The Performativity of Routines: Theorising the Influence of Artefacts and Distributed Agencies on Routines Dynamics

Dr Luciana D'Adderio

Advanced Institute of Management Research (AIM)
♦
Institute for the Study of Science, Technology and Innovation (ISSTI)
The University of Edinburgh

Version: September 2007
Please do not quote without the author's consent

Abstract

Drawing from advances in Organisation Studies and recent debates within Economic Sociology and the Sociology of Financial Markets, this paper proposes a theoretical framework that characterises the mutual adaptation between formal routines, rules and actual performances as iterative cycles of framing, overflowing and reframing of knowledge inputs and actions. This framework, combined with the ethnographic observation of the 'engineering freeze' process at a leading automotive manufacturer, allows us to advance routines' theory by 1- capturing the dynamics of convergence and divergence between procedures and performances; and 2- improving our understanding of the influence of artefacts and distributed agencies on routines' dynamics.

Keywords: Organisational Routines, Performativity, Standard Operating Procedures, Rules, Artefacts, Occupational Communities.

1. Introduction

As a unit of analysis, routines represent an invaluable resource to capture organisational change (Simon 1947, Cyert & March 1963, Nelson & Winter 1982, Becker et al. 2005, Pentland & Feldman 2005a). Revealing the internal structure of routines can indeed provide useful insights into many of the basic questions of organisation science (Pentland & Feldman 2005a). Yet, the complexity of this endeavour has meant that routines' theory to date has only just begun to address the routines dynamics that underpin core organisational phenomena such as learning, change and adaptation. In particular, notwithstanding the important recent advances in this debate, we are still short of a full theoretical understanding and empirical characterisation of the micro-level dynamics that underpin routines' evolution. These include the dynamics of interaction between different aspects of routines and the influence of artefacts and agencies on routines' evolution.

This gap in the theory has been exposed by authors who have advocated the need to unravel routines and capabilities' internal dynamics (Pentland & Reuter 1994, Cohen et al. 1996, Feldman 2000, Lazaric & Denis 2001, Zollo & Winter 2002, Feldman & Pentland 2003, D'Adderio 2001, 2003, Becker et al. 2005, Pentland & Feldman 2005a). This work has pointed to the need to "open up the routines' black box" to analyse the interactions between different sides, or aspects, of routines. Categories introduced to capture the routines' internal mechanisms include the distinctions between 'routines-as-representations' and 'routines-as-expressions' (Cohen et al. 1996), 'rules-to-be-interpreted' and 'rules-to-be-executed' (Reynaud 1996, ibid) and 'ostensive' and 'performative' (Feldman & Pentland 2003, Pentland & Feldman 2005a). These approaches have productively shifted the emphasis from a characterisation of routines as undifferentiated monolithic 'objects' to the more sophisticated and productive notion of routines as generative - and continuously emerging - systems characterised by internal structures and dynamics. In doing so, authors have opened up entirely new grounds for exploring some of the most relevant but as yet under-researched questions about the nature and dynamics of routines.

This paper aims to contribute to fill this gap in our understanding of routines' evolution and performance. In our quest to unravel routines' internal dynamics we set our focus on artefactual representations of routines – and, specifically, on standard operating procedures (SOPs) and associated rules – which we use as a starting point for our analysis[1]. There are two ways in which a focus on artefacts is useful. First, artefact-embedded rules and procedures provide vantage points to observe the ostensive (abstract) aspects of routines with respect to which they can serve as "proxies" (Pentland & Feldman 2005b). When embedded in material artefacts, rules and procedures can provide ideal loci for observing abstract understandings and otherwise

[1] In this paper we focus on formal rules and procedures thus neglecting other categories of 'stable systemic traits' (Cohen et al. 1996) such as 'rules of thumb' and 'heuristics'. We also assume that, while explicit rules and procedures are not the same, as formal statements they can afford the same analytical treatment.

embodied views of routines; this is because they becomes more stable and visible, which in turn allows them to act as reference points against which variations occurring to performances can be more easily detected[2]. The second way to understand the key role of artefacts as privileged points of observation – which is more akin to the framework developed here - is that abstract understandings of routines are not simply people-embodied but highly distributed across a complex web of people *and* everyday artefacts. Neglecting to include tools and artefacts in the study of routines dynamics can only provide at best a partial picture.

Starting from these premises, we identify three main research questions**.** First, how can we theorise the mutual adaptation of formal (artefact-embedded) routines and rules and actual performances? For example, what are the micro dynamics that influence the direction and intensity of their interactions, and how do these dynamics influence routines' evolution and adaptation? Second, what is the role of artefacts in mediating these interactions? Third, what is the influence of distributed agencies, including heterogeneous organisational communities, in shaping the co-evolution between different aspects of routines?

To answer these questions we need to acquire a new set of theoretical notions and constructs. Drawing from recent arguments within Economic Sociology and the Sociology of Financial Markets (Callon 1998 & 2006, MacKenzie 2003, 2005 & 2006), we theorise the interactions between formal SOPs and rules, on one hand, and performances, on the other, as iterative cycles of framing, overflowing and further reframing of knowledge inputs and actions. The framing action exerted by SOPs, rules and formal tools delimits and closes search spaces providing guidance and control. Framing by rules and SOPs, however, is never complete: there is always overflowing which opens up search spaces thus introducing scope for divergence, adaptation and change. Overflowing is often followed by further reframing which brings again convergence between the procedure(s) or rule(s) and performance(s). The combined novel empirical focus and theoretical characterisation provides important new insights

[2] See also D'Adderio 2003.

into the mutual adaptation between aspects of routines and the role of artefacts and other agencies as intermediaries in these interactions.

Specifically, our framework provides three main contributions to the routines debate. First, we are able to capture the **interactions between different sides of routines**: the cycles of framing, overflowing and reframing – and their emergent outcomes – form the micro dynamics underpinning the mutual adaptation of different aspects of routines and, ultimately, routines' evolution. Second, we characterise the **influence of artefacts** as intermediaries in shaping the interactions between different sides of routines. In contrast with extant literature that sees formal SOPs and rules as either *flawed representations* that can be easily dismissed/disused or as *full prescriptions* that are compulsively and automatically performed, we show that - in most cases – there is some kind of adaptation. Third, we theorise the **influence of distributed, heterogeneous agencies** (i.e. occupational communities, communities of practice) on routines' evolution. We show how routines' dynamics are the emergent result of contingent struggles amongst competing performative programmes.

Our data was obtained through the ethnographic observation of the 'engineering freeze' process, an upstream section of the product development process at a leading automotive manufacturer. Here, the introduction of a software-based data and process management tool (Product Data Manager, or PDM) and the consequent inscription of the "freeze" routine in software provide vantage points to observe the interaction and mutual shaping of SOPs and performances. The data collection, aimed at documenting the actual contents of routines (Cohen et al. 1996), involved a mix of participant-observation and in-depth, semi-structured interviews. These were conducted over a one and a half years period and span across most organisational functions (Industrial Design, Product Engineering, Analysis, Testing, Production, Manufacturing, Accountancy, Marketing) and levels (designers, shop floor technicians, project and programme managers). The author's technical background allowed for full immersion into the micro-level interactions among processes, artefacts, people and technologies. The data was subsequently analysed and compiled into a case study consistently with an inductive, 'grounded theorising' approach (Glaser & Strauss 1967, Eisenhardt, 1989).

The paper is structured as follows. Section 2 sets out the main theoretical and conceptual foundations for the three main themes in this paper: the interactions between aspects of routines and the influence of artefacts and agencies on these interactions. Section 3 outlines the empirical context, by describing the changes occurring to a key product development routine (the "freeze" process) as a consequence of software introduction. The main body of evidence is presented in Section 4. Section 5 discusses the insights that can be obtained by applying Performativity Theory to the study of routines. Section 6 concludes by drawing the implications for routines' theory.

2. Routines, Artefacts and Agency: from Representation to Performation

In this section we lay out the theoretical premises to the framework that we propose to use to capture the mutual adaptation between artefactual representations of routines (rules, SOPs) and performances, and the influence of agencies on their interactions[3]. Our aim is to argue for a shift away from two extremes that dominate the routines debate: a characterisation of SOPs and formal rules as fixed representations of actual process that fully prescribe actions; and a characterisation of SOPs as intrinsically flawed, descriptive representations of actual process that can be easily avoided. In turn, we put forward a "performative view" of routines dynamics[4]. This captures the interactions between SOPs and performances as iterative cycles of framing, overflowing and reframing, and highlights the fundamental role that artefacts and distributed agencies play in shaping their interactions.

[3] Our framework builds on Cohen et al.'s (1996) distinction between 'routines-as-representations' and 'routines-as-expressions'. At the same time, however, it is compatible with Feldman & Pentland's distinction between 'ostensive' and 'performative' (2003) with artefactual representations characterised as 'proxies of the ostensive'.
[4] The term "performative" as used here is borrowed from Performativity Theory (Barnes 1982, Callon 1998, MacKenzie 2003).

2.1 Routines, artefacts and the role of agency

In our quest to characterise routines' internal dynamics, we thus begin by focusing on the interactions between formal – often artefact-embedded - rules and procedures, on one hand, and actual performances, on the other. When we consider the key role that SOPs and rules play in the emergence and evolution of routines, the scarce attention that they have received in the literature, since Cohen et al.'s 1996 seminal paper, is rather surprising[5]. For routines scholars, this represents a missed opportunity to capture a fundamental issue in routines dynamics: while artefact-embedded representations of rules and routines are mostly introduced to design and manage routines, their outcomes often escape the agents' original intentions. In other words, artefactual representations of routines *are not* the routine (Pentland & Feldman 2005b)[6]. This however should not deter us from using artefactual manifestations of routines as starting points for our analysis. Rather, the processes of translation from SOP to performances and vice versa (coding and de-coding) provide very interesting standpoints to observe the production and reproduction of routines (Cohen et al. 1996, D'Adderio 2003).[7] For example, to what extent do explicit, formal rules and procedures govern practice? How are procedures and their outcomes shaped by contingencies? What are the mechanisms that

---

[5] Cohen et al.'s (1996) distinction between 'representations' and 'expressions' originates from the biological distinction between genotype and phenotype whereby a genome stores the pattern-guidance needed to reproduce its phenotypic expression, and the modifications of the genetic representation are the source of evolutionary variation. This distinction - akin to Organisation Theory's traditional quest to account for the roles of both formal and informal organisational features (Simon 1947) - provides the basis for a theoretical characterisation of the different but complementary roles of SOPs and performances: while 'SOPs-as-representations' consist of formalised statements of what actions *should* occur, the actions occurring as routines are expressed *in context*.

