Exploring the Role that SDMs can play in Influencing the Business Client – Systems Develper Relationship: an Institutional Theory Perspective

Bruce Rowlands
School of ICT
Griffith University
Brisbane, AUSTRALIA

b.rowlands@griffith.edu.au

Abstract. This paper reports on research into how two stakeholder groups - systems developers and the business client enact a systems development methodology (SDM). Our focus is on understanding how enactment is constrained by everyday social and organisational structures. The study develops a conceptual framework informed by institutional theory that integrates elements of Lamb and Kling's social actor model, and Scott's 3-pillars framework, concentrating on the relationships among systems developers, the business client, the SDM, and the context surrounding its use. The framework demonstrates that institutional structures such as authority, norms, symbolic values, and routines embedded within the methodology are active forces in the systems development process. In terms of theoretical contribution, the user as social actor model enables us to draw out the power concepts, and institutional theory explains how the SDM acts as a carrier of power, but not why developers are compliant with an unequal power arrangement.

1 Introduction

This paper reports on field research into the relations between developers and the business client and explores the role that systems development methodologies (SDMs) can play in influencing this relationship. In this paper, we argue that methods of systems development encode organisational values in the form of institutional structures; and that the method enables the business client to exercise power of developers. By examining how one methodology lays out the relationship between the business client and systems developers, how it distributes resources, how systems developers respond to its prescriptions, we gain insight into how structures embedded within the SDM constrain the actions of developers.

By structures, we refer to authority, norms and symbolic values embedded in the SDM that constitute the background condition for action enforcing constraints, giving direction, meaning, and setting the range of opportunities for undertaking action. This argument is advanced through an analysis of stake-holder relations in a large organisation in the financial sector (*The Bank* – a pseudonym) with an internal software development division. In *The Bank* it will be illustrated how institutional structures such as symbolic systems (rituals of development and widely held beliefs of developers), relational systems (based on authority relations, funding and the stage gate approval process), routines (the habitual use of the SDM as part of work culture), and the artefact itself (as a mandate) provides an overarching, framing context within which systems developers often made constrained choices about methodology use.

This research is distinct and important for the following reason. We believe that researchers still have an incomplete understanding of how systems developers collectively use methodologies in their day-to-day work, or the forces that impact on the situated use of these devices for systems development. We argue that the work that has been carried out is limited in its ability to consider the complex social and organisational **context** of methodology enactment. We believe that more theory-building research is needed as there has been insufficient consideration of the relations that exist in the client-developer relationship, especially when the client and developer come from the same organisation. Researchers have long called for research on methodologies in real life organisational situations (Beath and Orlikowski, 1994; Wynekoop and Russo, 1997) and they continue to do so (Kautz et al, 2007). With this call in mind, this paper addresses the following research question: how does an SDM influence the relationship between the business client and the systems developer?

Insight with respect to this question is derived from a case study of the deployment of an in-house developed methodology in a large IT department of a major Australian bank. The case studies an IT department operating an architecture of centralised mainframes, primarily custom software developed in-house, with the deployment of a mandated SDM that requires the services of professional systems developers. To develop, customise or maintain these software systems, the IT division has developed and documented an internal systems development methodology applicable for all development and maintenance tasks. The methodology, known as the *SDM* — a pseudonym, is a planning and document driven set of procedures based on traditional 'waterfall' lifecycle phases.

The paper is organised as follows. In the next section we review research relevant to the topic of method enactment. In the following, we explore the concept of method as an institution, and how an extrapolation of Lamb & Kling's (2003) social actor model and institutional theory can be used as an analytical research framework. The following section describes our research approach. The case findings are then presented and discussed illustrating how institutional structures such as authority, norms, symbolic values, and routines embedded within the methodology are active forces in the systems development process. In the final section, we conclude that the power relationship between developers and business clients is institutionalised through the enactment of the SDM and that it favours business clients in terms of control over the development process.

2 Review of Literature

A small, but growing body of research has been conducted on the specific topic of methodology enactment. Fitzgerald, Russo and Stolterman (2002:12) proposed a framework to understand the complex nature of systems development and how methods are enacted in practice. Their framework stressed the importance of understanding the myriad of factors, roles, and influences relevant to the systems development context and the effect they have on the development outcome. The pivotal component of their model, the *method-in-action* process, representing the enactment of the formalised method (while identified) is not described, and cannot be described in a general model as the context of each project, set of methods used, and the dynamics of the organisation are particular and unique. In the framework of Fitzgerald et al (2002, the role of the business client in the deployment of the SDM is ignored. Furthermore, it remains silent on any political role from the client's perspective.

In one of the first studies of the impact of organisational context involving methodology enactment, Nandhakumar and Avison (1999) highlighted various influences such as developers' knowledge about methodologies, implicit social norms, organisational form, and culture. Further studies advanced a growing argument that IT practice should be seen as more than a technical activity and as argued by Goulielmos (2004), method enactment can and should be understood as a complex social activity influenced by the organisational and institutional context in which it takes place. As noted in Aydin *et al* (2005) context played an important role in their study of the adaptation of an agile information systems development method — dynamic systems development method (DSDM). In a further contribution to the discussion about the enactment of SDMs, Madsen et al (2006) focussed on the unfolding development process, activities, and method elements that comprise this process. They describe the 'emergent method' as a process of social moderation of methodology use, covering both the change of methodology as formally prescribed through use, and the change of its users' actions that result from their interactions with the methodology. Madsen *et al* further conceptualised methodology enactment as a process of organisational innovation. Madsen *et al's* (2006) conceptualisation portrayed the role and usefulness of methodologies as a means for communication, coordination and (re)direction, rather than as a rigorous or rigid means for control. Huisman and Iivari (2006) studied the difference in perception between IS managers and developers about the deployment of SDMs, and found that both groups saw SDMs as a control technology in terms of keeping to deadlines and budget yet offered no discussion of how control was achieved. Managers perceived support for SDMs in terms of the production process, organisational effectiveness, and profitability (not surprising reflecting management's agenda). Systems developers, on the other-hand, were mainly concerned with the production of the final product in terms of system quality, goal achievement and individual reputation.

