Software Maintenance in a Service Level Agreement: Controlling the Customers Expectations

A.April (1), J.Bouman (2), A. Abran (3), D. Al-Shurougi (4)

1- Bahrain Telecommunications Company, aapril@batelco.com.bh
2- Eindhoven University of Technology, j.j.bouman@tm.tue.nl
3- Univesity of Quebec in Montreal, abran.alain@uqam.ca
4- Bahrain Telecommunications Company, ip@btc.com.bh

Abstract

Software maintenance constitutes an important part of the total cost of the lifecycle of a software application. Some even argue this might be the most important fraction of the cost (50-80 % according to Tony Scott, 75% according to Rand P. Hall, 60% Freedman). The added value of software maintenance is often not perceived by the customers. While the introduction of a new software application clearly shows new benefits, the work being done to maintain an existing application is usually only apparent when the application breaks down or small changes are being implemented (which sometimes also causes some downtime). This results in a negative perception of the software maintenance section. A proposed approach to turn this around is to provide the customer with insights in the activities performed by the maintenance section and to come to a clear agreement on the results and expectations of these activities.

The Service Level Agreement (SLA) originates from the practice of the specifications of results found in the contractual agreements of the large computing centres of the 50's (McBride 1990). Service Level Agreements could be the used by software maintenance for better managing customers' expectations by specifying with the customer what the service results will be. Until a few years ago, this management practice had been limited to operations and support services: the literature search about agreements on Software maintenance turned out some references to Software Maintenance Agreements (for instance Mueller 1994) but most of the agreements reported were limited to helpdesk support, bug fixes and the distribution of new releases. No detailed agreements were reported to include the full spectrum of maintenance services, including the management of the quality of the service.

In this paper the application of Service Level Agreements to the field of Software Maintenance is described, based on the experiences at Batelco. First, key differences between software maintenance and IT development are described, together with the difficulties of viewing software maintenance as an IT Service, and related challenges to tackle them in the design of an SLA. The context at Batelco is presented next, together with a description of the various aspects of the SLA implemented. Lessons learned on the application of SLAs on software Maintenance are presented as well as recommendations for future improvements.

1.0 Agreeing on Software Maintenance

1.1 Software maintenance versus software development

For the readers without work experience in a software maintenance section there is a need to clarify how the management of maintenance activities differ from the management of software project activities.

While project management is organized towards the delivery of a product within a specific time-frame, and a planned project closure date, the maintenance section must be structured to face the day to day work for its clients with a continuous service and, by definition, no closure date. Key characteristics in the nature and handling of small maintenance request have been highlighted in (Abran 1993), for example:

- Modification requests come in more or less randomly and cannot be accounted for individually in the annual budget-planning process;
- Modification requests are reviewed and assigned priorities, often at the operational level – most do not require senior management involvement;

- The maintenance workload is not managed using project management techniques, but rather queue management techniques;
- The size and complexity of each small maintenance request are such that it can usually be handled by one or two resources;
- The maintenance workload is user-services-oriented and application-responsibility oriented.
- Priorities can be shifted around at any time, and modification request of application correction can take priority over other work in progress;

1.2 Software maintenance as a service

The definition of service in the IT domain has been challenging. It is most pragmatic to say that an IT-service is a combination of a service surrounding an IT-object (Bouman 1999). When we look at the characteristics of a service in general, it can be seen that there are a number of aspects generally recognized:

- An emphasis on direct sales to the customer
- More direct contact with the customer
- Service delivered on demand rather than weeks or months later
- Shorter completion time
- Output created as it is delivered
- The product is not always a physical product
- The output cannot always be stored or transported
- Services are less standardized than goods
- Concurrency of consumption and production.

Software maintenance can be viewed as a service, for instance shorter completion time (compared to software development projects) and non-standardization, but also some differences. One of the key differences is the tangibility of the output as perceived by the end user. The result of software maintenance (a new release) can be inspected and tested before it is introduced to the customer. This is where the challenge lies to create awareness on the positive business contributions of software maintenance activities. The SLA might just be the best instrument for that purpose.

