GSDI-9 Conference Proceedings, 6-10 November 2006, Santiago, Chile

A Toolkit for Optimizing Cartographic Production from a Global Database

W. Wolfer
Grüner Weg 5
88719 Stetten
Germany
werner.wolfer@hs-karlsruhe.de
wolfer_gis@gmx.de

Abstract

Generating maps on demand from global datasets requires a chain of technically advanced processing tasks, regardless of whether the end product is needed in a planning department, placed on an organization's home page, or sold in a bookstore to a customer. Production can be facilitated by establishing an intelligently designed geoprocessing environment: the aim is to have a systematic, standardized work flow, consisting of effective methods and toolsets in all production phases from database organization all the way through product finishing.

On the example of the production of world atlases, this paper describes a geoprocessing toolkit that has been developed for effectively utilizing global databases at scales between ca. 1 : 500,000 and 1 : 4 Mill. This toolkit, in turn, is part of a production scheme based on work flow standards common to a wider range of products and, though being devised specifically for commercial map publishing, might serve as a repository of successfully tested methods with the capacity for streamlining the data handling infrastructure in an extended spectrum of mapping applications

At the outset, an outline of the overall production framework is given, indicating where tools provided by standard software packages reach their limits. The core part consists of a description of the production steps in which geoprocessing is applied, followed by a discussion of a representative selection of the tools in more detail. Emphasis is on explaining the functionality rather than on software architecture. The procedures encompass all major production activities, such as database and product editing, data verification, map preparation, product generation, and updating (i.e. synchronizing the activities taking place on the database and on the product sides), plus a set of general process handling functions. Throughout the system, the principle is that work performed during processing of a particular product may be, totally or partially, re-utilized for related products, thus keeping the amount of duplication of tasks to a minimum. In conclusion, a summary of the experiences in working with the system will be given, and some of the misconceptions in toolkit configuration will be pointed out.

Introduction

Like any other sector of the publishing industry, map producers need to respond to customers' demands that have been growing constantly over the years. These demands, among factors such as topicality and an appealing graphic presentation, include that the map is available at a price as low as possible. For the publisher, this means that his presence on the market depends on technically efficient production methods.

Two of the preconditions imperative for generating high quality cartographic products on a commercial basis are

- a thoroughly devised production framework tailored to an organization's product range

- a collection of tools covering the required geoprocessing functions, optimized and in tune with each other

Geoprocessing, in this context, is the collective term for all computational operations within a GIS-based production environment. It is required for these operations to be applicable by specialist staff (e.g. by a cartographer) familiar with the process and the principles of mapmaking, but unaware of the technical processes running behind the scenes. In the easiest case, this means just pushing a button, selecting from a menu, or typing a command - however, for the most part, a range of additional parameters are to be specified along with the commands in order to provide for flexibility in the execution of the procedures.

Up-to-the-minute, off-the-shelf GIS software packages offer a large number and variety of such operations, ranging from very general to highly exotic. For the cartographer working with these systems, apart from having to choose from functionality that usually extends far beyond the range he or she actually needs, the main problem is that most of the relevant functions perform actions of a very limited scope such as generating a buffer, positioning text on a map frame, or adding an item to a table in a database - to name but a few typical examples. Consequently, a considerable number of individual actions are required and need to be repeated many times in the proper sequence and with the correct parameters – something that product managers in private sector map publishing are keen to avoid.

In order to speed up processing, most software packages offer the capability for automating sequences of recurring tasks by generating pre-fabricated command sequences, or macros, in a programming language supported by the system (examples of such macro languages are the Arc Macro Language, Smallworld Magic, or, more recently, languages conforming to Component Object Modelling (COM) standards). This way, complex procedures may be generated, and, theoretically, entire production chains may be automated by implementing an all-embracing routine for a product and activating it with a few input parameters.

