



Object Oriented Visualization of Urban Energy Consumption

Kjems, Erik; Bodum, Lars

Published in:

Proceedings of 11th conference on Computers in Urban Planning and Urban Management

Publication date:

2009

Document Version

Publisher's PDF, also known as Version of record

[Link to publication from Aalborg University](#)

Citation for published version (APA):

Kjems, E., & Bodum, L. (2009). Object Oriented Visualization of Urban Energy Consumption. In Y. Anthony G. O., & F. Zhang (Eds.), *Proceedings of 11th conference on Computers in Urban Planning and Urban Management* ESRI Press.

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal -

Take down policy

If you believe that this document breaches copyright please contact us at vbn@aub.aau.dk providing details, and we will remove access to the work immediately and investigate your claim.

Object Oriented Visualization of Urban Energy Consumption

Erik KJEMS
Associated Professor
Centre for 3D GeoInformation
Aalborg University
Fibigerstraede 11
DK-9220 Aalborg
Denmark
Tel: +45 99408079
Fax: +45 98156541
<http://www.3dgi.dk>
kjems@3dgi.dk

Lars BODUM
Associated Professor
Centre for 3D GeoInformation
Aalborg University
Fibigerstraede 11
DK-9220 Aalborg
Denmark
Tel: +45 99408078
Fax: +45 98156541
<http://people.plan.aau.dk/~lbo>
lbo@3dgi.dk

Abstract: The scope of the project presented in this paper implies to develop a geographically based data model making different kinds of relevant information available through the use of a 3D city model creating a new innovative type of spatial computer-human interface. Our main goal in the project will be to combine an object oriented 3D city model with an energy consumption model called EnergyPLAN (Lund and Münster, 2003). The main motivation behind it is to make a serious change of behaviour within the whole community and make Frederikshavn an example city for other municipalities to follow. The idea of the project is to show how different scenarios from a numeric model produced by EnergyPLAN can be visualized through a 3D city model providing an interactive user interface. This could e.g. be the possibility of adding wind turbines to the model, moving people from cars to bikes, increasing the number of solar collectors on the roofs or letting local transport being carried out with electrical or hybrid cars. This will make an important contribution to the decisions that have to be taken to fulfil the goals of the overall project of Energy City Frederikshavn. The combined model will also become generally available through a web service where each individual or company has the possibility of seeing how a high share of renewable energy can be achieved.

Keywords: 3D-GIS, Object Oriented, Managed Object, Interaction, Energy

1. INTRODUCTION

Following the ideas and thoughts of Digital Earth (Gore, 1998), many different solutions for Digital Globes have emerged within the last 10 years (Butler, 2006). They have brought a wide range of geo-related applications for the mass market through the Internet (Goodchild, 2008). One of the most visual achievements in that respect is 3D city models. Unfortunately, most models have only been used sporadically and mostly for the purpose of visualizing planning propositions. This is of course an excellent purpose for 3D virtual models, but the costs and effort creating them definitely demands more. We have seen a lot of different approaches regarding using 3D city models. Sound issues for instance (Stoter, 2007) or pollution related visualizations in urban areas due to the use of volumetric rendering (Lin, 2008) just to mention two recent ones.

One of the important answers to the challenge of for instance the global climate change is local responsibility in energy consumption and consumer behaviour. The “Centre for 3D GeoInformation” has been asked to participate in a project where the municipality of Frederikshavn, the most northern municipality in Denmark, has committed itself and its inhabitants to reduce the use of carbon based energy sources and eventually become self-sufficient within energy consumption before the year 2015. The geographic project area is the city of Frederikshavn, approximately 37 km². The goal is to change the source of energy to 100% sustainable energy sources.

It is more or less obvious that such kind of spatial representation in a 3D city model does not fit into common solutions widely used for visualizations. In the Centre for 3D GeoInformation we have been working on a new platform called GRIFINOR (GRIFIN today) (Bodum et al. 2005) since 2001, which among other things is characterized by handling all objects in the model as code. The objects do therefore not only contain geometry and attribute data (methods), but also behaviour (functions). The communication between and handling of these enhanced objects is made possible by means of the concept introduced in Java where objects are created in byte-code and subsequently executed with Java Virtual Machine. This principle is here referred to as Managed Objects (MO) (Kjems et al., 2009). The main advantage of MOs is the ability to hold different representations and at the same time be dynamic due to the executable byte-code. However, the real challenge in the project will be to develop the system in such a way that the users can understand and interact with the output through a web-based interface.

