NPS-AM-10-048

EXCERPT FROM THE PROCEEDINGS

OF THE

SEVENTH ANNUAL ACQUISITION RESEARCH SYMPOSIUM WEDNESDAY SESSIONS VOLUME I

Acquisition Research

Creating Synergy for Informed Change

May 12 - 13, 2010

Published: 30 April 2010

Prepared for: Naval Postgraduate School, Monterey, California 93943

ACQUISITION RESEARCH PROGRAM
GRADUATE SCHOOL OF BUSINESS & PUBLIC POLICY
NAVAL POSTGRADUATE SCHOOL

The research presented at the symposium was supported by the Acquisition Chair of the Graduate School of Business & Public Policy at the Naval Postgraduate School.

To request Defense Acquisition Research or to become a research sponsor, please contact:

NPS Acquisition Research Program
Attn: James B. Greene, RADM, USN, (Ret.)
Acquisition Chair
Graduate School of Business and Public Policy
Naval Postgraduate School
555 Dyer Road, Room 332
Monterey, CA 93943-5103
Tel: (831) 656-2092
Fax: (831) 656-2253
E-mail: jbgreene@nps.edu

Copies of the Acquisition Sponsored Research Reports may be printed from our website www.acquisitionresearch.net

Queues in Acquisition

William Wiltschko

Abstract

Acquisition programs have inherent variability in their task durations, which often results in unforecasted completion delay. Using concepts from Lean Production and Lean Product Development, queues that are at the heart of these delays can be made visible and can be managed. Observing queues in acquisition programs can give early warning of project problems. Several techniques can be used to manage queues.

Keywords: Queues, queueing theory, acquisition, product development, lean product development, cost of delay, utilization

Introduction

This paper is intended to be an introduction to a portion of a large subject. Conferences, scores of books, hundreds of papers, and uncounted consultants have been devoted to product development in the public and private sectors. Issues such as configuration management tools, quality of the IMS (Integrated Management Schedule), domain-specific considerations, and people management—all important issues—will not be treated here. This paper focuses on a topic that may not be as well known as other product development topics, but I believe has great potential to better manage acquisition programs.

Queues are generally unrecognized entities in acquisition programs, yet they are valuable information sources and useful handles for controlling them. Lean manufacturing experts have long viewed queues as near-evils to be managed in a production environment, but only relatively recently have they viewed them as either problems or opportunities in product development. There is now a rich literature on lean production (Ohno, 2008) and lean product development (Morgan & Liker, 2006) that relies on insights gained from queueing theory.

Queues are easy to recognize in a production environment—piles of physical product in front of a workstation or machine make them obvious. What is a queue in acquisition, where the thing being manufactured, at least in the early stages, is only information? In this paper, I take the point of view of the Program Manager (PM) or PM leadership and focus on those acquisition phases having a project orientation.

Recognizing a Queue

Figure 1 shows a project overview. The top line shows how many tasks have been *started* since the beginning of the project as of the date on the *x* axis. The bottom line shows how many tasks have been *finished* since beginning of the project as of the date on the *x* axis. Thus, in period 13, there are 15 projects that have been started since the beginning and 10 projects that have been finished since the beginning, leaving 5 projects in the queue.

