

# **Why Designers Should Study Semiotics: Applications of Semiotics to User Interface Design**

Sophie Carroll

Edna Andrews, faculty advisor

Undergraduate senior thesis presented to  
Edna Andrews, Lee Baker, and Huijuan Ling to fulfill the  
requirements of graduation with distinction in Program II

Duke University

Durham, North Carolina

April 10, 2023

## Abstract

Adopting a semiotic perspective greatly benefits user interface designers, however its potential has remained largely untapped in the field of human computer interaction and user interface design. In this essay I explain the most pertinent theories of semiotics for designers, including Peirce's nonstructuralism and sign complex model, Eco's theory of sign production, critique of iconicity, and theory of interpretation, Jakobson's speech act model, Bolinger's rejection of the sign as arbitrary, and Lotman's semiosphere. I base my analysis in relevant theories of user interface design and human computer interaction (HCI) including Norman's cognitive engineering and user centered systems design models, as well as Kammersgaard's four perspectives on HCI. I synthesize these theories by analyzing existing applications of semiotics to HCI by Andersen, Nadin, and de Souza. The major themes that emerge from this analysis are frequent misinterpretations of Peirce rooted in structural semiotics, the usefulness of Eco and Lotman's semiosphere level view, the significance of viewing the interface as a mediating non-physical sign system, and the importance of using consistent logic and code within interface languages.

## **Table of Contents**

<i>Abstract</i> .....	<b>2</b>
<i>Introduction</i> .....	<b>4</b>
<i>Chapter 1: Fundamental Concepts of Semiotics</i> .....	<b>8</b>
<i>Chapter 2: Fundamental Concepts of User Interface Design</i> .....	<b>36</b>
<i>Chapter 3: Integrating Semiotic Models with User Interface Design</i> .....	<b>65</b>
<i>Conclusion</i> .....	<b>105</b>
<i>Bibliography</i> .....	<b>109</b>

## Introduction

In this thesis, I will explore the question “How is semiotics relevant to user interface designers?” To provide a foundation for my analysis, I will lay out the fundamentals of semiotics and user interface design in my first two chapters. In my third chapter, I will cover theorists who have attempted to connect the two and provide my own analysis of their applications. I will show that a semiotic perspective is incredibly helpful, even necessary, when designing interfaces; however, little research has been done to connect these two areas and I hope to demonstrate the necessity of more.

Semiotics can be and has been defined in multiple ways. One of the most succinct definitions of semiotics is Thomas Sebeok’s. Sebeok was one of the leading semioticians of the 20th century, and he describes semiotics as “the study of signification,” semiotics’ subject matter being “the exchange of any messages” (Sebeok, 1991 p.13). Sebeok explains that semiotics “never reveals what the world is, but circumscribes what we can know about it”—in other words, it is an epistemology (Sebeok, 1991 p.12). Following this line of reasoning, Edna Andrews, a semiotics scholar of the 21<sup>st</sup> century, writes that “semiotics is by its essence a theory of perception” (Andrews, 2006 p.2).

Semiotics is also important to define through the lens of cultural anthropology and literature . Umberto Eco’s semiotics definition in his introduction to Lotman’s *Universe of the Mind* represents this (Eco, 1990). Eco writes, “Semiotics aims to study the entire range of sign systems (of which verbal language is the most important) and the various processes of communication to which these systems give rise” (Eco, 1990 p.ix). Lotman expands this definition, as Eco

summarizes, when Lotman writes that “Semiotic systems are models which explain the world in which we live...in explaining the world, they also construct it”<sup>1</sup> (Lotman, 1967 in Eco, 1990 p.x). Eco and Lotman come from a background in literature, and they tend to focus on the processes and cultural impacts of impacts of semiotics—at the semiospheric level—more than other theorists like Peirce who work at the level of the sign. Eco and Lotman define semiotic systems as texts<sup>2</sup> and sets of these texts model the culture of a period (Eco, 1990 p.x). They both corroborate that semiotics is an epistemology—Eco writes “the code of a culture does not mean explaining all the phenomena of that culture, but rather allows us to explain why that culture has produced those phenomena” (Eco, 1990 p.x).

In conclusion, the commonly agreed upon characteristics of semiotics are that it is an epistemology that describes and models perception, usually using the terminology of signs and sign systems. The major theories of semiotics I will be covering that are relevant to user interface design include Peirce’s nonstructuralism and sign complex model, Eco’s theory of sign production, critique of iconicity, and theory of interpretation, Jakobson’s speech act model, Bolinger’s rejection of the sign as arbitrary, and Lotman’s semiosphere.

User interface design (UID) explores how to create interfaces that are the useful to a people who interact with them. I will focus specifically on user interfaces encountered on a computer (digital

---

<sup>1</sup> This is similar to Peirce’s perspective as interpreted by Sebeok, in which the sign model describes the “role of the mind in the creation of the world” (Sebeok, 1991 p.20). This perspective represents semiotic idealism, which postulates that signification is not just how we perceive the world, but the actual process of creating reality (Sebeok, 1991 p.20).

<sup>2</sup> The term text has a different meaning in semiotic terminology than in everyday usage—semiotic texts are not only written materials, but signs of any medium like speech or dance. As Lotman says, a text is a piece of “condensed cultural memory” (Long, 2019 p.373; Lotman, 1990 p.18). Eco expands that “texts represent [semiotic] models of the world” (Eco, 1990 p.x) and are machines for generating interpretations (Andersen, 1990 p. 130).

interfaces). I will use the abbreviation UID to refer to user interface design throughout, not to be confused with the UI in UI/UX. Human computer interaction (HCI) models the relationship between users, designers, and computer systems, and I will also cover the main models of HCI that contribute to UID. These models include Norman's cognitive engineering and Kammersgaard's four perspectives, with input from Carroll.

When I talk about the design of computer systems, I will focus on the interface. Andersen explains that the "interface is the ultimate success or failure of system" (Andersen, 1990 p.311); further, "the basic function of a system is to generate processes in which an interface can be expressed, just as the only reason for having costumes and wigs in a theater is the experience they may contribute to giving the audience" (Andersen, 1990 p.174). While I will define interfaces as relationships between users and computers and systems of signs that are not physical, examples of the physical manifestation of the interfaces I am covering include operating systems like the Apple or Windows desktop, which encompass how files and system preferences are visually organized, how different applications appear, and any other directly perceptible aspect of operation. They also include applications like Word and Photoshop, and more specific aspects of these applications like the process of bolding a sentence in Word or using the Photoshop paint bucket tool.

After I elaborate the central theories of semiotics and interface design, I will cover three authors who have connected semiotics and UID; Andersen, Nadin, and de Souza. These authors grapple with binary and triadic sign systems, nonstructuralism and emergent meaning, and speech acts they find in the relationships between users, designers, and computers. While Peirce is frequently

misinterpreted, an understanding of nonstructuralism, semiosis, and emergent meaning proves to be essential when designing for diverse users with different goals. Structural binary signs are also helpfully used to differentiate types of representations within interfaces. A semiosphere-level view of semiotics is also utilized to look at the relationships between signs and the interface within wider cultural code.

I have specialized in verbal and visual design while completing my degree in semiotics, and the conceptual overlaps between visual and interface design and semiotics have always struck me as plentiful and essential to explore. However, there is a very small body of research connecting the two, and in this essay I hope to lay a basis for future research to develop. Many UI design guidelines are relatively unstructured sets of do's and don'ts, found through retroactive testing and research in design and cognitive science (de Souza, 1993 p.753). While HCI models the relationships in problems and successes emerge, it is very difficult to use HCI models to explain the theory behind these interactions (de Souza, 1993; Carroll, 2003). Semiotics can bridge this gap. Semiotics expands the HCI view of the design process to include the relationship between culture, code, and all users, designers, hardware, and software. The main argument of this paper is that user interface designers should adopt a semiotic perspective because it expands the scope of what HCI models can tell us, identifies user expectations, and enables the designer to have a proactive qualitative perspective that counters retroactive, quantitative analysis.

## Chapter 1: Fundamental Concepts of Semiotics

“Following Peirce and Sebeok, we can define semiotics only in terms of the system of dynamic signs in interaction. This brings us to semiosis itself. Semiotics, as 'doctrine' (to use Locke's term), where the primary goal is to adequately represent the dynamic sign in semiosis, will ultimately focus on the production of the interpretant and its affiliation with the sign-object (in Peircean terms) (1931-58: 5.473).” (Andrews & Lotman, 2003 p.53)

While this essay will explore applications of semiotics to user interface design, it will be primarily about semiotics. The following section will cover the central terminology, paradigms, and scholars of semiotics.

### **1.a What is semiotics? Key terms and fundamentals**

#### **History of semiotics**

The earliest appearance of the term semiotics occurred in Ancient Greece in the 4th century BCE, when Aristotle published his work Rhetoric which discussed “semeions” at length (Aristotle, C.350 BCE; Aristotle, 1966; Di Piazza & Piazza, 2016; Deely, 1990 p.109). Aristotle wrote, “something can be considered a semeion of something else if it regularly occurs before or after it” (Di Piazza & Piazza, 2016 p.10). In other words, a semeion is associated with and signifies this “something else.” Aristotle developed multiple classes of signs including tekmaria and other non-named signs with different relationships to a designatum, which functioned similarly to a signified meaning (Di Piazza & Piazza, 2016).



Theories of signification remained undeveloped for many centuries after Aristotle until John Locke, a British philosopher working in the 17th century, brought the Greek term “σημειωτική” (semiotikes) back in his *Essays on Human Understanding* (Locke, 1689; Deely, 1990 p.113). Locke defined semeiotikes as the study of “the nature of the signs that the mind makes use of for understanding things and for conveying its knowledge to others” (Locke 4.21, 1689; Locke, 2017 p. 289). The signs in question in this “doctrine of signs” encompassed “mostly words” (Locke 4.21, 1689; Locke, 2017 p. 289).

The term semiology has also been used to describe semiotics. It was popularized by Saussure, but is not a perfect synonym of semiotics; semiology refers to specifically Saussure’s binary sign system (Deely, 1990 p.114). Some semioticians like Barthes and Eco continued to use this term, but it is less general than semiotics (Barthes, 1957; Eco, 1968; Eco, 1990).

### **Primary divisions within semiotic theory**

Scholars of semiotics take different stances on a few key issues. One of the most important is structuralism vs. nonstructuralism. Structuralism is a “universal semiotic theory (and method) whereby the rules governing each communicative sector [are] seen as variations of more general codes... or rather as a system of rules” (Eco, 1990 p.ix). Structuralism has been widely used in describing linguistics and cultural phenomena. Saussure represents a pure structuralist view, in which a certain signifier will mean the same signified to all people within one language system (Deely, 1990 p.115). Lotman, Jakobson, and Eco also began as structuralists. Lotman and Eco became less structural over time as a result of exposure to Peirce. Eco pinpoints this change to the 1960s, when even though Lotman “stressed the usefulness of the structural approach,” he

began to discover problems in structuralist theory for dealing with sign systems that change (Eco, 1990 p.ix). Additionally, as Eco wrote, structuralists faced the problem of how people with different codes could communicate (Eco, 1990 p.ix).

These problems within structuralism were rectified by Peirce, who is widely recognized as being the first nonstructural semiotician. In nonstructuralism, sometimes referred to as post-structuralism, sign systems do not identify rules that result in predictable or repeatable meanings. Meaning is not inherent in the sign, but emergent only in combination with the addressee, addresser, and context (Andrews, 1990 p. 56). The presence of perceiver (also called the addressee, user, etc.) within the nonstructural sign system is one its most important differences from structuralism. Nonstructuralism had an enormous effect on how semioticians viewed sign systems, and initiated the user inside the sign complex and emergent meaning.

Another division in semiotics is the binary vs. triadic sign model. Once again, Peirce generated this division by postulating a triadic sign model when most other semioticians had accepted a Saussure-like binary model. Peirce's sign complex, which I will review extensively later, is irreducibly triadic, meaning it can never be described in binary terms. If a semiotician works with a triadic sign model, they are most likely following Peircean semiotics.

Semioticians are also split on the arbitrariness of the sign. Saussure popularized the idea that the sign is binary; that is, the sound of the word "apple" has come to represent an actual apple in the addressee's mind through learned cultural association and bears no physical or qualitative resemblance to an actual apple. Saussure also explains this theory inversely by theorizing that the

signifier signifies the signified simply because the signifier doesn't signify anything else. Bolinger's *The Sign is Not Arbitrary* rebukes this line of semiotic thought (Bolinger, 1949). Bolinger critiques comparative linguistics for supporting Saussure's line of thought because the field studies words across language systems. Linguists like Saussure saw that words like "perro" in Spanish and "dog" in English were different, so they assumed the signs were arbitrary. But Bolinger argues that we need to analyze signs at the single language (system) level, where they are not arbitrary (Bolinger, 1965). For example, morphemes like "s" alter meaning (plural) across multiple words in the same language, and sign forms like homophones or homonyms alter language user behavior within a single language system. Signs are not arbitrary within their own sign systems.

One last distinction is semiotic theories that focus on the individual sign model vs those that focus on a larger, semiosphere level scale. I will expand more on the semiosphere later, but this second version fits with Eco, Lotman, and Sebeok's view of semiotics as a theory of culture and larger sign systems. Peirce, Uexküll, and Jakobson, however, fit in the first definition.

A few semiotic schools of thought exist which represent these splits. They include the Tartu-Moscow, Copenhagen-Tartu, and Prague school. The Tartu-Moscow school focused on a structuralist semiotic approach to literary and cultural studies in the 1960s and 70s, and was founded on the works of Lotman (Salupere et al., 2013). The Copenhagen-Tartu school succeeded the Tartu-Moscow school and focused on developing the field of biosemiotics, or the application of semiotics to nonhumans and biological processes, in the 1990s (Hoffmeyer & Kull, 2011). It used the works of Lotman, Peirce, and Jakob von Uexküll as theoretical bases

(Hoffmeyer & Kull, 2011). The Prague school was a school of structural linguistics that studied the elements of language and analyzed linguistic sound in the 1920s, in which Saussure was frequently discussed (Eco, 1990; Makaryk, 1993). The Prague school included the Russian-American linguist Roman Jakobson, who I will also cover.

### **Semiotic approaches to meaning(s)**

Meaning is a central concept in semiotics, but how it is generated varies across thinkers and theories. In Saussurian thought, meaning is identical to the signified, and in additional structural theories meaning was defined as the final, static, and pre-constructed result of the signification process of language (Saussure, 2011).

Lev Vygotsky, a Russian psychologist and semiotician known for his 20<sup>th</sup> century cultural analyses, defined meaning as the product of signification which is always dynamically changing (Vygotsky, 1987). In other words, meaning dynamically changes in the direction of signification (Vygotsky, 1987). In Vygotsky's perspective, meaning is inconsistent, revealed through generalization, and is always in process (Vygotsky, 1987 p. 249). It is a product of the unity of word and thought, which cannot be separated; Vygotsky only focused on verbal language, and he is known for his theories of universal human languages (Vygotsky, 1987 p. 244). This approach to meaning was somewhat less structural than Saussure's, but still suggested meaning was contained in the word-thought sign system.

In Peircean thought, which I will rely heavily upon in this paper, meaning is not a structural property of sign systems but emerges as a result of the sign in action (Peirce, 1957; Andrews

1990). Not only is meaning emergent, but it is plural, negotiated, and dynamic; hence, there is not one meaning, but many meanings. This process is called semiosis, and is the product of a flow of interpretants (I will define interpretants later) which never stops generating additional meaning (Peirce, 1957; Andrews, 1990 p.57). This is a direct product of Peirce's nonstructural perspective, in which meaning is not contained in the sign but relies on its interlocuters and context. According to Andrews, Peircean meaning is plural, historically situated, negotiated, and dynamic (Andrews 1990).

Eco's exploration of meaning in *Interpretation and Overinterpretation*, written late in his career and thus more nonstructural than his early works, represents a similarly nonstructural Peircean perspective. Eco writes that a text does not contain any inherent meaning until it is interpreted by a reader (Eco, 1992 p.64). The author of the text (or addresser) cannot determine the meaning the text will have, even if they produce it with intended meaning—this is compatible with Peirce's theory of emergent meaning.

A few nonstructural semioticians also approached, directly or indirectly, meaning as emergent and impacted by the context and addressee. Lotman supported the importance of context to meaning when he wrote that texts generate new meaning when they cross semiospheric borders (for example, being inserted into a new text or culture) (Lotman, 1988 p. 69) In Uexküll's biosemiotics, no object has meaning until it is perceived by the perceptual organ, at which point it becomes imprinted with meaning (Uexküll, 1982 p.31). Jakobson also falls in this category of thinkers, and meaning emerges from his speech act model as a property of six functions

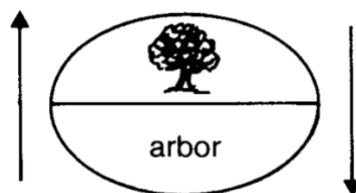
including context, addressee, addresser, message, contact, and code—I will define each of these functions later (Jakobson, 1960).

### **1.b Thinkers and theories**

While semiotic theory can be traced back to Aristotle and John Locke, in its contemporary conception semiotics was established through the work of Ferdinand de Saussure, Charles Sanders Peirce, Roman Jakobson, Dwight Bolinger, Thomas Sebeok, Juri Lotman, and Umberto Eco.

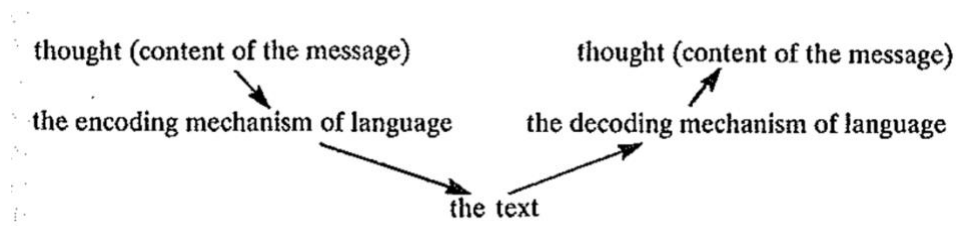
#### **Ferdinand de Saussure**

Saussure was a Swiss structural linguist who lived from 1857-1913. He is most known for “A Course in General Linguistics,” which is a compilation of his lectures by his students published posthumously in 1916. Saussure became a household name in linguistics after Jakobson popularized him within the Prague school of linguistics in the early 20<sup>th</sup> century. Saussure’s influential theories include his signification model and theory of langue and parole (Saussure, 1916 in Baskin, 2011). Saussure’s basic signification model stipulates that each sign contains a signifier (for example, the word “arbor”) and a signified (an actual tree), and that the structure of language is the primary form in which a message is signified (see diagram below) (Saussure, 1916 in Saussure, 1986 p.67).



### Saussure's binary sign model

Lotman also diagrams Saussure's model to represent three ways it can be criticized (see diagram below) (Lotman, 1990 p.11). In Lotman's diagram, text represents signified and language represents signifier. Lotman's first critique is that Saussure's model assumes thought and language are independent, when in reality thought and language depend on and cannot exist without the other (Lotman, 1990 p.12-13). This is because it is impossible to translate a thought exactly into language—thought is a different medium with different organization. Second, Saussure's model assumes the goal of a communication act is to perfectly transmit a signified message from the addresser to the addressee. However, two people will never share the exact same code or cultural memory, so this is impossible. Third, this model reduces the function of the communication act to only conveying intended signified messages, and this loses the potential of signification to create new messages and be creative. Despite its critiques, Saussure's early structural model was important in early semiotic understanding and is still often cited in popular culture.



Lotman's diagram of Saussure's sign model

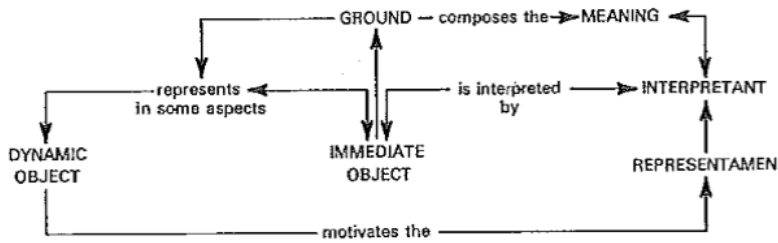
Another important idea Saussure popularized is the arbitrary (binary) sign. That is, the sound of the word “apple” has come to represent an actual apple in your mind through learned cultural association and bears no physical or qualitative resemblance to an actual apple. Saussure also explains this theory inversely by writing the signifier signifies the signified simply because the signifier doesn’t signify anything else. This concept will later be critiqued by Dwight Bolinger in his essay “The Sign is Not Arbitrary” (Bolinger, 1949).

### **Charles Sanders Peirce**

This essay will be based primarily on Peirce’s triadic theory of signification, although I will reference others. Peirce was an American philosopher who lived from 1839-1914. He is known for his work *Essays in the Philosophy of Science*, which was published in 1957 but written 50 years prior, and was not well known until Roman Jakobson began to interpret Peirce in mid-20<sup>th</sup> century (Peirce, 1957). Peirce’s signification model, or triadic sign complex, was groundbreaking in semiotics because it was nonstructural—it does not suggest there is any inherent rule-based meaning in a particular signification act—and more comprehensive and more generative than previous models. It highlighted the role of context, addressee, interpretation, and emergent meaning in signification where Saussure’s model ignored them.

