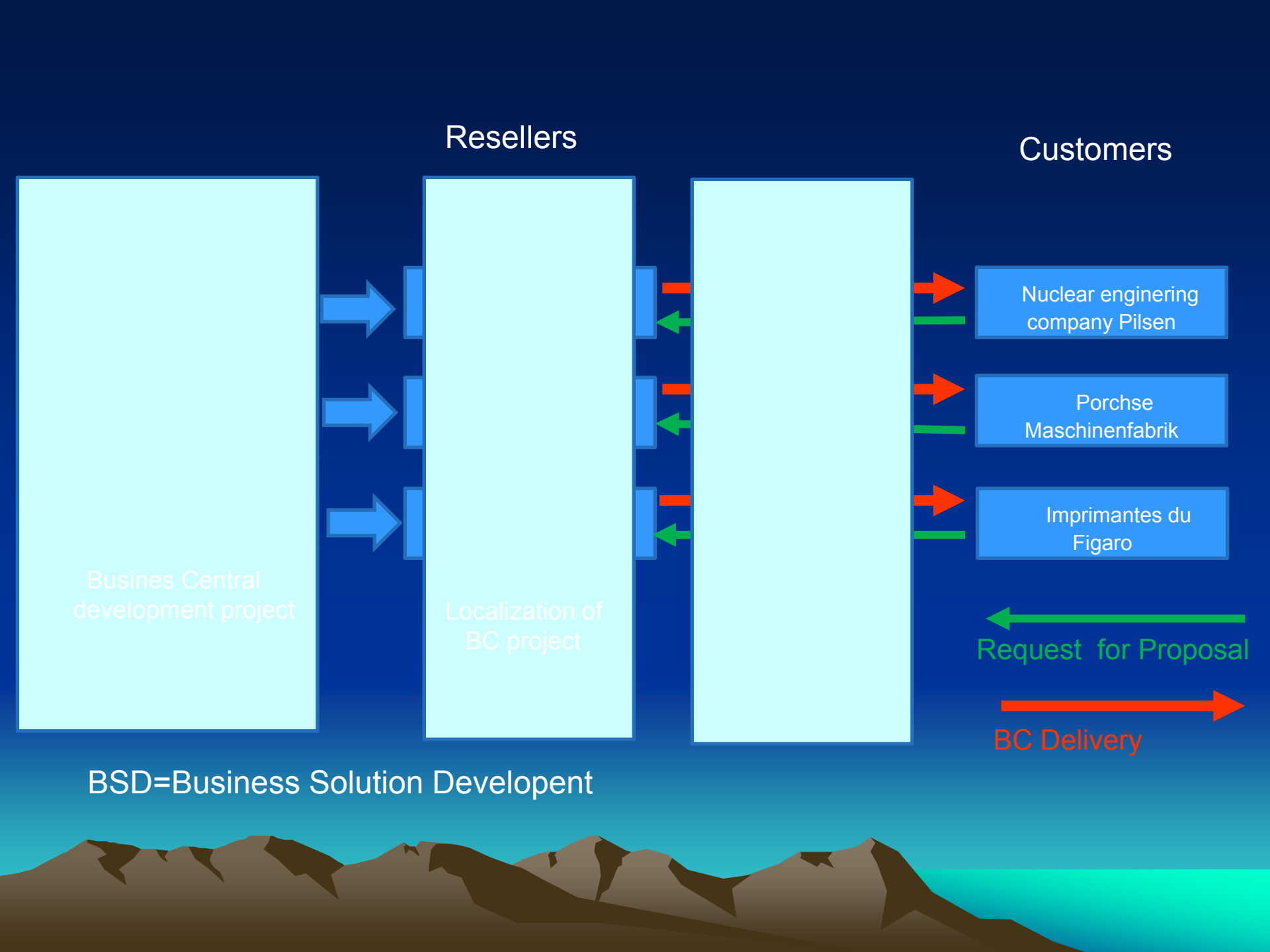
TOC – Critical chain

Ing.J.Skorkovský, CSc,
Department of Business Management
FACULTY OF ECONOMICS AND ADMINISTRATION
Masaryk University Brno
Czech Republic


Resellers
Customers
Busines Central development project
Localization of BC project
Nuclear enginering company Pilsen
Porchse Maschinenfabrik
Imprimantes du Figaro
Request for Proposal
BC Delivery
BSD=Business Solution Developent

Different approaches to Project Management

Waterfall methodology–Big design up front, milestones,..**no iterations !!**
It is based on prediction

Agile methodology– Scrum (Sprints,..)- **iterations approach**

Prince2 - Projects In Controlled Environments (rolling wave planning)

Lean management

Projects often end up being :

They exceed the planned budget

They are not completed on time

Not everything that was required is delivered or the volume of delivery is reduced

TOC concisely I (see PWP presentation about TOC)

A modest repetition of TOC, because the theory of the critical chain is built on this theory

- origin: Eliahu Moshe Goldratt, Jerusalem
- cost world<->throughput world
- analogy weight of the chain – solidity of the chain
- how to find a bottleneck? -> CRT
- tools of TOC – tree structures (5 trees)
- Five TOC steps
- CRT – EC – TT – PT – FRT meaning:
- Current Reality Tree - Evaporating Cloud Tree– Transition Tree - - Prerequisite Tree – Future Reality Tree

TOC concisely I (see PWP presentation about TOC)

- **bottleneck** in the project management is a critical path (so no one resource only)
- finding (assessment) of bottleneck is not easy and often it is not explicit (uncompromising)
- everybody knows something about TOC and nobody knows **how to implement it** to the real world- and this is again another bottleneck (tendon of Achilles from the heel to the nape)

TOC-five steps (revision)

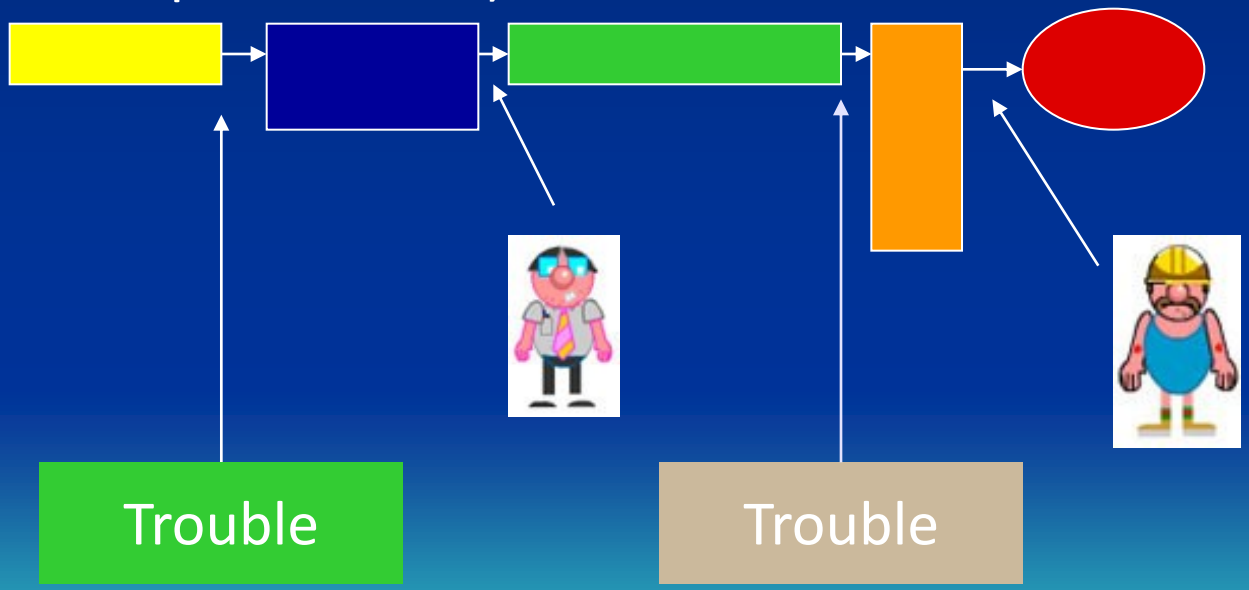
Five steps process:

Step 0. Identify the Goal of the System/Organization
Step 0.5 Establish a way to measure progress to Goal

- **Step 1.** ***Identify*** the system's constraint.
- **Step 2.** ***Exploit*** the system's constraint**.**
- **Step 3.** ***Subordinate*** everything else to the above decision.
- **Step 4.** ***Elevate*** the system's constraint**.**
- **Step 5.** If a constraint is broken (that is, relieved or improved), go back to Step 1. But don't allow ***inertia*** to become a constraint**.**

Linear image of very simple project

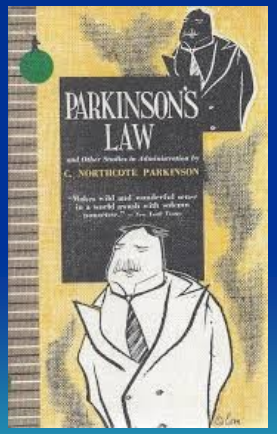
- activities – abscissas – Gantt graph
- constantly changing conditions (Parkinson low, Murphy low, Student syndrome, customer changes - „fancies “,“caprices“.. ….)



