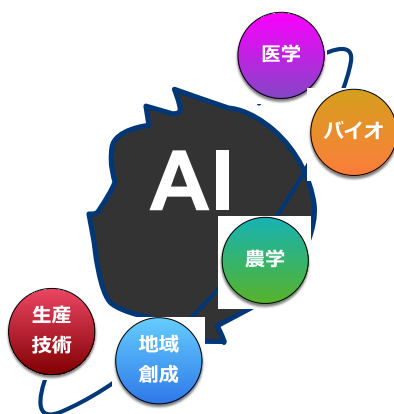


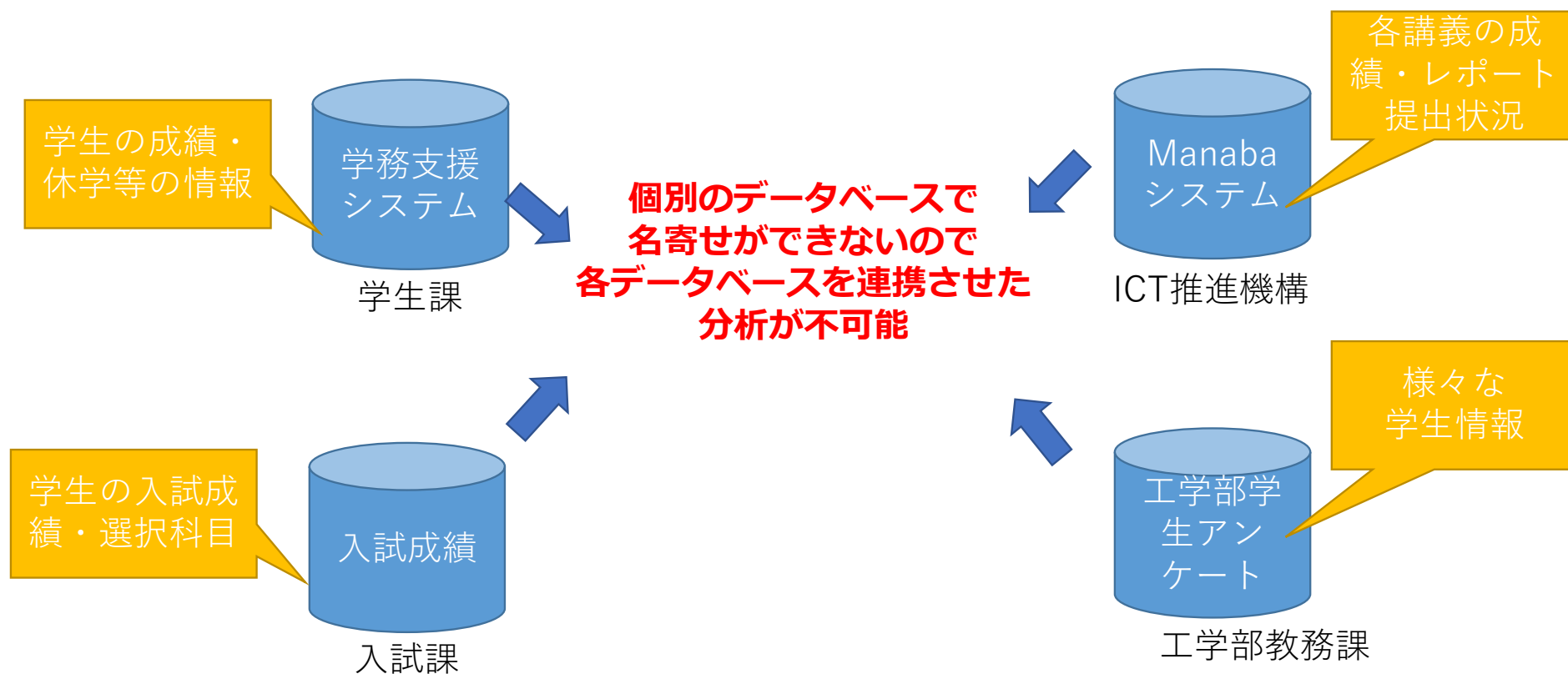
Learning Analytics 手法の確立と データ利活用環境整備

工学部附属クロス情報科学研究センター

AI部門 岩井儀雄



