INTEGRATED INFORMATION SYSTEM SUPPORTING DISTRIBUTED TASKS IN THE LIFECYCLE OF A ROAD

Rebolj D.[1] and Tibaut A.[2]

ABSTRACT: Road design, evaluation (analysis) and construction are closely connected processes in which design and analysis are iterated to converge on the optimal solution. Many errors and delays frequently appear when data is exchanged between particular tasks; inter-task connections are therefore of great importance for the quality of the final product.

The article describes an integrated information system which is intended to support all important stages of the road lifecycle: design, evaluation (through different analysis procedures), and construction. The concept of the system is object oriented. However, existing program packages are included to solve some common tasks. The system includes conventional relational database as well as spatial database with all relevant GIS functionality. Since particular tasks are often executed at different places and in different companies, the interconnections are supported by a special metafile which contains all specific data about the project as it progresses towards the construction phase.

KEYWORDS: task distribution support, integrated information system, road lifecycle, road metafile, spatial database, GIS

1. INTRODUCTION

Road construction is a responsible task as roads exert a great and long-term impact on the environment and different human activities, and as they present extreme financial pressure. Therefore, careful design and consideration of different solutions from all important aspects is of utmost importance.

Preliminary activities for road design, road design itself, variants analyses, road construction and maintenance can not be imagined without adequate information support. Unfortunately, there exist bad or no interconnections at all between individual program modules which support only some phases of a road lifecycle. Finding interconnections generally depends on the inventiveness of engineers and on traceability to more or less suitable standards for data transfer. Yet, ad hoc data interfaces are mostly only temporary solutions which can hinder systematic information development. Besides unconnected program modules for determining a road lifecycle, the incompatibility of programs to execute related functions (E.G. programs for road geometry design) represents another important hindrance.

The problem of software integration is presently being intensively solved in civil engineering, especially by object-oriented techniques, for instance on the basis of an object-oriented database system - see (Bergmann et.al., 1995), or by applying uniform models for construction - as in (Bakkeren, 1994). Solutions, however, generally refer to high structures and demand high concurrency of processes - as in (Kiliccote et.al., 1995) and (Khedro and Genesereth, 1994). Some

[1] Civil Engineering Informatics Laboratory, Faculty of Civil Engineering, University of Maribor
Smetanova 17, SI-2000 Maribor, Slovenija
rebolj@uni-mb.si
[2] andrej.tibaut@uni-mb.si

of more universal solutions could, however, be applied in road construction - an interesting approach is shown in (Stempfel, 1995) - yet, such solutions can not be traced in literature. The most frequent way of integrating a road in broader information environment is obtained by making use of the geographic information systems; they are, however, not suitable for all phases of a road project.

1.1 Lifecycle of a road

Speaking about integrated information systems, we often exceed the limits of single-user tasks or tasks falling into a narrower, strictly defined professional field. As a consequence, fields and tasks supported by such a system should be determined in detail. It will be of help here to establish a detailed lifecycle of a road, which consists, from the engineer's point of view, of the bellow stated phases:

- design which includes the analysis of traffic and, if necessary, the conception of the new road, determination of possible corridors, collection of necessary data on corridors, selection of corridors with respect to collected data and definition of criteria, conceptual project of the road in numerous variants, variance analysis based on different, preliminarily determined aspects and selection of the most suitable variant, and construction project of the road,
- preparatory activities such as obtaining the sites, selecting the suitable technology, determining the activities and the schedule,
- construction during which the activities are controlled, recorded and adjusted to the plan (or the plan is adjusted respectively),
- maintenance which means controlling the state and performing and recording maintenance activities.

1.2 Distribution of tasks

For a lifecycle of a civil structure, especially road, it is characteristic that individual tasks are performed in different environments, at different places. This fact greatly adds to the known difficulties occurring when individual phases are interconnected into a uniform information flow:

- partly automated processes only (isles of automation),
- data transfer between contractors in the "paper" form, which induces errors and standstills,
- digital data transfer on a low abstract level (text, graphical primitives), which induces errors in interpretation and documentation,
- variability of communication modes (post, telefax, telephone, computer network), which greatly hinders the application of modern methods for integrating individual program modules in a uniform information system,
- simultaneous execution of activities, which hinders data integrity.

