Tekijä(t):	Nummenmaa, Jyrki; Berki, Eleni; Hemming, Jyrki
Nimeke:	Academic Mindtrek Conference 2014 Open Source Software Liferay Solutions- E-Banking Account and MQ File System
Vuosi:	2014
Sivumäärä:	20
Tieteenala:	Sähkö-, automaatio- ja tietoliikennetekniikka, elektroniikka
Yksikkö:	Informaatiotieteiden yksikkö
Julkaisun tyyppi:	Erillisteos
Kieli:	en
URN:	URN:NBN:fi:uta-201408072033

Academic Mindtrek Conference 2014

Open Source Software Liferay Solutions

E-Banking Account and MQ File System

This ‘s an Academic Mindtrek Conference 2014 study work and there is provided an overview of the Java Message Service (JMS) with core banking system and offers the basics for understand programs that use it. This study is for CIS center manager PhD Jyrki Nummenmaa, researcher Eleni Berki SQM Software Quality Management Manager.

Author: PhD Prof Jyrki Nummenmaa CIS Manager, Eleni Berki, Researcher SQM 2014 Manager, Jyrki Hemming PhD studier, Research School Of UTA.

Abstract

Internet banking enables customers to save time, take control of their personal finances and even help the environment by opting to receive electronic statements. For Internet banking users, online banking services is the third most important driver of financial institution selection, falling just behind rates and fees, and also for better customer services. These days, it’s not good enough to simply offer online banking services, however. To maintain existing customers and attract new ones, financial institutions need to keep their offerings up to date with the latest features. As the world becomes more global one it will be changed by paperless systems. The idea of a cashless society is the hope of the future. Thus more innovations will still evolve which will make cashless transactions easily accessible and affordable.

Technology has had a remarkable influence on the growth of service. In the context of the above perspective, the paper will make an attempt to analyze the evolving sphere of Internet banking and the innovations both technological and conceptual which are sweeping the financial services industry in North of Europe, and in the context of the changes that are taking place in this sector across the Europe and world.

If the flow of information could be improved, more efficient at the same time the company's operations. Portals offer a solution to the flow of information problem. The aim of the thesis was to investigate the suitability of Liferay's business environment. Early stages of research it

became clear quickly that Liferay is a very sophisticated Open-source project, which has a lot to the portal market.

Corporate Environmental compatibility was solved by using the research objectives, which aim was to investigate the ability of Liferay cope with the targets set. All Liferay performed in all the research objectives set out in an excellent manner. Although Liferay appearance of a new control panel still needs finishing, it fulfilled its set the most important task, the maintenance of a clear and simple.

On the other hand Liferay's user management is not the clearest and easiest to reach. In general, information systems management is built on a user by user role, and the grouping model. Liferay user management is implemented by using organizations, which can be connected to communities, which can also actin dependently. Communities and organizations connected to users who have different roles. Roles are much more: is an organization, community and user roles. When all roles and user groups combined can be a complex user management hierarchy, which can be Liferay advantag or weakness, depending on the environment.

Problems in the study resulted in the limited research environment and there’s has not a good documentation and UML models missing, fault or not so systematic made for this application solution. There were a lot of functions, which have not been tested in practice. Functions that were not possible to test included the sinngle sign on, integrationand customization.

Liferay development methods would also deserve their own section. Other notable problem was the search function Liferay faulty operation. The search function did not take enough into account the effective content restrictions.

As a result, the search seemed to prohibited content, and at the same time formed a security problem. Liferay problems were the lack of documentation and performance problems. These problems must be prepared.

Liferay should be reserved for powerful enough for the server. Efficient server need to be studied along with techniques for Liferay's performance can be improved. Liferay's performance is not optimized for this thesis. Documentation problems in turn, make it difficult to test. Documentation of problems because developers can to go to a lot of time understanding the structure of Liferay, and it will develop their own portlets can take a long time.

While electronic banking can provide a number of benefits for customers and new business opportunities for banks, it exacerbates traditional banking risks. Even though considerable work has been done in some countries in adapting banking and supervision regulations, continuous vigilance and revisions will be essential as the scope of e-banking increases. In particular, there is still a need to establish greater harmonization and coordination at the international level. Moreover, the ease with which capital can potentially be moved between banks and across borders in an electronic environment creates a greater sensitivity to economic policy management. To understand the impact of e-banking on the conduct of economic policy, policymakers need a solid analytical foundation. Without one, the markets will provide the answer, possibly at a high economic cost. Further research on policy-related issues in the period ahead is therefore critical.

Account with Liferay Solutions

When creating a class to model the information for a savings account in Java with Liferay Solutions, a number of special considerations come into play. The most important of these is a subtle weakness in the Java implementation of the float and double primitive data types that causes them to occasionally return slightly inaccurate answers to arithmetic operations. These inaccuracies, under most circumstances, would be trivial. However, when dealing with records of currency, where small inaccuracies can add up to dramatic, and real world consequences over time, they can become serious.

Here has been created a Bank Account Application that will allow users to do their transactions. For this, user will have to enter all the required information like, name, account number, account type and initial balance and using the switch case statement they can select the type of transaction they want to do, and they get information regarding their balance also.

In its essence, a messaging system allows separate, uncoupled applications to reliably communicate asynchronously. The messaging system architecture generally replaces the client/server model with a peer-to-peer relationship between individual components, where each peer can send and receive messages to and from other peers. Messaging systems provide a host of powerful advantages over other, more conventional distributed computing models.

Primarily, they encourage "loose coupling" between message consumers and message producers.

There is a high degree of anonymity between producer and consumer: to the message consumer, it doesn't matter who produced the message, where the producer lives on the network, or when the message was produced. This permits dynamic, reliable, and flexible systems to be built, whereby entire ensembles of sub-applications can be modified without affecting the rest of the system.

Other advantages of messaging systems include high scalability just a commercial implementations boast the ability to support tens of thousands of clients and tens of thousands of operations per second. There is an easy integration into heterogeneous networks and reliability due to lack of a single point of failure.

Because of the reliable and scalable nature of messaging systems, they are used to solve many business and computing science problems. For example, they are the basis of such diverse applications as workflow, network management, communication services that’s are voice over IP, voicemail, pager, email, customer care, weather forecasting, supply chain management, and many other systems. In addition, messaging systems are also invaluable as "glue" to bring together the disparate systems that inevitably result from mergers and acquisitions.

MQ System Design

Messaging systems are used to build highly reliable, scalable, and flexible distributed applications. This article discusses messaging systems in general, providing an overview of their features and types, and then the messaging systems are used to build highly reliable, scalable, and flexible distributed applications. In these thesis is investigated messaging systems in general, providing an overview of their features and types, and then describes how developers can write message-based applications using Java Message Service (JMS) with a portal application and finally in core banking system. Distributed applications are proliferating, as are a host of previously unexplored problems of synchronization, reliability, scalability, and security.

