A Database Management System for Traffic Engineering Departments of Small and Medium-Sized Cities

KARSTEN G. BAASS AND BRUNO ALLARD

Most traffic information systems now in use are inadequate. The data they contain are often inaccessible, incompatible with other databases, and rarely up to date. Furthermore, the data are rarely fully exploited because they are usually in manual form. The proposed system attempts to solve these problems. It is a database management system based on dBASE III PLUS that includes an integrated urban data bank and three types of software for information systems management, information retrieval, and interfaces with application programs. A sector of the city of Quebec was used to test the feasibility of the system. Although limited, the use of the data shows that the proposed system has distinct advantages over existing approaches. The system will become even more attractive as cheaper, more powerful, and more versatile microcomputers enter the market.

In the transportation planning and traffic engineering field, information needs are of a great variety and consequently data at different levels of detail and precision are needed. One can distinguish between four principal groups of information users, which correspond to the levels of hierarchy of the transportation administration. These are the executing personnel at the technical level, the planners at the tactical level, the executives having administrative or managerial authority in the organization at the strategic level, and finally the users from outside the organization who represent an important and noticeable group due to the great public interest in the transportation domain. Figure 1 shows the differences in the information needs of these groups.

The technical personnel, responsible for the daily operation of the system, require detailed information that should be precise and up to date. This basic information is generated and collected mostly by the department itself. The need for data at this level is constantly growing due to the introduction of computerized procedures and expert systems for design and operation. On the other hand, planners need data coming from various sources (e.g., from other city departments) that may be less detailed and less recent. The administrative or managerial authority of the department requires synthesized and more global information, which should be available rapidly in order to allow optimal decision making based on facts. Finally, external users such as citizen groups, consulting engineers, and more frequently lawyers need particular and precise information. These users are often poorly served by the existing information systems and it is here where the public realizes its inadequacy. A good information system should satisfy all these different needs.

Ecole Polytechnique de Montreal, Department of Civil Engineering, P.O. Box 6079, Station A, Montreal H3C 3A7, Canada.

Computerized information systems have evolved rapidly; from application program oriented databases to complex database management systems (DBMS). Figure 2 shows the basic principle of such a system; the literature *(1–3)* can be consulted for further information.

Today many preprogrammed DBMS packages are widely available on large-scale computers. These have many interesting features, supply information in a user-friendly way, and supply information that one could otherwise obtain only by time-consuming and expensive programming. These programs can be used efficiently for the management of urban transportation data.

Some of the more important objectives (which can be attained by certain commercial DBMS) to pursue in the design of an urban information system are the following:

- Independence of data. (Two data are independent if modifications can be carried out on one, without regard to the other.)
- Nonredundancy of data. (This is an important requirement with respect to storage and to the updating process.)
- Interrelation between different urban data by common references.
- Integrity of data.
- Modularity of the system.

From the user's standpoint a good DBMS should provide

- Multiuser capability.
- Simplicity of use and comprehensive documentation (full-screen editing, on-screen display format generation, report generator, easy query facilities, etc.).
- Restructure ability. (Because the user's need may change in time, the system should allow addition to or deletion of data records.)
- Security of data by a password or other control procedure.
- Easy updating of data.

Due to the commercially available DBMS, the complex duties of data management become easy. These programs offer enormous advantages in the field of traffic engineering where great quantities of data have to be stored, treated, and analyzed. Because the traffic engineering department is part of the urban

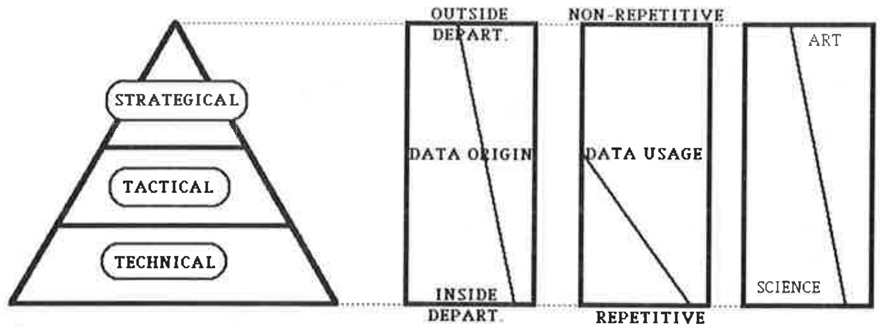


FIGURE 1 The different information needs.

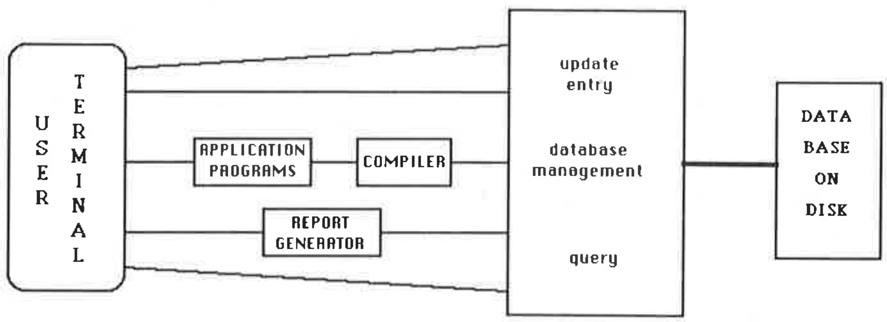


FIGURE 2 A database management system.

services and one wishes to solve urban problems in a more global way, a traffic information system should be integrated into the data of the other urban services to avoid the problems of redundancy and difficulties in data transfer between services. Figure 3 shows how one could design this urban information system (UIS). Common references, such as the links and geocoded nodes of the street network, would permit exchange and integration of data between city departments. All departments are using the same programs for interactive interrogation and exploitation of the database. Control, security, updating, and data entry depend on one authority. In theory, this UIS is ideal. It is fast and efficient, centralized, integrated, and modular and makes maximal use of the data that are so expensive to collect and to keep updated. But there are only a few municipalities undertaking the task of establishing such a system due to the large expenses. The administration of a city touches so many domains that an adequate design for all users seems to be very complex. So complex, indeed, that the creation of two systems,

