The most revolutionary consequence of the Internet’s power, scope and usefulness may not be that computers start to think like us but that we will come to think like computers. Our consciousness will thin out, flatten, as our minds are trained, link by link, to “do this with what you find here and go there with the result”. The artificial intelligence we are creating may turn out to be our own (Carr, 2008: 229)

1. INTRODUCTION

The entry of computer-based artefacts into every walk of life is having a merging effect between the hard and the soft sciences. The new bridge between the two camps comes from the new capability that humans have to represent large chunks of reality in terms of information. Science, including social science, is increasingly about the collection, organization and transformation of information. It is argued that in the 21st century applied computer science is playing the role which mathematics played between the 17th and the 20th centuries, that is, providing an “orderly, formal framework and exploratory apparatus for other sciences” (Foster, 2006: 419).

A new wave of research with a strong emphasis on the need to integrate knowledge from different fields and paradigms has caused a number of fresh issues to surface. One of these issues is Organizational Design and Engineering (ODE), a multidisciplinary research project born at the Department of Computer Science and Engineering of the Instituto Superior Técnico in Lisbon, Portugal and now established at the Centre for Organizational Design and Engineering (CODE). ODE is defined as the application of social science and computer science research and practice to the study and implementation of new organizational designs, including the integrated structuring, modelling, development and deployment of artefacts and people.

Why organizational design and engineering? No matter how positivist or deterministic this expression might sound, organizational engineering is a fact of organizational life. Increasingly, the flow of work in organization is being shaped by engineering forces emerging from the information revolution. For example, it is safe to say that the very large majority of large and many medium-size organizations around the world have adopted ERP (Enterprise Resource Planning) systems. ERP packages like SAP, Oracle Applications, PeopleSoft or Navision make a lot of sense in managerial terms since they relieve managers of the vague and unreliable nature of small-scale, in-house software development.

However, behind each ERP there is a great deal of organizational engineering, in the form of an idealized model of the enterprise’s workflows. Each ERP module is built on the
assumption that the company will have processes organized in a certain way and regardless of the amount of customization carried out major organizational redesign is always required. The cases of mismatch between technical requirements and organizational practices are well documented, but after a period of misunderstandings, conflict and trial-and-error eventually the organization settles down at a different stage of its life cycle. What this means is that the engineering which is behind much of the technology that finds its way into the organization is not matched by the design or the rules of interaction within the organization. This is just one example, among many possible, of the need to bring together the engineering and the design of organizations.

On the other hand, the pervasive effects of computer-based information systems are having a large impact on the form and effectiveness of organizational designs through a host of new capabilities in the coordination and control of organizational processes (De Sanctis and Fulk, 1999; Weick, 2001; Malone et al, 2003), competence management (Hoogervorst et al, 2002; Lindgren et al, 2004) strategic alignment (Chan, 2002) or boundary spanning mechanisms (Pawlowski and Robey, 2004; Levina and Vaast, 2005). This body of literature appears against a broader background of intellectual endeavour which brings together a variety of hard and soft characteristics of organization, under the banner of knowledge creation and dynamic capabilities (Kogut and Zander, 1992; Teece, 1998; Kogut, 2000; Eisenhardt and Martin, 2000). ODE is set against this general background.

So, what drives us to write a White Paper on the design and engineering in organizations? Although very active in the last three decades, we feel that research in the field of information systems has not yet brought many of the answers we need for a true integration of the social and technological architectures in organizations. There are various reasons for this. One of them is the fact that information systems research has focused almost entirely on the analysis while distancing itself from anything to do with the building and implementation of computer-based artefacts. The building has been left to the engineers who, in their large majority, do not read the results of information systems research. Another reason is that in higher education bridges are still to be built between design and engineering in organizations. Business schools know very little about what is going on in the real world of computer-based artefacts and schools of computer science have no interest in the developments in organization theory when 90% of the artefacts they build are indeed organizational artefacts.

We believe that a new approach is needed, which takes into account the latest thinking in organization and information systems research, but which is also very focused on the building and deployment of computer-based artefacts. It is only through the changes in behaviour (i.e. action and performance) brought about by the use of artefacts that changes
in organizational design can be observed and theorized. If this approach were to be adopted by both the organization/information systems community and the engineering community, then we might finally start to have truly integrated computer-based artefacts in organizations. Our vision of ODE is sketched in Figure 1.

Figure 1

In the sections that follow we go through the thinking which has led to this diagram, then put forward some of the key propositions which, in our view, characterize ODE. First we provide a very brief overview of organization design as well as of the field of engineering where it might be relevant to organization design. Next, an exposition is presented about the key issues in ODE, namely: (i) organizational knowledge creation as the raison d’être for ODE, (ii) socio-materiality as the favoured epistemological approach, (iii) the engineering of computer-based artefacts as the key technological driver behind ODE and (iv) requirements engineering and agile methodologies as topics of particular relevance for the building up of ODE. We conclude with a set of working propositions which hopefully will inspire further research and development into ODE.

2. ORGANIZATION DESIGN

There have been a number of approaches to organization design. In his classic *Designing Organizations*, Sadler (1998) claims that organization design is about the *foundations for excellence*, the book’s sub-title. Daft (2007) emphasizes the artificial nature of
organizations by describing them as designed “goal-directed social entities deliberately structured activity systems linked to the external environment” (Ibid, p. 11). Hatch and Cunliff (2006) argue that organizational design defines the rules of interaction, i.e. the relationships among people who assume the roles prescribed by the organization, and the relationships of organizational groups or units to which they belong. Stanford (2007) explains organizational design as a means of holistic thinking about the organization, i.e. its systems, structures, people, performance measures, processes and culture, and the way the whole operates in the environment.

The holistic approach to organization design has been followed by several authors, one of the most interesting being Galbraith (1973) and his assertion that most work activity involves information processing. Individuals talk, read, write, calculate, analyse and synthesize information in fulfilling their assigned tasks. Work tasks jointly constitute the design of the organization, which contains a variety of artefacts such as pens, calculators and computers but also hierarchical structures, rules and procedures used to facilitate these information processing activities. The choice and configuration of these artefacts are all part of organization design.