[6] STS scholars' exploration of the distinction between formal procedures and rules and informal practices here is relevant. They have distinguished between 'mental plans' and 'situated action' (Suchman 1987, Hutchins 1991), 'representations of work' and 'practical action' (Suchman 1983), 'modus operandi' and 'opus operatum' (Bourdieu, 1977), 'espoused' and 'actual' practice (Orr, 1990), 'time-objective' and 'time-in-process', 'object-world' and 'process-world' (Bucciarelli, 1988), 'rules-as-represented' and 'rules-as-guides-in-practice' (Orr 1991 in Tsoukas 1996), 'canonical' and 'non-canonical' practice (Brown and Duguid, 1991), the 'map' and the 'terrain' (Berg 1997). Continuous ad-hoc 'articulation work' (Suchman 1983 & 1987) of actors is thus constantly required to bridge the 'phronetic gap' (Spender 1989, Taylor 1993) between the formula (or rule) and its enactment, the representation and the actual process. While these literatures focus on the *tensions* between the two categories of elements, they do not address their mutual shaping.

[7] Contributions inspired by this literature so far have failed to emphasise that the interactions between 'expression' and 'representations' are far from straightforward as they involve the radical reconfiguration of both the form and the content of routines (cf. D'Adderio 2003). The outcome of the complex, socio-technical process of translation from artefactual representation or code (Hutchins & Hazelhurst 1991), to a new expression-type routine, for example, may contain parts of representation-type routines as the boundaries separating them are often undefined.

regulate the mutual influence and adaptation between artefact-embedded rules, procedures, and performances?

2.1.1 The influence of artefacts and agencies on routines' evolution: description vs. prescription

When considering the influence of formal rules and SOPs over performances, we can identify two contrasting schools of thought: a "framing" view, which focuses on the power of objective structures to define, prescribe and frame actions; and an "overflowing" view, which focuses the ability by human agents to interpret, modify and in some cases fully override a rule or procedure.

The "framing view", often also referred to as the 'rationalist', 'cognitivist' or 'mechanistic' view (cf. Tsoukas 1996) is common to much technical (AI, computer science) and positivistic management literature (since Taylor's 1911 classic contribution). This view sees rules and procedures as fixed and objective representations which reflect accurately the human practices that they are designed to prescribe. According to this view, thus, formal procedures and rules, as reified, external representations, univocally decide the course of human practices: they are reproduced automatically and diffuse linearly. This understanding at the extreme sees rules-as-representations as casually operative, thus providing a distortion that mistakenly conflates the rule or formula with its enactment (cf. Bourdieu 1990 in Taylor 1993). Human actors are reduced to just "[…] cogs in the wheel of a larger technical system", rule-following automata that reproduce a fixed routine for a fixed outcome (Berg 1998:467). In this case, thus, there is no distance between the rule or procedure as-represented and the rule in effect: *the rule* or procedure *determines the practice*.

The "overflowing view" provides a critique to the mechanistic view of rules and procedures portrayed above. Where the framing tradition emphasises the purity of logic-driven, automatically reproducing rules, the 'overflowing' view emphasises by contrast the intrinsic complexity, variety and adaptability of human practices as well as the power of discretion by human actors to interpret, modify and even completely reject a rule or procedure. Widespread in Sociology, Ethnomethodology, and, increasingly, in

Organisation Studies, this view often portrays the ‘objectivist’ representations of procedures as external entities as selective, at best incomplete and at worst fundamentally flawed reproductions of “real” processes which they attempt to capture, imitate, guide or direct often with only partial success (Lynch 2001 in Schaztki et al.)[8]. As “[…] no representation of the skills involved in performing appropriate human activity can ever be adequate […] the ability of formulations to guide what people do rests on abilities to use and understand them” (ibid: 8-9).

Views animated by Wittgenstein’s philosophy (including early social constructivists and ethnomethodologists), for example, have highlighted how structures, including rules and classifications, are never deterministic but always interpreted (Garfinkel 1967, Barnes 1982, Bloor 1973 in Hatherly et al. 2007, Lynch 1992). The irreducible interpretive flexibility of rules is such that – at least in theory - can lead to infinite regression as “… no course of action could be determined by a rule, because every course of action can be made out to accord with the rule” (Wittgenstein 1967:81). Due to the logical under-determination of behaviour by rules, there can be no closure as “*The rule is, at any given time, what the practice has made it*” (Taylor 1993:57-8 *emphasis in original*)[9]. In other words, the *practice determines the rule*.

Organisational scholars have drawn on these approaches in attempting to conceptualise the role that technology plays as a source of constraint in rule-following (Barley 1986, Orlikowski 1992). Also in this case, however, the emphasis remains firmly on the *discretion* by human agents to follow a procedure or rule or “choose to do otherwise” (Giddens 1993, Orlikowski 1992). According to this framework, the properties embedded inside artefacts are never predetermined but rather “the capacity to modify the “rule” that is drawn on in any action is an ever present possibility” (Cassell 1993:13 in Orlikowski 2000). Rules in this view exist only virtually and are consequential only to the extent that they are enacted by users through practice. The inherent flexibility and

[8] The assumption here is that no representation of the skills involved in performing human activity can *ever* be adequate.
[9] The fundamental difference between the “social constructivist” and the “ethnomethodological” approaches is that the former acknowledge the role played by constraints - form the psychological to the sociological - in the process of attributing meanings to rules (Bloor 1997 in Hatherly et al. 2007). While falling short of listing technology among the constraints, constructivists nonetheless accept that they allow foreclosure in practice.

adaptability of human practices implies that rules may attempt to guide behaviour (Spender 1989) but human actors can *always* operate discretion in interpreting the rule or procedure, assign meanings (Daft & Weick 1984) and ultimately decide whether, how and when to abide to, work around them or altogether reject them[10].

To summarise, while the contribution of the “overflowing view” is fundamental in dispelling technological determinism in showing how rules are interpreted and enacted in the context of actual practice, they tend to overemphasise the power of the human agent’s discretion. At the extreme, rules and procedures become consequential - they have an effect over reality – only when interpreted and/or enacted by humans. Thus, while according to the framing view, formal SOPs and rules are mainly *prescriptive*, according to the overflowing view, they can become merely *descriptive* as actors can always choose to avoid them. By contrast it is our contention that, while *there is* always scope for human intervention, formal rules and procedures play a more fundamental influence on rule-following than it is allowed by the latter view. This is especially true when rules and procedures are embedded in technological artefacts.

2.1.2 Rule-following as distributed cognition and the role of artefacts

While, in rule-following as well as in other realms of human practice, artefacts do not determine actions, nevertheless they “plainly matter” (Hatherly et al. 2007). Technical systems, for example, make it possible or easy to do certain things, and impossible to do others, so that, while in theory there can be infinite regression “the logical open-endedness of the application of terms to particulars and the logical under-determination of behaviour by rules are foreclosed *in practice*” (ibid:11, *emphasis added*). STS authors, for example, have shown how a complex range of rules and assumptions as ‘scripts’ are embedded within technology both at the design and usage stage (Akrich

[10] Mambrey & Robinson, for example, have shown how, in a highly hierarchical and rational organisation, existing rigid procedures are often dismissed in favour of intricate, informal processes (1995 in D’Adderio 2004); and Wynne showed how operators of large technical systems often deviate from formal, rule-binding operating practices to deal with complex interdependencies, unexpected circumstances and local conditions (1998). Similarly, Gasser (1986) and Orr (1996 and 1998 in Berg 1997) have discussed how a computer’s formal representation of the work clashes continuously with the actual contingent and complex logic of human work. Since the formal tool embodies an impoverished version of work, humans working with the tool need to repair the tool’s functioning whenever it is used in practice (Collins 1990).

1992, Latour 1992, Grint & Woolgar 1992)[11]. This involves the socio-technical process of "inscription" (Latour 1992) by which dominant interests are reflected in the form and functioning of a technology. The existence of technology-embedded rules implies that focusing solely on human actors to explain rule-following practice is inadequate.

Rule-following, as depicted in Ethnomethodology and Cognitive Anthropology, is in fact a highly distributed set of knowledge and activities which stretches across a mutually supportive constellation of elements including material devices, language modes and representation modes (Lynch 1995 in Preda 2000). In his ethnography of the two different scientific practices of "opticism" and "digitality", for example, Lynch shows how rule-following depends on the relationships between material devices, theoretical optics, geometry and modes of graphic representation (ibid). Similarly, Hutchins (1995) showed how the process of piloting a ship in and out of a harbour is a complex, rule-determined activity involving not only the coordination of crew members, but also the use of navigational instruments and maps. A gyrocompass, for example, incorporates some essential rules of sea and land orientation and for this reason it requires that the pilots' skills and activities be adjusted to its properties (Hutchins 1991). Once embedded in artefacts, skills and tacit knowledge (Latour 1992), rules (Hutchins 1991, Preda 2000), and procedures (D'Adderio 2003, Hatherly et al. 2007) tend to become more *stable* and *durable* which holds radical implications for rule-following behaviour[12].

The role of software

In this paper we focus on a category of artefacts which is both very interesting and very relevant: software. Information systems, as bundles of inscriptions, play a fundamental role in influencing rule-following. Such systems are "neither merely neutral media nor simply means of increasing the efficiency of what unaided human beings might do" (Hatherly et al. 2007:32)[13]. They structure work, extend interactions, increase visibility

[11] The term 'script' is the ANT equivalent of the notion of rule in sociological inquiry (Preda 2000).
[12] Latour's notions of 'immutable mobiles' and 'obligatory points of passage' (1987) play an important role in this context, though one that has been neglected so far in the organisational literature that has focused on artefacts as flexible intermediaries or 'boundary objects' (Betchky 2003). In contrast, D'Adderio (2001) has shown how coordination across diverse organisational communities requires a mix of 'boundary objects' *and* 'standardising devices'.
[13] On the influence of information systems on structuring knowledge and work see D'Adderio (2003).

of knowledge and actions, create a common platform for the accumulation of common knowledge, constrain the ability of practitioners to alter the results of another, regulate who has access to making changes, track progress of changes, link multiple sites in different time/geographical locations, facilitate data sharing and the reception of feedback (Orlikowski 2002). They solidify and stabilise rules, procedures and classifications thus making it more difficult to avert them (Hatherly et al. 2007). While technical constraint is never absolute and indeed many system's controls can be subverted if sufficient resources and incentives are applied to the task, there are several reasons why the influence of technologies – in general – and information systems – in particular - is critical. While in theory it is always possible to bypass software-embedded controls, *in practice* this does not always occur.