As an outcome of this brief review, we argue that SDM enactment needs to be understood in a wider institutional context comprising both social relations and social infrastructures in and outside the organisation. We conclude by arguing that the work that has been carried out is limited in its ability to consider the complex social and organisational context of methodology enactment. Most field research on the enactment of methodologies have neglected the messy and complex way people work and live, and the dynamics by which authority shapes SDM enactment. Inquiry into these institutional elements is a critical area of research for the field (Beath and Orlikowski, 1994; Silva, 2007).

Recognising that that few studies have directly addressed the enactment of methodologies in the context of authority between developers and the business client, attempts were made to seek a theoretical explanation within the organizational and information systems literature. We had a need for theory and an analytical framework that addressed issues of the technological artefact, the role that developers play in enacting the technology, authority relations between developers and the business client, and at different levels of analysis. This theory was found in the seminal work of Lamb & Kling (2003) in their conceptualization of the *user as a social actor*, with its theoretical antecedents in new institutional theory.

3 Conceptual Framework

3.1 Institutional Theory

New institutional theory has a sociological origin and does not examine phenomena at the individual level. Rather, the behaviour of actors – whether individuals (systems developers) or other social entities (project teams or The Bank) – is attributed not to the motives of that entity, but to its context or higher-order elements at a conceptual level above it. For example, individual action derives from scripts or schemas drawn from shared cultural systems of organizational sub-systems; firm level behaviour and attributes are shaped by the organizational field.

The institutional elements that structure people's work activities in formal organisations in modern society have been studied extensively in organisational theory (DiMaggio & Powell, 1983; Tolbert & Zucker 1996; Scott, 2001). According to Scott (2001), new institutional theory is a body of knowledge that studies the relationships between organisations and their environments focussing on how structures become established (or institutionalised) as guidelines for social behaviour. Institutions, therefore, are taken for granted standardised sequences of activity which establish and maintain features of social life (DiMaggio & Powell, 1983), and according to new institutional theory (Scott, 2001) these influence mechanisms force organisations to conform to norms, traditions, and social expectations.

In our case study, we understand *institution* to manifest itself as taken for granted, or standardised activities that shape – in some instances they constrain – work practices. Using this working definition, we consider the SDM has become institutionalised within *The Bank* because of its longevity of use and its associated practices have become routine.

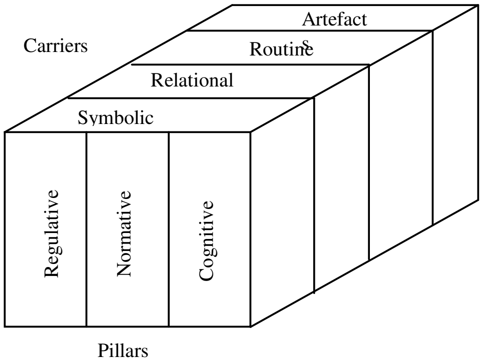

Figure 1. Institutional Pillars and Carriers (Adapted from Scott, 2001:77)

Scott (2001) provides an encompassing framework bringing some coherence to the wide-ranging literature on new institutional theory. This framework known as the 'three pillars' (see Figure 1) posits that institutions are comprised of *regulative*, *normative* or *cultural-cognitive* analytical elements, that together with associated activities and resources, provides a different basis for legitimacy, and hence, social conformance (Scott, 2001:48). The *regulative* pillar gives emphasis to the role of coercion, mandates, monitoring and sanctions to establish and maintain formal and informal systems of behaviour. The *normative* pillar draws on the concepts of appropriateness, expectation and introduces an obligatory dimension to social life. This view defines what people should do and prescribes how things should be done, legitimising role-based actions of individuals. The *cultural-cognitive* pillar stresses the frames through which meaning is made by individuals. This view explains how individual's everyday actions are constrained by the common beliefs and culturally supported norms and values that shape their interactions in their social world. The three pillars form a continuum moving from the conscious (legally enforced) to the unconscious (taken for granted). These three pillars of institutions, according to Scott (2001) are transmitted by being embedded in various types of repositories or carriers. Scott (2001:77) identifies four types of carriers: *symbolic systems, relational systems, routines*, and *artefacts* (*c.f.* Figure 1) and we provide preliminary examples from the case where these carriers materialise:

- Symbolic systems (rules, standard processes, values, to widely held beliefs or ideas in the heads of organisational actors). For example, many developers held the view that the SDM is helpful in that it provides a common language enabling communication of ideas between developers, the client, and those external to the organisation.

- Relational systems (governance systems emphasising authority or power). For example, the business client maintains control over the development process based on funding and the mandatory use of the SDM
- Routines (habitualised behaviour or repetitive patterns of activity such as standard operating procedures encoded into technology or soft organisational routines such as jobs). For example, developers were required to produce documentation at all stages of development, and to report to the client for sign-off before the next stage of development can occur.
- Artefacts (objects complying with mandated specifications, meeting standards, or objects possessing symbolic value). For example, we argue that the methodology itself is an 'artefact created by human ingenuity to assist in the performance of various tasks' (Scott, 2001:81). We illustrate how the SDM mandates signatures; follows recognised industry conventions of systems development; and for developers, possesses symbolic value in terms of demonstrating their professional identity.

