So why do projects slip?

Ron Holohan, PMP MBA

August 25th, 2009

UniForum Chicago
The International Association of Open Systems Professionals

http://www.pm411.org
ron@pm411.org

About Ron

- Director of Program Management at Shure, Inc

http://www.pm411.org
ron@pm411.org

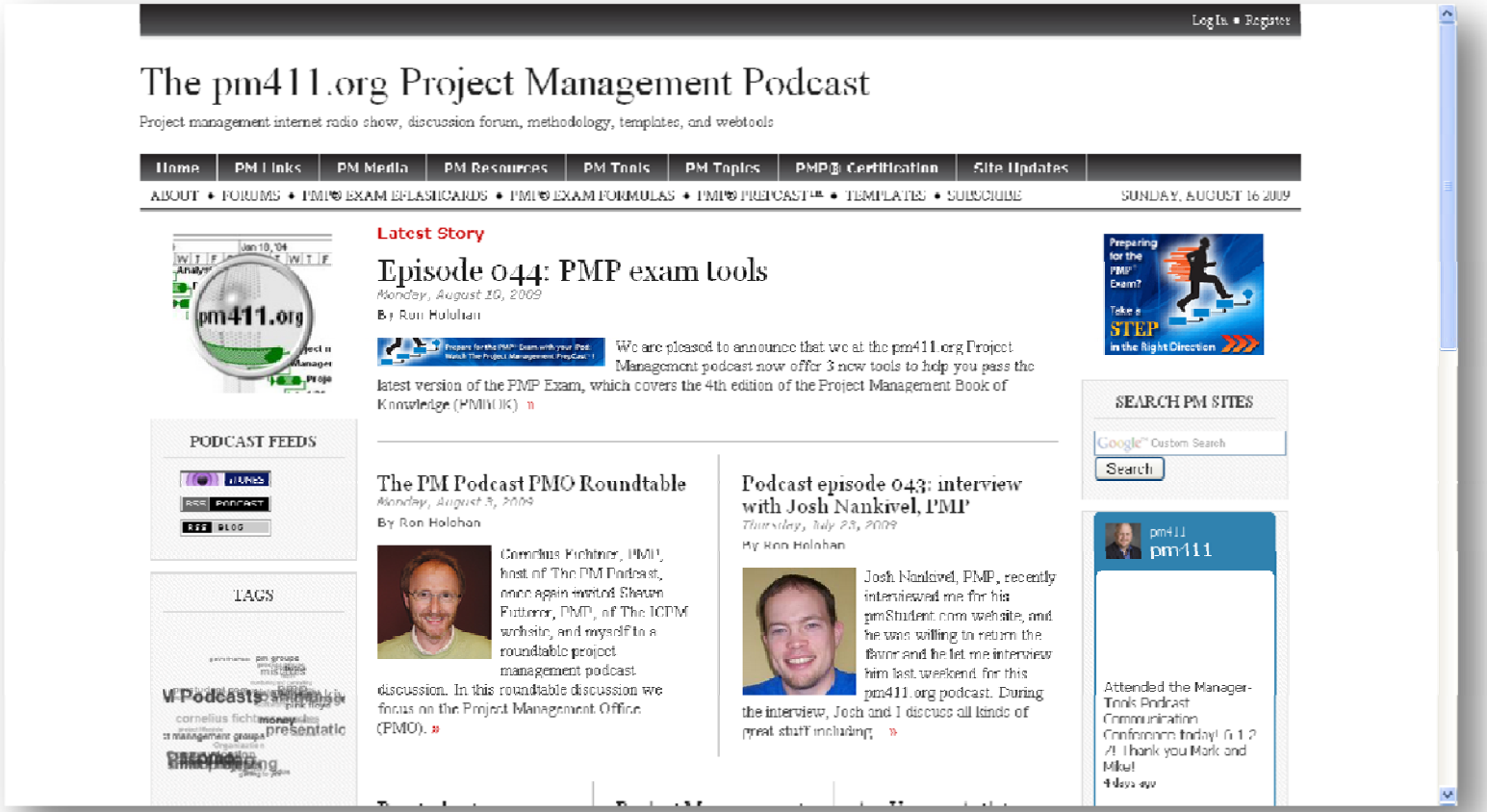
Software Development

- Development Process
  - Agile Development
- Software Development Tools
  - Code Collaborator for Code Reviews
  - Subversion Version Control
  - Deki Distributed Tools
  - Jira Issue Tracker w/GreenHopper Agile plugin
  - C++
  - Adobe Flex
- Embedded Hardware
  - Freescale
  - IBM PowerPC
  - Atmel XMEGA

http://www.pm411.org
ron@pm411.org

About Ron

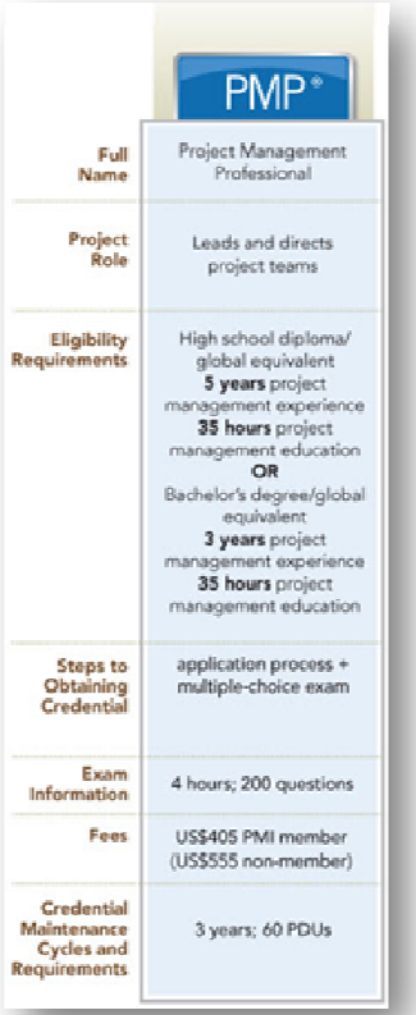
- Host of The pm411.org Project Management Podcast



http://www.pm411.org
ron@pm411.org

About the PMP

1. **Shows your commitment to the project management profession.**
2. **PMI credentials recognize your knowledge, skills and abilities.**
3. **PMI credentials reflect achievement.**
4. **Can lead to greater earnings.**
5. **Can lead to career opportunities and advancement.**
6. **Prepares you for greater job responsibilities.**
7. **Improves skills and knowledge.**
8. **Builds self confidence.**
9. **Allows for greater recognition from peers.**
10. **Enhances the profession.**



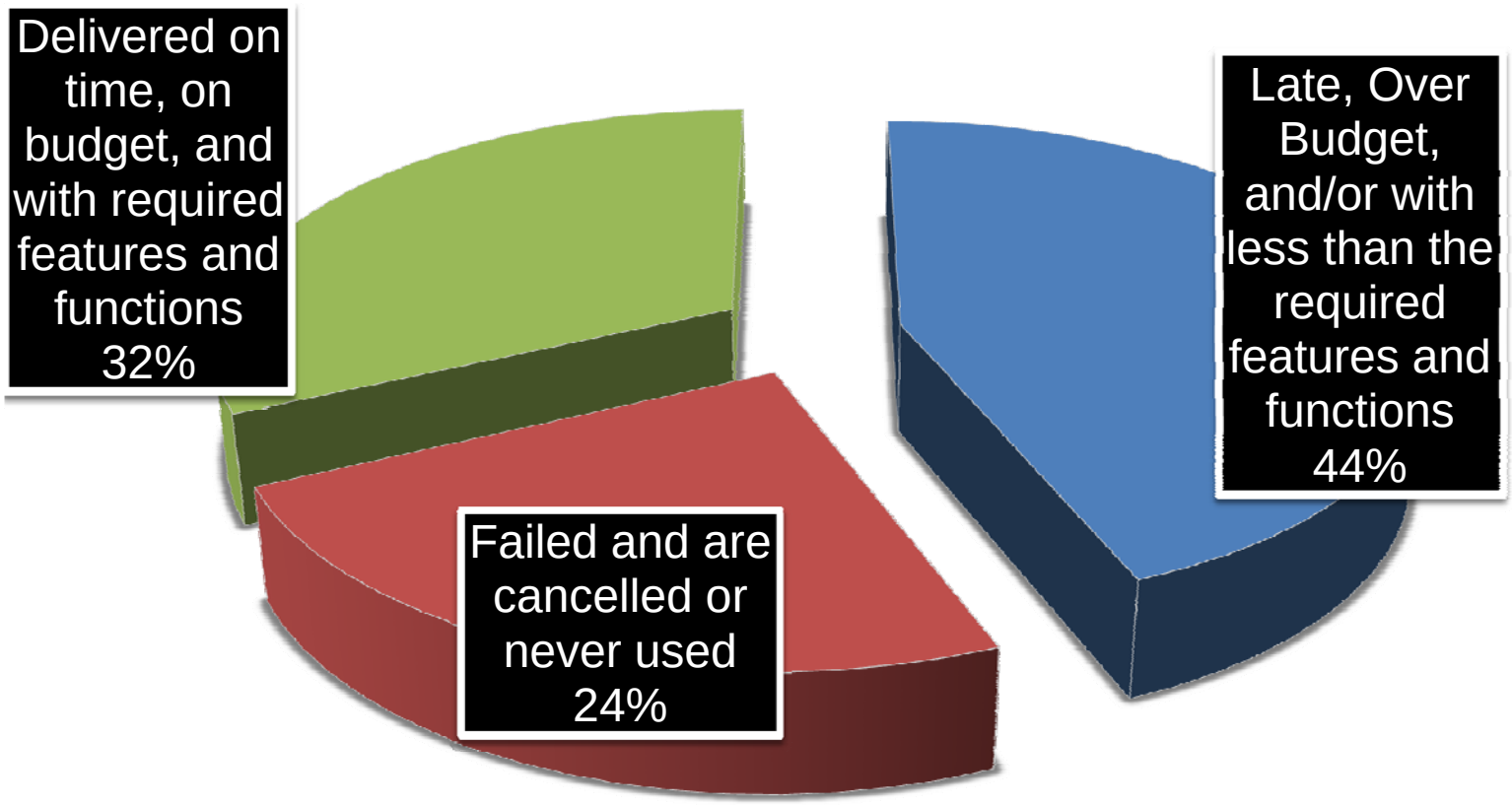
	PMP®
Full Name	Project Management Professional
Project Role	Leads and directs project teams
Eligibility Requirements	High school diploma/ global equivalent **5 years** project management experience **35 hours** project management education **OR** Bachelor's degree/global equivalent **3 years** project management experience **35 hours** project management education
Steps to Obtaining Credential	application process + multiple-choice exam
Exam Information	4 hours; 200 questions
Fees	US$405 PMI member (US$555 non-member)
Credential Maintenance Cycles and Requirements	3 years; 60 PDUs

