

# Program Modeling and CFD Providing Insights Into the Most Chaotic Program



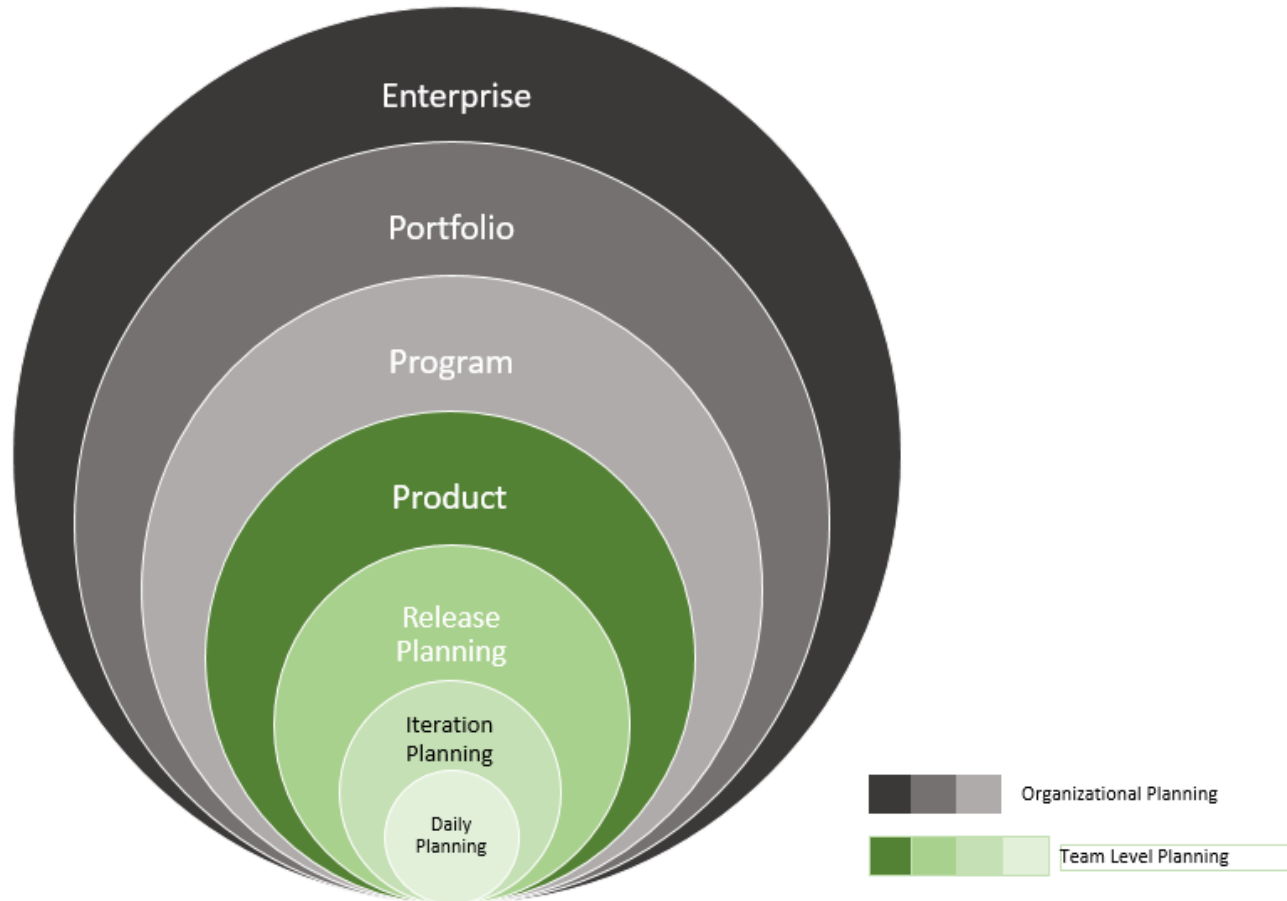
**April 2016**

# Session Overview

- Opening the session
  - Introduction
  - Cover the session agenda
- Progressive Program Modeling
  - Understanding the challenges of program management at an enterprise level.
  - Defining the concepts of Progressive Program Modeling
  - Progressive Program Modeling in practice.
- Cumulative Flow Diagrams
  - Understanding what a cumulative flow diagram is and how to read it.
  - Using it as a tool to monitor teams, and programs delivery
    - Identifying issue in with delivery
      - Organizational
      - Team behaviors
    - Data to seed retrospective conversation
- Closing remarks and Q and A

- Understanding of how to read and use cumulative flow diagrams to monitor teams, and program delivery
- How to manage program using Progress Program Modeling.