One solution is a Messaging System built from loosely coupled components communicating through messages. Scribes how developers can write message-based applications using Java Message Service (JMS). Distributed applications are proliferating, as are

a host of previously unexplored problems of synchronization, reliability, scalability, and security. One solution is a Messaging System built from loosely coupled components communicating through messages. The Java Message Service (JMS) with core banking system and offers the basics for understand programs that use it. JMS was developed by Sun Microsystems to provide a way for Java programs to access an enterprise messaging system, also known as message oriented middleware (MOM). MOM provides a mechanism for integrating applications in a loosely coupled, flexible manner by providing asynchronous delivery of data between applications in an indirect way through an intermediary. MQ JMS is a set of Java classes that enables JMS applications using Web Sphere MQ as the messaging provider.

The JMS API is an open standard API for sending and receiving messages. In order to illustrate a Web Service using Web Sphere MQ as the transport mechanism, it is necessary to develop some sample business functionality in order to create the service to be exposed. The main focus of this paper is the Web Banking Service infrastructure and MQ message service functionality. Therefore, the service functionality is deliberately kept simple. In this illustration, two simple classes are created to mimic some basic bank account functionality. Figure 1 illustrates the classes. Note the following points pertaining to these classes:

_ getStatement returns a complex user-defined type
_ debit throws a BankOperationException
_ BankOperation contains accessor methods in order to enable serialization.

The Web service will need a number of JMS resources or managed objects to be defined in the application server. In Figure is the topology Liferay's portal banking solution with Enterprise Architecture EA 8.0 program.

Reply-to queue: In order for this core banking system to work, the reply-to queue specified in the request message has to exist in the MQ queue manager referenced by the Reply Queue Connection Factory. Requests referencing the reply-to queue outside that Queue manager will fail at response.

- JMS Listener Port: Used by the Web service to collect the SOAP/JMS request messages from the request queue.
- The listener port is used by Glassfish Application Server runtime to pull the request queue to collect SOAP/JMS messages and pass it to activate the Web service.

- JMS Request Queue Connection Factory: Used by the listener port to connect to the MQ queue manager that hosts the request MQ queue.
- JMS Request Queue: The JMS managed object that represents the request MQ queue. It is used by the Web service to collect SOAP/JMS messages from the request queue.
- JMS Reply Queue Connection Factory: Used by the Web service to connect to the MQ queue manager that hosts the reply MQ queue.

The Core Bank JMS transport queue actions are shown in fig.8.3 with Enterprise Architecture EA 8.0 program. The quest Queue Connection Factory: Used by the listener port to connect to the MQ queue manager that hosts the request MQ queue. JMS Request Queue: The JMS managed object that represents the request MQ queue. It is used by the Web service to collect SOAP/JMS messages from the request queue. JMS Reply Queue Connection Factory: Used by the Web service to connect to the MQ queue manager that hosts the reply MQ queue. The Core Bank JMS transport queue actions are shown in fig.8.3 bellow with Enterprise Architecture EA 8.0 program.

Core Banking files system messages

The core banking system sends one of the three possible Request messages. The message contains a list of files to be uploaded from the core banking systems to the AOS web service. It is delivered to a Liferay MQ queue and read by the MDB. There are two separate use cases for the application in figure 1.

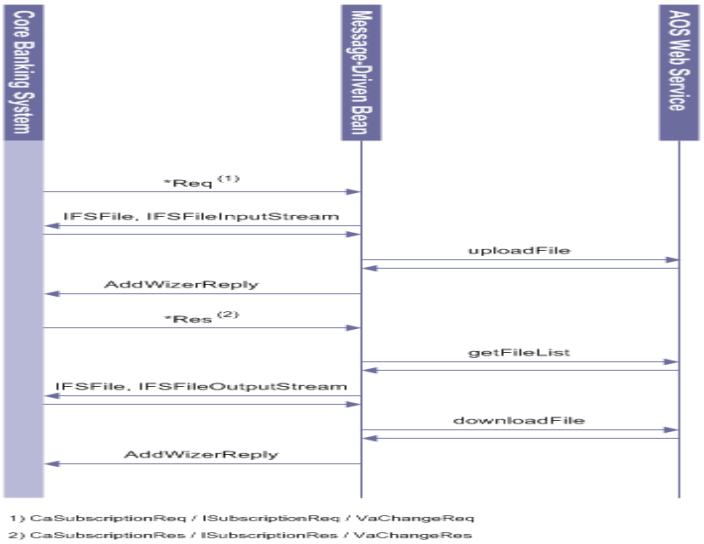

Figure 1 program calls on the core banking system on Liferay.[Liferay solution]

After a couple of minutes the core banking system will send the corresponding Response message. This instructs the MDB to download a set of reply files for the previous Request from the AOS web service and store them on the core banking system. The MDB use a method (getFileList) web service to check if any files are available, and they'll be downloaded, and stored to the core of banking system. Once it is finished it informs the core banking systems for the result by calling the reply function (ADDWIZERREPLY) program. The core banking system sends a (FileListReq) message to the Message Driven Bean (MDB). The message contains search criteria that the MDB use to query the AOS web service for files that match the criteria. The MDB informs the core banking system of the list of matching files by calling the ADDWIZERREPLY program call.

There is an investigating for consumer monitoring function in card management card solution in e-bank web pages. There find out many messages files of transactions for payment method and card applications orders. Those files find out messages date and timestamp for transaction and card orders. There find out the payment default transactions text files. Files are named with transaction date, and they are loaded under the Glassfish server to a gemalto consumer directory.

It is best for system's security that all the order's data and payments messages have to save in some tables of database. They are best stored there with a simple logical system. This payment method class calling system as in hierarchy has been modified by Enterprise Architecture EA 8.0 program.

The Payment Method Facade of a Banking System

There is a payment system in core banking CBS solution where is an invoice class and calculate the total of the amount payable but the requirement is such the amount calculation depends on the type of customer and discount offer. Also over the time the offers and discount may differ and may need to change the offer at run-time. The service is based on the reference number used in invoicing. The reference number is used to allocate the incoming payment in the invoice's accounts receivable.

The service produces separate items from the reference payments and their corrections paid to the invoice's account.

- Entry date specific sums are on the bank statement.

- The invoice updates the accounts receivable with these transactions.
- The invoice agrees on the use of the service with the account holding bank.
- Bank connection software is required to retrieve the data.

Now if have some special offer coming for the season sale all we need to do is add a new class derived from method (`Net Payable`) and then add the necessary calculation logic in that class and this way we can have the new algorithm added dynamically. Facade as the name suggests means the face of the building. The people walking past the road can only see this glass face of the building. They do not know anything about it, the wiring, the pipes and other complexities. The face hides all the complexities of the building and displays a friendly face. This is how facade pattern is used. It hides the complexities of the system and provides an interface to the client from where the client can access the system. In Java, the interface JDBC can be called a facade. We as users or clients create connection using the "java.sql.Connection" interface, the implementation of which we are not concerned about. The implementation is left to the vendor of driver.