Next slides

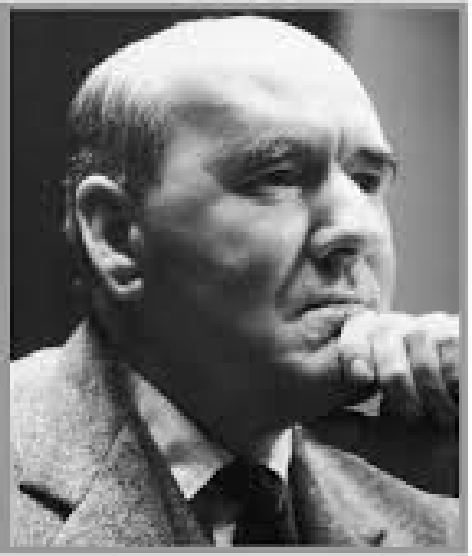
Parkinson s law

- **Parkinson's Law** is the old „folk wisdom“ that work expands to fill the time allotted.
- Put simply, the amount of work required adjusts (usually increasing) to the time available for its completion.
- The term was first coined by Cyril Northcote Parkinson



Parkinson s law

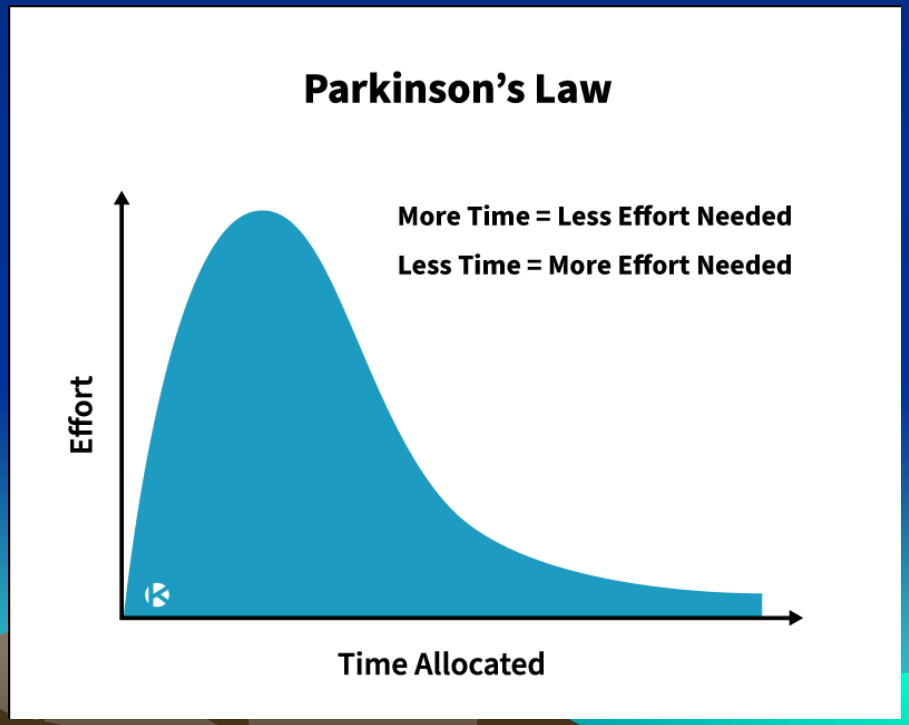
"The time spent on any item of the agenda will be in inverse proportion to the sum involved"



Cyril Northcote Parkinson
British Naval Historian
1909-1993

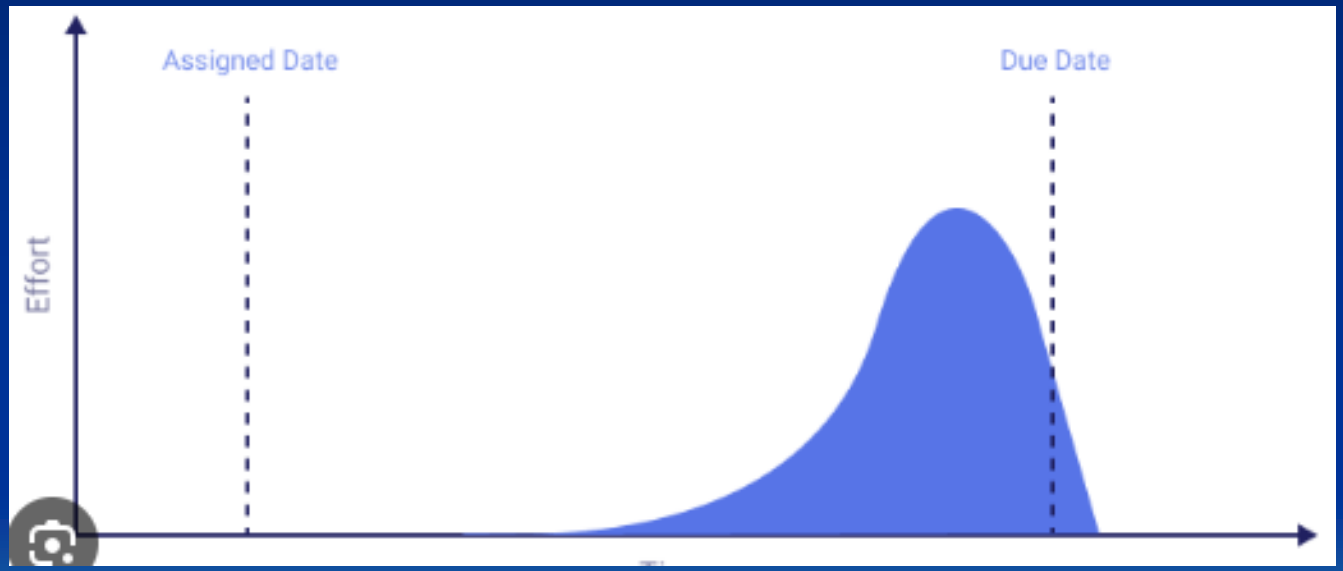
Parkinson s laws

- Work complicates to fill the available time
- The demand upon a resource tends to expand to match the offer of the resource. The reverse is not true.

