

# OSD Software Metrics Reporting Proposal

PSM Workshop

Vail, CO

July 27, 2000

Tom Coonce, OSD/PA&E

John Bailey, IDA

## Goal

- **Document both the anticipated and actual costs and sizes of DoD software developments so that software cost estimators can base future estimates on prior experience.**
  - As remarkable as it may seem, there is currently no way to study the outcomes of DoD software projects.
  - Anticipated cost, size, and schedule are documented as part of the planning process.  
However -
    - **Final size, schedule, and quality are rarely recorded.**  
Time passes, requirements change, and people leave projects as they wind down. Data on the actual experience is lost.

## Software Measurement Stakeholders

- **Cost Analysis Improvement Group (CAIG)** within PA&E is responsible for developing independent estimates for weapon systems (ACAT IC and ID programs)
- **Service Cost Centers** are responsible for estimating ACAT IA, IC, and ID programs
  - Interested in better data on both embedded and business applications (MAIS)
- **CAIG** sponsors a **Contractor Cost Data Reporting Project Office** that collects weapon system costs to support all estimators (CAIG and Service Cost Centers)
- **PA&E** is responsible for reviewing and advising C<sup>3</sup>I on MAIS life cycle cost estimates and Acquisition Program Baseline breaches

## Software Measurement Stakeholders (Concluded)

- **Service Cost Centers** requested **CCDR-PO** research how community can obtain better data to estimate software systems (weapon systems and MAISs)
  - Need historical cost and metric data to estimate similar future systems
  - Tried and failed to match CARDS with CCDR data
    - » Only initial metric data contained within CARD
    - » WBS did not go low enough to provide software cost data
- Goal is to collect a common minimal set of software data from embedded and MAIS systems to support estimating

## Software Metrics Considerations

- **Desire externally-useful high level records**
  - Provide analytically useful yet parsimonious data
- **Organize with respect to SEI core measures**
- **Study PSM approach**
  - Allow tailoring and use of within-project definitions
- **Detach reporting from CCDR**
- **Replace an existing form required of embedded systems (DD Form 2630)**
  - Existing form was four pages of inputs for several common software cost models
  - Proposed new version is two pages of meta-data

## Summary of Proposed Process

- **All projects**
  - **Cost Working-Level Integrated Product Teams (CWIPTs) identify measures, tailor data elements, create software metrics data plan and dictionary**
- **MDAPs**
  - **CAIG Chair approves software metrics plans and submits to PM who places on contract**
  - **Contractors submit data to central web site**
- **MAIS (very tentative):**
  - **Information Technology Working Integrated Product Team (ITWIPT) submits software metrics plan to PM who either places data on contract or obtains data through other means**
  - **PM submits data to central web site**

# Software Metrics Reporting Proposed Form (Tailorable)

## Overall Content

- **Six parts, identification plus core measures**
  - Context
  - Description
  - Size
  - Schedule and Effort
  - Staffing
  - Quality
- **Tailoring allowed**
  - Must retain core measures but specific metrics are negotiable
- **Three submission points**
  - CARD (by government program office)
  - Within 60 days of contract award (by developer or PM)
  - Final delivery (by developer or PM)



## Page One: Context

### ● Part 1: Report Context

- 1. System/Element Name (version/release):
- 2. Report As Of:
- 3. Authorizing Vehicle (MOU, contract/amendment, etc.):
- 4. Indicate Reporting Event:  
    \_\_\_CARD      \_\_\_Contract Award      \_\_\_Final

## Context Considerations and Issues

- **Wanted identification of an identifiable system or release for future reference**
  - Report expected at IOC or end of development
  - May not align with contract expenditures
  - Each block release of an iterative enhancement should probably be reported separately
- **Size thresholds for reporting based on dollar value of work or designation by government**
- **Subcontract work that integrates with named system should be reported in some way**
  - User-defined whether report includes subcontracts
  - Independent reporting of subcontracts is permitted
  - Issue might arise if subcontract is below size threshold