Let's try and understand the facade pattern better using a simple example. Let's consider a stored of money in account are savings. This store has a store keeper or account owner in banking system. In the storage, there are a lot of things stored e.g. bank payments, account money and money saved. You, as client want access to different saving and credits. You do not know where the different accounts of money are stored. You just have access to store keeper who knows his store well.

Whatever there's wanted, tell the store keepers and them takes it out of store and hands it over to you on showing him the credentials. Here, the store keeper acts as the facade, as he hides the complexities of the system paying facade. The system paying facade can very well be an interface. This only returns money for some accounts. The Payment Facade is of a one type as discussed earlier in this document. Money Accounts and PaymentMoneyAccount, all these classes can implement the account interface. Similarly, the stores are of three types and can implement the Account interface. Let's have a look at the code for one of the Accounts.

If we are using a specialized Java IDE, there is probably an option to create skeleton classes automatically for this solution in the File menu. Otherwise, it's simply created a file named "SavingsAccount.java" and writed the following info in it, this class represents information common to all savings accounts:

```
public class SavingsAccount {
```

```
// Everything else in the tutorial will be written in this space!
}
```

Declare the class fields. At the minimum, you will probably want to store an account number, the current balance, a list of account holders, and a record of all transactions on the account. Writing the following inside the class:

```
/** The account number.
String accountNumber;
/** The current balance. Never use floats or doubles for currency! There are inaccuracies in float and double arithmetic.*/
private BigDecimal balance;
/**
* This stores a list of account holder names as a string. An alternative might be to create an AccountHolder class, and store that here. */
ArrayList<String> accountHolders = new ArrayList<String>();
/** A history of transactions performed upon the account. Again, an alternative would be to create a Transaction class to hold more information than the amount of the transaction */
private ArrayList<BigDecimal> transactions = new ArrayList<BigDecimal>();
```

Create a constructor. In most classes, you'll want to overload your constructor; that is, you'll want to have more than one method for building a new account instance. You can decide for yourself what sorts of constructors will suit your needs, but at the minimum, there should be a constructor for creating a new account given an account number and a starting balance, so add the following method to your class:

```
/** Create a new account
* @param accountNumber the new account's number
* @param balance the account's start balance
*/
SavingsAccount(String accountNumber, BigDecimal balance) {
this.accountNumber = accountNumber;
this.balance = balance;
}
```

Create a balance "getter" and a performTransaction method. It is standard Object Oriented Programming practice to create getters and setters for all fields in a class. However, this is

inappropriate for the balance field. While you should certainly allow the balance to be viewed, you do not want arbitrary manipulations to the balance

```
boolean performTransaction(BigDecimal amount) {
if (amount.compareTo(BigDecimal.ZERO) == -1) {
// This is a withdraw
if (amount.abs().compareTo(balance) == 1) {
// The withdraw amount is greater than the balance.
return false;
} else {
// There is enough money to cover the with-draw. Modify the balance and log the transaction.
balance = balance.add(amount);
transactions.add(amount);
return true; }
} else
{  // This is a deposit. Add to the balance and log the transaction.
balance = balance.add(amount);
transactions.add(amount);
return true; }                    }
```

This provides your savings account class with the minimum it needs to function, though there are still enhancements that can be made to it. Account holder and transactions should be their own classes, rather than simple Strings and BigDecimals, since you might want to store more information about a transaction than simply the amount involved. This is clear that the complex implementation will be done by account name class himself. The client will just access the PaymentAccount and ask for money paid for, account money or account balance. How will the client program access this façade?

The introduction

There's the balance log table is used for storing of the current balance transaction by transaction. There whore the account status also balance can be read from this table from any day or time instance. The database table ACCBALLOGT includes following data of the account status: id, balance, available balance, date of log event, credit limit.

Create the current account debit message subflow, Use the following instructions to create the current account debit message subflow. Create a new message flow is the existing SCANodesSample message flow project, create New > Message Flow. Set Message Flow Name to CurrentAccountDebit, click Finish. [SUN Microsystem 2010]

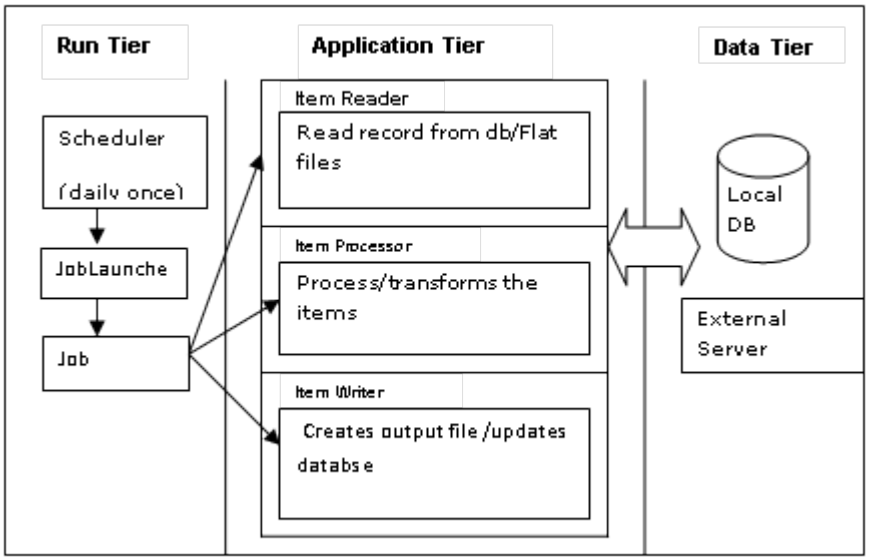

Figure 2 Three tier system architecture. [SUN Microsystem]

Reads current balance and creates a log for yesterday if nothing present. Uses a database query for making the entry, Questions: Why yesterday's balance has to be updated? Is the getting random date balance done with these data?

Concerns: This should never happen, but what happens if patch is not called in e.g. 4 days? Does the log become useless or does the balance-for-random-date functionality break? Someone can for example make a code mistake in job so that the launch always fails. Then it is recognized too late. Hypothetical is the 24/7 system, but in theory is a possible problem.

Documentation: 2011 dated technical specification document does not handle this functionality separately. Patch jobs are handled in general though. Other functions on this class

- When a transaction is processed, middle in the process the AccountBalanceLog information is also saved. Here interesting point is that how debit, credit or combo accounts are handled. Processing starts with same lines of code being executed, then the processing gets to two tracks and in the end to the database table (ACCBALLOGT) is saved again in an object of same data class at the end. Possibly this is for saving of coding work with identical tasks, but we have a concern that data between debit and

credit is not mixed. They are still so far away from each other with different accounts attached and so on.

General notes

We got to know with account balances and the difference between Balance and Available Balance. Usage of yesterday's balance triggered some discussion but it did not get 100% clear why that information is needed really. From the database schema point of view it is good that Account has the Balance separately although the same information is logged in the (AccountBalanceLog) table. It is good and logical that the account balance is found in one single location.

