2015 NDIA GROUND VEHICLE SYSTEMS ENGINEERING AND TECHNOLOGY SYMPOSIUM
SYSTEMS ENGINEERING (SE) TECHNICAL SESSION
AUGUST 4-6, 2015 – NOVI, MICHIGAN

AN INTEGRATIVE SYSTEMS ENGINEERING MODEL FOR PRODUCT DESIGN AND DEVELOPMENT

Adedeji B. Badiru
Anna E. Maloney

Graduate School of Engineering & Management
Air Force Institute of Technology
Wright Patterson Air Force Base, OH 45433
Correspondence Author: adedeji.badiru@afit.edu

ABSTRACT

Systems integration is crucial in highly technical products, not only for the current operational need, but also for future operations in a dynamic environment. A good case example is the various product development endeavors to support military operations. In a 2012 revelation by the U.S. Government Accountability Office (GAO), it was reported that the U.S. Air Force would spend $9.7 billion over 20 years to upgrade the capabilities of its F-22A Raptor as a result of the service's failure to anticipate the plane's long-term need for technology modernization. This is a product integration debacle. Applying a systems engineering technique could improve the systems efficiency and process effectiveness for new product development. This paper presents the **DEJI** *(Design, Evaluate, Justify, and Integrate) model as an enhancement technique that can facilitate the integration needs on the future continuum of new technological developments.*

Keywords: Acquisition life cycle, product development, learning curve, technology management

INTRODUCTION

Systems integration is crucial in highly technical products, not only for the current operational need, but also for future operations in a dynamic environment. A case example for this is the 2012 revelation by the U.S. Government Accountability Office (GAO) that the U.S. Air Force would spend **$9.7 billion** over 20 years to upgrade the capabilities of its F-22A Raptor as a result of the service's failure to anticipate the plane's long-term need for technology modernization. This is product integration gone awry. Applying a technique such as the **DEJI** (Design, Evaluate, Justify, and Integrate) model introduced by Badiru (2010, 2012) would have called attention to integration needs on the future continuum of new technological developments. Figure 1 illustrates the **DEJI** model for a product acquisitions life cycle.

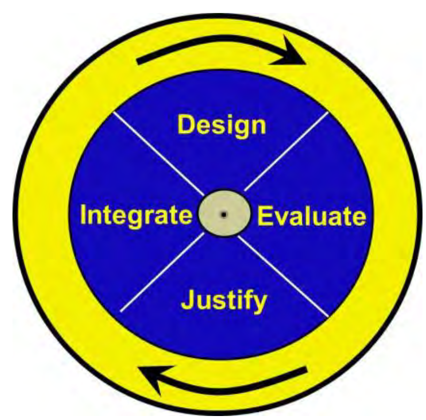

Figure 1: Stages of Implementation of **DEJI** Model for Product Development

The technique (Badiru, 2010) is unique among product development tools and techniques because it explicitly calls for a re-justification of the product within the product development life cycle. This is important for the purpose of determining when a program should be terminated even after going into production and what realignment of resources may be needed to keep the product current with new technological developments. If the program is justified, it must then be integrated and "accepted" within the ongoing business of the enterprise (Giachetti, 2010). The Department of Defense (DOD) has expressed the desire to have an integrated design and redesign of a product as it goes through its acquisition lifecycle. The **DEJI** model facilitates such a recursive design-evaluate-justify-integrate process for product evolution feedback looping. The biggest challenge for any program management endeavor is coordinating and integrating the multiple facets that affect the final outputs of a program, where a specific output may be a physical product, a service, or a desired result. Addressing the challenges of program execution from a systems perspective increases the likelihood of success. The **DEJI** model can facilitate program success through structural implementation of design, evaluation, justification, and integration. Although originally developed for product management (Badiru, 2012), the model is generally applicable to all types of endeavors because every program goes through the stages of process design, evaluation of parameters, justification of the output, and integration of the output into the core business of the organization. The model can be applied across the spectrum of the following elements of an organization:

1. People
2. Process
3. Technology

Figure 2 illustrates the integrative and hierarchical structure of the DEJI model to improve systems efficiency and process effectiveness in new technological product development.

