

ORACLE  
OPEN  
WORLD

# Oracle Analytic Views

## Enhance BI Applications and Simplify Development

October 1–5, 2017  
SAN FRANCISCO, CA

William (Bud) Endress, Director, Product Management

Oracle Database Server Technology  
October 4, 2017

## Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

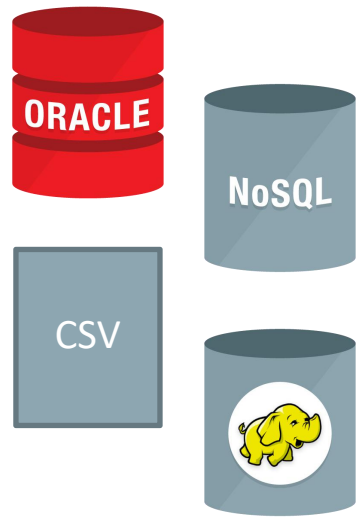
# New in 12.2 Analytic Views



- Moves business logic (Aggregations, Hierarchies, Calculations) back into database
- Simple SQL for complex analytic queries
  - no joins or GROUP-BY clauses necessary
- Works on top of pre-existing tables or views
  - no persistent storage
- Built-in data visualization via APEX

# Analytic Views

## Easier Access To You Data



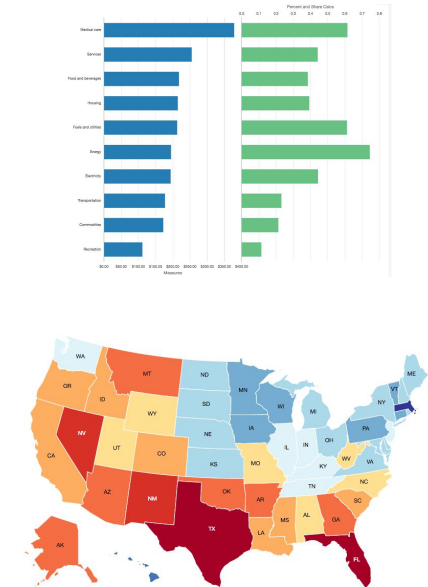
Your Data



Organized  
&  
Enhanced



Simple SQL



Your  
Applications

# Analytic Views

## Organize and Enhance Data

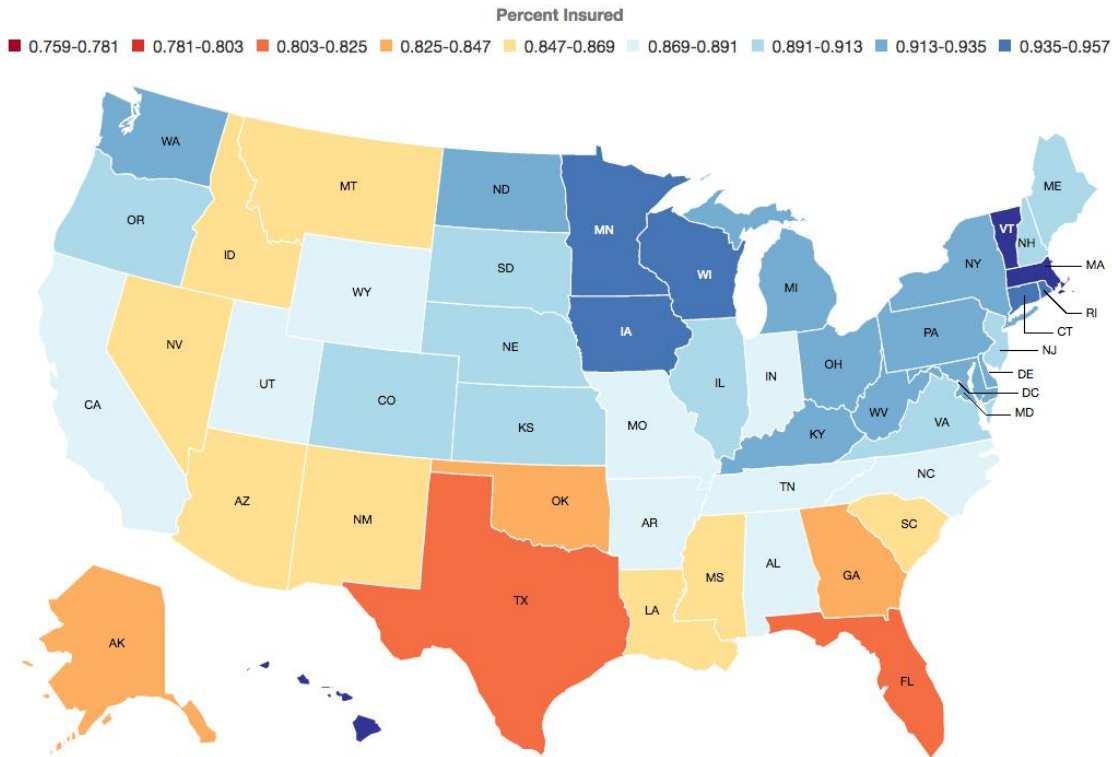
- Transforms data into a business model and presentation layer in the database
  - Data are organized for easy access and navigation
  - Data is easily extended with interesting calculations and aggregations
  - Data is easily queried with simple SQL
- Easily defined with SQL
  - Complete applications defined with just a few SQL statements
  - Supported by SQL Developer

