XULU - A generic JAVA-based platform to simulate land use and land cover change (LUCC)

Schmitz, M.[1]; Bode, Th.[1]; Thamm, H.-P.[2] & A.B. Cremers[1]

[1] Institute of Computer Science, Division III, University of Bonn, Germany
[2] Centre for Remote Sensing of Land Surface Application, University of Bonn, Germany
E-mail: hp.thamm@geographie.uni-bonn.de

Keywords: *LUCC, modelling, decision support systems, land use planning*

EXTENDED ABSTRACT

Land use and land cover (LUC) are key parameters within the hydrological cycle, the climate system and many other areas (e.g. food security, sustainable development). Today there is an increasing demand to obtain sound information about the future development of LUC. Climate modellers, hydrologists, landscape planners, decision makers need reliable scenarios for future development of LUC. Therefore different land use and land cover change models (LUCC) have been developed over the last 10 years. But most of them are difficult to use, not easy to extend or awkward to combine with other models.

Before that background, the ***eXtendable Unified Land Use and land cover modelling platform*** (XULU) was developed as a generic framework which can be used for several model types (e.g. statistic dynamic, agent based) in miscellaneous scopes of application (e.g. regional LUCC, urban growth). XULU is realised as a stand-alone JAVA application with a universal graphical user interface (GUI) which provides the general features needed for modelling: data storage, I/O methods, visualisation and model flow control. With these components realised once, there is no need for model developers to implement their own (especially complex GUI programming). With XULU it is possible to combine different models (e.g. agent based and statistical approaches). Because using the same data basis time-consuming data conversion is not necessary.

In XULU a model is implemented as a JAVA class. So the full power of this modern object-orientated programming language can be used for model implementations. To support model implementation for JAVA-unexercised users, XULU provides a little code generator which creates a source code frame of the necessary JAVA classes.
One of the main issues of XULU is the plug-in based concept. New data types, I/O methods and visualisation tools can be integrated without changing the core XULU application. At present there are plug-ins to maintain and visualize raster and vector data, but with new plug-ins it is also possible to use XULU for completely other approaches than LUCC (e.g. economical problems).

XULU was applied successfully for modelling LUCC in different scales in West Africa and Germany

The current work is to integrate a Geo-Toolbox into XULU – especially a statistical module –, which supports data pre-processing directly in XULU. To reach a good performance also for complex models and big scenarios the parallelisation of model processes to several computers is part of the current activity, as well as the coupling of various independent models.

.

1. INTRODUCTION

Over the last decades large changes of the land use and land cover (LUCC) have been observed in many areas of the world (Lambin & Geist, 2006, Geist, 2006, Gutman et. al, 2004). This LUCC has significant impacts on the hydrological cycle, the climate system, socio-economic conditions and many other areas such as food security, sustainable development. This causes a strong demand in science, planning, politics and decision making to compute scenarios of the future state of land use and land cover for different boundary conditions. As a consequence numerous models to compute LUCC have been developed in the last decades. A wide variety of methods, complexity and demand for input parameters can be found. A good overview about available LUCC models can be found in U.S. EPA (2000).

But most of the available models have limitations which are impedients for general application Hence the models are often not very user friendly, difficult to use and not well documented. Furthermore most of them are closed source, so it is not possible to control exactly how they produce the results ("black boxes"). As well it is not easy to extend or to combine them with other models (Briassoulis 2000). So specific problem orientated solutions can not be implemented quickly. Another problem is that the necessary input parameters have to be provided in complex file structures, where sometimes not even comment lines are allowed. So in general, the use of most of the up-to-date LUCC models, is very time consuming and demands highly skilled experts with extensive experience. This is an unsatisfactory situation and a big hindrance for the general use of LUCC models. In non-scientific environments like government and development agencies the problem is even greater.

