Using Petri Nets to Model Weltanschauung Alternatives in Soft Systems Methodology

John Lamp
School of Management Information Systems
Deakin University
Geelong, Australia
E-mail: John.Lamp@deakin.edu.au

Abstract

Soft Systems Methodology is often criticised as lacking rigorous tools. In this paper, an outline of Soft Systems Methodology and Petri Nets is given. Petri Nets are then explored as a formal tool for the description and amalgamation of weltanschauung alternatives at the conceptual modelling stage.

Introduction

Wand and Weber (1990; 1993) have considered system development methodologies and their associated system modelling schemes. They have attempted to consider formally the question of assessing how well information systems analysis and design methodologies address the three major tasks that need to be undertaken during the analysis and design of information systems:

1 Representation of the real world according to the views held by individual users.

2 Faithful tracking of the real world system as it undergoes transformations from one state to another.

3 Decomposition of the real world model in ways which reflect the structure and dynamics of the represented real system.

Soft Systems Methodology is extremely effective at eliciting the complexities of a real world problem situation. One particular strength of Soft Systems Methodology is the ability to provide a number of different perspectives (weltanschauung) on a particular problem domain. A difficulty arises, when trying to develop a single conceptual model that can retain the richness that has been identified.

Soft Systems Methodology

Soft Systems Methodology approaches the analysis of information systems from a systems approach. In particular, a fundamental belief is that the whole of a system is greater than the sum of its parts – that a system can display emergent properties which are lost when a traditional reductionist approach is taken. It is also important to see the system under scrutiny as an open system and part of the wider environment in which it finds itself.

The stages of Soft Systems Methodology are shown in Figure 1. In stages one and two, the information about the problem situation is gathered and expressed, usually using a technique known as a rich picture. In stages three and four, root definitions are devised and a conceptual model is created of the system. At the fifth stage, this conceptual model is compared against the real world. As a result of this comparison some areas where improvements can be made may be identified. The sixth stage consists of identifying feasible or desirable changes and the seventh stage is to take action on the changes.

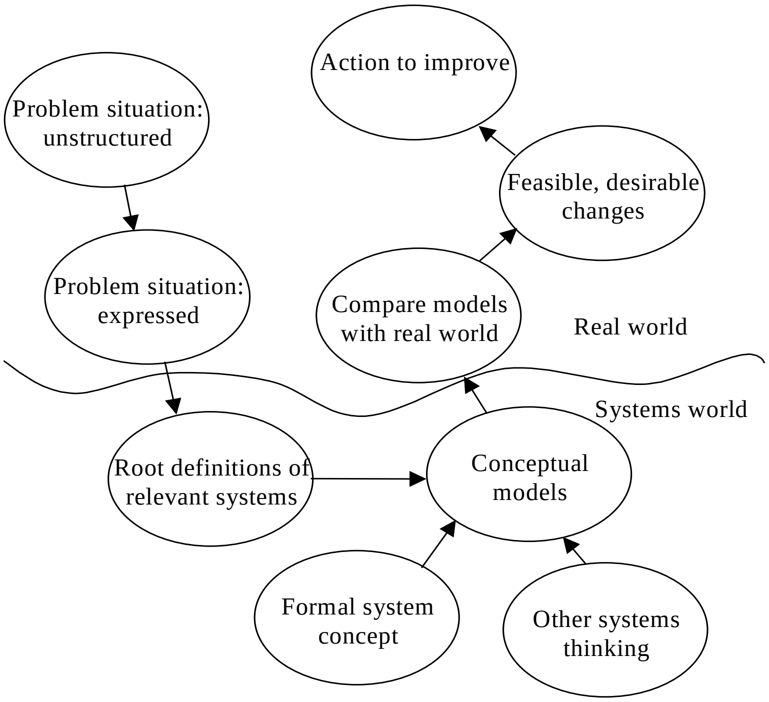
Action to improve
Problem situation: unstructured
Feasible, desirable changes
Problem situation: expressed
Compare models with real world
Real world
Systems world
Root definitions of relevant systems
Conceptual models
Formal system concept
Other systems thinking

Figure 1 – Soft Systems Methodology stages (after Checkland, 1981)

In this paper we are particularly concerned with stages three and four – the making of root definitions, and their transformation into a conceptual model.