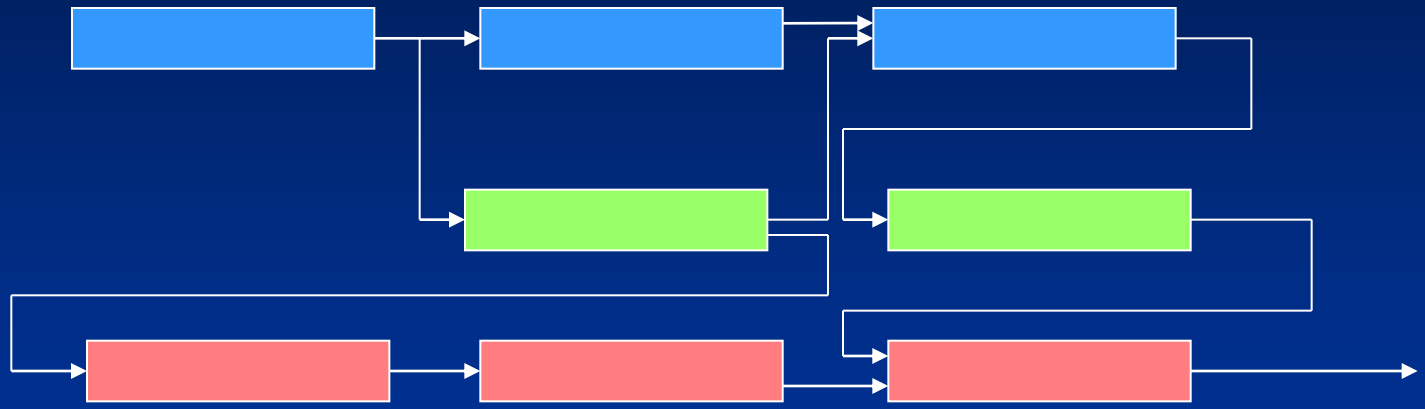


Student syndrom

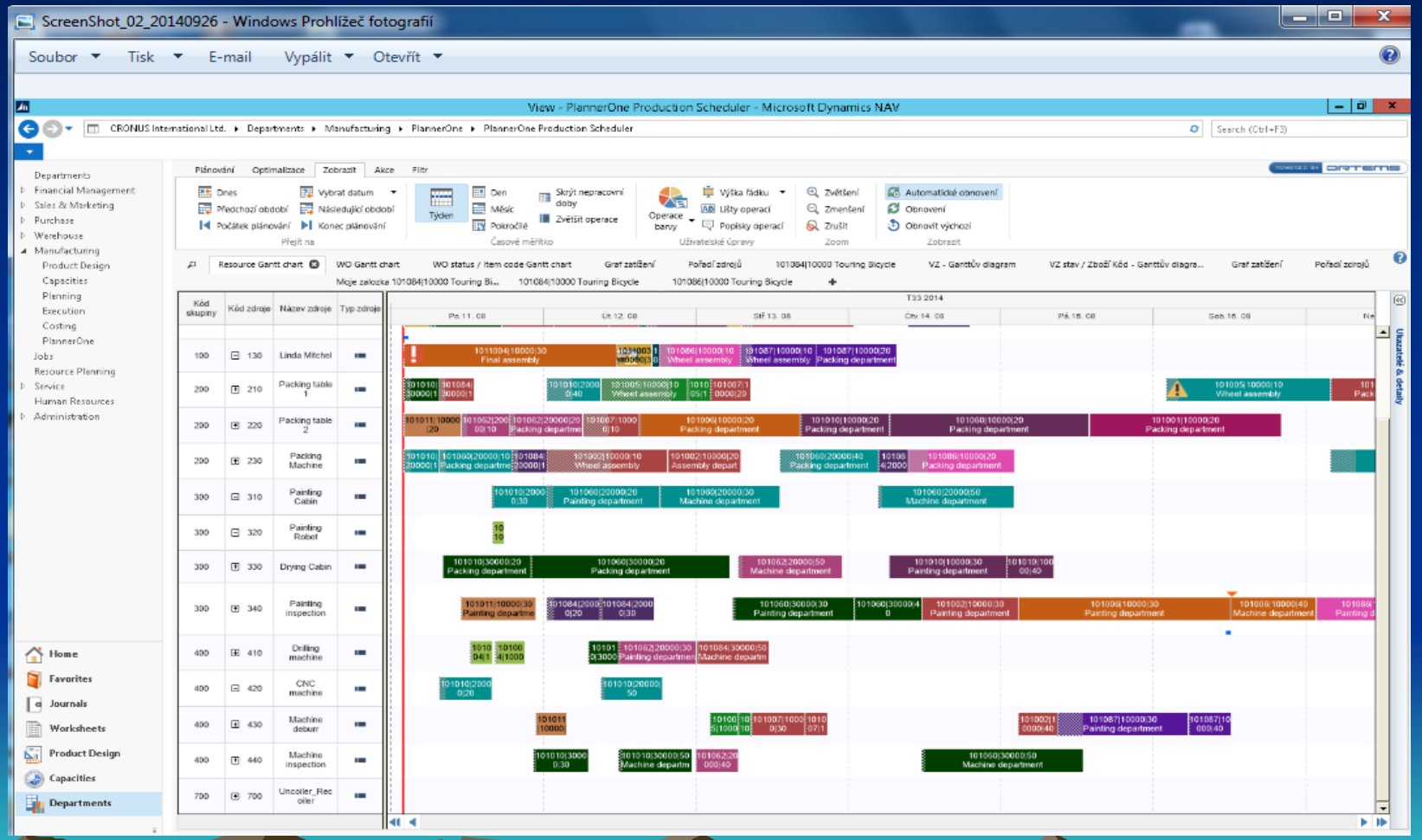
Professor: "Do you know how long it will take you to become fluent in Japanese?
"Student: "I know, one night before the exam."



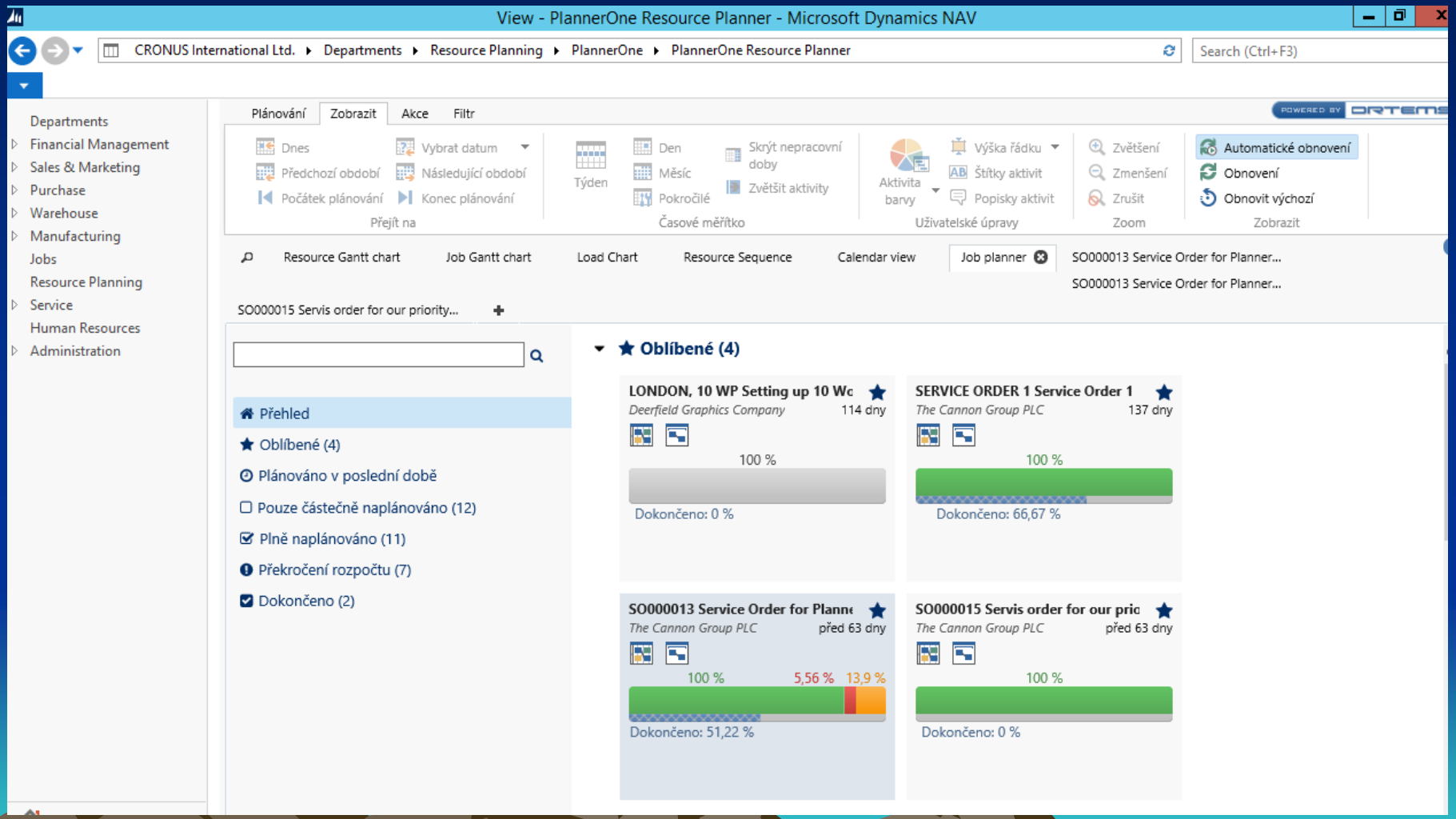
Parallel image of simple project



PlannerOne Scheduler- Central Business Central Application



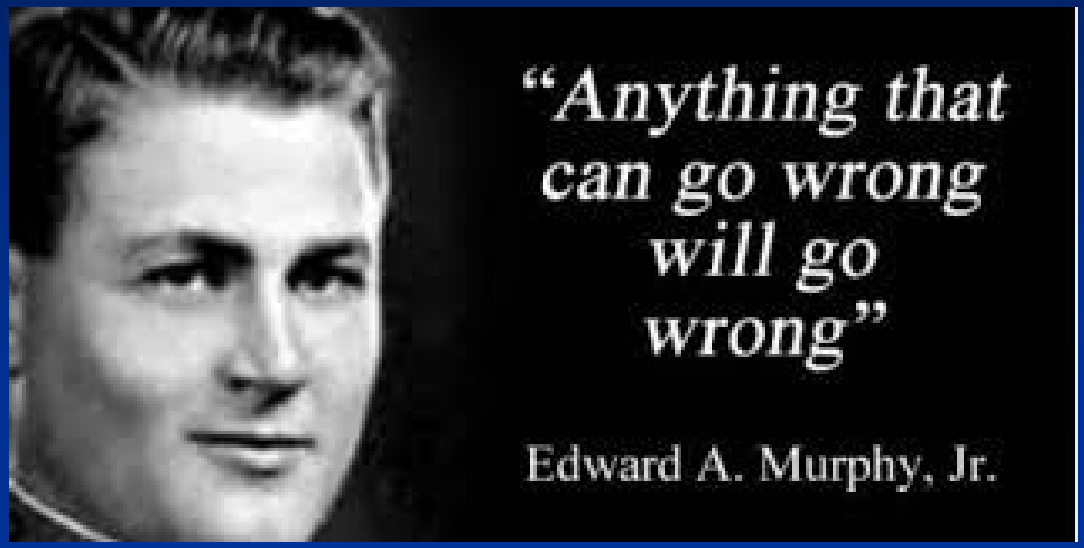
PlannerOne Resource Planner



Project and its budget

- price of the whole project (see MS Dynamics BC on next slide)
- project length (time) – duration
- project stages and length of each activity (task)
- assigned resources to every activity and their capacities (time per defined period)
- time reserves (buffers-time reserves) and estimation of the buffer sizes
- unfavourable influences (see Murphy s lows - http://murphy.euweb.cz, etc.)
- additional activities (unexpected costs)