This paper is divided into several not necessarily coherent chapters but into important parts which combined serve the overall goal of the project. To give the reader an understanding of the basic concept used in this approach the concept of MOs is explained beforehand. After that a short introduction to the energy consumption model is given. First some words about the city model used for the visualization and the means of user interaction. Finally, the overall approach is presented although the development has not yet been finalized at the present time.

2. CONCEPTUAL UNDERSTANDING OF MANAGED OBJECTS

This chapter will try to explain the basic setup and concepts of the overall system. One could argue that the technology is not of real importance when it comes to visualization, but in this project the technology used is a focal point. Though the approach of managed objects has been used widely in different fields of computer science, it has never before been used within a geographical information context. We have seen and see a lot of object oriented approaches for GIS, but using MOs is a completely new way of handling objects. Apparently, this concept seems hard to understand for most people for which reason the following will try to give an in-depth explanation of it. MOs are truly revolutionary in their behaviour and outline some great possibilities for 3D geoinformation systems in the future.

Geographical information systems (GIS) have developed since the sixties and are nowadays an indispensable tool for all kind of public administrations, real estate companies, marketing analyses and a broad range of other geographically related domains with a need for analytical tools and geographically related data storage.

These tools were developed for the two-dimensional representation of the world, also called maps, even though they are digitally presented. We have recently seen a new approach where the free available software tool from Google called Google Earth provides one with a new globe based platform and a rather spatial interface presenting mountains and 3D buildings in a more realistic 3D perspective. Nevertheless, even this approach is from a data modelling point of view not far away from the original two-dimensional and can scarcely be named 2.5D. The approach in this project does not only take its origin in a true object oriented three-dimensional representation of the world, but also enhances the objects with functionality. The geometry is merely the visual representation while the functionality gives the objects the possibility of behaviour. This opens a completely new paradigm on how to handle and to use these new objects. Relational databases become merely a pure data container and provider of information while other inputs can come from other systems like sensor data and other applications.

