THE SUCCESS OF SERVICE LEVEL AGREEMENTS

FOR

OUTSOURCED SOFTWARE APPLICATION SYSTEMS

by

CORNELIUS JOHANNES BOTHA

Dissertation submitted in partial fulfilment of the requirements for the degree

MASTERS IN BUSINESS ADMINISTRATION

in the faculty of

ECONOMICS AND MANAGEMENT SCIENCE

at the

NORTH-WEST UNIVERSITY

Supervisor: Professor J. Smith

November 2005

Potchefstroom

ACKNOWLEDGEMENTS

First and foremost my deepest gratitude towards the God Almighty who has granted me the ability and strength to take on the opportunities that he had kindly provided me, one of which is this study. I praise Him.

My deepest appreciation for the masters at the PBS, from whom I have learnt so much over the past three years. A special word of thanks to Professor Johan Smith, whose guidance and advice on this paper I sincerely appreciate.

My earnest appreciation and recognition to my parents, Neels and Petro, who have always believed in me and gave me the confidence to take on the challenges of live. I thank you for granting me the liberty to become the individual that I am today, yet never beyond the boundaries of proper values and beliefs.

To Paula, Hanco, Ebert and Brechtje, you share in my accomplishment with equal right. Thank you dearly for all the sacrifices you had to make over the past three years. I appreciate your prayers, full-hearted support and encouragement during difficult times.

To my employer, MIP Holdings, I appreciate your sponsorship and support. Thank you to all my colleagues and friends for the enthusiasm and interest that you have shown in my endeavour. Special thanks to Ingrid for your assistance with the language and grammatical editing of this paper.

Lastly, my deepest gratitude for you whom I've came upon in life and took from you the inspiration and courage to strive towards that which beholds the deeper, truer meaning of life.

ABSTRACT

Due to the constantly increasing emphasis on the service component of the total delivery by product and service providers, the levels of services delivered by providers are getting scrutinised with similar growing emphasis. Products are taking a backseat to the service experience, which consumers expect from service providers.

Within the software application systems environment products are intangible and, on their own, really only an experience by the users thereof. Software systems providers have to realise that they cannot rely on tangible product features that can cover up for poor service. The software development industry is thus primarily one of service delivery.

This paper conveys research conducted on the success of Service Level Agreements (SLAs) within the outsourced software application system environment. The research starts off with a literature study of previous research done in the related fields, i.e. outsourcing, SLAs and the Systems Development environment. The literature study was followed by the development of a model, suggesting the integration of the SLA life cycle with the Systems Development Life Cycle (SDLC). A measurement instrument was subsequently developed to assess the success of SLA integration as part of the SDLC.

Research data was gathered by means of the developed measurement instrument (questionnaire) and input from 24 experts in the field of outsourced software application systems development was obtained. The 24 experts represented 18 different organisations.

On completion of the analysis of the survey data, certain interpretations were made which led to the following conclusions:

- The survey had shown that in the majority of the organisations investigated, SLAs are not taken into consideration when the systems development process is initiated.
- Experts agreed with the importance of integrating the SLA and Systems Development life cycles.
- Participating individuals supported all the benefits of SLAs, i.e. improving service levels, increasing revenue and enhancing relationships.

To conclude the study, a number of recommendations were made, which will contribute towards enhanced application of SLAs within the outsourced software application systems paradigm. These recommendations include:

- Creation of greater SLA awareness in the software development community;
- Software development organisations should take initiative for addressing SLA requirements from the beginning of the development process; and
- Development methodologies should be extended to formalise the integration of SLAs into the SDLC.

TABLE OF CONTENTS

Page

LIST OF TABLES

Page

LIST OF FIGURES

Page

1 CHAPTER 1: PROBLEM STATEMENT AND RESEARCH PROPOSAL

1.1 Background to the Study

According to Leadbeater and Willis (2001:23) the future consumer culture will have four main ingredients of which the most important is "an emphasis on services rather than physical goods". They argue that within the new economy, the services industries' growth will accelerate much faster than that of the manufacturing industries. ICT will continue to increase productivity and the number of jobs in the services sector will rise, compared to a decline in the manufacturing sector. They estimate that by the year 2050, as little as 5% of the population will be employed by the traditional industrial sector.

In future, the value of manufactured goods such as cell phones will not be perceived to be in the physical product, but rather in the services they offer, like voicemail and Internet access. People will value deliverables, not by the value of the products, but rather by the experience they have when engaged in the use of the product. Some people even refer to this as the *'experience economy'*. (Leadbeater and Willis, 2001:23 - 24)

It is therefore very clear that the quality and levels of service that are offered will have to be of high standards and that measures must be in place to ensure that such service levels be maintained.

Although Service Level Agreements (SLAs) have been around for many years, there are still numerous cases today where the Agreement merely ends up as a static document without adding value to the organisation.

Application SLAs measure application performance. The software provider organisation agrees to a certain level of responsibility, different classes of service, performance parameters, and a manner of calculating both the demanded performance levels and penalties that result if it doesn't perform its services as contracted. On the contrary, the SLA is also intended as

protection to the service provider organisation, stating clearly what the client organisation's responsibilities are in order to enable the service-providing organisation to achieve the required service- and product levels. It is therefore clear that both the service providing organisation and the service-receiving organisation have equal commitment to delivery as far as the Service Level Agreement is concerned.

1.2 Problem Statement

One of the main problems regarding SLAs within the software application industry is that no *standards* for drafting and managing SLAs exist (McKeefry, 2001). This results in Information and Communications Technology [ICT] managers avoiding the issue, knowing that it is expected of them to deal with it, but with no enthusiasm or commitment. The result is more often than not, a diluted attempt that, if ever put into use, causes tedious disputes with associated negative returns.

Software applications (systems) are by nature intangible, hence in many aspects extremely difficult to quantify and therefore difficult to measure. The only real measurement is the experience that the user has when using the system. This should then be measured against the expectations that such users had, prior to system use. "In many cases, service level agreements (SLAs) are for technical performance and don't model consumer experience. In the end, effective SLM [Service Level Measurement] must start with assessing the consumers' experience; measuring the services' actual quality against customer expectations; and ensuring that the underlying IT infrastructure delivers." (Flannery, 2003)

A Service Level Agreement, if used to its full potential, can be one of the Chief Information Officer's greatest tools in maximising the ICT service to internal clients. For added benefits to be derived from SLAs, the agreement needs to be treated as an information source complete with *reporting* capabilities. Only if SLAs are *reported* on and the information made available to stakeholders, can it add value over and above being seen as a tool for justice.

A characteristic specific to the software industry is the rate of change. It therefore does not make sense to have SLAs that, when put in place, are cast in stone. This is a common problem, resulting in unusable and worthless documents adding no value to service levels.

1.3 Goals of the Study

The purpose of this study is to determine if SLAs put into practice do in fact deliver the results they are intended to and if so, do they in reality result in raised service levels.

A model will be compiled that could be used to assess the implementation of SLAs for outsourced software application systems. Actual implementation scenarios will be measured against certain criteria contained within the model.

Apart from derivatives obtained from the information gathered from various organisations, a further result of the study will be certain guidelines for Chief Information Officers and executives on how to compile and implement SLAs to ensure maximum value, taking into account the specific elements and constraints of the Outsourced Software Application systems environment.

1.4 Research Methodology

This study is a combination of *qualitative* and *quantitative* nature. Miles and Huberman (1994:1) claimed that during the decade preceding their publication, "more researchers in basic disciplines and applied fields (. . . organisational studies, business studies, . . .) have shifted to a more qualitative paradigm. They state that qualitative data "are a source of well grounded, rich descriptions and explanations of processes". They further argue that qualitative data lead to serendipitous findings and new integrations that again result in new or revised conceptual frameworks.

The field research results will contain both predefined multiple option questions as well as open-ended questions that allow respondents to voice their opinion on the subject. This allows for the use of quantitative and qualitative data analysis.

1.4.1 Research Classification and Reasoning

Within the framework of this study the research classification will be mainly qualitative with inductive reasoning. Cooper and Schindler (2000: 764 – 770) describe qualitative classification as gaining or obtaining a detailed understanding of the insights of the research dilemma or problems proposed. Their definition for inductive reasoning is that conclusions are drawn from particular facts or pieces of evidence selected from research previously performed.

1.4.2 Research Design and Strategy

The research design of this study will be exploratory in nature. Cooper and Schindler (2001:762) state that exploratory research implies that the proposed research focuses on the understanding of the research dilemma in detail by gathering background information. Therefore, an empirical study will be preceded by a literature study to obtain intensive background on SLAs and the application thereof within the Outsourced environment, specifically related to Software Application Systems. Sources to be used for the literature study will include various textbooks on the topic, recurring weekly- and monthly publications and journals on Information and Communications Technology and Outsourcing.

Subsequent to the literature study, a model will be designed and applied when conducting the empirical study and field research. Silverman (2000:77) defines the purpose of a *model* as providing an overall framework for how we look at reality. He states that models reflect what reality is like, the basic elements it contains and the nature and status of knowledge. Mouton and Marais (1989:140-141) argue that its heuristic function is the most prominent attribute of a

model. They state that models are constructed based on certain theories of which much is known and proven, and where there exist certain commonalities between the 'known' and the yet 'unknown' systems.

1.4.3 Data Collection and Analysis Method

The empirical study will entail the accumulation of information from various organisations that make use of outsourced software applications. Information will be gathered by means of questionnaires compiled as part of the study and quantitative as well as qualitative analysis will be performed. Apart from questionnaires, information will be obtained by means of non-structured interviews with both members of service-providing and members of service-receiving organisations. Observation will be used as a third method of research.

On completion of the empirical study, the results will be processed and interpreted in order to convey the outcome thereof as well as possible recommendations.

2 CHAPTER 2: LITERATURE STUDY

2.1 Partnerships

Michael Cunningham stresses the importance of business partners by stating that if an organisation should only use internal resources, it would dampen organisational growth. He continues his argument by saying that partnership systems, operating through a combination of business rules and technology infrastructure, provide food for organisational growth. (Cunningham, 2001:2). One of the two reasons he gives for using business partners is improvement of the internal operations of the business.

Cunningham puts all partnerships into three categories, i.e. Inside-, Nearby- and External Partnerships. *Inside* partnerships are partnerships that are formed between employees and the organisation as well as certain trusted parties that support core business operation. *Nearby* partnerships assist in the tactical implementation of business goals. Outsourcing of business critical functions would fall within this category, hence organisations need to exercise great control over such partners. *External* partnerships assist the business model but do not require daily or strategic co-ordination. (Cunningham, 2001:5). It becomes apparent that where an outsourcing partnership is present, the organisation outsourcing the service must be able to measure and control the levels of the service being outsourced.

2.1.1 Culture

Culture is undoubtedly one of the most significant aspects of any partnership. Understanding of the values and culture of both parties will assist in the resolution of almost all other matters that affect the partnership. Cunningham (2001:85) stresses that culture needs to be taken into account very early in the process of forming partnerships. While is it not important to have a complete cultural match there is one critical issue that needs to be in place: *Trust*. He further states that once trust is lost, months of effort can be wasted at a huge cost.

Every ASP (Application Service Provider) agreement is a partnership. What makes these partnerships even more important is the fact that the client is relying on the software provided by the ASP to run the business every minute of the day, hence the ASP must agree to keep the system running. This, in fact, requires a great deal of trust. Cunningham (2001:37) acknowledges that the culture of partnerships is very closely interlinked with policies and procedures. Service Level Agreements (SLAs) can therefore have a significant impact on the outsourcing partnership and hence can hugely impact (often negatively) the development of new relationships.

2.1.2 Agreements

Cunningham (2001:83) states that many partnerships fail as a result of delays in the finalisation of agreements, which could take months to reach a point of acceptance by both parties. Contributing to the delay in finalising the agreement are issues such as understanding what needs to be protected, measured and rewarded or penalised.

2.2 Outsourcing

2.2.1 Introduction

One of the first incidents of outsourcing dates as far back as 1963, where Perot's Electronic Systems (EDS) handled data processing services for Frito-Lay and Blue Cross. Where early forms of outsourcing typically involved 'single-system' contracts, more recent outsourcing agreements span multiple systems and the momentous transfer of assets, leases and staff to outsourcing companies. (Lacity & Hirschheim, 1995:1 - 2) The concept of outsourcing really became popular in the 1990's. In 1992 an estimated 20% of the Fortune 500 companies in the USA were outsourcing at least a part of their ICT function. (Howard Anderson quoted by Frenzel, 1992:542). This is affirmed by Lacity and Willcocks (2000:355) who note that IT Outsourcing revenues had grown from $US 9 in 1990 to $US 121 in 2001. More importantly, they state, "IT outsourcing has outlived the five-year period typical of a management fad". In a

study of 61 outsourcing agreements, they found that prior to 1989 there was a success rate of 40%; between 1989 and 1991 a success rate of 85%; and since 1991 the success rate was 90%.

Information is becoming increasingly important to businesses as part of their competitive advantage and is essentially used as a commercial tool. Colin Turner (2000:1) refers to the current economy as 'The Information Economy'. He argues that "the adoption of ICTs and the evolution of the information economy is driving the changing competitive environment of business". Turner states that information and knowledge assets become increasingly central to an enterprise's commercial success. He highlights that "if ICTs are to work to competitive advantage then their application has to be about more than simply inducing increased operational efficiency within the business". He believes they need to recognise that "outsourcing is a credible alternative for a number of functions where the organisation does not possess the necessary in-house expertise to maximise the value from a particular information capability". (Turner, 2000:187-188).

Lacity and Hirschheim (1995:3 – 4) caution that "outsourcing must be taken seriously". They believe that the key question is not "whether to outsource or not, but rather where and how to take advantage of the rapidly developing market of IS services providers". They refer to a *"Sourcing Dilemma"* that companies face when they need to consider how to optimally obtain their needed ICT services.

2.2.2 Reasons for Outsourcing

Lacity and Hirschheim (1995:2) believe there are four main reasons why organisations outsource ICT functions to vendors, i.e.:

Table 1: Reasons for Outsourcing

REASON	BENEFIT	BENEFIT DETAILS
FINANCIAL	Reduce Costs	Vendors enjoy economies of scale
	Improve Cost Control	Vendors implement cost controls that more directly tie usage to costs.
	Restructure ICT Budgets	ICT restructure budgets from complex capital based to flexible operational.
BUSINESS	Return to Core Competencies	Concentration on what the organisation does best.
	Facilitate Mergers and Acquisitions	Outsourcing should solve technical incompatibilities, absorb excess ICT assets and absorb additional ICT employees generated by mergers and acquisitions.
	Provide for Start-up Companies	Outsourcing is a cheaper and quicker way to provide ICT.
	Devolution of Organisational & Management Structures	Outsourcing is a means to reduce headcount and associated costs and overhead.
TECHNICAL	Improve Technical Service	Unsuccessful delivery of ICT by internal ICT department.
	Access to Technical Talent	It is difficult for organisations to find and retain technical skills.
	Access to New Technologies	Access to vendors' R&D facilities.
	Focus Internal ICT Staff on Core Technical Activities	Internal ICT to focus on core ICT functions like developing custom software and outsourcing non-core ICT functions.
POLITICAL	Reaction to the Efficiency Imperative	Managers measure ICT solely on cost efficiency and then outsource to increase efficiency.
	The Need to Acquire New Resources	By outsourcing, approval for the acquisition of new resources such as hardware, additional personnel and cash is inevitably obtained.
	Reaction to the Bandwagon	In reaction to favourable outsourcing reports.
	Reduce Uncertainty	With ICT demand being unpredictable, it is difficult for ICT managers to do planning.
	Eliminate a Troublesome Function	Senior management do not properly value the ICT function and when it is performing poorly, outsourcing removes the problem.

	Enhance Credibility	ICT in general is not fully valued by senior management; hence it is difficult for ICT managers to move up in ranks. By taking the outsourcing decision, ICT managers gain credibility with senior management, which allows for promotion.

Source: Compiled from Lacity and Hirschheim (1995:23 – 29)

Although many reasons might be stated for organisations to outsource, the growth in the outsourcing market is attributed to two reasons.

(1) A result of strategic decisions for organisations to focus on core competencies; and

(2) The perception of senior executives that internal ICT is a non-strategic business component and is considered to be an overhead, a cost to be minimised. (Lacity and Hirschheim, 1995:2- 3)

2.2.3 Degrees of Outsourcing

The outsourcing phenomenon is a widely-used one. Organisations that make use of outsourcing do so in many forms and to many different extents. An organisation can outsource its entire ICT function, including the management thereof. In such an event the organisation could make use of an outsourcing company to manage all ICT aspects, leaving the client organisation with no ICT staff. At the other extent, an organisation can choose to outsource only part of its ICT function, and possibly none of its ICT human resources function. An example hereof is where an organisation only outsources the development of software application systems.

Within the framework of outsourced software application systems, there are three basic alternatives.

1. Outsourcing whereby the client organisation makes use of a pre-developed software application. The application is then customised and parameterised

according to the client's specific business needs and the outsourcing company remains responsible for the ongoing maintenance thereof.

2. Outsourcing in the form of the client organisation contracting a software development company to develop a software application system from scratch according to specific client specifications. Once the system has been fully developed, the ongoing maintenance can either be performed in-house by the client organisation ICT department or the outsourcing agreement can be extended to include the ongoing maintenance and enhancement of the system.
3. The third and most significant form of outsourced software application systems is referred to as Application Systems Provision (ASP). In this case, the outsourcing company does not only develop the application, but also hosts the application. The client organisation then accesses the application remotely.

Cunningham (2001:17) defines Application Service Providers (ASP) as organisations which deliver complete software solutions for companies where the client:

- Obtains sophisticated software solutions at the fraction of the entry price;
- May have the application running in a matter of weeks;
- Has very little need for expensive internal ICT staff to support the system; and
- Experiences a very low overall cost of ownership.

Cunningham (2001:215) predicted that "as communications infrastructure becomes more stable, lower priced and more secure, the move towards ASPs will really take hold".

When selecting an ASP, attention should be given to the ASP's ability to adapt to the changing needs of the client organisation. It is thus important for an ASP to offer scalable as well as reliable services. Organisations should avoid a 'one-size-fits-all' offering from ASPs and ensure that the ASP must have the ability to meet the specific application needs of each individual client. (Haig, 2001:182-184).

2.2.4 Phases of Outsourcing

Lacity and Willcocks defined six phases with related activities and objectives that define the complete outsourcing agreement. Table 2 lists the relevant phases and their activities:

Table 2: IT Outsourcing Phases and Activities

PHASE	ACTIVITIES	OBJECTIVES
Scoping Phase	• Identify core IT capabilities • Identify IT activities for potential outsourcing using business, economic, and technical criteria.	Identify flexible IT organisation, including IT activities for potential outsourcing
Evaluation Phase	• Measure baseline services • Measure baseline costs • Create RFP • Develop evaluation criteria • Invite external and internal bids	Select best and final offer
Negotiation Phase	• Conduct due diligence to verify RFP baseline claims • Negotiate Service Level Agreement • Create responsibility matrices • Price work units • Negotiate terms for employee transfer • Negotiate mechanics for contractual change, including benchmarking, book accounting, non-exclusivity clauses, and pricing schedules	Sign contracts
Transition Phase	• Distribute contract to IT users • Interpret the contract • Establish post-contract management infrastructure and processes • Implement consolidation, rationalisation, and standardisation • Validate service scope, costs, levels, and responsibilities for baseline services • Manage additional service requests • Foster realistic expectations of supplier performance • Publicly promote the contract	Establish operational performance
Middle Phase	• Benchmark performance to (theoretically) reset prices • Realign the contract to reflect changes in technology and business • Involve the supplier in more value-added areas	Achieve value-added above the operational performance

Mature Phase	• Recalibrate investment criteria to reflect shorter time horizon for recouping investments • Determine if the relationship will be terminated or extended	No lapses in operational performance during final transition

Source: Lacity and Willcocks (2000:369)

As defined in Table 2 above, the Service Level Agreement needs to be agreed on during *Negotiation Phase*, which is rather early in the outsourcing process and before the actual service is outsourced. During the *Transition Phase*, opportunity is given for service levels to be validated and after the outsourcing implementation, during the *Middle Phase*, all contracts (including the SLA) need to be realigned to reflect changes in technology and business.