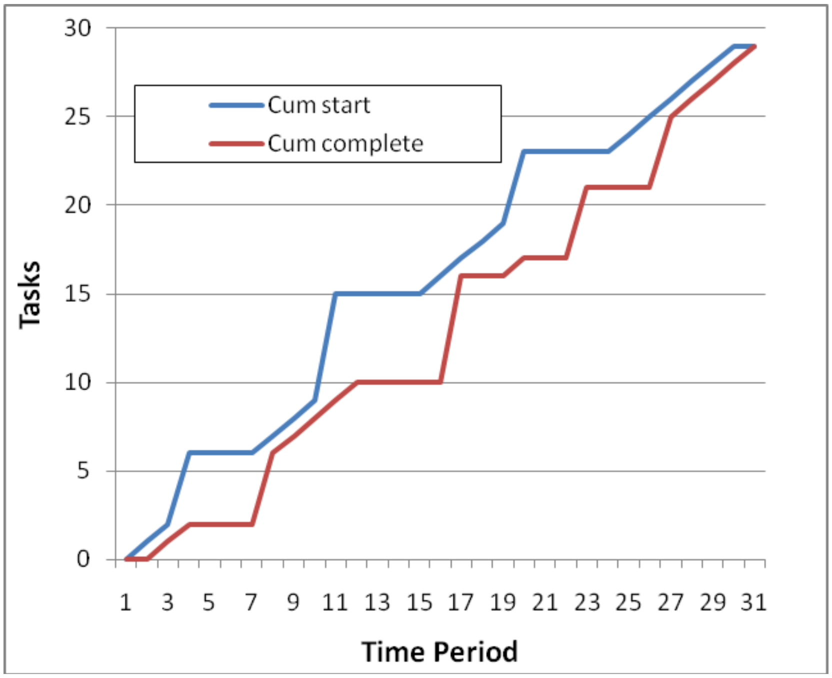


Figure 1. Where is the Queue?

Queues arise whenever there are unfinished tasks. Thus, some amount of queueing is inevitable. In the figure, the only points where the lines meet (where the queue has disappeared) are at the beginning and at the end. However, when the number of unfinished tasks is large, queues are large. In the figure, the gap between cumulative started and cumulative finished tasks is the queue size. Note that both the vertical (quantity) and horizontal gaps (time) grow for increasing queues. The key fact to note is that queue size will increase well before the task completion dates *prove* that the schedule is slipping. Thus, queue size is a leading indicator of schedule slips.

This graph can be created using the program management tool to create a scatter plot of numbered task actual start dates and actual finish dates, sorted by actual start date.

In the graph, there are a few points where the queues are dramatically reduced. This occurs when the cumulative complete line jumps up after going horizontal for some time (approximately periods 8, 17, and 24). These points correspond to authorization points such as milestone decisions. Here the queue arises not only because it takes some time to complete several tasks—in synchrony—but also because the milestone meeting may not occur immediately after the tasks are complete. The milestone decision meeting may be delayed. While teams will not completely stop work while they wait for the milestone decision, the milestone decision may render speculative work irrelevant.

What Makes Queues Large?

The factors that make queues larger are longer task durations, the number of tasks being worked on simultaneously, waiting for completion of other dependent tasks, and waiting for task or project reviews. Going a level deeper, these factors are caused by not breaking down tasks into small-enough chunks, multi-tasking key people (thus spreading them too thin), poor metrics that do not allow queues to be better managed, insufficient parallelization of tasks when staff is adequate to support more simultaneous tasks, and infrequent review meetings that increase team wait time.

A pernicious kind of queue is created by rework. Rework is usually not visible in common project management software. In particularly risky acquisitions, where new science and engineering knowledge are being developed, rework is inevitable, and is sometimes represented as a finite number (i.e., a guess) of iterations of a set of tasks. Other reasons that rework occurs is when it is due to team directions that are either under- or over-specified, when testing is delayed, or when authorization reviews do not take place regularly.

A more fundamental cause of large queues arises from the variable nature of work that dominates a typical acquisition. Task durations can only be approximately estimated, and duration varies widely among different tasks. Variability in both estimated and actual durations produces unexpected, non-obvious task duration (cycle time) increases, which in turn increases queues. Figure 2 illustrates this phenomenon The two curves result from two different values for coefficient of variation. The more variation there is in task duration, the more that cycle time tends to “blow up” with increasing utilization.

This phenomenon is well known to practitioners of lean *production*. Their normal response, as opposed to the response of lean *product development* practitioners is to aggressively reduce variability. This is often not an option in acquisitions that require knowledge work, such as science and engineering. Variability is inherent in knowledge work, so other approaches must be used to make an impact on project cycle time and queues.

How Do You Measure Queues?

Unlike estimated schedule completion, queues are measured with actual data. Their size is the accumulated person-hours actually spent on *started* but *unfinished* tasks. This number can be calculated from most project management software if incurred person-hours are entered into the tool.

