



# DeepSeeのご紹介

インターシステムズジャパン株式会社

© 2016 InterSystems Corporation. All rights reserved.



# DeepSeeとは?

- 以下のことが可能な強力なテクノロジーの集まり ...
  - トランザクションデータへ直接分析的なアクセスを提供
  - 分析を主要なブレークスルー機能として届けることで既存アプリケーションの価値を高める
  - トランザクションデータのみならず全てのデータを分析

THE KEYS TO  
BREAKTHROUGH  
APPLICATIONS



# 利点

- 組み込み
  - ダッシュボード
  - アプリケーションプログラマーインタフェース
  - MDX – Multi Dimensional eXpressions
  - リアルタイムかジャストインタイムか
  - 構造データ、非構造データ両方に対して分析
  - データのアクティブ分析
  - 事前ビルドしたダッシュボードは熟練ユーザーでなくともDeepSeeの分析機能に簡単にユーザーフレンドリーなアクセスを提供
  - 可動部分が少ない Caché & Ensemble

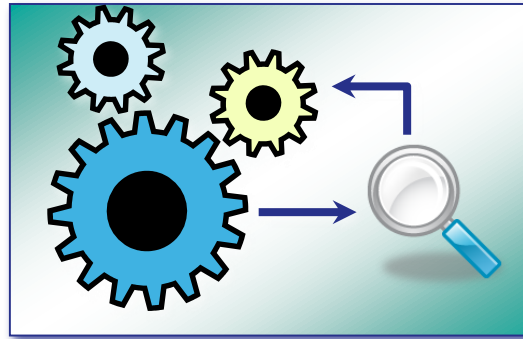


# アクティブ分析

分析は、アプリケーションに行動を引き起こすための不可欠な要素

業務上の意思決定に必要な全てのデータにアクセス

組み込みリアルタイム  
ビジネスインテリジェンス



# DeepSee主要な機能

アナライザー

データを探索、表示

アーキテクト

データモデルを定義

ユーザー  
ポータル

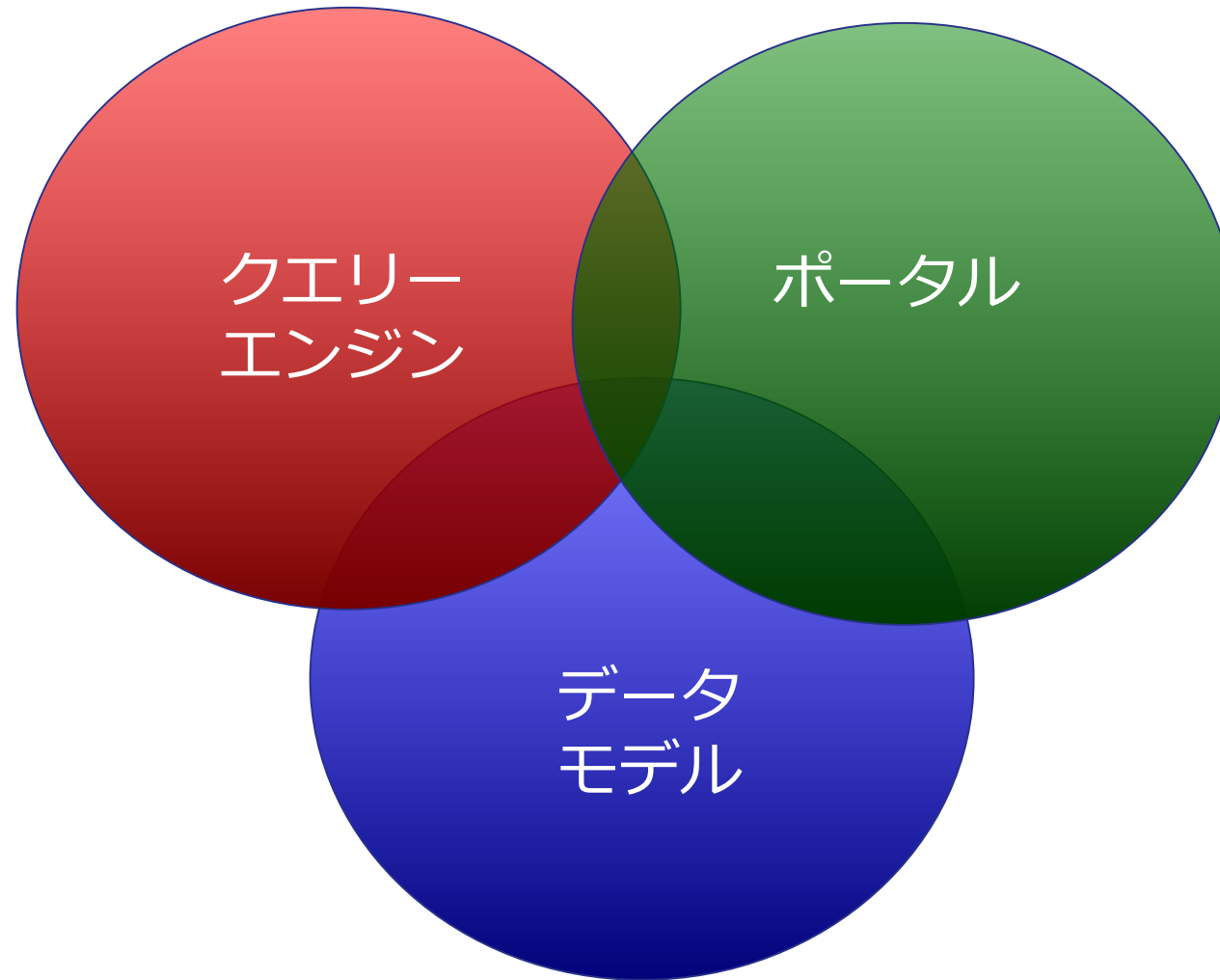
アプリケーションを拡張してダッシュボードを作成、行動を引き起こす

コネクター

外部データを使用



# DeepSeeは3つの部分で構成





# データモデル

- キューブ
  - ディメンジョンとメジャーを定義
- サブジェクトエリア
  - キューブ上のビュー
  - 自動フィルタリングを提供
- KPI
  - ダッシュボードで利用できるもっと洗練された経営指標を作る
  - DeepSee、SQL、カスタムロジック使用可能



# 実行エンジン

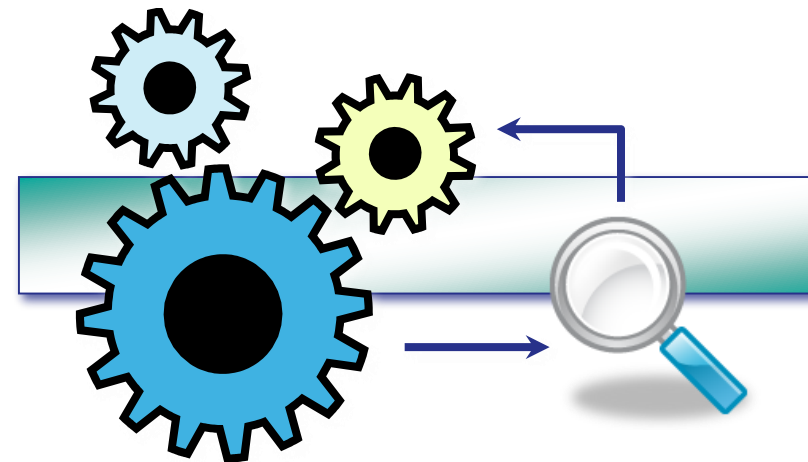
- ネイティブインデックス構造を基礎としたクリーンで無駄のないエンジン
- アプリケーションがDeepSeeの機能を利用できるResultSet APIが使える
- MDXクエリー言語のサポート
  - MDX標準の大きなサブセット
  - より完全な実装が必要
  - 拡張により機能を強化





# DeepSeeクエリーエンジン

- アナライザー
  - ビジュアルなクエリー作成
  - クエリーは保存可
- DeepSeeクエリーツール
  - ブラウザからMDXクエリー
- DeepSeeシェル
  - ターミナルからMDXクエリー
- DeepSee ResultSet
  - プログラミングAPI



# DeepSee性能とスケーラビリティ

- 大量データモデル(1億行を超えるファクト表)のサポートのため多レベル、インクリメンタルなキャッシング
- マルチコアアーキテクチャをうまく使うためにクエリーの並列実行をサポート:
  - クエリーはファクト数で分ける
  - クエリーはセル数で分ける
  - サブクエリーと結合
- キューブへの効率の良いインクリメンタルな更新のサポート



# DeepSeeユーザーインタフェース

- DeepSeeエンジンとDeepSee UIのきれいな分離
  - DeepSeeエンジンにアクセスするAPI利用可能
- Zenテクノロジーを使って構築
  - 複数ブラウザーのサポート
  - 標準Zenコンポーネント
  - DeepSee API使用

