

## Association for Information Systems AIS Electronic Library (AISeL)

ICIS 2010 Proceedings

International Conference on Information Systems  
(ICIS)

2010

# SEQUENCING DESIGN DNA: A SET OF METHODOLOGICAL ARTIFACTS FOR SEQUENCING SOCIO-TECHNICAL DESIGN ROUTINES

James Gaskin

*Case Western Reserve University, james.gaskin@case.edu*

Kalle Lyytinen

*Case Western Reserve University*

Veeresh Thummadi

*Case Western Reserve University*

Douglas Schutz

*Temple University, dschutz@temple.edu*

Youngjin Yoo

*Temple University*

*See next page for additional authors*

Follow this and additional works at: [http://aisel.aisnet.org/icis2010\\_submissions](http://aisel.aisnet.org/icis2010_submissions)

### Recommended Citation

Gaskin, James; Lyytinen, Kalle; Thummadi, Veeresh; Schutz, Douglas; Yoo, Youngjin; Weiss, Aaron; and Berente, Nicholas, "SEQUENCING DESIGN DNA: A SET OF METHODOLOGICAL ARTIFACTS FOR SEQUENCING SOCIO-TECHNICAL DESIGN ROUTINES" (2010). *ICIS 2010 Proceedings*. 202.  
[http://aisel.aisnet.org/icis2010\\_submissions/202](http://aisel.aisnet.org/icis2010_submissions/202)

This material is brought to you by the International Conference on Information Systems (ICIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in ICIS 2010 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact [elibrary@aisnet.org](mailto:elibrary@aisnet.org).

---

**Authors**

James Gaskin, Kalle Lyytinen, Veeresh Thummadi, Douglas Schutz, Youngjin Yoo, Aaron Weiss, and Nicholas Berente

# SEQUENCING DESIGN DNA: A SET OF METHODOLOGICAL ARTIFACTS FOR SEQUENCING SOCIO-TECHNICAL DESIGN ROUTINES<sup>1</sup>

*Completed Research Paper*

**James Gaskin\***

**Kalle Lyytinen**

**Veeresh Thummadi**

Case Western Reserve University, OH

Information Systems Department

\*james.gaskin@case.edu

**Douglas Schutz<sup>+</sup>**

**Youngjin Yoo**

**Aaron Weiss**

Temple University, PA

Information Systems Department

<sup>+</sup>dschutz@temple.edu

**Nicholas Berente**

University of Georgia, GA

Information Systems Department

berente@uga.edu

## **Abstract**

*With the introduction of new digital and physical tools into the workplace, the process of design has dramatically changed over the past few decades. Thus, design processes have evolved into many forms which vary, not only between organizations, but within organizations, and even within teams over time. These myriad “mutations” of the design process call for a new method to identify patterns of design activity and their change in order to deeply understand the design process. In this paper we suggest a new method for identifying patterns of activity in design teams. Such activity involves composites of distributed interactions – both socially and across digital and physical artifacts. We argue that these identifiable patterns comprise the “DNA” of design routines. To capture these patterns, we extend the sequence analysis techniques that are commonly used in genetic research to capture a design team’s interactions with both digital and physical tools over time.*

## **Introduction**

A revolution is underway in social science (Abbott 1995) as social inquiry has moved beyond the identification of unidirectional relationships between generalized, static factors, and is increasingly focusing on contextualized dynamic processes. The resulting process theories enable researchers to explicitly incorporate temporality in their inquiry, and to explain how and why observed output resulted due to patterns associated with specific sequences of activities and events (Van de Ven and Poole 1990). In this regard ample opportunities exist to relate such patterns with either positive or negative outcomes (Abbott 1990). Central to this process-centered view is a notion that infinite varieties of organizational practices can be generated from a finite number of generative elements that make up each activity – much like DNA produces an indefinite number of differences of biological forms (Abbott 1990; Pentland 2003). In recent years, researchers in different fields have devised a variety of methods to analyze sequences of human behaviors (Shoval and Isaacson 2007; Wilson 2001; Wilson 2006). These analyses, however, neither attend to generative and non-linear design tasks, nor do they account for the presence of material artifacts in organizational activity. Yet, such artifacts are inevitably embedded into contemporary work practices and deeply affect them (Leonardi and Barley 2008; Orlikowski and Scott 2008). Therefore, a new method is in order for the study the ‘DNA’ of organizational practices – one that incorporates material artifacts in addition to the human behaviors that underpin seemingly infinite varieties of organizational practices.

---

<sup>1</sup> This research is supported by NSF Grants: VOSS-0943157 and VOSS-0943010.

In this paper we report on our endeavor to craft such a methodological artifact (Hevner et al. 2004). The proposed methodology builds upon and extends event-sequencing techniques for the context of project-based design routines. We seek to capture patterns in the sequences of design tasks that produce varieties (mutate), and what the outcomes are of newly mutated design tasks. Consistent with Cross's (2007) taxonomy of design research, we follow a "design science" approach in order to devise a method for engaging in a "science of design" study. The methodological artifacts that we have developed (design science) are intended to analyze design activity sequences where design activities are presented in terms of actors, physical and digital artifacts and their affordances, spatial distribution, and temporal sequence (science of design).

The remainder of the paper is organized as follows. First, we characterize our domain of interest – design practices – as a context for sequence analysis. Then we develop a formal meta-model crafted specifically for understanding the sequences of design activities. This is followed by an illustration of each element of the meta-model and a justification of its inclusion for sequence analysis. We then review and illustrate the method. We conclude with a discussion of the applications for this notation for analyzing change in design activity, comparing design activities, and for detecting the impact of digitalization on design processes.

## Design Routines as Design DNA

Our interest lies in understanding how design teams transform and improve their work in conjunction with widespread digitalization. By design work, we refer to knowledge-based activities geared toward generating a novel product, material artifact, or service. Design involves a series of translations from ideas to different forms of representations, and eventually to the intended design outcomes. Thus, designers draw upon a variety of physical and digital tools, and each supports some aspect of the design. At each step, different tools support the creation or modification of different representations of design such as drawings, sketches, diagrams, models, requirements and specifications. Design also involves heterogeneous actors, and designers intensively use physical and digital artifacts to communicate with each other, and utilize multiple design representations across these diverse groups (Rosenman and Gero 1996). Representations are highly important to design, and iteration across representations – both individually and socially – forms the fundamental flow of *all* design (Berente & Lytinen 2009). This flow mobilizes heterogeneous bodies of knowledge, conforms to established output and success criteria, and involves significant levels of novelty and complexity. Design activity must therefore traverse syntactic, semantic, and pragmatic boundaries (Carlile 2002). Semantic boundaries that involve local interpretation, tend to be problematic (Boland and Tenkasi 1995; Dougherty 1992) as knowledge and related interests are embedded in the practices of the disparate groups, creating pragmatic boundaries (Carlile 2002) which call for negotiation, dialog, and learning (Bucciarelli 1994). As rationalities of diverse groups vary, design involves also argumentation between different logical modalities (Buchanan 1992). Accordingly, design processes are interactively complex and demand significant task and knowledge-based coordination, creating the potential for significant variations on how these tasks are carried out. Maintaining change in contemporary design work has also become highly critical since new forms of modularity and increased decentralization provide greater freedom for design processes (Yoo et al. 2008). Managers need to gain a better understanding of their design routines to ensure that they fit together with new modular design forms (Baldwin and Clark 1997). In a similar vein, digital tools enable new material forms of design, creating new design patterns (Alexander 1979).

