# IT Service Incidents Prioritization driven by Business Objectives

*C. Bartolini, M. Sallé, A. Boulmakoul*
*HP Laboratories*
*1501 Page Mill Rd.*
*Palo Alto,USA*

*S. Sabiani, J.M. Bronoel*
*HP France*
*950 Route Des Colles, Les Templiers*
*Sophia-Antipolis, France*

*{claudio.bartolini, mathias.salle, stephane.sabiani, abdel.boulmakoul, jean-marc.bronoel}@hp.com*

**Abstract**

In this paper we describe a solution for the management of IT service incidents. In particular, the solution deals with prioritization of incidents driven by an enterprise's business objectives. The solution integrates the Service Level Management component of Openview Service Desk (OVSD-SLM) with the Management by Business Objective (MBO) technology being developed at HP Labs. We give a thorough description of the algorithms and information models used for the prototype, and briefly describe the architecture of the integrated solution.

**Keywords**

Service Level Incident Management, Openview, Service Desk, Service Level Management, Decision Support

## 1. Introduction

As defined in the Information Technology Infrastructure Library (ITIL) [1], Service Level Management ensures continual identification, monitoring and reviewing of the optimally agreed levels of IT services as required by the business. Most targets set in a Service Level Agreement (SLA) are subject to direct financial penalties or indirect financial repercussions if not met. It is therefore critical for this management process to flag when service levels are projected to be violated in order for an IT organization to take proactive actions to address the issue. To this extent, ITIL defines an incident as *a deviation from the (expected) standard operation of a system or a service that causes, or may cause an interruption to (or a reduction in) the quality of the service*. The objective of Incident Management is to provide continuity by restoring the service in the quickest way possible by whatever means necessary (temporary fixes or workarounds). Example of incidents may be degradation in the quality of the service according to some measure of quality of service; unavailability of a service; a hardware failure; the detection of a virus.

In the incident management process it is of fundamental importance to classify, prioritize and escalate incidents. Priority of an incident is usually calculated through evaluation of impact and urgency. However, these measures usually refer to the IT domain. The central claim of our work is **that in order to achieve the strategic alignment between business and IT, the enterprise needs to drive incident prioritization from its business objectives**. This starts from evaluating the impact that an incident has at the business level, and its urgency in terms of the cost to the business of not dealing with it in a timely fashion.

In this paper we describe a solution for the management of IT service incidents. In particular, the solution deals with prioritization of incidents driven by an enterprise's business objectives. The solution integrates the Service Level Management component of HP Openview Service Desk (OVSD-SLM) with the Management by Business Objective (MBO) technology being developed at HP Labs.

The paper is structured as follows. In section 2 we give a thorough description of the MBO algorithms and information models. In section 3 we briefly describe the architecture of the integrated solution. In section 4 we touch on future work and move on to the conclusions.

## 2. MBO incident prioritization

The current research on Management by Business Objectives (MBO) is aimed at developing solutions for IT management driven by an enterprise's business objectives. We approach this goal through the development of a methodology for decision support in IT management. The methodology is based on a reasoning engine that solves the following class of decision problems: it computes the alignment to objectives that is expected for each of the possible given management options, or course of action aimed at managing the IT delivery systems. The engine is able to monetize the measure of alignment thus derived and use the monetization value together with other information on the cost of carrying out the respective course of action to ranks the available options. On ranking the options, it returns a suggestion on what course of action to take, substantiated by the evidence that it has for assessing the alignment with respect to the business objectives.

In the development and the deployment of the solutions, we follow the principle that the **cost of modeling should be kept low**; so that it is easily offset by the benefit obtained from the decision support.

For each of the various IT management domains the generic decision problem is specialized into a decision problem that pertains to that domain. This requires a mapping of the domain specific concept onto the generic concepts that are defined in the MBO information model.

The problem solved by the MBO component in the incident prioritization solution is to assign priority levels to a set of service level degradation incidents so as to maximize the alignment with a given set of business objectives. The following sub-section introduces some of the lexicon that we will use.

