U.S. ARMY COMBAT CAPABILITIES DEVELOPMENT COMMAND – ARMAMENTS CENTER

Five Agile Metrics for the Organization

Christopher Costello

Measurement Team Lead

Process and PM Engineer Support Branch

ABSTRACT

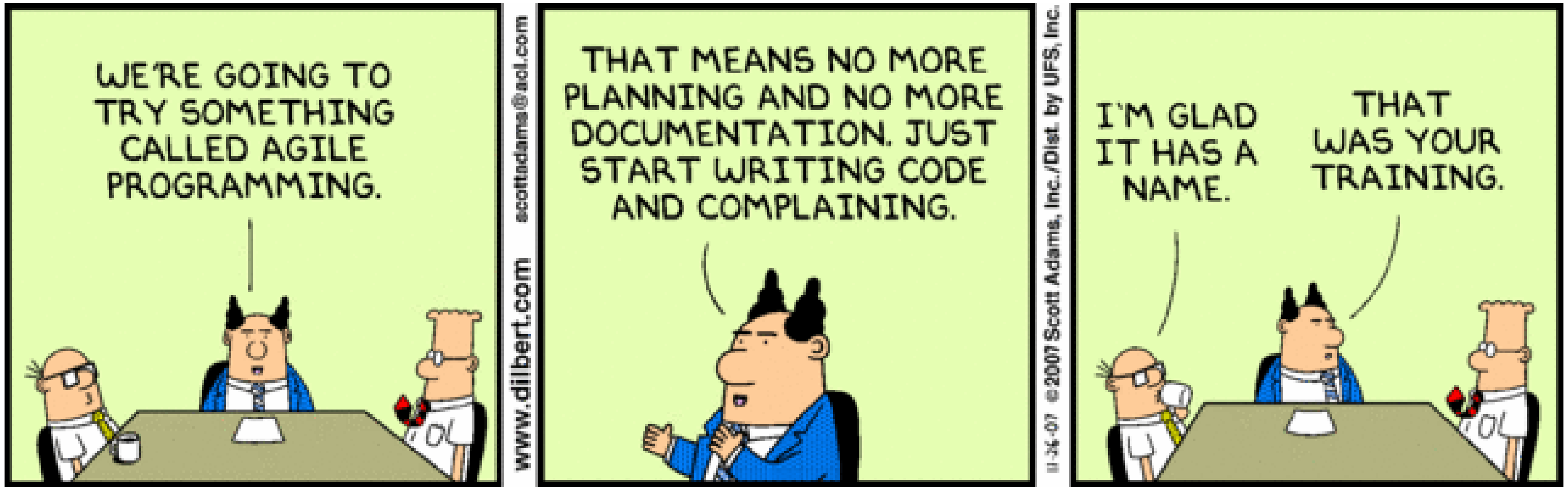
- The presentation will summarize the efforts that began after the Armament SEC approved the Agile lifecycle model for use by development teams in the organization.

- The Armament SEC is currently piloting five Agile measures that are structured using the Practical Software and Systems Measurement (PSM) Measurement Construct.

- The presentation will provide an overview of the role of the Measurement Team in the Process Engineering Group (PEG) insight into the rationale for selecting the measures being piloted, how they are being calculated, and a look at their Measurement Construct.

AGENDA

- **Background**
  - Overview
  - CCDC Armaments Software Engineering Center
  - PEG Measurement Team
  - Agile Lifecycle Model
  - Survey of Projects
  - Workshop
  - Key Findings
  - Conclusion
- **Metrics**
  - Evaluate and Select
  - Vacanti & Flow
  - Measures
- **What's Next**
  - Pilot
  - Assessment
  - Adoption
  - Baselining
  - Changes to Business Objectives

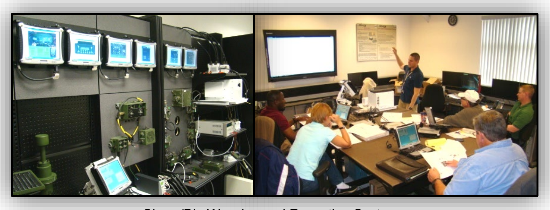


Background

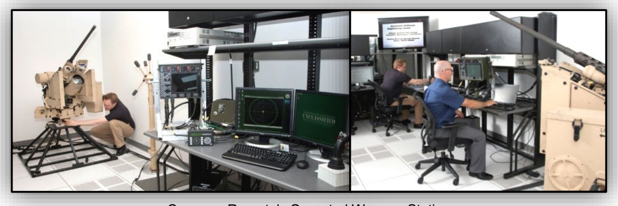
The Road to Agile

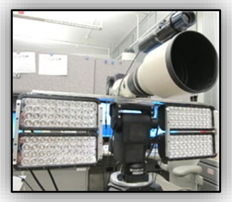
OVERVIEW

- **The Armament Software Engineering Center at Picatinny Arsenal has decades of experience implementing and overseeing software development and sustainment programs, using advanced measurement and analysis approaches (among other modern methods deployed).**
- **The influence of Agile software development is starting to be pervasive across DoD programs.**
- **Feedback from the CMMI Level 5 appraisal in 2016 indicated that additional organizational measures were needed for Agile projects**
- **Armament SEC Teamed with SEI for help and perspective due to limited organizational experience with Agile.**
- **The purpose of the study was to identify, document and communicate practical implementations of metrics and analysis models for Agile software development. The study includes metrics and models used by individual projects, as well as those serving an organizational-level capability performing measurement and analysis.**


Chem/Bio Warning and Reporting Systems


M1A1 Abrams


Common Remotely Operated Weapon Station


Biometrics Research and Engineering


Joint Precision Air Drop System

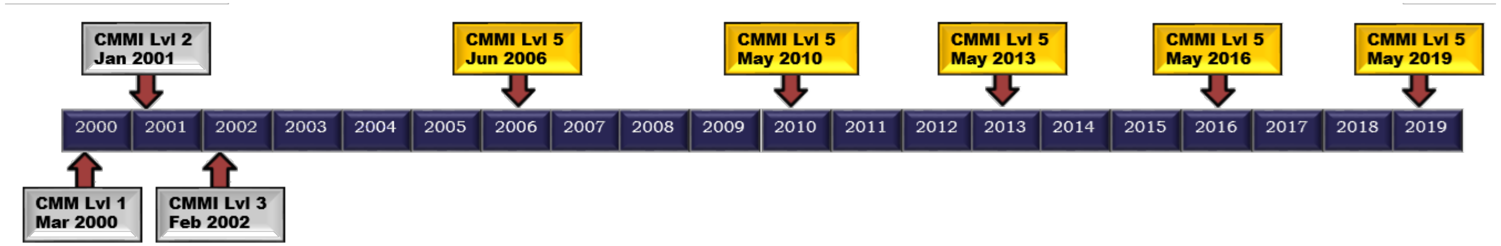
CCDC ARMAMENT SEC

- **The Armament SEC has made a joint commitment to continually maintain a formal process improvement initiative based upon the requirements of the Capability Maturity Model Integration (CMMI) for Development.**

- **The need to be recognized in the very competitive software intensive systems arena drove the selection of CMMI for Development and is the basis for process improvement.**