The relationship between information processing and organizational design become even more evident when computer and information technologies (CIT) invade all corners of organizational life, facilitating the process of coordination and control, increasing the effectiveness of current or previously abandoned organizational designs and making new organizational designs feasible (Weick, 2001). However, organizational design should not only be seen as a device for the reduction of information costs but also, and perhaps more importantly, as a distinct knowledge-creating input. Grandori and Soda (2006) and Grandori and Furnari (2008) introduce the notion that design has generative and relational capabilities, in the sense that it induces value-creating combinations of resources and activities. In these two articles, a micro-analytic methodology is proposed aimed at understanding and devising the dynamic organizational forms required for matching the fast-changing environments of our days.

The phenomenon of the representational qualities of IT and their impact on the transformation of work has been approached by Shoshana Zuboff (1988) in her classic In the Age of the Smart Machine. In it Zuboff puts forward the notion of “informating work”, explained as follows:

This century’s long trend towards the abstraction of work has been dramatically accelerated since the early 1980s by the adoption of information technologies. These new technologies were widely regarded as simply the next phase of automation. (…) The organization, its internal processes and exchange relationships, becomes visible in a wholly new way, whether that pertains to thousands of newly codified variables in the production process or the global flow of cash tracked on an hourly basis. The word I coined to capture this unique capacity of information technology is informate (Zuboff, 1995:15 bold added).
Throughout the 90s more and more information has been made available to the management of organizations and the resulting effect has been a dramatic increase in a broad organizational capability which we can call “intelligence”. Such capability means that the organization knows more about its production, i.e. how much is produced, how things are produced and so forth. But in spite of all the available information, does the organization know more about itself, in the sense of the thousands of processes that make it what it is? We believe that in most organizations still very little is known about their own execution. All you have to do is to ask a manager if she knows about a key process in her organization and you will be surprised at the response. The reason for this, in our view, is that the organization’s design has not changed and does not allow the masses of existing information to be converted into intelligence.

Contrary to the classical conception, organizational designs are not static. In an insightful similarity, Stanford (2007) compares organizational design to a shoal of fish that continuously change in shape, size and membership, yet it lasts over time. Good designs exhibit dynamic capabilities, allowing the organization to adapt to continuous environmental change while simultaneously keeping the business operations running. The best designs seem to develop an organizational capacity to morph, i.e. to change from one form to another (Stanford, 2007). This thinking dovetails with Weick’s (2001) considerations about organizational improvisation and the need to move from design as a noun to design as a verb (i.e. designing). Some examples of these two opposing conceptions of design are given in Table 1 below (Weick, 2001: 60):

<table>
<thead>
<tr>
<th>Traditional assumptions (design)</th>
<th>Alternative assumptions (designing)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A design is a blueprint</td>
<td>A design is a recipe</td>
</tr>
<tr>
<td>A design is constructed at a single point in time</td>
<td>Designing is continuously reconstructed</td>
</tr>
<tr>
<td>Designs produce order through intention</td>
<td>Designs produce order through attention</td>
</tr>
<tr>
<td>Design creates planned change</td>
<td>Design codifies unplanned change after the fact</td>
</tr>
</tbody>
</table>

From theoretical point of view organization design can be divided up into two approaches: the relational approach and the technical approach (Grandori and Furnari, 2008). The former is about the social processes that shape the design and which are of a behavioural or attitudinal in nature. The latter is about establishing dimensions or attributes which characterize the organization’s internal or external environment and putting forward general principles about how such dimensions or attributes should be combined (Burton et al,
2006). For example, in a highly dynamic competitive environment, the organization’s design should exhibit characteristics of decentralization. Now-a-days it is widely accepted that organizational design is shifting from the traditional stance on command and control to an emphasis on producing results through collaboration, joint initiative and collective learning (see Table 2).

Table 2

![Table 2](image)

3. ENGINEERING

Engineering builds artefacts to be used by people. These artefacts can be machines, material, structures, software programs, models, etc. When designing a new artefact, the engineers identify the set of qualities that the artefact should have in order to be useful to people. The qualities vary with the sort of artefact, for instance, a house should be pleasant to live in, a bridge should support a certain level of traffic, and a software program should be easy to use. In the context of modern organizations, computer-based artefacts, i.e. artefacts that execute or run in a Turing machine, are particularly relevant since today’s organizations manage, store and depend on large amounts of information. Given the encompassing consequences of such artefacts it has become difficult to separate the organization itself from its information.

The dramatic changes which we have witnessed in organizations and organizational design has been due to all the fields of computer science which deal with the engineering of computer-based artefacts. The engineering of computer based artefacts is a formalization process that goes from an informal stage, made up of a set of needs belonging to a group of stakeholders, to a formal stage, which takes the form of a set of executable specifications which (hopefully) contain the answers to such needs. During this formalization process,
several intermediate artefacts are built which reduce the gap between the informal needs and the final executing artefact. These artefacts comprise, among others, business viability plans, requirements documents, architecture blueprints, and test plans.

Several disciplines contribute to the formalization process, such as Information Systems, Artificial Intelligence (McCarthy et al, 1955) and Software Engineering (Naur and Randall, 1969). What is common to all these disciplines, in addition to the fact that their final artefacts must be targeted to execute in the Turing Machine, is that they all use intermediate artefacts. Intermediate artefacts are mostly models that represent intermediate states of the development process. Such models may aim at representing the problem or the solution space of the computer-based artefacts. The emphasis on modelling and their use to study the complexity of computer-based artefacts, by simulation or formal proof, has been growing. In same situations the models have become an end in itself. For example, an ontology is developed to represent the knowledge about a particular organizational domain or a business process is modelled to represent the features of particular aspects of the organization.