# Analytic Views

## Better for Everyone

- For the data warehouse architect and developer
  - Easily extend star schema with aggregate data and calculations
- For the application developer
  - Simplifies metadata management and SQL generation
- For the business user
  - Built-in, browser-based data visualization via APEX application

# Analytic Views

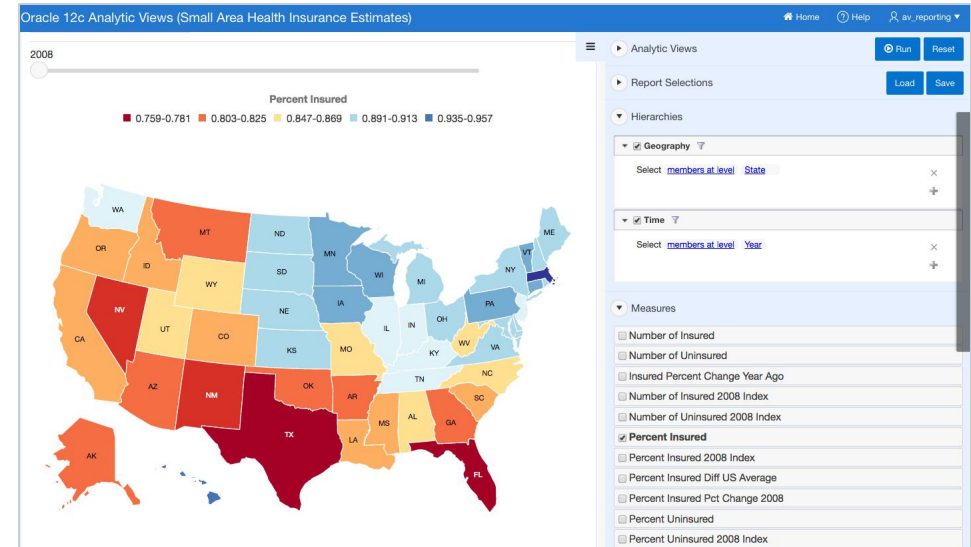


Health Insurance Coverage Rates by State, 2014

- How would you build this application?
  - Analysis of health insurance coverage rates in the United States
  - Coverage rates by time, counties and states
  - Geographic comparisons
  - Measure improvement over time
  - Interactive data visualization tools for end users

# Analytic Views

- This application can be built with 5 SQL statements
  - Create 2 hierarchies (4 SQL statements)
  - Create 1 analytic view (1 SQL statement)
- Is instantly accessible via APEX based application
- Is all in the Database



Simple SQL

Analytic View

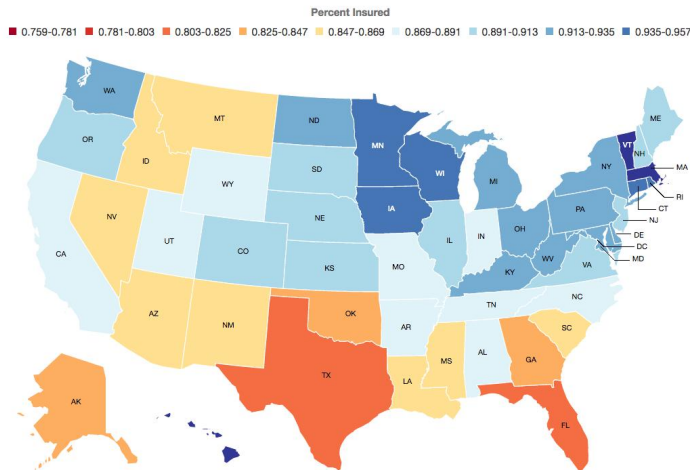
Data  
Tables, Views, etc.



