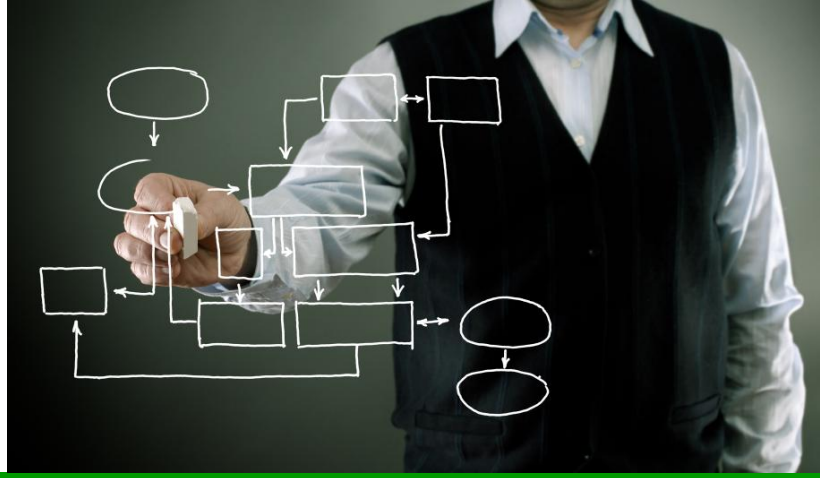
Critical Chain Project Management

Presented by
Rocco Surace, CPA, Jonah

GAINES KRINER ELLIOTT LLP
CERTIFIED PUBLIC ACCOUNTANTS & BUSINESS ADVISORS

100 Corporate Parkway, Suite 200
Amherst, New York 14226
Tel. 716.250.6600
www.gkecpa.com

What is the problem?



WHAT TO CHANGE?

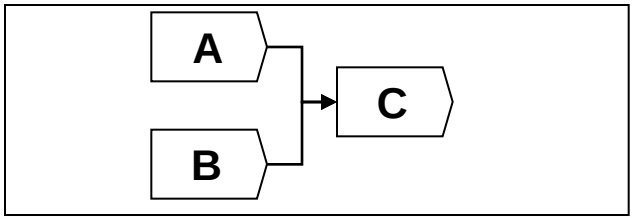
2

UNCERTAINTIES MAKE PROJECT EXECUTION CHALLENGING

- Requirements change midstream, but due-dates do not
- Work takes longer than expected
- More work is discovered in execution
- There is rework
- Resources are busy elsewhere (even when promised)
- Resources are pulled off for non-project work
- Resources are pulled into multiple directions at the same time
- Project manager may not have direct control over the resources
- Vendors do not deliver
- Technology fails, work takes longer than expected
- Necessary inputs not available (materials, specs, approvals etc.)
- New projects take resources away from current projects
- Projects are late, over budget and under scope
- …

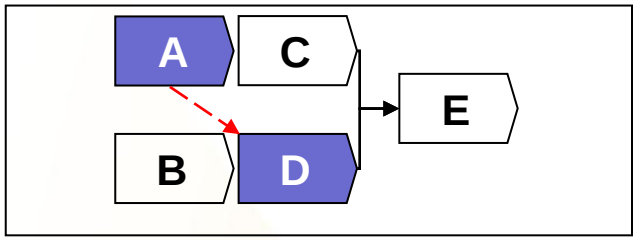
Uncertainty and Dependencies

Integration Dependencies



- Delays: C is delayed if either A or B is late
- Gains: Even if A or B finishes early, C cannot be started

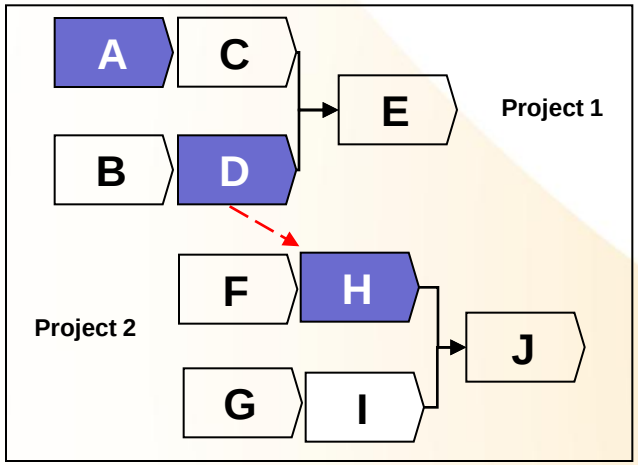
Resource Dependencies Within Project



A & D are done by the same resource

- Delays: If A is late, not only C but also D gets delayed
- Gains: Even if A finishes early, resource cannot start D as has to wait for B to finish

Resource Dependencies Across Projects



D & H are done by the same resource

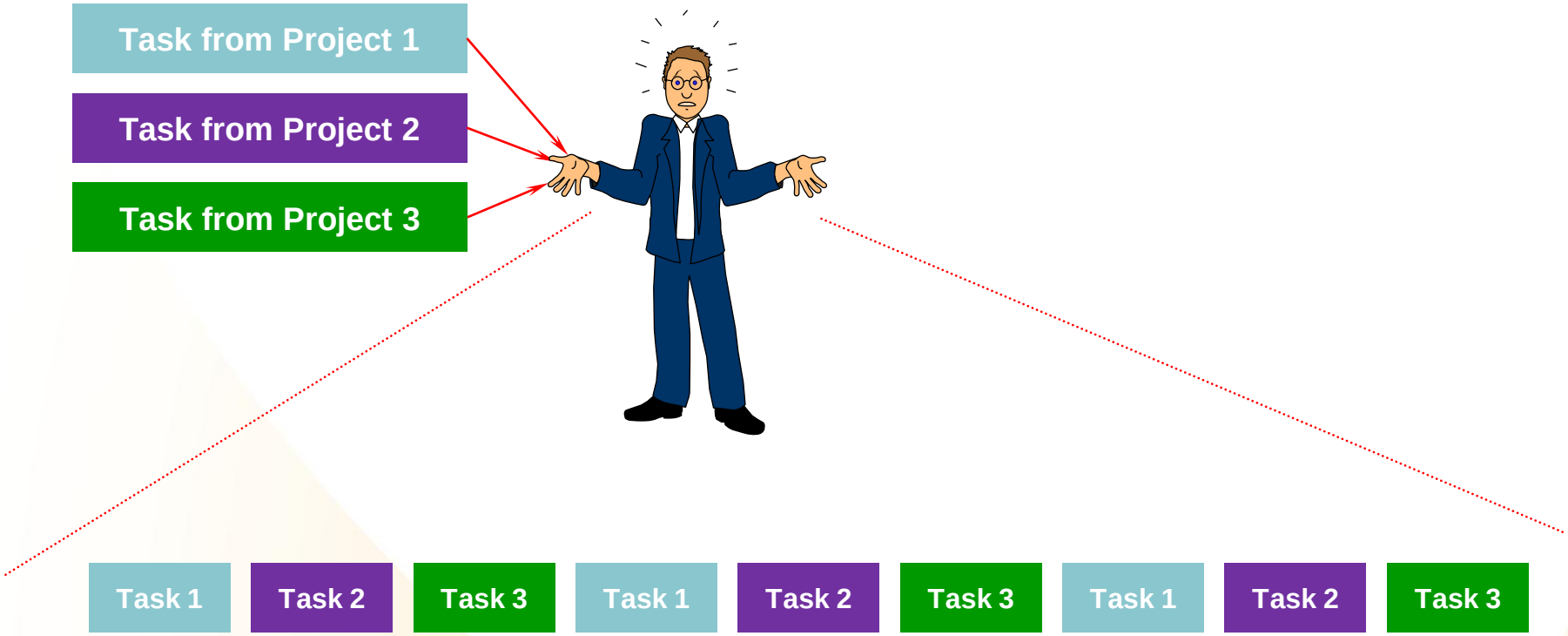
- Delays: If D on Project 1 is late, H on Project 2 also gets delayed as resource is stuck on D
- Gains: Even if D finishes early, resource cannot start H as has to wait for F to finish

Delays multiply

Gains do not add up

4

As delays mount, project managers fight for the shared resources and induce multitasking



A Common Management Paradigm

In order to ensure on-time completion of the whole project, we have to ensure that each task is completed on time

- Do you agree to the above statement?
- Do you think that the above statement is quite common among project managers?
- If this statement is false, how should you re-verbalized it?

Some Direct Ramifications

- The task manager is required to commit to the completion of the task at some specific time
  - Hence, the task manager assumes the intention is for “high confidence” timing, significantly longer than average time
- The task manager realizes that if the current task would complete long before his own estimation time, then next time he’d be required to cut his estimation
  - Hence, there is a tendency to refrain from early completion of tasks
  - How many tasks finish long before the planned time?

The direct impact of the common assumption

Project team members naturally tend to inflate time estimates for accountable tasks (add “safety” time)

Project team members wish to be well appreciated

Management implements formal or informal on-time measurements

Common Managerial Assumption

Management assumes that each individual task must be completed on time for the whole project to finish on time

A less direct impact, but quite devastating

Project team members try to fill all the time planned for each task (Parkinson Law is active)

Project team members recognize the need for safety time

When tasks finish ahead of time, management tries to cut the safety time of future estimates

Common Managerial Assumption

Management assumes that each individual task must be completed on time for the whole project to finish on time

Management strives to shorten the lead time of projects

Unfortunately…

Hidden Safeties Only get Wasted in Execution

Error Reporting

People do not report early completions.