Difference between debit and credit is causing different codes to be done depending on which of them is in concern. Sometimes the information is 'hijacked' to the place of use through a long function call hierarchy. The difference is glued to the calls instead of that it would be stored somewhere with the card information and only read when needed. [Liferay Wiki]

Documentation of card solution:

- Card solution wiki has parts that still are 'on phase'
- Card solution other documents also suffer from the same disease: draft is the most complete state in quite many of them
- Working so far with the card solution.
- Quite many hidden information pieces needed to build up solution.
- There has been built a guide on that. Available in another document called buildme.txt.
- Should be placed on wiki sometime in future.

The reference payments and their corrections are individual transactions in assembly language data.

- Data contains reference
- Payments paid with an account transfer form
- Successful and failed debiting transactions of the direct debit service.

File Structure

The format of the class file is fully described shortly, a class file consisting:

1. Header part. The header contains a magic number, which identifies Java class files, and the version number of the format of the class file. The header part has a fixed size of 8 bytes.
2. Constant part. The constant part is a variable sized structure that holds all the symbolic information used in the rest of the file. It is from here that the loader extracts the linking information and the interpreter fetches the constants used in the byte codes.
3. Class part. The class part contains information defining the class stored in the file. This information are the access flags, the references to this, to super, to all the interfaces implemented by the class, the number of fields, the number of methods and an attribute pool. The attribute pool is a variable sized structure that holds arbitrary additional properties of the class. The size of the constant part also belongs to the class part1.
4. Field part. The field part is a table of descriptions for all the class and instance variables defined. Each entry in the table contains the access flags, the name and the type of the field and an attribute pool similar to the one in the class part.
5. Method part. The method part is a table of descriptions for the initializers, the constructors and the methods defined. Each description contains the access flags, the name and the type of the procedure and an attribute pool. The byte code for the procedure is stored in a Code attribute in the attribute pool.

For each class, we measured the sizes of these five parts and computed how many percent of the file they occupy. Table 2 presents the results computed over all the classes and table 8.2 gives the results for the individual programs.[SUN Microsystem]

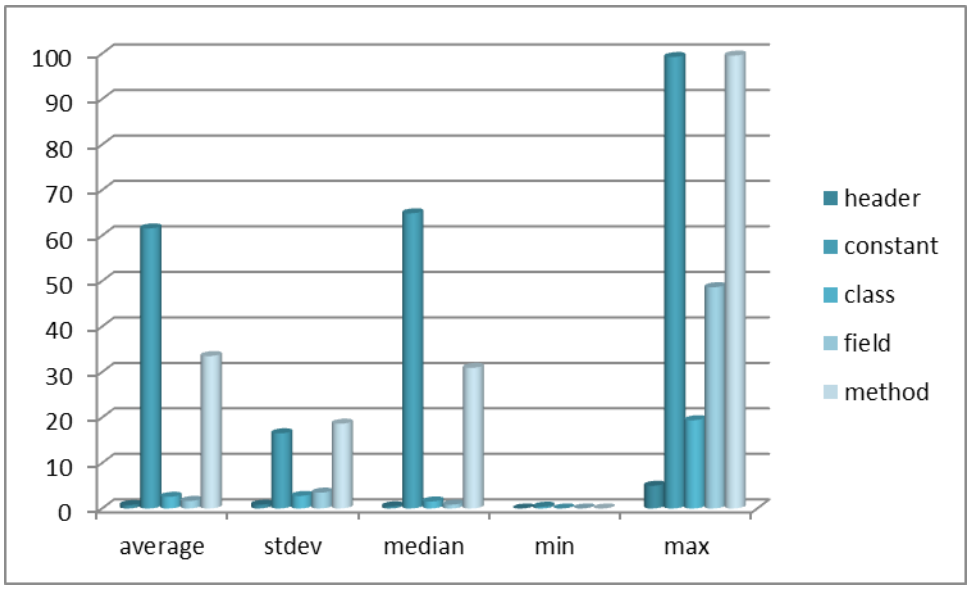

Figure 3 Summary of the repartition of the five parts of a class [% of the file size]

Sarake	averag	stdev	media	min	max
header	0,74	0,81	0,45	0.00	5
constant	61,55	16,51	64,87	0,39	99,23
class	2,55	2,74	1,53	0.00	19,39
field	1,65	3,47	0,84	0.00	48,63
method	33,49	18,62	30,89	0.00	99,54

Table 2 There's a Summary of the repartition of the five parts of a class [% file size].

The constant part and the method part make up on average 95 percent of the file size. This is not a surprise, but it is quite interesting to consider that the largest element in a class file is actually the constant part and not the method part, which contains the byte codes to execute. The constant part contains primarily the constants used in the byte codes and the symbolic information necessary to perform dynamic linking and type checking. But the constant part is accessible from all the other parts of the class file and may hence hold any other constants, as the discrepancy in the number of CONSTANT_Utf8 items suggested. We divided the entries of the constant part into three groups:

1. Constant. This group contains the CONSTANT_Integer, CONSTANT_Long, CONSTANT_Float and CONSTANT_Double as well as the CONSTANT_String entries, as they are intended to represent constants used by the bytecodes. The CONSTANT_Utf8 structures referenced by CONSTANT_String entries belong to this group.
2. Type and link. The CONSTANT_Class, CONSTANT_NameAndType, CONSTANT_Methodref, CONSTANT_InterfaceMethodref and CONSTANT_Fieldref entries constitute the second

group. They encode the type system and provide the linkage informations; with their help, the run-time sysaverage tem performs type checking and dynamic linking. The CONSTANT_Utf8 entries they reference pertain here.

3. Others. Any parts of the class file may reference the constant part, so there may be CONSTANT_Utf8s that belong neither to the constants nor to the typing and linking data; these entries build the third group.

Sun's java minimizes the size of the constant part by writing every constant once, so we had to check that the CONSTANT_Utf8 structures were only counted once. In case a CONSTANT_Utf8 structure should turn up in more than one group, it would be added preferentially to the type and link category. Even though a compiler can define an attribute that references any kind of CONSTANT structures, we did not ensure that they were actually used as expected. Of all the attributes defined, and the Exceptions attribute is the only one to reference anything other than a CONSTANT_Utf8 structure: the Exceptions attribute employs CONSTANT_Class entries to check the type of thrown exceptions; this usage conforms to our classification. Albeit this repartition appears rather conservative, it gives good practical results.

The file contains one or more batches, and a batch consists of the transactions of a single account. A batch contains a batch record, transaction record, and a sum-up record at the end of the batch. Sorting order of the transaction records:

- Invoice's account / entry date Record length is 90 characters.
- The records are separated from each other by a record separator.
- Every record ends with a carriage return and line feed character.
- The length and format of the fields have been defined in the record description.
- Leading zeros are put in the numeric field.
- Filling of an alpha numeric field is started at the left side of the field, and if necessary, the field end is filled with space characters.