An institutional perspective offers several advantages for our research. Firstly, a central principal of new institutional theory is that institutions operate at various levels (jurisdictions of the institutional form), from the world system to interpersonal action; and are transmitted by various types of carriers, including technical artefacts (Scott, 2001: 81). Secondly, institutional theory can be used to analyse all types of organisations, because all organisations are institutionalised, albeit to varying degrees (Scott, 2001). For instance, all organisations (and in particular, The Bank) are subject to regulative processes and operate under local and general governance structures. All organisations are socially constituted and are subject to institutional processes that define what forms they can assume and how they may operate legitimately (Scott, 2001).

New institutional theory therefore is a powerful framework to explain the actions of actors in work practices, and the deployment of ICTs. New institutional theory focuses on social and contextual aspects at a macro level – the environment – for understanding various actors' behaviour in a work practice, as opposed to 'behaviourism and microeconomics that argue a case for organisational behaviour based on rational choice, methodological individualism or actor-centred analyses' (Schneiberg and Clemens, 2006:195). As our interest lies in investigating the interactions between systems developers, business clients, and context in relation to a specific systems development methodology, we adopt an institutional lens in line with other information systems researchers (Avgerou, 2000; Currie, 2009; and Gosain, 2004).

In a survey of the 'landscape' of recent research employing an institutional lens in IS research, Mignerat and Rivard (2005) reported that studies at the organisational level constituted the majority of studies published, with a lack of contributions at lower levels of analysis such as organisational sub-systems (or groups). Lamb (2006) asserts that we need examples of how institutional influences manifest themselves in the day-to-day actions that propel collaboration and use of ICTs. Building on these methodological requirements and the need for contributions based on the role of groups (developers), we apply a particular new-institutional approach, Lamb & Kling's (2003) *user as social actor model* as a conceptual lens to understand methodology enactment within a single organisational setting.

3.2 Social Actor Model

Seen as a means to explain the role that SDMs play in influencing the systems developer-business client relationship, the *user as social actor model* is most appropriate and was chosen for theoretical and methodological reasons. As Lamb & Kling (2003: 219) offered, "the model provides a framework for the systematic research of complex, highly contextualised ICT use in organisations, rather than the study of isolated aspects of ICT use in de-contextualised settings". We also considered the model provides an appropriate theoretical lens to examine SDM enactment; first, because of its emphasis on exploring the impact of institutional structures on the enactment process in organisational settings; and second, because of its focus on networked technologies in increasingly knowledge-intensive industries such as the finance and IT industry.

Our research made use of the *user as social actor model* by illuminating methodology enactment at multiple levels or jurisdictions of the institutional form: individual (systems developer as a social actor), organisational sub-system (the IT department with *The Bank*), organisation (*The Bank*), and organisational field (the finance and IT industry). The following paragraphs provide descriptions of the model in terms of the context of the research example.

The *social actor model* involves four dimensions — interactions, identities, affiliations, and environment that characterise organisational members and their enactment context. According to Lamb (2006) *interactions* and *identities* relate organisationally situated individuals to others and to the information technologies they use to interact with and present themselves to others. The second two dimensions — *affiliations* and *environments* relate people to their organisation, and to the industries and environments of those organisations. To illuminate the institutional context that impacts on methodology enactment, we provide a general description of the four interdependent social actor dimensions.

- **Affiliations** represent inter and intra-organisational relationships created and supported by organisational members as a result of their day-to-day activities as part of the organisation. Systems developers work together comprising social networks. These networks exist within *The Bank* but also apply to the IT and financial industries as well, and to a wider national and international context.
- **Environments**. The environment an organisation operates in is formed by the kind of affiliations it has formed with industry, financial institutions and its clients. Methodology enactment recognises the regulated and/or institutionalised practices of *The Bank*, and other associations that circumscribe organisational action.
- **Interactions**. Systems developers see themselves as organisational members working with others (clients, and business partners exchanging information) enacting a methodology (and other media such as email, telephone, web sites) in support of their interactions. Information and resources are mobilised as systems developers engage with *affiliated* organisations.
- **Identities**. Systems developers regularly enact SDM s to compile and present information to various affiliates. In so doing, they create an identity for their organisation and for themselves. Systems developers are therefore defined by their avowed presentations of the self and ascribed profiles of organisation members as individuals (analyst programmer) or a collective entity (IT professional).

4 Research Approach

The research approach adopted in this study is that of an interpretive case study (Walsham, 1995; Klein & Myers, 1999). As pointed out by Kling *et al* (2005), people's interpretations of information technologies are based on prior beliefs, and the perceived new opportunities and demands it creates. How systems developers interpret an SDM is important because those with different interpretations will enact the SDM differently. Therefore, an interpretive case study was chosen to produce a subjective albeit shared (between the researcher and the interviewee) understanding of phenomena.

The research study was carried out in a large Australian bank. The banking and financial services sector was chosen because of the extremely important role that IT plays in the success of companies in this industry, and *The Bank* selected has extensive experience and use in practice of an in-house developed systems development methodology (SDM). Importantly, the banking industry is highly technical, highly competitive, highly regulated and institutionalised. The selection of the case site was based on a combination of accessibility to the company's IT managers and project members, and interestingness – in the sense that the chosen bank is one of Australia's top four banks, and its IT organisation is considered to be a leading player in providing state-of-the-art IS solutions to customers.

The sampling strategy for the interviews included a combination of purposeful and theoretical sampling (Schwandt, 2001:232). Three occupational functions within *The Bank* were selected for their similarities as well as their differences. 30 interviews were conducted with systems developers comprised of project managers, senior consultants, and consultants within the systems support, new development, and method support divisions of the IT division.

To commence analysis, the first author created a list of codes prior to fieldwork based on the *social actor model*. Each of the four dimensions of the model are further comprised of four characteristics and behaviours of connected and situated individuals. These sixteen characteristics (see Appendix) were used to develop and operationalise the initial coding scheme for the qualitative analysis of data (to be discussed in the Analysis section next). In attempting to gain theoretical understanding of the complexity of the text, the *social actor model* was used in a form of content analysis (Schwandt, 2001:34) where the text was systematically listed, coded and categorised according to the sixteen behaviours of the connected and situated individual.