A first reason is that assumptions, rules, procedures and classifications, embedded in software both at design and usage stages, tend to sink in and become invisible to users. In other words, they become part of the *habitual* background (Bourdieu 1977), or 'the way we do things around here' and as such are often unquestioned. Secondly, as distributed and pervasive technologies, information systems are often entangled into a *thick web of organisational relationships* which make them difficult to avoid. Once adopted, for example, software can influence what kind of information should be created, selected and shared, with whom, in which format and in what sequence. While practitioners can often choose to bypass the software, their boycott will hold consequences for them in terms of their ability to have their work taken into account by others in the organisation. Thirdly, software tends to make information more visible across an organisation thus making it easier to control that actions actually comply with the software. Fourthly and finally, while formal software controls can in theory be easily modified or averted, in practice this requires the deployment of resources (i.e. time and programming competences) which are often not available[14]. In these circumstances, the 'power of default' (Koch 1998 in D'Adderio 2003, Pollock & Cornford 2004) will prevent adaptation and customisation. The role of technological systems in influencing rule-following deserves thus to become a crucial topic.

[14] It is increasingly less the case as information systems grow in both scope and complexity.

2.2 From Representation to Performation

In our quest to characterise the influence of SOPs and rules over performances, we are therefore proposing a shift of emphasis from an objectified and detached view of rules and procedures as external objects that have fixed properties to a performative view where rule-following is characterised as a typically emergent, distributed and artefact-mediated activity. Such a shift does not however correspond to abandoning previous approaches: performation includes representation, but recognises that representation is only half the story. As ‘proxies of the ostensive’ or ‘representations’ SOPs and rules reflect - to an extent -abstract views, theories and imperatives. At the same time, however, SOPs and rules as statements are not alienated from but an intrinsic part of actual processes over which they exert substantial influence. Introducing performation allows us thus to explore the reciprocal influence of SOPs, rules and performances and can thus be read as “…an act of rebalancing” (Pickering 1994).

2.2.1 Relationship between theories and the reality they model: disentanglement, performativity and counter-performativity

The notion of performativity explains how theories and models are not simple descriptions of settings but they *transform* the settings that they describe. Economic theory, thus, according to Callon, does not simply *describe* but it *performs*, alters actual markets (1998, 1999). Similarly, in the context of industrial rule-following, SOPs and rules as models and theories of actual processes, alter the course of actual practices wherever they are introduced. In providing a framework to study the mutual adaptation of models of processes and actual processes, performativity theory thus can provide us with a novel and promising way to improve our characterisation of rules and routines’ dynamics. To explain what is meant by performativity and how this can help us in our quest to characterise the dynamics of routine- and rule-following we draw from the work of Callon and MacKenzie on the performativity of financial markets theory.

Building on the anthropologists’ notion of ‘entangled objects’ (Thomas 1991 in Callon 1998) economic sociologist Michel Callon portrays the construction of economic markets as involving processes of ‘disentangling’, ‘framing’ and ‘overflowing’ (1998; 1999). Callon starts from the premise that the market, as a method of coordination, implies the existence of agents capable of calculation. To make “calculativeness” possible, however, specific conditions must be put in place. If calculations are to be performed and completed the agents and goods involved must be ‘disentangled’ and ‘framed’. Framing involves the drawing of “…a clear and precise boundary between the relations which the agents will take into account and which will serve in their calculations and those which will be thrown out of the calculation” (1998:16).

Particular attention here is devoted to the role of tools, equipment and devices which contribute to the framing of transactions. Tools are mediators between the theory of economics and the economy: “not only are they responsible for the cross-relations between the two but, like any other mediator, they promote the construction and constitution of each of them” (Hennion 1993, ibid:28). Recalling Garcia’s (1986) example of the construction of the table strawberry market, Callon shows how tools and devices such as the display of transactions on the electronic board and the qualification of batches of strawberries on data slips were fundamental in giving the agent’s action shape, thus creating an arena or “space for calculability” (1999:191).

This first movement, or ‘framing’ identifies a process of convergence between the model and the economy. Framing, however, is *never complete*: any frame is necessarily subject to “overflowing” indicating a divergence between the model and the actual market. This is often followed by further “reframing” by which there is again convergence between the model and the actual markets. Economic theories and models are performed through these iterative cycles of framing, overflowing and reframing which regulate their mutual adaptation.

An important feature of a performative view is that economic models and theories are not ‘external’ to the market but an intrinsic part of it. There is in fact no real separation between ‘market models’ on one side, and ‘market practice’ on the other: market models are performed in practice. Models form a crucial part of markets, they are not purely detached external representations or virtual abstractions (cf. Miller 1998 in Holm 2002) but engines that make the markets tick. Similarly, while process models can be seen to an extent as blueprints of actual practices, or “the bit of sour dough that is used as a starter for the next loaf of bread" (Pentland & Feldman 2005b:5) they are not separate from those practices.

MacKenzie builds on Callon's notion of performativity with his study of the market for financial derivatives (2003, 2005 & 2006). He too shows how models are not simply a description of something resting outside the market (reality) but a constituent part of it. This is an important advance in our understanding the role of models: these are not just passive 'guiding principles', setting the boundaries of what can be done and what can't be done, as scholars have argued so far, but they contribute to shape actual processes.

MacKenzie's work is especially of interest in his finer grained identification of different categories of performativity or of influence of models on reality: "*generic performativity*", when an aspect of economics (a theory, model, concept, procedure, data set etc.) is simply used by participants in economic processes; "*effective performativity*", when the practical use of an aspect of economics has an effect on economic processes; "*Barnesian performativity*", when the practical use of an aspect of economics makes economic processes *more like* their depiction by economics; and "*counter-performativity*" when the practical use of an aspect of economics makes economic processes *less like* their depiction by economics. This classification is especially useful as it leads to a much finer grained and therefore insightful characterisation of the interactions between procedures, rules and performances, as argued later in this paper.

In particular, MacKenzie's framework highlights that, what previous theories considered the norm, are in fact often exceptions. At one extreme of performativity there is *prescription*. Prescription represents a very strong instance of performativity: automatic reproduction, pure repetition, no more recalcitrance, recurrent events (Sahlins, 1985 in Callon 2006). Full prescription thus corresponds to 'fiat lux et lux fuit', as in the case of an automatically reproduced sequence of computer algorithms. At this extreme, which corresponds to the 'framing view' outlined above, there is very little adaptation as models are automatically reproduced. At the other extreme, there is the full demise, rejection or disuse of a model or tool. This case corresponds to the ‘overflowing view’ outlined above: the influence of the model is so weak that it is bypassed, worked around or outright rejected and therefore is not enacted in practice. One way to explain the demise of a tool of course is that individual agents have made the conscious choice to reject the model. Performativity theory, however, while not denying this possibility, affords us a more interesting explanation: the model as statement has not been able to put into motion a world in which it can function. In other words, the statement or formula has not been able to produce a successful socio-technical *agencement*[15]. While full prescription and mere description are always possibilities, most of the time (and this is especially true in conditions of high uncertainty) there is performativity, implying some kind of dynamic adaptation between model and reality (Callon 2006).

In conclusion, we postulate that the performativity framework can be effectively harnessed to improve both our theoretical understanding and empirical characterisation of the interactions between procedures and performances. It will also provide new grounds to characterise the key role of artefacts and tools in general – and software in particular - in mediating these interactions.

3. Interactions between procedures and performances and the role of software

We thus focus our attention on software-embedded SOPs and observe their interactions with performances as they unravel in practice. For our analysis we have selected the “BoM freeze process”, a crucial segment of product development corresponding to the handover of a product configuration from Engineering to Production. In this context we examine the mutual adaptation between the computer-embedded freeze procedure (and related rules) and the actual process. Being so stable - and therefore easy to observe - the software procedure provides us with a useful device to capture the co-evolution of SOPs and performances.

[15] A socio-technical *agencement* is the assemblage of heterogeneous elements that is required for the world contained in the statement to be actualised: “A formula that progressively discovers its world and a world that is put into motion by the formula describing it” (MacKenzie 2003, in Callon 2006:19).

The fieldwork involved a one-and-a-half year ethnographic study based on participant observation at a leading automotive manufacturer. The data collection focused on a complex vehicle development programme – including one hundred vehicle variants - which was monitored for its entire duration (18 months). During this time the researcher was able to access the firm's facilities with daily frequency thanks to the provision of a contractor's badge. While being given a desk in the Advanced Technology and Product Development department (responsible for the implementation of enterprise software technologies), the badge allowed the researcher to circulate freely throughout the organisation. As part of the ATPD implementation team she was also invited to take part in company-wide seminars, workshops and away days which provided valuable opportunities to discuss findings and make informal acquaintance with potential interviewees from most organisational functions and levels.

Evidence gathering involved the direct observation of product development practices as they unravelled. It also involved recording practitioners' accounts of practices through in-depth semi-structured interviews with programme managers, directors, product and engineering administrators, industrial designers, marketing, sales and accounting personnel. Additional evidence was gathered through searching the company library and electronic archives, as well as scanning the many manuals, documents and databases that were created by the team and shared throughout the development process (up to manufacturing release). Less frequent update visits were carried out for a further period of 18 months following production. The combination of direct observation and interviews provided excellent grounds to observe the interplay of procedures and performances, as well as the role of software as an intermediary in these interactions.

4. The 'freeze' process

The evidence on which our analysis is based draws from the in-depth observation of the Bill of Materials (BoM) freeze process (from now on the "freeze process"), a critical sub-segment of product development. Our example focuses on the changes occurring to the freeze process following the introduction of Product Data Manager (PDM) software upstream, at the Engineering end of product development[16]. Here, the introduction of software provides a unique opportunity to capture the relationship between the formal freeze SOP (embedded in software) and the actual performances as enacted by practitioners in our automotive firm at the time of fieldwork.

4.1 The formal process

We begin with the detailed account of the freeze process as described by a senior engineering administrator and recorded by the researcher during observation[17]. Our engineer takes us through the complexities of the freeze process and the changes that are occurring as a consequence of software implementation. After highlighting the difference between the 'formal' and the 'actual' freeze process, our engineer begins to describe the formal freeze process, which is embedded in software as standardised 'best practice' (Fig.1).

