IFPUG 2004

Data Warehouses and Function points

Chris Kohnz

25 Years and ...

1

Agenda

- Terminology
- Structure
- Counting Transactions
- Counting Data Stores

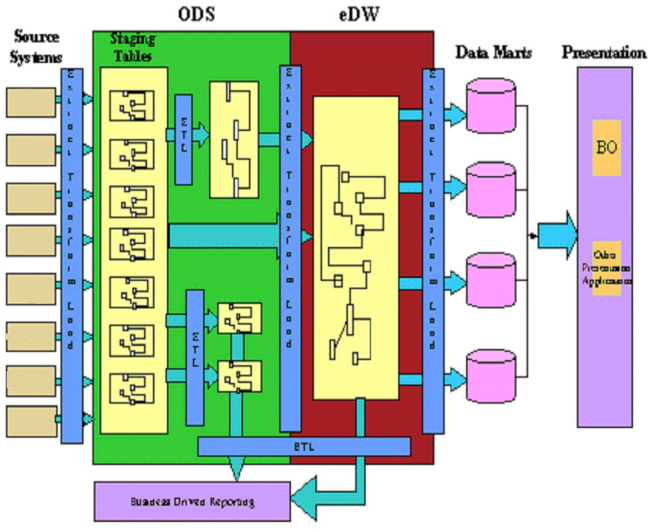
2

What is hard?

- New set of terms
- Similar data in different places
- Getting ‘sucked’ into the physical world

3

DW Structure



4

Terms

- Source system
- Staging Area
- eDW
- Data Mart
- ETL

5

Boundaries rules (IFPUG CPM 4.2)

- The boundary is determined based on the user's view. The focus is on what the user can understand and describe.
- The boundary between related applications is based on separate functional areas as seen by the user, not on technical considerations.
- The initial boundary already established for the application or applications being modified is not influenced by the counting scope.

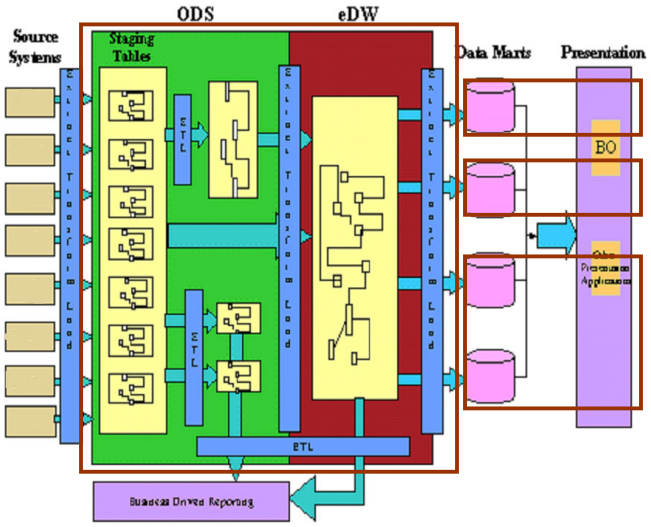
6

Boundaries rules

- User view is key
- IFPUG manual also lists hints to validate against during the boundary discussion.
  - Look at how groups of data are being maintained
  - Look at associated measurement data, such as effort, cost, and defects. The boundaries for function points and other measurement data should be the same.

