A Study of a Health Enterprise Information System, Part 5 - Database Relational Schema and Data Tables

Jon Patrick, PhD
Health Information Technology Research Laboratory
School of IT
The University of Sydney

Abstract

Ten tables of the relational schema from the Cerner Millenium product are analysed for weaknesses in design and implementation. The design is further assessed by considering the full content of 3 data tables and screenshots of the contents of the 10 tables. The potential consequences of the design weaknesses are described and discussed in terms of their risk for process productivity and institutional outcomes, and maintaining the protection of patient records from unauthorised interference.

Introduction

The database schemata and partial data tables for a number of the entities in the ERDs discussed in Part 4.1 have been provided by associates, plus some other schemata for relational tables without the corresponding ERDs are available. This information allows us to more deeply assess a number of aspects of the CIS design and implementation. Firstly, it allows us to address putative weaknesses in design as to whether they have been remedied in implementation. Secondly, they enable us to assess any new weaknesses introduced in the implementation process itself.

Schema Descriptions

The schema descriptions have all been extracted using the Cerner tool VisualExplorer. They are therefore subject to the vagaries that the tool might produce in differences between the underlying schemata declarations and the presented descriptions in the tool. We believe that the glossary characteristics of the tool date to an earlier stage of development so that there will be differences between attribute names and possibly definitions compared to the most current software installations. This will create problems for configuration and maintenance staff as they attempt to come to grips with the system whose documentation is out of phase with the implementation.

The VisualExplorer tool presents a number of different types of information, namely:
the name of the table,
for each attribute:

- the name of the attribute,
- the data type of the attribute (e.g. F8, DQ8, I2, Varchar2, Char, Long, etc),
- a Definition explaining something about the attribute.

The display also shows red and yellow key icons that are explained in the Cerner manual as "A yellow key displayed next to a field indicates an index for the field and a red key indicates a unique index" (see Fig 1). We interpreted the red key icon to mean that it indicates the PK in the schema. The yellow key is more enigmatic. At times we believed it was used to indicate FKs in the schema but at other times it truly seems to be acting to indicate just an index. No matter its underlying meaning we are left without any reliable indicator for FKs in the schemata and at times concerns about the correct denotation of the PK as indicated below. We have been unable to identify in the Cerner documentation any information about PKs and FKs. There is evidence from time to time and from naming conventions for attribute names ending in the string "_id" that attributes with the yellow key icon are FKs so we have adopted the convention to refer to these attributes as "index/FK". We will have to wait for further information to establish if they carry truly both characteristics or not, but it is a weakness not to have a clear discernment of FKs to enable their proper consideration by the development programmers. Named attributes without a red or yellow key icon ending in the string "*-id*' have been denoted as "putatative FKs" on the basis they have all the appearance of a FK but their actual status on this functionality is unknown.

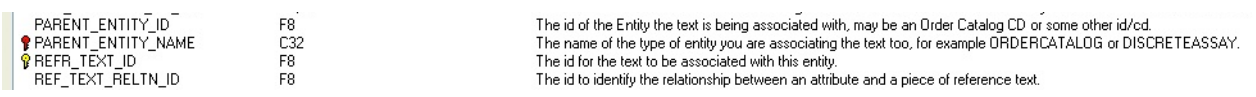


Figure 1. An example of the red and yellow key icons used to indicate a “unique index” and “index” as presented in the VisualExplorer.

We understand that the data table can have attribute names different to the displayed attribute name list in the schema glossary as it was compiled at sometime in the past and is not dynamically linked to the executable schema. This means that there is a programming interface that can change attribute entries in the real schema while the schema glossary tool shows an older version of the schema. This causes one to wonder how programming staff can operate with confidence that the design they are seeing in front of them is the actual design that is operating in the executable system.

A complete list of all tables studied in this work are listed in Appendix 2 along with their PK, index/FKs and putative FKs.

Primary Keys (PK)

The study of PKs in the schemata shows signs of engineering weaknesses that cause some concern.

1. PK Uniqueness

It is normal for PKs to be maintained so that they are unique for every record in the database. This can be done using functions built in to the database management system or it can programmed into by the supplier software. Appendix 1 presents an analysis of the table PHA_PRODUCT which illustrates this data table does not preserve the uniqueness of PKs.

DataType F8

The PKs are predominantly declared as of datatype F8, which can be interpreted as a floating point number of 8 memory bytes, Unfortunately, despite having found a number of interpretations of this data format we haven’t been able to unravel a reliable meaning of this declaration. However, the critical factor about PKs is that one wants to compare them for being exactly equal, and for this reason they would normally always be declared to be of integer datatype. In classical computing, floating point numbers are not compared for equality but rather for a comparison value, that is, greater than or less than. The critical issue is that once numbers get to be large the process of converting an integer number into a floating point number is susceptible to errors in the conversion algorithms or slip ups in the data entry or upgrades in software that change the processing methods of floating point numbers. It is true that a PK could be created using floating point numbers and work correctly over a period time but it is a highly risky strategy and pursued against the strongest possible advisory not to do so.

Foreign Keys (FK)

The use of FKs is to ensure that a piece of information about a major entity is correctly preserved in its association with that entity, and that the piece of information cannot become dissociated from its primary entity and therefore be inaccessible and clog up the database. It is conventional that when data is inserted into a table where an attribute is a FK, to check that the value for the FK does exist in the primary table under its PK value before inserting into the subsidiary table. It is also conventional that when a PK record is deleted then all other references to that PK in FKs are deleted. Both these acts of processing can be performed automatically by the database management software and do not require special programming effort when the PK and FK relationships are defined in the table schemata. If the relationships are not provided in the table schemata then either the integrity is not maintained or the software engineers must design their own processing systems to protect this integrity.

There are a number situations in which FK checking might not be implemented:

i. a system that has a lot of legacy data where the FKs where not enforced in the first place.
ii. to avoid the computational cost of checking the correctness of the FKs;
iii. to save the time cost of debugging data conflicts.
iv. to avoid dealing with the problem of the sequence of updating tables in a complicated transaction that requires dependencies to be properly preserved and inserted into the database in the correct sequence.
v. time dependency when the time cost is too great for the checking in realtime and it is deferred until a later time, however in this case a process that does the verification has to be defined and made operational in the application context.

On some occasions we can more definitely identify FKs because we either have a definite declaration of a PK in another table or we have a Definition that informs us the attribute is a FK or a PK from another table. Unfortunately we do not always have these clear indicators.

Attribute Usage

Naming of attributes is important in that they carry the base load of the semantics of the application. Failure to use attribute names that are appropriate for the data they carry makes it more difficult for programming staff to create the correct internal processing the system needs and to present the correct information values in the presentation layer. This leads to incorrect and confusing information being presented to the clinical user and potentially showing the wrong information associated with a given client. Another side effect is more subtle and less easy to detect, that is, derivative information that is computed from raw data values where the error in the values is masked by indirectness that is only discovered after considerable time has elapsed and a particularly severe case comes to the attention of watchful and astute staff.

Overloading attributes with multiple meanings is a particularly significant danger in this situation as the programming staff need to understand the exact meaning context they are programming to get the correct values manipulated in a given context. This is likely to reduce accuracy and reliability as the system gets older and newer staff join the programming team without the historical knowledge of the original designers.

Weaknesses that have been identified include: attribute overloading in PERSON, changing attribute names and datatypes between the relational schema and data tables in ITEM_MASTER and MEDICATION_DEFINITION, ambiguous definItions of attributes in DCP_FORMS_ACTIVITY, DCP_FORMS_REF, and missing definitions of attributes in MEDICATION_DEFINITION.

In the case of DCP_FORMS_REF there are fields for the height and width of print configurations which are mostly redundant nowadays with the software for processing web serviced data.

Data Values

Stored data values in the database tables give some indications of the types of processing that is done to the data. It can identify the consistency of data across PK-FK relations and show if invalid or questionable data is allowed into the database. This in turn can indicate to some extent whether entity and referential integrity is being enforced across the data tables.

Weaknesses identified by studying data values have been: duplication of attributes between DCP_FORMS_REF and DCP_FORMS_ACTIVITY, variable and dubious orthographies in MEDICATION_DEFINITION, sequential and contiguous data values for PKs in PRSNL, REF_TEXT, RET_TEXT_RELTN.

A miscellaneous set of weaknesses are: unusual records loaded into ITEM_MASTER, and selective processing of records based on date variations in ITEM_MASTER.

The set of data tables for the pharmacy schema were provided, namely ITEM_MASTER, MED_IDENTIFIER and MEDICATION_DEFINITION. The aim of analysing these tables was to establish if data values that violated referential integrity could be found. *Item_id* is declared to be the PK of ITEM_MASTER and in MEDICATION_DEFINITION defined as “Item_id inherited from item_master”. Also it is used as an index/FK in MED_IDENTIFIER and thus links each of these tables together. The following results are provided:

A = *Item_id* data values in ITEM_MASTER but not in MED_IDENTIFIER
A = {"590623", "590634", "590645", "590656", "1501266", "1495298"}

B = *Item_id* data values in ITEM_MASTER but not in MEDICATION_DEFINITION
B = {"590623", "590634", "590645", "590656"}

C = *Item_id* data values in MEDICATION_DEFINITION but not in MED_IDENTIFIER
C = {"1495298", "1501266"} = A-B

D = *Item_id* in MEDICATION_DEFINITION but not in ITEM_MASTER
D = {}

A and B = {"590623", "590634", "590645", "590656"}
These four records have an *update_cnt* of zero and were created in 2003 whilst the remainder of the table was created in 2009 and 2010.

These results beg the question as to why there should be data values in the ITEM_MASTER table and not related tables. It might satisfy some to say that the 4 old records in the ITEM_MASTER table are left over remnants from an earlier installation that have no effect on the operational system but such an argument still does not excuse the fact that PKs in one table are not validated in another table.

Likewise the 2 contemporary PK references in ITEM_MASTER and MEDICATION_DEFINITION that have no matching records in MED_IDENTIFER whilst not an integrity violation invite the question as to why there are only two records out of about 3700 that are missing this data.

Issues from Specific Tables

Different Datatypes for the same PK

Comparison of MEDICATION_DEFINITION and PHA_PRODCT Tables

In the MEDICATION_DEFINITION table an attribute *item_id* is declared as the PK with the definition “Item id inherited from item_master” (see Appendix 1, fig 6). The attribute *cki* is declared as an index/FK with a datatype of VC255 and has the definition “Cerner Knowledge Index field for MULTUM MMDC numbers. Syntax is “MUL.FRMLTN!<mmdc>” ”. It is a particularly concerning issue for a FK or PK to be declared as a character field as just as with floating point numbers errors creep into the processing system quite readily. The fact that the fields are VARCHAR means that they can be variable length strings and hence a key value might be inserted into the system in one place with the correct set of digits but in another place the the same digits might have a leading blank or zero and in another place might have a trailing blank or zero. In each case none will be correctly tested to be equivalent to the other.

In the table PHA_PRODUCT the attribute *item_id* has no PK nor index/FK declarations even though it carries the definition “Primary Key - Item id for the product”, which also seems to indicate it is the same attribute in both tables. There is no attribute in this table called *cki* but there is an attribute *gfc_cki* with the datatype VC100 and the definition “Cerner Knowledge Index for generic information as defined by MULTUM. CKI is MUL.FRMLTN!<main_multum_drug_code>” which appears to be *intensionally* the same as the *cki* attribute in the MEDICATION_DEFINITION table. The design weakness in these attributes are that they are different datatype sizes, 100 vs. 255 characters of variable length, which leaves open vagaries about how they might be compared correctly, and variably declared as non-keys and keys thus cutting off the internal checking mechanisms for ensuring the same data is correct and consistent and not corrupted or missing or formatted in a different manner across different tables.

Confused Primary Key Selection - REF_TEXT

The table REF_TEXT is shown, in the Cerner tool VisualExplorer, with a dual attribute PK, that is a primary key consisting of two attributes, namely *refr_text_id* and *ref_text_name*, with datatypes F8 and VC100 respectively (see Appendix 1, fig 22). This structure is also known as a concatenated key.

This formation is striking because the second attribute is a variable character datatype. This is wasteful for computation in the construction of an index for the PK and in the comparison of PKs in searching for records. Furthermore, as discussed above variable character fields are prone to variations of representation of the same values and so are not recommended for use as PKs.

The interpretation of this design has further issues to be considered. The name of the PK usually mimics the name of the table and so we would expect it to not have the second “r” and be *ref_text_id* or the name of the table changed to REFR_TEXT. Further confusion about the intended PK for this table is created by the definition of each of the participating attributes, being for *refr_text_id*, “the key to the table identifying the reference text”, and for *ref_text_name* being “Ref text name”, suggesting that only the first attribute is the PK.

The ERD sheds little light onto this problem as it shows no PK for the entity REF_TEXT and *refr_text_id* is shown as an ordinary attribute that is neither a *PK* nor *FK.*

Searching in the data table itself to use the actual data values to aid in interpretation also adds to this mystification as for many records *ref_text_name is empty* suggesting that although the field is declared as part of the PK, the values are not enforced, that is, there is no application of the PK Integrity rule.

Furthermore, in the REF_TEXT schema there are two other putative FKs, recognised by their name structure and definition, namely: *text_entity_id*, “The id where the text string is being stored...”, and *update_id*, “The *person_id* of the person from the personnel table [prsnl]...”, suggesting that FK referential integrity checking has not been applied to this table. Hence there will not be any automatic checking that a *person_id* inserted into this table will be a valid value registered in the PRSNL table, and likewise that *text_entity_id* value is not validated.

Undeclared PK - REF_TEXT_RELTN Table

The REF_TEXT_RELTN table is shown in the VisualExplorer tool (Fig. 2) to have a single PK of *parent_entity_name* with datatype C32, a single index/FK *refr_text_id* with datatype F8 and two attributes of interest *parent_entity_id* and *ref_text_reln_id*, that are putatively PKs judged by their name structure, that is they end in the string "_id" and their datatypes of F8, but they do not show the PK nor FK index icons.

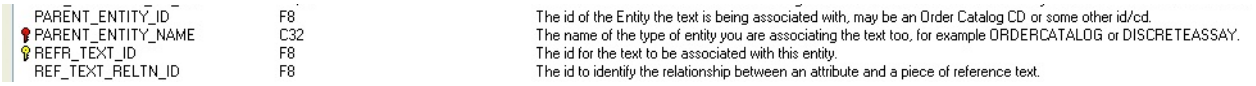


Figure 2. View using Cerner's Visual Explorer tool of the table REF_TEXT_RELN showing only the salient attributes under discussion.

An analysis of these definitions leads to a confusing picture. Normally, as we have seen in other parts of the system, the name of a PK of a table is constructed using the name of the table and adding the string "_id", in which case we would expect the attribute shown in the table *ref_text_reln_id* to be declared as the *PK,* which it plainly it is not. As well, its definition *"the id to identify the relationship between an attribute and a piece of reference text."* is indicative that it is consistent with being the appropriate PK for this table.

The attribute labeled as the PK is *parent_entity_name* which is a fixed length character datatype which as we have said before is undesirable as a PK due to the added computational cost of manipulating character strings to make comparisons between PK values. Further, the definition of this attribute is given as "*The name of the type of entity you are associating the text too(*sic*)..."*.

The attribute *parent_entity_id* seems to be the key value for the attribute *parent_entity_name* and so we would expect it to be defined as a FK, and it would be in the relationship as defined by this table with the attribute *refr_text_id* which is seemingly correctly defined as an index/FK.

In summary based on the naming conventions most commonly used in this schema and the definitions of the attributes we would expect the PK and index/FK declarations shown in Table 1. The actual assignments are also shown in Table 1.

1. Attribute	1. Actual Declarations	1. Predicted Declarations
parent_entity_id	none	index/FK
parent_entity_name	PK	none
refr_text_id	index/FK	index/FK
ref_text_reltn_id	none	PK

Table 1. Actual and expected index key declarations in the table REF_TEXT_RELTN.

The question remaining to be answered is - Does the allocation of keys shown in Table 1 make any difference? The answer is NO if you do not want the DBMS to manage the integrity of the identifiers that link data from one table to another. The answer is YES if you want strong checking that data is consistent and not erroneous before it is inserted into a table for storage, and you want to exploit the validated integrity of the data to construct other validation suites such as clinical decision support which is highly reliable both from a clinical data point of view and from a software engineering and code management point of view.

Compilation of Weaknesses

A complete study of each table and our observations about their weaknesses are presented in Appendix 1. A compilation of the principal weaknesses identified in the available schema and data tables has been collated in Table 2. The results show that there is no schema or table without a weakness of some kind. The most persistent weaknesses across the data set are non-integer PKs and non-declared FKs. Issues that would represent the highest risk for user sites would be: doubtful PKs, misnamed PKs, and overloading attributes as these potentially can interfere with more fundamental aspects of data management and hence veracity. Other weaknesses if they create disturbance to data will tend to be isolated to single items such as a single patient or pharmacy record.