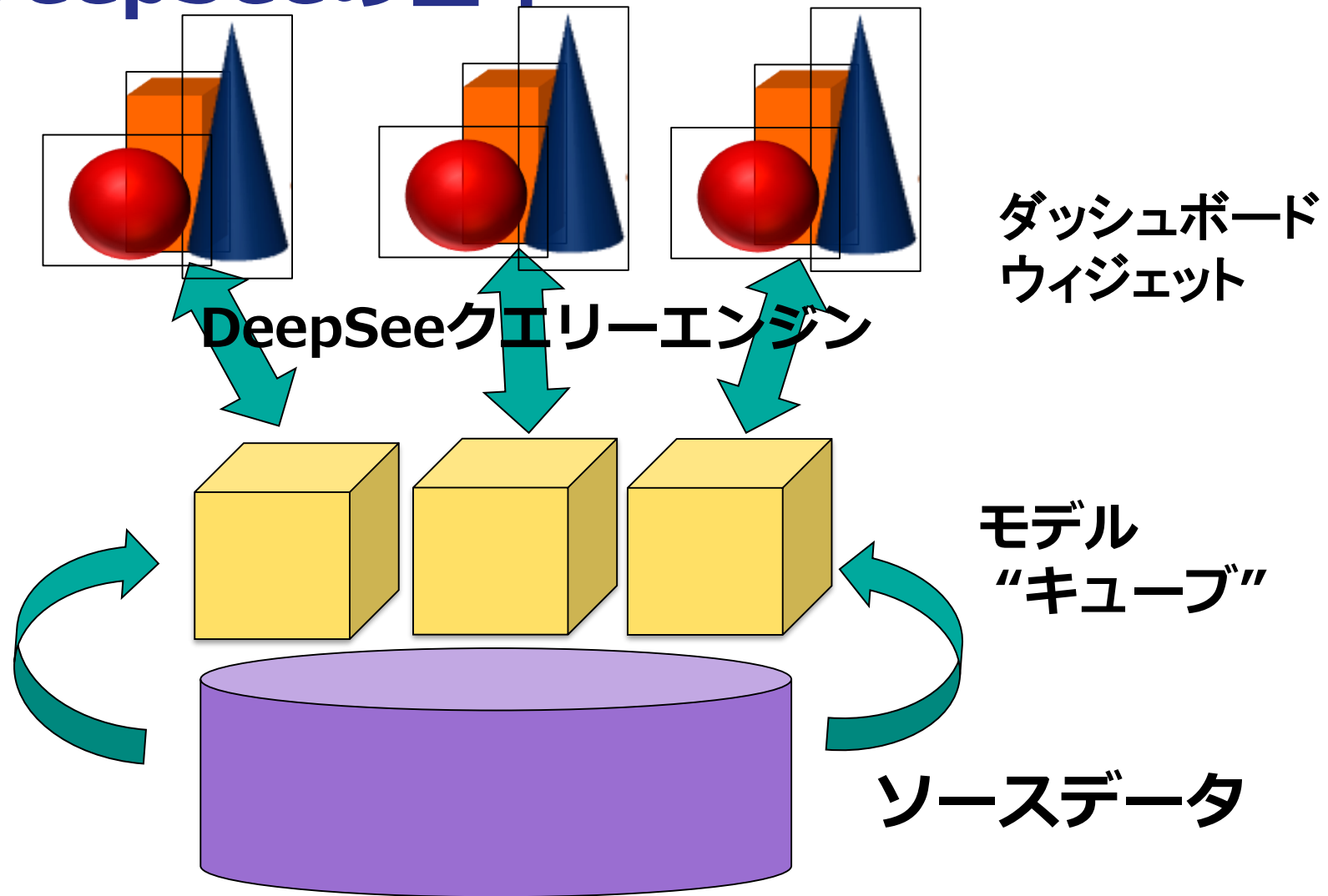


# DeepSeeポータル

- ユーザーポータル
  - ダッシュボード、レポート、保存したクエリーを見るためのクリーンで簡単な方法
- ダッシュボード
  - 保存したクエリー、KPIのデータを表示する
  - ビジュアルな “ウィジェット” でデータを表示
  - ダッシュボードのアクションは、アプリケーションロジックと連携可能

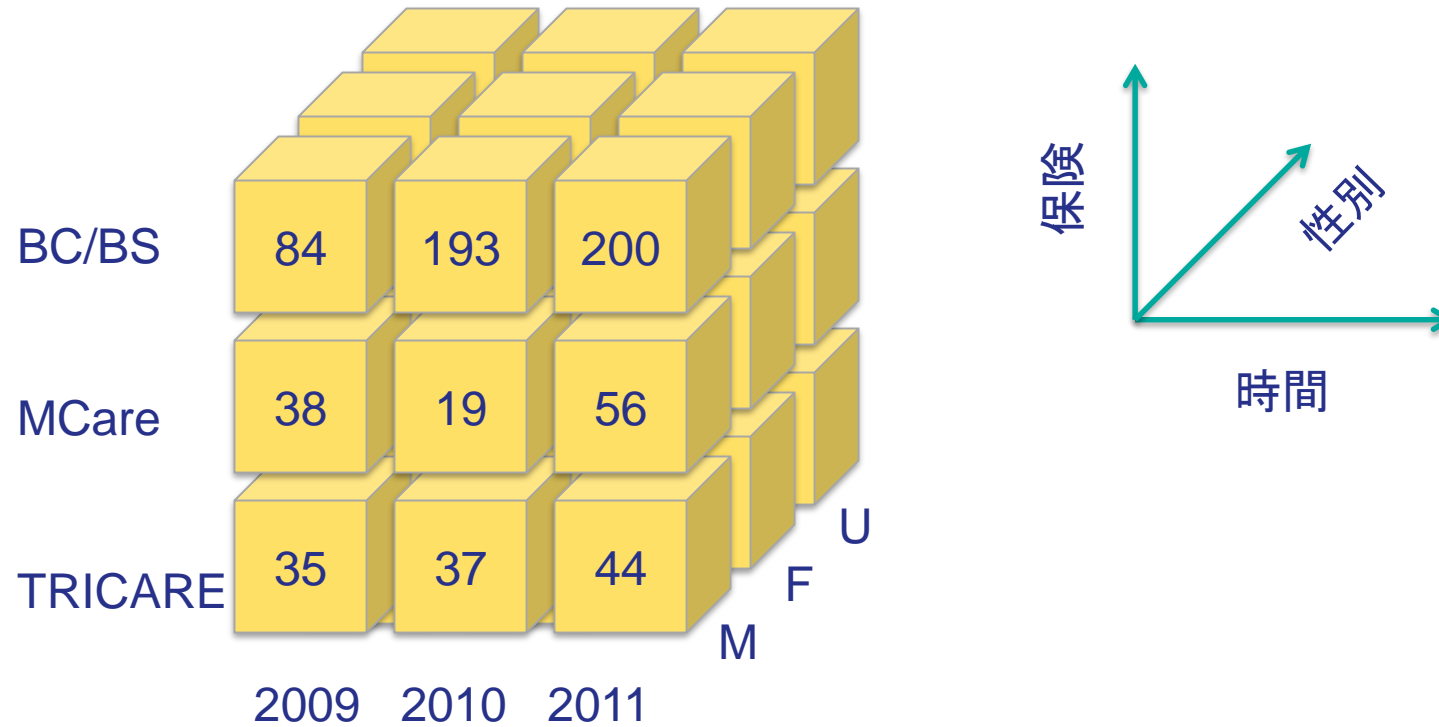


# モデルはDeepSeeの基本



# キューブ

DeepSeeキューブは異なる観点で分析するためのファクト表を構成



# DeepSeeクエリー

- DeepSeeクエリーは MDX (Multi-Dimensional Expressions)を使用
  - MDXはキューブ検索用、SQLはテーブル検索用

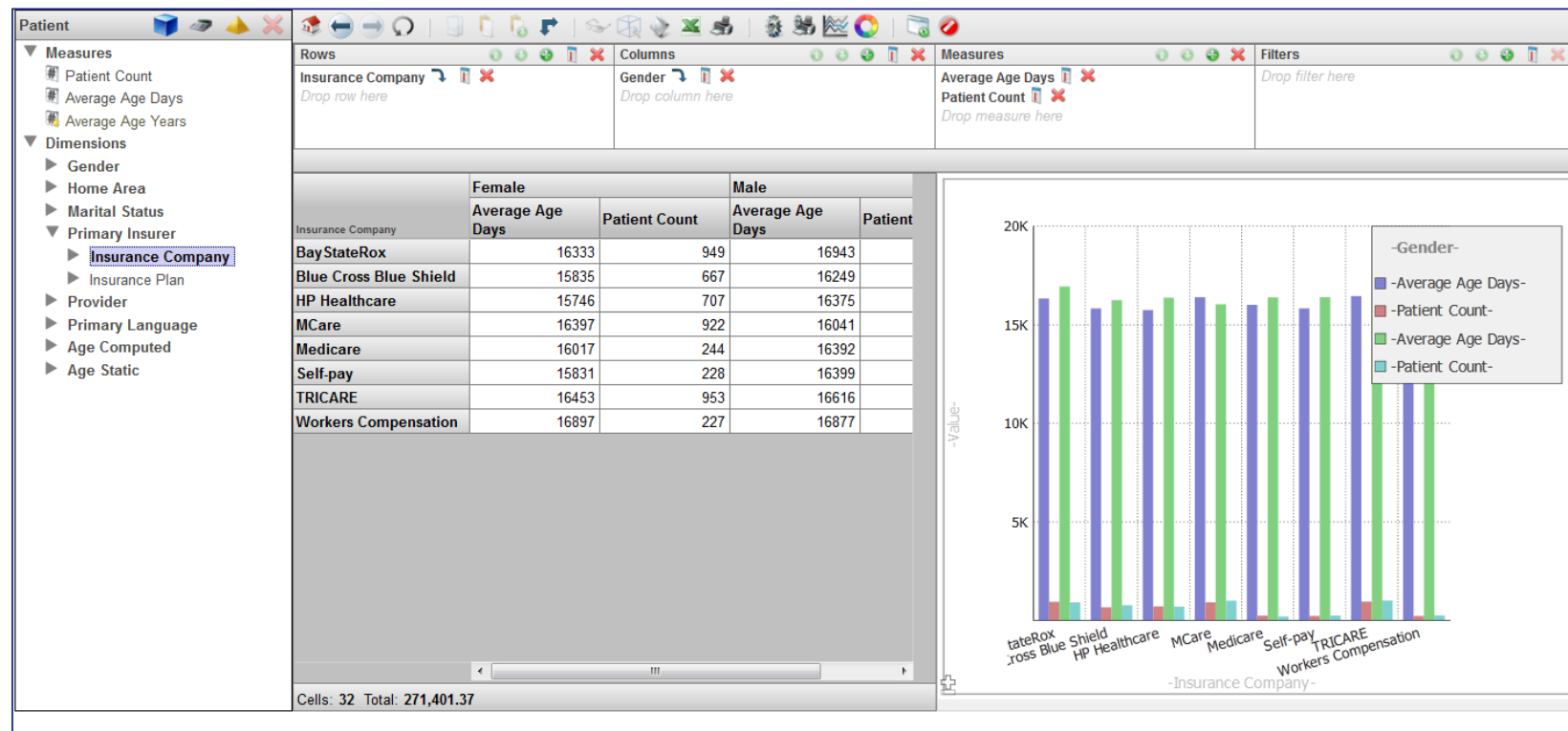
```
SELECT NON EMPTY NONEMPTYCROSSJOIN  
([Gender].[H1].[Gender].Members,{[Measures].[Average Age  
Days],[Measures].[%COUNT]}) ON 0,NON EMPTY [Primary  
Insurer].[H1].[Insurance Plan].Members ON 1 FROM [Patient]
```





