Viewing the world systemically.

# *ATIS* Theory Development

Prepared by: Kenneth R. Thompson
Head Researcher
*System-Predictive Technologies*

# *ATIS* Theory Development

## *ATIS* Properties

### Basic Properties

The ***Basic Properties*** define the attributes required for *General Systems Theory*. They are basic to the concept of a *General System*, $\mathcal{G}$. The *Basic Properties* include the *Component Set*, $\mathcal{G}_O$, the *Affect-Relation Set*, $\mathcal{GA}$, and the *Affect-Relation Qualifiers Set*, $\mathcal{Q}$.

The first property, group, defines the (*component*-) *object-set*, $\mathcal{G}_O$, of a system.

**Group, $\mathcal{G}_O$,** $=_{df}$ A set with at least two components within the universe of discourse.

$$\mathcal{G}_O =_{df} \{x \mid x \in \mathfrak{W} \in \mathcal{U}\} \wedge |\mathfrak{W}| > 1$$

In this definition, '$\mathcal{U}$' is the universe of discourse, '$\mathfrak{W}$' is an object-set, and $|\mathfrak{W}|$ is the set-cardinality function.

As the initial intent of this research is to be able to analyze *complex intentional systems* with a multitude of elements, various types of elements, and numerous types of connectedness, an effective process must be established to identify those elements, the elements of $\mathcal{G}_O$. Further, as a General System will be analyzed as a topology, the topological relation-set will be introduced.

Although 'General System' has not yet been defined, 'group' is defined in anticipation of its future use as the General System Object Set, $\mathcal{G}_O$. Although, at this point, it is nothing more than a "group," its construction is defined so as to be applicable to a *General System*.

In order to obtain precise property and affect relations' definitions, the object-set must be precisely defined. The *General System Object-Set, $G_O$, Construction Decision Procedure* is defined below.

## General System Object-Set, $\mathcal{G}_0$, Construction Decision Procedure

The logical construction of the *General System Object-Set*, $\mathcal{G}_0$, will be determined as follows:

1) Every *Information Base* ($\bar{I}_B$) defines affect relations, $\mathcal{A}_n \in \mathcal{A}$, by the unary- and binary-component-derived sets from the $\bar{I}_B$. That is, the components of $\mathcal{A}_n$ are of the form: $\{\{x_i\},\{x_i,y_i\}\} \in \mathcal{A}_i \in \mathcal{A}_n$ that indicates that an "affect relation" has been empirically determined to exist from "$x_i$" to "$y_i$."

2) However, in order to even determine the affect relations, the qualifiers that specify the affect relations must be established. Very simply, what is the nature of the system being considered? To define the system, the affect relations must be known, and those are defined by the system qualifiers, the predicates that define the affect relations; and, therefore, the components of the system.

   The *Affect Relation Qualifiers Set*, $\mathcal{Q}$ must be defined before any affect relations, and, therefore, any components can even be recognized. These are the predicates that define which affect relations will be considered for system inclusion.

3) The following functions, $\mu$ and $\beta$, define elements of a topology, $\tau_n$, that will allow for analysis of an affect relation. That is, $\mu,\beta:\mathcal{A}_n \rightarrow \tau_n$, such that:

   $$\mu\mathcal{A}_i = \{x_i\} \in \tau_n; \text{ and } \beta\mathcal{A}_i = \{x_i,y_i\} \in \tau_n.$$

   An additional function, $\varphi$, will also be required for certain properties, and will allow for specification of specific elements, as follows:

   $$\varphi\beta\mathcal{A}_i = y_i.$$

   Hence, the elements of $\mathcal{G}_0$ can be specified by $\varphi$ and $\mu \cap \beta$.

4) The set of initial elements of $\mathcal{G}_0$ will be defined by an existing $\bar{I}_B$ as follows:

   $$\mathcal{G}_0 = \{x \mid \exists i(x \in (\mu\mathcal{A}_i \cap \beta\mathcal{A}_i) \wedge \mathcal{A}_i \in \mathcal{A}_n\}.$$

5) New elements will be added to $\mathcal{G}_0$ by Rule 3) when the new element establishes an affect-connected relation with an existing element in $\mathcal{G}_0$ so that it is an element of an $\mathcal{A}_i \in \mathcal{A}_n$.

6) No other objects will be considered as elements of $\mathcal{G}_0$ except as they are generated in accordance with Rules 1) to 4).

Now that the object-set has been determined, the concept of system must be established.

# System

There are various definitions of 'system' in the literature.[1] A Mesarović system is frequently used and it relates to the traditional concept of what a system "should" be; that is, it consists of related components. In this definition, a system is a relation on non-empty sets:

$$S \subset \prod\{V_i : i \in I\};\text{ where 'I' is an index set.}$$

Lin extends the Mesarović definition so that multiple relations with a varying number of variables may be defined without having to change the object set, and defines a 'system', A, more conventionally as an ordered pair consisting of an object set, M, and a relation set, F:

$$A = (M,F).$$

Steiner and Maccia followed this convention and defined 'system' as follows:

**System, $\mathbb{S}$, $=_{df}$** A *group* with at least one affect relation that has information.

$$\mathbb{S} =_{df} (\mathbf{S}, \mathfrak{R}) = (\mathbb{S}_O, \mathbb{S}_\phi);\text{ where } \mathbf{S} = \mathbb{S}_O \text{ and } \mathfrak{R} = \mathbb{S}_\phi.$$

A **system** is an ordered pair defined by an *object-set*, **S** or $\mathbb{S}_O$, and a *relation-set*, $\mathfrak{R}$ or $\mathbb{S}_\phi$.

