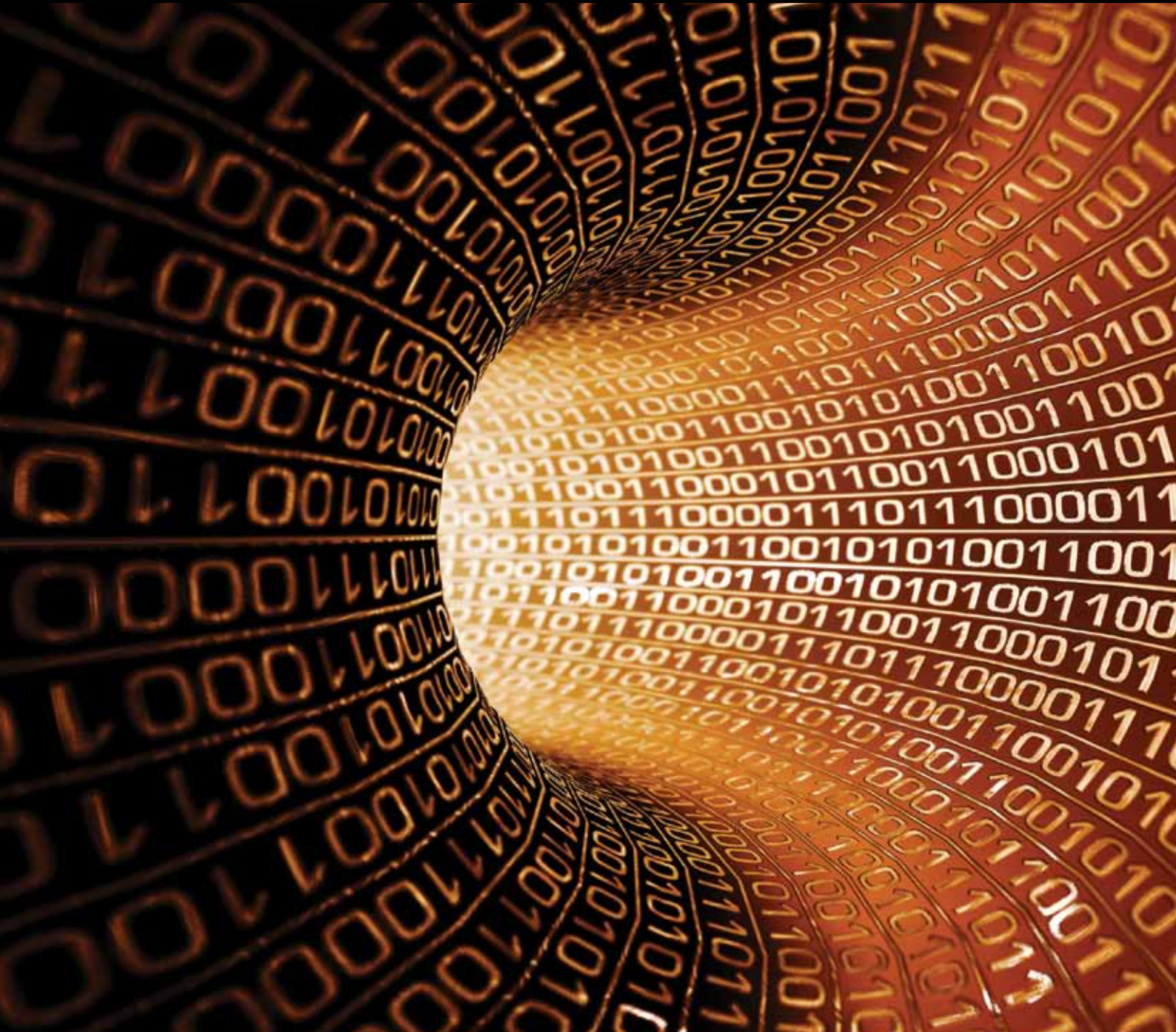# OBASHI White Paper

Understanding how business and IT fit together

By Simon Seow

Introduction

My first impression of OBASHI was one of jaded disinterest.

On the face of it, there is nothing fundamentally new about OBASHI. In fact, seen from the perspective of established frameworks and methodologies perspectives, OBASHI appeared oversimplified and burdened with rules that seemed to be superfluous statements of the obvious.

