

An IT Business Impact Management Framework

J. P. Sauvé, J. A. B. Moura, M. C. Sampaio

Abstract

Will be written at the end

1. Introduction

- IT-Business alignment
- Methods to manage IT infrastructure from the business perspective, that is using metrics understandable to business executives, and reflecting business priorities
- Business Impact Management and business process management
- Synonyms (footnote)
- What do we want to do with this (prioritize IT actions, drill-down, get a feel for alignment, get data for future capacity planning and infrastructure expansion decisions)
- Service impact model or BIM model sustains BIM
- Operational versus long-term (potentially strategic)
- State-of-the-art: goes up to Business processes with their own metrics, but although closer to business, these are still not business metrics.
- Definition of business metric and business process metric
- Our objectives: better capture the linkages all the way up to the business layer and provide low-intrusion methods. Framework allowing more or less intrusion
- Describe structure of paper

2. Framework Requirements

This section outlines the requirements that must be satisfied by the framework in order to accomplish its objectives.

1. [impact] Show the impact of IT faults or performance degradation couched as Business Metrics
2. [drill-down] Drill-down capabilities
3. [IT measurements] Map IT measurements to business metrics
4. [Low intrusion] define what we mean by this
5. [Flexible] (allowing the addition or removal of entities)
6. [Operational] environment: immediately calculate changes in business metrics as a result of changes in infrastructure measurements.
7. [framework] Must be a framework to produce several models, depending on instrumentation available.
8. [business changes] Allow business managers to change business priorities

3. The Framework

This section describes the framework. It is the main body of the paper and is organized through several subsections.

3.1. A Layered Model

- For requirements 1 2 and 3, Layered model

- Two basic layers: IT and business are necessary, but drill-down is better with more layers
- Explain the four layers (figure)
- Each layer has metrics
- Each layer maps or calculates metrics
- In software, metrics are shown on dashboard (dashboard at each layer)
- Drill-down can proceed from top layers to lower layers to see cause-and-effect relationships
- List possible users for drill-down model to cover
- Notion of framework: have an abstract model that all concrete models (instantiations of framework) will follow but with a lot of elbow room to account for different instrumentation realities. More instrumentation means you can measure more; less instrumentation means you have to calculate more metrics, even in imprecise ways

3.2. General Layer Organization

The framework's four layers have many characteristics in common and these are described in this section. The notion of entity, relationships, dependencies between layers, mapping functions, types of metrics, drill-down operations, etc. are discussed here insofar as they are generic and applicable to all layers.

- Refer to a figure showing layer organization
- Neighboring layers, linkages
- Aim of a layer:
 - Produce metrics for SLA calculations and dashboards
 - Produce metrics for layer above
 - Metrics can be measured or can be calculated from lower-layer metrics + internal layer info
 - Tendency to measure more at bottom and calculate more at the top
 - Provides a drill-down model (inside the layer)
- Entities organized as a “Composite” (hierarchical) structure using notion of dependency
- Explain what a dependency means
- All entities have metrics
 - Metrics frequently capture performance degradation
 - “Health” is a common metric
- Some entities provide metrics for the whole layer and these are available to the upper layer (together with the identification of the entity that produced it)
- Dependencies can have attributes (eg. Weight, importance, ...)

3.3. The IT Component Layer

- Entities are basic IT components
- Examples: router, switch, host, etc.
- Metric is health of component, value from 0 to 1 (better than status (which is binary and doesn't capture performance degradation) or availability (which can only be calculated over time). Health is an instantaneous measure (requirement 6). Health of zero means “down” or unacceptably bad.
- Individual component health is measured
- Dependencies and composite components. Examples (database) – use figure

- Composite component health is calculated through a function. The framework doesn't specify a particular function. Typically, it would be "worst of children"
- Examples
 - Host: how to calculate health through four metrics
 - All metrics can be made available to SLA/dashboards, although our model only allows health to go up to the above layer (simplicity, requirement 4)
 - Router/switch: CPU utilization, memory utilization (or drop rate)
 - How to convert utilization to health (figure)
 - $u \leq 70\% \Rightarrow \text{health} = 1$
 - $70\% < u \leq 85\% \Rightarrow \text{health} = 0.5$
 - $u > 85\%, \text{health} = 1 - u$
 - what to do with a database (composite)
 - How to model network

