A collective artefact design of decision support systems: design science research perspective

A Collective Artefact Design of Decision Support Systems: Design Science Research Perspective[1]

Shah Jahan Miah, *Information Systems Discipline, College of Business, Victoria University, Melbourne, VIC, Australia*

Don Kerr, *Informatics, Faculty of Business and Arts, University of the Sunshine Coast, QLD, Australia*

Liisa von Hellens, *School of Information and Communication Technology, Griffith University, QLD, Australia*

Abstract

Purpose –The knowledge of artefact design in design science research can have an important application in the improvement of decision support systems (DSS) development research. Recent DSS literature has identified a significant need to develop user-centric DSS method for greater relevance with respect to context of use. To address this, this study develops a collective DSS design artefact as method in a practical industry context.

Design/methodology/approach – Under the influence of goal-directed interaction design principles the study outlines the innovative DSS artefact based on design science methodology to deliver a cutting-edge decision support solution, which provides user-centric provisions through the use of design environment and ontology techniques.

Findings –The DSS artefact as collective IT applications through the application of design science knowledge can effectively be designed to meet decision makers' contextual needs in an agricultural industry context.

Research limitations/implications – The study has limitations in that it was developed in a case study context and remains to be fully tested in a real business context. It is also assumed that the domain decisions can be parameterised and represented using a constraint programming language.

Practical implications – We conclude that the DSS artefact design and this development successfully overcomes some of the limitations of traditional DSS such as low user uptake, system obsolescence, low returns on investment and a requirement for continual re-engineering effort.

Social implications – The design artefact has the potential of increasing user uptake in an industry that has had relevancy problems with past DSS implementation and has experienced associated poor uptake.

Originality/value –The design science paradigm provides structural guidance throughout the defined process, helping ensure fidelity both to best industry knowledge and to changing user contexts.

Keywords: user-centredness; DSS; artefact design; design science research

1. Introduction

The knowledge of artefact design in design science research can have an important application in the improvement of decision support systems (DSS) development research. Recent DSS literature has identified significant needs to improve quality and relevance of DSS development, particularly to achieve better engagement of industry and decision makers (Hosack, Hall, Paradice and Courtney, 2012; Arnott and Pervan, 2012; Arnott, 2006). Several distinct subfields have emerged in DSS design. Arnott and Pervan (2008) classified these subfields as personal DSS, group DSS, intelligent

[1] **This paper has been accepted for publishing in the Information Technology & People (Issue 3 or 4), 2013. This journal has been ERA listed as one of the 'A Ranked' Information Systems Journals.**

DSS, knowledge management based DSS, data warehousing, negotiation support systems and enterprise reporting and analysis. Our study targets the personal DSS class where a knowledge-base drives end-user processes and decision making. In this paper we wish to relate personal DSS development to mainstream information systems (IS) application design, and as such we are concentrating on a design research area of user-centred IS application. We refer to the design artefact outlined in our study as UCDE (User-Centred Design Environment).In this paper our aim is to describe the design research to develop UCDE as a method in order to improve relevance of DSS development particularly to meet decision makers' contextual needs in business.

In their analysis of historical importance of DSS research, Hosack et al. (2012) suggested that DSS research needs shift the focus to deliver more customer-centric solution. For IS development, Iivari and Iivari (2011) analysed existing literature on UCD (User-Centred Design) methods used in IS development. The authors suggested four aspects of user-centredness, namely, a focus on the system user; a focus on user work-centredness; a focus on user involvement; and a focus on system personalisation. As such,Iivari and Iivari (2011) mentioned that the extension of work-centredness to activity-centredness could be a significant UCD research challenge for the future. In our study, we aim to design an artefact that facilitates decision support features to stakeholders through their engagement and the roles they play in decision support. So in this paper, work-centredness is manifested through an activity-based principle in the case organisation used in our study. This organisation is a government-owner agricultural advisory and regulatory service called the Queensland Department of Primary Industries (QDPI).

In the design research domain, a DSS design artefact can be a construct, model, method or instantiation (Arnott and Pervan, 2012). March and Smith (1995) define a method to be "a set of steps (an algorithm or guideline) used to perform a task" (p. 257). In our research, the design artefact provides user-centred features for a particular group of decision makers in an agricultural industry context. The artefact can be seen as an accepted decision making protocol in which end-users (farmers), domain experts (extension professionals) and managers (regional area managers) are the key players. This paper illustrates the design artefact as a work-oriented activity-centred UCD method to DSS development, in which the method accommodates the key players' roles, particularly for task allocation, organising the knowledge-base and utilising the knowledge in a DSS application design, based on an established context.

1.1Our design artefact to DSS development

Our research extends the analytical framework of user-centredness towards an activity-centred method to facilitate context sensitive DSS design. The proposed design artefact as a method that is capable of generating usable DSS applications. This proposed method adopts features of design environment technique for decision makers under the principle of goal-directed interaction design (Iivari and Iivari, 2011). The entire design is guided through the knowledge of design science research in IS.

Iivari and Iivari (2011) defined the goal-directed interaction principle as a design for producing "power and pleasure for users" that includes user behavioural functions and their information needs. In addition, Hevner, March, Park and Ram (2004) stressed that, in design science research, the technical definitions and business understanding need to be consistently represented and assumed, and this includes subsequent modifications by target decision makers within its context of use. For well-defined domains, it is claimed that ontology, as a conceptual modelling technique, has the potential to improve the structuring of knowledge. Ontology refers to a particular view of the properties that comprise the world, and how those properties relate to each other (Gennari et al. 2003). The use of ontology to model knowledge can lead towards the development of a solid, contextually relevant cognitive base that enables effective knowledge representation for a specific problem domain (Evermann, 2005). This can result in a useful knowledge-based platform for the development of a contextually relevant knowledge-base. The ontology has been extensively used for

DSS developments, such as in the domain of medical emergency management for mass gatherings (Haghighi, Burstein, Zaslavsky and Arbon, 2013). The study by Haghighi et al. (2013) used ontology to resolve inconsistencies of terminology to enhance communication support among medical emergency personnel. Our study uses the ontology for better knowledge management in decision support, in terms of providing common vocabulary (for effective knowledge sharing) to different stakeholders in an agricultural industry context. Using such ontology, that is how we adapt the UCD method to go beyond the purpose of adding rules to reconfigure the DSS artefact within the problem context.

In many domains ontologies already exist or common industry usage provides a de facto standard that can be made explicit. In the system engineering literature, ontology is a formal knowledge-structuring technique in which explicit specification of the problem domain can be presented to aid in the design of a solution. Motivated by Evermann (2005), who promoted the concept of cognitive modelling, we strive to understand and articulate the perceived reality around a typical decision support problem. With this in mind, we use ontology as a concept for structuring and representing problem specific knowledge (e.g. decision making realities in a farming context) into a knowledge repository. In our design artefact (the DSS design environment), we call this an ‘ontology repository’, and it is the main component of the proposed artefact design.

In addition, the ontology technique is more flexible as it allows for interoperability with other systems, and for related applications to be developed as separate projects. The design environment technique suggests a solution suited to accommodating both domain knowledge and the local changing contextual information of decision makers. The design environment is therefore specified to be domain independent. This technique is relevant to our DSS design problem, as it is clearly desirable to have a collective DSS artefact that combines the technical integrity of professional development with the context relevance of a solution tailored to the individual context.