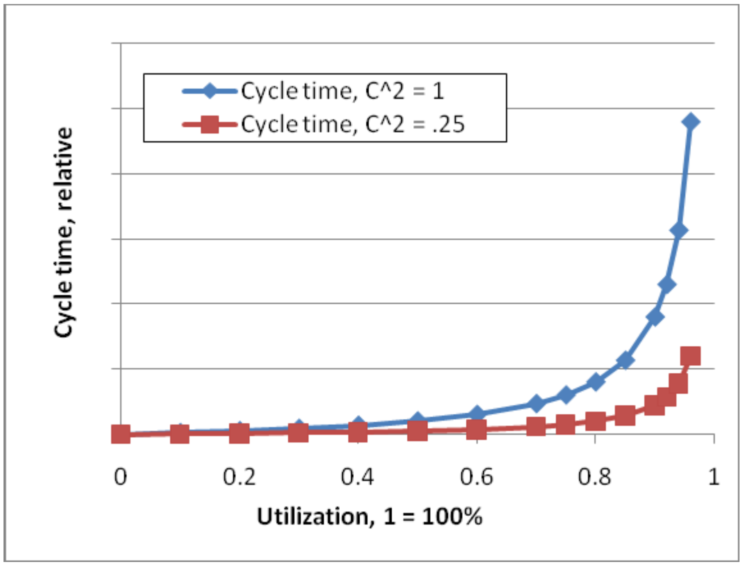


Figure 3. C

Cycle Time versus Utilization
(Hopp & Spearman, 1996)

Task	Start	Actual Finish	Projected Finish	WIP
Program Budget Cycle 2010 (Apr '08-Sept '09)	4/15/2008	9/30/2009	9/30/2009	0
Budget Execution Cycle 2010 (Jul '09 - Nov '10)	6/29/2009		12/1/2010	270
NCCA to review the draft CARD	3/19/2010		6/25/2010	7
NCCA Develop ICE	3/30/2010		7/7/2010	0

Figure 4. Measuring Queues from IMS

Above is a simple example taken from a typical IMS. For simplicity, only two started-but-unfinished tasks are shown, with 270 and 7 workdays invested in the two tasks. Workdays from any additional started-but-unfinished tasks would simply be added to 277. This value would be valid only on the day that this data is recorded. By recording this data every week and graphing it, queue size and trends would become apparent.

However, for very large programs, there may be scores of open tasks, and not very timely accounting for actual hours spent. Lack of timely data entry defeats the purpose of providing early warning, but there is an easier way to providing nearly the same information. Tracking only workdays (without regard to how many people are working on each task) spent on started-but-unfinished tasks provides a good substitute.

Figure 4 is a graph of queue task-periods for the graph shown in Figure 1. In other words, they have been calculated for every period in the project rather than just one period as in Figure 3. Figure 4 shows the queue in period 13 growing above the previous maximum. This is early warning that work may not be completed as scheduled. While the cause may be long duration tasks and not late tasks, the graph provides triggers to ask questions about what is going on. The height of the curve in Figure 4 represents the *area* between the two lines in Figure 1.