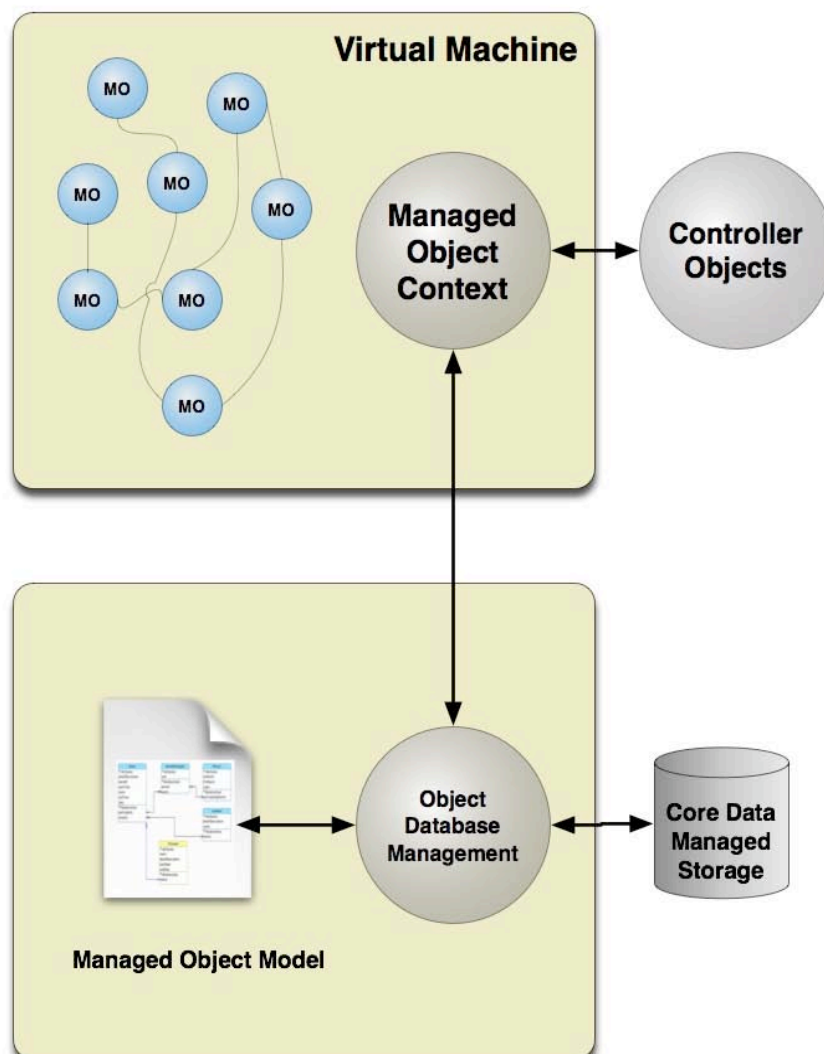


Figure 1 Conceptual structure of an application using managed objects.

Figure 1 shows an example of a MO scheme where an object database management system is exchanging objects with the managed object model and is also exchanging data to a core data managed storage (which could be any kind of database, also non-object based). The virtual machine executes the managed objects in relation to the managed object context. The execution may be a simple appearance of the object in the viewer or more complicated behaviour defined within the object. To ensure the state of the objects or eventually influence the behaviour there is a group of controller objects related to the managed object model as well.

To understand this better let us look at an example taken from the project. The whole basic idea with for instance GRIFIN was to model the real world. So each object in the model is representing a feature from the real world. The details and the extent of the object's complexity depend on the usage and the application itself. So this is definitely not only about representing the real world in a nice 3D model, rendering some nice pictures or navigating through the model. Well, it could be, but it could also be so much more. Since it is possible to store a lot more than only geometry in managed objects it would be a pity not to take advantage of that. Each MO can come "alive". In our case we are dealing with energy production and consumption in a city. So we have for instance attached solar collectors on the roofs of many houses represented by MOs. These solar collectors, understood as objects, retrieve data from a clock object informing what day of the year it is; eventually, a weather object provides data about the weather and the cloud cover obstructing the sun. The object itself calculates on the basis of this information and the information it holds about the solar collector, like efficiency, area etc., how much electricity it will produce today, in a month, or in a year; or a real sensor simply communicates to the solar collector object in our model how much it produces right now. And the result is shown in our 3D model as for instance a changing texture colour on the solar collector. One could argue that you will not need special objects for that functionality. Of course it is possible to build a system on a standard 3D model. But then again a MO can be moved to another model as long as it is a model controlled similar with a virtual machine executing these objects, they can be moved around freely. All object data in such a system are stored in a database and can be updated and changed as often as necessary. Other kinds of objects can be added and still executed in the same way. Some huge wind turbines are placed in the ocean but still close enough to the coast and to the city so that they are visible in the model. These wind turbines could provide a lot of data to our model making the presentation of them very real, for instance the wind direction and the speed of the wings circulating measured directly on the wind turbine. These data could easily be used by the MOs representing the wind turbines ensuring that the animation of the wind turbines is presented accordingly to the actual situation.

In one way using MOs always will be in connection with some specific application, where objects are designed for a specific purpose. But since this concept using a virtual machine either by Java or similar solutions as .NET applications is consistent and more or less standardized, it is a manageable job to do. In our case we have developed GRIFIN as a basic system, which is specifically created for real world topics containing coordinate referencing issues, terrain handling and more.

Managed objects have to be treated different from traditional objects since they are not only represented by geometry but actually by executable codes. To be able to access MOs fast and to be able to take advantage of objects being stored in several

distributed databases, the storage and indexing of the objects have to be especially designed. As in a pure object oriented solution, all the contents in the system are stored and accessed in the form of objects, in the case of GRIFIN as Java objects. This means that features represented by objects may, when convenient, carry only attribute data, be associated with a certain functionality (behaviours), or merely trigger a certain functionality while carrying a minimum amount of data or even no content data at all.

It should also be stressed that the objects in GRIFIN are organized or rather indexed for access through their position in geocentric coordinates. It is very essential for the system design to use a geocentric coordinate system, since this enables GRIFIN to handle globally related data without any kind of map projections, which have spatial inconsistencies on a global level. Most model maps are used for local areas like a city or minor landscape areas. For larger areas, the definition of datum and map projections can quickly become a challenge (Kjems, 2005).

