

SPID

SEC Project Intelligence Database

Provenance des données, fonctionnement du pipeline d'extraction
et portrait statistique de 19 227 projets stratégiques
extraits des filings de 500 sociétés cotées américaines (2010–2026)

 DuckDB  10-K · 10-Q · 8-K  Extraction par LLM  Graphe de connaissances  Embeddings

 **Manel Kammoun**
manel.kammoun@uqo.ca

 **Charli Tandja Mbianda**
djerrycharli.tandjambianda@uqo.ca

 **Simon-Pierre Boucher**
simon-pierre.boucher@uqo.ca

 Gatineau — Pavillon Alexandre-Taché, 283, boulevard Alexandre-Taché

 Rapport technique — 8 septembre 2026

Résumé

Ce rapport documente **SPID** (*SEC Project Intelligence Database*), une base de données qui reconstruit le portefeuille de *projets stratégiques* des grandes sociétés cotées américaines à partir du texte brut de leurs documents réglementaires déposés auprès de la Securities and Exchange Commission (SEC). Plutôt que de lire les formulaires 10-K, 10-Q et 8-K comme des documents financiers, SPID les traite comme une source d'*intelligence sur les projets* : usines, centres de données, acquisitions, programmes de R&D, initiatives d'IA, transition énergétique, etc.

Le document répond à trois questions. **D'où viennent les données ?** Elles proviennent du corpus EDGAR de l'UQO : les filings de 500 sociétés du S&P 500 téléchargés depuis l'API publique de la SEC, soit 134 124 documents et 88 Go de HTML brut, découpés en sections. **Comment fonctionne le pipeline ?** Cinq étapes reprenables (ingestion des 10-K, extraction par grand modèle de langage, résolution des mentions en projets, graphe de connaissances, embeddings) persistent leurs résultats dans un unique fichier DuckDB de 2,9 Go. **Que contient la base ?** 153 894 sections analysées, 39 930 mentions de projets, 19 227 projets résolus répartis en 21 types et 11 secteurs, un graphe de 32 226 nœuds et 19 227 vecteurs sémantiques de 384 dimensions. Une trentaine de figures et de tableaux dressent le portrait statistique du jeu de données, et une section finale en expose les limites.

Chiffres clés de la base `spid.duckdb` (exécution complète, juin 2026)

- | | |
|--|---|
| • Entreprises couvertes : 500 (S&P 500) | • Projets avec montant divulgué : 9 321 (48,5 %) |
| • Période des dépôts : janv. 2010 – mai 2026 | • Montant médian divulgué : 718 M\$ US |
| • Sections de filings analysées : 153 894 | • Localisations distinctes : 3 080 |
| • Filings contenant au moins un projet : 21 455 | • Technologies distinctes : 1 730 |
| • Mentions de projets extraites : 39 930 | • Partenaires distincts : 7 640 |
| • Projets résolus : 19 227 | • Nœuds / arêtes du graphe : 32 226 / 33 176 |
| • Types de projets (taxonomie) : 21 | • Embeddings (all-MiniLM-L6-v2, 384-d) : 19 227 |
| • Secteurs GICS : 11 | • Taille du fichier DuckDB : 2,9 Go |

Table des matières

1	Vue d'ensemble	4
2	Provenance des données	4
2.1	La source primaire : EDGAR	4
2.2	Le corpus EDGAR de l'UQO	5
3	Fonctionnement du pipeline	6
3.1	Étape 1 — Ingestion des 10-K	7
3.2	Étape 2 — Extraction par grand modèle de langage	8
3.2.1	Flux unifié	8
3.2.2	Maîtrise du coût : le « gate » et le « focus »	8
3.2.3	Consigne et schéma de sortie	9
3.2.4	Post-traitement et identifiant stable	9
3.2.5	Le backend déterministe rules	9
3.2.6	Exécution sur le corpus complet	9
3.3	Étape 3 — Résolution des mentions en projets	10
3.4	Étape 4 — Graphe de connaissances	11
3.5	Étape 5 — Embeddings et recherche sémantique	11
3.6	Interfaces : ligne de commande et API	11
3.7	Coût, débit et reproductibilité	12
4	Schéma de la base de données	12
5	Portrait statistique des projets	13
5.1	Couverture temporelle et formulaires	13
5.2	Types de projets	14
5.3	Secteurs	15
5.4	Statut et cycle de vie	17
5.5	Montants divulgués	18
5.6	Entreprises	20
5.7	Localisations, technologies et partenaires	21
6	Graphe de connaissances et espace sémantique	21
7	Qualité des données et limites	22
8	Reproduire ce rapport	23
A	Taxonomie des 21 types de projets	23
B	Fichiers du dépôt project-db/	24

1 Vue d'ensemble

La stratégie d'une entreprise se matérialise par des *projets* : la construction d'une usine, l'ouverture d'un centre de données, l'intégration d'une société acquise, le lancement d'un programme de R&D, la migration d'un système ERP vers l'infonuagique. Ces initiatives sont pourtant dispersées, non structurées et enfouies dans les sections narratives des documents réglementaires. Aucune base commerciale (Capital IQ, PitchBook, Gartner) ne les recense de manière systématique et *centrée sur le projet*.

SPID comble cette lacune. L'unité d'analyse n'est ni le document ni l'entreprise, mais le **projet**, décrit par un type (taxonomie contrôlée de 21 catégories), un statut, un montant, des localisations, des technologies, des partenaires, des bénéfices attendus, des risques et une chronologie de mentions dans les filings successifs.

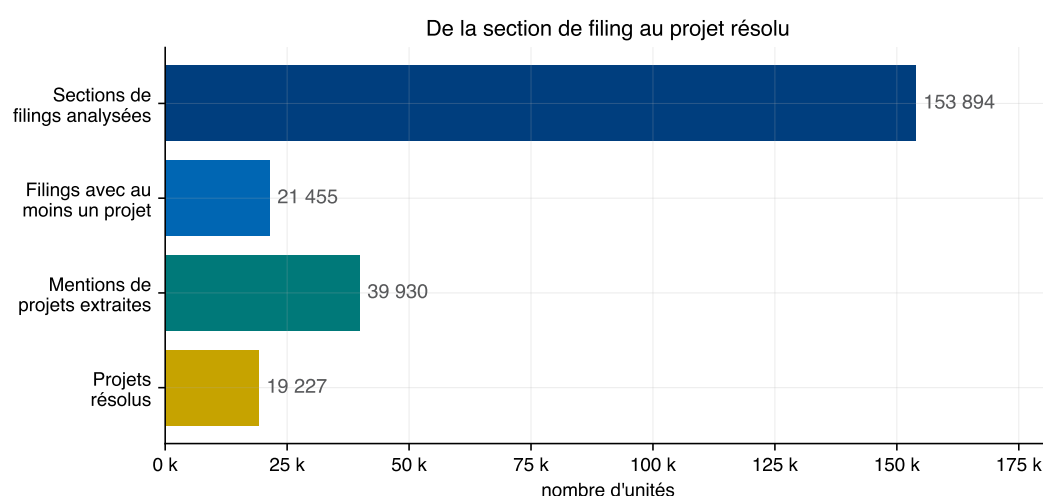


FIGURE 1. L'entonnoir SPID : des 153 894 sections de filings analysées aux 19 227 projets résolus. Chaque projet agrège en moyenne 2,1 mentions.

Le dépôt `project-db/` est volontairement léger (environ 1 900 lignes de Python dans le paquet `spid/`) : il s'appuie sur le corpus EDGAR déjà constitué par l'UQO (section 2) et y ajoute la « couche projet » (section 3). Le résultat tient dans un unique fichier `data/db/spid.duckdb` (section 4), dont la section 5 dresse le portrait statistique.

2 Provenance des données

2.1 La source primaire : EDGAR

Toutes les données textuelles proviennent d'**EDGAR** (*Electronic Data Gathering, Analysis, and Retrieval*), le système de dépôt électronique de la SEC. Toute société cotée aux États-Unis y publie obligatoirement :

- le **10-K**, rapport annuel exhaustif (activités, facteurs de risque, propriétés, rapport de gestion) ;
- le **10-Q**, rapport trimestriel abrégé ;
- le **8-K**, rapport d'événement courant déposé dans les quatre jours ouvrables suivant un fait important (acquisition, contrat, nomination, communiqué de résultats).

EDGAR est public, gratuit et accessible par API (<https://www.sec.gov/edgar>), sous réserve d'un *User-Agent* déclaratif et d'une limite de 10 requêtes par seconde. Le corpus a été téléchargé avec l'identifiant UQO-Research `simon-pierre.boucher@uqo.ca`.

2.2 Le corpus EDGAR de l'UQO

SPID est construit *par-dessus* le corpus UQO_RESEARCH_DB, qui réside dans le répertoire parent `../data` et que SPID consomme en lecture seule. Ce corpus a été constitué par la chaîne `src/acquisition` → `src/parsing` → `src/storage` du dépôt amont.

TABLE 1. Actifs du corpus source utilisés par SPID.

Actif	Volume	Contenu et usage par SPID
<code>data/raw/sec/</code>	88 Go, 500 entreprises	HTML brut de chaque dépôt, rangé par <code><cik>/<form>/<accession>/</code> ; SPID y lit les 10-K.
<code>parquet/filings/</code>	134 124 filings	Une ligne par dépôt : <code>accession_number</code> , <code>cik</code> , <code>ticker</code> , <code>company_name</code> , <code>form_type</code> , <code>filing_date</code> , <code>fiscal_year</code> .
<code>parquet/filing_sections/</code>	127 800 sections	Texte pré-segmenté des 10-Q et 8-K : <code>section_id</code> , <code>accession_number</code> , <code>section_name</code> , <code>item_number</code> , <code>text</code> , <code>word_count</code> .
<code>parquet/companies/</code>	500 lignes	Métadonnées : secteur et sous-industrie GICS, code SIC, état d'incorporation.

Sélection des entreprises (`src/acquisition/sp500.py`). L'échantillon est l'indice **S&P 500**. La liste des constituants (`ticker`, nom, CIK, secteur et sous-industrie GICS, date d'ajout) est lue dans la table de la page Wikipédia *List of S&P 500 companies* et mise en cache dans `data/metadata/sp500_constituents.json`. Chaque ticker est ensuite associé à son identifiant SEC (*Central Index Key*, CIK, complété à 10 chiffres) via le fichier public `company_tickers.json` de la SEC (`cik_mapper.py`). Il s'agit d'un **instantané courant** de l'indice, non d'une composition historique : les sociétés sorties de l'indice avant 2026 sont absentes (biais de survie, section 7).

