

ADV-DB

MACSIN4A0325: Advanced DB Management





Table des matières

1

Introduction

Présentation du dataset

2

Modélisation

Schéma, clés primaires

3

Contexte

Environnement

4

Sécurité

Privilèges, profiles

5

Optimisation

Requêtes SQL

6

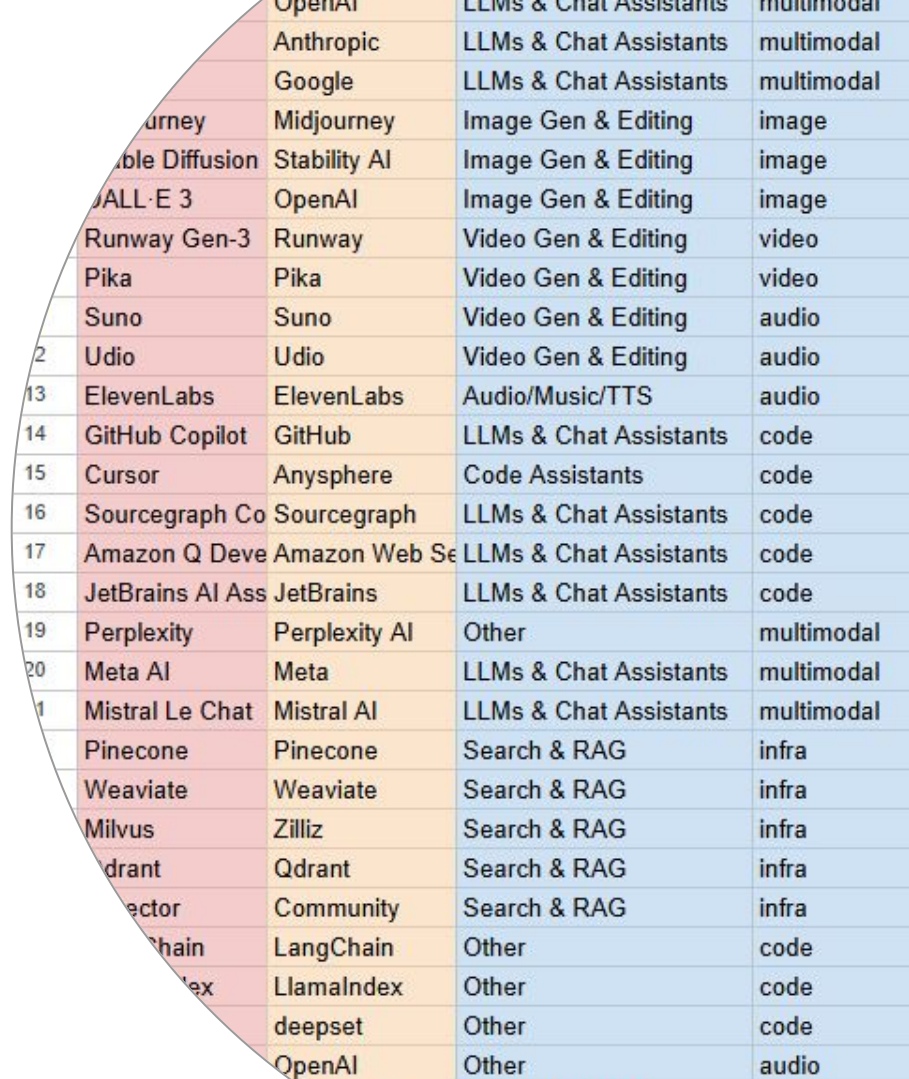
Conclusion

Conclusion et bonus



01 - Introduction

Présentation du dataset



	OpenAI	LLMs & Chat Assistants	multimodal
	Anthropic	LLMs & Chat Assistants	multimodal
	Google	LLMs & Chat Assistants	multimodal
	Midjourney	Image Gen & Editing	image
	Stability AI	Image Gen & Editing	image
	OpenAI	Image Gen & Editing	image
	Runway	Video Gen & Editing	video
	Pika	Video Gen & Editing	video
	Suno	Video Gen & Editing	audio
2	Udio	Video Gen & Editing	audio
13	ElevenLabs	Audio/Music/TTS	audio
14	GitHub Copilot	LLMs & Chat Assistants	code
15	Cursor	Code Assistants	code
16	Sourcegraph	LLMs & Chat Assistants	code
17	Amazon Q Developer	LLMs & Chat Assistants	code
18	JetBrains AI Assistant	LLMs & Chat Assistants	code
19	Perplexity	Other	multimodal
20	Meta AI	LLMs & Chat Assistants	multimodal
1	Mistral Le Chat	LLMs & Chat Assistants	multimodal
	Pinecone	Search & RAG	infra
	Weaviate	Search & RAG	infra
	Milvus	Search & RAG	infra
	Qdrant	Search & RAG	infra
	Community	Search & RAG	infra
	LangChain	Other	code
	LlamaIndex	Other	code
	deepset	Other	code
	OpenAI	Other	audio

