Information and Software Technology 54 (2012) 866–886

Contents lists available at SciVerse ScienceDirect

Information and Software Technology

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

Exploratory case study research: Outsourced project failure

J.M. Verner [a,*], L.M. Abdullah [b]

[a] EPSAM, Keele University, Staffordshire ST5 5BG, United Kingdom
[b] Department of Information Systems, International Islamic University, 50728 Kuala Lumpur, Malaysia

ARTICLE INFO

Article history:
Available online 18 November 2011

Keywords:
Outsourced software development
Project failure
Software development risk factors
Case study methodology
Risk framework

ABSTRACT

Context: IT plays an increasingly strategic role in the business performance of organizations, however, the development of strategic IT systems involves a high degree of risk and outsourcing the development of such systems increases the risk.
Objective: Using a case study approach we build on research that identifies risk factors leading to the failure of outsourced strategic IT development projects. We investigate the BSkyB project, a strategic development project, which was the subject of recent litigation in the British High Court. We wish to discover what factors led to the failure of such a high profile project; in particular we wish to identify which factors were under the control of the client. We also review the usefulness of the case study methodology when it is not possible to interview any of the people involved with a project.
Method: Detailed-step-by-step guidelines designed for multiple industrial case studies are used to investigate the failure factors of the BSkyB project. We use transcripts of court proceedings and media reports to determine the failure factors. We compare the factors identified with those in a conceptual risk framework developed in prior research thus providing an initial validation of that framework.
Results: The following factors were identified as problems in the BSkyB project: contract, requirements, project complexity, planning and control, execution, and team. A time and materials contract was a risk not originally included in the risk framework that we used.
Conclusion: The BSkyB project failed because of problems that can be traced to both client and vendor. According to the judge's summing up the major fault was with the vendor, although some problems did emanate from the client side. We found that many sections in the case study methodology we used were unnecessary for a single case study based on court proceedings and media reports. The risk framework helped with risk identification.

1. Introduction

Because we are interested in factors that cause the failure of strategic outsourced software development projects we use a case study approach to build on research that identifies risk factors leading to the failure of such projects (e.g., [2,24,101,117]). As any organization planning to work on a strategic IT outsourcing project needs to address risk we wish to help the client of such systems identify the risks most likely to affect the outcome of their project. As successful software development is concerned with not only technical solutions but also organizational issues, project management and human behavior [45,46,90,118], the case study, as a research methodology (though more established in the social science research field), is gaining increased recognition in software engineering research. There is increasing interest in the methodology within this community as researchers realize that case study research can provide an understanding of what actually happens in the real world. A concrete example, such as the BSkyB case discussed here, although rather atypical not only helps with the identification of risk factors but assists in a better understanding of the use of the case study methodology and allows for an exploration of the case study boundaries. We use the detailed step-by-step case study research guidelines developed by Verner et al. [116], to investigate the failure factors for the BSkyB case, an outsourced software development project which was the subject of a high profile court case in the London Technology and Construction Court [22]. Although a standard case study methodology does not fit the BSkyB case in many ways [44,51,101,116], we felt that this project was important; it is a landmark case that should be investigated and discussed further. As the methodology we use is intended for multiple industrial cases studies [116] not all sections may be appropriate for this research. Accordingly when we apply the methodology to investigate BSkyB failure factors (discussed briefly in [2] and written before judgment was made [22]), we comment when the methodology we use differs from that normally used, and discuss how useful we found the methodology.

* Corresponding author. Address: School of Computing and Mathematics, Keele University, Staffordshire ST5 5BG, United Kingdom. Tel.: +44 7917784678.
E-mail address: june.verner@gmail.com (J.M. Verner).

0950-5849/$ - see front matter © 2011 Published by Elsevier B.V.
doi:10.1016/j.infsof.2011.11.001

Our focus is on exploring or explaining a contemporary software engineering phenomenon within its real-life context. In particular why did a strategic outsourced software development project fail, and what failure factors were under the control of the client? Numbers of failed IT projects result in litigation and cross-litigation but the majority are settled out of court. When this occurs gag orders are normally imposed as part of the settlement; hence it can be difficult to discover what actually went wrong with such projects. With the BSkyB case, the actual litigation took nine months and cost the litigants over £45 million each in legal fees. The judge took 15 months to consider his verdict which awarded BSkyB £318 million. It has been suggested by some lawyers that this case could be the most important case thus far for the IT services sector and that it may alter the way that software development vendors do business [31]. The court documents are available online [22] and with the availability of these documents, we are able to investigate what actually happened in the BSkyB project.

While there is much published research on IT development and IT outsourcing risk, and risk management, research on risks for outsourced strategic IT system development from the client perspective, has largely been ignored. By studying the case of an unsuccessful outsourced strategic IT systems development project we gain greater insight into factors that threaten their success. Failure to understand and manage project risks can result in important losses, project failure, and subsequently affect the achievement of an organization's business objectives. Outsourcing the development of a strategic IT system, (particularly to a vendor in a different country, which is becoming more common and increases complexity), adds a high degree of uncertainty to an already risky business endeavor. Unsuccessful experiences with strategic IT development outsourcing have been reported, with not all projects achieving the intended benefits; some are terminated early [2]. The findings of our research should assist clients recognize and understand risk factors that affect such projects so that effective decisions and actions can be taken before the risks manifest themselves into problems that damage the project. The results are pertinent to clients' decisions and actions, and may improve or strengthen their practices and policies in this area.

This paper is organized as follows: in Section 2, we provide the background to our project failure research. In Section 3, a summary of the methodology used for this research is presented. Section 4 covers details of the data analysis, and we report our results. We conclude, in Section 5, with a discussion of the failure factors identified from this case, compare them with our initial risk framework, comment on our research methodology, discuss limitations to the research and suggest further work. An appendix provides further details of the methodology used for the research, including details of planning, constructing and reporting as suggested in [116] for multiple industrial case studies.

2. Background

Because IT plays such a strategic role in the business performance of organizations and the development of strategic IT systems involves innovation and entails a high degree of risk [28,122] we first define a strategic IT system, and discuss risk in more detail; we include a discussion of general project risks, outsourcing risks, then provide a summary of risks that are specific to strategic outsourced development projects. A very brief discussion of our risk framework which was synthesized from prior research follows. The development of this framework is described in more detail in [1].

We define a strategic IT system to be:

> *any type of hybrid IT system that may comprise internally or externally focused systems and is either applied to processes or practices that support the core product or service, or is part of the core product or service that the organization provides to its customers.*

Only a few research papers discuss risk in the context of strategic IT [43,29,58,119], and most of the research on IT project risk is focused on the vendor/development side of a project, identifying risks that threaten in-house systems or vendor software developments (e.g., [3,93,94]). Owing to an increasing level of IT outsourcing [79], the growing trend to global IT outsourcing [68,109], and most importantly, industry's growing concerns with project failures and dissatisfaction with IT outsourcing arrangements, research on risk in IT outsourcing has recently increased. Strategic IT outsourcing is now being pursued actively by vendors though unsurprisingly, there is little research recognition of the risks faced by the clients of outsourced strategic IT system development projects. Since the development of a strategic IT system and the development of any IT system have some common characteristics past research on IT risk cannot be discounted; all IT development projects require resources, requirements elicitation, planning and management, and the development processes are fundamentally similar. We now discuss general project risks, risks specific to outsourcing, risks associated with strategic IT outsourcing and the risk framework used in this research.

2.1. General project risks

The research on risk affecting IT development was first addressed in discussions about managing MIS development. In Alter and Ginzberg [4] eight risk factors were identified and mapped against a proposed MIS implementation process model. This mapping suggests that all risk factors are related to the early phases of a computer-based systems implementation process. There was little progress on IT development risk until Boehm [19,20] proposed a detailed software risk management model and the top ten sources of risk. Charette's research [25,26] complements Boehm's [19,20] by introducing the mechanics of software engineering risk analysis (identification, estimation and evaluation) and management (planning, controlling and monitoring) in detail, to guide its application in software development projects. Charette [25,26] suggests that the three primary causes of risk are: undercapitalization of resources, underperformance of resources, and lack of understanding of risk as it affects software acquisitions, development or application. He also showed that software risk exists within the process of developing the software, which includes the development process model, methods, techniques and/or the automation used to develop the product, and the product itself. Since then, more research has taken place with Barki et al.'s [12] measurement instrument containing twenty-three uncertainty factors grouped into five categories developed through a factor analysis of a list administered in the form of a survey.

In order to confirm that the risks mentioned in the literature actually reflect the concerns of real-world project managers, Moynihan [82] surveyed 14 experienced software project managers to identify situational factors they considered important when planning new development projects. His study showed similarities to Barki et al. [12], though the constructs developed also included further elaborations of some risk drivers. Jiang and Klein [54] also used the risk measures from Barki et al. [12] to examine the relationships between software development risks and project effectiveness. The findings from this survey of project managers indicate that the risks that have a significant impact on project effectiveness are lack of general team expertise, intensity of conflicts and lack of clarity of role definitions. In an attempt to theoretically relate the different risk management approaches available in the literature, Lyytinen et al. [75,76] examined software risk management based on the management of organizational change and consider causal dependencies between managerial action and observed events. When they surveyed project managers from Hong Kong, Finland and the United States, Schmidt et al. [96] attempted to provide a fresh, comprehensive and up-to-date list of risk factors, and to determine the importance to project managers of the various risk factors. Their survey resulted in 53 risk factors organized into 14 groups based on the source of the risk. Schmidt et al. [96] also compared their list with the previous work of Barki et al. [12], Boehm [19], and Moynihan [82], to see if there were any changes in risk factors over time. Their study suggested a more encompassing risk list than the risks discussed in previous studies. Three factors received a high ranking; they were, in order, lack of top management commitment; failure to gain user commitment; and misunderstanding of requirements. A framework that organized the risks into four categories (customer mandate, scope and requirements, execution, and environment), according to the project manager's perceived importance of the risk and perceived level of control was developed by Keil et al. [57]. The framework was later tested in a study by Wallace and Keil [120] who examined the relationships between risk factors, project process and product outcomes. The use of the framework in a multi-industry study of more than 500 software development projects indicated that risks associated with scope and requirements, and execution of the project affect the process outcome, while product outcome is affected by customer mandate, scope and requirements, and execution risks.

With risk management being increasingly recognized as providing a powerful approach for dealing with the complexities and uncertainties surrounding technological change and management, Baccarini et al. [10] studied perceptions of risk and risk treatment strategies among IT professionals in Western Australian firms. They identified the 27 most common risks in the literature and structured them according to Standards Australia's proposed risk categories [107]. The study found that the top five risks were personnel shortfalls; unreasonable project schedule and budget; unrealistic expectations; incomplete requirements; and diminished window of opportunity due to late delivery of software. Dey et al. [38] conducted a case study in a government agency in Barbados to test a risk management framework for software development projects from the developers' perspective. The executives and developers involved in the study identified and analyzed seven project risk events, and developed and implemented a risk management plan to respond to the risks. A survey was conducted by McLeod et al. [80], to investigate IS development practices in New Zealand organizations; fourteen factors inhibiting IS development were identified by IS managers and ranked according to their perceived importance. The rankings were found to be relatively consistent with comparable factors in past empirical studies. Shahzad et al. [98], and Shahzad and Safvi [99] conducted a survey of individuals in academia, management and software development from several different countries regarding software risk factors.

While Alter and Ginzberg [4], Boehm [19,20] and Charette [25,26] indicated the type of IT system in focus, other researchers generalized risks to the overall system or software development project. There have been a limited number of studies that focus on risks in strategic IT systems. Gilbert and Vitale [43] proposed a control-intelligence risk framework in which the internal and external locus of control of an organization are mapped against its technological and business intelligence to identify the risks of failure of a strategic IT system. Clemons and Weber [29,30] commented that strategic IT initiatives carry huge risk, are difficult to carry out and the decision to develop is seldom apparent beforehand. Owing to the difficulty companies have in evaluating opportunities to invest in strategic IT systems Clemons and Weber [29,30] studied a range of analysis methods used by major corporations in the US and offered seven guiding principles for evaluating a strategic IT venture. These principles include applying analysis tools, recognizing and managing risks, and preparing for unanticipated upside and downside implications. Though the emphasis was on the guidelines, Clemons and Weber [29,30] noted that risks arising from undertaking strategic IT programmes fall into six groups: technical, project, functionality, internal political, external environmental, and systemic.

In investigating system development challenges that present risks to organizations when they use IT strategically, Kemerer and Sosa [58] reviewed business and systems journals, interviewed eight American firms, and recorded 12 system development risks in strategic IT systems. These risks were classified according to three steps of a software development process model, definition, implementation and maintenance. Sumner [108] conducted case studies of seven companies implementing enterprise-wide/ERP systems while Wright and Wright [129] conducted semi-structured interviews with 30 experienced information systems auditors. Based on both studies, Huang et al. [52] conducted a study in China to identify ERP project risk factors and then used a multi criteria decision-making methodology to analyze and prioritize the factors identified. However, there was no reference to outsourcing in all these studies even though implementation of such systems most likely involves ERP providers. More recently, Wallace et al. [119,120] conducted a survey of software project managers to explore trends in risk dimensions across low, medium and high risk projects, and examined how project scope, sourcing practices and strategic orientation of a project affect the risk. The outcome of the study suggested that strategic IT applications are likely to be more complex than non-strategic IT applications while outsourced development projects show higher levels of team risk, and planning and control risk than in-house development projects. They also found that requirements risks, planning and control risks, and organizational risks are more prominent in high risk projects, whereas complexity risk is most prominent in low and medium risk projects. Though not clearly specified, this implies that strategic IT development projects are considered as low or medium risk projects, which contradicts the perception that strategic IT development projects are high risk. However, the strategic nature of the system was not detailed and specified only as providing competitive advantage. Furthermore, the study did not make any association between strategic IT development and outsourcing. The risks and dimensions from this study were used by Han and Huang [49] in an empirical investigation of the relationships between software risks and their impact on project performance, specifically, technical performance, cost, schedule and team, in 115 software projects. There was no indication of the type of software project used in the study but the authors indicated that requirements risk is the primary factor affecting project performance. Other notable risk dimensions identified included complexity, planning and control, and the team.

