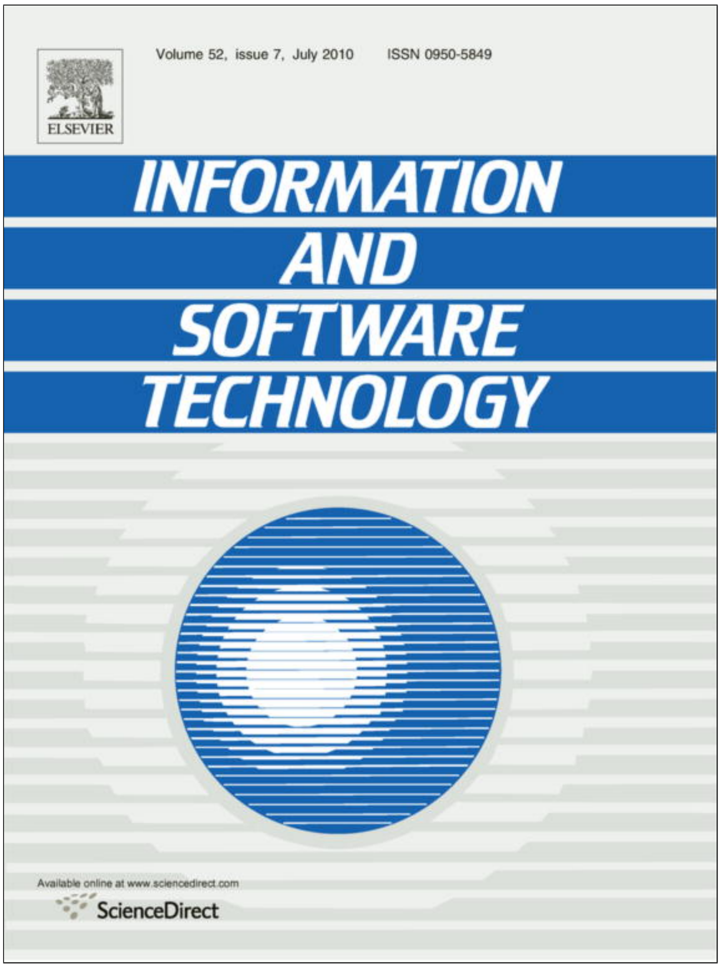
Volume 52, issue 7, July 2010
ISSN 0950-5849
ELSEVIER
INFORMATION AND SOFTWARE TECHNOLOGY
Available online at www.sciencedirect.com
ScienceDirect

Information and Software Technology 52 (2010) 707–719

Contents lists available at ScienceDirect

Information and Software Technology

journal homepage: www.elsevier.com/locate/infsof

A longitudinal study of development and maintenance

Magne Kristoffer Davidsen, John Krogstie *

Department of Computer and Information Science, Norwegian University of Science and Technology (NTNU), Trondheim, Norway

ARTICLE INFO

Article history:
Received 7 September 2009
Received in revised form 6 March 2010
Accepted 8 March 2010
Available online 23 March 2010

Keywords:
Application evolution
Application maintenance
Survey investigation

ABSTRACT

Context: The information systems we see around us today are at first sight very different from those that were developed 15 years ago and more. On the other hand, it seems that we are still struggling with many of the same problems, such as late projects and unfulfilled customer demands.
Objective: The paper presents finding relative to the distribution of work between maintenance and development tasks, comparing to the results reported earlier to assess the stability of important metrics within the area.
Method: This paper presents the main results of a survey-investigation performed in 2008 in 67 Norwegian organizations comparing the distribution of work to results from similar investigations performed in Norway in 1993, 1998, and 2003. Some comparisons to similar investigations performed in USA before this is also provided.
Results: The amount of application portfolio upkeep (work made to keep up the functional coverage of the application system portfolio of the organization, including the development of replacement systems), is at the same level as reported in 1998 and 2003. The level of application maintenance is also on the same level as the similar investigations conducted in 2003 and 1998. There was a significant increase in both maintenance and application portfolio upkeep from 1993 to 1998, which could partly be attributed to be the extra maintenance and replacement-oriented work necessary to deal with the "year 2000 problem", but this seemed to be reversed in 2003 and 2008. As for the 2003 investigation, the slow IT-market in general seemed to have influenced the results negatively seen from the point of view of application systems support efficiency in organization. No similar explanation can be used for the 2008 numbers.
Conclusion: Based on the last surveys it seems than a stable level of work distribution both on maintenance and application portfolio upkeep have been reached, although the underlying development technologies are still undergoing large changes. This is contrary to others claiming that the amount of maintenance is still increasing.

1. Introduction

Large changes in how we develop information systems and the underlying technology for information systems have been witnessed over the last 10–15 years. For instance, over this period the prevalent development methods, programming languages and general technological infrastructure have changed dramatically. In the early 90s, one was going from mainframe solutions to a client–server, and then to an internet architecture for many applications. Year 2000 and the dot.com crash had large impact locally on the development and maintenance of systems. More lately SOA, outsourcing, open source and agile development methodologies would also be expected to have an impact. According to [14] one of the impacts on the state of IS-development is the increasing amount of time used for maintenance of systems (instead of developing new systems). On the other hand, many of the intrinsic problems and aspects related to information systems support in organizations are similar now to what they were 15 years ago. Application systems are valuable when they provide information in a manner that enables people to meet their objectives more effectively [3]. Many have claimed that the large amount of system work that goes into maintenance is a sign on poor use of resources to meet these demands. On the other hand, as stated already in [4], it is one of the essential difficulties with application systems that they are under a constant pressure of change. Given the intrinsic evolutionary nature of the sources of system specifications, it should come as no surprise that specifications and the related information system must evolve as well [3,24]. Thus, all successful application systems are changed; there is nothing detrimental about this. Unsuccessful systems or systems that can no longer be maintained and enhanced are removed or replaced. Talking

* Corresponding author. Address: IDI, NTNU, N-7491 Trondheim, Norway. Tel.: +47 93417551; fax: +47 73594466.
E-mail address: krogstie@idi.ntnu.no (J. Krogstie).
URL: http://www.idi.ntnu.no/~krogstie (J. Krogstie).

0950-5849/$ - see front matter
doi:10.1016/j.infsof.2010.03.003

about having a high percentage of maintenance work being done as a measure of information system support efficiency can thus be highly misleading.

The goal of both development activities and general maintenance is to keep the overall information system support of the organisation relevant to the organization, meaning that it supports the fulfilment of actual organizational needs. A lot of the activities usually labelled as maintenance, are in this light value-adding activates, enabling the users of the systems to do new task. On the other hand, a large proportion of the 'new' systems being developed are so-called replacement systems, mostly replacing the existing systems without adding much to what end-users can do with the overall application systems portfolio of the organization.

Based on this argumentation we have earlier developed the concept application portfolio upkeep[1] as a high-level measure to evaluate to what extent an organisation is able to evolve their application system portfolio efficiently. How application portfolio upkeep is different from maintenance is described further below.

In this paper, we present the main results from a survey-investigation performed in Norwegian organisations during the end of 2008. The main motivation of the investigation is to compare the current development and maintenance situation in Norway with what has been reported in similar investigations in Norway and abroad earlier, to see if the pattern of resource use in IT departments is stable or if we can observe changes and in case why these changes are happening. Our main hypothesis in this regard is described in further detail in Section 3. A more comprehensive report from the investigation can be found in [7].

1.1. Outline of the paper

We will first give definitions of some of the main terms used within software development and maintenance, including the terms application portfolio upkeep and application portfolio evolution, notions which were introduced 15 years ago [18]. We then describe the research method and the main hypotheses investigated. The main results are then presented. In Section 5 we discuss threats to validity of the results. The last section summarises our results relative to the hypothesis, and presents ideas for further work.

2. Basic concepts

Maintenance has traditionally been divided into three types: corrective, adaptive and perfective [11] inspired by Swanson [35]. We here used the IEEE glossary terms in [11] with some clarifications.

Maintenance is defined as the process of modifying a software system or component after delivery.