Student Syndrome

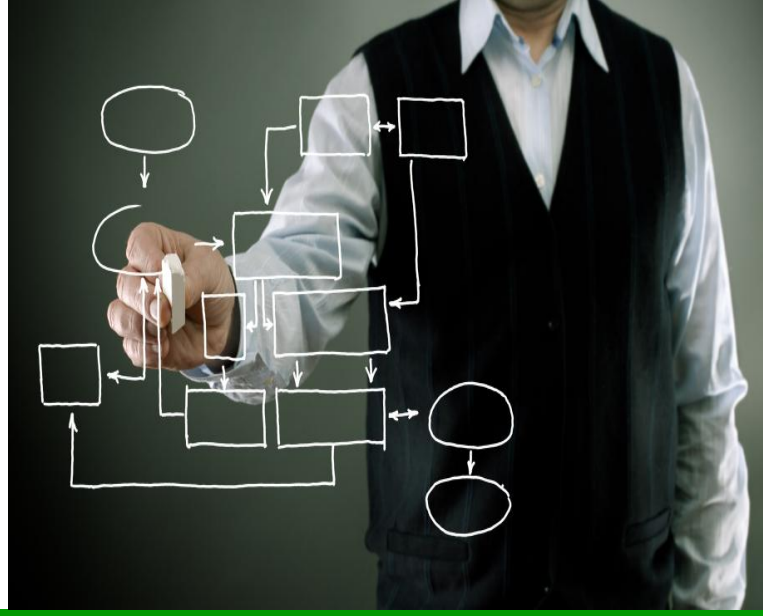
Start the task late, even if it arrives on time.

Parkinson's Law

Work expands to fill time available.

10

What is the solution?



WHAT TO CHANGE TO?

Proper Planning

Then

Proper Execution

11

CRITICAL CHAIN RULES

- ✓ **Control release of projects into execution**
- ✓ **Don’t create precise schedules in planning**
- ✓ **Plan aggressive cycle times with buffer time**

Planning

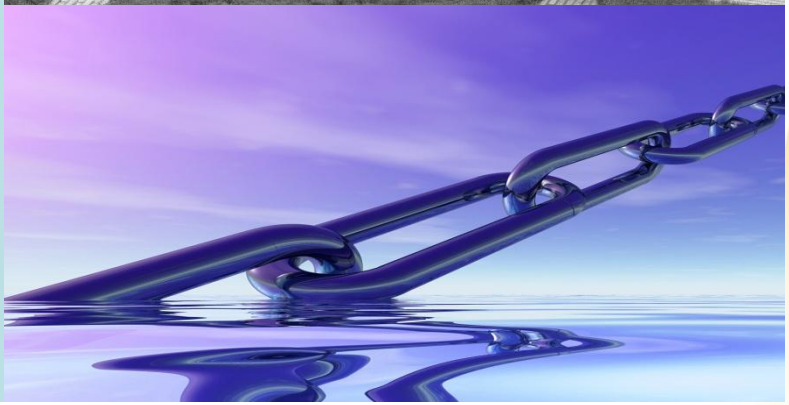
- The necessary characteristics of the proper planning
  - Provide realistic completion dates for a portfolio of projects
    - The due-dates for every project should be adequately protected
  - The planning should take into consideration capacity contention
    - Without wasting too much capacity due to the uncertain environment – allowing some controllable level of contention between projects
  - The planning should focus on the sensitive areas for achieving the objectives
    - And leave enough flexibility for the execution side

Execution

- The necessary characteristics of the execution
  - Being able to get a clear status of the project
  - Pointing to the exact areas where extra effort or support are truly needed in order to get the project on time
  - Setting a clear priority scheme for the assignment of resources between competing projects
  - Accumulating the necessary data for future analysis of the effectiveness of the protection mechanisms and the quality of the task time assessments

The Solution's Main Concepts

- Planning single projects
  - Protecting the project as a whole rather than protecting each task
  - Within every project, resource contention would be considered
  - This necessitates changing the 'critical path' definition to critical chain



The Solution's Main Concepts

- Execution control - Monitoring the buffers
  - The buffers are the protection mechanisms, monitoring them serves to evaluate the current state of the project
    - Consider where special efforts should be directed

The Solution's Main Concepts

Planning the whole project portfolio

– Locating one single criterion, usually the capacity of a relatively loaded resource, as the key for **staggering projects**

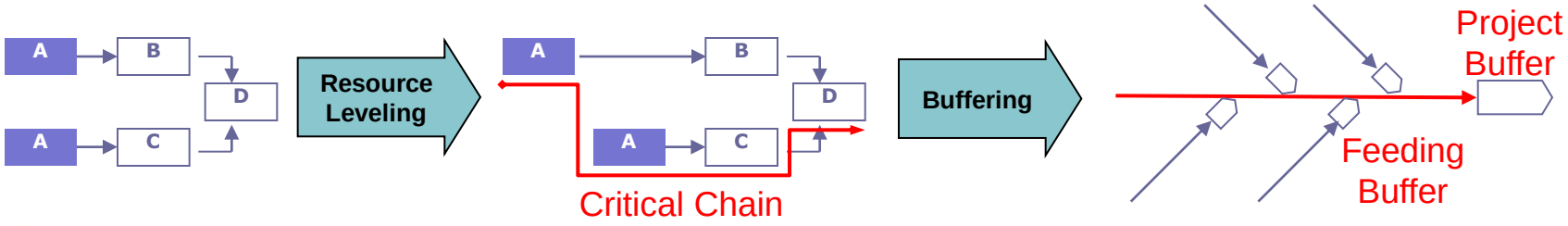
- so multi-tasking would be minimal and thus every project can proceed as fast as possible without interferences from other projects

Controlling the execution

– Basing the inter-project priorities on the state of the buffers

- That means assigning resources according to global and objective measures that predict the chances of meeting the planned completion times

Critical Chain Buffering



Critical Chain (CC)

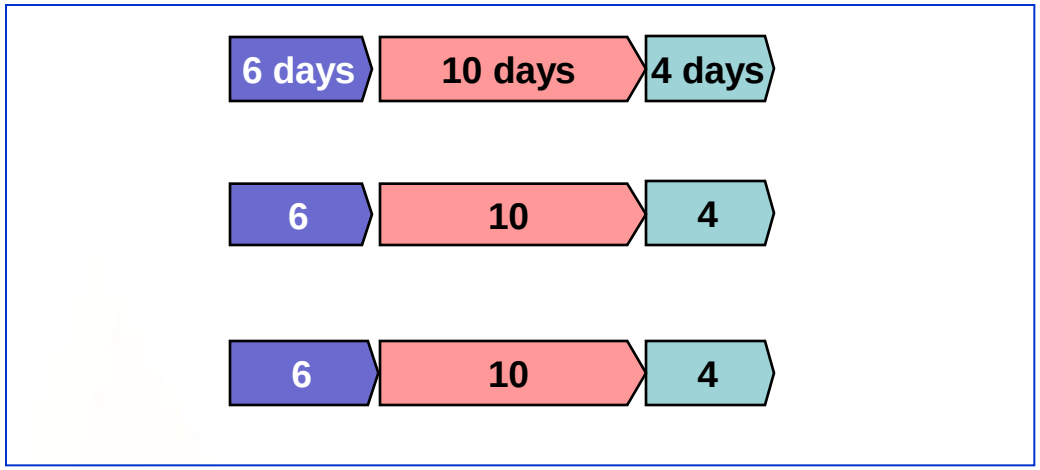
Longest sequence of activities after resolving resource contention within a project

Buffers

Blocks of unscheduled time placed at the end of a series of tasks to absorb cumulative delays.

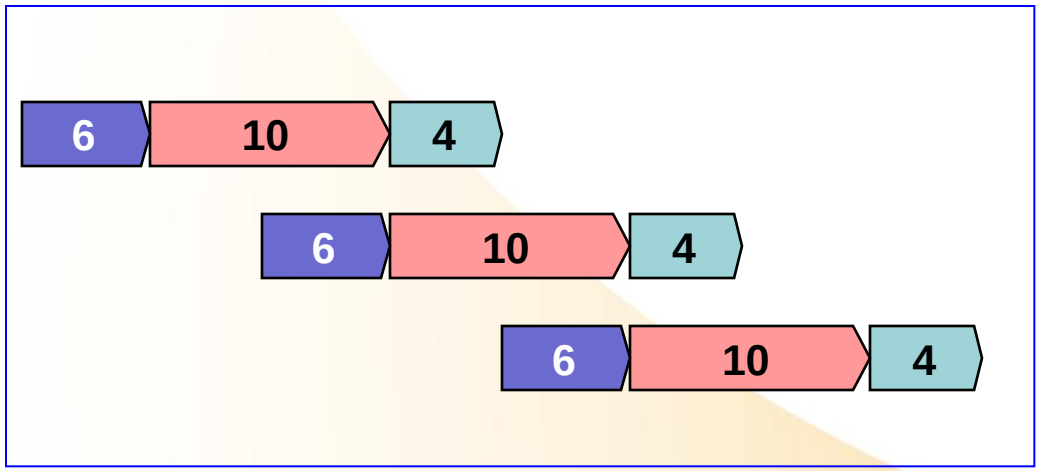
- **Feeding buffers at end of non-CC, which are “free”.**
- **Project buffer at end of CC protects due-date. Gains along CC pass on to project end.**
- **Cumulative buffers provide most protection without sand-bagging.**

