

A review of information flow diagrammatic models for product–service systems

C. Durugbo, A. Tiwari, J.R. Alcock

Abstract: A product–service system (PSS) is a combination of products and services to create value for both customers and manufacturers. Modelling a PSS based on function orientation offers a useful way to distinguish system inputs and outputs with regards to how data are consumed and information is used, i.e. information flow. This article presents a review of diagrammatic information flow tools, which are designed to describe a system through its functions. The origin, concept and applications of these tools are investigated, followed by an analysis of information flow modelling with regards to key PSS properties. A case study of selection laser melting technology implemented as PSS will then be used to show the application of information flow modelling for PSS design. A discussion based on the usefulness of the tools in modelling the key elements of PSS and possible future research directions are also presented.

1. Introduction

A product–service system (PSS) [1] is a concept that offers value proposition based on the delivery of integrated products and services [2–4]. It involves considering factors relating to the life cycle of products, closely linking products and services, and establishing links (with customers and other manufacturers) to aid product/service delivery. Once these factors have been considered, the next step should involve the modelling of business requirements for the development of an effective and efficient PSS [5]. These business requirements define how companies can meet customer demands for complex solutions that combine products and services. Function-oriented modelling [6, 7], object-oriented modelling [8], conceptual modelling [5] and service modelling [9] are some approaches to modelling the business requirements for a PSS.

Function orientation is an approach to design which decomposes a system into a set of interacting functions [10]. Decomposition or ‘top-down’ analysis is a term which describes the process of breaking down a system into smaller manageable parts. Modelling this functional decomposition is an activity which could aid in identifying products and services within a PSS [5]. This process could be applied to analyse existing systems, to redesign systems or to develop new systems [11].

The function-oriented approach is classified into three main categories or species: mathematical axioms, tree structure and function input and output [12]. Mathematical axioms are employed for complete description of the system in conformance with precise rules while tree structures

identify functions without the involvement of data used. Function input and output distinguishes system inputs and outputs with regards to how data are consumed and information is used. This article is concerned with the third category—function input and output, especially information flows in systems. Information flow is used here and throughout this article, as an approach to illustrating the architecture of a system or organisation describing inputs and outputs. This architecture influences the system's efficiency, adaptability and the reusability of components [13].

According to van Gigch [14], 'a system is an assembly or set of related elements'. A PSS therefore includes products and services as well as other related elements such as information and communication technologies [2] and infrastructure [15] that aid delivery processes of a PSS. Consequently, modelling information flow for a PSS is an important measure for assessing the level of redundancy and inefficiency in PSS delivery processes [6]. This is because an information flow model can be used to assess possible actors, roles and scenarios for the delivery of integrated products and services [16]. Recommendations based on these assessments can then be applied for improving quality, efficiency and financial performance of a PSS in accordance with ISO 9000 and ISO 14000 [6].

In addition, ISO TR 9007 maintains that models of information can provide a common basis for different working groups to represent, understand and manipulate the behaviour a set of entities [17]. Within the context of a PSS, an information flow model can therefore be used as a common basis for PSS designers and operators to describe PSS processes.

The aim of this article is to review existing, diagrammatic, function-oriented information flow modelling tools. The paper seeks to contribute to knowledge by analysing these modelling tools in terms of their usefulness as modelling tools for PSS design. A case study of selective laser melting technology implemented as a PSS will also be applied to demonstrate the use of information flow modelling for PSS. The paper concludes by discussing possible use of the reviewed modelling tools for PSS design and by making recommendation for future research.

2. Diagrammatic information flow tools

Hungerford et al. [18] have asserted that diagrams or diagrammatic reasoning are better suited to solving problems created by increasing complexity in systems when compared with text-based (sentential) representations. They highlight three main reasons for this assertion.

Firstly, diagrams promote information clusters (grouping of information), thus eliminating the need to conduct large amounts of searches associated with problem-solving inferences.

Secondly, diagrams promote information clusters based on a single element, hence eliminating the need to match symbolic labels. Thirdly, diagrams offer facilities that support a wide range of perceptual inferences, which are simple and easy to use.

Becker et al. [5] have suggested that standard models (as-is models) should be identified and serve as a starting point for models of planned systems (to-be-models). This paper presents, as a

first step towards this approach, an analysis of some key diagrammatic information flow models. Diagrammatic tools beyond the scope of information flow, for example models for timeline orientation (UML sequence diagrams), process orientation (flow chart diagrams) or state orientation (state transition diagrams), are omitted.

The information flow diagrammatic models identified from literature are tabulated in Table 1. They include data flow diagrams, Integrated DEFinition method of modelling functionality and information modelling (IDEFØ and IDEF1), Graphes à Résultats et Activités Interreliés (GRAI) grids and nets, Petri nets, Input-Process-Output diagrams and design structure matrices. Each modelling approach has its strengths and weaknesses, which must also be taken into consideration by PSS designers and operators.

Table 1 A list of Function-Oriented Information flow diagrammatic models found in literature

Modelling Tool	Description	Literature
Data flow diagrams	Analyses information flow within and between organisations or systems; applied for the design and deployment of information systems	[12, 18–26]
Integrated DEFinition method of modelling functionality and information modelling(IDEFØ and IDEF1)	Illustrates information flow along with constraints and mechanism which affect system functions; developed from the Structural Analysis and Design Technique (SADT) approach	[11, 34, 27–36]
Graphes à Résultats et Activités Interreliés (GRAI) grids and nets	Supports information flow in decision communication, feedback and review; part of the GRAI methodology	[25, 34, 37–43]
Petri nets	Represents automated and event-driven information flow in systems;	[44–51]
Input-process-output (IPO) diagrams	Describes and documents the organisation and logic of information flow; integral to the Hierarchy plus Input-Process-Output (HIPO) approach	[12, 52–56]
Design structure matrix	Depicts dependency, independency, interdependency and conditionality of information flow for systems and organisations	[57–64]

2.1. Data flow diagrams

Data flow diagrams (DFDs) are very popular diagrammatic models [18, 19] used in describing information exchanges in a variety of organisations [20]. They were developed by DeMarco [21] in the late 1970s as a tool for analysing sequential information flows [10, 22]. DeMarco defined DFDs as ‘network representations’ of automated, semi-automated or manual systems. DFDs describe how information flows logically or physically in a system. The logical view describes how information flow is expected to happen, while the physical view refers to what actually happens. In some cases, both the physical and logical views may be the same.

Although a wide range of symbols are used in DFD, most authors use a notation for DFDs which involves four key features: processes, external entities, data stores and data flows [21, 24, 25] as shown in Fig. 1a.

Data Flow Diagram (DFD)

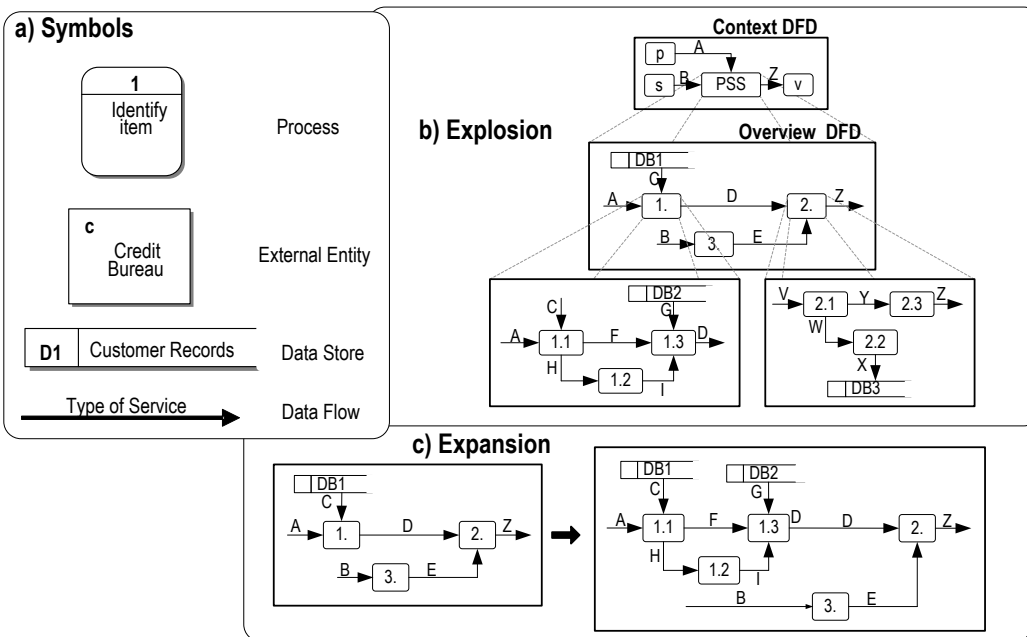


Fig 1 Approaches for Data Flow Diagrams (DFDs): (a) DFD representations; (b) Explosion approach to DFD development; and (c) Expansion approach to DFD development

With regards to design approaches, Du et al. [22] identified two main schemes for designing DFD (see Fig. 1b, c). The expansion approach described by Gane and Sarson [26] is the first scheme. It applies a single DFD, which is iteratively expanded till the entire system has been comprehensively modelled. In the other scheme, the expansion approach as explained in [21], a single diagram is created initially. This diagram is known as the context DFD. The system within this context DFD is then exploded to give the overview DFD. After these first two steps, multiple DFDs are constructed, with each successive model derived as an explosion from a single activity step in a parent or preceding diagram. This process is continued till the entire system has been comprehensively modelled. A slight variation of the explosion approach is employed in the Structural Analysis and Design Technique developed by Softech, Inc [21].

