OBJECT COMPOSING
BY
A CATCHING ALGORITHM

A Problem Sketch of GIS
within
The Netherlands Railways

Joop W. Blom
Oostelbeers
The Netherlands

Dissertation submitted in part fulfilment of
requirements for the Degree of
Master of Science in Geographical Information Systems

May 1997

To Stephanie, Joop, and Monique

Abstract

Composing objects out of spaghetti: feasible or not? This is the central question of this thesis. For this reason, a study was made of a complex object in a complicated environment i.e. rail points at a busy railways intersection in the Netherlands.

The features of a point are registered by means of a system of drawing conventions, which ensures that all these features are implicitly present in the files designed by the NS department of Geodesy and InfraData. Each rail point is located via the relevant intersection. The co-ordinates of this intersection are the geographical attributes of the point. In the drawing's Data eXchange File (DXF) all of the relevant attributes can be found in a rectangular area around the centrally situated geographical point of suspension. These implicit attributes become explicit by extracting them. By concatenating the X- and Y-co-ordinate one generates a code. This code can be characterised as a 'Spatial Code', because sorting by this 'Spatial Code' generates a spatial ordering of the map. A so-called 'ZigZagChain', which starts from the bottom for each X co-ordinate and, leads from left-hand bottom to right-hand top position.

On the map a catch algorithm defines a rectangle (a catch-box) and as a consequence an interval of X-co-ordinates (a small vertical strip), within which the element can be found. By reading all of the records within the strip, all of the possible elements are passed and tested to see whether they are situated within the box or not. When an element meets the requirements and further reading is possible then the element found defines a new box. The algorithm will be executed again. In this way the nearest and also the most likely element will be located. Repeat this algorithm for all of the desired features of the object to be composed, so that they can be identified and stored. The composed objects, complete or incomplete, can now be presented by an overlay as well as by a table.

The functions within the algorithms depend on the drawing conventions, this implies that the results say also something about the extent to which the rules were observed. In this case 266, out of the 294 points were located using the above mentioned algorithms, a result of 91%.

It is now possible to form a link between CAD and GIS.

Main Chapters

Table of Contents

List of Figures

Disclaimer

The results presented in this thesis are based on my own research in the Department of Environmental and Geographical Sciences at the Manchester Metropolitan University, England. All assistance received from other individuals and organisations has been acknowledged and full reference is made to all published and unpublished sources used.

This dissertation has not been submitted previously for a degree at any Institution.

Signed:

J.W. Blom

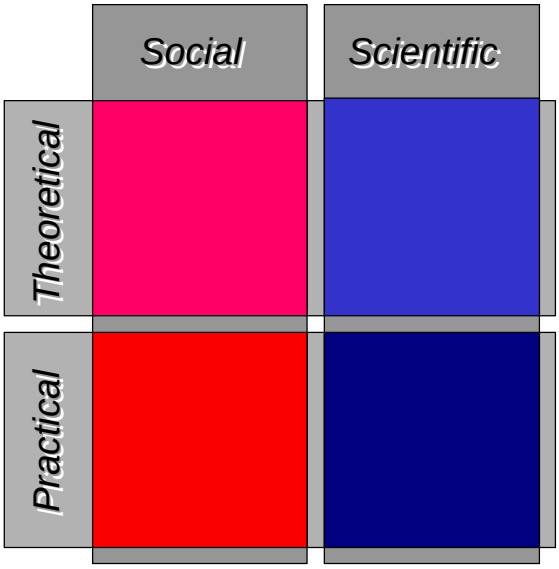
May 1, 1997

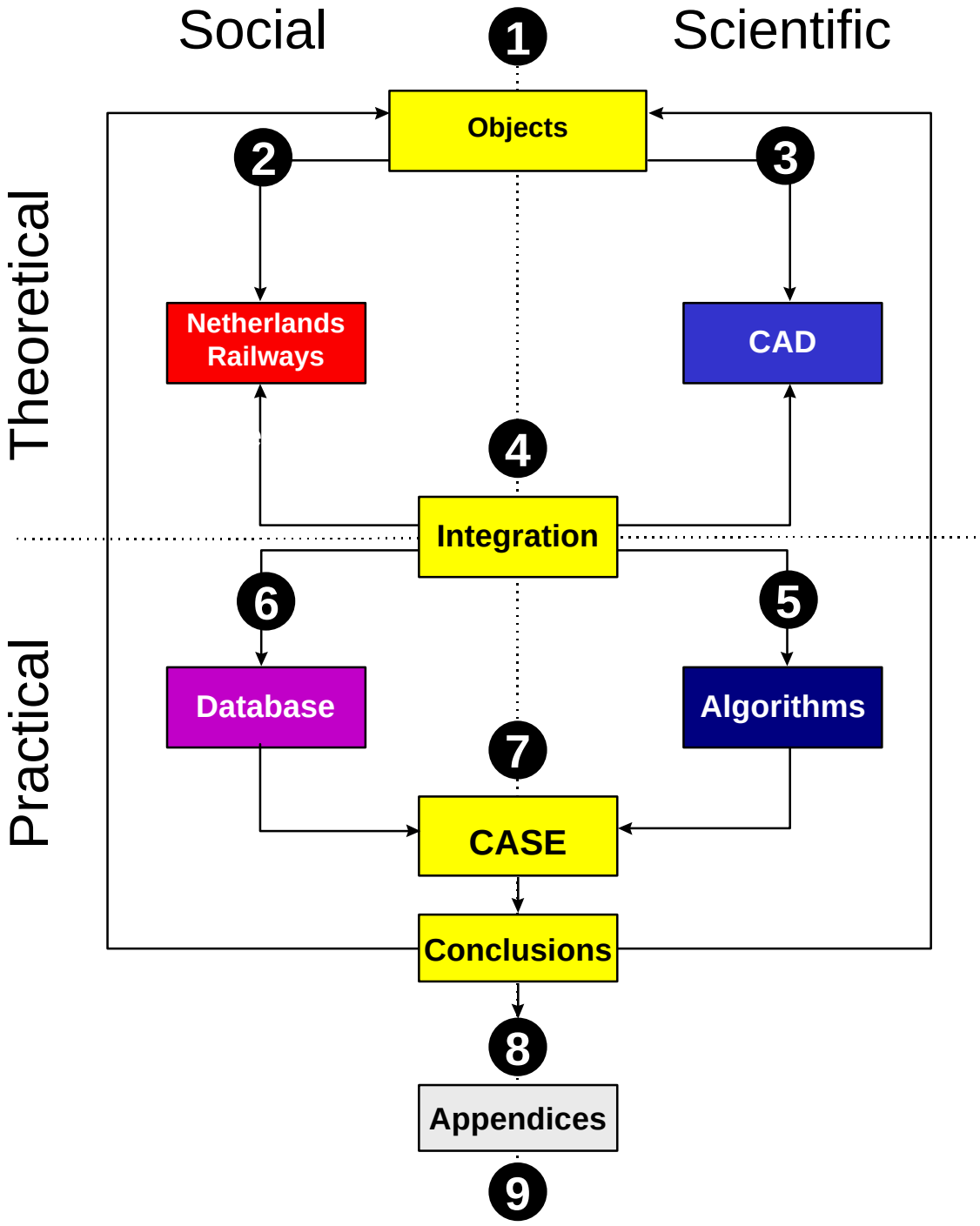
Oostelbeers, The Netherlands

Perspectives

Each sketch needs a perspective. However, a serious problem has a number of aspects and should be considered from various angles. A practical/theoretical approach versus a social/scientific one has been chosen. So the Social/Theoretical, Social/Practical, Scientific/Theoretical, and Scientific/Practical areas come into being.

The next page shows the structure of the thesis and the main chapter division in this view.




Social
Scientific
Theoretical
Practical
1
Objects
2
3
Netherlands Railways
CAD
4
Integration
6
5
Database
Algorithms
7
CASE
Conclusions
8
Appendices
9

1. OBJECTS

1.1 Assignment

This dissertation is based on a project with the Netherlands Railways, the objective of which was to develop an automated process of appointment and recognition to form objects within existing geometric files and link them with administrative data.

In other words to: ***Create Objects out of Spaghetti***.

1.2 Objects

The first question is, what do we understand by objects? Up until now there has been little uniformity in the way that real phenomena are described. As a result of this there are inconsistencies, overlaps and gaps in the databases. So, the land surveyor, the tax official and the public housing officer all have their own idea about the phenomenon building, but they do not use the same definition of a building. In order to be able to describe reality can be described in a more uniform way and to subsequently make exchanges of information easier, it is necessary to depart from the so-called ‘object oriented approach’.

1.3 Object oriented approach

In the object oriented approach, a clear distinction is made between, on the one hand, defining objects as they are in reality together with the associated features (*terrain model*), and on the other hand the manner in which these objects are of exchanged (*delivery format*) and presented (*map model*). Apart from issues relating to the technical exchange format or the final method of presentation, it is important to ensure that the same objects are dealt with in the same terms. On the basis of these three sets of criteria it becomes possible to deliver the data according to the terrain model or in accordance with the presentation model. In this approach the data are where possible defined, independently of the application. The result is, that data from different

task fields become more compatible and supply and demand can be more closely geared to each another. *Figure 1-1* shows how the concepts of the object-aimed approach are related.

1.4 Objects versus entities

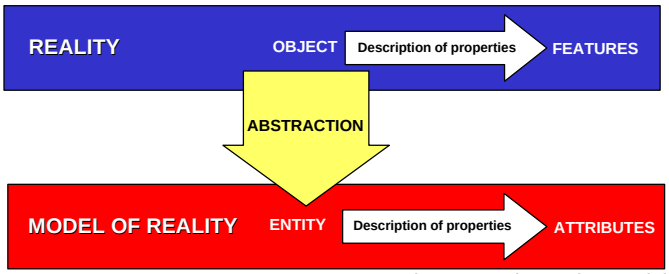


Figure 1-1:Abstraction model

Things in the real world are called 'objects'; elements in the conceptual scheme (the model of reality) are called 'entities'. An entity can be single or compound, it has a unique identification and at least one attribute (namely the classification code of the object). In the conceptual scheme it is possible to relate entities. A relation has one and only one identification and besides that at least two entity identifications and at least one attribute (namely the name of the relation)

1.5 Attributes

An attribute has a type identification and a value. Attributes can be classified as:

- Identifying:

  These attributes are added to an entity to make it possible to distinguish an entity from other entities in a unique way. Co-ordinates may also be used to describe an entity; these are geometrical attributes.
- Descriptive

  These attributes give a closer description of the entity. The value can be discrete (class, name and material category) as well as continuous (width, temperature and height).
- Geometrical

  These attributes are used to describe the position, form, and topology of an entity. The position and form can relate to points, lines, areas, grid, and pixels. These can take the form of discrete values (pixel number) as well as continuous values (co-ordinates).

- Graphical

  These attributes are applied to describe an entity with non-standard structures (graphics, photos and designs)

- Meta

  These attributes are employed to describe data.

In the norm-design of the Technical Committee (TC 287) of the Committee of European Normalisation (CEN) this approach is used to ensure that recognisable objects in the real world are the guideline for the projects in the future. The Dutch Terrain model ‘Real Estate’ (NEN 3610) also uses this object-aimed approach. In addition the standardisation at national level and at the European level needs to be considered.

1.6 European Normalisation

The CEN/TC 287 was established in 1992. This committee deals with the standardisation of geo-information, particularly the meta data. The products of this TC are explained in the report of CEN/TC 287and form an important pre-condition for standardisation projects. Under the Vienna agreement EU members are obliged to integrate European norms within National norms.

1.7 Consequences

The general objects within the Netherlands are classified according to NEN 3610. The specific , branch dependent objects have to be arranged by the branches themselves. The Netherlands Railways belongs to the branch “Ground, water, and roads” under the responsibility of the Ministry of Transport, Water- and Road Management. In the past there was a standard , but this standard has diverged as a result of a lack of central maintenance. In this case either NEN 3610 or for the time being the rail specific interpretation of the NS will be used when necessary.

1.8 Objectives

This investigation aims to:

- Establish a method for generic graphical entry for (rail)infrastructure databases.
- Establish a unifying system for object registration
- Implement the presentation of topography.

1.9 Constraints of the project

The project had to proceed within a number of constraints based on the work and production environment of the department Geodesy and InfraData.

- The costs of new systems are to be borne by each individual department.
- The quality of analogue maps must be maintained.
- Conversion to other systems of ME10, Eagle and AutoCAD is essential.
- A revision must be completed within 7 years.
- Changes must be tenable in the long term.
- Realisation must be accomplished by the current staff.
- The change over to a new structure must not get bogged down in many years of research. The approach must result in a rapid and phased delivery.
- The production of the management map does not depend on the needs of other disciplines.
- The scope of the system remains the same unless there is a principal with appropriate resources provided from elsewhere.

1.10 Classification

Taking the basic assumptions into account, it will be clear that any alterations to the digital map must be marginal. This means that the traditional layer structure, colours, line styles and symbols have to be maintained. The system has to however be able to recognise a building, a point……, and so on. Thus the system has to posses object-oriented tools. This makes it necessary to add an object classification to graphical elements. A simple classification model in *Figure 1-2* shows a point (x,y) which belongs to the class W (‘**w**issels’ Dutch for points)

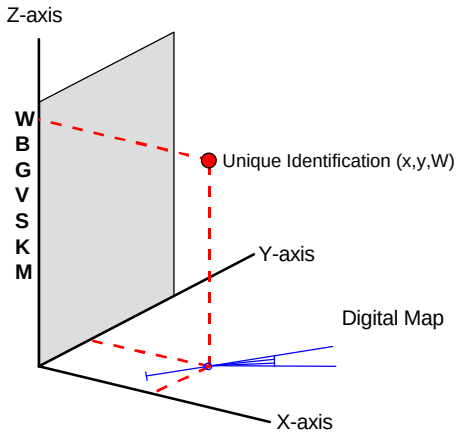


Figure 1-2: A Classification model

1.11 Object design

The map management system contains data and information concerning: entity geometry, overhead power supply, the railway, electrical installations, buildings and constructions, the property, paving and covering, azimuth, areas, annotations and the layer division. In such a complex array of data types and possible objects it is essential to carefully design objects. In order to illustrate the problem of design junction ‘points’ in rail lines could be considered.

1.12 Points

Points are railway specific and can be defined as devices which make it possible to alter the direction of the movement of train traffic.

1.13 Point types

The most frequently used types are: the regular, symmetric crossing, half English and English points. They consist of a number of characteristic elements

1.14 Elements

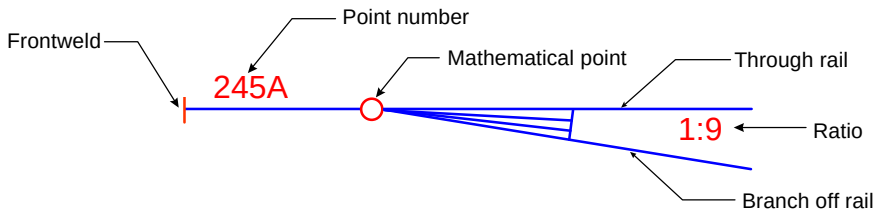


Figure 1-3: Rail Point elements

A point starts with a front weld, which on the map is represented by a narrow line at right angles with the representation of the rail, which in turn is represented by one line. The mathematical point is the intersection of the available rail traffic directions. According to the design conventions the point number is situated between the front weld and the mathematical point. Within and between the opposite open area of the through rail and the branch off rail we must find the ratio. A ratio of 1:9 means that in the direction of the rail at 9 metres, there is a 1 metre offset to the through rail.

1.15 Characteristics

The description of the elements above indicates that there is a system which is used by the rail engineers and that a map designer is bound by their conventions. This gives the rationale for goal-oriented searching which can lead to the ‘recognition’ of elements, which belong together as an object based on grouped elements.

1.16 Identification

The above can lead to the unique identification process aimed at.

1.17 Implementation

How can we manage this? We have to search for existing algorithms or if they are not available, or not useable, we have to develop them.

1.18 Restrictions

A great deal depends on the quality of the material available and on how strictly the design conventions are observed.

1.19 The Problem

Find and cluster the elements, that belong together and are situated within a certain area on the map, in an efficient way.

1.20 The Solution

A computer system reads sequences in a very fast and an efficient manner. But how do we get a sequence out of a map, or how do we get something into a linear form (dimension one) when it is two-dimensional. This dimensional reduction can be achieved, when we are able to obtain a linear ordering of the elements concerned, that is the key to the solution.

2. THE NETHERLANDS RAILWAYS

2.1 The context

This investigation is based on the work of NS Geodesy and InfraData, which is a small part of the NS operation. The rationale for the study is found in the political and economic changes taking place at European level.

2.2 The perspective

The European borders have opened up, and travel and trade between countries has become virtually unrestricted. The market for rail transport has also opened up, allowing for and even encouraging competition. From now on, national governments will restrict their involvement in the railways to the construction and maintenance of the rail infrastructure and at the same time gradually stop subsidising the operations deficit. Thus, the railway companies themselves have to see to it that their network becomes profitable. For an enterprising carrier, this new situation represents a unique challenge. Any entrepreneur who is in a position to offer the best possible transport services at the most favourable price will soon take over the market. The Netherlands Railways (NS) is attempting to meet this challenge.

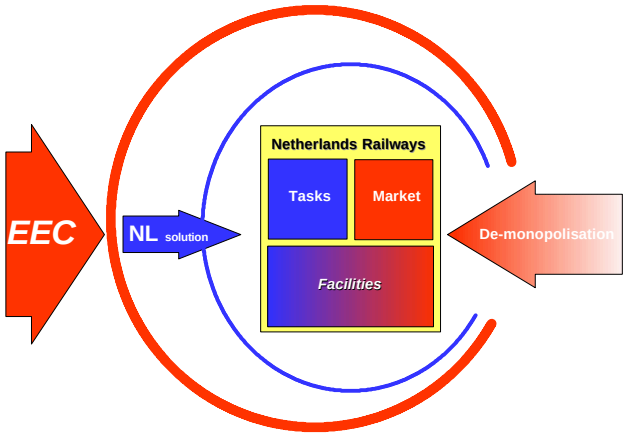


Figure 2-1: The influence of the EEC

2.3 The challenge

Any company, ambitious to become commercial, has to be able to swiftly adept to the market. It is for this reason, that the NS is gradually disengaging itself from central government control.

2.4 Separating operations and infrastructure

The NS already has taken the first steps on the road to becoming a commercial enterprise. Its main activities have been divided up in such a way that: commercial operations are now separate from the rail infrastructure. In other words, the customer-oriented activities are now independent from rail and traffic facilities. Customer services are now part of the NS Group, a commercial enterprise, while central government still remains reponsible for the infrastructure, contracting this work out to the NS. This task falls under an autonomous part of the NS organisation.

2.5 Active entrepreneurship

The challenge which the NS is tackling calls for more than just trains running on schedule. It requires active entrepreneurship in all kinds of markets: from the extension of real estate management to investments in telecommunications; from security services to technical research. And this, not only for the benefit of NS operations, but for each and every customer we serve.

Figure 2-2: The NS sectors

The Netherlands Railways (NV Nederlandse Spoorwegen) accommodates both the commercial business units of NS (the market sector) and those charged with the construction and maintenance of the infrastructure (the task sector).

2.6 The market sector

The NS Group covers four core commercial activities namely:

2.6.1 NS Passengers

Manages the transport of some 900,000 passengers per day and, moreover, offers special products such as vacation packages as well as the “Treintaxi” (a discount taxi service to and from stations for train-ticket holders).

2.6.2 NS Cargo

Is a transport company which core activity is rail transport. NS Cargo also offers other services in door-to-door combined transport, such as transhipment and access and egress transport. In this way rail transport is rapidly becoming a steady part of the total logistics chain. More than 70% of the NS Cargo transport services is cross-border.

2.6.3 NS Stations

Manages and exploits all of the Netherlands` 366 stations and is transforming them into pleasant, safe and clean places, with shops, new forms of commercial services and attractive cafes and restaurants.

2.6.4 NS Real Estate

Is a leader in the Dutch real estate market, owning 7660 parcels of land and virtually all of the buildings on it. As an operator and developer, this business unit provides services to all market parties in and outside of NS.

2.7 The task sector

Central government finances the infrastructure of the rail network in the Netherlands. It has contracted out the relevant activities to NS, as three government-commissioned rail infrastructure agencies.

2.7.1 Railned’s

Activities comprise the allocation and use of capacity on the Dutch rail network, the expansion of capacity and the supervision of rail safety.

2.7.2 NS Rail Infrastructure

Ensures the upkeep of the existing railway infrastructure and the construction of new infrastructure.

2.7.3 NS Traffic Control

Ensures the smooth and safe operation of rail traffic in the Netherlands.

2.7.4 Present situation

NS still has but a single shareholder, namely the Dutch State which exercises its influence on socially desirable services by contributing towards the costs of operations. This contribution will eventually be discontinued. In a few years, the NLG 450 million state subsidy for the operations deficit will be reduced to NLG 0,00. Central Government will then only pay for specifically commissioned services. In short, subsidies will become contracts. Various business units will have to become commercially active: that is, enter the market on a competitive basis. These are, in part, new activities for the NS. Within the NS organisation, each business unit is responsible for taking the decisions necessary for sound operations. Increasing income and saving on costs are key factors in this process. In this way, NS intends to develop into a financially healthy and profit-making company, on which customers and business partners can rely. Financial independence requires an efficient approach leading to greater productivity at lower costs. The aim is to guarantee the long-term continuity of our product, company and employment opportunities. To achieve this, the NS fosters customer relations, professionalism and entrepreneur ship. All of the NS' customers have different needs and requirements ranging, from the needs of an individual passenger to a large company requiring freight transport; a project developer or central government. NS provides customised products for everyone.

2.8 Distribution land

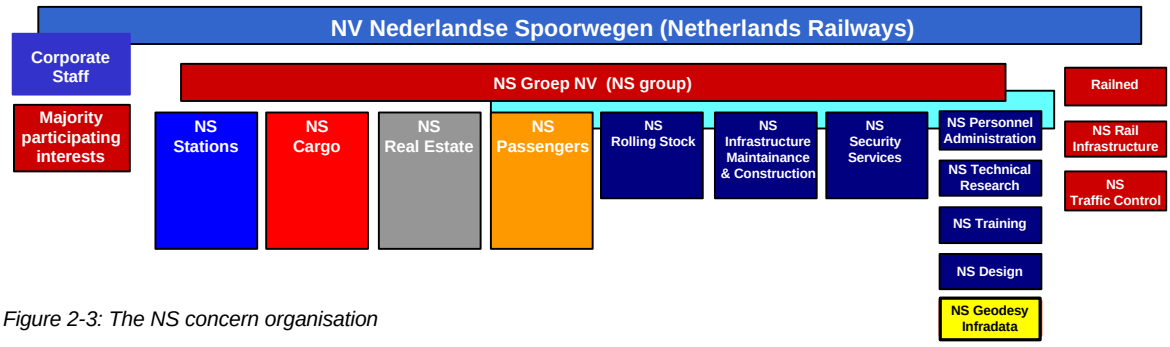


Figure 2-3: The NS concern organisation

The NS' contribution to the Netherlands as a distribution country is significant. The adjustment, expansion and renovation of rail links are only a part of this. High speed lines will reduce travel time in Europe making it more readily accessible for everyone. Also in the area of freight rail transport, new activities will yield a greater market share. An important condition for this is the development of the Betuwe Link (a dedicated freight link from the port of Rotterdam into the German hinterland).

2.9 In business

The NS' mission is not solely restricted to transport, however. In the future, they will be active in a variety of areas. For example, as a project developer; a partner in telecommunications; and as a supplier of personnel accounting systems, training courses and security services. The NS will furthermore be available for the maintenance of other transport means as well as bridges and viaducts. Moreover, NS will function as an architectural bureau, a partner in technical research and a graphic designer. In short, NS is becoming a multi-faceted enterprise, with dedicated employees eager to serve the NS' equally diverse customers

2.10 Facilities

The NS Group also comprises a number of business units, supplying products and services primarily to the NS, while also measuring up to external competitors. As a result, these units also focus on external markets.

The three largest of these are:

- NS Rolling Stock, charged with developing, repairing, maintaining and cleaning rolling stock.
- NS Infrastructure Maintenance and Construction, which handles the maintenance, renovation and small-scale construction of the infrastructure.
- NS Security Services, which provides a total package of integral security and safety services.

Smaller units within the NS group include:

- NS Personal Administration
- NS Technical Research
- NS Training
- NS Design.
- NS Geodesy and InfraData

NS now only consists of financially independent units with their own accounts departments. Since 1995 all of these units work have been operating in a free market.

2.11 Geodesy within the NS

Geodesy has a function in real estate management, the construction as well as the management of the railway infrastructure. This means: Geodesy and InfraData serves the task- (Rail Infra) as well as the market sector (Real Estate). Its annual turnover is about NLG 10 million, and its employs 50 people. The NS has a Geodesy Unit because of the fact that NS uses three geodetic information systems

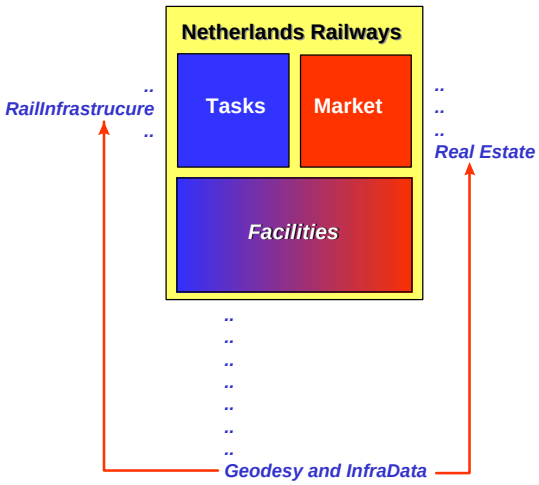


Figure 2-4:The position of G and I

- The topographical system 1:1000
- The land register system

- The railway geometry system

The primary task of the geodesy unit is the functional management of these three information systems.

This means:

- Defining the rail infra design
- Maintaining the infrastructure (partly through contracting)
- Database management
- Developing production processes.

These systems are under the supervision of a private NS company, since the issues of property and the systems are railway specific and therefore NS bound.

2.12 Functions and tasks

Functional management of the three geodetic information systems

Providing of geodetic support of the following processes within NS Railinfra and NS Real Estate.

- Infra projects (plan, land acquisition, realisation)
- Maintenance of railway infra structure.
- Real estate transactions

In the design phase of the infra projects external production capacity is often used in particular for the photogrammetric production of a Digital Terrain Model. There is one infra project, the Betuwe link that uses a completely different approach, namely full contracting out of the plan- and design phase, including the geodetic work. Modular construction orders in the field of GIS, including the accompanying databases. Upon completion Geodesy and InfraData will also be able to use these systems.

Examples

- Geographical Management System of Land Acquisition
- Registration system for NS licenses (third party objects in the railway)

- The so-called “Zone-plan” of NS Real Estate.

2.13 Organisation and financing

Geodesy and InfraData works in three areas of geodesy:

- Land surveying, land registration.
- Registration of licences.
- A technical information system concerning real estate and infra projects.

2.14 The principals

2.14.1 NS Rail Infra

With about 700 employees

Is one of the three ‘Task organisations’ within the NS and is the principal on behalf of the National Government for all affairs concerning rail infra (projects, management and maintenance)

And controller of that part of the real estate that contains rail infra.

2.14.2 NS Real Estate

One of the four primary business units in the market sector with circa 70 employees.

Income based on commercial exploitation of real estate (by transactions, development of projects and leasing and letting of business sites)

And controller of that part of the real estate that does not contain rail infra.

2.15 The project

For the project is it useful to know in the context, and the position of the department of Geodesy and InfraData holds within the NS concern. From this it is possible to deduce that the decision-makers are within NS Rail Infra and NS Real Estate.

3. COMPUTER AIDED DESIGN

3.1 A definition

CAD systems perform interactive geometric modelling in 2D and 3D, utilising the four components of description, processing, storing and model depiction. Data and graphic information are simultaneously processed, whereby the work in progress can be monitored on the screen.

3.2 The graphic approach

In many cases maps and images are converted into a digital format simply for selective retrieval and display. A surprising amount of digital cartography is merely electronic drafting. For cartographic applications, graphical entities are often traced electronically from existing maps only to be selectively redrawn with additional annotation and other embellishments. These operations are analogous to those involved in electronic drafting and are increasingly being handled by computer aided design (CAD) systems. In essence, CAD systems handle geographic data in the same manner as photographic separations in the production of topographic maps.

Different types of geographic features are placed on individual layers that are then combined and printed in different colours and line styles to generate the final product.

Although the concept is the same, CAD systems provide much more versatility in terms of display functions than their photographic counterparts and are particularly useful for editing and updating.

While offering major improvements over photo-mechanical methods of map production, CAD systems have severe limitations when it comes to analytical tasks. In particular, it is difficult to link attributes in a database to specific geographical entities and then automatically assign symbology on the basis of user-defined criteria [1].

For example, a CAD system could be used to create a graphical representation of a residential sub development consisting of all the property lines separating individual land parcels. In fact, the CAD system typically would generate smooth curves for cul-de-sacs and would force all the lines to join perfectly. The system would also enable the cartographer to point to a particular land parcel and shade it with a pattern. The CAD system by itself, however, could not automatically shade each parcel based on values stored in an assessor's database containing information regarding ownership, usage, or value.

In other words, a CAD system is merely a graphic system. This is not to suggest that such systems are not useful.

In fact, a PC-based CAD system linked to World Data Bank II has provided the State Department's Office of the Geographer with the capability to quickly generate base maps for any part of the world.

CAD systems however remain systems intended to design. That means for GIS in a limited environment.

3.3 Strategy

The sky is not the limit. We know that the possibilities are restricted, but let us go and see where the borders of CAD lie, and how the gap between CAD and GIS can be bridged. The operating rules of the draughtsmen will be the basic assumptions.

3.4 Transformations

In CAD transformations are very important. They make it possible to give a form the right proportions (scaling), the right direction (rotation) and the right place (translation). Here we consider a 'house' around the 'origin' of a x, y co ordinate system, and every square is a 1x1 unit. The origin house is always dashed in blue.

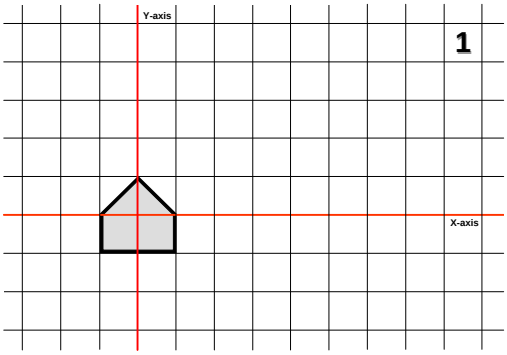


Figure 3-1: The original

3.4.1 Scaling

Scale(3,2) means here: every x-co ordinate becomes three times greater and every y co ordinate 2 times.

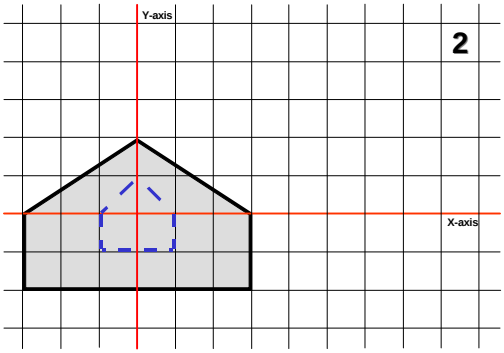


Figure 3-2: After scaling

3.4.2 Rotation

Here a clockwise 90 degrees rotation over the origin is illustrated. Every x co ordinate becomes a y co ordinate and every y co ordinate becomes a x co ordinate. Rotation can also be undertaken through any angle α.

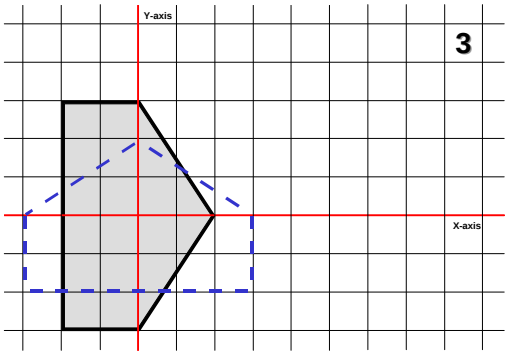


Figure 3-3: After rotation

3.4.3 Translation

Here we see a translation over 5 in the X- and 2 in the Y-direction. Thus add 5 to every X- and 2 to every Y co ordinate.

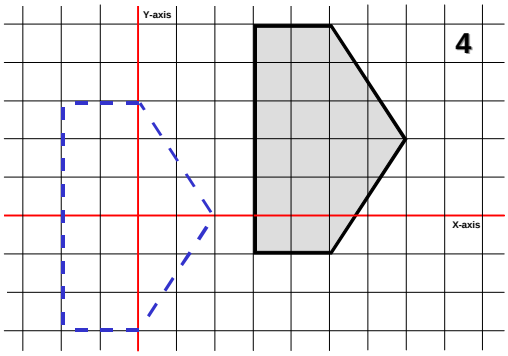


Figure 3-4: After translation

3.4.4 Any rotation

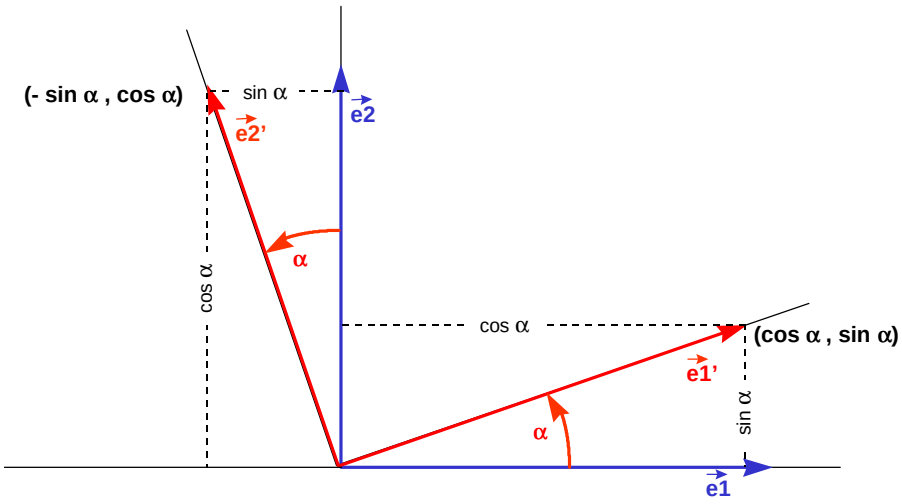


Figure 3-5: Any rotation

$$\vec{e}_1' = \cos a.\vec{e}_1 + \sin a.\vec{e}_2 \qquad (1)$$
$$\vec{e}_2' = -\sin a.\vec{e}_1 + \cos a.\vec{e}_2 \qquad (2)$$

$$\vec{v} = x.\vec{e}_1 + y.\vec{e}_2 \qquad (3)$$
$$\vec{v}' = x.\vec{e}_1' + y.\vec{e}_2' \qquad (4)$$

Substitution of (1) and (2) into (4) and remain (5).

$$\vec{v}' = x.(\cos a.\vec{e}_1 + \sin a.\vec{e}_2) + y.(-\sin a.\vec{e}_1 + \cos a.\vec{e}_2) \qquad (5)$$

Rearrangement of (5) and remain (6)

$$\vec{v}' = (x.\cos a - y.\sin a).\vec{e}_1 + (x.\sin a + \cos a).\vec{e}_2 \qquad (6)$$

or,

$$\vec{v}' = \begin{pmatrix} \cos a & -\sin a \\ \sin a & \cos a \end{pmatrix}.\vec{v} = A.\vec{v}$$

The matrix $A = \begin{pmatrix} \cos a & -\sin a \\ \sin a & \cos a \end{pmatrix}$ is orthogonal. Its determinant $A = 1$

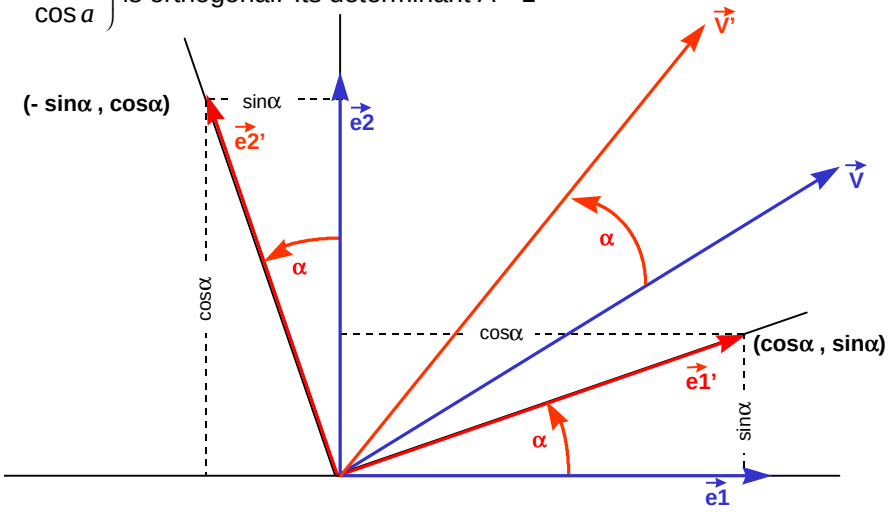


Figure 3-6: Rotation of a vector

3.5 A short description of DXF

A drawing interchange file is simply an ASCII text with a file type *.dxf* and a specially formatted text. It is possible to generate a drawing interchange file from an existing drawing by means of a DXFOUT (export) command.

