

**AUTOMATIC
NORMALISATION
AND
ENTITY - RELATIONSHIP
GENERATION
THROUGH
ATTRIBUTES AND
ROLES**

Work Paper

20th February, 1984

Ken Meyer
British Gas
and
John Doughty
Intelligence Engineering

The opinions expressed in this paper are those of the authors and do not represent the opinions, statements of policies of their respective employers and of the DDSWP.

This paper is subject to copyright and is to be circulated for publication.

1. Introduction

The development of methodologies for construction of a Pure Conceptual Data Model concerned only with the meaning and structure of data independent of all other considerations has proceeded for more than fifteen years (IOS 84). Some eleven methodologies with phases and guidelines have been in serious contention in the U.K., as universities, business sites, government agencies and software consultancies plan for long-term efforts in constructing conceptual views of data, modelling the data and designing application databases around these methodologies (MAD 83).

In the late 'seventies, the then existing methodologies were examined by Iosson (IOS 80) and the two most directly influencing the Gas Industry, International Computers Limited (ICL) Data Analysis Methodology (ICL 80) and the Learmonth and Burchett Management Systems (LBM 77) Logical Data Analysis, were specifically scrutinized in planning a methodology to be used at North Eastern Gas at Leeds, England (IOS 84).

In the methodology review process, a critical evaluation was performed by S. J. Pinder, I.G. Smith and D. Iosson, with the participation of P. Reynard and A. Gardner. Mr. Meyer suggested, in following the ideas set forth by Pinder, that a substantial saving could be made in producing conceptual data models by entry of roles between attributes. The roles between attributes were thought to describe the data relations better than relationships between entities prescribed in both the ICL Data Analysis and the LBMS LDA techniques, both of which heavily rely on normalisation techniques.

The investigation proceeded on separate, yet parallel paths, with this paper as Meyer and Doughty's enquiry and at North Eastern Gas, where a comprehensive implementation of a subset of the methodology outlined here has taken place, with M. Higgitt and I.G. Smith the principal planners and Mr. Smith the designer of the system which is accomplished on the ICL PERQ computer.

2. Current practice in data modelling

Current data modelling techniques are oriented around the development of data entity types by the systems analyst for the users of systems (NOT 1). Methodologies do not suggest how entities are established and each Data Subject Area contains about a dozen entities.

Outline models are built in diagrammatic form showing entities and their key attributes and relationships. Much of the work involved is a paper and pencil exercise which suggests that entities are listed prior to investigation, that interviews tend to reinforce the original list of entities and that details of events and operations forming the creation, updating and deleting entities, form the basis for the model.

Some analysts have constructed normalisation tables for optimised 3NF relations. This is an adjunct to the construction of the model (or rather, a secondary proof). The analyst then enters the conceptual level entities, attributes and relationships on the ICL Data Dictionary Systems (DDS) which contains names, descriptions, extents and keys discovered in the process (ICL 79). From these entries, the implementation level schemas, subschemas, areas, records, sets and data items are recorded on the DDS and the IDMS Data Description is generated from it.

As a matter of fact, in most cases, the conceptual level entries are omitted, as the pressure on the analyst is so severe to design the database and the process so lengthy that the model is never completed as written in the methodology. Thus, an imperfect model cannot be completed, let alone a full pure conceptual data model.

The practice indicates that (1) the method is inconsistent and (2) there is a need to reduce the length of time spent in the design process.

3. Inconsistent design

Entity type data modelling techniques are based on an independent development of entity types only very loosely tied to the processes accessing and/or updating the database.

There is need to define entity types in terms of their attributes and roles played by each attribute within the entity type. It is only by defining the entity types in terms of attributes that a data base design can be developed into one that correctly supports all users of the data in the database. The important test of the validity of the entity definition in the database decision is whether the attributes within the entity are correctly described and the interaction of the attributes (with their roles) is flexible enough to handle processing requirements.

This paper defines Role in Section 8 and then defines the types of Role which an Attribute may take within an Entity. It also describes how different Attributes within an Entity are related by the Roles they can play. These Roles connect Attributes within one Entity, giving a definition of a database entity type which is accurately defined.

The Role of each piece of data is closely described in a way that is capable of being described and entered in machine readable form. Thus, because the data description is well defined, the integrity of the data in the database is much better than otherwise.

The data dictionary should take an active part in the determination of the entity types to be maintained within the database. The data dictionary can help with database design process in that the different stages of normalisation of data may be carried out automatically using dictionary/database design software. The determination of entity types to be maintained within a database should not be a mechanical process, but a process whereby the dictionary outlines cases where an entity type is a rather weak entity, as shown in Sections 15 through 21.

A data dictionary should be made up of a pool of attribute types which may be flexibly combined or divided up into entities, as the database designer decides. There should not be a design process wherein entity types are preconceived and subsequently passed through different stages in the design process to be physically transposed from one record type within the dictionary database to another.

This paper sets out a better way of organising the design process such that high integrity databases are designed, because the Role is described within the dictionary, i.e., the purpose and data existence criteria of every Attribute exists and hence the purpose and data existence of the Entity exists in the database.

3.1 Functional analysis omissions

The description of the processing to be done within an organisation in many dictionaries is described in terms of events and operations needing to be carried out when an event occurs internally (or externally, provided it is relevant). To correctly model the processing side of an organisation, there must be a sequence and order dependency applied to the different operations and events (MEY 83).

Some of the current database design technique describe the processing with reference to the database in terms of data flow. While this is a satisfactory means of describing external input and output of data to a process needed and performed within the network of events and operations, it assumes a simplicity seldom found outside textbooks and software purveyors' examples in manuals. The essential idea behind the concept of database is to share data between users and to eliminate unnecessary copying of data, in order to ensure a higher degree of data integrity. Thus, any method relying on data "flowing" is against the fundamental principles of shared database design.

To obtain a complete model of a database, the Attributes must be described in detail so that data validation procedures may be automated and the relationships between entity types may be generated automatically from the description of the data entered by the analyst and confirmed by the database designer.

4. The de-skilling process

Many of the tasks taken in developing the model and designing the database take a far greater length of time for the designer than is realised and costs of the skilled, trained designer far outweigh advantages achieved in absolute perfection of the database. Methodologies have had to be formulated, as the older analysts moved onwards and upwards, and demands for trained designers exceeded supply. The rules of thumb needed to be codified into axioms and theorems put together which permitted a workable solution.

The breaking down of the total task into phases and sub-phases is a constructive step forward, as it permits discrete activities to be performed by analysts who are aspirants to the chair of chief database designer. Thus, the activities of user interviewing, the first recording of Attributes and their definition (within User Entity) and the placement of User-Entities and Attributes, can become jobs for junior analysts as parametric-class entrants to the dictionary.

The dictionary needs to be the holder of the recording of the interview and discovery of User-Entities, Attributes and Roles as quickly as the interview process takes place, in order to become a reliable, although sometimes contradictory, recording place for established facts about the real business. The dictionary cannot wait for the preliminary paper and pencil exercises to take place before information is available to one and all using, enquiring, recording, synthesizing, and planning the data about data.

Accomplishing the paper and pencil exercises (without active use of the dictionary) requires the assimilation of a large number of facts about data items, their keys, relations and strength/weakness in a process known as normalisation. A perfect state of knowledge of forty or fifty data items is required for smaller systems in a state of perfect grace

and requiring instantaneous recall. It is suggested that this perfect situation cannot be attained and, therefore, the process of normalisation is by-passed.

The pressure of accomplishing the completed database design as quickly as possible after the investigation is ever present. Thus, the performance of the designer is handicapped by the shortage of automatic tools stemming from the dictionary. The single act of committing information to the dictionary is not helpful, unless tools are created for use with the dictionary.

It was with this realisation that this current study began and it has been a strong motivating force behind the North Eastern Gas activity.

5. Entity Type

The element Entity type is a specification of a basic unit in an enterprise which can be termed data. There is no limit placed on definition of entity type stated. An Entity is any thing, object or concept (MEY 82).

Each entity type defines a member of a class of objects in the world of an enterprise. Each entity is at one time any thing, object or concept, but, once defined, assumes an exclusivity of existence for that part of the universe within the firm which fits the description.

The semantics of the object's description must be defined properly and its revision both condoned and encouraged. To set the definition too high and all-inclusive is to invite sub-entities and too low is to create super-entites. Experience has proven that every entity is a member of a larger group of entities, sometimes called entity class. This may be carried as a functional area which is roughly the size of an application plus the logical interface to adjacent applications. A functional area is taken as "an area of concern or applicability to the firm."

Experience has also shown that every entity which is a user conceived object, as opposed to a theoretical entity, is acceptable in presentation to the user, management, analysts and the host of business and computer staffs.

There must be several entity types (therefore) (PIN 80a).

The "User Entity" which is the most likely breakdown of the firm in its paperwork; the forms, cards, screens, communications, reports and displays.

The "Pure Entity" is the logically exclusive (normalised) description of a thing, object or concept. As such, it represents a collection of elements in a hierarchical arrangement such that at each level all subordinate elements, or the owner of the next level of hierarchy, are included with it. The essence of a normalisation is (1) in the network approach to show a hierarchical owner-member relationship; or (2) in the relational approach to show the logical pathway from the most major element to the least element for each combination of travel through a hierarchy (MEY 80). It is the least tangible.

