A study on the state-of-the-art in automated map generalisation implemented in commercial out-of-the-box software

By:
Jantien Stoter, ITC, Enschede

Karl-Heinrich Anders, University of Hanover
Blanca Baella, ICC, Catalonia
Dirk Burghardt, University of Zurich
Francisco Dávila, IGN, Spain
Cécile Duchêne, IGN, France
Maria Pla, ICC, Catalonia
Nicolas Regnauld, Ordnance Survey, Great Britain
Peter Rosenstand, KMS, Denmark
Stefan Schmid, University of Zurich
Guillaume Touya, IGN, France
Harry Uitermark, Kadaster, the Netherlands

ABSTRACT

This paper describes the set up and the progress of the EuroSDR project that studies the state-of-the-art in automated map generalisation implemented in commercial out-of-the-box software. The project started in October 2006 with a project team consisting of National Mapping Agencies (NMAs) and research institutes. From October 2006 till May 2007 four test cases of four different NMAs were selected, consisting of a large scale source data set, requirements for the smaller scale output map as well as symbolisation information. Much effort has been put in specifying and harmonising requirements for the output maps. These requirements have been defined as a set of constraints to be respected in the output maps. From June 2007 the project team tested the four test cases with four commercial out-of-the-box software systems: ArcGIS, Genesys, Change/Push/Typify and Clarity. The vendors of these systems performed parallel tests on the four test cases in which they were allowed to customise their systems. An evaluation methodology has been designed and partly implemented. Results are expected by the end of 2008.

1. Introduction

Research in automated generalisation has yielded many promising results. At the same time it seems hard for vendors to implement automated generalisation solutions in commercial software (see for example Stoter, 2005). Since National Mapping Agencies (NMAs) would benefit significantly from automated solutions, EuroSDR (European Spatial Data Research) started a research on the-state-of-the-art in automated map generalisation implemented in commercial out-of-the-box software. In EuroSDR NMAs, research institutes and private industry work together on research topics of common interests.

The main aim of this research project on generalisation is to get insight into what parts of the generalisation process can be automated using commercial software and what parts require manual editing or additional developments. The project can indicate either areas where research results have not yet been implemented in commercial software encouraging vendors to do so or areas where further research is needed. It is important to know that the project focuses on out-of-the-box versions of commercial software and therefore adjusting the software to a particular need, which might be appropriate for generalisation of topographic data, will not be studied.

This project is the first project that tests the quality of the main aspects of complete maps, generalised by different systems, different testers and taking the map requirements of several NMAs into account. Therefore the project also aims to get insight into several major side aspects, such as how to specify requirements for automated generalisation, how do automated generalisation processes work, how to set up a case for studying the-state-of-the-art in generalisation, how to perform evaluation of generalisation output, how does the constraint approach, as adopted in this project, work in practice and what other research is needed in this area?

The project team meets three times a year and consists of six NMAs (KMS (Denmark); ICC (Catalonia); IGN (France); IGN (Spain); OS (Great Britain); Kadaster (the Netherlands)) and three research institutes (University of Hannover, University of Zurich and ITC). Four vendors are participating in the project (ESRI, Axes systems, University of Hannover and 1Spatial). The project started in October 2006 and the final report is expected by the beginning of 2009.

This paper describes the methodology of the project - from problem definition and formalising map requirements, to testing, evaluating and analysing the tests - (section 3) as well as the insights obtained from defining map requirements (section 4). Some intitial results are presented in section 5. First the former research project on generalisation under OEEPE flag, predecessor of EuroSDR (Ruas, 2001) is shortly described in section 2, because the choices made in the current project are heavily influenced by the conclusions of the former project. The paper ends with conclusions in section 6.

It is important to note that the project is a research project which aim is explicitly not to make a ranking of the software tested. Instead the main research question is ‘what does the industry propose and offer in terms of automated generalisation and how can this be used by NMAs?’ Consequently this project will not publish details on the potentials and limitations of individual systems.

2. THE FORMER PROJECT ON AUTOMATED GENERALISATION

The OEEPE project (Ruas, 2001) focused specifically on obtaining insight into generalisation processes for cartographic purposes and less on evaluating the generalised output itself. Only for individual objects, the testers were asked to assess the result in terms of good, bad or medium. Main conclusions of OEEPE project relevant for the new project are:

- Generalisation of individual buildings performed best, while generalisation of roads and of buildings within urban areas showed low quality results. In urban areas contextual generalisation is required to remove and displace buildings in order to keep the pattern;
- The notion of constraints (or in another way formalised requirements for the output data) would have helped to choose the best solution;
- There is not a perfect match between what a user wishes to do (operator) and what (s)he uses to achieve this (algorithm); and
- The tests resulted in large differences in results.