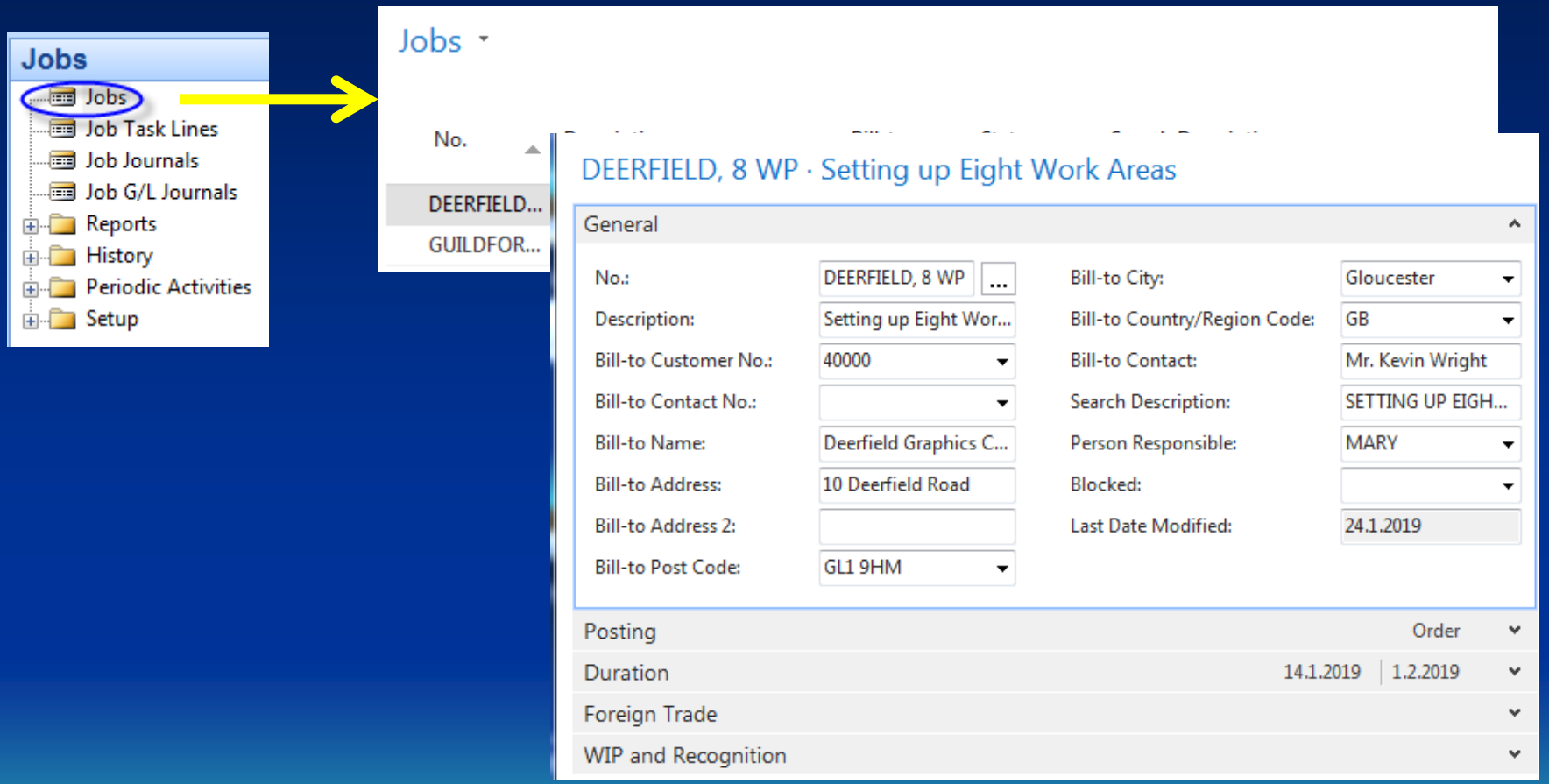
Murphy s law



Selected Murphy s laws

- If your attack is going well, you have walked into an ambush (trap)
- Planner is alerted about modification of the plan exactly in the moment, when the plan is finally adjusted
- To carry out **n+1** trivialities you need two times more time than time necessary to carry out **n** trivialities (law 99 %)
- **If anything can go wrong, it will**
- Any given program, when running, is obsolete
- No matter how many resources you have, it is never enough

Projects (jobs) in Buiness Central



Projects and MS Business Central

Job=Project->BC terminology

Job Task Lines

Type to filter (F3) | Job Task No.

Filter: DEERFIELD, 8 WP

Job Task No.	Description	Job Task Type	Totaling	Job Posting Group	WIP-Total	WIP Method	Start Date	End Date	Schedule (Total Cost)
1000	**Setting up Eight Work Areas**	Begin-Total							
1100	**Preliminary Services**	Begin-Total							
1110	Determining Specifications	Posting		SETTING UP			13.1.2019	13.1.2019	107,80
1120	Selecting Furnishings	Posting		SETTING UP			14.1.2019	14.1.2019	107,80
1130	Obtaining Customer Approval	Posting		SETTING UP			17.1.2019	17.1.2019	107,80
1190	**Total Preliminary Services**	End-Total	1100..1190						323,40
1200	**Assembling the Furniture etc.**	Begin-Total							
1210	Assembling the Furniture etc.	Posting		SETTING UP			23.1.2019	23.1.2019	11 000,10
1290	**Total Assembling the Furniture**	End-Total	1200..1290						11 000,10
1300	**Closing the Job**	Begin-Total							
1310	Meeting with the Customer	Posting		SETTING UP			27.1.2019	31.1.2019	107,80
1390	**Total Closing the Job**	End-Total	1300..1390						107,80
9990	**Total Setting up Eight Work Areas**	End-Total	1000..9990						11 431,30

List o tasks and related costs (scheduled and used)

Home study

Schedule : The planning line contains expected usage for the job that will **not be invoiced** to the customer. You use this option if the costumer will be invoiced from a different planning line (of type **Contract** or **Both Schedule** & **Contract** types), or if the expected usage for this planning line is not chargeable.

Contract : The planning line specifies an amount **that should be invoiced** to the customer, but no usage relates to the line so far. You use this option if no schedule of usage has been planned for the job, or if the expected usage for the job has been specified on different planning lines (of type Schedule).

Line Type	Planning Date	Document No.	Job No.	Job Task No.	Type	No.	Description	Unit of Measure Code	Quantity	Unit Cost	Unit Cost (LCY)	Unit Price	Total Cost
Schedule	25.01.12		DEERFIELD, ...	1210	Resource	MARK	Delivering and Assembling	HOUR	20	31,90	31,90	54,00	638,00
Contract	25.01.12		DEERFIELD, ...	1210	Resource	MARK	Delivering and Assembling	HOUR	20	31,90	31,90	54,00	638,00

Projects in Business Central

Resource and assigned capacitiy

No.	Name	02.10.15	03.10.15	04.10.15	05.10.15	06.10.15	07.10.15	08.10.15	09.10.15
LIFT	Lift for Furniture	0	0	0	0	0	0	0	0
LINDA	Linda Martin	0	0	0	0	0	0	0	0
MARK	Mark Hanson	4	0	0	8	8	8	8	4
MARY	Mary A. Dempsey	0	0	0	0	0	0	0	0
TIMOTHY	Timothy Sneath	0	0	0	0	0	0	0	0

