AI Agent for PMO: An exploration of n8n orchestration and prompt engineering applied to a project's communications [1]

Luis Sanhueza B.

Santiago, Chile

1. Summary

During the execution of large projects, high volumes of technically and temporally fragmented information are generated, dispersed in diverse formats (spreadsheets, PDFs, word processors, etc.) and physical locations. Although it is understood that the project manager and his work team can access all this information, paradoxically they cannot apply it to their decision-making as they would like.

Along these lines, the motivation arises to explore whether artificial intelligence tools and prompting methodologies can integrate diverse reports and generate prescriptive analyses of greater value. This exploration is materialized in a prototype of a PMO Agent that uses progress reports as a data source to deliver, on demand, integrated information and suggest preventive and corrective actions.

The Agent relies on platforms such as Hostinger[2], n8n[3], Supabase[4] and Postgres[5], and applies prompt engineering patterns (Persona, Disambiguation, System Simulator) to align its responses with PMBOK standards. Five critical dimensions were addressed: physical and cost projection, schedule evolution, project status, task tracking, and risk

[1] How to cite this paper: Sanhueza, L. B. (2026). AI Agent for PMO: An exploration of n8n orchestration and prompt engineering applied to a project's communications; *PM World Journal*, Vol. XV, Issue V, May.

[2] ***Hostinger*** *provides optimized VPS infrastructure with 1-click installation template and official API node, allowing you to run n8n in a self-managed, private, and cost-effective way*

[3] ***N8N*** *is an open-source workflow automation platform that allows you to connect multiple applications and services to automatically execute tasks.*

[4] ***Supabase*** *is an open-source Backend-as-a-Service (BaaS) platform that allows developers to build applications quickly without having to manually configure and manage servers.*

[5] ***Postgres*** *is an open-source, object-oriented relational database management system (RDBMS) recognized for its high reliability, high performance, and ability to handle complex data loads. It uses SQL for queries and supports advanced features, ACID (atomicity, consistency, isolation, and durability) features, and is highly extensible*.

management—the latter incorporating RAG technology to maximize the veracity of content.

The exercise was developed by simulating a real estate construction program of 1,032 activities, complemented by 3 weekly reports of physical progress and risk monitoring.

The results obtained are encouraging in this exploratory context (simulated project, structured data, defined rules). The Prototypical Agent showed that it is technically feasible to integrate diverse and dispersed data (schedule, physical advances, risks) in a single source (PostgreSQL + vectors) and query them on demand in natural language; that using prompting patterns it is possible to detect inconsistencies in the information and alert on them (not just raw data), and that it is possible to cross management dimensions (such as schedule and risks) in a single query, generating a prescriptive analysis aligned with PMBOK.

But it is suggested to be cautious with the results. Far from replacing human communication, the PMO Agent protects it by reducing noise, ordering the flow of information, and freeing leaders to focus their energy on judgment, negotiation, and decision-making in context.

Statement on the use of Artificial Intelligence:

The author declares that artificial intelligence has been used exclusively as an object of technical study and for editorial support purposes for the grammatical optimization of the manuscript. The PMO Agent's logical architecture, prompt engineering development, data analysis, and presented conclusions are the author's original intellectual work.

2. Fundamentals and Driving Idea of the exercise

The development of the **"PMO AGENT"** prototype is based on the need to address the communicational frictions that affect the success of projects. It is a practical laboratory to observe how AI can act as a bridge to transform raw and scattered data into strategic analysis.

A. The Value of Comprehensive Analysis in the Face of Dispersion

In a conventional PMO, the challenge is not only the availability of the data, but the difficulty of processing it holistically. The system does not intend to generate data by itself, but to deliver high-value information through a comprehensive analysis that crosses multiple dimensions: physical advances, schedules and risks. The curiosity behind this project is to verify whether an AI Agent can digest this complexity and offer a unified view that a human analyst, under pressure or fatigue, might not detect in time.

B. Mitigating Human and Organizational Barriers

The prototype shows that the technology properly applied can maintain a constancy in the reading of different types of reports, stored in different folders, with different formats (Spreadsheets, PDF), registered at different times. In short, its capacity to support critical challenges detected in equipment management is confirmed:

- **Continuity in the face of fatigue:** The project does not stop, but the human team experiences saturation. The Agent seeks to provide constant analytical monitoring (24/7), providing continuity so that the analysis does not depend on the state of the equipment.
- **Resilience to absence and carelessness:** Communication is often interrupted by absences or incomplete reports. By centralizing analysis in a single source of truth (PostgreSQL), AI allows strategic conclusions to flow without relying solely on an individual's momentary zeal.
- **Synchronization of operational chance:** Much of what happens in a project escapes the original plans. This exercise explores how AI can integrate unforeseen events, unifying structured data with the context of unstructured documents (PDFs) to reduce uncertainty.

C. A Humble but Strategic Beginning

We recognize that this initiative is just a "drop in the water" in the vast ocean of AI. It is not intended to have solved the total complexity of the communicational challenge of a PMO, but to set a solid example towards the modernization of management. The value of this prototype lies in its ability to demonstrate that it is possible to put technology at the service of a deeper, more transparent and agile analysis.

3. System Architecture

The system architecture is based on an Orchestrated Specialized Agents model. Unlike a conventional chat, this system uses a structure of five branches of knowledge that segment the business logic to guarantee answers based on evidence and not assumptions.

