCHITECTURE AND ITS BENEFITS

We decided to rely on a pure multi-tenant Cloud architecture, meaning that certain application layers are mutualized amongst different clients. This allows for considerable flexibility when scaling.

I won’t elaborate too much on all the benefits of the Cloud here, there is already plenty of content on the subject. I’ll nonetheless highlight 3 that are relevant to the product values mentioned above:

- **Operational costs**, both for the provider and the client, are somewhat lower than those of on-prem alternatives. It allows for a simpler and more attractive pricing model, and enables our clients to start quickly with a lower priced entry ticket ; and to roll out the solution progressively with greater ease and flexibility.

- Deploying with a new client takes a few minutes. **It ensures a rapid, straightforward and user-friendly onboarding for new users.**

- The base of the code being the same for every client, everyone benefits from any corrections made, product improvements or new functionalities. **It's the simplest way for our clients to keep their solution up to date.**

DEPLOYING A CLOUD DATA CATALOG IS NOT WITHOUT RISK

Although very appealing, a Cloud solution does carry some risk and can have some undesirable consequences that should be addressed early in the product life cycle.

The first risk is market acceptance. Even if it goes against the grain, some organizations still refuse Cloud solutions. For Zeenea, this particular risk is easy to accept. The data cataloging market spans the globe and there is a large enough pool of clients to ensure our success. In order for any Cloud solution to be accepted however, there still needs to be a thorough security and data confidentiality evaluation.

The second risk involves the actual choice of the Cloud provider, and that this choice will be restrictive. Indeed, investing in a portable architecture isn't easy. Although the services offered by the different providers are often very similar, the APIs won't meet any specific established standard. It is therefore difficult to find the experts able to build a robust architecture on multiple Cloud platforms. The risks here can come from multiple directions:

- Dependency on the Cloud provider will likely be strong, and his pricing model could change drastically and have a negative impact on the business model of the organization (we see this as a low risk at this stage).

- With the glaring lack of European Cloud providers, one has to turn to US (or Chinese) providers that can sometimes carry totally iniquitous regulations (the Patriot Act and the Cloud Act for instance).

- Lastly, some clients can object to the choice of the provider, either because it differs from their own choice, or simply by principle. AWS for example has a bad press in a distribution sector, in which Amazon is increasing its own market share.

All cloud providers rely on the principle of **the Shared Responsibility Model**. In practice, this means the providers make available highly secured and resilient components. However, they refuse any responsibility for the levels of security and resilience of the architectures that were developed around those components ; with the following consequence: building a state of the art Cloud architecture requires an incredibly high level of expertise for a sensitive strategic issue to the organization. Externalizing this expertise is therefore far from ideal and suffice it to say that finding Cloud architecture experts is a challenge…

INTEGRATING SECURITY AND PRIVACY ASPECTS FROM THE EARLIEST STAGES OF PRODUCT DESIGN

To mitigate these different risks, we therefore took a number of pivotal, and sometimes counter intuitive decisions. These often required an initial investment which might appear excessive when the solution still hasn't found its market, and when each expenditure expects a quick return on investment.

We chose Amazon as our Cloud provider (AWS). It was the most mature solution on the market at the time and, more importantly, it was the cloud technology we had the highest level of internal expertise on. To limit any dependency on AWS services, we decided not to integrate proprietary Amazon services in our product. **Our application components are therefore entirely independent from the AWS services that support them.** This rule of thumb is integrated in our development norms and guarantees a smooth and affordable portability to another Cloud provider should the need ever arise. We therefore use AWS very much like an IaaS (Infrastructure as a Service) rather than a PaaS (Platform as a Service). Amazon provides the infrastructure services (network, storage, routage, load-balancing, supervision, logs, etc.), but no application services. The only fly in the ointment: our DevOps tooling is very much adapted to the AWS API's. We chose to accept this situation even if the redevelopment costs of the tooling is relatively high. Who knows, by the time we need to diversify our offer, interoperability solutions may be available.

The second consequence of choosing a Cloud solution was a need to drastically **reinforce our risk management, security, availability and confidentiality practices.** The idea was to provide our clients with the guarantees they are entitled to from a Cloud solution provider.