The development of computer-based artefacts has become very complex and a variety of disciplines need to be considered, some more technical some less so. Systems Engineering is an interdisciplinary approach that focuses on the engineering of any complex system. It addresses the definition of customer needs and of the required functionality early in the development cycle, documenting requirements, through to design synthesis and system validation, while maintaining a holistic perspective (INCOSE, 2009). In the context of the engineering of computer-based artefacts Software Engineering applies the core knowledge of Systems Engineering while contributing to its regular practices.

The key aim of Software Engineering is the development of software (Naur and Randall, 1969; IEEE, 1993; Pfleeger and Atlee, 2005; Pressman, 2005; Sommerville, 2007). The software is an artefact that together with the supporting hardware, constitutes a computer-based artefact. To understand the impact of computer-based artefacts on organizations it is important to say something about Software Engineering techniques and methods. Software Engineering addresses two key aspects: products and process. The products are the set of artefacts that are built when developing the software system while the process defines the set of activities required to accomplish the final goal of building the software.

Computer-based artefacts can be classified into two groups, depending on what they intend to represent: they can be either problem artefacts or solution artefacts. Problem artefacts describe the problem that needs to be solved (Jackson, 2001) by the artefact while solution artefacts describe how the problem can be solved by a computer-based implementation.
Requirements engineering (Bubenko, 1995) is the part of software engineering that deals with the former while software architecture (Shaw and Garlan, 1996), software design (Gamma et al, 1994) and programming address the later.

The early methodologies for software development tried to emulate industrial processes for building various types of material artefacts proposing sequential development models as, for instance, the waterfall model described by Royce (1970). Such methodologies emphasize integration of the software development activities into a production chain, where each activity receives artefacts from the previous activity in the chain and produces new artefacts for the next activity in the chain. These methodologies were called bureaucratic because they split the development process into clearly separate activities, reducing the communication between the team members and enforcing the writing of documentation describing the intermediate artefacts. In bureaucratic methodologies people communicate through the artefacts they manipulate.

Bureaucratic methodologies presented several problems due to the intangibility of the artefacts. The communication based on the sharing of artefacts is error prone and creates a context where it is difficult to evaluate the current status of the development process: How close it is from delivery of a final artefact that satisfies the needs of the stakeholders? To cope with this problem a new set of methodological approaches were proposed in the end of the 90’s called Agile Methodologies (Agile Manifesto, 2001). These emphasize communication among team members and rapid feedback cycles between artefact construction and the satisfaction of stakeholder needs. The main idea behind these approaches is that stakeholder satisfaction through the use of artefacts is the tangible aspect of the development process.

4. ORGANIZATIONAL DESIGN AND ENGINEERING

So, what new forces might be at work in joining (or re-rejoining) design and engineering in organizations? Organization design, which is part of organization theory, is built on principles which are now quite old. The hierarchical structure and the well known organizational chart representing order and stability are still the way that most people think about organizations. On one hand, we have organizations designed for stability and on the other hand we have organizational environments which is characterised by discontinuities, surprises and permanent change. Without having the tools to do so, organizations are having to adapt to external and internal changes almost on a daily basis. Hence, today’s organizations – profit-making and not-for-profit – have to develop new capabilities for continuous sensing, learning and adapting to ever-changing environments.
Intelligence is the term usually referring to a general mental capability to reason, solve problems, think abstractly, understand new information, learn from past experience and adjust to new situations. When we think of the organizational challenge of the 21st century, this is exactly the type of capabilities which they need to develop. Taking care not to misuse the metaphor or to engage in fanciful thinking, it is legitimate to say that organizations “pursue intelligence” (March, 1999). Building up the organization’s intelligence is obviously an historical process which many authors have likened to a process of evolution. March (1999) suggests that the evolution of organizations can be engineered by means of small, timely interventions in the natural development processes of organizational histories. Such interventions are conceived to be of three different types: (1) altering the possibilities for transmission, retention and retrieval of lessons of history; (2) altering the structure of interactions among the units of evolution; (3) managing the balance between exploration and exploitation in the course of time.

Starting from the last intervention type, what this means is that in any evolutionary process there is a natural tension between exploring new adaptation possibilities and exploiting acquired characteristics which can be managed. Exploration involves experimenting with ideas or technologies, taking risks or relaxing control. Exploitation entails improving existing capabilities, standardizing, routinizing or reducing costs. As regards the second type of intervention, by changing the structure of interactions the engineer might change the advantages that accrue to alternative and competing evolutionary forms. Finally, by altering the possibilities for transmission, retention and retrieval of lessons of history, the engineer can change future events. One simple example of this is what can be achieved in terms of the construction of a database containing historical data about the organization, freely accessible by all organizational members.

4.1 The strategic nature of ODE: the pursuit of organizational knowledge and intelligence

In strategic terms, what does the ODE project try to achieve? The mission of ODE is to help organizations make better use of existing human, information and computer-based resources in order to build up the organization’s knowledge and intelligence in a sustainable fashion. In the early 21st century, we can safely talk of organizational knowledge not only as a competitive market pressure but also as a major cause/consequence of the organizational integration of information technologies (Magalhães, 2004). Furthermore, organizational knowledge is credited now-a-days as the key variable in sustaining competitive growth (Barney, 1991; Collis, 1994; Teece et al, 1997; Helfat and Peteraf, 2003; Teece, 2007). In a simple definition, organizational
knowledge is the ability that any organization has to create and to extend its output actions, i.e. its products or services or the development of new products or services.

