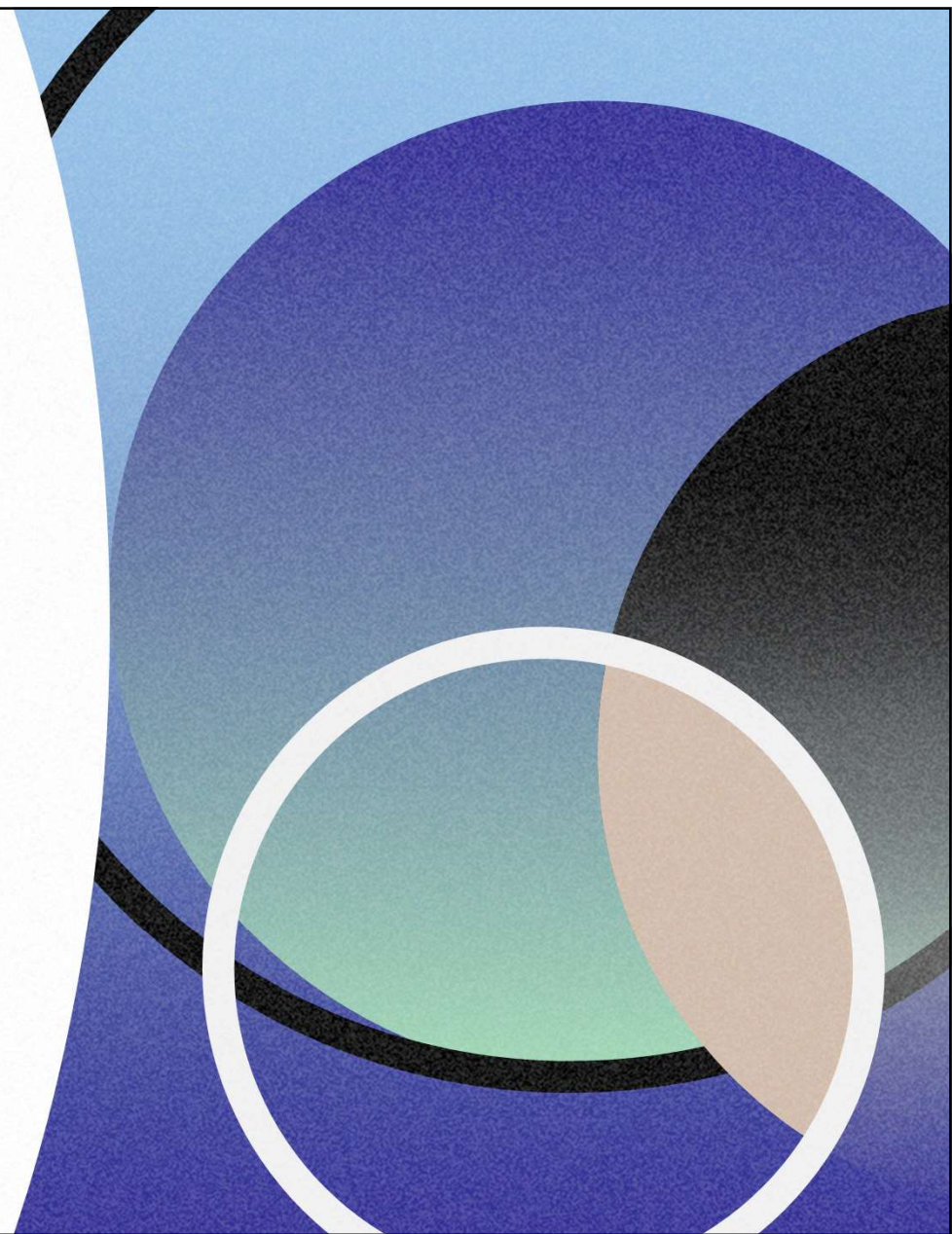




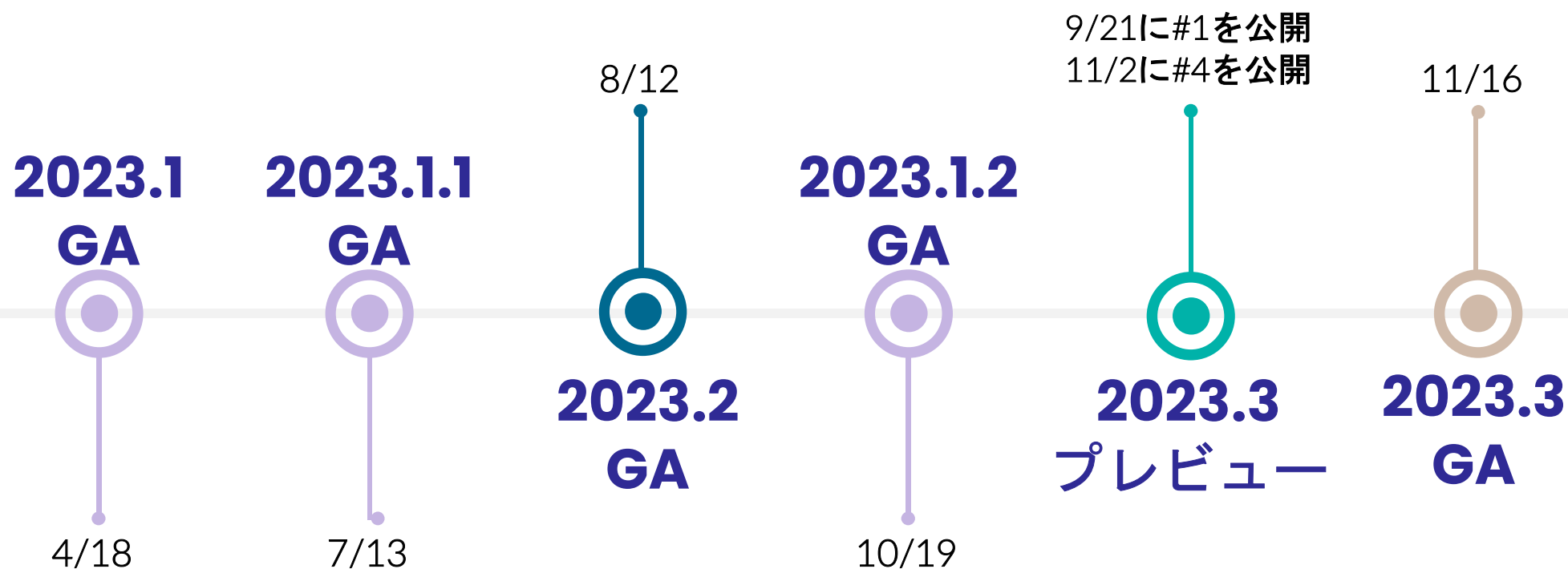
IRISの最新機能

インターシステムズジャパン株式会社
セールスエンジニア
秦 信之

2023年11月17日



IRIS 2023.Xリリースカレンダー



アジェンダ

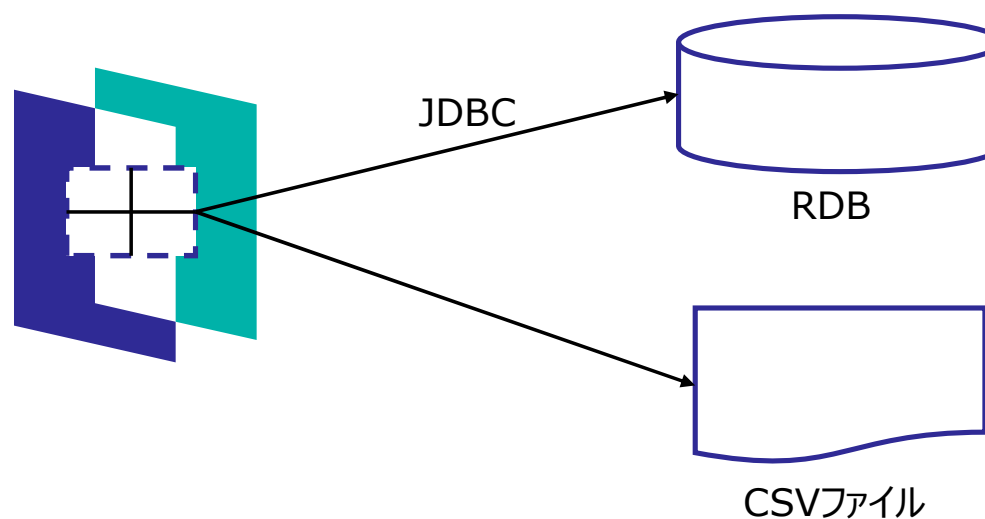
- 1 **Foreign Table**
- 2 **Federated Table**
- 3 **カラムナー事例**
- 4 **Integrated ML 時系列モデル**
- 5 **Farewell!?**

A vertical white line with small circular dots at both ends, positioned on the left side of the slide.

Foreign Table

Foreign Table 概要

- 外部テーブル
 - 外部のサーバ（Foreign Server）に格納されているデータを参照するテーブル
- 利用場面
 - 高い更新頻度かつ最新を参照したい場合
 - 利用頻度が低い巨大なデータがファイルに格納されている



Foreign Table について

- 実験的な機能 (Experimental Feature)
- SQLのSELECT文アクセスのみ