WHY CONTROL THE RELEASE OF PROJECTS INTO EXECUTION



All three projects started ASAP

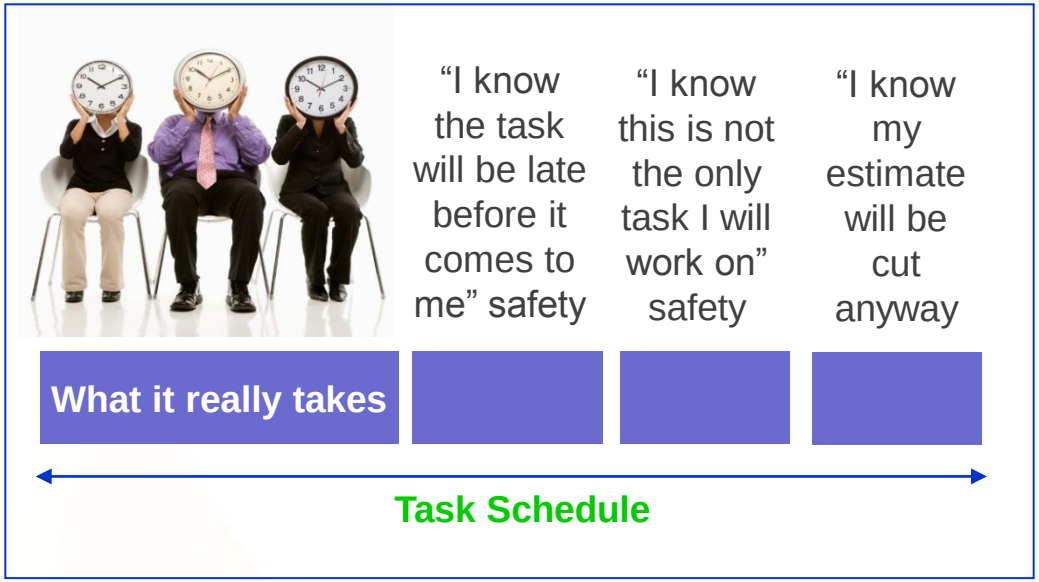
- ❑ Constant resource shuffling
- ❑ Priorities not synchronized
- ❑ All projects are delayed



Start dates are staggered

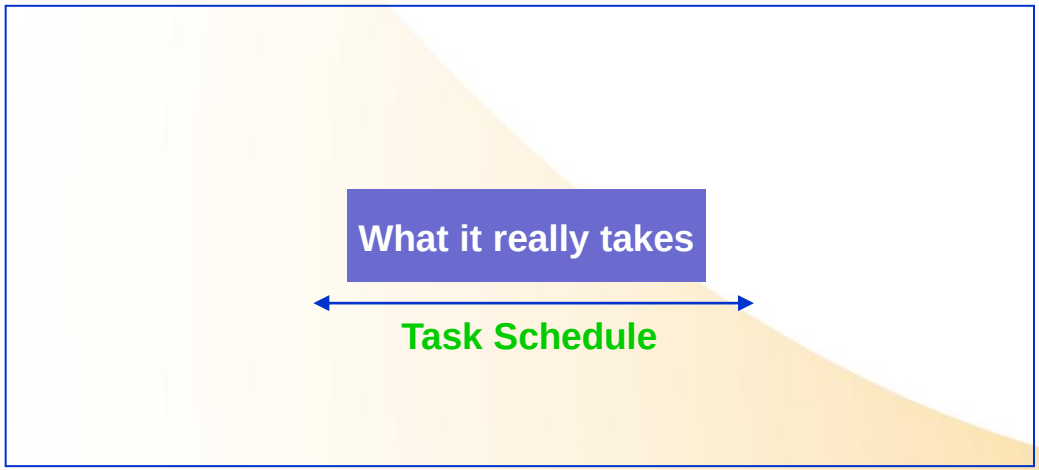
- ❑ Resources stay focused
- ❑ Tighter synchronization
- ❑ All projects finish faster
- ❑ More projects can be done

WHY NOT CREATE PRECISE SCHEDULES AT PLANNING TIME



Schedules set in planning

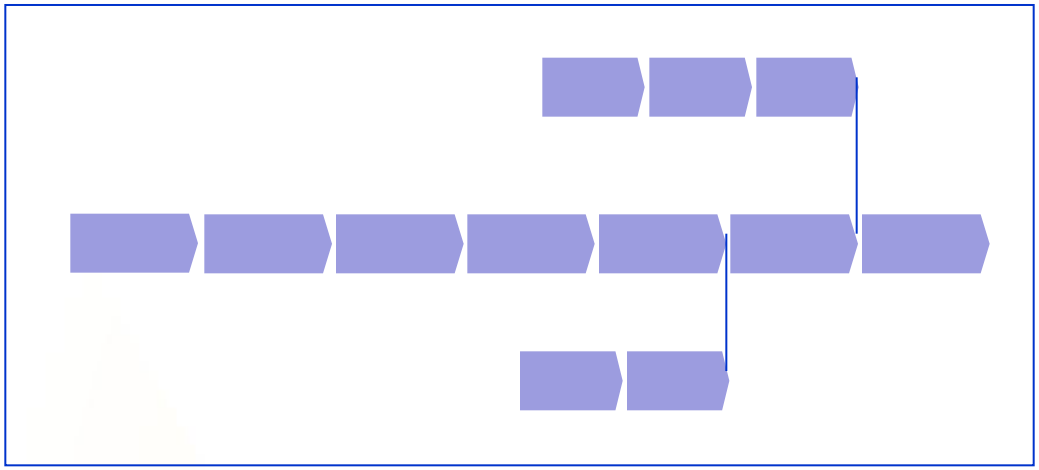
- ❑ **Uncertainties are unavoidable**
- ❑ **Wanting to be reliable, people add safeties into estimates**
- ❑ **Safeties get wasted in execution (Parkinson's Law)**



Schedules set in execution

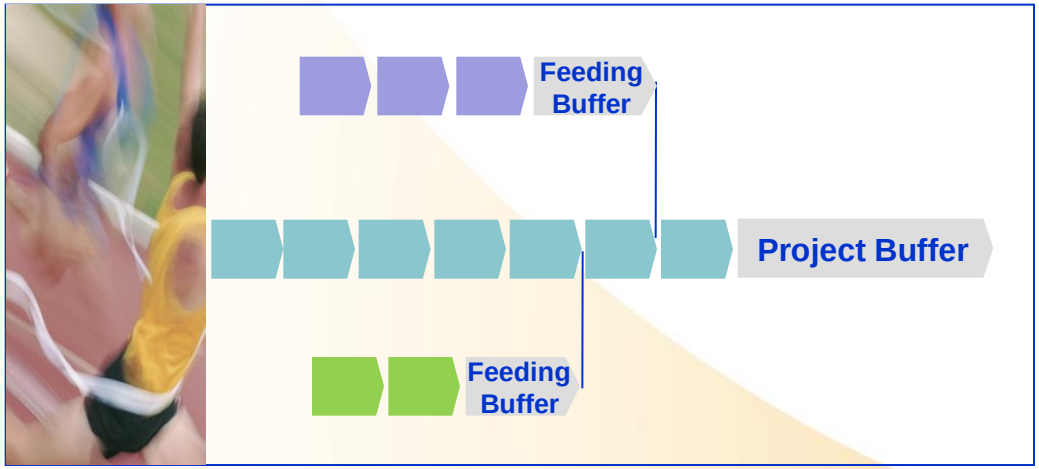
- ❑ **Estimates are used only to plan projects**
- ❑ **Task schedules determined in execution, when tasks are closer to being started**

WHY EXPLICIT BUFFER TIME MAKES PROJECTS RUN FASTER



No buffer times in projects

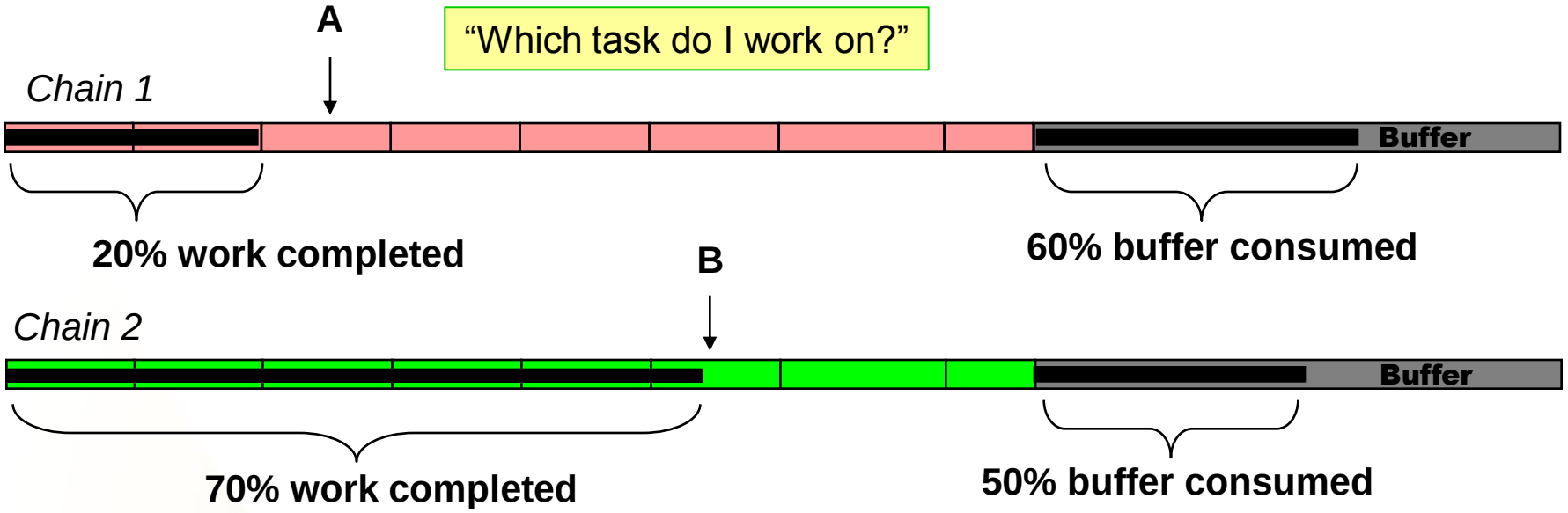
- Delay on feeding paths slow down the longest path
- People might hide safeties if they are accountable for delivering on time anyway



