



# WHAT IS A **SMART** DATA CATALOG?

And why it isn't only about machine learning




The idea of a *Smart Data Catalog* has been around for a few years in metadata management related literature, although it has no official definition. The general consensus is that a modern data catalog must have machine learning and AI to unlock its potential.

In this piece, we will attempt to define how Zeenea handles the idea of the Smart Data Catalog which, for us, cannot be limited to machine learning capabilities.

# Introduction

Before delving into what is meant by “*smart*”, let’s first define what a data catalog actually is.

 A **data catalog** is a detailed inventory of all the available company information assets, along with the metadata needed to make use of these assets.

A data catalog is meant to help data professionals find data assets quickly in order for them to carry their projects out effectively. It therefore goes beyond a structured data inventory and should provide an operational system aimed at accelerating data initiatives. The data catalog should meet the needs of a variety of different users - analysts, engineers, risk and conformity professionals, data scientists, product and business managers, etc. In short, the data catalog should cater for the needs of end users above all.

## Data as a corporate asset

The notion of information asset should be developed further. The perception of data as company assets is fairly recent and has grown by leaps and bounds: in a massively digitalized world, **the more successful organizations are those that make the most of the vast volumes of data in their possession**. The data can be exchanged, sold, analyzed to improve product/service positioning, or used to enter new markets. It can also help towards innovation and improving operational performance. For an indepth look into the various means of data monetization, I recommend reading [Infonomics \(Douglas B. Laney, 2017\)](#).

- ⊕ **Data here fits perfectly with an accounting definition of an **asset**: it is a company resource which can be used to produce added value or help the company function properly (unlike other assets, data do not show up in the balance sheet).**

Data becomes an asset on one key condition: it has to be exploitable. The quest for efficient data exploitation has led to substantial investments in most organizations and touches on many aspects which we cannot cover here - company culture, technical architecture, organization, etc. One aspect, which is at the heart of a data catalog, is metadata management.

## **The importance of metadata in information exploitation**

It is important to bear in mind that **company data constitutes a binary magma (zeros and ones) stored in what is often a magnetic support, and is incomprehensible to most people.**

To demonstrate this point, let's look at the data you use the most: the data stored in your computer or your mobile device. On the physical side, there is nothing unusual: data represents a series of zeros and ones on a hard drive (or any other storage system). You never actually access this raw data. A series of software solutions give you information on the data that enable you (or yet another software) to harness them:

- ✎ A **controller** will keep the information relating to the physical location of the bits that constitute an atomic dataset (typically in a file).
- 📁 A **filing system** will organize in a logical manner the datasets, and manage crucial information to make sense of the files (directories, names, extensions) or manage their security (owners, permissions, date of creation or update, etc.).
- 🔍 A **dedicated file explorer** will then harness this information in order to allow users to consult and understand the content - exploring the directory hierarchies, previsualizing files, search, linking extensions to applications, etc.

All the information managed by these different components is called metadata - literally data on data, and is indispensable to make sense of the content of the hard drive. It should be noted also that human intervention is very rare - you only need a filename and a place to put them in, the other metadata is managed automatically.

**Metadata is what turns binary content into exploitable information. At organization level, metadata plays this same role and the purpose of the data catalog is to consolidate the metadata with all available datasets, and present them in the simplest and most straightforward way to expectant data consumers.**

But the scaling - from a file system to an information system - leads to massive challenges.

## The purpose of the Smart Data Catalog: efficient consolidation of vast amounts of information

Regardless of its size, an information system contains several dozen systems and applications that store data through a wide variety of sources (relational and non relational databases, distributed file systems, APIs, cloud solutions, etc.), according to specific protocols, formats and rules. Each system manages hundreds or thousands of datasets - usually tables or files - themselves made of dozens of fields (or columns). And each dataset and each field feeds into a metamodel (in other words, an ensemble of structured metadata) which makes data exploration possible.

An enterprise metamodel is more sophisticated than that of an individual systems file. It potentially covers a wide range of aspects:

- ⚙️ **Technical metadata**, obviously - how and with which tools to access data, which protocols and authorizations, formats, types, etc.
- 🔗 **Semantic metadata** - what operational information does the dataset contain, which business rules govern them, etc.
- 🏢 **Organizational metadata** - who owns the data, who produces and controls it, how is it classified, etc.
- 📄 **Usage metadata** - where and how are they used, what is their quality, how are they monitored, etc.
- ✅ **Compliance metadata** - which internal rules and regulations have to be adhered to in order to use the data.

Ultimately, a data catalog will have to harness enormous amounts of very diverse information - and its volume will grow exponentially, just as the volume of usable data will. This volume of information will raise 2 major problems:

**How to feed and maintain the volume of information without tripling (or more) the cost of metadata management?**

**How to find the most relevant datasets for any specific use case?**

And it is in response to the 2 questions above that a data catalog will have to be *Smart*.

