TACIS Project

Support to the Implementation of

Land and Property Policy Tools

Workgroup "Business Processes of the Cadastre of Objects of the Immovable Property"

Suggestions on a reliable, scalable, secure and interoperable IT Architecture

Version 1.1 final

Mag. Walter Sedlacek, MSc MBA

Moscow/Taganrog/Vienna, July 2005

If you show problems and offer solutions people will act.

- Bill Gates

Abstract

This document outlines a generic IT architecture, which enables the synchronization between the two IT architectures of the *Registration of Rights* and the *Federal Agency of a Cadastre of Objects of the Real Estate*. In addition the IT architecture of the Taganrog solution for *Federal Agency of a Cadastre of Objects of the Real Estate* has been analyzed and is evaluated in this document. Generic considerations regarding public access to cadastral data, business processes, project management and Open Source Software close this document.

Document History

Date	Author	Version	Approved By	Change Summary
July 2, 2005	Walter Sedlacek	0.1	N/A	Creating initial document
July 5, 2005	Walter Sedlacek	0.2	N/A	Adding content from workgroup meeting
July 9, 2005	Walter Sedlacek	0.3	N/A	Adding content after Taganrog visit
July 12, 2005	Walter Sedlacek	1.0	N/A	Final review
July 13, 2005	Walter Sedlacek	1.1	approval pending	Changes only in formats
July 19, 2005	Walter Sedlacek	1.1 final	Prof. Reinhold Wessely	No changes

Content

1 Background 7
2 Method's of Investigation 8
3 Current Situation 9
3.1 IT Architecture 9
3.2 Interfaces 9
3.3 Access via Internet 9
3.4 Business Processes 9
4 Alternative Solution 10
4.1 IT Architecture 10
4.2 Interfaces 12
4.3 Access via Internet 13
4.4 Business Processes 13
4.4.1 Requirements for BPR tools 14
4.5 Preparation for World Bank Project 15
5 Pilot Implementation 17
6 Analysis and Evaluation of the Taganrog Solutions 20
6.1 Unified State Land Register (EGRZ) 20
6.2 Cadastre and Information Analytic Center (KIAC) 20
6.3 Summary for the Taganrog software solutions 23
7 Technical Enabler: Open Source Software 24
7.1 Definition 24
7.2 History 25
7.3 Projects 26
7.4 Traditional Software Management versus Open Source 27
8 Bibliography 29
8.1 TACIS project related 29
8.2 Technical 29
9 Conclusion and Outlook 31

Figures

Figure 1: Alternative IT Architecture 11
Figure 2: Interfaces to existing solutions 12
Figure 3: BPR: Design – Implementation – Controlling 14
Figure 4: Locations for pilot implementations 17
Figure 5: Equipment for Pilot Sites 18
Figure 6: Generic picture of the pilot installation 19
Figure 7: Taganrog cadastral data accessed from the internet 21
Figure 8: Taganrog cadastral data (details) accessed from the internet 21
Figure 9: KIAC solution for internet access 22
Figure 10: Open Source licence types 25
Figure 11: Most common Open Source projects 27

1 Background

Markets for land and real estate in the Russian Federation are under-developed and inefficient by the standards of developed market economies. Currently two parties, the *Registration of Rights* and the *Federal Agency of a Cadastre of Objects of the Real Estate* are responsible for cadastral and rights-related issues. They have their business processes partly documented and also partly electronically supported.
Whereas the *Registration of Rights* has many different software-solutions for their business processes the *Federal Agency of a Cadastre of Objects of the Real Estate* utilizes the same software from Taganrog in about 70% of their front offices. However both solutions are not designed for distributed data access and storage and do also not follow common standards or state-of-the-art technologies. Data synchronization between both parties is not possible or only on a very limited scale and the data is basically for local usage only.

One of the aims of the TACIS project *Support to the Implementation of Land and Property Policy Tools* is to overcome this situation and harmonize all software solution so that data is accessible remotely and from all relevant stakeholders in a secure and protected way. However the focus is on consulting the *Federal Agency of a Cadastre of Objects of the Real Estate* in order to establish a connection of all their databases.

Within the project a workgroup has been formed with the aim to investigate business processes. The name *Business Processes of the Cadastre of Objects of the Immovable Property* outlines the nature of the workgroup. The understanding of the business processes is the key to designing an IT architecture. However data consultation projects always lead to a common and well understood generic IT infrastructure. This infrastructure is outlined in this document. In addition the following topics are discussed: Interfaces to existing IT solutions, access via internet, business processes, preparation for World Bank project, pilot implementation and evaluation of the Taganrog software solution.

2 Method's of Investigation

Finding out the AS IS situation is always the key to migration and consolidation projects. The author of this document attended a workshop with experts of the TACIS project and related projects. In addition some documents were read; the most important ones are listed in the chapter *Bibliography*. Also a visit to Taganrog was made to gather more information on existing software solutions, which are produced there. The following issues made it difficult to outline the AS IS picture:

- Too many parties are involved in the project and their relationship to each other is not clear to the author of this document.
- Very often goals other than the success of harmonization of all software solution drive the information flow. Gathering correct information was sometimes not easy.
- Responsibilities and competences are not clearly defined and executed. This leads to wrong assumptions of the organizational hierarchical system.

However, in an environment of strong changes, such as Russia, the issues mentioned above were to be expected. The investigation was therefore done in a generic way, not trying to find out details. This way of fact-finding enables only a generic solution, again without details. A detailed solution would need an in-depth AS IS picture and this can only be achieved by overcoming the issues above.

3 Current Situation

3.1 IT Architecture

Data is stored in localized databases and is accessible mostly with client-server type applications.

3.2 Interfaces

No common interfaces are defined. All software solutions are independent and locally. Data exchange can only be done manually. However, in pilot installations, the Taganrog solution enables data-exchange via well defined interfaces. Details are discussed in the chapter *Analysis and Evaluation of the Taganrog Solutions*.

3.3 Access via Internet

Pilot access via internet is possible for data regarding land and it's evaluation. Details are discussed in the chapter *Analysis and Evaluation of the Taganrog Solutions*.

3.4 Business Processes

Business processes are partly defined but not widely agreed upon. A workgroup from Taganrog has been documenting business processes for one year with Microsoft's VISIO. These documents are static and the described business processes can not be simulated or monitored.

4 Alternative Solution

The following solutions are generic ones and open for discussion. Basically state-of-the-art technologies are used. They are well understood and therefore maintainable at low cost by many suppliers.

