

# The Akelius IT Environment in Sweden

Emanuel Bergsten, 2010-09-22

This short paper will provide an image about Akelius IT environment in Sweden. Furthermore, it will describe the IT governance for Akelius Sweden. Finally it will discuss future developments and provide some of the underpinning theoretical guidelines.

## IT Governance

In Akelius Sweden, the governance of IT is carried by the IT management board. This board is responsible for giving the business the IT support it requires. This responsibility is carried out through a number of specialized groups, system owner groups, consisting of interdisciplinary groups representing the different departments and users. The IT department is responsible for managing all applications and carrying through the changes decided by the IT management board or the system owners.

### IT management board

Consisting of the two regional managers in Sweden, the vice president and the IT-manager. They grant funds for projects that system owners wish to carry out. They also approve all annual system owner budgets.

### System owners

The system owners are experts in both the systems and in how tasks are carried out in their field of work. They conduct both investigations regarding the current state of their systems and take notice of changes in regulations or by competitive vendors. They keep contact with system vendors to express Akelius' point of view on future developments of the system. Our system owner groups are:

- Accounting
- Controlling
- Customer Savings
- Debt collection
- IT platform
- Finance
- Letting and rent administration

- Maintenance
- Marketing
- Personnel

# Systems

It is a wide variety of systems at Akelius Sweden. This paper will not explain every system but instead focuses on in which context systems are used.

## Accounting and tenant administration

Accounting mainly uses the main management information system which is *3L Pro*. It is a Swedish property administration system handling both accounting and tenant administration. This system is closely integrated to our supplier invoice system, *Baltzar*, which handles about 170.000 invoices every year. Other system they are using in this department is *Norstedts tax*, tax management system; *Predator*, debt collection. In this area we have three external connections, to import tenants' payment, and a printing service that prints and mails all our rent notices. Finally, we have a connection to an external debt collection service.

## Controlling

Controlling is mainly performed in the *ADW*-system. The system gets information from 3L about all rents, vacancies and costs. The system provides a wide scope of reports for use.

## Finance

Finance is mostly using the application CRM.  
Customer Savings

## Personnel

*Hogia* is our personnel system. We are about to implement electronic time sheets in this system, testing are being conducted at one branch office right now.

## Letting

*Husar letting system* is our main system for letting. Husar offers a highly automated marketing and tenant selection system. All interests are ranked according to the companies letting policy, automatic payment history is provided to our personnel handling the letting. This system is integrated with 3L which handles all financial relations with our tenants. This area has two external services attached to our internal systems, Megasol which is the system handling Customer Savings and service providing financial reports on conceivable tenants.

## Property maintenance

*Husar maintenance administration system* is the main system in this area to keep track of all the properties technical standards. Soon, it will also manage the work orders we send to our contractors.

We also have a couple of other systems in this field. *Ess200* which is an energy consumption management system. *Aptus*, *RCO*, *QT-system* are all electronic admission management systems. *KeyControl* and *Assa Performer* which are apartment key management systems. We also use a couple of other system to manage diverse IT systems, for example ventilation, in our properties around Sweden: *Diana Control*, *Bastec* and *T.A.C.*

## Property valuation

To value our properties *ADW* is used, for this purpose *ADW* gains information about all contracts, rent, vacancies from *3L Pro*.

## IT platform

This is the foundation of all systems: servers, printers, PC:s data communication and *Microsoft Office* are placed here to serve all other systems.