2. Study Background

Improvement of the artefact design knowledge is an essential component of design science research (Hevner et al., 2004; Hevner, 2007).Design research “…seeks to create innovations that define the ideas, practices, technical capabilities, and products through which the analysis, design, implementation, management, and use of information systems can be effectively and efficiently accomplished.” (Hevner et al., 2004, p.76). This understanding can be helpful in guiding straightforward IT artefact design, if the main focus is in designing a new IT solution. In addition, Hevner et al. (2004) suggested that design science research must talk about the creation of an innovation and purposeful development for a specific problem domain. Iivari (2007, p.56) argued that “The primary interest of Information Systems lies in IT applications and therefore Information Systems as a design science should be based on a sound ontology of IT artefacts and especially of IT applications”. Further to this, Iivari (2007) argued that the IS in design science builds from IT meta-artefacts that can support concrete IT application development. This implies that a collection of innovative IT artefacts that can reinforce quality by creating effective design to meet the needs of the users as well as being able to fulfil the process, users’ and situational requirements within organizations. The definitions of Iivari (2007) and Hevner et al. (2004) establish two useful views that can help define a useful DSS artefact design and its properties. The table 1 illustrates the previous arguments on both views in design science literature.

The view of IT artefact design	*The view of IS artefact design*
The artefacts are constructs, models, methods, and instantiations. Purposeful artefacts are built to address here to unsolved problems (March and Smith, 1995)	Gregor and Jones (2007) described two kinds of design artefacts: product artefact and process artefacts.

The output of design science research is virtual artefacts (software and systems) that alter the real world in beneficial ways (Blum, 1996; Purao, 2012)	IS in general comprises organizational (human) as well as technical (software) components. The IS in design science builds from IT meta-artefacts that can support concrete IT application development (Iivari, 2007).
The frameworks and approaches (Defined by Hevner et al. 2004; March and Smith, 1995) of IT artefact design have very little discussions and clarifications regarding underpinning philosophies, but most seem to be based on positivism, traditional realism, or pragmatism (Carlsson, 2006).	Artefact design in IS should have sound ontology and epistemology in terms of the types of knowledge associated with design science research. Design artefact can sometimes, be developed based on a positivistic epistemology reflecting a realistic ontology. The evaluation of the same artefact may follow an anti-positivistic epistemology and an anti-realistic ontology (Iivari and Venable, 2009)
The IT artefact view by March and Smith (1995) and Hevner et al. (2004) have a positivistic epistemological bias (Niehaves 2007).	It is suggested that the anti-positivistic epistemology is also relevant in designing innovative IS artefact in design science research (Iivari and Venable, 2009).
In designing IT artefact, design science research may apply both nomothetic and idiographic methods (Hevner et al., 2004).	Venable (2006) and Venable and Travis (1999) identify interpretive methods as appropriate for naturalistic outputs in design science research (Iivari and Venable, 2009).

Table 1: View of IS artefact design

The background above implies that there are two established understandings on design artefact in IS design science literature. However, attention must be paid to how the application of these theories can be different based on what types of system artefact will be designed for target users. For designing DSS, many solutions have been developed through the application of design science methodologies. Arnott and Pervan (2012) argued that most of previous DSS developments were somehow met through Havner's definition of "creates and evaluates IT artefacts intended to solve identified organisational problems" (Hevner et al. 2004, p.77). Evidence of this can be viewed through many DSS development studies (Muntermann, 2009; Purao and Storey, 2008). Beyond these DSS design studies where an IT based system solutions are the key focuses, our study explores the theories of designing combined dynamic DSS artefact that will be tailor-able for users' design need. The DSS artefact is seen from a collective innovation perspective as a socio-technical design. The adopted view is similar to the view of Mackrell, Kerr and von Hellens (2009), in which a socio-technical method has been utilised to develop an agricultural decision support system. Carlsson (2007) suggested that IS artefact design can be perceived as socio-technical systems. We will be looking at the relevant components (roles of different users and their context of use) of the socio technical view of design science as a guiding principle for our DSS artefact design.

Winter (2008) suggested that although many contributions have been made to the justification of design, the typology of artefacts, or specific problem solutions, rigour-related aspects are not yet sufficiently standardized to the design research community. To design a DSS artefact our study focuses on the issues associated with the development of stand-alone DSS solutions that fail to meet rapidly changing demands within a business context. The artefact we propose can support flexible space to develop DSS applications through an understanding of the end-users' work activities and the context in which they work (Iivari and Iivari, 2011). As an illustration of the work-oriented activity-centred method, the conceptual DSS artefact accommodates the key players' roles particularly in relation to task allocation, organising the knowledge-base and knowledge utilisation in the DSS application design. The sub-section below outlines some background with respect to design environments that focus on user-centred solution design.

2.1 Design environment and user-centeredness

Winograd (1995) described a “design environments” philosophy as one that, when an application is being developed, looks at the system, the users and the situation of use as a whole. Recent work expands on Winograd’s (1995) ideas, and these include domain-oriented design environments for empowering creative knowledge work (Fischer, 1999); constructive design environments for systems development (Gammack, 1999); wikis design for IS (information systems) teaching (Kane and Fichman, 2009); design for online communities (Ren, Kraut and Kiesler, 2007); and a call for application development where situation of use can be explicitly addressed within the design science tradition itself (Hevner et al., 2004). All these recognise the ongoing role of emergent knowledge and the importance of the ever-changing organisational context in technology uptake and use.

To understand user factors and the context of system use, Iivari and Iivari (2011) suggested an important perspective should be “the relationship between people, technology, work requirements and organizational constraints in work settings, where people are actors in situations, with a set of skills and shared practices based on their experience of working with others” (p.138). This implies that the chosen perspective can bring a meaningful sense that can lead to an activity-based solution design through the representation of the work domain. However, the requirement of the work domain can relatively be complex. For example, an enterprise resource planning system (ERP) needs to cover many business functions and thus uses holistic design approaches. In contrast, DSS design is more user-orientated, and the end-user’s role with respect to decision making is a vital aspect of the system design. The key question when focusing on a user’s work domain would be: how can we conceptualise the work domain related to a solution design (an artefact) under the focus of the target users and within their own context?

In designing the artefact we ensure that we have both top-level and lower-level controls on the design components. For example, end-users such as farmers are given a set of parameters to identify the most relevant situation for their business, while context-specific variables such as decision making rules used for setting options are given to experts. In this paper we refer to these people as “domain experts”. Our study promotes the idea (Outlined on the editorial note “The user-the great unknown of systems development: reasons, forms, challenges, experiences and intellectual contributions of user involvement”, Information Systems Journal, vol. 20, p 109-117) of “Users usually are the best experts on the local work practices to be aligned with and to be supported by a system. Users also are the final ‘implementers’ of the system, and evaluation of the system without any attention to subjective user-oriented criteria” (p.111). Similar conceptual issues in user participation and designer accountability in IS design process are also identified by Koh and Heng (1996). However, the integrity of our design artefact assures that the architecture’s controls, through a rule-based model and an inbuilt ontology repository for specific DSS applications design, are kept at the domain expert level. Thus end-users can only modify the system in relation to their own objectives and within their own problem domain. Both the design environment architecture and the ontological technique promise advantages over traditional DSS design approaches. With respect to the advantages, this research therefore, investigates an innovative design based on UCD method, in which an appropriate DSS application can be generated by the end users as ‘final implementers’ within their own context.