Objects that can change contents or even shape will need some kind of calibration. They will also need some kind of state control mechanism, which for instance can identify the state of origin and trace changes. Working with dynamic objects is demanding, and a whole new set of rules has to be set up. Even though the idea of one common model map is reasonable, and the GRIFIN system, which has implemented the concept of MOs, is designed with the goal to cover the whole world in one model, it might be difficult to apply one set of rules for one common model. But since each feature can be designed with its own set of rules, references and characteristics, it is not necessary to create one set of rules, which applies to all features but rather a few rules, which serve common functionality. Furthermore, features can even be the same in terms of real world objects and still comply with different rules. This is a breach of the traditional understanding of applying standards, where features are handled and described completely the same way as objects in the model.

One of the more surprising differences between MOs and traditional objects is that MOs do not use any kind of standards implemented beforehand in the system that is MOs imply some kind of standard due to the coding language used. But no semantic based standard. Normally one always has to deal with file formats and data structures to secure input and output to the system. There are some elaborated standards like CityGML, KML or IFC (the latter mostly relevant for buildings) just to mention some of the more essential ones for model maps. MOs do not really depend on any kind of standard. As long as the objects can be identified by, in our case, Java virtual machine, it doesn't really matter what the different parts are called. Of course it is important to identify data in the model, but this functionality is not dependent on any kind of standard, except for the standard format of Java byte-code. This means that the system will be able to show the user what is pointed at the moment when it is asked to do so. It really doesn't care about it until then, because the way the viewer is presenting the object is independent of the standard applied in the data. The standard scheme will be presented along with the data. This means that it is possible to implement any kind of standard through the use of managed objects. The real beauty of it is that MOs always can handle the newest standards, which are used for importing data as long as we can use Java byte-code to model the corresponding MOs.

Working with MOs is a huge challenge conceptually and also from an implementation point of view. The possibility of adding functionality to objects that can be controlled by the objects themselves or other objects within or outside the model is enhancing the traditional way of thinking and using 3D models way ahead of our imagination.

3. THE CITY MODEL USED IN THE PROJECT

Most of the 3D city models produced are unfortunately dumb models made only for the purpose of visualizing planned projects and supporting the decision-making or simply for sales literature. Well, these are certainly excellent purposes but the costs of city models do not justify the investment for these purposes alone. The produced models with identified building objects should prepare the ground for a more long-term use and reuse for instance in a public administration. The building industry has now widely agreed on a common open standard for information interchange or data exchange called IFC (Industry Foundation Classes). This standard is based on an object oriented data model. When it comes to city models the further use of the models depends on the level of abstraction that has been used or rather demanded. In the community of 3D Geoinformation the realism is very often related to the way the spatial world is represented, while the 2D GIS community is focusing on attribute data describing additional states or characteristics of a feature. The focus in 3D Geoinformation has most of the time been on the representation of spatial objects, on relations between them like topology, ontology, storing them and presenting them with more or less details. The new approach is to focus on objects that will not only contain geometry and its attribute data but also contain behaviour, processes and ways of communication. Enabling this functionality in for instance a city model, a completely new virtual world, will open up an almost endless numbers of possibilities.

Nevertheless, the city model used in this project is quite traditional though the implementation into GRIFIN is different from other systems. The model was delivered from the company Scankort, which actually is only one of several competitive companies producing excellent 3D city models in Denmark. Unfortunately, the first delivery collected the buildings in huge blocks containing several buildings and unusable for MOs, since each house and subelement of a house, for instance the roof, must be identifiable as a separate geometry object. This is caused by the demand not only showing geometry but also enhancing the single house with behaviour and functionality. As soon as the model is delivered with identifiable objects and clean geometry the GRIFIN system will produce MOs from these data, which can be enhanced further in the system. This has one big advantage, an advantage which every person who has ever been working with digital data should recognize as absolutely unique. One of the biggest problems in the area of digital data no matter if we talk about maps, buildings or whatever is that digital data very soon get inaccurate due to changes over time, changes which not automatically and often only with great effort are communicated to the model. In the case of 3D models this is definitely the most common case since nicely produced 3D models more or less represent a time stamp of a scenario rather than a data set for instance known from database systems. Before GRIFIN shows the city model it reads the model data from a database following the creation of MOs. This means that if the data set representing the city model is updated frequently, GRIFIN always takes its origin in the newest set of data. This makes sense if you want to use 3D data for more than visualizations. The great strength of 2D GIS systems is the possibility of relating data

to updated map productions while 3D representations until now were used in connection with smaller areas detached from the overall map coverage. The reasons for that are many and do not really matter anymore. But if 3D GIS systems are supposed to have a future for instance in connection with location-based services, a functionality with the one implemented in GRIFIN is quite essential.

