Business Alignment Viewed Through the Eyeglass of Promises

Mark Burgess
Oslo University College, Norway

***Abstract*— A definition of business driven technology management is discussed. Alignment of helper technologies with business goals is discussed in the framework of promise theory. It is shown how promises help indicate where alignment can be measured, by looking at the value of promises by a business, its sources and its customers. By having a simple model of these valuations, one can then use it as a platform on which to develop a chain of service level agreements for end-to-end service delivery from the perspective of business, rather than from the IT devices.**

***Index Terms*— Promise theory. Business driven IT management.**

I. Introduction

How can Information Technology (IT) Management be informed and guided by business objectives in an optimal manner? The scope of IT management is large and the realm of business objectives is no smaller, so at first glance it seems like a question that is likely to drown in its own uncertainty for lack of any concrete definitions. This paper will propose that this need not be the case.

Many interesting papers have sought to explore the boundaries of the problem by looking to technology, legal issues, security etc [1], [2], [3]. Several authors began by examining the efficiency of a particular technology configuration and then considering its optimization through rational means [4], [5], [6], [7]. The obvious examples for study are those that involve electronic commerce, such as online web services etc. The reason is clear: we understand the behaviour of technology far better than we understand the behaviour of humans, so applications in which humans play a negligible role are natural places to start. However, businesses are in fact human-computer systems[8] and we have to be able to incorporate both aspects into modelling businesses if we hope to truly see IT as an *effector* in a more general process network.

Recently in this vein it seemed germane to introduce a modelling framework for human-computer systems that captures the interactions and uncertainties in a network of humans and computers who plan their behaviour. The framework (which has come to be known as Promise Theory) can address the issue of "business alignment", or business driven management[9] by abstracting away humans and computers with generalized agents that make promises.

In this paper then, a simple discussion of business promises is begun with the aim of showing how the concept of promises leads to an understanding of *value* or potential profit, and hence how such promises can be thought of as the driver by which business goals can be achieved. The example of a service delivery chain is used to illustrate the ideas.

II. Modelling

The goal of a business is to deliver a service S (in some broad sense) in exchange for remuneration. This is a general definition, and we aim to keep it as non-specific as possible so that the conclusions might have the broadest generality. The exact nature of the service is unimportant might be delivery of a physical product, provision of information, etc; similarly, the exact nature of remuneration could be by money transfer, exchange of goods, or even profit of something more abstract like "goodwill".

The simplest way to model this is view it as two promises, back to back: a promise to provide a service and a promise to pay. In fact, most parties will only make promises on the condition that they are promised something in return (see fig 1), so we can add the notion of conditionals also. This will have considerable consequences for the complexity of the agreements between the parts of the business.

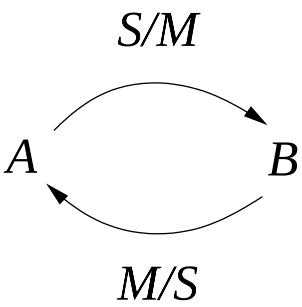

Fig. 1. A simple business relationship with a client: the business provides a service if money is promised, the customer promises money if service is promised.

In the basic configuration, A promises B the service S if it receives money M, and B promises A money M if it receives service S. This of course results in deadlock, unless there is an initial gesture of trust by one of the parties to offer their promise unconditionally (formally this can be seen an initial condition to the iterative process, at time t_0, so that the promises exist mutually-conditionally for all times after t_0). Both solutions for breaking deadlock exist in actual business relationships.

A. Business goals

Businesses have probably many goals in their grand designs: they have high level visions, notions of secure and best practices, sometimes even ethical policies. All of these can be couched in the language of promises to behave in some way. The overriding goal of any business, however, must be survival. To survive, the business must be worth something. The value of a business is obtained both as actual money through its balance of payments, and as fictitious value based on its potential to make future profits. The potential for new income is, in turn, affected by both human and technical factors: productivity, reputation, market interest, etc.

