Models in Business
Experiences from the Field

Coen Suurmond
RBK Group, Keulenstraat 18, 7418 ET Deventer, The Netherlands
csuurmond@rbk.nl

Keywords: Business Modeling, Software Design, Semiotics.

Abstract: This paper reflects on the developments over the past decades in business modeling and software design in the context of a very specific niche market. It represents my personal views and supported by theories of the firm and semiotic theories. In the end, the paper argues against a reductionist approach to developing information systems for companies, and pleads for a dissociation between formalised views and formalised modeling on one side, and business views and business modeling on the other side.

1 INTRODUCTION

The continuing theme in our company for nearly four decades of building systems for the food processing industry has been an orientation on business value, on data quality and on the fit of our solutions to the practical circumstances. This orientation did not originate from marketing considerations, but from our companies' background in both designing production facilities and building control and registration systems. The founder of the company, Hans Kortenbach, had a strong drive to combine his technical acumen and business knowledge to design innovative solutions for production processes, and our IT solutions had to make true on his promises to the customer and his intentions at design time. In other words: the solutions just had to work under practical (and often tough) circumstances.

Of course, during that time a lot of things changed in our company. The rapid developments of IT technology (both in hardware and in software development tools) were accompanied by a more and more reductive approach to information and information systems. Our thinking about the nature of the firm, about the nature of information in business processes and about the nature of IT systems moved in the opposite direction: information system development should start from real world business and its processes with their information needs, accommodate heterogeneity of information carriers, accommodate irregularities and possibly inconsistencies, and use IT systems only where those systems have added value. Reductive thinking about information and meaning that comes with IT oriented projects should be avoided.

In the paper the history of our developments in doing software projects, our view of organisations and our approach to modeling processes will be discussed.

2 EARLY YEARS

In 1978 I built my first commercial-use software for a tulip grower / trader. This was an invoicing program which used a manually entered order header data, customer number and order line data (item number – quantity – unit price) to produce a paper invoice. Master data for the customers and items were retrieved from a mini-cassette tape. This invoicing program saved administrative work and reduced the risk of errors. The paper invoice was processed further in the traditional workflow in the paper-based accounting and records keeping. Invoices were not stored in the system: the two Philips P300 Office Computers used here had a working memory of respectively 2kB and 6kB and they had only the mini-cassette drive for external memory. Disk storage was not an affordable option for this kind of computer. This was the automation of one part of a process, one step on from the 'smart' electronic typewriters; it was not an information system.

Within RBK we built dedicated stand-alone systems from the early eighties, but by then external ‘mass’ storage was available on floppy disks (360kB storage capacity). Hard disks were available but still very pricey (and they had a capacity of 10 or 20 MB). Filleting systems were the most important software product: registration of the input and output of individual filleters in a production line, with a weekly payment based on the volume produced and the efficiency achieved. Later this was extended with quality registration (e.g. fish bones found), giving a penalty in pay when the number of quality errors exceeded a norm. These programmes operated in a fixed weekly cycle which finished on Friday afternoon with the payment calculation for each filleter. A connection to the further financial processes was realised by generating the weekly results with such a lay-out that the data could be checked quickly and easily and typed over quickly and reliably by accounting. In some individual cases an intermediate file was created that could be read in by accounting. After all procedures had been carried out at the end of the week, the floppy disks were erased and the system was prepared for the upcoming week. We did not keep history inside the system. These were our first information systems, although rather limited. The system would provide the users with information about yields and productivity both on screen and on paper.

From the early eighties we built control systems for cooling/freezing processes in production lines and for cold storage warehousing. For the former the most important goals were to minimise the product quality loss and to optimise the energy efficiency of the cooling/freezing process. For example, in belt freezers for fish the product is frozen to a product temperature of -18 degrees Celsius and each individual product is encapsulated in a thin layer of ice. Improving the control accuracy results in significant gains; an improvement by just 0.5 degrees can generate an extra annual revenue of EUR 50 000. The same principle applies to other cooling and freezing processes in production lines. Optimal product quality and reduced weight loss are the key parameters. A similar approach to the cooling of cold stores resulted in power consumption savings of up to 30%.