http://www.pm411.org
ron@pm411.org

So why do projects slip?

http://www.pm411.org
ron@pm411.org

The sad truth about projects…

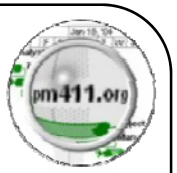


Source: Standish Group, 2009

http://www.pm411.org
ron@pm411.org

So why do projects slip?

http://www.pm411.org
ron@pm411.org



Because teams focus on protecting

the tasks

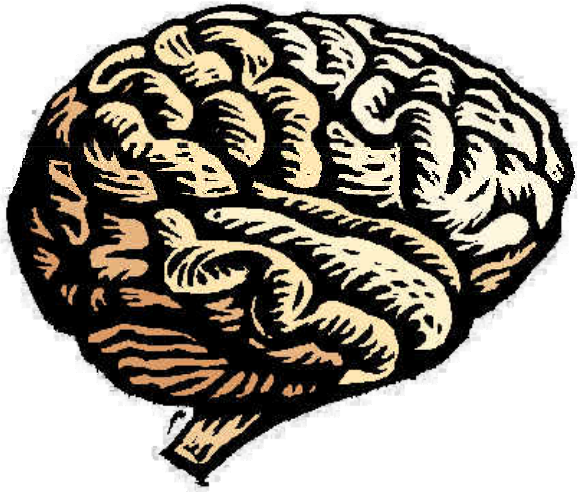
instead of

the project

http://www.pm411.org
ron@pm411.org

So why do we focus on tasks?

http://www.pm411.org
ron@pm411.org

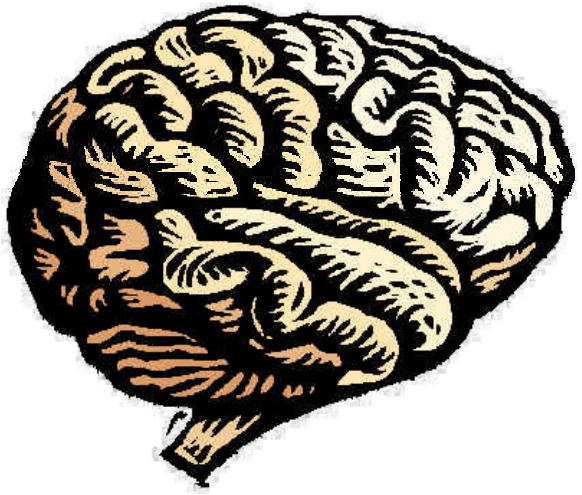


Because of Human Behavior

http://www.pm411.org
ron@pm411.org

3 Behaviors that affect projects

1. Student Syndrome
   - We don't usually start right away due to other priorities
2. Parkinson's Law
   - We tend to not turn things in early for fear of new benchmarks
3. Bad Multi-tasking
   - We jump between "critical" tasks



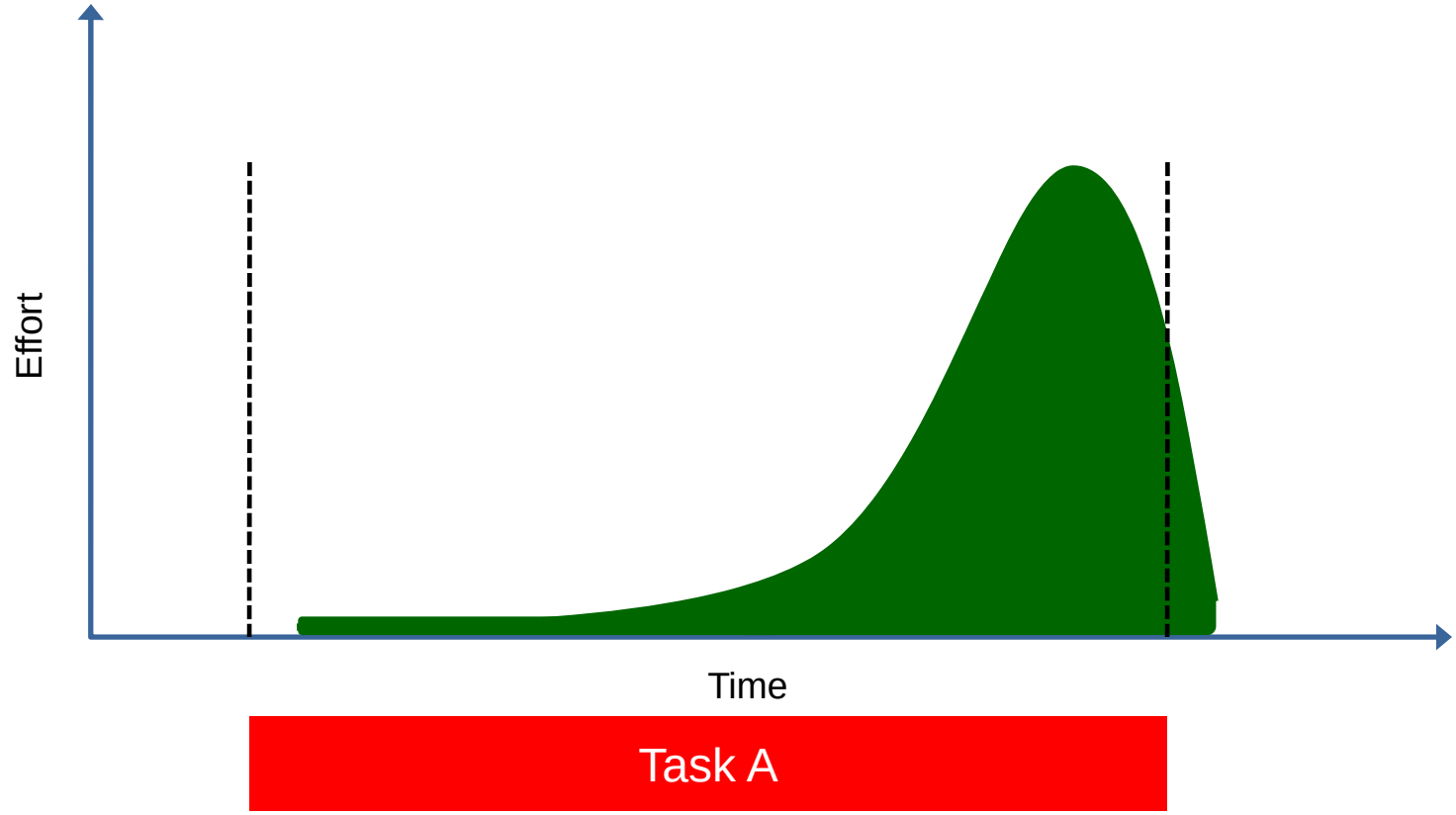
http://www.pm411.org
ron@pm411.org

Behavior 1: Student Syndrome

People will start to fully apply themselves to a task just at the last possible moment before a deadline. This leads to wasting any buffers built into individual task duration estimates.

http://www.pm411.org
ron@pm411.org

Student's Syndrome



http://www.pm411.org
ron@pm411.org

So, what is the solution?