## Page One: Context

- **Part 1: Report Context, continued**

*Only to be completed for second and third submissions*

- 5. Development Organization:
- 6. Certified CMM Level (or equivalent):
- 7. Certification Date:
- 8. Lead Evaluator:
- 9. Affiliation:
- 10. Precedents (list up to five similar systems by the same organization or team):

## **Context Considerations and Issues, continued**

- **Industry resistance to CMM level reporting**
  - Alternatives exist to CMM level rating
  - Some certifications are questionable
  - Certification of organization may not apply to project
- **Analysts hoped to replace many descriptive questions with CMM level**
  - Developer Capability, Tool Use, Modern Practices, etc.
- **It is understood that level could change between submissions**
- **Are the current questions reasonable?**
- **Is there a better way to get this insight?**

## Page One: Description

### ● Part 2: Product and Development Description

- 1. Primary Application Type:
- 2.           % of product size
- 3. Development Process
- 4. Upgrade or New?
- 5. Secondary Application Type (Third, Fourth, etc.)
- 17. Primary Language
- 18.           % of product size
- 19. Secondary Language, etc.
- 21. List COTS/GOTS Applications:

## Description Considerations and Issues

- **Wanted application types from constrained list**
  - Can we know them all or will tailoring add as needed?
- **Should all COTS/GOTS be listed or just a total percentage?**
- **Should the percentage contribution to the whole system from each COTS/GOTS item be listed?**

## Page One: Size

### ● Part 3: Product Size Reporting

- 1. Number of Software Requirements, not including External Interface Requirements:
- 2. Number of External Interface Requirements (i.e., not under project control):
- 3. Code Size Measures for items 4 through 11. For each, indicate S for physical SLOC (carriage returns); Snc for noncomment SLOC only; LS for logical statements; or provide abbreviation \_\_\_\_\_ and explain in Software Metrics Data Dictionary.
- 4. New Code developed for COTS/GOTS Integration and under Configuration Control (Size in \_\_\_\_\_):
- \*5. All Other New Code under Configuration Control (Size in \_\_\_\_\_):

## Page One, concluded: Size

### ● Part 3: Product Size Reporting, continued

- \*6. Modified Generated Code under Configuration Control (Size in \_\_\_\_\_):
- \*7. Unmodified Generated Code under Configuration Control (Size in \_\_\_\_\_):
- 8. Modified Internally Reused Code under Configuration Control (Size in \_\_\_\_\_):
- 9. Unmodified Internally Reused Code under Configuration Control (Size in \_\_\_\_\_):
- 10. Modified External Reused Code under Configuration Control (Size in \_\_\_\_\_):
- 11. Unmodified External Reused Code under Configuration Control (Size in \_\_\_\_\_):



## Size Considerations and Issues

- **Did not want to impose counting rules**
- **Wanted to distinguish new from reused**
- **Wanted to distinguish human-written products from generated software**
- **Wanted to understand the overhead for reuse**
  - **Hoped to distinguish internal from external reuse**
  - **Hoped to learn how much reused code is modified**
- **Questions**
  - **Are current categories too detailed?**
  - **Could we just use new, modified, and reused size?**
  - **Does configuration control signify deliverable code?**
  - **What will be tailored?**
  - **Will analysts be able to use tailored data?**

## Page Two: Resources and Schedule

- **Part 4: Resource and Schedule**
  - **Show Start and End Month after contract award and Total Labor Hours for each phase or activity shown**
  - **The following items should account for all direct hours (use item 7 for any hours not accounted for in items 1 through 6). Explain any contribution of indirect hours in the associated Software Metrics Data Dictionary**
    - 1. Software Requirements Analysis
    - 2. Software Architecture and Detailed Design
    - 3. Software Coding and Unit Testing
    - 4. Software Integration and System/Software Integration
    - 5. Software Qualification Testing
    - 6. Software Operational Test and Evaluation
    - 7. All Other Direct Software Engineering Development Effort