4.1 IT Architecture

A three-tier architecture has been chosen as an alternative solution. The first tier on the Federal level is a centralized solution. The second tier on the Cadastral Unit level consists of 89 distributed solutions, which are connected to the Federal level. From the Front Offices data on the Cadastral Unit level is accessible via the Internet. Please also refer to figure *Alternative IT Architecture*. The following describes the main features:

- Enable OLAP[1] (Online Analytical Processing) for in-depth analysis on a Federal level. Here queries for long term trends and detailed analysis can be made.
- Enable OLTP[2] (Online Transaction Processing) for fast analysis on a Cadastral Unit level. Here queries and entries for daily usage can be made.
- Data should be transported using IP technology. For security reasons VPN[3] (Virtual Private Network) gateways must be implemented and utilized. In addition the content should be signed electronically to ensure the correct source of the data.
- There shall be a backup data transport mechanism among all three levels, in case the online connection is down or not available at all.
- The whole IT architecture should be monitored to enable reliable IT operations and reduce maintenance costs. Here the free monitoring tool Nagios[4] should be

[1] http://en.wikipedia.org/wiki/olap

[2] http://en.wikipedia.org/wiki/oltp

[3] http://en.wikipedia.org/wiki/vpn

[4] http://www.nagios.org

used. Nagios is one of the most scalable, flexible and powerful monitoring tools available.

- Data on the Cadastral Level should be accessible via a Web-Browser. This enables low maintenance costs and is user-friendly.

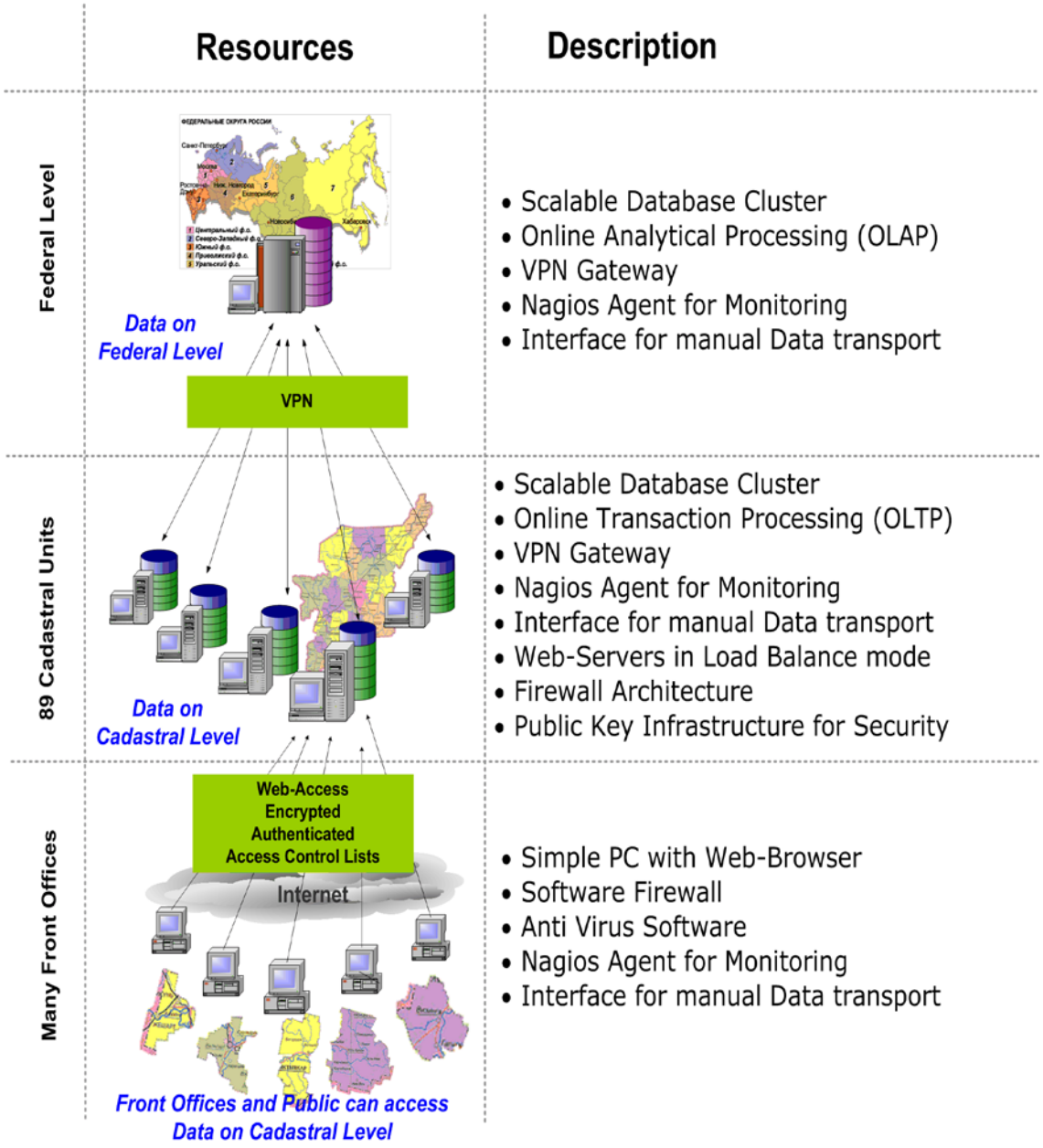

Figure 1: Alternative IT Architecture

4.2 Interfaces

Interfaces to existing solutions must be implemented. Permanent synchronizations between the existing solution of the *Registration of Rights* and the *Federal Agency of a cadastre of Objects of the Real Estate* must be possible. All solutions should be used in parallel. Later customers will be convinced that the new solution is better due to usability, consistency, and availability. Slowly the legacy solution can, but does not have to, be consolidated to the new solution.

The Open Source tool DBI Proxy[5] (Database Independent Interface for Perl) with some manual adjustments could be the tool of choice to enable the scenario mentioned above.