Weltanshauung or Worldview

Human activity can be considered from a number of points of view. Your point of view dictates what you see, or would like to see – a building can be viewed from front, side, overhead, as a plan etc; you can also view a building as an architect, user, artist. From these differing perspectives on a particular problem situation, a series of root definitions could be developed, each of which being based on using a different perspective as the worldview for that definition. Root definitions should be based on viewpoints that are explicit and can be supported. An architect values geometry, structural design; an artist values aesthetics; a user values access, functionality. Given the task, each would design a different building, achieving the same purpose, but with differing emphasis.

By explicitly considering the worldview, an analyst using Soft Systems Methodology validates the transformations proposed in the context of the image or assumptions that the worldview dictates. Carefully selecting relevant viewpoints can highlight differing perspectives that influence the design. None of these root definitions is necessarily more correct, and each could give rise to a different conceptual model, with different insights.

Conceptual Models

Conceptual models illustrate the root definitions of the system under study in a manner that shows the relationship between system activities. The development of a conceptual model may expose some problems with the root definition(s) that have been developed and an iterative process of revisiting the root definitions, and developing a conceptual model is undertaken.

One commonly used formalism proposes a scheme in which conceptual models are developed at a number of different levels. The top level contains about five to ten activities, which give an overview of the main activities within the system under study. Each of these major activities can then be expanded to show greater detail. This process is shown in figure 2.

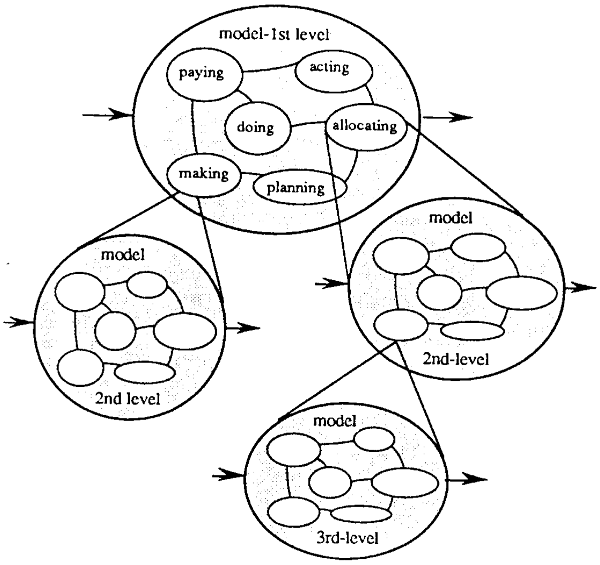

Figure 2 – A Formalism for Conceptual Models (Patching, 1990, p88)

There are a number of different schemes for developing conceptual models within Soft Systems Methodology. Brian Wilson suggests a less formally structured conceptual model (Figure 3).

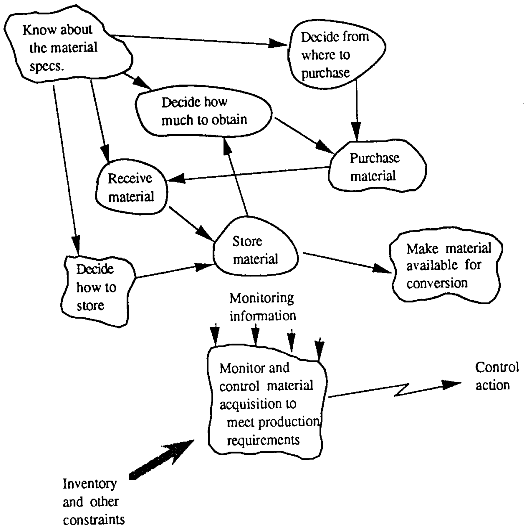
Know about the material specs.
Decide from where to purchase
Decide how much to obtain
Purchase material
Receive material
Store material
Make material available for conversion
Decide how to store
Monitoring information
Monitor and control material acquisition to meet production requirements
Control action
Inventory and other constraints

