

Functional Analysis Systems Technique (F.A.S.T.) as a Group Knowledge Elicitation Method for Model Building

J. E. Bartolomei; T. Miller

United States Air Force
Wright-Patterson AFB OH, 45431
(937) 255-6522; (937) 255-3672

Information gathering is the most important and often the most difficult phase of dynamic system model building. Many methods have been proposed to elicit knowledge from systems experts. Most modelers avoid the social and political barriers found in group elicitation and have focused on various interviewing techniques. If properly employed, group elicitation can be an effective and extremely efficient method to get the necessary knowledge to analyze and model a system. This paper discusses the use of the Functional Analysis Systems Technique (F.A.S.T.) diagramming method as a tool for eliciting knowledge during the information-gathering phase of the modeling process.

Key Words: Group model building, knowledge elicitation, Functional Analysis System Technique

The Functional Analysis System Technique Process: F.A.S.T

F.A.S.T. was invented during the Value Analysis (VA)/Value Engineering (VE) revolution of the 1960s. It is a rigorous method for understanding complex systems by converting the “activities” performed in a system to the “functions” performed by the system for its customers. System Engineers and Value Analysis specialists use this method for product improvement, process improvement, systems design, and systems architecting. Its creation marks the completion of their formal information-gathering phase and defines the current state of a system at a high level.

For example, which is more important to understand from a systems perspective, the fact that today someone reviewed a document (their activity) or the fact that that activity went to improve security (their function) in the organization? While activity is important to getting a job done, it does not necessarily benefit the customer. In fact, function is what the customer ultimately pays for while activity is what they get and it can become narrow and self-serving. The VA specialists have a term for this; they call it selfish-sectional efficiency. It means the

individual or unit becomes extremely efficient at the expense of the rest of the organization, and more importantly, at the expense of the customer.

A non-process example may help to illustrate why not understanding “function” can severely effect an organization. Gas powered lawn mowers have been around for a long time, and yet, these highly skilled people, with big fancy manufacturing plants missed one of the most important market niches of the last 40 years; the string trimmer. Why is this? It’s because mower people saw themselves by their everyday “activities”— making lawn mowers. They missed the function they were providing for their customers “groom property,” hence the multimillion-dollar string trimmer market. Unfortunately the lawn mower manufacturers and their customer didn’t even care that it was not done with a lawn mower. In fact, it was a non-mower manufacturer who introduced the string trimmer and stole a good chunk of the business from mower companies. Even today, many mower manufacturers have still not gotten into this market because they are still stuck in the activity of mowing grass vs. the higher function of grooming property.

The F-22 F.A.S.T. diagram Example

This paper will examine, using the F.A.S.T. exercise, to understand the extremely complex organization, a government weapon system program office (SPO) development team. The specific program is the Air Force’s F-22 Advanced Tactical Fighter System Program Office. Leadership from the F-22 System Program Office desired to use F.A.S.T. and other system analysis tools to improve their office and program management practices. As part of this effort, they decided to map their entire F-22 program using a F.A.S.T. methodology.

F.A.S.T. is a comprehensive, hierarchical block-diagramming tool that visually portrays the key functions an organization performs for its customers in a cause and consequence fashion. F.A.S.T. represents the current state of a system in functions without regard to timing and flow of activities. This subtlety raises the F.A.S.T. diagram from the normal activity-centered block-flow diagram to the higher-level function diagram. In Value Analysis, this change gives problem solving teams a new perspective on their situation, which, in turn, allows them to be much more creative in later brainstorming sessions. For the modeler, F.A.S.T. is a highly efficient method to capture a detailed model of a system’s functions.

Use Organizational Experts and Work Offsite

The key to generating a successful F.A.S.T. model is to ensure that the right people are chosen for the F.A.S.T. team. Organizational experts are the main contributors to F.A.S.T. diagram. Gathering this data is best accomplished at a meeting held offsite of your regular work environment and with the leadership of the organization. In this example it was done with the senior members of the Air Force's F-22 System Program Office.

The difficult aspect of F.A.S.T., that a system modeler will have to overcome, is team reluctance to convert activities to functions in two words. Until the modeler is comfortable with the F.A.S.T. diagramming technique, it may be better to have a F.A.S.T. expert guide your team through the process. Most certified value specialists (CVS) have this capability.

Knowledgeable Participants of the F-22

The F.A.S.T. team was lead by the second in command, the Deputy System Program Director. Other members included the Chief Engineer, Deputy Chief Engineer, Chief of the Contracting Division, Chief of the Financial Division, Chief of the Weapon System Program managers, Flight Test Director, and lead Weapon System Engineer. Each of the participants was selected because for their knowledge of the overall system and for their respective decision-maker role for their groups within the SPO.