- Break down tasks into the 8/80 rule
  - Tasks no less than 8 hours (1 day) in duration
  - Tasks no more than 80 hours (2 wks) in duration
- Focus on setting aggressive durations based on 50% chance of completion
- Remove any safety time from tasks (reduce initial task durations to allow 50% chance of on-time completion)
- Place this safety time at the end of the project

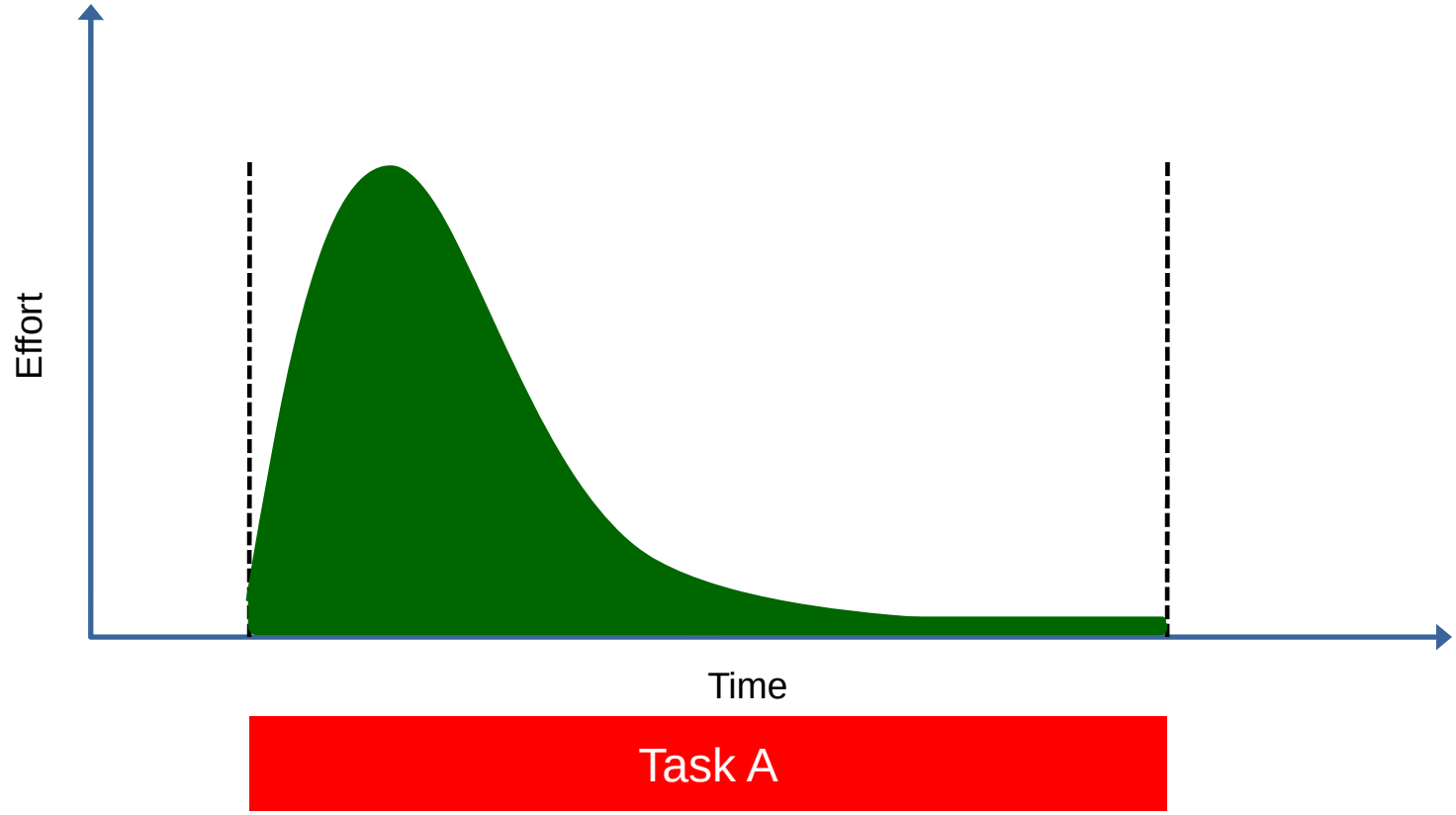
http://www.pm411.org
ron@pm411.org

Behavior 2: Parkinson's Law

- Work expands to fill the time available
- If you make the "box" bigger, work will tend to "expand" to fill the entire box.

http://www.pm411.org
ron@pm411.org

Parkinson's Law



http://www.pm411.org
ron@pm411.org

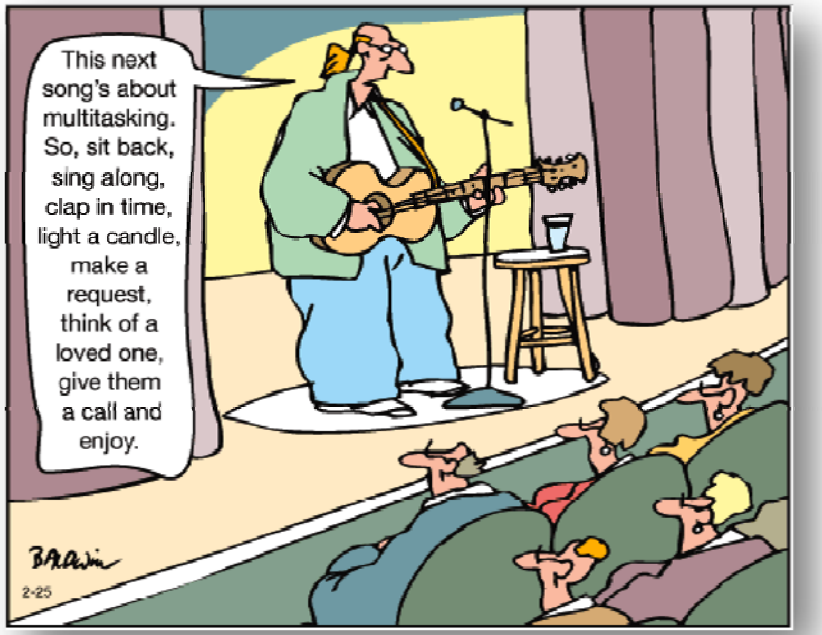
So, what is the solution?

- Remove any safety time from tasks (reduce initial task durations to allow 50% chance of on-time completion)
- Place this safety time at the end of the project
- Remember:
  - The chance that something will go wrong on any one task is high
  - The chance that something will go wrong on every task is quite low.

http://www.pm411.org
ron@pm411.org

Behavior 3: Bad Multitasking

- When resources constantly switch focus between critical path tasks on multiple projects AND someone is waiting for the output of their task before they can do their work.



http://www.pm411.org
ron@pm411.org

So, why do we do it?

- Boredom of working on one thing at a time
- Poor work prioritization
- Customer / Manager demands / pressure
- We are taught that it is a admirable skill set

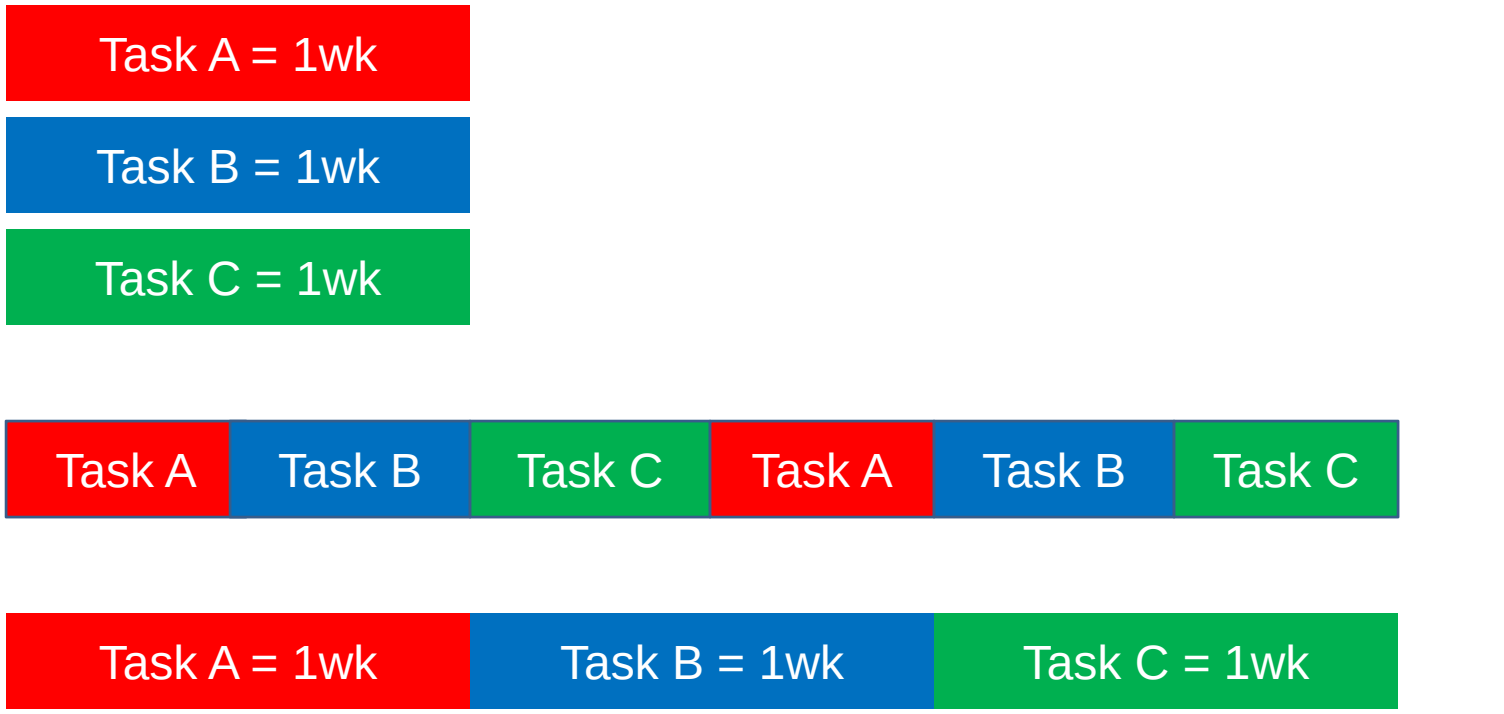
http://www.pm411.org
ron@pm411.org

So, why not Multitask?