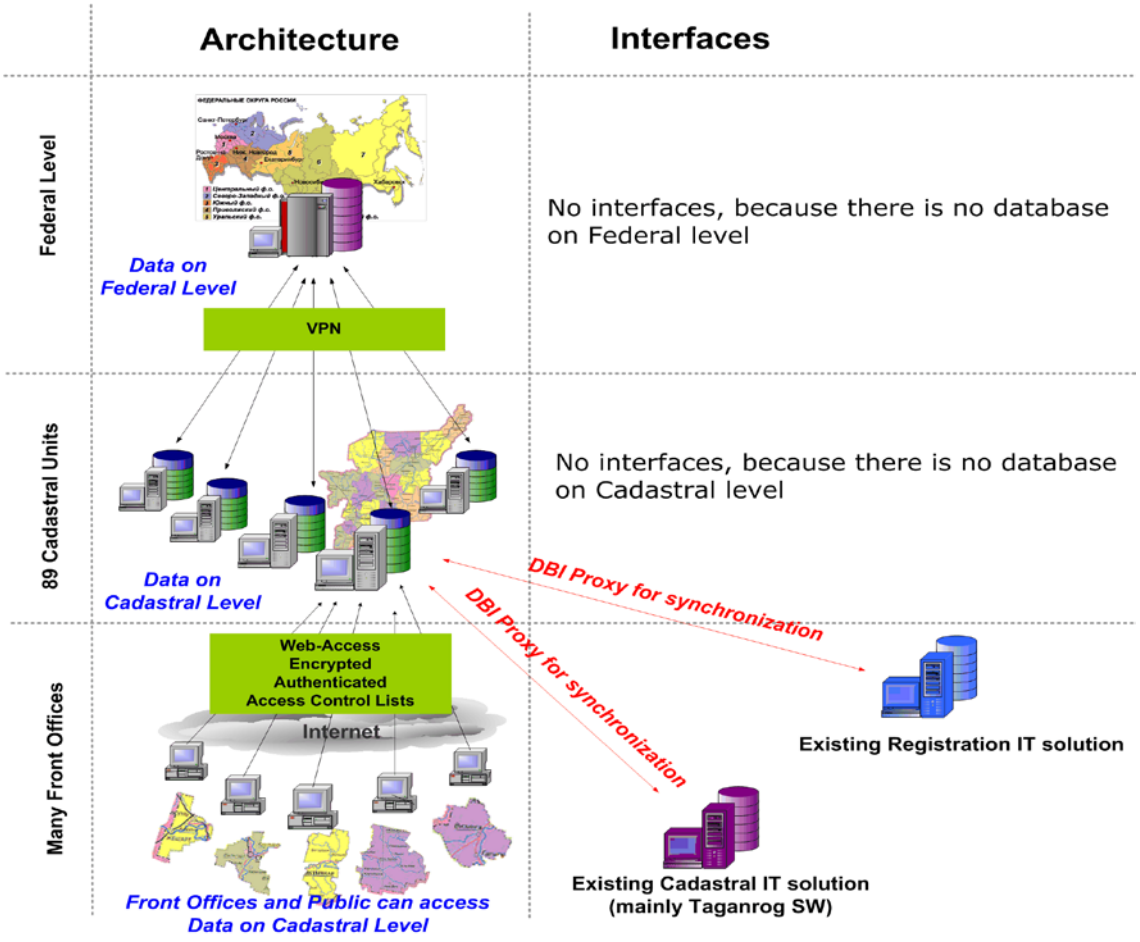

Figure 2: Interfaces to existing solutions

[5] http://dbi.perl.org

4.3 Access via Internet

Access via internet is vital for 3rd parties but also for the Front Offices. Due to the architectural design internet access is easily possible. The Web-Site shall be simple and read-able for all Web-Browsers and also the Web-Site must be optimized for low band-widths. In addition the Web-Site shall have a similar look-and-feel as the previous Cadastre or Registration-of-Rights applications.

The following add-ons must be implemented to ensure the required security:

- All access is encrypted so nobody can read the content of the Cadastral data without authorization. This should be done with secure-sockets-layer[6] (SSL, https) technology and public/private keys.
- Data must be electronically signed so the customer can be sure that the data-source is the Cadastral database.
- All clients must be protected by a local firewall, an Anti-Virus-System and automated updates.
- All clients are monitored to enable pro-active maintenance, to increase the uptime and to reduce maintenance costs.

4.4 Business Processes

Because 70% of all employees are knowledge workers, business processes are very important for them. If you give them poorly designed business processes they will deliver poor results. Business Process design is mainly Business Process Reengineering (BPR). BPR is a very difficult task and one will get all kinds of resistance when executing BPR projects.

There are two approaches for BPR: *Top-Down* and *Bottom-Up*. Whereas *Top-Down* follows the state-of-the-art and well known *Vision-Mission-Goals-Action-Feedback* idea, *Bottom-Up* measures and plots existing processes online for further improvements.

[6] http://en.wikipedia.org/wiki/Secure_Sockets_Layer

BPR always requires measurement of its implementation (controlling). When performing BPR one must always follow the following circle[7]:

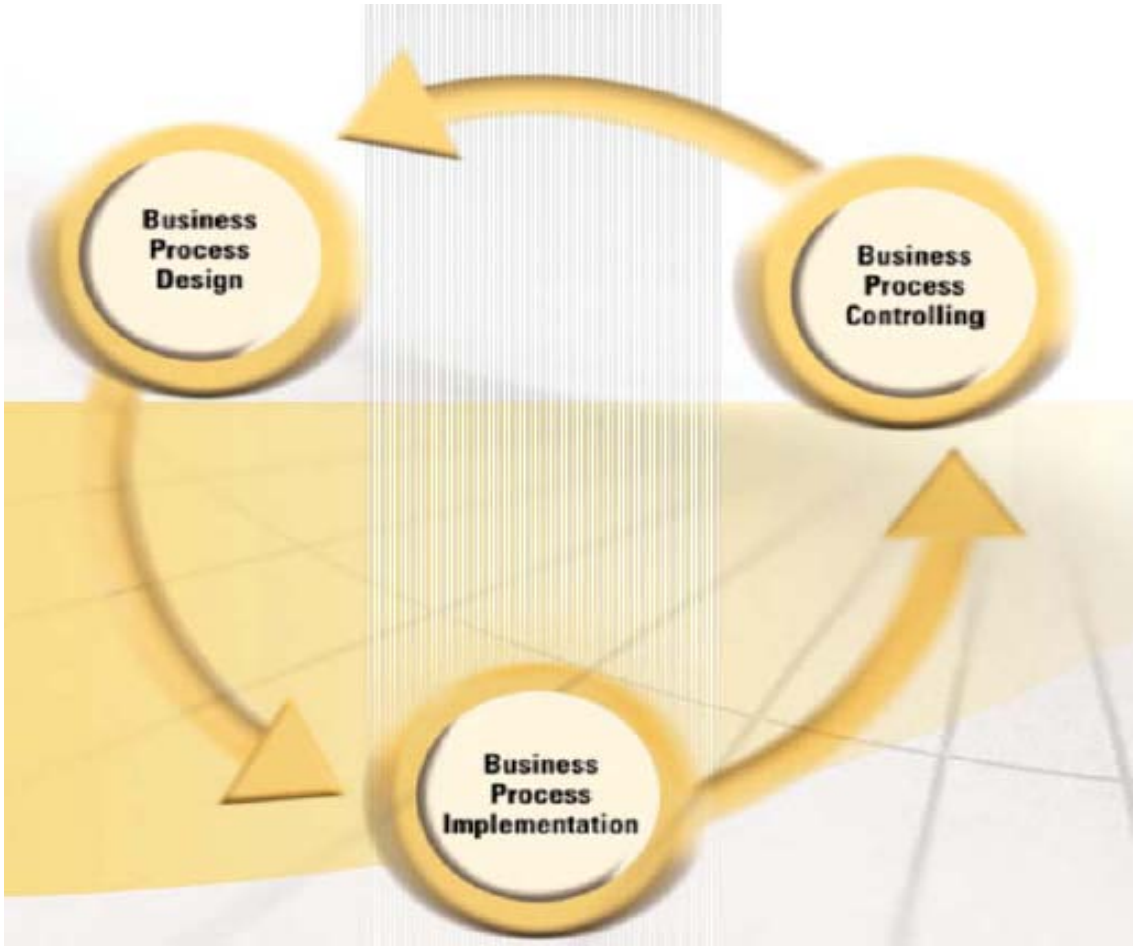

Figure 3: BPR: Design – Implementation – Controlling

4.4.1 Requirements for BPR tools

There are many tools which support BPR. The following features are a suggestion for features of a BPR-tool:

- Store business process data in a database
- Take data online from existing systems
- Define Key Performance Indicators (KPIs)

[7] picture is taken from IDS Sheer's ARIS presentation

- Enable statistics
- Enable standard interface for data exports
- Possibility to import of external data
- Scalability for enlargement

4.5 Preparation for World Bank Project

For the World Bank project the project management methodology *Waterfall Model* shall be utilized. Other models are possible too, but the complex environment requires a model which is easy to understand and to execute. The following activities describe the phases, which must be executed in sequence, within this model:

Plan

1. Create Project Charter
2. Plan for Define Phase
3. Tollgate Review
4. Sign off by Steering Committee

Define

1. Gather Requirements
2. Design Architecture
3. Select Supplier
4. Tollgate Review
5. Sign off by Steering Committee

Construct

1. Detailed Design
2. Build
3. Tollgate Review
4. Sign off by Steering Committee

Test

1. System Test
2. User Acceptance Test

3. Tollgate Review
4. Sign off by Steering Committee

Deploy

1. Pilot
2. Full Scale Deployment
3. Tollgate Review
4. Sign off by Steering Committee

5 Pilot Implementation

At three destinations a pilot implementation shall take place:

- Taganrog
- Perm
- Novgorod

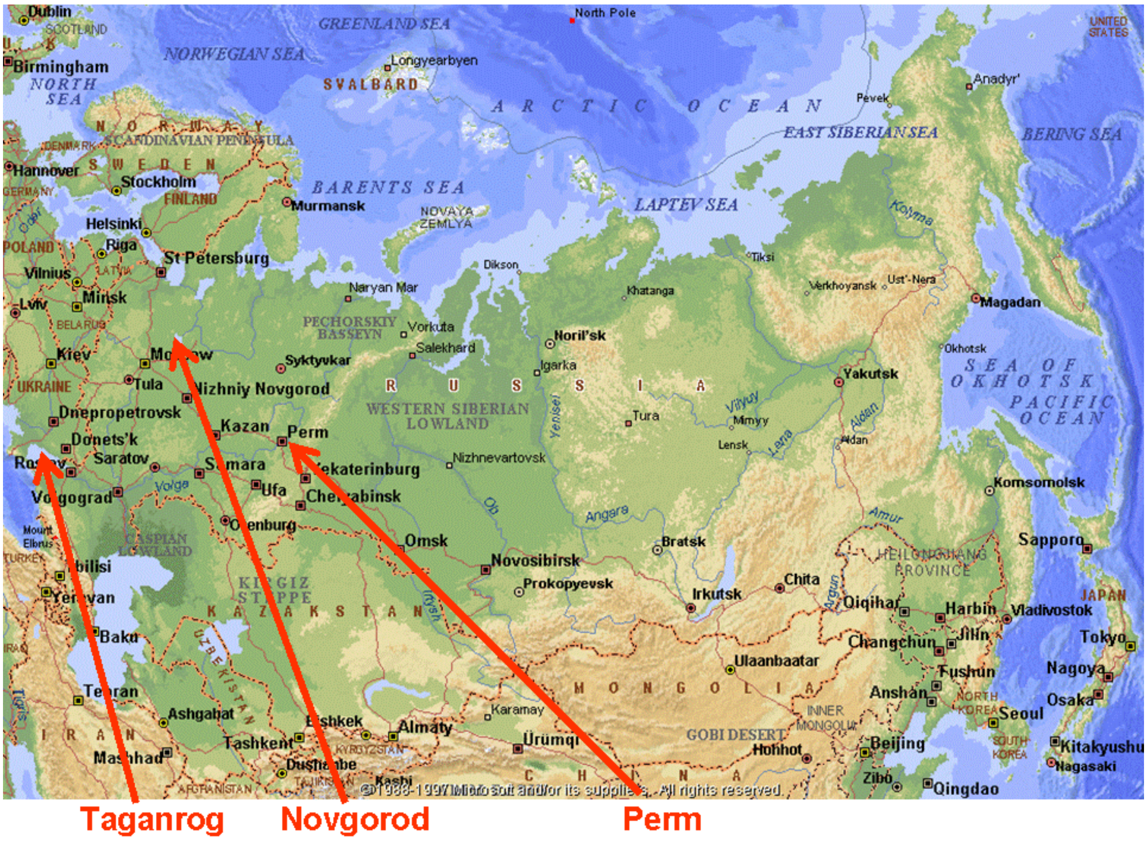

Figure 4: Locations for pilot implementations

It shall be demonstrated that the connection and synchronization to the existing solutions (*Registration of Rights* and the *Federal Agency of a Cadastre of Objects of the Real Estate*) and an internet access is possible.

