

Process Complexity: Towards a Theory of Intent-oriented Process Design

Newton Howard*, Colette Rolland**, Ammar Qusaibaty**

* George Washington University
Cyberspace Security and Policy Research Institute (CSPRI)
707 22nd St, Washington, DC 20052
newtonh@gwu.edu

** Université Paris 1 Sorbonne
Centre de Recherche en Informatique (CRI)
17 rue de Tolbiac 75013 Paris, France
rolland@univ-paris1.fr
ammar.qusaibaty@malix.univ-paris1.fr

This work was enabled in part by DOD IASP 24253-1-CCNS20178A grant and by the Center for Advanced Defense Studies (CADS).

Working paper submitted to the International Conference for Information and Systems (INFOS 2004), Egypt, March 2004.

ABSTRACT

This paper attempts to formulate basic building blocks of an intent-oriented process design theory. Motivated by process engineering literature, this work extends the notion of process to an interactive algorithm that executes relational order according to environmental rules and conditions to satisfy an intention. This generalization of process takes into account physical and metaphysical environments. To classify processes beyond the activity, product, decision, and context oriented categories of process models, process complexity is defined by adopting an object-oriented view of situations and actions. Action-types and situation-types are mathematically formulated. This view of situations and actions concludes that process complexity is caused by group representation of objects and group inter-dynamics as formed by individual representation of objects and individual intra-dynamics then vice versa. An extension of intention theory that considers group and individual representations with inter-dynamics and intra-dynamics is concluded. An extended Nash Equilibrium approach is recommended as a future direction to solve group and individual intention dominance. This work recommends an abstract view of process actors as complex systems consisting of stable and modular subsystems. This abstract view includes human agents, information systems, and organizations – groups of human agents and information systems executing collective actions. This work views process actors as C3I3K complex systems (for Command, Control, Communications, Intention, Information, Intelligence, and Knowledge). A basic argument for the evolution of C3I3K complex systems is outlined. Finally, a model of the environment as distributed multilevel infrastructures (DMLI) is recommended to extend the four world model of process engineering.

KEYWORDS

Process Design, Intention Theory, Rational Agents, Heterogeneous Systems.

1. INTRODUCTION

Process engineering is considered key by the software engineering and information systems engineering communities. The recent interest in process engineering is part of the shift from the product view to the process view of systems development [Rolland 98]. Little is known about development process of information systems [Lubars 93] and therefore there is a great need for a conceptual process engineering framework [Dowson 93]. The four worlds framework,

originally proposed in information systems engineering [Jarke 92], requirement engineering [Jarke 93], and method engineering [Rolland 97], can be used to help understand process engineering [Rolland 98]. The four worlds framework, combined with (a) a view of process engineering to each of its worlds and (b) a set of facets to each view, help identify and investigate four major questions of process engineering [Rolland 98]: 1) what are processes, 2) how are they represented, 3) how can their representation

be developed and used, 4) what does process engineering achieve.

The four worlds (see Figure 1 – The four worlds of process engineering [Rolland 98]) of process engineering are:

1. *The subject world:* is the world of processes to represent in information systems. It contains knowledge of the domain about the proposed information system.
2. *The usage world:* includes activities and intentional aspects of agents and stakeholders, and how the system is used to conceive work.
3. *The system world:* contains system specifications representing processes at required levels.
4. *The development world:* focuses on the entities and activities which arise as part of the engineering process itself.

Therefore, the subject world describes processes, the usage world instantiates them among activities and actors. The system world represents these processes in information systems and thus deals with process models. The development world deals with the process of constructing process models and their enactment. This paper focuses on process design within a larger context than the software engineering and information systems domain. The four worlds view is therefore extended to a distributed multi-layer infrastructure (DMLI) view. This view is motivated by actor network theory [Latour 87][Callon 86], distributed network-centric system of systems [Alberts 00], hierarchic systems [Simon 96], and intentional systems [Dennett 87]. This view sets the stage for formulating basic foundations of an intention-oriented process design theory. The first step is to formulate a general definition of process. This formulation is discussed in section 2. Section 2 covers a mathematical classification of action and situation types, process complexity, and C3I3K view of process actors as complex systems. The C3I3K allows a necessary abstraction of interacting complex systems through seven stable and modular subsystems: command, control, communications, intention, information, intelligence, and knowledge. A summary of how these seven subsystems could have evolved is presented in section 2. The inter and intra dynamics of these subsystems and how they become specialized is not discussed in this work. Command and control, as primary

subsystems, are assigned uncertainty measures to set the stage for evaluating the effectiveness of the overall complex system. Section 3 outlines future directions and suggests applying extended Nash equilibrium approaches to study group and individual inter and intra dynamics and object representations. The following statement would be investigated in future research:

In collective action, an agent will execute what is good for itself ontologically and what it perceives as good for the group to satisfy a dominant individual intention and a dominant group intention that minimize individual ontological “uncertainty” and maximize group perceived “utility”.

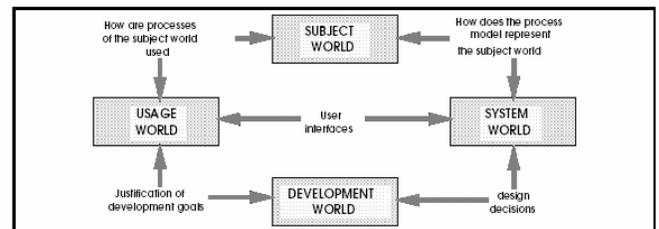


Figure 1 – The four worlds of process engineering [Rolland 98]

2. FRAMEWORK

An immediate extension to process engineering beyond the software engineering and information systems domain, concerns placing people and computers in the same class of systems. The term “symbol systems,” coined by Herbert Simon refers to this specific type. “A symbol system is a machine that, as it moves through time, produces an evolving collection of symbol structures that serve as internal representations of the environments to which the symbol system is seeking to adapt” [Simon 96]. Therefore, a symbol system must have means to: a) collect information from the external environment, b) encode it into internal symbols, c) manipulate internal symbol structures through “reasoning” procedures, and d) produce symbols that initiate action in the environment. The collection, re-organization, production, and the exchange of information enable the basic capabilities of symbol systems .