2.2.5 Strategy

Although the obvious reason for outsourcing is of a financial nature (whereby economies of scale makes it cheaper for an outsource company to perform the same ICT function as opposed to performing the same function in-house) the actual decision is of strategic nature. It is extremely difficult to reverse an outsourcing decision. Hence, outsourcing is a long-term partnership that should be carefully considered. Before deciding to outsource, clear answers should be obtained to strategic questions such as: Can the vendor provide long-term, high quality service? How will the client organisation and the vendor work together on problems, changes, recovery and operational issues? (Frenzel, 1992:542-453). Lacity and Hirschheim (1995:2) support this argument:

> *"Many companies have recently abandoned their diversification strategies – once pursued to mediate risk – to focus on core competencies. Senior executives have come to believe that the most important sustainable competitive advantage is strategic focus by concentrating on what the organisation does better than anyone else while outsourcing the rest."*

2.3 Service Level Agreements (SLAs)

2.3.1 Introduction

According to Carroll Frenzel the purpose of a Service Level Agreement is to ensure an understanding by the client organisation using the service and the ICT organisation providing the service. He argues that cost on the one hand needs to be balanced with the service levels offered on the other hand. The costs encountered by the organisation must be justified by the responsibilities discharged to the ICT organisation. (Frenzel 1992:324)

To develop (establish) a Service Level Agreement is a process that does not only involve two (or more) parties, but in fact requires extensive work and input from those parties. There is no 'off-the-shelf' SLA that will have any value. Frenzel states: "The process that establishes service levels involves a significant degree of negotiation between organisations. Users, their managers, and the firm's senior managers must be involved."

Köppel *et al.* states that a SLA will cover: (i) service hours; (ii) service availability; (iii) user support levels; (iv) response times; (v) restrictions; and, very important, (vi) service levels to be provided in a contingency.

They define the following questions to be answered by any SLA:

- Who delivers which service when?
- What happens if problems arise?
- What is the service and how is the service quality assessed?
- How to work changes into the SLA.

2.3.2 Service Levels

The main component or aspect of SLAs is the definition of *Service Levels*. Complete and specific service levels are the determining factor in the success of SLAs. These are also the

most complex part of SLAs and it is important that all the stakeholders agree on these aspects and how they are defined. Larson (1998:128) confirms this and states that SLAs should contain service levels that:

- Can be measured and managed;
- Can be audited;
- Can be provided at an economic price; and
- Give maximum value to the users of the services.

2.3.3 Internal- vs. External SLAs

SLAs can be divided into two basic types, i.e. *Internal SLAs* and *External SLAs*. As can be seen from Table 3 below, there are distinct differences in approach between the two types. What is interesting however is the fact that in both types *responsibilities* need to be defined.

Table 3: Internal- vs. External SLAs

INTERNAL	EXTERNAL
o Terminology is "understood"	o Terminology defined
o Not legalised	o Legalised
o Responsibilities defined	o Responsibilities defined
o Service definition not precise	o Service definition precise
o Processes understood	o Processed defined
o Cost rather than price, if at all	o Price rather than cost

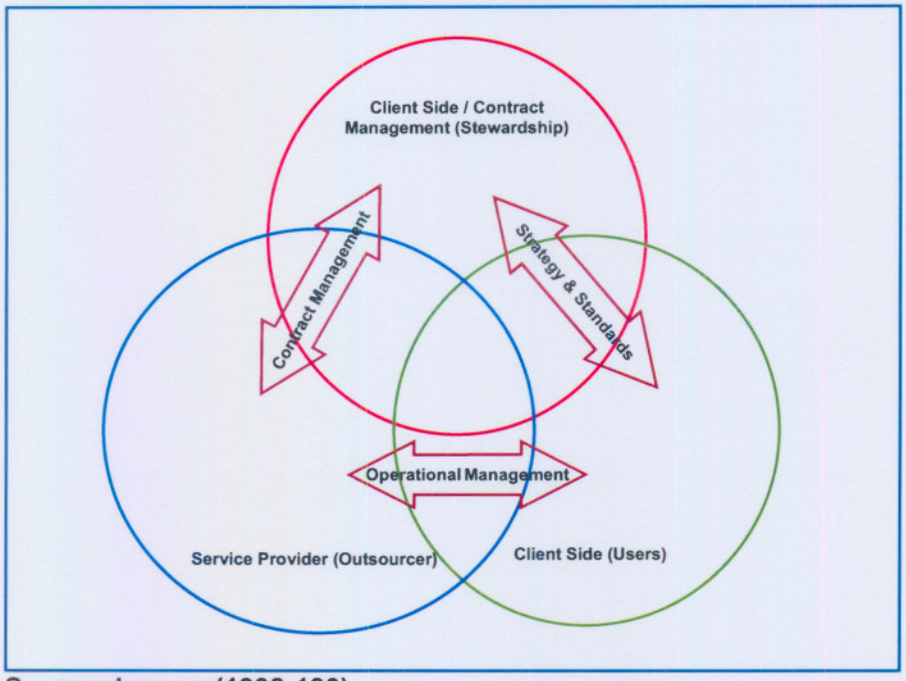
Source: Larson (1998:128)

2.3.4 Stakeholders in the SLA

As illustrated in Figure 1, Larson (1998:129) defines three main stakeholders in the management of SLAs i.e.

- The Steward (Client-side contract manager);
- Service Provider; and
- Users.

Figure 1: Main Stakeholders in SLA Management



Source: Larson (1998:130)

2.3.4.1 The Steward (Client Side)

The role of the steward is primarily one of measuring and auditing the service levels of the outsourcing company compared to the set service levels. In conjunction with the users, the steward is responsible for setting the structure within which services are delivered. Aspects such as contract terms and variations, standards, strategies and budgets are all contained in the set structure. In addition, the steward is also responsible for issues such as implementing contract variations, dealing with charges and compensation, and contracting for new services and projects. (Larson, 1998:129 - 130)

2.3.4.2 The Service Provider (Outsourcer)

The service provider (outsourcer) interacts with both the Steward and the Users. His role in terms of these two entities is comprised of:

The Steward (Client Side)

The function of the service provider is mainly the delivery of the contracted services. The Steward is responsible for maintaining the contract and variations to it, in conjunction with the supplier.

Users (Client Side)

The Outsourcer's responsibilities towards the users are resolving the day-to-day operational problems and handling the request and implementation of ongoing changes to the system.

(Larson, 1998:130)

Should the services not be outsourced but provided within the organisation, this function will be fulfilled internally where the ICT department will act as a supplier.

2.3.4.3 Users (Client Side)

The users are responsible for utilising the agreed services delivered by the outsourcer and maximising such services in order to add value to the organisation. Users only become involved in the operational management should the delivered service levels or contents deviate from the agreed standards. (Larson, 1998:130)

Singleton *et al.* (1988) state that involving users in the determination of acceptable service levels is also of critical importance when it comes to measurement of SLAs, especially with regards to the measurement of perceived performance. Singleton *et al.* further claim that users should be a party in the sign-off process when bringing SLAs into effect.

2.3.5 Changing Needs

Probably one of the biggest reasons which cause SLAs to fail or become meaningless is the fact that they are not continuously re-aligned with the ever-changing business environment and needs. Luevano describes the SLA as "a living document; that is, it is not cast in concrete because it is frequently changing" (1991, 125). Frenzel supports this by adding that, because of the dynamic nature of the business, changing user requirements and new technology, the process of establishing service levels is an iterative one. (1992:324)

2.3.6 The SLA Contents

Köppel *et al.* group the SLA contents into two categories, i.e.:

1. General elements; and
2. Application specific elements.

They note that SLA elements should not be confused with service elements, although SLA elements could include service elements. Each element should be properly described to ensure a clear understanding of the meaning and importance of the element. A proper description might also point out possible conflicts arising from diverging interests of provider and client.

They suggest elements be categorised, which should assist in obtaining a better overview of the different elements and the negotiation of the SLA. Typical categories could include measurability and importance.

2.3.6.1 General Elements

These will be part of any SLA and are not application-specific. They typically relate to the contractual aspects of the SLA. These elements would include contact personnel, the change process of the SLA, the review periods, costs, duration of the contract, sanctions in the event of service level violation, and obligations of the provider in case of contract termination.

2.3.6.2 Application Specific Elements

These are the elements that are specific to the characteristics of the application and represent service descriptive elements and service reporting elements. These should represent the maximum set of application-specific elements and each individual element should be named. Köppler *et al.* warn that these requirements are difficult to achieve but ensure that completeness is attained. Once elements are defined as part of the SLA negotiation process, they can always be refined when additional information becomes available or if the needs change.

2.3.7 Rules for SLAs

Cunningham suggests the following general rules to be applied when drafting SLAs:

- Minimise the legalese; try to make the agreement "legal" but understandable. (2001:101)
- He believes that "agreements that are unfair or unwise are guaranteed to fail". (2001:221)

2.3.8 Measurement

Cunningham (2001:104) emphasises the fact that the effectiveness of partnerships should be measured. He states that a lack of measurement is the main reason why partnerships fail.

While measurement methodologies within other (non-ICT) departments in the organisation seem easier and more quantifiable, Singleton *et al.* argue that ICT should be measured in exactly the same manner. They argue that there is no single measure that can be used for measuring the total ICT function, but the same applies to any other department. They state that a "set of measures, appropriate for the activity of objectives being measured" is required. (Singleton, 1988)

In 1965 Robert Anthony (cited Singleton *et al*, 1988) identified three levels of management within an organisation that each requires different measures. These levels are:

- Operational, where *efficiency* is the key;
- Managerial, where *effectiveness* of the organisation and the management are essential; and
- Strategic, where the *competitiveness* of the enterprise itself is of importance.

According to Anthony, ICT initially was focused almost exclusively on operational and transactional systems, and therefore the majority of all measures in use today still revolve around these aspects. He argues that these statistic-like measures are mostly meaningless for the end-users.

When service levels are not measured as part of a SLA or when such measurements are inaccurate or ambiguous, client organisations will start developing their own tracking measures. This in turn will lead to conflict and disputes. To avoid such wasted energy, clear and unambiguous measurements should be agreed on as part of the SLA. An important aspect of measurement is the fact that measurement must take place at the delivery point. It is useless to report a job as completed if it has not yet been delivered to the end user. Whatever system is used to measure performance must have high credibility with the client, maintaining bi-lateral trust. Only if mutual trust and confidence in the measurement systems exists, will objective discussions be possible in the event of performance being lower than planned or agreed. (Frenzel, 1992:331)

2.3.8.1 Key Result Areas

Hodgetts argues, "the key to effective measurement is to identify those outputs or results that provide the most accurate and useful feedback". He refers to those areas as *Key Result Areas (KRAs)*. Measuring KRAs means focussing on fewer targets and is based on the assumption that if performance in the KRAs is satisfactory, it is representative of the

performance of the complete operation. (Hodgetts, 1998:116 - 119). The measurements of the SLA should therefore be focussed on key aspects of the service delivery that have the significance of representing the complete service delivery, providing an indication of the success of an outsourcing partnership.

With specific reference to the Outsourced Software Application Systems service delivery package, there are two main categories of components that need measurement:

- The first category includes those elements that can technically or scientifically be measured, e.g. system availability, number of problems solved within the agreed timeframes.
- The second category consists of less-easily-measured aspects. These include the more abstract and subjective components of the bigger service delivery phenomenon. This is a measure of how the service delivery is 'experienced' by the client; how the users perceived the service to be; how client management recognise the value proposition of the outsource company. This will be referred to as 'Measuring Perceived Value' and discussed in more detail in section 2.3.8.2 below.

Singleton *et al.* (1988) refer to these two categories as the *actual-* and *perceived* performance. They argue: "If the actual and perceived differ significantly, it is a clear indication that the measurement systems are not working properly.

2.3.8.2 Measuring Perceived Value

One of the lessons that Hodgetts (1998:18 – 19) had learned in studying some of America's most successful organisations, is that "Customer Value Added is the Name of the Game: *Perceived Value is Critical*". One of the companies he had studied, Lucent Technologies, had defined some premises that form the basis for their quality-driven strategy, one of which is *'People buy on perceived value'*. They found that "customers who saw themselves as receiving higher quality were also willing to pay higher prices". From this study one can make the

conclusion that SLAs not only need to measure the quantifiable aspects of service delivery, but almost more importantly, need to measure the perceived value of the service as experienced by the end users.

User satisfaction surveys are a meaningful measure of determining what the users' perceptions are of the service that is being delivered by the outsource company. In many events, the outsource company can maintain agreed service level targets, but what users experience is less than that. User satisfaction surveys can even be used to 'detect' incorrectly established service agreements. (Frenzel, 1992: 332)

2.3.9 Performance Measurement

By measuring the performance of the complete service delivery, i.e. the inherent system performance as well as the related support from the outsourcing company, it reflects the adherence to the SLA requirements is assured.

2.3.9.1 Performance Metrics

The following aspects are of critical importance when it comes to the measurement of performance:

Availability

Availability represents the proportion (percentage) of time that the system is actually accessible and usable over the total measurement period, e.g. actual hours available from the total hours in a week or month.

Reliability & Accuracy

This measures the number of incidents that the system (or function within the system) not accessible during a set measurement period, e.g. three times during a week.

Response

Response measures the time delay between a demand for service and the subsequent reply. This is also measured as *Turnaround Time*, *Transfer Time or Cycle Time*.

Throughput

This is a measurement of volumes to be processed within a certain time frame. These are mainly system processes that process high volumes of records or data sets and store the resulting data. This metric can also be applied to large reports, or extracts, that process high volumes of data and create a report from the results.

Larson (1998:130) adds the following additional metrics to be used for the measurement of system performance:

Serviceability

He argues that this is an extension of reliability and measures the duration of available time lost between the point of service failure and service reinstatement, e.g. 95 percent of failures will be addressed and resolved within 30 minutes of the failure being reported.

User Satisfaction

This method of measurement is of non-scientific nature, but is indeed acknowledged as a very powerful indicator of performance. (See section 2.3.8.2). Larson (1998:131) suggests that frequent user satisfaction surveys be employed to measure user satisfaction over time. This is a definite indicator of performance but, due to its subjective nature, it is not recommended for inclusion in the SLA as a legally binding measurement of performance.

2.3.10 Congruence of Expectations and Performance

Even within the spirit of mutual trust and co-operation it often happens that the outsourcing organisation cannot live up to the expectations of the users. This outcome is a dilemma with two undesired alternatives, i.e.

- The SLA describes services that the ICT company cannot maintain, or
- The SLA describes services that the ICT company can maintain, but which are unacceptable to the users.

The single most important factor hampering the resolution of this problem is cost. It is the role of management to assess the situation and if possible, increase budgets in order to 'purchase' better service levels. If an increased budget is not possible, management should intervene in the process of managing user expectations downwards. (Frenzel, 1992:333 – 334)

2.3.11 Drafting the SLA

Although there are many possible processes for drafting an SLA, Luevano suggest, the following process to be followed for getting an SLA in place. (1991:126 - 130). (*The process as described by the Luevano does not specifically address the outsourcing component and hence will be slightly modified to include the outsourcing company as stakeholder)*

2.3.11.1 Planning

Within the client organisation, ICT cannot be solely responsible for drafting and agreeing SLAs. The process requires input from all levels of the organisation. Very often, the SLA is seen as a legal document, being enforced by ICT to ensure the outsourcing organisation lives up to its promises. Luevano (1991:126) argues very strongly that, should the ICT manager perform the SLA process in isolation, it will not work. He states that it requires total management support, from the chief executive officer all the way down to the lowest level of the organisation.

As with all management activities and projects, drafting and establishing SLAs should be carefully planned. Luevano (1991:126 - 27) suggests that a series of planning sessions be held between the ICT department and the system users, i.e. within the client organisation. Straightforward communication between all parties is required to provide a solid foundation on

which the SLA can be built. He warns that this is not an easy task and requires considerable time and effort.

> *"During these sessions, development of strategic plans and goal setting for the organisation as a whole become critical tasks that are needed to identify service-level goals and objectives. The integration of goals and strategies throughout the organisation must be included at this level to avoid duplication of effort and lack of clear and concise goal setting".*

2.3.11.2 Formalising the Agreement

Selecting Representatives

The first stage in formalising the agreement is to appoint representatives from the ICT department as well as the user corps. The ICT candidate will act as the Steward (as described in section 2.3.4.1), also referred to as the SLA manager. The user representative requires the full support of his/her line management to ensure no breakdown in communication. It is also a great advantage if this person has the clout in the user corps to ensure that fellow users realise that they too have a responsibility in properly defining and communicating their needs.

Reviewing Standards

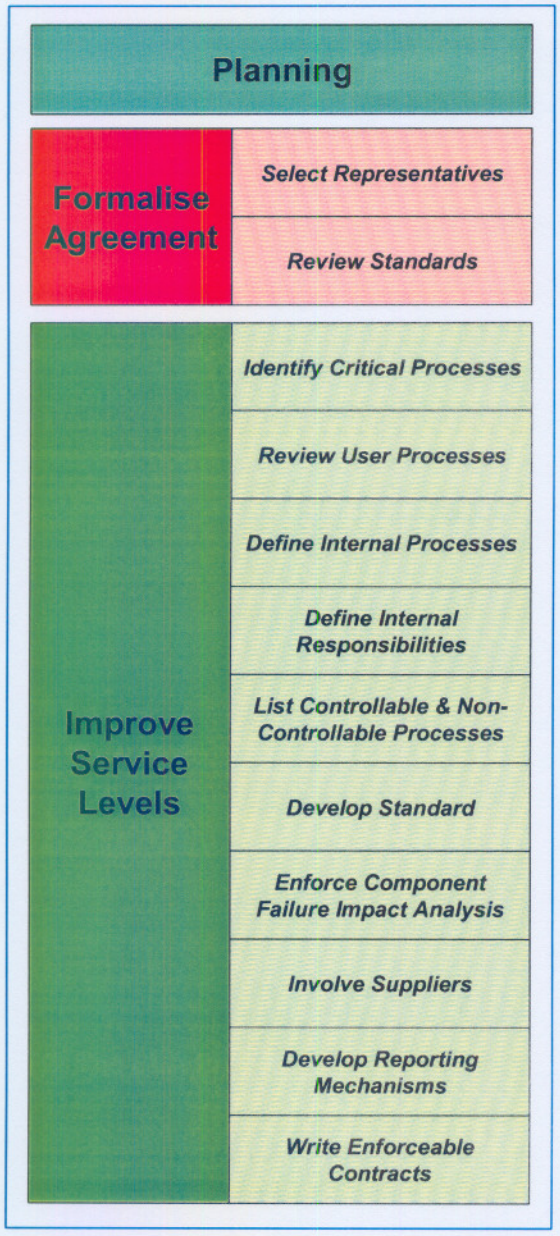
The ICT manager should review existing ICT standards and formulate any needed standards. The result of this process is to include process reviews as an integral part of the eventual SLA, which would allow the flexibility to adjust the SLA according to changing future requirements.

2.3.11.3 Improve Service Levels

Identify Critical Processes

Processes that are critical to the running of the business should be identified and documented. These processes will normally be those that will be closely measured and monitored.

Figure 2: The SLA Process



Source: Own

Review User Processes

The main purpose of this stage is to ensure that the business processes used by users are understood by the ICT department and that expectations are aligned between users, ICT and the service providing company.

Define Responsibilities

The different responsibilities for both the ICT department and the service provider company should be defined. This does not stop at department or organisational level but is defined down to individual level. The aim of this stage should be the primary mission of everyone in both the ICT department as well as the outsourcing organisation, namely meeting user expectations.

List Controllable and Non-Controllable Processes

Processes and particularly non-controllable processes should be identified and listed. Should a failure of any of these have a serious impact on service delivery, it should be assessed and a contingency put in place. This does not only apply to the client organisation but to the outsourcing organisation too. A typical example of this would be an organisation situated in an area where power failures are common. An obvious contingency would be the use of Uninterruptible Power Supplies (UPSs), but this might have a huge cost implication. Should a decision be taken not to purchase UPSs, this should be stated in the SLA and downtime as a result of power failure will thus be excluded from the measurement.

Develop Standards

The primary reason for the development of standards is to ensure that the users of the client organisation properly use the ICT function outsourced. Very often, users 'misuse' outsourced services, which directly result in a perceived sub-standard delivery by the outsourcing company. It is therefore a very important aspect for the outsourcing company in the SLA process and should be thoroughly defined.

Enforce Component Failure Impact Analysis

In any ICT system, there are normally a certain number of components of such a system that are very vulnerable to failure. Many times this is because of continuous development in those areas. One reason for this is the influence of legislation that requires systems to be adapted and changed to adhere to a change in such legislation. It is wise to analyse the impact

of a possible failure of these components and prepare plans to resolve such failure with minimum impact. It is better to be proactive than reactive in problem resolution.

Involve Outside Vendors

Apart from the users, the ICT department and the outsourcing company, there are other stakeholders in the total solution. Hardware vendors are typical of those. Hardware vendors should be involved in the drafting of service level agreements and having meetings with such vendors during the SLA process is beneficial. It is almost inevitable to have service SLAs with such vendors as well. They need to understand their contribution to the success of the SLA between the client organisation and the software application system outsourcing company. The extent of their possible impact on the ability of others to meet SLA requirements needs to be documented.

Develop Reporting Mechanisms

Reporting on SLAs should be easy to understand but still be comprehensive. The information should be accurate and honestly portrayed. Should service levels not be met, reasons for that should be easily derived from the SLA reports. In an event of poor delivery, specific reports should state the plans and actions that will be implemented to rectify such failure.