Buffers are made mandatory

- "Feeding Buffers" protect longest path from incoming delays
- "Project Buffer" protects project due-date
- Individual task durations can be made more aggressive

Buffer-driven Task Priorities and Measurement

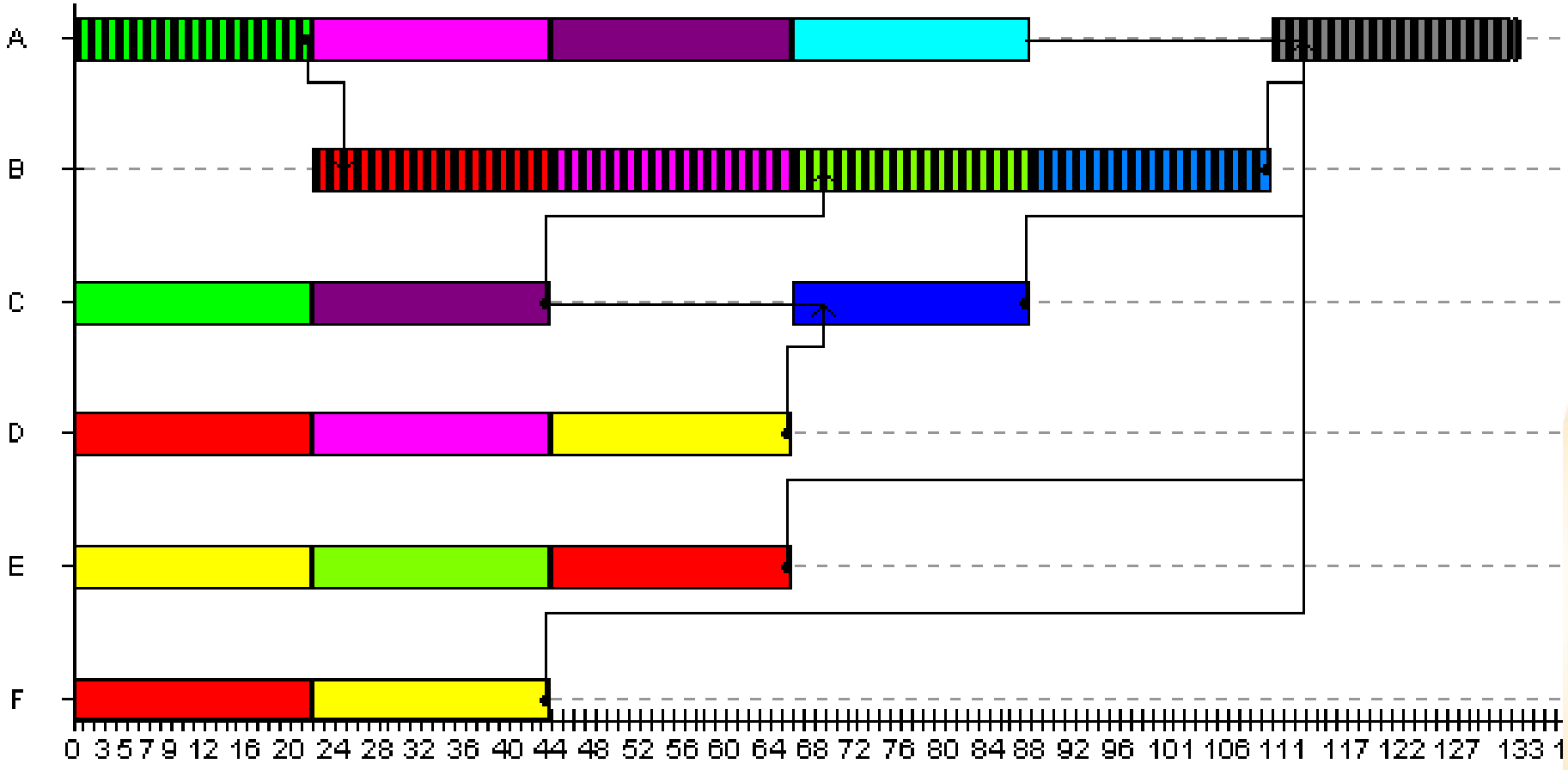


Safety Index: % of buffer consumed vs. % of work completed. Automatically calculated on an ongoing basis to assess how much buffer is still available for future uncertainties.

Tasks that lie on chains with low safety index are given top priority. This ensures that buffers are not wasted, and also reduces pressure to multitask.

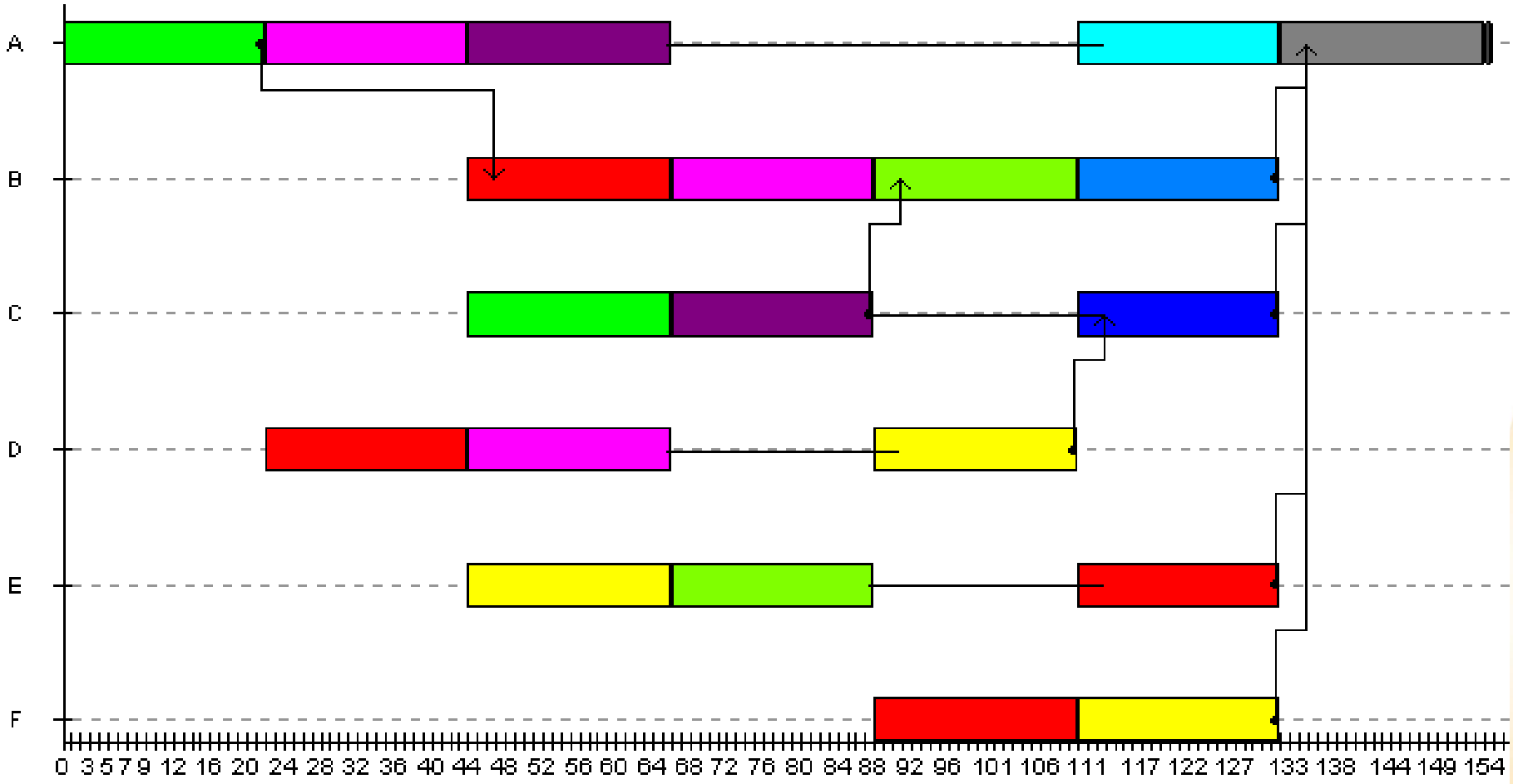
For Example: a typical project

Resource contention has not been resolved
The critical path contains six tasks (marked)