3.1 The Switch Node: Intent Classifier

The flow begins with a logical routing node (**Switch**) that parses the `user's chatInput`. Using filtering rules, the system identifies whether the query belongs to one of the five critical dimensions, deriving it to its corresponding branch:

- **Projection:** Term forecasts (costs and deadlines).
- **Evolution:** Analysis of trends and historical series.
- **Project:** General condition, milestones and critical path.
- **Task:** Specific detail of WBS activities.
- **Risk:** Uncertainty management and document consultation (RAG).

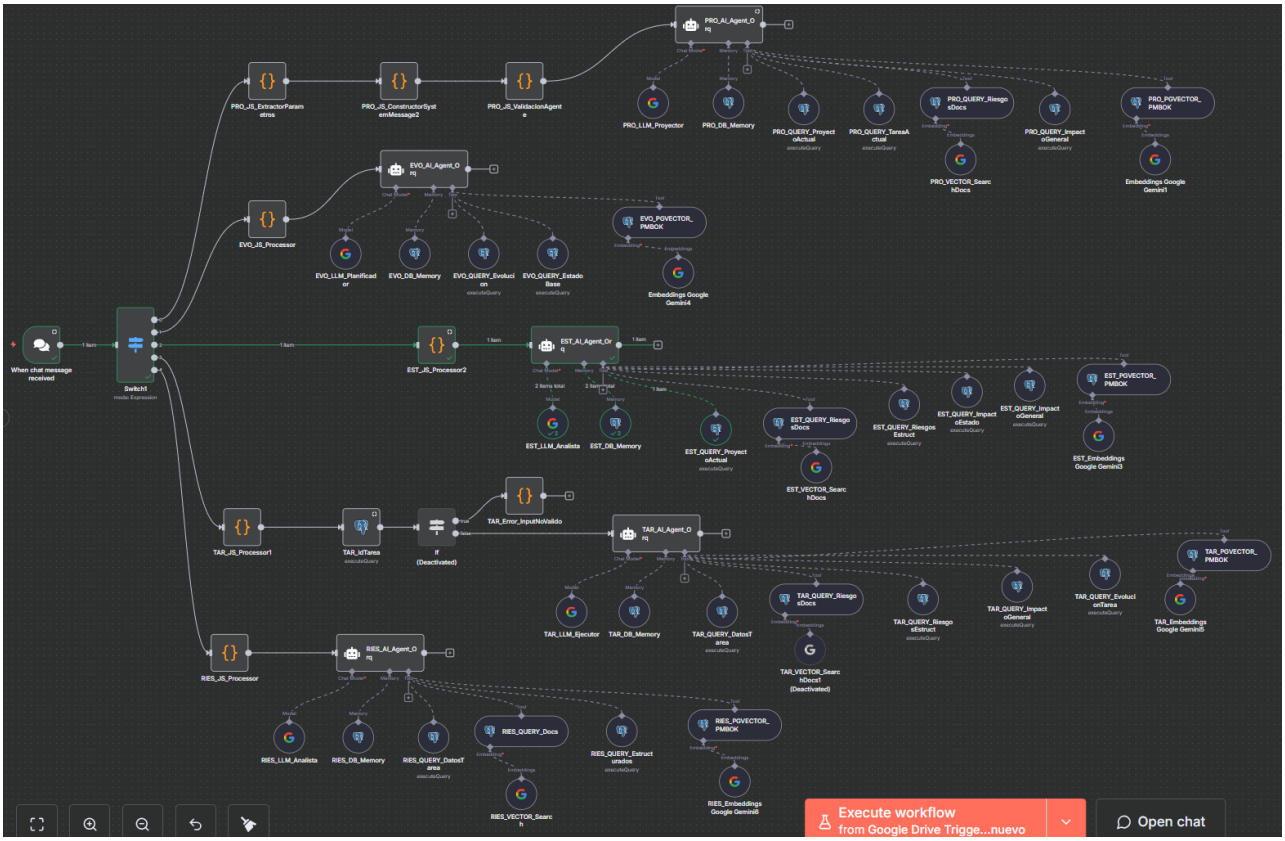

Figure 1: PMO Agent Diagram

3.2 Prompt Engineering: Application of Technical Standards

A central motivation for the development was the practical implementation of the 16 prompting patterns (White et al., 2023) to ensure the quality of the "co-pilot":

- **Person Pattern:** Each branch begins with a `JS_Processor` that defines an immutable personality (e.g., "PMO Director - Risk Module"), forcing the model to reason under PMBOK standards.
- **Disambiguation Pattern:** Extractor nodes clean up the query using Regex to identify exact WBS codes or task names, applying the pattern to avoid working with ambiguous information.
- **System Simulator Pattern:** The agent acts as the interface of the management system, simulating impacts on the critical path using SQL logic.

Complementing these patterns, and for the Project Manager Agent to act with senior professional rigor and not as a generic conversational chat, it is essential to "calibrate" the artificial intelligence engine (Google Gemini 1.5 Pro). This calibration is done through technical parameters that define the limits of your reasoning and the extent of your knowledge, ensuring that the success of the system depends not only on the quality of the data, but on how the engine processes that information.

The success of the system does not depend only on the quality of the data (SQL or PDF), but on how the engine processes that information. If the engine has too much freedom, it can "hallucinate" (invent data); If you have too little response space, you'll leave the reports half-baked. For this reason, specific configurations have been established to balance technical fidelity and analytical capacity.