2.2. Risks specific to outsourcing

There were no detailed or systematic analysis of risks in IT outsourced projects until studies by Aubert et al. [5–9] and Bahli and Rivard [11] somewhat formalized the research on risk in IT outsourcing. Due to what was perceived as incoherent views of risk in previous research, Aubert et al. [6] provided a conceptual definition of risk based on Boehm's work [19,20] that used risk definition and risk exposure. Risk factors that led to negative consequences were then identified from the literature and grouped. Aubert et al. [6] continued to discuss and examine how risk factors are linked to negative consequences. Subsequently Aubert et al. [5] presented a risk management framework that outlined four possible scenarios for negative consequences. In order to demonstrate the usefulness and adequacy of the framework, five case studies, which represented five organizations from different business sectors, were conducted [5–8]. The case studies presented differences in IT outsourcing risk management; some projects were successful and well managed while others were not. The contracts discussed varied from Y2K software development to an extensive range of IT applications such as data centre management, telecommunications and maintenance. Though the authors provide examples of lessons learned, and risks and risk management, the wide range of IT outsourcing cases and the descriptive account of the cases may make it difficult for practitioners to apply their results in a particular project. In a case study that reviewed the failed outsourced development of a billing system, which was strategic to the client, Natovich [83] highlights three risks which he categorizes as vendor risks: (1) adversarial relationships and mutual distrust between both parties as a result of a series of disputes concerning requirements' scope and definition, (2) vendor management de-escalation of commitment, attributed to expected vendor losses in a fixed price contract, and (3) the decline in strategic importance of the ventured area, and difficulty in terminating the project because of contractual obligations. Though this is the first specific account of strategic IT system development outsourcing, the focus was confined to the vendor's ability and willingness to fulfill its contractual obligations and was largely anecdotal in character.

Taylor [111–114] explored the vendor's perspective of risk associated with outsourced package implementation. While her study revealed that there are similarities between many of the risk factors identified by the vendor IT project managers and past surveys of largely in-house IT project managers, the findings also highlight risks that are specific to the vendor perspective, the package implementation approach and location of the respondents. Even though the study focused on the vendor's perspective, it also provides some insights into risks that are of interest to both the vendor and client project manager. Well-cited information system outsourcing risk research conducted by Gonzalez et al. [47] analyzed the risks that information system clients in large Spanish firms are likely to face and examined whether or not the risks change over time. IT outsourcing risks from the literature were first reviewed and used in two surveys administered to information system managers in 2001 (357 firms) and 2006 (329 firms). The study found no dramatic changes to the risks considered most important by information system managers over the five-year period.

With the growing trend to IT outsourcing in countries such as India and China, Kliem [68] based on his experience, presented several risks unique to offshore outsourcing of IT development projects. Later, Tafti [109] added a different perspective through a review of the offshore IT outsourcing literature. He identified and discussed major risk categories, and provided a framework for analyzing major risk factors when deciding to outsource IT activities offshore. Recently research attention on IT outsourcing risks has largely been on offshore IT outsourcing or IT off-shoring. Iacovou and Nakatsu [53] conducted a survey of 15 senior IT executives with offshore project experience. Building on conventional risk factors from earlier research, the survey produced a ranking of risk factors that apply to offshore-outsourced development projects. Extending past research relevant to IT off-shoring, Chatfield and Wanninayaka [27] produced a classification of IT off-shoring risks based on the sources of the risk: client, vendor and inter-firm relationships. The research was aimed at identifying the risks associated with offshore IT functions including the software development and the IT governance capabilities required to identify, manage and control those risks. The risks identified were mapped to a governance capabilities framework to show which IT capabilities are core to IT off-shoring risk management. DeHondt and Nezlek [36] also reviewed past literature to find out about IT offshore outsourcing risks but, in alignment with Kliem [68] and Iacovou and Nakatsu [53], their interest was in offshore systems development. However, this study on the impact of security, legal and general risks on the total costs of system development is preliminary and yet to be tested.

2.3. Risks in strategic IT development and outsourcing

Most research has generalized risks to the overall system or software development project. Only a few researchers have discussed risks in the context of strategic IT. Gilbert and Vitale [43] describe a control-intelligence risk framework as a guide to identify risks and plan for actions but this research lacks an in-depth analysis of the actual risks. Kemerer and Sosa [58] organized their risk discussion around a software development life-cycle model and provide examples of unsuccessful strategic IT attempts. However, their idea of strategic IT was based on a rather dated model that focused only on market and product growth. Clemons and Weber [29,30] highlight difficulties in evaluating strategic IT initiatives and the immense risks carried by such initiatives although their major concern was on producing guidelines for making decisions on investment in strategic IT. Wallace et al. [119,120] show that there are risk differences between a strategic IT application and a non-strategic IT application, and between an outsourced project and an in-house development project. However, these differences were not elaborated further, and there is no reference to a situation where a strategic IT application is outsourced. Huang et al. [52], Sumner [108] and Wright and Wright [129] investigated risks in enterprise-wide systems which are normally strategic projects depending on the organizations' objectives and how these projects are implemented. Their studies, however, do not refer to any outsourcing activities.

Research on risk is emerging in the IT outsourcing literature because of failures and dissatisfaction regarding IT outsourcing arrangements. Early work on risk is more centered on evaluating the decision on whether or not to outsource IT applications [40,55,67]. Subsequent studies made further research inroads into IT outsourcing risks and their management mechanisms [5–9,11,47,126]. Even though there has been some progress in addressing risks in IT outsourcing, the scope of IT outsourcing is rather generalized to a broad view of IT outsourcing which includes software development, software maintenance, support operations, disaster recovery and data center operations. More recently, attempts were made to address the risks in a specific IT outsourcing situation; i.e., strategic IT in Natovich [83] and package implementation in Taylor [11,112–114]. Overall, only Natovich [83] gave a specific account of strategic IT development outsourcing projects but his discussion of risks focused on a vendor's ability and willingness to fulfill their obligations and did not incorporate any analysis of strategic IT; the research was anecdotal with only a single case study. Most of the work on risk has been for IT development, mostly general software development projects, and there is little or no explicit attention paid to strategic IT. Interest in IT outsourcing risks is growing and with a growing trend to IT outsourcing to different countries, research on risk has expanded [27,36,53,68,109].

2.4. Risk framework

In prior research we used the research discussed above to develop a risk classification scheme for factors that contribute to a negative project outcome (Figs. 1 and 2).

Risks we identified were reviewed, sorted iteratively and interpreted based on:

- risks related to the client as members of the project team, as an organization, as part of management and as users of the completed system;
- risks related to the vendor as members of the project team and as an organization entrusted to provide a service; and

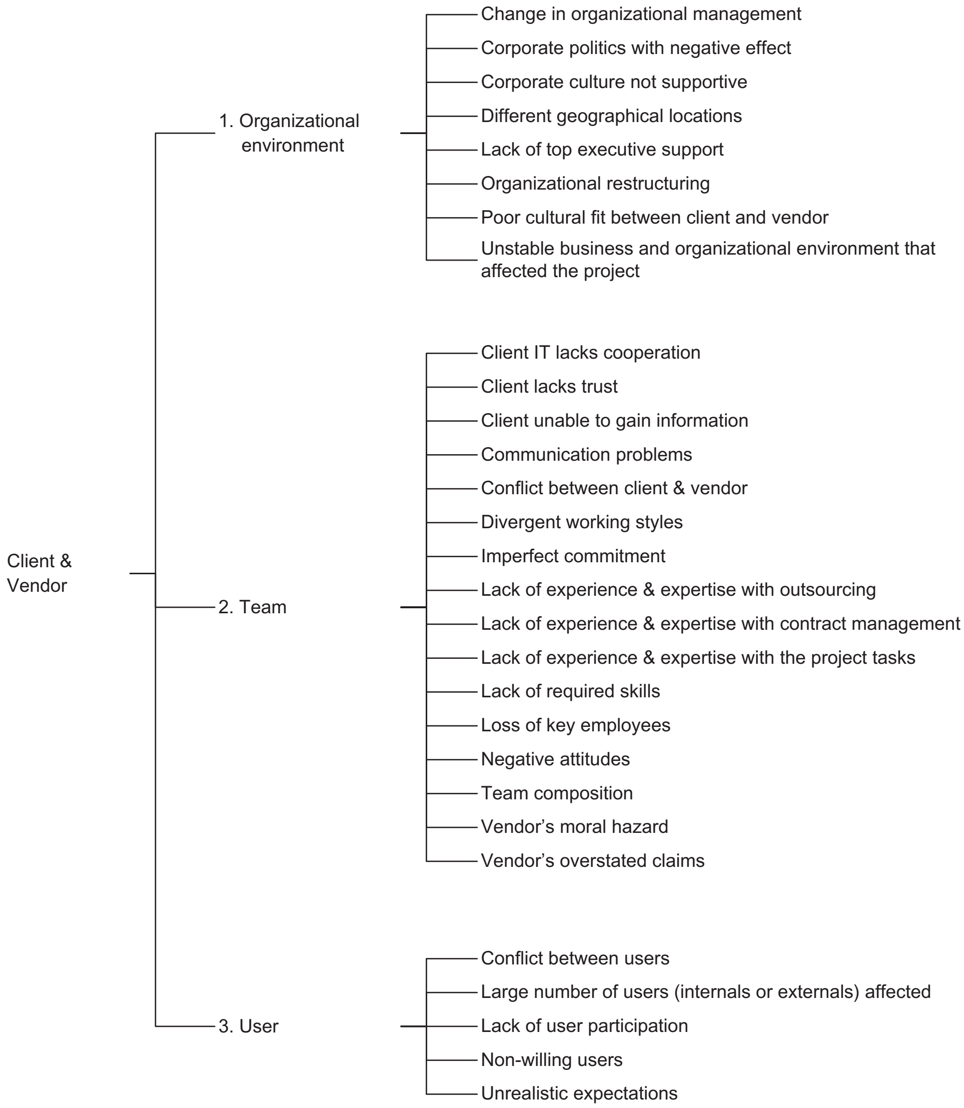
Client & Vendor
1. Organizational environment
Change in organizational management
Corporate politics with negative effect
Corporate culture not supportive
Different geographical locations
Lack of top executive support
Organizational restructuring
Poor cultural fit between client and vendor
Unstable business and organizational environment that affected the project
2. Team
Client IT lacks cooperation
Client lacks trust
Client unable to gain information
Communication problems
Conflict between client & vendor
Divergent working styles
Imperfect commitment
Lack of experience & expertise with outsourcing
Lack of experience & expertise with contract management
Lack of experience & expertise with the project tasks
Lack of required skills
Loss of key employees
Negative attitudes
Team composition
Vendor's moral hazard
Vendor's overstated claims
3. User
Conflict between users
Large number of users (internals or externals) affected
Lack of user participation
Non-willing users
Unrealistic expectations

Fig. 1. Client and vendor risk factors.

- risks relating to elements and activities of the outsourced strategic IT development project itself, from pre-contract to post-contract [1,2].

The resulting classification comprises three major sources of risks: the client organization(s), the vendor organization(s), and the outsourced strategic development project itself. Risks are further classified into ten groups. In Fig. 1, which provides an overview of risks associated with the client and vendor, risks associated with the client and vendor are grouped together, since many of the risks apply to both types of organization. However, risks that only refer to either the client, or the vendor, are clearly identified. Fig. 2 shows those risks specific to the outsourced strategic IT development project itself. The development of this framework and its partial validation are described in [1,2]. Further framework validation and improvements to the framework is the subject of ongoing research.

3. Methodology

Because little information on risk in the context of strategic IT development project outsourcing exists, we investigate, via a case study, the risks inherent in carrying out such a project from the client perspective. Understanding the client's perception of concepts such as risk factors and project outcome, either success or failure, is a challenge since individuals of the client organization may have differing views on the subject. We begin this section with a general discussion of the case study methodology and then provide a brief summary of the methodology actually used in this research. The complete methodology is described in the appendix.

3.1. Case studies

The case study methodology is useful not only for the evaluation of development methods, but also to observe, explain, and explore other phenomena within their real-life setting [132]. Thus we gain a greater understanding of why something happened as it did, and what else might be important for further investigation. Case study research involves an in-depth examination of a single case or a small number of cases. The methodology provides us with a systematic way of looking at events, collecting data, analyzing information, and reporting results. Normally one or more research questions are investigated [42,116]. While all case studies include qualitative evidence, some case studies may also present quantitative evidence [34,37,56,63,64,70,71,74,85,86,88,89,116,125,128]. Case study research is not accepted by all researchers as much of the data collected is qualitative, and may involve just a single case.

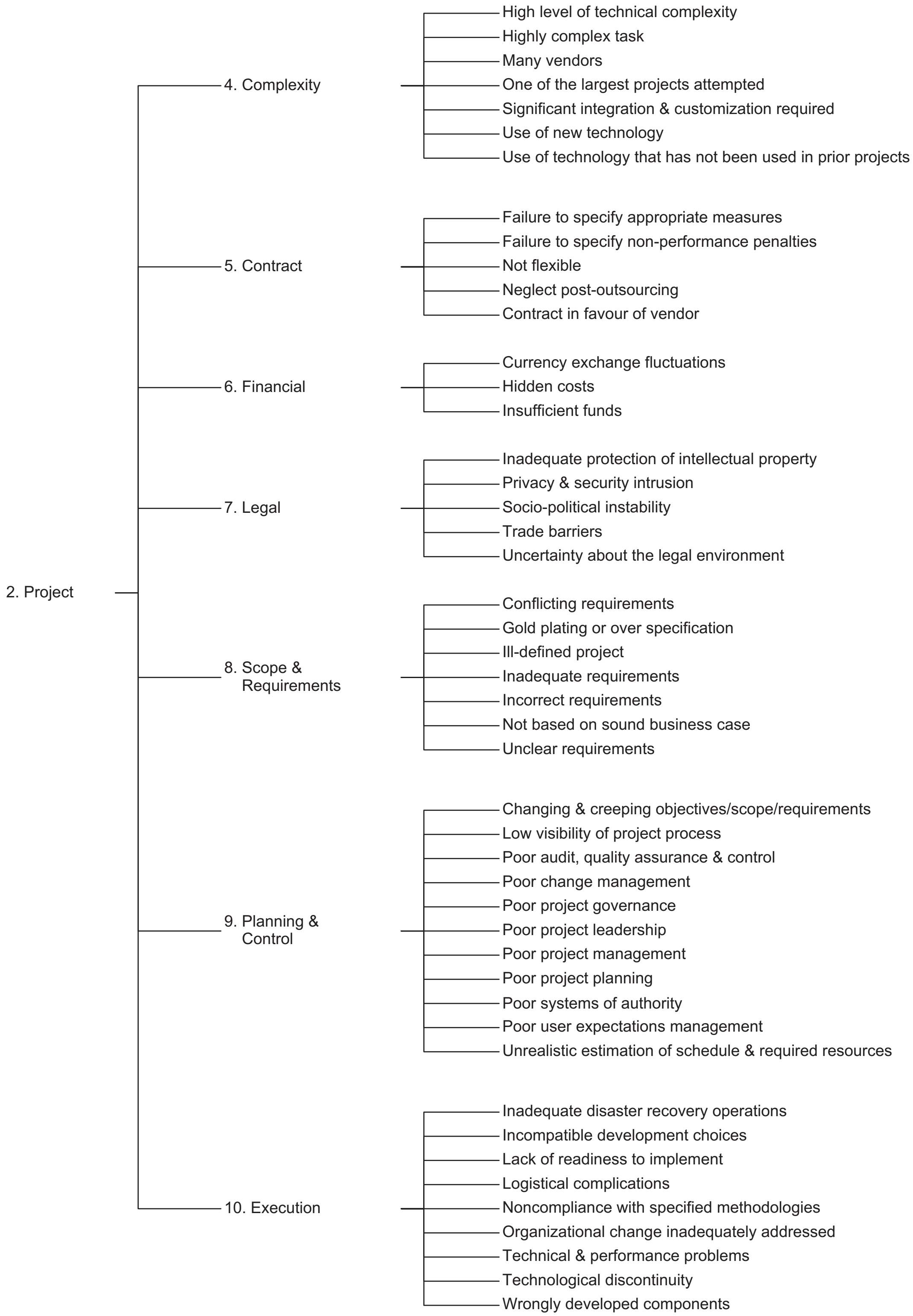
2. Project
4. Complexity
High level of technical complexity
Highly complex task
Many vendors
One of the largest projects attempted
Significant integration & customization required
Use of new technology
Use of technology that has not been used in prior projects
5. Contract
Failure to specify appropriate measures
Failure to specify non-performance penalties
Not flexible
Neglect post-outsourcing
Contract in favour of vendor
6. Financial
Currency exchange fluctuations
Hidden costs
Insufficient funds
7. Legal
Inadequate protection of intellectual property
Privacy & security intrusion
Socio-political instability
Trade barriers
Uncertainty about the legal environment
8. Scope & Requirements
Conflicting requirements
Gold plating or over specification
Ill-defined project
Inadequate requirements
Incorrect requirements
Not based on sound business case
Unclear requirements
9. Planning & Control
Changing & creeping objectives/scope/requirements
Low visibility of project process
Poor audit, quality assurance & control
Poor change management
Poor project governance
Poor project leadership
Poor project management
Poor project planning
Poor systems of authority
Poor user expectations management
Unrealistic estimation of schedule & required resources
10. Execution
Inadequate disaster recovery operations
Incompatible development choices
Lack of readiness to implement
Logistical complications
Noncompliance with specified methodologies
Organizational change inadequately addressed
Technical & performance problems
Technological discontinuity
Wrongly developed components

Fig. 2. Project risk factors.