To analyze organizational practices in design work, we approach design practices as repeated enactments of a set of *design routines* that consist of sequences of design tasks, which transform certain inputs to certain outputs. *Design tasks* are performed by specific *actors* who consume and generate design objects (i.e., representations and information) mobilizing different tools (Kock 2008). *Actors* in the design task can be either individuals or groups, collocated or distributed. Different *tools* – both physical and digital – are used to extend their cognition and generate design alternatives (Boland and Tenkasi 1995; Simon 1996). These tools provide certain *affordances* that are enacted to support design tasks. Actors communicate and coordinate these activities using various IT tools (Malone and Crowston 1994), often as boundary objects (Carlile 2002). Further, purposeful generation of design necessarily involves *design objects* that are representations used and produced by design activity.

We select routines consisting of the above elements as our fundamental theoretical construct for two reasons. First, routines have formed a natural unit of analysis within evolutionary accounts of organizational change (Nelson and Winter 1982). Firms grow due to natural selection of routines, while they may decline because they neglect to evolve their routines. Routines mutate, producing variations, through adaptations and searches (Nelson and Winter 1982). Further, variances of routines come from recombining or reconfiguring design tasks that act as slowly

changing building blocks of routines. Design *tasks* act as genes, which in biology are segments of a chromosome, while a design *routine* as a whole (a string of design *tasks* to accomplish a particular design objective) acts as the chromosome. The elements that make up each individual design task can be likened to DNA. The entire catalogue of design routines in an organization can be likened to the genome in organisms.

Taken together, we see three levels of variations. First, variations can come from the changes in the way in which the same design task is carried out due to learning (Type 1). Second, variations can also come from the changes in the way the same set of design tasks are sequenced to form a design routine (Type 2). Finally, variations can come from the changes in the fundamental elements of design tasks (Type 3). For example, an introduction of new digital tools into a design task might cause changes in the arrangement of elements that form the task, causing a mutation in the design task. These three types of changes can be highly interdependent. Our methods can capture the second and third sources of variations in organization routines in design. Next, we look at techniques that can detect such changes in design routines.

### ***Event Sequencing***

Sequences of human activities, such as work processes or buying behaviors, can be analyzed using space and time series (Shoval and Isaacson 2007), where space and time are used as reference points for estimating the sequences of events in organizations. We propose to apply techniques which enable us to detect changes in the order (type 2) and configuration of design tasks (type 3) within design routines interpreted as sequences. Accordingly, sequence analysis can be expected to reveal the range of mutations and the evolution in the “DNA” of design work, thus allowing us to explore what new design tasks are introduced or how they are sequenced differently.

We apply a sequence analysis method to analyze variations and changes in design activity. Genetics researchers seek to analyze the configuration of nature’s elements in the DNA. Similarly, we seek to discover patterns in design practices in project-based organizations and their mutations. Previously, sequence analysis has been used in social studies to analyze variance in spatial or temporal behaviors (Shoval and Isaacson 2007; Wilson 2001; Wilson 2006) and to reveal patterns of social change (Abbott 1990). In the IS field, Pentland (2003) used this technique to measure variations in work processes. Also, Sabherwal and Robey (1993) used this technique to study different IS implementation processes.

None of these analyses, however, have been extended to explicitly analyze different tools and their change. As Arthur (2009) notes, technology evolution, with its combinatorial nature, is similar to that of biological evolution. Therefore, sequence analysis should offer a powerful analytical lens to understand the evolution of technological artifacts in design work and to compare different sequences of design activity involving the entanglement of *different digital artifacts with physical work practices*.

To effectively apply event sequencing, we need to introduce an ‘alphabet’ to characterize elements of each design task (in the same manner as we use four proteins to describe genetic structure), and a way to string members of this alphabet together to build up the “genetic code” of design activity. In order to apply this technique to describe design routines, the basic elements of the design tasks outlined above need to be represented in a form that makes them amenable for sequencing. Simply put, each design task must be represented as a simple ‘string’ by concatenating a set of categorical values, each of which describe a unique element of a design task. Every task representation therefore includes the following five generative elements: (1) a value for a set of *actors* who have specific roles, (2) a value for an *activity* that needs to be carried out for the task, (3) a value for a design *tool* used for the activity, (4) a value for a *design object* used and/or produced by the activity, and (5) a value for an *affordance* enacted for the task.

To represent a unique design task, each category must have a fixed set of values (each part of the coding alphabet) by which they can be represented across all tasks. Each design task will pick up a unique value for each design element. Accordingly, we represent each instance of a design task with a value from each category and then concatenate these values to create one design task representation. When a task involves multiple tools, the use of each tool is represented as separate tasks. These design tasks then form basic building blocks of a design routine. Table 1 shows a basic example of how we might build an alphabet to represent design elements.

Consider the following simple example: The design task involves a design engineer and a modeler who discuss a user interface, using a whiteboard, to reach a consensus. Using the alphabet from Table 1, we could represent this design task as follows: R2R3D1T3A1Z1. If a similar task is done later using email, the design task representation

would look as follows: R2R3D1T2A1Z1. In this example, the underlined identifiers illustrate a ‘mutation’ between these two design tasks.

Once design tasks are sequenced, we map the sequence of the design routine by concatenating strings of design tasks. In the same way a sequence of genes makes up a chromosome in organisms, we sequence design tasks that make up the design routines in organizations.

To facilitate the creation and management of representations of design tasks for sequencing, we need to address three challenges: (1) the need for systematic and consistent representation of each design task and task sequence; (2) visual support for user related validation and creation of design tasks and their sequences; and (3) automatic generation of sequences (as the ones shown above) for sequence analysis. In order to address these challenges, we need to develop a meta-model – a formal grammar – that enables us to specify all legal variations of values for each design task (Type 3 variation) and to present and preserve their order (Type 2 variation). Such a meta-model should offer a high level, formal notation that enables us to encode all possible instances of design tasks and their combinations.

