Global Co-Operation in the New Millennium
The 9th European Conference on Information Systems
Bled, Slovenia, June 27-29, 2001

AN OBJECT ORIENTED ARCHITECTURE MODEL FOR INTERNATIONAL INFORMATION SYSTEMS? EXPLORING A POSSIBLE APPROACH

Hans Lehmann

Department of Management Science and Information Systems, University of Auckland, New Zealand
Private Bag 92019, Auckland, New Zealand
Tel: 64 9 373 7599 8659 , Fax: 64 9 373 7430
h.lehmann@auckland.ac.nz

ABSTRACT

Based on the analysis of four case vignettes the architecture and design principles of international information systems are explored. A two-dimensional topology – suggested in previous research - was confirmed as a useful working paradigm for the architecture of international information systems. In terms of this model, such systems are configured from two elements: 'core' systems (common for the whole enterprise) on the one hand and 'local' systems (different for each site) on the other. One case vignette in particular highlights the logical and organisational difficulties in defining these systems elements. Object orientation is suggested as a fundamental design approach to provide a solution for this problem. Because it enables implementation differentiation and flexibility for future functional changes, it is conjectured that object technology is a useful – technical - development strategy for international information systems. Directions for further research are outlined.

1. INTRODUCTION

Information systems technology is often critical to the international operations of the globally oriented firm, either as the key to its expansion, or even as the main profit driver. However, there is a dearth of academic research into this field. Detailed coverage of the literature on IIS is available elsewhere: Hamelink [1984] covers the early research; Sethi and Olson [1993] give a very exhaustive overview; Lehmann [1997] brings it more up-to-date and Gallupe and Tan [1999] establish a blueprint for future research. Lehmann [2001] shows that most past research into IIS is sporadic and spread over a wide array of largely academic topics. Only in the last few years have researchers begun to direct their attention to the design and development of IIS – examples are King et al [1993], Tractinsky [1995], Van den berg et al [1999] and King et al [1999]. Some of this recent research focuses on the structure and architecture of IIS (e.g. Gibson [1994], Burn et al [1996], Targowski [1996], Grover et al [1996] and Peppard [1999]). While scholarly research into this field is sparse, there is now an increasing amount of anecdotal evidence and technical reports that indicate a strengthening interest by practitioners in this field (as shown by Collins et al [1999]).

This exploratory paper firstly validates a generic architecture common to international systems. Selecting and building an appropriate IT architecture is considered an important building block for the successful

development of any complex system [Earl, 1989], in part because it simplifies the design process. Following this argument, it is then conjectured that defining the generic architecture in terms of object-orientation would have the potential to make the development of international information systems faster and less risky. Finally, recommendations for further research to validate – or otherwise – these suggestions are given.

2. THE ARCHITECTURE OF INTERNATIONAL INFORMATION SYSTEMS

The literature does not clearly identify a generally accepted term for information systems technology applied across borders. Based on [Lehmann, 1996] in this paper they are defined as *'the distributed information systems of a firm that support similar business activities in highly diverse environments, commonly found across country boundaries'.*

The common sense deduction from this definition is the obvious requirement that such systems would have parts that are common to all sites and other parts, which are specific to individual localities. The basis of this concept, i.e. the need for information systems to accommodate differing local circumstances has been established long ago [e.g. by Keen et al., 1982] by viewing information systems as a conglomeration of 'common core' applications with 'other components' or 'local alterations'. There has been little further development of this model as far as the functionality of application systems is concerned and researchers conclude that "the literature offers little guidance for...local versus common applications"[Ives et al, 1991]. In the following section, a generic architecture is introduced as a working model for further discussion. It is demonstrated on a number of case vignettes.

3. A GENERIC ARCHITECTURE MODEL

Building on the author's 'lived' experience in the development and implementation of international information systems, a two-dimensional topology has been postulated as an architecture model for international information systems [Lehmann, 1996, 1997]. The topology consists of a common 'core' and 'local variations' of such a system. Each local implementation of the international information system consists of the core and varying configurations of local variations (shown by the shaded 'local systems' in Figure 1 unten).