1. Corrective maintenance is performed to correct faults in hardware and software.
2. Adaptive maintenance is performed to make the computer program usable in a changed environment.
3. Perfective maintenance is performed to improve the performance, maintainability, or other attributes of a computer program Perfective maintenance has been divided into enhancive maintenance [5] and non-functional perfective maintenance. Enhancive maintenance implies changes and additions to the functionality offered to the users by the system.[2] Non-functional perfective maintenance implies improvements to the quality features of the information system and other features being important for the developer and maintainer of the system, such as modifiability. Non-functional perfective maintenance thus includes what is often termed preventive maintenance, but also such things as improving the performance of the system.

In addition to the traditional temporal distinction between development and maintenance, the concepts of application portfolio evolution and application portfolio upkeep as illustrated in Fig. 1 has been introduced earlier [10].

1. *Application portfolio evolution*: Development or maintenance where changes in the application increase the functional coverage of the total application systems portfolio of the organisation. These categories are found to the right in Fig. 1 and includes:
   - Development of new systems that cover areas, which are not covered earlier by other systems in the organisations.
   - Enhancive maintenance.
2. *Application portfolio upkeep*: Work made to keep up the functional coverage of the information system portfolio of the organisation. These categories are found to the left in Fig. 1 and includes:
   - Corrective maintenance.
   - Adaptive maintenance.
   - Non-functional perfective maintenance.
   - Development of replacement systems.

As argued above, this distinction is used because we believe it gives a better indication of the efficiency of the application systems support in an organisation than the traditional distinction between development and maintenance.

We note that some writers provide more detailed overview of maintenance tasks [6,14]. Jones [14] has in total 21 categories:

1. Major enhancements (new features of >20 function points).
2. Minor enhancements (new features of <5 function points).
3. Maintenance (repairing defects for good will).
4. Warranty repairs (repairing defects under formal contract).
5. Customer support (responding to client phone calls or problem reports).
6. Error-prone module removal (eliminating very troublesome code segments).
7. Mandatory changes (required or statutory changes).
8. Complexity analysis (quantifying control flow using complexity metrics).
9. Code restructuring (reducing cyclomatic and essential complexity).
10. Optimization (increasing performance or throughput).
11. Migration (moving software from one platform to another).
12. Conversion (Changing the interface or file structure).
13. Reverse engineering (extracting latent design information from code).
14. Re-engineering (transforming legacy application to client–server form).
15. Dead code removal (removing segments no longer utilized).
16. Dormant application elimination (archiving unused software).
17. Nationalization (modifying software for international use).
18. Year 2000 repairs (date format expansion or masking).
19. Euro-currency conversion (adding the new unified currency to financial applications).
20. Retirement (withdrawing an application from active service).

[1] This concept was originally termed 'functional maintenance', but we have realized that this term might be misleading.

[2] It is not obvious from the definitions above that this is in fact included as part of perfective maintenance. Reading, e.g. [23], it is obvious that this kind of tasks are included in the American investigations that we compare with.

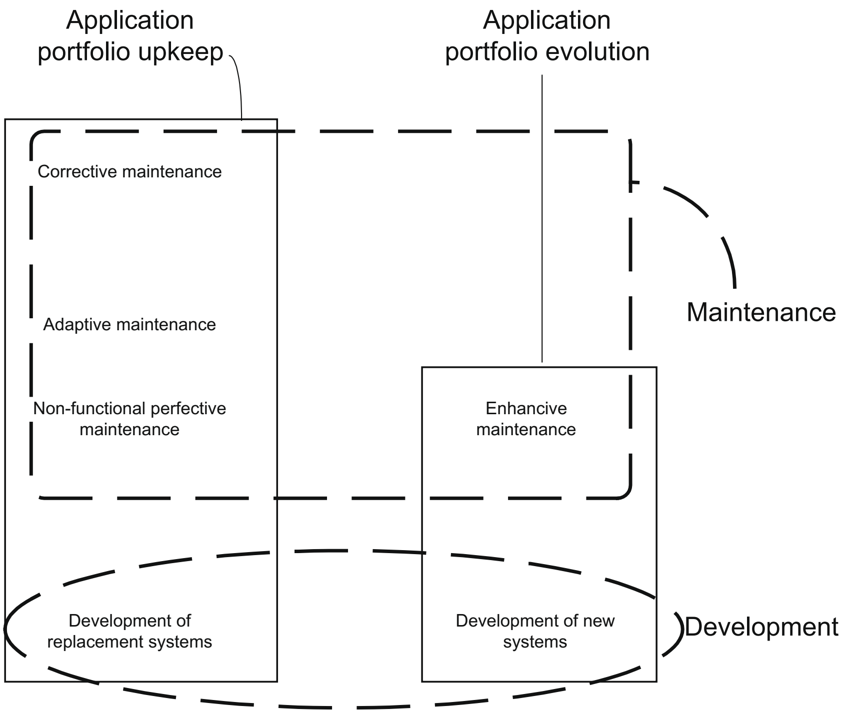

Fig. 1. Relationships between development and maintenance figures.

21. Field service (sending maintenance members to client locations).

Some of these areas are no longer relevant, whereas most of the others fall into the main categories described above. As an exception, note here in particular point 5 on the inclusion of user-support as part of maintenance (a view shared with, e.g. Dekleva as reported in [8]), an area usually looked upon as belonging to other work than development and maintenance. We will discuss this further below.

3. Research method

The survey form was implemented using the SurveyMonkey web-tool.[3] SurveyMonkey is a much used web-based survey tool, which makes the set-up and follow-up of surveys easy to perform. Invitations were distributed through SurveyMonkey by e-mail to 300 Norwegian organizations. For different reasons, only 278 of the invitations was delivered to the organizations (e.g. since some potential respondents did not want to receive request from SurveyMonkey or had changed their e-mail address from the one provided by NCS (see below)). It is not guaranteed that all of these were given to someone in the organization that would be able to answer the survey though. The organizations were randomly selected from the list of member organizations of DnD (The Norwegian Computer Society – NCS), and the e-mail send to the contact person in the organization (NCS has currently around 1000 member organizations).

The form contained 48 questions including demographic data. The contents of the form were based on previous investigations within this area; especially those described in [10,21,26,27,31,36]. The questions from the survey collecting data on the issues dealt with in this paper can be found in Appendix A.

The survey-form can be divided into 10 parts:

1. Question on name of responder and organiser (Questions 1–3, not included in Appendix A).
2. Questions related to characteristics of the responder (Questions 4–8). These are similar to questions asked in the earlier questionnaires on such matters [27,36].
3. Questions related to characteristics of the responding organisation (Questions 9–12). These are similar to questions asked in the earlier questionnaires on such matters [27,36].
4. Questions related to distribution of work (Questions 13–17). This is an extended list compared to [27,31], in that we have divided perfective maintenance into enhancive maintenance and other perfective maintenance as described in Section 2, and that we have explicitly asked for time used on system operations and support of end-users in addition to development and maintenance work (covered by 'other work' in the American investigations). In the 2008 investigation, we also asked about the amount of outsourcing within the main categories.
5. Question related to the IT department. (Questions 18–22). These questions are based on questions from [36].
6. Question on the application portfolio and user population (Questions 23–31). These questions are based on questions from [36].
7. Question on the applied technology for systems development (Questions 32–36). These questions are based on questions from [36]. In addition, we have included a question relating to the use of SOA.
8. Questions on ongoing development and development of replacement systems (Questions 37–42) question are taken from [36]. In addition we have included a questions related to reuse and use of SOA.
9. Question on use of methodology, use of development tools and use of organizational controls (Questions 43–47). Questions on organizational controls are also found in

[3] http://www.surveymonkey.com/.

[27,31,36]. Questions on use of methodology and development tools are developed specifically by us in our three previous investigations.
10. Questions on problems with maintenance (Question 48, not included in Appendix A). This is the same table of questions as has been used in [27,31,36].

As we see, we use to a large extend the questions that has traditionally been used in these kind of investigations, although translated into Norwegian in our investigations. Some new questions are provided, to investigate novel aspects of our investigations, in particular aspects related to application portfolio upkeep/evolution. Questions related to outsourcing and use of SOA was included specifically in the last investigations based on wishes from NCS to investigate these areas in particular. We conducted a pilot study in a few companies to detect unclear questions and whether the time for filling-in the forms was reasonable. We also got comments from several colleagues including experts in cognitive psychology which are highly familiar with the use of survey techniques concerning clarity of questions.