Write Enforceable Contracts

The ICT manager and outsourcing company should take time to plan and review the SLA contents and to ensure that the objectives are actually achievable. The SLA should not be agreed on if the expectations cannot be met. Organisations should be cautious not to set standards that are too low or goals that are not challenging. Should the outsourcer and ICT agree on service levels that are unrealistically low, they will lose credibility with the users, i.e. users might develop a perception that ICT is not bargaining on their behalf and is trying to protect the outsourcing company from delivering service of a required standard. Goals for service levels should therefore be stringent but achievable and practical.

2.3.12 SLA Reporting

There are numerous reasons that emphasise the importance of accurate reporting on SLAs. Luevano argues that if the outsourcer reports that it is meeting service levels while the users suspects false reporting, the outsourcer will lose credibility. He states that "it takes months and sometimes years to re-establish credibility lost because of misrepresentation of facts". A loss in credibility between users and the outsourcing company can result in broken relationships. (Luevano: 1991:130)

Lacity and Hirschheim (1995:208) suggest that service level reports should state at least:

- The service level agreed on;
- The service performance for the current period;
- Exception reporting for missed measures; and
- A trend analysis of the performance from previous reporting periods.

They argue that customers are often even willing to pay for proper SLA reports.

2.3.13 Penalties and Rewards

Cunningham (2001:213) maintains that an SLA should have 'teeth' to create penalties for the provider, so that they actually cannot 'afford' to let down on the service they are contracted to deliver. However he also argues that the SLA is not a weapon, but a measurement device and has the purpose of supporting the business. (Cunningham, 2001:242). Larson (1998:128) notes that the SLA should be used to condition the service provider's desired behaviour rather than to trigger penalties.

From this the conclusion can be drawn that penalties (and rewards) are not the primary objective of SLAs and that focussing too heavily on these aspects may in fact result in unproductive relationships with accompanying unacceptable service levels.

2.4 Software Application Systems

2.4.1 Measurement of Services

Being intangible, software applications are extremely difficult to measure, and so is the work or effort that goes into the development and maintenance of such a system. Very often, conflicts arise between the service providing organisation and the client due to perceptions of the client that the agreed level of service had not been delivered by the outsourcing company. Lacity and Hirschheim (1995:204) support this and argue that services regarding applications development and support are "difficult to measure since the activities are labour intensive". A result of previous research done by them indicates that participants decided to use headcount as a unit of measure. SLAs would then state that a client is entitled to a certain number of computer analysts and programmers, which is converted into 'Full Time Equivalents (FTEs)', i.e. a possible combination of resources that will make up the equivalent of the agreed number of resources, should they work on this account full time.

Four problems were identified in their research:

1. The service provider may eliminate resources and make the remaining employees work harder. This is theoretically in the spirit of the SLA, but excessive hours may result in fatigue with a lowered standard of deliverables;
2. The service provider may move the better quality resources of the specific client account to attract possible new business. The remaining resource pool then becomes diluted, with the follow-on effect that client-specific knowledge is lost.
3. The client has to 'pay' for non-productive hours, such as employee training, meetings and administrative tasks.
4. Lastly, the FTE does not measure quality of output. The SLA might state that the client pays for additional hours, but these may be caused by a number of reasons, e.g. poorly trained staff and bad supervision.

Lacity and Hirschheim (1995:205) clearly state that not all of the problems above should not be blamed on the outsourcing company, and that organisations that are not willing to face such problems should not agree to software application development outsourcing.

2.4.2 The Systems Development Life Cycle (SDLC)

The Systems Development Life Cycle is a common methodology followed when software application systems are developed. Laudon and Laudon (2002:321) mention that it has a very formal division of labour between end users (client) and information specialists (outsourcing company). End users are responsible for providing the information requirements and reviewing and assessing the work/deliverables of the information specialists.

2.4.2.1 SDLC Stages

Project Definition

The first stage in the SDLC is purely a client organisation concern. This is the stage whereby the client organisation recognises a need or problem, which might be solved by the development of a new system.

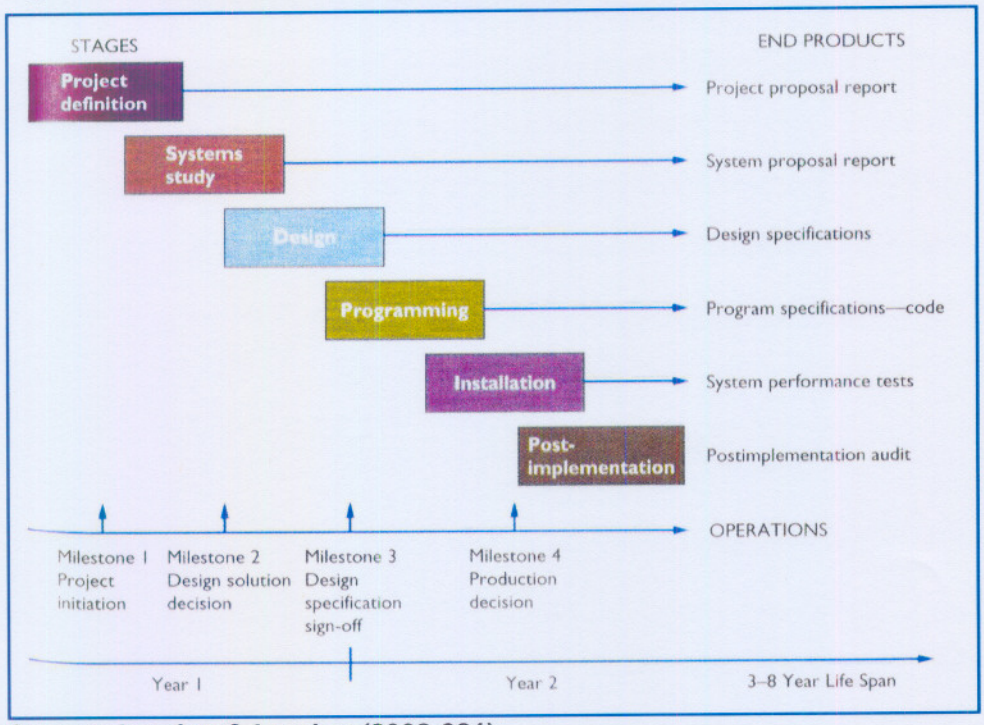
Systems Study

The Systems Study phase is also a client organisation function. The objectives and required outcomes of a possible solution are investigated and analysed in detail. The end users play a vital role during this stage.

Design

This is the first stage during which the information specialists become involved (outsourcing company). The exact designs for a possible solution are generated and documented, based on the requirements set in the previous stage.

Figure 3: The Systems Development Life Cycle Stages



Source: Laudon & Laudon (2002:321)

Programming

During this stage, the actual software application is developed. The main activity is the actual writing of program code that will result in a software application system. This is purely an information specialist function.

Installation

The installation stage complies of activities such as testing of new functionality, the conversion of data from a previous system to the new system, and training of end users. (Outsourcing organisation function)

Post-Implementation

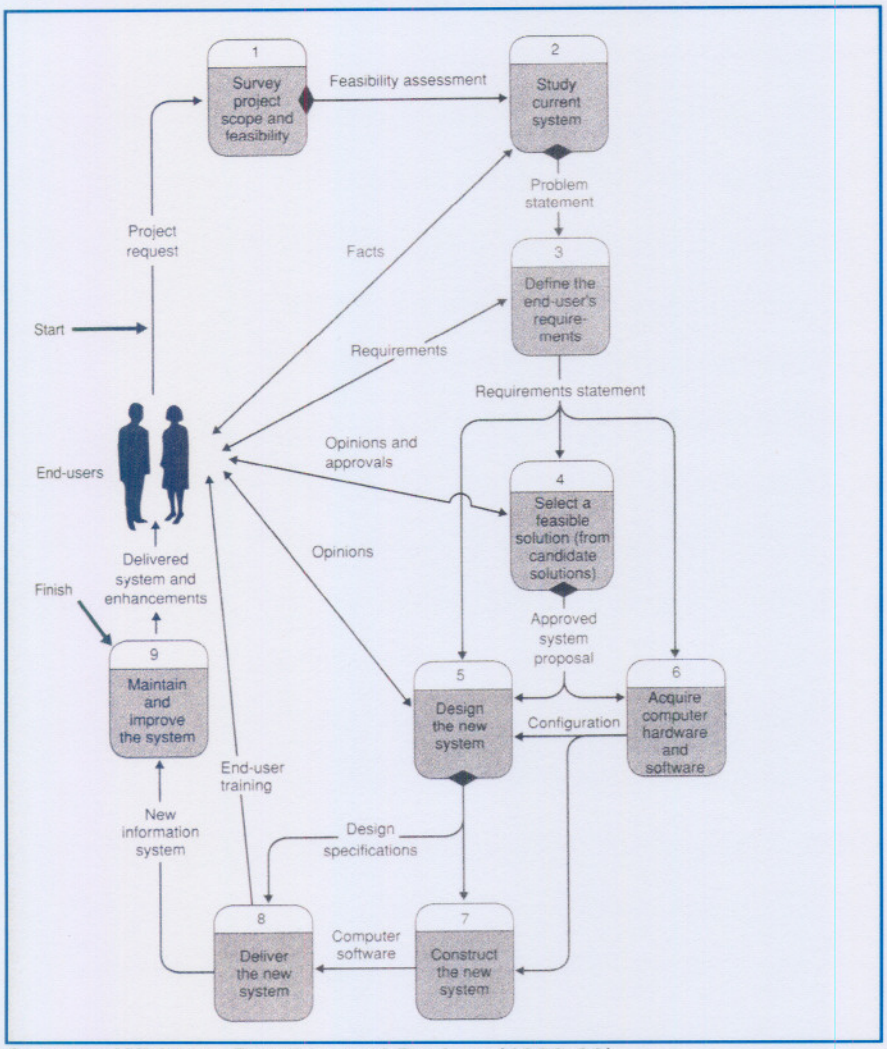
During this last stage, the system will be evaluated and measured against the initial requirements for completeness and accuracy, and to see if the system has met its set objectives. It is during this last stage, which might have a life span of three to eight years, that the Service Level Agreement will play a vital role.

2.4.2.2 The SDLC and the End-User

As previously discussed, the end user of the system is probably the most important entity when it comes to the success of SLAs. In the end, it is the users of the system that actually 'feel' the service provided by the outsourcing company. Whitten *et al.* (1989:81) maintain that there are six principles that should be adhered to when developing systems and that should be kept in mind when dealing with the SDLC. The first of the six principles reads: "The System is for the End-User". They argue that very often, the developers of a system fatally refer to the system as 'my system'. System developers work very hard to create technologically impressive solutions, but those systems very often backfire because they don't address the real needs of the users. Instead, they might introduce new business or technical dilemmas.

Figure 4 on the following page clearly illustrates the importance of the end-user during the entire systems development life cycle. Throughout the entire SDLC, the end-users are involved to some extent or another. This highlights the similarity between the end-user importance in SLAs and the end-user importance in the SDLC.

Figure 4: The SDLC and the End-User



Source: Whitten, Bentley and Barlow (1989:89)

2.5 Chapter Summary

The literature studied in this chapter revealed thorough details regarding the respective components of the subject under study. Outsourcing has been discussed and indications are there that outsourcing, as an option will continue to grow. With the increasing emphasis on the service aspect as part of the whole delivery package, the importance of SLAs will increase.

One shortcoming of the literature investigated, is a model for addressing the issue of SLAs appropriately from the commencement of the systems development process. The

following chapter contains details regarding a suggested model to be applied for integrating SLAs as part of the SDLC.

3 CHAPTER 3: A MODEL FOR INTEGRATING SLAS TO THE SYSTEMS DEVELOPMENT LIFE CYCLE

3.1 Introduction

Service Level Agreements are used in many forms in multiple disciplines throughout the business. They have much in common and surely the single aim of ensuring that the services that are rendered by the service provider adhere to the levels that were agreed on by the parties involved.

Depending on the nature of the services that are being rendered, the complexity with regards to the implementation and management of SLAs varies considerably. Within the ICT arena, SLAs that manage the service delivery of computer hardware are far easier to interpret and manage, due to the tangible nature of computer hardware. In contrast, (and still within the ICT domain) the development of computer software application systems as a service is an extremely difficult issue. This is purely because of the intangible nature of software applications and the lack of proper methods for quantifying and measuring such services. Köppel *et al.* admit that the typical questions to be addressed by SLAs (see section 2.3.1 above) are difficult to answer due to the extreme complexity and diversity of software applications.

This chapter will provide a model to be used as a tool for assisting in the assessment of SLAs specifically within the framework of outsourced software application development. Within this context and within the total outsourcing engagement, there are two distinct time-periods involved in delivering an outsourced software application, i.e.:

- The development and implementation of the software application as a product (Systems Development Life Cycle); and
- The post-implementation service and support of the developed software application in the production environment.

Köppel *et al.* refer to these two phases as the *Introduction- and Production phase* and comment as follows:

> *The lifecycle of an application is roughly characterised by the introduction phase and the production phase. The introduction phase represents the complete setup of an application system for its required use. The production phase follows the introduction phase and represents the final usage of the application in the productive environment.*

The aim of the proposed model is to cover both these components of the total service delivery as part of the SLA.

3.2 Outsourcing and SLAs within the Context of this Study

Within the context of this study, outsourcing will be referred to in the form as discussed in section 2.2.3 above, i.e. the outsourcing of the complete systems development function as well as the outsourcing of the ongoing maintenance after the commissioning of the system in an operational environment.

The SLA aspects that are addressed within this study exclude the SLA that might be in effect during the period of systems development. It is assumed that the Project Management function fulfils that requirement. The SLA that is being studied is the SLA that will be in force once the systems development is completed and the application is installed in the live production environment. Certain elements of the SLA will be addressed as part of the SDLC and should be seen as the 'construction' phase of the SLA that will eventually commence on completion of the development.

3.3 The SLA and the SDLC

Introducing the SLA during the Systems Development Life Cycle has some significant benefits and advantages for both the client organisation and the outsourcing company. The areas of the SLA that will be involved at this time are the areas that specifically pertain to the

level of performance of the application, and not the general service and support component that is expected from the outsourcing company.

From Figure 3 above it is apparent that the aspects of measurement specified in the SLA, become known after the second phase, i.e. 'Systems Study'. As described, the exact objectives to be attained by the development of the new system are specified during this phase. It is therefore possible to start compiling the SLA on conclusion of this phase.

There is an enormous benefit in having the SLA requirements available before the 'Design' phase is started. It allows for the outsourcing company to ensure that the design is in alignment with the user requirements and expectations. The fact that the 'Systems Study' phase is owned by the end-users ensures that their requirements and expectations are all incorporated in the 'System proposal report', which is the end product of that phase. This will assist in closing down the expectations gap.

On completion of the systems development, i.e. the programming of the system (Phase 4), the system will be introduced to the end users for the first time during Phase 5, the 'Installation'. A very prominent activity during the implementation phase is system testing. This allows the users the first opportunity to assess and measure the system performance against the SLA requirements set in the 'Systems study' phase. There is a huge advantage in the fact that during the systems post-development testing, the SLA requirements are considered and used as a benchmark for performance and delivery. The fact that the system is not yet signed-off or formally considered complete, still allows for changes to the system to be made. Should there be any indication of functionality of the system that in future might not meet the SLA requirements, enhancements to the system can still be made. The outsourcing company thus has the opportunity to address any risks for future breach of SLA. This will also avoid the situation where the outsourcing company will have to make system changes only after the system has done live, with higher associated costs and effort. Making changes to the system

during the implementation phase is significantly cheaper and more effective than addressing these after implementation.

3.4 Contents of an Outsource Software Application-specific SLA

3.4.1 General Elements

This section is not specific to the Software Application but deals with the contractual aspects of the agreement. Elements that should be present are:

- Roles and Responsibilities
- Period of the Agreement
- Term of Notice
- Contract Extension Details
- Effective Date
- Costs
- Privacy Clause and IP ownership issues
- Possible Expansions and Associated Cost
- Review Period and Process
- Change Management of the SLA
- Contract Termination Aspects
- Rewards and Penalties

3.4.2 Application Specific Elements

This section will be fundamentally different from application to application. The elements listed within this section mainly deal with the technical and functional aspects of the specific system.

3.4.2.1 System Environment

This section describes the technical environment of the system and typical elements include:

- Number and description of systems
- Specific application version numbers
- Database details
- Operating systems

- Application functional modules
- System printing facilities
- Peripheral devices
- System interfaces
- Development-, test- and production environment details

3.4.2.2 Application Maintenance & Support Management

This section contains the details pertaining to the management of ongoing maintenance of the system. This is a very important aspect of the SLA and is usually a requirement for external and internal auditing. The focus is on the policies and procedures that surround the support of the system.

The following elements must be included:

- Backup and recovery strategies
- Archiving strategy
- Test and authorisation strategies
- Change request management
- Issue prioritisation methodology and escalation procedures
- Code-change update procedure
- Software version control management
- System maintenance downtime
- The support structure and related communications channels
- Software upgrade management

3.4.2.3 Application Performance Measurement (Service Elements)

This is probably the most difficult section of the SLA in terms of definition and agreement. The actual system processes and the minimum performance and behaviour of each process / function need to be defined and agreed on. It is also the most important section of the SLA and bears the most significance with regards to the success of the SLA and client / service provider relationship. Detail in this section should not be limited but rather overstated to ensure a clear understanding by both parties of exactly what is being agreed on.

Larson (1998:129) suggests that each service element must have the following four components:

Service Description

This is a complete definition of the service boundaries from the user's perspective;

Constraints

This component defines:

- The rules and regulations within which the service will be delivered and achieved;
- The level of demand / activity beyond which the defined performance will not be assured; and
- Any requirement placed by the service provider on the users, which, if not fulfilled, will mean that the service provider may not achieve the agreed service levels.

Performance measures

This section defines how the specific service element will be measured. Typical performance measures are: (i) availability; (ii) reliability; (iii) response; and (iv) serviceability. Note that although user satisfaction can form part of the measurement of the overall service delivery (as described in section 2.3.9.1) a single service element cannot be tested against such a subjective measure.

Pricing

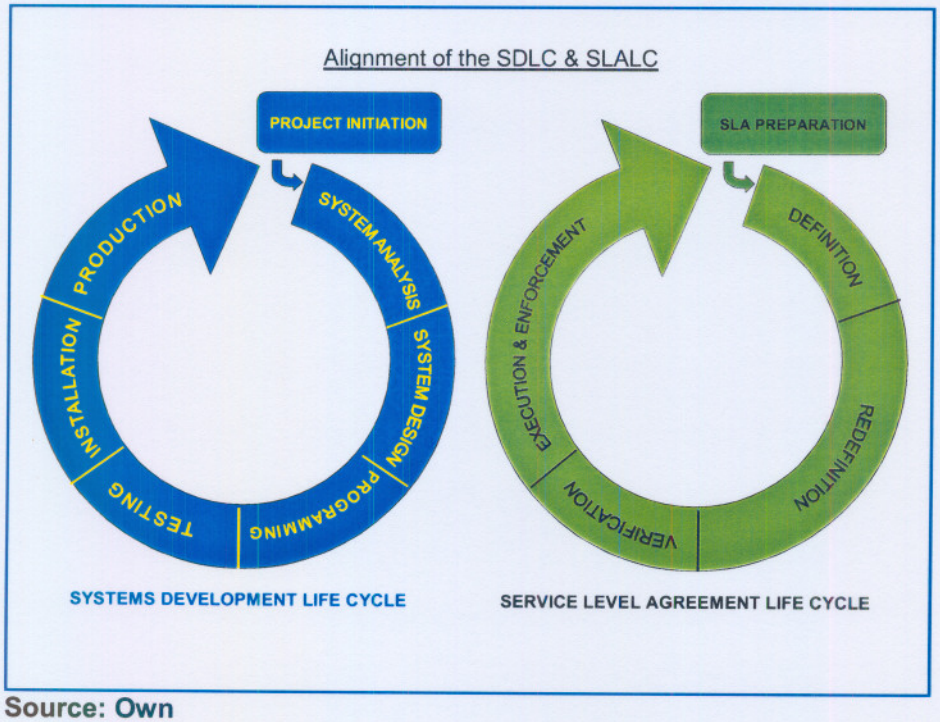
This is a discretionary component and defines the cost to the customer for the specific service element.

3.5 Aligning the SLA and the SDLC

When a software application is developed and implemented, as described in the SDLC (see section 2.4.2) there is an opportunity to align the SLA development and software application implementation as concurrent activities. Both these developments can be broken down into phases. Aligning the life cycles of both processes offers the benefit of assuring that once the developed application is signed-off and implemented into the operational environment, the SLA has been developed, refined and 'tested' at the same time. It also offers the advantage to the outsourcing company of ensuring that the development that is done adheres to the

minimum performance requirements of the SLA. Having the SLA requirements available as early as the design phase of the SDLC, allows the system design engineers to design solutions that fundamentally comply with the expectations of the end-users as defined during the early stages of the systems study phase.

