Problem-Solving Approach to Data Fusion

Alan Steinberg
Space Dynamics Laboratory/Utah State University

(asteinberg@sdl.usu.edu)

Abstract — This paper defines an approach for characterizing and solving data fusion problems in a system context. We suggest a general ontology of problem-solving processes and characterize several types of data acquisition/fusion problems in terms of this ontology. By relating data acquisition/fusion problem solution to a more general theory of problem-solving, the methodology framework will assist the information system planner or analyst to

- Find analogies in other domains to problems or sub-problems of interest
- Learn from experience in solving analogous problems
- Leverage techniques and metrics used in solving analogous problems.

Keywords: Data fusion, data acquisition, problem-solving, sensor management, resource management, planning, information fusion, pattern recognition, learning systems, adaptive modeling.

1. Data Fusion and Problem-Solving

In the most general sense, problem-solving involves planning and execution of actions. Often problem-solving involves elements of uncertainty. Such uncertainty can include:

1. Data Uncertainty — Lack of total confidence in available information (usually the result of imperfect information sources);
2. Model Uncertainty — Uncertainty in the expectations concerning the characteristics and behaviors of entities or situations;
3. Technique Uncertainty — Not knowing how to employ available resources to effect desired results;
4. Goal Uncertainty — Lack of clear goals (i.e. incomplete, inconsistent, imprecise or fragile valuation of possible consequent states).

Data fusion processes seek solutions to problems of a particular kind: estimation problems. Data fusion solves such problems by combining multiple data; e.g. by filtering commensurate data or by inferring characteristics that may not be directly observed.

Data fusion is generally not performed in isolation, for the sake of data fusion itself. Rather, fusion is employed in the process of solving problems; its role being that of reducing at least some of the uncertainty factors in problem-solving. The first three of these uncertainty factors are attacked directly or indirectly in the process of estimating:

1. the state of the problem domain,
2. the fidelity of models of for predicting or interpreting characteristics of that domain, and
3. the effectiveness of techniques available to operate within that domain; including techniques for data acquisition and fusion.

Furthermore, situational understanding can have a role in formulating and evaluating interim goals. It may influence even our highest-level goals (e.g. the value a person ascribes to selfish versus altruistic goals may vary with his perception of the situation).

The problems that are expected to confront information system designers and information analysts are notable not only for their difficulty, but also for their diversity. These problems include those that are dominated by sensor phenomenology as well as those that focus on human and organizational activity and intent.

We seek a general framework that will allow specific analytic methods to be developed and used effectively across this wide and expanding range of problems. Such a framework will facilitate the ability of planners and analysts to

1. Discover the significant characteristics of a new problem, using analytic and pattern-recognition techniques (Problem Pattern Discovery);
2. Recognize the similarities between the given problem and others that may have been solved already (Problem Pattern Recognition);
3. Decompose the problem so as to factor out sub-problems that have known solutions (Goal Deconstruction); and
4. Apply existing analytic procedures and tools to search for solutions to the problem or its parts (Plan Construction).

Such a framework, it will be noted, is broader than decision theory, in just the same way that data fusion is broader than estimation theory: what is sought is not merely a method for selecting among alternative responses to a situation (or, in fusion, alternative interpretations of a situation), but a method for operating over time within a complex, dynamic and uncertain situation for which goals and models may likewise be complex, dynamic and uncertain.

The specific challenge to the present research is to develop a process for information system planners and analysts to effectively perform the following appropriate to the individual problem:

- Select feature sets for exploitation;
- Discover exploitable context;
- Combine uncertain, or poorly-modeled information;
- Estimate and correct systematic errors in sensors/sources, in sensor alignment (registration), and in prior models;
- Model problem variability;
- Predict resource and system performance; and
- Predict solution cost.

More generally, we would like a framework that will systematically provide answers to the following questions whenever a new problem is identified:

- **Problem Characterization**
  - What are the problem assumptions (existing or required)?
  - What is the dimensionality of the problem?
  - What are the influential parameters and their ranges?
  - What are the problem constraints?
  - Can the problem be sub-divided into parts that are easier to solve?
- **Problem Recognition**
  - Has the problem been solved before?
  - Has the same problem appeared in a different form and is there an existing solution?
  - Is there a related problem with the same unknowns?

