

ORACLE

Oracle Autonomous Databaseを 使ったAI活用入門

Oracle Database Technology Night #85

山川 薫 / 藤本 万里奈

日本オラクル株式会社

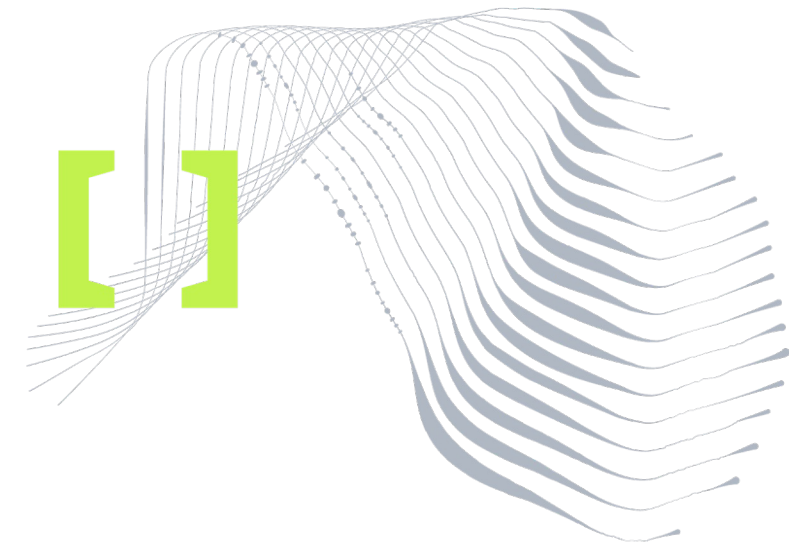
2024年12月19日

*

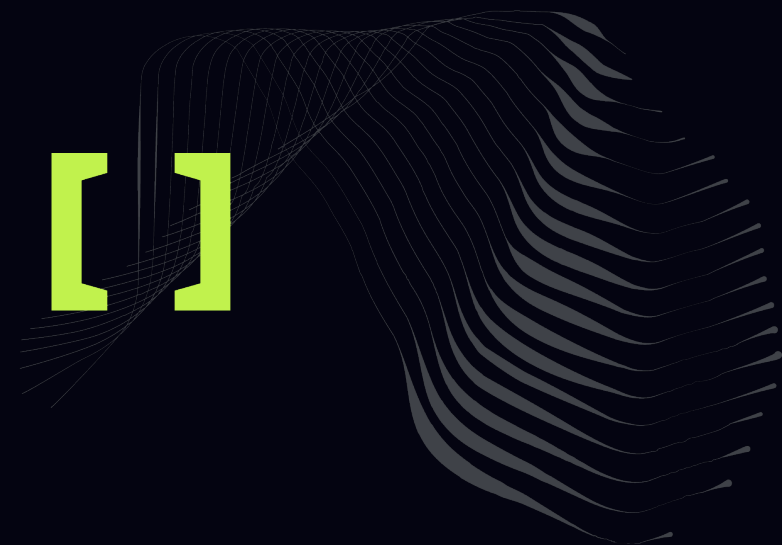


Agenda

1. Autonomous Database 概要
2. Autonomous DatabaseにおけるAIの活用
3. Oracle APEXにおけるAIの活用



Autonomous Database概要

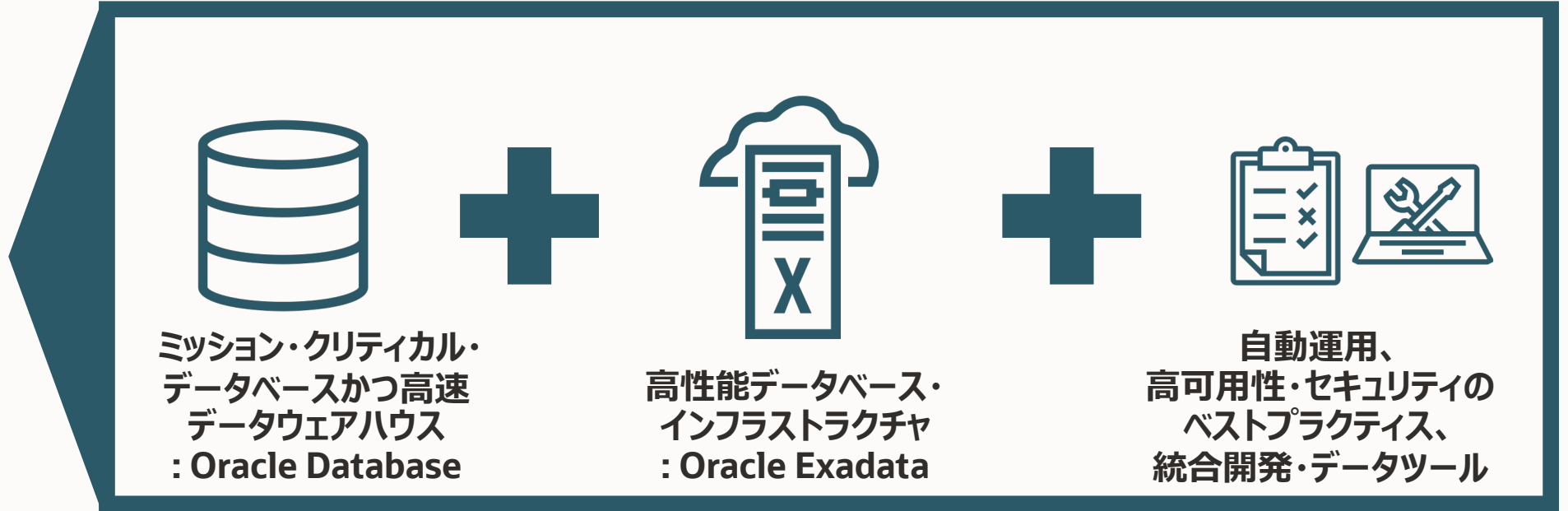


Autonomous Database (ADB) とは？

クラウドスケールで拡張可能なクラウドネイティブ・データベース



Autonomous Database



低コスト

データベースの管理と運用の自動化

最新の開発プラットフォーム

最新のデータ分析プラットフォーム

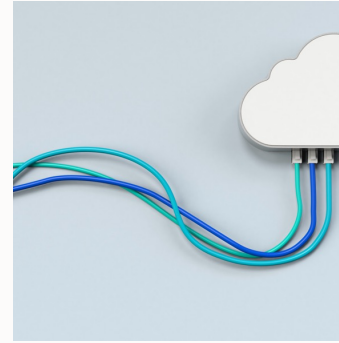
Autonomous Databaseの主な特徴

クラウドスケールで拡張可能なクラウドネイティブ・データベース



低コスト

- CPUサイズをオンラインで自由に変更、さらに自動スケール機能を提供
- クラウドスケールのシステム監視、約7/8の問題を自動解決



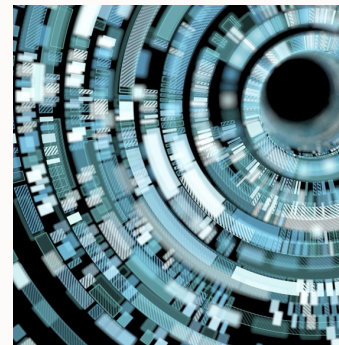
データベースの管理と運用の自動化

- セキュリティパッチを含むパッチの自動適用で安定運用に寄与
- テストの自動化
- 組み込みの高可用性とセキュリティ
- 使いやすいクローニング



最新の開発プラットフォーム

- AIを活用した組み込みローコード開発
- JSONとMongoDB API
- クラウドファンクションの呼び出し
- 位置情報、グラフなどの非構造データの扱いと専用開発ツールの提供



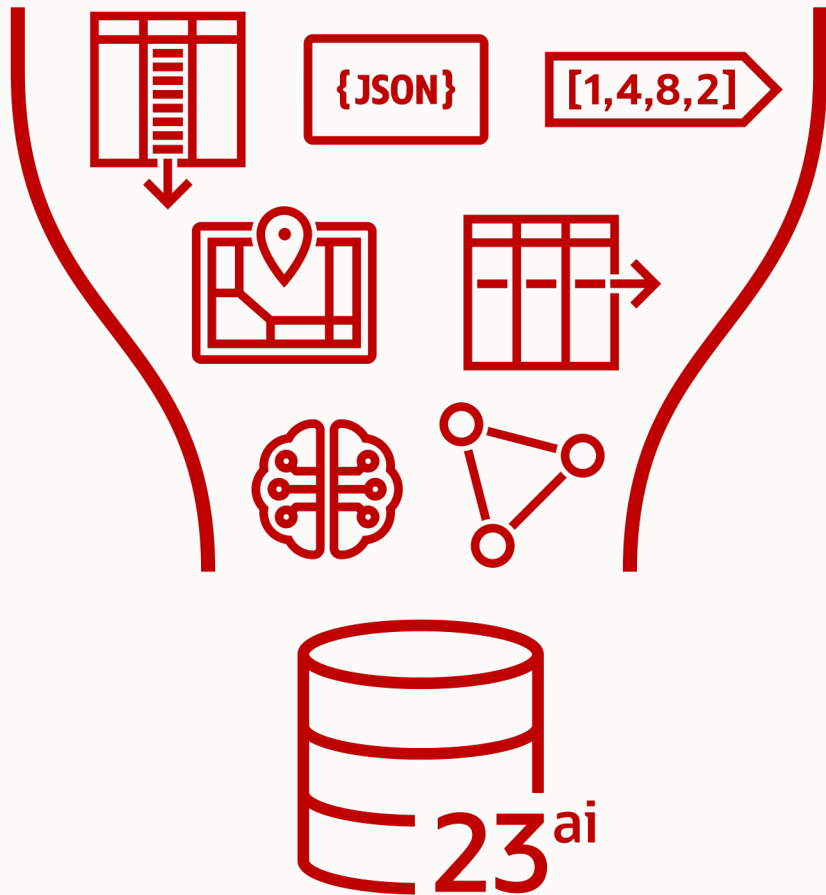
最新のデータ分析プラットフォーム

- マルチクラウド環境でのストレージやDBへのアクセス
- データロード・ETLツールを内蔵
- データ共有とクラウドリンク
- 機械学習、AutoMLを内蔵
- Select AIとベクトル検索 : AIで自然言語DB検索