Téléchargement. Modules `sec_client.py` et `downloader.py`. Pour chaque CIK, l'index des dépôts est lu à `data.sec.gov/submissions/CIK{cik}.json` (bloc recent puis fichiers d'archives paginés). Sont retenus les formulaires **10-K**, **10-Q** et **8-K** au sens strict (les amendements /A sont exclus) déposés à partir du **1^{er} janvier 2010**. Le document primaire de chaque dépôt est téléchargé dans `data/raw/sec/{cik}/{form}/{accession}/`. Le client respecte la limite de la SEC (intervalle minimal de 0,1 s, soit 10 requêtes/s), effectue jusqu'à 5 tentatives avec attente exponentielle (1, 2, 4, 8, 16 s) et tient un journal JSONL des fichiers reçus qui rend le téléchargement reprenable.

Nettoyage et segmentation amont (`src/parsing/`). Le HTML est nettoyé avec `lxml` (balises `script`, `style`, `meta`, `link` supprimées ; tableaux convertis en texte tabulé ; fichiers de plus de 50 Mo ignorés). Le découpage repère les en-têtes « ITEM *n* » et « PART I/II » par expressions régulières : les **10-Q** sont découpés en 7 sections (états financiers, MD&A, risques de marché, contrôles, litiges, facteurs de risque, ventes non enregistrées) ; les **8-K**, trop courts et hétérogènes, sont conservés en une seule section `full_body`. Toute section de moins de 20 mots est rejetée. Les **10-K**, en revanche, n'avaient été laissés qu'en HTML brut : c'est la première tâche de SPID (section 3.1).

TABLE 2. Composition du fichier `filings.parquet` (134 124 dépôts, 500 CIK).

Formulaire	Dépôts	Premier dépôt	Dernier dépôt
8-K	103 927	2010-01-04	2026-05-26
10-Q	22 542	2010-01-05	2026-05-22
10-K	7 655	2010-01-22	2026-05-22

TABLE 3. Sections éligibles à l'extraction (au moins 30 mots), par formulaire, après ingestion des 10-K par SPID.

Formulaire	Sections	Entreprises (CIK)
8-K	103 926	497
10-K	27 164	497
10-Q	22 804	84
Total	153 894	500

Pourquoi seulement 84 entreprises pour les 10-Q ?

Le corpus amont n'a segmenté les 10-Q que pour un sous-ensemble de sociétés ; pour les autres, les 10-Q existent en HTML brut mais n'ont pas été découpés. Les 8-K et les 10-K couvrent en revanche la quasi-totalité de l'échantillon (497 CIK chacun) et fournissent 85 % des mentions de projets (figure 4).

3 Fonctionnement du pipeline

Le traitement s'organise en cinq étapes séquentielles. Chacune lit la sortie de la précédente dans DuckDB, y écrit la sienne de façon **idempotente** (clés stables, INSERT OR REPLACE) et peut être relancée sans double comptage.

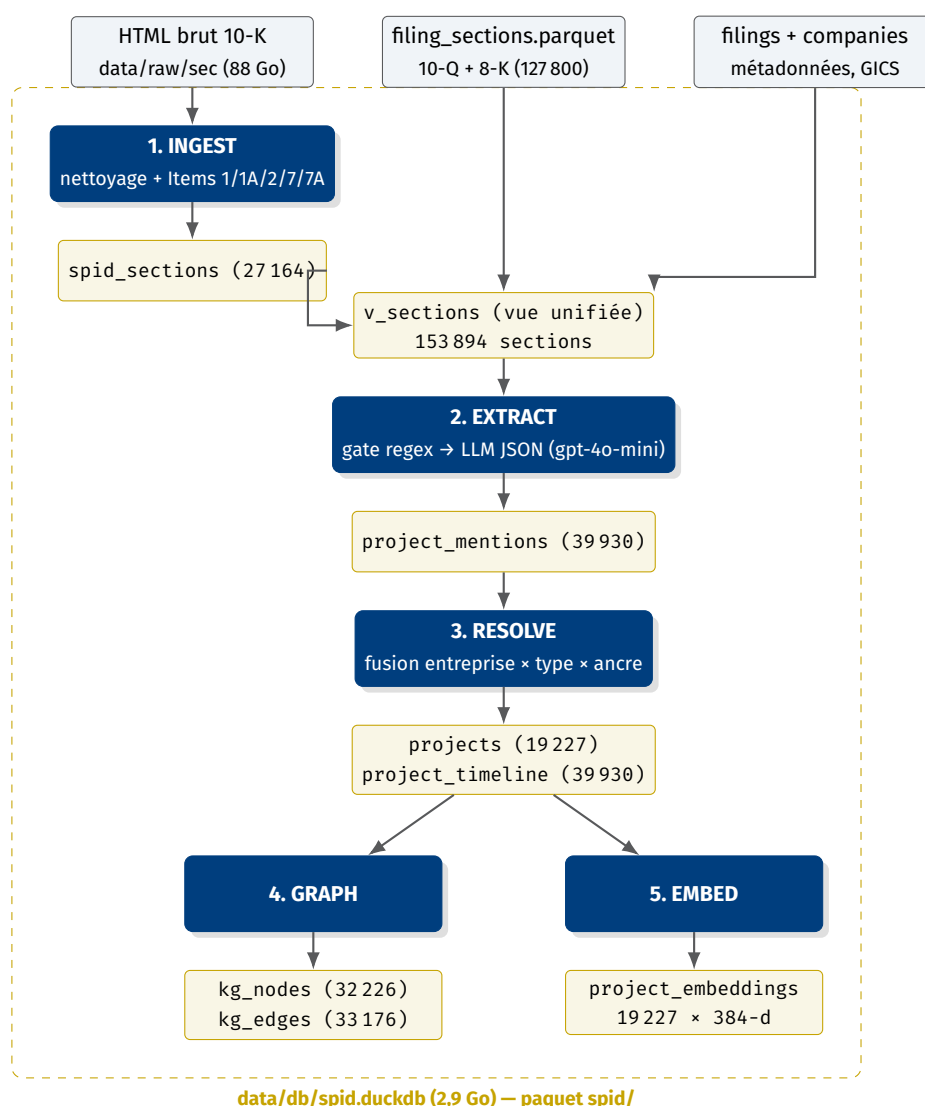


FIGURE 2. Architecture du pipeline SPID. Les rectangles gris sont les entrées (corpus EDGAR de l’UQO), les rectangles bleus les cinq étapes de traitement et les rectangles ocre les tables DuckDB produites.

3.1 Étape 1 — Ingestion des 10-K

Module : `spid/ingest.py`. Pour chaque dépôt 10-K du répertoire `data/raw/sec/<cik>/10-K/<accession>/` :

1. **Sélection du document primaire** : le plus gros fichier `.htm`, `.html` ou `.txt` du répertoire (les pièces jointes et les exhibits sont écartés).
2. **Nettoyage HTML** : suppression des balises `script`, `style` et `table` avec BeautifulSoup (parseur `lxml`), conversion en texte, dé-échappement des entités, normalisation des espaces. Contrairement au nettoyeur amont, les tableaux sont *supprimés* : les chiffres présentés en tableau dans les 10-K n’atteignent donc pas l’extracteur.
3. **Découpage en Items** : l’expression régulière `ITEM\s+(\d+[AB]?)\s*[\.\-:—]` repère les en-têtes et ne conserve que les Items les plus denses en projets : **1** (*Business*), **1A** (*Risk Factors*), **2** (*Properties*), **7** (*MD&A*) et **7A** (*Quantitative Disclosures*). Les occurrences de moins de 400 caractères (renvois de la table des matières) sont écartées ; pour chaque Item, la plus longue occurrence est retenue et tronquée à 600 000 caractères. Si aucun Item n’est reconnu, le corps complet est conservé (`full_body`, 1 230 cas).

4. **Persistence** dans `spid_sections` (identifiant UUID, compte de mots), par lots de 400 lignes. Une accession déjà ingérée est sautée.

TABLE 4. Sections 10-K produites par l'ingestion (7 368 dépôts 10-K, 27 164 sections, 230 millions de mots).

section_name	Item du 10-K	Sections	Mots (moy.)	Mots (méd.)
risk_factors	Item 1A — Facteurs de risque	5 605	10 073	8 562
business	Item 1 — Activités	5 545	7 554	5 869
md_and_a	Item 7 — Rapport de gestion (MD&A)	5 198	11 622	9 935
properties	Item 2 — Propriétés	5 084	821	190
quantitative_disclosures	Item 7A — Risques de marché	4 502	1 875	708
full_body	Corps complet (repli)	1 230	47 719	44 514

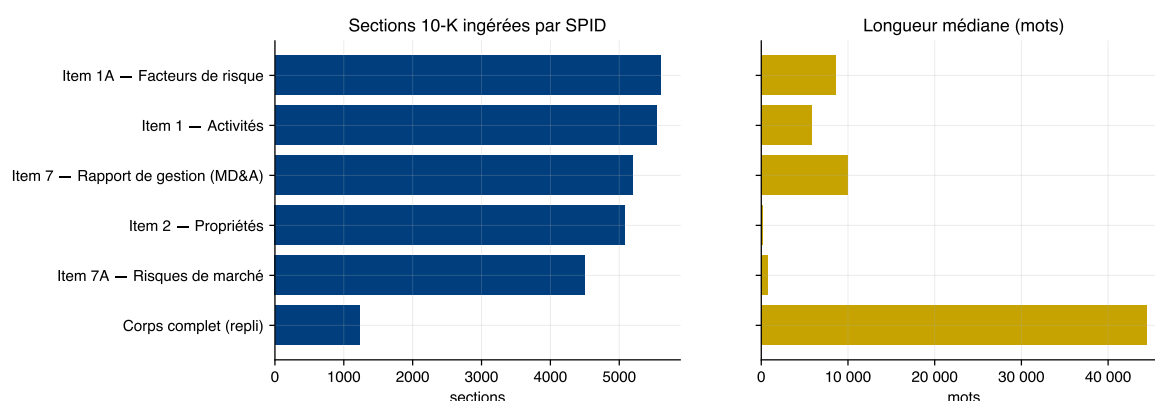


FIGURE 3. Sections 10-K ingérées par SPID : nombre par Item et longueur médiane en mots. Le rapport de gestion (Item 7) et les facteurs de risque (Item 1A) sont les sections les plus longues ; les propriétés (Item 2) sont courtes mais riches en localisations.