2.2 DSS solution issues

Knowledge-based and intelligent systems have been increasingly included as an addition to the traditional model-based DSS. This knowledge-based component typically involves knowledge representations (classically if–then rules) and system architecture considerations using rapid development tools (such as expert system shells). Acquiring the domain knowledge and formalising it for computation were often difficult, and inherently beset by (now) well-recognised problems including:

- missing concepts or relationships during the knowledge acquisition process
- changing priorities or contextual relationships during the development process
- a lack of interpretive nuance or adequate learning from experience
- forcing knowledge into possibly alien formalisms and
- making autocratic conclusions about predetermined goal variables

(Arnott, 2006; McCown, 2002; Cox, 1996)

Such limitations required users to make further judgements about the applicability of recommendations, and in some cases, systems were not aligned with requirements, resulting in a need to re-engineer the whole system. In addition, in cases where the end-user was not directly involved in knowledge acquisition, the problem-solving models and the terminologies used may not have mapped to the target user's vocabulary or understanding of the problem domain, leading to inappropriate applications and even non-adoption (Qin and Paling, 2001; Kerr and Winklhofer, 2005; Kerr, 2004).

Development of DSS applications by agricultural end-users is often not straightforward and there are risks associated with a potential lack of completeness in the identification and definition of the problem. Moreover, particularly in more complex problems, there is the danger of conceptual mismatches between developers and end-users (Janvrin and Morrison, 2000). For example, although probabilistic reasoning values, first-order logic and backward chaining may be useful, they may also be alien to an end-user's thinking (Wagner, 2000). Ineffective transfer of scientific knowledge and information sharing can also be problematic, especially where changes in industry regulation, markets and climate science affect existing knowledge models.

3. Method

Iivari (2007) argued that the IS artefact in design science can be built from IT meta-artefacts that supports concrete IS application development. Of the four UCD design methods outlined by Iivari and Iivari (2011), only one was deemed relevant to our DSS artefact design, namely "Goal-Directed Interaction Design", this is where the solution design for providing user provision was based on the definition of behaviour, functions and information needs for the system. This was considered applicable because our aim is to focus on the target user's decision making and a solution artefact construction based on empirical data. Furthermore we focus on decision making behaviour and needs that are in common with organisational decision making practices. Therefore our artefact design looks at how the system should behave and what the system looks like. These two aspects are highly design oriented, and consequently we have used the design science paradigm in our research.

Design science has attracted increasing attention to IS researchers in recent years. Baskerville (2008) suggested that, more than a methodology for developing design artefacts, it is an approach that enables researchers to create system artefacts. It not only provides solutions for identified organisational problems but also provides a new dimension in designing solutions for these problems (Baskerville, 2008; Hevner, 2008). As mentioned earlier our research aims to acquire knowledge of an identified problem domain in agricultural industry with the objective of improving the DSS development method to produce an innovative DSS artefact to support informed decision making. This will employ the UCD design method outlined earlier. The design method described by Hevner et al. (2004), as well as the concepts discussed by Iivari and Iivari (2011) guide our artefact design process.

Hevner's et al. (2004) seven guidelines (Table 2) provide supportive, (not purely prescriptive) insights for defining the problem space, outlining design, implementing design and evaluating the design artefact for the proper communication of research in a more human-centred way. Design science research provides further clarity for designing and constructing artefacts in social or natural settings. Next we describe how the design guidance of Hevner et al. (2004) relates to the present research.

The following section gives detailed background of the artefact design.

Guidelines of Design Research	Relevance within our artefact design research
Guideline 1: Design as an Artefact: Design-science research must produce a viable artefact in the form of a construct, a model, a method, or an instantiation.	An innovative artefact (software solution prototype) is to be developed, field-tested and specified as a replicable model.
Guideline 2: Problem Relevance: Design-science research aims to develop technology-based solutions to important business problems.	A real problem domain is identified that supports the outlined software solution prototype. The problems addressed are business-critical. Their general form is demonstrated through similar problems in other businesses.
Guideline 3: Design Evaluation: Utility, quality and efficacy of a design artefact must be rigorously demonstrated via well-executed evaluation methods.	A descriptive evaluation method is employed for prototype testing, both with industry users and other stakeholders, coupled with scenario analysis using secondary data.
Guideline 4: Research Contributions: Effective design-science research must provide clear and verifiable contributions in the areas of the design artefact, design foundations and/or design methodologies.	The models used for the decision outcomes within the artefact were developed by domain experts using practice-based knowledge. This knowledge has been used as a kernel to derive the decision outcomes by using constraint-based formulas. The development methodology of the prototype is explicitly specified, covering both established methods and other generically described and replicable techniques.
Guideline 5: Research Rigor: Design-science research relies upon the application of rigorous methods in the construction and evaluation of the design artefact.	Rigor is achieved through expert scrutiny of the developed models by peers within the problem domain and through the specification of the developed solution prototype, ensuring that the artefact is rigorously defined, coherent and internally consistent with industry requirements. Established development and testing techniques were used throughout.
Guideline 6: Design as a Search Process: The search for an effective artefact requires utilising available means to reach desired ends while satisfying laws in the problem environment.	The method of artefact is closely aligned to industry inputs and resources in use, enabling the solution to be constructed according to the problem space and within the constraints (economic, biological and other concerns) of the industry under consideration.
Guideline 7: Communication of Research: Design-science research must be presented effectively both to technology-oriented and management-oriented audiences.	This is achieved through system demonstrations and evaluations by target users and stakeholders within the case industry. The software prototype uses specific and general examples integrated with industry practice. Both technical and business-relevant evaluation criteria are provided in documents for practitioners and industry experts.

Table 2: Seven guidelines for design science research (Hevner et al., 2004, p. 83)

4. Development of the design artefact

4.1 Practical problem context and relevant analysis

Decision making in agricultural industries, particularly livestock-based businesses in Queensland, Australia, has been faced with rapid changes due to the effect of climate change, government regulations and changes in farming methods (Chataway, Walker and Callow, 2010), resulting in the disuse of many DSS applications (Kerr,2004). This disuse was due to the lack of fit to the needs of decision makers' contextual variables. In the agricultural context, the expectation of farmer use and intervention to their farm management practices have not been realised in DSS design (McCown, 2002).

The DPI development method outlined a hierarchy of approval processes designed to ensure appropriate resource allocation. For example, the role of the manager was considered important in order to monitor the domain experts and to establish the relevance of their activities in relation to DSS application development for the specific farmers' groups and within the political and knowledge content realities of their management context. On the other hand, end users such as farmers require their system to be tailored to their business context and current contingences, specific to their farming inputs and decision-making factors. For example, the business context varies from farm to farm and in different regions, as well as having to deal with seasonal differences. To address these changing needs, a generic method/platform that can, support farm managers with resource allocation and the organisation of knowledge for DSS design and be adjustable and applied to any agricultural business context (i.e. beef cattle, cotton and sugar industries) is of importance. The system also needs to provide adequate decision support for the decision makers and/or farm operators.

Current DSS technologies do not match these requirements in which both top-down and bottom-up approaches are required in a platform to reconcile, develop, tailor and utilise DSS applications within the industry context.

Our case study is in the domain of dairy farming in Queensland, Australia. This industry changed radically following deregulation. As consolidation occurred in the industry new business models and supply arrangements were required. Market forces caused new demands for differentiated levels of milk protein, and to remain competitive, dairy farmers had to supply milk with protein content at levels set by external markets. In dairy businesses, protein level is a function of various identified manageable factors such as diet and breed, with combinations of parameter values leading to different potential levels of milk protein. Farmers must make decisions based on a combination of these, adjusted for local conditions such as ambient temperature, water and feed availability.