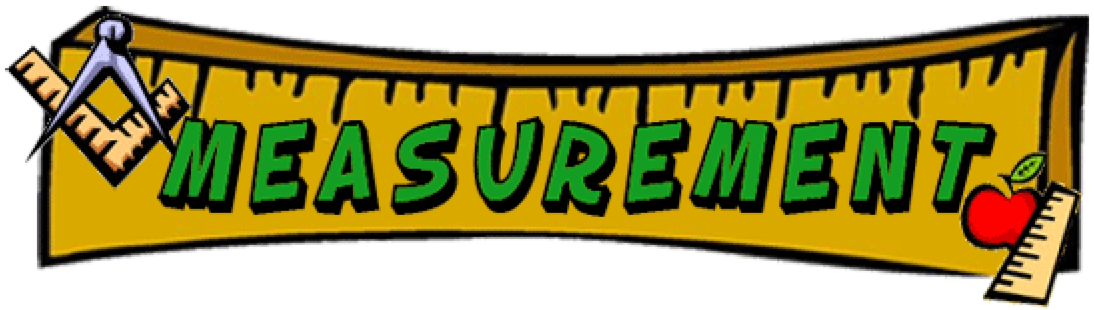
- **Practical Software and Systems Measurement (PSM), ISO/IEC 15939: Software Measurement Process, and CMMI V2.0 Managing Performance and Measurement Practice Area are used as a basis for the Armament SEC organizational measurement and analysis procedure.**

- **The Armament SEC implements statistical and other quantitative methods at the organizational and project levels to understand both past and future quality and process performance.**



PEG MEASUREMENT TEAM

- **The Measurement Team is responsible for all aspects of creating, analyzing and reporting of Organizational measures.**
- **The organizational measurement analyst investigates and recommends tools for defining, applying, sustaining, and improving the organizational measurement process.**
- **The organizational measurement analyst also performs periodic reviews of the measurement processes within the individual projects and ensures that project-level measures are integrated with the organizational measurement requirements.**



MEASUREMENT SELECTION PROCESS

- **The process performance measures outlined were drawn from information needs identified to meet the Armament SEC high level goals of the organization as described in the Capstone document:**
  - Improve Predictability, Consistency and Quality, of our Services and Products
  - Increase Productivity & Reduce Cycle Time
  - Maintain and Enhance our Core Competencies
  - Improve Customer Satisfaction
  - Improve our Competitive Advantage

- **Measurement Information Needs are based on the project's objectives, constraints, issues, and risks. It also takes into consideration the needs of the customer, and relevant stakeholders. The project specific information needs are grouped and prioritized into information categories based on the project's quantitative quality and process performance objectives:**
  - Schedule and Progress
  - Size and Stability
  - Resources and Cost
  - Product Quality and Process Performance

INFORMATION NEEDS

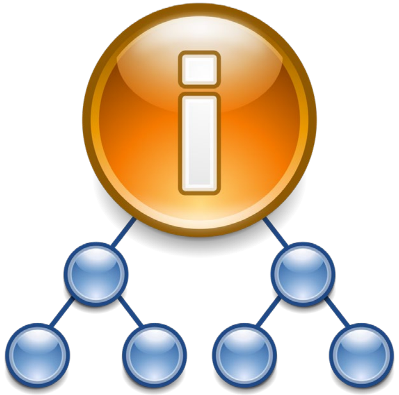
- **The project specific information needs are grouped and prioritized into information categories based on the project's quantitative quality and process performance objectives:**

- **Schedule and Progress**

- **Size and Stability**

- **Resources and Cost**

- **Product Quality and Process Performance**



Information Categories	Measure
Schedule and Progress	• Schedule Performance – Milestones • Task Completion Performance • Progress Forecast – (Agile Draft Measure)
Size and Stability	• Size (Development Projects) - Lines of Code (LOC) • Size (Acquisition Support, and Infrastructure Projects) – Total Number of Planned Tasks
Resources and Cost	• Effort • Cost • CPI • SPI
Product Quality and Process Performance	• Audit Profiles • Defect Containment Performance • Defect Discovery (Rayleigh Curve) • Peer Review indicators (Effectiveness & Efficiency) • Process Predictability – (Agile Draft Measure) • Process Proficiency – (Agile Draft Measure)

BUSINESS OBJECTIVES

1. Process compliance
2. Defect removal
3. Estimation variation
4. Customer satisfaction
5. Training compliance
6. Asset availability

AGILE LIFECYCLE MODEL

- **The CCDC AC SEC incorporated the Agile lifecycle model as part of its suite of available project lifecycle models.**

- **By implementing this lifecycle model, the organization expects to standardize tracking, reporting and metrics for projects using Agile development processes.**

- **The Agile lifecycle phases allow for improved organizational performance baselines which will enhance the analysis of organizational business objectives.**

- **Reduces project plan tailoring of life cycle model.**

- **The Agile Development Life Cycle Model is best suited for those projects where:**
  – The requirements are not fully understood.
  – Changes in requirements are expected during development.
  – Majority of development staff is self-organizing.
  – Customer/User is actively involved during the entire lifecycle.

- **References - The Scaled Agile Framework® (SAFe®), http://www.scaledagileframework.com/**

SURVEY

- Gathered information about Agile practices and needs through an organizational survey
- In 2016 a survey of the Armament SEC workforce determined that a total of five projects were Agile, one of which was reporting metrics to the organization's Process Engineering Group (PEG).
- As of May 2017, two agile projects are reporting metrics to the PEG and two others are formulating measurement plans.
- These four projects form the basis of the Agile Software Metrics study.
- This study was chartered to develop, pilot and analyze measurements and analysis models for projects using an Agile development approach.

WORKSHOP

- Coordinated a workshop to discuss Agile-specific goals with teams
- Interactive working sessions with project-focused and organization-focused personnel used to garner important considerations and opportunities for new work on software measurement indicators and analysis models in Agile development programs.
- The working sessions included brainstorming about software indicators. Current software indicators and models will be used as inputs to this task.
- This task will seed the details for further development, and seek to accomplish that with the participation and buy-in of those who will need to support the activity in order for it to provide lasting benefit.

KEY FINDINGS

- Customer-Driven Change
- Familiar Data, New Perspective
- Availability of Tools and Infrastructure
- Learning Organization
- Diverse Cross-Section of Project Types
- Product Quality and Process Management are Familiar Priorities
- Customer Emphasis on Formal Requirements and Testing

CONCLUSIONS

- The initiative taken by personnel working on projects using Agile methods provides a rich opportunity to learn from use of these new approaches to software development.
- The measurement and analysis framework established in the enterprise supports the process of defining measures and analysis techniques that meet the unique needs of Agile development approaches.
- While some established metrics and analysis models do not fit the Agile lifecycle model, other existing data collection regimes will continue to provide needed information – with appropriate adjustments to interpretations of the data.
- Some new metrics and analysis models are well established in the Agile projects, even while they continue to work on new ideas for implementation.
- Capturing the details of the metrics and analysis methods in use across the Agile projects, then supporting their effort to pilot new approaches will be a natural next step.
- The enterprise has a well-established mechanism for doing this.