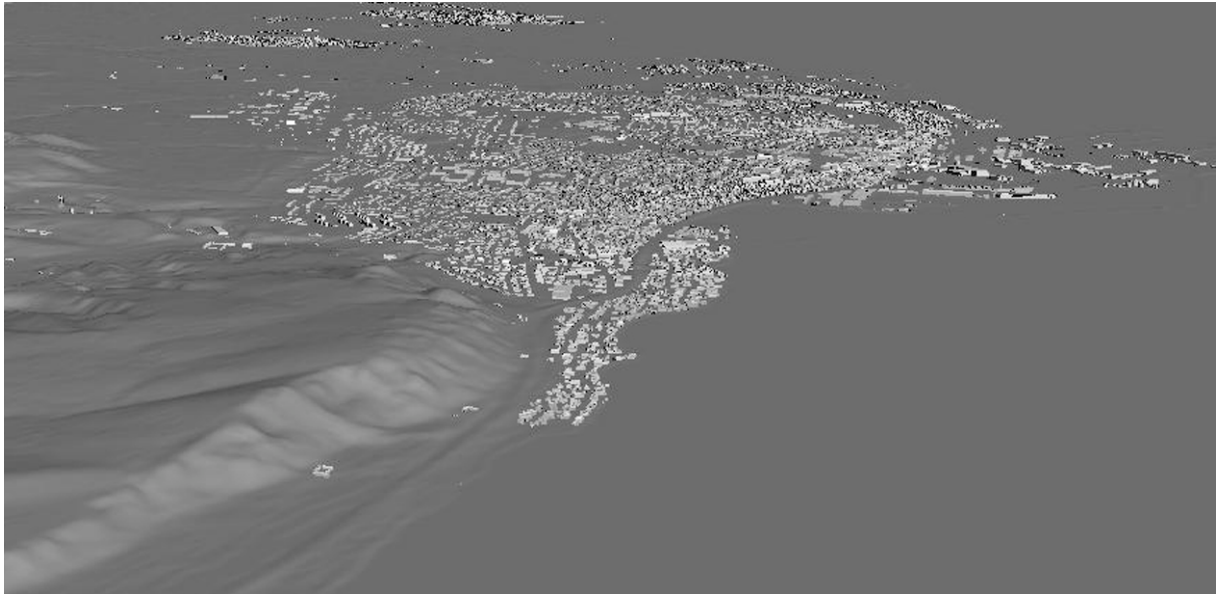


Figure 2 Birdseye view of the entire city model of Frederikshavn.

The shown city model is quite large. Also this is a reason why a database system is the right solution for city models. Traditionally, 3D models tend to get large and heavy and difficult to navigate in. Flight simulators and other systems that use large areas including terrain and buildings did discover this many years ago. But city models are still primarily made in large one file modelling applications.

4. THE ENERGY CONSUMPTION OF FREDERIKSHAVN CITY

The main functionality for the MOs in GRIFIN is provided through an energy calculating system (EnergyPLAN), which interacts with the city model through simple text file queries. If we add some square metres solar collectors or add another wind turbine to the system a new query and resulting calculation is changing the model in a certain way. A new wind turbine for instance first of all will add another model of a wind turbine to the city model but will also change the share of the renewable energy to the overall energy consumption, which is the main goal of this project.

Renewable energy is here defined as arising from natural resources such as sunlight, wind, rain, waves, tides and geothermal heat, which are naturally replenished within a time span of a few years. Renewable energy includes the technologies, which convert natural resources into useful energy services.