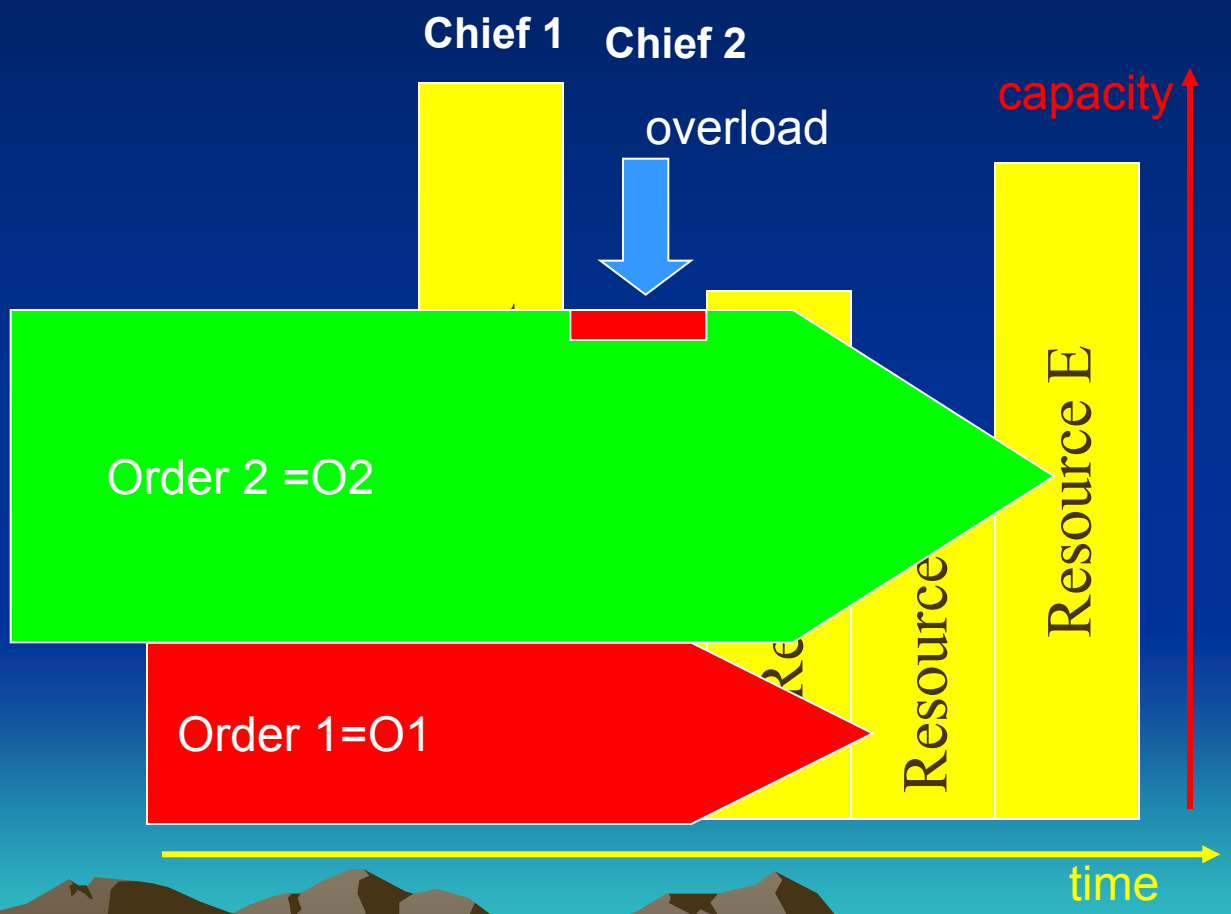
Hours and resource

Resources and orders

Matrix structure of multi-project environment – responsibility of project managers and responsibility of department managers are **in conflict**

Project manager 1 (manages O2

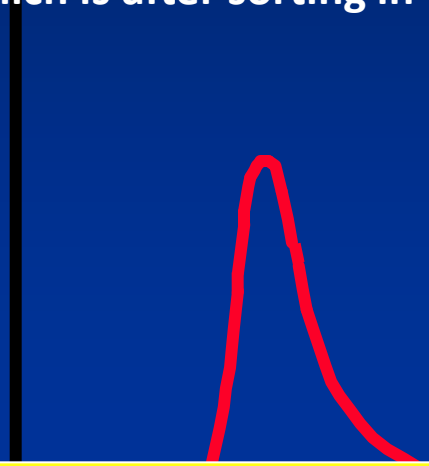
Project manager 2 (manages O1)



Partial time of any activity in the project

Variability of the real time assigned to activity

Probability– median an element of statistical file,
which is after sorting in the middle .Median of the set (1,5,2,2,1) is 2

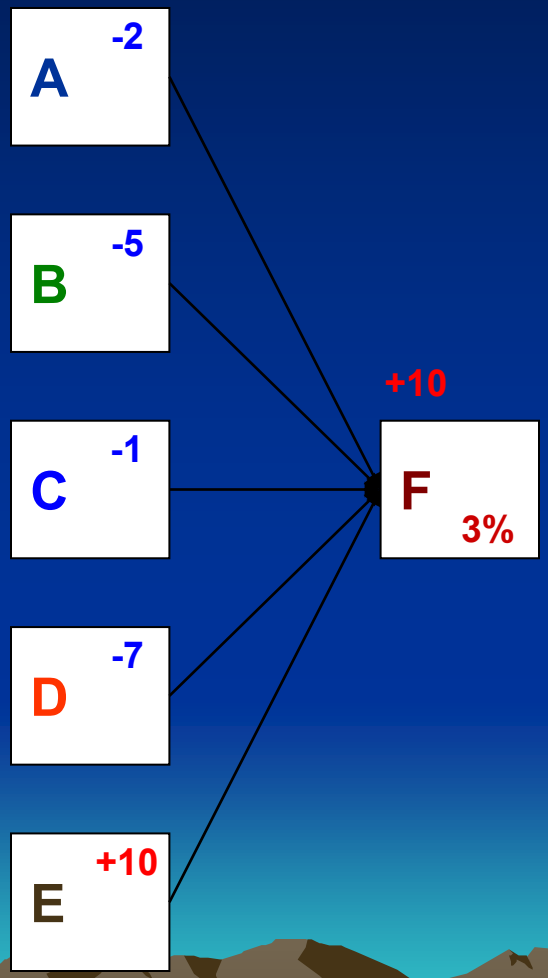


100 „5-miniutes meeting happened.
How many times it took 5 minutes only ?

Colleague ask for a quick rendez-vous: „Do not worry, it will take maximum 5 minutes!“.

How long it takes on average?

Project environment is very complicated because of integration linkages and their dependencies



Probability of finishing tasks A to E in time is 50%. (50*50*….*50=3,125 %)

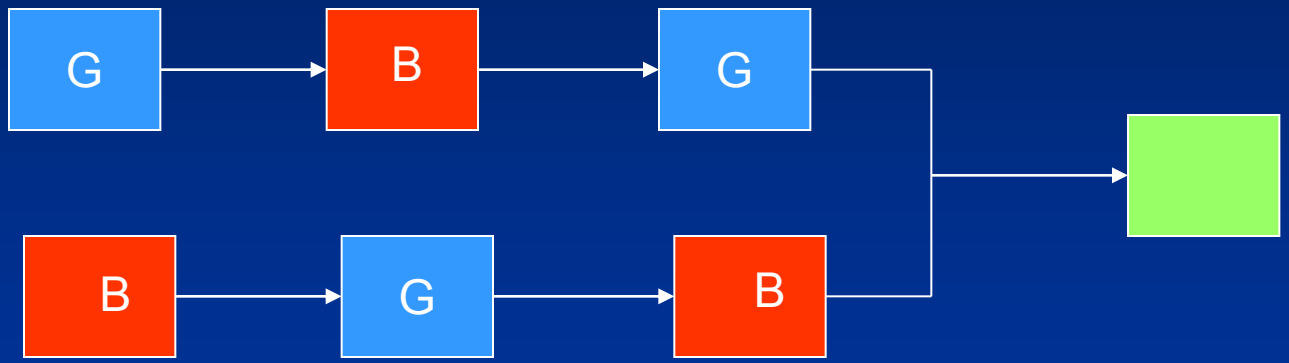
What is a probability, that task F will start in time ?