Table 3 - Organizational knowledge and organizational design

<table>
<thead>
<tr>
<th>Views on organizational knowledge</th>
<th>Epistemic perspectives</th>
<th>Consequences for organizational design</th>
</tr>
</thead>
<tbody>
<tr>
<td>What is knowledge?</td>
<td>Knowledge is pre-existent</td>
<td>The organization exploits the knowledge of individuals inside and outside its boundaries. Efficient coordination minimizes communication and thus the costs of integration</td>
</tr>
<tr>
<td></td>
<td>Knowledge is created</td>
<td>The organization is designed in a way that leaves room for knowledge creation. For example, communication and cross-skill activities become important</td>
</tr>
<tr>
<td>In what form does knowledge exist?</td>
<td>Knowledge is explicit</td>
<td>Knowledge is exploited through (design) rules that allow control</td>
</tr>
<tr>
<td></td>
<td>Knowledge is tacit</td>
<td>Knowledge is created by designing opportunities for craft apprenticeship</td>
</tr>
<tr>
<td>Where does knowledge exist?</td>
<td>Knowledge is in the individual</td>
<td>The organization is designed in such a way as to integrate the specialist knowledge of individuals in organization-wide capabilities</td>
</tr>
<tr>
<td></td>
<td>Knowledge is in the community</td>
<td>The organization is designed in such a way that most activities are performed in groups, teams or communities</td>
</tr>
<tr>
<td></td>
<td>Knowledge is in the surroundings</td>
<td>The organization is designed in such a way as to exploit external networks and gain access to knowledge</td>
</tr>
<tr>
<td></td>
<td>Knowledge is in the manager</td>
<td>The organization is designed in such a way that management has control of most activities. Management qualifies what knowledge is</td>
</tr>
</tbody>
</table>

Source: Adapted from Christensen (2003)

Organizational knowledge is about “combinative capabilities” (Kogut and Zander, 1992). This means that companies produce new capabilities by combining and recombining existing capabilities and the ability of companies to do this is affected by organizational design principles, including their operating procedures, formal structure and social relations. Thus, organizations have knowledge and learn not because they “think” or “behave” in an independent fashion. Rather, they learn when organizational structures, processes or routines change because of the actions and interactions of the individuals and groups within them. But for such learning to happen at the organizational level, a degree of sustained critical reflection about the established norms, values, attitudes and behaviours is required (Hislop, 2005). From this it becomes clear that there is a large amount of interaction between the notions of organizational knowledge of organizational design. The table above provides an overview.

Lastly, organizational knowledge is also the link between some collective pool of skills, incentives and opportunities that an organization has at its disposal and the direction and
economic effectiveness of the exploration, development and exploitation of such a pool of resources (Dosi et al, 2008). There is a hierarchy of concepts which make up organizational knowledge, starting from organizational capabilities at the top (i.e. the wider scope), organizational routines next and individual skills at the bottom. Routines are the building blocks of capabilities and skills are the building blocks of routines. There are no strict boundaries between capabilities and routines and a large routine might equally be called a routine. Routines can involve not only simple decision rules requiring low levels of information processing, but also complex behaviours that require high levels of repetitive information processing (Dosi et al, 2008). The link between organizational knowledge and organizational design is thus established through the notion of organizational routine.

Becker (2004) has carried out an extensive review of the vast literature on organizational routines and suggests a set of characteristics which help our understanding of this important concept. They are as follows:

- Patterns. Organizational routines exhibit regularities in behaviour that is followed repeatedly. Such patterns can be individual or collective and in the later case, routines are patterns of interaction among actors
- Recurrent. Repetitively and recurrence are key characteristics of organizational routines
- Collective. Organizational routines are collective phenomena which can be distributed across space or across the organization
- Simultaneously automatic and intentional. Organizational routine are characterized by being both mindless and effortful accomplishments
- Processual. A routine is a process, i.e. a somewhat stable sequence of interactions which exhibits a recurrent pattern
- Context-dependent and embedded. Routines are embedded in the organization’s structure, but they are also situated or dependent upon different organizational contexts
- Path-dependent. Organizational routines are shaped by history and can be predicted in part
- Triggers. Organizational routines need to be triggered by internal or external stimuli.

This rather lengthy listing of the characteristics of organizational routines is justified by the importance which we believe this concept will have in ODE research. Routines offer a researchable link between the organization as snap-shot and organizing as a process (Pentland and Rueter, 1994). They are the point where individual and collective skills (i.e. organizational knowledge) are transformed into organizational performance, hence they
provide an observational “window” to the drivers underlying organizational change (Becker, 2004). Such drivers are not independent or even interdependent. Rather, they are the entangled consequence of the interactions of people, decisions and technological artefacts, hence they are ODE drivers.

### 4.2 The epistemological nature of ODE: socio-materiality

ODE takes a realist and emergent stance on organization, reaffirming it as a socio-material phenomenon which self-realizes in the actions and interactions of its component parts. Socio-materiality is an epistemological perspective on technology and work, characterized by Orlikowski and Scott (2008) as the third and latest stream in this tradition of social research (see Table 4). The remaining two streams were characterized by perspectives emphasizing, respectively, a technological imperative and social constructivism. The technological imperative perspective views technology as an exogenous force which determines the behaviour of individuals in organizations and, therefore, as the principal force behind technology-related organizational change. This view is imbued with an optimistic and somewhat naive attitude in relation to effects of automation on organizations and society in general.

<table>
<thead>
<tr>
<th>Research Stream 1</th>
<th>Research Stream 2</th>
<th>Research Stream 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ontological Priority</td>
<td>Individuality</td>
<td>Duality</td>
</tr>
<tr>
<td>Key Mechanisms</td>
<td>Impacts</td>
<td>Interaction</td>
</tr>
<tr>
<td>Main Concepts</td>
<td>Technological Imperative Technology Effects</td>
<td>Structurational Perspectives Social Constructivism</td>
</tr>
<tr>
<td>Logical Structure</td>
<td>Variance</td>
<td>Process</td>
</tr>
<tr>
<td>View of Social and Technical Worlds</td>
<td>Humans/Organizations and Technology are assumed to be discrete, independent entities with inherent characteristics</td>
<td>Humans/Organizations and Technology are assumed to be interdependent systems that shape each other through interaction</td>
</tr>
</tbody>
</table>

Source: Orlikowski and Scott, 2008

The social constructivist perspective on technology and work follows Gidden’s (1984) structuration theory and argues that technology has a dual nature. On one hand, technology has objective reality in the sense that it has embedded in it objective features, such as the design of the hardware or of the software; but on the other hand, technology is also a socially
constructed product in the sense that new structures emerge in human action as people interact with the technology. Orlikowski (1992) has put forward a structurational model of technology, which is intended to throw new light on key aspects of the phenomenon of integration of technology into organizations as well as suggesting typical relationships and interactions. From this perspective, technology is not just equipment, methodologies and policies, but they is also the result of individual sense-making, that is, the perceptions and understanding of the role and value of the data and of the systems themselves (Symons, 1991; Campbell, 1996).