The correspondence between people and computers predicates itself upon certain distinctions whereby the brain as a physical machine made of flesh and blood corresponding to the computer’s fabricated glass and metal

hardware. In contrast, the human mind designates a virtual machine made of cognitive algorithms that correspond to the computer's software applications. Process engineering essentially concerns itself with virtual machines - minds and software applications - as they command and control physical infrastructures (and often other virtual infrastructures) - brains, machines, systems, and networks. Although the literature makes more use of "information-processing systems" than "symbol systems," both terms will be used as synonyms in this work. Information systems do not limit themselves merely to computer-based or interactive environments of people and computer-based systems such as organizations. Rather, they seek to include all entities that interact due to intentional and spatiotemporal interdependencies. These entities by virtue of their design are "complex systems". Without undertaking a formal definition of "complex systems", they will here be assumed to consist of a large number of many interacting parts. In turn, subsystems also have their own subsystems, and so on. Virtual machines such as the human mind and software applications, and social systems such as organizations will here serve as examples of complex systems.

Generalized Definition of Process

Process engineering literature, and the four worlds view in particular, refer to a definition of process dependent on possible ways to model a process in the system world. Since there are four classes of process models: (a) activity-oriented, (b) product oriented, (c) decision oriented [Dowson 88], and (d) contextual oriented [Bubenko 94] [Rolland 94], there are four adopted definitions of process. These consist of:

Activity-oriented: a process is a set of partially ordered steps intended to reach a goal.

Product-oriented: a process is a series of activities that cause successive product transformations to reach the desired product.

Decision-oriented: a process is a set of related decisions conducted for the specific purpose of product definition.

Context-oriented: a process is a sequence of contexts causing successive product transformations under the influence of a decision taken in a context. [Rolland 98]

All four of these definitions include the notions of "purpose" and spatiotemporal "order" leading towards that "purpose," or, more commonly known as "intention." Spatiotemporal "order"

implies a physical representation in the environment, whereas an environment is not necessarily limited to the spatiotemporal. "Process" refers to a "process model," a representation of the actual event. Similarly, spatiotemporal order designates a "relational" order in a physical model that can be represented in time and space. Differentiating between relational order and spatiotemporal order allows to preserve the initial definition of process regardless of the environment, given that the environment consists of entities that adapt to a given environment. In a physical environment, rules of time and space affect the distribution of entities and their relations. Thus, any relational order primarily designates a spatiotemporal one. In the cognitive or metaphysical environment of a human agent, however, rules of time and space as known in the physical world can be ontological or even nonexistent. Any process then demands the prerequisites of "intention" and relational "order." In the absence of intention, to pursue a specific action through relational "order" does not qualify as a process. Intention with cognitive relational action is one possible form of planning within this framework.

The notion of "order" – inasmuch as it relates to process and depends on the environment – involves actions that couple, reorganize, and redistribute entities. In a physical environment, these entities may include people, systems, objects, organizations, information, etc. Relational order consists of bundles of activity-oriented, product-oriented, decision-oriented, and context-oriented interactions. A process therein qualifies as an "interactive algorithm" relating a multitude of entities according to environmental rules in order to satisfy an intention. Gurevich introduces "interactive algorithm" as an example of algorithms not directly covered by Turing's analysis. This class consists of "algorithms that interact with their environments" and includes "randomized algorithms; you need the environment to provide random bits. It includes asynchronous algorithms; the environment influences action timing. It includes non-deterministic algorithms as well" [Gurevich 03]. In contextual models of process engineering, such as those found in the Nature process theory [Bubenko 94], every process refers to a subjectively perceived situation that itself refers to a specific intention. Spatiotemporal sequential dependencies of situation-decision couplings take place in the process relational order – with a context coupling

various situations and decisions. Process modeling uses contexts as its building blocks in Requirement Engineering (RE) process approach [Rolland 94]. Contexts also instigate use of scenario classes “chunks” as stable and modular substructures for cross-process classes [Ralyte 99].

A process model representation introduced by Rolland for requirement engineering contains a labeled directed graph called “map.” This “map” uses the notions of intention and strategy in which intention captures the notion of actions to be performed while strategy is the manner in which the intention can be achieved [Rolland 00]. Inasmuch, process may be defined as an *interactive algorithm that executes relational order according to environmental rules and conditions to satisfy an intention.*

This generalized definition allows for a theory of process design based on human agents, computer systems to include information systems, and organizations – animate and inanimate actors directly related by collective actions and intentions. Such a theory of process design necessarily requires classification of processes that ignores actors’ individual characteristics. Activity-, product-, decision- and context-oriented classification of processes is insufficient here. As an interactive algorithm executed according to environmental rules and conditions, process complexity – however “difficult”, “time” consuming or “resource” consuming for actors and the environment – would qualify as a way to classify processes.

Process Complexity

The four facets of process models and their respective attributes can be seen as potential “metrics” of process complexity if combined, and include:

- a) a contents facet including coverage and granularity attributes,
- b) an abstraction facet,
- c) a description facet including form and notation attributes, and finally
- d) a modularization facet. [Rolland 98]

The coverage attribute of the process contents facet takes into consideration all different types of activity, product, decision, and context oriented interactions. The granularity attribute considers structural levels of interactivity, especially as they apply to environmental rules and conditions. The abstraction facet measures

the process specificity level. The description facet addresses how to describe and communicate relational order and intention to convey meaning and thus guide process execution. The description facet plays a role similar to that of a plan in situated actions [Suchman 87]. The modularization facet measures the reducibility of a process into subprocesses or recursive processes or semi-recursive processes.