Weakness	Table Implicated	Attribute
Data duplication across tables	**DCP_FORMS_REF** and **DCP_FORMS_ACTIVITY**	*definition* and *description*
Doubtful PK	**REF_TEXT_RELTN**	*parent_entity_name*
Doubtful concatenated PK	**PRSNL**	*person_id+username*
Doubtful PK	**PRSNL**	*username*
Doubtful PK	**REF_TEXT**	*ref_text_name*
Floating Point PK	**All Tables**	all PKs
Inconsistent naming between glossary and data tables	**ACCESS_CONTROL_POLICY**	*a_access_control_decisio n_disp*
Inconsistent naming between glossary and data tables	**MED_IDENTIFIER**	*m_flex_type_disp*, *m_med_identifier_type_disp*, and *m_pharmacy_type_disp*
Inconsistent naming between glossary and data tables	**ITEM_MASTER**	*i_cost_center_disp, i_storage_requirement_disp* and *i_sub_account_disp*
Inconsistent naming between glossary and data tables	**MEDICATION_DEFINITION**	*m_alternate_dispense_cat egory_disp, m_dispense_category_disp, m_dispense_quantity_unit_di sp, m_formulary_status_disp, m_form_disp, m_legal_status_disp, m_order_alert1_disp_morder _alert2_disp, m_strength_unit_disp,* and *m_volume_unit_disp*
Misnamed PK	**PRSNL**	*person_id*
Misnamed PK	**REF_TEXT**	*refr_text_id*
Missing or Poor Definitions	**MED_IDENTIFIER**	all attributes
Missing or Poor Definitions	**DCP_FORMS_REF**	*description, definition*
Overloading attributes	**PRSNL**	*prsnl_type_cd*
Overloading attributes	**PERSON**	*person_type_cd*
Potential Problematic dates	**MEDICATION_DEFINITION**	*updt_dt_tm*
Potential Problematic dates	**ITEM_MASTER**	*updt_dt_tm*
Questionable Data Values	**MED_IDENTIFIER**	*value_key*
Questionable Data Values	**ITEM_MASTER**	*item_id, updt_id, updt_task*
Reused PK	**MEDICATION_DEFINITION**	*item_id*

Weakness	Table Implicated	Attribute
Undeclared FKs	**Most tables (10 out of 12)**	*51 cases*
Unidentified PK	**REF_TEXT_RELTN**	*ref_text_reltn_id*
Variation in datatypes across tables	**MEDICATION_DEFINTION** vs **PHA_PRODUCT**	*cki (VC255)* vs *gcr_cki* and *gfc_cki (CV100)*

Table 2. The principal weaknesses found in the analysis of 12 tables of the Cerner Millenium software.

Conclusions

Weaknesses in CIS implementation suggest a lack of attention to detail, particularly specific testing to validate schema designs and index/FK and PK selection. If the yellow key icon is truly an FK indicator then the apparent use of FKs for creating indices rather than their purpose of maintaining referential integrity is a misapplication of this functionality.

Identified weaknesses could reasonably be expected to produce faulty processing of user data manifesting as problems such as, missing parts of patient records, missing information about pharmacy products. Particularly, these occurrences will appear occasionally without any apparent systematic behaviour as they will not be triggered by each and every patient record but rather only where a particular record uses a combination of information that requires the correct data relationships. Hence any one clinical user will observe a fault on occasions so far apart in time they will not connect a set of failings as being related to an underlying systematic weakness.

Appendix 1. Analysis of Each Table for Design and Implementation Weaknesses

1. ACCESS_CONTROL_POLICY

This table is also represented in an ERD in which it has no common attributes apart from the auditing attributes inserted in all schema. A search of all ERDs for the Security Application (33 entities) shows no entity with any reasonable similarity to the implemented schema. Minimally this is an example of the documentation being entirely out of date with the implementation, but a more likely explanation is that the ERD diagram has been incorrectly labelled by staff who were unfamiliar with the larger design of the system.

There are two putative keys access_control_type_entity_id and data_source_entity_id which based on their descriptions appear to be references in other schema and would be expected to be index/FKs under normal circumstances. This would ensure no cross references could be created for Access Control Types or Data Sources that did not exist, and that such cross references would be deleted if the original referents were removed.

The data table shows an attribute a_access_control_decision_disp (Fig 2) which is not shown on the schema glossary (Fig. 1) but appears to be a renaming of the attribute access_control_decision_cd as its Definition in the schema glossary states that its values can be “permit” or “deny”, and these are the values found in the data table.

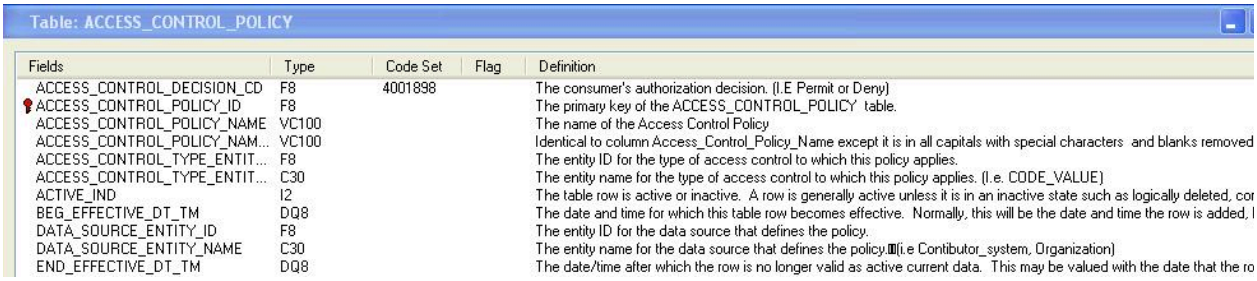
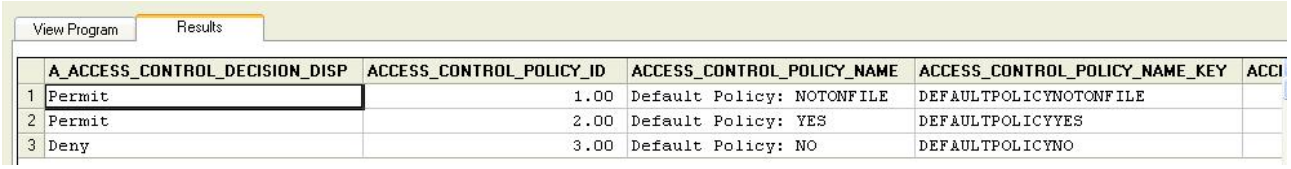


Table: ACCESS_CONTROL_POLICY

Fields	Type	Code Set	Flag	Definition
ACCESS_CONTROL_DECISION_CD	F8	4001898		The consumer's authorization decision. (I.E Permit or Deny)
ACCESS_CONTROL_POLICY_ID	F8			The primary key of the ACCESS_CONTROL_POLICY table.
ACCESS_CONTROL_POLICY_NAME	VC100			The name of the Access Control Policy
ACCESS_CONTROL_POLICY_NAM...	VC100			Identical to column Access_Control_Policy_Name except it is in all capitals with special characters and blanks removed
ACCESS_CONTROL_TYPE_ENTIT...	F8			The entity ID for the type of access control to which this policy applies.
ACCESS_CONTROL_TYPE_ENTIT...	C30			The entity name for the type of access control to which this policy applies. (I.e. CODE_VALUE)
ACTIVE_IND	I2			The table row is active or inactive. A row is generally active unless it is in an inactive state such as logically deleted, cor
BEG_EFFECTIVE_DT_TM	DQ8			The date and time for which this table row becomes effective. Normally, this will be the date and time the row is added,
DATA_SOURCE_ENTITY_ID	F8			The entity ID for the data source that defines the policy.
DATA_SOURCE_ENTITY_NAME	C30			The entity name for the data source that defines the policy. (i.e Contibutor_system, Organization)
END_EFFECTIVE_DT_TM	DQ8			The date/time after which the row is no longer valid as active current data. This may be valued with the date that the ro

Figure 1. Relational schema of the ACCESS_CONTROL_POLICY table as seen through the Cerner VisualExplorer tool.



View Program | Results

	A_ACCESS_CONTROL_DECISION_DISP	ACCESS_CONTROL_POLICY_ID	ACCESS_CONTROL_POLICY_NAME	ACCESS_CONTROL_POLICY_NAME_KEY	ACCI
1	Permit	1.00	Default Policy: NOTONFILE	DEFAULTPOLICYNOTONFILE	
2	Permit	2.00	Default Policy: YES	DEFAULTPOLICYYES	
3	Deny	3.00	Default Policy: NO	DEFAULTPOLICYNO	

Figure 2. Data table for the ACCESS_CONTROL_POLICY table. Note the name of the first attribute is different to that shown in the schema, although the values are consistent with the definition in the schema.

2. DCP_FORMS_ACTIVITY

This schema expresses the activity around using a form in the relationships between patient (*person_id*), encounter (*encntr_id*), task (*task_id*) and forms (*dcp_forms_ref_id*). It has the putative FK *lock_prsnl_id* from the PRSNL schema which is presumably the clinical staff member accessing the patient record.

An interesting declared index/FK is the audit field *updt_tm_tm* which is by this definition set up as a match to all other relational schema tables that have this auditing information (see Fig. 3). This suggests that the attribute is being used as an index rather than a FK unless it is matched with a very particular schema table. We know from the reports of clinical staff at live sites that searching recent records only allows for viewing the last 1000 updates performed on patient records. It may well be that this field is indexed to provide this service and time limited by logic coding in the software. However we also know that in the preparation of multiple pathology orders that each order for the same sample must have exactly the same time stamp otherwise it is rejected. An embedded computational strategy that makes this attribute a FK that requires matching would explain the user interface behaviour. If this were the case then it would be considered a weakness in implementation as logically two orders on the same sample cannot be made at a physically synchronous moment in time. A computational strategy which uses the same sample identifier would be more appropriate.

Another attribute is of interest, *description (VC255)*, as it is described as “the display that will show up in the form browser” without indicating the role of this text, that is, is it about the patient or is it about the form.

The data table shows the PK values as a 11 digit floating point number with 2 decimal points. This ostensibly confirms that the PKs are floating point numbers.

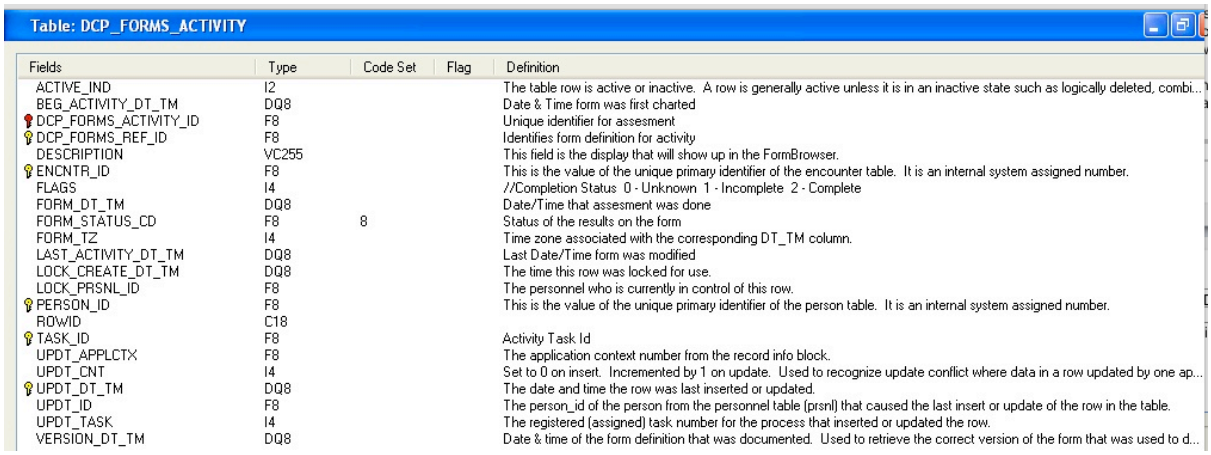
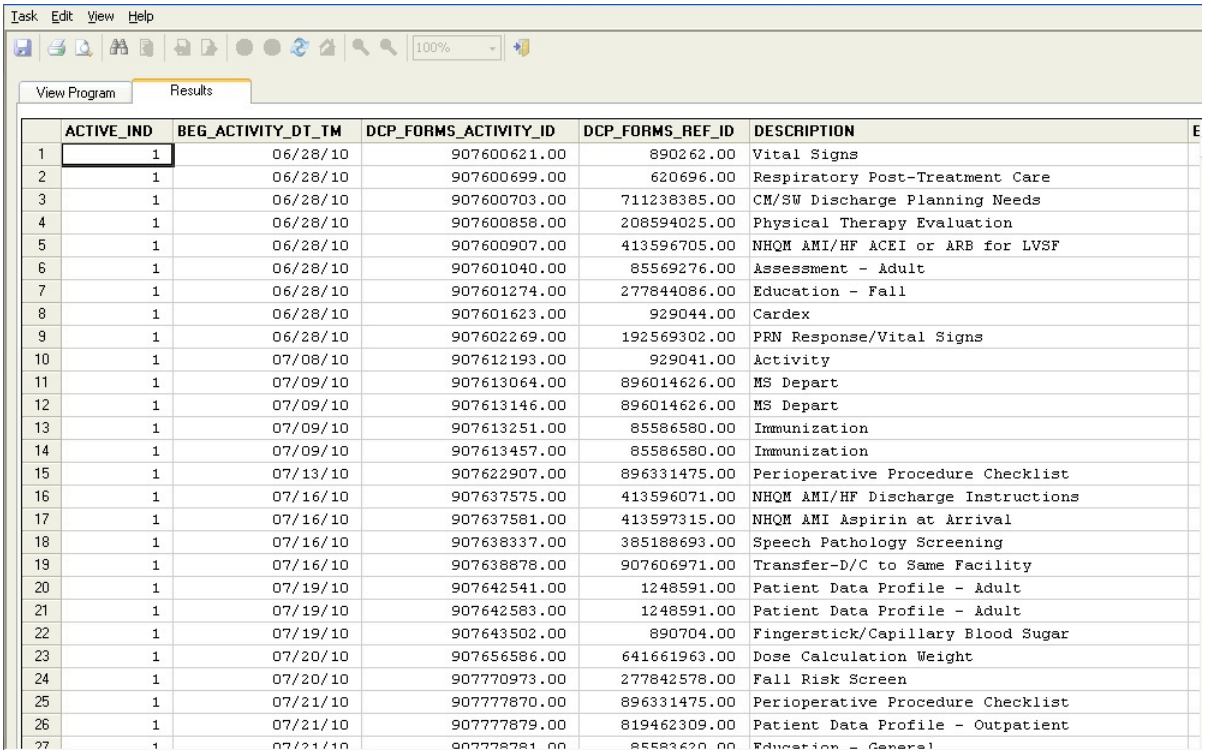


Table: DCP_FORMS_ACTIVITY

Fields	Type	Code Set	Flag	Definition
ACTIVE_IND	I2			The table row is active or inactive. A row is generally active unless it is in an inactive state such as logically deleted, combi...
BEG_ACTIVITY_DT_TM	DQ8			Date & Time form was first charted
DCP_FORMS_ACTIVITY_ID	F8			Unique identifier for assesment
DCP_FORMS_REF_ID	F8			Identifies form definition for activity
DESCRIPTION	VC255			This field is the display that will show up in the FormBrowser.
ENCNTR_ID	F8			This is the value of the unique primary identifier of the encounter table. It is an internal system assigned number.
FLAGS	I4			//Completion Status 0 - Unknown 1 - Incomplete 2 - Complete
FORM_DT_TM	DQ8			Date/Time that assesment was done
FORM_STATUS_CD	F8	8		Status of the results on the form
FORM_TZ	I4			Time zone associated with the corresponding DT_TM column.
LAST_ACTIVITY_DT_TM	DQ8			Last Date/Time form was modified
LOCK_CREATE_DT_TM	DQ8			The time this row was locked for use.
LOCK_PRSNL_ID	F8			The personnel who is currently in control of this row.
PERSON_ID	F8			This is the value of the unique primary identifier of the person table. It is an internal system assigned number.
ROWID	C18			
TASK_ID	F8			Activity Task Id
UPDT_APPLCTX	F8			The application context number from the record info block.
UPDT_CNT	I4			Set to 0 on insert. Incremented by 1 on update. Used to recognize update conflict where data in a row updated by one ap...
UPDT_DT_TM	DQ8			The date and time the row was last inserted or updated.
UPDT_ID	F8			The person_id of the person from the personnel table (prsnl) that caused the last insert or update of the row in the table.
UPDT_TASK	I4			The registered (assigned) task number for the process that inserted or updated the row.
VERSION_DT_TM	DQ8			Date & time of the form definition that was documented. Used to retrieve the correct version of the form that was used to d...

Figure 3. DCP_FORMS_ACTIVITY glossary schema. The attribute *updt_dt_tm* is an unusual for an index/FK role.