Due to the mentioned reasons, the extremely distributed execution of activities should be carefully considered if we are to create an information system for the control of several road phases.

2. ROAD DESIGN AND EVALUATION ENVIRONMENT - RODEE

The above arguments were a challenge for elaborating an integrated road design and evaluation environment. Its general notion is presented in (Rebolj, 1993). The then prototype of the program environment (RoDeE 1.0) was far from being so complete as to suit the real user environment. Also the next version RoDEE 1.1, which was run by personal computers, was not applicable in practice due to different limitations. Anyhow, the experiences obtained in designing and implementing previous versions were invaluable in the elaboration of the present one. This version is an important support for all contractors, starting from those who define the details of individual phases and have the facilitated interaction with the complete system, to those who control and manage the whole project and need an efficient flow and insights into information.

2.1 Conceptual model

The RoDEE system supports the following phases in the road lifecycle:

- selection of the adequate road corridor with respect to available geographic data,
- designing road geometry within the defined corridor,
- analysis of the road design, evaluation and comparison of variants,
- preparation for construction.

The realisation of the above phases is generally discontinuous with respect to time and place, therefore the target system must be very robust, open and flexible. RoDEE is as an open, object oriented system which includes multiple program modules of different generations and a modern attributive and spatial database with all necessary basic functions for database management and spatial data management (basic GIS functions), as well as additional functions for executing more complex specific processes such as geographic determination of the basic road corridor, different road analyses, visualisation, etc.

The target environment is a network of computers in which basic functions are run by special servers (SQL and S^2QL or Spatial SQL, possibly also named a GIS server), while the user-computer interaction is run with graphical user interfaces on multi-purpose personal computers in the multi-tasking operating system environment (FIG. 1).

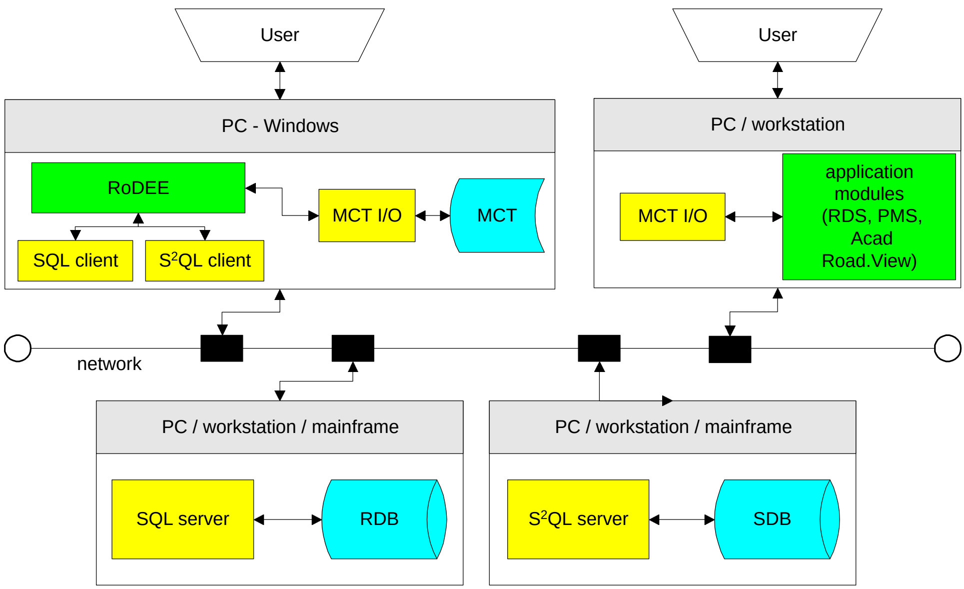

FIG. 1: Concept of the system architecture and process distribution.

2.2 Data structures

The system is composed of three main objects: *project, corridor* and *road body* - details are shown in (Rebolj, 1993) and (Rebolj, 1995a). These objects are stored partly in the attributive and partly in the spatial database. In the attributive database, the structure of data is adapted to the relational model, while in the spatial database all graphical data about the corridor and road are stored as polylines or polygons, respectively. Each object or its individually presentable part is stored as a spatial unit (*feature class* or *cover*). For the present, the following main units were defined: PSEUDO_AXIS (polyline which determines the approximate road track), CORRIDOR (polygon which determines the design boundary), ROAD_AXIS (polyline which represents the axis with crossection points), and ROAD_BODY (polygon which determines the road boundary).