**Service level degradation incidents**

A Service Level Agreement (SLA) is contracted with a customer and contains a set of

Service Level Objectives (SLO). Each SLO specifies an acceptable range of values for a given system metric, through the definition of a violation threshold for the SLO. A penalty cost is associated to SLO violation, which occurs when the metric value surpasses the violation threshold. Besides the violation threshold, a jeopardy threshold is also specified. Metric values are obtained by probes deployed by the management system and monitored by a monitoring component. A degradation of service level incident for an SLO occurs when the monitoring component reports on a metric value surpassing the jeopardy threshold for the SLO. An incident management system collects and organizes the information on the degradation of service level incidents by assigning priority values to them together with other information on the lifecycle of the trouble ticket associated to the incident. The problem that we solve is to suggest how to deal with the incidents so as to minimize the expected cost of violation of the SLOs. In this work we only consider incidents generated on detection of service level degradation or violation, although the general techniques that we present are more widely applicable.

To solve the incident prioritization problem, we operate on the following steps:

- Compute the likelihood of violation of an SLO in function of the time taken to close a jeopardy incident
- Compute the alignment with the business objectives in function of the likelihood of violation
- Prioritize the incidents in function of the alignment with the business objectives

### 2.1 Likelihood of SLO violation as a function of the time of closure of the service degradation incident

We make the assumption that the IT manager is required to specify a time value that represents the expected time that it will take for the system to move from the jeopardy state to the violation state if no measures are taken (expected time from jeopardy to violation). We assume an exponential distribution of the time from jeopardy to violation if no corrective actions are taken. The parameter of the exponential distribution, $\lambda$, is defined as the inverse of the expected time from jeopardy to violation.

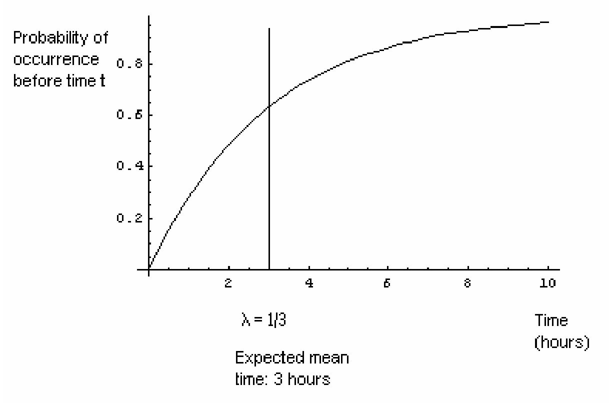

**Figure 1:** Likelihood of SLO violation as a function of the time of closure of the service degradation incident

The plot to the right side represents the cumulative distribution function of the distribution of probability associated with the time from jeopardy to violation. The analytic form of the curve is given by the equation $p = 1 - e^{-\lambda t}$ , where p represents the probability of violation if the incident is closed after t from its starting time. In the example given, for a value of $\lambda = 1/3$, corresponding to an expected mean time of 3 hours, the probability of a violation occurring within 4 hours from the jeopardy alarm if no measures are taken equals to $1\text{-}e^{-4/3}$=76%.

## 2.2 Alignment with the business objectives in function of the likelihood of violation

Before going in depth into the algorithm for computing the alignment of a prospective instance of incident prioritization instance with the enterprise business objectives, let us introduce the information model that underlies the representation of the business objectives in our system.

### 2.2.1 MBO Information model

The MBO Business Objectives information model (Figure 1) is articulated around a set of key concepts: Objectives, Key Performance Indicators (KPI), and Perspectives. The terminology used in this information model borrows where possible from the lexicon of the COBIT [2] (Common Objectives for Information and related Technology) framework and from Balance Scorecard [3]. COBIT is a framework addressing the management's need for control and measurability of IT. It provides a set of tools and guidelines to assess and measure the enterprise’s IT capability for the principal IT processes. Balance Scorecard is a tool for management that enables organizations to clarify their vision and strategy by capturing them into actionable objectives.