1.3 Service Level Agreements Concepts

Even though many publications have presented detailed elements of Service Level Agreements, there is still little known about the key underlying principle in this area. In (Bouman 1999) various concepts have been proposed to structure the knowledge about the structure and use of an SLA. One such concept is the principle that an agreement should be based on results rather than on effort. For example, in many instances there are attempts to describe the results of an IT-service in terms of quality levels to be achieved for the IT-object and with the expectations that this is to become the basis of the SLA. It has been observed in practice that whereas there is not yet a consensus on a proper description of results, it is still necessary to fall back to effort based agreements.

Another concept proposed in (Bouman 1999) is an inventory of guidelines templates for the specification of a SLA. One of these guidelines is to know the parties dealing with the service. Additionally SLA should clarify the expectations/requirements of each service. In the context of an internal agreement on software maintenance services between the maintenance section and their customers the following list of expectations and requirements have been observed:

A) A customer wants to concentrate on his business and expects a homogeneous service from his IT organization. The result of this homogeneous service for the customer is the ability to work with a set of information systems. The way this service and these systems are composed are of less interest to the customer. This attitude should be reflected in the agreements made with the customer by the IT organization, so the performance and quality of information systems should be described not the quality of the components.

B) The maintenance section wants to give the customer a realistic image of what he should expect and, in the rare case of an SLA where there is a notion of bonus/penalty, they want to make sure that commitments are met. This means insights are required on the services to be provided and the quality of the object which has to be serviced. Hardware manufacturers have done this for years in the form of warranties and service contracts on their products. However hardware components tend to have a higher degree of standardization making maintenance easier. There is more empirical information on breakdown of components and replacement is easier. Furthermore during the development of

software the focus tends to be on functionality rather then stability and maintainability.

In the next section the software maintenance services and activities will be analyzed for their implications on defining Service Level Agreements.

1.4 Software maintenance activities

The ISO standard (ISO/IEC 14764) for software maintenance divides software maintenance in four categories (Fig. 1). The two dimensions for these categories are the timing of the change (proactive or reactive) and the goal of the change (correction or enhancement). Using these dimensions we will discuss the various maintenance categories and the initial agreements made at Batelco on these categories.

	Correction	Enhancement
Proactive	Preventive	Perfective
Reactive	Corrective	Adaptive

Figure 1: ISO software maintenance categories

Batelco IT software Maintenance and Support organizations currently work on software systems developed internally with access to source code. Progressively, because of the planned acquisition of many turnkey software solutions (SAP/R3, eCommerce and IP-billing) the maintenance services will have to adapt: a third party then owns the source code and a maintenance and support contract with a third party is required to formalize the third party supplier Service Levels. The Maintenance and Support section then becomes the main interface to this supplier, which in turn Guarantees application services levels in the area of functionality corrections and enhancements.

1.4.1 Correction dimension

The activities on the correction dimension are aimed at bringing the software product at the quality level in conformity to the specifications. The fact that this activity is necessary is often caused by hidden faults in the software. These faults not uncovered by the development process should be dealt with by the supplier supplying the software (initially as a form of warranty and, after the end of the warranty, in the form of a software maintenance contract). According to the result-based concept this would mean that

agreements on corrective activities should be in terms of the maximum hindrance a customer might have to deal with concerning the failure of the specific Information system.

In the IT domain, there are of course various challenges to be addressed when attempting to implement such kind of agreements. The first problem is that the influence of the quality of the software on the whole information system is hard to assess formally. For example, an average client-server environment consists of 10 IT-components of which 3 to 4 are software related. Not only is the quality of the individual components often unspecified (especially for the software components), the quality implications of their interaction are often based on undocumented service assumptions. Therefore, the agreements proposed by the IT-organization often fall back to an effort basis rather than rely on a result -basis.

In creating an SLA it is important to represent both parties involved and to come to a mutually beneficial agreement. Research on the field of service quality has shown that customers don't have a singular expectation but rather a field of tolerance (Zeithalm 1993). This can be used to specify service levels (a service level guarantee and a service level target). By measuring the actual service level information can be gained about the quality of the (software) components and their interaction.

