Issues in Information Systems
Volume 23, Issue 4, pp. 317-327, 2022

DOI: https://doi.org/10.48009/4_iis_2022_128

Business-intent-based IT management

Abdur Rahim Choudhary, *Choudhary Associates, USA. arc@choudharyassociates.com*

Abstract

This paper presents a cross fertilization of ideas and technologies to allow for mechanisms to formally bind the business-intent to the technology infrastructure deployed to achieve it. A framework is proposed that allows the automation of the business processes in a way that supports a clear interface with the business-intent. The framework integrates the process of automation and the process of binding the business-intent with the enabling technology infrastructure. An audit mechanism is included in the framework to analyze the IT operations towards meeting the business-intent. The audit function provides feedback with respect to improvements, and a longer-term forecasting to accommodate evolution in the business-intent and the corresponding changes in the technology infrastructure.

Keywords: business-intent-based, business-decisions, decision translation, automation, audit, services, audit management, forecasting.

Introduction

The information technology (IT) was historically developed in its own scope, somewhat independent of any business and mission applications. The technology so developed was subsequently applied as a tool in achieving business and mission objectives. The evolution of the IT infrastructure gradually transcended its role as a business support tool, and became critical to business productivity and profitability. This happened as IT applications were deployed to greater and greater extent in business operations to automate the business processes, improve the quality of the products, reduce the time-to-market, and reduce production costs.

The increasing cost of provisioning and managing an IT infrastructure provided a motivation to use the same IT solutions in support of multiple businesses and missions. This trend gave rise to standardization of technology devices and application programming interfaces (APIs). Eventually there emerged a layer of services as the middleware between businesses objectives and technology deployed to achieve them. Today there are three distinct layers, namely technology, services provided by the technology, and businesses enabled by those services. Management decisions are made at each layer so that the business-intent can be achieved.

This leads to the business-intent-based information technology management (Baltzan 2020) (Machiraju et. al. 2004), which is an emerging concept (Baltzan 2021) and the technologies needed to implement this concept are also emerging. Therefore, the meaning of this concept has a range of interpretations. For the purpose of this paper the concept of business-intent-based IT management has the following three complementary objectives.

- Binding the business-intent with the IT infrastructure that supports them: It enables the alignment of IT infrastructure with the business-intent of the enterprise, via the business-intent-based management of the IT infrastructure.
- Automation: It enables automation of the IT operations in order to save operational costs, because such costs have become high and are rising with the complexity of the IT infrastructure.
- Auditing: Enables to envision and plan upgrades to the IT infrastructure to better leverage technology in support of the current business objectives and to support their future evolution.

There is a parallel but related development in the Federal Government of the United States, especially in the Department of Defense (DoD), known as 'information-centric operations' (Westzel 2017). This is an evolution of the earlier DoD doctrine of net-centric warfare (Fewell and Hazen 2003) (Choudhary 2009).

This paper is an original contribution to develop fundamental concepts in support of the above three objectives; it also proposes a solution framework to achieve them and defines its constituent modules as well as the interfaces between these modules.

Next section discusses the approach using the concept of Decision Translation, followed by the section that presents a solution framework using the translated decisions. Next section discusses how the above three objectives are achieved in this solution, and the final section summarizes the conclusions.

Decision Translations

The technology is a productivity tool to support the business-intent. It works by providing a number of services that directly improve the business processes designed to enhance productivity and reduce costs. Synchronized management decisions need to be made at all levels, namely business, services, and IT infrastructure. These decisions align the IT infrastructure with the business-intent via the services, as is depicted in Figure 1.

Figure 1. Illustration of a hierarchy of management decisions, and business and service level agreements.

For example, at IT infrastructure level management decisions can be for network management; at the services level the management decisions can be for the quality and security of services; and at the business level the management decisions can be to enhance customer satisfaction, increase sales, lower costs, and maximize profits. Such layers of management decisions (AXELOS 2019) (COBIT 2019) are not alternatives to each other; rather, they need to be complementary and mutually supportive.

However, different concepts and terminologies are used at the business, services, and technology levels. In order to make the technology align with the business-intent, it becomes necessary to translate the business management decisions into the technology management decisions. Decision translation, therefore, forms the bases for business intent-based technology management (BIBTM).

The management decisions at various levels of the hierarchy shown in Figure 1 are helpful in understanding the operational situation by providing information in an appropriate terminology and format that the decision makers are familiar with. The decisions for the business-intent are not directly implementable, except via their translation into the IT parameters that configure the enabling services. In other words, it is necessary to identify the services that would enable the business-intent, and it is necessary to identify the IT resources necessary to provide the needed services.