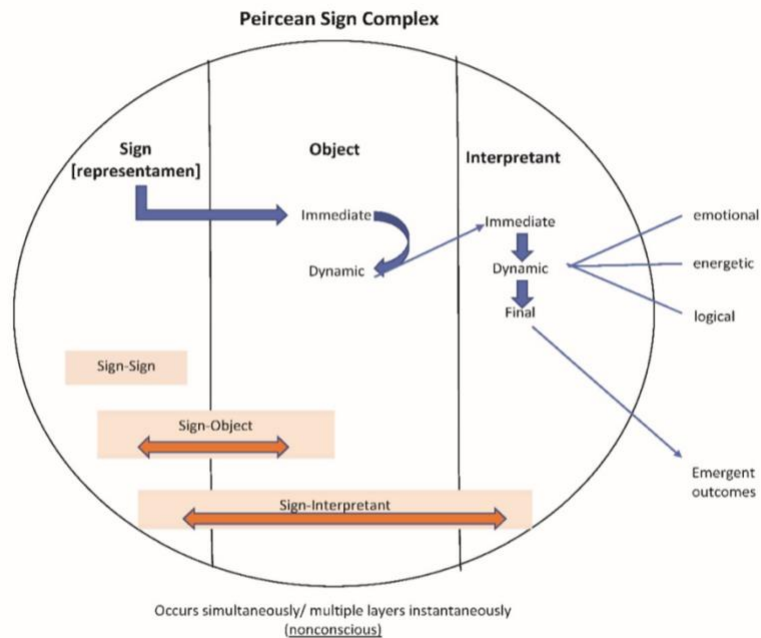
Following Jakobson’s introduction of Peirce into semiotic canon, the other best known interpreter of Peirce is Eco, and later Savan and Andrews. Eco was one of the first writers to try and model Peirce’s sign complex in his book *The Role of the Reader*, as shown below (Eco, 1984 p.183). Eco’s model contains Peirce’s concept of ground, which was an important early focus of his but one which Peirce later discarded and thus is not a focus of mine.





Eco's model of Peirce's sign complex

While the above model is a helpful look at one of the first attempts to diagram Peirce, I prefer the following model by Andrews because it incorporates Peirce's concept of firstness, secondness, and thirdness (Andrews et al., 2023 p.6). I will explain the parts of this diagram and Peirce's sign complex below.



Peirce's sign complex, as modelled by Andrews et. al

Firstness, secondness, and thirdness are natural, unlearned categories that contain each process in Peirce's sign complex (Andrews, 1990 p.47). Firstness represents anything that is a mere possibility, unrealized but with the potential to become realized, and contains Peirce's representamen. The representamen "determines something else (its interpretant) to refer to an object to which itself refers (its object)" (Peirce 2.303). The representamen is sometimes called the sign, but I will refer to it as the representamen in order to not confuse it with the sign complex. The representamen is where signification begins, and it is where the qualities necessary for signification are present but the meaning of the signification act is not yet realized.

Secondness is where signification starts to come into existence and can be experienced. Peirce's object belongs to the secondness category, and represents the actual "temporal and spatial" form of the sign (2.305). There are two parts of an object; first, the immediate object which is the actual material of the sign (for example the phonemes in a verbal utterance) (Savan, 1980 p.256). Second is the dynamic object, which includes the utterer of the sign (for example, the person who produced the verbal utterance) and all language users and hearers (Savan, 1980 p.256).

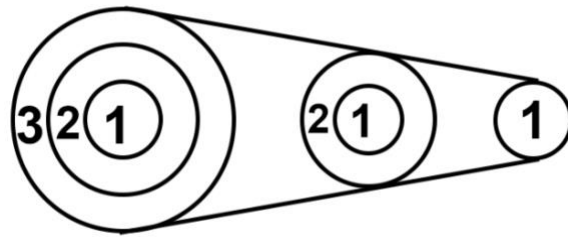
Thirdness is a category containing anything that is general and repeated rules. Eco distinguishes secondness and thirdness with the example that a piece of legislation is secondness, but law and order are thirdness (Eco, 1994 p.192). In Peirce's sign complex, an interpretant belongs to the thirdness category and is the form of the sign that can actually be interpreted by interlocutors. For example, if the word "apple" is uttered it is the repeated learning and cultural knowledge that allows us to interpret it as a representation of a physical apple. Interpretants mediate between the sign's object and the sign's translation, and an interpretant contains all sign associations and

signs associated with the original sign (Andrews, 1990 p.47; Sebeok, 1991 p.18). Meaning, although it is the complete immediate object, is only identifiable through a “flow of interpretants” (Peirce, 1931-1958; Bianchi 2015 p.7). There are three types of interpretants: immediate, dynamic, and final. Immediate interpretants are the total unanalyzed impression of a sign, or the impression a sign is expected to make. Immediate interpretants have “a propensity to replicate the same interpretation in further circumstances of a like kind” (Savan, 1980). Dynamic interpretants are what the user actually makes of a sign in one instance (8.135)<sup>3</sup>. There are three types of dynamic interpretant: emotional (qualities), energetic (actions), and logical (thoughts) (Andrews, 1990 p.66). After the dynamic interpretant, there is the final interpretant. Final interpretants are relatively final conclusions of the sign process, can be reanalyzed, and can be shared by communities. Peirce calls final interpretants “living habits” (8.184) Peirce mentions the existence of an ultimate logical interpretant that is the conclusion of the sign process and which is the final meaning, but because final interpretants continuously generate further signs and associations, the ultimate logical interpretant is an unachievable ideal.

In Peirce’s theory of firstness, secondness, and thirdness, each cannot exist without the presence of the lower categories. For example, without the representamen (firstness) and object (secondness) of a sign, the interpretant (thirdness) cannot exist (see my diagram below). Accordingly, without the representamen (firstness), an object (secondness) cannot exist. Even though each level must contain all below it, one has to keep in mind that none of them can exist independently, which is why I have illustrated them in perspective as parts of the whole complex (the circle on the left containing 1, 2, and 3).

---

<sup>3</sup> Volume#.subsection# is a citation method commonly used among scholars of Peirce, where volume# refers to the volume in The Collected Papers of Charles Sanders Peirce published by Harvard (Peirce, 1931 in my bibliography).

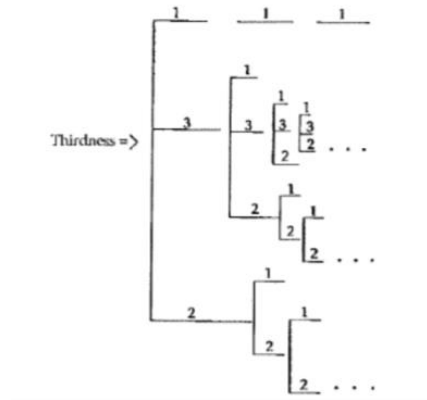


Peircean firstness, secondness, and thirdness

I have included this simple diagram as an introduction to firstness, secondness, and thirdness, however, each category dynamically also generates each category it contains. This means that the emergent meaning of Peirce's sign complex is always growing<sup>4</sup>. For example, thirdness produces more signification in the form of firstness and secondness and thirdness, and within that secondness generates firstness and secondness (Andrews, 1990 p.48). Firstness does not contain any other categories and is thus passive while secondness and thirdness are dynamic. A more complex version of the diagram above by Andrews is shown below, which shows the dynamic properties of firstness, secondness, and thirdness (Andrews, 1990 p.48).

---

<sup>4</sup> See my next footnote on symbols for more detail on the generative nature of Peirce's sign complex. The generativity of the sign is also studied by Barthes in his 1957 work *Myth Today*, albeit in a decidedly structuralist and Saussurian way. Barthes builds on Saussure's signified-signifier=sign model by designating that model as a first level sign, which leads to a second level sign in which the first level sign is a signifier itself for a more complex signified. This secondary sign Barthes calls cultural myth.



The dynamic generativity of firstness, secondness, and thirdness, as modelled by Andrews

Three modes of inference correspond with these categories. Abduction (effect to cause) corresponds to firstness, and induction (multiple effects to one cause) corresponds to secondness (Andrews, 1990 p.47). Both are synthetic modes of inference. Deduction (cause to effect) corresponds to thirdness, is analytic, and is the strongest mode of inference (Andrews, 1990 p.47).

Some other important parts of Peirce's sign complex are the categories of sign-sign (also known as the trichotomy of grounds, which contains qualisigns, sinsigns, and legisigns), sign-object (containing icons, indexes, and symbols<sup>5</sup>), and sign-interpretant (containing rhemes, dicents, and arguments) (Andrews, 1990 p.50). Each of these are categories of the representamen (sign) in relation to its representamen, object, or interpretant respectively. The sign-object relationship is by far the most well known of these, and its icon, index, symbol trichotomy has been a popular

<sup>5</sup> Symbols are especially important in Peirce's sign complex because of their dynamic generativity—they have the property of thirdness with the sign-object, so they are constantly growing (see the diagram of growing thirdness above). A symbol is a "rule that will determine its interpretant" (2.292), or something that *denotes* meaning but is not meaning itself as the interpretant is (2.302).

but often misunderstood categorization that Eco will critique in his book *A Theory of Semiotics*. According to Peircean theory, an icon is a “relation of likeness between a sign and its object,” an index a “real connection between the sign and its object,” and a symbol a “relationship between the sign and its object [that is] a general rule” (Andrews, 1990 p.59). However, Jakobson interprets these relationships in the context of Saussure’s signifier-signified model, eg. a signifier is an icon if it bears qualitative resemblance to the signified. This discrepancy likely contributed to the later misinterpretations of Peirce’s icon-index-symbol categories.

To summarize, Peirce’s triadic sign model contains a representamen, object, and interpretant existing simultaneously in the process of signification. While Peirce models one sign, many signs are in action in one communication act, which is undergoing the process of semiosis. In contrast to earlier structural models, Peirce does not suggest a sign has inherent meaning, but that meaning only emerges when a user engages in the process of signification. For example, the written word “apple” has no meaning until someone reads or hears it, and their emergent meaning may be different than the next reader or hearer (Eco, 1992). This theory of emergent meaning will become central to semiotic and cultural semiotic studies.

Another idea Peirce is well known for is his theory of fixation of belief. This theory postulates that people see continuity even if it’s not there, which will later support Eco’s analysis of hermetic thought in *Interpretation and Overinterpretation* (Peirce, 1957 p.10-12). However, Peirce also writes that belief is not fixed, and he stresses that belief is in a dynamic relationship with doubt and inquiry, which arise when we encounter conflicting beliefs (Peirce, 1957). There are four types of belief we can encounter: first, tenacity, in which the holder of the belief ignores

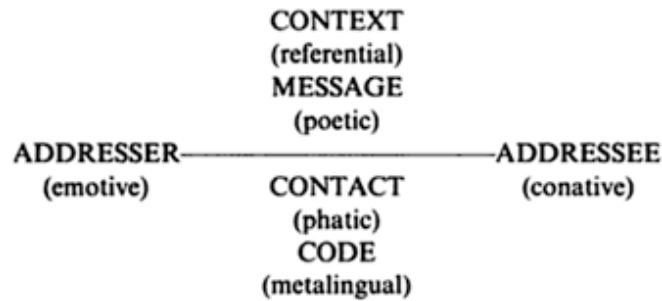
all challenges to the belief to avoid the discomfort of having to shift beliefs. Peirce uses the analogy of the ostrich with its head in the sand (Peirce, 1957 p.12). Second is authority, in which figures in authority position tell us to believe something for fear of consequences. Peirce gives the example of Roman emperors or religious wars (Peirce, 1957 p.13). Third, the a priori method “[promises] to deliver our opinions,” which meaning that we choose to believe phenomena supporting our implicit biases (Peirce, 1957 p.17). Peirce gives the example of scientists seeing natural phenomena support celestial authority in the Middle Ages. The fourth method is the “method of science,” or the scientific method in which a hypothesis is tested empirically. The fixation of belief fits within Peirce’s larger argument that signs are always growing because beliefs are constantly being challenged by new perceptions and being re-formed to fit our models of the world around us.

### **Roman Jakobson**

The next major thinker in semiotics I will cover is Roman Jakobson. Roman Jakobson lived from 1896-1982, and was born in Russia but lived in Europe until his immigration to the United States in 1941. He is known for establishing semiotics in the West, popularizing Saussure and Peirce, recruiting Lotman and Bolinger, becoming a world-renowned linguist, and his structural speech act model (Tribus, 2017 p.4 adapted from Jakobson, 1995 p.73). Jakobson’s most famous works are *On Linguistic Aspects of Translation* (1959)<sup>6</sup>, *Linguistics and Poetics* (1960), and *On Language* (1995).

---

<sup>6</sup> My citation of this work from 1971 in my bibliography will be the easier to find version.



Jakobson's speech act model

Jakobson's speech act model, as shown above, represents the generation of meaning in not only language but any semiotic process (Daylight, 2017 p.12). Jakobson's model is unique because it covers a whole speech act made of many signs, as opposed to the singular sign unit that the Peircean model covers. Jakobson's six factors (addresser, addressee, context, message, contact, and code) are the essential components of a communication act, and each has a corresponding function (emotive, conative, referential, poetic, phatic, metalingual) as shown in Andrews' diagram below (Jakobson, 1995; Andrews, 1990 p.75).

The addresser initiates the speech act and performs the emotive function, which demonstrates emotional reactions and is an "expression of the speaker's attitude" (Jakobson, 1960 p.66 in Andrews, 1990 p.75). The emotive function can be considered in Peirce's category of firstness according to Andrews, because it represents "the possibility that some quality may be abstracted or isolated" (Shapiro 1983 p.30 in Andrews, 1990 p. 75). The addressee perceives the speech act, and performs the conative function or "call to action" that is often "vocative and imperative" (Jakobson, 1960 p.67-68 in Andrews, 1990 p.75). The conative function is one of Peircean secondness. Next, the context factor corresponds to the referential function, which describes and



orients the addressee to a situation. The referential function also fits Peirce's definition of thirdness by bringing "a second and third into relation to each other" (Peirce, 1957; Andrews, 1990 p.75). Next, the message factor has a poetic function. The poetic function gives a speech act its artful function, as found in poetry, rhymes, puns, etc. The factor of contact has a phatic function, which concerns the opening or closing of the communication channel through eye contact, tone of voice, body language, etc. Last, the code factor has a metalingual function. Metalingual denotes discussing language itself using language, and when addressees and addressers share a code it means they share a language which allows them to interpret the message. Any community's code is always made up of smaller codes (Jakobson, 1995 p.72). All factors and functions in Jakobson's model are present during every speech act, but they are always in a varying hierarchy of importance.

### **Dwight Bolinger**

Dwight Bolinger was an American sociolinguist who lived from 1907-1992, and he is known for his works *Visual Morphemes* (1946), *The Sign is Not Arbitrary* (1949), *Forms of English* (1965), *Truth is a Linguistic Question* (a speech he made as president of the Linguistics Society in 1973), and *Intonation and Gesture* (1983). Bolinger is known for his work in sociolinguistics, and he believed that the border between the humanities and sciences should be eliminated and that semiotic analysis could bring them together. Bolinger's work is relatively structural.

Bolinger's *The Sign is Not Arbitrary* (1949) is a 20<sup>th</sup> century response to the 19<sup>th</sup> century assertion Saussure popularized that the sign is arbitrary. I covered this split in more detail above, but this means that the form of a sign and its meaning have no perceptual similarity. Bolinger

critiques comparative linguistics for supporting Saussure's line of thought by studying words across language systems, when they should be studying single language systems in which signs are not arbitrary.

Some of Bolinger's other important ideas are that intonation is part of gesture, and that we should take gesture seriously as a linguistic phenomenon (Bolinger, 1983). He also stressed that communication never presupposes truthfulness—for example, we have words which function equally well for “unicorn” and “horse,” as well as lies and truths (Bolinger, 1973; Bolinger, 1965). Bolinger also stressed that language can be used to abuse meaning and truth and urged linguists to apply linguistics in a way that makes the world a better place<sup>7</sup> (Bolinger, 1973).

### **Jacob von Uexküll**

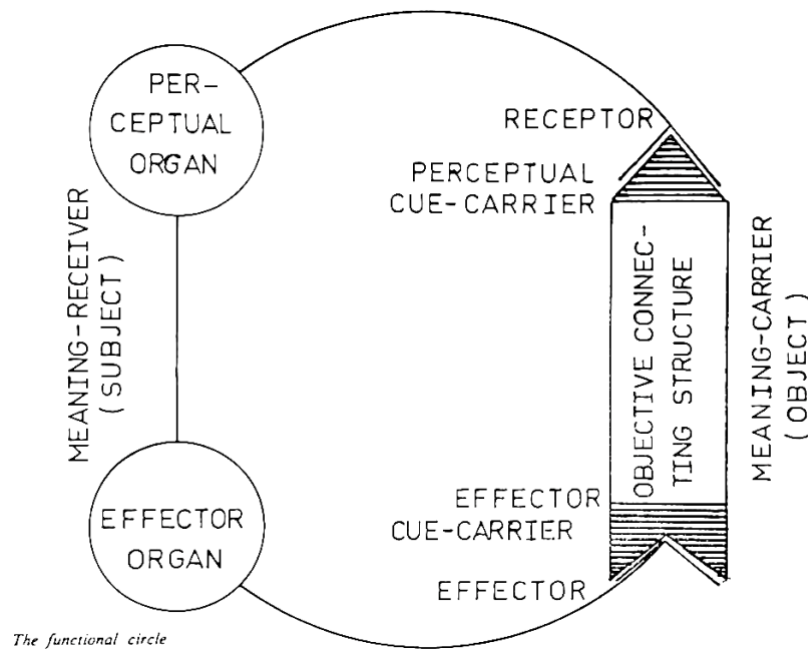
Uexküll was an important theorist who developed the field of biosemiotics. Biosemiotic theory was an important development in semiotics because it expanded sign systems from a solely human property (called anthroposemiotics, which was the leading form of structural semiotics) and vastly broadened the scope and subjects of semiotics. Uexküll was a German biologist who lived from 1864-1944. He is known for his work *The Theory of Meaning* (originally written in 1934 and published in English in 1982), in which he developed his influential theory of the *Umwelt*. Uexküll coined many new terms in theoretical biology that represented signification among animals, and while he did not include humans in his theory of *Umwelt*, his ideas have also been widely applied to human signification. Uexküll rejected positivism, the theory that

---

<sup>7</sup> Much in the way I have called on designers to use semiotics to make interfaces better for users.

everything can be explained by reason and logic, and was a naturalist who believed all animals were created with a purpose (von Uexküll, 1982).

Uexküll's umwelt is the external world known to an organism as perceived through its sense organs (von Uexküll, 1982 p.30). The internal world of the organism is its innenwelt. The collection of all umwelts, or objective world, is the umgebung. The umgebung is impossible to perceive. Uexküll illustrates how organisms subjectively perceive the environment and how perception determines behavior using his "functional circle" model (Von Uexküll, 1982 p.32).



Uexküll's functional circle

In Uexküll's model, the organism's perceptual organs perceive something in the *umwelt*, and report a perceptual cue which the internal receptors receive. The object that the perceptual cue came from in the *umwelt* becomes a meaning carrier once it is perceived, and an internal connecting structure in the organism creates an effector-cue, which directs the internal effector system to direct the effector organ to act upon the meaning carrier. The object is then affected and, as we see in the diagram, the circle process of perception continues.

### **Thomas Sebeok**

Thomas Sebeok was a Hungarian and American semiotician who lived from 1920-2001. Sebeok was both an interpreter of Peirce and another developer of the field of biosemiotics, influenced by Uexküll's work. Sebeok is known for his works *How Animals Communicate* (1977) and *A Sign is Just a Sign* (1991).

One of Sebeok's important contributions is the notion that semiotics is an epistemology, not an ontology (Sebeok, 1991 p.18). Semiotics is an epistemology because it describes how we can mentally model the world but does not tell us what it is. In Sebeok's view, semiotics cannot be reduced to physics, but it can decrease entropy and disorganization (Sebeok, 1991 p.14). Like Jakobson, Sebeok stresses the importance of context in its ability to override a message and regulate communication systems (Sebeok, 1991). Sebeok defines a message as a sign transmitted from a producer to a receiver (Sebok, 1991 p.13).

Sebeok stresses that semiotic processes are present in non-human animals, whether verbally, biologically, or chemically (Sebeok, 1991 p.16). He discusses two biological sign categories that

originated with Hippocrates: natural signs that are universal and biologically determined, and conventional signs that are culturally agreed upon and spoken or written (Sebeok, 1991 p.24).

## **Juri Lotman**

My next central thinker in semiotics is Juri Lotman. Lotman was a Russian and Estonian semiotician who lived from 1922-1992 and founded the influential Tartu-Moscow school of structural semiotics. Much of Lotman's early work was not widely read until it was republished in English, which happened many years after the original pieces were written because of censorship in the USSR. Lotman is known for his works *Universe of the Mind* (1990), *On the Semiosphere* (first published in Russian in 1984 but published in English in 2005), and *Culture and Explosion* (2009). Lotman was influential in the field of cultural semiotics, in analysis of oral and written literature, and in his theory of the semiosphere. His early works focused more on literature than semiotics, and Lotman began as a structuralist; only by the time he had written *Culture and Explosion* did Lotman become a significant player in semiotics who engaged with mainstream and Peircean semiotics.

According to Lotman, a semiosphere is a "semiotic space necessary for the existence and functioning of languages" and contains collections of related signs (Lotman, 1990 p.123). He gives the following metaphor:

... imagine a museum hall where exhibits from different periods are on display, along with inscriptions in known and unknown languages, and instructions for decoding them; there are also the explanations composed by the museum staff, plans for tours and rules for the behaviour of the visitors. Imagine also in

this hall tour-leaders and visitors and imagine all this as a single mechanism (which in a certain sense it is).