2.2. IDEFØ and IDEF1

The IDEF technique is an approach to modelling and analysing systems and enterprises. It is made up of a suite of models which contain a hierarchy of diagrams, text and glossary [27].

These models include IDEFØ, IDEF1, IDEF1X, IDEF3, IDEF4 and IDEF5.

The IDEFØ or the Integrated DEFinition method of modelling functionality is a widely used technique employed by organisations, industries and governments to support their enterprises and applications [28, 29]. Sullivan [30] asserted that the IDEFØ approach was borne out of the need for structured techniques which can be applied in systems, such as manufacturing systems, involving information flow.

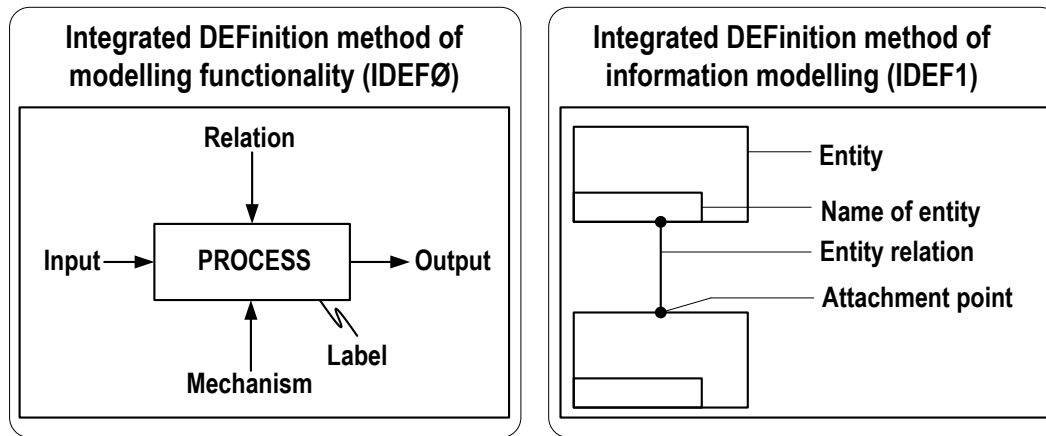


Fig 2 Representations for Integrated DEFinition method of modelling functionality and information modelling (IDEF0 and IDEF1)

The foundation for the IDEF0 modelling technique lies in the Structural Analysis and Design Technique (SADT) developed by Douglas T. Ross at SofTech, Inc in the early 1970s [28, 30–33]. SADT is a function-oriented approach which adopts an all-inclusive modelling framework, unlike data flow diagrams, which concentrate on information flow in an organisation [33]. In 1978, the U.S. Air Force adopted the SADT as its modelling technique to support its Integrated Computer Aided Manufacturing (ICAM) programme [29, 32]. It is this approach, later revised by SofTech, which now exists as the IDEF0 modelling approach. Consequently, IDEF0 can be used for all kinds of function-oriented modelling for system-based applications, such as operation, activity, process or behavioural modelling needed by a system such as a PSS. IDEF0 models contain two main diagrammatic modelling components: boxes and arrows [27, 29, 33] as shown in Fig. 2. The idea in an IDEF0 model is to begin with a single top-level diagram (tagged as A0) which provides a complete but abstract depiction of the system [28]. This top-level diagram is then decomposed into a series of child diagrams, applying the explosion approach (see Fig. 1b). Ho et al. [34] suggested that decomposition in IDEF0 modelling should continue until a complete description of the system has been attained. This process, they contend, removes ambiguity and aids its use and implementation. Based on the definition of information flow for this review, the IDEF0 approach can be used to model information flow. In other words, the IDEF0 approach illustrates the inputs and outputs of a system. For information modelling to complement the IDEF0 approach, the IDEF1 (Integrated DEFinition method of modelling Information) is recommended [35]. IDEF1 offers basic primitives for describing information that must be managed for an organisation to fulfil its objectives [36]. It identifies how functions described in IDEF0 can share data/information. It also offers three main modelling primitives: boxes that depict system functions, arrows that indicate data, information and object interface, and attachment points between arrows and points that represent types of interface (input, output, control or mechanism) described in the IDEF0 model.

2.3. GRAI grids and nets

The GRAI (Graphes à Résultats et Activités Interreliés) Engineering method was developed by GRAI Laboratory at the University of Bordeaux in the 1970s [37, 38]. Figure 3 shows the GRAI Modelling Technique which is based on a hierarchical conceptual model (the GRAI model) for supporting decision-making processes during manufacturing and establishing information flow for facilitating these decisions [37–40].

In [38], the GRAI model is divided into two parts: a macrostructure which displays the architecture of the overall system arranged in a hierarchy and a microstructure for system components which are identified in the macrostructure. The macrostructure of the GRAI model (see Fig. 3a) decomposes the system to be designed into three sub-systems [39, 41, 42].

The technological system presents the means for delivering products and services such as people, machines and materials. It is also concerned with information flows associated with these tangible/intangible offerings for meeting customer expectations. The decision system details the locus of decision in the hierarchy. This hierarchy is arranged according to decision-making levels which contain blocks known as decision centres. The information system links the decision and physical system and the enterprise environment. It also transforms and memorises information. The microstructure is concerned with decision centres in terms of their intelligence based on recognising the need for a decision to be made, their modelling capabilities of derived or gathered information, and their choice for selecting appropriate solutions based on criteria, constraints and context.

Two main diagrammatic tools are used in the GRAI model: GRAI grids and GRAI nets [34, 38]. The latter is designed to pinpoint discrepancies at the macrostructure, while the former reveals inconsistencies at the microstructure. Ho et al. [34] contended that the intention of these tools is for system designers or decision makers to review iteratively the GRAI model until discrepancies and inconsistencies are resolved or within acceptable limits of defined goals and objectives. Both tools are designed to model activities in systems. Doumeingts [38] defined an activity in a GRAI model as an operation which changes an initial state into a final state. Several GRAI grids can be developed based on requirements for realising goals and objectives or according to the complexity of the system [38, 43]. These grids are also characterised by cells for decision centres and relationships between these decision centres [43].

Relationships are used in GRAI models to specifically describe information flow and decision flow for co-ordination and synchronisation of activities in an organisation or system [41].

Relationships are depicted diagrammatically as arrowed lines (see Fig. 3b). Decision flow between two decision centres can be represented as large, emboldened lines, while information flow can be depicted as small dashed lines.

Graphes à Résultats et Activités Interreliés (GRAI)