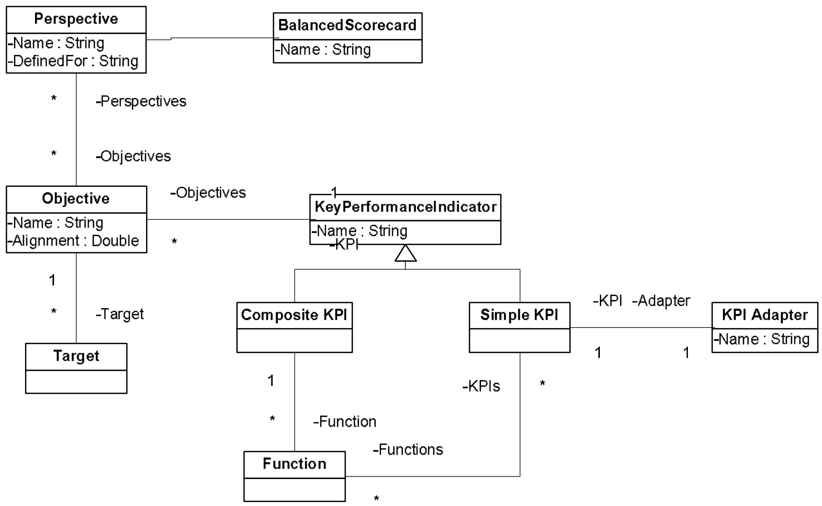


**Figure 2:** The MBO business objectives information model

In the remainder of this section, we briefly describe the principal concepts defined in the MBO Information Model.

**Objectives**

COBIT introduces Key Goal Indicators (KGI) as measurable indicators of the business objectives. In our model the objectives are represented through KGIs by expressing one or more target values[1] over a key performance indicator, or KPI – see below.

**Key Performance Indicators**

As defined by COBIT, Key Performance Indicators (KPI) are measurable indicators of performance of the enabling factors of IT processes, indicating how well the process enables the goal to be reached.

**Perspectives**

Perspectives are used to bundle objectives together that concern a certain angle of the business. The concept of perspectives is borrowed from the balanced scorecard [3]. A balanced scorecard defines four perspectives: Financial, Customer, Business Process and Learning and Growth. Our model defines a perspective as a first class object, not limiting its usage to the traditional balance scorecard model. Perspectives do not

[1] In all the examples given in the remainder of this paper, only one target region per objective is defined, in order to help the flow of the discussion. An example of objective with multiple targets is one defining a first threshold of acceptability and a further threshold that represents a *stretch goal*. Example: revenues for the quarter must increase 15% year over year, with a stretch goal of 20% increase. When multiple targets are defined for an objective, the measure of alignment with the objective needs the definition of *importance weights* for all the target regions.

represent a partition over the set of objectives defined. An objective can belong to more than one perspective.

An example of an objective defined through the model is “the aggregate service revenue generated over the current three-month period must be above 100,000 $”. This is modeled in MBO by defining a KPI ɩ representing the aggregate cost of SLA penalty paid over the current three month period, represented by a dollar amount. The target of the objective is the region in the KPI space characterized by the inequality ɩ < 100,000$.

An example of perspective is a financial perspective, containing objectives such as the one listed above on the aggregate cost of SLA violations, or an objective that defines a target over a KPI representing the aggregated revenue generated in a given time period. A customer perspective could contain objectives defining targets over some KPIs representive quantitative measures of the customer satisfaction (measures of TCE: total customer experience), and so on.

MBO assignes importance weights to objectives and perspectives. The weights are used by the MBO engine to compare utility values of different objectives. The weight assigned to one perspective is propagated down to the objectives belonging to that perspective, as exemplified in Table 1.

| **Perspective** | **Financial** | | **Customer** |
|---|---|---|---|
| weight | 80% | | 20% |
| **Objective** | Aggregated revenue in three month period | Aggregated cost of penalties for SLA violation in three month period | Total customer experience |
| weight | 40% | 60% | 100% |
| adjusted weight for perspective | 32% | 48% | 20% |

**Table 1:** Objectives, perspectives and importance weights

#### 2.2.2 Alignment with business objectives as utility of the management options

We define the alignment with a given business objective as **the measure of the likelihood –** *given the best knowledge about the current situation* **– that the objective will be met**.

Let’s take the sample objective given in the previous section: “the aggregate service revenue generated over the current three-month (KPI) period must be above 100,000$ (target)”. Let us suppose that 2 months into the period, the aggregate revenue figure amounts to 60,000$. Suppose an estimation is made that the revenue for the last month is uniformly[2] distributed in the interval [0$, 45,000$]. The likelihood of meeting the objective is equal to the likelihood of posting a top line figure for the last month of the quarter in excess of 100,000$ - 60,000$ = 40,000$, that amounts to (45,000$ - 40,000$) / 45,000$ = 11.11%. This measure is taken as the alignment with the business objectives.

