ACADA, An Anticipatory Design Approach

Stig C Holmberg
Mid Sweden University, 83125 ÖSTERSUND, Sweden
Fax : +4663 165700, E-mail: shbg@ieee.org, http://www.itm.mh.se/~stihol

Abstract
System design is still challenging human mental and intellectual capacity. Consequently system design projects often fail. At the same time, anticipation and the anticipatory paradigm are not used in the multiplicity of design methods at hand. As a remedy, the ACADA (Anticipatory Computer Aided Design Approach) is proposed to be added to the methods repertoire. With ACADA it will be possible to test and verify the design before its implementation. ACADA builds on anticipatory modelling and computing in a formal (algorithmic) computer aided system followed by an association from the formal system and its parameters to the concrete living system, i.e. the Human Activity System (HAS) in focus. This way of intervening into living systems, or designing HAS, represents a new application of anticipatory theory, method, and technique.

Keywords: Anticipation, Systems Design, Modelling, Computer Aided Simulation, Association.

1 Introduction

System design occupies itself with the conceptual definition of all types of systems. It is experienced as an extremely complex task which challenges human capacity to its uttermost limits. Hence, in order to support humans in the difficult design task a multiplicity of design methods have been proposed (Collen and Gasparski, 1995; Jackson, 2000). Warfield (1990) has even made ambitious and skilful contributions for establishing design as a separate scientific discipline, design science. Two observations, however, are crucial in this context. First, even with the many design methods at hand design remains a tricky endeavour and failures in design projects are frequent. Second, anticipation and the anticipatory paradigm, in the meaning of Rosen (1979; 1985), has not yet been applied for supporting system design activities.

Hence, the purpose with this paper is to demonstrate the strength and usefulness of anticipatory techniques in support of design activities. This will be attained by identifying the common systemic framework of design and anticipation and by adapting findings from systemic design to anticipatory modelling and computing. A further goal is to increase the potential of anticipatory research and anticipatory technological development.

2 Design Challenges and Design Knowledge

The following discussion of central experienced design challenges and scientifically established insights in design methodology will form a launching platform for the proposal for the ACADA solution to be proposed in section 3.

2.1 The Systems Development Process

System (re)design may be interpreted as a step in the system development process as pictured in figure 1. In the idealistic case there is an inner loop between design and testing. In this loop the design is verified and improved until it meets its objectives. Hence, after the testing it may be ascertained beforehand that the design will bring about the desired positive effects if it is realised in the concrete real world.

There is a great difference, however, between testing new technical artefacts, for example new cars or airplanes, and developing new Human Activity Systems (HAS) or SocioTechnical Systems (STS), as for example new business firms, universities, and hospitals. In the later case there is an obvious difficulty to ascertain beforehand that a design will bring about the desired positive effects. This is due to the fact that in this case it is not possible to develop and test a prototype in the same way as when working with a nonliving artefact. Hence, a new testing procedure is urgently needed for HAS designs.

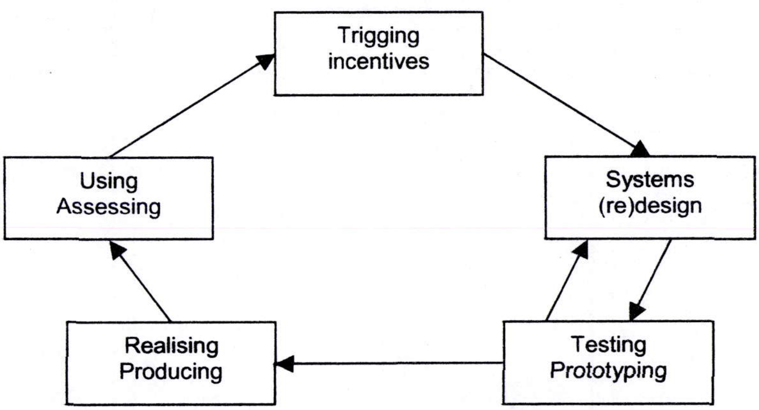

Figure 1. The system development process.

2.2 Design and Anticipation Liaisons

Concerning systemic design, Ackoff (1981) states that "*The future is largely subject to creation*", and "*the future depends at least as much on what we and others like us do between now and then as it does on what happened until now*". This means that it is necessary to develop a model (design) of the desired future and to take measures (actions) in order do attain that desired future, i.e. the design target. In terms of anticipation, this is exactly the same as prescriptive anticipation (PA) as in Holmberg (2000). Hence, there are close links between designing and anticipating, i.e., on a very basic plane designing is anticipating and anticipating is designing.