4.2 Solution principles

We have utilised the skeleton of IS design theory (Gregor and Jones, 2007) that seems to be useful to deal with both the process and the artefact design. The specification of skeleton helps us to summarise the detail of the purpose and functionalities of the DSS artefact we intend to design in this study.

In this study	***The specification of the design theory***
The introduction section describes the motivation of the study. Both practical and theoretical needs of better DSS artefact design are identified to address user centric DSS design. In study	**The purpose and scope of the artefact are addressed**

background section, the paper explains why this need is significant in design science.	
For using extracted knowledge in system, rule-based method is used in which constraint-based formalisms provide a expressive underlying generic knowledge representation that specifies how factors, parameters and specific values relate to and affect one another. The description can be seen in this section below.	**Principles of form and function incorporating underlying methods are described**
The argument is made that the artefact used UCD method incorporating ontology and design environment techniques allows for relatively simple tailorability for end-users. It is also argued that the DSS artefact can be re-used in other similar problem domain. In other words, generalisability of the developed artefact is defined and evaluated throughout this study (See below in this section).	**Artefact mutability is addressed**
The argument is made that the features of the DSS artefact have worked in different decision making problem space that is described in the Artefact evaluation section (Section 5).	**Testable proposition of the design artefact is defined**
The study is shown how the developed DSS artefact works; by reference to underlying tailorable design theory and also supporting user roles in three different layers (see sub section 4.3).	**Justificatory knowledge (kernel theory)is provided**
In the subsequent section (Section 4.3), guidelines are given on how to implement the artefact through a systematic procedure.	**Principles of implementation are given**
An illustration of working DSS artefact (as system prototype) is provided during the evaluation phase (Section 5).	**An expository instantiation is given**

Table 3: The details of the DSS artefact design

Many of the disuse of DSS are due to poor design, lack of shareholder involvement and poor implementation and planning (Arnott and Dodson, 2008). We design the DSS solution artefact based on the practical needs of the end-user within the industry context. This artefact to DSS development, although enacted in a specific case context, was kept consciously generic to avoid confounding it with domain properties and to allow replication for DSS development in other industries with problems characterised by constraints and changing parameters for decisions.

As our target system is related to a personal DSS, the language used for knowledge representation must be both familiar to the end-user and consistent with industry terminology. In agricultural industries, as underlying scientific knowledge or new market information becomes available, a facility to incorporate this immediately into local decision making without extensive re-engineering is required. In addition, whilst rule-based knowledge representations can explain decision rationale, they are not the only types of association among variables, and their inherently directional (antecedent, consequent) structure is inflexible when the goal variable of interest changes. Our target design needs to cater for these changes.

As mentioned, it was important that we apply a user-centred method. This is implemented through use of definitive terminology from the industry literature, by representing these in ontology, and verified in a focus group context with both industry experts and end-user representatives. The acquired domain knowledge components that enable reasoning (i.e. parameters, factors and their relations and constraints) specify a generic knowledge model for building a particular DSS. Such a structure of the DSS artefact can be re-used to build DSS applications in other knowledge domains, since the knowledge is not functionally bound into the architecture. As ontologies are domain specific, experts should be involved in interpreting and defining the domain knowledge before any actual development occurs at the end-user level.

Constraint-based formalisms provide a powerfully expressive underlying generic knowledge representation that specifies how factors, parameters and specific values relate to and affect one another (Jaffar and Lassez, 1987; Leler, 1988). Constraint languages subsume logic programming languages and are both semantically well-grounded and more intuitive to use (Jaffar and Lassez, 1987). By expressing domain associations as constraints, specific rules can be generated for any domain variable given a set of local values, and as new influences become relevant, constraints can be added (or removed) from the domain model. The design does not require the end-user to parameterise everything: domain model building is done by domain experts in conjunction with end-users, as we describe below.

4.3 Proposed user centred design environment (UCDE) as design artefact

The component design implemented in the proposed cutting-edge artefact allows for tailorability at different levels. Whilst scientifically informed domain models will be built by industry and/or government domain experts, the choice and focus of these is a policy matter, and their use and customisation is an end-user matter. The UCDE thus recognises different classes of user as defined by specific industry requirements and the relevant managerial responsibility. The example domain incorporates three functional areas of authority (layers). The first (layer at left-hand side in Figure 1) is an authorisation layer which allows line managers who allocate resources to assign one or more domain experts to specific DSS application development. The second layer (middle) allows access to the knowledge acquisition component of the system where the domain expert(s) will develop decision-making rules using knowledge acquired for the problem domain. The final layer (right-hand side) allows end-user access to the system, thus enabling them to build decision support specific to their own business. These rule changes can relate to their level of risk-taking (for example, modifying the expected benefit from a scenario by reducing the amount of resources needed, only allowing resources that are relevant to their enterprise to be considered, and/or selecting a low, medium or high response based on their own evaluation of their individual circumstances).However, these changes do not override the constraints identified by domain experts: although tailorability can be achieved by adding or removing components through constraint relaxation or augmentation, this requires a model building authorisation – much like planning permission for a house extension. The administrative layer is required to allocate limited resources and to provide accountability in the development of projects. The last two layers, namely knowledge acquisition from the problem domain and business specific options for end-users, are essential to develop relevant DSS functionality.

Figure 1 shows the overall design artefact (the UCDE) in which the three functional processes form three user layers.