Figure 3 – Wilson's Conceptual Model (Wilson, 1984 in Patching, 1990, p87)

There have also been a number of attempts to integrate structured systems design tools at the conceptual modelling stage. Tools suggested include data flow diagrams, JSD elements, predicate logic and the extension of the Soft Systems Methodology itself to include Maltese Cross analysis, and conceptual flow models and conceptual data models. These approaches have been discussed in Mingers (1995). Most of these proposals require a single conceptual model to be devised. This eliminates much of the richness that is a hallmark of Soft Systems Methodology. In particular the representation of alternative worldviews is lost. Other suggestions involve a major revision of the Soft Systems Methodology stages to accommodate their proposals.

In this paper the use of some of the features of Petri nets is proposed as a modelling formalism which preserves the richness of the development of alternative worldviews, and requires no revision of the Soft Systems Methodology stages.

Petri Nets

The foundation of Petri nets was presented by Carl Adam Petri in his doctoral thesis in 1962. The first nets were called Condition/Event Nets (CE-nets). Following this there was a large amount of work on various developments of Petri nets, in particular, Place/Transition Nets (PT-nets) which allowed for more than one token in a place, and Predicate/Transition nets (PrT-nets).

> A Petri Net is a directed graph with two types of nodes ... . The place nodes, or *places*, are usually represented by circles and the transition nodes, or *transitions*, by short lines or bars. Directed *arcs* may connect the different types of nodes, but not nodes of the same type. An arc from a place to a transition is called an *input arc* and an arc from a transition to a place is called an *output arc*. A place connected to a transition by an input arc is called an *input place*, and a place connected to a transition by an output arc is called an *output place*. The places may be occupied by any natural number of markers called *tokens*, shown as dots. … When all the input places to a transition contain at least one token, the transition is enabled and may *fire*. When the transition *fires*, one token is removed from each of its input places and one token is added to each of its output places. The collection of tokens in a net is called the *marking* of the net. (Billington, 1982, 97)

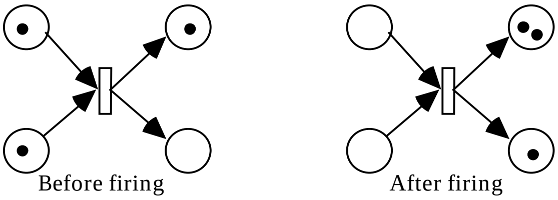

Figure 4 – The Firing of a Transition

Coloured Petri Nets (CP-nets) were proposed by Jensen (1981) as a more elegant way of dealing with some technical problems of PrT-nets. The following description of coloured Petri nets is largely drawn from Jensen (1991). Coloured Petri nets differ from other Petri nets in that

> each place may contain several tokens and each of these contains a data value - which may be of arbitrarily complex type (eg a record where the first field is a real, the second a text string, while the third is a list of integer pairs). The data value which is attached to a given token is referred to as the **token colour**. (Jensen, 1991, p47)

Thus, CP-nets allow for the tokens to be assigned data types and carry values through the net, and thus allow the construction of nets which can deal with data flow and manipulation.

Transitions may have guards – boolean expressions that must be fulfilled before a transition can occur. The expression may contain variables, constants, functions and operations that are defined in the declarations.

Object Petri Nets (OPNs) extend the formalism of Coloured Petri Nets with a complete integration of object-oriented features, including inheritance, polymorphism and dynamic binding. The object-orientation provides powerful structuring primitives allowing the modelling of complex systems, including those with multiple levels of activity. An object Petri net language and modelling scheme, LOOPN++ was presented in Keen and Lakos (1994). This language fully integrates the concepts of object-oriented language design and modular design into a coloured Petri net specification, and is based on several years' experience by the authors with previous modular Petri net schemes. LOOPN++ specifications are amenable to formal analysis because they are founded on the semantics of coloured Petri nets (Lakos 1994).