The list of researcher-constructed labels that best captured the description of the phenomenon was then deductively applied to the text to codify and extract the data associated with each interview. This same format was carried through the entire thirty interviews. In terms of data management issues, the process of analysis was assisted by and recorded in a database through procedures such as importing chunks of transcribed interviews, adding comments and reflections, sorting the interpretations by code; and text retrieval of selected instances into the body of the research report.

Following data collection and initial analysis, the author developed and shared a case report (including a case summary and preliminary elements of analysis) with a key and current project manager involved in the use of the SDM. This manager commented on the report and gave confirmation of many points and qualifications of others. Drawing on this report, the interviews, the scrutiny of informants, and the researcher's relativistic and subjective understanding of the case phenomena, the paper presents a summary of the analysis of the case.

5 Analysis

5.1 The Case

We used Lamb & Kling's (2003) multi-dimensional conceptualisation of the social actor to guide our qualitative analysis of data. In doing so, the paper operationalised Lamb and Kling's model in terms of mapping the case interview text to the model's sixteen constructs. Through empirically supported examples, we illustrate how dimensions of the *social actor model* manifest itself in the day-to-day actions of methodology enactment within *The Bank*. In this case narrative, we present informed perspectives in a coherent and convincing story. It is important to discuss how 'control' and 'power' related concepts were chosen. Initially, we did not set out to study power, instead we deliberately kept the case interview questions open leaving the developers to tell us their story about what influenced them in their enactment of the SDM. Using the *social actor model* as a lens to draw meaning from the interviews, a recurring theme among the responses was the developer describing their subordinate relationship when dealing with the business client. This key theme turned our attention towards power and the authority relations existing between the business client and systems developers. What follows are sample extracts of the application of the four dimensions written as a narrative with a focus on organisational context.

5.1.1 Affiliations

According to Lamb (2006) networks are a basic configuration for organising social, economic and political exchanges. According to Lamb (2006), because relationships are multi-level, multi-network (i.e. group, inter-group, organisation), and as social actors in their various forms and functions become enrolled in multiple networks, they begin to translate their interests. Within *The Bank* relationships between the social actors and their differing interests is demonstrated in the following excerpt where the method support manager is commenting on the unequal power relationship between business and developers:

Business sometimes do hold development to ransom, so to speak. So that's another part of the culture. Really they should be working together to try and deliver solutions rather than using contracts as ransom to force them to do something. So it ends up, at the end of the day, a lot of the management is structured such that project managers and CIO's are rewarded or punished based on their ability to deliver projects on time.

A relative new-comer to *the Bank* (a developer) was also speaking about how he saw the power relations between project leaders and CIO's being rewarded within *the Bank* based on delivering projects on time:

a lot of the management is structured such that project managers and CIO's are rewarded or punished based on their ability to deliver. So they apply that pressure downward. Management and business are probably the most in-flexible areas. Because they're very much used to business having a lot of control. They have a lot of power, because they hold the money. So business sometimes do hold development to ransom, so to speak.

Another interesting observation was how the development side referred to the business client. The dichotomy between the systems developer and business client is evident in the interviewee's reference to the business client as "business" rather than as, for example, "clients", "partners", or "domain experts". The term "business" as used by developers implies superiority: one who consumes, controls, prescribes and manages. From the perspective of developers, power was seen to be vested with the business client. Asked specifically who drives systems development, a senior analyst responded in a way that was representative of many similar comments:

It's the business, definitely. Sometimes the IT areas will, once they get a project, try to drive what they think. But on the whole, the business are paying [for services and products] and whatever they want, gets done.

The same senior analyst even admitted that some project managers are scared of the power that business wields:

I have worked for managers where they have agreed to deadlines that are too close and not reasonable. Sometimes too, I think they get a little bit scared of business. Like if the business wants something and they're demanding it, they're scared to tell them that their request isn't reasonable.

The above examples are saying that the development life-cycle and sign-off process embedded within the SDM creates a mechanism for the business client to exert and maintain power over the systems development group. Power and control is enacted as follows. Seeking sign-off and approval formalises the 'structural' dependence of the system developer on the business client. Business will only 'sign-off' on the specifications and deliverables and take responsibility for project outcomes if they are satisfied that the system will meet their needs. In turn, business commit themselves to fund the next phase of development. Without sign-off, in principal, development cannot continue. The developers are dependent on gaining 'business' internally and accordingly often under-estimate costs, and agree to unrealistic schedules.

5.1.2 Environment

By focussing outside the organisation, the *environment* framework draws attention to and enables us to better understand the stabilized and institutionalised practices that take place within *The Bank*. Lamb (2006) asserts that the *environment* an organisation operates in greatly affects the enactment of information technologies, and cites examples of institution such as regulatory agencies, professional codes of ethics, laws, and industry-wide practices. These environmental influences also have 'power' over the enactment of the SDM indicating it's not just the business client who has all the power.

An example of the *environment* exerting power in the form of institutional practices (such as industry standards) on developers in *The Bank* is that for the bank to gain quality certification it would need to demonstrate that certain standard processes [E-STAND] were in place, and that the bank followed a methodology in both project management and software development. A project manager agreed that with a mixture of skill sets within The Bank a common standard has advantages:

I believe the bank wants to reach some kind of a maturity where there is a format and standards. And it just provides some visibility as far as ensuring that the outcomes are doable. With a mixture of skill sets it's important that we all have the same standards and the outcomes are the same. A common benchmark is good because of this, and it can be used as a guideline – and [the methodology] allows everyone to follow a particular guideline.