For us, a Smart Data Catalog should have a much wider scope than the integration of AI algorithms, and should include a range of smart technological and conceptual features that provide answers to the 2 questions above.


**We have identified 5 areas**  
in which a data catalog can be Smart  
- most of which do not involve machine learning:

<b>Metamodeling</b>	<b>9</b>
<b>The data inventory</b>	<b>15</b>
<b>Metadata management</b>	<b>20</b>
<b>The search engine</b>	<b>25</b>
<b>User experience</b>	<b>31</b>
<b>Take away</b>	<b>38</b>



The background features a light blue surface with several 3D objects: a brown rectangular frame, a teal rectangular frame, a purple U-shaped frame, and a teal L-shaped frame. Dark grey 3D arrows point in various directions. On the left, there are vertical color bands in teal, orange, white, and blue.

# METAMODELING

 Smart Data Catalog

At an enterprise scale, the metadata required to harness in any meaningful way the informational assets can be considerable. And besides a narrow sub layer (technical metadata mostly), the metadata is specific to each organization, sometimes even amongst different populations within an organization. For example, a business analyst won't necessarily seek the same information as an engineer or a product manager might.

## **A universal and static metamodel cannot be *smart***

**Attempting to create a universal metamodel therefore does not seem very smart to us.** Indeed, such a metamodel would have to adapt to a plethora of different situations, and will inevitably fall victim to one of the 3 pitfalls below:

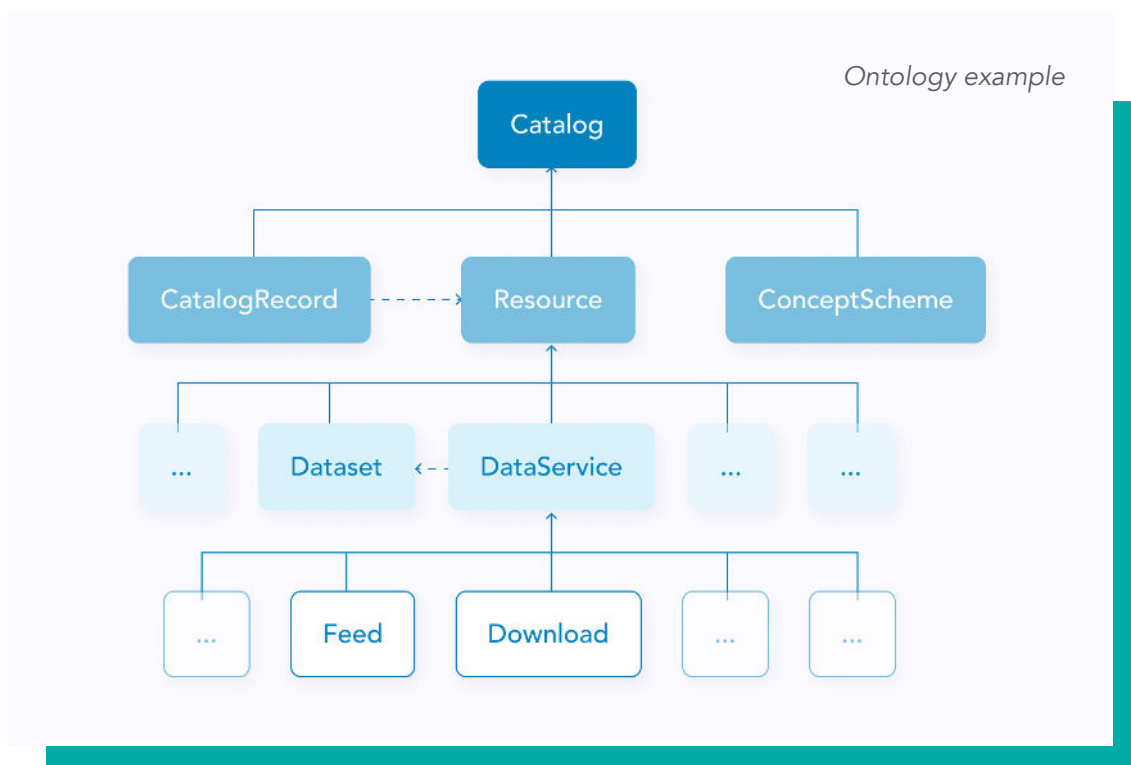
- » Excessive simplicity which won't cover all the use cases needed;
- » Excessive levels of abstraction with the potential to adapt to a number of contexts at the cost of arduous and time consuming training - not an ideal situation for an enterprise-wide catalog deployment;
- » Levels of abstraction lacking depth and ultimately leading to a multiplicity of concrete concepts bourn out of a combination of notions emanating from a variety of different contexts - many of which will be useless in any specific context, rendering the metamodel needlessly complicated and potentially incomprehensible.