A more realistic scenario, however, characterized by a broad variety of products derived from a wide choice of available input data, would require a more complex configuration. The range of products derived from global databases, for instance, may include country and road maps, atlas pages, maps published in travel guides, or maps of any other purpose at medium and small scales. The type of projection used may differ and so may the formats, sizes, pages composition, symbolization, and many other properties. The approach of fully automating long processing chains would result in an inacceptably rigid production environment (run by a desperate team of operators constantly busy modifying programs to accommodate for spontaneous and temporary exceptions from the rule). A compromise needs to be found between the two extremes.

Such a compromise would make use of the commands provided by the relevant GIS software and the programming language as the basic constituents and arrange them in blocks, or modules, of appropriate size and function range. These modules would represent the smallest operational units in the production setup and might be organized in categories reflecting, at the most general level, the main phases of map production (Figure 1). From the point of view of the cartographer (who may be seen as an operator controlling an instrument panel in the mapmaking process), the individual modules are the smallest available working units, or tools, and, provided the system is well designed, represent the optimum both in flexibility and in ease of operation.

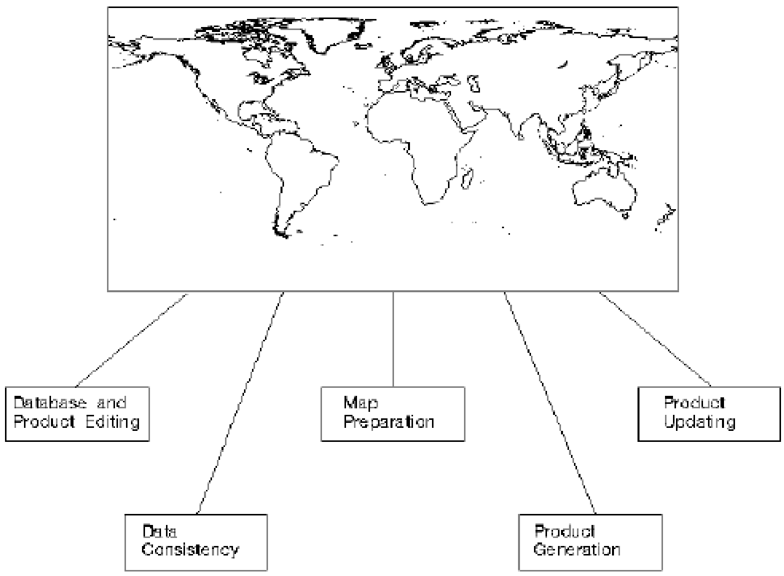

Figure 1: '*Control Panel'*

This paper introduces a toolkit encompassing all major phases of map production that has been developed by the author and that has, for the most part, been implemented and used as a matter of routine in a commercial map publishing environment. The tools, in turn, are part of a production scheme developed for specifically facilitating the production of medium to small scale cartographic products (scales from 1 : 500 000 to global) from a global database and, in making use of predefined interfaces and work flow rules, provide an optimum in processing instruments in the various phases of map production. The tools have been implemented *as a response to processing needs not covered by the base software*.

Within the framework of this paper, it is not possible to discuss the full range of tools that have been generated as part of the system - nor would it be realistic to provide a detailed description of the set of parameters controlling it. Instead, a representative selection will be povided by looking at the various steps in the preparation of a typical product. The main objective is to show the interaction of production framework, tools, and parameter definitions that make up the essentials of the system as a whole.

The Production Framework

In a typical production environment, the *database* is organized in a layer structure, the individual layers representing themes such as administrative boundaries at various levels, river networks, towns and settlements, roads, railways and other infrastructure features, plus a range of details specific to the publisher's area of business (e.g. information on travel and tourism). There is no need for the source data to be organized in one seamless database – in practice, it is more common that data are brought together from various sources, the individual components showing different types of detail, structure, or projection.

Products derived from such a database may include road maps, atlas pages, maps in travel guides, or other types of maps. Being separate commercial entities, the products are prepared in their own *product space* into which the relevant data are transferred. The transfer is activated by specifying the map frames and a few key parameters - this will start a program which, as part of the transfer, also handles the necessary transformations such as change in projection, scaling, and rotating the data to have the map frame horizontally aligned, if desired.