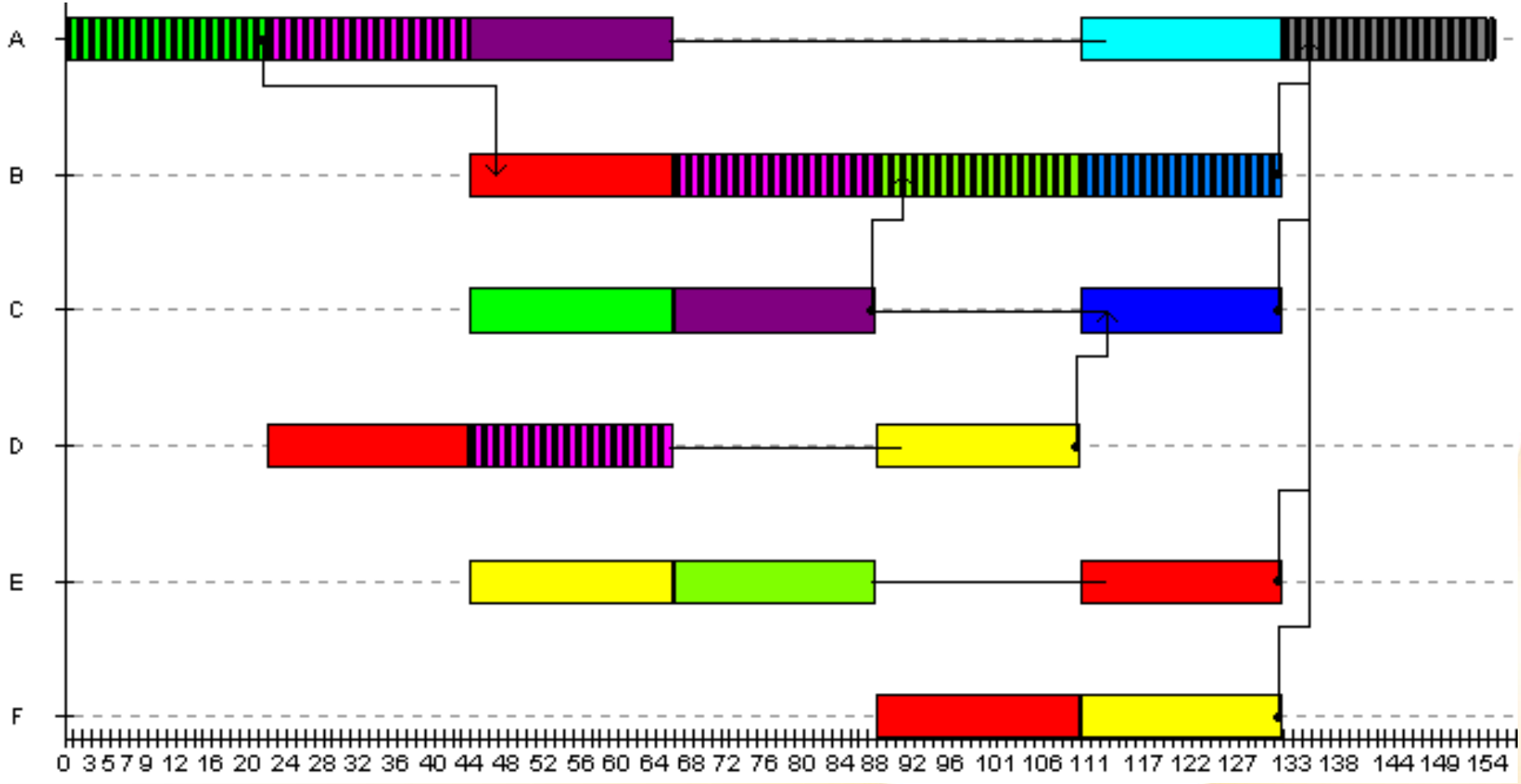
For Example: a typical project

Resource contention has been resolved



The Critical Chain Within the Project

The critical chain contains seven tasks

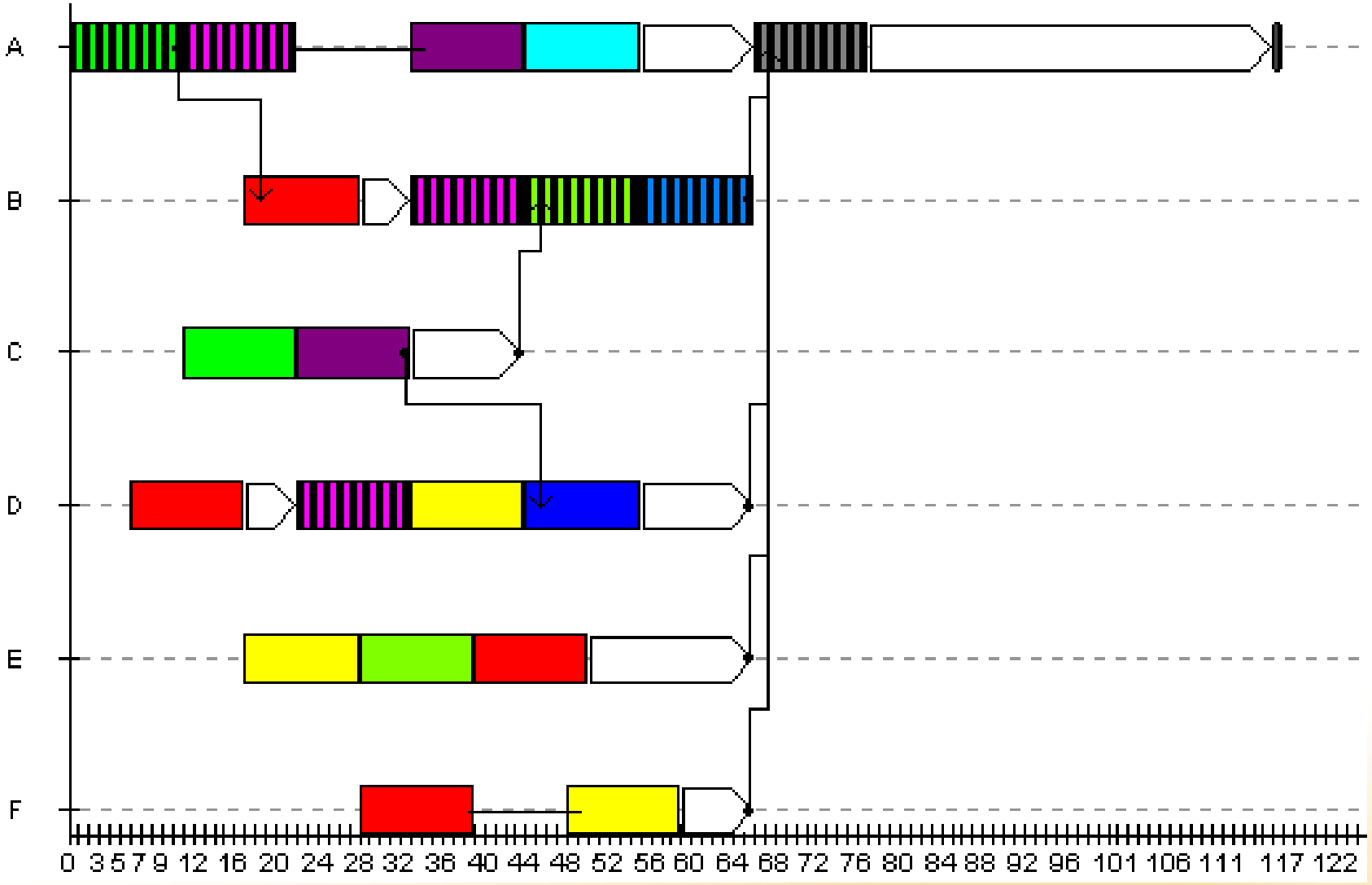


The TOC planning for that project

Every task is cut by 50% to get rid of the individual safety

Feeding buffer

Project buffer



Buffer Management
(Monitoring the buffers)

- In the execution phase, there is a need to evaluate the current impact of uncertainty on the state of the project
  - The state of the project buffer tells us the accumulative impact of the uncertainty so far
  - The state of the project buffer relative to the part of the critical chain completed also tells us the chances to finish on time, relative to the state at the beginning of the project
    - Are we now in better or worse shape than at the beginning?

Using Buffers to set Priorities in Execution

Green task in Project 1 has the higher priority because it has a lower Buffer Index