This reinforcement of our practices led to two main initiatives:

- The integration of security and confidentiality aspects from the very first stages of the product architecture and conception: security and privacy by design.

- Getting the SOC 2 certification (which we have had since December 2020), to which we will very shortly add the ISO 27001 certification.

The impacts of our choices on the product architecture and conception are numerous and it would be difficult to list them all. A few examples will however give you the gist of it:

- Even if the architecture is multi-tenant, **each client has dedicated storage space.** It allows them, firstly, to ensure that their data can be destroyed, saved and restored with ease ; and secondly, to manage dedicated encryption keys that allow for greater security levels.

- On the same theme, **end-to-end data encryption is ensured** (transit, storage and backup), de facto barring access to the Cloud provider.

- We do not store anything secret on the platform. **The authentication keys for the systems to which we are connected to are managed by agents**, who are deployed on our clients systems and are under their supervision.

- Some functionalities like sampling and profiling are specifically conceived to ensure the highest level of guarantees in terms of security and confidentiality.

SOC 2 compliance (and ISO 27001) was another investment we chose to enact early on. It was a considerable investment for a startup like Zeenea because of what it entails. Indeed, SOC 2 has over 300 controls that deal with the technical architecture, access management, development processes but also all corporate operational processes (HR, finance, sales, marketing, client relations, etc.). The impact is therefore enormous since all these controls have to be put in place, reviewed, piloted and improved over time.

Compliance takes time. It took us 12 months and, in spite of our full Cloud offering, Zeenea is regularly selected by clients the world over, especially clients for whom security is a priority.

TECHNOLOGICAL BREAKTHROUGH #2

A PROPRIETARY, DISTRIBUTED, OPEN AND UNIVERSAL CONNECTIVITY

In order to enact this vision of a data catalog built around the content of the operational systems (bottom-up approach), Zeenea is quite naturally a connected catalog. Connected to the information systems of its clients.

There are many ways to conceive a system's connectivity. For Zeenea, we chose the following attributes:

- **It should be proprietary** - we do not rely on a third party solution.
- **It should be distributed** - in order not to limit the catalog reach.
- **It should be open** - whoever wants to enrich the catalog can develop his own connectors.
- **It should be universal** and able to synchronise any metadata source.

IN-HOUSE CONNECTIVITY (PROPRIETARY)

It isn’t necessary to develop one’s own connectivity layer. There are some ready made solutions, either as a Cloud service (but the reach is limited to the other cloud systems), or as software components. Such components already exist and most data catalog providers fall back on them.

Unfortunately, they are not conceived specifically for the requirements of a data catalog but rather to cover a much larger span of use cases. This leads to a complexity that we find unhelpful.

To set up the simple SDK we wanted to provide the developers with, we opted for a simple connectivity layer. And there is no such ready-to-use product, it must be developed in-house.

DISTRIBUTED ARCHITECTURE VIA AGENTS

This question hardly needed to be raised. A Cloud architecture having been decided upon, a centralized connectivity would have limited the reach to other Cloud systems, as well as raised security issues to keep the secrets of the connections (passwords and API keys). And our goal was to catalog everything: data often originates from non-cloud systems as well.

The solution was therefore to adopt a distributed architecture: agents (small services deployed in the client infrastructure) connect to the operational systems, including those on the Cloud, to collect the information they need; and then send them on the Cloud platform. **The agents communicate with the platform, but the platform cannot contact them.** No need for a VPN. It is not necessary to open incoming data feeds.

These agents are of course themselves designed for easy installation, configuration and supervision: ops teams are also our users and we owe them the best possible experience.

Another advantage of having these agents placed under the total control of our clients lies in the fact that they can also be authorized to gather a wider set of information (sampling, data profiling etc.). All invaluable capabilities for a smart data catalog (we’ll come back to that further down).

OPEN CONNECTIVITY

The reason for an open connectivity is pragmatic: There are hundreds of systems a data catalog can be called upon to connect to. And each connection can cause versioning or configuration issues. With this in mind, it's difficult to imagine that we could manage this universe alone, without help from partners, integrators, even clients.

This openness happens in 2 distinct ways:

- We deliver **a streamlined SDK**, easy to understand and handle, in order to focus on metadata extraction, rather than their integration into the catalog.
- **The agent adds a plugin micro-architecture**, giving the possibility to package a connector and deploy it seamlessly by sticking the file in a repository. We provide of course a sandbox for the testing.