**In our view, *smart* metamodeling should ensure a metamodel that adapts to any context and can be enriched as use cases or maturity levels develop over time.**

## The organic approach to a metamodel

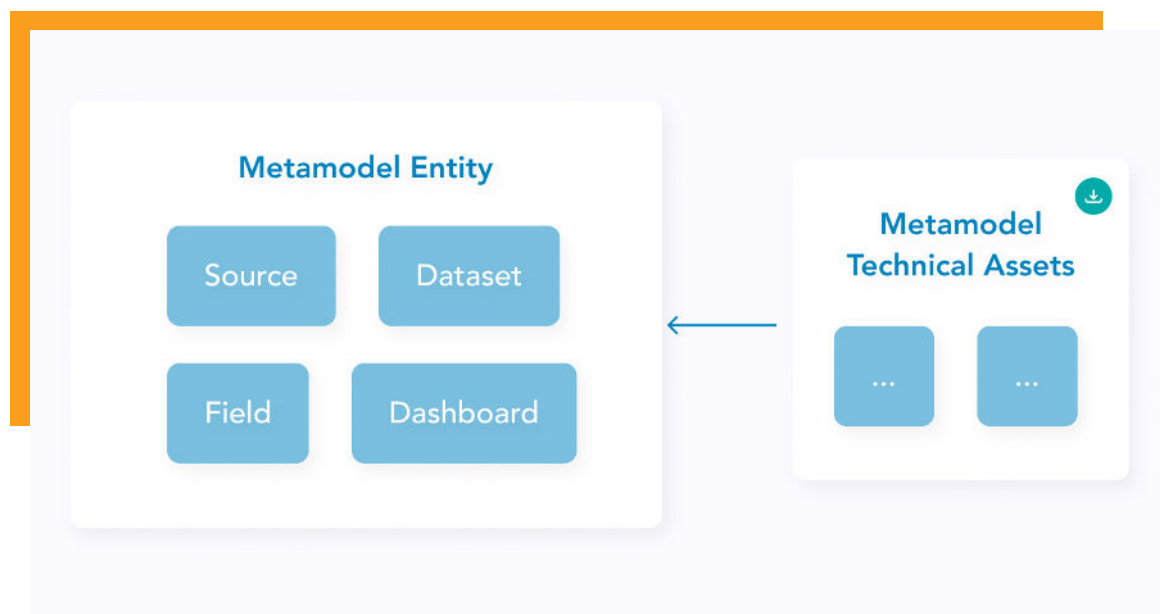
A metamodel is a field of knowledge, and the formal structure of a knowledge model is referred to as an ontology.

- 🔗 An **ontology** defines a range of object classes, their attributes, and the relationships between them. In a universal model, the ontology is static - the classes, the attributes and the relations are predefined, with varying levels of abstraction and complexity.



**Zeenea chose not to rely on a static ontology but rather on a scalable knowledge graph.**

The metamodel is therefore voluntarily simple at the start - there are only a handful of types, representing the different classes of information assets (data sources, datasets, fields, dashboards), each with a few essential attributes (name, description, contacts). This metamodel is fed automatically by the technical metadata extracted from the data sources which vary depending on the technology in question (the technical metadata of a table in a data warehouse differs from the technical metadata of a file in a data lake).



Moving forward, it is possible to define new object classes, or new attributes to existing classes, as well as the way in which these different objects relate to each other. The ontology isn't coded "in stone" and it is easy to integrate the evolutions of the metamodel in a manner that allows for trial and error. The progressive expansion of the data catalog to new use cases is made all the more straightforward.

**Our technology relies on a knowledge graph** - to find out more, I recommend you read our previous ebook, [the 5 technological breakthroughs of a Next-Gen Data Catalog](#).

For Zeenea, this organic metamodeling is the smartest way to handle the ontology issue in a data catalog. Indeed, it offers several advantages:

- » The metamodel can adapt to each context, often relying on a pre-existing model, integrating the inhouse nomenclature and terminology without the need for a long and costly learning curve;
- » The metamodel does not need to be fully defined before using the data catalog - you will only need to focus on a few classes of objects and the few necessary attributes to cover the initial use cases. You can then load the model as catalog adoption progresses over time;
- » User feedback can be integrated progressively, improving catalog adoption, and as a result ensuring return on investment for the metadata management.

## **Adding functional attributes to the metamodel in order to facilitate searching**

There are considerable advantages to this metamodeling approach, but also one major inconvenience: since the metamodel is completely dynamic, it is difficult for the engine to understand the structure, and therefore difficult for it to help users feed the catalog and use the data (two core components of a Smart Data Catalog).


We will expand on how we handle these issues a little further down. Part of the solution, however, relates to the metamodel and the ontology attributes. Usually, metamodel attributes are defined by their technical types (date, number, chain of characters, list of values, etc.). With Zeenea, these library types do include these technical types of course.

But they also include functional types - quality levels, confidentiality levels, personal touch, etc.