The formal process starts with the definition of the vehicle specification. This is driven by the 'Test Plan' whereby testing requirements are optimised against the vehicle to be produced and its variant configurations. The 'Prototype Build Specification' (PBS) document is then drawn, which states the number of prototypes that will be required for a specific vehicle, or vehicle family. Next, the prototype vehicles characteristics are specified according to the 'Product Description Summary' (PDS) or marketing spec, which is a description of the standard vehicle to be produced. This can be driven by customer requirements, industry trends, benchmarking assessments, or by technology (i.e. testing) requirements. The PBS, or vehicle spec, is then passed on to Engineering Release Systems (ERS), where process administrators build a matrix-based tick-list, which identifies and lists all available vehicles options and their variants.

After this, the vehicle specification returns to Engineering where it is divided up into batches. Taking in consideration one vehicle batch at a time Engineering fills in the tick-list matrix by identifying which variants are to be included in each vehicle specification. Subsequently, ERS 'connects up' the variants to the vehicle. 'Connecting

[16] Product Data Manager (PDM) is a leading state-of-the-art, commercial off-the-shelf (COTS) enterprise software application for integrated design and manufacturing.
[17] On the use of narratives as instruments for empirical analysis see Orr 1990, Narduzzo et al. (2000).

up’ involves the creation of a structural relationship between the variants and the vehicle. The formal process at this point states that, once the volume of changes has diminished substantially, ERS should call the Bill of Materials freeze (the date after which only small changes are permitted) and proceed to create a number of individual configurations in the software’s “Manufacturing View”, one view for each vehicle variant configuration. This is where the formal and actual processes begin to diverge as we see in the practitioners’ account of the actual process.

4.2 The actual process

The proportion of uncertainty and change that an organisation faces varies substantially across different stages of Product Development (Fig.2). The early stages are characterised by high levels of change cycles and design iterations with product parts and assemblies undergoing very frequent and radical modifications. Changes at this stage are implemented by raising an “Engineering Change Order” and can be either initiated by a designer who is seeking to improve a part’s design or performance, or they may originate as a consequence of other changes affecting a related part. The iteration cycles continue until the product structure and parts are completely defined and require only minor (or no further) alterations. The SOP at this point dictates that Programme Management and ERS should call the ‘freeze’. The freeze implies that the product data and structure are sufficiently stable for the configuration to be released to production and manufacturing. Ideally, no more Change Orders should take place after the freeze milestone, because these would cause disruption to downstream development functions. While some flexibility is allowed during the early stages of development to enable Industrial Designers and Product Engineers to experiment by trial and error with different alternative solutions for product parts and assemblies, after the freeze, control is mandatory to allow for the stabilisation and validation of product definition, which is required for optimising Production tools and Manufacturing processes.

To support the control of changes after the freeze the software instigates the transfer of the product configuration from one single Engineering View (E/View) to multiple Manufacturing Views (M/Views) (Fig.3). The M/Views are generated by loading up individual product variant configurations on PDM. The difference between the E/View and M/View is that in the former changes must be implemented only once, to be automatically propagated wherever the part is used; this is because in the E/View there is only one comprehensive structure from which all vehicle variants can be derived. All vehicles stored in the M/View, instead, are different as each one corresponds to a specific vehicle variant; every change introduced after freeze must therefore be manually duplicated across each of the vehicle variant configurations, one at a time. This means that, from the freeze onwards, the vehicle configurations have to be maintained and modified individually, according to manufacturing change requests (or Deviations) as and when these come through. By taking control of the transfer of the product configuration from the E/V to the M/View therefore, software makes changes difficult and cumbersome to implement, effectively slowing down and making the process of change implementation inefficient, thus favouring stability. This is in fact the rationale for calling the 'freeze': to reduce the number of changes and the extent of their fluctuations therefore providing stability to downstream development functions. According to our engineering administrator, however, this is the point where the actual process begins to diverge from the ideal process. While - in theory - after the freeze the Release function should create one manufacturing view for each vehicle variant, the actual process is less straightforward, as the detailed observation of the X100 Vehicle Programme (ongoing at the time of fieldwork) illustrates.

The X100 is a complex vehicle programme due to the high number of vehicle variants planned and the related high level of data, assemblies and configurations that have to be created and maintained. By the time X100 has reached the freeze milestone deadline, many parts have not yet been released, and a huge volume of change is still required before any prototypes can be manufactured: "…At this stage we are still being deluged with a substantial number of Change Orders…" (Interview/MC). The process is still far from the nominal conditions required to call a freeze: the product structure is still highly unstable, the number of changes is high, and most of the changes required are still major changes. A project milestone date has been reached, where ERS must 'freeze' the Bill of Materials and move the product variants configurations from PDM's E/View to the M/View in order to facilitate the control of manufacturing deviations. The development team however, is still lagging behind, struggling to manage the enormous amount of data and assemblies that have been generated.

The production-oriented logic embedded in PDM at its design stage is intentionally devised to control and inhibit the introduction of changes after the freeze in order to stabilise the product configuration early to the benefit of Production and Manufacturing[18]. PDM, however, works on the assumption that no major changes are required after the freeze, a situation that does not often occur in practice, especially in complex development programmes such as the one analysed here. Given the substantial amount of Change Orders that are still required to the first batch of vehicles at the time of freeze, and following the realisation that the migration to the M/Views would make it much harder to implement later changes, upstream engineers decided not to create the M/Views as required by the formal process and rules embedded in software:

> Theoretically, engineers are not allowed to attribute a 'Batch-1' effectivity status to an Engineering Change Order, they have to do it on a Deviation. However, most of the deviation requests we are receiving today are saying: 'Please can you apply this Order into Batch-1'. Today we are beyond freeze date and yet most requests concern Orders with Batch-1 effectivity. It is clear that, if we can process that change into the E/View, as an ordinary Change Order, we need only to do it once. So that's the main reason why we kept in the Engineering View. So that's what we are doing today, we are processing the Order into E/View with Batch-1 effectivity (Interview/MC).

Overwhelmed by the amount of change, the team has decided to remain in the E/View. "…Today, although we are *after* the freeze, we are keeping everything in the Engineering View. In the E/View you only need to make a change once, and it is automatically reflected everywhere the part affected by the change is used" (ibid). Given that the deviations incoming for specific vehicles are so few at the time of the freeze, while the number of Change Orders is still very high, the decision to bypass the software embedded rule and keep everything in the Engineering View facilitates the implementation of changes. The workaround is completed by introducing a sub-procedure which consists in attaching a Deviation document to each modified part. Because the change is implemented in the E/View, every time they load data from this into any of the Manufacturing Views, or every time they look at the data in the E/View,

[18] The philosophy behind this reflects established industry trends that see increasing the speed of the development cycle while increasing the efficiency of data integration across development functions as major sources of competitive advantage in the cutthroat mature automotive market.

the Deviation document will always appear in association to the part. This way, the material specification Deviation associated with the part can always be visualised, even though the official manufacturing views have not yet been created.

The decision to work around the software rule, however, is not without consequences. The formal rule that imposes to load the variant configurations in PDM's M/View is aimed at enforcing greater control over individual changes and on the way these affect each individual configuration. M/Views, for example, require an engineer to specify exactly from which view to which the change is to be propagated. For example, PDM will create an error every time one tried to propagate a change to a configuration where the part is non-existent. The software makes it necessary to be very specific as to where each part affected by the change is used, and to which views a specific change is to be propagated: "This is due to the structure of PDM's logic, which is of an incremental nature, meaning that the software will allow only incremental changes and make any other type of changes very difficult to implement" (Interview/MC). The underlying philosophy of Product Manager and many other engineering control systems is in fact one of sequential incremental change:

> In PDM you have to talk in terms of 'increment', you have to specify exactly what it is that you are going to change. …In other words, you have a starting point today, and you can only change today's data. So you make an incremental change, which gives you a new starting point, then you can make a further incremental change, then a further incremental change, etc. but they are all based on what has happened before. That gives you very clear control over the changes that you then send to production (Interview/MC).

PDM operates a selective control action by allowing or disallowing specific actions and by facilitating or preventing specific changes to be implemented at different times during the development process. While the production-oriented philosophy embedded in PDM is aimed at providing better control and favour the implementation of changes, it is at times be perceived as an unnecessary constraint and source of rigidity. In our case, to avoid such rigidity, the software procedure is bypassed: the decision is taken to postpone the creation of the M/Views until the time when the number of Deviations that are required for a specific vehicle will effectively and substantially outweigh the volume of Engineering Orders.

Workarounds in computing and other technologies are well documented in the technical, CSCW[19] and STS literature (Gasser 1996, Pollock 2005). However, authors have not sufficiently emphasised that, while it is often possible to bypass the software-embedded rules, this holds important *systemic* implications at the level of the organisation. While working around a software procedure or rule may be feasible, it always entails a degree of disruption. In our case, the decision to bypass the rule by managing Deviations in the E/View will generate confusion later in the process, when engineers will try to propagate a change across different M/Views. Since the M/Views are modified independently, there would not be a common starting point for the incremental change; any further modifications would have to be done therefore on an *ad hoc* basis, which is time-consuming, complicated and error-prone. These were precisely the type of drawbacks that the software was introduced to eliminate. Eluding the software's sequential logic implies therefore duplication of work and a higher risk that subsequent changes will be implemented incorrectly:

> For example, a request arrived to ERS, asking to make a further change to an Order that had been made 'Batch-2 effective' for the under-bodies; the problem was that, according to the request, the further change had to be made 'Batch-1 effective'. Theoretically this would be impossible, because the other change was Batch-2 effective and one cannot backdate effectivities, in theory. The reason that the first change had been made Batch-2 effective in the first place, was that there were complex interrelationships of parts and effectivities; and it was decided that the easiest thing was to make the whole thing Batch-2 effective. But then all the affected items on the change became Batch-2 effective, and now they are prevented from making that change Batch-1 effective (Interview/MC).

These problems are generated by the complex interrelationships between parts (structural complexity) and between 'effectivities' of different kind (time and process complexity). "…You get a huge merry-go-round that you have to untangle" (ibid). The control action exerted by software represents one way to handle such complexity: rules and SOPs act as stabilising factors in the context of the unstable and disordered development process. Software rules attempt to discipline the process by imposing a sequential and ordered logic; they can help to ensure that the work undertaken on one product part is consistent and concurrent with work undertaken on related parts elsewhere in the organisation:

[19] Computer-Supported Cooperative Work.

> A classic example of a rule is that you cannot release revision D of a part until after you have released revision C. One of the engineers gave us a Deviation saying: "I want revision F of this part in the BoM". But what is released today is just revision C. He hasn't released D, he hasn't released E or F, but he wants revision F in the BoM. He has already sent information to the supplier, and the supplier is going to produce the part according to revision F specifications. The problem is that that part is related to other parts, and the other parts have not been released to a revision that matches that parts revision, so you end up with a very long chain of interrelationships that you have to resolve (Interview/MC).