It was also concluded that automated generalisation could be improved by a formalised description of the expected data content. For a future test a more flexible and digital method to evaluate experiments was recommended, since the manual tracing was far too heavy and time consuming. Symbolisation information should be used in a new test to unify the outputs. Finally tests should start with evaluating the status of data with respect to constraints for choosing an operation.

3. METHODOLOGY OF THE RESEARCH

The main objective of the current project is to study the state-of-the-art in automated generalisation implemented in commercial software regarding map requirements of NMAs. The issue of cartographic versus model generalisation is not explicitly addressed in this project. The systems should produce a map according to map requirements. Intermediate results (for example after model generalisation) may be produced by a system but are not evaluated by the project. The scope was narrowed by focusing on large to middle scale generalisation because the involved NMAs considered this as most time-consuming generalisation task of current production lines. Furthermore it was decided to focus on generalisation processes applied to a complete data set; it should not be a sequence of operations that are triggered on individual objects as in the OEEPE project. Finally programming new algorithms was not allowed since the project concentrates on out-of-the-box versions of systems.

3.1 Test cases

A list of known complex map generalisation problems was generated from templates produced during the OEEPE research project, completed with own experiences. This yielded a list of transformations, classified by feature classes. Based on this list, four test cases were selected in such a way that it was assured that all interesting generalisation situations were included (see table 1). Some modifications were done on the initial data sets to make them ready as test cases for the project: details, such as rich classifications, were removed from the data sets and data sets, including their attributes, were translated into English if not yet available. Also symbol descriptions were defined in templates provided by the project team. Symbols were simplified for the project in order to address generalisation issues instead of symbolisation issues. For similar reasons map names were not considered. Figure 1 shows the four test cases of the project.

Area type	Source dataset	Target dataset	Provided by	Nr of input layers	Main layers
Urban area	1:1250	1:25k	OS Great Britain	37	buildings, roads, river, relief
Mountainous area	1:10k	1:50k	IGN France	23	village, river, land use
Rural area	1:10k	1:50k	Kadaster, NL	29	small town, land use, planar partition
Costal area	1:25k	1:50k	ICC Catalonia	74	village, land use (not mosaic), hydrography

Table 1: Test cases selected for the EuroSDR project.