Project 1

Buffer Consumption = 60%

Project Completion = 50%

Buffer Index = 0.5/0.6 = 0.83

Project 2

Buffer Consumption = 20%

Project Completion = 33%

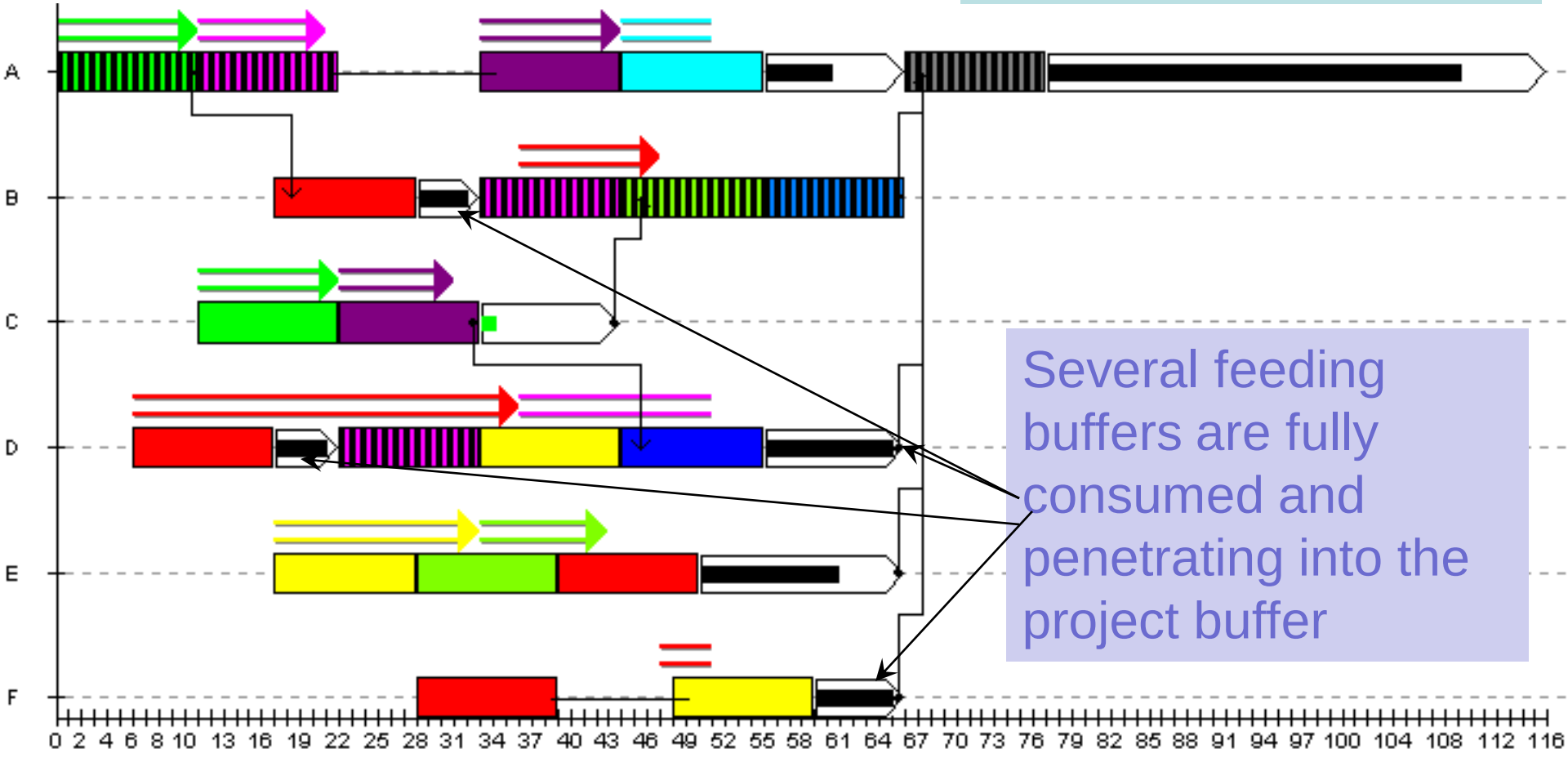
Buffer Index = 0.33/.2 = 1.65

Buffer Management

- What about the feeding buffers?
  - We can define the state of every feeding buffer in the same way as the project buffer
  - But, the damage from a fully consumed feeding buffer is much less than from a fully consumed project buffer
  - There are two benefits from monitoring the feeding buffers
    - One is that bad status of a particular feeding buffer means the non-critical chain might become critical and hence proper management efforts should be diverted to it
    - The second allows for learning from the state of many buffers, especially regarding the quality and bias of the initial assessments

The Status of a Project

Project buffer:
Length: 39 days
Consumption: 33



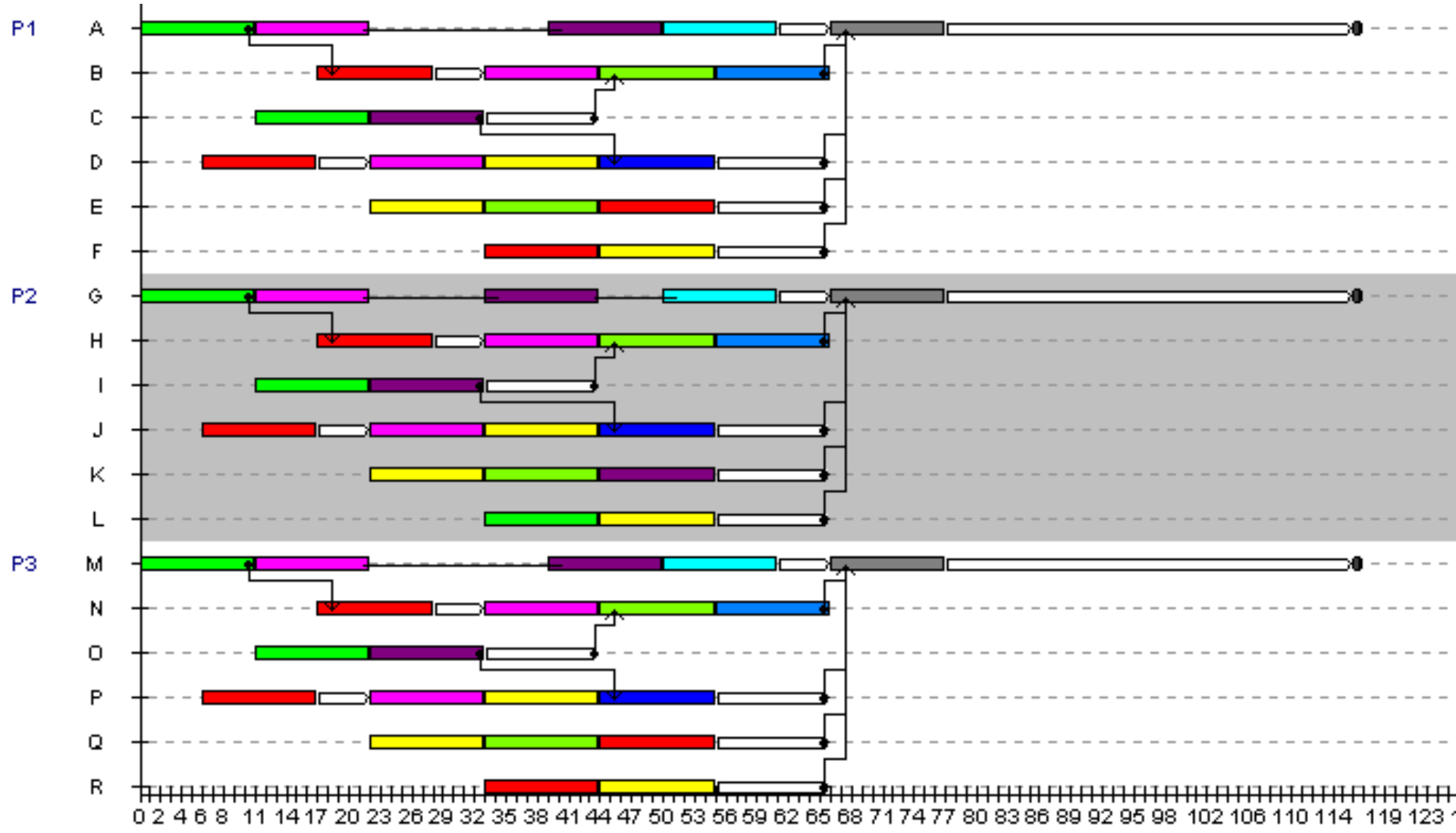
Several feeding buffers are fully consumed and penetrating into the project buffer

Locating One Single Criterion as the Key for Staggering Projects

- The chosen resource for staggering the projects is called the “drum resource”
  - The drum resource is a constraint because its capacity is constraining the number of projects we plan to do at a period of time
  - Usually the drum resource is NOT a bottleneck. It has to have excess capacity in order not to cause too long queue of projects
  - The “drum” in project management is not so tight as the “drum” in manufacturing
    - Because we need to supply every project, once started, as soon as possible

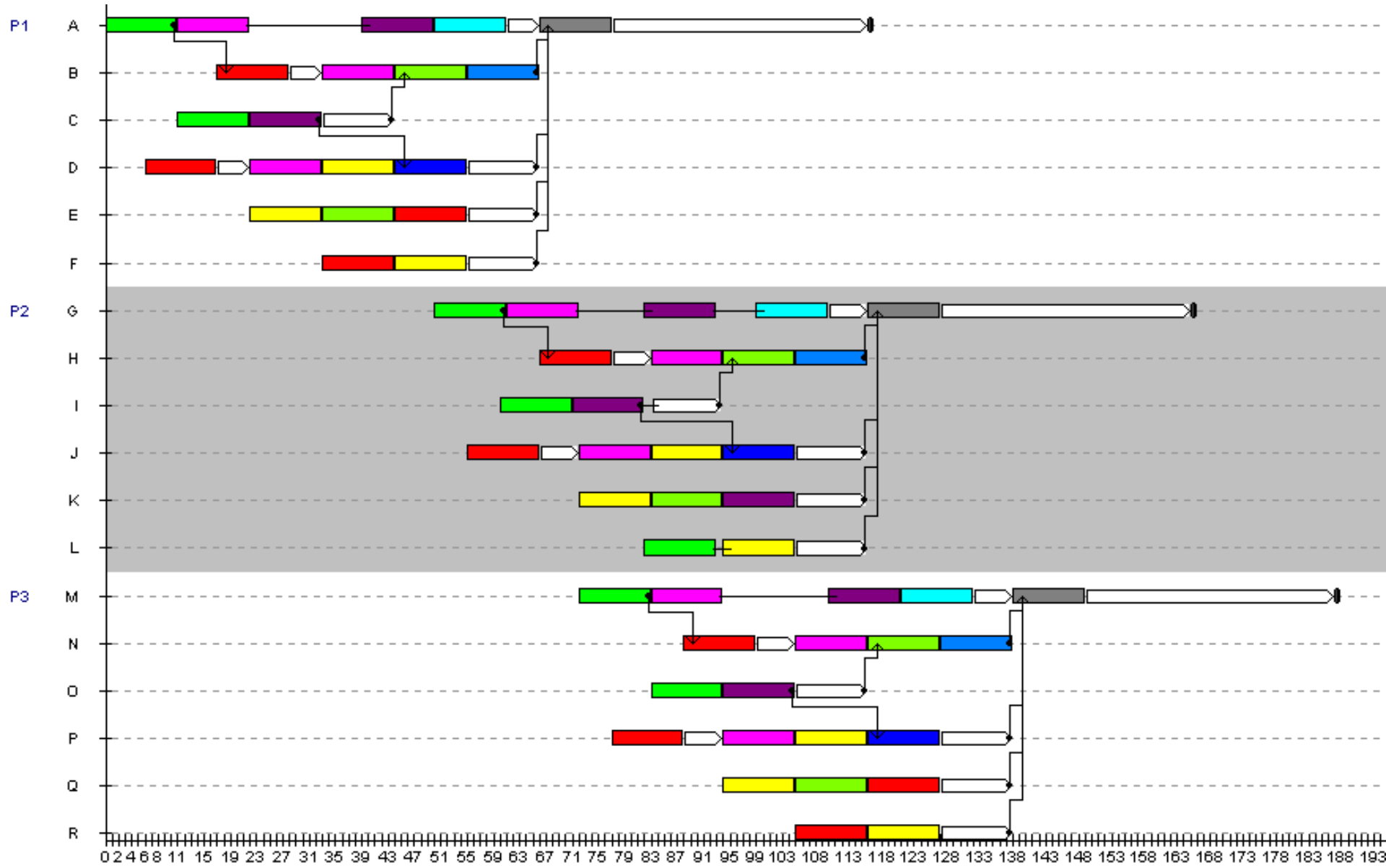
Three Projects are Competing for the Time of the Same Resources

How should we schedule the portfolio of three projects?