# Analytic Views

## Simple SQL

```
SELECT time_hier.member_name AS TIME,  
geog_hier.member_name      AS GEOGRAPHY,  
    pct_insured  
FROM insured_av HIERARCHIES(time_hier,geog_hier)  
WHERE time_hier.level_name = 'YEAR'  
AND   geog_hier.level_name = 'STATE'  
ORDER BY time_hier.hier_order ,  
          geog_hier.hier_order;
```



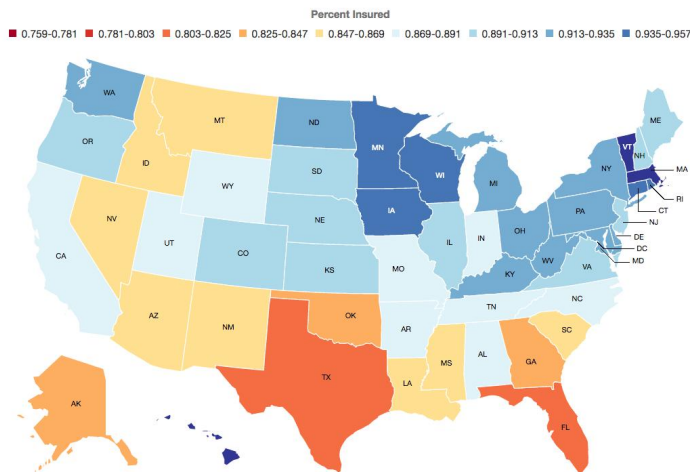
Fact data is selected from analytic views using SQL

Analytic views are views on top of a star schema.  
No storage structures

# Analytic Views

## Simple SQL

```
SELECT time_hier.member_name AS TIME,  
       geog_hier.member_name  AS GEOGRAPHY,  
       pct_insured  
FROM   insured_av HIERARCHIES(time_hier,geog_hier)  
WHERE  time_hier.level_name = 'YEAR'  
AND    geog_hier.level_name = 'STATE'  
ORDER BY time_hier.hier_order ,  
         geog_hier.hier_order;
```



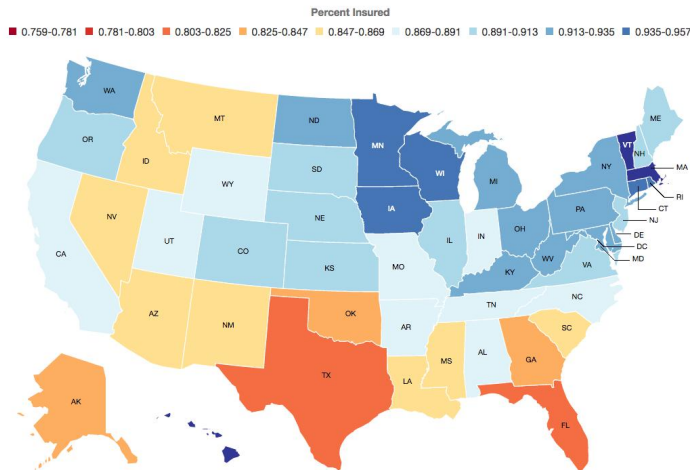
The HIERARCHIES clause specifies the dimensions and hierarchies for this query

No JOIN or GROUP BY clauses in analytic view queries

# Analytic Views

## Simple SQL

```
SELECT time_hier.member_name AS TIME,  
       geog_hier.member_name AS GEOGRAPHY,  
       pct_insured  
FROM insured_av HIERARCHIES(time_hier,geog_hier)  
WHERE time_hier.level_name = 'YEAR'  
AND    geog_hier.level_name = 'STATE'  
ORDER BY time_hier.hier_order ,  
         geog_hier.hier_order;
```