<b>Table 1. Example of an Alphabet for Representing Design Elements</b>		
<b>Category</b>	<b>Instance</b>	<b>Value</b>
Role	Project Manager	R1
	Design Engineer	R2
	Modeler	R3
Design Object	User Interface	D1
	Locking Mechanism	D2
	Product physical form	D3
Tool	CAD	T1
	Email	T2
	Whiteboard	T3
Activity	Discuss user interface	A1
	Design user interface	A2
	Model physical form	A3
Affordance	Consensus	Z1
	Generation	Z2
	Representation	Z3

**Meta-model**

To address these challenges effectively, the formal meta-model must be computer-readable. Therefore, we chose MetaEdit+ (Smolander et al. 1991; Tolvanen and Rossi 2003), a visual metaCASE tool, to build our own visual modeling environment. The meta-model of the design tasks in MetaEdit+ is represented as a set of consistency rules and model relationships captured in a notation called GOPRR (Graph, Object, Property, Role, Relationship model). A key benefit of MetaEdit+ is that once we have created the GOPRR model of the design tasks, the application permits us to create a visual notation to represent design tasks as shown in the Figure 1. MetaEdit+ also permits us to create queries into its repository to generate design task strings and their concatenations as presented. An extensive review of the literature on design work – and related field work done by the authors – indicate that at least five key elements make up design sequences and their semantics for event sequencing; these elements are actor, activity, tool, design object, and affordance. While many other elements may exist for describing design work, we believe these five elements sufficiently describe design work at a level abstract enough to apply to all types of design work, and are thus, more robust than a longer, or lower level list. Lastly, it should be noted that, by using MetaEdit+ we may easily add or remove elements from this list without great effort.

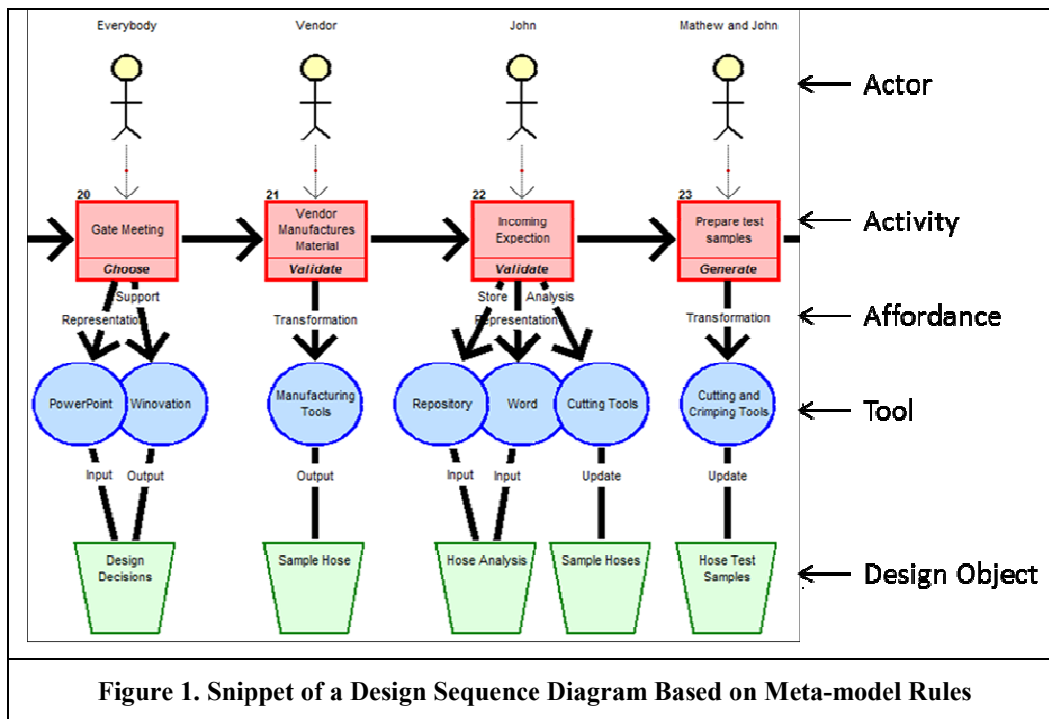


Figure 1. Snippet of a Design Sequence Diagram Based on Meta-model Rules

### Actor

Actors in design tasks can be an individual or a group of individuals who perform the task. Each individual has specific roles.

### Activity

In order to make activity comparable across different project contexts, we adopt a task classification scheme developed by McGrath (1984) that provides four generic types: (1) generate, (2) choose, (3) negotiate, and (4) execute. To the original list, we add *validate* as a fifth category (Bucciarelli 1994; Henderson 1991). We also capture the specific description of the activity for validation.

### Tool

For each tool that is used for a design task, we code whether it is a digital or physical tool. It can also be a design tool that is used to create and modify design representations, or a communication tool. Tools provide specific material features.

### Design Object

As noted throughout the design process, designers work with a variety of physical and digital artifacts, each intended to represent or support some aspect of the design. The design objects can be distinguished by their modalities: sketches, diagrams, requirements, specifications and their nature: digital vs. physical models (Rosenman and Gero 1996). Design objects can be either an input or an output, or they can be updated as part of that activity.

### Affordances

Affordances refer to “the possibilities for goal oriented action afforded by technical objects to a specified user group understood as relations between technical objects and users and understood as potentially necessary (but not necessary and sufficient) conditions for “appropriation moves” (IT uses) and the consequences of IT use” (Markus and Silver 2008). We expand this to cover both physical and digital tools. We do not see affordances as inherent to a tool. Rather, we see them as being enacted for a specific task *in situ*. Following Leonardi and Barley (2008), we

create affordance typologies and then compare them with seven categories developed by Henderson and Cooperider (1994) from 98 CASE tool functionalities. We chose this list as it was derived from analyzing design capabilities and it offers much higher granularity to describe different ways in which design tools are applied by actors. Their affordances list, shown in Table 2, overlaps considerably with Leonardi and Barley’s (2008) typology and covers all their affordances except a “Store” affordance, which we have added. In addition to these five categories of information, we also collect many other aspects of the performative (rather than ostensive) design process in order to create richer descriptions, and to allow for more complex sequences to be analyzed in the future. One specific aspect we collect is whether a task is happening in parallel with another task—Abbott calls these “ties” (Abbott 1990; Abbott 1995). These parallel activities make a unilinear sequence impossible without breaking the exact temporal sequence. However, our use of sequence analysis (described below) is more concerned with the composition of design routines rather than the exact temporal sequence. Thus, whether or not “ties” exist is not important to our analysis of similarities and differences between design routines.