The following equipment has already been purchased to support that:

Pilot Destination	**Equipment**
Perm, Rostov, Novograd	12 port Ethernet Switch
Perm, Rostov, Novograd	WAN router
Perm, Novograd	4x Workstation (P4, 256 MB RAM, 80-120 GB HDD)
Perm, Rostov, Novograd	Server (2x 2.4 GH CPUs, 1 GB RAM, 2x 77 GB HDD)

Figure 5: Equipment for Pilot Sites

The table above is not complete, but it shows that for the pilot the following software could be installed and configured:

- Operating System: Linux[8]
- Database management System: MySQL[9]
- Public/Private Key certificate: CA Cert[10]
- Firewall: IP Cop[11]
- Webserver: Apache[12]
- Anti-Virus-System / Intrusion Detection: Snort[13]
- Connector to existing databases: DBI Proxy
- Application Development: Perl[14]

[8] please see chapter *Technical Enabler: Open Source Software*

[9] http://www.mysql.com

[10] http://www.cacert.org

[11] http://www.ipcop.org

[12] http://www.apache.org

[13] http://www.snort.org

[14] http://www.perl.org

The connection between the existing solution and the new proposed one shall be done via VPN.

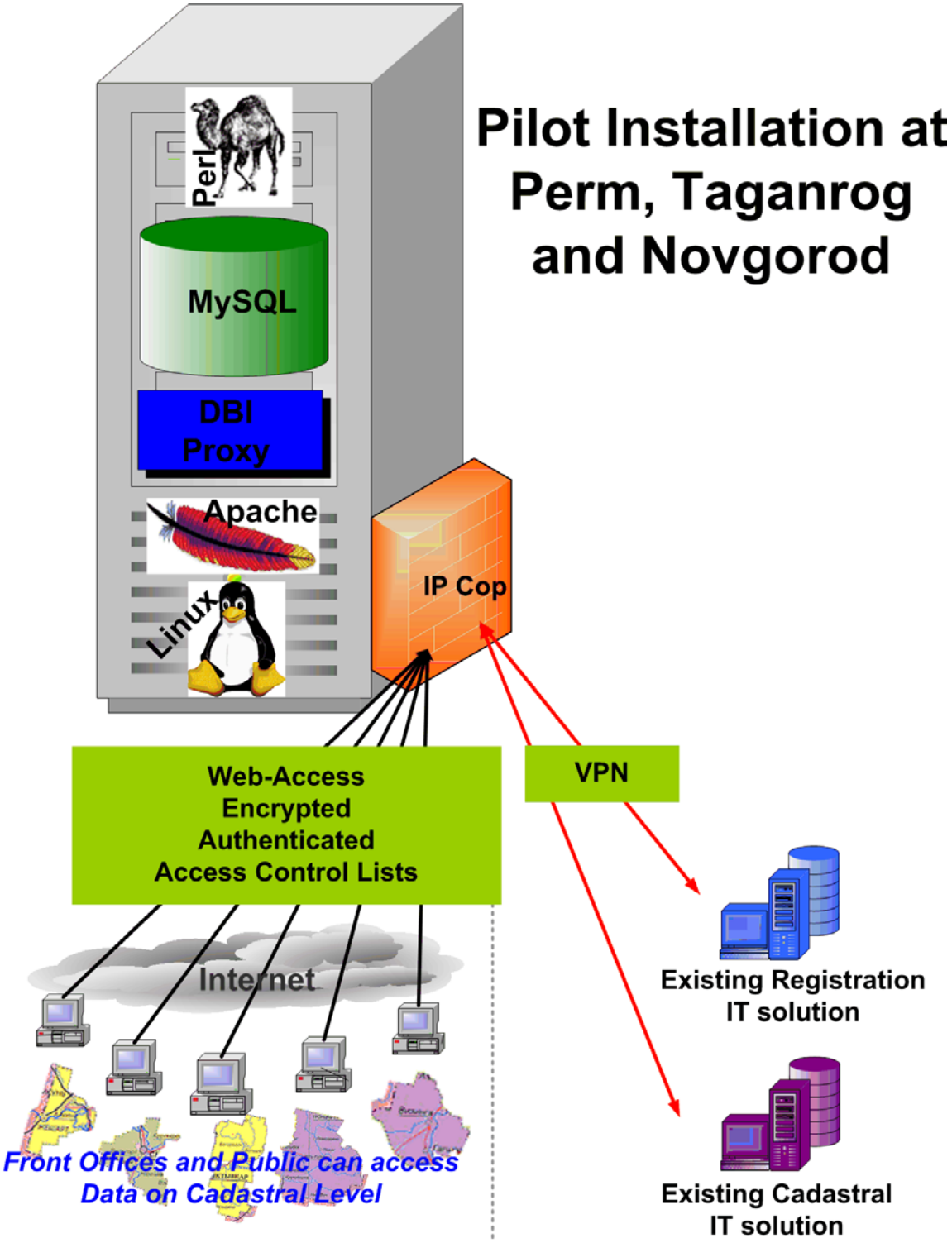

Figure 6: Generic picture of the pilot installation

6 Analysis and Evaluation of the Taganrog Solutions

Software from Taganrog is used in about 70% of all Front Offices of the *Federal Agency of a Cadastre of Objects of the Real Estate*. Migration of all Front Offices to this software is planned. Two software solutions from Taganrog are discussed in this document: the public access to cadastral data *Cadastre and Information Analytic Center* (KIAC) and the main software *Unified State Land Register* (EGRZ).

6.1 Unified State Land Register (EGRZ)

The main software from Taganrog is EGRZ. It is a local solution for cadastral data entries and accesses. The IT architecture is quite simple: It is a client-server system with a database engine behind it. All three components (client, server, and database) can run on one device or distributed on two devices. SQL is utilized in a native manner where possible. An XML or text-file export is possible and enables the offline synchronization with other parties. Basically this IT solution seems to be capable to serve the requirement from the *Federal Agency of a Cadastre of Objects of the Real Estate*. However it is only a local solution. For recommendations please refer to the chapter *Summary for the Taganrog software solutions*.

6.2 Cadastre and Information Analytic Center (KIAC)

KIAC is a pilot, which shows *Land and it's Evaluation* data on the internet[15]. The solution is accessible for everybody. A filter system makes sure that only legally permitted data is published on the Web-Site.