Similarly a drawing interchange file can be converted into 'any' CAD drawing by means of a DXFIN (import) command.

The overall organisation of a DXF file is as follows [3]:

HEADER section:

General information about the drawing is found in this section of the DXF file. Each parameter has a *variable name* and an associated value.

TABLES section:

This section contains definitions of named items.

Line type table (LTYPE)

Layer table (LAYER)

Text style table (STYLE)

View table (VIEW)

User Co ordinate System table (UCS)

View port configuration table (VPORT)

Dimension Style table (DIMSTYLE)

Application Identification table (APPID)

BLOCKS section:

This section contains Block Definition entities describing the entities that make up each Block in the drawing.

ENTITIES section:

This section contains the drawing entities, including any Block References.

END OF FILE

A DXF file is composed of many *groups*, each of which occupies two lines in the DXF file. The first line of a *group code*, which is a positive non zero integer. The second line of the group is

the *group value*, in a format that depends on the type of group specified by the group code. Although DXFOUT output has a fixed format, the DXFIN format is free. The specific assignment of group codes depends on the item being described in the file. The appearance of values in the DXF file is not affected by the setting of the UNITS command: co ordinates are always represented as decimal (or possibly E notation if very large) numbers, and angles are always represented in decimal degrees with zero degrees to the east of origin. Variables, table entries, and entities are described by a group that introduces the item, giving its type and/or name, followed by multiple groups that's supply the value associated with the item. In addition, special groups are used for file separators such as markers for the beginning and end of sections, tables, and the file itself. Entities, table entries, file separators are always introduced with a 0 group code that is followed by a name describing the item.

The maximum DXF file string length is 256 characters. If the drawing contains strings that exceed this number, those strings are truncated during DXFOUT. If a DXF file contains strings that exceed this number DXFIN will fail.

- Group codes are used both to indicate the type of the value of the group, and to indicate the general use of the group. The specific function of the group code depends on the actual variable, table item, or entity description. This section indicates the general use of groups, noting as "fixed" any that always have the same function.
- Comments are indicated by the 999 group code. That means the following line is a comment string. For example:

  999

  This is comment.

  999

  This is another comment
- File sections

  The DXF file is subdivided in four editable sections, plus the END OF FILE marker File separator groups are used to delimit these file sections. The following is an example of a void DXF file with only the section markers and table headers present:

```
0                    (Begin HEADER section)
SECTION
  2
  HEADER
Header variable items go here (General  drawing information) 1
  0
    ENDSEC                    (End HEADER section)
      0                       (Begin TABLES section
  SECTION
  2
  TABLES
Table definitions go here (Line types, Layers, Styles, Views….) 2
  0
        ENDSEC                (End TABLES section)
            0                 (Begin BLOCKS section)
            SECTION
            2
            BLOCKS
Block definition entities go here (Internal Library) 3

            ENDSEC            (End BLOCKS section)
                0             (Begin ENTITIES section)
SECTION
2
ENTITIES
Drawing entities and Block references go here 4

  ENDSEC              (End ENTITIES section)
  0
  EOF                 (End Of File)
```

In short: A DXF consists of four sections, each section contains a number of groups, which has a code and an accompanying value, or DXF = {SECTION [group (code <value>)] }

What this means is that a drawing is a collection of entities (design elements). An entity can be: a POINT, LINE, ARC, TEXT or other design primitive, but also a collection of entities, which form an element of the internal library (a BLOCK or a SYMBOL or a CELL, that depends on the ruling CAD-environment). If a line has to be drawn, then the CAD-program calls the LINE entity with the start and end point of the line to be drawn. When you are calling up an entity, you must also give the values of the parameters, which are necessary to draw the entity in its correct form. Thus to drawn a certain circle, you must call the entity circle and give at least the centre

and the radius. Other specific characteristics, of line type and so on, if not default ,must also be given.

Library elements are self defined (design primitives) entities. These library elements are defined around the origin of the system, they can be scaled in x and/or y direction, can be rotated and translated to the desired place in the drawing. In addition it is possible to give each a line type, hatching and much more. It is also possible to make library elements with one or more attributes. By using library elements with ‘attributes asking dialogue boxes’, they can then be ‘speaking with attributes’ after placement in the drawing.

3.6 Summary

- A CAD program makes it possible to draw with a collection of parameterised entities.
- Drawing is: Calling for entities and substitution of the parameters Or in brief:
- A drawing is a collection of value-supplied entities
- These entities contains a mine of implicit information.
- Make this wealth of information explicit.

3.7 References

[1] Cowen, David, J., 1990. *GIS versus CAD versus DBMS: What are the differences?* In: Introductory readings in Geographical Information Systems. Taylor and Francis, London.

[2] Autodesk Publication TD106-011-UK, 1988*. AutoCAD, Reference Manual*.

4. INTEGRATION

This chapter attempts to relate the possibilities of CAD systems (chapter 3) to the operating problems of NS (chapter 1) in order to meet the challenge expressed in chapter 2.

4.1 The organisation

I shall restrict myself to the work of the department of Geodesy and InfraData and its specific cultural context which is dominated by its primary tasks, namely the functional management of the three geodetic information systems:

The topographical system 1:1000

The land register system

The railway geometry system.

Other tasks are subordinate and not with respect to her two principals NS Railinfra and NS Real Estate.

The organisation is not equipped for GIS. Large projects such as ‘the Betuwe-link’ are fully contracted out. There is no time or budget available for developments and they must be planned in a seven-year update cycle with existing personnel. In this context the development of the existing CAD system is the only way for development. The means

A CAD-environment offers surprising perspectives. In particular the open nature (ASCII) of DXF gives possibilities for extracting, manipulating by calculating and composing of objects with specific attributes. I have opted to use BASIC as the programming tool. A familiar programming language for me and many others.

4.2 The situation

The infrastructure of the NS consists of circa 3300 km rail. For every 900-1000 metres of rail there is a so-called ‘management map’ scale 1:1000, which is built up and maintained in the CAD environment of MicroStation. Within this environment is it possible to put together main groups per file in transparent levels (layers). Each level can be seen as a transparent sheet.

The overall drawing is built up of such sheets, which lie one on top of each other, each sheet containing specific drawing information. There are 63 levels available. Eleven are in use for the management map. Besides the normal parameterised entities to design, there is a set of 40 prepared parameterised entities (library).

4.3 Terms and ideas

Two of them, namely the Front weld and the mathematical point are components of the particular railway device, the point, which is the focus of this study.

What do we want to know about a point? What is the location of the point and how is that defined? What is the direction and by what is the direction determined? Where do the possible directions intersect each other? (Mathematical point).

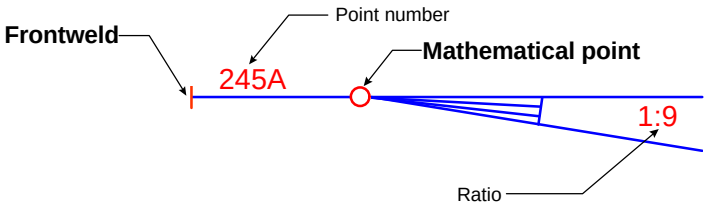


Figure 4-1: Terms around points

What is the number of the point? What is the point ratio? The entity Mathematical point is given through the location parameters of a circle. A circle is a library entity with certain radius translated to a position. That means: there is no possibility to derive a direction since a circle is not direction sensitive. On the other hand the Frontweld is an entity with location and direction parameters (rotate and translate to) and the frontweld is perpendicular to the rail direction and for that reason normative for the direction of the ‘object to compose’.

Step 1

The frontweld gives us the possibility to catch the other parts of the point. First the mathematical point, then we are able to find the point number and after that we can find the accompanying ratio.

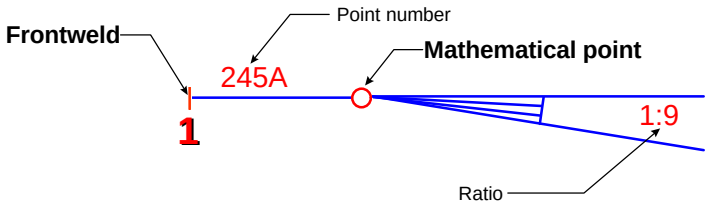


Figure 4-2: The characteristics of a point

Step 2

The width of the frontweld is 1 meter; the direction is given and the distance to the mathematical point is always less than 40 metres. It is reasonable to assume that it is 'certain to catch' the mathematical point within a rectangle of 2 by 80 metres

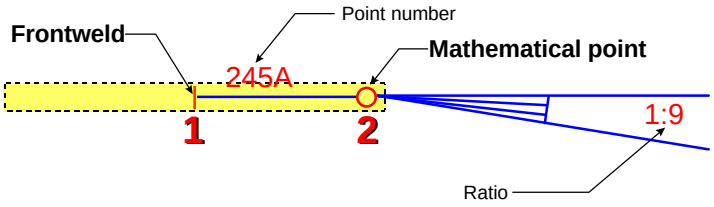


Figure 4-3: Catch-box to find the mathematical point

Step 3

It is certain that we are able to find the point number between 1 and 2 within a rectangle. We can therefore also catch the point number in the yellow box.

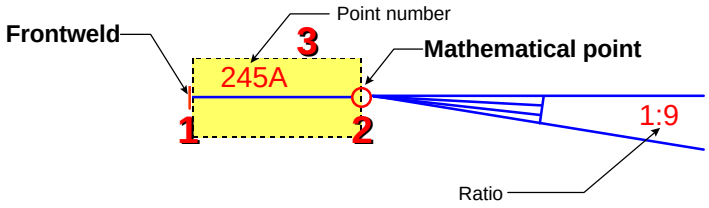


Figure 4-4: Catch-box to find the point number

Step 4

The ratio can now be caught too, since the position of the mathematical point between the ratio and the frontweld can be assumed, it becomes possible to define a complete catching box.

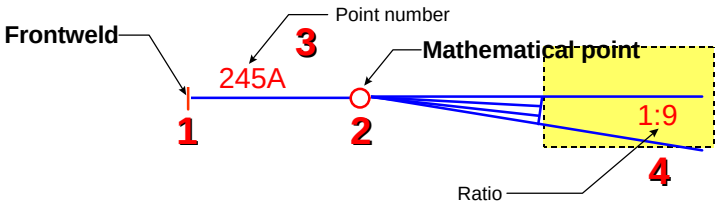


Figure 4-5: Catch-box to find the ratio

4.4 Possibilities

We can now give attributes to the point, symbolised by the yellow square. Attribute values can be added such as co-ordinates, direction, type number, and ratio date of last maintenance. It is of course necessary to have an identifier.

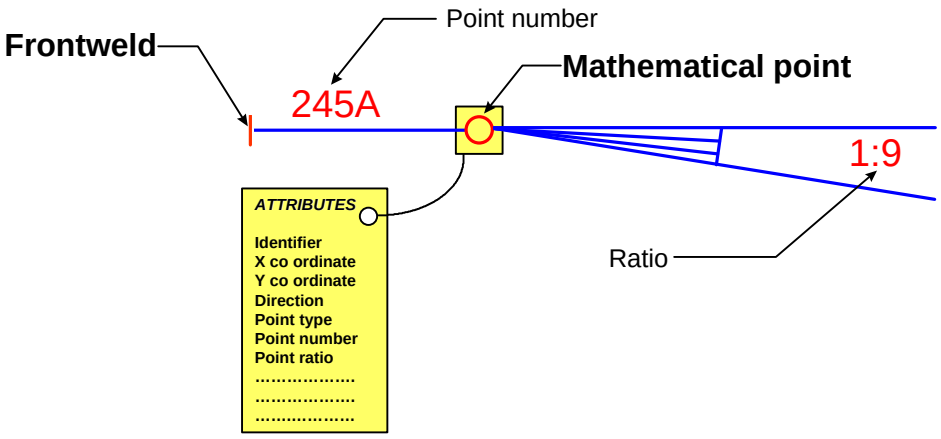


Figure 4-6: The mathematical point as hang-up point of attributes

4.5 Drawing enrichment

Within the NS eleven layers (levels) are used to produce a drawing in the management map at 1:1000. 63 layers are available. That means 52 layers remain unused.

This creates possibilities for using more layers containing, for example, ‘object information’, while the original information remains intact. *Figure 4-7* shows how BASIC routines can read implicit information and make it explicit by reading and writing the DXF.

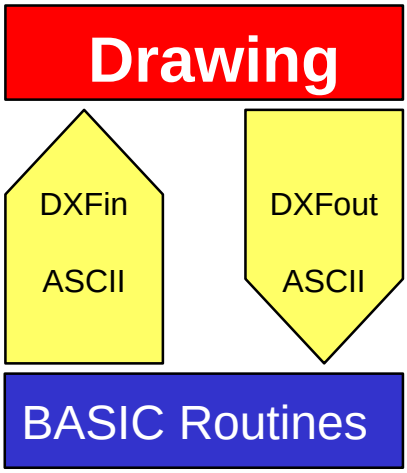


Figure 4-7: The Enrichment process-1

4.5.1 Drawing

The NS-map has a seven year maintenance cycle. This means about 10 sheets out of a total of 3000 sheets a week. This means that it would be attractive to include these routines that are able to show latent information from the existing map.

4.5.2 DXFout

Gives all the information required as input for the BASIC programming environment.

4.5.3 BASIC routines

These routines undertake that which is needed to write the DXF which contains the data to fill the supplementary layers. More specifically these routines are the ‘captive’ routines

4.5.4 DXFin

It is then possible to read ‘the object information’ in a separate layer of the original drawing using the DXFin option, possibly within the original drawing,.

4.5.5 Drawing

The result is a drawing containing the original data supplemented by ‘object information’.

4.6 The DXF structure

```
  0
SECTION
  2
HEADER
```

```
ENDSEC          end of HEADER
  0
SECTION
  2
TABLES
```

```
ENDSEC          end of TABLES
  0
SECTION
  2
BLOCKS
  0
BLOCK
  8
0               begin of block
  2             blockname
SQUARE
 70             blocktype flag
    64          bit 7 is on
 10             x-base point
0.0
 20             y-base point
0.0
 30             z-base point
0.0
  0             entity type
SOLID
  8             layer
0
  6             linetype
CONTINUOUS
 62             colour
     2
 10             x first point
-1.0
 20             y first point
-1.0
 30             z first point
0.0
 11             x second point
1.0
 21
-1.0
 31
0.0
 12             x third point
-1.0
 22
1.0
 32
0.0
 13             x fourth point
1.0
 23
1.0
 33
0.0
  0
ENDBLK          end of block
  8
0
  0
ENDSEC          end of BLOCKS
  0
SECTION
  2
ENTITIES
  0             entity follows
INSERT          a block
  8             on layer
0
 66             attributes flag
     1          with attributes
  2             blockname
SQUARE
 10             x-insert
0.0
 20             y-insert
0.0
 30             z-insert
0.0
  0
ATTRIB
  8
0
 10             x-insert
0.0
 20             y-insert
-1.5
 30             z-insert
0.0
 40             text height
0.25
  1             attribute content
001W
  2             attribute name
POINTTYPE
 70             visible?
     0          yes
  0
ATTRIB
  8
0
 10
0.0
 20
-2.0
 30
0.0
 40
0.25
  1
123A
  2
POINTNUMBER
 70
     0
  0
ATTRIB
  8
0
 10
0.0
 20
-2.5
 30
0.0
 40
0.25
  1
1:9
  2
RATIO
 70
     0
```

```
  0
ATTRIB
  8
0
 10
0.0
 20
-3.0
 30
0.0
 40
0.25
  1
05-09-95
  2
MAINTENANCE
 70
     0
  0
ATTRIB
  8
0
 10
0.0
 20
-3.5
 30
0.0
 40
0.5
  1
12
  2
DEGREES
 70             visible?
     1          no
  0
ATTRIB
  8
0
 10
0.0
 20
-4.0
 30
0.0
 40
0.25
  1
123456,567890
  2
COORDINATES
 70             visible?
                no1
  0
SEQEND          the end attribute
sequence
  8
0
  0
ENDSEC          the end of ENTITIES
  0
EOF             End Of File
```

4.7 A choice

The structure of DXF is complex, but it is also transparent, and that allows us to build up DXF independently of any CAD-environment.

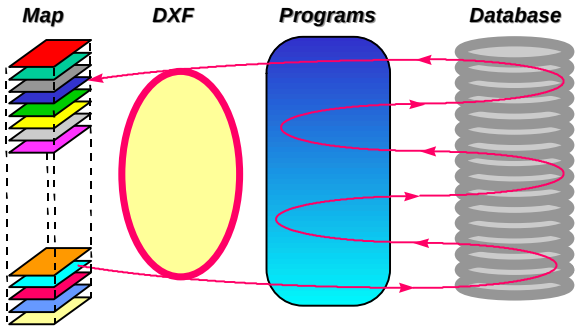


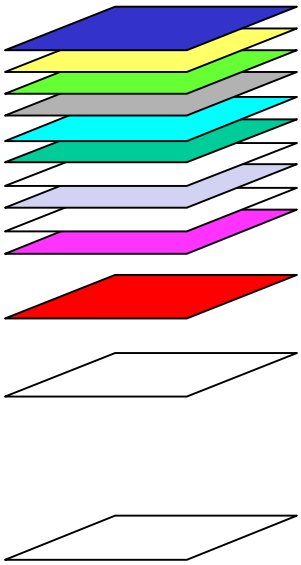
Figure 4-8: The Enrichment Process 2

4.8 The presentation

In each CAD-environment it is possible to import another drawing in a open drawing via a DXF-utility. This is the key to the presentation problem, which also meets the requirement to keep original drawings unchanged. The origin of drawing uses a member of layers, and by selecting layers that are free, it is then possible to keep the two apart.

This ‘Living apart and Working together’ configuration provides the basis for a solution to the problem of objects.

4.9 Additional layers



Level 01: Tracks, points, crossings and buffers. (see also Level 08).
Level 02: Contact wire and accompanying text
Level 03: Buidings, platforms and accompanying text.
Level 04: Remaining topography and text..
Level 05: NS property border and accompanying text.
Level 06: Signals, telephone, lamppost, level crossing security, switch boxes and accompanying text.
Level 07: Hectometre indication, North indicator and accompanying text.
Level 08: Hatching, pointnumbers, point ratio, and crossings (see also Level 01).
Level 09: Other hatching.
Level 10: Refence to adjoining sheet numbers.
..
..
Level ..: The Objects: POINTS
..
Level 33: Hatching of buildings.
..
..
..
Level 63: Sheet sizes.

Figure 4-9: Layers with objects

When we take ‘the Points’ as example. Then we select a free layer for the presentation of ‘the Points’: the red one. It then becomes possible to present The Object Layer POINTS ‘over’ the existing map.

5. ALGORITHMS

With the storage of spatial data, there must be a mapping to one dimension, because the two-dimensional data has to be stored in the one-dimensional memory of a computer. The aim is the clustering of the spatial data so that the elements - which in reality are close to each other - are stored, close to each other, because the chances are that they are also collected together.

5.1 Spatial ordering

The real problem is, how we get xy-co-ordinates (a two-dimensional problem) reduced to something ‘linear’ (one dimensional). We know that a computer system is only able to look in a file, to read forwards or backwards. That means, a problem is easier to tackle, if we can reduce it to a linear form. There are, of course, established methods for doing this.

5.1.1 Quadtree

One of them ‘the Quadtree’ is a generic name for a kind of search tree, which is built up by the recursive splitting of space in four quadrangles [1]. First the domain is enclosed by a quadrangle. Then the quadrangle is sub-divided into four quadrangles, and then each of the four are sub-divided again and so on. Quadtrees have many interesting advantages over other methods, but in general they are more suitable in a rastered environment. Why is not possible to base a Spatial-Location-Code-value on a quadcode? There are two possible options, but each presents difficulties: Each object is represented by one quadcode. However if an object (large or small) crosses one of the main split borders, then the result is a short quadcode string. It is clear that such a big area is not a good approximation for such an object. Each object could also be represented by one quadcode. The matching areas correspond better, but the whole becomes too complex to use within a database environment, because each object includes a collection of quadcodes. A solution to this problem is proposed here, namely the ZigZagChain Method

5.1.2 Concatenation

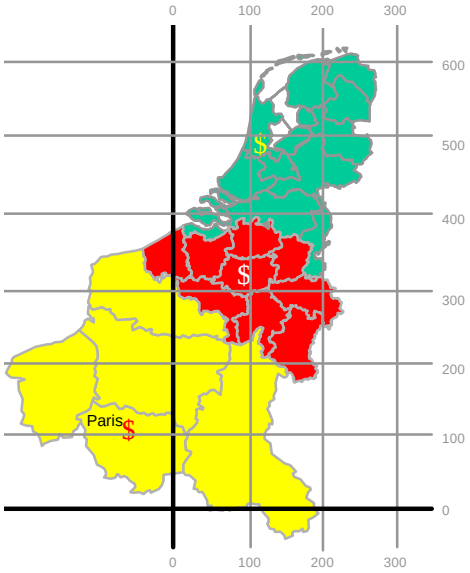


Figure 5-1: The co-ordinate system

Before describing the method we must first consider some other issues. The method has to do with co-ordinates. It is necessary to explain the system used in the Netherlands. The rectangle co-ordinate system of the national agency of triangulation is based on a stereographic projection, which is used for topographical mapping. The origin of this system is south-east of Paris. *Figure 5-1* shows the position of the Netherlands in this system. We see that the X co-ordinate is always between 0 and 300 kilometres, and the Y co-ordinate is always more than 300 kilometres. So that for places within the Netherlands the highest X-value is always lower than the lowest Y-value, and the two can not be mistaken.

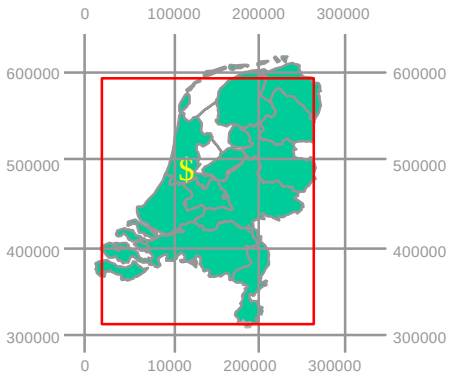


Figure 5-2: The Netherlands within the system

The NS infrastructure is situated within the red-lined rectangle and uses the co-ordinate pairs (X,Y). This means that X is between 0 and 300000 metres, and Y between 300000 and 600000 metres. This gives the basis for constructing a location related identification.

Consider an X co-ordinate symbolised by xxxxxx and if we multiply this by 1000000, we get: xxxxxx000000. Add the Y co-ordinate symbolised by yyyyyy and we get xxxxxxyyyyyy. (The X- and Y-co-ordinate are concatenated). The outcome must be between 300000 and 300000600000. The question now is, how can we handle these enormous decimal figures?

Figure 5-3: Spatial Coding

5.1.3 Another number system?

The decimal number system consists of powers of base 10. A greater base has give the numbers a shorter and more coded look. This can be found in the 36 number system.

It is possible to imagine that:

0,1,2,3,4,5,6,7,8,9,A,B,C,D,……X,Y,Z are the figures in a 36 (10+26) number system (composed of powers of 36. We here see that 36^8 is a number of 13 figures which is greater than 300000600000. All lower powers of 36 are less than 300000600000.

$36^8=2821109841920$
$36^7=78364164096$
$36^6=2176782336$
$36^5=60466176$
$36^4=1679616$
$36^3=46656$
$36^2=1296$
$36^1=36$
$36^0=1$

Figure 5-4: The powers of 36

This means that any co-ordinate combination in the Netherlands can be written as a number consisting of 8 figures from the 36 figures number system.

Example: (x,y)=(259035,456789)

Composition: 259035456789

36-Code: 3AZZ0TN9

Or: $3x\ 36^7 + 10x\ 36^6 + 35x\ 36^5 + 35x\ 36^4 + 0x\ 36^3 + 30x\ 36^2 + 23x\ 36^1 + 9x\ 36^0$

0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ

3×36^7	=	235092492288
10×36^6	=	21767823360
35×36^5	=	2116316160
35×36^4	=	58786560
0×36^3	=	0
30×36^2	=	37584
23×36^1	=	828
9×36^0	=	9
		259035456789

The next problem is to find out how such numbers are to be sorted or, rather, how they can be sorted as quickly as possible. An ordinary 'Bubble-sort' is inadequate. Hoare provides a solution.

5.2 The QuickSort

Figure 5-5: Sorting by Hoare

Sorting a great number of elements can be a problem, when these are not well-organised. The choice of the method of sorting is therefore important. A very quick method is QuickSort [2].

One example will suffice to explain how it works We consider the set V = {3,4,8,1,7,5,9,6,2}

5.2.1 Step 1

Choose the middle member of the set: (7), or in the case if an evenly numbered set, the first member after the second half of the set.

Consider each member from left to right and transfer the members equal or greater than the selected member (7) to its right. Similarly the elements to the right of (7) ,that are less than (7) must be transferred to the left of (7).

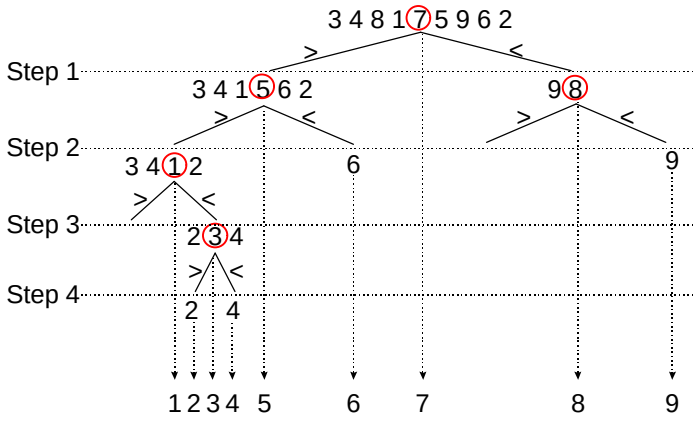


Figure 5-6: The sorting tree

The result is: {Elements < 7}, 7, {Elements >=7}, or {3,4,1,5,6,2,7,9,8}

5.2.2 Step 2

Repeat step 1 for the sub-sets {3,4,1,5,6,2} and {9,8}. (5) and (8) are now the breakpoints. We get {3,4,1,2,5,6} and{8,9} with the complete situation as {3,4,1,2,5,6,7,8,9}

5.2.3 Step 3

This is done for the sub-collections {3,4,1,2}, {6} and {9}. If a sub-set consists of one element it is sorted. As for the remaining we get nothing to the left-and {2,3,4} to the right of (1), or {1,2,3,4,5,6,7,8,9}

5.2.4 Step 4

The only remaining sub-set is {2,3,4}. For this no transfers. The remaining sub-sets consists of one element, so that we can stop. The final result is {1,2,3,4,5,6,7,8,9}. We shall see that this method is a very useful one.

5.3 Data access

In BASIC data files are divided into two groups:

The sequential file can be accessed only in an ascending manner. Data which are put in a file in a certain order, can only be read in the same order.

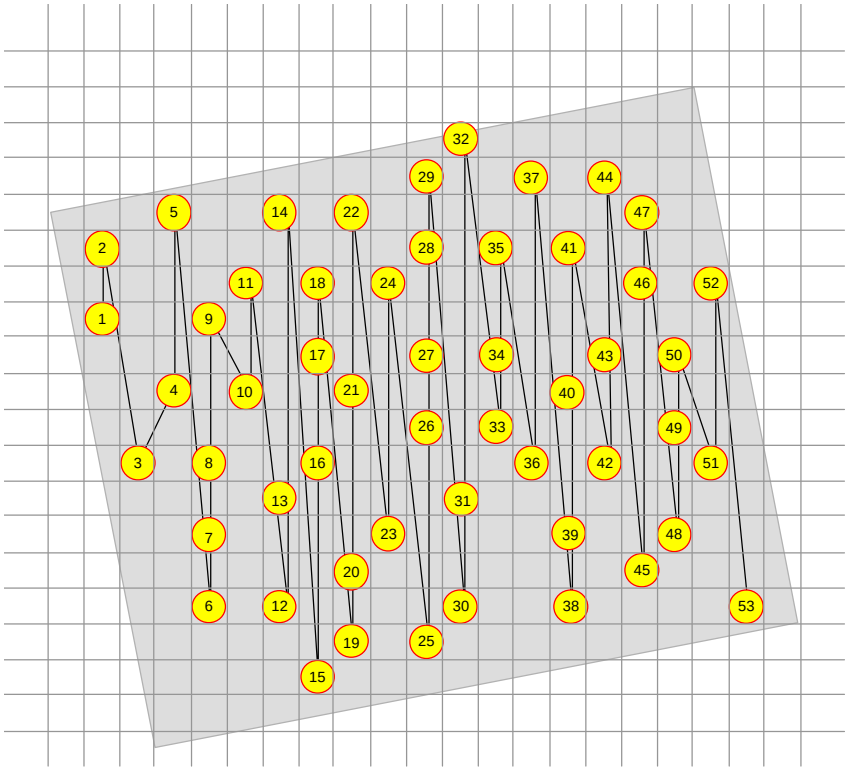


Figure 5-7: Solution through dimension reduction

The random file allows data to be approached in any order. In other words: We are able to read, say record number 34, without reading the first 33 records. This makes the approach easier and much quicker, and that is exactly what we want.

Figure 5-8: Reduction to dimension one.

We are now able to sort all of the elements of a map, because they are ‘spatially coded’ in the manner already explained. It is now possible to sequentially pass through the file. We have

threaded the elements in the manner illustrated on the next page. When we put the records in a file with direct access, we can pick out any record in the file and are free to read forwards as well as backwards.

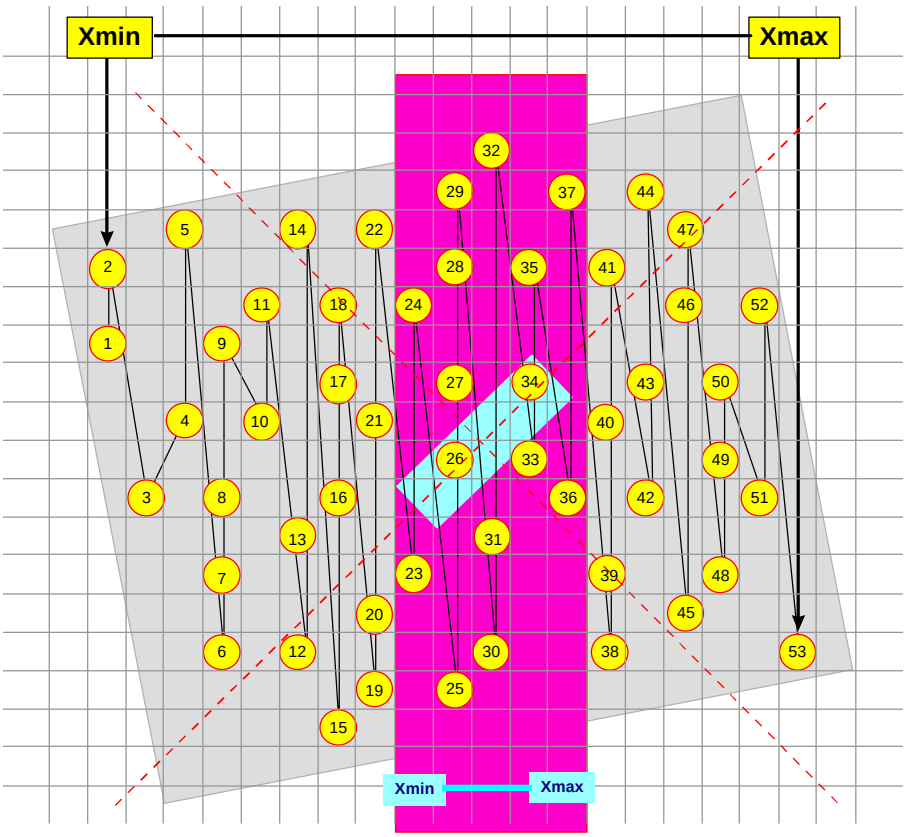


Figure 5-9: The relevant records

In *Figure 5-9* the grey map shows a rotated position in an XY co-ordinate system. Every set of entities from the map has an X-minimum as well as an X-maximum. This also applies to a box, see the purple strip Within the purple strip lie the entities we want to select, which may belong to the blue box. What you want to find out is whether an element belongs to a box or not.

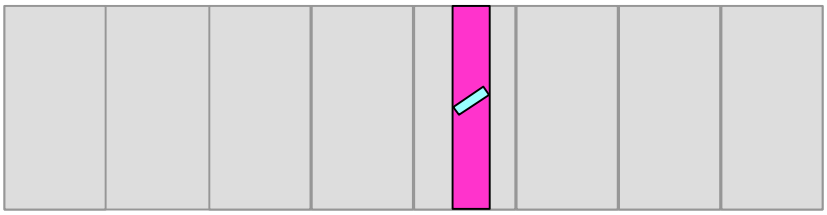


Figure 5-10: In a more realistic ratio

Figure 5-9 is only meant to illustrate the example but does not reflect the real situation. *Figure 5-10* comes closer to reality. The grey part is again the map of about 200 x 800 metres. The desired elements by X-co-ordinate for the box in blue are situated in the purple strip. We can now quickly read the attributes of the selected entities. One question remains. How can we decide whether a selected entity belongs or not?

5.4 Strip determination

The strip to read is prescribed by the positions of the angular point of the box. The co-ordinates of that point can be found, when we think in four steps.