Role of the Facilitator(s)

Because of the size of the F.A.S.T. team, and inexperience with F.A.S.T. diagramming, three facilitators were selected to guide the team. Two were new to F.A.S.T. and a third party was a F.A.S.T. expert. The two new facilitators were trained in the F.A.S.T. process before the team started. All three, collectively guided the team through the process.

Customer Definition

One of the principles of a quality design is to ensure all functions and activities are traceable back to a customer requirement. So, before any functional allocations could be performed, customers and objectives had to be defined. As such, the first step in the F-22 process was for the team to identify the customers of their system.

For the F.A.S.T. exercise, customers were defined as any party that directly benefited from the products and services that the System Program Office generated. After a short discussion, the team quickly identified the main customers of the system—Air Combat Command, the Pentagon (SAF/AQ), the AF Audit Agencies, internal SPO leadership, and the F-22 contractors.

Definition of Function

Since the primary objective of a F.A.S.T. diagram is to teach the team members to think in terms of higher-level functions rather than everyday activities, the team was given small samples of a systems to practice on. The functional approach to problem solving is the cornerstone of Value Analysis (VA), in that it translates the structure of any system into a structure of words. In short, synthesizing a system in terms of functions deepens the team's intuitive appreciation of the entire system. (Fowler:1990)

Functions, are a simple two-word, verb-noun descriptions of each activity. These functions can also be called a “factive” to minimize the confusion with organizational functions like engineering, finance, or contracting.

Below is an example of a factive taken from an automobile control system:

- Drive car
 - Monitor environment
 - Monitor instrument data
 - Control direction
 - Control speed
 - Control visibility
 - Control human comfort
 - Control car health

Drive car is a higher level function. The other functions represent the means to achieve the higher level function. This is an example from a hardware design. The functions defined for a management system are quite different.

Team Size

Studies have shown that teams of 5-6 people work well together. (Fowler:1990) Above this number, teams tend to drift into small informal groups with only a core of three to four people doing any real work. To head-off this tendency, keep your groups to five people. When teams are less than five, problems can also arise. Smaller groups lack sufficient understanding of the overall system to create an effective F.A.S.T. diagram.

Generating Functions

Once the team understood the concept of function, the facilitators led the team through an extensive brainstorming session—to create functions. Each sub-team was asked to create 50 - 100 separate functions. As a procedure point, functions were placed on Post-it Notes® and then on easel pads for all the team to see. One of the easiest ways to define functions is to ask “why” an activity is done. This questioning is not done to challenge someone’s job; it’s done to get at the reason the activity exists—from the customer’s perspective. Getting functions described in two words, a verb-noun combination, is often difficult because people tend to think in terms of the way we name things. For example, the Air Force has a coordination cycle on all documents before the boss can sign them. This means that every group referred to in the document has seen it in final form and either agrees or disagrees with it. Therefore, when the team was asked what the function of the coordination activity was—why coordinate—the response was varied. The first function suggested was “coordinate document.” However, this just describes the activity in two words; it is not a function. To get at the real function, the facilitator then ask “why” people coordinate on a document? At that point, the following real function(s) began to appear: acknowledge read, verify content, signify agreement or disagreement, and authorize action. It is typical that one activity will generate several functions. However, once a function is defined, it does not need to be defined again just because it occurred elsewhere in the system. Remember, the F.A.S.T. diagram shows the state of the system over all time and therefore does not need to show duplicate or repetitive functions.

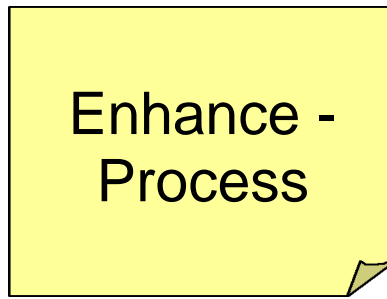


Figure 1. Two-word function description of the activity—file documents—on Post It Notes®

Mapping Functions

Upon completion of the function generation phase, the team began the functional analysis by creating the F.A.S.T. diagram. Each of the functions remaining after a scrub of repetitious functions was compared to the other functions and was placed on the map. The functions were organized by comparing each to each other using a “how and why” arrangement. A sample how-why relationship is shown below in Figure 1.