2.3 Focal Design Characteristics

Warfield and Cristakis (1987) have defined the dimensionality concept as visualised in figure 2. The main idea being that each possible design option or alternative can be grouped into its proper design dimension. For example, "white" and "blue" are two alternatives in the "colour" dimension while "steam", "gasoline", and "diesel" are three different alternatives in the "engine" dimension. All alternatives in all dimensions define the total design space. By combining one, or more, alternatives in each dimension a unique design alternative will emerge. Dimensionality has turned out to be a key concept for design work meaning that each design has its proper dimensionality. A design with too few, or even too many dimensions will not meet its objectives. Due to the close relations between design and anticipation it may be assumed that dimensionality has to be preserved also in the anticipatory phase of the system description.

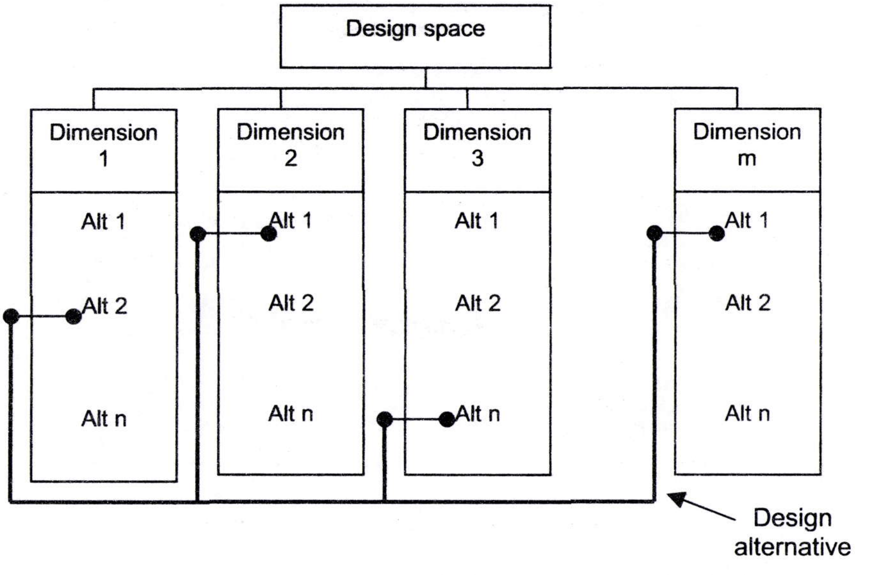

Figure 2: Design space with one design alternative example.

2.4 Focal Anticipation Characteristics

Three different types of anticipation, Exploratory (EA), Inhibitory (IA), and Prescriptive (PA), have been defined by Holmberg (2000) according to figure 3. They can easily be related to the design process as in figure 1. Hence, IA is best fitted for testing while EA and PA a most suitable for (re)design and realisation respectively.

Further, Ackoff (1981) has defined three attitudes to planning, reactive, inactive, and preactive. Here, evidently, the preactive attitude has the closest links with anticipation and design.

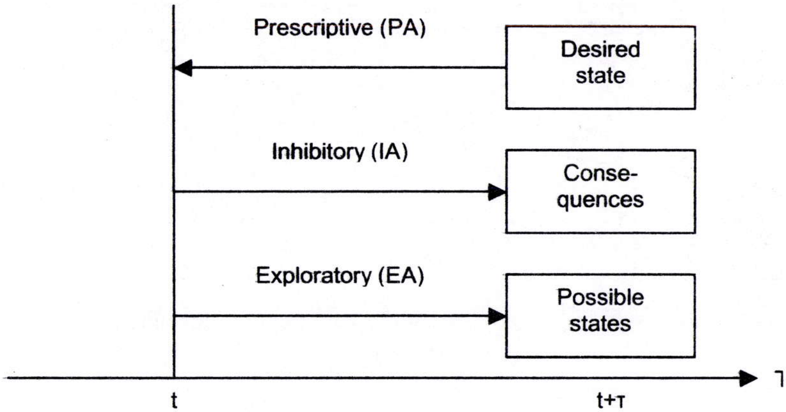

Figure 3: Different types of anticipation according to purpose.