PDM is conceived to introduce control and to clarify structural and process-time relationships. Nevertheless, there are circumstances where the action exerted by PDM is perceived as reducing flexibility: "PDM is introduced to control and discipline, but sometimes it gets in the way" (Interview/MC). A production-oriented system, PDM is designed to support data control and validation which is mandatory in the downstream Production and Manufacturing environments, as argued by a Manufacturing manager:

> "We need these BOMs; we need to schedule these BOMs into vehicles; and we need to have a material required date for each vehicle […] With that kind of [unreliable] scheduling, you just can't do just-in-time. So you schedule one MRD date, and that is actually the MRD date for vehicle 1. Everything else, [and] there is no time control. And of course we are talking about thousands of parts. And one vehicle every two weeks. You need a system to control that and you need people to operate that system (Interview/JK).

The PDM Project Manager shares a similar view:

> : "[…] lack of procedures breeds low quality due to lack of data integrity, history and process repeatability – [this is] not sustainable. PDM will provide an environment to support [controlled] procedures." (Interview/DA)

The Management and Manufacturing views are reflected in and supported by the software incremental change rule. However, the way they see the rule upstream in Engineering is quite different:

> That is a very good rule. But the problem is that we live in a very anarchic world, in development, there is very little time and there are very few resources. So it may be the case that even if the guy has not released his revision F of the part, he still wants to specify that he wants revision F in the vehicle (Interview/MC).

4.3 Mutual adaptation between formal and actual process

We have witnessed the unravelling of tensions between actual practices, on one hand, and the software procedure, on the other, which embodies assumptions and rules aimed at ordering and disciplining those practices. In this context, it appeared clear that the software's emphasis on control was often perceived as excessive for those functions that rely on flexibility in both process and product structures. This can result in practitioners working around the rule to restore flexibility. In other words, where the philosophy around which the software system is built and configured clashes with upstream engineering views of development, scope is created for deviations from the rule:

> A good example is this freeze. There will be a freeze. This is a rule, but then, in practice, this becomes only an ideal date by which it would be nice if everything would be out for these guys [read: programme management]. […] It is a schizophrenic world, everyone will tell you that there are rules, and that, of course, we must work according to the rules, but then they will immediately break those rules (Interview/MC).

Workarounds can favour the process of adaptation by reducing the dissonance between the worldviews of upstream (Industrial Design, Engineering) development functions, on one hand, and downstream (Production, Manufacturing) functions, as mediated by and embedded in software, on the other. We have also seen, however, that relaxing control by delaying the configuration freeze at the design stage is not without consequences. While excessive control can cause rigidity at the Engineering end, the lack of control becomes a liability in Production, where practitioners are faced with excessive change as well as having to work with delayed and unreliable data. While in fact at the Engineering end variation is highly desirable, in a Production environment control is mandatory, as one engineer convincingly put it: "…For our prototype we want to be able to control the anarchy. In production you definitely want sequential incremental change" (ibid). Excessive deviation from the rule would work against the very purpose for which software-embedded rules were introduced in the first place, that is avoiding duplication of efforts and facilitating change control, therefore providing Manufacturing with stable and reliable releases. One should:

> … Come back to the basic question: what is the reason for the freeze? The reason for the freeze is that Manufacturing is incapable of coping with the huge volume of change; so we are going to freeze it, we say: this spec is what we are going to build it to. And there will be a minimal amount of change after that time (ibid).

This view, shared by engineering and program management eventually prevails in this case. The fact that the rules and SOP are embedded in software, helps them to endure against the other agencies' views. Practitioners in Engineering were able to deviate from the course of the SOP but only to an extent, and for a limited period of time. Inscribing the SOP and rules in software has made them more visible and easier to enforce. In other words, once embedded in software, SOP and rules as statements have successfully managed to construct a world in which they can function. This world unravels as a result of the co-performation of competitive programs, in this case reflected by the Engineering vs. the Management and Manufacturing views of the world.

5. Discussion and Framework

This paper has addressed two main questions. First, how can we theorise the interactions between SOPs and rules on one hand and performances, on the other? And second, what is the role played by artefacts and agencies in mediating these interactions? Going back to and building on Callon and MacKenzie's notion of performativity we can now shed new light over the co-evolution of formal rules and SOPs (the model) with routines in context (the actual process).

5.1 The performativity of routines

In our example of the freeze process, formal routines, rules and SOPs act as a theory, a model, an abstracted version of the actual process. SOPs are produced via a process of articulation, codification and standardisation, through which actual processes are articulated or made explicit and 'disentangled' from the local. Our 'freeze' procedure is one such 'disembodied' process. In order to disentangle the SOP from the actual process, a precise boundary must be drawn between the actions and knowledge inputs that are allowed by the SOP and those that are disallowed. What is left out by this framing process (the overflow) is a testimony of the effortful work of codification and standardisation that made the relationships among people, parts of the artefact and process, and parts of the organisation visible, thus giving the SOP a role as 'guiding principle' for the development process. As a 'disentangled object', the SOP can provide a common reference point to coordinate heterogeneous knowledge and views across different communities (i.e. Engineering and Production), therefore helping to coordinate and direct actions.

Further, the processes of codification and standardisation allowed for the 'freeze routine' to be abstracted and embedded in software, thus making it visible and available to all functions that could hence coordinate and synchronise their work. Framing has made the freeze process more visible and predictable, by making relationships explicit (interactions, deliverables and deadlines) and by reducing ambiguity. The process of de-contextualisation, through codification and the consequent inscription of the process in software, has transformed the freeze process into something which is easier to describe, visualise, share, transfer and reproduce (at least in principle) ‘anywhere and anytime’ across diverse communities in multiple organisational locations. In other words, the process thus created was something more similar to a ‘standard’ (Bowker & Star 1999), a commodity (cf. Callon 1998) and could subsequently act as a reference ‘common model’ (D’Adderio 2001), a single ‘interpretative frame’ of central process (March & Simon 1958, Marengo 1996, West & Iansiti 2003). Our SOP was introduced with the specific intention to enforce and sustain global coordination and concurrency of work across heterogeneous development functions and teams. Once established, the SOP begun to serve as a ‘frame of reference’ able to reduce cognitive complexity (Simon 1947, Prietula & Augier 2005) as well as constraining and guiding divergent intentions, views and actions across the organisation.

We have seen, however, that such a strong control mechanism, whose influence was further reinforced by inscribing the SOP in software, was perceived as excessively rigid by upstream functions where the need for change was greater. As argued in earlier work (D’Adderio 2003), the process of codification and standardisation in the production and reproduction of routines and SOPs is never neutral - as economic diffusion theory would have it - but always performed according to one - or more - prevailing logics or rationales. It follows that our standard operating procedure is not merely a simplified version of the actual routine. The SOP in our example embodies a strong rationale, namely the Engineering Administrators, Programme and Project Management and Production and Manufacturing philosophy, at the expense of upstream functions (i.e. Industrial Design, Design Engineering).

This emphasis highlights potential sources of conflict amongst different agencies and competing performative programmes. One calculative agency can impose its own calculations and rules consequently forcing other agencies to engage in its own calculations, as in Callon's example of the notorious chess player: "[…] it is almost as if Kasparow […] had to start calculating his moves not by playing like Kasparow but by imagining himself in the computer's position, that is to say by borrowing from its algorithms and calculation rules […]" (46). The choice of a tool such as our software package can shift the balance in the competition between calculative agencies in favour of one agency or another; the probability of gain is in fact "[…] on the side of the agency […] whose tools enable it to perform, to make visible and to take into account the greatest number of relations and entities" (Callon 1998:44-5). It follows that "[…] Imposing the rules of the game can be done by imposing the tools in which these rules are incorporated" (46). In our example, the inflexibility of the software-embedded SOP whose philosophy clashed with the goals, views and resources that belonged to other functions, generated scope for overflowing, manifested as a deviation from the SOP. The deviation took the form of a workaround, which temporarily restored flexibility while preventing the freeze process from grinding to a halt.

The software procedure, thus, was only partially and temporarily bypassed. This was partly due to the fact that excessive variation from its course would have caused a loss of coordination and synchronisation among development functions and across these and the rest of the organisation – objectives sought after by both IT project management and PD programme management. The potential benefits of the software SOP in this case would not have been realised: global coordination would have been possible without the framing, which provided a common reference point about the states of product and process, recording possible actions, their timing and their expected outcomes. Analogously to Callon's 'space for calculability', our SOP acted as a standard, providing a common language, which enabled to "reduce heterogeneity" and "construct equivalence" (22).

Another reason why the SOP was not completely bypassed was the fact that, once embedded in software, controls are more invasive and pervasive and difficult to detect, modify or altogether remove. Modifying the code underlying the software would have in fact required resources such as time and complex programming experience that simply were not available. In addition, discarding the software-embedded rule would also have made the entire software philosophy void, invalidating the rationale underlying the expensive and resource-consuming software implementation programme. A process of reframing was thus instigated whereby high-level software-embedded rules were (mostly) held in place and abided to.

This analysis helps to clarify some crucial but subtle dynamics that have been so far overlooked within the routines debate. Of course people are free to interpret rules, and rules are never totally binding, as they imply irreducible margins of interpretation. At the same time, however, formal rules perform a function. As cogently argued by our engineer, ‘they are there for a reason’. Artefact-embedded rules and procedures constrain interpretation and shape subsequent action. Just as economic theory in MacKenzie’s and Callon’s examples is a constituent part of the market, our rules and SOPs are performative with respect to the actual ‘freeze’ process.

We can now relate each of MacKenzie’s (2005) performativity categories to our case evidence. We have shown that aspects of the SOP and related rules were actually used by engineers in the freeze process (the deadline, the configuration handover procedure) as in “generic performativity”. Further, aspects of our SOP and rules had an effect on the freeze process by instigating the emergence of a centralised, mostly visible, concurrent process (“effective performativity”). As a consequence, the practical use of an aspect of a rule or SOP made the freeze process more like its model (with the uploading of the ‘frozen’ configuration on the computer system) as in “Barnesian” performativity; in these cases the SOP was initially able to provide guidance and discipline the process, instigating *convergence* between the formal procedure and the actual performance; this continued up to the point where the SOP was perceived as being too rigid. At that point the actual process started to significantly *diverge* from the model; in other words, the practical use of an aspect of our rule and SOP actually made our freeze process *less* like its depiction by the SOP (the creation of the workaround), and its influence was therefore “counter-performative”. Finally, through the addition of a formal deviation document to the (unlawfully) revised part, the workaround was finally legitimised and embedded in the formal computer-managed SOP. Divergence was contained in this case by the prevailing management view that local drift would have offset the benefits of global coordination afforded by the SOP. The inscription of part of the informal workaround in software steered the process again towards *convergence*, but this time it was the SOP that was modified to resemble the actual process. This made the formal procedure more flexible than it had been initially.