 - CREATE, UPDATE, DELETE, オブジェクトアクセス不可
- Javaが必要
- JDBCおよびCSVファイルをサポート
- CSVファイルはIRISインスタンスからOSレベルでアクセス可能
- CSV以外のファイル (JSON等)
 - IRIS内に格納して下さい
- クエリ文のWHERE句
 - 等価、比較等の簡単な演算子は外部サーバ宛てのクエリに転送
- インスタンス内部のテーブルとJOIN可能

Foreign Table Step-By-Step

1. 管理ポータルで「SQLゲートウェイ接続」を作成

システム > 構成 > SQLゲートウェイ接続

SQLゲートウェイ接続

新規接続作成

オブジェクト/SQLゲートウェイ接続は、外部APIやデータソースとの接続します。現在、以下のゲートウェイ接続が定義されています:

システム > 構成 > SQLゲートウェイ接続 > ゲートウェイ接続

SQLゲートウェイ接続

以下のフォームでゲートウェイ接続を編集します:

接続の種類:	JDBC
接続名:	LOCALMSSQL
ユーザ:	irisql
パスワード:	*****
ドライバ名:	com.microsoft.sqlserver.jdbc.SQLServerDriver
URL:	jdbc:sqlserver://localhost/ELECTRIC:1433;encrypt=false
クラスパス:	C:\Localsqljdbc_12.4\jpn\jars\mssql-jdbc-12.4.1.jre11.jar (複数のjarファイルが必要な場合はカンマで区切ったリストにできます。)
プロパティ:	encrypt=false
デフォルトで区切り識別子を使用しない:	<input checked="" type="checkbox"/>
COALESCE使用:	<input checked="" type="checkbox"/>
IFNULL() の代わりに NVL() を使用する:	<input type="checkbox"/>
複合Row ID内で変換:	<input checked="" type="radio"/> 非文字値を変換しない <input type="radio"/> CAST を VARCHAR として使用 <input type="radio"/> CAST を CHAR として使用 <input type="radio"/> {fn convert ...} を使用
<div>テスト接続 保存 キャンセル</div>	

Foreign Table Step-By-Step

2. SQLで外部サーバ (Foreign Server)を作成

```
CREATE FOREIGN SERVER Test.MSSQL FOREIGN DATA WRAPPER JDBC CONNECTION '接続名'  
CREATE FOREIGN SERVER Test.CsvDir FOREIGN DATA WRAPPER CSV HOST '/data/files'
```