By casting data acquisition and fusion problems in a general problem-solving framework, system designers and analysts will be able to build systematically on a common body of resources (sensing and analysis procedures, software tools, knowledge bases, etc.) in planning and executing solutions to diverse problems. The desired problem-solving framework will incorporate the ability to learn; i.e. to construct more effective problem solving methods by systematic improvements in the tools for problem pattern discovery, problem pattern recognition, goal deconstruction and plan construction.

2. Problem-Solving Model

The characterization of problem types and of applicable solution methods is the concern of disciplines ranging from philosophical epistemology and cognitive psychology, to education theory and artificial intelligence, to systematic approaches to planning in diverse business or military applications.

Although there is no accepted general model for problem-solving, it is possible to abstract the key findings as they apply to data exploitation problems. Current research in software problem-solving methods has focused on characterizing problems in terms of problem frames. A problem frame is a template characterizing a class of simple problems; i.e. problems for which a reliable solution methods is known. It also characterized a class of problems whose solution is likely to be reusable. Generally speaking, a problem frame is a structure of principal parts (elements of the objective system and of its relevant environment and relationships among these elements) and a solution task. The solution task is always to bring about or maintain the required relationships.

The problem frame model is somewhat restrictive, in that it assumes that goals (the “required relationships”) can be specified with some clarity. There are cases, however, where goals and design constraints are not initially well defined. Some problems are not so much goal-driven as data-driven; in which exploitable characteristics of a domain are discovered.

There are also cases in which there is some uncertainty as to what data is relevant to the problem or as to the characteristics of entities or situations of interest. Furthermore, in many difficult problems there is no established set of tools – of analytic procedures, processing techniques, or data libraries – known to be effective in solving the problem or similar problems. Lacking predictive models of the operational domain, experimenters are driven to trial-and-error methods: do something and see what happens.

To develop a framework for characterizing and solving problems of these types, we propose a model that explicitly represents the contributions of the factors listed in Section I:

1. **The goals** of the problem-solver;

---

3 An analogy might be noted in the relation between Decision and Planning with that between Object Assessment (so-called level 1 fusion) and Situation Assessment (level 2 fusion). This analogy is not accidental: the duality between Resource Management and Data Fusion [1] results in a dual set of Resource Management “levels”, as discussed in Section 4.
2. The techniques available to solving the problem (to include all possible information assessment and response techniques using available human, processing, sensing and effecting resources);

3. The problem model; i.e. the assumptions that can be made about the problem situation: the state of the relevant environment or domain, including one’s resources and external entities and their physical, informational and cognitive relationships;

4. The data or information available to the problem-solver in understanding the problem situation and the effects of planned or executed actions.

Problems of interest to the information system designer and analyst can involve all of these four factors. Correspondingly, we can describe simple data fusion problem-solving processes as

1. Goal-Driven: answering specific questions
2. Technique-Driven: applying standard methods; e.g. in routine information collection and analysis
3. Model-Driven: recognizing objects or situations of known types
4. Data-Driven: discovering the characteristics of situations.

Data fusion developments to date have focused on goal- and model-driven processes; concerned with recognizing occurrences of activities of interest (e.g. specific types of adversarial military or business activities as a guide to responsive action; or, to some extent, in theoretical physics). However, there are data fusion problems that are primarily technique- and data-driven: concerned with detecting and understanding anomalies and discovering significant characteristics in available data and in collateral information (e.g. in inferring unforeseen opportunities or threats; or, to some extent, in experimental physics).

As depicted in Figure 1, the four processes are paired: a data-driven process may be considered to be model-seeking (i.e. the desire is to find a model that explains the data). Conversely, a model-driven process is data-seeking (i.e. looking for data that instantiates the model). Similarly, a goal-driven process seeks techniques (i.e. available actions) that can be assembled in a plan to achieve the goal. A technique-driven process attempts to accomplish useful results by following sound practices.

Many real-world problems can be characterized as being primarily of one or another of these four types and there are many problem-solving methods that have been designed to solve one or other of these one-dimensional problems, as illustrated in Table 1.

As suggested in the table, some methods are more suitable for some types of problems than for others. For example, model-driven methods presuppose the existence of valid models that describe the recognizable or, at least, discriminating, features of entities or situations of interest. Other methods are appropriate when either unknown (in the sense of unmodeled) entities are likely to be encountered or there is a concern for the fidelity of available models (as when camouflage, concealment or deception is used to diminish the recognizability of entities).