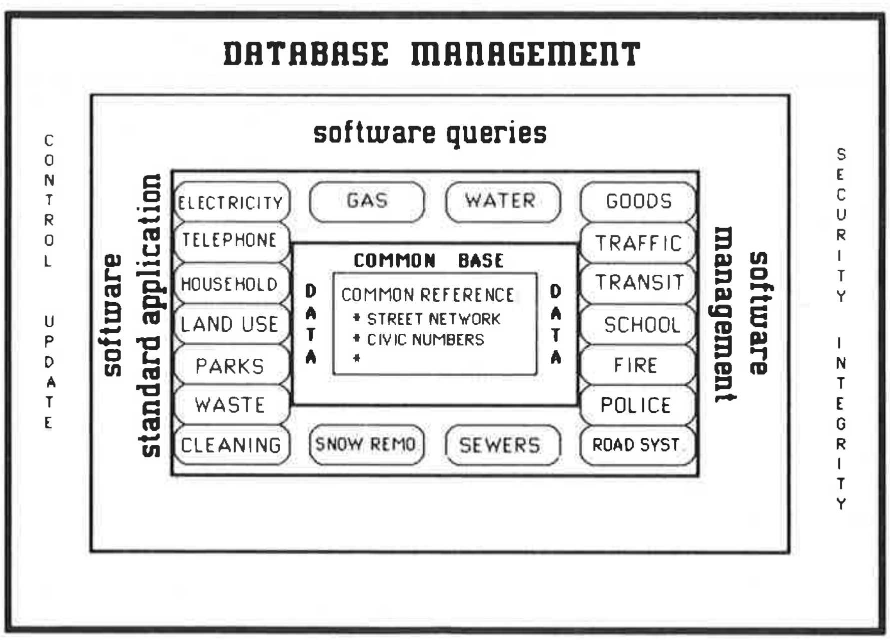


FIGURE 3 Diagram of an integrated urban information system (UIS).

one for planning and one for urban operation purposes, is often proposed.

From a practical standpoint, the centralized UIS is often questioned because the user is dependent on a central system over which the user has no control, especially if the user's queries are not considered a priority by the UIS administrator.

Finally, the mere size of such a centralized data management system seems to contribute to its failure. Recent developments in the field of networking of microcomputers indicate that the centralized approach could soon be replaced by a decentralized one.

For these reasons, the development of a global UIS does not seem to be the way to follow for a municipality with little or no experience in the domain of database management. To avoid the problems related to the implementation of a global information system, a municipality could start temporarily with a DBMS in each and every one of its departments.

A central authority should assure that these data systems remain compatible with one another (e.g., by providing a common geocoded network as a reference). Communication between databases of the different departments would then easily be possible. This procedure would, however, not eliminate redundancy of certain data and would cause some problems in updating elements common to all databases.

But because there are relatively few common data (at least at the operational level where the availability of recent data is a must), this approach would represent important advantages compared to the situation presently encountered in many municipal departments. This partial database would allow the start of computerization (once data are stored on disk they can always be reused or restructured), and would be a transitional stage necessary to fill the gap between the manual information system and a fully integrated UIS.

THE INFORMATION SYSTEM FOR A TRAFFIC ENGINEERING DEPARTMENT

The decentralized approach in the design of a DBMS in traffic engineering was adopted. In an early version, started at the end of the 1970s, the problem was attacked by programming on a centralized mainframe computer. Part of the interactive query and management programs were written. But soon it appeared that an information system would necessitate a significant investment in programming effort or would require the acquisition of expensive DBMS programs. Management often depends on the expertise of programmers and many wish to preserve that dependent relationship. Software houses offering programs for mainframe computers have traditionally not offered packages that permitted computer end users to bypass the programmers. The first database systems were thus complex and difficult to use. Only recently, systems on microcomputers as sophisticated or more so than those on large computers became available.

It was decided that an information system based on a commercial DBMS on a microcomputer should be designed, supposing that microcomputers would develop rapidly with respect to their storage capacities and rapidity of execution. It was also expected that the available DBMS would also become more and more sophisticated and would grow with the needs of the user community.

It was then decided to use dBASE II *(4)*, which was widely available, and even considering the limitations of this early version, it appeared after having conducted a pilot study that the commercial packages were in fact satisfactory for the purpose.

The dBASE system is a relational database management system in which data are stored and related to other data by relations. Under certain conditions, the files can be considered mathematical relations, and the mathematical theory of relational algebra can be applied to problems dealing with the data in these files. One of the basic advantages of a relational database is that one does not have to anticipate all the needs when the files are set up. The dBASE system is now in its third version (dBASE III PLUS), and nearly all of the objectives cited earlier have been attained by the program. A continuity between different versions is guaranteed, so that data can easily be accepted from the older versions.

The management of the data bank (including security, updating, data entry, modification of the data structure, adding new data items, etc.), the query language, and the report preparation are all well developed.

There is a query facility for nonprogrammers to view, enter, and update the data, and a report generator orienting the information to paper instead of to the screen. A special programming language as part of the DBMS can be used to program any report based on the data. The DBMS allows the data to be read and data files to be produced for application programs in BASIC, FORTRAN, Lotus, or other languages. One basic requirement in a practical environment is to allow access to the database by several users at the same time. The network environment is presently in a rapid state of evolution and the newest version of dBASE allows multiuser operation.

There are in fact a number of access and update security codes that not only protect the data values but also relationships between data. Files are protected by data encryption. They cannot be read until they are deciphered or decrypted. A user's profile is created with an encryption code and a password. No user can gain access to the DBMS unless he provides a log-in that can be validated. There are eight different privilege levels for reading, updating, extending, and deleting data where Level 1 is the fullest privilege. These privileges are established by the database administrator and fixed in the user's profile.

The built-in local area networking (LAN) (Figure 4), together with the password protect system, allows some network users only to enter data, whereas other users create and maintain the databases, command files, or applications. Figure 1 shows the scheme whereby different codes are attributed to different levels of users.

There is also a run time package that allows running a compiled version of the dBASE application program without buying the DBMS program. This package makes the application more efficient and faster.

THE DATABASE

Data Integration

Because the goals of data sharing between different city departments and future integration of the data should be pursued, the most basic question to answer is how to establish the basic relation by appropriate linkages.

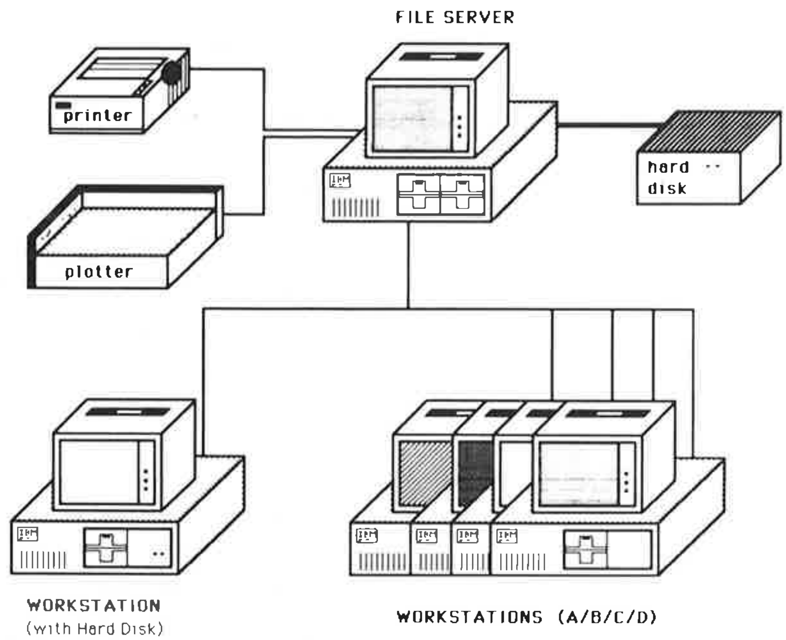


FIGURE 4 The LAN environment.