3. SQLで外部テーブル (Foreign Table)を定義

```
CREATE FOREIGN TABLE ff_root (rootId INT, message VARCHAR(32), value INT) SERVER Test.MSSQL TABLE 'iris.root'  
CREATE FOREIGN TABLE ff_root (rootId INT, message VARCHAR(32), value INT)  
    SERVER Test.MSSQL QUERY 'select * from iris.root'  
CREATE FOREIGN TABLE (  
    firstName VARCHAR(15),  
    lastName VARCHAR(15),  
    DOB DATE  
) Sample.Person SERVER Test.CsvDir FILE 'person.csv' USING { "from": { "file": { "header": true } } }
```

4. Run Query!

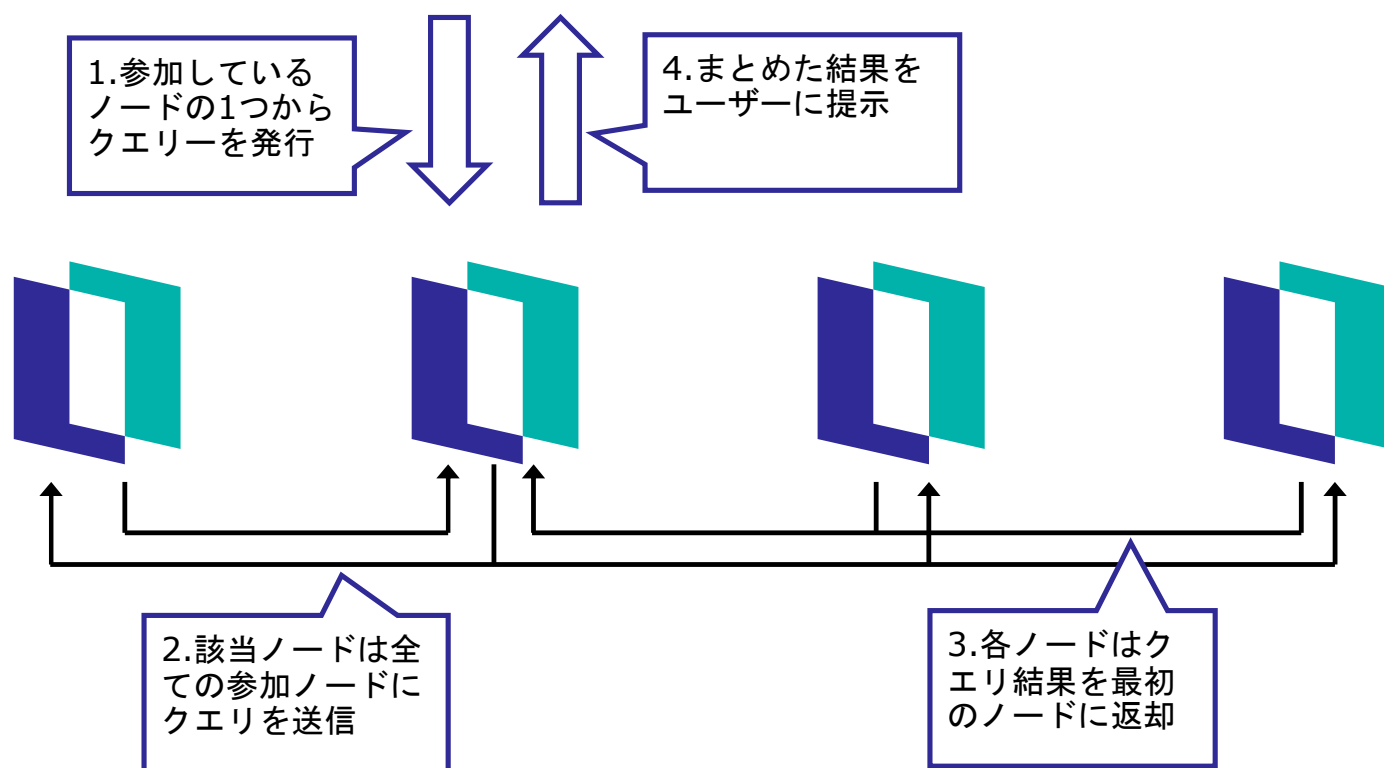
A vertical white line with small circular dots at both ends, positioned on the left side of the slide.