Products are organized - logically as well as technically - in a hierarchical structure as illustrated in Figure 2. Logically, this reflects the usual approach which is from the generic to the specific. Technically, this supports a systematic framework of data flow, with the side effect that work performed during processing of a particular product may be, totally or partially, re-utilized for related products, thus keeping the amount of duplication of tasks to a minimum. This is a direct consequence of the fact that, with the exception of the upper level which is of purely organisational significance, the levels correspond to major production steps in the process of product generation or revision, each resulting in a set of intermediate products or *deliverables*. It should be noted that there is no need to keep a complete set of data in each of the levels - only the details modified in the various preparation steps and the information needed to control the process are permanently stored. The levels are explained in the following in more detail.

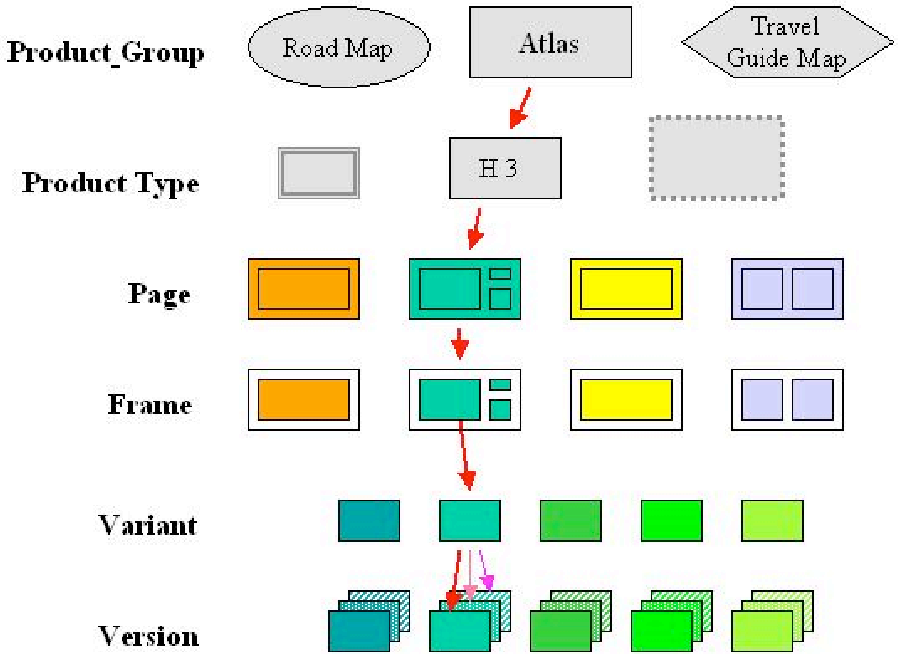

Figure 2. *Organisational chart of map product hierarchy.*

Level 1 - Product Group. These are major product families such as atlases, road maps, or maps specifically created for publication in travel guide series

Level 2 - Product Type. Within a group, products are further differentiated. Product types may differ in style, format, or volume or may be associated with brand names. They are normally referenced by internal product titles. This is the uppermost level in which product specific items are stored (such as the arrangement of pages for a particular atlas, style sets, or tables from which information needed in a product will be derived)

Level 3 - Page. This refers to the physical page and comprises both the map(s) graphics and all details on the margin. A page contains one or more maps (or frames). There is a n-to-n relationship between Levels 2 and 3: an atlas consists of many pages, and a particular page may be used in various atlas types (typically with variations in scale, layout, or symbolization properties)

Level 4 - Map (Frame). The map frame is the uppermost level affected by changes in the database. It is also the level in which tasks related to product refinement are performed (e.g. edge-cleaning). Data transfer from the database to the product usually involves transformations such as change in projection, rotation, scaling, etc. A page may consist of several frames, and a particular frame may be used in pages of several products

Level 5 - Variant. By definition, variants of a frame all cover the same area. Variants may differ in one or more of the following: compaction of features or text, reduction of features or text, inclusion of additional themes, map text in a different language, variation in scale, or a major change in the symbolization of features. There is a 1-to-n relationship between Levels 4 and 5

Level 6 - Version (Edition). This is the level reserved for recording the activities around the updating process, and its elements represent successive versions of the same product. The objective is to maintain the state of preparation from the previous edition.