<b>Table 2. List of Affordances</b>	
<b>Affordance</b>	<b>Definition</b>
<b>Representation</b>	Functionality to enable the user to define, describe or change a definition or description of an object, relationship or process
<b>Analysis</b>	Functionality that enables the user to explore, simulate, or evaluate alternate representations or models of objects, relationships or processes
<b>Transformation</b>	Functionality that executes a significant planning or design task, thereby replacing or substituting for a human designer/planner
<b>Control</b>	Functionality that enables the user to plan for and enforce rules, policies or priorities that will govern or restrict the activities of team members during the planning or design process
<b>Cooperative Functionality</b>	Functionality that enables the user to exchange information with another individual(s) for the purpose of influencing (affecting) the concept, process or product of the planning/design team
<b>Support</b>	Functionality and associated policy or procedures that determine the environment in which production and coordination technology will be applied to the planning and design process
<b>Infrastructure</b>	Functionality standards that enable portability of skills, knowledge, procedures, or methods across planning or design processes
<b>Store</b>	Functionality that allows information to be housed within a device.

## Sequence Analysis

Sequences are interpreted as ordered lists (not necessarily temporally ordered lists) of elements (as shown in the previous section). The elements of the sequences are drawn from a set of all possible ‘events’, which is usually referred to as the universe of the events. Several sets of methods have been proposed to analyze such sequences. The first set of methods is a step-by-step approach using time series. It is generally used when the central interest is fairly deep and complex. This set of methods assumes that causal relationships exist in the time series sequence. The second set of methods is a holistic approach, which treats these sequences as full units, rather than individual elements. They are used when the central theme is to find patterns that exist in the sequence (Abbott 1995). The third set of methods uses optimal matching techniques that produce a distance matrix, which can be used for tracing the relations in the sequences. It can also be used to cluster the event data for comparing similarities across different sequences (Abbott 1990). There are other methods like Bayesian techniques, neural networks, process mining, etc. which may be used to analyze the type of data that we have described. But, most of these tools use Markov approaches and algorithms which cannot capture the unique characteristics from larger sequences. The optimal matching technique has an advantage as it can capture the unique characteristic sequence or several characteristic sequences from larger sequences (Abbott and Forrest 1986). Hence we have chosen optimal matching through multiple alignment genetic algorithms available in DNA sequencing software, which are also useful for finding points of departure (mutations) in sequences, generating percent alignment scores, and clustering. “Optimal matching” refers here to the principle where investigators compare entire sequences or subsequences for similarities.



As with other metric-based techniques, these similarities or resemblances are then input for scaling, clustering, and other forms of categorization.

We use ClustalG software tool as the primary tool for sequence analysis (Wilson 2001; Wilson 2006; Wilson et al. 2005). ClustalG is a derivative of ClustalX and ClustalW, both widely used biological sequence analysis tools to detect protein and nucleotide molecules and their structures. ClustalG has been expanded from ClustalX and ClustalW in order to enable the analysis of multiple types of sequences in social sciences. For example, Wilson (2001) and his colleagues have used ClustalG to analyze and cluster behavioral patterns among Canadian women. The reliability of sequence analyses using ClustalG has been demonstrated by Wilson (2006).

In order to create sequences analyzable in ClustalG, we have written multiple scripts in MetaEdit+ query language to scrape relevant data from design task sequence diagrams, including property values of activities, actors, tools, affordances, and design objects associated with each task. The scripts also maintain and estimate the correct sequence for the activities for further sequencing based on order information (which design activity precedes), and end dates (later ending simultaneous activities are sequenced for later time points). This data is then generated as a tab separated file and imported into Excel for restructuring. By applying an “interpretation key” similar to the one shown in Table 1, we can restructure the data set into strings of sequences. These scripts concatenate sequences based on any given criteria. For example, we can create one sequence string for each activity and its related actor, tools, affordances, and design objects. Or, we can create these sequences based on affordances, making eight total sequences from the eight affordances described in Table 2. The same can be done for the five activity types.

These sequences can then be analyzed for descriptive analyses through pivot tables. Descriptive analyses reveal interesting patterns such as the frequency of specific types of affordances for each activity type (Figure 2). In this example we can see that ‘representation’ and ‘transformation’ are the most common affordances. We can also see that transformation most often occurs during ‘generate’ tasks, and representation most often occurs during ‘validate’ tasks.

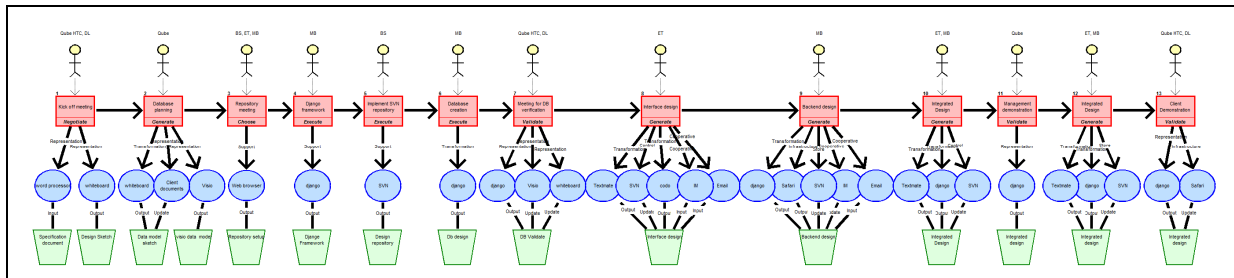
Count of Affordance	Column Labels					
Row Labels	Choose	Execute	Generate	Negotiate	Validate	Grand Total
Control			2			2
Cooperative functionality			4			4
Infrastructure			1		1	2
Representation			2	2	5	9
Store			2			2
Support	1	2				3
Transformation		1	8			9
<b>Grand Total</b>	<b>1</b>	<b>3</b>	<b>19</b>	<b>2</b>	<b>6</b>	<b>31</b>

**Figure 2. Example of Descriptives Using Excel Pivot Tables**

After descriptive analyses, we import the data into ClustalG for further alignment and sequence analysis. ClustalG helps find patterns in sequences and also identifies the sample members for each pattern. To this end, ClustalG performs a pairwise alignment of the sequences in order to construct a similarity matrix that is then converted into distance scores. Next it compiles multiple alignments based on the branching pattern of a tree calculated from pairwise distances and other conventions that affect gap patterns (Wilson 2001). Sequence alignment reduces, discovers, and analyzes patterns in social phenomena by providing a computational method to compare these patterns as sequences of social elements; this is something which would be otherwise highly difficult, especially with overly large, complex, or diverse behavioral pattern sets. This alignment procedure is useful for the study of multiple situations faced in social sciences. For example, it can be used to discover and confirm clusters, sets, and subsets of behavioral patterns in any given context for any unit of analysis – such as travel patterns of tourists based on demographics, or activity patterns of employees and associated performance outcomes (Wilson 2001), or patterns of use with information systems. In our case, the multiple alignments can be used to analyze design routines as a whole and their relationships with design outcomes. Alternatively, we will have the ability to pinpoint differences between patterns, large or small, across different routines or different parts of routines (e.g. early vs. late).