The third form is a "Functional Entity," the combination of attributes and values which accompany Event when a State is interrupted. This is the subject of a separate paper, "The Devolution of Fuctional Analysis" (MEY 83). It is tangible to the implementor.

For the purpose of this exercise, the definition of entity is meant in two senses and will be clearly indicated as (1) User-Entity to represent the paperwork, forms, cards, communications, reports and displays of an organisation at present under investigation; and (2) Pure-Entity to reflect the generated entity which emerges from the attribute-role descriptions. When the word "Entity" is used alone, it is applicable in principle to either User-Entity or Pure-Entity.

6. Attribute Type

An Attribute is a characteristic of a given Entity-type (MEY 82).

An Attribute is an internal expression of a characteristic or property of an Entity. The Attribute serves to distinguish an Entity type from another. Each Entity type has a set of Attribute types associated with an occurrence of the Entity.

The dictionary structure holds Entity-types and Attribute-types such that Entity occurrences have Attribute occurrences. An Attribute-type may be mandatory or optional. An occurrence of an Attribute-type may be associated with one or more occurrences of Entity-type. One Entity occurrence may have one or more Attribute occurrences. The natural relation between occurrences of Entity and Attributes is to form a network between Entities and Attributes.

An Attribute is an internal expression of an Entity. The external expression, sometimes termed an implied attribute, is a result of an expression of a network of Roles. The placement of Roles between Attributes implies that the characteristics of one Attribute-type occurrence are valid when in association with another or other Attribute-type occurrence(s).

The properties associated with an Attribute are at least name, description and unit of measure in which it is expressed.

7. Attributes Present Values

All Attribute occurrences have one or more Values. A Value is a finite symbol string which represents some object system value.

The Values associated with Attributes limit the applicability and validity of an Attribute. It permits a range or table of values, depending on the extent and meaning of the Attribute.

All Attributes have Values with specific meanings;

- (1) Minimum - the lowest numeric/alphabetic value permissible with the Attribute.
- (2) Maximum - the highest numeric/alphabetic value permissible with the Attribute.
- (3) Actual - the only numeric/alphabetic Value permissible with the Attribute.
- (4) Initial - the Value to which an Attribute is initialised whenever the occurrence of an Attribute is initiated.
- (5) Return - the Value to which an Attribute changes whenever the occurrence is inoperative or reaches the end of a cycle.

Attributes may have any or all of the above Value associations. A range of series or a series of ranges may be specified, seeming to negate the necessity for minimum and maximum. Actual may form a list and, as such, give rise to tables.

8. Roles Bind Attributes

A Role is a concept relating to defining associations between Attributes (PIN 80b).

Each Attribute is related to other Attributes in a given area of study. This is known as a Role. The Subject Attribute is the Attribute under study. The Object Attribute is a second Attribute studied in association with the Subject Attribute.

The binding of the Subject and Object Attributes are observable and real. The binding varies by the type of Role portrayed. The strength varies in the definition of the Role, as described in Section 9.

9. Roles Defined

The Role types are defined as follows:

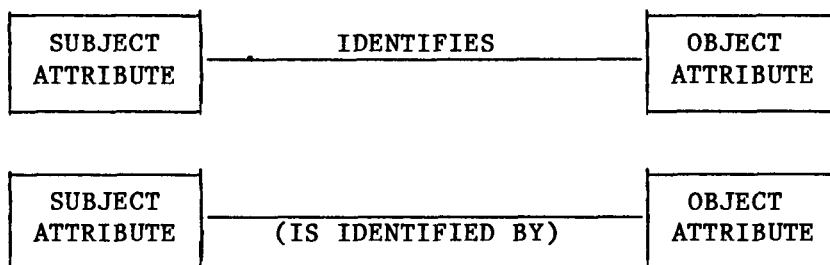
(1) Identification - Single Value (ISV)

A Value occurrence for the Subject Attribute type has one Value occurrence for the Object Attribute type.

The Role is one of identification of the Object Attribute. The Subject Attribute identifies the Object Attribute; for example, Account Number identifies Contract Number. Both are separate Attribute types. A contract can belong to only one account number.

Note that only one occurrence of Contract Number may belong under Account Number. Identification - Single Value (ISV) refers to a single occurrence of an Object Attribute type.

The pattern is shown;



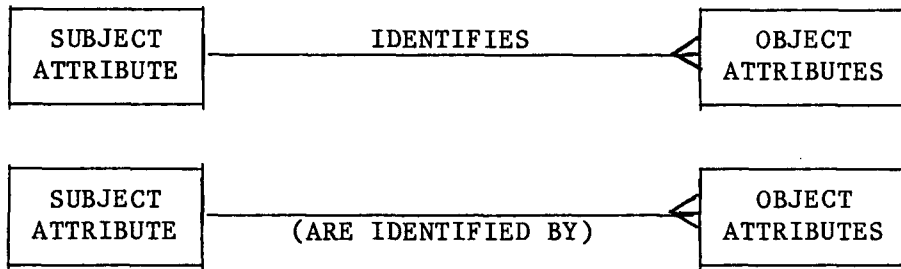
(2) Identification - Multiple Value (IMV)

A single Value occurrence for the Subject Attribute type has many Value Occurrences for the Object Attribute type.

The Role is also one of identification. The Subject Attribute identifies many values within the Object Attribute type; for example, Customer Number identifies Customer Addresses where the customer may be a chain store with many locations. All locations are separate occurrences of an Object Attribute type.

Identification - Multiple Value (IMV) refers to many occurrences of an Object Attribute type. In evaluation of IMV against ISV, it may be better to over estimate the number of occurrences and select IMV.

The pattern is shown:



(3) Descriptive (DES)

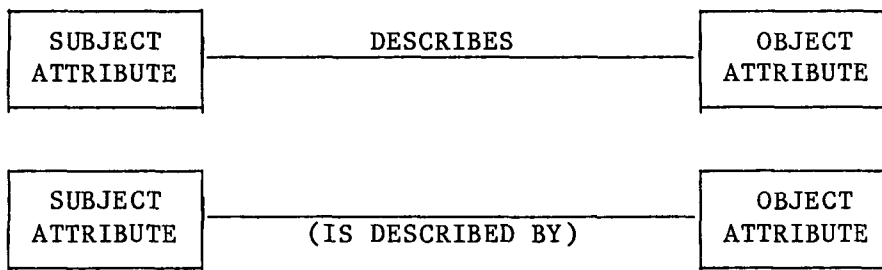
A Value occurrence of the Subject Attribute type provides further information related to the Object Attribute type.

The Role is one of description. This Subject Attribute normally would be text or numeric quantities and would not normally have the role of identification with respect to other Attribute types.

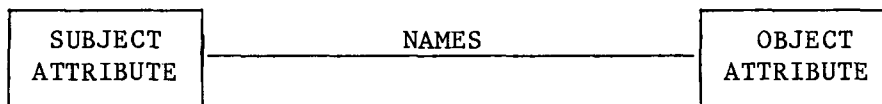
The Subject Attribute identifies the Object Attribute; for example, Customer Name describes textually the Customer Number associated. A Customer Name can only belong with one Customer Number and it is true that a reverse relationship (recognised or not) is an ISV.

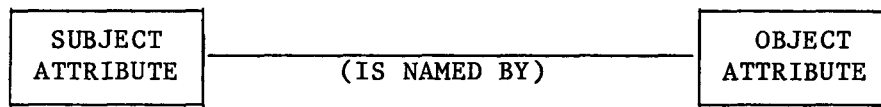
Note that only one occurrence of Customer Name may belong to Customer Number. Description refers to a single occurrence of an Object Attribute type.

The pattern is;



A variation of the pattern exists where the Subject Attribute has the Attribute (itself) of being Alphabetic or Alphanumeric, as follows;





(4) Status (STA)

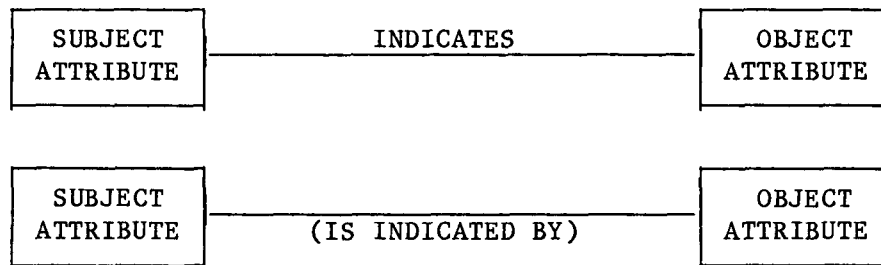
A Value occurrence of the Subject Attribute type is derived from conditional-expressions applied to the Value of one or more Object Attribute type(s).

The Role is one of a semaphore or code based on a conclusion reached by examination of Value(s) of Object Attribute(s); for example, Permanent Employment may be an indicator which applies to all employees with three years' service and for men, over the age of 18, and for women, over the age of 21.