3.2 Étape 2 — Extraction par grand modèle de langage

Modules : `spid/pipeline.py` (moteur), `spid/extract/base.py` (interface), `spid/extract/llm.py` (extracteur LLM), `spid/extract/rules.py` (extracteur déterministe). Le backend est *enrichable* : le même moteur alimente indifféremment le LLM ou les règles.

3.2.1 Flux unifié

La vue `v_sections` réunit les sections 10-Q/8-K du corpus amont et les sections 10-K ingérées, jointes aux métadonnées du dépôt (CIK, ticker, formulaire, date) et de l'entreprise (secteur, sous-industrie, SIC, état). Seules les sections d'au moins 30 mots sont éligibles.

3.2.2 Maîtrise du coût : le « gate » et le « focus »

- **Project gate.** Avant tout appel, une expression régulière de vocabulaire-projet (`_PROJECT_GATE`, racines *project, initiative, program, invest, expan, construct, build, launch, develop, deploy, implement, acqui, partner, facilit, capacity, capital, migrat, transform, pipeline, plant, data center, automat, sustainab, renewable, restructur*) filtre les sections. Une section sans signal de projet ne déclenche *aucun* appel API.
- **Focus.** Si la section dépasse 9 000 caractères, seules les *fenêtres* de texte entourant chaque signal (220 caractères avant, 380 après) sont conservées, triées, fusionnées lorsqu'elles se chevauchent et concaténées jusqu'à un budget de 9 000 caractères. Le message final (« SECTION : *nom* » + texte focalisé) est plafonné à 14 000 caractères. Ce mécanisme est déterminant pour les 8-K, transmis en une seule section `full_body`.

3.2.3 Consigne et schéma de sortie

Le modèle (gpt-4o-mini, température 0, mode JSON `response_format=json_object`, 2 000 jetons de sortie au maximum, délai de 120 s) reçoit une consigne système d'analyste financier et renvoie un objet `{"projects": [...]}` dont chaque élément respecte le schéma suivant :

 schéma d'une mention de projet (sortie JSON du LLM)

```
{
  "project_name": "",    // nom court (officiel si mentionné)
  "project_type": "",    // EXACTEMENT un des 21 types de la taxonomie
  "description": "",     // 1 phrase, ancrée dans le texte (pas de spéculation)
  "objective": "",       // le but / le pourquoi
  "status": "planned | in_progress | completed | mentioned",
  "amount_usd": 0,       // montant en USD si indiqué, sinon null
  "locations": [], "technologies": [], "partners": [], "suppliers": [],
  "benefits": [], "risks": [], "years": [],
  "confidence": 0.0      // 0..1 : à quel point c'est un vrai projet concret
}
```

Règles imposées au modèle (extrait textuel de la consigne) : « *Extract ONLY real initiatives, not generic risk-factor boilerplate or hypotheticals. If the text describes no concrete project, return {"projects": []}. Do not invent facts. Numbers/locations must appear in the text. Prefer fewer high-quality projects over many vague ones.* » Les appels sont parallélisés dans un `ThreadPoolExecutor` (16 *workers* lors de l'exécution complète); trois tentatives sont faites sur les codes 429 et 5xx (attentes de 1,5, 3 et 4,5 s), après quoi la section est considérée sans projet.

3.2.4 Post-traitement et identifiant stable

Chaque projet renvoyé est normalisé : le `project_type` est ramené à la taxonomie (correspondance exacte, puis floue; défaut `technology_deployment`); le statut est validé (défaut `mentioned`); une mention sans description est rejetée; les montants sont convertis en nombre; les listes sont tronquées à 8 éléments de 120 caractères; les années ne sont retenues qu'entre 1980 et 2040; la confiance vaut 0,6 par défaut. Si le modèle ne renvoie ni technologies ni années, les expressions régulières de la taxonomie les cherchent dans la description. Les mentions de confiance inférieure à **0,45** sont écartées par le moteur. Un identifiant **stable** est calculé :

```
mention_id = SHA-1(accession | section_id | project_type | ancre)[0:20], ancre = type:nom[0:40].
```

Ré-exécuter l'extraction réinsère les mêmes lignes (INSERT OR REPLACE) sans doublon : l'extraction est **incrémentale et repreneable**.

3.2.5 Le backend déterministe règles

Sans coût d'API, `rules.py` découpe la section en phrases, ne garde que celles de 20 à 700 caractères qui passent le *gate*, les classe par les signaux de la taxonomie (au plus deux types), repère un nom de projet explicite (« Project X », « X Program »), puis calcule une confiance : base 0,40, +0,20 pour un nom explicite, +0,18 pour un montant, +0,10 pour une localisation, +0,06 pour une année, +0,08 pour un statut déterminé, +0,06 pour un verbe d'engagement (*announced, committed*), -0,08 pour un modal d'incertitude (*may, could, potential*). Les phrases sous 0,30 sont abandonnées. Ce backend sert de base de comparaison et d'amorçage; la base actuelle est intégralement issue du backend `llm`.

3.2.6 Exécution sur le corpus complet

Le pilote `run_full_extract.py` traite les 500 CIK par lots de **15** (34 lots), pour borner la mémoire (la matérialisation des 153 894 sections d'un seul coup dépassait la limite de 2 Go des tampons de chaînes Arrow) et journalise les CIK terminés dans `logs/extract_done_ciks.txt`. Résultat : **39 930** mentions, toutes issues du backend `llm`.

TABLE 5. Six mentions tirées au hasard parmi celles de confiance $\geq 0,90$ et de montant > 100 M\$ (types énergie, construction). Montants en M\$ US.

Ticker	Dépôt	Type	Nom et description extraits	M\$	Conf.
AEE	8-K 2022-06-23	Transition énergétique	Renewable Generation Expansion Adding a total of 2,800 megawatts of renewable generation by 2030 and 4,700 megawatts by 2040.	4 300	0.90
NEE	8-K 2012-09-28	Construction d'usine	Canyon Wind Wind Generating Facilities Canyon Wind, LLC is developing and constructing wind generating facilities with a generating capability totaling approximately 219 megawatts located in Arizona and Michigan.	232	0.90
NUE	10-K 2025-02-27	Construction d'usine	Rebar Micro Mill in Pacific Northwest Nucor announced that the Board of Directors approved \$860 million to construct a rebar micro mill in the Pacific Northwest.	860	0.90
SO	8-K 2016-02-02	Construction d'usine	Kemper County Integrated Coal Gasification Combined Cycle Project The Kemper County Integrated Coal Gasification Combined Cycle Project is an initiative by Mississippi Power to construct a facility utilizing integrated coal gasification technology.	5 284	0.90
SRE	8-K 2023-03-20	Construction d'usine	Port Arthur LNG Project The development, construction and operation of large-scale natural gas liquefaction, production and export facilities near Port Arthur in Jefferson County, Texas.	13 000	1.00
NEE	8-K 2021-01-11	Transition énergétique	Solar Base Rate Adjustment (SoBRA) Mechanism The plan is expected to request authority for a Solar Base Rate Adjustment (SoBRA) mechanism to recover the revenue requirements of up to 900 megawatts (MW) of solar projects in 2024 and up to 900 MW in 2025.	140	0.90

3.3 Étape 3 — Résolution des mentions en projets

Module : spid/resolve.py, entièrement en SQL DuckDB. Un même projet est en général mentionné dans plusieurs filings et sur plusieurs années (le 8-K d’annonce, puis les 10-K et 10-Q de suivi). La résolution regroupe les mentions par une clé exacte à trois composantes :

gkey = cik | project_type | ancre,

ancre = {

première localisation (minuscules)si présente,

premier mot de 4 lettres ou plus du nomsinon,

generalà défaut,

et project_id = MD5(gkey). Il n’y a pas de similarité floue : deux mentions fusionnent si et seulement si elles partagent entreprise, type et ancre. Pour chaque groupe, une ligne de projects est produite :

- project_name et description : ceux de la mention la plus confiante (arg_max sur confidence);
- canonical_location : l’ancre géographique; technologies : union dédoublée;
- status : celui du dépôt le plus récent (arg_max sur filing_date);
- total_amount_usd : le **maximum** des montants divulgués (et non la somme, pour ne pas compter plusieurs fois un même montant répété de filing en filing);

- `n_mentions`, `n_filings` (accessions distinctes), `avg_confidence`;
- `first_seen` / `last_seen` : dates de dépôt extrêmes; `first_year` / `last_year` : années extrêmes citées dans le texte.

Chaque mention datée est conservée dans `project_timeline` (dépôt, formulaire, statut, montant, confiance, extrait de 300 caractères), à raison d'une ligne par mention, ce qui permet de suivre l'évolution d'un projet de son annonce à son achèvement. Contrairement à l'extraction, la résolution (comme le graphe et les embeddings) reconstruit ses tables intégralement à chaque exécution.

Résultat : 39 930 mentions → 19 227 projets; 66,9 % des projets ne reposent que sur une mention, tandis que les projets les plus suivis cumulent jusqu'à 185 mentions (tableau 10).

3.4 Étape 4 — Graphe de connaissances

Module : `spid/graph.py`. Les projets résolus sont projetés en un graphe orienté de cinq types de nœuds et trois relations :

Nœud	Préfixe	Relation
Company (500)	C:cik	owns → Project
Project (19 227)	P:project_id	located_in → Location
Location (3 080)	L:lieu	uses → Technology
Technology (1 779)	T:techno	
Partner (7 640)	PR:nom	

Les propriétés sont stockées en JSON (entreprise : cik, ticker, secteur; projet : type, statut, montant, nombre de filings, première et dernière observation; arête owns : montant). Le graphe est exporté en parquet et en Cypher (`data/exports/kg_import.cypher`, 6,6 Mo, instructions MERGE) pour Neo4j. Les nœuds de casse variable (partenaires, noms d'entreprise) sont agrégés par GROUP BY avec `any_value()` afin de respecter la clé primaire. Les 7 640 nœuds Partner sont créés à partir des mentions mais la relation `partners_with` n'est pas encore matérialisée : ils n'ont pas d'arête entrante dans la version actuelle.