The socio-materiality perspective has its roots in Actor-Network Theory (ANT) which emphasizes the “constitutive intertwining and reciprocal inter-definition of human and material agency” (Pickering, 1995 quoted in Orlikowski and Scott, 2008: 25). ANT uses the metaphor of networks as a tool for explaining and ordering relational interactions:

[the network] has the same relationship with the topic at hand as a perspective grid to a traditional single point perspective painting: drawn first, the lines might allow one to project a three-dimensional object onto a flat piece of linen; but they are not what is to be painted, only what has allowed the painter to give the impression of depth before they are erased Latour (2005: 131)

ANT describes the progressive constitution of a network in which both human and non-human actors assume identities, guided by specific interests and strategies. It also assumes the principle of symmetry for both human and non-human actors in the constitution of the network and implicitly attribute intentionality to technological artefacts. However, such assumptions have been found to be problematic by some critics, on the basis that humans and artefacts “do not constitute each other in the same way” (Suchman, 2007 quoted in Orlikowski and Scott, 2008:24). Hence, rather than treating human and material agents as being symmetrical, the socio-materiality perspective emphasizes the mutual constitution of the social and the material elements and the “performed or enacted nature of the boundaries between them” (Suchman, 2007 quoted in Orlikowski and Scott, 2008:25). Performativity means that the relations between humans and technologies is not pre-given or fixed but only enacted in practice.

The socio-materiality perspective defines organization as “a recurrently enacted and patterned set of relations, reproduced over space and time” (Orlikowski and Scott, 2008:26). If the expression organizational design was to replace the word organization the definition above would not change in the least. Hence, organizational design must be seen as a complex system of elements which may be subsumed under two major categories: the natural and the intentional (McKelvey, 1997). The natural are the human, behavioural, action-oriented and mostly intangible elements not amenable to formal modelling. The
intentional are the man-made, rational, planning-driven and mostly tangible elements which interact with the natural elements to form the socio-material whole. Just like in natural phenomena where it is not possible to understand the parts without simultaneously looking at the whole and where it is not possible to understand the whole without simultaneously looking at the parts (e.g. the phenomenon of the river and the whirlpool), the organization is aptly described as being in a permanent process of flux and transformation (Morgan, 1997) or becoming (Tsoukas and Chia, 2005).

4.3 The hard engineering nature of ODE: the engineering of computer-based artefacts

The entangling of technology and people in the organization, as stated by the social-material perspective on technology and work, has a counterpart in the work by Lehman (1980) on the classification of the models of programs which constitute a computer-based artefact. Lehman considers three different kinds of program models: S, P, and E (see Figure 2). In an S program model it is possible to spell out the problem completely, usually through formal specification, to be solved by a machine-executed program. So, once the problem is specified, it is the programmer’s task to implement a program that satisfies the specification.

![Figure 2](source: Adapted from Pfleeger and Atlee (2005))
Unfortunately, when going from the informal to the formal stage in the development process it is often not possible to specify a problem completely, either because it is a complex problem as, for instance, in a chess game or because the stakeholders cannot spell out their needs thoroughly. In such situations, it is necessary to define an abstraction that describes the parts of the problem considered relevant. In that context, a program is codified to implement what has been spelled out in the abstraction and is executed in order to verify whether it solves the problem. Very often, in these models of program, called a P model, it is necessary to change the abstraction because execution of the code reveals that the initial abstraction does not fit the real problem. As a consequence, it is necessary to have a new development cycle, changing the program to implement the new candidate abstraction.

The need for constant change in the program is even more obvious in implementations that follow the E program model. In this model, the program itself is embedded in the problem space. The program is part of the real world of which it is a part and it existence (i.e. its execution) changes the initial problem that the program was designed to solve. For instance, an expert system to advise on the buying and selling of equity in the stock market is specified by observing the functioning of the stock market in a given moment. After the expert system is in operation, the stock market will not have the same behaviour that it had displayed when observation took place. The expert system changes the stock market and for that reason, the very moment it starts to be used it becomes less suited to its original purpose. So, in the E model programs the problem changes every time a new program starts executing. Another example of a program following the E model is the Google Search Engine, as described by Orlikowsky and Scott (2008).

In fact, most of programs now-a-days follow the E model, and even programs following the S and P models end up by having some impact on the problem space, as stated by the social-material perspective. This view has had a huge impact on the way computer-based artefacts had been developed in the last decade or two. There has been a rethink of software development acknowledging the fact that in constantly changing environments the computer-based artefact is not an extraneous entity that provides a solution to the problem, but it is part of the problem. This is also the vision that Systems Engineering has of a complex system as dynamic systems with continuous feedback loops. The influence of System Engineering on ODE is exerted mainly through systems thinking and systems methodologies, especially those of the “soft” variety (Checkland, 1981; Checkland and Holwell, 1998; Checkland and Poulter, 2006), applied in the context of the development of computer-based artefacts.
Given that the development of computer-based artefacts should evolve in tandem with the evolution of organizations, it is important to revisit the engineering of computer-based artefacts in the light of ODE. Many new questions need to be asked as, for example, what does the phenomenon of organizational becoming (or permanent flux and transformation) mean for the evolution of computer-based artefacts?

