

Busca360: A Search Application in the Context of Top-Side Asset Integrity Management in the Oil & Gas Industry

Yenier T. Izquierdo¹, Melissa Lemos^{1,3}, Cleber Oliveira¹, Bruno Novelli¹,
Grettel M. García¹, Gustavo Coelho¹, Lucas Feijó¹, Bruno Coutinho¹, Tiago Santana¹,
Robinson Luiz Souza Garcia²,
Marco Antonio Casanova³

¹ Tecgraf Institute, PUC-Rio – Rio de Janeiro – RJ – Brazil

²Petrobras – Rio de Janeiro – RJ – Brazil

³Department of Informatics, PUC-Rio – Rio de Janeiro – RJ – Brazil

{ytorres,melissa,cleberoli,bnovelli,ggarcia}@tecgraf,puc-rio.br
{gustavocoelho,lucasfeijo,brunocoutinho,tiagotprs}@tecgraf.puc-rio.br
robinson.garcia@petrobras.com.br
casanova@inf.puc-rio.br

Abstract. *Oil and gas industry applications often require querying data of various types and integrating the query results. Data range from structured tables stored in databases to documents and images organized in digital libraries. The users typically have technical training but are not necessarily versed in Information Technology, meaning the data processing tasks may burden them significantly. This article introduces a multimodal search application, called Busca360, designed to alleviate this burden and discusses the main challenges that emerged during the research, implementation, and user experience. The application uses structured data in the context of asset integrity management and 360° images of equipment and installation locations. Finally, this article concludes with real-world use cases that show how the proposed multimodal search application helps perform planning and maintenance tasks.*

1. Introduction

The rapid advancement of the digital transformation process in the oil and gas industry constantly produces a significant volume of structured and unstructured data related to industrial asset management in various systems. This data ranges from structured tables stored in relational databases to documents and images organized in digital libraries, and sensor data persisted in specialized data stores.

End users typically have the technical training to understand the information coming from the data. Nonetheless, they do not always have a background in Information Technology, meaning accessing, searching, navigating, and integrating this data is challenging. Therefore, these professionals have a few options. They can rely on developing applications that provide search forms for the databases, which must be consistently updated to accommodate changes in the database structures. Alternatively, they may learn programming and query languages to create scripts for extracting and combining data

from these databases. Another option is to depend on data engineers or data scientists to create scripts and export the data into spreadsheets so they can analyze it.

This scenario led to the creation of Busca360, which aims to provide user-friendly semantic intelligent search and exploration of data. It focuses on integrated access to databases and document archives in industrial asset integrity management, specifically for stationary oil and gas production units, emphasizing compliance, integrity, and operational condition monitoring.

Busca360 streamlines access to accurate information, reducing time, uncertainties, and risks in decision-making for projects conducted by professionals in stationary oil and gas production units. The system deals with structured and unstructured data and aims to include advanced image and text interpretation capabilities, enabling navigation through a multimodal approach.

Currently, the application is deployed and running within the oil and gas industry, serving users across various oil platforms. It is part of a product ecosystem that integrates a Digital Twin. Given the Big Data scenario, dealing with a vast volume of data in different formats that require curation and enrichment, challenges related to infrastructure (hardware level) and algorithm optimization (software level) are addressed. This article will detail the main challenges, solutions, and lessons learned during the research and development of this product, as well as discuss the necessary next steps.

The remainder of this article is organized as follows. Section 2 summarizes the related work. Section 3 presents the main components of Busca360 and some development and deployment details. Section 4 spotlights relevant business challenges faced from the user's experiences of Busca360 and some approaches adopted to address them. Section 5 shows some real use cases of how the application helps business professionals perform searches that assist them with integrity and planning tasks. Finally, Section 6 presents the conclusions and directions for future work.

2. Related Work

This section briefly covers related work in areas directly connected to the main technical thrust of this article, namely, data integration, database keyword search engines, and multimodal search engines.

Data Integration. Data integration is an old and pervasive problem. Classic data integration is usually divided into the major sub-problems of data retrieval, data fusion, schema alignment, and entity linkage [Doan et al. 2012].

In the context of industrial data, [Nguyen et al. 2013] described the development of a framework for data integration to optimize the remote operations of offshore wind farms. [Espíndola et al. 2013] presented an approach that integrates data from mixed/augmented reality tools and embedded intelligent maintenance systems to support operators/technicians during maintenance tasks, providing easier access, understanding, and comprehension of information from different systems.