Autonomous Databaseで取り扱い可能なデータと開発手法

様々な種類のデータとワークロードをシングル・データ・プラットフォームで



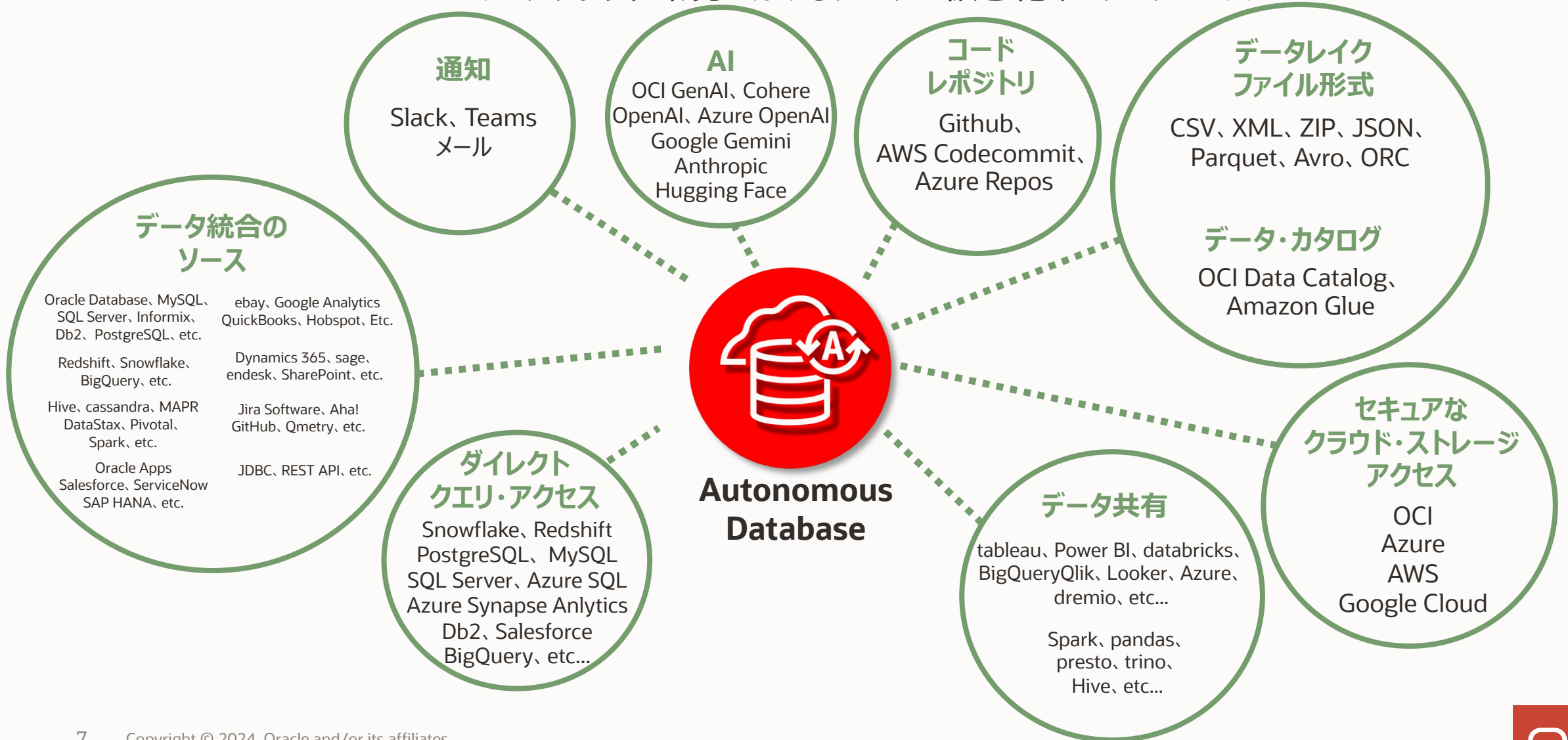
Relationalでの複合ワークロード、JSON、Cube、Graph、Spatial、Text、Event、Blockchainなどに対応

Relational	Graph	Documents	Key-Value
Vector	Blockchain	Geospatial	Cube

データの種類が異なっても1つのクエリーで処理、複数のDBを使い分ける必要がなく、システム運用・セキュリティ等の標準化が可能

Autonomous Databaseのマルチクラウド・エコシステム

Autonomous Databaseはマルチクラウド環境におけるデータの仮想化インターフェース



最新のデータ分析のための組み込みツール

インストール不要で数分でデプロイ可能



REST Services



データ・モデリング
Data Modeling



ロー・コード APEX
Low-code APEX



SQL ワークシート
SQL Worksheet



Notebooks



JSON ワークシート
JSON Worksheet



データ・セーフ
Data Safe



Database Vault



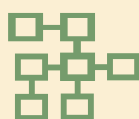
データ・カタログ
Data Catalog



データ共有
Data Sharing



データ統合
Data Integration



セマンティック・モデリング
Semantic Modeling



データ分析
Data Analysis



MLモデリング
ML Modeling



グラフ・モデリング
Graph Modeling



Spatial

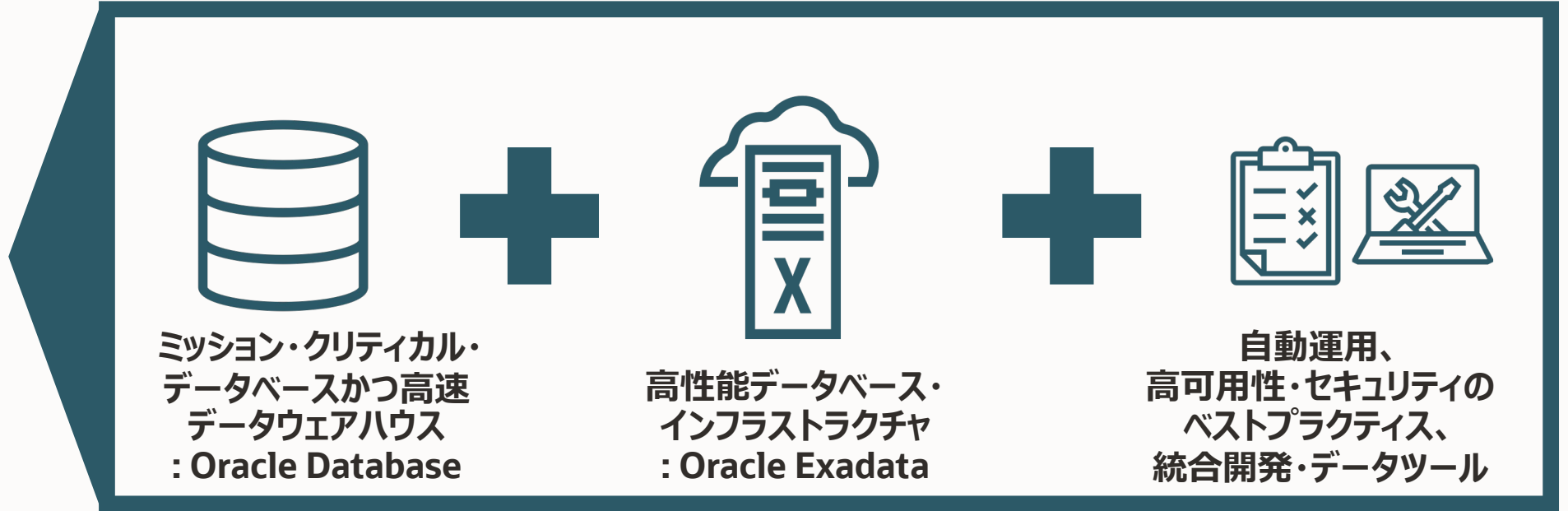


Autonomous Database (ADB) とは？ (再掲)

クラウドスケールで拡張可能なクラウドネイティブ・データベース



Autonomous Database



低コスト

データベースの管理と運用の自動化

最新の開発プラットフォーム

最新のデータ分析プラットフォーム



Autonomous DatabaseにおけるAIの活用



Autonomous DatabaseにおけるAIの活用

生成AIを活用した自然言語プロンプトによるDBデータ検索、ベクトル検索、合成データ生成

- **Select AI (for Natural Language Interaction)** **19c** **23ai**
 - 大規模言語モデル（LLM）を利用して、自然言語プロンプトからSQLを生成、実行
- **Select AI with Retrieval Augmented Generation (RAG)** **23ai**
 - クラウド・ストレージ上の企業データ（ファイル）からベクトル・ストアおよびベクトル索引を作成するプロセスを自動化
 - 入力された自然言語プロンプトからベクトル埋め込みを生成し、ベクトル索引を使用してベクトル検索を実行、ベクトル検索の実行結果を使用して入力された自然言語プロンプトを補強し、LLMに問合せを実行
- **Synthetic Data Generation(SDG)** **19c** **23ai**
 - 大規模言語モデル（LLM）を使用して、入力された自然言語プロンプトを考慮した、より適切でスキーマに特化した合成データを生成

Select AI (for Natural Language Interaction)

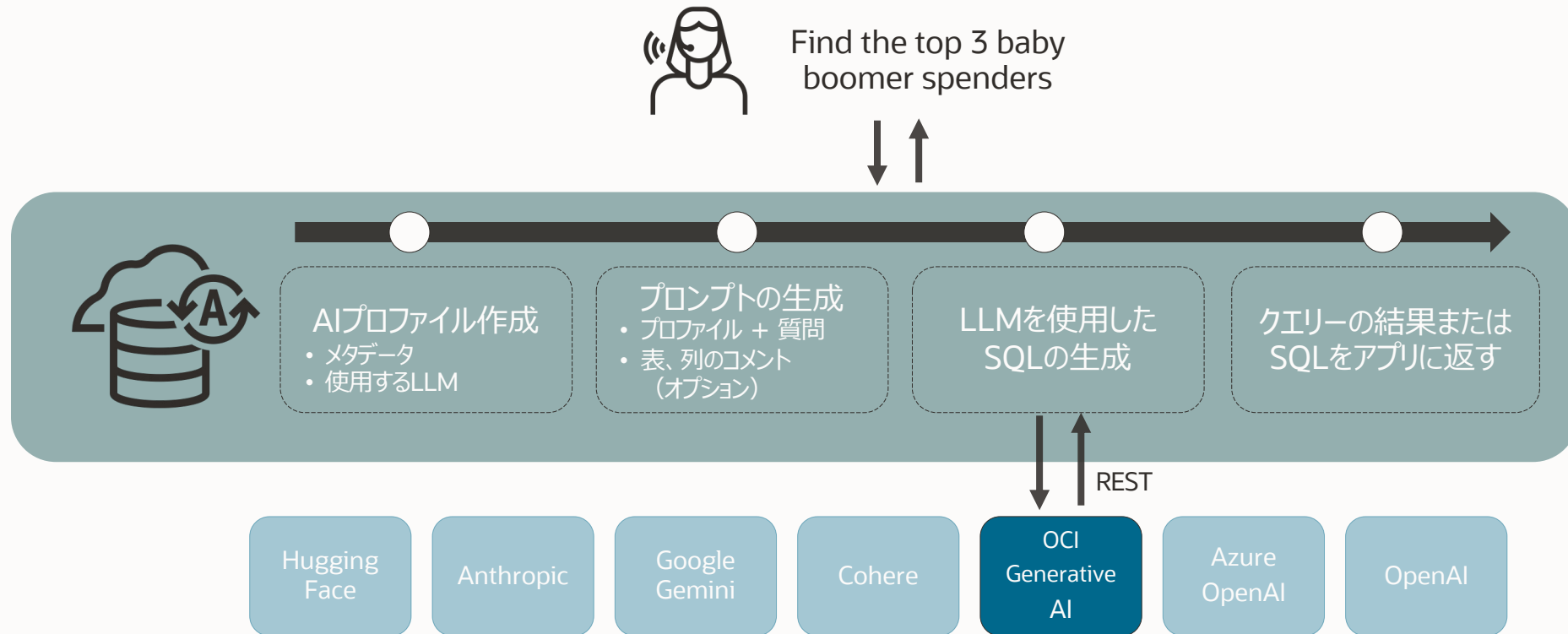
Select AI (for Natural Language Interaction)