We have thus come a full circle in understanding the performativity of SOPs and rules and the role of artefacts as intermediaries in these interactions. In doing so we have shown that formal, artefactual representations of routines (rules and SOPs) do not simply “guide” performances, as often argued in literature (cf. Blau 1955), but they are performed through iterative cycles of framing, overflowing and reframing.

5.2 Prescription, description, performativity, and the role of artefacts

We can now return to our earlier distinction between the mechanistic and interpretive schools of thought in characterising the influence of SOPs and rules. Our evidence demonstrates that both schools have placed themselves at the extreme ends of MacKenzie’s performativity chart (Fig.4).

At one extreme (represented by the mechanistic school of thought) is the view that procedures and rules completely prescribe actions. In interpreting the role of SOPS and rules as deterministic and equating actors to rule-following automata, this view focuses on the framing side of our performativity spectrum (see Fig.5). While we accept that SOPs and rules frame actions and viewpoints, we have seen that full prescription is a rare and extreme outcome which involves the absence of adaptation, no resistance and automatic reproduction.

At the other extreme is the agency-centred school that sees SOPs and rules as descriptive: a simplified copy of the actual process which they (often inadequately) attempt to mirror. This view, centred on overflowing, highlights the interpretive role of agencies which, in enacting rules are able to modify or completely dismiss them (rules in this case are counter- or non-performative) (Fig.5). While this is possible, it does not account for the argument that, by incorporating beliefs into material devices, algorithms, procedures and routines a model can have an effect "[…] even if those who use them are sceptical of the model's virtues, unaware of its details, or even ignorant of its very existence." (MacKenzie 2006: 19). While, thus, formal procedures and rules can always – in theory - be worked around and dismissed, in practice they often play a role. Especially when embedded in artefacts such as software, they become visible, pervasive, difficult to change or avoid, easier to enforce. While possible in theory, mechanistic prescription and full interpretive flexibility are in practice two extreme outcomes; to the extent that a rule is entangled in a web of tools and organisational relationships, some level of performativity is at play (Fig.4 & 5). Artefact-embedded SOPS and rules thus don't simply describe, don't often prescribe, they are performed. In other words, they are "engines, not cameras" MacKenzie (2006).

5.3 Performativity struggles and the influence of organisational communities

To conclude our discussion we want to say a few words over the role of distributed agencies in routines evolution. A performative view allows us to characterise the emergence of rules and routines as the outcome of performativity struggles among competing "*agencements*" that aim at constructing – in this case - the industrial freeze process in different manners. The more successful performative programmes are those that manage to enrol materials and tools to create a world in which they can function. In our case, Management and Manufacturing manage to enrol the software and thus to (at least partially) impose the culture and priorities that belong to their occupational communities, their idiosyncratic languages and worldviews (Galison 1999, D'Adderio 2001)[20]. They are thus able to do the classification and ordering of data and process and decide what is and what is not important. Once embedded in software, the SOP has become a very powerful statement: it is material, visible to all functions, spans across all relevant organisational boundaries and communities, it makes management intent clear and unequivocal, provides a means of comparing legitimate with illegitimate actions and viewpoints. The statement has managed to put its world into motion.

[20] For a discussion of the literature on organisational communities see also Brown & Duguid 1996, Cohendet & Llerena 2003.

Just like the uploading of the virtual product in software in D'Adderio (2001) and the tracing of price variation curves in Preda (2007), our software-embedded SOPs and rules, as inscriptions, "impose a principle of reality; they constitute an obligatory point of passage, a perfectly material reality to take into account […] They are articulated to socio-technical *agencements* that produce the traces that they use to inscribe the world in which they are participants and on which they will, in turn, make possible to act" (39-40). While thus our evidence could be interpreted as a mere clash between conflicting interests, there is more to it.

On one hand, organisational groups hold different views of the freeze routine. They also have different incentives and success criteria for the routine: for upstream engineers the criteria for success is the production of the best (most innovative, robust) design; for Management, success is about getting the product out the door on time; for Manufacturing it is about having a stable, well-tested product configuration; for PDM project management it is about getting every function to use the software and achieve greater control and speed in the development process; for engineering administration, success is about the seamless integration of upstream design and downstream manufacturing data.

If however, on the other hand, we reduced this complex reality to a mere clash between conflicting interests, we would fall short of capturing what is truly happening. To paraphrase Callon, the various organisational actors attempt to construct the world (socio-technical *agencement*) they believe to resemble their own assumptions, views and aims. Confrontation therefore takes place not simply between different *agencies* but different *worlds* that are struggling to exist, one at the expense of the other (ibid 2006). The result of these struggles is that often none of the actors are able to take their program to its conclusion, since no one function is able to exclusively frame the engineering freeze process. Each has compromise and accept (partially at least) the others' programme, meaning that only parts of their world are realised.

In our example, programme and IT management and manufacturing partially succeed in disciplining actions according to the software freeze rule - therefore enforcing the deadline - and yet engineering retain some of their discretion to make substantial and late alterations. Thus the world that ended up existing was a compromise, a patchwork containing elements from competing worlds. Each of the agencies thus managed to exert some influence over the overall process: in the end, as Callon as argued, the losers in such a complex situation are only the worlds that are excluded – or exclude themselves - a priori. By deciding to retain their legacy software, and not engage into the organisation-wide PDM implementation, for example, Industrial Designers find that it has become increasingly hard for them to bring their input, views and requirements to bear upon the rest of the development process.

6. Conclusions

Shifting the emphasis from routines as undifferentiated monolithic 'objects' to routines as generative - and continuously emerging - systems characterised by internal structures and dynamics provides promising new grounds for exploring some of the most relevant but as yet under-researched questions about the nature and dynamics of routines. These include the dynamics of interaction between aspects of routines and the role of artefacts and agencies in shaping these dynamics. This paper has made a contribution towards filling these important gaps in our understanding of routines dynamics. The combined novel empirical focus on artefacts and theoretical framework based on new developments in Economic Sociology and Sociology of Finance have provided important new insights into the mutual adaptation between aspects of routines and the role of artefacts and other agencies as intermediaries. This approach has allowed us to move beyond fully prescriptive, normative approaches, on one hand, and simply descriptive, interpretive approaches, on the other. In accordance with the most recent advances in economic sociology we have thus shown that prescription is an extreme type of performation, and the interpretive flexibility of social actors is not absolute as worldviews and theories – especially when embedded in artefacts – do play a role. Specifically, our framework has provided three main contributions to the routines debate.

First, we were able to capture the micro dynamics of interaction between different aspects of routines, namely, artefactual representations of routines (SOPs, formal rules) and actual performances. Drawing from recent theoretical developments within the field of STS, we have theorised routines evolution and adaptation as the emergent result of iterative cycles of framing (selective retention), overflowing (variation) and reframing (selective retention) by which SOPs and rules are performed[21]. This framework helps us understand how a routine's stable pattern emerges out of the mutual cycles of adjustment (convergence and divergence) among these elements and the competitive arrangements in which it is stabilised. Such stability, however, is constantly put into question. So, while from a distance the routine might look the same, in reality it is continually changing, "[…] tuned and retuned in the struggles in fields of agency that the performative idiom thematises" (Pickering 1994:415). As our evidence suggests, such struggles between agencies underpin the dynamics of convergence and divergence as well as the intensity of interactions among routines' constituent parts.

Second, we have characterised how artefacts as intermediaries shape the interactions between different sides of routines. In contrast with much extant literature that sees SOPs and rules as either imperfect process representations that can be easily dismissed/disused or as prescriptions that are compulsively and automatically reproduced, we have shown that – once they are embedded in a web of technological artefacts and organisational relationships - there is some kind of adaptation. Therefore while, of course, rules do not suggest their own correct application, and in certain cases actors are able to enact their own interpretation of the rule, on the other rules – especially when embedded in artefacts such as software – do have an influence. In other words, while SOPs and rules can be – in extreme cases - fully descriptive (a passive, fixed representation of the actual process) or fully prescriptive (they univocally order and structure the process), mostly they are performed.

Here the distinction between performativity and prescription – while a matter of degree – becomes relevant: performation refers to uncertain situations where there is dynamic adaptation, while prescription refers to automatic reproduction and pure repetition. This

[21] This view is aligned with the evolutionary framework where endogenous change emerges from cycles of interactions between *performances or expressions* (variation) and coded artefacts or *representations* such as SOPs and rules (selective retention) (Cohen et al. 1996). This is compatible with but different from Feldman & Pentland's framework where it is *performances* that provide variations which are selectively retained in the *ostensive* aspect of the routine (see Becker et al. 2005 for a discussion).

represents an extreme case where the socio-technical *agencements* and the worlds corresponding to its models have been realised, and there is therefore no more recalcitrance, roles are performed automatically, and events are recurrent. When this type of adaptation occurs, the performation comes to resemble a prescription. This situation, however, is quite different from our case, and indeed from most firms that use and produce complex technologies, whose operations are distributed across diverse organisations and work communities, and who are faced with uncertain environments and subjected to rapidly changing innovation regimes[22]. These are cases where counter-performativity prevails and existing *agencements* have to be rearranged or even profoundly transformed in order to become successful.

This takes us to the third issue dealt with in this paper that is the role and influence of heterogeneous, distributed agencies such as occupational communities and CoPs (Lave & Wenger 1991) on routines' evolution and adaptation. In analysing how abstract views of routines become embedded in artefacts, we were able to account for the fundamental role of distributed, and often conflicting, agencies in shaping routines[23]. This entailed an important shift of emphasis from the existing paradigm where routines constitute the structure and individuals the agency to socio-technical *agencements* which involve at the same time people and artefacts, material and non-material elements. We have shown how, in the struggle between competitive performative programmes, some agencies are able to inscribe their own worldviews in artefacts. These agencies are the most likely to succeed in exerting their own influence: enrolling artefacts tend to create stronger *agencements* that are more stable, interconnected into the web of organisational relationships and therefore more difficult to oppose. Rule-following is indeed a form of distributed cognition.