Figure 2: Integrative Structure of **DEJI** Model for Efficiency and Effectiveness

Figure 3 illustrates the stages involved in implementing **DEJI** for product acquisition. The model is complemented by existing tools and techniques of process improvement, such as DMAIC (Define, Measure, Analyze, Improve, Control), SIPOC (Suppliers, Inputs, Process, Outputs, Customers), DRIVE (Define, Review, Identify, Verify, Execute), PDCA (Plan, Do, Check, Act), 6's (Sort, Stabilize, Shine, Standardize, Sustain, Safety), CEDAC (Cause and Effect Diagram with the Addition of Cards), and OODA loop (for Observe, Orient, Decide, and Act). Thus, **DEJI** not only addresses the product development, but also considers process improvement requirements for developing the product. The benefit that using **DEJI** provides is that it espouses using existing analytical tools and techniques for implementing product design, evaluation, justification, and integration. Selected specific analytical examples for each stage of the model are provided in the sections that follow.

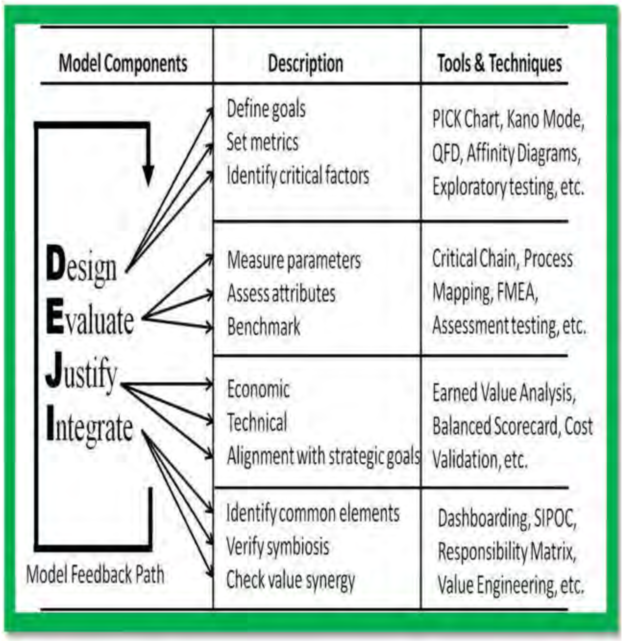

Figure 3: Tools and Techniques of Implementing **DEJI** Model

DESIGN STAGE OF DEJI

Product design should be structured to follow point-to-point transformation. A good technique to accomplish this is the use of state-space transformation, with which we can track the evolution of a product from concept stage to a final product stage. For the purpose of product acquisition, the following definitions are applicable:

Product state: A state is a set of conditions that describe the product at a specified point in time. The *state* of a product refers to a performance characteristic of the product which relates input to output such that a knowledge of the input function over time and the state of the product at time $t = t_0$ determines the expected output for $t \geq t_0$. This is particularly important for assessing where the product stands in the context of new technological developments and the prevailing operating environment.

Product state space: A product *state-space* is the set of all possible states of the product lifecycle. State-space representation can solve product design problems by moving from an initial state to another state, and eventually to the desired end-goal state. The movement from state to state is achieved through actions. A goal is a description of an intended state that has not yet been achieved. The process of solving a product problem involves finding a sequence of actions that represents a solution path from the initial state to the goal state. A state-space model consists of state variables that describe the prevailing condition of the product. The state variables are related to inputs by mathematical relationships. Examples of potential product state variables include schedule, output quality, cost, due date, resources, resource utilization, operational efficiency, productivity throughput, and technology alignment. For a product described by a system of components, the state-space representation can follow the quantitative metric below:

$$Z = f(z, x)$$
$$Y = g(z, x)$$

where f and g are vector-valued functions. The variable Y is the output vector while the variable x denotes the inputs. The state vector Z is an intermediate vector relating x to y. In generic terms, a product is transformed from one state to another by a driving function that produces a transitional relationship given by:

$$S_s = f(x \mid S_p) + e$$

where
S_s = subsequent state
x = state variable
S_p = the preceding state
e = error component (equivalent to a contingency)

The function f is composed of a given action (or a set of actions) applied to the product. Each intermediate state may represent a significant milestone in the product. Thus, a descriptive state-space model facilitates an analysis of what actions to apply in order to achieve the next desired product state. Putting the above equation into simple operational terms, gives us the type of statement below:

> Action A applied to Initial State "I" will result in Output "j," which will require so many units of resource type "k."