UNIVERSAL CONNECTIVITY

Another consequence of our data catalog vision: we do not wish to connect exclusively to the storage systems on the Cloud (or otherwise) - databases, data lakes, data warehouses, etc. Rather, **we wish to connect to any system capable of producing metadata.** Storage systems of course, but also ETLs, ERPs, BI platforms, quality monitoring tools, business modeling tools, etc.

Our SDK therefore rests on certain simple concepts, easily understandable that virtually enables us to collect any type of information, and then integrate it in the catalog by linking it to the objects already present.

TECHNOLOGICAL BREAKTHROUGH #3

A PLATFORM ARCHITECTURE

The architectural styles are numerous and opting for the Cloud changes nothing. Should one choose a monolithic application, easier to develop in the short term, but difficult to evolve later on? Or should one invest from the start in a microservice architecture, perfectly adapted to the Cloud but that requires a longer time to build and is harder to design?

We chose neither.

We did not conceive our architecture to handle questions of velocity or even scalability. Rather, we conceived it to align with one of the components of our product vision.

We believe that **a data catalog is by essence a cross-functional tool**, with a diverse set of users, all with varying needs (data management teams, engineers and architects, data scientists, business analysts, project/product managers, compliance teams, DPOs, etc.). And we do not believe it is possible to meet the expectations of all these people with the same and only user experience, not if we want to ensure simplicity which is at the heart of our product values.

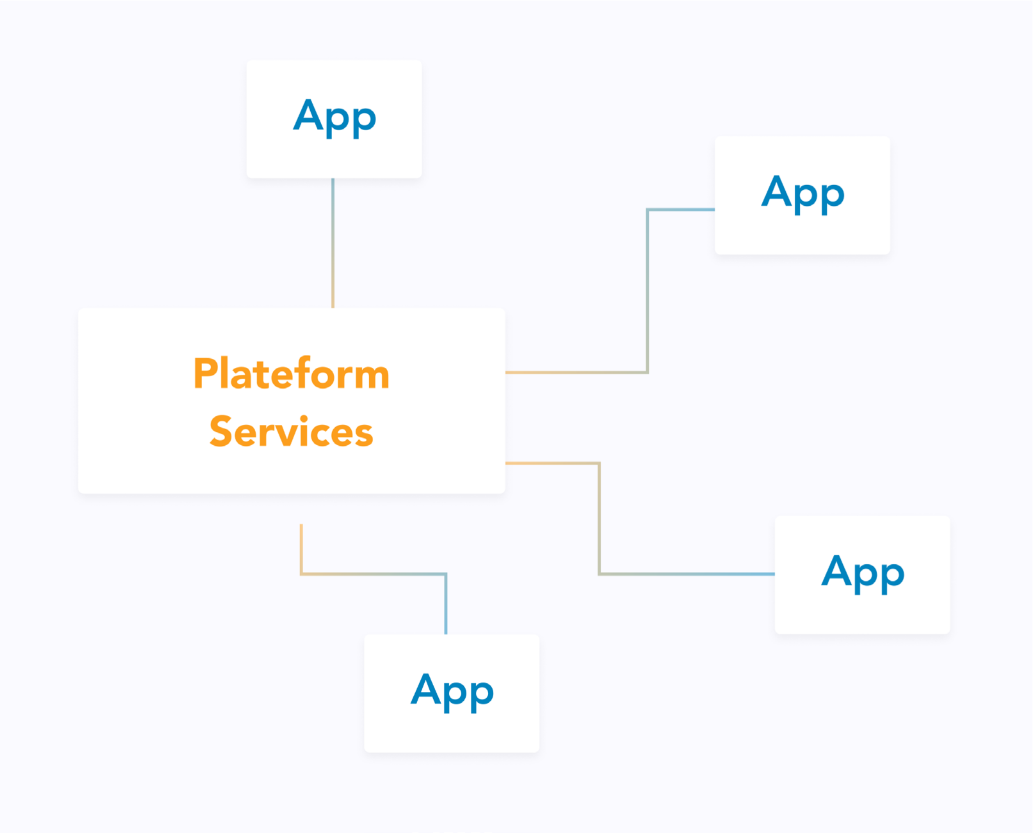
APPLICATIONS DEDICATED TO THE DIFFERENT USES OF A DATA CATALOG

The solution, naturally, is to offer not just one application, but several, each handling a number of specific use cases, but with a user experience built specifically for those use cases.