**These functional types enable the Zeenea engine to better understand the ontology, refine the algorithms and adapt the representation of the information - we'll come back to this.**



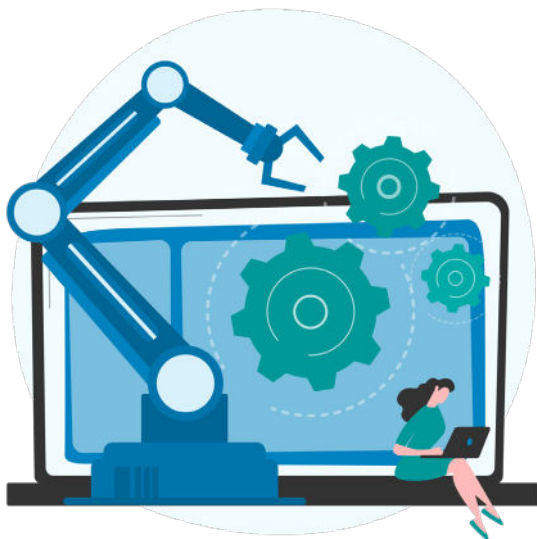
# THE DATA INVENTORY

 Smart Data Catalog

The second way to make a data catalog “smart” is through its inventory. A data catalog is essentially a thorough inventory of information assets that include a bunch of metadata, which help harness the information as efficiently as possible. **Setting up a data catalog therefore depends first of all on an inventory of the assets from the different systems.**

## Automating the inventory: the challenges

A declarative approach to building the inventory doesn’t strike us as particularly smart, however well thought out it may be. It involves a lot of work at the launching and the up-keeping of the catalog - in a fast changing digital landscape, the initial effort quickly becomes redundant.







**The First (obvious) step in creating a smart inventory, is of course to automate it.** With a few exceptions, enterprise datasets are managed by system specialists (involving distributed filing systems, ERPs, relational databases, software packages, data warehouses, etc.). They manage all these systems along with all the metadata required for them to work properly. There is no need to recreate this information manually: you just need to connect to the different registries and synchronize the catalog content with the source systems.



In theory, this should be straightforward but putting it into practice is actually rather difficult. **The fact is, there is no universal standard to which the different technologies conform to for a universal means of access to their metadata.** Some systems offer simple and well-documented protocols to access the metadata but others require more in-depth studies. Even with relational databases and other common systems, the standard methods are disappointing. The JDBC API for instance (which provides the connectivity to the relational databases for Java) provides standard interfaces to access the metadata. It should therefore be easy to gather this information for all systems that have a driver (basically all technologies around today). In reality however, this standard API only provides basic information. The system tables from these solutions are in possession of much richer and complete metadata, and it is often better to consult them directly.

## The essential role of connectivity to the system sources



**A smart connectivity layer is a key part of the Smart Data Catalog.** For a more detailed description of Zeenea's connectivity technology, I recommend reading our previous ebook, [5 technological breakthroughs of a next generation catalog](#), but its main characteristics are:

-  **Proprietary** - we do not rely on third parties so as to maintain a highly specialized extraction of the metadata.
-  **Distributed** - in order to maximize the reach of the catalog.
-  **Open** - anyone looking to enrich the catalog can develop their own connectors with ease.
-  **Universal** - it can synchronize any source of metadata.

This connectivity can not only read and synchronize the metadata contained in the source registries, it can also produce metadata.

This production of metadata requires more than a simple access to the source system registries. It also requires access to the data itself, which will be analyzed by our scanners in order to enrich the catalog automatically.




To date, **we produce 2 types of metadata:**

-  **Statistical analysis:** to build a profile of the data - value distribution, rate of null values, top values, etc. (the nature of the metadata depends obviously on the native type of the data being analyzed);
-  **Structural analysis:** to determine the operational type of specific textual data (email, postal address, social security number, client code, etc. - the system is scalable and customizable).

*In the next few months, we will be extending metadata production, by analyzing request journals or permission systems, when possible.*


## The inventory mechanism must also be smart

Our inventory mechanism is also smart in several ways (beyond boasting a connectivity which can automatically feed the catalog with the assets contained in the different systems):

-  **Dataset detection** relies on an extensive knowledge of the storage structures, particularly in a Big Data context. For example, an IoT dataset made up of thousands of files of time series measures can be identified as a unique dataset (the number of files and their location being only metadata).
-  **The inventory** is not integrated in the catalog by default to prevent the import of technical or temporary datasets that would be of little use (either because the data is unexploitable, or because it is duplicated data).
-  **The selection process for the assets** that should be imported in the catalog also benefits from some assistance - we strive to identify the most appropriate objects for integration in the catalog (with a variety of additional approaches to make this selection).



# METADATA MANAGEMENT

 Smart Data Catalog

It is in the field of metadata management that the notion of the Smart Data Catalog is most commonly associated with algorithms, machine learning, and AI.

## How is metadata management automated?

**Metadata management is the discipline that consists in valuing the metamodel attributes for the inventoried assets.** The workload required is usually proportional to the number of attributes in the metamodel and the number of assets in the catalog. And as we now know, the volume of metadata to feed and maintain can be enormous.