2. GOALS

Firstly, some background on the eXdendable Unified Land Use modelling platform. It was developed as a generic framework to compute scenarios of the future state of land use and land cover under different boundary conditions. It is independent from the spatial scale of the test areas. So it is possible to compute changes of vegetation in a field due to plant diseases as well as the deforestation in West-Africa.

XULU should overcome most of the limitations of the models used up to now. It is user-friendly, well structured and flexible. From the beginning it was designed to be extendable. It should allow an integration of different types of LUCC or geographical models, even for persons with only limited programming skills. Even the coupling with other data bases and the use of XULU as decision support systems (DSS) ought to be possible. XULU should be open source and distributed widely, not only in the scientific community, where a lot of applications are possible, but in government departments, planning bureaus and developing agencies as well.

3. METHOD

When starting with a new (and often very simple) model idea, the model algorithm stands in the foreground of interest. Anticipating that the model idea could not succeed, spending time in complex GUI programming seems to be a waste of time in the first step of model realisation. But on the other hand: Without keeping model handling, usability and GUI in mind from the beginning of programming, in most cases it is very difficult or even impossible to realise it later.

XULU is developed to balance the trade-off between easy and fast implementation for model developers on the one hand and comfortable usability for model (end-) users on the other hand.

3.1. Separating model from application

The first major goal of XULU is to make the work of model developers easier. This is reached by separating the *model implementation*[1] from the generic *XULU framework*, which provides model independent the application functionalities that are usually the same for all models:

- data maintenance (so-called: *datapool*)
- I/O routines for data import and export
- data structures and internal memory management
- model flow control
- data visualisation

The XULU framework is implemented in JAVA 1.5 and provides the fore mentioned functionalities in a generic and model-independent GUI application, so there is no need for model developers to spend time on them. Several models can be plugged into the framework at the same time. Therefore it is possible to combine different models and model types. Because all models are based on the same data structures and use the same datapool, the output of one model can be used immediately as input for another model. Time-consuming data conversion is not necessary.

[1] short: *model*

In XULU a model is implemented as a JAVA class, so the full power of this modern object-orientated programming language and all its extensions can be used for model developing[2]. The model is integrated into the framework as a plug-in.

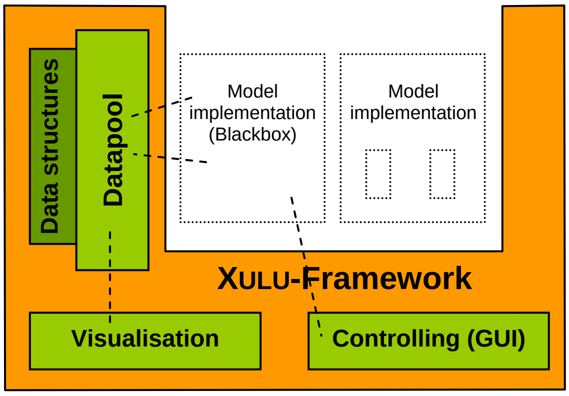


Figure 1: Separation between the model implementations and the XULU framework.

One benefit of this framework architecture is that the effort to implement a model is reduced essentially to the algorithm. And since the interface between the XULU framework and a model is designed very simply and reduced to a few functions the possible structure of the algorithm is not limited.

The technical split between framework and model, also has some consequences for the interaction between model and framework: One major aspect is the data and memory management remain completely within the framework. The model only defines the algorithm and has no influence on the type of data source (ASCII raster file, TIFF file, local database, WMS[3], …) or how the data are stored internally (completely in memory, dynamic reloading, …). Vice versa the framework only holds a data object without information about its sage in the model algorithm. Therefore XULU does not differ between model input, output and temporary data. The model simply proclaims, which type of data – so-called *model resources* – is needed for the modelling procedure (especially for storing the output!). The user "creates" suitable data objects in the datapool, which afterwards are allocated to the model resources during initialisation phase (see Figure 2). Hence switching between different scenarios becomes very easy, in particular because various data sources can be used. The GUI needed for this *resource mapping* is completely generated by the XULU framework. There is no need for the model developer to implement a model specific GUI![4]

