

# **Data Migration – The Component View**

**Dr. Martin SCHEU and Dr. Andreas ROSE, Germany**

**Key words:** Data Migration, Life Cycles, Data Transfer, Integration, Redesign, Five Components, Best Practice.

## **ABSTRACT**

Due to the ever shorter life cycles of hard- and software migration projects should be alltag everyday tasks and not worth to talk about. This is not true in many cases. The complexity of GIS systems makes migration to everthing but a simple data transfer. This fact regularly is underestimated and underestimations of project costs and duration henceforth are common. The migration of legacy GIS system therefore is dreaded and avoided if only possible. This paper presents a more general organizational approach which includes all mayor parts of IT-Systems as well as their relationships and dependencies. The approach simplifies and such increases the reliability of the planning phase. An authentic migration project is presented which shows that migration is not a painful task but rather an opportunity to optimize structures and processes.

## **ZUSAMMENFASSUNG**

Die Migration von Geodaten ist nicht nur die einfache Übertragung von Daten vom System A zum System B. Vielmehr treten bei einer unvorbereiteten Herangehensweise Seiteneffekte auf, die das Migrationsproblem schnell zu einem Flächenbrand werden lassen, der die gesamte IT-Landschaft der Anwendung erfaßt. Dies läßt sich durch die Verwendung der in diesem Beitrag vorgeschlagenen komponentenorientierten Sichtweise verhindern. Dabei werden fünf Komponenten benannt, die von einer Migration betroffen sein können: Hardware, Software, Daten, Geschäftsprozesse und Personal. Der Beitrag zeigt, daß die Komplexität der Migrationsaufgabe auch mit dem Grad der Abhängigkeit zwischen den Komponenten steigt. An einem Beispiel wird verdeutlicht, wie diese Planungsmethode in die Praxis umgesetzt werden kann.

## **CONTACT**

Dr.-Ing. Martin Scheu  
Technical University Berlin  
Straße des 17.Juni 135  
D-10623 Berlin  
GERMANY  
Tel. + 49 30 466 06 280  
Fax + 49 30 466 06 282  
E-mail: [scheu@inge3.bv.tu-berlin.de](mailto:scheu@inge3.bv.tu-berlin.de)  
Web site: [www.survey.tu-berlin.de](http://www.survey.tu-berlin.de)

Dr.-Ing. Andreas Rose  
grit GmbH: Berlin – Werne - Olpe  
Landwehrstraße 143  
D-59368 Werne  
GERMANY  
Tel. + 49 2389 9827 0  
Fax + 49 2389 9827 27  
E-mail: rose@grit.de  
Web site: www.grit.de

# Data Migration – The Component View

Dr. Martin SCHEU and Dr. Andreas ROSE, Germany

## 1. INTRODUCTION

Migration projects mainly are triggered by the replacement of hardware. And while it is obvious for the majority of IT managers that they have to take care of the software as well, many of them won't see much more problems with their data than to copy them to a new disk volume. Data Base managers and GIS managers have a more complex view. They are well aware of the fact that their data is their most expensive asset and they also know crucial their data are for their applications and their clients.

Hence the standard components of migration are:

- Hardware
- Software and
- Data.

Nevertheless this view still is much too restricted to get the whole picture. What about the following items?

- Business process
- People: those running the system and those using it.

While the first 3 components are obvious the latter two are mostly forgotten – with bad results. Business processes for instance should be verified and redesigned if necessary. Negligence of preparation, education and staff training results in mayor delays, severe degradation of system acceptance and henceforth system use.

Our experience proves that the complexity of migrations projects almost always is grossly underestimated. If you verify the reasons you will find that the planning process was too short sighted, that relations and dependencies were not recognised in due time and that the funding therefore was much too small.

This can be avoided by better preparation. And this requires to take a broader point of view.

## 2. DATA MIGRATION – WHY?

Let's first look at the causes for migration. It is a good practice to ask why we should initiate a migration project and what our aims are. So why do we migrate ?

First of all: Natural death. Every IT system one day reaches the end of it's life cycle and has to be replaced. Even if the hardware is still functional the signs of decease are:

- Tremendous service & maintenance costs
- Loss of connectivity (networking)

- Storage and user bottlenecks
- Unbearable performance

Other reasons follow the customers needs: WWW connectivity and integration of other data sources can be summed up to “informational needs” and often require new IT systems as well as the refurbishment of data structures.

Other reasons may be the implementation of a new data processing paradigm for the whole company, central data base servers for instance or new operating systems.

As manifold as the reasons are the resulting aims. All of them should be met although some of them may contradict each other. So you have to find appropriate compromises. And you have to look for intrinsic consequences.

In sum: Migration without thorough planning will lead to disaster. Let’s have a look at the following example.