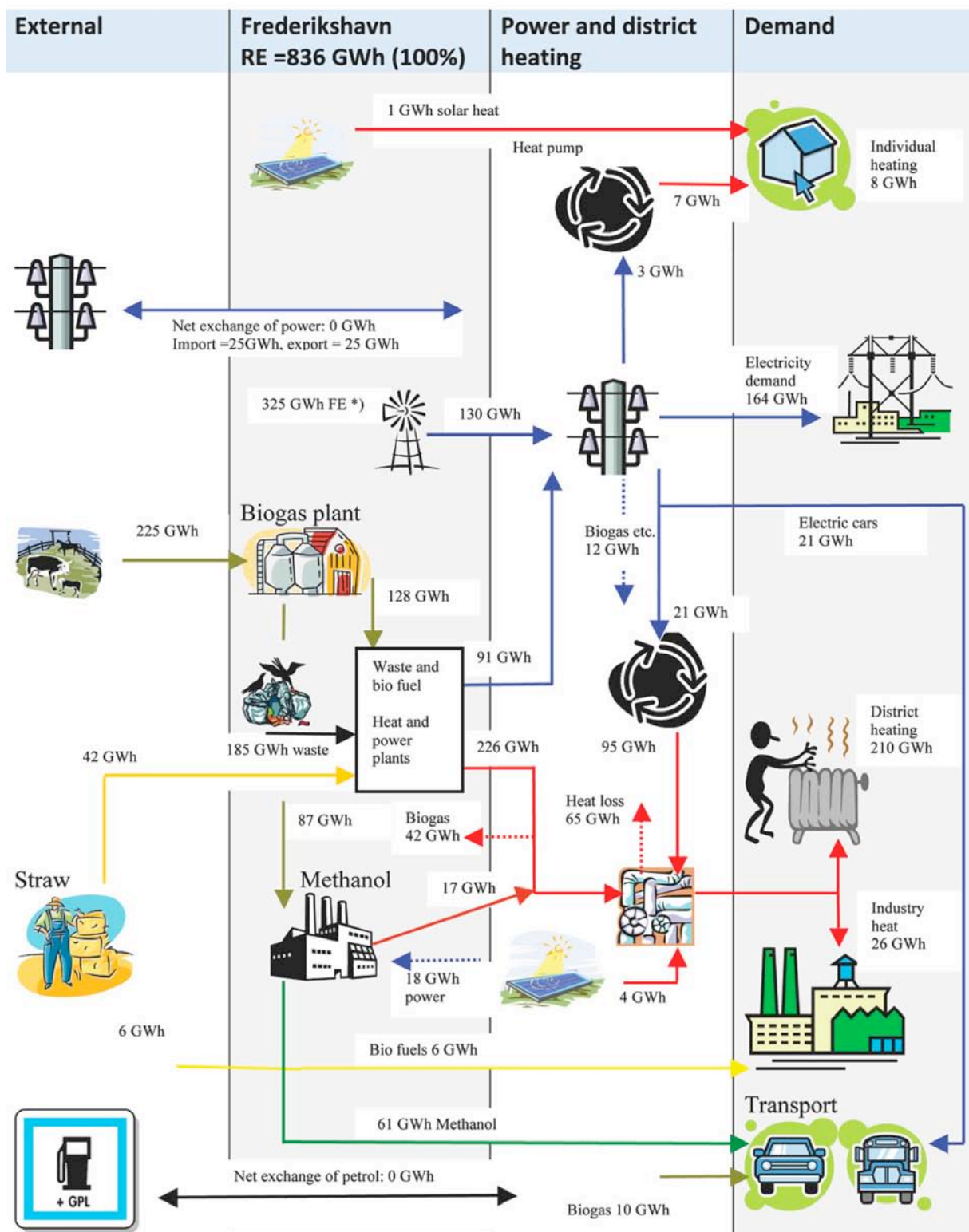


Figure 3 Frederikshavn in 2015 using 100% renewable energy (Clark, 2009).

The EnergyPLAN model illustrated schematically by its components in figure 3 for a certain scenario of Frederikshavn is a deterministic input/output model. General inputs are demands, renewable energy sources, energy station capacities, costs, and a number of optional different regulation strategies emphasizing import/export and excess electricity production. Outputs are energy balances and resulting annual productions, fuel consumption, import/exports of electricity, and total costs including income from the exchange of electricity. The model can be used to calculate the consequences of operating a given energy system in such a way that it meets the set of energy demands of a given year. Different operation strategies can be analyzed. Basically, the model distinguishes between technical regulation, i.e. identifying the least fuel-consuming solution, and market-economic regulation, i.e. identifying the consequences of operating each station on an electricity market with the aim of optimizing the business-economic profit. In both situations most technologies can be actively involved in the regulation. And in both situations the total costs of the systems can be calculated (Clark 2009).