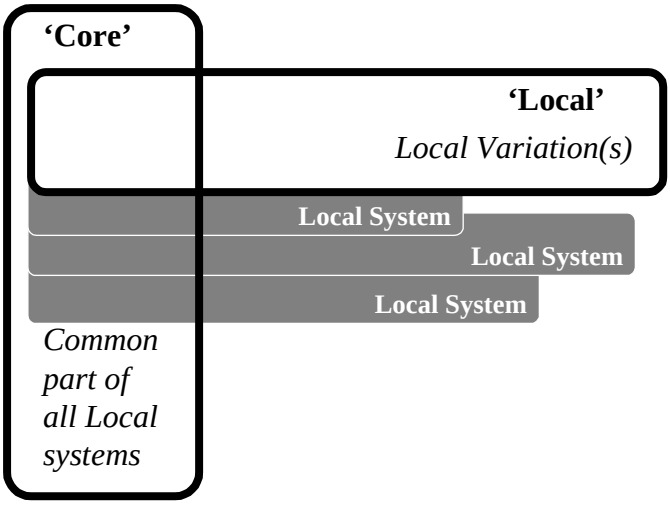

Figure 1: *Conceptual architecture model for international information systems*

The following sections contain brief descriptions of case vignettes, which were developed with a grounded theory approach [Glaser & Strauss, 1967]. Grounded theory emphasise that not just large research cases, but also small informal cases and 'anecdotal comparisons' are a legitimate source for the conceptualisation

of theoretical constructs [*ibid.* p67] as long as they are the researcher's 'lived experience'. The stories are followed by their interpretation in terms of the postulated generic architecture. This is recommended grounded theory procedure this to see if a theoretical concept 'works' [*ibid.* p93].

All case firms are disguised. The first three vignettes are described in more detail in Lehmann [1996] and are presented here in abbreviated form. Whilst they all stem from the mid-90s, their architectures are unchanged in principle – although in some cases they now use different enabling technology. The fourth vignette is the summation of the architecture perspective of a large case (set in the late nineties), where opposing visions of an IIS's architecture contributed to it never being implemented. This not only emphasises the importance of an appropriate architecture paradigm, but it also points to how the use of object technology could have ameliorated or eliminated this destructive conflict. The case is contained in more detail in Lehmann [2000a & b].

Case Vignette A: The Leasing Subsidiary of an International Bank

This UK firm with branches in the USA, the UK and three European countries was acquired by an Asian international bank to give them a base in the leasing business. The central computer into which all offices were linked was in New York. It soon transpired that systems were inadequate for expansion and the hardware was obsolete. A complete re-design of their information systems technology was therefore needed.

The business in the various countries was firmly limited to instalment credit transactions and differed mainly in magnitude. This premise of business being the same did, however, not translate into systems terms. While the lease set-up process was similar, the lease administration part, which makes up more than two thirds of the system, differed significantly from country to country.

To cope with this diversity the new international system had two main components:

- A common leasing module dealt with credit and exposure management; it would set up the lease deals and the receivables stream;
- Local country modules would receive the basic data for input into the country-specific receivables and marketing modules, to cope with local languages as well as differing business practices.

The topology of the new systems was thus characterised by

- A 'core of central application software for those key functions which need to be globally controlled,
- Freedom for 'local' information technology applications, apart from stringent data interface standards;
- core and local systems are of equal size in terms of their functionality.

Case Vignette B: An Australasian Merchant Bank

This New Zealand based merchant bank with substantial branches in Australia and London had expanded rapidly. The business of the bank consists mainly of money market dealing, investment banking and stock broking. Information systems technology at the head office was fragmented and the branches ran odd assortments of software and equipment, loosely linked by public networks. A newly appointed central Treasurer set out to install global systems and controls.

There were two levels of information and systems needs:

- A comprehensive management control system to monitor exposure and risk internationally as well as for each local firm;
- Operations, specifically in the money market and stock broking activities required more systems support.

The resultant international system consisted thus of a narrow 'core' which only contained detailed and stringent data interface and communication standards for input into the Treasurer's own monitoring system. As an incentive to comply with the standards, an electronic mail and bulletin board service was offered on the 'core'. The 'local' systems were seen as entirely composed of the best suited local software, adapting to the diverse environments for money market and stock broking operations, provided they could comply with the data and information interface requirements. The branches would rely entirely on local, external technical and applications support.

Case Vignette C: A New Zealand Commodity Exporting Board

The Board has a virtual monopoly in the purchase of fruit from producers. It owns packing houses and cool stores in New Zealand, operates its own charter fleet and runs a number of European sales offices, while North American sales are controlled through an agency. Asia is an important target for development in the near future.

The Board's systems strategy focuses strong production systems with some marketing modules integrated into them. The international part is, however, the smallest part of the system and consists mainly of common messaging formats. The shipping system is designed to send shipment details and forecasts to the branches who return sales statistics back to the centre. Some of the overseas sales offices use sales order entry systems specific to their environment.

The architecture consists of:

- A - large - 'core' of extensive and sophisticated systems in the production, logistics and marketing (forecasting and decision-support/modelling) areas;
- The messages and the independent sales and administration support systems at the branch offices are thus the only 'local' parts in the topology of this international system.

