UNIVERSE DESIGN BEST PRACTICES

Roxanne Pittman, InfoSol

May 8, 2014

© 2014 InfoSol

INFOSOL®
seeing beyond™

SEVEN PRINCIPLES OF UNIVERSAL DESIGN

BY THE CENTER FOR UNIVERSAL DESIGN (CUD) NORTH CAROLINA STATE UNIVERSITY

1. *Equitable use*.
   The design is useful and marketable to people with diverse abilities.
2. *Flexibility in Use*.
   The design accommodates a wide range of individual preferences and abilities.
3. *Simple and intuitive*.
   Use of the design is easy to understand, regardless of the user's experience, knowledge, language skills, or current concentration level.
4. *Perceptible information*.
   The design communicates necessary information effectively to the user, regardless of ambient conditions or the user's sensory abilities.
5. *Tolerance for error*.
   The design minimizes hazards and the adverse consequences of accidental or unintended actions.
6. *Low physical effort*.
   The design can be used efficiently, comfortably, and with a minimum of fatigue.
7. *Size and space for approach and use*.
   Appropriate size and space for approach.

Architectural principles apply to Universe design, too!

©2014 InfoSol

CONNECTIVITY

- Use Native Connectivity
  - Always use the vendor's native connectivity where possible. Avoid using generic drivers, e.g. ODBC for Oracle. You always get better performance and more parameters when using the vendors own drivers.
- Avoid ODBC
  - When accessing Microsoft SQL Server databases use an OLEDB connection rather than an ODBC connection. The connection is faster. OLEDB connections store their parameters in the connection definition itself, whereas an ODBC connection refers to a predefined ODBC connection name which must be defined on every PC and server.

©2014 InfoSol

UNIVERSE DEVELOPMENT

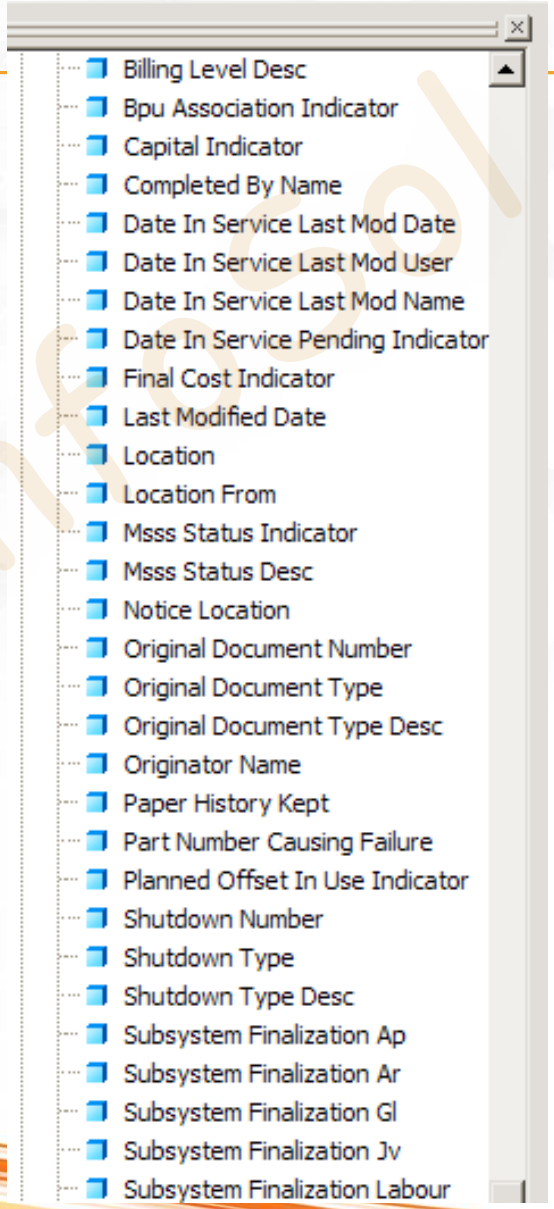
- The most important stages in designing a successful Universe :
  - Data analysis
  - User requirements
  - Design
  - Deploy

- 80% of the development time of a Universe should be spent on the first 2 steps
- Users should be heavily involved in development. Take advantage of subject-matter experts
- If this is done well, then implementation and maintenance should be quick and easy

©2014 InfoSol

FOLDERS & SUBFOLDERS

- **Folders**
  - Each folder should be uniquely named and have an appropriate business description.
- **Subfolders**
  - Try not to nest subfolders too deep - reduces the navigability and usability
  - However, consider organizing subfolders so that users will not have to scroll down the query panel when they expand a folder