The role of the Smart Data Catalog is to automate this activity as much as possible, or at the very least to help the human operators (Data Stewards) do so in order to ensure greater productivity and reliability.

**A smart connectivity layer enables the automation of part of the metadata but this automation is very much restricted to a limited subset of the metamodel - mostly technical metadata.** A complete metamodel, even a modest one, also has dozens of metadata that cannot be extracted from the source systems registries (because they are not there to begin with).

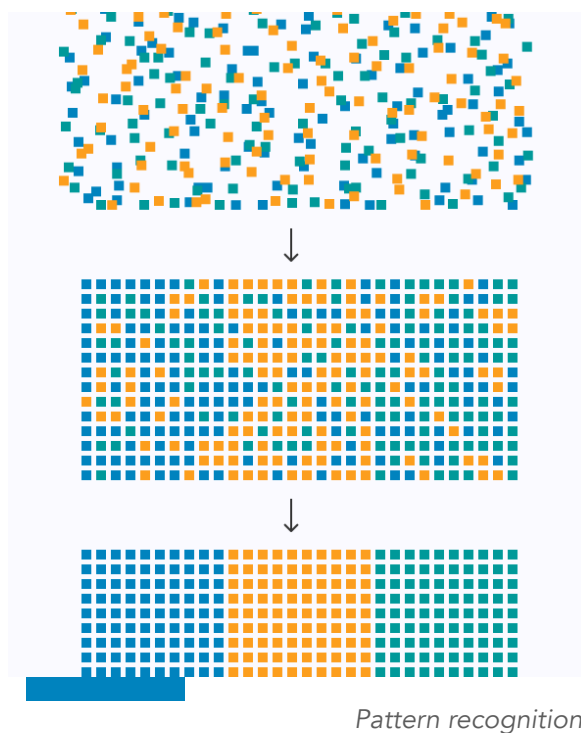
To solve this equation, several approaches are possible.

## Pattern recognition

The most direct approach consists in looking to identify patterns in the catalog in order to suggest metadata values for new assets.

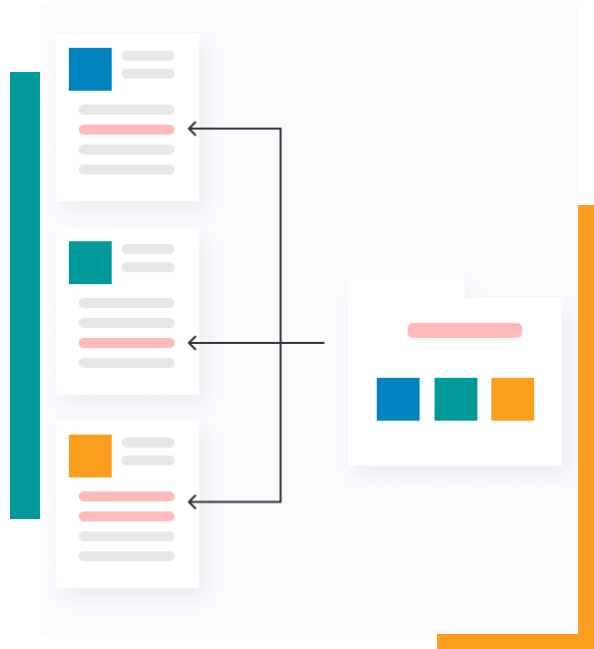
Put simply, a **pattern** will include all the metadata of an asset, and the metadata of its relations with other assets or other catalog entities. Pattern recognition is typically done with the help of **machine learning algorithms**.

The difficulty with the implementation of this approach is precisely qualifying the information assets in a numerical form in order to feed the algorithms and select the relevant patterns. A simple structural analysis is not enough: two datasets can contain identical data but in different structures. Relying on the identity of the data isn't efficient either: two datasets can contain identical information but with different values. For example, *2020 client invoicing* in one dataset, *2021 client invoicing* in the other.



In order to solve this problem, Zeenea relies on a technology called *Fingerprinting*.

**Fingerprinting** consists of shortening a dataset (more accurately a field in a dataset) to a numerical vector describing that data (otherwise known as a **feature**).



In order to build the fingerprint, we pull up 2 types of features from our clients' data:

- 🔗 A group of **Features** adapted to the numerical data (mostly statistical indicators)
  
- 🔗 Data emanating from **word embedding models** (word vectorisation) for the textual data.

**Fingerprinting is at the heart of our intelligent algorithms.**

## The other embedded approaches in a suggestion engine

While pattern recognition is indeed an efficient approach for suggesting the metadata of a new asset in a catalog, it rests on an important prerequisite: in order to recognize a pattern, there has to be one to recognize. In other words, this only works if there are a number of assets in the catalog (which is obviously not the case at the start of a project).

And it's precisely in these initial phases of a catalog project that the metadata management load is the highest. **It is therefore crucial to include other approaches likely to help the Data Stewards in these initial phases, when a catalog is more or less empty...**

The Zeenea suggestion engine, which provides intelligent algorithms to assist the management of the metadata, also provides other approaches (which we enrich regularly). Here are some of these approaches:

» **Structural similarity detection**

- which will work when several datasets have identical structures, which is commonplace in *layered* data lake architectures.