Task Edit View Help

View Program | Results

	ACTIVE_IND	BEG_ACTIVITY_DT_TM	DCP_FORMS_ACTIVITY_ID	DCP_FORMS_REF_ID	DESCRIPTION
1	1	06/28/10	907600621.00	890262.00	Vital Signs
2	1	06/28/10	907600699.00	620696.00	Respiratory Post-Treatment Care
3	1	06/28/10	907600703.00	711238385.00	CM/SW Discharge Planning Needs
4	1	06/28/10	907600858.00	208594025.00	Physical Therapy Evaluation
5	1	06/28/10	907600907.00	413596705.00	NHQM AMI/HF ACEI or ARB for LVSF
6	1	06/28/10	907601040.00	85569276.00	Assessment - Adult
7	1	06/28/10	907601274.00	277844086.00	Education - Fall
8	1	06/28/10	907601623.00	929044.00	Cardex
9	1	06/28/10	907602269.00	192569302.00	PRN Response/Vital Signs
10	1	07/08/10	907612193.00	929041.00	Activity
11	1	07/09/10	907613064.00	896014626.00	MS Depart
12	1	07/09/10	907613146.00	896014626.00	MS Depart
13	1	07/09/10	907613251.00	85586580.00	Immunization
14	1	07/09/10	907613457.00	85586580.00	Immunization
15	1	07/13/10	907622907.00	896331475.00	Perioperative Procedure Checklist
16	1	07/16/10	907637575.00	413596071.00	NHQM AMI/HF Discharge Instructions
17	1	07/16/10	907637581.00	413597315.00	NHQM AMI Aspirin at Arrival
18	1	07/16/10	907638337.00	385188693.00	Speech Pathology Screening
19	1	07/16/10	907638878.00	907606971.00	Transfer-D/C to Same Facility
20	1	07/19/10	907642541.00	1248591.00	Patient Data Profile - Adult
21	1	07/19/10	907642583.00	1248591.00	Patient Data Profile - Adult
22	1	07/19/10	907643502.00	890704.00	Fingerstick/Capillary Blood Sugar
23	1	07/20/10	907656586.00	641661963.00	Dose Calculation Weight
24	1	07/20/10	907770973.00	277842578.00	Fall Risk Screen
25	1	07/21/10	907777870.00	896331475.00	Perioperative Procedure Checklist
26	1	07/21/10	907777879.00	819462309.00	Patient Data Profile - Outpatient

Figure 4. DCP_FORMS_ACTIVITY data table. The Description attribute has values match contents in the DCP_FORMS_REF table, e.g. “Patient Data Profile - Adult”.

3. DCP_FORMS_REF

The schema glossary shows a PK *dcp_forms_instance_id* and one index/FK *dcp_forms_ref_id*. It appears to capture information about a designed form and its version (see fig. 5). It is notable that there are two attributes that are difficult to separate in their meaning intention:

- *description* (VC200) described as “textual description of the form”, and
- *definition* (VC200) described as “textual definition of the form”.

Two other attributes are also notable in that they define the physical size of a form, *height* and *width*. In the modern interface the sizing of presentation content is a more flexible matter as web pages can be manipulated extensively by the user making it of very limited opportunity to predefine document sizes. This information might explain the complaints from clinical staff that many pages are printed from reports that

are mostly blank and contain limited content of value. Another complaint, that these attributes may have a bearing on, is the very small size of fixed windows that carry critical information that then require horizontal and vertical scrolling to read the information, thus delaying clinicians in their patient care.

The values in the data table column labelled *definition* appear to be the same as in the column *description* of the DCP_FORMS_ACTIVITY table adding further to the previously mentioned terminological ambiguity (see fig. 6) Such a duplication of data are normally unnecessary and without a very secure method of loading data is prone to producing inconsistencies in the purportedly same data kept in different places.

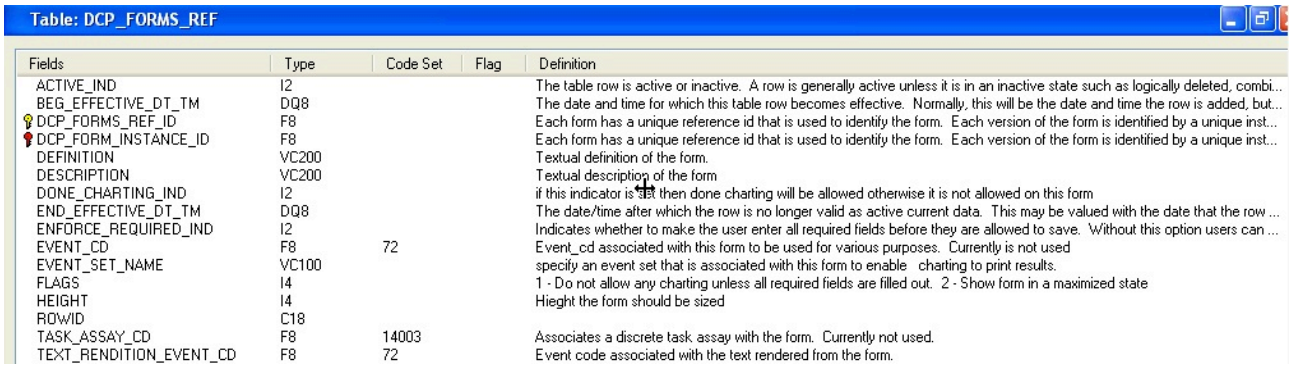
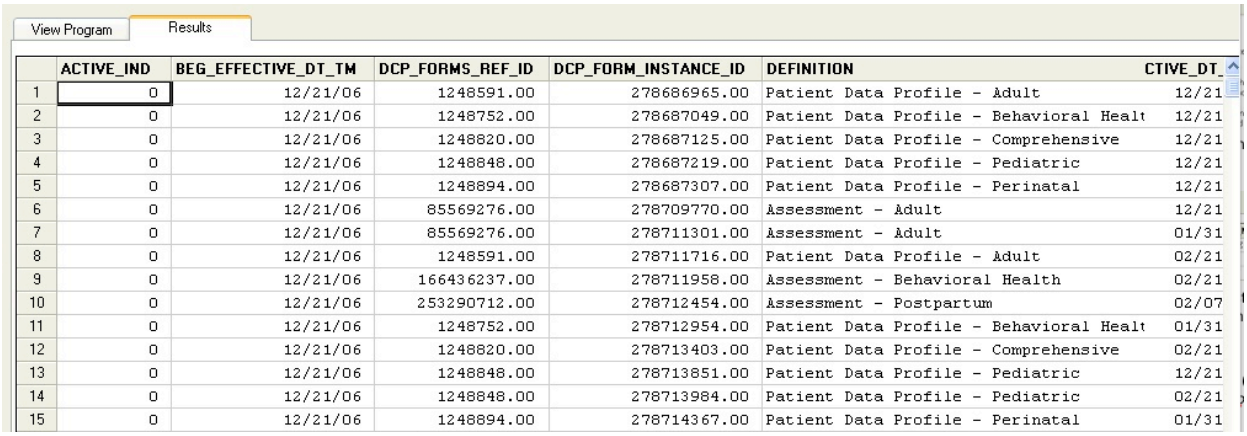


Table: DCP_FORMS_REF

Fields	Type	Code Set	Flag	Definition
ACTIVE_IND	I2			The table row is active or inactive. A row is generally active unless it is in an inactive state such as logically deleted, combi...
BEG_EFFECTIVE_DT_TM	DQ8			The date and time for which this table row becomes effective. Normally, this will be the date and time the row is added, but...
DCP_FORMS_REF_ID	F8			Each form has a unique reference id that is used to identify the form. Each version of the form is identified by a unique inst...
DCP_FORM_INSTANCE_ID	F8			Each form has a unique reference id that is used to identify the form. Each version of the form is identified by a unique inst...
DEFINITION	VC200			Textual definition of the form.
DESCRIPTION	VC200			Textual description of the form
DONE_CHARTING_IND	I2			if this indicator is set then done charting will be allowed otherwise it is not allowed on this form
END_EFFECTIVE_DT_TM	DQ8			The date/time after which the row is no longer valid as active current data. This may be valued with the date that the row ...
ENFORCE_REQUIRED_IND	I2			Indicates whether to make the user enter all required fields before they are allowed to save. Without this option users can ...
EVENT_CD	F8	72		Event_cd associated with this form to be used for various purposes. Currently is not used
EVENT_SET_NAME	VC100			specify an event set that is associated with this form to enable charting to print results.
FLAGS	I4			1 - Do not allow any charting unless all required fields are filled out. 2 - Show form in a maximized state
HEIGHT	I4			Hieght the form should be sized
ROWID	C18			
TASK_ASSAY_CD	F8	14003		Associates a discrete task assay with the form. Currently not used.
TEXT_RENDITION_EVENT_CD	F8	72		Event code associated with the text rendered from the form.

Figure 5. DCP_FORMS_REF. The PK is incorrectly named as *dcp_form_instance_id* and what should be the correct PK name, *dcp_forms_ref_id* is shown as an index/FK.



View Program | Results

	ACTIVE_IND	BEG_EFFECTIVE_DT_TM	DCP_FORMS_REF_ID	DCP_FORM_INSTANCE_ID	DEFINITION	CTIVE_DT_
1	0	12/21/06	1248591.00	278686965.00	Patient Data Profile - Adult	12/21
2	0	12/21/06	1248752.00	278687049.00	Patient Data Profile - Behavioral Healt	12/21
3	0	12/21/06	1248820.00	278687125.00	Patient Data Profile - Comprehensive	12/21
4	0	12/21/06	1248848.00	278687219.00	Patient Data Profile - Pediatric	12/21
5	0	12/21/06	1248894.00	278687307.00	Patient Data Profile - Perinatal	12/21
6	0	12/21/06	85569276.00	278709770.00	Assessment - Adult	12/21
7	0	12/21/06	85569276.00	278711301.00	Assessment - Adult	01/31
8	0	12/21/06	1248591.00	278711716.00	Patient Data Profile - Adult	02/21
9	0	12/21/06	166436237.00	278711958.00	Assessment - Behavioral Health	02/21
10	0	12/21/06	253290712.00	278712454.00	Assessment - Postpartum	02/07
11	0	12/21/06	1248752.00	278712954.00	Patient Data Profile - Behavioral Healt	01/31
12	0	12/21/06	1248820.00	278713403.00	Patient Data Profile - Comprehensive	02/21
13	0	12/21/06	1248848.00	278713851.00	Patient Data Profile - Pediatric	12/21
14	0	12/21/06	1248848.00	278713984.00	Patient Data Profile - Pediatric	02/21
15	0	12/21/06	1248894.00	278714367.00	Patient Data Profile - Perinatal	01/31

Figure 6. DCP_FORMS_REF data table. The attribute *dcp_form_ref_id* should be the PK by naming convention and shows duplicate values.

4. ITEM_MASTER

This is a simple schema as shown in the glossary with a PK *item_id* and no index/FKs nor putative FKs (see Figure 7). There is no explanation as to the meaning intention of the PK.

The data table shows the attribute *i_cost_center_disp* which is not on the schema design as presented in the VisualExplorer tool, although a possible equivalent attribute *cost_center_cd* is present (see figure 8). Two other attributes also appear to be renamed in a similar manner; *i_storage_requirement_disp* and *i_sub_account_disp* appear in the data table whereas *storage_requirement_cd* and *sub_account_cd* appear in the schema glossary.

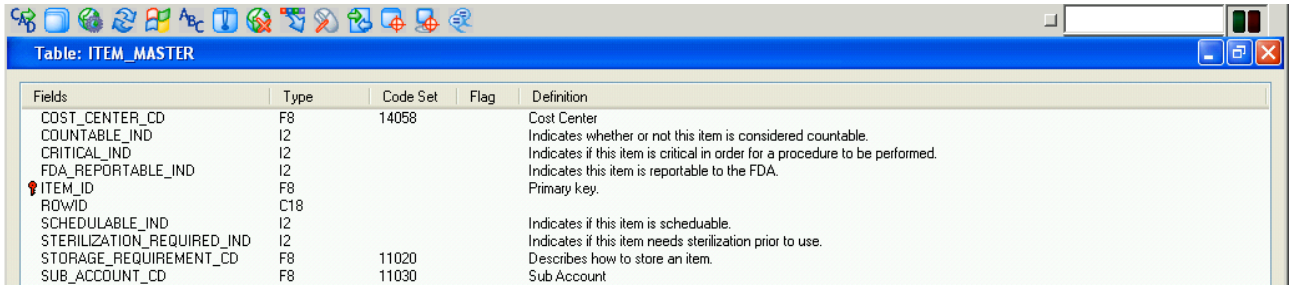


Table: ITEM_MASTER

Fields	Type	Code Set	Flag	Definition
COST_CENTER_CD	F8	14058		Cost Center
COUNTABLE_IND	I2			Indicates whether or not this item is considered countable.
CRITICAL_IND	I2			Indicates if this item is critical in order for a procedure to be performed.
FDA_REPORTABLE_IND	I2			Indicates this item is reportable to the FDA.
ITEM_ID	F8			Primary key.
ROWID	C18			
SCHEDULABLE_IND	I2			Indicates if this item is scheduable.
STERILIZATION_REQUIRED_IND	I2			Indicates if this item needs sterilization prior to use.
STORAGE_REQUIREMENT_CD	F8	11020		Describes how to store an item.
SUB_ACCOUNT_CD	F8	11030		Sub Account

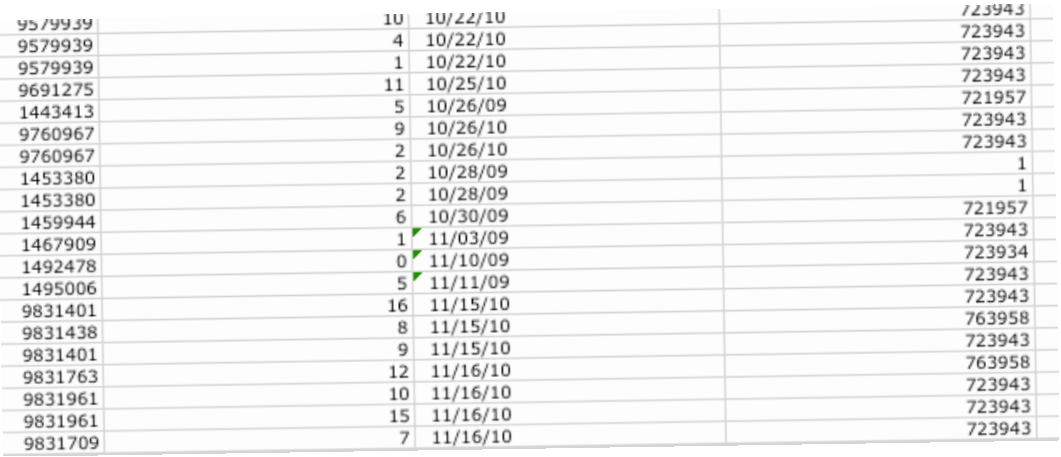
Figure 7. ITEM_MASTER Schema Glossary.

There are 3,764 entries in the data table made available to us. There are 4 entries, described above in detail, that are striking in that their *updt_dt_tm* value is 08/27/03, whereas all other records are dated in 2009 and 2010. These records also have a unique *updt_id* =2 and *updt_task* =900126.

The values of the PKs in this data table also show some unusual regular patterns, the significance of which has not been established. All PKs are odd numbers and predominantly the separation between consecutive values is 60 or some multiple of 60. Variations of 58, 62 and 64 can be found. At about two-thirds of the way through the sequence it switches to odd and even digits although there are long runs of even digits with intervals varying around a median of 44. From inspection of other columns there does not appear to be any correlation with the values of any of the other fields in the table including the audit information of dates, and applications. Although not entirely regular this appears to be systematic enough to warrant an explanation. A further inspection of the other columns show all values are either zero or blank. The only content in this table apart form the auditing information and the ROWID is the *item_id*. So this table is just a list of *item_id* values and nothing else which makes it more unclear as to the reason it is not integrated into the MEDICATION_DEFINITON table which has the same PK.

The comparison of the PK values for *item_id* with MED_IDENTIFIER table (see details below) show a closer correspondence with its PK *med_identifier_id*, where the former value appears to be the next number value after the latter value. This indicates a software process that manages the PK values outside of the database management software and is usually inadvisable. The regularities of the PK values intimates a process that has been implemented in the application software that manages the PK values across multiple tables. If adopted this is an unconventional approach that would put the economy of programming ahead of more rigorous methods of assigning PK values by the database management system.

Another concern for data veracity arises from a study of the dates of the update transactions to this table. The spreadsheet of values shows that any update between the first of the month and the 12th of the month triggers a warning message that “this cell contains a date string represented with only 2 digits for the year”. As this is true for ALL dates in the file this implies there are two different processing activities in the preparation of the ITEM_MASTER records that lead to different internal representations that in turn have unforeseen and undetected representations at least in the viewing spreadsheet software and potentially elsewhere.



9579939	10	10/22/10	723943
9579939	4	10/22/10	723943
9579939	1	10/22/10	723943
9691275	11	10/25/10	723943
1443413	5	10/26/09	721957
9760967	9	10/26/10	723943
9760967	2	10/26/10	723943
1453380	2	10/28/09	1
1453380	2	10/28/09	1
1459944	6	10/30/09	721957
1467909	1	11/03/09	723943
1492478	0	11/10/09	723934
1495006	5	11/11/09	723943
9831401	16	11/15/10	723943
9831438	8	11/15/10	763958
9831401	9	11/15/10	723943
9831763	12	11/16/10	763958
9831961	10	11/16/10	723943
9831961	15	11/16/10	723943
9831709	7	11/16/10	723943

Figure 8. Screenshot of ITEM_MASTER data table indicating dates in the *updt_tm_dt* attribute that trigger a warning (green triangles) and dates that show no such warning in a spreadsheet.

5. MEDICATION_DEFINITION

This is a schema with the PK *item_id* and therefore on a principle of best practice it could be merged into the ITEM_MASTER schema. It has not been possible to determine the special differences between the two schemata to warrant their separation.

There are 4 index/FKs, *cki* (VC255), *inv_master_id, parent_item_id, primary_manufacturer_item_id,* and 6 putative FKs, *comment1_id, comment2_id, compound_text_id, mdx_gfc_nomen_id, order_sentence_id, price_sched_id.*

The intension of this schema seems to be to hold information that defines medicines. The *cki* is an index/FK, in the datatype format VC255, but there is no indication as to which table it originates from either as a PK or common attribute. It seems to be present in the PHA_PRODUCT schema glossary as two attributes *gcr_cki* and *gfc_cki* in the data formats VC100 and do not have icons indicating any index/FK status (see Fig 9).