As a second consequence of the strict separation between model and framework is that the responsibility of data control remains completely in the hands of the framework and the user. After the model starts, it runs practically autonomously. In particular, it is not the model algorithm which defines, whether or when something happens to the data (e.g. visualisation or storage), but the user. XULU's *event concept* only provides the possibility that the model initiates events on significant moments during the model algorithm procedure. These events allow the user to define *event handlers* which react automatically on model events during the run. Besides the freedom of choice the event concept is advantageous in terms of performance. For example, frequent data visualisation makes the model procedure slow and This is only necessary for evaluation, debugging or the presentation of cases. Often the user is only interested in a fast model run and the end result. XULU allows the user the possibility of switching between these conflicting cases on the front-end

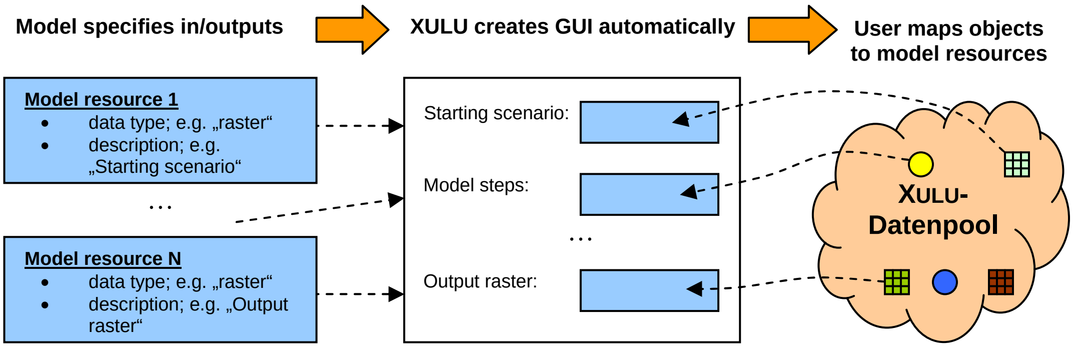

Model specifies in/outputs
XULU creates GUI automatically
User maps objects to model resources
Model resource 1
data type; e.g. „raster“
description; e.g. „Starting scenario“
…
Model resource N
data type; e.g. „raster“
description; e.g. „Output raster“
Starting scenario:
Model steps:
…
Output raster:
XULU-Datenpool

Figure 2: XULU encapsulates the model data in model resources, which are mapped to datapool objects by the user.

[2] XULU provides a little code generator to support JAVA-unversed developers; see 3.2

[3] WMS = Web Map Service

[4] An additional model specific GUI can also be added on, but it is not required.

level without changing the model algorithm.