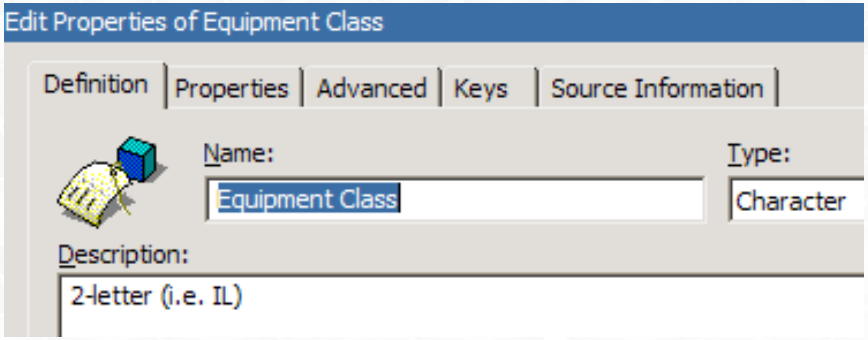


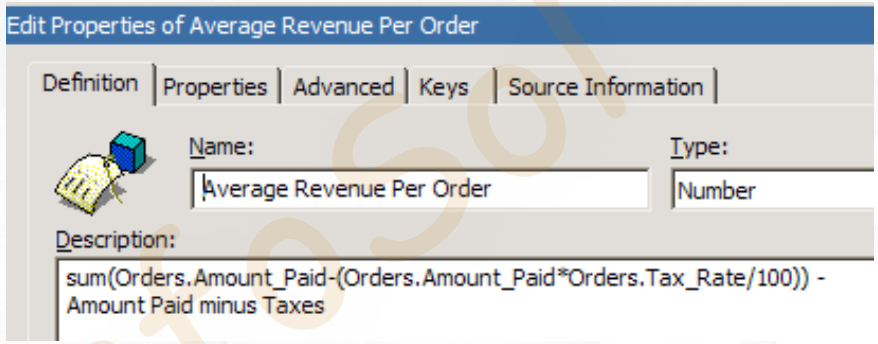
OBJECTS

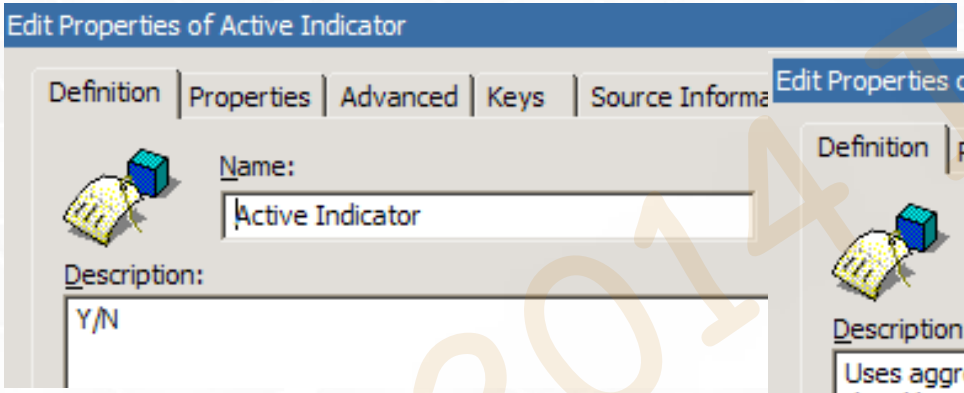
- Objects
  - Every object should be uniquely named and have an appropriate business description.
  - Consistent naming conventions – company wide
  - Consider how long the name looks in query panel and how easy it will be to find in the list
  - If there is a need to have more than one of the “same’ object, like “Year”, give more description, for example, “Order Year” ; “Shipped Year”
  - **Use clear, intuitive business terms**
  - **Description considerations:**
    - Describe object in business terms
    - Describe actual SQL
    - Describe suggested use
    - Describe which input screen it came from
    - Give example of format, for example: FY07-08

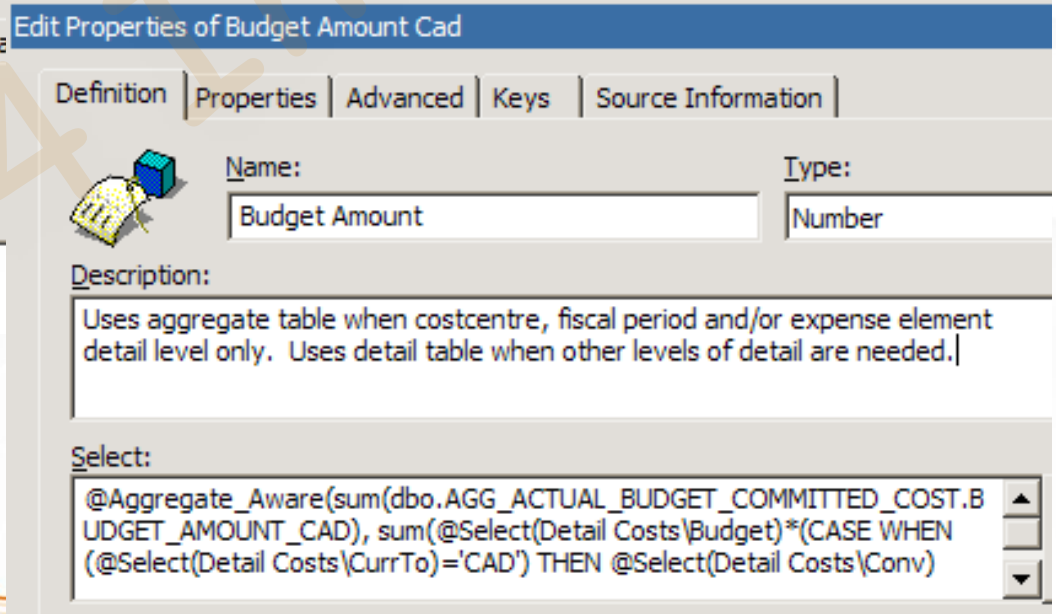
©2014 InfoSol

DESCRIPTION EXAMPLES









©2014 InfoSol

LOV'S

- Associate list of values for objects that have:
  - **relatively static distinct values**
  - **small - moderate number of distinct values, maybe up to 100 values**

- For objects with more values, consider custom list of values with prompted/cascading list of values

©2014 InfoSol

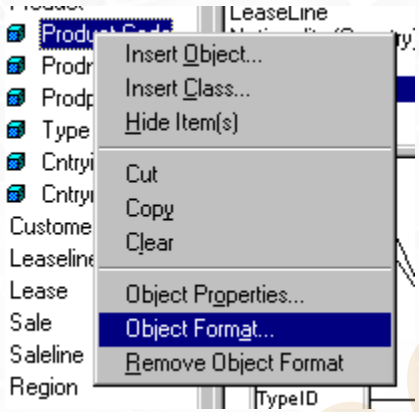
LOV'S

- However, to speed Up Report Design Time, consider *Removing* LOV's for:
  - Measures (default behavior)
  - Dates
  - Meaningless values – ID's, periods, Y/N indicators, etc.