So, discounting sadistic intent, why am I suggesting that it is worth your while to take a look at it and to seriously consider its use for your organization?

The reason is, that this is one case where subsequent empirical evidence has altered an initial impression, and this paper provides a glimpse of some of the thought processes and experiences that led to my reevaluation.

In this paper I will firstly describe the basic ideas behind OBASHI, making some comparisons with older, established techniques and frameworks where that will help to get a quick handle on OBASHI. Secondly, I will go on to describe my experiments and experiences with it that have convinced me of its tremendous value as a technique/tool for any organization with significant investments in IT. Lastly, I will suggest some ways where OBASHI can be effectively used by itself, and as a small but important part of much larger initiatives.

Simon Seow

OBASHI is an acronym

OBASHI is acronym for Organization, Business (Process), (Software) Application, (Operating) System, Hardware and Infrastructure.

The idea is to classify assets and resources, all referred to as "elements", under one of these six types referred to in the acronym, and to define their interrelationships (the existing and/potential link between the elements), on a diagram known as the B&IT diagram. These six types of elements also have a natural layer relationship with each other, as can be seen from figure 1.
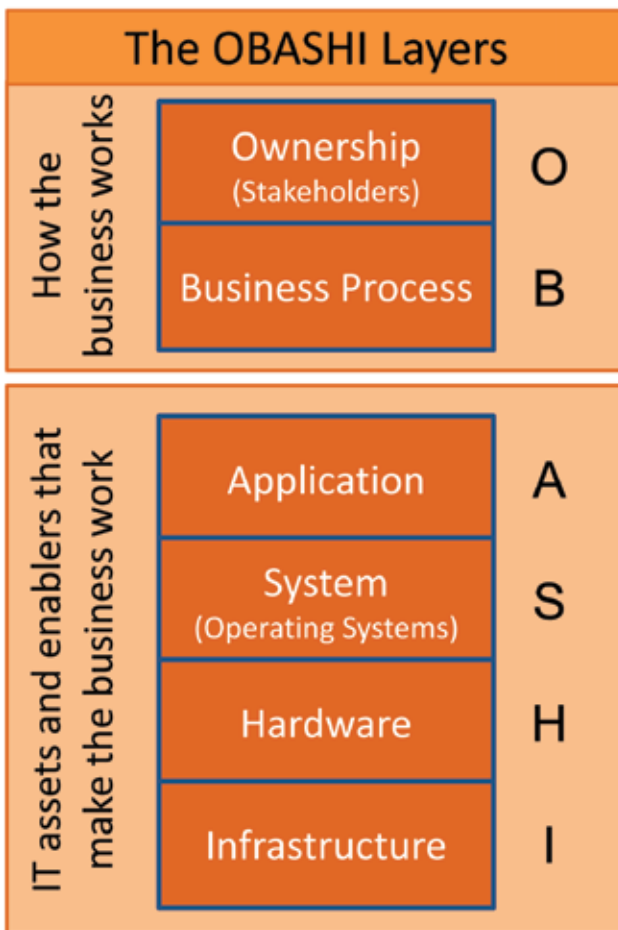

Figure 1 shows the overview architecture of the B&IT Diagram.

LAYERING OF ELEMENTS

Every part of a system, whether it is a stakeholder, a process, or a technology resource can be classified under one of the six OBASHI element types. The total collection of elements of one type are assigned to one of the layers in an OBASHI

diagram. You can think of each layer of an OBASHI diagram as a kind of swim lane in a swim lane type diagram.

OBASHI asserts that the lifeblood of business is the flow of data through various channels formed by connected business and IT elements. Nothing new about this assertion, so far.