2.3 Task distribution support

In developing RoDEE, from the very beginning great attention was paid to interconnecting different applicative program modules. To this end, two methods were used which serve two different ways of interconnections:

- *Object shell method* is intended for tightly interconnecting the modules in object oriented environment. It is more suitable for the centralised principle or the client / server principle. The original idea of the object shell method is described in detail in (Rebolj, 1993), while reports on later up-dated versions have also been published several times; E.G. (Rebolj, 1995b).
- *Data interface method* is a classical method intended for a loose interconnection of modules. It is more suitable for distributed processing - on locations which cover particular phase of a process.

Due to distributed tasks in the lifecycle of a road, we have concentrated lately more on the data interface method. To this end, we defined a special *road metafile (MCT)* which completely corresponds to the road database structure in the RoDEE environment. Individual main groups are completed independent wholes so that also partial data, needed or gathered in different project evolution phases can be transferred. Main data groups refer to basic data about a project, description of a corridor, axis, crossections and crossection element types (see FIG. 2). By its role, the metafile corresponds best to AEC product model descriptions but its present syntax is not compatible with any of the known standards; as for instance STEP (ISO CD 10303, 1993). The reason for this lies mainly in the complexity of standards, which hinders the introduction of MCT or RoDEE application in real environment rather than supports it. In addition, the suitable standards are still being established (ISO TC184/SC4/WG3 N434, 1995).

FIG. 3 shows the RoDEE environment and its integrated supporting application modules which are equipped with interfaces for reading and writing the MCT metafile (so called VCTs). Applications which need some more sophisticated additional information are not directly connected to MCT (as for instance Project Management System due to its completely different context).

Supporting applications are being called only for the execution of particular methods and are therefore not obligatory for the operation of the rest of the system. If a supporting application called by a RoDEE method is missing, the user is simply notified about the fact. On the other hand, supporting applications may be used independent from RoDEE. The integration of results is still possible with the help of VCT afterwards. In this case, however, some missing links in the RoDEE data structure might be required.

```
[PROJECT]
'comments
```

```
        [NAME]
                ProjectName(text)
        [DESCRIPTION]
                ProjectDescription(text)
        [SCHEDULE]
                Start_date(date)  Actual_date(date)
        [WINDOW]
                Y_min  Y_max  X_min  X_max
[CORRIDOR]
        [PSEUDOAXIS]
*               X  Y
        [BUFFER]
*               X  Y
[AXIS]
        [HORIZONTAL]
*               X_start  Y_start  X_end  Y_end Radius  A  Length
        [VERTICAL]
                [TANGENTS]
*                       L_start  Z_start  L_end  Z_end
                [ROUNDINGS]
*                       L_start  Z_start  L_end  Z_end  Radius  Length
[CROSSECTIONS]
*       [POSITION]
                Number  L  X  Y  Z
*       [TERRAIN]
*               X'  Y'
*       [ELEMENTS]
                [POINTS]
*                       Id  X'  Y'
                [CONNECTIONS]
*                       Id_startpoint  Id_endpoint  Element_Type(text)
[CROSSECTION ELEMENT TYPES]
*       Element_Type(text)  Class(text)  Description(text)
(eof)
```

FIG. 2: Structure of the Road metafile (MCT); stars indicate the possibility of repetitions