In general what is needed to keep a business alive is a promise of income, or a perceived value to its investors. We can model both kinds of value within the framework of promise theory. The former is a transaction that is promised, the latter is a perceived value that could be connected to a promise to sell the entire company in the future. The value of future sale could be motivated by other promises by the business agent; for instance, a potential buyer might be impressed by the capacity for growth based on a promise to keep up a particular marketing or development strategy.

B. Alignment

Now, we can ask: what does it mean to align an IT infrastructure to this business goal to provide S? First, for IT systems to have any impact on the business goal at all, the business must rely on the IT system in some way. This could either be directly, in the manner of an e-commerce web-site, or it might be indirectly, for instance by providing drawing and modelling software in an architect's office. In either case there is a workflow in which an IT system plays an intermediary role in the workflow process.

In fact, it does not matter whether this is an IT system, a human being or a steam-powered engine. What is key is that there is a technology playing an intermediate role in the performance of a service. We can display this as the workflow diagram shown by the dotted lines in fig 2. The business B would like to provide service S to its customer C; in actuality this requires the help of intermediary I.

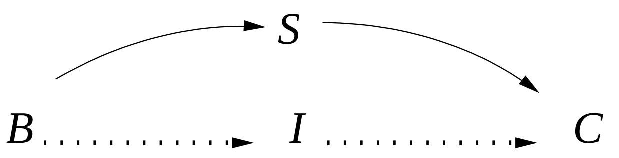

Fig. 2. Inserting an intermediate agent into a business process. The dotted lines show a work flow path. The arc shows a promise the business would like to make to the end customer – but promise theory says that it cannot if it does not have direct contact.

Promise theory has several implications, and one of them is that an agent cannot promise something to an agent it is not directly in contact with. This is because agents can only vouch for their own behaviour. They cannot promise what an intermediate agent would do.

To explore the implications of this, let us make the simple abstraction shown in the diagram, i.e. the flow of work through a chain of intermediaries, and frame this problem in promise theory.

III. Basic promise theory

Promise theory is a set of assumptions and a language for describing promises made between autonomous agents[9], [10], [11]).

Assumption 1 (Voluntary Cooperation): Agents (i.e. humans, computers or indeed any entity that can reasonably thought to hold some kind of promise, whether actually, vicariously or by association with its owner or designer) are said to be autonomous if they cannot be forced to make any promises about their behaviour by any outside agent.

This assumption means that agents can only extend their influence on others by making personal promises that ensure others will value their promises and cooperate voluntarily. This 'extreme' viewpoint is a somewhat realistic model of how people, businesses and even technology actually behave in the final analysis. Some readers will find it cynical, but cynicism is based on worst case realism, and so one can think of it as embodying the assumption of *risk*. More pragmatically, it forces us to document every condition for cooperative behaviour between the parts of a system that could (for any reason) cease to behave in the manner we might prefer for the good of the business.

Promise theory agents are therefore impenetrable to outside influence, possess private knowledge, and the promises that they make to one another cannot be forced onto them by someone's outside will. This apparent limitation confuses newcomers into thinking that this is a flaw in the theory. After all, everyone knows that when the boss commands, slaves obey. However, this is not a flaw in promise theory but a strength: command is an illusion that survives only in systems where voluntary cooperation can be taken for granted. The challenge in promise theory is to show how or why such a command could be realistically given and obeyed, given the fact that no agent of intermediary in the chain of promises is always willing or able to keep all of its promises (even with the best of intentions).

A. Promises in a nutshell

Promises are given by "agents" which may be humans, computers or any other representative entity in a system. A promise with body $+b$ is understood to be a specification to exhibit or "give" behaviour from one agent to another (possibly in the manner of a service), while a promise with body $-b$ is a specification of what behaviour will be "received" or "used" by one agent from another (see table I). A promise *valuation* $v_i\left(a_j \xrightarrow{b} a_k\right)$ is a subjective interpretation by agent a_i (in any local currency) of the promise in the parentheses. Usually an agent can only evaluate promises in which it is involved, as promises are only received (observable) by their recipient and their giver.

Not all promises are made without condition. The example of payment for a service above illustrated a case in which we want to make a promise only if a condition is met, such a the receipt of a counterpromise. A conditional promise body is written X/Y if we promise X subject to the condition that Y is promised by another agent.