Process complexity informally measures the “difficulty” of describing and executing a process. Unlike Turing-machine based definitions of complexity, which are mainly focused on computable functions or deterministic procedures – e.g. Kolmogorov complexity¹ or Bit complexity² – process complexity has to incorporate a measure of the environment’s complexity. As discussed earlier, the generalized definition of process applies to environments of different structures and rules – physical, metaphysical, etc. As a result, an applicable definition of complexity must take the environment into consideration. Relative process complexity that excludes all measures of environmental complexity may also serve as an alternative.

Generally speaking, “crude complexity is defined as the length of the shortest message that will describe a ‘system,’ at a given level of coarse graining (granularity), to someone at a distance, employing a language, knowledge, and the understanding that both parties share (and know they share) beforehand” [Gell-Mann 94]. A process may refer to a “system” driven by intention and organized by a relational order of entities. Crude complexity addresses the descriptive component of process complexity without addressing execution or environment.

Basic building blocks of Requirement Engineering process as found in project F3 [Rolland 94] are 4-tuple <situation, argument, decision, action>. Arguments and decisions describe types of actions conducted mainly on metaobjects instead of physical ones. To create relational order, the pair <situation, action> can

¹ “The complexity of a pattern parameterized as the shortest algorithm required to reproduce it” [MathWorld].

² “The number of single operations (of addition, subtraction, and multiplication) required to complete an algorithm” [MathWorld].

serve as basic building block instead of the 4 tuple. Using the pair, process defines a sequence of situations and actions leading to an intention. The intention may specify situations to occur and actions to be executed. Any measurement of process complexity therein requires measurement of situation complexity, action complexity, and intention complexity. Assume that “I” is the intention of process “P”. To measure situation complexity relative to “I” requires:

1. The number of actors involved and necessary in the situation “S”. Let N = number of actors involved in situation “S”; actors $a_1, a_2, \dots, a_{N-1}, a_N$ may be animate, inanimate, objects or metaobjects.
2. Actors’ commitment to intention “I”. Let $I_1, I_2, \dots, I_{N-1}, I_N$ be actors’ level of commitment to intention “I”, $I_i \in (0,1]$ with 1 being committed and 0 not committed, $\forall i : I_i \rightarrow 0$ but $I_i \neq 0$.
3. Actors’ order of importance in relation to satisfying the intention “I”. Let $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_{N-1}, \mathbf{a}_N$ be actors’ order of importance in relation to satisfying the intention. $\mathbf{a}_i \in (0,1]$ with 1 being important and 0 not important, $\forall i : \mathbf{a}_i \rightarrow 0$ but $\mathbf{a}_i \neq 0$.
4. Cooperative dynamic relations among all actors. Let $\forall i, j, i \neq j : C(a_i, a_j)$ be cooperation and consistency relations between two actors, $C(a_i, a_j) \in (0,1]$ with 1 being cooperative and consistent while 0 uncooperative or inconsistent, $\forall i, j, i \neq j : C(a_i, a_j) \rightarrow 0$ but $C(a_i, a_j) \neq 0$.
5. Actors’ level of convergence to an ideal actor with skills, experience and knowledge. Let $\forall i : C(a_i, a_i) \in (0,1]$ with 1 being ideal skill, experience and knowledge, and 0 no skill, no experience, and no knowledge, $\forall i : C(a_i, a_i) \rightarrow 0$ but $C(a_i, a_i) \neq 0$.
6. The likelihood of the situation to occur, i.e. the probability that the environment

X is perceived as situation S: $P(X \approx S)$.

7. The importance of the situation to intention satisfaction, $\mathbf{b}(S \rightarrow I) \in (0,1]$; with 0 being an important situation and 1 an insignificant one. $\mathbf{b}(S \mapsto I) \rightarrow 0$ but $\mathbf{b}(S \mapsto I) \neq 0$.

It is then possible to measure situation complexity according to the following equation:

$$c(S, I) = \frac{N}{\mathbf{b}(S \mapsto I) P(X \approx S)} \prod_{i=1}^N \frac{1}{I_i \mathbf{a}_i} \prod_{i,j=1}^{N^2} \frac{1}{C(a_i, a_j)}$$

In a similar way, a measure of action complexity can be defined as:

$$c(A, I) = \frac{M}{\mathbf{b}(A \mapsto I) P(X \approx S | A)} \prod_{i=1}^M \frac{1}{I_i \mathbf{a}_i} \prod_{i,j=1}^{M^2} \frac{1}{B(a_i, a_j)}$$

A is the action, $P(X \approx S | A)$ is the probability that the environment would be in a situation S’ given that action A is executed. $B(a_i, a_j)$ designates relative “cooperation” of a_i with a_j , it also signifies the accuracy of required relational distribution directed from a_i to a_j with 1 being the most accurate and 0 the least accurate. $C(a_i, a_j) \in (0,1]$ $\forall i, j, i \neq j : C(a_i, a_j) \rightarrow 0$ but $C(a_i, a_j) \neq 0$.

In a given process, relational order contains a series of situations and actions redistributing all actors, animate, inanimate, physical, and metaphysical, so as to correspond to the desired intention. In this generalized “object-oriented” view, all “actors” and “objects” form interactive networks, information systems, organizations, physical objects, metaphysical ones (ideas, concepts, themes, relations, etc.). Similar perspectives have been developed by actor network theory [Latour 87][Callon 86] and object-oriented conceptual models to integrate all three perspective of information systems conceptual models: a) process oriented, b) data-oriented, and c) behavioral oriented [Rolland 00].

Every activity consists of unique actions and situations that can never be replicated. Any

action or situation, in all its details, occurs exactly once. The observed variation depends on which granularity level or structural scale, see Figure 2 - Motivated by Galilean space-time.