This is an image of the semiosphere. (Lotman, 1990 p.126-127)

Semiospheres can never exist alone, and they overlap and communicate with other semiospheres (Lotman, 1990). They are heterogeneous, asymmetric, constantly in motion (dynamic), and tied to communities of practice and institutions (Lotman, 1990). To use another metaphor, Eco writes that a semiosphere is “in analogy with the concept of the biosphere,” where a semiosphere contains sign systems as a biosphere contains ecosystems (Eco 1990, p. xii). This is an important analogy because it draws parallels between anthroposemiotics and biosemiotics. At the center of any semiosphere is developed and structurally organized natural language (Lotman, 1990 p.27, 128). Lotman’s semiosphere is important because it gives a nuanced picture of how meaning is generated and shared or not shared between different communities. Each semiosphere encompasses a code of shared associations within a culture or speech community. The semiosphere helps us understand how culture functions; according to Lotman, what is considered part of a certain culture is any semiosphere that overlaps at a certain “in group” position in semiotic space, and which has semiospheres constantly moving in and out of that space (Lotman, 1990). At the center of a semiosphere is a cultural space with things we understand, institutions, and rules, and at the edge are unfamiliar languages, outliers, and outgroups (Lotman, 1990 p.134). Eco makes sure to note that no culture is encompassed by one semiosphere or code; he writes that “Lotman was and is aware of the fact that no historical period has a sole cultural code...there exist simultaneously various codes” (Eco, 1990 p.x). At the overlap of semiospheric borders is where innovation and creativity occur (Lotman, 1990 p.142; Lotman, 1988 p.69). Semiospheres are bounded by space and time (Lotman, 1990 p.133).

Lotman's work in literature and culture builds on his theory of semiospheres. According to Lotman, literature is social thought, and the language in which this literature is written is the primary modeling system of the world (Eco, 1990; Lotman, 1990). Like Bolinger, he believed the border between the sciences and humanities should be eliminated (Eco, 1990). Texts from different cultural codes or languages—different semiospheres—enter another semiosphere easily from the boundary, but have a difficult time permeating the center of a semiosphere. Lotman traces the permeation of these outside texts into the center of other cultural semiospheres in *Culture and Explosion* in the following process: texts come into the new semiosphere and hold their strangeness and high position, and are often read in a foreign language. The imported and native cultural texts in the new semiosphere interact, and both become metalingually connected. The native culture creates a more developed version of the imported text, and the imported original falls out of favor. This inspires even more texts and knowledge to be generated in the semiosphere and they become the center of it. The process of importing and adopting texts from an external semiosphere, often across language boundaries, leads Lotman to write “to express something in another language is a way of understanding it” (Lotman 2009, 127). The influenced culture then becomes influencer and begins to export many of its own texts into other cultural semiospheres (Lotman 2009, 147).

### **Umberto Eco**

The last major semiotic thinker I will cover in this chapter is Umberto Eco. Eco was an Italian novelist, medievalist, semiotician, and cultural critic who lived from 1932-2016. He is known for his interpretation of Peirce, development of Lotman's cultural semiotics theories, and application of semiotics to literature. Eco's most well-known works include *The Open Work* (1968), *A*

Theory of Semiotics (1976), *The Role of the Reader* (1984), his introduction to Lotman's *Universe of the Mind* (1990), and *Interpretation and Overinterpretation* (1992). While Eco began as a relatively structuralist thinker, he heavily relies on Peircean analysis in *A Theory of Semiotics* and accepts most of Peirce's nonstructural theory.

Eco's theory of interpretation is one of his most influential. In *Interpretation and Overinterpretation*, Eco theorizes that texts only have meaning when they are interpreted by a reader (Eco, 1992). Because Eco had become relatively nonstructural by the time he wrote this piece and follows Peirce's logic of emergent meaning, Eco theorizes that what the author intends is unimportant to the final interpreted meaning of a text, which can only exist once the dynamic object (reader) enters the sign complex. Eco does write about this theory as early as 1976, however, when he writes that signification can only happen with an interpretive response from the receiver (Eco, 1976). He concedes that the reader may bring the author into consideration when the reader takes into account what semiospheres they think the text inhabits, which includes guessing the author's historical and cultural semiospheres. However, the reader is only able to choose from the semiospheres they are familiar with. Thus each interpretation of the text will be vastly varied, and many new interpretations will arise over time as cultural semiospheres change. Eco's theory of interpretation is an example of Peirce's interpretant, in which dynamic meanings are endlessly generated. Eco also warns against overinterpretation, in which the reader believes the text must conceal a secret (also known as hermetic thought) and sees false contiguity (Eco, 1992). Eco writes that legitimate forms of interpretation must use empirical analysis.



One of Eco's other important contributions is his critique of iconicity, or the overuse of Peirce's icon. Eco corrects the common misconception that Peircean icons are types of perceivable signs, popularly expressed as "factual similarity" between the signified and signifier (eg. a picture of an umbrella would be an icon of an umbrella) (Eco, 1976; Jakobson 1980 p.11). The definition of icon as factual similarity, as written by Jakobson, is wrong for two reasons; first, factual similarity does not exist because all facts are culturally constructed. For example, even if a picture represents an umbrella, which umbrella is it representing? Each person's mental representation of an umbrella and its visual features, functions, and associations is culturally their own (Bucholtz & Hall, 2011 p.371-377; Eco 1976). Second, Peircean icons are firsts of seconds, and because they are not yet at the level of thirdness they cannot be perceived (Eco, 1976; Savan, 1980)—any part of Peirce's sign-object triad merely denotes meaning which the interpretant actually is (2.301). A Peircean icon is something that suggests the possibility of the form of the object-sign that will emerge. If something is interpreted to resemble another sign, this could be because of similar perceptual effects, says Eco, but not because one is an icon of the other. This critique will become especially important when I discuss applications of semiotics to design, where "icon" has two competing terminological traditions.

Another of Eco's contributions that is very important in the context of semiotics and its applications to design is his theory of sign production, found in his book "A Theory of Semiotics." Eco's theory of sign production concerns the labor required to produce different types of signs (Eco, 1976 p.151). Eco developed this theory early in his career and it represents a mostly structural, non-Peircean perspective, which means the types of signs Eco are concrete utterances that do not include the utterers and dynamic meaning. Many properties of a sign

determine its labor, and each kind of labor interacts with and is hierarchically organized with the others, much like Jakobson's factors and functions (Eco, 1976 p.153). The types of effort which are always present are choosing the type of signal, physically producing the signal, composing a string of expressions, and making the sign utterance be perceived by the addressee (Eco, 1976). Inventing a new type of sign that is not already embedded in cultural code, creating new associations, using unfamiliar codes, and codeswitching more often leads to more labor (Eco, 1976 p. 152). In another parallel to Jakobson, Eco writes that labor is also demanded to open contact (a communication channel) with the addressee. The latter step is essential, because as Eco will theorize, signification only occurs if it elicits a response (much like Peirce's emergent meaning only exists with the interpretation of the receiver) (Andrews, 1990).

## **Conclusion**

In this section I have covered the fundamental terms and theories needed to understand my applications of semiotics to UID. Peirce and Eco's focus on the addressee as the interpreter and creator of emergent meaning will be important to users' relationship to digital interfaces, and Eco's A Theory of Semiotics and sign production will be important to UID interpretation. Eco's critique of iconicity will be especially important when discussing visual sign systems, and Lotman's semiosphere and Jakobson speech act model will help us understand the cultural codes that underly sign systems in design. Jakobson's speech model will also be important in analyzing interface pages, sequential layout, and the poetic goal of design. Uexküll's model of stimuli and interaction will be informative in the context of models of perception and user-computer

interaction. In the next chapter, I will similarly elaborate the history, fundamental terms and concepts, and important scholars of UID and HCI.

## Chapter 2: Fundamental Concepts of User Interface Design

“Many devices are difficult to use ... So the goal of Cognitive Engineering is to come to understand the issues, to show how to make better choices when they exist, and to show what the tradeoffs are when, as is the usual case, an improvement in one domain leads to deficits in another.” (Norman, 1986 p.31)

In this overview of UID, I will define the field, its goals, and what a well-designed user interface possesses. The goal of my later semiotic applications will be to design better user interfaces.

### **2.a What is UID? Key terms and fundamentals**

#### **Goals of user interface design**

If the goal of UID is to make interfaces more useful to users, what does useful mean in this context? Jakob Nielsen of Nielsen Norman, the most widely recognized and respected interface design agency, writes that the usefulness of a digital interface has two parts: utility (does the interface give a user the information they need?) and usability (is the interface easy and pleasant to use?) (Nielsen, 2012). John Carroll, one of the founders of the field of Human-Computer Interaction, corroborates this when he writes that UID’s goal is to make a “effective, conceptually coherent, and usable interface”—effective is synonymous with utility, although Carroll does add “conceptually coherent” to Nielsen’s differentiation (Carroll, 1991 p. 293).

It's intuitive to see that utility is a goal of interfaces, given that we recognize the importance of information retrieval and visualization in application use, and that interface success is usually deemed by how easily we can find information (Carroll, 2003). Carroll writes that “provid[ing] the richest information, with the least cost to access it (easy information retrieval)” is one of most important success factors of an interface (Carroll, 2003; p.159-160; Sharritt, 2008). Nielson breaks utility into five components:

1. Learnability: How easy is it for users to accomplish basic tasks the first time they encounter the design?
2. Efficiency: Once users have learned the design, how quickly can they perform tasks?
3. Memorability: When users return to the design after a period of not using it, how easily can they reestablish proficiency?
4. Errors: How many errors do users make, how severe are these errors, and how easily can they recover from the errors?
5. Satisfaction: How pleasant is it to use the design?

(Nielson, 2012)

The “pleasantness” (usability) of an interface is equally important as utility. Nielson's utility + usability theory includes usability as a direct consequence of the research of Donald Norman, who is widely known as a founder of user experience design and an expert on UID with Nielson at Nielsen Norman. Norman writes that the goal of a user interface designer is to make the interface as pleasant to use as possible for the first time in his essential text *Cognitive Engineering* (Norman, 1986 p.32). He develops this theory in his book *Emotional Design* (Norman, 2004). In it, Norman cites Kuroso and Kashimura, two researchers in Japan who designed an experiment which concluded that aesthetic pleasantness of use didn't just make the

user have more positive emotions—these positive emotions “broaden the thought processes and facilitate creative thinking,” which makes it “easier for people to find solutions to the problems they encounter” (Kurosu & Kashimura, 1995 in Norman, 2004 p.19). This happens because “the emotional system changes how the cognitive system operates,” a phenomenon rooted in cognitive science (Norman, 2004 p.18).

### **Terminology and history**

Two of our most important terms will be user and interface. Grudin defines users simply as whoever interacts with the interface (Grudin, 1990). Many people interpret this as the typical end user—a customer of an application, a home-computer owner, etc.—but Grudin makes sure to note that “in the beginning...most users were engineers and programmers” (Grudin, 1990 p.261) and users are anyone who needs to interact with an interface at any point in the development process (Carroll, 2003 p.379). This includes designers. Carroll offers some nuance, defining users as “complex adaptive agents who shape their strategies and actions to be more efficient and functional with respect to their information ecology” (Carroll, 2003 p.161). This defines users relative to their actions, where anyone who interacts with an interface with a strategy for retrieving information is a user.

Carroll defines interface as the following: “The interface serves to transform the properties of the [computer’s] representational system to those that match the properties of the person” (Carroll, 1991 p.26). The physical manifestation of this is almost always a visual screen with physical inputs, although early physical manifestations could be composed of only buttons or other physical inputs. Grudin lays out the history of what was considered a physical interface:

Initially, the user interface was located at the hardware itself—most users were engineers working directly with the hardware. The focus then moved to the programming task—higher-level programming languages and environments progressively freed users from the need to be familiar with the hardware. Next, with the widespread appearance of interactive systems and non-programming “end users,” the user interface shifted to the display and keyboard, with early attention to perceptual and motor issues. (Grudin, 1990 p.262)

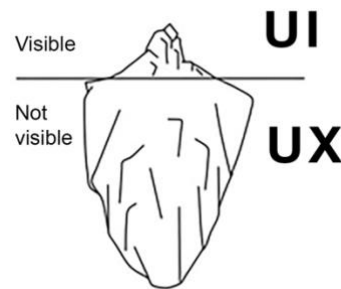
HCI and UID researchers fall into two camps: those who believe the interface is the physical interaction system the user sees, and those who believe the interface is the relationship between the internal representation of the user and the internal representation of the computer. Grudin clearly falls into the first camp, which we can see when he defines the interface as physical parts of the computer as well as when he writes that a computer interface exists regardless of if it has a user<sup>8</sup> (Grudin, 1990 p.261). Andersen represents the second perspective, where “the interface is a sign system manifested in computational processes that people consciously or unconsciously create when using and interpreting the computer system” (Andersen, 1990 p.310). Andersen comes from a clearly semiotic perspective, which is based in the idea of emergent meaning: he writes “interfaces ‘emerge’ when the system is used” (Andersen, 1990 p.130). I will be primarily relying on this perspective. When I refer to interfaces, I am referring to this representation system, which can be perceived in a physical form but is primarily a relation (Andersen, 1990 p.129). Newman and Sproull as cited by Kammersgaard, who I will be discussing later, also agree with this perspective, writing “user interfaces [consist] of the following four components: the user's model, the command language, the information display, and the feedback” (Newman

---

<sup>8</sup> Grudin went as far as to propose calling the user interface the computer interface instead, because the user interface assumes it needs a user while the computer interface exists regardless of user use.

& Sproull, 1986 in Kammersgaard, 1988 p.345). The interface is not just the physical display, but the relation itself.

There are a number of subfields and synonyms under the umbrella of UID. UI/UX is a common synonym for UID that splits it into UI (user interface) and UX (user experience) (Hartson & Pyla, 2012). In this context UI does not stand for general user interface design, but for only the visible graphic design of an interface (eg. the color scheme and visual appeal of buttons). UX refers to everything invisible, or the ease of use of the information architecture of the interface. UX asks “how easy is it to navigate a menu bar?” or “how easy is it to complete the process of checking out, which requires interaction with multiple pages?” Many people find it useful to view UI as the part of iceberg we can see, and UX as the equally important but invisible underwater portion (see my diagram below). Interaction design is a subfield of UX, and focuses on any action the user takes which results in a reaction in the interface (eg. how does the user use a mouse to drag an object, or what happens when the user clicks the “new tab” button?) (Hartson & Pyla, 2012).



Iceberg diagram of the difference between UI and UX

Another popular term in the sphere of UID is user centered design. The term user centered design originated in the 1970s and 80s and was explored deeply in Norman’s User Centered



System Design (Norman, 1986; Hartson & Pyla, 2012). In an article in Norman's book, Hooper explains that user centered design changed the focus of the design process from the functionality of the computer with predicted user activity via the designer and programmer, and changed it to center the user's needs and values and involving them in the design process before moving to computing functionality (Hooper, 1986 p.16 in Norman, 1986). User centered design profoundly changed interface design and as I will analyze later changed how designers work in a way which more closely aligns with semiotic perspectives.

UID is rooted in theories of human-computer interaction (HCI), which studies "how people make use of devices and systems that incorporate or embed computation, and how such devices and systems can be more useful and more usable"<sup>9</sup> (Jorgensen, 2007; Carroll, 2003 p.1). At its simplest level, HCI explores "the roles played by users and systems" (de Souza, 1993 p.753). Carroll continues, "HCI professionals analyze and design user interfaces and new user-interface technologies. In the past, they helped to develop and investigate the concepts of the now-pervasive graphical user interface paradigm of windows, menus, icons, and mouse pointing" (Carroll, 2003 p.1). HCI preceded the term user interface design but studied the same subject matter before the term became popular—this is because, as Grudin writes, "The term 'user interface' was not needed in the beginning, when most users were engineers and programmers" (Grudin, 1990 p. 261; Carroll 2003). The term user interface and subsequently user interface design became popular in the 1980s with the rise of consumer computers and "non-programming 'end users'" (Grudin, 1990 p. 262).

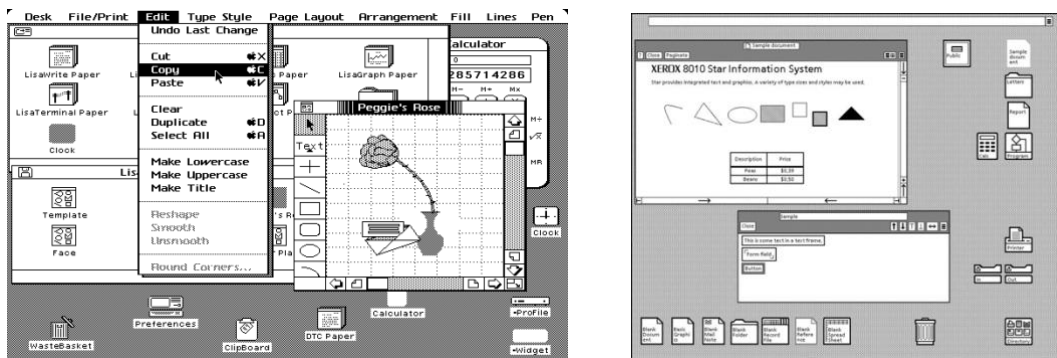
---

<sup>9</sup> HCI is sometimes referenced as computer-human interaction (CHI) (Myers, 1998).

In contemporary terminology, HCI and user interface design still cover the same subject matter, but HCI tends to gravitate towards modelling and academic research while user interface design often implies an industry-facing attitude with a focus concrete interface designs. Different models of HCI which user interface design is rooted in will be especially important in our semiotic analysis and they tend to be highly compatible models of perception.

The earliest developments in modern HCI took place in the 1960s, when early research for the Xerox star and Apple Lisa—the two first widely used digital interfaces—began at MIT, Imperial College, Stanford, and Xerox PARC (see below) (Myers, 1998; Norman 1986). In these early interfaces, users could only type in strings of characters which would be sequentially processed (Nadin, 1988 p.279). In the 1970s, digital word processing (WordPerfect), spreadsheets (VISICALC), email (network mail), and drawing interfaces (Sketchpad) were also developed. These interfaces utilized the design theory of what you see is what you get (WYSIWYG), in which the interface the user manipulated resembled the final result (eg., the WordPerfect document resembled a printed piece of paper) because the user is unable to interact with anything that is not displayed but would be part of the final product (Myers, 1998). The Apple Lisa, Xerox Star, and Apple Macintosh became widely distributed in the 1980s (Myers, 1998). These devices' interfaces utilized the mouse and cursor, direct manipulation of objects on the interface using the cursor, and multiple windows the user could view simultaneously. As we can see in the images of the Star and Lisa interfaces below, they were both designed to replicate the mental schema of the office desk, which has continued to be one of the most common and effective “mental metaphors” of UID (Myers, 1998; Norman, 1986 p.48; Kammersgaard, 1988 p.355). Later in the 1980s and incorporated in the 1990s, components (reusable pieces of code

for buttons, menus, and other interface parts) became instrumental in the standardization of digital interfaces. User Interface Agents that provided users with proactive help like Microsoft's Bob or Paper Clip were also developed during this period (Jorgensen, 2007 p.34). Designers also began to focus on search methods as users became more familiar with and approached interfaces with goals, instead of the previously dominant browsing function in which the user discovers their goal.



User interfaces of the Apple Lisa (left) and Xerox Star (right)

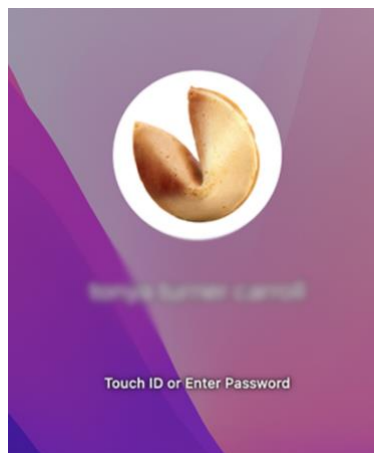
These advances in HCI preceded the widespread use of the internet. When the internet became widely accessible in the 1990s, users were able to interact with interfaces (websites) that changed and grew more frequently than the applications they were used to and were more diverse (The Interaction Design Foundation, 2023; Grudin, 1990). Users began to use the computer not to accomplish their own tasks but as a tool to communicate with others (The Interaction Design Foundation, 2023; Grudin, 1990 p.262).

## Principles of good user interface design

The current principles of good UID vary across designers and intended functions, but there are a few widely recognized guidelines. Blair-Early and Zender give a good overview of these guidelines in their article “User Interface Design Principles for Interaction Design,” which I will list in their order shortly but are also supported in Steve Krug’s “Don't make me think! a common sense approach to Web usability,” Usability.gov’s “User Interface Design Basics,” and Apple’s “Human Interface Guidelines” (Blair-Early & Zender, 2008; Krug, 2006; Usability.gov, 2023; Apple Developer, 2023). Their general principles of good UID are as follows:

1. Well-designed interfaces have a clear starting point for a user’s action (Krug, 1990)

Good example: when a user opens their Apple computer, they are given clear way to start using it: either Touch ID or Enter Password.



Bad example<sup>10</sup>: when a user lands on the homepage of Zara.com, there is no obvious starting place or pathway to begin shopping or browsing.

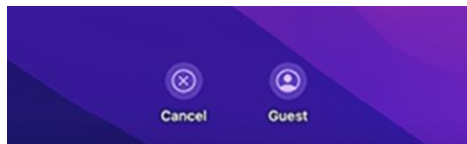
---

<sup>10</sup> For some intentionally bad examples see: UserInyerface <https://userinyerface.com/> and SebastianLyserena <http://sebastianlyserena.dk/>

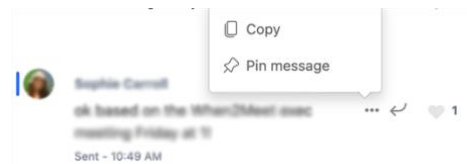


2. Well-designed interfaces have a clear way to exit/stop/reverse a user's action (Nielson, 1994). With user feedback, this can lead to continual updates of the website which proactively stop common errors (Nielson, 1994; Usability.gov 2023)

Good example: On the same Apple screen that appears when a user opens their computer, there is an obvious "Cancel" button in case the action wasn't intentional

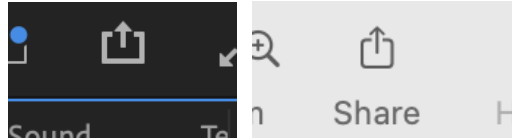


Bad example: in theory, users are able to delete GroupMe messages they sent. In the computer application, that option is not easily findable.