In our quest to unravel routines' internal dynamics we have set our focus on artefacts – and, specifically, on SOPs and rules – as starting points for our analysis. Artefacts such as written rules and procedures – especially when embedded in software as in our case –

[22] In these cases, "it is more difficult for performations to produce regularities and repetition as they are constantly faced with unexpected events that they sometimes absorb, but *only sometimes, for a while*" (Callon 2006: 61-2, *emphasis added*).
[23] The topic of routines and governance has been largely neglected in the routines literature since Nelson & Winter (1982) and Coriat & Dosi (1994)'s pioneering contributions.

have proven to be useful standpoints to observe the ostensive (abstract) aspects of routines with respect to which they can serve as indicators or "proxies" (Feldman & Pentland 2005b). We have thus embraced a *pragmatic* view of meanings and understandings which sees these as not simply residing "in people's heads" but as distributed across a thick organisational web including people, everyday artefacts, tools and procedures. For this reason, neglecting to include tools and artefacts in the study of routines dynamics can only provide at best a partial picture. So where existing literature has rightly emphasised the individual agent's ability to 'turn exceptions into rules' (Feldman & Pentland 2003:110), we have shown that they can do so only to the extent that they are able to construct a successful *agencement*, which in turn often entails enrolling tools and technological artefacts.

Indeed, as MacKenzie has shown with his example of the Merton Scholes formula - while the alignment of beliefs, views and intentions can indeed work for a while, these tend to provide temporary arrangements unless they are able to create a world in which they can function (2003). This is done by enrolling a 'principle of reality', which is often achieved through the involvement of material objects and artefacts[24]. We can therefore argue that (stable) SOPs and rules emerge not so much as the mere result of beliefs alignment but as the emergent outcome of competing *agencements* some of which are more and some less able to enrol materials and therefore are more or less successful. In this sense, the fact that a procedure or rule 'works' is the result - and not the premise - of successful performation, a formula that – over time – has been able to create the world in which it can function and therefore now encounters little or no resistance.

Performativity struggles between competing *agencements* lead to their mutual adjustment involving the (temporary) predominance of strong programme, or the emergence of a new programme from the coexistence/assemblage of different ones. The resulting stability is indeed similar to Nelson and Winter's notion of 'truce' (1982) – in its inception as a continuously challenged and emergent achievement – but here we can

[24] Feldman's (2000) example of the failure to implement a new routine for university housing can be given in this light a complementary reading as a failure to involve people and artefacts to create a successful *agencement* that brings a procedure to life and keeps it operative.

see clearly what are the forces at play that are responsible for stabilising or destabilising the routine. By means of a new framework that builds on Performativity Theory's set of notions and constructs and a fine-grained analysis based on ethnographical data we were thus able to achieve valuable progress towards an improved characterisation of routines' dynamics and the fundamental influence of artefacts and agencies on their evolution.

References

Akrich, M., 1992. "The De-scription of Technical Objects", in: Bijker, W.E. and Law J. (eds.), 1992. *Shaping Technology/Building Society - Studies in Sociotechnical Change*. Cambridge (MA): MIT Press.

Barley, S., 1986. "Technology as an Occasion for Structuring: Evidence from Observation of CT Scanners and the Social Order of Radiology Departments," *Administratiue Science Quarterly*, 31: 78-108.

Barnes, B., 1982. *T.S. Kuhn and Social Science*. London and Basingstoke: Macmillan.

Bechky, B.A., 2003. "Sharing Meaning Across Occupational Communities: The Transformation of Understanding on a Production Floor", *Organization Science* 14(3):312-330.

Becker, M.C., Lazaric, N., Nelson, R.R., Winter, S.G., 2005. "Applying organizational routines in understanding organizational change". *Industrial and Corporate Change* 14(5): 775-791.

Berg, M., 1997. "Of Forms, Containers, and the Electronic Medical Record: Some Tools for a Sociology of the Formal". *Science, Technology and Human Values* 22(4): 403-433.

Berg, M., 1998. "The Politics of Technology: On Bringing Social Theory into Technological Design". *Science, Technology and Human Values* 23: 456-490.

Bourdieu, P., 1977. *Outline of a theory of practice*. Cambridge: Cambridge University Press.

Bowker, G.C., Star, S.L. 1999. *Sorting things out : classification and its consequences*. Cambridge (MA): MIT Press.

Blau, P., 1955. *The Dynamics of Bureaucracy*. Chicago (IL): University of Chicago Press.

Brown, J. S., Duguid, P., 1991. "Learning and Communities-of- Practice: Toward a Unified View of Working, Learning, and Innovation", in Cohen, M.D., Sproull, L.S., *Organizational Learning*. London: Sage.

Bucciarelli, L. L., 1988. "Engineering Design Process", in: Dubinskas, F.A. (ed), Making Time: *Ethnographies of High-Technology Organisations*, Philadelphia: Temple University Press, Ch.3: 92-122.

Callon, M., 1998. "An essay on framing and overflowing: economic externalities revisited by sociology", in Callon, M., (Eds.), *The Laws of the Markets*, Oxford: Blackwell, 244-269.

Callon, M., 1999. "Actor-network theory – the market test", *The Editorial Board of the Sociological Review*, 181-195.

Callon, M., 2006. "What does it mean to say that economics is performative?" Forthcoming in: D. MacKenzie, F. Muniesa, L. Siu (Eds.), 2007. *Do Economists Make Markets? On the Performativity of Economics*, Princeton University Press.

Cohen, M.D., Burkhart, R., Dosi, G., Egidi, M., Marengo, L., Warglien, M., Winter, S., 1996. "Routines and Other Recurring Patterns of Organisations: Contemporary Research Issues". IIASA Working Paper, March 1996.

Cohendet, P., Llerena, P., 2003. “Routines and incentives: the role of communities in the firm”. *Industrial and Corporate Change* 12, 271-297.
Collins, H.M., 1990. *Artificial experts. Social knowledge and intelligent machines*. Cambridge(MA): MIT Press.
Coriat, B., Dosi, G., 1994. “Learning How to Govern and Learning How to Solve Problems: On the Co-evolution of Competences, Conflicts and Organisational Routines”. Prepared for the *Prince Bertil Symposium*, Stockholm School of Economics, June 1994.
Cyert, R., March, J., 1963. *A behavioural theory of the firm*. 2nd edition, Cambridge (MA): Blackwell.
D’Adderio, L., 2001. “Crafting the Virtual Prototype: How Firms Integrate Knowledge and Capabilities Across Organisational Boundaries”. *Research Policy* 30(9), 1409-1424.
D’Adderio, L., 2003. “Configuring software, reconfiguring memories: the influence of integrated systems on the reproduction of knowledge and routines”. *Industrial and Corporate Change* 12(3), 321-350.
D'Adderio, L., 2004. *Inside the Virtual Product: How Organisations Create Knowledge Through Software*. Cheltenham (UK) and Northampton (USA): Edward Elgar.
Daft, R.L., Weick, K.E., 1984. “Toward a Model of Organizations as Interpretation Systems”. *The Academy of Management Review* 9(2): 284-295.
Eisenhardt, K.M., 1989. “Building Theories from Case Study Research”. *Academy of Management Review* 14(4), 532-550.
Feldman, M.S., 2000. “Organizational Routines as a Source of Continuous Change”. *Organization Science* 11(6): 611-629.
Feldman, M.S., 2003. “A performative perspective on stability and change in organizational routines”. *Industrial and Corporate Change* 12(4), 727-752.
Feldman, M.S., Pentland B.T., 2003. “Reconceptualizing Organizational Routines as a Source of Flexibility and Change”. *Administrative Science Quarterly* 48: 94-118.
Galison, P. 1999. “Trading zone: Coordinating action and belief”, M. Biagioli, ed. *The Science Studies Reader,* New York: Routledge, 137–160.
Garfinkel, H., 1967. *Studies in Ethnomethodology,* New York: Prentice-Hall.
Gasser, L. 1986. “The integration of computing and routine work”. *ACM Transactions on Office Information Systems* 4:257-270.
Giddens 1993. *New Rules of Sociological Method,* Cambridge (UK): Politi Press.
Glaser, B., Strauss, A., 1967. *The Discovery of Grounded Theory: Strategies of Qualitative Research*. London: Wiedenfeld and Nicholson.
Grint, K., Woolgar, S., 1992. “Computers, guns, and roses: What’s social about being shot?” *Science, Technology and Human Values* 17(3): 366-380.
Hatherly, D., Leung, D., MacKenzie D., 2007. “The Finitist Accountant”. The University of Edinburgh, April 2007.
Holm, P., 2002. “Which way is up on Callon? A review of a review: Daniel Miller’s “Turning Callon the right way up.” Norwegian College of Fishery Science, Tromsø, Norway.
Hutchins, E., 1991. “Organizing work by adaptation”. *Organization Science* 2(1):14-39.
Hutchins, E., Hazelhurst, B., 1991. “Learning in the Cultural Process”, in Langton et al. (eds), 1991. *Artificial Life II: Studies in the Sciences of Complexity*. Reading (MA): Addison-Wesley.
Hutchins, E., 1995. *Cognition in the wild*. Cambridge (MA): MIT Press.
Latour, B., 1987. *Science in action*. Cambridge (MA): Harvard University Press.
Latour, B. 1992. “Where are the missing masses? The sociology of a few mundane artefacts”. W. E. Bijker, J. Law (Eds.) *Shaping Technology/Building Society: Studies in Sociotechnical Change.* Cambridge (MA): MIT Press, 225-258.

Lazaric, N., Denis, B. 2001. “How and why routines change: some lessons from the articulation of knowledge with ISO 9002”, *Economies et Societes*, 6.

Lynch, M., 1992. "Extending Wittgenstein: The Pivotal Move from Epistemology to the Sociology of Science." in A. Pickering (ed), *Science as Practice and Culture*, Chicago: Chicago University Press, 215-265.

MacKenzie, D., 2003. “An Equation and Its Worlds: Bricolage, Exemplars, Disunity and Performativity in Financial Economics”.*Social Studies of Science* 33(6)831-868.

MacKenzie, D., 2005. “Is Economics Performative? Option Theory and the Construction of Derivatives Markets”. Presented to the annual meeting of the History of Economics Society, Tacoma,WA, June 25, 2005.

MacKenzie, D., 2006. *An Engine, not a camera: How Financial Models Shape Markets*. Cambridge (MA): MIT Press.

March, J., Simon, H., 1958. *Organizations*. New York (NY): Wiley.

Marengo, L., 1996. “Structure, competencies and learning in an adaptive model of the firm”, in: Dosi, G., Malerba, F. (Eds.), *Organization and Strategy in the Evolution of the Enterprise*, London: MacMillan.