Rebates can be sent from external systems, by using an internal described file (2). Consist of a file from external party:

Ssn	1 - 11	(ddmmyy-1234) (social security number)
Date	12 - 21	(25.03.2009)
Service code	22 - 24	(nnn)
Rebate.	25 - 29	(05000)50 %

Due date	30 - 39	(25.10.2009)
Spare field	40 - 40	( )

The banking system processes the file and updates the banks rebate information:

- Information is stored by service codes. New service codes are used for cards created in the card system.
- REPLICATED to the bank when created (web service) and the bank need to create the rebate file to the Card system by using card number and other needed information.
- Rebate file Bank < ---- > Card system.

The card system receives the rebate file: Updates the Card system database (6) with rebates, and the bank system using a service (web Service) (7). Function will also be used when a user makes a rebate in the Card system.

A forced system failure is induced to test a backup recovery procedure for file integrity. Inaccurate data are entered to see how the system responds in terms of error detection and protection. Related to file integrity is a test to demonstrate that data and programs are secure from unauthorized access. Usability Documentation & Procedure: The usability test verifies the user-friendly nature of the system. This relates to normal operating and error-handling procedures.

Proper documentation is must for mainframe of any software. Apart from in- line documentation while coding. Help coding, help files corresponding to each program were prepared so as to tackle the person-dependency of the existing system.

Conclusion and Future Scope of Improvement

The Banking Online System is a big and ambitious project. I am thankful for being provided this great opportunity to work on it. As already mentioned, this project has gone through extensive research work and evaluation processes. On the basis of the research work, we have successfully designed and implemented banking online card System with the Liferay banking solution. This system is based upon 3-tier client server architecture. The regulatory and taxation issues of Internet banking present formidable problems and the paper attempts to get an insight into these two important issues. The Internet banking project now wants to integrate an ethnographic approach with flexible systems methodology to focus on the communication design issues for web-portals specifically devoted to consumers. The specific research questions are under this study project has been in my mind for future research work:

- What are the major information design features for Finland and Scandinavian countries today and in future?
    - What are the major communication media tools to be used for banking websites in Finland?
    - What are the modes employed to transcend communication and cost barriers for specific user groups?
- What local language solutions are to be provided and through which media?
- What non-textual solutions can be provided for the under-educated, untrained user?
- What makes Internet banking relevant for the no served communities? What is customers' perception about Internet banking?
- What are the factors that result in the perception of the e-banking?
- How can these factors are evaluated for current banking service?
- What ways and means to be followed for increasing the usage of Internet banking in Finland and all around in Europe?

Under this study has been also for future development work in high-knowledge organizations and companies. There's commonly a problem with the information flow. It is a common problem in SW development and also in IT technology companies in Finland, but also in the financial markets solutions, and especially the high-knowledge sector companies. There it's just today as a known problem generally, but not so easy to solve problem. This is a leadership and

peoples management problem for companies. The leadership of high technology companies needed for much more resources and new education for business management.

Several tasks were set for this investigate for back office banking with a portal Liferay solution technology. In what follows has been discussed is how well these tasks were achieved, what problems and limitations of Liferay were encountered during the case of investigate, and how useful the reverse engineering techniques of Liferay solution were in practice.

Result of the core banking portal investigate case

Internet platform will be renewed for portal services or cloud services. The Liferay environment is only one solution for future finance services. The future of the Internet to research in Finland invests heavily. ICT competence center The World Wide Web Consortium (W3C) is an international community that develops open standards to ensure the long-term growth of the Web. There's launched a year ago, the Future Internet research program, whose first results are more than encouraging. Internet users and traffic volumes are exploding. Security Guide at the same time is used more and more critical tasks, and also often in mobile environment.

Some forty years ago, designed for the Internet technology platform is already in need of reform. A year ago, companies and research institutions joined forces with the developers of the future Internet requirements and characteristics.

Internet has been created for a different task. Internet was originally conceived in the 70s to support the research of information exchange and networking between researchers and trusted developers. Network users were familiar with each other and all the transport and exchange of information were necessary and desirable.

Today the web is part of the modern information society. The network is vast, and users have different estimates, billions. In addition, in the future most of them use the Internet for mobile wireless networks.

There are new uses for internet being developed, but currently there's no way to use the Internet just right way. There's is no longer able fully to meet the demands, but it has to be more and more to correct if some problems occur. The fall of internet, has been discussed for a long time, and a variety of reasons have been suggested but there is always invented ways to circumvent the problems. They may have been circumvented, or reducing their negative impact for internet users. Soon, however, we are in a situation where the basic Internet technical

architecture requires a proper renovation. Below is a list of current Internet the biggest problems and challenges of future.

Internet biggest problems:

- Undesirable traffic,
- Routing System choking,
- Mobility management and multi-network interfaces, gaps,
- Underdeveloped market mechanisms for traffic and resource management,
- Right to Privacy and the limits of accountability,
- Trust and reputation management deficiencies.

Challenges:

- Information Networking,
- Availability and usage,
- Internet socio-economics,
- Autonomy and fault tolerance,
- Energy,
- New bottlenecks.

Future of Internet portal service's development have already been widely recognized around the world and in Finland it has been selected as a priority in the spring of year 2009, launched the IT sector's strategic center. Future Internet research program was started less than a couple of years ago. There was as one of the IT cluster's research work main mode of the first projects. The program's main objective to improve the future of the Internet and technology needed to remove the final obstacles to a more flexible and efficient solution for the road. The aim was also to create a more transparent criteria, and by more powerful applications and solutions for imports.

There was the aim of the research topic is to develop a contrast to the current for the Internet, which is based on the idea that knowledge is always a physical computer. For example, when the user wants to see this day coming week, the weather, to keep Game to give the server address (Domain Name Server, DNS name or IP address), which must be retrieved from the weather. In the future, the user can simply ask the browser to weathering and the information is sought from the place where it can best be obtained. Weather may even be a combination of several different sources.

References

[Agile 2008], Agile Manifesto (2008). Manifesto for agile software development. WWW-sivusto, http://agilemanifesto.org (26.4.2008).

[Al-Hawari et.al 2005], Marketing Bulletin, 2005, 16, Article 1 Page 1 of 19 http://marketing-bulletin.massey.ac.nz Measuring Banks' Automated Service Quality: A Confirmatory Factor Analysis Approach Mohammed Al-Hawari, Nicole Hartley and Tony Ward

[Arjen tietoyhteiskunta, 2008], Ubiquitous Information Society (2008). Finnish Information Society in everyday life of progress, and future challenges. Ubiquitous Information Society Advisory Board Annual Report 2008, the State Council. Retrieved 11/29/2011 from http://www.arjentietoyhteiskunta.fi/?s=102

[Automatia, 2011], Banking Automation, cash withdrawals from ATM machines in Finland 06-09.
Source: by Automatia in Finland 2011, a banking automation Research.

[Basel Committee], That's a reported of Bank for International Settlements, Basel Committee on Banking Supervision, 2000, Electronic Banking Group Initiatives and White Papers, October (Basel) (referred, 15.05.2012).