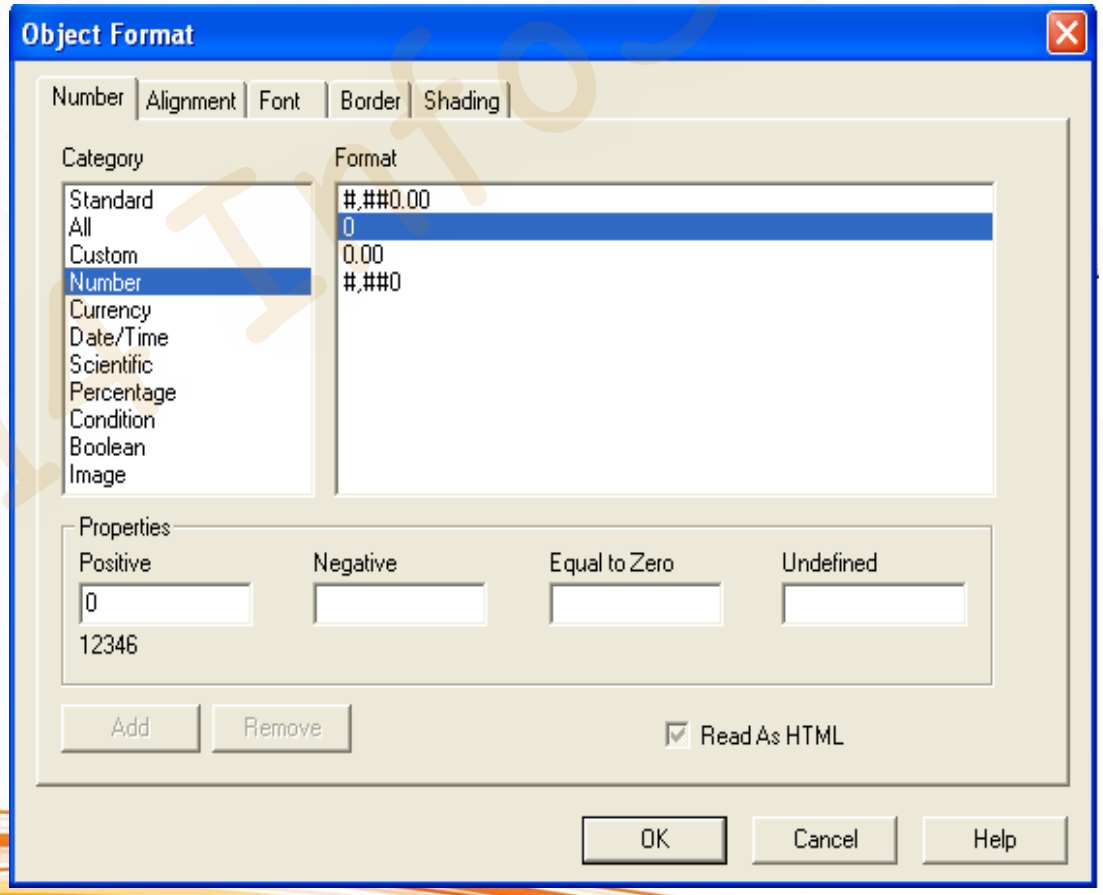
©2014 InfoSol

OBJECT FORMATTING

- Apply a default format to the Object
  - Ex : a *"Number of"* object should have no decimals



Right-click on the Object's name



©2011 InfoSol

OBJECT FORMATTING

- Format:
  - **Currency for amounts. Consider no cents where appropriate.**
  - **Counts: as #,##0 and no decimals**
  - **Format dates without the timestamp**
  - **ID's: Set 0 for objects of type number where object should project as whole numbers … for example, cost-center or customer ID number**

©2014 InfoSol

OBJECT FORMAT EXAMPLE

Formatting these 3 objects would save one user about 15 clicks

Year	SKU number	Sales revenue
2,001	115,121	9422
2,001	116,256	45
2,001	119,427	45
2,001	122,709	217
2,001	128,390	5679
2,001	128,969	45
2,001	136,786	2214
2,001	137,638	137
2,001	138,627	31455
2,001	139,590	91
2,001	140,515	371

Year	SKU number	Sales revenue
2001	115121	$9,422
2001	116256	$45
2001	119427	$45
2001	122709	$217
2001	128390	$5,679
2001	128969	$45
2001	136786	$2,214
2001	137638	$137
2001	138627	$31,455
2001	139590	$91
2001	140515	$371

©2014 InfoSol

CUSTOMIZING OBJECTS

- Consider creating common report requirements such as:
  - **MTD, QTD and YTD measures**
  - **Current and Prior Period measures**
  - **Current Period and Same Period Prior Year measures**

©2014 InfoSol

DATES

- Consider having a generic Date table in every universe
  - Aliased and used over and over
  - Actual date key to link to other dates
  - Create a Set of date Objects
    - Today
    - Yesterday
    - This Month
    - Last Month
    - Today minus 10 days
    - First of month
    - Last of month

©2014 InfoSol

CONDITIONS

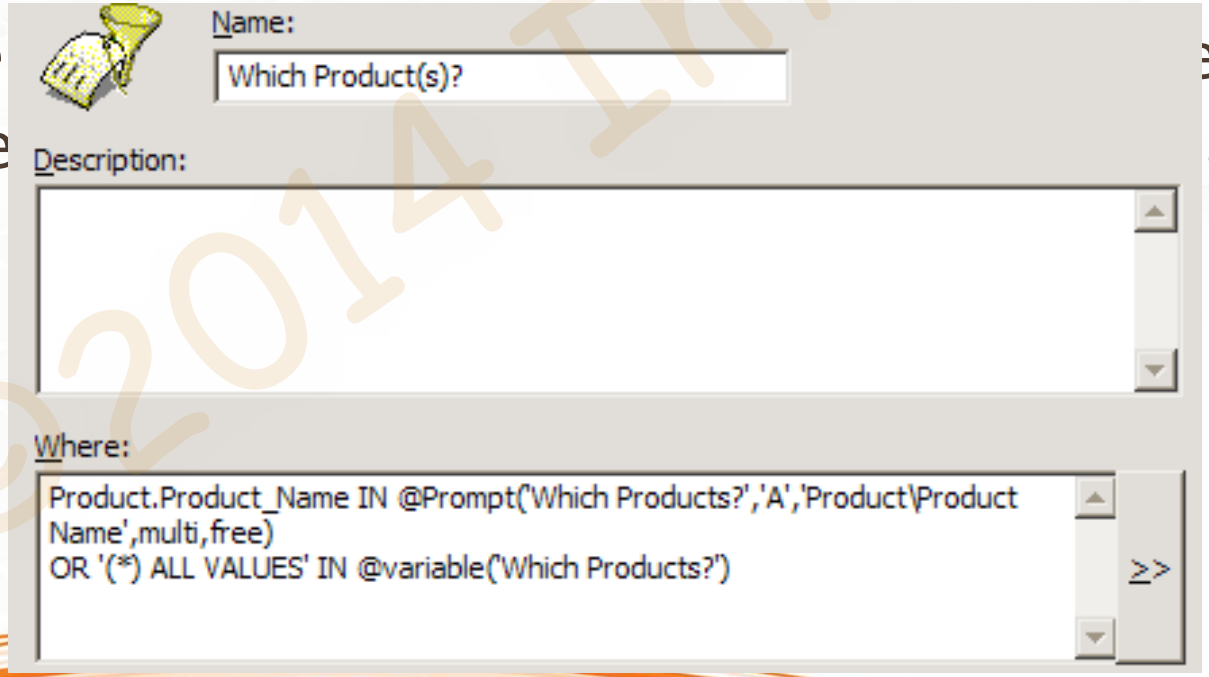
- Create condition objects as necessary - it should be the exception that users have to manually build a condition.
  - Interactive filters will use the prompt feature to as the user to enter information, for example “Which years?”.
  - Static filters will apply a restriction without any visible interaction, for example “Current Year”.

©2014 InfoSol

CONDITIONS

- Use pre-defined conditions rather than applying “Where clause” restrictions on objects.