[15] http://www.srcc.ru

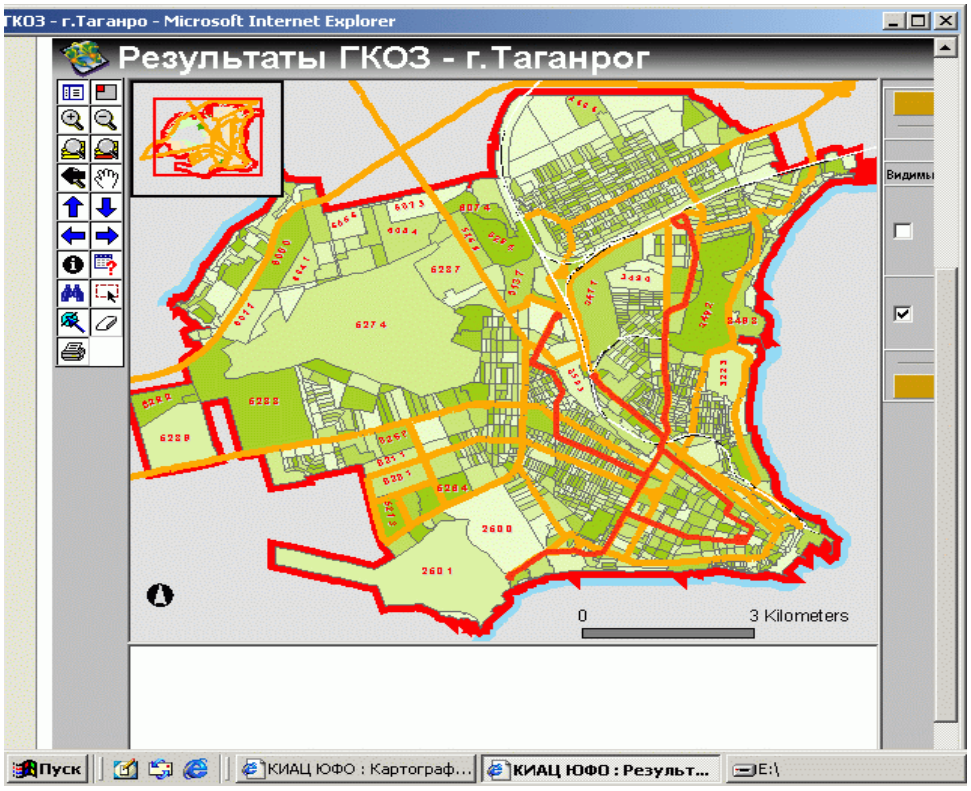

Figure 7: Taganrog cadastral data accessed from the internet

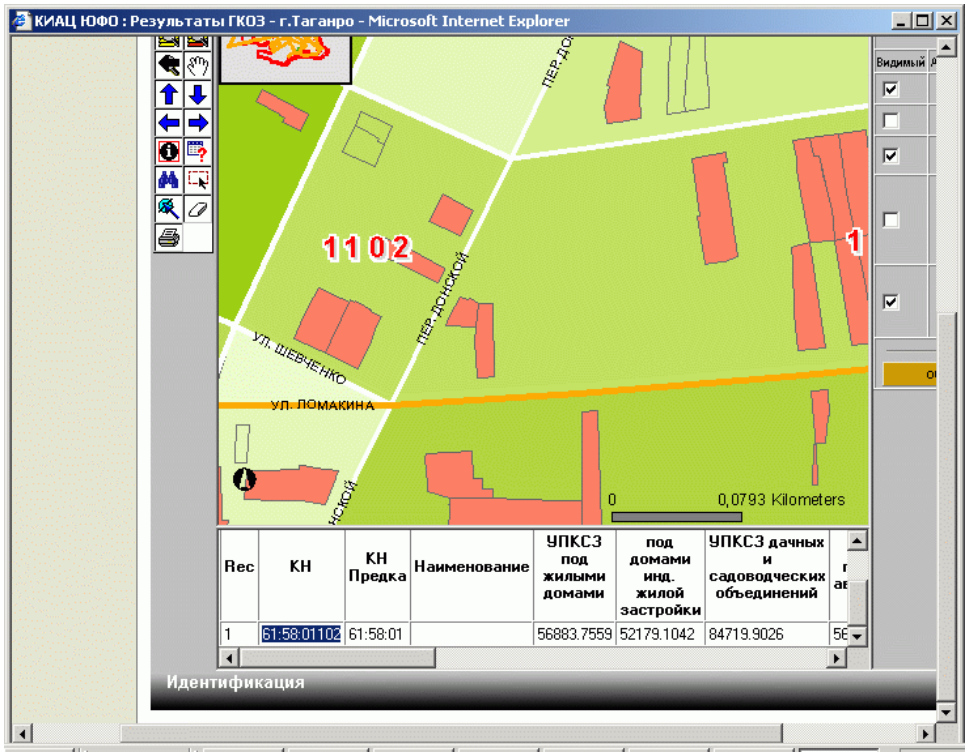

Figure 8: Taganrog cadastral data (details) accessed from the internet

The figures above show that access to cadastral data is possible and the solution is up-and-running. The IT architecture for that solution utilizes five existing databases from the main Taganrog software EGRZ and transfers incremental data within 6 hours to the KIAC database. Data is transferred using encryption, compression and the SMTP[16] protocol. Besides other issues SMPT (simple mail transfer protocol) is generally not recommended for file transfers. Therefore FTP[17] (file transfer protocol) should be utilized – also for the KIAC solution. Please refer to the chapter *Summary for the Taganrog software solutions* for details. The figure below illustrates the current KIAC IT architecture.

