



Lifecycle Manager and the Dynamic Query

Ashley Perry – IBM Cognos CTP

Business Analytics software

© 2012 IBM Corporation



Session's objectives

Lifecycle Manager

- Learn how lifecycle management cuts effort, time and complexity from every phase in the life of your BI and PM solution
- Learn about the capabilities that IBM Cognos 10 delivers to enable efficient lifecycle management
- Learn how to access these capabilities and take advantage of the benefits of lifecycle management

...mainly by demonstration!!! 😊

Dynamic Query Mode

- What is Dynamic Query and Dynamic Query Mode?
- Who should use Dynamic Query?
- Why would I use Dynamic Query?
- When should I use Dynamic Query?

...a few slides 😞

...but mainly by demonstration 😊

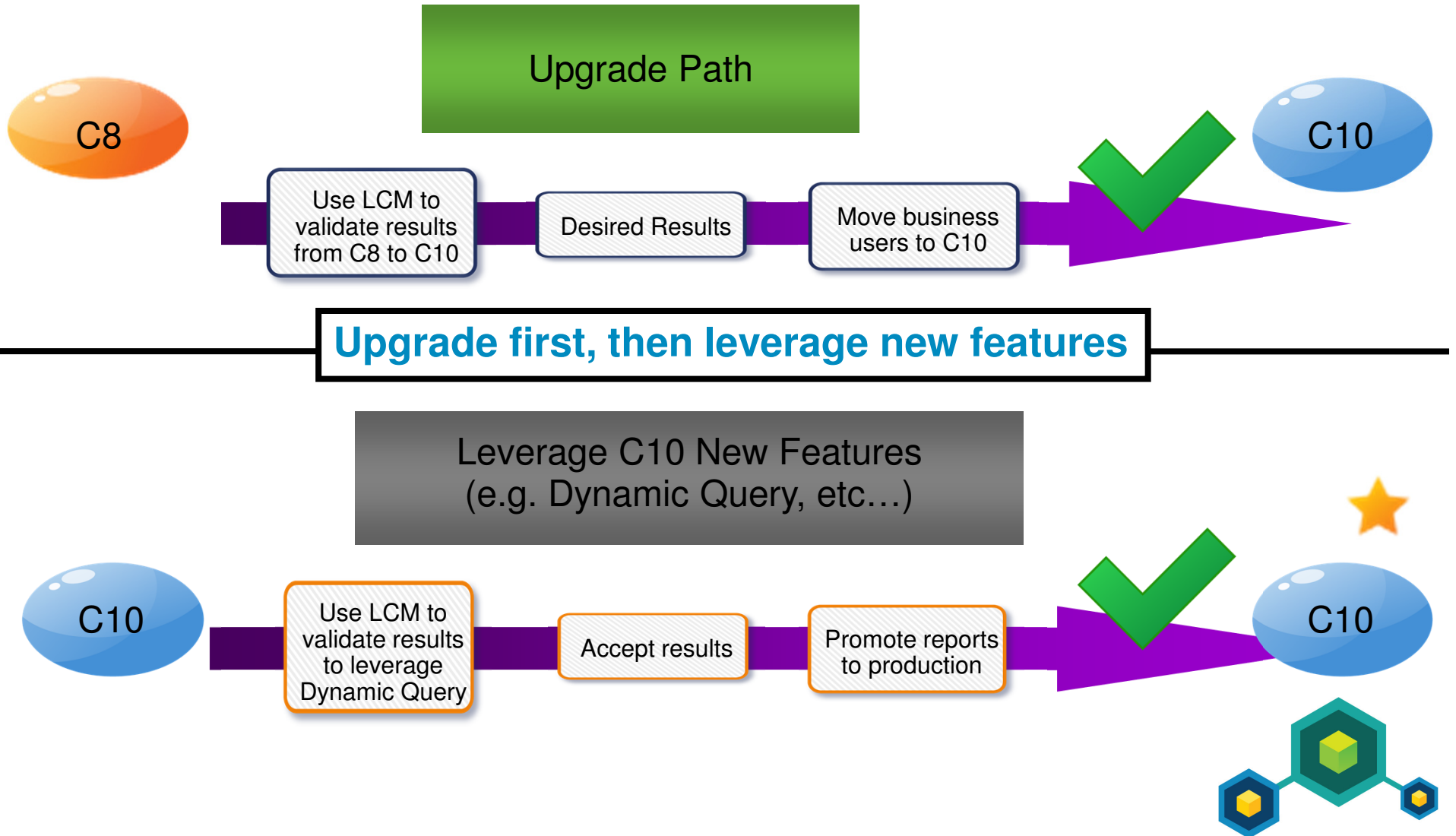
- How to use Dynamic Query? (if time...)



Lifecycle Manager



IBM Cognos 10 streamlines lifecycle management with the ability to seamlessly upgrade



IBM Cognos Lifecycle Manager

Master Projects Settings Help

Welcome

To begin, please create a [new project](#) or [open](#) an existing one.

Demo IBM Cognos Lifecycle Manager

Projects and Archives

A project contains data such as prompt values, validation and run results, and report outputs, related to the source and target applications being compared.
An archive is a project that cannot be modified. A new project may be created by using an archive as baseline.

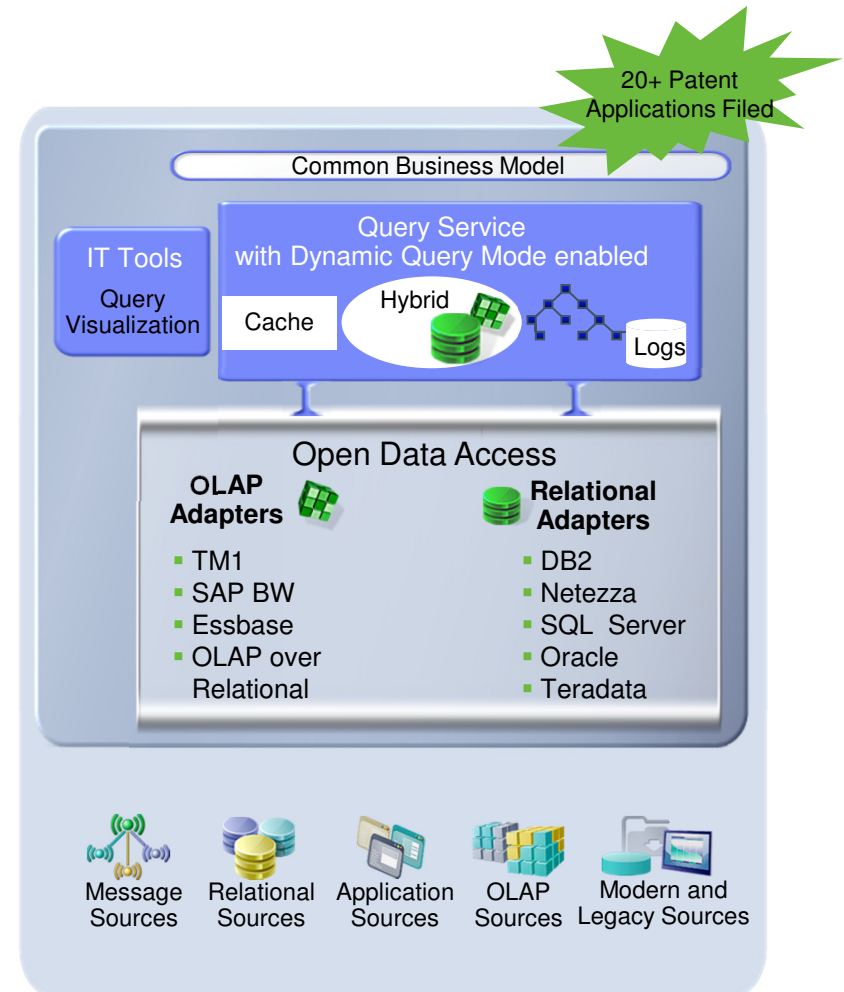


Dynamic Query Mode



What is Dynamic Query?

- A new query processing mode designed to improve query performance
 - Can be used with or in place of existing compatible query service
- Leverages 64-bit architecture
- Optimized query processing
- Security aware caching
- Query visualization tool
- JDBC connectivity to relational sources
- OLAP Sources
 - TM1, SAP BW, Essbase, OLAP over Relational (DMR)
- Relational Sources
 - DB2 (LUW and Z), Netezza, Microsoft SQL Server, Oracle, Teradata



Dynamic Query Mode Leverages 64-bit architecture

64-bit does not provide:

- Improved query performance on its own
 - CPUs are slightly faster and have slightly faster memory access

64-bit architecture provides:

- Dynamic Query mode has access to more memory
 - Access to more addressable memory
 - More caching and that enables performance
- Retain cache as long as necessary: day, week, month...
 - Dynamic Query Mode can hold onto things longer in memory to be leveraged over and over



Who should use Dynamic Query?

Dynamic Query provides unique opportunities for IT to better service their clients



**IT
Administrators**

- 64-bit, Java based query service



**Data
Professionals**

- In-memory processing and data cache



**BI
Professionals**

- Enhanced query performance and usability



Who should use Dynamic Query?

Dynamic Query enables delivery of applications that delight Business users



Executives

- High performance executive dashboards delivering insight into the business



Business Analysts

- Gain insight from greater volumes of data coming from disparate data sources.



Business Users

- Access to the information they need in a timely manner



Why Use Dynamic Query?

- Improved query performance!
- Dynamic Query has been designed and developed to provide high performance query results through:
 - Optimized, multi-phase SQL query
 - Metadata, query plan and query result re-use
 - Shareable caches providing in-memory, balanced local processing
 - Hybrid query request processing



Why is Dynamic Query important?