How the timely finishing of the tasks A,B,C and D will influence the integration point ?

a) saving are fully wasted

b) delay in one task will be immediately transferred to the next project task (activity) see +10

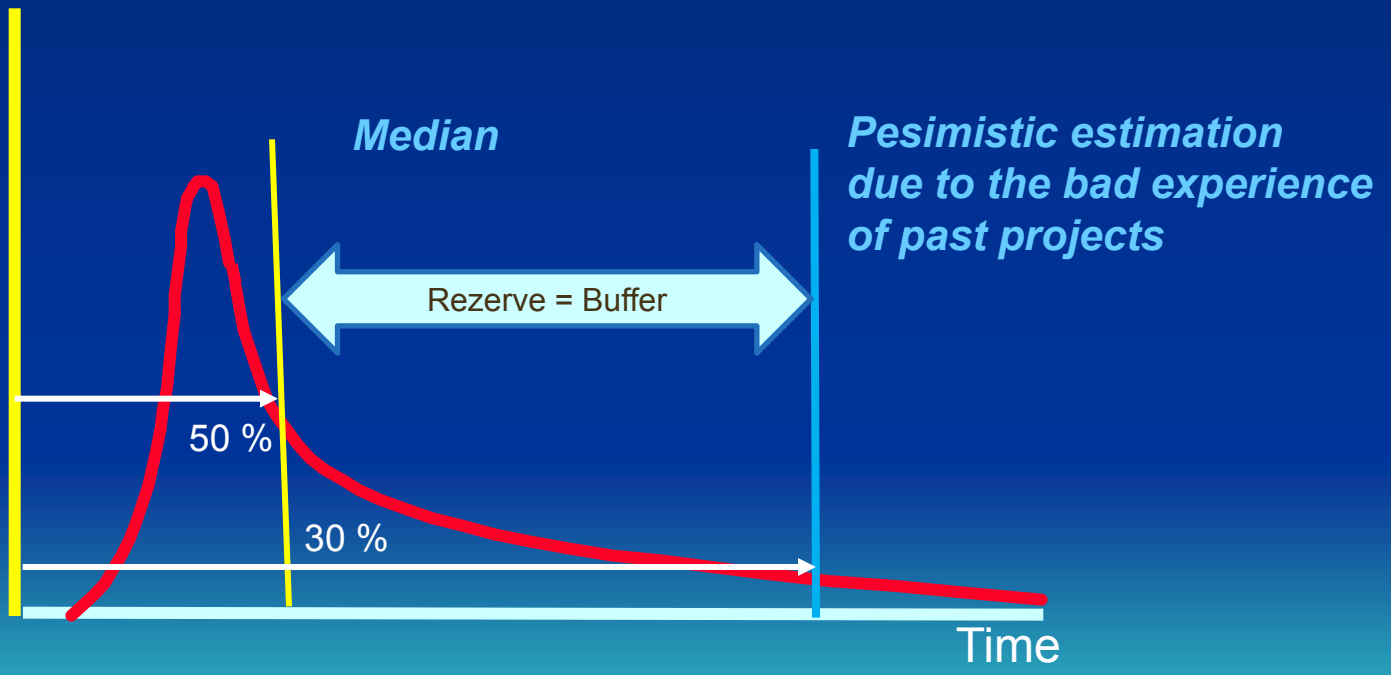
Project environment is very complicated because of integration linkages and their dependencies I



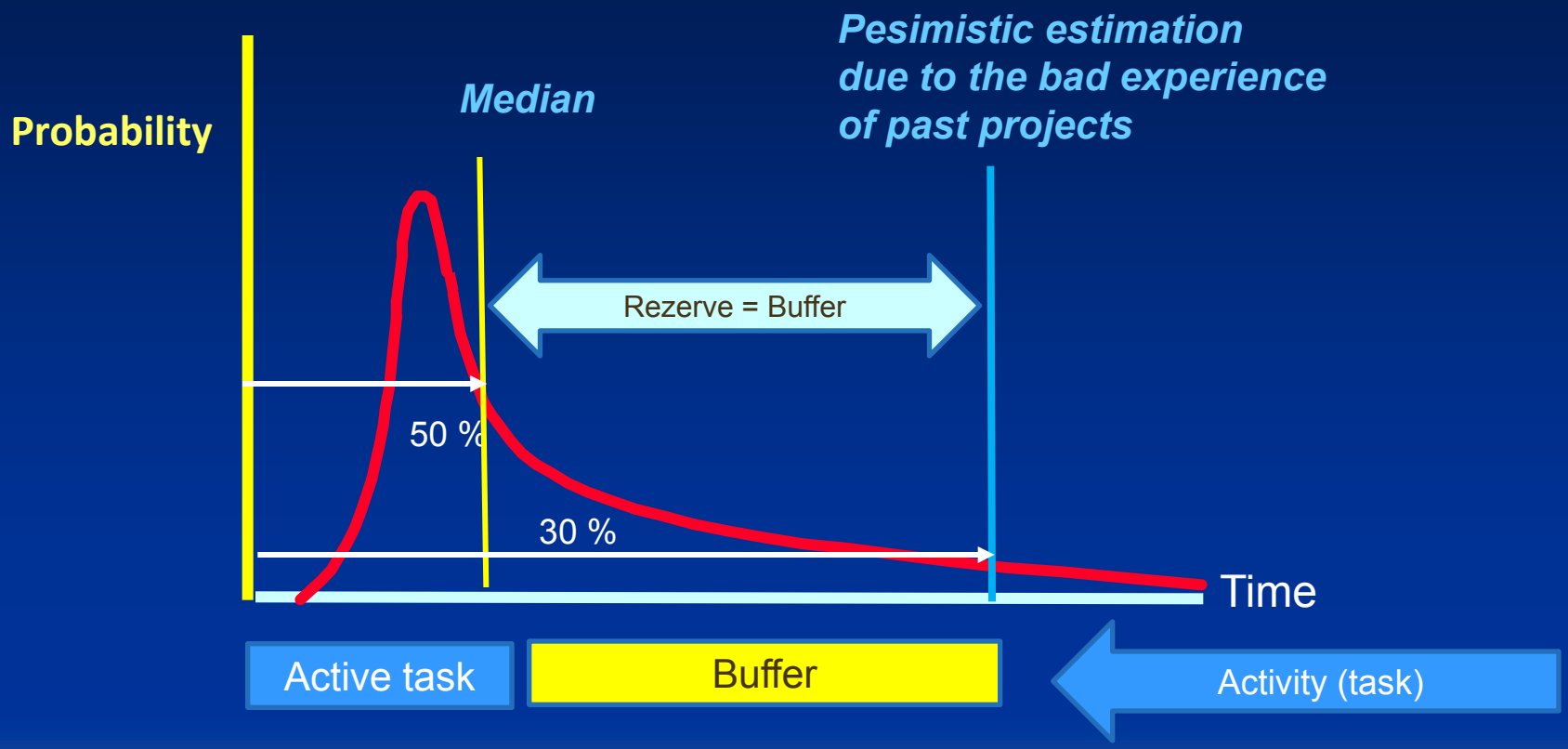
In order to start **B** in the upper branch, you have to finish **G** and also **B** in the lower branch. The probability, that **B** start in time is 50 % worse, than it was shown on the previous slide.

Further options for setting the duration of the activity (Task) I.

Probability– median an element of statistical file,
which is after sorting in the middle .Median of the set (1,5,2,2,1) is 2