## **Resource and Schedule Considerations and Issues**

- **Interested in realism of original project plan**
  - Did project go as expected?
- **Use general process categories**
  - Patterned after 498 / 12207 processes, not phases
- **We may be forcing processes to be phases**
  - Can we get both or would that be too complicated?
  - Are analysts better off with tailored phases, start/end dates and effort? Could reviews or intermediate product deliveries be phase delimiters?
  - Or, would analysts prefer more standardization of processes but without start/end dates, only effort?
- **Should we make resource and schedule reporting more general and drop default terms?**

## Page Two: Staffing

### ● Part 5: Staffing Profile

- 1. Peak FTE's directly charging to project: \_\_\_\_\_  
*The maximum "burn rate" of project labor charges.*
- 2. Month number of midpoint of peak: \_\_\_\_\_
- 3. Percent of personnel:
  - » Highly experienced in project domain: \_\_\_\_%
  - » Nominally experienced: \_\_\_\_%
  - » Entry level, no experience: \_\_\_\_%

## Staffing Considerations and Issues

- **Interested in cross check of schedule planning**
  - Are total effort, duration, and peak staffing consistent?
  - Ambitious schedules may have some bearing on final cost
  - Analysts will use insight into these relationships to estimate new jobs better
- **Domain experience is independent of CMM level**
  - This question might overlap with “precedents” list
- **Are these questions reasonable?**
  - Could they be ambiguous?

## Page Two: Product Quality

### ● Part 6

- **First question only for initial submission by government P.O., prior to contract award:**
- 1. Required Mean Time to Defect (MTTD) at Delivery.  
\_\_\_\_\_ Hours  
Or, provide an alternate method that compares the required reliability of this system with the nominal reliability for systems of this type: \_\_\_\_\_  
*E.g., quantitative or qualitative comparison, as appropriate*
- Questions 2 through 7 are to be answered at the final submission submission of form, after project completion.  
*Could the values also be anticipated at time of project award for second submission?*

## Page Two, concluded: Quality

Report cumulative defect counts since project start in each category. Use associated Software Metrics Data Dictionary to define counting rules as necessary.	Actuals After Completion of Software Qualification Test	Actuals After Completion of Operational Test and Evaluation
2. Cumulative Number of Critical Defects Discovered		
3. Cumulative Number of Serious Defects Discovered		
4. Cumulative Total Number of Defects Discovered		
5. Cumulative Number of Critical Defects Resolved		
6. Cumulative Number of Serious Defects Resolved		
7. Cumulative Total Number of Defects Resolved		

## Quality Considerations and Issues

- **Analysts are eager to know this fourth leg of the core measures set**
  - Could explain why two similar projects had different costs or schedules
  - But, not clear how this will impact new estimates, since most projects specify “high” quality in the beginning
- **Considerable industry resistance to defect reporting has been encountered**
- **Many practical considerations exist**
  - Definition of a defect
  - When to start counting
  - How to count related issues
- **Is there a better way to get a quality measure?**