The two pillars of this configuration are:

1. **Temperature Control (Accuracy vs. Creativity)**: Temperature determines how predictable AI is. For a PMO environment, a near-zero temperature is used. This ensures that the Agent is "deterministic": when asked the same question about costs or deadlines, they will always respond based strictly on the data from the PMBOK® (Project Management Institute, 2021) and the documentation uploaded, without adding subjective interpretations.

2. **Output Window (Tokens)**: Tokens are the units of measurement for text. By extending this limit, we allow the Agent to have the "lung capacity" necessary to write complex status reports, earned value analyses (EVMs), and entire budget deviations, preventing the response from being cut off at critical points in the analysis.

Parameter	Value Set	Impact
Model	Gemini-1.5-Pro	Ability to process bulk contexts of technical documents.
Temperature	0.1 – 0.4 (depending on branch)	Maximum Rigor: Guarantees that the Agent does not invent data outside the original source.
Max Output Tokens	4096 – 8192 (depending on branch)	Continuity: Enables extensive executive reporting and in-depth analysis without interruption.

Top P / Top K	0.95	Lexical coherence in technical vocabulary of projects.

Table 1 : Technical Configuration of the Google Gemini Node

3.3 The Data Model: Single Source of Truth

The system replaces the AI's volatile memory with accurate queries to a **PostgreSQL** structure designed for project integrity:

- **Relational Structure (Master Data):**
  - `cronograma_proyecto`: Stores WBS, durations, predecessors and successors, allowing the calculation of clearances and critical paths.
  - `documents_metadata` and `bodyreport`: Manage the traceability of reports, including status dates and physical progress (Prog vs Real).
  - `riesgos_proyecto`: Structured matrix with probability, impact and action plans linked to those responsible.

- **Vector Structure (Semantic Knowledge):**
  - `n8n_vectors` and `n8n_vectors_PMBOK`: Embedding repositories that allow the agent to "read" PDF documents and technical regulations using RAG technology.

- **Session Memory:**
  - `n8n_chat_histories: Ensures continuity of strategic dialogue by storing message history by session_id.`

3.4 Resilience and Error Control

The design includes layers of technical security. If a user requests information from a non-existent task, the system activates specific error nodes (e.g. `TAR_Error_InputNoValido`), preventing AI from generating a made-up response. This robustness ensures that the Agent is a reliable co-pilot for managerial decision-making.

3.5 Technology Stack and Support Platforms

To ensure data robustness, scalability and sovereignty, the system relies on a set of state-of-the-art tools that allow the integration of complex engineering processes:

n8n (Flow Orchestrator): Acts as the central motor and "nervous system" of the project. It is a workflow automation platform that allows you to connect intelligence branches with databases and LLM models, facilitating the creation of routing logic through the Switch node.

Supabase (BaaS / Backend as a Service): Provides the infrastructure to manage the PostgreSQL database in the cloud, as well as offering essential vector (pgvector) capabilities for embedding storage and running RAG (Generation Augmented by Recovery) technology.

Hostinger (Hosting Infrastructure): It is the hosting platform that supports the deployment of the solution, ensuring the availability of the service and the stable connection between the Google Drive triggers and the n8n orchestrator.

PostgreSQL (Relational Database): It is the Single Source of Truth where all the structured information of the project resides (WBS, schedules, physical advances and risk matrices). It is used for its high reliability and ability to handle complex SQL queries required for Earned Value (EVM) analysis.