### 4.4 The role of computer-based artefacts

Requirements engineering methods, when specifying the problem space, define the set of requirements a computer-based artefact should hold. Usually, these requirements are distinguished as functional and non-functional, where the latter are also referred to as qualities. An example of a quality is availability, measured as the percentage of time a computer-based artefact provides service to its clients. So, in the context of computer-based qualities for organizations, the above question should be restated as: “What are the organizational qualities of computer-based artefacts?”

In our view a fresh look at functional and non-functional requirements is needed in the light of their organizational impact. For a long time, engineers have perceived the impact of their artefacts on organizations, but until now they have not explicitly tried to represent them. Very few methods adopt an organizational perspective (Yu, 1993; Fox et al, 1998) and when addressing it they do not use the accumulated knowledge on organizational design. Hence, ODE research should focus on the organizational qualities of a computer-based artefact according to contemporary research on organizational design. Existing software qualities as, for instance, availability or security, should be rethought in terms of their organizational properties. For example, it would be interesting to investigate the relationship between the access control qualities of a computer-based artefact and the degree of delegation in the organization where it is deployed.

The organizational qualities of a computer-based artefact can be seen as a feature of organizational design encoded in the artefact when it implements an activity or a function of the organization. The features of organizational design contained in computer-based artefacts may not necessarily be the same features of the organization where the artefact is deployed. This can bring us to the conclusion that there is misalignment between the organization and the computer-based artefacts and that it is necessary for them to be aligned (Weston, 1997). We do not share this view and consider that it might even be misleading. Hence, an important research topic in ODE would be the study of the emergence resulting from the co-evolution (Lewin and Volberda, 1999; Lewin et al, 1999) of organizational qualities contained in computer-based artefacts and those contained in the organization’s design.
As an example, we might consider Amazon.com’s emerging organizational design resulting from the qualities held by the computer-based artefact the clients use and the organizational characteristics of a bookstore. It can be observed that the resulting design is different from a traditional bookstore, even though the goals were initially the same, i.e. to sell books. However, organizational qualities of the computer-based artefact, such as 24-hours/day and seven days/week availability, as well as the possibility given to readers to write online reviews of the books, have made a new kind of organization emerge.

4.4.1 The tangible quality of artefacts

The intermediate artefacts built during the requirements elicitation and requirements specification stages have themselves impact on the organization. Intermediate artefacts in requirements engineering include models specifying the functionality of final artefact, for example, use cases models (Jacobson et al, 1992) and problem domain entities models (Arango and Prieto-Diaz, 1991). Current research proposals in requirements engineering stress the need to model the artefact’s environment in order to capture the artefact’s goals and avoid an implementation bias when describing the problem (Zave and Jackson, 1997). So, requirements engineering activities create intermediate artefacts that model the problem in several dimensions, from the artefact’s environment to the artefact’s functionalities and by describing these dimensions there is an added value to the organization’s knowledge.

However, intermediate artefacts do not model the problem “as-is”, given that they are already a construction that represents the problem as something different from what it was before the start of the requirements elicitation stage. So, both, final and intermediate artefacts represent the explicit understanding of what the organization is. This understanding is the result of a knowledge creation process that emerges from the interactions between developers and stakeholders during the development process.

Since intermediate and final artefacts contain knowledge representations of the problem and both are agents of organizational change, during their construction and use, a new question should be raised: In terms of impact on organizational design, is it there any difference between an intermediate and a final artefact? From an ODE perspective, the main difference between an intermediate and a final artefact is that the latter is going to be used by people inside the organization, i.e. the performativity mechanism of socio-materiality. Of course, intermediate artefacts can also be used by people, but final artefacts were developed to support parts of the business and were the chief reason for the investment made. Hence, their use will be enforced. Another distinguishing feature is that the final artefact is more tangible than the intermediate artefacts: People will interact by using the artefact and the interactions can be audited. Auditability is an organizational
quality of a computer-based artefact enabling the use of the lessons of history (March, 1999).

Because of these two qualities of final artefacts, i.e. performativity and auditability, we claim that final artefacts have a larger impact on the organization than intermediate artefacts. Artefacts will be used, thus creating a new organization. On the other hand, the organizational encoding contained in the artefact can be observed, due to its auditability quality, thereby inviting the study of the emergent organization.

The tangibility of final artefacts is essential to provide a point of observation for the study of the organization. Final artefacts are part of the organizational routines that provide an observational “window” to the drivers underlying organizational change (Becker, 2004). A research topic that ODE should address is the study of the correlation between final artefacts characteristics and organizational routines characteristics.

4.4.2 Agile software development

The perspective that the software development process is a knowledge creation process changes our vision of an artefact development process limited in time. The organization continuously change and computer-based artefacts necessarily change continuously as well. This requires continuous software development. The idea of continuous software development is not new and has been proposed in the Agile methodologies literature (Beck and Andres, 2004; Schwaber, 2004). For instance, XP (Beck and Andres, 2004) divides up the development process into small iterations and the set of functionalities to implement in the next interaction can be redefined by the client as a result of what happened in the previous interaction.

Any engineering activity is made up of an observation stage followed by an intervention stage. The engineer observes the problem, analyses it and builds a solution. If the organization is constantly changing, even during the process of observation and analysis, it will not be possible to build effective artefacts, because when the artefact is built, the problem has already changed. This is especially true in times of very rapid environmental change. In order to solve this problem, the software development process should be done in cycles of small duration, such that the time interval between observation and intervention is short enough for that the final artefact to continue addressing the observed problem. Thus, an ODE approach to software development should entail a combination of a short observation stage, where the relevant parts of the organization’s design are analysed (observed), followed by a rapid artefact building and deployment stage, where an
intervention on the organization occurs. This process should go on continuously as a sequence of short observation steps followed by short interventions steps.