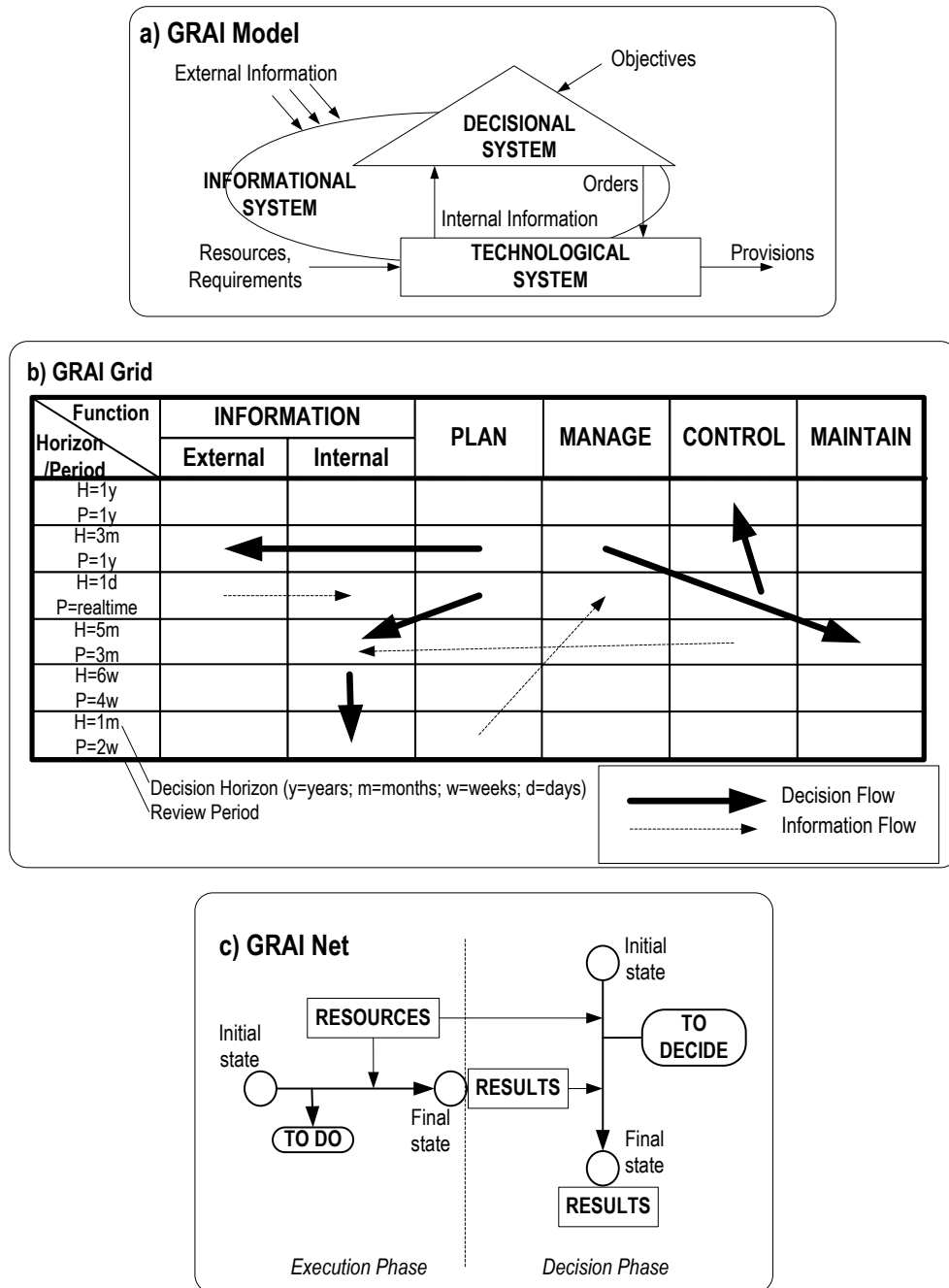


Fig 3 Graphes à Résultats et Activités Interreliés (GRAI) Modelling Technique

GRAI nets are developed after GRAI grids and describe the activities in a decision centre [40]. They are done to complement GRAI grids which give high-level diagrammatic representations of decisions without providing information about how decisions are made [41]. GRAI nets, as shown in Fig. 3c, are made up of three constructs: states, activity and supports. States are

represented by circles or ovals. Activities are represented as directed arrows, while the supports (information and technological resources) are represented as rectangles.

GRAI grids show information flow but do not represent or model them. GRAI nets on the other hand can be used to model this information flow. GRAI nets represent information flow by means of events or sequences of events in the manufacturing process or system. GRAI nets also depict states and state changes. Although originally designed for the development of production management systems, GRAI models can be used where a system is required among different groups or processes [20] like a PSS.

2.4. Petri nets

Petri nets (PNs) or place-transition nets were first proposed by Carl Adam Petri in 1962 for modelling processes in an event-driven system [44, 45]. These systems exhibit a wide range of characteristics such as non-determinism, concurrency, synchronicity as well as distributed and/or parallel features. PNs can also be used for representing the information flow in development and simulation of automated manufacturing systems [46, 47]. Murata [48] described PNs as useful mathematical and diagrammatical tools for representing control flow in systems.

Diagrammatically, PNs can be used to methodically describe and communicate ideas among designers and implementers. A PN is depicted as a directed, weighted, bipartite graph made up of four main symbols as shown by Fig. 4. Black dots represent tokens. Tokens may be resources, counters, metrics or attributes. Circles show places and are marked with a non-negative integer k of token. Bars depict transitions, while arcs connect places to transitions. In the modelling of PNs, transitions represent events in a system, while places illustrate conditions for occurrence. The tokens provide the premise for the conditions just as input and output places offer pre- and post-conditions for the event respectively.

Mathematically, PNs are presented as tuples [44, 45]. A tuple is a fixed, ordered list of elements or objects. Tuples may contain multiple occurrences of elements and objects. A Petri net is defined as a quad-tuple (P , T , I and O) where:

- P is a set of places i.e. $P = \{p_1, p_2, p_3, \dots, p_n\}$;
- T is a set of transitions i.e. $T = \{t_1, t_2, t_3, \dots, t_n\}$, with $P \cap T \neq \emptyset$ and $P \cup T \neq \emptyset$ (\emptyset refers to a tuple with no elements or objects);
- I is an *input function* specifying *Arcs* directed from places to transitions i.e.
 $I : (P \times T) \rightarrow N$ (where N is a tuple of non-negative integers); and
- O is an *output function* specifying *Arcs* directed from transitions to places i.e.
 $O : (P \times T) \rightarrow N$.

By applying these definitions, state and algebraic equations can be derived to define the behaviour and mathematical models which govern the behaviour of systems.

Lien [49] described two main principles applied in Petri-net theory: transition enabling and transition firing. A transition is enabled or fireable if its input places all hold at least one token. A transition can be fired by two processes. First, one token is removed from each input place and

secondly, the addition of a token to an output place. These symbols and configurations used in PNs can assist designers in describing some important system characteristics. These and other related principles, theories and formulae are extensively covered in literature [45, 48, 50]. Wakefield and Sears [51] identified six possible constructs during the development of PNs. These constructs are depicted in Fig 4 and can be described in terms of information flow as follows. *Sequential* execution imposes precedence in the flow of information; *concurrency* shows parallel information flow; *synchronisation* coordinates information; *merging* combines information required to carry out a function; *conflict*, in which multiple functions request access to transactions are enabled but firing is disabled; and *confusion* which allows conflict and concurrency to coexist. The two latter issues can be remedied by assigning priorities or associating probabilities to appropriate transitions [48].

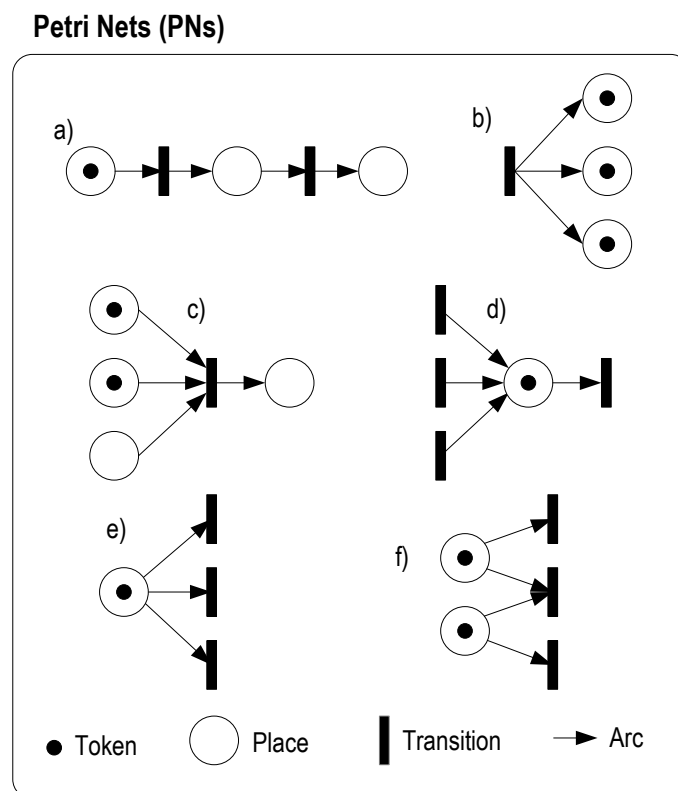


Fig 4 Petri net representations and constructs: a) Sequential execution; b) Concurrency; c) Synchronisation; d) Merging; e) Conflict; and f) Confusion

2.5. Input-process-output diagrams

The HIPO (Hierarchy plus Input-Process-Output) technique was developed by IBM's System Development Division (SDD) in the late 1970s [52, 53]. It offers diagrammatic and textual representations for the documentation of systems, programs and processes. The HIPO technique is made up of two main components [52-56]: Visual Table of Contents (VTOC) and Input-Process-Output (IPO) diagrams (See Fig 5).

The VTOC is represented as a chart showing how functions of a system or modules of a program are decomposed in a tree format. It offers a top-down analysis of a program, system or process and is made up of three main parts as shown in Fig 5b. The hierarchical diagram contains an echelon of numbered and named boxes which correspond to IPO diagrams and is read from left to right. A legend and an optional description for each function may also be included in the VTOC.