To make conditional promises work in an intuitive way, we need some basic rules. We say that a promise which is

Symbol	Interpretation
$a \xrightarrow{+b} a'$	Promise from a to a' with body b
$a' \xrightarrow{-b} a$	Promise to accept b
$v_a(a \xrightarrow{b} a')$	The value of promise to a
$v_{a'}(a \xrightarrow{b} a')$	The value of promise to a'
$\oplus$	Combination of promises in parallel
$\otimes$	Combination of promises in series

TABLE I
SUMMARY OR PROMISE NOTATION

made conditionally is not a promise, unless the condition that predicates it is also promised (a promise is empty if one makes it conditional on something unknown). So the first rule (fig 3) expresses this as follows. Let us represent our abstract

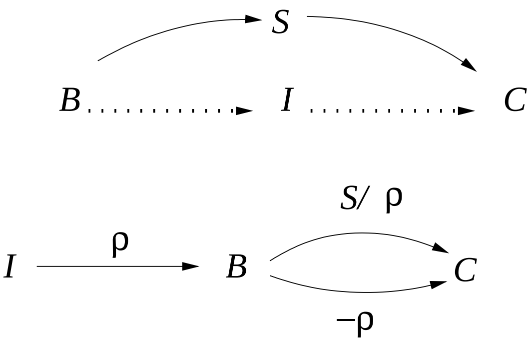

Fig. 3. Conditional axiom 1

business as an agent B and the customer as an agent C and any intermediary involved in completing the business process by an agent I. Now let B promise C a service S on the condition that it gets what it has been promised ρ by its intermediary technology I. This is promise at all unless it also makes a promise to C that it will use the service ρ provided to it. In other words, it confirms that it is going to do its best to get the service it has indicated it relies on.

This can be summarized as a rule:

$$B \xrightarrow{b} C \simeq B \xrightarrow{b/c} C \ , \ B \xrightarrow{-c} C. \qquad (1)$$

A conditional promise and a promise to acquire the conditional requirement is equivalent to an actual promise without conditions[1].

This can be simplified in the case the provider of dependent service is the customer, i.e. when $I \leftrightarrow C$ (see fig 4).

There is a lot more one could say about promises, but for the sake of simplicity let us suppress the details here. Readers are referred to refs. [10], [11] for details.

It is worth mentioning one more thing, however. Any promise that is made can be evaluated in a number of ways: what does it cost us to keep a promise and how long will it take? are two questions. What is the value to an agent

[1] Clearly this is not true in general, but we need not belabour the difference here. Both kinds of promise are subject to uncertainty. The key difference is precisely where the source of the uncertainty lies about the promise being kept.

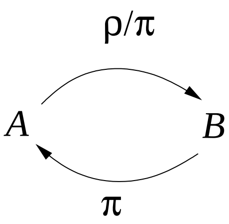

Fig. 4. Conditional axiom 2

when it is the recipient of a promise? Since each agent is autonomous and possesses private information, each agent must necessarily make its own judgment about what promises are worth, how much they cost and how long they take, from its own viewpoint. It does not have to agree with another agent. One agent might measure value in dollars, another in Euros, for instance. We can denote the value of a promise in the manner of a function, e.g. $v_i\left(n_i \xrightarrow{b} n_j\right)$ would mean the value of the promise in the parenthesis from the viewpoint of agent n_i.

The currency of value is a subtle matter. One thinks naturally first of money, as this is the modern face of value exchange, but any kind of beneficial trade can be used as a measure of value. Value could be in the form of goods, owed debt, reputation, goodwill, or as some economists have recently suggested even happiness[12].

IV. WHAT DOES BUSINESS ALIGNMENT MEAN?

Let us now consider what business alignment means, assuming that the important issues that explain the working of a business have been suitably described as promises[2]. As mentioned, a business might have many goals with which to align. Let us make a working hypothesis

Assumption 2 (Goals are valuable promises): All of business goals can be considered promises that that yield positive value to the company in some abstract currency.