The application introduced in this article addresses a problem close to those discussed in [Nguyen et al. 2013, Espíndola et al. 2013] but includes image data, as in [Boehm et al. 2022].

Database Keyword Search Engines. Early relational keyword-based query processing tools [Bergamaschi et al. 2016, de Oliveira et al. 2015, Ramada et al. 2020] explored the foreign/primary keys declared in the relational schema to compile a keyword-based query into an SQL query with a minimal set of join clauses – and this is a key idea – based on the notion of candidate networks.

As an evolution of these earlier tools, [García et al. 2017] and the tool named QUIOW [Izquierdo et al. 2018] proposed a fully automatic, schema-based tool that supports keyword-based query processing for relational and RDF environments. The tool first constructs a Steiner tree that covers a set of nodes (relation schemes or RDF classes) whose instances match the largest set of keywords. It then compiles the keyword-based query into an SQL (or SPARQL) query that includes restriction clauses representing keyword matches and join clauses connecting the restriction clauses. Without such join clauses, an answer would be a disconnected set of tuples (or nodes of the RDF graph), which hardly makes sense. The generation of the join clauses builds upon the idea of candidate networks. This algorithm was enhanced and reported in [Garcia 2020].

The application introduced in this article is powered by DANKE [Izquierdo et al. 2021], a data and knowledge retrieval platform based on knowledge graphs.

Multimodal Search Engines. Many recent academic papers have explored the concept of a multimodal search engine. The main focus has been on queries based on a combination of image and text, where the expected results are also related to a combination of images and text, with some works exploring the use of videos and audio.

yu2022commercemm introduced a multimodal model for understanding commerce topics, where the content is composed of image, text, or image plus text. The model architecture includes an image encoder (ViT-B/16), a text encoder (XLM-R), and a multimodal fusion encoder, where the two encoders are combined, building contextualized multimodal embeddings. The implemented image co-processor in this multimodal search application encodes image fragments into high-dimensional vectors using the *Vision Transformer* (ViT) model [Dosovitskiy et al. 2021].

A common aspect between the aforementioned search engines is that queries and results are associated with the same set of modalities (e.g., queries composed of images and text return another combination of the same modalities). In a different approach, “Google Multisearch”¹ addresses a wide variety of scenarios in image and text multimodal search. In addition to returning images similar to a query image and a text description, the search engine allows alternative results based on the user’s inputs. The same logic can be used to locate other items, such as a dish in a local restaurant, shoes, home goods, etc.

Although all the listed references relate to some extent to the present work, the idea of returning different entities addressed by Google Multisearch is especially important. The return of nearby restaurants from a picture of a dish is somehow related to the idea of returning technical objects from a given image and text description. This concept extends the search engine used in the proposed application from an image-to-text (or text-to-image) search to a more holistic architecture, adding value to the user’s experience.

¹<https://blog.google/products/search/multisearch/>

3. Busca360: Components, Development, and Deployment

3.1. Main Components

Busca360 mainly comprises the front-end and the back-end components powered by DANKE, a data and knowledge retrieval platform based on knowledge graphs. The front-end component has a user interface (UI) that offers a user-friendly experience for searching, navigating, and exploring the data. It communicates with the back-end by requesting services defined in a RESTful API. The back-end is compounded by three modules: *Storage Module*, *Preparation Module*, and *Knowledge Extraction Module*.

The *Storage Module* houses a relational database, constructed from various data sources (from different systems at the company), conceptually modeled as a knowledge graph, created in this context following international standards. Domain experts with in-depth knowledge of various data sources lead the curation and enrichment process, improving the repository beyond its original data. This knowledge graph also facilitates other applications in interpreting and utilizing the data from the repository. Besides the data, this module also holds data indices required to support keyword searches.

Figure 1 illustrates a fragment of the knowledge graph (KG) that represents the industrial asset integrity management domain, specifically for stationary oil and gas production units, emphasizing compliance, integrity, and operational condition monitoring. It depicts the entity classes as rectangles and their relationships as single arrows (representing the foreign key/primary key relationships).