Figure 5: SDLC & SLALC Alignment



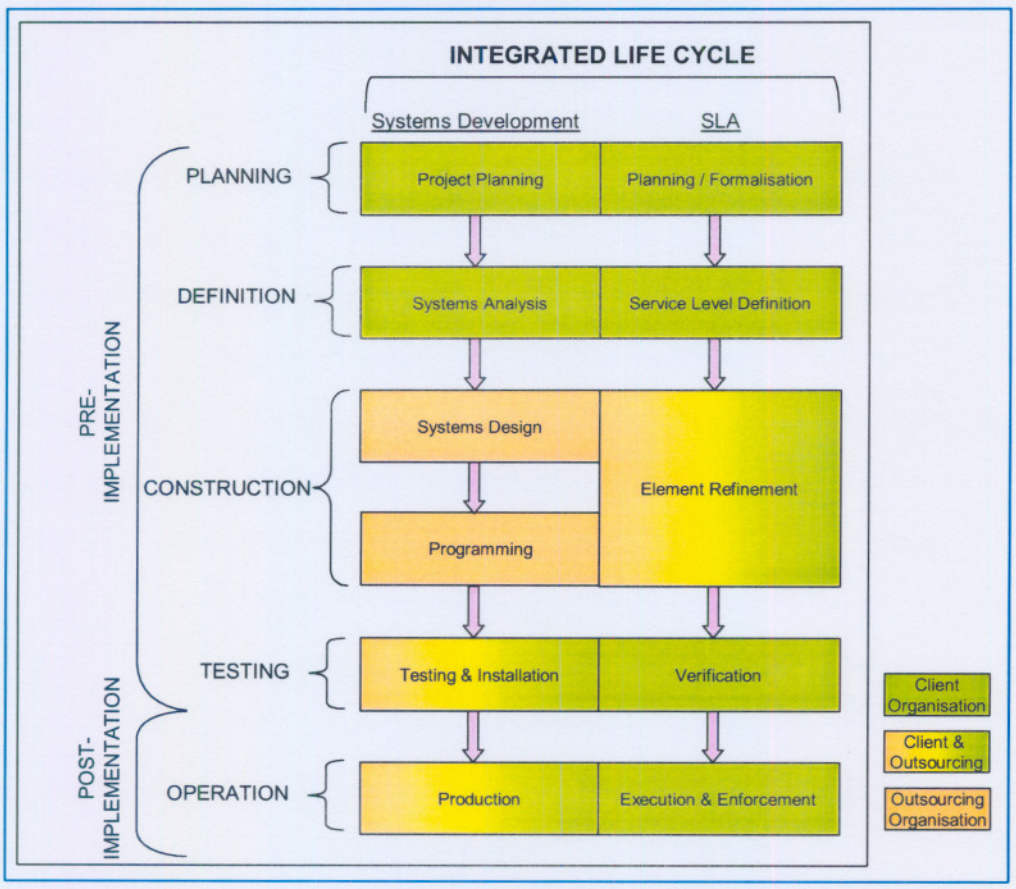
Source: Own

As shown in Figure 5 there are definite similarities in the life cycles for Systems Development and the Service Level Agreement. Both these life cycles start off with an *initiation* or *preparation* phase. This is a once-off process and deals mainly with the planning activity. The second part of the life cycle deals with *definitions / specifications*. Part three allows for both the application and the SLA-definitions to be *tested* and *verified*. The fourth and last part of the life cycles involves the execution of both the developed system and the SLA. These similarities make it possible to integrate the processes by linking activities in each phase of one process (Life Cycle) to activities in the corresponding phase of the other process.

3.6 Integrating the Systems Development - and SLA Life Cycles

Having a significant degree of alignment between the Systems Development and SLA Life Cycles allows for the integration of the two processes. As mentioned, activities from both processes can be tied successfully to form a logical integrated process.

Figure 6: An SDLC & SLA Integration Model



Source: Own

3.6.1 The Integrated Life Cycle

Combination of both life cycles results in a single integrated process containing the following stages:

Table 4: Integrated Life Cycle Activites

PHASE	STAGE		SYSTEMS DEVELOPMENT	SERVICE LEVEL AGREEMENT	STAKEHOLDER
PRE-IMPLEMENTATION	PLANNING		Activity: • Initiate Project • Define Scope • Set Project Goals • Determine Feasibility Output: • Feasibility Assessment • Project Definition Report	Activity: • Coordinate Resources for SLA Management • Identify All Affected Users • Draft Legal & General Components Output: • Draft SLA (Partial)	Client: • Management • ICT • Lawyers
	DEFINITION		Activity: • Analyse Business Application • Define Current Problems • Define Opportunities • Define End-User Requirements Output: • Functional Specification	Activity: • Define Application Specific Elements, i.e. (i) System Environment; (ii) Maintenance & Support Management; and (iii) Application & Support Measurement Output: • Draft SLA (Complete)	Client: • ICT • End-Users
	CONSTRUCTION	DESIGN	Activity: • Design Database Structure • Design System Inputs, Processes & Outputs • Design Internal Controls Output: • Technical Specification	Activity: • Re-Define Application Specific Elements • Review & Fine-Tune SLA Elements	Client: • ICT Outsourcer: • Systems Analyst • Programmer
		PROGRAMMING	Activity: • Code Software Application • Debug Programs • Perform Module Testing Output: • Software Application System		

	TESTING	Activity: • Systems Integration Testing • Performance Testing • Train End-User • Deliver & Install System • Convert & Verify Data • Sign off Data Conversion • Compile Documentation Output: • User Accepted & Signed-off System • Documentation	Activity: • Verify SLA Elements • Endorse SLA Output: • Endorsed SLA	Client: • Management • End User • ICT Outsourcer: • Programmer • Systems Analyst
POST-IMPLEMENTATION	OPERATION	Activity: • Maintain System • Improve Application • Update Documentation • Re-Train Users Output: • Efficient & Effective Software Application System	Activity: • Review SLA Output: • Up-to-Date SLA	Client: • End User • ICT • Lawyers Outsourcer: • Programmer • Systems Analyst

Source: Own

3.6.2 Integrated Activities

Table 4 demonstrates the integration of the life cycles of the Systems Development and SLA processes with regards to the various phases and stages during the complete process. To obtain a real benefit, one must ensure that the activities performed during the life cycles are also integrated. The following integration in that regard is suggested:

3.6.2.1 Planning

As can be seen from the table, the activities in the first stage mainly focus around the planning of resources and the scope of the project. Activities that should be amalgamated during this stage are:

- Include the scope of the SLA in the overall project scope.
- Identify ownership of the SLA as part of the project definition report.

- Identify and allocate the resources required for the drafting and the management of the SLA together with the overall project resources and their responsibilities.

3.6.2.2 Definition

This stage in the process probably has the biggest scope for integration between the SDLC and SLALC. When the end-user requirements are specified in the functional specification, an integral part of that specification must be the performance requirements as would be stated in the SLA. Integrated activities should include:

- Specifying the SLA performance and service level requirements as part of the functional specification for each component of the system. This will include user interfaces, system processes, reports, etc. Performance metrics that should be explicitly defined are (see section: 2.3.9.1)
  (i) Response times;
  (ii) Throughput;
  (iii) Reliability & Accuracy; and
  (iv) Serviceability.

  Although some of these metrics are more specifically related to the entire system and not to a smaller function within the system, they should be adhered to as far as possible.
- As an extension of the standard SLA performance measures, users should specify other performance related measures, i.e. statistical operational-like information, e.g. number of records processed during a certain period (week, month, etc.). Although this is not a direct SLA element, it will benefit both the client and the outsourcing organisation in tracking down performance problems. It might also be an advantage to the outsourcing company as part of its constraints for delivering the agreed service levels and would clearly indicate systems 'abuse', if any.

3.6.2.3 Design

This is the stage that will take the greatest advantage of integrating the SLA to the SDLC. During systems design, the systems analyst is relying absolutely on the information made available to him during the previous stage is the SDLC. One of the biggest problems in introducing the SLA only on completion of the systems development, is that very often the SLA requirements demand functionality or performance from the system that is not embedded in the design of the system. This results in a real dilemma because adherence to such SLA requirements would necessitate a re-design of certain core and embedded structures of the system.

Knowing the SLA demands before the commencement of the design stage allows the systems analyst to ensure that the core functions and design features of the system will support the SLA requirements. SLA integrated activities should include:

- Design embedded system core structure in alignment with SLA performance requirements.
- Design specific system functionality that allows for system-inherent measurement features to 'automatically' measure SLA performance requirements. This would allow for real-time and continuous monitoring of service levels. Having such proper features embedded in the system would contribute towards clear and objective measurement methods and results.
- Negotiate any unachievable or problematic SLA requirements with end-users and agree on alternatives or re-definition of such SLA requirements. This would avoid inadequate system deliverables once the system is in full operation.

3.6.2.4 Programming

This stage will offer the first opportunity to measure the success or achievability of the SLA performance requirements. On completion of programming of any system function, the programmer should perform initial tests to verify correct behaviour of the program. Should there be any inefficiency in the design, it would become noticeable at this early stage. Possible

solutions to such inefficiencies include a re-design or alternative programming structures. Should these alternatives not solve the problem, the client should be called upon to resolve the matter in collaboration. Trade-offs can be made to attain a win-win solution.

3.6.2.5 Testing

The end-users are the main players during this stage. The software application has now been developed and is awaiting acceptance and sign-off from the client. This will also be the acid test for the SLA service level requirements with regards to the system performance levels.

This stage allows for the first of the typical periodic SLA service level measurements. The system should now be geared towards the exact performance level requirements defined in the SLA. It not only gives the users the opportunity to assess the system performance, but also gives the outsourcing company the assurance that the system will be signed off, not only against the functionality requirements specification but also against the SLA requirements. The performance measurement features embedded in the system can now also be verified for accuracy and effectiveness.

The actual sign-off documentation should include the stated SLA requirements. In the exceptional event of system functionality still under-performing, the outsourcing organisation still has the opportunity to rectify the cause of such under-performance.

3.6.2.6 Operation

At this point the system is signed off, implemented, documented and the end-users trained. The SLA will now be in force and the outsourcing company exposed to the legal implications of the agreement.

An important aspect that now comes into play for the first time since the initial stage of the total life cycle is the service component of the SLA. These are the application maintenance and support management services (see section 3.4.2.2) offered by the outsourcing company. Thus

far only the application performance elements of the SLA have been put to the test and verified. Only once the software application system is in a live operations environment will the outsourcing company have the opportunity to display its commitments and abilities with regards to the associated support services.

3.6.3 Impact of Software Application Upgrades on SLA

In a dynamic operational environment, there will be a continuous demand for changes and enhancements to the software application. These might be due to changing business processes, changing legislation or other factors. Any such changes have to be carefully planned and co-ordinated to determine their impact on the SLA. It might be inevitable to redefine service elements that refer to system functions that no longer exist, or new elements might have to be added as a result of new functionality.

The 'Application Maintenance & Support Management' section of the SLA (see section 3.4.2.2) will dictate how software upgrades will be handled and managed. Part of the logistical arrangements might require the system to be unavailable to end-users and hence will affect the performance measurement metrics by which the outsourcing company is assessed.

What is of extreme importance is that, once system changes are requested, a reiteration of the integrated life cycle will take place. It is critical that the SLA activities related to the systems development activities are performed to ensure that the SLA remains in alignment with the operational and business requirements, and the related solutions. Both the client organisation and the outsourcing company should be aware of the implications of engaging in the entire process. Not only does it require resources from the outsourcing company to design and develop new functions or applications, but it would also require the client organisation to engage in the testing and acceptance sign-off process. The management and re-alignment of the SLA also requires attention and allocated resource time from the client organisation.

3.6.4 Critical Success Factors of the Integrated Model

For the *SLA and SDLC Integrated Model* to be successful, certain key factors need to be present. These features underline the uniqueness of the model and form the foundation for the successful assimilation of the two life cycles. From the various stages and related activities of the integrated model, the following factors are deemed critical to the overall success:

PRE-IMPLEMENTATION

3.6.4.1 Planning Stage

Critical factors that should be present during this initial stage include the following:

- Involvement and commitment from the client executive level to the allocation and dedication of resources for the duration of the entire project. If not part of the executive team himself, the *Steward* should be granted full authority and empowerment by the executive team. The SLA Steward should form an integral part of the overall project resource pool and participate in all relevant project meetings and decision-making forums. (See section 2.3.4.1)
- Selected representatives from the user corps should be identified at this kick-off stage. As with the Steward, these users must have clear objectives and directives as to what is expected from them. The typical profile of such users should identify their ability and experience to simultaneously fulfil the roles of SLA 'owners' as well as main players in the development life cycle of the new software application system. As stated in section 2.3.4.3, these users should be empowered to define the application performance levels that will be included in the SLA, together with the authority to sign off the eventual system and confirm that the SLA performance metrics are met.

3.6.4.2 Definition Stage

The critical aspects during this stage are:

- Additional to the standard application behaviour requirements, the SLA performance requirements should be included as an actual part of the functional specifications documentation for each application requirement.
- Together with the SLA requirements, the end-users must define exactly the measurement of the required SLA performance and what they would require from the application to enable them to test the SLA performance requirements, i.e. a suggested test plan.
- It is important that the necessary authorities from the client side are involved in these SLA-orientated specifications, as they will become legally enforcable due to the nature of the agreement.

3.6.4.3 Design Stage

In contrast with the previous two stages, which were primarily client responsibilities, this is the first of the outsourcing company responsibility stages. Critical features that should be pre-set in this stage include the following:

- The embedded application design structure and fundamentals should be aligned with the SLA performance requirements to avoid re-design of core features at a later stage for adherence to SLA performance requirements.
- The design should include measurement functions to allow for the regular independent measurement of application performance. These measurement functions should be embedded system functions.
- The database design should allow for the storage of performance statistical values, typically a throughput history. This would allow for the calculation of performance trends.

3.6.4.4 Testing Stage

To ensure that the SLA is effectively integrated with the systems development, the testing stage is of critical importance. Specific issues that need to be successful include:

- The application embedded SLA performance requirements must be tested in accordance to the previously defined test plan, as specified by the end users.
- Specific effort should be put into high volume testing to ensure that the system will perform well under normal production conditions.

POST-IMPLEMENTATION

3.6.4.5 Agreement

Once the system is installed and performing in a production environment it is necessary to review the SLA on a regular basis. Is it critical to:

- Abide by the integrated life cycle whenever either of the SLA- or the system contents is changed. It is absolutely vital to keep these components synchronised.
- Conduct regular performance reviews based on the set performance metrics. This will be a requirement from the general SLA terms and will have to be reported on.
- Utilise the Quantitative Analysis features developed as part of the application to determine performance trends and act as a possible early-warning system for possible future performance problems due to increases in business process volumes.

3.7 Enhanced service levels during peak periods

In many organisations there are specific periods during the months, e.g. month-end, where a higher level of service is required due to peak volumes of data processing. In many investment-like organisations money needs to be invested within a certain number of days after receipt, which is typically at month-end. The SLA should provide for higher service levels and quicker response times during these periods, with an associated premium in the costs.

An alternative could be an increased service level arrangement linked to a certain system function. This is similar to the first alternative but contains the risk that the outsourcing company

cannot anticipate the exact time when such processes will be run and hence might be caught unguarded to deliver the raised service levels when needed.

3.8 Chapter Summary

During this chapter, the development of a proposed model for integrating the SLA requirements to the systems development life cycle was discussed. All aspects of the SLA life cycle were analysed and compared to those related to the SDLC. Subsequently, a model was developed, containing combined elements from both the SLA and SDLC life cycles.

The model was based on certain critical success factors, that without, the model's effectiveness would be in jeopardy. These critical success factors were translated into a questionnaire to be used for gathering survey data. The next chapter contains details regarding the questionnaire and the subsequent results obtained from the survey.

4 CHAPTER 4: EMPIRICAL STUDY

4.1 Research Alternatives

At this point the study offers a choice of empirical research applications, i.e.:

(1) Conduct a survey of the success of SLAs for Outsourced Software Application Systems.

This alternative would clarify the general speculations regarding the success of SLAs within the stated context, but inherits various difficulties, amongst which the following: A suitable definition of *success* will have to be defined. Due to the numerous metrics of the qualification as well as the multiple components (with related contributions) that determine the success of SLAs, it might become a very subjective hypothesis.

(2) Test the success of the model in a practical environment over time, i.e. longitudinal study.

To conduct an actual implementation of the model in a real world environment will probably deliver the best proof as to whether the suggested model will in fact improve the successful application of SLAs within the Software Application Systems environment. Increased empirical success could be obtained by implementing the model at multiple software application systems development projects.

Due to the typically considerable time-lines of software development projects and the time constraints of this study, this would however not be an attainable alternative for empirical research.

(3) Verify the possible success of the model by obtaining and analysing various opinions from field experts.

Given the constraints of the previous two alternatives, this option provides a pragmatic opportunity for empirical research within the framework of this study and would further contribute to the value thereof.

Seeing that the *Critical Success Factors* of the suggested model have been defined, research could be done to establish whether or not those critical success factors are currently present in software development projects. In addition to that, research could be structured in a way to also determine whether or not field experts are of the opinion that the suggested model could in fact be successfully implemented in practice.

4.2 Data Collection

As one of the ten ground rules for good research, Denscombe (2002: 99) states that, as components of *Accuracy*, data must have the attributes of *Validity* and *Reliability*. He quotes Carmines and Zeller:

> *"reliability and especially validity are words that have a definite positive connotation. For anything to be characterised as reliable and valid is to be described in positive terms. . . If it is reliable and valid, then it has gone a long way toward gaining scientific acceptance."*

Within the context of research, he defines *Validity* and *Reliability* as follows:

- *Validity* concerns the accuracy of the questions asked, the data collected and the explanations offered. Generally it relates to the data and the analysis used in the research.
- *Reliability* relates to the methods of data collection and the concern that they should be consistent and not distort the finings. Generally it entails the evaluation of the methods and techniques used to collect the data.

Denzin and Lincoln (1994:353) suggest five methods and techniques for producing and analysing empirical material, i.e.:

- The Interview
- Observational Methods
- Reading Material and Records
- Visual Methods
- Personal Experience

They are of opinion that interviewing conveyed in a structured form is the method most commonly used.

Both Kreitner and Kinicki (2001:28 - 29) as well as Cummings and Worley (2005:116 – 122) list four major techniques for data collection that are more recently being used in the research field. As can be seen from Table 5, all four techniques have their advantages and disadvantages.

Table 5: A Comparison of Methods of Data Collection

METHOD	ADVANTAGES	POTENTIAL DISADVANTAGES
Questionnaires	• Responses can be quantified and easily summarised • Easy to use with large samples • Relatively inexpensive • Can obtain large volumes of data	• Non-empathy • Predetermined questions • Over-interpretation of data • Response bias
Interviews	• Adaptive – allows data collection on a range of possible subjects • Source of 'rich' data • Emphatic • Process of interviewing can build rapport	• Expense • Bias in interviewer responses • Coding and interpretation difficulties • Self-report bias
Observations	• Collects data on behaviour, rather than reports of behaviour • Real time, not retrospective • Adaptive	• Coding and interpretation difficulties • Sampling inconsistencies • Observer bias and questionable reliability • Expense
Unobtrusive Measures	• Non-reactive – no response bias • High face-validity • Easily quantified	• Access and retrieval difficulties • Validity concerns • Coding and interpretation difficulties

Source: Nadler (1997 cited by Cummings and Worley 2005:117)

For this study it was decided to primarily make use of a questionnaire for the following reasons:

- Inexpensive
- Research time constraints (Benefit of quick turn-around time of questionnaires)
- Quick data analysis
- Access to more field specialists (availability)

4.3 The Research Instrument

The questionnaire was compiled and made available to respondents on the Internet. This allowed all respondents to have easy access to the questionnaire and allowed them to complete the questionnaire on-line. The secondary benefit of a web-based questionnaire is that the results can be analysed and process more rapidly due to the fact that the results are already gathered and centralised, ready for processing. The questionnaire as displayed on the Internet can be seen in Figure 7.

A very specific group of people were individually identified as being experts in the field of systems development within the paradigm of outsourcing and post-implementation support of the delivered systems. The reason for not publishing the questionnaire open to an un-controlled audience is to eliminate the risk of getting valid and reliable data from pre-identified sources diluted with data of which the validity and reliability cannot be guaranteed. The invited respondents represented *eighteen* different organisations involved in outsourced systems development and support.

The questionnaire compiled contained eleven questions. The first set of questions, i.e. questions one through to six, was directly related to the critical success factors derived from the integrated model earlier (See Section 3.6.4 above). The aim of these questions was purely to measure the current state of affairs regarding the use of SLA within the systems development paradigm. This was highlighted to respondents within the introduction to the questionnaire.

The second section (Question 7) of the questionnaire allowed the experts to state their opinion on the importance of the critical success factors embedded in the model. As opposed to the first section of the questionnaire (Questions 1 to 6), which assessed the current situation, the purpose of the second part, where respondents rate the importance of the critical success factors, was to assess the possible success or acceptance of the suggested model based on expert opinion.