As depicted in Figure 1, the four processes are paired: a data-driven process may be considered to be model-seeking (i.e. the desire is to find a model that explains the data). Conversely, a model-driven process is data-seeking (i.e. looking for data that instantiates the model). Similarly, a goal-driven process seeks techniques (i.e. available actions) that can be assembled in a plan to achieve the goal. A technique-driven process attempts to accomplish useful results by following sound practices.

![Figure 1. Types of Problem-Solving Processes](image)

Table 1. Categorization of Problem-Solving Processes

<table>
<thead>
<tr>
<th>Process Type</th>
<th>Procedure</th>
<th>Applicable Techniques</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Goal-Driven</td>
<td>Decomposition:</td>
<td>• Back-Chaining</td>
<td>• Trip Planning</td>
</tr>
<tr>
<td></td>
<td>Analyze Goal into Sub-goals</td>
<td>• Parsing</td>
<td>• Reconnaissance</td>
</tr>
<tr>
<td>Technique-Driven</td>
<td>Construction:</td>
<td>• Forward-Chaining</td>
<td>• Following a Recipe</td>
</tr>
<tr>
<td></td>
<td>Assemble Plan</td>
<td>• Genetic Algorithms</td>
<td>• Playing by the Rules</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Synthetic Annealing</td>
<td>• Surveillance</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Trial-and-Error</td>
<td></td>
</tr>
<tr>
<td>Model-Driven</td>
<td>Deduction:</td>
<td>• Templating</td>
<td>• Target Recognition</td>
</tr>
<tr>
<td></td>
<td>Recognize Problem Type</td>
<td>• Case-Based Reasoning</td>
<td>• Signal Classification</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Operant Conditioning</td>
<td>• Medical Diagnosis</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Supervised Learning NN</td>
<td>• Fault Diagnostics</td>
</tr>
</tbody>
</table>
More complex problems must be solved using combinations of these approaches. Often, one or more of the problem-characterization factors is inadequately specified to allow effective problem solution. For example, if goals are not well defined or are subject to revision, the problem solution must be driven by other factors; i.e. by considerations of resources, situation and data.

3. Illustrations of the Methods

Figure 2 provides illustrative comparisons of the four basic types of problem-solving processes. These concepts are discussed in the following paragraphs.

3.1. Data-Driven Problem-Solving

Data-driven processes involve discovering patterns in data. Classic examples are in problems of creating new taxonomic schemes: Linnaeus in botany or Mendeleev in Chemistry. The JDL Data Fusion Model [8,9,10] was developed by such a process, as, indeed, was the present analysis. The goal of such taxonomic analysis is to characterize the factors of a problem domain and their relationships.

People use data-driven processes when trying to understand unfamiliar objects or situations: consider the situation that would occur when a primitive tribesman detects an artifact of modern civilization, say a flashlight left on a beach; or when a linguist attempts to decipher an unfamiliar language. The person employs inductive (generalization) and abductive (explanation) methods to build and test explanations that can account for the new data. The person is seeking to build a model – i.e. an explanation – for the data. Formal data-driven methods include principal component analysis, entropy techniques, statistical learning methods, clustering and unsupervised neural networks. The distinction of one type of data-driven method from model-driven methods is stated in [11, p. 37]: “… cluster analysis is mostly used as a descriptive or exploratory tool, in contrast with statistical tests which are carried out for inferential or confirmatory purposes.”

Another type of data-driven method, statistical learning theory, is applicable when

(…one) does not have reliable a priori information about the statistical law underlying the problem or about the function that one would like to approximate. It is necessary to find a method to infer an approximating function from the given examples in this situation. [1, p. 3]

A typical data-driven application is that of establishing discrimination criteria among a set of entity types (e.g. to distinguish hostile military aircraft) from other objects (e.g. friendly aircraft). The effectiveness of various features sets in separating the distributions of the target class of interest is used in developing and evaluating candidate discrimination algorithms for operational use (e.g. in air defense targeting decisions).

3.2. Model-Driven Problem Solving

A model-driven process is the complement of a data-driven one. The process involves recognizing known patterns in data, rather than discovering new patterns. Model-driven processes are appropriate to recognition problems: i.e. those in which the objective is to determine the presence and particular state of entities of known types.