1. Starting point is a quadrangular box around (0,0) with right top (1,1)
2. After X-scaling with factor DY and Y-scaling with factor DX. So that the right top of this box is (DY,DX)

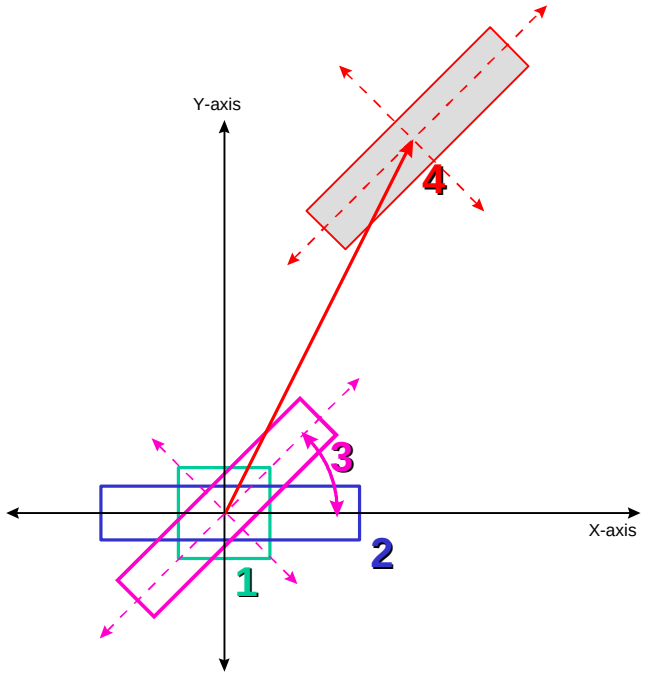


Figure 5-11: Box positioning

3. This box comes about after a rotation over an angle α. Application of the transformation given through the matrix of the right top position becomes ($\cos\alpha$.DY-$\sin\alpha$.DX,$\sin a$.dy+$\cos\alpha$.DX)
4. A translation to P(a,b), with a right top position of (a+$\cos\alpha$.DY-$\sin\alpha$.DX,b+$\sin a$.dy+$\cos\alpha$.DX) for the final box

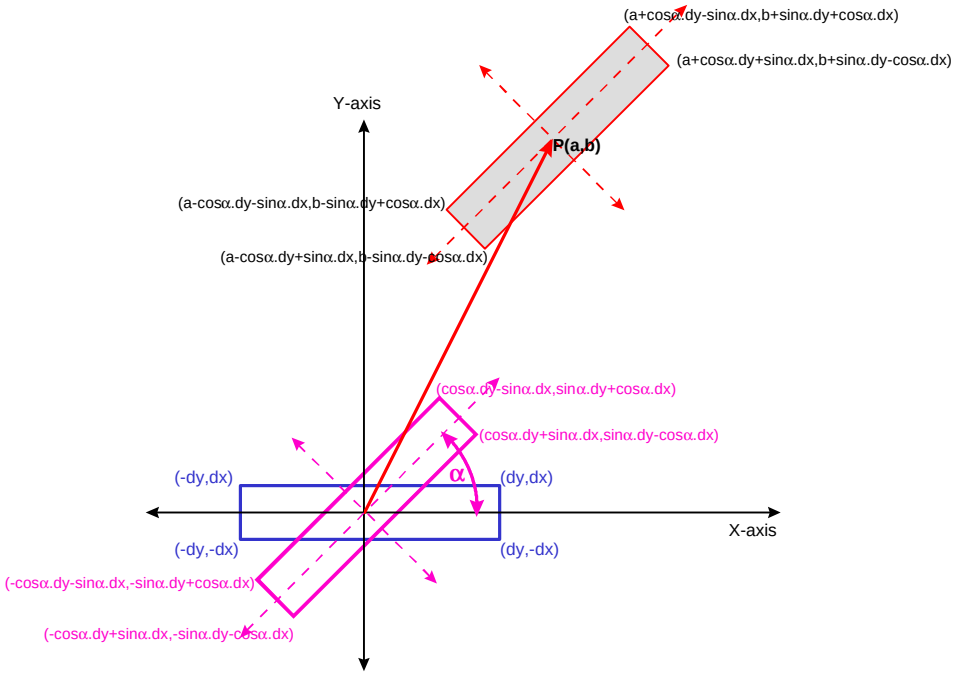


Figure 5-12: Box co-ordinates

5.4.1 Box co-ordinates

The co-ordinates of the other angular point are easy to derive via mirroring.

RightTop: RT(a+cosα.DY-sinα.DX,b+sina.dy+cosα.DX)

LeftTop: LT(a-cosα.DY-sinα.DX,b-sina.dy+cosα.DX)

RightBottom: RB(a+cosα.DY+sinα.DX,b+sina.dy-cosα.DX)

LeftBottom: RT(a-cosα.DY+sinα.DX,b-sina.dy-cosα.DX)

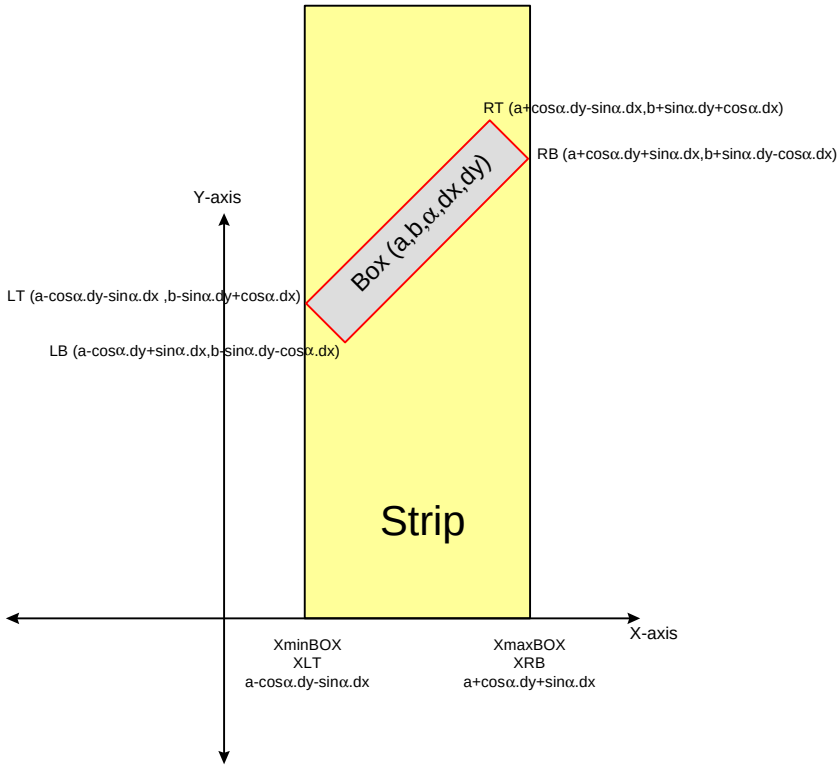


Figure 5-13: Strip limits

5.4.2 Extremities

The angular points of the box determine the strip, which in turn depends of the rotation angle α

5.4.3 Interval

The interval depends on the X-co-ordinates of the extremities. In the situation of *Figure 5-13* the strip is determined by the interval [XLT;XRB]

5.5 Binary file read control

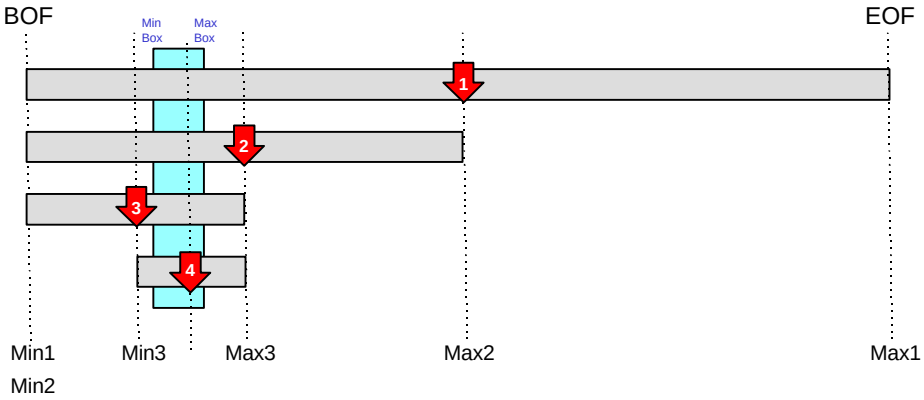


Figure 5-14: The binary search method

This binary file read control is based on a directly accessible and sorted file by the spatial location code. Consequently it is possible that there is more than one Y-value for one X-value. For this reason this binary file read control is more than just an ordinary binary search. This file read system is composed of three steps.

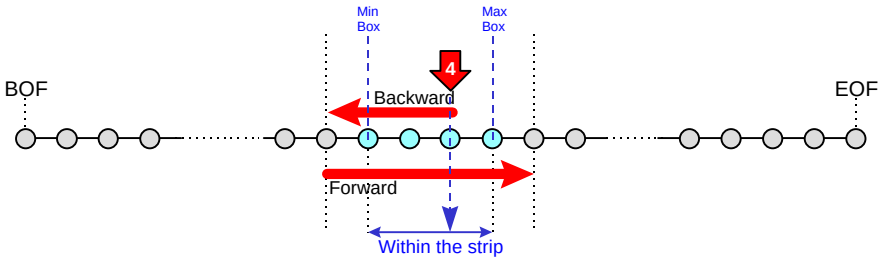


Figure 5-15: How to find the starting point

Using the binary search method, as illustrated, we find in a few splits, a record with an X-value within the permitted interval

Then (see above) the sequence is read backwards just over the minimum.

From that position the sequence is read forwards until the maximally permitted x-value is passed.

In this way all of the records are passed within the interval. The next question is: how do you check whether an element is within the box or not? It was for this purpose that ‘the Catch-function” was developed, which will be explained below.

5.6 The catch-functionality

The catch-function allocates the interval that has to be read and addresses the problem of whether an element is located within a box with a given direction around a given point or not.

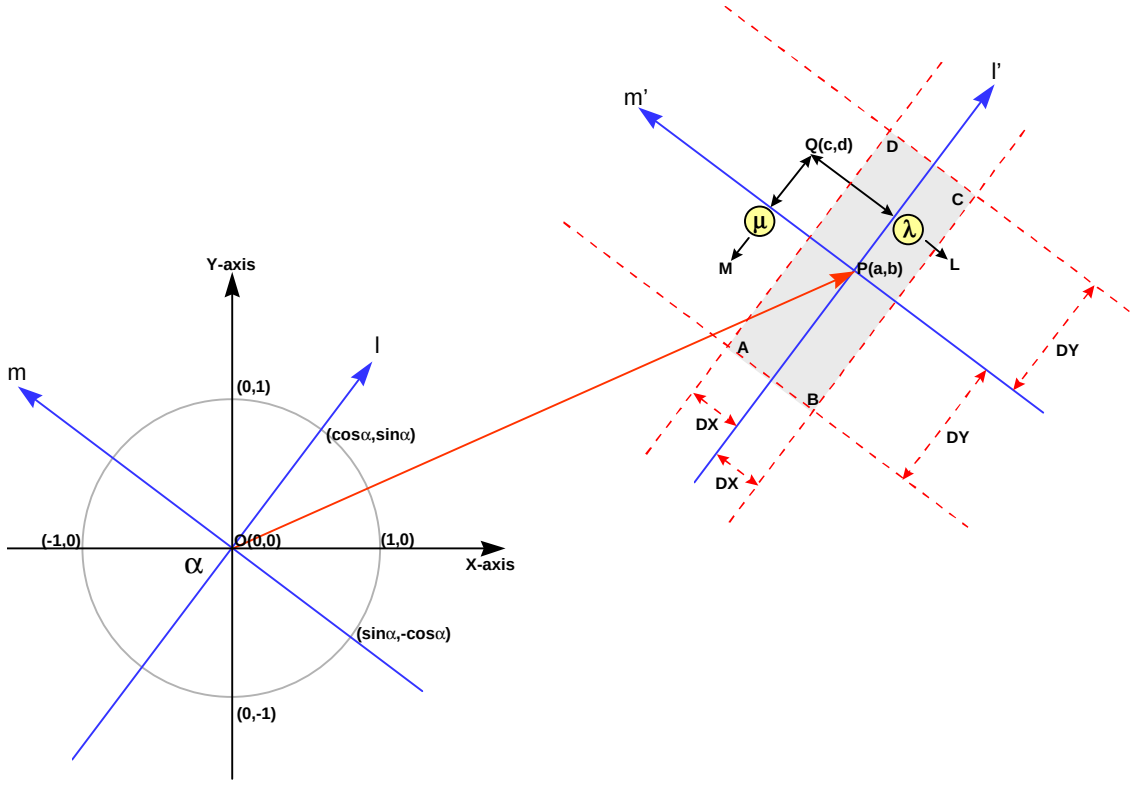


Figure 5-16: The catch-box construction

We assume a box ABCD defined by DX,DY, the angle α around a point P(a,b). In what conditions does point Q(c,d) lie within the box. That is easy to see: namely if the distance to line l' is less than DX (d(Q,l')<DX) and the distance to line m' is less than DY (d(Q,m')<DY).

Vector representation of the line l': $\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a \\ b \end{pmatrix} + \lambda . \begin{pmatrix} \cos\alpha \\ \sin\alpha \end{pmatrix}$

Vector representation of the line m': $\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a \\ b \end{pmatrix} + \mu . \begin{pmatrix} \sin\alpha \\ -\cos\alpha \end{pmatrix}$

For line l' holds: There is a point L so that QL=d(Q,l) (for certain l)

For line m' holds: There is a point M so that QM=d(Q,m') (for certain m)

If LQ ⊥ m' then: direction vector of LQ ⊥direction vector m' or the scalar product equals zero.

$$\begin{pmatrix} c - a - \lambda.\cos\alpha \\ d - b - \lambda\sin\alpha \end{pmatrix}.\begin{pmatrix} \cos\alpha \\ \sin\alpha \end{pmatrix} = 0 \Leftrightarrow c.\cos\alpha - a.\cos\alpha - l.\cos^2\alpha + d.\sin\alpha - b.\sin\alpha - l.\sin^2\alpha = 0 \Leftrightarrow$$

$\Leftrightarrow$ $l.(\sin^2\alpha + \cos^2\alpha) = (c.- a).\cos\alpha + (d - b).\sin\alpha \Leftrightarrow$ $\lambda.= (c - a).\cos\alpha + (d - b).\sin\alpha$. (1)

, while, $\sin^2\alpha + \cos^2\alpha = 1$.

Similarly: $\mu.= (c - a).\sin\alpha - (d - b).\cos\alpha$. (2)

The co-ordinates of L and M are now known

L($a + \lambda.\cos\alpha$, $b + \lambda.\sin\alpha$)

and $QL^2 = (a - c + \lambda.\cos\alpha)^2 + (b - d + \lambda.\sin\alpha)^2$. (3)

M($a + \mu.\sin\alpha$, $b - \mu.\cos\alpha$)

and $QM^2 = (a - c + \mu.\sin\alpha)^2 + (b - d - \mu.\cos\alpha)^2$. (4)

If $QL^2 < DX^2$ and $QM^2 < DY^2$ then point Q lies within BOX (ABCD) (5)

Equations (1) to (5) should later be used to build and to program the functions concerned.

5.7 Achievements

By concatenating the X- and Y co-ordinates we get a spatial code.

Sorting by the spatial code provides a spatial ordering the so-called ZigZagChain.

The extremities of the box determine which strip must be read in order to find matching elements.

While the Binary File Read Control efficiently finds the strip for reading the strip sequence, the Catch-functionality decides whether a strip element is located within the box or not.

5.8 References

[1] Burrough, P.A., 1985. *Principles of Geographical Information systems for Land Resources Assessment*, Oxford University Press.

[2] Dahl, O.J., Dijkstra., E.W., and Hoare, C.A.R. 1972. *Structured programming.* Academic Press

6. DATABASE

The purpose of a map is, to give a momentary relevant representation of reality for the user by means of graphic abstraction. Whereas in the past a host of designers were needed our digital era provides a simpler way of creating the required map. Until recently, analogue maps were used in the railway-field. All of the data, for the user needed were present, and the skilled railway employee was able to build up the information with the aid these data, in his/her mind. The ability to do this varies from person to person. However, a computerised process is able to pick up and present a maximum amount of information and to produce the same information under the same circumstances. The question now is: How we can breathe more life into these static data.

The answer is provided here in 10 steps.

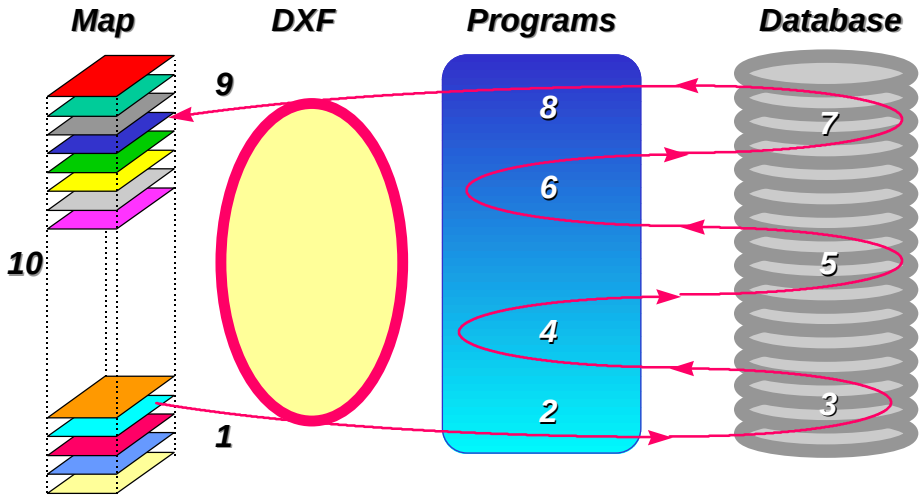


Figure 6-1: The enrichment process

6.1 The steps

The map was built up by digitising aerial photographs in combination with land-surveying and a lot of design conventions in a CAD-environment. We know that every CAD-program has DXF out/in utilities.

6.1.1 DXFout

Here we use the DXF-out utility, and everything which the map can tell us is written in ASCII-text format.

6.1.2 Extract elements

We now need a program to extract the relevant data (elements).

6.1.3 Storage 1

Store the result of (2) in file(s)

6.1.4 Catching

Then we need a program to read the file(s) from (3), which will also enable us to group the elements concerned, using the catching algorithm appropriate elements.

6.1.5 Storage 2

Again storage of these data.

6.1.6 Attributes

When we talk about objects, we also talk about attributes. So we need a program to equip objects with attributes.

6.1.7 Storage 3

We store the results again.

How can we materialise these objects in the map, in a free layer. Write a DXF file!

6.1.8 Objects

The CAD-program puts the objects in a free or any other layer via the DXF-in utility.

6.1.9 DXFin

Now the objects are within the CAD-environment and it is possible to present them without or in combination with other layers.

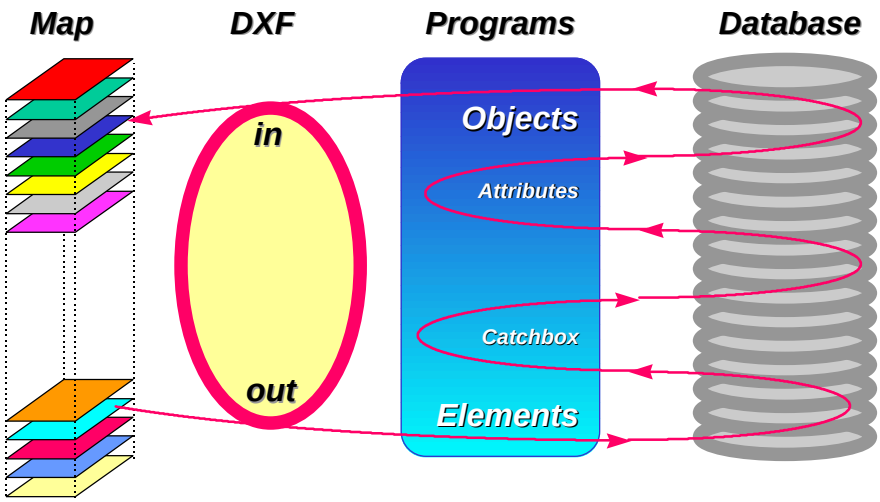


Figure 6-2: From elements to objects

6.1.10 Presentation

We are now able to present the attributed objects in the added layers

Remark:

6.2 Terms

When we refer to ‘a Database’ at this point, we have in mind a collection of related files, and that is not the same as a relational database or even a relation database management system.

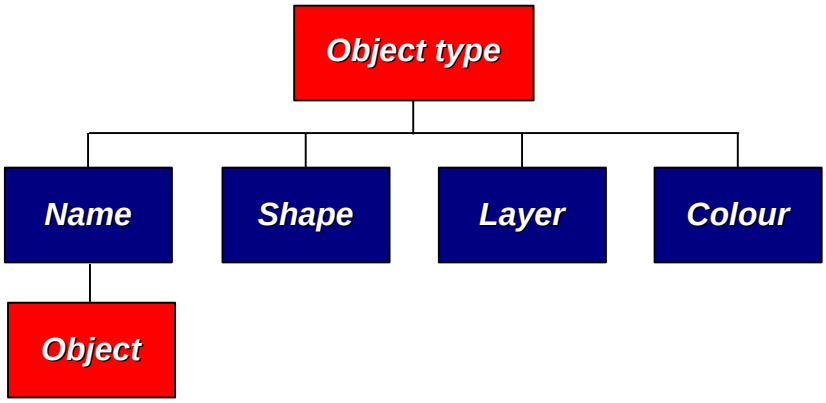


Figure 6-3: A global data model

The question now is, how to organise the storage of 'The Objects' in an adequate manner. Adequate means a minimum amount of of redundancy (via normalisation) and in related tables. First of all we have to examine the properties determining an object type.

Object types must have an identifying name, be represented in a certain shape, in an allocated layer and a chosen colour. That means: for each object type there must be a name, a shape, a layer and a colour. chosen. Each object has no, one or several attributes.

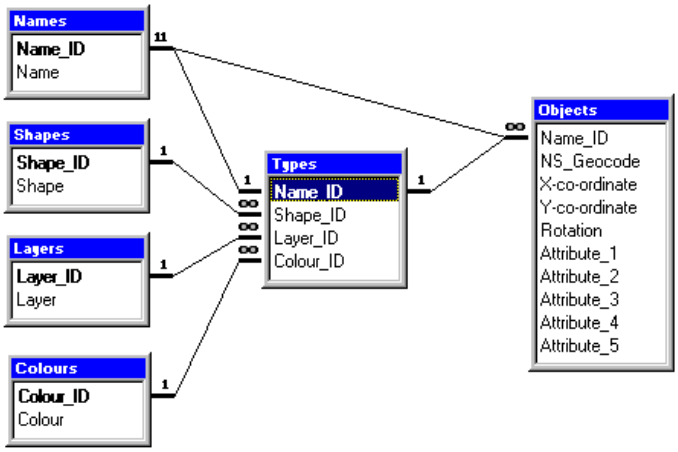


Figure 6-4: The data model in MS Access

6.3 Tables

We can represent this in related tables. Looking at figure 7-4, we can conclude that the table of object types is composed of the identifiers of the tables: Names, Shapes, Layers and Colours. The choice from the Names collection has to be unique! The choice of the shape, layer, and colour does not need not to be unique. Each object type belongs to a row in the accompanying table of objects. The following we will exemplify this.

Names: Tabel

	Name_ID	Name
	1	Point
	2	Sign
	3	Electricity
	4	Gas
	5	Water
	6	Post
▶	0	

Figure 6-5: Names

The tables Names, Shapes, Layers, and Colours are basic for the table Types. In this example there are 6 object types, which have 3 possibilities with regard to their shape, 9 possibilities with regard to layer, and 7 with regard to colour.

This means that there are 3 x 9 x 7 alternatives for the first type. In principle there are also 3 x 9 x 7 alternatives for the other types, but for the sake of recognisability it is better to chose as many as different representations as possible.

Shapes: Tabel

	Shape_ID	Shape
	1	Square
	2	Circle
	3	Diamant
▶	0	

Figure 6-6: Shapes

In a CAD-environment is a solid (filled or not) or a circle easy to handle for this purpose.

Layers: Tabel

	Layer_ID	Layer
	1	One
	2	Two
	3	Three
	4	Four
	5	Five
	6	Six
	7	Seven
	8	Eight
	9	Nine
▶	0	

Figure 6-7: Layers

It is advisable to place different object types in different layers, but it is conceivable that the choice is limited or that different object types are related in such a way that it would be better to keep them in one layer. For example sewers and sewer drains will usually appear in the same layer.

It is preferable to use distinctive colouring of the object types.

Colours: Tabel

	Colour_ID	Colours
	1	Red
	2	Lime
	3	Yellow
	4	Blue
	5	Fuchsia
	6	Gray
	7	Purple
▶	0	

Figure 6-8: Colours

The objects types table is threfore a collection of identifiers referring to the related tables.

Types: Tabel

	Name_ID	Shape_ID	Layer_ID	Colour_ID
	1	2	3	2
	2	3	4	6
	3	1	6	4
	4	3	7	4
	5	1	5	1
	6	1	1	2
▶	0	0	0	0

Figure 6-9: Types

Objects: Tabel

	Name_ID	NS_Geocode	X-co-ordinate	Y-co-ordinate	Rotation	Attribute_1	Attribute_2	Attribute_3	Attribute_4	Attribute_5
	3	123	123232	543211	10	SEP	342	78		
	1	231	245276	565433	2	231B	1:7	3-jan-1994	NMR	
	6	321	200066	453212	0					
	5	432	231454	342334	30	4W	20	120	Waterloo	87
	1	453	123457	345678	15	453A	1:9	4-mrt-1996	Rail-Experts	
	2	456	231457	432191	45					
	4	532	234323	453220	10	10R	230	Gazog	3-mai-89	
	3	564	234568	432156	345					
▶	0	0	0	0	0					

Figure 6-10: Objects

The objects table can be one table, while the Name_ID refers to the related object type.

That is, every row belongs to an object, and the rows with the same Name_ID belong to the same object type.

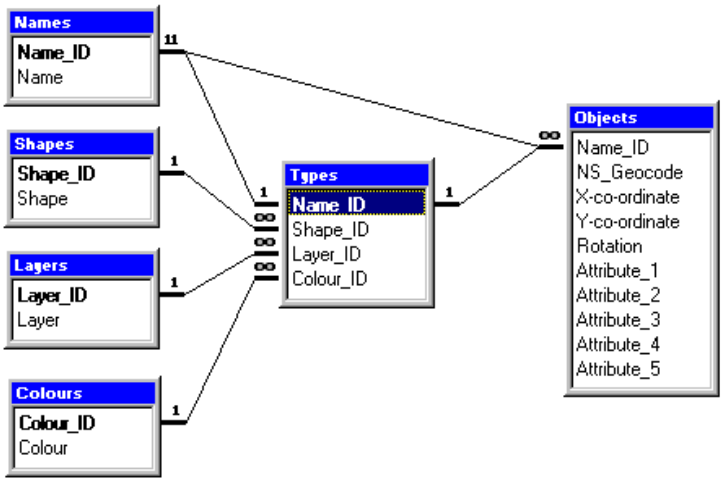


Figure 6-11: The Entity Relation Table

This collection of tables is related in the way as sketched above.

6.4 Queries

When the whole is built up in a RDMS-environment (here MS Access), it is possible to make queries as shown in *Figure 6-12*

Veld:	Name	Shape	Layer	Colour	NS_Geocode	X-co-ordinate	Y-co-ordinate
Tabel:	Names	Shapes	Layers	Colours	Attributes	Attributes	Attributes
Sorteervolgorde:					Oplopend		
Weergeven:	☑	☑	☑	☑	☑	☑	☑
Criteria:					>300 En <500		
Of:							

Figure 6-12: A query

In simple words: Show the fields: Name, Shape, Layer, Colour, NS_Geocode, X-co-ordinate,

Query1: Selectiequery

Name	Shape	Layer	Colour	NS_Geocode	X-co-ordinate	Y-co-ordinate
Post	Square	One	Lime	321	200066	453212
Water	Square	Five	Red	432	231454	342334
Point	Circle	Three	Lime	453	123457	345678
Sign	Diamant	Four	Gray	456	231457	432191

Figure 6-13: The result of the query

and Y-co-ordinate for all off the available records on condition that the NS_Geocode is between 300 and 500. The result is presented in *Figure 6-13*.

7. THE CASE STUDY

7.1 Introduction

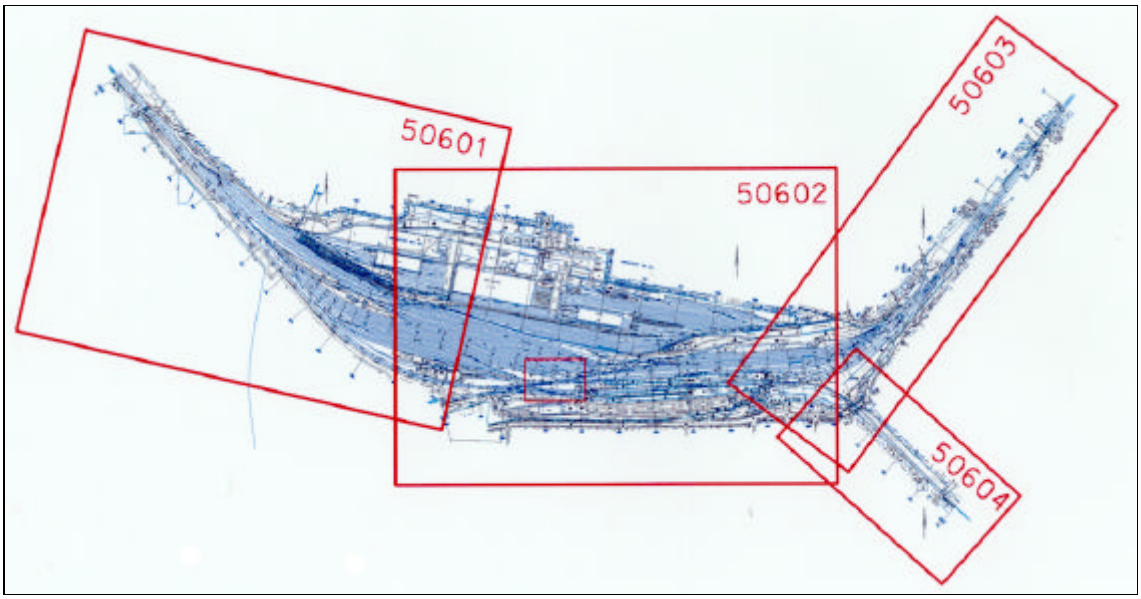


Figure 7-1: Geocode 506 - Amersfoort

This chapter explains the Object Composing Process using as a case study the railway yard in Amersfoort in the middle of the Netherlands. However, it is very recommendable to do the case with the enclosed DEMO disk at the back. And, please, do not forget to read the instructions of *9.5*. The situation in Amersfoort is rather complicated and therefore useful for testing.

Amersfoort is represented by four map sheets i.e. geocode 506 sheets 01, 02, 03 and 04. The process consists of a number of stages:

- Cell Scanning

This identified the used library elements to be used.

- Spatial Coding

Collects the relevant data, provides them with a spatial code, which makes the data linearly accessible, which is in turn conditional for significant reading by a computer system.

- QuickSort

The sorting of the elements by spatial code is the last step in the preparation for object composing.

- Object Catching

Before starting the catching process an overview of the data traffic and storage structure is given.

- Common Point

The explanation for catching the common rail point (001W) by means of flexible boxes is the next, them to be discussed.

- English Point

This particular type demands a special kind of treatment because of the absence of a frontweld, which defines the box direction. This has to be found now in a different way.

- Objects

Tables show the complete as well as the incomplete caught objects. The incomplete parts are marked yellow (...)

- Problem Points

Frontwelds and mathematical points of incomplete rail point are presented in tables here.

- The Final Results

By comparing that which is found by the CellScan program and composed by the Catching program it is possible to present the final results.

- Solids Generator

This program produces filled solids in a dxf-file, which represents the status of the points as regards completeness.

- Correction Boxes

These boxes shows that their is something wrong within their boundaries. This can be useful for draughtsmen.

- Objects Shower

This program can present the objects found on the screen.

7.2 Technical Data

The following technical specifications were employed in the procedure.

7.2.1 Hardware

PC486 100 Mhz processor, with 32 Mb RAM, and 1+.4 Gb, Hard disks

HP DeskJet 694C

HP ScanJet 5P

7.2.2 Operating system

Windows 95

7.2.3 Graphics

AutoCAD 10

MicroStation 95

Microsoft PowerPoint 7.0

7.2.4 Scanning

HP Picture Scan

7.2.5 Word processing

Microsoft Word 7.0

7.2.6 Programming

Microsoft GW-BASIC 3.23

7.2.7 Compiling

Microsoft Quick BASIC 4.5

7.2.8 Structure

The programs are structured but not fully. A lot of remarks are added in the source-file. The working of the programs will be explained later on.

7.2.9 Menu Generator

A program, which makes it possible to run the desired the programs. The source code can be found in the Appendices. To get the required results we have to go through the menu from top to bottom. It is possible to link all off the programs together, but in this way it is easier to acquire a step by step insight into the working of the programs.

7.2.10 Storage

All of the *.BAS, *.EXE, IN- OUT- and TEXT files relating to the case will become available after installation from the enclosed disk in the directory C:\ANALYSER. The complete TREE can be seen in *9.1*

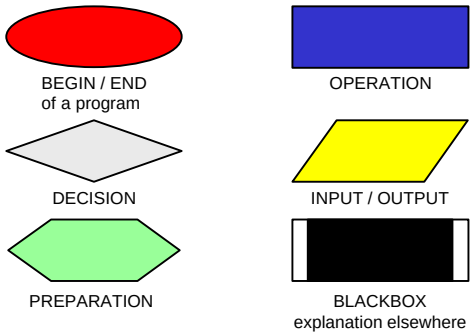


Figure 7-2: Flow Chart Symbols

7.2.11 Flow Charts

With the aid of the Flow Chart Symbols in *Figure 7-2* it is possible to represent, and to make clear the working of a program in general terms.

7.3 Cell Scanning

This program scans the four files of geocode 506 to identify NS specific library elements for this case study. These library elements can be used as components for the objects to form later on. For that reason is it useful to make an inventory of these elements. A number of elements are used in this case and yellow marked. See therefore *7.3.13* and *7.3.14*. These figures are used later on in *0* in the determination of the quotient of what is found and what could be found as indicator of the final result.

7.3.1 Library elements.

A CAD program is made up of a number of parameterised standard drawing routines, which after the parameters have been filled in produces generic entities such as lines, circles and so on. Each institute has a number of branch specific affairs which have been drawn again and again. Such specific elements can be laid down and called up again. We call such a collection, a library. These library elements are in turn a parameterised drawing routine, which after have been filled in the parameters, produce branch specific entities. NS G&I has defined 40 objects which were entered in a library. Library elements within AutoCAD/DXF are called ‘blocks’ and ‘cells’ within MicroStation ‘cells’. These blocks or cells can be called up by name, and inserted after the parameters for the co-ordinates of the insertion point, rotation, scaling, colour, line thickness and so on have been entered. In the Netherlands we have 3500 kms of railroads. We can assume that a map is drawn for every kilometre , and that this is stored in a design-file. Each of the design-files can be converted to a DXF. These inserted blocks or cells can be found in the Entity Section of the DXF as an ‘insert’. This gives us the possibility to search for ‘inserts’ within a DXF, and to collect information about objects. For an explanation of the terms used see *Figure 7-4*.

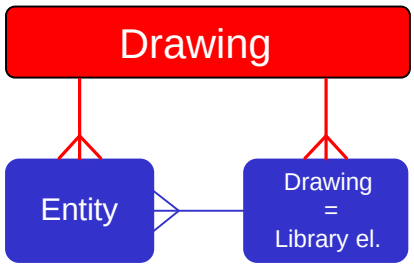


Figure 7-3: Drawing Structure

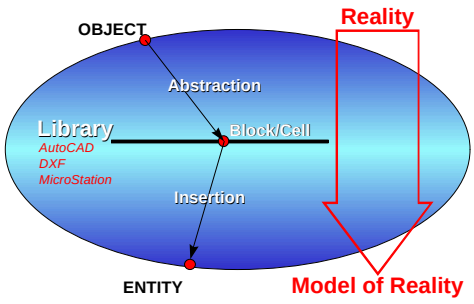


Figure 7-4: Context of Terms

7.3.2 Concatenating

Each cell has a name of up to five characters. When we place right-justified each of the forty cell names in a field of five spaces and concatenate the forty fields, then we get a word (string) of 40 x 5 = 200 characters. In BASIC, the maximum possible length of a text string is 255 characters. When the word-length exceeds 255, then the field has to be split into fields no greater than 255 characters.

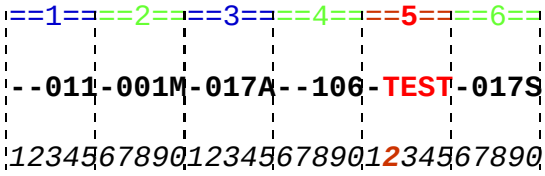


Figure 7-5: The In-String-Function

7.3.3 The In-String-Function

One of the BASIC functions is the In-String function which checks the presence of a group of characters within a word. It gives a zero when the group is not present. Otherwise it gives the position of the first character of the group within the word. In *Figure 7-5* we see that it has to give 22. In *Figure 7-5*, for example the word TEST is the fifth cell, because 22 divide by 5 is greater than 4.

