Automating the generalisation of geographical data: the age of maturity?

Anne Ruas
COGIT Laboratory
2 avenue Pasteur 94165 Saint Mandé - France
anne.ruas@ign.fr

Abstract

The paper tends to present current generalisation needs in NMA and summarises current state of research in generalisation, analysing the results of the Agent project. Then we try to identify points on which research should be carry on in order to improve future generalisation packages for NMA data bases and maps production.

1- Generalisation needs in NMA production

Automating generalisation process became a real challenge as soon as NMA started to create geographical data bases in the 80's. At that time the mental schema was the following: Let's create one data base and generate maps (or data bases) from this data base, automatically (figure 1).

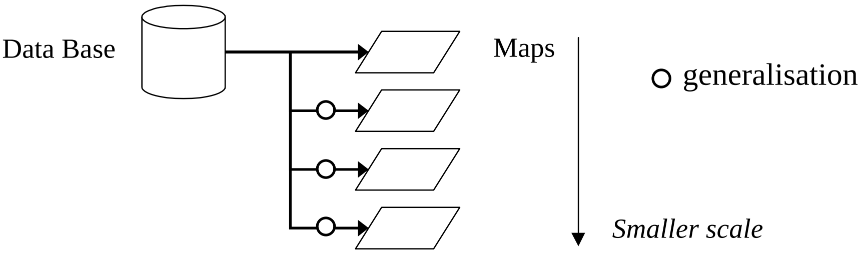


Figure 1 : Mental Schema: one data base, several outputs: a dream?

Unfortunately, we know that creating a data base takes time (the more accurate the longer) (Figure 2) and that generalisation process is hard to automate.

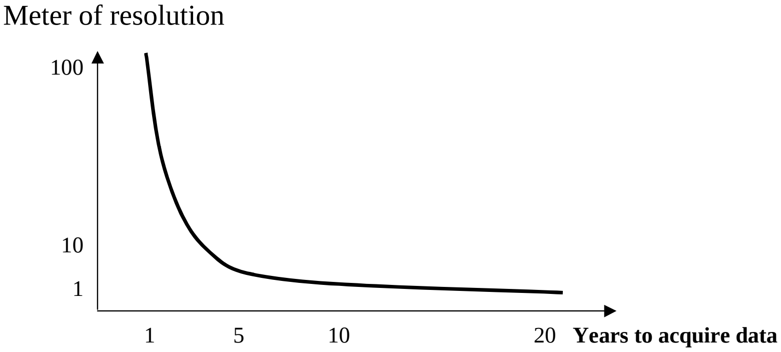


Figure 2 Relationships between data base resolution and duration for the acquisition of a large data base

Consequently, NMA have decided to create several data bases at different resolutions, with different time schedule. Of course, generalisation remains necessary as each data base is supposed to produce different maps at different scales. This solution was certainly the optimal one to ensure the production of different maps (and data bases) without waiting for acquisition delay.

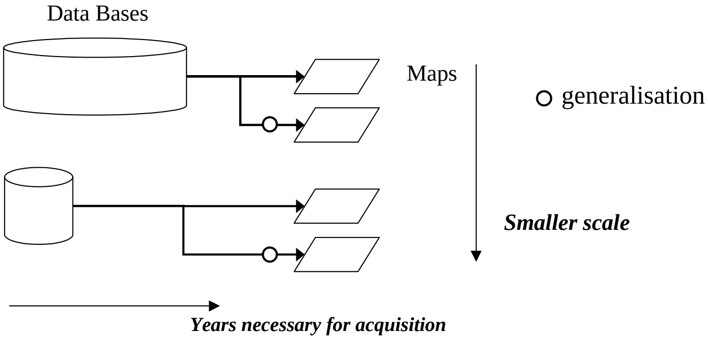


Figure 3 Time and scale

But then appears the problem related to map and data base updating. It is well known that one of the main quality of a data base is its actuality. In the absence of automated generalisation processes, NMA introduced the concepts of cartographic data bases (see ATKIS DKM DLM). As generalisation is costly, the idea is to generalise once the data base with specific symbolisation constraints and to save this new data base as a 'cartographic data base' (Figure 4).