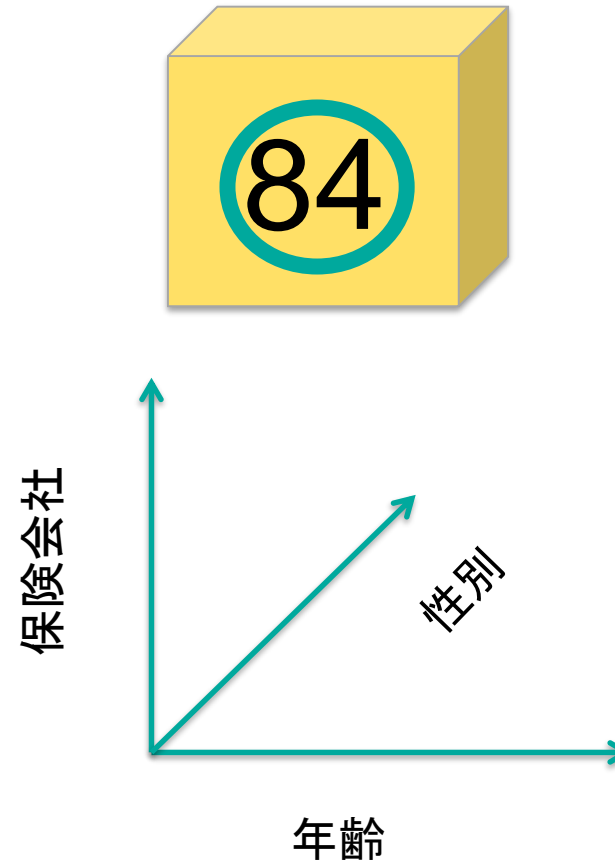
# アナライザー

- ビジュアルクエリツール
  - ピボットテーブルとキューブを使ったチャートを作成



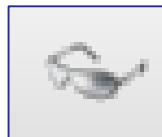
# キューブの要素

- メジャー
  - レコードセットの集計値
    - ・ 患者数
    - ・ 患者平均年齢
- ディメンジョン
  - データ分析の視点
    - ・ 保険会社
    - ・ 性別
    - ・ 年齢



# キューブの要素（続き）

- 詳細リスト
  - ピボットテーブルのセルで表現しているソースレコードの詳細
- リストを見るには:
  - ピボットテーブルのセルをクリック
  - “メガネ”アイコンをクリック

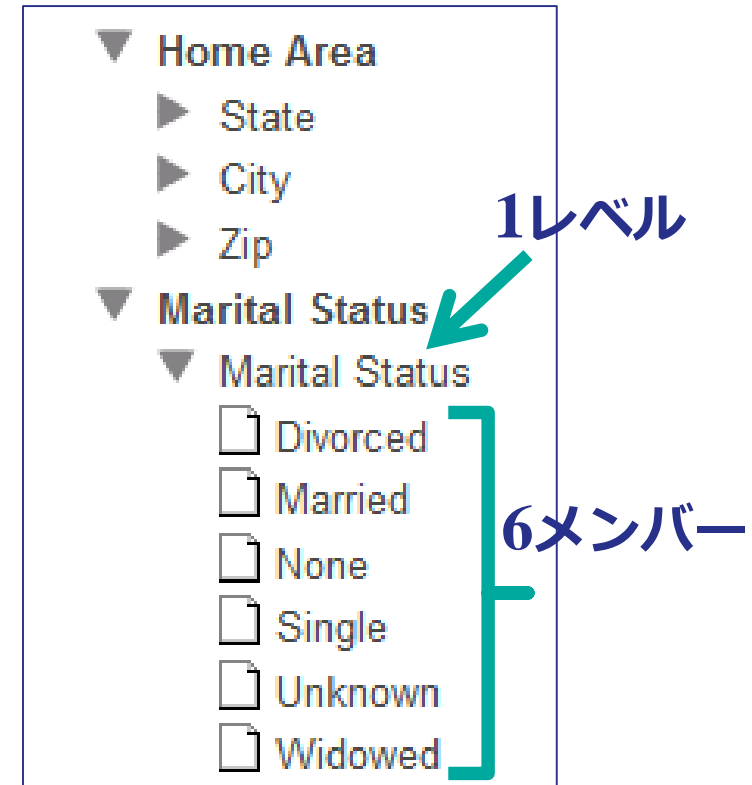


<

# ディメンジョンの構造

- ディメンジョンの構造というMDX要件がある
  - ディメンジョン は1つ以上の階層を持つ
  - 階層は1つ以上のレベルを持つ
  - レベルはメンバーを含む
  - メンバーはそのレベルで可能な値のリスト

アナライザー



# 値のソース

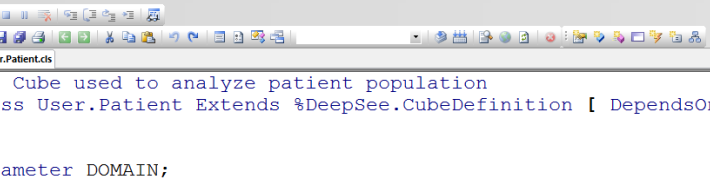
- ソースプロパティ
  - ソースクラスのプロパティから直接
- ソース表現
  - ソースクラスのプロパティや補助関数を含むCache ObjectScript (COS)表現
- **計算メンバー**
  - 他キューブ要素を含むMDX表現 (ディメンジョンとメジャー)



# キューブ定義はクラス

# スタジオ

## キューブ定義



```
/// Cube used to analyze patient population
Class User.Patient Extends %DeepSee.CubeDefinition [ DependsOn = Dat
{
    Parameter DOMAIN;

    /// Cube definition from Architect.
    %XData Cube [ XMLNamespace = "http://www.intersystems.com/deepsee" ]
    {
        %cube xmlns="http://www.intersystems.com/deepsee" name="Patient" di
        <dimension name="Gender" disabled="false" hasAll="false" allCapt
        <hierarchy name="H1" disabled="false">
            <level sourceExpression="$Case(%source.Gender, &quot;F&quot;
            </level>
        </hierarchy>
        </dimension>
        <dimension name="Home Area" disabled="false" hasAll="false" allC
```

```
// Cube used to analyze patient population
Class User.Patient Extends %DeepSee.CubeDefinition [ DependsOn = Data.Patient, ProcedureBlock ]
{

Parameter DOMAIN;

// Cube definition from Architect.
%XData Cube [ XMLNamespace = "http://www.intersystems.com/deepsee" ]
{
=<cube xmlns="http://www.intersystems.com/deepsee" name="Patient" displayName="Patient" description="Cube used to analyze patient population"
=<dimension name="Gender" disabled="false" hasAll="false" allCaption="All Gender" allDisplayName="Gender" type="C">
=<hierarchy name="H1" disabled="false">
=<level sourceExpression="%Case(%source.Gender,%quote,F&quot;%quote;%quote;Female&quot;%quote;%quote;Male&quot;%quote;%quote;Male&quot;%quote;%quote)">
=</level>
=</hierarchy>
=</dimension>
=<dimension name="Home Area" disabled="false" hasAll="false" allCaption="All Home Area" allDisplayName="Home Area">
=<hierarchy name="H1" disabled="false">
=<level sourceProperty="HomeAddress.State" name="State" disabled="false" timeFunction="Year" list="false" useList="true">
=</level>
=<level sourceProperty="HomeAddress.City" name="City" disabled="false" timeFunction="Year" list="false" useList="true">
=</level>
=<level sourceProperty="HomeAddress.Zip" name="Zip" disabled="false" timeFunction="Year" list="false" useList="true">
=</level>
=</hierarchy>
=</dimension>
=<dimension name="Marital Status" disabled="false" hasAll="false" allCaption="All Marital Status" allDisplayName="Marital Status">
=<hierarchy name="H1" disabled="false">
```