Urban information pertains to different entities of which Nora *(5)* has identified several:

- Individuals,
- Corporations,
- Public equipment,
- Buildings,
- Building lots, and
- Street segments.

It appears that geographical coordinates are the best, although not exclusive, means of urban data integration because most of the entities have in fact a spatial dimension. Traffic and transportation data are primarily associated with road segments, which can be geocoded by x–y coordinates and by map digitizing after having represented the street network as a network composed of links and nodes. A road segment is thereby represented by two nodes at the street intersections with intermediate nodes, if necessary, for a better graphical representation.

The Dual Independent Map Encoding (DIME) developed by the U.S. Bureau of the Census in 1970 is one such system *(6)*. The proposed reference system is similar to the DIME system (see Figure 5) and uses the Urban Transverse Mercator (UTM) system coordinates, allowing an integration of Canadian census data into the database.

The Contents of the Traffic Engineering Database

There are two kinds of data related to traffic engineering, data related to an intersection (e.g., traffic light parameters) and data related to street segments. Some rules for data collection and data entry are necessary, but most departments have adopted a way similar to the one shown by Figure 5 to enter segment-related data. There is a special application program that calculates for each item entered in the database its x–y coordinates

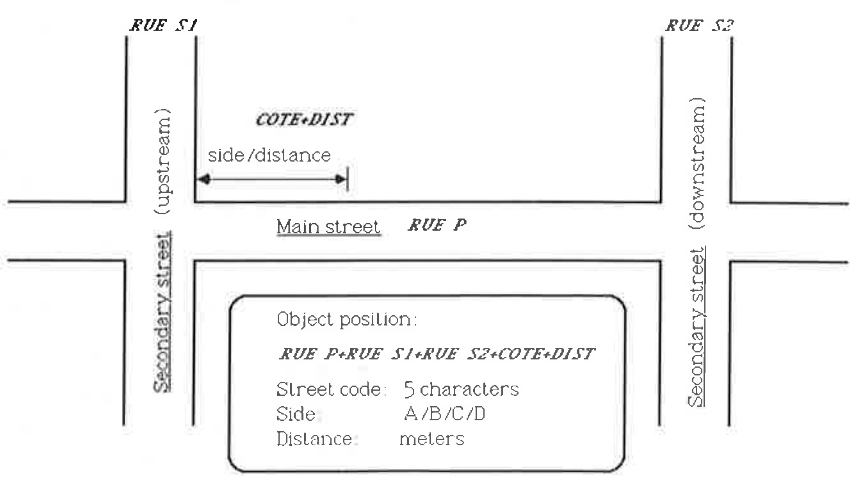


FIGURE 5 Description of the reference system.

based on the coordinates of the nodes describing the segment. This allows an easy and fast graphical scaled representation of the data.

The following data are considered.

Segment-related data

- General data

  Civic numbers
  Private entrances
  Street lamps
  Bus shelters

- Geometry

  Width
  Nodes
  Links
  Islands
  Turning lanes

- Traffic control devices

  Location
  Traffic sign type
  Conditions day and night
  Maintenance dates
  Pavement markings
  Sign posts

- Traffic information

  Volume
  Traffic composition
  Spot speeds
  Overall travel speeds
  Delays
  Capacity
  Class of the street
  Parking

- Accident information (information corresponding to the standard accident report used in Quebec)
- Pavement nature and road surface condition

  Roughness
  Structure
  Bearing capacity
  Cracking
  Maintenance

Intersection-related data

- Geometry

  Turning radius
  Islands
  Sidewalks
  Nodes and links
  Lanes and turning lanes

- Traffic signs

  Location
  Type with code
  Condition of sign
  Maintenance
  Pavement markings
  Sign posts

- Traffic lights

  Cycle
  Splits
  Phases
  Controller
  Traffic light type and condition

- Traffic information

  Volumes
  Traffic composition
  Capacity
  Saturation flow
  Buses
  Parking

This information, when gathered and stored in the DBMS, will contribute to better management, will permit identification of deficient system elements, and will allow prioritizing and scheduling maintenance work. This information will be most helpful for answering queries from the public and will also be useful in legal problems when it has to be shown that the traffic inventory is up to date and stored and treated on a modern and efficient state-of-the-art system.

GENERAL DESCRIPTION OF THE SYSTEM

The initial problem of design was to define an efficient structure of data storage because the way data were stored influenced the efficiency of the DBMS. Yet the structure had to be flexible enough to adapt to the changing demands of the user. This requirement was met because dBASE allowed easy modification of the data structure. Certain weaknesses of the information system appearing during its use could thus be corrected in a convenient way.

After the starting screen giving the logo of the system designed by Allard *(7)* appears, the user is asked to fill out the log-in menu with a code and a password. This menu enables the user to change certain features as, for example, the colors of the menus and the databases to be used (e.g., if there are different city departments).

There are two modes of operation of the system, by menus and by macrocommands. Menus facilitate greatly the use of the program, but the user is limited to the options offered on the menus, underlining the necessity of providing an adequate selection for the user. In fact, the options on the menus were chosen by the potential system users. Standardization and simplicity of the menus were achieved by designing all menus corresponding to the standard layout shown on Figure 6. The menus were designed with the help of an on-screen cursor design program [e.g., QUICKCODE *(8)*].

There are several levels of menus distinguished by different color arrangements defined in the user's profile. The hierarchy of the first two levels is shown in Figure 7.

The main menu in Figure 8 shows the basic options available at the moment that correspond to the basic tasks, data management, traffic studies, and analysis.