45%

des modèles sont sortis en **2023**

67%

des IA gen ne sont **PAS** open-source

27%

sont des **LLMS**



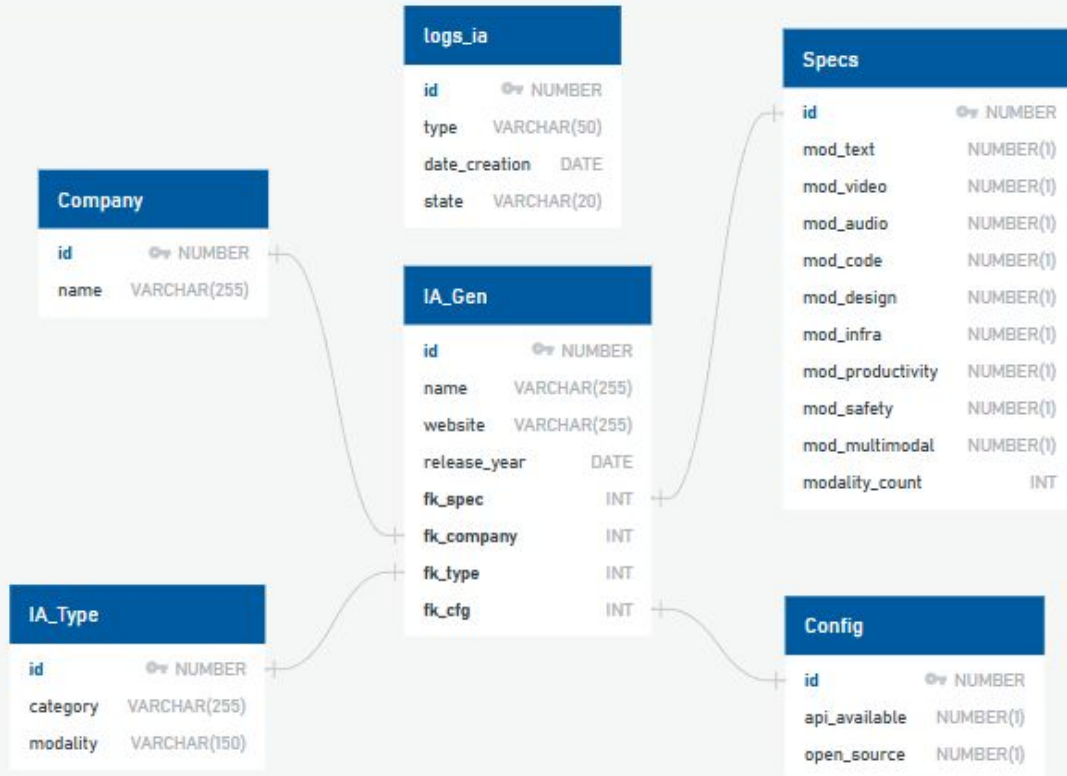
A propos du dataset

115 lignes au **total**

0% de **mismatched** ou **missing**

22 colonnes





02 - Modélisation

```
CREATE TABLE IA_Gen (
    id NUMBER DEFAULT
iagen_seq.nextval PRIMARY KEY,
    name VARCHAR(255) NOT
NULL,
    website VARCHAR(255),
    release_year DATE,
    fk_specs INT,
    fk_company INT,
    fk_iatype INT,
    fk_cfg INT,
    FOREIGN KEY (fk_specs)
REFERENCES Specs(id),
    FOREIGN KEY (fk_company)
REFERENCES Company(id),
    FOREIGN KEY (fk_iatype)
REFERENCES IA_Type(id),
    FOREIGN KEY (fk_cfg)
REFERENCES Config(id));
COMMIT;
```

```
CREATE TABLE Specs (
    id NUMBER DEFAULT
specs_seq.nextval PRIMARY KEY,
    mod_text NUMBER(1),
    mod_image NUMBER(1),
    mod_video NUMBER(1),
    mod_audio NUMBER(1),
    mod_code NUMBER(1),
    mod_design NUMBER(1),
    mod_infra NUMBER(1),
    mod_productivity NUMBER(1),
    mod_safety NUMBER(1),
    mod_multimodal NUMBER(1),
    modality_count INT);
```

```
CREATE TABLE logs_ia (
    id NUMBER DEFAULT
logs_seq.nextval PRIMARY KEY,
    type VARCHAR(50) NOT NULL,
    date_creation DATE NOT
NULL,
    state VARCHAR(255));
```

```
CREATE TABLE Company (
    id NUMBER DEFAULT
company_seq.nextval PRIMARY KEY,
    name VARCHAR(255) NOT
NULL);
```

```
CREATE TABLE IA_Type (
    id NUMBER DEFAULT
iatype_seq.nextval PRIMARY KEY,
    category VARCHAR(255) NOT
NULL,
    modality VARCHAR(150));
```

```
CREATE TABLE Config (
    id NUMBER DEFAULT
cfg_seq.nextval PRIMARY KEY,
    api_available NUMBER(1),
    open_source NUMBER(1));