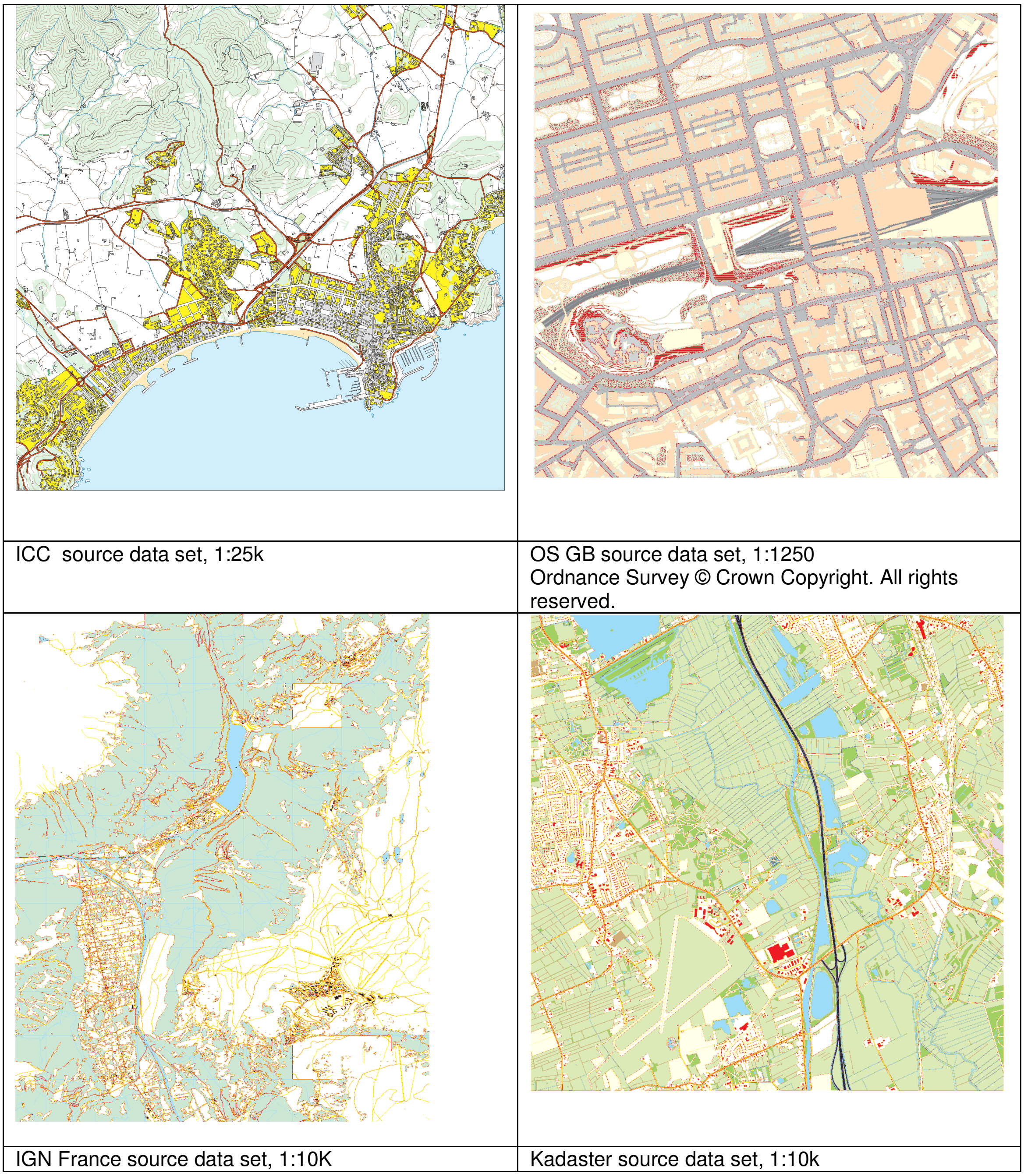
3.2 Defining map requirements

How to define map requirements in a way that it can be unambiguously understood by testers was the next challenge. This is important since it should be exactly clear what a tester should express into the tested system. Specifying map requirements for generalisation of topographic maps is not straightforward. In general terms one can define expectations for a satisfying generalisation solution, e.g. reducing the details to discern regional patterns; an aesthetically pleasant map; a map that reveals or conceals information inherent among a set of abstracted data; a map enabling a user to succeed a given task as exploring, route finding, observing (Mackaness and Ruas, 2007). The difficulty is to specify these types of requirements into such a format and knowledge level that they can steer the automated generalisation process.

Map specifications can be expressed in cartographic constraints to be respected. Constraints are a way of expressing how the generalisation output should look like without addressing the way this result should be achieved. This motivated the project to express the expected outputs in the form of constraints so that it was not needed to define sequences of operations.

There are several examples of different uses of constraints for generalisation. Bard (2004) uses constraints inside an evaluation system that evaluates generalised data once generalisation is finished. Barrault et al. (2001) and Ware et al. (2003) use constraints to control the generalisation process. Burghardt and Neun (2006) use constraints to assess intermediate solutions during the generalisation process, but also in order to compare automated generalisation solutions to known situations. In other works constraints are translated into equations that are solved by optimisation techniques such as the least-squares adjustment of (Sester, 2000) implemented in Push software tested in this project.

Cartographic constraints always exist when you do generalisation. The notion of cartographic constraints is either implicitly or explicitly implemented in generalisation systems. If it is implicitly implemented, it is the user who predefines a sequence of operations that will be applied to sets of objects and constraints are implicitly taken into account by the person who designs the sequence. If the system uses constraints explicitly, optimisation methods are implemented to obtain a situation with as less violated constraints as possible. The possibilities of the specific software in the tests imposed the way the sequence of constraints was addressed during the tests.



ICC source data set, 1:25k

OS GB source data set, 1:1250
Ordnance Survey © Crown Copyright. All rights reserved.

IGN France source data set, 1:10K

Kadaster source data set, 1:10k

Figure 1: Source data sets in the EuroSDR generalisation project. Maps have been reduced in size.

A template was developed by the project team for the definition of constraints for the four test cases in order to have a uniform way for expressing the constraints. The template distinguishes between constraints on one object, on two objects and on group of objects. A weight of the constraint could be indicated to express the importance of satisfying the specific constraint in the final output. Note that this value does not say if the constraint should be solved before the others in the process. Ruas (1999) showed that it may be required to satisfy less important constraints first before very important constraints can be satisfied. For example for building generalisation it is usually better to reduce density before trying to cope with overlaps, even if non-overlapping constraints are more important than density constraints in the final results.

A suggestion ('proposed action') to handle the specific constraints could be indicated in the constraint template as well. The reason for this is that in some cases it was more straightforward to describe an expected result by means of the action required to achieve it. In other cases the way to achieve the satisfaction of the constraint is known by the NMA. In those cases a proposed action could make sure the constraint is well understood. Finally a suggested action may support the tester in solving the violation of the constraint.

3.3 Software tested in the project

The tests were performed on commercial software systems available in June 2007. Based on the defined case studies and the conditions for the project, vendors were invited to participate in the project. Four vendors agreed to participate in the project, which was very valuable for the project. These are: ESRI (ArcGIS), 1Spatial (Clarity), Axes systems (Genesys) and University of Hannover (Change, Push, Typify). In November 2007, first results were discussed within the project team as well as with the vendors. During this meeting it was realised that it would be beneficial if vendors would submit improved versions of their software to be tested by the project team. The main drive for such an extension was that the availability of map requirements and the first test experiences might support vendors to improve their systems. Vendors were invited to submit a new version of their software by 31st of March 2008. Although some vendors showed high interest into submitting a new version of their system, they all withdrew this decision in March 2008.