Main Production Phases

Product generation consists of a chain of tasks of varying degrees of automation. These phases are listed in Table 1:

1. Preparatory work (e.g. map frame positioning)
2. Data collection
3. Database updating
4. Preparation of product specific symbols and text fonts
5. Data transfer from database to product
6. Refinement of product data
7. Preparation of frame details
8. Proofreading
9. Final corrections
10. Production of printfiles

Table 1. *Main Production Phases*

The tools developed for performing the tasks required in the various production phases are synchronized in such a way that data generated as output by a particular tool can be used as input data in a tool further along the production chain. Technically speaking, the whole production process is organized as a series of *function blocks* (eventually corresponding to the range of functions in the individual tools), with the aim of minimizing the number of such blocks and, at the same time, providing for the highest possible flexibility with regard to the range of products that may be generated. In other words, the modules need to be big enough to combine frequently used command sequences in one tool, and small enough to ensure modularity, i.e. evading the need for command sequences to be part of more than one tool.

Procedurally, tools may be grouped into front-end tools activated by the operator and tools called by other tools. A large proportion of the tools is a combination of both, i.e. they are capable of perforning an explicitely requested task *and* of contributing a sub-task to another process. The method of being activated is implemented in form of a relevant parameter setting.

Parameters are used in most of the tools in varying numbers and with the respective functionality. In contrast to directly using the commands of the base software, there is a *catalogue* of reserved parameters pre-defined in name and purpose which is common to the entire production environment. Other than contributing to the ease by which the tools can be handled, the main benefits of such standardization are related to maintaining the system at the software level. Parameters may represent items of various type, such as

- files, path names, references to intermediate or final products,

- scale numbers, projection parameters, etc.,
- lists and text strings,
- display, error handling, and other type of processing modes,
- data formats,
- expressions (e.g. selections),
- a broad variety of other items.

In the following, a typical production chain will be explained by going through the processing steps of a sample product. In the course of this, some of the tools developed to perform relevant tasks are discussed, and some others are briefly mentioned. As already indicated, the objective is not to present the full set of available tools but to demonstrate the way the tools interact with the production framework illustrated in Figure 2.

Processing of a Sample Product

The sample product might be part of a world atlas, showing parts of Europe at a scale of medium detail (1 : 750 000) and featuring themes such as administrative boundaries at various levels, river networks, towns and settlements, and infrastructure, plus a more detailed set of information on road classification, travel distances, and tourism. The headings correspond to the production phases listed in Table 1.

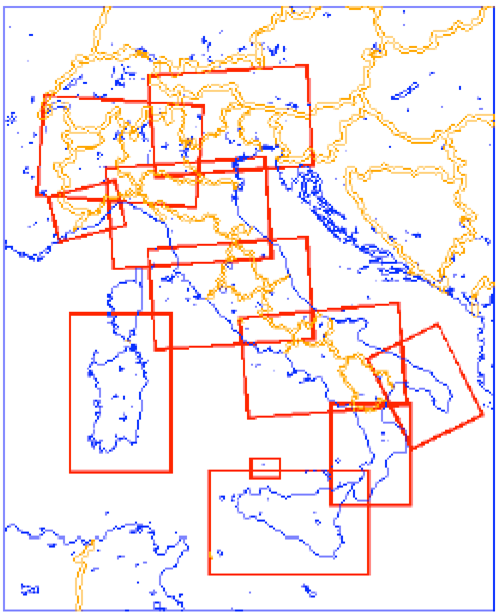

Figure 3. *Map frames in sample product*

Preparatory work (e.g. map frame positioning)

The starting position is an arrangement of atlas pages that has been created in the relevant projection using standard functions provided by any GIS software (Figure 3). The individual frames contain unique identifiers later used for establishing links to tables associated with individual products and holding product specific information. This may be the page numbers actually shown in the atlas, map titles and subtitles in the relevant language(s), or details specifying the position of scale bars, images, logos, or any other information shown on the final product pages.