All these systems were based on fundamental and innovative thinking by Hans Kortenbach, the founder of our company. Taking into consideration (1) the physical temperature processes and their effects on the products, (2) the characteristics and possibilities of industrial refrigeration systems (both equipment and control), (3) issues of product value and business value in relation to the markets at that time, and (4) the interdependence between the three aspects, he designed the installations, and we (the software developers), built the control systems.

2.1 Business Value and Data Quality

For us as software developers, building and implementing our software solutions was rather straightforward at the time. The relation between the business and the software was not problematic: either our software “created” reality, as in the technical control systems for freezing and chilling products, or our software was “just” representing the quantity, quality and yields of one production line. Such was the naiveté of our world as programmers. From a broader point of view, however, the combination of the design of the physical system by Hans Kortenbach and our control systems represented a kind of business process reengineering without thinking in those terms. There was no business modeling, only technical designs that reflected new ways of thinking about business and that took all aspects of the business into account.

A fundamental issue in this early phase was the emphasis on data quality. We were imparted with the importance of getting reliable data into the system, as a prerequisite for the control quality and the quality of calculations and decisions based on information from our systems. In particular, we learned the hard way how much effort goes into creating reliable registration systems for the shop floor. In years since, we often have been astonished by the neglect of this subject and by the sloppiness in thinking about data quality.

3 COVERING PROCESSES

Starting in 1989 we developed new software for slaughterhouses and for control of industrial refrigeration systems. All shop floor production processes in slaughterhouses were covered by our systems, as well as the commercial processes of the invoicing of livestock and for the sales processes. The invoicing system for livestock was very specific and very advanced, reflecting the competitive market in the Netherlands at that time. The commercial systems were AS/400 based, the shop floor systems based on PC’s in a Novell network, and the industrial control systems were based on the combination of industrial control hardware with a stand-alone PC for the user interface and data management. Exchange of information between systems was standard. We were specialising in a very specific niche market where we were acquainted with all key processes. A few years after the start of this development, products for meat processors and for producers of pre-packaged meat for retailers were added to our software solutions. Short lead times, perishable products, variability of qualities and quantities formed the main characteristics of our markets. In subsequent decades, quality control requirements generated by food safety concerns and market requirements by the big retailers could be added to the list of key requirements.

Technically our shop floor systems were event-driven, (semi-)real time with a response time of less than 0.5 sec, and provided with multi-tasking mechanisms within the application. Events were either generated by peripheral equipment (weighing systems, hardware contacts) or generated via the keyboard (input from users). We developed our software in Borland Pascal in a MS-DOS environment with text-based user interfaces. All data was stored in binary files. Each registration would open / modify / close the relevant files, risk for loss of operational data was very low.

In later years, our programming concepts prohibited an easy way to move to the Windows / SQL platform. We needed to have access to the physical world in our systems, and we needed our guaranteed response times for our real time tasks. The Windows environment would shield the hardware from our software. Apart from that issue we did not trust the response times of the databases in those years.

Concepts for individual identification of crates and containers with barcodes were developed in that time, partially as an instrument to facilitate registration processes and production management, and partially as a method for tracking and tracing. Each container would have a fixed identification number, and the tare of the container is kept in the master data, which improved the collection of weight data. We could capture a lot of information connected to the physical unit of handling, and the scan of a barcode is a reliable and fast way of registration. The origins of this concept dated back to 1988, and it paid off very well. We did our first experiments for identification with RF-tags in 1987, but this concept has one very important drawback: it provides information just for systems, not for people. And on the shop floor visual information is important. Information flows for the shop floor must be designed taking all kinds of information for the shop floor into account, and not only deal with information in computer systems.