One of the features of working with Petri nets is the ability to form sub-nets by encapsulating parts of the net to emphasise a functional aspect of the net. Sub-nets should be formed with transitions on their boundary, in which case they are represented as a large transition, or with places on their boundary, in which case they are represented as a large place This allows for clarity without sacrificing the structure of the net. Figure 5 shows part of the Z39.50 search and retrieval protocol represented as a sub-net to illustrate the message passing through the protocol layer between an application and the network.

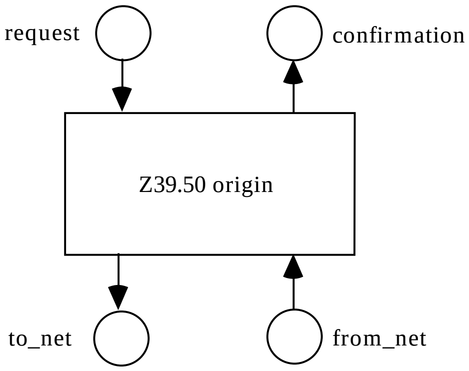

Figure 5 – Z39.50 Origin Ideal Structure (Lamp, 1994)

It is this sub-net feature of Petri nets which holds promise for conceptual modelling in Soft Systems Methodology.

Modelling Weltanschauung Alternatives

Developing root definitions is done using the CATWOE checklist.

C *Clients* or customers of the system, ie those who benefit from, or are affected by, the outputs from the system

A *Actors* who carry out activities within the system

T *Transformation* ie the change that takes place within or because of the system (ie the conversion of input to output)

W *Weltanschauung* or *Worldview* ie how the system is perceived from a particular (explicit) viewpoint - sometimes described as 'assumptions made about the system'

O *Owner* of the system, ie to whom the system is answerable, and or who could cause it to cease to exist

E *Environment* ie the world that surrounds and influences the system, but has no control over it

So, for example, a possible root definition for a hire car firm from the point of view of customers might be constructed in this way:

C the casual and regular customers of the car hire firm

A the employees of the car hire firm

T satisfying customer needs for temporary use of a car

W a car hire firm exists to provide cars to hirers

O the manager of the agency

E the legislative and regulatory framework concerning registration and licensing of motor vehicles, and making them available for hire

The following root definition could be constructed from that checklist:

A system owned by the manager of the care hire firm, operated by the firm's employees that satisfies the needs of customers for the temporary use of a motor vehicle, constrained by the legislative and regulatory framework concerning registration and licensing of motor vehicles, and making them available for hire.

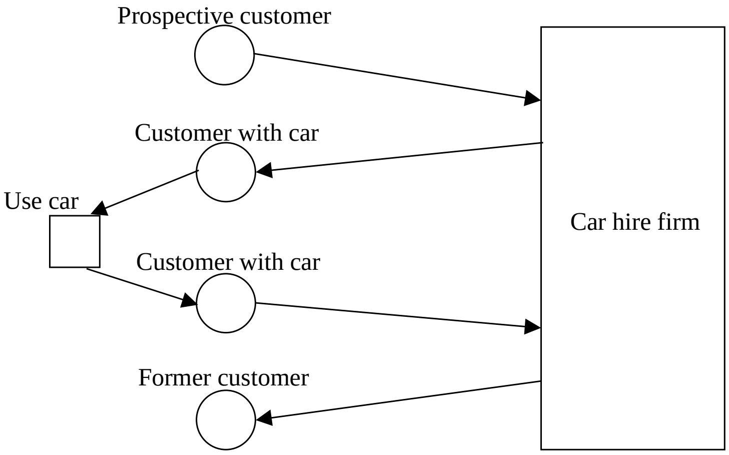

Figure 6 – The customer's worldview

A Petri net illustrating the customer's interactions with the car hire firm, reflecting this worldview is shown in Figure 6.

Another possible root definition for a hire car firm is to take the point of view of the owners of the car hire firm:

C the casual and regular customers of the car hire firm

A the employees of the car hire firm

T satisfying customer needs for temporary use of a car

W a car hire firm exists to issue hire contracts in exchange for money

O the manager of the agency

E the legislative and regulatory framework concerning registration and licensing of motor vehicles, and making them available for hire

The following root definition could be constructed from that checklist:

A system owned by the manager of the care hire firm, operated by the firm's employees that generates income for the car hire firm by temporarily hiring out motor vehicles, constrained by the legislative and regulatory framework concerning registration and licensing of motor vehicles, and making them available for hire.