3.2. Extensibility

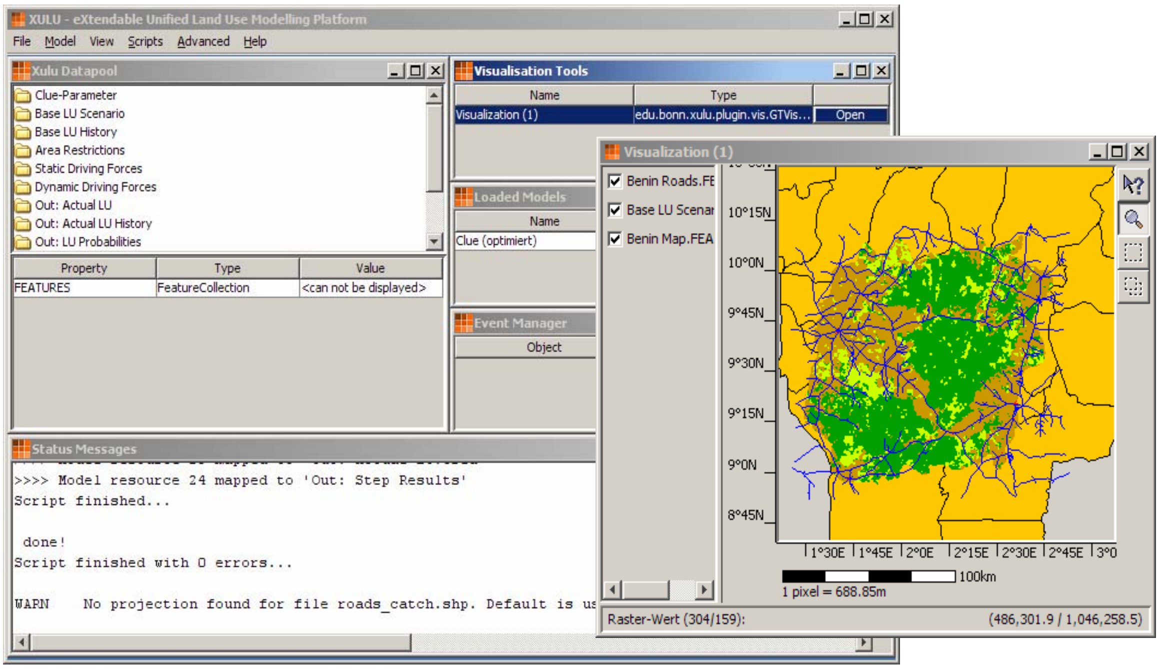
XULU follows the motto “Enhancements by extensions” in contrast to “Enhancements by changes”. So – as mentioned above – XULU is developed as an open and plug-in-based application. Besides the models, all main application functionalities (data types, I/O routines, visualisation tools) are integrated in the XULU framework as plug-ins. And because of the separation described in section 2.1 all extensions are immediately available to all models.

In case of data types and I/O routines XULU follows the factory concept (Gamma 1998, Brügge & Dutoit 2004): In short this means that complex data types (e.g. memory critical type Raster) are specified by JAVA interfaces and their internal structure and organisation (e.g. memory resistant, tile loaded, database) is decided by a loading and storing factory. All components working with a data type (e.g. datapool, models) only know the interface methods and not the internal structure. So it is possible to switch to a more efficient data type only by changing or replacing the factory. Afterwards all components use the improved data type automatically, while remaining completely unchanged.

The plug-in makes XULU very easy to extend and facilitates an independency from a specific model as well as a fixed field of application. Currently following plug-ins for LUCC modelling are realised: spatial data types for raster and vector data, I/O routines for shape files and different raster types (e.g. ASCII and GeoTIFF) and a layer-based visualisation for raster and vector maps (see Figure 3). The latter is based on the free JAVA library GEOTOOLS (http://www.geotools.org). With some new plug-ins XULU can also be used in non-geographical fields (eg. growth of bacteria in Petri dishes).

Programming JAVA classes for new model approaches can be difficult for inexperienced users. Therefore XULU provides a model code generator. By a GUI the user specifies the model resources and the variable names, which should be used in the source code. With this information, the code generator creates a compliable source code frame of the model class. The user only has to implement the model algorithm part manually. In future works the code generator should be extended to create also parts of the model algorithm, which can be specified interactively by the user.

Besides the code generator, XULU supports the model implementation phase by another useful function. In contrast to the common (model) applications, in XULU it is possible, that changes of the model source code become effective during XULU runtime. After compiling, the model class can be reloaded in XULU. Because it is not necessary to restart the XULU application, reload data and reconfigure the model, this feature saves



Clue-Parameter
Base LU Scenario
Base LU History
Area Restrictions
Static Driving Forces
Dynamic Driving Forces
Out: Actual LU
Out: Actual LU History
Out: LU Probabilities

Property	Type	Value
FEATURES	FeatureCollection	<can not be displayed>

Name	Type	
Visualization (1)	edu.bonn.xulu.plugin.vis.GTVis...	Open

Name
Clue (optimiert)

Object

```
>>>> Model resource 24 mapped to 'Out: Step Results'
Script finished...

 done!
Script finished with 0 errors...

WARN    No projection found for file roads_catch.shp. Default is us
```

☑ Benin Roads.FE
☑ Base LU Scenar
☑ Benin Map.FEA

Raster-Wert (304/159): (486,301.9 / 1,046,258.5)

Figure 3. The main window of the *XULU modelling platform* and some visualised geo data much time during the model evaluation and calibration phase.