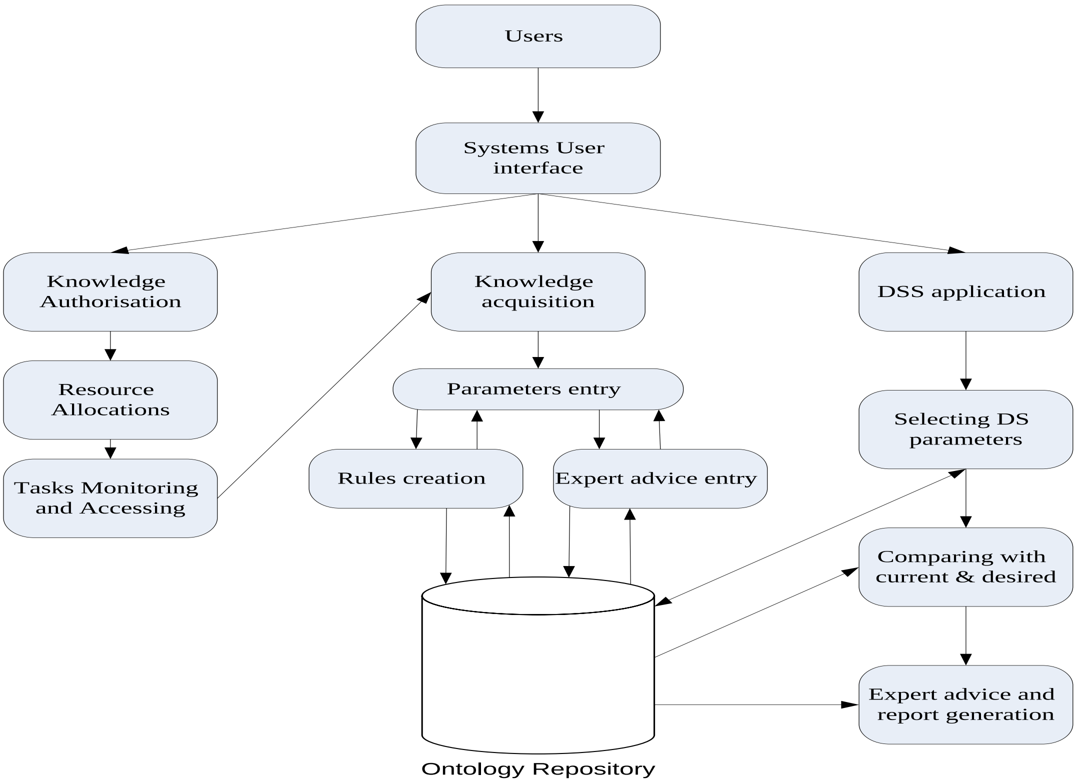

Figure 1: The overall architecture of the DSS artefact

In the UCDE, the primary design approach outlines the generic capability of the main solution architecture to accommodate domain knowledge independently, in terms of useful components for situation-specific building of applications by the end-user. The secondary design function is for end-user interaction for their specific application development. In other words, the primary design architecture (generic features) recognises end-users in the creation or re-creation of specific application through the secondary design function. The artefact is informed through a tailorable design theory (Germonprez, Hovorka and Collopy, 2007), in that the technology contains dynamic, recognisable components and conventions for enabling users to tailor IS features. This theory can offer user-customisable features so that they can easily be adapted to a user's particular needs, activities or within their settings (Iivari and Iivari, 2011).

The proposed artefact allows specific application to be designed by end-users through the selection of relevant system components. In other words, the generic DSS artefact helps produce a specific artefact (at secondary level) using the design components. In this instance, the artefact (UCDE) remains in its original form (i.e. in its primary design state) for any tailoring action, as the end-users engage themselves in developing a one situational specific artefact. We found these two main functionalities are useful for handling user involvement and centredness issues within their work space, and this in turn can assist with DSS uptake. This secondary design and the steps used are shown in Table 4 below.

Design Steps	***Tasks and activities***
Decision-support parameters	End-users select the appropriate set of decision-support parameters to define their business-specific situations, e.g. scale, size and relevant climate or regional conditions prior to the specific DSS

	application design. End-users can go through this process again and again until they have satisfied the conditions set for their application design.
Compare with current and desired states	End-users provide the current inputs in their specific application for obtaining a comparative analysis for optimisation within the current situation.
Obtain expert analysis and report generation	End-users select and consider over and over for each and every aspect by varying their target goals or budgets for improvement. Depending on their tailored selection they can seek expert advice.

Table 4: Secondary design methodology in the proposed UCDE

In our targeted problem context, primary design provides functions for three key roles. Line managers can define the scope and allocate resources, whereas the domain expert converts the problem domain into system components that can be useful for general end-users. The end-users (farmers) can apply their own knowledge and understanding to build specific applications. Farmers can build as many applications as they need and store their developed applications and outcomes for further comparisons and analysis. The tailorable technology is defined by Germonprez et al. (2007) as enabling "*end users to select and integrate technology features in the ongoing creation and recreation of unique information systems that match their concerns and activities*" (pp. 352). The innovative aspect of design artefact is that it is capable of generating many DSS applications to suit the best need, and is re-usable in other agricultural business domains as the decision making aspects in the farming context are similar. For instance, decision making parameters vary season to season and farmers can add/remove the parameters in order to build their context-specific DSS application.

5. Artefact evaluation

Several IS artefact evaluation methods have been outlined by design science researchers, including: observation, analytics, experiments, testing, descriptive analysis, and action research (Baskerville and Myers, 2004; Hevner et al., 2004). Our evaluation strategy focussed on the descriptive (analysis) evaluation method for design science research (Hevner et al. 2004) as this is more appropriate for evaluating innovative design artefacts than other forms of evaluation. This is because the IS artefacts can be evaluated in terms of the selected evaluation metrics such as "functionality, completeness, consistency, accuracy, performance, reliability, usability, fit with the organization, and other relevant quality attributes" (Hevner et al. 2004; pp.85).The developed method as an innovative design artefact was outlined within the business environment of dairy farming through a case study of the milk protein enhancement problem. The method's functionality was originally conceived, then iteratively prototyped, refined and evaluated with industry decision makers. Simultaneously the UCDE artefact was used to build a simple expert system (for fleet car purchase decisions), and continually assessed to exclude any domain specific features from the developing architecture.

Analytically the artefact's components lend themselves to generic applications. The domain ontology can be replaced with another domain ontology without requiring redesign and this was tested first as a thought-experiment by asking both a beef-cattle farmer and a Professor specialising in this area to assess the method as a DSS generator for beef cattle applications. Neither could see why this could not happen. Secondly, a published data set in a cropping domain (Bell, Graham and Langford, 2007) was used to develop an ontology and generate a specific DSS. The artefact's design allowed this without requiring architectural change.

In design science literature Venable (2006) suggested that the evaluation of the developed artefacts should be done artificially before attempting to evaluate naturalistically. Iivari and Venable

(2009) re-assessed this idea in an action research context. In our research, although the UCDE prototype was intended only as a concept demonstrator, to be re-implemented to the industry's house style, a software evaluation was undertaken and presented to exemplary audiences. This demonstrated that, without change, the UCDE artefact could be used to model and reason with knowledge in another domain. Qualitative evaluations of the UCDE were undertaken through focus groups and interviews with four respondent types, namely student proxies for farmers and extension officers, extension officers, managers of research projects and farmers themselves. In addition, in-depth interviews with practitioners in the equivalent domain of beef cattle were used to indicate whether the system had a priori utility beyond the test case domain. An evaluation of the UCDE using published secondary data in a crop domain was used to show the generic utility for agricultural industries beyond livestock. Finally all results were presented to, and approved by, the senior industry manager responsible for part-funding of the research. A procedure for administering the evaluation was developed as shown below. Specific questionnaires used in Step 4 and 6 can be seen in Appendix B.

Step 1: Introduction to the project and its goals
Step 2: General information of the developed method given to all participants
Step 3: Prototype method is demonstrated by running industry relevant examples
Step 4: Participants are asked specific questions about the method and if there were areas they were unsure of
Step 5: A time interval is offered to the participants to use the method
Step 6: Questionnaires are given to the participants to capture their views
Step 7: Participants are requested to provide more information about their understanding and views of the method
Step 8: At the end of the workshop, the participants are thanked for their time and effort

5.1 Evaluation from proxy stakeholders

In order to triangulate findings and to obtain a complete picture of the usefulness of the UCDE prototype, students were chosen as proxies to reflect the typical education level of users. The students were from the information systems/information technology discipline with first year students (51) assessed as being typical of the average farmer. Postgraduate students (50) were classed as proxies for extension officers as they usually have a degree and are familiar with DSS development. The procedure above was used. Both groups rated the professional look of the system the lowest, however this was expected as it was still in prototype stage and was designed to demonstrate the basic functions rather than being a completed commercial product. 31 postgraduate and 27 undergraduate commented that the method was useful for DSS development and easy to understand. Remaining comments related to other aspects of the prototype such as its transferability to other problems and the practicality of the method.

5.2 Evaluation from Industry stakeholders

Three workshops (18 farmers) were conducted along with face-to-face evaluations with extension officers and policy makers. The same procedure as outlined above was used.

Farmer and extension officer participants were categorized into two groups, expert or novice according to their experience with DSS applications. Both expert and novice stakeholders rated the method highly with all scores rated 4 or above (see appendix A).

Table 5 provides an overview of comments about the method by farmers and extension officers.