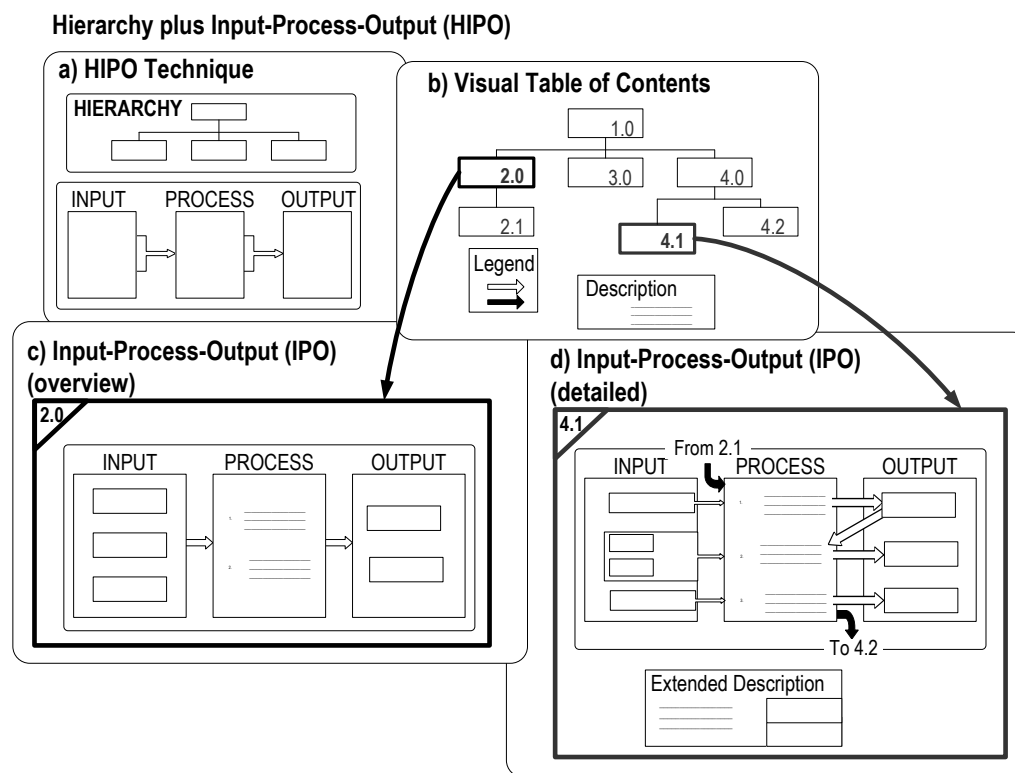


Fig 5 The Hierarchy plus Input-Process-Output (HIPO) technique and Input-Process-Output (IPO) diagrams

IPO diagrams are developed after the VTOC has been constructed. They describe functions (or modules) in the VTOC in terms of their inputs and outputs by means of processes which may be enclosed or encapsulated in the system. IPO diagrams are presented as pages in a form of pseudo-code showing local or functional information flow [55]. A page is developed for each function (or module). Each page (IPO diagram) contains three main blocks labelled as input, process and output as shown in Fig 5d. The idea is to show what is used (input) by the module; processing performed (translations and transactions) by the module represented as a high level textual representation pseudo code; and fields changed or written to (output) by the module [54-56].

Stay [52] traces the origin of the HIPO approach to structured design which offers methods for transforming a description into a functional, modular program structure. He identifies two important concepts of structured design exploited by the HIPO technique: module strength (relationships within a module) and module coupling (relationship between modules). Originally

designed for the documentation of programs [52, 56], its use can also be extended for other system related activities. The activities can include planning, development and implementation where the HIPO technique can offer information about the functions or ‘what a system does’ [53].

Martin and McClure [12] also suggested its use for both the analysis and the design of systems. They recommend its use during analysis to aid definition of various system components as a means of kick-starting the design process. For design, they highlight its use as an enabler for describing procedures of system components.

2.6. Design structure matrix

The Design Structure Matrix (also known as problem solving matrix, dependency structure matrix and design precedence matrix) is a compact, visual, generic matrix-based framework for the graphical and numerical analysis of decomposition and integration in systems [57-59]. Syed and Berman [60] traced the history of the Design Structure Matrix (DSM) approach to earlier concepts such as matrix mathematics, network precedence diagrams, network relationship diagrams and Interface-to-interface (N-to-N or N2). However, DSM in its current form was developed by Donald Steward as a tool ‘to analyse the flow of information’ in the design, development and operation of systems [61, 62].

The DSM is implemented as an N-square matrix (See Fig 6) which represents functions and processes of systems in constructs of four forms: sequential, concurrent, coupled or conditional [58, 60, 63]. This representation can be applied to depict information flow among types of systems and organisations. These system and organisational types can contain *elements* in the form of components or parameters or resources of the system, development phases, position or responsibilities of members in an organisation and so on.

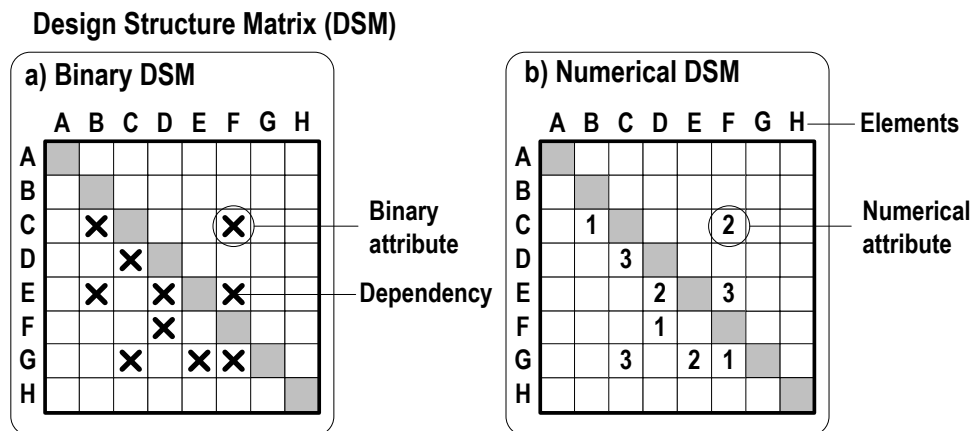


Fig 6 Design structure matrices according to attributes of marked cells

A DSM can also be configured according to *attributes of marked cells* such as between binary DSM and numerical DSM [58, 61, 64]. Binary DSM typically involves the presence or absence of a mark (‘X’ or ‘•’) while numerical DSM could be applied to indicate importance or

probability of repeating an element. In the DSM example in Fig 6, system elements or components are represented along the shaded diagonal. Off-diagonal 'X' marks and numerical values indicate dependency i.e. of one element on another. The labelled 'X' symbol in Fig 6a indicates the dependency of element E on element F.

3. Information flow models and product-service systems

3.1. Overview of Product-Service Systems

According to the United Nations, a Product-Service System (PSS) is 'a competitive system of products, services, supporting networks and infrastructure' [15]. It is made up of services such as product maintenance and parts recycling, that focus on fulfilling customer needs in a manner that is competitive, life-cycle oriented and environmentally friendly. Several authors have defined, described and reviewed the PSS concept by focusing on research and industrial perspectives. The reader is referred to works by Mont [1], Aurich [8], Morelli [16] and Baines et al. [65] for detailed discussions of the PSS concept.

However, two perspectives on the PSS approach have received the most interest in literature: Sustainable Product-Service System (SPS2) and Industrial Product-Service System (IPS2). In the SPS2 approach designers are encouraged to creatively generate ideas so as to reduce the environmental impacts of companies by factors between 4 and 20 [7, 66]. An IPS2 concentrates on business-to-business (B2B) relationships to deliver 'integrated product and service shares' [67]. It offers opportunities for dealing with difficulties and high capital costs associated with expensive core technologies [8]. Core or enabling technology is used in this context to describe a typically expensive device, tool or machinery that offers high-level technological competences and capabilities.

A widely accepted approach to the design and delivery of a PSS identifies three main categories of PSS value proposition: product-, use- or result-orientation [1, 7]. In the product-oriented approach, a company offers the sale of products and also opens channels for customers to access additional services such as upgrades, maintenance, professional advice and consultancy. For the use-oriented approach, a company maintains the rights to a product for use in a service environment made accessible to customers via services such as product leasing, product sharing, product pooling and product renting. Result-oriented approaches involve a company delivering contents of a service to customers, independent of product choice. Schemes such as paying-per-unit service, delivering functional results, activity management and outsourcing, all fall under this category.

3.2. Properties of Product-Service Systems and Information Flow Models

A Product-service system exhibits several aspects that useful diagrammatic tools for information flow would be required to model successfully. These are summarised in Table 2.