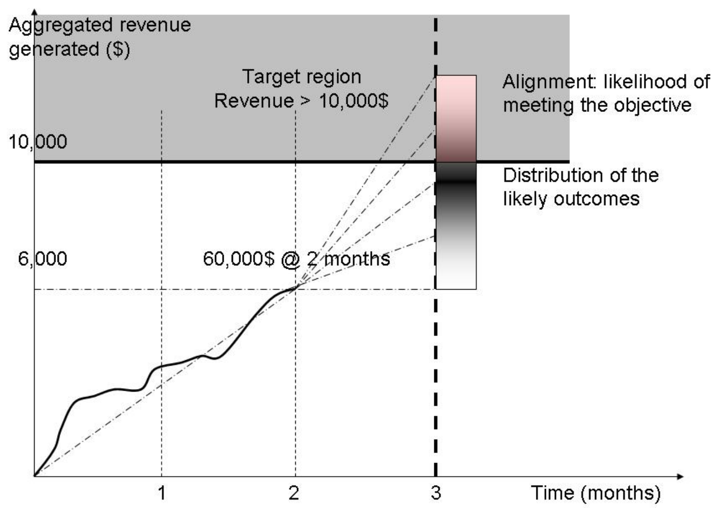


**Figure 4:** Alignment as likelihood of meeting the business objective

From the definition of alignment used here, and the kind of business objectives that we consider, it follows that our method requires some estimate of the future value of the KPIs. The estimate will be captured as a distribution of probability over the relevant KPI spaces. In the rest of the document, we refer to such an estimate as a likely outcome. An outcome is characterized by the distributions of probability over the KPI spaces that it entails.

The simple method that we have chosen for this system predicts the mean value of the KPI at the end of the period simply by extrapolating its current value. The forecasted value of the KPI is considered to be normally distributed with mean calculated as above and variance set at a sensible customized value (for example the square of 5% of the mean value). For the example given above, when the revenue KPI reads 60,000$ two months into the period, we will forecast its value at the end of the three month period to be normally distributed with mean 90,000$ and a variance of $(4{,}500\$)^2$, and therefore characterized by

$$pdf_{default}(kpi_{revenue}) = NormalDistribution(90{,}000\$, (4500\$)^2)(kpi_{revenue})$$

2 Used here for simplicity of calculation.

### 2.2.3 Forecasting the effect of likely SLO violations

Our goal in this section is to provide a mapping between the closing time of a service incident and the measure of alignment with the business objectives just defined. In order to do so we need to evaluate the impact that a likely SLO violation has on the KPIs that the objectives are based on. In the current version we use deterministic functions to express that dependency. For example, when considering the effect of a likely SLO violation on the aggregated cost of penalties KPI, we will represent it through the function

$$f_{impact}(kpi_{penalty}) = kpi_{penalty} + penalty(SLO)$$

The formula is intended to mean that when a SLO violation occurs, the value of the KPI measuring the aggregate cost of penalty is increased by the penalty relative to the SLO violation. Similarly, considering a customer related KPI that measures the number of violations experienced by all customers in a period of time, we write

$$f_{impact}(kpi_{violation}) = kpi_{violation} + 1$$

The outcome that follows an SLO violation is therefore obtained by composing the effect of the SLO violation with the default outcome as indicated by the following formula

$$pdf_{violation}(kpi) = pdf_{default}(f_{impact}^{-1}(kpi))$$

We now determine the likely outcome of closing an incident by a given time. We have already determined the likelihood of SLO violation $\lambda$ in function of the time taken to close the incident. The likely outcome is given by the combination of the default outcome if the violation doesn't occur with the modified outcome if the violation does occur.

$$pdf_{combined}(kpi) = (1-\lambda)\, pdf_{default}(kpi) + \lambda\, pdf_{violation}(kpi)$$

The method followed is depicted in Figure 5.