» **Fingerprint similarity detection**

- which does not rely on pattern recognition, but on a straightforward euclidean distance calculation between fingerprints (the fingerprints of two datasets with similar content are likewise very similar).

» **Name approximation**

- which consists in dynamically building a technical names dictionary associated with specific metadata, which works well for specific types of associations. This is the case with the semantic layer, which we could for example suggest associating a field named *txt\_email* with a glossary definition named *Email*.

This suggestion engine, which analyzes the catalog content in order to determine the probable values of the metadata from the assets that have been integrated, is an everlasting subject of experimentation. We regularly add new approaches, sometimes very simple and sometimes much more sophisticated. In our architecture, it is a dedicated service whose performances improve as the catalog grows and as we enrich our algorithms.


As you may have noticed, the development of the suggestion engine is presented as experimental - I identify a promising approach, I implement it, I measure its performance, and I start again. It's a standard approach but one which raises a major question: how to measure the performance of the data catalog's intelligent algorithms?

**Zeenea has chosen to use the lead time as our main measuring metric for the productivity of the Data Stewards** (which is the ultimate objective of smart metadata management). Lead time is a notion that stems from lean management and which measures, in a data catalog context, the time elapsed between the moment an asset is inventoried and the moment all its metadata has been valued. This approach will be detailed in a later publication.





# THE SEARCH ENGINE

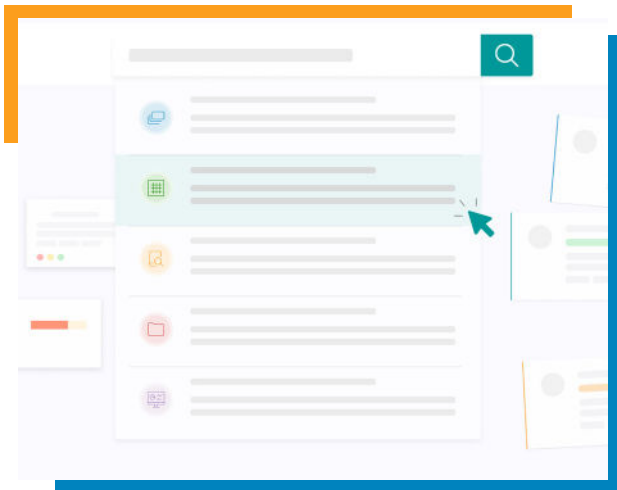
 Smart Data Catalog

As we saw in the introduction, **quickly finding the most relevant assets to service a given asset is the main objective of a data catalog project.** This chapter echoes the chapter dealing with search engines in our previous ebook, [5 technological breakthroughs of the next generation data catalog.](#)

## A powerful search engine for an efficient exploration

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 they are 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 search approach

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 it 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 optimization. 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.



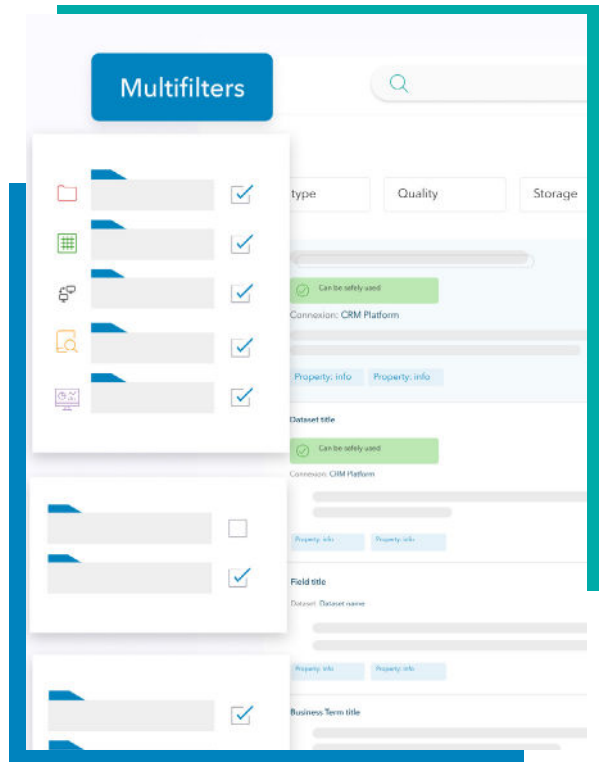
A personalization layer that currently relies on a simple user classification according to their uses, and will in the future be enriched by individual profiling.

## 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 (such as 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.




# USER EXPERIENCE

 Smart Data Catalog



A data catalog should also be smart in the experience it offers to its different pools of users. Indeed, **one of the main challenges with the deployment of a data catalog is its level of adoption from those it is meant for: data consumers.** And user experience plays a major role in this adoption.