### **3. IT-MANAGERS’ NIGHTMARE**

I-MAP Inc. is a Washington company selling international maps by means of an E-Commerce application delivered by “Micro E.STORE AG ” a trendy German Software Company. In January 02 John Doe, IT manager of I-Map gets aware of the fact that most of their customers have changed their currency to EURO. Besides of the new symbol “€” they expect the whole accounting to be changed to the new currency.

John Doe calls Micro E.STORE’s hotline. Fortunately the problem can be solved with a little software patch. John Doe opens his vendors web site and begins to download “SP\_1101.EXE” - a service pack of 81Mbyte.

Shortly after he starts SP\_1101.EXE and soon is rewarded with the following message box:  
“SP 1101 requires SW-Release 10.7. Installation aborted”.

In fact John Doe still uses SW-Release 8.2.. A Year ago he couldn’t install later versions of E.STORE because of problems with his data base system. But only a few telephone calls later John Doe knows, what he has to do:

First of all: purchase and installation of E.STORE V12.0 the latest release of his E-Commerce application. There are hints that E.STORE V12.0 somewhere has made minor changes of the data model, but that can be taken care of later. E.STORE V12.0 depends on VISION V9.20, the latest version of his RDBMS. That has to be installed as well. Last not least he has to take care of the operating system because both programs need APERTURE 2010, the latest release from MINISMOOTH Inc.

After installing that heap of software John Doe launches E.STORE V12.0. Everything runs well with exception the customer data base. Due to minor changes in the data structure the new system cannot access the old data. John Doe attains a workaround by exporting the old

customer data base to a spread sheet. From now on customer data is not taken from the data base instead they are copied into the other applications with “copy & paste”.

But there are good news too. E.STORE V12.0 provides XML-based functions to import the map catalogues of I-MAP’s suppliers. Up to now these catalogues came on paper and were typed in manually. Unfortunately John Doe didn’t get aware of that fact because he neither read the release notes nor underwent the appropriate “E.STORE Configuration Training Course”. IMAP so still uses its old procedure. The Catalogues are typed in manually.

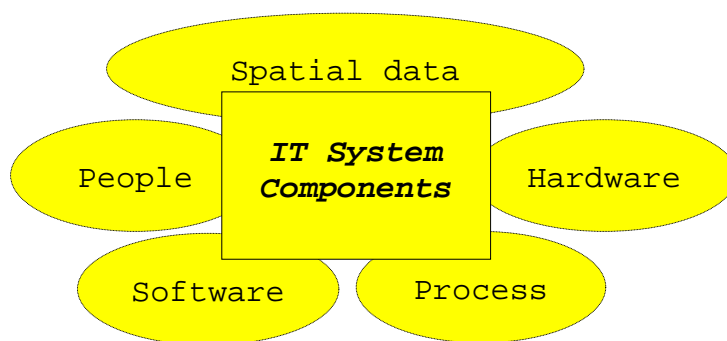
All ends well that begins well. E.STORE V12.0 is running now. Life just became a little bit more complicated. Company revenues have decreased also...

#### 4. THE COMPONENT VIEW

Our example surely is completely fictional but it shows what can happen if migration projects neglect parts of the system. And this is what the Component View aims at. The Component View is not the Swiss Knife for migration projects or a substitute for bad IT managers but a consistent and formal organisational view on IT systems which helps to keep the big picture and to recognise dependencies.

Let’s have a closer look to figure 1. It shows a complete view of an IT systems components. Obviously a migration project has to take care of each of these components to be successful.

Fig.1: Five components of data migration in GIS



These five components must be analysed before the migration process can start. As a result you get current IT systems paradigms.

Component	possible future paradigms
Data	File system, database
Software	Stand alone system, based intranet/internet
Hardware	Mainframe, PC
People	Central, local
Processes	Simple, complex

Table.1: Paradigm and components

Every migration process ends up in modification of one or more components. Table 1 shows possible future paradigms. Unfortunately the five components are not independent. Fig. 2 shows the dependencies.

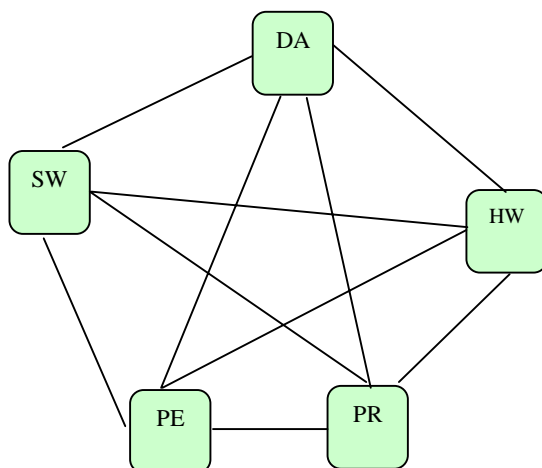


Fig.2: Components are not independent

Evidently complexity grows with dependencies. In fact you can measure the operational quality of a system architecture by measuring its dependencies. The less dependent its components are the better is the architecture. That is true for every IT system but for GIS systems in special degree.