A further example of the environment exerting institutional practices upon the Bank (in the form of standards) is the recent requirement for software applications to comply with the Australian Prudential Regulatory Authority (APRA) codes of practice. A senior analyst remarked that this was an unusual type of project because it wasn't client initiated:

The project I'm on now, it's a bit different because it's about compliance. It's to do with the code of practice, so it's something that [The Bank] previously hasn't had to do, but it relates to business lending and mortgages. It's hard to define who the users of this project are because the project hasn't been initiated by a business unit. It's initiated by The Bank to comply with the code.

In summary, the environment exerts power in the form of technical and institutional practices on developers within the Bank. The origin of this power did not come from the business client. For instance, *The Bank* followed industry standards because of the high mobility of workers in the IT profession, and the requirement to conform to regulatory body's code of ethical practice.

5.1.3 Interactions

According to the *social actor model* individuals are involved in networks that take shape within and among organisations. Networking refers to the *interactions* where organisational members work and interact with affiliates using a methodology (and other media) in support of their interactions. In terms of *The Bank*, the SDM is seen by developers as a vehicle to bring project members together and coordinate their tasks when interacting with clients, industry bodies and business partners. The examples describe systems developers in their day-to-day working roles networking and relating themselves to others and to the SDM they interact with. The analysis identified a range of interaction behaviours such as producing documentation, communicating, and acting in constrained ways that made the development process cumbersome and inefficient.

A programmer commented on the methodology as a control mechanism saw the amount of documentation required as a lot of red tape and involving excessive time:

basically the main argument against this red tape or what is seen as a red tape is the time factor involved. …obviously the more approvals you need the more people you need to contact. If certain people aren't around then it takes longer for the approval to come through and if that goes back to the business, they complain about time and you have to explain to them about the time it takes to get these approvals…

However, the main reason cited for seeking to communicate in legitimate ways was to get documents signed and gain approval from the business client to commence the next stage of development. To do so, there needed to be visibility of development work as a project leader commented:

… producing a document is one way of providing visibility of what's actually happening and with all the formal documents that are required to be signed it's approval to go to the next stage and that the work can be done. Also with funding - project funding is dependent on these documents having to be produced. In a large scale development the funding is very important in each phase. So therefore you have to produce some kind of deliverable to prove what you've done.

Developers also saw sign-off negatively – it's a way whereby business clients maintain power and control over the development process. A project manager mentioned a functional role of the methodology, through sign-off, was for the business client to keep control of the project:

…you have to get sign off at various points. Yes, the methodology is used by the technology people to build things. But, before you can get funding for the next stage the technology group needs to provide to business things for the project to then proceed to the next phase. So if you want funding to go on further, you'll need to do things. So it [the SDM] forces you to do things [produce deliverables].

However, the same project manager commented that the process of gaining sign-off was an impediment to delivering systems on time:

the areas where things can be a real barrier is the sign-off process. So let's say we produce a design document and we send it out for review and sign off. Well, you might then have to wait a week or two weeks for other areas to review it and sign off and chase them up because they've all got busy lives as well. So you're in a constant spin trying to get things resolved and issues *being raised and closed off. You reissue the document, so you could spend a month chasing all that stuff through.*

Another developer agreed that gaining sign-off was time consuming and frustrating:

It's a fact that software needs a lot of documentation anyway. But [this methodology] adds an extra step… to say that you've done it, you have to get it signed off. It's another hoop to run through. And chasing sign off is a real pain. You've got to know who to go to, when, and the turn around time - it's not always pretty.

Gaining sign-off is a work structure imposed by the methodology and while not unanimously favored by developers, the business client maintains strict compliance to these rules. The business client is able to exercise power through control of the SDM mandating the generation of specifications becoming actionable documents requiring a signoff at each stage. This is work culture imposed by the business client via the SDM on developers.

The prominent interactions illustrate two important points: that sign-off is a work structure unobtrusively imposed by the business client and is embedded within the SDM. From a power perspective, the life-cycle, sign-off, and routine patterns of work create a covert mechanism for the business client to exert and maintain control over the systems development group. In this case, control equals power. Second, the excerpts also illustrate that the business client has 'ownership' of the SDM and therefore controls important aspects of systems development, and the systems development process.

5.1.4 Identities

According to the model, social actors enact ICTs to create a positive image to construct identities and to control perceptions. This identity was used to project an individual's legitimacy or an organisational unit's strength to the business client. A support programmer was able to comment that 'using' the methodology made his boss happy and the business client was satisfied. By using the method the analyst saw himself in a legitimate role in the eyes of those he had affiliations with in *The Bank*:

I've got a small job at the moment, so I drew up a detailed design document. I put it on a screen, with the [costing] amounts and that sort of thing. I showed it to my manager and he was happy with it and was happy with the estimate of what I thought it would cost. He was happy to get funding for that, and business is happy because we've saved them time.

The *identities* dimension described the visible identity of individual social actors as organisational members having a methodology use component. In terms of the example, knowing how to use the methodology and using the methodology competently can construct identities, legitimises their role, and constructs perceptions that they're professional in the eyes of the business client.

5.2 Summary of the Findings

The ***affiliation*** transcripts confirm the inherent power of the business client. The interviews are saying that in the end it is the business client who has control over the systems development process, and bears the most responsibility for the system in terms of funding and signing off on it. Systems developers need the business client to fund the design and construction of new or enhanced systems. However, there is a dichotomy of mind-sets. The business client is portrayed as more interested in controlling costs, monitoring deadlines and delivering projects on time, whereas the developer is more interested in building quality systems and employing their technical expertise.

The **affiliation** section tells us that it is the policies and practices embedded in the SDM through sign-off and stage-gate funding constitutes a form of 'structural' exercise of power (Markus and Bjørn-Anderson, 1987) in the form of developer dependence on the business client for important resources. While a form of overt power, this finding indicates that the constraints based around the accepted and everyday use of a methodology by systems developers obviates the need for more direct forms of control.