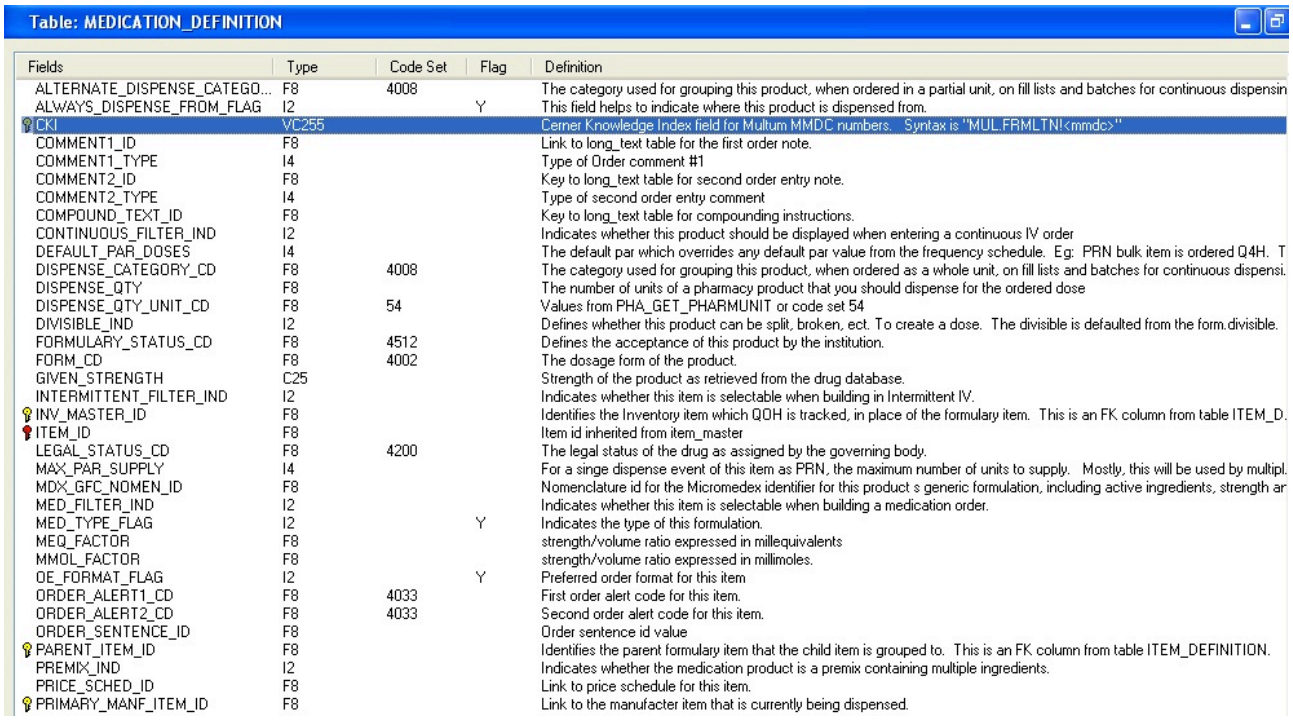


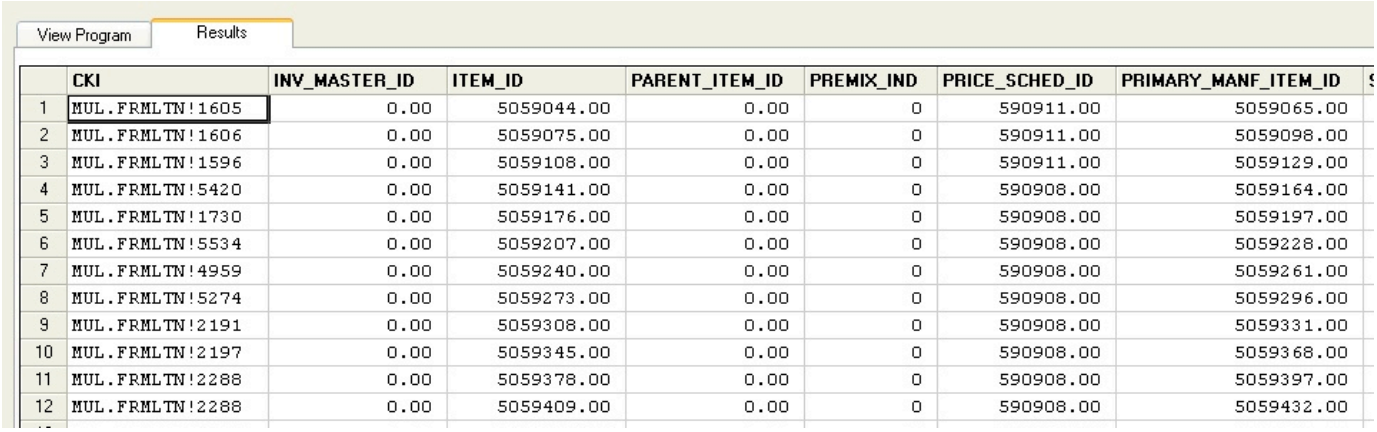
Table: MEDICATION_DEFINITION

Fields	Type	Code Set	Flag	Definition
ALTERNATE_DISPENSE_CATEGO...	F8	4008		The category used for grouping this product, when ordered in a partial unit, on fill lists and batches for continuous dispensin
ALWAYS_DISPENSE_FROM_FLAG	I2		Y	This field helps to indicate where this product is dispensed from.
CKI	VC255			Cerner Knowledge Index field for Multum MMDC numbers. Syntax is "MUL.FRMLTN!<mmdc>"
COMMENT1_ID	F8			Link to long_text table for the first order note.
COMMENT1_TYPE	I4			Type of Order comment #1
COMMENT2_ID	F8			Key to long_text table for second order entry note.
COMMENT2_TYPE	I4			Type of second order entry comment
COMPOUND_TEXT_ID	F8			Key to long_text table for compounding instructions.
CONTINUOUS_FILTER_IND	I2			Indicates whether this product should be displayed when entering a continuous IV order
DEFAULT_PAR_DOSES	I4			The default par which overrides any default par value from the frequency schedule. Eg: PRN bulk item is ordered Q4H. T
DISPENSE_CATEGORY_CD	F8	4008		The category used for grouping this product, when ordered as a whole unit, on fill lists and batches for continuous dispensi.
DISPENSE_QTY	F8			The number of units of a pharmacy product that you should dispense for the ordered dose
DISPENSE_QTY_UNIT_CD	F8	54		Values from PHA_GET_PHARMUNIT or code set 54
DIVISIBLE_IND	I2			Defines whether this product can be split, broken, ect. To create a dose. The divisible is defaulted from the form.divisible.
FORMULARY_STATUS_CD	F8	4512		Defines the acceptance of this product by the institution.
FORM_CD	F8	4002		The dosage form of the product.
GIVEN_STRENGTH	C25			Strength of the product as retrieved from the drug database.
INTERMITTENT_FILTER_IND	I2			Indicates whether this item is selectable when building in Intermittent IV.
INV_MASTER_ID	F8			Identifies the Inventory item which QOH is tracked, in place of the formulary item. This is an FK column from table ITEM_D.
ITEM_ID	F8			Item id inherited from item_master
LEGAL_STATUS_CD	F8	4200		The legal status of the drug as assigned by the governing body.
MAX_PAR_SUPPLY	I4			For a singe dispense event of this item as PRN, the maximum number of units to supply. Mostly, this will be used by multipl.
MDX_GFC_NOMEN_ID	F8			Nomenclature id for the Micromedex identifier for this product s generic formulation, including active ingredients, strength ar
MED_FILTER_IND	I2			Indicates whether this item is selectable when building a medication order.
MED_TYPE_FLAG	I2		Y	Indicates the type of this formulation.
MEQ_FACTOR	F8			strength/volume ratio expressed in millequivalents
MMOL_FACTOR	F8			strength/volume ratio expressed in millimoles.
OE_FORMAT_FLAG	I2		Y	Preferred order format for this item
ORDER_ALERT1_CD	F8	4033		First order alert code for this item.
ORDER_ALERT2_CD	F8	4033		Second order alert code for this item.
ORDER_SENTENCE_ID	F8			Order sentence id value
PARENT_ITEM_ID	F8			Identifies the parent formulary item that the child item is grouped to. This is an FK column from table ITEM_DEFINITION.
PREMIX_IND	I2			Indicates whether the medication product is a premix containing multiple ingredients.
PRICE_SCHED_ID	F8			Link to price schedule for this item.
PRIMARY_MANF_ITEM_ID	F8			Link to the manufacter item that is currently being dispensed.

Figure 9. The Schema glossary for the table MEDICATION_DEFINITION showing the same PK *item_id* as the table ITEM_MASTER (fig. 5) and the attribute *cki* as datatype VC255.

The *Definition* descriptions of putative FKs *comment2_id* and *compound_text_id* both refer to them as being keys in the LONG_TEXT table, and *comment1_id* referred to as a "link". Hence this is a clear indication that the PK functions of these attributes are recognised descriptively but possibly ignored in this implementation, potentially bringing with it all the concomitant risks.

The data table consists of 3760 records. It shows an attribute *m_alternate_dispense_category_disp* which is not shown in the schema glossary but appears to be a match for *alternate_dispense_category_cd*. Other attributes that show the same apparent change to their names are *m_dispense_category_disp, m_dispense_quantity_unit_disp, m_formulary_status_disp, m_form_disp, m_legal_status_disp, m_order_alert1_dips, m_order_alert2_disp,* plus two additional attributes *m_strength_unit_disp,* and *m_volume_unit_disp.* The schema glossary description using the old attribute names declares all these fields as F8 and there is no indication of the datatype of the renamed fields, but many of the fields are empty and where they do have content it is sometimes text material, hence it is likely these fields have had their datatypes changed (see Fig. 10.2). This is a particularly problematic task in a large software system as the maintenance programmers must find every reference to these data fields and ensure they are all changed to the new format.



View Program | Results

	CKI	INV_MASTER_ID	ITEM_ID	PARENT_ITEM_ID	PREMIX_IND	PRICE_SCHED_ID	PRIMARY_MANF_ITEM_ID
1	MUL.FRMLTN!1605	0.00	5059044.00	0.00	0	590911.00	5059065.00
2	MUL.FRMLTN!1606	0.00	5059075.00	0.00	0	590911.00	5059098.00
3	MUL.FRMLTN!1596	0.00	5059108.00	0.00	0	590911.00	5059129.00
4	MUL.FRMLTN!5420	0.00	5059141.00	0.00	0	590908.00	5059164.00
5	MUL.FRMLTN!1730	0.00	5059176.00	0.00	0	590908.00	5059197.00
6	MUL.FRMLTN!5534	0.00	5059207.00	0.00	0	590908.00	5059228.00
7	MUL.FRMLTN!4959	0.00	5059240.00	0.00	0	590908.00	5059261.00
8	MUL.FRMLTN!5274	0.00	5059273.00	0.00	0	590908.00	5059296.00
9	MUL.FRMLTN!2191	0.00	5059308.00	0.00	0	590908.00	5059331.00
10	MUL.FRMLTN!2197	0.00	5059345.00	0.00	0	590908.00	5059368.00
11	MUL.FRMLTN!2288	0.00	5059378.00	0.00	0	590908.00	5059397.00
12	MUL.FRMLTN!2288	0.00	5059409.00	0.00	0	590908.00	5059432.00

Figure 10.1 MEDICATION_DEFINITION data table from site 1. The PK item_id is shown as a floating point number.

P	Q	R	S	T	
M_FORM_DISP	GIVEN_STR	INTERMITTE	INV_MASTE	ITEM_ID	M
		0	0	0	
Tab	300 mg	0	0	1252047	
Soln-IV	2 mg/mL	0	0	1252107	
Tab-EC	333 mg	0	0	1252171	
Tab	25 mg	0	0	1252231	
Tab	50 mg	0	0	1252291	
Cap	200 mg	0	0	1252351	
Supp	120 mg	0	0	1252411	
Soln-Oral	160 mg/5 mL	0	0	1252471	
Susp-Oral	160 mg/5 mL	0	0	1252533	
Supp	325 mg	0	0	1252593	
Tab	325 mg	0	0	1252653	
Supp	650 mg	0	0	1252713	
Tab-Chew	80 mg	0	0	1252773	
Supp	80 mg	0	0	1252833	
Soln-Oral	80 mg/0.8 mL	0	0	1252893	

Figure 10.2. MEDICATION_DEFINITION data table from site 2. Column P, *m_form_dips* has text values when it is defined as a datatype of F8.

The data values as shown in the VisualExplorer tool for *item_id* have the float format with 9 digits and 2 decimal points (see Fig. 10.1).

The dates in the spreadsheet show the same warning for *updt_tm_dt* between the 1st and 12th of each month as for ITEM_MASTER (see Fig 8).

6. MED_IDENTIFIER

The MED_IDENTIFIER schema is poorly described with no meaningful Definitions provided in the glossary, almost all of them being a repeat of the attribute name themselves with no further information. The PK is shown as *med_identifier_id* with two index/FKs *item_id* and *pharmacy_type_cd*. The latter is one of the rare occasions were a "*_cd"* denoted attribute is used as a PK or index/FK (see Figure 11). We are not able to determine a schema table in which this is the PK.

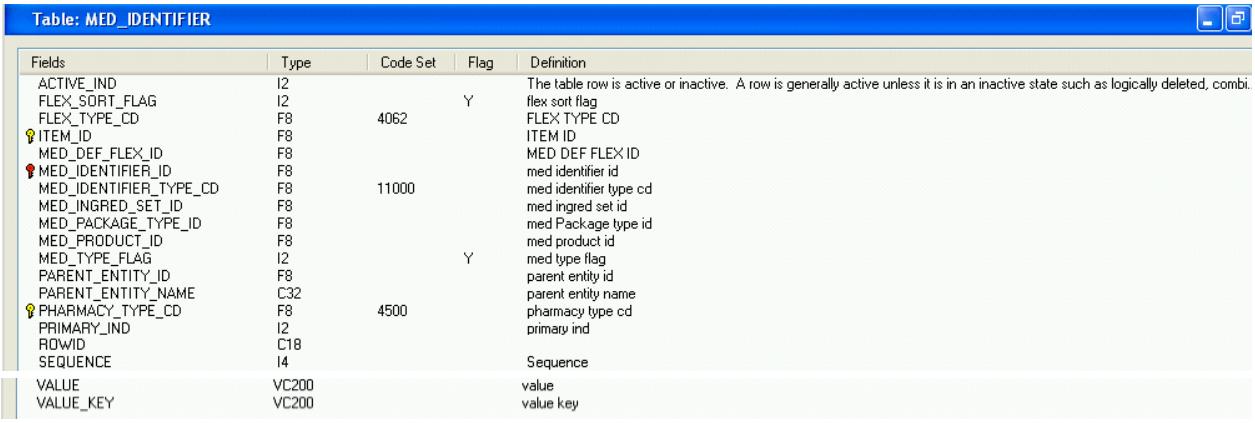
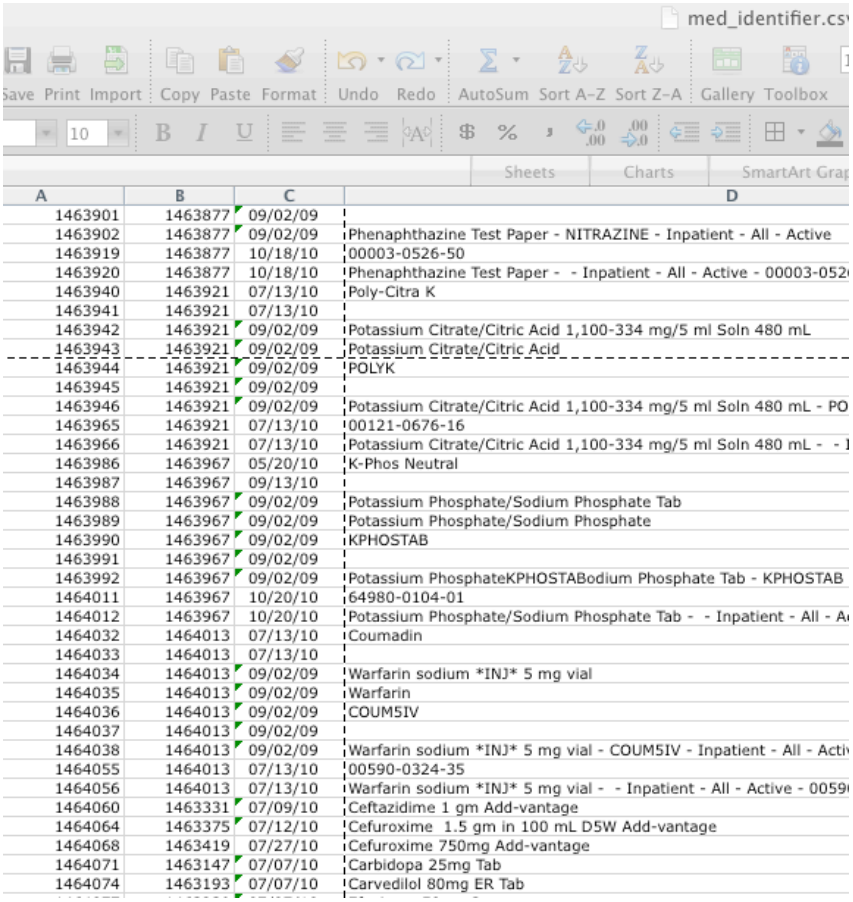


Figure 11. Parts of the schema glossary for the table MED_IDENTIFIER.