While this is only a minor change of perspective for the development of the root definition, the Petri net shown in Figure 7 is considerably different.

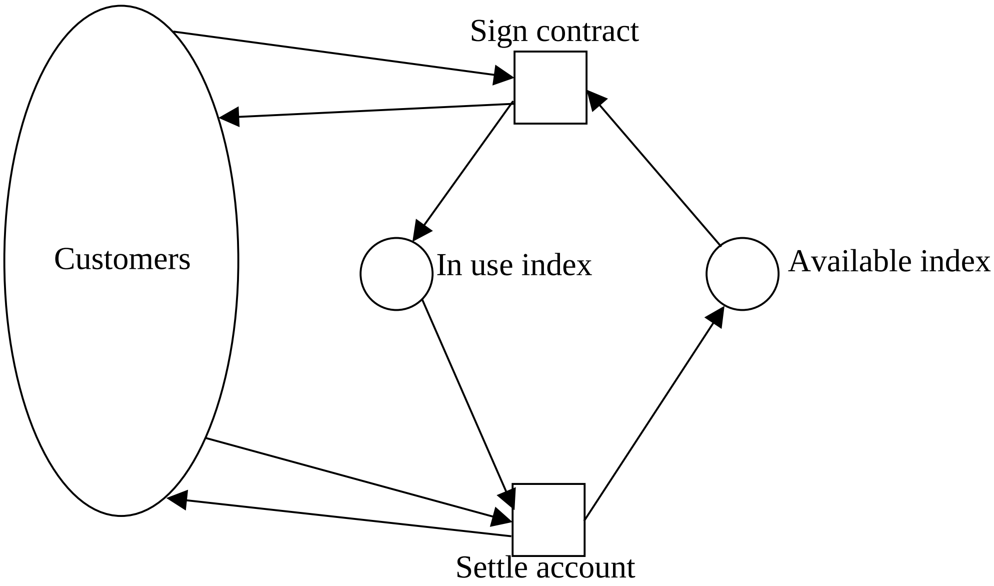

Figure 7 – The car hire firm's worldview

These two worldviews produced similar root definitions, but very different Petri nets. A third possible worldview would be that of an insurance company providing cover for the vehicles used by the car hire firm, based on whether they were in use, or in secure storage.

C the care hire firm

A the employees of the insurance company

T providing appropriate levels of insurance cover to the vehicles and customers of the car hire firm

W indemnifying the owners and customers of the car hire firm against specified risks

O	the underwriters of the insurance company

E	the legislative and regulatory framework concerning motor vehicle and personal injury insurance

The following root definition could be constructed from that checklist:

A system owned by the underwriters of the insurers, operated by the insurer's employees that provides insurance cover for the motor vehicles and customers of the car hire firm against specified risks, constrained by the legislative and regulatory framework concerning motor vehicle and personal injury insurance.