Unclassified

# Revised DD Form 2630, Page 1

Software Product Development Report			
Page 1: Report Context, Project Description and Size			
<b>1. Report Context</b>			
1. System/Element Name (version/release):		2. Report As Of:	
3. Authorizing Vehicle (MOU, contract/amendment, etc.):		4. Indicate Reporting Event: ___ CARD ___ Contract Award ___ Final	
Items 5 through 10 are to be answered only for Contract Award and Final reports.			
5. Development Organization:		6. Certified CMM Level (or equivalent):	8. Lead Evaluator:
		7. Certification Date:	9. Affiliation:
10. Precedents (list up to five similar systems by the same organization or team)			
<b>2. Product and Development Description</b>			
1. Primary Application Type:		2. %	3. Development Process
5. Secondary Application Type:		6. %	4. Upgrade or New?
9. Third Application Type:		10. %	7. Development Process
13. Fourth Application Type:		14. %	8. Upgrade or New?
17. Primary Language:		18. %	11. Development Process
19. Secondary Language:		20. %	12. Upgrade or New?
21. List COTS/GOTS Applications:			
<b>3. Product Size Reporting</b>			
1. Number of Software Requirements, not including External Interface Requirements (unless noted in associated Software Metrics Data Dictionary)			Provide Estimates at CARD and Contract Award, Actuals at Final
2. Number of External Interface Requirements (i.e., not under project control)			
3. Code Size Measures for items 4 through 11. For each, indicate <u>S</u> for physical SLOC (carriage returns); <u>Src</u> for noncomment SLOC only; <u>LS</u> for logical statements; or provide abbreviation ____ and explain in Software Metrics Data Dictionary.			
4. New Code developed for COTS/GOTS Integration and under Configuration Control (Size in ____ )			
5. All Other New Code under Configuration Control (Size in ____ )			
6. Modified Generated Code under Configuration Control (Size in ____ )			
7. Unmodified Generated Code under Configuration Control (Size in ____ )			
8. Modified Internally Reused Code under Configuration Control (Size in ____ )			
9. Unmodified Internally Reused Code under Configuration Control (Size in ____ )			
10. Modified External Reused Code under Configuration Control (Size in ____ )			
11. Unmodified External Reused Code under Configuration Control (Size in ____ )			

DD Form 2630-R

Page 1 of 2

Unclassified

## Revised DD Form 2630, Page 2

Software Product Development Report			
Page 2: Project Resources, Schedule, Staffing and Quality			
<b>4. Resource and Schedule Reporting</b>		Provide estimates at CARD and Contract Award, Actuals at Final	
Show Start and End Month after contract award (counting from month 1 at contract award), and Total Labor Hours for each phase or activity shown		Start Month	End Month
Total Hours			
The following seven items should account for all direct hours charged to the software development project (use item 7 for any direct hours not accounted for in items 1 through 6). Explain any contribution of indirect hours in the associated Software Metrics Data Dictionary.			
1. Software Requirements Analysis			
2. Software Architecture and Detailed Design			
3. Software Coding and Unit Testing			
4. Software Integration and System/Software Integration			
5. Software Qualification Testing			
6. Software Operational Test and Evaluation			
7. All Other Direct Software Engineering Development Effort (Describe: _____) Report hours only:			
<b>5. Staffing Profile</b>			
1. Peak FTE's directly charging to project:		2. Month number of midpoint of peak:	
3. Percent of personnel: Highly experienced in project domain: ____% Nominally experienced: ____% Entry level, no experience: ____%			
<b>6. Product Quality Reporting</b>			
1. Required Mean Time to Defect (MTTD) at Delivery (only complete at CARD). Alternatively, provide an alternate method of comparing the required reliability of this system with the nominal reliability for systems of this type: _____ hours			
Report cumulative defect counts since project start in each category. Use associated Software Metrics Data Dictionary to define counting rules as necessary.		Actuals After Completion of Software Qualification Test	Actuals After Completion of Operational Test and Evaluation
2. Cumulative Number of Critical Defects Discovered			
3. Cumulative Number of Serious Defects Discovered			
4. Cumulative Total Number of Defects Discovered			
5. Cumulative Number of Critical Defects Resolved			
6. Cumulative Number of Serious Defects Resolved			
7. Cumulative Total Number of Defects Resolved			
Filename and Revision Date of Applicable Software Metrics Data Dictionary :			
Name of person to be Contacted	Signature	Telephone Number	E-Mail
			Date