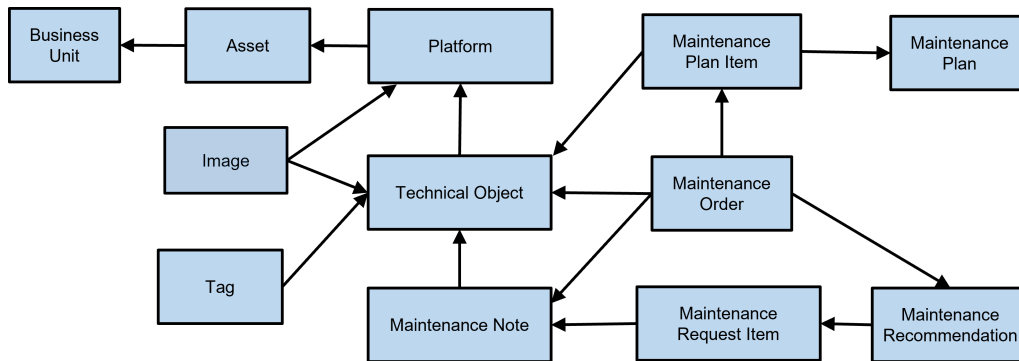


Figure 1. A fragment of the Busca360 knowledge graph.

A platform is an entity that represents a large structure installed offshore or on-shore to drill wells, extract oil and natural gas, and process hydrocarbons. It is part of an asset in a business unit. A technical object is an equipment or a physical object that requires maintenance and inspection. It has its characteristics, including an identifier tag, and is associated with a set of engineering documents. Maintenance and inspection specialists are involved in planning tasks or monitoring the progress of ongoing maintenance demands. For this purpose, they are involved with maintenance plans, notes, requests, orders, and recommendations. More details about the KG of Busca360 are described in [Molina et al. 2024].

The *Preparation Module* provides tools for creating the knowledge graph and for constructing and updating the centralized database through a pipeline responsible for a typical data integration process. This includes indexing data, collecting data from sources,

transforming and enriching data, and ingesting data into the database. Briefly, this module has three sub-modules:

- (a) *Data Setting* aims at setting up and maintaining the database. The main tasks are defining a *KG*, creating the mappings between the *KG* and the relational, structured data, and the media datasets stored in the database, and indicating which indexes should be built. These tasks are typically the responsibility of an expert user.
- (b) *Data Ingestion* populates the database. For non-image data, it aims at populating the data in the database from the data sources, considering structured, semi-structured, and unstructured (including multimedia) data. For image data, given an image dataset or a set of images stored as attribute values, D , this module indexes D , following the offline ingestion pipeline depicted in Figure 2.
- (c) *Data Enrichment* is responsible for making the original data more useful and insightful to extract information and knowledge from them. This process enriches data through transformation, categorization, standardization, aggregation, and annotation tasks to facilitate users' search and analysis.

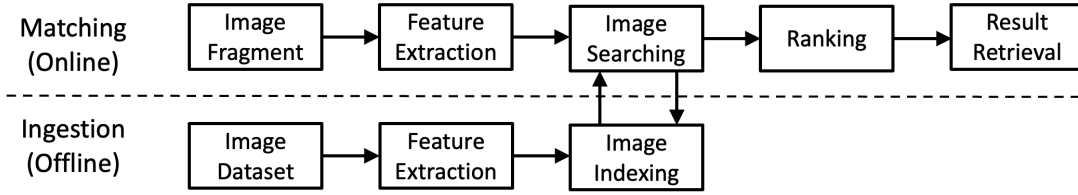


Figure 2. Image ingestion and matching pipeline (adapted from [Li et al. 2021]).

The *Knowledge Extraction Module* uses the technology described in [Izquierdo et al. 2021] and explores the knowledge graph and the data indices to compile a keyword query into an SQL (or SPARQL) query that returns data that best match the keywords. It features an algorithm that accepts as input a keyword query Q_K , using a *KG* and the indices, that follows the next steps: (1) finds matches with the keywords in Q_K ; (2) creates a conceptual query Q_C by exploring the keyword matches found and *KG*; (3) compiles Q_C into an SQL query Q_S , which is then executed.

If the query involves a search by images, step (3) of the above algorithm is extended as follows. Given an image fragment F occurring in the multimodal query, the algorithm first encodes F as a vector v_F . Then, it uses the image indexes to find a ranked list of image fragments (image identifiers) whose encoding vectors are approximate nearest neighbors of v_F , following the matching online pipeline illustrated in Figure 2. Finally, this list is included in the `WHERE` clause of Q_S , and a join clause is also added by connecting the image table to the rest of the tables in Q_S .

To process a multimodal query, the image-matching approach goes as follows. Given an image fragment F occurring in the multimodal query, DANKE first encodes F as a vector v_F and then uses the image indexes to find a ranked list of image fragments whose encoding vectors are approximate nearest neighbors of v_F . Finally, the module re-ranks the remaining candidate results to prioritize the images that contain higher scores related to the multimodal query.