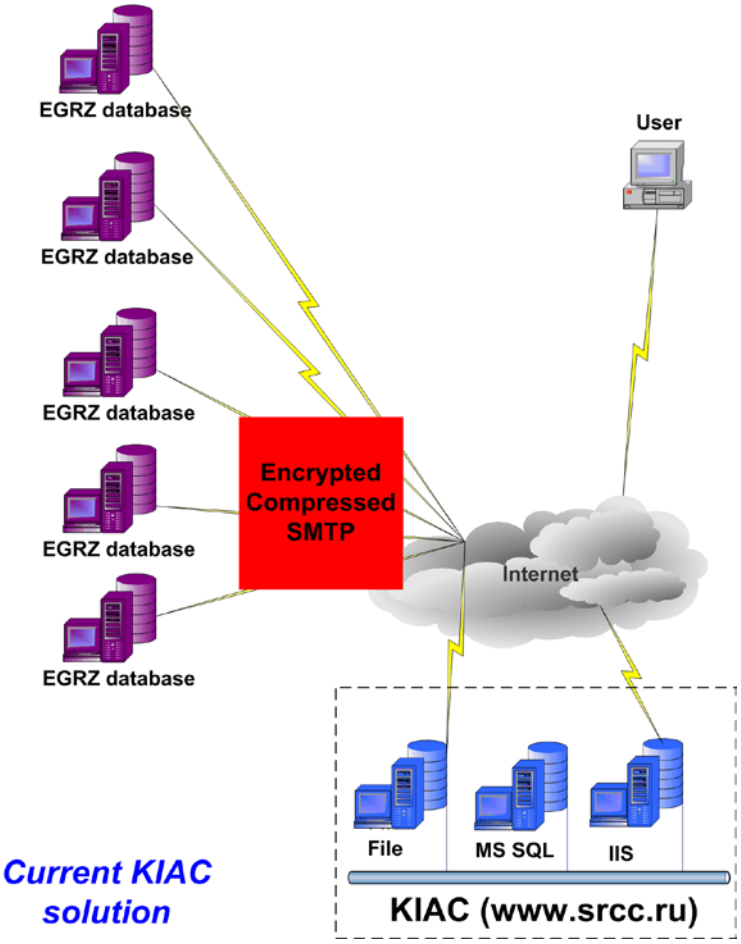

Figure 9: KIAC solution for internet access

[16] http://en.wikipedia.org/wiki/smtp

[17] http://en.wikipedia.org/wiki/ftp

Because electronic signatures are not legally binding in Russia, hard-copies must be transferred additionally. This slows down the underlying business process.

In future the KIAC solution will also show data for *Real Estate*, but due to legal regulations this is currently not possible. The underlying law for *Real Estate* data is not official yet.

6.3 Summary for the Taganrog software solutions

The following suggestions would help to move the Taganrog software solution towards state-of-the-art technologies:

- For encryption VPN technology should be utilized. VPN is more reliable than any other encryption technology, because it operates on IP layer.
- On the application layer FTP instead of SMTP should be utilized for file-data transfers. SMTP is used for Email transportation whereas FTP is used for file transportation. Because of the VPN tunnel there is no security risk when using FTP.
- Enable legally binding electronic signatures in Russia.
- EGRZ software should be coded in a more modern way. Here web-technology would be the best choice. This will reduce maintenance, implementation and design costs dramatically. Even if there is no WAN available – web-technology also makes sense in a local environment.
- Open Source Software (see chapter *Technical Enabler: Open Source Software*) should be used. Details and arguments are outlined in the next chapter.

The main rules applied in Taganrog for developing Taganrog software (always implement interfaces and always document them) are state-of-the-art and must be mentioned as a very positive step.

7 Technical Enabler: Open Source Software

Open Source Software (OSS) shall be utilized for enabling the suggested IT architecture. OSS is a very cost effective and also reliable solution. The following chapters outline the basics of OSS.

"No traditional developer can match the pool of talent the Linux community can bring to bear on a problem. Very few could afford even to hire the e.g. 800 people who have contributed to Open Source projects."[18]

"Hewlett-Packard is hosting a number of Open Source software projects that run on various Hewlett-Packard systems."[19]

7.1 Definition

One of the main problems in traditional software development is that the one who finds an error in an application communicates this to another person – the software engineer – who fixes the problem. One can easily see that this process is time-consuming and not very effective due to various sources of misinterpretation.
The idea behind Open Source is very simple: When programmers can read, redistribute, and modify the source code for a piece of software, the software evolves. People improve it, people adapt it, people fix bugs. It is often the same person who finds the error, fixes it and tests the fix.
There are many Open Source licence types. The main difference is the possibility of re-using the software code for commercial or non-commercial applications:

[18] http://catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/ar01s11.html

[19] http://www.opensource.hp.com/

Licence type	Can be used with commercial software	Changes must be free again	Can be published under other conditions	Contains specific rights for the licence owner
GPL	No	Yes	No	No
LGPL	Yes	Yes	No	No
BSD	Yes	No	No	No
NPL	Yes	No	No	Yes
Public Domain	Yes	No	Yes	No

Figure 10: Open Source licence types

Open source does not just mean access to the source code. The distribution terms of Open Source software must comply with the following main criteria:

- The license shall not require a fee for selling the product.
- The program must include source code, and must allow distribution in source code as well as compiled form.
- The license must allow modifications and derived works, and must allow them to be distributed under the same terms as the license of the original software.

7.2 History

In 1956 AT&T[20] was forced to limit its activities to its core business tele-communication and sell its patents on the free market. One of those patents was an operation system called UNIX, developed 1969 by Ken Thomson and Dennies Ritchie in the Bell Labs[21]. AT&T did not want other companies to have UNIX and therefore the price was incredibly high and there was no support at all. Universities got UNIX for small fees but still no support was available. That is why the exchange of knowledge and source code between universities was a must in order to enhance UNIX and get errors fixed. The coordination was done by the University of Berkeley[22], which later published UNIX by using the in-house software distribution BSD[23]. One of the first

[20] http://www.att.com/

[21] http://www.bell-labs.com/

[22] http://www.berkeley.edu/

[23] http://www.bsd.org/

versions was announced in 1978 by Bill Joy, who later founded SUN[24] . In the same year Arpanet – the former Internet – was established, based on UNIX computers. 1982 IBM[25], HP[26] and DEC[27] published their versions of UNIX and sold it on the free market, now including support. This was the end of more or less free UNIX versions.
In order to keep free UNIX versions available Richard Stallmann founded the company Free Software Foundation[28] and the GNU project[29]. With GPL[30] Stallmann defined a new license-model, which basically ensures that free software stays free after being modified by others. At this time no Kernel for a free UNIX system was available. This problem was first solved by Linus Torvalds[31] by publishing the first Linux[32] Kernel in 1991. This date can be seen as the birthday of Open Source, because now a free operating system including major utilities and applications was available.

7.3 Projects

Most users are not aware that many applications they use are from the Open Source world. Most of the UNIX utilities are developed under GPL license. For example the user-friendly Domain Names for Internet Web-Sites would be not possible without BIND[33], another Open Source product. The most frequently used web-server is Apache[34], which is also free. The following table gives an overview of the most common Open Source applications.