# Levels of Planning



# Visualizing the Work

- Opportunity Backlog
  - Collection of Opportunities a company may want to complete during a fiscal year
    - At the portfolio level this could be Regulations and Risk
- Road Maps
  - Road Maps are visualization of what work is target for a specific time.
    - Road Mapping can be used at any level of planning
  - Using Road Mapping at the Organizational levels of Planning aids in collision detection across Multiple delivery teams, or Specific area of an application.



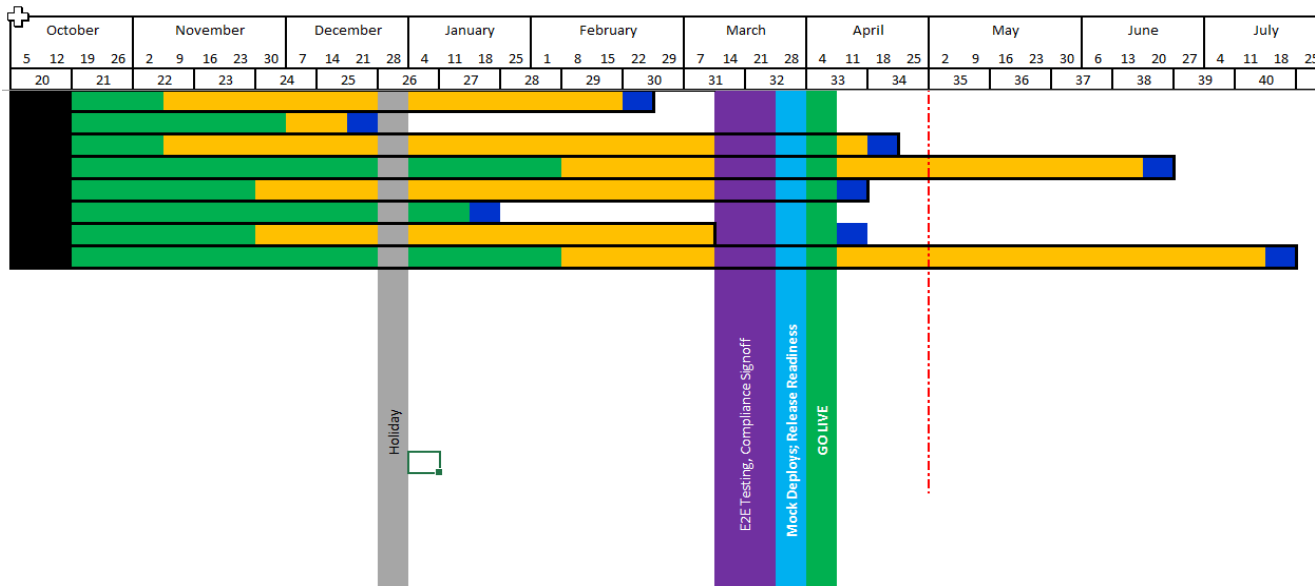
# Concepts of Progressive Program Modeling

Progressive Program Modeling – Is a forecasting tool used to monitor the health of portfolio, program, product.

- Using models allows for detection of issue before they happen
- Monitor scope changes
- Monitor teams backlog health
- Show how new discovery's or feedback will impact delivery
- Project the ability meet specific Milestones for thing like enterprise release, end to end testing