There are 5 putative FKs, *med_def_flex_id, med_ingred_set_id, med_package_type_id, med_product_id,* and *parent_entity_id.* As the Definitions give no meaningful information we can make no inferences as to what information they are meant to contain and how they might relate to other schema tables.



med_identifier.cs

A	B	C	D
1463901	1463877	09/02/09	
1463902	1463877	09/02/09	Phenaphthazine Test Paper - NITRAZINE - Inpatient - All - Active
1463919	1463877	10/18/10	00003-0526-50
1463920	1463877	10/18/10	Phenaphthazine Test Paper - - Inpatient - All - Active - 00003-052
1463940	1463921	07/13/10	Poly-Citra K
1463941	1463921	07/13/10	
1463942	1463921	09/02/09	Potassium Citrate/Citric Acid 1,100-334 mg/5 ml Soln 480 mL
1463943	1463921	09/02/09	Potassium Citrate/Citric Acid
1463944	1463921	09/02/09	POLYK
1463945	1463921	09/02/09	
1463946	1463921	09/02/09	Potassium Citrate/Citric Acid 1,100-334 mg/5 ml Soln 480 mL - PO
1463965	1463921	07/13/10	00121-0676-16
1463966	1463921	07/13/10	Potassium Citrate/Citric Acid 1,100-334 mg/5 ml Soln 480 mL - - I
1463986	1463967	05/20/10	K-Phos Neutral
1463987	1463967	09/13/10	
1463988	1463967	09/02/09	Potassium Phosphate/Sodium Phosphate Tab
1463989	1463967	09/02/09	Potassium Phosphate/Sodium Phosphate
1463990	1463967	09/02/09	KPHOSTAB
1463991	1463967	09/02/09	
1463992	1463967	09/02/09	Potassium PhosphateKPHOSTABodium Phosphate Tab - KPHOSTAB
1464011	1463967	10/20/10	64980-0104-01
1464012	1463967	10/20/10	Potassium Phosphate/Sodium Phosphate Tab - - Inpatient - All - A
1464032	1464013	07/13/10	Coumadin
1464033	1464013	07/13/10	
1464034	1464013	09/02/09	Warfarin sodium *INJ* 5 mg vial
1464035	1464013	09/02/09	Warfarin
1464036	1464013	09/02/09	COUM5IV
1464037	1464013	09/02/09	
1464038	1464013	09/02/09	Warfarin sodium *INJ* 5 mg vial - COUM5IV - Inpatient - All - Acti
1464055	1464013	07/13/10	00590-0324-35
1464056	1464013	07/13/10	Warfarin sodium *INJ* 5 mg vial - - Inpatient - All - Active - 0059
1464060	1463331	07/09/10	Ceftazidime 1 gm Add-vantage
1464064	1463375	07/12/10	Cefuroxime 1.5 gm in 100 mL D5W Add-vantage
1464068	1463419	07/27/10	Cefuroxime 750mg Add-vantage
1464071	1463147	07/07/10	Carbidopa 25mg Tab
1464074	1463193	07/07/10	Carvedilol 80mg ER Tab

Figure 12. Extract from the MED_IDENTIFIER data table with *med_idenitfier_id* (col A) and *item_id* (col B) values. Each new *item_id* value is one more than the previous *med_identifier_id* value. About 70% of the data set follows this sequence which is broken at *med_identifier_id* value 1464064.

The data table has 48060 entries. The following attributes appear to be renamed relative to the schema glossary: *m_flex_type_disp*, *m_med_identifier_type_disp*, and *m_pharmacy_type_disp*. The *value* and *value_key* attributes show data values with many inconsistent orthographies amongst like information (e.g. strings beginning with “zz”, number strings for national Drug Codes of the USA, with missing hyphens), which could point to data entry errors.

The data values for this table show a correspondence between the values of the PK, med_identifier_id and the FK item_id, that is systematic for about 35,000 records. For a small set of PK values, all of which have the same item_id value, it follows a contiguous sequence of integers of 10 values followed by a small gap in the sequence followed by a contiguous set of four numbers. The next item_id value will be the number following the last value of the med_identifier_id (See Figure 12). This allocation of one attribute value, which is a PK in another table, dependent on another PK is a strange configuration. It is usual to allocate PK values as an independent series of numbers, that ensures they are unique within their own set. An important strategy used for high security applications would be to generate them as series of digits using public/private key encryption which make them appear to be a random set of digits and so subsequently have no meaning to their sequential values. Also this sequence suggests to us that the PK values for MED_IDENTIFIER are allocated before those of ITEM_MASTER indicating the loading dependency of the data. This throws up an intriguing issue on the 6 item_id values that appear in ITEM_MASTER but not in MED_IDENTIFIER as discussed above {"590623", "590634", "590645", "590656", "1501266", "1495298"}. If these values are determined by the sequence of loading recods into the MED_IDENTIFIER table, wha was the weakness in the process that allowed these values to be loaded into ITEM_MASTER.

In an effort to understand the role of PKs and FKs in this application a study was made of the values in the table of key values from other tables. In this case focus was placed on the *ndc* PK attribute from the PHA_PRODUCT table. Although the *ndc* is not an attribute in the MED_IDENTIFIER table and therefore there is no referential integrity constraint to be enforced between the two tables, the values are in the table in the attribute VALUE. It seemed important to understand the reliability of National Drug Codes no matter where they are stored in the system even in redundant locations.

Value	Frequency	Irregular values	Value	Frequency	Irregular Values
Brand Name	7339	4	Pyxis Interface ID	3118	6
Charge Number	3174	6	Rx Device 1	312	3
Description	8350	42	Rx Misc 1	3018	18
Generic Name	3236	3	Rx Misc 2	2387	3
HCPCS Code	791	8	Rx Unique ID	8160	4
NDC	4404	9	Short Description	3770	12
			TOTAL	**48060**	**118**

Table 1. The complete set of values of the attribute *m_med_identifer_type_disp* and their frequencies from the table MED_IDENTIFIER.

The search for NDC values starts with the attribute *med_identifier_type_cd* as shown in the schema glossary but which seems to be transformed to the name *m_med_identifer_type_disp* in the data table itself. Table 1 shows the complete set of values of the attribute and their frequencies.

Across all the category values of the attribute *m_med_identifer_type_disp* there are a total of 118 irregular values. These are values that based on the surrounding values in the category seem to be decidedly out of place. For example the RX Device category clearly lists alphanumeric codes for devices but Figure 9 shows two dates and one number in three records.

med_identifier.csv

New Open Save Print Import Copy Paste Format Undo Redo AutoSum Sort A-Z Sort Z-A Gallery Toolbo

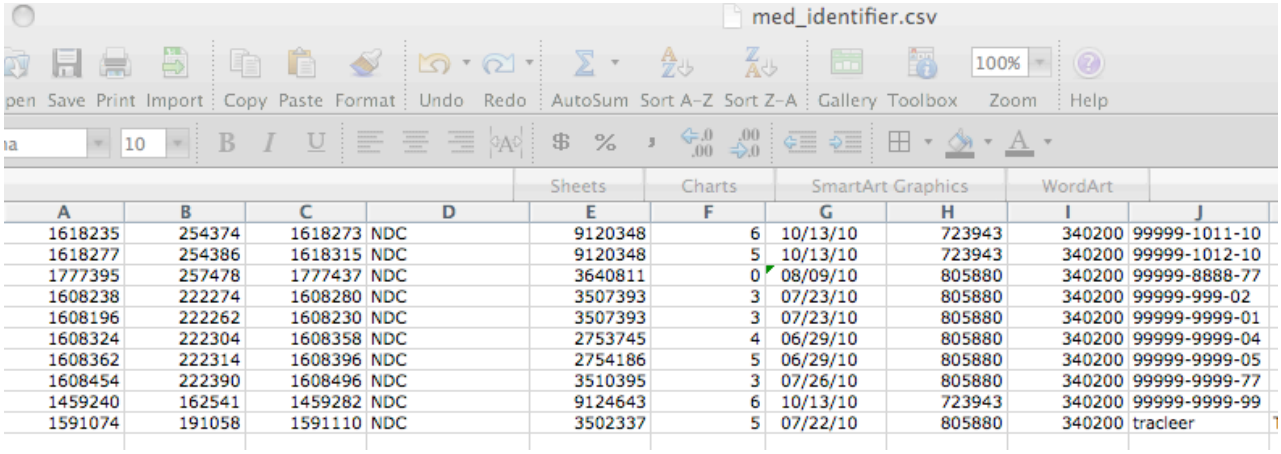
Verdana 10

Sheets Charts SmartArt Graphics

	A	B	C	D	E	F
1	ITEM_ID	MED_DEF_FLEX_ID	MED_IDENTIFIER_ID	M_MED_IDENTIFIER_TYPE_DISP	VALUE	VALUE_KEY
2	1320237	121497	1615460	Rx Device1	Apr-10	Apr-10
3	1289841	116539	1615804	Rx Device1	40516	40516
4	1320361	121517	1615462	Rx Device1	Apr-25	Apr-25
5	1368259	129335	1614059	Rx Device1	ACT15	ACT15
6	1368319	129345	1614061	Rx Device1	ACT30	ACT30
7	1386067	132225	1614063	Rx Device1	ALD25	ALD25
8	1344555	125463	1614065	Rx Device1	ALD250	ALD250
9	1376741	130715	1614067	Rx Device1	ALT25	ALT25
10	1376801	130725	1614069	Rx Device1	ALT5	ALT5
11	1481276	172060	1614071	Rx Device1	AMA2	AMA2
12	1481316	172070	1614073	Rx Device1	AMA4	AMA4
13	1258789	111422	1614075	Rx Device1	AMO250	AMO250
14	1259149	111482	1614077	Rx Device1	AMO500	AMO500

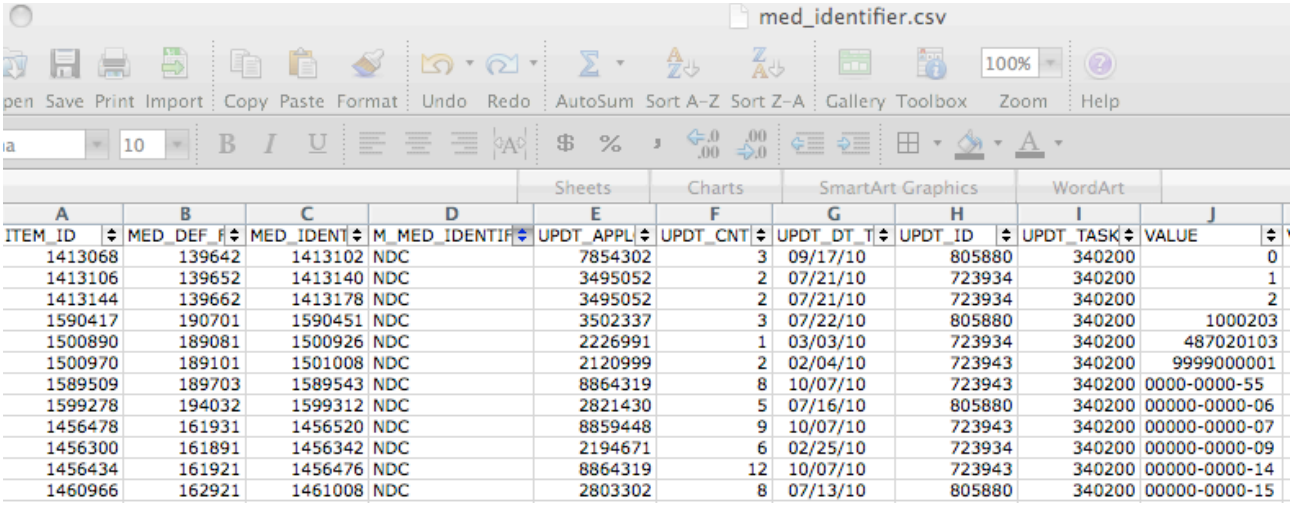
Figure 13. Three records (rows 2-4) for the RX Device *value* attribute that appear entirely irregular compared to other values in the table for the same category.

Filtering on the NDC value of attribute *m_med_identifer_type_disp* (col D) enables one to see the actual NDC values (col J) shown in the VALUE attribute and check if any show an irregular form. Figures 13 and 14 show the irregular data values of NDC codes in this attribute.



A	B	C	D	E	F	G	H	I	J
1618235	254374	1618273	NDC	9120348	6	10/13/10	723943	340200	99999-1011-10
1618277	254386	1618315	NDC	9120348	5	10/13/10	723943	340200	99999-1012-10
1777395	257478	1777437	NDC	3640811	0	08/09/10	805880	340200	99999-8888-77
1608238	222274	1608280	NDC	3507393	3	07/23/10	805880	340200	99999-999-02
1608196	222262	1608230	NDC	3507393	3	07/23/10	805880	340200	99999-9999-01
1608324	222304	1608358	NDC	2753745	4	06/29/10	805880	340200	99999-9999-04
1608362	222314	1608396	NDC	2754186	5	06/29/10	805880	340200	99999-9999-05
1608454	222390	1608496	NDC	3510395	3	07/26/10	805880	340200	99999-9999-77
1459240	162541	1459282	NDC	9124643	6	10/13/10	723943	340200	99999-9999-99
1591074	191058	1591110	NDC	3502337	5	07/22/10	805880	340200	tracleer

Figure 14. The first example of NDC values (col J) in the MED_IDENTIFIER data table which show irregular structure in the 4th and last rows.The regular structure can be seen in rows 1-3, and 5-9.



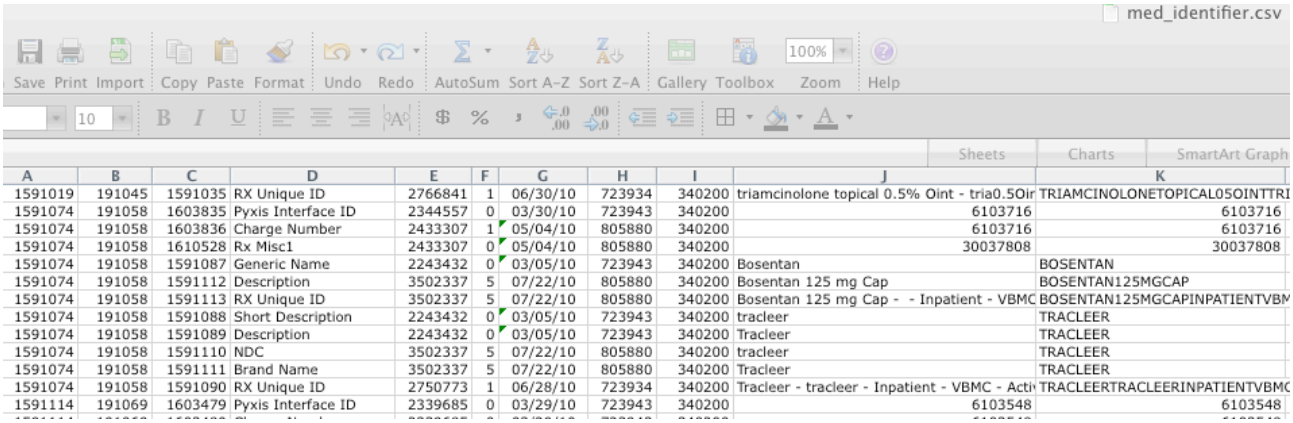
A	B	C	D	E	F	G	H	I	J
ITEM_ID	MED_DEF_F	MED_IDENT	M_MED_IDENTIF	UPDT_APPL	UPDT_CNT	UPDT_DT_T	UPDT_ID	UPDT_TASK	VALUE
1413068	139642	1413102	NDC	7854302	3	09/17/10	805880	340200	0
1413106	139652	1413140	NDC	3495052	2	07/21/10	723934	340200	1
1413144	139662	1413178	NDC	3495052	2	07/21/10	723934	340200	2
1590417	190701	1590451	NDC	3502337	3	07/22/10	805880	340200	1000203
1500890	189081	1500926	NDC	2226991	1	03/03/10	723934	340200	487020103
1500970	189101	1501008	NDC	2120999	2	02/04/10	723943	340200	9999000001
1589509	189703	1589543	NDC	8864319	8	10/07/10	723943	340200	0000-0000-55
1599278	194032	1599312	NDC	2821430	5	07/16/10	805880	340200	00000-0000-06
1456478	161931	1456520	NDC	8859448	9	10/07/10	723943	340200	00000-0000-07
1456300	161891	1456342	NDC	2194671	6	02/25/10	723934	340200	00000-0000-09
1456434	161921	1456476	NDC	8864319	12	10/07/10	723943	340200	00000-0000-14
1460966	162921	1461008	NDC	2803302	8	07/13/10	805880	340200	00000-0000-15

Figure 15. The second example of NDC values (col J) in the MED_IDENTIFIER data table which show irregular structure in rows 1-7.The regular structure can be seen from row 8 onwards.

A study of record sets for each *item_id* value of these irregular NDC values reveals no noticeable consistency between all the records with the irregular digit patterns. The most perplexing record is the last record in Figure 14 with no digits but the value of the string “tracleer”. All records that have the same *item_id* value of this record, 1591074, are shown in Figure 16. They show a semantically consistent set of records about the medicine Bosentan, also known as Tracleer.