Select AIとは

- ✓ LLMを利用して、自然言語プロンプトからSQLを生成、実行
- ✓ データベースに対する自然言語を使用した問合せ
- ✓ 様々なAIプロバイダをサポート
 - ✓ OCI Generative AI Service
 - ✓ OpenAI
 - ✓ Cohere
 - ✓ Azure OpenAI Service
 - ✓ Google Gemini
 - ✓ Anthropic
 - ✓ Hugging Face

Select AI : SQL生成のプロセスフロー

大規模言語モデル（LLM）を利用して、自然言語プロンプトからSQLを生成、実行



Select AI : AIプロファイルの作成

DBMS_CLOUD_AI.CREATE_PROFILEプロセス

```
BEGIN
  DBMS_CLOUD_AI.CREATE_PROFILE(
    profile_name => 'OPENAI_GPT4', -- 作成するAIプロファイル名
    attributes   => '{"provider": "openai", -- AIプロバイダ
                    "model": "gpt-4", -- モデル
                    "credential_name": "OPENAI_CRED", -- モデルへのアクセス時に使用するクレデンシャル
                    "object_list": [{"owner": "MOVIESTREAM", "name": "movies"}, -- 使用するオブジェクト
                                    {"owner": "MOVIESTREAM", "name": "streams"},
                                    {"owner": "MOVIESTREAM", "name": "v_customer"},
                                    {"owner": "MOVIESTREAM", "name": "genre"},
                                    {"owner": "MOVIESTREAM", "name": "actors"}]
                    }'
  );
END;
/
```

Select AI : AIプロファイルの作成

DBMS_CLOUD_AI.CREATE_PROFILEプロシージャの主なオプション (Attribute)

属性 (Attribute)	説明
provider	使用するAIプロバイダ
model	使用するLLMのモデル
credential_name	使用するクレデンシャル名
object_list	使用するスキーマ・オブジェクトのリスト
comments	LLMに送信するメタデータにコメントを含める (TRUE/FALSE)
conversation	会話履歴を有効化する (TRUE/FALSE)
max_tokens	使用する最大トークンの数
stop_tokens	ストップ・トークン (ストップ・シーケンス) をJSON配列で指定
temperature	生成モデル使用時の温度 (temperature) を指定
azure_resource_name	Azure OpenAI Serviceのリソース名
azure_deployment_name	Azure OpenAI Serviceのデプロイ名



Select AI : AIプロファイルの作成

使用可能なAIプロバイダとLLM (1)

- OCI Generative AI Service (oci)
 - **meta.llama-3.1-70b-instruct**
 - meta.llama-3.1-405b-instruct
 - meta.llama-3.2-90b-vision-instruct
 - cohere.command-r-16k
 - cohere.command-r-plus
- Cohere (cohere)
 - **command**
 - command-nightly (experimental)
 - command-r
 - command-r-plus
 - command-light
 - command-light-nightly (experimental)
 - カスタムモデル
- OpenAI (openai)
 - **gpt-3.5-turbo**
 - gpt-4o
 - gpt-4o-mini
 - gpt-4
 - gpt-4-0613
 - gpt-4-32k
 - gpt-4-32k-0613
 - gpt-3.5-turbo-0613
 - gpt-3.5-turbo-16k
 - gpt-3.5-turbo-16k-0613
- Azure OpenAI Service (azure)
 - GPT-4o
 - GPT-4
 - GPT-4 Turbo with Vision
 - GPT-3.5-Turbo

Select AI : AIプロファイルの作成

使用可能なAIプロバイダとLLM (2)

- Google (google)
 - **gemini-1.5-flash**
 - gemini-1.5-pro
 - gemini-1.0-pro
- Anthropic (anthropic)
 - **claude-3-5-sonnet-20240620**
 - claude-3-opus-20240229
 - claude-3-sonnet-20240229
- Hugging Face (huggingface)
 - **Mixtral-8x7B-Instruct-v0.1**
 - Meta-Llama-3-70B-Instruct
 - Qwen1.5-1.8B
 - その他のチャットモデル



OCI Generative AI Service

利用可能なリージョンと事前トレーニング済みモデル（専用AIクラスタ不要モデル）

リージョン	リージョン識別子	Chat Model	Embedding Model
Brazil East (Sao Paulo)	sa-saopaulo-1	cohere.command-r-08-2024 cohere.command-r-plus-08-2024 meta.llama-3.1-70b-instruct meta.llama-3.2-90b-vision-instruct	cohere.embed-english-v3.0 cohere.embed-multilingual-v3.0
Germany Central (Frankfurt)	eu-frankfurt-1	cohere.command-r-08-2024 cohere.command-r-plus-08-2024 meta.llama-3.1-70b-instruct	cohere.embed-english-v3.0 cohere.embed-multilingual-v3.0
Japan Central (Osaka)	ap-osaka-1	cohere.command-r-08-2024 cohere.command-r-plus-08-2024 meta.llama-3.1-70b-instruct meta.llama-3.2-90b-vision-instruct	cohere.embed-english-v3.0 cohere.embed-multilingual-v3.0
UK South (London)	uk-london-1	cohere.command-r-08-2024 cohere.command-r-plus-08-2024 meta.llama-3.1-70b-instruct	cohere.embed-english-v3.0 cohere.embed-multilingual-v3.0
US Midwest (Chicago)	us-chicago-1	cohere.command-r-08-2024 cohere.command-r-plus-08-2024 meta.llama-3.1-70b-instruct meta.llama-3.2-90b-vision-instruct meta.llama-3.1-405b-instruct	cohere.embed-english-v3.0 cohere.embed-multilingual-v3.0 cohere.embed-english-light-v3.0 cohere.embed-multilingual-light-v3.0



Select AI利用時の注意点

LLMのハルシネーション

LLMがSQLを生成する際にはハルシネーションが発生しうるため、以下のようなケースが発生する可能性がある

- SELECT AI実行時に、生成されたSQL問合せを特定の自然言語プロンプトに対して実行できない
- SELECT AI実行時に、特定の自然言語プロンプトに対してSQL問合せを生成できない
- SELECT AI実行時に、自然言語プロンプトに対して正しい結果を生成するSQL問合せがLLMによって生成されない

例えば、

- スキーマ・オブジェクトのメタデータ（表名、列名）が英語表記の環境で、日本語でプロンプトを入力した場合
- 列データが実際の名称や値ではなくコード化されていて、実際の名称、値はマスタ表に格納されている場合

対応策：

- 表および列にコメントを設定し、AIプロファイルでコメントの送信を有効にする（"comments":"TRUE"を設定）
 - COMMENT ON TABLE、COMMENT ON COLUMN
- ビューを作成して、列値に実際の名称や値が格納されるようにする

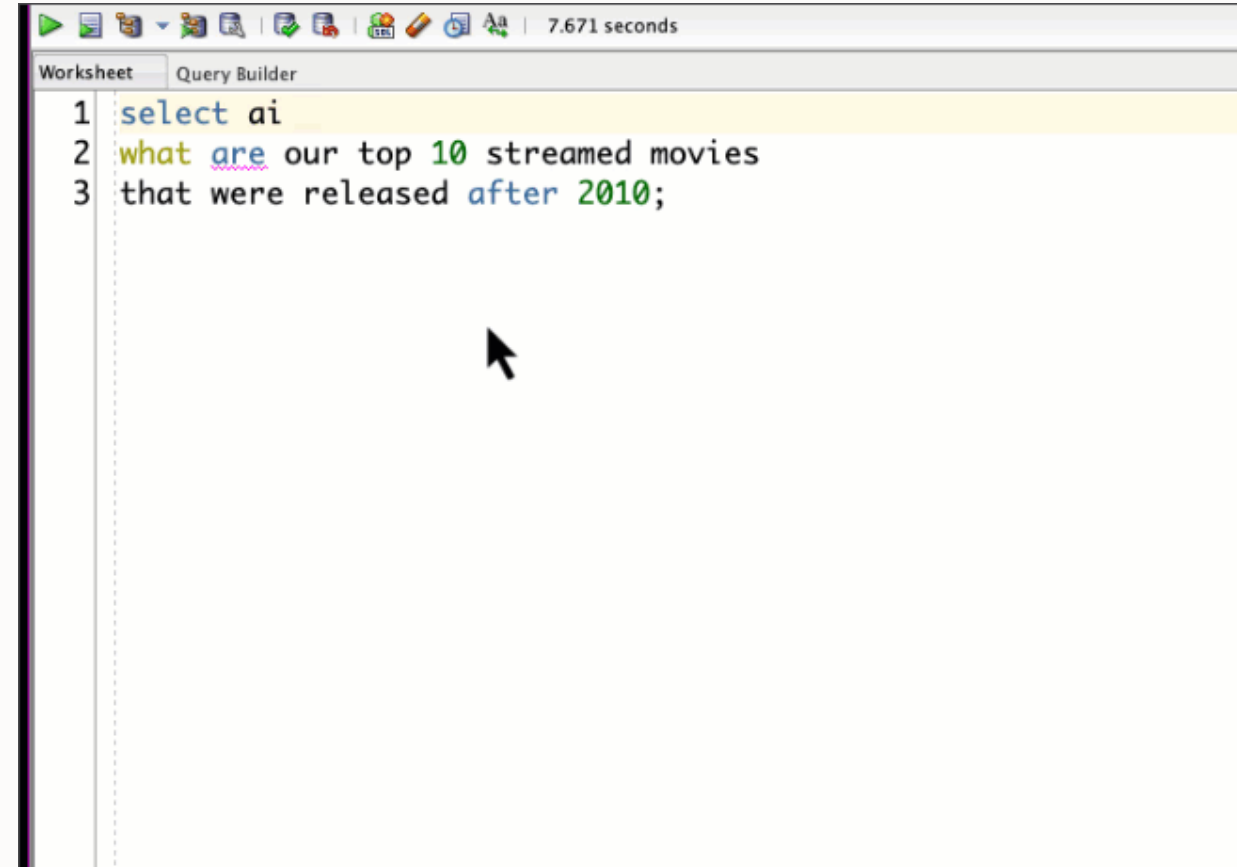
Select AI : 実行例

新しい自然言語アプリケーションの拡張や構築が容易

標準的なSELECT文を使用し、その後にAIというキーワードと自然言語による質問を続ける

オプションの ACTION キーワード:

runsql	生成されたSQLを実行し、結果セットを返す (指定しない時のデフォルト)
narrate	生成されたSQLの実行結果をLLMに送信し、 自然言語で結果を返す
showsql	生成されたSQLを返す
explainsql	生成されたSQLの説明を自然言語で返す
chat	一般的なAIチャットを実行する (DBデータは利用しない)