3.4 The test process

The tests were performed from June 2007 by project team members on commercially available out-of-the-box versions. In these tests no customisation of the software was allowed nor was it allowed to edit results afterwards, i.e. results were obtained without any interactive editing. Every system was tested two to three times for four data sets. In total 30 outputs have been delivered by the project team testers.

To assure that the results would not be limited by software experiences of the testers and to assure that the results would not be limited to June 2007, vendors were invited to do parallel tests with as much customisation as they want (including developing new algorithms) as long as they would report on this. Another reason for allowing (or even encouraging) customisation by vendors in the parallel tests was that some systems contain a toolbox that can be customised with algorithms or services for particular needs. Therefore the parallel tests could show all the potentials of such systems in contrast to the tests on out-of-the-box versions.

In every test the tester tried to translate all defined constraints into a form understandable by the specific software. The generalisation process must either be triggered by a class of objects (theme) or by spatially indicated areas (partitions), i.e. the tester was not allowed to trigger operations on an object by object basis as in the OEEPE project.

For every test case, the following information was produced by the testers: 1) a file which lists every action of the tester and the amount of time that the action took (installing the software, reading the manuals, input of data, pre-processing etc), 2) a file describing how the tester implemented every constraint (fully/partially/not; how was the constraint expressed; how was the constraint handled), 3) ESRI's Shape files of all output layers, 4) pdf-file of the output map. In addition, testers provided general information on the functionalities and performance of the systems. Templates were designed for all these outputs in order to capture all testers' information in a structured and consistent way enabling a flexible method for evaluation.

The first evaluations of the maps show that having so many testers involved, it is unavoidable to have different symbolised outputs, even if symbolisation descriptions were provided. This is also due to differences in implementation of the symbolisation in the systems used to obtain the outputs. Since symbolisation heavily influences the way a map is perceived by the reader of the map, it was decided that one person would redo the symbolisation based on the output shapes in one system and in consultation with the four NMAs before they would be evaluated.

3.5 Evaluation methodology

The three main questions of this research project are:

(1) What are the possibilities and limitations of commercial out-of-the-box software systems for automated generalisation with respect to NMA requirements?

(2) How differently do the outputs respect the specifications and, more importantly, why? This will provide insight into which software is capable of handling which kind of problems.

(3) How different are the outputs for one test case? Solutions that are expected to meet the same map requirements are compared in order to learn more about automated generalisation processes.

In the project there are two types of outputs to evaluate.

The first type of output is the testers' feedback. This information is analysed to see what generalisation functionality is available and missing in current systems as well as how easy it is to use the systems. Analysing the testers' information is done by a) a schematic study of the output documents compiled by the testers on how they expressed every constraint, b) an analysis of the testers' experiences as documented in the system templates, and c) by using the testers' feedback from the project meetings.

The second type of outputs to evaluate is the generalised data. According to Mackaness and Ruas (2007) evaluation of generalised data can be applied in different phases of the generalisation process: before, during or after the generalisation process, depending on the purpose of evaluation. This purpose can be respectively: 1) setting parameters and tuning generalisation processes; 2) controlling the process during generalisation, i.e. trigger generalisation operations and perform intermediate evaluation to see if the situation was improved; 3) determining the overall quality of the solution. The purpose of evaluation carried out in this project falls in the last category.

The evaluation of generalised data is the most time consuming and challenging evaluation for the project. In this evaluation the output maps and output files are being analysed. The evaluation of the generalised data (including the outputs of the vendors) consists of three evaluation processes:

- An expert evaluation in which experts assess the output maps with respect to the requirements
- An automated constraint-based evaluation, in which for a selection of constraints the satisfaction level of the constraints is calculated
- An evaluation which compares the output data to highlight differences in the solutions for one test case and to see how differently the outputs respect the requirements. This comparison of output data will be partly done automatically (i.e. based on statistical numbers per output) and partly by a visual analysis of a selection of situations.

These evaluation processes have been designed in 2007 in an initial state. Based on test evaluations with the first versions of the methodology and based on a project meeting in April 2008 where these initial experiences were further discussed, the methodology has been improved and better aligned with the research questions of the project (see for more details (Burghardt et al., 2008)).

4. INSIGHTS OBTAINED FROM DEFINING MAP REQUIREMENTS

4.1 Capturing map requirements in constraints

The NMAs of the test cases studied their current specifications and processes in order to specify their map requirements in a set of constraints. Several insights were gained during this process.