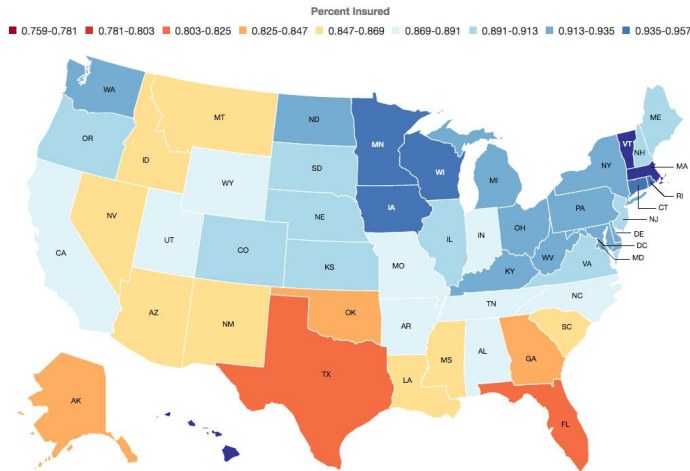
Standardized columns such as 'member\_name' are selected from the hierarchies

Standardized columns such as 'member\_name' are selected from the hierarchies

# Analytic Views

## Simple SQL

```
SELECT time_hier.member_name AS TIME,  
       geog_hier.member_name   AS GEOGRAPHY,  
       pct_insured  
FROM   insured_av HIERARCHIES(time_hier,geog_hier)  
WHERE time_hier.level_name = 'YEAR'  
AND   geog_hier.level_name = 'STATE'  
ORDER BY time_hier.hier_order ,  
         geog_hier.hier_order;
```



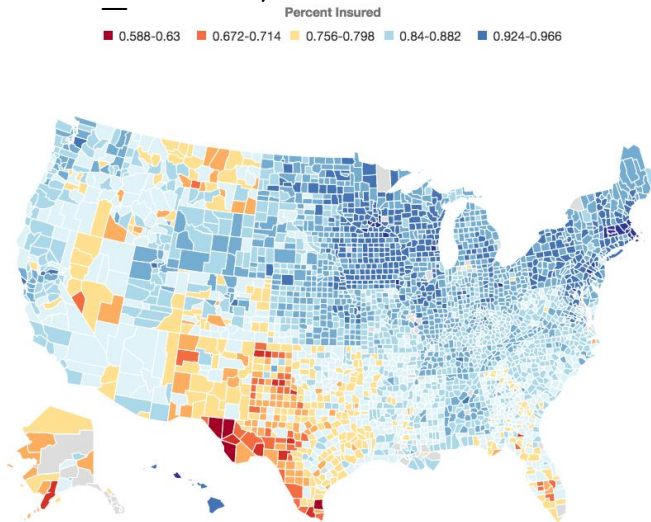
Levels of aggregation are specified in the WHERE clause

When filtering on the level 'State' for the time hierarchy, the member named will include California, New York, etc

# Analytic Views

## Simple SQL

```
SELECT time_hier.member_name AS TIME,  
       geog_hier.member_name AS GEOGRAPHY,  
       pct_insured  
FROM insured_av HIERARCHIES(time_hier,geog_hier)  
WHERE time_hier.level_name = 'YEAR'  
AND    geog_hier.level_name = 'COUNTY'  
ORDER BY time_hier.hier_order ,  
         geog_hier.hier_order;
```



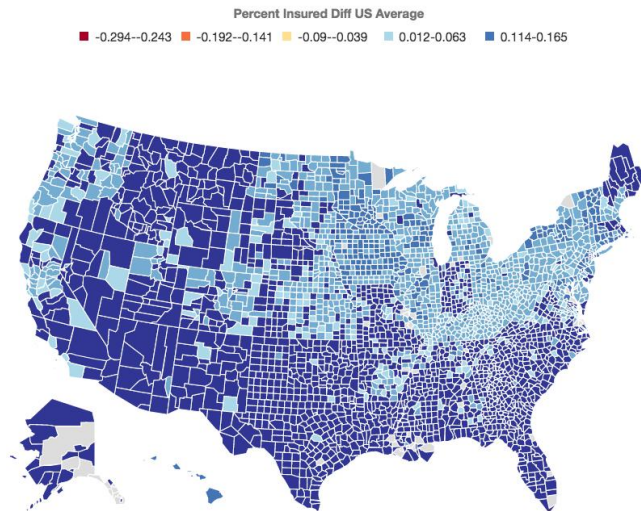
To drill, just update the WHERE clause. Everything else remains the same.