All capabilities against any data creates query challenges

User requests a sorted list-style report



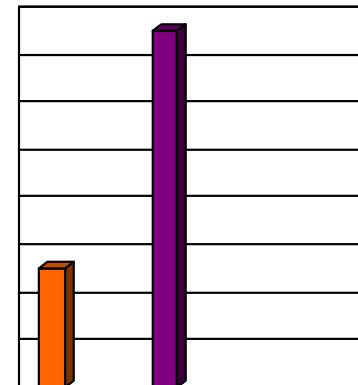
MDX source requires more complex processes to respond

```
select
  {[Measures].[Unit Sales]} on columns,
  {GENERATE (
    ORDER (
      [Customers].Levels(1).MEMBERS,
      [Customers].CURRENTMEMBER.PROPERTIES
      ("MEMBER_UNIQUE_NAME"), BDESC),
    UNION( {[Customers].CURRENTMEMBER},
    GENERATE (
      ORDER (
        [Customers].CURRENTMEMBER.PROPERTIES
        ("MEMBER_UNIQUE_NAME"), BASC),
      UNION( {[Customers].CURRENTMEMBER},
      GENERATE (
        DESCENDANTS (
          [Customers].CURRENTMEMBER,
          [Customers].Levels(3)),
      UNION( {[Customers].CURRENTMEMBER},
      DESCENDANTS (
        [Customers].CURRENTMEMBER,
        [Customers].Levels(4)),
        ALL),
        ALL),
        ALL),
    ALL) } on ROWS
  from [Sales]
```



SQL source can respond easily to this type of request

```
SELECT Level1, Level2, Level3, Level4, Cost
FROM table
ORDER BY Level1 DESC, Level2 ASC
```



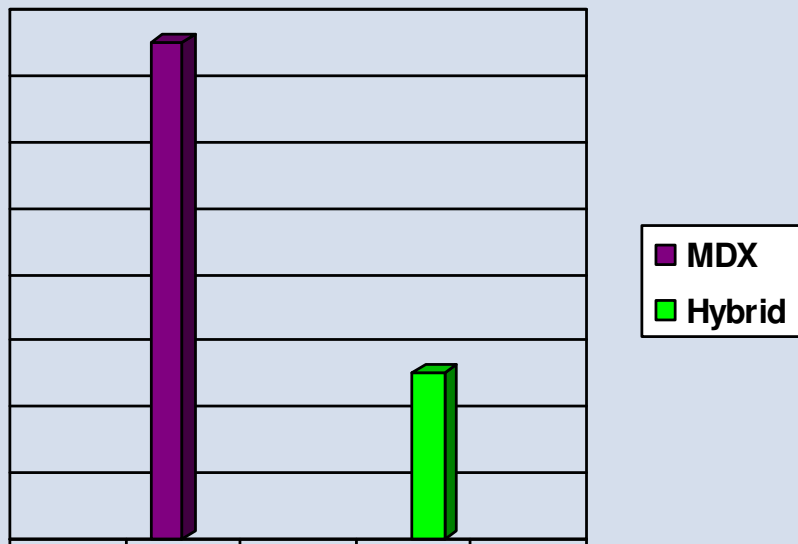
SQL Query
MDX Query

Query response time



Hybrid query delivers optimal performance across all sources

Hybrid Query dynamically combines MDX and SQL in the single query request



Query response time

Leverage strengths of both technologies to drive fastest results possible

```

WITH
  [OLAPQuery] AS
  SELECT
    {[Measures].[Unit Sales]} ON COLUMNS,
    DRILLDOWNLEVEL(
      [Customers].[Level1].MEMBERS,
      [Customers].[Level4] ) ON ROWS
  FROM SALES
  SELECT [Customers],
         [Measures].[Unit Sales]
  FROM
    [OLAPQuery]
  ORDER BY
    [Customers].[Level1] DESC,
    [Customers].[Level2] ASC
    
```



How to use Dynamic Query...Demo

- Sample Data – Star Schema
 - Main Fact: 15,319,150 rows
 - Staff Dimension: 104
 - Retailers: 191
 - Product Line: 5
 - Product Type: 21
 - Products: 115
 - Order Method: 7
 - Periods: 39,585

Quantity		1900	1901	1902	1903	1904	1905	1906	1907	1908	1909	1910	1911	1912	1913	1914	1915	1916	1917	1918	1919	1920
Camping Equipment	Cooking Gear	415,220	707,420	864,120	412,980	690,120	846,960	437,600	686,540	814,060	471,560	574,680	831,940	584,980	563,140	814,600	611,800	533,700	792,700	656,500	494,720	783,700
	Lanterns	721,780	1,223,520	1,505,680	731,420	1,225,320	1,454,460	773,820	1,207,780	1,381,160	830,820	1,059,220	1,351,120	1,081,700	1,023,280	1,334,300	1,106,740	978,340	1,329,460	1,175,500	904,340	1,311,400
	Packs	205,920	334,580	415,020	211,320	331,100	398,760	232,220	319,300	387,480	241,160	276,620	387,120	298,280	268,000	383,880	300,820	268,560	359,540	331,280	256,740	349,500
	Sleeping Bags	198,400	340,940	423,120	210,680	324,460	421,980	218,780	321,880	415,920	220,540	281,560	393,760	301,220	275,600	391,320	306,860	263,000	386,100	323,400	247,240	374,500
	Tents	276,920	459,480	570,240	298,420	425,560	560,880	333,720	420,100	543,420	336,860	370,900	538,180	418,180	345,140	553,360	425,940	337,020	533,080	447,160	321,660	519,680
Golf Equipment	Golf Accessories	91,780	190,540	197,560	89,140	187,420	188,700	98,800	183,180	187,160	98,660	158,300	189,400	129,780	157,840	186,960	133,960	148,680	187,240	139,480	132,200	196,500
	Irons	26,920	53,940	61,580	26,960	53,760	59,280	29,160	50,120	59,100	31,480	44,220	57,640	41,740	42,420	57,340	42,720	42,160	56,000	42,620	38,460	57,060
	Putters	44,800	89,920	97,720	44,120	88,720	96,020	48,100	82,740	97,280	49,420	72,300	94,000	67,160	69,820	96,720	66,500	67,860	95,360	68,740	57,700	99,680
	Woods	25,620	52,780	60,840	27,140	51,560	58,020	29,960	49,180	57,760	30,800	43,880	54,480	42,140	44,340	53,280	43,340	42,760	52,640	43,680	35,980	56,500
Mountaineering Equipment	Climbing Accessories		657,740	763,760	23,560	613,760	760,440	70,060	567,840	722,260	154,980	420,820	685,040	329,220	368,160	706,220	370,680	311,360	677,020	456,680	221,340	675,060
	Rope		188,340	218,440	9,300	173,880	217,260	24,940	159,740	212,400	43,940	122,300	201,840	91,940	111,720	200,820	103,540	93,100	199,580	123,400	66,520	199,680
	Safety		103,540	119,780	4,120	96,120	118,680	12,640	89,020	112,680	25,740	66,760	108,040	52,640	58,580	106,620	62,240	49,980	104,980	72,480	36,060	103,100
	Tools		446,000	521,960	23,460	412,000	518,880	60,560	373,480	516,000	101,960	279,200	498,720	213,520	250,440	491,520	249,480	206,580	489,300	295,560	151,620	479,780
Outdoor Protection	First Aid	361,380	223,260	138,860	351,580	230,900	140,260	339,140	242,860	139,000	311,040	253,560	146,240	291,020	270,720	157,300	250,760	283,240	174,220	227,860	292,860	185,920
	Insect Repellents	1,324,760	831,780	544,200	1,327,140	837,500	542,760	1,321,480	848,860	545,540	1,274,400	812,780	583,760	1,177,260	876,320	620,180	1,024,180	955,180	673,040	925,140	1,013,280	711,800
	Sunscreen	1,060,380	658,780	437,160	1,052,920	669,860	431,480	1,047,060	681,040	434,580	1,019,920	675,260	438,960	948,440	722,060	470,900	804,800	801,020	505,400	714,340	849,780	554,500
Personal Accessories	Binoculars	86,880	136,840	209,580	91,560	133,160	211,580	91,900	128,900	199,520	98,840	115,120	184,360	140,900	113,620	177,080	147,600	108,920	173,120	157,400	98,840	170,500
	Eyewear	122,720	187,460	224,920	128,220	184,280	223,680	127,620	183,780	214,480	132,700	162,600	212,800	163,900	160,760	207,940	168,760	153,960	200,640	180,260	142,600	203,800
	Knives	302,760	457,700	580,860	310,540	456,200	569,800	316,380	442,480	555,800	310,880	399,820	521,640	430,680	388,540	511,680	440,820	369,320	517,260	458,860	353,540	490,000