Agile Measures

What to Measure and How to Measure It

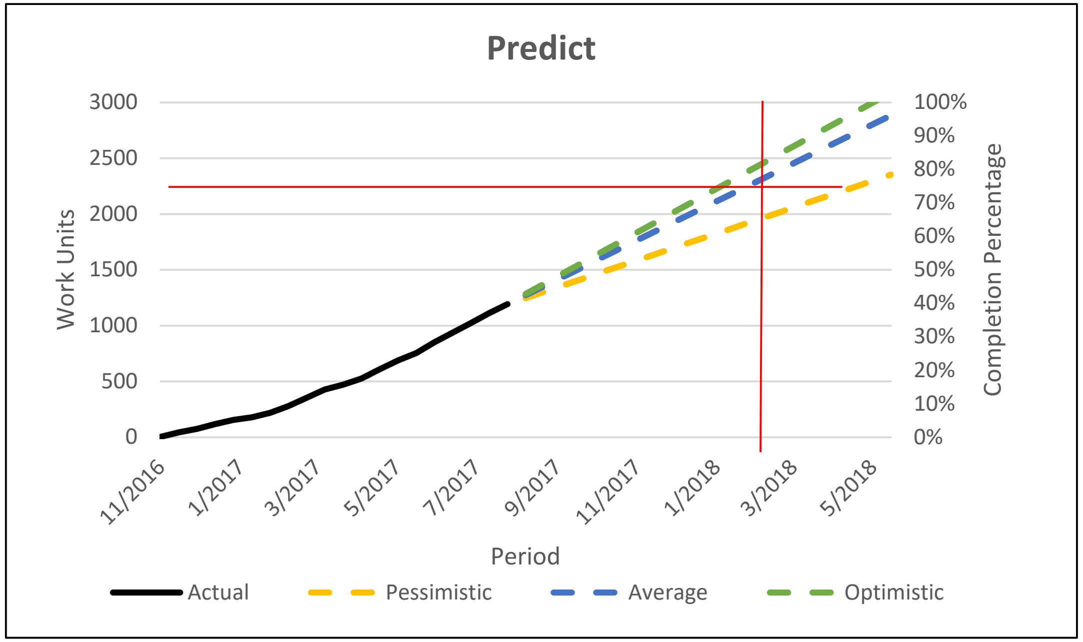
AGILE MEASUREMENTS – UNDERSTAND & CONTROL TO PREDICT

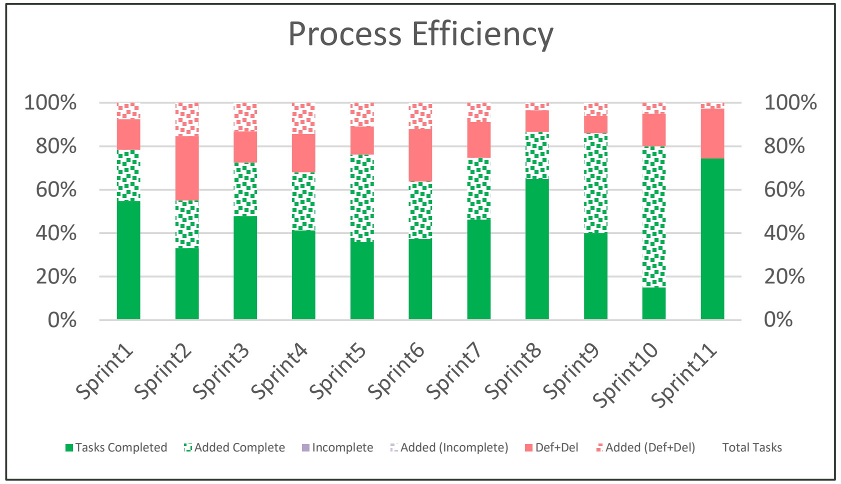
DEVCOM ARMAMENTS CENTER

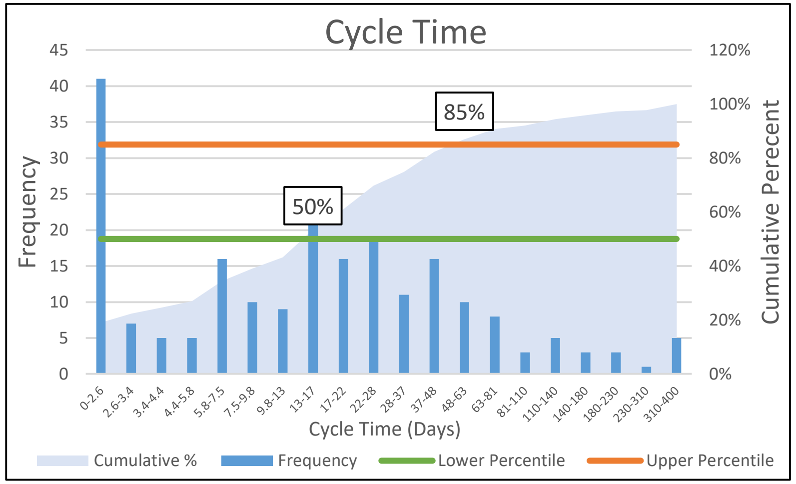
Defect Containment Matrix			Sprint Defect Detected										Total Defects (Contained & Leaked)	Total Defects Contained	Total Defects Leaked
		Version	V8.0.0	V8.0.0	V8.0.0	V8.0.0	V8.0.0	V8.0.0	V8.0.0	V8.0.0	V8.0.0	V8.0.0			
	Version	Sprint	Sprint 1	Sprint 2	Sprint 3	Sprint 4	Sprint 5	Sprint 6	Sprint 7	Sprint 8	Sprint 9	Sprint 10			
Sprint Defect Originated	V8.0.0	Sprint 1	7	16	0	12	6	2	6	1	1	0	51	7	44
	V8.0.0	Sprint 2		10	0	0	0	0	0	1	0	0	11	10	1
	V8.0.0	Sprint 3			3	17	4	1	0	0	0	0	25	3	22
	V8.0.0	Sprint 4				20	4	1	1	0	0	1	27	20	7
	V8.0.0	Sprint 5					15	4	1	0	0	1	21	15	6
	V8.0.0	Sprint 6						2	5	0	1	0	8	2	6
	V8.0.0	Sprint 7							12	2	0	0	14	12	2
	V8.0.0	Sprint 8								5	0	0	5	5	0
	V8.0.0	Sprint 9									2	0	2	2	0
	V8.0.0	Sprint 10										0	0	0	0
Total Defects per Sprint			7	26	3	49	29	10	25	9	4	2	164	76	88

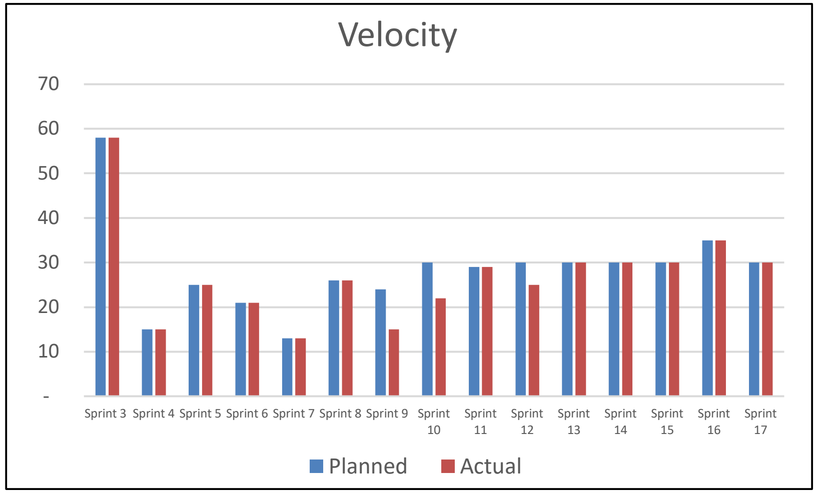
Defect Containment Effectiveness 46.34%

Defect Leakage Effectiveness 53.66%









EVALUATE AND SELECT

Information Category	Example Measures Provided by Projects	
Schedule & Progress	• Product Backlog Weight • Sprint Velocity • Sprint Burndown • Epic/Release Burndown • Cumulative Flow • Version Report from JIRA	• Task Completion • Iteration Reports • % Complete By Capability • Theme Burn Up Chart • Task Cycle Time Control Chart
Size & Stability	• Sprint SLOCS • Sprint Report from JIRA • LOC Bar Chart	• LOC Change Over Time • LOC by Extension
Resources & Cost	• Productivity – Effort per Product Demo	• Time Spent Per Issue Type
Product Quality & Process Performance	• Sprint Defects • Cycle Time Control Chart • Sprint Health Indicators • McCabe's Complexity	• Defect Containment Matrix • Defect Counts Phase/Total • Rayleigh Curve • Peer Review Measures