Qualitative data can, and has been viewed by some reviewers as rather suspect and case study papers have been rejected when reviewers have objected to the subjective nature of the data. This has prompted researchers in various disciplines to conduct research geared towards advancing the case study method, for example, [13,35,51,65,92,116,131,132].

The case study is an empirical inquiry that investigates a phenomenon within its real-life context [131,132]. Case studies can be very diverse and may be classified based on their purpose as descriptive, explanatory, exploratory or evaluatory [95,121,128]. All case studies rely on multiple sources of evidence and benefit from the prior development of theoretical propositions and hypotheses [116,131,132]. In the case of exploratory research it is important to use industrially-based cases because of the importance of the context from which a theory or artifact may emerge [39,116,128]. Although case studies can be a very useful research method, the software engineering case study methodology is relatively immature, with case studies often being poorly performed. Published papers tend to omit details, such as a description of the protocol which would allow other researchers to replicate the study and most researchers, although they do cite research questions, do not include hypotheses or propositions [116]. The method is difficult to apply rigorously [21] and many problems are evident in papers reporting results. In particular, there are issues related to poor planning, design and reporting of case study research [116]. Many of the case study research guidelines are either incomplete, somewhat vague, and/or do not provide helpful concrete examples [51,131,132] although there are some recent research papers that complement each other and provide useful guidelines for using the case study methodology [51,92,116].

Owing to the exploratory nature of our research, we need to take a qualitative approach. From the client side, key project individuals, such as the project owner and project manager, are usually members of senior management or executives in the client organization. Rather than being seen as a limitation to the case study, the differing perceptions of these individuals can provide opportunities to gain familiarity with the project and new leads to follow. Qualitative research focuses on the natural settings of a phenomenon and generates detailed information since ‘richer’ answers to questions can be obtained from the subjects being studied [34,72].

Because the case study methodology we use is organized into several phases, and as most of the actions in the first three phases take place before the case(s) are actually chosen, we found that not all of the research phases were entirely appropriate for the case reported here; we have already chosen our case. In order to improve readability we include here only a very brief summary of the methodology. This section, and the appendix which describes the methodology in detail, is based on [116]. Our methodology includes six phases: preplanning, planning, design, data collection, data analysis and reporting.

3.2. Pre-planning

In the first phase we define research objectives, perform a comprehensive literature review, decide if a case study is appropriate, and set up a preliminary case study protocol template.

> **Objective 1.** *To identify the client risk factors for a strategic IT system development outsourcing project that cause a project to fail.*

Client risks in this research means risks under the control of the client, or those risks that the client has the ability to manage. In alignment with the research objective, our specific research question aims to discover critical client risks.

> **RQ.** What are the critical client risks in an outsourced strategic IT systems development outsourcing project that may cause a project to fail?

Apart from Natovich’s [83] discussion of vendor risks in an outsourced strategic IT development project setting, and some research in enterprise-wide or ERP systems, there is little recognition in the literature of the overall risks faced by clients of strategic IT system development outsourcing projects. Research in this context is sparse and therefore, it is important that research is conducted in this area.

3.3. Case study planning

This phase establishes the basic characteristics of the case study and here we need to refine the research objectives and formalize the research question. This step is followed by a definition of clear research proposition(s) and the identification of the unit(s) of analysis.

> **RQ1.** What are the critical client risks in a strategic IT system development outsourcing project that may cause a project to fail?
> **P1.** IF the client does not manage all the critical risks under their control THEN the strategic IT system development outsourcing project will fail.

3.4. Design

The design phase establishes a plan of action in order to get conclusions from the initial research question. The research methodology for this phase includes: convert propositions to hypotheses, minimize the effects of confounding factors, ensure strategies for data validity and for data collection, design the case study plan, QA on the data and the set up of a case study storage system.

Given that our case study deals with phenomena in a “real-world” setting, there may exist confounding factors that threaten the validity and results of the case study. Triangulation of different sources improves the validity of the case study processes [131]. One source can corroborate the evidence from another source, thus reducing the risk that any single interpretation of data shapes the results. This is important in a court case, and though we have no control over subjectivity in a participant’s comment or testimony, we can be sure that not only will legal counsel for both plaintiff and defendant ensure that each side is well represented but that the judge will make clear where he believes the truth lies. If either plaintiff or defends disagree with the judge’s comments then they have the right to appeal the judgment.

We will analyse media reports, reports from the court proceedings, and follow this with the judge’s findings. Media reports after the conclusion of the case study will be investigated, as well as any information pertaining to any appeal lodged. Data is to be collected by both researchers independently and then the results compared; any disagreements are to be resolved though discussion (in fact there were no disagreements between the two researchers).

Our case study database includes court transcripts, case study documentation such as newspaper articles, collected during the case study and analysis of the evidence. A spreadsheet was initially used as a repository for the case study data and as a tool to help the researcher organize and analyze the data; a spreadsheet allows a researcher to classify, sort and arrange the data.

3.5. Data collection

The three principles of data collection are: (1) use multiple sources of evidence, (2) create a case study database, and (3) validate data and maintain a chain of evidence [66,71,73,131,132]. These principles are included in our plan.

As we were not able to conduct any interviews ourselves we must rely on data from interviews conducted by reporters, evidence presented in court and responses to questions asked by counsel for both sides. We believe that case studies can be of value when there is enough secondary data from documents to answer the research question and that interviews are only one way of getting answers. Bleistein et al. used secondary data for a case study that investigated a strategic IT development project for Seven-Eleven Japan in a number of research papers, e.g., [15–18]. All the material was obtained from print sources including exercises developed for students at the Stanford Business School, e.g., [14,124], articles about the strategic development that appeared in books such as [91], and newspapers such as the Economist [115] and journal articles, e.g., [50,69,78,123].

While we have no control over the questions asked by reporters (which may well have been biased), and the number of witnesses, we trust that what is published is relatively unbiased. In addition the judge's findings are critical. We must also rely on the British justice system, and trust that the court case was conducted in an unbiased manner and that the witnesses present the views of all parties to the litigation. We are fortunate that the data we use as the basis of our research is already in a machine readable form. This certainly was a huge time saver. From the numerous documents that provide references and different perspectives on the case, we selected only those that provided a reasonable coverage. This enabled us to compare the reports in the different documents to understand the facts, and validate the risks we identified against our conceptual risk framework. Documents and articles, other than the ones already selected, that provide duplicated or similar account and discussion of the cases were excluded.

We applied a coding process suggested by qualitative researchers, for example, Dey [38], Miles and Huberman [81] and Verner et al. [116] on the published case documents. We combined the responses to specific questions posed to witnesses in the court case and documented in the judgment, with those collected from media discussion of the projects. The different sources of documented evidence were analyzed individually and used for triangulation. The only deviation from the plan occurred here because judgment took 15 months after completion of the court proceedings.

3.6. Data analysis

We found that a useful method for maintaining a chain of evidence is to highlight and number important material in interview transcriptions. The numbered material can then be copied, inserted into tables, coded and sorted. The sorting helps with cross-case analysis, should this be possible, in later research. The numbering ensures that the material can be traced directly back to its source, i.e., to the actual document, line number and page within the document. At this stage we must consider if there are threats to validity and address them in a systematic way. The resulting coding scheme corresponds to the main concepts in the research proposition. Answers to our research question were sought by exploring and interpreting the coded pieces of text, code by code, identifying patterns that match the research hypothesis.

3.7. Reporting

Runeson and Höst [92] and Verner et al. [116] provide examples of case study report structures suitable for academic reporting which we have adapted for this paper. Although a number of reporting structures have been suggested in the literature we used the type of sequence normally used for journal articles. We found it useful to start writing as soon as possible – the literature and design can be reported in the protocol. The methodology section was started during the case study planning phase. A case study should be completed to a predefined schedule and not finished purely because of time constraints. This was a consideration in our case as we were very aware that we could not complete the study without a judgment. We consistently wondered if we had missed the judgment because it took so long, and as the case would have been meaningless without it.

4. Results

We selected the BSkyB case as it is one of the few failed projects where litigation proceeded to court. As noted earlier, when projects fail, and the client sues the vendor as a result of that failure, in many cases the litigation is settled out of court and gag orders imposed on the litigants, e.g., Intrico's CONFIRM, FoxMeyer's Delta III Information System [2]. Although there may be some documentation and press coverage available for those cases we are not provided with an unbiased legal judgment.

The BSkyB project was a Call Center and Customer Relationship Management (CRM) project for British Sky Broadcasting Group PLC [22,23,31,59–61,84,87,102–105,110]. It officially started in November 2000 with Electronic Data Systems (EDS) as the system integrator but the BSkyB subsidiary, Sky Subscribers Services Ltd., took over the development work in 2002. The project took six years to complete and initially failed to deliver the intended benefits. BSkyB sued EDS (the vendor, later bought by Hewlett Packard) and the case was settled by the UK High Court of Justice on January 26, 2010, (in a 458 page written judgment by Mr. Justice Ramsey (available on-line [22]), making it one of the IT industry's longest-running and most expensive court cases. Estimated legal costs for each of the litigants were at least £45 million. The litigation lasted for 5 years, with nine months of evidence being given [22]. The case ended July 2008 though judgement was not handed down until January 2010 because of the case's complexity. Although the judge's findings in the main upheld the client's complaint, there were some areas where he found in favor of the vendor and though the vendor applied for leave to appeal they did not go ahead with the appeal and a settlement was reached in June 2010 [105]. The written judgment contains a "narrative of the project from inception to collapse, and contains a wealth of stories, insights and anecdotes into how an IT project should – and should not – be conducted. Sales teams, business managers, procurement teams, risk managers and lawyers will find the judgment – parts at least – compelling reading, with many salutary lessons".

4.1. Project

BSkyB, a wholly owned subsidiary of British Sky Broadcasting Group Plc, is the UK's largest digital pay television platform operators and a leading broadcaster of sports, movies, entertainment and news. BSkyB also offers broadband and telephone services. Unlike the normal sales, marketing and support CRM software used by the sales people to manage accounts activities, BSkyB wanted to take a more strategic view of CRM deployment. With the number of digital customers expected to rise to seven million by 2003, BSkyB announced that it wanted to install a cutting-edge CRM system that not only enabled it to respond quickly to fast changing customer demands but also to allow it to lead innovation in customer service, and to maintain its industry-leading levels of customer retention. The CRM system formed an essential part of BSkyB's interface with its customers. It was designed to facilitate, manage and record all customer-related transactions, for example, opening and closing an account, and reporting a fault. It also allowed the client to bill and process payments from their customers. Because BSkyB envisioned an advanced system that would strengthen its ability to offer more relevant, personalized services to customers and to identify new sales opportunities to put it ahead of its competitors in areas such as churn management and call resolution, we regard this project as a strategic IT development project.

Following a tender process, BSkyB in 2000 appointed the US company EDS, on a time and materials basis to design, develop, implement and manage the CRM system. The system, built around Sun Microsystems/Forte Software, hardware and middleware, and Chordiant Software's CRM software, was to be installed at BSkyB's contact centres in Livingston and Dunfermline, Scotland, where 1000 agents handled calls at any given time. An intelligent communications system provided by Lucent Technologies was included to enhance BSkyB's automated pay-per-view booking process. Lucent Technologies was responsible for the billing package, and design and installation of the advanced communications solution at the two contact centres. EDS was the system integrator for the project. Following a tender process, EDS subcontracted Arthur Andersen to be in charge of the ''Business Process'' and ''People & Change'' work-streams.

The project was estimated to take 18 months but in 2001, BSkyB and EDS revisited the terms and conditions of the contract. Renegotiation broke down and in March 2002, EDS handed over its system integrator responsibilities to a BSkyB subsidiary, Sky Subscribers Services Ltd, but stayed onto provide consultancy support for the project. According to the evidence in the eight months to March 2002, when EDS' work was terminated, it had done ''little work'' [22]. In December 2002, EDS pulled out of the contract completely and the contract ended. The contract had a baseline budget of around £48 million but when it was terminated, BSkyB had spent £170 million on software, systems integration, infrastructure costs, and the remodelling of contact centre facilities. The project ultimately took six years to complete and cost £265 million. In July 2002, BSkyB claimed £49 million compensation from EDS for lost time and benefits but the amount grew to £221 million by December 2003 and £430 million by August 2004. BSkyB began legal proceedings against EDS in August 2004 and at the commencement of the hearing, the claim stood at £709 million. The British High Court finally ruled [22] after almost 6 years that:

- EDS made fraudulent misrepresentations about its ability to deliver the project within the stipulated time;
- EDS made negligent misrepresentations prior to the renegotiation in 2001, inducing BSkyB to remain with EDS; and
- EDS breached the contract by failing to properly resource the project and failing ''to exercise reasonable skill and care or conform to good industry practice'' [22, p. 465]. However, the breaches were mainly concerned with delayed performance and were not judged serious enough to amount to repudiatory damages or a fundamental breach of contract.

BSkyB was awarded £318 million (of the £712 million they eventually asked for) in damages from EDS, now HP Enterprise Services UK.