7.3.4 The Geocode

The forty library elements are railway specific. They can tell us a great deal about the Rail Infrastructure in general, by geocode or even by sheet. In this case the geocode 506 (Amersfoort) is examined with the four sheets 50601, 50602, 50603, and 50604. The Cell Scanner Program scans the frequency of cells, extracts and gathers information about the cells within the box, and the box itself , which forms the boundaries of the map sheet concerned.

7.3.5 Identification

By typing in one of the available geocodes, in this case 506, the program collects the dxf-files concerned, and gathers information for the sheet. This information can be found in the section HEADER of the DXF. The information about extremities, rotation angle, co-ordinates of the insertion point and the paper output format can be found in the first four lines of the specific reports of the four files as shown in *7.3.13*

7.3.6 Counting

Information about the inserted cells can be found in the section ENTITIES. After each INSERT the In-String-Function returns cell was inserted, after which the cell counter is raised.

7.3.7 Totalling

The program keeps a score per sheet and the total, which is presented in a total report in *7.3.14*.

7.3.8 Storage

Meanwhile information about the cells is stored in c:\analyser\cells\cells506.txt. Part of the file is shown below. The names of the fields speak for themselves

GEOCODE	NUMBER	X-coordinate	Y-coordinate	ROTATION
506	30	153076.165	463100.361	168
506	16	152696.364	463265.632	0
506	20	152694.971	463266.869	0
506	14	152384.409	463541.561	-9
506	5	152441.364	463540.482	131
506	2	152644.374	463285.086	0
506	36	152629.422	463262.595	125
506	2	152788.625	463145.375	0

7.3.9 Results

With the configuration of *7.2* the program scans geocode 506 for all the rail specific library elements in less than 15 seconds and prints out the reports referred in *7.3.13* and *7.3.14* in less than one minute.

7.3.10 Compound Object

What to do with a compound object? A common rail point is made up of two CELL INSERTS and two TEXT INSERTS. How do we bring them together? The answer is: catch them.

7.3.11 Source

The source can be found in *9.2.2*

7.3.12 The Scan Process

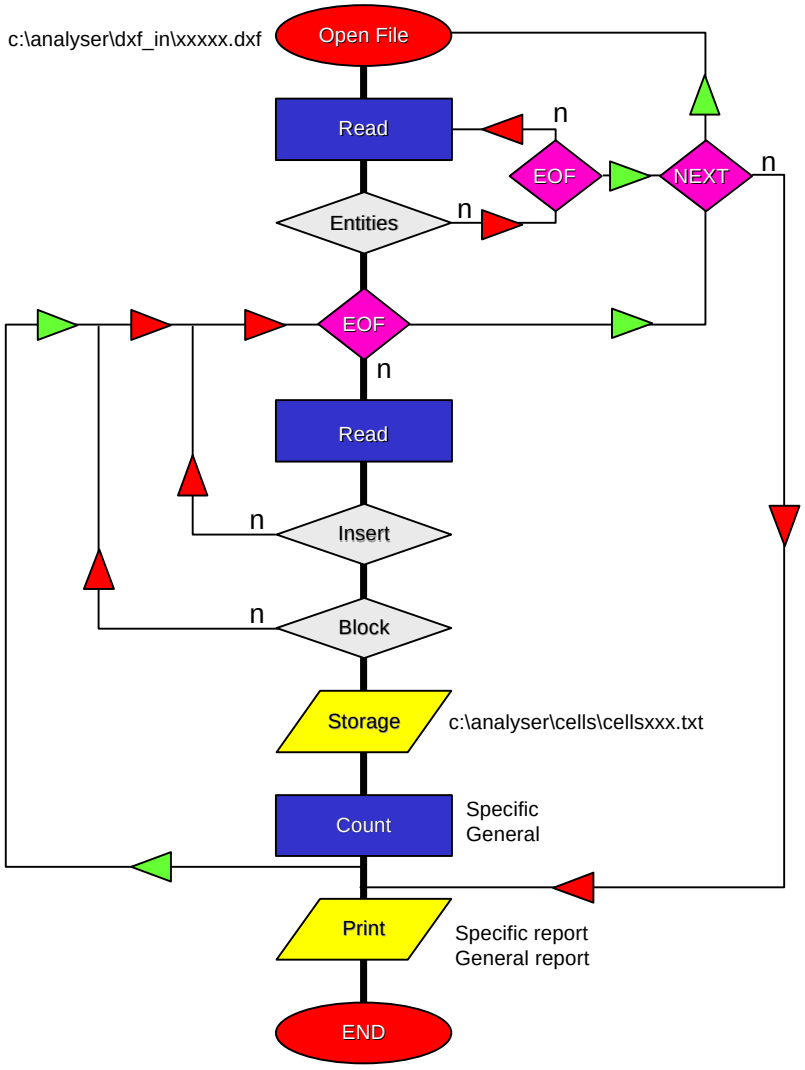


Figure 7-6: FlowChart of CellScan

7.3.13 Specific Report

7.3.13.1 \analyser\dxf_in\50601A.DXF

```
ExtmaxX : 153408.759    ExtmaxY : 463790.311
ExtminX : 152051.621    ExtminY : 462653.140
Xinsert : 152081.939    Yinsert : 462971.886
Format  : A0            Rotation: -13.999941
  011   001M    017A    106     105     017S    017L    001W    001K    001H
 16     72      20      3       5       0       76      81      0       0
 001E   002A    213     214     215     002E    002T    002S    002D    001L
 12     2       0       3       0       27      4       10      21      15
 002G 001SP     209     210     201     202     223     203     204     224
 0      0       10      0       0       0       14      7       2       26
  205   206     207     208     216     225     212     001     211     002H
 0      0       4       0       15      17      0       0       1       12
```

7.3.13.2 \analyser\dxf_in\50602A.DXF

```
ExtmaxX : 154272.668    ExtmaxY : 463398.369
ExtminX : 153083.668    ExtminY : 462557.369
Xinsert : 153113.668    Yinsert : 462562.369
Format  : A0            Rotation: 0.000000
  011   001M    017A    106     105     017S    017L    001W    001K    001H
 8      168     48      3       16      0       144     158     0       0
 001E   002A    213     214     215     002E    002T    002S    002D    001L
 25     0       0       0       0       34      5       8       46      52
 002G 001SP     209     210     201     202     223     203     204     224
 0      0       3       1       0       1       73      54      13      43
  205   206     207     208     216     225     212     001     211     002H
 28     21      16      10      52      5       0       0       1       10
```

7.3.13.3 \analyser\dxf_in\50603A.DXF

```
ExtmaxX : 155025.119    ExtmaxY : 463774.894
ExtminX : 153951.358    ExtminY : 462585.978
Xinsert : 154293.078    Yinsert : 462612.459
Format  : A3X4          Rotation: 51.078308
  011   001M    017A    106     105     017S    017L    001W    001K    001H
 12     68      22      0       12      0       34      32      0       2
 001E   002A    213     214     215     002E    002T    002S    002D    001L
 1      6       0       6       0       14      12      11      15      0
 002G 001SP     209     210     201     202     223     203     204     224
 0      0       0       2       1       0       29      66      5       76
  205   206     207     208     216     225     212     001     211     002H
 35     3       1       1       11      5       0       0       1       18
```

7.3.13.4 \analyser\dxf_in\50604A.DXF

```
ExtmaxX : 154754.631    ExtmaxY : 462920.017
ExtminX : 154126.462    ExtminY : 462288.170
Xinsert : 154730.038    Yinsert : 462514.228
Format  : A4X3          Rotation: 134.498537
  011   001M    017A    106     105     017S    017L    001W    001K    001H
 0      1       0       0       0       0       0       0       0       0
 001E   002A    213     214     215     002E    002T    002S    002D    001L
 0      2       0       2       0       3       1       0       0       0
 002G 001SP     209     210     201     202     223     203     204     224
 0      0       1       0       0       0       4       18      0       14
  205   206     207     208     216     225     212     001     211     002H
 6      0       2       2       2       0       0       0       1       2
```

7.3.14 General Report

1	011	bovenleidingpaal in RD	36
2	001M	bovenleidingpaal	309
3	017A	ankerblok	90
4	106	ontspoorinrichting	6
5	105	stootjuk	33
6	017S	stootbalk	0
7	017L	voorlas wissel	254
8	001W	math. punt wissel	271
9	001K	math. punt kruising	0
10	001H	math. punt half engels wissel	2
11	001E	math. punt engels wissel	38
12	002A	ahob	10
13	213	aki	0
14	214	knipperlicht	11
15	215	andreaskruis	0
16	002E	E-kast	78
17	002T	baan-telefoon	22
18	002S	sein	29
19	002D	dwergsein	82
20	001L	lichtmast	67
21	002G	gaskast of wisselverwarming	0
22	001SP	spanningspaal	0
23	209	water klein	14
24	210	water groot	3
25	201	keiverharding normaal	1
26	202	keiverharding groot	1
27	223	klinkerverharding klein	120
28	203	klinkerverharding normaal	145
29	204	klinkerverharding groot	20
30	224	tegelverharding klein	159
31	205	tegelverharding normaal	69
32	206	tegelverharding groot	24
33	207	asfaltverharding normaal	23
34	208	asfaltverharding groot	13
35	216	betonverharding	80
36	225	onverhard (puin, grind, zand, etc.)	27
37	212	moeras	0
38	001	dukdalf	0
39	211	noordpijl	4
40	002H	hectometerpaal	42

7.4 Spatial Coding

This has already has been referred to in *7.3.10*. We have to find a solution for compound objects. One of the most complicated objects is the common rail point, which consists of a cell named 001W (mathematical point), a cell named 017L (front weld), text between the two cells (point number) and text of the type “1:” (ratio). These INSERTs can be found in the section ENTITIES.

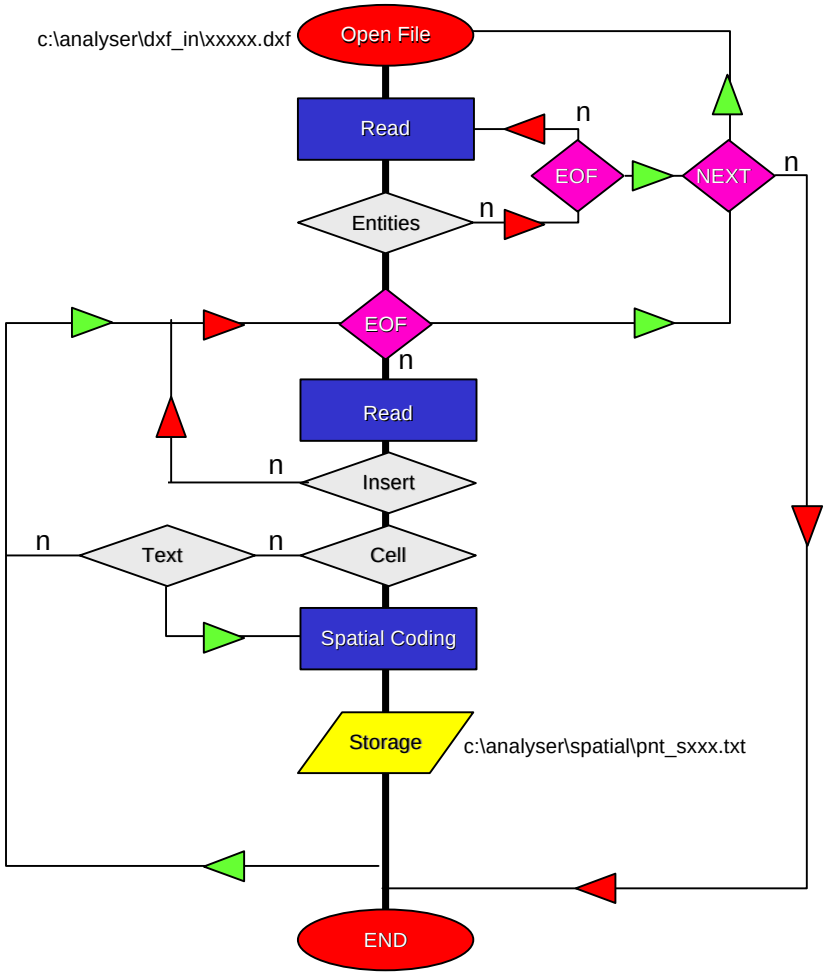


Figure 7-7: Flow Chart Spatial Coding

In our case the files c:\analyser\dxf_in\5060Xa.dxf. (X from 1 to 4) have to be opened in succession. The procedure always checks an End Of File (EOF), and then continues to read until it finds ENTITIES, after which it starts to look for all the INSERTs.

Check the type, CELL or TEXT. Concatenate the X and Y co-ordinates as described in 6.1.

Store the result in c:\analyser\spatial\pnt_s506.txt. Above you can see the result in a part of the file. The spatial code is the key for the ZigZagChain Method as described in *5.1.2*. You can also see that up until now the file is unsorted. Remarks are to the right of the line.

7.4.1 Purpose

With the spatial code as key it is now possible to sort the elements, so that the elements become linearly accessible. This is conditional for effective reading by a computer system.

```
Spatial Code X-coord.   Y-coord.     Type  Rot REMARKS
152562463416,152562.457,463415.767, 001W,   0 mathematical point
153047463244,153047.090,463244.139,  1:7, 349 ratio
153020463250,153020.418,463249.557,  408, 349 point number
153031463247,153030.952,463247.064, 001W,   0 mathematical point
153019463249,153018.882,463249.308, 017L, 169 front weld
153114463231,153113.663,463231.493,  1:7, 349 ratio
153085463237,153085.309,463236.977,  409, 349 point number
153096463234,153096.404,463234.462, 001W,   0 mathematical point
```

7.4.2 Source

see *9.2.3*

7.5 QuickSort

This programs produces, a file sorted by the ‘spatial code’ using Hoare’s algorithm see *5.2*. As a result you get a spatial ordering, which is linear, the so-called “ZigZagChain”. That chain makes sequentiay reading of the elements of the file possible. A step which is of vital importance to the composing process.

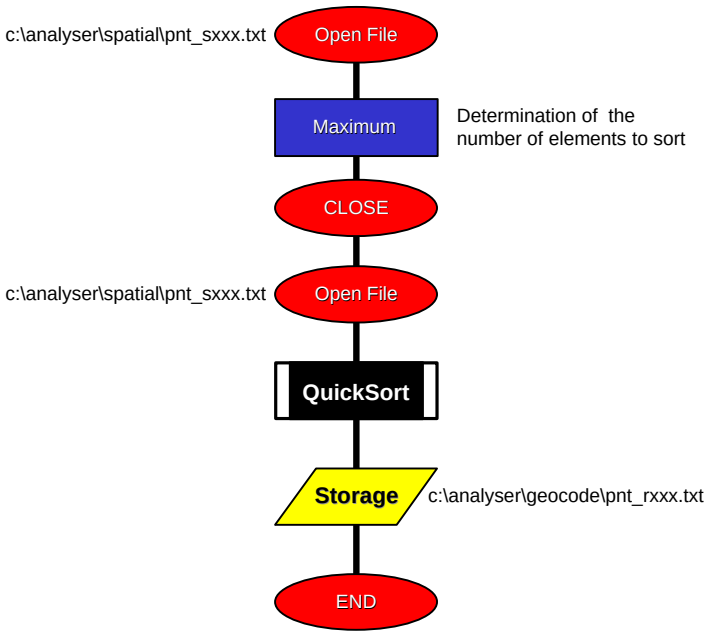


Figure 7-8: The QuickSort Flow Chart

7.5.1 The Key

Key of the sort is the field “Spatial Code” of the file c:\analysis\spatial\pnt_s506.txt.

7.5.2 The Total Number

It is necessary to know the total number of records to be sorted. This requires reading the file in its entirety.

7.5.3 Storage

The sorted file has been stored as c:\analyser\geocode\pnt_r506.txt

7.5.4 Source

see *9.2.4*

7.6 Object Catching

The Object Catching Program is the main part of the whole procedure. It allows objects to be composed from spaghetti data in a CAD system. An overview of the process is given by *Figure 7-9*, which shows the context of the program, and the files used to reach positions 1, 2 or 3.

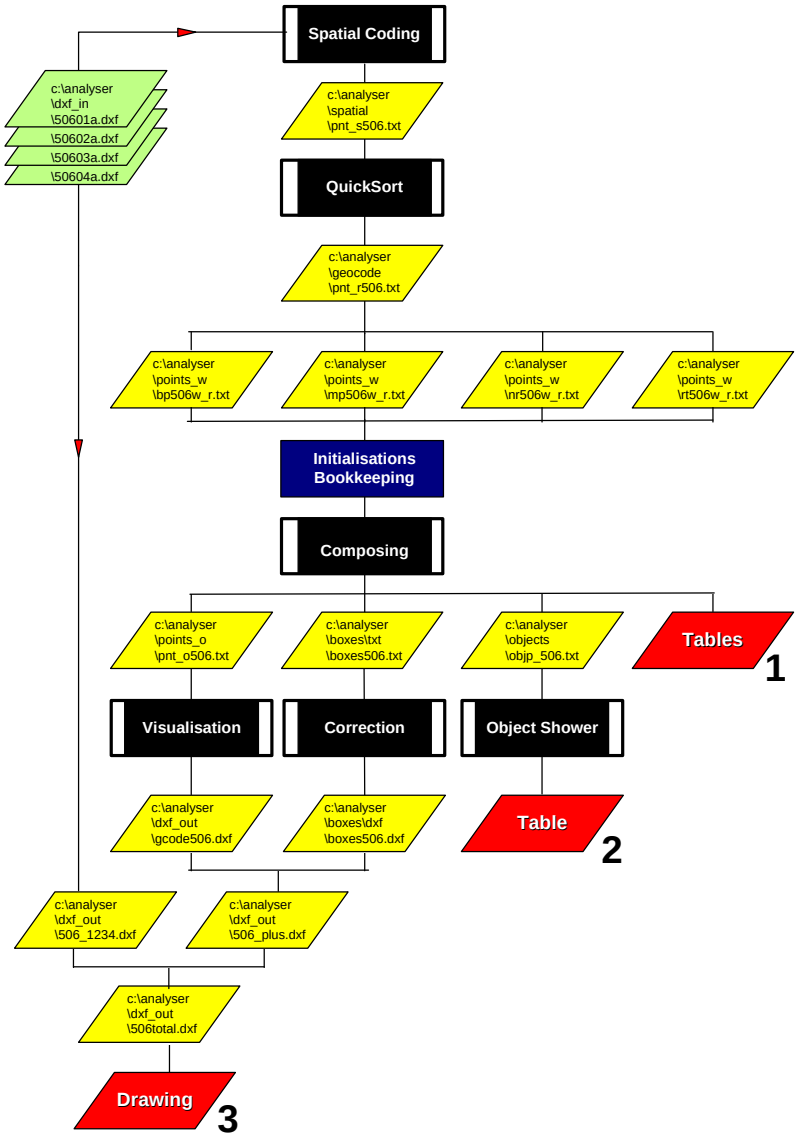


Figure 7-9: The Context of Object Catching

Under 1 (*7.9*) is shown what has been caught. Table 2 shows the records of the objects obtained on the screen (*9.2.8*), and 3 shows the objects as solids (*0*) just like the correction boxes of the problem points (*7.13*)

7.6.1 Splitting

The spatially coded and sorted file c:\analyser\geocode\pnt_r506.txt can now be split up into four files which contain the specific elements to build the objects, i.e. the cells frontwelds, and mathematical points, and texts for point numbers and ratios.

bp506w_r.txt for the frontwelds

mp506w_r.txt for the mathematical points.

nr506w_r.txt for the Point numbers

rt506w_r.txt for the ratios.

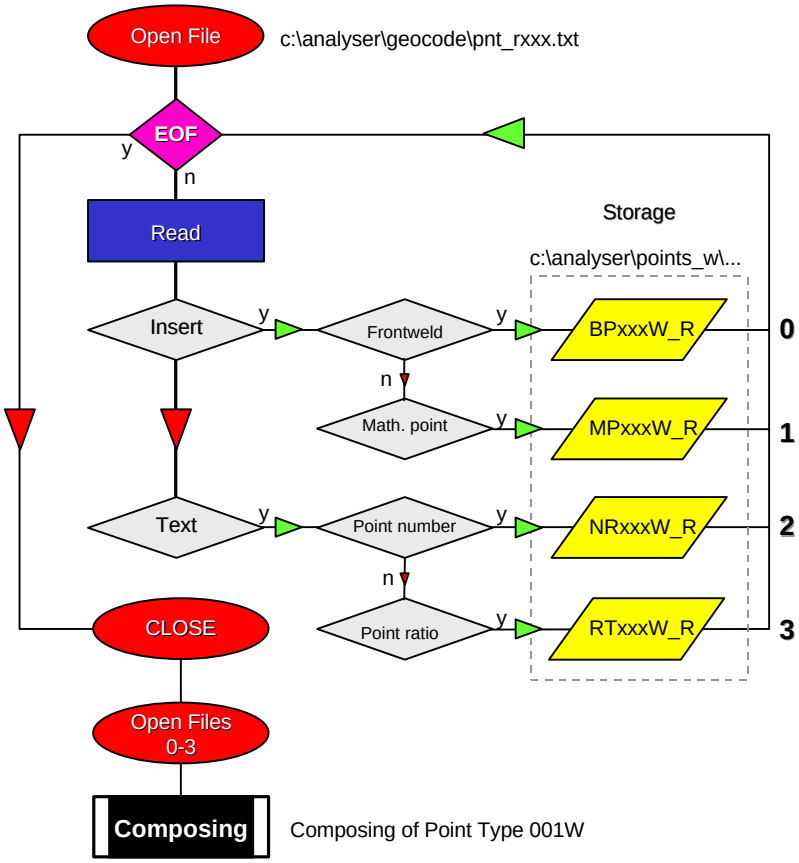


Figure 7-10: FlowChart Splitting

7.6.2 Bookkeeping

The files are stored in c:\analyser\points_w and provided with a bookkeeping field. This field shows an “A” when the record is available and a “U” when the record has already been used for an object, and is subsequently not longer available for object composing.

7.6.3 Source

see *9.2.5.*

7.7 Common Point

In the following pages we will set out how the process of object composing of a common point (type 001W) works.

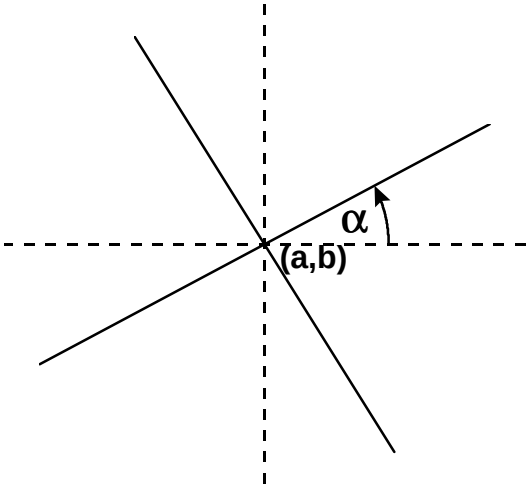


Figure 7-11: Any Rotation

7.7.1 Box transformations

A BOX is defined as a library element. In this case as a quadrangle around the origin. The question is what are the co-ordinates of the angular point after a transformation which consist of a rotation, a scaling in x as well as y direction and a translation over a certain distance and direction.

Through the transformation the quadrangle becomes a rectangle with angular points which can be characterised as Left Bottom (LB), Right bottom (RB), Left Top (LT)and Right Top (RT). The rotation determines the position of the angular point, which in turn determines the extremities.

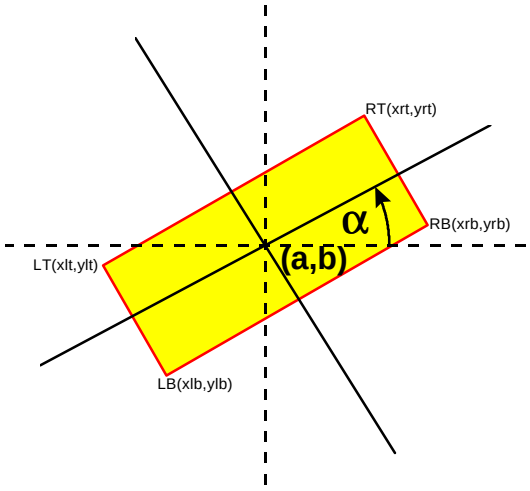


Figure 7-12: Angular points of a box

Figure 8-13 furthermore also tells us about the extremities of the X-co-ordinate values of the angular points of a box. When the minimum is found the maximum is also known or the other way around. If the maximum X-co-ordinate is XRB the minimum is XLT

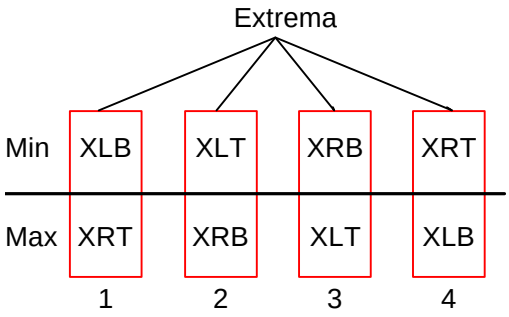


Figure 7-13: Extremities of a box

7.7.2 Boxes

We consider a file within BOF (Begin Of File) and EOF (End Of File). The extremities of the X-co-ordinates of the over angle α rotated yellow box round (a,b) determine the part of the file that have to be read. *Figure 7-14* illustrates that the file must be read between Minbox1 and Maxbox1. The candidate object elements are situated in the blue area. The entry of the blue area is found by a binary search (1) (see also *Figure 5-14*), then read back (2) up to or just past the Minbox1 value. Read forward (3).and check the candidate elements (p,q) for location within the yellow box. This principle is applied to all of the boxes in the catching program. This catching algorithm is very efficient, because the relevant part of the file can be read in just a

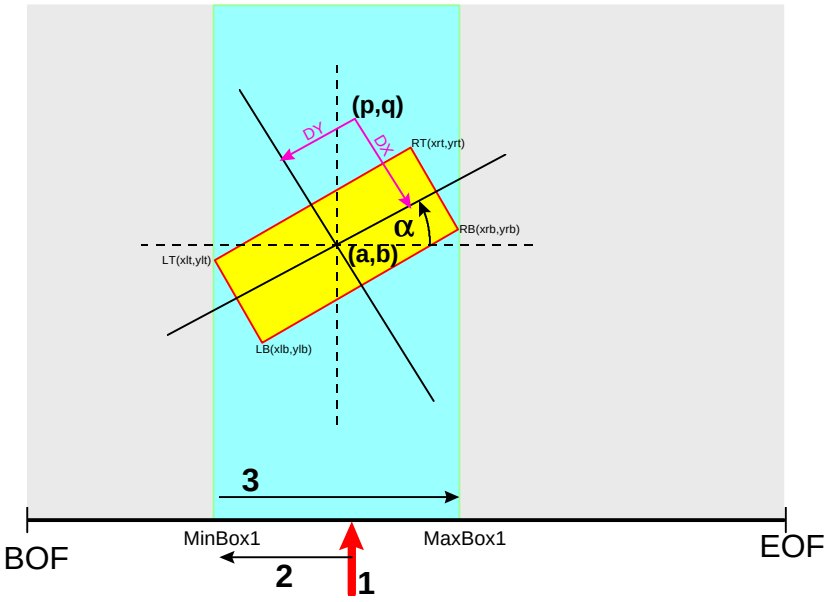


Figure 7-14: Partial File Reading

few steps and one can check whether the elements are located within the box or not..

The next page explains how the boxes are defined and how they are related to each other. Box1 (see also *Figure 7-15*) is aimed at the catching of the mathematical point of a common point with the position and direction of the frontweld of the rail point as a starting-point. When the mathematical point is found Box2 and Box3 can be defined, which are dependent of the position of the mathematical point and the frontweld. A box is defined by means of 5 parameters: The X and Y co-ordinates, the rotation angle α and the distances dx an dy of (a,b) to the sides of the rectangular box. In a general notation Box(a,b,α,dx,dy)

7.7.3 Box1

Box1(a,b,α,1,40); (a,b) is in this case the insertion point of the frontweld with rotation angle α, which is presented by the file c:\analyser\points_w\bp506w_r.txt. An empirical fact is that the matching mathematical point must lie within 40 metres. As the frontweld is at a right angle to the rail direction we can assume that the mathematical point is to be found in a box of 80 by 2 metres positioned round (a,b) under an angle α. Supposing the co-ordinates of the mathematical point are (p,q). Unfortunately the file c:\analyser\points_w\mp506w_r.txt provides nothing about the direction of the insert of the mathematical point symbol. The symbol is a circle and a circle is not sensitive to direction.

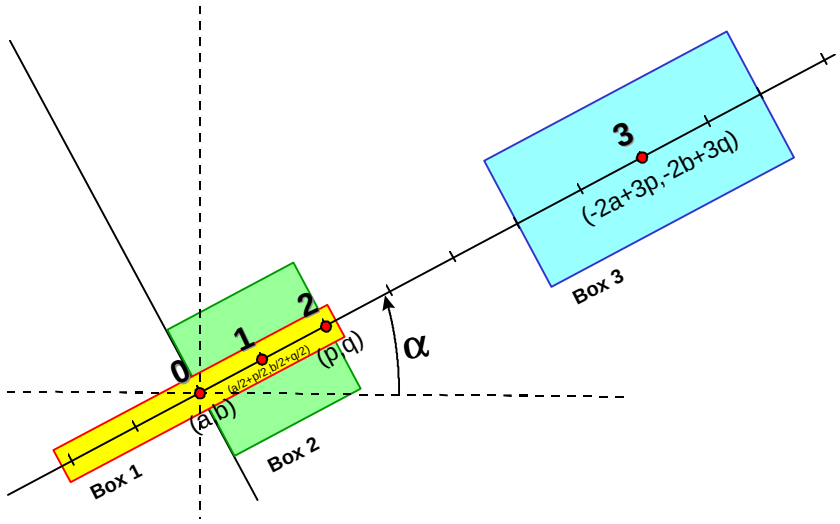


Figure 7-15: Proportions and positioning

7.7.4 Box2

Box2(a/2+p/2,b/2+q/2,α,4,$\sqrt{[(p/2-a/2)^2+(q/2-b/2)^2]}$; The co-ordinates of the centre point are the average of (a,b) and (p,q), while the dx=4 is an empirical fact, and the dy is the distance of (a,b) and (p,q) and found by the application of the Pythagoras' theorem. The candidates for this box are situated in the file c:\analyser\point_w\nr506w-r.txt.

7.7.5 Box3

Box3=(-2a+3p,-2b+3q,α,4,2.$\sqrt{[(p/2-a/2)^2+(q/2-b/2)^2]}$

Now the ratio of the point must be found in file c:\analyser\point_w\rt506w_r.txt.

The centre point of box3 is the result of a point multiplication of the centre point of box2 with the factor -4 from (p,q). The dx remains the same and the dy becomes twice the dy of box2.

These facts are also empirical and based on design conventions within NS G&I.

The co-ordinates of the centre point of box3 can be traced back as follows.

Translate (p,q) to the origin which means decrease the x-co-ordinate by p and the y-co-ordinate by q. Then multiply by factor -4 from the origin and then the first translation in the opposite direction.

Translation to the origin:	(a/2+p/2-p,b/2+q/2-q)=a/2-p/2,b/2-q/2)
Multiplication by -4 of (a/2-p/2,b/2-q/2) =>	(-4.a/2+-4.-p/2,-4.b/2+-4.-q/2)=(-2a+2p,2b-2q)
Translation to (p,q):)=	(-2a+2p+p,-2b+2q+q)=(-2a+3b,-2b+3q)

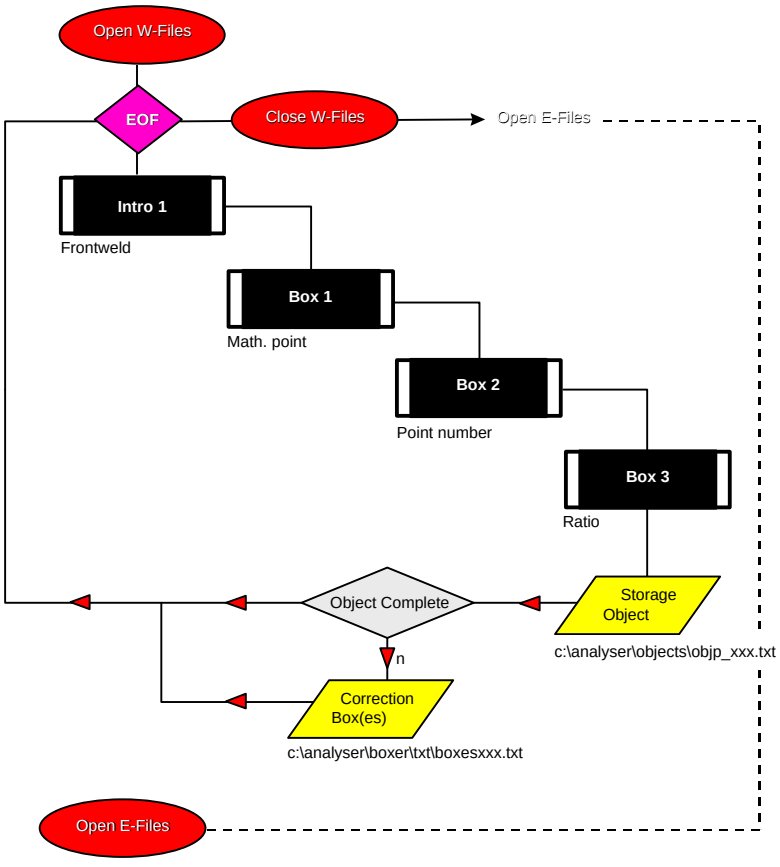


Figure 7-16: Flow Chart Boxes

In *Figure 7-16* an object is composed via the box cascade. In this case a common point.