# アーキテクト

View

☐ Home
 ☒ New
 ☐ Open
 ☐ Save
 ☐ Compile
 ☐ Build
 ☐ Documentation

Architect

Source Class	Model Elements	Add Element	Undo	Expand All	Collapse All	Render	
<input checked="" type="checkbox"/> Data Patient	<b>Patient</b>						
<input type="checkbox"/> SId	<b>Measurements</b>						
<input type="checkbox"/> DOB	Average Age Days					measure	AVG DOB ✗
<input type="checkbox"/> Gender	<b>Dimensions</b>						
<input checked="" type="checkbox"/> HomeAddress	<b>Gender</b>					data dimension	✗
<input type="checkbox"/> HomePhone	H1					hierarchy	✗
<input type="checkbox"/> MaritalStatus	Gender					level 1 (expression)	✗
<input type="checkbox"/> MobilePhone	<b>Home Area</b>					data dimension	✗
<input type="checkbox"/> Name	H1					hierarchy	✗
<input checked="" type="checkbox"/> PCP	State					level 1	✗ HomeAddress.State ✓✓
<input type="checkbox"/> PatientID	City					level 2	✗ HomeAddress.City ✓✓
<input type="checkbox"/> PrimaryInsurance	Zip					level 3	✗ HomeAddress.Zip ✓✓
<input type="checkbox"/> PrimaryLanguage	<b>Marital Status</b>					data dimension	✗
<input type="checkbox"/> SSN	H1					hierarchy	✗
<input checked="" type="checkbox"/> WorkAddress	Marital Status					level 1	✗ MaritalStatus ✓
<input type="checkbox"/> WorkPhone	<b>Primary Insurer</b>					data dimension	✗
	H1					hierarchy	✗
	Insurance Company					level 1	✗ PrimaryInsurance.Company ✓✓
	Insurance Plan					level 2 (expression)	✗ ✓✓
	<b>Provider</b>					data dimension	✗
	H1					hierarchy	✗
	Practice					level 1	✗ PCP.PracticeName ✓✓
	Provider Name					level 2	✗ PCP ✓✓
	Name					property	✗ PCP.Name ✓✓
	Specialty					property	✗ PCP.SpecialtyName ✓✓
	NPI					property	✗ PCPNPINumber ✓✓
	<b>Primary Language</b>					data dimension	✗
	H1					hierarchy	✗
	Language					level 1	✗ PrimaryLanguage ✓
	<b>Age Computed</b>					data dimension	✗
	H1					hierarchy	✗
	Age In Decade					level 1	✗ DOB ✓

**Details** **Tools**

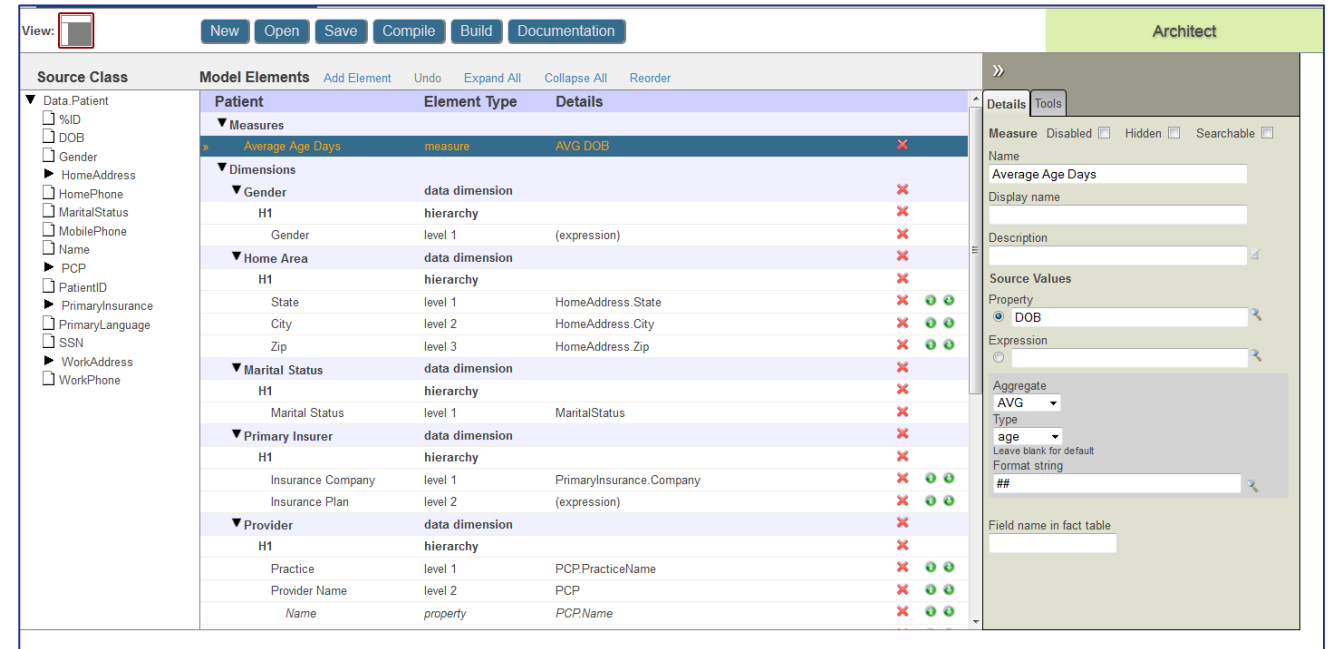
Select an item to view details



# アーキテクト

## キューブを作るためのブラウザーツール

- ドラッグ&ドロップ
- キューブのコンパイルと構築





# キューブの作成

1. アーキテクトを開く
2. ソースクラスを選択
3. 要素 (ディメンジョン、メジャー、リスト)の追加/編集
4. キューブのコンパイルとビルド
5. アナライザーのテスト
6. 3～5を必要に応じて繰り返す



# キューブデザインプロセス

## Source

どんなデータ  
があるか？

## Cube

全てをどのように  
押し込めるか？

## Source

どうやって取り出すか？

## Cube

必要なデータはどれか？



# キューブを最新に維持する: 3オプション

**%ProcessFact**  
1つのソースレコード  
IDのリビルド

**%BuildCube**  
全てのソース  
レコードのリビルド



**%SynchronizeCube**  
全ての“変更済み”ソースレコー  
ドのリビルド



# 何が変わったかがどの様にわかるか？

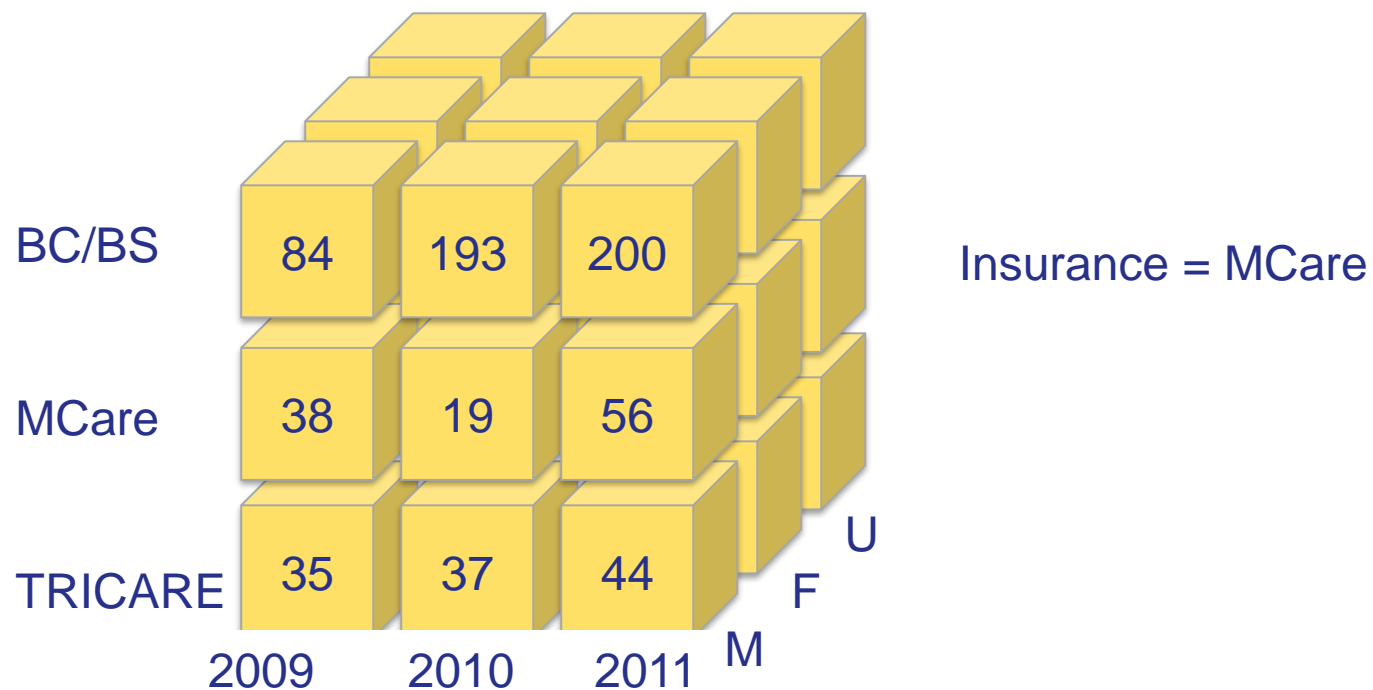
- ソースクラスのDSTIMEパラメータを設定
  - **auto** – %Persistentが自動的に変更されたソースレコードのIDリストを維持管理する (SQLまたは オブジェクト経由で)
  - **manual** – アプリケーション開発者が変更されたソースレコードのIDリストを維持管理するためにAPIを使用する
- %SynchronizeCube
  - IDリストのキューブをリビルド



# サブジェクトエリア

DeepSeeサブジェクトエリアはキューブのサブセットを表現

- 基本キューブのビューをフィルター - 追加ストレージなし
- 基本キューブのメジャーやディメンジョンを隠ぺい、リネーム可能
- 新しいリストの追加や既存のもののリネーム可能