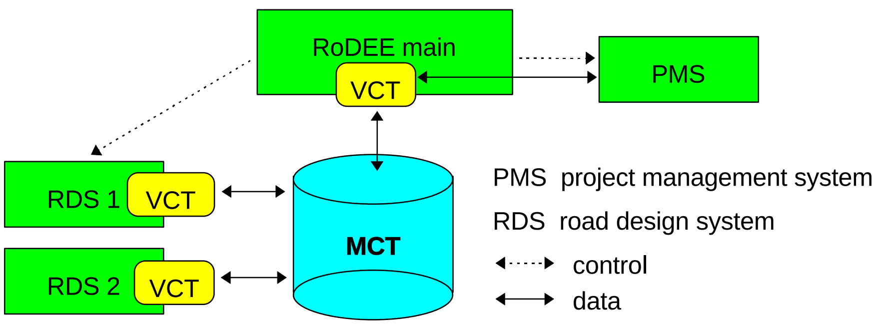

FIG. 3: Application concept of a road metafile (MCT) in the RoDE environment

Since particular tasks which are being executed on different sites don't need the potential of the whole RoDEE, the main (coherent) part of RoDEE is modularly built as well. It consists of a compact and small kernel and optional "plug-ins" in form of OLE servers (see FIG. 4). OLE servers are responsible for the execution of optional methods (E.G. quick visualisation, computation of hazardous substances emissions, etc.). In this way it is possible to configure a user-tailored RoDE environment which suits each particular user-site.

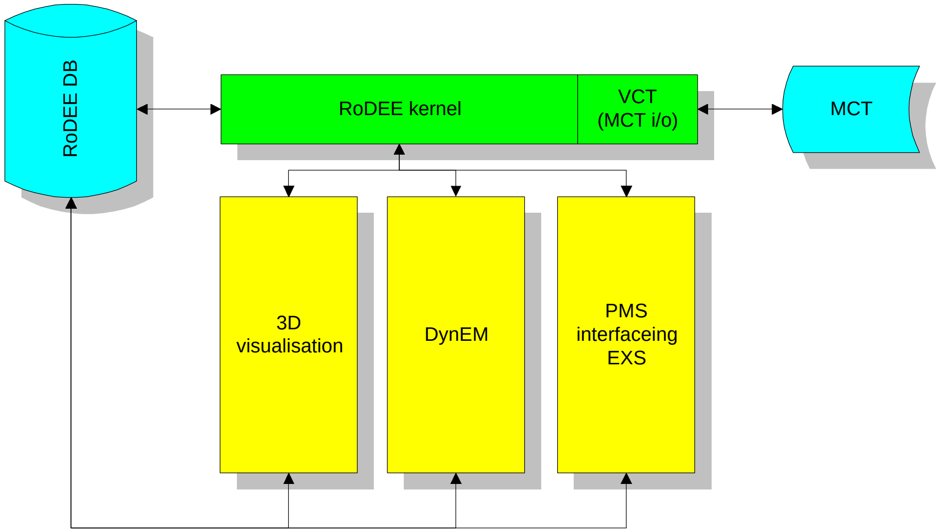

FIG. 4: The architecture of the main part of RoDEE with multiple “plug-in” modules.

2.4 Implementation of the current version (tools, components, architecture)

A high capacitance personal computer with installed Microsoft Windows (95 or NT) is foreseen at the end-user workplace. The connection to the network and the application of file or SQL servers are desirable, as well as the availability of a wide area network for communicating with contractors in the course of the project evolution.

For the implementation of the basic module of RoDEE 2, we used Visual Basic 3.0 Pro which is an efficient program environment, especially for creating integrated, distinctly interactive applications in Windows. The attributive database was created with the help of MS Access which is, however, not necessary for the running of the system.

For the management of the spatial database and for the execution of geographic functions we used the Prime Meridian Power GIS made by Essential Planning Systems from Canada - it is described in detail in (Essential Planning Systems, 1994). The communication runs in the special language S^2QL (Spatial Structured Query Language), which is based on the standardised and well established SQL and is expanded with functions for spatial data processing. It contains a systematic set of all important GIS functions, together with the network and 3D analysis. According to the statements and readiness of greater producers of geographic information systems, see for example (Dangermond and Lauzon, 1995), it is hoped that an international standard for a similar language is going to be established in the near future.