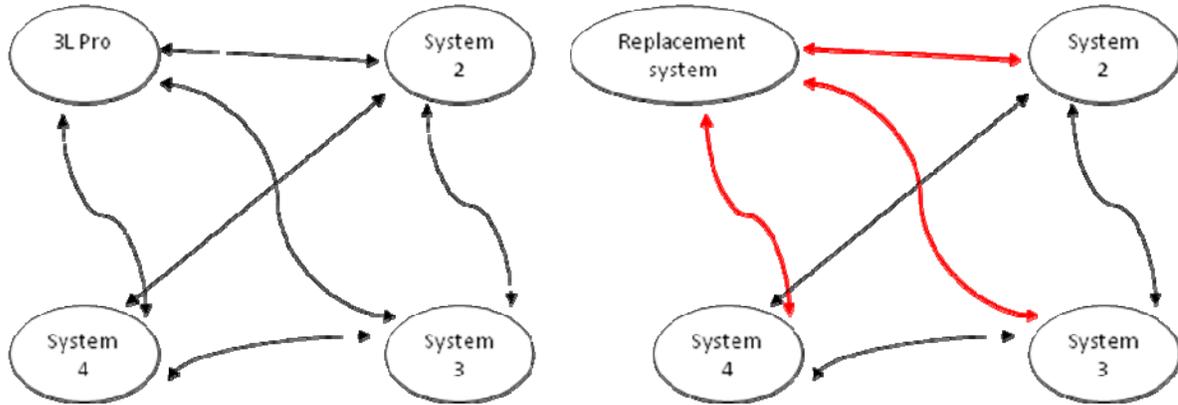
# The Future IT

A traditional IT environment consists of wide spectra of systems. To be able to reuse data the system are connected to each other by integrations. This is not a problem if there are a couple of systems. But if like Akelius, there are somewhere in the region of 20 to 30 systems it is very challenging to maintain every system if they are connected to each other.

In Akelius Sweden there are visions about integrating the whole chain from tenant's administration in *3L Pro* via *The Husar* program suite all the way down to the key management and the admission systems. However, that task would call for a lot of integration between the participating systems. Adding new integration would increase the complexity of the environment even more. Another idea is to interconnect our service order management system with our contractors who supply janitorial services. New integration today requires development performed by external consultants. This means that the complexity of our systems is growing rapidly.

## The solution

One major change in architecture is now coming – a new paradigm. This like so much else in IT, is spellt with a three letter acronym - SOA. SOA stands for Service-oriented



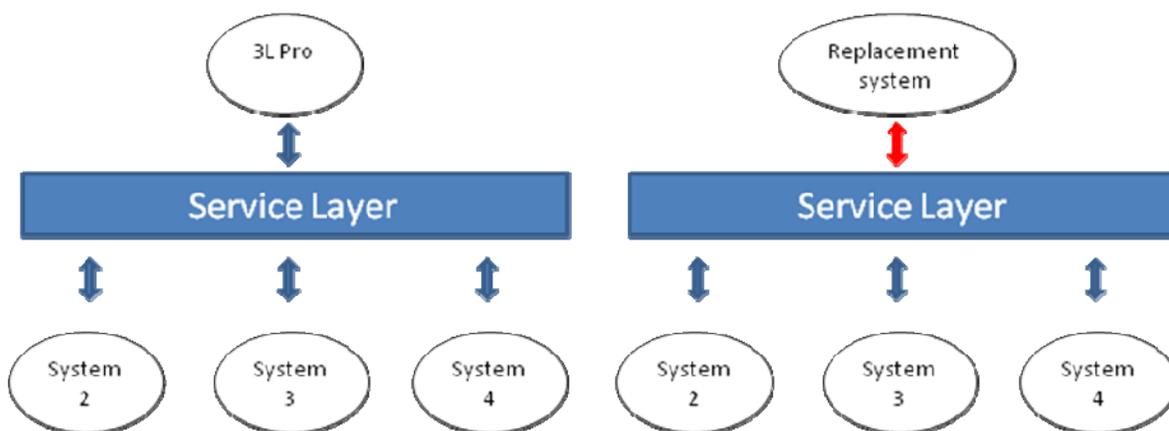
Architecture and represents a whole new way of building an IT environment.

**Illustration 1: A traditional system integration schedule – the spaghetti.**

### SOA or the Spaghetti Incident?

Today, most of our applications are stand alone applications. Every application keeps their own set of data. Most integration are done by users who copy-paste data between applications. This is a really bad idea in many ways. The most obvious is of course two things: It is very expensive to use humans to copy-paste and second, humans do make errors, one misplaced digit can be catastrophic.

What about the pasta? – traditional tightly coupled integration connecting two stand alone systems is in IT slang also called spaghetti integration due to the fact that a system schedulereminds us very much of a plate of spaghetti, see illustration 1. This illustration also demonstrates a major disadvantage with traditional integration, a change of one system may affect three different connections. That is expensive and demands a lot of testing to ensure that none of the remaining systems are affected if one system is replaced or upgraded.