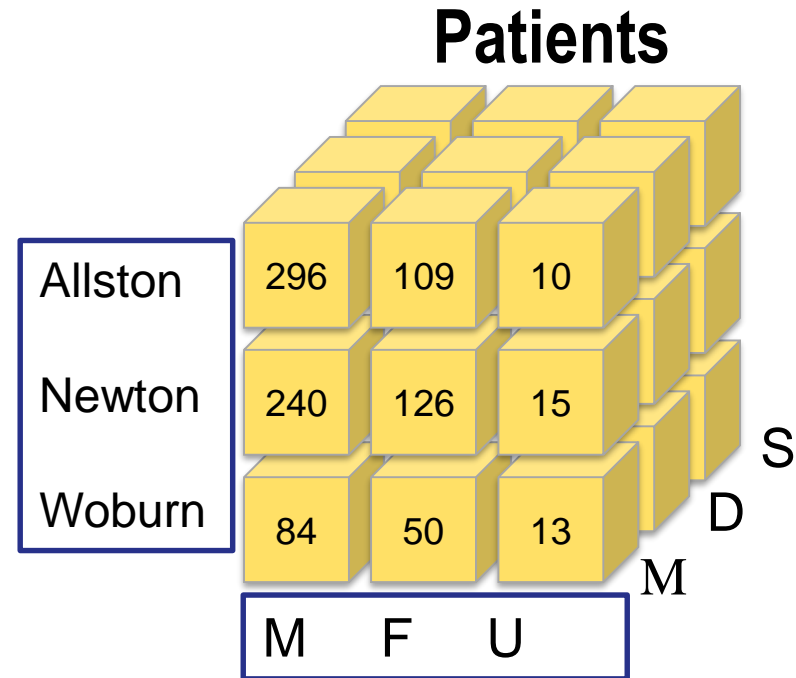
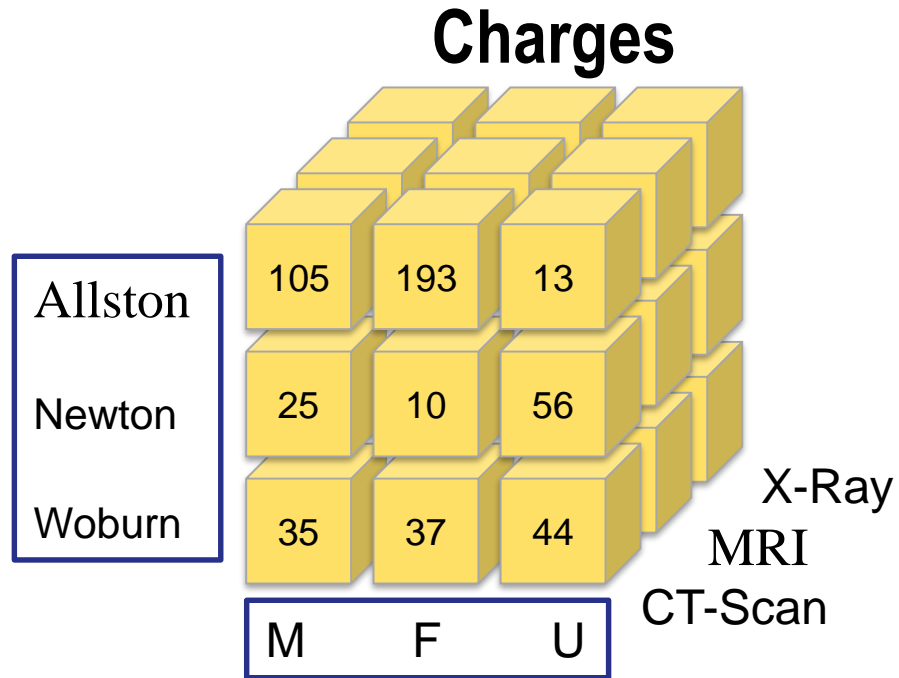
# サブジェクトエリア（続き）

- アーキテクトでサブジェクトエリアを作る
  - ソースキューブのメジャーとディメンジョンを上書き – 削除、新しい名前を提供するなど
  - 詳細リストを追加したり削除したり
  - データを静的にフィルターするためにMDXフィルター表現を提供
- スタジオ:
  - データを動的にフィルターするために %OnGetFilterSpec()を上書き
    - ・ 例：ユーザー名、ロールなどで



# キューブリレーションシツプ

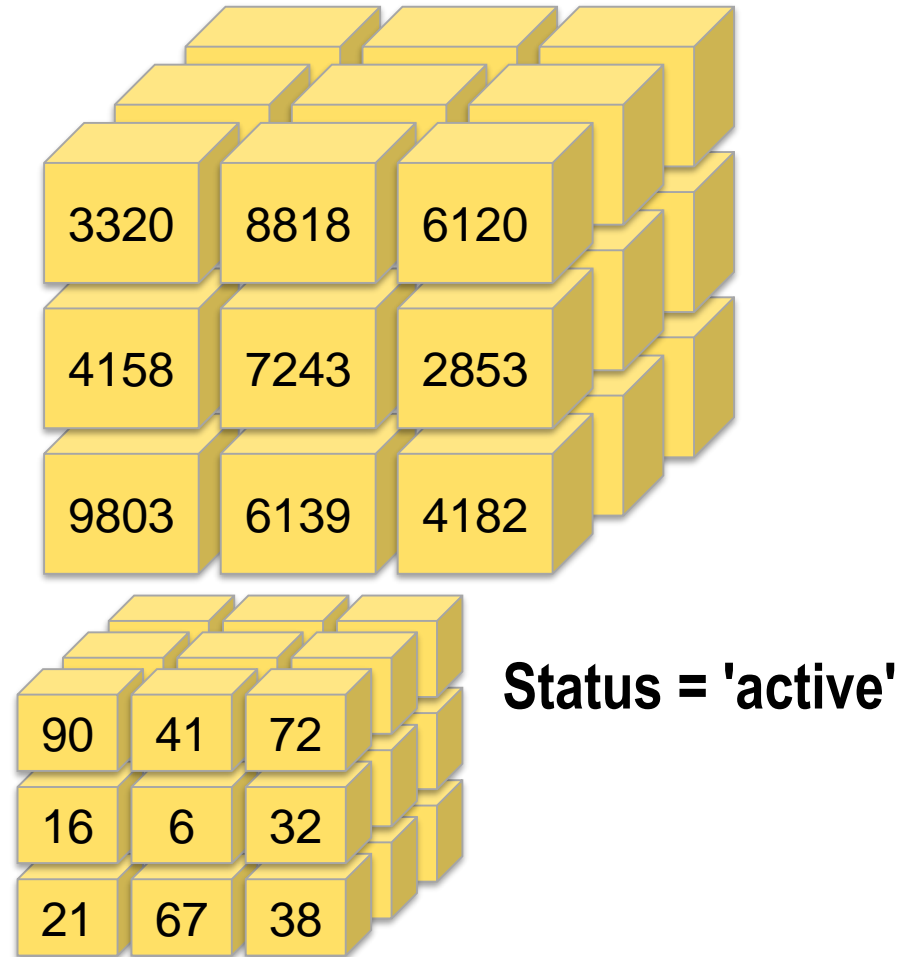
- 関連キューブはディメンジョンとレベルを共有





# ビルド制約

- ソースレコードのサブセットのキューブをビルド
  - SQL 比較表現
- 2つのキューブ:
  - 業務分析には小さいキューブ
- 時系列分析には大きなキューブ