- Conside ead of multiple an ‘ALL’ option.



©2014 InfoSol

CONDITIONS

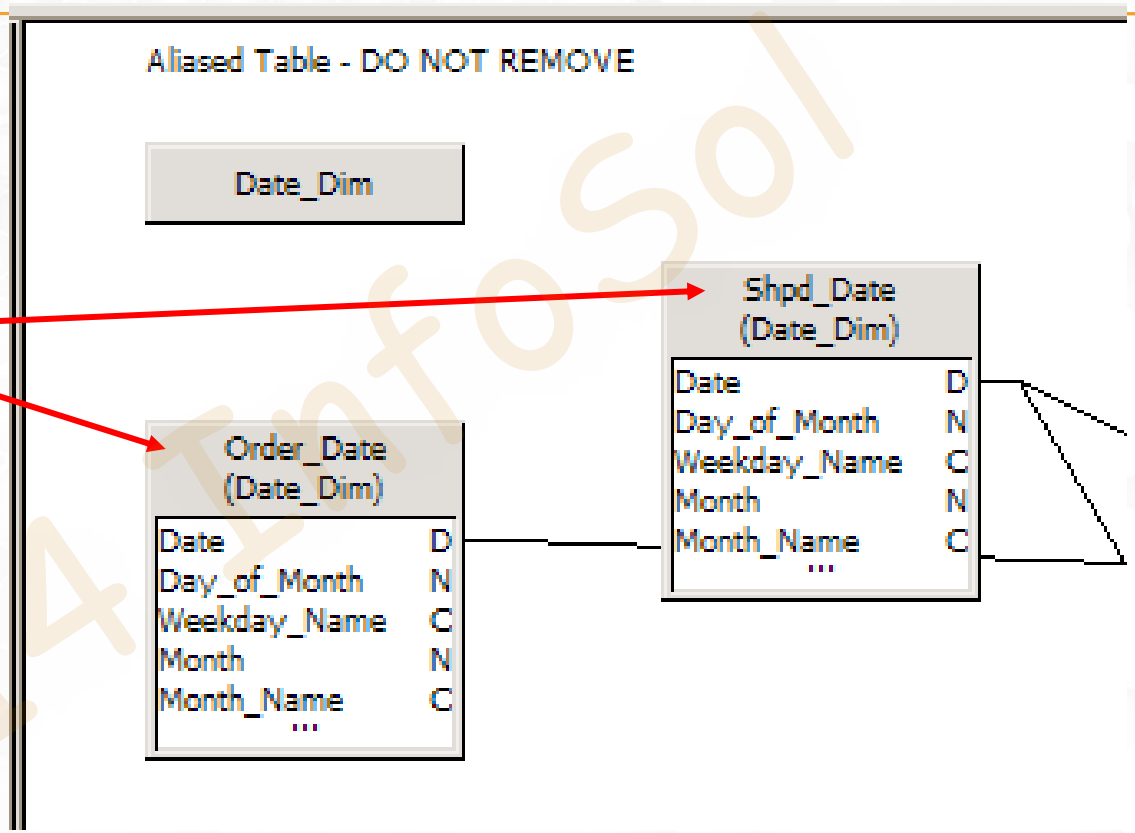
- Give good descriptions for each pre-defined condition.

- If Condition is resulting in a Prompt, make sure the associated dimension Object has a LOV.

©2014 InfoSol

ALIASED TABLES

- Alias Tables should be named with proper functional use.

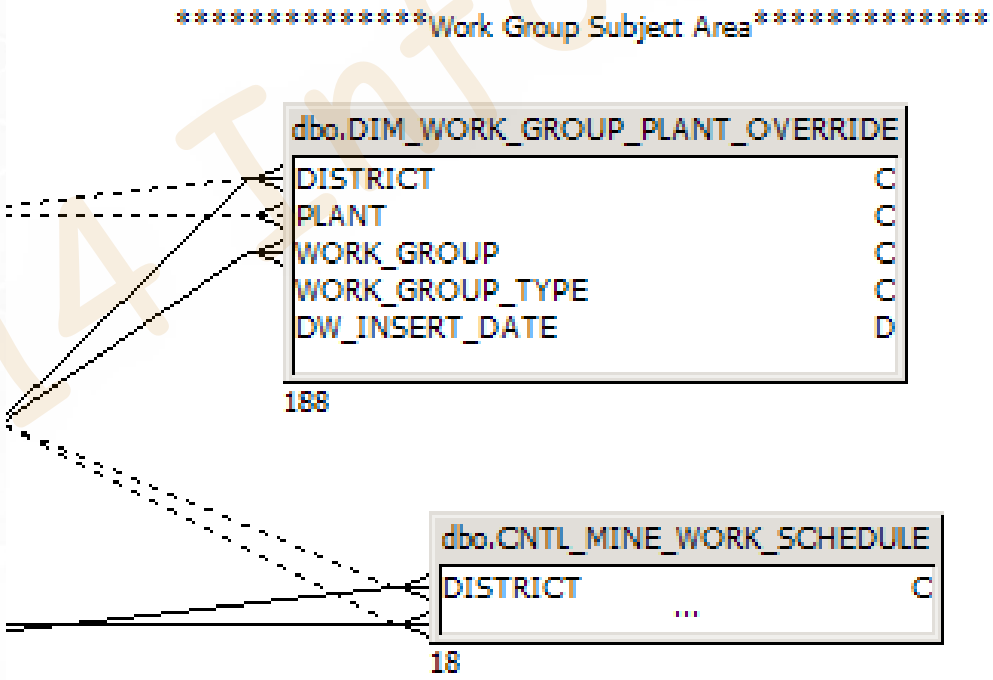


- Consider putting "orphaned" Alias tables at top of structure pane and Labeling for easier maintenance