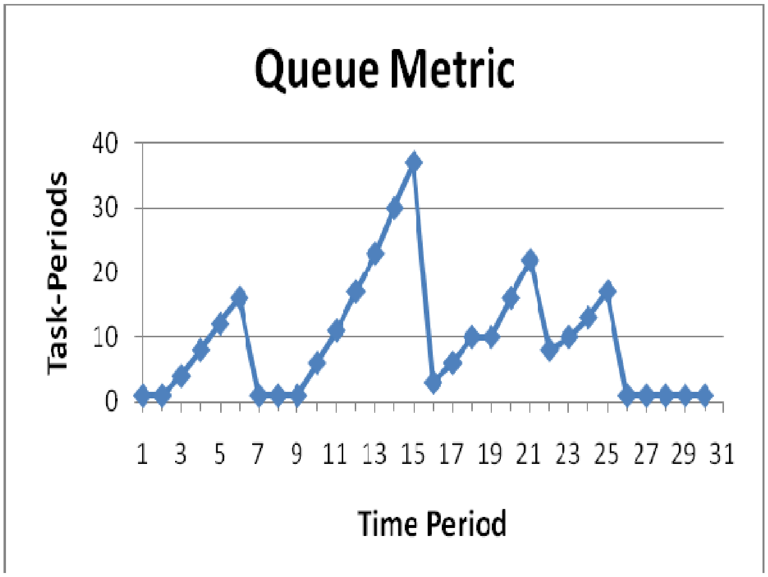


Figure 5. Size of Queue in Task-Periods

The beauty of this metric is that it can be calculated from PMO (Program Management Office) IMS data with little extra work. In other words, no matter the size of the program, this metric is easily calculated and tracked.

How Can We Reduce Queues?

Three general measures can be taken to reduce queues:

- Manage demand,
- Increase capacity, and
- Project management.

Demand can be managed via requirements management. Most requirements' development processes bucket requirements into "must haves" versus "nice to haves." This can be expanded to ranking (possibly by dollarizing) requirements so that when a schedule slip with a given set of requirements looks likely, there is a list of "nice to have" requirements that can be jettisoned in rank order. The key is to *rank order* requirements. The program would then have a requirements relief valve.

Increasing capacity will reduce utilization, thereby reducing queues and cycle time (see Figure 2). Capacity can be increased by staff additions or staff adaptation—that is, intelligently and dynamically allocating staff. Many programs assume that only IMS people do IMS, only acquisition people do acquisition planning, only manpower people do manpower planning, etc. Using people with multiple domain capabilities can help increase utilization and decrease cycle time. They can either be teams of senior people who move from function to function as problems crop up, or teams of junior people who may not need as much domain-specific knowledge to change function and still perform adequately in a reduced role. These teams are sometimes called SWAT teams or tiger teams.

Understanding queues gives the program manager extra tools. First, as mentioned above, queues are a leading indicator of program health. Second, developing expectations of where specific queues are likely to occur makes it possible to prevent them, not just fix them. For example, task parallelization can be concentrated on the potentially longest (riskiest) queues, and efforts can be made to move these potential queues from the critical path. Third, as mentioned in the paragraph above, SWAT-like teams can be constructed with the right skill sets for expected queues. Fourth, with the right economic guidelines, which we will discuss next, the program manager can respond quickly to rapidly developing queue problems. Finally, the program manager and his or her leadership can schedule reviews of the program both internally and externally on a frequent, regular basis (a cadence) so that queues don't build while waiting for a decision.

Quick Response Based on Solid Economic Guidelines

Controlling queues means getting the right resources put on solving real-time design and planning problems. This requires knowing what level of resources is reasonable to apply to the problem. This can only be done well if tradeoff guidelines *based on data* are created at the beginning.

Figure 5 shows these tradeoff guidelines. There are three tradeoff ratios:

1. Between development cost and procurement cost (or cost to field)
2. Between procurement cost and cost of delay
3. Between development cost and cost of delay