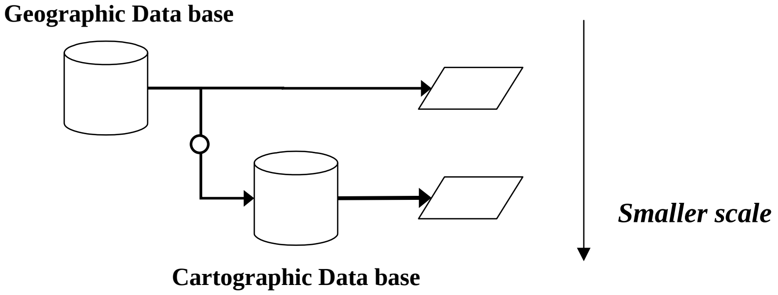


Figure 4 Adding a data base between the source and the generalised map

In such a way, instead of generalising all the data set after each update, generalisation process can be reduced to a local operation around the evolution (see δ and δ' in figure 5) (*the evolution refers to the set of data that have changed between two versions of the data base*). Unfortunately, even this local generalisation is not automated as managing the side effect of an update has not been studied in this context. The introduction of evolutions in a geographical data base may have side effect. For example the construction of a new district of houses is more than the creation of n new houses: during the generalisation of the evolution, the new district should be identified and generalised as a meso object [Ruas 99]. Consequently, generalising evolution data is also a complex task even if the quantity of operations is less important than generalising the entire data set. Of course to maintain data bases consistency, the evolution data should not be directly introduced in the cartographic data base.

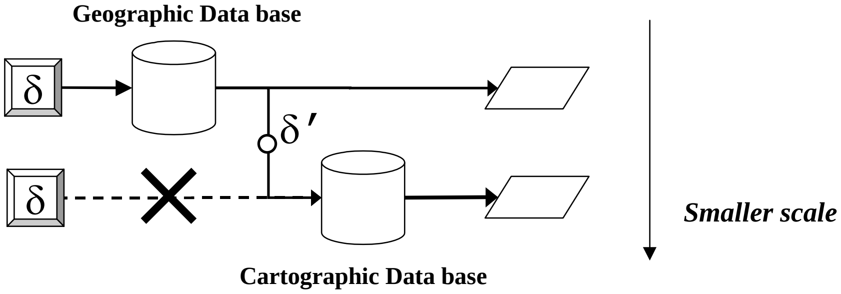


Figure 5 Generalisation of the evolution data

Updating several geographical data bases makes things more complex and the NMA would like to reduce this operation as much as possible. For that the ideal process would be to collect accurate evolution data and to propagate it to all the data bases by generalisation. A solution proposed by Badard (see for example [Badard and Lemarié 01]) consists in 'linking' existing data bases by means of matching process and to use these new relationships during the propagation process.

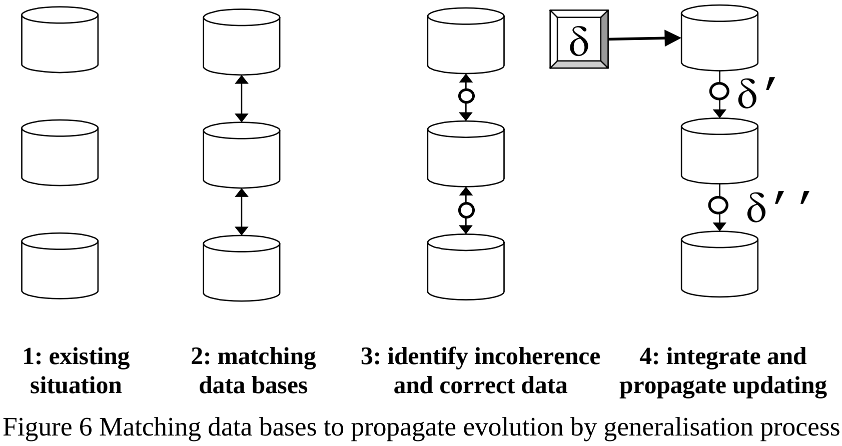

δ
δ'
δ''
1: existing situation
2: matching data bases
3: identify incoherence and correct data
4: integrate and propagate updating

Figure 6 Matching data bases to propagate evolution by generalisation process.