First well-defined requirements as needed for machine-based processes are not directly available. Although some automation in generalisation has been introduced in a few NMAs, this does not necessarily mean that the requirements are formally expressed. The reason is that currently no formalism has proved to be adequate for fully capturing the specifications of a map. Requirements are available at NMAs either in documents (i.e. specifications), software code or in human minds. These requirements are mostly available at the human knowledge level, since they are meant to be used by cartographers in current semi-automated processes that allow some human interpretation.

A case study was carried out by (Van Smaalen and Stoter, 2008) in order to study if strictly applying currently available map specifications, without adding any human interpretation, will result in expected output. Generalisation regulations meant for cartographers (Topografische Dienst Kadaster, 2005) were formalised in the template proposed by this project and implemented in automated processes. In this case study it was concluded that a map, that is produced by strictly applying a set of regulations will always differ from an interactive generalised. Cartographers can allow themselves some freedom in applying regulations, even in adhering to threshold values. In addition the case study showed that text based map specifications are not complete. On the one hand the case study showed that cartographer's interpretation is impossible to formalise (see also Muller and Mouwes, 1990). On the other hand it showed that we might need better, detailed specifications than the available map specifications which are meant to be interpreted by humans.

In the project, available specifications were converted into the constraint templates as provided by the project team. Although a template was provided, the first versions of constraints, finished by January 2007, showed differences between NMAs, even for similar situations. As a consequence, approximately 250 different constraints were defined covering all four case studies. Hereupon the project team decided to harmonise the constraints in order to simplify the tests: once a tester has expressed the constraint for one case study, it is easy to express a similar constraint for a second case study. Even more important than making the tests easier was to reduce the number of constraints for evaluation. This enables comparing results on similar constraints across the test cases. In the harmonisation process constraints covering similar situations were identified and these constraints were standardised in a way that they were applicable to all four test cases. This resulted in a list of generic constraints for topographic map generalisation at mid-scale within this project. All four NMAs redefined their constraints by specialising the generic constraints using their own feature classes, thresholds, parameter values and preferred actions. There are a few constraints that remained NMA specific because they are dealing with very specific situations.

Constraints on one object		
Constraint type	***Property***	***Format generic constraint***
Minimal dimension	Area	target area > x map mm^2; target area = initial area ± x %
	Width of any part	target width > x map mm
	Area of protrusion/recess	target area > x map mm2
	Length of an edge/line	target length > x map mm
Shape	General shape	target shape should be similar to initial shape
	Squareness	[initial value of angle = 90° (tolerance = ± x°)] target angles = 90°
	Elongation	target elongation = initial elongation ± x %
Topology	Self-intersection	[initially, no self-intersection] no self-intersection must be created
	Coalescence	coalescence must be avoided
Positon/Orientation	General orientation	target orientation = initial orientation ± x %
	Positional accuracy	target absolute position = initial absolute position ± x map mm
Constraints on two objects		
Minimal dimensions	Minimal distance	target distance > x map mm
Topology	Connectivity	[initially connected] target connectivity = initial connectivity
Position	Relative position	target relative position = initial relative position
Constraints on group of objects		
Shape	Alignment	initial alignment should be kept
Distribution&Statistics	Distribution of characteristics	target distribution should be similar to initial distribution
	Density of buildings (black/white)	target density should be equal to initial density ± x %

Table 3: Examples of harmonised constraints.

The harmonisation process resulted in 21 generic constraints on one object, 11 constraints on two objects and 13 constraints on group of objects. Some examples of harmonised constraints on one object, on two objects and on group of objects are shown in table 3 (constraint type as used in this table is explained in the next section). These examples show that still some human interpretation is needed to translate the constraints into the systems. For machine-based implementations further formalisation would be required.
In defining the sets of constraints, NMAs tried to be as complete as possible. However limited time was available for defining the constraints (a period of two months from start to end in which this activity had to compete with 'normal' tasks). In addition the constraints were defined without running any automated generalisation process which would have made it possible to show how specific constraints work when they are brought into practice. Finally the NMAs concentrated their efforts to define constraints on the main problems within the selected test areas. Because of this it was recognised at the start that the constraints would not be complete. The results of the project can help to identify more specifically which constraints were missing. Some missing constraints were identified by the harmonisation process in which the four constraint sets were compared. That is why there were 299 constraints defined as specified generic constraints at the beginning of the tests (50 more than initially).

4.2 Analysis of the defined constraints

It is interesting to have a closer look at the differences and similarities between the constraints defined for the four cases (see table 4). The high number of constraints in the ICC test case in comparison to the other cases is caused by a larger number of specified thematic classes (see table 1). Another difference in number of constraints is caused by the specific test area: for example constraints on coastal features are expected to be dominant in the ICC case.

