

Reuse Repository

Ewa Stemposz, Alina Stasiecka, Kazimierz Subieta
Polish-Japanese Institute of Information Technology, Warsaw
Institute of Computer Science PAS, Warsaw
{ewag, alas, subieta}@ipipan.waw.pl

Abstract. The essence of reuse is using some product of human activity to support production of other products. Reuse can significantly improve quality of software products, reduce development costs and shorten development time. Reuse is unfortunately connected with serious socio-psychological problems, called in the literature “cultural obstacles to reuse”. In the project devoted to the Reuse Repository system we have tried to achieve two goals: to provide physical storage and retrieval mechanisms for reusable assets, and to provide simple mechanisms to support the formation of the reuse culture in the software company. The system is fully implemented and experimentally tested. The paper discusses its basic goals, issues, assumptions, architecture and solutions.

1. Introduction

The term *reuse* was coined to denote the practice in the software development when some piece of one process, documentation or product is used to make subsequent products. Reuse has the potential to reduce cost, increase quality and shorten time of application development. Reuse makes sense because the similarity found across software systems is significant. It is usually found that 60-70% of one development activity is common to a next activity. From this point of view reuse can be promoted as a productivity and quality enhancement. As it is observed nowadays, software reuse is taking more attention in such domains in which the competition is strong and where the cost pressure is increasing (e.g. telecommunication).

The problem of reuse is recognized for more than thirty years. Today’s strategies are focused on earlier phases of the life cycle: the requirement specification, analysis and design. As it has been proven by years of experience such a strategy results in the greatest return on investment. Usually, when products of earlier phases are reused, there is also reuse on further phases (implementation, testing, etc.). Another perspective is the reuse of expert knowledge. Knowledge reuse assets, called patterns, could be shared with novices in order to improve their performance. Patterns have been proven to be beneficial as a cost-effective means to write better software. Technologies, in particular object-oriented methodologies, languages and tools can significantly support the reuse practice.

It has turned out, however, that technology itself is not sufficient to achieve the long-seen goals of widespread reuse. Reuse rarely appears as a side effect of normal software development practice. Many researchers noticed critical obstacles in introducing the reuse technology into an organization, which are commonly referred to as cultural barriers to software reuse. An important question concerns how to make reuse an integral cultural part of the software engineering environment. This requires systematic approach to reuse in an organization, which must involve the key factors of success: the structure of an organization, development practice, management, education and promotion.

In this paper we present a Reuse Repository (RR) system designed and implemented in the Polish-Japanese Institute of Information Technologies. The goal of RR is twofold. Firstly, the system presents a physical repository of reusable assets that can occur in a software company. Secondly, the system has been designed to support establishing the culture of reuse, in which various developers have to be

directly motivated to use the reuse technology in their everyday practice. While the first goal is common to various software libraries (see e.g. [Con93, Ed99]), the second goal presents many unexplored issues, which combine psychological, organizational, economical and technical aspects. The presented system is the result of some hypothesis concerning how to support reuse by some direct or indirect profits for creators and users of reusable assets.

The rest of the paper is organized as follows. In Section 2 we present the requirements that we collected for the system that has to support reuse. Section 3 discusses the management of reusable assets. In Section 4 we discuss how to make foundation for formation of the reuse culture. Section 5 presents the summary and conclusion.

2. Requirements for the Reuse Repository

The design of RR we started from market exploration. The project was not supposed to be dedicated to some concrete client. Our intention was to make it fairly generic. During our exploration we used specially prepared questionnaires. We asked thirty persons of various age, various levels of education, having various jobs (not only connected with software) working at companies of various range of activities.

We made our inquiries among people for whom computers mean a kind of tool in their profession. Some questions were not clear - at least inexact - for a lot of investigated persons, some of them had difficulties with understanding the essence of questions and a lot of effort was needed for additional explanations to get concrete answers in the end. It was evident, our poor experience in such investigations was the reason of that, but only partly. We observed some symptomatic phenomenon - much greater difficulties had people from outside information technology trade, but those connected also needed some explanations from time to time.