The calculations automatically use new hierarchy levels.

# Analytic Views

## Simple SQL

```
SELECT time_hier.member_name AS TIME,  
       geog_hier.member_name  AS GEOGRAPHY,  
       pct_insured_diff_us_avg  
FROM insured_av HIERARCHIES(time_hier,geog_hier)  
WHERE time_hier.level_name = 'YEAR'  
AND    geog_hier.level_name = 'COUNTY'  
ORDER BY time_hier.hier_order ,  
         geog_hier.hier_order;
```



To select a calculation, just  
select columns

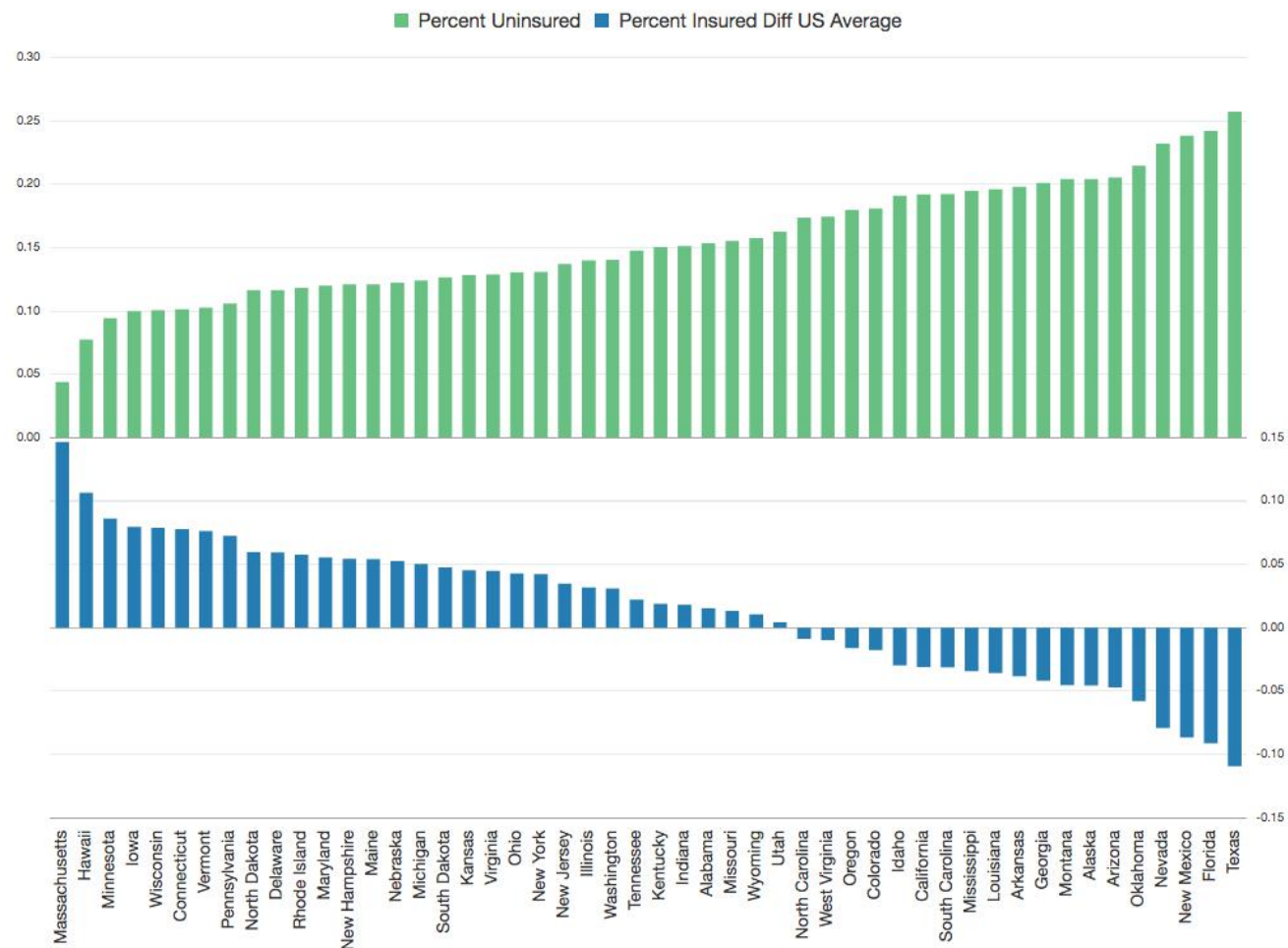
Calculations are express in the  
analytic view so they can just be  
selected in the query