**An Illustrative Example**

To illustrate this novel sequencing method, we present a simple example from a real case study to show how one can carry out a sequence analysis for a design process. The case involves a design routine of a small design firm specialized in web application development. The example below is their design routine for a small web application. The design team consisted of four developers, a project manager, and a customer, and was carried out over a period of four months. The entire routine involved thirteen design tasks. We developed a design sequence diagram based on interviews with members of the software development team. Once we constructed the sequence diagram it was validated with the project manager to ensure the correct sequence of tasks and their elements. The final sequence diagram for the project created with MetaEdit+ is shown in Figure 3.



**Figure 3. A Design Routine Sequence for a Web Application Design**

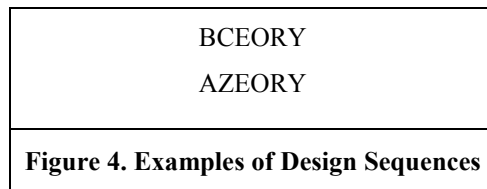
The diagram was next scraped and the data was imported into an Excel file as shown in Table 3a. Using a script, we then turn Table 3a into Table 3b. Figure 4 shows the final sequences generated from the two design tasks shown in Tables 3a and 3b. Each row in these tables represents a design task's composition: tool materiality, design object data flow, affordance type, activity type, and configuration of actors. In Figure 3, the first design task (red box) has two sequences associated with it. The next activity has three, and the next has one, etc.

**Table 3a. Data Before Sequence Representation (partial set)**

Materiality	Data Flow	Affordance	Activity Type	Location	Configuration
Digital	Input	Representation	Negotiate	Collocated	individuals and groups
Physical	Output	Representation	Negotiate	Collocated	individuals and groups

**Table 3b. Data After Sequence Representation (partial set)**

Materiality	DataFlow	Affordance	Activity Type	Location	Configuration
B	C	E	O	R	Y
A	Z	E	O	R	Y



These sequences can then be concatenated using multiple criteria. In Figure 5, we have concatenated sequences associated with each of the 13 design tasks in this project. Thus, each design task (red box) in Figure 3 involves one or more sequences. For example, the first design task sequence is: BCEORYAZEORY. This is the result of

concatenating two smaller sequences BCEORY and AZEORY, each consisting of six basic elements. Other activities result in a longer sequence due to the greater number of affordances involved. For example, the ninth design task, “Back End Design” looks like this: BCIMSUBDIMSUBDLMSUBZGMSUBZKMSU. After we map design tasks, we can concatenate a representation of the entire routine, resulting in an even longer sequence.

Before importing these sequences into ClustalG, we can extract some basic descriptive patterns using Excel’s pivot table feature as shown in Figure 2. Once in ClustalG, multiple sequencing alignments can be performed to produce alignment scores and a percent difference matrix. These outputs describe the similarities and differences between sequences. Sequences are then grouped based on these scores (see Figure 5; for now the colors can be ignored).

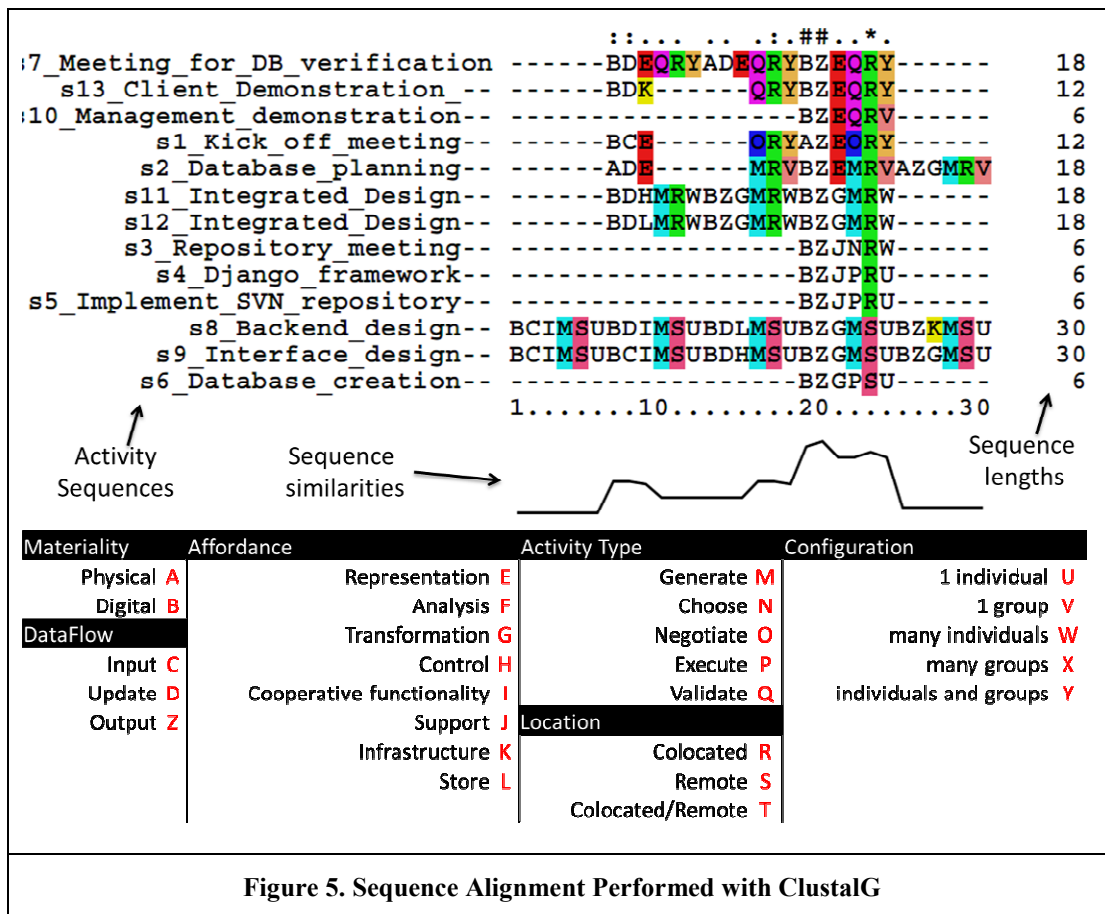
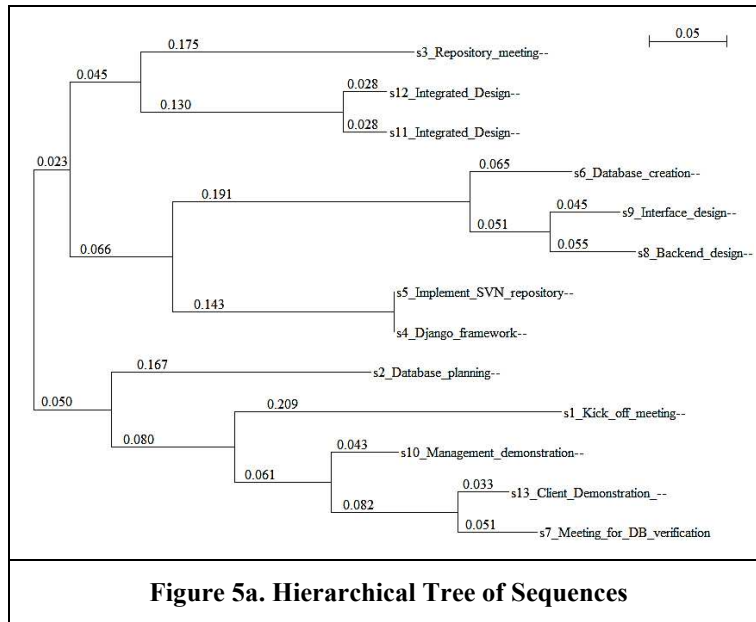
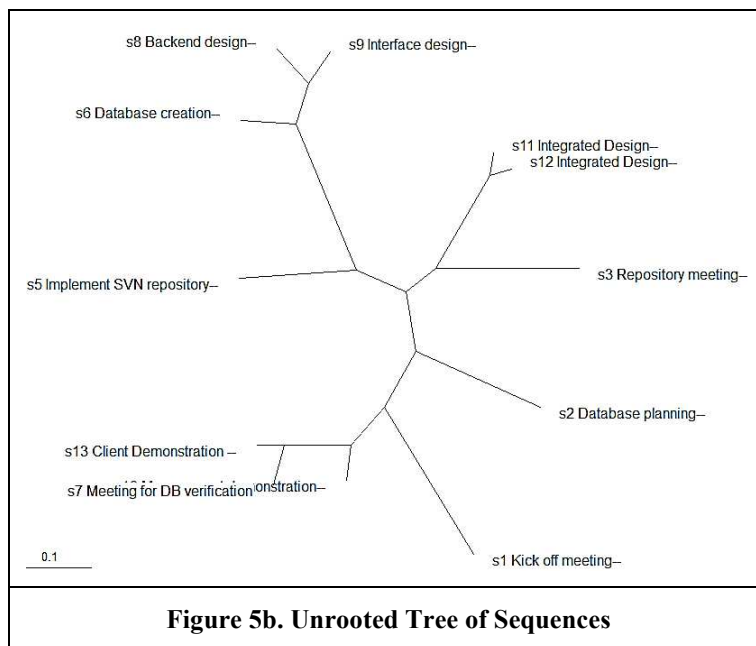