Although previous business process mapping techniques have had the notion of modelling a string of business processes linked together in a chain, weaving between swim lanes, the premise of OBASHI is that we need to include the owner element and four types of technology elements as well as business processes. These six types of elements should be distinguished so that it is clear where responsibility lies, and so that problem identification and opportunity analysis can zero down to specific elements that can then be acted upon. After all, a problem or a re-engineering opportunity could very well be identified by someone in one part of the organization (e.g. Business Analyst), while the critical point of failure or opportunity may be under the control of another (e.g Infrastructure Manager). It is like always having the same defined six swim lanes of OBASHI, rather than leaving that decision to each modeller. However the analogy with swim lanes stops here, and we will refer to them as "layers" from now on. The layers are fixed in their position from top to bottom. There is a natural connectivity between adjoining layers and a flow cannot jump over an adjoining layer to connect to a distant layer. It is not expected that, for example, application software can run on, or "connect"
to, a piece of hardware without going through some sort of operating system.

Each layer can be seen as a register of the type of business or IT assets, called elements that the layer represents. Seen this way, the reason for the choice of layer types begin to make sense, as they are also often the ways that assets are classified and tracked, as for example for licensing or audit purposes.

Element positioning rules

One primary aim of OBASHI is to have diagrams that are, as far as possible, self-explanatory. In addition to the named six levels, this is achieved by always positioning an element above or below the elements in adjoining levels (the level above it, or the level beneath it) that it is related to.

This spatial relating of elements gives the viewer of the diagram an intuitive feel for the relationship between elements in adjoining rows, even without the use of connecting lines and arrows. The eye tends to naturally see groups in rows and in columns, even when there are no vertical lines drawn to explicitly define the columns.

The rule is that an element, (drawn as a rectangle of varying lengths) should always be placed in its own row (according to its type, among the six), and should be vertically aligned to the element that it is related to, or that is "using" it.

Often, an element has sub-components, or has direct relationships with other elements of the same type, and these are positioned in the same row, in either top-down or bottom-up fashion. (Examples are like a product breakdown structure, or organization chart.)

Let us look at an example in figure 2 on the next page. Looking at the right hand part of the diagram, we can intuitively read that Jane Smith, the Director of Operations (sitting in the Ownership level), in charge of Logistics of Data Centre 1, owns the business processes Sales Transactions and Address Book (because Sales Transactions and Address Book sit in the business process layer and is placed directly under Logistics in the Owner row). Running our eyes further down the diagram, we can tell that the Sales Transactions process is handled by the SAP General Ledger software, running on the MVS operating system that sits inside their Main Frame Cluster hardware, which is connected to Bridge 4 Live II of Main Backbone Router 1 of the infrastructure.

That's quite a mouthful, just to describe one section of the information conveyed by the diagram.

You can see how the use of the diagram adds significantly to communication efficiency whenever this sort of information is involved.

Notice that some information can be implied, even if there were no connecting lines shown in the B&IT diagram. Position within the same "column" (not an OBASHI term) implies a relationship with other elements in the same column. This is the Spatial Relationship in OBASHI. The length/width of a rectangle representing an element has nothing to do with the element's real size or importance. It is stretched or shrunk on the diagram purely to enable it to follow the rule of aligning itself to the related element above or below it. To be sure, there are a few other rules that can make things a lot more explicit, but the important thing is that anyone, from business or IT, should be able to have an immediate, intuitive, understanding of the sort of information contained in the diagram.

In case you are wondering why there are lines in this particular diagram, they are "Connection Relationships" to show that there is an explicit bi-directional physical or logical connection between those elements.

You will have guessed by now that the B&IT diagram can get very large and unwieldy, as more and more elements are placed in it, as will be the case with complex systems. We can end up with a scroll a mile long.