On some of the questions, we were interested in the quality of the answers, recognising that some of the information called for might not be easily obtainable. It was also room for issuing open-ended remarks on most questions. This possibility together with the possibility to crosscheck numbers between different questions was the mechanisms used to identify possible misunderstandings among the respondents, which were followed up afterwards.

Galtung [9] regards that the minimum sample size that is meaningful in a survey is 40 units. Since earlier survey-investigations in the area of development of application systems toward the same population had given a response rate of about 22% [2,10,18,21] and the response rate of similar surveys has been around 20–25% (e.g. [27,31]), an answer ratio of approximately 20% was expected. This would have resulted in around 60 responses. 79 responses were returned, giving a response rate of 28%. Out of these 67 responses could be used for the analysis since the rest of the 12 responses were incomplete.

The forms were filled in using the web-form defined in SurveyMonkey by people with long experience with application systems related work (average 17.5 years), typically filling the role as IT director in the company. Of the respondents, 57 out of 67 (85%) indicated that IT was of extremely (5) to large (4) strategic importance for the organization (on a Likert scale from 1 to 5). This indicates that application systems support including own development and maintenance is an area of importance for the majority of respondents. Judged by the responses, all organizations where doing work on all support-line levels (1–3) [16], but with different emphasis on different types of support. Because of this and the relatively low response rate, we will be cautious in our interpretations of the results.

The data could be exported from SurveyMonkey as Excel-files, and these could be imported into a more advanced statistics tool. The results from the survey were analysed using the statistical package SPSS. Statistical significance of some of the results is determined using the two-tailed Student t-test for normally distributed data, and the Mann–Whitney non-parametric test when this was not the case. To decide what type of test to perform the variables used in the comparisons were tested for normality. Where for at least one of the two figures to investigate either the Shapiro–Wilks (S–W Sign) and/or the Kolmogorov–Smirnov (Lilliefors-Sign) significance levels were less than 0.05, we used the non-parametric Mann–Whitney test.

Statistically significant results are highlighted in Section 4 using boldface font. The validity of the results is discussed in more detail in Section 5.

3.1. Previous investigations

We will compare some of the results with the results of similar investigations. The most important of these investigations are:

1. The investigation carried out by Krogstie, Jahr and Sjøberg in 2003 reported in [21].
2. The investigation carried out by Holgeid, Krogstie and Sjøberg in 1998 reported in [10].
3. The investigation carried out by Krogstie in 1993 reported in [18–20].
4. The Lientz and Swanson investigation (LS) [27]: that investigation was carried out in 1977, with responses from 487 American organizations on 487 application systems.
5. The Nosek and Palvia investigation (NP) [31]: a follow-up study to Lientz/Swanson performed in 1990 asking many of the same questions as those of Lientz and Swanson. Their results are based on responses from 52 American organizations.
6. The Swanson and Beath investigation (SB) [36]: reports on case-studies of 12 American companies that in addition to questions given in the Lientz/Swanson study focused on portfolio analysis and replacement systems. These aspects are also a major part of our investigation.

The three first surveys in the list are the main Norwegian investigations in the Lientz/Swanson tradition. They contain the results from investigations of 54, 52 and 53 Norwegian organizations, respectively. The hypothesis testing will be focusing on comparing the results from the 2003 and 2008 investigation. In addition to these investigations, a number of later investigations have been done in related areas, but they typically focus on the distribution of maintenance tasks only [13,23,29,33,34], many only looking on the situation in one organization, or maintenance related to one system.

Several of the organizations that received a survey-form in the 1993, 1998 and 2003 studies also received the invitation to fill out the form in 2008 (by chance, since they were also at that time members of the Norwegian Computer Society), and many of the same questions have been asked. The methods that are used are also similar, enabling us to present a 'longitudinal' survey study, although the overlap among actual respondents to the survey is limited to only a few organizations across different instalments of the survey. Even if the population selection process was the same, the actual organizations in these have changed quite a bit over the period of fifteen years, both because of changed focus on IT, and because of the volatile business environment, with a number of acquisitions, mergers and bankruptcies in the area. To illustrate, only a third of the organizations that where answering the survey in 1993, were members of NCS in 2008, and several of these had changed quite a bit due to mergers and acquisitions. Concretely, one organization has answered three of the four surveys, and 6 organizations have answered two of the surveys.

3.2. Hypotheses

The following hypotheses were formulated to investigate the development of the different measures for distribution of work.

Hypotheses:

- H1: There is no difference between the percentage of maintenance time in our survey and what are reported in previous surveys.

Rationale: When comparing the percentage of time used for maintenance activities in organisations earlier, we have found this to be stable on around 40% of the overall time in investigations both in the 70s, 80s, and 90s in both USA and Norway. We would not expect this to be different in this survey. Still it is interesting to investigate given the high expectations to new technologies such as SOA.

- H2: There is no difference between the breakdown of maintenance work (in corrective, adaptive, and perfective maintenance) in our survey and what are reported in previous surveys.

Rationale: A number of investigations (also from the later years) reporting on the distribution of time among maintenance tasks [11] reports very different numbers for these distributions. On the other hand the scope of the investigations varies greatly. Whereas some look on single systems of numerous organizations and the whole portfolio of numerous organizations, other look only at one or a few applications in one organization. Since this distribution naturally will differ according to where the system is in the lifecycle (development, evolution, servicing, phase-out, closed [32]), this different results might be expected. When averaging across a large number of application portfolios on the other hand, we would expect a more stable distribution. We will investigate this relative to the percentage of corrective, adaptive, and perfective maintenance.

- H3: There is no difference between the percentage of development time in our survey and what is reported in previous surveys.

Rationale: When comparing the percentage of time used for development activities in organisations earlier, we have found this to be decreasing when comparing our study in 1993 with earlier studies. Whereas maintenance activity is held mostly constant, the amount of time used for user-support and system operations have grown since the 70s and early 80s, mainly because end-users have much more advanced machines (PDAs and PCs vs. dumb terminals), and because of opening up many more systems to external users over internet. The overall infrastructure is also much more complex involving *n*-layer architectures and multi-channel solutions being more challenging to operate. We would thus expect this trend to continue, i.e. that this would be rejected.

- H4: There is no difference between the percentage of combined maintenance time and time used for user support in our survey and what are reported in previous surveys.

Rationale: As seen in Section 2, some authors (e.g. Capers Jones as reported in [14]) that have claimed that increasing time is used on maintenance, have included user-support as part of maintenance. Thus it is interesting to also combine the time used on user support with maintenance as we have defined it to see if this combined figure is increasing. With the increasing user population for systems (including the increasing number of external user), we would expect this to increase, following closer the pattern described by Capers Jones, i.e. rejecting this hypothesis.

- H5: There is no difference between the distribution of work among maintenance and development in our survey and what is reported in previous surveys when disregarding other work than development and maintenance.

Rationale: Since the amount of other work than development and maintenance is taking up more time now than earlier, we found it beneficial also in the surveys in 1993, 1998, and 2003 to look at only the proportion between development and maintenance time. The amount of time used for maintenance has earlier shown to be stable on around 60% (i.e. 40% for development) in investigations both in the 70s, 80s, and 90s, across countries (and not increasing to take up a larger and larger part of the work. When a larger percentage of maintenance is claimed, this often includes, e.g. user support [14]). This pattern was broken in the study from 1998 (with significantly higher proportion of maintenance), which was found to be partly due to the Y2K problem, but we would expect this to be reversed having put this well behind us.

- H6: There is no difference between the distribution of application portfolio upkeep and application portfolio evolution in our survey and what is reported in previous surveys.

Rationale: Since these numbers have been investigated only three times before, we were eager to find out if they also had the same stable distribution as the maintenance figure. They were significantly higher in 1998 than in 1993, but on the same level in 2003 as in 1998.