In this research, the definition of system will be extended to more adequately account for all system parameters. This extension will more clearly define the topology and/or relatedness of a system by its object-sets and relation-sets; as well as allow for a more rigorous and comprehensive development of the system logic required for a logical analysis utilizing the Predicate Calculus and other required logics.

A *General System* is defined within a ***Universe of Discourse***, $\mathcal{U}$, that includes the system and its environment. The only thing that demarcates the systems under consideration is the "*Universe of Discourse*." And, while that universe may be somewhat *fuzzy* or *rough*, whatever systems are being considered will be well defined. In the case of *Education Systems*, the boundary of the universe may be quite fluid, or possibly unknown, especially with respect to the object-sets.

[1] See *System*.

$\mathcal{U}$ is partitioned into two disjoint systems, $\mathbb{S}$ and $\mathbb{S}'$. Therefore, ***Universe of Discourse*** has the following property:

$$\mathcal{U} = \mathbb{S} \cup \mathbb{S}'; \text{ such that, } \mathbb{S} \cap \mathbb{S}' = \varnothing.$$

The disjoint systems of $\mathcal{U}$, $\mathbb{S}$ and $\mathbb{S}'$, are defined as "system" and "negasystem," respectively.

*System environment* and *negasystem environment* are defined as follows:

**System environment, $\mathbb{S}'$,** $=_{df}$ The system's corresponding negasystem, $\mathbb{S}'$.

**Negasystem environment, $\mathbb{S}$,** $=_{df}$ The negasystem's corresponding system, $\mathbb{S}$.

# General System

A ***General System***, $\mathcal{G}$, is defined by the following parameters:

1) *Family of Affect Relations Set*, $\mathcal{A}$;
2) *Affect Relation Qualifiers Set*, $\mathcal{Q}$
3) *Component Partitioning Set*, $\mathcal{P}$;
4) *Transition Function Set*, $\mathcal{T}$;
5) *Linearly Ordered Time Set*, $\mathscr{T}$; and
6) *System State-Transition Function*, $\sigma$.

That is:

> ***General System*** ($\mathcal{G}$) $=_{df}$ a set of affect-relations ($\mathcal{A}$) defined by affect-relation-qualifiers ($\mathcal{Q}$), which determine a set of partitioned components ($\mathcal{P}$), a transition functions set ($\mathcal{T}$), a linearly-ordered time set ($\mathscr{T}$), and a state-transition function ($\sigma$). Therefore:
>
> $$\mathcal{G} =_{df} (\mathcal{A}, \mathcal{Q}, \mathcal{P}, \mathcal{T}, \mathscr{T}, \sigma)$$

This definition is more accurately defined as follows:

$$\mathcal{G} =_{df} [\mathcal{A}|\mathcal{Q} \Vdash \mathcal{P}(\mathcal{T}, \mathscr{T}, \sigma)];$$[2]

[2] '⊩' is read "determines" or "which determine" or "from which is/are derived", as appropriate for the sentence in which it is used. This symbol is similar in intent to the logical "yields", but whereas "yields" is a logical relation for a deductive proof, this is a predicate relation identifying that which is derived from the existent set. This definition is used as it emphasizes the fact that no system can be recognized without first knowing the affect-relations as defined by the qualifying predicates. If no affect relation is recognizable, then no components can even be found.

That is, ***General System***, $\mathcal{G}$, is defined as the *Affect-Relations Set*, $\mathcal{A}$, given the *Affect-Relation Qualifier Set*, $\mathcal{Q}$, which determine the *Component Partitioning Set*, $\mathcal{P}$, explicated by the *Transition Functions Set*, $\mathcal{T}$, the *Linearly-Ordered Time Set*, $\mathscr{T}$, and the *State-Transition Function*, σ.

The sets that define $\mathcal{G}$ have the following elements:

$\mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_n \in \mathcal{A}$; and

$T_\mathcal{P}, I_\mathcal{P}, F_\mathcal{P}, O_\mathcal{P}, S_\mathcal{P}, \mathbb{S}_{B0}, \mathbb{S}'_{B0} \in \mathcal{P}$;

$\mathscr{L}, \mathscr{L}' \in \mathcal{Q}$;

$f_I, f_O, f_T, f_B, f_S, f_N, f_E \in \mathcal{T}$;

$t_1, t_2, \ldots, t_k \in \mathscr{T}$.

Let the object-set of a General System, $\mathcal{G}_0$, be such that $\mathcal{G}_0 = \mathbb{S}_0 \cup \mathbb{S}'_0$; where $\mathbb{S}_0$ and $\mathbb{S}'_0$ are the object-sets of $\mathbb{S}$ and $\mathbb{S}'$, respectively. Then, $\mathcal{G}_0$ is defined by the following:

$$\mathcal{G}_0 =_{df} \mathbb{S}_0 \cup \mathbb{S}'_0 = (I_\mathcal{P} \cup F_\mathcal{P} \cup S_\mathcal{P} \cup \mathbb{S}_{B0}) \cup (T_\mathcal{P} \cup O_\mathcal{P} \cup \mathbb{S}'_{B0})$$

Further, as all of these sets are disjoint, the following holds:

$$I_\mathcal{P} \cap F_\mathcal{P} \cap S_\mathcal{P} \cap \mathbb{S}_{B0} \cap T_\mathcal{P} \cap O_\mathcal{P} \cap \mathbb{S}'_{B0} = \varnothing.$$

$T_\mathcal{P}, I_\mathcal{P}, F_\mathcal{P}, O_\mathcal{P}, S_\mathcal{P}, \mathscr{L}, \mathscr{L}'$, $\mathbb{S}_{B0}$, and $\mathbb{S}'_{B0}$ represent the following sets:

'$T_P$' represents "topput."