Current research related to multiple representation is proposing models either to link the schema and the data by means of association links or to allow objects to hold multiple representation of themselves. The ideal solution in order to guaranty an optimal data management would be to unify data in a single data base (using data matching, control of coherence and local multiple representation) and to derive data bases and maps on demand (Figure 7).

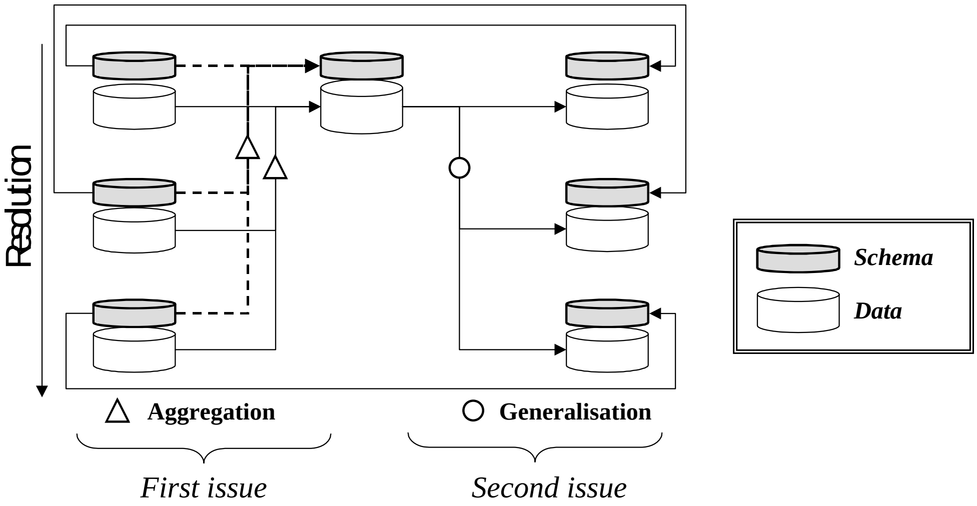


Figure 7 Unifying data base and deriving dynamically appropriate products (maps or data bases)

To conclude, generalisation process is essential in NMA : We need generalisation to produce maps (when they are less detailed than data bases figure 4), we need generalisation to propagate updating between data bases (figure 5 and 6), we need generalisation as a new way to produce data bases and maps from an integrated unified data base (figure 7). The next chapters tend to evaluate the gap between today results and these NMA production needs.

2- Building GIS generalisation package : the Agent prototype

Generalisation is studied since the seventies. Research can be decomposed in three main parts:

- the principles of generalisation: How does/should it work? What information is necessary to make it work ?
- the conception of algorithms going from line compression to more contextual operations such as aggregation, selection or displacement. This part also includes the conception of measures to guide the generalisation process.
- the conception of generalisation packages (e.g. MGE/MG; Change; Mage) that effectively generalise geographical information by means of implemented algorithms and a model that lays on principles of generalisation.

To solve generalisation, we need principles to know how it should work, a large set of algorithms (often new algorithms) and GIS platforms. In reality, some researchers are proposing principles, some are conceiving few algorithms on their own home system (few because these algorithms are complex) and GIS producers are proposing closed systems. In this context, progress is very slow and difficult. Some very good algorithms exist but the point is: could they work together to generalise geographical data? It is not possible to propose a good generalisation package without a strong collaboration between researchers, developers and GIS producers. This kind of collaboration should be based on the integration of robust principles, the integration of robust algorithms and an evaluation stage in order to improve principles and algorithms according to the quality of the results (see Figure 8).