Select AI "Conversation"

会話履歴を使用

- AIプロファイルのattributeに「"conversation": "true"」を追加することで、会話履歴を有効化可能
- AIプロバイダにOCI Generative AI Service、OpenAIおよびAzure OpenAI Serviceを指定した場合に利用可能

```
-- データベースのデータと会話をする
--
-- 初めに質問をし、それを基に会話を進める
select ai ストリーミング回数の合計値は何ですか?;
--
-- 詳細を質問
select ai その結果を、ジャンル別に分けて表示して下さい。;
--
-- さらに詳細に質問
select ai その結果に、それぞれのジャンルをよく視聴する顧客の顧客セグメントも追加して下さい。;
--
-- 結果を絞り込む
select ai
総視聴回数で上位5位までの顧客セグメントとジャンルを表示して下さい。
ランキングも結果に表示して下さい。;
```



DBMS_CLOUD_AI.GENERATEファンクション

Select AIをステートレスで実行

- SELECT AI文はDatabase Actions、APEXではサポートされない
- 代わりにDBMS_CLOUD_AI.GENERATEファンクションを使用
- サポートされるアクションはshowsql（デフォルト） /explainsql/narrate/chatのみでrunsqlはサポートされない

```
SQL> SELECT
  2     DBMS_CLOUD_AI.GENERATE( prompt => 'how many customers', -- 自然言語プロンプト
  3                             profile_name => 'GENAI', -- 使用するAIプロファイル
  4                             action => 'showsql') sql - 実行するSELECT AIアクション
  5 FROM dual;
```

SQL

```
SELECT COUNT("c"."CUST_ID") AS "customer_count" FROM "SH"."CUSTOMERS" "c"
```

SQL>



Select AIの利用手順

0. ネットワークACLの設定（OCI Generative AI Service以外を使用する場合）

```
-- adminユーザとして実行
BEGIN
  DBMS_NETWORK_ACL_ADMIN.APPEND_HOST_ACE(
    host => 'api.openai.com', -- AIプロバイダのAPIエンドポイント
    ace  => xs$ace_type(privilege_list => xs$name_list('http'),
                       principal_name => 'ADB_USER', -- Select AIを使用するDBユーザ名
                       principal_type => xs_acl.ptype_db)
  );
END;
/
```



Select AIの利用手順

1. Select AIを使用するユーザーに必要なパッケージの実行権限を付与

```
-- adminユーザとして実行  
GRANT EXECUTE ON DBMS_CLOUD_AI TO select_ai_user;
```

Select AIの利用手順

2. AIプロバイダにアクセスする際に使用するクレデンシャルの作成

```
BEGIN
  DBMS_CLOUD.CREATE_CREDENTIAL (
    credential_name => 'GENAI_CRED', -- 作成するクレデンシャルの名前
    user_ocid       => 'ocid1.user.oc1....', -- OCIユーザのOCID
    tenancy_ocid    => 'ocid1.tenancy.oc1....', -- テナンスのOCID
    private_key     => 'vZ6c0...', -- OCIユーザのプライベートキー
    fingerprint     => '86:7d:...' -- フィンガープリント
  );
END;
/
```



Select AIの利用手順

3. AIプロファイルの作成

```
BEGIN
  DBMS_CLOUD_AI.CREATE_PROFILE(
    profile_name => 'GENAI', -- 作成するAIプロファイル名
    attributes => '{
      "provider"      : "oci", -- Select AIで使用するAIプロバイダ
      "region"        : "ap-osaka-1", -- OCI GenAI Serviceを使用するリージョン
      "model"         : "meta.llama-3.1-70b-instruct", -- Select AIで使用するモデル
      "credential_name" : "GENAI_CRED", -- AIプロバイダアクセス時に使用するクレデンシャル
      "comments"      : "TRUE", -- 表や列のコメントの利用を有効化
      "conversation"  : "TRUE", -- 会話履歴を有効化
      "object_list"   : [ -- Select AIの対象とするスキーマ・オブジェクトのリスト
        {"owner": "SH", "name": "customers"},
        {"owner": "SH", "name": "products"}]
    }'
  );
END;
/
```



Select AIの利用手順

4. セッションで使用するAIプロファイルの設定し、SELECT AIを実行

```
EXEC DBMS_CLOUD_AI.SET_PROFILE('GENAI'); -- 使用するAIプロファイル名を指定
```

```
SELECT AI how many customers exist;
```

```
Number of Customers
```

```
-----
```

```
55500
```

Select AI with Retrieval Augmented Generation (Select AI RAG)

Select AI with Retrieval Augmented Generation (RAG)

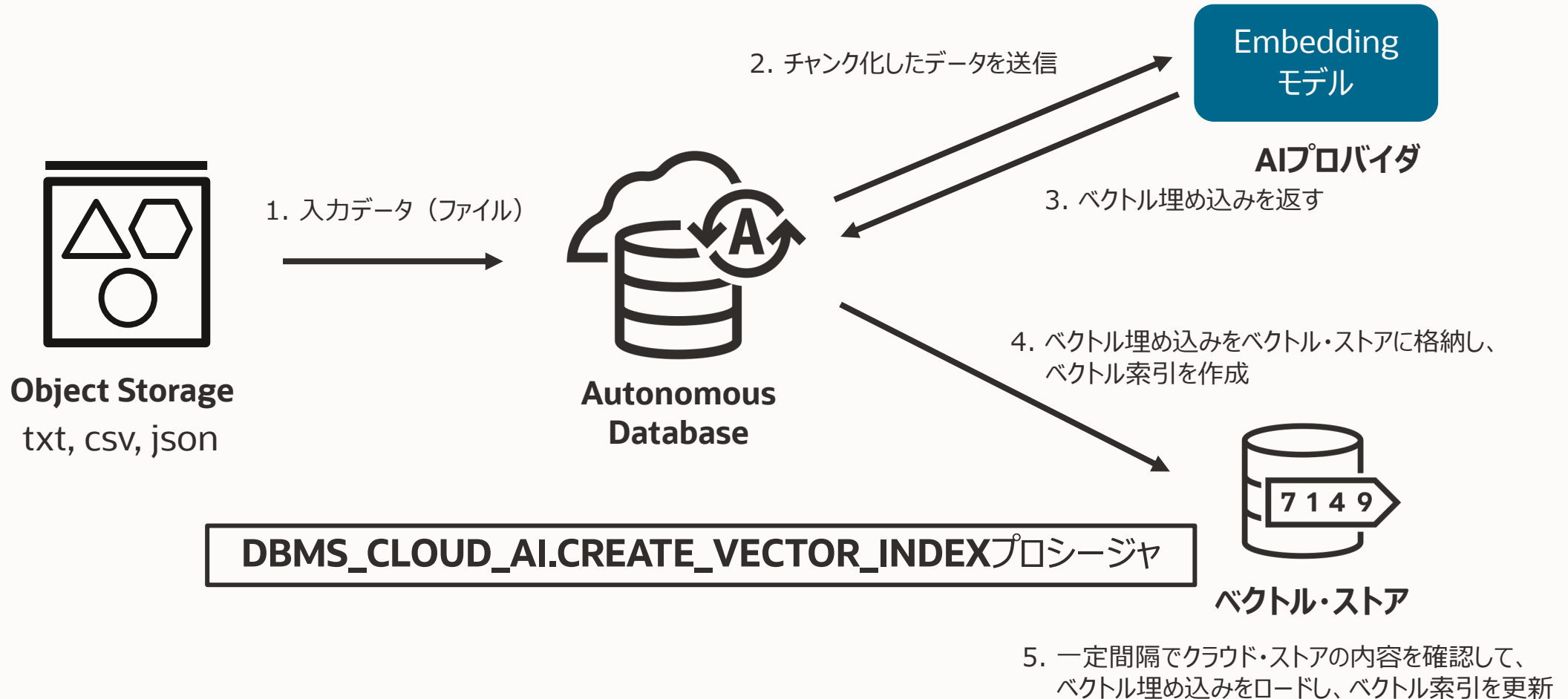
Select AI RAGとは

- ✓ LLMの知識とエンタープライズ・データの間ナレッジ・ギャップを解消
- ✓ データ・レイクに対する自然言語プロンプトを使用した問合せ
- ✓ フルマネージドのベクトル索引パイプラインにより、新しいデータのオーケストレーション・ステップを自動化
 - ✓ ベクトル・ストア（表）の作成
 - ✓ クラウド・ストレージ内にあるデータのチャンク分割
 - ✓ ベクトル埋め込みの生成とベクトル・ストアへの格納
 - ✓ ベクトル索引の作成
 - ✓ 一定間隔でクラウド・ストレージ内にあるデータの変更を確認し、変更があったらベクトル・ストア、ベクトル索引を更新



Select AI with Retrieval Augmented Generation (RAG)

Select AI RAGはベクトル・コンテンツの作成と更新を自動化



DBMS_CLOUD_AI.CREATE_VECTOR_INDEXプロシージャ

ベクトル・ストア、ベクトル索引を作成し、RAGパイプラインを使用してオブジェクト・ストアからデータを投入

RAGの利用に必要な以下のプロセスを1つのプロシージャで実現

- ベクトル・ストア（表）の作成
- クラウド・ストレージ内にあるデータのチャンク分割
- ベクトル埋め込みの生成とベクトル・ストアへの格納
- ベクトル索引の作成
- 一定間隔でクラウド・ストレージ内にあるデータの変更を確認し、ベクトル・ストア、ベクトル索引を更新

DBMS_CLOUD_AI.CREATE_VECTOR_INDEXプロシージャ

ベクトル・ストア、ベクトル索引を作成し、RAGパイプラインを使用してオブジェクト・ストアからデータを投入

```
BEGIN
  DBMS_CLOUD_AI.CREATE_VECTOR_INDEX(
    index_name    => 'MY_INDEX', -- 作成するベクトル索引の名前
    attributes    => '{"vector_db_provider" : "oracle", -- 'oracle'固定 (2024/12現在)
                      "profile_name" : "OCI_GENAI", -- 使用するAIプロファイル
                      "location" : "https://objectstorage.us-phoenix-1." ||
                      "oraclecloud.com/n/mynamespace/b/mybucket", -- オブジェクト・ストアのURI
                      "object_store_credential_name" : "OS_CRED", -- オブジェクト・ストアのクレデンシャル
                      "vector_table_name" : "MY_VECROT_STORE", -- ベクトル・ストアとして作成する表の名前
                      "vector_distance_metric" : "COSINE", -- ベクトル距離計算のタイプ
                      "chunk_size" : 2048, -- チャンクサイズ
                      "chunk_overlap" : 256, -- 隣接するテキストのチャンク間で重複する文字の量
                      "refresh_rate" : 720}' -- ベクトル・ストアの更新間隔
  );
END;
/
```