So if you have to replace your system try to restrain your dependencies at the same time. That is a reason why scaleable hardware is so important.

- The number of your data or that of your users has doubled ? Just add more CPUs, RAM or hard disks.
- You cannot retrieve your data in a jungle of file systems ? Use a Database-Management-Systems to order and hide the complexity of your data structures.
- You have problems to give access to your data and provide information to the whole company ? Use WWW-Technology and OGC compatible Web Services to clean up multiple protocols and data distribution problems.

As we see modern IT-technology gives you an opportunity to achieve a minimum of dependencies between the five components. But technology is just one part. You have to analyse the whole system to find the appropriate measures. Let's give an example.

## 5. BEST PRACTICE

The following project was realized in 2001 and gives an example of our concept. The client, a state administration, is assigned to manage the state's real estates (about 70.000 land parcels). When the project began the administration used a GIS system which was a based on own developments. The system as well as the data had grown over the years. The maintenance costs had grown also. Especially the manpower needed to keep the data up to date had grown beyond tolerable measure.

One reason was that the system used data whose capture and maintenance lay in the responsibility of other administrations – the cadastral administration in this case. The update of the cadastral data was performed by inadequate means and took too much time. As a further consequence data links were not reliable and system use degraded.

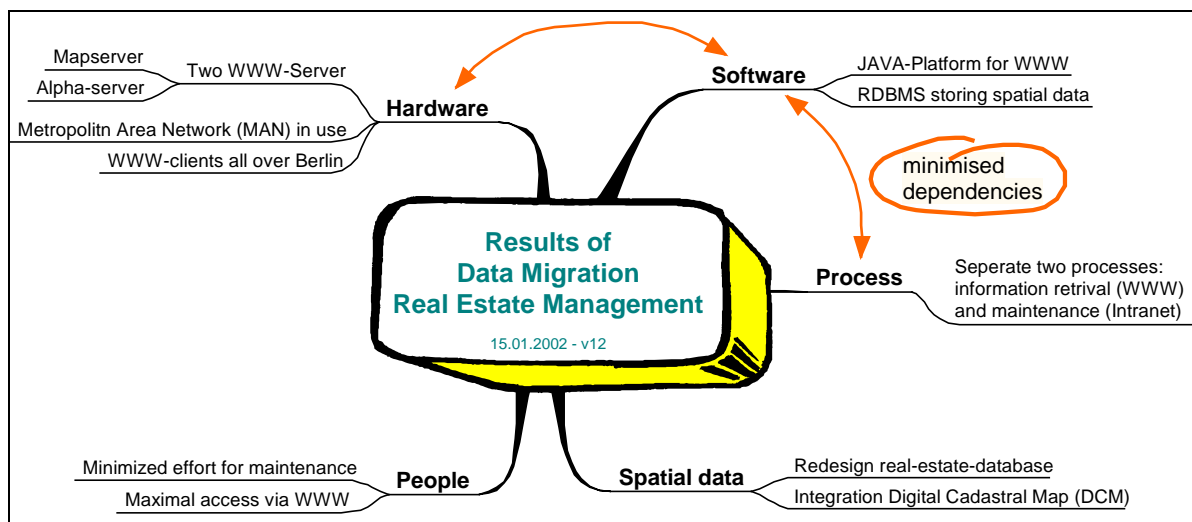


Fig. 3: Best practice: Redesign objectives

Luckily in 2000 the cadastral information got available my means of Internet Map Services so the whole concept could be changed. Instead of a stand alone system with local data the whole information retrieval process was changed to an internet based approach where information was kept and maintained at those places where the original responsibility laid.

As consequence the whole data base architecture had to be redesigned. Web Services took place where local data were. Specific data services delivered the data just when they were needed and of just those items which were needed. Such actuality and minimal effort was secured. Fig. 3 shows the result of one of our workshops during the redesign phase.

This concept presented a completely new data processing paradigm and required appropriate measures on side of the personnel as well. Work shops and training provided the know how which was needed to administrate and operate the new system.

Last not least: Every realization requires concise project planning. Four main tasks (Integration of DCM, Data Migration, Access via WWW and Application maintenance) could be identified which were able to be performed at the same time. This approach required more ressources but the project was recompensated by a substatial gain in time. Overall project time such could be reduced to 8 months. The data migration itself could be performed within 2 days so that the overall system downtime didn't exceed 5 days.