The output of the EnergyPLAN defines the framework for GRIFIN to visualize. Constraints in EnergyPLAN are limitations for the visualization. For instance the greatest drawback of the system is that it is not possible to calculate the energy consumption on an individual household level. Figure 3 therefore also shows the input and partly output limitations or, if you like, the limitations of what to visualize.

The visualization assignment in the project is divided into two parts, an interactive user interface, which must be designed from scratch, and an output according to the cities' energy consumption. The interface will mostly contain some sliders where one can control the share of external power sources; solar collectors; time of day or day of year; or the share of local public transport using electrical or hybrid motive power to mention some. The big challenge however is to visualize an intuitively understandable output in relation to a whole city model as shown in figure 2. The dynamic behaviour possible in the managed objects makes it technologically a manageable job, but the overall visualization part regarding the city model is quite a challenge. Works until now have been focusing on texturing the single building or showing other kinds of graphical illustrations (Kuroda 2001). Also Google Earth has been used to show energy consumption in a larger scale. See the example provided by Gerardo Paz in figure 4 below (<http://tinyurl.com/bdfcg9>). Our situation is different and we have to come with something new and are presently working on it.

At the present time the final solutions have not yet been decided on. The challenge here is to ask the user what kind of visualization the user wants, knowing the user has no clue what he or she wants or might want. Presenting good ideas or examples will therefore be essential for a good outcome. For the same reason a mock-up of the city model has been created. This mock-up will be used to illustrate possible scenarios of the energy consumption in Frederikshavn city. The model view shown in figure 2 is actually from that mock-up. The visualizations planned but not yet confirmed are texture colour changes of the buildings due to the energy balance or rather the share of the renewable energy compared with the overall energy consumption, animation of the wind turbines, flow between the main energy production and energy consumption referring to figure 3 and animation of local transport due to the motive power used.