Test case	Total number of constraints	Number of constraints			Number of constraints on different constraint types								Number of constraints on different feature classes							
		on one object	on two objects	on group of objects	Model generalisation	Minimum dimension	Position	Orientation	Shape	Topology	Distribution / Statistics	Other	Building	Land use	Road	Water	Relief	Coastal features	Any	Other
ICC	137	86	23	28	12	80	0	4	19	12	5	5	39	20	16	25	8	19	9	1
kadaster	52	27	21	4	11	18	1	0	1	6	0	15	10	13	23	3	0	0	0	3
IGNF	61	32	15	14	2	15	2	4	15	12	2	9	33	2	12	9	2	0	2	1
OSGB	49	24	13	12	2	16	1	0	0	8	0	22	24	1	8	1	8	0	2	5
Total	299	169	72	58	16	129	4	8	35	38	7	51	106	36	59	38	18	19	13	10

Table 4: Analysis of constraints for the EuroSDR project, addressing different characteristics

The constraints were classified in order to see what types of constraints were of interest for the four NMAs to put in this project. The classification used in our project extends the work of Beard (1991); Ruas and Plazanet (1996); Weibel and Dutton (1998); Galanda

(2003) and (AGENT, 1997). In our classification of constraints, explained in detail in (Burghardt et al, 2007), a distinction is made between two main categories: legibility constraints and constraints for preservation. At scale transitions, preservation constraints are completely satisfied. These are constraints prescribing topology, position/orientation, shape, pattern and distribution/statistics. Preservation constraints can become violated when operations are applied for improving legibility (minimal dimensions, including granularity). Legibility constraints may be violated through scale changes and applied symbolisation. Harrie (2001) argues further that legibility constraints aim at changing the data, while preservation constraints strive to maintain them. Violation of legibility constraints can be investigated independently from the source data set, while preservation constraints have always to be evaluated in correlation with the source data.

The constraints for 'model generalisation' refer mainly to constraints for removing certain feature types from the data (e.g. 'cycle path' in Kadaster test case or 'wall' in ICC test case), but also to avoid that objects with different attributes are aggregated (for example different types of buildings in OSGB test case should not be aggregated). From table 4, we can see that constraints for keeping minimal dimensions play an important role in all four specifications, showing the importance in the cartographic generalisation process. Topological constraints are defined on a more general level such as 'preserve topological consistency and connectivity', 'self intersection not allowed', or 'keep adjacency'. Noticeably there are only a few shape constraints defined by Kadaster. Position/orientation constraints are sparsely specified by all NMAs and they refer only to buildings. A reason could be that buildings are expected to be moved more than networks with their connectivity constraints (topology). Most constraints are defined for one object whereas fewest constraints are defined for groups of objects, because it was more difficult to formalise the constraints on groups of objects than the constraints on one object. The variation of constraints among feature classes is most probably a result of the relative importance of certain thematic classes within the four chosen cases.

5. Results from analysing the testers' feedback

This section presents an insight into the results of the analysis of the testers' feedback. This is only a first insight as all the tests have not been completed at the time of writing; in addition all the aspects of the analysis have not been completed yet.

For every test case, each tester filled in a template describing how they were able to express each constraint in the system. All these tables were combined and analysed. From this analysis it was summarised what percentage of the constraints could be expressed in the systems either 'fully' 'partially' and 'not'. The summary is done by grouping the numbers a) by constraints on one object; constraints for two objects and constraints for group of objects, b) per type of constraints (see classification introduced in section 4.1), c) per test case, and d) per system. Some observations can be made from this analysis. Note that this analysis only addresses how much of the constraints were *expressed* by the testers in the systems and not how the constraints were handled, i.e. the quality of the proposed solution is not addressed here. This will follow from the evaluation of generalised data. From the analysis of the testers' feedback that was carried out we can extract the following observations:

- About 50% of all the constraints could be expressed fully or partially in the systems. This seems not much, but it shows that systems have developed some automatic generalisation capabilities.
- The most supported constraints were those applying to a single object, i.e. functionality addressing constraints for two objects and for groups of objects is less available in commercial software. This was already highlighted during the OEEPE project: one of the main conclusions was the general lack of contextual generalisation capabilities in the commercial software. Contextual operations have since then started to appear in commercial systems.
- The number of constraints that could be expressed in the systems is dependent on the specific expertise area of a software system. For example more than 60% of all constraints on the building class could be expressed in Change/Push/Typify (CPT) while it shows less constraints expression facilities for other feature classes. This is because Change and Typify are specifically meant for building generalisation. Also more constraints defined for two objects could be expressed in CPT than in the other systems. This can be explained because Push is specifically meant for keeping minimal distance between objects.
- The number of constraints that could be expressed in the systems is test case dependent: CPT behaves better for IGNF and OSGB data set compared to the other two test cases. This could be explained by the fact that about 50% of the constraints in these two cases are defined for the building class. Another conclusion on the number of constraints expressed per test case in the systems is the overall low values for OSGB data set compared to the other test cases. In some systems even less than 25% of the OSGB constraints could be expressed in the systems. This might be because of the complexity of the (NMA specific) constraints influenced by the large scale transition from 1:1250 to 1:25K when compared to the other test cases. Examples are the constraints on slope hachures (to be introduced in the target scale) and the nine constraints addressing how buildings should be aggregated depending on the initial pattern. The complex and OSGB specific constraints on buildings also explain a high number of partially expressed constraints in CPT (instead of fully).
- The numbers of constraints that could be expressed in the systems is also dependent on the experience level of the tester with the software. For example we can observe a high variance in the number of constraints that were *not* expressed in the Clarity system among different testers. This can be explained because Clarity is an open system which requires some experience to be used optimally. Note that all systems were tested by both novice and expert testers.