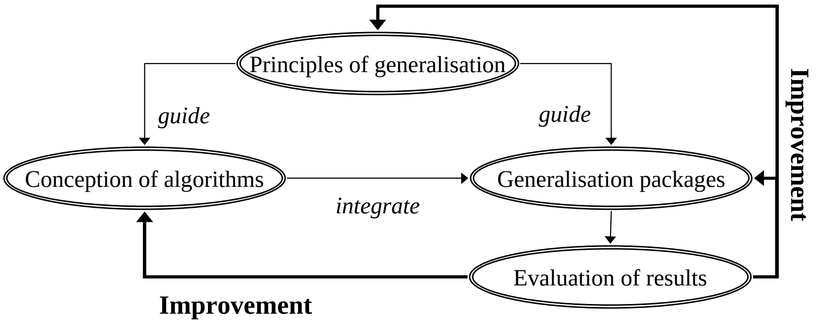


Figure 8 Including evaluation stage in order to improve concepts and algorithms

The Agent project (see http://agent.ign.fr) is a first attempt of such collaboration between GIS producer and generalisation researchers and developers. This three years European project started in November 1997 and finished last December 2000. It associated the COGIT laboratory (A. Ruas), the university of Zurich (R. Weibel), the university of Edinburg (W. Mackaness), the Leibniz laboratory (Y. Demazeau) and the GIS producer Laser-Scan. A lot of recent papers ([Haire and Hardy, 01] [Barrault and Weibel 01] [Duchêne 01] [Regnauld 01]) are presenting some parts of the project and its results.

The main generalisation principles used in this project are the following [Ruas 99] :

- Generalisation is a state change process: initial state and ongoing state are necessary to find the best next state.
- At each step of the process, the point is to find the best next action to realise. This action is the one that will solve conflicts while maintaining, as well as possible, the geographical meaning
- Operations are decided at the geographical object level: An object analyses itself (its level of happiness) and chooses the action that might improve itself. A geographical object that generalises itself is called an agent.
- The level of happiness is based on an evaluation of constraints violation at the agent level. Constraints allow to translate user needs (implicit and explicit) into constraints on object properties (such as size, shape, position, orientation, density). For example a building should be bigger than 300 m² , if its size is 200m², its size constraint is violated.
- The choice of the next best solution is based on procedural knowledge represented at the constraint level (e.g. to solve a building size conflict, use a dilation). The choice of the best constraint to solve first is based on constraints severity and type (e.g. it is better to solve density conflict before proximity conflict, whatever the severity of proximity).
- To manage contextual decision, meso agents are created : these meso agents are geographical objects composed of single objects: A urban block composed of buildings is a meso agent, a road composed of road segment objects is also a meso agent,
- As procedural knowledge is not perfect, after each trial (decision of an object), the object analyses itself in order to validate its state or to backtrack and to try another solution if its state is worse.
- [Regnauld 01] introduced the 'hill climbing' mechanism to the Agent project in order to test different solutions and to choose the best one amongst several. This mechanism is very useful for uncertain procedural knowledge or for complex geographical configurations.

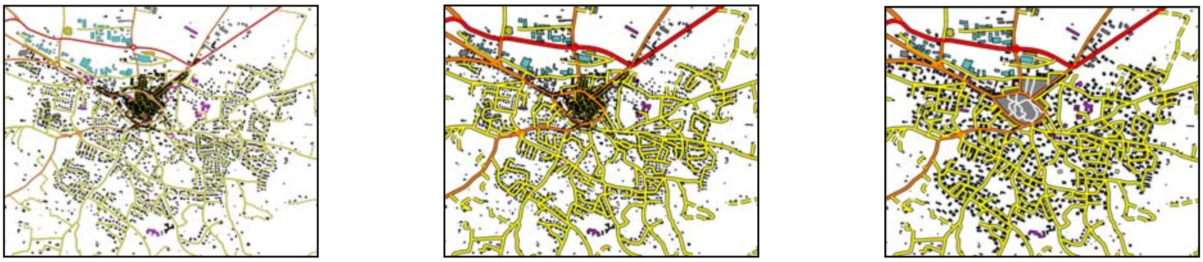
In terms of modelling and dynamic, the existing Agent package owns :