Figure 5. Sequence Alignment Performed with ClustalG

The percent difference matrix can now be used to produce hierarchical trees and unrooted trees as shown in Figures 5a and 5b, using tools like TreeView and NJPlots. These trees offer insights into the clustering structure of sequences. For example, in Figure 5a we can see that sequences 11 and 12 are highly similar in their “DNA”, but they are widely different from sequences 7 and 13. We can also see that sequences 1 and 2 are rather unique and different from the others.



**Figure 5a. Hierarchical Tree of Sequences**



**Figure 5b. Unrooted Tree of Sequences**

## Discussion

A perennial issue in project-based organizations involves managing variation (Cusumano and Nobeoka 1998, Yoo et al. 2006). As each design project represents different design challenges, it is natural and necessary to expect variations in design practices. However, most tools that managers use to depict routines do little to account for this natural variance. Managers and scholars alike need a better way of understanding, measuring, and managing variations. Such a method allows us to explore the causes and consequences of variations, different types of variations and their relationships, and the performance implications of various forms of variations.

In this paper, we propose an empirical method to study variations in design routines in project-based organizations. Drawing on a perspective that views organizational practice in terms of dynamic, evolutionary patterns of activity (Nelson & Winter 1982; Abbott 1995; Pentland 2003), we put forward a view that finite numbers of generative elements that make up design tasks can give birth to a seemingly infinite number of variations in design routines

through recombination of those elements and mutations of the design tasks themselves. Such mutation becomes increasingly important as more firms are introducing powerful digital tools to support design tasks. The method that we introduce in this paper provides a powerful analytic method for carrying out empirical studies to explore these theoretical ideas.

The analytic method we present here does not fall within one of the seven strategies for analyzing process data suggested by Langley (1999). Rather, it seems to be a hybrid of three strategies with some extensions. In some ways, it is like *Visual Mapping* that uses rules and grammars to represent data and relationships through process diagrams. Many of these diagrams can be compared to find patterns and evolutions, as in (Langley and Truax 2007). In other ways our strategy is like *Quantification* that systematically codifies and quantifies qualitative data to enable discrete analyses of interconnected data, as in (Van de Ven and Poole 1990). Our method is also somewhat like the *Synthetic* strategy, which enables the comparison of whole processes for the purpose of identifying regularities across processes, as in (Eisenhardt 1989). However, our strategy extends beyond these three to allow us to analyze multiple units of analysis within a hierarchy, maintain rich details of the relationships in the data, respecify – on the fly – a dynamic grammar used to guide our visual mappings, retain temporal precedence between low level processes, and pinpoint elemental sources of alignment and misalignment.

Our method can provide new insights on variations in design practices at different levels. First, as we demonstrated in our illustrative example, it allows us to explore how different elements – actors, tools, affordances, design objects and activity – make up various design tasks. Even within a single routine, the same task might be enacted in a number of different locations. Also, the same design task is used in a number of different design routines. Since project-based design organizations can change their design methods, tools, and actors with different roles for different projects, the same design tasks might in fact have very different sequences of elements in different contexts. At the same time, we might find that different design tasks might have a high degree of resemblance. Using distance matrices, we can easily assess variations of a single design task within and across design routines.

Sequence analysis also allows us to capture the mutation of design tasks. As organizations introduce new design tools such as 3D CAD tools and 4D simulation systems, the basic nature of certain design tasks might evolve. Our sequence analysis technique enables us to analyze a design team's interactions with both digital and physical tools over time. Earlier case studies on the impact of 3D CAD in architectural design practices show that tools change the nature of specific tasks by providing new affordances and, thus, involving different actors (Boland et al. 2007). Such changes in design task can be represented as mutations using sequence analysis. This allows us to explore evolutions of organizational design routines and practices over time in a concrete way.