'$I_\mathcal{P}$' represents "input."

'$F_P$' represents "fromput."

'$O_P$' represents "output."

'$S_P$' represents "storeput."

'$\mathscr{L}$' represents "system logisticians" or "system qualifiers."

'$\mathscr{L}'$' represents "negasystem logisticians" or "negasystem qualifiers."

'$\mathbb{S}_{B0}$' represents "system background components," known or unknown.

'$\mathbb{S}'_{B0}$' represents "negasystem background components," known or unknown.

The system background components and negasystem background components are defined in terms of the "population"; however, such background components can in fact be any type of component that is necessary to properly analyze the system.

In view of the foregoing, the system object-set, $\mathfrak{S}_O$, and negasystem object-set, $\mathfrak{S}'_O$, are defined as follows:

$$\mathfrak{S}_O =_{df} I_{\mathcal{P}} \cup F_{\mathcal{P}} \cup S_{\mathcal{P}} \cup \mathscr{L} \cup \mathfrak{S}_{BO}; \text{ and}$$

$$\mathfrak{S}'_O =_{df} T_{\mathcal{P}} \cup O_{\mathcal{P}} \cup \mathscr{L}' \cup \mathfrak{S}'_{BO}.$$

**<u>Corollary</u>:**

$$\mathfrak{S}_{BO} = \mathfrak{S}_O \setminus (I_{\mathcal{P}} \cup F_P \cup S_P); \text{ and } \mathfrak{S}'_{BO} = \mathfrak{S}'_O \setminus (T_P \cup O_P).$$

*Background Components* may arise when the object-set is *fuzzy* or *rough* (see fuzzy set theory or rough set theory); that is, not all components are specifically known, but it is known that such components exist. For example, you may know that there are over 10,000 people in a particular town, but you do not know who they all are.

Now that the object-sets have been defined, the relation-sets must be defined.

*Transition functions* give the system dynamics. These are the functions that are operated on by the *System State-Transition Function*, σ, so as to change the system structure and thereby the "behavior" or "intention" of the system.

*System behavior* is defined as a sequence of *system states*.

A consistent pattern of system states defines *System Dispositional Behavior*.

The transition functions required for state-transition analysis are described as follows: $\oint_B$, $\oint_E$, $\oint_F$, $\oint_I$, $\oint_N$, $\oint_O$, $\oint_S$, $\oint_T$ are the transition function-sets and represent the following functions:

'$\oint_B$' is "feedback."
'$\oint_E$' is "feedenviron."
'$\oint_F$' is "feedfrom."
'$\oint_I$' is "feedin."
'$\oint_N$' is "feedintra."
'$\oint_O$' is "feedout."
'$\oint_S$' is "feedstore."
'$\oint_T$' is "feedthrough."

# Affect Relations

*Affect relations* determine the structure of the system by the connectedness of the components. $\mathcal{A}_1$, $\mathcal{A}_2$, …, $\mathcal{A}_n$ are the affect relation-sets of $\mathcal{G}$. These sets are elements of the family of affect relations, $\mathcal{A}$. These sets define each subsystem of $\mathcal{G}$. For example, a *T/I-put interface system* will be defined as: T/I $=_{df}$ $T_{\mathcal{P}} \cup I_{\mathcal{P}}$, and is defined by the affect relations that define the feedin function, $f_I$, that results in the input resulting from a *System State-Transition* of toput into the system, $\mathfrak{S}$. For example, this subsystem may have three affect relations, $\mathcal{A}_1$, $\mathcal{A}_2$, and $\mathcal{A}_3$, that will generate the transition functions, $f_I$. That is:

$$f_{I(1)} \subset \mathcal{A}_1,\ f_{I(2)} \subset \mathcal{A}_2, \text{ and } f_{I(3)} \subset \mathcal{A}_3.$$

Then, the *System State-Transition Function*, $\sigma$, operating on the transition functions, $f$, "move" the qualified components from $\mathfrak{S}'$ to $\mathfrak{S}$ for each type of affect relation.

Steiner and Maccia define **affect relation** as follows:

**Affect relation, $\mathcal{A}$,** $=_{df}$

A connection of one or more components to one or more other components.

$$\mathcal{A} =_{df} \{\{\{x\},\{x,y\}\} \mid P(x,y)|\mathcal{Q} \wedge x,y \in \mathfrak{X} \subset \mathcal{G}_0 \vee [(x = \mathfrak{U} \subset \mathfrak{X} \subset \mathcal{G}_0 \wedge y = \mathfrak{V} \subset \mathfrak{Y} \subset \mathcal{G}_0)]\}$$

**Affect relations** define the connectedness of the system.[3]

In the current research, affect relation, as defined below, is a binary-relation of the form $\{\{x\},\{x,y\}\}$ as empirically derived from an $\bar{I}_B$ (information base). If the direction of the affect relation is unknown, then both $\{\{x\},\{x,y\}\}$ and $\{\{y\},\{x,y\}\}$ will be included in the affect relation set.