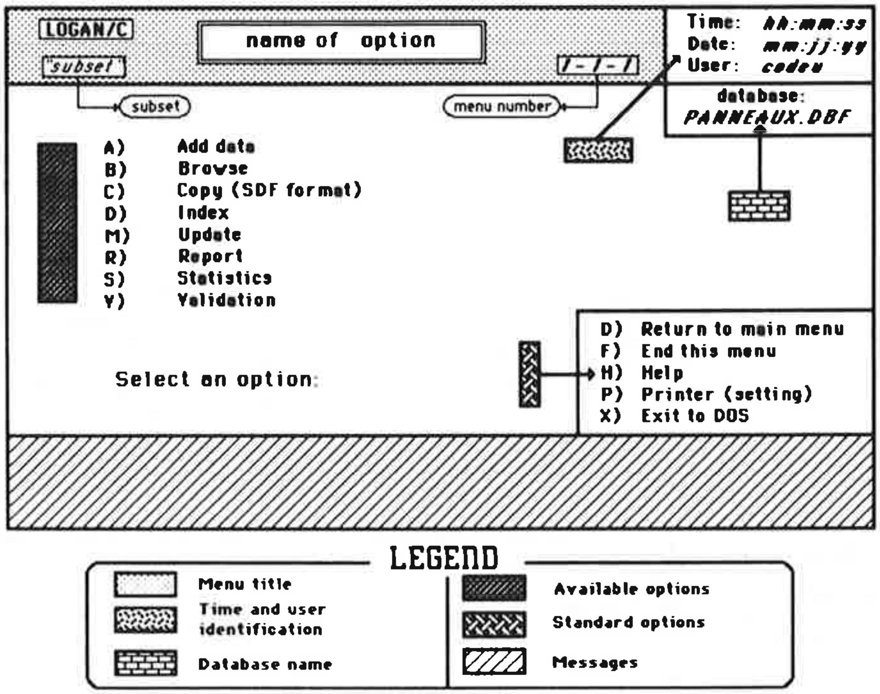


FIGURE 6 The standard for menu design.

Options 1, 5, and 6 allow data entry, validation, and updating. Option 2 can be chosen to establish a new database as a subset of the original one. Option 3 represents the query part, for which direct commands available by dBASE permit the display, on screen or in print, of the contents of the database or part of it. On an intermediate level, between these in-house commands and the application programs, are the preprogrammed repetitive reports generated by the report generator of dBASE. The design of these reports requires some knowledge of the dBASE programming language. The elements described are necessary for the inventory and operation of the traffic service and as such are useful on the technical level, but graphical output is also useful.

These applications cannot be done directly by existing commands of the database management system. These applications are accessible by choosing Option 4 of the main menu. The programs are written in FORTRAN and data are extracted interactively from the database.

The remaining options facilitate work with the system, for example, by giving explanations about the functioning of the system (i.e., the HELP option). LOGAN/C was designed to be used in the French language but the menus in the following application examples were translated for easier understanding.

Examples of Work To Be Done by the Information System

Updating the Information in the Database

This task is done by choosing Option 1 on the main menu. Menu 1–1 (Figure 9) appears on the screen, from which Number 1 is chosen for updating civic numbers.

This choice produces the Menu 1–1–1 shown in Figure 10 and displays nine different options, from adding of data to updating, validating, browsing through the database, and so on.

Option A on this menu produces a lower level of menu, which permits the addition of new information (see Figure 11).

Predefined and Ad Hoc Analysis

Option 3 from the main menu gives access to analysis on the database. Certain analyses are required periodically. A program extracting the information and displaying it in a certain manner on screen and by printer is preprogrammed to facilitate the production of these repetitive reports. Four typical examples are given on the menu in Figure 12.

One of the advantages of the DBMS is the possibility of querying the database and extracting the desired information directly. Thus, a subset of data is obtained without complex programming. Boolean operators of relational algebra can be used to create new associations or to obtain a subset of data. Two examples of such ad hoc analyses may illustrate its potential.

In the first example, a list is needed of all traffic signs on Côte d'Abraham, whose five-letter code is CABRA. The database PANNEAU (or traffic sign) is queried.

The commands are as follows:

```
USE PANNEAU
LIST FOR RUE1='CABRA'
```

Figure 13 shows the variable names and database names used in the examples.

In the second example, traffic signs are obtained that have a day visibility code of 3 or more on Côte d'Abraham and are upstream from the intersection with the street with code name SVALL between 20 and 1000 m from that intersection. Figure 14 shows the menu that achieves this and that also displays the