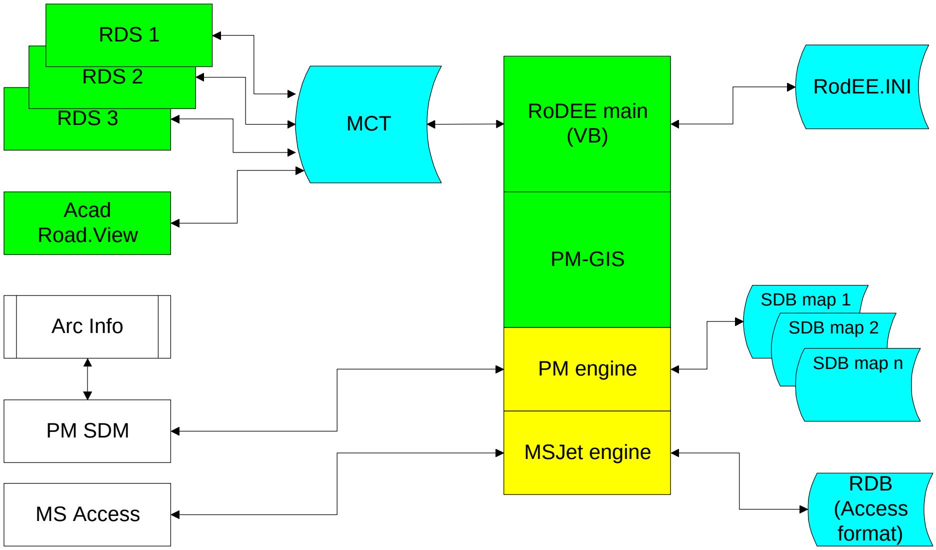

FIG. 5: RoDEE 2 architecture.

The existing application program packages are needed especially for the road geometry design. For efficient functioning of the RoDEE system, the programs must be equipped with the MCT interface which is presently applicable in the PCDOR program, while it is being implemented for the programs Plateia of the firm CGS and Caddy/Road Design of the firm Ziegler Informatics.

A quick 3D visualisation of the road was implemented in C++ with the aid of AutoCAD ADS; for its functioning AutoCAD must be installed - for details see (Tibaut and Rebolj, 1994).

Additional useful software is MS Access, which allows direct access to the attributive database, and EPS Prime Meridian Spatial Data Manager for direct access to the spatial database. The architecture of the present version of the Road Design and Evaluation Environment is shown in FIG. 5.

3. RODEE SUPPORT OF THE ROAD LIFECYCLE

As the basic functions of RoDEE are integration, co-ordination and documentation of tasks, individual supporting components are not obligatory in every environment; they are necessary only in cases where they execute a specifically defined task (E.G. module for road design is not necessary in the phase of determining suitable corridors). Therefore, no special additional hardware and/or software is necessary at specific places. What is important is a consistent use of RoDEE in the process of road development.

3.1 Projects

The basic object in RoDEE is the *project* that connects all appertinent structures. All data about projects are stored in linked databases and need not be extra opened, read or written. By using the command *Project.Directory* we obtain the table of basic data that can be edited, deleted or completed with new projects according to our needs. Besides the command *Directory* in the menu

Project, there are two other commands for reading and writing the road metafile: *MCTin* and *MCTout*.

3.2 Corridor definition

After opening a new project, we first define the corridor to limit the design area by using the collected geographic data (E.G. isolines, lots cadastre, landuse, etc.). The corridor can be determined in different ways. We generally make use of the so called pseudo axis, a simple polyline which roughly defines the road layout. In the next step, the corridor is generated automatically (as a buffer in a selected distance from pseudo axis) or manually.
FIG. 6 shows a manual input of individual points of the corridor where pseudo axis is being used for orientation only. Such input is more suitable for sensitive areas because it allows to avoid critical objects or zones. On the top of the GIS window there are utility buttons which allow zooming in and out as well as shifts of the displayed co-ordinate window. They can be activated also while in graphic editing mode.