3. Well-designed interfaces use conventions widely recognized across other interfaces (Krug, 1990; Nielson, 1994; Norman 1986)

Good example: The arrow out of a box icon is easily recognizable as an export/share action and is used across many application interfaces (shown: Adobe Premiere (left), Apple Preview (right))<sup>11</sup>



Bad example: The web application version of Word automatically sets an 8pt spacing after every new paragraph. The desktop version of Word, the web version of Google Docs, and the majority of other word processors set this to 0 points (no space between lines), which confuses the web Word user.

Starting to write

Default settings

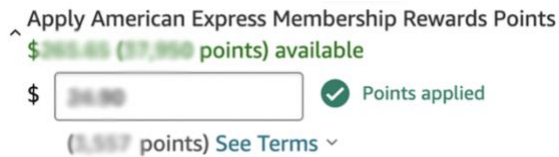
More writing in default settings More writing in default setti  
writing in default settings

4. Well-designed interfaces provide feedback (how far is one in the process, is the process proceeding without error?) (Nielson, 1994; Usability.gov 2023)

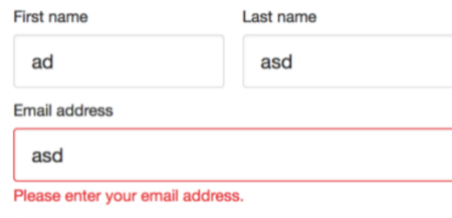
---

<sup>11</sup> At first, I had used a floppy disk “save” icon as the example. While this is widely used, upon further thought the floppy disk was not a good choice of icon because it represents a form of data storage that would soon become obsolete and fall out of most users’ codes. The representation did not stand the test of time as the designer might have expected it to.

Good example: When a user hits enter on the number of points they want to use, a confirmation “Points applied” message appears



Bad example: on some websites when a user is typing their email in a textbox, an “incorrect format: must include @” error will appear before the user has finished typing @, even though they are filling in this field as they are supposed to. A good error message would occur when the user hits enter or moves to the next field.

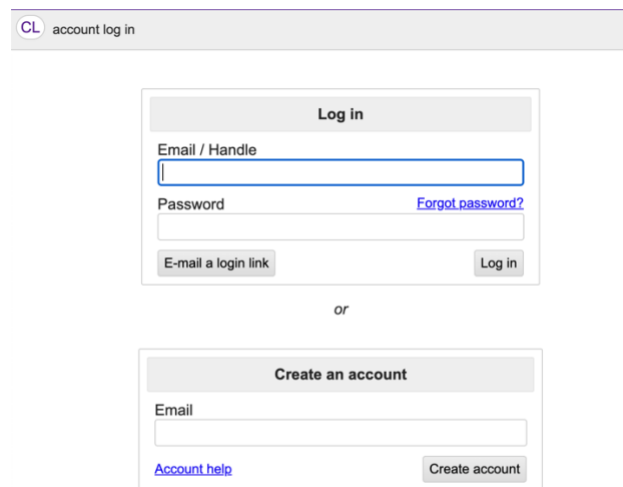


5. Well-designed interfaces locate the user in the interface (what path did they take to get there, what landmarks allow them to find the homepage again?) (Krug, 1990)

Good example: In this database, the user is at a page called “Product Details.” They can see the path they took to get there and they are able to navigate to previous levels using what is called the breadcrumb menu (Home->Products->Product Details), which details the hierarchy of pages. The user is also able to navigate Home using the top level menu (Home Customers Products).



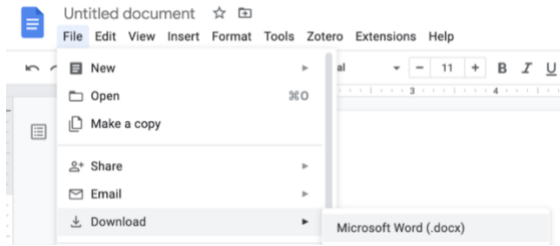
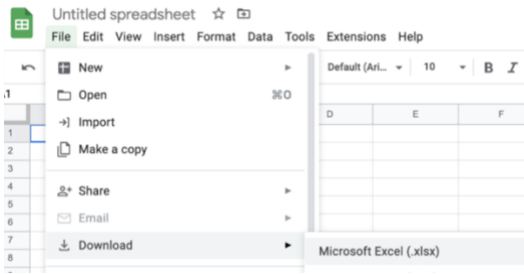
Bad example: On the Craigslist login page, how do you get back to the homepage of the site? Where is the account page in relation to the rest of the pages? There is no way to tell.



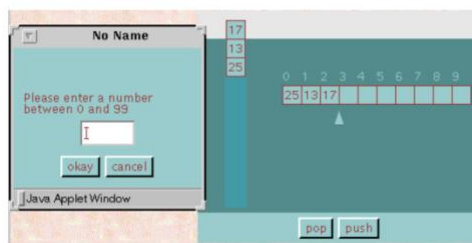
- Well-designed interfaces have consistent logic throughout the interface (the process of taking similar actions should be the same regardless of the time/place of the action in the interface) (Nielsen, 1994; Usability.gov 2023)

Good example 1: Downloading a Google Sheets document as an Excel file and downloading a Google Docs document as a Word file are similar actions—both export a Google format document to a Microsoft format document. The process is consequently logically similar (File->Download->Excel/Word) across Google Drive applications

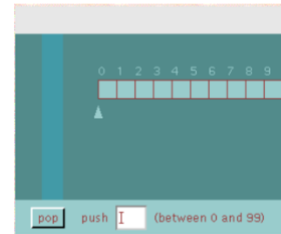




Bad example: Goguen uses an example app for the principle of logical consistency (Goguen & Malcolm, 199, p.7). He refers to the common computer science data structure called a stack, where users can push a number (that they input) onto the top of the stack and pop whatever number is on top (that they don't input). In the version 1 of the application, the user only has to select pop for the top value (17) to be popped. However when they select push—which is the counterpart of pop and should function in a similar way—the user must interact with a new popup window with additional buttons that makes it much more complex and effortful than popping (see below). Goguen fixes this problem in version 2, where the number of actions a user has to take to push is greatly decreased. The user sees pop and push as similar actions, where besides inputting the number both pop and push are both completed in the same number of clicks and windows.



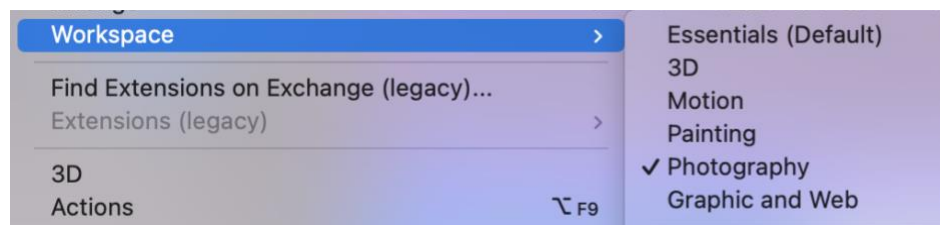
Version 1 (bad)



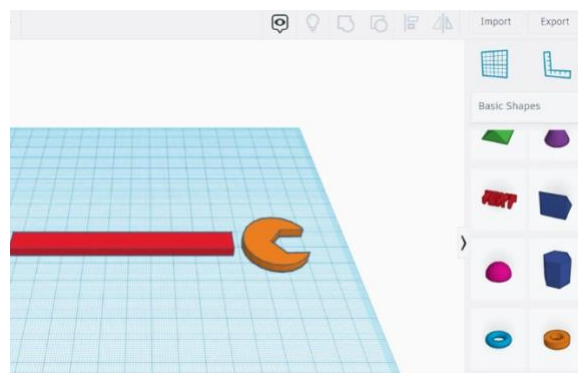
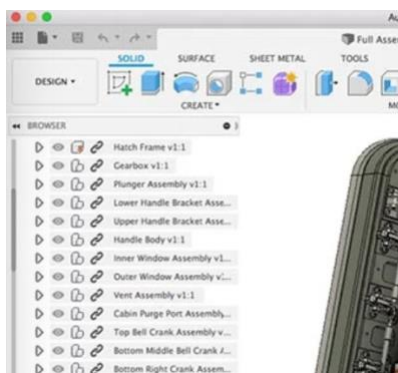
Version 2 (good)

7. Well-designed interfaces are adaptive, meaning the user should be able to customize the interface based on their needs or choose one of multiple paths to their goal that is most intuitive to them (Nielson, 1994)

Good example: in Photoshop, the user can change the layout and tools provided in the interface based on their type of work. Many other interfaces have advanced/basic settings.



Bad example: Autodesk Fusion, an essential engineering application that is very challenging to learn, has no “beginner” version (picture below on left). Instead, TinkerCAD, which was developed as a response and is Fusion for beginners, is hosted online and not within the Fusion application (picture below on right).



8. Well-designed interfaces are visually designed to support and not distract from the content. This includes announcing the website's purpose and content provided immediately (Krug 1990; Nielson, 1994)

Good example: Refugee.info is an example of a good interface. Distributing information content is its central goal and it announces its purpose and identity clearly on the landing page—the text is uncluttered.



Bad example: To use Zara as an example again, the content they want to prioritize is their clothes and how users can buy them. But the landing page gives no indication where we might find that content.



Krug and Nielson add one important point to Blair-Early and Zender's eight points: well-designed interfaces have easily findable help resources and FAQs (Nielson, 1994; Krug, 1990).

Nielson and Carroll give us two ways to analyze the effectiveness of an interface, which provide more complexity than simply having or not having the traits above. According to Nielson, the following are five ways we can judge the usefulness of interfaces (in comparison to other interfaces) (Nielson, 2012):

1. Learnability: How easy is it for users to accomplish basic tasks the first time they encounter the design?
2. Efficiency: Once users have learned the design, how quickly can they perform tasks?
3. Memorability: When users return to the design after a period of not using it, how easily can they reestablish proficiency?
4. Errors: How many errors do users make, how severe are these errors, and how easily can they recover from the errors?
5. Satisfaction: How pleasant is it to use the design?

Kammersgaard makes sure to warn against overemphasizing learnability, as user interface designers sometimes do, because “this leads to a tendency towards development of computer applications which may be easy to learn, but which are not sufficiently powerful to accomplish the task they are intended to perform” (Kammersgaard, 1988 p. 355).

Carroll gives us another framework, GOMS, to evaluate the utility and usability of interfaces (Carroll, 2003). GOMS stands for Goals, Operators, Methods, and Selection rules, as described below (Carroll, 2003 p. 58):

1. Goals: The users' goals: what does the user want to accomplish with the software, and by when?
2. Operators: The actions that the software allows the user to take. Generally it is a command such as a button press, menu selection, or a direct-manipulation action.
3. Methods: the sequence of sub-goals and operators that can be used to accomplish a goal: such as the sequence of moves necessary to cut and paste text to move it from one location to another.
4. Selection Rules: Giving the user the personal "choice" (selection) to accomplish the goal in those times when there is more than one method available to accomplish the goal or certain task

GOMS can be used to quantitatively predict performance and learning time when completing a task, or qualitatively describe user thinking patterns in order to find inefficient systems (Carroll, 2003; Sharritt, 2008).

In conclusion, UID develops and studies principles of good design and also theoretically models user and computer interaction. Today digital distribution of information is dominant and many people's livelihoods revolve around digital interfaces to accomplish tasks, making these digital user interfaces more pleasant, easy, and effective to use is paramount. Using a semiotic perspective to deepen our understanding of how users perceive, find meaning in, and interact with these interfaces will help us achieve these goals and is research we should continue to pursue.

## **2.b HCI and User Interface Models; Thinkers and Theories**

Modelling the relationships between users, computers, and interfaces will be essential to my later applications of semiotics. In this section I will cover the HCI models of Norman and Kammersgaard. The goal of this essay is to find practical and concrete takeaways for designers, and discussing HCI and semiotics theory and models is helpful because it gives designers a shared vocabulary, enables them to understand the high level conscious and subconscious strategy of the user, and gives them a theoretical perspective to base their problem solving thought processes in (Carroll, 2003; Sharritt, 2008; Nadin, 1988; de Souza, 1993).

### **Norman's Cognitive Engineering model**

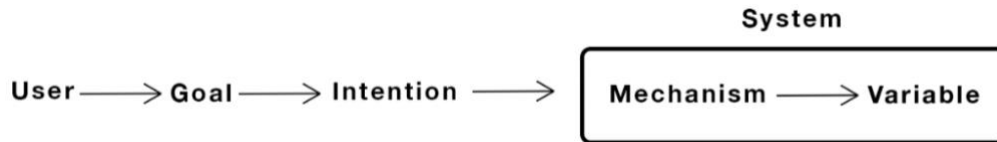
Norman's 1986 article Cognitive Engineering in User Centered System Design is one of the foundational models of computer systems, and many of Norman's ideas were precursors to the contemporary field of interface design (Norman, 1986 p.60). Many of his theories are rooted in a cognitive science perspective (de Souza, 1993 p.754). Norman's goals are as follows (Norman, 1986; p.32):

1. To understand the fundamental principles behind human action and performance that are relevant for the development of engineering principles of design.
2. To devise systems that are pleasant to use-the goal is neither efficiency nor ease nor power, although these are all to be desired, but rather systems that are pleasant

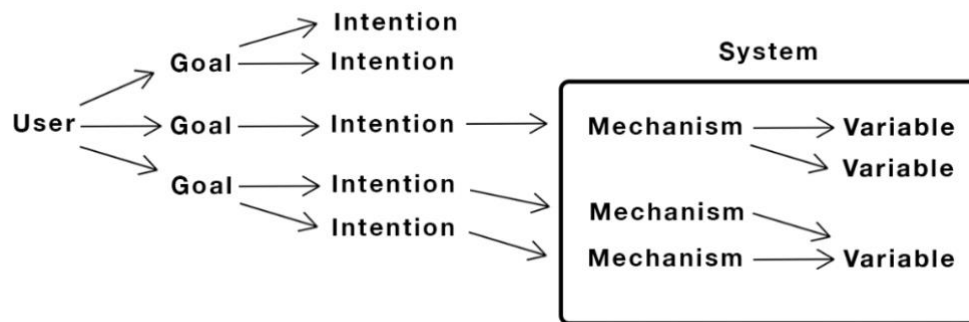
Regarding his first point, the principles of design that are relevant are the external interaction factors that effect users, not the hardware or software engineering of the computer. The second point clearly supports Norman's early thinking that the goal of UID is pleasantness of use (utility

will become more important later, as Nielson recognized in my previous section). Norman says the four ways we can identify good computer systems design is mapping (identifying and using the correct inputs), ease of control (how easy it is to use these inputs with the goal of desired outcomes), evaluation (if the outcome is correct), and feedback (how long did it take to determine the result) (Norman, 1986 p.33, 35).

Norman models these four criteria using goals, intentions, mechanisms, variables, and systems (see my diagram below). A user approaches the computer with a goal, and once they do mapping, they set an intention; if their goal is to type a paragraph, they might set an intention to click a new line in Word (Norman, 1986 p.34). To accomplish a goal one may set multiple intentions. The computer is referred to as the system: anything that changes the computer interface happens within the system (see the mechanism and variable inside the system in my diagram below). The system contains mechanisms to change it—physical or digital inputs—and each mechanism changes a variable of interest. The system is a function of the variables it contains. Multiple mechanisms can work together to change one variable of interest (like an EtchASketch, which uses x and y-axis knobs to draw one diagonal line) or one mechanism can change multiple variables (like a single shower handle which changes water pressure and temperature) (Norman, 1986 p.36). When the user sets an intention, it is to manipulate one of these mechanisms with the hope of changing their desired variable of interest in a way that completes their goal. To accomplish a complex task like writing an essay, a user may approach a system with many goals (open Word, type a paragraph, highlight or bold words, save the document, etc.).



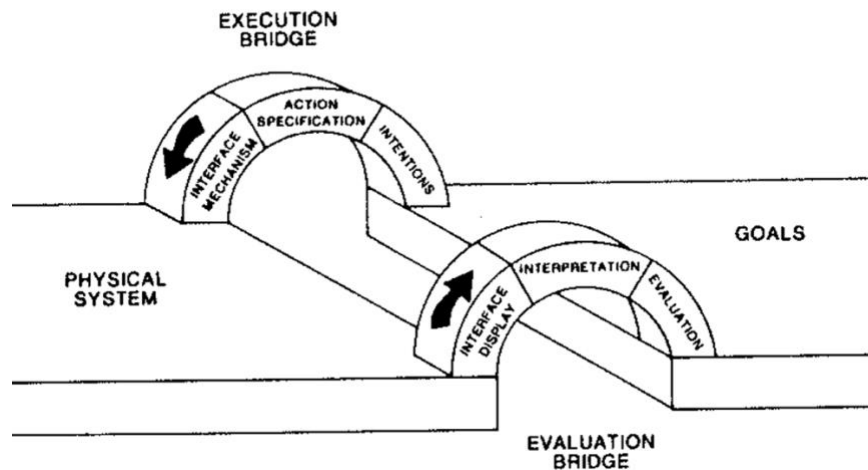
A simple version of Norman's model



A more realistic version of Norman's model

Where problems arise when a user interacts with a computer system is, according to Norman's theory of action, the fact that goals and intentions are psychological variables expressed relative to the user, while mechanisms and variables and physical variables are expressed in relation to the computer (Norman, 1986 p.38). The goals of the user and the system must be obviously, directly, and closely related in a well-designed computer system, but a perfect version of this is impossible. Norman identifies this mismatch as two gulfs: the gulf of evaluation and the gulf of execution (diagram shown below (Norman, 1986 p.40)) (Norman, 1986 p.38, 52). The designer can bridge the gulf of evaluation by designing inputs and outputs that are easier for a user to evaluate, and the user can bridge the gulf of execution by learning to use the computer closer to its intended use (Norman, 1986 p.39).



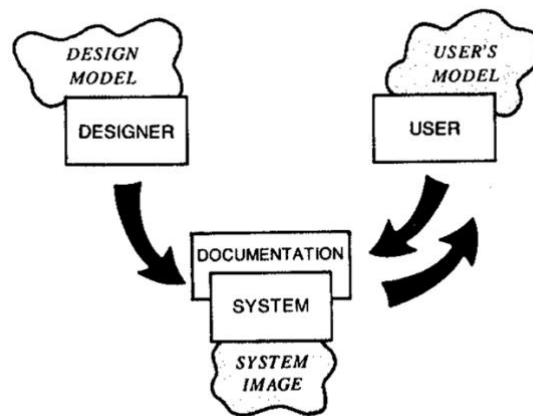


Norman's gulfs of evaluation and execution

The model above also describes the seven steps of user activity: establish goal, form intention, specify action sequence, execute action, perceive resulting system state, interpret state, and evaluate state compared to original goal (Norman, 1986 p.40). While this describes one neat model, in real life stages are out of order, skipped, or repeated. The evaluation gap can be hard for designers to bridge because users are diverse and have diverse goals; there is no one model user (Norman, 1986 p.48). The evaluation gap is not as large when designing a less complex interface for a smaller and more specific audience with more defined uses. When this isn't possible, deeply understanding your user base is essential (Norman, 1986 p.58).

The more the user interacts with the computer system, the more they form a mental "user's model" of the computer (see diagram below) (Norman, 1986 p.46). The designer's job is to make sure they use consistent, logical rules, because the designer and engineer are not able to transfer their intended, imperceptible "designer's model" to the user (even if they try documentation,

users don't read it) (Norman, 1986 p.48). The designer's model is everything the designer knows about how the computer functions, but the user actually builds their model using the system image—as you can see in the diagram below, the system image mediates between the two. The system image contains any function of the computer in relation to itself, including relationships between mechanisms and variables, as well as any part of the computer they interface with (including instruction manuals, etc.). The user must constantly identify and refine the rules that the computer system uses, and they may never interface with many aspects of the designer's model.



User's model vs. design model

Norman stresses that design decisions always have tradeoffs for another feature of functionality, so looking at the design system as a whole and ordering changes by priority is essential (Norman, 1986 p.56-57). Nadin models examples of these tradeoffs below using Norman's 1983 "Design principles for human-computer interfaces" (Norman, 1983 in Nadin, 1988 p.300).

Table 2. *Language chart.*

Natural language	Formal language	Communications characteristics of formal languages		
		write	select	read
general	specific	understand	read system	model system
unlimited	limited vocabulary	system	intuition	intuition
vocabulary	predefined grammar	concrete	have purpose	indefinite
indefinite grammar	logical structure	knowledge	semantics	purpose
intuitive structure	difficult to acquire	have purpose	error	semantics
easily acquired	competence	syntax error		error
competence	easy to attain high			
difficult to obtain	performance/easy			
performance	to learn			

Norman's tradeoffs as diagrammed by Nadin

Norman's theory of human computer interaction was deeply impactful in the field and is frequently cited as the basis of other models, especially his gulfs of evaluation and execution. He placed the user inside the model, and emphasized the importance of the user's interpretation to meaning, as opposed to meaning inherent in the computer interface. Norman's use of goals, intentions, mechanisms, and variables made his model very easy to understand and intuitive, and his exploration of the user's and designer's model drew clear connections to semiotics later.

### **Kammersgaard, Four Perspectives on HCI**

John Kammersgaard was an early scholar of HCI at Aarhus University in Denmark, which played an integral role in developing the field of HCI and continues to harbor many of the top HCI scholars of the world (Aarhus University, 2023). His particularly influential article "Four different perspectives on human-computer interaction" synthesized many popular models of HCI

into four main schools of thought in the HCI community, and provides an excellent overview (Kammersgaard, 1988).