Further, with help of Holmberg's (2000) Weighted Incursive Procedure (WIP) it becomes possible to handle those different planning attitudes in an anticipatory context. The basic idea behind the WIP approach is that the system's development is due, not only to its history, but also to its actuality and future objectives. This is controlled by three weights, the history weight (wh), the actuality weight (wa), and the potentiality weight (wp). Those weights can be associated with Ackoffs planning attitudes as in table 1. Hence, by varying the weights wh, wa, and wp it becomes possible to change between Ackoff's (1981) different planning attitudes

Table 1: Connection between planning attitudes and WIP weights.

Reactive -------- History (wh)

Inactive --------- Actuality (wa)

Preactive ------- Potentiality (wp)

The connection to anticipation may be a bit further clarified with the help of figure 4. Here the time axis is divided into three zones. With a reactive attitude you are looking backwards into your system's history, i.e., you have a high value on wh. The inactive attitude tries to keep things like they are. In WIP this corresponds to a high wa value. A preactive attitude, at last, corresponds to high wp values. In connection with anticipating the outcome of system designs, the preactive attitude with a high wp value seems to be the most relevant approach.

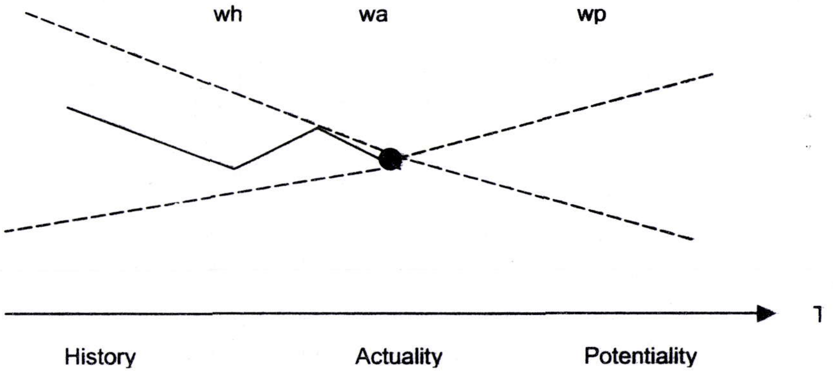

Figure 4: WIP procedure time regions.

2.5 Design Language Transformations

Warfield (2002) has stressed the importance of language in communicating and computing of information, "*in working with complexity, the choice of object language, and the choice of formal language and the choice of formalism to support that language is critical*". Hence, the choice stands between natural, formal, and hybrid language as in figure 5. The natural language being best for a comprehensive description of any aspect or perspective of any complex real life situation. It has a flexibility and richness that makes it ideal for describing complex matters. However, natural language is also ambiguous and it masks structure and enforces linearity. It is not suitable for computing.

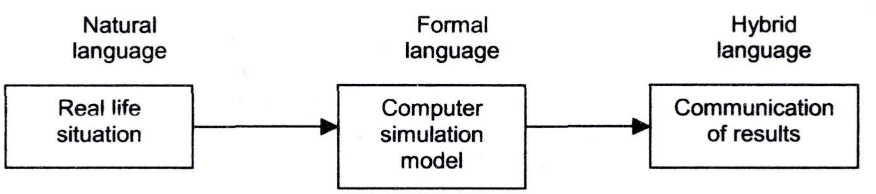

Figure 5: Language transformation.

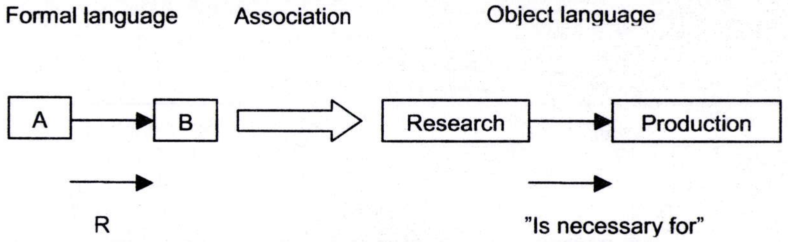

Figure 6: Association from formal to object language.