To test for the correct sequence between the Maintain Communications and Sustain Operations functions, put the two in a line and ask “why” of one to the right. In the example shown, ask “why maintain communication. If the answer—to sustain operations—makes sense, the functions are in the correct order. Test this by asking “how” in the opposite direction—how does one sustain operations, if the answer is by maintaining communication, then the two are in the correct order. The how-why logic also has a built in test. To see this, switch the two

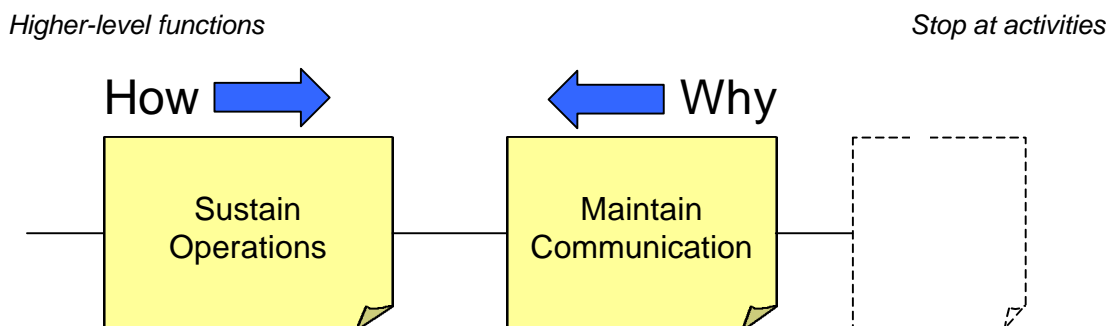


Figure 1. How –Why relationship verifies diagram order

functions and ask the same questions again. You will notice that the how-why questions no longer make sense. This shows the functions are in the wrong order. Repeat this how-why process for the next logical functions to the left and right of these two. In the end, a F.A.S.T. diagram will be created.

The final structure is a horizontal, logic diagram verified with how-why logic. Primary functions are leftmost on the diagram and supporting functions extend to the right in descending order of generality. The diagram stops on the right when further asking “how” could only be answered by putting “activities” on the diagram.

After several iterations of the mapping process, the team agreed on a final F.A.S.T. diagram that was representative of the all the key functions the system performed.

Ultimately, the F.A.S.T. exercise is a process of creating a functional flow block diagramming of a system that already exists—its current state. The entire process mirrors what designers of complex system perform when creating a system’s architecture.

Products and Services

Once the team developed the structure of the functions performed within the system, they then identified all of the key products and services performed in support of each function. In addition, the team identified the customers associated with each product and service and the party responsible for executing the tasks. A detailed list of the products and services identified by the SPO is located in Appendix.

Allocation of Efforts

In the next step, the team determined the resource allocation in the support for each function. Experts, representing each critical area, identified the number of resources that supported each function. The team was then able to assign those resources to each function. This information allowed the team to locate areas of excess and make efforts to lean those areas.

Unfortunately, when the teams used F.A.S.T. as an analytic tool to make decisions, there was no consideration of the dynamic relationships between function and activities. The failure to consider these interdependencies limits the ultimate usefulness of the F.A.S.T. model as an independent tool for decision making. Though not effective as a stand-alone analytical tool, the F.A.S.T. does provide a rigorous approach for identifying the functional architecture of the

system and therefore has great value to the expert system designer. Other systems analysis tools, like system dynamics modeling, can enhance the benefit of F.A.S.T. models by capturing the interactions of the components of the system..

F.A.S.T. diagram Colors

The F-22 SPO F.A.S.T. diagram shows their primary functions are the following: maintain support, sustain operations, lead the enterprise (AF – contractor combination), manage risk, and obtain funding. Each of these primary functions is followed by supporting functions and is ordered using the how-why logic explained earlier.

The colors on the F.A.S.T. diagram were added later when the F-22 teams decided to allocate people to the F.A.S.T. It was discovered that this allocation could not easily be done by function. Therefore, the diagram was subdivided by color and people were allocated by everyday activities even when they crossed functional areas.

The F-22 SPO's diagram was created in one day. Its form is that of the technical F.A.S.T. diagram. The information provided by the F.A.S.T. model was used in the information gather phase during the development of system dynamics model of the engineering function within the organization, and used in concert with other values analysis tools to improve the management function of the SPO. (Bartolomei et al:2001) Figure 2 represents the entire F-22 SPO model.