This is One Possibility

Scheduling the Start of Projects According to the "Red Resource"



Actual Results from Realization – www.realization.com

Company	Before	After
ABB Córdoba Power Transformers, Engineer-to-Order	Engineering cycle time was 8 months. On-time delivery was 85%.	Engineering cycle time reduced to 3 months. On-time delivery improved to 95%. 16% increase in manufacturing throughput (revenues).
Danisco (Genencor International) Biotechnology Plant Engineering	20% projects on time.	87% projects on time. 15% immediate increase in throughput.
Dr. Reddy's Laboratories Pharmaceutical New Product Development	6 projects completed in first 12 weeks. 20% projects on time in 12 weeks. 85 global generics and PSAI filings in 2009. 85 product launches in 2009. 915 days cycle time for full development in 2008.	11 projects completed (83% increase). 80% projects on time (60% increase). 110 filings in 2010 (30% increase). 149 launches in 2010 (75% increase). 563 days cycle time for full development in 2010 (40% faster).

Actual Results from Realization – www.realization.com

Company	Before	After
Hamilton Beach Brands, Inc. New Product Development Home Appliances	34 new products per year. 74% projects on time.	Increased throughput to 52 new products in 1st year, and to 70+ in 2nd year, with no increase in head count. 88% projects on time.
Oregon Freeze Dry Food Preparation & Packaging	72 sales projects completed per year.	171 sales projects completed per year. 52% increase in throughput-dollars.
US Air Force, Warner Robins Air Logistics Center **C5 Production Line** Aircraft Repair and Overhaul	Turnaround time 240 days. 13 aircraft in repair cycle.	Turnaround time 160 days. 7 aircraft in repair cycle. 75% fewer defects.

A Nice Success Story

Maintenance Center, Albany, GA

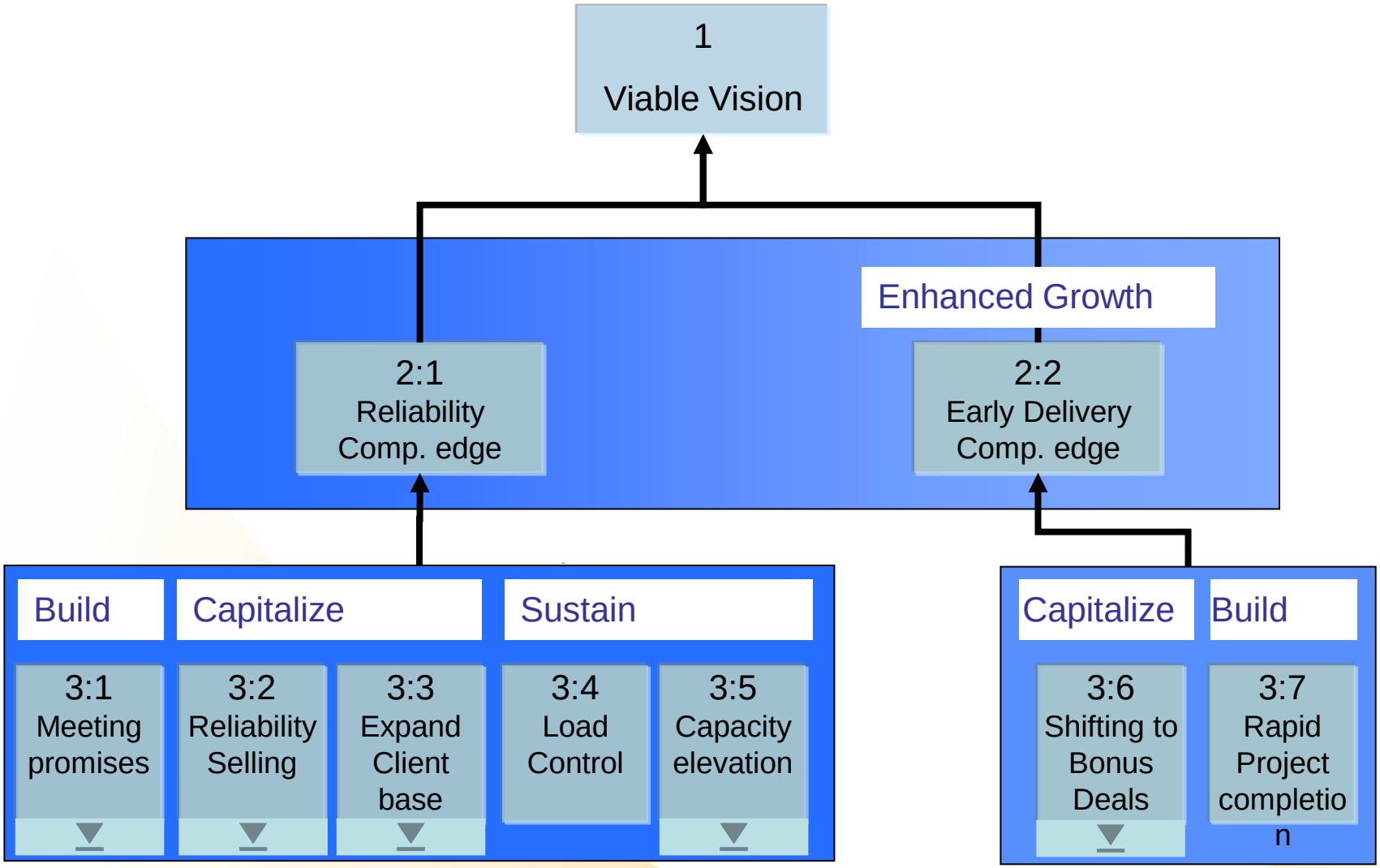
Matcom Video

GOLDRATT consulting

The Strategy & Tactic Tree

Projects

PROJECTS S&T



3.1.4	Load Control
Necessary assumption	When sales are growing fast the chances increase to miss completion due-dates or to offer completion dates which are too far into the future.
Strategy	The due dates the Company gives are (almost) always accepted and met, irrespective of the growth in sales.
Parallel assumptions	When sales are growing fast, the load on key resources increases. It is relatively easy to have high due-dates performance when the commitments are given based on the staggering mechanism of CCPM. Given enough warning it is feasible to train/add suitable resources.
Tactic	The staggering mechanism of CCPM is strictly obeyed even if it results in losing some bids in the short term.

4:11.1	Reducing Bad Multi-Tasking & WIP
Necessary assumptions	When too many projects are executed simultaneously many resources will find themselves under pressure to work on more than one task – bad multi-tasking is unavoidable. Prolific bad multi-tasking significantly prolongs each project's lead-time.
Strategy	Flow is the number one consideration (the target is not how many projects the Company succeeds to start working on, rather it is how many projects are completed).
Parallel assumptions	➢ The statement, "the earlier we start each project, the earlier each project will be finished," is not correct for multi-project environments (not only the first elephant but also the last elephant will go through a door much faster if they go in procession). ➢ Vast experience shows that in multi-project environments, reducing the number of open projects can reduce bad multi-tasking without causing starvation of work and therefore significantly reduces the lead time of all projects – it increases the flow. (cont.)

Reducing Bad Multi-Tasking & WIP (cont.)

Tactic	The Company properly controls the number of projects that are open at any given point in time.
Sufficiency assumption	Adjusting the amount of work is not enough. The company must also ensure that as time passes the proper amount of work will be always maintained.

4:11.2	Full-Kitting
Necessary assumptions	The current pressure often causes projects to be in execution without the needed preparations being completed (detailed specifications, authorizations, etc.).
Strategy	A project is rarely launched before its preparations are complete.
Parallel assumptions	➢ The resources dealing with preparations are caught in a never-ending catch-up cycle. ➢ Freezing of projects frees up, for a while, ample capacity of the resources dealing with preparations.
Tactic	The company uses the window of reduced load on resources that do the preparations to ensure that “full kit” practice will become the norm.
Sufficiency assumption	An exception to the rule might be misused in order to by-pass the rule.

4:11.3	Critical Chain Planning and Buffering
Necessary assumptions	➢ Contrary to the common belief, safety embedded at the task level prolongs the project without providing sufficient safety to the project completion. ➢ Contrary to the common belief, having detailed visibility (having too detailed a PERT network) almost guarantees that control will be lost.
Strategy	Flow is the number one consideration (it is not important to finish each task on time, it is essential to finish each project on time).
Parallel assumptions	➢ The bigger the uncertainty, the bigger the safety embedded in the task's time estimates. In the vast majority of project environments safety is at least half of the time estimate. ➢ Shifting the safeties from the tasks to the end of their respective task sequences (paths) not only places the safety in the place where it should be but also requires much less safety than the sum of safeties removed from the tasks. This requires that resources will no longer be judged by meeting their time estimates. ➢ Critical Chain methodology provides a proper guide for where and how much safety should be inserted in project planning. ➢ To get excellent control, it behooves keeping the number of tasks in the PERT network to less than 300 (for huge projects zooming might be needed). ➢ Using templates (when applicable) significantly reduces the planning time and reduces unneeded variations.

Critical Chain Planning and Buffering (cont.)

Tactics	For all projects proper PERT networks are built (using templates where appropriate). The time estimates are cut in half and projects and feeding buffers are inserted according to CCPM. The projects are properly staggered. Proper actions are taken to ensure that resources are aware that their estimates are regarded as just estimates - they will no longer be judged according to meeting their time estimates. The resulting plan is used to properly release projects into operations. The resulting planning ability is used to determine reliable and acceptable due-date commitments for new projects.
Sufficiency assumption	Planning is useless unless it significantly helps operations.