7

Boundary example



8

FPA

- Elementary Process importance
- Processing logic

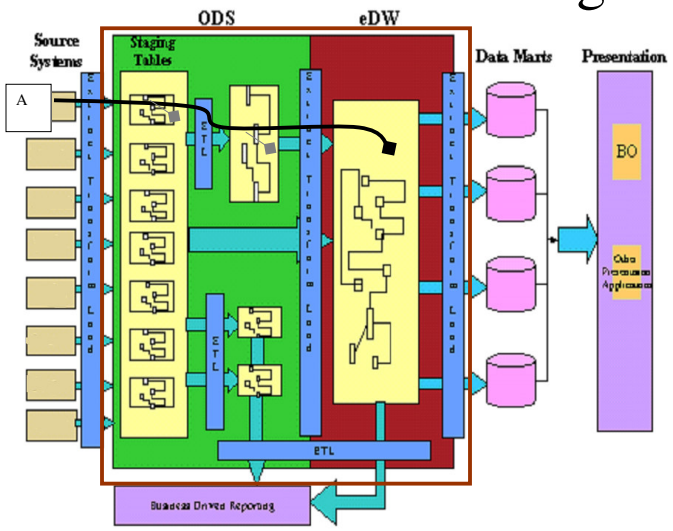
9

ETL and transactions

- Extract
- Transformation
- Load

10

Transaction counting



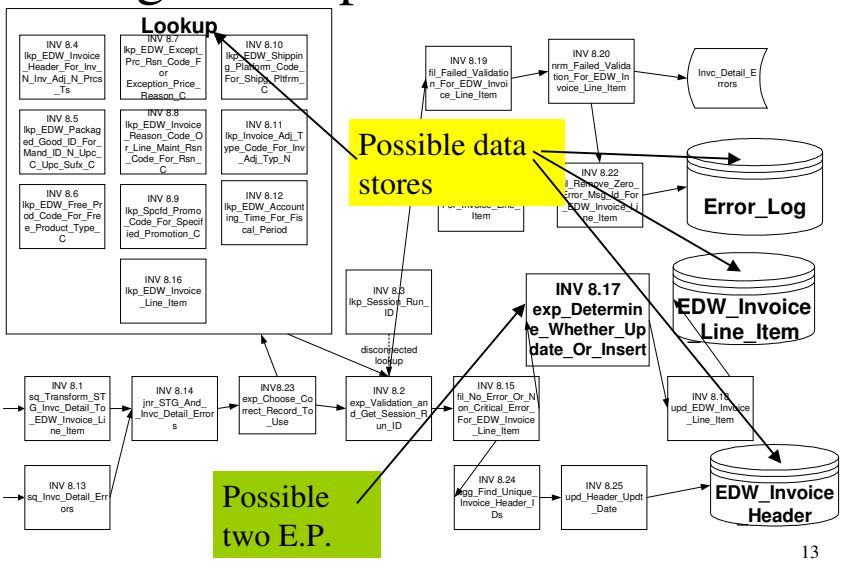
11

Transactions

- ETL Map
  - Overview information
  - Source/Target
  - Required transformations
  - High level process overview
  - Process description
  - Source to Target Field Matrix

12

High level process overview



13

Invoice ETL map

Source to Target Field Matrix

Target Table	Target Column	Datatype	Source Table	Source Column	Datatype	Data Issues/Quality
EDW_Invoice_Line_Item	Invoice_Line_Item_ID	Number(10)		Sequence Generator		EDW Primary key
EDW_Invoice_Line_Item	Invoice_N	Varchar2(7)	STG_Invc_Detail	Inv_N	Varchar2(7)	Source/STG Primary key
EDW_Invoice_Line_Item	Invoice_Adjustment_N	Number(5)	STG_Invc_Detail	Inv_Adj_N	Number(5)	Source/STG Primary key
EDW_Invoice_Line_Item	Invoice_Line_N	Number(5)	STG_Invc_Detail	Inv_Lin_N	Number(5	Source/STG Primary key
EDW_Invoice_Line_Item	Process_Ts	Da				Source/STG Primary key
EDW_Invoice_Line_Item	Invoice_Header_ID	Nu 0)		der	(1 0)	
EDW_Invoice_Line_Item	Packaged_Good_ID	Number(10)	EDW_Packaged_Good	Packaged_Good_ID	Number(10)	
EDW_Invoice_Line_Item	**Fiscal_Period**	**Number (6)**	**STG_Invc_Detail, EDW_Month**	**Use Actg_Mth_N + Actg_Yr_N to look up Fiscal_Period**	**Number (2)** **Number (4)**	
EDW_Invoice_Line_Item	**Load_Order_N**	**Varchar2(7)**	**STG_Invc_Detail**	**Ord_N, Entry_D**	**Varchar2(7**	**Look up shipment_id on EDW_Order_Header (by ord_n, entry_d). Look up load_ord_n on EDW_Shipment (by shipment_id)**
			EDW_Order_Header	**Shipment_Id**		
			EDW_Shipment	**Load_Ord_N**		
EDW_Invoice_Line_Item	Order_Type_C	Varchar2(2)	STG_Invc_Detail	Ord_N, Entry_D	Varchar2(2)	Look up order_type_c on EDW_Order_Header (by ord_n, entry_d)
			EDW_Order_Header	Order_Type_C		
EDW_Invoice_Line_Item	Submit_Method_C	Varchar2(2)	STG_Invc_Detail	Ord_N, Entry_D	Varchar2(2)	Look up submit_method_c on EDW_Order_Header (by ord_n, entry_d)