- The generic data schema that allows to represent the necessary information for generalisation:
  - The meso agent and micro agent generic classes and their communication mechanisms
  - The generic class of constraints with its appropriate attributes (ongoing value, priority, severity, importance, flexibility, proposed solutions) as well as functions to compute the value of the attributes
- The Agent generic engine that allows to check the state of the constraints, to choose an operation, to activate this operation, to evaluate the result, to compare the state with previous stored states, to validate, store or unvalidate the state, to choose another operation in case of non satisfaction without requiring to recompute all the constraints. The meso Agent generic engine also activates the generalisation of its components.
- A geographic data schema adapted to the data we have studied during the project : roads, buildings, urban blocks, and towns. These classes own algorithms, measures, constraints and procedural knowledge necessary for their generalisation.

During the project, the team developed 25 generalisation algorithms, 8 control algorithms and 29 measures.

- The building class owns 10 algorithms and 10 measures
- The urban block class owns 5 generalisation algorithms and 3 measures
- The town class owns 2 algorithms (one for street selection, one for urban block management) and 1 measure
- The micro road class owns 8 algorithms (6 for generalisation and 2 for segmentation) and 11 measures
- The meso road class owns 4 algorithms of micro road control
- The road network class owns 4 algorithms (3 for generalisation and 1 for micro control) and 4 measures.

Figure 9 is showing some results produced by the AGENT package at the end of the project (December 2000)



Données initiales, symbolisation au 1:15 000

Données initiales, symbolisation au 1:50 000

Après généralisation, symbolisation au 1:50 000

Figure 9 Some AGENT results produced with one click

First conclusions

First of all, this project shows that a defined, structured and financed collaboration between GIS producers and researchers is of great interest. The conception of such a package would have been very difficult in a university or a laboratory because it requires human resources and a robust GIS. At the COGIT laboratory the two home platforms developed for generalisation research (PlaGe for roads and Stratège for urban areas) were very useful to understand more about the process, and to conceive new algorithms but they were definitely incomplete and they more specifically suffered from a lack of GIS capacity (such as loading a large data set or managing well state versioning).

Secondly, in terms of results, even if some improvements are still necessary (we had no time for an improvement process as proposed in Figure 8) the principles of generalisation introduced in the system (i.e. autonomy, levels of analysis and constraints) are appropriate for generalisation. It does not mean that this is the only solution but at least this solution works.

Lastly, the real generalisation of data sets, without human interaction (one click generalisation) is absolutely necessary to identify the limits of proposed models and to investigate new research domains to improve (and allow) generalisation.

3- Lessons from the Agent prototype

In the past, one of difficulty to generalise data was related to the lack of package that would integrate a set of algorithms and an engine that could dynamically chose an action. The Agent prototype allows us to evaluate what is still missing to perform generalisation as it is required in chapter 1.

3.1 The lack of spatial analysis tools: Characterisation and Evaluation as a core of the success

In the step by step process proposed in the Agent package, one of the main limit is due to the lack of spatial analysis tools. In order to generalise buildings one by one, we used 10 measures, and for the roads 11 measures (some are still missing such as a granularity measure). If the principle of 'analysis, action and validation' works properly, it relays on the existence of measures that should be both robust and complete. When evaluating the Agent generalisation results and process, we noticed for example that some bad solutions were accepted because a measure was missing, or that some good solutions were rejected because the interpretation of the value of the measure was not perfectly tuned. Moreover we noticed that different measures should be used to guide the process (characterisation) and to evaluate each proposed solution (evaluation).