3.2. Development and Deployment

The database management system adopted is Oracle 19c². The back-end component is mostly implemented using Java³ (using the spaCy library⁴ to interpret natural language queries), and the front-end component in Angular⁵.

The development process follows the agile Scrum methodology and includes participation, monitoring, and alignment with company professionals, applying the SAFe (Scaled Agile Framework) methodology.

Deploying the application to the company's infrastructure follows a standard CI/CD pipeline using the Jenkins⁶ configured for the build. This pipeline aims to build the code deposited in the GitLab repository of the company, resulting in a corresponding image docker⁷. Then, this image docker is deposited into the Harbor registry. Here, using the deploy Jenkins, a target image docker (with this version) is deployed in the Kubernetes environment⁸, allowing the application to be available for execution.

The production repository is monitored to maintain good performance by fine-tuning slow queries and improving execution plans and indexes. The repository is recreated daily because the data sources do not indicate which data has been updated, preventing incremental updates. During the process, an auxiliary repository is created and, after validation, replaces the old one to ensure consistency and stability. To improve performance, high-volume data sources that users do not need daily updates are excluded from this process and inherited from the old repository.

Currently, Busca360 operates in the company's production environment. Access control is managed using the internal authentication and access authorization platform, allowing users to log in only with the right to access all data available in the repository.

Also, training sessions are planned and held with analysts from different needs and business areas to promote the system and engage them in its use. After each training, a questionnaire is administered to collect the user's opinions regarding the use of the system. At the same time, feedback about the user experience is gathered, identifying proposed improvements, that become part of the development backlog.

4. Business Challenges and Adopted Solutions for Busca360

Busca360 was initially created on top of a search engine that translates keywords into SQL queries. Therefore, users must type all the necessary keywords into a text box (using a "Google-like" interface) so the system can correctly interpret it and generate the appropriate SQL.

Maintenance and inspection specialists are involved in complex tasks, creating and monitoring ongoing maintenance demands' progress which involves maintenance plans,

²<https://docs.oracle.com/en/database/oracle/oracle-database/19/index.html>

³<https://www.java.com/>

⁴<https://spacy.io/>

⁵<https://angular.io/>

⁶<https://www.jenkins.io/>

⁷<https://www.docker.com/>

⁸<https://kubernetes.io/>

maintenance notes, maintenance orders, and technical inspection reports. To perform such tasks, the specialists must access the details of each demand, such as description, due date, criticality, priority, status (whether they are pending), or location (specific portfolio).

In this scenario, users are required to construct queries by typing multiple keywords, such as “*maintenance notes technical objects platform A-1*”. In some cases, by adding reserved words to include filters on dates (such as “*creation date between 01/01/2022 and 12/31/2023*”), filters on numbers (such as “*failure impact greater than 3*”), or to apply aggregations (for total and average calculations, for example).

This section reveals the challenges that emerged during the research, implementation, and user experience of a keyword search engine on top of a database, described by a complex knowledge graph, in this real-world scenario. The following text presents the main challenges (with prefix **C**) and proposed solutions (with prefix **S**) to address them.

Concerning the “search moment”, when the user needs to inform its intention to query data:

C-01: Users may not be aware of the scope of the data in the centralized repository, i.e., which data sources or systems were utilized. Consequently, they may become frustrated when searching for data that is not included.

S-01: A home page was designed for Busca360 with more than just a search bar, unlike a “Google-like” visual.

C-02: Users may not be familiar with, and consequently may not be able to type, the keywords that represent the entities, attributes, and values defined in the systems or data sources comprising the centralized repository.

C-03: The users can not type everything at once that would be enough to define a complex SQL query (a lot of joins and filters), whether through a text with keywords or even in natural language. Additionally, sometimes, it is impractical to type all necessary keywords that define the desired entities, properties, and value filters.

S-02/03: Two features were developed for Busca360: (1) a *search-by-selection* option, where the entities and properties are shown grouped by their respective data sources, and the users can select by clicking what entities and properties they want in the result, and also to specify filters; (2) the definition, for each entity, a set of properties as default, where the user can activate this option in the UI if they desire to get these properties in the result, even if they have not been explicitly cited in the search (this applies to *search-by-text* and *search-by-selection* options).

C-04: Users would like to query data not stored in the sources. Sometimes, the information they need is obtained through processing, combining, and enriching data from the data sources.