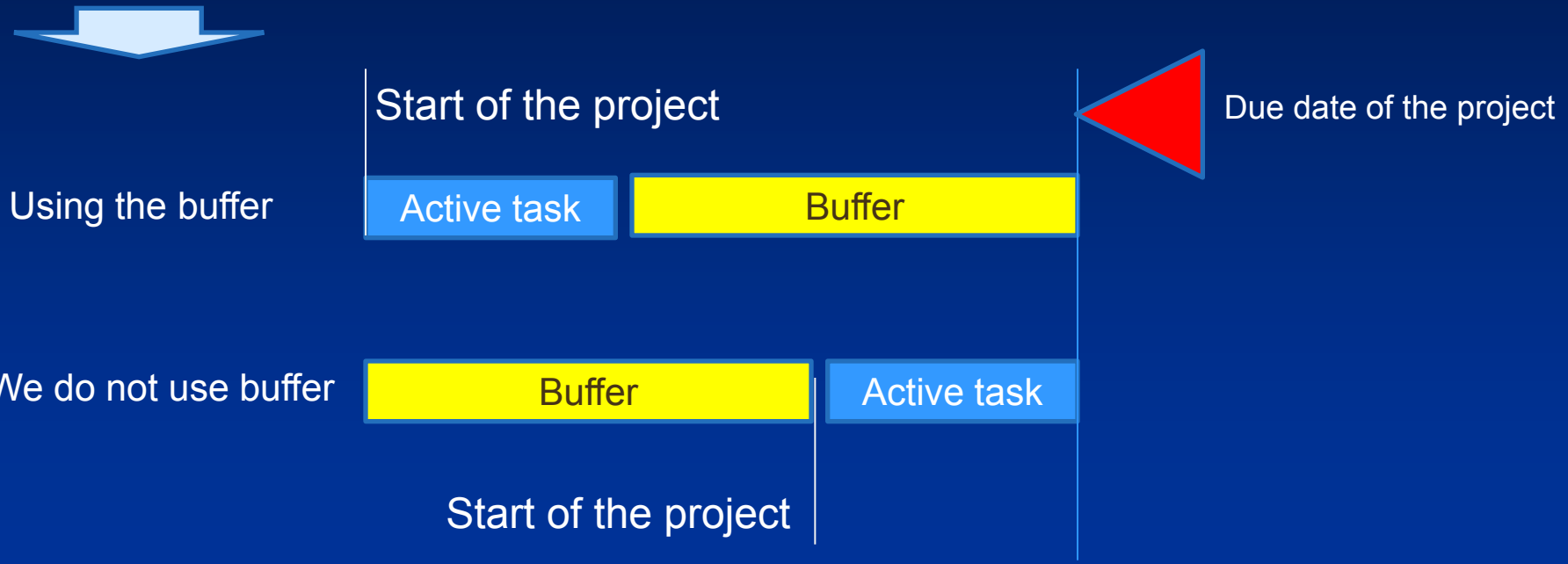


Further options for setting the duration of the activity (Task) II



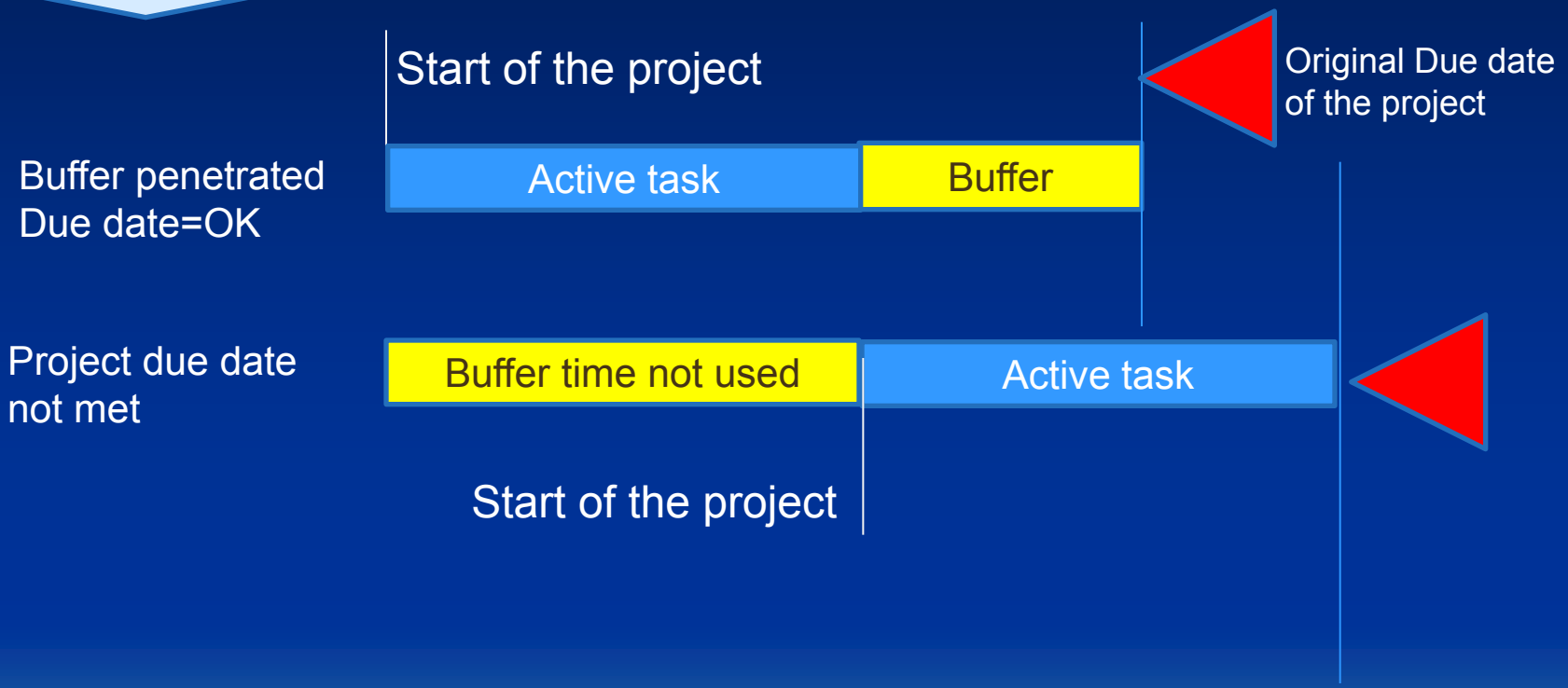
Two approaches to utilization (reserve-buffer placement)

A project that has only one activity

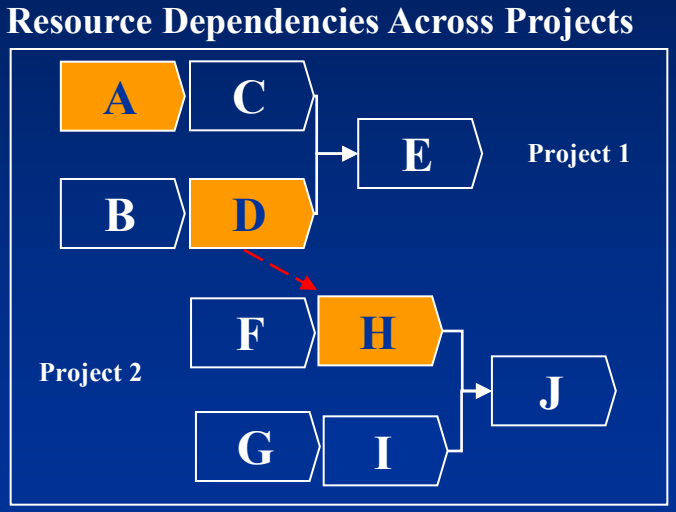


Unexpected project troubles and their impact

A project that has only one activity



Project environment is very complicated because of integration linkages and their dependencies II



Op = operation

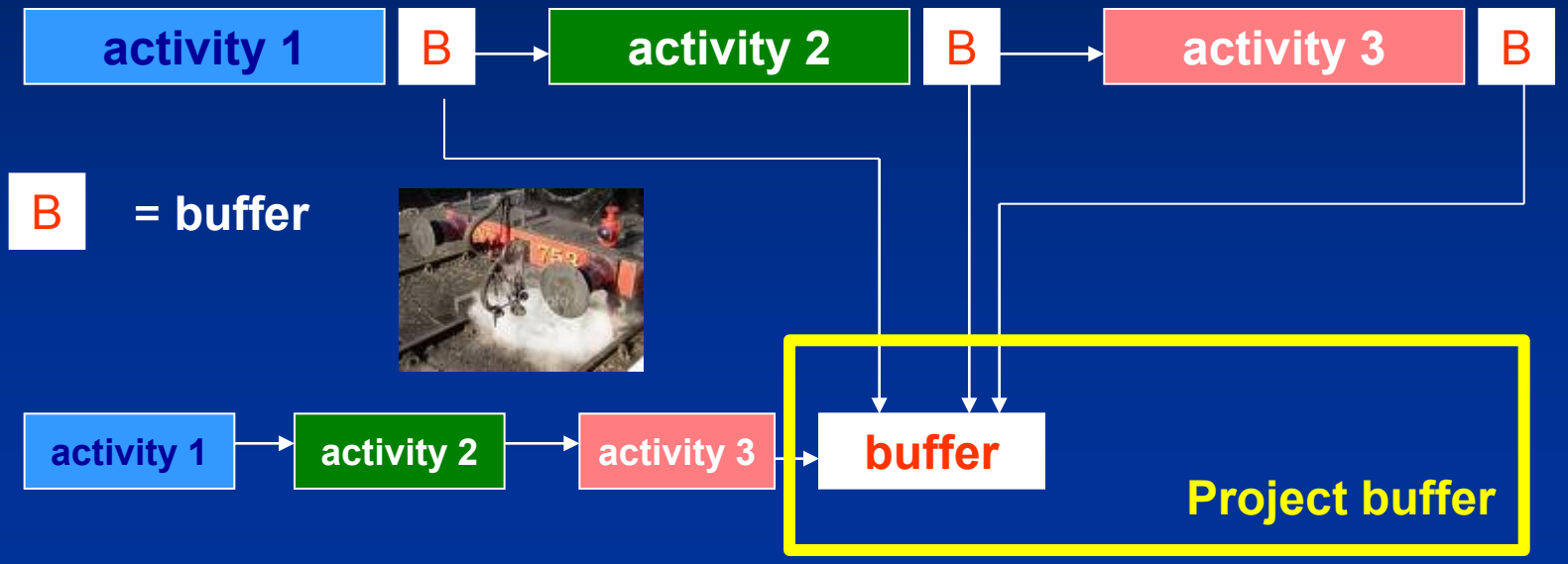
Operations A , D and H are done by the same resource