```

SQL Create table



Contraintes SQL



Not Null

Name, category...

Foreign key

fk_specs, fk_company...

Primary key

id





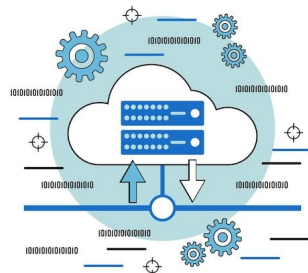
03 - Contexte

Environnement

Installation d'Oracle

```
Mathieu@DESKTOP-KK11NT3 MSYS ~  
$ scp chemin/local/LINUX.X64_193000_db_home.zip oracle@ip:/data/engines/oracle
```

```
prodigieux@eteris-cloud-provider:/tmp/docker-images/OracleDatabase/SingleInstance/dockerfiles$ sudo docker run --name oracle-advdb \  
-e ORACLE_PWD="oracleadvdb" \  
-e ORACLE_EDITION="EE" \  
-e ORACLE_SID="ADVDB" \  
-p 1521:1521 \  
-v /data/projects/advdb/db/oracle:/opt/oracle/oradata \  
-d oracle/database:19.3.0-ee  
26b972f6a8cd1c6febfa9bccaa0aa161a03f293c104e4217ae5c3dde4cbdb82e
```



Cloud server

04 - Sécurité

Gestion des users, privilèges



Création des users

```
SQL> CREATE USER admin_orclpdb1 IDENTIFIED BY "2F:@?RipG7E)49vp";  
User created.  
SQL> █
```

```
SQL> CREATE USER u_rw_orclpdb1 IDENTIFIED BY "n4g$~8xnLF86C~G:";  
User created.
```

```
SQL> CREATE USER u_r_orclpdb1 IDENTIFIED BY ",-pe,z2P!D9R4m6Z";  
User created.
```

NOMBRE DE CARACTÈRES	UNIQUEMENT DES CHIFFRES	LETTRES MINUSCULES	LETTRES MINUSCULES ET MAJUSCULES	LETTRES MINUSCULES ET MAJUSCULES + CHIFFRES	LETTRES MINUSCULES ET MAJUSCULES + CHIFFRES + CARACTÈRES SPECIAUX
4	IMMÉDIATEMENT	IMMÉDIATEMENT	IMMÉDIATEMENT	IMMÉDIATEMENT	IMMÉDIATEMENT
6	IMMÉDIATEMENT	IMMÉDIATEMENT	IMMÉDIATEMENT	1 sec	5 sec
8	IMMÉDIATEMENT	5 sec	22 min	1 heure	9 heures
10	IMMÉDIATEMENT	58 min	1 mois	7 mois	5 ans
12	45 sec	3 semaines	300 ans	2000 ans	34 000 ans
14	41 min	51 ans	800 000 ans	9 millions d'années	200 millions d'années