It remains to explain how these irregular values were lodged into these tables in the first place. We have no direct knowledge of the loading mechanism for these values. Our deduction is that the PHA_PRODUCT has *ndc* as its PK so that is the first point at which these data values should be loaded and form the primary authoritative source for these values. Subsequently the values could be copied to other locations for local use but they should be referenced as a FK in those places so that automated checking with the authoritative source would always be operational when they are modified.

If these values are drawn from another source to be inserted into the table then they have not been validated against the authoritative source which would indicate missing referential integrity checking. If the values are inserted first in the authoritative table and these values are inapplicable then the mechanism for data validation has seemingly failed to operate effectively. This does not indicate that the PK uniqueness integrity checking has failed, as all these values {“0”, “1”, “2”, “1000203”, “487020103”, “9999000001”, “0000-0000-55”, “99999-999-02”, “tracleer”} although irregular are unique, though they are of variable datatypes, at least notionally (see Fig. 15).

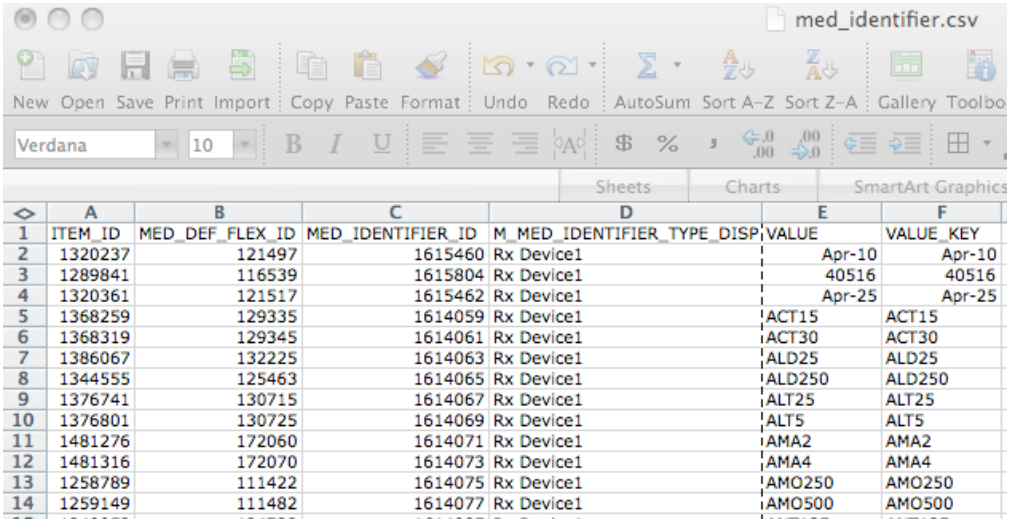

med_identifier.csv

A	B	C	D	E	F	G	H	I	J	K
1591019	191045	1591035	RX Unique ID	2766841	1	06/30/10	723934	340200	triamcinolone topical 0.5% Oint - tria0.5Oir	TRIAMCINOLONETOPICAL05OINTTRI
1591074	191058	1603835	Pyxis Interface ID	2344557	0	03/30/10	723943	340200	6103716	6103716
1591074	191058	1603836	Charge Number	2433307	1	05/04/10	805880	340200	6103716	6103716
1591074	191058	1610528	Rx Misc1	2433307	0	05/04/10	805880	340200	30037808	30037808
1591074	191058	1591087	Generic Name	2243432	0	03/05/10	723943	340200	Bosentan	BOSENTAN
1591074	191058	1591112	Description	3502337	5	07/22/10	805880	340200	Bosentan 125 mg Cap	BOSENTAN125MGCAP
1591074	191058	1591113	RX Unique ID	3502337	5	07/22/10	805880	340200	Bosentan 125 mg Cap - - Inpatient - VBMC	BOSENTAN125MGCAPINPATIENTVBM
1591074	191058	1591088	Short Description	2243432	0	03/05/10	723943	340200	tracleer	TRACLEER
1591074	191058	1591089	Description	2243432	0	03/05/10	723943	340200	Tracleer	TRACLEER
1591074	191058	1591110	NDC	3502337	5	07/22/10	805880	340200	tracleer	TRACLEER
1591074	191058	1591111	Brand Name	3502337	5	07/22/10	805880	340200	Tracleer	TRACLEER
1591074	191058	1591090	RX Unique ID	2750773	1	06/28/10	723934	340200	Tracleer - tracleer - Inpatient - VBMC - Acti	TRACLEERTRACLEERINPATIENTVBMC
1591114	191069	1603479	Pyxis Interface ID	2339685	0	03/29/10	723943	340200	6103548	6103548

Figure 16. The complete set of records (rows 1-3) for the irregular NDC code “tracleer” with the *item_id* values (col A) of 1591074. This shows that the set of records, under the same *item_id* code value, of themselves make a coherent grouping about the medicine Bosentan.

On the other hand if the table PHA_PRODUCT is not the authoritative source for the NDC values then it is difficult to understand the role of the NDC values as the PK of this table.

Across all the category values of the attribute *m_med_identifer_type_disp* there are a total of 118 irregular values. These are values based on the surrounding values in the category that seem decidedly out of place. For example the RX Device category clearly lists alphanumeric codes for devices but Figure 167shows two dates and one number in three records.


med_identifier.csv

	A	B	C	D	E	F
1	ITEM_ID	MED_DEF_FLEX_ID	MED_IDENTIFIER_ID	M_MED_IDENTIFIER_TYPE_DISP	VALUE	VALUE_KEY
2	1320237	121497	1615460	Rx Device1	Apr-10	Apr-10
3	1289841	116539	1615804	Rx Device1	40516	40516
4	1320361	121517	1615462	Rx Device1	Apr-25	Apr-25
5	1368259	129335	1614059	Rx Device1	ACT15	ACT15
6	1368319	129345	1614061	Rx Device1	ACT30	ACT30
7	1386067	132225	1614063	Rx Device1	ALD25	ALD25
8	1344555	125463	1614065	Rx Device1	ALD250	ALD250
9	1376741	130715	1614067	Rx Device1	ALT25	ALT25
10	1376801	130725	1614069	Rx Device1	ALT5	ALT5
11	1481276	172060	1614071	Rx Device1	AMA2	AMA2
12	1481316	172070	1614073	Rx Device1	AMA4	AMA4
13	1258789	111422	1614075	Rx Device1	AMO250	AMO250
14	1259149	111482	1614077	Rx Device1	AMO500	AMO500

Figure 17. Three records for the RxDevice value that appear entirely irregular compared to other values in the table for the same category.

An investigation of the HCPCS codes showed a value of 90371 which should contain an alpha character. On using the *item_id* to get a description of the item and then searching the HCPCS code list for the description we were able to establish the value should most likely be one of {J1571, J1573, Q4090}.

Overall the irregularity rate in the MED_IDENTIFIER table appears to be the order of approximately 1 in 400 (see Table 1). For some categories of information this would not appear to be critical but in a few cases for example where categories seem to be unique codes for medicines this may well be of crucial importance. One would have thought that categories such as NDC, HCPCS Code, Pyxis Interface ID and Rx Unique ID carried information important enough for patient care not to want it to have any anomalies.

The anomalies in the NDC codes seem to have serious consequence for the user if they use the correct code to search for a drug as they will not find these drugs, so in effect information about that drug is not available to the user.

See PHA_PRODUCT for related discussion on the integrity of National Drug Codes.

7. PERSON

The PERSON schema contains the contents of the record you would expect for a patient in a clinical setting. The PK is *person_id* as to be expected (see Fig. 18). There are 13 index/FKs, namely, *deceased_dt_tm (DQ8), last_accessed_dt_tm (DQ8), name_first_key (VC100), name_first_key_a_nls (VC400), name_first_key_nls (VC202), name_first_phonetic (C8), name_last_key (VC100), name_last_key_a_nls (VC400), name_last_key_nls (VC202), name_last_phonetic (C8), name_middle_key_a_nls (VC400), name_phonetic (C8), next_restore_dt_tm (DQ8).* Three are Date/Time attributes that might not normally be used as a PK. Two seem to be for auditing record purposes, *last_accessed_dt_tm* and *next_restore_dt_tm,* while the third*, deceased_dt_tm,* would appear to be for a particular analytics purpose The remaining attributes are forms of a patient's names which normally might need to be indexed for rapid retrieval but not for joining with other tables where the patient unique identifier, *person_id*, should be used. The datatypes of these attributes also mitigates against them being used as PKs and FKs which normally would be expected to be integers. Hence we envisage that all of these attributes are intended to be used for indexing purposes to enhance rapid search and retrieval, which is entirely appropriate. However this does reinforce the weakness of not having a clear separation between the identification of FKs and indices.

Table: PERSON

Fields	Type	Code Set	Flag	Definition
DECEASED_DT_TM	DQ8			The date and time of death for the person.
DECEASED_DT_TM_PREC_FLAG	I2		Y	Used to denote the precision of the deceased date and time. 0 - Precision is unknown. 1 - Date is precise to the full date a.
DECEASED_ID_METHOD_CD	F8	4002029		The Deceased_Id_Method_Cd will store code values defining the specific way a patient was identified as being deceased.
DECEASED_SOURCE_CD	F8	25513		Defines the particular source that gave deceased information concerning a person. For example, from a Formal (Death Cert
DECEASED_TZ	I4			The time zone where the deceased date was entered.
END_EFFECTIVE_DT_TM	DQ8			The date/time after which the row is no longer valid as active current data. This may be valued with the date that the row .
ETHNIC_GRP_CD	F8	27		Identifies a religious, national, racial, or cultural group of the person.
FT_ENTITY_ID	F8			Used to identify the ID of the freetext entity which this freetext person row is associated with.
FT_ENTITY_NAME	C32			Name of the entity for which this freetext person row is associated.
LANGUAGE_CD	F8	36		The primary language spoken by the person.
LANGUAGE_DIALECT_CD	F8	328		The dialect of the primary language spoken by the person.
LAST_ACCESSED_DT_TM	DQ8			This column contains the date/time this person was last accessed. It affects when the person will be archived when archi.
LAST_ENCNTR_DT_TM	DQ8			The date and time of the last encounter for the person.
LOGICAL_DOMAIN_ID	F8			The unique identifier for a logical domain. This identifier allows the data to be grouped by logical domain. For example, If yo.
MARITAL_TYPE_CD	F8	38		This field identifies the status of the person with regard to being married.
MILITARY_BASE_LOCATION	VC100			The location of the military base at which the person is stationed.
MILITARY_RANK_CD	F8	14758		Military ranking of individual (i.e. Private, Seargent, General, etc.)
MILITARY_SERVICE_CD	F8	14757		Military status of an individual (i.e. Active Duty, Reserves, etc.)
MOTHER_MAIDEN_NAME	VC100			The mother's last name she was given at birth.
NAME_FIRST	VC200			This is the person's given first name.
NAME_FIRST_KEY	VC100			This is the person's first given name all capitals with punctuation removed. This field is used for indexing and searching for
NAME_FIRST_KEY_A_NLS	VC400			NAME_FIRST_KEY_A_NLS column
NAME_FIRST_KEY_NLS	VC202			First Name Key field converted to NLS format for internationalization requirements
NAME_FIRST_PHONETIC	C8			Phonetic representation of person's first name.
NAME_FIRST_SYNONYM_ID	F8			First Name Synonym Id
NAME_FULL_FORMATTED	VC100			This is the complete person name including punctuation and formatting.
NAME_LAST	VC200			This is the person's family name.
NAME_LAST_KEY	VC100			This is the person's family name all capitals with punctuation removed. This field is used for indexing and searching for a pe
NAME_LAST_KEY_A_NLS	VC400			NAME_LAST_KEY_A_NLS column
NAME_LAST_KEY_NLS	VC202			Last Name Key field converted to NLS format for internationalization requirements
NAME_LAST_PHONETIC	C8			Phonetic representation of person's last name.
NAME_MIDDLE	VC200			This is the given middle name for the person.
NAME_MIDDLE_KEY	VC100			This is the person's middle name with all capitals with punctuation removed. This field is used for indexing and searching fo
NAME_MIDDLE_KEY_A_NLS	VC400			NAME_MIDDLE_KEY_A_NLS column
NAME_MIDDLE_KEY_NLS	VC202			Last Name Key field converted to NLS format for internationalization requirements
NAME_PHONETIC	C8			This is the Soundex coded representation of the person's name. This field is used for indexing and searching for a patient .
NATIONALITY_CD	F8	14652		This field Identifies the nationality associated with the person.
NEXT_RESTORE_DT_TM	DQ8			This column contains the date/time when this person needs to be restored from archive.
PERSON_ID	F8			This is the value of the unique primary identifier of the person table. It is an internal system assigned number.
PERSON_TYPE_CD	F8	302		The person type field identifies the general type of data being stored in a given person row. As a general guideline, most r.
PURGE_OPTION_CD	F8	46		OBSOLETE: Purge Option Code Value
RACE_CD	F8	282		A group of people classified together on the basis of common history, nationality, or geographical distribution.
RELIGION_CD	F8	49		A particular integrated system of belief in a supernatural power.

Figure 18. Part of the schema glossary for the table PERSON. The PK is *person_id*.

There are no data tables available for the PERSON schema but a screenshot shows that the PKs are listed as floating point numbers (see Figure 19).

View Program | Results

	PERSON_ID	P_PERSON_TYPE_DISP	P_PURGE_OPTION_DISP	P_RACE_DISP
1	0.00			
2	1.00			
3	2.00	Person		
4	3.00			
5	21.00			
6	589763.00	Person		
7	589823.00	Person		
8	589824.00	Person		
9	589825.00	Person		
10	589826.00	Person		
11	589827.00	Person		
12	589828.00	Person		
13	589829.00	Person		
14	589830.00	Person		
15	589831.00	Person		
16	589832.00	Person		Caucasian/White

Figure 19. A screenshot of the data table PERSON showing the floating point values and sequential values of the PK *person_id*.

The meaning of the PK attribute *person_id* is enigmatic as it is used again as the PK of the PRSNL schema. A study of the "code" sets of each of these attributes is revealing. The attribute *person_type_cd* is defined by a code set 302 and has the 4 values 899=Contributor System, 900=Freetext, 901=Numeric Name, 902=Organisation, and 903=Person (see Fig. 20). One interpretation of this structure is that PERSON is a set of 4 "categories of persons", that is not "persons" per se, but that doesn't fit well with the remaining attributes in the schema, such as first, middle and last name, military rank, deceased date, etc. This more than likely seems to be a process known as "overloading" where an attribute, *person_id*, is used for more than one purpose and the meaning in any particular record has to be interpreted from the other attribute information that accompanies it. This is generally an ill-advised strategy as it requires all people maintaining the code that has dependencies that are not self-evident to understand the overloading characteristics, in perpetuity. See PRSNL for an extension of this topic.

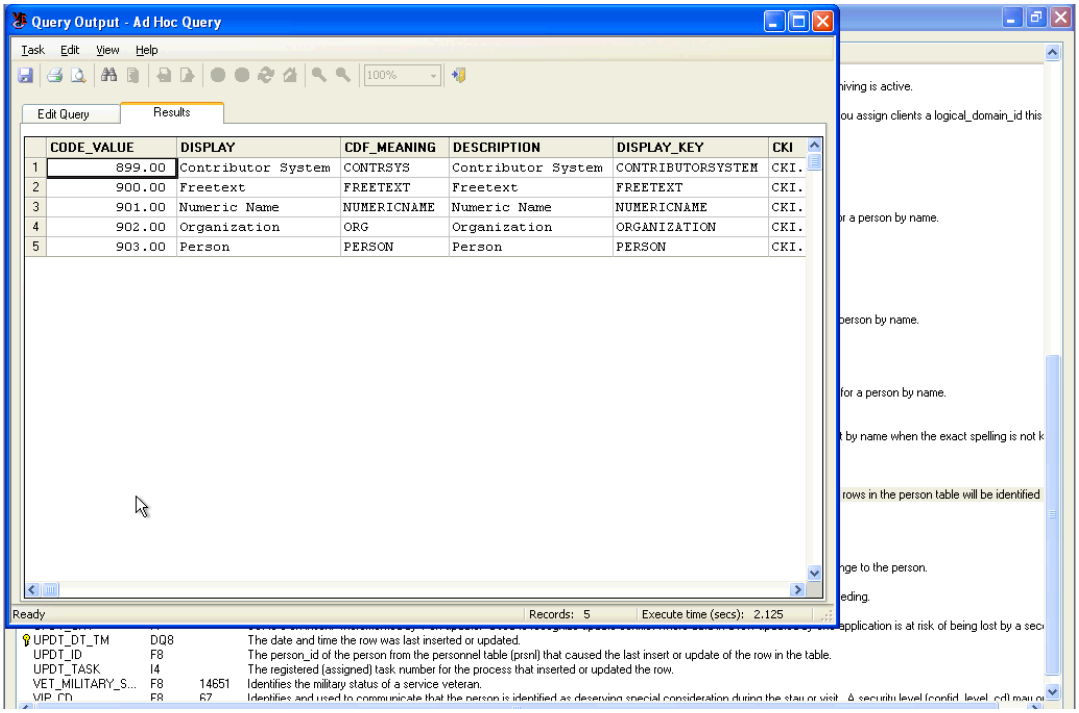


Figure 20. The code set values for the attribute *person_type_cd* from the table PERSON.

8. PHA_PRODUCT

The PHA_PRODUCT schema is shown in the schema glossary as having a single PK *ndc* (National Drug Code) (see Figure 21). This PK has values repeated multiple times in the data table (see Figure 22) and this would appear to be serious weakness in the design as we would expect all values to be unique. It may be that in the implementation of this PK it is concatenated with another attribute to create a unique PK but there is no evidence for that strategy in the information we have available. On the face of the evidence this is an a example of non-unique PKs in the data table.