To obtain better results, the research community needs to put some more efforts on the conception of contextual analysis tools. If we analyse again advice given by [McMaster and Shea 89], or [Weibel and Dutton 98] we can notice that very few progress has been made to enrich the contextual measure library. Certainly, as long as a measure is not related to a concrete use in a process, it seams to be useless or not gratifying. We hope that the existence of a package that allows the easy integration of complex measures will incite new works in this domain.

3.2 Enriching representation in order to allow direct interactions and negotiation mechanisms

Amongst contextual operations we can distinguish the operations that concern a single type of objects (such as selection/removal, aggregation) and the ones that can be applied to a mixture of objects such as displacement (in a way, displacement could be viewed as the generalisation of 'free space' between geographical objects). In the Agent package, in order to generalise a set of objects, some meso agents are created. These meso agents are necessary to perform locally some contextual operations such as street removal, building removal or building displacement. The creation of meso agents allows to define some working areas where algorithms will be applied. The definition of working areas based on spatial analysis (a town is defined by the limit of dense urbanisation, a urban block is defined by a minimum road cycle) greatly simplifies and enables the choice and control of contextual algorithms.

But this hierarchical space partitioning has drawbacks: Whenever an object belongs to a meso agent, it can not easily maintain some properties with an object that belongs to another meso agent: for example two close buildings that belong to two different urban blocks can not a priori control the preservation of their relationships. The hierarchical organisation of data proposed in the project helps a lot to generalise the data but is not sufficient to manage properly all geographical situations.

To sum up: two objects can not interact one to another if they do not belong to the same meso object, and moreover, they never directly interact (except for displacement). If the concept of meso agent helps a lot and simplify decision making, it is essential to add the possibility of having direct interactions between agents: for example it would be meaningless to be obliged to create a meso agent to describe the relationship of proximity between an isolated building and a vertex of a road. However this relationships might be essential to maintain as well as possible geographical meaning during the process.

Representing and describing proximity between objects:

In order to allow direct interaction between objects, proximity relation should be explicitly represented. The interaction between two agents would allow the introduction of a very necessary concept : the dynamic negotiation. The meaning of a negotiation is a following: As an agent, if I am in relationship with another agent, my decision should take into account this relationship. At this level different strategies are possible: I can inform the other agent of what I want to do and ask for an acceptation or better, my decision is not only driven by my own satisfaction but also to the nature of my relationship with this agent and its own desire.

Proximity between a small area (or symbol) and a line could certainly be represented by relative polar coordinates (ρ, θ), from a position (x,y) on the line (Figure 10). In such a case, the building would easily receive a message from the line, preserve the orientation (θ) and change the distance (ρ) to maintain 'minimal distance' between them. Even better, the building should know that it has 'the same orientation' than the road.

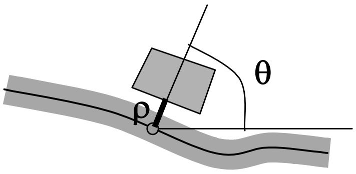


Figure 10 Representing neighbourhood relationship between a building and a line

Representing free space

Also, free space could be explicitly represented as an actor of generalisation as generalisation is related to space competition. Structures such as Voronoï diagram and Delaunay triangulation used for analysis or displacement are partial solutions in order to correct the drawback of vector representation of geographical information. The field is the underlying and hidden connection between objects: a large part of contextual analysis is more or less trying to correct this lack of data modelling introducing complex data structures. In non urban area, we could try to make analysis of free spaces, distinguishing separation area between objects, objects self free areas (that are object and not field dependant) and free area which also is an information. Before starting generalisation, dilation and aggregation mechanisms (eventually based on Voronoï) could be used in order to structure free and separating space and we could imagine using weighted grid structures (as in name placement process) in order to facilitate and control interactions and negotiations between agents. Moreover this kind of new linked field-and-object representation would allow to take into account DTM as a constraint to better generalise topographic objects. Of course the principle is not to create areas with fixed boundaries.