3.5 Étape 5 — Embeddings et recherche sémantique

Module : `spid/embed.py`. Le texte de chaque projet (« *nom. type. description. technologies : ...* ») est encodé en un vecteur de **384 dimensions** normalisé (norme L2 unitaire) par le modèle local `all-MiniLM-L6-v2` (sentence-transformers, lots de 256), sans appel API. Les vecteurs sont stockés dans `project_embeddings`. La recherche sémantique (`spid semantic "AI projects launched by banks"`) encode la requête, calcule la similarité cosinus par un simple produit matriciel sur les 19 227 vecteurs (recherche exhaustive, sans index approché), sur-échantillonne cinq fois le nombre de résultats demandés puis applique les filtres SQL éventuels.

3.6 Interfaces : ligne de commande et API

 commandes du paquet `spid` (exécutées depuis `project-db/`)

```
PY=../.venv/bin/python
PYTHONPATH=. $PY -m spid ingest --tickers NVDA,AAPL,MSFT # 1. sectionner les 10-K bruts
PYTHONPATH=. $PY -m spid extract --backend llm --workers 16 # 2. extraire (ou --backend rules)
PYTHONPATH=. $PY -m spid resolve # 3. mentions -> projets + timeline
PYTHONPATH=. $PY -m spid graph # 4. graphe + export Cypher/parquet
PYTHONPATH=. $PY -m spid embed # 5. embeddings MiniLM
PYTHONPATH=. $PY -m spid analytics | stats # tableaux de bord agrégés
PYTHONPATH=. $PY -m spid search "data center" --type data_center --min-amount 5e8
PYTHONPATH=. $PY -m spid semantic "AI projects launched by banks"
PYTHONPATH=. $PY -m spid all --backend llm # tout le pipeline
uvicorn spid.api:app --port 8088 # API HTTP : /projects /semantic /analytics
```

La configuration passe par `project-db/.env` : `OPENAI_API_KEY`, `SPID_LLM_BASE_URL`, `SPID_LLM_MODEL` (défaut `gpt-4o-mini`), `SPID_EMBED_MODEL`, `SPID_BACKEND`. L'interface étant compatible OpenAI, deux variables suffisent pour viser la passerelle du cluster de calcul de l'UQO (`llm.maclustr.io`) ou tout modèle local.

3.7 Coût, débit et reproductibilité

- **Coût** : de l'ordre du dixième de cent US par section grâce au *gate* et au *focus*; l'exécution complète est estimée entre 100 et 130 \$ US de `gpt-4o-mini`.
- **Débit** : appels LLM parallèles (16 *workers*); résolution, graphe et analytique sont du SQL DuckDB pur et passent à l'échelle de millions de mentions.
- **Reprenabilité** : identifiants stables, INSERT OR REPLACE, journal des CIK terminés; une interruption ne re-facture jamais le travail déjà fait.

4 Schéma de la base de données

La base est un unique fichier DuckDB (`data/db/spid.duckdb`, 2,9 Go, dernière écriture le 18 juin 2026). Elle contient sept tables physiques, trois vues sur le corpus amont et une vue unifiée.

TABLE 6. Tables persistées dans `spid.duckdb`.

Table	Lignes	Colonnes
<code>spid_sections</code>	27 164	<code>section_id</code> , <code>accession_number</code> , <code>section_name</code> , <code>item_number</code> , <code>text</code> , <code>word_count</code>
<code>project_mentions</code>	39 930	<code>mention_id</code> , <code>accession_number</code> , <code>cik</code> , <code>section_id</code> , <code>section_name</code> , <code>project_name</code> , <code>project_type</code> , <code>description</code> , <code>objective</code> , <code>amount_usd</code> , <code>amount_raw</code> , <code>locations[]</code> , <code>technologies[]</code> , <code>partners[]</code> , <code>suppliers[]</code> , <code>benefits[]</code> , <code>risks[]</code> , <code>years[]</code> , <code>status</code> , <code>confidence</code> , <code>backend</code> , <code>ticker</code> , <code>company_name</code> , <code>form_type</code> , <code>filing_date</code> , <code>fiscal_year</code> , <code>sector</code>
<code>projects</code>	19 227	<code>project_id</code> , <code>cik</code> , <code>ticker</code> , <code>company_name</code> , <code>sector</code> , <code>project_type</code> , <code>project_name</code> , <code>canonical_location</code> , <code>technologies[]</code> , <code>status</code> , <code>total_amount_usd</code> , <code>n_mentions</code> , <code>n_filings</code> , <code>first_seen</code> , <code>last_seen</code> , <code>first_year</code> , <code>last_year</code> , <code>avg_confidence</code> , <code>description</code>
<code>project_timeline</code>	39 930	<code>project_id</code> , <code>accession_number</code> , <code>filing_date</code> , <code>form_type</code> , <code>status</code> , <code>amount_usd</code> , <code>confidence</code> , <code>snippet</code>
<code>kg_nodes</code>	32 226	<code>node_id</code> , <code>node_type</code> , <code>label</code> , <code>props</code> (JSON)
<code>kg_edges</code>	33 176	<code>src</code> , <code>rel</code> , <code>dst</code> , <code>props</code> (JSON)
<code>project_embeddings</code>	19 227	<code>project_id</code> , <code>model</code> , <code>dim</code> , <code>vector</code> (FLOAT[384])

TABLE 7. Vues définies dans `spid.duckdb`.

Vue	Définition
<code>src_filings</code>	<code>read_parquet(.../parquet/filings/filings.parquet)</code>
<code>src_sections</code>	<code>read_parquet(.../parquet/filing_sections/filing_sections.parquet)</code>
<code>src_companies</code>	<code>read_parquet(.../parquet/companies/companies.parquet)</code>
<code>v_sections</code>	<code>src_sections</code> $\cup$ <code>spid_sections</code> , jointes à <code>src_filings</code> (<code>cik</code> , <code>ticker</code> , <code>formulaire</code> , <code>date</code> , <code>exercice</code>) et <code>src_companies</code> (<code>secteur</code> , <code>sous-industrie</code> , <code>SIC</code> , <code>état</code>)

Point d'attention : chemins des vues

Les trois vues `src_*` pointent en dur vers l'ancien emplacement du corpus (`~/Desktop/UQO/UQO_RESEARCH_DB/data/parquet/...`). Depuis la réorganisation du Bureau, le cor-

pus se trouve sous ~/Desktop/Academique/UQO/UQO/UQO_RESEARCH_DB/. Les sept tables physiques restent parfaitement lisibles, mais toute requête sur `v_sections` ou `src_filings` échoue tant que les vues ne sont pas recréées avec le nouveau chemin. Toutes les figures de ce rapport ont été produites à partir des tables physiques uniquement. Par ailleurs, la colonne `fiscal_year` de `project_mentions` est vide sur toutes les lignes : les analyses temporelles s'appuient sur `filing_date`.

5 Portrait statistique des projets

5.1 Couverture temporelle et formulaires

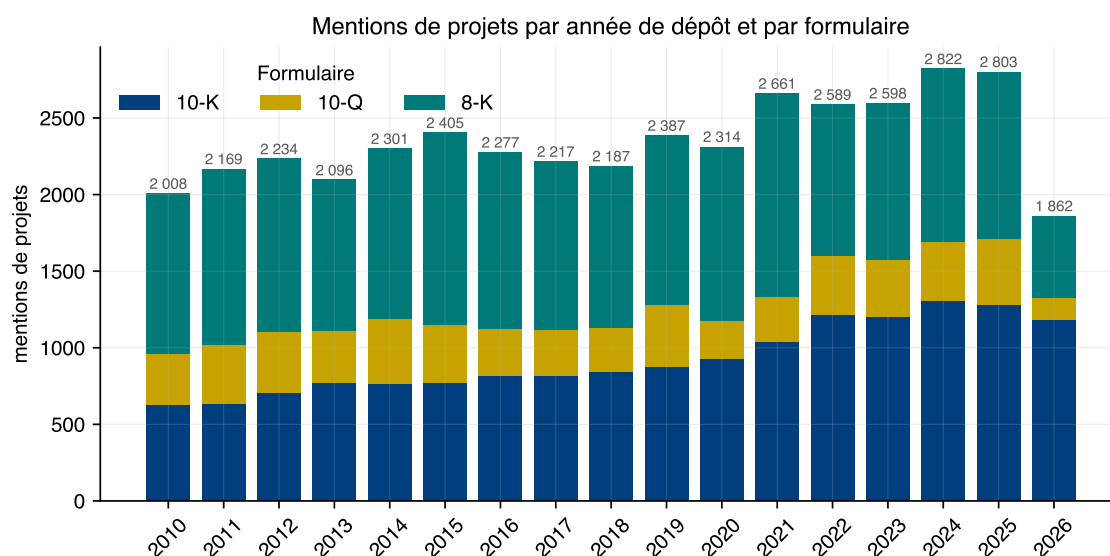


FIGURE 4. Mentions de projets par année de dépôt et par formulaire (18 326 en 8-K, 15 787 en 10-K, 5 817 en 10-Q). Le flux est stable, entre 2 000 et 2 800 mentions par an ; 2026 est une année partielle (dépôts jusqu'au 26 mai).