If the above reasoning is applied iteratively throughout the product lifecycle, all the players involved at each stage of the product will have a better understanding of what is required at each stage and how to move to the next desired stage. In a dynamic DOD type of product development, having a current view of the product will facilitate better control of the product development process. Thus, applying a quantitative assessment of the **DEJI** model to the FIST (Fast, Inexpensive, Simple, and Tiny) concept (Ward, 2012) can pave the way to realizing the much-sought-after acquisition reform in a real sense.

PRODUCT TRANSFORMATION DUE TO TECHNOLOGY CHANGES

The Weapon Systems Acquisition Reform Act (**WSARA**) and the Defense Acquisition Workforce Improvement Act (**DAWIA**) represent two of the several initiatives designed to improve the acquisition process. But to realize real and lasting improvements, which have been elusive so far, new practical approaches must be explored. The issue is not that of not having the ideas or not implementing a new strategy. It is often the lack of consistent and dedicated application of proven tools and techniques. This is where the long-standing and consistent utility of state space representation can be brought to bear on the prevailing challenges. **Figure 4** shows a representation of a product development example involving the transformation of a product from one state to another through the application of human or machine actions. This simple representation can be expanded to cover several components within the product information framework. Hierarchical linking of product elements provides an expanded transformation structure. The product state can be expanded in accordance with implicit requirements. These requirements might include grouping of design elements, linking precedence requirements (both technical and procedural), adapting to new technology developments, following the required communication links, and accomplishing reporting requirements. The actions to be taken at each state depend on the prevailing product conditions. The nature of subsequent alternate states depends on what actions are implemented. Sometimes, there are multiple paths that can lead to the desired end result. At other times, there exists only one unique path to the desired objective. In conventional practice, the characteristics of the future states can only be recognized after the fact, thus, making it impossible to develop adaptive plans. In the implementation of the **DEJI** model, adaptive plans can be achieved because the events occurring within and outside the product state boundaries can be taken into account on the fly.

	(INPUTS)	State Transformations	(OUTPUTS)
S_0	Initial condition		
T_1	Planning	→ $S_1 = T_1 (S_0)$	Product specs
T_2	Defining	→ $S_2 = T_2 (S_1)$	Problem statement
T_3	Formulating	→ $S_3 = T_3 (S_2)$	Overall function
T_4	Synthesizing	→ $S_4 = T_4 (S_3)$	Sub-function structure
T_5	Abstracting	→ $S_5 = T_5 (S_4)$	Basic operation
T_6	Varying effects	→ $S_6 = T_6 (S_5)$	Effect variants
T_7	Varying effectors	→ $S_7 = T_7 (S_6)$	Effector variants
T_8	Representing principles	→ $S_8 = T_8 (S_7)$	Solution principles
T_9	Combining	→ $S_9 = T_9 (S_8)$	Assembly variants
T_{10}	Combining	→ $S_{10} = T_{10} (S_9)$	System variants
T_{11}	Varying forms	→ $S_{11} = T_{11} (S_{10})$	Varying forms
T_{12}	Laying out	→ $S_{12} = T_{12} (S_{11})$	Qualitative layout
T_{13}	Dimensioning	→ $S_{13} = T_{13} (S_{12})$	Scale layout
T_{14}	Analyzing	→ $S_{14} = T_{14} (S_{13})$	Preliminary layout
T_{15}	Elaborating	→ $S_{15} = T_{15} (S_{14})$	Final layout
T_{16}	Detailing	→ $S_{16} = T_{16} (S_{15})$	Detail drawing
T_{17}	Production preparation	→ $S_{17} = T_{17} (S_{16})$	Production documents
T_{18}	Producing	→ $S_{18} = T_{18} (S_{17})$	Product
T_{19}	Marketing	→ $S_{19} = T_{19} (S_{18})$	Delivery to Market