The decision translations between the business-intent and the enabling services play a key role in the business outcome. These decisions are not automatable because the intent of a human decision maker can in general not be trusted to a machine for its interpretation and execution. This means that the business decisions are manually translated into the supporting services; though various science and technology tools are available to assist.

The result of this translation is documented as a Business Level Agreement (BLA), shown schematically in figures 1 and 2. It represents a coupling between the business-intent and the enabling services.

This manual translation needs to include enough details. For example, it would include a list of measurable business metrics such as product quality, customer satisfaction, time to market, sales, cost, and profits; and it would also include quantifiable services metrics such as reliability, maintainability, availability, quality of service, and security of operations. Decision translation minimally maps these two sets of metrics and provides some level of accountability, responsibility, and guarantees. This mapping would include a verifiable understanding with respect to how the provision of the services can generate the business metrics, which in turn needs to be comprehensive enough to be able to determine if the intent of the business decisions is fulfilled. This part of the decision translation is documented as a business level agreement (BLA).

The services part of a business level agreement is realized by translating it into the needed technology devices and their configurations. The services are enabled by provisioning the devices and configuring them for the agreed quality, security, and other attributes of the enabling services. The translation of these service metrics into a specification of the devices and their configurations, is documented as a service level agreement (SLA), shown schematically in figures 1 and 2. The specified devices and their configurations must support the required services and their quality under all envisioned operational conditions.

The translation from the service behavior to the configuration parameters of the enabling technology can be automatable. The extent of this automation is a function of the technology itself, including the state of the service provisioning technology, and any applicable IT management capabilities and their automation.

The BLA and SLA are not new as concepts, but our formulation of them as well-defined mappings between sets of well-defined metrics is new.

Solution Framework

The fidelity of a design for business-intent-based IT management rests on the correctness of the translations embodied in BLA and SLA. There is an inherent subjectivity in them, especially for the BLA.

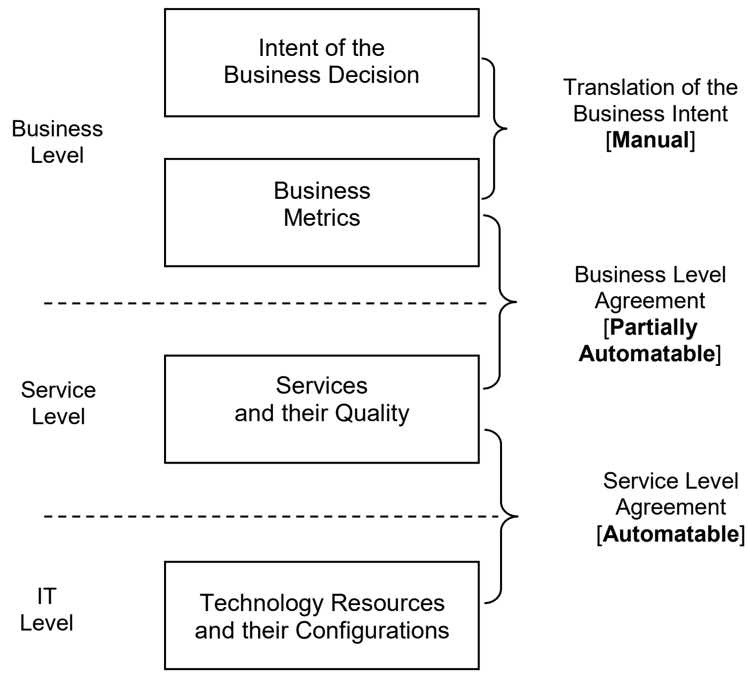

Figure 2: Translations embodied in BLA and SLA

The business intent like increasing the profits is not directly translatable into technology services. Therefore, it is translated into suitably measurable business metrics which further admit translation into technology services. As shown in Figure 2, the translation of business intent into the business metrics is manual and rather subjective. The business metrics and the metrics for the enabling technology services are mapped onto one another, and the mapping formulates a business level agreement (BLA).

The uncertainty arises from the uncertainty in the choice of business metrics and technology services. The business decision may not be achievable if it is not well represented by the chosen business metrics; which in turn may not be well achieved with the choice for the enabling services.

The services can be easily provisioned and configured. This process is again specified using a number of service quality metrics and the metrics for their enabling technological devices and their configurations. These two sets of metrics are mapped onto each other, and this mapping formulates a service level agreement (SLA).