# Analytic Views

## Embedded Calculations

- Easily create new measures
  - Simplified syntax based on business model
  - Includes dimensional and hierarchical functions

Add Percent Uninsured  
Difference from US Average with  
a single line of code

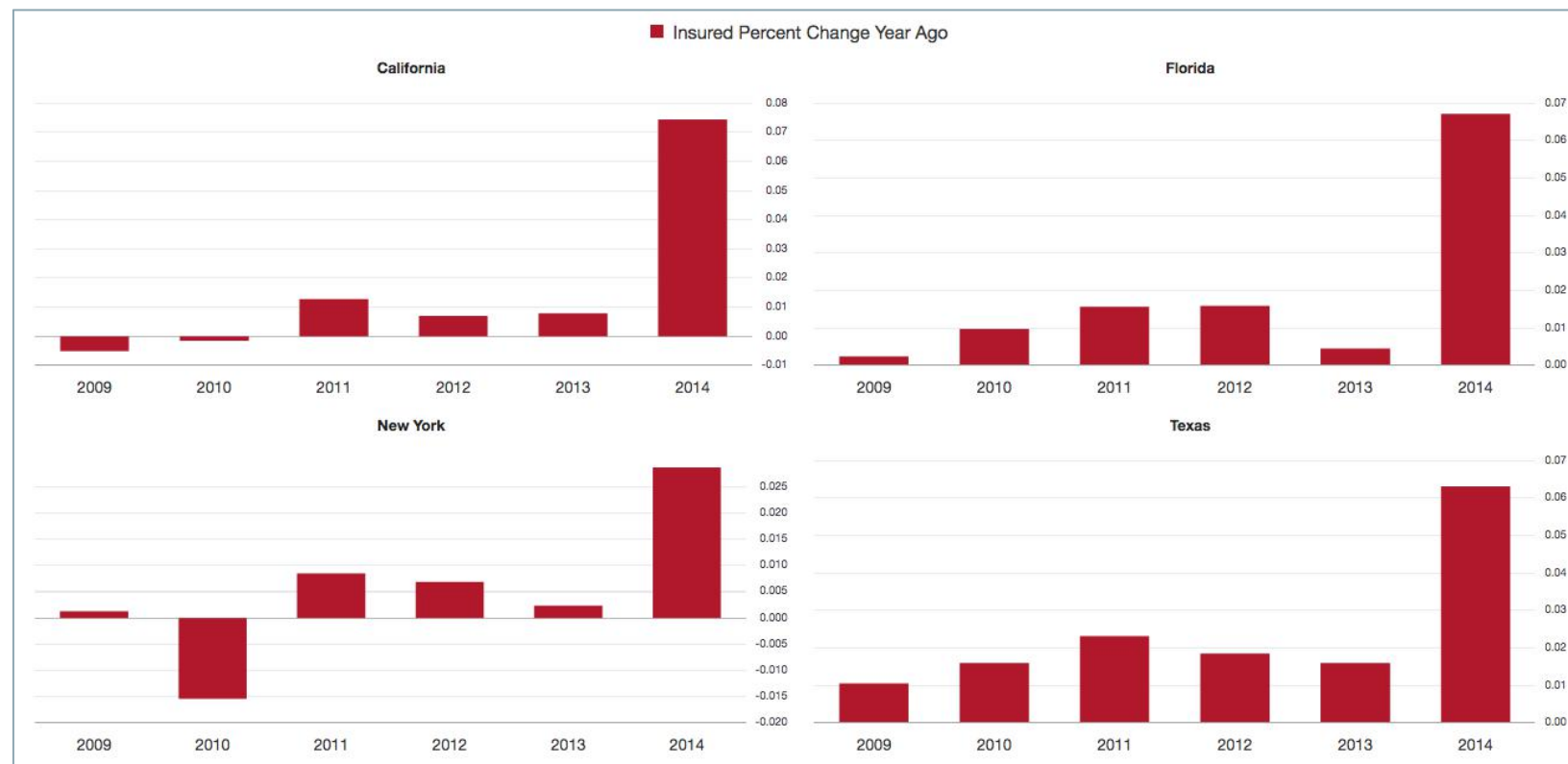


```
SHARE_OF(pct_uninsured HIERARCHY  
geog_hier  
MEMBER country ['USA']) - 1)
```