Federated Table

Federated Table 概要

- Federated Table

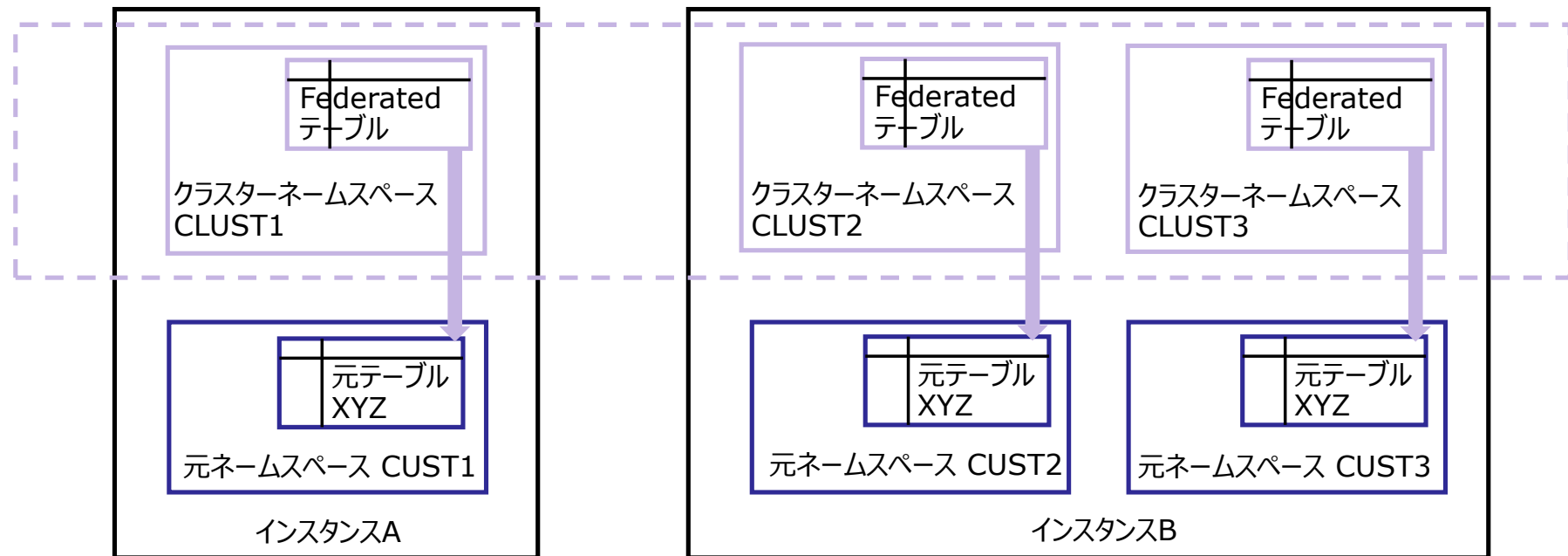
- 同一もしくは似たスキーマのテーブルを管理している複数のIRISインスタンスにまたがってクエリを発行する仕組み



Federated Table について

- 実験的な機能 (Experimental Feature)
- 利用場面
 - 複数サイトのデータを横断的に分析
 - マルチテナント環境で統合して分析
- 事前準備
 - 対象のテーブルがある全インスタンスにまたがるシャードクラスター
 - 対象のテーブルがあるネームスペース毎にシャードネームスペース
- シャードテーブルとの違い
 - 元テーブルには影響しない
 - 複数テーブルの共通部分を投影した読み取り専用テーブル
 - データを自動分散しない
 - 各インスタンスが自身のデータを管理