Second, in the ***environment*** section, examples were provided where the environment the bank operates in greatly effects the enactment of the SDM. Examples of adherence to industry-wide and global work practices included: development phases being based on the traditional water-fall lifecycle, systems built complying to standards imposed by regulatory agencies such as *APRA*, *The Bank* mimicking other organisations by placing the methodology on an intranet site, and having to conform to specifications agreed to with major technology partners. These examples illustrate a source of power emanating from other than the business client. The environment imposes on the developer a requirement to comply with industry, national and global work practices, where the enactment of the SDM is subject to external institutional forces.

From the ***interactions*** section, the examples illustrate how the development life-cycle, sign-off, and routine patterns of work embedded within the SDM create a mechanism for the business client to exert and maintain control over the systems development group. The examples illustrate that the business client has 'ownership' of the SDM and therefore has control over important aspects of systems development, and accordingly is able to exert unobtrusive power over systems developers. What has not been reported in the literature before is that through ownership and control, the business client can be considered a user of the methodology too.

As reported in the ***identities*** section, developers are dependent on the business client to validate and legitimate their contributions to the organisation. Knowing how to use the methodology and using the methodology competently can construct their *identities*, legitimise their role, and construct perceptions that they are professional. There were multiple data points confirming that the enactment of the methodology legitimises their role as a systems developer in the eyes of a project manager or the business client. Hence, systems developers pursue their interests directly by invoking 'directives' prescribed by the methodology, while acknowledging the legitimacy of the business client.

In sum, we cited examples of how institutional structures operating at various levels of the organisational field provide an overarching, framing context within which systems developers were constrained in their use of the SDM. We inquired into the circumstances within which systems developers used the SDM, and we identified conditions that resulted in the subjugation of developers by the business client, leaving them with little control over the development process. We concluded that the advantages in terms of whose interests are met in the systems development process are clearly in favour of the business client.

6 Discussion and Conclusion

Using Lamb & Kling's (2003) *social actor model* as a means for analysing the case, we found that pre-existing structures embedded in the SDM constrain the actions of the systems developer. An analysis of the transcripts through the lens of the *social actor model* enabled us to identify the source of authority and power afforded the business client, and to identify mechanisms of unmistakable power operating in The Bank. One local source of power in favour of the business client is a set of development procedures (sign off, and the stage gate approval process) that have transpired over time to institutionalize their interests in structures embedded in the SDM. As a consequence of the SDM being ‘owned’ and controlled by the business client, and the SDM being mandated, developers were constrained in their actions by the apparent neutral technology of the SDM (the methods and techniques of systems development); and the need to ‘rationalise’ their work practice – a common theme among developers is that ‘we all need to use the SDM to speak a common language’. A local source of power in favour of the client is the inevitable market pressures such as the clients’ ability to outsource development work rendering the developer dependent on the client and subject to unreasonable demands in terms of schedules.

To further answer our research question we frame our discussion by drawing upon contributions from institutional theory (Scott, 2001). This discussion extends our understanding of SDM enactment by showing how institutional structures (such as authority, norms, and symbolic values) embedded within the methodology (*c.f.* Table 1) are active forces in the systems development process. We found that the SDM is a carrier of institutional logics and can be used to explain how the method carries power.

In illustrating how institutional logics (processes and social structures) shape the method enactment process, the case shows that pre-existing structures that have developed over time (such as rules, norms and beliefs) embedded in the SDM play an active role in constraining and enabling developers in the ISD process. As this case demonstrated, the day-to-day work activities involved in systems development are rather fixed or predetermined by the institutionalised nature of the technical artefact – the SDM. For instance as summarised in Table 1, the structures of the SDM provide a repertoire of already existing institutional principles of work (e.g. conventions, work practices, common understandings, authority relationships) that developers enrol in their activities.

With reference to Table 1, under the headings of the regulative and normative pillars, examples from the case provide grounded evidence of how the SDM encodes and embodies institutional principles that constrain the routines of organisational actors. Examples from the cultural cognitive pillar show how enactment of the SDM over time, leads to the development of change-resistant cognitive schemas (norms and values) that are perceived as natural and legitimate by developers. Furthermore, this case has demonstrated, as it is the business client who is in control, it is the values and conventions of the business client that holds legitimacy in *The Bank*.

Table 1. Institutional Pillars and Carriers from the Case (adapted from Scott (2001:77)

		Pillars	
Carriers of institutional logic	*Regulative: regulations and* ***rules*** *that govern behaviour.*	*Normative: appropriateness, expectations and customs that define and prescribe how things should be done.*	*Cultural-Cognitive: frames through which meaning is made, such as shared* ***beliefs*** *and mental models.*
Symbolic systems	“Walk-through” meetings took on a ritualistic character in order to convey a powerful message to developers: ‘cooperate, come to us, and we will reward you’.	Through the habitual use of the SDM template in producing lifecycle deliverables, developers used this as evidence of design creativity and work performance.	Many developers held the view that the SDM provides a common language and valued standardized terms enabling communication of ideas between developers, the client, and those external to the organisation.
Relational systems	The business client	Systems developers wanted	Developers when they join The Bank

	maintains control over developers through funding. Developers can't proceed until each stage is signed off.	the SDM to be changed (updated) but the business client resisted.	accept their role in the existing order of things because they see it as natural.
Routines	Systems development requires a standard set of documents to be completed.	*The Bank* has standard job roles of consultant, senior consultant, project leader, CIO etc, involving a hierarchy of authority.	The use of the SDM goes relatively unnoticed by developers. It is habitualised and part of the work culture of the organisation.
Artefacts	SDM use is mandated for all new projects & maintenance projects	The SDM is based on the traditional SDLC. External parties incl. consultants, contractors, H/W & telecoms. providers know of the phases used within *The Bank*.	Using the SDM creates an image for developers that they are professional.