The next part of the questionnaire (Question 8) contained questions pertaining to SLAs in general. Although this set of questions offered pre-defined answers, the objective was to acquire a rating of agreement or disagreement with the statements put forward. These statements represent current frequently-debated issues regarding the success of SLAs.

To prevent manipulated results, Question 9 was introduced to the questionnaire with the specific aim of allowing respondents to avow their unbiased opinion, i.e. not being bounded to the suggested model. In contrast to the first two sections (Questions 1 to 7), respondents were given the opportunity to freely comment on the suggested model of integrating the SLA Life Cycle with the SDLC. No limitation was given on the form or extent of the answer to this question.

To allow the author to analyse the data from either the Outsourcing Organisation – or the Client Organisation perspective, Question 10 required the respondents to state their current position within the organisation. Five pre-defined answers were presented as options as well as an option for *'Other'*, should the description of the respondent's position not be exactly the same as the options offered. All options were grouped into the two basic groups of Outsourcing Organisation – or the Client Organisation.

The last question (Question 11) was specifically included to consider the success / failure of the questionnaire of being a valid instrument to assess the status of SLAs within the software development environment. Only two Yes/No options were presented as alternatives.

Apart from the last three questions of general nature, all remaining questions were put as statements and allowed the respondents to contemplate their agreement / disagreement with the statements. Where applicable, an option was given to respondents to indicate that they were not sure about the current situation. This is a preventative measure to maintain valid and accurate data.

To strive towards well-thought-through responses, no middle value was allowed for questions one to six, i.e. an even number of choices were available, forcing experts to explicitly agree or disagree to some extent with the statements.

The questions were preceded by an introduction and invitation to the pre-identified respondents as shown in Figure 7 below.

Figure 7: Research Questionnaire

SLAs within the Software Application Development Paradigm

Dear Field Expert

From your own experience, you will probably agree with me that SLAs within the Software Development Paradigm can easily become conundrums. I am currently doing research on this topic in an attempt to try and establish how SLAs are dealt with in this environment.

I would appreciate your input and opinion on this subject and would be greatful if you can complete the following questionnaire with alacrity. In return I am willing to share the outcome of my reasearh, should you be interested.

Please note that questions 1 to 6 assess your current situation.

Regards

Corné Botha

1) During the **Planning Phase** of the Systems Development Life Cycle (SDLC):

	Always	Sometimes	Seldom	Never	Don't Know
Client SLA requirements are known to the Systems Development organisation	○	○	○	○	○
Client resources are allocated for ownership of the SLA development process	○	○	○	○	○
Client & Systems Developer are collaboratively agreeing on SLA contents	○	○	○	○	○

2) The **User Functional Requirement Specification (UFRS)**, compiled during the **Analysis Phase** of the SDLC, includes:

	Always	Sometimes	Seldom	Never	Don't Know
SLA Application Performance Requirements	○	○	○	○	○
Metrics to measure the required Application Performance (as per SLA)	○	○	○	○	○
A formal Test Plan to be used for Application Performance testing (E.g. transactions processed per minute)	○	○	○	○	○

3) During the **Design Phase** of the SDLC:

	Always	Sometimes	Seldom	Never	Don't Know
The database design allows for the storage of Application Performance data, to be used for SLA reporting and Trend Analysis	○	○	○	○	○
The system's core design is aligned with SLA Application Performance requirements	○	○	○	○	○
The system's design includes embedded application functionality to measure application performance (E.g. displaying average transaction processing time on completion of a batch run)	○	○	○	○	○

4) During the **Coding Phase** of the SDLC, programmers develop:

	Always	Sometimes	Seldom	Never	Don't Know
Applications which meet SLA Application Performance requirements	○	○	○	○	○
Application performance measurement functionality as embedded features	○	○	○	○	○

5) During the **Test Phase** of the SDLC:

	Always	Sometimes	Seldom	Never	Don't Know
Testing is done according to the Test Plan specified by users as part of the UFRS	○	○	○	○	○
High-Volume testing is done to assess system performance under operational conditions	○	○	○	○	○

6) During the **Post Implementation Phase**:

	Always	Sometimes	Seldom	Never	Don't Know
SLAs and Systems are kept in alignment whenever changes are made to either one	○	○	○	○	○
Regular System Performance Reviews are done, based on SLA requirements	○	○	○	○	○
SLA Reporting is done and is based on stored Application Performance data	○	○	○	○	○
Penalties/Rewards are applied in cases of under- or over delivery	○	○	○	○	○

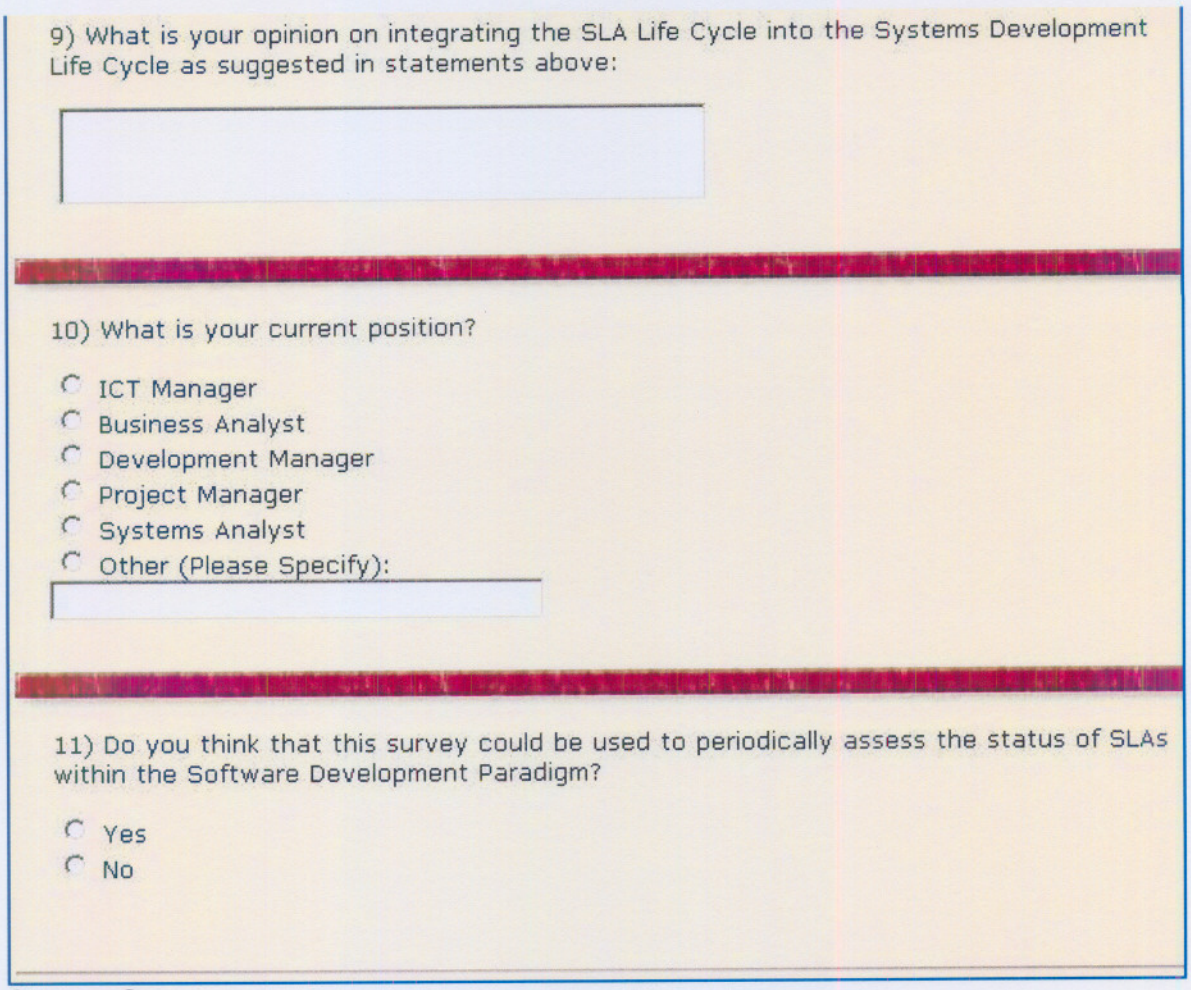
7) Rate the importance of the following: (Refer to the statements in Questions 1 to 6 above)

	High	Medium	Low
Client SLA requirements known in Planning Phase of SDLC	○	○	○
Client resources allocated for SLA ownership	○	○	○
Client & Systems Developer collaboratively agreeing on SLA contents	○	○	○
User Functional Requirement Specification (UFRS) to include Application Performance Requirements	○	○	○
UFRS to include metrics for measurement of Application Performance	○	○	○
UFRS to include formal test plan	○	○	○
Database design to cater for storage of application performance data	○	○	○

System's core design to be aligned with SLA requirements	○	○	○
System's design to include embedded functionality for application performance measurement	○	○	○
Application coded to meet to SLA performance requirements	○	○	○
Application performance measurement functionality coded as embedded features	○	○	○
System testing done according to specified test plan	○	○	○
High-volume testing to form part of user acceptance testing	○	○	○
SLAs and systems kept in alignment when changes made to either one	○	○	○
Regular systems performance reviews to be conducted	○	○	○
Regular SLA reporting to be done	○	○	○
Penalties & Rewards to be applied	○	○	○

8) Within the Software Development Paradigm, SLAs:

	Always	Sometimes	Seldom	Never
Contribute towards higher service levels	○	○	○	○
Offer financial benefits that exceed the additional resource cost and effort	○	○	○	○
Result in better client/service provider relationships	○	○	○	○



Source: Own

4.4 Research Results

4.4.1 Research Sample

Thirty individuals considered by the author as being qualified experts in the field under study were invited to complete the questionnaire. These individuals represent *twenty* different organisations currently involved in outsourced software application development, either being the outsourcing organisation providing software application systems, or alternatively being a client organisation making use of outsourced software application systems development and support. Two of the invitees were acting as an academic tutor and ICT consultant respectively.

From the thirty experts invited, **twenty-four**, representing **eighteen** different organisations, actually completed the survey,.

4.4.2 Results Analysis

As stated previously, Questions 1 to 6 are direct derivatives from the Critical Success Factors of the SLA and SDLC Integration model. The result of these first six questions (put as statements) represent the current status of SLAs within the various organisation targeted.

All answers with the option 'Don't Know' selected were eliminated from the results. This represented on 1.2% (5/408) of all answers provided where this option was allowed. Weightings as displayed in Table 6 were assigned to the remaining options. The assigned weightings imply that a neutral outcome (weight = 0) would be obtained if four respondents should each select a different option for a specific answer.

Table 6: Weightings: Questions 1 to 6

Option	Weight
Always	2
Sometimes	1
Seldom	-1
Never	-2

Source: Own

Quantitative Analysis was done on the associated weightings (described above) to determine the average, or *Arithmetic Mean, Median, Range,* as well as the *Standard Deviation.* Wisniewski defines the Standard Deviation as "typically how much the items in the data set differ from the mean value". He describes the Median as another measure of average, which represents the middle value of a data set. The main difference between the Arithmetic Mean and the Median is that the Median eliminates the value of extreme data items, included as part of the Arithmetic Mean. The Range is simply the difference between the maximum- and the minimum value in a data set and represents the variability of responses (Wisniewski, 2002:93-96). The *Frequency Distribution*, defined by Cummings and Worley (2005:127) as "a

graphical method for displaying data that shows the number of times a particular response was given", is shown by means of a graphical chart for each sub-statement of each question.

For Questions 1 to 6, questions were also analysed in whole, i.e. not only analysing sub-statements individually but also analysing all sub-statements as a combined result. This is made possible by the nature and structure of the questions, i.e. grouping sub-statements into questions that represent a certain phase of the SDLC. The analysis and subsequent conclusions from questions as a whole would therefore be interpreted within the context of the specific phase of the SDLC.

4.4.2.1 Question 1 (Planning Phase)

During the Planning Phase of the Systems Development Life Cycle (SDLC):

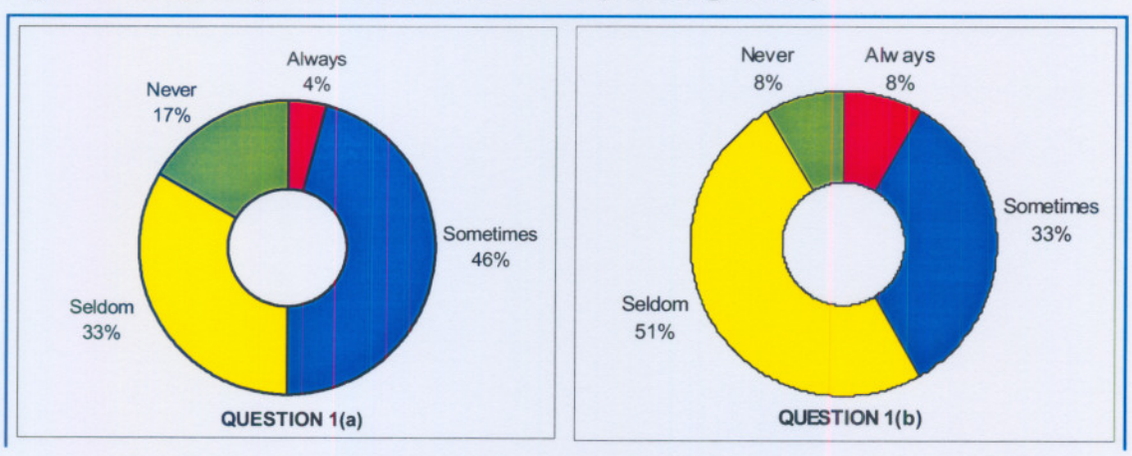
a. Client SLA requirements are known to the Systems Development organisation

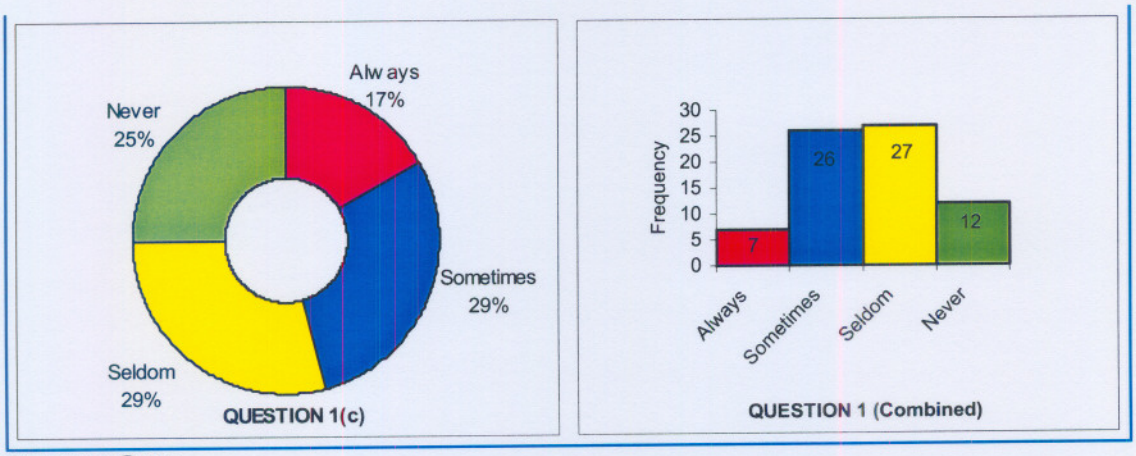
b. Client resources are allocated for ownership of the SLA development process

c. Client & Systems Developer are collaboratively agreeing on SLA contents

Quantitative Analysis

Figure 8: Frequency Distribution: Question 1 (Planning Phase)





Source: Own

Figure 8 shows a graphical presentation of the Frequency Distribution for each sub-statement for Question 1. As can be seen from the graph *'Question 1 (Combined)'*, the overall distribution for all combined responses for Question 1, is fairly *Normal*, i.e. a symmetrical spread across the X-Axis. The *Sometimes* and *Seldom* values are almost equal and represent 74% of all responses. The remaining responses lie primarily in the *Never* quadrant (17%) with a cumulative value of only 10% for *Always*. For Question 1(a) the positive/negative response split is equal, i.e. 50/50 (4% + 46%) for *Always* and *Sometimes* combined compared to 50% (17% + 33%) for *Never* and *Seldom* combined. For Question 1(b) the split is 41/59 positive/negative and 46/54 for Question 1(c). The combined positive/negative ratio for Question 1 is 46/54.

Table 7: Quantitative Analysis: Question 1 (Planning Phase)

	Q1 a	Q1 b	Q1 c	Q1 Combined
Arithmetic Mean	-0.1	-0.2	-0.2	-0.2
Median	0.0	-1.0	-1.0	-1.0
Standard Deviation	1.3	1.2	1.5	1.3
Range	4	4	4	4
Frequency Distribution				
Always	1	2	4	7
Sometimes	11	8	7	26
Seldom	8	12	7	27
Never	4	2	6	12

Source: Own

As can be seen from Table 7, the Arithmetic Mean for the three sub-statements for Question 1 are –0.1 for Q1a and –0.2 for both Q1b and Q1c., whereas the Median for Q1a is 0

with both Q1b and Q1c having a median of -1. Maximum variety in responses occurred reflected by the Range value of 4 for all sub-statements. Respondents reported fairly similarly around the Mean for Q1a and Q1b, indicated by the Standard Deviation of 1.3 and 1.2 for these cases. Somewhat more deviation was found for Q1c with the Standard Deviation of 1.1.

Conclusion

From the qualitative analysis it is clear that in the first phase of the SDLC, SLA integration is to a great extent ignored. Only one 1 out of 24 respondents confirmed that SLA requirements are *always* known during the planning phase, while 4 confirmed that they *never* know what the requirements are. Even with regards to collaborative agreement of SLA contents, the majority responded negatively to this key requirement.

4.4.2.2 Question 2 (Analysis Phase)

The User Functional Requirement Specification (UFRS), compiled during the Analysis Phase of the SDLC, includes:

a. SLA Application Performance Requirements
b. Metrics to measure the required Application Performance (as per SLA)
c. A formal Test Plan to be used for Application Performance testing.

Question 2 is once again directly extracted from the critical success factors related to the Analysis Phase of the integrated SLA and SDLC model and deals exclusively with the User Functional Requirement Specification (document) compiled by the users.

Quantitative Analysis

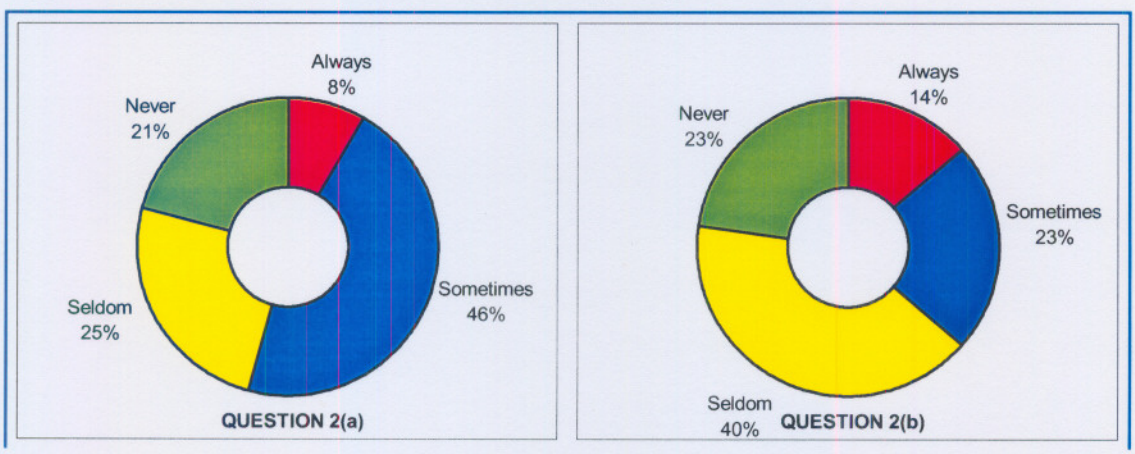
Table 8: Quantitative Analysis: Question 2 (Analysis Phase)

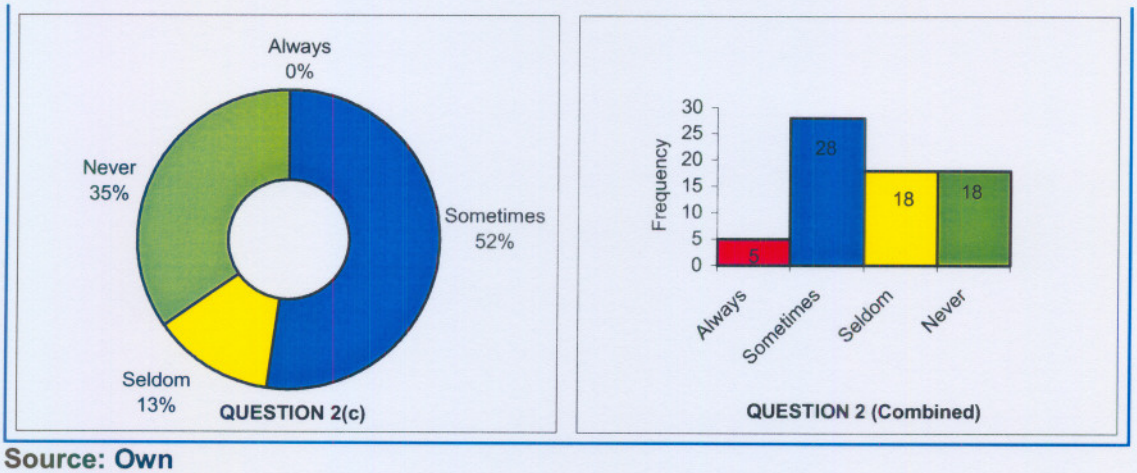
	Q2 a	Q2 b	Q2 c	Q2 Combined
Arithmetic Mean	0.0	-0.4	-0.3	-0.2
Median	1.0	-1.0	1.0	-1.0
Standard Deviation	1.4	1.4	1.4	1.4
Range	4	4	3	4
Frequency Distribution				
Always	2	3	0	5
Sometimes	11	5	12	28
Seldom	6	9	3	18
Never	5	5	8	18

Source: Own

Table 8 shows the Arithmetic Mean for Question 2 sub-statements that fluctuates between 0 for Q2a and –0.4 for Q2b, with Medians of 1 for Q2a and Q2c and -1 for Q2b. Whilst Q2a and Q2b shows maximum variety in responses reflected by the Range value of 4, Q2c shows a range of only 3, indicating only three options were chosen for that sub-statement. Respondents reported fairly disparately around the Mean, indicated by a maximum Standard Deviation of 1.4 for all sub-statement as well as Question 2 combined.