The decision translations are incorporated into the choice of business metrics hoping that their achievement will result in the actual achievement of the business-intent. This part of the decision translation is manual, and its verification also is manual.

The business metrics are computable entities. They can be mapped onto another set of metrics that specify the enabling IT services. This mapping is formally documented as a BLA.

Mapping: Business Metrics ←→ Services Metrics → BLA

To ensure that the services are correctly provisioned, the service quality and security metrics are mapped onto a chosen set of devices and their configurations. This mapping is formally documented as an SLA.

Mapping: Services Metrics ←→ Device and Configuration Metrics → SLA

A BLA can be verified by measuring the business metrics that are used to formulate it. To validate that the achievement of these metrics actually achieves the business-intent of the enterprise is a much harder problem.

Verification of the SLA is relatively well understood. However, SLA specification in a way that admits automated auditing is a work in progress (Trienekens et. al. 1999) (Sahai et. al. 2002).

Once the service level decisions are translated to the configuration parameters for the underlying technology devices, the objective is to automatically evaluate and enforce the SLA, which requires that the service level decisions are reducible to a reliable set of technology metrics that permits evaluability using computers. This is further discussed below.

Decision Server and Client Agent

When the decision translations become computer evaluable, automatic decision evaluation and enforcement then becomes possible. The conceptual design presented in Figure 3 is for this stage of computer evaluability of the translated decisions.

The solution framework has a server component (here called Decision Server) and a client component (here called a Managed Object). In Figure 3, the Client Agents are shown as shaded regions within the managed objects. The software in the client agent is a function of the object that it is intended to manage. Therefore, a special Client Agent is needed to manage each type of managed entity. Copies of the same Client Agent can be used when multiple copies of the same object are managed. However, a different type of managed entity will require a specialized Client Agent of a corresponding type. The Client Agents shown in Figure 3 are like the Policy Enforcement Points (PEPs) in a Policy Based Management System (PBMS).

Each Client Agent registers with the Decision Server in order for the server to serve it.

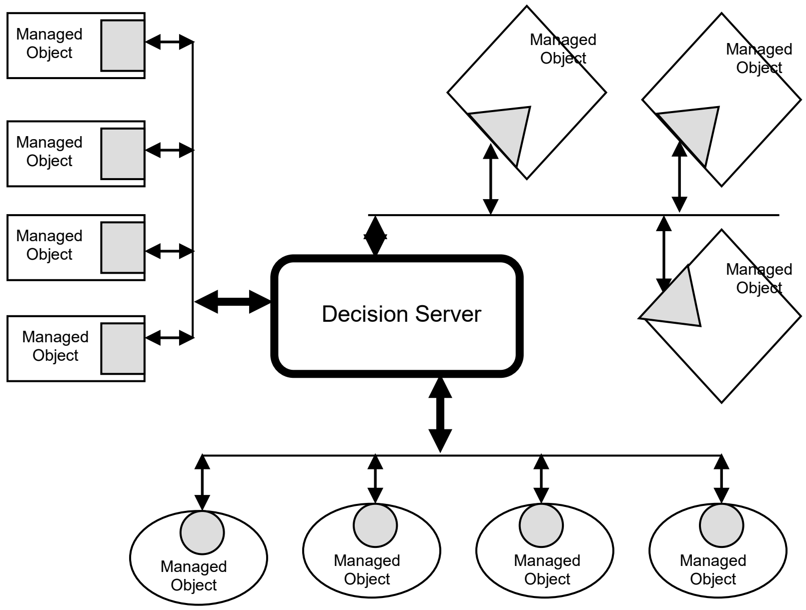

Figure 3: Decision Server is shown connected to the Client Agents, which are shown shaded. Each Client Agent connects to the object that it manages.

Decision Logic

Figure 4 shows further details regarding the internal structure of the Decision Server and the Client Agent as well as the interfaces between them. The Decision Server contains all the rules that govern the dynamics of BLA and SLA.

A rule is a statement of the following type: IF (EVENT) THEN ACTION

All management decisions at business, services, or device level are eventually translated into a set of this kind of rules, that we will refer to as a ruleset. The rules state that if a particular EVENT takes place, then a specified ACTION is to be taken. This simple IF (X) THEN Y type formulation is deceptively simple. In reality each such statement is often quite complex. The complexity comes in specifying a compound event as a sequence of events. Similarly, an action can be complex, embodying a sequence of choices and a set of actions corresponding to each choice.

Further design complications invariably arise because the choices, Events and Actions have their own parametrizations for precise determination. There are specialized languages that are used to specify the rulesets. The decision server needs to support interfaces with these languages.