---

[3] To be accurate, any predicate, P, should be defined as being derived from the affect-relation-qualifier set, $\mathcal{Q}$; however, such will be assumed unless stated otherwise. That is, $P(x,y)|\mathcal{Q}$ (P(x,y) given $\mathcal{Q}$) will simply be written as P(x,y).

This definition of affect relation is comparable to a Mesarović system, which is consistent with the current development since each relation defines a Mesarović system. Further, Mesarović refers to such systems as "input-output" systems, where

$$X = \prod \{V_i | i \in I_X\}, \text{ the "inputs"}; Y = \prod \{V_i | i \in I_Y\}, \text{ the "outputs"; and}$$

$\{I_X, I_Y\}$ is a partition of the index set, I. Since $X \cap Y = \varnothing$, the partition condition is satisfied. Now, this definition can be written to look very similar to that intended by Steiner and Maccia; that is:

$$\mathcal{A} \subset \mathfrak{X} \times \mathfrak{Y} = \{(\mathfrak{x},\mathfrak{y}) | \mathfrak{x} \in \mathfrak{X} \wedge \mathfrak{y} \in \mathfrak{Y}\}$$

And, from this, the family of affect relations can be obtained, such that: $\forall n(\mathcal{A}_n \in \mathcal{A})$.

As with the object-set, an effective procedure must be established for determining the elements of the affect relations. The *Affect Relation-Set, $\mathcal{G}_{\mathcal{A}}$, Construction Decision Procedure* is such an effective procedure and is given below.

**Affect Relation-Set, $\mathcal{G}_{\mathcal{A}}$, Construction Decision Procedure**

The logical construction of the affect relation-set, $\mathcal{G}_{\mathcal{A}}$, will be determined as follows:

1) *Affect Relation-Set Predicate Schemas*, $P_n(x_n, y_n)|\mathcal{Q}_n = P_n(\mathcal{A}_n)$, are defined as required to empirically define the family of affect-relations, $\mathcal{A}_n \in \mathcal{A}$, as extensions of the predicate schemas. The elements of $\mathcal{A}_n$ are of the form $\{\{x\},\{x,y\}\}$ that indicates that an "affect relation" has been empirically determined to exist from "$x$" to "$y$." '$P_n(\mathcal{A}_n)$' designates the predicate that defines the elements of $\mathcal{A}_n$ as derived from the predicate-qualifier $\mathcal{Q}_n$.

2) The *Affect-Relation Transition Function*, $\phi_n$, is defined by:

$$\phi_n: \mathfrak{X} \times \mathfrak{Y} \to \mathcal{A}_n \mid \mathfrak{X}, \mathfrak{Y} \subset \bar{I}_B \;.\wedge.\; \phi_n(\mathfrak{X} \times \mathfrak{Y}) = \{\{\{x_n\},\{x_n, y_n\}\} \mid P_n(\mathcal{A}_n) \wedge x_n \in \mathfrak{X} \wedge y_n \in \mathfrak{Y}\}.$$

3) The family of affect relations, $\mathcal{A} = \mathcal{G}_{\mathcal{A}}$, is defined recursively by applications of the function in 2) for all elements in $\bar{I}_B$ to each $P_n(\mathcal{A}_n)$ defined in 1).

4) New components are evaluated for each $P_n(\mathcal{A}_n)$ defined in 1) and included in the appropriate extension when the value is "true".

5) No other objects will be considered as elements of $\mathcal{A}_n \in \mathcal{A} = \mathcal{G}_{\mathcal{A}}$ except as they are generated in accordance with rules 1) through 4).

By convention, $\{\{x\},\{x,y\}\} \equiv (x,y)^{\rightarrow} \equiv (x,y)$, where the latter can be used if there is no confusion concerning direction of the relation.

# Transition Functions

The transition functions will now be defined in a manner to allow for temporal analysis of the system.

**Feed-Function Schema.** The "feed-" functions, $\oint_F$; that is, $\oint_I$, $\oint_O$, $\oint_T$, $\oint_B$, $\oint_S$, $\oint_N$, and $\oint_E$, are defined as follows:

$$\oint_F: X_{\mathcal{P}} \rightarrow Y_{\mathcal{P}} \mid \oint_F(\mathrm{x}) = \mathrm{y}.$$

$X_{\mathcal{P}}$ and $Y_{\mathcal{P}}$ are the corresponding "-put" sets defined for each function. For example, $\oint_I: T_{\mathcal{P}} \rightarrow I_{\mathcal{P}} \mid \oint_I(\mathrm{x}) = \mathrm{y}$ defines the movement of toput components to input components.

**-Put Set Schema.** For all of the "–put" sets, $\mathcal{P}$; that is, $T_{\mathcal{P}}$, $I_{\mathcal{P}}$, $F_{\mathcal{P}}$, $O_{\mathcal{P}}$, and $S_{\mathcal{P}}$, a **time function**, $f(\mathrm{x})_{\oint F(t)}$, is defined from the product set of a "-put" set, $\oint_F$, and a time set into the real numbers, $\Re$.

$$f(\mathrm{x})_{\oint F(t)}: \oint_F \times \mathcal{T} \rightarrow \Re = \mathcal{A}$$

For example, $f_{T_{\mathcal{P}}(t)}: T_{\mathcal{P}} \times \mathcal{T} \rightarrow \Re = \mathcal{A}$; that is, $\mathcal{A}$ is the APT-value of $T_{\mathcal{P}}$ at time t.