大学の学生に関するデータの現状

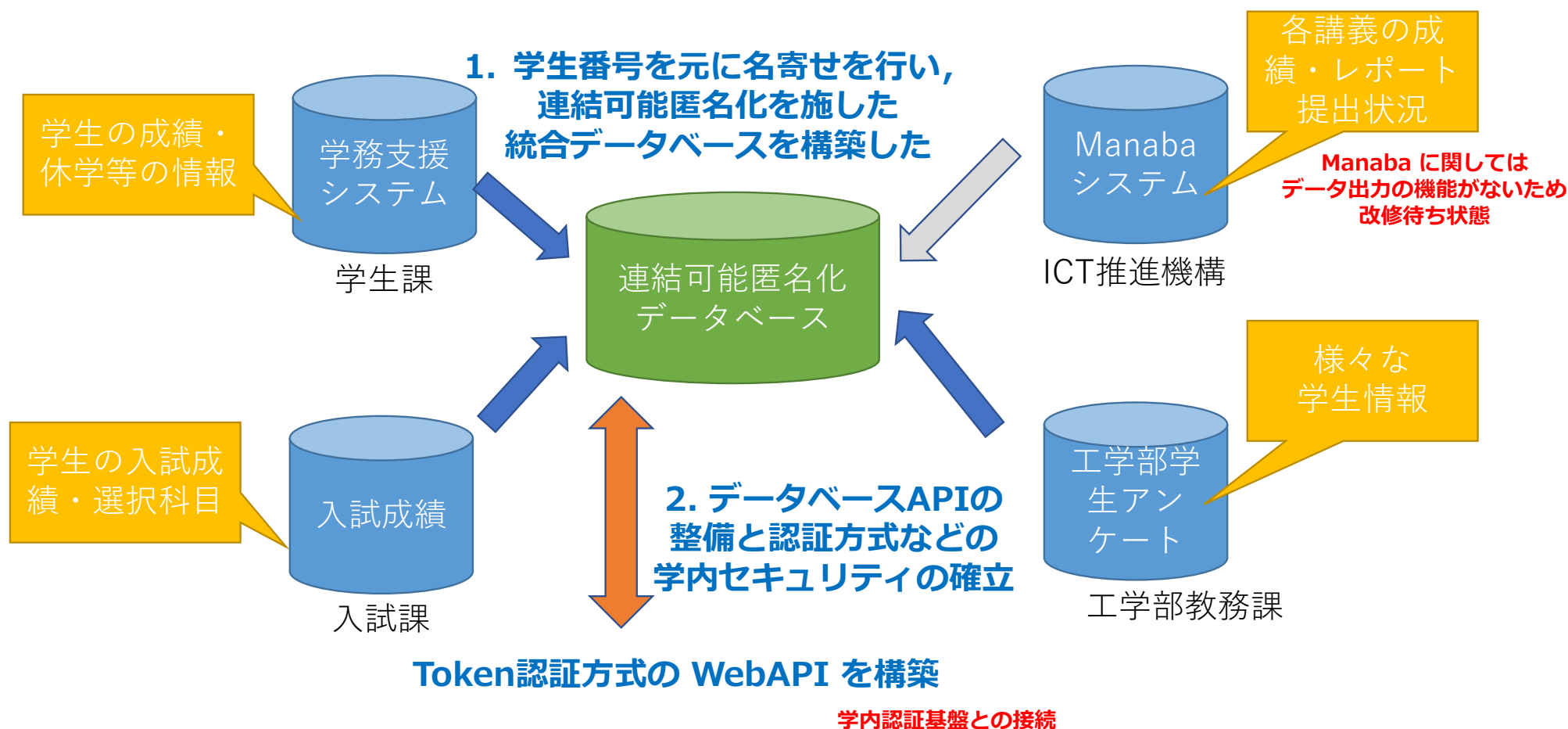


本来は学生番号で名寄せが出来るはず

原因 1 : 各部署でデータを保持し守秘義務が発生している

原因 2 : 名寄せでデータを統合して匿名化し, 対応表を持つ機関が無い