## User experience within the data catalog

 **User experience** is a complex subject which we won't dive too much into here. The underlying purpose of user experience is the **identification of personas whose behavior and objectives we are looking to model in order to provide them with a slick and efficient graphic interface.**

Pinning down personas in a data catalog is challenging - it is a universal tool that provides added value for any company regardless of its size, across all sectors of activity anywhere in the world.

Rather than attempting to model personas that are hard to define, it's possible to handle the situation by focusing on the issue of data cataloging adoption. Here, there are two user populations that stand out:

-  Metadata producers who feed the catalog and monitor the quality of its content - this population is generally referred to as Data Stewards;
-  Metadata consumers who use the catalog to meet their business needs - well we will call them Users.

These two groups are not totally unrelated to each other of course: some Data Stewards will also be Users.



## The challenges of enterprise-wide catalog adoption

Data Stewards are not really the issue here since they use the solution regularly. They can live with a longer learning curve so long as the solution is user friendly and helps them in their daily activities. This brings us back to the previous chapters in this paper: the capacity of the data catalog to produce metadata.

**The real value of a data catalog resides in large-scale adoption by a substantial pool of (meta) data consumers, not just the data management specialists.**

This pool is very diverse. It includes data experts (engineers, architects, data analysts, data scientists, etc.), business people (project managers, business unit managers, product managers, etc.), compliance and risk managers. And more generally, all operational managers likely to leverage data to improve their performances.

**Data Catalog adoption by Users is often slowed down for the following reasons:**

- » **Data catalog usage is sporadic:** they will log on from time to time to obtain very specific answers to specific queries. They rarely have the time or patience to go through a learning curve on a tool they will only use periodically - weeks can go by between catalog usage.
- » **Not everyone has the same stance on metadata.** Some will focus more on technical metadata, others will focus heavily on the semantic challenges, and others might be more interested in the organizational and governance aspects.
- » **Not everybody will understand the metamodel or the internal organization of the information within the catalog.** They can quickly feel put off by an avalanche of concepts that feel irrelevant to their day to day needs.

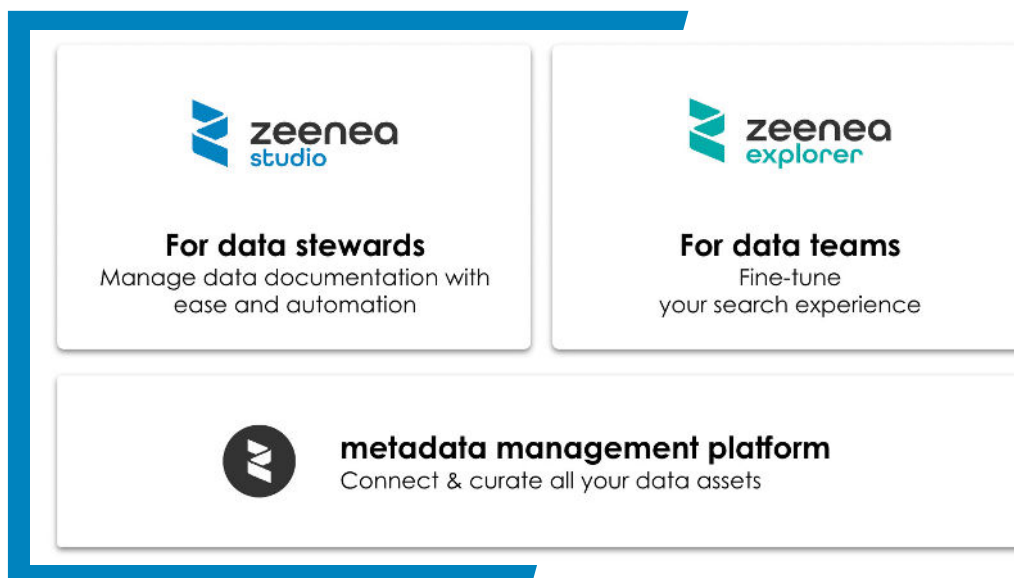
The Smart Data Catalog attempts to jump these hurdles in order to accelerate catalog adoption. Here is how Zeenea meets these challenges.

## How Zeenea Facilitates catalog adoption

### A familiar graphic interface, inspired from e-commerce

The first solution is the **graphic interface**. The Users' learning curve needs to be as short as possible. Indeed, the User should be up and running without the need for any training. To make this possible, we made a number of choices.

The first choice was to provide **two different interfaces, one for the Data Stewards and one for the Users:**



**Zeenea Studio** - the management and monitoring tool for the catalog content - an expert tool solely for the Data Stewards.

**Zeenea Explorer** - for the Users - it provides them with the simplest search and exploration experience possible.

Our approach is aligned with **the user-friendly principles of marketplace solutions** - the recognized specialists in catalog management (in the general sense). These solutions usually have two applications on offer. The first, a “back office” solution, which enables the staff of the marketplace (or its partners) to feed the catalog in the most automated manner possible and control its content to ensure its quality. The second application, for the consumers, usually takes the form of an e-commerce website and enables end users to find articles or explore the catalog. Zeenea Studio and Zeena Explorer reflect these two roles.