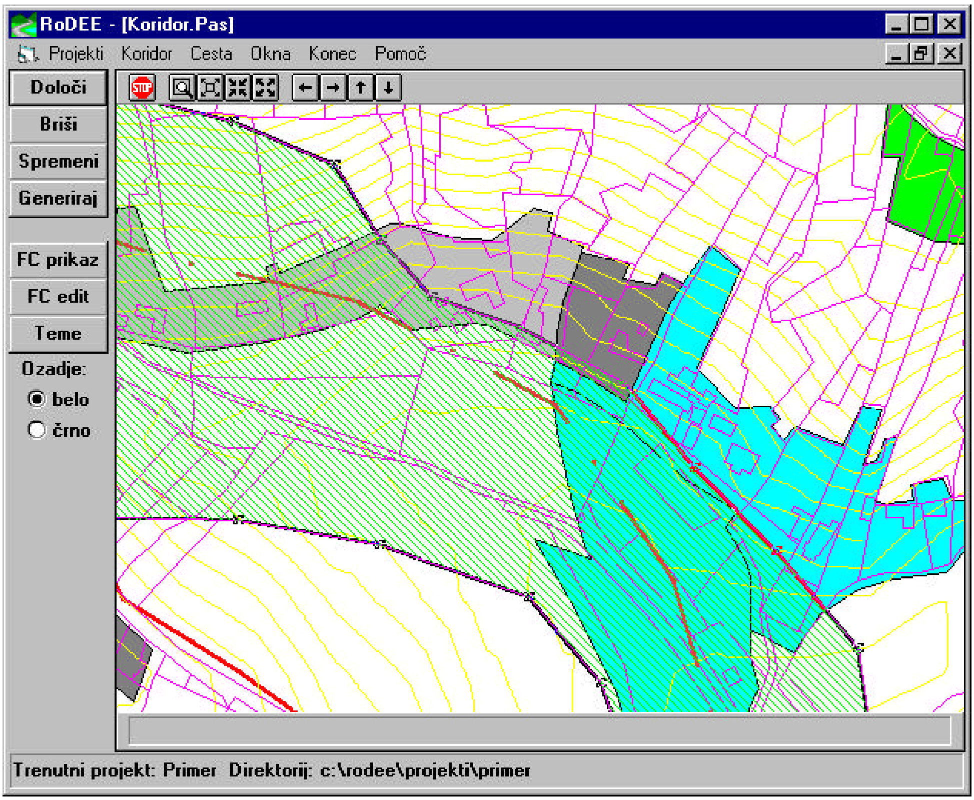

FIG. 6: Defining the corridor

3.3 Road design

The road design is generally carried out at a different location from that of the corridor definition. For data transfer, the metafile of the road is used. Other needed geographical data (especially elevation points or a terrain model) are transferred to the contractor in some other form convened, depending on the available program tools. The geometry design is carried out within a defined corridor according to standard procedures. When the road geometry or its part are defined, a designer writes suitable groups into MCT and transfers it back or to the contractor who is responsible for subsequent processing. It is useful, however, that also a designer can make use of RoDEE or, at least, of a module for quick 3D visualisation which can operate also individually.

3.4 3D visualization

The interactive program tool for the road display RoadView is created as a module which runs in the AutoCAD environment, yet it can be activated also from the RoDEE menu. Making use of input data (road components from MCT) and the graphic features offered by AutoCAD, we can display a three-dimensional graphical model of the road in perspective as shown on FIG. 7.