S-04: This challenge is tackled by enriching the raw data from the sources (using the *Enrichment Module* described in Section 3.1). For example, the “*Pending*” property was created for entities: maintenance notes, requests, and orders based on the combination of other properties present in these entities. Hence, users can perform the query: *maintenance notes pending technical object platform A-1*.

C-05: Metadata and data can present a high degree of ambiguity, and users may not provide contexts that resolve some ambiguities in the terms they use in their query (such as date, center, and type).

S-05: A feature that shows a query interpretation was implemented. This allows users

to check whether the interpretation made by the *Knowledge Extraction Module* is as expected before returning the result.

C-06: Some users have a well-defined focus in their work and do not need to be familiar with all the entities, attributes, and data within the data repository, but only those relevant to their part of the business. This restricts the searches to be performed, both at the level of the graph to be queried and at the level of the data.

S-06: The implementation of business facet could address this challenge. The implementation of this feature is in the application development roadmap.

C-07: User employs different terms to retrieve the same data; in other words, there are synonyms. On the other hand, there are divergences; identical terms can mean different things to different users.

S-07: To deal with this, a standard vocabulary was defined into the *KG*, including synonyms for the metadata.

In the “post-search moment”, when the users get results, other challenges arise:

C-08: Users need all records from the result, not just the first ones. Therefore, there is no ranking in the output, so having an overview of the result could be tricky.

C-09: Users require interpreting the result through different methods: organized within a table, spatially visualized on maps and 3D models, or grouped in graphs.

S-08/09: To deal with these issues, a Result Analysis feature was developed that allows users to see the results as a data overview through statistics of the values and charts about the value results of individual attribute values and combining two attributes. Also, the application was equipped with features that allow integration with internal 3D visualization tools, such as ENVIRON [Raposo et al. 2009].

C-10: Users might think the query result is inaccurate. However, it is essential to consider the possibility that the data sources may contain errors and inconsistencies, and the centralized data repository may be outdated.

S-10: A feature was included to inform users whether the available data is updated.

In addition to all the above, a critical challenge regarding the “database performance” was also faced:

C-11: Predicting all the SQL queries generated by users using the application is not feasible. Users can type any keywords, potentially overloading the database server. It is possible to create indexes on data and improve search performance. However, traditional database tuning, guided by the creation of indexes to improve the execution time of pre-established queries, is not a suitable solution.

S-11: In this case, as Busca360 is deployed into an industrial infrastructure, the Database Administrators (DBAs) were ordered to analyze the suggestions for possible improvements indicated by the Automatic Database Diagnostic Monitor⁹ and the SQL Tuning Advisor¹⁰, and execute scripts to conduct the database tuning.

Unlike the academic scenario, the business environment imposes further challenges that are not necessarily fronted in the research and development phases.

⁹<https://docs.oracle.com/en/database/oracle/oracle-database/19/tdppt/automatic-database-performance-monitoring.html>

¹⁰<https://docs.oracle.com/en/database/oracle/oracle-database/19/tgsql/sql-tuning-advisor.html>

5. Real Use Cases

This section covers day-to-day real-world use cases where Busca360 helps professionals retrieve relevant data that helps manage the integrity of the company's industrial assets¹¹. Unlike the use cases related to the keyword-based queries that users perform in the company's production environment, the use cases related to searches involving images are only available in the test and validation environment.

Keyword Search. Specialists frequently need to retrieve the details of maintenance and inspection demands, such as description, due date, criticality, priority, or the pending or closed demands of specific locations. For instance, to retrieve the pending (not closed) maintenance notes of platform A-1 (the true identity of the platform has been preserved for privacy reasons). Data about the technical objects (equipment, piping, or installing locations) could also be requested.

The user would then submit the following keyword query: *situation of maintenance notes pending of technical objects of platform A-1*. Figure 3(a) depicts the schema fragment of the Busca360 repository involved in the keyword query response. Figure 3(b) shows the compiled SQL query from the above keyword query, this SQL code will not be shown to the user; it is depicted here just for illustrative purposes. Note that the WHERE clause has three equijoins between 4 tables (lines 7 to 9) and two restriction clauses (lines 10 and 11). Figure 3(c) shows a fragment of the answers for the keyword query provided by Busca360 in a tabular way.