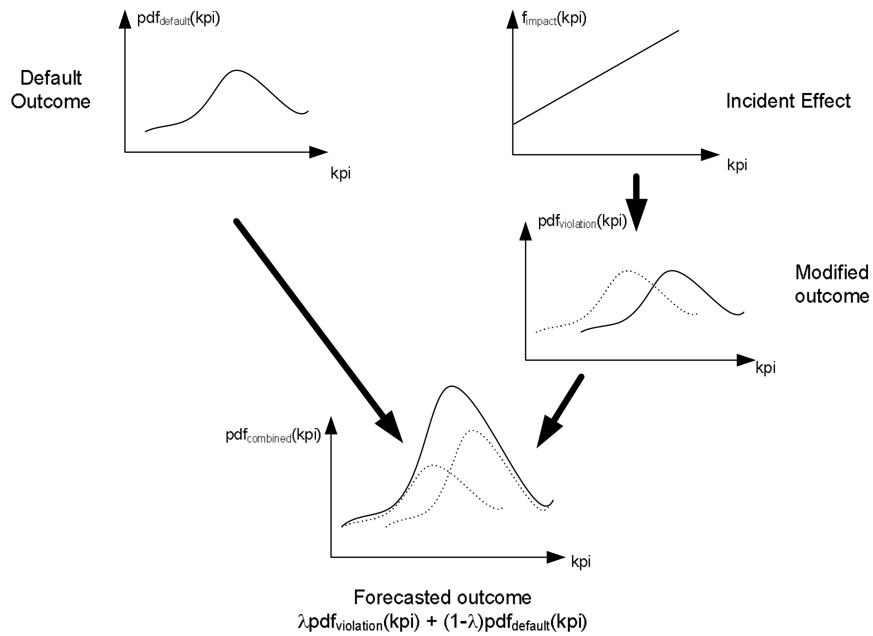


**Figure 5:** Forecasting the effect of SLO violations on the KPI space

### 2.2.4 Computation of alignment

To compute the alignment of the forecasted outcome with a given business objective, we simply integrate the probability density function for the outcome within the target region defined by the objective in the KPI space

$$alignment_{objective} = \int_{kpi \in Target(objective)} pdf_{combined}(kpi)dkpi$$

The alignment of the outcome with all the business objectives is finally obtained by summing the contribution of the alignment with each of the business objectives, each taken with their own weight.

$$alignment = \sum_{\forall objective} weight_{objective} \cdot alignment_{objective}$$

It can be noted that because of the linearity of the combination operated above, we also could first independently compute the default alignment and the alignment in case of violation; and later linearly combine them with the likelihood that the violation will in fact happen. The end result will not change, and in this way the calculations are much simpler and faster to carry out, since it is much simpler to deal with scalars than with probability density functions, and the number of integration required is dramatically reduced.

### 2.2.5 Worked example

Let's consider a service degradation incident that relates to an SLO for which the expected time from jeopardy to violation is three hours. As we saw in section two, the likelihood of SLO violation expected from closing the incident in four hours is given by $1 - e^{-4/3}$ = 76%. The penalty associated to violating the incident is set at 1,000$.

Let's assume that two simple business objectives have been defined. The first objective states that the aggregate cost of penalty for SLO violation in a three month should be less than 10,000$. The second objective states that the total number of SLO violations for all customers in a three month period should be less than 15. The objectives are deemed to be equally important, therefore each carrying an importance weight of 1/2.

Two months into the period, the current readings of the KPIs are 6,000$ in penalties paid for 10 SLO violations. For simplicity's sake and without loss of generality we here assume that the forecasting module predicts a default outcome characterized by uniform distributions with extremes [7,500$, 10,500$] for the penalty KPI and {13, 14, 15, 16, 17} for the violations KPI.

The alignment with the business objectives entailed by the default outcome is computed as the probability of meeting the objectives given the expected default outcome. For the first objective we obtain an alignment measure of 5/6 (such the probability that the penalty KPI will be below 10,000$ given that it's distributed uniformly in [7,500$, 10,500$]. For the second objective it's obvious to observe that the alignment measure is 2/5. The alignment with all business objectives is therefore 1/2*5/6 + 1/2*2/5 = 31/60 = 0.517.

The effect of the likely SLO violation on the KPIs is to increase the penalty figure by 1,000$ and the number of violations by 1. This would therefore define an outcome characterized by uniform distributions with extremes [8,500$, 11,500$] for the penalty KPI and {14, 15, 16, 17, 18} for the violations KPI. If the violation were to occur, that would entail level of alignment of 1/2 and 1/5 for each objective respectively, and therefore an alignment with all business objectives of 1/2*1/2 + 1/2*1/5 = 7/20 = 0.350.