Having diagrams at different levels of detail can sometimes address this problem, but, as with the old rule in drawing DFDs (data flow diagrams) where we were asked to stick to one A4 sheet of paper per diagram and to use functional decomposition to produce separate diagrams for showing detail, the number of diagrams can get out of hand, as can the problem of trying to
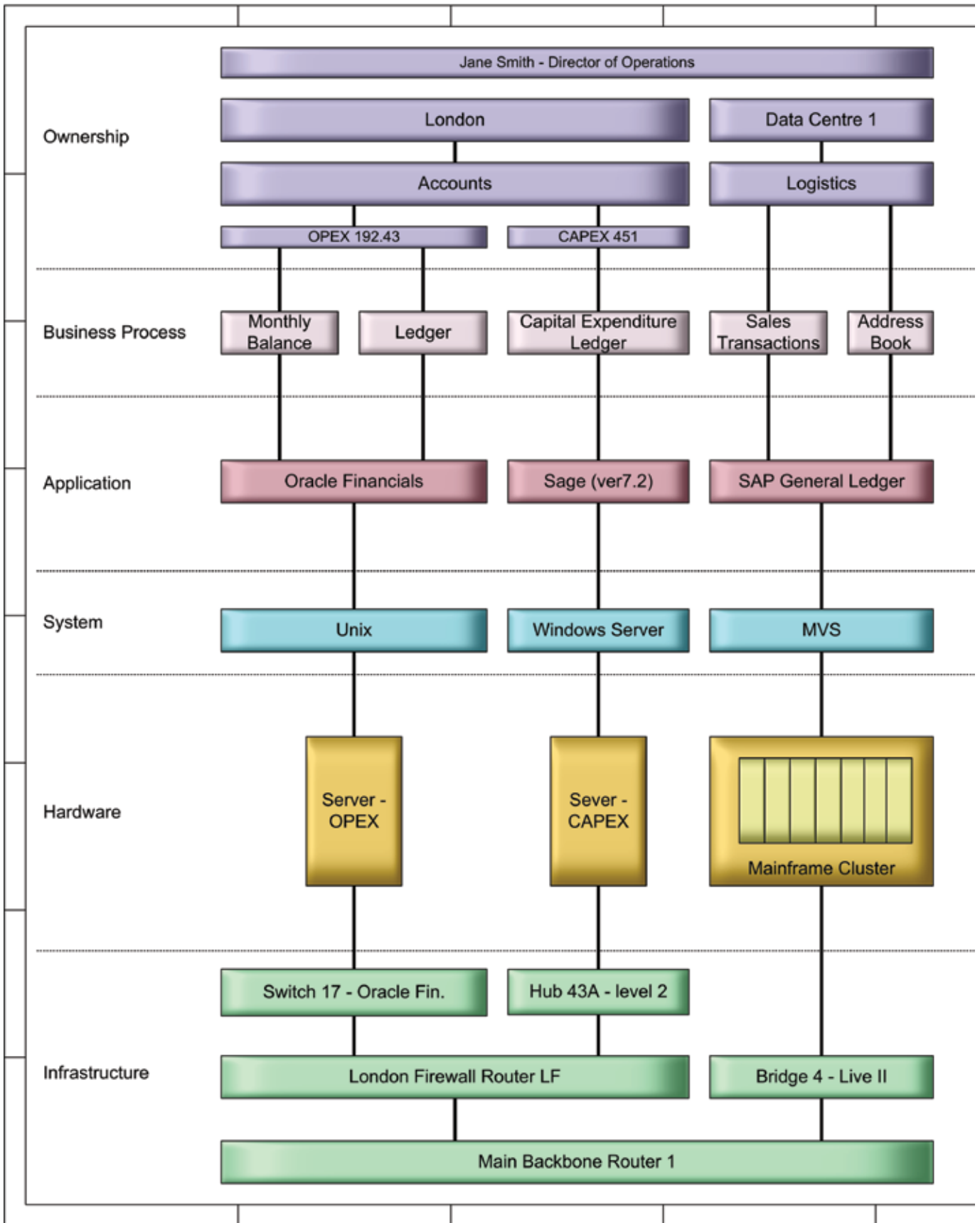
Jane Smith - Director of Operations

Ownership	London	Data Centre 1
	Accounts	Logistics
	OPEX 192.43	CAPEX 451		

| Business Process | Monthly Balance | Ledger | Capital Expenditure Ledger | Sales Transactions | Address Book |

| Application | Oracle Financials | Sage (ver7.2) | SAP General Ledger |

| System | Unix | Windows Server | MVS |

| Hardware | Server - OPEX | Sever - CAPEX | Mainframe Cluster |

Infrastructure	Switch 17 - Oracle Fin.	Hub 43A - level 2	
	London Firewall Router LF		Bridge 4 - Live II
	Main Backbone Router 1		

Figure 2

relate low-level elements from one diagram to another. (Anyone remember off-page connectors in flowcharts?)