Formal languages on the other hand are normally precise and well suited for computing. As a drawback, however, they are not well suited for human comprehension and communication. Hybrid languages, i.e., combinations of prose and graphical items such as boxes and arrows, at last, are well suited for communication of computing and simulation result to the human receptors (Warfield, 2002).

This transformation between languages is made possible with the help of associations as illustrated in figure 6. Here a formal relation R between entity A and B in the left part of the figure is transformed into an object system relation by associating A with "research", B with "production", and R with is "is necessary for". This powerful mechanism will be used in the following in an illustration of a formal verification of a design with help of anticipatory computing.

3 Design with an Anticipatory Approach

Obviously it would be advantageous to be able to test and verify the design before its completion. One way of doing such a verification is to express the design as a formal system and to simulate the coming behaviour of the object system, or target system, in that formal model, possibly with help of computer support. Hence, in so doing an anticipating design system will emerge. This means that it will be possible to test, verify, and improve the design in the formal system before an expensive realisation of the object system. In figure 7, a scheme for implementing such an Anticipatory Computer Aided Design Approach (ACADA) is proposed. This can be seen as an improved or augmented design process compared to the original systems development process displayed in figure 1.

The crucial measure of that operation is to make the correct associations between the variables and operators of the formal system and the corresponding features of the object system. It is not obvious that it is always possible to do such associations but, as Warfield (2002) has pointed out, the usefulness of Bool's algebra is due to the successful realisation of such associations. Hence, I initially assume that they will be possible also in the design case.

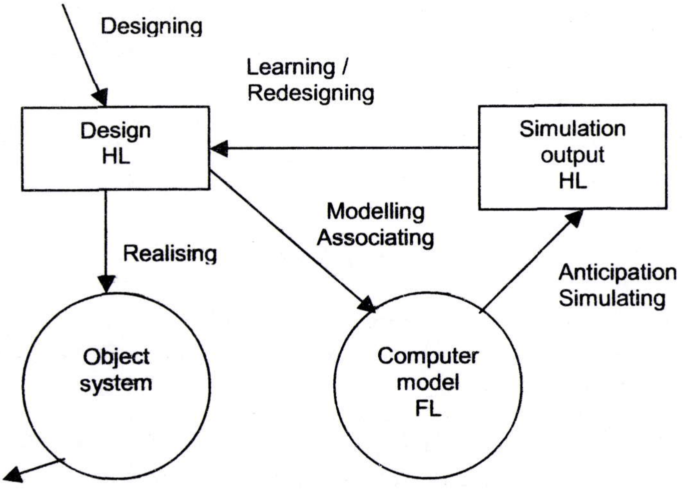

Figure 7: Anticipatory Computer Aided System Development Approach with a switching between Hybride Language (HL) and Formal Language (FL).

Putting the previous discussion together, the following iterative steps could constitute a workable procedure for supporting the design of complex Human Activity Systems (HAS) by simulations in a corresponding formal anticipatory system:

- Develop the design.
- Transform the design into a formal system
- Perform (computer supported) simulations in the formal model
- Associate variables and operators in the formal model with features and actions in the concrete system.
- Act in real system according to outcome of simulations, or redo the design or/and transformation.

Further, in many cases the original system, i.e. the system before redesign, may be what Rosen (1985) calls a reactive system. That means that the formal reactive system (sr) is developing only according to a reactive or deterministic function R, which is only reacting on the system's historical states sr(h) according to eq. 1.

$$sr(t+1) = R[sr(h)] \tag{1}$$

In this case, an actor or stakeholder has no possibility to influence the outcome of future system states. Seen in the light of systemics, this have a clear similarity with what Ackoff (1981) calls a reactive planning attitude. The (re)design (D) of R, however, could transform it into an anticipatory function (A) according to equation 2. In Ackoff's parlance, a preactive or interactive planning attitude has emerged. With this new function, future anticipatory system states (sa) may now be calculated in an anticipatory mode, i.e. due also to system objectives (o), and future system states (s(p)) according to eq. 3.

$$D:R \rightarrow A \tag{2}$$
$$sa(t+1) = A[sa(h), sa(t), sa(p), o] \tag{3}$$

In this anticipatory case (eq. 3), system targets, or objectives (o), will have an influence on future system states. This means that the system will become controllable and by associating objectives (o) in the formal system with actor and stakeholder objectives in the real world system it becomes possible to test and validate designs developed for concrete purposeful systems. This main idea will be exemplified in the following section.