**Illustration 2: The SOA approach. Since we are using SOA we only have to change the way the replacement system communicates with the service it is supporting**

OK, let's skip the spaghetti – what to do?  
SOA is a new way of thinking. Instead of interconnecting system, all systems offer services. A service is set of defined questions and answers that systems interchange.

Example: *3L Pro* should offer every other system services. It announces to everybody that

“OK, I know every tenant at Akelius, I know their rent, I know every apartment and where they are situated. Just go ahead and ask me!”

In this example every system which would like to know something about this asks a question, and *3L* will answer. It's like if you lost an invoice. Maybe you call the company who sent you it to get a copy. They respond by sending you a copy of your invoice. Nothing more, nothing less. On this copy you can find out how much you are supposed to pay, when and by which way. You also find additional information like the company address and phone number etc. but since you do not need it you just disregard it. The same way we define a standardized question which many systems may use, they just disregard the superfluous data.

This architecture has three major advantages:  
Firstly, if we need to upgrade *3L Pro* no other system needs to take notice, as long as we make sure that the answer to every defined question remains the same. Secondly, if we are getting a new system that needs the same data, we just need to tell the new system who and how to ask. This dramatically reduced implementation time and costs for every new system. Finally, let us say we want to replace *3L Pro*? Of course that would mean a lot of work! But at least we do not have to worry about every other system, as long as we make sure the new system answers in the same way as the old system would have.

Are there really no disadvantages with SOA?  
Of course there are disadvantages with this manner of building an environment. Criticisms often use a couple of points:  
Firstly, in a very small environment it might be more complex than a traditional integration. If there are only two systems exchanging information there would in fact be three different systems to maintain instead of two using a separate service layer.

A second problem that is discussed is that SOA may use more network traffic, and therefore need more powerful servers and faster network equipment to exchange the same amount of usable data. But that is all as a result of implementation, there are several different ways to implement SOA.

## Conclusion

A modern IT environment contains a lot of systems. Nothing suggests that there will be fewer in the future. This development calls for action to ensure a robust cost effective environment in the future.

### What does SOA do?

SOA offers a way of easily using the same data set in many different applications with no or little extra demand for resources. In this way you may shorten implementation time and costs.

### What does SOA not do?

SOA does not change the way systems function. There are still the same systems doing the same benefit. SOA is not noticed by users of the system. This architecture does not change the fact that there must be an awareness about the data and where it is situated in the environment.

### Q n´ A

1. What is the major advantage with SOA compared with traditional interactions?
  - a. A You will not need any IT professionals
  - b. You may change one single system more easily
  - c. You get new features in your applications
  - i. Correct B
2. What is the name of the main management information system at Akelius Sweden?
  - a. 3L Pro
  - b. Husar
  - c. ADW
  - i. Correct a
3. What does the acronym SOA stands for?
  - a. Software Open Architecture
  - b. Services Over Air
  - c. Service-Oriented Architecture
  - i. Correct C
4. How does Akelius Sweden market available apartments?
  - a. By posting notes at the local supermarket
  - b. By a highly automated internet publishing system
  - c. By using a marketing Agency

- i. Correct B
- 5. Which statement is wrong?
  - a. The system Baltzar helps our tenants pay their rent.
  - b. The system ADW is used then valuating properties
  - c. Baltzar handles about 170.000 invoices every year
  - i. Answer a
- 6. Which statement is Wrong?
  - a. With SOA, You still need to track which system keep the desired data.
  - b. With SOA, you always get a less complex system
  - c. With SOA, you may get a more robust environment
  - i. Answer B
- 7. When talking about spaghetti and integration, are you talking about:
  - a. You should never mix spaghetti and macaroni
  - b. The reality of many complex IT environments
  - c.
  - i. Answer b
- 8. What does not belong to the IT-platform
  - a. MS Excel
  - b. Data communications
  - c. Aptus
  - i. Answer C
- 9. What is the major disadvantage against integrating more systems?
  - a. The tenants do not want us to
  - b. The complexity is growing fast
  - c. We do not have enough capable servers.
  - i. Answer B
- 10. What is IT departments responsibility at Akelius?
  - a. Rule how everybody in the company is supposed to work.
  - b. Make sure that everybody's paycheck is on time every month
  - c. Carrying out changes decided by the IT management board or the system owners.
  - i. Answer C