7.8 English Point

The object English Point (type 001E) can be composed with the same box catching technique. An English point differs from a common rail point in various ways. An English point does not have a frontweld. This gives rise to a complication, because an English point is not direction sensitive and for that reason we can not give a direction to the catching box. We must find a

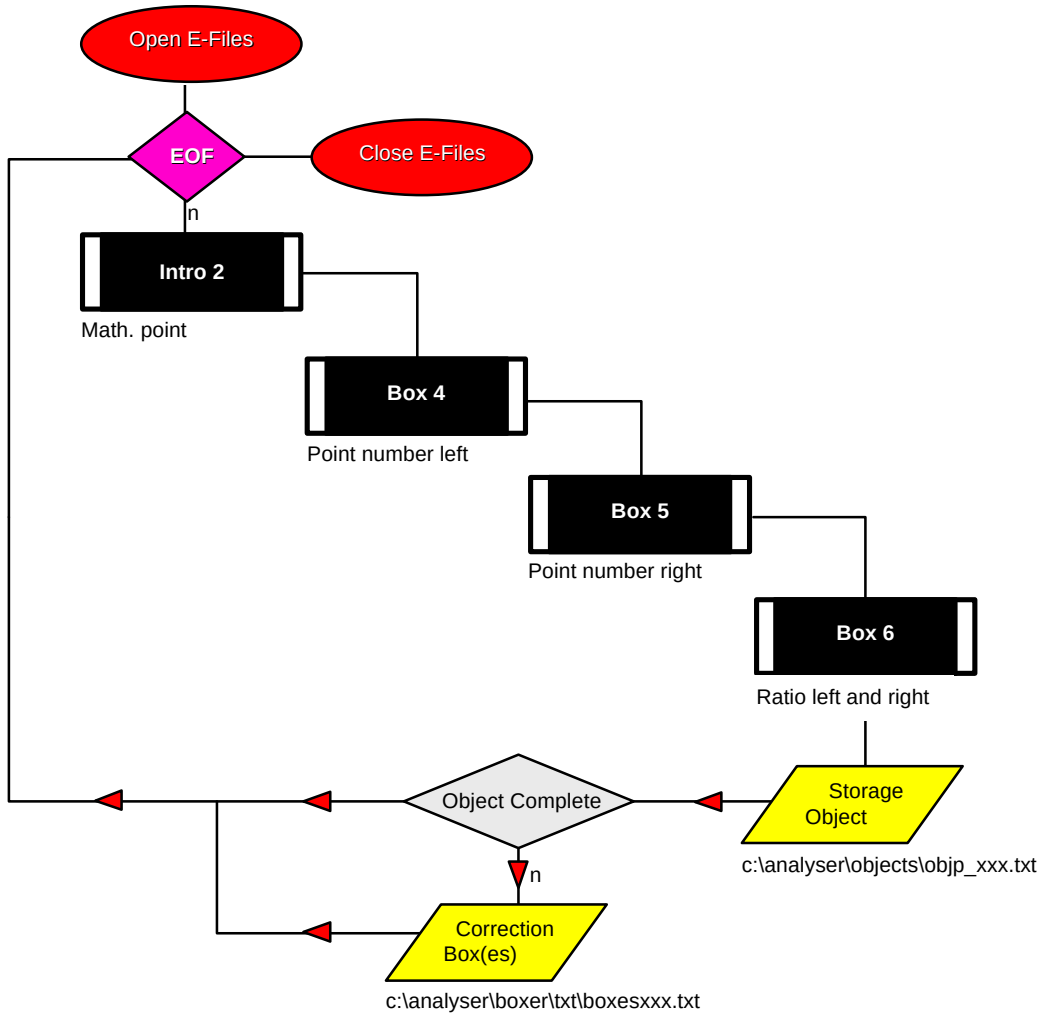


Figure 7-17: Flowchart English point

solution in another way. In addition to a mathematical point an English point, has a point number and a ratio on both sides of the mathematical point. In order to find them, we look at the files of the directory c:\analyser\points_w again. However, only the records that are available according to the bookkeeping, so those with an “A” for Available in that particular field can be considered. An English point is not direction sensitive from the start. For that reason I

have created the reversed catching box. It is certain to include each mathematical point of the type 001E, and each insertion point (a,b) on both sides of the mathematical point must be found at both sides on both sides of the mathematical point. We start by the finding candidates for the point number with X-co-ordinate less than a, and then greater than a. First we start searching for the insertion points (p,q) of the point number. Then we check the system to see if (a,b) is within the box round (p,q) with the direction of the text found. (reversed catching). In other words, if the answer is 'yes' the system adopts the direction of the point number text. In this way we can test whether the two insertion points (a,b) and (p,q) belong to each other or not. The point number to the right can now be found by defining a box ,which is mirrored in relation to the mathematical point. Searching for the ratios is undertaken by enlarging the boxes.

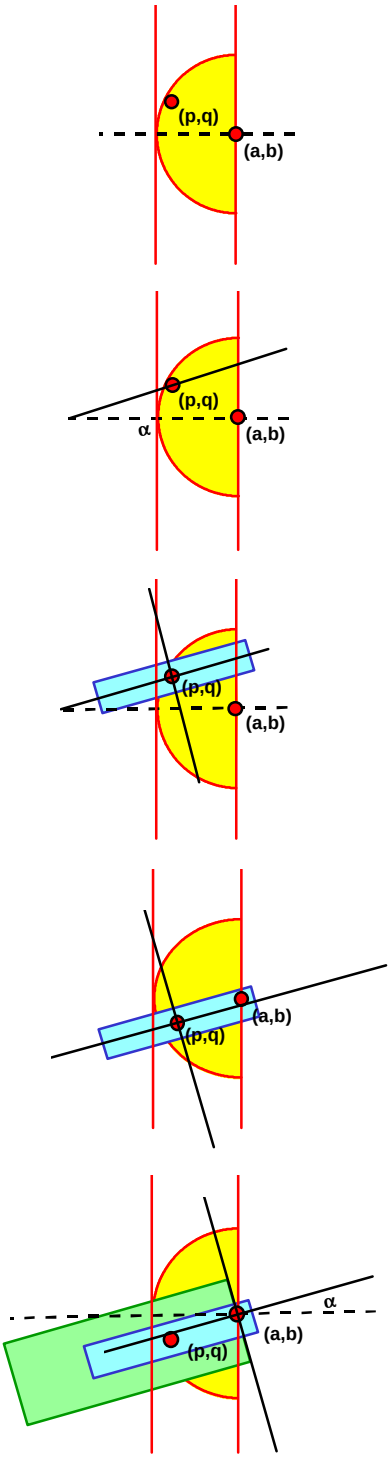


Figure 7-18: Catching of an English Point

7.8.1 Box4

Here we search for candidate 'left' point numbers in

c:\analyser\point_w\nr506w_r.txt

Box4=(a,b,a,1,18)

7.8.2 Box5

Here we search for candidate 'right' point numbers in

c:\analyser\point_w\nr506w_r.txt

Box5(a,b,a,1,18)

7.8.3 Box6

Here we search for candidate 'left' and right ratios in c:\analyser\point_w\rt506w_r.txt.

Box6=(a,b,2.5,30)

7.9 Objects

7.9.1 Points of the type 001W/K

Number	X-co”rdinate	Y-co”rdinate	Type	Point	Ratio	
1	152529.992	463446.711	001W	23	1:9	
2	152556.263	463421.781	001W	501	1:9	
3	152581.928	463402.515	001W	503	1:9	
4	152595.190	463385.216	001W	505	1:9	
5	152617.906	463350.835	001W	506	1:9	
6	152622.535	463320.850	001W	25A	1:9	
7	152624.233	463370.702	001W	508	1:9	
8	152644.039	463320.100	001W	507	1:9	
9	152647.828	463285.075	001W	27	1:9	
10	152661.235	463284.584	001W	25B	1:9	
11	152686.935	463324.143	001W	509	1:9	
12	152723.993	463231.872	001W	35	1:9	
13	152734.098	463279.383	001W	512	1:9	
14	152720.461	463298.880	001W	511	1:9	
15	152765.673	463255.376	001W	513	1:7	
16	152775.868	463255.345	001W	514	1:8	
17	152760.121	463254.407	001W	520	1:9	
18	152795.526	463158.225	001W	71	1:9	
19	152801.931	463243.218	001W	515	1:8	
20	152811.282	463123.977	001W	51A	1:9	
21	152812.571	463129.358	001W	51A	1:9	
22	152825.976	463235.631	001W	401	1:9Sy	
23	152826.843	463134.897	001W	73	1:9	
24	152839.324	463225.539	001W	402	1:7	
25	152835.311	463095.753	001W	61	1:9	
26	152854.195	463228.152	001W	404	1:7	
27	152856.478	463223.577	001W	403	1:7Sy	
28	152840.543	463078.810	001W	51B	1:9	
29	152861.741	463065.031	001W	59	1:9	
30	152862.080	463052.210	001W	601	1:9	
31	152879.317	463031.343	001W	602	1:9	
32	152888.693	463207.911	001W	410	1:7Sy	
33	152898.754	463012.862	001W	603	1:9	
34	152888.693	463207.911	001W	...	1:7Sy	1
35	152912.931	463115.300	001W	551	1:9	
36	152914.089	463008.757	001W	606	1:9	
37	152895.608	463156.443	001W	522	1:9	
38	152916.422	463197.385	001W	411	1:7	
39	152919.777	462996.396	001W	604	1:9	
40	152938.107	462996.584	001W	607	1:9	
41	152942.150	463177.789	001W	414	1:7	
42	152952.956	463183.096	001W	415	1:7	
43	152955.597	462974.608	001W	605	1:9	
44	152963.189	462987.139	001W	608	1:9	
45	152964.945	463168.894	001W	413	1:7	
46	152948.326	463166.463	001W	416	1:7Sy	
47	152968.895	463162.752	001W	415	1:7	
48	152988.420	462977.593	001W	609	1:9	
49	152977.558	463071.676	001W	552	1:9	
50	152999.689	463100.283	001W	524	1:9	
51	153026.499	463050.670	001W	554	1:9	
52	153028.064	462917.663	001W	618	1:9	
53	153030.952	463247.064	001W	408	1:7	
54	153030.585	463044.940	001W	555	1:9	
55	153052.650	463044.168	001W	556	1:9	
56	153034.029	463072.093	001W	526	1:9	
57	153053.340	463030.173	001W	559	1:9	
58	153057.454	463039.388	001W	557	1:9Sy	
59	153060.285	463030.123	001W	560	1:9Sy	
60	153065.481	463105.939	001W	424	1:9	
61	153079.735	463040.626	001W	561	1:8	
62	153082.502	463036.657	001W	562	1:9	
63	153084.714	463019.612	001W	563	1:9Sy	
64	153064.255	463065.766	001W	525	1:9	
65	153088.489	463029.704	001W	558	1:9Sy	
66	153096.404	463234.462	001W	409	1:7	
67	153096.195	463004.028	001W	564	1:9	
68	153081.471	463051.316	001W	528	1:9	
69	153100.077	463051.170	001W	527	1:9	
70	153126.522	463034.108	001W	570	1:9	

71	153115.219	463071.728	001W	422	1:7	
72	153125.757	463043.765	001W	568	1:9	
73	153158.817	463118.028	001W	425	1:7	
74	153136.223	462979.017	001W	565	1:9	
75	153139.621	463066.991	001W	423	1:7	
76	153165.322	463032.450	001W	572	1:9	
77	153197.139	463035.161	001W	...	...	2
78	153199.762	463116.285	001W	428	1:7	
79	153224.691	463025.220	001W	576	1:9 S	
80	153232.371	463033.240	001W	575	1:8	
81	153218.341	462883.032	001W	610	1:9	
82	153238.574	462862.027	001W	619	1:9	
83	153263.582	463013.119	001W	577	1:9	
84	153256.112	462875.878	001W	611	1:9	
85	153280.507	462896.459	001W	77A	1:9	
86	153304.828	463050.848	001W	...	...	3
87	153312.674	462799.724	001W	87	1:10	
88	153292.138	462869.104	001W	612	1:9	
89	153319.655	462796.132	001W	85A	1:10	
90	153303.006	462817.750	001W	83	1:9	
91	153330.479	462812.230	001W	81	1:10	
92	153332.114	462816.353	001W	81	1:10	
93	153318.460	462880.379	001W	77B	1:9	
94	153331.687	462861.745	001W	615	1:9	
95	153339.898	462846.449	001W	620	1:9	
96	153356.461	462864.229	001W	615	1:9	
97	153393.003	463039.144	001W	432	1:7	
98	153371.948	462854.317	001W	614	1:9	
99	153373.804	462838.672	001W	621	1:9	
100	153399.450	462804.751	001W	123A	1:10	
101	153401.966	462800.861	001W	125A	1:10	
102	153397.615	462846.474	001W	618	1:9	
103	153400.042	462832.297	001W	622	1:9	
104	153450.679	462811.107	001W	131A	1:9	
105	153429.162	462836.708	001W	617	1:9	
106	153428.392	463027.050	001W	433	1:7	
107	153455.854	462850.335	001W	107A	1:8	
108	153437.861	462803.219	001W	125B	1:9	
109	153442.475	462866.153	001W	109	1:9	
110	153443.489	462807.361	001W	123B	1:5	
111	153485.768	462801.035	001W	127A	1:9	
112	153486.211	462830.210	001W	101	...	4
113	153497.338	462864.462	001W	121	1:9	
114	153501.962	462849.399	001W	107B	1:9	
115	153508.326	462852.772	001W	111	1:9	
116	153539.779	462990.220	001W	441	1:9	
117	153522.757	462795.221	001W	127B	1:9	
118	153553.329	462829.345	001W	145A	1:9	
119	153538.534	462849.905	001W	113	1:9	
120	153561.931	462793.554	001W	139A	1:9	
121	153566.202	462801.986	001W	137A	1:10	
122	153544.656	462897.908	001W	626	1:9	
123	153568.062	463061.975	001W	438	1:7	
124	153547.323	462856.282	001W	119	1:9	
125	153577.640	462855.997	001W	117A	1:9	
126	153577.646	462851.399	001W	115A	1:9	
127	153584.130	463078.615	001W	437	1:9	
128	153595.650	462836.248	001W	149A	1:9	
129	153575.444	462987.317	001W	442	1:9	
130	153576.897	462881.360	001W	630	1:9	
131	153581.524	462896.061	001W	625	1:9	
132	153581.160	463032.877	001W	440	1:7	
133	153588.144	462890.252	001W	627	1:9	
134	153600.781	462782.878	001W	139B	1:9	
135	153605.532	462804.280	001W	137B	1:10	
136	153612.993	462875.299	001W	631	1:9	
137	153615.230	462885.537	001W	628	1:9	
138	153616.957	463058.756	001W	439	1:7	
139	153618.052	462853.684	001W	117B	1:9	
140	153618.057	462858.341	001W	115B	1:9	
141	153621.945	462878.924	001W	629	1:9	
142	153627.563	462872.270	001W	632	1:9	
143	153628.069	462778.598	001W	143	1:9	
144	153654.060	462864.336	001W	167A	1:9	
145	153669.041	462853.306	001W	157A	1:9	
146	153648.955	462877.430	001W	633	1:9	
147	153670.742	462874.834	001W	634	1:9	
148	153675.846	462878.921	001W	635	1:9	

149	153677.637	462831.968	001W	151B	1:9	
150	153705.444	462901.151	001W	653	1:9	
151	153735.772	462864.567	001W	...	1:9	5
152	153743.439	463048.216	001W	443	1:7	
153	153742.202	462956.411	001W	448	1:9	
154	153742.675	462887.079	001W	638	1:9	
155	153765.317	462804.500	001W	181A	1:9	
156	153749.714	462880.588	001W	...	1:9	6
157	153770.407	462874.986	001W	169A	1:9	
158	153770.269	462884.146	001W	167B	...	7
159	153775.555	462871.382	001W	163B	1:9	
160	153798.976	462885.542	001W	654	1:7	
161	153798.976	462885.542	001W	654	1:7	
162	153777.715	462937.509	001W	675	1:9	
163	153789.133	462952.685	001W	449	1:9	
164	153803.185	462930.338	001W	677	1:7	
165	153805.572	462811.345	001W	181B	1:9	
166	153806.941	462890.540	001W	655	1:9	
167	153810.813	462881.686	001W	169B	1:9	
168	153813.849	462940.506	001W	676	1:9	
169	153819.412	462925.629	001W	680	1:7	
170	153842.191	462817.857	001W	185A	1:9	
171	153842.614	462813.426	001W	185A	1:9	
172	153853.041	462846.602	001W	203A	1:9	
173	153853.475	462842.229	001W	201A	1:9	
174	153859.682	462897.631	001W	657	1:7	
175	153836.586	462938.106	001W	678	1:7	
176	153862.107	462893.570	001W	658	1:9	
177	153844.159	462959.999	001W	450	1:9	
178	153844.377	462875.484	001W	175B	1:9	
179	153844.750	462935.349	001W	681	1:7	
180	153883.576	462811.416	001W	189A	1:9	
181	153859.682	462897.631	001W	...	1:7	8
182	153868.026	462954.378	001W	451	1:7	
183	153873.811	462946.656	001W	682	1:7	
184	153874.392	462958.283	001W	452	1:7	
185	153875.830	462950.596	001W	679	1:9	
186	153882.495	462820.319	001W	183B	1:9	
187	153893.738	462844.457	001W	203B	1:9	
188	153898.889	462956.966	001W	453	1:7	
189	153901.595	462891.096	001W	660	1:9	
190	153909.469	462963.884	001W	454	1:7	
191	153934.765	463016.961	001W	455	1:7	
192	153923.436	462818.192	001W	189B	1:9	
193	153923.886	462813.575	001W	187B	1:9	
194	153930.087	462959.737	001W	459	1:7	
195	153933.803	462855.795	001W	205B	1:9	
196	153948.804	462965.475	001W	460	1:7	
197	153954.758	462961.835	001W	461	1:7	
198	153967.969	462924.199	001W	662	1:9	
199	153979.195	462963.974	001W	462	1:7	
200	154017.643	462856.093	001W	233A	1:9	
201	153999.468	462969.621	001W	463	1:7	
202	154003.690	462966.102	001W	464	1:7	
203	154007.846	462991.616	001W	457	1:7	
204	154052.525	462970.508	001W	...	1:7	9
205	154036.829	462981.715	001W	458	1:7	
206	154037.979	463005.940	001W	...	1:7	10
207	154046.630	462974.138	001W	466	1:7	
208	154052.525	462970.508	001W	467	1:7	
209	154066.444	462996.190	001W	473	1:7	
210	154072.081	462944.975	001W	663	1:9	
211	154076.941	462972.530	001W	468	1:7	
212	154101.439	462851.378	001W	221	1:8	
213	154106.565	462833.226	001W	219	1:8	
214	154097.886	462985.134	001W	472	1:7	
215	154124.077	462820.438	001W	213A	1:9	
216	154101.319	462974.679	001W	469	1:7	
217	154127.705	462976.900	001W	470	1:7	
218	154155.405	462979.376	001W	471	1:9	
219	154164.819	462822.641	001W	215B	1:9	
220	154164.378	462827.317	001W	213B	1:9	
221	154174.913	462847.467	001W	219B	1:8	
222	154228.114	462917.452	001W	690	1:9	
223	154245.504	462891.376	001W	241	1:9	
224	154290.112	462905.244	001W	243	1:9	
225	154317.269	462922.467	001W	245	1:9	
226	154361.031	462943.757	001W	235	1:9	

227	154375.429	462959.955	001W	253	1:9	
228	154391.138	462962.908	001W	251	1:9	
229	154417.074	462966.884	001W	271A	1:12	
230	154397.149	462986.518	001W	701	1:9	
231	154415.409	462982.890	001W	259A	1:9	
232	154400.193	462980.140	001W	255	1:9	
233	154420.704	462952.402	001W	725	1:10	
234	154422.876	462994.868	001W	255B	1:9	
235	154444.332	463024.113	001W	702	1:9	
236	154440.192	462971.346	001W	730	1:10	
237	154440.192	462971.346	001W	730	1:10	
238	154470.701	463048.034	001W	703	1:9	
239	154457.263	463011.859	001W	271B	1:12	
240	154456.488	463004.549	001W	271B	1:12	
241	154491.827	463050.760	001W	275A	1:12	
242	154475.779	463039.641	001W	261B	1:9	
243	154515.715	463089.029	001W	277A	1:9	
244	154545.887	463114.766	001W	...	1:9	11
245	154539.707	463120.723	001W	283	1:9	
246	154555.565	463141.576	001W	715	1:8	
247	154542.632	463117.375	001W	281B	1:9	
248	154545.887	463114.766	001W	277B	1:9	
249	154564.936	463139.631	001W	285A	1:9	
250	154564.936	463139.631	001W	...	...	12
251	154597.358	463196.616	001W	720	1:9	
252	154589.843	463165.859	001W	285B	1:9	

7.9.2 Points of the type 001E/H

Number	X-co”rdinate	Y-co”rdinate	Type	LPoint	RPoint	LRatio	RRatio	
253	152701.119	463253.124	001E	31A	33	...	1:9	1
254	152727.613	463222.286	001E	39A	31B	1:9	1:9	
255	152757.416	463194.089	001E	41A	37B	1:9	1:9	
256	152757.008	463187.676	001E	43	39B	1:9	1:9	
257	152786.308	463160.326	001E	45	41B	1:9	1:9	
258	152994.205	463135.671	001E	418A	418B	1:7	1:7	
259	153001.462	463059.653	001E	553A	...	1:9	...	2
260	153020.156	463118.728	001E	419A	419B	1:7	1:7	
261	153046.649	463103.216	001E	420A	420B	1:7	1:7	
262	153084.667	463083.601	001E	...	...	...	...	3
263	153125.202	463039.251	001E	569A	569B	1:9	1:9	
264	153165.810	463036.866	001E	571A	...	...	1:9	4
265	153359.386	462802.405	001E	89	85B	...	1:5	5
266	153426.250	462817.541	001E	97A	95	1:10	1:10	
267	153469.570	462824.630	001E	99A	97B	1:10	1:10	
268	153472.434	462820.316	001E	141A	91	1:10	1:10	
269	153486.066	462805.455	001E	133A	131B	1:9	1:9	
270	153512.880	462831.661	001E	103A	99B	1:9	1:9	
271	153525.948	462799.251	001E	135	133B	1:9	1:9	
272	153555.441	462834.109	001E	153A	141B	1:9	1:9	
273	153589.177	462844.328	001E	105A	103B	1:9	1:9	
274	153594.209	462827.074	001E	147	145B	...	...	6
275	153630.876	462846.739	001E	155	153B	1:9	1:9	
276	153636.747	462834.180	001E	151A	149B	...	...	7
277	153658.096	462855.975	001E	159A	105B	1:9	1:9	
278	153699.934	462858.482	001E	161	157B	1:9	1:9	
279	153702.647	462880.248	001E	636A	636B	1:9	1:9	
280	153735.208	462869.086	001E	165	159B	1:9	1:9	
281	153782.203	463046.463	001E	445A	445B	1:7	1:7	
282	153811.050	462877.227	001E	...	173B	1:9	1:9	8
283	153830.026	462891.765	001E	656A	656B	1:7	1:7	
284	153883.173	462815.964	001E	187A	185B	1:4	...	9
285	153893.570	462848.964	001E	205A	201B	...	1:9	10
286	153899.218	462952.616	001E	683A	683B	1:7	1:7	
287	153971.951	463004.175	001E	456A	456B		1:7	
288	154058.670	462853.769	001E	225	233B	1:9	1:9	
289	154123.754	462825.093	001E	215A	217	...	1:9	11
290	154354.381	462959.371	001E	255A	...	...	...	12
291	154443.698	463012.172	001H	...	259B	...	...	13
292	154520.694	463088.393	001H	...	279	...	1:9	14

7.10 Problem points

7.10.1 Frontwelds

Nr	A(vailable)	X-coordinate	Y-coordinate	Type	Angle
1	A	152899.310	463202.052	017L	151
2	A	153185.593	463035.787	017L	176
3	A	153293.021	463053.090	017L	169
4	A	153496.824	462830.752	017L	2
5	A	153724.843	462862.695	017L	189
6	A	153759.199	462880.183	017L	177
7	A	153781.127	462884.639	017L	2
8	A	153871.792	462900.078	017L	191
9	A	154040.345	462969.311	017L	4
10	A	154049.170	463002.170	017L	-19
11	A	154536.304	463102.267	017L	52
12	A	154574.430	463152.209	017L	232

7.10.2 Mathematical points

Nr	A(vailable)	X-coordinate	Y-coordinate	Type	Angle
1	A	152562.457	463415.767	001W	0
2	A	152588.774	463397.353	001W	0
3	A	152674.380	463260.242	001W	0
4	A	152701.119	463253.124	001E	0
5	A	152730.599	463225.722	001W	0
6	A	152810.076	463239.330	001W	0
7	A	152910.124	463196.060	001W	0
8	A	153001.462	463059.653	001E	0
9	A	153084.667	463083.601	001E	0
10	A	153165.810	463036.866	001E	0
11	A	153197.139	463035.161	001W	0
12	A	153304.828	463050.848	001W	0
13	A	153359.386	462802.405	001E	0
14	A	153370.778	462814.535	001W	0
15	A	153378.758	462810.102	001W	0
16	A	153398.880	462813.495	001W	0
17	A	153401.941	462809.267	001W	0
18	A	153421.904	462812.447	001W	0
19	A	153486.211	462830.210	001W	0
20	A	153514.872	462827.199	001W	0
21	A	153594.209	462827.074	001E	0
22	A	153597.875	462854.835	001W	0
23	A	153636.747	462834.180	001E	0
24	A	153735.772	462864.567	001W	0
25	A	153749.714	462880.588	001W	0
26	A	153770.269	462884.146	001W	0
27	A	153770.120	462879.697	001W	0
28	A	153790.813	462878.390	001W	0
29	A	153811.050	462877.227	001E	0
30	A	153862.903	462816.962	001W	0
31	A	153873.264	462845.673	001W	0
32	A	153883.173	462815.964	001E	0
33	A	153883.961	462902.515	001W	0
34	A	153893.570	462848.964	001E	0
35	A	153903.775	462814.786	001W	0
36	A	153971.951	463004.175	001E	0
37	A	154028.009	462968.262	001W	0
38	A	154037.979	463005.940	001W	0
39	A	154123.754	462825.093	001E	0
40	A	154144.227	462823.877	001W	0
41	A	154354.381	462959.371	001E	0
42	A	154443.698	463012.172	001H	0
43	A	154455.173	462987.668	001W	0
44	A	154520.694	463088.393	001H	0
45	A	154524.207	463086.350	001W	0
46	A	154586.482	463168.165	001W	0
47	A	154611.473	463207.605	001W	0

7.11 The Final Results

For the evaluation of the final results I have used some statistics. The figures were derived from various tables in the paragraphs, referred to.

Figure 7-19 shows that:

- The CellScan program referred to in *7.3* detected 294 mathematical points of rail points. This means that there are up to 294 point objects to compose.
- Using the Catching program 266 complete rail points are composed.

266 out of a possible 294 gives a 91% score when it comes to the number of complete points located by the Catching program.

240 of the up to 254 common points are located, which gives a score of 95%, and 26 of the up

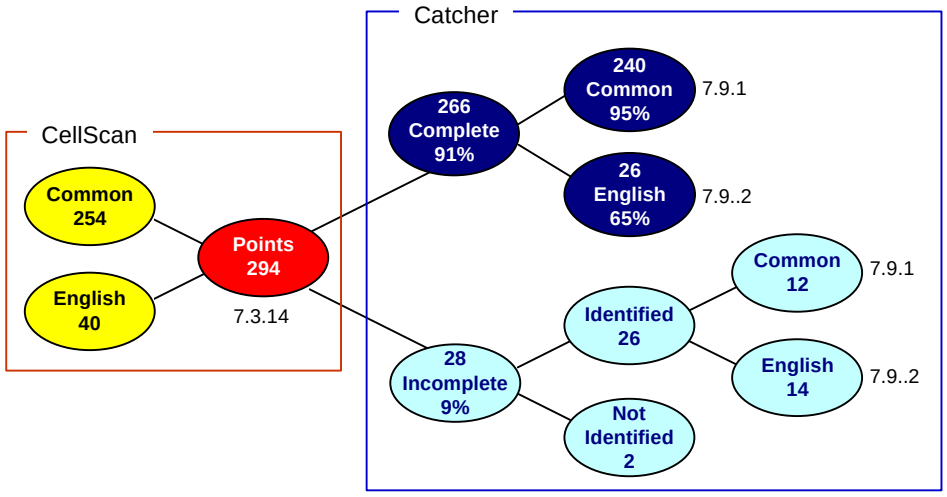


Figure 7-19: The Final Results -1

to 40 English points are located, which make a score of 65%. We can conclude that the English points give the most problems.

A solution could perhaps be found by improving the catch rules for them, but the problems are also caused by the complexity of the point environment.

Figure 7-20 shows the results from a somewhat different perspective.

- The Cellscan programs shows that 86% of the points are Common and 14% are English.
- The Catching program composed 266 complete (91%) and 28 incomplete (9%) points. Of the complete points 240 are Common (90%) and 26 are English (10%). Of the 28 incomplete points 26 have been identified (93%) and 2 have not (7%).

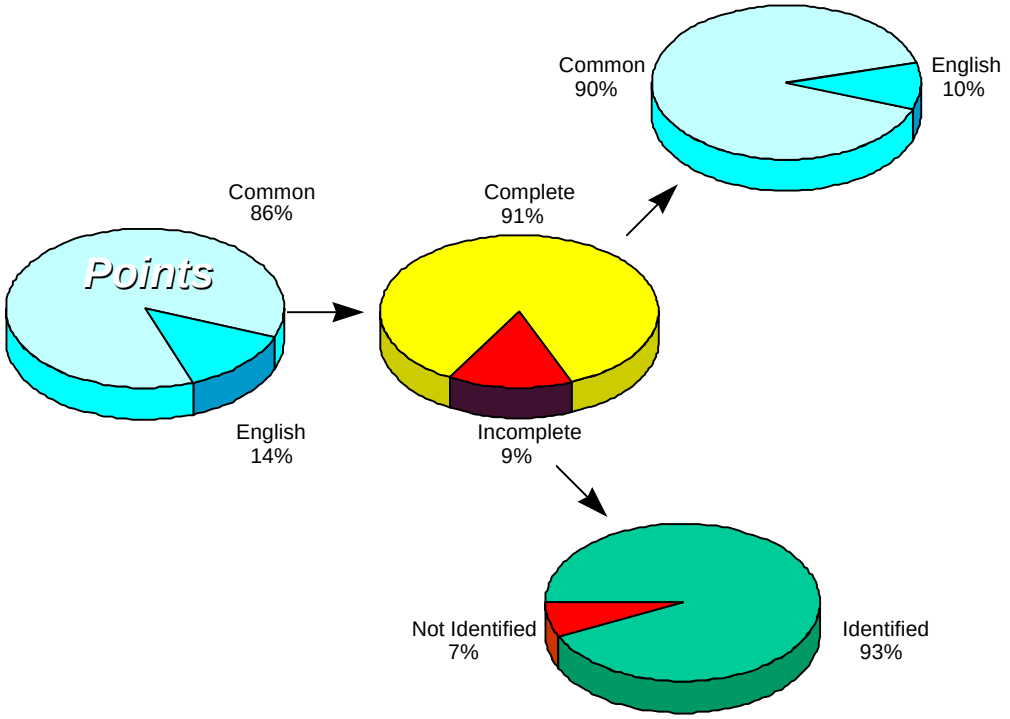


Figure 7-20: The Final Results - 2

7.12 Solids Generator

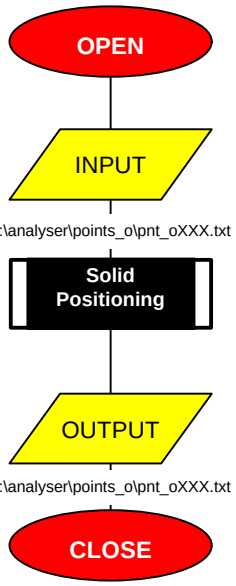


Figure 7-21: Flowchart Solids

Aim of the Solids Generator is to mark the position of the objects found, and using a solid we can express the status and type of object found. These solids or coloured-in symbols can be presented using an overlay. By visualising thigs in such a way the original map data remains intact but the results and shortcomings nevertheless come to light.

7.12.1 Transformations

After the input (c:\analyser\points_o\pnt_oXXX.txt, XXX means: a certain geocode.) are given, besides the eventual other attributes, the co-ordinates of the insertion point as well as the direction of the caught objects. When all the desired elements and attributes of an object are found then the status is complete, otherwise it is incomplete. The transformation of a dashed quadrangle to a yellow quadrangle is a translation from (0,0) to (p,q) after a rotation over the angle α (). The rotation over α is given by the matrix A.

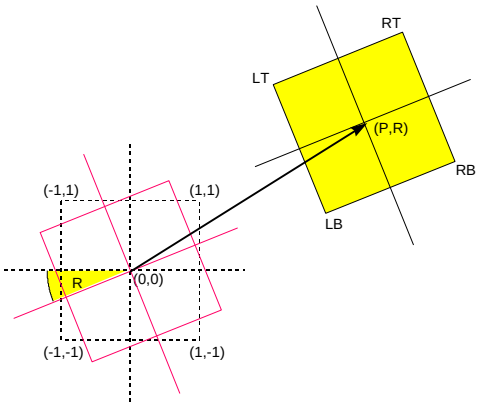


Figure 7-22: Positioning of solids

$$A = \begin{bmatrix} \cos a & -\sin a \\ \sin a & \cos a \end{bmatrix}$$.(see *3.4*)

7.12.2 Positioning

A(1,1) =(cosα-sinα,sinα+cosα) ⇒RT=(P+cosα-sinα , Q+sinα+cosα).

A(1,-1) =(cosα+sinα,sinα–cosα) ⇒RB=(P+cosα+sinα, Q+sinα–cosα).

A(-1,-1) =(-cosα+sinα,-sinα-cosα)⇒LB=(P-cosα+sinα , Q-sinα–cosα).

A(-1,1) =(-cosα-sinα,-sinα+cosα)⇒LT=(P-cosα-sinα , Q-sinα+cosα).

7.12.3 Presentation

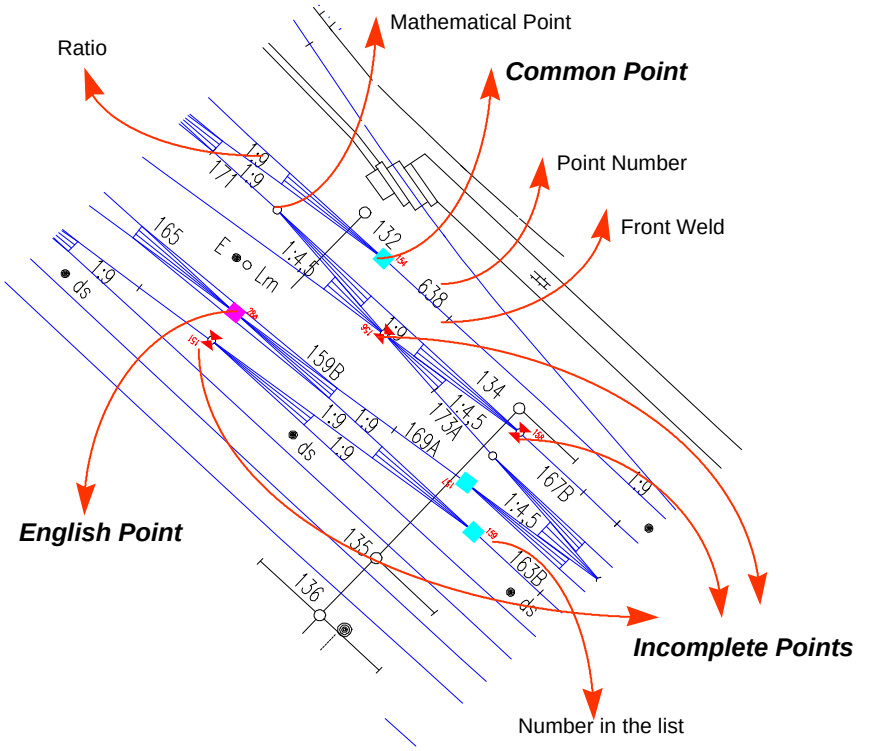


Figure 7-23: Presentation with solids

7.12.4 Source

See *9.2.6*

7.13 Correction Boxes

This program can be especially for draughtsmen, to help them find mistakes in cases where drawing conventions have been violated. *Figure 7-25* shows a correction box indicating that a point number is missing. The record concerned can be found under number 151 in the table of paragraph *7.9.2* Number 280, a complete object of the type 001E, can be found in *Figure 7-25.*

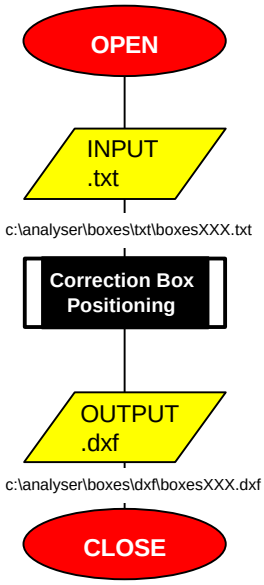


Figure 7-24: Correction Box

7.13.1 The Box Entity

By defining a box entity in a CAD program as library element it is possible to mark the drawing borders by means of a red-lined box inside which where is something which must be corrected.

7.13.2 Parameters

By giving the parameters X and Y for the insertion point the scaling factors dx,and dy, and the angle α over which must be rotated, positioning of the correction box is a established.