It is not terribly important to whom these promises are made. Some of them are promises to business associations, some are perhaps promises to self or to society at large. Others are promises to the law courts, or whatever representative agent is the considered custodian of litigious correctness. All that is important from the viewpoint of promise theory is that the business agent is linked to some other agent by such a promise, and that the business agent itself places a value on that promise, whether it be coming or going, positive or negative.

From the perspective of promises, a business benefit resulting from a promise could be understood in two ways:

- A promise might be a "make or break" i.e. an *enabler* without which the service S cannot be provided.

$$I \xrightarrow{\rho} B \xrightarrow{S/\rho} C \qquad (2)$$

[2] Many businesses expect employees to follow a corporate credo or ideology. The act of articulating an ideology is like making a promise to comply with it, hence promises abstract all aspects of a business's direction.

- A promise is more often an *effector* or amplifier which parametrically increases business result.

$$I \xrightarrow{\rho} B \xrightarrow{S(\rho)/\rho} C \tag{3}$$

- Any miscellaneous promise is aligned with business goal if it confers a positive value to the business agent B according to its own judgment.

$$v_B\left(\text{Any} \xrightarrow{b} B\right) > 0. \tag{4}$$

V. Workflow chains

Using the language of promises, and the basic algebra of its conditionals, we can examine ability of a business to make a promise to a remote customer through intermediate agents. It should already be clear that, by the basic (cynical) assumptions of the theory, the business could never make an absolute promise through an intermediary.

Suppose our business wants to make a promise to a customer, but has to pass its good intentions through a number of intermediaries in order to complete the transaction. Typical intermediaries are transport agents (networks, ships, the post office), or building/construction contractors in the completion of a building. This is often called the *end-to-end delivery problem*.

Given that we cannot force one of these intermediary agents to behave perfectly, what promises do we need to make to guarantee that a necessary and sufficient effort will be directed to convincing the business agent B that the service will be delivered as expected. In other words what are the necessary promises to allow the business to be able to make a promise of service with the expectation that its promise will be kept?

The start of a chain relationship is the basic rule illustrated in fig 3. This guarantees that a single intermediary will be used in the deliverance of the promise.

Suppose the we add a second intermediate. This increases the complexity of the problem considerably. Each time a new intermediary is added (assuming that it is known to the business), it must make its promise conditional on the promises made by the agents. Similarly each of the agents that needs to rely on other agents to keep its own promises need to promise the suspicious agents ahead of it in the chain that the necessary relationships will be made and honoured.

Consider fig. 5. The business agent makes a promise conditionally on three other promises from the intermediaries ρ_i for $i = 1, 2, 3$. This means it must promise C that it will use these promises $-\rho_i$. This each of the agents must promise its particular ρ_i to B, but the agent that depends on its predecessor is only willing to offer this conditionally, i.e. on the condition that its predecessor promises delivery to it. Thus there is an instance of the basic relationship in fig. 3 between all but the final agent in the delivery chain: i.e. that agent which can complete delivery without further conditions (I_2 in this case).

Fig. 5 shows the additional promises for three intermediaries, with the additional relationships added. We begin to see a pattern emerge. Let i label any of n intermediaries, i.e. i

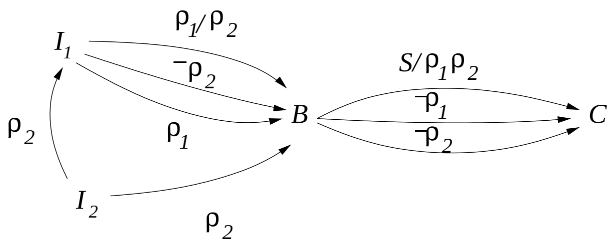

Fig. 5. Two intermediaries in a chain of dependent promises.