Narduzzo A., Rocco E., Warglien M., 2000. “Talking about routines in the field”, in Dosi. G. Nelson R.R., Winter S.G. (Eds.). *The Nature and Dynamics of Organizational Capabilities*. New York: Oxford University Press.

Nelson, R.R., Winter, S.G., 1982. *An Evolutionary Theory of Economic Change*. Cambridge (MA): Belknap Press.

Orlikowski, W.J., 1992. “The Duality of Technology: Rethinking the Concept of Technology in Organizations” *Organization Science* 3(3): 398-427.

Orlikowski, W.J., 2000. “Using Technology and Constituting Structures: A Practice Lens for Studying Technology in Organizations” *Organization Science* 11(4): 404-428.

Orlikowski, W.J., 2002. “Knowing in Practice: Enacting a Collective Capability in Distributed Organizing”. *Organization Science* 13(3): 249-273.

Orr, JE, 1990. "Sharing Knowledge, Celebrating Identity: War Stories and Community Memory in a Service Culture," in D. S. Middleton, D. Edwards (Eds.), *Collective Remembering: Memory in Society*, Beverley Hills (CA): Sage Publications.

Pentland, B.T., Feldman, M.S., 2005(a). “Organizational routines as a unit of analysis”. *Industrial and Corporate Change* 14(5), 793-815.

Pentland, B.T., Feldman, M.S., 2005(b). “Designing Routines: Artifacts in Support of Generative Systems” Presented in Sofia Antipolis, France, 21-2 January 2005.

Pentland, B. T., Rueter, H. H., 1994. “Organizational Routines as Grammars of Action”. Administrative Science Quarterly 39: 484-510.

Pickering, A., 1994. “After Representation: Science Studies in the Performative Idiom”, *Proceedings of the Biennial Meeting of the PSA* 1994(2): 413-9.

Pollock, N., Cornford, J., 2004. “ERP systems and the university as a “unique” organisation”. *Information Technology and People* 17(1), 31-52.

Pollock, N., 2005. “When is a work-around? Conflict and negotiation in computer systems development”. *Science, Technology and Human Values* 30(4), 1-19.

Preda, A., 2000. “Order with Things? Humans, Artifacts, and the Sociological Problem of Rule-Following". *Journal for the Theory of Social Behaviour* 30: 269-298.

Preda, A. 2007. "Where Do Analysts Come From? The Case of Financial Chartism" in M. Callon, Y. Millo, and F. Muniesa (Eds.), *Market Devices*. Oxford: Blackwell.

Prietula, M., Augier, M., 2005. “Adventures in Software Archeology: Seeking (ABTOF) Theory in the Code”. Working Paper, Goizueta Business School, Emory University, Atlanta (GA).

Schatzki T. R., K. Knorr Cetina, E. von Savigny (Eds.) 2001. *The Practice Turn in Contemporan Theory*. London: Routledge.
Simon, H., 1947. *Administrative behavior*. New York (NY): Macmillan.
Suchman, L.A., 1983. "Office procedure as practical action: models of work and system design". *ACM Transactions on Office Information Systems* 1(4): 320-328.
Suchman, L.A., 1987. *Plans and Situated Action: The Problem of Human-Machine Communication*. Cambridge (MA): Cambridge University Press.
Spender, J.C. 1989. *Industry Recipes.* Oxford: Blackwell
Taylor, C., 1993. "To Follow A Rule". In Schatzki et al (eds.) 2001.
Taylor, F.W., 1911. T*he Principles of Scientific Management*. New York: Harper.
Tsoukas, H. 1996. "The firm as a distributed knowledge system: A constructionist approach". *Strategic Management Journal* 17: 11-25.
West, J., Iansiti, M., 2003. "Experience, experimentation, and the accumulation of knowledge: the evolution of R&D in the semiconductor industry". *Research Policy* 32: 809-825.
Wittgenstein, L., 1967. *Philosophical Investigations*. Oxford: Blackwell.
Wynne, B., 1988. "Unruly Technology: Practical Rules, Impractical Discourses and Public Understanding," *Social Studies of Science* 18: 147-167.
Zollo, M., Winter, S.G., 2002. "Deliberate Learning and the Evolution of Dynamic Capabilities", *Organization Science* 13: 339-351.

List of figures

Figure 1. The Prototype Bill of Materials (BoM) Process

Source: Interview/MC

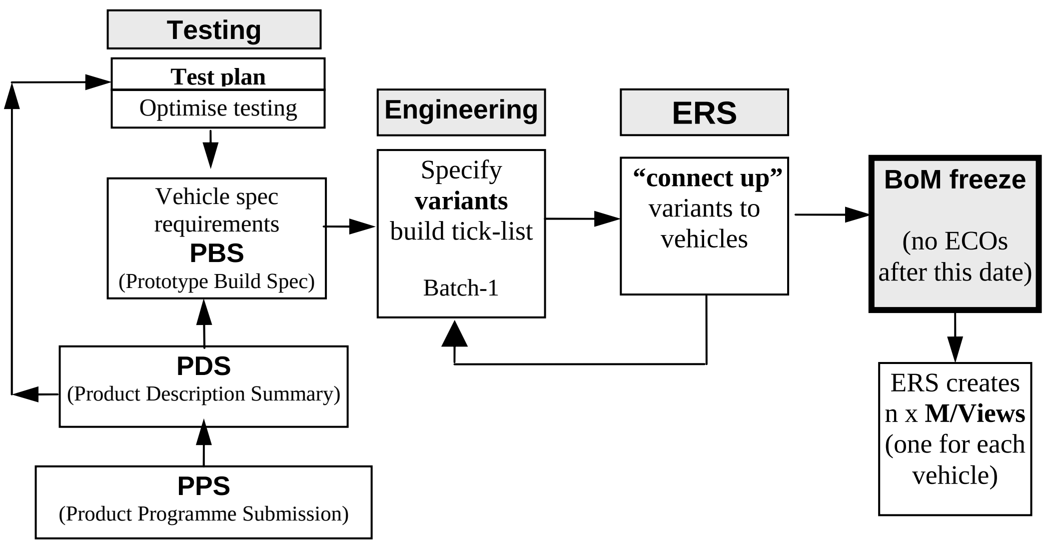

Figure 2. Incidence of changes along the product development life cycle.

Source: Interview/NG

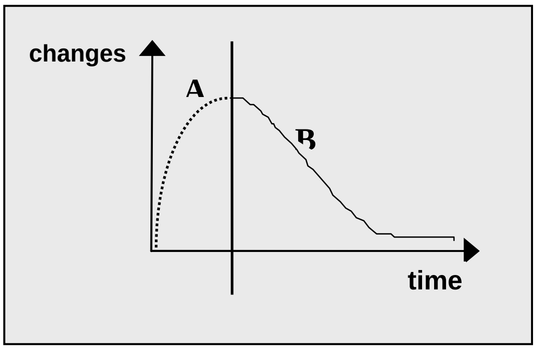

Figure 3. 'Request to Change' documents

Source: Interview/MC

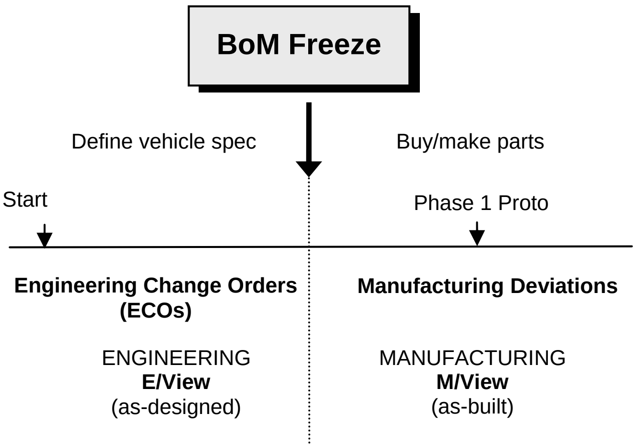

Figure 4. Degrees of Performativity

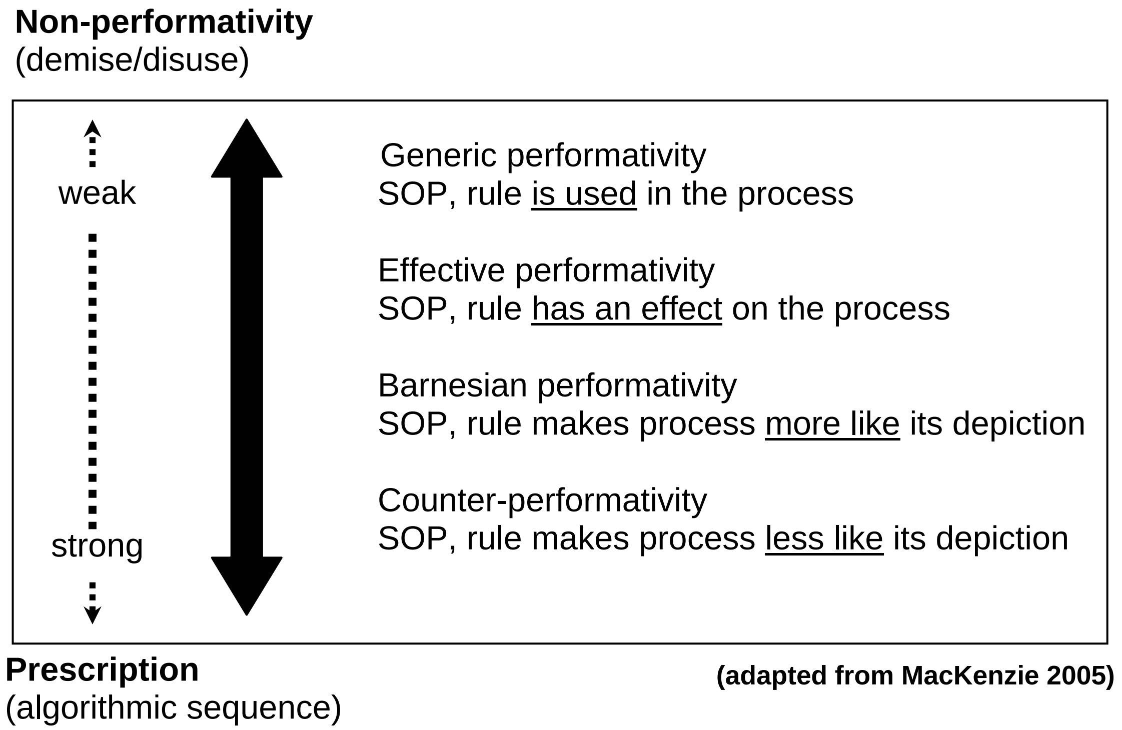

Figure 5. Theoretical framework