This was a complex project in view of its large cost and scope, significant level of integration, and technical and task complexity. BSkyB employed EDS on a time-and-materials basis. BSkyB had no specific and clear project goals other than knowing it wanted a strategic CRM system. Apparently, ambiguities from the start made it difficult to get clear requirements. As details of the project began to emerge, the project scope and complexity amplified. The project requirements remained unclear and kept changing until 2003. The estimated go-live delivery within 9 months and completion within 18 months, represented by EDS were not based on any proper analysis of its capabilities to comply with the timescales.

When the contract ended, overall completion was still nowhere in sight as the project was performing poorly with delays and extra costs incurred. BSkyB alleged that EDS knew about the importance of timing to BSkyB and in a bid to secure the contract produced a plan that overstated or oversold their capabilities and unrealistically estimated the schedule and resources required. BSkyB added that EDS had been incompetent, and dishonest in its sales representation and service agreement. Of the several fraudulent misrepresentations made by EDS alleged by BSkyB, it was found that EDS had knowingly made a false representation as to the time needed to complete the project and capabilities to deliver within the timescales. EDS admitted that it was, to an extent, responsible for some difficulties experienced in the project including failure to deliver several aspects of the contract, poor specifications later in the project, poor initial user expectations management and lack of required skills in large-scale CRM integration. It was ruled that EDS' resources lacked the appropriate qualities to carry out the project, and EDS had difficulties in project management and technical leadership. EDS had not demonstrated it had the ability to fully manage the project and deliver on its promise [22].

The contract ran into difficulties in the initial twelve months and this case highlights several factors leading to an unsuccessful contract that went beyond just disagreements in renegotiating the terms and conditions. Problems identified relate to the following high level risk factors: complexity, contract, requirements, planning and control, execution, and team. These risk factors are discussed next, in Section 4.2.

4.2. Risks

As noted earlier problems relate to the following high level risk factors (which are all present in our risk framework): complexity, contract, requirements, planning and control, execution, and team. We review each of these below.

4.2.1. Complexity

The project has a total of five complexity risks: a high level of technical complexity, a highly complex task, one of the largest projects attempted by the client, significant integration and customization required, and use of technology that has not been used in prior projects. These risks, although under the control of the client, may be difficult to remove completely. Strategic systems by their nature are complex. However, if a client project manager is aware of these risks steps to minimize them can be taken right from the start of the project. These risks, as manifest in our project, are discussed in more detail below.

4.2.1.1. One of the largest projects attempted. This was a large-scale project considering its costs and with a scope that would not only integrate different data sources and create more comprehensive customer profiles, but also allow subscribers to access account, billing and other information and services via an agent, by phone, the Web or an interactive television service.

4.2.1.2. Highly complex task, with significant customization required. The new Chordiant infrastructure was to replace the company's existing Digital Customer Management System, a Field Management System that provided for all parts of Sky installations. The system also involved implementation of an intelligent telephony solution, integration with multiple legacy systems and a data warehouse. The level of integration was significant. The hearing noted the project's large size and complexity of the task; these were underestimated by EDS and this resulted in insufficient skilled resources being assigned [22].

4.2.1.3. Use of technology that has not been used in prior projects, high level of technical complexity. BSkyB wanted a leading edge solution. The Chordiant, Arbor BP (Lucent) and Forte Fusion software on which the project was based were noted by the vendors as proven leading edge technology but the vendors had not before used or integrated them together. Chordiant 5, based on open systems standards (J2EE, XML and SOAP), was the software industry's first end-to-end J2EE-standard CRM solution. Launched in January 2002, it was also reported to be the only CRM platform, at the time able to scale to millions of individual customers, and flexible enough to fit the enterprise.

4.2.2. Execution risks

We identified only one execution risk which was related to technical and performance problems; we discuss this further below.

4.2.2.1. Technical and performance problems. Chordiant 5's enterprise CRM platform, and Chordiant 5 Marketing Director, a campaign management automation tool, were implemented by 2002. However, when the contract ended later in the year, overall completion was still nowhere in sight as the project was performing poorly with delays and extra costs incurred. The delivery of the system was divided into two phases but there were serious problems throughout.

This risk is one that the client project manager needs to be aware of right from the start of the project. Questions that should be asked include, has anyone else used the technology in this form? Who are they? Can I talk to them regarding their experiences with the technology?

4.2.3. Contract

Only one contract risk was manifest in the project, this was a time and materials based contract. This type of contract can be a risk if the organization has not allocated the total amount that they are prepared to spend and are prepared to cancel the project when that amount is gone (and there is reason to believe that the software is not nearly ready to deploy). Although contract risk is identified in the risk framework developed from the literature, this particular contract risk was not present in the framework, hence it should be included in an updated framework.

4.2.3.1. Time and materials based contract. BSkyB had no specific and clear project goals other than knowing it wanted a strategic CRM system. Mike Hughes, former managing director of Sky Services noted that BSkyB had made it clear from the beginning that it wanted a flexible programme, with service providers adapting to needs along the way. BSkyB acknowledged that there was uncertainty in the amount of work and cost involved but, as claimed by EDS, was determined to arrange things in a way that it paid as little as possible. As a result, BSkyB employed EDS on a time-and-materials basis [22].

4.2.4. Scope and requirements

Three scope and requirements risks were manifest in the project. These were unclear requirements, inadequate requirements and an ill-defined project. This project illustrates how important it is for the client to ensure that clear requirements are defined. Without clear requirements it is impossible for the any project manager (client or vendor) to monitor what is actually happening and whether or not milestones are reached in a timely manner. This results in the client organization being entirely in the dark with regard to progress and at the mercy of the vendor regarding the true state of progress against requirements. These risks are described further below.

4.2.4.1. Ill-defined project. It was alleged by EDS that BSkyB had no specific and clear project goals other than knowing it wanted a strategic CRM system. The former managing director of Sky Services, noted that BSkyB had made it clear from the beginning that it wanted a flexible programme. BSkyB did acknowledge that there was uncertainty in the amount of work and cost involved [22].

4.2.4.2. Unclear requirements, inadequate requirements. Apparently, ambiguities from the start made it difficult to get clear requirements. According to EDS, BSkyB took more than five months to select the system integrator and then took another 4 months to finalize the contract while producing only a preliminary specification. EDS' Queen's Council Mark Barnes said that the "main problem with this project was that it was wholly unspecified" [31] and that Sky's requirements "kept on emerging like handkerchiefs from a magician's sleeve" [87] Barnes said that three years into the contract, during 2003, the project specifications remained so unclear that Sky had to set up a special team in order to define the exact requirements of the project [22]. "Sky pulled into this effort all sorts of business people whom the CRM team had never seen before to explain what they really needed," and the move meant "there was at last an understanding of Sky's business requirements." [22]. The hearing confirmed that many of the problems that arose during phase 2 were partly attributed to the poor state of the requirements. The court noted that the functional specifications delivered were incomplete and unsatisfactory and that the vendor had tried to re-write the specifications through Joint Application Development sessions held in an attempt to clarify the requirements. The non-functional requirements were captured late and inadequately recorded, affecting the quality of the system architecture design produced.

4.2.5. Planning and control

There were six planning and control risks manifest in the project. They are changing and creeping objectives or scope or requirements, poor audit, quality assurance and control, unrealistic estimation of schedule and required resources, vendor's poor project management and vendor's poor user expectations management. Although the client project manager needs to be convinced of the expertise and experience of the vendor project manager he cannot leave all the planning and control to the vendor. The client project manager needs to design his own plan including quality assurance, monitor progress carefully, and to question the vendor when expectations are not met. We discuss each of these risks further below.

4.2.5.1. Changing and creeping objectives/scope/requirements. As details of the project began to emerge, the project scope and complexity amplified. An analyst at Ovum Holway, Phil Codling, stated that the project was a classic case of "scope creep". The court noted that the functional specifications delivered were incomplete and unsatisfactory and that the vendor had tried to re-write the specifications in an attempt to clarify the requirements. As noted earlier it was reported that the project requirements remained unclear and kept changing until 2003 when a special team was set up to define the requirements [22].

4.2.5.2. Unrealistic estimation of schedule and required resources. The original invitation to tender made it clear that time and cost were important factors in the selection of the ultimate supplier. "EDS made representations as to the time within which the project could be accomplished, and thereby represented to BSkyB that they had undertaken a proper analysis of the tasks to be done and (BSkyB) had reasonable grounds for believing that the project could be carried out in the timescales required" [22]. This risk factor was the main issue in the dispute and closely linked to the following two risk factors. BSkyB had expected the system to be delivered as early as possible and prior to the peak Christmas customer sales period. The court found that the estimated go-live delivery, within nine months, and completion within 18 months represented by EDS were not based on any proper analysis of its capabilities to comply with the timescales [22]. EDS had made a number of incorrect claims about its software, and BSkyB alleged that these were not merely negligent, but amounted to fraudulent misrepresentation. The court agreed that one claim made by EDS – that the system would be delivered within nine months – had been made fraudulently, as the individual making it had made no attempt to check the claim and indeed had ignored concerns within EDS over whether the timescale was achievable. In his judgment Mr. Justice Ramsey said that Mr. Galloway, who led EDS's pitch for BSkyB's business, had been "cavalier" in providing estimates of timing for the project that he knew were not backed by sufficient analysis.

4.2.5.3. Vendor's poor project management, Vendor's poor user expectations management, Vendor lacks required skills. EDS denied BSkyB claims, and attributed time and cost overruns to the undefined project scope. However, it also appeared that EDS had not demonstrated that it had the ability to fully manage the project and deliver on its promise. EDS admitted that it had failed to deliver several aspects of the contract and was partly responsible for poor specifications later in the project. In addition, an EDS project manager had indicated that the project was in crisis stating "poor initial customer expectations management as to EDS' delivery capability", "lack of required and requested resources across the programme", and "lack of experienced software developers within the CRM practice who have a successful track record of large scale integration programmes" [110]. The court found that EDS' resources lacked the appropriate qualities to carry out the project, and EDS had difficulties in project management and technical leadership. In summary, the judge ruled that,

> *"EDS failed to exercise reasonable skill and care or conform to good industry practice because there was no effective programme management, the design and development of the solution was not properly documented and EDS did not provide sufficient technical or managerial resources"* [22, p. 465]

4.2.5.4. Poor audit, quality assurance and control. The judge found that the phase 1 system delivered by EDS "was not fit for purpose because it was defective in relation to DCMS System Problems and Telephony Problems and had not been properly validated and tested" [22, p. 266]]. EDS did not conduct the required testing before the phase 1 roll-out and the roll-out was suspended. Stress testing the system detected problems and these problems were resolved EDS also made little progress in achieving the milestones and deliverables outlined for phase 2 [22].

4.2.6. Team

Three vendor team risks were manifest in the project. These were vendor's overstated claims, vendor's moral hazard, and vendor lacks required skills. It is difficult for a client project manager to anticipate and deal with "vendor's moral hazard". Fraud and lies can be difficult to detect. Good client project management practices can help detect some fraudulent misrepresentations. It is imperative that the client does their homework on the vendor and the members of the vendor team, and having begun with good requirements, monitors progress carefully. Each of the risks is discussed further below.

4.2.6.1. Vendor's overstated claims, Vendor's moral hazard. According to BSkyB, EDS knew about the importance of the timing to BSkyB and in a bid to secure the contract produced a plan that overstated or oversold their capabilities and also unrealistically estimated the schedule and resources required. BSkyB claimed that it was led to believe that EDS had the resources, proven technology and methodology readily available to deliver the solution within a suitable timescale and cost. BSkyB's legal claim against EDS was filed on the basis of "deceit, negligent misrepresentation, and breach of contract" in EDS' sales presentation [103] and the service agreement BSkyB signed. BSkyB also blamed the vendor's incompetence for the failure. The contract included compensation for underperformance, but that was limited to £30 million. Alleging fraud was the only way BSkyB could claim more than the £30 million liability limit. Of the several alleged fraudulent misrepresentations made by EDS, it was ruled that EDS had knowingly made a false representation as to the time needed to complete the project and capabilities to deliver within the timescales. Fraudulent misrepresentation is hard to prove, and this case does not make it any easier. However, it highlights the risks to vendors of rash statements made by salespeople. Even a statement that is merely reckless as to the truth may constitute fraudulent misrepresentation and expose a supplier to unlimited liability.

According to a London Law Office [130] "The crux of BSkyB's case here was that EDS misrepresented, against a project that was already facing severe difficulties, that it had a plan which would allow it to finish the project on time, and that this plan had been developed following a proper analysis of the situation and tasks to be done. BSkyB alleged that EDS made this misrepresentation in order, essentially, to stay alive on the project. The misrepresentations that related to schedule were made prior to the selection of EDS to carry out the CRM project, and comprised two elements:

- that EDS had made a proper analysis of the amount of time needed to complete the initial delivery and go-live of Sky's CRM-enabled contact centre"; and
- that BSkyB "held the opinion that, and had reasonable grounds for holding the opinion that, EDS could and would deliver the project within the timescales referred to in their responses to BSkyB.

Both of these representations were found to be false, and to have been made dishonestly. The judge found that this was a representation that EDS had analysed the requirements and had a reasonable belief that the work could be achieved within the original timescales. Joe Galloway (EDS Managing Director of CRM Solutions at the time) had used his position of seniority to push through the timescales, when the evidence pointed to some dissension within EDS' more junior staff as to whether the project could be accomplished to such tight deadlines."... "The judge found that EDS did not undertake a proper exercise to work out what was required in order to deliver go-live in nine months, and the whole project in eighteen months. Likewise, they did not undertake a proper exercise of planning, sequencing and resourcing to establish that go-live could be delivered in nine months and the project as a whole in eighteen. The Court concluded that timescales were given to Sky which were thought to be those which Sky desired, knowing that there was no reasonable basis for doing so, or at the very least being reckless about the timescales, not caring whether what he said was right or wrong. He acted in this way despite the 'strong disagreement' of those working for him in his team: indeed, he caused his own colleagues 'clear distress' "[130].

The judge commented that this "was made dishonestly by Joe Galloway who knew it to be false" [22, p. 464]. The judge also commented that "In my judgment his conduct went beyond carelessness or gross carelessness and was dishonest. I consider that he acted deliberately in putting forward the timescales knowing that he had no proper basis for those timescales. At the very least he was reckless, not caring whether what he said was right or wrong." During the hearing, it was discovered that he had, "dishonestly covered up a fake degree, forged an email to cover up a mistake in key rates provided to Sky, and lied to cover up the unsatisfactory process by which the timescales in the EDS Response were assessed ..." [22, p. 178]. It became clear during his evidence that he had provided incorrect evidence as to his qualifications, claiming that he had a degree when in fact the claimed college was an online "college" offering degrees without any academic foundation whatsoever. Galloway's credibility as a witness was severely undermined by his defence of his mail-order degree from Concordia University. Although he "gave detailed evidence on how he took plane journeys between the (Virgin) Islands and attended a college there" [22], a member of the opposing legal team managed to obtain the same degree for his dog "Lulu". Without any difficulty "the dog was able to obtain a degree certificate and transcripts which were in identical form to those later produced by Joe Galloway," the judge noted, "but with marks which, in fact, were better than those given to him." [33]. The blow to the witness' credibility was reported as pivotal to the success of the case. The emphasis in the judgment is on the dishonesty of one man, since dismissed from EDS, and there were no findings of systemic or widespread internal failures or recklessness in relation to EDS's wider sales processes [33].