Figure 4: Design Transformation due to Technology Changes

If we describe a product by *P* state variables s_i, then the composite state of the product at any given time can be represented by a vector **S** containing *P* elements. That is,

$$\mathbf{S} = \left\{ s_1,\ s_2,\ \ldots,\ s_P \right\}$$

The components of the state vector could represent either quantitative or qualitative variables (e.g., cost, energy, color, time). We can visualize every state vector as a point in the state space of the product. The representation is unique since every state vector corresponds to one and only one point in the state-space. Suppose we have a set of actions (transformation agents) that we can apply to the product status so as to change it from one state to another within the project state-space. The transformation will change a state vector into another state vector. A transformation may be a change in the raw material or a change in the design approach. The number of transformations (or actions) available for a product may be finite or countably infinite. We can construct trajectories that describe the potential states of a product evolution as we apply successive transformations with respect to technology forecasts. Each transformation may be repeated as many times as needed. Given an initial state $\mathbf{S}_0$, the sequence of state vectors is represented by the following:

$$\mathbf{S}_1 = T_1(\mathbf{S}_0)$$

$$\mathbf{S}_2 = T_2(\mathbf{S}_1)$$

$$\mathbf{S}_3 = T_3(\mathbf{S}_2)$$

$$\cdot\ \cdot\ \cdot\ \cdot$$

$$\mathbf{S}_n = T_n(\mathbf{S}_{n-1})$$

The final State, $\mathbf{S}_n$, depends on the initial state $\mathbf{S}$ and the effects of the actions applied.

EVALUATION STAGE OF DEJI

A product can be evaluated on the basis of cost, quality, schedule, and meeting requirements. There are many quantitative metrics that can be used in evaluating a product at this stage. Learning curve productivity is one relevant technique that can be used because it offers an evaluation basis of a product with respect to the concept of growth and decay. The half-life extension (Badiru, 2010) of the basic learning is directly applicable because the half-life of the technologies going into a product can be considered. In today's technology-based operations, retention of learning may be threatened by fast-paced shifts in operating requirements. Thus, it is of interest to evaluate the half-life properties of learning curves. Information about the half-life can tell us something about the sustainability of learning-induced technology performance. This is particularly useful for designing products whose life cycles stretch into the future in a high-tech environment, such as the F-22A Raptor. Figure 5 shows a graphical representation of performance as a function of time under the influence of performance decay. Technology performance degrades as time progresses. Our interest is to determine when performance has decayed to half of its original level. Figure 6 shows an example of a learning curve with the half-life indicated.