From the explanation of testers for not being able to express certain constraints, we can also conclude that functionalities for parameterization are missing, that conflict detection tools are still lacking as in the OEEPE project (especially for contextual constraints) and that the software systems lack functionalities for defining sensible groups for the constraints that work on group of objects, such as ‘building blocks’. Current systems group these objects by the partitions built from linear features. This does not always yield the best solution since objects are mostly unevenly distributed among these partitions.

6. CONCLUDING REMARKS

This paper presented the EuroSDR project studying the state-of-the-art of automated generalisation, implemented in commercial software. Complete results on the output of the tests will be available after the evaluation on the generalised data has been finalised. However we can already elaborate on some insights obtained from the project so far.

Firstly, formalising map requirements as constraints was a good experience for NMAs. It was felt as a very time consuming exercise, but it gave insight into the importance of formalising generalisation needs to enable automation. The specification of map requirements as a set of constraints is an implementation of the generalisation theory on constraints. In that sense the final conclusion of the project will help to get more insight into some current limitations of constraint based techniques. According to (Harrie and Weibel, 2007) the main limitation of these techniques is the limitations of constraints themselves. Important cartographic constraints still remain to be defined, others are poorly defined. Although not complete, the list of constraints produced by this project is representative for NMA requirements and therefore the exercise of NMAs of specifying their requirements as constraints is a first step towards these limitations.

Having all the outputs of the tests ready, we realise that it will be impossible to analyse all results obtained from this extensive project; this would mean at least one year work for one person, whereas the personnel input in the current project is fully covered by the participating organisations. Instead the results will only focus on some main aspects. Since this is one of the first projects on automated generalisation on complete data sets taking the requirements of several NMAs into account, the project will mainly yield insights into the general level of automated generalisation possible in currently available generalisation systems as well as into the methodology of such a project. A future test can make use of the findings and the set up of this project.

The project team recognizes that care should be taken when publishing results of this (and similar) projects, because of biasness introduced at several stages. Constraints are firstly used to *direct* the automated generalisation process. However the level of constraint satisfaction is also used as an indicator of success of the generalisation in the *evaluation*. According to (Harrie and Weibel, 2007) this approach has some drawbacks. Since the constraints do not describe all the aspects of the generalised map, a constraint based approach to evaluation might give an overly optimistic view of the map quality. In addition the cause of a non-satisfied constraint can be dual: it may be because it was not unambiguously understood by testers or that the system was not capable of handling the constraint. In addition basing the evaluation of the output mainly on satisfaction level of constraints does not address sufficiently how constraints interact. Good results for one, specific constraint does not say anything about the quality of the overall solution because another constraint could have been violated to enable good results for this constraint. Also bad results for one, specific constraint might have been necessary in order to satisfy more important constraints (what also is done during interactive generalisation). Finally some constraints are not directly suitable for automated evaluation because they cannot be expressed in computer understandable format, examples are constraints addressing patterns, networks and spatial distributions. For the same reason these types of constraints might be not complete in the current project.

The project concentrates on out-of-the-box solutions. NMAs have specified a number of constraints, both generic and NMA specific. Generic requirements may be specifically suitable to be addressed by out–of-the-box solutions. However the first experiences show that customisation is required in order to meet the NMA versions of the generic constraints most optimally. Therefore the most realistic way for NMAs to address their requirements is to develop customised solutions extending the out-of-the-box software. This also implies a shift for NMAs from investing primarily in developing cartographic knowledge towards building expertise in (extending) cartographic and generalisation software. The possibilities for extending the software or integrating the software with other systems are an important aspect which is not addressed by this project.

ACKNOWLEDGEMENTS

We would like to express our gratitude to our colleagues who joined us in carrying out the tests. They are:
Magali Valdepérez (IGN, Spain), Patrick Revell, Stuart Thom, Sheng Zhou (Ordnance Survey), Willy Kock (ITC), Annemarie Dortland (Kadaster), Maarten Storm (formerly Kadaster, now with AQUAGIS), Patrick Taillandier (IGN, France).
In addition we would like to thank the vendors participating in this project for their very important contributions to this project.