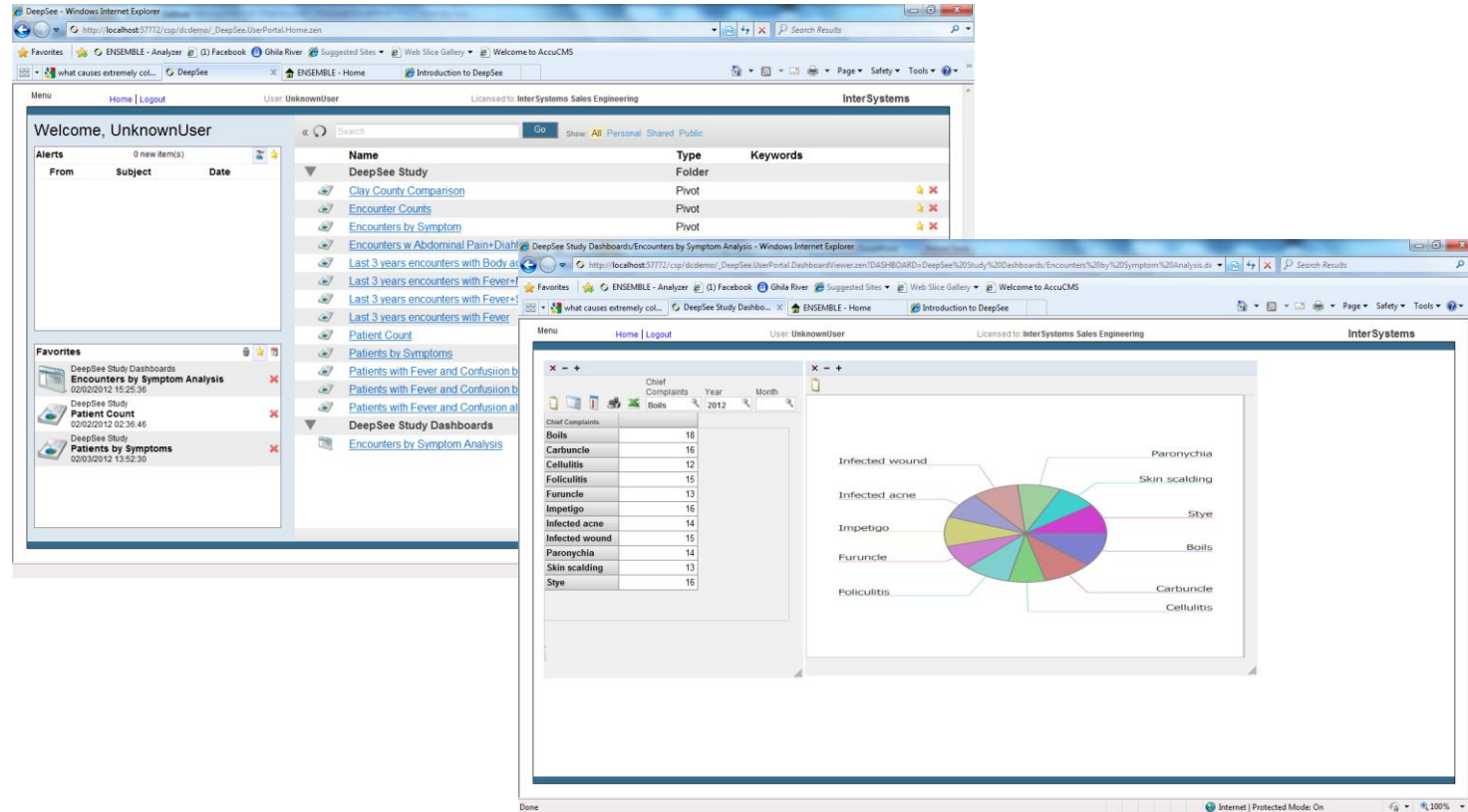
# コネクター

- ソースデータがCaché永続クラスで表現出来ない時にはコネクターを使用
- コネクターはクラス:
  - %DeepSee.DataConnectorを拡張
  - 任意のSQLクエリーやコードの結果をDeepSeeの公開プロパティにマップする
  - DeepSeeキューブのソースクラス



# ポータル

## ■ ユーザーポータル



# KPI定義

## Key Performance Indicator

“目標達成に向けて組織の向上を反映する数量化できるメジャーの集まり”



# DeepSeeプロジェクトステップ

## 1. モデリング

- データモデルやキューブを作る(アーキテクトを使って)



## 2. データソース

- ピボットテーブルを作る(アナライザー)
- KPIを作る (スタジオを使って)**

Product Name	SKU	Unit Price
Budget (Biscuit)	SKU-101	2.95
Budget (Coke)	SKU-102	19.95
Cashew (Biscuit)	SKU-103	29.95
Cashew (Coke)	SKU-104	3.95
Cashew (Biscuit)	SKU-105	2.95
Cashew (Coke)	SKU-106	2.95
Cashew (Biscuit)	SKU-107	2.95
Cashew (Coke)	SKU-108	2.95
Cashew (Biscuit)	SKU-109	2.95
Cashew (Coke)	SKU-110	2.95
Cashew (Biscuit)	SKU-111	2.95
Cashew (Coke)	SKU-112	2.95
Cashew (Biscuit)	SKU-113	2.95
Cashew (Coke)	SKU-114	2.95
Cashew (Biscuit)	SKU-115	2.95
Cashew (Coke)	SKU-116	2.95
Cashew (Biscuit)	SKU-117	2.95
Cashew (Coke)	SKU-118	2.95
Cashew (Biscuit)	SKU-119	2.95
Cashew (Coke)	SKU-120	2.95

Product Name	SKU	Unit Price
Budget (Biscuit)	SKU-101	2.95
Budget (Coke)	SKU-102	19.95
Cashew (Biscuit)	SKU-103	29.95
Cashew (Coke)	SKU-104	3.95
Cashew (Biscuit)	SKU-105	2.95
Cashew (Coke)	SKU-106	2.95
Cashew (Biscuit)	SKU-107	2.95
Cashew (Coke)	SKU-108	2.95
Cashew (Biscuit)	SKU-109	2.95
Cashew (Coke)	SKU-110	2.95
Cashew (Biscuit)	SKU-111	2.95
Cashew (Coke)	SKU-112	2.95
Cashew (Biscuit)	SKU-113	2.95
Cashew (Coke)	SKU-114	2.95
Cashew (Biscuit)	SKU-115	2.95
Cashew (Coke)	SKU-116	2.95
Cashew (Biscuit)	SKU-117	2.95
Cashew (Coke)	SKU-118	2.95
Cashew (Biscuit)	SKU-119	2.95
Cashew (Coke)	SKU-120	2.95

## 3. プレゼンテーション

- ダッシュボードを作る (ユーザーポータルを使って)

テーブルウィジェット、グラフウィジェット、スコアカードウィジェット、メーターウィジェット...

Widget Name	Value
Budget (Biscuit)	2.95
Budget (Coke)	19.95
Cashew (Biscuit)	29.95
Cashew (Coke)	3.95
Cashew (Biscuit)	2.95
Cashew (Coke)	2.95
Cashew (Biscuit)	2.95
Cashew (Coke)	2.95
Cashew (Biscuit)	2.95
Cashew (Coke)	2.95
Cashew (Biscuit)	2.95
Cashew (Coke)	2.95
Cashew (Biscuit)	2.95
Cashew (Coke)	2.95
Cashew (Biscuit)	2.95
Cashew (Coke)	2.95
Cashew (Biscuit)	2.95
Cashew (Coke)	2.95
Cashew (Biscuit)	2.95
Cashew (Coke)	2.95
Cashew (Biscuit)	2.95
Cashew (Coke)	2.95

# DeepSee KPI

## KPI 構造:

- 単独プロパティ
- 複数プロパティ
- 複数プロパティの連続



	Revenue	Expenses	Profit
Sales	\$5,072,084.90	\$3,499,297.90	\$1,572,787.00
South	\$5,322,395.40	\$3,130,838.40	\$2,191,557.00
West	\$6,240,563.80	\$5,292,515.80	\$948,048.00
East	\$9,663,238.60	\$9,282,686.60	\$380,552.00



# KPI ウィザード

The screenshot shows the 'DeepSee KPI Wizard' dialog box. The title bar includes the 'Studio Template' logo and the text 'DeepSee KPI Wizard'. On the right side of the title bar, it displays 'User: UnknownUser' and 'Namespace: SAMPLES'. The main area of the dialog contains the following elements:

- A message: 'This wizard creates a new DeepSee KPI class. Fill in the form below and then press Finish to create a new class.'
- A 'Package Name' dropdown menu with 'MyApp' selected.
- A 'Class Name' text input field.
- A 'KPI Caption' text input field.
- A 'KPI Name' text input field with a hint: 'Such as, MyFolder/MyKPI'.
- A 'Description' text area.
- A 'Source Type' section with three radio buttons: 'mdx' (selected), 'sql', and 'manual'. Below this is the text 'Specifies where the data for this KPI comes from'.
- Three list boxes labeled 'Properties', 'Filters', and 'Actions' for defining KPI behavior. Below these is the instruction 'One entry per line'.
- At the bottom right are three buttons: 'Finish', 'Cancel', and 'Help'.





# DeepSee KPIを作る 3つの方法

## SQLクエリー

```
/// This XData definition defines the KPI.
%XData KPI [ XMLNamespace = "http://www.intersystems.com/deepsee/kpi" ]
{
  <kpi xmlns="http://www.intersystems.com/deepsee/kpi"
    name="Academy/ExercisesOne" sourceType="sql"
    caption="SQL KPI"
    sql="SELECT Name, SKU, Price, AVG(Price) as Average FROM HoleFoods.Product"
  >
    <property name="SKU"      displayName="SKU"      columnNo="2"/>
    <property name="Price"    displayName="Price"    columnNo="3"/>
    <property name="Average"  displayName="Average"  columnNo="4"/>
  </kpi>
}
```

## MDXクエリー

```
/// Return an MDX statement to execute.
%Method %OnGetMDX(ByRef pMDX As %String) As %Status
{
  set pMDX = "SELECT {[Measures].[Amount Sold]} ON 0,NON EMPTY ORDER([Outlet].[H1].[Country].Members,Measures.[%COUNT],BD
  Quit $$$OK
}
```

## 手動 (カスタムコード)

```
/// Notification that this KPI is being executed.
/// This is a good place to override properties, such as range and threshold.
%Method %OnLoadKPI() As %Status
{
  // For each Row
  for i=1:1:..%seriesCount{
    // Add The Two New Properties or "Columns"
    set ..%data(i,"Cost") = ##class(Academy.Util).CalculateCost( ..%seriesNames(i) )
    set ..%data(i,"Profit") = ..%data(i,"AmountSold") - ..%data(i,"Cost")
  }
  Quit $$$OK
}
```