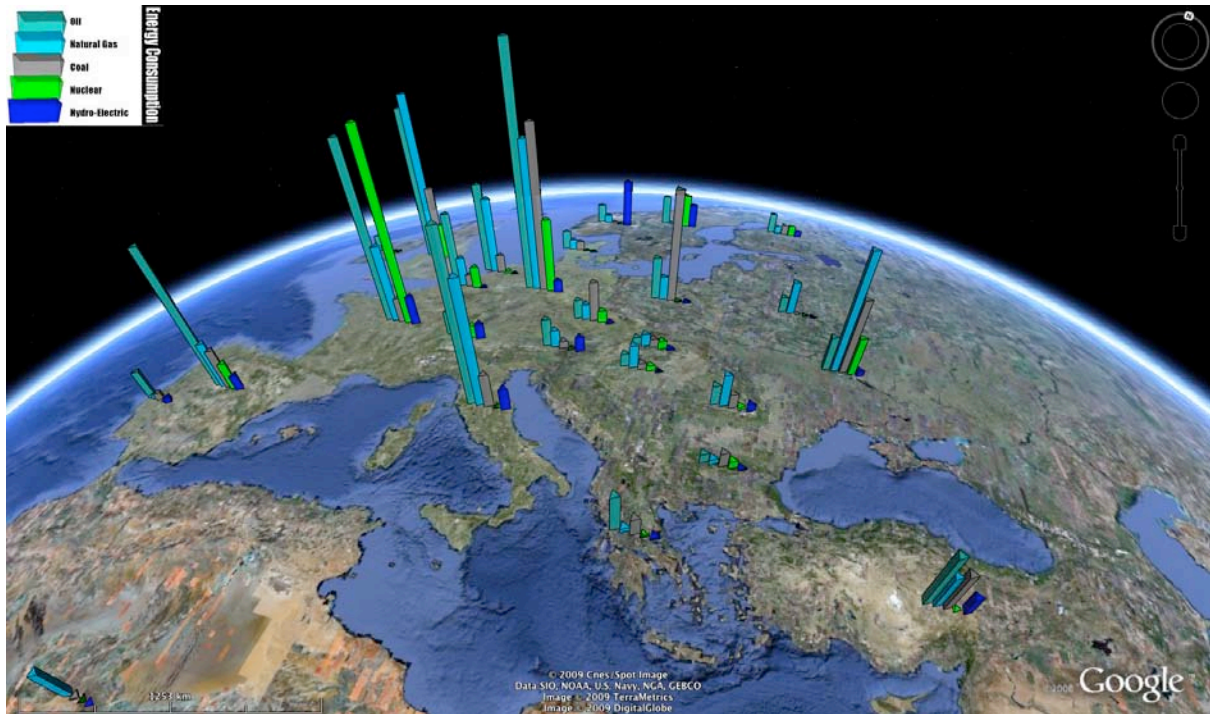


Figure 4 Energy consumption presented in a Google Earth environment

5. CONCLUSION

It is not easy to conclude anything about an ongoing project where the final result evolves day by day. However, a few important results are worth mentioning. The concept of managed objects has already proven to be very efficient when it comes to enhancing 3D models with dynamic behaviour due to the functionality defined within an object. Several years of research and programming lie prior to the GRIFIN system and it will not be an easy task to implement the concept in other known systems. The system has to be designed from scratch - more or less. However, GRIFIN is available as open source.

Another important lesson learned until now is regarding the model size and the handling of huge data sets. It is inevitable to use databases together with huge data sets. The handling of terrain and city model data gets tough not only because of the loading time and navigation in the model but also due to changes in the model, which is the case in this project. An appreciable side effect is the advantage always to load the newest available data set for the 3D model. The usability of the data set in the 3D environments reaches far beyond simple visualizations.

Finally concerning this specific application it is a pity that the energy calculation is limited to the flow between the energy production and large energy consumer units. With energy consumption data retrieved from each house it would be possible to design a completely different kind of interactive system, where every single inhabitant in the city would be able to see how he or she could contribute to the overall energy account. Danish law of privacy is prohibiting it.

BODUM, L., KJEMS, E., KOLAR, J., ILSØE, P. M. & OVERBY, J. (2005) GRIFINOR: Integrated Object-Oriented Solution for Navigating Real-time 3D Virtual Environments. IN OOSTEROM, P. V.

BUTLER, D. (2006) Virtual globes: The web-wide world. *Nature*, 439, 776-778.

CLARK, W (2009) Editor: Sustainable Communities, Springer Press, In print.

GOODCHILD, M. F. (2008): The use cases of digital earth. *International Journal of Digital Earth*, 1, 31-42.

GORE, A. (1998): The Digital Earth: Understanding our planet in the 21st Century. California Science Center, Los Angeles.

LIN, H., ZHU, J., XU, B., LIN, W. & HU Y. (2009): A Virtual Geographic Environment for a Simulation of Air Pollution Dispersion in the Pearl River Delta (PRD) Region. *3D GeoInformation Sciences*. Heidelberg, Springer.

KJEMS, E., BODUM, L. & KOLAR, J. (2009): Managed Objects for Infrastructure Data. *3D GeoInformation Sciences*. Heidelberg, Springer.

KJEMS, E. & KOLAR, J. (2005): From mapping to virtual geography. IN BATTY, S. E. (Ed.) CUPUM '05: Computers in Urban Planning and Urban Management: Abstracts of the 9th International Conference. London, Centre for Advanced Spatial Analysis, University College London.

KURODA T., NAKAMURA A., KOJIMA Y., MANABE Y., CHIHARA K.: "Interactive Visualization of Energy Consumption using VRML", In *Proceedings of SPIE*, Vol.4311, pp.218-225, (2001/01/25) San Jose/USA.

LUND, H. & MÜNSTER, E. (2003): Modelling of energy systems with a high percentage of CHP and wind power. *Renewable Energy*, 28, 2179-2193.

STOTER, J., KLUIJVER, H. de & KURAKULA, V. (2007): Towards 3D environmental impact studies: example of noise. *Advances in 3D Geoinformation Systems*. Heidelberg, Springer.

CV of Erik Kjems

Title: Associate Professor, PhD, M.Sc., B.Com

1994 Ph.D. degree

1994 Bachelor of commerce.

1987 Master of Science in civil engineering.

1997 Associate professor within the fields of GIS and road-design at Aalborg University.

2000 - 2007 Director of the VR Media Lab at Aalborg University.

2007 - Director of the Centre for 3D GeoInformation at Aalborg University

The latest projects are the “GRIFINOR” project, funded by the European Regional Development with € 2.5 Mill. for the period of 2001 – 2006 (2011). “InfraWorld” funded by the Norwegian Research Council with € 4.3 Mill. for the period of 2008 – 2012. And a smaller project “Energy City Frederikshavn” funded by a local funding institute for regional development with € 700.000 from which he will present some result today.