On the basis of gathered information we formulated the following conclusions:

- The majority of inquired individuals work in Windows'95 and Windows NT environments. Considerable part of them use a variety of assets in their professional work so they already have some experience in reuse. Even those not dealing with reuse so far have seen the necessity of systematic approach to it. They appreciated the assets management system as very useful. They also presented an opinion that the reuse technology will be more efficient if it can facilitate everyday work. Most of them have also seen the necessity of creating versions of assets, to support their application in changing situations.
- The following groups of assets (categories) were specified by inquired individuals:
 - ❖ templates for CVs, motivation letters, contracts, information exchange documents (e.g. with banks), various kind of documentation (e.g. describing some studies or investigations);
 - ❖ advertisement designs, graphical templates;
 - ❖ presentation layouts, styles and fashions for products' descriptions;
 - ❖ procedures for contract preparing, information exchange, market exploration and the like;
 - ❖ procedures for psychological or social diagnosis executions, templates for psychological tests;
 - ❖ procedures for different but typical activities met in every day life (e.g. how to behave when applying for a new job);

- ❖ sets of characters for text editors;
- ❖ software functions libraries;
- ❖ frameworks for data bases, subsystems and entire applications;
- ❖ information about tools useful in information technology (including description of their deficiencies and “recipes” how to correctly use them);
- ❖ useful addresses (also URL addresses);
- ❖ dictionaries and thesauri;
- ❖ ready to use information bases.

It was turned out that the inquired people used mostly that group of assets which can be qualified as the text file category.

- We got a variety of answers to the question concerning a preferred kind of information model, which could facilitate the process of finding and recognizing useful assets. The most frequent suggestion was to use a short functional description possibly expanded with the asset creation date. It was important to have the possibility of finding assets through key words.
- Specifying the method of asset access many inquired persons voted after using Internet. They wanted to have the possibility to find assets by themselves or to be informed automatically about selected categories of assets.
- Concerning how to gratify assets authors there were different opinions; the most popular one was based on some financial bonus.
- Concerning asset/author/user associations there were diversity of opinions. Some of the inquired persons didn't see a need of dealing with that problem at all. Many of them were of the opinion that it would be valuable to pass remarks about an asset virtues and drawbacks to its author.