3.3. Working with XULU

As shown in **Figure 3**, XULU is a stand-alone JAVA application with a user-friendly and clearly arranged GUI. The user has to load the necessary data object into the XULU datapool and allocate them to the individual model resources. This gives the user the freedom to choose, which data are used for modelling. Also switching between different scenarios is very comfortable. The user only has to change a resource entry in the XULU-generated model GUI. The GUI automatically offers only the appropriate objects from the datapool. There is no need to restart XULU to start a new model run. Nevertheless according to the quantity of required model data, the loading and resource mapping procedure can be very time-consuming for the user. At this point XULU supports the user with scripting functionality. Often required and time-consuming queues of data loadings and resource mappings can be defined in script files which can be run by mouse click.

Even though it is a modelling framework and not a GIS application, XULU also provides some features for data pre-processing. With the plug-in raster calculator arithmetical and boolean rules (+, -, *, /, if then else, AND, OR, and other) can be evaluated on (and between) raster data from the datapool. Also the use of filter matrices is possible. The raster calculator is designed for the interactive use in the XULU framework, as well for the internal use within a model algorithm.

4. USING XULU TO MODEL LAND USE LAND COVER CHANGE IN WEST AFRICA

In IMPETUS (Integrative Management Project for an Sustainable and Efficient Use of Fresh Water), a big interdisciplinary project of the Universities Cologne and Bonn, which investigates the change in the hydrological cycle in West Africa and the therewith related changes in the ecosystem and the socio-economic conditions IMPETUS (2003), XULU was used to compute the future land use and land cover for different scenarios with their specific boundary conditions. The area of investigation is the catchment of the Upper Ouémé, which is located in the north west of Benin (latitude of the catchment from lat 8° 58’ N and 10° 11’ N, lon 1° 30’ E and 2° 47’ E) and covers an estimated area of 100 km x 100 km. The vegetation belongs to the transition between the southern Guniee and northern Sudanese one (Adjanohoun 1967) and is characterised by tree and scrub savannah with a high grassy understorey. Some patches of dense forest are found at special locations (soil, water availability). The population density is comparatively low with around 19 persons per square kilometre but strong immigration has been recorded (INSAE 2003). The main source of income is small-scale subsistence agriculture. The area has been subject to large changes in the land use and land cover over the last decade, caused by agricultural colonisation and, mostly illegal, logging (Doevenspeck, 2005, Götze et al., 2005). Using LANDSAT images the extension of agricultural land and other LUC changes between 1991 and 2000 was derived using a decision tree based approach (Friedl & Brodley 1997).

To model future states of land use and land cover a statistical dynamic model was used – the CLUE-S model of Verburg (2002). It was the most suitable approach for addressing the problem regarding the processes causing the changes in the area. Additionally the available input data had been tested successfully in different tropical environments (Verburg et al. 2004, Verburg et al. 2004a). The CLUE model algorithm was implemented as a plug-in into XULU. It turned out that with only six parameters (initial land use, usability of the soil for agriculture, population density, distance to roads, distance to water, protected natural reserves), the LUCC can be explained with an overall accuracy of 83 % (Thamm et al 2005b). Based on these analyses, probability functions for changes for every land use class were set up. Following the IMPETUS guidelines three scenarios where defined (“business as usual”, “economic growth and stability”, “declining economic and weak institutions”). For every scenario the demand for new agricultural land was defined. The scenario “business as usual” was based on the outcome of a representative survey and the projections of future population (Doevenspeck, 2005) the other scenarios were defined in a quite complex process in collaboration with experts of other faculties (eg. Economics, Agriculture, Climatology). Detailed description of the model set up and outcomes can be found in (Thamm et al. 2005a, Thamm et al 2007). The results were in even in the business as usual scenario in the year 2025 nearly all forests and tree savannah in the area of investigation will be converted into farmland. The results also identified ‘hot spots’ of LUCC where a shortage of arable land will occur. This allows the introduction of more adequate measures such as more efficient agricultural techniques to prevent distribution conflicts.