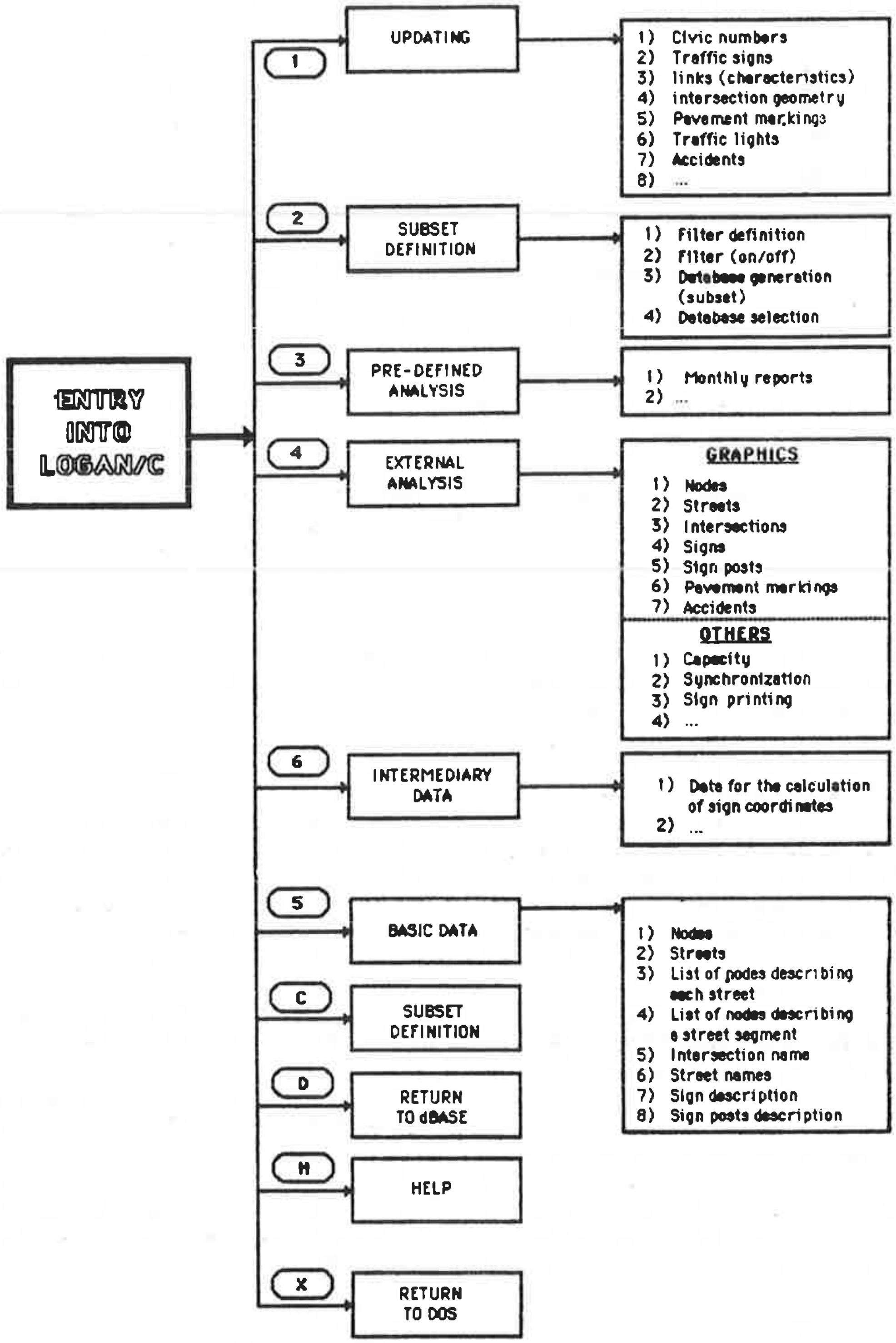


FIGURE 7 The first two menu levels of the information system: (a) first menu level, (b) second menu level.

```
| L O G A N / C |                     | Time:    14:50:08
                                      | Date:    06/20/86
                                      | User:    BRUNO

   1)     Updating data
   2)     Subset definition
   3)     Pre-defined analysis
   4)     External application
   5)     Entrance of basic data
   6)     Intermediairy data

   C)     Configuration/user profile
   D)     End of dBASE
   H)     Help
   X)     Return to DOS
```

FIGURE 8   The main menu of the information system.

```
logan / C     | DATABASE UPDATE |          | Time:    13:59:05
                                           | Date:    06/20/86
                                   1-1     | User:    BRUNO

   1)      Civic numbers
   2)      Traffic Signs and sign posts
   3)      Links (characteristics)
   4)      Intersection geometry
   5)      Pavement marking
   6)      Traffic lights
   7)      Accidents on links
   8)      Traffic counts
                                        | D) Return to main menu
                                        | F) Exit this menu
                                        | H) Help
                                        | X) Return to DOS
```

FIGURE 9   Menu giving access to the different databases.

```
LOGAN / C       | CIVIC  NUMBERS |          | Time:    13:55:12
                ------ update -------       | Date:    06/20/86
                                    1-1-1   | User:    BRUNO

                                            | database:
   A)      Add data                         | CIVIQUE.DBF
   B)      Browse
   C)      Copy (SDF format)
   I)      Index
   M)      Update
   R)      Report
   S)      Statistics
   V)      Validation
   W)      Subset
                                        | D) Return to main menu
                                        | F) Exit this menu
         Select an option:              | H) Help
                                        | P) Printer (setting)
                                        | X) Retunr to DOS
```

FIGURE 10   Menu for work on the civic numbers.

```
................................ : ADD :.................................
| LOGAN / C          ------------------             | Time:    14:59:05 |
|                    | CIVIC   NUMBERS |            | Date:    06/20/86 |
| :____:             ----- update -------   1-1-1.A | User:    BRUNO    |
..........................................................................
|                           Census track ID                              |
|        Codes                                 Full names (output)       |
| main street:                 :____:    |  :                        :   |
| second. street(upstream)     :____:    |  :                        :   |
| second. street(downstr.)     :____:    |  :                        :   |
| street side:                 :_:       |                               |
| distance from intersect.     :_____:   |                               |
| (upstream)                             |                               |
| . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .  |
|                           Address number                               |
|                                                                        |
|     civic number:            :___:                                     |
|     fraction code:           :_:                                       |
|                                                                        |
..........................................................................
```

FIGURE 11 Entrance form for data on civic numbers.

```
..........................................................................
| LOGAN / C      ..........................             | Time:  15:34:05 |
|                |  PRE-DEFINED  ANALYSIS  |            | Date:  06/20/86 |
|                ..........................    1-3      | User:  BRUNO    |
..........................................................................
|                                                                         |
|      1)   List of signs requiring maintenance                           |
|      2)   Number of signs per type                                      |
|      3)   Number of fatal accident per street                           |
|      4)   Number of accident per type                                   |
|                                                                         |
|                                          ...............................|
|                                          | D) Return to main menu       |
|                                          | F) Exit this menu            |
|                                          | H) Help                      |
|                                          | P) Printer (setting)         |
|                                          | X) Retunr to DOS             |
..........................................................................
```

FIGURE 12 Predefined analysis.