VACANTI – FLOW

- **Daniel S. Vacanti, *Actionable Agile Metrics for Predictability* (Leanpub, 2015).**

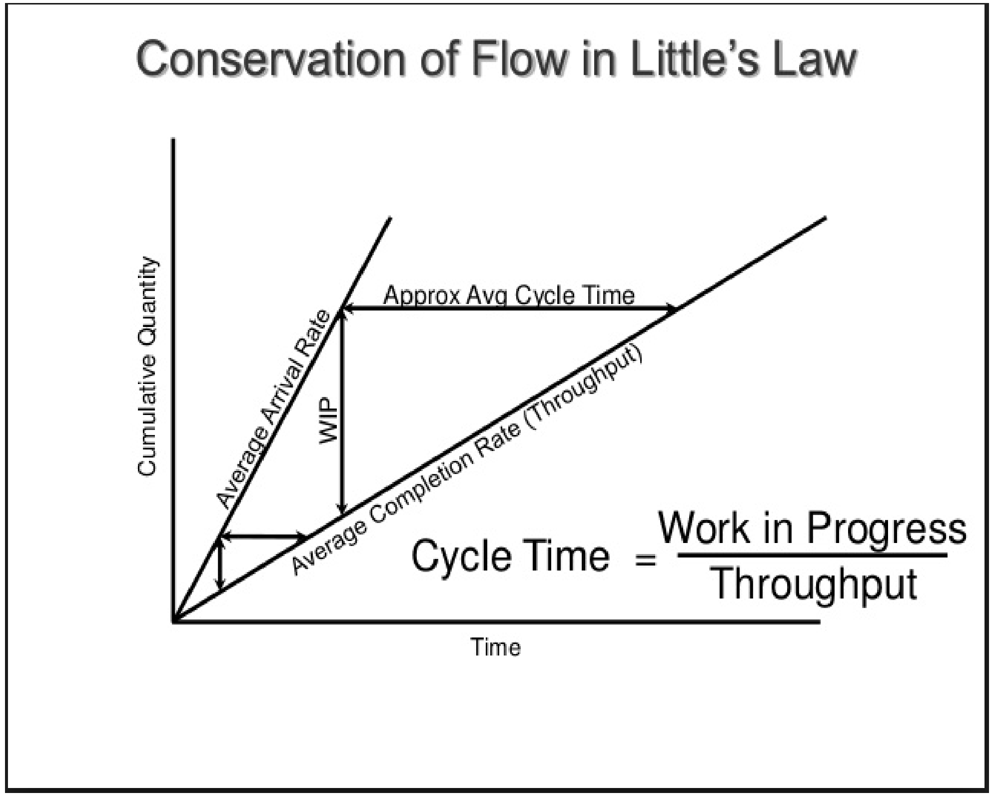
Little's Law is
Average Items in Queue = Average Arrival Rate * Average Wait Time

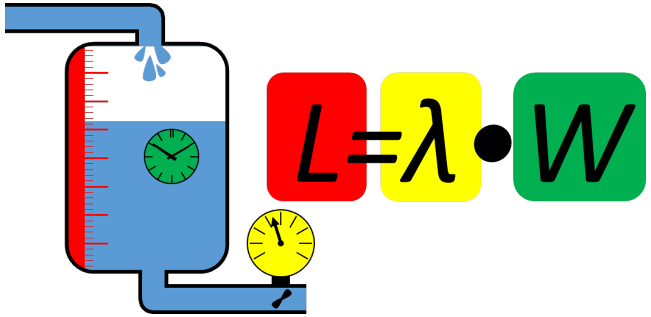
In knowledge work, we use it as
"Work In Progress = Cycle Time * Throughput"

With this form of the law, there are some assumptions are important for discussion:

1. The state of the process must be steady (long enough to give good measures)
2. All work that is started will eventually be completed and exit the system