Table 2 Key properties of PSS (when viewed as information flows) that a modelling tool would need to successfully model

Property of Product-Service Systems	Characteristics	Information Flow Issues	Literature
Open systems	Input and outputs	Resource use	[2, 4, 65]
	Functions and processes	Value creation for customer and manufacturer	
	System architecture		
	System components		
Social construct	Actors, roles and scenarios	Interfaces and interconnections	[6,16]
	Technological and socio-cultural interactions	Independency, collaboration and distribution	
Business model	Nature of business	Mode and structure of operation	[1-3, 5, 65]
	Domain of application	Scale and scope of operation,	
	Available resources	Orientation and size of operation	

Firstly, PSS can be viewed as open process systems to describe processes and functions, which receive inputs as resources and transmit outputs in the form of value. Inputs could include materials, manpower or information from market studies or customer feedback. Outputs in PSS come in the form of integrated product and service provision. Baines et al. [65] described this as a key premise for the PSS logic. They suggest that the PSS logic is based on the manufacturers' understanding of processes and functions. This understanding, they contend could be used to decrease input (resources) and increase output (value). By applying an appropriate information flow model, a PSS could distinguish between functions and processes performed by its system or organisation. While processes can be used to describe specific activities that have to be carried out in an organisation, functions can be thought of as important areas of activity (such as research) within organisations [12].

Secondly, PSS can also be modelled as a social construct made up of actors, roles and scenarios [6]. This description is required to address what Morelli [16] described as communicational concerns, which are important for the accurate design and development of a PSS.

Communicational concerns for PSS can include how processes and functions are controlled by independent, collaborative or distributive means. This could be achieved by means of interfaces and interconnections for co-operative use and co-ordination of resources in multi-objective and multi-goal driven systems like PSS. These facilities offer channels for technological and socio-cultural interactions which are necessary for the functioning of organisations [16]. Modelling these interfaces and interconnections can also serve as an enabler for realising system objectives and goals. This is because a model presents a depiction of a set of system components or subject area; realised for understanding, analysing or improving the system [28]. When understood, these interfaces and interconnects can be improved or redesigned for enhanced organisational productivity [24]

Thirdly, PSSs offer value propositions in business models for social, economical and technical benefits in relation to the nature of business, domain of application and available resources. This is discussed in further detail below.

The nature of the business is used in this context to describe the mode and structure of business operation. It can be used to model various aspects of a PSS such as: the set up and operation of an organisation; the interfacing of software which controls a product; interconnects and interfaces for a product; processes and functions during manufacturing; schedule and record of information flow in service provisions; and activities requiring exchange of information throughout the lifecycle of a provision for a customer.

The domain of application considers issues such as the orientation and scope of operation for a PSS. This consideration could be useful in identifying inter-domain actors, processes and function. These elements can in turn be used to propose how information flow policies are applied and operations are performed within a domain such as health-care or telecommunications. Orientation is used here to describe the placement or adaptation of a business to its market. This could be in terms of competing companies or exiting products. This consideration with regards to the domain of application offers opportunities to support innovative design and development activities for a PSS based business. Scope of operation for a PSS focuses to the area of operation of the business. The focus may be on a single area such as microfluidics or it could bring together various domains in a large system or organisation. Available resources such as capital, investment and manpower can present important considerations for choice of modelling technique. This relates to the scale or size of the company implementing a PSS. It could also be crucial in deciding the technological facilities and capabilities for communication and networking between business units. Some of these facilities could include virtual private networks, video conferencing and other communication and networking solutions. Scale or scalability is used in this context to describe the ease or grace with which a PSS can handle increasing business operations. Size on the other hand relates to the magnitude of a PSS in terms of available resources of material, information and energy.

4. Modelling information flows for product-service systems: a case study of selective laser melting technology

Selective laser melting (SLM) is an example of a manufacturing technique which promises improvements in time to market, production rate, functionality, part accuracy and versatility when compared to traditional manufacturing techniques. [68] However, its mainstream adoption has had one significant barrier – capital cost. A typical machine for SLM manufacturing such as the MCP RealizerSLM [69] can cost as much as half a million pounds. The argument is that such high capital costs may not be justifiable since at the current rate of return, it could take a very long time before the initial investment can be recouped.

This section presents a case study of possible value propositions designed to implement SLM technology as an IPS2. A data flow model depicting information flow for the implementation of SLM as an IPS2 is also presented.

4.1. Value proposals for SLM implemented as an IPS2

The IPS2 approach offers opportunities for SLM machine manufacturers to develop and deliver business models to facilitate the use of SLM technology. It does this by shifting focus from the sale of an SLM machine (a product) to value propositions associated with the SLM machine and the integrated services required to support the life cycle of the machine.

From the perspective of a user (or customer), the life cycle of a product consists of product purchase, product use and product disposal [8]. During the purchase of the machine, services provided by the manufacturer could include machine installation, providing samples of prototyping units and offering machine insurance. During the use of the machine, the machine manufacturer could offer services such as regular maintenance and the supply of metal powder and replacement gaskets/motors. For the disposal phase or end of life of the machine, services provided by the manufacturer could include take back schemes and machine replacement. Based on these services, possible value propositions for SLM implemented as an IPS2 could include: the sale of an SLM machine backed by a range of integrated services (product-oriented), an SLM machine backed by a range of integrated services made available to customer (use-oriented), and an SLM machine backed by a range of integrated services that meets customer needs (result-oriented).

4.2. Information flow model for SLM implemented as an IPS2

An example of information flow between a customer and the information system provided by a SLM manufacturer is presented. The focus of the example is to model information flow within the social construct of an SLM implemented as an IPS2. In the example a customer interacts and negotiates with an information system in order to request a service.

In order to accomplish this, a data flow diagram (DFD) has been developed as shown in Fig. 7. DFDs are suitable for this case because they are widely applied in the design of information systems in organisations due to their ability to depict data flows and data stores associated with processes. The DFD in Fig. 7 shows information flow for the possible orientations (product, result and use) of SLM implemented as an IPS2. In the diagram, the processes shown as rounded rectangles centre on interactions between the customer and the information system. The customer in this scenario is a business that requires the use of a SLM machine to develop product components. Within the DFD, the customer is an external entity that initiates and terminates the function provided by the information system. Other external entities in this case are training standards (regulatory bodies and standards regarding the SLM technology) and credit bureaus (firms that provide credit information for financial, investment and insurance purposes). The DFD also contains five main data stores: customer records, machine records, service team records, business records and sales records. Customer records provide a log of service that have been delivered to customers whereas machine records detail transactions during a machine's lifecycle such as maintenance, upgrades and software updates. Service team records shows the schedules and availability of service teams while business records provide details of SLM instructors and financial advisers. The sales records contain sales entries for SLM customers.

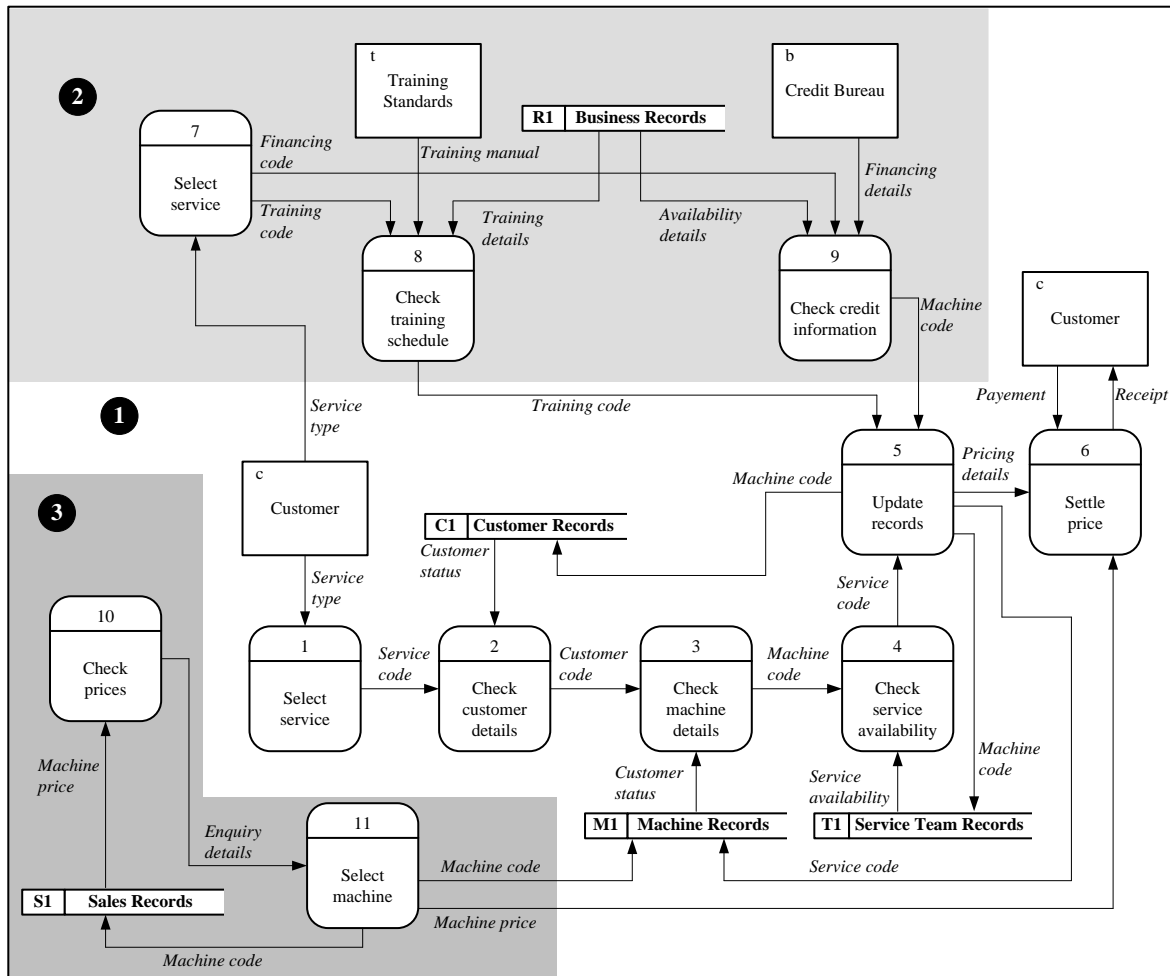


Fig 7 Data flow diagram for an information system to implement selective laser melting as an industrial product-service system

5. Discussion

5.1. Analysis of Modelling Tools

As earlier highlighted, each modelling approach has its strengths and weaknesses. Table 3 highlights some strengths and weakness of the diagrammatic information flow modelling tools described in previous sections. The table highlights relative ease of use, ease of interpretation, time taken to construct and ability to model aspects of a system.