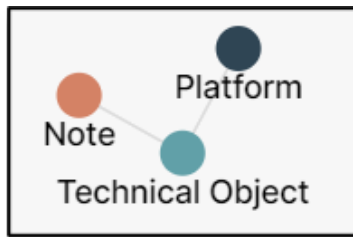
To add more information to the answer is quite simple in Busca360, for example, using the *search-by-selection* feature, the user only needs to select the desired attributes, such as: "Start Date" or "Due Date", etc. Also, the user may specify filters using specific patterns, depending on the attribute data type, for example: "*End Date* \geq 01/04/2024" or "*End Date between 01/04/2024 and 30/04/2024*" or "*Situation = Approved*", etc.

Aggregation Search. A common example of information that needs aggregation is to retrieve the number of maintenance notes "closed" in a given work month (for example, in April 2024) for a given platform or business asset. The following aggregation search provides an example: *total of maintenance notes with end date between 01/04/2024 and 30/04/2024 on platform A-23*. Note the keywords *end* and *date* correspond to the property "End Date". Then, the *Knowledge Extraction Module* compiles an aggregation SQL query to answer the search, and Busca360 executes it and shows the result through its UI.

Fuzzy Text Search over Structured Data and Documents. Busca360 supports fuzzy text search over structured data and document content. The contents of documents are stored in the Oracle database using character large object (CLOB) columns. The attribute designed to represent these texts in the *KG* of Busca360 is named "*Long Text*" for the entities: maintenance notes, maintenance orders, maintenance plans, and technical inspection reports.

A daily work use case that involves searches on document data is to retrieve the pending technical inspection reports ("pending" is a semantic value that corresponds to a set of literal values of attribute "*Situation*") that contain the word "leak" in the long text data on platform A-12.

¹¹The real data language is Portuguese; however, the terms were translated to English for convenience in writing this article



(a) Fragment of the Busca360's KG.

```

1. SELECT PENDENCIA('NOTA', T4.STATUS_DOCUMENTO) AS PENDENCIA
2. , T4.SITUACAO_DOCUMENTO AS SITUACAO
3. , T2.COD_PLATAFORMA AS COD_PLATAFORMA
4. , T1.ID_OBJETO AS ID_OBJETO
5. , T4.ID_NOTA AS ID_NOTA
6. FROM OBJETO_TECNICO T1, PLATAFORMA T2, DEMANDA_MANUTENCAO T3, NOTA_MANUTENCAO T4
7. WHERE T1.COD_PLATAFORMA = T2.COD_PLATAFORMA
8. AND T3.ID_OBJETO = T1.ID_OBJETO
9. AND T4.ID_NOTA = T3.ID_DEMANDA
10. AND T2.COD_PLATAFORMA = 'A-1'
11. AND LOWER(PENDENCIA('NOTA', T4.SITUACAO_DOCUMENTO)) = LOWER('SIM')
12. FETCH NEXT 10 ROWS ONLY;
  
```

(b) SQL query compiled.

Platform	Technical Object	Note (Maintenance Notes, Notes)	
Code	Number	Number	Situation
A-1	BSA.10	12831001	Waiting for approval
A-1	2577685	13561189	Order Generated
A-1	2585526	13149662	Order Generated
A-1	2636115	13104701	Approved

(c) Tabular presentation provided by Busca360 to the keyword search answers.

Figure 3. Busca360 response to the keyword-based query: “situation of maintenance notes pending of technical objects of platform A-1”.

Busca360 enables to perform the following query as an example: *technical inspection reports with situation pending and long text contains leak in platform A-12*. The Knowledge Extraction Module compiles an SQL query with a filter over the “Long Text” attribute mapped to the column table `TEXTO_LONGO` using the `CONTAINS` operator indicating a *fuzzy* indicator for the value “leak”. Note that the “Long Text” attribute values must be indexed for this query to succeed; otherwise, an Oracle Text¹² error is raised.

Image Similarity Search. Image similarity search is convenient when the user is trying to locate a particular object, for example. This approach can potentially increase efficiency compared to the manual approach of physically inspecting entire areas and searching for a specific object.

Figure 4 illustrates one particularly useful example, where a user needs to inspect if there are fire extinguishers that are partially obstructed. It is based on identifying all fire extinguishers by image similarity search. A user can then individually inspect each retrieved image for possible obstructions. One downside of this method is the possibility of a fire extinguisher being obstructed at a level that prevents it from being recognized by the image search. In addition, new obstructions might have been introduced, and others might have been cleared during the time spent between the image capture and the search.