4. Results

First, we present some of the overall demographics of the survey. Similar results from our previous surveys conducted in 2003, 1998, and 1993 are included in parenthesis where the numbers are comparable.

42% (2003 – 20%; 1998 – 43%) of the organizations had a yearly data processing budget above 10 mill NKr (approx. 1.5 mill USD), and the average number of employees among the responding organizations was 1083 (2003 – 181; 1998 – 656; 1993 – 2347). The average number of full-time personnel in the IS-organizations reported on was 15.1 (2003 – 9.8; 1998 – 10.9; 1993 – 24.3), whereas the average number of full-time application programmers and/or analysts was 2.7 (2003 – 4.1; 1998 – 4.6; 1993 – 9.5). Whereas one found significant differences between the size of the IT-department and the number of systems developers and the proportion of portfolio upkeep in the investigation from 1993 [20], such significant results have not been found in the later investigations, thus we do not expect that the difference in average size of organization or data department to have a large influence on the result. The average number of full time hired IT consultants was 2.8 (2003 – 0.7; 1998 – 2.7). The dip in 2003 on the number of consultants reflects the limited activity at the time in the Norwegian consultant-market, where all the major consultant-companies had to lay off hundreds of employees. An area not asked about in previous investigations was the amount of outsourcing. In the current investigation around a third of the IT-activity was outsourced. Whereas only two of the respondents reported to have outsourced all IT-activities, 84% of the responding organisations had outsourced parts of their IT-activity. As we see, the responding companies are larger than the last surveys, but on average smaller than the first survey, whereas they have less people working as systems developers internally. Whereas one found significant differences between the size of the IT-department and the number of systems developers and the proportion of portfolio upkeep in the investigation from 1993 [20], such significant results have not been found in the later investigations [7,10,21], thus we do not expect that the difference in average size of organization or data department to have a large influence on the result relative to the distribution of time. It might influence some of the other numbers, e.g. the use of COBOL as reported in the next section appears to have increased. The average experience in the local IS-department was 5.6 (2003 – 5.4; 1998 – 6.3; 1993 – 6.4) years.

4.1. Portfolio analysis and replacement systems

The mean number of main systems in the organizations was 7.8 (2003 – 4.5; 1998 – 9.6; 1993 – 10.3). Main systems were defined as systems being vital for performing the activities of the organization. A large number of smaller systems, spreadsheets, etc. typically exist in most organisations, but they usually have a small user base, and limited overall impact on the IT-activities. The mean user population of these systems was 4661 (2003 – 314; 1998 – 498; 1993 – 541) persons. It is in particular the number of external users that has increased dramatically (also relative to number of employees in the organizations); the average number of internal users was 944, i.e. 88% on average of the employees in the organizations (2003 – 64%, 1998 – 76%, 1993 – 23%). The age distribution of the systems in our studies and the Swanson/Beath study is provided in Table 1. The average age of the systems was 4.9 years (2003 – 3.9; 1998 – 6.4; 1993 – 4.6; Swanson/Beath – 6.6).

In 1993, 58% of the systems were developed by the IS-organization, and only 1% was developed *in* the user organisation. In 1998, however, 27% of the systems were developed by the IS-organisation and 27% as custom systems *in* the user organisation. In 2003, 23% of the systems are developed in the IS-organisation, whereas in 2008 only 12% was developed in the IS organisation. The percentage of systems developed by outside firms is higher in 2008 (40% vs. 35% in 2003, vs. 22% in 1998 vs. 12% in 1993 vs. 15% in Swanson/Beath). The percentage of systems developed based on packages with small or large adjustments is also comparatively high (41% in 2008 vs. 39% in 2003 vs. 24% in 1998 vs. 28% in 1993 vs. 2% in Swanson/Beath). The new category we introduced in 1998, component-based development (renamed "use of external web services" in 2008) is still small (5% in 2008) although increasing (1.0% in 2003, 0.4% in 1998).

The average number of different programming languages in use for the main systems was 2.5 (median 2). This is similar to the investigation in 2003, 1998 and 1993. Table 2 provides an overview of the percentage of systems reported being developed using the different programming languages. As we see, from being dominant ten to fifteen years ago, COBOL is very little used in this century. Many other old languages are no longer used neither.

The languages that are used in most organizations and for most systems are now Java (40%, 27% in 2003) and C++ (33%, 24% in 2003). Java was just starting to be in widespread use in 1998 and C++ was barely included in 1993. The percentage of organizations reporting to have COBOL applications has decreased from 73% in 1993 to 26% in 1998 to 5% in 2008.

94 new systems were currently being developed, and as many as 60 of these systems (64%) were regarded as replacement systems (2003 – 60%; 1998 – 57%; 1993 – 48%; SB – 49%). The portfolio of the organisations responding to this question contained 446 systems, meaning that 13% of the current portfolio was being replaced (2003 – 13%; 1998 – 9%; 1993 – 11%; Swanson/Beath – 10%). The average age of systems to be replaced was 7 years (2003 – 5.5 years; 1998 – 7.7 years; 1993 – 8.5 years).

Table 3 summarizes reasons for the replacements, which have changed slightly from earlier investigations. The most important reasons for replacement in the 2008 investigation are partly a need for integration and burden to maintain and operate which appears to become more important again, a bit surprising giving the relatively young age of the systems that are replaced. On the other hand, the web-systems developed over the last 10 years relates to much more demanding and volatile operating environments than the primarily in-house systems developed earlier.

One area which is expected to influence the software development and maintenance landscape is Service Oriented Architecture (SOA) [25]. Transferring to SOA was very important as a reason to create replacement systems for only two organizations. This is not shown directly in Table 3, where we have put SOA as a replacement reason together with the development of new technical architecture in general (thus there are other architectural changes that for many are more important than SOA). Less than 20% of the organizations had started implementing SOA, and we could not find any significant impact on the use of SOA yet on maintenance figures.

4.2. Distribution of work

Work on application systems was in the survey divided into the six categories presented in Section 2. The same categories were also used in 1993, 1998 and 2003. We also asked for the time used for user-support and for systems operations which took up the additional time for the work in the IS departments. Tasks relating to management of the IS-department are spread relative to the proportion of other tasks.

Table 4 shows the distribution of work in previous investigations by us and by others, listing the percentage of maintenance work, the study reported, and the year of the study. Based on this we find that in most investigations of this sort, between 50% and 60% of the effort is done to enhance systems in operation (maintenance), when disregarding other work than development and maintenance. An exception from this was our study in 1998 that was found to be influenced particularly by the amount of Y2K-oriented maintenance [10]. The numbers from Dekleva and Capers Jones were also higher than 60%, but these also include user sup-

Table 1
Age distribution of systems, percentage of systems in each age interval.

Age of systems	2008	2003	1998	1993	Swanson/Beath
0–1	11	20	7	13	7
1–3	25	37	19	38	17
3–6	32	27	33	22	24
6–10	25	8	23	18	26
>10	7	8	18	9	26

Table 2
Percentages of systems developed using different programming languages.

Language/Investigation	2008	2003	1998	1993	Swanson/Beath	Nosek/Palvia	Lientz/Swanson
COBOL	4.5%	0.5%	32.6%	49%	63%	51%	51.6%
Different 4GL	12%	13.5%	16.9%	24%		8%	
C	2.5%	12.5%	15.4%	4%		3%	
C++	17.2%	23.1%	15.1%				
C#	5.2%						
RPG/Script	6.6%		12.9%	4%	2%	10%	22.4%
Java	22.6%	29.8%	2%				
Assembler	0.4%		0.9%	3%	8%		11.9%
Fortran			0.6%	4%	2%	7%	2.4%
PASCAL			0.3%	2%			
PL/1			0.3%	2%	25%		3.2%
Other	29.4%	20.2%	2.6%	6%		21%	7.7%

Table 3
Reasons for replacement of systems.