F22- VSA FAST Diagram

16 Dec 98
Rev 14 May 98

HOW →
Why ←

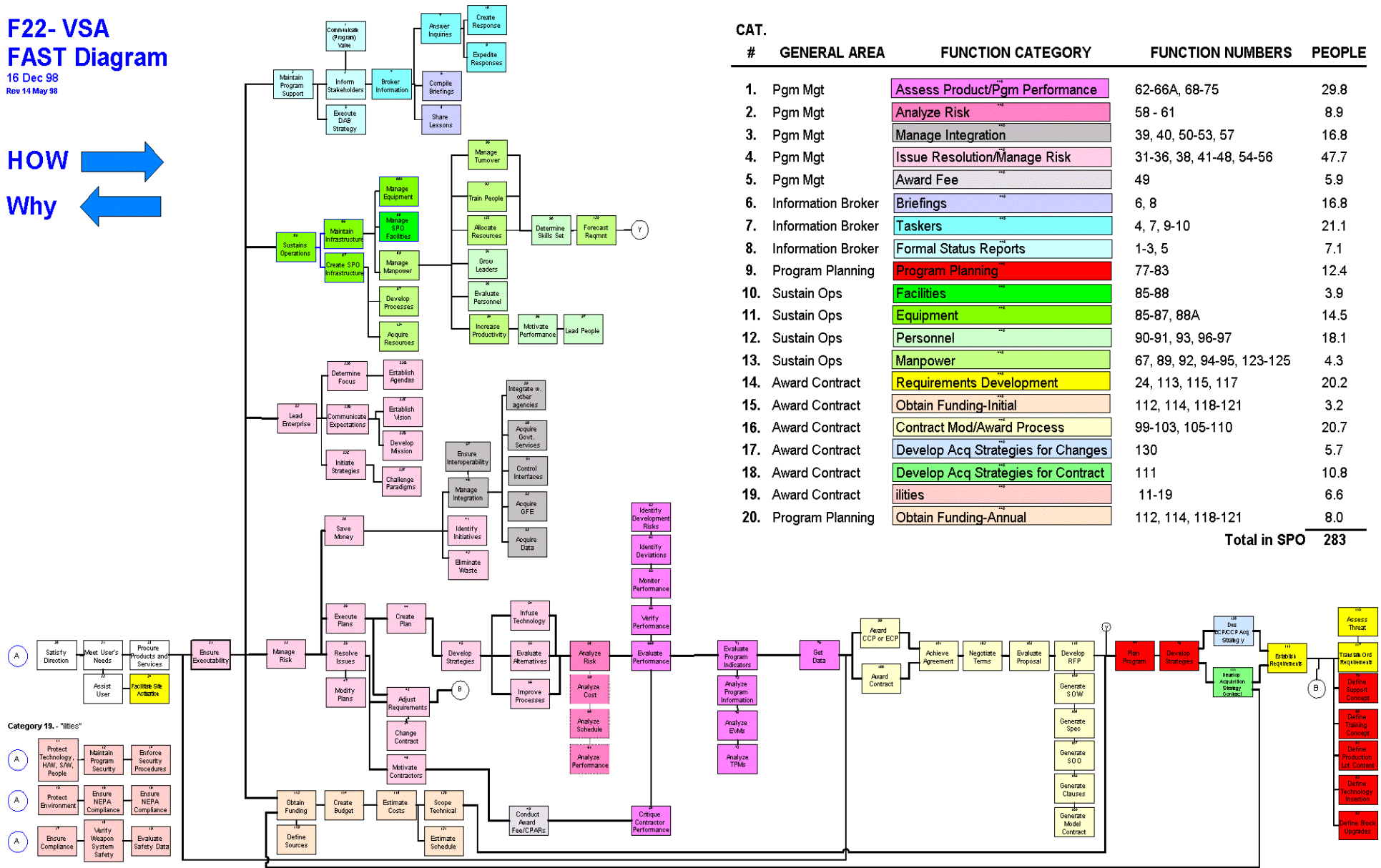


Figure 2. F-22 FAST Diagram

CONCLUSIONS:

Model builders can learn a great deal about an organization quickly with this powerful diagramming tool. The F.A.S.T. diagramming exercise will provide the system dynamics modeler multiple benefits. First, F.A.S.T. is an efficient method for defining a systems architecture. Second, completing the F.A.S.T. exercise allows the modeler and the customer greater insight into the system. Lastly, the F.A.S.T. model can enhance a modeler's credibility with their customer and can shorten the information gathering phase for model development.

For more information on learning technical F.A.S.T. see SAVE International's web site (Previously the Society of American Value Engineers) at <http://www.value-eng.org/calendar.htm>. Many VA consultants also use a similar, but vertical F.A.S.T. diagram called a customer-oriented F.A.S.T. diagram. It is more hierarchical in nature and lends itself to allocating cost, resources, and customer attitudes easier (Fowler, Value Analysis in Design, 1990, ISBN 0967921708). Fowler's book is now also in CD-ROM format.

Reference:

Bartolomei, J. and Smith, E. 2001. A System Dynamics Model of Government Engineering Support during the Development Phase of a Military Acquisition Program, 19th International Conference of The System Dynamics Society, Atlanta GA, July 23-27, 2001.

Fowler T. 1990. Value Analysis in Design. Van Nostrand Reinhold: New York.