¹²<https://docs.oracle.com/en/database/oracle/oracle-database/19/ccapp/index.html>

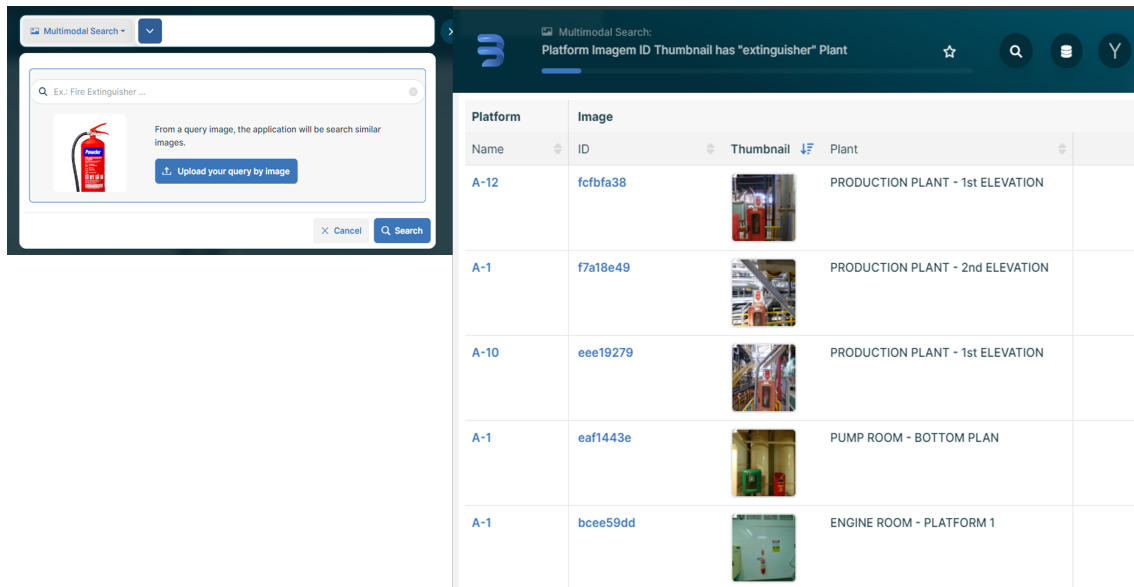


Figure 4. Example of image similarity search for inspection of fire extinguishers.

Multimodal Search. Busca360 also supports multimodal queries specified by combining keywords and sample media objects (restricted to images in the current implementation). This search type is especially useful when the results are expected to contain a particular object (given by the query image) with additional features that further specify the search criteria (given by keywords). For example, the user performs the multimodal query where the query image is shown in Figure 4 and the keyword query is: *platform A-1*. Here, the user requests the image with a fire extinguisher, and the text query indicates that the consultation is limited to a specific location. The result is also similar to that shown in Figure 4, but filtered by the platform with value A-1.

6. Conclusion and Future Work

This article introduced Busca360, a multimodal search application for asset integrity management. The application allows users to specify multimodal queries through a few keywords (writing them or by selection) or media objects (such as image fragments), and returns data that matches the query specification and can be joined together to create an integrated data collection. It represents a paradigm shift in accessing industrial asset databases, both within the company and in the oil and gas industry.

Busca360 enables any user, regardless of their technical computing skills, to retrieve desired information without the need to develop or complete pre-built forms, execute queries directly in the database, or extract data into spreadsheets from other systems or databases and then consolidate that information. It currently runs on the production environment of an oil and gas company to enhance the user's search experience. This includes implementing query recommendations, enhancing the natural language capabilities of the search engine, and including conversational interfaces based on Large Language Models (LLM). The latter presents significant challenges, such as fine-tuning a model for the specific domain repository and generating context-based dialogues using stored data.

Additionally, the research project supporting the application's development aims to evolve multimodal searches. In this way, users can (i) enable users to create more

complex searches combining text and images; (ii) conduct spatial searches, finding images of or near specific locations; and (iii) search for text that appears within images, such as object identification through text present in the image.

Future work focuses on four directions: (i) to incorporate the Natural Language (NL) processing capabilities with the help of LLM, to allow users to retrieve data using NL questions, covering language constructs not supported by keyword queries as reported in [Pinheiro et al. 2023, Nascimento et al. 2023, Nascimento et al. 2024]; (ii) to extend the multimodal capabilities; (iii) to include more data from the actual and new data sources; and (iv) to allow users to define business facets according to this context.