4 Demonstration of the ACADA Approach

The cornerstone in the ACADA approach is its possibility to test a design in a formal system or to compare alternative designs in such a system. A simple demonstration of that ability will be given here with the Perl-Verhulst function serving as the formal system. No associations, however, will be made with a corresponding object system.

The Pearl-Verhulst function or logistic function according to eq. 4 is strictly deterministic and reactive. Further, it also exhibits a chaotic behaviour for certain values of the "regeneration" variable (r) (Dubois, 1999). However, the function may be transformed or redesigned in order to change its behaviour and characteristics. Eq. 5 shows such a redesign according to Holmberg's (2000) WIP approach. This is also a concrete example of using the computer as a laboratory for systems research, as it has been expressed by Holmberg (1999).

$$sr(t+1) = r * sr(t)\,[1 - sr(t)] \tag{4}$$
$$sa(t+1) = wh * [h(sa(t-n),..sa(t)] + wa * [r * sr(t) * (1 - sr(t))] + wp * p\,[\,(sa(t), o)] \tag{5}$$
with $wh + wa + wp = 1$

The basic idea behind the WIP approach is that the system's development is due, not only to its history, but also to its actuality and future objectives. First, to a certain part the development is given by the history weight wh and the regression function h over the system's historical system states (sa(t-n,..,t)). Next, the actuality weight (wa) determines to what degree the current system dynamics, i.e. the original function according to eq. 4, will influence the system future. The potentiality weight (wp), at last, sets the strength of the impact of the system's objectives (o) by potentiality function (p). Hence, by varying the weights wh, wa, and wp it is possible to change between Ackoff's (1981) different planning attitudes as they have been discussed above in section 2.4.

The planning or potentiality function (p) in eq. 5, of course, may take many different forms representing different designs. In this experiment, however, for illustrative reasons a very simple approach is taken according to eq. 6. The ε-value is here representing imperfection and approximation errors in the model.

$$p(sa(t), o) = [(1 - \varepsilon)(\varepsilon + o)] \tag{6}$$

The history function is given by eq. 7 with just the two latest values taken into account.

$$h(sa(t-n,..,sa(t)) = sa(t) + [sa(t) - sa(t-1)] \tag{7}$$

Further, the system objective is not necessarily fixed over time. Most often it will be changing. That change, of course, may take many different forms but is here illustrated with the simple oscillatory relation given in eq. 8.

$$o(t) = o(0) + a * \sin(f * t) \tag{8}$$

a, amplitude of oscillation
f, frequency factor

An illustration of all this put together is given in figure 8.

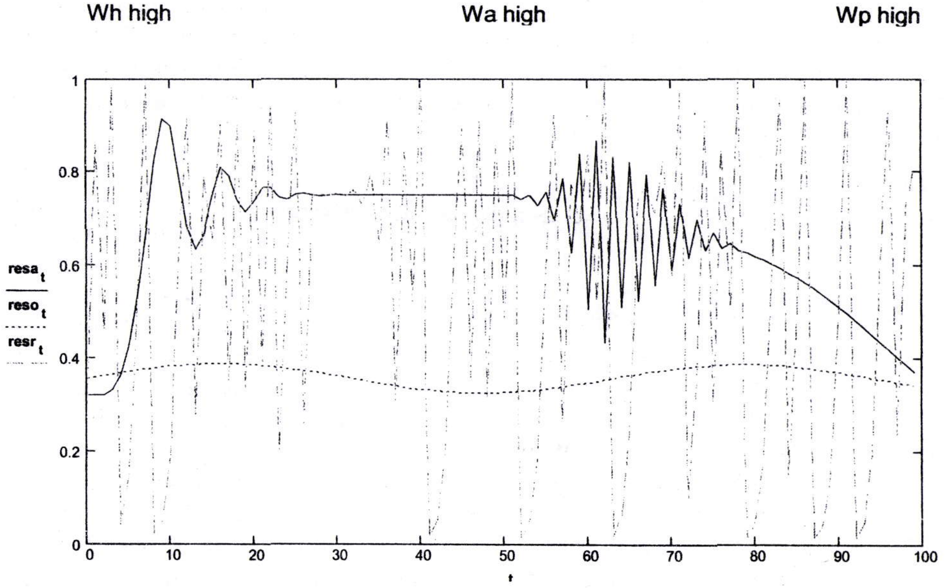

Figure 8: Example of simulation results applying the ACADA approach.