Another growing necessity is the use, and alignment to internal SLA, of underpinning contracts. The modern IT-organization uses more and more third party software and services. When the maintenance of software is provided as a thid party service the service level provided by the supplier towards the customer (in this case the maintenance section) should be aligned to enable the maintenance section to guarantee the total service quality for the application. This approach is also necessary internally with other parties within the IT-organization to align maintenance activities and ensure end to end service to be coherent.

1.4.2 Enhancement dimension

The activities in the enhancement dimension are even more difficult to define precisely the in terms of results rather than activities. Proactive enhancements (perfective maintenance) are usually initiated by the IT organization

themselves to enhance the maintainability of the software and improve their software maintenance effectiveness. This type of change does not directly influence the information system functionality and should therefore be highlighted as such in the SLA. The enhancements required by the customer do have a large impact on his perception of the software section value and therefore can not be easily ignored. It is difficult to predict the number and content of the enhancements that will be requested by the customer during the agreed upon service period. Three approaches on how to deal with this uncertainty and mostly a combination are illustrated here:

1- An enhancement plan for the SLA period. The SLA can include the descriptions of the major enhancements which are foreseeable for the service period, such as for example an organizational change or the Euro introduction. However, most smaller enhancements can not be predicted far ahead and in most maintenance contracts the duration of the information system service period is 2 to 10 years;

2- A description of capacity. The available work capacity the provider has planned for performing adaptations during the server period can be described in the SLA. The transient nature of work capacity should be taken into account. This can be done by specifying the capacity for each week. During the service period for each enhancement requested by the customer an estimation of the needed capacity is given by the provider on basis of which the customer can decide whether to commission the change or not.

Apart from describing the capacity in work hours it could be possible to use the dimension of function points (Abran, 1995; Maya 1996) to specify the capacity. The problem with this approach is that the concept of function point might be rather vague to the maintenance section and sometimes to customer themselves. Only higher maturity organizations use such techniques.

3- A list of specific enhancements. This list is composed of end user issued modification requets where a priority has been assigned. In this list enhancements are described and for each enhancement the IT capacity needed is estimated and specified. Examples of the content of a specific enhancement list are the addition of a

database field or the design of a new menu or report. Without the availability of previous productivity studies, it is very difficult (and highly risky) to relate high-level generic enhancements to the capacity needed. This approach is positioned somewhere in the middle of the two previous ones.

2.0 The Batelco SLA's

2.1 The organization

When it comes to application availability, one section of Batelco IT does not "own" the entire set of end-to-end components. At Batelco, as for many other IT organizations, end users workstation view of applications are managed by at least six (6) IT sections across the IT organization and within two major divisions, each division with different objectives, responsibilities and skill sets, as illustrated by fig 2.
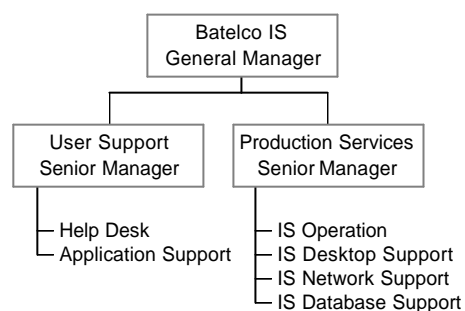

Figure 2 : Batelco IT Support Structure

The six (6) IT sections must thoroughly understand service failure from an end user point of view. This understanding must also reflect the third party warranty and support contract terms to ensure that an Integrated Problem Resolution process view is promoted. It is that full end to end view that we are trying to capture in the renewed Service Level Agreement where each player in this service chain has clear understanding of his role and responsibility to achieve a stated application level of service.

2.2 Some problems with the services and agreements

When an application failure occurs, users typically call the central Help Desk (also called

level 1 support, which is mostly workstation access level help). If the Help Desk cannot address the issue it refers the problem to the application Maintenance (also called level 2 support). Investigation may result in identifying an infrastructure or third party product failure. When it is not an application or high level environment issue, the call is referred to the Service Management support (also called level 3 support) which looks at in depth desktop issues, telecommunications, middleware, Database Management Systems, Operating System, Disk and other operating system, base software and hardware related issues. Service Management may involve third parties (level 4 support) for some third party supported hardware or base software.