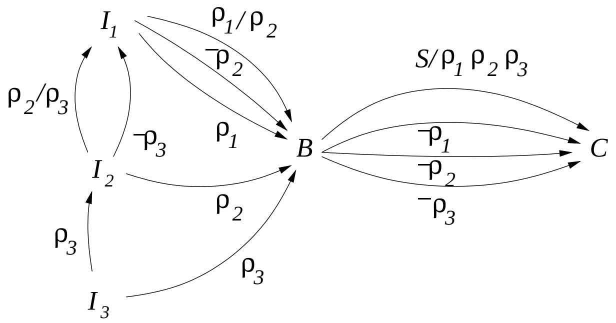

Fig. 6. Three intermediaries in a chain of dependent promises.

runs from $1..n$. The general case can therefore be written:

$$\begin{array}{ccc} I_n & \xrightarrow{\rho_n} & I_{n-1} \\ I_i & \xrightarrow{+\rho_i} & B \\ I_i & \xrightarrow{\rho_i/\rho_{i+1}\,,\,-\rho_{i-1}} & I_{i-1} \\ I_1 & \xrightarrow{\rho_1/\rho_2\,,\,-\rho_2} & B \\ B & \xrightarrow{S/\rho_1\ldots\rho_n\,,\,-\rho_1,\ldots,-\rho_n} & C \end{array} \tag{5}$$

This provides necessary and sufficient conditions for B to be able to make a promise knowingly based on a chain of n intermediary agents.

We see from these rules that the final agent B up the chain of command plays exactly the same role as the other agents in the chain of assurances: it is simply the end point for the intermediate promises so we could rename it I_0 and simplify the rules above:

$$\begin{array}{ccll} I_n & \xrightarrow{\rho_n} & I_{n-1} & \text{(initial condition)} \\ I_i & \xrightarrow{+\rho_i} & I_0 & \\ I_i & \xrightarrow{\rho_i/\rho_{i+1}\,,\,-\rho_{i-1}} & I_{i-1} & 0 < i < n \\ I_0 & \xrightarrow{S/\rho_1\ldots\rho_n\,,\,-\rho_1,\ldots,-\rho_n} & C & \end{array} \tag{6}$$

Using the basic axioms illustrated in fig. 3, a proof follows by construction. B makes a promise to C, so the conditions above are therefore the necessary and sufficient conditions for that promise to be made.

VI. Productivity: simplest is best?

A consequence of the foregoing model is to make the following prediction. It is reasonable to assume that there exists a generalized cost associated with establishing or changing a particular promise. Many business promises are in fact commitments, i.e. promises that demand a non-returnable investment of resources such as time or money to make. This means that a business will often prefer to avoid such impedences to workflow, choosing simple promises of low cost, especially in the case where the details of promises have to be modified on the fly. This is a strategy for aligning with productivity.

This simple prediction must also apply equally to all kinds of promises within a business chain, no matter whether they come from humans or computers, tools or software. One might expect then that a business would choose a certain software package over another if it were easy to learn and install, rather than a different one that was highly specialized (making many more specific and specially customized promises) but is costly to introduce and learn. There might be exceptions, of course, in which one could trade a loss against another promise of value (such as a special discount, or subsidiary benefits), but the simplest prediction we take from promises is that the cost of business process is related to the *network complexity* of the promises made and the cost of change (teardown and rebuild) within it, i.e. "change-management" in ITIL parlance[13], [14]).

VII. Discussion and conclusions

The inductive pattern shown in eqn. (6), reveals a grammar – we might even call it a law (part of an "the end-to-end" law). What we see from this grammar is that each agent I_i must make service promises to the business-promiser B, whenever B intends to make a promise to its customer using the intermediaries. Also, each intermediary must have the same kind of agreement with its predecessor. Thus interestingly this exercise shows us how a chain of command emerges from this necessity, through the appointment of the end points in the chain. Moreover, it shows that the intermediate agents have their place in this chain and are fully able to sabotage the business agent's promise. The resulting map helps more pragmatically to plan the necessary Service Level Agreements needed between the parts.

What is possibly counter-intuitive here is that these promises point back towards the origin of the promise B, not forward in the direction of the workflow. In fact, any promises that happened to be made to C by the intermediaries would be entirely irrelevant from the perspective of ensuring a promise from B. This is because we have taken the viewpoint of the relationship between the business and the customer. These are the end points of the chain and they struggle against the basic limitations of promised behaviour, which can be expressed simply as follows: *wishes and promises do not flow unimpeded through intermediaries*.