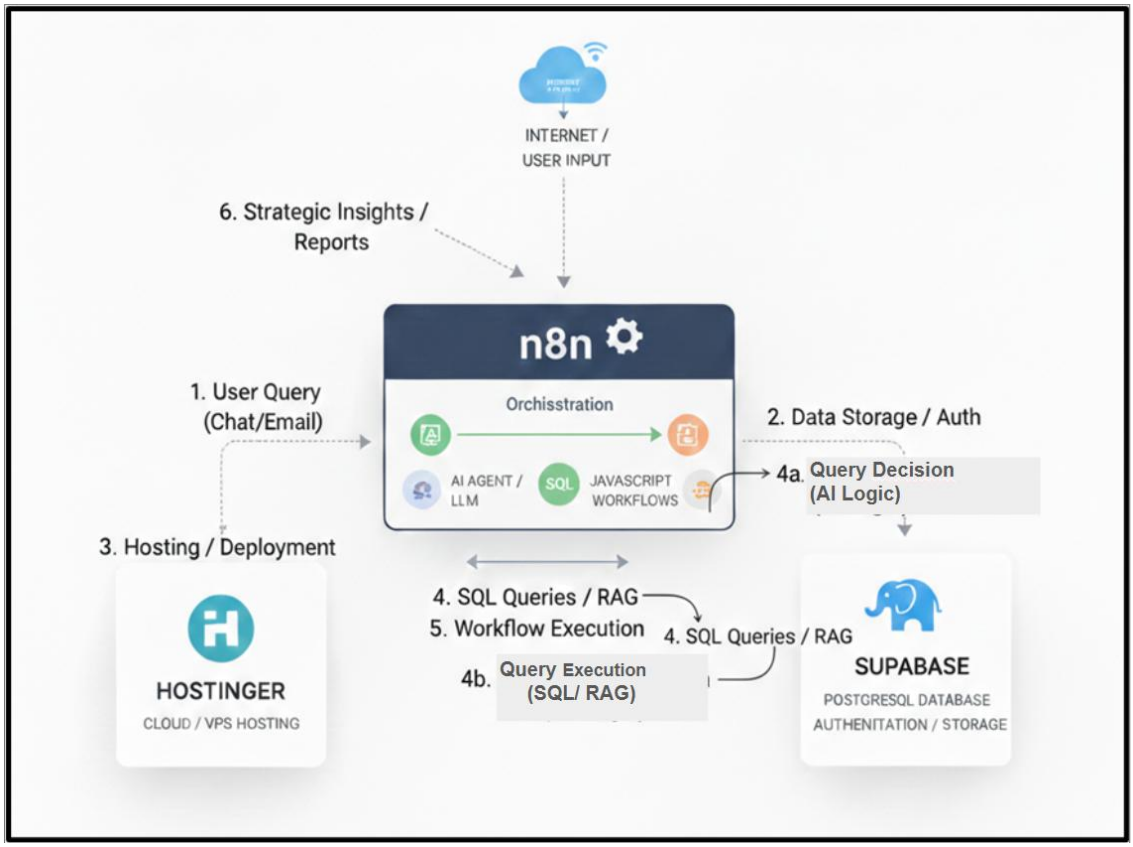

Figure 2 : Orchestration Ecosystem - Strategic PMO Agent Architecture

Entrance and Accommodation (Points 1 and 3): Where the user makes his query. The entire system (the co-pilot's engines) is installed on a server hosted by Hostinger.

The Orchestrating Brain (Point 2): The query escalates to n8n, where the control center is that decides what to do, who should respond, and where to get the necessary information.

The Intelligent Consultation Cycle (Points 4a and 4b): This is where the diagram becomes specific to separate the "idea" from the "action":

4a. Query Decision (AI Logic): Before going to the database, the AI analyzes your question and decides what to look for. For example, if asked about delays, the Agent decides, "I need the SPI and the Critical Path." It is the thought prior to action.

4b. Query Execution (SQL/RAG): Once the decision is made, the system sends the technical instruction to Supabase (Postgres). Here the exact data is extracted or the technical document (RAG) is "read". It is the physical round trip of data.

Strategic Result (Point 6): With the data obtained, n8n processes it and delivers an Insight, which is not just a number, but a recommendation based on the PMBOK.

4. Data Ingestion and Loading Layer (ETL)

The effectiveness of the PMO Agent depends critically on the quality and timeliness of the data. This layer acts as the project's circulatory system, automating the capture of information from various sources and ensuring that the Agent always consults the "Single Version of the Truth".

4.1 Data Flow Automation

The process uses an ETL (*Extract, Transform, Load*) engine orchestrated in n8n that is triggered by event triggers (Google Drive Triggers). Each time a new document is registered, the system identifies its nature and processes it according to three main flows:

- **Schedule Logging:** Extracts WBS structure, durations, dependencies (predecessors/successors), and critical path logic from spreadsheets, populating the cronograma_proyecto table.

- **Weekly Physical Report Log:** Processes progress data. It separates the information into documents_metadata (headers, status dates, global percentages) and body report (detail by activity), allowing a granular analysis of progress.

- **Risk Management:** Captures the structural risk matrix and weekly risk reports, feeding both relational tables and vector repositories.

- **PMBOK Guide Registration**: The Project Management Institute (PMBOK Guide) regulations are fragmented, converted into embeddings and stored in independent vector repositories (n8n_vectors_PMBOK), allowing the Agent to base its analyses and recommendations on international standards through semantic search.