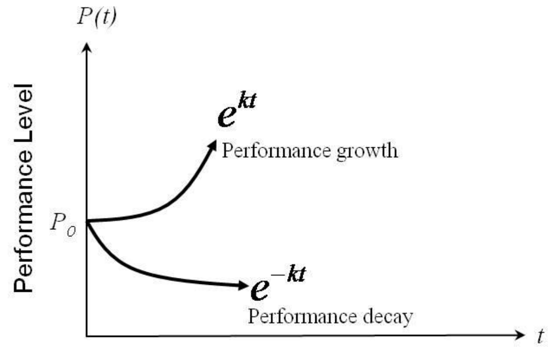

Figure 5: Concept of Learning Curve Growth and Decay

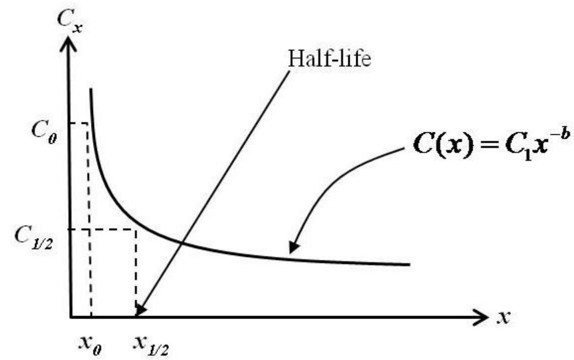

Figure 6: Profile of a Learning Curve with Half-life Point

JUSTIFCATION STAGE OF DEJI

We need to justify a program on the basis of quantitative value assessment. The Systems Value Model (SVM) is a good quantitative technique that can be used here for product justification on the basis of value. The model provides a heuristic decision aid for comparing product alternatives. Value is represented as a deterministic vector function that indicates the value of tangible and intangible attributes that characterize the product. It is represented as:

$$V = f \;\; A_1, A_2, \ldots, A_p$$

Where V is the assessed value and the A values are quantitative measures or attributes. Examples of product attributes are quality, throughput, manufacturability, capability, modularity, reliability, interchangeability, efficiency, and cost performance. Attributes are considered to be a combined function of factors. Examples of product factors are market share, flexibility, user acceptance, capacity utilization, safety, and design functionality. Factors are themselves considered to be composed of indicators. Examples of indicators are debt ratio, acquisition volume, product responsiveness, substitutability, lead time, learning curve, and scrap volume. By combining the above definitions, a composite measure of the operational value of a product can be quantitatively assessed. In addition to the quantifiable factors, attributes, and indicators that impinge upon overall product value, the human-based subtle factors should also be included in assessing overall value.

CONTEMPORARY EARNED VALUE TECHNIQUE

A companion analytical technique to use for the justification stage is the conventional earned value technique (EVT), which can be used for cost, quality, and schedule elements of product development with respect to value creation. The technique involves developing important diagnostic values for each schedule activity, work package, or control element as shown in Figure 7. The definitions of the variables in the figure are summarized below:

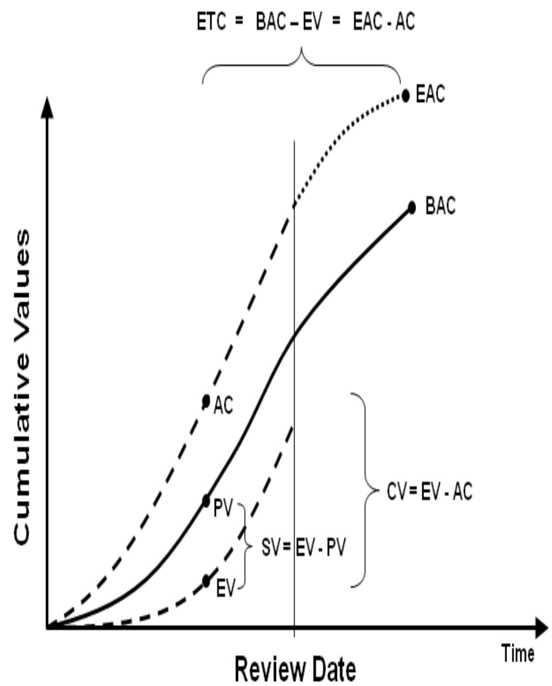

Figure 7: Graphical Plot of Earned Value Performance Analysis

Planned Value (PV): This is the budgeted cost for the work scheduled to be completed on an activity up to a given point in time.

Earned Value (EV): This is the budgeted amount for the work actually completed on the schedule activity during a given time period.

Actual Cost (AC): This is the total cost incurred in accomplishing work on the schedule activity during a given time period. AC must correspond in definition, scale, units, and coverage to whatever was budgeted for PV and EV. For example, direct hours only, direct costs only, or all costs including indirect costs. The PV, EV, and AC values are used jointly to provide value performance measures of whether or not work is being accomplished as planned at any given point in time. The common measures of project assessment are cost variance (CV) and schedule variance (SV).