Demo Report 1

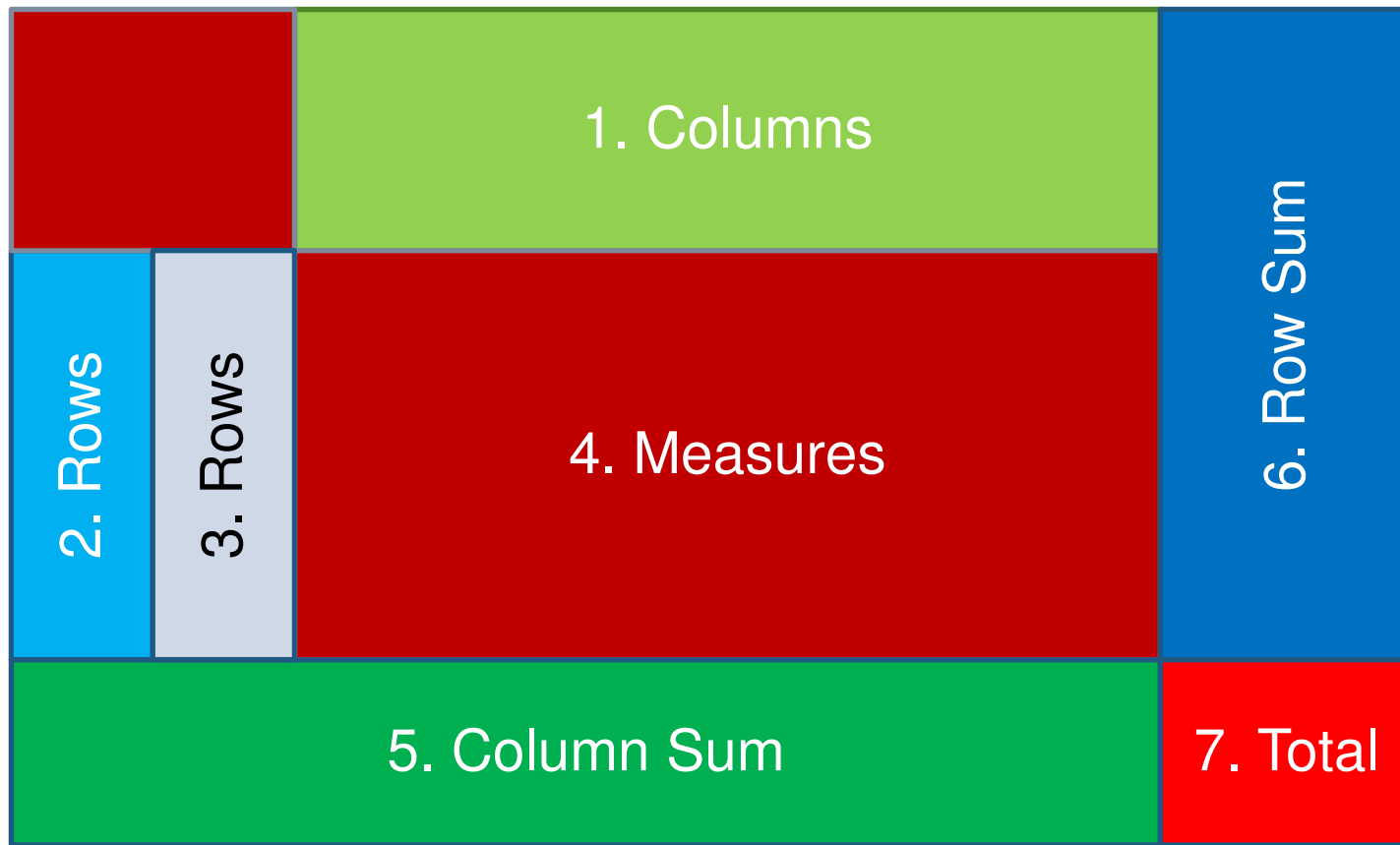


Quantity		1900	1901	1902	1903	1904	1905	1906	1907	1908	1909	1910	2006	2007	2008	All Periods
Camping Equipment	Packs	205,920	334,580	415,020	211,320	331,100	398,760	232,220	319,300	387,480	241,160	276,620	271,860	385,340	201,580	34,312,600
	Sleeping Bags	198,400	340,940	423,120	210,680	324,460	421,980	218,780	321,880	415,920	220,540	281,560	277,420	394,500	211,960	34,717,900
	Tents	276,920	459,480	570,240	298,420	425,560	560,880	333,720	420,100	543,420	336,860	370,900	351,720	552,580	284,120	47,235,300
Golf Equipment	Golf Accessories	91,780	190,540	197,560	89,140	187,420	188,700	98,800	183,180	187,160	98,660	158,300	159,240	186,540	92,180	17,004,400
	Irons	26,920	53,940	61,580	26,960	53,760	59,280	29,160	50,120	59,100	31,480	44,220	42,940	57,400	27,820	5,070,100
	Putters	44,800	89,920	97,720	44,120	88,720	96,020	48,100	82,740	97,280	49,420	72,300	69,860	96,440	44,360	8,284,500
	Woods	25,620	52,780	60,840	27,140	51,560	58,020	29,960	49,180	57,760	30,800	43,880	44,260	53,580	29,660	4,984,000
Mountaineering Equipment	Climbing Accessories		657,740	763,760	23,580	613,780	760,440	70,860	567,840	722,260	154,980	420,820	385,820	697,180	362,080	50,577,800
	Rope		188,340	218,440	9,300	173,880	217,260	24,940	159,740	212,400	43,940	122,300	115,460	200,440	100,180	14,562,800
	Safety		103,540	119,780	4,120	96,120	118,680	12,640	89,020	112,680	25,740	66,760	61,200	106,000	60,240	7,960,400
	Tools		446,000	521,980	23,460	412,000	518,880	60,560	373,480	516,000	101,960	279,200	259,420	495,540	236,480	34,700,400
Outdoor Protection	First Aid	361,360	223,260	138,860	351,580	230,900	140,260	339,140	242,860	139,000	311,040	253,560	268,200	149,440	46,720	25,526,200
	Insect Repellents	1,324,760	831,780	544,200	1,327,140	837,500	542,760	1,321,480	848,860	545,540	1,274,400	812,780	873,820	609,420	189,680	95,734,800
	Sunscreen	1,060,380	656,780	437,160	1,052,920	669,860	431,480	1,047,060	681,040	434,580	1,019,920	675,260	722,960	448,780	142,240	76,297,900
Personal Accessories	Binoculars	86,880	136,840	209,580	91,560	133,160	211,580	91,900	128,900	199,520	98,840	115,120	112,920	179,040	110,860	15,639,400
	Eyewear	122,720	187,460	224,920	128,220	184,280	223,680	127,620	183,780	214,480	132,700	162,600	160,900	209,780	105,020	19,232,500
	Knives	302,760	457,700	580,880	310,540	456,200	569,800	316,380	442,480	555,800	310,880	399,820	393,860	510,040	306,880	48,098,400
	Navigation	115,840	201,080	249,740	114,700	199,720	242,800	125,780	190,840	241,900	124,840	168,220	162,460	216,420	134,140	20,263,600
	Watches	235,020	353,660	434,000	230,620	349,720	419,300	236,400	339,380	428,500	210,740	313,320	307,680	404,820	203,420	36,204,000
All Products	5,617,060	7,897,300	8,639,180	5,719,920	7,735,140	8,481,980	5,976,920	7,569,040	8,266,000	6,121,280	6,671,440	6,654,840	8,108,080	4,075,280	791,412,300	

- Report 1: Xtab – an overhead with HTML rendering
 - Rows: Products Type nested below Product Line
 - Columns: Time (108 Years).
 - Measure: Quantity
 - Rows and Columns aggregated