---

[24] http://www.sun.com/

[25] http://www.ibm.com/

[26] http://www.hp.com/

[27] http://www.digital.com/

[28] http://www.gnu.org/fsf/fsf.html

[29] http://www.gnu.org/

[30] http://www.gnu.org/copyleft/gpl.html

[31] http://www.cs.helsinki.fi/u/torvalds/

[32] http://www.linux.org/

[33] http://www.isc.org/products/bind/

[34] http://www.apache.org/

Project name	Description	URL
GNU	e.g. compiler suite	http://www.gnu.org/
Apache	Most used Web Server	http://www.apache.org/
BIND	Implementation of DNS protocols	http://www.isc.org/products/BIND/
FreeBSD	UNIX derivative from Berkly	http://www.freebsd.org/
Linux	UNIX derivative originally based on Intel platform	http://www.linux.org/
Sendmail	Most used agent for transporting emails	http://www.sendmail.org/
Samba	Agent to emulate file and print-service for Windows on UNIX	http://www.samba.org/
Perl	Scripting language for e.g. CGI - also known as "Web Glue"	http://www.perl.org/

Figure 11: Most common Open Source projects

7.4 Traditional Software Management versus Open Source

One of the most common arguments against Open Source is that those projects are lacking management in a traditional way. Software managers often argue that each software project needs the following qualities, executed by a manager[35]:

1. To define goals and keep everybody pointed in the same direction
2. To monitor and make sure crucial details don't get skipped
3. To motivate people
4. To organize the deployment of people for best productivity
5. To marshal resources needed to sustain the project

ad 1: One of the best-known theorems of software engineering is that 60% to 75% of conventional software projects either are never completed or are rejected by their intended users. If that range is anywhere near true then more projects than not are being aimed at goals that are either not realistically attainable, or simply wrong. In summary many goals fail in *normal* software project management and many have great success in the Open Source world (e.g., the longevity of Emacs[36], or Linus Torvald's ability to mobilize hordes of developers). Thus the project leaders and tribal elders who fill the manager's role in the Open Source world deliver equal or better qualities in defining goals than traditional software managers do.

[35] http://catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/ar01s12.html

[36] http://www.gnu.org/software/emacs/emacs.html

ad 2 and 3: The strongest argument of the Open Source community is that decentralized peer review trumps all the conventional methods for trying to ensure that details don't get out of control. Motivation is one of the key elements why tracking and monitoring works in such an environment.

ad 4: Open Source has been successful partly because its culture only accepts the most talented 5% of the programming population. It is often cheaper and more effective to recruit self-selected volunteers from the Internet than it is to manage buildings full of people who would rather be doing something else.

ad 5: Resource marshalling is basically defensive; once you have your people and machines and office space, you have to defend them from other managers competing for the same resources. But Open Source developers are volunteers, self-selected for both interest and ability to contribute to the projects they work on. The volunteer ethos tends to take care of the “attack” side of resource-marshalling automatically - people bring their own resources to the table.

8 Bibliography

8.1 TACIS project related

Ulf **Bjalkefors**, Nikolay Sazonov: **Findings Novgorod and Leningrad**; October 2004

Cadastre Project – ru / **LARIS II**; March 8, 2005

Support for the Implementation of Land and Property Policy Tools
Conference May 2004

Klaus **Went**: **Land allocation in Russia**
Some advice and hints fort he new law on Land allocation, second reading

Prof. Reinhold **Wessely**: **TACIS Project Work Plan 2005**

Prof. Reinhold **Wessely**:
Launch of Working Group on Cadastre Issues; Moscow, January 28, 2005

Prof. Reinhold **Wessely**: **Cadastral valuation of land and real estate**

8.2 Technical

John **Deer**: Network Management with **Nagios**; Linux Journal, July 2003

Free Software Foundation; http://www.fsf.org; September 10, 2003

GNU General Public Licenses
http://www.gnu.org/copyleft/gpl.html; September 10, 2003

GNU's Not Unix; http://www.gnu.org; September 16, 2003

HP Openview: Enterprise Management System
http://www.openview.hp.com; September 8, 2003

HP: Open Source; http://www.opensource.hp.com; September 22, 2003

IBM: **Tivoli** Enterprise Management System
http://www-3.ibm.com/software/tivoli; September 8, 2003

Peter C. **Lockmann**: Architektur von **Datenbanksystemen**; dpunkt Verlag 2004

Martin **Müller**: **Open Source**; O'Reilly Verlag, 1999

Microsoft: **Windows 2000** Core Requirements; 2nd edition, Microsoft Press, 2003

Networkuptime: Ethernet
http://www.networkuptime.com/faqs/ethernet; October 20, 2003

Open Source; http://www.opensource.org; September 16, 2003

Eric Steven **Raymond**: **The Cathedral and the Bazaar**
http://catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar; September 13, 2003

Andreas **Zink**: **Lokale Netze**; 7. Auflage Addison Wesely, 2003

9 Conclusion and Outlook

The suggestions discussed above are simple enough for quick implementation. The pilot in particular can be installed within a short time-frame. But this solution must not necessarily be adopted – other solutions are also welcome. The beauty of this solution is that the implementation needs very limited support from Taganrog and can be executed nearly independently.

The two discussed software solutions from Taganrog *Cadastre and Information Analytic Center* (KIAC) and *Unified State Land Register* (EGRZ) have the generic capability to enable public access over the internet and synchronization to other databases. Pilots are currently in place and prove this. Attempts to move to standards and state-of-the-art technologies can be observed. The most important issue is to enable legally binding electronic signatures in Russia. Without this, all attempts to transfer confidential data, no matter which encryption technology is used, will fail because hard-copies must be transferred additionally. This slows down the business process dramatically and therefore limits the acceptance of all electronic solutions.

The future will show how public access and synchronization of data of the *Registration of Rights* and the *Federal Agency of a Cadastre of Objects of the Real Estate* will be implemented. A move from the current Taganrog software to a more open IT architecture and the utilization of web-technology would be beneficial for both the *Registration of Rights* and the *Federal Agency of a Cadastre of Objects of the Real Estate*. The team in Taganrog is – in my opinion – able to execute this transition.

Mag. Walter Sedlacek, MSc MBA
walter.sedlacek@gmx.net