Case Vignette D: An Australasian Food Producer's Co-op

Similar to the previous vignette, the Australasian Food Producer's Co-operative (the 'Co-op') is a statutory monopoly for the export of its produce from the home country. The UK was its main market until that country joined the EC and severely restricted the Co-op's access. In response, however, the Co-op rapidly built a global network of sales offices. This helped it first to become a global exporter and later, as it included local manufacturing resources, it turned into a truly international operator of considerable size: It ranks 25th among the world's food industry giants. Organised into regional enterprises with great autonomy, it found it difficult, however, to implement the global branding, which its competitors began to use to good effect during the last decade. Some five years ago, a new CEO began a strategic migration towards a 'transnational' global business strategy, balancing the regions' power with a corresponding measure of central control over global product, marketing and branding policy.

The Co-op's IT people misunderstood this move as a return to a 'global' strategy, i.e. one aimed at minimising regional autonomy. For this reason, they set out to implement a globally standard IIS (a large 'core'), with little local input and with hardly any information systems under local management (a small 'local' component), discarding the substantial regional information systems in existence.

Regional/local management had a different view of an appropriate IIS for the Co-op. Their vision was based on their existing systems (i.e. large 'local' components), loosely linked by a common framework for production and marketing planning (a small 'core'). They furthermore interpreted the IT proposal as a covert way to increase central control and resisted this attack on the independence of their local 'fiefdoms' vigorously.

The IT people responded to this strong resistance from regional business management with political manoeuvres. This made the communication between the business and IT more and more antagonistic. Because of their diagonally juxtaposed visions of what IIS architecture would be most appropriate for the Co-op, they went through several iterations of a cycle of business rejection and political reaction from the IT people.

In the end, the CEO was forced to arbitrate in this ever more acrimonious war of politics between the IT and business factions. Realising that the differences were practically unbridgeable, he terminated the project and re-allocated the team's people and technology resources.

4. SUMMARY OF THE CASE VIGNETTES IN TERMS OF THE TWO-DIMENSIONAL TOPOLOGY

The case vignettes seems to confirm that the two dimensional topology is flexible and can accommodate a variety of systems architectures. Variety can occur not only in terms of 'core' versus 'local' across the board (as in the merchant Bank and Commodity cases), but also each 'core'/'local' mix and interface can be defined, maintained and changed in precise response to individual local site requirements – such as in the Leasing case and the business vision of the Co-op case. Table 1 unten summarises the topology and systems element characteristics of the four cases.

A number of architecture models for IIS have been developed in the literature (e.g. Butler Cox [1991], Karimi & Kosynski [1996], Sankar et al [1993], Ives & Jarvenpaa [1994]), which can be reduced to three different ones [Lehmann, 1997]. The two dimensional topology can implement all three: *Centralised* architectures have few, if any, 'local' variations. *decentralised* architectures have a small 'core', reflecting little common functionality and *integrated* architectures (reflecting a more equal balance between central and local business and systems functions) have a varying 'core' to 'local' ratio for each location.

Architecture Elements	***LEASING FIRM***	***MERCHANT BANK***	***COMMODITY BOARD***	***CO-OP IT Vision***	***CO-OP Business Vision***
'Core'	'Medium core'; Selective Customer Data Base; Central Leasing Application for 'Deal-Set-up'	'Thin core'; Data & Information standards and communicatio n standards (stringently enforced);	'Large core''; Large central system at the production sites and head office;	All key business and control functions would reside and be managed centrally from Co-op HQ	Only central co-ordination functions would be managed by Co-op HQ
'Local'	Local variations of the Leasing Receivables and Marketing systems;	Locally selected technology (mostly out-sourced) for all key business functions;	Message formats and protocols for interfacing to smaller, independent local systems;	Only local sales support functions would be managed at regional/local level	All key business and control functions would be managed at regional/local level

Table 1: *Summary of the case vignettes*

Furthermore, the global business strategy, i.e. the balance between local autonomy and global control, seems to be correlated to the topology of the international information systems in the case vignettes. This confirms the influence of strategy on information systems structure conjectured in the literature [e.g. Simon & Grover, 1993]. A topology of large 'local' technology, compared to thinner 'core' components seems to correlate to the higher degree of local autonomy and low level of global control as reflected in the Merchant Bank's organisational strategy. The balance in global vs. local control in the structure of the leasing company is reflected in a medium-to-large 'core' and equally sized 'local' systems. High global control and little autonomy for the branches in the commodity firm describe an architecture characterised by a large 'core' of systems at the centre and only a thin smattering of 'local' systems. The contradictory visions of what would be a suitable IIS structure held by the Co-op's IT people and local/regional management shows up as diametrically opposed. **Figure 2** unten shows the position of the four case vignettes with respect to their topology and global business strategy.