## Proposed Processes

- **For ACAT IC and ID programs:**
  - PM prepares and submits DD Form 2630 with CARD
  - PM-led CWIPT identifies software elements, prepares draft data dictionary and documents into a Program Software Measurement Plan. Sends plan to CAIG Chair for approval
  - CAIG Chair approves plan and sends to PM along with proposed RFP language
  - PM develops Contract Software Measurement plan and requests data through RFP, DIDs, and CDRL
  - Developers propose Software Development and Measurement Plans (tailored DD Form 2630) and updated dictionary
  - Developer and PM negotiate contract or MOA (for CDAs)

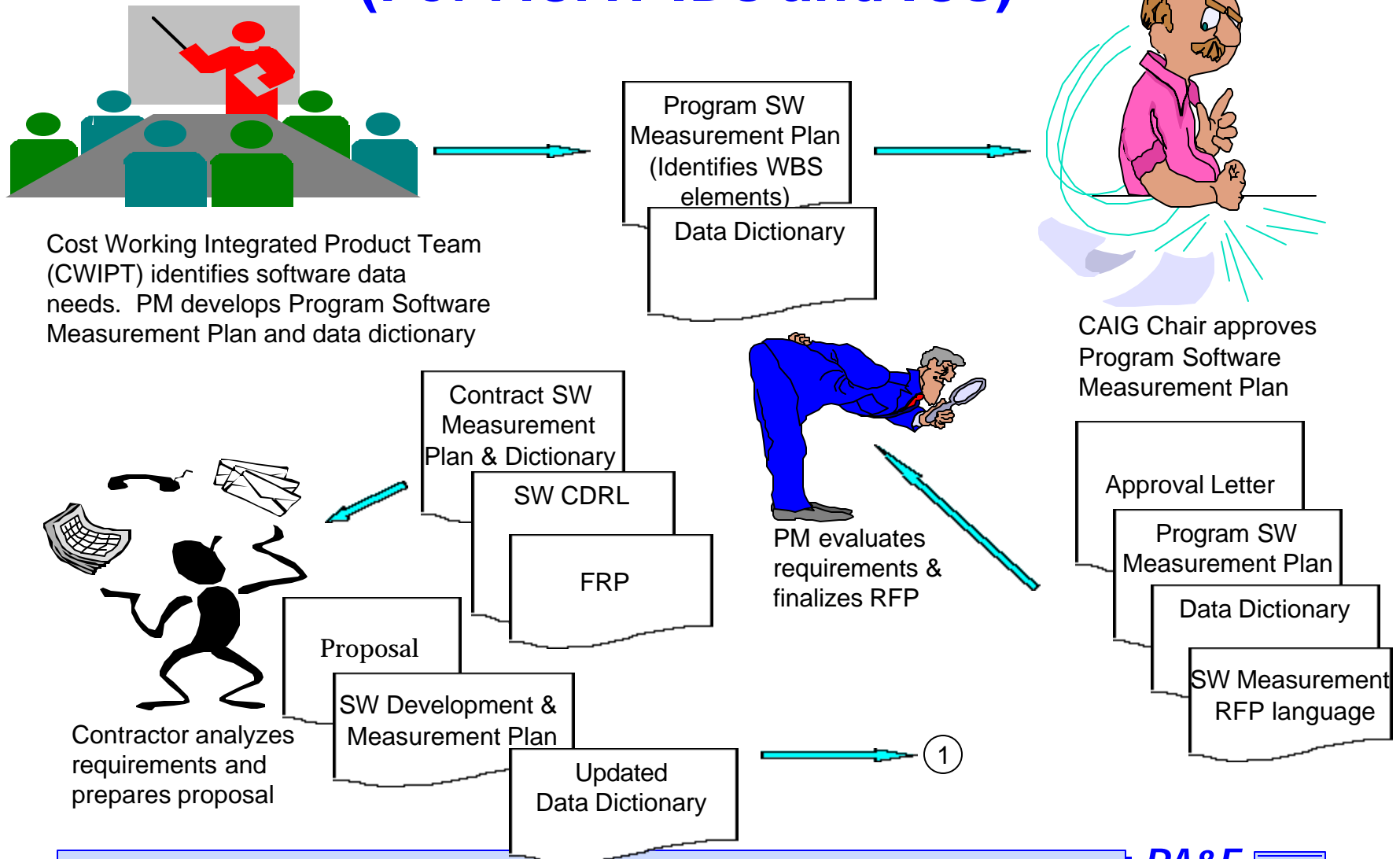
## Proposed Processes

- **ACAT IC and ID programs, continued:**
  - Developer submits DD Form 2630 and updated dictionary to central web site 60 days after award and after product delivery (block, fielded release, or contract completion)
  - PMs approve data for limited distribution
  - Cost analysts access data through secure web connection

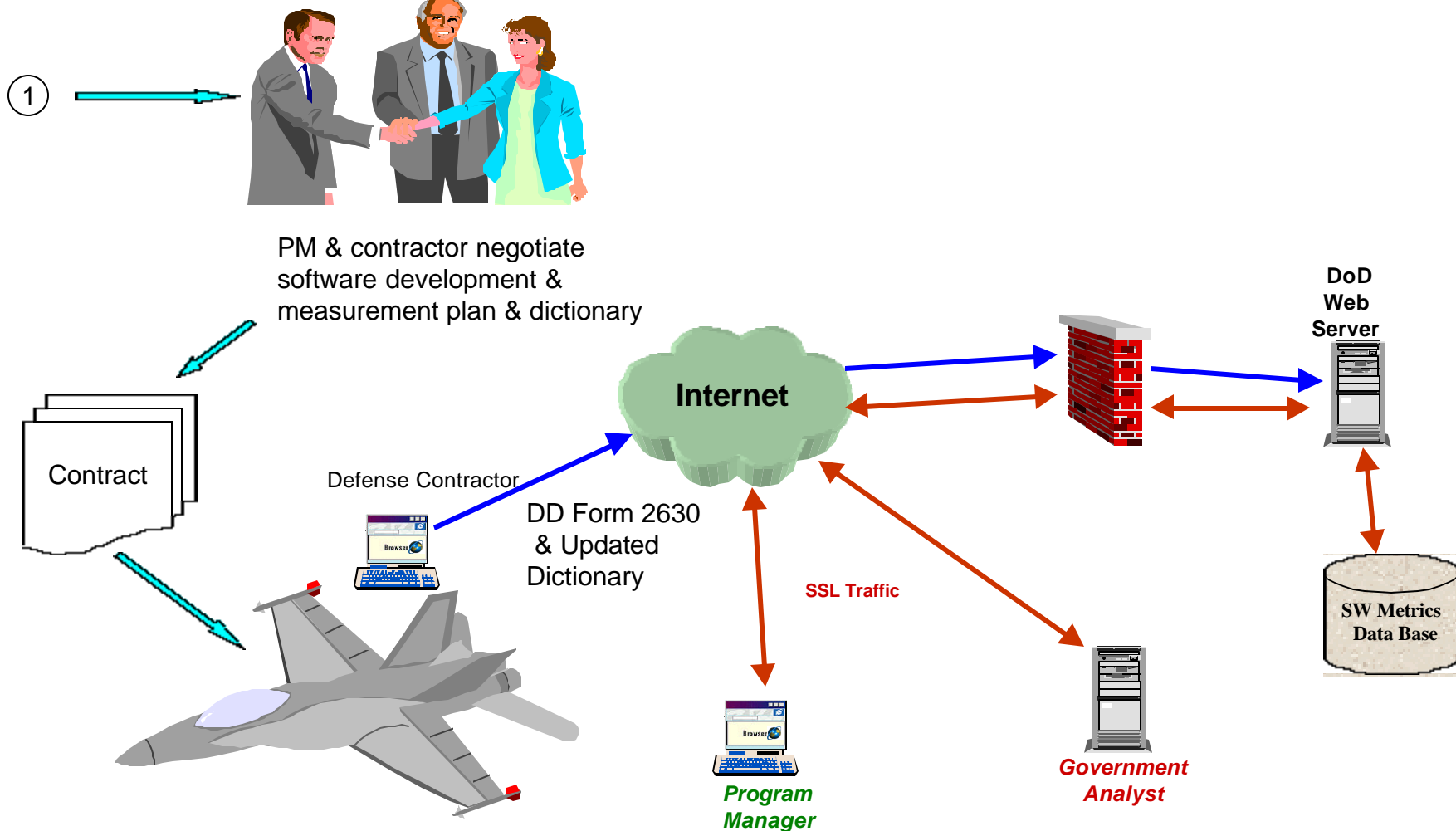
## Proposed Process (Concluded)