[Briand, Langley, 2000] L.C. Briand, T. Langley, and I. Wieczorek, "A Replicated Assessment and Comparison of Common Software Cost Modeling Techniques," Proc. Int'l Conf. Software Eng. (ICSE 22), pp. 377-386,2000.
[Brown, Vessey] Brown, C.V. and Vessey, I. "ERP Implementation Approaches: Toward a Contingency Framework," Proceedings of the International Conference on Information Systems (ICIS), Charlotte, North Carolina, 1999, pp. 441-416. 52 R.L.I.M.S. Vol. 2, May 2001
[Chan] Chan Brian, http://www.sanjeevg.com/2009/03/liferay-portal-review.html [linked 02.10.2011]." Liferay Portal was first developed in the year 2000 by Brian Chan as an open source project with the objective of providing an inexpensive collaborative portal to non-profit organization".
[Chicago 2009] Ragan's annual Web Content Management Conference in Chicago 2009. The Portal products continue to be hot topics of discussion and it certainly held the attention of the 150+ attendees who listened to my keynote presentation on portals at: Portal products promise a lot.
[Colonia-Willner], Colonia-Willner, R. (2004). Self-service systems: New methodology reveals
customer real-time actions during merger. Computers in Human Behavior, 20,243–267.
[ComScore, 2012], Total usage of Internet in European countries, UK, 16 February 2012, comScore, Inc. According to the Pan European Internet Monitor by comScore, Inc. (NASDAQ: SCOR), (UK, 16 February 2012).
[Creswell] Creswell, J.W. (2008). Educational research:" Planning, conducting, and evaluating quantitative and qualitative research "(3rd). Upper Saddle River, NJ: Prentice Hall. 2008 ISBN: 10 0136135501 (pages 8-9).

[Denney 2005] Denney, Richard (2005) Succeeding with Use Cases: Working Smart to Deliver Quality. Addison-Wesley Professional Publishing. ISBN Discusses the use of software reliability engineering in use case driven software development.
[Durkin 2004] Durkin, M. (2004). In Search of the Internet Banking Customer, Exploring the Use of Decision Styles. International Journal of Bank Marketing, 22(7), 484–523.
[Eclipse 2011] Eclipse modeling tool project (http://www.eclipse.org/modeling/) Referred 31.11.2011.
[Electronics Engineer], Institute of Electrical and Electronics Engineers (1990) IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries. New York, NY ISBN 1-55937-079-3
[EU Comission 2010] Communication from the Commission of 1 June 2005 to the Council, the European Parliament, the European Economic and Social Committee and the Committee of the Regions entitled "i2010 - A European Information Society for growth and employment"
[Filotto et.al] Filotto, U., Tanzi, P. M., & Saita, F. (1997). Customer Needs and Front Office Technology Adoption. International Journal of Bank Marketing, 15(1), 13.

[Financial Statistic Finland 2009a], Finanssialan Keskusliitto (2009a). Nuorten rahankäyttötutkimus. Tutkimusraportti Kevät 2009. Referred 2.11.2011 https://www.fkl.fi/www/page/fk_www_3769
[Financial Statistic Finland 2009b], Finanssialan Keskusliitto (2009b). Pankit Suomessa 2009. Referred 5.1.2012 https://www.fkl.fi/www/page/fk_www_6151
[Financial Statistic Finland 2010a], Finanssialan Keskusliitto (2010a). Senior raport. Raport fron June 2010. Referred2.11.2010 https://www.fkl.fi/www/page/fk_www_3770

[Fitzgerald, B. (2006)], The Transformation of Open Source Software. MIS Quarterly, Vol. 30,
No. 3, pp. 587-598.
[Fitzgerald, B. (2009)] Open Source Software Adoption: Anatomy of Success and Failure.
International Journal of Open Source Software & Processes, Vol. 1, No. 1, pp. 1-23.

[Flyvbjerg] Flyvbjerg Bent, 2006, "Five Misunderstandings about Case Study Research." Qualitative Inquiry, vol. 12, no. 2, April, pp. 219-245.
[ForeSee Results 2011], this is data and analysis related to the U.S. 2011 ForeSee Results Online Banking Study. A UK edition of this research will be released May 25 at www.ForeSeeResults.com.
[M. Fowler] M. Fowler et. al., "Refactoring: Improving the Design of Existing Code", Addison-Wesley, 1999.
[Gauch, Jr., H.G. (2003)], Scientific method in practice. Cambridge, UK: Cambridge University Press. 2003 ISBN: 0 521 81689 0 (page 3).
[Glass] Glass R., "Maintenance: Less is not More", IEEE Software July/August 1998
[Greenfield, 2004], Greenfield J. & Short K. (2004) Software Factories: Assembling Applications is With Patterns, Models, Frameworks and Tools. Wiley Publishing, Inc., Indianapolis, USA. 666 p.
[Gregor, S. (2006)]. The Nature of Theory in Information Systems. MIS Quarterly, Vol. 30, No. 3,
pp. 611-642.

[Haasio 2008], The Social media of the Internet for interactive group work and publishing systems is a typical for social media systems, and that's participants can produce content, edit, comment, discuss, share information and network with each other and this is evident by Ari Minna Haasio consoles on the virtual stream BTJ Publishing, 2008. In this study there is a criterion to guide information retrieval and social media for teaching.

[Haikala I, 2004], Haikala, I & Märijärvi J., P. 2004. Software Engineering. Hämeenlinna: Talentum Media Oy.
[Hammond], Kathy Hammond, Gil McWilliam, and Andrea Narholz Diaz (1998),"Fun and Work on the Web: Differences in Attitudes between Novices and Experienced Users", this research explores the differences between novice and more experienced Web users and their appreciation of the Web's entertainment and informational value. A two-stage study with the same subjects after a 4-month interval was conducted. Our findings show that while prior experience is an important moderator of users' attitudes towards the Web, its influence is not linear. The heaviest users are enthusiasts for the medium, while moderate and light users perceive it as a source of information, but not for entertainment or fun. If this finding generalizes, it will have a significant impact on the growth and development of the Web as a mass medium. in Advances in Consumer Research Volume 25, eds. Joseph W. Alba & J. Wesley Hutchinson, Advances in, Pages: 372-378.