In the previous section, the properties of PSS that require successful modelling were highlighted. Below, the strengths and weaknesses of the diagrammatic tools as methods of modelling these properties are discussed.

For a PSS as an open system, different tools may be chosen depending on what is being modelled. For instance, DFDs may be suitable for modelling the organisation of a PSS while the

IDEF or GRAI approach may be suitable for representing PSS manufacturing process. An automated device, software system or a manufactured instrument could be ideally modelled by PNs, IPO charts and Design Structure Matrices respectively. Similarly, a small company for instance, may adopt a flexible and easy to maintain model such as a DFD while a large company requiring a complete mapping of its functions and processes may opt for a DSM.

Table 3 Strengths and Weaknesses of Information Flow Diagrammatic Models

Modelling Tool	Strengths	Weaknesses	Related Tool
Data flow diagrams	<p>Suitable for sequential representation of information flow</p> <p>Flexible and easy to maintain</p> <p>Readily available context makes it easy to translate and read</p> <p>Varying levels allows focus on area of interest</p> <p>Popularly used and supported in industry</p>	<p>In large systems such as enterprises, these models may become cumbersome in representation difficult to interpret</p> <p>time consuming in construction</p> <p>Ignores time dependent events or event driven processes</p>	-
Integrated DEFinition method of modelling functionality and information modelling (IDEF0 and IDEF1)	<p>Suitable for analysing a business</p> <p>Ideas and concept are easy to grasp and apply</p> <p>Allows for controlled and incremental system description</p> <p>Supported by standards and widely used in industry</p> <p>Supported by closely related methodologies such as IDEF3 for process flow</p> <p>Makes use of limited notation making them easy to interpret</p>	<p>Can be time-consuming and inconsistent</p> <p>Can be difficult to integrate related methodologies</p> <p>May not be suitable for system development and documentation</p>	IDEF modelling technique
Graphes à Résultats et Activités Interreliés (GRAI) grids and nets	<p>Suitable for supporting decision making processes in manufacturing enterprises</p> <p>Highlights opportunities for synchronicity and concurrency in systems by depicting the durations for the system processes</p> <p>Enhances enterprise performance by offering diagnosing mechanisms which can identify defects in operation and reasons for management gaps</p>	<p>Only concentrates on the information flow related to decision making processes</p> <p>Fails to provide structure details such as: enterprise processes, the distribution and use of resources and the organisation or enterprise being modelled</p>	GRAI modelling technique
Petri nets	<p>Suitable for automated or event-driven systems</p> <p>Based on a solid mathematical foundation</p> <p>Allows for extensions and modifications</p>	<p>Tough to learn and popularize</p> <p>Easily becomes too complicated even in reasonably sized systems</p>	-

Input- process- output (IPO) diagrams	Suitable for hierarchically structured programs Presents a useful avenue to begin program and system designs Provides ready-made documentation of a system after its implementation Identifies procedural flow from input to output Offers clear definitions	Can quickly become cluttered in big programs or systems; becoming difficult to interpret Can be bulky since it uses a page for each module irrespective of module size Difficult to maintain Not widely used in industry Lacks support for loops, conditions, data structures or data links	HIPO (Hierarchy plus Input-Process-Output) modelling technique
Design structure matrix	Suitable for representing the entire range of interactions among functions compact and clear representation can assist a company identify and focus on key issues supports continuous learning, development and innovation	Difficult to construct since data may not always be available Data required may be vast and difficult to assimilate Do not include task duration, time lines or estimates for task duration	-

Further consideration for PSS as an open system can raise issues which relate to standardisation and support for documentation. Collaboration in delivering provisions may require the preparation, presentation and submission of documents using standard information flow techniques such as DFDs and IDEFØ/IDEF1. In these cases, PSS based companies must follow the required regulations for documentation. It would also be sensible and consistent (where possible) to employ the same modelling and documentation schemes throughout the company. As a social construct, what ever model is chosen should be constructed from the customer perspective. Doing this can aid PSS operators concentrate on customer fulfilment and promote this focus during interactions. These models can then assist operators by offering either a great deal of insight or an insightful model for design processes [57].

Within a social construct, the selection and application of these information models can also be based on communication considerations for PSS. These considerations can be applied for issues relating to size, scope, resources etc. For instance, the model representation in a large corporation or system may not be suitable for a small company seeking to integrate product and service provisions to its customers. The selected models can then act as information management mechanisms for an information system to drive effective and efficient organisation and communication in a PSS.

For PSS as business models, modelling tools for PSS need to be developed with flexibility in mind so as to ease time and financial costs associated with organisation shifts. This flexibility can also have the added benefit of fostering the development of conceptual models. These diagrams or models for information flow will also have to be first: carefully selected and secondly: rigorously and creatively designed if they are to offer intuitive and computational benefits [70, 71]. A DSM for instance can be used for clearly defined representation of functions

and processes of an innovative product. PNs, likewise, are suited to information flow representations which require a great deal of mathematical support.

Table 4 Key properties of PSS (when viewed as information flows) and design implications

Property of Product-Service Systems	Characteristics	Modelling Considerations	Implications for Design
Open systems	Input and outputs	Standardisation and support for documentation	Clear definitions of system functions and processes
	Functions and processes	Consistency in approach	
	System architecture	Focus on system interactions	
	System components		
Social construct	Actors, roles and scenarios	Organisation for the system	Creation of models from customer perspective
	Technological and socio-cultural interactions	Communication in the system	
		Management of the system	
Business model	Nature of business	Flexibility of models	Development based on context of use for business and applications
	Domain of application	Rigorous and creative models	
	Available resources		

All three considerations, open systems, social construct and business model as summarised in Table 4, must be considered independently and collectively by the designers and implementers of the PSS. The interplay of these issues, in particular, could narrow down the choice to the most suitable tool. This tool can then be used to model various aspects of the business. Good designers may also decide on a combinatorial approach in which different models are amalgamated to give a more generic approach. Conversely, designs could be based on modifications to a model to reflect characteristics of a particular PSS.

5.2. Future research directions

This section recommends three possible directions for future research. The first area recommends studies for examining problems of selection and suitability of information flow models. The second area suggests studies for investigating the wider issue of modelling for a PSS, while the third area recommends research for factors exposed by the case study in Section 4.

5.2.1. Information flow models: selection and suitability

It is important to note that the purpose of modelling information is to communicate knowledge and understanding [17]. Consequently, modelling tools must be suited to the purpose for which they are required. Possible future research could therefore focus on methodologies for selecting and applying models for information flow. These studies could focus on characteristics such as those presented in Section 4 (social construct, open system and business model) to make recommendations and assign tools to different cases. Initial empirical studies of companies could be conducted to capture modelling needs and information flow requirements of designers and

operators. Such studies may adopt a sector approach by exploring the needs of particular industries such as the agriculture sector or the semiconductor industry. Alternatively, the studies may focus on the requirements of company departments or business processes. These industrial studies could complement the literature analysis for the creation of robust frameworks for model selection and application.

5.2.2. PSS modelling: completeness and strategy

Modelling information flow for a PSS presents a partial view of a PSS. As shown in this review, all the tools depict a function as a set of processes connected by inputs and outputs. Other features represented by the modelling tools reviewed in this paper include: data stores and external entities (DFD), interfaces, controls and mechanisms (IDEF) and decision flows (GRAI). Consequently, to create a 'complete picture of a PSS' other aspects would have to be considered and modelled. These aspects include business processes, product and services, processes, work flow and company structure.