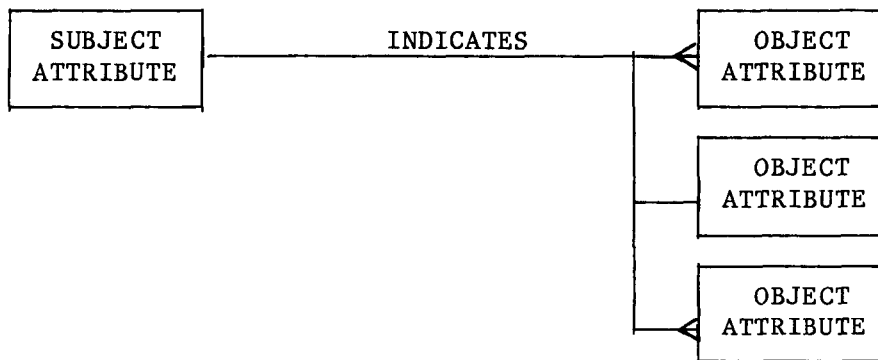
Note that only one occurrence of Regular Employment (status) may be shown and that the answer (true to that of a semaphore) is "yes" or "no".

On the other hand, Employment Class could carry a wider variety of codes to indicate different situations and groupings; P for Permanent; T for Trainee; S for Senior Management and R for Retired. Semaphores are usually a temporary solution and quite often expropriated as Status Codes.

The pattern is;



Note that where more than one Object attribute is determinative, it will be helpful to show;



Note that in all cases the indication of single or multiple dependency should be observed and shown.

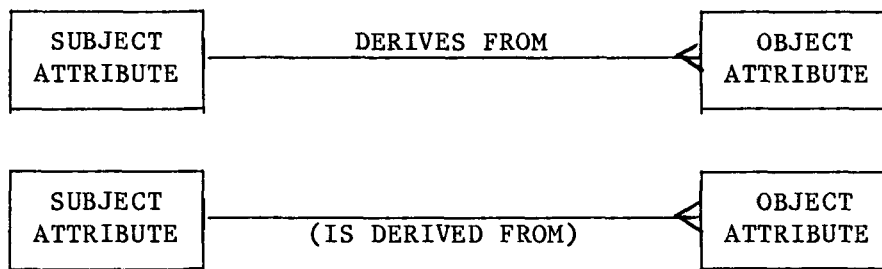
(5) Derived (DER)

A Value occurrence of the Subject Attribute type is determined from the Value occurrence(s) of the Object Attribute type(s).

The Role is one of derivation from Object Attribute type(s); for example, Contract Net Total is derived from Contract Gross Total less Discount. All Attributes are separate and distinct and may be in themselves derived items (or see (6), Aggregated). The simplest form is a derived item which carries the rounded amount to the nearest £1000 of an exact calculation, for reporting purposes.

Note that many occurrence of many Object Attributes may enter into the Value shown as the Subject Attribute.

The pattern is;



The two notes following the pattern of STA will apply equally to DER.

(6) Aggregate (AGG)

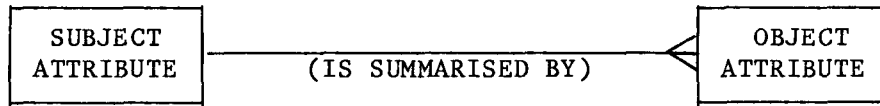
The occurrence of the Subject Attribute type is derived from the Values of multiple occurrences of the Object Attribute type(s).

The Role is one of summarising Object Attribute type(s); for example, Contract Total Gross is a Subject Attribute type representing the total of line items of the Object Attribute type Contract Item Value. Without the existence of the Contract Item Values the Contract Total Gross would exist (probably with an ISV from Contract Number) with a zero value.

Note that many occurrences usually make up the Subject Attribute type Value in Contract Total Gross. It can also refer to several Object Attributes type Values, but there also exists a case for separation of the multiple Object Attribute cases into a new category, Combined (COM), which would permit crossing of account code boundaries, whereas AGG could be reserved for simple totals.

The pattern is;





The two notes following the pattern of STA will apply also to AGG, but consider adoption of COM.

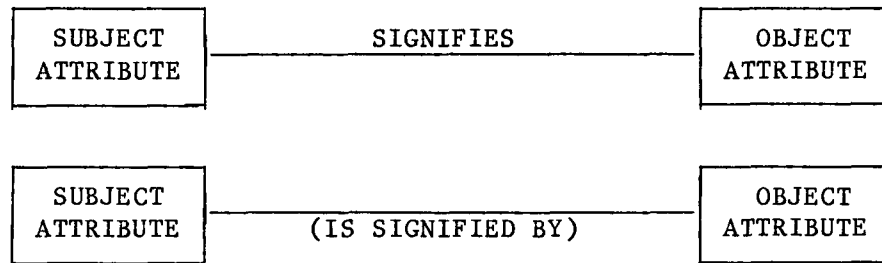
(7) Existence (EXI)

The Value occurrence of a Subject Attribute type indicates the condition of existence for a Value occurrence of the Object Attribute type.

The Role is one of testifying to existence, a pure semaphore; for example, the Object Attribute type Sex determines the Subject Attribute type Maternity Leave.

Note that under current laws that Maternity Leave can be granted only to Female employees. The Existence refers to a single occurrence at the Subject level is determined by a single Value occurrence at the Object level. It is the same as STA for single occurrence, single Attribute Roles.

The pattern is;



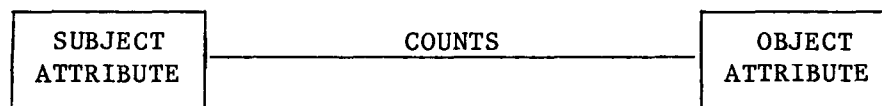
(8) Population (POP)

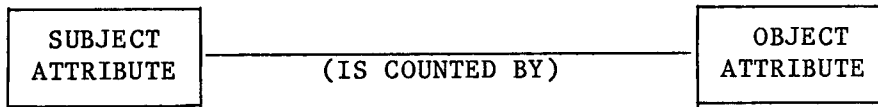
The Value occurrences of the Subject Attribute type specifies the number of Value occurrences of the Object Attribute type.

The Role one is of counting; for example, Title as a Subject Attribute type will count the number of each category with the Object Attribute type. With Mr., Mrs., Miss and Dr. as Titles in Subject position and the Employee Name in Object position will yield a count of each category in the Subject Attribute type Value for the four occurrences.

Note that any number of occurrences of the Object are recorded in the Value of the occurrences for the Subject.

The pattern is;





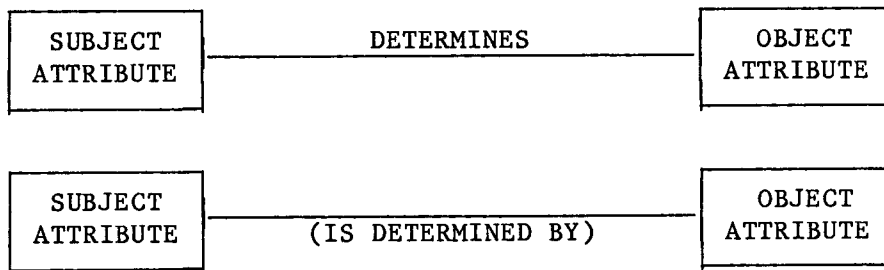
(9) Content (CON)

The Value occurrence of the Subject Attribute type determines the range of values which the Value occurrence of the Object Attribute type may take.

The Role is one of limitation; for example, Job Title as a Subject Attribute type will limit the range of the Object type Salary.

Note that any number of Values or a range of Values may appear in the Value for of the occurrence for the Subject.

The pattern is;



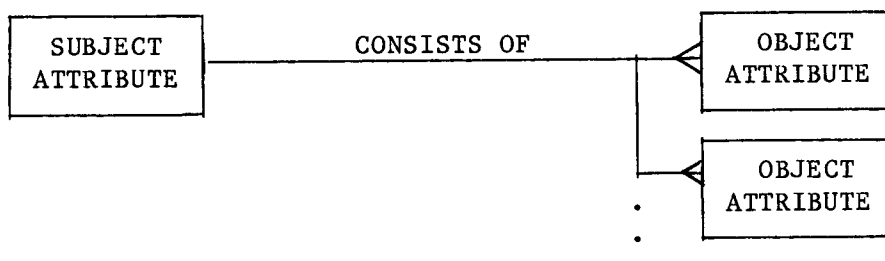
(10) Logical Group (GRP)

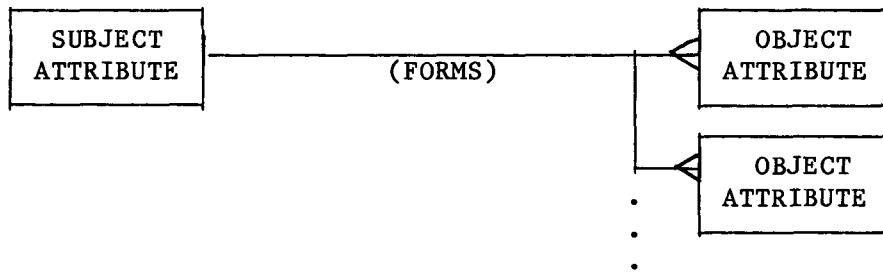
The Value occurrence of the Subject Attribute type consists of the Values associated with the Value occurrence of the Object Attribute types which it comprises.

The Role is one of grouping; for example, Date as a Subject Attribute type consists of Month, Day and Year as an Object Attribute types. Month, Day and Year are separate Attributes. Other Attributes, day of week, century, or similar material might also be included.