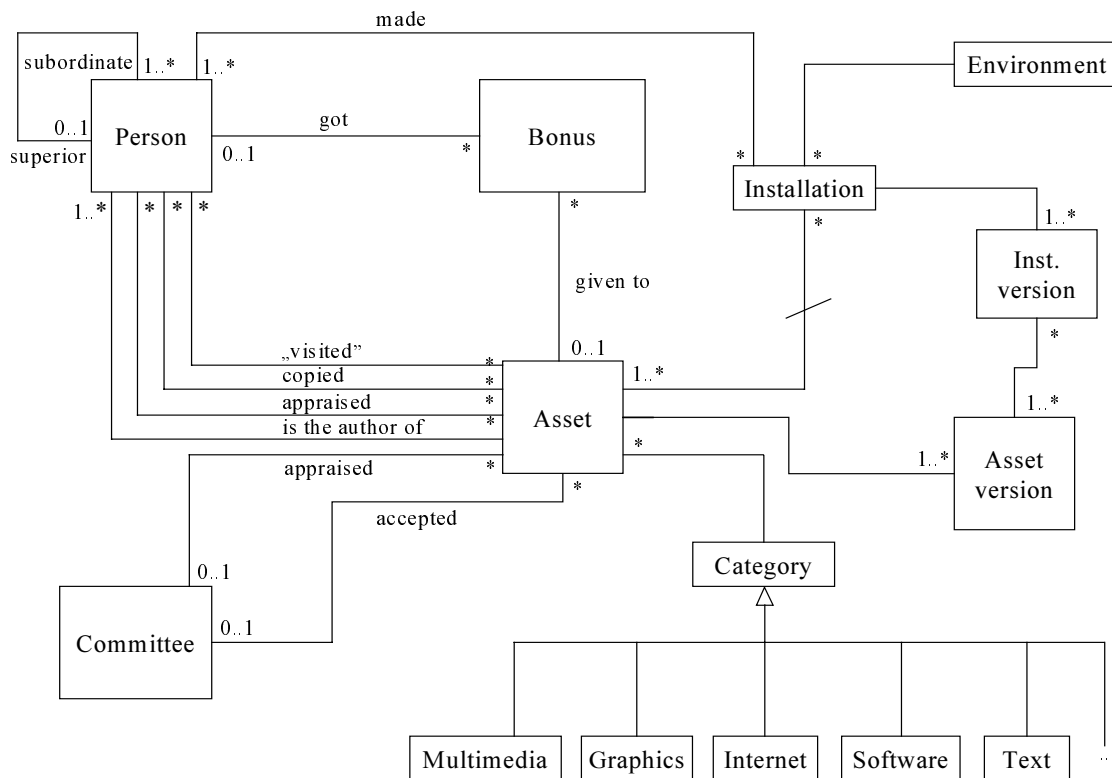


Fig.1. Reuse Repository Conceptual Scheme

During the design of RR we tried to take into consideration all the remarks and conclusions presented above. Our idea was that RR through appropriate tools should facilitate the process of gathering, storing and accessing information useful for a group of co-operating people. We assumed that RR should turn out pretty useful wherever computers are extensively used as working tools, for example, at a school, office or enterprise. We did not enforce limitations concerning kinds of stored assets. We assumed that the users could make decisions concerning the RR content, tailoring it to their own needs. RR users can also be reviewers of assets prepared by other users. Obviously, the system should support the possibility of removing information recognized as useless or poor quality. The users were given both the possibility of getting information in a passive way (through messages sent from RR) and actively by searching for appropriate assets by themselves. We wanted also to give them possibility of copying assets suitable for their current needs.

The conceptual class diagram of RR presented in Fig.1 has been designed on the basis of the above assumptions.

3. Assets Management

An efficient assets management requires a supporting infrastructure, i.e. mechanisms and tools facilitating assets' acquiring, cataloguing, classification and quality. Before the discussion on what we did in our application to prepare the infrastructure for ensuring RR efficiency, we will shortly discuss kinds of RR users that we have distinguished.

3.1 Architecture of the Reuse Repository

As usual for nowadays applications, we have assumed that RR has to be available on Web and accessible through a Web browser. The RR architecture is the typical client-server architecture with three layers: client, business and database. The client and the server communicate through the HTTP protocol. The client side has two tasks to do: sending a user query to a database and displaying the answer to the query. An Internet browser is used for accomplishing both tasks. We have tested two browsers: Microsoft Internet Explorer ver. 3.x and Netscape Navigator (Communicator) ver. 3.x. The business and database layers are accomplished on the Windows NT 4.0 platform. The business layer consists of WWW server, FTP server and ASP scripts server (HTML pages generator). The Microsoft Internet Information Server (IIS ver. 3.0 with ASP) is responsible for WWW and FTP. The database layer is implemented on Microsoft SQL Server 6.5. An access to a database employs an ODBC driver, which constitutes an element of the ADO technology. The RR database stores the asset catalogue and information about the users. The asset library is separately located on the asset server.

3.2 Reuse Repository Users

We have distinguished the following categories of users: the guest (any anonymous user), the registered user, the committee and the administrator. Each specified group has special access rights to RR resources and some set of application functions assigned to it (c.f. Fig. 3). A part of functions is common for every user category; in particular everybody can look through the asset library or reports. Specified functions are the only possible for a user belonging to the guest category. Other possibilities vary dependently of the role the user plays in concrete situation.

The RR registered users are the most common. Its members, besides having access to public functions, can insert new assets by putting them into a “waiting-place”, where they wait for committee approval. The registered user can also appraise assets used by him/her and can copy suitable assets from the RR library.

The committee is a special body, which has to be established to verify asset usefulness as a reuse element. If verification concerns an asset from the “waiting-place”, the positive committee approval (acceptance) makes the condition of introducing it to RR library. The process of verification the assets already comprised in the library should be done in a systematic way, by some period of time. An asset rarely visited and/or negatively evaluated by users can be specified as the candidate for removing from the Reuse Repository base.