Derived field

14

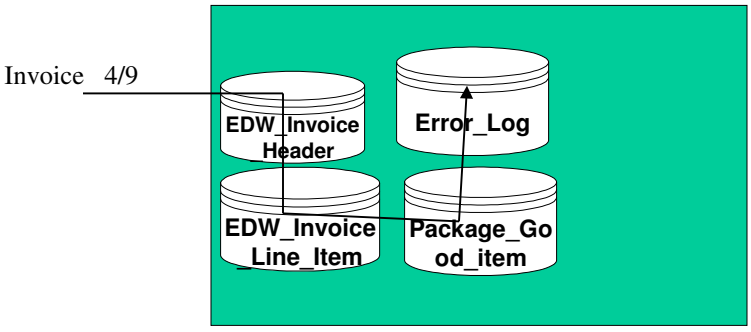
DET identification

Target Table	Target Column	Datatype	Source Table	Source Column	Datatype
EDW_Invoice_Line_Item	Invoice_Line_Item_ID	Number(10)		Sequence Generator	
EDW_Invoice_Line_Item	Invoice_N	Varchar2(7)	STG_Invc_Detail	Inv_N	Varchar2(7)
EDW_Invoice_Line_Item	Invoice_Adjustment_N	Number(5)	STG_Invc_Detail	Inv_Adj_N	Number(5)
EDW_Invoice_Line_Item	Invoice_Line_N	Number(5)	STG_Invc_Detail	Inv_Lin_N	Number(5)
EDW_Invoice_Line_Item	Process_Ts	Date	STG_Invc_Detail	Prcs_Ts	Date
EDW_Invoice_Line_Item	Invoice_Header_ID	Number(10)	EDW_Invoice_Header	Invoice_Header_ID	Number(10)
EDW_Invoice_Line_Item	Packaged_Good_ID	Number(10)	EDW_Packaged_Good	Packaged_Good_ID	Number(10)
EDW_Invoice_Line_Item	Fiscal_Period	Number(6)	STG_Invc_Detail, EDW_Month	Use Actg_Mth_N + Actg_Yr_N to look up Fiscal_Period	Number(2) Number(4)
EDW_Invoice_Line_Item	Load_Order_N	Varchar2(7)	STG_Invc_Detail EDW_Order_Header EDW_Shipment	Ord_N, Entry_D Shipment_Id Load_Ord_N	Varchar2(7
EDW_Invoice_Line_Item	Order_Type_C	Varchar2(2)	STG_Invc_Detail EDW_Order_Header	Ord_N, Entry_D Order_Type_C	Varchar2(2)
EDW_Invoice_Line_Item	Submit_Method_C	Varchar2(2)	STG_Invc_Detail	Ord_N, Entry_D	Varchar2(2)

15

Invoice transaction

- What do we have so far?



16

Table types

- Facts
- Dimensions
- Aggregates
- Metadata
- Error Logs
- Release or publish tables

17

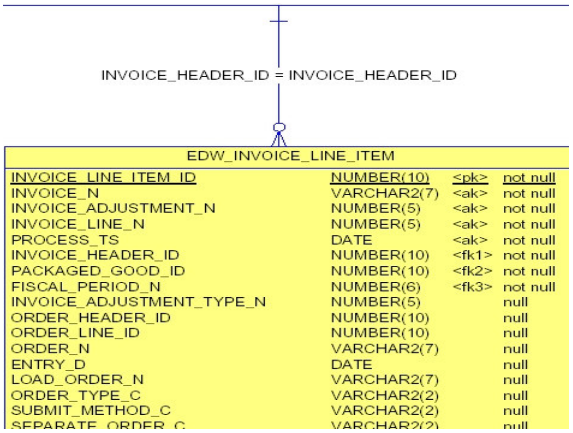
Dimensional Model

- Fact tables
- Dimensions

18

Fact tables

EDW_Invoice_Header



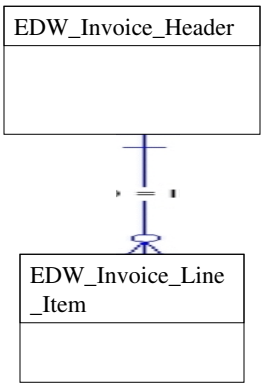
19

Dimension tables

EDW_PACKAGED_GOOD : 1			
PACKAGED_GOOD_ID	NUMBER(10)	<pk>	not null
ITEM_ID	NUMBER(10)	<ak>	not null
PRODUCT_STATUS_C	VARCHAR2(1)		null
PACKAGE_INDICATOR_N	NUMBER(1)		null
UCC_EAN_N	NUMBER(1)		null
EAN_N	NUMBER(14)		null
NUMBER_SYSTEM_CHARACTER_N	NUMBER(1)		not null
MANUFACTURER_ID_N	NUMBER(9)		not null
UPC_C	NUMBER(5)		not null
UPC_SUFFIX_C	VARCHAR2(2)		not null
CHECK_DIGIT_N	NUMBER(1)		not null
LONG_DESCRIPTION_NA	VARCHAR2(50)		null
PACKAGED_GOOD_TYPE_C	VARCHAR2(3)		null
ORDERABLE_F	VARCHAR2(1)		null
CONSUMER_UNIT_F	VARCHAR2(1)		null
UNIT_LOAD_F	VARCHAR2(1)		null
PRIMARY_PACKAGE_F	VARCHAR2(1)		null
EXTERNAL_ITEM_F	VARCHAR2(1)		null
PRIMARY_GROSS_WEIGHT_N	NUMBER(8,4)		null