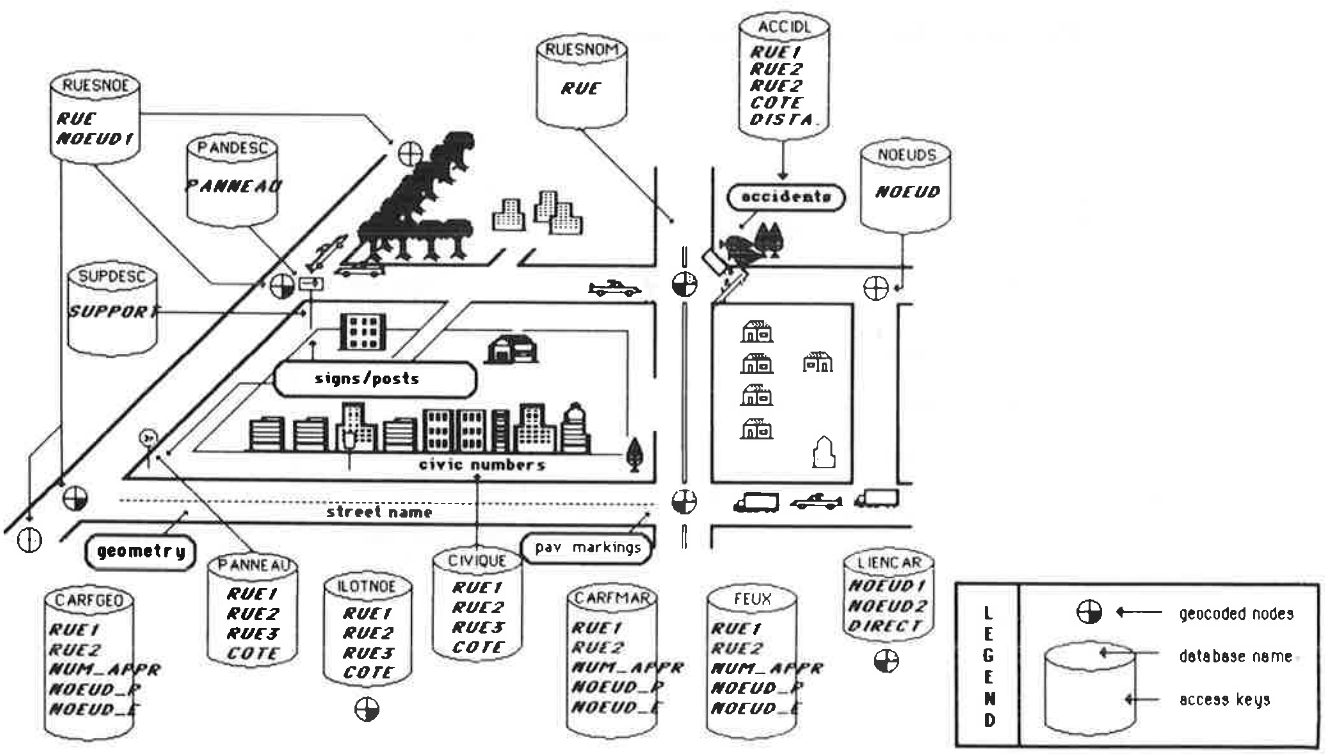


FIGURE 13 Names of variables and databases.

```
LOGAN / C     -------------------------------        | Time:     11:24:35
              |  DEFINITION / UTILISATION   |        | Date:     06/21/86
              ------   OF  SUBSET    ------    1-2   | User:     BRUNO
-----------------------------------------------------------------------

     1)       Definition of filters on database
     2)       Filters (on/off)
     3)       Generation of new databases   [*]
     4)       Choice of database subset

                                            ----------------------------
                                            | D) Return to main menu
                                            | F) Exit this menu
              * Use option 1 to specify     | H) Help
                the conditions              | P) Printer (setting)
                                            | X) Retunr to DOS
```

FIGURE 14 Menu used for the definition of a condition for extraction of data from the database.

```
FIELD NAME    /  DESCRIPTION                 | FIELD NAME  /  DESCRIPTION
       type                                  |        type
RUE1:    (C5) Main street name               | D_FABRICA(D) Fabrication date
RUE2:    (C5) Second. street name (up)       | D_INSTALL(D) Installation date
RUE3:    (C5) Second. ... (downstream)       | D_ENTRETI(D) Maintenance date  .
COTE:    (C1) Street side (A/B/C/D)          | HAUTEUR (N2) Hight  (cm)
DISTANCE(N6) Dist. from intersect.           | DIM_LARG(N2) Width
PANNEAU (N4) Sign number                     | DIM_LONG(N2) Length
CO_FLECHE(C1) Arrow code (from sign)         | C_REMPLAC(N1)Reason for maintenance
SUPPORT (N4) Sign post number                | FABRICANT(N1)Supplier
SURFACE (N1) Sign post surface               |
ETAT_JOUR(N1) Sign visibility: day           |
ETAT_NUIT(N1) Sign visibility: night         |
VISIBILIT(N1) Sign reflectorization          |
--------------------------------------------------------------------------------
Example: RUE1='CABRA' .OR. RUE1='SREAL' .AND. RUE2='SVALL' .AND. DISTANCE > 20
.AND. DISTANCE < 1000 .AND. D_ENTRETI < CTOD("10/20/85").AND. ETAT_JOUR > 3
            ENTER YOUR CHOICE
```

FIGURE 15 Menu used to define a subset on the database.

```
---------------------------------- :  ADD  :-----------------------------------
LOGAN / C       ----------------------                | Time:     14:59:05
                | D E F I N I T I O N |               | Date:     06/20/86
                ------  subset  -------     1-2-1.    | User:     BRUNO
------------------------------------------------------|-------------------------
                                                      | database name
Subset title:                 :__________:            |    FILTRE.DBF
Subset name                   :________:              -------------------------
Description of subset         :_:
(optional)
--------------------------------------------------------------------------------
    Database name          :ACCIDENT:

    Conditions

:____________________________________________________________:
:____________________________________________________________:
```

FIGURE 16 Menu for the definition of a subset of the accident database.

```
-----------------------------------------------------------------------------
| LOGAN / C      ----------------------------          | Time:    10:21:15 |
|                | EXTERNAL  ANALYSIS        |          | Date:    06/20/86 |
|                ----------------------------    1-4   | User:    BRUNO    |
-----------------------------------------------------------------------------
|                                       |                                   |
|        G)   GRAPHICS                  |     A)   Others                   |
|    1)    Nodes                        |  1)  Capacity analysis            |
|    2)    Streets                      |  2)  Traffic light coordination   |
|    3)    Intersections                |  3)  Printed list of traffic signs|
|    4)    Traffic signs                |  4)  Printed list of sign posts   |
|    5)    Sign posts                   |                                   |
|    5)    Pavement markings            |                                   |
|    6)    Accidents                    |      ------------------------------
|                                       |      | D) Return to main menu      |
|                                       |      | F) Exit this menu           |
|                                       |      | H) Help                     |
|                                       |      | P) Printer (setting)        |
|                                       |      | X) Retunr to DOS            |
-----------------------------------------------------------------------------
```

FIGURE 17 Menu to give access to the application programs.

names of the variables and their description, thus rendering research in the dictionary unnecessary.

Another interesting feature is presented by Option 2 of the main menu, which permits generation of subsets of data by defining filters on the databases. Figure 15 is the corresponding menu, and Figure 16 is displayed after having chosen Option 1 in Menu 1–2.

In many cases in traffic engineering, it becomes necessary to use the data as inputs to application programs. Option 4 of the main menu gives access to a number of application programs that will be increased as the system grows. Figure 17 illustrates this menu.

There are several graphical programs to display the information contained in the database. The first program gives a map of the geocoded network (Figure 18).

The second program draws the intersections in two ways, one with traffic markings and dimensions and the other with the existing traffic control signs. One of these options is shown in Figure 19.

Option A3 permits use of a program that demonstrates the coded traffic sign information in a stylized way on a line printer. Signs are not drawn but are represented by their code; their positions, nevertheless, are to scale. Figure 20 shows an example.

The explanations of the sign codes are printed beneath the drawing. Details of the program, which works in an interactive mode and can easily and rapidly access the data, are given by Gourvil *(9)*.