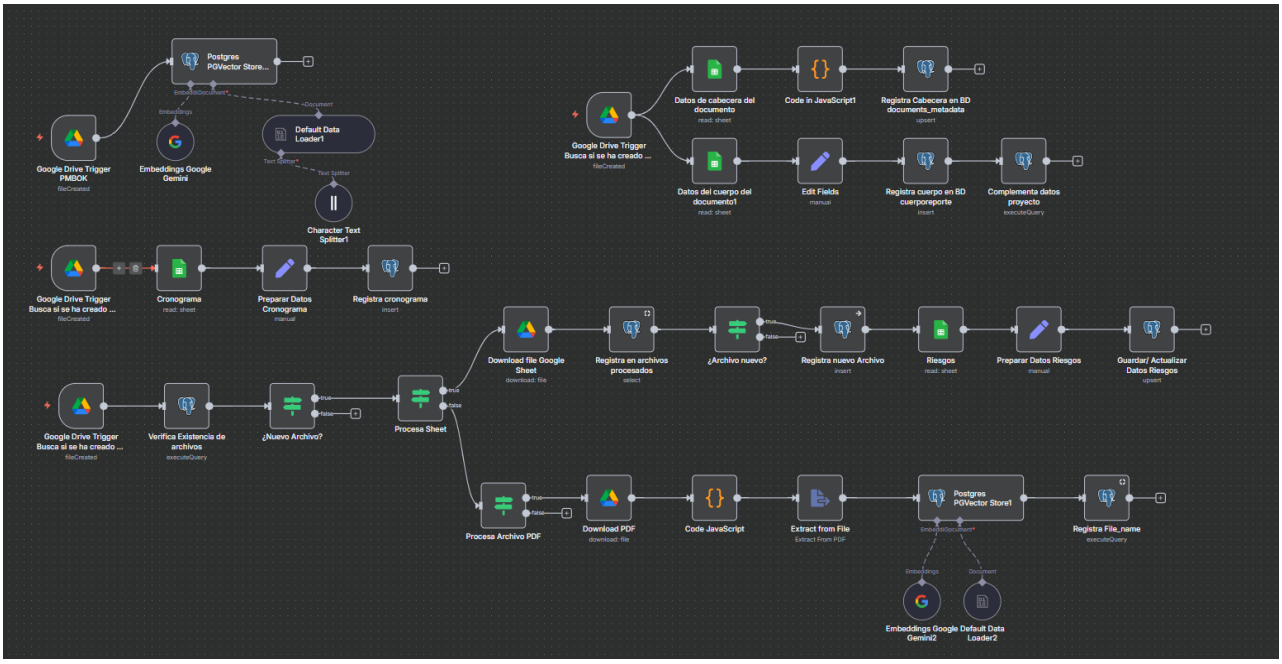

Figure 3: ETL documents

4.2 Hybrid Processing: Structured and Unstructured Data

A key innovation of this system is its ability to treat information in a differentiated way according to its format:

1. **Relational Intake (SQL):** Quantitative data (costs, dates, percentages) that are inserted into normalized tables for accurate Earned Value (EVM) calculations.

2. **Semantic Ingest (RAG):** PDF documents (such as PMBOK or historical risk reports) are fragmented using a Text Splitter and converted into numerical vectors (Embeddings). This allows the Agent to retrieve relevant textual fragments through semantic search, feeding the response context without hallucinations

4.3 Validation and Integrity Control

To avoid duplication and ensure the robustness of the system, the ETL flow implements verification nodes (Verifica Existence). If a file has already been processed, the system identifies it by its file_id, avoiding redundant records that could skew trend analyses or financial projections.

4.4 Resilience to Information Dispersion

This design solves one of the biggest problems of traditional PMOs: the loss of information in shared folders. By centralizing ingestion, the system ensures that regardless of who uploads the file, the Agent has immediate access to comprehensive analysis, mitigating communication barriers and absences of key personnel.

5. The Specialized Agent: PMO Intelligence Branches

The core of the system lies in its cognitive segmentation capacity. To avoid informational "noise" and ensure technical accuracy under PMBOK standards, the Agent is broken down into five specialized branches. Each branch is a self-contained flow orchestrated in n8n, with its own validation logic, SQL tools, session memory, and vector search engines.

5.1 Logical Processing Structure (JS-AI Flow)

Each branch follows a design pattern oriented to the quality of the data, mitigating the tendency to hallucination of LLM models:

- **Parameter Extractor (Disambiguation Pattern):** Using JavaScript routines with regular expressions, the system cleans up the query to identify entities such as WBS code, task names, or query types (general/specific). An evaluation node (IF) validates the input; If it is not possible to identify the requested task, it is derived to a controlled error node that requests greater precision from the user. System Messages incorporate explicit truth-to-use instructions, prohibiting the invention of data not returned by the tools.

- **System Message Builder (Person Pattern):** The Gemini model is instructed to adopt an immutable expert role (e.g. "PMO Director - Risk Module"), limiting its knowledge to the context of the tools provided. The session memory (DB_Memory) ensures the continuity of the strategic dialogue. Parameterized SQL queries normalize dates from the file's metadata (last 8 characters of the filename) and calculate percentage advances using relative cost and duration weights.

- **Orchestration Agent and Tools:** The model does not respond from memory; it decides which tools (SQL queries or vector lookups) to execute to obtain numerical truth from PostgreSQL or normative grounding from PMBOK vector bases. Each tool includes descriptions of purpose, parameters, and enforceability, allowing the Agent to select the appropriate query based on the context.

5.2 Specialization by Dimensions

A. Earned Value Projection and Reporting (EVM) Branch: Aimed at answering "How will we finish?". It uses queries to project and task tables, integrates approved change orders to adjust the BAC, calculates SPI/CPI and projects cost and time EACs, substantiating the recommendations with the PMBOK vector base.

B. Evolution and Historical Trends Branch Focused on "How have we varied over time?". It employs time series with at least 3 historical points, calculates gaps and weekly throughput, and identifies whether a delay is an isolated event or a systemic trend.

C. Risk Management Branch (RAG & Structured)
Transforms uncertainty into actionable management. Cross-references the SQL risk matrix (probability, impact, level, action plan) with the semantic knowledge of vectorized reports, detecting if a documented risk directly impacts the WBS of a critical task and linking specific change orders.

D. Project Status BranchAttention to the general condition, milestones and critical path. It analyzes the current project, the impact on the chain of delayed tasks (identifying direct successors) and structured risks, providing a comprehensive vision of performance with a confidence level (high/medium/low).

E. Specific Task BranchOperational detail of WBS activities. It validates the existence of the task (with error control), retrieves data on progress, historical evolution, impact on direct successors, associated risks and linked change orders, basing the recommendations on PMBOK standards retrieved via vector search.

6. Data Flow and Sequence Diagrams

The PMO Agent operation flow is divided into two master processes: the Ingestion Cycle (ETL), which builds the knowledge base, and the Query Cycle (RAG/SQL), which assists the user.

6.1 The Cycle of Ingestion: Construction of the "One Truth"

This process ensures that AI works on real, up-to-date data, eliminating information dispersion:

1. **Trigger:** A webhook in **n8n** detects the upload of files to Google Drive.