©2014 InfoSol

STRUCTURE AREA

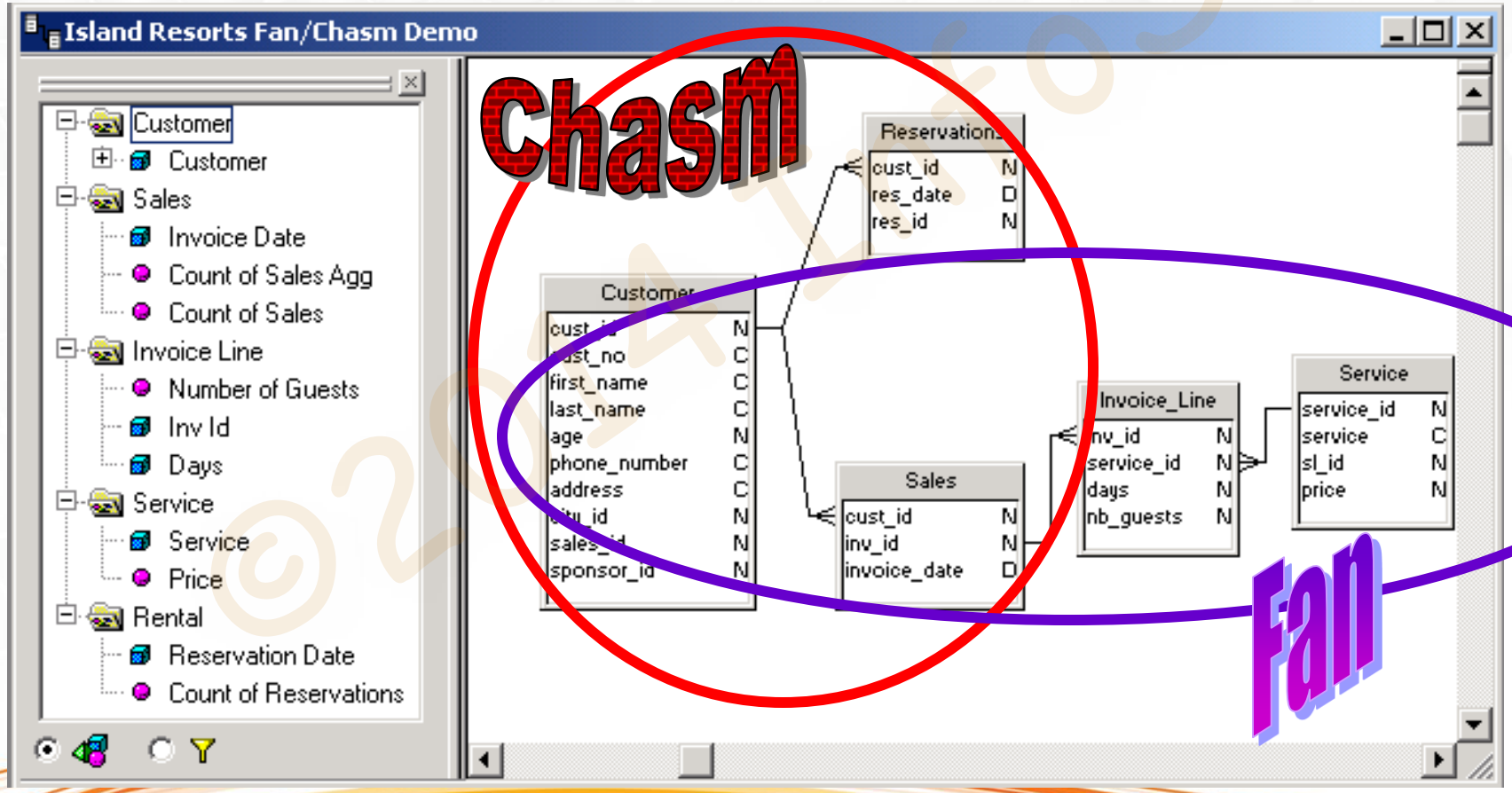
- **Consider adding labels to structure area in large universes for easier/quicker analysis & maintenance**



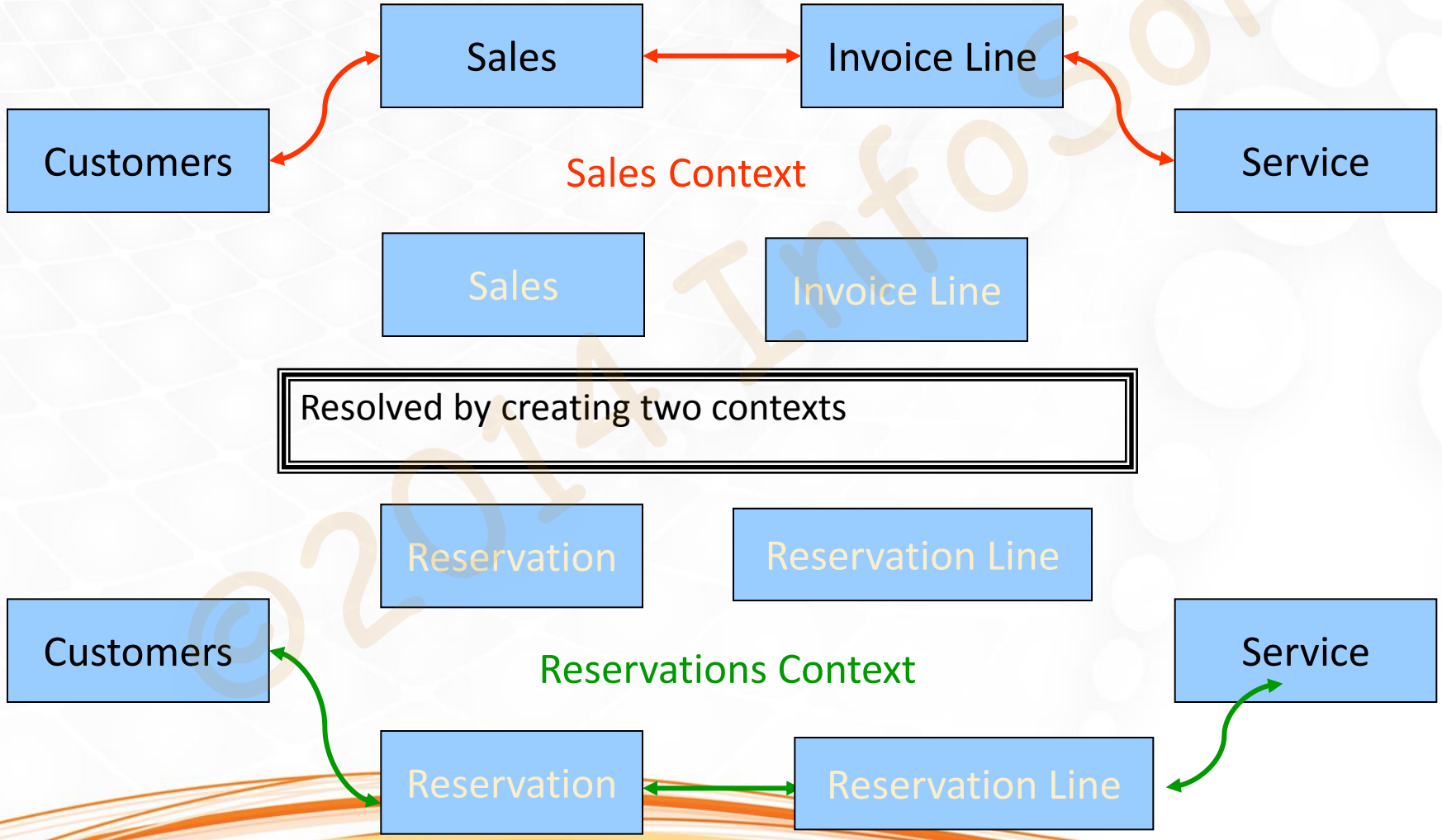
©2014 InfoSol

TOLERANCE FOR ERROR

- Understand fan & chasm traps from training and user guide. Learn how to detect and resolve …