- The brain is not wired for it
- Multitasking *always* makes a task take longer than it should.
- Forces people to give longer task duration estimates than necessary.
- Quality problems also escalates.
  - What was done / not done?
  - Rush to return to previous task.
  - Overlook small details.
- Stress / Frustration / No sense of completion

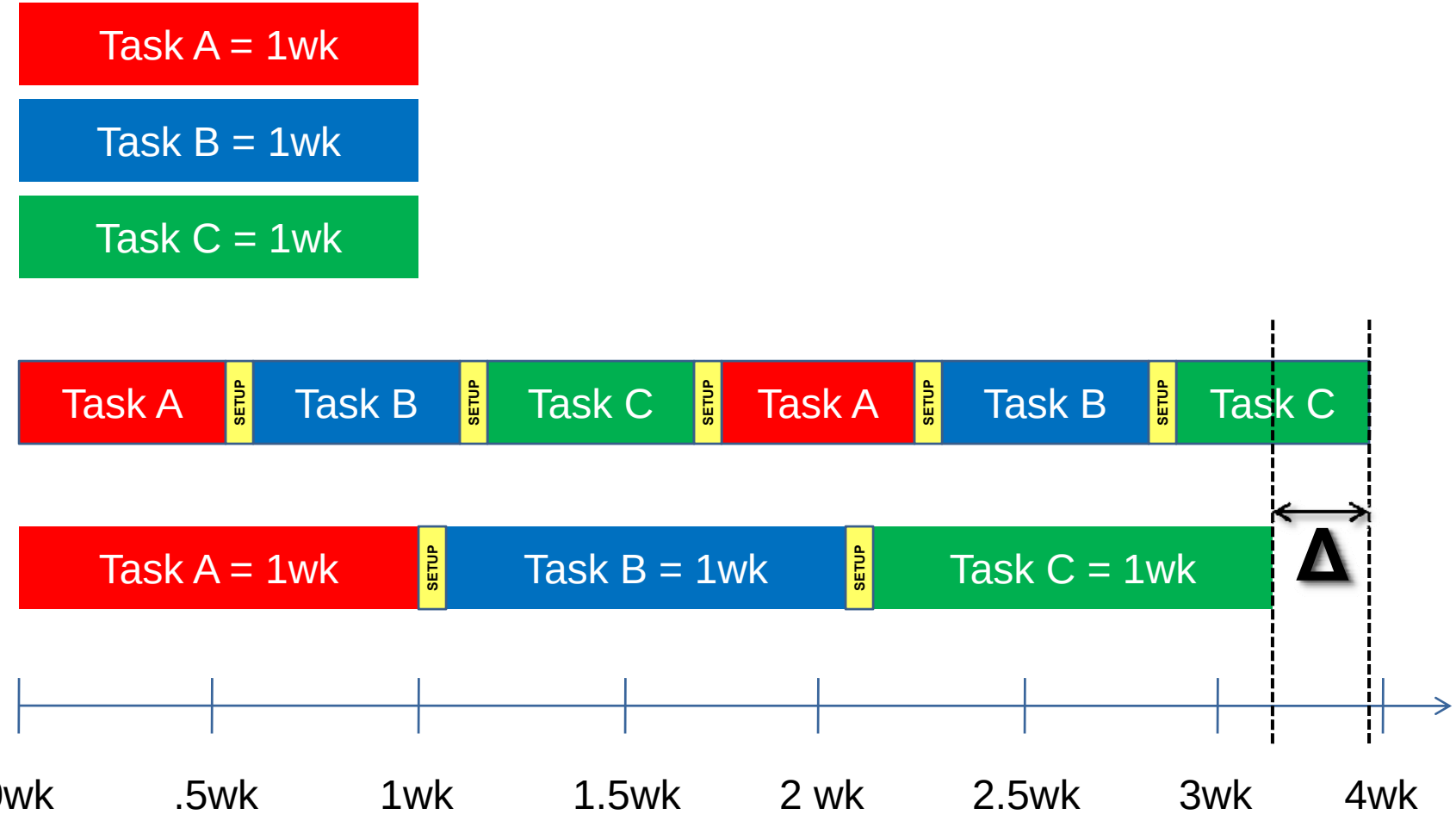
http://www.pm411.org
ron@pm411.org

Bad Multi-tasking Example



http://www.pm411.org
ron@pm411.org

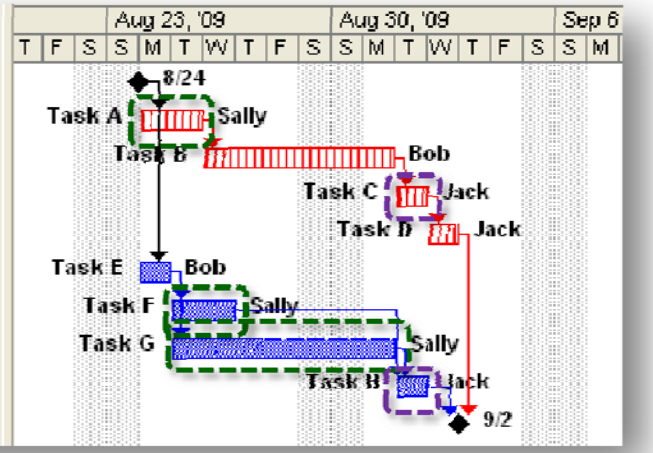
Bad Multi-tasking Example



http://www.pm411.org
ron@pm411.org

Resource Leveling Example

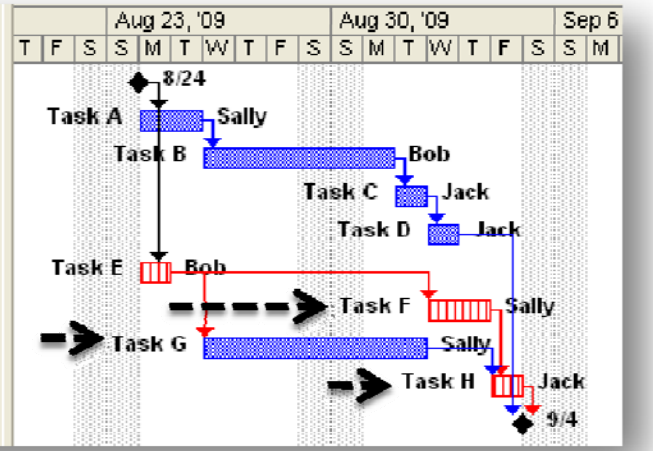
Task Name	Duration	Start	Finish	Predecessors	Resource Names
Start	0 days	8/24/09	8/24/09		
Task A	2 days	8/24/09	8/25/09	1	Sally
Task B	4 days	8/26/09	8/31/09	2	Bob
Task C	1 day	9/1/09	9/1/09	3	Jack
Task D	1 day	9/2/09	9/2/09	4	Jack
Task E	1 day?	8/24/09	8/24/09	1	Bob
Task F	2 days	8/25/09	8/26/09	6	Sally
Task G	5 days	8/25/09	8/31/09	6	Sally
Task H	1 day?	9/1/09	9/1/09	8,7	Jack
End	0 days	9/2/09	9/2/09	5,9	



http://www.pm411.org
ron@pm411.org

Example – Resources Leveled

Task Name	Duration	Start	Finish	Predecessors	Resource Names
Start	0 days	8/24/09	8/24/09		
Task A	2 days	8/24/09	8/25/09	1	Sally
Task B	4 days	8/26/09	8/31/09	2	Bob
Task C	1 day	9/1/09	9/1/09	3	Jack
Task D	1 day	9/2/09	9/2/09	4	Jack
Task E	1 day?	8/24/09	8/24/09	1	Bob
Task F	2 days	9/2/09	9/3/09	6	Sally
Task G	5 days	8/26/09	9/1/09	6	Sally
Task H	1 day?	9/4/09	9/4/09	8,7	Jack
End	0 days	9/4/09	9/4/09	5,9	



http://www.pm411.org
ron@pm411.org

So, what is the solution?