Here the objective function (reso) represents the system objective o, i.e. the objective we want the system to achieve. This function, reso, is calculated with help of eq. 8 and with $o(0) = 0.36$, $a = 0.03$ and $f = 0{,}1$.

Next, the original system, i.e. the current system before it has been redesigned is represented by resr calculated according to eq. 4. The starting value sr(0) is set to 0.32 and the r-factor has the value 3.99.

The redesigned system, at last, is represented by resa in figure 8. The function is basically calculated according to eqs. 5, 6, and 7. In addition to this, however, the weight factors wh, wa, and wp are changed in each iteration step. The simulation starts with wh = 1.0 and wa = wp = 0.0. During the first 50 iteration steps wh decreases uniformly to 0.0 while wa increases in corresponding degree to 1.0. After that, from iteration 50 to 100 wa decreases back to 0.0 while wp starts to increase from 0.0 at iteration 50 to 1.0 at the final iteration step. Even here sa(0) is set to 0.32 while the modelling error ε has the value 0.003.

The simulation results, as visualized in figure 8, demonstrate some of the main properties of the ACADA approach. First, the big differences between resr and resa shows that it is possible to simulate the result of a system redesign,

Further, different behaviour of resa during the simulation is due to changes in the WIP weights. This can be used for comparing different design alternatives but may also be interpreted as the result of different planning attitudes as discussed by Ackoff (1981). It is also worth noticing that, for most weight combinations, the redesign function resa is rather well stabilised thanks to the WIP procedure.

At last, it is also to notice that with high values on wp the resa function will approach the objective function reso. Due to an ε-value reflecting modelling errors, however, the fit will never be perfect.

However, as said in the beginning of this section, so far no associations are made between this formal system and a corresponding object system. Hence, this example indicates a potential for improving current design methods but further work remains for making ACADA a workable procedure in practical design work.

5 Conclusion

It is demonstrated that anticipatory modelling and computing has a given place in complex system design as an ACADA approach has the potential for significantly strengthen most current design methods. This, however, may be seen as just an example of a multitude of possible synergistic combinations of anticipatory modelling and computing and systemic design methods.

References

Ackoff, R. A. (1981); Creating the Corporate Future; John Wiley & Sons, New York.

Collen Arne, Gasparski Wojciech W. (Eds) (1995), *Design and Systems*. Praxiology: The International Annual of Practical Philosophy and Methodology, Vol 3, Transaction Publishers, New Brunswick.

Dubois, Daniel M. (1999); Emergence of Chaos, Diffusive Chaos and Strange Attractors in Evolving Volterra Ecosystems; I. J. of Computing Anticipatory Systems, Vol 3, 3-28.

Dubois, Daniel M. 2004; Incursive Predictive Control for Stabilizing Chaos Maps. In: Anticipative and Predictive Models in Systems Science, D. M. Dubois & G. E. Lasker (Editors), published by the International Institute for Advanced Studies in Systems Research and Cybernetics, Volume I, 2004, pp. 1-6, ISBN 1-894613-49-X

Holmberg, S. C. (1999); Multimodal Systemic Anticipation; I. J. of Computing Anticipatory Systems, Vol 3, 75-84.

Holmberg, S. C. (2000); Designing and Prototyping Towards Anticipatory Applications. In *Computing Anticipatory Systems, CASYS'99 – Third International Conference,* edited by D. M. Dubois, AIP Conference Proceedings no. 517, pp 31-44, American Institute of Physics, Melville.

Jackson Michael C. (2000), Systems Approaches to Management. Kluwer Academic / Plenum Publishers, New York.

Rosen, R. (1979); Anticipatory Systems in Retrospect and Prospect; General Systems Yearbook, Vol. 24, pp 11-31.

Rosen, R. (1985); Anticipatory Systems; Pergamon Press, Oxford.

Warfield John N., Christakis A. (1987): Dimensionality. Systems Research, Vol. 4, No 2, pp 127-137.

Warfield John N. (1990), A Science of Generic Design, Managing Complexity Through Systems Design. Intersystems Publications, Salinas.

Warfield, John N. (2002); Understanding Complexity: Thought and Behavior; Ajar Publish. Comp, Palm Harbor, FL.