Participants	***Comments***
Farmer 1	*"The systems … simplicity… obtaining extension officers feedback…. ……* *Feel system will be of good value and applicable to modern farmers……"*
Farmer 2	*"easy to use, simple to understand and user friendly, compatible to normal computer systems"* *"…very good, to follow prompts in the system and easy to understand. It has many applications. I am just thinking how I could use this system to assess goat diseases."*
Farmer 3	*"I think this system could be used for different farming methods and help with decision making. By looking at this system I could find many answers for results and estimating costs*"
Farmer 3	*"It is handy and useful for everyday use, farming can be improved in the dairy goat industry & making production better"* *"information can be passed through the system by the DPI and can be used by us the farmers"*
Farmer 6	*"Simple means of organising thoughts into a logical framework …. Ability to modify, and suited to changing environment in addressing specific issues on a industrial farm*"
Extension officer	*"The system seems overall simple and straightforward in data entry to me, however, it needs to incorporate the biological settings to improve the ability of the system which could be done by a knowledgeable user".* (Research diary: 15 February 2008)
DPI management	*"This is a nice little piece of software where we may control the decision support tools development activities which are very important from the management point of view*".(Research diary: 22 August, 2007)

Table 5: Farmers, Extension officers and DPI management's comments on the UCDE artefact.

6. Limitations of the method

The study has a number of limitations. Firstly, it was developed in a case study context, and whilst some other indicative domains were assessed to evaluate the design's generic qualities, these remain to be fully tested in real business contexts. Secondly, it is assumed that the domain decisions can be parameterised and represented using a constraint programming language. Whilst the class of constraint problems is large, care must be taken to ensure the domain is effectively scoped. Thirdly, the UCDE method has three levels of access control, argued for on principle, but not experimented with, nor directly valued. It may be desirable to allow model building at local levels, using other parameters reflecting local decision making considerations, and monitor those to see if more general learning can occur. This, along with the analytical evaluation of the proposed artefact once implemented within the industry- operating environment remain issues for further research.

7. Discussion

The aim of the paper was to describe a DSS design artefact as a new method to DSS development. Through the application of design science knowledge the developed artefact was based on underlying principles of UCD. The practical functionalities of design environment and ontology were applied to operationalise the UCD principles. The ontology was used for effective knowledge-base construction and to improve vocabulary within the problem context (in terms of knowledge sharing between relevant stakeholders). The design environment was provided functionality to support flexible and tailorable options for end users. It is how the UCD method goes beyond the purpose of adding rules to reconfigure the DSS artefact within the problem context. In this end users can apply their subjective judgement to reconfigure decision support rules from their own practice based knowledge. This will help reduce conceptual mismatches and increase dynamicity in decision making. This artefact's phenomenon of tailoring in a practical industry context was informed through the tailorable design theory, as it was discussed earlier. Through this study a broader practice-based view of artefact design was promoted in the design science research. Such new and innovative artefact design created new reality, rather than explaining existing decision support reality or helping to make sense of it. Beyond the descriptive evaluation of the artefact reported in section 5, the proposed artefact has been theoretically verified within the design science specification (defined in table 3).

Winter (2008) suggested that the typology of artefacts to specific problem solutions are not yet sufficiently standardized to the IS design research community. For DSS design, traditional DSS development methods have several limitations in supporting businesses, including conceptual mismatches, static models and inflexibility. This has resulted in poor uptake or disuse (Cox 1996; Kerr and Winklhofer 2005). To address these problems, the proposed method of design artefact, namely called a user-centred design environment (UCDE) provided an innovative way for generating appropriate DSS applications in a context sensitive manner. The example shown in this paper used a straightforward rule-based method, as that was considered most relevant to our industry context.

The presented research was based on a doctoral thesis by the first author of the paper (Miah, 2008). We described the new artefact creation by identifying its phenomenon of tailoring in a practical context of use for target decision makers. This research conceptually contributed to design science literature in relation to construction of complex artefacts that has promises in addressing decision maker's ultimate problems in an agricultural aspect. The proposed understanding of the artefact design can also be reused for creating similar artefact. This research also contributed to a new DSS development method informed through a user-centred theory (Iivari and Iivari 2011), the work also sits within the work activity-based reality concept described by (Norman, 2005). We argue that the collective IS artefact as a solution methodology that "extend(s) the boundaries of human problem solving and organisational capabilities by providing intellectual as well as computational tools" (Hevner et al., 2004, p. 76). This move towards the incorporation of both the user-centred method and design science has not previously been done within the agricultural context, and it is expected to improve DSS outcomes for agricultural industries. We also expect it to address concerns expressed by Cox (1996) and Hayman and Easdown (2002) through a more robust and dynamic method that relates to specific information systems theories rather than solely on domain knowledge and off-the-shelf expert system shells. Such new artefact design can be considered as cutting-edge design, as Venable (2006) acknowledged that a "Solution Technology Invention" is the core of design science research.

The aim of good design should include generalisability of the artefacts and the utility of the design artefacts in other problem contexts (Venable, 2006). With respect to this, our work has also added to design science theory by creating a new generic artefact within the DSS context, and has indicated how it generalises beyond its immediate development case context to be of wider value. This will be of particular value to research funding bodies, as it will reduce the duplication of efforts and costs across industries. The UCDE method has improved the context and relevance of DSS development as it uses a flexible model where data and decision making priorities can be changed easily in their context of use. For example, as this generic artefact is transparent to management, domain experts and end-users through the three layers of access and input, it will assist agricultural DSS acceptance by overcoming a significant inhibitor described by Cox (1996), namely, the concern about the DSS being a 'black box' in that the inner workings and logic are not transparent to end-users.

Our proposed method advances previous design environment-based solutions by explicitly allowing end-users to incorporate their own factors into application development in a more general way than previous software components environments permitted. Simultaneously, this research extends ontological development into the agricultural DSS application domain. The solution goes beyond either simple expert systems architecture or an uncontrolled end-user approach, and both the processes for development of domain ontology and its specification within a larger architecture have been detailed at a generic level. In addition, the proposed artefact offers new features over the traditional DSS technologies for solving known issues such as systems rigidity, end-user subjectivity in the context of use, obsolescence, intermediary requirements and differences in problem solving approaches between end-users and designers.

The developed UCDE provides transparency, updatability and interoperability compared to the traditional solution methods in agricultural businesses, as well as providing customisable options in building industry-specific applications by easily adjusting to changing problem situations. The UCDE enables end-users to apply locally specific and contextual knowledge using their subjective judgement and specific business goals and both this, and other aspects were positively evaluated using focus groups method which is justified for artefact refinement and evaluation in design science research paradigm (Tremblay, Hevner, Berndt, 2010).

8. Conclusions and future research

This study described a design science research to address DSS development issues. The design science knowledge to artefact design has contributed to the DSS literature within the agricultural industry context, and this has the potential to overcome many of the problems of the classic DSS method as outlined Cox (1996). These problems were significant to agricultural industries and resulted in a marked reduction in the development of DSS, despite the clearly articulated advantages of DSS development for decision makers in this domain (Kerr, 2004;McCown, 2002). The collective DSS artefact outlined here has the potential to improve this situation and may result in a resurgence of DSS development projects in agricultural industries.