We found inspiration in things such as the architecture of large market places or mobile reservation platforms for carpooling services (or the like). Their architectures rest on the same principle: at the heart, **a component (the platform)** that concentrates on the most abstract capabilities - essential data management, optimizations, special treatments, event buses, permissions, etc.

Then around it, a myriad of specialized applications consuming the services of this platform are integrated. Let's take a marketplace for instance: a back-office tool to manage and control the catalog content, an e-commerce website, a mobile app, marketing tools, professional client applications, etc.

The schema below illustrates this principle :



It's worth pointing out that the platform, and even some applications, are not necessarily monolithic, they can easily be broken down, so long as the overall schema remains the same: applications depend solely on the platform, never on other apps.

It is this approach that we have chosen at Zeenea as it matches our vision of a data catalog best.

zeenea
Be Data Fluent

THE CENTRAL ROLE OF THE PLATFORM

Our platform has the following core responsibilities:

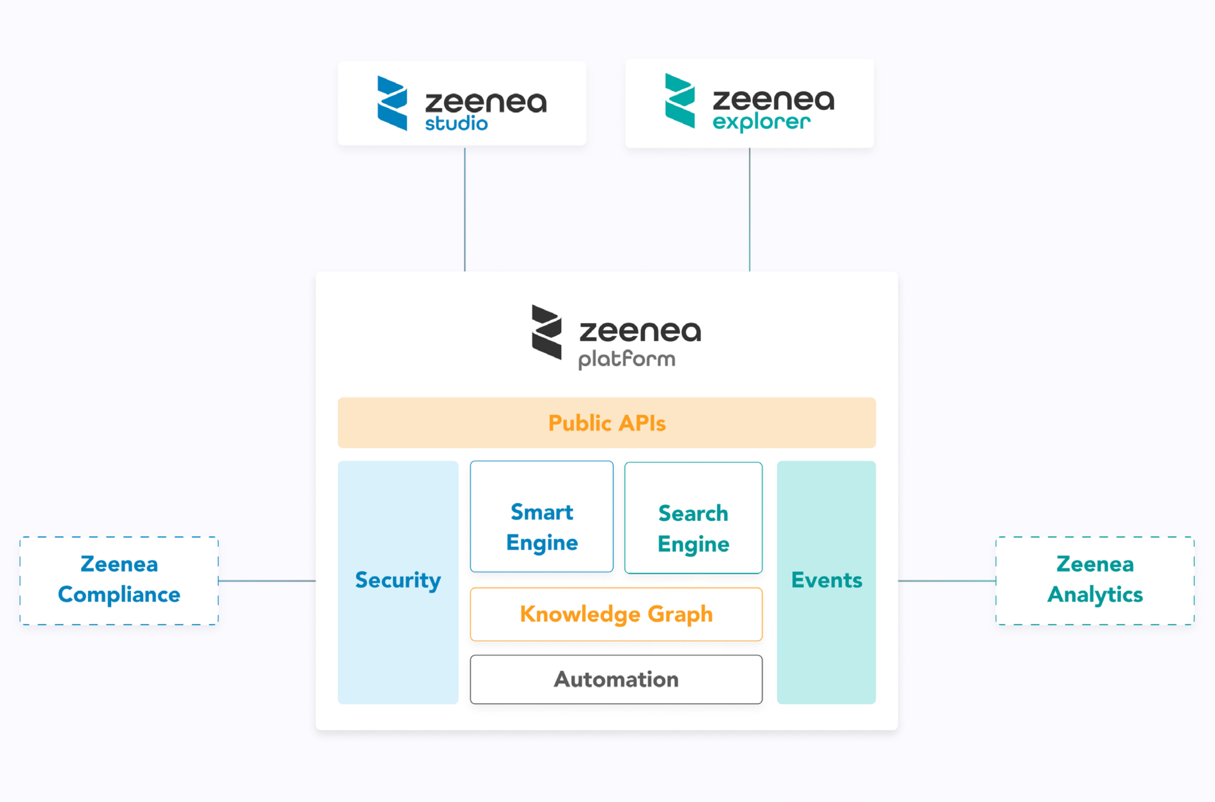
- It stores, enriches and modifies the graph database which is the core of the catalog and the pillar of our knowledge graph.
- It manages the interface of our connectivity system - monitoring the agents, handling incoming data to connect them to the rest of the catalog, managing errors, etc.
- It takes on the intelligent algorithms (link resolutions, similarity detection, auto-tagging, semantic suggestions, etc.)
- It feeds the search engine – which we will discuss further down.
- It manages the users and ensures that the configured permissions are adhered to in the system.
- It manages events and guarantees traceability.
- It provides internal and public APIs.