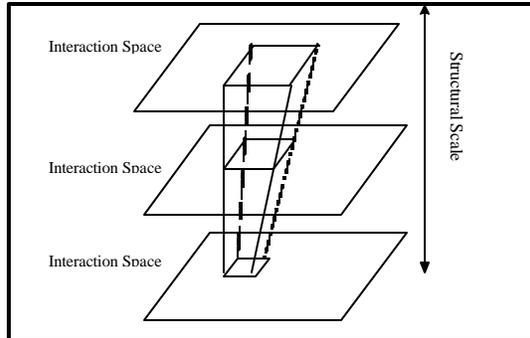


Figure 2 - Motivated by Galilean space-time

Therefore to modularize processes, non-trivial frequencies are to be assigned to actions and situations, i.e. both actions and situations must be regarded as instances or instantiations of more general action-types and situation-types. Many actions and situations can be viewed similarly with respect to a finite set of qualities and a bounded condition ascribed by intention. Consequently, these “conditionally similar” actions and situations may be counted as instances or occurrences of one action-type and situation-type, therefore a notion of frequency is established. The conditional similarity criterion, however, does not eliminate the notorious reference class problem, which is an inherent problem of any notion of frequency. It only shifts the situation, action reference class problem into intention reference class problem.

Thus far, activities are continuous series of actions and situations exercised at multiple structural scales and spaces of interaction (Figure 2 - Motivated by Galilean space-time). Conditionally similar intention-guided activities may then also comprise a notion of frequency. The perceived environment consists of actors that interact according to environmental rules and conditions, both on the physical and metaphysical level. Groups of actors (groups of people and systems such as organizations) form actor-networks.

While Strawson focuses more on physical rather than metaphysical actors, the same notion may be extended to the metaphysical, assuming that

properties of metaphysical spatiotemporal rules differ from those of physical spatiotemporal ones. [Strawson 90] Strawson defines the identity of objects by recourse to spatiotemporal localizations, identifying objects (indirectly) in relation to previously identified objects in their specific location within a comprehensive spatiotemporal frame. Strawson-influenced formal definitions of situations, situation-types, actions and action-types are given below:

Situation: A spatiotemporal distribution of physical-metaphysical actors and actor-systems, in an interaction space of a structural scale.

Situation Equivalency: Two situations, S1, and S2 are said to be equivalent if the following three conditions are met:

1. S1 and S2 belong to an interaction space of a structural scale,
2. S1 and S2 are determined by the same actors and actor-networks in a structural scale,
3. $v(S1,S2) = o(1)$ as time approaches zero, where v is the variation in the distribution of actors and actor-networks, and $o(1)$ is a function that approaches zero as time approaches zero.

Condition 3, means that the variation in the distribution of actors and actor-networks goes to zero as time goes to zero. Conditions 1, 2, and 3 combined describe an indifference criterion of situations called the indifference condition of situations. The indifference condition defines a reflexive, symmetric and transitive binary relation among situations. It is reflexive because every situation is equivalent to itself. It is symmetric by construction and finally, it is transitive because if S1 is equivalent to S2, and S2 is equivalent S3 then S1 is equivalent to S3. In a given set S, a binary relation is an equivalence relation when reflexive, symmetric, and transitive. An equivalence relation on a set S is a partition into disjointed equivalence classes. Each one of these equivalence classes represents a situation-type. Every element of an equivalence class is equivalent to every other element in that equivalence class. Consequently, situation-types are defined with respect to an interaction space of a structural scale. From a set of situations, it is possible to partition a situation-type into disjointed subsets comprised of non-equivalent, separate elements.

Similarly, proposed definitions of action and action equivalence include:

Action: Within an interaction space of a structural scale, an action refers to spatiotemporal redistribution of physical-metaphysical actors and actor-networks to satisfy intent.

Action Equivalence: Two actions A1 and A2 are equivalent if all of the following conditions are met:

1. A1 and A2 meet the same intent
2. A1 and A2 are executed in equivalent situations
3. A1 and A2 have equivalent anticipated situations
4. $w(A1,A2) = o(1)$ as time approaches zero, where w is the variation of distributing actors and actor-networks.

Condition 4 means that the variation goes to zero as time goes to zero. Conditions 1 to 4 define an indifference condition of actions in an interaction space of a structural scale. The indifference condition of actions defines a binary equivalence relation on actions. It therefore partitions actions into disjoint action-classes. Definition of action-types occurs with respect to an interaction space of structural scale.

Intention: An ontological situation convergence point in progress. From an initial condition of the environment, a series of actions and situations lead toward an explicit situation. An explicit specification of a conceptualization [Gruber 93], ontology describes what "exists" in terms of its possible representation(s). When represented in declarative formalism, the knowledge of a domain is called the universe of discourse specified in terms of the set of objects that can be represented.

The defined complexity of situations and actions assumes all variables have the same order, and thus the rate at which they increase or decrease complexity is equal. Increasing the number of actors increases situations and actions complexity at a polynomial rate. But, the rate would increase exponentially if there are contextual (situations and decisions) dependences between situations and actions. If situations create multiple feasible actions that are likely to be executed and if actions to follow create multiple situations that are likely to occur, then a process as a series of situations and actions will grow as a tree. This tree grows exponentially and of order 2^x if there are

x instances of maximum uncertainty – all possible choices are equally likely.

The basic structure of situations and actions and therefore, activities, consist of pairs of two types; the situation-action type and action-situation type. These pairs will be denoted by $S(.) \mapsto A(.)$ and $A(.) \mapsto S(.)$ respectively.

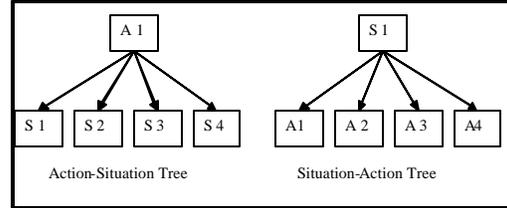


Figure 3 - Action Situation Trees

Assume an action $A(1)$ coupled with situations $S(1)$, $S(2)$, $S(3)$, and $S(4)$, the formed action-situation tree will be denoted by; $A(1) \mapsto S(1,2,3,4)$, or $\perp_j A(1) \mapsto S(j)$

consisting of the pairs; $A(1) \mapsto S(1,)$, $A(1) \mapsto S(1,2,3,4)$, $A(1) \mapsto S(2)$, $A(1) \mapsto S(3)$, and $A(1) \mapsto S(4)$ (see Figure 3 - Action Situation Trees).