As earlier mentioned, a PSS can be modelled by several techniques such as function-oriented modelling, object-oriented modelling, conceptual modelling and service modelling. A comparative analysis of different modelling techniques for use in PSS modelling is therefore recommended for future studies. Such analysis could act as a starting point for strategizing the process of modelling a PSS. In order to accomplish this, studies would need to be conducted to propose an appropriate modelling technique to present a complete picture of a PSS. An example of research in this area is the System Modelling Language (SysML) that was developed by the Object Management Group (OMG) consortium and the International Council on Systems Engineering (INCOSE) [72]. SysML was developed to support the analysis and concept phase of systems design. Similar approaches could also be adopted within the PSS community whereby future work could concentrate on studying the characteristics of PSSs and proposing a comprehensive PSS modelling technique.

5.2.3. Information flows for Product-Service Systems: interactions and negotiations

In terms of PSS information flows, future research could focus on managing interactions and negotiations during the design and delivery of PSSs. Information flow may involve several actors receiving the same input for different processes. Furthermore, several processes may run concurrently or interdependently. These different combinations present a wide range of possible scenarios for information flow. It is for this reason that the management of interactions and negotiations is important for information flow in PSSs.

Managing interactions and negotiations for PSS delivery could play an important role in harmonising perspectives of PSS designers and operators. For instance, some divisions of company that delivers a PSS may adopt a 'reactive' approach to their operation whereas others may be 'proactive' or adaptive in nature. Linking both divisions is a challenge that could require the development of new PSS information flow or operating policies.

Possible future research is therefore required to explore negotiations during PSS design and delivery processes. This research area may examine PSS processes involving a single company (intra-organisational), between companies (inter-organisational) or between a company and its customers (customer-focused). Other areas of research for managing interactions could explore techniques for keeping track and maintaining an audit trail of information on PSS processes. An information system such as the one modelled in Section 4 provides a useful avenue for modern businesses to manage interactions via communication channels such as information communication technologies (e.g. electronic mail and video conferencing). Future research could therefore focus on developing, customising or identifying information systems suitable for use in PSS design and delivery.

6. Conclusions

Product-Service Systems (PSSs) are schemes proposed for integrated solutions of products and services as a single provision. They can be modelled by means of function-oriented design which involves breaking down systems into manageable interacting parts. Information flow representations for function-oriented design provide useful mechanisms for describing function input and output in systems and organisations.

This review paper has attempted to identify some key diagrammatical information flow models. Data-flow diagrams (DFDs) can be used in organisations to propose information flow path (logical view) and to represent actual flows (physical view). DFDs do this by depicting processes, external entities, data stores and flows in sequential representations. Information flows in manufacturing can be highlighted by the Integrated DEFinition method of modelling functionality and information modelling (IDEFØ/IDEF1) approaches which makes use of boxes (representing functions) and arrows which indicate relations, input, control, output, and mechanisms associated with the function. Graphes à Résultats et Activités Interreliés (GRAI) grids and nets provide information flow descriptions to support decision making processes in an organisation or a system. PNs deliver representations of information flow in development and simulation of event-driven and automated manufacturing systems. Input-Process-Output (IPO) diagrams offer information flow descriptions in programs but can also be extended to describe systems with varying complexity. Design Structure Matrices present compact, visual, matrix representations for systems analysis; offering a roadmap of system level knowledge. An example of a DFD for selective laser melting implemented as a PSS was presented as a case study to show the application of information flow models. The various information flow diagrams reviewed in the paper aid comprehension of PSS by modelling their key properties as social constructs in open process systems for value propositions in business models.

The review paper has also attempted to highlight how these key properties are related to information flow, what characteristics of PSS any information flow modelling tools are required to model successfully and possible future research directions. These considerations are important

in selecting, modifying or combining approaches for information flow representation suited to planned or existing PSS.

Acknowledgments: The authors would like to extend their sincere thanks to the EPSRC, for its support via the Cranfield IMRC, of the work carried out in the preparation of this paper.

References

- [1] Mont OK (2002) Clarifying the concept of product-service system. *J Clean Prod* 10: 237-245. doi: 10.1016/S0959-6526(01)00039-7
- [2] Manzini E, Vezzoli C (2002) Product-service systems and sustainability: Opportunities for sustainable solutions. UNEP TIE, Paris. Available at: <http://www.unep-tie.org/scp/design/pdf/pss-imp-7.pdf>. Accessed 15 Nov 2008
- [3] Tukker A, Tischner U (2006) Product-services as a research field: past, present and future; Reflections from a decade of research. *J Clean Prod* 14:1552-1556. doi:10.1016/j.jclepro.2006.01.022
- [4] Komoto H, Tomiyama T, Nagel M (2005) Life Cycle Simulation for Analyzing Product Service Systems, *EcoDesign* 2005 386-393. doi:10.1109/ECODIM.2005.1619251
- [5] Becker J, Beverungen D, Knackstedt R (2008) Reference Models and Modeling Languages for Product-Service Systems Status-Quo and Perspectives for Further Research. *HICSS-41*: 105. doi: 10.1109/HICSS.2008.369
- [6] Durugbo C, Tiwari A, and Alcock JR (2009) An Infodynamic Engine Approach to Improving the Efficiency of Information Flow in a Product-Service System. *Proceedings of the 1st CIRP IPS2 Conference*, pp. 107–112.
- [7] Tukker A (2004) Eight types of product-service system: eight ways to sustainability? Experiences from *SusProNet*. *Bus Strat Environ* 13: 246–260. doi: 10.1002/bse.414
- [8] Aurich JC, Fuchs C, Wagenknecht C (2006) Life cycle oriented design of technical Product-Service Systems. *J Clean Prod* 14:1480-1494. doi: 10.1016/j.jclepro.2006.01.019
- [9] Komoto H, Tomiyama T (2008) Integration of a service CAD and a life cycle simulator, *CIRP Ann-Manuf Techn* 57: 9-12, doi: 10.1016/j.cirp.2008.03.001.
- [10] Sommerville I (1992) *Software Engineering*, 4th edn. Addison Wesley, London
- [11] Colquhoun GJ, Baines RW, Crossley R (1993) A state-of-the-art review of IDEF0, *Int J Comp Integ M* 6:252-264. doi: 10.1080/09511929308944576
- [12] Martin J, McClure C (1985) *Diagramming Techniques for Analysts and Programmers*, Prentice Hall Englewood Cliffs, New Jersey
- [13] Philipps J, Rumpe B (1997) Refinement of information flow architectures. *IEEE ICFEM* 1997 203-212. doi: 10.1109/ICFEM.1997.630427
- [14] van Gigch, J.P. (1991), *System Design Modeling and Metamodeling*, Plenum Press, New York
- [15] United Nations. 1987. "Report of the World Commission on Environment and Development." General Assembly Resolution A/RES/42/187, 11 December 1987. Available at <http://www.un.org/documents/ga/res/42/ares42-187.htm>. Accessed 21 May2008
- [16] Morelli N (2006) Developing new product service systems (PSS): methodologies and operational tools. *J Clean Prod* 14: 1495-1501. doi:10.1016/j.jclepro.2006.01.023
- [17] Scheller A (1990) Information modelling for distributed applications. *IEEE Workshop on Future Trends of Distributed Computing Systems*, 494-500. doi: 10.1109/FTDCS.1990.138369
- [18] Hungerford BC, Hevner AR, Collins RW (2004), Reviewing software diagrams: a cognitive study, *IEEE T Software Eng* 30:82-96. doi:10.1109/TSE.2004.1265814
- [19] Canfora G, Sansone L, Visaggio G (1992) Data flow diagrams: reverse engineering production and animation. *ICSM* 1992 366-375. doi: 10.1109/ICSM.1992.242522