4.2.6.2. Vendor lacks required skills. Lawyers for the plaintiff told the judge that if the contractor had been honest about its abilities, BSkyB would have chosen a different company. An EDS project manager had indicated that the project was in crisis stating "poor initial customer expectations management as to EDS' delivery capability", "lack of required and requested resources across the programme", and "lack of experienced software developers within the CRM practice who have a successful track record of large scale integration programmes" [110]. The judge found that EDS' resources lacked the appropriate qualities to carry out the project [22].

5. Conclusion, discussion and further work

The BSkyB project failed because of problems that can be traced to both client and vendor; the major fault was with the vendor [22]. As noted earlier, EDS, had made fraudulent misrepresentations about its ability to deliver the project within the stipulated time, made negligent misrepresentations prior to the renegotiation in 2001 inducing BSkyB to remain with EDS; and breached the contract by failing to properly resource the project, and by failing "to exercise reasonable skill and care or conform to good industry practice" [22].

It is interesting to note that much of the discussion in the media about the fall-out from this case, and the settlement amount won by BSkyB, is focused on how the case's judgment will likely affect software suppliers and how they will approach their bids in the future [62,130]. For example, "the Chairman of the National Outsourcing Association said the case set an important precedent for outsourcing, and would lead to changes in the way companies go about tendering, due diligence, selection and contracting. Yesterday's settlement will not, though, be the end of the matter for the IT industry, and the BSkyB/EDS case could well push up the cost of IT outsourcing and services contracts" [133]. While fraudulent misrepresentations are rare in the IT world companies bidding for contracts will need to ensure that they can back up any statements they make about project costs and delivery. This may mean more compliance checks within companies bidding for contracts and it might also force bidders to spend more time scoping and planning projects, to ensure projections are realistic. This may lead to IT projects being better planned and thought out, and a translation into on time and on budget delivery. Suppliers need to review who within their organization has responsibility for the tendering process and what checks and balances exist to ensure that one person cannot override the views of the team as a whole; more caution is going to be required in ensuring that unverified or unverifiable claims are not made as part of the tendering process [133].

We did find however, a few comments about how this case could affect client behavior. For example, "From the customer's perspective, it is no good simply accepting at face value what the supplier says: some more cooperative way of working on tenders may be called for by this judgment, where the customer is allowed to verify for itself what claims the supplier is making in a tender", "why was the customer so willing to believe what it was being told?", and "customers should be more assiduous in positively verifying what is being said. If you are selecting people on the basis of the representations they are making, you need to be sure that those representations are verifiable and verified and then written down".

While it is easy to be wise after the event, what could the client have done if the major risks had been recognized early? The project was large, and highly complex with a significant level of integration and customization required. Complexity is a common problem with strategic IT projects whether they are outsourced or not. While the client could not easily alter the project's complexity characteristics, the client project manager needed to be aware of the potential risks complexity causes so that some mitigating action could be taken. There were significant problems with requirements and project scope. This is a very common problem with failed software development projects. Unclear requirements, inadequate requirements, and an ill-defined project provide significant risks. It is impossible to estimate cost and schedule for a project if you do not know what it is supposed to do. You also cannot properly manage such a project. The client project manager should have been aware of this. If you don't know what the requirements are, it is a good idea to set up a smaller project that defines the requirements before the main project starts. If the client really does not know what they want, it is naïve to expect the vendor to collect the requirements without some kind of formal agreement regarding requirements. A senior member from a large Japanese software development (vendor) company recently commented (personal communication). "We don't get paid for the requirements". He obviously perceived that requirements elicitation can be a significant overhead for the vendor.

We perceive that most of the problems with planning and control were a consequence of the project's poor requirements, and, although lack of vendor expertise and failing to properly resource the project are correctly blamed on the vendor by the court, this is a risk that an experienced and savvy client project manager should quickly recognize should it occur.

Under some circumstances time and materials projects can be problematic. While the vendor will be likely to try to maximize time, for a maximum profit, the client will try to minimize time. This can put the two sides in conflict. Time and materials contracts can work, but the client must know what their limits are, and be prepared to walk away once that limit has been reached. Because the contract risk was a risk that we had not encountered before, we have accordingly included it in an updated risk framework.

The team risks of vendor's overstated claims, vendor's moral hazard, and vendor lacks required skills, together with fraud and lies can be difficult to detect. Had the requirements been better, the client project manager may have had some warning that team risks were rendering the project problematic. In outsourcing software development the client should remember the saying, (which has been variously attributed to both the Chinese strategist, Sun-tzu (about 400BC), and Machiavelli in *The Prince*) [125], "keep your friends (in-house IT) close and your enemies (outsourcing vendor) closer"; that is, if you get to know your vendor, then you will know what he is up to, and you can keep a close eye on him. This is easier than if you don't know him at all, as that can make him unpredictable. In outsourcing a difficult development project a client project manager cannot just hand the project over to the vendor and then take a hands-off approach. Therein lies the seeds of failure.

The case study guidelines that we followed were in some sense overkill for a case study which used only third party data. A number of actions were found to be unnecessary, for example, we did not need to spend time finding appropriate people to contact within likely organizations, sell our research to that organization, get ethical approval, sign confidentiality agreements, find an appropriate project, do interviews and transcribe them, provide copies of the transcribed data to the interviewees for accuracy checking, get approval to publish, etc. We found the data collection part of the study edifying and at times entertaining, although waiting 15 months for the court judgment and wondering if somehow we had missed it, was stressful. We believe that using third party data as the basis of a case study is useful as long as enough data is available, and that data is reliable. Investigating failed projects via their court documents can be very rewarding and time saving in-so-far as the insights provided. We would not suggest that such studies should take the place of the traditional case study but they can complement traditional research.

Further work includes investigating other failed projects that have resulted in litigation, via media reports and court documents. Such projects are uncommon because of the legal costs involved. We are however, aware of one such case in Australia, where the client was awarded damages, and one which is scheduled to go to court in the UK, if the parties do not settle beforehand. It would be interesting to discover what similarities there are in the cases that do go to court, when compared with those that are settled out of court. Airing the court judgments should help both clients and vendors to do better in the future.

Appendix A

A.1. Case study methodology

Because the case study methodology we use is organized into several phases, and as most of the actions in phases 1–3 are meant to take place before one or more cases are actually chosen, we found that not all of the research phases were entirely appropriate given that we had already chosen our case. In order to improve the readability of the paper, we have included in the main text only a summary of the methodology and incorporated only those sections appropriate to the research. The complete methodology is described in this appendix where we have used the same numbering as in the main body of the paper. Our methodology includes preplanning, planning, design, data collection, data analysis and reporting; the whole of this section is based on [116].

A.1.1. Pre-planning

This first phase ensures that at the outset of a project sufficient preparation and training is undertaken by the case study investigators. Its four steps are: define research objectives, perform a comprehensive literature review, decide if a case study is appropriate, and set up a preliminary case study protocol template. The preplanning phase ensures that a researcher makes adequate preparation before undertaking the research.

Define the study objectives. In this preparatory step, broad and overarching objectives for the research project are defined. In general, they will be a set of high-level objectives for the study based on the identified area of interest [41].

Objective 1. *To identify the client risk factors for a strategic IT system development outsourcing project that cause a project to fail.*

Client risks in this research means risks under the control of the client, or those risks that the client has the ability to manage. In alignment with the first research objective, the specific research question below aims to discover critical client risks.

RQ. What are the critical client risks in an outsourced strategic IT systems development outsourcing project that may cause a project to fail?

Undertake a comprehensive literature review. A comprehensive literature review and analysis with sufficient breadth and depth is used to form a solid foundation for the research. Only once an extensive prior art study has been completed can a researcher successfully identify where additional contributions are possible. Documenting the motivation for further investigation allows the researcher to narrow the scope of the planned research, and helps to ensure that the study will be possible given time and resource constraints.

In the background section, Section 2, a series of risks connected with IT development, IT outsourcing and strategic IT projects from case studies, surveys, focus groups and accounts of developer experiences were discussed. Sources include both technical and nontechnical books, academic and industrial journals, and conference proceedings. Articles that appeared in publications such as the Sloan Management Review, Industrial Management & Data Systems, Journal of Management Information Systems, IEEE Software, Communications of the Association for Computing Machinery (ACM), Journal of Information Technology and the International Conference on System Sciences Proceedings have provided the most useful material for the literature review.

The major authors in the field whose work has been frequently cited include Barry W. Boehm, Henry Barki, Robert Charette, Kalle Lyytinen, Mark Keil, Roy Schmidt, Chris F. Kemerer, Linda Wallace, Leslie P. Willcocks, Mary C. Lacity, Benoit A. Aubert, Mohammed H.A. Tafti and Hazel Taylor. Strategic IT system development outsourcing is now being pursued actively by vendors and, because of the importance of such projects to the client organization, unaddressed risks that develop into problems can have seriously detrimental effects on the organization. Apart from Natovich's [83] discussion of vendor risks in an outsourced strategic IT development project setting, and some research in enterprise-wide or ERP systems, there is little recognition in the literature of the overall risks faced by clients of strategic IT system development outsourcing projects. Research in this context is almost non-existent and therefore, it is important that research is conducted in this area.

Set up the protocol template. For case study research, a case study protocol needs to be established first. The protocol serves as a guide to the researcher in carrying out data collection and analysis for the cases. It outlines procedures and rules to be followed, and the instruments to be used for collecting data [132]. Preparation of a protocol enables the researcher to have a good grasp of the focus of the case study and to foresee problems that might arise during its conduct. As such, the use of a protocol increases the reliability and quality of case study research. A well defined protocol also enables other researchers to exactly replicate the research. A case study protocol is a living document that is updated as the research progresses and is only complete when the research is complete. Details of this methodology section are, essentially, summarized in the case study protocol. The case study protocol developed for this research integrates the main areas proposed for a case study protocol in Brereton et al. [21] and Maimbo and Pervan [77]. Although we have not provided the case study protocol as a separate document here, all the material contained in the protocol is included in this appendix. The protocol developed for this research includes material included in the (1) background, (2) case study procedures, (3) data analysis guidelines, (4) plan validity, and (5) reporting sections.

Case study administration. Overall, administration plays an important role in ensuring that issues related to (a) legal, (b) ethical, (c) intellectual property (including publishing), (d) scheduling, and (e) working conditions, are met. There are five main activities or steps in this phase, but administration is an 'ad hoc' sub-process, in the sense that its activities can be done in any order, with any frequency, and in parallel with almost any other phase.

We had no need for legal agreements, ethical clearance or issues regarding the confidentiality of data for this case as the data collected is all in the public domain, in newspapers, trade journals, or in British High Court documents (available on line [22]) where testimony was obtained from witnesses who were questioned in court and were under oath to speak the truth; no other form of intellectual property protection was required for this research thus, it is not likely that our investigation will have risk of (further) harm to the research subjects.

A.1.2. Case study planning

In this section we describe in detail the steps that comprise the phase that determines basic characteristics of the case study. The order of the steps need not be sequential in all circumstances. This phase establishes the basic characteristics of the case study.

Refine research objectives and formalize research question. The focus of the study can be established by refining research objectives into questions about the situation or problem to be studied and the purpose of the study. The research questions must be precise and unambiguous, not have been addressed by previous literature, achievable with the case study approach and not better suited for another scientific approach.

Case study research generally answers one or more questions which begin with "how" or "why." The questions are targeted at considering a limited number of events or conditions and their inter-relationships. To assist in targeting and formulating the questions, the literature review conducted in the research pre-planning phase established characteristics of past research, which leads to refined, insightful problem questions. A careful definition of questions pinpoints where to look for evidence and helps determine the analysis methods to be used in the study. It is recommended that the researcher to explicitly state the research questions in this step.

Our central or overarching research question is:

> *What are the critical client risks in an outsourced strategic IT systems development outsourcing project that may cause a project to fail?*

The object of our research is an outsourced strategic IT system development project. The literature review we conducted allowed us to narrow down our focus. Subsequent sub-sections outline the research plan, which involves translating the question into a research proposition, outlining appropriate concepts, and measures and how they will be analyzed.

Define clear research proposition(s) and identify unit of analysis. Propositions are predictions about the world that may be deduced logically from theory [100]. Dubin [39] remarks that the usual form of a proposition takes an "IF ... THEN ..." format, but not all researchers agree on this (as this format may not always be the clearest; furthermore it implies causal relationships, while in some cases only correlation of facts can be determined). According to Yin [131,132], a proposition "directs attention to something that should be examined within the scope of the study. Research propositions should be placed together with their corresponding research question to make it clear how they have been derived.

Propositions assist the researcher in identifying relevant evidence in the case(s) studied. The data collection and analysis can then be structured so that evidence can be collected, in order to corroborate or refute the propositions. As clearly stated earlier, this constitutes a situation where the organization, as the client, contracts out to a vendor(s) the development of either all or significant part(s) of a strategic IT system. The research question identified earlier is refined into the following, which further clarify what we will investigate.

> **RQ1.** What are the critical client risks in a strategic IT system development outsourcing project that may cause a project to fail?
>
> **P1.** IF the client does not manage all the critical risks under their control THEN the strategic IT system development outsourcing project will fail.

At this stage the researcher should decide what a case will consist of; that is, the researcher should decide whether a case is a project, a department, a set of different projects, an organization, etc. For the research conducted here our unit of analysis is obviously a project.

Define conceptual framework. In Step 2 of the research pre-planning phase, a comprehensive literature review was performed in order to understand the key points and factors that can be used as a basis for data collection. These key points become the foundation of the conceptual framework that will evolve over time as the research is carried out and the findings analysed. Figs. 1 and 2 present a conceptual framework developed from the literature. A conceptual framework is a research tool useful for understanding and summarizing the findings from the literature and to communicate these to interested parties. It also provides us with a basis to begin structuring our data collection and analysis process. It should be noted that the classifications and risk factors articulated in the framework may evolve over time as the research advances, the framework is tested, and findings analyzed. The framework creates opportunities to gain further insight into the research and evidence to support our proposition.

Define concepts and measures. A concept is an idea being investigated; e.g., from our example, "client risk", is a concept that should be defined, whereas a measure actually quantifies in some way the "client risk". Some concepts such as "project failure" obviously need to be refined to much greater detail in order to be measured. At this stage, the researcher should ensure that every defined measure is: (a) precise and unambiguous; (b) relevant for the concept(s) it is supposed to measure; (c) measurable within the context of the case study; (d) with a precisely determined measurement unit (in the case of several possible measurement units, it is advisable to determine one of them to be used throughout the case study); (e) illustrated with examples for every concept (this improves understandability). The concepts found in the research propositions should be defined and added to a research glossary; whereas the measures should be defined and made explicit in a table included in the main body of the research methodology section. The researcher should make explicit the concepts and measures to be analysed or evaluated. It may happen that although a concept is interesting and should be defined, it may not be possible to measure and analyse it explicitly. With this in mind, the researcher should systematically go through each defined concept and measure and state explicitly whether or not it will be analysed.