Batelco IT client/server applications are configured in quite the same way as common industry practices. This results in a high number of components and potential points of failure. For example, our phone center cash register application service is dependent, at least, on eleven (11) individual components, as illustrated in figure 3. If each component is to be available 99.9 % of the time we still have a resulting workstation net value that drives down the availability to only 97.8 percent of the time. Downtime for the customer facing cash register application would approach 8 hours/month, which is unacceptable for Batelco critical applications. Although distributing computing may deliver significant business value, it also challenges IT to optimize its Problem Resolution Process or it will fail in bringing a stable operational environment to its customer which is something they now have with their centralized mainframe applications.

Component	Availability Objective	Workstation Net Availability Objective
Client Hrdwr	99.9%	99.9%
Client O/S	99.5%	99.5x99.9=99.4%
Client Middlwr	99.9%	99.9x99.4=99.3%
Client Applic.	99.9%	99.9x99.3=99.2%
Cable Hub/Router	99.9%	99.9x99.2=99.1%
Network Server	99.8%	99.8x99.2=98.9%
Net. Server O/S	99.9%	99.9x98.9=98.8%
Database System	99.8%	99.8x98.8=98.6%
Server O/S	99.6%	99.6x98.6=98.2%
Server Middlwr	99.8%	99.8x98.2=98.0%
Server Applicat.	99.8%	99.8x98.0=97.8%

Figure 3 : **Cumulative effect to availability**

As an important first step to provide an integrated view of service we have chosen to clarify and include the application maintenance service in the SLA's.

3.0 The SLA's Improvement Program
3.1 The current SLA's

The current Batelco IS SLA's have limited scope and covered the following aspects of service:

- Online Availability timings and objective;
- Batch Processing timings and objective;
- Maintenance window timings;
- Outage resolution times by priorities;
- Escalation procedure by priorities;
- Service Performance review items

They also have not been deployed fully to provide customers with a better understanding of services and how they are provided. Finally no service level reporting is in place.

When compared to the set of concepts presented in section 2, it becomes obvious that the current SLA's suffers from most of the problems identified (Bouman 1999) such as:
- Specification of effort rather than specification of results;
- Unclear service specification;
- Incomplete service specification;
- Insufficient cost management.

This comparison highlights the need for the current Batelco SLAs to be significantly improved to reach a 'negotiated agreement' designed to create a common understanding about services, priorities and responsibilities' [3]. It will also be used as a 'communication tool' and a 'conflict-prevention tool' (Karten 2000)

3.2 Improved SLA structure

In this section we present the resulting enhanced SLA structure. We have opted to reflect the needs of heterogeneous business users with common service needs (McBride 1996) where only few (four) SLA's are created and multiple applications are described for each Major customer. We present the text developed for our Residential, Business and Major Customer Business Unit Client. It contains the following eight (8) sections:

1. Purpose

3.2.1 Introduction text of the new SLA

This Service Level Agreement describes the computer systems support services to be provided to Residential, Business and Major Customer Business Units by Batelco IT. These services are provided as part of the Annual IT Work Program to the Batelco Residential, Business and Major Business Unit.

This Service Level Agreement also contains a number of Schedules, listed in Section 5.18 hereafter, referred to for specific purposes such as application/platform Schedules which describes the specific requirements for the Maintenance and Support of a specific application/platform.

3.2.2 Client Identification/ Responsibilities

The Residential, Business and Major Business Unit is the recognized owner and prime client for the applications/ platforms covered in this Service Level Agreement. Residential, Business and Major is responsible for designating a work program coordinator to interface with the IS account manager- Residential, Business and Major. This coordinator is responsible for the management of this Service Level Agreement.
The coordinator is responsible for the integration of the needs of each section of its organization. Residential, Business and Major is also responsible for the internal funding of fees identified in Section 7. Annual IS SLA Fees.

In accordance with funds available and for each application/platform, the coordinator is responsible for the identification of the services required and for their level of priority as defined in the priority scale, schedule D of this Service Level Agreement. The coordinator is also responsible for the appointment of an application prime to manage the Modification Request process.