During the modelling it turns out that XULU is very handy to use. The implementation of the different scenarios can be performed quickly. Other advantages are the integrated views, which inform about the state of the modelling and have the ability to display the result or other parameters at every iteration step. Therefore good control of the whole modelling is possible. At the moment XULU is used to model LUCC at a local scale for different hot spots and at the sub-national scale for the whole catchment of the Ouémé.

5. FUTURE WORK

Due to the flexibility and the convincing modular concept, XULU can be used for various tasks within spatial modelling. At the moment a fire model for West-African savannahs is being developed and tested successfully. The analysis results of satellite data with a high resolution in time are used as input data in addition to non variable data like topography and infrastructure. The wind speed and wind direction as well as the variability of the both parameters can be adjusted. These first results are very promising. Another planned application is the modelling of the spread of crop diseases at the field level in collaboration with the department of agriculture in a project for precision farming.

To improve the speed of the modelling and to enable XULU to be used on large areas with small pixel size a version for parallel computing was created (“XULU/V”, Appl 2007). The user just has to install a small program on every computer which is available and XULU/V distributes the computing tasks. Tests have proven that the computing speed for bigger test areas can be reduced significantly.

As a next development step it is planned to integrate the statistical software R. This will assist in the time consuming parameterisation of the models.

6. CONCLUSION

XULU has proven to be a very useful tool for different tasks in the spatial modelling especially to compute future state of land use and land cover. The big advantage is the modular concept which enables the scientist to concentrate on the research question because he does not has to take care of the overhead of model development. This opens as well interesting new options like combining different types of models.

The realisation in JAVA enables easy integration of databases or automatic processing chains. Another benefit is the good usability. Computing of different scenarios can be done quickly. As well the control of all parameters and preliminary results is very helpful.

Re-iterating it can be stated that XULU is an interesting, very flexible and user friendly system for people and organisations which are interested in spatial modelling.

7. ACKNOWLEDGEMENT

This work was supported by the Federal German Ministry of Education and Research (BMBF) under grant No. 01 LW 06001B and by the Ministry of Innovation, Science, Research and Technology (MIWFT) of the federal state of Northrhine-Westfalia under grant No. 313-21200200.

8. REFERENCES

Appl, D. (2007): Diploma thesis “XULU / V – Eine Erweiterung der XULU-Plattform zur Unterstützung verteilter Simulationen in der Landnutzungsmodellierung”; University of Bonn, Germany.

Adjanohoun, E. & L. Ake Assi (1967): Inventaire floristique des forêts claires subsoudanaises et soudanaises en Côte d’Ivoire septentrionale ; in : Ann. Univ. Abijan, Fasc. Sc., Vol. 3, S. 89-147

Briassoulis (2000): Analysis of Land Use Change: Theoretical and Modelling Approaches. In: The Web Book of Regional Science (www.rri.wvu.edu/regscweb.htm), Regional Research Institute, West Virginia University.

Brügge, B. & A. H. Dutoit (2004): Object-Oriented Software Engineering: Using UML, Patterns and Java; Prentice Hall International.

Doevenspeck, M. (2004): Migration im ländlichen Benin – Sozialgeographische Untersuchungen an einer afrikanischen Frontier. Studien zur Geographischen Entwicklungsforschung, Saarbrücken, p. 282.

Friedl, M.A. & C.E. Brodley (1997): Decision tree classification of land cover from remotely sensed data. In: Remote Sensing of Environment 61; S. 399-409.