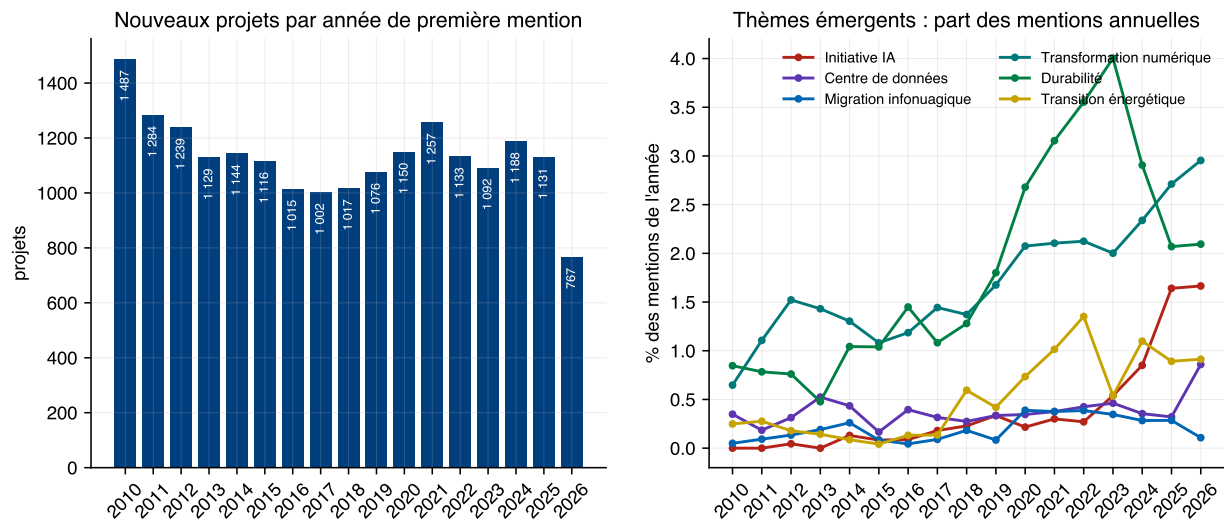


FIGURE 5. Gauche : nouveaux projets par année de première mention. Droite : part des mentions annuelles consacrée à six thèmes émergents. La transformation numérique décolle à partir de 2018, la durabilité culmine en 2023 et les initiatives d'IA passent de moins de 0,3 % des mentions avant 2022 à 1,7 % en 2025–2026.

5.2 Types de projets

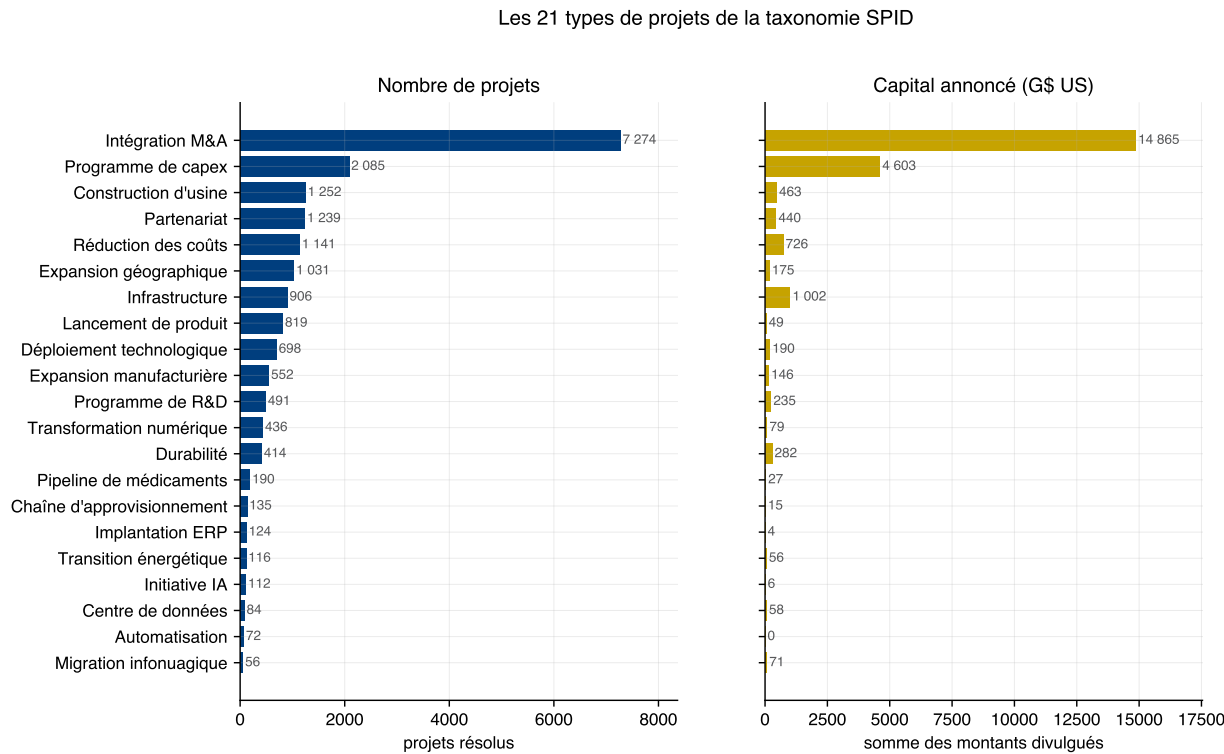


FIGURE 6. Répartition des 19 227 projets selon les 21 types de la taxonomie et capital annoncé par type. L'intégration post-M&A domine en nombre (38 %) comme en capital, suivie des programmes de capex.

TABLE 8. Les 21 types de projets : nombre de projets, capital annoncé cumulé (G\$ US) et montant médian par projet (M\$ US, projets avec montant).

project_type	Libellé	Projets	Capital (G\$)	Médiane (M\$)
ma_integration	Intégration M&A	7 274	14 864,9	864
capex_program	Programme de capex	2 085	4 602,6	1 000
plant_construction	Construction d'usine	1 252	463,0	325
partnership	Partenariat	1 239	440,4	300
cost_reduction	Réduction des coûts	1 141	725,7	204
geographic_expansion	Expansion géographique	1 031	175,0	332
infrastructure	Infrastructure	906	1 002,3	329
product_launch	Lancement de produit	819	48,8	100
technology_deployment	Déploiement technologique	698	190,4	330
manufacturing_expansion	Expansion manufacturière	552	145,6	232
rd_program	Programme de R&D	491	235,3	408
digital_transformation	Transformation numérique	436	78,5	500
sustainability	Durabilité	414	282,2	120
drug_pipeline	Pipeline de médicaments	190	26,8	250
supply_chain	Chaîne d'approvisionnement	135	15,5	125
erp_implementation	Implantation ERP	124	4,4	32
energy_transition	Transition énergétique	116	55,8	675
ai_initiative	Initiative IA	112	6,2	825
data_center	Centre de données	84	58,4	312
automation	Automatisation	72	0,1	35
cloud_migration	Migration infonuagique	56	71,1	1 350

5.3 Secteurs

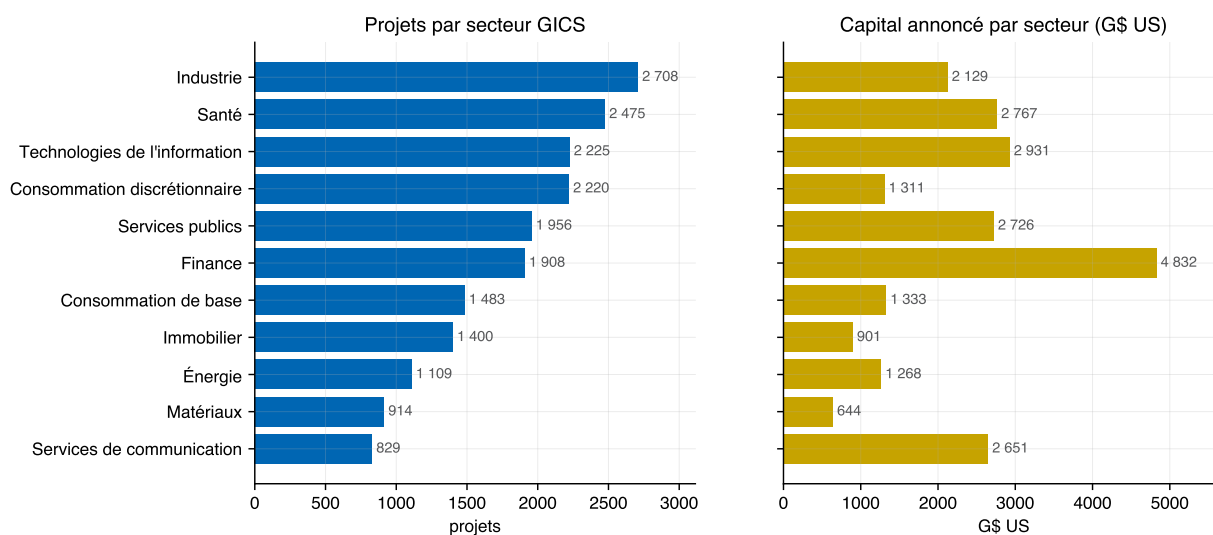
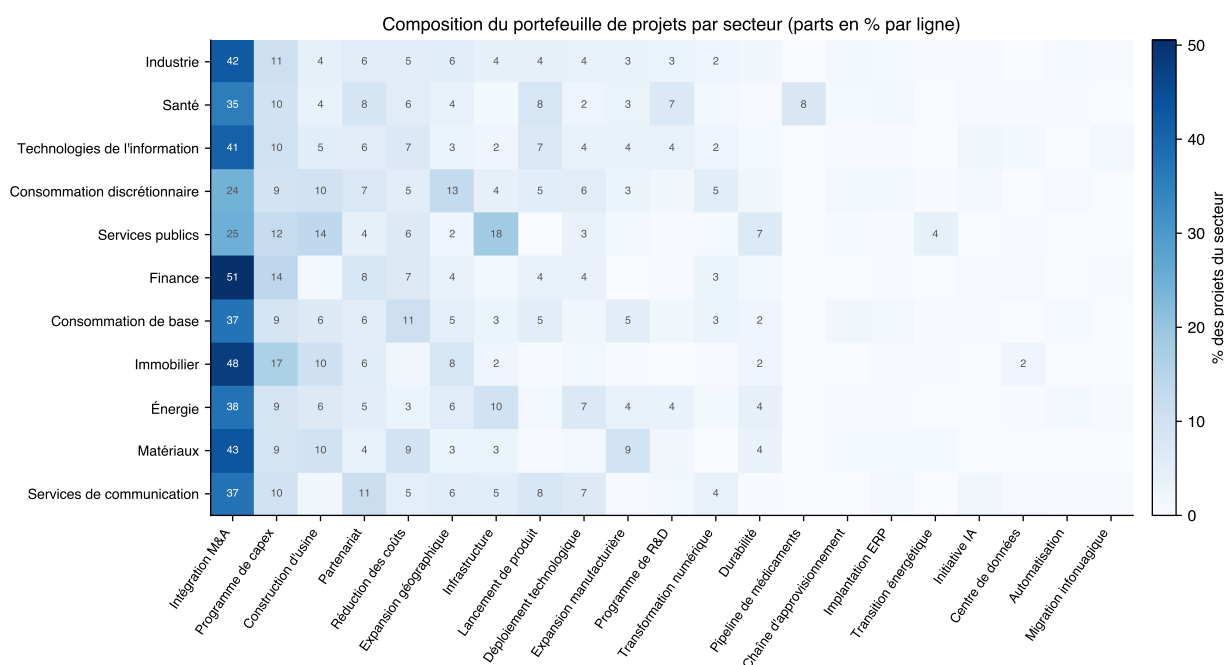


FIGURE 7. Projets et capital annoncé par secteur GICS. L'industrie, la santé et les technologies de l'information comptent le plus de projets; la finance concentre le plus de capital annoncé, du fait des grandes acquisitions bancaires.