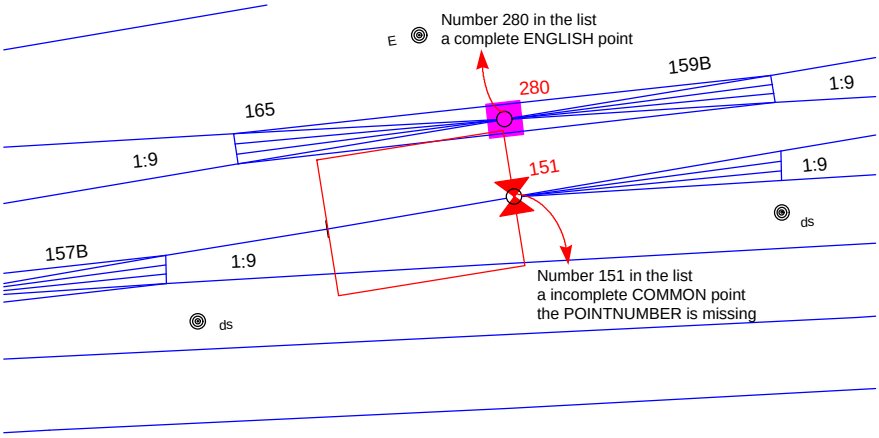


Figure 7-25: Clip to show correction box

7.13.3 Source

see *9.2.7*

7.14 Object Shower

Additionally it is possible to inspect the result of object composing. By typing one of the available geocodes presented, a table of the composed objects for that geocode appears on the screen.

An example of the possible screen output of c:\analyser\objects\objp_506.txt is given below.

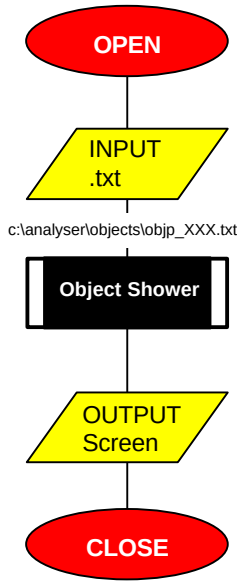


Figure 7-26: Object Shower

```
Nr. Gcd X coord.    Y coord.     Type Angle    Nr Ratio

247,506,154542.632,463117.375, 001W,  51, 281B,  1:9
248,506,154545.887,463114.766, 001W,  53, 277B,  1:9
249,506,154564.936,463139.631, 001W, 231, 285A,  1:9
251,506,154597.358,463196.616, 001W, 231,  720,  1:9
252,506,154589.843,463165.859, 001W,  52, 285B,  1:9

Nr. Gcd X coord.    Y coord.     Type Angle    Nr Ratio     Nr  Ratio

254,506,152727.613,463222.286, 001E, 313,  39A,  1:9,  31B,  1:9
255,506,152757.416,463194.089, 001E, 313,  41A,  1:9,  37B,  1:9
256,506,152757.008,463187.676, 001E, 313,   43,  1:9,  39B,  1:9
257,506,152786.308,463160.326, 001E, 313,   45,  1:9,  41B,  1:9
```

7.14.1 Source

See *9.2.8.*

8. CONCLUSIONS

What has been achieved and can be achieved as a result of the experiences gained through this case study.

With the creation of a spatial code for each relevant element it is possible to sort the set by that code. The result is a spatial ordering, which is linear. This so-called “ZigZagChain” makes it possible to approach vertical strips of a map in sequence.

The order of composing can be deduced from the spatial relationship between the components of the object in question.

After the first component has been identified, an object can be made up by repeated application of the “CatchBox” algorithm.

Each “CatchBox” allocates a strip of the map, and as a result an interval of the “ZigZagChain”. By reading this “ZigZagChain” interval all of the elements of the strip allocated by the “CatchBox” are passed. An element is accepted, if it is located within the “CatchBox”, otherwise the reading of the strip continues.

Repeated application completes each object located, which eventually can be provided with other features.

The questions now are: Where to go from here with Objects? What can be improved? How can the NS find a way out of its revitalisation process, and what about the innovations?

8.1 Objects

The map has an added value, because of the fact that implicit map data can now be made explicit. Raw data is structured to give more information. Moreover through the linking of the map with external administrative data it is possible to gain understanding of administrative processes, which can be visualised, while leaving the original map data intact. A start can now be made with an object-oriented map, which can be realised through a gradually and iterative development of objects.

8.2 Improvement

The NS-map is based on aerial-photographs, digitised and drawn within a system of conventions. Until the present the control on cartographic presentation was visual. Such a control by a human being is not only subjective, but also can not fully consider issues of completeness and accuracy. The operating programs in question are suitable for making the desired connection between the graphical- and administrative environments. In addition the method provides an objective, quick and reliable tool to check and improve the quality of maps. MicroStation is the graphical environment of G&I, while the software of the firm SAP forms the administrative environments of the NS. Both have import and export facilities. The next step has to be, the creation of Black Boxes between them, so that bilateral exchanges and information building can be realised, and the administrative processes can be managed via a graphical entry.

8.3 Revitalisation

In the NS, large-scale projects are often frustrated, due to the unremitting reorganisations, that are the results of one being in two minds as to whether or not to decentralise. The changeover from a national public service to an enterprise, which is made up of financially independent units in accordance with the open conditions as required by the EEC, have so far not been completely normalised, stabilised and fully absorbed. Such as these changes are likely to cause culture shocks, especially in an official environment. Symptoms of over steering are also visible. In short the NS is experiencing a 'forced revitalisation syndrome'.

8.4 Innovation

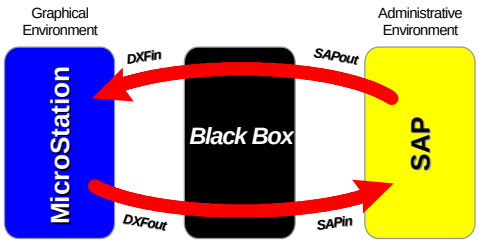


Figure 8-1: The Black Box Adapter

The main aim of the G&I management is to realise, via a graphical entry, the unlocking and the control of maintenance operations. However G&I is an almost exclusive provider for NS Reality and NS RailInfra. This dependency has repercussion for

innovative activities. It is impossible for G&I-experts to take initiatives, because they can not dispose of the budgets, so that the necessary innovations fail to come about in spite of the improvements in the areas of hard-, soft- and humanware. Does one now wait for the principals to wake up for things, or does one shake them up? An additional handicap for G&I is that the administrative part of the whole has not been well-organised up until now. There are too many islands as the result of the fluctuating long-term strategy. Only when one gets a consistent centralised approach towards database management can an unambiguous interaction between the graphical be warranted. However for the experts of G&I it is high-time to start beating the great drum.

9. APPENDICES

9.1 Directory Tree

```
C:\ANALYSER
ÃÄÄÄPROGRAMS
ÃÄÄÄDXF_IN
ÃÄÄÄLOOKUP
ÃÄÄÄGEOCODE
ÃÄÄÄDXF
ÃÄÄÄPOINTS_W
ÃÄÄÄPOINTS_E
ÃÄÄÄPOINTS_X
ÃÄÄÄPOINTS_O
ÃÄÄÄDXF_OUT
ÃÄÄÄSOURCES
ÃÄÄÄSPATIAL
ÃÄÄÄHELP
ÃÄÄÄDGN
ÃÄÄÄBOXES
³   ÃÄÄÄDXF
³   ÀÄÄÄTXT
ÃÄÄÄOBJECTS
ÃÄÄÄTREE
ÃÄÄÄDWG
ÀÄÄÄCELLS
```

9.2 Source Code Listings

9.2.1 Menu Generator

```
1000 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
1010 '³ For NS Geodesy and InfraData                  ³
1020 '³ by Joop W. BLOM                               ³
1030 'ÃÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ´
1040 '³ menugenerator: nsmenu00.bas                   ³
1050 '³ versie 18-01-1997                             ³
1060 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
1070 '
1080 '
1090 CLEAR
1100 DEFINT E-N
1110 MAX = 9
1120 DIM KEUZE$(MAX)
1130 IF MAX > 4 THEN K = 1 ELSE K = 2
1140 MENU$ = "    Object Composer "
1150 KEUZE$(1) = "þ Read Me First     þ"
1160 KEUZE$(2) = "þ Cell Scanner      þ"
1170 KEUZE$(3) = "þ Spatial Coder     þ"
1180 KEUZE$(4) = "þ QuickSort         þ"
1190 KEUZE$(5) = "þ Object Catcher    þ"
1200 KEUZE$(6) = "þ Visualisation     þ"
1210 KEUZE$(7) = "þ Correction Boxes þ"
1220 KEUZE$(8) = "þ Objects Shower    þ"
1230 KEUZE$(9) = "þ Stop the Program þ"
1240 IX = 30: IY = 7
1250 NORMAAL1 = 7: NORMAAL2 = 0
1260 INVERS1 = 15: INVERS2 = 1
1270 INVERS3 = 11: INVERS4 = 13
1280 INVERS5 = 15: INVERS6 = 0
1290 INVERS7 = 4: INVERS8 = 5
1300 KEY OFF: COLOR NORMAAL1, NORMAAL2: CLS
1310 GOSUB 2490
1320 GOSUB 1380                        'menuinvoer
1330 '
1340 ON KEUZE GOSUB 2680, 2730, 2760, 2790, 2820, 2850, 2880, 2910, 2940
1350 GOTO 1000
1360 '
1370 '
1380 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
1390 '³ Menugenerator                                 ³
1400 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
1410 'Functie: Laten zien van een menu, vragen om een keuze
1420 'Invoer  : MAX         = Aantal keuzen
1430 '          KEUZE$()    = Tekst keuzen
1440 '          IX,IY       = Positie linkerbovenhoek menu
1450 '          NORMAAL1,2  = Kleuren menu
1460 '          INVERS1,2   = Kleuren keuze
1470 'Uitvoer : KEUZE       = Keuze
1480 '          ESCAPE      = GOED/NOT GOED
1490 '
1500 GOSUB 1630                          'initialisatie
1510 GOSUB 1710                          'vertoon menu
1520 GOSUB 2400                          'uitleg toetsen
1530 ENTER = NOT GOED: ESCAPE = NOT GOED
1540 WHILE ENTER = NOT GOED AND ESCAPE = NOT GOED
1550 GOSUB 1770                          'verandering
1560 GOSUB 1860                          'registratie
1570 WEND
1580 RETURN
1590 '
1600 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
1610 '³ Initialisatie                                 ³
1620 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
1630 REF1$ = CHR$(13) + CHR$(27) + CHR$(30) + CHR$(31) + CHR$(50) + CHR$(56)
1640 REF2$ = CHR$(72) + CHR$(80)
1650 KEUZE = 1: EERDER = K: IY = IY - K
1660 RETURN
1670 '
1680 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
1690 '³ Vertoon menu                                  ³
1700 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
1710 COLOR NORMAAL1, NORMAAL2
1720 FOR I = 1 TO MAX
1730 LOCATE IY + 2 + K * I, IX - 2: PRINT KEUZE$(I)
1740 NEXT I
```

```
1750 RETURN
1760 '
1770 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
1780 '³ Verandering                                      ³
1790 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
1800 COLOR NORMAAL1, NORMAAL2
1810 LOCATE IY + 2 + K * EERDER, IX - 2: PRINT KEUZE$(EERDER);
1820 COLOR INVERS1, INVERS2
1830 LOCATE IY + 2 + K * KEUZE, IX - 2: PRINT KEUZE$(KEUZE);
1840 RETURN
1850 '
1860 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
1870 '³ Registratie                                      ³
1880 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
1890 A$ = INKEY$
1900 WHILE A$ <> "": A$ = INKEY$: WEND   'Buffer schoon
1910 A$ = INKEY$
1920 WHILE A$ = "": A$ = INKEY$: WEND    'Karakter inlezen
1930 'Len(a$)=1,Len(a$)=2
1940 ON LEN(A$) GOSUB 2040, 1970
1950 RETURN
1960 '
1970 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
1980 '³ LEN(A$)=2                                        ³
1990 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
2000 '<Cursor omhoog>,<Cursor omlaag>
2010 ON INSTR(REF2$, RIGHT$(A$, 1)) GOSUB 2110, 2180
2020 RETURN
2030 '
2040 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
2050 '³ LEN(A$)=1                                        ³
2060 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
2070 '<ENTER>,<Esc>,<Cursor omhoog>,<Cursor omlaag>,<Cursor omlaag>,<Cursor omhoog>
2080 ON INSTR(REF1$, A$) GOSUB 2250, 2310, 2110, 2180, 2180, 2110
2090 RETURN
2100 '
2110 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
2120 '³ <CURSOR OMHOOG>                                  ³
2130 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
2140 EERDER = KEUZE
2150 IF EERDER = 1 THEN KEUZE = MAX ELSE KEUZE = KEUZE - 1
2160 RETURN
2170 '
2180 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
2190 '³ <CURSOR OMLAAG>                                  ³
2200 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
2210 EERDER = KEUZE
2220 IF EERDER = MAX THEN KEUZE = 1 ELSE KEUZE = KEUZE + 1
2230 RETURN
2240 '
2250 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
2260 '³ <ENTER>                                          ³
2270 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
2280 ENTER = GOED
2290 RETURN
2300 '
2310 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
2320 '³ <Esc>                                            ³
2330 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
2340 ESCAPE = GOED
2350 RETURN
2360 '
2370 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
2380 '³ Uitleg toetsen                                   ³
2390 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
2400 LOCATE 20, 23
2410 COLOR INVERS1, INVERS2: PRINT " " + CHR$(24) + " =Up  "; : COLOR NORMAAL1, NORMAAL2
2420 PRINT "  "; : COLOR INVERS1, INVERS2: PRINT " " + CHR$(25) + " =Down "; : COLOR
NORMAAL1, NORMAAL2
2430 PRINT "  "; : COLOR INVERS1, INVERS2: PRINT " <" + CHR$(196) + CHR$(217) + " =OK ";
: COLOR NORMAAL1, NORMAAL2
2440 RETURN
2450 '
2460 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
2470 '³ Kader                                            ³
2480 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
2490 CLS
2500 KLANT$ = " NS Geodesy and InfraData "
```

```
2510 RH$ = STRING$(1, 196) + CHR$(191)
2520 LH$ = CHR$(218) + STRING$(1, 196)
2530 BREED = LEN(LH$ + KLANT$ + RH$)
2540 LL$ = CHR$(192) + STRING$(BREED - 2, 196) + CHR$(217)
2550 ZIJ$ = CHR$(179) + SPACE$(BREED - 2) + CHR$(179)
2560 LOCATE IY - 2, IX - 7: PRINT LH$; : COLOR 15, 12: PRINT KLANT$; : COLOR 7, 0: PRINT
RH$
2570 FOR I = 1 TO K * MAX + 4
2580 LOCATE IY - 2 + I, IX - 7: PRINT ZIJ$
2590 NEXT I
2600 LOCATE IY - 2 + I, IX - 7: PRINT LL$
2610 LOCATE IY, IX - 4: COLOR INVERS5, INVERS6: PRINT MENU$: COLOR NORMAAL1, NORMAAL2
2620 RETURN
2630 '
2640 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
2650 '³ Executor                                ³
2660 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
2670 '
2680 COLOR 7, 0: CLS
2690 SHELL "type c:\analyser\help\help.txt|more"
2700 SHELL "pause"
2710 RETURN
2720 '
2730 SHELL "c:\analyser\programs\cellscan.exe"
2740 RETURN
2750 '
2760 SHELL "c:\analyser\programs\spatial.exe"
2770 RETURN
2780 '
2790 SHELL "c:\analyser\programs\slcqsort.exe"
2800 RETURN
2810 '
2820 SHELL "c:\analyser\programs\catcher.exe"
2830 RETURN
2840 '
2850 SHELL "c:\analyser\programs\writer.exe"
2860 RETURN
2870 '
2880 SHELL "c:\analyser\programs\boxes.exe"
2890 RETURN
2900 '
2910 SHELL "c:\analyser\programs\objects.exe"
2920 RETURN
2930 '
2940 CLS
2950 END
```

9.2.2 Cell Scanner

```
1000 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
1010 '³ SCANNING FOR CELLS                       ³
1020 '³ Interpreter: GWBASIC.exe                 ³
1030 '³ version    : 015-12-1996                 ³
1040 '³ Source     : cellscan.bas                ³
1050 '³ Executable : cellscan.exe                ³
1060 '³ For        : NS Geodesy en InfraData     ³
1070 '³ Author     : Joop W. BLOM                ³
1080 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
1090 '
1100 '
1110 '
1120 '
1130 '
1140 DIM TEL(40), TOTAL(40), A$(12)
1150 AANTAL = 0: REGEL = 0
1160 CLIENT$ = "  For NS InfraDocumentation and Geodesy"
1170 '
1180 '
1190 CLS
1200 LINE INPUT "Geocode: "; GCODE$
1210 IF LEN(GCODE$) <> 3 THEN 1200
1220 SHELL "dir c:\analyser\dxf_in\" + GCODE$ + "*.dxf /b >
c:\analyser\lookup\dxffiles.txt"
1230 '
1240 '
1250 OPEN "c:\analyser\lookup\dxfnames.txt" FOR OUTPUT AS 1
1260 '
1270 '
1280 OPEN "c:\analyser\lookup\dxffiles.txt" FOR INPUT AS 2
1290     WHILE NOT EOF(2)
1300       INPUT #2, L$
1310       IF INSTR(L$, ".DXF") > 0 OR INSTR(L$, ".dxf") > 0 THEN PRINT #1, L$
1320     WEND
1330 CLOSE 1, 2
1340 '
1350 CLS : PRINT : PRINT "Processing..........Report": PRINT
1360 '
1370 GOSUB 1630                                  'cellnames
1380 GOSUB 2850                                  'printer
1390 GOSUB 3000                                  'header
1400 GOSUB 1730                                  'examination
1410 IF AANTAL / 4 = INT(AANTAL / 4) THEN 1480
1420 GOSUB 3090                                  'footer
1430 GOSUB 3240                                  'form feed
1440 '
1450 PRINT : PRINT "Processing.........Total report": PRINT
1460 '
1470 GOSUB 3000                                  'header
1480 GOSUB 2300                                  'totalisation
1490 GOSUB 3090                                  'footer
1500 GOSUB 3240                                  'form feed
1510 '
1520 '
1530 '
1540 CLOSE
1550 '
1560 '
1570 '
1580 END
1590 '
1600 '
1610 '
1620 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
1630 '³ READ CELLNAMES              ³
1640 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
1650 '
1660     OPEN "c:\analyser\lookup\allcells.txt" FOR INPUT AS 1
1670             INPUT #1, CELLS$
1680     CLOSE 1
1690             CELLS$ = SPACE$(2) + CELLS$
1700  RETURN
1710 '
1720 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
```

```
1730 '³ OPEN DXF-FILE                ³
1740 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
1750 '
1760     OPEN "c:\analyser\lookup\dxfnames.txt" FOR INPUT AS 1
1770            WHILE NOT EOF(1)
1780                    INPUT #1, INBEST$
1790                    INBEST$ = "c:\analyser\dxf_in\" + INBEST$
1800 '
1810 '                  ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
1820 '                  ³ READ INSERTS                ³
1830 '                  ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
1840 '
1850                    OPEN INBEST$ FOR INPUT AS 3
1860                         WHILE NOT EOF(3) AND IN$ <> "ENTITIES"
1870                           INPUT #3, IN$
1880                'EXTENSIONS
1890                           IF IN$ = "$EXTMIN" OR IN$ = "$EXTMAX" THEN GOSUB 2480
1900                         WEND
1910 '
1920                         WHILE NOT EOF(3)
1930                           INPUT #3, IN$
1940                'CHARACTERISTICS
1950                           IF IN$ = "INSERT" THEN GOSUB 2740
1960                         WEND
1970 '
1980 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
1990 '³ REPORT                       ³
2000 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
2010 '
2020                    PRINT INBEST$
2030 '
2040                    LPRINT STRING$(4, 254); "  "; INBEST$
2050                    LPRINT "ExtmaxX : "; EXTMAXX$; TAB(25); "ExtmaxY : "; EXTMAXY$
2060                    LPRINT "ExtminX : "; EXTMINX$; TAB(25); "ExtminY : "; EXTMINY$
2070                    LPRINT "Xinsert : "; XINS$; TAB(25); "Yinsert : "; YINS$
2080                    LPRINT "Format  : "; FORMAT$; TAB(25); "Rotation: "; ROTATION$
2090            FOR P = 1 TO 4
2100                    FOR K = 1 TO 10
2110                            CL$ = MID$(CELLS$, ((K + ((P - 1) * 10)) * 5) - 4, 5)
2120                            LPRINT TAB((K - 1) * 7); CL$;
2130                    NEXT K
2140                            LPRINT
2150                    FOR K = 1 TO 10
2160                            LPRINT ; TAB((K - 1) * 7); TEL(K + ((P - 1) * 10));
2170                            TEL(K + ((P - 1) * 10)) = 0
2180                    NEXT K
2190                    LPRINT
2200            NEXT P
2210                    LPRINT
2220            CLOSE #3
2230                    AANTAL = AANTAL + 1: REGEL = REGEL + 14
2240                    IF AANTAL / 4 = INT(AANTAL / 4) THEN GOSUB 3090: GOSUB 3240:
GOSUB 3000
2250        WEND
2260 RETURN
2270 '
2280 '
2290 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
2300 '³ TOTAL REPORT                 ³
2310 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
2320 '
2330    LPRINT
2340    LPRINT
2350    LPRINT ; STRING$(4, 254); "  TOTAL REPORT: geocode"; GCODE$
2360    LPRINT ; DATE$
2370    LPRINT
2380            OPEN "c:\analyser\lookup\cells.txt" FOR INPUT AS 2
2390    FOR K = 1 TO 40
2400     INPUT #2, A, B$, C, D, E$
2410     LPRINT MID$(CELLS$, K * 5 - 4, 5); : LPRINT TAB(10); E$; : LPRINT TAB(55);
TOTAL(K)
2420    NEXT K
2430            REGEL = 48
2440 RETURN
2450 '
2460 '
2470 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
2480 '³ EXTENSIONS                   ³
```

```
2490 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
2500 '
2510    INPUT #3, ONE$, TWO$, THREE$, FOUR$
2520    IF IN$ = "$EXTMIN" THEN EXTMINX$ = TWO$: EXTMINY$ = FOUR$: RETURN
2530    IF IN$ = "$EXTMAX" THEN EXTMAXX$ = TWO$: EXTMAXY$ = FOUR$: RETURN
2540 '
2550 '           ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
2560 '           ³ FORM FORMAT                ³
2570 '           ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
2580 '
2590            A$ = "": FORMAT$ = ""
2600            FORMAT$ = FOURTH$
2610            XINS$ = "": YINS$ = "": ROTATION$ = ""
2620                    WHILE A$ <> "50" AND NOT EOF(3)
2630                            A$ = "": B$ = ""
2640                            INPUT #3, A$
2650                            INPUT #3, B$
2660                            IF A$ = "10" THEN XINS$ = B$
2670                            IF A$ = "20" THEN YINS$ = B$
2680                            IF A$ = "50" THEN ROTATION$ = B$
2690                    WEND
2700 '
2710 RETURN
2720 '
2730 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
2740 '³ CHARACTERISTICS                      ³
2750 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
2760 '
2770    INPUT #3, FIRST$, SECOND$, THIRD$, FOURTH$
2780    IF SECOND$ = "LEVEL63" OR SECOND$ = "63" THEN GOSUB 2550 'format
2790    IF INSTR(CELLS$, FOURTH$) = 0 THEN RETURN
2800    N = (INSTR(CELLS$, FOURTH$) + 4) / 5
2810    TEL(N) = TEL(N) + 1: TOTAL(N) = TOTAL(N) + 1
2820 RETURN
2830 '
2840 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
2850 '³ PRINTER                              ³
2860 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
2870 '
2880    LPRINT CHR$(27); CHR$(69);                  'reset
2890    LPRINT CHR$(27); "&l26A";                   'A4
2900    LPRINT CHR$(27); "&l0O";                    'portrait
2910    LPRINT CHR$(27); "&a25l100M";               'marges
2920    LPRINT CHR$(27); "(10U";                    'character set PC-8
2930    LPRINT CHR$(27); "(s0P";                    'portrait
2940    LPRINT CHR$(27); "(s16.67H";                'cpi
2950    LPRINT CHR$(27); "(s10V";                   'points
2960    LPRINT CHR$(27); "(s0T"                     'typeface Univers
2970 RETURN
2980 '
2990 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
3000 '³ HEADER                               ³
3010 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
3020 '
3030    LPRINT STRING$(4, 254); "  Scanning for cells"; TAB(50); DATE$; "  "; TIME$
3040    LPRINT STRING$(70, 196): LPRINT
3050    REGEL = REGEL + 3
3060 RETURN
3070 '
3080 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
3090 '³ FOOTER                               ³
3100 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
3110 '
3120    I = 60 - REGEL
3130    PAGE = PAGE + 1
3140            WHILE I > 0
3150                    LPRINT
3160                    I = I - 1
3170            WEND
3180    REGEL = 0
3190    LPRINT STRING$(70, 196)
3200    LPRINT STRING$(4, 254); CLIENT$; : LPRINT TAB(62); "Page: "; PAGE
3210 RETURN
3220 '
3230 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
3240 '³ PAGE FORCING/FORM FEED              ³
3250 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
3260 '
```

```
3270     LPRINT CHR$(12)                        'form feed
3280 RETURN
```

9.2.3 Spatial Coding

```
1000 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
1010 '³ PREPARATION for determination         ³
1020 '³ Interpreter: GWBASIC.exe              ³
1030 '³ version    : 31-12-1996               ³
1040 '³ Source     : spatial.bas              ³
1050 '³ Executable : spatial.exe              ³
1060 '³ For        : NS Geodesy en InfraData  ³
1070 '³ Author     : Joop W. BLOM             ³
1080 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
1090 '
1100 '
1110 '
1120 CLS
1130 DEFDBL E, S, X-Y
1140 DEFINT F-M
1150 DIM IN$(16)
1160 LEVEL01$ = "105 106 017S017L001W001K001H001E"
1170 '
1180 SHELL "dir c:\analyser\dxf_in\*.dxf /b > c:\analyser\lookup\dxffiles.txt"
1190 '
1200 OPEN "c:\analyser\lookup\dxfnames.txt" FOR OUTPUT AS 1
1210 '
1220 OPEN "c:\analyser\lookup\dxffiles.txt" FOR INPUT AS 2
1230 '
1240    WHILE NOT EOF(2)
1250       INPUT #2, L$
1260       IF INSTR(L$, ".DXF") > 0 OR INSTR(L$, ".dxf") > 0 THEN PRINT #1, L$
1270    WEND
1280 '
1290 CLOSE 1, 2
1300 '
1310 '
1320 OPEN "c:\analyser\lookup\dxfnames.txt" FOR INPUT AS 3
1330 '
1340 PRINT : PRINT "Processing.........": PRINT
1350 '
1360 GCODE$ = ""
1370 '
1380 WHILE NOT EOF(3)
1390            INPUT #3, INBEST$
1400            IF LEFT$(INBEST$, 3) <> GCODE$ THEN GOSUB 2270    'write GEOCODE
1410            INBEST$ = "c:\analyser\dxf_in\" + INBEST$
1420            PRINT INBEST$
1430            OPEN INBEST$ FOR INPUT AS 1
1440            ' ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
1450            ' ³ Read INSERT within ENTITIES                         ³
1460            ' ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
1470                    WHILE NOT EOF(1) AND IN$ <> "ENTITIES"
1480                            INPUT #1, IN$
1490                    WEND
1500 '
1510                    WHILE NOT EOF(1)
1520                            INPUT #1, IN$
1530                                                    'INSERT characteristics
1540                            IF IN$ = "INSERT" THEN GOSUB 1680
1550                                                    'TEXT characteristics
1560                            IF IN$ = "TEXT" THEN GOSUB 1930
1570                    WEND
1580 '
1590            CLOSE 1
1600 '
1610 WEND
1620 '
1630 END
1640 '
1650 '
1660 '
1670 ' ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
1680 ' ³ Read INSERT Characteristics                         ³
1690 ' ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
1700            A$ = ""
1710    WHILE A$ <> "50" AND NOT EOF(1)
1715            A$ = "": B$ = ""
1720            INPUT #1, A$
```

```
1725           INPUT #1, B$
1730           IF A$ = "8" AND B$ <> "LEVEL01" THEN RETURN
1735           IF A$ = "2" THEN CELLNAME$ = B$
1740           IF A$ = "10" THEN XINS$ = B$
1742           IF A$ = "20" THEN YINS$ = B$
1743           IF A$ = "50" THEN ROTATION$ = B$
1750   WEND
1770           ANSWER = INSTR(LEVEL01$, CELLNAME$)
1780           IF ANSWER < 13 THEN RETURN
1790 GOSUB 2150                                              'slcode
1800   PRINT #2, USING "############"; S;
1810   PRINT #2, ",";
1820   PRINT #2, USING "######.###"; VAL(XINS$);
1830   PRINT #2, ",";
1840   PRINT #2, USING "######.###"; VAL(YINS$);
1850   EMPTY$ = SPACE$(5)
1860   RSET EMPTY$ = CELLNAME$
1870   PRINT #2, ",";
1880   PRINT #2, USING "\   \"; EMPTY$;
1890   PRINT #2, ",";
1900   PRINT #2, USING "####"; INT(VAL(ROTATION$))
1910 RETURN
1920 ' ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
1930 ' ³ Read TEXT characteristics                         ³
1940 ' ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
1950           A$ = ""
1960   WHILE A$ <> "50" AND NOT EOF(1)
1970           INPUT #1, A$
1972           INPUT #1, B$
1980           IF A$ = "8" AND B$ <> "LEVEL08" THEN RETURN
1990           IF A$ = "10" THEN XINS$ = B$
1992           IF A$ = "20" THEN YINS$ = B$
1994           IF A$ = "1" THEN TEXT$ = B$
1996           IF A$ = "50" THEN ROTATION$ = B$
2000   WEND
2010 GOSUB 2150                                              'slcode
2020   PRINT #2, USING "############"; S;
2030   PRINT #2, ",";
2040   PRINT #2, USING "######.###"; VAL(XINS$);
2050   PRINT #2, ",";
2060   PRINT #2, USING "######.###"; VAL(YINS$);
2070   EMPTY$ = SPACE$(5)
2080   RSET EMPTY$ = TEXT$
2090   PRINT #2, ",";
2100   PRINT #2, USING "\   \"; EMPTY$;
2110   PRINT #2, ",";
2120   PRINT #2, USING "####"; INT(VAL(ROTATION$))
2130 RETURN
2140 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
2150 '³ Spatial Location Code                              ³
2160 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
2170           XIN = VAL(XINS$)
2180           YIN = VAL(YINS$)
2190           S1 = INT(XIN + .5) * 10 ^ 6
2200           S2 = INT(YIN + .5)
2210           S = S1 + S2
2220 RETURN
2230 '
2240 '
2250 '
2260 ' ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
2270 ' ³ Write the GEOCODE based files                      ³
2280 ' ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
2290   CLOSE 2
2300   GCODE$ = LEFT$(INBEST$, 3)
2310   OPEN "c:\analyser\spatial\pnt_s" + GCODE$ + ".txt" FOR OUTPUT AS 2
2320 RETURN
```

9.2.4 QuickSort

```
1000 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
1010 '³ QUICKSORT for TABLES with a SLC        ³
1020 '³ Interpreter: GWBASIC.exe              ³
1030 '³ version    : 31-12-1996               ³
1040 '³ Source     : slcqsort.bas             ³
1050 '³ Executable : slcqsort.exe             ³
1060 '³ For        : NS Geodesy en InfraData ³
1070 '³ Author     : Joop W. BLOM             ³
1080 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
1090 '
1100 '
1110 '
1120 CLS
1130 '
1140 SHELL "dir c:\analyser\spatial\pnt_s*.txt /b > c:\analyser\lookup\pntsfile.txt"
1150 '
1160 OPEN "c:\analyser\lookup\pntsname.txt" FOR OUTPUT AS 1
1170 OPEN "c:\analyser\lookup\pntsfile.txt" FOR INPUT AS 2
1180 '
1190    WHILE NOT EOF(2)
1200       INPUT #2, L$
1210       IF INSTR(L$, "pnt_s") > 0 OR INSTR(L$, "PNT_S") > 0 THEN PRINT #1, L$
1220    WEND
1230 '
1240 CLOSE 1, 2
1250 '
1260 OPEN "c:\analyser\lookup\pntsname.txt" FOR INPUT AS 3
1270 '
1280 PRINT : PRINT "Processing..........": PRINT
1290 '
1300 DIM ELEMENT(1500), TOTAL$(1500)
1310    WHILE NOT EOF(3)
1320            INPUT #3, INBEST$
1330            GCODE$ = MID$(INBEST$, 6, 3)
1340            INBEST$ = "c:\analyser\spatial\" + INBEST$
1350            PRINT INBEST$
1360            GOSUB 2140                      'open the file to sort
1370 '
1380            GOSUB 2190                      'determination of MAX
1390 '
1400            IF MAX = 1 THEN 1490
1410 '
1420            GOSUB 2140                      'open the file again for sorting
1430            GOSUB 2300                      'read
1440            GOSUB 1580                      'sort subroutine
1450 '
1460            GOSUB 2500                      'open to write the sorted file
1470            GOSUB 2560                      'write
1480 '
1490    WEND
1500 '
1510 CLOSE
1520 '
1530 END
1540 '
1550 ' ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
1560 ' ³ QuickSort                                          ³
1570 ' ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
1580 HOOP1(1) = 1: HOOP2(1) = MAX: K = 1
1590 WHILE K > 0
1600    LAAG = HOOP1(K): HOOG = HOOP2(K)
1610    WHILE HOOG > LAAG
1620            I = LAAG: J = HOOG
1630            CHOICE = ELEMENT((HOOG + LAAG) / 2)
1640            WHILE I <= J
1650                    WHILE ELEMENT(J) > CHOICE AND J > 1
1660                            J = J - 1
1670                    WEND
1680                    WHILE ELEMENT(I) < CHOICE AND I < MAX
1690                            I = I + 1
1700                    WEND
1710                    IF I <= J THEN GOSUB 1840
1720            WEND
1730 '
```

```
1740 'HOOG OP HOOP?, LAAG OP HOOP?
1750 'ÄÄÄÄÄÄÄÄÄÄÄÄÄ
1760            IF (J - LAAG) < (HOOG - I) THEN GOSUB 1920 ELSE GOSUB 2030
1770    WEND
1780 K = K - 1
1790 WEND
1800 RETURN
1810 '
1820 'VERWISSEL
1830 'ÄÄÄÄÄÄÄÄÄÄÄÄÄ
1840    SWAP ELEMENT(I), ELEMENT(J)
1850    SWAP TOTAL$(I), TOTAL$(J)
1860    I = I + 1
1870    J = J - 1
1880 RETURN
1890 '
1900 'HOOG OP HOOP?
1910 'ÄÄÄÄÄÄÄÄÄÄÄÄÄ
1920    IF I < HOOG THEN GOSUB 1980
1930    HOOG = J
1940 RETURN
1950 '
1960 'HOOG OP HOOP
1970 'ÄÄÄÄÄÄÄÄÄÄÄÄÄ
1980    HOOP1(K) = I: HOOP2(K) = HOOG: K = K + 1
1990 RETURN
2000 '
2010 'LAAG OP HOOP?
2020 'ÄÄÄÄÄÄÄÄÄÄÄÄÄ
2030    IF LAAG < J THEN GOSUB 2090
2040    LAAG = I
2050 RETURN
2060 '
2070 'LAAG OP HOOP
2080 'ÄÄÄÄÄÄÄÄÄÄÄÄÄ
2090    HOOP1(K) = LAAG: HOOP2(K) = J: K = K + 1
2100 RETURN
2110 ' ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
2120 ' ³ Open the file to sort                                                ³
2130 ' ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
2140    OPEN INBEST$ FOR INPUT AS 1
2150 RETURN
2160 ' ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
2170 ' ³ Determination of MAX                                                 ³
2180 ' ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
2190            I = 1
2200    WHILE NOT EOF(1)
2210            LINE INPUT #1, Z$
2220            IF NOT EOF(1) THEN I = I + 1
2230    WEND
2240            MAX = I
2250            CLOSE 1
2260 RETURN
2270 ' ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
2280 ' ³ Input of the elements to sort                                       ³
2290 ' ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
2300            I = 1
2310    WHILE NOT EOF(1)
2320            INPUT #1, S, X$, Y$, SP$, ROT$
2330            ELEMENT(I) = S
2340            EMPTYX$ = SPACE$(10)
2345            RSET EMPTYX$ = X$
2350            EMPTYY$ = SPACE$(10)
2355            RSET EMPTYY$ = Y$
2360            EMPTY1$ = SPACE$(5)
2390            RSET EMPTY1$ = SP$
2400            EMPTY2$ = SPACE$(4)
2410            RSET EMPTY2$ = ROT$
2420            TOTAL$(I) = EMPTYX$ + EMPTYY$ + EMPTY1$ + EMPTY2$
2430            I = I + 1
2440    WEND
2450            CLOSE 1
2460 RETURN
2470 ' ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
2480 ' ³ OPEN TO WRITE or READ THE SORTED FILE                               ³
2490 ' ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
2500    OPEN "c:\analyser\geocode\pnt_r" + GCODE$ + ".txt" FOR RANDOM AS 1 LEN = 29
2510    FIELD #1, 10 AS X$, 10 AS Y$, 5 AS SP$, 4 AS ROT$
```

```
2520 RETURN
2530 ' ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
2540 ' ³ WRITE THE SORTED ELEMENTS TO A RANDOM ACCESS FILE ³
2550 ' ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
2560       I = 1
2570    WHILE I < MAX AND NOT EOF(1)
2580           RSET X$ = MID$(TOTAL$(I), 1, 10)
2590           RSET Y$ = MID$(TOTAL$(I), 11, 10)
2600           RSET SP$ = MID$(TOTAL$(I), 21, 5)
2610           RSET ROT$ = MID$(TOTAL$(I), 26, 4)
2620       PUT #1, I
2630       I = I + 1
2640    WEND
2650       CLOSE 1
2660 RETURN
```