Demo Report 1 Queries



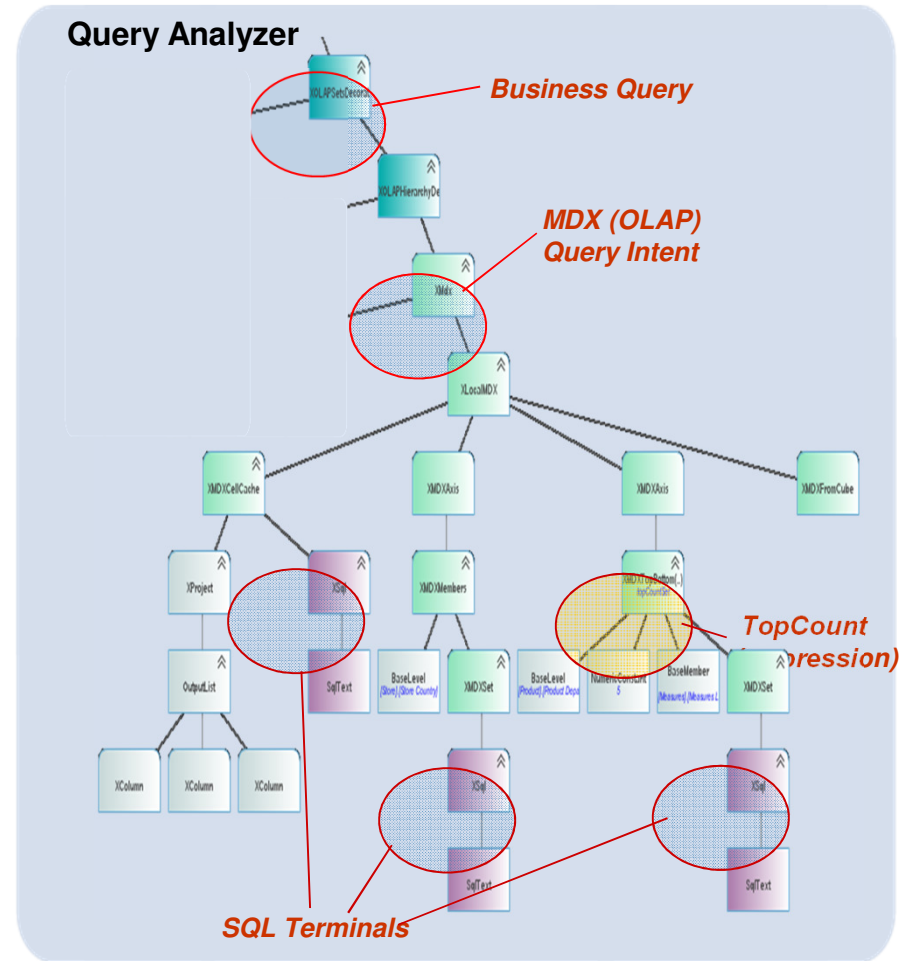
Demo Report 1

Quantity		1900	1901	1902	1903	1904	1905	1906	1907	1908	1909	1910	2006	2007	2008	All Periods
Camping Equipment	Packs	205,920	334,580	415,020	211,320	331,100	398,760	232,220	319,300	387,480	241,160	276,620	271,860	385,340	201,580	34,312,600
	Sleeping Bags	198,400	340,940	423,120	210,680	324,460	421,980	218,780	321,880	415,920	220,540	281,560	277,420	394,500	211,960	34,717,900
	Tents	276,920	459,480	570,240	298,420	425,560	560,880	333,720	420,100	543,420	336,860	370,900	351,720	552,580	284,120	47,235,300
Golf Equipment	Golf Accessories	91,780	190,540	197,560	89,140	187,420	188,700	98,800	183,180	187,160	98,660	158,300	159,240	186,540	92,180	17,004,400
	Irons	26,920	53,940	61,580	26,960	53,760	59,280	29,160	50,120	59,100	31,480	44,220	42,940	57,400	27,820	5,070,100
	Putters	44,800	89,920	97,720	44,120	88,720	96,020	48,100	82,740	97,280	49,420	72,300	69,860	96,440	44,360	8,284,500
	Woods	25,620	52,780	60,840	27,140	51,560	58,020	29,960	49,180	57,760	30,800	43,880	44,260	53,580	29,660	4,984,000
Mountaineering Equipment	Climbing Accessories		657,740	763,760	23,580	613,780	760,440	70,860	567,840	722,260	154,980	420,820	385,820	697,180	362,080	50,577,800
	Rope		188,340	218,440	9,300	173,880	217,260	24,940	159,740	212,400	43,940	122,300	115,460	200,440	100,180	14,562,800
	Safety		103,540	119,780	4,120	96,120	118,680	12,640	89,020	112,680	25,740	66,760	61,200	106,000	60,240	7,960,400
	Tools		446,000	521,980	23,460	412,000	518,880	60,560	373,480	516,000	101,960	279,200	259,420	495,540	236,480	34,700,400
Outdoor Protection	First Aid	361,360	223,260	138,860	351,580	230,900	140,260	339,140	242,860	139,000	311,040	253,560	268,200	149,440	46,720	25,526,200
	Insect Repellents	1,324,760	831,780	544,200	1,327,140	837,500	542,760	1,321,480	848,860	545,540	1,274,400	812,780	873,820	609,420	189,680	95,734,800
	Sunscreen	1,060,380	656,780	437,160	1,052,920	669,860	431,480	1,047,060	681,040	434,580	1,019,920	675,260	722,960	448,780	142,240	76,297,900
Personal Accessories	Binoculars	86,880	136,840	209,580	91,560	133,160	211,580	91,900	128,900	199,520	98,840	115,120	112,920	179,040	110,860	15,639,400
	Eyewear	122,720	187,460	224,920	128,220	184,280	223,680	127,620	183,780	214,480	132,700	162,600	160,900	209,780	105,020	19,232,500
	Knives	302,760	457,700	580,880	310,540	456,200	569,800	316,380	442,480	555,800	310,880	399,820	393,860	510,040	306,880	48,098,400
	Navigation	115,840	201,080	249,740	114,700	199,720	242,800	125,780	190,840	241,900	124,840	168,220	162,460	216,420	134,140	20,263,600
	Watches	235,020	353,660	434,000	230,620	349,720	419,300	236,400	339,380	428,500	210,740	313,320	307,680	404,820	203,420	36,204,000
All Products	5,617,060	7,897,300	8,639,180	5,719,920	7,735,140	8,481,980	5,976,920	7,569,040	8,266,000	6,121,280	6,671,440	6,654,840	8,108,080	4,075,280	791,412,300	



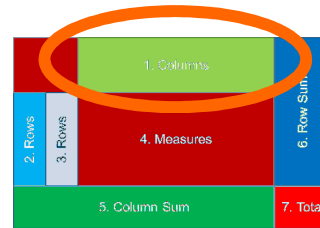
Ease maintenance with Dynamic Query Analyzer

- **Visual display of result flow** to easily troubleshoot dynamic query issues and resolve problems quickly
- **Easy access to logs remotely from dynamic query analyzer** to reduce maintenance time
- **Ability to run reports from dynamic query analyzer** to increase IT efficiency



Demo Report 1 Queries

1. Columns



```

SELECT "Large_Periods"."Year_Key" AS "column0" ,
       "Large_Periods"."Year Name" AS "column1" ,
       "Large_Periods"."Year_Key" AS "C_sortColumn0"
FROM "DQMSampleData"."dbo"."Large_Periods" "Large_Periods" GROUP BY "Large_Periods"."Year_Key",
       "Large_Periods"."Year_Key",
       "Large_Periods"."Year Name" ORDER BY "C_sortColumn0" ASC,
       "column1" ASC,
       "column0" ASC
    
```

	column0	column1	C_sortColumn0
1	1900	1900	1900
2	1901	1901	1901
3	1902	1902	1902
4	1903	1903	1903
5	1904	1904	1904
6	1905	1905	1905
7	1906	1906	1906
8	1907	1907	1907

Timing

Total Time: 694.2989 ms

Working Times:

- 200 XValueDecoration (93.75 ms)
- 199 XSql (46.875 ms)

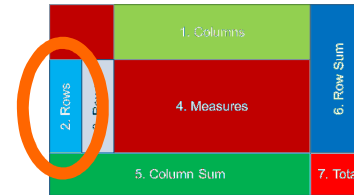
Wait Times:

- 199 XSql (474.21524 ms)
- 200 XValueDecoration (79.458626 ms)



Demo Report 1 Queries

2. Rows - Outer



```

SELECT "Product_Line"."Product_Line_Code" AS "column0" ,
       "Product_Line"."Product Line" AS "column1"
FROM "DQMSampleData"."dbo"."Product_Line" "Product_Line" GROUP BY "Product_Line"."Product_Line_Code",
       "Product_Line"."Product Line" ORDER BY "column1" ASC,
       "column0" ASC
    
```

	column0	column1
1	1	Camping Equipment
2	5	Golf Equipment
3	2	Mountaineering Equipment
4	4	Outdoor Protection
5	3	Personal Accessories

Timing

Total Time: 68.73275 ms

Working Times:

- 146 XSql (15.625 ms)
- 147 XValueDecoration (15.625 ms)

Wait Times:

- 146 XSql (38.057243 ms)
- 147 XValueDecoration (0.574489 ms)



Demo Report 1 Queries

3. Rows - Inner



```

SELECT "Product_Line"."Product_Line_Code" AS "column0" ,
       "Product_Type"."Product_Type_Code" AS "column1" ,
       "Product_Type"."Product_Type" AS "column2" ,
       "Product_Line"."Product_Line" AS "C_sortColumn0"
FROM "DQMSampleData"."dbo"."Product_Line" "Product_Line" INNER JOIN "DQMSampleData"."dbo"."Product_Type" "Product_Type"
ON "Product_Line"."Product_Line_Code" = "Product_Type"."Product_Line_Code" GROUP BY "Product_Line"."Product_Line",
"Product_Line"."Product_Line_Code",
"Product_Type"."Product_Type_Code",
"Product_Type"."Product_Type" ORDER BY "C_sortColumn0" ASC,
"column0" ASC,
"column2" ASC,
"column1" ASC
    
```

	column0	column1	column2	C_sortColumn0
1	1	1	Cooking Gear	Camping Equipment
2	1	5	Lanterns	Camping Equipment
3	1	4	Packs	Camping Equipment
4	1	3	Sleeping Bags	Camping Equipment
5	1	2	Tents	Camping Equipment
6	5	21	Golf Accessories	Golf Equipment
7	5	18	Irons	Golf Equipment
8	5	20	Putters	Golf Equipment
9	5	19	Woods	Golf Equipment
10	2	8	Climbing Accessories	Mountaineering Equipment
11	2	6	Rope	Mountaineering Equipment
12	2	7	Safety	Mountaineering Equipment
13	2	9	Tools	Mountaineering Equipment
14	4	17	First Aid	Outdoor Protection
15	4	15	Insect Repellents	Outdoor Protection
16	4	16	Sunscreen	Outdoor Protection
17	3	13	Binoculars	Personal Accessories
18	3	11	Eyewear	Personal Accessories
19	3	12	Knives	Personal Accessories
20	3	14	Navigation	Personal Accessories
21	3	10	Watches	Personal Accessories

Timing

Total Time: 71.02606 ms

Working Times:

- 260 XSql (15.625 ms)
- 261 XValueDecoration (15.625 ms)

Wait Times:

- 260 XSql (30.773916 ms)
- 261 XValueDecoration (9.002152 ms)



Demo Report 1 Queries

4. Measures



```

SELECT "Large_Periods"."Year_Key" AS "column0" ,
      "Product_Line"."Product_Line_Code" AS "column1" ,
      "Product_Type"."Product_Type_Code" AS "column2" ,
      SUM("MainFact"."Quantity") AS "column3"
FROM "DQMSampleData"."dbo"."Large_Periods" "Large_Periods" INNER JOIN "DQMSampleData"."dbo"."MainFact" "MainFact"
ON "Large_Periods"."Day_Key" = "MainFact"."Order_Day_Key" INNER JOIN "DQMSampleData"."dbo"."Product_Dimension" "Product_Dimension"
ON "Product_Dimension"."Product_Key" = "MainFact"."Product_Key" INNER JOIN "DQMSampleData"."dbo"."Product_Type" "Product_Type"
ON "Product_Type"."Product_Type_Code" = "Product_Dimension"."Product_Type_Code" INNER JOIN "DQMSampleData"."dbo"."Product_Line" "Product_Line"
ON "Product_Line"."Product_Line_Code" = "Product_Type"."Product_Line_Code" GROUP BY "Large_Periods"."Year_Key",
      "Product_Line"."Product_Line_Code",
      "Product_Type"."Product_Type_Code"
    
```

	column0	column1	column2	column3
1	1906	1	1	437600
2	1909	1	1	471560
3	1912	1	1	584980
4	1915	1	1	611800
5	1922	1	1	479020
6	1925	1	1	440260
7	1928	1	1	428340
8	1931	1	1	416780
9	1938	1	1	689600
10	1944	1	1	570500