Note that the Value occurrence of each Attribute may appear once for each occurrence for the Subject. The Subject Attribute can be logically mapped, but may not physically be represented.

The pattern is;





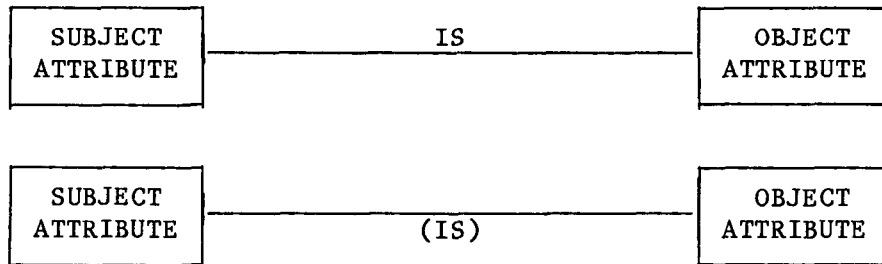
(11) Synonyms (SYN)

The name associated with the Subject Attribute type has a different name than that associated with the Object Attribute type, but the description, meaning and Value occurrences are identical.

The Role is one of equivalence. The Subject Attribute has a meaning to one group within the firm under one name and another name to another group. A number of classical examples come to mind; Spanner versus Wrench; Supplier Address versus Customer Address; Inventory quantity versus Stock Balance.

Note that the synonym of the Subject is the Object and also that the reverse is true. Pairs of entries ought to be entered.

The pattern is shown;



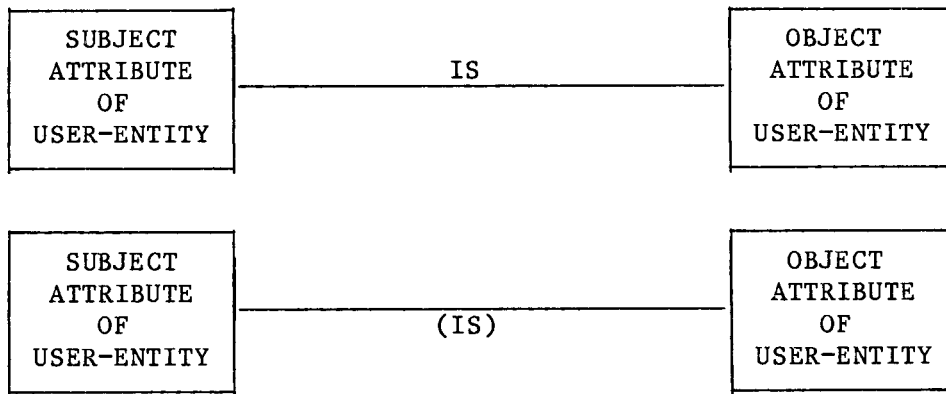
The hierarchical plane is not raised or lowered by the use of a SYN Role. There are possibly two means of raising or lowering hierarchy;

- (1) Use of an ISV or IMV to indicate strength; or
- (2) Use of a strength parameter as described in Section 9.

The surviving name within a system overall plan is difficult to reconcile and, unless context is shown by connection from a functional area (and then assuming only one name from each area), another name must be contrived.

The treatment is best served by reference to User-Entity by means of qualifiers, as this comes closer to the point of reference for reconciliation within the systems gathering phase.

Therefore, the better diagram is;



(12) Homonym (HOM)

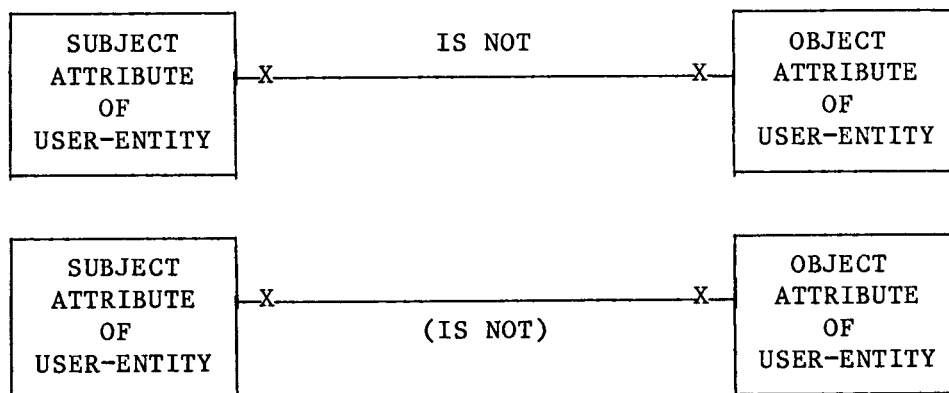
The name associated with the Subject Attribute was two separate and distinct meanings within the firm with different descriptions, meanings and Value occurrences, such that it necessitates a separate and distinct identity from the Object Attribute from which it was created.

The Role is one of distinction. The Subject Attribute has a meaning to one group with a set description, meaning and range of values. The Object Attribute has a meaning to another group with a different description, meaning and range of Values.

Note that the Subject Attribute is such, when considered, only as subject and not object, because the Object Attribute is permanently recorded. Pairs of entries ought to be entered.

Distinction of meanings is reconciled within the hierachies with the same tool as with SYN, a qualifier.

The pattern is shown;



10. Judgements in Design

In working with the Attributes and Roles and with the paperwork generation of Relational and Network distributions prior to the synthesis of databases, some additional inputs were thought to be necessary (1) to force one solution above another, and (2) prohibit and break circular chains, as in bill-of-materials.

The Role types added to force design considerations are defined as follows;

(13) Top of Pyramid (TOP)

A judgement has been made by the Analyst which declares that the Subject Attribute is superior to the Object Attribute in establishment of a hierarchy of data.

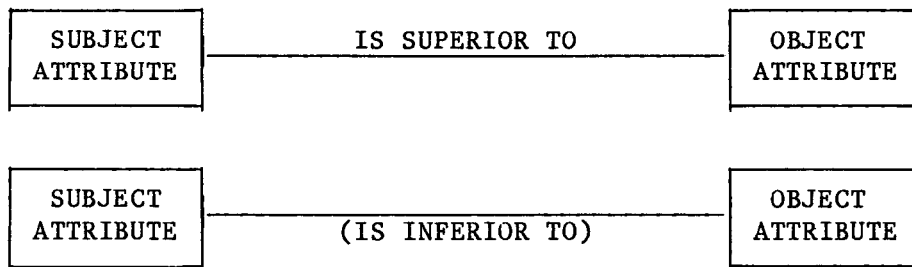
The Role declares that this Attribute is superior to every other Attribute of equal hierarchy. In this case, ISV or IMV Roles are directed at some point in the hierarchical chain lead to a commonly owned Attribute. This is not to imply the degree of complexity of either one chain or the other or its length. As a matter of fact, the usual fact is that the Subject Attribute has a less complex path, almost to create a by-pass of intermediate levels, through which the chain passes to the Object Attribute.

Note that only one occurrence of a TOP can appear within a Functional Area. If more than one is used, the purpose of the TOP is defeated and both the Subject Attributes or the several nominated Subject Attributes will be positioned as though the TOP Roles were ISV or IMV Roles.

Note also that of an ISV or IMV Role points toward the Subject Attribute the effect is null and void. An error should accompany the entry and should be rejected.

It is thought that one TOP Role might be substituted for another in order to understand or plot alternative normalisations.

The pattern is shown:



(14) Non-Circular (NOC)

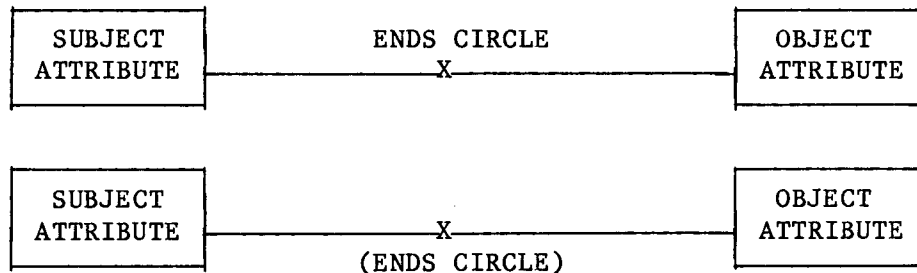
A judgement has been made by the Analyst which declares that the Subject Attribute has a valid Role in respect to the Object Attribute, but because of the circularity of Roles between Attributes, the Role or Roles are to be considered broken (or not considered).

The Role declares that this Attribute has at least one Role to the same Object Attribute, but the Role or Roles declared are taken out of consideration in the normalisation. Where the recursion occurs between two items, a bill-of-materials case, is identified by the NOC. Where the distance in the chain is further removed, the break in the structure will guarantee the creation of a non-recursive Pure Entity or the Relation chain will be split.

Note that the use of two or more of these Roles will effectively break the chain of Attributes and Roles, but that a more clearly descriptive Role exists in BRK (see below).

In this Role, the use of pairs of Roles will have the same effect as one.