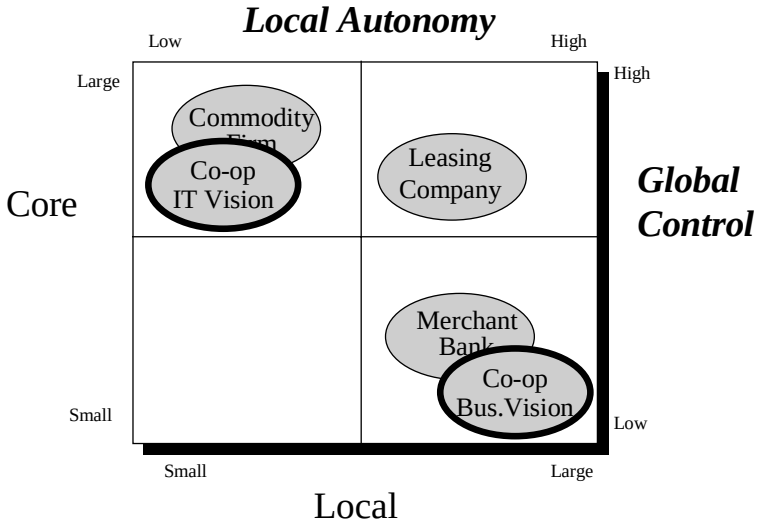

Figure 2: *Link between global business strategy and the IIS topology of the four case vignettes*

The case vignettes furthermore show that two steps in the design of an IIS are crucial: first the definition of the 'core' in terms of infrastructure as well as functionality and subsequently the determination of the 'complementing local' applications. Despite guidance from the global business strategy, designing and specifying the common parts of international information systems in the operational detail required for building and installing them turned out to be very difficult to do in practice. In the Co-op case the determination of the 'core' functionality and where, and how, the 'local' applications complement it proved to be altogether impossible. The following paragraphs discuss this process and suggest a direction for simplifying and error-proofing this process.

5. TRADITIONAL APPROACHES TO THE DEFINITION OF GLOBAL AND LOCAL SYSTEM ELEMENTS

CSC [1995] in a case study of the way in which a number of multinational firms deals with the issues of global information systems versus local business requirements summarise that building systems to satisfy a multiplicity of diverse business needs can take two different forms:

- The 'core' system is formed around the '*lowest common denominator*' of all the requirements, i.e. the sum of all *identical* local and central business needs; in system building terms, however, this can be a disappointingly small proportion of the overall information system;

- The opposite stratagem, in CSC terms 'the grand design', attempts to specify a system which contains all the requirements of all local and central business units and agglomerates them into one information system; in mathematical terms this may be called the '*lowest common multiple*' - and just as such a number can be alarmingly large, so can information systems built along this principle; some of the more spectacular information systems failures fall into this category: during the systems development time the business changed so much that there could never be a 'final version' of the software.

The mathematics simile, however, points to a third possible stratagem for finding common elements among divergent number sets - *multiples of common prime factors*. In systems terms, these would be 'components' in the form of building blocks that would be used to assemble systems. The 'components' would carry the *global* standards, but their assembly could then follow individual *local* requirements. Information systems built in this way would satisfy both 'common' and 'local' needs and would avoid any conflicting trade-off stances altogether.

Such 'prime factors' for the establishment of global commonality can be implemented in three ways:

1. As infrastructure to enable common basic applications (such as email in the case vignette of the *Merchant Bank*); in this way, global standards are implemented in a form which would be immediately useful for the local business unit; using infrastructure in systems design in this way is a much researched topic (e.g. Weill [1992,1993], Weill et al. [1994a/b, 1995,1998] and Broadbent et al. [1997]) and has now become an accepted design principle, especially for web-centric systems;
2. As a 'design template', i.e. a set of design outlines and specifications for the global standard part of an application, from which the individual local systems can be built; the case vignette of the *Leasing Subsidiary* is a variation on this theme, with the template fairly firmly embedded in actual software templates;
3. As software components;

Both design templates and actual software components will consist of data and processes – which clearly and unambiguously defines the resulting conglomeration as an *Object*.

6. THE BENEFIT OF 'OBJECT' QUALITIES FOR 'CORE' SYSTEMS ELEMENTS

Object orientation was introduced into the literature concerned with information systems development some 15 years ago (see, e.g. Booch [1986]). The notion of re-usable and adaptive pre-created parts of programmes proved has gathered a sizable following among the information systems community, but it is not entirely without critique (e.g. Fichman et al. [1992]). Despite this, the fledgling 'movement' grew and the first consolidating works appeared (such as Rumbaugh et al. [1991], Lorenz et al. [1992], Jacobson et al. [1992] and even James Martin et al. [1992]). This was accompanied by research blueprints (such as Monarchi et al. [1992] and Snyder [1992]). Another development beginning to take shape was the notion that any practical distribution of objects relies on conventions, especially modelling standards [Embley et al. 1992]. Cockburn [1993] gives a comprehensive treatment of the impact of object-orientation on the development of information systems applications.