Our proposition highlights several key concepts that need further elaboration and refinement since they are fairly abstract at this stage. We need to move from the abstract to the empirical and measurable. Without establishing an explicit operational set of measures and/or descriptions for the concepts being studied, there is a risk that the design of the research and collection of data will be overly subjective and thus, findings will be inconclusive or unfounded. Additionally, a researcher will have difficulty in linking the data collection questions to the propositions and research questions. Establishing correct operational measures for the concepts helps to establish construct validity.

We define a concept as a generalized idea to be examined, derived from the research propositions, whereas a descriptor is a term that characterizes the concept. A measure allows us to quantify the concept so that it can be evaluated appropriately. However, as some concepts may not be directly measurable only an interpretation or representation of those concepts can be measured [116]. The measures and/or descriptors for the concepts are presented in Table 1.

Define how you will analyse results. It is important that the researcher has a plan for analyzing the defined concepts and measures. A tabular format may be useful to represent each concept

Table 1
Concepts and measures.

1. Client	Project stakeholders		Coding of client's business, organization, respondent, experience and expertise in development outsourcing projects
2. Strategic IT system development outsourcing project	Goals		Coding of project background, team (client and vendor), cost, duration, and contract and post-contract activities
	Scope	Large/average/small	
	Team size	Number of team members	
	Budgeted and actual cost	In £	
	Scheduled and actual duration	In years and months	
3. Risk under the control of the client	Known risk factor that client has the ability to manage	Yes/No	Thematic coding of risk factors following the pre-set risk factors and categories of the conceptual risk framework, and emergent risk factors and categories
	Manifest risk factor is a risk factor that actually occurred that client has the ability to manage	Yes/No	
4. Aware	Client identifies the risk factor under their control	Yes/No	Determination of client's known and residual risk factors
5. Project outcome	Project delivery of the intended benefits to client	Successful/partially successful/failure	Coding of client's perception of level of project success
6. Critical risk	Impact of risk under the control of the client on project outcome	High/medium/low	Pattern identification between risk under the control of the client and project outcome
7. Complexity	Existence of complexity risk factor (examples, large project scale, complex task and complex technology used)	Yes/No	Frequency count of critical client risk factors pertaining to complexity in project
8. Team	Existence of client and vendor people risk factor (examples, conflict, lack of required skills and loss of key employees)	Yes/No	Frequency count of critical client risk factors pertaining to team in all considered projects

Note: We have defined the concepts and measures so that we can conduct multiple case studies although only one such study is described here.

and measure, and the planned analysis for each. At this stage, it is sufficient to state, at a relatively high level, how the concepts and measures will be analysed, but it is important that the researcher plans for specific and sufficient data for an analysis. For example, a number of the "concepts" identified, defined and selected, may be analysed using qualitative data analysis techniques, such as thematic coding and rich interpretation; whereas the researcher may plan to analyse the "measures" using simple quantitative techniques, Likert scales, or more sophisticated statistical methods.

For each planned analysis the researcher may provide a description of how a particular analysis will be undertaken, e.g., using a comparative analysis of case study participants responses to a particular question or set of questions. For the case study described here we intend to combine the responses to specific questions posed to witnesses in the court case and documented in the judgment, with those collected from media discussion of the project. Data collected in this case study research will be qualitative with data elicited from a number of sources including the media, web pages and the British High Court documents, will be assessed. The different sources of evidence will be analyzed individually and used for triangulation. Triangulation combines different data sources with the aim of corroborating the findings or conclusions [132]. Findings are more persuasive and accurate if based on several sources. For the purposes of triangulation we have evidence from the client organization, vendor organization, and comments from the judge.

Miles and Huberman [81] and other qualitative researchers in software engineering, such as Runeson and Höst [92], Seaman [97], and Verner et al. [116], suggest a coding process to reduce the unstructured body of data into a manageable, meaningful format for data analysis. The raw data from relevant documents will be inspected for categories derived from the concepts in the research propositions and any emerging concepts from the research. These data segments will then be grouped under the categories, referred to as codes (in a spreadsheet). Codes are tags or labels used to assign meaning to segments of textual data compiled during the study [81]. A code can represent a word, a phrase, a sentence or a paragraph identified as important in the research. Aside from qualitative analysis, quantitative interpretation or simple count may also be performed on the coded data.

Identify and justify sites and/or individuals that are the focus. We define the individuals of interest as those who participated in the BSkyB court case including members of the vendor organization, those from the client organization, legal counsel and the judge(s) as well as any other individuals quoted in the media.

Define boundary of case study. Describe the scope of the study, and what criteria are used to determine its scope:

> *"Data will be gathered from the media, such as newspapers and trade publication as well as from court proceedings of the BSkyB case. Comments made by the witnesses and legal counsel during the case and by the judge in his summing up and in presenting judgement are also relevant".*

We consider that, because all our witnesses are under oath, their responses will be truthful in so far as their knowledge of any particular factor being investigated. We also consider that the newspapers and trade journals will have reported honestly as they would not wish to be involved in defamation litigation.

Select case(s) to study. The BSkyB case is precise and unambiguous; much information is available, it corresponds to the defined research objectives and is observable and measurable within an appropriate time frame. Both client and vendor witnesses were called during the court case and if possible one side will attempt to refute arguments, evidence and facts from the other side.

Select pilot case(s). Although we only intend to conduct a single case study for this research to all intents and purposes the case described here could be considered to be a pilot case when we examine further failed strategic outsourced IT projects.

Decide if you will get appropriate level of confidence. We believe that we will have a high level of confidence from this particular

Table 2
Example proposition and hypothesis.

RQ1.	What are the critical client risks in a strategic IT system development outsourcing project that may cause a project to fail?		
P1.	IF the client does not manage all the critical risks under their control THEN the strategic IT system development outsourcing project will fail	**H1.**	The risk framework includes all the critical risks of the BSkyB project under the control of the client"

case as the data collected is from witnesses are under oath, counsel, who will ask the kinds of questions that we may have asked, and the judge who will provide an unbiased summation of the evidence.

A.2.1. Design

Earlier phases present what we will investigate in our research and the case we have selected to study. The design phase establishes our plan of action in order to get to conclusions from the initial research question. This is important and helps to ensure that the evidence gathered does address the actual research questions. The following sub-sections describe the research methodology for this phase: convert propositions to hypotheses, minimize the effects of confounding factors, ensure strategies for data validity and for data collection, design the case study plan, QA on the data and the case study storage system.

Convert propositions to hypotheses. A hypothesis is an empirically testable statement that is generated from propositions defined in the planning phase. See example in Table 2. According to Shanks and Parr [34], values of concepts that appear in propositions should be defined before empirical testing is conducted. In addition, it is possible that one proposition may generate many hypotheses. A very useful outcome of hypotheses generation is that the researcher refines the definitions of the concepts and measures more precisely so they can be evaluated.

First, the researcher must identify empirical indicators for the terms in the propositions. This is called operationizing a concept, as it makes the concept ready to be used. For example, if the researcher considers positive impact, then there is a need to define what this is and how it will be measured.

The propositions defined in the case study planning phase earlier are converted to hypotheses, which are empirically testable statements to support the collection and analysis of the data [116]. Generating hypotheses involves substituting the concepts that appear in the propositions with empirical indicators [100]. The case studies then provide the opportunity to confirm or refute the hypotheses. Hence, the hypotheses must be stated in a form suitable for confirmation or refutation. The following summarizes the research questions, propositions and hypotheses of this research.

Minimize the effect of confounding factors. Confounding factors are risk factors that may affect the results of a study. Given that our case study deals with phenomena in a "real-world" setting, there may exist confounding factors that threaten the validity and results of the case study. The researcher does not have the same control over a case study as in an experiment [127] and understanding the existence of confounding factors is important in increasing confidence in inferences drawn from the research findings. Confounding factors can never be entirely known in case study research but the most significant confounding factor which can be identified in this research is subjectivity in a participant's testimonial. This could be due to a number of factors such as difficulty in accurately remembering details about the case and the events that occurred, reservations in disclosing information, and the researcher's own limitations and lack of skills in collecting the data. It may be impossible to eliminate this factor completely, but it can be controlled and the effects minimized through the use of multiple sources of evidence to obtain multiple perspectives. Triangulation of different sources improves the validity of the case study processes [131]. One source can corroborate the evidence from another source, thus reducing the risk that any single interpretation of data shapes the results. It is not our intention to focus on causal relationships. Yet, if under these varied circumstances, we can still make some inferences from the case, we will have considerably increased the validity of our findings. This is important in a court case, and though we have no control over subjectivity in a participant's comment or testimony, we can be sure that not only will legal counsel for both plaintiff and defendant ensure that each side is well represented but that the judge will make clear where he believes the truth lies. If either plaintiff or defends disagree with the judge's comments then they have the right to appeal the judgment.

Ensure strategy for data validity. One main concern for case study research is in ensuring quality in terms of validity – the extent to which the research accurately assesses the stated propositions – and reliability – the extent to which repeating the same research under the same circumstances would produce the same results. Establishing validity and reliability in case study research provides confidence in the evidence collected and results drawn from the research. Strategies to address validity and reliability must be incorporated from the start [92,127]. The validity and reliability design tests proposed by Yin [131] which are similar to the tests suggested by Wohlin et al. [127] in controlled software engineering experiments are adopted for this research. At this stage, it is possible to describe an initial construct validity whereby the "constructs" are developed by the researcher using multiple sources of evidence, such as literature and expert opinion. It is not until later on that the actual constructs are tested. Design tests commonly applied to judge a well constructed case study design to improve the research quality are [131,132]:

1. Construct validity: establishing correct operational measures for the concepts being studied;
2. Internal validity: establishing a causal relationship, whereby certain conditions are shown to lead to other conditions, as distinguished from spurious relationships;
3. External validity: establishing the domain to which the study's findings can be generalized; and
4. Reliability: demonstrating that the operations of a study – such as the data collection procedures – can be repeated, with the same results."

Several tactics have been proposed by Yin [132] for dealing with the four design tests. Table 4 summarizes how our research satisfies these four case study research design tests.

Define the data collection strategy and process. The researcher needs to consider: what are the sources of data and what is the order in which you can get it? The researcher must also at this stage define how the chain of evidence will be preserved. Furthermore, the process of collecting the data must be planned for, including the order in which the data will be obtained. We will analyse media reports, reports from the court proceedings, and follow this with the judge's findings. Media reports after the conclusion of the case study will be investigated, as well as any information pertaining to any appeal lodged.

Data was collected by both researchers independently and the results were compared.

Design the case study plan step by step. Before commencing the case study it is important to plan and document the expected sequence of events. A logical plan should set out how the initial questions are to be answered and conclusions derived. The plan includes major phases such as the collection and analysis of relevant data and is one means to help avoid the situation where the evidence collected does not address the initial research questions. Furthermore, the researcher should also try to anticipate where deviations from the case study plan might occur and what should be done to address such changes. The use of multiple data sources refers to triangulation of evidence, and addresses potential problems concerning construct validity; multiple sources of evidence provide multiple measures of the same fact or finding, and increase the reliability of the data collection process. Another important data collection strategy, which increases the reliability of information presented in the study, is to show how the chain of evidence is maintained in the research. With the amount of raw data collected, a researcher needs to convincingly show how conclusions are derived and connect to the research questions. Akin to an audit trail, the chain of evidence allows the reader to trace the steps from the evidence presented to the conclusions drawn.

In accordance with Yin [132], the case study database will include references to actual categories derived from the conceptual framework and the data will be sorted using these categories or themes. The categories in the conceptual framework are directly linked to the research question.

Define QA on conduct of the plan. The purpose of the QA definition is so that at the end of the study, an evaluation report can be written up including recommendations for changes in procedures. The original plan should be systematically compared with progress and results at each stage of the study.

Design a case study storage system. A case study storage system is needed to maintain the data collected in a manner such that the data can easily be retrieved and reviewed not only by the primary researcher but also other researchers, thus increasing the reliability of the research [131,132].

Our database includes court transcripts, case study documentation such as newspaper articles, collected during the case study and analysis of the evidence. A spreadsheet was used as a repository for the case study data and as a tool to help the researcher organize and analyze the data. The spreadsheet allows a researcher to classify, sort and arrange the data.

Produce the first draft of the plan. The draft of the plan should contain precise and unambiguous descriptions and formal documentation of: (a) the chosen protocol; (b) the chosen data collection strategy and process; (c) the designed case study storage system; (d) QA procedures on conduct of the plan; and (e) other steps in the case study (to the extent known at this time).

The case study plan was prepared to ensure the smooth conduct of the case study. This is a logical plan that sets out how the case study will be conducted in order to draw conclusions and find answers to the research question. The plan is used in conjunction with the case study protocol.

There were no deviations from the baseline case study plan in terms of the timing and completion of the tasks. The only delay we encountered was that we had to wait for 15 months for the judgment and another few months while Hewlett Packard decided whether or not to appeal the decision. They eventually decided not to appeal the findings and the amount of damages they were ordered to pay to BSkyB [105].

A.2.2. Data collection

The boundary between data collection, data processing, data analysis, data interpretation, data presentation, and reporting can at times become somewhat blurred. Data triangulation, which must never be lost sight of during the data collection stage, crosscuts data processing and data interpretation. The three principles of data collection are: (1) use multiple sources of evidence, (2) create a case study database, and (3) validate data and maintain a chain of evidence [66,71,73,131,132]. These principles are included in our plan.

Obtain the data from multiple sources. Interviews provide an essential source of evidence for most case studies. The advantages of using interviews as a form of data collection are that they focus directly on the case study topic and can provide perceived casual inferences. The outputs of the interviewing process are interview transcriptions, and interview notes which can be very time consuming to develop. While we were not able to conduct any interviews ourselves we rely on data from interviews conducted by reporters, evidence presented in court and responses to questions asked by counsel for both sides. In addition the judge's findings are critical. We are fortunate that the data we use as the basis of our research is already in a machine readable form. This certainly was a huge time saver.

Interviews. Key weaknesses that may arise from using interviews are: bias due to poorly constructed questions, response bias, inaccuracies due to poor recall and reflexivity – where the respondent says what the interviewer expects or wants to hear [132]. Yin [132] recommends that during interviews questions are asked in an unbiased manner. One suggestion to avoid bias in the results is to make sure that "interviews" are conducted with an appropriate number of interviewees and ensuring other sources of evidence are used. While we have no control over the questions asked by reporters (which may well have been biased), and the number of witnesses, we trust that what is published is relatively unbiased. We must also rely on the British justice system, and trust that the court case was conducted in an unbiased manner and that the number of witnesses presents the views of all parties to the litigation.