Sequence analysis can also be used to analyze entire design routines. Here, we can compare different design routines using distance matrices. Again, since the same design routines are enacted in different projects, responding to different local contexts, we can compare design routines within and across different projects. We can also cluster different types of design routines to understand similarities and dissimilarities among them. If an organization wants to understand how different design tools are being used for different design purposes in different project contexts, such an analysis can provide useful insights. Furthermore, since design practices emerge from the repeated enactment of design routines, one can map out how different design practices emerge over time.

For instance, the example provided in Figure 5 provides the alignment of different sequences in a specific design project. It can be inferred from the picture that some design tasks like Repository Meeting, Django Framework and Implementing SVN Repository exhibit a great similarity. This is referred to as “homology” which means a structural correspondence (Sluys 2009). If we observe these design tasks separately they look very different. But, our sequence analysis has showed us that they are similar. This indicates that their elements like actors, tools, design objects, affordances, and activity types mutate in a similar fashion. In this way sequence analysis can help in planning similar tasks in project-based activities. These, and other potential research questions, are outlined in Table 4.

Recently, scientists have pinpointed what they think is the common causal gene in male pattern baldness (MPB). They were able to do this by comparing sequences of DNA from men with and without MPB. These men were very similar in many other ways, enabling the scientists to pinpoint the likely culprit for the differences between them (i.e., presence or absence of MPB). Likewise, using sequence analysis, we can compare different design organizations in terms of their design DNA. Just as we can identify human genetics using DNA samples, we can compare and contrast different design organizations in terms of their design DNA. When different routines from organizations are fully mapped, we can pinpoint sources of differences between organizations in terms of their design practices at the level of the design task and its constituent elements, just as we can point out differences among people at the gene level.

**Table 4. Example of Research Questions Based on Biology Analogy**

<b>Biological Term</b>	<b>Design Term</b>	<b>Biological Question</b>	<b>Design Question</b>
Gene	Task	What are the outcomes of different configurations of genetic elements, and how do those outcomes change when elements are replaced, removed, or inserted?	What are the outcomes of different configurations of design elements, and how do those outcomes change when elements are replaced, removed, or inserted?
		To what extent are genes within a single chromosome, genome, or organism similar and different, in what ways, and how do they mutate over time?	To what extent are design tasks within a single design routine, project, or organization similar and different, in what ways, and how do they mutate over time?
		How are genes from one organism different from another organism and what are the common threads?	How are design tasks from one organization different from another organization and what are the common threads?
Chromosome	Routine	What are the outcomes (or implications) of different configurations of genes, and how do those configurations change as the organism changes over time?	What are the outcomes (or implications) of different configurations of design tasks, and how do those configurations change as the organization changes over time?
		To what extent are chromosomes within a single genome or organism similar and different, in what ways, and how do they mutate over time?	To what extent are design routines within a single project or organization similar and different, in what ways, and how do they mutate over time?
		How are chromosomes from one organism different from another organism and what are the common threads?	How are design routines from one organization different from another organization and what are the common threads?
Genome	Organization	What are the outcomes (or implications) of different configurations of chromosomes, and how do those configurations change as the organism changes over time?	What are the outcomes (or implications) of different configurations of design routines, and how do those configurations change as the organization changes over time?
		To what extent are genomes within a single organism similar and different, in what ways, and how do they mutate over time?	To what extent are design projects within a single organization similar and different, in what ways, and how do they mutate over time?
		How are genomes from one organism different from another organism and what are the common threads?	How are design projects from one organization different from another organization and what are the common threads?

## Conclusion

In this paper, we introduce a set of methodological artifacts to describe and analyze design routines using sequence-mapping techniques. As a next step, we are currently applying this methodology across multiple design contexts, including software development, mechanical engineering, civil engineering, and microprocessor design. Based on these studies, we will revise the methodological artifacts including the notation, analysis techniques and interpretation (Hevner et al. 2004). Our hope is to gain, for the first time, insights into the “design DNA” of organizations, its variation across multiple design domains, and the impact of digitalization on this activity.

All human actions take place in time. Thus, sequence analysis opens up a powerful avenue to understand the generative grammar that gives birth to variations in many different forms of human actions. We apply this essential theoretical idea based on a structuralist view to understand variations in design routines in organizations.

As digital technologies increasingly mediate human actions, we need to better understand how various forms of digital artifacts are entangled with our social and physical practices. We believe the sequence analysis technique that we introduce in this paper enables us to represent the entanglement of digital tools in design practices in organizations. By explicating elements including materiality of digital tools and representations from them, we believe our method will allow researchers to study the elusive nature of materiality in a more concrete way.

Increasing use of digital technologies in work and life also offers unique opportunities to expand the method we introduce in this study in other domains of human behaviors. Now through the use of digital tools, much of our behaviors leave digital traces behind (Yoo 2010). From the data from electronic patient record systems that show detailed sequences of activities that took place to care for a patient, to mobile phone records that show when and where the caller was for each call, to click stream data captured on a web site, these digitalized sequence data offer unprecedented opportunity to study human behaviors and underlying generative structures. Lazer et al (2009) show how the use of digitalized network data can transform social science. The network-centric view that they presented captures static structure that underpins human behaviors. Our approach provides an alternative route for computational social science. By analyzing digitalized sequence data, we can understand temporal dynamic structures of human behaviors. These two views can complement each other, providing a more comprehensive understanding of human behaviors.

Just as decoding human DNA has brought in an unprecedented level of innovations in medicine, science and engineering, and has ultimately improved the quality of human life, we expect that decoding the DNA of organizational routines will unleash many innovations in organizational science. Our method represents an initial step toward such a goal; a goal, which if realized, promises to have significant implications for managers of design teams (e.g., predicting the impact certain design process compositions/configurations may have on risk, budget, and schedule, and explaining what can be done about poorly configured design processes).

## References

- Abbott, A. "A Primer on Sequence Methods," *Organization Science* (1:4) 1990, pp 375-392.
- Abbott, A. "Sequence Analysis: New Methods for Old Ideas," *Annual Review of Sociology* (21:1) 1995, p 93.
- Abbott, A., and Forrest, J. "Optimal Matching Methods for Historical Sequences," *Journal of Interdisciplinary History* (16:3) 1986, pp 471-494.
- Alexander, C. *The Timeless Way of Building* Oxford University Press, USA, New York, 1979, p. 561.
- Arthur, W.B. *The Nature of Technology: What It Is and How It Evolves* Free Press, 2009, p. 248.
- Baldwin, C.Y., and Clark, K.B. "Managing in the Age of Modularity " *Harvard Business Review* (75:5) 1997, pp 84-93.
- Berente, N. and Lyytinen, K., (2009), "Iteration in Systems Analysis and Design: Cognitive Processes and Representational Artifacts," in Chiang, Siau, & Hardgrave eds, *Information Systems Analysis and Design: Techniques, Methodologies, Approaches, and Architectures*, M.E. Sharpe, Inc. (AMIS Monograph Series), March 2009.