Factor	Investigation	Order	Extreme importance%	Substantial importance%	Moderate importance%	Slight importance%	No importance%	Mean[a]
Integration of systems (h)	2008	1	29	39	14	4	14	3.7
	2003	1	30	33	21	8	15	3.4
	1998	2	23	40	0	10	27	3.2
	1993	1	32	40	20	0	8	3.9
Burden to maintain (a)	2008	2	24	44	18	4	9	3.7
	2003	4	8	23	29	27	12	2.9
	1998	3	30	13	13	20	23	3.1
	1993	2	45	21	10	10	14	3.7
	Swanson/ Beath	1						3.8
Burden to operate (b)	2008	3	16	30	34	9	11	3.3
	2003	6	4	12	42	25	17	2.6
	1998	5	0	13	33	27	27	2.3
	1993	6	8	16	32	16	28	2.6
	Swanson/ Beath	3						2.8
New technical architecture (f)	2008	4	11	41	18	5	25	3.1
	2003	3	10	25	33	15	17	3.0
	1998	4	27	23	7	3	40	2.9
	1993	2	35	24	21	17	4	3.7
Standardisation (g)	2008	5	20	27	13	13	27	3.0
	2003	2	29	19	21	12	19	3.3
	1998	1	20	50	3	7	20	3.4
	1993	4	17	29	25	0	29	3.0
Burden to use (c)	2008	6	12	16	37	21	14	2.9
	2003	6	4	21	33	19	23	2.6
	1998	7	0	17	10	43	30	2.1
	1993	4	20	1	32	16	28	3.0
	Swanson/ Beath	1						3.8
Package alternative (d)	2008	7	7	25	16	23	29	2.6
	2003	5	12	17	29	17	25	2.8
	1998	6	7	13	13	17	50	2.1
	1993	7	9	22	13	17	39	2.4
	Swanson/ Beath	4						1.9
Generator alternative (e)	2008	8	2	9	9	33	47	1.9
	2003	8	0	6	23	29	42	1.9
	1998	8	0	7	13	17	63	1.6
	1993	8	0	5	25	15	55	1.8
	Swanson/ Beath	5						1.3

[a] The use of a mean value here is only to have the possibility to get a crude comparison with the investigation by Swanson/Beath, which did not report the distribution. This number has been calculated by giving the value 5 to extreme importance, 4 for substantial importance etc. Formally, the scale used here is neither a ratio nor an interval scale, thus it is not meaningful to use the mean value for further analysis.

Table 4
Result on maintenance from previous investigations on maintenance (disregarding other work than development and maintenance).

Percentage maintenance	Investigation	Year
49	Arfa et al. [1]	1990
52	Caper Jones [14]	1995
53	Lientz/Swanson[27]	1980
56	Jørgensen [15]	1994
58	Nosek/Palvia [31]	1990
58	Yip [37]	1995
59	Krogstie/Sølvberg[18]	1993
62	Krogstie et al. [21]	2003
66	Dekleva [8]	1990
72	Holgeid et al. [10]	1998
73	Capers Jones [14]	2000
79	Capers Jones [14]	2010 est.

port as part of maintenance contrary to what we have done in our surveys. We will return to this below.

Table 5 summarizes the descriptive results on the distribution of work in the categories in our investigation.

Table 5
Distribution of the work done by IS-department, percentage of overall work effort.

Category	Mean	Median	SD
Corrective maintenance	8.2	5.6	8.0
Adaptive maintenance	6.2	5.3	5.5
Enhancive maintenance	11.3	10.0	10.5
Non-functional perfective maintenance	9.1	5.9	7.9
Total amount of maintenance	**34.9**	**35.7**	**17.6**
Replacement	9.7	6.6	9.1
New development	11.4	5.9	13.8
Total amount of development	**21.1**	**17.6**	**16.4**
Technical operation	23.7	22.2	16.5
User support	20.1	15.8	19.0

Looking first on the distribution of maintenance activities, as also discussed in section two, a number of later studies have been looking at this in particular. As reported in [12], it appears to be very large differences reported in different studies. Whereas Lientz and Swanson [27] reported 60% perfective, 18% adaptive and 17% corrective maintenance when asking about selected systems from a large number of organisations (one per organisation), Sousa and

Table 6
Comparisons of distribution on maintenance tasks, percentage of maintenance work.

Category	2008 Mean	2003 Mean	1998 Mean	1993 Mean	Lientz/ Swanson Mean
1 Corrective maintenance	24	24	31	26	17
2 Adaptive maintenance	19	20	20	10	18
3 Enhancive maintenance	30	35	37	51	
4 Non-functionalperfective maintenance	27	21	13	13	
Perfective maintenance (3+4)	57	56	50	64	60

Moreira [34] reported (based on a number of systems in one organisation) 49% adaptive, 36% corrective, and 14% perfective maintenance. Schach et al. [33] reported 53% corrective, 36% perfective and 4% adaptive maintenance, based on data on three open source products. Min-Gu Lee and Jefferson [23] reported 62% perfective, 32% corrective, and 6% adaptive maintenance based on data from one application in production. Table 6 summarizes the results from our investigations where we look upon the complete portfolio of a number of organisations. Most interesting for comparison with other surveys is looking at corrective, adaptive, and perfective maintenance, which appears to be much more stable than the numbers reported from others above. We do note though that the enhancive maintenance part of perfective maintenance appears to be declining. We use this as a basis for investigating H2 below.

In Table 7 we compare our results on the level of development and maintenance. 34.9% of the total work among the responding organizations is maintenance activities, and 21.1% is development activities. When disregarding other work than development and maintenance of application systems, the percentages are as follows: maintenance activities: 65.7%, development activities: 34.3%. This is similar to what was reported in 2003. Since the amount of outsourced work is as earlier stated large (around a third of the overall activity when excluding those that have outsourced all IT-activities), we also looked at the numbers taking this into account. The result was a distribution of work of 21.1% development, 34.1% maintenance, 28.2% operations and 16.6% on user support (a larger percentage than average of operations are outsourced, and a smaller amount of user support is), thus not influencing the overall picture as for the distribution of work between development, maintenance, and other work. 63% of development and maintenance work was application portfolio upkeep, and 37% was application portfolio evolution. This is very similar to what was reported in 2003 and 1998, which in turn was significantly different from the situation in 1993 where application portfolio upkeep- and application portfolio evolution respectively amounted to 44% and 56% of the work. Further comparisons of descriptive results between different studies are also presented in Table 7. The first column lists the category, whereas the other columns list the numbers from our investigation in 2008, the one in 2003, the one in 1998, the one in 1993, the Nosek/Palvia investigation and the Lientz/Swanson investigation. The first set of number compare the numbers for development, maintenance and other work. The amount of *other work* reported in our investigations is much larger than in the American investigations. Therefore, in the second set of figures, we compare the data without considering other work than maintenance and development. For the categories application portfolio evolution and application portfolio upkeep, we only have numbers from our own investigations. We look below on hypothesis H1, H3, H5 and H6 in more detail with closer comparisons between the numbers from 2003 and 2008.

Given that some authors include user support as part of maintenance, Table 8 provides a similar overview, but where user-support is included as part of maintenance (and as application portfolio upkeep). For this we only have data from our own studies from 1998, 2003, and 2008, and also this appears to be on a quite stable level. Taking outsourcing into account, the amount of maintenance including user support is a bit lower in 2008 than what is listed here. We look below on hypothesis H4 in more detail with closer comparisons between the numbers from 2003 and 2008.

Before looking for significant relationships to follow up the stated hypothesis related to trends, the variables used in the comparisons were tested for normality as illustrated in Table 9. We provide here data to test the distribution of the relevant variables from the 2003 and 2008 investigation. As indicated by the number in boldface, data for a number of variables cannot be investigated as if they where normally distributed, since we in these cases must reject the null-hypothesis that the numbers are normally distributed (either from 2003 or from 2008), since either the Shapiro–Wilks (S–W Sign) and/or the Kolmogorov–Smirnov (Lilliefors-Sign) significance levels are less than 0.05. On some variables (amount of maintenance disregarding other work than development and maintenance, and application portfolio upkeep) we could use the assumption of normal distribution in the tests below, using *t*-tests. For the others we have used non-parametric tests as described in Section 3.

We tested H1–H6 by comparing with our previous survey as summarised in Table 10.

We list the number of cases, the mean and the standard deviation for all relevant figures to test the eight hypotheses (for H2, there are three test, for the difference in corrective, adaptive, and

Table 7
Comparisons of maintenance figures with previous investigations.