DBMS_CLOUD_AI.CREATE_VECTOR_INDEXプロシージャ

主なオプション (Attribute)

属性 (Attribute)	説明
chunk_size	入力データをチャンク化する際のチャンクサイズ
chunk_overlap	隣接するテキストのチャンクの間文字の重なり量
location	オブジェクト・ストアの場所 (URL)
match_limit	ベクトル検索で返す結果の最大数
object_storage_credential_name	オブジェクト・ストアにアクセスする際に使用するクレデンシャル
pipeline_name	作成するデータロード・パイプラインの名前
profile_name	使用するAIプロファイルの名前
refresh_rate	ベクトル・ストア内のデータの更新間隔 (分)、デフォルトは1日 (1440分)
vector_distance_metric	データベース内のベクトルを比較するために使用される距離計算のタイプ
vector_table_name	ベクトル埋め込みとチャンクされたデータを格納する表の名前
vector_dimension	ベクトル・ストア内の各ベクトルの次元数
vector_db_provider	ベクトル・ストアを管理するプロバイダ名 ('oracle'固定)



DBMS_CLOUD_AI.CREATE_VECTOR_INDEXプロシージャ vector_distance_metricに指定可能なベクトル距離計算のタイプ

- EUCLIDEAN (ユークリッド距離)
- L2_SQUARED (ユークリッド平方距離)
- COSINE (コサイン類似度 : デフォルト)
- DOT (ドット積類似度)
- MANHATTAN (マンハッタン距離)
- HAMMING (ハミング距離)



ベクトル索引の作成 例

```
BEGIN
  DBMS_CLOUD_AI.CREATE_PROFILE(
    profile_name => 'GENAI_TEXT_TRANSFORMER',
    attributes   => '{"provider": "oci",
                    "credential_name": "OCI$RESOURCE_PRINCIPAL",
                    "embedding_model": "cohere.embed-multilingual-v3.0" }'
  );
END;
```

AIプロファイルの作成

埋め込みモデルの指定

ベクトル索引の作成

ベクトル・ストア、
ベクトル索引、
RAGパイプラインの作成

```
BEGIN
  DBMS_CLOUD_AI.CREATE_VECTOR_INDEX(
    index_name  => 'MY_VECTOR_INDEX',
    attributes  => '{"vector_db_provider": "oracle",
                    "location": "https://my_namespace/my_bucket/my_data_folder",
                    "object_storage_credential_name": "OCI$RESOURCE_PRINCIPAL",
                    "profile_name": "GENAI_TEXT_TRANSFORMER",
                    "vector_dimension": 1536,
                    "vector_distance_metric": "cosine",
                    "chunk_overlap": 128,
                    "chunk_size": 1024,
                    "refresh_rate": 60}'
  );
END;
```

ベクトル索引の作成

Select AI RAGで使用するAIプロファイルの主なオプション (Attribute)

属性 (Attribute)	説明
provider	使用するAIプロバイダ
model	使用する生成モデル
embedding_model	使用する埋め込み (embedding) モデル
credential_name	使用するクレデンシャル名
vector_index_name	使用するベクトル索引名
max_tokens	予測に使用するトークンの数
temperature	生成モデル使用時の温度 (temperature) を指定
azure_resource_name	使用するAzure OpenAI Serviceのリソース名
azure_deployment_name	使用するAzure OpenAI Serviceのデプロイ名
azure_embedding_deployment_name	使用するAzure OpenAI Serviceの埋め込み (embedding) モデル名



ベクトル索引の作成

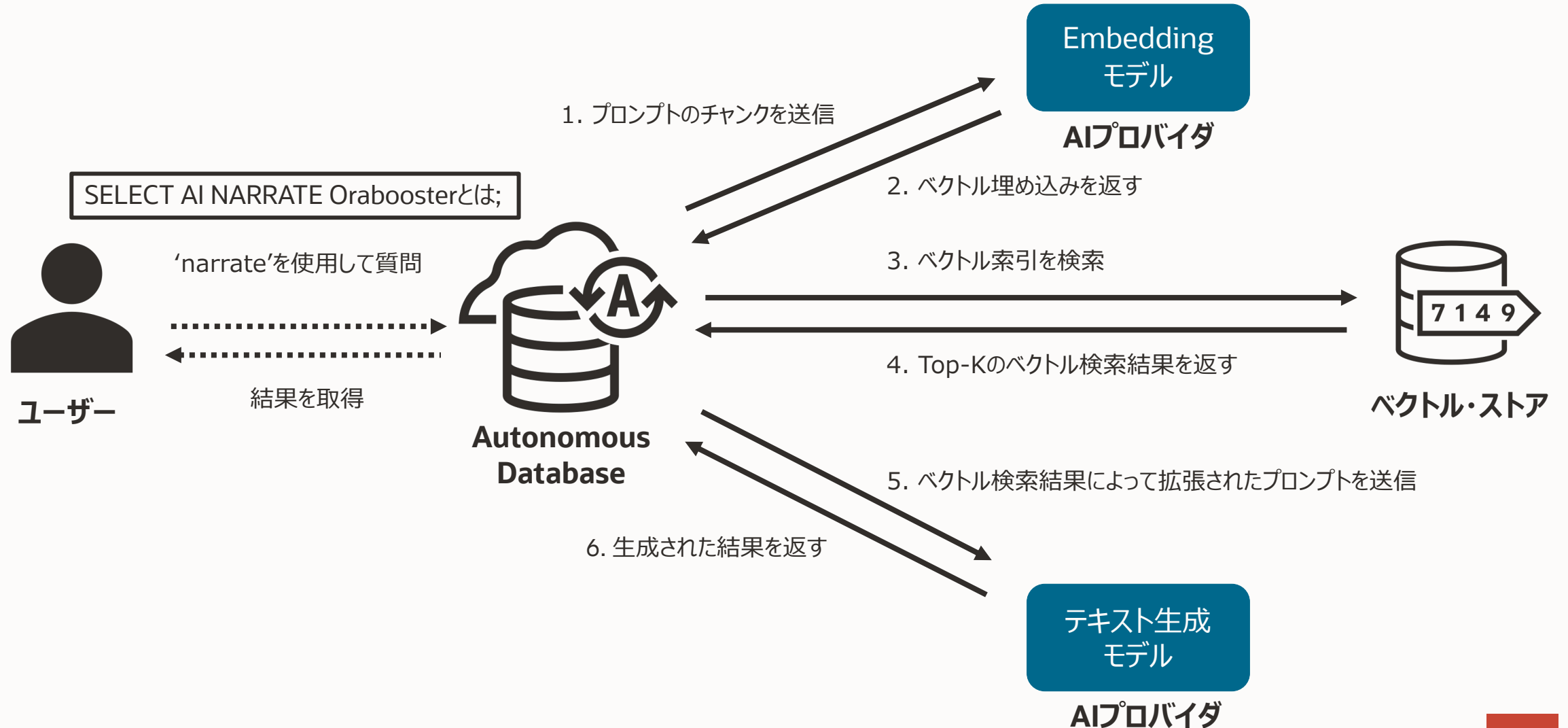
使用可能なAIプロバイダと埋め込み (Embedding) モデル

- OCI Generative AI Service (oci)
 - **cohere.embed-english-v3.0**
 - cohere.embed-multilingual-v3.0
 - cohere.embed-english-light-v3.0
 - cohere.embed-multilingual-light-v3.0
- Cohere (cohere)
 - **embed-english-v2.0**
- OpenAI (openai)
 - **text-embedding-ada-002**
- Azure OpenAI Service (azure)
 - text-embedding-ada-002
- Google (google)
 - **text-embedding-004**



Select AI RAGの実行

‘narrate’ アクションを使用して、Select AI RAGを実行



Select AI RAGの利用手順

0. ネットワークACLの設定（OCI Generative AI Service以外を使用する場合）

```
-- adminユーザとして実行
BEGIN
  DBMS_NETWORK_ACL_ADMIN.APPEND_HOST_ACE(
    host => 'api.openai.com', -- AIプロバイダのエンドポイント
    ace  => xs$ace_type(privilege_list => xs$name_list('http'),
      principal_name => 'ADB_USER', -- Select AIを使用するDBユーザ名
      principal_type => xs_acl.ptype_db)
  );
END;
/
```



Select AI RAGの利用手順

1. Select AI RAGを使用するユーザに必要なパッケージの実行権限を付与

```
-- adminユーザとして実行  
GRANT EXECUTE ON DBMS_CLOUD_AI TO select_ai_user;  
GRANT EXECUTE ON DBMS_CLOUD_PIPELINE TO select_ai_user;
```



Select AI RAGの利用手順

2. AIプロバイダにアクセスする際に使用するクレデンシャルの作成

```
BEGIN
  DBMS_CLOUD.CREATE_CREDENTIAL (
    credential_name => 'OCI_GENAI_CRED',
    user_ocid       => 'ocid1.user.oc1....',
    tenancy_ocid    => 'ocid1.tenancy.oc1....',
    private_key     => 'vZ6c0...',
    fingerprint     => '86:7d:...'
  );
END;
/
```



Select AI RAGの利用手順

3. Object Storageにアクセスする際に使用するクレデンシャルの作成

```
BEGIN
  DBMS_CLOUD.CREATE_CREDENTIAL (
    credential_name => 'OCI_OS_CRED',
    user_ocid       => 'ocid1.user.oc1....',
    tenancy_ocid    => 'ocid1.tenancy.oc1....',
    private_key     => 'vZ6c0...',
    fingerprint     => '86:7d:...'
  );
END;
/
```

Select AI RAGの利用手順

4. AIプロファイルの作成

```
BEGIN
  DBMS_CLOUD_AI.CREATE_PROFILE(
    profile_name => 'GENAI_RAG',
    attributes   => '{
      "provider" : "oci",
      "region"   : "ap-osaka-1",
      "model"    : "meta.llama-3.1-70b-instruct",
      "embedding_model" : "cohere.embed-multilingual-v3.0",
      "credential_name" : "OCI_GENAI_CRED",
      "vector_index_name" : "MY_VECTOR_INDEX"
    }'
  );
END;
/
```