In traffic engineering, many programs exist for different tasks. For example, for traffic light coordination there are TRANSYT and PASSER and for volume data treatment there is COUNTS. These programs are generally prepared by companies outside the traffic engineering department; data entry for them is extremely rigid and time-consuming, diminishing their usefulness. However, the integrated database contains all the needed information for these programs except certain parameters that have to be entered interactively. There is a special interface program that automatically extracts the necessary data and asks the user to add other needed parameters. This process makes the data independent of the special application programs. The methodology of the data extraction is similar for different programs. Work is underway to design an efficient program that prepares data for application programs. Presently there are two interfaces, one for coordination of traffic lights on an artery (algorithm of Little, *10*) and one for the treatment of traffic volume data. The principle of use of a program interface is briefly described by an example of the traffic light coordination. Necessary data are parameters like progression speeds, cycle lengths, volumes, distances between intersections, and time splits at traffic lights. All data except the speed and cycle length, if the existing values are not used, are contained in the database. The user has to specify the name of the artery to be synchronized in a five-letter code and the beginning and ending intersection of the coordination. The interface program asks for cycle lengths and for speeds. Outputs can be directed to screen or to the printer.

These menus and underlying programs contribute to an easy and pleasant use of the information system, even if the present system is in the developmental phase. The following objectives are attained:

- Better use (analysis and extraction) of the collected data.
- Reduction of the traffic data collection work.
- Standardization of present working procedures.
- More frequent use of the application programs and easier program development.
- In the future, use of the information system for planning purposes and for decision making.

The successful implementation of such a computer-based information system depends on the participation and continuous consultation of the potential users during its development. The system has to adapt to the demands of the user (and not vice versa) and has to grow with the needs. One should not believe that a complete system can be developed in a single step. Given the modularity of the chosen DBMS that allows easy adding and changing of the data and data storage structure, the information system can, in a first step, be conceived globally, limiting data introduction to the presently available data and allowing a progressive enlargement of the database with future integration into other urban databases.

Stepwise introduction of such a system also alleviates certain psychological difficulties to the user created by the introduction of a global system that needs a large amount of data without producing an immediate beneficial and practical result.

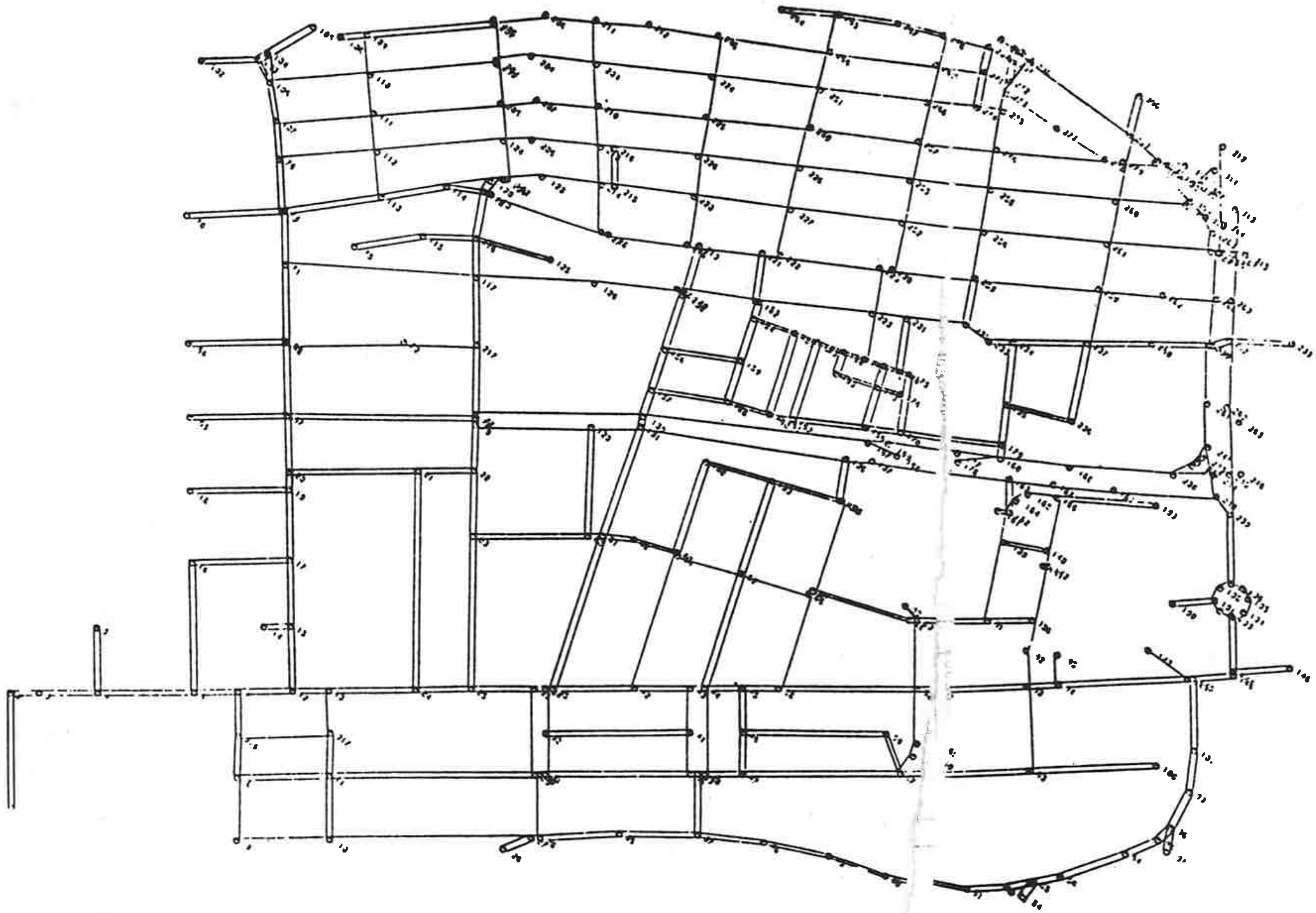


FIGURE 18 Drawing of the network used in the pilot study.

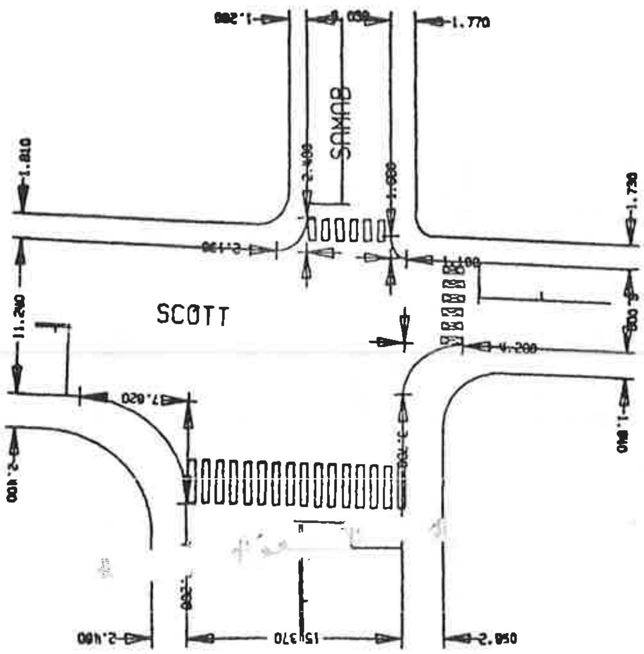


FIGURE 19 Intersection with pavement markings.