OBASHI neatly handles this with a device called a "Dependency Relationship". It is simply a red colored arrow, drawn, from one element to another element anywhere else in the diagram. This maintains the diagrams integrity when it is desirable to omit some parts of the diagram for the sake of targeting a specific audience with details that are only relevant and of interest to it.

There can/will be more than one B&IT diagram drawn for any particular organization. An element may appear in more than one diagram. A server (say Server007) can, for example, run two different operating systems (say Unix and Win2000), each of which runs several applications programs. In this case, where each application is shown in a separate B&IT diagram with its connected elements in the other five layers, that particular server (Server007) can appear in each of those diagrams. The alternative will be to use OBASHI's "relationship persistence", which means that a relationship on one B&IT diagram holds true across all B&IT diagrams, but you might have to add off-page connectors which necessitates constant cross-references to other diagrams each time they are encountered. The strategy chosen by OBASHI to allow repetition of the same element in different diagrams, or even within the same diagram is not new either. Some DFD drawing conventions make use of such duplication of symbols to avoid spaghetti lines all over the diagram.

DAV diagrams

Dataflow Analysis
SwiftNet Messaging Trade Payments

Number of Steps : 31
Number of Elements : 16
Dataflow Valuation : £17,000,000.00

Notes:
SwiftNet Messaging Trade Payments - Asia Market

Total Latency : 20.864 secs.

Breakdown:
Propagation latency is 2.8 secs.
Queuing latency is 16.1 secs.
Serialisation latency is 1.7 secs.
Switching latency is 48 msecs.
Transmission latency is 216 msecs.

This dataflow comprises 31 steps through 16 different elements. The process path through these elements are detailed below.
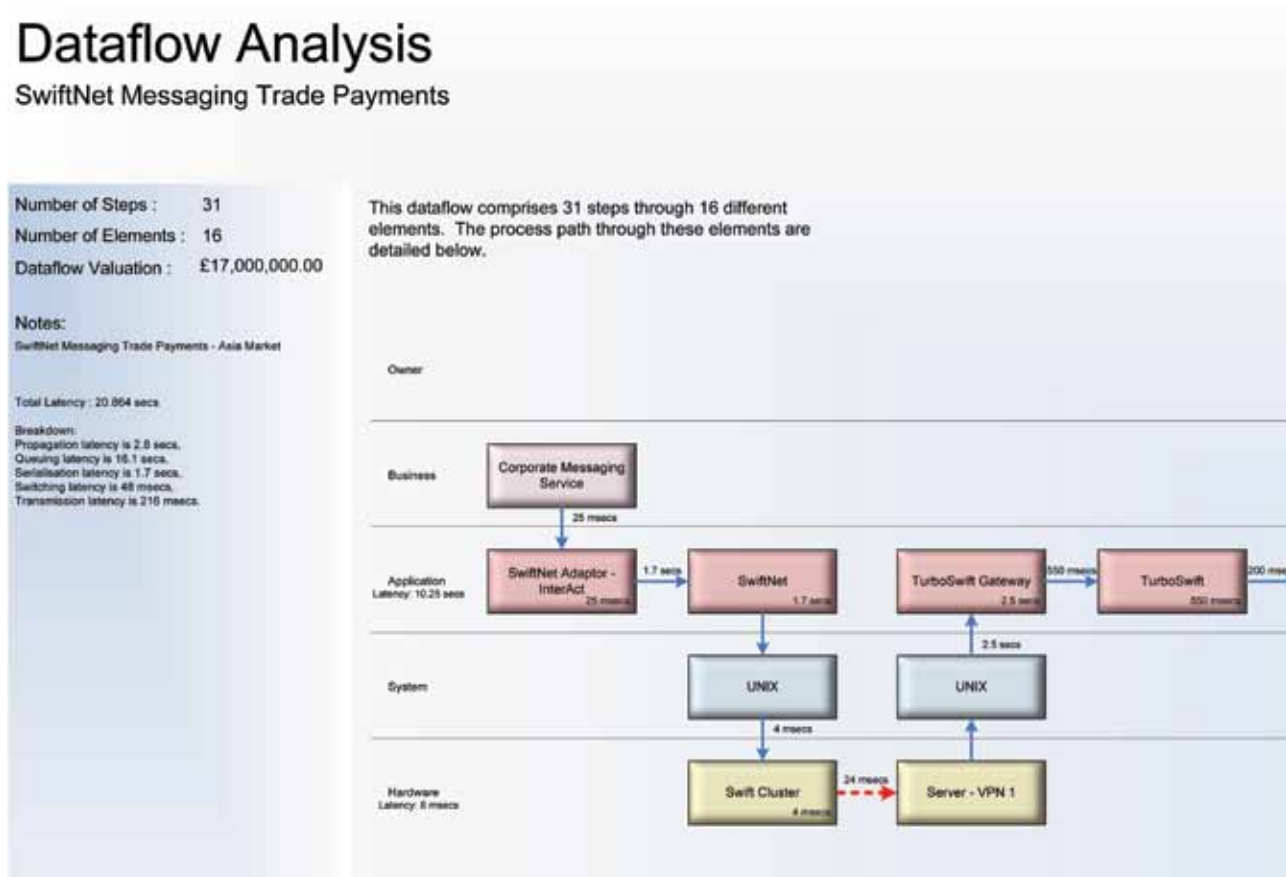
Figure 3