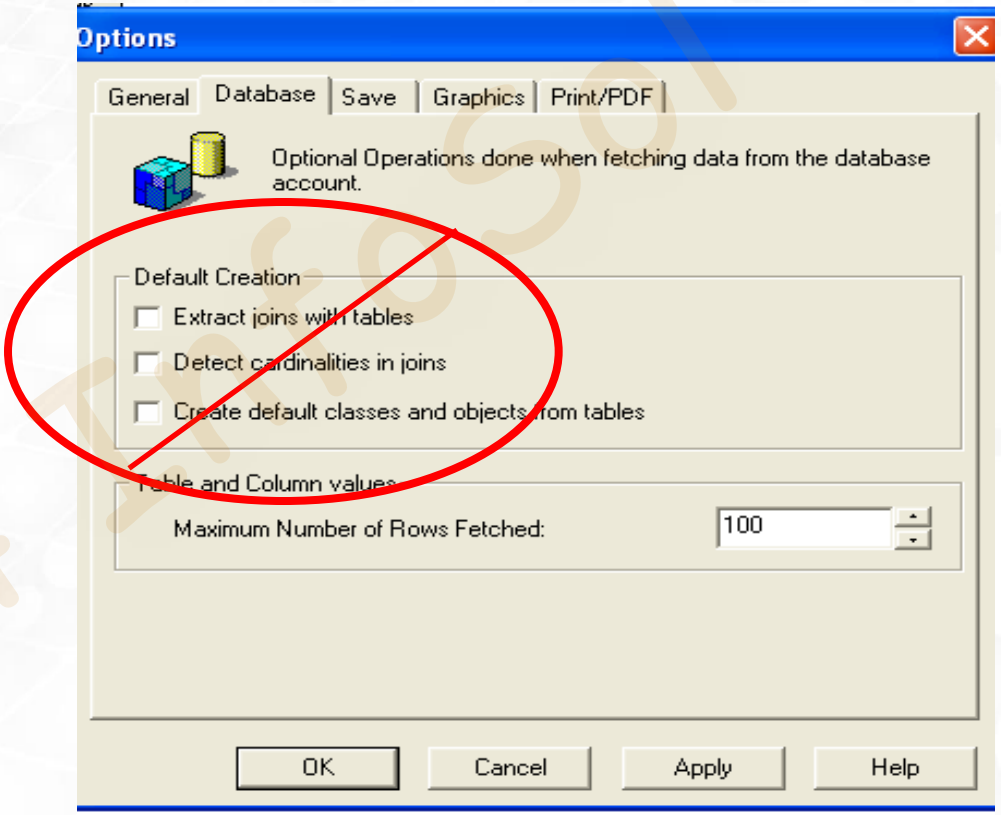
DON'T BE AFRAID TO USE CONTEXT



©2014 Infosol

AVOID THESE AUTOMATIC OPTIONS

- Avoid most automatic options
  - Extract joins with tables
  - Detect Cardinalities in joins
  - Create default classes and objects from tables



©2014 Infosol

PARAMETER SETTINGS

- Consider modifying these universe parameters
  - Turning on automatic sort of LOVs
  - Turning on ANSI92 SQL – Moves Joins to the From clause, which can decrease the quantity of data parsed by the database

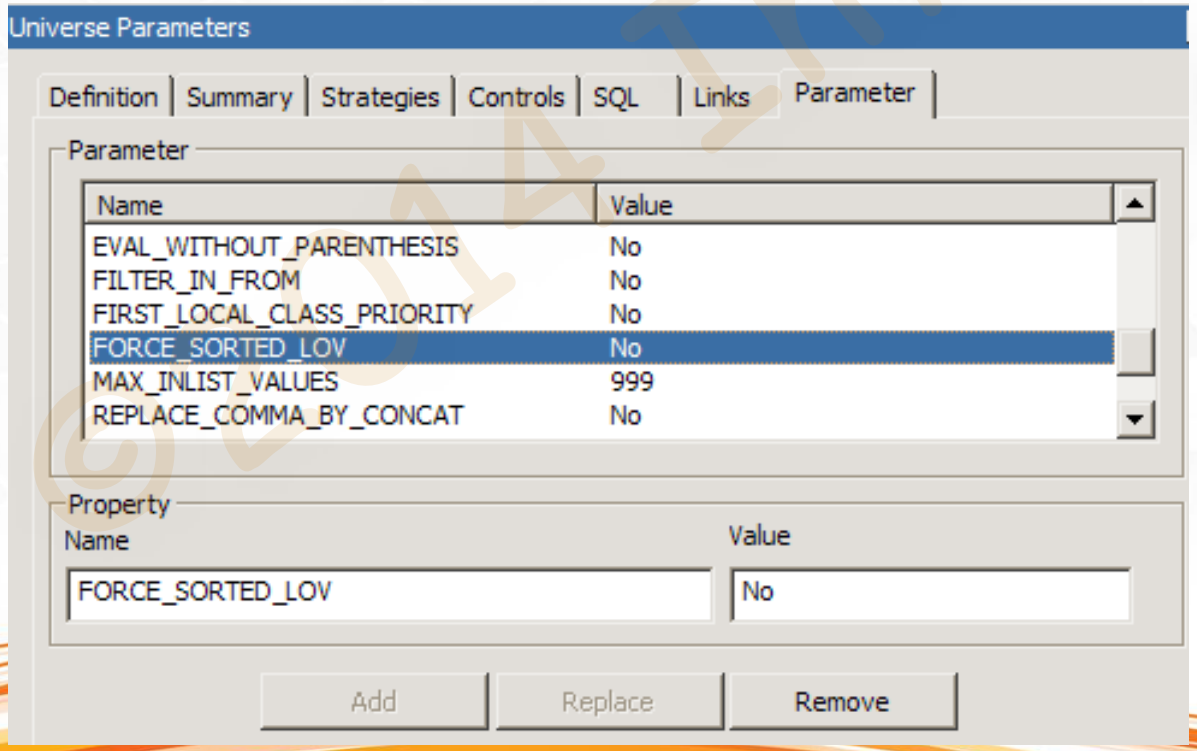
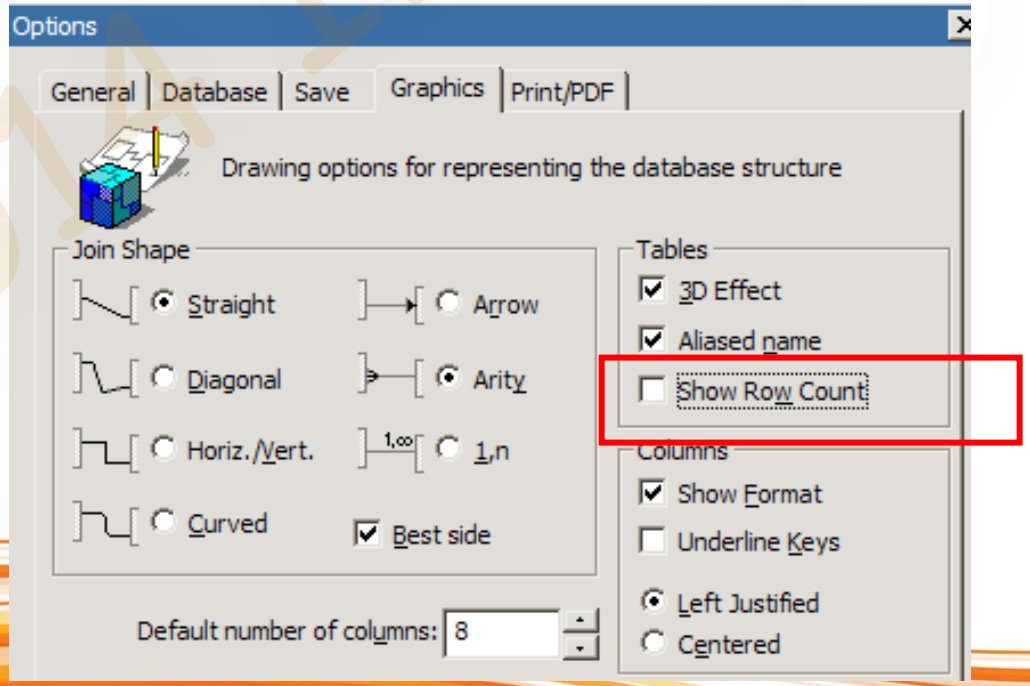