# Progressive Program Modeling in Practice

Product Area	Team	Avg Velocity	Scheduled (in # of sprints pts)		Not Scheduled (in pts)		UnRefined (in story ct)	Unknown (story ct)	BEST Case	Probable Case	Worst Case	Backlog Health	Projected Delivery
			Scheduled (in pts)	# of sprints groomed	Not Scheduled (in pts)	# of sprints not assigned							
Apply	Voltron	20	14	0.7	160	8			2/29/2016	2/29/2016	2/29/2016	OK	GOOD
Apply	Anubis	30	111	3.7	33	1.1			1/4/2016	1/4/2016	1/4/2016	GOOD	GOOD
Communications	Tunder Cats	14	19	1.35714	154	11			4/25/2016	4/25/2016	4/25/2016	OK	NOT GOOD
Customer Servicing	Care Bears	15	116	7.73333	148	9.86666667			7/4/2016	7/4/2016	7/4/2016	GOOD	NOT GOOD
Payments	War Hammers	17	51	3	137	8.05882353			4/11/2016	4/11/2016	4/11/2016	GOOD	AT RISK
3pi	A-Team	12	83	6.91667	0	0			2/1/2016	2/1/2016	2/1/2016	GOOD	GOOD
PET	Scrum Dog Millionaires	25	82	3.28	223	8.92			4/25/2016	4/25/2016	4/25/2016	GOOD	AT RISK
Collections	Benny and the Jets	8	68	8.5	91	11.375			8/1/2016	8/1/2016	8/1/2016	GOOD	NOT GOOD





# Kanban

## 1. Visualize Work

By creating a visual model of your work and workflow, you can observe the flow of work moving through your Kanban system. Making the work visible—along with blockers, bottlenecks and queues—instantly leads to increased communication and collaboration.

## 2. Limit Work in Process

By limiting how much unfinished work is in process, you can reduce the time it takes an item to travel through the Kanban system. You can also avoid problems caused by task switching and reduce the need to constantly reprioritize items.

## 3. Focus on Flow

By using work-in-process (WIP) limits and developing team-driven policies, you can optimize your Kanban system to improve the smooth flow of work, collect metrics to analyze flow, and even get leading indicators of future problems by analyzing the flow of work.

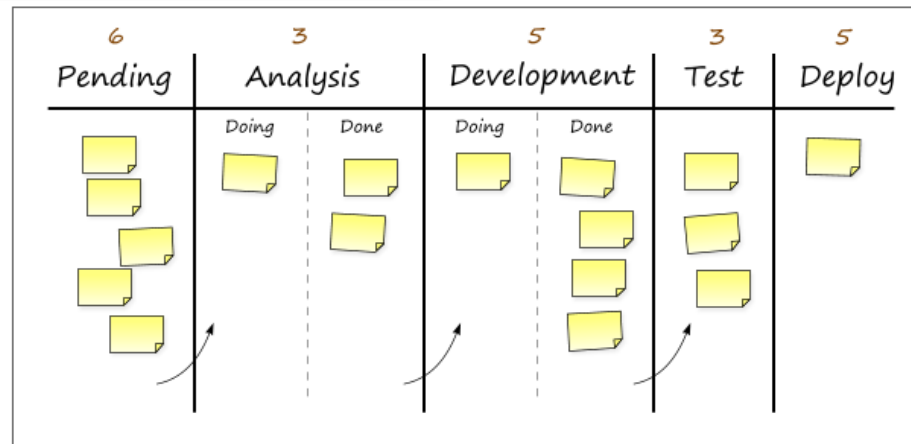
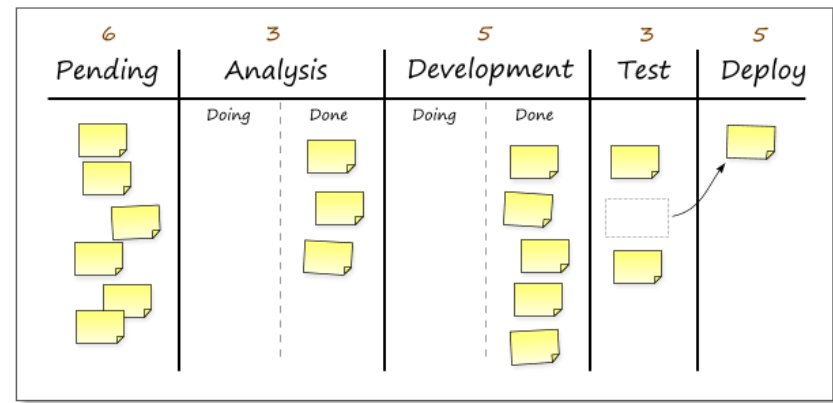
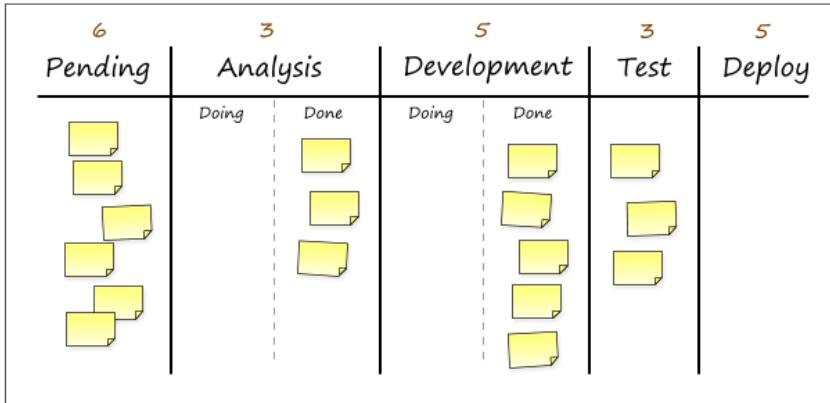
#### 4. Make Process Policies Explicit