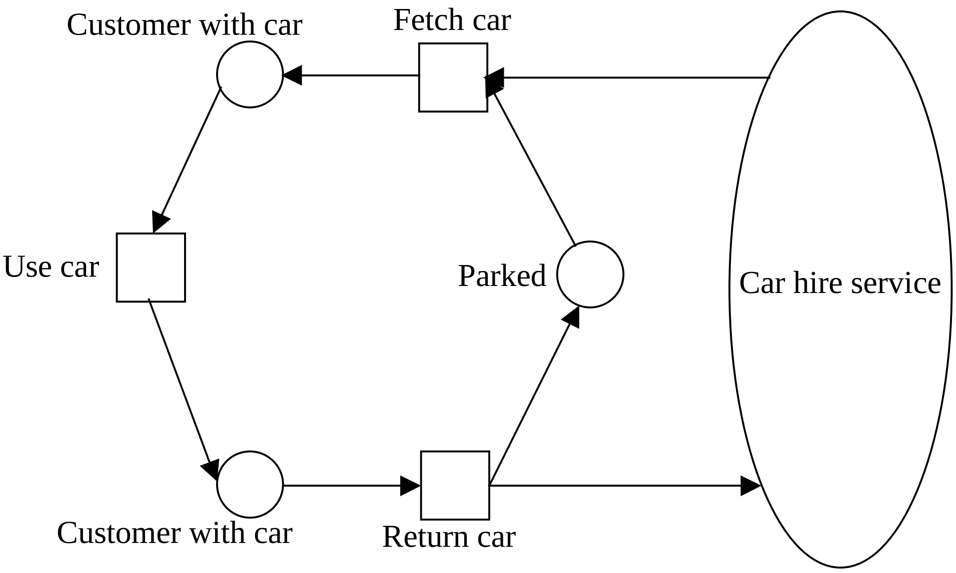

Figure 8 – The insurance company's worldview

This process would be extended until all possible worldviews had been taken into account. The next stage of Soft Systems Methodology is to attempt to develop a conceptual model that would be able to reconcile these differing points of view. By removing the sub-netting from the individual Petri nets for the different worldviews, a comprehensive net can be developed which reconciles the different worldviews.

Such a Petri net is shown in Figure 9. This net successfully accommodates all three worldviews given in the example. Line A divides the net at the customer worldview; line B divides the net at the insurance company worldview (*Prospective customer* and *Former customer* are included in the sub-net); and line C divides the net at the car hire company worldview.

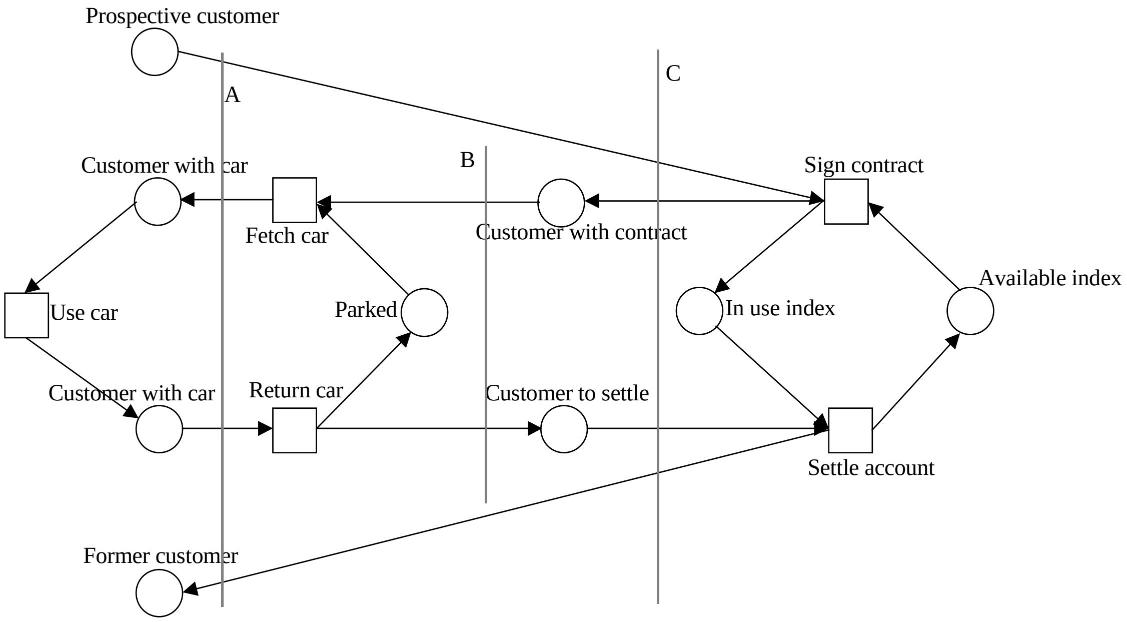

Figure 9 – The Petri net without sub-netting

Conclusions

Soft Systems Methodology allows an analyst to take into account a number of differing worldviews in an attempt to elicit as much information from a complex situation as possible. Using the sub-netting features of Petri nets allows the development of a conceptual model that can take into account differing worldviews without losing their richness.