Documentation. Documentary information is an important source of evidence in a case study and is particularly important here. Supplementary documents play an explicit role in data collection throughout our study. It may however, be difficult to determine the reporting bias of the author who compiled the document. A documentation report should be produced by the case study researcher describing and classifying the different types of documents to be analysed and how they may link to other sources of evidence. We believe that case studies can be of value when there is enough secondary data from documents to answer the research question and that interviews are only one way of getting answers.

Store the data into the database. Input to this step is the raw material, obtained from documents collected during data collection, and the output is the case study database which will hold all facts and evidence. The data consists of coded data; a coding scheme, and data displays.

Validate the data. The data collected must be validated and a data validation report prepared. This report discusses the actual data collection and any deviations from the data collection plan. The researcher must check to see if any data is missing or invalid and must ensure that the collected data is sufficient to answer the research question and proposition of the case study. Our only deviation from the plan occurred here because judgment took 15 months after completion of the court proceedings.

Documents collected are the basis of our case study storage repository, and are kept as both softcopy and hardcopy and identified by source. From numerous documents that provide references and different perspectives on the case, we selected only those that provide a reasonable coverage of the case. This enabled us to compare the reports in the different documents to understand the facts, and validate the risks identified against our conceptual risk

Table 3
Strategies for case study design tests.

Test	Tactic [132]	Strategy
Construct validity	Use multiple sources of evidence	Review evidence from all witnesses via court transcripts, counsel comments and questions as well as comments and findings from the judge and information in the press and trade journals
	Establish chain of evidence	Categorize number and identify each manifest risk for all sources. Sort based on categorization
Internal validity	Do pattern-matching	Predictions derived from proposition matched to empirical patterns drawn from the cases
	Do explanation building	N/A
	Address rival explanation	
	Use logic models	
External validity	Use theory in single-case studies	Framework is developed from prior research
Reliability	Use case study protocol	A case study protocol is developed and used
	Develop case study database	Case transcripts; Case notes; spreadsheet used to store evidence form multiple sources

framework. Documents and articles, other than the ones already selected, that provide duplicated or similar account and discussion of the cases were excluded. The process began with each researcher independently inspecting the data, line by line, and grouping meaningful data segments according to specific codes consistent with the main concepts outlined in the research proposition and conceptual framework. Codes were developed as the data was examined and themes found as were issues useful for the research. The conceptual risk framework built earlier served to provide initial codes for the risk factor coding scheme but another theme, not in the framework did emerge during the actual coding process; thus a new code was added to the database. The codes and overall evidentiary base are the starting point for analysis and report writing.

We applied the coding process suggested by qualitative researchers for example, Dey [38], Miles and Huberman [81] and Verner et al. [116] on the published case documents. We combined the responses to specific questions posed to witnesses in the court case and documented in the judgment, with those collected from media discussion of the projects. The different sources of documented evidence were analyzed individually and used for triangulation. Triangulation combines different data sources with the aim of corroborating the findings or conclusions [132]. The triangulation of evidence is used to minimize bias. As noted earlier that while we have no control over the questions asked by reporters, which may well have been biased, we trust that what is published is relatively unbiased as newspapers and trade journals. We must also rely on the UK justice system, and trust that the court case was conducted in an unbiased manner. It may be naive to assert that research involving human perceptions is without bias but we consider the data from all sources in the published cases are trustworthy. NVivo 8 was used as a repository for the case data and as a tool to help the researchers organize, analyze and model the data. While the use of a tool such as this may be seen as overkill for a single case the case discussed here is part of a larger body of research into project failure and was to be later integrated with other cases.

During data validation we must check for any missing data in order to ensure that the data collected sufficiently addresses the research proposition. This coincides with the chain of evidence referred to by Yin [132]. In this regard, we ensured that our data collection followed the case study protocol and used the triangulation of data to provide support for answering the research question. A clear link can be observed from the research question to the framework, and to the evidence in the database. The case study report builds on the evidence; and in maintaining the chain of evidence in our study, the report will, in certain parts, present relevant citations from the data to support the findings drawn.

A.2.3. Data analysis

In this phase we must ensure that a complete chain of evidence is kept, alternate perspectives and explanations are considered, and that we develop clear conclusions for practice/further research. We found that a useful method for maintaining a chain of evidence is to highlight and number important material in interview transcriptions. This material can then be numbered, copied, inserted into tables, coded and sorted. The sorting then helps with cross-case analysis should this be possible in later research. The numbering ensures that the material can be traced directly back to its source; i.e., to the actual document, line number and page within the document. At this stage we must consider if there are threats to validity and address them in a systematic way.

In the data analysis phase, a researcher synthesizes the data, interprets meanings and draws conclusions from the data. The analysis strategies outlined in the research analysis plan, Table 3, were executed at this stage. Thematic coding described in the previous section is a crucial early step in the analysis of data. Collecting and analyzing data occurred concurrently. The initial codes developed earlier were examined to ensure data were correctly categorized and codes correctly reflect the tone in the data. Duplicated or similar codes were combined and codes redefined. Some codes were grouped according to larger themes that connect the codes. The resulting coding scheme corresponds to the main concepts in the research proposition. Answers to our research question were sought by exploring and interpreting the coded pieces of text, code by code, identifying patterns that match the research hypothesis. This enhances the external validity of the research findings.

A.2.4. Reporting

The report is an important device to communicate the research and findings to the intended audience, As such, it is imperative that the research is presented in an interesting and convincing way. The quality of study in terms of methods and theories adopted, insights and reasoning demonstrated, and care conducted in the research is judged based on the report [92,132]. Runeson and Höst [92] and Verner et al. [116] provide examples of case study report structures suitable for academic reporting which we have adapted for this paper.

We must ensure that any reporting is easy to read, is suitable for its audience, and is written in an engaging manner. In addition to reporting the research questions and answers, case study context should be adequately covered. Further, related theory, hypotheses and propositions must be clear, and data collection procedures should be presented with inclusion of adequate raw data. Sufficient evidence should be displayed, and it must be made clear how the evidence was selected for inclusion. Additionally, conclusions and implications for practice and future research should be covered [106,128,132].

Various good practices, such as writing up while the case study is conducted (which reduces the effort in the long run) can help the reporting process. We found it useful to start writing as soon as possible – the literature and design can be reported in the protocol. The methodology section was started during the case study planning phase. Descriptive data can be reported prior to analysis. The case study report must be complete with the boundaries between case and context defined; all relevant evidence must have been collected; the case study should have been completed to a predefined schedule and not finished purely because of time constraints. This was certainly true in our case as we consistently wondered if we had missed the judgment because it took so long, and as the case would have been meaningless without the judgment.

The structure of the report has to fit with the case study design and the needs of the reader. The basic format of a single case study narrative report; was appropriate for this case. Although a number of reporting structures have been suggested in the literature we have used the type of sequence normally used for journal articles: (1) issue or problem, (2) review of prior literature, (3) research method used, (4) findings from data collected and analysed, and (5) conclusions and implications [51,116].

References

[1] L.M. Abdullah, J.M. Verner, Risk framework for outsourced strategic IT system development from the client perspective, in: Software Metrics European Forum, Milan, Italy, May, 2008, pp. 1–12, 9-788870-909999.

[2] L.M. Abdullah, J.M. Verner, Outsourced strategic IT systems development risk, in: IEEE Research Challenges in Information Systems, Fes, Morocco, April 2009, pp. 309–320.

[3] J. Ahonen, P. Savolainen, Software engineering projects may fail before they are started: post-mortem analysis of five cancelled projects, Journal of Systems 83 (11) (2010) 2175–2187.

[4] S. Alter, M. Ginzberg, Managing uncertainty in MIS applications, Sloan Management Review 20:1 (Fall) (1978) 23–31.

[5] B.A. Aubert, S. Dussault, M. Patry, S. Rivard, Managing the risk of IT outsourcing, in: Proceedings of the 32nd Annual Hawaii International Conference on System Sciences, HICSS-32, 1999, pp. 2–11.

[6] B.A. Aubert, M. Patry, S. Rivard, Assessing the risk of IT outsourcing, in: Proceedings of the Thirty-First Hawaii International Conference on System Sciences, 1998, pp. 685–692.

[7] B.A. Aubert, M. Patry, S. Rivard, Managing IT Outsourcing Risk: Lessons Learned, 2001, pp. 1–19. <http://gresi.hec.ca/SHAPS/cp/gescah/formajout/ajout/test/uploaded/cahier0111.pdf>.

[8] B.A. Aubert, M. Patry, S. Rivard, A framework for information technology outsourcing risk management, The Database for Advances in Information Systems 36 (2005) 9–28.

[9] B.A. Aubert, M. Patry, S. Rivard, H. Smith, IT outsourcing risk management at British Petroleum, in: Proceedings of the 34th Annual Hawaii International Conference on System Sciences, 2001, pp. 1–13.

[10] D. Baccarini, G. Salm, P.E.D. Love, Management of risks in information technology projects, Industrial Management & Data Systems 104 (4) (2004) 286–295.

[11] B. Bahli, S. Rivard, Validating measures of information technology outsourcing risk factors, Omega 33 (2005) 175–187.

[12] H. Barki, S. Rivard, J. Talbot, Toward an assessment of software development risk, Journal of Management Information Systems 10 (Fall) (1993) 203.

[13] I. Benbasat, D.K. Goldstein, M. Mead, The case research strategy in studies of information systems, MIS Quarterly 11 (1987) 369–386.

[14] M. Bensaou, Interorganizational cooperation: the role of information technology an empirical comparison of US and Japanese supplier relations, Information Systems Research 8 (2) (1997) 107–112.

[15] S.J. Bleistein, K. Cox, J.M. Verner, K. Phalp, B-SCP: a requirements analysis framework for validating strategic alignment of organizational IT based on strategy, context, and process, Information and Software Technology 48 (9) (2006) 846–868.

[16] S.I. Bleistein, K. Cox, J.M. Verner, Validating strategic alignment of organizational IT requirements using goal modeling and problem diagrams, Journal of Systems and Software 79 (3) (2006) 362–378. ISSN: 0164-1212.

[17] S. Bleistein, K. Cox, J. Verner, K. Phalp, Requirements engineering for e-business advantage, Requirements Engineering Journal 11 (2006) 4–16.

[18] Steven J. Bleistein, Karl Cox, Verner, Strategic alignment in requirements analysis for organizational IT: an integrated approach, in: 20th Annual ACM Symposium on Applied Computing, Santa Fe, New Mexico, March 13–17, 2005, vol. 2, pp. 1300–1307.

[19] B.W. Boehm, Tutorial: Software Risk Management, Los Alamitos, IEEE Computer Society Press, California, 1989.

[20] B.W. Boehm, Software risk management: principles and practices, IEEE Software 8 (1) (1991) 32–41.

[21] P. Brereton, B.M. Kitchenham, D. Budgen, Zhi Li, Using a protocol template for case study planning, EASE, 2008. <http://www.bcs.org/upload/pdf/ewic_ea08_paper5.pdf> (retrieved April 2011).

[22] BSkyB Limited, Sky Subscribers Services Limited, HP Enterprise Services UK Limited, Electronic Data Systems LLC, High Court of Justice, Queen's Bench Division, Technology and Construction Court, 2010, pp. 1–468. Retrieved March 31, 2010, from BAILII.

[23] BSkyB Selects Chordiant to Enable Multi-Stage, Multi-Channel Customer Communications Strategy. Business Wire, Navember 7, 2008. <http://findarticles.com/p/articles/mi_m0EIN/is_2002_Nov_7/ai_94039581> (retrieved 13.11.08).

[24] N. Cerpa, J. Verner, Why did your project fail?, Communications of the ACM 52 (12) (2009) 130–134

[25] R.N. Charette, Software Engineering Risk Analysis and Management, McGraw-Hill, New York, 1989.

[26] R.N. Charette, Applications Strategies for Risk Analysis, Intertext, New York, 1990.

[27] A.T. Chatfield, P. Wanninayaka, IT offshoring risks and governance capabilities, in: Proceedings of the 41st Hawaii International Conference on System Sciences, 2008, pp. 1–9.

[28] E.K. Clemons, Evaluation of strategic investments in information technology, Communications of the ACM 34 (1) (1991) 22–36.

[29] E.K. Clemons, B.W. Weber, Strategic information technology investments: guidelines for decision making, Journal of Management Information Systems 7 (1990) 9–28. Fall.

[30] E.K. Clemons, Evaluation of strategic investments in information technology, Communications of the ACM 34 (1991) 22–36.

[31] T. Collins, BSkyB v. EDS judgement could shake IT suppliers. Computer Weekly, September 26, 2008. <http://www.computerweekly.com/blogs/public-sector/2008/09/bskyb-v-eds-a-crucial-judgemen.html> (retrieved April 2011).

[33] V. Connor, Every Dog Has His Day, Kluwer Law International, 2010. <http://kluwer.practicesource.com/blog/> (retrieved 07.04.11).

[34] J.W. Creswell, Qualitative Inquiry and Research Design: Choosing Among Five Traditions, Sage, Thousand Oaks, CA, 1998.

[35] P. Darke, G. Shanks, M. Broadbent, Successfully completing case study research: combining rigor, relevance and pragmatism, Information Systems Journal 8 (4) (1998) 273–289.

[36] G. DeHondt II, G. Nezlek, The cost of risk in offshore systems development, in: Proceedings of the 15th Americas Conference on Information Systems (AMCIS), San Francisco, CA, 2009, paper 718.

[37] P.C. de Weerd-Nederhof, Qualitative case study research: the case of a PhD research project on organizing and managing new product development systems, Management Decision 39 (7) (2001) 513–538.

[38] K. Dey, J. Kinch, S.O. Ogunlana, Managing risk in software development projects: a case study, Industrial Management & Data Systems 107 (2) (2007) 284–303.

[39] R. Dubin, Theory Building, Free Press, New York, 1978.

[40] M.J. Earl, The risks of outsourcing IT (information technology), Sloan Management Review 37 (7) (1996) 26.

[41] K.M. Eisenhardt, Building theories from case study research, Academy of Management Review 14 (4) (1989) 532–550 (see also rejoinder by Dyer & Wilkins, 1991, and response by Eisenhardt, 1991).

[42] K.M. Eisenhardt, Better stories and better constructs: the case for rigor and comparative logic, Academy of Management Review 16 (3) (1991) 627–629 (response to Dyer & Wilkins, 1991).

[43] A.L. Gilbert, M.F. Vitale, Containing strategic information systems risk: control and intelligence, in: Proceedings of the Twenty-First Annual Hawaii International Conference on System Sciences, Decision Support and Knowledge Based Systems Track, Hawaii, USA, vol. III, 1988, pp. 52–60.

[44] R.L. Glass, Pilot studies: what, why, and how, Journal of Systems Software 36 (1) (1997) 85–97.

[45] R.L. Glass, I. Vessey, V. Ramesh, Research in software engineering: an analysis of the literature, Information and Software Technology 44 (8) (2002) 491–506.