9.2.5 Object Catcher

```
1000 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
1010 '³ CATCHING FOR OBJECTS                    ³
1020 '³ Interpreter: GWBASIC.exe                ³
1030 '³ version    : 21-01-1997                 ³
1040 '³ Source     : catcher.bas                ³
1050 '³ Executable : catcher.exe                ³
1060 '³ For        : NS Geodesy en InfraData    ³
1070 '³ Author     : Joop W. BLOM               ³
1080 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
1090 '
1100 '
1110 '
1120 '
1130 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
1140 '³ DEFINITIONS OF THE FUNCTIONS     ³
1150 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
1160 '
1170 CLS
1180 '
1190 PI = 3.1415926536#
1200 USED$ = "U"
1210 M = 1
1220 K = 1: N = 0: T = 0: LINENR = 0: PAGENR = 0                ' counters
1230 '
1240 DEF FNL (A, B, P, Q, R) = (P - A) * COS(R) + (Q - B) * SIN(R)    'lambda
1250 DEF FNU (A, B, P, Q, R) = -(P - A) * SIN(R) + (Q - B) * COS(R)   'mu
1260 '
1270 DEF FNDX (A, B, P, Q, R) = (A - P + FNL(A, B, P, Q, R) * COS(R)) ^ 2 + (B - Q +
FNL(A, B, P, Q, R) * SIN(R)) ^ 2
1280 DEF FNDY (A, B, P, Q, R) = (A - P - FNU(A, B, P, Q, R) * SIN(R)) ^ 2 + (B - Q +
FNU(A, B, P, Q, R) * COS(R)) ^ 2
1290 '
1300 DEF FNXRT (A, DX, R) = A + (DY * COS(R) - DX * SIN(R))
1310 DEF FNYRT (B, DY, R) = B + (DY * SIN(R) + DX * COS(R))
1320 DEF FNXLT (A, DX, R) = A - (DY * COS(R) + DX * SIN(R))
1330 DEF FNYLT (B, DY, R) = B - (DY * SIN(R) - DX * COS(R))
1340 DEF FNXRB (A, DX, R) = A + (DY * COS(R) + DX * SIN(R))
1350 DEF FNYRB (B, DY, R) = B + (DY * SIN(R) - DX * COS(R))
1360 DEF FNXLB (A, DX, R) = A - (DY * COS(R) - DX * SIN(R))
1370 DEF FNYLB (B, DY, R) = B - (DY * SIN(R) + DX * COS(R))
1380 '
1390 SHELL "dir c:\analyser\geocode\pnt_r*.txt /b > c:\analyser\lookup\pntrfile.txt"
1400 '
1410 OPEN "c:\analyser\lookup\pntrname.txt" FOR OUTPUT AS 1
1420 '
1430 OPEN "c:\analyser\lookup\pntrfile.txt" FOR INPUT AS 2
1440 '
1450     WHILE NOT EOF(2)
1460               INPUT #2, L$
1470               IF INSTR(L$, ".TXT") > 0 OR INSTR(L$, ".txt") > 0 THEN GEOCODE$ =
GEOCODE$ + MID$(L$, 6, 3): PRINT MID$(L$, 6, 3),
1480     WEND
1490 PRINT : PRINT : PRINT
1500 '
1510     INPUT "Welke Geocode: "; GCODE$
1520     IF INSTR(GEOCODE$, GCODE$) = 0 THEN 1510
1530 '
1540 CLOSE 1, 2
1550 '
1560 PRINT
1570 LINE INPUT "Print the output (y/n)? ", ANSWER$
1580 IF INSTR("YESyes", ANSWER$) > 0 THEN PRINTER = 1 ELSE PRINTER = 0
1590 '
1600 PRINT : PRINT "Processing .............": PRINT
1610 '
1620 INBEST$ = "c:\analyser\geocode\pnt_r" + GCODE$ + ".txt"
1630 PRINT INBEST$
1640 '
1650 GOSUB 6350                                         ' open inbest$       #12
1660 '
1670 GOSUB 5910                                         ' open bp...W_r.txt  #1
1680 GOSUB 5950                                         ' open mp...W_r.txt  #2
1690 GOSUB 5990                                         ' open nr...W_r.txt  #3
1700 GOSUB 6030                                         ' open rt...W_r.txt  #4
```

```
1710 '
1720 GOSUB 2420                                         ' insertion points
1730 '
1740 CLOSE 1, 2, 3, 4, 12
1750 '
1760 IF PRINTER = 1 THEN GOSUB 8260                     'printer initialisation
1770 '
1780 GOSUB 6390                                         ' open points_o.txt  #13
1790 '
1800 GOSUB 6420                                         ' open boxes....txt  #14
1810 '
1820 GOSUB 6450                                         ' open points_o.txt  #15
1830 '
1840 GOSUB 5910                                         ' open bp...W_r.txt  #1
1850 GOSUB 5950                                         ' open mp...W_r.txt  #2
1860 GOSUB 5990                                         ' open nr...W_r.txt  #3
1870 GOSUB 6030                                         ' open rt...W_r.txt  #4
1880 '
1890 GOSUB 2970                                         ' type 001W/001K
1900 '
1910 CLOSE 1, 2, 3, 4
1920 '
1930 GOSUB 5910                                         ' open bp...W_r.txt  #1
1940 GOSUB 5950                                         ' open mp...W_r.txt  #2
1950 GOSUB 5990                                         ' open nr...W_r.txt  #3
1960 GOSUB 6030                                         ' open rt...W_r.txt  #4
1970 '
1980 GOSUB 6070                                         ' open mp...E_r.txt  #5
1990 GOSUB 6110                                         ' open nr...E_r.txt  #6
2000 GOSUB 6150                                         ' open rt...E_r.txt  #7
2010 GOSUB 6190                                         ' open bp...X_r.txt  #8
2020 '
2030 GOSUB 8970                                         ' write availabele 1
2040 '
2050 CLOSE 1, 2, 3, 4, 5, 6, 7, 8
2060 '
2070 GOSUB 6070                                         ' open mp...E_r.txt  #5
2080 GOSUB 6110                                         ' open nr...E_r.txt  #6
2090 GOSUB 6150                                         ' open rt...E_r.txt  #7
2100 '
2110 GOSUB 4320                                         ' type 001E/001H
2120 '
2130 CLOSE 5, 6, 7
2140 '
2150 GOSUB 5910                                         ' open bp...W_r.txt  #1
2160 GOSUB 6070                                         ' open mp...E_r.txt  #5
2170 GOSUB 6110                                         ' open nr...E_r.txt  #6
2180 GOSUB 6150                                         ' open rt...E_r.txt  #7
2190 '
2200 GOSUB 6190                                         ' open bp...X_r.txt  #8
2210 GOSUB 6230                                         ' open mp...X_r.txt  #9
2220 GOSUB 6270                                         ' open nr...X_r.txt  #10
2230 GOSUB 6310                                         ' open rt...X_r.txt  #11
2240 '
2250 GOSUB 9580                                         ' write availabele 2
2260 '
2270 CLOSE 8, 9, 10, 11
2280 '
2290 GOSUB 6190                                         ' open bp...X_r.txt  #8
2300 GOSUB 6230                                         ' open mp...X_r.txt  #9
2310 GOSUB 6270                                         ' open nr...X_r.txt  #10
2320 GOSUB 6310                                         ' open rt...X_r.txt  #11
2330 '
2340 IF PRINTER = 1 THEN GOSUB 10810                    ' print the problems
2350 '
2360 END
2370 '
2380 '
2390 '
2400 '
2410 ' ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
2420 ' ³Insertion points                                   ³
2430 ' ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
2440 I1 = 1: I2 = 1: I3 = 1: I4 = 1: I12 = 1
2450       WHILE NOT EOF(12)
2460                   Z = 0
2470           GET 12, I12
2480                   IF EOF(12) THEN 2560
```

```
2490                    I12 = I12 + 1
2500                    IF SP12$ = " 017L" THEN Z = 1
2510                    IF INSTR(SP12$, "001") <> 0 THEN Z = 2
2520                    IF INSTR(SP12$, "1:") <> 0 THEN Z = 4
2530                    IF Z = 0 THEN Z = 3
2540            ON Z GOSUB 2590, 2680, 2770, 2860
2550        WEND
2560    RETURN
2570 '
2580 ' ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
2590 ' ³Begin of the point                                                    ³
2600 ' ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
2610 '
2620    RSET SIGN1$ = "A": RSET X1$ = X12$: RSET Y1$ = Y12$: RSET SP1$ = SP12$: RSET
ROT1$ = ROT12$
2630    PUT 1, I1
2640    I1 = I1 + 1
2650 RETURN
2660 '
2670 ' ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
2680 ' ³Math. point of the point                                              ³
2690 ' ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
2700 '
2710    RSET SIGN2$ = "A": RSET X2$ = X12$: RSET Y2$ = Y12$: RSET SP2$ = SP12$: RSET
ROT2$ = ROT12$
2720    PUT 2, I2
2730    I2 = I2 + 1
2740 RETURN
2750 '
2760 ' ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
2770 ' ³Number of the point                                                   ³
2780 ' ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
2790 '
2800    RSET SIGN3$ = "A": RSET X3$ = X12$: RSET Y3$ = Y12$: RSET SP3$ = SP12$: RSET
ROT3$ = ROT12$
2810    PUT 3, I3
2820    I3 = I3 + 1
2830 RETURN
2840 '
2850 ' ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
2860 ' ³Ratio of the point                                                    ³
2870 ' ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
2880 '
2890    RSET SIGN4$ = "A": RSET X4$ = X12$: RSET Y4$ = Y12$: RSET SP4$ = SP12$: RSET
ROT4$ = ROT12$
2900    PUT 4, I4
2910    I4 = I4 + 1
2920 RETURN
2930 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
2940 '³ POINTS of the type 001W                                              ³
2950 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
2960 '
2970 WHILE NOT EOF(1)
2980 '
2990 'INTRO 1
3000 '=======
3010            P1 = 0: Q1 = 0
3020            USED1 = 0: USED2 = 0: USED3 = 0: USED4 = 0
3030            XBP$ = "": YBP$ = "": BPP$ = ""
3040 '
3050            XMP$ = "": YMP$ = "": TMP$ = ""
3060 '
3070            XNR$ = "": YNR$ = "": PNR$ = ""
3080 '
3090            XRT$ = "": YRT$ = "": RAT$ = "": ROT$ = ""
3100 '
3110  BOX = 1
3120 '=======
3130                    GET 1, M
3140                    XBP$ = X1$: YBP$ = Y1$: BPP$ = SP1$: ROT$ = ROT1$
3150                    A = VAL(XBP$): B = VAL(YBP$): R = VAL(ROT$) / 180 * PI
3160                    DX = 1: DY = 40
3170                    GOSUB 7130                              ' boundary box
3180                    GOSUB 7270                              ' extrema
3190                    MIN1 = MINX
3200                    MAX1 = MAXX
3210                    L1 = 1: H1 = LOF(2) / 30: OK1 = 0: M1 = 1
3220 '
```

```
3230    WHILE NOT EOF(2) AND OK1 <> 1
3240            K = M1: M1 = INT((L1 + H1) / 2)
3250            IF M1 = 0 THEN M1 = 1
3260            GET 2, M1
3270                    IF VAL(X2$) >= MIN1 AND VAL(X2$) <= MAX1 THEN OK1 = 1
3280                    IF VAL(X2$) > MAX1 THEN H1 = M1
3290                    IF VAL(X2$) < MIN1 THEN L1 = M1
3300                    IF M1 = K THEN OK1 = 1
3310    WEND
3320 '
3330    WHILE M1 > 0 AND VAL(X2$) >= MIN1 AND NOT EOF(2)
3340            GET 2, M1
3350            M1 = M1 - 1
3360    WEND
3370 '
3380            IF M1 = 0 THEN M1 = 1
3390 '
3400    WHILE VAL(X2$) <= MAX1 AND NOT EOF(2)
3410            GET 2, M1
3420            P = VAL(X2$): Q = VAL(Y2$)
3430            GOSUB 6490                                      ' in/out box1
3440            M1 = M1 + 1
3450    WEND
3460 '
3470    IF USED1 * USED2 = 0 AND NOT EOF(1) THEN GOSUB 11680    ' correction box
3480 '
3490  BOX = 2
3500 '=======
3510            IF (P1 = 0 OR Q1 = 0) THEN 4240
3520 '
3530            A1 = A: B1 = B
3540            A = (A1 + P1) / 2: B = (B1 + Q1) / 2
3550            DX = 4: DY = (((P1 - A1) / 2) ^ 2 + ((Q1 - B1) / 2) ^ 2) ^ .5
3560            GOSUB 7130                                      ' boundary box
3570            GOSUB 7270                                      ' extrema
3580            MIN2 = MINX
3590            MAX2 = MAXX
3600            L2 = 1: H2 = LOF(3) / 30: OK2 = 0: M2 = 1
3610 '
3620    WHILE NOT EOF(3) AND OK2 <> 1
3630            K = M2: M2 = INT((L2 + H2) / 2)
3640            IF M2 = 0 THEN M2 = 1
3650            GET 3, M2
3660                    IF VAL(X3$) >= MIN2 AND VAL(X3$) <= MAX2 THEN OK2 = 1
3670                    IF VAL(X3$) > MAX2 THEN H2 = M2
3680                    IF VAL(X3$) < MIN2 THEN L2 = M2
3690                    IF M2 = K THEN OK2 = 1
3700    WEND
3710 '
3720    WHILE M2 > 0 AND VAL(X3$) >= MIN2 AND NOT EOF(3)
3730            GET 3, M2
3740            M2 = M2 - 1
3750    WEND
3760 '
3770            IF M2 = 0 THEN M2 = 1
3780 '
3790    WHILE VAL(X3$) <= MAX2 AND NOT EOF(3)
3800            GET 3, M2
3810            P = VAL(X3$): Q = VAL(Y3$)
3820            GOSUB 6490                                      ' in/out box2
3830            M2 = M2 + 1
3840    WEND
3850 '
3860    IF USED3 = 0 AND NOT EOF(1) THEN GOSUB 11680            ' correction box
3870 '
3880  BOX = 3
3890 '=======
3900            A = -2 * A1 + 3 * P1: B = -2 * B1 + 3 * Q1
3910            DX = 4: DY = 2 * (((P1 - A1) / 2) ^ 2 + ((Q1 - B1) / 2) ^ 2) ^ .5
3920            GOSUB 7130                                      ' boundary box
3930            GOSUB 7270                                      ' extrema
3940            MIN3 = MINX
3950            MAX3 = MAXX
3960            L3 = 1: H3 = LOF(4) / 30: OK3 = 0: M3 = 1
3970 '
3980    WHILE NOT EOF(4) AND OK3 <> 1
3990            K = M3: M3 = INT((L3 + H3) / 2)
4000            IF M3 = 0 THEN M3 = 1
```

```
4010            GET 4, M3
4020                    IF VAL(X4$) >= MIN3 AND VAL(X4$) <= MAX3 THEN OK3 = 1
4030                    IF VAL(X4$) > MAX3 THEN H3 = M3
4040                    IF VAL(X4$) < MIN3 THEN L3 = M3
4050                    IF M3 = K THEN OK3 = 1
4060    WEND
4070 '
4080    WHILE M3 > 0 AND VAL(X4$) >= MIN3 AND NOT EOF(4)
4090            GET 4, M3
4100            M3 = M3 - 1
4110    WEND
4120 '
4130            IF M3 = 0 THEN M3 = 1
4140 '
4150    WHILE VAL(X4$) <= MAX3 AND NOT EOF(4)
4160            GET 4, M3
4170            P = VAL(X4$): Q = VAL(Y4$)
4180            GOSUB 6490                                      ' in/out box3
4190            M3 = M3 + 1
4200    WEND
4210 '
4220    IF USED4 = 0 AND NOT EOF(1) THEN GOSUB 11680            ' correction box
4230 '
4240    IF NOT EOF(1) THEN GOSUB 7560                           ' write what is matching
4250    IF NOT EOF(1) AND PRINTER = 1 THEN GOSUB 7660           ' print what is matching
4260    M = M + 1
4270 WEND
4280    IF PRINTER = 1 THEN GOSUB 10670                         ' form feed
4290    RETURN
4300 '
4310 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
4320 '³ POINTS of the type 001E, 001H   ³
4330 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
4340 '
4350 M = 1
4360 '
4370 WHILE NOT EOF(5)
4380 '
4390 'INTRO 2
4400 '=======
4410            XMP$ = "": YMP$ = "": TMP1$ = ""
4420 '
4430            XNR1$ = "": YNR1$ = "": PNR1$ = ""
4440            XNR2$ = "": YNR2$ = "": PNR2$ = ""
4450 '
4460            XRT1$ = "": YRT1$ = "": PRT1$ = ""
4470            XRT2$ = "": YRT2$ = "": PRT2$ = "": ROT$ = ""
4480 '
4490            USED5 = 0: USED6A = 0: USED6B = 0: USED7A = 0: USED7B = 0
4500 '
4510  BOX = 4
4520 '=======
4530            RADOUT = 18: RADIN = 0
4540 '
4550            L4 = 1: H4 = LOF(5) / 30: OK4 = 0: M4 = 1
4560            DX = 1: DY = (RADOUT + RADIN) / 2
4570 '
4580            GET 5, M
4590            IF SP5$ = " 001W" OR SP5$ = " 001K" THEN 5820
4600            MP$ = SIGN5$: XMP$ = X5$: YMP$ = Y5$: TMP$ = SP5$
4610            A = VAL(XMP$): B = VAL(YMP$)
4620            MIN4 = A - RADOUT: MAX4 = A
4630 '
4640    WHILE NOT EOF(6) AND OK4 <> 1
4650            K = M4: M4 = INT((L4 + H4) / 2)
4660            IF M4 = 0 THEN M4 = 1
4670            GET 6, M4
4680                    IF VAL(X6$) >= MIN4 AND VAL(X6$) <= MAX4 THEN OK4 = 1
4690                    IF VAL(X6$) > MAX4 THEN H4 = M4
4700                    IF VAL(X6$) < MIN4 THEN L4 = M4
4710                    IF M4 = K THEN OK4 = 1
4720    WEND
4730 '
4740    WHILE M4 > 0 AND VAL(X6$) >= MIN4 AND NOT EOF(6)
4750            GET 6, M4
4760            M4 = M4 - 1
4770    WEND
4780 '
```

```
4790            IF M4 = 0 THEN M4 = 1
4800 '
4810 START4 = M4: DX = DX + .5: INBOX = 0: P = 0: Q = 0
4820 '
4830    WHILE INBOX = 0 AND P < MAX4 AND NOT EOF(6)
4840            GET 6, START4
4850            P = VAL(X6$): Q = VAL(Y6$): ROT$ = ROT6$
4860            IF INBOX = 0 THEN INDONUT = 0
4870            IF INDONUT = 0 THEN GOSUB 7460                  ' in/out donut
4880            IF INDONUT = 1 AND (Q - B) * SIN(R) <= (A - P) * COS(R) THEN GOSUB
11820: GOSUB 6490'in/out box4
4890            IF INBOX = 0 THEN START4 = START4 + 1
4900            IF INBOX = 0 THEN A = VAL(XMP$): B = VAL(YMP$)
4910    WEND
4920 '
4930 IF INBOX = 0 AND DX < 4.5 THEN 4810
4940 '
4950 IF USED5 * USED6A = 0 THEN GOSUB 11820: GOSUB 11680        ' correction box
4960 '
4970  BOX = 5
4980 '=======
4990            RADOUT = 18: RADIN = 0
5000 '
5010            L5 = 1: H5 = LOF(6) / 30: OK5 = 0: M5 = 1
5020            DX = 1: DY = (RADOUT + RADIN) / 2
5030            A = VAL(XMP$): B = VAL(YMP$)
5040            MIN5 = A: MAX5 = A + RADOUT
5050 '
5060 '
5070    WHILE NOT EOF(6) AND OK5 <> 1
5080            K = M5: M5 = INT((L5 + H5) / 2)
5090            IF M5 = 0 THEN M5 = 1
5100            GET 6, M5
5110                    IF VAL(X6$) >= MIN5 AND VAL(X6$) <= MAX5 THEN OK5 = 1
5120                    IF VAL(X6$) > MAX5 THEN H5 = M5
5130                    IF VAL(X6$) < MIN5 THEN L5 = M5
5140                    IF M5 = K THEN OK5 = 1
5150    WEND
5160 '
5170    WHILE M5 > 0 AND VAL(X6$) >= MIN5 AND NOT EOF(6)
5180            GET 6, M5
5190            IF VAL(X6$) >= MIN5 THEN M5 = M5 - 1
5200    WEND
5210 '
5220 IF M5 = 0 THEN M5 = 1
5230 '
5240 START5 = M5: DX = DX + .5: INBOX = 0: P = 0: Q = 0
5250 '
5260    WHILE INBOX = 0 AND P < MAX5 AND NOT EOF(6)
5270            GET 6, START5
5280            P = VAL(X6$): Q = VAL(Y6$): ROT$ = ROT6$
5290            IF INBOX = 0 THEN INDONUT = 0
5300            IF INDONUT = 0 THEN GOSUB 7460                  ' in/out donut
5310            IF INDONUT = 1 AND (Q - B) * SIN(R) > (A - P) * COS(R) THEN GOSUB 11900:
GOSUB 6490 ' in/out box5
5320            IF INBOX = 0 THEN START5 = START5 + 1
5330            IF INBOX = 0 THEN A = VAL(XMP$): Y = VAL(YMP$)
5340    WEND
5350 '
5360 IF INBOX = 0 AND DX < 4.5 THEN 5240
5370 '
5380 IF USED5 * USED6B = 0 THEN GOSUB 11900: GOSUB 11680        ' correction box
5390 '
5400  BOX = 6
5410 '=======
5420            A = VAL(XMP$): B = VAL(YMP$)
5430            DX = 2.5: DY = 30: INBOX = 0
5440            GOSUB 7130                                      ' boundary box6
5450            GOSUB 7270                                      ' extrema  box6
5460            MIN6 = MINX
5470            MAX6 = MAXX
5480            L6 = 1: H6 = LOF(7) / 30: OK6 = 0: M6 = 1
5490    '
5500    '
5510    WHILE NOT EOF(7) AND OK6 <> 1
5520            K = M6: M6 = INT((L6 + H6) / 2)
5530            IF M6 = 0 THEN M6 = 1
5540            GET 7, M6
```

```
5550                    IF VAL(X7$) >= MIN6 AND VAL(X7$) <= MAX6 THEN OK6 = 1
5560                    IF VAL(X7$) > MAX6 THEN H6 = M6
5570                    IF VAL(X7$) < MIN6 THEN L6 = M6
5580                    IF M6 = K THEN OK6 = 1
5590    WEND
5600 '
5610    WHILE M6 > 0 AND VAL(X7$) >= MIN6 AND NOT EOF(7)
5620            GET 7, M6
5630            M6 = M6 - 1
5640    WEND
5650 '
5660            IF M6 = 0 THEN M6 = 1
5670 '
5680    WHILE VAL(X7$) < MAX6 AND NOT EOF(7)
5690            GET 7, M6
5700            P = VAL(X7$): Q = VAL(Y7$)
5710            IF (INBOX = 1 AND (Q - B) * SIN(R) > (A - P) * COS(R)) THEN DX = 2.5
5720            GOSUB 6490                                   ' in/out box6
5730            M6 = M6 + 1
5740    WEND
5750 '
5760    IF USED7A = 0 THEN GOSUB 11990: GOSUB 11680          ' correction box 6 left
5770    IF USED7B = 0 THEN GOSUB 12070: GOSUB 11680          ' correction box 6 right
5780 '
5790    IF PRINTER = 1 AND T = 0 AND EOF(5) THEN GOSUB 10360: GOSUB 10670 '
header/footer
5800    IF NOT EOF(5) THEN GOSUB 7790                        ' write what is matching
5810    IF PRINTER = 1 AND NOT EOF(5) THEN GOSUB 7900        ' write what is matching
5820    M = M + 1
5830 WEND
5840    IF PRINTER = 1 AND NOT T = 0 THEN GOSUB 10670        ' form feed
5850 RETURN
5860 '
5870 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
5880 '³ OPENING OF THE FILES              ³
5890 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
5900 '
5910    OPEN "c:\analyser\points_w\bp" + GCODE$ + "W_r.txt" FOR RANDOM AS 1 LEN = 30
5920    FIELD #1, 1 AS SIGN1$, 10 AS X1$, 10 AS Y1$, 5 AS SP1$, 4 AS ROT1$
5930 RETURN
5940 '
5950    OPEN "c:\analyser\points_w\mp" + GCODE$ + "W_r.txt" FOR RANDOM AS 2 LEN = 30
5960    FIELD #2, 1 AS SIGN2$, 10 AS X2$, 10 AS Y2$, 5 AS SP2$, 4 AS ROT2$
5970 RETURN
5980 '
5990    OPEN "c:\analyser\points_w\nr" + GCODE$ + "W_r.txt" FOR RANDOM AS 3 LEN = 30
6000    FIELD #3, 1 AS SIGN3$, 10 AS X3$, 10 AS Y3$, 5 AS SP3$, 4 AS ROT3$
6010 RETURN
6020 '
6030    OPEN "c:\analyser\points_w\rt" + GCODE$ + "W_r.txt" FOR RANDOM AS 4 LEN = 30
6040    FIELD #4, 1 AS SIGN4$, 10 AS X4$, 10 AS Y4$, 5 AS SP4$, 4 AS ROT4$
6050 RETURN
6060 '
6070    OPEN "c:\analyser\points_e\mp" + GCODE$ + "E_r.txt" FOR RANDOM AS 5 LEN = 30
6080    FIELD #5, 1 AS SIGN5$, 10 AS X5$, 10 AS Y5$, 5 AS SP5$, 4 AS ROT5$
6090 RETURN
6100 '
6110    OPEN "c:\analyser\points_e\nr" + GCODE$ + "E_r.txt" FOR RANDOM AS 6 LEN = 30
6120    FIELD #6, 1 AS SIGN6$, 10 AS X6$, 10 AS Y6$, 5 AS SP6$, 4 AS ROT6$
6130 RETURN
6140 '
6150    OPEN "c:\analyser\points_e\rt" + GCODE$ + "E_r.txt" FOR RANDOM AS 7 LEN = 30
6160    FIELD #7, 1 AS SIGN7$, 10 AS X7$, 10 AS Y7$, 5 AS SP7$, 4 AS ROT7$
6170 RETURN
6180 '
6190    OPEN "c:\analyser\points_x\bp" + GCODE$ + "X_r.txt" FOR RANDOM AS 8 LEN = 30
6200    FIELD #8, 1 AS SIGN8$, 10 AS X8$, 10 AS Y8$, 5 AS SP8$, 4 AS ROT8$
6210 RETURN
6220 '
6230    OPEN "c:\analyser\points_x\mp" + GCODE$ + "X_r.txt" FOR RANDOM AS 9 LEN = 30
6240    FIELD #9, 1 AS SIGN9$, 10 AS X9$, 10 AS Y9$, 5 AS SP9$, 4 AS ROT9$
6250 RETURN
6260 '
6270    OPEN "c:\analyser\points_x\nr" + GCODE$ + "X_r.txt" FOR RANDOM AS 10 LEN = 30
6280    FIELD #10, 1 AS SIGN10$, 10 AS X10$, 10 AS Y10$, 5 AS SP10$, 4 AS ROT10$
6290 RETURN
6300 '
6310    OPEN "c:\analyser\points_x\rt" + GCODE$ + "X_r.txt" FOR RANDOM AS 11 LEN = 30
```

```
6320    FIELD #11, 1 AS SIGN11$, 10 AS X11$, 10 AS Y11$, 5 AS SP11$, 4 AS ROT11$
6330 RETURN
6340 '
6350    OPEN INBEST$ FOR RANDOM AS 12 LEN = 29
6360    FIELD #12, 10 AS X12$, 10 AS Y12$, 5 AS SP12$, 4 AS ROT12$
6370 RETURN
6380 '
6390    OPEN "c:\analyser\points_o\pnt_o" + GCODE$ + ".txt" FOR OUTPUT AS 13
6400 RETURN
6410 '
6420    OPEN "c:\analyser\boxes\txt\boxes" + GCODE$ + ".txt" FOR OUTPUT AS 14
6430 RETURN
6440 '
6450    OPEN "c:\analyser\objects\objp_" + GCODE$ + ".txt" FOR OUTPUT AS 15
6460 RETURN
6470 '
6480 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
6490 '³ IN/OUT BOX                            ³
6500 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
6510 '
6520            DX1 = FNDX(A, B, P, Q, R)
6530            DY1 = FNDY(A, B, P, Q, R)
6540 '
6550            IF NOT (DX1 <= DX ^ 2 AND DY1 <= DY ^ 2) THEN RETURN
6560 '
6570            ON BOX GOSUB 6620, 6700, 6770, 6840, 6930, 7020
6580    RETURN
6590 '
6600 'Box1
6610 '
6620            DY = DY1 ^ .5: P1 = P: Q1 = Q
6630            XMP$ = X2$: YMP$ = Y2$: TMP$ = SP2$
6640            USED1 = M
6650            USED2 = M1
6660    RETURN
6670 '
6680 'Box2
6690 '
6700            DX = DX1 ^ .5: DY = DY1 ^ .5
6710            XNR$ = X3$: YNR$ = Y3$: PNR$ = SP3$
6720            USED3 = M2
6730    RETURN
6740 '
6750 'Box3
6760 '
6770            DX = DX1 ^ .5: DY = DY1 ^ .5
6780            XRT$ = X4$: YRT$ = Y4$: RAT$ = SP4$
6790            USED4 = M3
6800    RETURN
6810 '
6820 'Box=4
6830 '
6840            DX = DX1 ^ .5
6850            INBOX = 1
6860            USED5 = M
6870            XNR1$ = X6$: YNR1$ = Y6$: PNR1$ = SP6$
6880            USED6A = START4
6890    RETURN
6900 '
6910 'Box=5
6920 '
6930            INBOX = 1
6940            DX = DX1 ^ .5
6950            XNR2$ = X6$: YNR2$ = Y6$: PNR2$ = SP6$
6960            USED6B = START5
6970    RETURN
6980 '
6990 'Box=6
7000 '
7010            DX = DX1 ^ .5
7020            IF (Q - B) * SIN(R) < (A - P) * SIN(R) THEN GOSUB 7050
7030            IF (Q - B) * SIN(R) > (A - P) * SIN(R) THEN GOSUB 7080
7040    RETURN
7050            INBOX = 1: XRT1$ = X7$: YRT1$ = Y7$: PRT1$ = SP7$
7060            USED7A = M6
7070    RETURN
7080            INBOX = 2: XRT2$ = X7$: YRT2$ = Y7$: PRT2$ = SP7$
7090            USED7B = M6
```

```
7100    RETURN
7110 '
7120 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
7130 '³ BOUNDARY BOX                      ³
7140 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
7150 '
7160    XRT = FNXRT(A, DX, R)
7170    YRT = FNYRT(B, DY, R)
7180    XLT = FNXLT(A, DX, R)
7190    YLT = FNYLT(B, DY, R)
7200    XRB = FNXRB(A, DX, R)
7210    YRB = FNYRB(B, DY, R)
7220    XLB = FNXLB(A, DX, R)
7230    YLB = FNYLB(B, DY, R)
7240 RETURN
7250 '
7260 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
7270 '³ EXTREMA                           ³
7280 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
7290 '
7300    X(1) = XRT: X(2) = XLT: X(3) = XRB: X(4) = XLB
7310    Y(1) = YRT: Y(2) = YLT: Y(3) = YRB: Y(4) = YLB
7320 '
7330    MAXX = 0: MAXY = 0: MINX = XRT: MINY = YRT
7340 '
7350    FOR I = 1 TO 4
7360            IF X(I) < MINX THEN MINX = X(I)
7370            IF X(I) > MAXX THEN MAXX = X(I)
7380 '
7390            IF Y(I) < MINY THEN MINY = Y(I)
7400            IF Y(I) > MAXY THEN MAXY = Y(I)
7410    NEXT I
7420 '
7430 RETURN
7440 '
7450 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
7460 '³ IN/OUT DONUT                      ³
7470 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
7480 '
7490    IF ((A - P) ^ 2 + (B - Q) ^ 2 <= RADIN ^ 2 OR (A - P) ^ 2 + (B - Q) ^ 2 >=
RADOUT ^ 2) THEN RETURN
7500    INDONUT = 1
7520    R = VAL(ROT$) / 180 * PI
7530 RETURN
7540 '
7550 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
7560 '³ THE MATCHING 001W POINTS          ³
7570 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
7580    N = N + 1
7590    IF USED1 * USED2 * USED3 * USED4 = 0 THEN 7640         ' write the problem
7600    GOSUB 8070                                             ' write the point
7610    GOSUB 8480                                             ' mark what is used
7620    GOSUB 12160                                            ' write the object
7630 RETURN
7640    GOSUB 8210                                             ' write the problem
7650 RETURN
7660    IF LINENR / 57 = INT(LINENR / 57) THEN GOSUB 10230     ' header 1
7670    LINENR = LINENR + 1
7680    LPRINT N;
7690    LPRINT TAB(17); XMP$;
7700    LPRINT TAB(32); YMP$;
7710 ROT$ = ROT6$
7720    LPRINT TAB(49); TMP$;
7730    LPRINT TAB(62); PNR$;
7740    LPRINT TAB(74); RAT$
7750    IF LINENR / 57 = INT(LINENR / 57) THEN GOSUB 10670     ' page forcing
7760 RETURN
7770 '
7780 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
7790 '³ THE MATCHING 001E POINTS          ³
7800 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
7810    N = N + 1: T = T + 1
7830    IF USED5 * USED6A * USED6B * USED7A * USED7B = 0 THEN 7880
7840    GOSUB 8140                                             ' write the point
7850    GOSUB 8700                                             ' mark what is used
7860    GOSUB 12370                                            ' write the object
7870 RETURN
7880    GOSUB 8210                                             ' write the problem
```

```
7890 RETURN
7900    IF LINENR / 57 = INT(LINENR / 57) THEN GOSUB 10360      ' header 2
7910    LINENR = LINENR + 1
7920    LPRINT N;
7930    LPRINT TAB(12); XMP$;
7940    LPRINT TAB(27); YMP$;
7950    LPRINT TAB(39); TMP$;
7960    LPRINT TAB(47); PNR1$;
7970    LPRINT TAB(56); PNR2$;
7980    LPRINT TAB(65); PRT1$;
7990    LPRINT TAB(74); PRT2$
8000    IF LINENR / 57 = INT(LINENR / 57) THEN GOSUB 10670      ' page forcing
8010 RETURN
8020 '
8030 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
8040 '³ Points of the type 001W         ³
8050 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
8060 '
8070    PRINT #13, N; ","; "A"; ","; VAL(XMP$); ","; VAL(YMP$); ","; VAL(ROT$)
8080 RETURN
8090 '
8100 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
8110 '³ Points of the type 001E/H       ³
8120 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
8130 '
8140    PRINT #13, N; ","; "B"; ","; VAL(XMP$); ","; VAL(YMP$); ","; VAL(ROT$)
8150 RETURN
8160 '
8170 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
8180 '³ The points with a problem       ³
8190 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
8200 '
8210    PRINT #13, N; ","; "C"; ","; VAL(XMP$); ","; VAL(YMP$); ","; VAL(ROT$)
8220 RETURN
8230 '
8240 '
8250 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
8260 '³ PRINTER                         ³
8270 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
8280 '
8290    WIDTH LPRINT 140
8300    LPRINT CHR$(27); CHR$(69);                               'reset
8310    LPRINT CHR$(27); "&l26A";                                'A4
8320    LPRINT CHR$(27); "&l0O";                                 'portrait
8330    LPRINT CHR$(27); "&a20l140M";                            'marges
8340    LPRINT CHR$(27); "(10U";                                 'character set PC-8
8350    LPRINT CHR$(27); "(s0P";                                 'portrait
8360    LPRINT CHR$(27); "(s16H";                                'cpi
8370    LPRINT CHR$(27); "(s8V";                                 'points
8380    LPRINT CHR$(27); "(s3T"                                  'typeface Line Printer
8390 RETURN
8400    LPRINT CHR$(12)                                          'form feed
8410 RETURN
8420    LPRINT CHR$(27); "(s1s3B"                                'bold/italic
8430 RETURN
8440    LPRINT CHR$(27); "(s0s0B"                                'normal/upright
8450 RETURN
8460 '
8470 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
8480 '³ MARK WHAT IS USED of type 001W  ³
8490 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
8500 '
8510 'BP001W
8520            GET 1, USED1
8530            RSET SIGN1$ = USED$
8540            PUT 1, USED1
8550 'MP001W
8560            GET 2, USED2
8570            RSET SIGN2$ = USED$
8580            PUT 2, USED2
8590 'NR001W
8600            GET 3, USED3
8610            RSET SIGN3$ = USED$
8620            PUT 3, USED3
8630 'RT001W
8640            GET 4, USED4
8650            RSET SIGN4$ = USED$
8660            PUT 4, USED4
```

```
8670 RETURN
8680 '
8690 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
8700 '³ MARK WHAT IS USED OF TYPE 001E  ³
8710 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
8720 '
8730 'MP001E
8740            GET 5, USED5
8750            RSET SIGN5$ = USED$
8760            PUT 5, USED5
8770 'NR001E A
8780            GET 6, USED6A
8790            RSET SIGN6$ = USED$
8800            PUT 6, USED6A
8810 'RT001E A
8820            GET 7, USED7A
8830            RSET SIGN7$ = USED$
8840            PUT 7, USED7A
8850 'NR001E B
8860            GET 6, USED6B
8870            RSET SIGN6$ = USED$
8880            PUT 6, USED6B
8890 'RT001E B
8900            GET 7, USED7B
8910            RSET SIGN7$ = USED$
8920            PUT 7, USED7B
8930 RETURN
8940 '
8950 '
8960 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
8970 '³ FILES WITH AVAILABLE ELEMENTS 1 ³
8980 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄDÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
8990 '
9000    X = 1: Y1 = 1
9010 WHILE NOT EOF(2)
9020    GET 2, X
9030    X = X + 1
9040    IF SIGN2$ <> "A" THEN 9120
9050            RSET SIGN5$ = SIGN2$
9060            RSET X5$ = X2$
9070            RSET Y5$ = Y2$
9080            RSET SP5$ = SP2$
9090            RSET ROT5$ = ROT2$
9100            PUT 5, Y1
9110            Y1 = Y1 + 1
9120 WEND
9130 '
9140    X = 1: Y1 = 1
9150 WHILE NOT EOF(3)
9160    GET 3, X
9170    X = X + 1
9180    IF SIGN3$ <> "A" THEN 9260
9190            RSET SIGN6$ = SIGN3$
9200            RSET X6$ = X3$
9210            RSET Y6$ = Y3$
9220            RSET SP6$ = SP3$
9230            RSET ROT6$ = ROT3$
9240            PUT 6, Y1
9250            Y1 = Y1 + 1
9260 WEND
9270 '
9280    X = 1: Y1 = 1
9290 WHILE NOT EOF(4)
9300    GET 4, X
9310    X = X + 1
9320    IF SIGN4$ <> "A" THEN 9400
9330            RSET SIGN7$ = SIGN4$
9340            RSET X7$ = X4$
9350            RSET Y7$ = Y4$
9360            RSET SP7$ = SP4$
9370            RSET ROT7$ = ROT4$
9380            PUT 7, Y1
9390            Y1 = Y1 + 1
9400 WEND
9410    X = 1: Y1 = 1
9420 WHILE NOT EOF(1)
9430    GET 1, X
9440    X = X + 1
```

```
9450    IF SIGN1$ <> "A" THEN 9530
9460            RSET SIGN8$ = SIGN1$
9470            RSET X8$ = X1$
9480            RSET Y8$ = Y1$
9490            RSET SP8$ = SP1$
9500            RSET ROT8$ = ROT1$
9510            PUT 8, Y1
9520            Y1 = Y1 + 1
9530 WEND
9540 '
9550 RETURN
9560 '
9570 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
9580 '³ FILES WITH AVAILABLE ELEMENTS 2 ³
9590 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
9600 '
9610    X = 1: Y1 = 1
9620 WHILE NOT EOF(1)
9630    GET 1, X
9640    X = X + 1
9650    IF SIGN5$ <> "A" THEN 9720
9660            RSET SIGN8$ = SIGN1$
9670            RSET X8$ = X1$
9680            RSET Y8$ = Y1$
9690            RSET SP8$ = SP1$
9700            RSET ROT8$ = ROT1$
9710            PUT 1, Y1
9720            Y1 = Y1 + 1
9730 WEND
9740 '
9750    X = 1: Y1 = 1
9760 WHILE NOT EOF(5)
9770    GET 5, X
9780    X = X + 1
9790    IF SIGN5$ <> "A" THEN 9870
9800            RSET SIGN9$ = SIGN5$
9810            RSET X9$ = X5$
9820            RSET Y9$ = Y5$
9830            RSET SP9$ = SP5$
9840            RSET ROT9$ = ROT5$
9850            PUT 9, Y1
9860            Y1 = Y1 + 1
9870 WEND
9880 '
9890    X = 1: Y1 = 1
9900 WHILE NOT EOF(6)
9910    GET 6, X
9920    X = X + 1
9930    IF SIGN6$ <> "A" THEN 10010
9940            RSET SIGN10$ = SIGN6$
9950            RSET X10$ = X6$
9960            RSET Y10$ = Y6$
9970            RSET SP10$ = SP6$
9980            RSET ROT10$ = ROT6$
9990            PUT 10, Y1
10000            Y1 = Y1 + 1
10010 WEND
10020 '
10030    X = 1: Y1 = 1
10040 WHILE NOT EOF(7)
10050    GET 7, X
10060    X = X + 1
10070    IF SIGN7$ <> "A" THEN 10150
10080            RSET SIGN11$ = SIGN7$
10090            RSET X11$ = X7$
10100            RSET Y11$ = Y7$
10110            RSET SP11$ = SP7$
10120            RSET ROT11$ = ROT7$
10130            PUT 11, Y1
10140            Y1 = Y1 + 1
10150 WEND
10160 '
10170 RETURN
10180 '
10190 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
10200 '³ HEADER 1                        ³
10210 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
10220 '
```

```
10230   GOSUB 8420                                               'bold on/italic
10240   LPRINT "þþþþþ Objects of the type 001W";
10250   LPRINT TAB(67); "Geocode: "; GCODE$
10260   LPRINT STRING$(78, 196): LINENR = 4
10270   LPRINT "Number";
10280   LPRINT TAB(15); "X-co"rdinate";
10290   LPRINT TAB(30); "Y-co"rdinate";
10300   LPRINT TAB(50); "Type";
10310   LPRINT TAB(62); "Point";
10320   LPRINT TAB(74); "Ratio"
10330   GOSUB 8440                                               'normal/upright
10340 RETURN
10350 '
10360 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
10370 '³ HEADER 2                           ³
10380 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
10390 '
10400   GOSUB 8420                                               'bold on/italic
10410   LPRINT "þþþþþ Objects of the type 001E/H";
10420   LPRINT TAB(67); "Geocode: "; GCODE$
10430   LPRINT STRING$(78, 196): LINENR = 4
10440   LPRINT "Number";
10450   LPRINT TAB(10); "X-co"rdinate";
10460   LPRINT TAB(25); "Y-co"rdinate";
10470   LPRINT TAB(40); "Type";
10480   LPRINT TAB(46); "LPoint";
10490   LPRINT TAB(55); "RPoint";
10500   LPRINT TAB(64); "LRatio";
10510   LPRINT TAB(73); "RRatio"
10520   GOSUB 8440                                               'normal/upright
10530 RETURN
10540 '
10550 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
10560 '³ HEADER 3                           ³
10570 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
10580 '
10590   GOSUB 8420                                               'bold on/italic
10600   LPRINT "þþþþþ Problem points of the type 001E/H/K/W";
10610   LPRINT TAB(67); "Geocode: "; GCODE$
10620   LPRINT STRING$(78, 196): LINENR = 3
10630   GOSUB 8440                                               'normal/upright
10640 RETURN
10650 '
10660 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
10670 '³ FOOTER                             ³
10680 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
10690 '
10700 WHILE LINENR < 57
10710   LPRINT : LINENR = LINENR + 1
10720 WEND
10730   GOSUB 8420                                               'bold on/italic
10740   LPRINT STRING$(78, 196): PAGENR = PAGENR + 1
10750   LPRINT "Joop W. BLOM þ for NS Geodesy and InfraData þ "; TIME$; " þ "; DATE$; "
þ Page  "; : LPRINT USING "##"; PAGENR
10760   GOSUB 8440                                               'normal/upright
10770   LPRINT CHR$(12)
10780 RETURN
10790 '
10800 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
10810 '³ THE PROBLEM POINTS                 ³
10820 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
10830 '
10840   GOSUB 10560                                              ' header 3
10850 '
10860   LPRINT "þþþþþ The begin points": LPRINT
10870   LINENR = LINENR + 2
10880   X = 1
10890 WHILE NOT EOF(8)
10900   GET 8, X
10910   IF NOT EOF(8) THEN LPRINT X;
10920   LPRINT TAB(6); SIGN8$;
10930   LPRINT TAB(25); X8$;
10940   LPRINT TAB(40); Y8$;
10950   LPRINT TAB(55); SP8$;
10960   LPRINT TAB(65); ROT8$
10970   X = X + 1
10980   LINENR = LINENR + 1
10990   IF LINENR > 53 THEN GOSUB 10670: GOSUB 10560             ' footer/header
```

```
11000 WEND
11010 '
11020 '
11030 '
11040 '
11050   LPRINT "þþþþþ The mathematical points": LPRINT
11060   X = 1
11070   LINENR = LINENR + 2
11080 WHILE NOT EOF(9)
11090   GET 9, X
11100   IF NOT EOF(9) THEN LPRINT X;
11110   LPRINT TAB(6); SIGN9$;
11120   LPRINT TAB(25); X9$;
11130   LPRINT TAB(40); Y9$;
11140   LPRINT TAB(55); SP9$;
11150   LPRINT TAB(65); ROT9$
11160   X = X + 1
11170   LINENR = LINENR + 1
11180   IF LINENR > 53 THEN GOSUB 10670: GOSUB 10560            ' footer/header
11190 WEND
11200 '
11210 '
11220 '
11230 '
11240   LPRINT "þþþþþ The numbers": LPRINT
11250   LINENR = LINENR + 2
11260   X = 1
11270 WHILE NOT EOF(10)
11280   GET 10, X
11290   IF NOT EOF(10) THEN LPRINT X;
11300   LPRINT TAB(6); SIGN10$;
11310   LPRINT TAB(25); X10$;
11320   LPRINT TAB(40); Y10$;
11330   LPRINT TAB(55); SP10$;
11340   LPRINT TAB(65); ROT10$
11350   X = X + 1
11360   LINENR = LINENR + 1
11370   IF LINENR > 53 THEN GOSUB 10670: GOSUB 10560            ' footer/header
11380 WEND
11390 '
11400 '
11410 '
11420 '
11430   LPRINT "þþþþþ The ratios": LPRINT
11440   LINENR = LINENR + 2
11450   X = 1
11460 WHILE NOT EOF(11)
11470   GET 11, X
11480   IF NOT EOF(11) THEN LPRINT X;
11490   LPRINT TAB(6); SIGN11$;
11500   LPRINT TAB(25); X11$;
11510   LPRINT TAB(40); Y11$;
11520   LPRINT TAB(55); SP11$;
11530   LPRINT TAB(65); ROT11$
11540   X = X + 1
11550   LINENR = LINENR + 1
11560   IF LINENR > 53 THEN GOSUB 10670: GOSUB 10560            ' footer/header
11570 WEND
11580 '
11590 '
11600 '
11610 '
11620   GOSUB 10670                                               ' footer
11630 '
11640 '
11650 RETURN
11660 '
11670 ' ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
11680 ' ³Correction boxes                                  ³
11690 ' ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
11700   PRINT #14, USING "######.###"; A;
11710   PRINT #14, ",";
11720   PRINT #14, USING "######.###"; B;
11730   PRINT #14, ",";
11740   PRINT #14, USING "##.#"; DX;
11750   PRINT #14, ",";
11760   IF BOX < 6 THEN PRINT #14, USING "##.#"; DY;
11770   IF BOX = 6 THEN PRINT #14, USING "##.#"; .5 * DY;
```

```
11780   PRINT #14, ",";
11790   IF BOX < 4 THEN PRINT #14, USING "###"; VAL(ROT1$)
11800   IF BOX > 3 THEN PRINT #14, USING "###"; VAL(ROT6$)
11810 RETURN
11820 '
11830 '         BOX=4
11840 '
11850           R = VAL(ROT6$) / 180 * PI
11860           A = VAL(XMP$): B = VAL(YMP$)
11870           A = A - .5 * RADOUT * COS(R)
11880           B = B - .5 * RADOUT * SIN(R)
11890 RETURN
11900 '
11910 '         BOX=5
11920 '
11930           R = VAL(ROT6$) / 180 * PI
11940           A = VAL(XMP$): B = VAL(YMP$)
11950           A = A + .5 * RADOUT * COS(R)
11960           B = B + .5 * RADOUT * SIN(R)
11970 RETURN
11980 '
11990 '         BOX=6 (left)
12000 '
12010           R = VAL(ROT6$) / 180 * PI
12020           A = VAL(XMP$): B = VAL(YMP$)
12030           A = A - .5 * DY * COS(R)
12040           B = B - .5 * DY * SIN(R)
12050 RETURN
12060 '
12070 '         BOX=6 (right)
12080 '
12090           R = VAL(ROT6$) / 180 * PI
12100           A = VAL(XMP$): B = VAL(YMP$)
12110           A = A + .5 * DY * COS(R)
12120           B = B + .5 * DY * SIN(R)
12130 RETURN
12140 '
12150 ' ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
12160 ' ³ Objects of the type 001W                              ³
12170 ' ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
12180 '
12190   PRINT #15, USING "###"; N;
12200   PRINT #15, ",";
12210   PRINT #15, USING "\ \"; GCODE$;
12220   PRINT #15, ",";
12230   PRINT #15, USING "######.###"; VAL(XMP$);
12240   PRINT #15, ",";
12250   PRINT #15, USING "######.###"; VAL(YMP$);
12260   PRINT #15, ",";
12270   PRINT #15, USING "\   \"; TMP$;
12280   PRINT #15, ",";
12290   PRINT #15, USING "####"; VAL(ROT$);
12300   PRINT #15, ",";
12310   PRINT #15, USING "\   \"; PNR$;
12320   PRINT #15, ",";
12330   PRINT #15, USING "\   \"; RAT$
12340 RETURN
12350 '
12360 ' ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
12370 ' ³ Objects of the type 001E/H                            ³
12380 ' ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
12390 '
12400   PRINT #15, USING "###"; N;
12410   PRINT #15, ",";
12420   PRINT #15, USING "\ \"; GCODE$;
12430   PRINT #15, ",";
12440   PRINT #15, USING "######.###"; VAL(XMP$);
12450   PRINT #15, ",";
12460   PRINT #15, USING "######.###"; VAL(YMP$);
12470   PRINT #15, ",";
12480   PRINT #15, USING "\   \"; TMP$;
12490   PRINT #15, ",";
12500   PRINT #15, USING "####"; VAL(ROT$);
12510   PRINT #15, ",";
12520   PRINT #15, USING "\   \"; PNR1$;
12530   PRINT #15, ",";
12540   PRINT #15, USING "\   \"; PRT1$;
12550   PRINT #15, ",";
```

```
12560   PRINT #15, USING "\   \"; PNR2$;
12570   PRINT #15, ",";
12580   PRINT #15, USING "\   \"; PRT2$
12590 RETURN
```