To date, there are 2 applications:

Zeenea Studio: this tool is for the data management teams. The application enables **the design of the catalog structure, to feed it and monitor the quality of its content.** It is designed as a productivity tool.

Zeenea Explorer: this tool enables a **simplified search and exploration experience.** The application is aimed primarily at direct data consumers - data analysts, data scientists, data engineers, etc.

ZEENEA PLATFORM



AN ARCHITECTURE OPEN TO FUTURE USE CASES (APPLICATIONS)

These 2 applications do not completely cover the other populations of users in the catalog and we are aware of this. But, we are also aware that, given the way **the architecture is conceived, we will be able to develop new applications very quickly, address new uses,** simply by using the services offered by the platform, and by focusing on the most important thing: user experience.

We already have some thoughts: a BCBS 239 compliance application (Zeenea Compliance), an application aimed at personal data managers (Zeenea Privacy), a catalog content analysis application (Zeenea Analytics)…They will all take form in the near future.

In the same spirit, this architecture allows the integration of the catalog to third party tools with great ease (data preparation tools, visualization tools, modeling tools, etc.). Indeed, each integration can be seen as a micro-application on top of the platform.

Lastly, as the platform boasts of a list of APIs conceived in part for application development, it is very easy for another party to build his own applications – for instance to integrate certain capabilities of Zeenea in a portal, or suggest a new specific usage.

TECHNOLOGICAL BREAKTHROUGH #4

A KNOWLEDGE GRAPH AT THE CORE

A knowledge graph does not appear to have a formal definition. Despite the appearance of the term as early as the seventies, it was popularized by Google in 2012, followed by other web giants such as Facebook, Linkedin, Amazon, Airbnb, Microsoft, Uber, etc.

> A knowledge graph is a data structure that represents a universe of knowledge like a set of entities linked to each other through semantic relations. This structure should enable a human as well as a machine to reason on its content and infer or derive new relations.

THE KNOWLEDGE GRAPH: THE IDEAL STRUCTURE FOR A DATA CATALOG

Such a structure **fits perfectly with our vision of a data catalog:** an set of physical objects (sources, data sets, columns, treatments, dashboards, etc.) linked semantically between themselves, as well as to other more abstract objects – applications, domains and business glossaries, activities, organizational units, business managers, etc.

Another quality to highlight with a knowledge graph: **its flexibility.** It doesn’t need to be fully designed from the start. It can on the contrary start small and be enriched incrementally as the usage of the data catalog grows. A knowledge graph is a very modulable structure, designed to grow and adapt progressively. Ideal for a data catalog.

- **It provides a rich context** for the catalog objects and improves their graph representations and those of their relations.
- **It can feed the recommendations system,** which will spontaneously suggest links towards certain objects during the exploration of the catalog.
- **It can feed the suggestions system,** which will help the data stewards feed and monitor the catalog content.
- **It greatly improves the performance of the search engine,** particularly for low intent searches (we will go into more detail).

THE CHALLENGE OF IMPLEMENTING A KNOWLEDGE GRAPH

The actual implementation of the Zeenea knowledge graph rests on some very simple principles:

- **The information is stored in a graph database in the most abstract manner possible:** each object is materialized by a simple entity with certain attributes (such as its type and all its properties). Those entities are linked to each other through bi-directional links whose attributes are the multiplicity and the semantic of the relation.