The definition of the action-situation tree uncertainty or simply of action uncertainty is given as follows:

Definition: Given a action-situation tree; $\perp_j A(i) \rightarrow S(ij)$, $j = 1,2,3,\dots,m$ with probability distribution of $P(A(i) \rightarrow S(ij) = P_{ij}$ the uncertainty of action is

$$H(\perp_j A(i) \rightarrow S(ij)) = -\sum_{j=1}^m P_{ij} \log P_{ij}$$

This definition of an action uncertainty is highly influenced by Shannon's definition of entropy. The uncertainty of an action-situation tree increases if the number of "branches" increases, i.e. the number of situations coupled with the action increases. Also the uncertainty of an action-situation tree increases if the all situations are equally likely, this means that the probabilities of situations are equal. The uncertainty of action-situation tree is inherently a positive quantity and is zero only if one action has probability 1 (the remaining actions have therefore probability equal to zero). It is likely

that in action-situation trees an outcome situation is not anticipated. However, the definition of the uncertainty of an action-situation tree accommodates this case, as long as the assumed probability distribution takes into account an unknown situation. This assumption is implicitly stated unless in some cases, it is important to isolate the unknown situation in the formula to measure the effect of the unknown on the uncertainty.

In a similar way, the uncertainty of situation-action tree $\perp_j S(i) \rightarrow A(ij)$ can be defined.

Thus, the complexity of a process “P” with respect to intention “I” can be finally defined as:

$$H(P) = \sum_i c(S_i, I) H(\perp_j S(i) \mapsto A(ij)) + c(A_i, I) H(\perp_j A(i) \mapsto S(ij))$$

Intention in this work is viewed as an ontological condition of the environment – an ontological distribution of actors and actor-networks – to be reached by non-unique series of collective actions and situations. However, as concluded by the theory of intention awareness [Howard 02], a collective action among actors, mainly human agents, is a convergence of competing intentions and perceived situations. Whether actors or actor-networks’ intentions are adaptable, or reflexive and deterministic is immaterial. Human agents and organizations are not the only actors and actor-networks that have intentions, computer systems have probably less adaptive and more reflexive and deterministic intentions. Reflexive and deterministic informally speaking implies that for a finite set of inputs (causes, impulses) there are predefined finite set of outputs (effects, responses). The competition of actors’ intentions is manifested by communication, argumentation, negotiation, and persuasion – in short different forms of communication that lead to shared agreement or disagreement.

In conclusion, intention is viewed as a condition of the environment in a form of explicit specification of actors and actor-network distribution. Such condition of the environment occurs as a result of competing intentions applying spatiotemporal physical and metaphysical rules and conditions. This view allows looking at process and consequently process design as collective activity that seeks an intention from different angles:

1. Intention as a condition of the environment expressed as a distribution of actors and actors-networks in the environment.
2. The physical and metaphysical dynamics (rules and conditions) that actors and actors-networks apply while their individual intentions and perceived situations compete.
3. Intention as a condition of the each actor’s physical and metaphysical environment expressed as a distribution of actors and actor-networks include self.
4. The physical and metaphysical dynamics (rules and conditions) of forming, keeping, and evolving intentions of each actor.

These points essentially examine the distribution of actors and actor-systems as perceived by a group of actors dictated by group inter-dynamics formed by individual intra-dynamics as well as distribution of actors and actor-systems. This consists of group representation of entities and group inter-dynamics as formed by individual representation of entities and individual intra-dynamics then vice-versa.

There are three extensions and abstract concepts necessary for intent-oriented design theory development:

1. Abstract and extended view of actors and actor-systems.
2. An extension of a theory of intention that considers individual and group dynamics.
3. Generic view of the environment, actors and actor-networks and their interaction.

I3C3K Complex Systems

A complex system will adapt to its environment and rapidly evolve if its subsystems are stable and modular. “The potential for rapid evolution exists in any complex system that consists of a set of stable subsystems, each operating nearly independently of the details processes going on within the other subsystem, hence influenced mainly by the net inputs and outputs of the other subsystems” [Simon 96]. The efficiency of one subsystem and therefore its contribution to the overall complex system is independent of other subsystem details.

The basic subsystems of complex systems are command and control. A communication subsystem develops if there is a need for collective action due to bounded rationality or scarcity of resources [Arrow 74], informally speaking, if a system has more needs than it can achieve by its resources – system design limitation - cannot continue to exist without interacting and exchanging with other systems. The first law of thermodynamics dictates communications. A communication subsystem therefore necessitates the notion of boundary, internal versus external, self versus environment. The notion of boundaries “establishes” (is highly connected to) the notion of convergence and differentiation, similarity and dissimilarity. If the environment consists of at least two separable entities that are equally likely to interact - maximum uncertainty - with the complex system then the problem of choice emerges. To select among possible interactions (communications) with other entities in the environment impose the need for a subsystem of intention. Thus a complex system has two basic subsystems, being command and control and two emergent subsystems being communication and intention highly influenced by the notion of an environment. Sequences of interaction among entities in the environment create patterns, and hence create relations. As soon as a complex system creates a relation between two entities it has created a virtual entity – a metaobject. This means that coupling, creating a relation of convergence or differentiation, is the prime element of an “idea” that doesn’t have to have a physical representation of existence to exist and be listed. The first ontologies are spaces of ideas. The first order of cognition is in essence based on having a notion of environment, sameness and difference, thus creating primarily relations. This creates an infrastructure of interaction and relation for complex systems. Collective and relative first order cognition and second order cognition create an infrastructure of information from the infrastructures of relations and interaction at lower levels. The Shannon-Weaver Model of communications [Shannon 63] concludes that there are three levels of communications or what is known as three design problems when it comes to interaction: the technical problem, the semantic problem, and the effectiveness problem, See Figure 4 - Communication levels - motivated by Warren Weaver [Shannon 63].