By everyone really understands what you are doing now and what your goals are, then you can begin to make decisions regarding change that will move you in a positive direction. The choices will be more rational, empirical, objective discussion of issues. This is more likely to facilitate consensus around improvement suggestions.

#### 5. Continuous Improvement

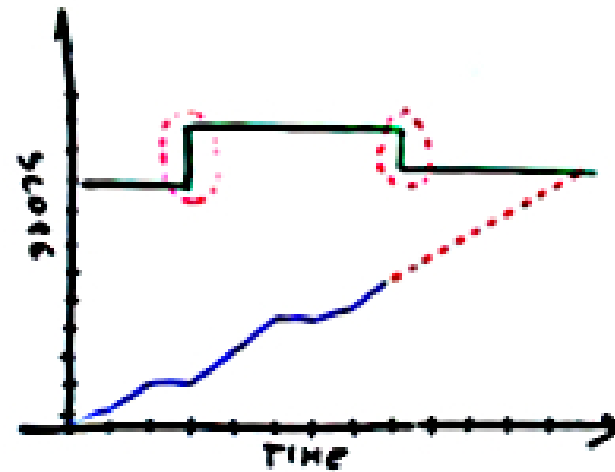
Once your Kanban system is in place, it becomes the cornerstone for a culture of continuous improvement. Teams measure their effectiveness by tracking flow, quality, throughput, lead times and more. Experiments and analysis can change the system to improve the team's effectiveness.

# Kanban

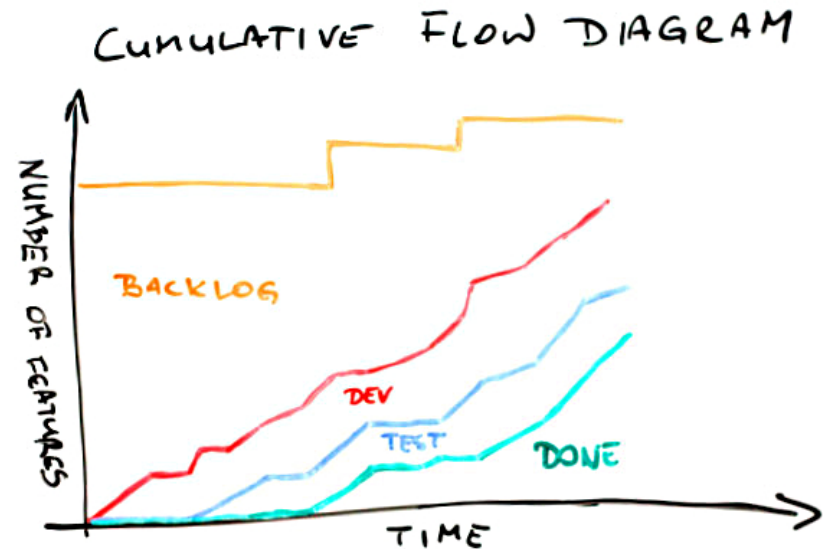
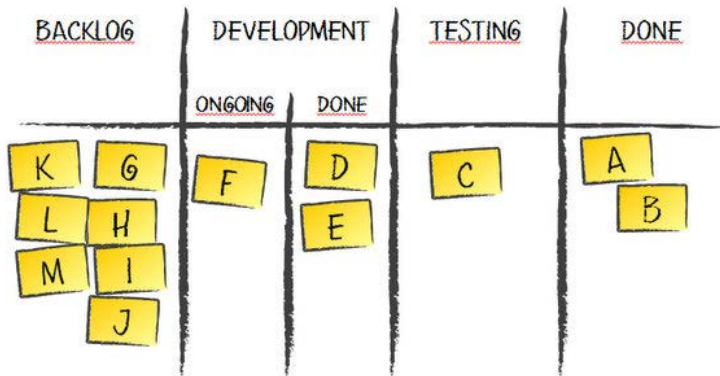