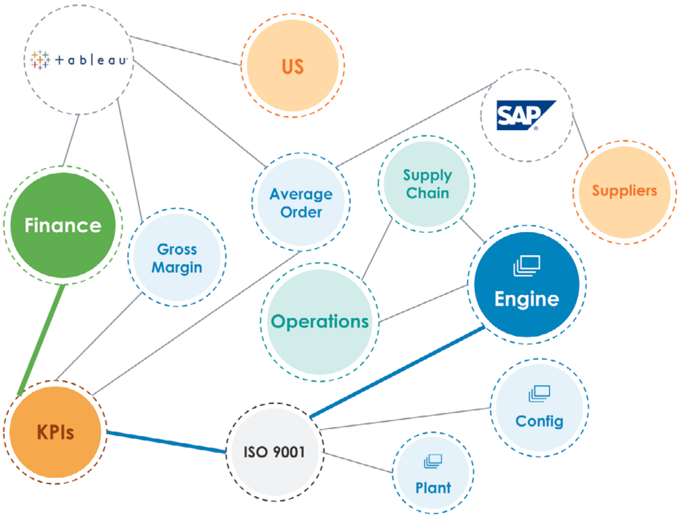
- **A management layer of the graph is in charge of the ontology** - meaning the graph integrity (what types of objects can be created, which links can be established between each other). For instance, at Zeenea, we consider a column in a table to be a type of object which must be linked to one (and only one) data set type object, with a strict notion of belonging. Such rules exist for most object types, whether they belong to a physical universe, a semantic universe or a business univers.

- **A request layer helps query the graph** to read the content in a less abstract form than the underlying generic model does, whilst conforming to the standard GraphQL.

- **An analysis layer runs a processing on the graph** in order to spot faulty links, suggest new ones, index its content and work out useful features for the search engine.

The knowledge graph is one of the most critical layers of our platform. It is at the heart of our technological know-how and our deep tech positioning. It is however a complex layer that is very abstract and not in sync with our self proclaimed focus on simplicity. The complexity of the knowledge graph remains to a great extent hidden from view in our interfaces:

KNOWLEDGE GRAPH



- The Zeenea applications (Studio and Explorer) never show the most abstract concepts and instead seek to offer highly ergonomic solutions to harness their strength.

- Public query APIs, based on GraphQL, help streamline how the entities and relations are represented in order to simplify the processing.

TECHNOLOGICAL BREAKTHROUGH #5

A MULTI-DIMENSIONAL SEARCH ENGINE

The last fundamental architectural layer of a data catalog is the search engine.