Software development is carried out in an organizational context, which includes the stakeholders and their goals. As such, the principles of ODE can be applied to the organizations where software is being developed as well. There is evidence that the developing organization has impact on the technical structure of the developed artefacts. For instance, it has been described how the developing organization influences the artefact’s software architecture and in turn the architecture influences the organization. This is called the architecture-business cycle (Bass et al, 2003). Based on his industry experience, Riehle (1997) describes how the architectural styles used in the artefacts are a consequence of the organizational context. Software development projects follow different organizational designs, as is the case of open-source projects (Raymond, 1999) where the qualities of the produced artefacts as well as the software business model are reflected.

In brief, ODE research should address the organizational design of software development communities. Following the tenets of socio-materiality and given the entangling between people and technology, we claim that from an ODE perspective there is also an entangling between the software development organization and the organization for which the artefacts are being developed.

5. ODE: SOME WORKING PROPOSITIONS

5.1 The normal process of organizational evolution can be designed and engineered especially by intervening at the level of the routines

Organizations evolve naturally but their evolution is marked by human interventions. March (1999) defends this position but goes further by saying that the evolution of organizations can be engineered by means of small, timely interventions in the natural development processes of organizational histories. Organization and management theory and more recently computer science have been greatly responsible for such evolution and such interventions. The more these sciences develop and the more sophisticated the interventions techniques become, the greater the need to ensure that organization/management theory and computer science are developing in the same direction. That, we believe, is the role of ODE.

Changing the possibilities for transmission, retention and retrieval of lessons of history (March, 1999) can be applied to many aspects of organizational design, however from an engineering point of view organizational routines should be the focus of such application.
Routines contain a great deal of the organization’s histories and by intervening on them it is possible to bring about sustainable change. Such change at the routine level would, in turn, contribute to overall changes at the level of organizational design. Routines are also the appropriate locus for engineering interventions due to their semi-stable characteristics. Engineering can only be carried out in aspects of organizations which show relative stability over time. Computer-based artefacts, intermediate and final, contain parts of organizational routines. However the large majority of intermediate artefacts are not actually used and artefacts will only become useful if they are used within the organization. By using computer-based artefacts people are enacting organizational routines and given the characteristics of such artefacts behaviour in the organization acquires a certain level of stability.

5.2 The development of computer-based artefacts is a process of knowledge-creation and knowledge-creation is tightly linked to organization design

Organizational becoming describes a fundamental feature of social organizations, where the organization is understood as a living entity, always changing and where change is the result of a never-ending process of knowledge creation. In the same sense, the development of software can be seen as a knowledge creation process where its final product, the computer-based artefact, is only one of the results of this process. However, so much more occurs! There is learning in the minds of stakeholders, developers, clients and users of the software, and knowledge is represented in the final and intermediate artefacts that were created during the development process. Moreover, due to this knowledge sharing between the developing organization and the target organization, there is an entangling of both organizations which, in turn, creates a need for new business models of software development that emphasize the knowledge creation aspect of software development.

In particular, it is interesting to observe that the specification of the problem is actually, a construction of a problem space that did not exist before. When the different stakeholders interact during the requirements elicitation phase and afterwards during problem specification, they are defining a new problem that did not exist before. So, requirements engineering is about constructing a problem. This process of construction can also be seen as a learning process. Hence, the role of a software engineer is not restricted to the construction of a solution to a problem. She builds the final artefact, but she also participates in the problem construction. She builds intermediated artefacts and defines new domain abstractions together with the other stakeholders (Taylor, 1999). This is why some authors refer to Requirements Engineering as Constructionist Requirements Engineering (Ramos et al, 2005).
5.3 Organizational knowing, designing or engineering need bracketing

The literature has been calling for a more dynamic and process-oriented approach to organization studies. Following the lead set by Weick’s (1979) use of the term “organizing”, Orlikowski (2002) has emphasized “knowing-in-practice” as opposed to organizational knowledge, McGrath (2006) suggests that “structuring” should be preferred to structure when theorizing about contemporary organizational forms and along the same lines, Youngjin et al (2006) claim that a designing is the key activity of organizational design. Although this process-based view is crucial for a proper understanding of organizational phenomena, such fluxes of action must be bracketed for purposes of planning, building or evaluation of computer-based artefacts.

Thus, the organization must be understood as being neither wholly subjective (what it is) nor wholly objective (what it does), but as the result of the dialectical interplay of the two. This view is behind McKelvey’s (1999) proposals for a “model-centered” organization science where research would be bifurcated into two types of activity. On one hand, idealized models of organized or organizing activity should be devised and tested and, on the other hand, descriptive analyses and case studies should be carried out in order to compare “the isomorphism of the model’s idealized processes/structures with that portion of real-world phenomena” (Ibid, p.18). Models do not attempt to explain real-world behaviour; they only attempt to explain “model” behaviour. Hence, in order to make them meaningful and useful to real-world organizations, the idealized models must be checked or validated against real-world phenomena.

In terms of ODE, the suggestion is that organizational development projects (i.e. projects involving design and engineering activities) should be planned and executed through a series of small activities of short duration, such that after each intervention a new observation is carried out to identify how the organization was changed by the last intervention. The organizational routines contained in the computer-based artefacts provides the required stability for observation points to be created. This idea does not follow some of the enterprise architecture practices where intermediate artefacts, usually models representing the organization (“as-is” and “to-be”) are built with the intention of establishing alignment between the business and the information technology applications or infrastructure (Weston, 1997). Such practices force long observation periods with the intervention stage occurring long after the observations has been carried out. In our view this approach is prone to failure given the strong likelihood that the model will no longer match the reality-on-the-ground when the time comes for implementation.
5.4 From organizational alignment to organizational steering

Concurring with Ciborra (2000), our proposals for ODE question the current faith on alignment as one of the key aims of information technology management in organizations. Adding to the dangers of considering that alignment is something that can be achieved through (good) modelling, there is the more profound issue of the nature of alignment. Alignment implies a reference point for something to align by or to align with. Traditionally, the organization’s strategy was considered as a possible reference point, but in an environment where strategy has to change as fast as everything else in the organization, strategy loses its usefulness as an alignment point. Also, if emergence is considered as a true organizational phenomenon (which includes strategy), how does one reconcile alignment with emergence? Alignment, in our view, is only useful if considered as a process of aligning and this cannot be achieved through modelling. It has to be aligning-in-action.