9.2.6 Solids Generator

```
1000 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
1010 '³ SOLIDS GENERATOR DXF                      ³
1020 '³ Interpreter: GWBASIC.exe                  ³
1030 '³ version    : 31-12-1996                   ³
1040 '³ Source     : writer.bas                   ³
1050 '³ Executable : writer.exe                   ³
1060 '³ For        : NS Geodesy en InfraData      ³
1070 '³ Author     : Joop W. BLOM                 ³
1080 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
1090 '
1100 DEFDBL P-Q
1110 PI = ATN(1) * 4
1120 CLS
1130 SHELL "dir c:\analyser\points_o\pnt_o*.txt /b > c:\analyser\lookup\pntofile.txt"
1140 '
1150 OPEN "c:\analyser\lookup\pntoname.txt" FOR OUTPUT AS 1
1160 '
1170 OPEN "c:\analyser\lookup\pntofile.txt" FOR INPUT AS 2
1180 '
1190      WHILE NOT EOF(2)
1200              INPUT #2, L$
1210              IF INSTR(L$, "TXT") > 0 OR INSTR(L$, "txt") > 0 THEN GEOCODE$ =
GEOCODE$ + MID$(L$, 6, 3): PRINT MID$(L$, 6, 3),
1220     WEND
1230 PRINT : PRINT : PRINT
1240 '
1250     INPUT "Welke Geocode: "; GCODE$
1260     IF INSTR(GEOCODE$, GCODE$) = 0 THEN 1250
1270 '
1280 CLOSE 1, 2
1290 '
1300 PRINT : PRINT "Processing .............": PRINT
1310 '
1320 '
1330 INBEST$ = "c:\analyser\points_o\pnt_o" + GCODE$ + ".txt"
1340 DXFOUT$ = "c:\analyser\dxf_out\gcode" + GCODE$ + ".dxf"
1350 '
1360 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
1370 '³ FUNCTIONS                                 ³
1380 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
1390 '
1400 DEF FNXLB (P, R) = P - COS(R) + SIN(R)
1410 DEF FNYLB (Q, R) = Q - SIN(R) - COS(R)
1420 DEF FNXRB (P, R) = P + COS(R) + SIN(R)
1430 DEF FNYRB (Q, R) = Q + SIN(R) - COS(R)
1440 DEF FNXLT (P, R) = P - COS(R) - SIN(R)
1450 DEF FNYLT (Q, R) = Q - SIN(R) + COS(R)
1460 DEF FNXRT (P, R) = P + COS(R) - SIN(R)
1470 DEF FNYRT (Q, R) = Q + SIN(R) + COS(R)
1480 '
1490 OPEN DXFOUT$ FOR OUTPUT AS 1
1500 '
1510 OPEN INBEST$ FOR INPUT AS 2
1520 '
1530 PRINT #1, 0
1540 PRINT #1, "SECTION"
1550 PRINT #1, 2
1560 PRINT #1, "ENTITIES"
1570 '
1580    WHILE NOT EOF(2)
1590       G = 0: P = 0: Q = 0: R = 0
1600       INPUT #2, NUMBER
1610       INPUT #2, COLOUR$
1620                           IF COLOUR$ = "A" THEN COLOUR = 4
1630                           IF COLOUR$ = "B" THEN COLOUR = 6
1640                           IF COLOUR$ = "C" THEN COLOUR = 1
1650       INPUT #2, P 'X insert
1660       INPUT #2, Q 'Y insert
1670       INPUT #2, G 'Rotation in degrees
1680       R = G * PI / 180'Rotation in Radials
1690 '
1700            PRINT #1, 0
1710            PRINT #1, "SOLID"
1720            PRINT #1, 8
```

```
1730            PRINT #1, "0"
1740            PRINT #1, 6
1750            PRINT #1, "CONTINUOUS"
1760            PRINT #1, 62
1770            PRINT #1, COLOUR
1780 '
1790            'LEFT BOTTOM
1800             XLB = FNXLB(P, R)
1810             YLB = FNYLB(Q, R)
1820                    PRINT #1, 10
1830                    PRINT #1, USING "######.###"; XLB
1840                    PRINT #1, 20
1850                    PRINT #1, USING "######.###"; YLB
1860 '
1870            'RIGHT BOTTOM
1880             XRB = FNXRB(P, R)
1890             YRB = FNYRB(Q, R)
1900                    PRINT #1, 11
1910                    PRINT #1, USING "######.###"; XRB
1920                    PRINT #1, 21
1930                    PRINT #1, USING "######.###"; YRB
1940 '
1950                    IF COLOUR <> 1 THEN LXT = 12: LYT = 22: RXT = 13: RYT = 23:
GOSUB 2030: GOSUB 2120'left/right
1960                    IF COLOUR = 1 THEN LXT = 13: LYT = 23: RXT = 12: RYT = 22: GOSUB
2120: GOSUB 2030'right/left
1970            GOSUB 2210                      'text
1980    WEND
1990            GOSUB 2400                      'foot
2000 '
2010 '
2020 END
2030            'LEFT TOP
2040             XLT = FNXLT(P, R)
2050             YLT = FNYLT(Q, R)
2060                    PRINT #1, LXT
2070                    PRINT #1, USING "######.###"; XLT
2080                    PRINT #1, LYT
2090                    PRINT #1, USING "######.###"; YLT
2100 RETURN
2110 '
2120            'RIGHT TOP
2130             XRT = FNXRT(P, R)
2140             YRT = FNYRT(Q, R)
2150                    PRINT #1, RXT
2160                    PRINT #1, USING "######.###"; XRT
2170                    PRINT #1, RYT
2180                    PRINT #1, USING "######.###"; YRT
2190 RETURN
2200 '
2210            'TEXT
2220             PRINT #1, 0
2230             PRINT #1, "TEXT"
2240             PRINT #1, 8
2250             PRINT #1, "0"
2260                    PRINT #1, 10
2270                    PRINT #1, USING "######.###"; XRT
2280                    PRINT #1, 20
2290                    PRINT #1, USING "######.###"; YRT
2300                    PRINT #1, 30
2310                    PRINT #1, 0
2320                    PRINT #1, 40
2330                    PRINT #1, .75
2340                    PRINT #1, 1
2350                    PRINT #1, RIGHT$(STR$(NUMBER), LEN(STR$(NUMBER)) - 1)
2360                    PRINT #1, 50
2370                    PRINT #1, G
2380 RETURN
2390 '
2400    PRINT #1, 0
2410    PRINT #1, "ENDSEC"
2420    PRINT #1, 0
2430    PRINT #1, "EOF"
2440    CLOSE
2450 RETURN
```

9.2.7 Correction Boxes

```
1000 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
1010 '³ BOXES GENERATOR DXF                      ³
1020 '³ Interpreter: GWBASIC.exe                 ³
1030 '³ version    : 31-12-1996                  ³
1040 '³ Source     : boxes.bas                   ³
1050 '³ Executable : boxes.exe                   ³
1060 '³ For        : NS Geodesy en InfraData ³
1070 '³ Author     : Joop W. BLOM                ³
1080 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
1090 '
1100 DEFDBL X,Y
1110 PI = ATN(1) * 4
1120 CLS
1130 SHELL "dir c:\analyser\boxes\txt\boxes*.txt /b > c:\analyser\lookup\boxfile.txt"
1140 '
1150 OPEN "c:\analyser\lookup\boxname.txt" FOR OUTPUT AS 1
1160 '
1170 OPEN "c:\analyser\lookup\boxfile.txt" FOR INPUT  AS 2
1180 '
1190      WHILE NOT EOF(2)
1200              INPUT #2, L$
1210              IF INSTR(L$, "TXT") > 0 OR INSTR(L$, "txt") > 0 THEN GEOCODE$ =
GEOCODE$ + MID$(L$, 6, 3): PRINT MID$(L$, 6, 3),
1220     WEND
1230 PRINT : PRINT : PRINT
1240 '
1250     INPUT "Welke Geocode: "; GCODE$
1260     IF INSTR(GEOCODE$, GCODE$) = 0 THEN 1250
1270 '
1280 CLOSE 1, 2
1290 '
1300 PRINT : PRINT "Processing .............": PRINT
1310 '
1320 '
1330 INBEST$ = "c:\analyser\boxes\txt\boxes" + GCODE$ + ".txt"
1340 DXFOUT$ = "c:\analyser\boxes\dxf\boxes" + GCODE$ + ".dxf"
1350 '
1360 OPEN INBEST$ FOR INPUT  AS 2
1370 OPEN DXFOUT$ FOR OUTPUT AS 1
1380 '
1390 PRINT #1,0
1400 PRINT #1,"SECTION"
1410 PRINT #1,2
1420 PRINT #1,"BLOCKS"
1430 PRINT #1,0
1440 PRINT #1,"BLOCK"
1450 PRINT #1,8
1460 PRINT #1,"0"
1470 PRINT #1,2
1480 PRINT #1,"BOX"
1490 PRINT #1,70
1500 PRINT #1,64
1510 PRINT #1,10
1520 PRINT #1,0!
1530 PRINT #1,20
1540 PRINT #1,0!
1550 PRINT #1,30
1560 PRINT #1,0!
1570    PRINT #1,0
1580    PRINT #1,"LINE"
1590    PRINT #1,8
1600    PRINT #1,"0"
1610    PRINT #1,10
1620    PRINT #1,1!
1630    PRINT #1,20
1640    PRINT #1,-1!
1650    PRINT #1,30
1660    PRINT #1,0!
1670    PRINT #1,11
1680    PRINT #1,-1!
1690    PRINT #1,21
1700    PRINT #1,-1!
1710    PRINT #1,31
1720    PRINT #1,0!
```

```
1730            PRINT #1,0
1740            PRINT #1,"LINE"
1750            PRINT #1,8
1760            PRINT #1,"0"
1770            PRINT #1,10
1780            PRINT #1,-1!
1790            PRINT #1,20
1800            PRINT #1,-1!
1810            PRINT #1,30
1820            PRINT #1,0!
1830            PRINT #1,11
1840            PRINT #1,-1!
1850            PRINT #1,21
1860            PRINT #1,1!
1870            PRINT #1,31
1880            PRINT #1,0!
1890                    PRINT #1,0
1900                    PRINT #1,"LINE"
1910                    PRINT #1,8
1920                    PRINT #1,"0"
1930                    PRINT #1,10
1940                    PRINT #1,-1!
1950                    PRINT #1,20
1960                    PRINT #1,1!
1970                    PRINT #1,30
1980                    PRINT #1,0!
1990                    PRINT #1,11
2000                    PRINT #1,1!
2010                    PRINT #1,21
2020                    PRINT #1,1!
2030                    PRINT #1,31
2040                    PRINT #1,0!
2050                            PRINT #1,0
2060                            PRINT #1,"LINE"
2070                            PRINT #1,8
2080                            PRINT #1,"0"
2090                            PRINT #1,10
2100                            PRINT #1,1!
2110                            PRINT #1,20
2120                            PRINT #1,1!
2130                            PRINT #1,30
2140                            PRINT #1,0!
2150                            PRINT #1,11
2160                            PRINT #1,1!
2170                            PRINT #1,21
2180                            PRINT #1,-1!
2190                            PRINT #1,31
2200                            PRINT #1,0!
2210                    PRINT #1,0
2220                    PRINT #1,"ENDBLK"
2230                    PRINT #1,8
2240                    PRINT #1,"0"
2250            PRINT #1,0
2260            PRINT #1,"ENDSEC"
2270            PRINT #1,0
2280    PRINT #1,"SECTION"
2290    PRINT #1,2
2300    PRINT #1,"ENTITIES"
2310    WHILE NOT EOF(2)
2320            INPUT #2,XINS,YINS,DX,DY,DEGREES
2330                    PRINT #1,0
2340                    PRINT #1,"INSERT"
2350                    PRINT #1,8
2360                    PRINT #1,"0"
2370                    PRINT #1,2
2380                    PRINT #1,"BOX"
2390                    PRINT #1,10
2400                    PRINT #1,USING "######.###";XINS
2410                    PRINT #1,20
2420                    PRINT #1,USING "######.###";YINS
2430                    PRINT #1,30
2440                    PRINT #1,0!
2450                    PRINT #1,41
2460                    PRINT #1,USING "##.#";DY
2470                    PRINT #1,42
2480                    PRINT #1,USING "##.#";DX
2490                    PRINT #1,50
2500                    PRINT #1,DEGREES
```

```
2510    WEND
2520            PRINT #1,0
2530            PRINT #1,"ENDSEC"
2540            PRINT #1,0
2550            PRINT #1,"EOF"
2560            CLOSE 1,2
2570    END
```

9.2.8 Object Shower

```
1000 'ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
1010 '³ OBJECTFILE SHOWER                       ³
1020 '³ Interpreter: GWBASIC.exe                ³
1030 '³ version     : 01-01-1997                ³
1040 '³ Source      : objects.bas               ³
1050 '³ Executable  : objects.exe               ³
1060 '³ For         : NS Geodesy en InfraData   ³
1070 '³ Author      : Joop W. BLOM              ³
1080 'ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
1090 '
1100 CLS
1110 SHELL "dir c:\analyser\objects\objp_*.txt /b > c:\analyser\lookup\objfile.txt"
1120 '
1130 OPEN "c:\analyser\lookup\objname.txt" FOR OUTPUT AS 1
1140 '
1150 OPEN "c:\analyser\lookup\objfile.txt" FOR INPUT  AS 2
1160 '
1170      WHILE NOT EOF(2)
1180              INPUT #2, L$
1190              IF INSTR(L$, "TXT") > 0 OR INSTR(L$, "txt") > 0 THEN GEOCODE$ =
GEOCODE$ + MID$(L$, 6, 3): PRINT MID$(L$, 6, 3),
1200     WEND
1210 PRINT : PRINT : PRINT
1220 '
1230     INPUT "Welke Geocode: "; GCODE$
1240     IF INSTR(GEOCODE$, GCODE$) = 0 THEN 1230
1250 '
1260 CLOSE 1, 2
1270 '
1280 PRINT : PRINT "Processing ..............": PRINT
1290 '
1300 '
1310 INBEST$ = "c:\analyser\objects\objp_" + GCODE$ + ".txt"
1320 '
1330 OPEN INBEST$ FOR INPUT  AS 2
1340 '
1350 T=1
1360 WHILE NOT EOF(2)
1370 LINE INPUT #2, A$
1380 PRINT A$:T=T+1
1390 IF INT(T/21)=T/21 THEN LINE INPUT "Press <Enter> to continue . . .",X$
1400 WEND
1410 SHELL "pause"
1420 CLOSE
1430 END
```

9.3 Acknowledgements

Now that this thesis has seen the light, I realise that it was only possible for me to do this with the support of a lot of people. I would first of all like to thank my present employer NedGraphics, and also my former employer Rocomp, in particular Ir. Ko Rozema, who encouraged me to attend this course. Then of course the people of the Amsterdam InterGIS Office, Bart for his balanced and constructive comments, and Mathilde for her kind and accurate liaison work. Then , I would like to thank Dr. Ir. Herman Quee, Ing. Rob Gerritsen, and everyone else at the Netherlands Railways department Geodesy and InfraData, who made it possible for me to study the problem of object composing in a railway environment. Specifically with regards to the realisation of this MSc-thesis I would like to thank Prof. Dr. Henk Scholten and Dr. Jim Petch for their inspiring support, good advice, channelling of ideas, and the correction of my grammatical derailments.
The person I should thank most of course is Margriet, who has had a tough time over the past four years, but always managed to keep smiling.

Joop

9.4 Map

A map has been added at the back, which shows the results of the case within a sub-area of geocode 506. The solids with accompanying figures correspond with the tables in *7.9* and *7.10*

9.5 Disk

9.5.1 Instructions

By following the instructions on the label of the enclosed disk at the back, the programs can be installed, after which the case can be completed.

9.5.2 Read Me First

When the installation of ‘Object Composer’ is completed.

The programs can be used in full if ***a printer*** is available and ***on-line***.

If you want to test the programs later on, then you can quit now.

Later on you can run the program menu by typing: ***c:\analyser\programs\nsmenu00.exe***.

This set of programs has been developed for the Netherlands Railways department, Geodesy and InfraData, and is aimed to test ‘Object Composing’.

For the program sequence it is necessary to run Cell Scanner up to Object Shower.

Rail points of the railway yard Amersfoort, in the centre of the Netherlands are point of study.

In NS-terms, rail points of the map sheets 50601a up to 50604a, which form geocode 506.

If ***"Which Geocode?"*** appears, then the input must be ***506***.

The Object Composer menu (c:\analyser\programs\nsmenu00.exe) consists of:

Cell Scanner

Input	c:\analyser\dxf_in\50601a.dxf	**A**
	c:\analyser\dxf_in\50602a.dxf	**B**
	c:\analyser\dxf_in\50603a.dxf	**C**
	c:\analyser\dxf_in\50604a.dxf	**D**
Process	Makes an inventory of the rail typical library elements and reports this via a file and printer output.	
Output	c:\analyser\cells\cells506.txt	
	printer: special report	
	printer: general report	

Spatial Coding

Input	c:\analyser\dxf_in\50601a.dxf
	c:\analyser\dxf_in\50602a.dxf
	c:\analyser\dxf_in\50603a.dxf
	c:\analyser\dxf_in\50604a.dxf
Process	Provides the set of object elements with a spatial code, to prepare spatial ordering.
Output	c:\analyser\spatial\pnt_s506.txt

QuickSort

Input	c:\analyser\spatial\pnt_s506.txt
Process	The result of this sorting by spatial code is a linear accessible spatial ordered file.
Output	c:\analyser\geocode\pnt_r506.txt

Object Catcher

Input	c:\analyser\geocode\pnt_r506.txt
Process	Objects have been composed by successively catching of the matching elements, by repeated application of the algorithms discussed.
Output	c:\analyser\point_o\rpnt_o506.txt
	c:\analyser\boxes\txt\boxes506.txt
	c:\analyser\objects\objp_506.txt

Visualisation

Input	c:\analyser\point_o\rpnt_o506.txt
Process	Generates numbered solids (see 7.9 en 7.10) for each object located via a DXF.
Output	c:\analyser\dxf_out\gcode506.dxf **E**

Correction Boxes

Input	c:\analyser\boxes\txt\boxes506.txt
Process	Generates correction boxes
Output	c:\analyser\boxes\dxf\boxes506.dxf **F**

Object Shower

Input c:\analyser\objects\objp_506.txt

Process Restores the file with the complete objects and shows the table on the screen.

Output screen: table

Map production

Via the import facilities of any CAD program you can combine the DXF's concerned, as follows.

$(A \cup B \cup C \cup D) = X$

$(E \cup F) = Y$

$X \cup Y = Z$

Plotted Map $\subset$ Z (at the back)

Questions

Should you have any difficulties, pleas do not hesitate to contact me.

Telephone: +31 (13) 5136067

Telefax: +31 (13) 5136068

E-mail: joop.blom@pi.net