*source : SCSP Community (Seasoned Cyber Security Professionals)

```
export ADM_ORCLPDB1='admin_orclpdb1'  
export ADM_PASSWD='2F:@?RipG7E)49vp'  
export URW_ORCLPDB1='u_rw_orclpdb1'  
export URW_PASSWD='n4g$~8xnLF86C~G:'  
export UR_ORCLPDB1='u_r_orclpdb1'  
export UR_PASSWD=',-pe,z2P!D9R4m6Z'
```



Profiles, rôles & privilèges

```
SQL> CREATE PROFILE urw_profile LIMIT connect_time 5 failed_login_attempts 5 password_life_time 31;
```

Profile created.

```
SQL> █
```

```
SQL> CREATE PROFILE ur_profile LIMIT connect_time 5 failed_login_attempts 3;
```

Profile created.

```
SQL> █
```



Profiles, rôles & privilèges

```
SQL> GRANT ALL PRIVILEGES TO admin_orclpdb1;  
Grant succeeded.
```

```
SQL> CREATE ROLE rw;
```

Role created.

```
SQL> CREATE ROLE r;
```

Role created.

```
SQL> █
```

```
SQL> GRANT SELECT, INSERT, UPDATE, DELETE ON Company TO rw;  
GRANT SELECT, INSERT, UPDATE, DELETE ON logs_ia TO rw;  
GRANT SELECT, INSERT, UPDATE, DELETE ON IA_Type TO rw;  
GRANT SELECT, INSERT, UPDATE, DELETE ON Specs TO rw;  
GRANT SELECT, INSERT, UPDATE, DELETE ON Config TO rw;  
GRANT SELECT, INSERT, UPDATE, DELETE ON IA_Gen TO rw;
```

```
SQL> GRANT rw TO u_rw_orclpdb1;
```

Grant succeeded.

```
SQL> GRANT r TO u_r_orclpdb1;
```

Grant succeeded.

```
SQL> █
```

```
SQL>  
Grant succeeded.
```

```
SQL>  
Grant succeeded.
```

```
SQL>  
Grant succeeded.
```

```
SQL>  
Grant succeeded.
```

```
SQL>  
Grant succeeded.
```

```
SQL> SQL> SQL> █
```

05 - Optimisation

Requêtes, index, vues



Optimisation par **cache**

```
CREATE SEQUENCE logs_seq START WITH 1 INCREMENT BY 1 CACHE 20;  
CREATE SEQUENCE company_seq START WITH 1 INCREMENT BY 1 CACHE 20;  
CREATE SEQUENCE iatype_seq START WITH 1 INCREMENT BY 1 CACHE 20;  
CREATE SEQUENCE cfg_seq START WITH 1 INCREMENT BY 1 CACHE 20;  
CREATE SEQUENCE specs_seq START WITH 1 INCREMENT BY 1 CACHE 20;  
CREATE SEQUENCE iagen_seq START WITH 1 INCREMENT BY 1 CACHE 20;
```

Optimisation des séquence avec pré-allocation des valeurs



Optimisation par Index

1ère requête - Les IA qui sont a la fois disponibles via API & Open-source :

```
SELECT g.name, g.release_year
FROM IA_Gen g
JOIN Config cfg ON g.fk_cfg = cfg.id
WHERE cfg.api_available=1 AND cfg.open_source=1;
```

```
CREATE INDEX idx_cfg_api_open
ON Config (api_available, open_source, id);
```

Résultat :

Coût : 4

Coût après l'optimisation : 2

```
rdrails

Guardrails AI ( 01-DEC-23
framework)

LMQL          01-DEC-23
DeepSeek R1   01-DEC-25
Mistral Large 01-DEC-23

NAME          RELEASE_Y
-----
Llama 3.1     01-DEC-23
Qwen 2.5      01-DEC-25
Gemini 2      01-DEC-24
DBRX          01-DEC-24
Phi-3         01-DEC-24
MPT-7B        01-DEC-23
Mixtral 8x7B  01-DEC-23
Mistral Codestr
al            01-DEC-23

Code Llama    01-DEC-23

NAME          RELEASE_Y
-----
Llama Guard   01-DEC-23
Transformers  01-DEC-18
Diffusers     01-DEC-22
Gradio        01-DEC-19
Hugging Face Sp
aces          01-DEC-21

ComfyUI       01-DEC-23
Black Forest La
bs FLUX       01-DEC-24

NAME          RELEASE_Y
-----
Coqui TTS     01-DEC-21
TruLens       01-DEC-23
Ollama        01-DEC-23
WhisperX      01-DEC-22
Silero VAD    01-DEC-20
Papers with Cod
e SOTA        01-DEC-18

Kaggle Models 01-DEC-23
OpenHands     01-DEC-24
LlamaGuard 2  01-DEC-23

41 rows selected.

SQL>
```