Data collection

Any product release should go along with bringing database information up to date. The normal procedure is to generate color proofs of the area of interest, to be used for verification and as a base for compiling work. As a rule, regular tiling systems are used for printing out the data, but any other reasonably sized areas may be considered. The updates marked on the compilation sheet will subsequently be incorporated into the database for products to show the current information.

One single front end tool has been implemented for all kinds of database visualization. In the process of actually generating the plot, this tool interacts with a sizeable number of other tools, each of them handling a particular aspect of the graphics or, in most instances, controlling the display properties of an individual map theme. Overall control is achieved by the choice of parameters specified when invoking the tool which, for instance, include references to the projection, the product type and number, the tiling system to be used, and key settings such as output format and resolution. A total of three parameters are reserved for scale settings - referring to the scales of the base data, to the printed map, and to the symbols. Optional parameters specify in which language the map texts will be shown (provided this information exists in the database), whether certain features should be printed in true or in subdued colors, or which state of the data is to be considered (there is the option to visualize data for verification *before* they are fully integrated into the database).

Database updating

This is editing work for which most GIS packages provide a large selection of standard tools. However, it has turned out that, with the help of even more advanced functions, editing could be speeded up significantly, and data quality could be improved. From the variety of additional tools that have been generated, a small selection only is presented.

Alignment to geographic grid: it is common in medium and small scale maps that features (mainly point symbols and texts) are shown aligned to the geographic parallels. This need is not covered by any commercially available GIS software, and a number of tools have been developed around this need. Parameters to be specified simply include the scale of the data and a key for the chosen projection. In this context, it was also found useful to create a tool for temporarily showing auxiliary lines (such as a particular geographic parallel or longitude) in order to facilitate the manual positioning of map features.

Display modes: symbols are shown during editing by relating map features to symbol sets, or styles, and displaying them at a specified display scale. While this is sufficient for most purposes, it has been found advantageous to implement additional capabilities that may be used in complex editing tasks. One of them is the combined display of the fully dimensioned, scaleable symbol and a thin, non-scalable center line. It is also possible to show the fully dimensioned symbols in a subdued color (which is useful when editing symbols of dark colors) or in a single, uniform color shade. Also, a mode distinguishing edit from background features has been implemented. While the latter is not principally a new thing, it is the ease by which switching from one mode to the others can be done distinguishing it from standard software. Another extension is the capability of automatically optimizing the order in which map themes are displayed, as a function of the nature of the current editing job.

Feature combinations: the efficient positioning of frequently occurring combinations of map features is rarely provided by standard software. With a set of tools dedicated to this purpose, symbols frequently shown in the same combination may be automatically pre-positioned by taking into consideration the graphic dimensions specified from a choice of relevant input parameters. Another group of tools exists for aligning the symbol pairs along accompanying line elements in the correct distance. A second example is the automatic orientation of distance markers at angles being calculated from the course of a road.

Map texts: quite a number of functions have been implemented in connection with positioning map texts. Other than aligning texts geo-parallel (which has already been mentioned), the automatic placing of regularly spaced texts plays a major role. The key issue is the efficient positioning of individual characters. Both single line and multi line spaced texts are covered by the tools. Another set of functions provide for the quick and flexible replacement of characters or combinations of characters in the text strings.

Copying and moving features can be a cumbersome task with standard software if feature attributes are to be involved. By implementing the capability of considering any combination of attributes, domain and default values when features are copied, a major source of error has been eliminated. Additionally, the capability of copying / moving features along standard lines such as geographic longitudes and parallels has speeded up many of the tasks.

Preparation of product specific symbols and text fonts

Generating products to be published in their first edition normally involves experimenting with cartographic symbols and text fonts. Again, this is one of the areas where most standard software lack efficiency: with most of them, systematically modifying symbols in large numbers is tedious and takes an excessive amount of time. Using the tools developed in this category, it is easy to process groups of symbols (text symbols as well as points and lines) in combination. For example, the casings of all types of 4-lane highways may be broadened by a factor with one single instruction. A second group in this category is the spontaneous generation of symbols of subdued colors, uniform color shades, and other effects. This is also an example of sets of tools capable of being called by other tools such as the ones mentioned in the previous paragraph dealing with display modes in database or product editing.