- Break up your project into tasks based on 8/80 rule
- Resource level your project where possible
- Serialize work where possible
- Staff the project with the right people at the right time.
- Hold frequent (daily, 3x / week, 2x /week) short standup meetings to insure people are focused
- Focused = No other tasks, no meetings, no email!

http://www.pm411.org
ron@pm411.org

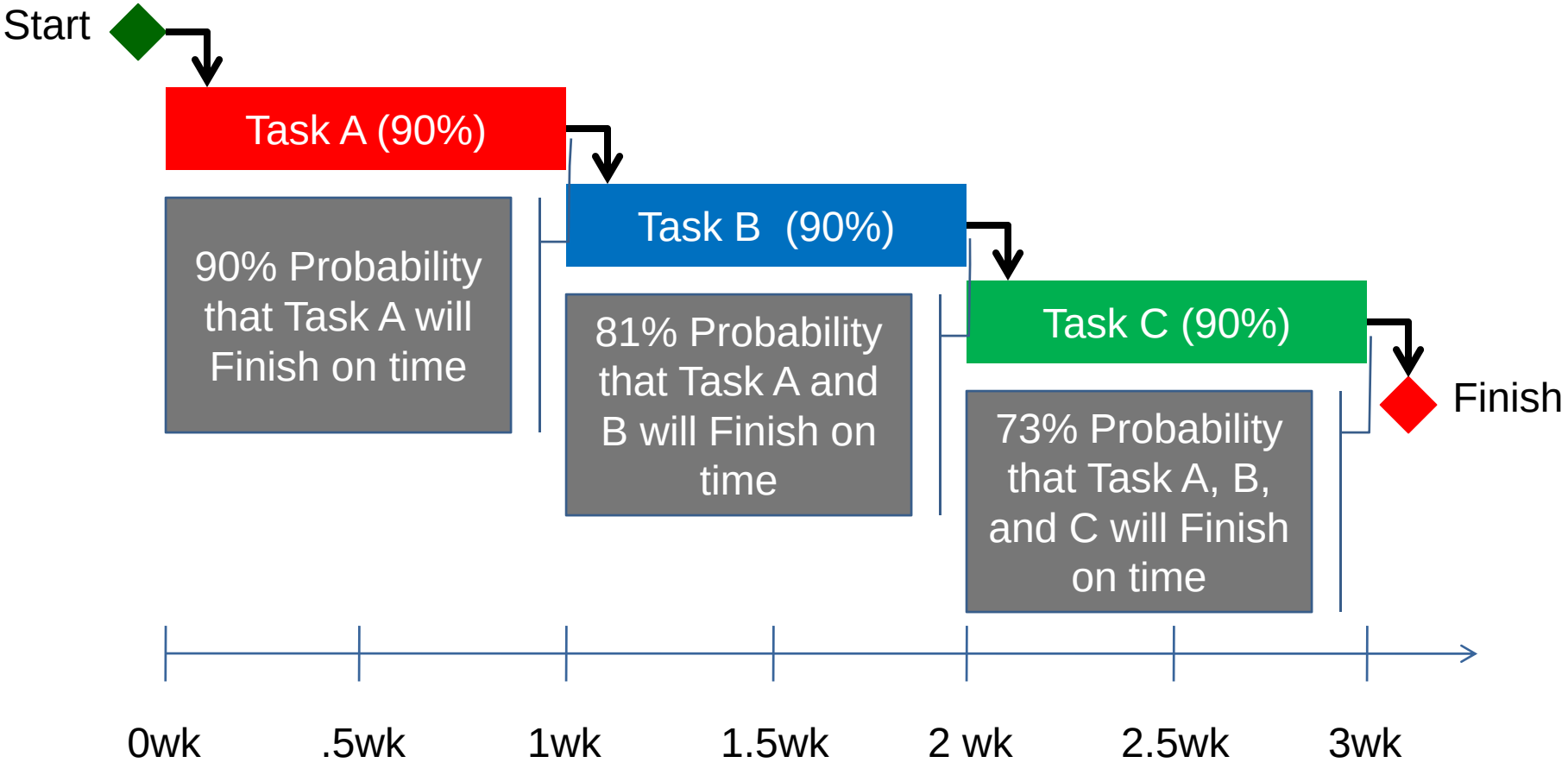
Shouldn't tasks average out (some early, some late)?

- Unfortunately, resources that finish early typically don't pass the work on because:
  - The receiving resource is not yet available
  - Fear that the early duration may now become expected on future projects by management
  - People like to keep "shining the apple" if they have extra time
- So what happens is that only "lateness" gets passed on and early completion does not.