Bringing together the different streams of object-orientation was the establishment of the Object Management Group (OMG) with a view to creating a vehicle for object exchange. This was accomplished with the specification of CORBA, the Common Object Request Broker Architecture [1991]. Further OMG specifications were developed to set standards for modelling distributed software architectures and systems along with their CORBA Interfaces. There are three complementary specifications currently available: Unified Modelling Language (UML), Meta-Object Facility (MOF), and XML Metadata Interchange (XMI). Developed in the mid-90s, they are continuously updated. A further attraction for systems developers is that modelling for object orientation can also be pursued with rigorous approaches (as demonstrated by DeLoach et al. [2000]).

The generic two-dimensional topology architecture for international information systems, postulated above based on the analysis of case vignettes, has two attributes that suggest that object-orientation principles may be of use in the building and ongoing operation of such systems:

- the 'core' elements of an IIS have all the characteristics of reusable objects: they are identical, but need to be implemented in a number of different environments;
- the 'local' variations of the system can either be independent and separate systems altogether or they can be manifestations of the 'core' system elements, albeit with significant local deviations.

For this reason, the following paragraphs examine the potential usefulness of an object-oriented approach for the design of international information systems. Given the high level of abstraction of the architecture model upon which these considerations rest, the outcome of such an examination is likely to be a conceptualisation of sweeping proportions, limited to establishing whether this topic is worth of further, more detailed and empirical investigation.

Three key qualities of object orientation with respect to the common/local issue in international information systems are discussed below and examples from the case vignettes are provided:

1. Objects are defined as *encapsulating* both data and processes/functions in one unit. This combination makes them very useful for vehicles of 'global;' standards, incorporating both data/information standards as well as 'prescribing' standard ways of operating.
2. Objects communicate with other objects using 'messages'. *Polymorphism*, defined as the capability of objects to deal differently with identical messages, is an essential quality for implementing 'local' requirements onto standard processes. Examples are
   (a) The application of direct/sales taxes, e.g. in the case vignette of the *Commodity Exporting Board;* local applicability rules, rates, etc. would be kept with the appropriate module of the local information systems object (such as the local sales order entry system), reflecting the different local statutory and tax regimes;
   (b) If the risk and exposure assessment rules in the 'core' of the *Merchant Bank's* information system were implemented as an object-oriented system, then one such a globally enforced rule-set would have been the compulsory calculation and reporting of the risk exposure of each subsidiary bank deal. Again, the processing modules in the 'message' would act on 'local' and individually different data (e.g. local cost-of-money rates, tax-rebate rules, etc.). **Figure 3** depicts this;

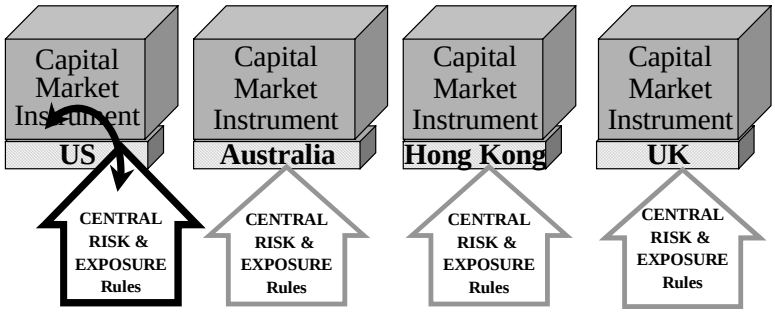

Figure 3: *The standard risk/exposure assessment rule 'message' acts on all objects, taking country variations into account.*

3. *Inheritance* is the quality of objects to structure themselves hierarchically into 'super-classes and sub-classes pass 'down' characteristics (data and/or processes). This has two main uses in the global/local dichotomy:

(a) Consider 'Payments' transactions in the *Leasing Subsidiary* case vignette: whilst the gist of payment processing (application into a ledger, cash-book/bank reconciliations, etc.) is common, the operational detail of the payment process is not; each 'local' object would inherit the common 'core' processes from a standard Accounts Receivable module [the 'super-class'], but implement typical local payment types (e.g. Direct Debits in the UK, Bank-Account-Transfers in Germany, negotiable promissory notes in Italy, etc.) in the 'local' sub-class, denoted as the white squares; Figure 4 illustrates this;