The pattern is shown;



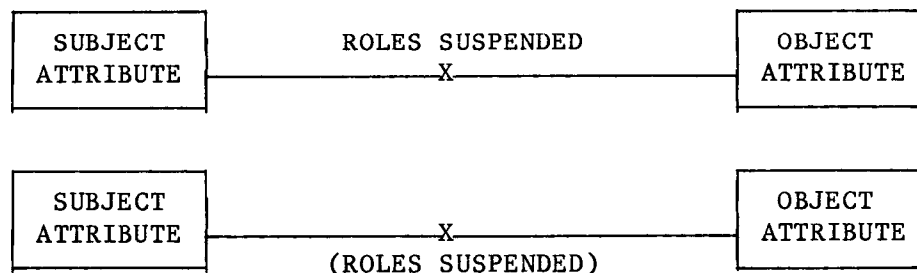
(15) Broken (BRK)

A judgement has been made by the Analyst which declares that the Subject Attribute, even though it has a valid Role or Roles in respect to the Object Attribute, should be separated from the Object Attribute and considered not to have had a Role between Subject and Object Attributes.

The Role declares that this Attribute has at least one Role to the same Object Attribute, but the Role or Roles declared are taken out of consideration in the normalisation. The break in the structure will guarantee the creation of two separate Pure Entities unless circular logic appears. The Relational chain will not include the Object Attribute and subordinate Attributes not joined by common Roles further down in the chain of Attributes and Roles.

Note that the use of two or more BRK Roles duplicates the use of the NOC Role. The BRK Role should be entered with caution, as it may be operative in one direction and not in the other. For an absolute break in the system, a pair of BRK Roles should be shown exchanging Subject and Object Attributes.

The pattern is shown;



11. A note on Charting/Text

Modified from the conventions of the DDSWP in the examples of patterns, the straight left-to-right (and top-to-bottom) placement of text is shown.



This means that the "Subject Attribute identifies the Object Attribute", in the case of the ISV, Identification-Single Value.

Sometimes it is more meaningful to show the textual representation with the Object Attribute more prominent. Thus the second chart;



This means that the "Object Attribute is identified by the Subject Attribute."

Whenever an Object Attribute is to the left of the Subject Attribute or above it, square brackets are shown around the text.



This also means that the "Subject Attribute identifies the Object Attribute."

Similarly, an Object Attribute is to the left of the Subject Attribute or above it and the text reads more meaningfully from left-to-right and with the Object Attribute more prominent, a combination of square brackets and parenthesis are used.



To summarise, the parenthesis indicates that the text accompanying the Role reads with the Object Attribute as the most prominent in order. The square brackets indicate that is Subject Attribute is to the right of or above the Object Attribute.

12. Holding Values with Attributes

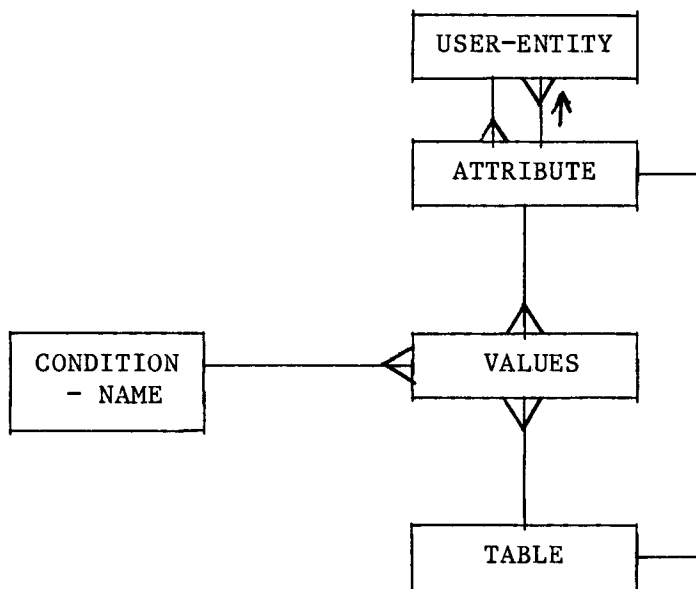
The Attributes and Values stored in a dictionary have a clear relationship and are simplified in a network database. Values may form single or multiple values, ranges or their opposites. For every Value provision, a non-value is predisposed to not be valid. Similarly for every non-value, an exclusion, a predisposed Value exists as a valid Value.

The contents of the Values held should define exact meanings to the Values; minimum, maximum, actual, initial or return value. Where ranges are prescribed, an indication should be shown. Whether the Value is valid or invalid should be indicated.

Similarly, where Values with Attributes convey a meaning, a condition-name should form a separate recording with a relationship to the Value (or Values) associated.

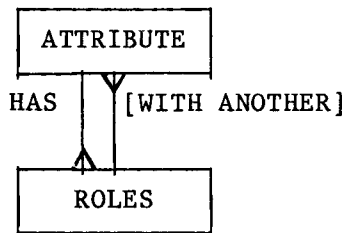
In dynamic workings or in cases where by-product lists can be produced (as in Product Code books), a Value Table is necessary. The contents may include arguments, relations and arithmetic-expressions (with formulas) and be ascending or descending in the key.

The several elements are shown as follows;



13. Holding Roles with Attributes

The network representation of Roles with Attributes in a dictionary have a simpler structure reflecting the clear relationship between two Attributes.



In this diagram, any Attribute has a Role with another Attribute. The relationship of Attribute to Roles to 1-to-many. Every Role has an at least one Object Attribute.

The contents of Role are as follows;

- (1) Type of Role;
- (2) A description of the Role;
- (3) Strength of Role

There is also an identifiable role-name.

A further clarification and customisation of Role could be made by permitting the acceptance of a short textual description (say, 16 characters) to be substituted for the standard textual Role Descriptions (of Section 11 referring to sections 8 and 10 for the text). These should include the appropriate parenthesis, where applicable.

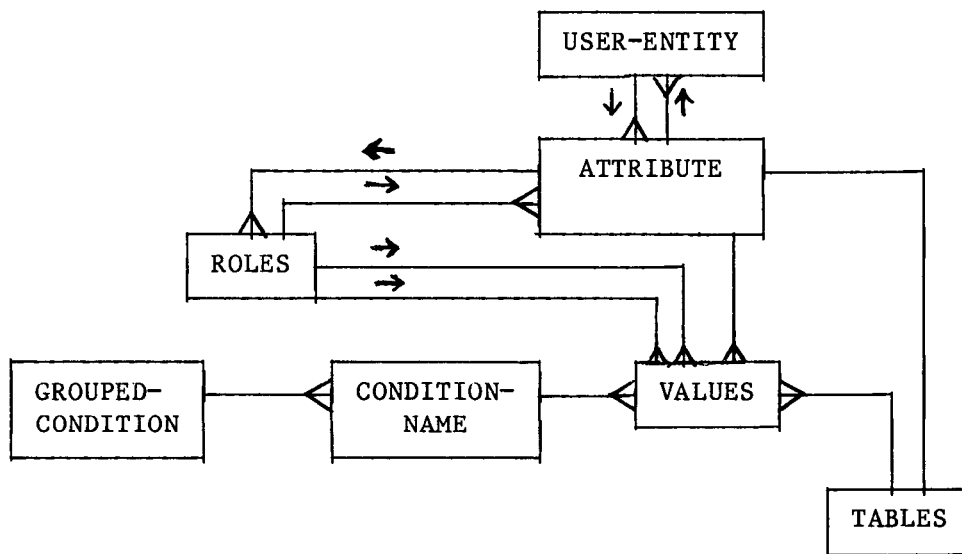
Some Roles are applicable to only when certain Values are present and it is thought necessary to relate the occurrence of Roles to Values directly.

Since the Values can apply to the Subject and Object Attributes, two relationships are drawn.

Furthermore, the presence of specific Values often convey a condition and this condition may be expressed best with a conditional-name. The establishment of this condition can be tested in the Logical Consequence paths emerging when an Event disturbs a State and yields a series of other Events and Operations.

A further modification which permits a macro reference to condition-names is useful for development of retrieval situations where users require constant report-type information. This is called a Grouped-Condition.

With the Attribute-Value complex, the resulting diagram emerges;



14. Pre-normalisation Structure

The major structure formed from User-Entity, Attribute, Roles, Values, condition-name and Tables shown above (in Section 13) presents all of the necessary information for structuring Pure-Entities and Relationships for the purpose of this paper. This structure, the pre-Normalisation structure contains more information than is required, because of the need to document material for a data dictionary in an over abundant style (to convey to users the meaning, definitions and parts of a system as they have described it) and the need to justify data bases with structural details.

The simpler structure involves only Attributes and Roles (only) provided the attribute population can be reduced by functional area definitions directly or via User-Entities.

15. A Practical Problem Described

The pilot was chosen from the ICL Data Dictionary Course handouts (1979) T2408 and T2409 dealing with Television repairs. It is re-phrased here. Note that a similar exposition is recorded in Iosson's paper (IOS 84).

Two User-Entities are thought to exist and each has Attributes, as follows:

User-Entity	Attribute	
	Abbrev.	Name
Repairman	EN	Employee-Number
	RN	Repairman-Name
	VR	Van-Registration-Number
	VM	Van-Mileage
	MV	Make of Van
Visit	SD	Service-Date
	CNO	Customer-Number
	CNA	Customer-Name
	CA	Customer-Address
	SN	Set-Serial-Number
	ST	Set-Type
	H	Hours

The accompanying material draws the assumptions that;

1. A visit is regarded as being to a set rather than a customer.
2. Only one set is dealt with at a customer's premises on one visit.
3. A customer's premises will only be visited once on a given date.
4. Set-Serial-Number is unique across all TV sets.
5. Each repairman has his own van.