# Cumulative Flow Diagram

Single page chart that can give you a quick overview of what's happening in a project or product such as how much work is done, ongoing and in backlog, what is the pace of progress, and Potential issue with flow of the work to allow for further investigation.



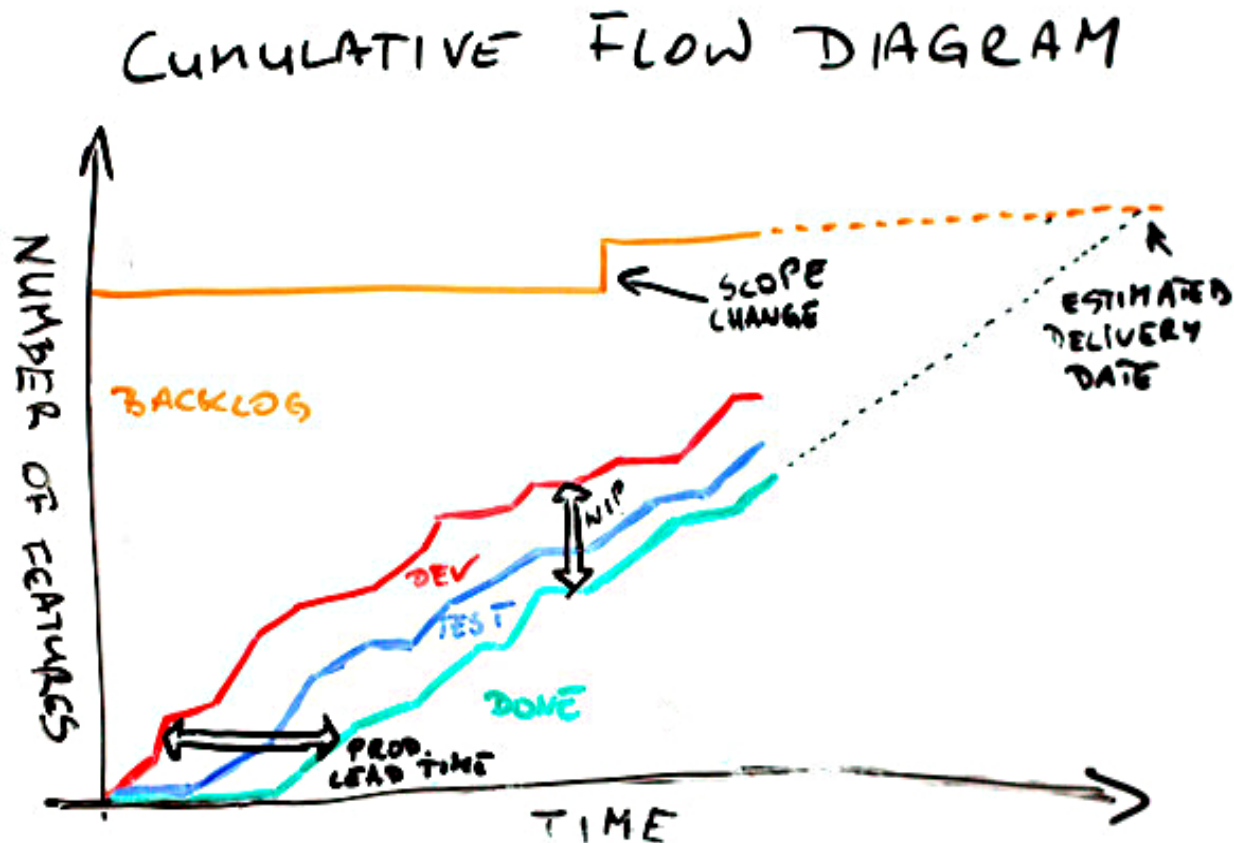
# Cumulative Flow Diagram



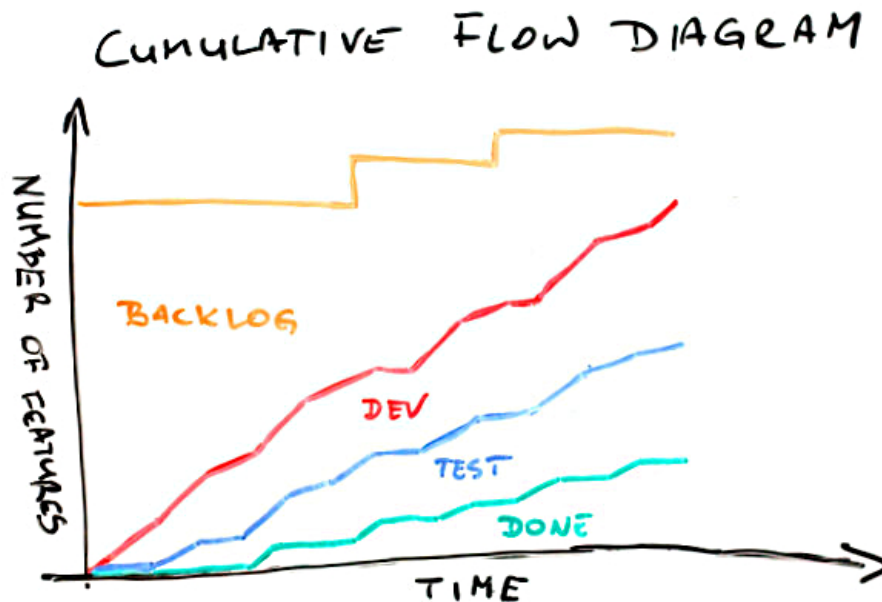
## Meaning of the lines

- The green one shows how many items have been delivered over time.
- Between the blue and the green curves is stuff that is in testing.
- The area between the red and the blue lines shows how much stuff is in development (either ongoing or done).
- Orange line is the backlog – how many items weren't yet started.