Timing

Total Time: 14158.743 ms

Working Times:

- 468 XValueDecoration (234.37502 ms)
- 467 XSql (62.5 ms)

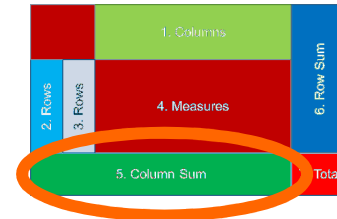
Wait Times:

- 467 XSql (13850.011 ms)
- 468 XValueDecoration (11.857152 ms)



Demo Report 1 Queries

5. Column Sum



```
SELECT "Large_Periods"."Year_Key" AS "column0" ,
    SUM("MainFact"."Quantity") AS "column1"
FROM "DQMSampleData"."dbo"."Large_Periods" "Large_Periods" INNER JOIN "DQMSampleData"."dbo"."MainFact" "MainFact"
ON "Large_Periods"."Day_Key" = "MainFact"."Order_Day_Key" GROUP BY "Large_Periods"."Year_Key"
```

Results Messages

	column0	column1
1	1900	5617060
2	1901	7897300
3	1902	8639180
4	1903	5719920
5	1904	7735140
6	1905	8481980
7	1906	5976920
8	1907	7569040
9	1908	8266000
10	1909	6121280
11	1910	6671140

▼ Timing

Total Time: 2925.896 ms

Working Times:

- 250 XValueDecoration (78.125 ms)

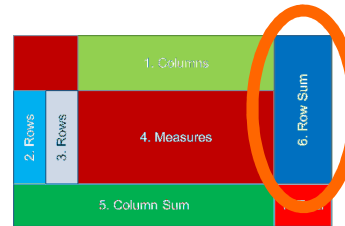
Wait Times:

- 249 XSql (2859.999 ms)
- 250 XValueDecoration (12.228072 ms)



Demo Report 1 Queries

6. Row Sum



```

SELECT "Product Line"."Product Line Code" AS "column0" ,
       "Product Type"."Product Type Code" AS "column1" ,
       SUM("MainFact"."Quantity") AS "column2"
FROM "DQMSampleData"."dbo"."Product Dimension" "Product Dimension" INNER JOIN "DQMSampleData"."dbo"."MainFact" "MainFact"
ON "Product Dimension"."Product Key" = "MainFact"."Product Key" INNER JOIN "DQMSampleData"."dbo"."Product Type" "Product Type"
ON "Product Type"."Product Type Code" = "Product Dimension"."Product Type Code" INNER JOIN "DQMSampleData"."dbo"."Product Line" "Product Line"
ON "Product Line"."Product Line Code" = "Product Type"."Product Line Code" GROUP BY "Product Line"."Product Line Code",
       "Product Type"."Product Type Code"
    
```

	column0	column1	column2
1	1	1	70966700
2	1	2	47235300
3	1	3	34717900
4	1	4	34312600
5	1	5	124038600
6	2	6	14562800
7	2	7	7960400
8	2	8	50577800
9	2	9	34700400
10	2	10	26700000

Timing

Total Time: 5323.313 ms

Working Times:

- 363 XSql (15.625 ms)
- 364 XValueDecoration (15.625 ms)

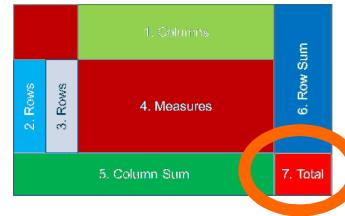
Wait Times:

- 363 XSql (5286.0303 ms)
- 364 XValueDecoration (6.0325 ms)



Demo Report 1 Queries

7. Total



```
SELECT
    SUM("MainFact"."Quantity") AS "column0"
FROM "DQMSampleData"."dbo"."MainFact" "MainFact"
```

Results Messages

	column0
1	791412300

Timing

Total Time: 47203.617 ms

Working Times:

Wait Times:

- 140 XSql (47189.504 ms)
- 141 XValueDecoration (14.113523 ms)



Clear the Cache...

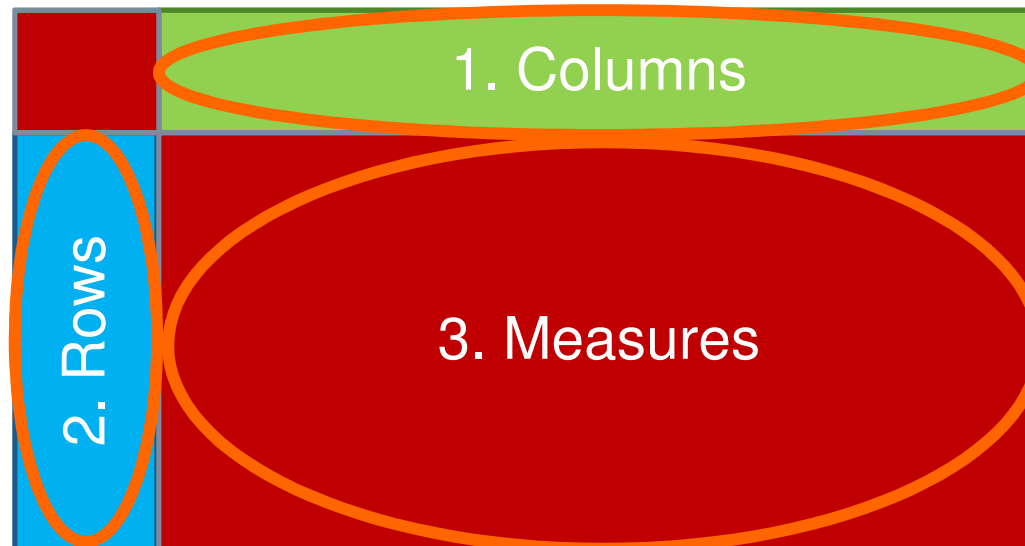
- Administration
- Configuration
- Query Service Caching
- Clear Cache



Demo Report 2 Queries (Xtab no summaries)

Quantity	E-mail	Fax	Mail	Sales visit	Special	Telephone	Web
Camping Equipment	50,027,600	15,045,800	16,079,700	75,184,200	6,045,200	79,104,900	69,783,700
Golf Equipment	5,530,700	1,794,800	2,147,600	8,811,600	1,178,100	8,259,300	7,620,900
Mountaineering Equipment	17,044,300	5,865,300	4,799,900	24,151,400	2,158,800	28,448,000	25,333,700
Outdoor Protection	30,986,900	10,724,000	13,632,500	47,671,400	3,571,400	50,675,100	40,297,600
Personal Accessories	21,222,600	7,443,100	7,205,100	34,772,500	2,720,900	36,340,500	29,733,200

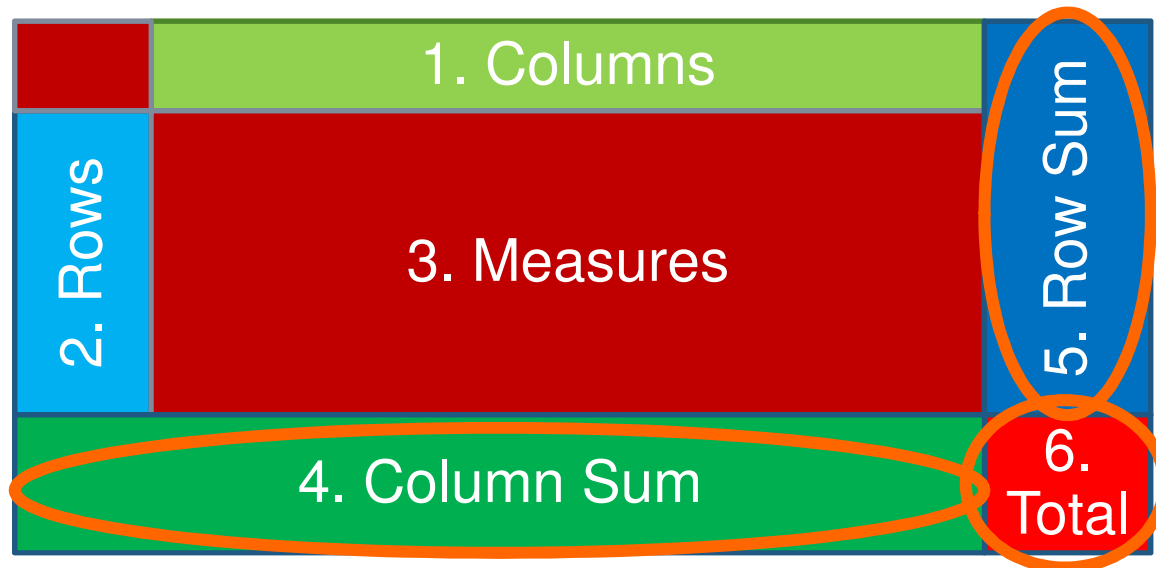
3 Query Profiles



Demo Report 3 Queries (Xtab with summaries)

Quantity	E-mail	Fax	Mail	Sales visit	Special	Telephone	Web	Total
Camping Equipment	50,027,600	15,045,800	16,079,700	75,184,200	6,045,200	79,104,900	69,783,700	311,271,100
Golf Equipment	5,530,700	1,794,800	2,147,600	8,811,600	1,178,100	8,259,300	7,620,900	35,343,000
Mountaineering Equipment	17,044,300	5,865,300	4,799,900	24,151,400	2,158,800	28,448,000	25,333,700	107,801,400
Outdoor Protection	30,986,900	10,724,000	13,632,500	47,671,400	3,571,400	50,675,100	40,297,600	197,558,900
Personal Accessories	21,222,600	7,443,100	7,205,100	34,772,500	2,720,900	36,340,500	29,733,200	139,437,900
Total	124,812,100	40,873,000	43,864,800	190,591,100	15,674,400	202,827,800	172,769,100	791,412,300

HOW MANY QUERIES WILL BE RUN?