3.1 Business Viewpoints

The development of our systems was based on very close cooperation with our users. The customer would express a problem or wish, we would look into it and together we would discuss and try solutions. Sometimes this resulted in a prolonged iterative process, finding out what was really the case and finding out about side effects, adapting the problem, and so on. In a few cases we would get the feeling that creating a satisfactory solution was a kind of a random walk through possible problems and possible solutions. Fortunately, more often than not it was a more linear exploration through different viewpoints and different contexts. As a specialist and supplier of standardised software in this specific market we would try to satisfy two objectives: (1) the separation and expression of the general problem abstracted from the concrete questions and the specific circumstances of the individual customer, and (2) the identification of the proper interests of the different stakeholders. The term ‘proper’ is important here: a lot of people will express many interests (and sometimes provide detailed directions for solutions), but only interests related to the constructive role of the person in the business process are to be considered proper interests. In a nutshell, the steps in finding and verifying solutions were: Identification of the business processes, identification of the roles of people and systems in the respective processes, identification of the specific questions, generalisation of the questions, finding solutions to the generalised questions, making the solutions available to the specific context, and finding out if every proper information need is met.

Explicit modeling of business was not an issue at the time, but for myself thinking about businesses and organisations was very much an issue in these years. During my education and in the first 10 years of my work I was strongly oriented on theories of the organisation as a means to analyse and understand the functioning of an organisation. Herbert Simon with his model of bounded rationality (Simon, 1976), Jay R. Galbraith with the concept of slack in processes (Galbraith, 1973), and Henry Mintzberg with his Structured Organisation (Mintzberg, 1979) were important sources. In studies of organisations the difference between the formal organisation and the informal organisation was of course a major theme. To me this was an easy and a-theoretical escape explanation. Organisational theory would provide explanations about and the rationale behind the formal structures of the organisations; any deviation could be explained away by referring to the irrationality of human behaviour and to the informal social structures in an organisation. This was an unsatisfying state of affairs for me.

A project for a producer of pre-packaged meat products changed my thinking about business and organisation. The company was led by a dominant owner/director and had a very flat organisation. In the management positions would be either trusted old hands with a fair amount of leeway to make decisions, or employees who had a more or less token position without discretionary powers. Old hands would be overruled occasionally, the others frequently. Operational decisions regarding production and distribution were more based on experience and organisational patterns than on organisational roles. Operationally, this was a sound company with a good reputation and with good financial results. It was a well running and responsive organisation that allowed the director / owner to realise his commercial vision.

The breakthrough was partly triggered by a question a colleague asked me during the project: why do you think that this company gives us all this money? What do you think that the company wants to achieve with our systems? I then started to think in a different way about how the functioning of an enterprise should be understood. Not the organisation of an enterprise should be the starting point, but its markets and products. The characteristics of the markets and products determine the behaviour and the business processes of an enterprise and the (formal) organisation is a means to stabilise the business processes. I started to think and analyse from the opposite direction, outside-in instead of inside out. And, as a consequence, I started to look for the foundations for information systems in the business processes, instead of in the organisational structures.

4 YEARS OF RENEWAL

The period started around 2001 with the conversion of our software environment from MS-DOS to Windows, which brought changes in programming language, data management, and user interface. It also meant the replacement or abolishment of most of our software patterns, established and fine-tuned over a decade. These patterns might be either ‘just habits’ or skilfully engineered fundamental solutions for basic problems. Patterns for dealing with the multi-tasking and real-time aspects of our systems belonged to the latter category. We had to think of new ways for coordination between our systems and the physical world, and the coordination between processes in our software.

Apart from dealing with the more technical software issues, we used this transition to rethink our fundamental concepts for representing business processes in our software. On the shop floor, we used two basic concepts: the production order and the individual container. Stock management was problematic in our software, a problem which we could neglect for a long time because (1) in production of fresh food, stocks are a minor issue in the business processes and (2) we had made some nice and creative work-arounds for representing fresh stock in production orders or in containers.