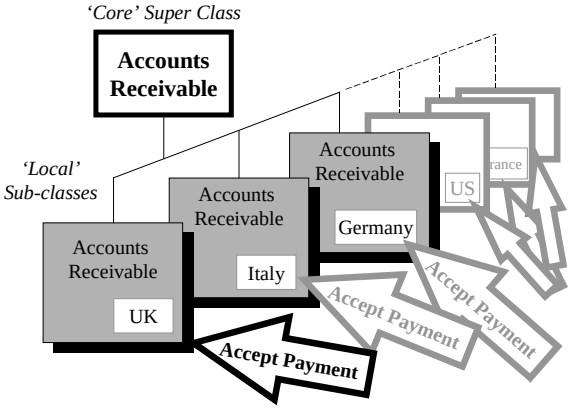

Figure 4: *The PAYMENT transaction is applied differently in each country, although the accounting module/object is a global standard*

(b) The second use would be the introduction of new functionality across the organisation such as new or updated global standards or new operational software developed in one site but potentially useful elsewhere. In the *Co-op* case vignette the UK subsidiary developed a system of vendor-managed-inventory with a large supermarket chain who would pay for goods sold on the basis of their own point-of-sale records, without any orders or invoices involved (the white 'VMI' box in the 'UK' sub-class). Implemented in object-oriented form, this functionality could have been incorporated as an attribute in the super class of the Sales & Inventory object. In this way, this functional module would have been instantly available to all other 'local' sub-classes through *inheritance* from the super-class of the Sales & Inventory object. In the actual case, it was the Malaysia local office that used it to strong competitive advantage. This is illustrated in **Figure 5**.

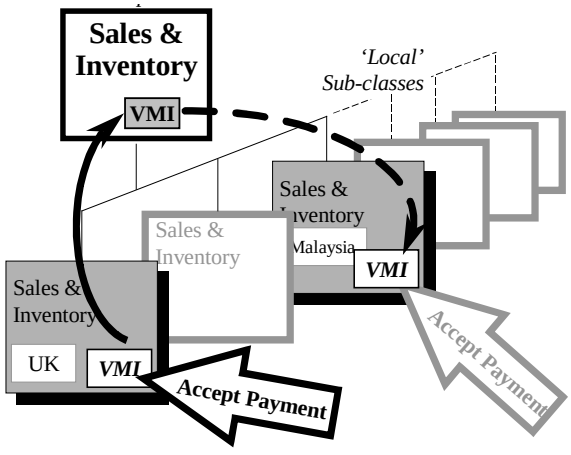

Figure 5: *Once the Vendor-Managed-Inventory (VMI) system functionality, developed locally in the UK, becomes part of the* ***Sales & Inventory*** *super-class it is instantly available to any 'local' sub-class – here demonstrated on Malaysia*

(c) A similar application of *inheritance* could be the use of an 'expert system' for the *Leasing* case vignette. Such a system would have been created to reflect the enterprise wide rules and process for establishing and evaluating a lease that would then have localised across all subsidiaries to reflect local conditions such as legislation, banking industry structure and local/regional competitive strategies. Villeneuve et al. [1997] give a good treatment of the object-orientation issues involved in such an application.

The advantages of using an object oriented approach to the design/definition of the common and local parts of an international information system are, however, not restricted to the building of the system. As Butler Cox [1991] postulate, the business style of multinational enterprises is fluid and changes with their development. Moreover, King and Sethi [1993] demonstrated that multinational enterprises are hardly ever homogenous - they work at the same time in different modes and at differing degrees of 'penetration' into the 'local' systems of different countries, e.g. applying a 'global' style in small subsidiaries and a 'transnational' style in larger, more sophisticated local environments. The ease and flexibility with which an object oriented information systems architecture can be maintained and changed would certainly seem to make such an object oriented approach an essential design consideration.

7. CONCLUSION

Previous research into the nature of international information systems has lead to the proposition of a generic architecture paradigm, consisting of a two-dimensional topology. The model configures any international information system as consisting of a 'core' of information systems common for all and 'local' systems used in subsidiary sites. This generic structure model has been confirmed by applying it to the information systems of a set of four different multinational enterprises. It could be shown to be a practical and flexible tool to describe and understand the nature of international firms' information systems.