To determine the temporal transitions of components, an APT-Analysis is performed with respect to the components of an affect relation such that: $\oint_F(\mathrm{x}) = \mathcal{A}$; where $\mathcal{A}$ is the APT value. When $\mathcal{A}$ is greater-than or equal-to a predetermined value, or is 0, then component $\mathrm{x}$ has "moved" to the target set as $\mathrm{y}$; that is, $\oint_F(\mathrm{x}) = \mathrm{y}$. That is:

$$\exists \mathcal{A} \left[\oint_F(\mathrm{x} \in X_{\mathcal{P}}) = \mathcal{A} \mid \mathcal{A} = 0 \vee \mathcal{A} > \alpha \supset \oint_F(\mathrm{x}) = \mathrm{y} \in Y_{\mathcal{P}}\right]$$

**State-Transition Function Schema.** Then the **state-transition function**, $\sigma$, is defined by the following composition:

$$\sigma_x(\oint_F(\mathrm{x}) \circ f(\mathrm{x})_{\oint F(t)}) = \nu = 0 \supset x \in \oint_F(\mathrm{x} \in X_{\mathcal{P}}) = \mathrm{y} \in Y_{\mathcal{P}}.$$

# Descriptive Analysis of General Systems

The descriptive analysis of an empirical system will be accomplished by using an *APT Analysis* developed by Frick. Further, the direct approach taken by an *APT Analysis* makes it readily applicable to a computer-based analysis of an $\bar{\boldsymbol{I}}_B$. Frick describes the process as follows:

> Analysis of patterns in time (APT) is a method for gathering information about observable phenomena such that probabilities of temporal patterns of events can be estimated empirically. [With an appropriate analysis] temporal patterns can be predicted from APT results.
>
> The task of an observer who is creating an APT score [since renamed 'temporal map'][4] is to characterize simultaneously the state of each classification as events relevant to the classifications change over time.
>
> An APT score ['temporal map'] is an observational record. In APT, a score ['temporal map'] is the temporal configuration of observed events characterized by categories in classifications.
>
> [This contrasts significantly from the linear models approach (LMA) common to most research.] The worldview in the LMA is that we measure variables separately and then attempt to characterize their relationship with an appropriate mathematical model, where, in general, variable Y is some function of X. A mathematical equation is used to express the relation. In essence, the relation is modeled by a line surface, whether straight or curved, in *n*-dimensional space. When such linear relations exist among variables, then a mathematical equation with estimates of parameters is a very elegant and parsimonious way to express the relation.
>
> In APT, the view of a relation is quite different. First, a relation occurs in time. A relation is viewed as a set of temporal patterns, not as a line surface in *n*-dimensional space. A relation is measured in APT by simply counting occurrences of relevant temporal patterns and aggregating the durations of the patterns. This may seem rather simplistic to those accustomed to the LMA, but Kendall (1973) notes,
>
>> "Before proceeding to the more advanced methods, however, we may recall that in some cases forecasting can be successfully carried out merely by watching the phenomena of interest approach. Nor should we despise these simple-minded methods in the behavioral sciences."

---

[4] While temporal information obtained from observing a particular system was initially referred to as an APT 'score' (e.g., Frick, 1990), the nomenclature was later changed to 'temporal map' as MAPSAT was further developed. See, for example: https://www.indiana.edu/~tedfrick/MAPSATAECTOrlando2008.pdf, and http://educology.indiana.edu/affectRelationTemporal.html. This was necessary, since the most common meaning of 'score' is that of a number, such as the score in an athletic event or game. Frick was, however, using 'score' in the sense of a *musical* score that consisted of a temporal description of music by notation on staffs for musicians to follow, e.g., to play *Beethoven's 3rd Symphony*. Each musical 'score' is not a number; rather it is a configuration or map which is unique. Such a configuration is indexical and represents something unique; see for example: http://educology.indiana.edu/sign.html. Further, the musical map (score) for *Beethoven's 3rd Symphony* is different from his 9th or his 5th symphonies. In addition, the distinction in MAPSAT between temporal maps and structural maps is consistent with the discussion of dynamic and structural properties of a given system.

For this research, *APT Analysis* lends itself quite readily to establishing patterns that indicate new objects and relations that should be added to the system. System state will be defined by system properties. System properties will be defined by the connectedness of the system components; which defines the system structure.

The most direct way to define the structure required is by utilizing graph-theoretic properties. These properties will be defined as required for the further development of *ATIS*.

# Affect Relation Properties, $_{x}\mathcal{A}$

*Affect Relation Properties* will be defined in terms of path-connected elements, $_{pc}\boldsymbol{E}$. The properties are defined in set-theoretic terms so that they can be used to define a topology.

Therefore, before proceeding with the definitions of *Affect Relation Properties*, the relevant *Graph Theoretic Properties* will be presented.[5]

---

## Graph-Theoretic Connected Properties (Elements), $_{x}\boldsymbol{E}$

**Path-connected elements, $_{pc}\boldsymbol{E}$, =**$_{df}$

$$\{(x,y) | (x = x_0, x_1, x_2, \ldots x_{n-1}, x_n = y) \wedge \forall (x_i, y_i)_{i<n} [y_i = x_{i+1}]\}$$

**Path-connectedness** is intuitively defined as the ability to get from one element to another by following a sequence of elements. The connected paths are "channels," in terms of information theory, or "communications" between the elements of a system, or affect relations. These are graph-theoretic properties that will be used to define system properties.