3.3 Identifying and divided specific tasks for the generalisation process

To perform generalisation, five main tasks are necessary:

1. To load the data into the generalisation package by inheritance, without losing the initial data information.
2. To acquire user needs to adapt constraints and priority between constraints for each generalisation
3. To create structural objects such as town, street network, urban blocks and others: This stage is a data enrichment that looks for specific (and classical) meso objects that will be used during generalisation process
4. To generalise the objects
5. To evaluate the result of the generalisation to give accurate information on the obtained results.

Each of these tasks requires specific research and appropriate tools:

Task 1: Load the user data task

Current research on conceptual modelling [Parent et al 99] are proposing ways to create spatio-temporal conceptual model: Interfaces under development allow to describe classes, relationships and constraints for a future data base. For generalisation, it would be very useful to be able to view by reverse-engineering the user and generalisation existing data schema (e.g. the Agent one) and to fit these schemas together in order to propose an appropriate and well constrained new data schema for generalisation (Figure 11).

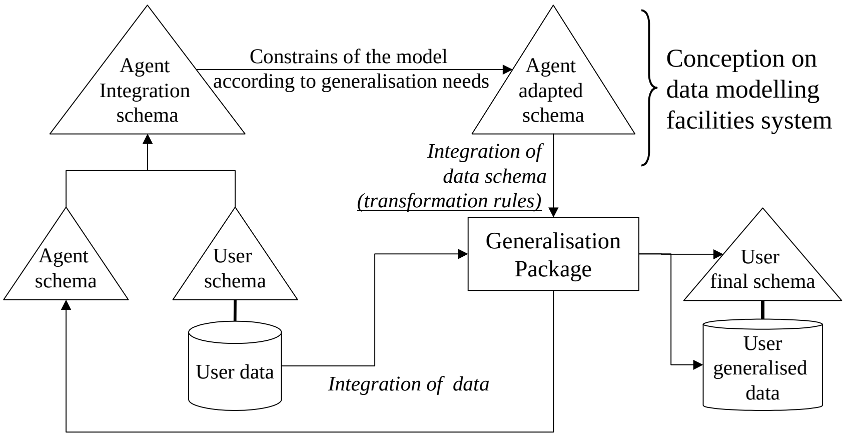


Figure 11 Toward new generation of data modelling facility for generalisation purpose

In such a way - and if such a modelling system could be conceived - aggregation rules and constraints between initial data schema and the required generalised one could be easily introduced in the adapted generalisation schema. The adapted schema for generalisation (based on the agent schema, the initial user schema and the aggregation rules) could be easily created and integrated into the generalisation package. At the end of the process, the unnecessary classes (as meso classes, constraints) will be removed to provide the required data schema and generalised data.

Task 2: User needs acquisition task

Another weak point of the Agent package is the difficulty to easily introduce user needs. If it is possible by means of constraints, it is still limited to the experts. Some ongoing IGN research [Hubert 01] tries to make it easier for non expert users using the concept of samples to define constraints goal value and relative importance. The objective is to propose to the user visual samples of treated data that the user selects to explain its needs. Internally, each sample is described by means of a set of constrains value that can be directly understood by the system to tune each generalisation.

Task3: Enrichment task

The enrichment task would be triggered as a first step of the process to look for meso objects and to instanciate meso classes. This task would detect towns, urban block, district, ring road, building alignment, free space, relief structure and other gestalt or phenomenological information.

Task 4: Generalisation task

A generalisation task would automatically trigger the generalisation and could change its 'global plan' (see [Ruas Plazanet 96]) according to geographical situation: as it is now in the Agent package, the global sequence of generalisation is fixed.

Task 5: Evaluation task

The evaluation task would check for each agent happiness and would be able to summarise intelligently these information (by object type, by severity) in order to give a synthetic and structured view of the quality of the final generalised data.

The introduction of these specific global tasks would offer the possibility to better structure the generalisation process, allowing both to make progress in these areas but also to improve generalisation packages. As it is now, tasks are hidden and reserved to experts.