The client responsibilities defined in this section are however subject to their specific responsibilities as provided in the detailed

"Tasks and Responsibilities" in Schedules B 1-3 to this Service Level Agreement.

3.2.3 Batelco IT Identification/Responsibilities

User Support, in Batelco IT is the recognized prime of the support services defined in this SLA and covered under this Service Level Agreement.

User Support is responsible to offer and supply the services defined in the Service Level Agreement and to appoint a Service Manager to such effect. This Service Manager will be responsible to interact with the Account Manager - Residential, Business and Major to understand its business needs and enable User Support to anticipate and respond to changing business requirements in a timely fashion.

User Support will maintain a professional and competent workforce in support and maintenance. User Support will be responsible for the coordination across all IT units of the performance criteria and service levels detailed in Schedules E of this Service Level Agreement. The Service Manager will provide, on a timely basis, relevant performance results measured against service commitments specified in the SLASchedules.

3.2.4 Description of Services covered

This section identifies and describes the application/platform support services to be provided by User Support under this Service Level Agreement.

The following sub-sections are structured by service category, as follows:

1: Service Level Agreement Management
2: Help desk
3: Application Management and Support
4: Operation Management
5: Problem Management
6: Additional Services

Additional attachments are developed to reflect the individual application profile, volumes, number of users, criticality, service performance criteria and related third party agreements :

Schedule A :List of Applications per priority
Schedules B 1-3 :Application Schedules
Schedule C : Budget Details
Schedule D : Escalation and priorities

We have used Abran 1993 to describe the application maintenance description of services.

4.0 Service # 3: Application Management and Support

The following text has been developed to describe the Batelco maintenance section services: Application Management and Support involves two distinct functions. The first function is an application level support (also called Level 2 Support) which is identified here as Corrective Maintenance. The second function, which is handled by raising a Modification Request, provides for: Application Modification Request management, assessments, classification into preventive/perfective/ adaptive maintenance, and the coordination of release and configuration management associated with the changes to an existing application in the production environment. Application Modification Request will be accepted when the total effort is estimated to be within 5 work days. Larger modifications may be routed to the Program manager to be assessed as a major enhancement or software project. This service is subject to the Modification Request process and is further described as follows:

4.1 Modification Request Management/Assessments

This service will provide Modification Requests assessments associated with Maintenance and Support services. This service will provide planning, control and management of changes namely to application/platforms, to production environment and disaster recovery procedures in order to assure client continuity of operations where change is initiated. These change assessments do not extend to major enhancements or development projects that are excluded under this Service Level Agreement.

4.1.1 Corrections - Corrective and Preventive Maintenance

This service performs reactive and preventive maintenance on existing applications. Modification Requests may be initiated by User Support, Client, and/or Account Manager - Residential, Business and Major.

4.1.1.1 Corrective maintenance

This service will provide application failure assessments and corrections associated with a support call. It is also referenced as Level 2 Support. This service relates to the elimination of an error condition in a software application which is impacting the operability of a production system. This service covers all associated coding, testing, change control, software distribution, documentation, job rerun, and file recovery required to fix the error condition. In summary, these activities include:

- Provide immediate assistance to Level 1 Support (Help Desk);
- Assist Level 1 in problem (trouble ticket) prioritization;
- Provide immediate support and priority on application failure;
- Providing application recovery from outages;
- Fixing job failures;
- Fixing production system problems;
- correcting erroneous data, data fill and flow problems.
- Provides feedback and status report to level 1 until problem is resolved and problem is closed;
- Use surveillance and data collection tools to gather application level fault information;
- Analyze application fault information and conduct root cause analysis;
- Disseminate information to help prevent problems and ensure effective client support;
- Discuss and review with application developers, system designers operations personel and third party suppliers future upgrades, changes and quality standards;
- Coordinate problem resolution with Help Desk, Operation Management and Service Partners;
- Provide technical recommendations in negotiations involving service/technology enhancements;

4.1.1.2 Preventive maintenance

This service provides activities to prevent failure of end users application by advanced detecting

and correcting of latent errors. Examples include:

- Correcting edit rules to prevent recurrence of failures;
- Implementing and using monitoring tools to detect problems before they impact the client.