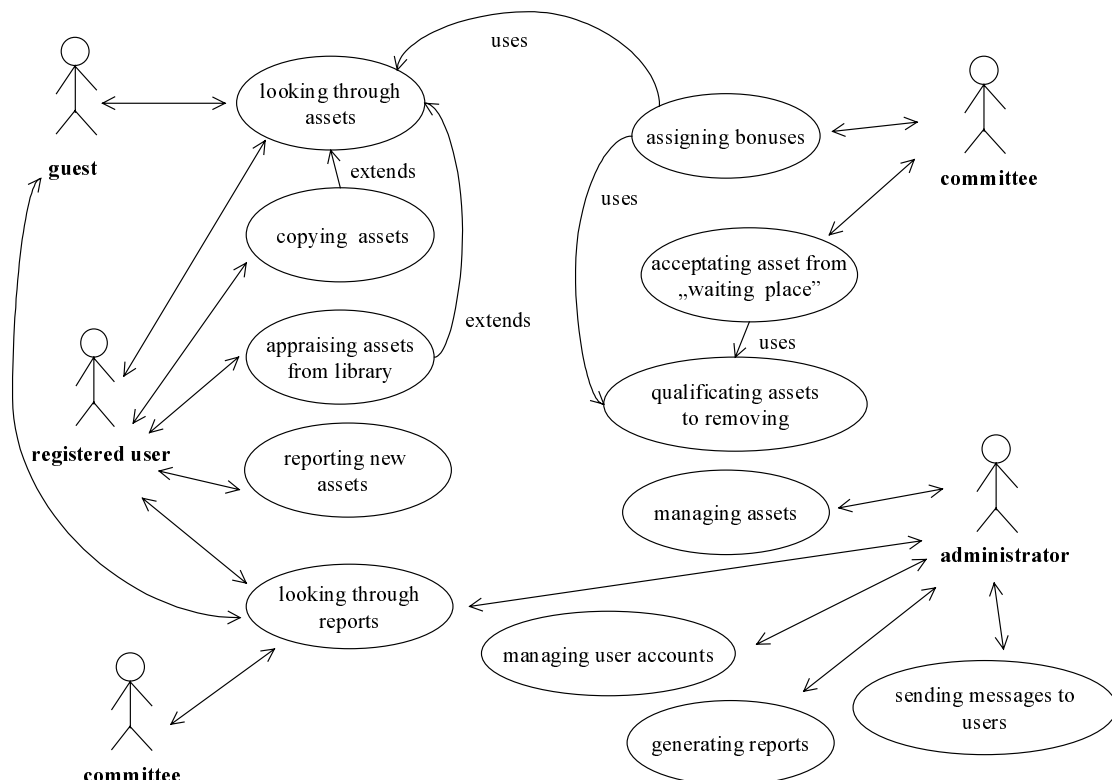


Fig.2. Reuse Repository use case diagram

The main task of individuals belonging to the administrator category is maintenance of the repository. It is a very important task from the perspective of fighting against psycho-sociological barriers, which impede introducing reuse as an efficient technology. The administrative management includes errors detecting and making improvements and extensions to the application of RR. The tasks the administrative group has to do can be specified as follows: assets and users accounts managing, report generating, and sending messages about some interesting events. Concrete users can be interested in the following events: a category of assets is to be added or removed, assets are to be added/removed, new versions of existing assets appear, and so on. Fig.2 presents a use case diagram for the RR.

3.3 Assets Acquiring

On purpose, the Reuse Repository has been designed as a generic application, with no bias towards needs of special group of users. We have just taken into consideration results of inquires that we did. We have supposed that build-in RR gratification system (we’ll discuss later) not only should facilitate acquiring of new assets but also

should make possibly the process of quality verification. In long time terms, we hope it should result in a great amount of assets with suitable high reuse potential according to the needs of a cooperating group of people.

3.4 Distribution of Assets

The assets stored in the Reuse Repository are subdivided onto the following categories:

- Internet (e.g. ASP files, HTML pages, useful Internet addresses, and so on),
- graphics (graphical files of different formats),
- multimedia (sound, video, animation files),
- software (files storing source or binary code),
- text (text files of different formats).

The undertaken subdivision of assets (i.e. division into some meaningful structures) allows the user to match his/her needs to components contained in the library. The category an asset belongs to makes the part of one from three classification schemes that we will discuss later.

3.5 Tools to Acquiring Information Stored in RR

It is obvious that the effort necessary to find and recognize a component must be lower than building it from scratch. Components, which fulfill user requirements, should be retrieved as easily as possible. Browsers are important tools used to locate desired components. Unfortunately, a typical browser could not facilitate lookup through a big library. If a component is not named in a meaningful way or the library content is not organized into meaningful classes the component could be difficult to find. Using directories with precise component descriptions and classification schemes could significantly facilitate the process of identification and acquiring information about useful assets. Besides asset descriptions the directories also include references to the storage with physical assets or information how to acquire them.

In the Reuse Repository we have designed tools to facilitate the process of information acquiring based on active and passive strategy. By active strategy we mean looking through catalogue information through the classification schemes. The catalogue information consists of the following elements: an asset name, the category it belongs, date of introducing it into the library, short description, users' opinions, bonuses it has been granted, links to installations where it has been used, and links to persons from several categories (author, visitor, evaluator, user, etc.).

3.6 Information Acquiring Based on the Active Strategy

It is connected with the active user attitude. We have delivered special forms where a user can choose between classification schemes based on the following attributes:

- “text” - means any sequence of characters to be matched in the assets' description;
- “period of time” specifies the period of time when assets were introduced into RR;
- “role” specifies a kind of person connected with assets (author, user, etc.)