**Discrete segment, $|(x,y)^{\rightarrow}_{n=1}| = 1$**, =$_{df}$ A path between two and only two elements.

$$|(x,y)^{\rightarrow}_{n=1}| = 1 \equiv \{(x,y) \mid (x = x_0, y = x_1)\}.$$

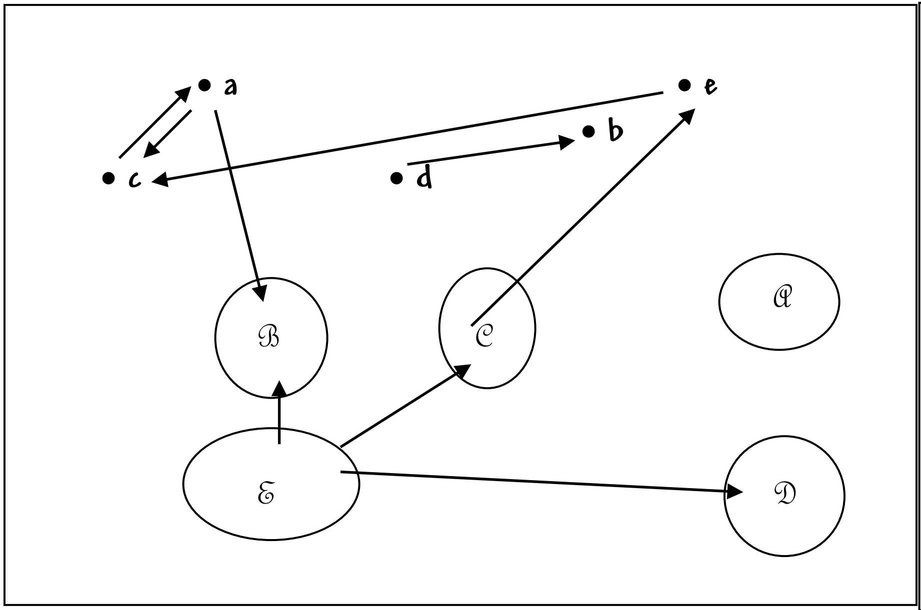
**Segment cardinality, $|(x,y)^{\rightarrow}_{n}| = n$**, =$_{df}$ The number of discrete segments between elements.

$$|(x,y)^{\rightarrow}_{n}| = n \equiv \{(x,y) \mid (x = x_0, y = x_n)\}.$$

The following graph depicts the path-connectedness of elements a, b, c, d, and e; and the path-connectedness of subsets, $\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$, $\mathcal{D}$, and $\mathcal{E}$.

---

[5] For a more thorough discussion of graph theory for *ATIS*, go to *ATIS Graph Theory*, and *ATIS: Connected Components and Affect Relations*.




The following component path-connections are depicted: (a,c), (c,a), (e,c), (e,a), (d,b).

The following subset path-connections are depicted: ($\mathcal{E}$,$\mathcal{B}$), ($\mathcal{E}$,$\mathcal{C}$), and ($\mathcal{E}$,$\mathcal{D}$).

The following component and subset path-connections are depicted: (a,$\mathcal{B}$), ($\mathcal{C}$,e), ($\mathcal{C}$,c), ($\mathcal{C}$,a), (e,$\mathcal{B}$), ($\mathcal{E}$,e), ($\mathcal{E}$,c), ($\mathcal{E}$,a).

The following diagram and symbol conventions will be used to clarify and define the graph-theoretic properties.

Arrows ($\rightarrow, \leftrightarrow, \leftarrow$) will be used to show direction of an affect relation between elements of a system.

‘(p,q)’ designates the **connected elements** p and q.

‘p $\rightarrow$ q designates the ordered pair **path-connected elements** from p to q.

The following diagram, in addition to helping to clarify the connectedness properties, will also be used to introduce terminology that is useful for describing connectedness.

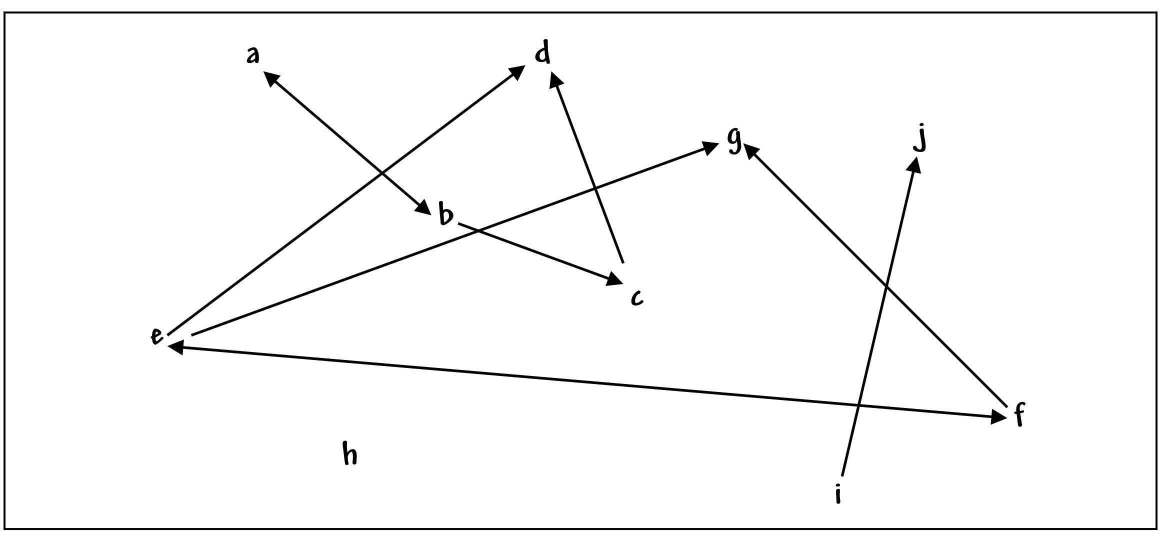



**Diagram of Directed Component Connectedness**

The following list is presented to facilitate the understanding of the various connectedness relationships. From the above graph, the following relations are determined:

**Path-connected elements**:

(a,b), (b,a), (a,c), (a,d), (b,c), (b,d), (c,d), (e,d), (e,f), (f,d), (f,e), (f,g), and (i,j).

**Path-connected elements with three segments**: (a,d).

**Completely connected elements**: (a,b) and (e,f).

**Unilaterally connected elements**: (a,c), (a,d), (b,c), (b,d), (e,d), (e,g), (f,g), and (i,j).

**Disconnected elements**: (a, h), (h, j), all h-pairs of elements, and all i and j pairs except for (i,j).

**Receiving elements**: a, b, c, d, e, f, g, and j.

**Initiating elements**: a, b, c, e, f, and i.

**Primary initiating elements**: i; that is, it initiates, but does not receive.

h may be considered as a trivial **primary initiating element**.

**Terminating elements**: d, g, and j.

h may be considered as a trivial **terminating element**.

All terminating elements must be **unilaterally terminating elements**.

**Connected** but not **path-connected elements**: (a,e), (a,f), (a,g), (b,e), (b,f), (b,g), (c,e), (c,f), and (c,g).

The terms described above will be formally defined below. **Path-connected elements** will be restated so as to bring all of the graph-theoretic properties together in one listing.

**Path-connected elements,** $_{pc}E$, $=_{\text{df}} \{(x,y) \mid (x = x_0, x_1, x_2, \ldots x_{n-1}, x_n = y) \wedge \forall (x_i,y_i)_{i<n}[y_i = x_{i+1}]\}$

**Completely connected elements,** $_{cc}E$, $=_{\text{df}} \{(x,y) \mid \forall (x,y)[(x,y), (y,x) \in {}_{pc}E]\}$

**Unilaterally connected elements,** $_{uc}E$, $=_{\text{df}} \{(x,y) \mid \forall (x,y)[(x,y) \in {}_{pc}E .\wedge. (y,x) \notin {}_{pc}E]\}$

**Disconnected elements,** $_{d}E$, $=_{\text{df}} \{x \mid \forall (x,y)[(x,y),(y,x) \notin {}_{pc}E]\}$

**Initiating elements**, $_{i}E$, $=_{\text{df}} \{x \mid \forall x[(x,y) \in {}_{pc}E]\}$

**Receiving elements**, ${}_{r}\boldsymbol{E}$, $=_{\mathrm{df}}$ $\{y|\ \forall y[(x,y) \in {}_{pc}\boldsymbol{E}]\}$

**Terminating elements**, ${}_{t}\boldsymbol{E}$, $=_{\mathrm{df}}$ $\{y|\ \forall y[(x,y) \in {}_{pc}\boldsymbol{E} \wedge \forall u(y,u) \notin {}_{pc}\boldsymbol{E}\}$

**Primary initiating elements**, ${}_{pi}\boldsymbol{E}$, $=_{\mathrm{df}}$ $\{x|\ \exists y[(x,y) \in {}_{pc}\boldsymbol{E} \wedge \forall u(u,x) \notin {}_{pc}\boldsymbol{E}]\}$

**Connected elements,** ${}_{c}\boldsymbol{E}$, $=_{\mathrm{df}}$ $\{(x,y)|\ \exists y((x,y) \in {}_{pc}\boldsymbol{E} \vee (y,x) \in {}_{pc}\boldsymbol{E}\}$

The distinction must be made between **component properties** and **system properties**.

**Component properties** describe relations between components; for example, that two components are unilaterally connected.

**System properties** describe the characteristic pattern of all components of the system with respect to a specific component property; for example, the unilateral connections of the system components are such that the system is characterized by *strongness*.

In view of the above Graph Theoretic developments, the *Affect Relation Properties* can now be defined. To bring all of the *Affect Relation Properties* together, ***affect relation*** will again be defined.

**Affect relation,** $\mathcal{A}$, $=_{df}$ A connection of one or more components to one or more other components.

$$\mathcal{A} =_{df} \{\{\{x\},\{x,y\}\} \mid P(x,y) \wedge x,y \in \mathfrak{X} \subset \mathcal{G}_0 \vee [(x = \mathfrak{U} \subset \mathfrak{X} \subset \mathcal{G}_0 \wedge y = \mathfrak{V} \subset \mathfrak{Y} \subset \mathcal{G}_0)]$$

**Directed affect relation,** ${}_{d}\mathcal{A}$, $=_{df}$ An *affect relation* that is path-connected.

$${}_{d}\mathcal{A} =_{df} \mathcal{A} \mid (x,y)^{\rightarrow} \in \mathcal{A} \supset (x,y)^{\rightarrow} \in {}_{pc}\boldsymbol{E}\}.$$

**Directed affect relations** may pass through more than one component. **Directed affect relations**, when also assigned a "magnitude" will be interpreted as a vector that will allow for topological analyses of the system vector fields.