Create Measurements with Work Item Size or Cycle Time



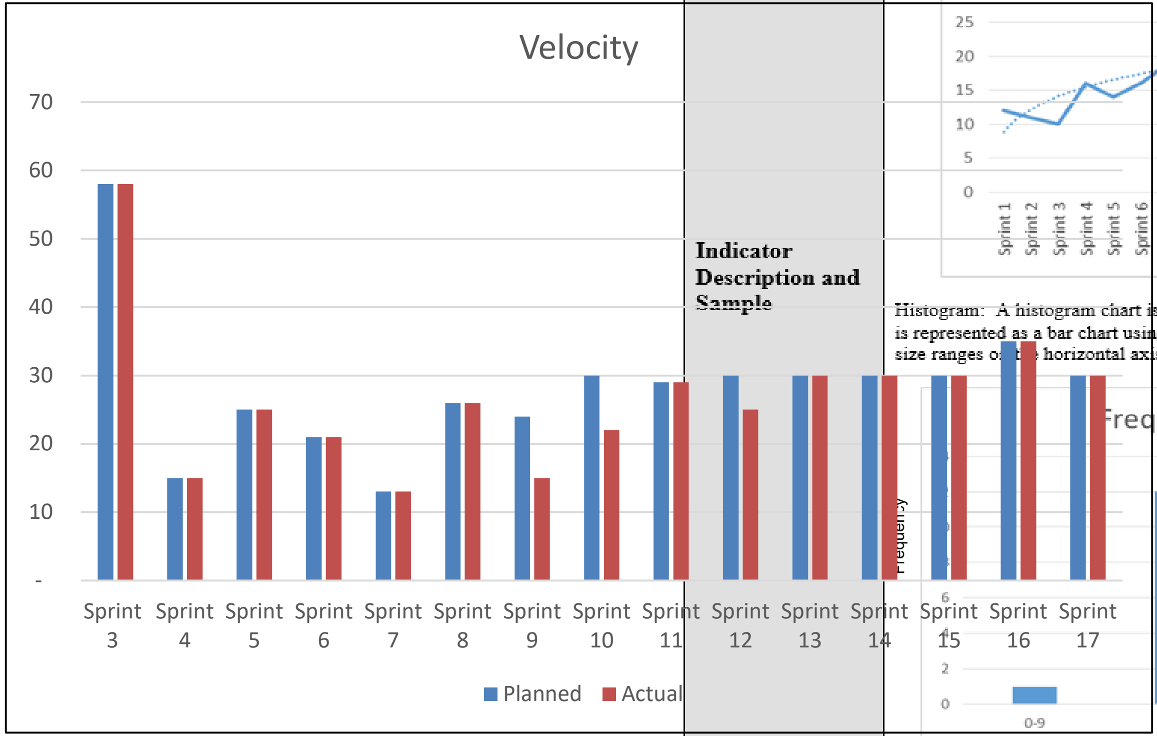


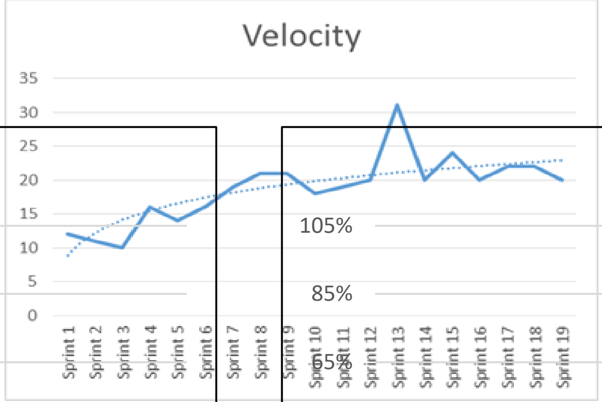
MEASURES

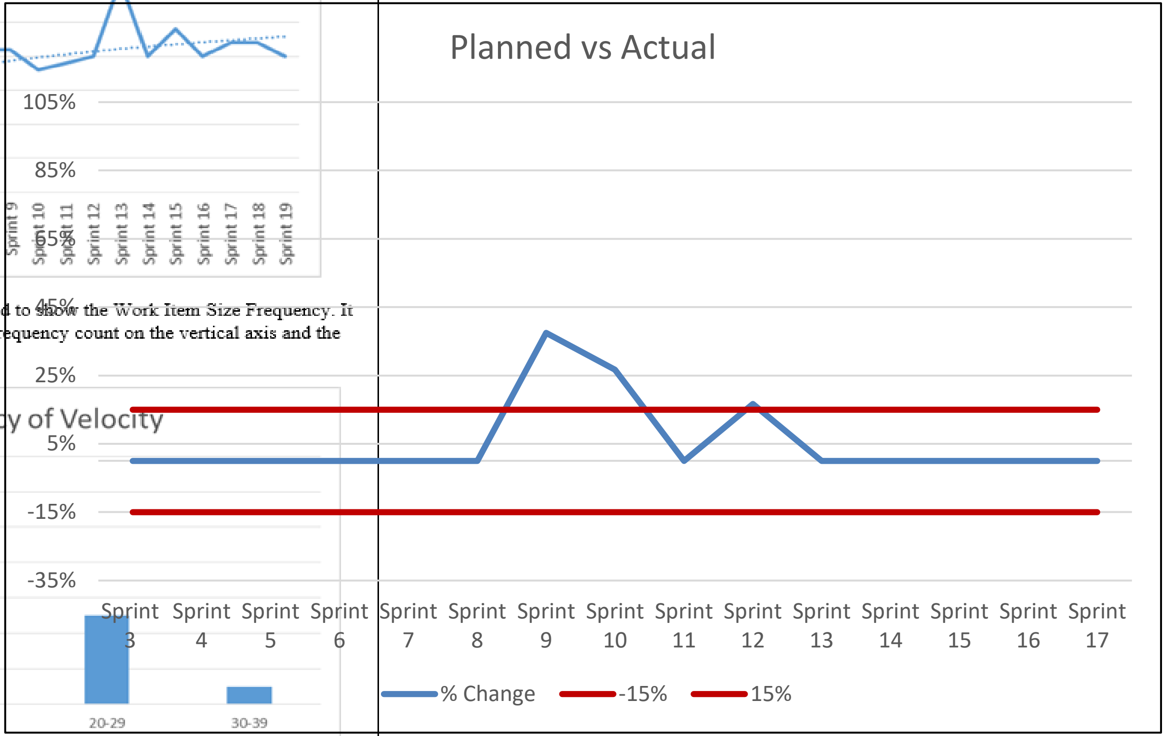
- Process Predictability (Velocity)
- Process Predictability (Cycle Time)
- Process Efficiency
- Progress Forecast
- Sprint DCM

- Use PSM Construct

PROCESS PREDICTABILITY (VELOCITY)



Indicator Specification	
Indicator Description and Sample	Work Item Velocity Scatterplot Line Chart: is created to show the time interval along the horizontal axis and the vertical axis is the velocity.  Histogram: A histogram chart is created to show the Work Item Size Frequency. It is represented as a bar chart using the frequency count on the vertical axis and the size ranges on the horizontal axis.
Analysis Model	Are the variations outside the bounds of the project's expectations?
Decision Criteria	Scatterplot: Investigate if there is a pattern of velocities falling far outside the typical velocity ranges. Histogram: Investigate if there is a pattern of frequencies falling far outside project's typical frequency range.



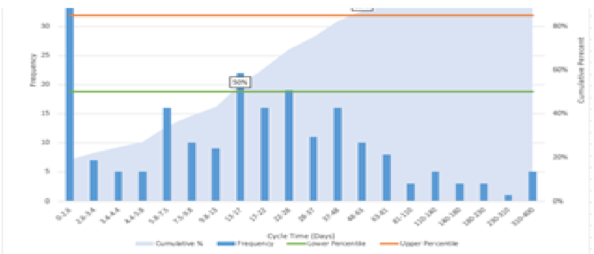
PROCESS PREDICTABILITY (CYCLE TIME)

Indicator Specification

Derived Measure Specification	
Derived Measure	For qualitative analysis: • Cycle Time • Percentile Lines • Cycle Time Frequency For quantitative analysis: • Average WIP • Average Throughput: The number of work items completed per unit of time. • Average Cycle Time: • Approximate Average Cycle Time (AACT):
Measurement Definition	For qualitative analysis: • Throughput: number of units completed per unit of time / exclusive percentile of cycle times for a set of work items (that complete the full process) between two dates. Upper and Lower throughput are calculated with the chosen percentiles. Used on the Cycle Time Scatterplot indicator. • Cycle Time: The date a work item was completed and the days it took to complete. • Percentile lines are calculated based on the percent to be used for analysis (e.g. 50th, 85th, and/or 95th). • Cycle Time Frequency: are frequencies of time based on bins/percentages For quantitative analysis: • Average WIP: Average of the WIP between two points in time • Average Throughput: Identify the time a work item enters the process and the time it exits the process. Count the number of completed items for a given period of time. • Average Cycle Time = Average Work in Progress / Average Throughput • Approximate Average Cycle Time = The horizontal distance between any two lines on a CFD for the time period chosen