16. Developing Roles

The Roles become clear when examining the Attributes and many be listed as follows;

Subject Attribute (Abbrev. in Section 15)	Role (Abbrev. in Section 9 or 10)	Object Attribute (Abbrev. in Section 15)	Role Description
EN	ISV	RN	identifies
EN	ISV	VR	has or uses
RN	DES	EN	describes
VR	ISV	VM	has
VR	ISV	MV	has
SN	ISV	CNO	identifies
CNO	ISV	CNA	identifies
CNA	ISV	CA	identifies
CNA	DES	CNO	describes
CA	IMV (W)	CNA	identifies
CNO	IMV (W)	SN	has
SN	ISV	ST	has
MV	IMV (W)	VR	identifies
SD	ISV	H	has
EN	ISV	SN	visits
SN	ISV	SD	(is visited on)
EN	ISV	SD	visits on

As developed, it becomes apparent that the fixed descriptions do not always fit and that ISV is often substituted as has, uses, possesses or owns. In re-reading the definition, the phrase "identifies" derives from the word "has" in the definition.

The weak role which was developed in the analysis refers to those cases which are not always true or are irrelevant in context.

For example, the Customer-Number identifies the possession of more than one Television set having a Set-Number. Similarly, a Make-of-Van identifies (visually) Van-Registration-Number (or at least increases the probability of making a correct decision of finding a vehicle in a parking lot).

17. Classifying Attributes

A short cut is available to the Classification of Attributes to create a hierarchy. Clearly four categories of Roles exist which identify the top or, on the other hand, the erroneous (or incomplete) Attribute populations.

- (1) Superior includes all Attributes marked with TOP in respect to an Object Attribute; or those Attributes which have ISV, IMV or GRP Roles to Object Attributes, but there are no ISV, IMV or GRP Roles in which the Attribute is an Object Attribute. The Attribute in question "owns" other Attributes and is not "owned" by another Attribute, if you prefer.
- (2) Intermediate includes all Attributes which have any Superior Role (all Roles except DES) to another Attribute, but have an inferior Role as an Object Attribute in respect to another Attribute. The Attribute in question "owns" other Attributes and is "owned" by another Attribute, if you prefer.

- (3) Bottom includes all Attributes which have no Superior Role to another Attribute and have only an inferior Role as an Object Attribute. The Roles most common will be a Subject Attribute DES (the only inferior Role) or an any other Object Role. In other words, the Attribute in question is "owned" by another Attribute, but never "owns" another Attribute.
- (4) Null indicates that the Attribute is not in a Superior or Inferior Role to any other Attribute. This indicates that the analysis of data has not proceeded far enough to be conclusive. All Null items may be discarded pending investigation.

The classification can be made as Attributes and their Roles are entered on the dictionary, as in the following precedence table;

Attribute exists (with Roles) on dictionary?	Object's Role Category Previously	Role Studies		Resultant Object's Role Category
		ISV IMV GRP TOP	DES	
No	None	No	Yes	3
No	None	Yes	No	1
No	None	No	No	3
Yes	3	Yes	No	2
Yes	2	Yes	No	2
Yes	1	Yes	No	2
Yes	3	No	Yes	Note 1
Yes	2	No	Yes	2
Yes	1	No	Yes	Error
Yes	3	No	No	2
Yes	2	No	No	2
Yes	1	No	No	2

The categories can be expressed, it is suggested, as sorted system owned sets in a network data base in the DDS. This method will identify possible candidates for the top of the hierarchy, the most superior.

The BRK is not a function at this time. The NOC causes a bottom entry for the Subject Attribute. HOM will cause two separate Attribute entries and therefore requires User-Entity.

It is useful for charting purposes also to accumulate the number of times an Attribute is a subject or object. Note that weak roles are not tabulated and that DES roles are also not tabulated, because the end purpose of counting the number of times Subject Object is to form a ranking system which will determine which Attributes are to be charted before others.

As it turns out, only those Attributes which have greater than one Object Attribute in a Role are of interest. These require some logic which will allow their placement next to both or all items which have an ownership Subject Role to them. More recent reading has suggested that a small PROLOG program would accomplish the same purpose in a cleaner solution with a larger base.

The results of the operation to assign Role-Categories is shown below.
 A tabulation of the results is shown in Section 17.1.

Subject Attribute (Abbrev. in Sec. 14)	Role (Abbrev. in Sec. 8 or 10)	Object Attribute (Abbrev. in Sec. 14)	Order of Entry	Role-Category		Number of Times	
				Before Entry	After Entry	Subject	Object
EN	ISV			-	1	1	0
		RN	1	-	3	0	1
EN	ISV			1	1	2	0
		VR	2	-	3	0	1
RN	DES			3	3	0	1
		EN	3	1	1	2	0
VR	ISV			3	2	1	1
		VM	4	-	3	0	1
VR	ISV			2	2	2	1
		MV	5	-	3	0	1
SN	ISV			-	1	1	0
		CNO	6	-	3	0	1
CNO	ISV			3	2	1	1
		CNA	7	-	3	0	1
CNA	ISV			3	2	1	1
		CA	8	-	3	0	1
CNA	DES			2	2	1	1
		CNO	9	2	2	1	1
CA	IMV (W)			3	3	0	1
		CNA	10	2	2	1	1
CNO	IMV (W)			2	2	1	1
		SN	11	1	2	1	0
SN	ISV			2	2	2	0
		ST	12	-	3	0	1
MV	IMV (W)			3	3	0	1
		VR	13	2	2	2	1
SD	ISV			-	1	1	0
		H	14	-	3	0	1
EN	ISV			1	1	3	0
		SN	15	2	2	2	1
SN	ISV			2	2	3	1
		SD	16	2	2	1	1
EN	ISV			1	1	4	0
		SD	17	2	2	1	2

17.1 Tabulated Results

Attribute Abbreviation	Role Category	Number of Times	
		Subject	Object
EN	1	4	0
RN	3	0	1
VR	2	2	1
VM	3	0	1
MV	3	0	1
SD	2	1	2
CNO	2	1	1
CNA	2	1	1
CA	3	0	1
SN	2	3	1
ST	3	0	1
H	3	0	1

The pivotal Attributes are selected, so that the sequence of handling ensures that the problem cases are handled before any others.

Role Category	Object Attributes > 1	Subject Attributes > 1	Attribute
1	0	1	EN
2	1	0	SD
2	0	1	SN
2	0	1	VR
2	0	0	CNA
2	0	0	CNO
3	0	0	CA
3	0	0	H
3	0	0	MV
3	0	0	RN
3	0	0	ST
3	0	0	VM

(Yes = 1, No = 0)

18. Using the Classification of Attributes

Having set the precedence for the handling of Attributes in charting recalls the difficulties in discovering a means for their ranking. One numerically valued hierarchy was tried with numbers assigned to each attribute and raised with reference to it as a Subject and lowered as an Object. It was discovered that whatever numbers were assigned a chart could not be drawn without crossing lines. Nor was the solution any closer to defining pure entities and relationships. Nor were the Attributes coded or arranged to permit normalisation.

The clue is in the numerical categories at hand. The highest ranking of all the Attributes is a Role Category 1 always, only one Role Category 1 should exist by the rules stated in Section 17 and applied to the Television Repair problem. If there is more than one Role-Category 1, there will be two separate structures connected in the middle of Role-Category 2. A TOP Role will arbitrate and the two tops may be tried in turn.

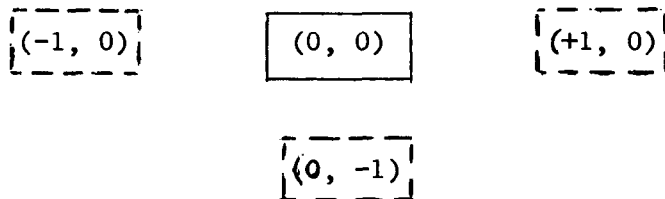
19. Charting Attributes and Roles

The first output from the charting phase should be a list of Attributes with (x, y) co-ordinates as a separate and independent file. The details needed include Attribute-name and, in case of homonyms or synonyms, the User-entity.

The second independent output from the charting phrase should be a list of connections showing the (x, y) co-ordinates of the Subject Attribute and the Object Attribute. Attribute-names should be carried, the Role designation and the bridging text for the Role.

The top of the hierarchy should be placed in the centre of the top of the logical page. Conceptually the page is divided along a row (the x-axis) and a column (the y-axis). The top of the hierarchy is placed at (0,0).

The next box to the right on this row is known as (+1, 0) and the next box to its left is known as (-1, 0). The box immediately underneath is (0, -1).



The EN box is the top of the hierarchy, as it is the first of the sorted Tabulated Results (by Role-Category, Object Attributes > 1 indicator, Subject Attributes > 1 indicator and Attribute-name). It is placed in (0, 0).

Go through the Role records connected to the Attribute. There are 4 Objects to be connected to the Subject Attribute. A simple sort of Subject Attribute/Object Attribute will permit examination in a short table.

The first object Attribute in alphabetical sequence for EN is RN and this is assigned a position to the right of EN and down one row, under the rule that an Object inferior to the Subject never appears on the same line, in the same row. The placement by column is derived by adding number of Objects divided by 2 to the starting column position. Therefore, RN is placed at (+2, -1). The notations of Attribute assignments can best be kept in a two dimensional array.