# Cumulative Flow Diagram



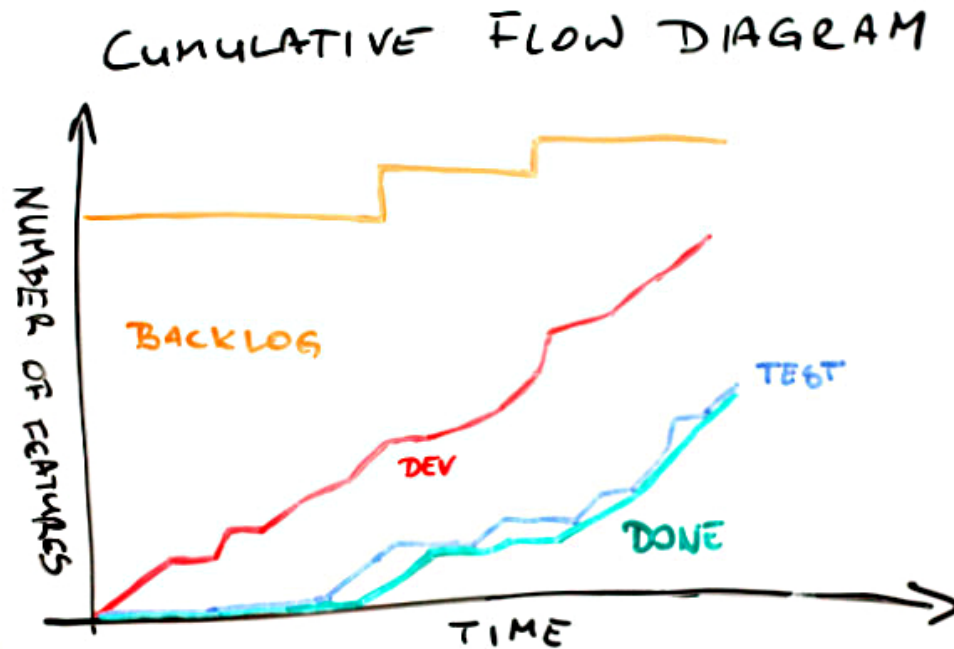
# Cumulative Flow Diagram



Increased Work in progress overall

- Increase in resources on the team
- Increase in blocked stories

# Cumulative Flow Diagram

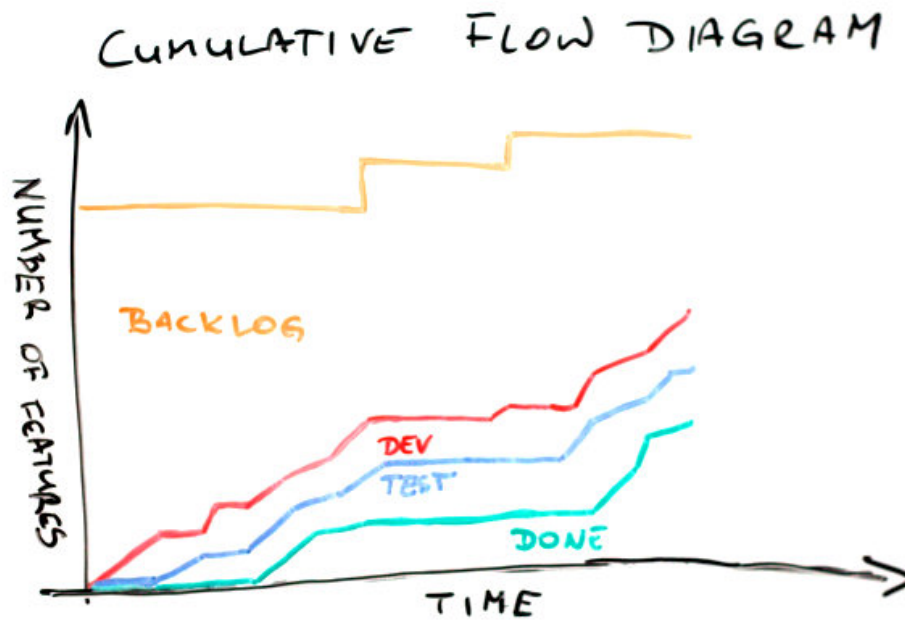


## High WIP In Development

- Few stories could be ready to test.
- Stories waiting for testing but availability of testers is very limited