Federated Table について



Federated Table Step-By-Step

1. Federated Tableを定義

`$SYSTEM.Sharding.CreateFederatedTable(shardNS, fedTable, sourceNS, srcTable, colList)`

- 実行直後より全シャードネームスペースからsrcTableのデータを参照可能

shardNS: Federated Table用に作成したネームスペース

fedTable: Federated Table名

sourceNS: 元テーブルが存在するネームスペース

srcTable: 元テーブル名

colList: Federated Tableと元テーブル間のカラムマッピングをlist形式で指定、省略された場合は元テーブルと同一

2. 他のテーブルをFederated Tableに接続

`$SYSTEM.Sharding.ConnectFederatedTable(shardNS, fedTable, sourceNS, srcTable, colList)`

3. Run Query!



カラムナー事例

カラムナー事例

- システム概要
 - 医療用DWH
 - 対象テーブル: 120フィールド（患者情報、診察・検査情報、請求情報等）
 - データ件数: 約1億
 - ハードウェア: 192core / 521GB RAM / NVMe SSD
 - IRIS 2023.3プレビュー, Linux
- レビュー: カラムナーが有効なユースケース
 - レコード数が100万以上
 - トランザクションシステムのOLAP、かつ性能に満足していない場合
 - カラムナーインデックスの適用を検討
 - データウェアハウス、BI等の分析が主なユースケースの場合
 - カラムナーストレージの適用を検討

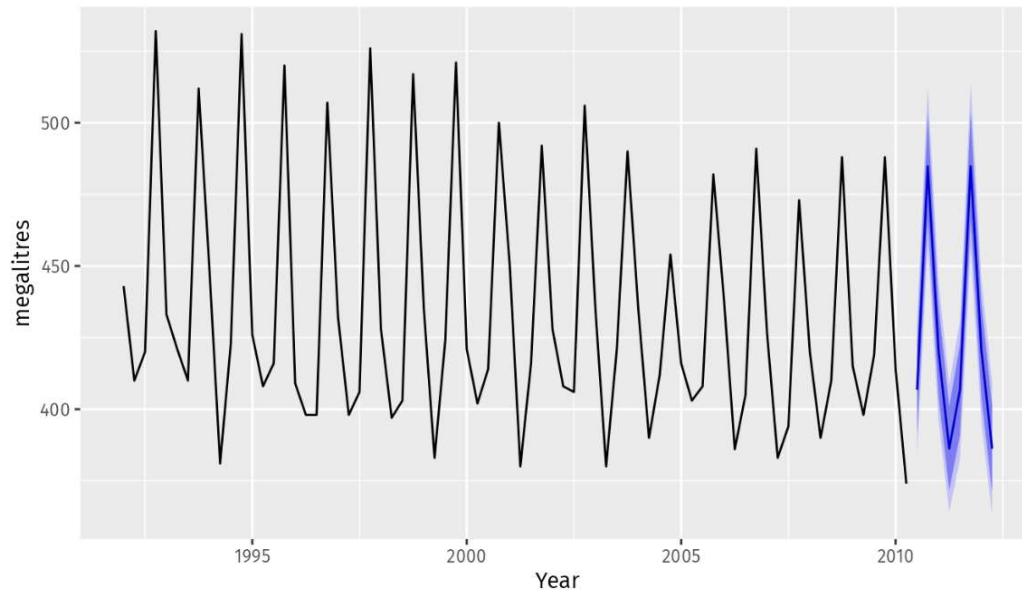
カラムナー事例

このテーブルは現地での投影のみとさせていただきます



Integrated ML 時系列モデル

時系列データとは



Australian quarterly beer production: 1992Q1–2010Q2, with two years of forecasts.