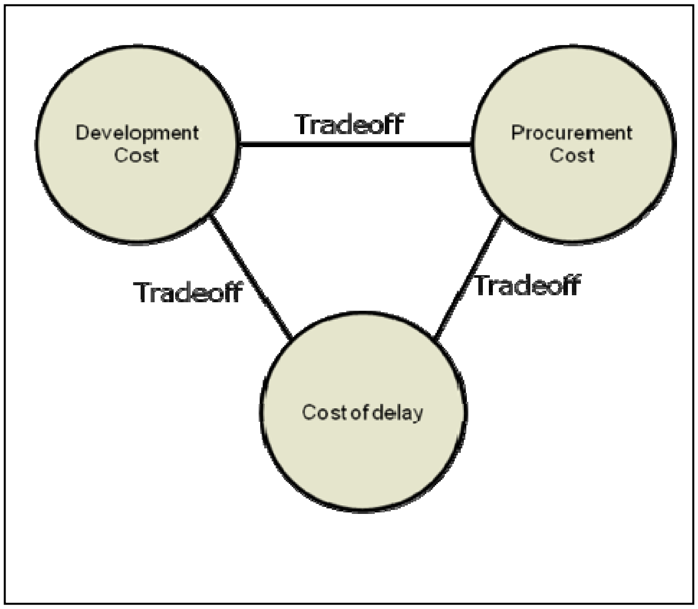
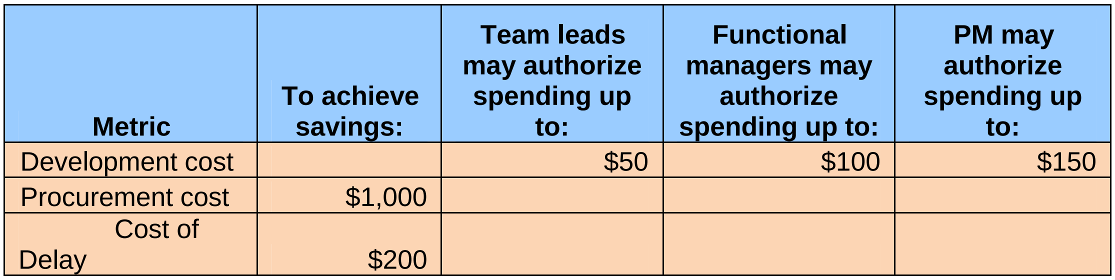


Figure 6. Economic Tradeoffs

The most difficult metric to calculate is the cost of delay. This is not often done in either government or private industry, since the cost of delay has a high subjective component. Nonetheless, an intelligent guess, especially if there is buy-in at every level of leadership, is better than none at all. For, if there is not even a guess, many decisions that may affect queues and cycle time—and thus the program being on schedule—may have to be made above the PMO or after the program slips. Or to put it another way, having upfront guidelines to make these tradeoffs makes it possible to push many decisions down to the PMO's teams where quick response at the most detailed level may help prevent schedule slips.

An example of a tradeoff guideline is, "you are authorized to spend up to $100 to save $200 cost of delay, without asking for permission." One dollar value can be given to the PMO, who may give smaller limits to the teams below depending on the degree of oversight desired. As Reinertsen has pointed out regarding product development (2009), this kind of guideline can become a core part of mission-type orders (Lind, 1985) to the PMO. Figure 6 below shows a notional way to transmit guidelines to the PMO.



Metric	To achieve savings:	Team leads may authorize spending up to:	Functional managers may authorize spending up to:	PM may authorize spending up to:
Development cost		$50	$100	$150
Procurement cost	$1,000			
Cost of Delay	$200			

Figure 7. Sample Economic Guidelines

Summary

Queues in acquisition are good leading indicators of future schedule slips. Queues can be managed by ranking requirements, controlling task starts, staffing adaptively, setting up "SWAT teams" of acquisition experts, parallelizing tasks, reviewing cadences, and establishing guidelines for tradeoffs. A list of references is provided to direct the reader to the latest writing I found on the subject.

This publication contains general information only and Deloitte is not, by means of this publication, rendering accounting, business, financial, investment, legal, tax, or other professional advice or services. This publication is not a substitute for such professional advice or services, nor should it be used as a basis for any decision or action that may affect your business. Before making any decision or taking any action that may affect your business, you should consult a qualified professional advisor. Deloitte shall not be responsible for any loss sustained by any person who relies on this publication.

About Deloitte

As used in this document, "Deloitte" means Deloitte Consulting LLP, a subsidiary of Deloitte LLP. Please see www.deloitte.com/us/about *for a detailed description of the legal structure of Deloitte LLP and its subsidiaries.*

References

Hopp, W.J., & Spearman, M.L. (1996). *Factory physics: Foundations of manufacturing management*. Chicago: Irwin.

Lind, W.S. (1985). *Maneuver warfare handbook*. Boulder, CO: Westview Press.

Morgan, J.M., & Liker, J.K. (2006). *The Toyota product development system: Integrating people, process, and technology*. New York: Productivity Press.

Ohno, T., & Bodek, N. (2008). *Toyota production system: Beyond large-scale production*. New York: Productivity Press.

Reinertsen, D.G. (2009). *The principles of product development flow: Second generation lean product development*. Redondo Beach, CA: Celeritas.