Instead of alignment we propose organizational steering. Steering considers the entangled interplay of people and computer-based artefacts in organizational change. It emphasises continuous analyses, through observation, of the organization’s evolution, making small adjustments between interventions, in moving the organization towards the goals defined by the strategy. The engineering activities should be of short duration followed by organizational design syntheses, where design is not a starting point but an ever changing destination. As the result of the analysis, sometimes a change may be needed to the organization’s strategy. Also, with organizational steering it is not necessary to consider the complete organization as it is often the goal of enterprise architectures, but to identify the key aspects that need to be steered to bring about the desired change.

In order to better explain organizational steering we have borrowed the metaphor of Crossing a River by Searching Useful Stepping Stones on a Sketchy Migration Path:

To cross a river [i.e. to achieve successfully integrated ODE] first the desired position to reach the other side is roughly determined as well as a rough migration path that seems passable. Directed by the rough migration path, stepping stones are searched to come closer towards the desired position on the other side. On each stone, the choice is made about which step should be taken next, based on the possibilities that arise, the rough migration path that has been chosen, and the experience of the preceding steps. Sometimes, it appears that the rough migration path must be adapted to avoid difficulties or to take advantage of an easy to use sequence of stepping stones. It may also appear that another point is more favourable to reach the other side (De Jong, 1994: 150)

To support organizational steering the organizational qualities of computer-based artefacts should be studied in terms of their impact in the organization. The artefact should also be designed with those organizational qualities which are more likely to move the organization in the desired direction. Of course, due to the organizational entangling of people and
technology it is necessary to keep paying attention to the possible emergence resulting from the interplay between the organizational qualities of the artefacts and the current organizational design qualities.

5.5 More organizational self-awareness as one of the outcomes of ODE

By establishing a similarity with architectural design projects, Youngjin et al (2006) suggest that the design of organizations should be the end result of a holistic endeavour involving three factors: (1) a (business/organizational) vision; (2) tight coupling of multiple representational technologies; (3) commitment to a collaborative process of design and construction. We propose that ODE ought to follow the same pattern. First of all, any organizational project should be led by a long-term mission and vision. Secondly, in order for organizational objectives to be turned into implementation plans rigorously, modelling techniques and computer-based artefacts have to be deployed in an integrated fashion with the social structures. Lastly, for an effective execution of the plans, commitment towards true collaboration between the organizational and the engineering sides of the projects is an absolute must.

When applied to organizational settings, the representational technologies that Youngjin et al (2006) talk about are crucial sources of self-awareness. Organizational self-awareness (Tribolet, 2005) is part of the organization’s knowledge creation process. It comprises a number of capabilities that jointly give the organization greater power to know itself. Self-awareness is of course a human capability which can be greatly enhanced or diminished by the existing organizational designs combined with the existing computer-based artefacts. The concept is obviously important because the larger the degree of self-awareness of any organization, the more cohesive it will become and the readier it will be at reacting to environmental change (external or internal). Organizational self-awareness and organizational agility go hand-in-hand.

There are many different types of organizational self-awareness, such as awareness of the strategy, of the identity, of the stakeholders or of the common goals of the organization. Another type of organizational self-awareness where ODE can play a crucial part is the awareness of the routines and the role of the individual in the routines of which she is a part. This is achieved by means of computer-based applications which increasingly store and manipulate information which can potentially be used by the organization for purposes of self-awareness. Although the information processing capabilities exist, most organizations are not doing anything about it because the organization design (i.e. the operating rules or the rules of interaction) often presents an obstacle. The relationship between organization design and organizational self-awareness has already been
approached by Weick (1995; 2001) and Weick et al (2005) through the theory of sensemaking. Sensemaking, which is defined as the structuring of unknown contexts or actions and assigning them with meaning, lies at the foundation of a consciousness or awareness that organizational actors develop of the organization as a whole, starting from their immediate work environment.

ODE research should address, among other possibilities, the study of the organizational designs that facilitate self-awareness most effectively. Moreover, the organizational qualities of computer-based artefacts that enable self-awareness need also to be identified. Besides core business artefacts, other kind of artefacts that enhance communication between the different players of the organization may reveal themselves to be powerful tools for organizational self-awareness. For example, some companies participate in user communities or discussion lists in order to better understand about their own business.

6. CONCLUSION

The following bullet points summarize our proposals.

- Organizations are not about either design or engineering. The "either-or" mindset has been a major obstacle to the development of organizational thinking in the 21st century and this artificial divide must be abolished.
- ODE is about the socio-material entanglement of the “D” and the “E”, meaning that while design defines, recommends or uncovers the interactions between organizational actors (human and non-human), in an intertwined fashion engineering improves, streamlines, monitors or changes the nature of such interactions.
- ODE implementation methodologies must entail short sequential cycles of observation-and-intervention of “E” + “D”
- ODE is about building up the organization’s knowledge and intelligence through an economically sensible use of existing human, information and computer-based resources
- The software development process is a knowledge creation process where the final computer-based artefact is a tangible representation of the organization’s knowledge
- ODE emphasizes the tangibility of computer-based artefacts, meaning their building and use of in the organization or business context.
- Organizational design dimensions or qualities offer a very promising avenue for research and development in ODE, given that they can investigated from both points of view: the social architecture and the technological architecture
• Another recommended locus for research and development in ODE is the organizational routine as the building block of organizational competencies. Computer-based artefacts embody organizational routines and shape the organization’s design through the organizational qualities of artefacts
• Organizational innovation is induced by continuously provoking misalignments between people and technology, harnessing and steering the emerging outcomes.

ACKNOWLEDGMENT

We would like to acknowledge Professor José Tribolet of Instituto Superior Técnico, Technical University of Lisbon, Portugal as an important source of inspiration for this paper.

REFERENCES