Figure 9: Frequency Distribution: Question 2 (Analysis Phase)





Source: Own

Figure 9 shows a graphical presentation of the Frequency Distribution for each sub-statement for Question 2. As can be seen from the graph *'Question 2 (Combined)'*, the overall distribution for all combined responses for Question 2, shows no signs of a symmetrical spread across the X-Axis. The mid-values (*Sometimes* and *Seldom*) are significantly different and represent 64% of all responses. As with the *Seldom* option, *Never* makes up an equal value of 25% of all responses. The *Always* option is significantly lower at only 5%. For Question 2(a) the positive/negative response split is 54/46, i.e. 8% + 46% for *Always* and *Sometimes* combined compared 25% + 21% for *Never* and *Seldom* combined. For Question 2(b) the split is 36/64 positive/negative and 52/48 for Question 2(c). The positive/negative ratio for Question 2 combined is 48/52. The positive/negative split for both Q2a and Q2c is fairly marginal, whilst the 37/63 split for Q2b shows a significantly 2/3rd negative result.

Conclusion

This phase, which is primarily a client responsibility, shows no definite proof of SLA integration in *all* cases. Despite the fact that there are a significant number of cases where SLA integration is *sometimes* incorporated, the majority of respondents indicated that SLA requirements are *seldom* or *never* considered when user requirements specifications are drawn up.

4.4.2.3 Question 3 (Design Phase)

During the Design Phase of the SDLC:

a. The database design allows for the storage of the Application Performance data, to be used for SLA reporting and Trend Analysis
b. The system's core design is aligned with SLA Application Performance requirements
c. The system's design includes embedded application functionality to measure application performance (E.g. displaying average transaction processing time on completion of a batch run)

Question 3 represents the critical success factors from the Design Phase of the SDLC and exclusively involves the systems development organisation.

Quantitative Analysis

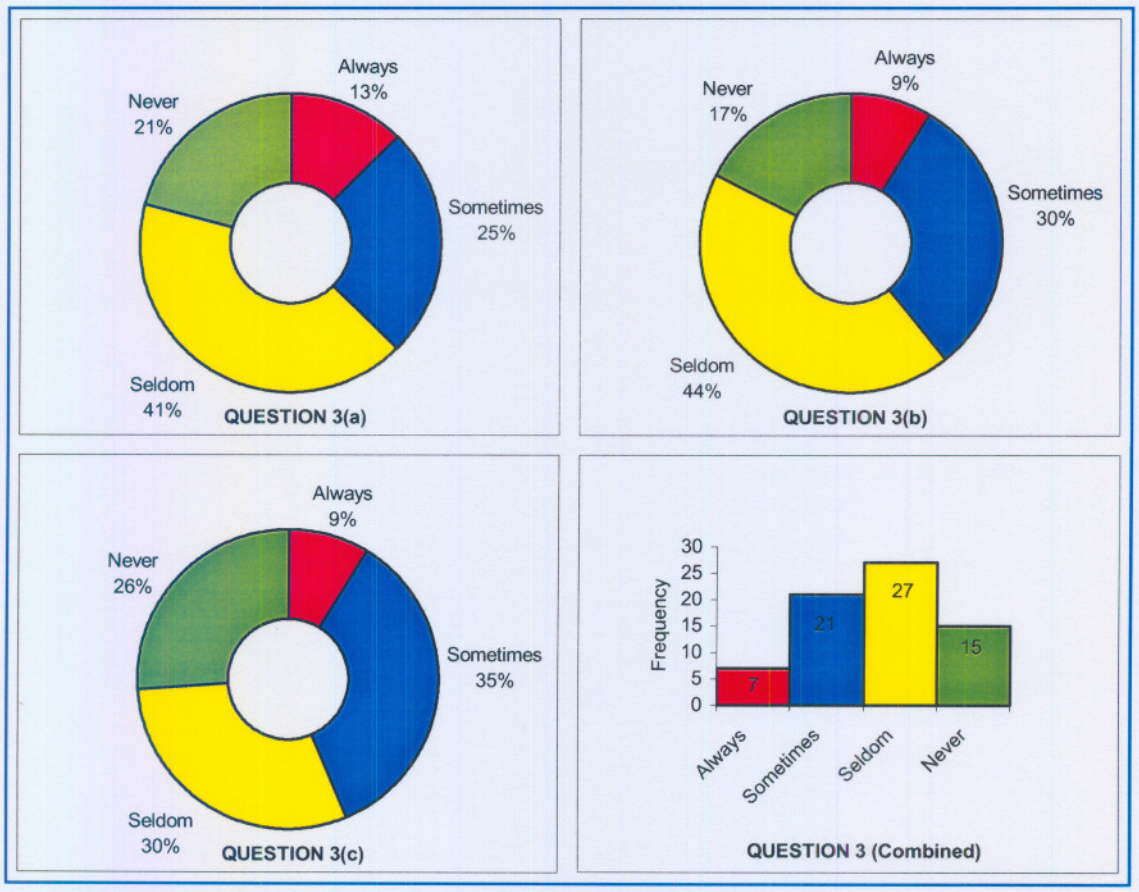
Table 9: Quantitative Analysis: Question 3 (Design Phase)

	Q3 a	Q3 b	Q3 c	Q3 Combined
Arithmetic Mean	-0.3	-0.3	-0.3	-0.3
Median	-1.0	-1.0	-1.0	-1.0
Standard Deviation	1.4	1.3	1.4	1.4
Range	4	4	4	4
Frequency Distribution				
Always	3	2	2	7
Sometimes	6	7	8	21
Seldom	10	10	7	27
Never	5	4	6	15

Source: Own

The Arithmetic Mean as revealed in Table 9 maintains a constant –0.3 throughout Question 3 with the same pattern repeated for the Median at a constant –1.0. All possible choices were elected by respondents, reflected by the Range value of 4 throughout the question. Respondents reported with quite a variety around the Mean, indicated by the Standard Deviation of between 1.3 and 1.4 for all sub-statement as well as Question 2 combined.

Figure 10: Frequency Distribution: Question 3 (Design Phase)



Source: Own

The Frequency Distribution as illustrated in Figure 10 supports the significantly negative response to all sub-statements of Question 3 as indicated by the *Means* and *Median* stated in Table 9. As can be seen from the graph *'Question 3 (Combined)'*, the overall distribution for all combined responses for Question 2, shows some signs of a symmetrical. The mid-values (*Sometimes* and *Seldom*) are somewhat different and represent 69% of all responses. This is supported by the Combined *Standard Deviation* of 1.4. The *Never* option makes up 21% of all responses whereas the *Always* option is around 50% lower at 10%. For Question 3(a) the positive/negative response split is 38/62. For Question 3(b) the split is 39/61 positive/negative and 44/56 for Question 3(c). The positive/negative ratio for Question 2 combined is 40/60. All sub-statements for Question 3 show a fairly decisive tendency towards the negative side.

Conclusion

Once again, the results show that during the Design Phase (a development responsibility) SLA integration is mostly ignored. A database and systems core design that caters for SLA requirements is the exception rather than the rule, with systems that offer embedded SLA performance requirements reported even less often.

4.4.2.4 Question 4 (Programming Phase)

During the Programming Phase of the SDLC, programmers develop:

a. Applications which meet SLA Application Performance requirements

b. Application performance measurement functionality as embedded features

Question 4 is directly derived from the model's critical success factors relating to the Programming Phase of the SDLC and as with the previous phase is a development organisation function.

Quantitative Analysis

Table 10: Quantitative Analysis: Question 4 (Programming Phase)

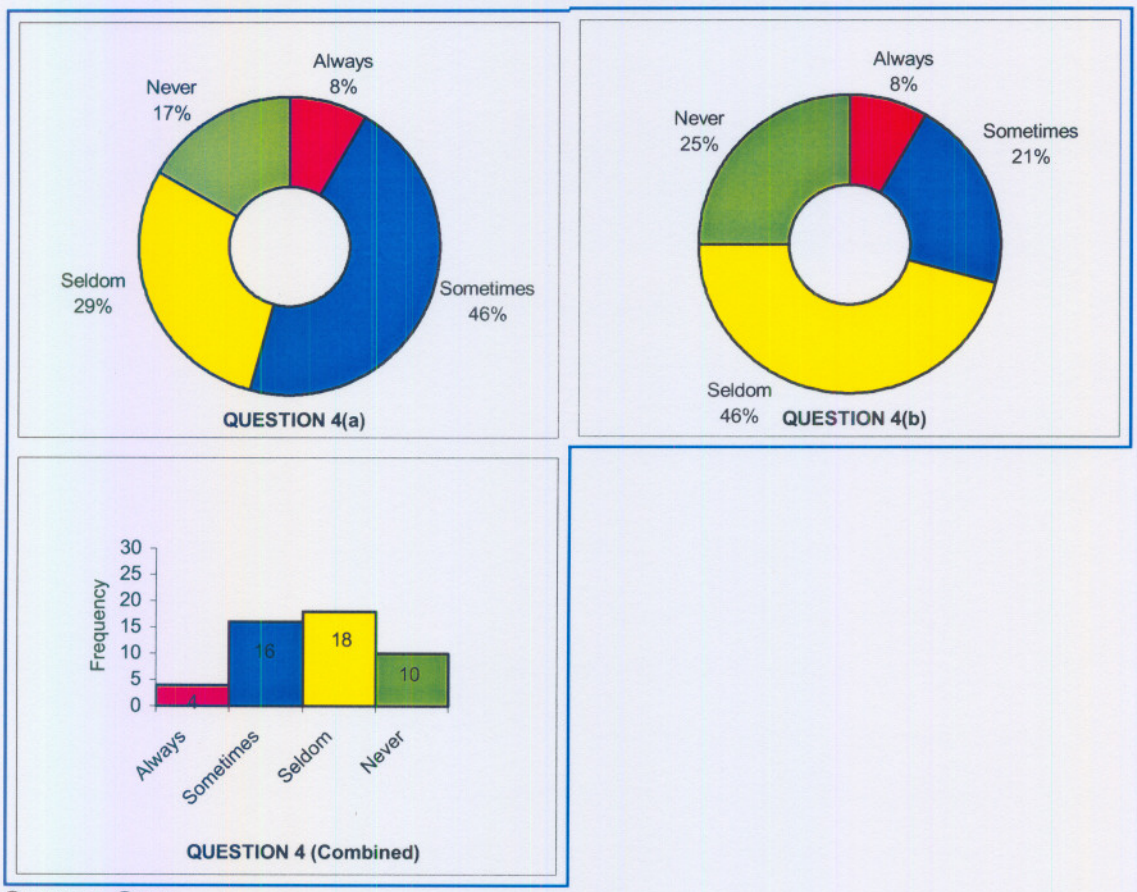
	Q4 a	Q4 b	Q4 Combined
Arithmetic Mean	0.0	-0.6	-0.3
Median	1.0	-1.0	-1.0
Standard Deviation	1.4	1.3	1.4
Range	4	4	4
Frequency Distribution			
Always	2	2	4
Sometimes	11	5	16
Seldom	7	11	18
Never	4	6	10

Source: Own

The Arithmetic Mean as shown in Table 10 shows a neutral value for Q4a (0.0) and a –0.6 for Q4b. this phenomena is also reflected in the Median values of 1 for Q4a and –1 for Q4b. once again all possible choices were selected as indicated by the Range value of 4 throughout

the question. Respondents reported with quite a variety around the Mean as shown by a Standard Deviation of between 1.3 and 1.4 for all sub-statement including Question 2 combined.

Figure 11: Frequency Distribution: Question 4 (Programming Phase)



Source: Own

Figure 11 shows the frequency distribution for Question 4. There is a distinct difference in the positive/negative ratios for Q4a and Q4b, with a 54/46 ratio and an unbalanced 29/71 ratio respectively. The combined ratio is 42/58. As can be seen from the graph *'Question 4 (Combined)'*, the overall distribution for all combined responses for Question 4 show definite signs of symmetry, i.e. a normal distribution. The mid-values (*Sometimes* and *Seldom*) are only slightly different and represent 71% of all responses. As with Question 3, this is supported by the Combined *Standard Deviation* of 1.4. The *Never* option makes up 21% of all responses whereas the *Always* option is markedly lower at 8%.

Conclusion

The situation created during the Design Phase is repeated fairly consistently during the Programming Phase of the SDLC. Very few programs that are written offer embedded SLA performance requirements reporting functionality. Although more respondents pertinently admit that their programs *sometimes* meet SLA requirements than those admitting that their programs *seldom* meet SLA performance requirements, one may have doubts whether the adherence is pure coincidence and not necessarily a result of conscious consideration of SLA requirements. This is supported by the results of Question 3b, where respondents categorically admitted that they do not design programs that adhere to SLA performance requirements.

4.4.2.5 Question 5 (Testing Phase)

During the Test Phase of the SDLC:

a. Testing is done according to the Test Plan specified by users as part of the UFRS
b. High-Volume testing is done to assess system performance user operational conditions

Question 5 represents the critical success factors involved during the last pre-delivery phase of the SDLC and both developers and users are responsible for these actions.

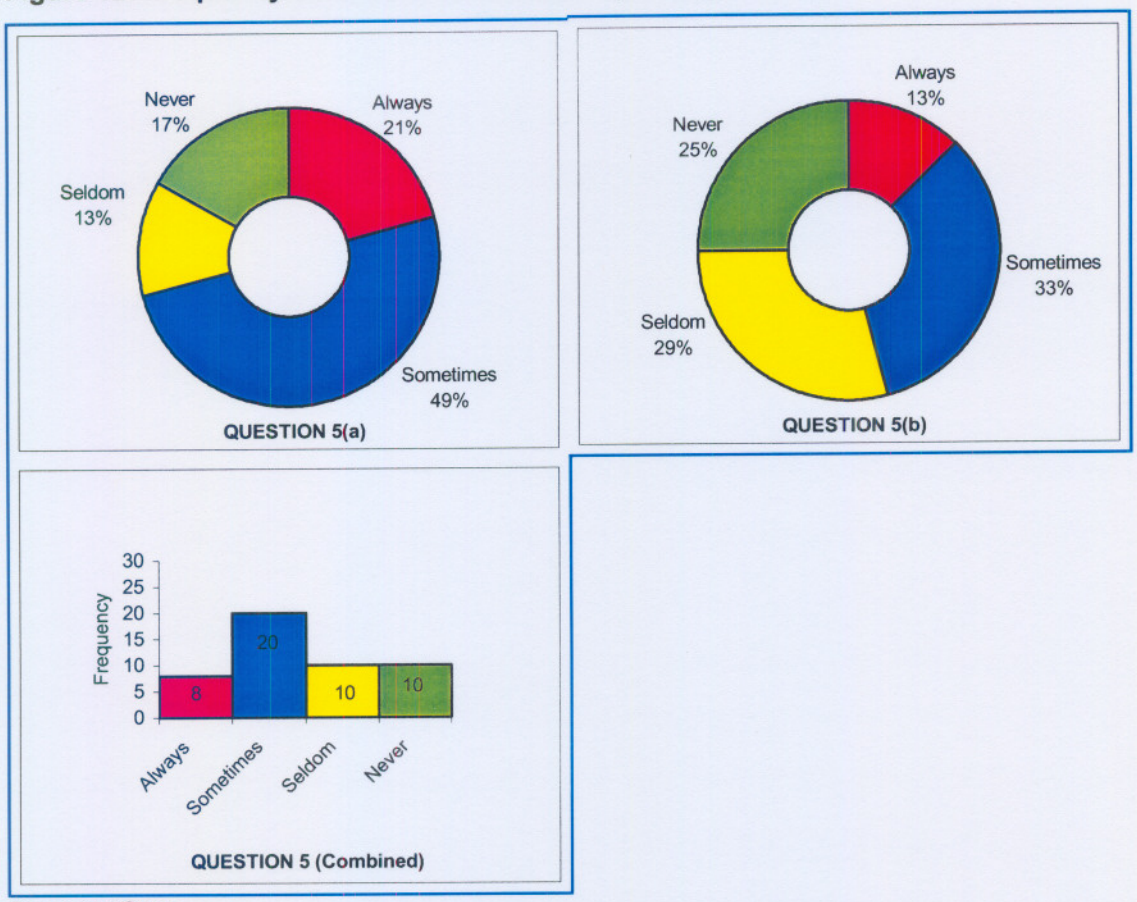
Quantitative Analysis

Table 11: Quantitative Analysis: Question 5 (Testing Phase)

	Q5 a	Q5 b	Q5 Combined
Arithmetic Mean	0.5	-0.2	0.1
Median	1.0	-1.0	1.0
Standard Deviation	1.4	1.5	1.5
Range	4	4	4
Frequency Distribution			
Always	5	3	8
Sometimes	12	8	20
Seldom	3	7	10
Never	4	6	10

Source: Own

Figure 12: Frequency Distribution: Question 5 (Testing Phase)



Source: Own

As indicated by Table 11, Q5a and Q5b show prominently different *Mean* values, 0.5 and –0.2 respectively. This is confirmed by the *Median* values in the same table as well as the *Frequency Distribution* shown in Figure 12. The *Range* value of 4 throughout Question 5 indicates the selection of all possible option for the question. The even spread shown in the '*Question 5 (Combined)*' graph in Figure 12 is confirmed by the *Standard Deviation* of 1.5. It can be clearly seen that the field experts had significantly different responses to the two sub-statements of Question 5, Q5a definitely confirmative and Q5b marginally negative.

As stated above, the *Frequency Distribution* shown below shows a predominantly positive response to Q5a with a positive/negative ratio of 71/29, whilst Q5b shows a moderate 46/54 ratio. For Question 5 combined, the mid-values (*Sometimes* and *Seldom*) are considerably

different with *Sometimes* (42%) exactly double the value of *Seldom* (21%). These mid-values thus represent 63% of all responses. The *Never* option makes up 21% of all responses with the *Always* option a similar 17%.

Conclusion

The results of this question to some degree support the corresponding questions from Question 2, specifically related to Test Plans and subsequently a noteworthy positive response were given to the question. From this one can make the conclusion that SLA requirements are integrated to some extent to the SDLC in this regard. 12.5% of respondents stated that they *always* do high-volume testing, which is an critical aspect of measuring real application performance.

4.4.2.6 Question 6 (Post-Implementation Phase)

During the Post-Implementation Phase:

a. SLAs and Systems are kept in alignment whenever changes are made to either one
b. Regular Systems Performance Reviews are done, based on SLA requirements
c. SLA Reporting is done and is based on stored Application Performance data
d. Penalties/Rewards are applied in cases of under or over delivery

Question 6 addresses the factors critical to the success of SLAs in the software development environment that should be present after the system is developed, implemented and is operational.