TABLE ROW COUNTS

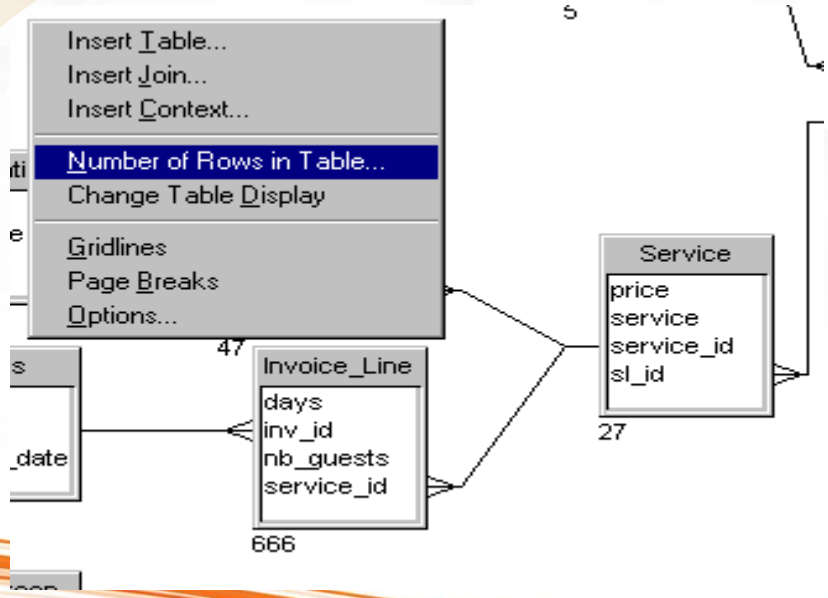
- Display Table Row Counts
  - If data is missing from a particular table it is easier to spot
  - If you are using validation tools (like Check Integrity) it needs the row counts to check the cardinality of the joins.



©2014 InfoSol

NUMBER OF ROWS IN TABLE

- Number of Rows in Table – Query Optimizing
  - Right-click on the window background to show the size of the tables in your structure.
  - This information is used by Business Objects to order multiple joins in a query, from highest limiting row number (large tables) to lowest limiting row number (smaller tables).
  - Modify manually to force Business Objects to custom order the joins in the Where clause. Talk to DBA to determine best order.

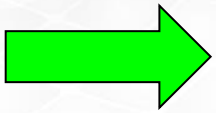


©2014 InfoSol

CONSIDER TARGET AUDIENCE

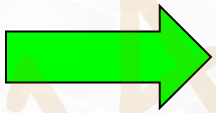
Who will use the universe?