We also wanted to solve two conceptual problems in representing the physical flows in our new software. One conceptual problem is specific to our kind of industry, the other is generic. The specific problem is exemplified best by the curing process. The curing of products, whereby the products are biochemically changing over time, can take a few hours (tumbling), a few days (brining) or up to a few weeks (dry sausages). In the processing of herring, for example, the product is successively graded, filleted, cured, frozen, packed, and stored. In curing the herring is put in a sour bath for two or three days, while stirred every 12 hours. The curing process has characteristics of both a production order (semi-finished products are transformed into other semi-finished products) and stocks (products in a storage area for several days). This leads us to the generic problem of a real time representation of a production flow as a concatenation of stocks and production orders: the products are ‘lost’ between the input and output on a production order.

Due to our background and driven by our motivation to represent an uninterrupted flow of goods in our systems, we decided to replace our basic concept of production order by the basic concepts of stock, lot and location. Locations are either “storage” or “process”. An input on a production order is represented as a stock movement from storage stock to process stock, and an output as a stock movement from process stock to storage stock. At the end of the production process, the resulting stock balance on the process stock represents the loss of materials in the production process. These few very simple concepts allowed us to represent any flow of goods, and give us a lot of freedom to model the flow of goods in a concrete project.

4.1 Business Models

In this time we would start projects by a descriptive and informal model of the business processes at the customer, supported by a few generic business models for typical process patterns. It was more or less a model based approach along the lines of the concept of Max Weber of the ideal type. Ideal in his concept does not denote how the world should be, it does not mean perfection. “Ideal” in ideal type is a construct of the mind, it is logically coherent idea (model) about some part of reality. An ideal type, therefore, can be a useful instrument to look at specific business processes to compare the ideal type with the actual processes. Differences between them should be analysed to find the causes or reasons. Sometimes they are caused by unchangeable circumstances, sometimes they are there for a good reason, and sometimes they represent patterns evolved over time, either better left in place or detrimental to the process and to be erased. But the first step always is to try to find the possible rationale behind the specific practices.

My way of thinking about firms changed further in these years. Not the organisation, but the markets and products would now be my starting point in the analysis of a firm. An understanding of the markets and the products of the firm provides both background and norms for the analysis, understanding and evaluation of its business processes. The formal organisation was increasingly side-lined as a peripheral phenomenon. This approach was supported by the study of works about the theory of the firm (Coase, 1937; Kay, 1993; De Geus, 1997) and about knowledge in organisations (Weick e.a. 2001; Patriotta, 2004; Boisot, 1998). The study Thought and Choice in Chess (De Groot, 1978) about human problem solving and especially about the role of the perceptual processes of the expert was important for the importance of intangible patterns in business processes.

Another line of study was the theoretical semiotic analysis of signs, sign systems and interpretation processes (in theory), together with the practical analysis of how individuals work with information in business processes and how emerged patterns give stability in working practices. How do individuals deal with regularities (day-to-day patterns) and with irregularities (both recurring and truly incidental incidents)? A lot of relevant information in business processes is either background routine or background knowledge, both for regular situations and for unforeseen situations.

Increasingly I became aware of the limitations and drawbacks of rational-mechanistic approaches of business processes and information systems. Of course, rational models are necessary as a means for understanding and communicating. Models are important for analysis and can be useful instruments for change. Software systems incorporate models of reality. The danger lies in the inversion of the relation between model and reality. At the start of the project the model is a representation of reality, and at the end reality is considered to be an implementation of the model. Misfits between model and reality are at the end of the day regarded as problems of reality to be corrected, the model is rational and “true”. Incidentally, this kind of problem is of course very old. Many discussions between accounting departments (‘bean counters’) and operational departments can be traced back to this type of argument.

5 HETEROGENEITY

In what can be considered as the fourth phase of our information systems we moved into heterogeneous system landscapes (to borrow a term from German). A first example is the replacement of our systems for slaughtering and grading processes. Our first system in this field dated back as far as 1987, and from that starting point it grew out gradually to octopus-like structures. Two decades of meeting a variety of information demands in one monolithic system will result in a lot of add-ons. The old system was (and still is) very stable and dependable, but increasingly difficult to adapt and maintain. A further major drawback was the dependence on the one programmer who had originally developed the system and adapted it since, and who was the only one with the knowledge and experience to support the system.