Cost Variance (CV): This equals earned value minus actual cost. The cost variance at the end of the project will be the difference between the budget at completion (BAC) and the actual amount expended.

$$\mathbf{CV = EV - AC}$$

Schedule Variance (SV): This equals earned value minus planned value. Schedule variance will eventually become zero when the project is completed because all of the planned values will have been earned.

$$\mathbf{SV = EV - PV}$$

Cost Performance Index (CPI): This is an efficiency indicator relating earned value to actual cost. It is the most commonly used cost-efficiency indicator. CPI value less than 1.0 indicates a cost overrun of the estimates. CPI value greater than 1.0 indicates a cost advantage of the estimates.

$$\mathbf{CPI = \frac{EV}{AC}}$$

Cumulative CPI (CPI^C): This is a measure that is widely used to forecast project costs at completion. It equals the sum of the periodic earned values (Cum. EV) divided by the sum of the individual actual costs (Cum. AC).

$$\mathbf{CPI^C = \frac{EV^C}{AC^C}}$$

Schedule Performance Index (SPI): This is a measure that is used to predict the completion date of a project. It is used in conjunction with CPI to forecast project completion estimates.

$$\mathbf{SPI = \frac{EV}{PV}}$$

Estimate to complete (ETC) based on new estimate: Estimate to complete equals the revised estimate for the work remaining as determined by the performing organization. This is an independent non-calculated estimate to complete for all the work remaining. It considers the performance or production of the resources to date. The calculation of ETC uses two alternate formulae based on earned value data.

ETC based on atypical variances: This calculation approach is used when current variances are seen as <u>**atypical**</u> and the expectations of the project team are that similar variances will <u>not</u> occur in the future.

$$\mathbf{ETC = BAC - EV^C},$$

where BAC = Budget at completion.

ETC based on typical variances: This calculation approach is used when current variances are seen as <u>**typical**</u> of what to expect in the future.

$$\mathbf{ETC = \frac{BAC - EV^C}{CPI^C}}$$

Estimate at completion (EAC): This is a forecast of the most likely total value based on project performance. EAC is the projected or anticipated total final value for a schedule activity when the defined work of the project is completed. One EAC forecasting technique is based upon the performing organization providing an estimate at completion. Two other techniques are based on earned value data. The three calculation techniques are presented below. Each of the three approaches can be effective for any given project because it can provide valuable information and signal if the EAC forecasts are not within acceptable limits.

EAC using a new estimate: This approach calculates the actual costs to date plus a new ETC that is provided by the performing organization. This is most often used when past performance shows that the original estimating assumptions were fundamentally flawed or that they are no longer relevant due to a change in project operating conditions.

$$\mathbf{EAC = AC^C + ETC}$$

EAC using remaining budget: In this approach, EAC is calculated as cumulative actual cost plus the budget that is required to complete the remaining work; where the remaining work is the budget at completion minus the earned value. This approach is most often used when current variances are seen as <u>**atypical**</u> and the project management team expectations are that similar variances will not occur in the future.

$$\mathbf{EAC = AC^C + (BAC - EV)},$$

where (BAC – EV) = remaining project work = remaining PV.

EAC using cumulative CPI: In this approach, EAC is calculated as actual costs to date plus the budget that is required to complete the remaining project work, modified by a performance factor. The performance factor of choice is usually the cumulative CPI. This approach is most often used when current variances are seen as <u>**typical**</u> of what to expect in the future.

$$\mathbf{EAC = AC^C + \frac{(BAC - EV)}{CPI^C}}$$

Other important definitions and computational relationships are summarized below:

Earned ➔ Budgeted cost of work actually performed
Planned ➔ Budgeted cost of work scheduled
Actual ➔ Cost of actual work performed

Ending CV = Budget at completion – Actual amount spent at the end
= BAC – EAC = VAC (Variance at Completion)

EAC= ETC + AC = (BAC – EV) + AC = AC + (BAC – EV)

ETC= EAC – AC = BAC - EV

INTEGRATION STAGE OF DEJI

Without being integrated, a system will be in isolation and it may be worthless. We must integrate all the elements of a system on the basis of alignment of functional goals. The overlap of systems for integration purposes can conceptually be viewed as projection integrals by considering areas bounded by the common elements of sub-systems as shown in Figure 8. Quantitative metrics can be applied at this stage for effective assessment of the product state.