3.4 Dynamic learning

At least, we believe that a learning task could be very useful in order to detect and improve process. During generalisation, many repetitive sequences of operations are done (such as building or road generalisation). If procedural knowledge is not appropriate or not well tuned, the engine might repetitively try a wrong sequence of operations. A learning task could easily trace and structure the process (type of object, type and severity of conflicts, operation tried, success/failure) in order to learn more about the system efficiency (the capacity to solve the given problem in the best way). At first time the process evaluation could be used to better tune the system after a first trial on a subset of objects. Then, we could imagine that 'learning agents' could dynamically identify repetitive errors and add rules in order to adapt dynamically the process.

4- Generalisation: the age of maturity?

If the Agent prototype brings new hope to automatically manage generalisation (because contextual generalisation is becoming possible), it allowed us to see the limit of today system and to identify what is missing in order to use such system in production environment.

To go from the present situation to a generalisation package that would answer to production needs (see chapter 1 and future needs including internet access to data bases and on line generalisation), some important points should be improved :

1. The five tasks of generalisation should be clearly divided: data loading, user needs acquisition, data enrichment, data generalisation and data evaluation.
2. In this context, conceptual modelling facility would be very useful for data loading task
3. Researcher should carry on working on spatial analysis in order to create new contextual measures.
4. In order to take better decision, objects should be able to interact directly one to another, which requires a better description of their spatial environment and the conception of negotiation mechanism. Neighbourhood relationships should be enriched and described. GIS package should also start to propose vector-raster joined representation, to identify free space and relief structures as actors of generalisation.
5. User interface adapted to generalisation should be conceived in order to facilitate user needs integration.
6. Evaluation should be introduced as a core of generalisation process in order to improve results and to provide information that would better describe the quality of the generalised data.
7. Procedural knowledge should be improved by means of evaluation tests and learning techniques.

What is mature, after years of research, is a good understanding of what is necessary is terms of tools, knowledge and engine to perform generalisation. The Agent project main result is that we have one solution (not the only one) to combine processes according to data analysis and to procedural knowledge. But to go from this situation to a real use of it, not only basic elements of the system should be improved (measures, algorithms and knowledge) but complementary tasks should be added as a part of the process.

Then, and to answer to the production needs presented in chapter 1, generalisation methods should be studied in the specific case of updates propagation between data bases.

References

Badard T and Lemarié C (2001) 'Cartographic database updating' ACI/ICC Beijing China

Barrault M and R. Weibel, (2001) 'Road Network Generalisation: A framework using a Multi Agent System Approach' GISRUK 2001 Glamorgan,UK pp321-324

Duchêne C, (2001) 'Road Generalisation using Agents' GISRUK 2001,Glamorgan,UK pp 325-328

Haire K and P Hardy, (2001) 'Active Agent Base Approach to automated generalisation' GISRUK 2001 Glamorgan,UK pp 319-320

Hubert F (2001) 'Assistance mechanism use for needs specification in geographical information on the web' ACI/ICC Beijing, China

Mc Master R.B. & Shea K.S. (1989) 'Cartographic generalization in digital environment: When and How to generalize?' In AutoCarto 9 Baltimore, USA pp 56-67

Parent C, S Spaccapietra and E Zimanyi (2000) 'Spatia-Temporal Conceptual models: Data structures + Space + Time' in AAAI-2000 Workshop on Spatial and Temporal Granularity, Austin, USA, July 30, 2000

Regnauld N, (2001) 'Constraint Based Mechanism to activate automatic Generalisation using Agent Modelling' GISRUK 2001 Glamorgan, UK pp 329-332

Ruas A and C Plazanet (1996) 'Strategies for automated generalization' In proceedings 7th International Spatial Data Handling Delft, Netherlands pp 319-335

Ruas A. (1999) 'Modèle de généralisation de données géographiques à base de contraintes et d’autonomie'. PhD Thesis University of Marne-La-Vallée France.

Weibel R. et G. Dutton (1998) "Constraint-based automated map generalization" in Proceedings 8th International Spatial Data Handling, July, 1998, Vancouver, Canada , pp 214-224.