# KPIフィルター

KPIはフィルターを含むことが可能

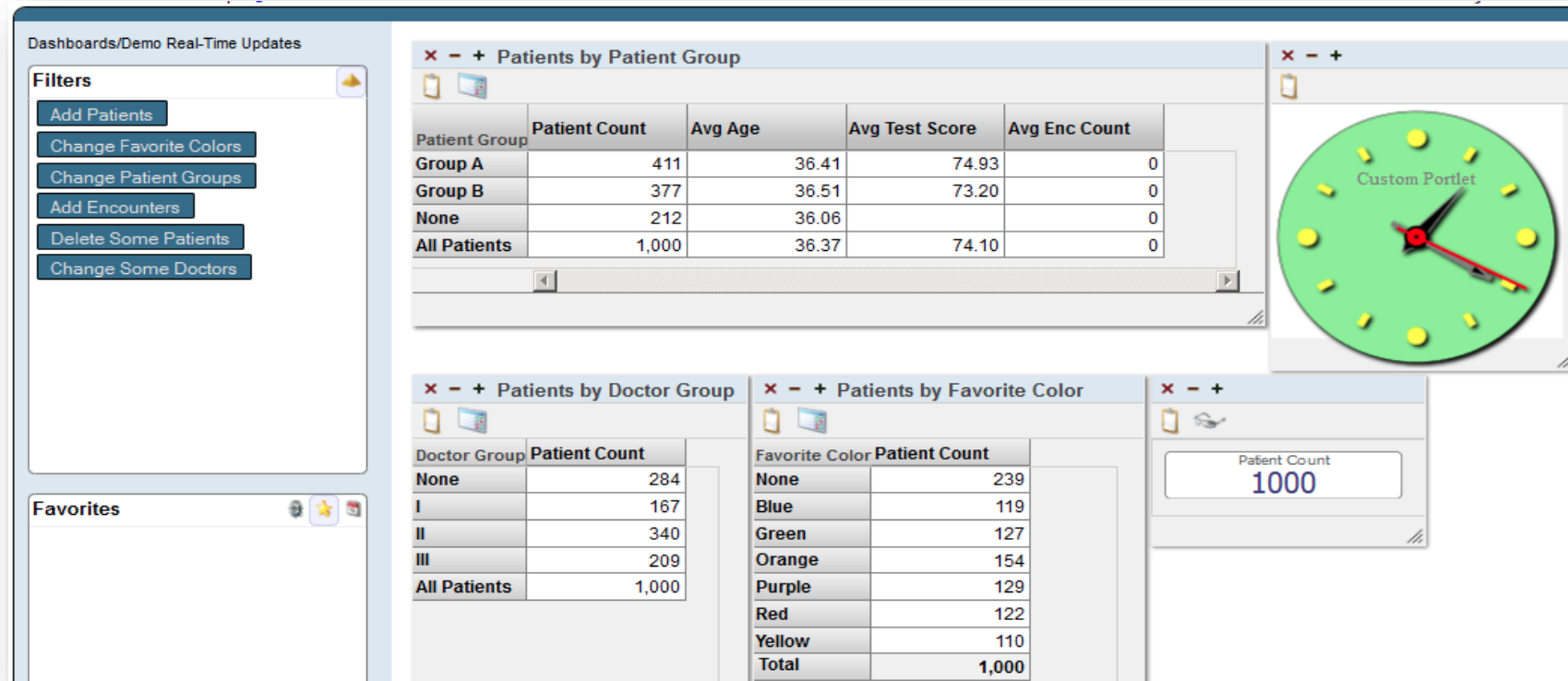
**Filters** 2 filter(s)

City	Product
<input type="text"/>	<input type="text"/>
<input type="button" value="Submit Query"/>	

KPIフィルターは以下の定義必須:

- 各フィルターの可能な値を取得する方法
- 選択フィルター値がKPIに影響する方法

# KPIアクション



# KPIテストページ

←

localhost:57772/csp/samples/HoleFoods.KPISQL.cls

KPI Test Page

KPI

Class	HoleFoods.KPISQL
Name	HoleFoods/KPISQL
Caption	KPI SQL

Filters 2 filter(s)

City	Product
<div></div>	<div></div>
<div>Submit Query</div>	

Query: sql

SELECT TOP 10000 Product->Name Product, Product ProductCode, Outlet->City As City, Outlet As CityId, Count(\*) FROM HoleFoods.SalesTransaction GROUP BY Product

KPI Values 17 series

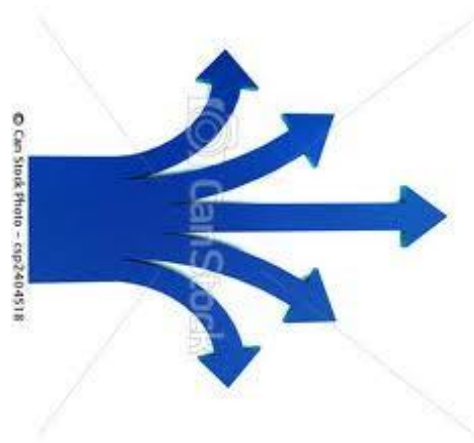
Series	Product	SKU	City	CityId	Count
Bagels (dozen)	Bagels (dozen)	SKU-101	Osaka	5	105
Donuts (dozen)	Donuts (dozen)	SKU-192	Beijing	1	45
Free-range Donuts (dozen)	Free-range Donuts (dozen)	SKU-195	Shanghai	2	140
Bundt Cake	Bundt Cake	SKU-199	Atlanta	24	170
Pretzels (bag)	Pretzels (bag)	SKU-203	Los Angeles	28	126
Unsalted Pretzels (bag)	Unsalted Pretzels (bag)	SKU-204	Houston	27	112
Onion ring	Onion ring	SKU-222	Paris	12	81
Onion ring	Onion ring	SKU-223	Paris	12	65
Penne (box)	Penne (box)	SKU-287	Tokyo	6	124
Ziti (box)	Ziti (box)	SKU-296	Beijing	1	112
Lifesavers (roll)	Lifesavers (roll)	SKU-451	New York	29	158
Calamari (frozen)	Calamari (frozen)	SKU-601	Osaka	5	180
Fruit Loops (box)	Fruit Loops (box)	SKU-708	London	9	108
Cheerios (box)	Cheerios (box)	SKU-709	Seattle	30	164



# データソース vs. データ表示

任意のウィジェットに任意のデータソースを表示可能

- ピボットテーブル
- KPIs
- ...

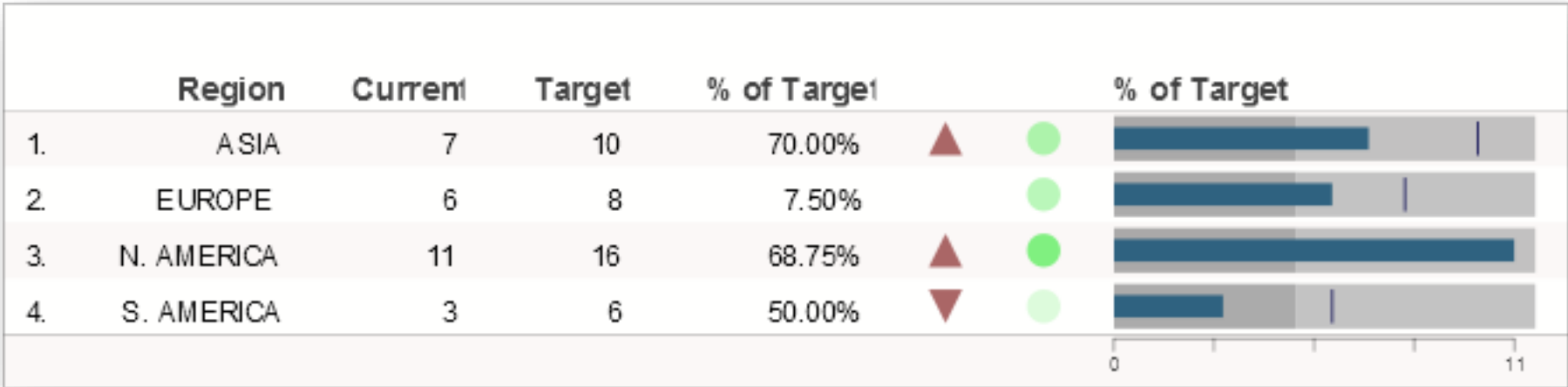


- チャート
- テーブル
- スコアカード
- Map
- メーター
- ...