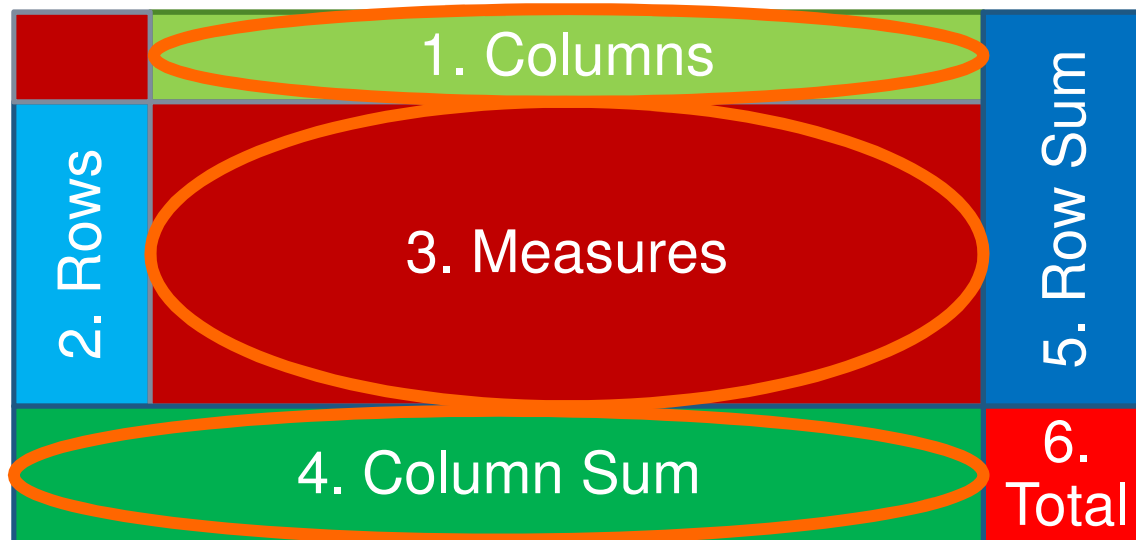
3



Demo Report 4 Queries (Xtab by Time)

Quantity	1900	1901	1902	1903	1904	1905	2006	2007	2008	All Periods
Camping Equipment	1,818,220	3,065,940	3,778,180	1,864,820	2,996,560	3,683,040	2,513,840	3,487,220	1,883,320	311,271,100
Golf Equipment	189,120	387,180	417,700	187,360	381,460	402,020	316,300	393,960	194,020	35,343,000
Mountaineering Equipment		1,395,620	1,623,960	60,460	1,295,780	1,615,260	821,900	1,499,160	758,980	107,801,400
Outdoor Protection	2,746,500	1,711,820	1,120,220	2,731,640	1,738,260	1,114,500	1,864,980	1,207,640	378,640	197,558,900
Personal Accessories	863,220	1,336,740	1,699,120	875,640	1,323,080	1,667,160	1,137,820	1,520,100	860,320	139,437,900
Total	5,617,060	7,897,300	8,639,180	5,719,920	7,735,140	8,481,980	6,654,840	8,108,080	4,075,280	791,412,300

HOW MANY QUERIES WILL BE RUN?



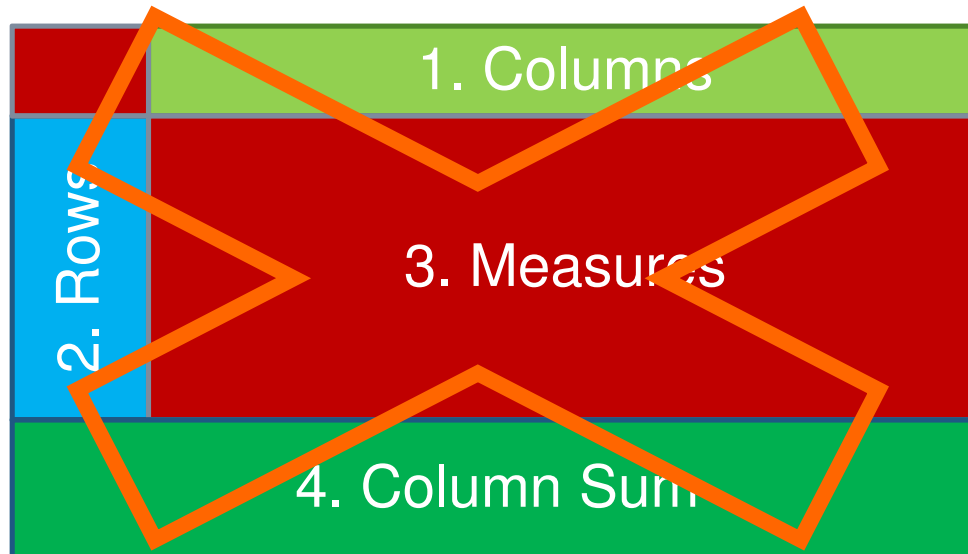
3



Demo Report 5 Queries (Xtab by Time first 10 years)

Quantity	1900	1901	1902	1903	1904	1905	1906	1907	1908	1909
Camping Equipment	1,818,220	3,065,940	3,778,180	1,864,820	2,996,560	3,683,040	1,996,140	2,955,600	3,542,040	2,100,940
Golf Equipment	189,120	387,180	417,700	187,360	381,460	402,020	206,020	365,220	401,300	210,360
Mountaineering Equipment		1,395,620	1,623,960	60,460	1,295,780	1,615,260	169,000	1,190,080	1,563,340	326,620
Outdoor Protection	2,746,500	1,711,820	1,120,220	2,731,640	1,738,260	1,114,500	2,707,680	1,772,760	1,119,120	2,605,360
Personal Accessories	863,220	1,336,740	1,699,120	875,640	1,323,080	1,667,160	898,080	1,285,380	1,640,200	878,000
Total	5,617,060	7,897,300	8,639,180	5,719,920	7,735,140	8,481,980	5,976,920	7,569,040	8,266,000	6,121,280

HOW MANY QUERIES WILL BE RUN?



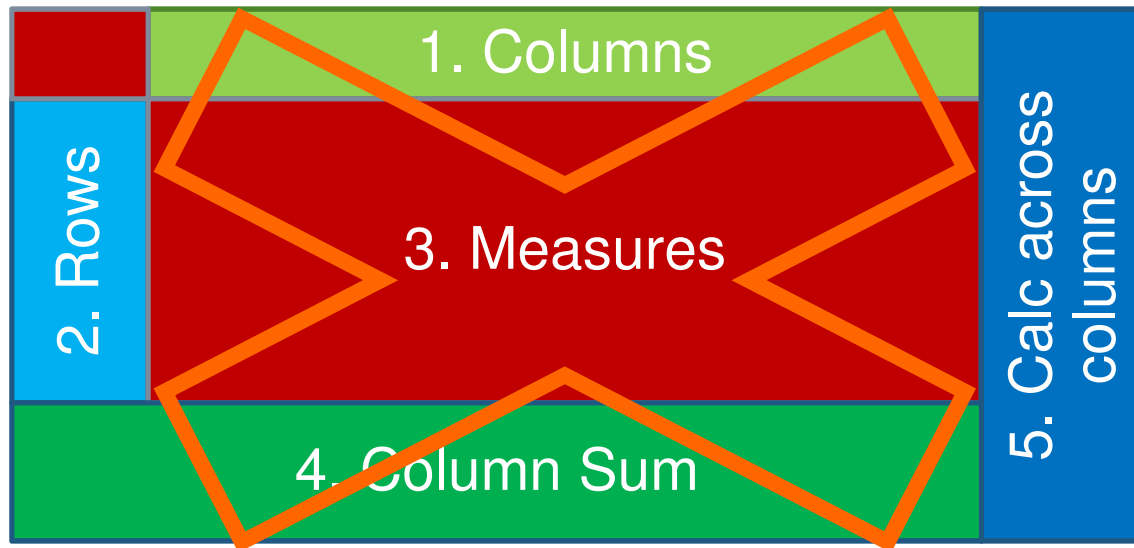
0



Demo Report 6 Queries (Xtab by Time first 10 years with calcs across years)

Quantity	1900	1901	1902	1903	1904	1905	1906	1907	1908	1909	1909 - 1908
Camping Equipment	1,818,220	3,065,940	3,778,180	1,864,820	2,996,560	3,683,040	1,996,140	2,955,600	3,542,040	2,100,940	-1,441,100
Golf Equipment	189,120	387,180	417,700	187,360	381,460	402,020	206,020	365,220	401,300	210,360	-190,940
Mountaineering Equipment		1,395,620	1,623,960	60,460	1,295,780	1,615,260	169,000	1,190,080	1,563,340	326,620	-1,236,720
Outdoor Protection	2,746,500	1,711,820	1,120,220	2,731,640	1,738,260	1,114,500	2,707,680	1,772,760	1,119,120	2,605,360	1,486,240
Personal Accessories	863,220	1,336,740	1,699,120	875,640	1,323,080	1,667,160	898,080	1,285,380	1,640,200	878,000	-762,200
Total	5,617,060	7,897,300	8,639,180	5,719,920	7,735,140	8,481,980	5,976,920	7,569,040	8,266,000	6,121,280	-2,144,720

HOW MANY QUERIES WILL BE RUN?