**Direct directed affect relation,** ${}_{dd}\mathcal{A}$, $=_{df}$ A *directed affect relation* with a single directed-path.

$${}_{dd}\mathcal{A} =_{df} \{(x,y)^{\rightarrow} \mid (x,y)^{\rightarrow}{}_{n=1} \in \mathcal{A}_m \in \mathcal{A}\}.$$

**Indirect directed affect relation,** ${}_{id}\mathcal{A}$, $=_{df}$

A *directed affect relation* in which the path-connection is through other components.

$${}_{id}\mathcal{A} =_{df} \{(x,y)^{\rightarrow} \mid (x,y)_{n>1} \in \mathcal{A}_m \in \mathcal{A}\}.$$

**Connected affect relation,** ${}_{c}\mathcal{A}$, $=_{df}$

Connected components of an *affect relation* irrespective of direction of path-connectedness.

$${}_{c}\mathcal{A} =_{df} \{(x,y) \mid (x,y) \in \mathcal{A}_m \in \mathcal{A}. \vee (y,x) \in \mathcal{A}_m \in \mathcal{A}\}.$$

Connected affect relations may be used to analyze a system in terms of its total connectedness to determine potential behaviors under varying assumptions of connectedness.

# Information-Theoretic Properties

**Information** obtained from an *Information Base*, $\bar{I}_B$, will be analyzed to determine various affect relations. An *APT Analysis* will provide a sequence of system states that may be used to define various *Dispositional Behaviors*, $_{\mathcal{D}}\mathcal{B}$'s. Further, the $\bar{I}_B$ will be used to construct an *Extended-*$\bar{I}_B$ that will be used to make predictions concerning system behavior. The *Extended-*$\bar{I}_B$ is constructed using the *Behavior-Predictive Algorithm* (the *Phoenix Algorithm*) developed by *Raven58 Technologies*.

**Information** is made explicit for analysis by the use of *mathematical probabilities*. *Probabilities* define **information**. And, the probabilities used do not have to be "true"; they only have to lend themselves to a proper analysis of the system and its outcomes—its predictions.

In *ATIS*, the probabilistic definition of **information** is mitigated by the fact that behavioral predictions are **not** founded on the information, but on a structural analysis of the system derived from that information. That is, *behavior prediction* made possible by *ATIS* is dependent on *logical and topological analyses* rather than on the specific **information** input itself. **Information** for *ATIS* is used to determine *system structure* and is **not** the decision-making tool.

Further, information as used in *ATIS* is discrete. As the information "H" function is defined below, *ATIS* only uses a few discrete values of "H," normally equal to "0" or "not 0."

For example, input occurs when the value of "H" in the toput is such that H = 0. If "H" is anything other than 0, then the component is still toput, regardless of whether H = 0.1, H = 0.2, H = 0.7, etc. However, various analyses of H will be used to construct the *Extended-*$\bar{I}_B$. That is, the value of H will determine the category assigned a new system component so that the new system structure may be determined and analyzed.

**Information** is that which reduces **uncertainty**. In information theory, **uncertainty** is defined by a value, H, the entropy. **Uncertainty** is a *measure* of **variety** such that **uncertainty**, H, is zero when all elements are in the same **category**.

Information is defined as follows:

**Information,** $\wp$**, =**$_{df}$ A mathematical probability of occurrences defined by:

$$\wp =_{df} \{(c,v)|c \in W \subset \mathcal{G}_0 .\wedge. v \in (0,1]\}.$$

**Information** is a set of *ordered pairs* consisting of components, c, of the set "W", a subset of $\mathcal{G}_0$, and the real number "v," which is the probability distribution, p, that the *component* "c" occurs in W.

**Information** is represented as a probability so as to convey the *uncertainty* of that **information**. Thus, **information** will be the result of an "*uncertain event*," and referred to below as "*event uncertainty*." **Information** is a *Measure Property*.

**Event uncertainty, H,** =$_{df}$ A measure of information due to statistical uncertainty, real uncertainty, or enemy action.

$$H = -K \sum_{i=1\ldots n} p_i \log p_i;$$

where "K" is a constant related to the choice of a unit of measure, and "$p_i$" is the probability of occurrence of event "i". In *ATIS*, the **information**-*probability* will indicate the "reliability" of that information and the resulting assignment to an appropriate subset.

'**Event uncertainty**' is used here to imply probabilities that are subject to objective determination. Further, whereas '**event uncertainty**' may be defined in terms of *statistical uncertainty* or *real uncertainty*, the *information* in this research will normally be due to *enemy action*; that is, tychistic events that must be dealt with in the continuity of a behavioral system, the society.

**Non-conditional event uncertainty,** $_{nc}$**H,** =$_{df}$

*Information* that does not depend on other event uncertainty.

$$_{nc}\mathbf{H} =_{df} f\mathbf{H} \mid \sim\exists f\mathbf{H}_1(f\mathbf{H}: f\mathbf{H}_1)$$

**Conditional event uncertainty,** $_c$**H,** =$_{df}$ *Information* that depends on other event uncertainty.

$$_{c}\mathbf{H} =_{df} f\mathbf{H}: \exists f\mathbf{H}_1(f\mathbf{H}: f\mathbf{H}_1)$$