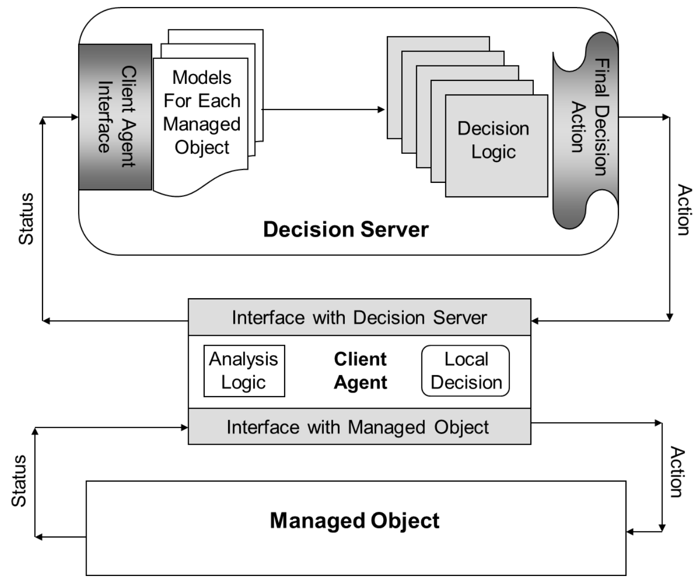

Figure 4: Internal details of Decision Server and Client Agent and their interfaces.

Backend and its Interface

The backend is not shown in Figure 4. It is needed for the Decision Server to be populated with the rulesets needed to manage the objects via their specific Client Agents. These rulesets are formulated using a standardized language and format in such a way that the Decision Server does not need to know the details about the managed object. Such details are necessary but contained within the respective Client Agents.

The decision server needs to support interfaces with the backend to populate itself with the rulesets and to modify them as operationally necessary. There are additional data in the backend database, like those for the set of all Client Agents and managed objects.

Client Agent and its Interface

There is a Client Agent instance for each managed object. Each type of managed object, like a router, a switch, or a database, has a Client Agent specially designed for the type.

For the Decision Server the Client Agent serves as a proxy for the associated managed object. The decision server receives the device status and operational events from a client agent, evaluates the applicable rulesets relevant to the operational events, and sends back the results for actions to be taken on the managed object. Minimally, the decision server needs to support the interfaces required for these communications with the client agent: to receive the status and events from the Client Agent and to send back the decision regarding the action to be taken. This is shown in Figure 4.

Decision evaluation can involve information coordination, depending upon the nature of the decision being evaluated. For example, a decision regarding a distributed security attack (Dahiya and Gupta 2019) would require correlation from multiple client agents. In such scenarios, the managed object is really a whole network rather than an individual network element.

The managed object has at least two interfaces with its Client Agent. One interface is for the client agent to receive device status information and operational events that occur in the managed object. The other interface is for the client agent to reconfigure the managed object according to the action-decision received from the decision server.

Each Client Agent is represented within the Decision Server through the registration record. This information is used when the Client Agent sends the Event status information, or the Decision Server sends the result of a Decision Evaluation. The Decision Server also makes use of a model for the managed object, as is shown in Figure 4. However, these models are not detailed, they include enough information to accomplish the interfaces.

Figure 4 also shows a special function for a Client Agent, namely to act as a local decision server for certain specified functions in its managed object. If a client agent is to play this role, an Analysis Logic module and a Local Decision module are then included.

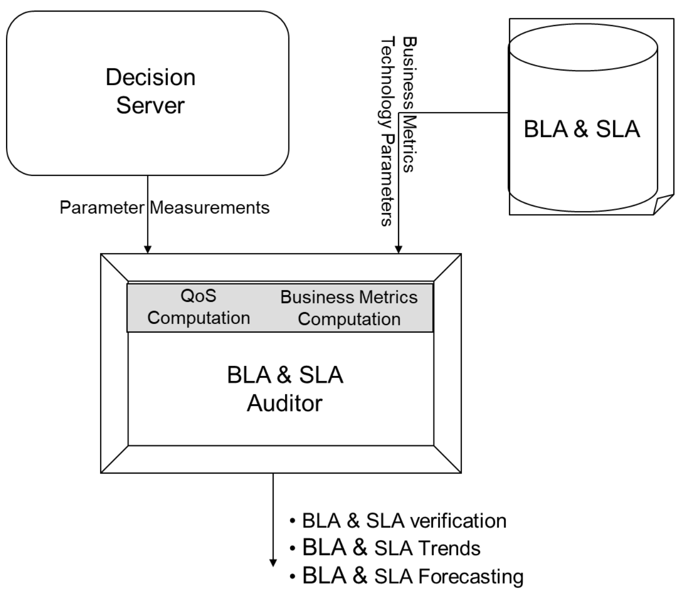

Figure 5: Auditing of BLA and SLA.