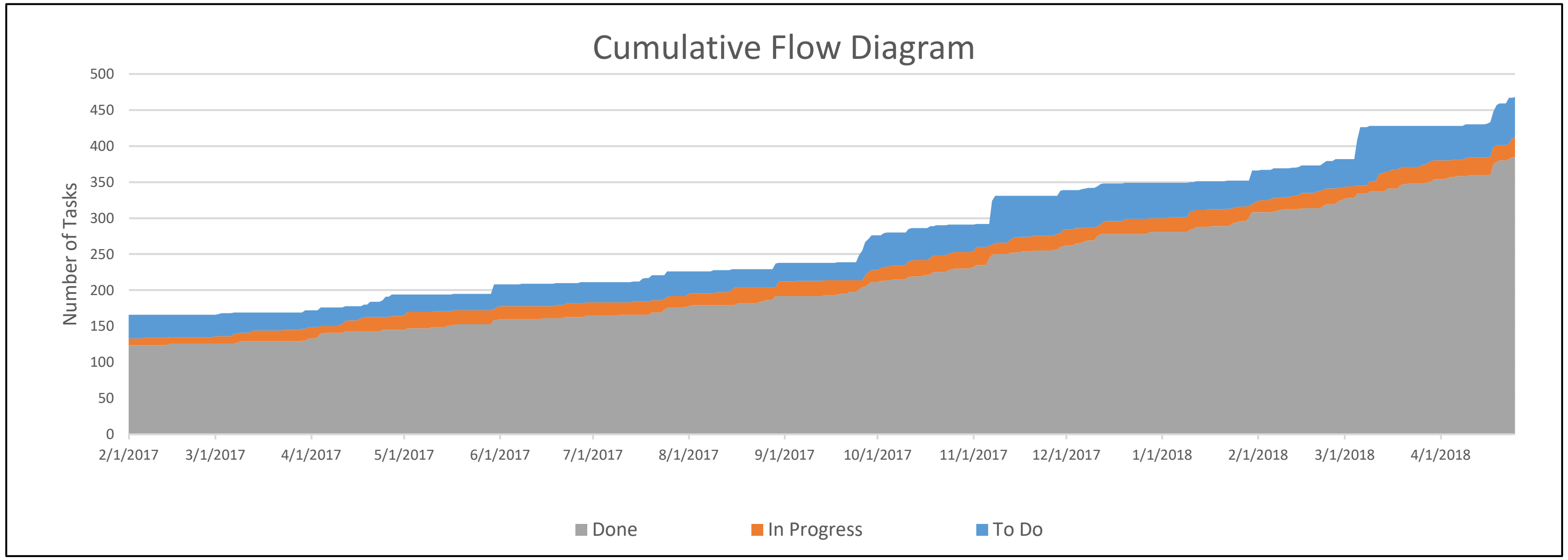
Scale

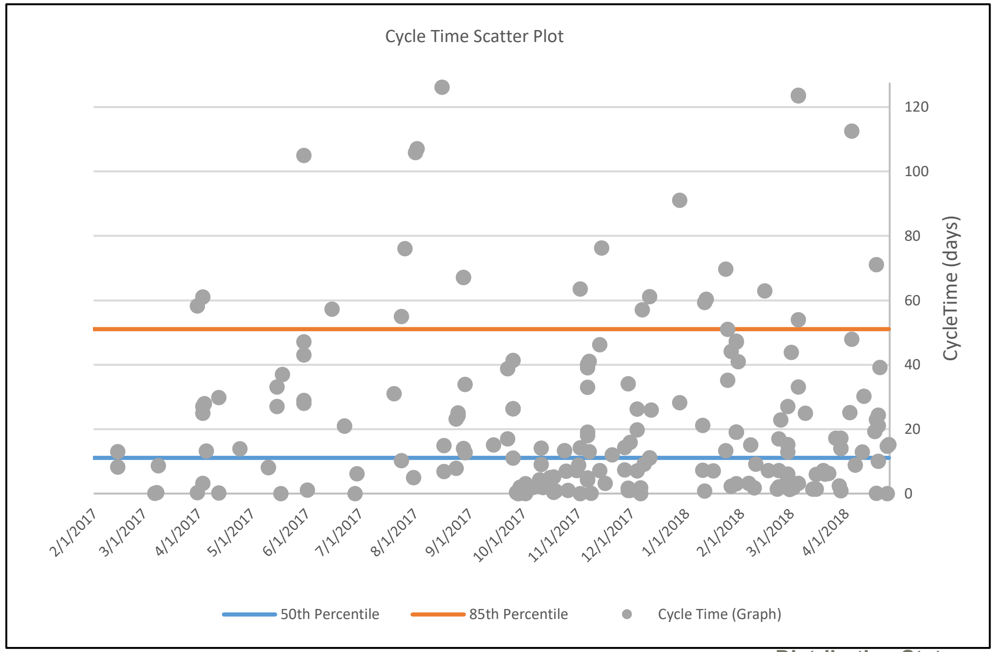
Unit of Meas

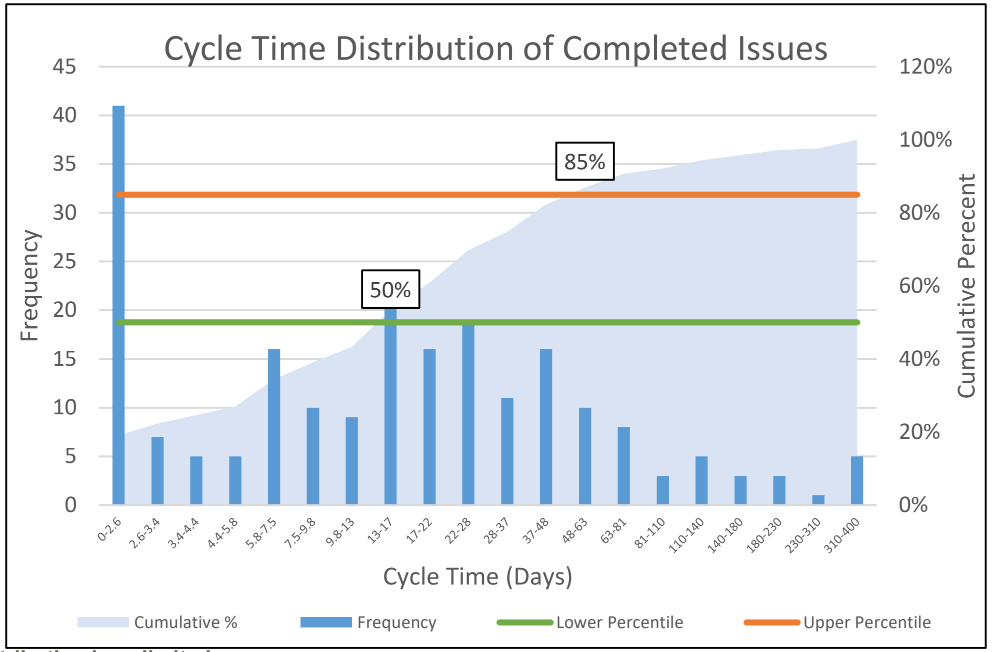


limited.

PROCESS PREDICTABILITY (CYCLE TIME)



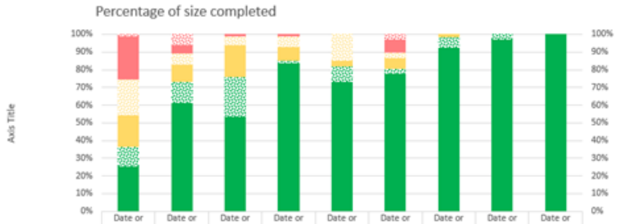




PROCESS EFFICIENCY

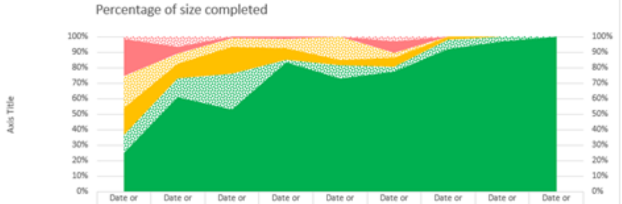
Indicator Specification

Indicator Description and Sample



	Date or Sprint1	Date or Sprint2	Date or Sprint3	Date or Sprint4	Date or Sprint5	Date or Sprint6	Date or Sprint7	Date or Sprint8	Date or Sprint9
Added (Def+Del)	1	5	0	0	0	2	0	0	0
Def+Del	19	4	1	1	0	5	0	0	0
Added (Incomplete)	16	5	4	4	9	2	0	0	0
Incomplete	14	8	13	5	2	4	1	0	0
Added Complete	9	10	17	1	5	2	4	2	0
Tasks Completed	20	50	40	56	44	52	60	68	75
Total Tasks	79	82	75	67	60	67	65	70	75