These four perspectives are “the systems perspective, the dialogue partner perspective, the tool perspective, and the media perspective” (Kammersgaard, 1988 p.343). The dialogue partner and tool perspectives focus on individual users, while the systems and media perspectives focus on users in a collective context (Kammersgaard, 1988 p.344). You can see the four perspectives split by this paradigm as well as by content and expression in the diagram below

(Kammersgaard, 1988 p.343). The “contents level” of the media and tool perspectives means that the model is mostly concerned with the design of the content the user interacts with and uses in a specific work flow. The “expression level” of the systems and dialogue partner perspective means that the design model is less concerned with how to use the computer, and more with the system as a whole, where the users and computers are comparable components in a greater system. Content and expression is a common semiotic separation stemming from the binary sign complex, and this language will continue to be used by the scholars I analyze who apply semiotics to HCI.

Source of design principle	Use context	Collective	Individual
Expression level		Systems perspective	Dialogue partner perspective
Contents level		Media perspective	Tool perspective

Four HCI perspectives and their attributes

Three key terms Kammersgaard uses are media structure, or the visual layout of the interface (similar to UI), domain representation, or the backend and how information is stored and structured (similar to information architecture), and operative cognition, or the thought processes and mental representations of a user (Kammersgaard, 1988 p.347). He demonstrates the differences between his four HCI models using these factors.

The systems perspective is the first model Kammersgaard describes. In the systems perspective, all users and the computer are equal components in a system that exists to complete a task (Kammersgaard, 1988 p.348). The goal of the designer is to make an interface which facilitates the user and computer functioning in the most similar and homogenous way (de Souza, 1993 p.753), where “all interaction is seen as transmission of data between a human and an automatic component” (Kammersgaard, 1988 p.348). This form of interaction changes the domain representation, but not the media structure or cognitive operations. The systems perspective is the most common from a hardware or software development point of view, but creates issues with interface functionality when other perspectives are not considered, especially in regards to aesthetics, the psychological processes of users that cannot match the computer’s, and any interface use that is not data transmission (Kammersgaard, 1988 p.350). Kammersgaard does not prefer the systems perspective.

The second perspective is the dialogue partner perspective. In the dialogue partner perspective, the designer tries to design an interface that is as “humanlike” as possible, and stresses human behavior and natural language use (Kammersgaard, 1988 p.351). This perspective is most popular in the field of AI, in which the goal is to replicate human thinking and behavior patterns.

This model changes domain representation in the computer and operative cognition in the users, while media structure stays relatively stable. A disadvantage of this approach is that right now, computers cannot communicate perfectly like humans too, so this is not a realistic approach and does not account for communication breakdowns. Kammersgaard finds “the dialogue partner perspective problematic” for these reasons and finds it less relevant than the other three perspectives (Kammersgaard, 1988 p.358).

Kammersgaard’s third perspective is the tool perspective. In the tool perspective, the computer is seen as providing the user with a number of abilities that the user can create tools out of (Kammersgaard, 1988 p.352). This model is primarily applied to expert users. Operative cognition, domain structure, and media structure are all changed. Computer interfaces designed under this framework often prioritize functionality over learnability and ease of use (Kammersgaard, 1988 p.355). While these computer systems can be quite powerful and effective, not all users are capable of approaching them and onboarding is quite difficult and energy intensive.

The last perspective is the media perspective. In this perspective, the computer is a “medium through which humans communicate with each other; i.e. something which is comparable to newspapers, books, films, television, telephones, video, etc.” (Kammersgaard, 1988 p.356). Kammersgaard, while not a semiotician, uses the idea of the Saussurian sign to explain the computer’s role; the sender sends a sign which is composed of its expression (signifier) and its meaning (signified), and in conveying this meaning to the receiver it passes through the computer, which “can manipulate the level of expression only” (Kammersgaard, 1988 p.356).

Meaning is the main goal of the interface, and the sender can be the designer, the programmer, another user, or a group of users. Using our Peircean semiotic background, we know that the computer as a mediator must change the meaning of the sign because changing expression changes the sign complex (expression isn't the only thing changed, as Kammersgaard says). However, Kammersgaard's analysis is still useful because of its perspective of the computer as mediator. The media structure is changed because it is the signifier/expression level of Kammersgaard's sign/message that is being sent from sender to receiver, and while domain representation doesn't need to change, it's often important to the sender as they formulate their message. A disadvantage of this perspective is that it focuses on natural language, but when interfaces need concern non-language activities (eg. 3D modeling), it is less suited. The media perspective will be one of the most essential in later semiotic analysis.

Kammersgaard's models give us useful insight into different approaches to HCI, especially the role of the user and meaning. It is a good summary of the many HCI models that exist and gives us a broad perspective we, and later theorists, can use to apply semiotics.

In conclusion, Norman and Kammersgaard's HCI models give us a complex picture of the different ways designers and users view their interaction with computer systems. Norman's cognitive engineering, being closer to cognitive science, gives us concrete steps to bring the user closer to completing their goals using the computer system, and Kammersgaard's four perspectives observationally analyze the relationships between users, designers, and computers. These theories are concerned with representation, expression, perception, and meaning in

context. This is the subject matter of semiotics, which once applied will greatly deepen our understanding of these processes.



## Chapter 3: Integrating Semiotic Models with User Interface Design

“Design principles are semiotic by nature. To design means to structure systems of signs in such a way as to make possible the achievement of human goals.” (Nadin, 1988 p.269)

In this section I will connect the models of human computer interaction and interface design I have covered to semiotics. The relevant semiotic concepts will be Peirce’s nonstructuralism and sign complex model, Saussure’s sign-signifier model, Eco’s theory of sign production, critique of iconicity, and theory of interpretation, Sebeok and Uexküll’s theories of biosemiotics and stimuli, Jakobson’s speech act model, Bolinger’s rejection of the sign as arbitrary, and Lotman’s semiosphere.

A deep understanding of these main semiotic concepts in addition to the HCI models of Norman and Kammersgaard will help us analyze the theories of three current scholars who have attempted to find links between the two areas of study; Andersen, Nadin, and de Souza. I chose these three authors because their semiotic analysis is the most developed and accurate using the foundational semiotic theories I have covered, while also being respected in the HCI field. They are less established in the field of semiotics than the fundamental semioticians I have covered, so even though they will make very valuable contributions, I will look at their theories with a critical eye as I recapitulate them. After I do this I will draw any parallels I see left and synthesize major takeaways for user interface designers.

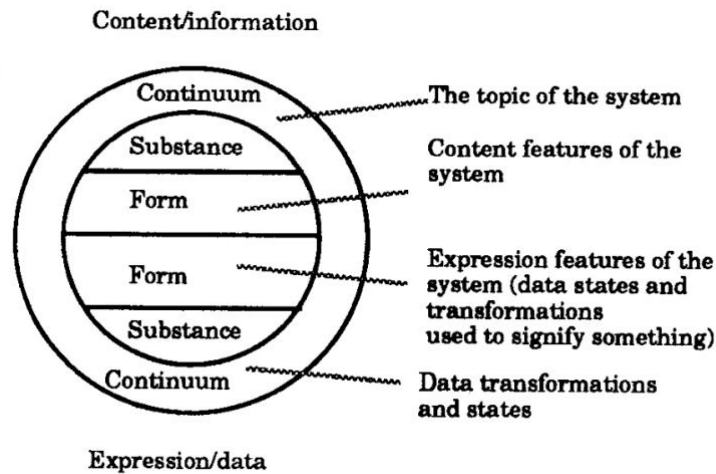
### **3.a Critique of existing applications: Andersen, Nadin, and De Souza**

#### **Peter Bøgh Andersen**

Andersen lived from 1945-2010 and was an HCI researcher at the University of Aarhus in Denmark, which houses many important HCI researchers (including Kammersgaard). Andersen came from a background in literature and media studies which he applied to computer systems primarily through his theory of computer semiotics (Ole Finnemann & Olesen, 2010). Andersen defines computer semiotics as a subdiscipline analyzing “computer systems...as signs that users interpret to mean something” (Andersen, 1992 p.2). His goal was to “sketch a possible discipline of computer semiotics,” however computer semiotics as a subfield title has not become widespread in semiotics (Andersen, 1992 p.2). However, Andersen’s application of semiotics to computer systems was still groundbreaking and he is widely cited by other researchers of HCI and semiotics including De Souza, Nadin, and Kammersgaard. Andersen and Nadin cited each other frequently. Andersen’s work in computer semiotics originated in his dissertation thesis titled “A theory of computer semiotics: semiotic approaches to construction and assessment of computer systems” (1990), and is expanded in “Computer Semiotics” (1992).

Andersen believes that the semiotic “European structuralist tradition is broad enough” as a framework to be the basis of his sign theory, consisting of computer and user signs, “provided its bias towards structural descriptions is balanced by a Peircean understanding of sign processes” (Andersen, 1992 p.27). Andersen is a structuralist who worked with a binary sign, and while he demonstrates tendencies at the edge of structuralism like belief in emergent meaning, his evocations of Peirce still fail to break out of a structural binary framework and I would not say

he achieves a Peircean understanding. Andersen uses the following sign model in his theory of computer semiotics (Andersen, 1992 p.13):



Andersen's sign model in Computer Semiotics

Two planes exist within the sign—expression and contents, each split into form and substance. The bottom half of the circle is the expression plane, which Andersen compares to Saussure's signifier. The top set is defined as the contents plane, comparable to Saussure's signified. Form and substance operate like a simpler version of signifier and signified, where the signifier could take verbal form with phoneme substance, and the signified could take the form a physical apple with the substance of the specific apple's taste and relationship to the user. The continuum represents the sign system in which the sign occurs with many other signs, comparable to Lotman's semiosphere.

Andersen expands his binary sign model to fit the context of a computer system. Computer signs have the same structure but have three distinct features: first, the handling feature, which contains mechanical output and send electronic signals to the processor (Andersen, 1992 p.14).

This is reminiscent of Uexküll’s effector in biosemiotics. Second, a computer sign has a permanent feature, which means a constant form (usually a digital representation). Third, Andersen’s computer sign has a transient feature, which changes with interaction and adds to the permanent feature to indicate state. For example, if a button is a computer sign, the permanent feature is the digital representation of the shape of the button, the handling feature could be how you click it, and the transient feature could be a drop shadow added when the button enters a pressed state. Andersen diagrams examples of each in the following figure, which describes examples of each sign with or without a certain feature<sup>12</sup> (Andersen, 1992 p.15).

		+action		-action
		+handling	-handling	
+permanent	+transient	Interactor	Actor	Object
	-transient	Button	Controller	Layout
-permanent		Ghost sign		

Andersen’s sign features in Computer Semiotics

Andersen believes that an interface is not a physical object or digital screen, but a product of the relationship between users and computer systems (Anderson, 1990 p.129). More specifically, Andersen writes that “The interface is a sign system manifested in computational processes that people consciously or unconsciously create when using and interpreting the computer system” (Andersen, 1990 p.310). Most HCI authors with an understanding of semiotics agree with this approach, and this is the line of thought I have adopted in this essay. To Andersen, as stated

<sup>12</sup> Detailed descriptions of each sign type can be found on page 15 of “Computer Semiotics” (Andersen, 1992 p.15)

above, the interface is the “the basic function of a system” and it is the most important part; Andersen’s applications of semiotics to computers focus mostly on the interface (Andersen, 1990 p.174).

Why is the interface the most important part of the computer system when discussing HCI and semiotics? This is because, as Andersen states, “interfaces ‘emerge’ when the system is used” (Andersen, 1990 p.130) and “it does not make sense to say that signs occur except when they are interpreted by some interpreter” (Andersen, 1992 p.17). Many different interfaces emerge when interacted with by different user groups. Because of these many different users, Andersen “claims that standardization should be restricted as much as possible and only be allowed if good arguments are given explicitly, and that users must have the possibility to modify the means of expression when new needs arise” (Kammersgaard, 1988 p.357; Andersen, 1985). This key takeaway for designers originates in the nonstructural semiotic theory of emergent meaning, introduced by Peirce and evident in Eco’s “Interpretation and Overinterpretation.”

While Andersen believes in emergent meaning, he gets trapped in the struggle I described in my introduction to semiotics; structuralists struggle with how to reconcile changes in sign systems and codes and differences across users. Poststructuralism emerged in response. Andersen himself reiterates this debate when he writes that “the European structuralist [paradigm lacks] one important concept, namely that of the interpreting person and his reactions to the sign” (Andersen, 1992 p.17). He draws in Peirce to reconcile this:

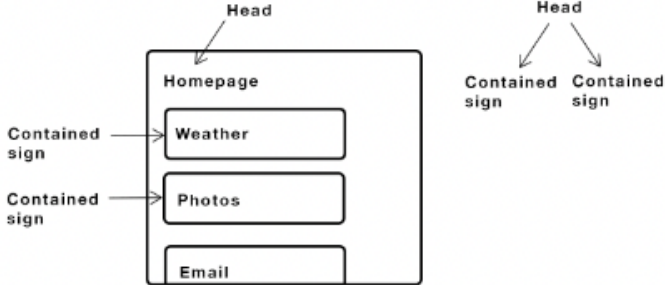
Peirce's variant of the sign concept can help us clarify this point: in his framework it does not make sense to say that signs occur except when they are interpreted by some interpreter. Furthermore, in this process the

interpreter cannot help reacting to the sign, producing a new sign called an interpretant. Therefore Peirce's sign contains three parts: the representamen (analogous to the expression plane), the object (analogous to the content plane), and the interpretant (which has no analogue in the structuralist sign concept). (Andersen, 1992 p.17)

It does not appear that Andersen's interpretation of Peirce solves these structural issues, however. Andersen attempts to connect Peirce's triadic sign components to Saussure's binary components, but the binary expression plane which he compares to the representamen is still perceptible while the representamen is not. Additionally, the content of the sign is the final meaning and Andersen compares this to the Peircean object, but the interpretant, not the object, is the final meaning in Peircean semiotics. Thus, I think it would be more accurate to change Andersen's comparison: object is more similar to expression, and interpretant is more similar to content. However, ultimately comparing binary and triadic systems is not fruitful because each of these components operates very differently within the larger sign system. Andersen tries to fit Peircean nonstructural theory inside the framework of the structural binary sign, which while evoking Peirce, does not really contribute to using Peircean semiotics as analysis.

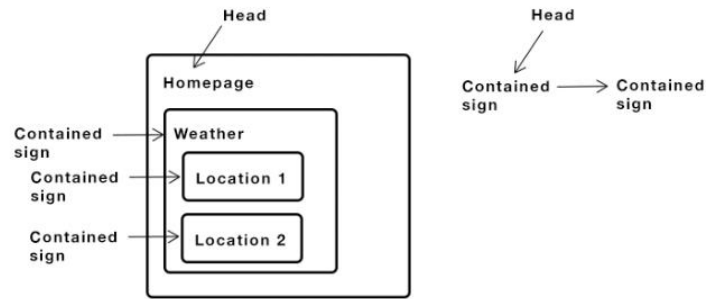
One of Andersen's most thought-provoking insights is his analysis of "composite signs" (Andersen, 1992 p.21). He defines a composite sign as a sign (called the head) containing other signs. Andersen's composite signs model the fact that "a computer based sign in an interface acquires meaning from its relation to other signs the user sees in the interface" (Andersen, 1992 p.23). Peirce's semiosis describes this process of signs generating other signs and endlessly signifying interpretants (even if Andersen's larger analysis is rooted in the binary sign). He defines three different types of composite signs based on the head's relationship to its contained

signs. First, in a constellation, each contained sign needs the head, but the head could lose either contained sign and it wouldn't effect the other contained signs. An example of this could be a card-based homepage where the card needs the homepage (head) but the homepage can lose or add cards without effect to other cards (see my diagram below).



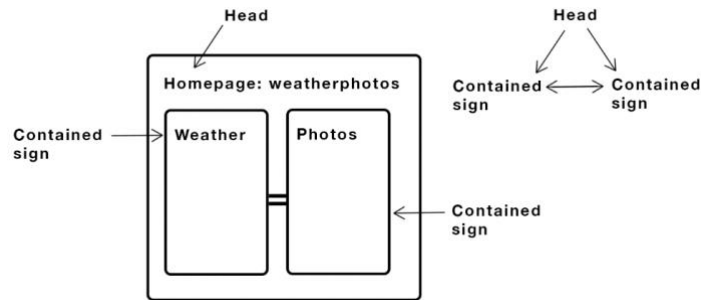
Constellation composite sign

Second, in a subordination composite sign “only the head makes a direct connection to other meanings in the text” (Andersen, 1992 p.21). This means that one or more of the contained signs is dependent on another contained sign which is not dependent on these subparts but is dependent on the head. For example, a weather app widget is dependent on the head homepage, but the different locations are dependent on the weather widget. If the locations are removed this does not alter the relationship of the weather widget to the head, but if the weather widget is removed it does affect the locations.



Subordination composite sign

Third, an interdependence relationship within a composite sign means that each contained sign is dependent on every other and changing any will change their relationship with the head. For example, if you had an interface that told you what photos you took based on what the weather was, taking away either the photos or weather widget would transform the head.



Interdependence composite sign

When these composite signs are created through semiosis, this creates a hierarchy of levels which become increasingly complex, as Andersen writes:



A total picture of the whole system will depict semiotic activities from the top down to the very bottom of the system. A computer system can be seen as a complex network of signs, and every level contains aspects that can be treated semiotically. (Andersen, 1992 p.7)

Andersen's last topic of conversation extends Norman's discussion of user and programmer models. Andersen writes that from a computer programmer's perspective, the domain (software and hardware) is designed for optimal functionality and efficiency (Andersen, 1992 p.28). However, Andersen warns against the programmer's perspective and advocates for a semiotic perspective. In this perspective, users and computers are viewed as one system, interface is not separate from functionality, and tasks are executed via "sign-vehicles users interpret in their context of work"<sup>13</sup> (Andersen, 1992 p.28). This adds nuance to semiotic theories of emergent meaning because in interface design, "both users and programmers interpret the system, and [there is] clash between those two kinds of interpretations" (Andersen, 1992 p.7). The designer builds their own model of the system, and the user builds their model based on emergent meanings from interactions with the computer. Digital interfaces are especially different from other types of media because the designer is able to assess the accuracy of user models and update the system design if designer and user models do not match as intended.

Andersen advocates for users and computers to be seen as one system, and Kammersgaard says this represents the media perspective<sup>14</sup> (Kammersgaard, 1988 p.356). Kammersgaard writes, "Andersen (1985) treats the designer as having the role of one sender in a collective of senders,

---

<sup>13</sup> Andersen frequently refers to his models in the context of the user at their job in which the code of work language is the most important. This is probably because Andersen was writing in the early 90s, which is only shortly after personal computers became widely popular outside of the workplace (Myers, 1998).

<sup>14</sup> Kammersgaard is referring to Andersen's earlier work "Semiotics and Informatics" (Andersen, 1985), which is the foundation of his more developed theories of computer semiotics like Computer Semiotics.

who makes a contribution to each message sent through the medium” (Kammersgaard, 1988 p.356). The media perspective is one of the most compatible HCI models with semiotic analysis, and de Souza will also use it.

Andersen touches on one other relevant point. Using symbols (by the design definition, not the semiotic) that are common and shared between users’ semiotic codes is often cited as a fundamental design principle. However, always following this principle of consistency may stymie innovation. Andersen posits the following question:

Should we base our system on a metaphor that users understand in order to ensure understandability, but running the risk of constructing a system that really do not give users new opportunities, or should we invent new ways of doing and looking at things, risking that nobody will understand it? (Andersen, 1992 p.12)

In conclusion, while Andersen was writing from a primarily structural non-Peircean semiotic perspective, his observations within this framework provide us with some new and productive ways to analyze user interfaces using both HCI and semiotic theories. His analysis of interface sign features gives us vocabulary to discuss sign types, and his description of interfaces as emergent meanings which cannot be standardized gives the designer insight on the importance of customization and diversity of sign expression. Last, Andersen’s composite signs are compatible with semiosis and amplify the importance of viewing signs at a system level in which they interact with other signs; designers can use this principle to order users’ perception of signs.

## **Mihai Nadin**

Mihai Nadin, born in Romania in 1938, is an HCI and computer science researcher at the University of Texas at Dallas. He is well established enough in the semiotics community to have written for *Semiotica* multiple times, a widely respected semiotics publication started by Thomas Sebeok. Nadin cites two guiding questions in his research: first, “What cognitive processes or mechanisms engender and support the creativity of the human being?,” and second, “Can machines emulate such processes in order to help creative endeavors?” (Nadin, 2023). De Souza writes:

In Nadin’s agenda there are four basic goals to be reached: a semiotics foundation for HCI; the incorporation of semiotic concepts in the analysis and design of operating systems; the introduction of Semiotics in academic HCI training programs; and a complementation of usability evaluation with semiotic adequacy testing. (de Souza, 2001 p.416)

Some of Nadin’s most important works are “Semiotics, advertising, and marketing” (1987), “Interface design: A semiotic paradigm” (1988), “Interface design and evaluation—semiotic implications” (1988), and “The Civilization of Illiteracy” (1997).