Searching based on the first attribute returns information about those of assets, which have in the “description field” of catalogue the sequence of characters specified by a user. The search can be additionally qualified by a period of time. The searching returns the following information about assets: name, category, date of introducing into RR resources, and asset short description. Additionally, they return links to every

person associated with the concrete asset, the links to the installations, where the asset have been used, and the links making possible to copy assets.

The searching “from person to asset” retrieves persons fulfilling specified criteria; then a user can navigate to assets associated with these persons, according to their role in a particular asset. The system gives the link, which makes it possible to send e-mail to a selected person, for example to get more detailed information about an asset.

3.7 Information Acquiring Based on the Passive Strategy

In this mode the user waits for information. The user has the possibility of determining asset categories being in the field of his interest. A message sent from RR to users is connected with some predefined events: adding new category of assets, removing a rarely “visited” category, new version of some assets, and so on. The list of events, which trigger messages, could be expanded during the development of a RR application. Information is sent to users by e-mail.

3.8 Facilitating the Asset Quality Assurance

An asset undergoes the first verification when it is staying in the “waiting place”. The committee is the body responsible for introducing assets into RR resources. The committee makes the decision about each asset. Assets accepted by this procedure are catalogued and stored in the library. The system provides also their further verification and approval by users. The system makes it possible to remove assets if they were recognized as useless or poor quality.

We assumed that the asset appraisal should take a sufficiently concise form to facilitate analysis of them. From the other point of view the appraisal form should be sufficiently extensive to involve various asset aspects. As a trade-off we have decided that appraisal will comprise the following parts:

- idea (usefulness),
- technical solution,
- interface,
- documentation,
- comments.

Each of the first four parts can take value from the 0 to 10, where 0 means the worst appraisal and 10 the best one. When a user wants to pass some additional more detailed remarks, he/she can use the comment field to explanation the evaluation in detail. The asset author, after user appraisals and opinions, can improve the quality of the asset. The author may also want to take a new approach to introduce new assets into RR resources in case of low user appraisals of his/her previous assets.

4. Making Foundation for Formation of Reuse Culture

To encourage practising reuse we attempted to build a mechanism supporting gratification of asset authors and asset users. We want to motivate people not only to use assets already stored in RR but also to improve assets quality and to produce brand new ones. The mechanism is based on bonuses given to good assets, to good authors and to the most active RR users. Bonuses take a form of points conferred by the committee. An asset appraisal done by the committee can be based on user appraisals and on some events tracked by the application. Such events include visiting

an asset, making copy, evaluating an asset by a user, creating brand new asset (or some version of it), efficient using an asset in some project(s), etc. The most active users were those mostly visiting, copying and appraising assets. Making appraisal we treated as a form of cooperation in the process of improving asset quality, which implicitly affect on quality of RR resources. We assumed that the bonus procedure should be executed in some period of time (e.g. once a month). The procedure assumes that points given to the assets and persons are stored in the bonus history.

Although we do not have real awards in our system we consider that bonuses given to assets and RR users take part in asset quality inspection. Bonuses allow some users to receive public recognition because of their activity (for example, as the best assets authors). Additionally, they can and should be taken into consideration when awarding authors and the most active RR users in their workplace. We have designed some mechanism which allows storing a worker hierarchy in enterprise, which is using Reuse Repository (association “superior-subordinate”). This mechanism facilitates gathering information about activity of members of every team. Gratification points can be turned into the reality for example through financial awards in proportion to amount of received points.

Implementation of any new work technology requires a process of change. People are slaves of habits and are resistant to changes unless convinced through education and management support that change could be beneficial. Reuse significantly changes a way of production, what requires the change of mentality and like every change will be resisted, in the first reaction. It is commonly known that existing of a proper incentive system facilitates fighting against reluctance to implement reuse technology. Establishing such a system makes an important step on a reuse implementation.

It is important from another reason, too. Reuse technology requires more skill specialization; some people have to construct high quality generic elements, some have to join them together and some to manage sets of such elements. Specialization needs more knowledge than traditional technologies but significantly improves productivity of every worker - but this often results in work monotony. The monotony in turn reduces motivation. An incentive system is often pointed out as a suitable medicine for such a problem. Awards (especially financial), as was revealed by many research and empirical studies of high-technology professionals, are really strong incentives in encouraging people to change their attitude to reuse technology, but only at the first stage in introducing it. Considering a long term of time, rewards relating to job content, recognition by superiors and coworkers and participation in interesting, challenging work seem to be more important than financial bonuses for a great amount of professionals. What is obvious and emphasized by many authors, positive management support constitutes a key factor in all successful reuse programs.