SD is the second and, working clockwise it receives a position at (+1, -1). SN is the third and receives a position (0, -1). The fourth, VR, receives (-1, -1).

At this stage, having completed inspection of all Roles for CN and, being satisfied that no more relations than the size of the diagram (hypothetically) will permit, the placements are complete.

Inspect the next entry in Sorted Tabulated Results, SD. Determine the second Subject which is to determine placement. In this case, the second Subject is SN which is adjacent. Had it not been, places would have been exchanged to find an adjacent position.

SD has one subordinate Object Attribute, H. This is placed directly below SD at (+1, -2).

SN is the next entry in the Sorted Tabulated Results. A note is made of the connection to SD. The two subordinate Objects are placed to the right of SN, but this space is occupied by H and one additional row is subtracted. CNO is placed at (+1, -3) and ST IS placed at (0, -3).

VR is the following entry in the Sorted Tabulated Results. Being to the left of the principal Role Category 1, its placement is downward or to the left. MV is placed at (-1, -2) and VM is placed at (-2, -2).

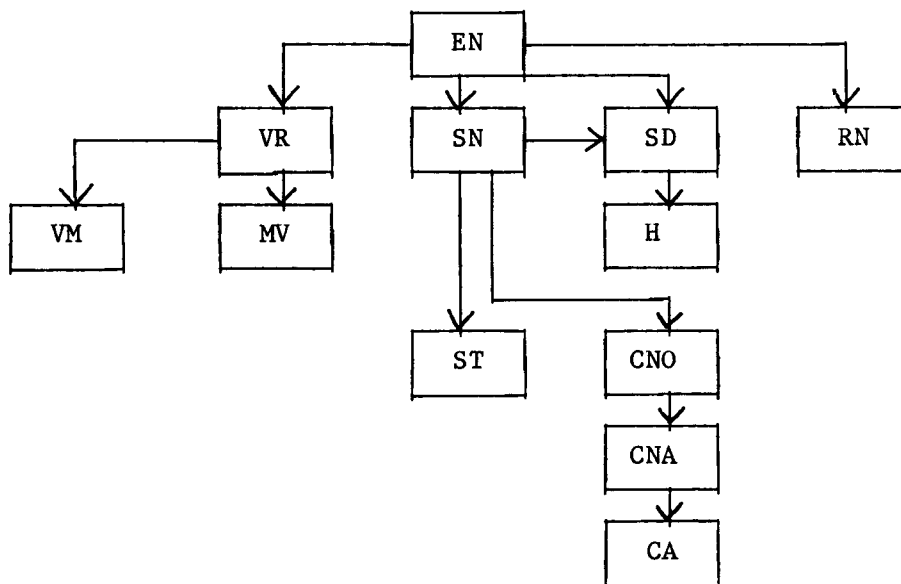
CNA is the next entry in the Sorted Tabulated Results. It is not placed and therefore is skipped, as it is developed as a Role Category 2 and appears in the next position only because of an alphabetical sort.

CNO is placed as an Object of SN. CNO has one Object, CNA, which is placed directly below it at (-1, -4).

CNA may now be processed and has one Object, CA, which is placed directly below it at (-1, -5).

As the following entries are successively processed no additional Attributes are placed. A simple counter of Attributes can reduce the processing involved.

The final placement is shown below.



19.1 Writing away Roles

At the same time as the placement is taking place, the Roles may be charted and written away when the placement is completed. The placement of EN, RN, SD and VR gives all the information necessary for writing away the second output, a list of connections.

The EN ISV RN placement is recorded as;

Employee-Number	Subject Attribute
0	X co-ordinate
0	Y co-ordinate
Identifies	Bridging Text
Employee-Name	Object Attribute
+ 1	X co-ordinate
- 1	Y co-ordinate

A line may be drawn and text within the box and the line text may be printed.

19.2 Acknowledging Weak Roles

After the placement has taken place, a final check is made with the Role processing output described in Section 18. Weak Roles and DES Roles do not determine placement. DES is a duplicate entry facing in the opposite direction from an ISV or IMV.

This adds the following connections;

Subject	Role	Object	Entry
RN	DES	EN	3
CNA	DES	CNO	9
CA	IMV (W)	CNA	10
CNO	IMV (W)	SN	11
MV	IMW (W)	VR	13

Where lines are not shown, are used.

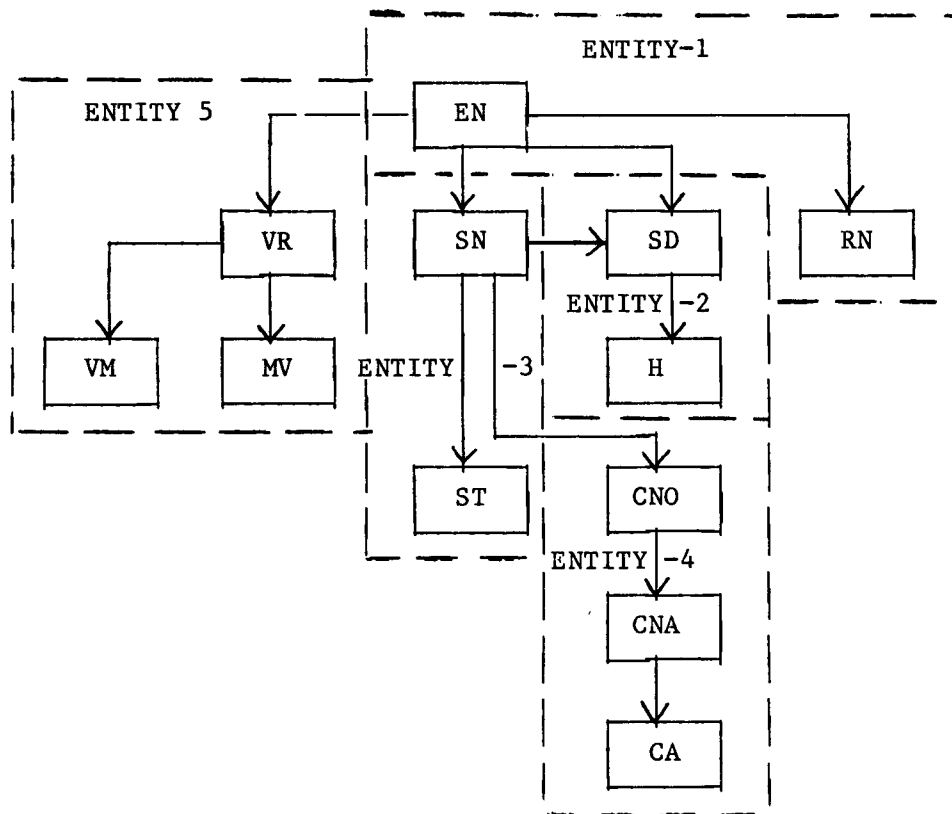
20. Generating Pure-Entities

The rules for creating Pure-Entities from Attributes connected with Roles may be applied during the charting phase or as a separate step. There is probably greater evidence on the side of deferral, because some iteration may be required during initial stages (particularly in multiple Object Role Category 2 items).

The rules for Role-Category 1 Subject Attributes are not complex in generating Pure-Entities. Read through the Role records. If the Role-Category of the Object-Attribute is 3, include the Object-Attribute within the same Pure-Entity as the Role-Category 1 Subject-Attribute.

The contents of the first Pure-Entity cannot be determined until all of the Roles which are attached to the Subject Attribute have been examined. If the Role-Category is 2 (or a Role-Category 1 which is the Object Attribute of a TOP), a separate Pure-Entity is formed around it.

Thus, RN is included with EN, because RN is a Role-Category 3 Object Attribute, but no other Attributes are included. This is Entity-1.



The rules for Role-Category 2 Subject Attributes are not complex in generating Pure-Entities and are the same as those for Role-Category 1 Subject Attributes.

There is one exception. If the Object Attributes are Role-Categories 2 and 3 and all the Role-Category 2 items have the same Role-Category 3 items, without other Subject Attributes of Role-Category 2 related to them, all the Attributes may be included in the same Pure-Entity.

SD is the first of the Role-Category 2 Object Attributes which is an Object of more than one Subject. It has one Object Attribute, H, and the two Attributes are included in Entity-2.

SN is the first of Role-Category 2 items which refer to Object Attributes which refer to Object Attributes SD and CNO also in Role-Category 2. This prevents their inclusion with SN in the same entity. ST is a Role-Category 3 and is included with SN in Entity-3.

The second exception is that if a Role-Category 2 Subject refers to a single Role-Category 2 which further refers to a single Role-Category 3,

the Pure Entity may include all Attributes. CNO, CNA and CA are included in Entity-4.

VR is the third of the Role-Category 2 Objects of EN. VR has two Objects, MV and VM, in Role-Category 3. The three are part of Entity-5.

21. Generating Relationships

The creation of Pure-Relationships from Attributes connected with Roles may also be applied during the charting phase or in the same step as the generation of Pure-Entities. A note appears in Section 20.

The contents of the first Pure-Entity, consisting of the principal Role-Category 1 and any attached Role-Category 3 items, is Entity-1.

The contents of the remaining Entities, (Entity-2, Entity-3 and Entity-5) are determined as consisting of the subject Role-Category 2 item and any attached Role-Category-3 items.