One of Nadin’s contributions is an explicit explanation of the benefit of applying semiotics to UID. First, “Since the technology upon and for which we build interface changes very rapidly, pan-logical semiotic principles, in their breadth and depth, provide a foundation for improved interface design (user and process interfaces)” (Nadin, 1988 p.283). Second, “designers approaching interface issues, particularly communication aspects, are obsessed with quantitative aspects or make intuitive decisions... Knowing that communication is a semiotic activity, we can rely on semiotic principles in order to improve the communication functions of interfaces”

(Nadin, 1988 p.292). This concept is known as semiotic adequacy, where instead of evaluating HCI quality retroactively, designers can proactively make good user interface languages by experimenting with semiotic principles of substitution (variation of representation), insertion (inserting representamen material until object adequately represented), or omission (leaving out things that could obscure semiosis) (Nadin, 1988; Nadin, 2020). Nadin also does a good job explaining how semiotics and programming can support each other. For example, he writes that “The higher the quality of images, the better the possibilities to generate a visual language for the interface and to support high quality applications,” and analyzes programmers’ ability to use certain coding languages and commands following the logic of Eco’s theory of sign production (Nadin, 1988 p.282).

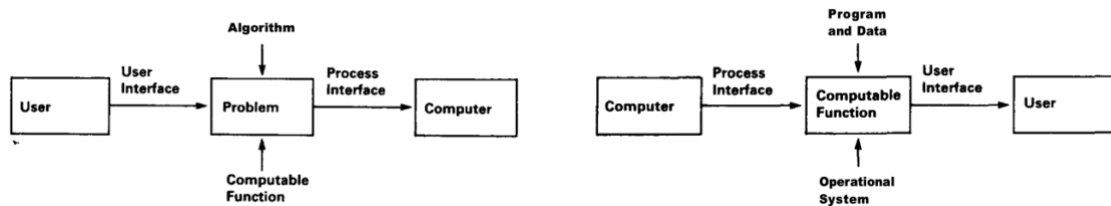
Nadin defines semiotics at the semiosphere level, writing “The object of semiotics is sign systems and their functioning within culture” (Nadin, 1988 p.269). Nadin also addresses the split between binary and triadic sign systems, calling triadic the “pan-logical definition” and deeming it most appropriate for his analysis (Nadin, 1988 p.270). This is because, according to Nadin, the triadic sign is based in a logic of signs rather than arbitrary cultural codes. Nadin diagrams Peirce’s sign complex below (Nadin, 1988 p.275) and interprets it much like Andersen, writing “Representation = that which represents; Object = that which is represented; Interpretant = the process of interpretation” (Nadin, 1988 p.270; Nadin, 2020). In this definition, Nadin seems to be conflating the non-perceptible representamen as Saussure’s signifier and the object as signified. However, Nadin does have a fairly accurate definition of the interpretant as the final meaning that emerges when the sign is perceived within a semiotic context; his evocation of Peirce is slightly more accurate than Andersen’s.

Nadin, like Andersen, also stresses the importance of semiosis and the presence of semiotic levels which combine increasingly complex and generative signs (Nadin, 1988 p.271). He writes, “no sign can be considered independent of its relation(s) to other signs” (Nadin, 1988 p.271). Nadin defines semiosis as a “Sign process, that is, the process through which the representations of an object and the associated representation are brought together”—this aligns with Peirce’s definition of interpretants and semiosis (Nadin, 1984). Nadin then evokes Peirce’s icon-index-symbol categorization for types of signs, but he uses the common misunderstanding of this triad that I have critiqued using Eco’s critique of iconicity, so it is not useful analysis for our purposes.

One of Nadin’s most useful contributions is his analysis of the mechanics of semiosis and emergent meaning. Nadin uses Jakobson’s speech act model to explain the generation of meaning using his six factors and functions. Meaning emerges because the user recognizes the sign through convention (perceptually identifies the pattern), then decodes its function in two ways: “(1) recognizing parts of the object in relation to each other and to the whole and arriving at some inference based on their interrelationship; and (2) recognizing the whole and inferring from the whole to the parts” (Nadin, 1988 p.283). Some factors that feed into Jakobson’s speech act context in Nadin’s model are “cognitive skills, emotional factors, [and] esthetic components” (Nadin, 1988 p.289).

Nadin also analyzes Shannon and Weaver’s 1947 theory of communication which bears striking resemblance to Jakobson’s speech act model. This model consists of an encoder, decoder, source, channel, noise, and receiver. To use one example, we can see Jakobson’s code factor in

their model when Shannon and Weaver write “The observer can insert himself at each sequence, conveying his observation in metalanguage” (Nadin, 1988 p.292). Synthesizing Shannon and Weaver and Jakobson’s models leads Nadin to create his own model called information theory (see user to computer and computer to user information models below) (Nadin, 1988 p.294, 295). As we can see, the information act does not operate the same depending on the direction of information flow, but it does resemble Norman’s evaluation and execution gulfs.



Nadin’s information theory model, based on Jakobson’s speech act model.

(left) User to computer, (right) computer to user

Nadin identifies 8 functions associated with his information theory model, not all matching up with one function (Nadin, 1988 p.294). The functions not native to Jakobson’s speech act model are expressive and design.

1. The function of communication - actually the function of maintaining communication, identified as the phatic function.
2. The expressive function - relating addresser and the message.
3. The metalinguistic function - dealing with the functioning of the code(s) used (expressing both the addresser-code and addressee-code relations).

4. The pragmatic function - dealing with the context and the way it influences communication (relation between addresser, addressee, and context).
5. The conative function - representing the attitude of the addressee toward the message (imperative messages are quite different from optional or query messages).
6. The design function - reflecting the way the addresser and addressee (in particular) relate to the medium.
7. The referential (or cognitive) function - dealing with the meaning of the message.
8. The formal (poetic) function - pertaining to the message's formal qualities (syntax errors, for instance, are but an indication of this function).

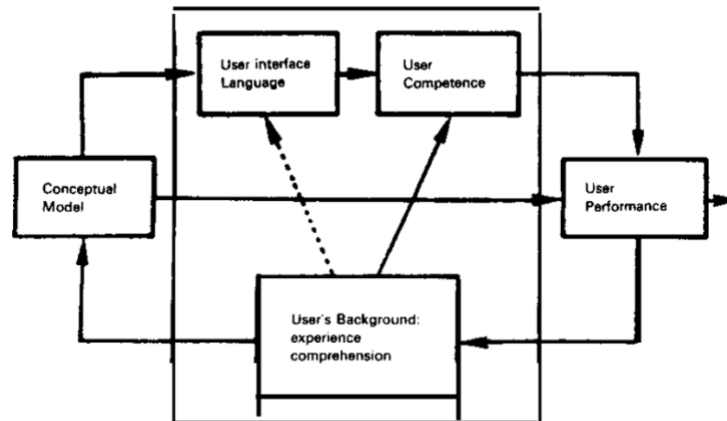
Nadin defines an interface as “the meeting place between two different entities that are supposed to come in contact, to be brought together - that is, to communicate .... It follows that interface has the nature of a sign” (Nadin, 1988 p.272). Nadin also expands the interface from just the inputs, outputs, and interaction language to “every point of contact between the computer and the user ... from product design to service (support, documentation, tutorials, seminars, packaging, etcetera)” (Nadin, 1988 p.274). Nadin continues, “Interface is also a problem of human-to-human relations, especially in the context in which human contact and interinfluence become more and more mediated” (Nadin, 1988 p.272). This doesn't only happen between users, but also between designers: “Once the user accepts a language, he will apply it according to the rules the designer embedded in the interface, and their communication, mediated by a certain machine, will take place” Nadin writes (Nadin, 1988 p.284). This is a clear instance of Kammersgaard's media perspective of HCI (Kammersgaard, 1988). This also leads Nadin to view “the sign as a mediating entity and semiotics as the theory and practice of mediation” (Nadin, 1988 p.272). Nadin focuses on mediation between people as opposed to non-human entities, which would place semiotics in the lens of human to human multimedia communication.

Nadin takes a similar stance as Andersen on Norman's user models. He writes, the "user interface should allow for the personal user model to be formed while the user learns the language of the interface...Quite often, the personal user model differs from the conceptual model upon which user interface is developed" (Nadin, 1988 p.275). Nadin's user interface is a semiotic system that makes sure these systems remain logically consistent—in other words, bridging Norman's gaps. Nadin expands on this design principle of consistency. He defines semiotic consistency as "uniformly using whatever means of representation are considered adequate," in order to create and reinforce a code of representation within the user's model. This user model is influenced not only by the digital screen but also "influenced by manuals, guides, tutorials, etcetera" (Nadin, 1988 p.287). The set of consistent rules a designer uses to create an interface can be called its design language (Nadin, 1988 p.285). He writes, "The language should function according to a logical structure which the user can grasp and which, while adhering to the spirit of computing logic, should not contradict so-called natural logic (cultural background as the environment of human logic)" (Nadin, 1988 p.285). A good user interface language has "reliability (with tolerance toward the user, if possible), self-sufficiency, ease of use, and adaptability" (Nadin, 1988 p.298).

Eco's theory of sign production also helps us understand this principle; as I cite above on page 32, "Inventing a new type of sign that is not already embedded in cultural code, creating new associations, and codeswitching"—which happens when a designer uses inconsistent signs—"leads to more labor," which makes the interface less easy to use (Eco, 1976 p.152). While Nadin does not draw this connection to Eco, placing the design principle of consistency in a semiotic context



gives it complexity and depth. Nadin's user model diagram is shown below; it adds to Norman's (shown previously) by specifying the processes within the user model (Nadin, 1988 p.288).



Nadin's user model

I see another parallel between Nadin's work and Eco's theory of sign production; Nadin writes "amount influences the time required to process," meaning larger amounts of signs result in longer processing times per Eco's TSP (Nadin, 1988 p.280). Nadin also specifies three command types that affect user (not computer) processing times; prefix, infix, and postfix. He condemns focusing on computer processing times because "instead of taking time to realize how a command should be represented in order to optimally support HCI, the technologist will more often than not rely on more memory, higher speed, and more bandwidth" (Nadin, 2020). This does not align with the goals of the semiotic user model, which prioritizes the user's ease of interpretation. Nadin's command types use the language of object oriented programming, specifying three ways we can represent the relationship between an object (interface sign) and operations that change its representation. Prefix commands specify an operation acting on a specified object, postfix selects the object then specifies the operation, and infix connects several

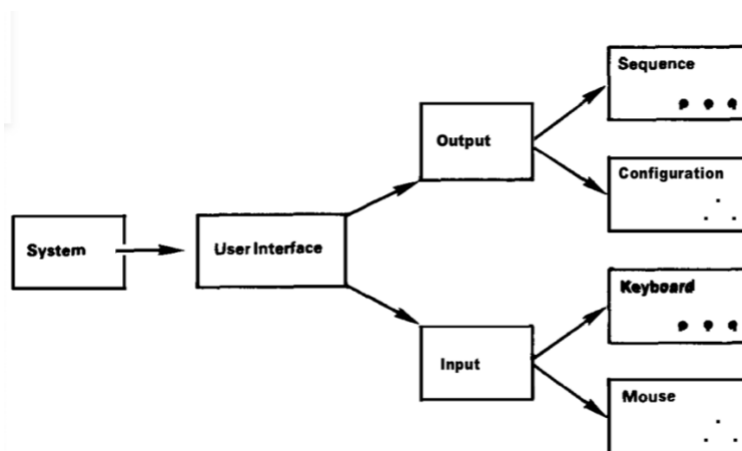
operations together<sup>15</sup> (Nadin, 1988 p.276-279). Higher processing time occurs when using prefix commands because every time the object of the command is changed, the command must be change as well; “the prefix command is ‘governed’ by a ‘regime’ that is very ‘repressive’” says Nadin (Nadin, 1988 p.281). In Nadin’s regime metaphor, the more repressive the regime, the stricter the grammar of the command is and the less “wobble room” it has. The postfix command is more of a “benign regime... There is no real ‘repression’ in respect to choosing an object, yet every user is bogged down in a relatively great amount of ‘red tape’ ... The logical design is such that the editor makes available only actions that may be taken at the system’s current state” says Nadin (Nadin, 1988 p.282). In other words, postfix takes less time than prefix but time increases if the wrong object is selected for the user’s desired outcome of command. Infix is a combination of prefix and postfix and contains both limitations, but takes less time than either when multiple commands are needed. It is the popularity of postfix commands that created the What You See is What You Get (WYSIWYG) HCI paradigm. While this can be a simple way to create visual interfaces, it has the drawback of alienating the user from the inner workings of the computer system; it “gives no clues as to what influences a certain format, why some changes are not possible, and why there is no consistency between the different formatting capabilities or between the different applications used” (Nadin, 1988 p.282). As Brian Kernighan remarks, sometimes “What you see is *all* you’ve got” (Kernighan in Nadin, 1988 p.282).

Nadin builds on another idea Andersen introduced: the designer is not designing for a homogenous user base, and interfaces must be designed to incorporate different user goals. Novice and expert is too simple a separation for users, so Nadin develops and multi-dimensional

---

<sup>15</sup> Postfix commands are particularly useful in visual language interfaces (Nadin, 1988 p.280).

matrix of user types. Users exist on a scale of experience (with the interface), comprehension (“from culture and general education”), imagination (“adapting programs to new functions”) (Nadin, 1988 p.290). The imagination axis is an early precursor of Norman’s theory of emotional design. Nadin suggests allowing different users to customize their interfaces based on two information processing modes: sequential (time based, differentiation focused) and configurational (space based, integration focused) (diagram below) (Nadin, 1988 p.191). Time and space appear to be the most natural way users mentally organize computer systems, as Andersen also finds in research on Swedish post office workers (Andersen, 1992 p.23). Designers should “avoid abrupt switching from one mode to the other,” while acknowledging “The two modes are interrelated, interfere with, or try to suppress each other; [but] under certain circumstances, they enhance each other” (Nadin, 1988 p.191). However, too many customization options can confuse the user instead of increasing their ability to accomplish goals: “to provide a really user-friendly interface means to make possible not everything, but only what is acceptable” (Nadin, 1988 p.292).



Nadin’s information processing diagram

Nadin also touches on a topic which is still in its early phases today; localization. Localization means customizing the interface based on users with different codes, usually in the context of regions of the world for working designers (Hartson & Pyla, 2012). Localization is about changing the entire system, not just translating words; as Nadin says, “Only a superficial designer, one who targets the lower level of the market, would think that cultural adequacy is reducible to emulation of characters used in foreign languages” Hartson and Pyla corroborate:

Localization goes beyond languages and translation. If one were to examine the home page of [Yahoo.com](http://Yahoo.com) in English and [Maktoob.com](http://Maktoob.com), one of the Arabic world’s most popular portals in Arabic, one would find not only language differences, but differences in color, imagery, organization, and topics of primary interest. There may be geographic, historical, political, aesthetic, and language differences. (Hartson & Pyla, 2012)

Arabic is an excellent example because if the designers wants to make a useful interface, the designer must design this interface so that it uses the most conventional organization possible, which the user can navigate fastest and with least effort. Arabic reads left to right, and if the designer stayed in the American right to left convention the interface would be extremely challenging for an Arabic-speaking user; localization is simply about applying these principles to differences in world languages, but its logic is fundamental to every user who has a different code to the next.

Nadin also expands the popular user interface analogy of the office desk, writing “it is not a collection of hardware, but essentially an environment where communication (exchange of documents, storage and retrieval of data, planning, etcetera) is possible and necessary” (Nadin,

1988 p.276). This view of the office metaphor explains why “office systems successful in supporting individual office work but not communication have never really made it” (Nadin, 1988 p.276).

In conclusion, Nadin gives us a number of useful applications of semiotics to UID. One of the most impactful is his focus on the mechanics of emergent meaning; he develops Jakobson’s speech act model into a computer specific model of the factors that influence user interpretation. Nadin’s semiotic perspective of emergent meaning also allows us to see that designers can’t develop a standard message that is transmitted from the designer to all users, and instead the interface must adapt to the preferred representation for the user and their goals. Nadin also gives nuance to the mechanics of the user modal, focusing on how semiotic consistency is achieved and used to build the user model, as well as how designers can develop consistent interface languages.

### **Clarisse de Souza**

De Souza is an HCI researcher from the Pontifical Catholic University of Rio de Janeiro with backgrounds in computational linguistics, translation, and informatics, and is a visiting professor at Stanford University (PUC Rio, 2023). She is a member of the Association for Computing Machinery’s Special Interest Group on Computer-Human Interaction (SIGCHI), which is a prestigious group of HCI researchers including Norman, Carroll, Nadin, Grudin and Myers who I cite in this essay (ACM, 2023). De Souza’s major contribution has been her Semiotic Engineering Research Group, which explores applications of semiotics to HCI through her theory of Semiotic Engineering, which I will elaborate below. While she began publishing in the

1980s, de Souza is still active and has the advantage of being able to draw from a more developed body of research on semiotics and interface design than pioneers Nadin and Andersen. In her development of semiotic engineering theory she draws heavily from theories I have covered from Peirce, Saussure, Eco, Sebeok, Norman, Kammersgaard, Andersen, and Nadin. Three of her most relevant works are “The Semiotic Engineering of User Interfaces” (1993), “Semiotic approaches to user interface design” (2001), and “The Semiotic Engineering of Human-Computer Interaction” (2005).

De Souza defines semiotic engineering as a subfield which analyzes “the synthesis of communication codes and protocols between humans and computer systems” in the service of “[designing] good user interface languages” (de Souza, 1993 p.753). The goal of semiotic engineering is “how Semiotics can contribute to central issues in HCI, namely software evaluation, multi-user interface design and extensible applications design” (de Souza, 2001 p.417). While Andersen focuses on high level semiotic modeling of human computer interaction, de Souza’s approach is more similar to Nadin. Nadin analyzes more low-level processes of emergent meaning in interfaces, and de Souza operates at this level but with more focus on the production of user interface languages. However, de Souza chooses to focus most on Eco’s theory of sign production because “Eco’s TSP grants us a synthesis-oriented perspective which is paramount to all design activities, and allows us to sketch the basis of a theoretic approach” as opposed to Nadin and Andersen’s primarily analytical theories<sup>16</sup> (de Souza, 1993 p.754).

---

<sup>16</sup> From the Swedish morphological society: “analysis is defined as the procedure by which we break down an intellectual or substantial whole into parts or components. Synthesis is defined as the opposite procedure: to combine separate elements or components in order to form a coherent whole” (Ritchey, 1996).

De Souza explains the benefits of applying semiotics to UID in a similar way as Nadin. She writes that designing good user interfaces currently relies on a “loosely structured set of DOs and DON'Ts” derived from retroactive “successful and unsuccessful attempts,” but semiotics and HCI together could provide a theoretical background for this practice that supports and defines the “synthesis of communication codes and protocols between humans and computer systems” (de Souza, 1993 p.753). She also writes that “The main contribution of semiotic theory is to provide HCI designers with new perceptions on the process and product design, binding *all* the stages of software development and use in a unique homogenous space” (de Souza, 2001 p.417). These contributions lead to de Souza’s main thesis of semiotic engineering:

If designers are led to conceive of systems as a distinctive type of message they are sending to users an engineered metacommunication artefact many of the misunderstandings and deadlocks possibly occurring in human-computer interaction can be avoided. When users realize they are not interacting with autonomous machines, but with a rational product of a human mind, they can resort to a wealth of beliefs and expectations they have regarding the intellectual and creative behavior of other people. (de Souza, 1993 p.754)

She further differentiates the benefit of semiotics to HCI from the benefit of cognitive science. The name semiotic engineering is based on Normans’ cognitive engineering, and de Souza says the two complement each other but while “Cognitive Science presents designers with the appropriate targets to be hit, Semiotics presents valuable guidance for making successful shots” (de Souza, 1993 p.754). This is reminiscent of Sebeok’s differentiation of semiotics as an epistemology, not an ontology.

De Souza defines semiotics as “the study of signs, signification processes, and how signs and signification take part in communication” (de Souza, 2005 p.3). Semiotics is “generally concerned with ‘meaning-making and representation in many forms’ (Chandler 2002) and with ‘everything that can be taken as a sign’ (Eco 1976)” (de Souza, 2005 p.35). De Souza embraces the more anthropological perspective of semiotics as I introduced via Eco and Lotman, citing its appropriateness for studies of the internet and the computer as mediator. She writes, “Since sign systems are produced and perpetuated by cultures, some contemporary semioticians define semiotics as a theory of culture (Eco 1976; Danesi and Perron 1999)” (de Souza, 2005 p.3).

De Souza’s focus is on user interface *language* design, and she defines user interface languages as “the expressive system in which all messages are coded” (de Souza, 1993 p.754). This takes a more observational approach than Nadin, who defined user interface/design languages as “The set of consistent rules a designer uses to create an interface” (Nadin, 1988 p.285). While a user interface is broader than a user interface language, designers can only really create the user interface language contained within the user interface, so the term user interface language design (UILD as de Souza abbreviates) is basically the same process as UID.

De Souza roots her analysis in Kammersgaard’s four models of HCI, focusing on the dialogue partner and media perspectives. De Souza defines the dialogue partner perspective as when “users and systems are taken as equivalent parties in a conversation” (De Souza, 1993 p. 753). While this is true within Kammersgaard’s definition, she fails to stress that “equivalent” in this model means the computer participating at the level of a human in conversation, not the human as computer or both at a compromised communication level (Kammersgaard, 1988). Focusing on



the dialogue-partner perspective does not seem to be the best approach: as I wrote earlier, Kammersgaard himself wrote this perspective does not account for communication breakdowns and he finds “the dialogue partner perspective problematic” and less relevant than the other three perspectives (Kammersgaard, 1988 p.358). Nadin also finds that “[interacting] with the computer as if it were a helpful human being, perhaps chatting in natural language... should be contradicted, not only because it raises false hopes, but also because the underlying principles of digital computers are those of Boolean logic - a reduction from the multivalued logic of natural languages to two-valued logic” (Nadin, 1988 p.300). De Souza might have focused on this approach as a result of her work in “AI interpretability and explainability” (PUC Rio, 2023). However, it is puzzling that in 2001 de Souza herself writes about Nake and Grabowski’s view, which she agrees with, that “computers lacks some of the fundamental capacities that enable humans to communicate...computer meanings are something we humans cannot experience, and vice-versa” (de Souza, 2001 p.416-417). De Souza seems to have moved away from the dialogue partner perspective later, and in her 2005 book “The semiotic engineering of human-computer interaction” she only cites the media perspective (de Souza, 2005).