Category	2008	2003	1998 Mean	1993 Mean	Nosek/ Palvia Mean	Lientz/ Swanson Mean
Percentage of all work						
Development	21	22	17	30	35	43
Maintenance	35	37	41	40	58	49
Other work	44	41	42	30	7	8
Disregarding other work than development and maintenance						
Development	34	34	27	41	38	47
Maintenance	66	66	73	59	62	53
Functional effort, disregarding other work than development and maintenance						
Application portfolio evolution	37	39	38	56	N/A	N/A
Application portfolio upkeep	63	61	62	44	N/A	N/A

Table 8
Comparisons of maintenance figures, where user support is include as part of maintenance.

Category	2008 Mean	2003 Mean	1998 Mean
Percentage of all work			
Development	21	22	17
Maintenance	55	54	60
Other work	24	24	24
Disregarding other work than development and maintenance			
Development	26	28	22
Maintenance	74	72	78
Functional effort, disregarding other work than development and maintenance			
Application portfolio evolution	28	31	32
Application portfolio upkeep	72	68	68

Table 9
Test for normality of maintenance figures.

Figure	Kolmogorov–Smirnov	Sign (p)	Shapiro–Wilks	S-W Sign (p)
Corrective maintenance 2003	.111	.153	.968	.180
Corrective maintenance 2008	.116	**.047**	.907	**.000**
Adaptive maintenance 2003	.147	**.007**	.848	**.000**
Adaptive maintenance 2008	.087	.200	.942	**.007**
Perfective maintenance 2003	.115	.086	.974	.312
Perfective maintenance 2008	.127	**.019**	.948	**.014**
Maintenance 2003	.127	**.034**	.967	.155
Maintenance 2008	.066	.200	.986	.751
Development 2003	.144	**.009**	.917	**.002**
Development 2008	.111	.066	.946	**.011**
Maintenance relative to development 2003	.079	.200	.970	.202
Maintenance relative to development 2008	.076	.200	.971	.177
Maintenance including user support 2003	.146	**.007**	.971	.231
Maintenance including user support 2008	.091	.200	.972	.157
Application portfolio upkeep 2003	.103	.200	.970	.220
Application portfolio upkeep 2008	.076	.200	.977	.310

perfective maintenance respectively), Δ is the absolute difference in the mean from the 2008 and the 2003 study, and p is the probability for erroneously rejecting the equality of means. None of H1–H6 is rejected showing a very stable pattern. We will discuss these results further after discussing the validity of the study in the next section.

5. Validity of study

The results of our study should be interpreted cautiously as there are several threats to its validity. The discussion below is based on recommendations given in [15,17].

5.1. Population

The sample of our study is intended to represent the population of Norwegian companies or organizations with own development and maintenance work. Since a substantial number of the major Norwegian companies are members of The Norwegian Computer Society (NCS) we pragmatically chose the around 1000 member companies of NCS as our population. This includes in addition to all large organizations in Norway, also organizations within the IT-industry. We made clear that in it was their own IT-activity we were interested in (and not what they did for other organisations). Some of the responses that we had to dismiss, were from IT-companies not reporting on own activities (and thus not being able to answer more than a few introductory questions). We distributed our survey forms to a random selection of 300 of the member companies of NCS. (As described before, some of these bounced for different reasons.) Other studies also use member lists as a source of subjects, e.g. [27]. In particular, the same source of subjects was used in our studies in 2003, 1998 and 1993. As noted in Section 3, the actual responding organizations have changed a lot between the different studies which is inevitable in such studies as long as we are not actually trying to follow some selected organizations our time. When comparing with the American studies, we should be even more cautious in the comparison, given potential cultural differences. We note though that when these comparisons where first provided [18–20], the results were well received by the software maintenance community.

5.2. Response rate

The response rate of 28% can be argued to be low, although it is higher than in the previous investigations of this kind (both in Norway and in the US). We experienced the same in 2003, 1998 and 1993 (with even lower response rate). A problem with a low response rate is that the respondents may not be representative of the population, that is, the companies may be particularly mature, have less pressure (they have time to answer survey forms), etc. However, the same selection mechanism was used in the 2003, 1998 and 1993 studies, so a comparison between those four studies should be reasonable.

5.3. Respondents

Most of the persons who responded were IT managers in the company. They may have different views of the reality than developers and maintainers. For example, Jørgensen [15] found that manager estimates of the proportion of effort spent on corrective maintenance too high when based on best guesses instead of good data, see also [33] which report a similar effect. We could also find a slight tendency in this direction judging from the quality of the

Table 10
Tests of inter-investigational hypotheses.

	Year	N	Mean	SD	Δ	p
Maintenance, percentage of all work (vs. H1)	2008	63	34.9	17.6	1.8	.884
	2003	52	36.7	15.5		
Corrective maintenance, percentage of all work (vs. H2a)	2008	59	23.8	16.8	0.3	.903
	2003	52	23.5	13.9		
Adaptive maintenance, percentage of all work (vs. H2b)	2008	59	19.1	14.1	2.3	.376
	2003	52	22.3	16.8		
Perfective maintenance, percentage of all work (vs. H2c)	2008	59	57.0	19.8	2.8	.470
	2003	52	54.2	21.6		
Development, percentage of all work (vs. H3)	2008	63	21.1	16.4	1.4	.774
	2003	52	22.5	17.9		
Maintenance including user support (vs. H4)	2008	63	55.1	16.4	1.4	.879
	2003	52	53.7	16.8		
Maintenance, disregarding other work (vs. H5)	2008	61	65.7	21.5	0	1.000
	2003	52	65.7	21.6		
Application portfolio upkeep (vs. H6)	2008	61	63.0	20.9	1.9	.613
	2003	52	61.1	20.2		

data (see below), but this difference was not significant. There might be biases in our study, but they may not affect the comparison with the 2003, 1998 and 1993 studies as the survey forms of these studies were also filled in by IT managers. Also in the American studies we compare with, IT managers have responded to the surveys.

5.4. Understanding of concepts

Achieving consistent answers requires that the respondents have a common understanding of the basic concepts of the survey form. This may be difficult to achieve in practice. For example, Jørgensen [15] found that the respondents used their own definition of, for example, "software maintenance" even though the term was defined at the beginning of the questionnaire. We conducted a pilot study in a few companies to detect unclear questions and whether the time for filling-in the forms was reasonable. We also got comments from several colleagues including experts in cognitive psychology which are highly familiar with the use of survey techniques concerning clarity of questions. The forms were refined based on feedback from the pilots. For many questions, there was space available to issue comments. This possibility together with the possibility to crosscheck numbers between different questions was the mechanisms used to identify possible misunderstandings among the respondents, which were followed up afterwards. We also built upon earlier surveys that had undergone similar pilot and full use, where some of the questions had been refined earlier.

5.5. Biased questions

Among the risks when designing survey forms are leading or sensitive questions, resulting in biased or dishonest answers. We believe that we have mostly avoided this problem. One exception may be the question of whether IT is of strategic importance. It may be difficult for IT managers to admit that IT is of little strategic importance to their company. Nevertheless, we promised and effectuated full confidentiality.

5.6. Quality of data

On some of the questions, we were interested in the quality of the answers, recognizing that some of the information called for might not be easily obtainable. Answers of some of the quantitative questions were checked against each other for control, and where followed up with the respondents afterwards when there were inconsistencies. The remarks made on the questions gave more insight into the answers. We qualified for instance all data regarding distribution of work both in our study and the studies in 1993, 1998, and 2003 without finding significant differences on the variables we have used in the hypothesis testing between those reporting having good data and those coming with qualified guesses [10,18,21].

6. Conclusion and further work

Revisiting our hypotheses, we conclude the following:

- H1: There is no difference between the percentage of maintenance time in our survey and what are reported in previous surveys.

Not rejected: The overall time used for maintenance tasks is also in our investigation around 40% on average (when not including user support as maintenance. Obviously, as indicated by the SD-numbers, there are large variation between different organizations in the different years, but over a number of organizations, being in different phases (e.g. after a year or two with large proportion of development, there are larger proportion of maintenance the following years), this seems to even out.