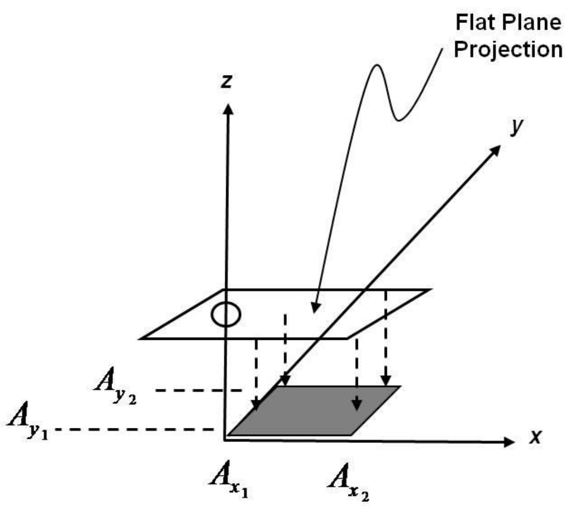

Figure 8: Technology Alignment Surface for Product Integration

Presented below are guidelines and important questions relevant for product integration.

- What are the unique characteristics of each component in the integrated system?
- How do the characteristics complement one another?
- What physical interfaces exist among the components?
- What data/information interfaces exist among the components?
- What ideological differences exist among the components?
- What are the data flow requirements for the components?
- What internal and external factors are expected to influence the integrated system?
- What are the relative priorities assigned to each component of the integrated system?
- What are the strengths and weaknesses of the integrated system?
- What resources are needed to keep the integrated system operating satisfactorily?
- Which organizational unit has primary responsibility for the integrated system?

HYPOTHETICAL CASE EXAMPLES

These hypothetical examples are presented to illustrate the diverse array of potential applications of the **DEJI** model.

Example 1: Classified Information Database

Researchers at the CIA want to come up with a new database in which they can store their enormous amount of classified information without fear of theft by adversaries. Using the **DEJI** model, the CIA is able to choose whether or not a new database is effective and efficient.

Design- The CIA's goal is to create a secure database for top secret information. The most important aspects of this new system is that it is completely secure and can hold large amounts of information that can be easily reached by CIA employees.

Evaluate- During the research and development process of this new database, the computer scientists found new ways to create a computer system. The CIA decided that this endeavor was worth pursuing because of the researchers' great advancements thus far.

Justify- The CIA has a large budget for technological advancements, thus this project was deemed a great investment. Technically, this project was advanced and proved safe for classified information. It aligned perfectly with the strategic goals of the CIA.

Integrate- Integration was a major problem in the process of developing a new secure database. The CIA realized that it would take months to transport all of their data into the new system, which would severely deteriorate the efficiency of the CIA and put the USA at a higher risk from adversaries during that time. Also, the analysts at the CIA who used the new database did not enjoy how the system worked and deemed it unsuitable for their tasks.

Example 2: App to Learn a New Language

The Department of Defense (DOD) asked developers if they could make an app so DOD employees could learn a new language quickly and effectively. Using the **DEJI** model, the DOD is able to keep an eye on the project at every step.

Design- The DOD wishes the create an app that allows DOD employees to both learn ne languages and keep language skills sharp. They request that the app be user friendly for people of ages 20-70. This is a tall order as people of different ages learn at different rates and are accustomed to an array of technology. They want the new language program to be less expensive than other language programs on the market. The developers also asked linguists and language teachers to assist in the making of the product to ensure the use of effective learning techniques.

Evaluate- During the evaluation process, the developers looked at other language programs to see how those programs adapt to the changing technological environment. For example, 15 years ago, people did not have smart phones, and the recent emergence of these devises have made dissemination of computer programs much quicker and widespread. The DOD developers decided to make the language app simpler to ensure it could evolve to the changing times.