Data transfer from database to product

As already indicated, products are prepared in their own product space into which the relevant data are transferred from the central database. The process is fully automated and only requires specifying a few key parameters in the relevant tool - this will start a program which, as part of the transfer, automatically handles the necessary transformations such as change in projection, scaling, and rotating (i.e. aligning the data horizontally along with the map frame). A second tool, assembling main and add-on maps, is activated if the product contains more than one map. Again, this is a fully automated procedure handling all the necessary map to page transformations. The required coordinate details are extracted from the arrangement of map frames that have been prepared in Phase 1 (Figure 3). The parameters in both tools refer to the number of the product and the page, the map scale, reference to the final projection, and a list of themes to be considered. Since data may come from a distributed database with the individual components potentially showing differences in database definition, a third key tool, handling various harmonization aspects, has been developed in connection with data transfer.

Refinement of product data

The refinement step basically consists of two activities: generating *second level features*, and optimizing cartographic visualization. The former refers to deriving features not kept as separate entities in the database from original data - for instance, this is the case with buffers being drawn around administrative boundaries, or with frames of various types around texts to be highlighted. The latter activity is required because features, which exist in the central database in a single representation, need to be manipulated to suit a potentially indefinite number of map frames (characterized by locality and extent) and variants of a frame (characterized by the type of information shown): features, as they arrive from the database, may be cut off at map edges, overlap with features or themes that have been added from sources other than the central database, or shown otherwise inappropriately as a result of changes to map scale or to symbolization (for example, it may be desirable to keep symbols at their original size while scaling down the map by a factor). Consequently, tools are organized in two function groups.

Since the range of products is constantly expanded and derived elements may be shown differently in each product (for instance, buffers around boundaries may vary in many details), the set of tools around this topic needs a thoroughly designed set of parameters. Another key requirement is the systematic and logically consistent arrangement of tools and menues.

The key issue in the feature positioning topic is concentrated around synchronizing the activities taking place on the database and on the product sides. This starts with a specialized tool for automatically flagging the features being modified in the process of optimizing their position (e.g. edge-cleaning), and ends with a procedure for emulating those activities when products are to be updated with the latest database information. A more detailed description of this method is beyond the scope of this paper.

Preparation of frame details

Having completed work on the map data, details on the map frame need to be prepared. Again, the range of possibilities is extremely wide. Tools have been arranged according to the degree of relationship with map and product specific details. In this way, three categories have been identified.

For a start, it is the rule that the set of available standard tools referring to legends, scale bars, north arrows, and to the frame itself needs to be extended to meet the requirements given by a company's product range. The most urgent need for own tool development, however, arises in connection with frame elements directly associated with the map's data and properties (i.e. its geographic position). Elements in this category include references to adjoining pages, references to geographic grid (longitude / latitude numerals), index grid numerals and destination indicators. Obviously, tools automatically calculating such elements need to access data specific details as part of the algorithm. A third group refers to product specific details and includes items such as map titles and other illustrative text, page numbers, and the exact position of particular frame elements. Most of the elements in this category are extracted from product specific data tables as outlined in the 'Preparatory Work' paragraph.

Proofreading

Processing work needs to be verified before the final printfiles can be generated: a minimum of one iteration of proofreading and performing final corrections is essential. There is no specific tool for creating proofs – this is implemented as one of the options in the tool producing the final printfiles.

Normally, a separate printout is generated of each of the product pages. In products consisting of pages showing neighbouring regions with significant overlap it is useful to have a tool that temporarily accumulates the individual pages to one dataset. Having completed proofreading or other tasks that may be dealt with, the same tool subsequently distributes the data again to the various pages, carrying out the tasks in reverse order. The procedure is controlled by the scheme of map frames as introduced in the 'Preparatory Work' paragraph and illustrazed in Figure 3.