- H2: There is no difference between the breakdown of maintenance work (in corrective, adaptive, and perfective maintenance) in our survey and what are reported in previous surveys.

Not rejected: Although other work has reported large differences here, we found a very stable distribution, being similar also to the Lientz/Swanson survey. Looking at software life-cycle models that also take maintenance into account (see e.g. [32]) it should come as no surprise that the maintenance distribution differs between systems in different phases (the amount of perfective maintenance would be expected to be larger when the system is in the evolution phase, than when it is in the servicing and phase-out phase). When we look upon this across a number of application portfolios on the other hand, these differences appear to be evened out.

- H3: There is no difference between the percentage of development time in our survey and what are reported in previous surveys.

Not rejected: Comparing with 2003, there is no significant change. On the other hand, the amount of work related to development activities had declined from 30% in 1993 to 17% in 1998. This had risen (but not significantly) to 22% in 2003 and 21% in our last study in 2008. In our study, the total amount of development and maintenance activity adds up to close to 60% (as it also did in 2003 and 1998). Technical operation and user support account for 44% of the total work in the IS departments of the responding organisations. Compared with the study in 1993 and the earlier studies in the US, the amount of work related to technical operation and support of users has increased on expense of time left to new development.

- H4: There is no difference between the percentage of combined maintenance time and time used for user support in our survey and what are reported in previous surveys.

Not rejected: Based on reports e.g. from Jones [14] it was expected that this would increase. As we see in Table 8 and the hypothesis testing, it appears that this is also stable, a total of 75–80% of the time is used to maintenance when including user support (and disregarding other work), and this has been stable over the last 10 years. Given the increasing number of users (and in particular external users) this can actually be regarded as a positive result relative to proving more users with appropriate functionality, without using a larger percentage of the work force to do it.

- H5: There is no difference between the distribution of work among maintenance and development in our survey and what is reported in previous surveys when disregarding other work than development and maintenance.

Not rejected: When disregarding other work than maintenance and development activities, there is very little difference from the previous survey in 2003.

- H6: There is no difference between the distribution of application portfolio upkeep and application portfolio evolution in our survey and what is reported in previous surveys.

Not rejected: 63% of development and maintenance work was application portfolio upkeep; and 37% was application portfolio evolution. This is almost the same as in 2003 and 1998, which in turn was a significant change from the situation in 1993 where application portfolio upkeep and application portfolio evolution respectively amounted to 44% and 56% of the work. The study in 1993 indicated that larger data departments had a more widespread use of organisational controls and methodology, and thus performed better regarding application portfolio upkeep. This pattern is more pronounced in the survey from 1993 than in the one from 1998 though, which appears to have been influenced by the specific situation in connection to Y2K. We did not find such a pattern in last investigations.

Some indications were given that the amount of Y2K-oriented work had a significant impact on the result in 1998, such as the number of replacement systems that were Y2K-oriented. Y2K-work affected both the maintenance figures (when existing systems were kept) and application portfolio upkeep figures (when systems were replaced, or maintenance budget were used up for Y2K-fixes instead of being used for adding new functionality to existing systems). We investigated this further in several organisations. Many of those used more than 10% of the budget for development and maintenance on Y2K fixes or replacements in 1998. If this trend were general, it would explain *some* of the difference, but not all of it. We would expect that this effect would be erased in 2003 and at least in 2008. Still we see that the numbers for application portfolio upkeep is as high as those in 1998. At first sight this is worrying. On the other hand, we see how the application portfolios of organisations are utilised by more and more people, both inside and in particular outside the organisation. Development of new systems are to a larger extend done further away from the organisation, see e.g. how the number of systems being developed in the IT or user-organisation is decreasing, thus one is more dependent on the skills of external providers of services, and seems thus not so able to do as much enhancive maintenance as before.

6.1. Future work

Several of our results have spurred new areas that could be interesting to follow up in further investigations, either in the form of further surveys, or more likely by developing additional detailed case studies. To come up with better empirical data onto what extent the application systems support in an organisation is efficient, would take another type of investigation, surveying the whole portfolio of the individual organisation, and getting more detailed data on the amount of the work that is looked upon as giving the end-user improved support, and how efficient this improved support is provided. This should include the views of the users of the application systems portfolio in addition to those of the IS-managers and developers. We have in connection to this work performed a number of more in-depth interviews (not reported on here) in selected organizations. On the other hand it is hard to generalise from such studies, as can be witnessed from some of the studies looking at the distribution on maintenance tasks referred to above. We would also like to investigate more closely the reasons for more or less efficient application systems support in an organisation, taking into account theories on system evolution [24,28,30,32], given the impact witnessed by macro-phenomena in the environment and the aspect of sourcing, which in general seems be done further and further from the core (internal) users of applications. In connection to this it would also be interesting to look at how the magnitude of different types of maintenance evolves as the system evolves, to help support planning [22,32].

A long-term plan is to do a similar investigation in 2013, following up our 5-year cycle and compare with the four existing consecutive investigations.

Acknowledgements

We would like to thank all the participants of the survey-investigation for their effort in filling in the forms. We would also like to thank everyone helping us in the piloting and refinement of the questionnaire, and The Norwegian Computer Society for practical support in the arrangement of the survey. Finally thanks to the reviewers for useful comments and to Birgit Krogstie helping with the layout of the tables.

Appendix A. Contents of the survey form

Below is listed the main questions from the survey form. This is not an exact copy of the form used. For reasons of brevity, we have only included the questions relevant to the results presented in the paper. We have below changed the layout and removed most of the room for giving additional information and qualification of the answers provided in the SurveyMonkey forms. We have neither included the additional material explaining the format and vocabulary used in the form. The survey form content has also been translated into English from Norwegian.

4. Current position: ___Manager
___Project manager
___System developer, designer etc.

7. Years of computer experience: ____

9. Type of organisation (Telecom, banking, etc.):_____

10. Is IT of strategic importance to your organisation?

Extremely __5 __4 __3 __2 __1 No *importance*

11. Number of employees in your organisation: ____

12 What is the annual budget of the IS-organisation in 2008 including hardware, software and personnel (in mill. NOK)?

		2008
a.	more than 50	___
b.	between 40 and 50	___
c.	between 30 and 40	___
d.	between 20 and 30	___
e.	between 10 and 20	___
f.	between 1 and 10	___
g.	less than 1	___

13. How much of the following activity is outsourced:

a. ___ The total IT-activity (%)
b. ___ Development of new applications (%)
c. ___ Maintenance of existing application (%)
d. ___ Operations(%)
e. ___ User support (%)
f ___ Other specify: ____________________________

15. Distribute your IS department's work into the following categories:

%
a. ___Correcting errors in systems in operation
b. ___Adapt the system to changed technical architecture
c. ___Develop new functionality in existing systems
d. ___Improve non-functional properties (e.g. performance)
e. ___Develop new systems which provide similar functionality as existing systems
f. ___Develop new systems to cover new functional areas
g. ___Operation
h. ___User support
i. ___Other, specify: ____________________________
Total: **100%**

17 Your answer above is:
a. ___Reasonable accurate, based on good data
b. ___A rough estimate, based on minimal data
c. ___A best guess, not based on any data

18. Specify the number of full-time positions in the IS department? _______ positions

19. How many of these positions are dedicated to system developers? _______ positions

20. Specify the distribution of system developers with respect to their experience in the IS department?

Years	Persons
0-1	_______
1-3	_______
3-6	_______
6-10	_______
> 10	_______

21. What is the annual average number of hire IT consultants (converted to full-time personnel?
_______ persons

23. Specify the number of current main systems in your organisation _______ systems

24. What is the total number of end-users?
Internal_______ External_______

29. Specify age of the main systems (years since first installation)?

Years	Systems
0-1	______
1-3	______
3-6	______
6-10	______
> 10	______

30. What is the distribution of development backgrounds of the current installed application system portfolio?

Developed by the IS department	______ systems
Developed by the user organisation	______ systems
Developed by outside firm	______ systems
Package with large internal adjustments	______ systems
Package with small internal adjustments	______ systems
Solutions based on external web-services	______ systems