20

Table counting



1. Identify the entities that should be considered for counting.
2. Identify the user view (=business view) of the data grouping by investigating

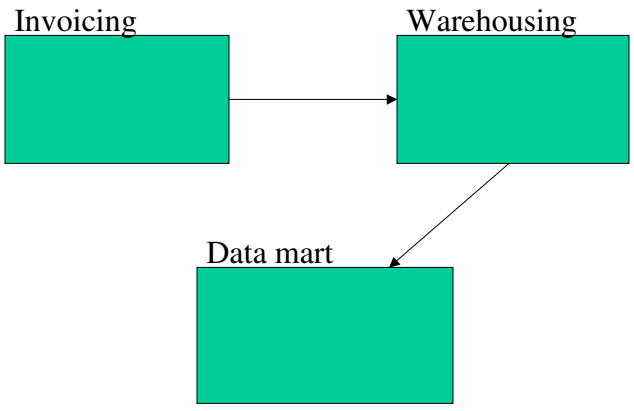
a) How the data is accessed as a group by elementary processes within the application boundary

b) The relationships between the entities and their interdependency based on business rules

3. Classify each identified logical file as ILF or EIF

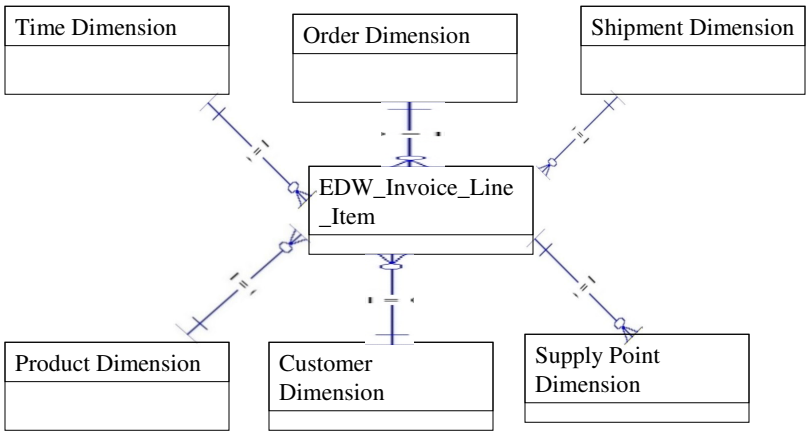
21

Data mart consideration boundary



22

Data mart dimensional model



23

Mapping matrix

Warehouse					
Function Point transaction	**Type**	**FTRs**	**DETs**	**ETL mappings**	**Comments**
				Inv 7	
				Inv 8	
				Inv 9	
				Inv 10	
				Inv 11	
				Inv 13	
				Inv 14	
				Inv 15	
				Iinv 16	
Load Invoice	EI	>3	>16	Inv 4	
Data store		**RETs**	**DETs**	**Comments**	
		Invoice header			
Invoice	ILF	Invoice Line item	>51		
Package Good Items	EIF	Package good items	>51		
Data Mart					
Function Point transaction	**Type**	**FTRs**	**DETs**	**ETL mappings**	**Comments**
Not covered					
Data store		**RETs**	**DETs**	**Comments**	
Time	?				
Order	?				
Shipment	?				
Product	?				
Customer	?				
Supply Point	?				
Invoice	?				

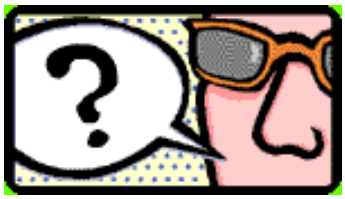
24

Summarize

- Application boundaries
- Understand the terminology used for this domain of applications

25

Questions



26

Sources/References

- Kimball, Reeves, Ross, Thornthwaite, *The Data Warehouse Lifecycle Toolkit:Expert methods for Desighning, Developing, and Deplying Data Warehouses,* WILEY 1998
- IFPUG, *FP-380 Function Point Counting Workshop: Counting Data Warehouse Applicatoins*
- Santillo, Luca, *Size & Estimation of Data Warehouse Systems*, Data Processing Organization
- IFPUG, *Function Point Counting Practices Manual Release 4.2*

27