The objectives for our new system were: (1) replacing the old monolithic and entangled system serving heterogeneous needs (physical input/outputs, real-time aspects, user interface, data management, decision rules) by a heterogeneous landscape with dedicated, single-function subsystems; (2) independence of support by individual persons with special knowledge; (3) a clear overarching model, understandable to business people without technical knowledge; and (4) full specification of all information flows, their effects in related processes and their origins in the production lines. The latter objective is not realistic in general, but in this case attainable because the business domain is highly specific. Further, it is important because the use of terms in this domain can be highly confusing and coloured by local habits.

This resulted in a model with four different subsystems. The first subsystem handles all physical aspects and tracks the movements of all individual pieces of meat in the conveyors, the second subsystem handles the user interfaces (touch screens) in the production lines, the third subsystem is responsible for data management and decouples the real time world from database actions (making response times independent of possible lateness of database transactions), and the fourth subsystem connects the other three and handles all business rules. The very knowledgeable employee who had developed the system from its origins some decades ago to the existing system would be the developer of the first subsystem with the real time and physical issues. He could share his knowledge in this field with his technically oriented colleagues. Informational aspects would be handled by other people, and this was made possible by the full specification of all information flows.

In another interesting recent project we developed a control system for individualised deboning processes. Each individual piece of meat produced on the new production lines would be traceable to the original animal. Depending on the demand of finished products and depending on the individual characteristics of the raw material the system will decide which finished products are to be produced out of which raw material and the system will show individualised instructions to the people in the production line.

At the start of this project the customer knew exactly what he wanted to achieve (full traceability to improve his market position and improvement of yields to earn his investments back) and had general ideas about how to achieve it. The translation of the general ideas into working solutions was up to the main contractors for all physical equipment and the IT system (us). In this kind of projects the customer is catapulted from a situation under direct control of foremen, with a lot of flexibility, a lot of room for making (and correcting) errors, and a lot of buffers and internal transport into a world that is computer controlled, very straightforward and rigid, and with a very efficient throughput. Preparing the customer for the change is a big challenge for several reasons. The customer is used to make snap decisions on the work floor, in the new situation this is not possible any more. Once the quality grade is assigned to a piece of meat (before processing), all decisions are computer-controlled. The only human decision during the processing and packaging of the meat is to reject meat and take it out of the line, which should rarely happen.

To design the system and to prepare its configuration asks for a lot of information from the customer about his current processes, which comes mostly in a highly unstructured way with a lot of exceptions, a lot of imprecise terms, and a lot of qualifiers as “normally”, “basically”, “mostly” or “at least, that should be the case”. Finding understandable and verifiable models in such a project asks for both creativity and background knowledge. The latter is not only important for asking the right questions and understanding the answers, but also for listening to what the customer is not saying.

5.1 Process Logic and Real Business

In the projects mentioned above the modeling phase was not only based on direct observation of visible processes or on interviewing the customer to inform the analyst about his processes. Rather, it was based on a two-step analysis where in the first step a generic model of the underlying and invariant processes was obtained by logical analysis and background knowledge, and in the second step the actual customer’s business processes were modeled. The invariances of the model in the first step are partially determined by the characteristics of products, partially by the characteristics of market conventions in dealing with customers, and partially by social and legal norms belonging to that kind of markets and products. The first kind of invariance is more stable over time than the second as markets, norms and regulations will change over time, but at any given time any company that serves a certain market (in a certain country) must obey the rules and conventions of that market.

The generic model is most stable in its ontology and static structures. In a market-oriented company, demand expectation will always be captured and be translated into quantities to be produced, and via production planning be translated into demand of raw materials and resources. Also belonging to the first business model, but possibly less stable over time, are the dynamic aspects of the model. The market for pre-packaged fresh food has typical lead times and a typical planning/production cycle that can be observed by every company in that market. Lead times may gradually shift a bit over time due to market expectations and due to new packaging methods that prolong shelf-time, but the general dynamics of the planning/production cycle will be unaffected and stable.