- IT Personnel



- Large, complex universe *may* be okay

- Business Analysts & Mainstream Users

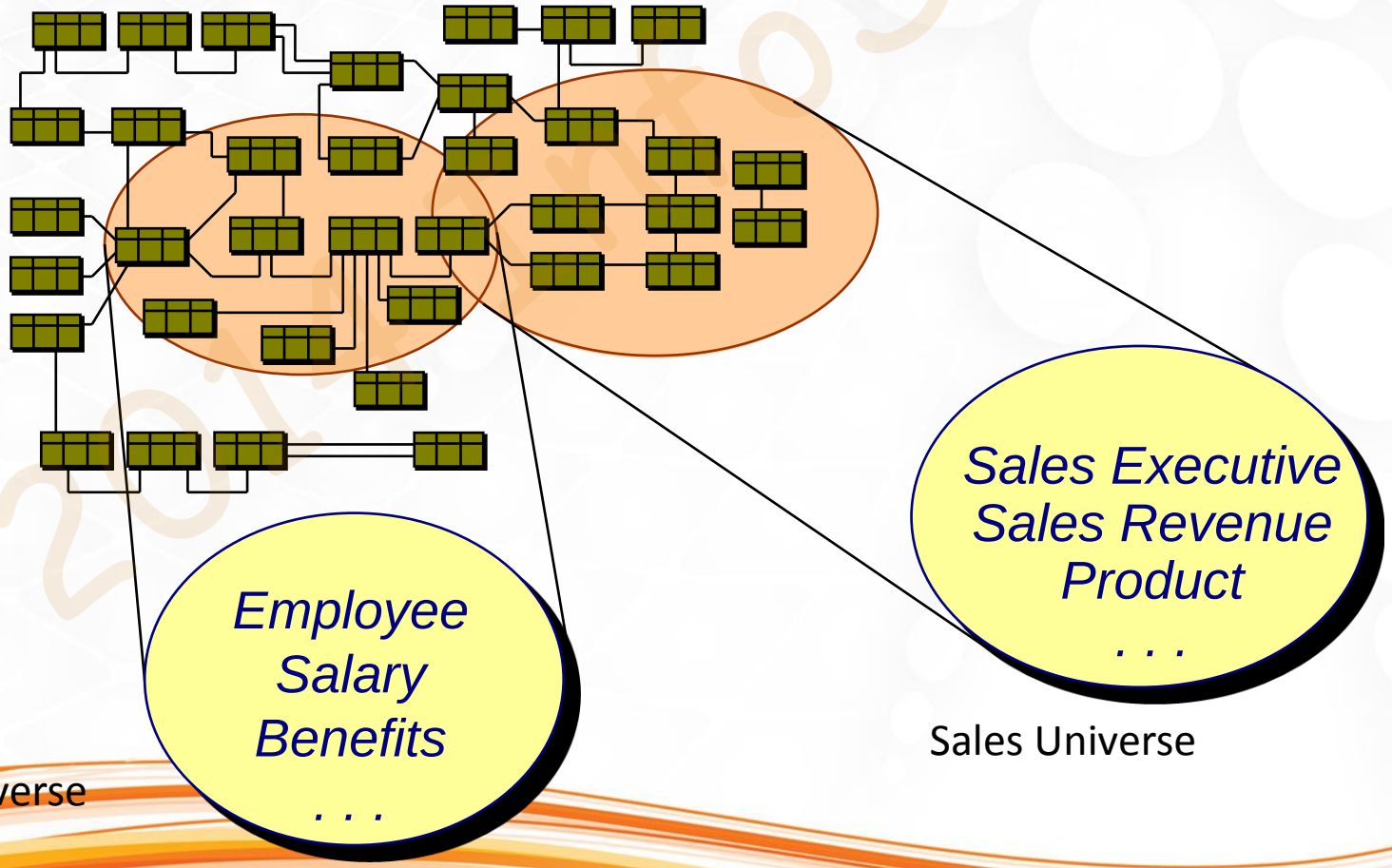


- Smaller, focused universe
- Descriptions on ALL Objects
- Custom list of values
- Custom objects for common variables
- Custom Pre-defined Conditions

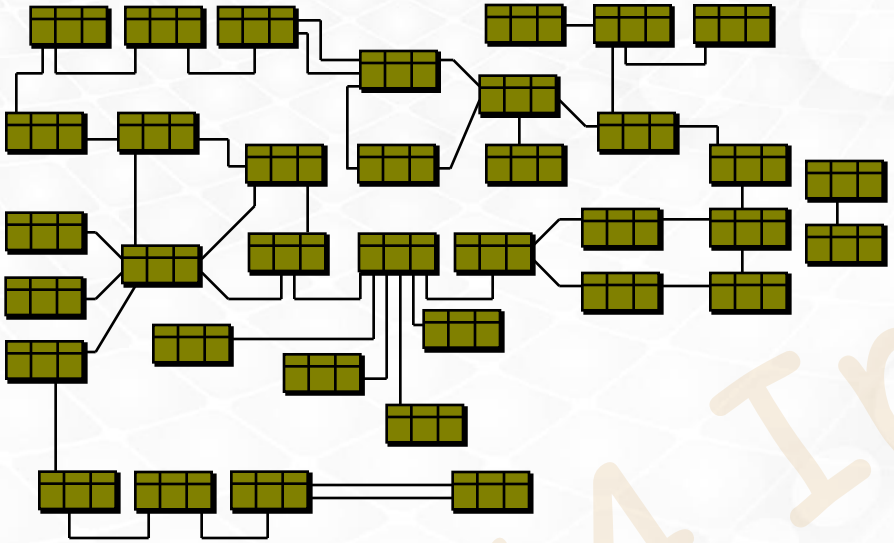
©2014 InfoSol

FOCUSED PURPOSE

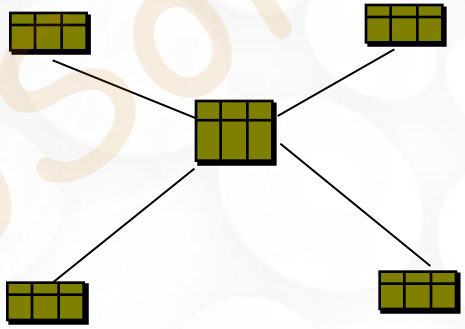
A Universe should be a focused Business-Oriented Data Representation defined for a specific group of Users



PROCESSING DATA EFFICIENTLY



Transactional Database (OLTP)



Star Schema
(OLAP)

- BusinessObjects will work with either a normalized transactional database or a data warehouse “Star” Schema. As with any query system, it works best against a Star Schema.
- If reports are taking a long time to process consider pushing for a data warehouse.

©2014 InfoSol

OPTIMIZE PERFORMANCE

- Use derived tables to reduce complex queries or in place of views or procedures.

- Use aggregate tables /materialized views with aggregate awareness set up to improve query performance

©2014 InfoSol

WORK WITH YOUR DBA

- A DBA has a bag of tricks available to optimize your queries.
  - Query Analyzer : Can tell you how long your query is taking and point you to where you will need indexes.
  - Indexes : Will order or provide fast pointers to specific data for quicker look ups on commonly used objects.
  - Partitioning : Archived data is often not used as often as Current data. Partitioning tables will allow you to search just recent data for fast return, while still allowing access to archived data.
- A DBA is your best friend!

©2014 InfoSol

CHECK YOUR WORK!

- Save yourself trouble - DO an "Integrity Check" before Exporting the Universe.

©2014 InfoSol

SEVEN PRINCIPLES OF UNIVERSAL DESIGN

BY THE CENTER FOR UNIVERSAL DESIGN (CUD) NORTH CAROLINA STATE UNIVERSITY

Remember, make your universes….

1. *Equitable*
2. *Flexible*
3. *Simple and intuitive*
4. *Perceptible information*
5. *Tolerance for error*
6. *Low physical effort*
7. *Size and space for approach and use*

© 2014 InfoSol

QUESTIONS?

©2014 InfoSol