[Hatcher & Loughran] Java Development with Ant by Erik Hatcher and Steve Loughran (Manning, 2002).
[Hale, Householder, Greene 2003], Hale, J.L.; Householder, B.J. & Greene, K.L. (2003). The theory of reasoned action. In J.P. Dillard & M. Pfau (Eds.), The persuasion handbook: Developments in theory and practice (pp. 259–286). Thousand Oaks, CA: Sage.
[von Hippel, E., & von Krogh, G. (2006)], Open Source Software and the 'Private-Collective' Innovation Model: Issues for Organization Science. Organization Science, 14(2), 209-223.
[Hughes, T. (2001)], Market orientation and the response of UK financial services companies to changes in market conditions as a result of e-commerce, International Journal of Bank Marketing, Vol. 19 No. 6, pp. 222-231
International Data Corporation (2005). Available http://www.epaynews.com/statistics/bankstats.html#15

[IBM WaterFall] Transitioning from waterfall to iterative development
Per Kroll, Manager of Methods, IBM Rational, IBM, Software Group
http://www.ibm.com/developerworks/rational/library/4243.html, Date: 16 Apr 2004

[IEEE standard], IEEE 1471-2000, Recommended Practice for Architectural Description for Software-Intensive Systems, Project scope: The purpose of this standard is to facilitate the expression and communication of architectures and thereby lay a foundation for quality and cost gains through standardization of elements and practices for architectural description. Despite significant efforts to improve engineering practices and technologies, software-intensive systems continue to present formidable risks and difficulties in their design, construction, deployment and evolution, http://standards.ieee.org/cgi-bin/status?1471-2000 (referred 31.12.2011)

[Internet Economy Indicators 2000] Internet WWW page, available at URL:
http://www.internetindicators.com/global.html. Version current as of September 17, 2011.

[Internet Banking Journal 2006], in Internet purchase, the Journal of Internet Banking and Commerce, Financial institutions offer new banking channels to their customers, as technology adds new dimensions to the classic banking systems. One of the most popular self-service banking technologies is Internet banking. This study examines how young consumers perceive Internet banking in relation to other six banking channels to brick and mortar, automated teller machine (ATM), phone banking, wireless application protocol (WAP), electronic fund transfer at point of sale (EFTPOS), and bank branches in stores, 2006.

[Janus Boye] in CMS Watch's 2nd Edition of the Enterprise Portals Report released in October (2009).“Enterprises painfully discover that portal solutions are much less out-of-the-box than expected. Many face challenges due to misaligned requirements, underestimated implementation costs, or overestimated user adoption rates,”
[Jayawardhena], Chanaka Jayawardhena, Paul Foley, (2000) "Changes in the banking sector – the case of Internet banking in the UK", Internet Research, Vol. 10 Iss: 1, pp.
[Javelin] Javelin Strategy & Research's 2007 Online Banking and Bill Payment: Trends, Forecasts, and Strategies for Reinvigorating Growth and Adoption, http://www.javelinstrategy.com/.. F_2007 Online Bankingand Bill Payment_Brochure.pdf.
[Journal of Internet Banking], There's a banking services survey, 'Innovated Banking Services Provided by your bank is very good' by Journal of Internet Banking and Commerce. There's an Innovated Technology in Banking Services Journal of Internet Banking and Commerce, August 2011, vol. 16, no.2.
[Journal Economics], Josh Lerner and Jean Tirole, “Some Simple Economics of Open Source,” Journal of Industrial Economics, Vol. 50:2, forthcoming June 2002;
[Juniper] Juniper research articles on web juniper web publications, http://juniperresearch.com/networks_and_technologies.
[Jarvenpaa, Todd], Jarvenpaa, S.L. and Todd, P.A. (1977), “Is there a future of retailing on the Internet?” in Peterson, R.A. (Ed.), Electronic Marketing and the Consumer, Sage Publications,
Thousand Oaks, CA., pp. 139-154.
[Järvinen ja Järvinen, 2000] Pertti Järvinen ja Annikki Järvinen, *Tutkimustyön metodeista*, Opinpajan kirja, Tampere 2000.
[Kanninen] Liisa Kanniainen. Mobile Banking Services: Mobey Forum and Europe, April
2010. Survey.
[Kangassalo, 1993] Due to the IS-A relation, it is natural to have one concept that is contained in every concept. If the containment relation is applied to other sorts of containment e.g. whole-part, this study has need to employ another graphical method, the one that is presented. There, the most abstract concept is depicted above all the other concepts.
[Karanta], Karanta I. Methods and problems of software reliability estimation. VTT WP 63, 2006. 57 p.
[Karjaluoto, H. 2001] Measuring attitudes towards Internet banking: Empirical evidence from Finland. Proceedings of the European Marketing Academy Conference. Bergen, Norway, May 8-11, 2001.
[Karjaluoto, H. & Mattila, M. 2001] Security challenges of Internet Banking – consumers' opinion of security. University of Jyväskylä (not published).
[Karjaluoto, H., Mattila, M. & Pento, T. 2002a] Electronic banking in Finland –
consumer beliefs and reactions to a new delivery channel. Journal of Financial Services Marketing 6 (4), forthcoming.
[Karjaluoto, H., Mattila, M. & Pento, T. 2002b.] Factors underlying attitude formation toward Internet banking. Paper under review with the European Journal of Marketing.

[Kettunen], Kettunen, Jari, Simons, Magnus, Toiminnanohjausjärjestelmän käyttöönotto pk-yrityksessä - teknologialähtöisestä ajattelusta kohti tiedon ja osaamisen hallintaa, Valtion teknillinen tutkimuskeskus, In this issue the Information systems and their use in research can be seen in this article essence of the construction and operation of information systems for a solution and methodological development work. Against information systems research is in front of the same problems as the rest of the research activities in general. Information systems are

complex and their design and use is based, as well as generally through complex organizations. In practical terms, there is always a question of complex strategic issues, technical and economic issues, and organizational solutions that cannot be solved merely empirical level. Espoo, 2001, <http://www.vtt.fi/inf/pdf/julkaisut/2001/J854.pdf>.
[Koch, 2005], Koch, A. S. (2005). Agile Software Development: Evaluating the Methods for Your Organization. Artech House, Boston.

[Koskimies, 2006] Koskimies, K. & T. Mikkonen, P. 2005. Software architectures. New York: Talentum Media Ltd

[Kuisma et.al] Kuisma, T., Laukkanen, T., & Hiltunen, M. (2007). Mapping the reasons for resistance to internet banking: A means-end approach. International Journal of Information Management, 27(2), 75–85.

[Kuusela, H., Spence, M. T. and Kanto, A. J.], Expertise effects on prochoice decision processes and final outcomes, it's a protocol analysis, European Journal of Marketing, Data were collected by a questionnaire administered over three days at a department store that generates the second largest turnover in Finland. A total of 364 shoppers completed the questionnaire. Findings was empirical evidence supports our tripartite conceptualization of total customer value. In particular, social value is an independent construct. Further, social value varies by day-of-week, with a significant increase on Saturday versus weekdays, when the store is more crowded, whereas no such differences in utilitarian and hedonic values were detected Vol. 32 No. 5/6, pp. 559-576.