http://www.pm411.org
ron@pm411.org

Probability of Dependent Events

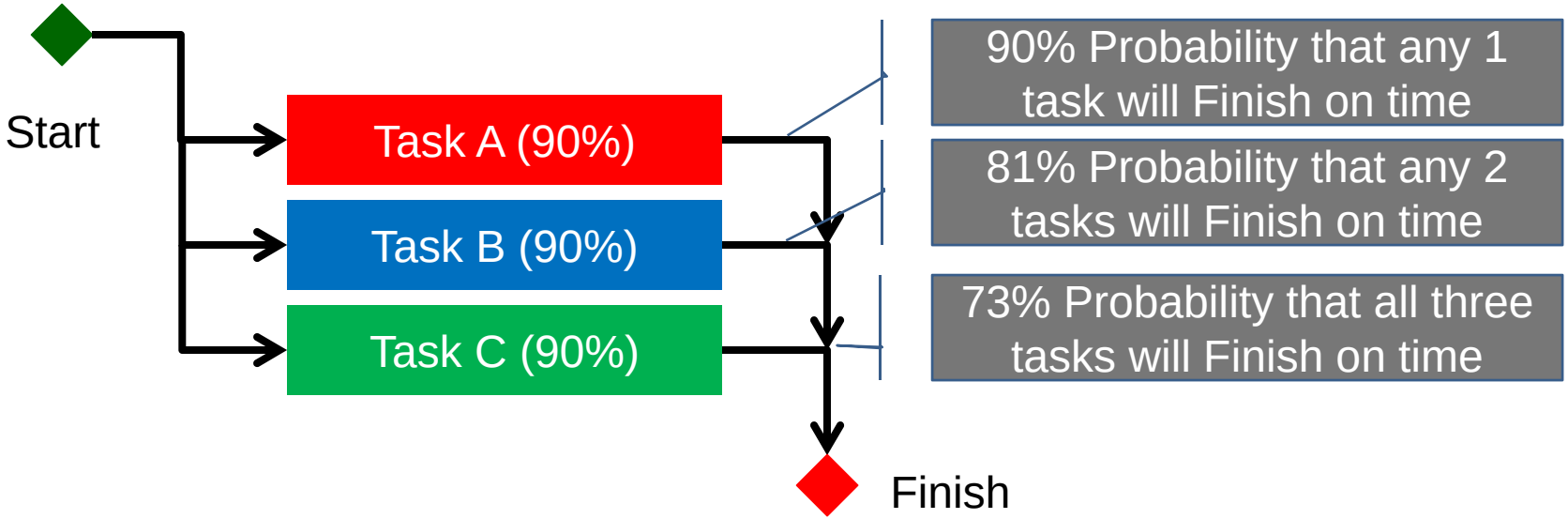
- Serial Events

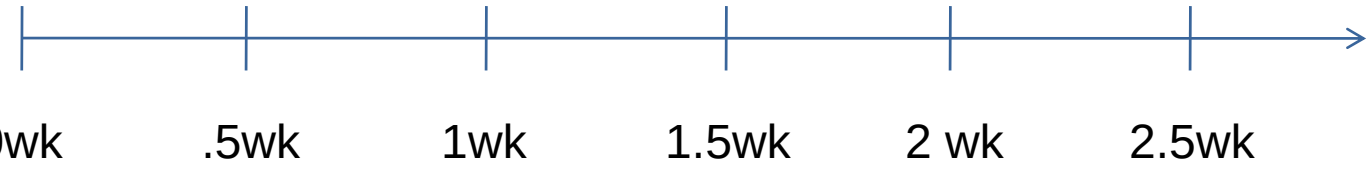


http://www.pm411.org
ron@pm411.org

Probability of Dependent Events

- Parallel Events





http://www.pm411.org
ron@pm411.org

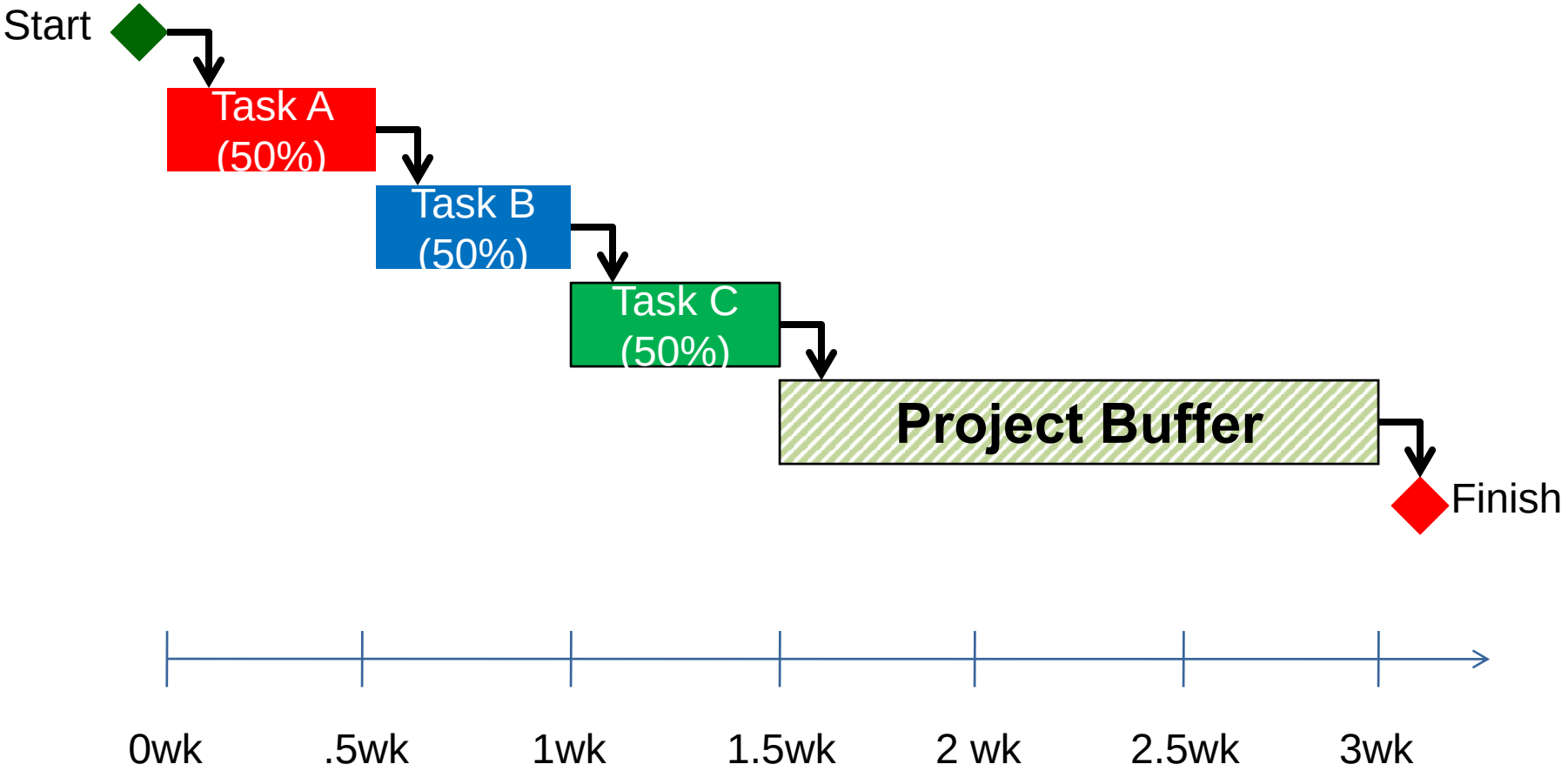
Projects are usually late

- As a result most organizations find that their projects are rarely on time, or they go through extraordinary measures ($ / scope slaughter) to meet their schedule commitments
- In addition to missed budget and scope requirements, additional extraordinary measures also often include overtime, worn out staff, poor morale, high turnover, etc.

http://www.pm411.org
ron@pm411.org

So, what should we do?

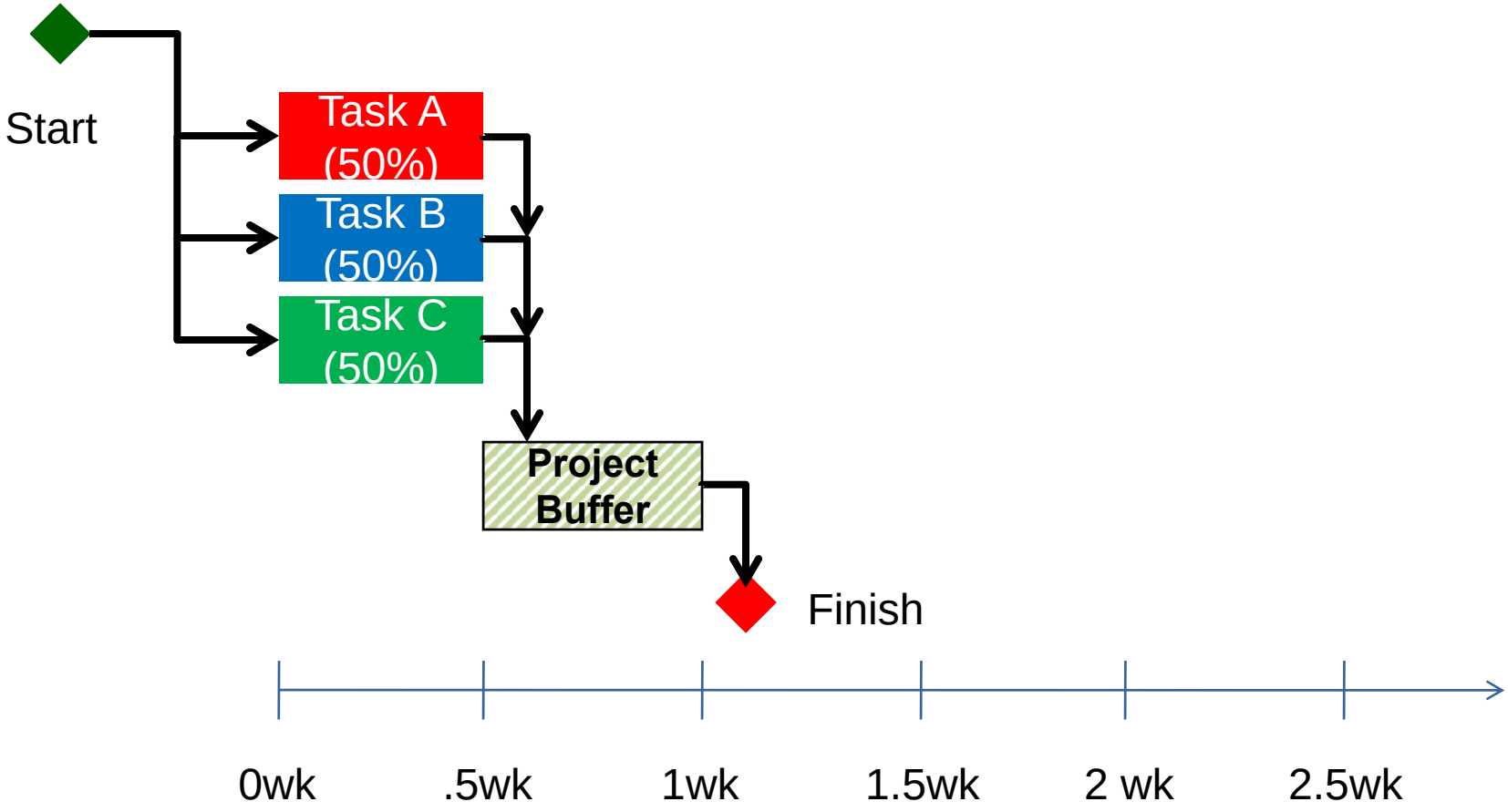
- Serial Events



http://www.pm411.org
ron@pm411.org

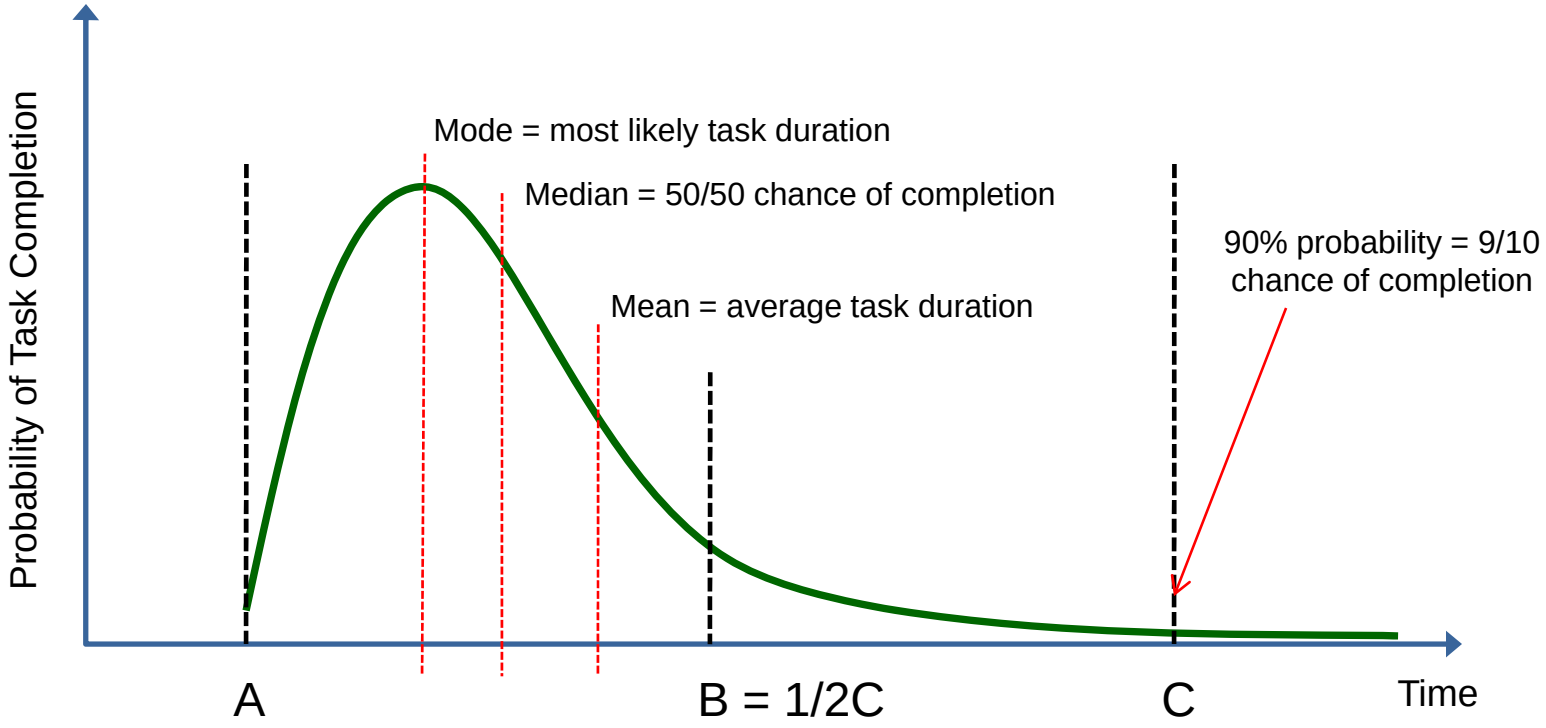
So, what should we do?