- [20] Tucker D, Leonard R (2001) An Innovative Approach for Using the GRAI Methodology for Reengineering the New Product Introduction Process. *Int J Flex Manuf Sys* 13:177-193. doi: 10.1023/A:1011135518865
- [21] DeMarco T (1979) *Structured Analysis and System Specification*. Yourdon Press, New Jersey
- [22] Du TC, Lin CJ, Liu C-G (2000) Proposing an information flow analysis model to measure memory load for software development. *Inform Software Tech* 42:743-753. doi: 10.1016/S0950-5849(00)00117-8
- [23] Ross DT, Schoman KE (1977) *Structured Analysis for Requirements Definition*. IEEE T Software Eng SE-3:6 – 15. doi:10.1109/TSE.1977.229899
- [24] Turetken O, Schuff D (2007) The impact of context-aware fisheye models on understanding business processes: An empirical study of data flow diagrams. *Inform Manage* 44:40-52. doi:10.1016/j.im.2006.10.004
- [25] Butler G, Grogono P, Shingha R, Tjandra I (1995) Retrieving information from data flow diagrams. *WCRE* 1995 22-29. doi: 10.1109/WCRE.1995.514690
- [26] Gane C, Sarson T (1979) *Structured Systems Analysis: Tools and Techniques*. Prentice-Hall, New Jersey
- [27] Knowledge Based Systems Inc (2006) *IDEF Family of Methods: A structured approach to enterprise modelling and analysis*. Available at: <http://www.idef.com/>. Accessed 23 March, 2008
- [28] Federal Information Processing Standards (1993) FIPS PUB 183, *Integration Definition for Function Modelling (IDEF0)* National Institute for Standards and Technology, Maryland
- [29] Software Engineering Standards Committee of the IEEE Computer Society (1998) IEEE Std 1320.1-1998 *Standard for Functional Modelling Language—Syntax and Semantics for IDEF0*. The Institute of Electrical and Electronics Engineers, Inc, New York
- [30] Sullivan DO (1991) Project management in manufacturing using IDEF0. *Int. J. Project Manage* 9:162-168. doi:0263-7863/91/030162-07
- [31] Ang CL, Luo M, Gay RKL (1995) Knowledge-based approach to the generation of IDEF0 models. *Comput Integr Manuf* 8:279-290. doi: 10.1080/002075497195380
- [32] Kusiak A, Larson TN, Wang J (1994) Reengineering of design and manufacturing processes. *Comput Ind Eng* 26:521-536. doi:10.1016/S0360-8352(00)00059-0
- [33] Bernus P, Schmidt G (1998) *Architectures of Information Systems*. In: Bernus P, Mertins K, Schmidt G (eds) *Handbook on Architectures of Information Systems*, Springer-Verlag, Berlin, pp 1-10
- [34] Ho KF, Ridgway K, Lake BC (1994) To Design Cellular Manufacturing Systems For An Electric Motor Manufacturer Using Grai Methodology. *Factory 2000* 189-195. doi:10.1049/cp:19940859
- [35] Lingzhi L, Leong AC, Gay RKL (1996) Integration of information model (IDEF1) with function model (IDEF0) for CIM information systems design. *Expert Syst Appl*, 10 (3-4 SPEC. ISS.), 373-380
- [36] Chen P-H, Wan C, Tiong RLK, Ting SK, Yang Q (2004) Augmented IDEF1-based process-oriented information modeling. *Automat Constr*, 13:735-750. doi:10.1016/j.autcon.2004.04.013
- [37] McCarthy I, Menicou M (2002) A classification schema of manufacturing decisions for the GRAI enterprise modelling technique, *Comput Ind* 47:339-355. doi:10.1016/S0166-3615(02)00002-7
- [38] Doumeingts G (1989) GRAI Approach to Designing and Controlling Advanced Manufacturing Systems in CIM Environment. In: *Advanced Information Technologies for Industrial Material Flow Systems*, Nof SY, Moodie CL (eds.), NATO ASI Series, Springer-Verlag, Berlin, pp. 461-529
- [39] Merlo C, Girard Ph. (2004) Information system modelling for engineering design co-ordination. *Comput Ind* 55: 317-334. doi:10.1016/j.compind.2004.08.008
- [40] Wainwright CER, Ridgway K (1994) The Application of GRAI as a Framework for Manufacturing Strategy Process. *Factory 2000* 294-301. doi:10.1049/cp:19940874
- [41] Doumeingts G, Vallespir B, Chen D (1998) GRAI grid decisional modelling. In: *Handbook on Architectures of Information Systems*, Bernus P, Mertins K, Schmidt G (eds.), Springer-Verlag, Berlin, pp 313-337
- [42] Vernadat F (1996) *Enterprise Modeling and Integration*. Chapman and Hall, London
- [43] Leondes CT (1995) *Intelligent Knowledge-Based Systems: Information Technology*. Kluwer Academic Publishers Norwell, Massachusetts
- [44] Hilt B, El Mhamedi A, Noaghiu C (1994) Modelling Of Information Flow Approach Using SADT and Petri Nets, *Factory 2000* 326-331

- [45] Zhou MC, DiCesare F (1993) Petri Net Synthesis for Discrete Event Control of Manufacturing Systems, Kluwer Academic Publishers, London
- [46] Bonney MC, Zhang Z, Head MA, Tien CC, Barson RJ (1999) Are push and pull systems really so different? *Int J Prod Econ* 59: 53-64. doi:10.1016/S0925-5273(98)00094-2
- [47] Ou-Yang C, Lee JC (2000) Evaluation of the information flow in a hierarchical shop floor control framework. *Int J Syst Sci* 31: 977-1001. doi:10.1080/002077200412131
- [48] Murata T (1989) Petri nets: properties, analysis, and applications. *Proc IEEE* 77: 541-580. doi:10.1109/5.24143
- [49] Lien YE (1976) A Note on Transition Systems. *Inform Sciences* 10: 341-362. doi: 10.1016/S0020-0255(76)91010-0
- [50] Varadharajan V (1990) A formal approach to system design and refinement. *CompEuro* 90: 544-545. doi:10.1109/CMPEUR.1990.113675,
- [51] Wakefield RR, Sears GA (1997) Petri nets for simulation and modelling of construction systems. *J Constr Eng M ASCE* 123: 105-12. doi:10.1061/(ASCE) 0733-9364(1997)123:2(105)
- [52] Stay JF. (1976) HIPO and Integrated Program Design. *IBM Syst J* 15:143-154
- [53] Katzan H (1976) Systems Design and Documentation: An Introduction to the HIPO Method, Van Nostrand Reinhold, New York
- [54] LaBudde K (1987) Structured programming concepts, McGraw-Hill, New York,
- [55] Nosek, JT, Schwartz RB (1988) User validation of information system requirements: some empirical results. *IEEE Trans Software Eng* 14: 1372-1375. doi: 10.1109/32.6180
- [56] Davis WS (1998) HIPO (hierarchy plus input-process-output) In: The Information System Consultant's Handbook: Systems Analysis and Design, Davis WS, Yen DC (eds.) CRC Press LLC, Florida, pp 503-510
- [57] Eppinger SD (1991) Model-based approaches to managing concurrent engineering. *J Eng Design* 2: 283-290. doi:10.1080/09544829108901686
- [58] Browning TR (2001) Applying the design structure matrix to system decomposition and integration problems: a review and new directions. *IEEE Trans Eng Manage* 48: 292-306. doi:10.1109/17.946528
- [59] Helo PT (2006) Product configuration analysis with design structure matrix, *Ind Manage Data Syst* 106: 997-1011. doi: 10.1108/02635570610688896
- [60] Syed AS, Berman K (2007) DSM as a knowledge capture tool in CODE environment. *J Intell Manuf* 18: 497-504. doi: 10.1007/s10845-007-0058-0
- [61] Steward DV (1981) The design structure system: a method for managing the design of complex systems. *IEEE Trans Eng Manage* 28: 71-74
- [62] Oloufa, AA, Hosni YA, Fayez M, Axelsson, P (2004) Using DSM for Modeling Information Flow in Construction Design Projects. *Civ Eng Environ Syst* 21: 105-126. doi:10.1080/10286600310001638474
- [63] Farid A, McFarlane DC (2006) An approach to the application of the design structure matrix for assessing reconfigurability of distributed manufacturing systems. *DIS 2006*: 121-126. doi:10.1109/DIS.2006.10
- [64] Yassine A (2007) Investigating product development process reliability and robustness using simulation. *J Eng Design* 18: 545-561. doi:10.1080/09544820601011690
- [65] Baines T, Lightfoot HW, Evans S, Neely A, Greenough R, Peppard J, Roy R, Shehab E, Braganza A, Tiwari A, Alcock JR, Angus JP, Bastl M, Cousens A, Irving P, Johnson M, Kingston J, Lockett H, Martinez V, Michele P, Tranfield D, Walton IM, Wilson H (2007) The state-of-the art in Product Service Systems. *P I Mech Eng B-J Eng* 221: 1543-1552. doi: 10.1243/09544054jem858
- [66] Roy R (2000) Sustainable product-service systems. *Futures*, 32:289-299. doi: 10.1016/S0016-3287(99)00098-1
- [67] CIRP (2009) CIRP (College International pour la Recherche en Productique) Working Group - Industrial Product-Service-Systems (IPS²) Available at <http://www.lps.rub.de/schwerpunkt/cirp/>. Accessed 21 February 2008
- [68] Yadroitsev I, Thivillon L, Bertrand Ph and Smurov I, (2007) Strategy of manufacturing components with designed internal structure by selective laser melting of metallic powder, *Appl Surf Sci*, vol. 254, issue 4, pp 980-983 doi: 10.1016/j.apsusc.2007.08.046
- [69] MCP-Group, 2008, MCP RealizerSLM –SLM Technology. Available at <http://www.mcp-group.com/rpt/rptslm.html>. Accessed 14 February 2008

- [70] Gurr C, Tourlas K (2000) Towards the principled design of software engineering diagrams. ICSE 2000 509-518. doi: 10.1145/337180.337371
- [71] Sen T (1992) Diagrammatic knowledge representation. IEEE T Syst Man Cyb 22:826-830. doi: 10.1109/21.156595
- [72] Object Management Group, 2009, OMG Systems Modeling Language (OMG SysML™) Version 1.1, Available at: <http://www.omg.org/spec/SysML/1.1>. Accessed 15 October, 2008