As mentioned the primary motivation of this study was to design a collective artefact by identifying its phenomenon of tailoring in a practical context of use. We found that an UCD based design science principle may have application to such artefact construction. In relation to uptake of DSS the proposed UCD based artefact has the potential for increasing user uptake in an industry that has had relevancy problems with past DSS implementation and has experienced associated poor uptake. In recent years, funding bodies have been reluctant to commit funds to DSS development in the agricultural sector due to failed projects. It is hoped that the proposed UCDE will help convince funding body decision makers of the advantages of this flexible, generic method to DSS development. Based on the discussion throughout, we argue that this study may offer a unique contribution to design science knowledge applied to the area of DSS development research.

This study raised some interesting and relevant areas for future research in designing DSS artefact for unstructured or semi-structured decision support issues. One potential area is to explore the principles of interplaying design roles in order to outline an appropriate boundary for activity centredness for each relevant end user. This would help shape how end users could be more responsible for major activities in their own application development and in turn this could add value to the current topology of the collective DSS artefact design. It may be that in a new business context the typology of the collective DSS artefact would be quite different. In such cases, the DSS artefact should have different functionalities in order to address new problems. It would also be significant to continue this research to establish trends in DSS artefact design in IS by undertaking longitudinal research.

Acknowledgement

The authors would like to acknowledge the Australian Research Council for funding this research and the following individuals; Professor Tom Cowan for dairy research knowledge; Professor John Gammack for earlier work in this research and the initial conceptual approach; Dr. Kay Bryant for mentoring assistance; and first author's mother who has been provided huge inspiration throughout this project.

References

Arnott, D. (2006),"Cognitive biases and decision support systems development: a design science approach", *Information Systems Journal*, Vol. 16(1), pp. 55-78.

Arnott, D. and Dodson, G. (2008), "Decision support systems failure", In Burstein, F. and Holsapple, C. W. (Eds.), *Handbook on decision support systems* (Vol. 1, pp. 763-790), Berlin, Springer.

Arnott, D. and Pervan, G. (2008),"Eight key issues for the decision support systems discipline", *Decision Support Systems*, Vol. 44, pp. 657-672.

Arnott, D. and Pervan, G. (2012),"Design Science in Decision Support Systems Research: An Assessment using the Hevner, March, Park, and Ram Guidelines", *Journal of the Association for Information Systems,* Vol. 13 (11), pp. 923-949.

Baskerville, R. and Myers, M.D. (2004),"Special Issue on Action Research in Information Systems: Making Is Research Relevant To Practice-Foreword", *MIS Quarterly,*Vol. 28 (3), pp. 329.

Baskerville, R. (2008),"What Design Is Not", in Hart, D. N. and Gregor,S. D. (Eds), The Keynote, *The 4th Biennial ANU Workshop-Information Systems Foundations: The role of design science,* October 2-3, 2008Canberra, Australia.

Bell, A., Graham, P., and Langford, C. (2007), "How pasture characteristics influence sheep production, Profitable and sustainable primary industries", Technical paper, New South Wales DPI, Australia, (URL: http://www.dpi.nsw.gov.au/__data/assets/pdf_file/0004/144517/how-pasture-characteristics¬influence-sheep-production.pdf, Accessed on April20, 2008).

Blum, B. (1996),*Beyond programming: To a New Era of Design*, Oxford University Press, New York, NY.

Carlsson, S.A. (2006),"Design Science Research in Information Systems", 6-8 Dec, Adelaide, Australia. (URL: http://www.ejbrm.com/vol3/v3-i2/v3-i2-art1-carlsson.pdf, Accessed on May 23, 2008).

Carlsson, S.A. (2007),"Developing Knowledge Through IS Design Science Research: For Whom, What Type of Knowledge, and How", *Scandinavian Journal of Information Systems*, Vol. 9(2), pp.75-86.

Chataway, R.G., Walker, R.G. and Callow, M.N. (2010),"Development of profitable milk production systems for northern Australia: a field assessment of the productivity of five potential farming systems using farmlets", *Australian Journal of Experimental Agriculture*, Vol. 50, pp. 246–264.

Cox, P.G. (1996),“Some Issues in the Design of an Agricultural Decision Support Systems”, *Agricultural Systems*, Vol. 52, pp. 355-381.

Evermann, J. (2005),“Towards a cognitive foundation for knowledge representation”, *Information Systems Journal,*Vol. 15, pp. 147-178.

Fischer, G. (1999),“Domain-Oriented Design Environments: Supporting Individual and Social Creativity", in J. Gero and M.L. Maher (Eds), *Computational Models of Creative Design IV*, Key Centre of Design Computing and Cognition, Sydney, Australia, pp. 83-111.

Gammack, J.G. (1999),“Constructive Design environments: Implementing End-users Systems Development”, *Journal of End User Computing*, Vol. 11, pp.15-23.

Gennari, J.H., Musen, M.M., Fergerson, R.W., Grosso, W.E., Crubezy, M., Eriksson, H., Noy, N.F.andTu, S.W. (2003),“The evaluation of Protégé: An Environment for Knowledge Based systems development”, *International Journal of Human-Computer Studies*, Vol. 58, pp. 89-123.

Germonprez, M., Hovorka, D. and Collopy, F. (2007),“A theory of Tailorable Technology Design”, *Journal of the Association for Information Systems*, Vol. 8, pp. 351–367.

Gregor, S. and Jones, D. (2007), The anatomy of a design theory, *Journal of the Association for Information Systems,* Vol. 8, pp. 321–335.

Haghighi, P. D., Burstein, F., Zaslavsky, A. and Arbon, P. (2013),“Development and evaluation of ontology for intelligent decision support in medical emergency management for mass gatherings”, *Decision Support Systems*, Vol. 54, pp. 1192–1204.

Hayman, P.T. and Easdown, W.J. (2002),“An ecology of a DSS: reflections on managing wheat crops in the North-eastern Australian grains region with WHEATMAN”, *Agricultural Systems*, Vol. 74, pp.57-77.

Hevner, A. (2008),“Design Science Research in Information Systems: Try-cycles and Hula Hoops”, Key note,in Hart, D. N. and Gregor, S. D. (Eds), *The 4th Biennial ANU Workshop on Information Systems Foundations, Information Systems Workshop-Information Systems Foundations: The role of design science,October 2-3,2008,* Australian National University, Canberra, Australia.

Hevner, A. (2007), “A Three Cycle View of Design Science Research”, *Scandinavian Journal of Information Systems*, Vol. 19 (2), pp. 87-92.

Hevner, A., March, S., Park, J. and Ram, S. (2004),“Design Science in Information Systems Research”, *MIS Quarterly*, Vol. 28, pp. 75-105.

Hosack, B., Hall, D., Paradice, D. and Courtney, J. F. (2012),“A Look Toward the Future: Decision Support Systems Research is Alive and Well”, *Journal of the Association for Information Systems*, Vol. 13, pp.315-340.

Iivari, J. (2007),“A paradigmatic analysis of Information Systems as a design science”,*Scandinavian Journal of Information Systems*, Vol. 19, pp. 39–64.

Iivari, J. and Iivari, N. (2011),“Varieties of user-centredness: an analysis of four systems development methods”, *Information Systems Journal*, Vol. 21, pp. 125-153.

Iivari, J. and Venable, J. (2009),“Action Research and Design Science Research-Seemingly Similar but Decisively Dissimilar”, in Newell, S., Whitley, E., Pouloudi, N., Wareham, J.,Mathiassen, L. (Eds), *Proceedings of the 17th European Conference on Information Systems*, June 8-10, Verona, Italy, paper 73.