Whereas data-driven methods have a prominent role in technical intelligence (e.g. in building and validating physical models of foreign weapon systems), model-driven methods are the principal approaches in classical information analysis; e.g. in Automatic Target Recognition (ATR). Here the problem is that of recognizing the presence of entities and situations for which models exist (the models generally having been developed and validated by data-driven processes). Model-driven methods are characteristically deductive: covariance analysis, Fisher likelihoods, Bayes’ law, etc.

Sensor measurements are compared with measurements that are predicted from an object/situation model that is a candidate representation of the given situation.

Most developments in data fusion to date have been model-driven: inferring the identity, location, track, activity, etc., of entities by matching the expected distribution of entity characteristics and behaviors. Most of the workaday tools of data fusion are model-driven: Kalman filters, Bayesian or Dempster-Shafer classifiers, validation gates, formation templates, match factors and many (but not all) knowledge-based techniques.

3.3. Goal-Driven Problem Solving

Goal-driven problem solving involves the decomposition of problem goals and constraints into a plan for achieving the goals. This is the classical approach used in
Operations Research and in most AI problem-solving applications. Often, high-level goals are recursively decomposed into sub-goals. Figure 3 shows the characteristics of such a hierarchical planner.

Here, candidate sub-plans are generated, evaluated and selected on the basis of goals and constraints that are propagated downward. Performance expectations are propagated upward, possibly inducing successive redefinition of goals and constraints.

Figure 2. Comparison of Problem-Solving Processes

As each planning node receives lower-level candidate plan segments, it estimates possible contentions, interferences and other interactions among the plan segments and the impact on their respective higher-level goals. In this way, plans responsive to global mission goals are assembled in a hierarchical fashion.

A goal-driven approach to data acquisition and fusion is presented in [13,14], in which the goal is that of choosing a sequence of (data acquisition or fusion) actions and a decision function to maximize the expected net payoff.

Analogous to the methods in data fusion to constrain searches for association hypotheses, are methods in planning to search only over assets, goals and implementation times with high \( a \) priori net payoff. [15]

Mission planning and mission management are familiar goal-driven processes. Another is that of traditional system engineering, as discussed in [15,16]. High-level system requirements and design constraints are decomposed in a hierarchical fashion, providing specification of ever more detailed system functions. A corresponding performance assessment is composed at each design level, providing the designers assessments of the integrated performance of system elements. This diagnostic function is primarily a model-and data-driven one.

3.4. Technique-Driven Problem Solving

Technique-driven processes involve adapting methods that have worked successfully in presumably similar problems. Such methods tend to be conservative and are most appropriate in applications in which problem-variability is expected to be small.

Routine information production generally involves “handbook” methods. Amateurs can cook like professionals by strictly following recipes. People adhere to codes of ethics, of etiquette and of law in the hope of achieving good results. System developers often use “best practices” as a method for selecting software tools, software selection. Historically useful techniques are forward-chained to seek useful solutions. The rat in a maze or the fly on a windowpane attempts to solve its problems by drawing from a small repertoire of methods.

There is the potential for learning from experience. Means of technique adaptation include operant conditioning based on adventitious behavior, genetic algorithms, and supervised Neural Networks.
3.5. Hybrid Problem Solutions

More difficult problems require hybrid approaches. As a familiar example, consider the way in which we solve a jigsaw-puzzle problem. As depicted in Figure 4, the initial goals are readily defined, as are the available techniques and available materials (the “data” in our metaphor). The fourth element – a situation model – may not be available; e.g. we may not have the picture on the puzzle box to use as guide in solving the puzzle as a recognition problem.

![Figure 4. A Familiar Hybrid Problem-Solving Example](image)

In this case, we employ our repertoire of puzzle-solving techniques based on some general expectations: that our pile of pieces comprises all and only those relevant to the solution, that the result will form a regular shape – probably a rectangle – and a recognizable image.

As we proceed, we may find that our initial goals need to be revised if some or all of these assumptions are not met. Significantly, we employ diverse problem-solving processes without explicit pre-planning. In the early stages, we may try to define the puzzle edges (a technique/goal-driven process). We may intersperse such activities with attempts to separate pieces by color or texture (data-driven). At various stages, we use recognizable picture elements – a hat, part of a face, etc. – to guide our assembly process (model-driven). Particularly in the end-game, when there are few pieces left (generally ones with little pictorial information to aid the model-driven process), we search for pairs of pieces that will fit together (data-driven).

The figure indicates in red italics the analogy with data acquisition/fusion problem-solving. Once again, the analysis process cannot always be pre-planned. Rather, various techniques must be employed as the problem-solution evolves. Often there is uncertainty as to the relevance or the completeness of available data.