A second type of diagram, called a DAV, or Dataflow Analysis View, is a graphical and statistical document that illustrates a subset of elements, in a pre-defined sequence. The elements in a DAV can be extracted from one or from several B&IT diagrams.

It is intended to show the chain of elements involved in specific flow of data (or a transaction pathway). Adding directional arrows and removing elements that do not contribute to the particular flow under analysis makes the DAV diagram more useful when analyzing a specific flow. This focuses attention to the relevant elements and connections. When specific common attributes are added to each element in the chain, the power of the DAVs are brought into play. Values captured in these attributes can be manipulated for a wide variety of uses. Any kind of attribute deemed useful can be specified for the elements in a DAV.

Figure 3 illustrates the diagram being used for timing and valuation analysis. This is where the tool proves its worth, when you can plug in actual or hypothetical figures to have a much higher degree of confidence in decision making, or in pin-pointing problem and opportunity spots in a system.

There is a big difference between saying "I think (meaning I am guessing) we can save cost by reducing the number of servers we operate", without upsetting our customers and saying,

"We can have the option of removing five servers (not referring to any diagram here) and still keep within our contracted service levels. I have done the calculations based on data available in our DAVs and the figures bear this out."

Again, this simulation capability is not new. Much older techniques, such as IDEF0 did allow for such process simulation, but OBASHI makes it much more feasible to make comparisons of models from different sources because of the few simple rules standardizing the type, positioning and relationship of elements.

My own experiences

Earlier this year, I passed the OBASHI book to a friend who works in a company that takes on IT outsourcing work and asked her to let me know what she thought of it. A fortnight later, I got a call from her and she sounded excited.

"I did a presentation to my management on a new proposal for a client, using the OBASHI diagraming method and guess what? They loved it. I didn't have to explain to them what the diagram meant, and we had a productive discussion around the various configurations options and cost-efficiency implications by changing some of the elements on the diagram. They have suggested that I put up an "as-is" and a "to-be" diagram, with some cost attributes, to show the kinds of savings we can make for the prospective client."

I told her that was exactly why I gave her the OBASHI book.

Subsequently, I have tested its acceptance by incorporating OBASHI into some of my business analysis courses. The feedback was all positive. Delegates all expressed the opinion that OBASHI was something that will be really easy for their co-workers, management and clients to understand and to have a useful discussion around. Not so surprisingly now, some of the most enthusiastic delegates are users of heavyweight methodologies and frameworks like TOGAF. OBASHI will give them the immediate acceptance and recognition while they continue to quietly plod on towards their long term goals.

It dawned on me then, that experience and passion can sometimes be a huge handicap to seeing things as they are, rather than as they should be. I had been blind to the fact that the average person has neither the experience, nor the time and desire, to think long and deep about modeling and methodologies in their quest for a permanent solution to the issues of business-IT disconnect. While we caution against the prevalent instant-gratification

culture, we cannot deny the fact that we need to demonstrate quick-wins to management and clients, in order to gain interest in, and to sustain longer-term initiatives. These quick-wins sometimes cannot even wait until implementation. Management/clients need to be convinced of the value and safety of the proposal up-front. They look for something more tangible than pure concepts or high-level generalizations to get a good feeling that you know what you are doing, before giving the go-ahead to a project. Gone are the days when you can fall back on "Trust me, I am using the best practices enshrined in this 700 page general guideline (which, by the way, is far too complex for you guys up there to understand)."