# Analytic Views

## Embedded Calculations

Add time series calculations with a single line of code



```
LAG_DIFF_PERCENT(pct_insured)  
OVER (HIERARCHY time_hier  
OFFSET 1 ACROSS ANCESTOR AT LEVEL year)
```



# Analytic Views

## Descriptive Metadata

- Descriptive metadata available in data dictionary
  - Descriptive names and descriptions
  - Translatable
  - Measure formatting
  - User/application extensible

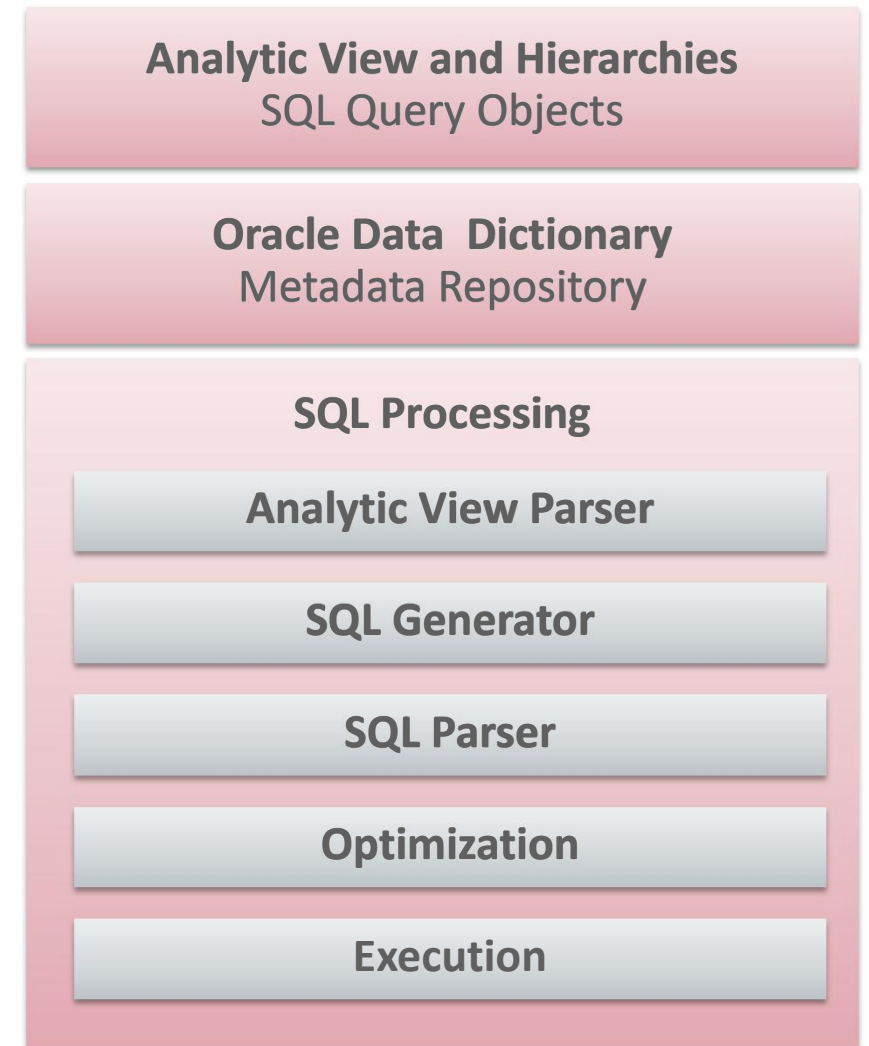
```
CREATE OR REPLACE FORCE HIERARCHY "AV_DEMO"."SAHIE_GEOG_HIER"  
  CLASSIFICATION "CAPTION" VALUE 'Geography'  
  CLASSIFICATION "DESCRIPTION" VALUE 'Geography'  
  CLASSIFICATION "AV_TYPE" VALUE 'GEOGRAPHY'  
  USING "SAHIE_GEOG_ATTR_DIM"  
  ("COUNTY"  
   CHILD OF "STATE"  
   CHILD OF "COUNTRY")  
;
```

```
CREATE OR REPLACE FORCE ANALYTIC VIEW "AV_DEMO"."SAHIE_GEOG_AV"  
  CLASSIFICATION "CAPTION" VALUE 'Small Area Health Insurance Estimates'  
  CLASSIFICATION "DESCRIPTION" VALUE 'US Cenus Bureau - Model-based Small Area'  
  CLASSIFICATION "DATA_SET_ORIGIN" VALUE 'www.census.gov/did/www/sahie/'  
  USING "SAHIE_GEOG_FACT_V" AS "SAHIE_GEOG_FACT_V"  
  DIMENSION BY  
  ( "SAHIE_TIME_ATTR_DIM" AS "SAHIE_TIME_ATTR_DIM"  
    KEY ("YEAR")
```

```
"INSURED_SHARE_2008" AS  
  (SHARE_OF(num_insured HIERARCHY sahie_time_hier MEMBER year['2008']))  
  CLASSIFICATION "CAPTION" VALUE 'Number of Insured 2008 Index'  
  CLASSIFICATION "DESCRIPTION" VALUE 'Number of Insured 2008 Index'  
  CLASSIFICATION "FORMAT_STRING" VALUE '9,999.99',
```