TABLE 9. Répartition par secteur GICS : nombre d'entreprises, de projets, projets par entreprise et capital annoncé.

Secteur	Entreprises	Projets	Projets / entr.	Capital (G\$)
Industrie	79	2 708	34,3	2 128,6
Santé	59	2 475	41,9	2 767,3
Technologies de l'information	73	2 225	30,5	2 931,4
Consommation discrétionnaire	48	2 220	46,2	1 310,8
Services publics	31	1 956	63,1	2 725,9
Finance	76	1 908	25,1	4 832,3
Consommation de base	36	1 483	41,2	1 332,5
Immobilier	31	1 400	45,2	901,5
Énergie	21	1 109	52,8	1 267,8
Matériaux	26	914	35,2	643,5
Services de communication	20	829	41,5	2 651,5

**FIGURE 8.** Composition du portefeuille de projets par secteur (parts en pourcentage de chaque ligne). Les secteurs se spécialisent : services publics dans l'infrastructure et la construction d'usines (centrales), santé dans les pipelines de médicaments et lancements de produits, consommation discrétionnaire dans l'expansion géographique, finance et immobilier dans les acquisitions.

5.4 Statut et cycle de vie

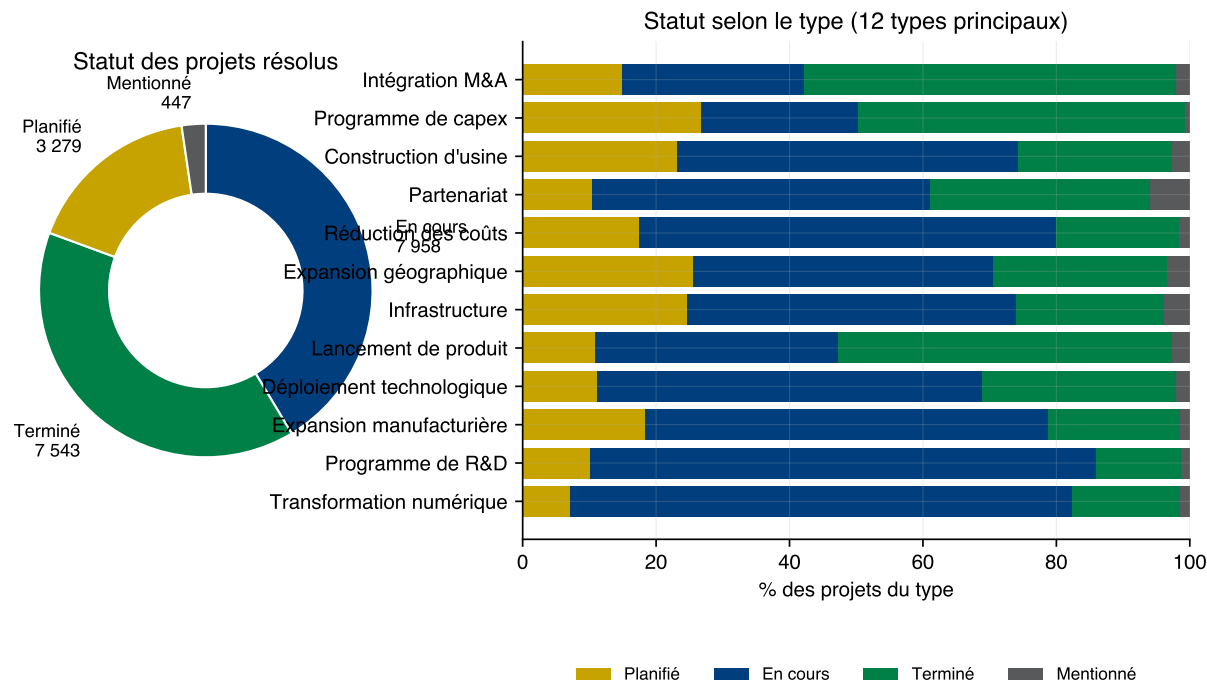


FIGURE 9. Gauche : statut courant des projets résolus. Droite : composition par statut des douze types principaux. Les programmes de capex et les intégrations M&A sont les plus souvent déclarés terminés ; les programmes de R&D et la transformation numérique restent majoritairement « en cours ».

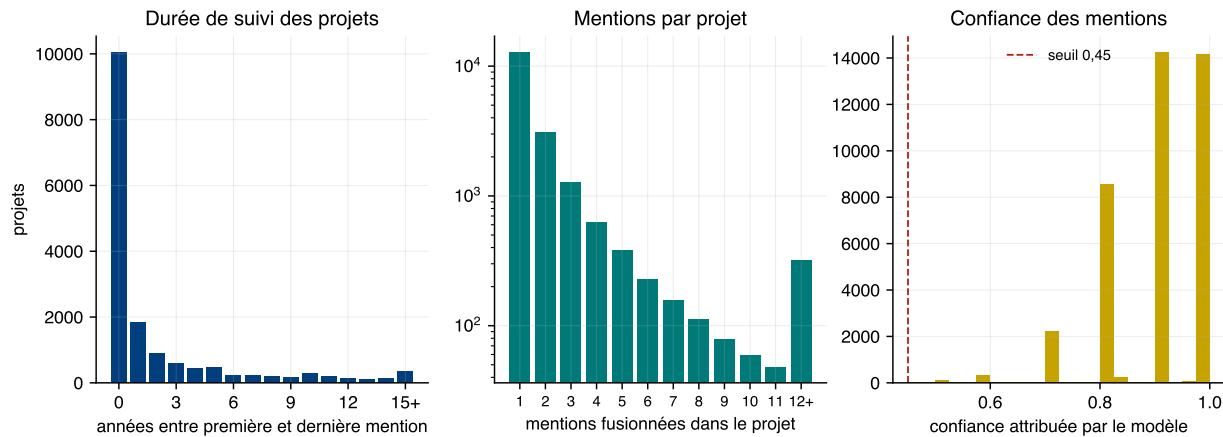


FIGURE 10. Gauche : durée de suivi (écart entre première et dernière année citée) ; 61,4 % des projets ne sont cités que sur une année, 15,6 % sur cinq ans ou plus. Centre : nombre de mentions fusionnées par projet (échelle log). Droite : confiance attribuée par le modèle à chaque mention (moyenne 0,90 ; 93,3 % des mentions à 0,80 ou plus).

TABLE 10. Les dix projets les plus suivis (nombre de mentions et de filings distincts). Les années sont celles citées dans le texte des filings et peuvent inclure des horizons projetés.

Ticker	Projet	Type	Statut	Années	Ment.	Filings
NWSA	Sale of News America Marketing	Intégration M&A	En cours	2020–2026	185	185
TDG	Acquisition of Dukes Aerospace	Intégration M&A	En cours	2009–2033	158	84
SYK	Acquisition of Ascent Healthcare Solutions	Intégration M&A	Terminé	2009–2025	96	54
TYL	Acquisition of Assessment Evaluation Services,...	Intégration M&A	Terminé	2009–2026	95	44
ZBRA	Acquisition of StepOne Systems	Intégration M&A	Terminé	2012–2025	91	53
TDY	Acquisition of Inteltek plc	Intégration M&A	Terminé	2009–2024	88	37
AME	Acquisition of Haydon Enterprises	Intégration M&A	Terminé	2010–2026	81	63
VRSK	Acquisition of Strategic Analytics, Inc.	Intégration M&A	Planifié	2009–2025	75	32
TEL	Acquisition of Sensitive Object	Intégration M&A	Terminé	2010–2026	70	51
WAB	Acquisition of Xorail LLC	Intégration M&A	Terminé	2009–2026	69	41

5.5 Montants divulgués

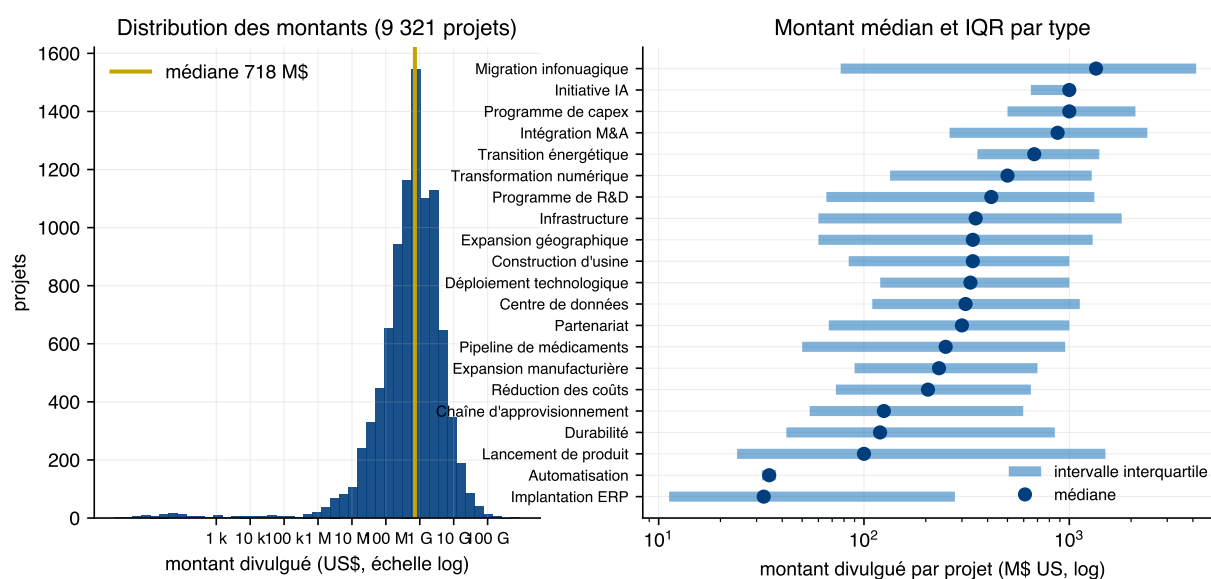


FIGURE 11. Gauche : distribution des montants divulgués (9 321 projets, échelle logarithmique); médiane 718 M\$, moyenne 2 521 M\$. Droite : médiane et intervalle interquartile par type. Les intégrations M&A et les programmes de capex sont d'un ordre de grandeur supérieur aux lancements de produits ou aux implantations ERP.