[出展](#)

[1.4 Forecasting data and methods | Forecasting: Principles and Practice \(2nd ed\) \(otexts.com\)](#)

- 時間の経過順に並んだ過去の数値データ
- 例
 - 株価
 - 降水量
 - 売上
- 主な3つの要因
 - 長期的なトレンド
 - 季節的な変動
 - その他
- 主なモデル
 - 回帰モデル
 - 今日までのデータを元に予測
 - 移動平均モデル
 - 過去の平均および予測と実績の誤差を元に予測

Integrated ML 時系列モデル

- 実験的な機能 (Experimental Feature) 2023.3から通常の機能となりました
- 時系列データに基づき将来値を予測するモデルをサポート

元データ: tsdata

Date1	Value
2023/11/14	150.0
2023/11/15	149.5
2023/11/16	150.4

予測結果

Date1	Value
2023/11/14	150.0
2023/11/15	149.5
2023/11/16	150.4
2023/11/17	????

- 予測結果は新しい行としてクエリーに返されます
- Integrated ML 時系列データ Step-By-Step

1. モデル作成

```
CREATE TIME SERIES MODEL forecastUsdJpy PREDICTING(*) BY (Date1) FROM tsdata USING {"forward":5}
```

2. 学習

```
TRAIN MODEL forecastUsdJpy
```

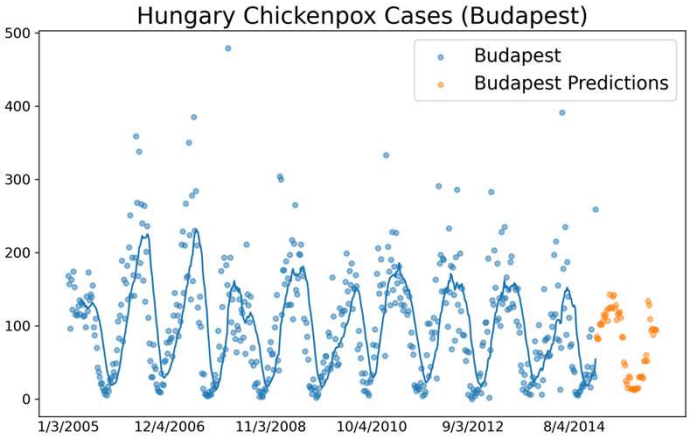
3. 結果取得

```
SELECT WITH PREDICTIONS(forecastUsdJpy) * from tsdata
```

Integrated ML 時系列モデル

ハンガリーにて水疱瘡の地域別症例数を予測

```
CREATE TIME SERIES MODEL hungary60
  PREDICTING (*) BY (DATE1)
  FROM chickenpox_training
  USING { "Forward" : 60 }
```



* 現在はAutoMLプロバイダーのみ、DataRobotとH2Oサポートを追加予定

```
SELECT WITH PREDICTIONS (hungary60) *
  FROM chickenpox_test
```

元データ

予測

ID	DATE1	BACS	BUDAPEST	FEJER	HAJDU
515	11/10/14	10	33	20	3
516	11/17/14	26	85	28	9
517	11/24/14	14	16	9	6
518	12/1/14	41	95	16	14
519	12/8/14	31	43	2	25
520	12/15/14	15	35	7	4
521	12/22/14	8	30	1	10
522	12/29/14	49	259	11	61
	1/11/15	35	85	14	27
	1/18/15	39	82	14	31
	1/25/15	39	82	18	32
	2/1/15	33	102	20	39
	2/8/15	34	103	22	41
	2/15/15	31	101	26	43
	2/22/15	45	106	29	40



Farewell !?

製品ライフサイクル関連

- 非推奨 プライベートWebサーバ
 - PWSを新規にインストールされません
 - Apache/IISがインストールされていれば自動的にIRIS接続を設定
- 非推奨 スタジオ
 - VS Codeへ移行を
- 非推奨 System Alerting and Monitoring (SAM)
- 非推奨 InterSystems Cloud Manager (ICM)
 - KubernetesおよびInterSystems Kubernetes Operator (IKO) へ移行を
- サポートプラットフォーム変更
 - Windows 2012サーバは2023.2よりサポート対象外

ありがとう
ございました