The *user as social actor model* has been applied to the study of information systems elsewhere: Ferneley & Light (2008) in a study of different user groups' appropriation of mobile and ubiquitous computing, and Rowlands (2009) in a reflective essay; however these works do not draw upon the three pillars of institutional theory to understand the active role of the ICT as an institutional carrier. Our research, on the other hand, does. Specifically, one theoretical contribution is illustrated in understanding the active role of SDM as an institutional carrier.

The findings show that the SDM can assume the properties of an institution on the basis that it constitutes the background condition for action, enforcing constraints, giving direction and meaning, and setting the range of opportunities for undertaking action.

The case reported in this paper builds on research suggesting that SDMs significantly inform and shape the cognitions and actions of organisational members engaged in systems development (Hirschheim and Klein 1989). We found that the influence of the methodology occurs through the prescriptions of process mandated by the methodology, through the experiences and learning from previous use of the methodology, and through habitualised behaviour or routines that shapes developers' approaches to using the SDM in their workplace. This finding is consistent with other research (Gosain, 2004) that technical artefacts act as an important institutional embodiment (as a carrier) serving to preserve rules by constraining the actions of human agents on the one hand; and that the technical artefact is subject to institutional forces and institutional processes that set the rules of rationality, on the other.

In this case, we have provided grounded description of systems developers working within a discipline that provides scaffolding for their actions. Through the application of Lamb & Kling's (2003) model, we identified how the discipline of systems development provides generalisable procedures (stage gate funding, sign-off, etc) applied in the enactment of the SDM that are largely based on power structures involving the client and developer. Through the application of Scott's (2001) framework – also known as the 3 pillars – we illustrated how the discipline of systems development within *The Bank* is legitimized by change resistant norms and values.

In conclusion, we have argued that an institutional perspective of methodology enactment brings about the identification of different elements across various levels of the organizational field that might otherwise escape analysis. As stated in the *Introduction*, a lack of established theory about SDM enactment necessitated the generation of a number of new perspectives and empirical insights adding to the existing body of knowledge in this arena. Indeed, the findings developed in this study and summarised in Table 2 define five theoretical statements or high-level propositions about the distribution of power, control and responsibility between systems developers and the business client from an institutional theory perspective.

Table 2. Theoretical Statements from the Findings

1	Enactment is a complex INSTITUTIONAL process. Methods of systems development encode organisational values in the form of INSTITUTIONAL STRUCTURES.
2	The developer is CONSTRAINED. Pre-existing structures (such as rules, norms and beliefs) embedded in the SDM and EXTERNAL forces to the organisation play an active role in constraining human agency in the systems development process.
3	The SDM is a CARRIER of institutional logic. The SDM acts as a major institutional CARRIER where its enactment becomes taken for granted in the form of habitual behaviours and routine patterns of work.
4	Enactment is CONTROLLED by the business client. The business client has 'ownership' of the SDM, controls the resources, and is able to exert power over the systems development process.
5	Enactment is POWER based. The life-cycle and sign-off process embedded within the SDM creates a mechanism for the business client to mobilise power and thereby maintain control over the systems development process.

6.1 Limitations & Further Research

First, data for our case only came from developers, and no business clients were interviewed or their opinions heard. We recognise the limitation of this approach. A natural area for future research is to study the other half in the business client — systems developer pair.

Secondly, while we claim to have conducted multi-level research by focussing on constructs such as affiliations, interactions and the environment, all our empirical evidence was arrived at from an individual perspective or collection of individuals (group level) through personal interviews. The evidence collected under-represents influences of the organisational field. This limitation poses opportunities for advancement in terms of methodological tools for the collection and analysis of data at a level higher than the individual (*c.f.* Table 1).

Finally, in terms of theoretical contribution, the paper found that the *user as social actor model* enabled us to draw out the power concepts, and institutional theory explains how the SDM acts as a carrier of power, but not why developers are compliant with an unequal power arrangement. The analysis needs to be strengthened by the incorporation of models of power allowing us to understand why there is cooperation from developers with the business client in a scenario involving a conflict of interest

6.2 Conclusion

This paper has applied Lamb and Kling's (2003) user as social actor model and new institutional theory (Scott, 2001) to contemporary systems development by focussing on a field study of SDM enactment. Two reasons make the findings an unusual case in the research literature.

The first is that the business client is able to maintain control over developers unobtrusively through institutionalised structures embedded within the SDM. Prior SDM enactment studies have either not encountered or not recognised cases where institutional structures embedded with the SDM directly determine power relationships between the developer and the client. The second is that according to the literature, systems development is supposedly a collaborative process (Bjerknes and Mathiassen, 2000). However, in this case developers saw systems development as an unequal process, with a conflict of interests in which the business client achieves their objectives to the relative disadvantage of developers. Developers saw the business client directly controlling the systems development process; and developers compelled to follow the method process found that the method has structures that controlled their behaviour.

References

Avgerou, C. "IT and Organisational change: an Institutionalist Perspective", *Information Technology & People*, 13(4), 2000, pp. 234-262.

Aydin, M. Harmsen, F. van Slooten, K. and Stegwee, R. "On the adaptation of an agile information systems development method", *Journal of Database Management*, 16(4), 2005, pp. 24-40.

Beath, C. and Orlikowski, W. "The Contradictory Structure of Systems Development Methodologies: deconstructing the IS-user relationship in Information Engineering", *Information Systems Research* 5(4), 1994, pp. 350-377.

Bjerknes, G., and Mathiassen, L., "Improving the Customer-Supplier Relation in IT Development", Proceedings of the 33rd Hawaii International Conference on Systems Sciences (HICSS), 2000.

Currie, W. "Contextualising the IT aretefact: towards a wider research agenda for IS using institutional theory", *Information Technology & People*, 2009, 22(1), pp. 63-77.