The discussion in this paper aims to show how the valuation of promises can inform us about business value. By attaching these values to promise theory with its parsimonious assumptions, we gain the leverage of its simple formulations and patterns. The rules of promise theory embody real world limitations that we have to find ways around. The strength of the model lies in forcing us to confront these limitations.

The example of end-to-end services provided shows us how to apply a simple, essentially algebraic rule to model a very common and important component in business process. To make the final step of connecting this to IT management, we only have to see the intermediaries as technological subsystems with operational policies represented as promises. Thus we reduce IT systems just to agents that make promises. How we then tune those promises to maximize their value to the business is the answer we seek. This probably requires the addition of a model for how a valuation is actually made – an issue which we can gladly defer to another occasion.

To fully realize this programme of steps in a practical case, it might be necessary to take each technology as a kind of black box, and open it up to see the many promises within that allow it to make promises without. This leaves plenty of challenges for future research.

The "end-to-end law" itself deserves further discussion and a more rigorous presentation on another occasion. This could include the case in which promises are only kept with a certain probability; it could also be adapted to encompass systems that piggyback on networks, such a grids. These are topics for future work.

Finally, let us end by noting that such an important topic as the fusion of business modelling and IT management is unlikely to be solved by a single paradigm. Multiple points of view are always an advantage. We can hope, however, that businesses themselves are open-minded enough to permit the use of sufficient abstraction in addressing this issue. It is only through approximation and idealization that we can hope to make progress in describing the science of human-computer behaviour.

Acknowledgments

I am grateful to Siri Fagernes for helpful discussions. This paper was inspired by the presentations of Jacques Sauvé and John Wilkes at the 2006 BDIM workshop in Munich.

This work is supported by the EC IST-EMANICS Network of Excellence (#26854)

References

[1] H. Chesbrough and R.S. Rosenbloom, "The role of the business model in capturing value from innovation: Evidence from xerox corporation's technology spinoff companies," *Industrial and Corporate Change*, vol. 11, no. 1, 2002.
[2] K. Gaarder, "Business models – what are they and how do we design them?," Tech. Rep. R21/2003, Telenor Research and Development, 2003.
[3] J. Gordijn, H. Akkermans, and H. van Vliet, "Business modelling is not process modelling," 2000.
[4] C. Bartolini and M. Sallé, "Business driven prioritization of service incidents," in *Proc. 15th IFIP/IEEE Distributed Systems: Operations and Management (DSOM 2004)*. 2004, Springer.
[5] M. Sallé and C. Bartolini, "Management by contact," in *Proceedings of 8th Network Operations and Management Symposium NOMS 2004*. IEEE/IFIP, 2004.
[6] J. Sauvé et al, "Sla design from a business perspective," in *IFIP/IEEE 16th international workshop on distributed systems operations and management (DSOM), in LNCS 3775*.
[7] D.A. Menascé and V.A.F. Almeida, *Scaling for E-Business: Technologies, Models, Performance, and Capacity Planning*, Prenctice Hall, 2000.
[8] M. Burgess, *Analytical Network and System Administration — Managing Human-Computer Systems*, J. Wiley & Sons, Chichester, 2004.
[9] Mark Burgess, "An approach to understanding policy based on autonomy and voluntary cooperation," in *IFIP/IEEE 16th international workshop on distributed systems operations and management (DSOM), in LNCS 3775*, 2005, pp. 97–108.

[10] M. Burgess and S. Fagernes, "Pervasive computing management: A model of network policy with local autonomy," *IEEE Transactions on Software Engineering*, p. (submitted).
[11] M. Burgess and S. Fagernes, "Voluntary economic cooperation in policy based management," *IEEE Transactions on Network and Service Management*, p. (submitted).
[12] Richard Layard, *Happiness: Lessons from a New Science*, Penguin, 2006.
[13] Office of Government Commerce, Ed., *Best Practice for Service Support*, ITIL: The Key to Managing IT Services. The Stationary Office, London, 2000.
[14] Office of Government Commerce, Ed., *Best Practice for Service Delivery*, ITIL: The Key to Managing IT Services. The Stationary Office, London, 2000.