Skip all front matter: Jump to Page 1

R E P O R T

Improving Air Force Depot Programming by Linking Resources to Capabilities

Don Snyder, Julie Kim, Manuel Carrillo,
Gregory G. Hildebrandt

Prepared for the United States Air Force

Approved for public release; distribution unlimited

RAND PROJECT AIR FORCE

The research described in this report was sponsored by the United States Air Force under Contract FA7014-06-C-0001. Further information may be obtained from the Strategic Planning Division, Directorate of Plans, Hq USAF.

Library of Congress Cataloging-in-Publication Data

Improving Air Force depot programming by linking resources to capabilities / Don Snyder ... [et al.].
p. cm.
Includes bibliographical references.
ISBN 978-0-8330-5967-3 (pbk. : alk. paper)
1. United States. Air Force—Appropriations and expenditures. 2. United States. Air Force—Equipment—Maintenance and repair—Costs—Evaluation. 3. United States. Air Force—Supplies and stores—Costs—Evaluation. I. Snyder, Don, 1962-

UG633.2.I63 2012
358.4'18—dc23

2012020058

Published 2012 by the RAND Corporation
1776 Main Street, P.O. Box 2138, Santa Monica, CA 90407-2138
1200 South Hayes Street, Arlington, VA 22202-5050
4570 Fifth Avenue, Suite 600, Pittsburgh, PA 15213-2665
RAND URL: http://www.rand.org
To order RAND documents or to obtain additional information, contact
Distribution Services: Telephone: (310) 451-7002;
Fax: (310) 451-6915; Email: order@rand.org

Summary

Current planning and programming policies with a capabilities-based focus have spawned an intense interest in how to discern, define, and quantify Air Force capabilities. Capabilities-based programming has meant different things to different organizations, but the core spirit of the policy on which we focus is that it aims to provide a programmed force ready to meet a range of possible contingencies given an uncertain future. Previous RAND Project AIR FORCE research presented a framework for defining capability metrics and, using these metrics and methods, developed models for expeditionary combat-resource programming and budgeting.[1]

This work builds on and expands that previous research to embrace programming for depot-level maintenance, as illustrated by DPEM commodities. DPEM is a budgeting area that comprises numerous depot-level maintenance and inspection activities on capital assets, as well as storage and other supporting activities that span numerous program elements, budget programs, and appropriations. It funds the bulk of the work done at the Air Force depots that does not involve the repair of spares or the work on modification programs.

The challenges of relating money spent to capabilities rendered for expeditionary combat support resources and depot maintenance are quite different. For the former, the challenge lies in identifying resources needed to carry out the various planning scenarios and then determining the appropriate resource levels to meet those plans. In contrast, most depot-level maintenance is done on large capital items. The *resource levels* are already determined; the challenge is to prioritize the *allocation of sustainment funds* across weapon systems and to accurately quantify the risks engendered by various funding levels.

A first step in resolving these challenges is to define appropriate capability metrics. We sought metrics possessing three attributes: (1) that they relate directly to U.S. Department of Defense (DoD)–level planning objectives; (2) that they relate to program elements, subsets of program elements, or sets of program elements in a clearly understandable manner; and (3) that they apply across a wide range of resources.

Two metrics that satisfy these criteria and span most of depot-level maintenance are the ability to generate aerospace expeditionary force (AEF) and training sorties and the availability of Minuteman III intercontinental ballistic missiles (ICBMs). These metrics are broad enough to encompass many program elements and yet still relate resources to DoD-level planning objectives in a clear, quantitative manner. These metrics can also embrace other useful, lower-level submetrics. One example is aircraft availability, which is a submetric of the ability to generate AEF and training sorties.

1 Snyder et al., 2009.

We have developed a prototype model and associated software tools that use aircraft availability as a metric. This model is designed to give guidance to a programmer on how to best distribute offsets among the aircraft and engine commodities of DPEM.[2] This model and its associated tools are designed with a view to be expanded in the future to include broader metrics, such as sortie generation, and to embrace a wider range of commodities.

Central to the model is a means of prioritizing across disparate weapon systems. By selecting metrics that relate to DoD-level plans, the planning objectives provide a natural prioritization. Weapon systems with high demands in plans relative to the number in inventory have a high weighting in prioritization; those with low demands in plans relative to the number in inventory have a low weighting. The caveat is that prioritization solely according to the operator's priorities will lead to nearly all the risk being placed on a small number of aircraft types. For example, if there were one mission design series that had an inventory higher than called for in war plans, a model that considers only the war plans for prioritization might suspend nearly all the maintenance on that aircraft type. Such a distribution of risk is unacceptable because of the need for training to maintain readiness.

To address the competing needs to meet warfighter priorities and to meet training requirements, the model provides a sliding rheostat for the user to select a position in the trade space between a prioritization according to operational plans and one that is a proxy for meeting peacetime training needs. Other inputs to the model are the depot induction requirements for each item by year of the Future Years Defense Program (FYDP), the topline budget for each year of the FYDP for the assets of interest, weighting factors for weapon system prioritization extracted from planning scenarios, costs of production for each depot induction, a penalty cost for deferring work, the maximum proportion of the work that can be deferred for each item, air logistic center capacities for each production task for each year in the FYDP, and force structure for each year in the FYDP.