In short, the process of problem-solution often is a highly adaptive one: problem definition and solution often interact in ways that cannot be anticipated.

4. Problem-Solving and Data Fusion

4.1. General Process Model

Figure 5 shows the direct functional relationship between (a) the management of the data acquisition and fusion process – as a typical problem-solving process – and (b) the state estimates that result from such a process. This relationship derives from the formal duality between the management and fusion processes. [1,16] As indicated in Figure 5(a), the general goal of data acquisition and fusion will be to estimate

1. the component elements of a domain (e.g. military equipment, their types, locations, capabilities and readiness state);
2. the relationships among such entities (i.e. the “structure” of the situation): organizational, communications, physical disposition and interaction and intentional relations);
3. the event histories of entities and structures;
4. the expected consequences of the situation; to include intentions of elements and structures and unintended consequences.

These goals are those of the various JDL data fusion levels, as noted in the figure.

The neat symmetry between Figures 5a and 5b is explained by the realization that all processes can be described in terms of a handful of factors. This includes both the data acquisition/fusion process (Figure 5a) and the processes that describe the world state to be estimated (Figure 5b).
Figure 5a will be recognized as a version of the IDEF0 process model. Figure 5b is, of all things, a variant of Aristotle’s model of physical events.\footnote{Aristotle [17,18] postulated four factors, generally translated as Material, Formal, Efficient and Final “causes”. These correspond more or less to our Data-, Model, Technique- and Goal-Driven problem-solving methods, respectively as shown in Figure 6b (with terminology more commonly used in fusion system engineering). Of course, the word ‘cause’ (‘κατηγορία’ in Classical Greek; translated as ‘causa’ in Latin) cannot be equated to our modern sense of ‘cause’. Rather, it has more the sense of factor, as it is rendered in [18].}

As discussed in [14,16], there are deep parallels between (a) the goal and technique-driven process of resource management – to include the management of sensors and of data fusion resources – and (b) the data and model-driven process of data acquisition and data fusion. The one involves finding a suitable assignment of available resources to a set of goals; the other involves finding a suitable assignment of available data to a set of models.\footnote{Note also the similarities between the process of building and using data fusion systems: System analysis feeds performance estimates back to the system engineering process (including the process of developing data acquisition and fusion systems) much as data fusion can feed back performance estimates to the resource management process (including the process of managing data acquisition and fusion).}

Figure 6 shows the relationship between the management and fusion processes and the parallels between the management and execution of data acquisition/fusion processes are summarized in Table 2.\footnote{The JDL data fusion level 4 is actually a resource management function, whereas the other levels relate to data association and estimation [2,5]. Analogously, resource management level 4 is actually an estimation function, whereas the other levels relate to planning and control.}

We can extend the fusion/management duality by defining a set of resource management “levels” that correspond to the JDL data fusion levels as presented in [9,14]. The result is shown in Table 3.\footnote{The JDL data fusion level 4 is actually a resource management function, whereas the other levels relate to data association and estimation [2,5]. Analogously, resource management level 4 is actually an estimation function, whereas the other levels relate to planning and control.}

4.2. Adaptive, Hybrid Data Fusion

To date, most data fusion developments have been model-driven: variably employing Bayesian, evidential, fuzzy, or cognitive models.

Many practical data fusion applications, however, are hampered by the lack of robust models of entities, behaviors and relationships. In these cases, there can be a benefit to use data-, goal- and technique-driven methods to supplement and refine model-driven ones.

There has been some research into hybrid fusion techniques in the U.S., notably within DARPA’s DMIF (Dynamic Multi-User Information Fusion) and AFRL’s Adaptive Sensor Fusion programs.

These programs have begun the investigation into systems in which fusion processes are selected and controlled on the basis of their expected effectiveness in achieving a desired information state, given the system’s current assessed information state, employing a planning processes similar to that described in 3.2 above.

4.2. Adaptive, Hybrid Data Fusion

To date, most data fusion developments have been model-driven: variably employing Bayesian, evidential, fuzzy, or cognitive models.

Many practical data fusion applications, however, are hampered by the lack of robust models of entities, behaviors and relationships. In these cases, there can be a benefit to use data-, goal- and technique-driven methods to supplement and refine model-driven ones.