2. **Document Classification:** The system identifies whether the file is a Sheet, a progress report, or a technical document (PDF).
3. **Normalization and Cleanup:** Using JavaScript routines, dates and numerical formats are standardized (handling commas and decimal points).
4. **Dual Persistence:**
   - **Structured:** Milestones and costs are injected into PostgreSQL relational tables (`cronograma_proyecto`, `bodyreport`).
   - **Vector:** The content of PDFs is fragmented and stored in the vector base (n8n_vectors) for semantic searches.

6.2 The Response Cycle: From Chat to Strategic Insight

When a user interacts with the chat, a high-precision sequence based on prompt engineering patterns is triggered:

- **Input and Context:** The message is received and the history is retrieved from `n8n_chat_histories` to maintain the continuity of the dialogue.
- **Intelligent Routing (Switch):** The system identifies whether the query requires Projection, Evolution, Project, Task or Risk data.
- **Predictive Analytics and Tool Query:**
  - The Agent selects the appropriate SQL tool (e.g. `PRO_QUERY_ProyectoActual`) to get fresh metrics.
  - If the query is about uncertainties, turn on the RAG tool (`RIES_QUERY_Docs`) to search the vectorized reports.
- **Synthesis and Recommendation:** The Gemini Agent combines the raw data with your "Director PMO" instruction to deliver an answer that includes: The Analysis (what happened), the Projection (what will happen), and the Suggested Action (what to do).

6.3 Platform Interaction Map

The flow is materialized thanks to the integration of the technological stack:

- **Hostinger/Supabase:** Provides the runtime and PostgreSQL/Vector database.
- **n8n:** Orchestrates each decision and data transformation node.
- **Google Gemini:** Provides the reasoning and synthesis layer under PMBOK standards.

7. Quality Patterns and Metrics

To ensure that the PMO Agent behaves like an expert consultant and not a generic text generator, the design incorporates validation layers and performance metrics based on software engineering and project management standards.

7.1 Ticket Validation and Match Score

The system applies strict quality control before processing any queries to prevent the propagation of errors:

- **Entity Detection:** Using the PRO_JS_ExtractorParametros node, the system assigns a match score to key terms (WBS or task name).
- **Ambiguity Control:** If the system detects that the user requests a task but the "Match Score" is low or non-existent, the Disambiguation pattern is activated, redirecting the user to the TAR_Error_InputNoValido node to request greater precision.
- **Data Integrity:** Validation nodes such as PRO_JS_ValidacionAgente verify that both the systemMessage and chatInput exist before execution, throwing controlled exceptions if critical data is missing.

7.2 Management of the Strategic Report

The quality of the response depends on the system's ability to "remember" the strategic thread of the conversation:

- **Context Window:** The n8n_chat_histories table in PostgreSQL is used to store the last 5 interactions.
- **Few-Shot Pattern:** The system uses examples from past interactions to "teach" the Gemini model how to structure responses according to the PMBOK standard, improving consistency in successive queries.

7.3 Prompting Patterns for Fidelity

To combat "hallucinations," prompt design patterns have been implemented that act as cognitive safeguards:

- **Person Pattern (PMO Director):** Forces the model to exclusively use data from executed SQL tools, prohibiting the invention of figures or dates.

- **Hidden Assumptions Pattern:** In system messages (e.g. EST_JS_Processor), the model is instructed to explicitly state what information is missing rather than trying to extrapolate it if data is insufficient.

8. Technical Recommendations and Conclusions

The **"PMO AGENT" prototype** has shown that the integration of AI through structured design patterns makes it possible to transform passive communication into a strategic driver. However, the maturity of the system requires addressing technical challenges identified during the exercise.

8.1 Recommendations for Scalability

To strengthen the Agent's robustness and analytical capacity, the following technical actions are suggested:

- **Task Name Normalization:** Implement a more aggressive data cleansing layer in the ETL to reduce noise in SQL queries when the names in the schedule contain special characters or redundant spaces.
- **Context Window Optimization:** Evaluate the increase in memory in n8n_chat_histories for long-term projects, allowing the Agent to relate events that occurred several months ago without losing the strategic thread.
- **Prompt Refinement with "Few-Shot":** Incorporate more examples of correct responses (EVM and PMBOK) in System *Messages* so that the Agent learns to handle edge cases, such as tasks with physical progress but without recorded cost.

8.2 Conclusions: Towards a Cutting-Edge PMO

The experience of developing this system allows us to conclude that:

1. **Technical Feasibility:** The architecture based on orchestration (n8n), a single source of truth (PostgreSQL) and specialized prompt patterns (Persona, RAG) proved to be technically feasible to integrate structured and unstructured data from a project from multiple reports and status dates, effectively minimizing hallucinations in responses.
2. **Generation of Actionable Analysis:** The system went beyond mere data retrieval, managing to synthesize raw information in prescriptive analysis. It was able to identify inconsistencies (e.g., progress >100%), correlate documented risks with WBS tasks, and suggest actions aligned with the PMBOK based on the available data.

3. **A Prototype as a Foundation**: This exercise constitutes a successful proof of concept that lays the technical and methodological foundations for the development of AI assistants in project management. It demonstrates that it is possible to build an "analytical co-pilot" that transforms historical and current data into strategic insights, marking a clear path for future more scalable developments.
4. **A Valuable Starting Point:** Although the system is a small piece in the vast field of AI, this exercise puts project management on par with current technologies, establishing a solid foundation for the modernization of communications on large construction sites.