The output of the model gives three insights. The first is how to prioritize—based on the weighting factors for weapon systems—and distribute the allocated workload and deferrals across the FYDP to avoid aircraft groundings. These outputs are the production levels for each year, what work is deferred, how much capacity is used, and how much all of this costs for each year in the FYDP. The optimal allocation will be such that no aircraft are grounded and that all work is performed by the end of the FYDP.

A second insight is how much the budget can be reduced in one or more years in the FYDP and still be able to execute the depot work at some time within the FYDP without grounding an aircraft, subject to the user-specified constraints. In this exercise, the user can decrease the budgets available in one or more years of the FYDP until the model returns an infeasible result. An infeasible result means that the program cannot be executed without grounding an aircraft. The user can then explore which constraints cause this problem (e.g., insufficient budget, lack of capacity) and which aircraft are the first to be grounded.

A third insight elucidates the operational consequences of a proposed POM. Given the underlying data used to construct the weighting factors, the resulting aircraft availability from the proposed programmed depot work can be used to see how many of each aircraft are available relative to those needed to execute the plans. This calculation gives a direct impact of the depot work on aircraft and engines in terms of operational priorities. The programmer

---

[2] *Offsets* are funds moved from one approved program to another approved program because the benefiting program is considered to be a higher priority than the paying program.

can explore risk expressed in these terms while varying any of the above inputs, including via the sliding rheostat that adjusts in the continuum between operational plans and training priorities.

We designed this prototype set of software tools to assist the programmer in using capability metrics to make balanced trades across weapon systems and to express quantitatively the risk those decisions carry in terms of the ability to meet DoD-level planning objectives. Unfortunately, some depot-maintenance business areas elude expression using the above metrics. One such area, which costs roughly $600 million per year, is software maintenance.

In the course of examining the numerous ways in which software maintenance challenges conventional operational metrics, we identified a number of areas open for improvement and recommend (1) clearer policy guidelines and standards for the baseline for operational acceptance of functional capability and better documentation of the software baseline; (2) additional refinements, where practical, in classifying software maintenance activities on the basis of what operational impacts can be identified; (3) standardization in data documentation across weapon systems; (4) more rigorous statistical analysis of historical data; and (5) additional analyses to establish operationally relevant metrics where possible and to identify the full extent of when analysis based on such metrics is appropriate.

Software maintenance serves as an important case study, but it also reveals four attributes that make capabilities-based programming so challenging that are also characteristic of other sustainment areas, such as sustaining engineering and technical orders. These are (1) not being easily related to operational objectives, such as sortie generation; (2) having a long lag time between funding and any operational impact; (3) whatever impact occurs does so across the entire fleet, not individual tail numbers; and (4) possessing a certain ambiguity in what constitutes a requirement. We entertained three policy options for programming these difficult business areas. We used the first two to make broader points but advocate only the last.

The first option highlights that policy decisions often drive ostensible operational consequences of sustainment. The linkage between a task, such as an inspection in a programmed depot maintenance (PDM) task list, to an operational metric, such as aircraft availability, is no more obvious than the linkage of a software change request to this metric. The consequences that not performing some PDM task would have for aircraft availability ensue from bundling the tasks together as a PDM, to be all done or all deferred. If the task is deferred, policies dictate how long until the aircraft is no longer certified to fly. Would it make sense to bundle software change requests into a task list, such as a PDM, for up or down inclusion in the budget?

Two factors make such policies as grounding a fleet for not performing software maintenance requests untenable. One is that these requests originate from the users of the weapon systems, not from engineers certifying the safety of the fleet, and hence do not carry the weight to warrant groundings. But more importantly, the impact of deferring software and related maintenance cannot be attributed to individual aircraft but must be assigned to an entire fleet. It is a very coarse, severe judgment that declares an entire fleet either available or unavailable given a diminished, but difficult-to-define, ability to carry out the designed mission.

A second option is to introduce a metric for the overall long-term health of a weapon system to complement metrics that reflect the near-term readiness of the fleet, such as aircraft availability. Each task to be funded would be assigned a score depending on the probability of occurrence of the problem it is meant to rectify and the potential impact to the mission should that problem occur. Conceptually, such a long-term health metric would be informative; in practice, it is not clear how to estimate the probabilities of problem occurrence and criticality to the mission.

A third option, and the one that we find the most promising, is a radical departure from the current paradigm of capabilities-based programming. The idea emerges from a recognition that the ability to perform any type of software change at any time on any system is more critical than any individual routine software change request. Hence, rather than evaluating risk to a program by appraising the relative merits of unfunded requests, risk would be assessed by the impact of funding on the capabilities of handling any possible emergency or urgent request related to any part of the software or interface with hardware. Software, and related areas, would be viewed as capabilities themselves, and thereby assessed in terms of readiness.