THE APPLICATION

The system was initially designed for small or medium-sized cites. It was applied to a sector of the city of Quebec with a network of 322 nodes and 290 geocoded links. The area of the sector was approximately 1.2 km^2; there were 2,040 traffic signs.

Data were collected from a traffic sector in Montreal for another application intended to prove the feasibility of the system in a larger city.

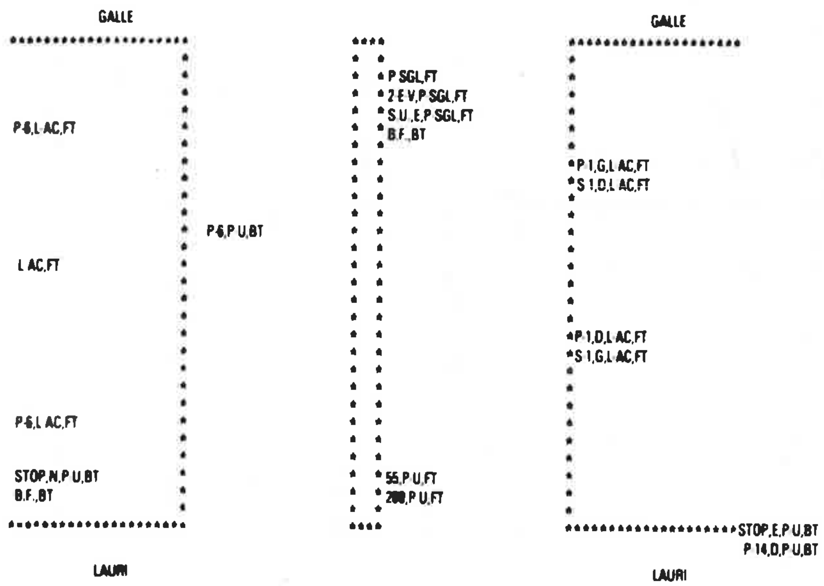
The information system resides on an IBM AT computer with color graphics and a 20-MB hard disk. Response time, even for the longest search through the file of traffic signs, is good.

CONCLUSION

The DBMS programs are in rapid and constant evolution and provide more and more useful commands and user-friendly information treatment. As microcomputers become more powerful and storage capacities increase constantly, the proposed approach becomes feasible also for medium- and larger-sized municipalities. The advantages of the proposed information system based on the well-known and widely distributed DBMS are numerous.

Data are not redundant and are easily kept up to date, and the access is controlled by password security. Data or combinations of data can be treated directly by means of available commands or can be used for application programs. Due to a common link, the information is accessible to all city departments, improving planning, operations, and decision making.

A global, centralized urban information system would be



PANNEAUX

ABBREV.	CODE	DESCRIPTION
STOP	1	ARRÊT/STOP 600X600
S.U.	50	SENS UNIQUE 900X300
	55	TERRE-PLEIN; CONTOURNER PAR LA DROITE 600X750
	290	OBSTACLE; ÉVITER PAR LA DROITE 300X900
P-1	901	STATIONNEMENT INTERDIT EN TOUT TEMPS
P-6	906	STATIONNEMENT INTERDIT DANS CETTE RUE
P-14	914	POSTE EN COMMUN - TAXIS
S-1	941	STATIONNEMENT PERMIS 10 MINUTES EN TOUT TEMPS

SUPPORTS

ABBREV.	CODE	DESCRIPTION
P-U,BT	1	POTEAU EN «U», BORD DU TROTTOIR
P-U,FT	11	POTEAU EN «U», FOND DU TROTTOIR
L-AC,FT	34	LAMPADAIRE D'ACIER, FOND DU TROTTOIR
P-SGL,FT	41	POTEAU DE SIGNAL LUMINEUX, FOND DU TROTTOIR
B.F.,BT	46	POTEAU D'INCENDIE, BORD DU TROTTOIR

FIGURE 20 Representation of the position of traffic signs on a line printer.

theoretically perfect. However, in practice it is difficult, time-consuming, and costly to develop from scratch. With the objective of integration in mind, stepwise development is preferable.

It is recommended that each municipal department develop a database at its rhythm, corresponding to its needs, and following a development plan that ensures a perfect compatibility of data coming from different departments. This recommendation would allow common use of the data, possible because of recent development of the local-area networks.

The pilot project has demonstrated the feasibility of the approach. Supplementary development in the field of interfaces between application programs and the information system will be carried out in order to improve the usefulness of the system and make it an efficient tool for economic decision making and providing aid for planning, development, and day-to-day operation of the traffic engineering system.

REFERENCES

1. C. J. Date. *An Introduction to Database Systems*. 3rd ed. Addison-Wesley, Reading, Mass., 1982, Vol. 1.
2. D. Kroenke. *Database Processing: Fundamentals, Design, Implementation*, 2nd ed. Science Research Associates Inc., Chicago, 1983.
3. D. Kruglinski. *Data Base Management Systems. A Guide to Microcomputer Software*. Osborne/McGraw-Hill, Berkeley, Calif., 1983.
4. *dBASE III PLUS The Data Management Standard*. Ashton-Tate, Torrance, Calif., 1985.
5. H. Nora. Méthodologie pour la planification des systèmes intègres d'information municipale; application au cas des villes nouvelles. *Proc., 4th European Symposium on Urban Data Management*. Planning and Transport Research Computation, Inc., Madrid, 1974.
6. J. Silver. *Special Report 187: Geographic Data Resources for Small and Medium-Sized Urban Transportation Planning Programs*. TRB, National Research Council, Washington, D.C., 1980, pp. 65–69.
7. B. Allard. *Système d'information municipal en circulation sur microordinateur*. M.A.Sc. thesis. Ecole Polytechnique de Montreal, Canada, 1985.
8. *QUICKCODE III. The dBASE III Program Generator*. Fox & Geller, Elmwood Park, N.J., 1984.
9. L. Gourvil. *Conception d'une banque de données de circulation comme sous-ensemble d'un système d'information urbain*. M.A.Sc. thesis. Ecole Polytechnique de Montreal, Canada, 1982.
10. J. D. C. Little et al. Synchronization of Traffic Signals for Maximal Bandwidth. In *Highway Research Record 118*. HRB, National Research Council, Washington, D.C., 1966, pp. 21–47.