The use of Petri nets gives access to a modelling technique which is based in mathematical formalisms and can be analysed using Petri net tools.

The example given is a relatively simple one which would probably not be a candidate for analysis using Soft Systems Methodology in the normal run of things, but was given here in the interests of clearly explaining how the sub-netting process could be used. Further work, looking at more complex human activity systems, which would normally be analysed using Soft Systems Methodology, would be needed to verify this approach as feasible.

References

Billington, J. 1982. ‘Specification of the transport service using numerical Petri nets.’ in Sunshine, C (ed) *Protocol specification, testing and verification*. North Holland Publishing Company. The Netherlands. pp 77-100

Checkland, P. 1981. *Systems Thinking, Systems Practice,* Wiley, Chichester

Jensen, K. 1981. ‘Coloured Petri nets and the invariant method’. *Theoretical Computer Science*. No 14. pp 317-336

Jensen, K. 1990. ‘Coloured Petri Nets: A High Level Language for System Design and Analysis’. In Advances in Petri Nets 1990, G. Rozenberg, Ed. *Lecture Notes in Computer Science,* p 483. Springer-Verlag, Berlin.

Jensen, K. 1991. ‘Coloured Petri nets: A high level language for system design and analysis.’ in Jensen, K & Rozenberg, G (eds) *High-level Petri Nets*. Springer-Verlag. Berlin. pp 44-122

Jensen, K.; S. Christensen; P. Huber; and M. Holla. 1992. *Design/CPN™: A Reference Manual* . MetaSoftware Corporation.

Keen, C.D. and C.A. Lakos. 1994. "Information Systems Modelling using LOOPN++, an Object Petri Net Scheme" In *4th International Working Conference on Dynamic Modelling and Information Systems*. Noordwijkerhout, The Netherlands. pp 31-52.

Lakos, C.A. 1994. "Object Petri Nets – Definition and Relationship to Coloured Nets". *Technical report TR94-3*. Department of Computer Science, University of Tasmania.

Lakos, C.A. and C.D. Keen. 1994. "LOOPN++: A New Language for Object-Oriented Petri Nets".*Technical report 94-4*, Department of Computer Science, University of Tasmania.

Lamp, J., 1994, *Encoding the ANSI Z39.50 Search and Retrieval Protocol using LOOPN*. Department of Computer Science, University of Tasmania,. [Online] Available URL: http://www.man.deakin.edu.au/jw_lamp/BCthesis.pdf [Accessed 26 Jun 98].

Mingers J., 1995 “Using Soft Systems Methodology in the Design of Information Systems” in Stowell F (ed) *Information Systems Provision: the Contribution of Soft Systems Methodology,* Maidenhead McGraw-Hill pp 18-50

Patching, D., 1990 *Practical Soft Systems Analysis* London: Pitman

Reisig, W. 1985. "Petri nets: An Introduction". In *EATCS Monographs on Theoretical Computer Science*, 4. Springer-Verlag, Berlin.

Reisig, W. 1986. "Petri Nets in Software Engineering". In Proceedings of Advances in Petri Nets. *Lecture Notes in Computer Science*, 254. Springer-Verlag, Berlin.

Reisig, W. 1992. *A Primer in Petri Net Design*. Springer-Verlag, Berlin.

Wand, Y. and R. Weber. 1990. "An ontological model of an information system". *IEEE Transactions on Software Engineering*, 16: 1282.

Wand, Y. and R. Weber. 1993. "On the ontological expressiveness of information systems analysis and design grammars". *Journal of Information Systems* 3(4): 217.

Acknowledgements

Dr Charles Lakos initiated and has maintained my interest in Petri nets. The ideas for this paper came during a seminar he gave at the University of Tasmania *On the Abstraction of Coloured Petri Nets*. The car hire firm example was used in that seminar. Ms Cathy Urquhart introduced me to Soft Systems Methodology, but only I was perverse enough to think of combining them.