5. Summary and Conclusion

In assumptions, the Reuse Repository allows gathering of reuse assets and accessing them by a collaborating group of users. Every group member has rights to decide on reuse resources, voting after or against. In that sense, every user can be treated as a repository co-author.

A system such as the Reuse Repository can be useful for every group of people using computers and cooperating through network. Depend on specialization (or interests) of a group as a whole or of its individual members the Reuse Repository can include assets belonging to different categories. Exemplary assets, for a software

developers' team, could be function or class libraries, subsystem or application frameworks, design patterns, and so on. Furthermore, for a desktop publishing group a possible set of useful assets could include: graphical files, advertisement slogans, document patterns, etc. During designing of the system we haven't put any limitations on assets categories. We have also provide in the system special tools for managing assets and tools which can support the formation of the reuse culture.

6. Bibliography

- [Amb98] S.Ambler. A Realistic Look at Object-Oriented Reuse. Reprinted from Software Development Magazine, Microsoft Development Network, 1998
- [Ans99] D.Ansorge, K.Bergner, B.Deifel, N.Hawlitzky, C.Maier, B.Paech, A.Rausch, M.Sihling, V.Thurner, S.Vogel. Managing Componentware Development - Software Reuse and the V-Model Process. CAiSE 1999, 134-148
- [App97] B.Appleton. Patterns and Software: Essential Concepts and Terminology. <http://www.enteract.com/~bradapp/>
- [Col94] D.Coleman, P.Arnold, S.Bodoff, Ch.Dollin, H.Gilchrist, F.Hayes, P.Jeremas. Object-Oriented Development - The Fusion Method. Prentice Hall, Englewood Cliffs, 1994
- [Con93] P.Constantopoulos, M.Jarke, J.Mylopoulos, and Y.Vassiliou. The Software Information Base: A Server for Reuse. VLDB Journal, The Boxwood Press, 1993
- [DMcGr95] J.D.McGregor, J.Doble, and A.Keddy. A Pattern for Reuse: Let Architectural Reuse Guides Component Reuse. SIGS Publications, 1995, Object Magazine, <http://sw-eng.falls-church.va.us/ReuseIC/pubs/journals/>
- [Ed99] O.Edelstein, A.Yaeli, G.Zodik. E-Colabra: An Enterprise Collaboration & Reuse Environment. NGITS 1999, 229-236
- [Fra98] S.Fraser, M.Davies, M.L.Griss, L.Hohmann, I.Hopper, R.Joos, B.Opdyke, Software Reuse: Nemesis or Nirvana? (Panel). OOPSLA 1998, 417-420
- [GoRu95] A.Goldberg, K.S.Rubin. Succeeding with Objects, Decision Frameworks for Project Management. Addison- Wesley Publishing Company, 1995
- [Gri91] M.L.Griss, S.S.Adams, H.Baetjer Jr., B.J.Cox, A.Goldberg. The Economics of Software Reuse (Panel), OOPSLA 1991, 264-270
- [Gri97] M.L.Griss, J.Favaro, M.d'Alessandro. Featuring the Reuse-Driven Software Engineering Business. Draft for Object Magazine, 1997
- [Jac97] I.Jacobson, M.L.Griss, P.Jonsson. Software Reuse, Architecture, Process and Organization for Business Success. ACM Press, 1997
- [Lim98] W.C.Lim. Managing Software Reuse, A Comprehensive Guide to Strategically Reengineering the Organization for Reusable Components. Prentice-Hall, 1998
- [Lam98] W.Lam. Change Analysis and Management in a Reuse-Oriented Software Development Setting. CaiSE 1998, 219-236
- [McCl95] C.McClure. Model-Driven Software Reuse: Practicing Reuse Information Engineering Style. Extended Intelligence, 1995, <http://www.reusability.com/papers2.html>
- [McCl97] C.McClure. Software Reuse Techniques: Adding Reuse to the System Development Process. Prentice-Hall, , 1997
- [PaHeVe96] Y.Pant, B.Henderson, and J.M.Verner. Generalization of Object-Oriented Components for Reuse: Measurements of Effort and Size Change. SIGS Publications, 1996, Journal of Object-Oriented Programming, <http://sw-eng.falls-church.va.us/ReuseIC/pubs/journals/>