Jaffar, J. and Lassez, J. (1987),“Constraint logic programming”, paper presented at*the 14th ACM SIGACT-SIGPLAN Symposium on Principles of Programming Languages (Munich, West Germany, January 21 - 23, 1987). POPL '87.ACM, New York*, NY, pp.111-119.

Janvrin, D. and Morrison, J. (2000),“Using a structured design approach to reduce risks in end user spreadsheet development”, *Information & Management*, Vol. 37, pp.1-12.

Kane, G.C. and Fichman, R.G. (2009),“The Shoemaker's Children: Using Wikis for IS Teaching, Research, and Publication”, *MIS Quarterly*, Vol. 33 (1), pp.1-22.

Kerr, D.V. (2004),“Factors Influencing the development and adoption of knowledge based decision support systems for small, owner operated rural businesses”,*Artificial Intelligence Review*, Vol. 22(2), pp. 127-147.

Kerr, D.V. and Winklhofer, H. (2005),“The effect of rapid rural industry changes on the development of a decision support system for dairy farmers in Australia”, *Computers and Electronics in Agriculture*, Vol. 50, pp. 61-69.

Koh, I.S.Y. and Heng, M.S.H. (1996),“Users and designers as partners – design method and tools for user participation and designer accountability within the design process”, *Information Systems Journal*, Vol. 6, pp. 283-300.

Leler, W. (1988),*Constraint programming languages: their Specification and Generation*, Addison-Wesley Longman Publishing Co. Inc.

Mackrell, D., Kerr, D.V. and von Hellens, L. (2009),“A qualitative case study of the adoption and use of an agricultural decision support system in the Australian cotton industry: the socio-technical view”, *Decision Support Systems*, Vol. 47, pp. 143–153.

March, S.T. and Smith, G. (1995),“Design and natural science research on information technology”, *Decision Support Systems*, Vol. 15(4), pp. 251-266.

McCown, R.L. (2002),“Changing systems for supporting farmer’s decisions: problems, paradigms, and prospects”, *Agricultural Systems*, Vol. 74, pp. 179 -220.

Miah, S.J. (2008), “An ontology based design environment for rural decision support”, *Unpublished PhD Thesis*, Griffith University, Brisbane, Australia.

Muntermann, J. (2009),“Towards ubiquitous information supply for individual investors: a decision support system design”, *Decision Support Systems*, Vol. 47, pp. 82–92.

Niehaves, B. (2007),“On epistemological diversity in design science – new vistas for a design-oriented IS research”, in *Proceedings of the 28th International Conference on Information Systems, Montreal, Canada,9-12 Dec 2007*, pp. 1-13.

Norman, D.A. (2005), Human-centered design considered harmful. *Interactions*, Vol. 12, pp. 14–19

Purao, S. (2012),“Design Research in the Technology of Information Systems: Truth or Dare”, School of Information Science and Technology, The Pennsylvania State University, (URL: http://iris.nyit.edu/~kkhoo/Spring2008/Topics/DS/000DesignSc_TechISResearch-2002.pdf, Accessed on November 12, 2012)

Purao, S. and Storey, V.C. (2008),“Evaluating the adoption potential of design science efforts: the case of APSARA”, *Decision Support Systems*, Vol. 44, pp. 369–381.

Qin, J. and Paling, S. (2001),“Converting a controlled vocabulary into an ontology: the case of GEM”, *Information Research*, Vol. 6 (2), pp.1-11. (URL: http://InformationR.net/ir/6-2/paper94.html, Accessed on May 20, 2012).

Ren, Y., Kraut, R. and Kiesler, S. (2007), "Applying Common Identity and Bond Theory to Design of Online Communities",*Organizational Studies*, Vol. 28(3), pp. 377-408.

Tremblay, M. C., Hevner, A. and Berndt, D. J. (2010),"Focus Groups for Artifact Refinement and Evaluation in Design Research", *Communications of the Association for Information Systems*, Vol. 26, pp. 599-618.

Venable, J.R. and Travis J. (1999),"Using a Group Support System for the Distributed Application of Soft Systems Methodology", in Yoong, Pak and Hope, Beverley, (Eds), *Proceedings of the 10th Australasian Conference in Information Systems, Wellington, New Zealand,* 1-3 December, pp. 1105-1117.

Venable, J.R. (2006),"The Role of Theory and Theorising in Design Science Research", in Chatterjee, S. and Hevner, A. (Eds), *First International Conference on Design Science Research in Information Systems and Technology, Feb 24 2006. Claremont, CA*: Claremont Graduate University.

Wagner, C. (2000),"End Users as Expert System Developers?",*Journal of End User Computing*, Vol. 12, pp. 3-13.

Winograd, T. (1995),"From Programming Environments to Environments for Design", *Communication of ACM*, Vol. 38, pp. 65-74.

Winter, R. (2008),"Design science research in Europe", Guest editorial, *European Journal of Information Systems*, Vol. 17, pp. 470–475.

Appendix A – Score rating by farmers and extension officers

Items for system effectiveness and its applicability (1-very poor, 5-excellent)	***Average rating of expert farmers***	***Average rating of novice farmers***
The system overall	4.67	4.33
Simplicity of the system navigation	4.67	4.33
Easy to add/remove parameters	4.67	4.00
The system offers a generally useful way of building decision support applications	4.00	4.67
Easy to build a new decision support system	4.33	4.33
It is a generic model for building DSS tools	4.67	4.00
Farmers can benefit from the system	5.00	4.67
Extension professionals/experts can benefit from the system	4.33	4.67
Transferring the expert's knowledge to general users	4.00	4.67
Relatively simple and straightforward to use	4.33	4.33
Does not ask too many questions and does not require too much information	4.00	4.33
Whole system is very easy to understand	4.67	4.67

Appendix B- Questionnaire used in the system evaluation

Questionnaire of Step 4

1. Do you think the methods can be workable to your business, subject to change its values or rangesin scale?
2. Do you think the methods used in the decision support are accurate and adequate to your business?
3. Would you suggest adding any parameters to decision support in the current methods?
4. Are there any areas you are unsure of the used methods?
5. In your business context, are there any methods to add specific to any relevant support?

Questionnaire of Step 6

A. The system overall to you? (1-very poor, 5- excellent)

B. Simplicity of the system navigation? (1-very hard, 5-very easy)

C. How useful do you think the system for adding/removing decision making parameters? (1-difficult, obscure, 5- easy, obvious)

D. The system offers a generally useful way of building decision support applications (1-strongly disagree, 5-strongly agree)

E. It was easy to build a new decision support system (1- strongly disagree, 5-strongly agree)

F. Do you think the system can be used for other rural application developments (e.g. beef, sheep….)?

G. This system can be used as generic model for building DSS tools (1- strongly disagree, 5-strongly agree)

H. Rural industry users such as farmers can benefit from the system (1- strongly disagree, 5-strongly agree)

I. The system can be helpful for extension professionals/experts (1- strongly disagree, 5-strongly agree)

J. This system can transfer the expert's knowledge to general users (1- strongly disagree, 5-strongly agree)

K. This system is relatively simple and straightforward to use (1- strongly disagree, 5-strongly agree)

L. This system does not ask too many questions and does not require too much information to operate (1- strongly disagree, 5-strongly agree)

M. I found the whole system is very easy to understand (1- strongly disagree, 5-strongly agree)

For specific suggestions on issues

A. Can decision making for rural businesses be improved using this system?
B. What do you think are the strengths of this system?
C. What do you think are the weaknesses of this system?
D. Do you have any suggestions for improving the system?
E. Do you have any other comments on aspects of the system overall?