9. Agent Testing and Result Validation

To validate the operability of the prototype, test sessions were carried out using data from weekly construction reports. The objective was to verify that the Agent not only retrieved records from the PostgreSQL database, but that it was capable of generating actionable insights of high strategic value for a Project Manager.

Below are three success stories where the Agent demonstrated the ability to detect inconsistencies, correlate sparse dimensions, and generate prescriptive recommendations aligned with PMBOK.

Case 1: Early Warning of Inconsistency between Physical Progress and Cost

Manager's Query: "What is the progress of the 'Concrete' task?"

Agent Response:
Identified WBS 1.3.1.3, actual progress 40.92% vs scheduled 39.88% (ADVANCED status). Simultaneously, it detected a negative cost deviation of -59.14% (current cost 30.7M vs planned 75.2M).

Strategic Insight:
The Agent warned about the contradiction between the good performance in the schedule and the potential overestimation of the base budget or errors in recording real costs.

Manager Value:
Avoids decisions based solely on physical advancement, allowing the cost baseline to be investigated before the deviation spills over to other tasks.

Case 2: Correlation of Materialized Risks with Critical Path Impact

Manager's Query: "Given the delay of the project, what actions do you recommend to execute according to PMBOK?"

Agent Response:
Identified 9 critical tasks overdue, including one with a 500% delay (*Concrete, beams and chains: 3-4-9-10*). He linked these delays with 6 active risks at the High level, highlighting the materialized R-001 (supply/labor crisis) and the preparation of OC-2016-002. It calculated the knock-on impact on direct successors.

Strategic Insight:
The Agent prioritized the execution of the shock plan (double shifts, external search for labor) and recommended formalizing change orders to adjust baselines.

Value for the Manager:
Centralize schedule, risk, and financial information in a single response, allowing management to focus on the few tasks that really threaten the completion date.

Case 3: Resilient Management in the Face of Partial Data (Without Hallucinations)

Manager's Query: "What dependencies does the 'Masonry' task have?"

Agent's Response:

It explicitly stated: "Status date: Not available", "Actual progress: Not available", "Risks: Not found". It only reported the planned completion date (11/17/2016). Confidence level: LOW. He recommended initiating risk identification (PMBOK 11.2) and collecting progress data.

Strategic Insight:

The Agent did not invent information or speculate. He stated a lack of data and suggested concrete corrective actions.

Manager Value:

Absolute reliability: the Manager knows that he can trust that the Agent will not "hallucinate" answers, even in the face of incomplete information, protecting the quality of decision-making.

9.2 The Leap from Raw Data to Strategic Insight

The tests confirm that the temperature-controlled architecture and sufficient token window allows the Agent to overcome mere data retrieval:

1. **Descriptive Level (The Data):** Extracts exact figures (e.g. "Concretes 40.92%").
2. **Diagnostic Level (The Analysis):** Compares against baseline and history (e.g. "Fluctuating gap", "SPI 1.006").
3. **Prescriptive Level (The Recommendation):** Suggests concrete actions aligned with PMBOK (e.g., "Review cost baseline", "Monitor OC-2016-002").

Conclusion of tests: The Agent proved to be robust even with partial data, consisting of cross-correlations (risks + EVM) and stable in generating long responses after token adjustment. The combination of low temperature + high tokens eliminates response cuts and ensures complete executive reports.

Conclusions

The results of the study confirm three main findings. First, it is technically feasible to integrate diverse and dispersed data (project schedule, physical progress reports, updated risk matrix report, risk monitoring report, PMBOK guide) into a single source and query them on demand in natural language. Second, the use of prompting patterns allows you to detect inconsistencies in the information and alert on them, not just deliver raw data. Third, it is possible to cross management dimensions (such as schedule, physical progress and risks) in a single query, generating a prescriptive analysis aligned with PMBOK.

The tests also confirm that the architecture based on Prompt Patterns and RAG provides the Agent with a "technical criterion" that reduces the cognitive load of the Project Manager, allowing him to focus on managing solutions instead of manually searching for data.

It was observed that the agent not only describes, diagnoses or predicts. Based on the patterns of people trained with PMBOK standards, it suggests corrective and preventive actions based on the best management practices promoted by the Project Management Institute. This leap – from the descriptive report to the prescriptive recommendation with normative support – constitutes the enhancement of the greatest strategic value for the Project Manager.

Despite these encouraging results, the call is to be cautious and not to overestimate them: the agent does not intervene in human communication as a relational phenomenon – which continues to operate in its social dimension, with its trusts and informal contexts – but acts instrumentally by developing the work of integrating data dispersed in multiple sources (timesheets, progress reports, etc.). risk matrices, PDF documents), interpret natural language queries, extract and correlate relevant information from a unified database, and synthesize it into structured, traceable responses.

In short, what a PMO agent like the one developed does, is to absorb operational load from a PMO to execute it whenever someone from the project team needs it. Far from suggesting the replacement of the human being, it makes it possible to move the frontier of their work from the mere search and delivery of raw information to strategic analysis and action.

10. References

Google. (2024). Gemini 1.5 Pro Technical Documentation. Retrieved from https://ai.google.dev/.

n8n.io. (2024). Workflow Automation and Orchestration Documentation. Retrieved from https://docs.n8n.io/.