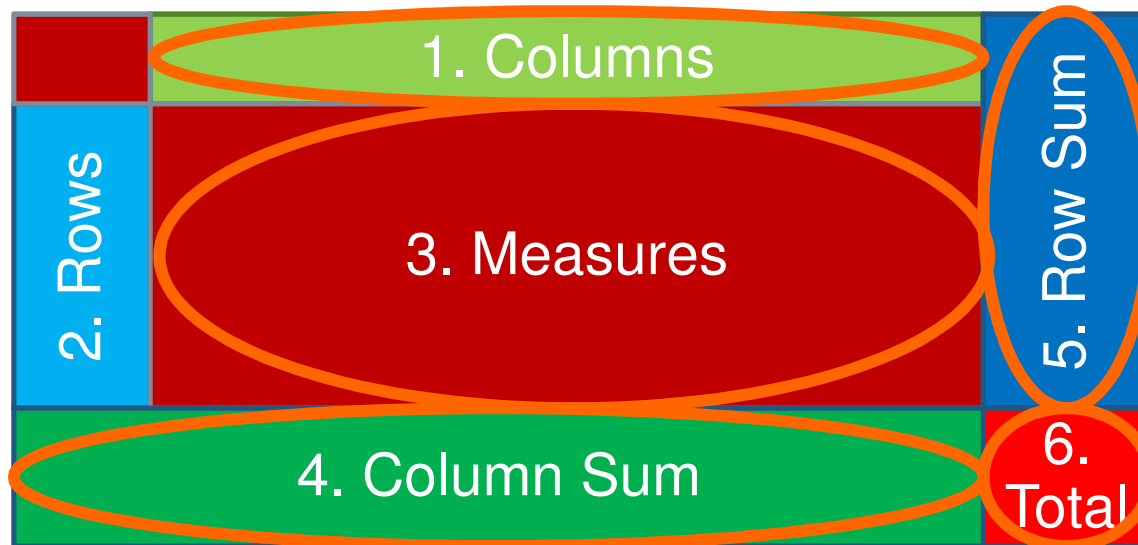
0



Demo Report 7 Queries (Drill to Month)

Quantity	19001201	19001202	19001203	19001204	19001205	19001206	19001207	19001208	19001209	19001210	19001222	19001223	19001224	December 1900
Camping Equipment	2,800	420	1,380	10,920	12,540	11,120	48,120	11,540	720	8,820	8,380	2,700	920	148,060
Golf Equipment	20	140	2,680	720	180	1,140	880	780	1,380	380			500	13,760
Outdoor Protection	200			9,480	1,740	8,800	32,020	3,640			2,100	1,660		72,260
Personal Accessories	1,200		4,360	4,660	1,460	3,000	17,040	3,680	720	2,160	1,120	2,120	360	60,400
Total	4,220	560	8,420	25,780	15,920	24,060	98,060	19,640	2,820	11,360	11,600	6,480	1,780	294,480

HOW MANY QUERIES WILL BE RUN?



5



Demo Report 7 Queries (Drill to Month)



```

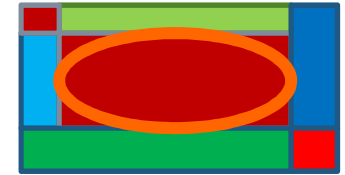
SELECT "D1"."CO" AS "column0" ,
      "D1"."C1" AS "column1" ,
      "D1"."C4" AS "column2" ,
      "D1"."C2" AS "column3" ,
      "D1"."C5" AS "column4" ,
      "D1"."C3" AS "column5" ,
      "D1"."C6" AS "column6" ,
      "D1"."CO" AS "C_sortColumn0" ,
      "D1"."C1" AS "C_sortColumn1" ,
      "D1"."C2" AS "C_sortColumn2" ,
      "D1"."C3" AS "C_sortColumn3"
FROM
(
SELECT "Large_Periods"."Year_Key" AS "CO" ,
      "Large_Periods"."Quarter_Key" AS "C1" ,
      "Large_Periods"."Month_Key" AS "C2" ,
      "Large_Periods"."Day_Key" AS "C3" ,
      (((('Q'+CAST("Large_Periods"."Quarter Number" AS
                CHAR(1))))+' ')+ "Large_Periods"."Year Name") AS "C4" ,
      ((( "Large_Periods"."Month Name" +' ')+ "Large_Periods"."Year Name") AS "C5" ,
      CAST("Large_Periods"."Day_Key" AS

```

	column0	column1	column2	column3	column4	column5	column6	C_sortColumn0	C_sortColumn1	C_sortColumn2	C_sortColumn3
1	1900	19001	Q1 1900	190001	January 1900	19000101	19000101	1900	19001	190001	19000101
2	1900	19001	Q1 1900	190001	January 1900	19000102	19000102	1900	19001	190001	19000102
3	1900	19001	Q1 1900	190001	January 1900	19000103	19000103	1900	19001	190001	19000103
4	1900	19001	Q1 1900	190001	January 1900	19000104	19000104	1900	19001	190001	19000104
5	1900	19001	Q1 1900	190001	January 1900	19000105	19000105	1900	19001	190001	19000105
6	1900	19001	Q1 1900	190001	January 1900	19000106	19000106	1900	19001	190001	19000106
7	1900	19001	Q1 1900	190001	January 1900	19000107	19000107	1900	19001	190001	19000107
8	1900	19001	Q1 1900	190001	January 1900	19000108	19000108	1900	19001	190001	19000108



Demo Report 7 Queries (Drill to Month)



```

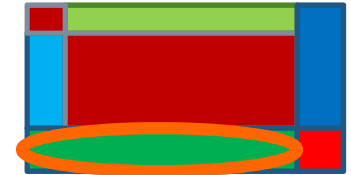
SELECT "Product_Line"."Product_Line_Code" AS "column0" ,
       "Large_Periods"."Year_Key" AS "column1" ,
       "Large_Periods"."Quarter_Key" AS "column2" ,
       "Large_Periods"."Month_Key" AS "column3" ,
       "Large_Periods"."Day_Key" AS "column4" ,
       SUM("MainFact"."Quantity") AS "column5"
FROM "DQMSampleData"."dbo"."Large_Periods" "Large_Periods" INNER JOIN "DQMSampleData"."dbo"."Ma
ON "Large_Periods"."Day_Key" = "MainFact"."Order_Day_Key" INNER JOIN "DQMSampleData"."dbo".
ON "Product_Dimension"."Product_Key" = "MainFact"."Product_Key" INNER JOIN "DQMSampleData".
ON "Product_Type"."Product_Type_Code" = "Product_Dimension"."Product_Type_Code" INNER JOIN
ON "Product_Line"."Product_Line_Code" = "Product_Type"."Product_Line_Code"
WHERE "Large_Periods"."Year_Key" IN
(1900 )AND "Large_Periods"."Quarter_Key" IN
(19004 )AND "Large_Periods"."Month_Key" IN
(190012 )AND "Large_Periods"."Day_Key" IN
(19001217,
 19001216,
 19001215,
 19001214,
 19001213,
 19001212,
 19001211,

```

	column0	column1	column2	column3	column4	column5
1	1	1900	19004	190012	19001201	2800
2	3	1900	19004	190012	19001201	1200
3	4	1900	19004	190012	19001201	200
4	5	1900	19004	190012	19001201	20
5	1	1900	19004	190012	19001202	420
6	5	1900	19004	190012	19001202	140
7	1	1900	19004	190012	19001203	1380



Demo Report 7 Queries (Drill to Month)



```

SELECT "Large_Periods"."Year_Key" AS "column0" ,
       "Large_Periods"."Quarter_Key" AS "column1" ,
       "Large_Periods"."Month_Key" AS "column2" ,
       "Large_Periods"."Day_Key" AS "column3" ,
       SUM("MainFact"."Quantity") AS "column4"
FROM "DQMSampleData"."dbo"."Large_Periods" "Large_Periods" INNER JOIN "DQMSampleData"
ON "Large_Periods"."Day_Key" = "MainFact"."Order_Day_Key"
WHERE "Large_Periods"."Year_Key" IN
(1900 )AND "Large_Periods"."Quarter_Key" IN
(19004 )AND "Large_Periods"."Month_Key" IN
(190012 )AND "Large_Periods"."Day_Key" IN
(19001217,
 19001216,
 19001215,
 19001214,
 19001213,
 19001212,
 19001211,
 19001210,

```

	column0	column1	column2	column3	column4
1	1900	19004	190012	19001201	4220
2	1900	19004	190012	19001202	560
3	1900	19004	190012	19001203	8420
4	1900	19004	190012	19001204	25780
5	1900	19004	190012	19001205	15920
6	1900	19004	190012	19001206	24060
7	1900	19004	190012	19001207	98060



Demo Report 7 Queries (Drill to Month)



```
SELECT "Large_Periods"."Year_Key" AS "column0" ,
       "Large_Periods"."Quarter_Key" AS "column1" ,
       "Large_Periods"."Month_Key" AS "column2" ,
       SUM("MainFact"."Quantity") AS "column3"
FROM "DQMSampleData"."dbo"."Large_Periods" "Large_Periods" INNER JOIN "DQMSampleData"."dbo"."MainFact" "MainFact"
ON "Large_Periods"."Day_Key" = "MainFact"."Order_Day_Key"
WHERE "Large_Periods"."Year_Key" IN
      (1900 )AND "Large_Periods"."Quarter_Key" IN
      (19004 )AND "Large_Periods"."Month_Key" IN
      (190012 )GROUP BY "Large_Periods"."Year_Key",
      "Large_Periods"."Quarter_Key",
      "Large_Periods"."Month_Key"
```