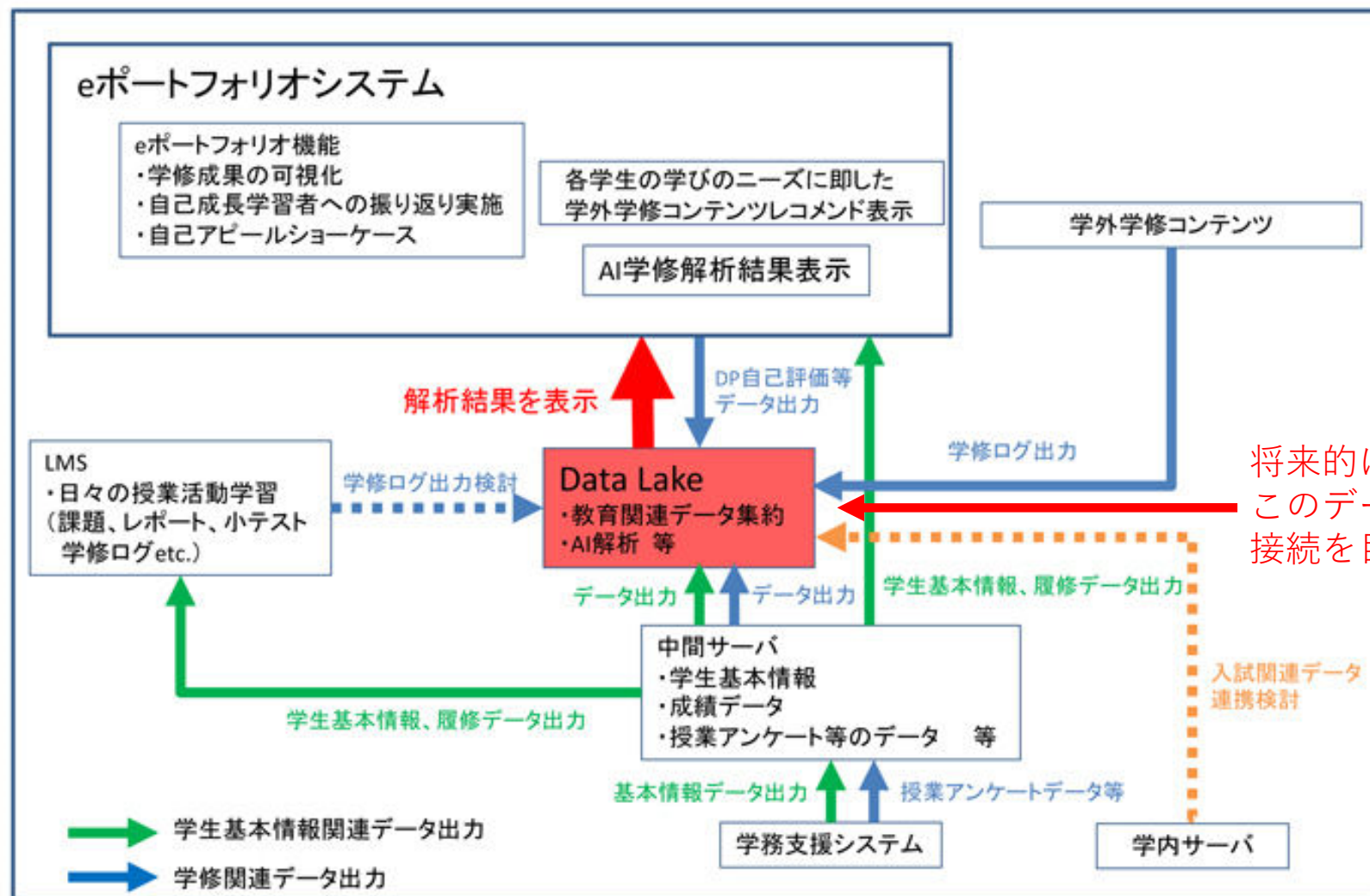
実施内容



- R4年度実施内容

- ユーザ認証 の構築(学内RADIUS認証と接続して認証)
- データベースの更新
- ドキュメント類の整備
- デジタルユニバーシティとの調整（調整中）
 - 鳥取大学のデータレイク構想との接続検討