Since the violation is expected to occur with a 76% likelihood, the measure of alignment for the combined case becomes 24%*0.517 + 76%*0.350 = .390.

## 2.3 Incident prioritization to maximize alignment with business objectives

Once the business impact of the incidents has been computed, we are faced with the problem of prioritizing them so as to minimize the total impact on the business. Our system requires the use of a *priority scheme*. Together with the definition of a set of priority levels that are used to classify the incidents (defined by the ITIL [1] guidelines for incident management), we require the user to express constraints on what are the acceptable distributions of incidents into priority levels. For any priority level the users can either force the incidents to be classified according to some predefined distribution (e.g. 25%-30% high, 40%-50% medium, 25%-30% low), or define a minimum and maximum number of incidents to be assigned to each priority level. Our method finally requires an *expected maximum closing time* for the incidents that are assigned to a certain priority level.

### 2.3.1 The Incident Prioritization Problem

We here present a mathematical formulation of the incident prioritization problem as an instance of the generalized assignment problem. The generalized assignment problem is an integer optimization problem that is well studied in the operation research literature and for which very efficient algorithms have been developed.

Suppose we are required to prioritize between $n$ incidents $i_1..i_n$ into $m$ priority levels $p_1..p_m$. We introduce a variable $x_{jk}$, $j=1..m$, $k=1..n$ that assumes the value $x_{jk}=1$ if the $k^{th}$ incident is assigned to the $j^{th}$ priority level and $x_{jk}=0$ otherwise.

By observing that the alignment of each incident can be calculated depending on what priority level it is assigned to, if $t_j$ is the expected time of completion for incidents assigned to priority level $j$, then obviously the alignment yielded by assigning the $k^{th}$ incident to the $j^{th}$ priority level is given by the alignment of closing the incident by the time $t_j$, which we know how to compute from the previous sections. We'll call this measure of alignment $a(i_k,t_j)$ for short

The next thing to be noticed is that the constraints that the user imposes on the distribution of the incidents into priority levels can be trivially translated into minimum and maximum capacity constraints for the priority levels. For example, when dealing with $n$=200 incidents, the requirement that at least 40% of the incidents will be assigned medium priority (assume that is priority level $p_2$) would read: $\sum_{k=1}^{n} x_{2k} \geq 80$

In general we assign a minimum ($c_j$) and maximum ($C_j$) capacity constraint for a priority level $j$ that are symbolized as

$$\sum_{k=1}^{n} x_{jk} \geq c_j \quad and \quad \sum_{k=1}^{n} x_{jk} \leq C_j \quad \forall j = 1..m \tag{1}$$

The mathematical formulation of the incident prioritization problem (IPP) becomes:

$$(IPP) \quad \min \quad \sum_{j=1}^{m} \sum_{k=1}^{n} a(i_k, t_j) \cdot x_{jk} \tag{2}$$

$$s.t. \quad \sum_{k=1}^{n} x_{jk} \geq c_{jk} \quad and \quad \sum_{k=1}^{n} x_{jk} \leq C_{jk} \quad \forall j = 1..m \tag{3}$$

$$\sum_{j=1}^{m} x_{jk} = 1 \quad \forall k = 1..n \tag{4}$$

$$x_{jk} = 0 \ or \ 1 \quad \forall j = 1..m, k = 1..n \tag{5}$$

The solution of this problem will yield the optimal assignment of priorities to the incidents.

### 2.3.2 Incident prioritization algorithm

Here is a pseudo-code description of the incident prioritization algorithm

**Input:**

A number of SLOs, each modeled with the following information:

- Violation threshold for an SLO
- Jeopardy threshold for an SLO
- Penalty cost for SLO violation
- Expected time from jeopardy to violation (if no corrective measures are taken)

A set of priority levels for incidents, modeled with the following information:

- Constraints over the number of incidents to be assigned to each priority level
- Expected maximum closing time of incidents assigned to the priority level

A number of service degradation incident, modeled with the following information:

- Incident start time (the time when a jeopardy alarm was raised)

**Output:**

A complete prioritization of the incidents that assigns to each of them a priority level