REFERENCES

AGENT, 1997, Map generalisation by multi-agent technology, ESPRIT project 1997-2000, http://agent.ign.fr/

Barrault, M., N. Regnauld, C. Duchêne, K. Haire, C. Baeijs, Y. Demazeau, P. Hardy, W. Mackaness, A. Ruas & R. Weibel (2001), Integrating multi-agent, object-oriented, and algorithmic techniques for improved automated map generalization. In: Proceedings of the 20th International Cartographic Conference (ICC 2001), 6-10 August 2001, Beijing, China, pp. 2110-2116.
Bard, S., 2004, Quality Assessment of Cartographic Generalisation. Transaction in GIS, Vol. 8, No. 1, pp. 63-81.

Beard, M. K., 1991. Constraints on Rule Formation. In Buttenfield, B. P., and R. B. McMaster (eds), Map Generalization: Making Rules for Knowledge Representation, Longman Group, pp. 121-135.

Burghardt, D. and M. Neun, 2006. Automated sequencing of generalisation services based on collaborative filtering. In: M. Raubal, H. J. Miller, A. U. Frank and M. Goodchild (eds): Geographic Information Science. 4th Int. Conf., GIScience 2006, IfGIprints 28, pp. 41-46.

Burghardt, D., S. Schmidt and J.E. Stoter, 2007, Investigations on cartographic constraint formalisation, Workshop of ICA commission on generalisation and multiple representation, August 2-3, Moscow, Russia.

Burghardt, Dirk, Stefan Schmid, Cecile Duchêne, Jantien Stoter, Blanca Baella, Nicolas Regnauld, Guillaume Touya, 2008, Methodologies for the evaluation of generalised data derived with commercial available generalisation systems, 11th ICA Workshop on Generalisation and Multiple Representation, 20-21 June 2008, Montpellier

Galanda M., 2003, Modelling constraints for polygon generalization. Proceedings 5th ICA Workshop on Progress in Automated Map Generalization, Paris, France, CD-ROM.

Harrie, L., 2001, An Optimisation Approach to Cartographic Generalisation. Ph.D. thesis, Department of Technology and Society, Lund University.

Harrie, L. and R. Weibel, 2007, Modelling the Overall Process of Generalisation, Chapter 4 Generalisation of Geographic information: cartographic modelling and applications, edited by W.A. Mackaness, A. Ruas and L.T. Sarjakoski

Mackaness, W.A. and A. Ruas, 2007, Evaluation in Map Generalisation Process, Chapter 5 in Generalisation of Geographic information: cartographic modelling and applications, edited by W.A. Mackaness, A. Ruas and L.T. Sarjakoski

Muller, J.C. and P.J. Mouwes, 1990, Knowledge acquisition and representation for rule based map generalization: An example from the Netherlands, GIS/LIS proceedings 90, Anaheim, California, Vol1, pp.58-67

Ruas, A. and C. Plazanet, 1996. Strategies for Automated Generalization. Proceedings of the 7th Spatial Data Handling Symposium, Delft, the Netherlands, pp. 319-336.

Ruas, A., 1999. Modèle de généralisation de données géographiques à base de contraintes et d'autonomie. Doctoral Thesis, Université de Marne-la-Vallée.

Ruas, A., 2001, Automatic Generalisation project: Learning process from interactive generalisation, OEEPE, Report nr 39

Van Smaalen, J. and J.E. Stoter, 2008, Automatische generalisatie van TOP10NL naar TOP50NL haalbaar?, co-authors: N. Bakker and H. Uitermark, GEO-INFO (2), pp. 48-53

Sester, M., 2000, Generalization based on least-squares adjustment, the XIXth International Congress, Commission IV, International Archives of Photogrammetry and Remote Sensing, Amsterdam, The Netherlnads, 931-938

Stoter, J.E., 2005, Generalisation: the gap between research and practice, Proceedings of the 8th ICA workshop on generalisation and multiple representation, 7-8 July, 2005, A Coruña, Spain.10 p.

Topografische Dienst Kadaster, 2005, Generalisatievoorschriften TOP50vector, Handleiding, *Generalisation regulations TOP50vector, Handbook*

Weibel, R., and G. Dutton, 1998. Constraint-Based Automated Map Generalization. Proceedings of the 8th Spatial Data Handling Symposium, Vancouver, pp. 214-224.

Ware, J. M., C. B. Jones and N. Thomas, 2003. Automated Map Generalization with Multiple Operators: A Simulated Annealing Approach. International Journal of Geographical Information Science, Vol. 17, No. 8, pp. 743-769.