1. Level One: (Technical Problem) How accurately can the symbols of communication be transmitted?
2. Level Two: (Semantic Problem) How precisely do the transmitted symbols convey the desired meaning?
3. Level Three: (Effectiveness Problem) How effectively does the received meaning affect conduct in the desired way?

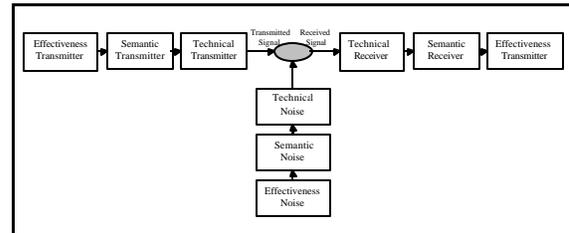


Figure 4 - Communication levels

Level two of the Shannon-Weaver model deals with information – assigning meaning to data communicated. Level three of the Shannon-Weaver model deals with intelligence. Information is an abstract concept. As a function it is coupling two entities and as a subsystem it is a collection of metaobjects. Information theorists define information – for human agents - as news that changes belief and desire. Economists define it as observations that change individual probability distribution [Arrow 74]. Information is individual temporal couplings of objects, metaobjects that cause order redistribution. A complex system will need two additional subsystems for information, one for intelligence and the other one is for knowledge to deal with representation and organization of information efficiency see Figure 5 - C3I3K Evolution and Figure 6 - C3I3K Complex Systems.

This summary describes how a complex system would evolve responding to individual design limitations and environmental constraints (rules and conditions). It highlights necessary set of subsystems generic enough to formulate intent-oriented process design theory that applies on human agents, computer systems and organizations.

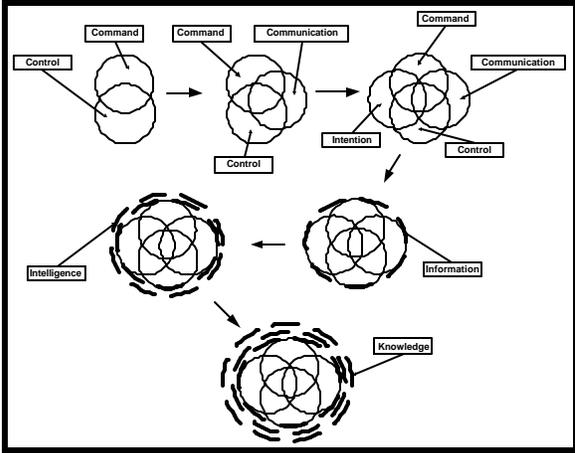


Figure 5 - C3I3K Evolution

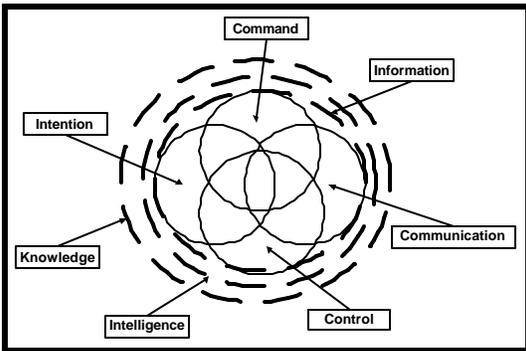


Figure 6 - C3I3K Complex Systems

Command and Control Uncertainty

In what follows the uncertainty of command and control – as primary subsystems in complex systems - is defined. The uncertainty of command is defined by realizing that during a period of time, starting t_{in} and ending t_f , multiple action-situation trees, referenced by time t , see Figure 7 - Command Control Sequence. The following defines a positive quantity, called the uncertainty of command, calculated as:

$$H(com) = \frac{1}{t_f - t_{in}} \int_{t_{in}}^{t_f} H(\perp A(t) \rightarrow S(t_j)) dt$$

In a similar fashion, the uncertainty of situation-action tree, $\perp S(i) \rightarrow A(ij)$,

$j = 1, 2, 3, \dots, n$ with probability distribution of $P(S(i) \rightarrow A(ij)) = q_{ij}$ is :

$$H(\perp S(i) \rightarrow A(ij)) = - \sum_{j=1}^n q_{ij} \log q_{ij}$$

The uncertainty of control over a period of time, starting t_{in} and ending t_f , is defined as:

$$H(con) = \frac{1}{t_f - t_{in}} \int_{t_{in}}^{t_f} H(\perp S(t) \rightarrow A(t_j)) dt$$

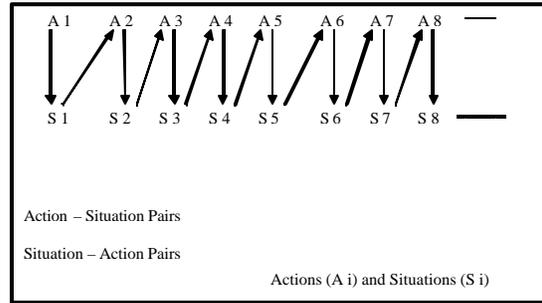


Figure 7 - Command Control Sequence

DMLI View

In this work there are three dimensions to consider in studying human agents, computer systems, and organizations. These dimensions are spatial levels, temporal levels, and representational levels. A view of the overall environment as distributed multilevel infrastructures at each dimension of granularity, temporality, and abstraction, is considered. This view is referred to as DMLI view (Figure 8 - DMLI Model) and is motivated by actor network theory [Latour 87][Callon 86], distributed network-centric system of systems [Alberts 00], hierarchic systems [Simon 96], and intentional systems [Dennett 87]. This approach looks at a network of actors connected by links, relations, rather than autonomous agents. Actors include human and non-human actors such as artifacts. The normally used term is "actant". Translations along links among actants occur to hold the network together. There are layers of networks expressing order relations and power dependencies. A network recruit actants to translate their language and values (ontology) in

a socio-technical mechanism where compromises are made and consequently projects are coordinated and systems are created.