Optimisation par Index

2 ème requête - Nombre d'IA par entreprise :

```
SELECT c.name AS company,  
       COUNT(g.id) AS ia_count  
FROM Company c  
LEFT JOIN IA_Gen g ON g.fk_company = c.id  
GROUP BY c.name  
ORDER BY ia_count DESC;  
  
CREATE INDEX idx_cfg_api_open  
ON Config (api_available, open_source, id);
```

Résultat :

Coût : 5

Coût après l'optimisation : 2

COMPANY	

IA_COUNT	

OpenAI	9
Meta	5
Mistral AI	4
COMPANY	

IA_COUNT	

Hugging Face	4
Community	3
Anthropic	3
COMPANY	

IA_COUNT	

Adobe	3
Google	2
xAI	2



Optimisation par Index

3ème requête - IA sorties après la moyenne des années de sorties :

```
SELECT name,  
       EXTRACT(YEAR FROM release_year) AS release_year  
FROM IA_Gen  
WHERE EXTRACT(YEAR FROM release_year) >  
      (SELECT AVG(EXTRACT(YEAR FROM release_year)) FROM IA_Gen);
```

```
CREATE INDEX idx_iagen_release_year  
ON IA_Gen (EXTRACT(YEAR FROM release_year));
```

Résultat :

Coût : 4

Coût après l'optimisation : 2

NAME	RELEASE_YEAR
Claude	2023
Gemini	2023
DALL?E 3	2023
Pika	2023
Suno	2023
Cursor	2023
Sourcegraph Cod y	2023
Amazon Q Develo per	2023
NAME	RELEASE_YEAR
JetBrains AI As sistant	2023
pgvector	2023
Deepgram	2023
NVIDIA NeMo Gua rdrails	2023
Lakera Guard	2023
Guardrails AI (2023
NAME	RELEASE_YEAR
framework)	
LMQL	2023
OpenRouter	2023
Vercel v0	2023
Framer AI	2023
Notion AI	2023
xAI Grok	2023
Mistral Large	2023
Llama 3.1	2023
MPT-7B	2023
NAME	RELEASE_YEAR
Mixtral 8x7B	2023
GPT-4o	2023
Claude 3.7 Sonn et	2023
Mistral Codestr al	2023
Code Llama	2023

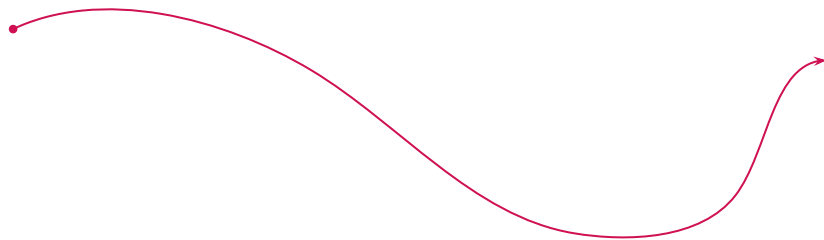
Optimisation par Index

4ème requête - Nombre d'IA par catégorie fonctionnelle :

```
SELECT t.category,  
       COUNT(g.id) AS total_ia  
FROM IA_Type t  
JOIN IA_Gen g ON g.fk_iatype = t.id  
GROUP BY t.category  
ORDER BY total_ia DESC;
```

```
CREATE INDEX idx_iagen_iatype ON IA_Gen(fk_iatype);
```

Résultat :



Coût : 5

Coût après l'optimisation : 3

CATEGORY
TOTAL_IA
LLMs & Chat Assistants
32
Other
27
Video Gen & Editing
13
CATEGORY
TOTAL_IA
Image Gen & Editing
12
Search & RAG
6
Audio/Music/TTS
6
CATEGORY
TOTAL_IA
Safety & Guardrails
5
Design & UI
4
Infra & Inference
3
CATEGORY
TOTAL_IA
Evaluation & Benchmarks
2
Speech-to-Text (ASR)