Zeenea Explorer is therefore the key to large scale catalog adoption, and we decided to emulate e-commerce websites for its design:

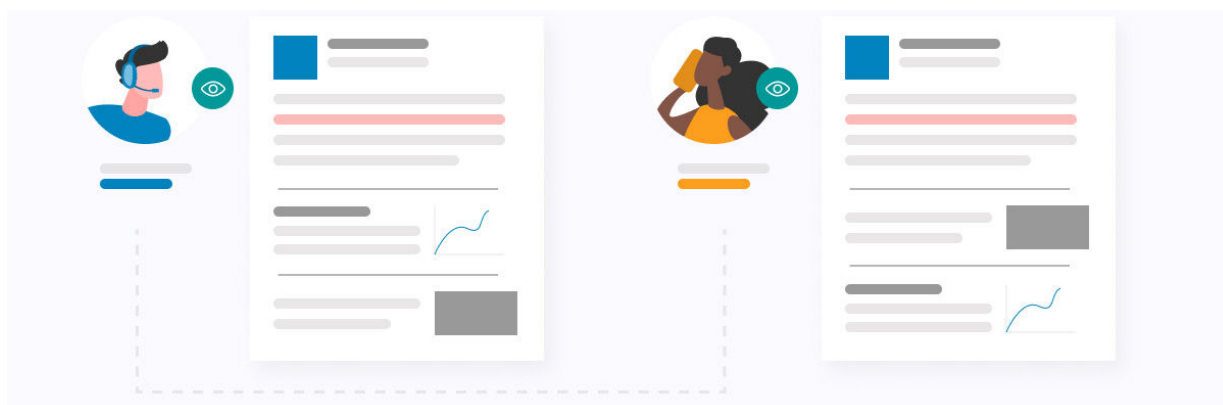
- » **The user-friendly design follows in the footsteps of these large marketplace sites:** efficient search engine, great exploration capabilities and a recommendation system that can push a specific set of objects adapted to each user profile.
- » **As with marketplaces, searching is the main means to access the information:** the search engine interface is heavily inspired by mainstream and e-commerce search engines.

The learning curve is considerably shortened as a result of giving Users a familiar looking system. An interactive guide feature for new Users makes this learning curve even simpler - there is no need for any training.

## The information is ranked in accordance with the role of the user within the organization

Our second choice is still at the experimental stage, and consists in **dynamically adapting the information hierarchy in the catalog according to User profiles.**

This information hierarchy challenge is what differentiates a data catalog from a marketplace type catalog. Information hierarchy in a marketplace is the same for all users - photos, description, price, and delivery information will always be the most important information followed by technical details, opinions, delivery conditions, etc.



**For a data catalog on the other hand, information hierarchy depends on the operational role of the user.** For some, the most relevant information in a dataset will be technical: location, security, formats, types, etc. Others will need to know the data semantics and their business lineage. Others still will want to know the processes and controls that drive data production - for compliance or operational considerations.

The Smart Data Catalog should be able to dynamically adjust the structure of the information to adapt to its different prisms - it is one of the areas for development that we are currently working on for Zeenea Explorer.

The last remaining challenge is the manner in which the information is organized in the catalog in the form of **exploration paths by theme** (something similar to shelving in a marketplace). It is difficult to find a structure that agrees with everybody. Some will explore the catalog along technical lines (systems, applications, technologies, etc.).

Others will explore the catalog from a more functional perspective (business domains), others still from a semantic angle (through business glossaries, etc.).

The challenge of having everyone agree on a sole universal classification seems (to us) insurmountable. The Smart Data Catalog should be adaptable and should not ask Users to understand a classification that makes no sense to them. Ultimately, user experience is one of the most important success factors for a data catalog. This user experience depends on the application designs offered by the catalog, and also by the efficiency and simplicity of the metamodel - which is a subject of its own.

# Take away

One of the cornerstones of a successful data strategy is effective metadata management across the organization. A Smart Data Catalog is the most effective tool for implementing metadata management.

The *smart* side of the Data Catalog is not reduced to the integration of a few more or less intelligent algorithms, but must be displayed in all aspects of the solution:

- » The way the metamodel can be designed and added to/ altered/improved as catalog adoption grows;
- » The most advanced automation of the inventory of data assets and the collection of metadata in the systems that host them;
- » The ability to assist the Data Stewards in the activities of feeding and controlling the content of the catalog;
- » The search engine, which is the simplest and most direct access point for data consumers;
- » The user experience, which must take into account the wide variety of profiles that will use the catalog.



## Want more information on our Data Catalog?

Contact us now to get a  
free personalized demo!

**#BeDataFluent**

**Contact us**

[www.zeenea.com](http://www.zeenea.com) - [info@zeenea.com](mailto:info@zeenea.com)