Select AI RAGの利用手順

5. DBMS_CLOUD_AI.CREATE_VECTOR_INDEXを実行

(ベクトル・ストア、ベクトル索引およびベクトル・ストア更新のためのパイプラインの作成)

```
BEGIN
  DBMS_CLOUD_AI.CREATE_VECTOR_INDEX(
    index_name => 'MY_INDEX',
    attributes => '{"vector_db_provider": "oracle",
                  "location": "https://objectstorage.ap-tokyo-
1.oraclecloud.com/n/namespace/b/bucketname/o/",
                  "object_storage_credential_name": "OCI_OS_CRED",
                  "profile_name": "GENAI_RAG",
                  "vector_distance_metric": "cosine",
                  "vector_dimension": 1024,
                  "chunk_overlap":128,
                  "chunk_size":400
                  }'
  );
END;
/
```



Select AI RAGの利用手順

6. セッションで使用するAIプロファイルの設定し、Select AI RAG (SELECT AI NARRATE) を実行

```
EXEC DBMS_CLOUD_AI.SET_PROFILE('GENAI_RAG');
```

```
SELECT AI NARRATE OraBoosterとは;
```

Synthetic Data Generation (SDG)

Synthetic Data Generation (SDG)

Synthetic Data Generation（合成データ生成）とは

- ✓ 入力された自然言語プロンプトを考慮し、LLMを使用して現実的な合成データを生成
- ✓ Autonomous Database メタデータ・クローンとの統合によるテスト/開発シナリオ
- ✓ シンプルかつスケーラブル



Synthetic Data Generationのユースケース

主なユースケース

新規プロジェクトの
早期立ち上げ

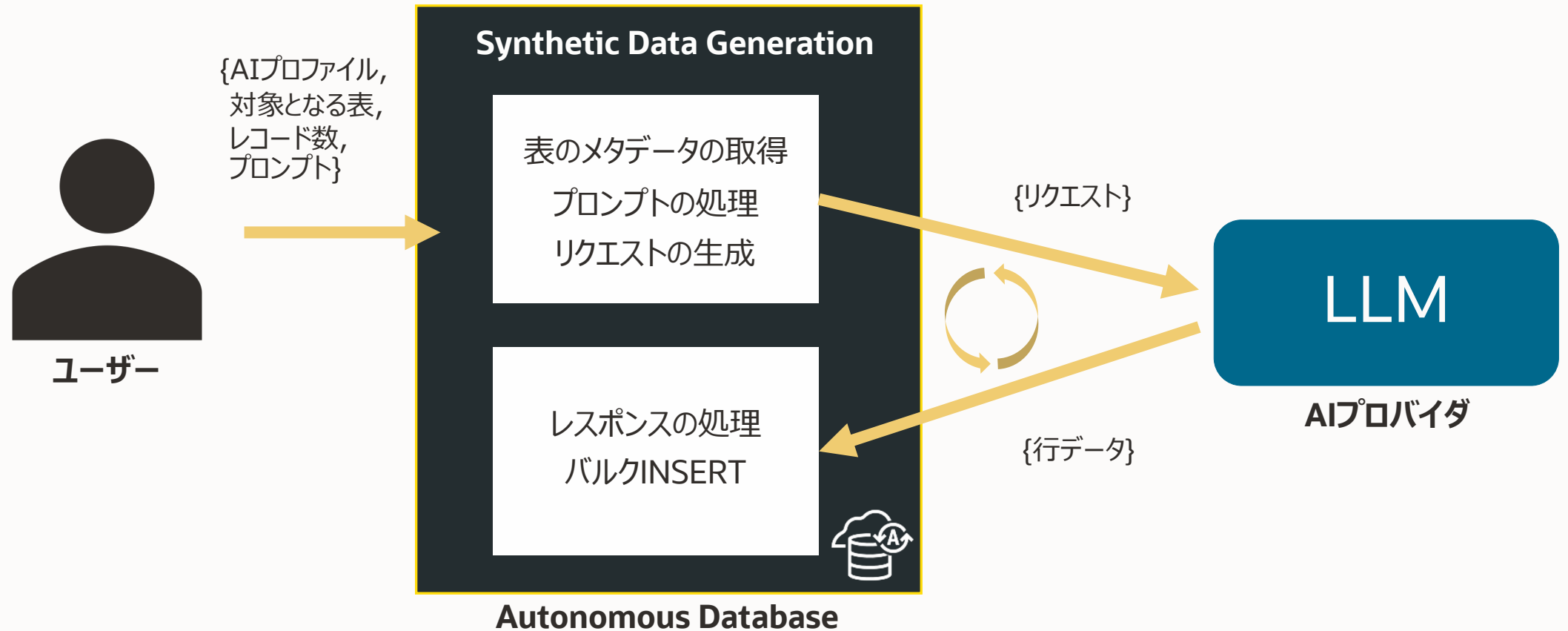
メタデータ・クローンへの
テストデータの投入

ユーザー・エクスペリエンス
の検証

機械学習とAIプロジェクト

Synthetic Data Generationのプロセス

自然言語プロンプトに沿った合成データをLLMを使用して生成し、既存の表にロード



DBMS_CLOUD_AI.GENERATE_SYNTHETIC_DATA プロシージャ



DBMS_CLOUD_AI.GENERATE_SYNTHETIC_DATAプロセス

LLMを使用して自然言語プロンプトに沿った合成データを生成し、既存の表にロード

- 単一の表に対してだけでなく、一度に複数の表に対して、指定したレコード数の合成データを生成してロード
- ユーザー・プロンプトを指定して、合成データ生成のためのガイドを追加可能
- 空の表だけでなく、すでにデータが格納されている表にも合成データをロード（追加）可能
 - サンプルした既存のデータをガイドとした合成データの生成
 - 表の統計情報を使用した合成データの生成（メタデータ・クローン）
- 処理の優先度（並列度）を指定可能



DBMS_CLOUD_AI.GENERATE_SYNTHETIC_DATAプロセス

例) 単一表に対する合成データのロード

```
BEGIN
  DBMS_CLOUD_AI.GENERATE_SYNTHETIC_DATA(
    profile_name => 'OCI_GENAI, -- 使用するAIプロファイル
    object_name  => 'EMP' -- 合成データの生成対象となる表
    owner_name   => 'SCOTT', -- 合成データの生成対象となる表の所有者
    record_count => 100, -- 生成するレコード数
    user_prompt  => 'COMM for all employees is 1000', -- 合成データ生成をガイドするプロンプト
    params       => '{"comments":"true", -- データ生成をガイドするためのLLMへのコメント送信を有効化
                    "priority":"MEDIUM"}' -- LLMに送信されるパラレル・リクエストの数を定義する優先度
  );
END;
/
```



DBMS_CLOUD_AI.GENERATE_SYNTHETIC_DATAプロセス

例) 複数表に対する合成データのロード

```
BEGIN
  DBMS_CLOUD_AI.GENERATE_SYNTHETIC_DATA(
    profile_name => 'OCI_GENAI', -- 使用するAIプロファイル
    object_list => '[           -- 合成データの生成対象となる表のリスト (配列)
      {"owner": "ADB_USER", "name": "Director", "record_count":5},
      {"owner": "ADB_USER", "name": "Movie_Actor", "record_count":5},
      {"owner": "ADB_USER", "name": "Actor", "record_count":10},
      {"owner": "ADB_USER", "name": "Movie", "record_count":5,
        "user_prompt":"all movies are released in 2009"}
    ]'
  );
END;
/
```



DBMS_CLOUD_AI.GENERATE_SYNTHETIC_DATAプロセス

主なオプション (パラメータ)

	説明
profile_name	使用するAIプロファイルの名前
object_name	合成データの生成対象となるオブジェクト名 (表名)
owner_name	合成データの生成対象となるオブジェクト名の所有者 (ユーザ)
record_count	生成する合成データのレコード数
user_prompt	データ生成をガイドするためのプロンプト
object_list	合成データの生成対象となるオブジェクトのリスト (複数表を対象とする場合)
sample_rows	データ生成をガイドするサンプルとして使用する表の行数
priority	合成データを生成するためにLLMに送信されるパラレル・リクエストの数を定義する優先度 (HIGH/MEDIUM/LOW)
comments	データ生成をガイドするためのコメントを有効化または無効化 (TRUE/FALSE)



Synthetic Data Generationの利用手順

0. ネットワークACLの設定（OCI Generative AI Service以外を使用する場合）

```
-- adminユーザとして実行
BEGIN
  DBMS_NETWORK_ACL_ADMIN.APPEND_HOST_ACE(
    host => 'api.openai.com', -- AIプロバイダのエンドポイント
    ace  => xs$ace_type(privilege_list => xs$name_list('http'),
      principal_name => 'ADB_USER', -- Select AIを使用するDBユーザ名
      principal_type => xs_acl.ptype_db)
  );
END;
/
```



Synthetic Data Generationの利用手順

1. Synthetic Data Generationを使用するユーザに必要なパッケージの実行権限を付与

```
-- adminユーザとして実行  
GRANT EXECUTE ON DBMS_CLOUD_AI TO select_ai_user;
```



Synthetic Data Generationの利用手順

2. AIプロバイダにアクセスする際に使用するクレデンシャルの作成

```
BEGIN
  DBMS_CLOUD.CREATE_CREDENTIAL(
    credential_name => 'GENAI_CRED',
    user_ocid       => 'ocid1.user.oc1....',
    tenancy_ocid    => 'ocid1.tenancy.oc1....',
    private_key     => 'vZ6c0...',
    fingerprint     => '86:7d:...'
  );
END;
/
```

Synthetic Data Generationの利用手順

3. AIプロファイルの作成

```
BEGIN
  DBMS_CLOUD_AI.CREATE_PROFILE(
    profile_name => 'GENAI_SDG',
    attributes   => '{
      "provider": "oci",
      "credential_name": "GENAI_CRED",
      "model": "meta.llama-3.1-70b-instruct",
      "region": "ap-osaka-1"
    }'
  );
END;
/
```

Synthetic Data Generationの利用手順

4. 合成データの生成

```
BEGIN
  DBMS_CLOUD_AI.GENERATE_SYNTHETIC_DATA(
    profile_name => 'GENAI_SDG',
    object_list => '[
      {"owner": "ADB_USER", "name": "Director", "record_count":5},
      {"owner": "ADB_USER", "name": "Movie_Actor", "record_count":5},
      {"owner": "ADB_USER", "name": "Actor", "record_count":10},
      {"owner": "ADB_USER", "name": "Movie", "record_count":5,
        "user_prompt":"all movies released in 2009"}
    ]'
  );
END;
/
```