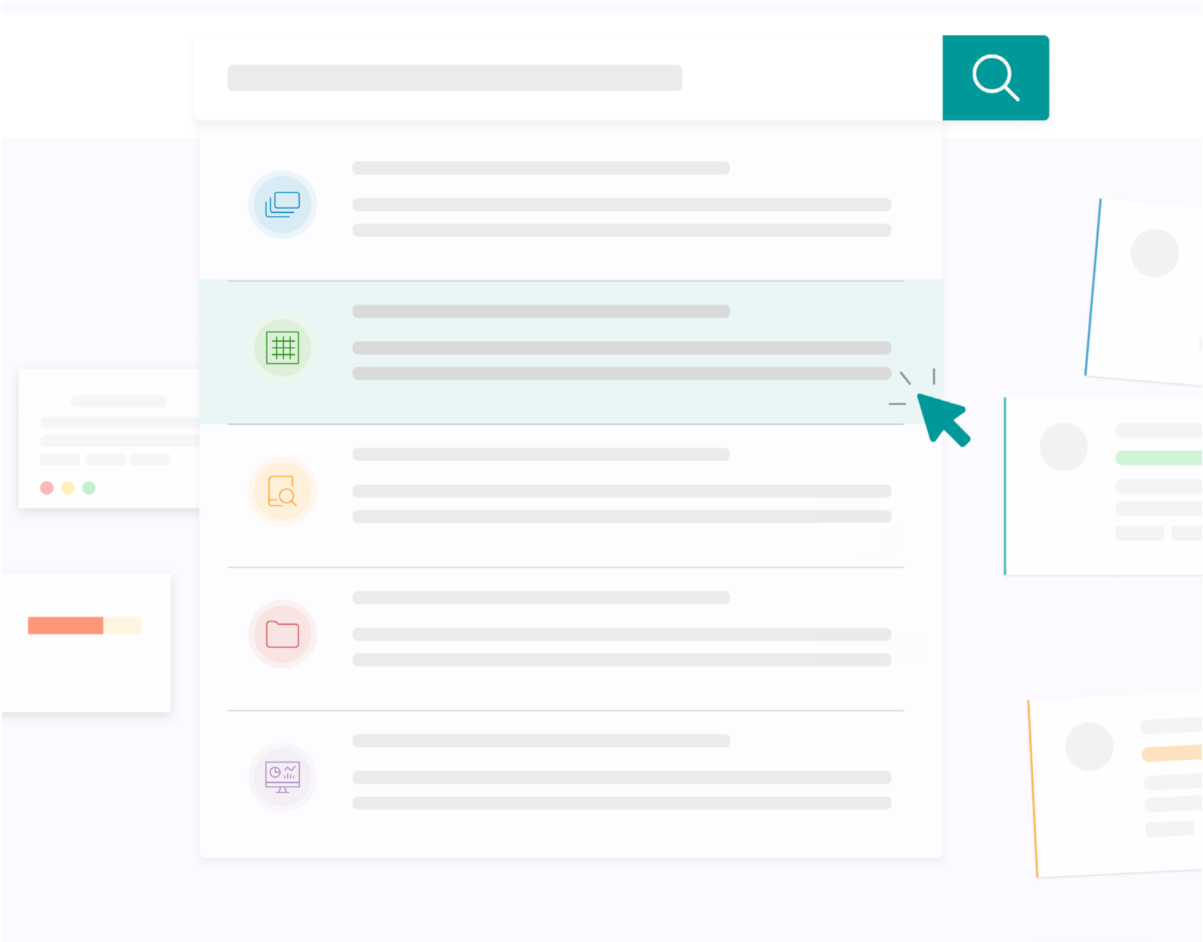
THE SEARCH ENGINE: A NECESSARY MECHANISM FOR EXPLORING THE DATA CATALOG

Given the enormous volumes of data involved in an enterprise catalog, **we consider the search engine the principal mechanism through which users can explore the catalog.** The search engine needs to be easy to use, powerful and, most importantly, efficient - the results must meet user expectations. Google and Amazon have raised the bar very high in this respect and the search experience they offer has become a reference in the field. This second to none search experience can be summed up thus:

- I write a few words in the search bar, often with the help of a suggestion system that offers frequent associations of terms to help me narrow down my search.

- The near instantaneous response provides results in a specific order and I fully expect to find the most relevant one on page one.

- Should this not be the case, I can simply add terms to narrow the search down even further, or use the available filters to cancel out the non relevant results.



Alas, the best currently on offer in the data cataloging market in terms of search capabilities seems to be limited to capable systems indexations, scoring and filtering. This approach is satisfactory when the user has a specific idea of what he is looking for (high intent search) but can prove disappointing when the search is more exploratory (low intent search) or when the idea is simply to spontaneously suggest relevant results to a user (no intent).

In short, simple indexation is great for finding information whose characteristics are well known, but falls short when the search is more exploratory. The results often include false positives and the order in which the search comes out is over-represented with exact matches.

A MULTIDIMENSIONAL APPROACH FOR EFFECTIVE RESEARCH

We decided from the get go that a simple indexation system would prove limited and would fall short of providing the most relevant results for the users. We therefore chose to isolate the search engine in a dedicated module on the platform and to turn into a powerful innovation (and investment) zone.

Our goal is to move away from search engines and data indexation on flat information processing. We naturally took an interest in the work of the founders of Google on Page Rank, their algorithm. Page Rank takes into account several dozen aspects (called features), amongst which are the density of the relation between different graph objects (hypertext links in the case of internet pages), the linguistic treatment of search terms or the semantic analysis of the knowledge graph.

Of course, we do not have the means Google has, nor its expertise in terms of search result optimisation. But we have integrated in our search engine several features that provide a high level of relevant results, and those features are permanently evolving - testing and green lighting the performance of a search engine is already a considerable achievement.