Diagrams have to show numbers and detail when asked for, in order to be convincing to the ones controlling the purse strings, because at the end of the day the bottom-line is a number. OBASHI fulfills that need.

There are other advantages to the adoption of OBASHI, though I see the ease of use factor as the primary one.

The official reference is published by TSO and therefore openly available to anyone for the cost of the book. Most people should be able to use it after studying the book, although sending your team onto a short course can help you gain a faster start-up. And if training is needed, the method is supported by a worldwide network of accredited, independent trainers that are quality monitored by the APMG. If you subsequently decide to adopt the methodology as one of your corporate standards, there is a certification program available as well.

A software tool is available for download from the OBASHI website (there is a free time-limited trial.)

I recommend use of the good old flip-chart first, until you have got the hang of things. The tool

will be a really big help in getting the most out of the methodology once you are clear how to proceed with the methodology.

Finally, my tuppence-worth of suggestions on how to proceed from here, if you think you would like to research further.

1. Get the book, and go to the OBASHI and the APMG websites for some more formal case studies. Arm yourself with the case studies found there to be ready to answer the "Who is using it" type questions.

2. As soon as you can, (half-way through the book) start drawing up a few diagrams. You probably need to do two or three before you get the feel of the level of detail and the symbol size to use, that will let the diagram fit on a flip-chart (or a slide). Your audience should be able to comfortably "read" the diagram from where they are sitting. The first-cut diagram is a discussion tool and you want to encourage them to ask you "what-if" questions based around the diagram. Do the diagram together with at least one other member of your team. This can be an enlightening exercise when you discover that you need to find out who can provide you with the information you need for the types of elements in the rows that are outside your control.

3. Use a B&IT and a DAV for a presentation to management/client whenever you have the opportunity. Your initiative will not last long if the diagrams are in your private collection.

If your organization is already well into frameworks like the Zachman Framework or TOGAF, give a little thought to how you can immediately use OBASHI to do the do the work required by them. Frameworks traditionally do not, and cannot, prescribe how the detailed work should be done. OBASHI will fit in very nicely at this level.

Look for other areas where modelling, blueprints, system specification, etc are used, and see if OBASHI can complement the way things are currently being done, to add speed or clarity.

In summary, OBASHI is an excellent technique to show how IT is part of the business. It can, and should, be used whenever possible, for simulating alternate scenarios, and for pin-pointing parts of a system for error correction, for improvement or for opportunities to exploit. Like all good tools, avoid the temptation to treat it as a silver bullet. Use it in conjunction with existing tools, techniques and frameworks in your organization. With OBASHI, you can use it and reap immediate benefits.

About Simon Seow

Simon Seow is currently in transition between managing a training company to a retired status of freelance writer-speaker.

Born in Malaya in 1950 and receiving his tertiary education and his early working years in the UK, before resettling in Malaysia, Simon continued to pursue his interest in methodologies and standards.

He is an accredited trainer in PRINCE2®, MSP®, MoP®, P3O®, Obashi and of the Zachman Framework® for Enterprise Architecture.

He was an examiner for the BCSs SAEB/ISEB and a regional moderator for the NCC. He had contributed in an honorary capacity to many regional and international standards, professional and industry organizations at board as well as at work-group levels. Simon has spoken at many conferences and has had his own column "Simon Says" in the IT section of the *Star* newspaper (Malaysia) where he enjoys providing insight and challenges to business and technology issues.

He can be contacted at **simon@theanalyst.com**

Further information

Further information

The OBASHI Manual and Examinations are available from APMG International:
www.apmg-international.com

APMG-International
Accrediting Professionals

APMG-International

Head Office, Sword House, Totteridge Road
High Wycombe, Buckinghamshire HP13 6DG

Tel: +44 (0) 1494 452 450
Fax: +44 (0) 1494 459 559
Email: servicedesk@apmg-international.com
Web: www.apmg-international.com