De Souza’s focus on the media perspective, on the other hand, is a very useful one and the model which I will focus on. In the media perspective, she writes that “systems are viewed as a communication *medium* through which messages are passed between human parties” (de Souza, 1993 p.753). Since messages passed between human parties constitute communication, her definition is the same as Kammersgaard’s. Her model is based on her diagram of the media perspective (see (b) below: (a) is her diagram of the dialogue partner perspective) (de Souza, 1993 p.755). De Souza defines a system as a message that “can send and receive other messages-

they are metacommunication artefacts that should be engineered according to explicit semiotic principles” (de Souza, 1993 p.753). The designer is the coder of the user interface language, and the user decodes it. Users primarily run into difficulty understanding the system’s behavior, which is where de Souza evokes Nadin and his emphasis on designing an interface language with consistent rules. De Souza says the user sending messages back to the designer via data and updates is out of the scope of her analysis; this makes her model much closer to the author-reader relationship and is reminiscent of Eco’s theory of interpretation. However she still differentiates the user-designer relationship in the context of interface design because it is a “performing message” (de Souza, 1993 p.756) which performs a range of functions in a range of contexts which books do not (de Souza, 2005 p.24).

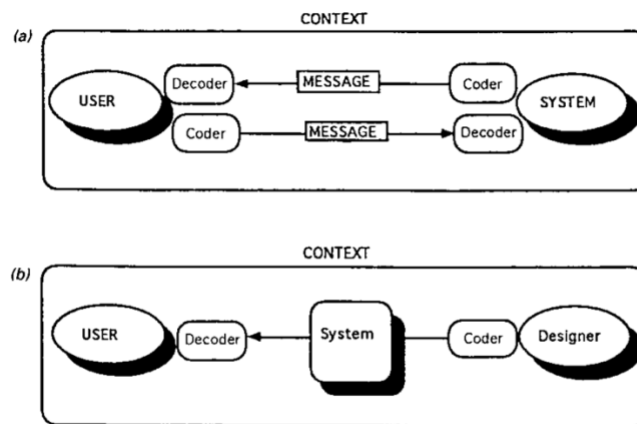
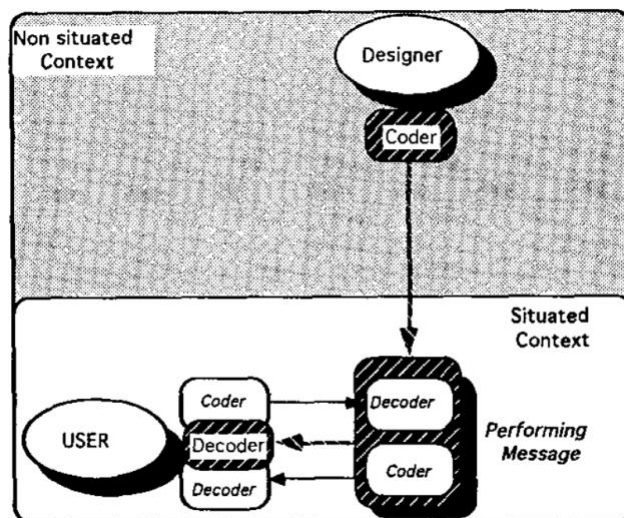


FIGURE 2. Possible communicative frameworks in HCI. (a) User in communication with system; (b) user in communication with system’s designer.

### De Souza’s media perspective diagram (b)

De Souza adds complexity to the media perspective in her diagram below, in which the system is the designer’s performing message (de Souza, 1993 p.756). This is her two-message model. The black elements with diagonal lines correspond to the original media diagram (b) above.

However, de Souza also places the dialogue partner perspective (the white elements) within the system-user relationship. If the system is viewed not as isolated but as a message from the designer, this somewhat resolves the tension within the dialogue partner perspective that the system cannot be considered a human interlocutor. The user being viewed as a coder in the dialogue partner perspective also adds to the media perspective by acknowledging that as the user builds a mental model of the interface language via each performing message (an instance of the system), the user adjusts the code they use to send messages back to the system as they interact. This two-message model is mindful of emergent meaning; the “situated context” area represents each instance a user interacts with the system, in which a different performing message and decoded message will emerge every time. In contrast, the designer only designs the system once and not in the context of each user interaction.

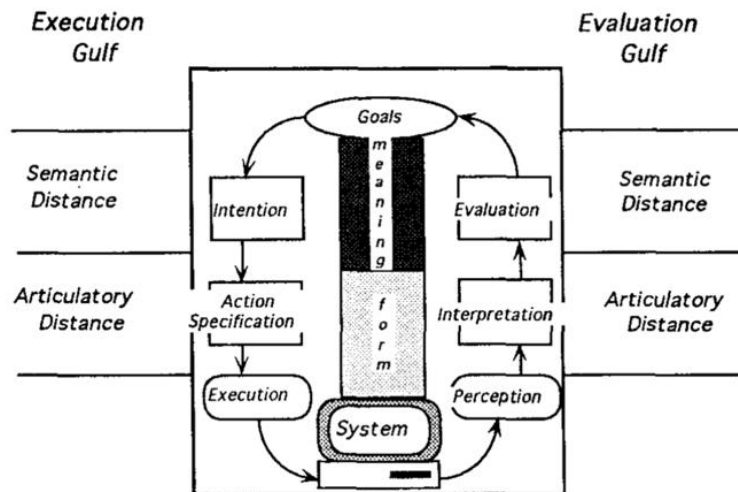


De Souza's two-message HCI model (de Souza, 1993 p.756)

De Souza focuses deeply on the content and expression levels of the performing message, which she says is the user's main pain point in understanding the system's behavior. "Semiotic theory

plays an important part in supporting UILD at the *expression* level; the design of message *content* is substantially supported by cognitive psychology,” she says (de Souza, 1993 p.757). She draws on Norman’s execution/evaluation gulfs and Eco’s theory of sign production to do this.

De Souza analyzes Norman’s expansion on the execution and evaluation gulfs found in his 1986 article on direct manipulation interfaces with Hutchins, in particular his semantic and articulatory distances (Hutchins et al., 1985). Semantic distance is the distance between the intention and the interpreted meaning of the expression for the user, and articulatory distance is the distance between the physical form of the expression and its meaning within the interaction language (de Souza, 1993 p.758). When distances are smaller, this means the interface is “more direct,” or better in respect to our interface design goals. See diagram below (adapted from Hutchins et. Al, 1985 p.331 in de Souza 1993, p.759).



Norman’s semantic and articulatory distances

To close these distances, de Souza cites a need for a taxonomy of signs which “explain or predict what happens when signs must be put together to create messages” (de Souza, 1993 p.762). She discusses the history of using the Peircean icon-index-symbol triad as a taxonomy, which has been the primary method of applying semiotics to design. This is perhaps because of the confusion surrounding the word “icon,” which is appealing to designers familiar with the word, but which has entirely different meanings in Peircean semiotics and visual design. De Souza discusses the misinterpretation that arises from thinking about the icon as a visual sign and evokes Eco’s critique of iconism to counter this—however, she concludes that the icon-index-symbol triad is unfit because “there are dozens of others in his original proposal, many of which overlap, leading to a labyrinth of types and criteria” (de Souza, 1993 p.762). So while she uses Eco’s critique which none of the other authors do and which suggests a more nuanced understanding of Peirce, she doesn’t conclude that this triad is “firsts of seconds, and because they are not yet at the level of thirdness they cannot be perceptually analyzed” (as I say above on page 31); she instead concludes that we are unsure what Peirce’s taxonomy is and it might include many other types. This analysis doesn’t align with mine, but because she focuses on Eco and not Peirce in the end, it doesn’t have an enormous impact on her argument.

De Souza prefers Eco’s TSP because Peirce’s triad is structure oriented (models signification) and supports the “evaluation of designed codes, but [it fails] to support the design process of metacommunication artefacts” (de Souza, 1993 p.762). “Eco’s TSP, on the other hand, with its explicit account of sign production processes, makes a number of potential contributions to design (Anderson, 1990)” (de Souza, 1993 p.762). She says TSP is not structure oriented and models communication, and is thus better for design. This is because by Eco’s definition, communication

occurs when a message, encoded in a signification system, is perceived by an addressee, and the end to end process of communication is more fruitful to study than the individual instances of signification when analyzing user goals (de Souza, 1993 p.762; Eco, 1976 p.8-9). This is also the reason she prefers Eco's TSP over the other half of his "Theory of Semiotics," which is the structure-oriented theory of codes.

To review, this is how I have interpreted Eco's TSP (from page 32):

The types of effort which are always present are choosing the type of signal, physically producing the signal, composing a string of expressions, and making the sign utterance be perceived by the addressee (see diagram below) (Eco, 1976). Inventing a new type of sign that is not already embedded in cultural code, creating new associations, using unfamiliar codes, and codeswitching leads to more labor (Eco, 1976 p. 152).

De Souza identifies "four parameters which provide for a classification of the modes of production of signs: the physical labor performed by the agent, the token/type ratio of the sign, the type of continuum selected for expression and the mode and complexity of articulation" (de Souza, 1993 p.763). These are based on Eco's sign production chart (shown below), which also details which sign production types require more labor (Eco, 1976 p.218). Choosing the token/type ratio is choosing the type of sign, and type of continuum and mode and complexity of articulation both concern the relationship of the sign to signs around it. She does not address the labor in opening a communication channel. The token parameter addresses the labor required to use a certain expression system; if there is a word for something and you can verbally express it that requires less labor, but if there is not a word or picture or other expression system already recorded for expressing something that would take more labor (de Souza, 1993 p.763). Moving

on to strings of expressions, type of continuum addresses what expression we use in the context of our other expressions in an act of communication (de Souza, 1993 p.763). For example, quoting someone in an essay where both tokens of expression are writing (homomaterial) would take less labor than showing someone a verbal picture in the middle of a verbal explanation (heteromaterial)<sup>17</sup>. Mode and complexity of articulation refers to the presence or lack of coded combination rules within an expression system (de Souza, 1993 p.763). For example, visual forms of expression are often easier to express hierarchy within given size, color, and position, but expressing hierarchy in a written expression form if that is one's message is less codified.

Physical labour required to produce expressions	Recognition			Ostension			Replica				Invention	
<i>Ratio difficilis</i>	imprints						vectors					congruences
<i>Ratio facilis</i>		symptoms	clues	examples	samples	fictive samples	stylisations	combinational units	pseudo-combinational units	programmed stimuli	projections	transformations graphs
Continuum to be shaped	motivated heteromaterial			homomaterial			arbitrary heteromaterial					
Modes of articulation	pre-established, coded and overcoded grammatical units according to different modes of pertinence										proposed undercoded texts	

Eco's modes of sign production

<sup>17</sup> CAD presents an interesting environment for homomaterial/heteromaterial signs. Designers must use 2D computer interfaces to represent 3D shapes (as in virtual reality and CAD); something 3D cannot be expressed homomaterially (de Souza, 1993 p.770). I/O and direct manipulation interfaces also almost always have to rely on heteromaterial representation.

De Souza's four identified parameters of TSP lead her to four major semiotic guidelines that support better interface design:

Guideline 1: User interface language designers should produce signs they recognize as existing codified expressions of the intended contents.

Guideline 2: User interface language designers should try to select expressions that are recognized as a token of an established type of expression system which accounts for the intended contents (*ratio facilis*).

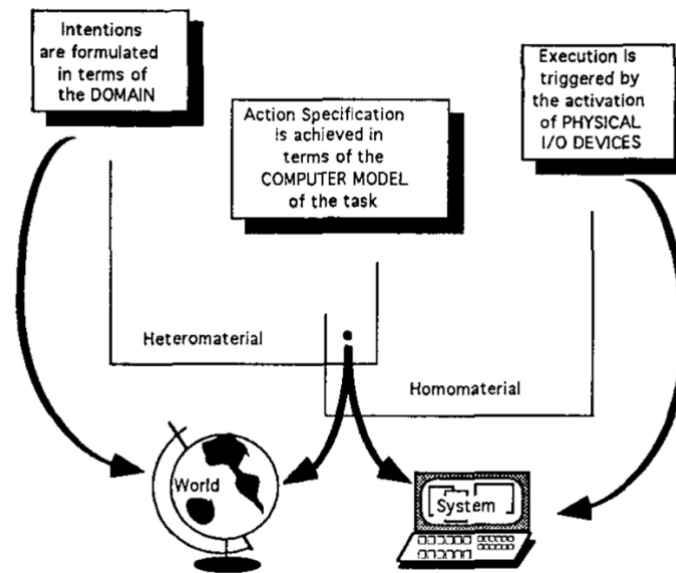
Guideline 3: User interface signs referring to domain objects and to computer-modeled solutions for problems should be heteromaterial, whereas the representation of I/O elements should be homomaterial and subject to direct manipulation.

Guideline 4: User interface language designers should always resort to expressions derived from a recognizably codified (rule-based) system. (de Souza, 1993 p.764)

Guideline three refers to domain objects, which are the instances of Kammersgaard's "domain representation." Domain objects are stored in the language of the computer and only specific properties of these objects are useful to the user, which means that the expression form of the domain object should not be the same as the property which is expressed to the user (it should be heteromaterial). For those familiar with computer science, this is the object in an object-oriented programming language. For example, if a domain object is all the properties of a virtual store inside the computer, the user might need the number of items inside the store, which could be expressed as a numeric string of text on the screen. User interface signs must be expressed as I/O (input/output) elements, which are perceptible via the visual screen, audio speakers, etc. I/O elements must be homomaterial within themselves if the elements are to interact in order for the



easiest interface usage. De Souza inserts homomaterial and heteromaterial signs into Norman's interaction model, as seen below (de Souza, 1993 p.766). The execution gulf (user to system) is bridged when the user forms intentions based on their user model of the domain and its objects, and then they identify how to express their intention using the mechanisms which will change their desired variables, after which they execute using the I/O elements.



The use of homo/heteromaterial expression in bridging Norman's gulf of execution

Guidelines one, two, and four can be recognized as part of the principle of consistency (both within logic and wider cultural code), which is also supported by Nadin's semiotic analysis.

While de Souza demonstrates a deep understanding of semiotics, one of the largest problems I have with her analysis is the assumption that the designer is able to recognize what is

“systematically coded and deeply embedded in the users' culture” (de Souza, 1993 p.766). For example, between two flower icons how is the designer to say which is more recognizable in the

user's code? The answer seems to lie in research and user testing for recognizability, however this is time consuming and expensive and can only be done for the largest and most important elements of an interface. This also raises the dilemma of the diverse user base with diverse goals; one type of expression will not be the most recognizable within every user's code at every instance they use the interface. This demonstrates the importance of Andersen's insistence on customization for different user codes.

One way de Souza seems to offer a solution is to prefer using natural language, which "is an available highly coded expressive system," as opposed to pictorial expressive systems are too complex and un-analyzable (de Souza, 1993 p.767). However, earlier in her own writing and in agreement with Norman, she writes that "Conversational interfaces tend to offer greater distances and weaker engagement than direct manipulation ones"; conversational interfaces are textual based and direct manipulation interfaces are based in visual icons (icons in visual design language, not Peircean) (de Souza, 1993 p.759). An interesting tradeoff she raises between conversational and direct manipulation interfaces is that people tend to learn via insight learning in conversational interfaces, meaning they experience "a sudden leap from error prone to error free performance" (Svendson, 1991 p.379 in de Souza, 1993 p.768). In direct manipulation interfaces, they tend to learn through trial and error, which is "a sudden leap from error prone to error free performance (Svendson, 1991 p.379 in de Souza, 1993 p.768). While no method makes the user more of an expert in the infinite long run, "since users typically do not like to make conscious efforts in dealing with computers, direct manipulation interfaces might be taken to be more user-friendly than command-driven ones" (Svendson, 1991 in de Souza, 1993 p.768). However, for more complex and effortful tasks, conversational interfaces may be more fitting.

De Souza also cites the usefulness of analogies in direct manipulation interfaces. Analogies can be a foundation for the signification system the designer is creating which speeds the user's learning. The function of the system inherits all the rules of the analogized system without having to be specified; no two systems function exactly the same so this can create confusion later in the learning process, but has been called "the best way to introduce a new user to a computer system" (Halasz & Moran, 1982 in de Souza, 1993 p.768). "The designer should also be aware that one concept may be structured by more than one metaphor" (Lakoff and Johnson, 1980 in de Souza, 1993 p.768).

In conclusion, de Souza's overarching goal is one of metacommunication, in which users and designers see user interface language as a message from the designer to the user. This leads the user to "realize they are not interacting with autonomous machines, but with a rational product of a human mind, [and] they can resort to a wealth of beliefs and expectations they have regarding the intellectual and creative behavior of other people"<sup>18</sup> (de Souza, 1993 p.754). She does not propose specific ways to raise this awareness in users, but I assume it would spread through the adoption of semiotic engineering in the design community and publicization to the public. Semiotics' primary purpose is to provide a theory that gives recommendations for how this model should express itself. De Souza does not attempt to base her analysis on Peirce, but her HCI model, which is based on Kammersgaard's media perspective but incorporates elements of dialogue partner, does touch on emergent meaning by situating the user in the context of one

---

<sup>18</sup> This could also lead to more awareness of bias in computing. When the user assumes the computer is perfectly logical, they erase the potential to recognize designer biases. This fits into de Souza's larger argument because bias is merely one part of the designer's code.

instance. She uses this model when stating the goals of interface language design, which is to design signifying systems that require the least sign production labor to interact with (using Eco) and which minimize the evaluation, execution, articulation, and semantic gulfs (using Norman). In order to meet these requirements she creates four semiotic guidelines, which rely on semiotics models of cultural expression (in the vein of Lotman's semiosphere, though she does not cite him) and content as opposed to the single sign complexes that Nadin and Andersen focus on. These guidelines are based in TSP but describe the difficulty of interacting with such sign systems, which suggests it is not the difficulty for the computer to produce the signs that is important but the signs as they are produced with user interaction, which is a nonstructural point of view. Consistency is once again enforced, and de Souza ends with an analysis of textual/direct manipulation interfaces.

### **3.b Connections and synthesis**

I have shown that semiotics is very productive, and even necessary, to apply to UID. The three theorists I have analyzed who apply semiotics to interface design—Andersen, Nadin, and de Souza—are at the forefront of this effort, on which little has been written. My intent in this section is to analyze which of their ideas are the most semiotically sound and which we can recommend to bring into UID.

These three authors come from a primarily HCI background and are not native scholars of semiotics. A common thread among them is the frequent misinterpretation of Peirce, especially the icon-index-symbol triad, the rebuttal to which I discussed using Eco's critique. De Souza identifies this triad most accurately, but still does not apply it in a fruitful way. While these

authors evoke Peircean semiotics, many demonstrate through their analysis that they are still working with a binary sign. Only de Souza never explicitly says she is a structuralist who uses the binary sign, however she focused the semiosphere (not individual sign complex) level. Peircean semiotics is very useful in interface design when we view the perceivable interface as immediate object, the users (including designers) as the dynamic object, emergent meaning as the final interpretant, and semiosis as a process which is essential to the decoding of interfaces. While none of these authors drew the connection between the immediate object and the perceivable interface, every author discussed the interface as the result of emergent meaning, and Nadin in particular shows a good understanding of dynamic interpretants. Nadin and Andersen both implicitly refer to semiosis when they detail the generative relationship of interface signs to other signs in an increasingly complex hierarchy. The binary sign also has its place, and Andersen's handling features exemplify how analyzing binary signs can be helpful for UID. The distinction between content and expression also comes from the binary sign, and understanding that semiotics complements HCI in the realm of expression (as de Souza writes) gives designers a better idea of its relevancy.

Looking at the user interface from a semiosphere level and using Eco and Jakobson, as all three authors have, has proved to be just as if not more useful than the sign complex level in the context of interface design. This situates the interface interaction in context and views the communication process as a whole, which helps us understand processes of mediation and communication better.

One consistent semiotic concept is that the user interface is the relationship between the user and the computer system, not just the physical inputs/outputs that many HCI and design researchers refer to. I prefer Andersen and Carroll's definitions, which agree that the interface is a sign system which mediates between the computer and the user's representational systems. Although Peirce did not model sign systems, only signs in the process of semiosis, we can use Peircean terminology to describe some interface aspects. Meaning emerges from the interface via final interpretants each instance a user interacts with the computer system, and the perceptible components of this instance (input and output interaction involving the screen, speakers, etc.) can be considered the immediate object. Nadin details the mechanics of emergent interfaces using Jakobson's speech act model and his own information theory model. Andersen's composite sign model gives us a good theoretical background for the different types of Peircean semiosis that can happen depending on the relationships signs in a hierarchy have with each other within the interface, even if the signs Andersen discusses are binary signs.

The user interface/design language is the set of rules the designer uses when creating the interface, with better languages being easier for the user to learn. Seen in both de Souza and Nadin, this language is made easy to learn by the consistent use of many factors, including sign expression types, combinations, logical structure, and cultural code. The rules a user learns the user interface language through are supported by Peirce's four methods of fixation of belief, which dictates how a user learns, unlearns, and updates their user model based on the cycle of belief, doubt, and inquiry. While interface language is not a focus in Andersen, his sign handling features also explore different types of expression that must be consistent within hardware-display, display-user, and user-culture codes. The designer should code the fewest new

associations possible when creating an interface language, which they can do through consistency with cultural code, especially the use of analogies which carry their own rules. One of the most successful examples of this continues to be the office-desk analogy, which is used by most successful consumer computer interfaces. This principal of consistency is one of the most fruitful areas in which to apply semiotics because it gives it nuance, depth, and theoretical backing above and beyond what HCI and design fields contribute.