DiMaggio, P. and Powell, W. "The iron cage revisited: institutional isomorphism and collective rationality in organisational fields", *American Sociological Review*, 48(2), 1983, pp.147-160.

Ferneley, E. and Light, B. "Unpacking user relations in an emerging ubiquitous computing environment: introducing the bystander", *Journal of Information Technology*, 23, 2008, pp. 163-175.

Fitzgerald, B. Russo, N. and Stolterman, E. *Information Systems Development: Methods in Action*, McGraw Hill, Berkshire, 2002.

Gosain, S. "Enterprise Information Systems as Objects and Carriers of Institutional Forces: the new iron cage?" *Journal of the Association of Information Systems*, 5(4), 2004, pp. 151-182.

Goulielmos, M. "Systems Development Approach: transcending methodology", *Information Systems Journal*, 14, 2004, pp. 363-386.

Hirschheim, R. and Klein, H. "Four paradigms of information systems development", *Communications of the ACM*, 32(10), 1989, pp. 1199-1216.

Huisman, M. and Iivari, J. "Deployment of Systems Development Methodologies: Perceptual Congruence Between IS Managers and Systems Developers", *Information & Management*, 43, 2006, pp. 29-49.

Kautz, K. Madsen, S. and Nørbjerg, J. "Persistent Problems and Practices in Information Systems Development", *Information Systems Journal*, 17, 2007, pp. 217-239.

Klein, H., & Myers, M., "A Set of Principals for Conducting and Evaluating Interpretive Field Studies in Information Systems", *MIS Quarterly*, 23(1), 1999, pp 67-94.

Kling, R. Rosenbaum, H. and Sawyer, S. *Understanding and Communication Social Informatics: a framework for study and teaching the human contexts of ICTs*, Information Today, Medford NJ, 2005.

Lamb, R. "Alternative Paths Toward a Social Actor Concept", *Proceedings of the Twelfth Americas Conference on Information Systems,* Acapulco, Mexico, 2006, pp. 4113-4123.

Lamb, R. and Kling, R. "Reconceptualising Users as Social Actors in Information Systems Research", *MIS Quarterly*, 27(2), 2003, pp. 197-235.

Madsen, S. Kautz, K. and Vidgen, R. "A Framework for Understanding How a Unique and Local IS Development Method Emerges in Practice", *European Journal of Information Systems*, 15, 2006, pp. 225-238.

Markus, M, and Bjørn-Andersen, N. “Power over users: its exercise by systems professionals”, *Communications of the ACM*, 30(6), 1987, pp. 498-504.

Mignerat, M. and Rivard, S. “Positioning the Institutional Perspective in Information Technology Research”, working paper 05-01, Department of Information Technologies, HEC Montreal, 2005.

Nandhakumar, J. and Avison, D. “The Fiction of Methodical Development: a field study of information systems development”, *Information Technology & People*, 12(2), 1999, pp. 176-191.

Rowlands, B. “A Social Actor Understanding of the Institutional Structures at Play in Information Systems Development”, *Information Technology & People*, 22(1), 2009, pp. 51-62.

Schneiberg, M. and Clemens, E. “The Typical Tools for the Job: Research Strategies in Institutional Analysis”, *Sociological Theory*, 24(3), 2006, pp. 195-227.

Schwandt, T. *Dictionary of Qualitative Inquiry*, 2nd Ed, Sage, 2001.

Silva, L. “Epistemological and Theoretical Challenges for Studying Power and Politics in Information Systems”, *Information Systems Journal*, 17, 2007, pp.165-183.

Scott, W. *Institutions and Organisations*, Thousand Oaks, CA, Sage, 2001.

Tolbert, P. and Zucker, L. The Institutionalisation of Institutional Theory”, in *Handbook of Organization Studies,* Clegg, S. et al (eds.), Sage, London, 1996, pp. 175-190.

Walsham, G. “Interpretive Case Studies in IS Research: Nature and Method”, *European Journal of Information Systems*, 4(2), 1995, pp.74-81.

Wynekoop, J. and Russo, N. “Studying Systems Development Methodologies: an examination of research methods”, *Information Systems Journal*, 7, 1997, pp. 47-65.

APPENDIX

Table A1. Multi-dimensional Conceptualisation of a Social Actor

SOCIAL ACTOR DIMENSIONS	**CHARACTERISTICS and BEHAVIOURS of Connected and Situated Individuals (Lamb and Kling, 2003:213)**
Affiliations (Definition: organisational and professional relationships that connect an organisation member to industry, national and international networks).	Social actor relationships are shaped by networks of organisational affiliations.
	Relationships are dynamic, and related informational exchanges change with flows of capital, labour, and other resources.
	Relationships are multilevel, multivalent, multi-network ie. local/global group, organisation, inter-group, inter-organisational culture.
	As relationships change, interaction practices migrate within and across organisations.
Environments (Definition: stabilised, regulated and/or institutionalised practices, associations and locations that circumscribe organisational action).	Organisational environments exert technical and institutional pressures on firms and their members.
	Environmental dynamics require a display of overall competence.
	ICTs are part of the organisational environment.
	ICTs are part of the industry, national, and/or global environment.
Interactions (Definition: information, resources and media of exchange that organisation members mobilise as they engage with members of affiliated organisations).	Organisational members seek to communicate in legitimate ways.
	Organisational members build, design and develop interactions that make information actionable.
	ICTs become part of the interaction process as people transform and embed available informational resources into connections and interactions.
	As organisational members, people perform socially embedded (role-based), highly specialised actions on behalf of the organisation.
Identities (Definition: avowed presentations of the self and ascribed profiles of organisation members as individual and collective entities).	Social actor identities have an ICT use component.
	ICT-enhanced networks heighten multiple identities as expert or novice.
	ICT-enhanced connections among organisation members transcend roles.
	Social actors use ICTs to construct identities and control perceptions.