In the DMLI view, the world is considered as multilevel infrastructures at each scale with interaction at the primary layer consisting of actors, processes, and systems. Collective and relative first order cognition “creates” an infrastructure of relations from the infrastructure of interaction by associating, grouping, and other basic cognitive processes. Collective and relative first order cognition and second order cognition create an infrastructure of information from the infrastructures of relations and interaction at lower levels.

The DMLI (Figure 8 - DMLI Model) view proposes a shift from considering the interactive environment as states in cellular automata into relative hierarchical layers with horizontal and vertical processes communicated by agents and systems. A process in this work is viewed as an interactive algorithm [Gurevich 03] in a distributed heterogeneous multi-agent multi-infrastructure operating environment anchored on relative structures of intentions and actions being initiated from command control, and communication subsystems.

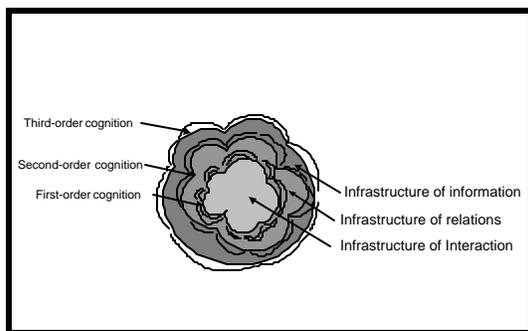


Figure 8 - DMLI Model

3. FUTURE DIRECTIONS

As indicated in Process Complexity section an extension of a theory of intention that considers individual and group dynamics is needed. Bratman [Bratman 87] [Bratman 88], Cohen and Levesque [Cohen 90a] identified seven specific properties that must be satisfied by a reasonable theory of intention:

1. Intentions pose problems for agents, who need to determine ways of achieving them.
2. Intentions provide a “filter” for adopting other intentions, which must not conflict.
3. Agents trace the success of their intentions, and are inclined to try again if their attempts fail.
4. Agents believe their intentions are possible.
5. Agents do not believe they will not bring about their intentions.
6. Under certain circumstances, agents believe they will bring about their intentions.
7. Agents need not intend all the expected side effects of their intentions.

As a future direction, the above mentioned requirements will be generalized to apply to complex systems considered in this work. Furthermore, the notion of intention dominance, the center of intention gravity in which intentions form a modular lattice with contextual order – motivated by Wegner [Wegner 02] and Dennett [Dennett 03], and Wille [Wille 92] – will be analyzed. The analysis would include individual and group perception, action, and dynamics. Nash equilibrium approach or one of its extensions [Nash 50] will be applied on intentions to evaluate accuracy such an approach. The following informal claim – proposed by the author - will also be investigated within the framework of Nash equilibrium:

In collective action, an agent will execute what is good for itself ontologically and what it perceives as good for the group to satisfy a dominant individual intention and a dominant group intention that minimize individual ontological “uncertainty” and maximize group perceived “utility”.

Furthermore, the I3C3K framework will be extensively formulated by adopting a Triality Model of cognition for complex systems built on a model of the human mind that considers three “realities” and multiple selves: ontological, perceived and operating. The Triality Model is based on the following themes that will be formally argued:

1. The human mind has three areas of cognition; ontological reality, perceived reality, and operational reality.
2. The mind’s cognitive processes framework is based on a specialized heterogeneous multi-agent architecture,

- in which agents in each of the three realities - ontological, perceived and operational – interface, communicate, negotiate, cooperate, dominate, and dispute.
3. Each reality is a multi-layered domain, consisting of three or more layers – proactive, active, and reactive.
 4. Representations in all three realities of the mind vary and their distribution differs among human minds. Rule-based, case-based, and graph-based representations are three general classes to be discussed.

The Triality Model will be formulated within the multilayer perceptrons (MLP) framework [Reed 99] in artificial neural networks and the Real-Time Control System (RCS) reference architecture [Albus 01].

Finally, several case studies will be considered and analyzed utilizing the overall proposed process design theory. These case studies include but not limited to the following areas: grid computing [Blythe 03], automatic software generation, self-organizing software environment [Negoro 02], context-oriented requirement engineering [Rolland 00], and US Military operations [Howard 02].

4. CONCLUSIONS

A process is an interactive algorithm that executes relational order according to environmental rules and conditions to satisfy an intention. Process complexity, as an alternative to classify process models, can be defined by object-oriented abstraction of situations and actions. Process actors, including human agents, computer systems and organizations, can be viewed as C3I3K complex systems consisting of seven layers of subsystems: command, control, communications, intention, information intelligence, and knowledge.