Final corrections

Again, editing functions are needed, and, in principle, the tools developed for database editing are used. Some of them had to be furnished with additional complexity in order to cover the capabilities of performing the required functions in a data set that has been rotated (i.e. has been aligned horizontally along with the map frame). Also, the functions referring to automatically flagging features when modifying them need to be applied, as described in the 'Refinement of product data' paragraph.

In this step, it is also important that thorough checks on data quality and consistancy are applied. A wide range of general and specific tools have been developed in connection with data quality and consistancy checks. Exemplary, only a few of them are mentioned, such as the ones scanning for

- invalid lengths and angles in lines,
- logical consistency of road signs to roads, texts to underlines, and the like,

- features not being geo-parallel,
- duplicates in both the spatial and attribute domains (e.g. identifiers occurring more than once),
- deviations from values exceeding a treshold.

Many of the tools in this category can be used in the mode of being accessed from a plotting routine and, thus, provide an early warning mechanism, preventing execution of the final step of printfile generation with imperfect data.

Production of printfiles

In this final step, separate tools exist for each of the product groups (see Figure 2). Most of the parameters are common to those used in the database visualization tool, and also the routines controlling graphics parameters and display properties of individual map themes are accessed by all plotting tools. As a major difference, there is the capability of performing various combinations of consistency checks as described in the previous paragraph.

Conclusion

In total, about 150 individual tools have been developed and organized in functional units. Initially, there was a clear distinction between front end tools and those performing sub-tasks in the background. Meanwhile, a significant proportion of the tools may be operated in either mode, i.e. they are capable of carrying out an explicitely requested task *and* of being called from another tool. In addition to these 150 tools, approximately 100 others exist specifically for contributing to the plotting environment, e.g. by controlling the display properties of individual map themes.

The production system, as described previously, has been gradually established in various steps of refinement. Although there has not been a systematic evaluation that allows quantification of performance, some trends could be observed in the course of expanding and fine-tuning the functionality.

As a rule, problems tended to increase in situations where spontaneous solutions were added focussing on the needs a particular product type or product. In all such cases, it proved to be inescapable to go through a phase of decreased overall performance when, in the urgancy to counteract system disintegration, harmonizing measures eventually needed to be taken. The gist of this is that tools should, at any rate, be implemented in a form as generic as possible, even if the benefit of doing so is not immediately apparent. Also, a top-down approach in the conception phase is preferable to a bottom-up strategy putting too much emphasis on specific details. Provision should also be made for every conceivable extension in the definition of system components such as variables, parameters, and value domains.

What has proved to be extremely useful is the consistent separation of activities and *deliverables* conforming to the various levels as described with the production framework (illustrated in Figure 2). A logical structure in the production framework as a whole offers decisive advantages with respect to controlling and optimizing data flow, and, in the end, provides the ground for tools to be manageable and powerful at the same time.

Consistency in the source data is another issue. Although there is no need for the data to reside in one single database, it is highly recommended to keep the components as homogeneous as possible in terms of data definition and other properties. To be avoided at all costs is the dissemination of structural changes made in the database to the products before all the tools and routines affected by the modifications have been adapted.

A common misconception is that standard software packages offer the optimum in efficiency. They can't. It is not possible for software vendors to anticipate the whole range of needs of everyone of their prospective customers, and what might be a function extremely useful for many of them might be a nightmare for someone else. In any production environment characterized by a certain degree of regularity, the development of tools customized to suit the organization's specific needs definitely has the capacity to provide a substantial return. As could be expected, the highest rates of savings are achieved in products consisting of a large number of individual maps (such as atlases), product series using the same base data with additional themes in varying combinations (which is common in maps on travel and tourism), and in products that are to be updated frequently.

As a matter of fact, the main concern in a commercial production environment is related to cost benefit considerations, and a trade-off needs to be found between time spent on program development and manually working on tasks not covered by customized procedures. At any rate, the potential for effectively employing time-saving algorithms is higher if the production environment has been set up in a way that supports work flow standards and technical parameters common to the whole range of products. If designed and implemented properly, such a framework can significantly increase overall productivity, and free the resources available from tedious routine tasks to be spent more sensibly on other efforts such as data collection or generally improving product quality.