- Boland, R., Lyytinen, K., and Yoo, Y. "Wakes of innovation in project networks: the case of digital 3-D representations in architecture, engineering, and construction," *Organization Science* (18:4) 2007, pp 631-647.
- Boland, R.J., Jr, and Tenkasi, R.V. "Perspective Making and Perspective Taking in Communities of Knowing," *Organization Science* (6:4) 1995, pp 350-372.
- Bucciarelli, L.L. *Designing Engineers* MIT Press, Woburn, MA, 1994, p. 207.
- Buchanan, R. "Wicked Problems in Design Thinking," *Design Issues* (8:2) 1992, pp 5-21.
- Carlile, P.R. "A Pragmatic View of Knowledge and Boundaries: Boundary Objects in New Product Development," *Organization Science* (13:4) 2002, pp 442-455.
- Cross, N. "From a Design Science to a Design Discipline: Understanding Designerly Ways of Knowing and Thinking," *Design Research Now* 2007, pp 41-54.
- Cusumano, M., and Nobeoka, K. *Thinking Beyond Lean: How Multi-project Management is Transforming Product Development at Toyota and Other Companies*. Free Press, NY, 1998.
- Dougherty, D. "A Practice-centered Model of Organizational Renewal Through Product Innovation," *Strategic Management Journal* (13) 1992, pp 77-92.
- Eisenhardt, K. "Making fast strategic decisions in high-velocity environments," *Academy of Management Journal* (32:3) 1989, pp 543-576.
- Henderson, J.C., and Coopriker, J. "Dimensions of IS Planning and Design Aids: a Functional Model of CASE Technology," *Information Technology and the Corporation of the 1990s: Research Studies* 1994, p 221.
- Henderson, K. "Flexible Sketches and Inflexible Data Bases: Visual Communication, Conscriptio Devices, and Boundary Objects in Design Engineering," *Science, Technology & Human Values* (16:4) 1991, p 448.
- Hevner, A.R., March, S.T., Jinsoo, P., and Ram, S. "Design Science in Information Systems Research," *MIS Quarterly* (28:1) 2004, pp 75-105.
- Kock, N. "E-Collaboration and E-Commerce In Virtual Worlds," *International Journal of e-Collaboration* (4:3) 2008, pp 1-13.
- Langley, A. "Strategies for Theorizing from Process Data," *The Academy of Management Review* (24:4) 1999, pp 691-710.
- Langley, A., and Truax, J. "A Process Study of New Technology Adoption in Smaller Manufacturing Firms," *Journal of Management Studies* (31:5) 2007, pp 619-652.
- Lazer, D., Pentl, A., Adamic, L., Aral, S., Barabási, A., Brewer, D., Christakis, N., Contractor, N., Fowler, J., and Gutmann, M. "Perspectives Social Science, Computational Social Science," *Relation* (10:1.119) 2009, p 8099.
- Leonardi, P.M., and Barley, S.R. "Materiality and Change: Challenges to Building Better Theory about Technology and Organizing," *Information and Organization* (18:3) 2008, pp 159-176.
- Malone, T.W., and Crowston, K. "The Interdisciplinary Study of Coordination," *ACM Computing Surveys (CSUR)* (26:1) 1994, p 119.
- Markus, M.L., and Silver, M.S. "A Foundation for the Study of IT Effects: A New Look at DeSanctis and Poole's Concepts of Structural Features and Spirit," *Journal of the Association for Information Systems* (9:10) 2008, p 5.
- McGrath, J.E. *Groups: Interaction and performance* Prentice-Hall Englewood Cliffs, NJ, 1984.
- Nelson, R.R., and Winter, S.G. *An Evolutionary Theory of Economic Change* Belknap Press of Harvard University, Cambridge, 1982, p. 248.
- Orlikowski, W.J., and Scott, S.V. "Chapter 10: Sociomateriality: Challenging the Separation of Technology, Work and Organization," *The Academy of Management Annals* (2) 2008, pp 433 - 474.
- Pentland, B.T. "Sequential Variety in Work Processes," *Organization Science* (14:5) 2003, pp 528-540.
- Rosenman, M.A., and Gero, J.S. "Modelling Multiple Views of Design Objects in a Collaborative Environment," *Computer-Aided Design* (28:3) 1996, pp 193-205.
- Sabherwal, R., and Robey, D. "An Empirical Taxonomy of Implementation Processes Based on Sequences of Events in Information System Development," *Organization Science* (4:4) 1993, pp 548-576.
- Shoval, N., and Isaacson, M. "Sequence Alignment as a Method for Human Activity Analysis in Space and Time," *Annals of the Association of American Geographers* (97:2) 2007, pp 282-297.
- Simon, H.A. *The Sciences of the Artificial* The MIT Press, Cambridge, 1996.
- Sluys, R. "The notion of homology in current comparative biology," *Journal of Zoological Systematics and Evolutionary Research* (34:3) 2009, pp 145-152.
- Smolander, K., Lyytinen, K., Tahvanainen, V.P., and Marttiin, P. "MetaEdit—A Flexible Graphical Environment for Methodology Modelling," Springer, 1991, pp. 168-193.



- Tolvanen, J.P., and Rossi, M. "MetaEdit+: Defining and Using Domain-specific Modeling Languages and Code Generators," ACM, 2003, pp. 92-93.
- Van de Ven, A.H., and Poole, M.S. "Methods for Studying Innovation Development in The Minnesota Innovation Research Program," *Organization Science* (1:3) 1990, pp 313-335.
- Wilson, C. "Activity Patterns of Canadian Women: Application of ClustalG Sequence Alignment Software," *Transportation Research Record: Journal of the Transportation Research Board* (1777:-1) 2001, pp 55-67.
- Wilson, C. "Reliability of sequence-alignment analysis of social processes: Monte Carlo tests of ClustalG software," *Environment and Planning A* (38:1) 2006, p 187.
- Wilson, C., Harvey, A.S., and Thompson, J. "Clustalg: Software for Analysis of Activities and Sequential Events Methods," in: *Workshop on Sequence Alignment*, October, Halifax, 2005.
- Yoo, Y. "Computing in Everyday Life: A Call for Research on Experiential Computing," *MIS Quarterly*, 2010.
- Yoo, Y., Boland, R.J., Jr., and Lyytinen, K. "From Organization Design to Organization Designing," *Organization Science* (17:2), March 1, 2006 2006, pp 215-229.
- Yoo, Y., Lyytinen, K., and Boland, R. "Innovation in the Digital Era: Digitization and Four Classes of Innovation Networks," Working Paper, Temple University.