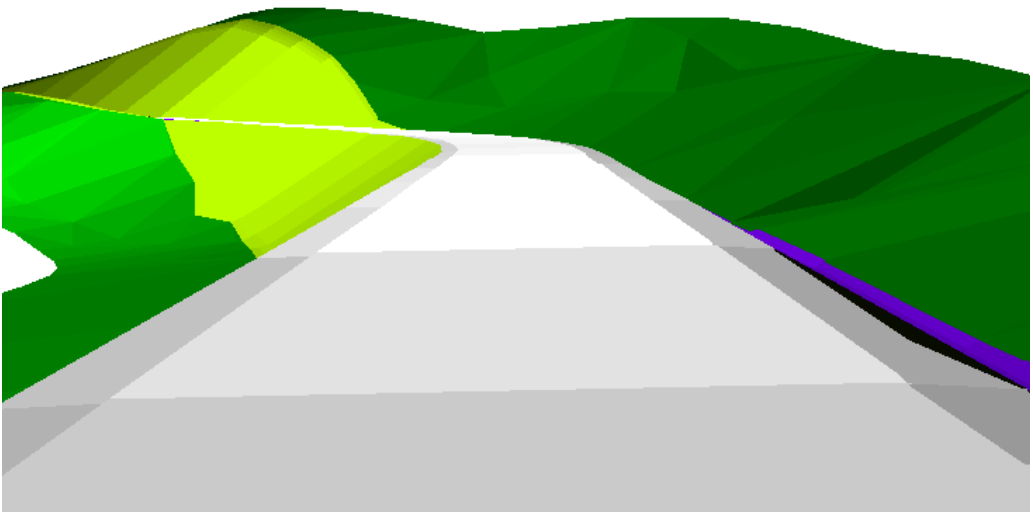

FIG. 7: Road display with the appertinent corridor in perspective

3.5 Emission Analysis

The program module DynEM - Dynamic Emission Model (Rebolj, 1995c), supports the computation as well as geographic representation of some hazardous substances emissions (CO, NOx and soot). DynEM is completely integrated in the RoDEE environment, as it reads and records the relevant data directly in the central database. Some data on traffic must be typed in manually, in later phases of the RoDEE development, they could be read from the suitable road network database.

The display of results is possible immediately in the geographic window (FIG. 8) or in the form of diagrams which can be drawn with the help of MS Access.

3.6 Preparation for construction

The preparation of location documentation is a rather time-consuming process, as it includes a great deal of repetitive processes such as signing compensation contracts for damages, acquisition contracts or expropriation contracts for lots (or their parts) which will be crossed by a road and its appertinent elements.

As the spatial database already contains data about the road boundary, the support of the location documentation preparation was also included into RoDEE. It is based on the overlay of thematic layers ROAD_BODY and CADASTRE FIG. 9).

The spatial database stores data on surfaces covered by the road, which are the basis for the location procedure, while the relational database stores the necessary attributes (lot number, landuse, cadastre class, owner, etc.)

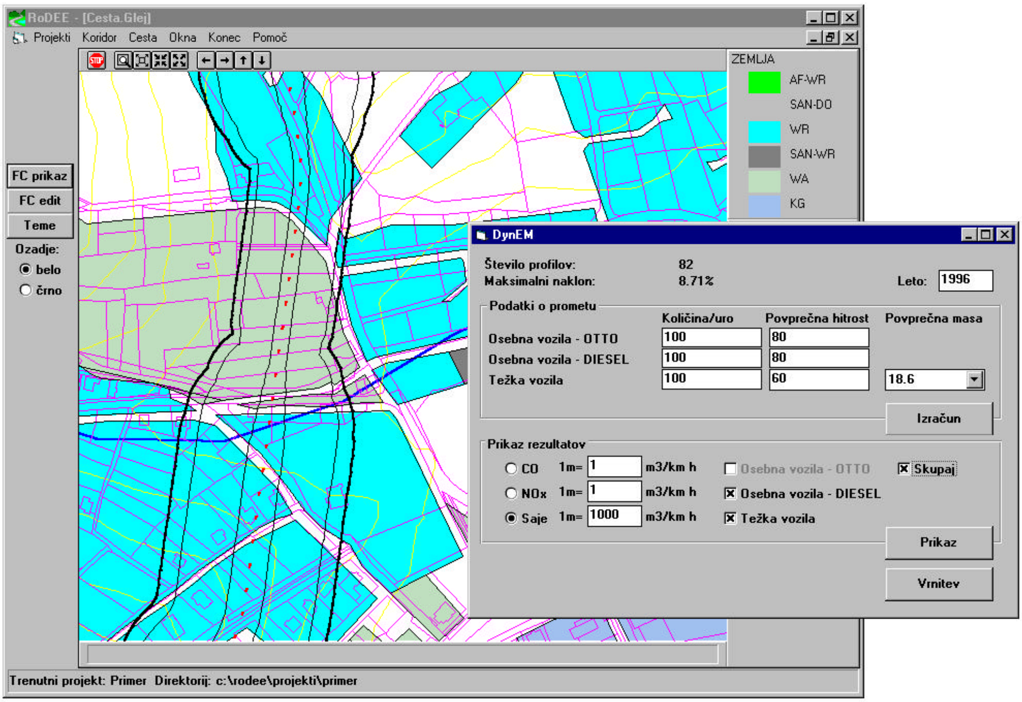

FIG. 8: Soot emissions computed with the DynEM module

When we have created the basic data and obtained the official approval, we can begin with the procedures for concluding contracts. The procedures are performed automatically for all lots / owners that are included in the overlay of the road body on the cadastre. They result in obtaining the documents which are stored either in a digital or material form but are linked to basic records in the intersection via uniform identificators.