And/or



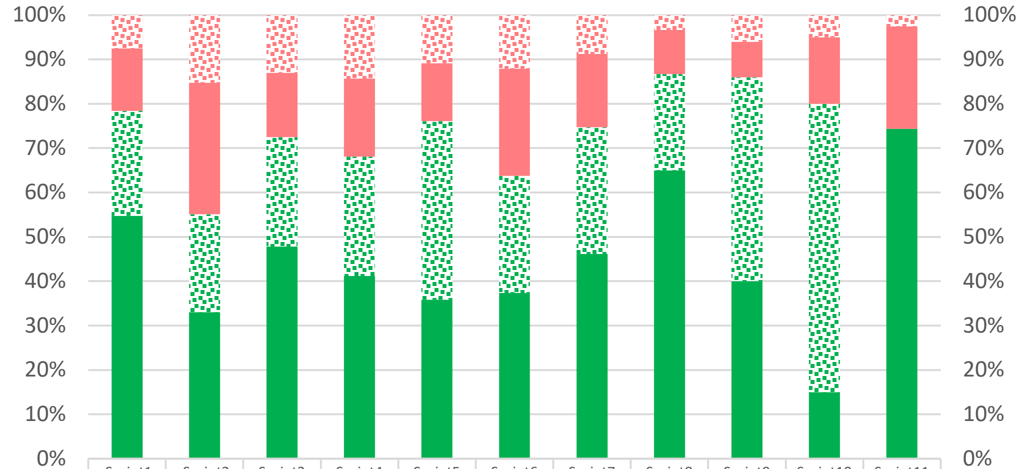
	Date or Sprint1	Date or Sprint2	Date or Sprint3	Date or Sprint4	Date or Sprint5	Date or Sprint6	Date or Sprint7	Date or Sprint8	Date or Sprint9
Added (Def+Del)	1	5	0	0	0	2	0	0	0
Def+Del	19	4	1	1	0	5	0	0	0
Added (Incomplete)	16	5	4	4	9	2	0	0	0
Incomplete	14	8	13	5	2	4	1	0	0
Added Complete	9	10	17	1	5	2	4	2	0
Tasks Completed	20	50	40	56	44	52	60	68	75
Total Tasks	79	82	75	67	60	67	65	70	75

Analysis Model	Compare the deviance of planned vs actual to determine if it meets project goals and identify areas of improvement as required.
Decision Criteria	Greater than (org standard for variance) on all work not completed or other percentage reflecting what is considered acceptable for project
Analysis Guidance	Examine breakdown of work item types to determine the focus of analysis and the formulation of a corrective action. Note: May want to separate task types to ensure more accurate picture of process performance (dev.) vs project performance (all aspects).

PROCESS EFFICIENCY

DEVCOM ARMAMENTS CENTER

Efficiency

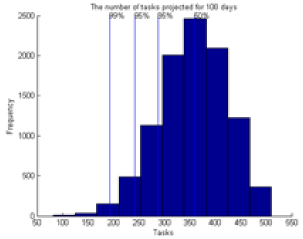
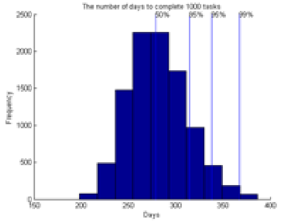
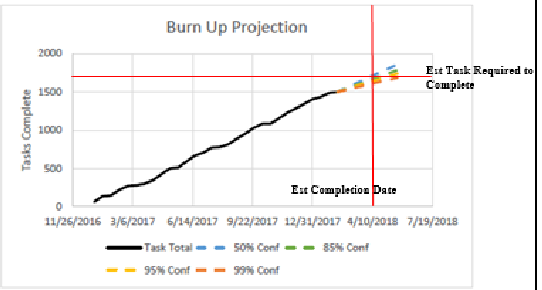


	Sprint1	Sprint2	Sprint3	Sprint4	Sprint5	Sprint6	Sprint7	Sprint8	Sprint9	Sprint10	Sprint11
Added (Def+Del)	15	18	9	17	10	11	8	2	3	1	1
Def+Del	28	35	10	21	12	22	15	6	4	3	9
Added (Incomplete)	0	0	0	0	0	0	0	0	0	0	0
Incomplete	0	0	0	0	0	0	0	0	0	0	0
Added Complete	47	26	17	32	37	24	26	13	23	13	0
Tasks Completed	109	39	33	49	33	34	42	39	20	3	29
Total Tasks	199	118	69	119	92	91	91	60	50	20	39