A Relationship exists whenever a Subject Attribute in Role-Category 1 or 2 has a Role with an Object in Role-Category 2. Therefore, three Relationships may be drawn from examination of the first Role-Category 1;

```

Entity-1 Relationship-1 Entity-2
Entity-1 Relationship-2 Entity-3
Entity-1 Relationship-3 Entity-5
  
```

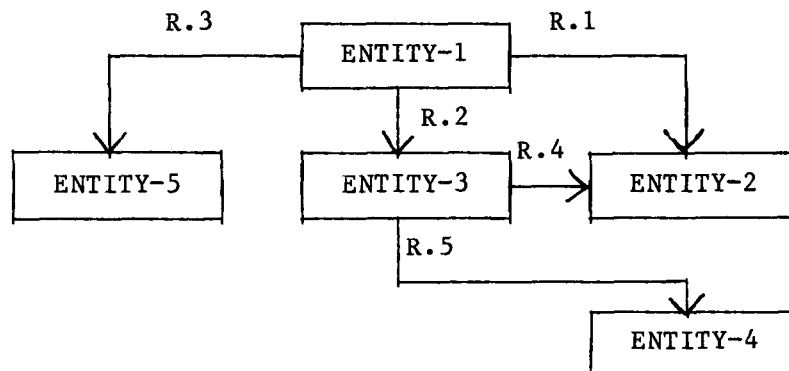
From examination of the Entity-3 constituents, SN is the Subject Attribute of SD, the principal Object in Role-Category 2 within Entity-2, and to CNO, the principal Object in Role-Category 2 for Entity-4. Therefore, two Relationships may be concluded from examination of a Role-Category 2 item which has Objects other than Role-Category 3 items;

```

Entity-3 Relationship-4 Entity-2
Entity-3 Relationship-5 Entity-4
  
```

It was also necessary to examine CNO and its Objects to determine the extent of Entity-4. Had additional Objects as Role-Category 2 or 3 items been attached to CNO, additional Entities and Relationships would have been necessary.

The Entity-Relationship chart derives from Attribute-Role chart. It can be drawn with the x, y axes shown for Attribute-Role.



21.1 Writing Away Entities and Relationships

The Entities generated above are written away, when generated, including the information shown below.

The Entity-1 generation is shown as follows;

Entity-1	Entity
0	X co-ordinate
0	Y co-ordinate
Employee-Number	Key
1	Number of other
	Attributes
	(variable)
Employee-Name	Other Attributes

The Relationships generated above are written away, when generated, including the information shown below.

The Entity-1 Relationship to Entity-2 is shown as follows;

Relationship-1	Relationship
Entity-1	Entity (Subject)
0	X co-ordinate
0	Y co-ordinate
Entity-2	Entity (Object)
+1	X co-ordinate
-1	Y co-ordinate

22. Naming Entities and Relationships

The Entities and Relationships generated from Attributes and Roles need to have meaningful names associated with them for future reference and in documentation of the database. Several techniques may be suggested.

(1) Entity by name dropping

In this method, a short routine is written to compare names of Attribute included within the Entity. Names which are dissimilar in the full-name of the Attribute are discarded.

```

Employee-number      )
Employee-name        ) Employee

Customer-Number      )
Customer-Name        ) Customer
Customer-Address     )

Van-Registration-Number )
Van-Mileage           ) Van
Make-of-Van          )

Set-Serial-Number    )
Set-type              ) Set
  
```

(2) Entity by combination

A list of standard abbreviations can be established for use in a company, instead of full names. Precedence in naming can be established to use further detailed and specific abbreviations to the right and successively more general subdivisions to the left.

These standard abbreviations can be used to name Attributes and concatenated with other abbreviations used in the same way to form Entity-names.

Service-Date	SERV-DATE
Hours	HRS

forms a concatenated name for Entity-2;

SERV-DTE-HRS

(3) Analyst changes

Whether one of the two methods used above or a hybrid or an arbitrary name like Entity-1 is assigned, provision is needed for changing the name of the Entity within the system to satisfy external, duplicative or aesthetic sensibilities (of the Analyst). However, the Pure-Entity is not complete until it is defined completely, as suggested in Section 10, as an occurrence of Entity-type. A description of each Pure-Entity is a basic requirement.

In this respect, provisions should be made for adding a textual description and for changing the name of Entity.

Using the above methods, more likely a concatenation of names with intervening hyphen, are applicable to Relationships.

23. Network Uses of Generated Entities and Relationships

The generation of Entities and Relationships from Attributes and Roles affords a greater flexibility to the design process. Roles are defined to meet the specific needs of the Data Analyst in that all relations between Attributes appear as Roles and defined after intensive study of applications areas in several industries.

The generated Entities and Relationships are created with an automatic process, yielding comparable results with Third Normal Form. For comparable results with Third Normal Form. For comparable results, the process enables the Analyst to consider a greater number of Attributes in more Roles than is possible with normalisation techniques rendered manually. This consideration was primary in the design of the automated process, as it is believed that an Analyst cannot deal with more than a dozen items at one time and more than forty altogether.

23.1 Network Mapping

Following the common practice in ICL Data Dictionary usage (to DDS 650), the mapping between Conceptual and Implementation Quadrants; namely, Entity to Record, Attribute to data item and Relationship to Set permits direct use in network-type structures. The actual naming, description in implementation use and the arrangement of the nature of the Set for

Order, Mandatory/Optional and pointers remain the province of the Analyst.

The Keys are developed from the Subject Attribute of each Role and can be carried over between the Conceptual and Implementation Quadrants for each one of the Generated Entities.

24. Conclusions

This paper has dealt with the feasibility of automatically generating entities and relationships from a series of entries dealing with attributes and their roles with other attributes. This has been a paper and pencil exercise, while developments in actual implementation have proceeded at North Eastern Gas.

The technique satisfactorily provides for early transcription of interview forms and important information regarding Attributes, Roles and User-Entities which are vital to the documentation of the system. If the techniques are followed, these elements of the data dictionary should contain named occurrences of the elements, a complete description of the occurrence and values for each of the Attributes such that additional validation of occurrences can take place.

The exercise also shows that reduction of the larger task to smaller tasks can permit use of junior analysts and of trained user staffs to complete the body of information on the dictionary. This reduces the workload on senior analysts and designers, who may then concern themselves with the actual design process.

It is suggested that the method, now tried, has introduced a greater degree of flexibility into the design structure, as provisions for suspension and introduction of Roles creates different structures. It is also suggested that substantial time and resource savings are achievable with these measures.

The search for additional techniques continues, as well as an attempt to enquire into the upcoming Relational Database standard to see whether the same Roles which achieved flexibility in the Network Database structure might not have the same effect there.

25. Acknowledgements

The authors acknowledge the implementation effort at North Eastern Gas by I.G. Smith and M. Higgitt; the ideas put forth by S.J. Pinder; the careful analysis work of D. Iosson; the assistance of M. Newton of the Open University and A. Johnson of ICL in defining the elementary roles; and the review by members of the British Computer Society Data Dictionary Systems Working Party.

BIBLIOGRAPHY

- ICL 79 ICL Data Dictionary systems, TP 6504, International Computers Ltd., Putney, England, as amended.
- ICL 80 ICL, "A Detailed Methodology for Data Analysis," ICL Training Publication Number T2392-4, 1980, ICL Training, Beaumont, Old Windsor, Berks., England.
- IOS 80 Iosson, D., "A Pure Conceptual Data Modelling Methodology: Ets Background, Development and Testing," M.Sc. Dissertation, 1980, Sheffield City Polytechnic, Sheffield, England.
- IOS 84 Iosson, D., and Meyer, K.H., "An Early Proposal: A Methodology for Data Analysis and Conceptual Data Model Building," 1984.
- LBM 77 L.B.M.S., "Logical Data Analysis" Course Manual, Available from: Learmonth and Burchett Management Systems, 22 Newman Street, London W1P 3HB, England. The edition used was prepared in 1977.
- MAD 82 Maddison, R.N., et al., Feature Analysis of Contemporary Information System Methodologies, The Open University, Walton Hall, Milton Keynes MK7 6AA, England, 1982. Updated and expanded in MAD 83.
- MAD 83 Maddison, R.N., et al., Information System Methodologies, BCS Information System Analysis Design Working Party (ISADWP), 1983, John Wiley, Available from BCS.
- MEY 80 Meyer, K.H., Diagrams associated with normalisation of data items within a projected dictionary, 7 December, 1980, private.
- MEY 82 Meyer, K.H., and Morse, C.C., Ch. eds., DDSWP Journal of Development, BCS Data Dictionary Systems Working Party, 1982, British Computer Society, 13 Mansfield St., London W1M 0BP, England.
- MEY 83 Meyer, K.H., and Doughty, J.R., "Devolution of Functional Analysis," SIGMOD Record, Vol. 13 No. 3 (April, 1983), available from Association for Computing Machinery.
- NOT 1 Information Analyst, Systems Analyst, Data Base Administrator, Data Base Manager and staffs. In general, the systems investigation and development function, as opposed to maintenance, within the firm at present.
- PIN 80a Lecture of Stephen Pinder to the ICL Data Dictionary Sub-group, 24 November, 1980.
- PIN 80b Lecture of Stephen Pinder to the BCS DDSWP with I.G. Smith and D. Iosson, November, 1980.