TABLE 11. Les quinze projets au montant divulgué le plus élevé (au moins trois mentions, hors programmes de capex). Le montant retenu est le maximum observé dans les mentions du projet; plusieurs valeurs de cette liste sont manifestement des erreurs d'extraction (valeur d'entreprise ou chiffre d'affaires attribué à une transaction), voir la note ci-dessous.

Ticker	Projet	Type	Capital (G\$)	Années	Ment.
STT	Acquisition of Goldman Sachs Administration Services	Intégration M&A	200,0	2012–2022	14
GE	Sale of NBCUniversal Interest	Intégration M&A	157,0	2013–2020	7
VZ	Wireless Transaction	Intégration M&A	130,0	2013–2020	4
VZ	Acquisition of Terremark Worldwide, Inc.	Intégration M&A	130,0	2011–2026	52
IVZ	Acquisition of Morgan Stanley's Retail Asset Manag...	Intégration M&A	114,6	2010–2019	13
AVGO	Acquisition of Symantec's Enterprise Security Busi...	Intégration M&A	86,3	2018–2024	24
AVGO	VMware Acquisition	Intégration M&A	83,8	2023–2025	4
CHTR	TWC Transaction	Intégration M&A	79,0	2015–2016	3
CVS	Aetna Acquisition	Intégration M&A	77,0	2017–2021	6
MSFT	Acquisition of Skype	Intégration M&A	75,4	2011–2023	14
NTRS	Acquisition of Omnium LLC	Intégration M&A	73,0	2011–2016	3
MTB	Acquisition of People's United Financial, Inc.	Intégration M&A	64,2	2022–2022	5
DUK	North Carolina Coal Ash Management Strategy	Durabilité	63,0	2005–2030	12
XOM	Acquisition of XTO Energy Inc.	Intégration M&A	63,0	2010–2024	6
BAC	Exchange Agreements	Intégration M&A	61,0	2011–2019	9

Comment lire le « capital annoncé »

Le montant d'un projet (`total_amount_usd`) est le *maximum* des montants divulgués dans ses mentions; le « capital annoncé » d'un type ou d'un secteur est la somme de ces maxima sur les projets concernés. Ces montants sont auto-déclarés, de portée hétérogène (coût total d'un programme, dépense annuelle, valeur d'une transaction, parfois un chiffre d'entreprise capté par erreur) et un même programme pluriannuel peut apparaître sous plusieurs projets si son nom ou sa localisation varie. Le total de 23 494 G\$ sur l'ensemble de la base, comme les valeurs par type ou par secteur, se lit donc comme un **ordre de grandeur supérieur**, non comme un investissement net. Pour une analyse économétrique, les auteurs recommandent de travailler sur le montant *médian* des mentions d'un projet (`table project_timeline`) et d'écarter les valeurs extrêmes.

5.6 Entreprises

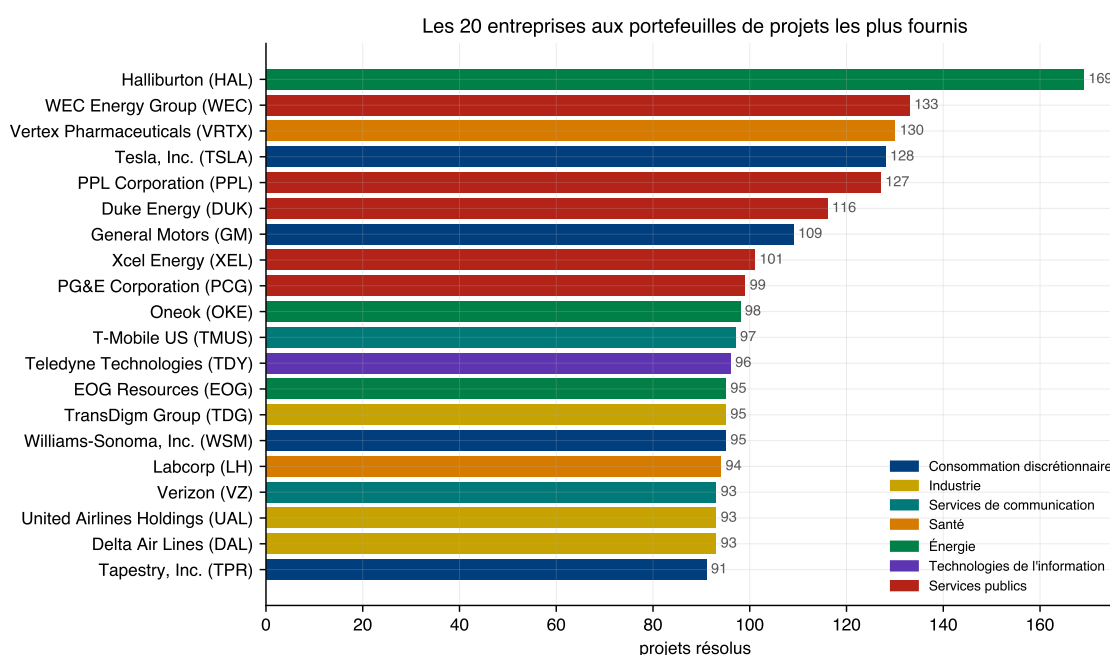


FIGURE 12. Les vingt entreprises aux portefeuilles de projets les plus fournis, colorées par secteur. Chaque entreprise compte en médiane 34 projets (moyenne 38,5, maximum 169). Les services publics (WEC, PPL, Duke, Xcel, PG&E) sont surreprésentés : leurs plans d'investissement réglementés sont décrits en détail dans chaque filing.

TABLE 12. Les quinze entreprises comptant le plus de projets résolus.

Ticker	Entreprise	Secteur	Projets	Capital (G\$)
HAL	Halliburton	Énergie	169	5,5
WEC	WEC Energy Group	Services publics	133	93,8
VRTX	Vertex Pharmaceuticals	Santé	130	31,0
TSLA	Tesla, Inc.	Consommation discrétionnaire	128	40,2
PPL	PPL Corporation	Services publics	127	119,8
DUK	Duke Energy	Services publics	116	589,7
GM	General Motors	Consommation discrétionnaire	109	205,2
XEL	Xcel Energy	Services publics	101	100,0
PCG	PG&E Corporation	Services publics	99	185,5
OKE	Oneok	Énergie	98	102,1
TMUS	T-Mobile US	Services de communication	97	255,1
TDY	Teledyne Technologies	Technologies de l'information	96	54,7
EOG	EOG Resources	Énergie	95	54,3
TDG	TransDigm Group	Industrie	95	72,7
WSM	Williams-Sonoma, Inc.	Consommation discrétionnaire	95	7,6

5.7 Localisations, technologies et partenaires

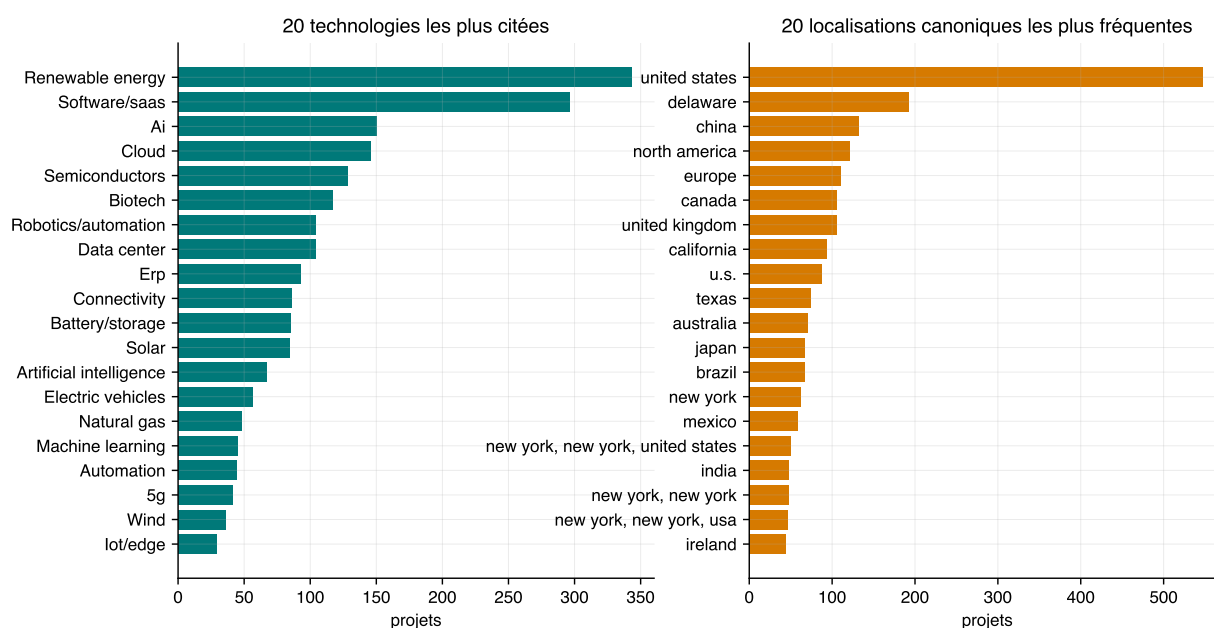


FIGURE 13. *Gauche* : les vingt technologies les plus citées (15,5 % des projets en portent au moins une ; 1 730 libellés distincts). *Droite* : les vingt localisations canoniques les plus fréquentes (46,8 % des projets sont localisés ; 3 080 localisations distinctes).