In the second modeling step the business processes of the specific company are analysed and modeled against the background of the first model. The first basic assumption is that the second model can be mapped on and abstracted to the first model (with a loss of information), and the second basic assumption is that this mapping is not trivial and certainly not one-on-one. Some elements of the first model may be completely implicit and "invisible" in the second model, some elements may be combined and some elements may be differentiated. Interaction between elements may be consecutive in one company, iterative in a second company and even seemingly reversed in a third company. In the latter case, the planning cycle might start at the raw material level (as a bottleneck), taking demand for finished products for granted. In this situation the demand for finished products will be implicitly translated and generalised by the planner into raw material demand, and checked later on in the planning process.

6 DEVELOPMENTS OVER TIME

The continuity in the business processes of our customers is found in the physical processing of fish and meat into products fit to be used for further processing or to be consumed. The anatomy of fish and meat has not changed over the last century, nor have the basic ways of deskinning, deboning, cutting, and portioning. What has changed much since 1990 are conservation and packaging techniques (prolonging shelf life), tracking and tracing requirements, branding of products ("Welfare" or "Good Farming" products) and market demands (shorter lead times, vendor managed inventory). The first change brought more flexibility to the business processes, all other changes brought additional information requirements and the necessity to separate and monitor more and more different physical product flows.

The two basic ways to respond to the changes in the environment are (1) to consider it as a burden and to try to meet the extra demands at minimal cost, doing just enough to satisfy the specific requirement of the specific stakeholders; or (2) to let the externally triggered change induce internal improvement of processes. In the second response the challenge is to use new requirements on information, induced by one stakeholder, to reflect on the essentials of the processes and the information and to generate the most value of the change for all stakeholders. In particular the extra information needs generated by tracking and tracing regulations can be used to improve production management and control. Business processes are rarely changed in fundamental ways, but rather adapted incrementally by fundamental analysis of the value of the information involved for all stakeholders.

For me, working with business models over time shifted from a latent background notion via heuristic models to ideal-types. Later on, I started to use business models in two different ways: (1) as models representing the process logic of the underlying business processes of a typical company in a certain market, and (2) as a description of an actual company with its idiosyncrasies. The first model supports software development, as it represents basic business functions and relations and as it is specified in formal terms. The second model supports information system development (configuration of the software being part of it), as it describes the real company in its specific environment.

In my view, business modeling for information system development too often tend to mix the two uses of business models. As a result, it is reductionist in two ways: (1) it reduces real process structures to formal schemes, (2) it reduces information to computerised data. This reductionist business model is then projected onto the real company with its real business as the model to be realised.

REFERENCES

Boisot, M. H., 1998. *Knowledge Assets*, Oxford University Press, Oxford.

Coase, R. H., 1937. The Nature of the Firm. *In Williamson, O.E., Winter, S.G., 1993 The Nature of the Firm;* Oxford University Press, Oxford.

De Geus, A., 1997. *The Living Company,* Nicholas Brealey Publishing, London.

De Groot, A. D., 1978. *Thought and Choice in Chess,* Mouton Publishers, The Hague.

Galbraith, J. R., 1973. *Designing Complex Organisations*, Addison Wesley, Boston.

Kay, J, 1933. *Foundations of Corporate Success*, Oxford University Press, Oxford.

Mintzberg, H., 1979. *The Structuring of Organisations*, Prentice Hall, Englewood Cliffs.

Patriotta, G., 1979. *Organizational Knowledge in the Making*, Oxford University Press, Oxford.

Simon, H. A.., 1976. *Administrative Behavior*, The Free Press, New York, 32rd edition.

Weick, K. E., Sutcliffe, K.M., 2001. *Managing the Unexpected*, Jossey-Bass, San Fransisco.