Results Messages

	column0	column1	column2	column3
1	1900	19004	190012	294480



Demo Report 7 Queries (Drill to Month)



```

SELECT "Product_Line"."Product_Line_Code" AS "column0" ,
       "Large_Periods"."Year_Key" AS "column1" ,
       "Large_Periods"."Quarter_Key" AS "column2" ,
       "Large_Periods"."Month_Key" AS "column3" ,
       SUM("MainFact"."Quantity") AS "column4"
FROM "DQMSampleData"."dbo"."Large_Periods" "Large_Periods" INNER JOIN "DQMSampleData"."dbo"."MainFact" "MainF
ON "Large_Periods"."Day_Key" = "MainFact"."Order_Day_Key" INNER JOIN "DQMSampleData"."dbo"."Product_Dimer
ON "Product_Dimension"."Product_Key" = "MainFact"."Product_Key" INNER JOIN "DQMSampleData"."dbo"."Product
ON "Product_Type"."Product_Type_Code" = "Product_Dimension"."Product_Type_Code" INNER JOIN "DQMSampleData
ON "Product_Line"."Product_Line_Code" = "Product_Type"."Product_Line_Code"
WHERE "Large_Periods"."Year_Key" IN
      (1900 )AND "Large_Periods"."Quarter_Key" IN
      (19004 )AND "Large_Periods"."Month_Key" IN
      (190012 )GROUP BY "Product_Line"."Product_Line_Code",
      "Large_Periods"."Year_Key",
      "Large_Periods"."Quarter_Key",
      "Large_Periods"."Month_Key"
    
```

	column0	column1	column2	column3	column4
1	1	1900	19004	190012	148060
2	3	1900	19004	190012	60400
3	4	1900	19004	190012	72260
4	5	1900	19004	190012	13760



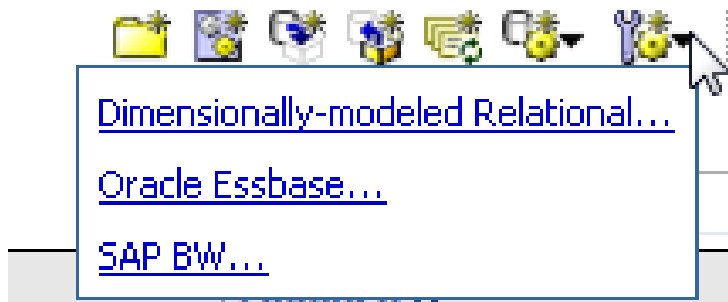
How to Use Dynamic Query...Before You Begin...

- Dynamic Query Mode is a new capability within Cognos Business Intelligence V10
 - Recommended for new application development
- Existing applications require migration from Compatible to Dynamic Query Modes
- Use Lifecycle Manager to determine effort required to migrate existing applications
 - Can be performed without permanently altering package
 - Both visual and performance differences can quickly be obtained
- Not all existing applications will benefit from migrating to Dynamic Query Mode
 - ensure benefits will be worth the effort before beginning.
- Ensure Dynamic Query Mode conformance meets your needs
 - Packages containing multiple data sources must all be supported by Dynamic Query



Clearing or Resetting the Cache

- Memory caches can be cleared to prevent using outdated data
- Clearing the cache can be scheduled and configured as required
- Can clear selected data sources or package (wildcard * allowed)
- Available under the New Query Service Administration option (Configuration > Content Administration)
- Can clear cache or write cache state (to view current state of the caches)



Operation:

- Clear cache
- Clear cache
- Write cache state



Data Source Preparation – Cognos Connection

- Relational data sources - copy the vendor's JDBC drivers
- OLAP sources, install vendor's full / thick client
- Start the Cognos services
- Create new data source or edit existing to add a JDBC connection

General **Connection** Permissions

Specify the parameters for the connection.

Type:
IBM DB2 [Test](#)

Isolation level:
 Use the default object gateway
 Specify a value:
Cursor stability

Connection string:
^User ID: ^?Password: ;LOCAL;D2;DSN=GS_DB;UID=%s;PV

CLI **JDBC**

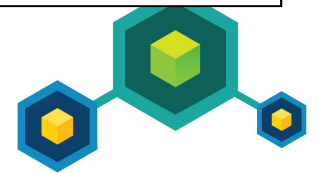
Edit the parameters to build a DB2 (driver: com.ibm.db2.jcc.DB2Driver) connection string.

Enable JDBC connection

Server name:
localhost

Port number:
50000

Database name:
GS_DB



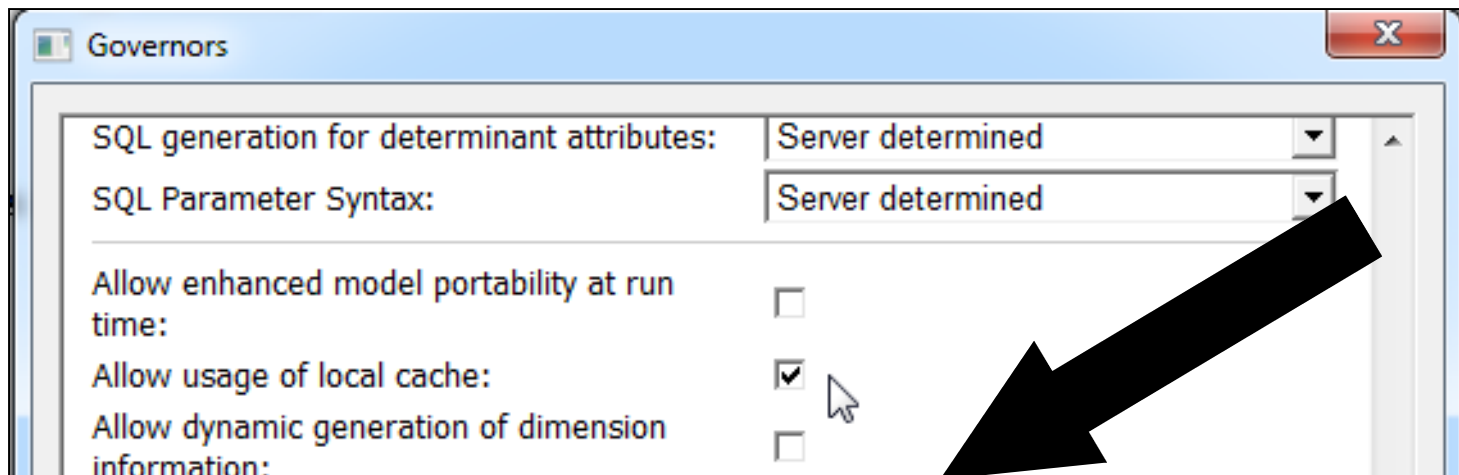
Bypassing the DQM Data Cache

- Some reports must be current. They cannot show outdated / cached data
- Often, the data does not need to be up-to-the-minute; it simply must be relatively fresh
- If data must always be live, one option is to leave the package as CQM
- There are 2 ways to bypass cached data while using DQM
 - Turn off the Local Cache in Framework Manager before publishing the DQM package
 - Turn off Local Cache in individual report Queries



Bypassing the DQM Data Cache in FM

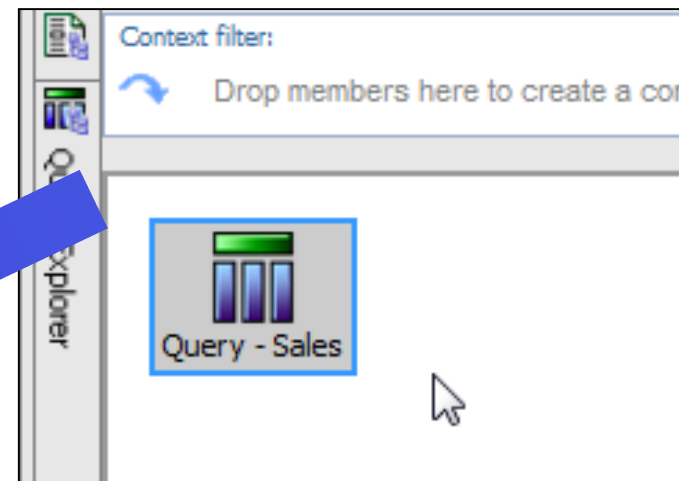
- To publish a package in DQM that does not use the cache
 - Disable the governor “Allow usage of local cache”
 - Publish the DQM package
 - If you wish other DQM packages to use caching – re-enable the governor
- Disabling the local cache affects all DQM queries generated through the published package



Bypassing the DQM Data Cache in an RS Query

- Another option available to Report Studio authors is to disable Use Local Cache on individual report queries.
- This is a property of the Query, not the report

Cross Product Allowed	
Use SQL With Clause	
Use Local Cache	
Execution Method	(Default)
Use for Parameter Info	No
SQL Join Syntax	Yes
Use SQL Parameters	



Summary

- Dynamic Query Mode should be considered when starting new projects
- Enabling DQM on existing projects will require a migration
- Use Lifecycle Manager to understand ROI of migrating existing applications to DQM
- Dynamic Query Mode facilitates
 - Substantial performance improvements via
 - Optimized, multi-phase SQL
 - Hybrid query processing
 - Security-aware caching of metadata, data, and query plans and balanced local processing
 - Suppression
 - Consistent OLAP experience across data sources and studios
 - Member sorting
 - FIRST / LAST and NULL values
 - Aggregate computations
 - NULLs as zeroes in calculations