PostgreSQL Global Development Group. (2024). PostgreSQL 16 Database Documentation. Retrieved from https://www.postgresql.org/docs/.

Project Management Institute. (2021). A Guide to the Project Management Body of Knowledge (PMBOK® Guide) (7th ed.). Newtown Square, PA: Author.

White, J., Fu, Q., Hays, S., Sandborn, M., Olea, C., Gilbert, H., Elnashar, A., Spencer-Smith, J., & Schmidt, D.C. (2023). A Prompt Pattern Catalog to Enhance Prompt Engineering with ChatGPT. Vanderbilt University. Nashville, TN.

Addendum: Agent Inquiries and Responses Log

Some examples are described to see how SQL database tools, RAG, and temperature/token parameters integrate site management.

A. Projection and Performance Branch (EVM)

- **Question:** *"What is the project's projected end date?"*
- **Agent's Response (09/09/2016):**
  - SPI: 1,006 → advance.
  - CPI: 1,022 → savings.
  - Estimated end date: February 5, 2017 (advance of 1.38 days).
  - EAC Cost: 2,499,852,277 (savings compared to the BAC).
  - Confidence level: HIGH (stable SPI and CPI, close to 1).
- **Value: Quantitative projection with PMBOK foundation (EAC = BAC / CPI).**

B. Evolution and Historical Trends Branch

- **Question:** *"How has the project been trending?"*
- **Agent's Response (09/09/2016):**
  - 08/26: 15.15% real vs 15.84% prog (0.69% gap)
  - 09/02: 17.28% real vs 18.55% prog (gap 1.27%)
  - 09/09: 21.40% real vs 21.27% prog (gap -0.13%)
  - Analysis: Significant recovery in the last week (SPI > 1). General growing trend.
- **Value:** Identifies the turning point where the project goes from behind to ahead.

C. Risk and Document Management Branch (RAG)

- **Question:** *"What risks are associated with the 'Concrete' task?"*
- **Agent's Response (09/09/2016):**
  - R-001 (Supply and labor crisis): Probability 4%, Impact 5%, High level, Materialized.
  - Action plan: Shock plan with double shifts + prepare OC-2016-002.
  - Related documents: Delay in concrete supply (alternative supplier activated).
  - Confidence level: HIGH.
- **Value:** Cross-referencing of structured risk matrix with unstructured documents **(PDF).**

D. Task Status Branch (Operational Detail)

- **Question:** *"Give me information about the task 'Formwork'"*
- **Agent's Response (09/09/2016):**
  - WBS 1.3.1.4, Type ABSTRACT.
  - Real progress 32.53% vs scheduled 32.31% → ADVANCED.
  - Cost variance: -48.509.269 (-67.41%).
  - Associated risks: R-001 and R-002 (labour shortage, High level).
  - Note: Despite being advanced, critical risks threaten sustainability.
- **Value: Detects warning signs even in tasks with good apparent performance.**

E. Specific Technical Consultation (WBS)

- **Question:** *"Current and planned cost of the task 'Formwork'"*
- **Agent's Response:**
  - Current Cost: 23,450,251
  - Planned Cost: 71,959,520
  - Deviation: -67.41%
- **Value:** Direct, accurate, and unambiguous response thanks to the disambiguation pattern.

F. Comprehensive Strategic Response and Alignment with PMBOK (Executive Summary)

Base question: "Current status, evolution and risks of the 'Concrete' task" (09/09/2016)

F1. Accurate Situational Diagnosis

The Agent (temperature 0.2, tokens 4096) identified:

- Schedule: Advance of 1.04% (40.92% real vs 39.88% prog), but with a fluctuating historical gap (-2.97%, +3.60%, -1.04%).
- Risks: R-001 materialized (supply/labor crisis, High level), linked to OC-2016-002 in preparation.
- Cost: Negative deviation of -59.14% (current cost 30.7M vs planned 75.2M), critical inconsistency.
- Critical path impact: Extreme delays in subtasks (e.g. 500% in concrete 3-4-9-10 beams) threaten the end date.

F2. Recommendations Structured by PMBOK Areas

- Schedule (6.6): Apply Crashing (double shifts) on tasks with a delay >30%.
- Risks (11.6): Execute shock plan for R-001/R-002 and activate alternative concrete supplier.
- Changes (4.6): Formalize OC-2016-002 and OC-2016-003 in committee; the current BAC does not reflect them.
- Costs (7.4): Audit baseline of the task (CPI 0.408) and validate initial estimate.

Final comment: The Agent integrated three dimensions (schedule, cost, risks) in an executive report, with inconsistency alerts and actions traceable to the PMBOK, demonstrating a leap from raw data to strategic insight.

About the Author

Luis Sanhueza B.

Santiago, Chile

Luis Sanhueza B. is an industrial civil engineer with a master's degree in industrial engineering, with more than 35 years of experience in the management, direction and control of projects of varying complexity. Since 2007, he has been certified as a Project Management Professional (PMP®) by the Project Management Institute (PMI).

Throughout his professional career, he has specialized in project management and the implementation of Microsoft solutions for enterprise project management (EPM). Currently, he offers consulting services in industrial engineering and expertise, while researching the convergence of emerging technologies and project management. His main focus is on the development of prompt engineering architectures for cognitive support and strategic decision making in Project Management Offices (PMOs).

Luis can be contacted at lsanhba@gmail.com