- Parallel Events



http://www.pm411.org
ron@pm411.org

Task Statistics

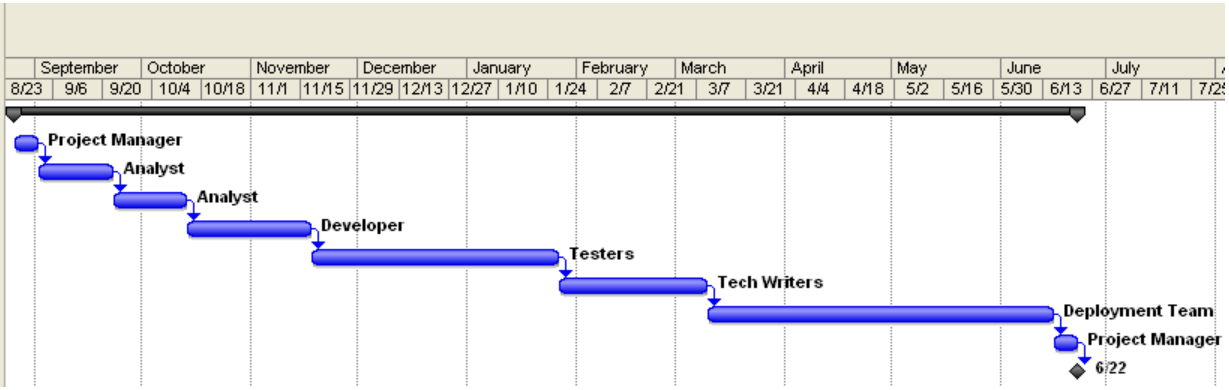


http://www.pm411.org
ron@pm411.org

Project Example

- Step 1: Team members provide <u>conservative</u> task durations that they are 90% confident they can hit

Task Name	Duration	Conservative (90% Probability) Durations	Aggressive (50% Probability) Durations
Software Dev Project	215 days	215 days	
Scope Defined	5 days	5 days	
Analysis/Software Requirements	15 days	15 days	
Design	15 days	15 days	
Development	25 days	25 days	
Testing	50 days	50 days	
Documentation	30 days	30 days	
Pilot	70 days	70 days	
Close out	5 days	5 days	
Finish	0 days	0 days	

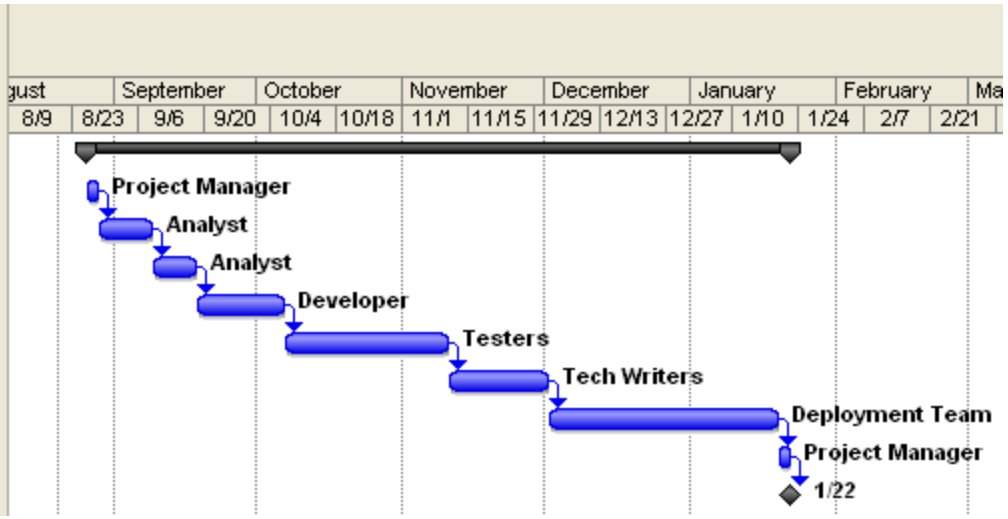


http://www.pm411.org
ron@pm411.org

Project Example

- Step 2: Team members provide <u>aggressive</u> task durations that they are 50% confident they can hit

Task Name	Duration	Conservative (90% Probability) Durations	Aggressive (50% Probability) Durations
Software Dev Project	**107.5 days**	**215 days**	**107.5 days**
Scope Defined	**2.5 days**	5 days	**2.5 days**
Analysis/Software Requirements	**7.5 days**	15 days	**7.5 days**
Design	**7.5 days**	15 days	**7.5 days**
Development	**12.5 days**	25 days	**12.5 days**
Testing	**25 days**	50 days	**25 days**
Documentation	**15 days**	30 days	**15 days**
Pilot	**35 days**	70 days	**35 days**
Close out	**2.5 days**	5 days	**2.5 days**
Finish	**0 days**	0 days	**0 days**

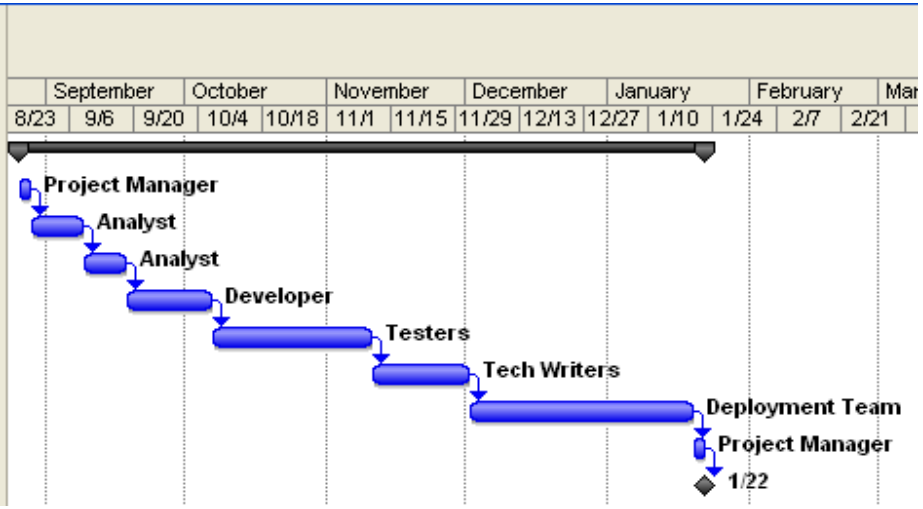


http://www.pm411.org
ron@pm411.org

Project Example

- Step 3: Calculate Project Buffer Contribution (PBC) of each task
- PBC = Conservative Dur – Aggressive Dur

Task Name	Duration	Conservative (90% Probability) Durations	Aggressive (50% Probability) Durations	Project Buffer Contribution (Difference)
Software Dev Project	**107.5 days**	**215 days**	**107.5 days**	**107.5 days**
Scope Defined	2.5 days	5 days	2.5 days	**2.5 days**
Analysis/Software Requirements	7.5 days	15 days	7.5 days	**7.5 days**
Design	7.5 days	15 days	7.5 days	**7.5 days**
Development	12.5 days	25 days	12.5 days	**12.5 days**
Testing	25 days	50 days	25 days	**25 days**
Documentation	15 days	30 days	15 days	**15 days**
Pilot	35 days	70 days	35 days	**35 days**
Close out	2.5 days	5 days	2.5 days	**2.5 days**
Finish	0 days	0 days	0 days	**0 days**

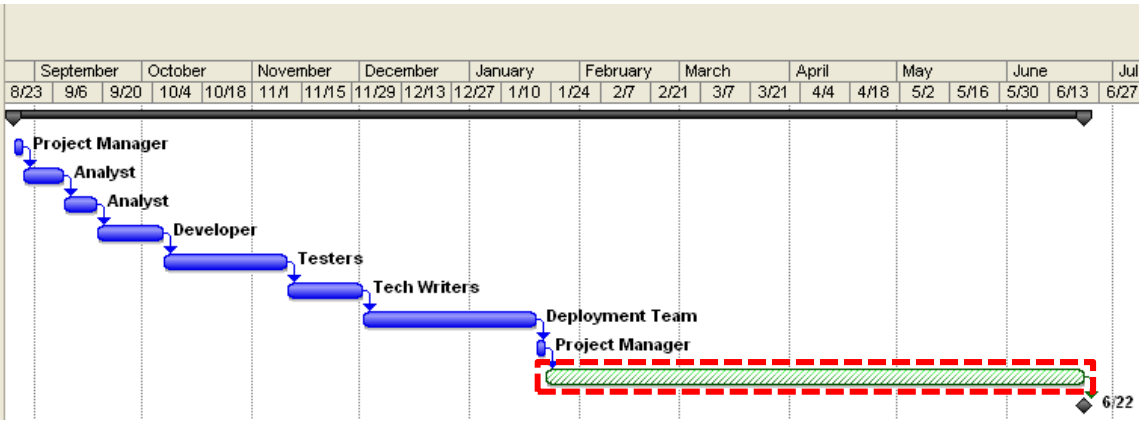


http://www.pm411.org
ron@pm411.org

Project Example

- Step 4: Add Project Buffer to end of schedule

Task Name	Duration	Conservative (90% Probability) Durations	Aggressive (50% Probability) Durations	Project Buffer Contribution (Difference)
Software Dev Project	**215 days**	**215 days**	**107.5 days**	**107.5 days**
Scope Defined	2.5 days	5 days	2.5 days	**2.5 days**
Analysis/Software Requirements	7.5 days	15 days	7.5 days	**7.5 days**
Design	7.5 days	15 days	7.5 days	**7.5 days**
Development	12.5 days	25 days	12.5 days	**12.5 days**
Testing	25 days	50 days	25 days	**25 days**
Documentation	15 days	30 days	15 days	**15 days**
Pilot	35 days	70 days	35 days	**35 days**
Close out	2.5 days	5 days	2.5 days	**2.5 days**
Project Buffer	**107.5 days**			
Finish	0 days	0 days	0 days	**0 days**