例えばKPIやピボットテーブルのデータをスコアカードウィジェットに実装できるし、テーブルウィジェットにも実装できる

# スコアカード

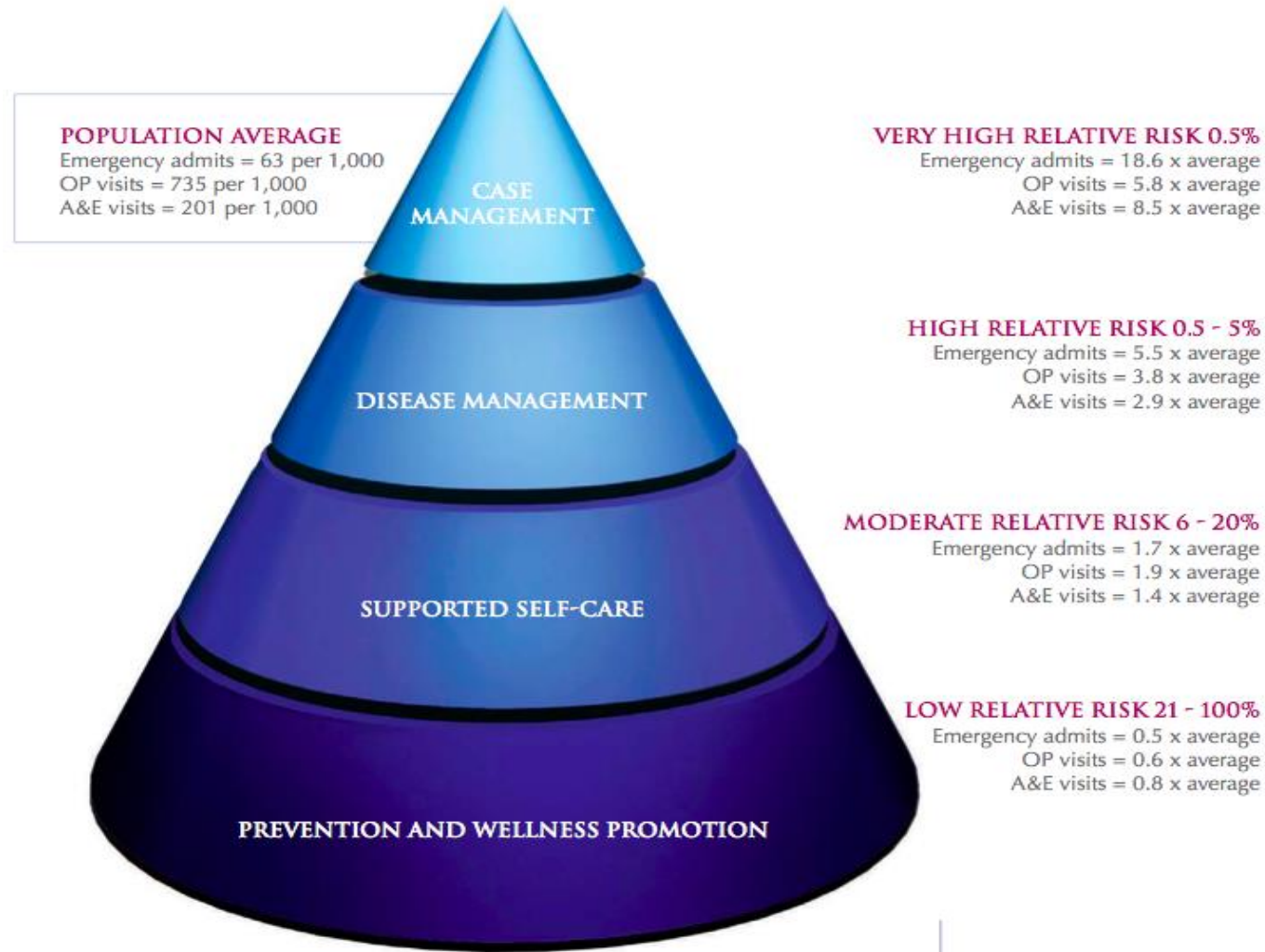
パフォーマンススコアカードはいくつかの特別なゴールに向けて企業や従業員、ビジネスユニットなどの時間経過に伴う向上の**ビジュアル表現**



# DeepSeeの利用例



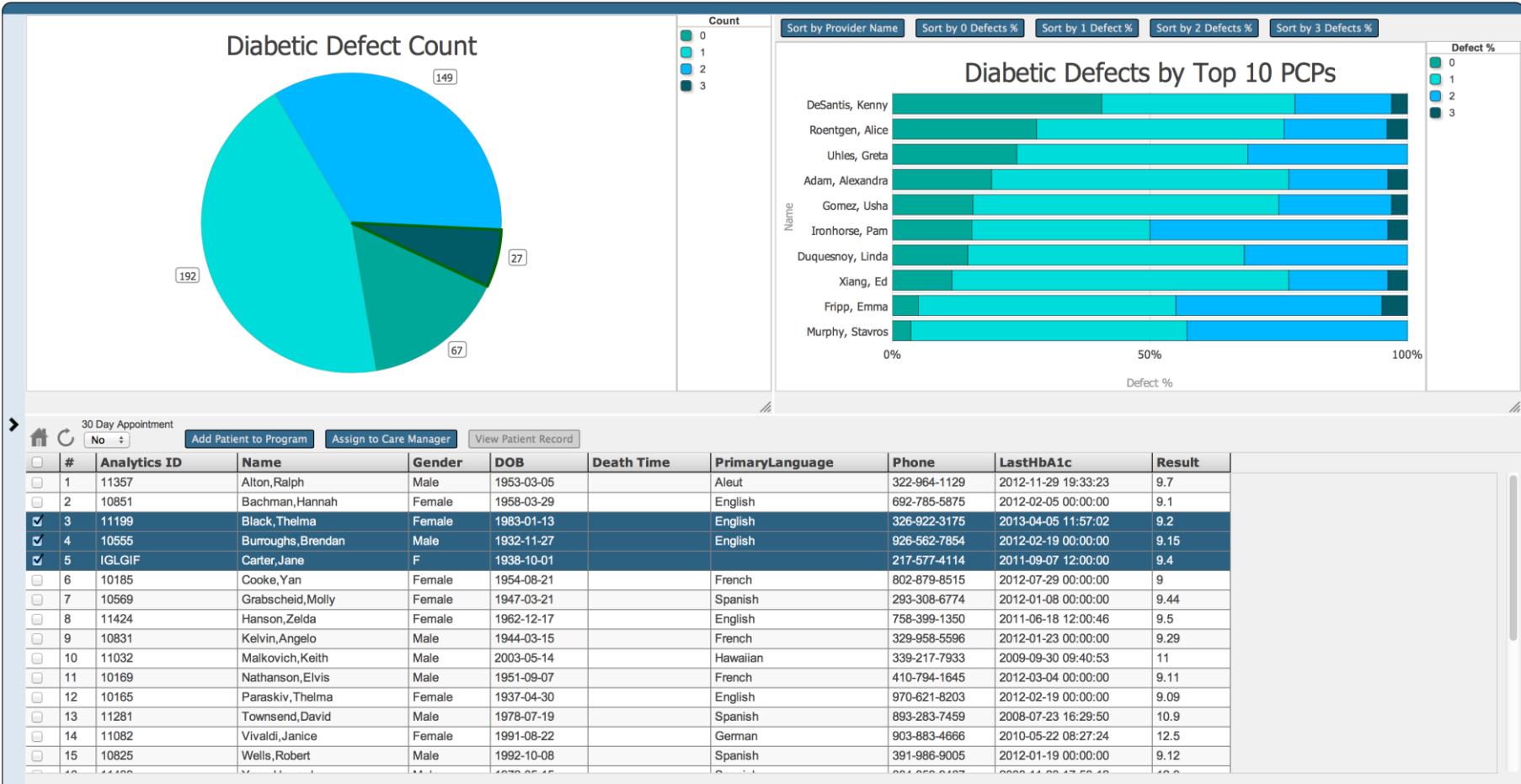
# Population Risk Stratification 集団リスク階層化





# Population Risk Stratification

集団リスク階層化



# Patient Program (List) 患者プログラム

Clinician Portal

Show Connection Details

My List

Recent

Programs

Relationships

Patient Search

Messages (1)

My Subscriptions

Program: Diabetes Education

Name	DOB	Age	Gender	PCP	PCP Phone	
Anderson, Orson	05/14/91	21	Male			
Basile, Umberto	01/17/81	31	Male			
Burroughs, Umberto	12/03/54	57	Male			
Carter, Jane	10/01/38	74	Female	Doctor Primary	617-6210600	
Geoffrion, Dave	08/23/34	78	Male			
Hernandez, Charlotte	06/13/36	76	Female			
Hertz, Patrick	05/07/01	11	Male			
Nagel, Diane	11/20/99	12	Female			
Pascal, Valery	12/12/93	18	Female			
Tesla, Elvira	09/18/35	77	Female			
Williams, Pam	02/26/04	8	Female			



## 患者サマリーに直接ドリルダウン

[Logout](#)

## 74 Years



11 [Select All](#)  
[Deselect All](#)  
[Preferences](#)

Summary	<b>Allergies</b>				<b>Medications</b>			
Allergies & Alerts	<div><div>Food Allergy</div><div>Shellfish</div><div>Rash</div><div>Drug Allergy</div><div>Penicillin</div><div>Rash</div></div>				<div><div>ACTOS</div><div></div><div>03/09/2011</div></div> <div><div>CITRACAL WITH D</div><div></div><div>03/09/2011</div></div> <div><div>MULTIVITAMIN</div><div></div><div>03/09/2011</div></div> <div><div>LISINOPRIL</div><div></div><div>04/23/2010</div></div> <div><div>DILTIAZEM ER</div><div></div><div>11/02/2009</div></div>			
History								
Documents								
Lab Results								
Radiology Results	<b>Diagnoses</b>				<b>Documents</b>			
Medications	<div><div>F</div><div>Other or unspecified hyperlipidemia</div><div></div><div>06/14/2011</div></div>				<div><div>Procedure Note</div><div></div><div><a href="#">Ophthalmology Note (w/Diagram)</a></div></div>			
Vaccinations	<div><div>F</div><div>Chronic kidney disease</div><div></div><div>06/14/2011</div></div>				<div><div>Diagnostic Imaging</div><div></div><div><a href="#">Radiology Image</a></div></div>			
Conditions	<div><div>F</div><div>Benign essential hypertension</div><div></div><div>03/09/2011</div></div>							
Procedures	<div><div>F</div><div>Diabetes mellitus without complication, type II</div><div></div><div>03/09/2011</div></div>							
Discharge Summaries	<b>Laboratory Results</b>				<b>Radiology &amp; Other Results</b>			
Physical Exams	<div><div>CBC+ AUTOMATED DIFFERENTIAL</div><div></div><div>10/30/2011 18:00</div></div>				<div><div>THYROID FUNCTION</div><div></div><div>09/07/2011 12:00</div></div>			
Plan	<div><div>CBC+ AUTOMATED DIFFERENTIAL</div><div></div><div>10/09/2011 18:00</div></div>				<div><div>Left Hand CT</div><div></div><div>02/14/2011 08:00</div></div>			
Encounters	<div><div>BASIC METABOLIC PANEL</div><div></div><div>09/07/2011 12:00</div></div>							
Appointments	<div><div>Hemoglobin A1c</div><div></div><div>09/07/2011 12:00</div></div>							
Care Team	<div><div>LIPID PANEL</div><div></div><div>09/07/2011 12:00</div></div>							
Programs								