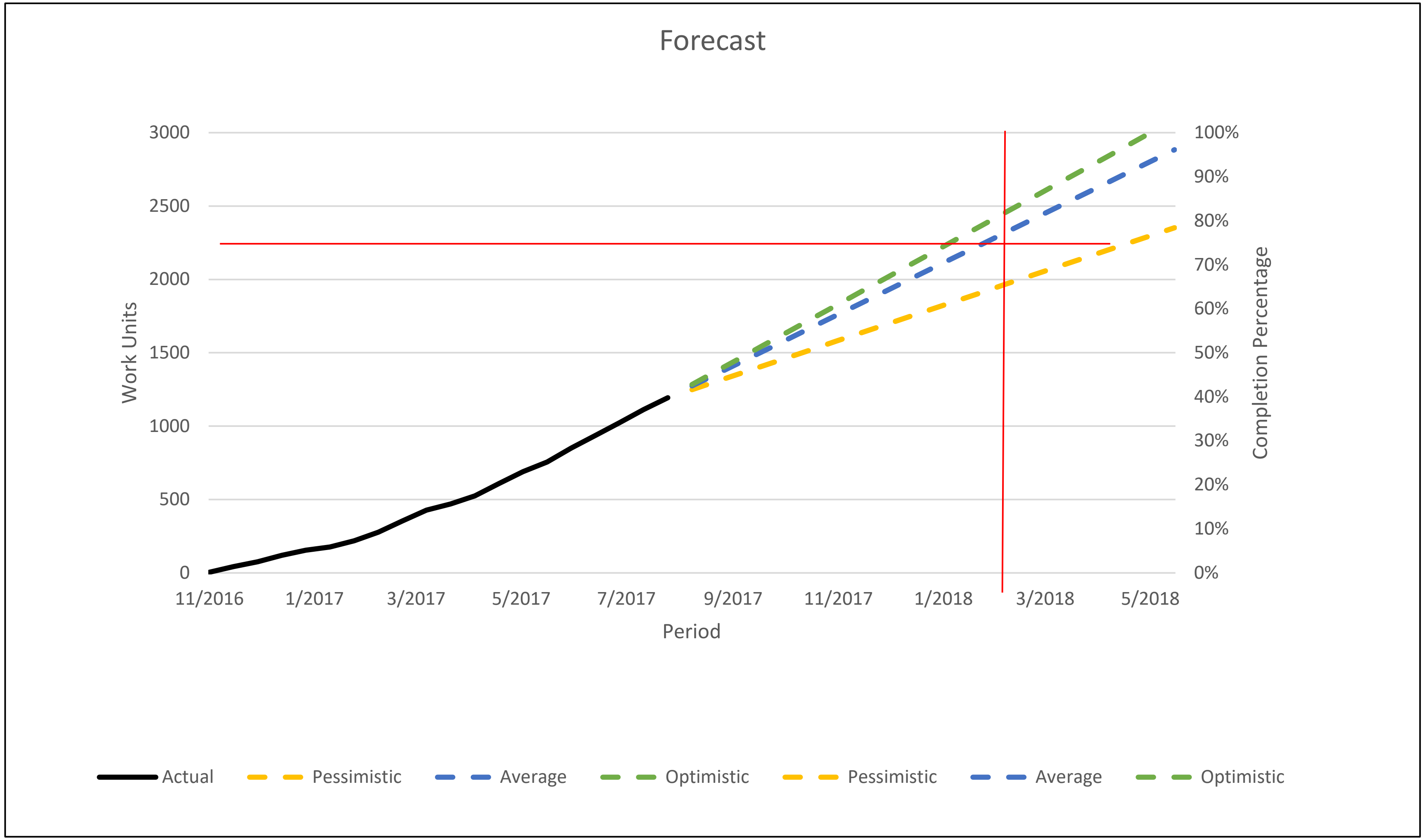
Efficiency

	Sprint1	Sprint2	Sprint3	Sprint4	Sprint5	Sprint6	Sprint7	Sprint8	Sprint9	Sprint10	Sprint11
Added (Def+Del)	15	18	9	17	10	11	8	2	3	1	1
Def+Del	28	35	10	21	12	22	15	6	4	3	9
Added (Incomplete)	0	0	0	0	0	0	0	0	0		
Incomplete	0	0	0	0	0	0	0	0	0		
Added Complete	47	26	17	32	37	24	26	13	23	13	0
Tasks Completed	109	39	33	49	33	34	42	39	20	3	29
Total Tasks	199	118	69	119	92	91	91	60	50	20	39

PROGRESS FORECAST

Indicator Specification	
Indicator Description and Sample	The Monte-Carlo simulation will take the known throughput data and create a representative distribution for either forecasted tasks or days. Forecast Tasks by Percentile  Percentile | Tasks: 50th | 358.57; 85th | 283.71; 95th | 237.57; 99th | 186.07 *MATLAB Output - Percentiles* Forecast Days by Percentile  Percentile | Days: 50th | 278.91; 85th | 314.46; 95th | 338.03; 99th | 367.35 *MATLAB Output - Percentiles* Burnup Chart and Projection 1500 Total Tasks Completed as of 02-11-18 Percentile | Task Completed | Total Tasks: 50% Conf | 359 | 1859; 85% Conf | 284 | 1784; 95% Conf | 238 | 1738; 99% Conf | 186 | 1686 Number of Days | Projected Date: 100 | 05-22-18 
Analysis Model	Are the variations outside the bounds of the project's expectations? Is the estimated completion date and required number of task needed to be completed within the bounds of the projection?
Decision Criteria	Investigate if the actual results vary from the projected results by more than

PROGRESS FORECAST



SPRINT DCM

Responsible Individual	Project Measurement Representative, Project Leader, Org. Measurement Team
Phase or Activity in which Analyzed	Ongoing
Source of Data for Analysis	JIRA
Tools Used in Analysis	Excel
Review, Report, or User	Projects- IPT or equivalent Organization- Monthly Measurement Report

Additional Information	
Additional Analysis Guidance	Dual validation of point of injection is recommended to ensure accuracy and quality. (Initial designation by tester/developer/tech-lead with subsequent validation (for example, validation of the defect data can be conducted during the Project's DRB))
Implementation Considerations	• Defect should have a unique ID # (key) independent of Task # assigned for a unique Sprint (will allow for tracking if in multiple sprints) • If a Legacy defect occurs then project should insert a column with the Release number as header, associated with the defect that has occurred. • Create the fields sprint originated and discovered within a defect form with a validation check to ensure the fields are populated

Complete this section for each base measure listed on the previous page.

Frequency of Data Collection	Monthly
Responsible Individual	Project Measurement Representative
Phase or Activity in which Collected	Ongoing
Tools Used in Data Collection	JIRA
Repository for Collected Data	PAL Measurement Repository

Data Analysis Procedure (for each Indicator)	
Frequency of Data Reporting	Monthly

SPRINT DCM

Defect Containment Matrix

		Sprint Defect Detected										Total Defects (Contained & Leaked)	Total Defects Contained	Total Defects Leaked
	Version	V8.0.0	V8.0.0	V8.0.0	V8.0.0	V8.0.0	V8.0.0	V8.0.0	V8.0.0	V8.0.0	V8.0.0			
Version	Sprint	Sprint 1	Sprint 2	Sprint 3	Sprint 4	Sprint 5	Sprint 6	Sprint 7	Sprint 8	Sprint 9	Sprint 10			
Sprint Defect Originated V8.0.0	Sprint 1	7	16	0	12	6	2	6	1	1	0	51	7	44
V8.0.0	Sprint 2		10	0	0	0	0	0	1	0	0	11	10	1
V8.0.0	Sprint 3			3	17	4	1	0	0	0	0	25	3	22
V8.0.0	Sprint 4				20	4	1	1	0	0	1	27	20	7
V8.0.0	Sprint 5					15	4	1	0	0	1	21	15	6
V8.0.0	Sprint 6						2	5	0	1	0	8	2	6
V8.0.0	Sprint 7							12	2	0	0	14	12	2
V8.0.0	Sprint 8								5	0	0	5	5	0
V8.0.0	Sprint 9									2	0	2	2	0
V8.0.0	Sprint 10										0	0	0	0
Total Defects per Sprint		7	26	3	49	29	10	25	9	4	2	164	76	88

Defect Containment Effectiveness 46.34%

Defect Leakage Effectiveness 53.66%

Sprint DCM Measurement Results

v8.1.0	Sprint 1	Sprint 2	Sprint 3	Sprint 4	Sprint 5	Sprint 6	Total	Contained	Leaked
Sprint 1	13	4	4	1	1		23	13	10
Sprint 2		6	3	0	1		10	6	4
Sprint 3			8	1	0		9	8	1
Sprint 4				8	1		9	8	1
Sprint 5					4		4	4	0
Sprint 6									
Total	**13**	**10**	**15**	**10**	**6**		**55**	**39**	**16**

	100%	60%	53%	80%	58%
		40%	20%	10%	14%
			27%	10%	0%
				0%	14%
					14%

	Containment	Total	%
	In Phase	39	70%
	1 Phase Leakage	9	16%
	2 Phase Leakage	4	7%
	3 Phase Leakage	2	4%
	3+ Phase Leakage	1	2%
	Total Leakage	16	30%

Containment Effectiveness = 70%

Leakage Effectiveness = 30%

What’s Next

Where do we go from here

WHAT'S NEXT

- **Pilot (In Progress)**
  - Limited due to the number of projects using Agile and being monitored
  - Cycle Time currently has no projects using this measure
- **Assessment (Dec 2019)**
  - Are the measures valid and sufficient
  - Are other metrics required
  - Are they useful at the organizational level - Aggregation
- **Adoption (Dec 2019)**
  - More Projects using Agile lifecycle
- **Baselining (Dec 2019 – Jan 2020)**
  - Aggregation
  - By Mission Type (Towed, Tracked, Dev, S&T, etc.)
- **Changes to Business Objectives (?)**
  - Evaluated Annual against all organizational measures