Autonomous DatabaseにおけるAIの活用（再掲）

生成AIを活用した自然言語プロンプトによるDBデータ検索、ベクトル検索、合成データ生成

- **Select AI (for Natural Language Interaction)** **19c** **23ai**
 - 大規模言語モデル（LLM）を利用して、自然言語プロンプトからSQLを生成、実行
- **Select AI with Retrieval Augmented Generation (RAG)** **23ai**
 - クラウド・ストレージ上の企業データ（ファイル）からベクトル・ストアおよびベクトル索引を作成するプロセスを自動化
 - 入力された自然言語プロンプトからベクトル埋め込みを生成し、ベクトル索引を使用してベクトル検索を実行、ベクトル検索の実行結果を使用して入力された自然言語プロンプトを補強し、LLMに問合せを実行
- **Synthetic Data Generation(SDG)** **19c** **23ai**
 - 入力された自然言語プロンプトを考慮し、大規模言語モデル（LLM）を使用して、より適切でスキーマに特化した合成データを生成

参考情報

Select AI

DBMS_CLOUD_AIパッケージのサブプログラム (1)

サブプログラム	用途
CREATE_PROFILEプロシージャ	AIプロファイルを作成する
DROP_PROFILEプロシージャ	AIプロファイルを削除する
ENABLE_PROFILEプロシージャ	AIプロファイルを有効化する
DISABLE_PROFILEプロシージャ	AIプロファイルを無効化する
GET_PROFILEファンクション	現行セッションに設定されているAIプロファイルを返す
GET_PROFILEプロシージャ	現行セッションに設定されているAIプロファイルとそのオーナーを取得する
SET_ATTRIBUTEプロシージャ	AIプロファイルの属性 (attribute) を設定する
SET_PROFILEプロシージャ	現行セッションで使用するAIプロファイルを設定する
GENERATEファンクション	ステートレス方式でAI翻訳を実行する



Select AI

DBMS_CLOUD_AIパッケージのサブプログラム (2)

サブプログラム	用途
CREATE_VECTOR_INDEXプロシージャ	ベクトル・ストア、ベクトル索引を作成し、非同期スケジューラ・ジョブを使用してオブジェクト・ストアからデータを投入する
DROP_VECTOR_INDEXプロシージャ	CREATE_VECTOR_INDEXプロシージャで作成されたベクトル索引、ベクトル・ストア、スケジューラ・ジョブ等を削除する
ENABLE_VECTOR_INDEXプロシージャ	ベクトル索引を有効化する
DISABLE_VECTOR_INDEXプロシージャ	ベクトル索引を無効化する
UPDATE_VECTOR_INDEXプロシージャ	CREATE_VECTOR_INDEXプロシージャで作成した既存のベクトル索引の属性を更新する
GENERATE_SYNTHETIC_DATAプロシージャ	合成データを生成する



Select AI

ディクショナリ・ビュー

ディクショナリ・ビュー	説明
DBA_CLOUD_AI_PROFILES	データベース内のAIプロファイルのリスト
USER_CLOUD_AI_PROFILES	スキーマ内のAIプロファイルのリスト
DBA_CLOUD_AI_PROFILE_ATTRIBUTES	データベース内のAIプロファイルの属性の詳細
USER_CLOUD_AI_PROFILE_ATTRIBUTES	スキーマ内のAIプロファイルの属性の詳細
DBA_CLOUD_VECTOR_INDEXES	データベース内のベクトル索引のリスト
USER_CLOUD_VECTOR_INDEXES	スキーマ内のベクトル索引のリスト
DBA_CLOUD_VECTOR_INDEX_ATTRIBUTES	データベース内のベクトル索引の属性の詳細
USER_CLOUD_VECTOR_INDEX_ATTRIBUTES	スキーマ内のベクトル索引の属性の詳細

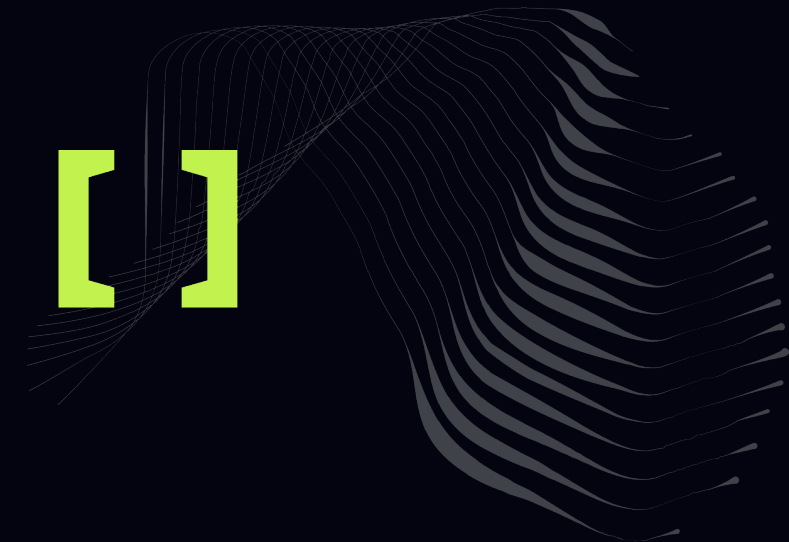


Select AI

関連リンク

- Select AIについて（マニュアル）
https://docs.oracle.com/cd/E83857_01/paas/autonomous-database/serverless/adbsb/select-ai-about.html#GUID-D3FFFC75-0BC6-4D17-9FB0-C60E1FEC2E46
- Select AIの使用例（マニュアル）
https://docs.oracle.com/cd/E83857_01/paas/autonomous-database/serverless/adbsb/select-ai-examples.html
- LiveLabs : Chat with Your Data in Autonomous Database Using Generative AI
<https://apexapps.oracle.com/pls/apex/r/dbpm/livelabs/view-workshop?wid=3831>
- OCIチュートリアル : SELECT AIを試してみよう
<https://oracle-japan.github.io/ocitutorials/adb/adb111-select-ai>
- OCIチュートリアル : SELECT AI with RAGを試してみよう
<https://oracle-japan.github.io/ocitutorials/ai-vector-search/ai-vector108-select-ai-with-rag>
- Qiita : Select AIを使ってデータベース内のデータと会話してみる
<https://qiita.com/marfujim/items/3e2fe1861a7b148e41b1>
- Qiita : SELECT AI for Synthetic Data Generationで日本語のテストデータを自動生成を試してみた
<https://qiita.com/500InternalServerError/items/855d58cd0d7b1733639c>

Oracle APEXにおけるAIの活用



Oracle APEX

Oracle APEXとは

APEXは、**世界有数の**エンタープライズ・**ローコード**・アプリケーション・プラットフォーム

ワールドクラスの機能を持った、スケーラブルでセキュアなエンタープライズ・アプリケーションを、**20倍高速かつ、100分の1のコード量**で構築可能

Oracle APEXは、クラウドとオンプレミスを含む、あらゆる場所に**モバイル、Web、デスクトップ**・アプリケーションを開発およびデプロイするための最も生産的な方法を提供



ā'pěks
(#orclapex)

2000万+

これまでに作成されたアプリケーション

6,000+

一日に作成されるアプリケーション

850,000+

世界中のAPEX開発者

60,000+

150か国にわたる顧客

90%

のFortune 500社がOracle APEX
を使用



Oracle APEXの特徴

データベース中心のWebアプリケーション開発フレームワーク



デスクトップ向けと
モバイル向けの
Webアプリ開発



データベースが
提供する能力を活用



SQLのスキルを
活用



APEXのAI機能

APEXのAI機能：自然言語で開発し、設計やコーディングのコストを最小化

アプリケーション作成アシスタント

自然言語の指示を基に
アプリケーションを設計・作成

「Let's create an app together. Describe the app you want to create, including the data or tables it will use. Use HIGHSCHOOLS table to create an app that allows students or parents to search for schools. In addition to that, create a dashboard page with a pie chart showing the number of schools per district, and a chart showing the top 10 schools with the highest percentage of applicants and seats.

はい、次のページがある。学校検索アプリというアプリケーションを作成します。

アプリケーションの作成

ブループリントを作成

ブループリントを基にアプリを作成

Method	Count
Method 1	120
Method 2	150
Method 3	180
Method 4	210
Method 5	240
Method 6	270
Method 7	300
Method 8	330
Method 9	360
Method 10	390
Method 11	420

APEX AIアシスタント

自然言語の指示で
JavaScript、CSS、SQL、PL/SQLの
コード生成を自動化、コード改善も支援

出願者と席数を計算し、倍率が高い学校の上位10校を棒グラフで表示してください

```

select SCHOOL_NAME,
       APPLICANTS,
       SEATS,
       (APPLICANTS / SEATS) as RATIO
from HIGHSCHOOLS
order by RATIO desc
fetch first 10 rows only;
    
```

SQLクエリを生成

会話型AIダイアログ

コード不要で簡単に実装できるチャットボット
チャットボットを通じて自然言語で
データベース内のデータを問い合わせ可能

学校情報アプリ

学校情報アプリ

検索...

特徴

- 人文科と数学 (18)
- 科学と数学 (45)
- コンピュータサイエンスとテクノロジー (15)
- 芸術や音楽 (25)
- 法律と政府 (25)
- 舞台芸術 (20)
- エンジニアリング (18)

距離

- 10-15マイル (75)
- 15-20マイル (4)
- 5-10マイル (205)
- 0マイル (46)

地区

ウィリアムズブリッジ・オクスビル

サイプレスヒルズ

詳細

ようこそ！お手伝い出来る事はありますか？

棒グラフで表示したい学校はありますか？

はい、バスケットボールは男子と女子の両方のためにPALSプログラムとして提供されています。学校のバスケットボール部は、その中でプレーする機会によって異なる場合があります。

この学校の特色は何ですか？

ヒルサイドアーツ&レジャーアカデミーは、地産地消、音楽、作文に重点を置いたカリキュラムを提供することが特徴です。学生は、芸術に対する理解を深め、自主的で革新的な思考を持ち、協力する力を身につけた独立した思考者として卒業できるよう支援されます。

他の学校の事は作家になることなのですが、それに関連する授業はありますか？

はい、ヒルサイドアーツ&レジャーアカデミーでは作文に関連する授業が提供されています。具体的には「ライティング」の授業があり、作家になるためのスキルや知識を学ぶことができます。その他にも、英語圏と作文のAPコースなどもあります。これが作家としての夢を追求する方に最適な機会が揃っています。



アプリケーション作成アシスタント

The image shows a multi-step process for creating an application using the APEX Assistant. It starts with a chat interface where a user provides a prompt in Japanese. The assistant then generates a blueprint, which is shown in a separate window. Finally, the application is created and displayed as a web page with a data table.