31. Of the systems in the current installed application system portfolio, how many rely on other systems for their input data? _______ systems

32. Which programming languages are in use? Please specify number of systems developed in each programming language.

Language	Number of systems	
COBOL	_______	
Assembler	_______	
C	_______	
C++	_______	
C#	_______	
Java	_______	
Scripts (PHP/Pearl etc)	_______	
4GL	_______	Specify
Other	_______	Specify

38. What is the number of systems currently being developed?
_______ systems

39. Of the total number of new systems currently under development, how many of these are replacement systems (for systems currently in the application system portfolio)?
_______ systems

40. What is the age distribution of the systems to be replaced?

Years	
41-10	_______ systems
1-3	_______ systems
3-6	_______ systems
6-10	_______ systems
>10	_______ systems

41. In the case of systems to be replaced in the current installed application system portfolio, what are the important reasons for replacement?
(For each statement below, pick a score from 1 to 5, 5 being most important)

a.	Excessive burden to maintain the system	______
b.	Excessive burden to operate the system	______
c.	Excessive burden to use the system	______
d.	Existence of application package alternative	______
e.	Existence of application generator alternative	______
f.	Changes in technical architecture (e.g.SOA)	______
g.	Standardisation with rest of organisation	______
h.	Integration with new or existing systems	______
i.	Other, specify: ____________________________	______

5: Extreme importance
4: Substantial importance
3: Moderate importance
2: Slight importance
1: No importance

References

[1] L.B. Arfa, A. Mili, L. Sekhri, Software maintenance management in Tunisia: a statistical study, in: Proceedings of ICSM 1990, IEEE, 1990, pp. 124–129.
[2] L. Bergersen, Prosjektadministrasjon i systemutvikling: Aktiviteter i planleggingsfasen som påvirker suksess. PhD thesis, ORAL, NTH, Trondheim, Norway, 1990 (in Norwegian).
[3] B. Boehm, K. Sullivan, Software economics: status and prospect, Information and Software Technology 41 (14) (1999) 937–946.
[4] F.P. Brooks, No silver bullet. Essence and accidents of software engineering, IEEE Computer 20 (4) (1987) 10–19.
[5] N. Chapin, Software maintenance types – a fresh view, in: Proceedings of the International Conference on Software Maintenance (ICSM'2000), 2000, pp. 247–252.
[6] N. Chapin, J. Hale, K. Khan, J. Ramil, W.-T. Tan, Types of software evolution and software maintenance, Journal of Software Maintenance 13 (2001) 3–30.
[7] M.K. Davidsen, A Study of Development and Maintenance in Norway, Technical Report IDI, NTNU, 2009.
[8] S.M. Dekleva, Software maintenance: 1990 status, Journal of Software Maintenance 4 (1992) 233–247.
[9] J. Galtung, Theory and Method of Social Research, Universitetsforlaget, Oslo, Norway, 1967.

[10] K.K. Holgeid, J. Krogstie, D.I.K. Sjøberg, A study of development and maintenance in Norway: assessing the efficiency of information systems support using functional maintenance, Information and Software Technology 42 (2000) 687–700.
[11] IEEE Standard Glossary of Software Engineering Terminology, 1991.
[12] A. Gupta, The profile of software changes in reused vs. non-reused industrial software systems, Doctoral thesis 2009:90 NTNU.
[13] A. Gupta, O.P. Slyngstad, R. Conradi, P. Mohagheghi, H. Rønneberg, E. Landre, An empirical study of software changes in statoil ASA – origin, priority level and relation to component size, in: Proceedings of the International Conference on Software Engineering Advances (October 29–November 03, 2006), ICSEA, IEEE Computer Society, Washington, DC, 2006, p. 12.
[14] C. Jones, The Economics of Software Maintenance in the Twenty First Century, 2006. <http://www.compaid.com/caiinternet/ezine/capersjones-maintenance.pdf> (last accessed on April 2009).
[15] M. Jørgensen, Empirical Studies of Software Maintenance, PhD thesis University of Oslo, 1994, Research Report 188, ISBN 82-7368-098-3.
[16] M. Kajko-Mattsson, C. Ahnlund, E. Lundberg, CM3: service level agreement, in: Proceedings of the 20th IEEE International Conference on Software Maintenance (ICSM'04), 2004.
[17] B.A. Kitchenham, S.L. Pfleeger, L.M. Pickard, P.W. Jones, D.C. Hoaglin, J. Rosenberg, Preliminary guidelines for empirical research in software engineering, IEEE Transactions on Software Engineering 28 (8) (2002) 721–734.
[18] J. Krogstie, A. Sølvberg, Software maintenance in Norway: a survey investigation, in: H. Muller, M. Georges (Eds.), Proceedings of the International Conference on Software Maintenance (ICSM'94), IEEE Computer Society Press, Victoria, Canada, 1994, pp. 304–313.
[19] J. Krogstie, Use of methods and CASE-tools in Norway: results from a survey, Journal of Automated Software Engineering 3 (1996) 347–367.
[20] J. Krogstie, On the distinction between functional development and functional maintenance, Journal of Software Maintenance 7 (1995) 383–403.
[21] J. Krogstie, A. Jahr, D.I.K. Sjøberg, A longitudinal study of development and maintenance in Norway: report from the 2003 investigation, Information and Software Technology 48 (2006) 993–1005.
[22] H.-J. Kung, Quantitative method to determine software maintenance life-cycle, in: Proceedings of the 20th IEEE International Conference on Software Maintenance (ICSM'04), 2004.
[23] Min-Gu Lee, Theresa L. Jefferson, An empirical study of software maintenance of a Web-based Java application, in: ICSM'05, 2005.
[24] M.M. Lehman, Laws of software evolution revisited, in: Proceedings from EWSPT '96, LNCS, vol. 1149, 1996, pp. 108–124.
[25] G.A. Lewis, D.B. Smith, Service-Oriented Architecture and its Implications for Software Maintenance and Evolution Frontiers of Software Maintenance (FoSM 2008), 2008.
[26] B.P. Lientz, E.B. Swanson, G.E. Tompkins, Characteristics of application software maintenance, Communications of the ACM 21 (6) (1978) 466–471.
[27] B.P. Lientz, E.B. Swanson, Software Maintenance Management, Addison Wesley, 1980.
[28] N. Madhavji, V. Nanda, The impact of environmental evolution on requirements changes, in: Proceedings of ICSM'2002, 3–6 October 2002, Montréal, Canada.
[29] Parastoo Mohagheghi, Reidar Conradi, An empirical study of software change: origin, acceptance rate, and functionality vs. quality attributes, in: International Symposium on Empirical Software Engineering (ISESE'04), 2004, pp. 7–16.
[30] A.P. Nikora, J.C. Munson, Understanding the nature of software evolution, in: Proceedings of the 19th IEEE International Conference on Software Maintenance (ICSM'03), 2003.
[31] J.T. Nosek, P. Palvia, Software maintenance management: changes in the last decade, Journal of Software Maintenance 2 (1990) 157–174.
[32] V.T. Rajlich, K.H. Bennett, A staged model for the software life cycle, IEEE Computer 33 (2000) 66–71.
[33] S.R. Schach, B. Jin, L. Yu, G.Z. Heller, J. Offutt, Determining the distribution of maintenance categories: survey versus measurement, Empirical Software Engineering 8 (4) (2003) 351–365.
[34] H. Sousa, H. Moreira, A survey of the software maintenance process, in: Proceedings of ICSM'98, IEEE CS Press, Bethesda, MD, 1998, pp. 268–274.
[35] E.B. Swanson, The dimensions of maintenance, in: Proceedings of the Second International Conference on Software Engineering, San Francisco, USA, August 1976, pp. 492–497.
[36] E.B. Swanson, C.M. Beath, in: Maintaining Information Systems in Organizations, Wiley Series in Information Systems, John Wiley & Sons, 1989.
[37] S.W.L. Yip, Software maintenance in Hong Kong, in: G. Caldiera, K. Bennett (Eds.), Proceedings of the International Conference on Software Maintenance (ICSM'95), IEEE Computer Society Press, Nice, France, 1995, pp. 88–97.