Because the architecture model prescribes a way of structuring international information systems it could also have significant implications for the method of developing them. The 'goodness' of the design for an international information system seems to hinge on how well the 'core' systems (technology and applications) is designed, as this determines to a large extent how easy it will be to apply, maintain and change the global standards of the enterprise. 'local' systems - and their interfaces - are contingent on, and complement the 'core's ' technology. Object Orientation, as a base paradigm for the design of 'core' elements, was found useful. In particular, the principles of *encapsulation*, *polymorphism* and *inheritance* ensure that the 'core' systems can be implemented in differing degrees of 'penetration'. Furthermore, an object-oriented design is supposed to foster ease of maintenance, enhancements and other changes as future business needs and the evolution of the international firm itself dictate. As a result of this first, notional and cautious exploration of the use of object-orientation it can be conjectured that Object Orientation should be the preferred approach and method of analysis, design and development for international information systems. So far, however, neither the literature nor the author's own experience and research have witnessed international information systems projects which use Object Orientation as the main design principle

To be of practical use, however, this architecture model now needs to be validated on a much larger, diversified and altogether more representative scale. More empirical research aimed at analysing the structure and architecture of international information systems, for which a grounded theory approach seems appropriate, is needed.

REFERENCES

Booch. G., [1986] Object-Oriented Development, *IEEE Transactions on Software Engineering* 12, No. 2, 211-221.

Broadbent, M. and Weill, P., [1997]. Management by maxim: how business and IT managers can create IT infrastructures. Sloan Management Review; 38 [3], pp. 77¯92.

Burn, J. M., Cheung, H. K., [1996], Information Systems resource Structure and Management in Multinational Organisations, in Palvia, P. C., Palvia, S. C., Roche, E. M., [Editors], *Global Information Technology and Systems Management - Key issues and Trends*, Ivy League Publishing, Ltd, p293-324.

Butler Cox plc. [1991]. Globalisation: The Information Technology Challenge. *Amdahl Executive Institute Research Report*. London.

Cockburn, A. A. R., [1993]. The impact of object-orientation on application development. *IBM Systems Journal*, Vol. 38, Nos. 2 & 3, June 1999 (reprint), p308-332

Collins, R. W., Kirsch, L., [1999]. Crossing Boundaries: The Deployment of Global IT Solutions. Practice-Driven Research in IT Management Series™, Cincinnati, OH.

CSC Research and Advisory Services, [1995], *Globalisation and Localisation: Implications for IS.* CSC Foundation Final Report, London.

DeLoach, S. A., Hartrum, T. C. 2000. A theory-based representation for object oriented domain models. *IEEE Transactions on Software Engineering*, Vol.26, No. 6, p500-517

Earl, M. J. [1989*]. Management Strategies for Information Technology*. Prentice-Hall, London.

Embley, D., Kurtz, B., Woodfield, S., [1992] *Object-Oriented Systems Analysis and Specification: A Model-Driven Approach*, Prentice-Hall, Inc., Englewood Cliffs, NJ.

Fichman, R.G., Kemerer, C. F., [1992]. Object-oriented and conventional analysis and design methodologies: comparison and critique. *IEEE Computer*, October, p22-39

Gallupe, R. B., Tan, F. B., [1999]. A research manifesto for global information management., Journal of Global Information Management, 7[3]: p5-18

Gibson, R. [1994]. Global information technology architectures. *Journal of Global Information Management.* 2[1]: 28-38.

Glaser, B.G., Strauss, A.L., [1967]. *The discovery of grounded theory.* Aldine Publishing Company, Hawthorne, New York.

Grover, V., Segars, A H., [1996], The relationship between organizational characteristics and information system structure: An international survey., *International Journal of Information Management*, 16[1]: p9-25

Hamelink, C.J. [1984].Transnational data flows in the information age. Lund, Sweden: Student-litteratur AB

Ives, B., Jarvenpaa, S.L. [1991]. Applications of Global Information Technology : Key Issues for Management. *MIS Quarterly, [15, [1* [March] p33-50.

Ives, B., Jarvenpaa, S.L. [1994]. MSAS Cargo International: Global Freight Management . In T. Jelass, C. Ciborra [Eds.] *Strategic Information Systems: A European Perspective.* John Wiley and Sons, New York.

Jacobson I., Christerson M., Jonsson P., Overgaard G., [1992] *Object-Oriented Software Engineering*, Addison-Wesley Publishing Co., Wokingham, England.

Karimi, J. and Konsynski, B. R. [1996]. Globalisation and information management strategies. In: Deans, C. and Jurison, J. [Ed's]: *Information Technology in a Global Business Environment.* Boyd & Fraser, London. P169-189

Keen, P. G. W., Bronsema, G. S. and Auboff, S. [1982]. Implementing Common Systems: One Organisation's Experience. *Systems, Objectives and Solutions.* 2.

King, W. R. and Sethi, V. [1993]. Developing Transnational Information Systems: A Case Study. *OMEGA International Journal of Management Science,* 21, [1, 53-59.

King, W. R., Sethi, V. [1999]. An empirical assessment of the organization of transnational information systems. *Journal of Management Information Systems* [15[4]: 7-28.