Justify- Because the federal government has begun emphasizing the need for foreign language skills, the DOD feels it is acceptable to spend time and money developing a language learning app for its employees. The DOD also realized during the design and evaluation stages that it could choose vocabulary that was pertinent to DOD employees. During the justification process, the DOD looked at how much they had planned to spend on the project and how much they had spent already, and realigned their budget to allow for more time to perfect the app.

Integrate- The DOD did a fantastic job of disseminating their new language learning app to its employees. The app was very user friendly, while still enticing to use. People of all ages were able to choose their best learning style at the beginning, allowing the app to adjust to the needs of the learner. The DOD realized, however, that this app would need near constant supervision due to over-usage crashes.

Example 3: Surveillance Vans

The FBI decided that a new fleet of surveillance vans were needed in order to protect American citizens from homegrown terrorist cells. Using the **DEJI** model, the FBI was able to create a fleet of vehicles conducive to their needs.

Design- During the design process of the surveillance vans, the developers realized that there was a new police van in production that could be outfitted for surveillance. The FBI wanted a vehicle large enough to fit four agents and all of their equipment, and the new police van had more than enough space. The design process continued.

Evaluate- The cost, quality, and schedule of the project were all on point. The new surveillance vans had a great life expectancy and were able to be built quickly and inexpensively.

Justify- The justification stage of this process saw a CPI value greater than 1.0, indicating a cost advantage of the estimates. Because the design tem utilized technology from other branches of the government, they were able to save both time and money in the design process. The FBI decided to continue with the surveillance vans into distribution.

Evaluation- Even though the three previous steps of the **DEJI** model pointed at a successful integration of the surveillance vehicles into the FBI fleet, FBI agents complained that the new vehicles were very easily distinguishable from regular vehicles. This obviously is the antithesis of what a surveillance vehicle is supposed to be. The aesthetic of the vehicle made it unsuitable for its original process, even though the new van had ample room and wonderful technology. The police force, though, did very much enjoy the modifications and was able to use the vehicle for different purposes.

CONCLUSION

Recent cost over-run and technology misalignment reports on some DOD product development programs have necessitated the search for alternate or additional techniques for assessing technical products. This has presented an additional robust technique that can complement existing efforts. The paper presents an application of the **DEJI** product development model as a complementary approach to executing new technology-based product acquisition. The proposed approach will facilitate a better alignment of product technology with future development and needs. The stages of the model involve design, evaluation, justification, and integration. Existing analytical tools and techniques are presented for implementing the stages. The methodology of the paper adds to the repertoire of tools and techniques available to DOD acquisitions community.

REFERENCES

1. Badiru, Adedeji B. (2010), "Half-life of Learning Curves for Information Technology Project Management," *International Journal of IT Project Management*, 1(3), pp. 28-45.
2. Badiru, A. B. (2012), "Application of the DEJI Model for Aerospace Product Integration," *Journal of Aviation and Aerospace Perspectives (JAAP),* Vol. 2, No. 2, pp. 20-34, Fall 2012.
3. Giachetti, Ronald E. (2010), **Design of Enterprise Systems: Theory, Architecture, and Methods**, Taylor and Francis/CRC Press, Boca Raton, FL.
4. Ward, Dan (2012), "Faster, Better, Cheaper: Why not pick all three," ***National Defense Magazine***, April 2012, pp. 1-2.

AUTHORS BIOGRAPHICAL SKETCHES

Adedeji Badiru is head of Systems and Engineering Management at the Air Force Institute of Technology. He is a registered professional engineer (PE), a certified Project Management Professional (PMP), and a fellow of the Institute of Industrial Engineers.

Anna Maloney is an intern at the Air Force Institute of Technology. She is currently attending The Ohio State University where she is pursuing degrees in International Security & Intelligence and Russian.

Unclassified. This paper is cleared for public release. Distribution Unlimited.

Disclaimer:

The views expressed in this article are those of the authors and do not reflect the official policy or position of the Air Force Institute of Technology, United States Air Force, Department of Defense, or United States government.