Optimisation par Index

5ème requête - IA multimodales avec leur entreprise et leur types :

```
SELECT g.name AS ia_name,  
       c.name AS company,  
       t.category,  
       t.modality  
FROM IA_Gen g  
JOIN Company c ON g.fk_company = c.id  
JOIN IA_Type t ON g.fk_iatype = t.id  
JOIN Specs s ON g.fk_specs = s.id  
WHERE s.mod_multimodal = 1;
```

```
CREATE INDEX idx_specs_multimodal ON Specs(mod_multimodal);
```

Résultat :

Coût : 8

Coût après l'optimisation : 7

IA_NAME
COMPANY
CATEGORY
MODALITY
ChatGPT
OpenAI
LLMs & Chat Assistants
multimodal
IA_NAME
COMPANY
CATEGORY
MODALITY
OpenAI o3-mini
OpenAI
LLMs & Chat Assistants
multimodal
IA_NAME
COMPANY
CATEGORY
MODALITY
GPT-4o
OpenAI
LLMs & Chat Assistants
multimodal
IA_NAME
COMPANY
CATEGORY
MODALITY

Optimisation par vues

6ème requête - Classement des entreprises selon le nombre d'IA

```
SELECT company, ia_count, RANK() OVER (ORDER BY ia_count DESC) AS rank_company
FROM (
    SELECT c.name AS company, COUNT(g.id) AS ia_count
    FROM Company c
    LEFT JOIN IA_Gen g ON g.fk_company = c.id GROUP BY c.name);
```

```
CREATE MATERIALIZED VIEW mv_company_rank AS
SELECT c.name AS company, COUNT(g.id) AS ia_count FROM Company c
LEFT JOIN IA_Gen g ON g.fk_company = c.id GROUP BY c.name;
```

```
SELECT company, ia_count, RANK() OVER (ORDER BY ia_count DESC) AS rank_company
FROM mv_company_rank;
```

Résultat sur la slide suivante

Coût : 6

Coût après l'optimisation : 3



Optimisation par vues

COMPANY		

IA_COUNT	RANK_COMPANY	

OpenAI	9	1
Meta	5	2
Mistral AI	4	3
COMPANY		

IA_COUNT	RANK_COMPANY	

Hugging Face	4	3
Community	3	5
Anthropic	3	5
COMPANY		

IA_COUNT	RANK_COMPANY	

Adobe	3	5
Google	2	8
xAI	2	8
COMPANY		

IA_COUNT	RANK_COMPANY	

Canva	2	8
Amazon Web Services	2	8
Google Cloud		





06 - Conclusion

Remerciements & Bonus

Bonus



```
SshTunnelConfig.java

@Configuration
@ConditionalOnProperty(name = "ssh.tunnel.enabled", havingValue = "true")
public class SshTunnelConfig {

    private static final Logger logger = LoggerFactory.getLogger(SshTunnelConfig.class);

    @Value("${ssh.tunnel.host}")
    private String sshHost;

    @Value("${ssh.tunnel.port}")
    private int sshPort;

    @Value("${ssh.tunnel.username}")
    private String sshUsername;

    @Value("${ssh.tunnel.password}")
    private String sshPassword;

    @Value("${ssh.tunnel.local-port}")
    private int localPort;

    @Value("${ssh.tunnel.remote-host}")
    private String remoteHost;

    @Value("${ssh.tunnel.remote-port}")
    private int remotePort;

    private Session session;
```

```
DbConnection.java

@Configuration
public class DbConnection {

    @Value("${oracle.db.url}")
    private String dbUrl;

    @Value("${oracle.db.username}")
    private String username;

    @Value("${oracle.db.password}")
    private String passwd;

    @Value("${oracle.db.driver}")
    private String driver;

    @Bean
    @Primary
    public DataSource dataSource(){
        return DataSourceBuilder.create()
            .url(dbUrl)
            .username(username)
            .password(passwd)
            .driverClassName(driver)
            .build();
    }
}
```

Demo



Merci de nous avoir écoutés !

Liens :

<https://www.kaggle.com/datasets/wardabilal/global-generative-ai-tools-landscape-2025>

<https://github.com/MathieuAudibert/advdb>

https://docs.google.com/document/d/14W2J3o2_DF9VccvgDo429PwX9op7GCDTn-Hi6Aufqh0/edit?tab=t.0

BIGAND Théo, AGOSTINO Roméo & AUDIBERT Mathieu