Quantitative Analysis

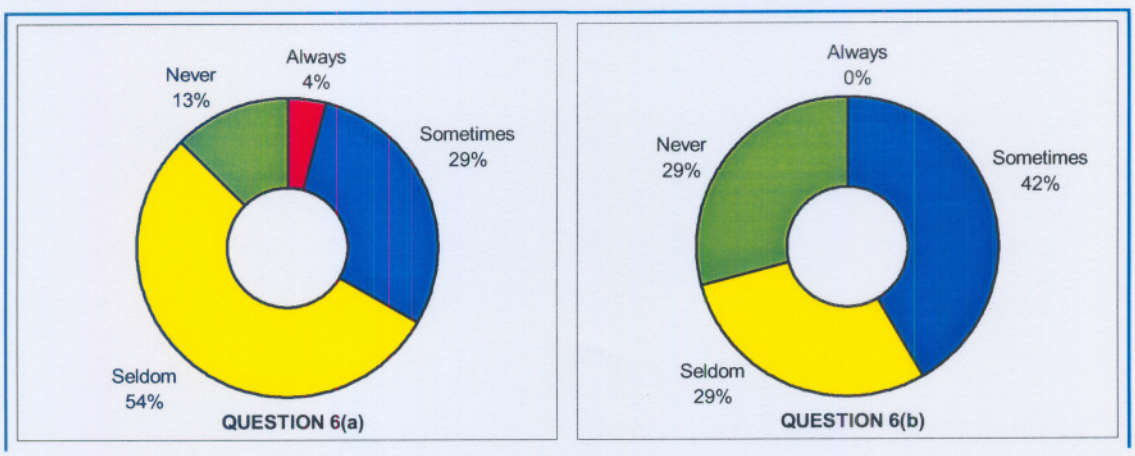
Table 12: Quantitative Analysis: Question 6 (Post-Implementation Phase)

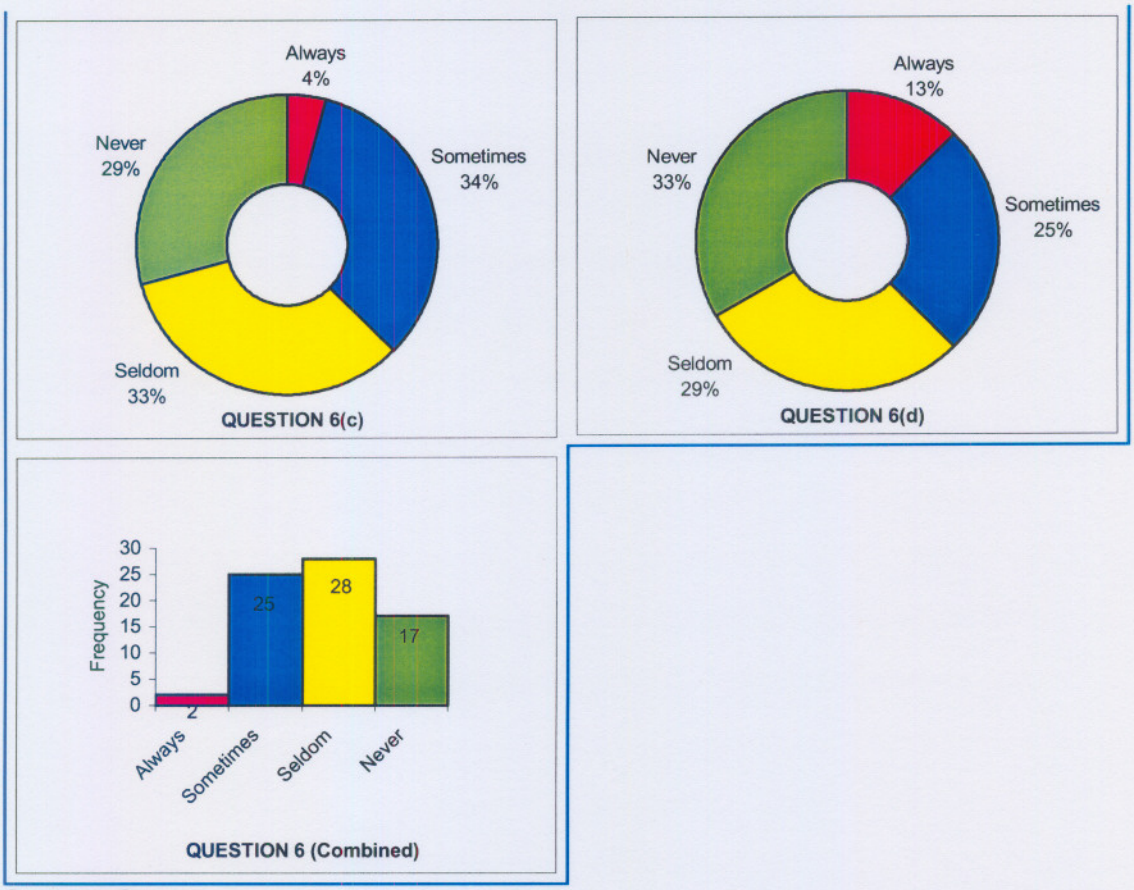
	Q6 a	Q6 b	Q6 c	Q6 d	Q6 Combined
Arithmetic Mean	-0.4	-0.5	-0.5	-0.5	-0.5
Median	-1.0	-1.0	-1.0	-1.0	-1.0
Standard Deviation	1.2	1.3	1.4	1.5	1.3
Range	4	3	4	4	4
Frequency Distribution					
Always	1	0	1	3	2
Sometimes	7	10	8	6	25
Seldom	13	7	8	7	28
Never	3	7	7	8	17

Source: Own

All sub-statements for Question 6 had an overall negative result. This is defined by the *Arithmetic Mean* values of –0.4 for Q6a and –0.5 for the remainder. All sub-statements show a *Median* of –1. The *Standard Deviation* increases from 1.2 for Q6a up to 1.5 for Q6d. This indicates a tighter grouping around the *Arithmetic Mean* for Q6a, incrementally moving towards a somewhat looser grouping for Q6d. The *Range* value of 3 for Q6b shows that, apart from this case, all options were selected for the remaining sub-statements. The Combined values clearly confirms that the responses for Question 6 overall were definitely towards the negative side.

Figure 13: Frequency Distribution: Question 6 (Post-Implementation Phase)





Source: Own

As confirmed by the Quantitative Analysis aspects shown in

Table 12 above, the *Frequency Distribution* for Question 6 sub-statements are predominantly towards the negative half of the scale. The *Always* option was only selected in three of the four cases, with a combined overall value of just 3%. To the other extreme, the *Never* option was selected in all four cases with a combined value of 24%. This is supported by the *Seldom* option with a combined value of 39%, with the remaining 35% allocated to the *Sometimes* option.

Conclusion

The results from Question 6 imply that very little aftercare occurs when it comes to the integration of SLAs and the SDLC. Issues like regular reviews, reporting and alignment seem to be overlooked once the software application system is up and running. The fact that only 13% of respondents *always* apply penalties and rewards, compared to 33% that *never* apply might

an indication of the state of affairs regarding SLAs during the Post-Implementation Phase, i.e. not functional and hence not measurable for the application of penalties and rewards.

4.4.2.7 Question 7

Rate the importance of the following: (Refer to the statements in Questions 1 to 6 above)

a. Client SLA requirements are known in the Planning Phase of the SDLC
b. Client resources allocated for SLA ownership
c. Client & System Developer collaboratively agreeing on SLA contents
d. User Functional Requirement Specification (UFRS) to include Application Performance Requirements
e. UFRS to include metrics for measurement of Application Performance
f. UFRS to include formal test plan
g. Database design to cater for storage of application performance data
h. System's core design to be aligned with SLA requirements
i. System's design to include embedded functionality for application performance measurement
j. Application coded to meet SLA performance requirements
k. Application performance measurement functionality coded as embedded features
l. System testing done according to specified test plan
m. High-volume testing to form part of user acceptance testing
n. SLAs and systems kept in alignment when changes are made to either one
o. Regular systems performance reviews to be conducted
p. Regular SLA reporting to be done
q. Penalties & Rewards to be applied

In contrast with Questions 1 to 6, Question 7 were included to assess the necessity or importance of the existence of the critical success factors under ideal conditions, i.e., the questions (statements) were intended to revoke the respondents' opinion of how important

he/she considers each of the critical success factors as part of the integration model for SLAs and the SDLC.

Where questions 1 to 6 had a rating that ranged from *'Always'* on the one end to *'Never'* on the other, the assumption was made that for the model to be successful (under ideal circumstances), the critical success factors should *always* be present. It would therefore not bear the same meaning should the same questions as stated in 1 to 6 were asked as an assessment under *ideal* conditions instead of under *current* conditions as this could result in respondents indicating the *occurrence frequency* of critical success factors under ideal circumstances, instead of rather conveying their opinion of *importance*. The author rather wished for an assessment on how strong the respondent's belief was regarding the presence of the critical success factor within the model assuming that if it should be present, it will in all cases be present. Three options were offered, i.e.: *High*, *Medium* and *Low* with no option for *Don't Know*, seeing that this is an opinion.

For the analysis of Question 7, the sub-statements related to Questions 1 to 6 were extracted and plotted against the results of the individual Questions 1 to 6. Seeing that questions 1 to 6 had four different options (Always, Sometimes, Seldom and Never) compared to Question 7 with only three (High, Medium and Low), the frequency distributions for *combined* values were plotted on a graph with *alternative X- and Y Axis*. Trend lines were added for each of the series to obtain the gradient of each, and from that is determined if the results for the *current situation* are in conflict or in harmony with the *rating of importance* for the same critical success factors.

Table 13 contains the *Gradients* calculated from the *Frequency Distribution* data sets for each sub-statement for Questions 1 to 6, compared to the *Gradients* derived from the *Frequency Distribution* data sets for each sub-statement as enclosed in Question 7. The *Gradient* reflects the slope of the *Linear Regression Line* that could be calculated from the same data sets. If graphically plotted, the *Regression Line* would illustrate the distribution towards either the Positive or Negative pole of the *Frequency Distribution* chart. From that one can conclude what

the majority response was on any of the stated questions/sub-questions, as previously analysed and expressed by the *Positive/Negative* ratio. The column 'Difference' shows the extent to which the two gradients in fact differ. A 'Y' in the last column, i.e. 'Equal Sign' should indicate that although the gradients differ in value, the slope of the illustrative regression lines are similar in terms of increasing/decreasing towards the span of the X-Axis. This in fact suggests that the same tendency exists with regards to the frequency distribution of the responses, i.e. both indicating a primarily positive or negative outcome.

Quantitative Analysis

Table 13: Gradient Comparison: Current Situation vs. Importance Rating

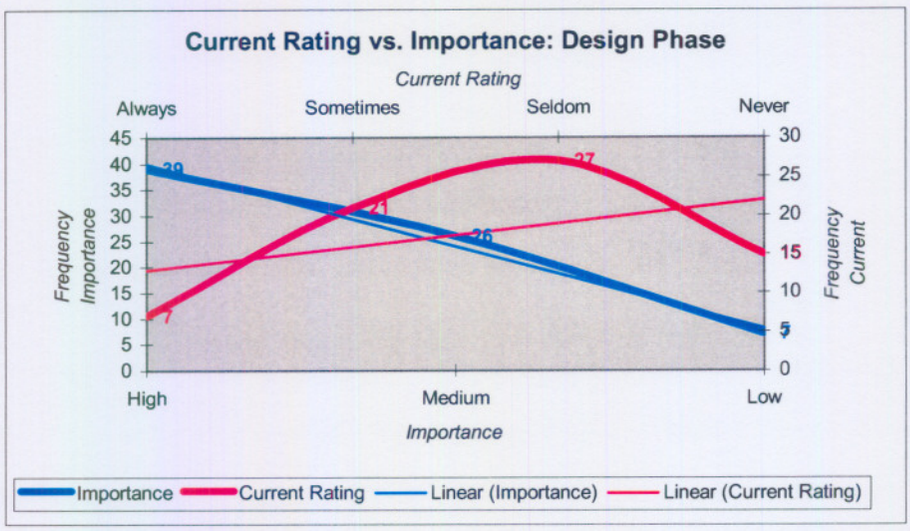
			Gradient			
		Related Question	Current Situation	Importance	Difference	Equal Sign
SDLC Phase	Planning	Q1a	0.6	-9.5	-8.9	N
		Q1b	0.4	-5.0	-4.6	N
		Q1c	0.6	-8.0	-7.4	N
	Analysis	Q2a	0.4	-6.5	-6.1	N
		Q2b	1.0	-7.5	-6.5	N
		Q2c	1.5	-8.0	-6.5	N
	Design	Q3a	1.0	-5.5	-4.5	N
		Q3b	0.9	-5.5	-4.6	N
		Q3c	1.1	-5.0	-3.9	N
	Programming	Q4a	0.2	-6.5	-6.3	N
		Q4b	1.8	-3.0	-1.2	N
	Testing	Q5a	-1.2	-8.0	-6.8	Y
		Q5b	0.8	-6.5	-5.7	N
	Post-Implementation	Q6a	1.2	-7.5	-6.3	N
		Q6b	1.8	-4.0	-2.2	N
		Q6c	1.8	-6.0	-4.2	N
		Q6d	1.6	-4.0	-2.4	N

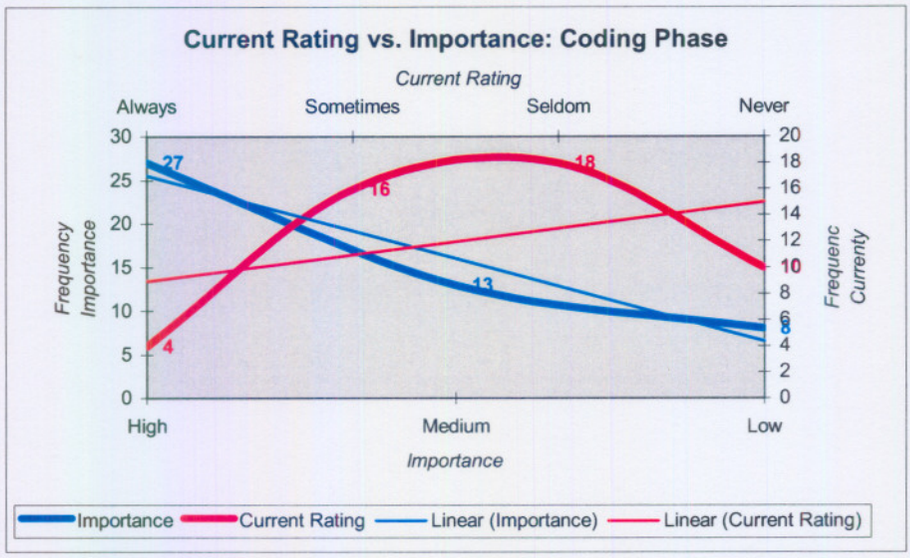
Source: Own

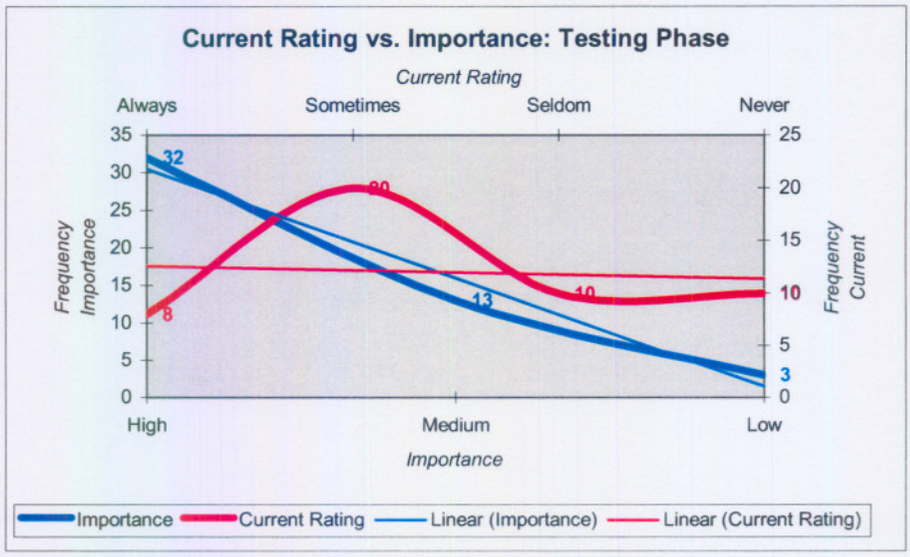
As shown in Table 13, only one sub-statement, Q5a which relates to the statement '*Testing is done according to the Test Plan specified by the users as part of the UFRS*' shows a

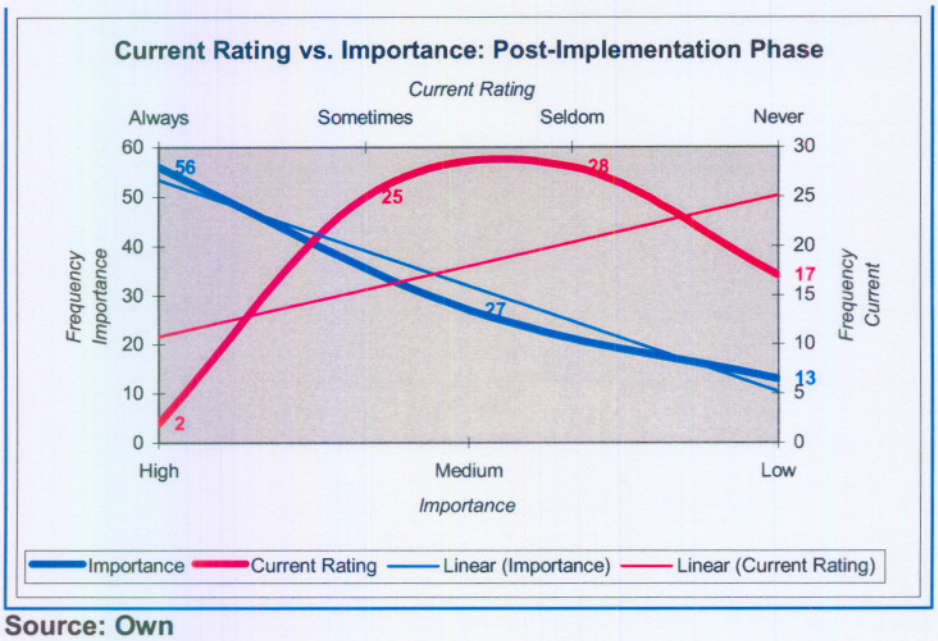
similar gradient for both the *Current Situation* and the *Importance* assessment. By implication one can therefore assume that field experts are in agreement that, apart from the fact that 67% of them rated this as of High importance and the remaining 33% of Medium importance, they are also currently exercising testing according to the specified Test Plan. This is supported by the 71/29 positive/negative ratio stated previously. (See Table 11). The remainder of the sub-statements for all phases of the SDLC do not only show significant differences in terms of gradient, but also opposite signs, implying contradictory results for the Current Situation and the Importance rating.

Figure 14: Comparison: Current Rating vs. Importance: SDLC Phases


Current Rating vs. Importance: Design Phase
Current Rating
Always
Sometimes
Seldom
Never
Frequency Importance
Frequency Current
High
Medium
Low
Importance
Importance
Current Rating
Linear (Importance)
Linear (Current Rating)


Current Rating vs. Importance: Coding Phase
Current Rating
Always
Sometimes
Seldom
Never
Frequency Importance
Frequenc Currenty
High
Medium
Low
Importance
Importance
Current Rating
Linear (Importance)
Linear (Current Rating)


Current Rating vs. Importance: Testing Phase
Current Rating
Always
Sometimes
Seldom
Never
Frequency Importance
Frequency Current
High
Medium
Low
Importance
Importance
Current Rating
Linear (Importance)
Linear (Current Rating)



Source: Own

The graphs displayed in Figure 14 contains a graphical representation of the frequency distribution of the combined responses for questions 1 to 6, compared to the responses for Question 7. Smoothed curves as well as a linear regression line for both cases present the data series. Of all the phases represented, the Test Phase has the best correlation between the current scenario and the importance rating. Therefore the conclusion can be made that of all the SDLC phases the critical success factors set for the Test Phase are not only considered of high importance but are in fact currently applied at the various organisations surveyed.

Conclusion

The results from Question 7 linked to the results of question 1 to 6 can perhaps be summarised by the idiom *"They don't practise what they preach"*. The critical success factors for SLA and SDLC integration inherent in this questionnaire seem to be of low importance if judged by the incidence thereof in the current environment (questions 1 to 6). This is in absolute contradiction to the opinion of the same field experts when asked to rate their importance of the existence of the same critical success factors. All critical success factors were rated higher in terms of importance compared to the actual presence of thereof.

4.4.2.8 Question 8

Within the Software Development Paradigm, SLAs:

a. Contribute towards higher service levels

b. Offer financial benefits that exceed the additional resource cost and effort

c. Result in better client/service provider relationships

This question was purely opinion-based and dealt with general SLA issues often debated. Although the approval/disapproval of these is beyond the scope of this study, they were included in the survey as to obtain an indication of expert opinion. Options that ranged from *Always* to *Never* were offered.