4.1.2 Enhancement – Adaptive and Perfective Maintenance

This service performs proactive maintenance on existing applications. Modification Requests may be initiated by User Support, Client, and/or Account Manager - Residential, Business and Major.

4.1.2.1   Adaptive maintenance

Adaptive maintenance provides activities required to adapt the system to a change in the current hardware, operating environment, or volume where no new or changed functionality is required. Examples include:

- Compiler changes;
- Hardware upgrades;
- Media conversions;
- Utility changes;
- Making adjustments to accommodate changes in load;
- Evolving the "Application Recovery Manual" and testing the disaster recovery plans.

4.1.2.2   Perfective maintenance

Perfective maintenance provides quality, maintainability and operability improvements for a specific application or the wole application portfolio with an objective of reducing current level of resource consumption. This does not include any functional changes to the application. Examples include:

- Optimizing code and resources;
- Restructuring code logic;
- Clarifying and improving system documentation.

4.1.3 Release Planning and Management

This service proactively manages the size, timeline and activities associated with introducing an application software change into production environment. Examples include:

- Release planning which involves analyzing requirements versus schedule of release window;
- Release evaluation which involves the evaluation summary based on impact assessment, ongoing benefits, risk assessment and strategic considerations;
- Release management which involves application validation and certification, integrated validation testing, acceptance and soak testing;
- Release sizing which involves weighting the impact point value.

5.0 Lessons Learned and Future directions

Based on the implementation of this improved SLA in a pilot group, we can report on the following lessons learned:

- Missing management information and missing baselines on existing applications makes it very challenging to establish detailed definition of services;
- Existing Third party agreements which are not integrated with the new SLA approach demonstrate immediately the weakness of the existing service agreement.

furthermore, it has been challenging to motivate software maintenance personnel to implement these changes: they are much more familiar with the previous technical perspective of software applications, and defining maintenance services performance from a client perspective leaves them much less in control with respect to the measurement of their own performance.

These lessons learned will help deploy current improvements and planned further work. Similarly, our Release 2 of SLA's will widen its scope to include additional services, such as Help Desk Services.

6.0 References

[1] Abran, Alain; Nguyenkim, H.: Measurement of the Maintenance Process from a Demand-based Perspective *Journal of Software Maintenance: Research and Practice*, Vol. 5, no 2, 1993, p. 63-90.

[2] Abran, Alain; Maya, Marcella, "A Sizing Mesure for Adaptive Maintenance Work Products," *International Conference on Software Maintenance - ICSM-95*, Opio, France, 1995.

[3] Bouman, Jacques; Trienekens, Jos; van der Zwan, Mark, Specification of Service Level Agreements, Clarifying Concepts on the Basis of Practical Research, Procedings of Software Techology and Engineering Practice '99, 1999.

[4] Freedman, D.H, Programming without tears, High Tech. 6, 4 (1986), 38-45.

[5] Hall, Rand P, Seven Ways to Cut Software Maintenance Costs, Datamation; Barrington; Jul 15, 1987.

[6] Karten N, Establishing Service Level Agreements. Karten Associates, http://www.nkarten.com/sla.htm.

[7] Maya M; Abran A; Bourque P; "Measuring the Size of Small Functional Enhancements to Software," *6th International Workshop on Software Metrics*, University of Regensburg, Germany, 1996.

[8] McBride D, 'Successful Deployment of IT Service Management in the Distributed Enterprise', 1996, Hewlett-Packard Web site: http://www.hp.com/go/rpm

[9] McBride D, 'Service Level Agreements: A Path to Effective System Management and Application Optimization', Hewlett-Packard Professional, 1990 can be found on Web site: http://www.hp.com/go/rpm

[10] Mueller, Bob, Software maintenance agreements poised for change, Systems Management 3X/400, 1994.

[11] Scott, Tony; Farley, Dennis; Slashing Software Maintenance Costs ; Business Software Review; Indianapolis; Mar 1988.

[12] Zeithalm, Valerie; The nature and determinants of customer expectations of service; Journal of the Academy of Marketing Science, Vol. 21, Winter 1993.

[13] ISO/IEC- FCD 14764 Software Engineering – Software Maintenance, International Standard, 2000.