6 Graphe de connaissances et espace sémantique

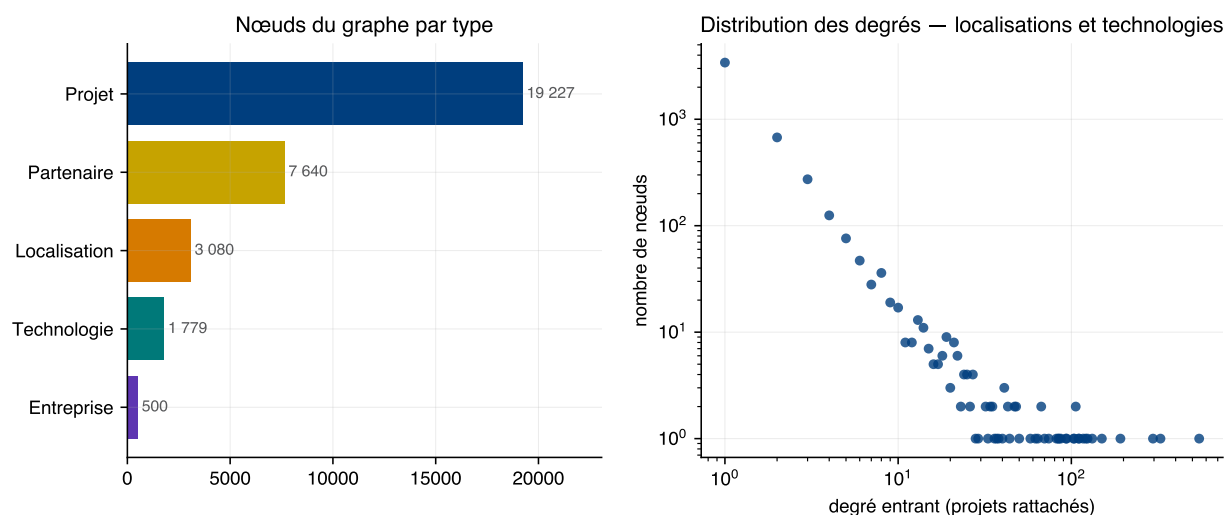


FIGURE 14. *Gauche* : composition du graphe (32 226 nœuds ; arêtes : 19 227 owns, 8 992 located_in, 4 957 uses). *Droite* : distribution des degrés entrants des localisations et technologies (log-log) ; la queue lourde traduit quelques hubs (États-Unis, Texas, Chine ; IA, énergie renouvelable, logiciel) reliés à des centaines de projets.

Le graphe permet des requêtes impossibles sur des tables plates : « quelles entreprises construisent au Texas des installations utilisant des batteries ? », « quels partenaires reviennent dans les projets d'IA du

secteur financier ? ». L'export Cypher (`data/exports/kg_import.cypher`) charge le graphe dans Neo4j en une commande. Les embeddings complètent le graphe par une recherche *par le sens* : deux projets décrits avec des mots différents mais un même objet (une « gigafactory » et une « usine de cellules de batteries ») se retrouvent voisins dans l'espace à 384 dimensions. Le projet de recherche `research/` exploite ces deux couches (détection de communautés technologiques, regroupement *K*-means et projection t-SNE des projets).

7 Qualité des données et limites

- **Divulgarion, non réalité.** SPID mesure ce que les entreprises *disent* de leurs projets dans leurs filings. Une initiative jamais évoquée est invisible, et les incitations à divulguer varient selon les secteurs (les services publics réglementés décrivent tout, les sociétés technologiques peu).
- **Montants.** Auto-déclarés, hétérogènes en portée et absents pour la moitié des projets (voir la note de la section 5). Quelques valeurs aberrantes subsistent (montant d'un rachat d'actions ou d'une valeur d'entreprise attribué à un « projet »).
- **Échantillon non historique.** La liste du S&P 500 est un instantané de 2026 : les sociétés sorties de l'indice entre 2010 et 2026 (faillites, acquisitions, déclassements) sont absentes, ce qui introduit un biais de survie.
- **Tableaux des 10-K supprimés.** L'ingestion SPID retire les tableaux HTML ; les montants présentés en tableau (plans de capex chiffrés, par exemple) ne sont captés que s'ils sont repris dans la prose.
- **Extraction par LLM.** Faux positifs (formules génériques prises pour des projets) et faux négatifs (projets décrits sans vocabulaire de projet) existent. Le seuil de confiance, l'ancrage textuel obligatoire et la température nulle les réduisent, mais aucun échantillon étiqueté à la main d'une taille suffisante ne permet encore d'estimer précision et rappel.
- **Résolution heuristique.** Le regroupement par entreprise, type et nom normalisé peut scinder un même projet nommé différemment d'un filing à l'autre (surestimation du nombre de projets) ou fusionner deux projets homonymes (sous-estimation). Le nom d'un projet peut aussi être une reformulation du modèle plutôt qu'un nom officiel.
- **Années citées.** `first_year` et `last_year` proviennent des années mentionnées dans le texte : elles peuvent précéder la date du dépôt (rappel historique) ou lui succéder (horizon projeté, jusqu'en 2033 ou 2040). Pour dater l'observation, préférer `first_seen` et `last_seen`.
- **Couverture des 10-Q.** Seules 84 entreprises ont des 10-Q segmentés dans le corpus amont.
- **Échantillon.** 500 grandes capitalisations américaines ; la validité externe aux petites entreprises, aux sociétés privées ou non américaines est limitée.
- **Vues cassées.** Les vues `src_*` doivent être recréées avec le nouveau chemin du corpus avant toute ré-exécution du pipeline (section 4).

8 Reproduire ce rapport

 docs/rapport_uqo/

```
python3 make_figures.py          # lit data/db/spid.duckdb, écrit figures/*.pdf et tables/*.tex
latexmk -lualatex rapport_spid.tex
```

Toutes les valeurs numériques du texte (macros \nProjets, \medianeMontant, etc.) et tous les tableaux sont régénérés depuis la base par le script; aucun chiffre n'est saisi à la main.

A Taxonomie des 21 types de projets

La taxonomie (spid/taxonomy.py) est un vocabulaire contrôlé de 21 catégories mutuellement exclusives. Chaque catégorie est dotée d'une description transmise au modèle et de mots-clés servant au *gate*, au backend déterministe et à la correspondance floue du post-traitement.

Type	Définition retenue
manufacturing_expansion	Extension de la capacité de production d'un site existant (lignes, équipements, quarts).
plant_construction	Construction d'une nouvelle usine, centrale, raffinerie ou installation industrielle.
data_center	Construction, extension ou location de centres de données et d'infrastructures de calcul.
cloud_migration	Migration de systèmes ou de charges de travail vers l'informatique.
ai_initiative	Programme d'intelligence artificielle ou d'apprentissage automatique (produits, plateformes, modèles).
digital_transformation	Modernisation numérique transversale (plateformes, canaux, processus).
erp_implementation	Implantation ou remplacement d'un système ERP ou de gestion intégrée.
rd_program	Programme de recherche et développement identifié (hors médicaments).
product_launch	Lancement commercial d'un produit ou service nouveau.
drug_pipeline	Candidat-médicament, essai clinique ou dépôt réglementaire.
ma_integration	Acquisition, cession, fusion ou intégration post-transaction.
geographic_expansion	Entrée sur un nouveau marché géographique, ouverture de magasins ou de bureaux.
capex_program	Programme d'investissement pluriannuel chiffré (plan de capex).
supply_chain	Réorganisation de la chaîne d'approvisionnement, logistique, entrepôts.
automation	Automatisation industrielle ou robotique de processus.
sustainability	Programme environnemental ou de durabilité (émissions, déchets, eau).
energy_transition	Production renouvelable, stockage, décarbonation, électrification.
infrastructure	Réseaux, pipelines, lignes de transport, infrastructures lourdes.
cost_reduction	Programme de réduction des coûts, restructuration, fermetures.
partnership	Partenariat stratégique, coentreprise, alliance, contrat majeur.
technology_deployment	Déploiement d'une technologie identifiée ne relevant d'aucune autre catégorie (type par défaut).

B Fichiers du dépôt `project-db/`

Chemin	Rôle
<code>spid/config.py</code>	Chargement de <code>.env</code> , chemins, journalisation.
<code>spid/store.py</code>	Connexion DuckDB, création du schéma et des vues, primitives d'écriture.
<code>spid/taxonomy.py</code>	Les 21 types, descriptions, mots-clés, correspondance floue.
<code>spid/ingest.py</code>	Nettoyage et sectionnement des 10-K bruts.
<code>spid/extract/base.py</code> , <code>llm.py</code> , <code>rules.py</code>	Interface d'extraction, backend LLM (prompt, gate, focus), backend déterministe.
<code>spid/pipeline.py</code>	Moteur d'extraction : sélection des sections, parallélisme, écriture des mentions.
<code>spid/resolve.py</code>	Fusion des mentions en projets et chronologies.
<code>spid/graph.py</code>	Graphe de connaissances et exports parquet/Cypher.
<code>spid/embed.py</code> , <code>spid/search.py</code>	Embeddings MiniLM, recherche sémantique et filtrée.
<code>spid/analytics.py</code> , <code>spid/api.py</code> , <code>spid/__main__.py</code>	Agrégats, API FastAPI, interface en ligne de commande.
<code>run_full_extract.py</code>	Pilote d'extraction par lots de 15 CIK avec points de reprise.
<code>data/db/spid.duckdb</code>	La base (2,9 Go).
<code>data/exports/</code>	<code>kg_nodes.parquet</code> , <code>kg_edges.parquet</code> , <code>kg_import.cypher</code> .
<code>logs/</code>	<code>ingest.log</code> , <code>extract.log</code> , <code>post.log</code> , <code>extract_done_ciks.txt</code> .
<code>research/</code>	Projet de recherche aval : scripts 01–10, figures, tableaux et article (<code>paper/main.pdf</code>).
<code>docs/</code>	Documentation, dont ce rapport (<code>docs/rapport_uqo/</code>).