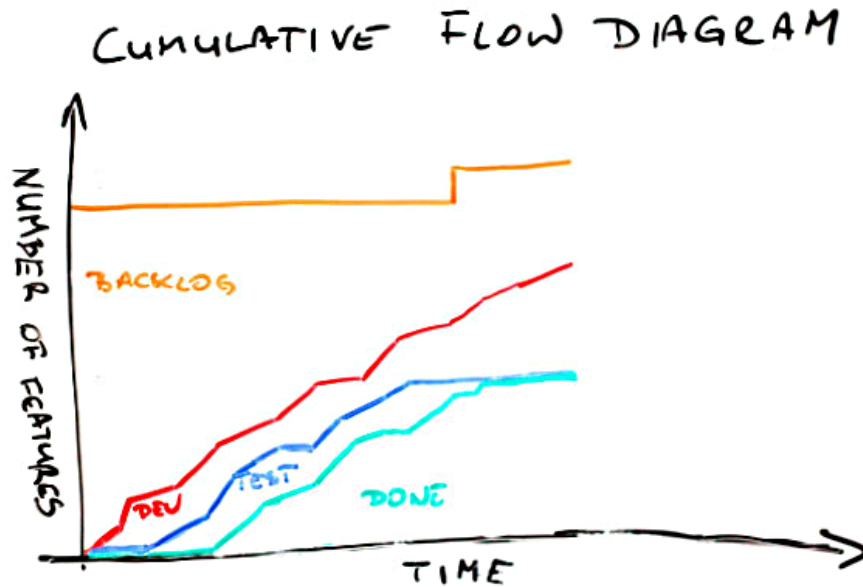
# Cumulative Flow Diagram



## WIP Flatted Out

- All Day company events
- Poor Planning of team vacations
- Environment Outage

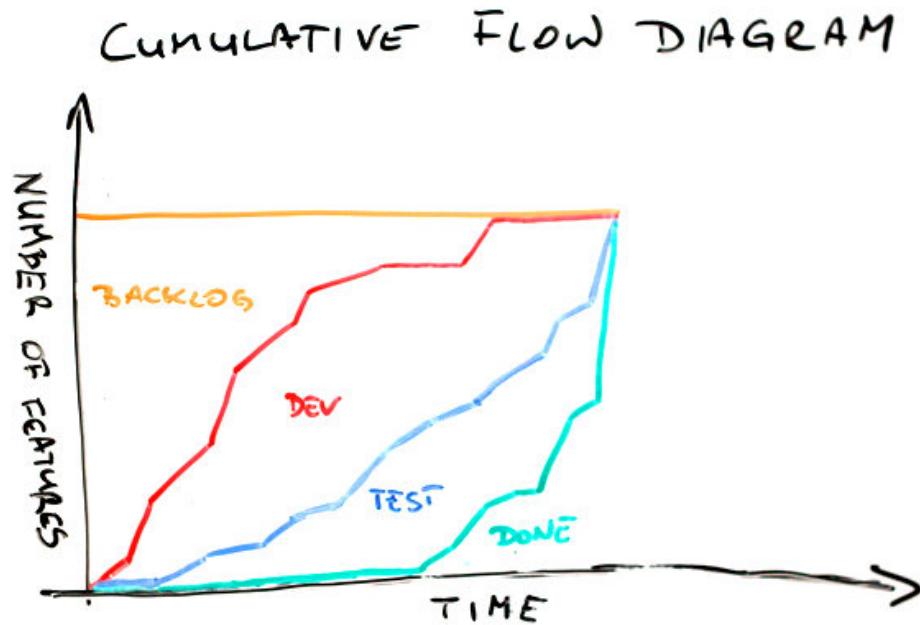
# Cumulative Flow Diagram



Development Continues Test Flattens out

- Poor Planning of team vacations
- Single point of frailer resource
- Environment Outage
- Development team not observing WIP

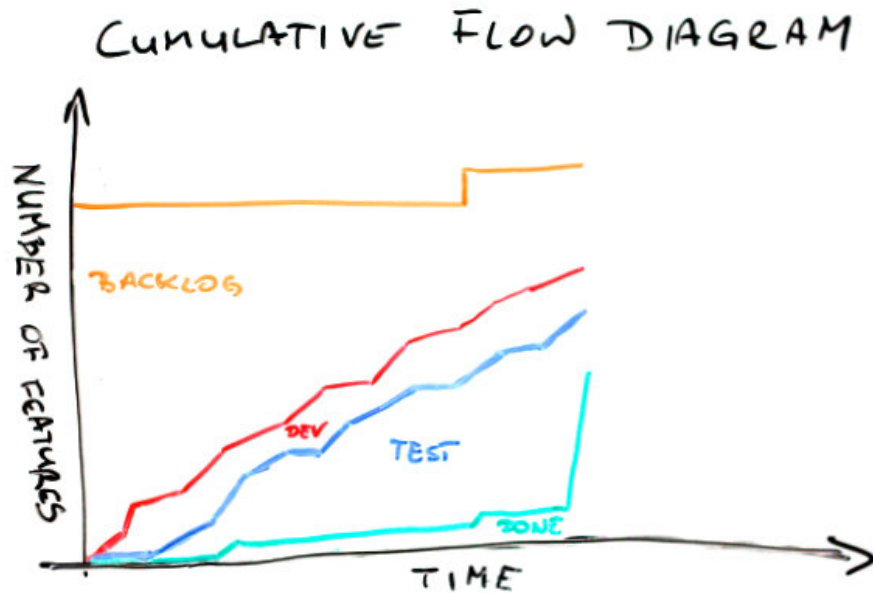
# Cumulative Flow Diagram



Time boxed work

- Development with no WIP limits
- Mad Dash to the end of the sprint to finish it all.

# Cumulative Flow Diagram



Work is Stacking waiting to be deployed  
Normal for big release activities  
Like Enterprise release.  
Extended UAT cycles.

# Questions ?

# James Gifford

Twitter : @scrummando

Blog: [www.scrummando.com](http://www.scrummando.com)

Email: [james.gifford@genesisconsulting.com](mailto:james.gifford@genesisconsulting.com)

Upcoming Speaking events

Heart of Agile Philadelphia - 5/23-5/24

<http://heartofagile.com/heart-of-agile-philadelphia/>

- 34 Session over 2 Days
- Ken Schwaber and Alistair Cockburn Live