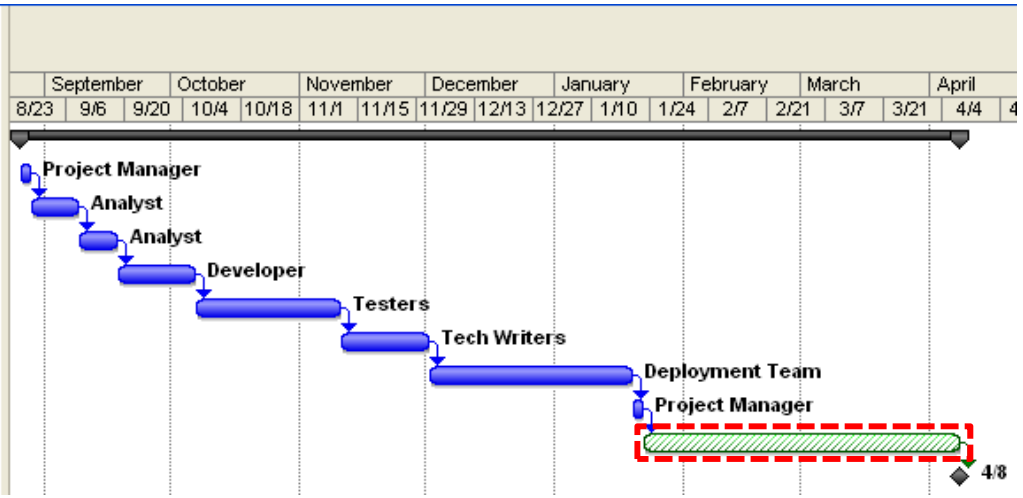


http://www.pm411.org
ron@pm411.org

Project Example

- Step 5: Divide Buffer in half (since 50% chance that task will not use up buffer)

Task Name	Duration	Conservative (90% Probability) Durations	Aggressive (50% Probability) Durations	Project Buffer Contribution (Difference)
Software Dev Project	**161.5 days**	**215 days**	**107.5 days**	**107.5 days**
Scope Defined	2.5 days	5 days	2.5 days	**2.5 days**
Analysis/Software Requirements	7.5 days	15 days	7.5 days	**7.5 days**
Design	7.5 days	15 days	7.5 days	**7.5 days**
Development	12.5 days	25 days	12.5 days	**12.5 days**
Testing	25 days	50 days	25 days	**25 days**
Documentation	15 days	30 days	15 days	**15 days**
Pilot	35 days	70 days	35 days	**35 days**
Close out	2.5 days	5 days	2.5 days	**2.5 days**
Optimized Project Buffer	**54 days**			
Finish	0 days	0 days	0 days	**0 days**

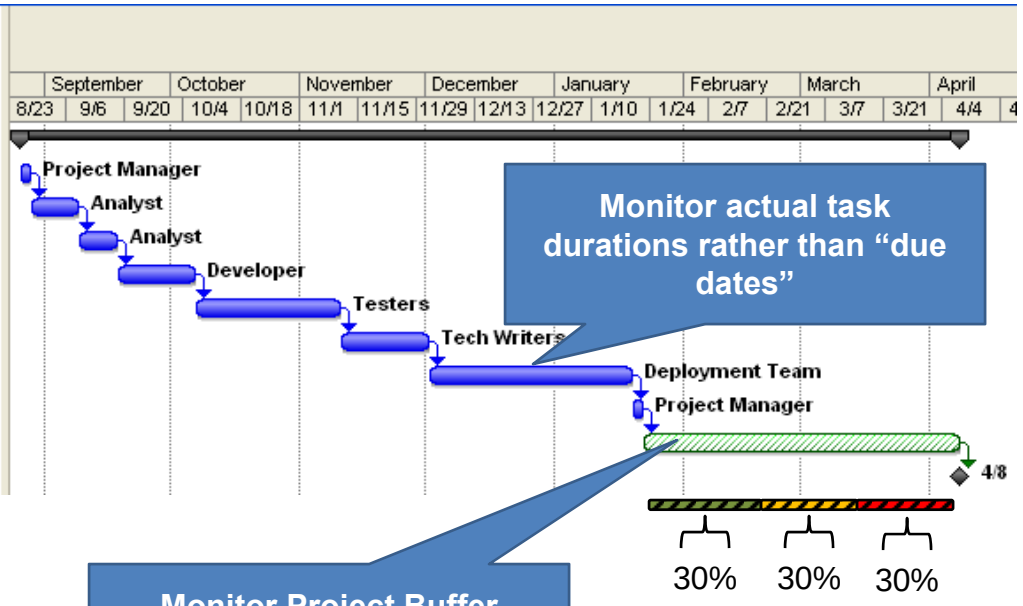


http://www.pm411.org
ron@pm411.org

Project Example

- Step 5: Divide Buffer in half (since 50% chance that task will not use up buffer)

Task Name	Duration	Conservative (90% Probability) Durations	Aggressive (50% Probability) Durations	Project Buffer Contribution (Difference)
Software Dev Project	**161.5 days**	**215 days**	**107.5 days**	**107.5 days**
Scope Defined	2.5 days	5 days	2.5 days	**2.5 days**
Analysis/Software Requirements	7.5 days	15 days	7.5 days	**7.5 days**
Design	7.5 days	15 days	7.5 days	**7.5 days**
Development	12.5 days	25 days	12.5 days	**12.5 days**
Testing	25 days	50 days	25 days	**25 days**
Documentation	15 days	30 days	15 days	**15 days**
Pilot	35 days	70 days	35 days	**35 days**
Close out	2.5 days	5 days	2.5 days	**2.5 days**
Optimized Project Buffer	**54 days**			
Finish	0 days	0 days	0 days	**0 days**

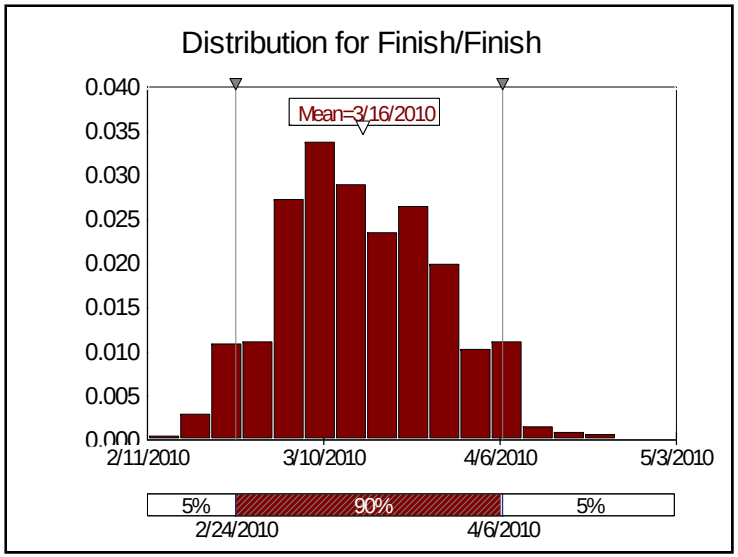


http://www.pm411.org
ron@pm411.org

Project Example

- Let's see if this really works by simulating this with risk!

Summary Statistics			
Statistic	**Value**	**%tile**	**Value**
Minimum	2/11/10	5%	2/24/10
Maximum	4/28/10	10%	3/1/10
Mean	3/16/10	15%	3/3/10
Std Dev	12.17500533	20%	3/4/10
Variance	148.2307549	25%	3/5/10
Skewness	0.204069957	30%	3/9/10
Kurtosis	2.676354044	35%	3/10/10
Median	3/15/10	40%	3/11/10
Mode	3/3/10	45%	3/12/10
Left X	2/24/10	50%	3/15/10
Left P	5%	55%	3/16/10
Right X	4/6/10	60%	3/18/10
Right P	95%	65%	3/19/10
Diff X	40.84375	70%	3/22/10
Diff P	90%	75%	3/24/10
#Errors	0	80%	3/26/10
Filter Min		85%	3/30/10
Filter Max		90%	3/31/10
#Filtered	0	95%	4/6/10



http://www.pm411.org
ron@pm411.org

Bottom Line: Finish each task as soon as possible!

- Be aware of those 3 behaviors that will end up affecting the whole team
  - Student Syndrome
  - Parkinson's Law
  - Bad Multi-tasking

- Take the safety in tasks and move it to the end of the project
  - Use the 50% Probablility Duration for tasks
  - Use ½ of the sum of the difference between the 90% probability duration and the 50% probability duration for the project buffer

http://www.pm411.org
ron@pm411.org

Questions?

http://www.pm411.org
ron@pm411.org