**Steps:**

```
1. Compute the default alignment with business objectives
(section 3.2)

2. For each incident:
      2.1 For each priority level:
            2.1.1 Compute the likelihood of violation of
```

```
this incident if assigned to this priority level (section 2,
using the expected maximum closing time for the priority
level)
              2.1.2  Compute the alignment with business
objectives if a violation occurs (section 3.4)

3.  Solve the incident prioritization problem (section 4,
using the values of alignment derived from the previous
step)
```

## 3. Architecture of the integrated solution

The natural point of integration for our prototype is with the service level management capability of Openview Service Desk (OVSD). OVSD is the tool that falls more squarely in the domains of service level management, incident management and problem management. It allows a user to define a hierarchical service structure with multi-tiered SLA capabilities to describe the relationship between a higher level business service and the supporting operation management service.

OVSD was an excellent starting point for us because it provides most of the dependencies information necessary to build the data structures that we use as the basis of our incident prioritization method. Our MBO prototype complements OVSD by helping the IT personnel faced with the incident prioritization problem with support for their decision based on data and models that are readily available through OVSD.

HP OpenView Internet Services (OVIS) provides monitoring capabilities necessary to service level management, such as monitoring of availability and response time, along with notifications and resolutions of outages and slowdowns. It builds on a highly scalable and extensible architecture that allows programmers to build probes for a wide variety of data sources.

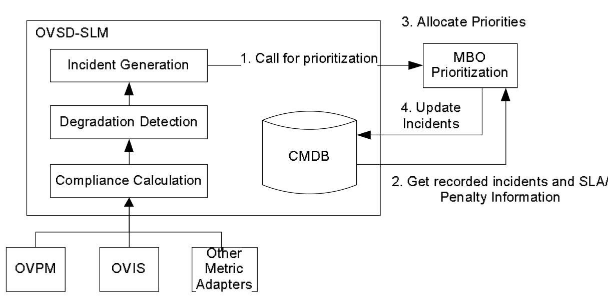


**Figure 6:** Architecture of the integrated solution

Figure 1 presents the architecture of the integration of the MBO prototype with Openview Service Desk (OVSD). OVSD receives data feeds from sources as diverse as OpenView Internet Services (OVIS), OpenView Performance Manager (OVPM) and other data feeders. Aside from its reporting activity, the OVSD internal machinery that has to do with service level management -- referred to as OVSD-SLM -- can be summarized in a three step process. The first step is compliance checking during which OVSD-SLM seeks to assess whether current measurements comply with existing service level objectives (SLO). This compliance phase uses service level agreements contained in the Configuration Management Database (CMDB) from which are extracted SLOs. Multiple compliance thresholds can be defined for each SLO such as violation and jeopardy thresholds. This allows for proactive management of degradation of service. The second step is Degradation and Violation Detection during which it is detected that a particular metric associated with an SLO has either met a degradation threshold or has reported values that are violating that SLO. In both cases, this leads to the next phase, Incident Generation, which reports the violation or degradation as an incident.

At that stage, it is necessary to characterize the incident from a business perspective. This is done (step 1) using the MBO prototype prioritization engine. The MBO engine is automatically notified on occurrences of SLA jeopardy alarms. On notification, the MBO engine fetches (step 2) all the open incidents from the CMDB and extracts the ones that have not yet been handled, along with their related SLAs and penalties. Finally, once the priorities are computed (step 3), the MBO engine updates (step 4) all the incidents with their new priorities.

## 4. Conclusions and future work

We describe a solution for the management of IT service incidents. In particular, the solution deals with prioritization of incidents driven by an enterprise's business objectives. The solution integrates the Service Level Management component of Openview Service Desk (OVSD-SLM) with the Management by Business Objective (MBO) technology being developed at HP Labs.

MBO will be integrated into the next release of OVSD-SLM.

We are now planning to apply similar techniques for decision support to the domain of IT change management.

## References

[1] Office of Government Commerce (OGC), editor. "The IT Infrastructure Library (ITIL)" The Stationary Office, Norwich, UK, 2000.

[2] F. Gallegos, D. P. Manson, S. Allen-Senft, C. Gonzales, "Information Technology Control and Audit", *Auerbach Publications*, 2004.

[3] R. S. Kaplan, D. P. Norton, "The Balanced Scorecard", *Harvard Business School Print*, 1996.