3.4. The IT Services Layer

- Entities are applications or other top services provided by IT (mail, DB app, web service)
- Composite to get ancillary services (DNS, ...other apps)
- Dependencies on lower layer (which components are used to provide service)
- Basic metric is health (must capture performance degradation)
- Health of an entity is a function of the health of dependents (figure)
 - Can be any function, but typically would be "worst of children"
- Typical to measure response time and calculate health from this
 - Database application as example: measure RT for business transaction
 - How to convert RT to health? (refer to above figure about utilization)
 - Até 100 ms: 1
 - 100-> 300 ms: .7
 - 300->700 ms: .5
 - 700-> 2s: .3
 - 2->10 s: .1
 - > 10s: 0
- Can create IT service for a particular user group
 - Example: to get geographic view and drill-down capability

3.5. The Business Process Layer

- Entity is BP.
- Explain what a BP is: workflow. For low intrusion, we don't model whole workflow.
- We only identify BPs
- Can be composite (sub-processes)
- Basic metric is health (must capture performance degradation)
- Dependencies with IT services in lower layer (figure)
- We assume no BPMS: therefore, health is calculated
 - If BPMS exists, can measure metrics such as End-to-end response time, etc. and relate that to health. List other possible BP metrics
 - Note that many papers call these metrics "business metrics". We haven't reached the business layer yet!
 - These other measures can feed SLA auditor and dashboards

- How to calculate health? Many alternatives
 - Worst of children's health
 - Product of children's health
- These would be adequate if all IT services are necessary.
- Another example requiring different health calculations: widely dispersed BP
 - IT services modeled separately for each region
 - Associate attribute to dependency relationship: number of people using the BP in that region
 - Overall BP health would be $\sum(w_i, h_i)$ where w_i is weight of i th child (user population) and h_i is health of i th child
 - Thus., if BP is working fine for 30% of user population, health would be 0.3

3.6. The Business Layer

Layer 4 is described in more detail. Characteristics particular to this layer are presented. Examples of entities, metrics, etc. are discussed. Furthermore, examples of mapping functions between the lower layer and this one are given. Particular attention is given to discussing Business Impact Metrics couched in business language.

- Entities are corporate entities for which business impact metrics are desired.
- Entities are enterprise, business unit, LOB, ...
- Therefore: Composite structure (see figure)
- Health could also be calculated here based on health of BPs used by a business entity
- Not very useful because it isn't couched in business terms
- We chose "loss" as metric
- Explain what loss is (negative impact measure: we don't go for positive impact), explain rate of loss versus accumulated loss (both measures are important)
- Dependencies (between business entities and also with BPs) receive a "criticality" attribute. Sum must be 1. This can be based on
 - Revenue generating power
 - Number of people using the entity
 - Importance based on the BP classification (explain operate/support/manage processes)
- Health (if desired) calculated as $\sum(\text{criticality} \times \text{health})$
- Loss is calculated as $\text{function}(\text{criticality of children, health of children})$
- In lower entities (connected to BP layer), loss must be carefully defined to be meaningful. If possible, can be a financial measure, but at least must capture relative values. absolute vs relative impact metrics. We would like absolute but accept relative. If absolute, could be lost revenue, cost of productivity losses (for stopped business processes)
- for entities higher up, loss can just be the sum of loss coming up from each child. So try to make the calculations coming up from different entities to be coherent so as to make addition meaningful
- can be Business metrics other than loss (give examples)

3.7. Service-Level Agreements and Alignment Metrics

- The decisions of IT-business alignment are expressed as SLAs
- Important to say how SLAs are included in the framework
- SLAs may be included in any layer and may be based on any of a layer's metrics

- SLAs may be used in drill-down operations
- New aggregate metrics are proposed that can be used to measure business impact of IT faults or performance degradations.
- They are based on SLAs
- SLAs typically involve a long time period
- A basic BIM measure we use is the misalignment index: how current IT service quality makes alignment deviate from an ideal value but compromising SLA compliance. How bad is my SLA compliance getting?
- One metric is IT-business alignment: how much is a particular health value compromising SLA compliance (could be called SLA non-compliance risk).
- It may not be a business metric (say why) but is a Key Performance Indicator (give a definition)
- Good characteristic of index is it gets worse with time if problems not fixed (negative impact is cumulative). Can show rate, can show sum, can show “recoverability index”
- Give formal definition of the metric
- A single index per SLA. Could combine with a weight (Cost of non-compliance)

3.8. Framework Formalization

Space and time permitting, this section will be developed to formalize the model using appropriate notational syntax.