Lehmann, H.P. [1996]. 'Towards a common architecture paradigm for the global application of information systems' in Glasson, B.C., Vogel, D.R., Bots, P.W. and Nunamaker, J.F. (Editors) '*Information Systems and Technology in the International Office Of The Future'*, Chapman and Hall, London, 1996, p199-218.

Lehmann, H.P.]1997]. A Definition of Research Focus for International Information Systems. *Proceedings of the Thirtieth Annual Hawaii International Conference on Systems Science*s, Maui, Hawaii, January,.

Lehmann, H.P., [2000a] Towards a Grounded Theory of Information Systems for the International Firm: Critical Variables and Causal Networks'. *Proceedings of ECIS 2000 – The 8th European Conference on Information Systems* [CD]; Vienna, Austria

Lehmann, H.P., [2000b]. 'The Fatal Politics of Multinational Information Systems: A Case Study'. *Journal of Information Technology Cases & Applications,* Volume 2, Number 3. New York, NY., p40-64

Lehmann, H.P., [2001]. *A Grounded Theory of International Information Systems.* Unpublished Doctoral Thesis at the University of Auckland, New Zealand, p11-14.

Lorenz, M., [1992]. *Object-Oriented Software Development: A Practical Guide*, Prentice-Hall, Inc., Englewood Cliffs, NJ

Martin, J. Odell J., [1992] *Object-Oriented Analysis and Design*, Prentice-Hall, Inc., Englewood Cliffs, NJ.

Monarchi, D., Puhr, D., [1992] A Research Typology for Object-Oriented Analysis and Design, *Communications of the ACM* 35, No. 9, 3547 (September).

OMG Document 91.12.1, Rev. 1.1, [1991] *The Common Object Request Broker: Architecture and Specification*, Object Management Group, 47 Walnut, Suite 206, Boulder, CO 80301 (December).

Peppard, J., [1999]. Information management in the global enterprise: An organising framework. European Journal of Information Systems, 8[2]: 77-94

Rumbaugh, J., Blaha, M., Premerlani, W., Eddy, F., Lorensen, W., [1991]. *Object-Oriented Modeling and Design*, Prentice-Hall, Inc., Englewood Cliffs, NJ.

Sankar, C., Apte, U. & Palvia, P.[1993]. Global Information Architectures: Alternatives and Trade-offs. *International Journal of Information Management,* [1993], [13, 84-93.

Sethi, V., Olson, J.E. [1993] An integrating framework for information technology issues in a transnational environment. In *Global Issues in Information Technology*, Idea Publishers, Harrisburg

Simon, S. J., Grover, V. [1993]. Strategic use of information technology in international business: A framework for information technology application. *Journal of Global Information Management*. [1[2]: 29-42.

Snyder. A., [1993]. The Essence of Objects: Concepts and Terms, *IEEE Software* 10, No. 1, p31-43

Targowski, A. S., [1996], Global Information Infrastructure – The Birth, Vision and Architecture, *Idea group Publishing, London, UKJ,*

Tractinsky, N., Jarvenpaa, S. L., [1995]. Information systems design decisions in a global versus domestic context. MIS Quarterly, 19[4], December, p 507-534.

Van den Berg, W., Mantelaers, P., [1999, Information systems across organisational and national boundaries: an analysis of development problems. *Journal of Global Information Technology Management* , Vol. 2, No. 2

Villeneuve, A. O., Fedorowicz, J. [1997]. Understanding expertise in information systems design, or, What's all the fuss about objects? *Decision Support System*s, Vol. 21/1997, p111-131

Weill, P. [1992]. The Role and Value of information Technology Infrastructure: Some Empirical Observations. *Working Paper No. 8, University of Melbourne.* Melbourne, July.

Weill, P., [1993]. The role and value of information technology infrastructure: some empirical observations. In: Banker, R., Kauffman, R. and Mahmood, M.A. [Editors], *Strategic Information Technology Management: Perspectives on Organizational Growth and Competitive Advantage.* Idea Group Publishing, Middleton, PA.

Weill, P., Broadbent, M., [1994a]. Infrastructure goes industry specific. *MIS* July, pp. 35¯39.

Weill, P., Broadbent, M., [1998]. . *Leveraging the New Infrastructure: How Market Leaders Capitalize on Information Technology*. Harvard Business School Publishing, Boston, MA.

Weill, P., Broadbent, M., Butler, C. and Soh, C., [1995]. . Firm-wide Information Technology Infrastructure investment and services. *Proceedings of the [16th] International Conference on Information Systems,* Amsterdam.

Weill, P., Broadbent, M., St.Clair, D. [1994b]. I/T Value and the Role of I/T Infrastructure Investments. In *Strategic Alignment*, Luftman, J. [Ed], Oxford University Press.