FIG. 9: Overlay of a road body and lot cadastre

4. FUTURE DEVELOPMENT

Future development of the Road Design and Evaluation Environment greatly depends on experiences which were or will be gained in using this tool in different phases of the road lifecycle. Nevertheless, we can predict some improvements and perfections even at this development step:

- implementation of specialised SQL in S^{2}QL servers and complete architecture support of client / server (VB 4.0 Enterprise edition, Prime Meridian S^{2}QL server),
- quicker 3D visualisation by making use of graphical OLE components (OCX)) 3d Graphic Tools of the firm Micro System Options,
- adding new analyses which will gradually allow a universal evaluation of a designed road,
- display of effects of hazardous emissions on objects in the environment with a dispersion model and intersection of adequate layers,
- support of variants comparisons with a suitable expert system which could assure the objective evaluation of results obtained by individual analyses,
- support of the construction phase by linking the cost planning system to the schedule planning system (PMS),
- integration/linking of other data on roads and corresponding objects to the RoDEE database,
- MCT upgrade, standardisation and the implementation of MCT interface in different application programs.

5. CONCLUSION

The present experience with the RoDEE environment has pointed to the necessity and advantages of information links between all phases of the road lifecycle. Especially the metafile has proved to be of great importance since it functions as an interconnecting link between individual processes. Much attention will therefore be paid in future to its open structure on one side and to flexible connection mechanisms on the other side. Only in this way we can assure a modular and open environment which will be easy to adopt to users requirements. As roads are closely linked with the environment, future development of the system will greatly depend on overall development and especially on standardisation in the field of geographic information systems.

6. REFERENCES

Bakkeren, W.J.C. (1994). Integrating Structural Design and Analysis through Product Modelling, *Proceedings of CIB W78 Workshop on Computer Integrated Construction* Helsinki, Finland.

Bergmann, A., Bode T., Cremers, A.B. and Reddig, W. (1995). Integrating civil engineering applications with object-oriented database management systems. *Proceedings of Computing in Civil and Building Engineering* Berlin, Germany. 67.

Dangermond, J. and Lauzon, J.P. (1995). ESRI's approach to Interoperability. *ESRI ArcNews.* **17**, 36.

Essential Planning Systems. (1994). EPS Prime Meridian Product Overview. Victoria, Canada.

ISO CD 10303 -41, -42, -43 (1993). STEP, Standard for exchange of product data.

ISO TC184/SC4/WG3 N434 (T12) (1995). Product data representation and exchange.

Khedro, T. and Genesereth, M. R. (1994). The federation architecture for interoperable Agent-based concurrent engineering systems. *International journal on concurrent engineering, research and applications* **2**, 125.

Kiliccote, H., Garrett, J.H. and Choi, Jr. & B. (1995). A distributed architecture for standards processing, *Proceedings of Computing in Civil and Building Engineering*. Berlin, Germany. 343.

Rebolj, D. (1993). Computerunterstützter integrierter Straßenentwurf in einer objekt-orientierten Umgebung. Verlag für die Technische Universität Graz. Graz, Austria.

Rebolj, D. (1995a). Integrated road design and evaluation environment. *Proceedings of Computing in Civil and Building Engineering.* Berlin, Germany. 1001.

Rebolj, D. (1995b). Integration of Procedural Program Packages into Object Oriented Program using Object Shell Method. *Proceedings of Design to Manufacture in Modern Industry.* Bled, Slovenia. 748.

Rebolj, D. (1995c). Simulation of harmful emissions caused by road traffic with the use of dynamic emission model. *Proceedings of Power Engineering.* Maribor, Slovenia.

Stempfel, B. (1995). Spatial information system of structures. *Proceedings of Computing in Civil and Building Engineering.* Berlin, Germany. 979.

Tibaut, A. and Rebolj, D. (1994). Visualisierung des Straßenkörpermodells. *Proceedings of Internationales Kolloquium über Anwendungen der Informatik und der Mathematik in Architektur und Bauwesen.* Weimar, Germany. 151.