学修分析基盤整備のためのシステム関連図



• データベースサーバ

- 学務支援システム内の工学部データを元にサーバを立ち上げ
- 学内認証基盤と接続してユーザ認証

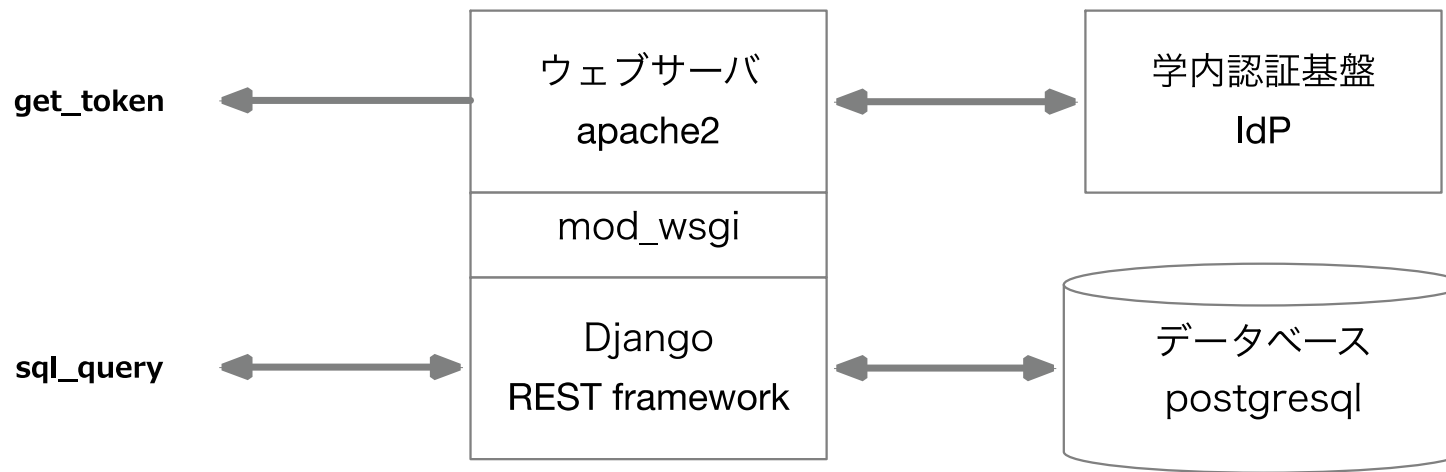
The screenshot shows the pgAdmin 4 web interface. The left sidebar displays a tree view of the database structure, with 'public' expanded to show '1.3 シーケンス' and '28 テーブル'. The '科目成績情報' table is selected. The main panel shows the 'クエリエディタ' (Query Editor) with a SQL query: `SELECT * FROM public."科目成績情報" ORDER BY ...`. The 'クエリの履歴' (Query History) tab is active, showing a list of executed queries. The bottom panel displays the 'データ出力' (Data Output) tab, showing a table of data with columns: 匿名ID, 学生所属コード, 学年, 現況区分コード, 科目コード, 認定年度, 認定学期区分コード, 認定学年, 必選区分コード, 評価, 評語点, 評語コード. The '匿名ID' column is circled in red.

匿名ID	学生所属コード	学年	現況区分コード	科目コード	認定年度	認定学期区分コード	認定学年	必選区分コード	評価	評語点	評語コード
1	02RVVLSYUN	46316	4	AA010600	2017		1	1	4	80	B
2	02RVVLSYUN	46316	4	AA180300	2017		1	1	4	75	C
3	02RVVLSYUN	46316	4	AG100100	2017		1	1	1	78	C
4	02RVVLSYUN	46316	4	AG200100	2017		1	1	1	93	A
5	02RVVLSYUN	46316	4	AG300100	2017		1	1	1	S	S
6	02RVVLSYUN	46316	4	AJ005100	2017		1	1	1	89	B
7	02RVVLSYUN	46316	4	AJ005200	2017		2	1	1	90	A
8	02RVVLSYUN	46316	4	AJ501500	2017		1	1	1	85	B
9	02RVVLSYUN	46316	4	AP200200	2017		2	1	5	90	A
10	02RVVLSYUN	46316	4	AP401100	2017		2	1	1	83	B
11	02RVVLSYUN	46316	4	AP631100	2017		1	1	5	90	A
12	02RVVLSYUN	46316	4	AP631200	2017		1	1	5	75	C

- R5年度の予定

- 大学院生のデータと紐付け（進学した学生の調査が出来るように）
- デジタルユニバーシティとの調整
- 利用しやすいフロントエンドサーバの構築
- データ利用に関する諸規定の検討

データベースサーバの構成



- データベースはデータベースサーバにより保持されており，WebAPI を介して，SQL出アクセスできる．
- WebAPI のアクセスには認証トークンが必要

Python によるデータベースへのアクセス

```
# -*- coding: utf-8 -*-  
import learning_analytics as la # モジュールのインポート  
import pandas as pd  
  
# トークンの設定  
token = 'your token's here'  
  
# SQL 文の設定  
sql = 'SELECT * FROM public.m_admissions;'  
  
# データベースサーバへクエリの実行  
result = la.sql_query( sql, token )
```