- **For ACAT IA (MAIS) programs (very tentative):**
  - PM prepares and submits DD Form 2630 with CARD
  - ITWIPT likely to coordinate the Software Measure Plan similar to MDAPs, but more coordination necessary to define the process
  - PMs are to provide the data to the central web site

# Software Metrics Planning Process (For ACAT IDs and ICs)



# Software Metrics Data Collection Process (For ACAT IDs and ICs)



## **Proposed Software Metric Language for DoD 5000.2-R (New Section 7.11.7.6)**

**ACAT I programs that contain software intensive elements must submit software metrics data. The specific data to be submitted will be determined by the IPT process using the Software Product Development Report (DD Form 2630) as the baseline. Data will be submitted at three intervals during the life of the program: at the time of cost analysis requirements description (CARD) submission, within 60 days of contract or task award, and within 60 days after software product delivery.**



## **Proposed Request for Proposal (RFP) Language**

**The contractor must prepare a Contract Software Development and Measurement Plan following the structure and elements shown in the attached Contract Software Measurement Plan and dictionary. The data elements identified in the Contract Software Measurement Plan are the software elements for which the Government desires measurement information. The contractor shall report these elements at the frequency indicated in the plan. The contractor may propose additions, deletions or modifications to those elements identified in the plan if the proposed elements are used by the contractor to manage the development effort. If changes are proposed, the contractor shall so indicate in the Measurement plan and describe them in an updated Software Measurement Data Dictionary.**

## Software Metrics: Possible Pilot Projects

- **Pilot testing**
  - Test the data (or meta-data)
  - Test the process
  - Seeking volunteer programs (on-going and “to-start” programs)
- **Current possible pilot test projects**
  - JSF
  - AV-8B
  - CEC (Cooperative Engagement Capability)
  - C-130
  - E-2C
- **Other possibilities?**

## Goal of Pilot Testing

- **Questions we hope to answer with pilot tests**
  - How readily available are the data?
  - How much effort is required to provide the data as defined?
  - To what extent is tailoring needed?
  - Is it difficult to tailor the DD Form 2630?
  - How well does the specification, tailoring, collection, and reporting procedure integrate with contractor processes?
  - How well can analysts use tailored data?

## Possible Improvements

- **As a result of review and pilot tests:**
  - **Should data descriptions be more general or more specific?**
  - **Have we struck the correct balance between precise definitions and meaningfulness**
    - » Rigid definitions would be ideal for analysts
    - » Past experience has proven this is unworkable
    - » Infinitely flexible reporting would be ideal for projects
    - » Calibration of future estimates is much harder with unconstrained data
  - **Should we be more or less ambitious in the data categories?**
    - » Should we drop some of the data, such as quality reporting or staff reporting

## Software Metrics: Next Steps

- **Revise proposal, DD Form 2630, DID and processes based on today's discussion and agreements**
- **Expand pilot tests**
- **Revise the proposal, form and processes based on test results**
- **Seek PA&E approval of the proposal and DoD 5000.2-R language**
- **Continue to coordinate with C<sup>3</sup>I on their software metric effort**
- **Create supporting documentation and implementing instructions -- DoD 5000.4-M-2**