4. Instantiating the Framework: An Example

- The framework discussed previously is an abstract notion that must be instantiated in order to be applied to a concrete situation.
- The meaning of instantiating the framework is clarified.
- Finally, a small example showing a concrete BIM model using the framework is given.
 - DB component on top of host and network
 - DB app, with dependence on ancillary services and DB component. Split in two to get geographical effect
 - One BP using a DB App over 2 regions, population of 20, 50
 - (show figure, show all mapping functions)

5. Results and Validation

This section discusses the methodology used to validate our work and discusses preliminary results obtained to that effect.

- Methodology
 - Have we satisfied the requirements?
 - Are results promising?
- Have we satisfied the requirements?
 - [impact] Show the impact of IT faults or performance degradation couched as Business Metrics
 - model produces business metric showing the (negative) impact, that is “loss”, due to IT component faults and performance degradations
 - [drill-down]Drill-down capabilities
 - several layers satisfy stakeholders

- show which relationships can be followed to do drill-down? (dependencies, measured metrics that are related to one another (eg. RT for database), SLA definitions)
- [IT measurements] Map IT measurements to business metrics
 - IT entities on which measurements are performed are explicitly represented in the model
- [Low intrusion]
 - simple models
 - little modeling
 - No BPMS (no BP execution data)
 - Little new instrumentation
 - May need more work
- [Flexible] (allowing the addition or removal of entities)
 - no entity is obligatory
 - entities are freely introduced
 - can even remove whole layer if can map between remaining metrics
- [Operational] environment: immediately calculate changes in business metrics as a result of changes in infrastructure measurements.
 - Loss captures instantaneous “rate of loss”
 - SLA non-compliance risk is also instantaneous
- [framework] Must be a framework to produce several models, depending on instrumentation available.
 - Entities are generic, metrics are generic, mapping functions are generic, what is measured can be defined
 - Having a “health metric” is generic. How to calculate it is specific.
 - Model must be adequately instantiated. More work, more flexible.
 - How to strike a balance between the two? More investigation
- [business changes] Allow business managers to change business priorities
 - can change weights on the dependencies present in the business layer
- Are results promising?
 - We need more time to see this.
 - We’ll have to depend on the opinion of IT and business managers as to usefulness
 - Objectively, can see if SLAs are better met
 - Can baseline business metrics and see if they improve with time.

6. Conclusions

What have we achieved?

- A BIM model including the business layer
- A way to capture business metrics
- Flexibility to model many different situations
- New metrics to measure alignment based on SLAs

What are the consequences?

- Low-intrusion way of better aligning IT with business

- Can prioritize IT actions based on business impact
- Can drill-down to focus on highest-return problems

What will we do in the future?

- Other model for long-term strategic view, BSC
- External entities
- Use it and factor more things into the framework to get a better job done, easier, (may segment by industry for example, with standard workflows)
- Pursue new low-intrusion avenues
- Dynamic context (dynamic infra [grids], dynamic services and service compositions, short-term business processes, ...)
- Can include cost alignment in metrics
- If we integrate with a trouble ticket to better track problem onset on resolution, can exhibit total cost of a failure (according to metrics chosen for cost)
- Modeling language support
- Refactor model for reuse (after we use it a bit)
- Better capture business impact
 - Regulation, Responsiveness, Revenue growth, Revenue efficiency (profit), Risk management?)
-

We would like to thank João Jornada and Eduardo Radziuk for a number of useful comments and criticisms. This research was supported by grants from Hewlett-Packard/Brazil

7. References

Provide references in text

Ref: managing with measures: talks of operate/support/manage processes