[46] R.L. Glass, I. Vessey, V. Ramesh, An analysis of research in computing disciplines, Communications of the ACM 47 (6) (2004) 85–94.

[47] Reyes Gonzalez, Jose Gasco, Juan Llopis, Information systems outsourcing risks: a study of large firms, Industrial Management & Data Systems 105 (1) (2005) 45–62.

[49] W. Han, S. Huang, An empirical analysis of risk components and performance on software projects, Journal of Systems and Software 80 (1) (2007) 42–50.

[50] H. Hosoya, M. Schaefer, Brand zone, in: C.J. Chung, J. Inaba, R. Koolhaas, S.T. Leong (Eds.), The Harvard Design School Guide to Shopping/Harvard Design School Project on the City 2, Taschen, Köln, 2001, pp. 165–173.

[51] M. Host, P. Runeson, Case studies on software engineering programmes, 2008. <http://serg.telecom.lth.se/research/publications/docs/420_Technical%20report.pdf (retrieved 27.10.08).

[52] S. Huang, I. Chang, S. Li, M. Lin, Assessing risk in ERP projects: identify and prioritize the factors, Industrial Management & Data Systems 104 (8) (2004) 681–688.

[53] C.L. Iacovou, R. Nakatsu, A risk profile of offshore-outsourced development projects, Communications of the ACM 51 (6) (2008) 89–94.

[54] J. Jiang, G. Klein, Software development risks to project effectiveness, Journal of Systems and Software 52 (1) (2000) 3–10.

[55] J. Jurison, The role of risk and return in information technology outsourcing decisions, Journal of Information Technology (Routledge, Ltd.) 10 (1995) 239.

[56] B. Kaplan, D. Duchon, Combining qualitative and quantitative methods in information systems research: a case study, MIS Quarterly 12 (4) (1988) 571–586.
[57] M. Keil, P. Cule, K. Lyytinen, R. Schmidt, A framework for identifying software project risks, Communications of the ACM 41 (11) (1998) 76–83.
[58] C.F. Kemerer, G.L. Sosa, Systems development risks in strategic information systems, Information and Software Technology 33 (1991) 212–223.
[59] L. King, Sky EDS dispute 'shows service contract risks', CIO, October 23, 2000. <http://www.cio.co.uk/news/index.cfm?RSS&ArticleID=2123> (retrieved 06.01.08).
[60] L. King, Sky and EDS clash over 'cutting edge' CRM system, Computerworlduk, November 13, 2007. <http://www.computerworlduk.com/management/it-business/services-sourcing/news/index.cfm?newsid=6155> (retrieved 06.01.08).
[61] L. King, Sky and EDS clash over ''cutting edge'' CRM system, Computer worlduk (2007).
[62] L. King, Sky EDS dispute 'shows service contract risks', CIO (2007).
[63] B.A. Kitchenham, L.M. Pickard, Evaluating software engineering methods and tools – Part 9: Quantitative case study methodology, Software Engineering Notes 23 (1) (1998) 24–26.
[64] B.A. Kitchenham, L.M. Pickard, Evaluating software engineering methods and tools – Part 10. Designing and running a quantitative case study, Software Engineering Notes 23 (3) (1998) 20–22.
[65] B.A. Kitchenham, L.M. Pickard, S.L. Pfleeger, Case studies for method and tool evaluation, IEEE Software 12 (4) (1995) 52–62.
[66] H.K. Klein, M.D. Myers, A set of principles for conducting and evaluating interpretive field studies in information systems, MIS Quarterly 23 (1) (1999) 67–93.
[67] R. Klepper, W.O. Jones, Outsourcing Information Technology, Systems & Services, Prentice Hall, New Jersey, 1998.
[68] R. Kliem, Managing the Risks of Offshore IT Development Projects, EDPACS 32 (2004) 12–20.
[69] R. Kunitomo, Seven-eleven is revolutionising grocery distribution in Japan, Long Range Planning 30 (1997) 887–889.
[70] A.S. Lee, A scientific methodology for MIS case studies, MIS Quarterly 13 (1) (1989) 33–50.
[71] A.S. Lee, .Case studies as natural experiments. Human Relations 42 (2) (1989) 117–137.
[72] P. Leedy, J. Ormrod, A Handbook for Teacher Research from Design to Implementation, Pearson Education, New Jersey, 2005.
[73] T.C. Lethbridge, S.E. Sim, J. Singer, Studying software engineers: data collection techniques for software field studies, Empirical Software Engineering 10 (3) (2005) 311–341.
[74] Y.S. Lincoln, E.G. Guba, Judging the quality of case study reports, in: A.M. Huberman, M.B. Miles (Eds.), The Qualitative Researcher's Companion, Sage Publications, Thousand Oaks, CA, 2002, pp. 205–215 (reprinted from Qualitative Studies in Education, vol. 3, no. 1, 1990).
[75] K. Lyytinen, L. Mathiassen, J. Ropponen, A framework for software risk management, Scandinavian Journal of Information Systems 8 (1) (1996) :53–68.
[76] K. Lyytinen, L. Mathiassen, J. Ropponen, Attention shaping and software risk: a categorical analysis of four classical risk management approaches, Information Systems Research 9 (3) (1998) 233–255.
[77] H. Maimbo, G. Pervan, Designing a case study protocol for application in IS research, in: P. Chau (Ed.), Proceedings of the Ninth Pacific Asia Conference on Information Systems (PACIS'05), Hong Kong, 2005, pp. 1281–1292.
[78] N. Makino, T. Suzuki, Convenience stores and the information revolution, Japan Echo 44 (1997) 44–49.
[79] A. McCue, Global IT Outsourcing Deals Rocket up to $163bn. Silicon, 2005. <http://management.silicon.com/itdirector/0,39024673,39127146,00.htm>.
[80] L. McLeod, S. MacDonell, B. Doolin, IS development practice in New Zealand organisations. Journal of Research and Practice in Information Technology 41 (1) (2009) 3–24; P. Michael, EDS legal team hits at BSkyB claim, Financial Times (2007) 4.
[81] M.B. Miles, A.M. Huberman, Qualitative Data Analysis: A Sourcebook of New Methods, Sage Publications, Beverley Hills, CA, 1994.
[82] T. Moynihan, An inventory of personal constructs for information systems project risk researchers, Journal of Information Technology (Routledge, Ltd.) 11 (1996) 359.
[83] J. Natovich, Vendor related risks in IT development: a chronology of an outsourced project failure, Technology Analysis & Strategic Management 15 (4) (2003) 149–409.
[84] T. Nikki, EDS to address charges it deceived BSkyB, Financial Times 21 (2007) 19.
[85] G. Paré, J.J. Elam, Using case study research to build theories of IT implementation, in: A.S. Lee, J. Liebenau, J.I. DeGross (Eds.), Information Systems and Qualitative Research: Proceedings of the IFIP TC8 WG 8.2 International Conference on Information Systems and Qualitative Research, Philadelphia, Pennsylvania, USA, Chapman & Hall, London, 1997, pp. 542–568.
[86] G. Paré, Investigating information systems with positivist case study research, Communications of the AIS 13 (2004) 233–264.
[87] M. Peel, EDS legal team hits at BSkyB claim, Financial Times, 23 (October) (2007) 4 (retrieved 21.11.08, from ProQuest).
[88] D.E. Perry, S.E. Sim, S.M. Easterbrook, Case studies for software engineering, in: Proceedings of the 26th International Conference on Software Engineering (ICSE'04), 2004.
[89] J. Platt, What can case studies do?, Studies in Qualitative Methodology 1 (1988) 1–23.
[90] C. Potts, Software-engineering research revisited, IEEE Software. 10 (5) (1993) 19–28.
[91] W. Rapp, Information Technology Strategies: How Leading Firms use IT to Gain an Advantage, Oxford University Press, USA, 2002.
[92] P. Runeson, M. Höst, Guidelines for conducting and reporting case study research in software engineering, Empirical Software Engineering 14 (2) (2009) 131–164.
[93] P. Savolainen,, J.M. Verner, L.P.W. Land, G. Low, What happens before a project starts?—Project start-up from the supplier perspective. ISD August 2010.
[94] P. Savolainen, Vague project start makes project success of outsourced software development projects uncertain, in: P. Savolainen (Ed.), PROFES2010: The 11th International Conference on Product Focused Software Development and Process Improvement, LNCS, vol. 6156, Springer, 2010, pp. 351–365.
[95] R. Scheepers, H. Scheepers, Contexts of relevance in exploratory case studies in information systems: ubiquitous information technology implementation in organizations, in: S.T. March, A.P. Massey, J.I. DeGross (Eds.), Proceedings of the Twenty-Fourth International Conference on Information Systems, Seattle, WA, December 14–17, 2003, pp. 25–35.
[96] R. Schmidt, K. Lyytinen, M. Keil, P. Cule, Identifying software project risks: an international Delphi study, Journal of Management Information Systems 17 (Spring) (2001) 5–36.
[97] Seaman, Qualitative methods in Software Engineering, IEEE Transactions on Software Engineering 25 (4) (1999) 1–16.
[98] B. Shahzad, I. Ullah, N. Khan, Software risk identification and mitigation in incremental model, in: International Conference on Information and Multimedia Technology, 2009, pp. 366–370.
[99] B. Shahzad, S.A. Safvi, Risk mitigation and management scheme based on risk priority, Global Journal of Computer Science and Technology 10 (4) (2010) 108–113.
[100] G. Shanks, A. Parr, Positivist, single case study research in information systems: a critical analysis, in: Proceedings of the European Conference on Information Systems, Naples, June 2003.
[101] N.C. Smith, The case study: a useful research method for information management, Journal of Information Technology 5 (3) (1990) 13–123.
[102] M. Songini, Media giant BSkyB sues EDS over troubled CRM system, Computerworld 2008 (August) (2004) (retrieved April 2011).
[103] M.L. Songini, Broadcaster sues EDS Over CRMContract, Computerworld 38 (August 23) (2004) 8 (retrieved April 2011).
[104] Sonne Paul, BSkyB wins legal victory against EDS, Wall Street Journal (2010) 27.
[105] Sonne Paul, BSkyB settles claim against ESD, Wall Street Journal (2010).
[106] R.E. Stake, The art of Case Study Research, Sage, Thousand Oaks, CA, 1995.
[107] Standards Australia's, AS/NZS ISO 31000:2009 Risk management – Principles and Guidelines, 1999. <http://infostore.saiglobal.com/store/Details.aspx?ProductID=1378670>.
[108] M. Sumner, Risk factors in enterprise-wide/ERP projects, Journal of Information Technology 15 (2000) 317–327.
[109] M.H.A. Tafti, Risks factors associated with offshore IT outsourcing, Industrial Management & Data Systems 105 (2005) 549–559.
[110] N. Tait, EDS to address charges it deceived BSkyB, Financial Times 22 (October) (2007) 19 (retrieved 10.11.08, from ProQuest).
[111] H. Taylor, The move to outsourced IT projects: key risks from the provider perspective, in: Proceedings of the 2005 ACM SIGMIS CPR Conference on Computer Personnel Research, 2005, pp. 149–154.
[112] H. Taylor, Critical risks in outsourced it projects: the intractable and the unforseen, Communications of the ACM 49 (November 1) (2006) 75–79.
[113] H. Taylor, Outsourced IT projects from the vendor perspective: different goals, different risks, Journal of Global Information Management (JGIM) 15 (2) (2007) 1–27.
[114] H. Taylor, Vendor vs Client risks in outsourced IT projects: an agency theory perspective, in: M.G. Hunter, F.B. Tan, Handbook of Research on Information Management and the Global Landscape, IGU Global, 2009.
[115] The Economist Newspaper Limited, E-Strategy Brief: Seven–Eleven Over the Counter e-Commerce: How to Blend e-Commerce with Traditional Retailing, May 24 2001 (retrieved April 2011).
[116] J.M. Verner, J. Sampson, V. Tosic, N. Bakar, B. Kitchenham, Guidelines for industrially-based multiple case studies in software engineering, IEEE Research Challenges in Information Systems, Fes, Morocco (2009) 347–358.
[117] J. Verner, J. Sampson, N. Cerpa, S. Bleistein, What factors lead to software project failure and whose fault was it?, Ingénierie des Systèmes d'Information 14 (4) (2009) 55–75.
[118] I. Vessey, V. Ramesh, R.L. Glass, Research in information systems: an empirical study of diversity in the discipline and its journals, Journal of Management Information Systems 19:2 (Fall) (2002) 129–174.
[119] L. Wallace, M. Keil, A. Rai, Understanding software project risk: a cluster analysis, Information & Management 42 (2004) 115–125.
[120] L. Wallace, M. Keil, Software project risks and their effect on outcomes, Communications of the ACM 47 (2004) 68–73.
[121] G. Walsham, Interpretive case studies in IS research: nature and method, European Journal of Information Systems 4 (1995) 74–81.

[122] P. Weill, S. Aral, IT Savvy Pays Off: How Top Performers Match IT Portfolio and Organizational Practices. MIT Sloan Research, 2005. <http://web.mit.edu/cisr/working%20papers/cisrwp353.pdf> (retrieved April 2011).

[123] P. Weill, M.R. Vitale, A Place to Space, Harvard Business School Press, Boston, 2001.

[124] S. Whang, C. Koshijima, H. Saito, T. Ueda, S.V. Horne, Seven Eleven Japan (GS18), Stanford University Graduate School of Business, 1997.

[125] Wilkipedia. <http://en.wikipedia.org/wiki/Participant_observation> (retrieved 24.10.08).

[126] L.P. Willcocks, M.C. Lacity, IT outsourcing in insurance services: risk, creative contracting and business advantage, Information Systems Journal 9 (1999) 163–180.

[127] C. Wohlin, M. Höst, K. Henningsson, Empirical research methods in software engineering, in: Reidar Conradi, Alf IngeWang (Eds.), Empirical Methods and Studies in Software Engineering: Experiences from ESERNET, LNCS, Springer, 2003, pp. 7–23.

[128] A.G. Woodside, E.J. Wilson, Case study research methods for theory building, Journal of Business & Industrial Marketing 18 (6-7) (2003) 493–508.

[129] S. Wright, A. Wright, Information System Assurance for Enterprise Resource Planning Systems: Implementation and Unique Risk Considerations. Working Paper, Boston University, 2001.

[130] BSkyB versus EDS ''Wanna buy a used IT system (1)'', The Law Office of Richard Stevens. <www.lorsonline.com/article/bskyb-vs-eds.aspx> (retrieved 23.08.11).

[131] R.K. Yin, Applications of Case Study Research, second ed., Sage Publications, Thousand Oaks, CA, 2003 (first ed., 1993).

[132] R.K. Yin, Case Study Research: Design and Methods, third ed., Sage Publications, Thousand Oaks, CA, 2003 (second ed., 1994).

[133] Zdnet ''BSkyB beats HP in outsourced CRM suit''. <http://www.zdnet.co.uk/news/it-strategy/2010/01/27/> (retrieved 23.08.11).