REFERENCES

- [Alberts 00] D.S. Alberts, J.J. Gartska, and F. P. Stein, *Network-centric warfare: developing and leveraging information superiority*, 2nd Edition (Revised), DoD C4ISR Cooperative research program, 2000.
- [Albus 01] James S. Albus, Alexander M. Meystel, *Engineering of Mind: an introduction to the science of intelligent systems*, John Wiley & Sons: Wiley-Interscience book, 2001.
- [Arrow 74] K. Arrow, *The limits of organization*, W. W. Norton Company INC., 1974.
- [Blythe 03] Jim Blythe, Ewa Deelman, Yolanda Gil, Carl Kesselman, *Transparent Grid Computing: a Knowledge-Based Approach*, Fifteenth Innovative Applications of Artificial Intelligence Conference (IAAI-03), 2003.
- [Bratman 87] M. E. Bratman, *Intention, Plans, and Practical Reason*. Harvard University Press: Cambridge, MA, 1987.
- [Bratman 88] M. E. Bratman, D. J. Israel, and M. E. Pollack, *Plans and resource-bounded practical reasoning*. Computational Intelligence, 4:349-355, 1988. [BDI].
- [Bratman 90] M. E. Bratman, *What is intention?* In P. R. Cohen, J. L. Morgan, and M. E. Pollack, editors, *Intentions in Communication*, pages 15-32. The MIT Press: Cambridge, MA, 1990.
- [Bubenko 94] J. Bubenko, C. Roland, P. Loucopoulos, V. De Antonellis, *Facilitating fuzzy to formal requirements modeling*, In Proc. of 1st ICRE Conf., Colorado Springs, USA, 1994.
- [Callon 86] M. Callon, *Elements of sociology of translation*, In John Law, editor, *Power, action and belief: A new sociology of knowledge?* Routledge and Kegan Paul, 1986.
- [Cohen 90a] P. R. Cohen and H. J. Levesque, *Intention is choice with commitment*. Artificial Intelligence, 42:213-261, 1990.
- [Cohen 90b] P. R. Cohen, H. J. Levesque, *Rational interaction as the basis for communication*. In P. R. Cohen, J. Morgan, and M. E. Pollack, editors, *Intentions in Communication*, pages 221-256. The MIT Press: Cambridge, MA, 1990.
- [Collins 99] Harry Collins, Martin Kusch, *The shape of action*, The MIT Press, 1999.
- [Dennett 87] Daniel C. Dennett, *The intentional stance*, MIT Press: A Bradford Book, 1987.
- [Dennett 03] D. C. Dennett, *Freedom Evolves*, New York: Viking Penguin, 2003.
- [Dowson 93] M. Dowson, *Software process themes and issues*, IEEE 2nd Int. Conf. on the software process, pp 28-40, 1993.
- [Edward 98] Paul N. Edwards, *Virtual Machines, Virtual Infrastructures: The New Historiography of Information Technology*, Isis: Essay review, University of Chicago Press, 1998.
- [Feiler 93] P. H. Feiler, W. S. Humphrey, *Software Process Development and Enactment: Concepts and Definitions*, Proc. 2nd Int. Conf. on "Software Process", 1993.
- [Fodor 83] J. A. Fodor, *The modularity of mind*, Cambridge, MA: MIT Press, 1983.
- [Gäbler 03] Michael Gäbler, Moritz Stefaner, and Martin Schmidt, *Presentation for the*

- Philosophy of Mind practice*, University of Osnabrück, 2003.
- [Gell-Mann 94] M. Gell-Mann, *The Quark and the Jaguar: Adventures in the Simple and the Complex*, New York: W.H. Freeman and Co, 1994.
- [Georgeff 99] Michael Georgeff, Barney Pell, Martha Pollack, Milind Tambe, and Michael Wooldridge, *The Belief-Desire-Intention Model of Agency*, 1999.
- [Gruber 93] T. R. Gruber, *A translation approach to portable ontologies*, Knowledge Acquisition, 5(2):199-220, 1993.
- [Gurevich 03] Yuri Gurevich and Andreas Blass, *Algorithms: A Quest for Absolute Definitions*, Bulletin of European Association for Theoretical Computer Science 81, 2003.
- [Howard 02] Newton Howard, *Theory of Intention Awareness in Tactical Military Intelligence: Reducing Uncertainty by Understanding the Cognitive Architecture of Intentions*, 1stBooks Library, 2002.
- [Latour 87] B. Latour, *Science in Action*, Harvard University Press, 1987.
- [Lubars 93] M. Lubars, C. Potts, C. Richter, *A review of the state of the practice in requirement modeling*, Proc. Int. Symposium of Requirement Engineering, 1993.
- [Marr 82] D. Marr, *Vision*, San Francisco: W. H. Freeman, 1982.
- [MathWorld] <http://mathworld.wolfram.com>
- [Moore 77] R. C. Moore, *Reasoning about knowledge and action*, In Proceedings of the Fifth International Joint Conference on Artificial Intelligence (IJCAI-77), Cambridge, MA, 1977.
- [Nash 50] J. F. Nash, *Equilibrium points in n-person games*, Proceedings of the National Academy of Sciences 36: 48-49, 1950.
- [Negoro 00] F. Negoro, Principle of Lye Software, Proceedings of 2000 International Conference on Information Society in the 21st Century (IS2000), pp. 441-446, 2000.
- [Negoro 02] Fumio Negoro, *Lye's Hypothetical World*, The proceedings of the workshop of Lye International Joint Scientific Research Project, Paris, France, 2002.
- [Reed 99] Russell D. Reed and Robert J. Marks II, *Neural Smithing: supervised Learning in Feedforward Artificial Neural Networks*, MIT Press: Bradford Books, 1999.
- [Ralyte 99] J. Ralyte, R. Collect, V. Plihon, *Method enhancement with scenario based techniques*, Proc. CAISE 99, 11 Conf. Advanced information systems engineering, Germany, 1999.
- [Rolland 00] C. Rolland, N. Prakash, *From conceptual modeling to requirements engineering*, Annals of Software Engineering 10, 151-176, 2000.
- [Rolland 98] C. Roland, *A comprehensive view of process engineering*, Proceedings of the 10th international conference CAiSE98, B. Lecture notes in computer science 1413, Pernici, C. Thanos (Eds), Springer. Pisa, Italy, 1998.
- [Rolland 94] C. Rolland, *A contextual approach for the requirements engineering process*, 1994.
- [Schroeder 03] Michael Schroeder and Ralf Schweimeier, *Arguments and Misunderstandings: Fuzzy Unification for Negotiating Agents*, 2003.
- [Shannon 63] C. Shannon, W. Weaver, *The mathematical theory of communication*, University of Illinois Press, 1963.
- [Simon 96] Herbert A. Simon, *The sciences of the artificial*,

- Cambridge, MA: MIT Press, 1996.
- [Suchman 87] L. Suchman, *Plans and situated actions: the problem of human-machine communication*, Cambridge, 1987.
- [Strawson 90] P. Strawson, *Individuals, an essay on descriptive metaphysics*, London Routledge, 1990.
- [Wegner 02] Daniel Wegner, *The Illusion of Conscious Will*, Cambridge, MA: MIT Press, 2002.
- [Wilenski 83] Wilenski, *Planning and understanding*, Addison Wesley, 1983.
- [Wille 92] Rudolf Wille, *Concept lattices and conceptual knowledge systems*, Computers & Mathematics with Applications, Volume 23, Issues 6-9, March-May, 1992.