[Liferay, 2011] Liferay Portal is not only a portal platform. It also provides a 60+ applications and tools included in the form of portlets. These portlets are available out of the box or in plugins. http://www.liferay.com/web/guest/community/wiki/-/wiki/Main/Liferay+Portlets[referred, 02.10.2011]
[Liferay, Wiki] Liferay Wiki Guide, an online collection of Liferay knowledge and know-how. Articles about how to use Liferay Portal: http://www.liferay.com/community/wiki/-/wiki/
[Referred, 24.05.2011]

Lintula, J. (2009). E ousted ATMs. Central Finnish, 09/12/2009. Retrieved 01/04/2011 from http://www.ksml.fi/uutiset/kotimaa/verkkopankki-syrjäyttituulikaappin. dispensers, payment machines, operating is very few in future.
[Lintula, 2009], Electrification of banking has changed sector banks significantly and created a new, low-cost supply channel services.
[Liukkonen] Liukkonen Marjaana, Aalto University, Helsinki. A Survey study work. http://www.soberit.hut.fi/t-121/shared/thesis/kandityot/kandi_marjaana_liukkonen.pdf.

[MathWorld 2009], Chi-squared distribution, from MathWorld, retrieved Feb. 11, 2009 Weisstein, Eric W. "Chi-Squared Distribution." From MathWorld--A Wolfram Web Resource. http://mathworld.wolfram.com/Chi-SquaredDistribution.html.

[MetaEdit 2011], Finnish Company named as MetaCase, http://www.metacase.com/ there's a MetaEdit program, it's a tool for designing your modeling language: its concepts, rules, notations and generators. The language definition is stored as a metamodel in the MetaEdit+ repository. . Referred 31.11.2011
[NERA project], Evans is with NERA Project in Cambridge and the Center for the New Europe in Brussels, Belgium. Reddy is with NERA Project in Cambridge. In this publishing there's a grateful for financial support for our research from Microsoft, and also thank Robert Hahn and Anne Layne-Farrar for helpful comments and James Hunter, Bryan Martin-Keating, and Irina Danilkina for exceptional research assistance.
[Niemeyer,Poteet] Extreme Programming with Ant by Glenn Niemeyer and Jeremy Poteet (Sam's, 2003).
[NIST (2006)], Engineering Statistics Handbook - Chi-Squared Distribution, NIST/SEMATECH e-Handbook of Statistical Methods, http://www.itl.nist.gov/div898/handbook/, date. A significant update was made to the Handbook April, 2012.

[Nugroho Ariadi], Nugroho, Ariadi:"The effects of UML modeling on the quality of software", Phd thesis of Leiden Institute of Advanced Computer Science (LIACS), Department of Mathematics and Natural Sciences, Faculty of Science, Leiden University Issue Date: 2010-10-21. https://openaccess.leidenuniv.nl/handle/1887/16070

[Ollila, 2008], Elina M. I. Ollila, Riku Suomela, Jussi Holopainen, Using Prototypes in Early Pervasive Game Development, Computers in Entertainment (CIE) – Theoretical and Practical Computer Applications in Entertainment, Volume 6 Issue 2, April/June 2008, ACM New York, NY, USA, 2008

[OMG Architecture], OMG. (Accessed: 27.12.2011) Model Driven Architecture (MDA) FAQ. URL: http://www.omg.org/mda/faq_mda.htm.

[Palomäki Jari], Jari Palomäki, From Concepts to Concept Theory: Discoveries, Connections, and Results. I read this release of the request for such information and ontological doctrinal questions are what Palomäki calls the "theory of concepts" ("theory of concepts"). That, he said, "a number of statements relating to the concepts of the nature and role, and it aims to answer the question, 'What concepts are?'". Palomäki to distinguish the concepts of the theory of "the concept of the theory" ("concept theory "). This latter provides a formal and axiomatic concepts" representation ". Acta Universitatis Tamperensis, ser A, vol. 416, Tampere 1994. 127 s.

Mika Pantzar 15.3.2013Towards rhythm-based service economy: preliminary outlines
(Keynote paper presented at the 2nd Nordic Conference on Consumer Research,Göteborg May 2013.

[Pantzar, M. 1996] Kuinka teknologia kesytetään: kulutuksen tieteestä kulutuksen taiteeseen. Pantzar includes these study assets in isolation, however, the technological background. This is a picture of "loose" new assets, in which people have reacted with varying degrees of enthusiasm. Change in attitude is not necessarily exposed to the product itself, but its supporting system. For example, to obtain the phone was quite natural after the phone company had pulled into the house, the village, district or telephone cables. Pantzar is, however, very close to Hughes's view when he says that "the analytical mind can be separated into three parallel product domestication dimensions: technological change domesticated, or how the product will change the product life cycle time, social change tame or how innovations spread in social interactions in the structure and the need for a dynamic change in domestic or how an individual's choice of motives changing products stabilizes, Helsinki: Tammi.

[Pantzar, M., & E. Shove, 2010]. Time in practice - Discussing rhythms of practice complexes. Ethnologia Europaea. Journal of European Ethnology, Forthcoming.

[Pantzar, M. (2011)] Asiakkaan aika ja talouden rytmiliike. Helsinki: Tehokkaan Tuotannon Tutkimussäätiö.

[Statistics Finlanfd 2009], Tilastokeskus (2009). Internetin käyttö on yleistä ja arkista. Referrred 12.11.2011 by http://tilastokeskus.fi/artikkelit/2009/art_2009-09-30_007.html

[Statistics Finlanfd 2010], Tilastokeskus (2010). Tieto- ja viestintätekniikan käyttö 2010. Referred 12.11.2011 by http://www.stat.fi/til/sutivi/2010/sutivi_2010_2010-10-26_tie_001_fi.html

[STT. 26.3.2011], Huono tiedonkulku yrityksissä hidastaa kehitystä. Helsingin Sanomat. [Net Press article]. [referred: 29.3.2011].http://www.hs.fi/talous/artikkeli/Huono+tiedonkulku+yrityksissä+ hidastaa+kehitystä/113525497093

[SYTYKE] Sytyke ry - Systeemityö http://www.pcuf.fi/sytyke, Referred 8/2012. Sytyke ry links Finnish Association of Information Professionals business technical experts. Current software engineering topics, exchange of ideas and learning professionals, hype common sense. Software engineering or information system of work includes the organizations' individual needs based on the analysis of information systems design and development, which aims to automate the method suitable for the implementation of the processes by means of software.

[Thomas] Thomas G. (2011) typologies for the case study in social science following a review of definition, discourse and structure. Qualitative Inquiry, 17, 6, 511-521

[Tuxedo], Eric Raymond, “The Cathedral and the Bazaar,” http://www.tuxedo.org/~esr/writings/cathedral-bazaar/ (Referred April 2, 2012).

[Vokolos98] Performance testing of software systems; Filippos I.Vokolos, and Elaine J.Weyuker; Proceedings of the first international workshop on Software and performance , 1998, Pages 80 – 87 Performance testing papers are very rare in literature, although performance is what most people care about. This paper gives a discussion on approaches to software performance testing. A case study describing the experience of using these approaches for testing the performance of a system used a s a gateway in a large industrial client/server transaction processing application is presented. It is not a very representative paper, though.

[World Bank, paper no.431]"Electronic Finance. A New Approach to Financial Sector Development?" World Bank Discussion Paper No. 431 (Washington).