ブループリントを作成

アプリを作成

Borough	School Name	Neighborhood	Interest	Method	Total Students	Graduation Rate	Attendance Rate	College Career Rate
クイーンズ	アカデミー オブ メディカル テクノロジー カレッジ ボード スクール	ファーロックウェイ-ハイ スクオーター	医療専門職	編集オプション	639	73	8	32
クイーンズ	アストリアのヤング・ウイメンズ・リーダーシップ・スクール (YWLS)	オールドアストリア	人文科学と学際	上映	565	96	94	92
クイーンズ	アメリカ研究アカデミー	クイーンズブリッジ-LIC	人文科学と学際	編集オプション	1,038	93	94	72
クイーンズ	イーグルアカデミーIII	セントオールバンス	人文科学と学際	編集オプション	531	93	91	0
クイーンズ	イーストウエスト国際学院	フラッシング	人文科学と学際	編集オプション	649	93	91	87
クイーンズ	ウィリアム・カレン・ブライアント高校	アストリア	法律と政府	上映	2,394	72	89	52
クイーンズ	イクセルシオール予備高等学校	スプリングフィールド カレッジ ノース	科学と数学	編集オプション	483	79	87	52
クイーンズ	エネルギーテック高校	クイーンズブリッジ-LIC	エンジニアリング	編集オプション	511	93	94	0

- ユーザーが入力した**プロンプト（自然言語）**を基に、**アプリケーションのブループリント（設計図）**を作成
- **追加で指示**を出し、生成されたブループリントを**改良**
- 作成した**ブループリント**を基に、**アプリケーションを簡単に作成**
- **アプリ開発が簡素化**され、**カスタマイズ機能の開発に集中**



APEX AIアシスタント

コード・エディタ - SQL問合せ

APEXアシスタント

クエリービルダー

チャットのクリア

```
1 select ID,
2 BOROUGH,
3 SCHOOL_NAME,
4 NEIGHBORHOOD,
5 INTEREST,
6 METHOD,
7 TOTAL_STUDENTS,
8 GRADUATION_RATE,
9 ATTENDANCE_RATE,
10 COLLEGE_CAREER_RATE,
11 SAFE,
12 SEATS,
13 APPLICANTS,
14 DBN,
15 LATITUDE,
16 LONGITUDE,
17 LANGUAGE_CLASSES,
18 ADVANCED_PLACEMENT_COURSES,
19 SCHOOL_SPORTS,
20 FAQ_URL,
21 OVERVIEW_PARAGRAPH,
22 A
23 A
24 D
25 E
```

出願者と席数を計算し、倍率が高い学校の上位10校を棒グラフで表示してください

```
sql
select SCHOOL_NAME,
APPLICANTS,
SEATS,
(APPLICANTS / SEATS) as RATIO
from HIGHSCHOOLS
order by RATIO desc
fetch first 10 rows only;
```

ここにメッセージを入力し

取消 OK

SQLクエリを生成

```
select SCHOOL_NAME,
APPLICANTS,
SEATS,
(APPLICANTS / SEATS) as RATIO
from HIGHSCHOOLS
order by RATIO desc
fetch first 10 rows only;
```

- ユーザーが入力した**プロンプト**（**自然言語**）からPL/SQL、CSS、HTML、JavaScriptの**記述を自動化**
- PL/SQL、CSS、HTML、JavaScriptコードの**改善、説明、バグの修正も簡単**に行う事が可能
- **生成AIの支援**によって、より簡単で、よりリッチな**アプリ開発エクスペリエンス**を提供

会話型AIダイアログ



- **コード不要**で、直ぐに簡単に実装できる**チャットボット**（会話型インターフェース）
- 生成AIや自然言語処理コンポーネントの、ゼロからの**構築が不要**
- **生成AI**と連動し、エンド・ユーザーとアプリが**自然言語で「会話」**する事を可能に
- **ユーザー・インターフェース上から**データベース内の**データを問い合わせる**ことが可能に



単純なRAGであれば、データベースの機能のみで実装可能

一般的なAIチャットボット開発



複数の要素技術の組み合わせ
インテグレーション作業が必要

```
import getpass
import os

os.environ["OPENAI_API_KEY"] = getpass.getpass()

from langchain_openai import ChatOpenAI
llm = ChatOpenAI(model="gpt-3.5-turbo-0125")

import hub
from langchain import hub
from langchain_chroma import Chroma
from langchain_community.document_loaders import WebBaseLoader
from langchain_core.output_parsers import StrOutputParser
from langchain_core.prompts import RunnablePrompt
from langchain_openai import OpenAIEmbeddings
from langchain_text_splitters import RecursiveCharacterTextSplitter

# Load, chunk and index the contents of the blog.
loader = WebBaseLoader(
    web_paths=["https://lilianweng.github.io/posts/2023-06-23-agent/"],
    bs_kwargs=dict(
        parse_only=[SoupStrainer(
            class_=["post-content", "post-title", "post-header"]
        )],
    ),
)

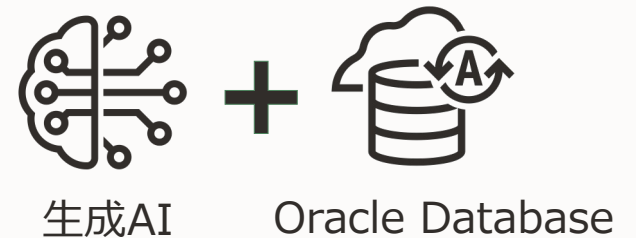
import gradio as gr

def greet(name):
    return "Hello " + name + "!"

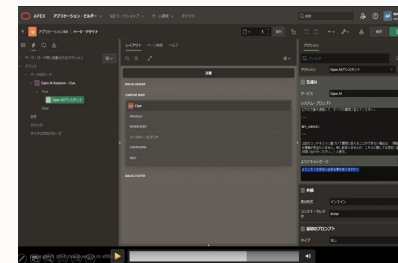
demo = gr.Interface(fn=greet, inputs="text", outputs="text")
demo.launch()
```

プログラミングによる開発
(コード開発)

APEXでのAIチャットボット開発



必要な技術は事前統合済
APEXはOracle Databaseで無償提供



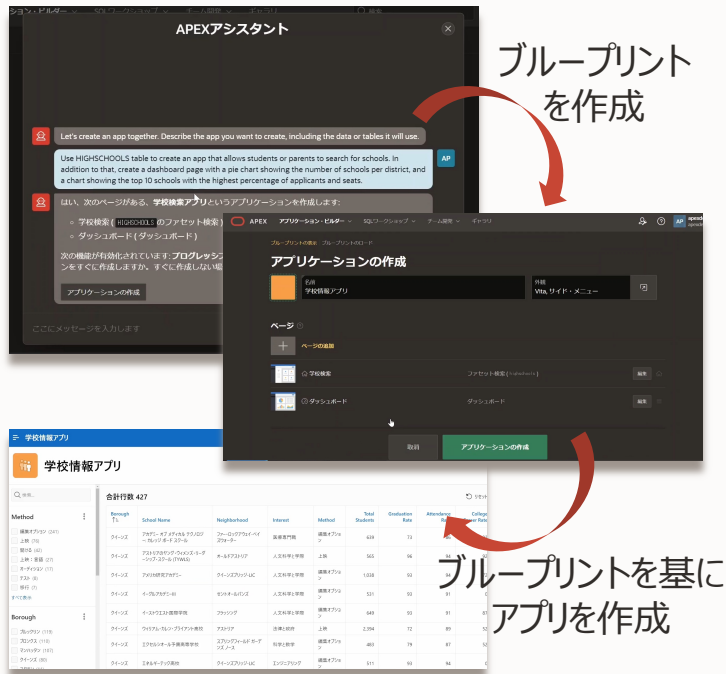
ブラウザ上で設定を進める開発
(ローコード開発)



APEXのAI機能：自然言語で開発し、設計やコーディングのコストを最小化（再掲）

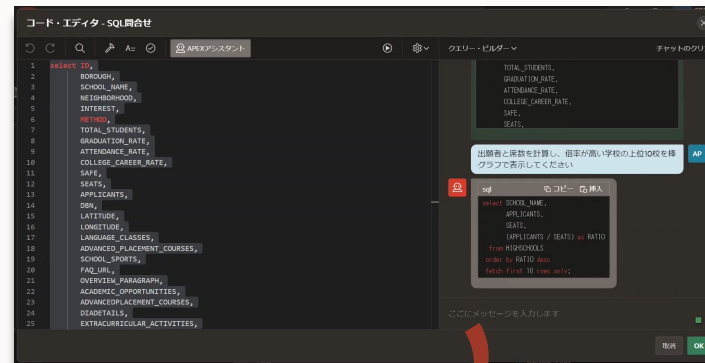
アプリケーション作成アシスタント

自然言語の指示を基に
アプリケーションを設計・作成



APEX AIアシスタント

自然言語の指示で
JavaScript、CSS、SQL、PL/SQLの
コード生成を自動化、コード改善も支援



```
select SCHOOL_NAME,  
       APPLICANTS,  
       SEATS,  
       (APPLICANTS / SEATS) as RATIO  
from HIGHSCHOOLS  
order by RATIO desc  
fetch first 10 rows only;
```

会話型AIダイアログ

コード不要で簡単に実装できるチャットボット
チャットボットを通じて自然言語で
データベース内のデータを問い合わせ可能



Oracle APEX AI機能のメリット

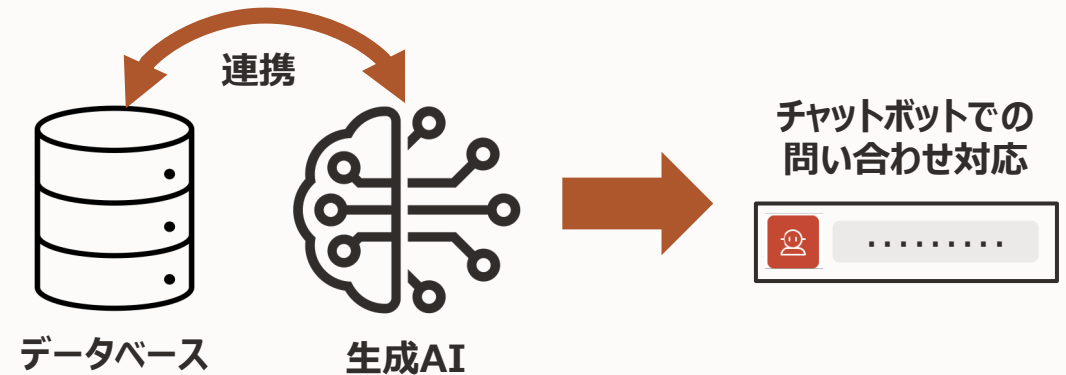
開発生産性のさらなる向上

- もともとコードをほとんど書かないローコード開発ツール
- 生成AIを利用したコード記述支援を提供
- 経験のないエンジニアでも一定レベル以上のコードを記述可能



ユーザービリティの大幅な向上

- 本来、沢山のコードを記述して実装する必要があるAIチャットボットをローコードで実装
- データベース内のデータを自然言語で問い合わせ可能
- 知りたい情報への回答、サマリーの作成、FAQシステムの作成など様々なユースケースで利用可能



ORACLE