# Analytic Views

- Analytic Views and Hierarchies
  - Objects that are queried with SQL
- Data Dictionary
  - All metadata for analytic views
- Analytic View Parser
  - Syntax and semantic checks
- SQL Generator
  - Transforms AV SQL into executable SQL
- SQL Parser, Optimization and Execution
  - Oracle SQL engine



# “Standard” and Analytic Views

	“Standard” View	Analytic View
Data Sources (FROM)	Yes	Yes
Joins	Yes	Yes
Business Model-Based Calculations	No	Yes
Automatic Hierarchical Columns	No	Yes
Automatic Multi-Level Aggregation	No	Yes
Automatic Filter Expansion	No	Yes
Automatic Outer Join	No	Yes
Automatic Order of Calculation	No	Yes
Presentation Metadata	No	Yes

## My Tutorials

Search My Tutorials

Learn More at [LiveSQL.Oracle.com](https://livesql.oracle.com)

### Querying Analytic Views

This tutorial provides an introduction to querying hierarchies and analytic views. This is a good starting point for anyone who is new to analytic views.

SQL General / Public

### Creating Time Series Calculations in Analytic Views

This tutorial provides examples for creating time series calculations such as prior periods, year ago and period-to-dates. These types of calculations can be used with

SQL General / Public

### Attribute Reporting with Analytic Views

This tutorial introduces attribute reporting using analytic views.

SQL General / Private

### Creating Analytic Views - Getting Started

In this tutorial you will learn how to create an analytic view using data in the AV sample schema. If you are new to analytic views this is a great place to start.

SQL General / Public

### Creating an Analytic View for the Sales History (SH) Sample Schema

This tutorial will create an analytic view and supporting objects using data in the Sales History (SH) sample schema. It offers a quick overview of the process of

SQL General / Public

### Creating SHARE\_OF (Hierarchical Ratio) Calculations in Analytic Views

SHARE\_OF expressions calculate the ratio of a hierarchy value to the value of a parent or an ancestor. For example, the share (ratio) of sales for a store within a

SQL General / Public

### Using Materialized Views With Analytic Views

This tutorial shows how to use materialized views to accelerate queries that select from analytic views.

SQL Analytics / Public

### Adding Multiple Language Support to Analytic Views

This tutorial provides examples of adding multiple language support to analytic views.

SQL Analytics / Public



ORACLE®