Acknowledgements

This work was partly funded by FAPERJ under grants E-26/200.834/2021, by CAPES under grant 88881.134081/2016-01 and 88882.164913/2010-01, by CNPq under grant 305.587/2021-8, and by Petrobras Contract 2018/00716-0.

References

- Bergamaschi, S., Guerra, F., Interlandi, M., Trillo-Lado, R., and Velegakis, Y. (2016). Combining user and database perspective for solving keyword queries over relational databases. *Information Systems*, 55:1–19.
- Boehm, K. M., Khosravi, P., Vanguri, R., Gao, J., and Shah, S. P. (2022). Harnessing multimodal data integration to advance precision oncology. *Nature Reviews Cancer*, 22:114–126.
- de Oliveira, P., da Silva, A., and de Moura, E. (2015). Ranking candidate networks of relations to improve keyword search over relational databases. In *2015 IEEE 31st International Conference on Data Engineering*, pages 399–410.
- Doan, A., Halevy, A. Y., and Ives, Z. G. (2012). *Principles of Data Integration*. Morgan Kaufmann, San Francisco, CA, USA, 1st edition.
- Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., et al. (2021). An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference ICLR 2021*, page 21. OpenReview.net.
- Espíndola, D. B., Fumagalli, L., Garetti, M., Pereira, C. E., Botelho, S. S., and Ventura Henriques, R. (2013). A model-based approach for data integration to improve maintenance management by mixed reality. *Computers in Industry*, 64(4):376–391.
- Garcia, G. M. (2020). *A Keyword-based Query Processing Method for Datasets with Schemas*. PhD thesis, Graduate Program in Informatics, PUC-Rio.
- García, G. M., Izquierdo, Y. T., Menendez, E., Dartayre, F., and Casanova, M. A. (2017). Rdf keyword-based query technology meets a real-world dataset. In *Proceedings of the International Conference on Extending Database Technology*, pages 656–667.
- Izquierdo, Y. T., Garcia, G. M., Lemos, M., Novello, A., Novelli, B., Damasceno, C., Leme, L. A. P. P., and Casanova, M. A. (2021). A platform for keyword search and its application for covid-19 pandemic data. *Journal of Information and Data Management*, 12(5).

- Izquierdo, Y. T., García, G. M., Menendez, E. S., Casanova, M. A., Dartayre, F., and Levy, C. H. (2018). Quiow: a keyword-based query processing tool for rdf datasets and relational databases. In *International Conference on Database and Expert Systems Applications (DEXA)*, pages 259–269. Springer.
- Li, X., Yang, J., and Ma, J. (2021). Recent developments of content-based image retrieval (cbir). *Neurocomputing*, 452:675–689.
- Molina, E., Hamazaki, G., Izquierdo, Y., Lemos, M., Britto, P., Corseuil, E., and Garcia, R. (2024). A proposal of a knowledge graph for digital engineering systems integration for operation and maintenance activities in industrial plants. In *XX Brazilian Symposium on Information Systems (SBSI)*.
- Nascimento, E., García, G., Victorio, W., Lemos, M., Izquierdo, Y., Garcia, R., Leme, L., and Casanova, M. (2023). A family of natural language interfaces for databases based on chatgpt and langchain. In *42nd International Conference on Conceptual Modeling – Posters&Demos*, pages 1–5.
- Nascimento, E., Izquierdo, Y., García, G., Coelho, G., Feijó, L., Lemos, M., Leme, L., and Casanova, M. (2024). My database user is a large language model. In *26th International Conference on Enterprise Information Systems*, pages 800–806.
- Nguyen, T. H., Prinz, A., Friisø, T., Nossun, R., and Tyapin, I. (2013). A framework for data integration of offshore wind farms. *Renewable Energy*, 60:150–161.
- Pinheiro, J., Victorio, W., Nascimento, E., Seabra, A., Izquierdo, Y., Garcia, G., Coelho, G., Lemos, M., Leme, L. A. P. P., Furtado, A., et al. (2023). On the construction of database interfaces based on large language models. In *19th International Conference on WEBIST*, pages 373–380.
- Ramada, M. S., da Silva, J. C., and de Sá Leitão-Júnior, P. (2020). From keywords to relational database content: A semantic mapping method. *Information Systems*, 88:101460.
- Raposo, A., Santos, I., Soares, L., Wagner, G., Corseuil, E., and Gattass, M. (2009). Environ: Integrating vr and cad in engineering projects. In *IEEE Computer Graphics and Applications*, volume 29.