Quantitative Analysis

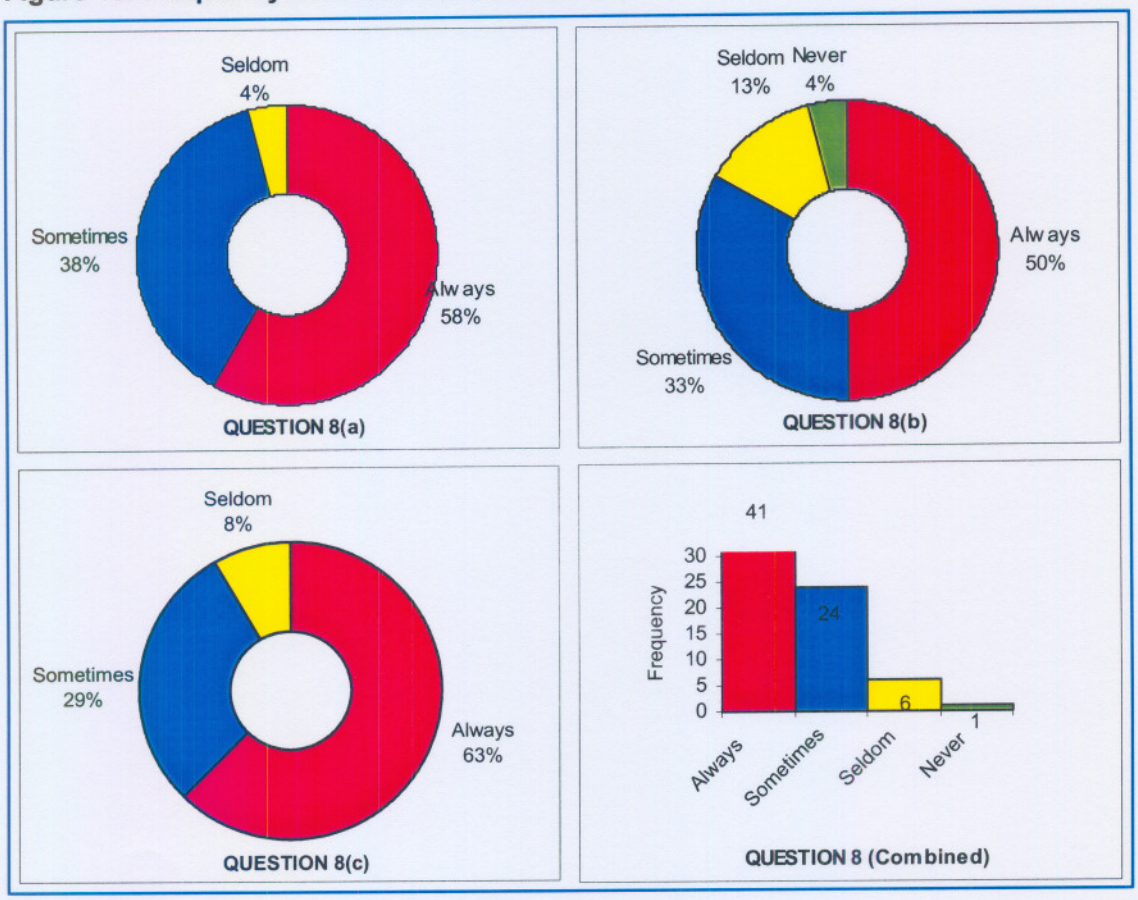
Table 14: Quantitative Analysis: Question 8

	Q3 a	Q3 b	Q3 c	Q3 Combined
Arithmetic Mean	1.5	1.1	1.5	1.4
Median	2.0	1.5	2.0	2.0
Standard Deviation	0.7	1.2	0.9	1.0
Range	3	4	3	4
Frequency Distribution				
Always	14	12	15	41
Sometimes	9	8	7	24
Seldom	1	3	2	6
Never	0	1	0	1

Source: Own

The same weightings as used for the analysis of questions 1 to 6 as illustrated in Table 6 were applied for the analysis of Question 8. The response on all three sub-statements for this question was remarkably positive. This is reflected by the *Arithmetic Mean* of 1.5 for both Q8a and Q8c and a 1.1 for Q8b. The *Median* shows an even more positive response with values of 2 for Q8a and Q8c and 1.5 for Q8b. as can be seen from the *Frequency Distribution* only 1 out of the total 72 responses was 'never' (1.4%). The combined positive/negative ration was 90/10.

Figure 15: Frequency Distribution: Question 8



Source: Own

The combined Frequency Distribution in Figure 15 confirms the stated 90/10 positive/negative ratio. Only for Question 8(b) did one respondent select *Never*, while *Always* exceeded 50% in all questions.

Conclusion

From the responses to the questions under Question 8, it is absolutely clear that all experts consider SLAs within the software application systems environment to be of great value. One can therefore categorically say that supported by the research, SLAs are believed to increase service levels and the quality of relationships, and in addition, result in higher financial benefits. It is important though to remember that this conclusion should be read within the contexts, that it is general belief and does not by any means imply that these benefits are in fact realised currently.

4.4.2.9 Question 9

What is your opinion on integrating the SLA Life Cycle into the Systems Development Life Cycle as suggested in statements above

The purpose of Question 9 was to allow experts a free and unbiased opportunity to raise their opinion on the research topic. This was an open-ended question with no pre-set ratings or options and respondents were free to state their view. No limit was set on the length of the answer. Answers were processed and once again weightings of 2,1,-1,-2 were assigned to the answers based on the extent to which the opinion was positive or negative. One respondent stated 'No comment', which led to the response being discarded. Some of the details of these responses will be discussed later in this study as supportive or non-supportive arguments.

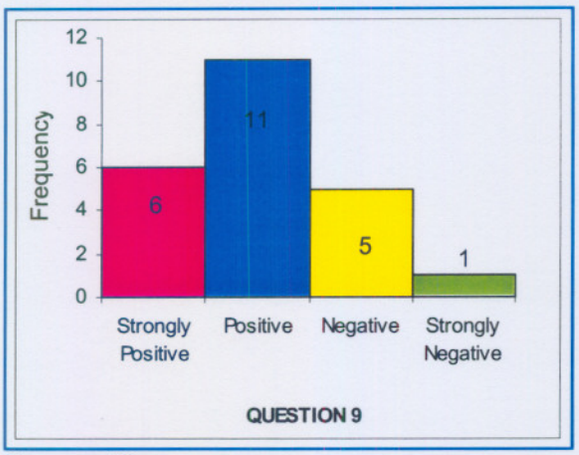
Quantitative Analysis

Table 15: Quantitative Analysis: Question 9

	Q9
Arithmetic Mean	0.7
Median	1.0
Standard Deviation	1.2
Range	4
Frequency Distribution	
Strongly Positive	6
Positive	11
Negative	5
Strongly Negative	1

Source: Own

Figure 16: Frequency Distribution: Question 9



Source: Own

From both Table 15 and Figure 16 it can be seen that the comments (unbiased) related to the integration of the SLA Life Cycle with the SDLC were surprisingly positive. The *Arithmetic Mean* is 1.3, supported by a *Median* of 1. The widely spread frequency distribution (although significantly to the positive side) resulted in the *Standard Deviation* of 1.2. Figure 16 illustrates the positive/negative ratio of 74/26 and only 4% of the respondents that strongly disagree with the suggested integration, in contrast with the 26% that strongly agree.

Qualitative Analysis

The majority of experts responded positively on the suggested integration of SLAs to the SDLC. Various valid and constructive suggestions were offered, amongst others, a suggestion to treat SLA management with the same 'respect' as project management. The specific respondents argued that organisations develop proper methodologies to manage projects, and that similar methodologies should be developed as standard procedure to manage SLAs.

A couple of respondents raised their concerns with regards to the ongoing alignment of SLAs to the software application systems during the post-implementation phase. Together with this, various experts mentioned the issue of profitability, especially for small development organisations and argued that additional time and resources for SLA integration and management could often significantly affect financial results and related profitability.

4.4.2.10 Question 10

What is your current position?

- ICT Manager
- Business Analyst
- Development Manager
- Project Manager
- Systems Analyst
- Other

Question 10 was included to give the author the ability to assess the 'qualification' of the respondents. This was merely a control variable and also to put the opinion allowed in Question 9 into some perspective, based on the respondent's position.

4.4.2.11 Question 11

Do you think that this survey could be used to periodically assess the status of SLAs within the Software Development Paradigm?

- Yes
- No

Question 11 was included as a last question to obtain opinion on the Research Instrument, i.e. the questionnaire. This was the only question not related to SLAs in operation. Respondents were allowed a 'Yes' or 'No' reply.

Quantitative Analysis

23 of the 24 respondents (96%) agreed that this questionnaire could be used to frequently assess the status of SLAs within the Software Development environment. Interestingly enough, the same respondent was very positive about the integration of the SLA with the SDLC.

Conclusion

From the results one can assertively say that the questionnaire used for the research, could confidently be used as an instrument for regularly assessing the state of SLA and SDLC integration.

4.5 Chapter Summary

Chapter Four conveys the aspects regarding the research instrument used for performing the empirical study. It also includes the actual research results along with the quantitative and qualitative analysis thereof. Finally, it states a brief interpretation and conclusion for each section of the research information.

5 CHAPTER 5: CONCLUSION AND RECOMMENDATIONS

5.1 Introduction

This chapter contains the conclusions derived from the results of the research findings, recommendations for the implementation and management of SLAs within the outsourced software application environment, as well as suggestions for possible future research. The initial problem statement was considered throughout the study and the literature studied taken into definite account.

The rationale of this study was to explore the success of service level agreements within the outsourced software application system environment. The reasoning was that various properly qualified authors and authorities very often raised their concern about the successful application of SLAs and therefore the necessity to conduct this research.

The initial objectives set were met by gathering appropriate and valid data, analysing that and interpreting the results and formatting such derivatives into recommendations for future use and possible research.

5.2 Conclusions

Based on inputs from previous literature researched, a model was constructed for the integration of SLAs with the SDLC. *Certain critical success factors were identified as essential embedded components of the model for SLAs to be successful within the outsourced software application system* environment. Each of the six phases of the SDLC contained number of critical success factors.

The empirical study shown that not even in the first phase, i.e. the Planning Phase, the set critical success factors were present, indicating that SLA and SDLC alignment is not happening. This is of great concern. This tendency is followed throughout the Analysis, Design and

Programming phase of the SDLC. The Post-Implementation Phase, as with the pre-implementation phases, also raised some concerns regarding the continuous alignment and measurement of SLAs to the SDLC. The only phase that showed a somewhat positive result was the Testing Phase.

The research showed that the majority of the experts who took part in the research are of opinion that SLAs can in fact have positive effects on the business in terms of profitability and better relationships with customers. This was supported by the importance rating of the critical success factors of SLA and SDLC integration, where the majority of experts clearly agreed to the importance of integrating the SLA and SDLC life cycles.

An interesting aspect is that of penalties and rewards, which this research has shown, are not enforced by many organisations. Half the respondents however indicated that it is of high importance to have obligatory penalties and rewards as part of SLAs.

Although the study and related research was done within the *outsourced* software application system development environment, it became clear that this model could also be used for *internal* software systems development. The post-development handover will be different in the sense that the internal ICT department will hand over the system to the production or business operations department, compared with the handover from the outsourcing company to the ICT/Production department in the case of outsourced development.

Lastly, a positive outcome of this study was the fact that the research instrument (based on the SLA and SDLC integration model) proved to be an effective tool for regular measurement of the status of SLA and SDLC integration.

5.3 Recommendations

To increase the success of SLAs within the software application systems development environment, the following is recommended:

5.3.1 SLA Awareness

Awareness for SLAs must be created, especially at the development organisation. Due to the technical nature of the expertise of development organisations, they prefer to focus on the technical aspects of software application systems and the functional outcomes thereof. By establishing SLA awareness in these organisations, SLAs could more easily be integrated into the SDLC.

5.3.2 Integration Initiative

SLAs still seem to be considered a client responsibility. Development organisations should take the initiative in introducing SLA requirements early in the development process. Client organisations are often not aware that SLA requirements need to be communicated and specified up front, and assume that once systems are delivered they will be in adherence to SLA requirements.

5.3.3 Systems Development Methodologies

Development organisations should expand their Systems Development Methodologies to specifically address SLA requirements and to ensure that such requirements are incorporated throughout the SDLC. The SLA and SDLC Integration model could be incorporated into such methodologies.

5.3.4 Account Management

To ensure SLA and application alignment during the post-implementation period of the system, appropriate alignment plans should be made the responsibility of client account managers, responsible for managing the ongoing maintenance and support of client software application systems. Account managers should form part of the SLA committees, together with client representatives.

5.3.5 SLA Status Measurement

It is recommended that regular measurement be done on the status of SLA and SDLC integration. The model and survey instrument used in this study, which proved to be effective, would be used for assessment. From regular assessments, a trend analysis should be done to allow management to properly manage the success of SLAs for outsourced software application systems.

5.4 Limitations in Research

The first and foremost limitation was a limited sample of the population. Although utmost effort was put into obtaining quality and expert input from field experts, there is room for enlarging the sample. As with many similar studies, the problem of limited generalisation is warranted in this case too.

The suggested SLA and SDLC integration model could not been tested by means of a longitudinal study and the author had to measure the success of the model based on expert opinion.

The scope of the research was limited to only certain aspects of SLAs within the software application systems paradigm, i.e. application performance. SLA components of equal importance such as the legality and the pure service aspect did not form part of this study.

The suggested model for integrating the SLA life cycle with the SDLC was the only model applied as basis for empirical research. The critical success factors identified as crucial embedded elements of the model, were the only factors measured by means of the data survey. No other possible models were considered during empirical data collection.

A technical limitation encountered during the data gathering phase of the study was research questions, intended to be quantitatively compared by means of data set correlation, which offered a different number of response options on the survey. This complicated the quantitative analysis process and restricted the author in the use of the data.

5.5 Recommendations for Future Research

The first suggestion would be to conduct confirmative research on the success of the suggested SLA and SDLC integration model in the form of a longitudinal study. Due to time constraints the model's success had to be measured by means of expert opinion.

As mentioned in section 5.4 above, the scope of this study was narrowed to the implementation and management of SLAs within the SDLC. Specific attention was given to the software application performance elements, which represent only one aspect of the total SLA. Apart from the legal aspect, another crucial aspect of the SLA is the support service component that becomes of high importance at the start of the post-implementation phase. Very little existing research could be found on this specific element of SLAs in the outsourced software application system environment. This opens up a possibility for future exploratory research.

Another possibility for future research is certainly to repeat similar research on larger samples of the population. More representative data might be obtained, which could lead to more accurate findings. The relatively limited sample used in this study is an acknowledged weakness.

As stated earlier, this study was conducted from a management perspective and although the importance of user involvement was duly mentioned and taken into account, the users as such were never involved during the empirical study. Possible future studies could include user opinion and experience as part of the empirical research.

5.6 Summary

As mentioned right at the beginning of this document, the future economy will be dominated by service industries. Add to that the tendency for organisations to focus on what they do best and allow external sources to perform supporting functions; the importance of proper service levels becomes apparent.

With this increasing importance of service levels and the successful management thereof, previous literature was studied, with specific focus on SLAs within the outsourced software application environment. No substantial previous research on SLAs within this specific context could be found, which led to this study.

As part of this study, a model has been developed for the integration of SLAs with the SDLC. Additional to the model, a measurement instrument has also been constructed that will allow management to frequently measure the state of SLAs within the systems development paradigm.

The author made use of a questionnaire for the gathering of data from field experts from various organisations involved in outsourced software application systems development. Supplementary to that, informal discussions with various role players in this arena were held.

The first chapter of this document discusses the background to the problem as well as specific problems that are encountered in the field of SLAs within the software application development environment.

A literature study makes up Chapter Two. Various other sources of literature were studied and compared to form an understanding of what has previously been researched on the topic. Literature on the various components of this study were specifically researched, i.e. outsourcing, SLAs in general and the systems development life cycle.

Chapter Three contains the recommended model for the integration of the SLA life cycle with the systems development life cycle. The key success factors that are required for this model to be successful were stipulated and discussed in this chapter.

An empirical study was done and the methodology, as well as the findings thereof is enclosed in the fourth chapter. An analysis of the field expert responses was done and the conclusions thereof noted.

The last chapter, Chapter Five, lists the overall conclusions of the study was well as certain recommendations to improve the success of SLAs. In addition, certain recommendations for future study are contained in this chapter.

---oooOooo---

BIBLIOGRAPHY

COOPER, D.R. & SCHINDLER, P.S. 2001. Business Research Methods. 7th ed. New York : McGraw-Hill Irwin. 798 p.

CUMMINGS, T.G. & WORLEY, C.G. 2005. Organisational Development & Change. 8th ed. Ohio : Thomson South Western. 694 p

CUNNINGHUM, M.J. 2001. Partners.com. The Blueprint for Innovative Alliances in a Networked Economy. Harlow : Perseus. 280 p.

DENSCOMBE, M. 2002. Ground Rules for Good Research: A 10 Point Guide for Social Researchers. Philadelphia : Open University Press. 219 p.

DENZIN, N.K. & LINCOLN, Y.S. 1994. Handbook of Qualitative Research. California : Sage. 643 p.

FLANNERY, E. 2003. Want the most from IT service levels? Be proactive! [Web:] http://www.searchcio.com [Date accessed: 5 March 2005].

FOWLER. F.J. & MANGIONE. T.W. 1990. Standarised Survey Interviewing: Minimising Interviewer-Related Error. Calefornia : Sage. 151 p.

FRENZEL, C.W. 1992. Management of Information Technology. Boston : Boyd and Fraser. 550 p.

HAIG, M. 2001. The B2B e-Commerce Handbook. London : Clays. 263 p.

HODGETTS, R.M. 1998. Measures of Quality and High Performance: Simple Tools and Lessons Learned from America's most Successful Corporations. New York : AMACON. 224 p.

KÖPPEL, A., BÖNING, D. & ABECK, S. [ca. 2000]. How to support the Negotiation of Service Level Agreements (SLAs) for Your Client/Server Application? Karlsruhe : Institute for Telematics, University of Karlsruhe.

KREITNER, R. & KINICKI, A. 2001. Organizational Behavior. 5th ed. New York : McGraw-Hill. 774 p.

LACITY, M.C. & HIRSCHHEIM, R. 1995. Beyond the Information Systems Outsourcing Bandwagon. West Sussex : John Wiley & Sons. 227 p.

LACITY, M.C. & WILLCOCKS, L.P. 2000. Relationships in IT Outsourcing: Stakeholder Perspective. (*In*: ZMUD, R.W., *ed.* Framing the Domains of IT Management: Projecting the Future. . . Through the Past. Ohio : Pinnaflex Educational Resources. p. 355-384.)

LARSON, K.D. 1998. The Role of Service Level Agreements in IT Service Delivery. Information Management & Computer Security, 6/3, 128 – 132.

LAUDON, K.C. & LAUDON, J.P. 2002. Management Information Systems: Managing the Digital Firm. 7th ed. New Jersey : Prentice Hall. 547 p.

LEADBEATER, C. & WILLIS, R. 2001. Mind Over Matter: Greening the New Economy. (*In*: WILSDON, J., *ed.* Digital Futures : Living in a dot-com world. London : Earthscan. p. 16-38.)

LUEVANO, F. (Jr.) 1991. MIS's Role in Service-Level Agreements. (*In*: UMBAUGH, R.E., *ed.* Handbook of IS Management. 3rd ed. Boston : Auerbach. p. 125-134.)

McKEEFRY, H.L. 2001. Service level agreements: Get 'em in writing. [Web:] http://www.cio.com [Date accessed : 6 March 2005].

MILES, M.B. & HUBERMAN, A.M. 1994. Qualitative Data Analysis: An Expanded Sourcebook. 2nd ed. California : Sage. 336 p.

MOUTON, J. & MARAIS, H.C. 1989. Metodologie van die Geesteswentenskappe: Basiese Begrippe. Pretoria : Raad vir Geesteswetenskaplike Navorsing. 303 p.

SILVERMAN, D. 2000. Doing Qualitative Research: A Practical Handbook. London : Sage. 316 p.

SINGLETON, J.P., McLEAN, E.R. & ALTMAN, E.N. 1988. Measuring Information Systems Performance: Experience with the Management by Results System at Security Pacific Bank. (*In* : WATSON, H.J., CARROLL, A.B. & MANN, R.I. *eds.* Information Systems for Management. 4th ed. Illinois : Richard D Irwin. p.319-335.)

TURNER, C. 2000. The Information e-conomy – Business Strategies for Competing in the Digital Age. London : Creative Print and Design. 204 p.

VAN DER MERWE. H. 1996. The Research Process: Problem Statement and Research Design. (*In* : GARBERS. J.G., *ed*. Effective Research in the Human Sciences. Pretoria : Van Schaik. p. 277-291.)

WHITTEN, J.L., BENTLEY, L.D. & BARLOW, V.M. 1989. Systems Analysis & Design Methods (International Edition). 2nd ed. Boston : Irwin. 797 p.

WISNIEWSKI, M. 2002. Quantitative Methods for Decision Makers. 3rd ed. Essex : Prentice Hall – Financial Times. 575 p.