Gamma, E., Helm, R., Johnson, R. & J. Vlissides (1998): Design Patterns – Elements of Reusable Object-Oriented Software; Addisson-Wesley.

Lambin, E.F. & H.J. Geist (2006): Land-Use and Land-Cover Change, Local Processes and Global Impacts, Global Change, The IGBP Series, Berlin, Heidelberg, New York

Geist, H.J. (Ed., 2006): Our Earth changing land: An encyclopaedia of land of land use and land cover change, vol.2 (L-Z), Westport, London.

Gutman, G.; Janetos, A.C.; Justice, C.O., Moran, E.F.; Mustard, J.F.; Rindfuss, R.R., Skole, D.; Turner II, B.L. & M.A. Cochrane (2004): Land Change Science, Observing, Monitoring and Understanding Trajectories of Change on the Earth's Surface. Dordrecht, Norwell.

GEOTOOLS Project – An open source GIS toolkit; http://www.geotools.org.

IMPETUS (2003): IMPETUS West-Africa. Final Report 01.05.2000-30.04.2003. Cologne.

INSAE (2003): Troisièmere récensement général de la population et de l'habitation. Cotonou, Bénin.

Götze, D.; Hennenberg, K. J.; Koulibaly, A.; Orthmann, B.; Porembski, S. & Traor, D. (2005): Investigations to maintain tropical biodiversity under the processes of land-use intensification and climate change in West Africa. Annales de Botanique de l'Afrique de l'Ouest, in press.

Schmitz, M. (2005), Diploma thesis "XULU – Entwicklung einer generischen Plattform zur Implementierung von Simulationsmodellen am Beispiel der Landnutzungsmodellierung".

Thamm, H.-P., Bode, Th., Schmitz, M. and A.B. Cremers (2007), XULU – Eine JAVA-basierte Plattform zur Implementierung von Simulationsmodellen. In Strobl, J. Blaschke, T. & G. Griesebner (ed): Angewandte Geoinformatik. Beiträge zum 18 AGIT-Symposium, Heidelberg. (In Press)

Thamm, H.-P., Judex, M. & G. Menz (2005a): Modelling of Land-Use and Land-Cover Change (LUCC) in Western Africa using Remote Sensing. Zeitschrift für Photogrammetrie und Fernerkundung 3/2005. 191- 199.

Thamm, H.-P., Judex, M. & G. Menz (2007): Assessment of future States of Land Use and Land in a regional scale with XULU as information base for decision making. (submitted)

Thamm, H.P.; Schütz, O. & Christoph, M. ed. (2005b): IMPETUS Atlas - Benin - Research Results, Remote Sensing Research Group, Bonn

U.S. EPA (2000): Projecting Land-Use Change: A Summary of Models for Assessing the Effects of Community Growth and Change on Land-Use Patterns. EPA/600/R-00/098. U.S. Environmental Protection Agency, Office of Research and Development, Cincinnati, OH. 260 pp.

Verburg, P. H.; Soepboer, W.; Veldkamp, A.; Limpiada, R.; Espaldon, V. & S. S. A. Mastura (2002): Modeling the Spatial Dynamics of Regional Land Use: The CLUE-S Model. In: Environmental Management 30 (3) S. 391–405.

Verburg, P.H.; Overmars, K.P.; & N. Witte (2004): Accessibility and land use patterns at the forest fringe in the North-eastern part of the Philippines. In: The Geographical Journal 170 (3): 238-255

Verburg, P.H.; Veldkamp, A., Engelsman, W., van Zalinge, R., van Mensvoort, M.E.F., & K.P Overmars (2004a): The use of models to assess the impact of land use change on ecological processes: case-studies of deforestation in SE Asia. In: Gerold, G., Fremerey, M., and Guhardja, E. (Hrsg.): Land use, nature conservation, and the stability of rainforest margins in Southeast Asia. S. 475-494 (Springer-Verlag) Berlin.