Womack, J.P., & Jones, D.T. (2003). *Lean thinking: Banish waste and create wealth in your corporation*. New York: Free Press.

THIS PAGE INTENTIONALLY LEFT BLANK

2003 - 2010 Sponsored Research Topics

Acquisition Management

- Acquiring Combat Capability via Public-Private Partnerships (PPPs)
- BCA: Contractor vs. Organic Growth
- Defense Industry Consolidation
- EU-US Defense Industrial Relationships
- Knowledge Value Added (KVA) + Real Options (RO) Applied to Shipyard Planning Processes
- Managing the Services Supply Chain
- MOSA Contracting Implications
- Portfolio Optimization via KVA + RO
- Private Military Sector
- Software Requirements for OA
- Spiral Development
- Strategy for Defense Acquisition Research
- The Software, Hardware Asset Reuse Enterprise (SHARE) repository

Contract Management

- Commodity Sourcing Strategies
- Contracting Government Procurement Functions
- Contractors in 21st-century Combat Zone
- Joint Contingency Contracting
- Model for Optimizing Contingency Contracting, Planning and Execution
- Navy Contract Writing Guide
- Past Performance in Source Selection
- Strategic Contingency Contracting
- Transforming DoD Contract Closeout
- USAF Energy Savings Performance Contracts
- USAF IT Commodity Council
- USMC Contingency Contracting

Financial Management

- Acquisitions via Leasing: MPS case
- Budget Scoring
- Budgeting for Capabilities-based Planning

- Capital Budgeting for the DoD
- Energy Saving Contracts/DoD Mobile Assets
- Financing DoD Budget via PPPs
- Lessons from Private Sector Capital Budgeting for DoD Acquisition Budgeting Reform
- PPPs and Government Financing
- ROI of Information Warfare Systems
- Special Termination Liability in MDAPs
- Strategic Sourcing
- Transaction Cost Economics (TCE) to Improve Cost Estimates

Human Resources

- Indefinite Reenlistment
- Individual Augmentation
- Learning Management Systems
- Moral Conduct Waivers and First-tem Attrition
- Retention
- The Navy’s Selective Reenlistment Bonus (SRB) Management System
- Tuition Assistance

Logistics Management

- Analysis of LAV Depot Maintenance
- Army LOG MOD
- ASDS Product Support Analysis
- Cold-chain Logistics
- Contractors Supporting Military Operations
- Diffusion/Variability on Vendor Performance Evaluation
- Evolutionary Acquisition
- Lean Six Sigma to Reduce Costs and Improve Readiness
- Naval Aviation Maintenance and Process Improvement (2)
- Optimizing CIWS Lifecycle Support (LCS)
- Outsourcing the Pearl Harbor MK-48 Intermediate Maintenance Activity
- Pallet Management System
- PBL (4)
- Privatization-NOSL/NAWCI
- RFID (6)

- Risk Analysis for Performance-based Logistics
- R-TOC AEGIS Microwave Power Tubes
- Sense-and-Respond Logistics Network
- Strategic Sourcing

Program Management

- Building Collaborative Capacity
- Business Process Reengineering (BPR) for LCS Mission Module Acquisition
- Collaborative IT Tools Leveraging Competence
- Contractor vs. Organic Support
- Knowledge, Responsibilities and Decision Rights in MDAPs
- KVA Applied to AEGIS and SSDS
- Managing the Service Supply Chain
- Measuring Uncertainty in Earned Value
- Organizational Modeling and Simulation
- Public-Private Partnership
- Terminating Your Own Program
- Utilizing Collaborative and Three-dimensional Imaging Technology

A complete listing and electronic copies of published research are available on our website: www.acquisitionresearch.org

THIS PAGE INTENTIONALLY LEFT BLANK

ACQUISITION RESEARCH PROGRAM
GRADUATE SCHOOL OF BUSINESS & PUBLIC POLICY
NAVAL POSTGRADUATE SCHOOL
555 DYER ROAD, INGERSOLL HALL
MONTEREY, CALIFORNIA 93943

www.acquisitionresearch.org