We have integrated the following core features:

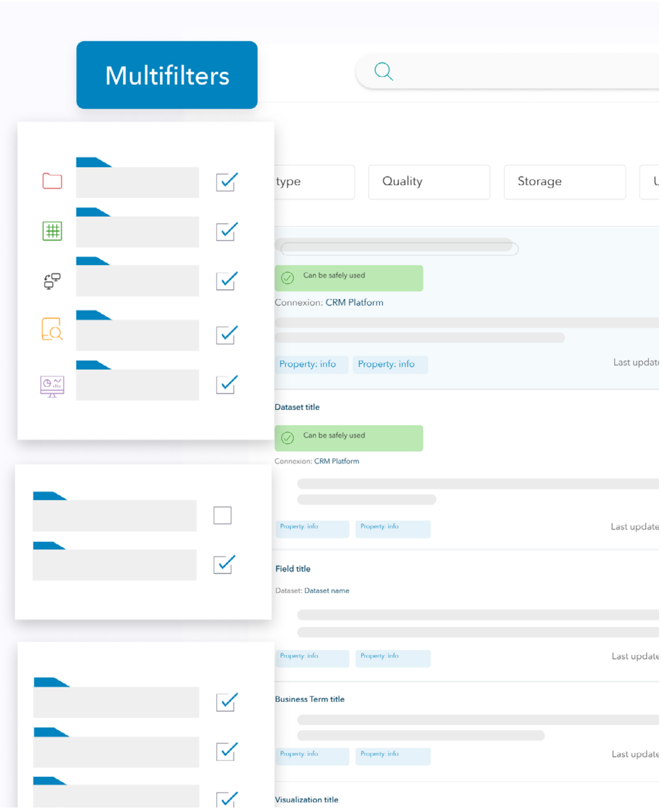
- **Standard, flat, indexation** of all the attributes of an object (name, description and properties) weighing it up in accordance with the type of property.
- **An NLP layer (Natural Language Processing)** that takes into account the near misses (typing or spelling errors).
- **A semantic analysis layer** that relies on the processing of the knowledge graph.
- **A personalisation layer** that currently relies on a simple user classification according to their uses, and will in the future be enriched by individual profiling.

SMART FILTERING TO CONTEXTUALIZE AND LIMIT SEARCH RESULTS

To complete the search engine, we also provide what we call a **smart filtering system.** Smart filtering is something we often find on e-commerce websites (Amazon, booking.com, etc.) and it consists in providing contextual filters to limit the search result.

These filters work in the following way:

- Only those properties that help reduce the list of results are offered in the list of filters - non discriminating properties do not show up.
- Each filter shows its impact - meaning the number of residual results once the filter has been applied.
- Applying a filter refreshes the list of results instantaneously.



With this combination of multi-dimensional search and smart filtering, we feel that we offer a superior search experience to any of our competitors. And our decoupled architecture (the search engine is an autonomous component) enables us to explore new approaches continuously, and rapidly integrate those that seem efficient.

TAKE AWAY

Founded by innovation enthusiasts, Zeenea made the decision very early on to endow its data catalog solution with the following qualities: simplicity, flexibility, and deep tech. The architecture and design of our solution are the pillars of these qualities - and constitute technological breakthroughs that ensure the durability of this vision:

- The multi-tenant cloud architecture ensures ease of deployment and rapid adoption of the solution - at the cost of significant efforts to integrate security and privacy into all design choices (Security and Privacy by Design).
- Proprietary, universal, and open connectivity enables rapid integration of the data catalog into heterogeneous technical environments, and provides connectors with innovative capabilities to efficiently produce metadata.
- The platform architecture ensures that the user experience is optimized according to their use cases - each application can be specifically designed to cover the needs of certain user populations.
- A knowledge graph at its core - despite its conceptual complexity, this is the natural structure of a data catalog, allowing the most sophisticated processing (evolving meta-model, semantic analysis, machine learning, intelligent scoring, etc.).
- A multidimensional search engine, which not only indexes information but, based on the knowledge graph, enables the most modern approaches - recommendation, personalized ranking, fuzzy search, etc.

zeenea

Be Data Fluent

More information about our Data Catalog?

Contact us now for a free demo!

#BeDataFluent

Contact us

www.zeenea.com - info@zeenea.com