There has been some research into hybrid fusion techniques in the U.S., notably within DARPA’s DMIF (Dynamic Multi-User Information Fusion) and AFRL’s Adaptive Sensor Fusion programs.

These programs have begun the investigation into systems in which fusion processes are selected and controlled on the basis of their expected effectiveness in achieving a desired information state, given the system’s current assessed information state, employing a planning processes similar to that described in 3.2 above.

Figure 7 extends this adaptive fusion process to a coordinated adaptive information exploitation process: integrating the management of information acquisition/, fusion and communications. This concept allows model, like situation estimates, to be refined by data-, goal- and technique-driven processes. Goals themselves are subject to modification by model-, data-, and technique-driven processes.

5. Research Topics

We have argued for the value of techniques that will systematically integrate data acquisition (including interactive
data acquisition), data communication and fusion, model validation, resource performance estimation, technique development and refinement into goal-directed systems.

Among suggested areas of research are the following:

- Robust methods to recognize and adapt to stochastic ill-posed problems (i.e. problems in which the values of function cannot be measured directly, resulting in the possibility of unstable approximations);
- Systematic methods to characterize off-nominal data as a function of process noise, sensor random and calibration bias errors, model error, and cross-sensor biases;
- Methods to define cost functions in terms of the above uncertainties in data and models together with uncertainties in the effects of information acquisition, processing and responsive actions (including contentions, interference and other interactions among actions).
- Methods for systematically refining goals, incorporating situation refinement/state estimation goals with “operational mission” goals.

![Diagram](image-url)

**Figure 6.** Resource Management/Data Fusion Model

**Table 2.** Parallels between the Management and Execution of Data Acquisition/Fusion

<table>
<thead>
<tr>
<th>Resource Management (including Data Acquisition/Fusion Management)</th>
<th>Data Acquisition/Fusion</th>
</tr>
</thead>
<tbody>
<tr>
<td>Problem Recognition</td>
<td>Object/Situation Recognition</td>
</tr>
<tr>
<td>Plan Composition</td>
<td>Object/Situation Model Composition</td>
</tr>
<tr>
<td>Resource Assignment and Coordination</td>
<td>Data Association and Explanation</td>
</tr>
<tr>
<td>Resource Management Tree alternatives:</td>
<td>Data Fusion Tree alternatives:</td>
</tr>
<tr>
<td>- Centralized Planning</td>
<td>- Centralized Fusion</td>
</tr>
<tr>
<td>- Distributed Planning</td>
<td>- Distributed Fusion</td>
</tr>
<tr>
<td>- Networked Planning</td>
<td>- Networked Fusion</td>
</tr>
<tr>
<td>Resource Management Node functions:</td>
<td>Data Fusion Node functions:</td>
</tr>
<tr>
<td>- Problem Alignment</td>
<td>- Data Alignment</td>
</tr>
<tr>
<td>- Task Association (Planning)</td>
<td>- Data Association</td>
</tr>
<tr>
<td>- Plan Generation</td>
<td>- Hypothesis Generation</td>
</tr>
<tr>
<td>- Plan Evaluation</td>
<td>- Hypothesis Evaluation</td>
</tr>
<tr>
<td>- Plan Selection</td>
<td>- Hypothesis Selection</td>
</tr>
<tr>
<td>- Control (Plan Realization)</td>
<td>- Estimation (Hypothesis Realization)</td>
</tr>
</tbody>
</table>

**Table 3.** Resource Management Levels Corresponding to the JDL Data Fusion Levels
<table>
<thead>
<tr>
<th>LEVEL</th>
<th>RESOURCE MANAGEMENT</th>
<th>DATA ACQUISITION &amp; FUSION</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>React (Stimulus Response)</td>
<td>Detect (Infer Signal)</td>
</tr>
<tr>
<td>1</td>
<td>Decide (Single Action)</td>
<td>Assign (Infer Object)</td>
</tr>
<tr>
<td>2</td>
<td>Plan (Aggregate Actions)</td>
<td>Relate (Infer Aggregation)</td>
</tr>
<tr>
<td>3</td>
<td>Coordinate (Coordinated Actions)</td>
<td>Cost (Infer Impact)</td>
</tr>
<tr>
<td>4</td>
<td>Infer Performance (Estimation)</td>
<td>Process Refinement (Management)</td>
</tr>
</tbody>
</table>

![Figure 7. Adaptive, Hybrid Data Acquisition and Fusion Management](image)

### 6. References