Auditing BLA and SLA

The Auditor module is shown in figure 5. It collects the Measured values of the parameters from the Decision Server. The measured values of the parameters are used to construct the current values of BLA and SLA metrics. Thus, compliance with SLA and BLA is monitored and audited. The process also uses additional details about the SLA and BLA from a backend database.

The feedback obtained from this auditing is analyzed for compliance and trend analysis, both for business and technology. The results are used to help correct or finetune the business and technology rulesets, and also to forecast the evolution of the service features and business operations.

Assessing the Objectives

The analysis and solution presented in this paper meets the three objectives for the business-intent-based IT management as follows:

Formal Binding between Business intent and Technology is achieved using the rulesets that populate the decision server. These rulesets implement a binding between the business metrics like enhanced customer satisfaction, increased sales, lower costs, and increased profits and the technology metrics like reliability, maintainability, availability, quality, and security. The binding can be revisited for correction and finetuning using the feedback from the auditor.

Automation of Technology infrastructure Management is amply achieved through the use of rulesets that automate the management operations using the decision server and the client agents.

Auditing for Improvements and Evolution of operations is achieved by implementing the BLA and SLA auditor. It verifies the current BLA and SLA compliance, analyzes trends, and helps forecast future evolution in both the business and technology performance.

Conclusions

This paper has proposed three objectives for the business-intent-based IT management: namely to formally bind business-intent with technology infrastructure; automation of operations; and auditing capabilities. The paper presents a solution framework to achieve these objectives. First, a decision translation mechanism is used to bind the business-intent with the enterprise IT infrastructure, using business level agreements (BLAs). BLAs are enabled using technology services whose quality is assured by mapping service metrics onto the technology devices and their configurations. This mapping is documented as service level agreements (SLAs). The BLAs and SLAs are enforced using mathematically defined rulesets. The enforcement is automated using a framework consisting of a decision server and a network of client agents. An audit module is part of the framework which monitors the compliance with BLAs and SLAs, provides feedback to revise and finetune the rulesets, and provides forecasts about the evolution of business-intent and technology infrastructure.

Acknowledgements

The author thanks Choudhary Associates for providing the environment for this research. He also thanks Yasmeen Sultana for inspiration and admin assistance.

References

AXELOS, "Information Technology Infrastructure Library (ITIL) ver. 4", axelos.com, 2019.

Baltzan, P., "Business Driven Information Systems", 7th Edition, McGraw-Hill Education, ISBN13: 9781260262483, 2021.

Baltzan, Paige, "Business Driven Technology", ISBN 978-1-260-42524-6, 8th Edition, McGraw-Hill Education, New York, NY, 2020.

Choudhary A. R., "Network Management in Net-Centric Systems", Proceedings of the Military Communications (MILCOM) 2009, San Diego, CA, USA, November 2009.

COBIT, Information Systems Audit and Control Association, "Control Objectives for Information and related Technology (COBIT) 2019", isaca.org, 2019.

Dahiya, A., and Gupta, B., "A PBNM and economic incentive-based defensive mechanism against DDoS attacks", Enterprise Information Systems Vol. 16 Issue 3, 2019.

Fewell, M. P. and Hazen M.G., "Net-Centric Warfare – Its Nature and Modeling", Maritime Operations Division, Systems Sciences Laboratory, Defense Science and Technology Organization, Department of Defense, Australian Government, Report DSTO-RR-0262 and AR-012-876, September 2003.

Machiraju V., Bartolini C., and Casati F., "Technologies for Business-Driven IT Management", HPL-2004-101, HP Laboratories, June 2004.

Sahai, A., Durante A., Machiraju V., "Automated SLA Monitoring for Web Services", p. 28-41, in "Management Technologies for E-Commerce and E-Business Applications", ISBN 978-3-540-00080-8, Springer, Berlin/Heidelberg, 2002.

Trienekens, J., Zwan, M., Bouman, J., "Specification of Service Level Agreements, clarifying concepts on the basis of practical research", Proceedings of Software Technology and Engineering Practice, p. 169-178, Pittsburgh PA, USA, 1999.

Westzel, T., "Information-Centric Operations: Airpower Strategy for the 21st Century", 2017.