There are 22 putative FKs and no actual index/FKs shown in the glossary. The *ndc* attribute is not used elsewhere in the schemata available to us but that does not exclude its use in parts of the schemata we have not seen. Nevertheless the ITEM_MASTER schema seems to also be one of the tables recording pharmacy details and its PK, *item_id,* is present in this schema without being identified as an index/FK.

It is not a requirement of any software using a commercially supplied database management system, in this case Oracle, to create a PK using the built-in functions for maintaining the uniqueness of PKs. It is entirely feasible for an application to ignore the supplied functionality and there is a long history in the 1960s and 1970s of software applications that managed the PKs in the application code. However since the 1980s vendor supplied functions have been seen as the safest strategy for protecting the uniqueness integrity of PKs. In this data table we have not been able to identify a likely attribute that we might expect has truly unique values for each record which might be maintained by the internal software as a PK.

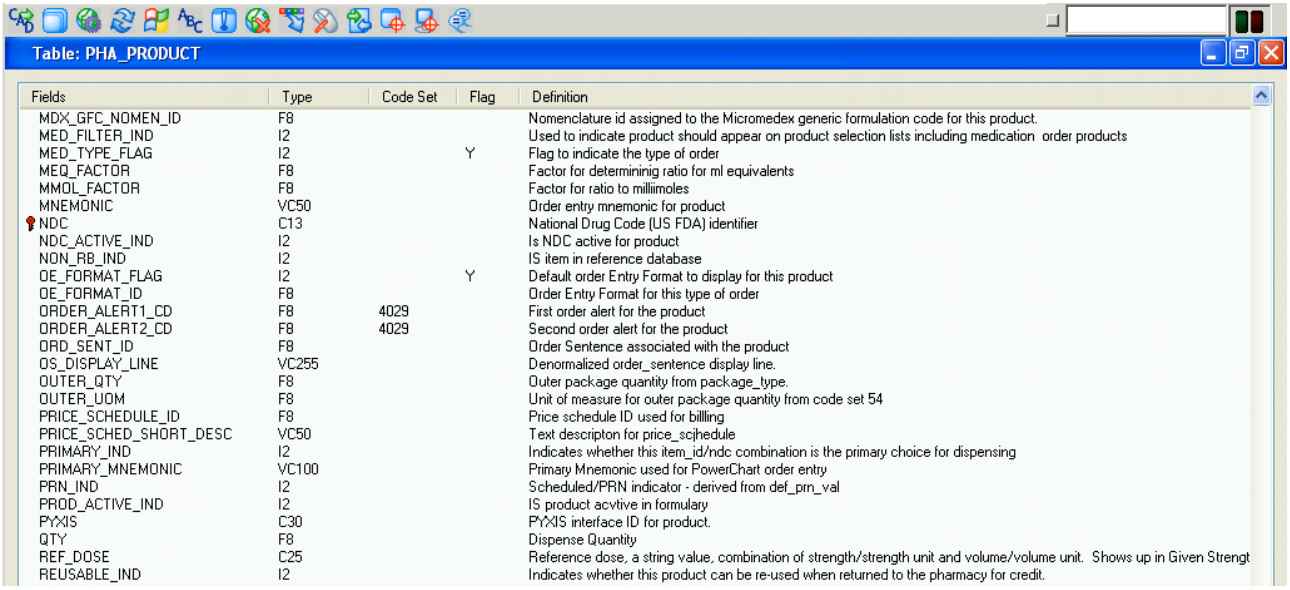
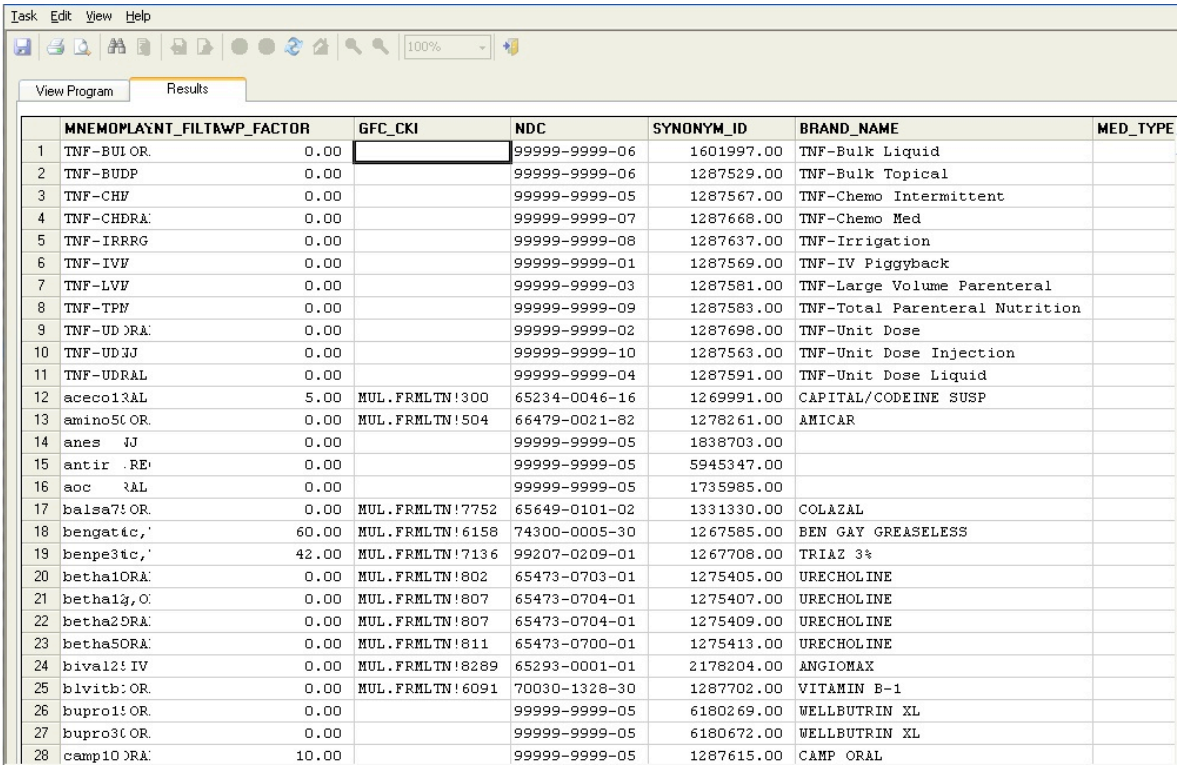


Table: PHA_PRODUCT

Fields	Type	Code Set	Flag	Definition
MDX_GFC_NOMEN_ID	F8			Nomenclature id assigned to the Micromedex generic formulation code for this product.
MED_FILTER_IND	I2			Used to indicate product should appear on product selection lists including medication order products
MED_TYPE_FLAG	I2		Y	Flag to indicate the type of order
MEQ_FACTOR	F8			Factor for determininig ratio for ml equivalents
MMOL_FACTOR	F8			Factor for ratio to milliimoles
MNEMONIC	VC50			Order entry mnemonic for product
NDC	C13			National Drug Code (US FDA) identifier
NDC_ACTIVE_IND	I2			Is NDC active for product
NON_RB_IND	I2			IS item in reference database
OE_FORMAT_FLAG	I2		Y	Default order Entry Format to display for this product
OE_FORMAT_ID	F8			Order Entry Format for this type of order
ORDER_ALERT1_CD	F8	4029		First order alert for the product
ORDER_ALERT2_CD	F8	4029		Second order alert for the product
ORD_SENT_ID	F8			Order Sentence associated with the product
OS_DISPLAY_LINE	VC255			Denormalized order_sentence display line.
OUTER_QTY	F8			Outer package quantity from package_type.
OUTER_UOM	F8			Unit of measure for outer package quantity from code set 54
PRICE_SCHEDULE_ID	F8			Price schedule ID used for billling
PRICE_SCHED_SHORT_DESC	VC50			Text descripton for price_scjhedule
PRIMARY_IND	I2			Indicates whether this item_id/ndc combination is the primary choice for dispensing
PRIMARY_MNEMONIC	VC100			Primary Mnemonic used for PowerChart order entry
PRN_IND	I2			Scheduled/PRN indicator - derived from def_prn_val
PROD_ACTIVE_IND	I2			IS product acvtive in formulary
PYXIS	C30			PYXIS interface ID for product.
QTY	F8			Dispense Quantity
REF_DOSE	C25			Reference dose, a string value, combination of strength/strength unit and volume/volume unit. Shows up in Given Strengt
REUSABLE_IND	I2			Indicates whether this product can be re-used when returned to the pharmacy for credit.

Figure 21. Part of the schema glossary for the table PHA_PRODUCT with the annotation of *ndc* as a PK.



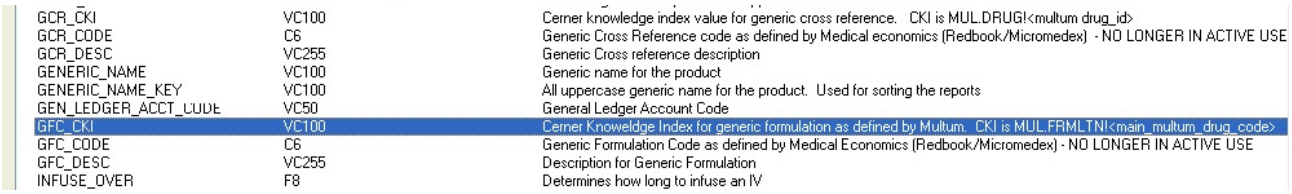
Task Edit View Help

View Program | Results

	MNEMONIC_FILTER_FACTOR	GFC_CKI	NDC	SYNONYM_ID	BRAND_NAME	MED_TYPE
1	TNF-BULK OR. 0.00		99999-9999-06	1601997.00	TNF-Bulk Liquid	
2	TNF-BULK 0.00		99999-9999-06	1287529.00	TNF-Bulk Topical	
3	TNF-CHEMO 0.00		99999-9999-05	1287567.00	TNF-Chemo Intermittent	
4	TNF-CHEMO 0.00		99999-9999-07	1287668.00	TNF-Chemo Med	
5	TNF-IRRIG 0.00		99999-9999-08	1287637.00	TNF-Irrigation	
6	TNF-IVPB 0.00		99999-9999-01	1287569.00	TNF-IV Piggyback	
7	TNF-LVP 0.00		99999-9999-03	1287581.00	TNF-Large Volume Parenteral	
8	TNF-TPN 0.00		99999-9999-09	1287583.00	TNF-Total Parenteral Nutrition	
9	TNF-UD ORAL 0.00		99999-9999-02	1287698.00	TNF-Unit Dose	
10	TNF-UD INJ 0.00		99999-9999-10	1287563.00	TNF-Unit Dose Injection	
11	TNF-UDLIQ 0.00		99999-9999-04	1287591.00	TNF-Unit Dose Liquid	
12	acecol3AL 5.00	MUL.FRMLTN!300	65234-0046-16	1269991.00	CAPITAL/CODEINE SUSP	
13	amino5(OR. 0.00	MUL.FRMLTN!504	66479-0021-82	1278261.00	AMICAR	
14	anes IJ 0.00		99999-9999-05	1838703.00		
15	antir .RE 0.00		99999-9999-05	5945347.00		
16	aoc AL 0.00		99999-9999-05	1735985.00		
17	balsa7! OR. 0.00	MUL.FRMLTN!7752	65649-0101-02	1331330.00	COLAZAL	
18	bengatic,' 60.00	MUL.FRMLTN!6158	74300-0005-30	1267585.00	BEN GAY GREASELESS	
19	benpe3ic,' 42.00	MUL.FRMLTN!7136	99207-0209-01	1267708.00	TRIAZ 3%	
20	betha1ORA 0.00	MUL.FRMLTN!802	65473-0703-01	1275405.00	URECHOLINE	
21	betha1g,O 0.00	MUL.FRMLTN!807	65473-0704-01	1275407.00	URECHOLINE	
22	betha2ORA 0.00	MUL.FRMLTN!807	65473-0704-01	1275409.00	URECHOLINE	
23	betha5ORA 0.00	MUL.FRMLTN!811	65473-0700-01	1275413.00	URECHOLINE	
24	biva125 IV 0.00	MUL.FRMLTN!8289	65293-0001-01	2178204.00	ANGIOMAX	
25	blvitb OR. 0.00	MUL.FRMLTN!6091	70030-1328-30	1287702.00	VITAMIN B-1	
26	bupro15 OR. 0.00		99999-9999-05	6180269.00	WELLBUTRIN XL	
27	bupro3(OR. 0.00		99999-9999-05	6180672.00	WELLBUTRIN XL	
28	camp10 ORA 10.00		99999-9999-05	1287615.00	CAMP ORAL	

Figure 22. Screenshot of the data of PHA_PRODUCT Table. Entries in the PHA_PRODUCT table with multiple occurrences of the PK, *ndc* : 9999-9999-06, 9999-9999-05, 65473-0704-01 and again 9999-9999-05.

The *gcr_cki* and *gfc_cki* attributes are defined with a datatype of CV100 (see Fig. 23) whereas in the MEDICATION_DEFINTION schema the *cki* is defined as VC255 (see Fig. 9). Their Definitions do identify these two attributes as the same information as *cki* but there is no information as to which schema uses them as the PK. On the surface, this does appear to be another example of data duplicated across tables without a clear raison d'etre.



GCR_CKI	VC100	Cerner knowledge index value for generic cross reference. CKI is MUL.DRUG!<multum drug_id>
GCR_CODE	C6	Generic Cross Reference code as defined by Medical economics (Redbook/Micromedex) - NO LONGER IN ACTIVE USE
GCR_DESC	VC255	Generic Cross reference description
GENERIC_NAME	VC100	Generic name for the product
GENERIC_NAME_KEY	VC100	All uppercase generic name for the product. Used for sorting the reports
GEN_LEDGER_ACCT_CODE	VC50	General Ledger Account Code
GFC_CKI	VC100	Cerner Knowledge Index for generic formulation as defined by Multum. CKI is MUL.FRMLTN!<main_multum_drug_code>
GFC_CODE	C6	Generic Formulation Code as defined by Medical Economics (Redbook/Micromedex) - NO LONGER IN ACTIVE USE
GFC_DESC	VC255	Description for Generic Formulation
INFUSE_OVER	F8	Determines how long to infuse an IV

Figure 23. Part of the schema glossary for the table PHA_PRODUCT showing references to *gcr_cki* and *gfc_cki* FK attributes as datatypes VC100.

9. PRSNL

The schema of PRSNL has the PK *person_id* whereas by convention it should be *prsnl_id* and in fact this form is used in various places throughout the schemata and ERDs. It also has *username* (VC50) as a PK and using the search function in the glossary tool produces a reference to "PERSON_ID+USERNAME" suggesting a concatenated PK (see Fig. 24). The need for a concatenated PK using these two fields is without explanation and difficult to comprehend. As *username* is the login name of a user and it is easy to understand that an index/FK might be sensibly created on this attribute, but this does not require a PK declaration. There are 4 index/FKs of attributes using the names of persons, namely, *name_first_key_a_nls* (VC400), *name_last_key* (VC100), *name_last_key_a_nls* (VC400), *name_last_key_nls* (VC202). There are 2 index/FKs, namely, *physician_ind* (I2), and *position_cd* (F8), which are most likely to be indices for search and retrieval.

There are 6 putative FKs all of which indicate a weakness in design. The three attributes *active_status_prsnl_id*, *create_prsnl_id*, and *data_status_prsnl_id* appear to be references to personnel and hence are self references to this schema table. The attribute *ft_entity_id* is defined as the "primary key of the table row associated with a free text row". This unambiguously indicates that this attribute is a PK elsewhere in the system and warrants referencing here as a FK.

NAME_FIRST	VC200		This is the person's first given name.
NAME_FIRST_KEY	VC100		This is the person's first given name all capitals with punctuation removed. This field is used for indexing and searching f
NAME_FIRST_KEY_A_NLS	VC400		NAME_FIRST_KEY_A_NLS column
NAME_FIRST_KEY_NLS	VC202		First Name Key field converted to NLS format for internationalization requirements
NAME_FULL_FORMATTED	VC100		This is the complete person name including punctuation and formatting.
NAME_LAST	VC200		This is the person's family name.
NAME_LAST_KEY	VC100		This is the person's family name all capitals with punctuation removed. This field is used for indexing and searching for a
NAME_LAST_KEY_A_NLS	VC400		NAME_LAST_KEY_A_NLS column
NAME_LAST_KEY_NLS	VC202		Last Name Key field converted to NLS format for internationalization requirements
PASSWORD	VC100		Encrypted form of the data store password used to authorize Millenium users access to Millenium Mobile devices.
PERSON_ID	F8		This is the value of the unique primary identifier of the person table. It is an internal system assigned number.
PHYSICIAN_IND	I2		Set to TRUE, if the personnel is a physician. Otherwise, set to FALSE.
PHYSICIAN_STATUS_CD	F8	14647	Physician status code identifies the status of the physician. (For Example: In, In Surgery, Out, etc...)
POSITION_CD	F8	88	The position is used to determine the applications and tasks the personnel is authorized to use.
PRIM_ASSIGN_LOC_CD	F8	220	Primary Assigned Location Code identifies the primary location to which a personnel will be assigned.
PRSNL_TYPE_CD	F8	309	The personnel type is used to group personnel with common characteristics (I.e., user, non-user, template user)
USERNAME	VC50		The system user name for the personnel used to gain primary access to the computer system.