5:111:1	Freezing
Necessary assumptions	➢ Reducing the number of open projects by delaying the introduction of new projects is too slow – freezing open projects is required. ➢ It is unrealistic to expect that project managers will reach a consensus on which projects should be frozen *("I fully agree… as long as my elephant goes through the door first!")*.
Strategy	The number of open projects is quickly reduced to be more inline with better flow and throughput.
Parallel assumptions	➢ In the extreme case, when there are not enough projects in execution, "Starvation" lowers the rate of projects completion. In the opposite extreme, when there are too many projects in execution, "Bad-Multi-Tasking" lowers the rate of projects completion. Between these two extremes there is a (almost) plateau. ➢ Having prolific Bad-Multi-tasking is a clear indication that a system is in the second extreme case. Reducing the load by 25% will move the system away from one extreme without the danger of reaching the other extreme. ➢ A person in charge of a cluster of projects can and should decide on their relative priorities.
Tactic	➢ The top manager in-charge of all projects, after consulting with his subordinates, determines the prioritization of projects and instructs to freeze (cease activities on) enough* of the lowest priority projects. * "Enough" means: responsible for at least 25% of the load. ➢ The proper actions are taken to ensure full adherence to the freezing decision.

5:111:2	Accelerate project completion
Necessary assumptions	There is an optimal number of resources per task and per project. In most multi-project environments the eagerness to start all projects as fast as they are won causes spreading resources too thin between projects. This practice causes the lead time of all projects to increase and promotes bad multi-tasking.
Strategy	There is good assignment of resources to projects.
Parallel assumptions	➢ Manning of projects according to their optimal number of resources (rather than trying to squeeze in more projects) leads to an overall increase in the rate at which the Company finishes projects while decreasing the projects' lead-times (in some environments by up to 25%). ➢ The freeze causes many people not to have an active assignment (and people standing idle spread demoralization).
Tactics	➢ The optimal number of the various types of resources needed for each open project is determined. The freed resources are used to prudently strengthen the open projects. ➢ Proper manning decisions are also done for the frozen and to be released projects.

5:111:3	Defrost Mechanism
Necessary assumptions	➢ Defrosting projects too early will, again, flood the system with work. ➢ Defrosting projects too late will lead to starvation of work and unnecessarily extend projects' lead times.
Strategy	Frozen projects are defrosted at a pace that maintains the reduced load.
Parallel assumptions	➢ The level of the reduced load is approximately maintained when defrosting projects is in-sync with projects being completed. ➢ Defrosting projects in-sync with the link that determines the pace of projects completion, also provides focusing on which actions/initiatives help and which jeopardize the flow. ➢ In multi-project environments the factor that determines the pace of project completions is not the most loaded department but the synchronization between the various "legs" of the projects. ➢ Integration is the link where, for each project, the various legs are coming together. ➢ Having too many projects in integration diffuses the efforts to complete projects according to their priorities since whenever a problem that requires chasing a resource from another department is encountered the tendency is to work on another project.
Tactic	The company chooses integration (or part of it) as the VIRTUAL DRUM : The number of projects allowed in that section is restricted to be, at most, 75% of the current number. When a project completes this integration a frozen project is defrosted. The sequence of defrosting projects is according to the agreed projects prioritization.

More on Integration:

The primary goal is to increase flow

To maximize flow you select the slowest operation (bottleneck) as the drum signal and after a project passes through it, that is a signal to release the next project into operations.

This technique minimizes WIP and speeds up flow

In project management it is difficult at best to predict which resource will be the slowest (bottleneck) operation or resource. It will likely vary – be dependent on the variability and timing of resource needs of any particular project.

Remember the goal is to increase flow / Through Put of the system. What dictates the Through Put (T) of a Project Management system?

T = the rate at which the system completes projects

More on Integration

The bottleneck when exposed in a critical chain implementation, will reveal that it is overloaded by bad multitasking, expediting and severely impacted by bad scheduling disguised as efficiency measurements.

The real answer of what dictates the T of the system is minimizing / eliminating mis-synchronization
Mis-synchronization is what limits the T of the system

Integration is the point in time where the parallel (concurrent and dependent legs) parts of a project network come together to complete the project from that point forward. Considering the real life variability as described above it makes sense that the integration is the point chosen as the virtual drum to dictate the pace of release projects into the system.

Experience in Critical Chain Project Management (CCPM) has told us that integration is the best point in a project network timeline to be the drum point signal - as a project passes through the integration point, it is the signal to release the next project.

Side rule from experience –
All the mis-synchronization is accumulated at integration

5:111:4	Releasing of new projects
Necessary assumptions	For most projects there is vast difference between the lead-times of their various “legs”; there is no one date for release of a project. Release of all legs of the project at one shot increases unnecessarily the load. Note: For frozen projects most “legs” have already been released.
Strategy	The timing for the release of each “leg” of a new project takes into account the lead-time of the leg.
Parallel assumptions	➢ For most multi-project environments it is too cumbersome to manually calculate properly the release dates of the various legs of new projects. ➢ Most project environments (and most commercially available software) do not consider the fact that the lead time of the various “legs” of a project are also a function of the load on the various resources (Critical Path vs. Critical Chain – removing resource contentions). ➢ The lead time of a project and the lead time of the various legs of a project are a function of the way safety is included (safety in the task level or in the project level - Project and Feeding Buffers). Most project environments (and most commercially available software) do not use the concept of Project and Feeding Buffers.
Tactic	When the time arrives to release new projects, steps 4.12 and 4.13 should be in place. At that stage, a system to release new projects using the CCPM concepts is ready.

5:112:1 Preparations according to priorities

Necessary assumptions	In most multi-project environments the importance of complete preparations – “full kit” – is frequently/constantly radiated by top operational managers. The mere fact that delays and even rework caused by missing preparations are so prevalent, indicates that usually the drive to “full kit” quickly deteriorates to lip service.
Strategy	Resources and project leaders are used to working on projects whose preparations are (almost) fully completed.
Parallel assumptions	➢ A powerful way of turning a good mode of operation into the norm is to ensure that each resource experiences first hand that mode of operation, and enjoys the outcome. This can be accomplished by using the freed-up time to complete the preparations on the running projects. ➢ The things that are missing are usually things for which there is some difficulty to complete. Therefore, if given the option, resources working on preparations would prefer to focus on preparing new projects about to be released rather than relentlessly chasing the preparations gaps on open projects.
Tactic	A Full-Kit manager is appointed. The relevant resources are instructed to complete the preparation steps first for the running – not frozen - projects. Then to complete the preparations for frozen projects. Only when (most of) the above is done they are guided to work on the preparations for the new projects waiting to be released. They always follow the projects priority.

5:113:3	Staggering Project Portfolio
Necessary assumptions	In multi-project environments most key resources work across projects. Not considering resource contentions across projects makes the plan unrealistic to start with and encourages, by design, bad multi-tasking.
Strategy	Projects are planned to ensure effective operation.
Parallel assumptions	➢ An effective way to deal with resource contention across projects is not to try and resolve each resource contention (a futile, exhausting, exercise bearing in mind that the actual time the work is performed is likely to be shifted due to the high variability) but rather to do good enough smoothing of the load on each resource type. The temporary peak loads that remain in the plan (and the many more peak loads caused by Murphy) are absorbed by the buffers. ➢ A VIRTUAL DRUM staggers the projects in accordance with the actual pace of the system. Therefore, it effectively smoothes the load on each resources type. ➢ Emulating the VIRTUAL DRUM in the planning stage resolves the resource contention problem. (cont. on the next page)

Staggering Project Portfolio (Cont.)

Parallel assumptions

Emulating VIRTUAL DRUM in planning – the STAGGERING mechanism:

1. For all projects consider ONLY the tasks performed by the chosen integration area.
2. Following the projects priority, place these tasks on a time line, obeying the restriction of number of projects allowed to be worked on in that integration area - Staggering.
3. Adjust the time estimations of the tasks on the time line to reflect the actual rate at which projects finish this integration.
4. For each project use the time determined for the integration tasks as an anchor to place all other activities.
5. Examine the resulting load on key resource types. If there are peak loads that cannot be absorbed within half of the corresponding buffers check for and correct errors in the data.
6. If a certain project is planned to be completed significantly after its committed due-date, better inform the client now.

Tactic

- A proper team invests the time needed to emulate the VIRTUAL DRUM and to identify and correct the crucial data errors.
- Actions are taken to ensure that projects are released according to the plan (legs having different lead-times are released at correspondingly different dates).
- Actions are taken to ensure that due dates for new projects are committed ONLY according to the STAGGERING mechanism (or top management's decision to postpone a specific existing project).

5:114:1	Task completion reporting
Necessary assumptions	➢ Variability (and its big brother Murphy) changes priorities. ➢ In most multi-project environments, frequent reporting on progress by task managers is constantly demanded. Still the frequency and accuracy of the reports is far from satisfactory.
Strategy	The required data is always adequately available.
Parallel assumptions	➢ People tend to procrastinate on their reporting when reporting doesn't have an immediate/significant impact on them. ➢ Traditionally the things demanded to be reported by task managers are used for financial purposes (calculating the cost absorbed by the projects). In multi-project environments this use has no relevancy to the task managers. ➢ In multi-project environments the pressure, exerted from all sides, makes it very important for task managers to know the true priorities. ➢ The data that is essential to determine priorities is not the amount of time already invested in a task but the estimation of the time still required for the task to be finished (task status). ➢ A delay in a task (and an expected delay) can change the critical chain resulting in a major change in priority to tasks of many task managers. Conclusion: when there is a proper priority system, daily reporting on tasks' expected completion dates is extremely helpful to task managers.
Tactics	➢ Proper explanation is given to all task managers: what is required from them to report on a daily basis, how this information is going to be used and that they will, at last, be able to obey ONLY the formal priority list. ➢ The company launches the daily reporting (by task managers – not by the resources) procedure and relentlessly enforces it.