An understanding of Eco's theory of sign production has also proven useful in UID. Because the production and existence of a sign in the first place requires a user, signs that are harder to produce are also harder to process for the user. TSP gives theoretical backing to the principle of consistency by explaining *why* inconsistent sign usage makes interfaces more difficult to use, from the physical form of expression, to expression of surrounding signs, to cultural or logical newness. De Souza uses this theory to support her UID guidelines of consistency. While he doesn't explicitly mention TSP, Nadin also looks at problems of processing speeds in his command types, which connect how a sign is produced to the ease of production of its desired operation.

Kammersgaard's media perspective has proved to be a model consistent with semiotics and which Nadin and de Souza, and to some extent Andersen, rely on. This idea of the interface mediating between designer and user perspective is highly compatible with Norman's designer, system, and user models, in which the system image mediates between designer and user models. Viewing HCI through the lens of these two semiotically compatible models is highly fruitful. However, a caveat which De Souza exemplifies is that trying to close execution and evaluation

gaps by using a singular, general user model is impossible because of users' diverse codes. As Andersen introduces and Nadin supports, interface representations must instead be designed based on narrower codes within situated contexts that users can switch between and have the option to customize by. However, the diversity of expression to account for becomes smaller as users become fewer, more homogenous, and with more specialized goals.

An important aspect of the media perspective which de Souza extrapolates to become her main argument is that if the computer system is viewed as a message from the designer—a human—users might get less hung up on building a perfect logical model of the system and instead give it the room for error and poetics that we would give the designer. This fits well with Nadin's model based on Jakobson's speech act, in which viewing the system as a human communication act gives us eight factors by which we can analyze the message with.

I also want to make the explicit connection between user centered design and emergent meaning. As I wrote earlier, "user centered design changed the focus of the design process from the functionality of the computer with predicted user activity via the designer and programmer, and changed it to center the user's needs and values and involving them in the design process before moving to computing functionality." This has been one of the most major developments in recent UID and is a buzzword in the field. Norman, who is associated with the creation of the term, was groundbreaking because he placed the user inside the HCI model, which is obviously in agreement with theories of semiotics which place the user inside the sign system like emergent meaning. Without the user, there can be no interface and no interface design in user centered



design. The only person who suggests the possibility of a message/interface independent of the user is Andersen.

In conclusion, the benefit of applying semiotics to UID is to have theories that explain existing principles of design we have discovered through cognitive science. By connecting and explaining them, semiotic theories are also able to predict instead of observe other potential principles of good UID.

## Conclusion

The overlap between semiotics and UID has always been apparent to me, but during my research for this paper I realized just how deeply they align and complement each other. This is best viewed through the connections between specific models and concepts; for example, how emergent meaning helps define what an interface is, or how Eco's TSP helps close Norman's gulfs. One of the most difficult concepts I grappled with is that semiotics won't give designers concrete do's and don'ts; this is the job of cognitive science and design research. Instead, semiotics is a perspective the designer can adopt to become more aware of the processes occurring during computer system use. Awareness of these processes and the relationships between designers, users, computers (hardware and software), user interfaces, and user interface languages changes the designer's high-level strategy.

The digital interface as media is important to study because while there are extensive semiotic theories of interpretation concerning literature, cultural anthropology, works of art, and other more traditional media, digital interfaces contain unique complexity. User interfaces are

“performing messages,” as de Souza says, and creating a user interface language requires the designer to create a deeper and more complex code than is common in non-computer based media (de Souza, 1993 p.52). For example, while the representation of the physical structure of a book may affect how users interact with it, its form remains consistent, but the computer system contains hundreds of inputs and outputs that change as users interact with different and ever-changing pages/windows/widgets. The designer often has more autonomy than the author/playwright/other senders of media as well because the designer must create an entire system of logic which needs to be conveyed and understood by the user in order to complete their goals, whereas in other forms of media it is less essential that the author convey their model because the artistic interpretation of the receiver fulfills the media’s purpose. To phrase it another way, the designer’s predictions of user interpretation are very central to the design process, even if they are not central to the meaning which emerges later.

Along these lines, one of the areas of greatest need for further research is how the designer can update computer systems based on their assessment of how well user models match designer models. Most research focuses on the interface as a one-shot message from the designer, but how can we look at the interface as a two-way conversation, where the users send information back to the designer? The ability to monitor and update the actual computer system the user interacts with as needed is one of the computer’s greatest advantages as media.

Another area in which further research is needed is how Peircean semiotics can be applied to UID. Some of the authors I analyzed attempted to do this, but none were completely successful, and I couldn’t find anyone who applied semiotics while completely rooted in a triadic model.

Further research could also compare and contrast applications of the broader semiosphere level and narrower sign complex level of semiotics to UID.

Another area of further thought, and research if possible, is how designers can become more aware of cultural code. This is of course not the “job to be done” (to use UX terminology) of semiotics, which only tells us how codes work, but semiotics supports principles of consistency that come up again and again and rely on the designer’s awareness of what is in or out of a user’s code. While there is certainly not a way to do this on an individual level, perhaps a larger emphasis on cultural anthropology and cognitive neuroscience within design curriculum could make designers more aware of codes. However, the most important question is still to ask if it is even useful to have a design principle that calls for using “existing codified expressions,” because in a nonstructural environment meaning can emerge from infinite pattern less codes (de Souza, 1993 p.764).

A principle more in line with contemporary semiotic thought is that a code is something a user possesses which varies so much between people that it can never be identified or generalized across a population. In this case, making the interface representation as customizable as possible is key so that the users can adjust it to their own codes. More research is needed into how this can take place, but this is one of the most intriguing and productive outcomes I have found from my analysis in this paper.

While each of these topics would be very productive to study for UID, the generation of any new research in combining semiotics and UID is urgent. Most research I found dates from the 1980s

and 90s, but computer systems and the environments for digital interfaces have evolved vastly in the last 30 years. The rise of artificial intelligence, machine learning, virtual reality, and an unforeseen level of connectedness and visibility via social media have changed the way we interact with and interpret digital interfaces (Jin, 2013). This could be especially interesting in the context of the dialogue partner perspective, which was deemed impossible in the 1990s but could very well be more possible in the 2020s.

In conclusion, adopting a semiotic perspective enables the user interface designer to have a much more advanced understanding of the interface they are designing. The most relevant semiotic concepts designers should start with are those I elaborated in my chapter 1: Peirce's nonstructuralism and sign complex model, Eco's theory of sign production, critique of iconicity, and theory of interpretation, Jakobson's speech act model, Bolinger's rejection of the sign as arbitrary, and Lotman's semiosphere. While more contemporary research is needed, Andersen, Nadin, and de Souza have given us an excellent foundation in the benefits of semiotics to UID. These include that semiotics gives us a proactive qualitative method of analysis, one which doesn't change with rapidly developing technology, that semiotics teaches us interfaces emerge uniquely each instance they are used and should be designed for customization, and that semiotics helps us think about how computer signs are expressed and related to each other in a generative hierarchy of complexity. This essay calls for user interface designers and semioticians to become more familiar with the other's respective field in order to generate innovative research and better interfaces.

## Bibliography

1. Andersen, P. B. (1990). *A theory of computer semiotics: Semiotic approaches to construction and assessment of computer systems*. Cambridge University Press.
2. Andersen, P. B. (1992). Computer Semiotics. *Scandinavian Journal of Information Systems*, 4.  
[http://aisel.aisnet.org/sjis/vol4/iss1/1?utm\\_source=aisel.aisnet.org%2Fsjis%2Fvol4%2Fiss1%2F1&utm\\_medium=PDF&utm\\_campaign=PDFCoverPages](http://aisel.aisnet.org/sjis/vol4/iss1/1?utm_source=aisel.aisnet.org%2Fsjis%2Fvol4%2Fiss1%2F1&utm_medium=PDF&utm_campaign=PDFCoverPages)
3. Andrews, E. (Ed.). (1990). *Markedness Theory*. Duke University Press.  
<https://doi.org/10.1215/9780822382881>
4. Andrews, E. (1994). The Interface of Iconicity and Interpretants. In M. Shapiro (Ed.), *Peirce Seminar Papers* (Vol. 2).
5. Andrews, E. (2006). The Role of Semiotics in the Study of Language, Linguistics and Communication: An Overview. *Glossos*, 8.
6. Andrews, E., Bierman, H., Hannon, B., & Ling, H. (2023). *Semiosis and Embodied Cognition: The Relevance of Peircean Semiotics to Cognitive Neuroscience*.
7. Apple Developer. (2023). *Human Interface Guidelines*.  
<https://developer.apple.com/design/human-interface-guidelines/guidelines/overview/>
8. Aristotle. (1966). *The Basic Works of Aristotle* (A. J. Jenkinson, Trans.). Random House.
9. Baecker, R. M. (Ed.). (1995). *Readings in human-computer interaction: Toward the year 2000* (2nd ed). Morgan Kaufmann Publishers.
10. Barthes, R. (1957). Myth Today. In *Mythologies*. Les Lettres nouvelles.
11. Barthes, R. (1964). *Elements of Semiology* (A. Lavers & C. Smith, Trans.). Hill and Wang.

12. Bianchi, C., & Vassallo, C. (2015). *Introduction: Umberto Eco's interpretative semiotics: Interpretation, encyclopedia, translation*. 2015(206), 5–11. <https://doi.org/10.1515/sem-2015-0017>
13. Blair-Early, A., & Zender, M. (2008). User Interface Design Principles for Interaction Design. *Design Issues: Interaction Design Research in Human-Computer Interaction*, 24(3), 85–107. JSTOR.
14. Bolinger, D. (1946). Visual Morphemes. *Language*, 22.
15. Bolinger, D. (1949). The Sign is Not Arbitrary. In *Boletin del Instituto Caro y Cuervo*. Instituto Caro y Cuervo.
16. Bolinger, D. (1965). *Forms of English*. Harvard University Press.
17. Bolinger, D. (1973). Truth is a Linguistic Question. *Language*, 49.
18. Bolinger, D. (1983). Intonation and Gesture. *American Speech*, 58.
19. Bowman, J. (2021). *The Best UX Research Methods*. <https://uxtools.co/blog/best-ux-research-methods-in-a-pinch/>
20. Bucholtz, M., & Hall, K. (2011). *CHAPTER 16 Language and Identity*.
21. Carroll, J. (Ed.). (1991). *Designing Interaction: Psychology at the Human Computer interface*. Cambridge University Press.
22. Carroll, J. M. (Ed.). (n.d.). *HCI models, theories, and frameworks: Toward a multidisciplinary science*. Morgan Kaufmann, c2003. <https://ebookcentral.proquest.com/lib/duke/reader.action?docID=294610>
23. Cuccio, V., & Gallese, V. (2018). A Peircian account of concepts: Grounding abstraction in phylogeny through a comparative neuroscientific perspective. *Royal Society Publishing*.

24. Dal Yong, J. (n.d.). Media history in the age of globalization. *Globalization and Media in the Digital Platform Age*.
25. Daylight, R. (2017). Saussure and the model of communication. *Semiotica*, 2017, 173–194.
26. de Saussure, F. (1916). *Course in general linguistics*. Duckworth.
27. la Saussure, F. (1986). *Course in General Linguistics*. Open Court.
28. de Souza, C. (1993). The semiotic engineering of user interface languages. *International Journal of Man-Machine Studies*, 39(5). <https://doi.org/10.1006/imms.1993.1082>
29. Deely, J. (1990). *Basics of Semiotics* (T. Sebeok, Ed.). Indiana University Press.
30. Di Piazza, S., & Piazza, F. (2016). The words of conjecture. Semiotics and epistemology in ancient medicine and rhetoric. *Rhetorica: A Journal of the History of Rhetoric*, 34(1), 1–26.
31. Eco, U. (1968). *La struttura assente. Introduzione alla ricerca semiologica*. Bompiani.
32. Eco, U. (1976). *A Theory of Semiotics*. Indiana University Press.
33. Eco, U. (1984). *The Role of the Reader*. Indiana University Press.
34. Eco, U. (1990). Introduction to Universe of the Mind. In *Universe of the Mind*. Indiana University Press. [https://books.google.com/books?id=0-xSQHU1YNYC&pg=PR5&source=gbs\\_selected\\_pages&cad=3#v=onepage&q&f=false](https://books.google.com/books?id=0-xSQHU1YNYC&pg=PR5&source=gbs_selected_pages&cad=3#v=onepage&q&f=false)
35. Eco, U. (1992). *Interpretation and Overinterpretation*. Cambridge University Press.
36. Experience, W. L. in R.-B. U. (Director). (n.d.). *Ethnography in UX (Video)*. Retrieved January 29, 2023, from <https://www.nngroup.com/videos/ethnography-ux/>
37. Gallese, V., & Lakoff, G. (2005). The Brain's concepts: The role of the Sensory-motor system in conceptual knowledge. *Cognitive Neuropsychology*.

38. Goguen, J., & Malcolm, G. (1998). *Signs and Representations: Semiotics for User Interface Design*. <https://cseweb.ucsd.edu/~goguen/pps/sigreps.ps>
39. Grudin, J. (1990). The Computer Reaches out: The Historical Continuity of Interface Design. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 261–268. <https://doi.org/10.1145/97243.97284>
40. Hartshorne, C., Weiss, P., & Burks, A. (Eds.). (1931). *Collected Papers of Charles Sanders Peirce*. Harvard University Press.
41. Hartson, H. R., & Pyla, P. S. (2012). *The UX Book: Process and guidelines for ensuring a quality user experience*. Elsevier. [https://learning.oreilly.com/library/view/the-ux-book/9780123852410/xhtml/Title\\_page.html](https://learning.oreilly.com/library/view/the-ux-book/9780123852410/xhtml/Title_page.html)
42. Hawkes, T. (1977). *Structuralism & semiotics*. University of California Press.
43. Hoffmeyer, J., & Kull, K. (2011). Chapter 14: Theories of Signs and Meaning: Views from Copenhagen and Tartu. In *Towards a Semiotic Biology: Life is the Action of Signs*. Imperial College Press.
44. IDEO (Ed.). (2015). *The field guide to human-centered design: Design kit* (1st. ed). Design Kit.
45. *IDEO Design Thinking*. (n.d.). IDEO | Design Thinking. Retrieved January 29, 2023, from <https://designthinking.ideo.com/>
46. Jakobson, R. (1959). On Linguistic Aspects of Translation. In *On Translation*. Harvard University Press.
47. Jakobson, R. (1971). On Linguistic Aspects of Translation. In *Volume II Word and Language* (pp. 260-266). Berlin, New York: De Gruyter Mouton. <https://doi.org/10.1515/9783110873269.260>



48. Jakobson, R. (1960a). Linguistics and Poetics. In *Roman Jakobson Selected Writings: Volume III Poetry of Grammar and Grammar of Poetry*. de Gruyter Mouton.
49. Jakobson, R. (1960b). The Dominant. In *Roman Jakobson Selected Writings: Volume III Poetry of Grammar and Grammar of Poetry*. de Gruyter Mouton.
50. Jakobson, R. (1980). *The Framework of Language*. Michigan Studies in the Humanities.
51. Jakobson, R. (1995). *On Language*. Harvard University Press.
52. Jendt, F. (2020). Values and identity: Dominant US cultural patterns using value orientation theory. *An Introduction to Intercultural Communication*.
53. Jørgensen, A. H. (2007). Mainstream versus Mainstream: Two Approaches to User Interface History. *Seventh Danish HCI Research Symposium*, 33.
54. Kammersgaard, J. (1988). Four different perspectives on human-computer interaction. *Journal of Man-Machine Studies*, 28, 343–362.
55. Krug, S. (2006). *Don't make me think! A common sense approach to Web usability* (2nd ed). New Riders Pub. <https://learning.oreilly.com/library/view/dont-make-me/0321344758/ch07.html>
56. Kurosu, M., & Kashimura, K. (1995). Apparent usability vs. inherent usability: Experimental analysis on the determinants of the apparent usability. *Conference Companion on Human Factors in Computing Systems - CHI '95*, 292–293. <https://doi.org/10.1145/223355.223680>
57. Locke, J. (1689). *An Essay Concerning Human Understanding* (Vol. 3).
58. Locke, J. (2017). *An Essay Concerning Human Understanding Book IV: Knowledge* (J. Bennett, Ed.).

59. Long, J. (2019). Semiotic Study of English Text. *Advances in Social Science, Education and Humanities Research*, 342.
- <file:///Users/tonyturnercarroll/Downloads/125916396.pdf>
60. Lotman, J. (1967). Exact Methods in Russian Literary Science. *Strumenti Critici*.
61. Lotman, J. (1988). Text within a Text. *Soviet Psychology*.
62. Lotman, J. (1990). *Universe of the Mind*. Indiana University Press.
63. Lotman, J. (2009). *Culture and Explosion*. de Gruyter Mouton.
64. Lotman, J., & Clark, W. (2005). On the semiosphere. *Sign Systems Studies*, 33(1), 205–226.
65. Makaryk, I. (1993). Semiotic poetics of the Prague School (Prague School). In *Encyclopedia of Contemporary Literary Theory: Approaches, Scholars, Terms*. University of Toronto Press.
66. Myers, B. (1998). A brief history of human-computer interaction technology. *Association for Computing Machinery: Interactions*, 5(2), 44–54.
- <https://doi.org/10.1145/274430.274436>
67. Nadin, M. (1988). Interface design: A semiotic paradigm. *Semiotica*, 69.
68. Nadin, M. (2023). *Semiotics for the HCI Community*. <https://nadin.ws/hci/>
69. Nielson, J. (1994). 10 Usability Heuristics for User Interface Design. *Nielson Norman Group*. <https://www.nngroup.com/articles/ten-usability-heuristics/>
70. Nielson, J. (2012). *Usability 101: Introduction to Usability*. Nielson Norman Group. <https://www.nngroup.com/articles/usability-101-introduction-to-usability/>
71. Nielson, J. (Director). (2019). *The Immutable Rules of UX (Jakob Nielsen Keynote)*. <https://www.youtube.com/watch?v=OtBeg5eyEHU>

72. Norman, D. (1986). Cognitive Engineering. In *User Centered Systems Design*. Lawrence Erlbaum Associates.
73. Norman, D. (1988). *The Design of Everyday Things*. Perseus Books Group.
74. Norman, D. (2004). *Emotional Design: Why We Love (or Hate) Everyday Things*. Perseus Books Group.
75. Norman, D. (Director). (2019). *Principles of Human-Centered Design*.  
<https://www.youtube.com/watch?v=rmM0kRf8Dbk>
76. Norman, D., & Draper, S. (Eds.). (1986). *User Centered Systems Design*. Lawrence Erlbaum Associates.
77. Peirce, C. S. (1931). *The Collected Papers of Charles Sanders Peirce* (Vol. 1–1–8). Harvard University Press.
78. Peirce, C. S. (1957). *Essays in the Philosophy of Science* (V. Tomas, Ed.). The Liberal Arts Press.
79. Piller, I. (2017). Nation and culture. *Intercultural Communication: A Critical Introduction*.
80. Polidoro, P. (2015). Umberto Eco and the problem of iconism. *Semiotica*, 129–160.
81. Salupere, S., Torop, P., & Kull, K. (Eds.). (2013). *BEGINNINGS OF THE SEMIOTICS OF CULTURE*. University of Tartu Press.
82. Savan, D. (1976). *An Introduction to C. S. Peirce's Semiotics: V. I*. Victoria University.  
<https://books.google.com/books?id=a41FNQAACAAJ>
83. Savan, D. (1980a). Abduction and Semiotics. In I. Rauch & G. Carr (Eds.), *The Signifying Animal*. Indiana University Press.

84. Savan, D. (1980b). Language: At the Crossroads of Linguistics and Semiotics. In I. Rauch & G. Carr (Eds.), *The Signifying Animal*. Indiana University Press.
85. Scott, R. (2020). *What are Best Practices in User Experience Design?*  
<https://www.uxconnections.com/what-are-best-practices-in-user-experience-design/>
86. Sebeok, T. (1977). *How Animals Communicate*. Indiana University Press.
87. Sebeok, T. (1991). *A Sign is Just a Sign*. Indiana University Press.
88. Shapiro, M. (1985). THE SENSE OF GRAMMAR: LANGUAGE AS SEMEIOTIC.  
*Studies in Second Language Acquisition*, 7(3), 375–377. JSTOR.
89. Sharritt, M. (2008). *HCI Models, Theories and Frameworks: Toward a Multidisciplinary Science (Summary)*. <https://www.sharritt.com/CISHCIExam/carroll.html>
90. Srinivasan, R. (2017). Digital stories from the developing world. *Whose Global Village: Rethinking How Technology Shapes Our World*.
91. The Interaction Design Foundation. (2023). *Human Computer Interaction*.  
[https://www.interaction-design.org/literature/book/the-encyclopedia-of-human-computer-interaction-2nd-ed/human-computer-interaction-brief-intro#:~:text=Human%2Dcomputer%20interaction%20\(HCI\),science%20and%20human%20factors%20engineering.](https://www.interaction-design.org/literature/book/the-encyclopedia-of-human-computer-interaction-2nd-ed/human-computer-interaction-brief-intro#:~:text=Human%2Dcomputer%20interaction%20(HCI),science%20and%20human%20factors%20engineering.)
92. Tribus, A. (2017). The Communicative Functions of Language: An Exploration of Roman Jakobson's Theory in TESOL. *SIT Digital Collections*, 4.
93. Usability.gov. (2023). *User Interface Design Basics*. <https://www.usability.gov/what-and-why/user-interface-design.html>
94. von Uexküll, J. (1982). The Theory of Meaning. *Semiotica*, 42.

95. Vygotsky, L. S. (1987). *The Collected Works of L.S. Vygotsky* (R. Rieber, Ed.). Plenum Press.