Figure 24. Two parts of the schema glossary for the table PRSNL showing the multiple PKs *person_id* and *username*.

The putative FK *logical_domain_id* is defined to be “the unique identifier for a logical domain” and so should normally be enforced as a FK.

As with the PERSON schema there is an attribute that moderates the grouping of the PRSNL record. The attribute is *prsnl_type_cd* with a code set 309 which has 3 code values: 904=Contributor System, 905=Template, 906=User (see Fig. 25). It seems more than coincidence that the code values across the PERSON and PRSNL data tables have a continuous sequence from 899 to 906 and that they are more than likely compiled at the one time. The code item common to both code sets is “Contributor System”, the meaning of which escapes us when the tables are meant to be about PERSONs and PERSONNEL.

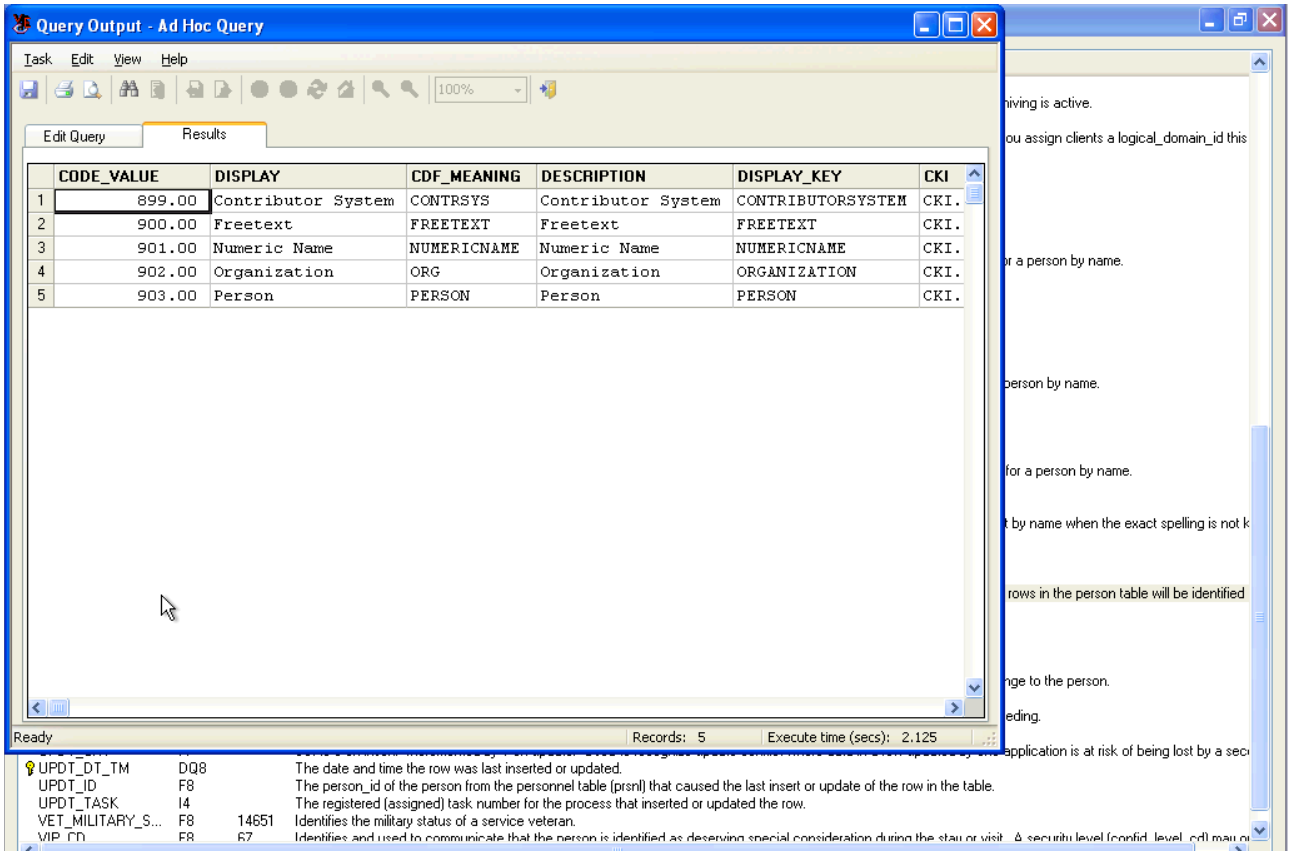


Figure 25. The code values for the attribute *prnsl_type_cd* from the table PRSNL.

10. REF_TEXT

The data table reveals as with other tables the PKs and index/FKs are implemented as floating point numbers. The PK is a two-part concatenated key (see Figure 26). The PK values for *refr_text_id* are consecutive odd numbers while the values for *text_entity_id* are consecutive numbers (see Figure 27). See REF_TEXT_RELTN for further discussion. The PK attribute *ref_text_name* is a character field and would not normally be expected to be used as a PK as well as being empty in the data table. There is no obvious reason to expect that the attribute *refr_text_id* wouldn’t be a unique identifier and therefore the use of *ref_text_name* as a PK is redundant.

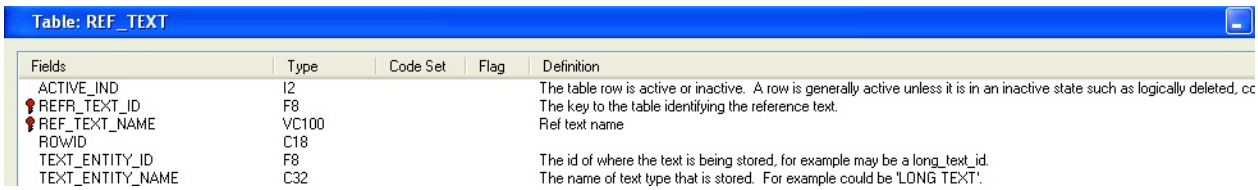


Table: REF_TEXT

Fields	Type	Code Set	Flag	Definition
ACTIVE_IND	I2			The table row is active or inactive. A row is generally active unless it is in an inactive state such as logically deleted, cc
REFR_TEXT_ID	F8			The key to the table identifying the reference text.
REF_TEXT_NAME	VC100			Ref text name
ROWID	C18			
TEXT_ENTITY_ID	F8			The id of where the text is being stored, for example may be a long_text_id.
TEXT_ENTITY_NAME	C32			The name of text type that is stored. For example could be 'LONG TEXT'.

Figure 26. Part of the schema glossary for the table REF_TEXT showing a two-part PK, *refr_text_id* and *ref_text_name*.

	ACTIVE_IND	REFR_TEXT_ID	REF_TEXT_NAME	ROWID	TEXT_ENTITY_ID	TEXT_ENTITY_NAME
1	0	808021.00		AAA2GkAAGAAAANkAAA	606923.00	LONG_TEXT
2	0	808023.00		AAA2GkAAGAAAANkAAB	606924.00	LONG_TEXT
3	0	808025.00		AAA2GkAAGAAAANkAAC	606925.00	LONG_TEXT
4	0	808027.00		AAA2GkAAGAAAANkAAD	606926.00	LONG_TEXT
5	0	808029.00		AAA2GkAAGAAAANkAAE	606927.00	LONG_TEXT

Figure 27. Screenshot of the data table for REF_TEXT showing consecutive odd numbered values of the PK *refr_text_id* and consecutive numbers of the putative FK *text_entity_id,* and empty values for the PK *ref_text_name*.

11. REF_TEXT_RELTN

An interesting phenomena is the allocation of PK functions for *refr_text_id* and *ref_text_reltn_id* are always successive numbers (see Fig. 29). This implies that the process of allocating the PK values uses successive counting managed by the application software and not by the database management system which would be the normal method. The data table reveals as with other tables the PKs and index/FKs are implemented as floating point numbers.

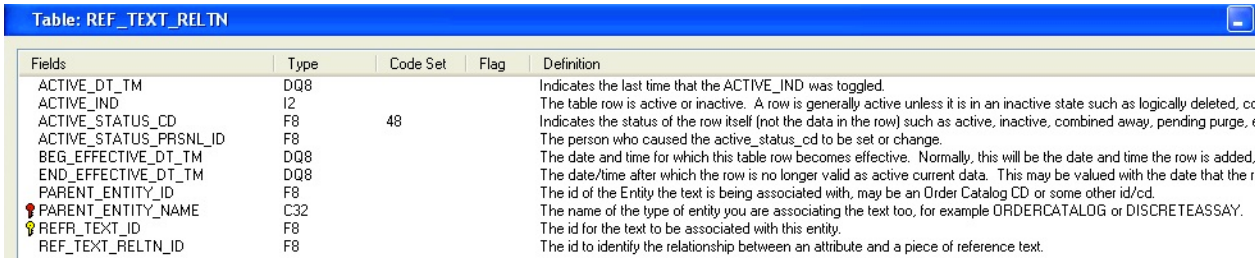


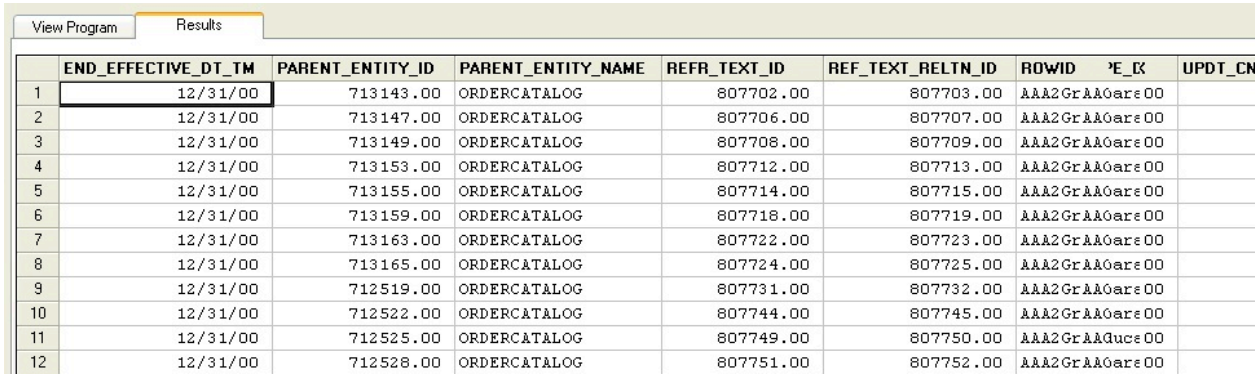
Table: REF_TEXT_RELTN

Fields	Type	Code Set	Flag	Definition
ACTIVE_DT_TM	DQ8			Indicates the last time that the ACTIVE_IND was toggled.
ACTIVE_IND	I2			The table row is active or inactive. A row is generally active unless it is in an inactive state such as logically deleted, cc
ACTIVE_STATUS_CD	F8	48		Indicates the status of the row itself (not the data in the row) such as active, inactive, combined away, pending purge, e
ACTIVE_STATUS_PRSNL_ID	F8			The person who caused the active_status_cd to be set or change.
BEG_EFFECTIVE_DT_TM	DQ8			The date and time for which this table row becomes effective. Normally, this will be the date and time the row is added,
END_EFFECTIVE_DT_TM	DQ8			The date/time after which the row is no longer valid as active current data. This may be valued with the date that the r
PARENT_ENTITY_ID	F8			The id of the Entity the text is being associated with, may be an Order Catalog CD or some other id/cd.
PARENT_ENTITY_NAME	C32			The name of the type of entity you are associating the text too, for example ORDERCATALOG or DISCRETEASSAY.
REFR_TEXT_ID	F8			The id for the text to be associated with this entity.
REF_TEXT_RELTN_ID	F8			The id to identify the relationship between an attribute and a piece of reference text.

Figure 28. Part of the Schema Glossary for the table REF_TXT_RELTN where the declared PK, *parent_entity_name* is a character field, and the putative PK *refr_text_reltn_id* is declared as an ordinary attribute.

The attribute *parent_entity_name* (C32) is shown in the schema as a PK however the data values in the example provided are all the same, “ORDERCATALOG” so this is contradictory (see Figure 29). This is a clear example where PK uniqueness is not enforced.

One mystifying aspect in the the schema glossary is the attribute *ref_text_reltn_id*. The attribute *ref_text_reltn_id* is a set of numbers that appear a reasonable candidate for the PK and whose *definition* fits the notional purpose of this table, that is, “The id to identify the relationship between an attribute and a piece of reference text.”



View Program | Results

	END_EFFECTIVE_DT_TM	PARENT_ENTITY_ID	PARENT_ENTITY_NAME	REFR_TEXT_ID	REF_TEXT_RELTN_ID	ROWID ?E_I?	UPDT_CN
1	12/31/00	713143.00	ORDERCATALOG	807702.00	807703.00	AAA2GrAAGare00	
2	12/31/00	713147.00	ORDERCATALOG	807706.00	807707.00	AAA2GrAAGare00	
3	12/31/00	713149.00	ORDERCATALOG	807708.00	807709.00	AAA2GrAAGare00	
4	12/31/00	713153.00	ORDERCATALOG	807712.00	807713.00	AAA2GrAAGare00	
5	12/31/00	713155.00	ORDERCATALOG	807714.00	807715.00	AAA2GrAAGare00	
6	12/31/00	713159.00	ORDERCATALOG	807718.00	807719.00	AAA2GrAAGare00	
7	12/31/00	713163.00	ORDERCATALOG	807722.00	807723.00	AAA2GrAAGare00	
8	12/31/00	713165.00	ORDERCATALOG	807724.00	807725.00	AAA2GrAAGare00	
9	12/31/00	712519.00	ORDERCATALOG	807731.00	807732.00	AAA2GrAAGare00	
10	12/31/00	712522.00	ORDERCATALOG	807744.00	807745.00	AAA2GrAAGare00	
11	12/31/00	712525.00	ORDERCATALOG	807749.00	807750.00	AAA2GrAAGuce00	
12	12/31/00	712528.00	ORDERCATALOG	807751.00	807752.00	AAA2GrAAGare00	

Figure 29. Screenshot of REF_TEXT_RELATION data table showing identical values of the supposed PK, *parent_entity_name*.

Appendix 2 - Table of relational tables with PKs, Index/FKs, and putative FKs

Table Name	PK	Index/FK Attributes	_id-denoted putative FK attributes
ACCESS_CONTROL_POLICY	*access_control_policy_id*	nil	*access_control_type_entity_id* *data_source_entity_id*
ACCESS_CONTROL_POLICY	*access_control_policy_id*	nil	*access_control_type_entity_id,* *data_source_entity_id*
DCP_FORMS_ACTIVITY	*dcp_forms_activity_id*	*dcp_forms_ref_id* *enctr_id* *person_id* *task_id* *updt_dt_tm (DQ8)*	*lock_prsnl_id*
DCP_FORMS_REF *	*dcp_form_instance_id*	*dcp_fomrs_ref_id*	nil
ITEM_MASTER	*item_id*	nil	nil
MEDICATION_DEFINITION	*item_id*	*cki (VC255), inv_master_id,* *parent_item_id,* *primary_manufacturer_item_id.*	*comment1_id, comment2_id,* *compound_text_id,* *mdx_gfc_nomen_id,* *order_sentence_id,* *price_sched_id,*
MED_IDENTIFIER	*med_identifier_*	*item_id* *pharmacy_type_cd*	*med_def_flex_id* *med_ingred_set_id* *med_package_type_id* *med_product_id* *parent_entity_id*
PERSON	*person_id*	*deceased_dt_tm (DQ8)* *last_accessed_dt_tm (DQ8)* *name_first_key (VC100)* *name_first_key_a_nls (VC400)* *name_first_key_nls (VC202)* *name_first_phonetic (C8)* *name_last_key (VC100)* *name_last_key_a_nls (VC400)* *name_last_key_nls (VC202)* *name_last_phonetic (C8)* *name_middle_key_a_nls (VC400)* *name_phonetic (C8)* *next_restore_dt_tm (DQ8)*	*ft_entity_id* *logical_domain_id* *name_first_synonym_id*

Table Name	PK	Index/FK Attributes	_id-denoted putative FK attributes
PHA_PRODUCT	*ndc*	nil	*alt_sel_category_id,* *comment1_id,* *comment2_id,* *compound_text_id,* *def_durunit_id,* *def_dur_id,* *def_freq_id,* *def_ftdose_id,* *def_prn_id,* *def_route_id,* *def_stop_id,* *def_strunit_id,* *def_str_id,* *def_volunits_id,* *def_vol_id,* *diluent_id,* *ing_sent_id (not in use)* *gcr_cki (VC100),* *item_id (Primary key – Item id for the product),* *manf_item_id (Item_id for manufacturer_item associated with this product),* *mdx_gfc__nomen_id,* *oe_format_id,* *order_sent_id,* *price_schedule_id,* *synonym_id.*
PRSNL	*person_id* *username (VC50)*	*name_first_key_a_nls (VC400)* *name_last_key (VC100)* *name_last_key_a_nls (VC400)* *name_last_key_nls (VC202)* *physician_ind (12)* *position_cd (F8)*	*active_status_prsnl_id* *create_prsnl_id* *data_status_prsnl_id* *ft_entity_id* *logical_domain_group_id* *logical_domain_id*
REF_TEXT_RELTN	*parent_entity_name (C32)*	*refr_text_id*	*parent_entity_id* *ref_text_reltn_id*
REF_TEXT	*refr_text_id* *ref_text_name*	nil	*text_entity_id*