Delays: If Op D on Project 1 is late, Op H on Project 2 also get delayed as its resource is blocked on Op D

Gains: Even if Op D finishes early, the resource cannot start Op H as has to wait for Op F to finish

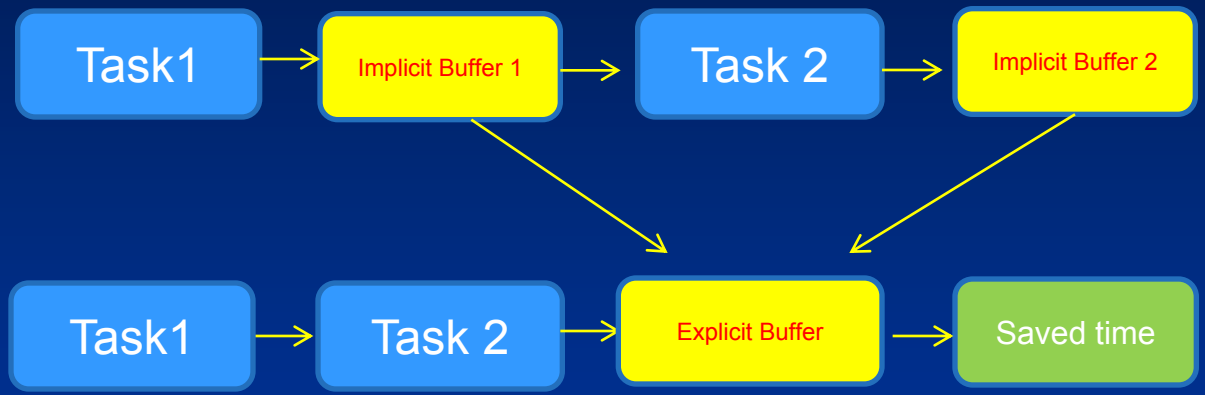
The project must be protected against influences of breakdowns (troubles)

Standard estimation with protecting buffers for every activity



1st step : every activity is shorten to 50 % of its original time size.
2nd step : **critical path buffer** at the end of the project will have size of 50 % of the total sum of saved time created by shortening all partial activities

Simplified scenario CPM and CCPM



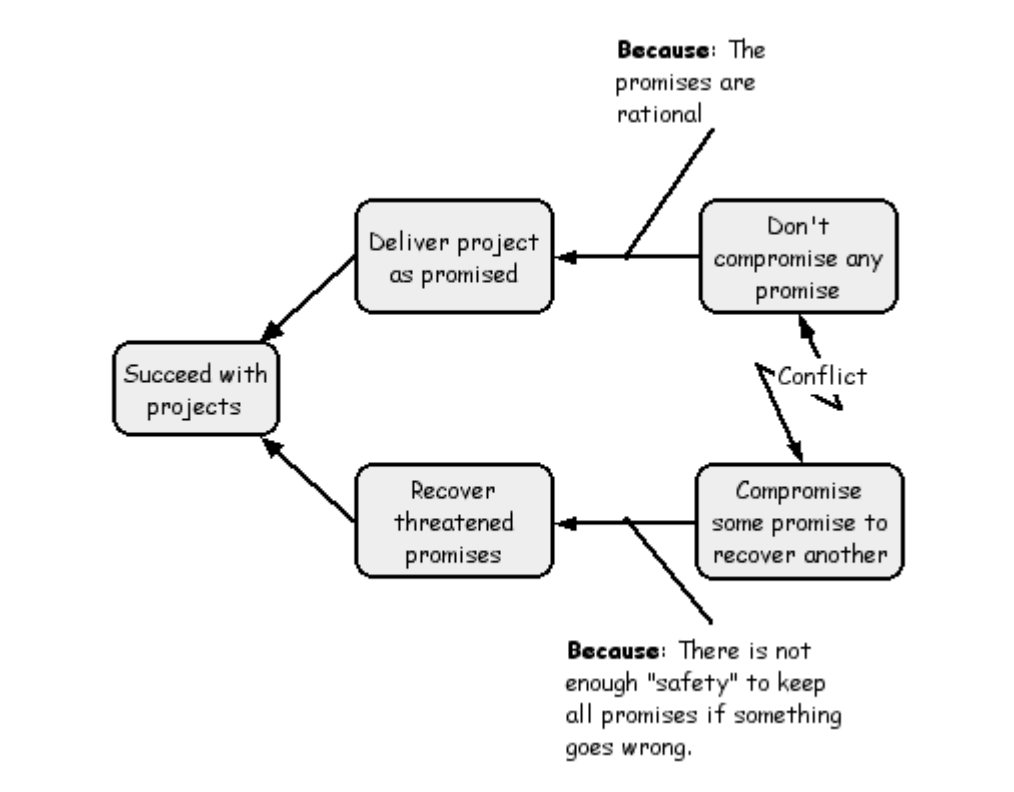
Explicit = directly specified, opened
Implicit = hidden, internally defined, indirect

CPM = Critical Path Method

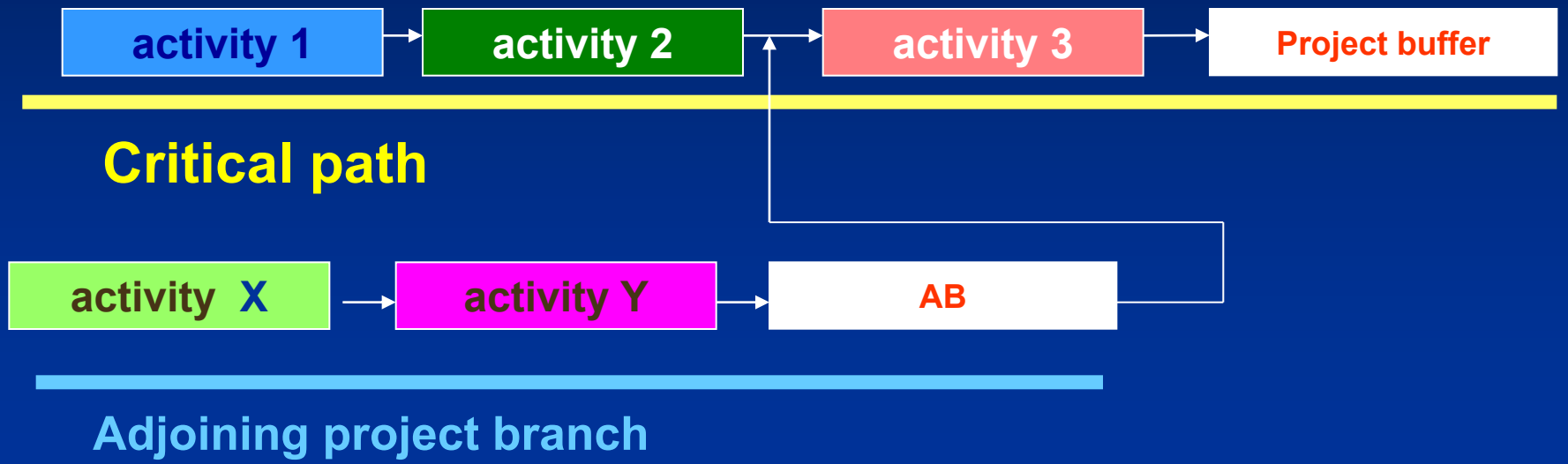
CCPM = Critical Chain Project Methodology

EC and project management

EC=evaporating cloud=see PWP related to TOC



Critical path, adjoining branches of the project and adjoining buffers (AB)



Buffer serves as a safety tool to accumulate reasons of **expected** and **unexpected** delays

Critical Path (**CP**)

• Critical path is defined as **the longest way** (meaning time) from the starting point of the project graph to the ending point.

• Every project has at least one critical path

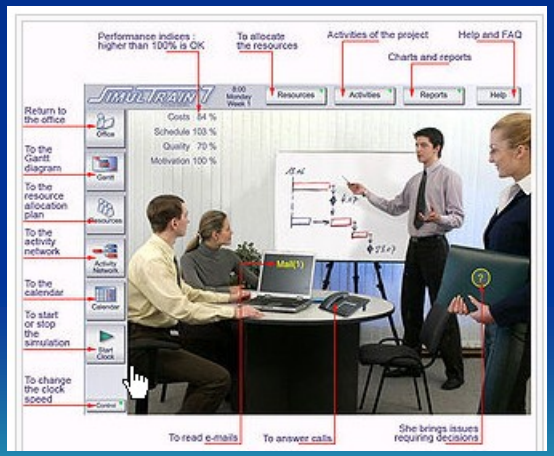
The rules of Critical Path:

• Every delayed task on **CP** will essentially delay the whole project

• Truncation of duration of any task on **CP** will shorten whole project

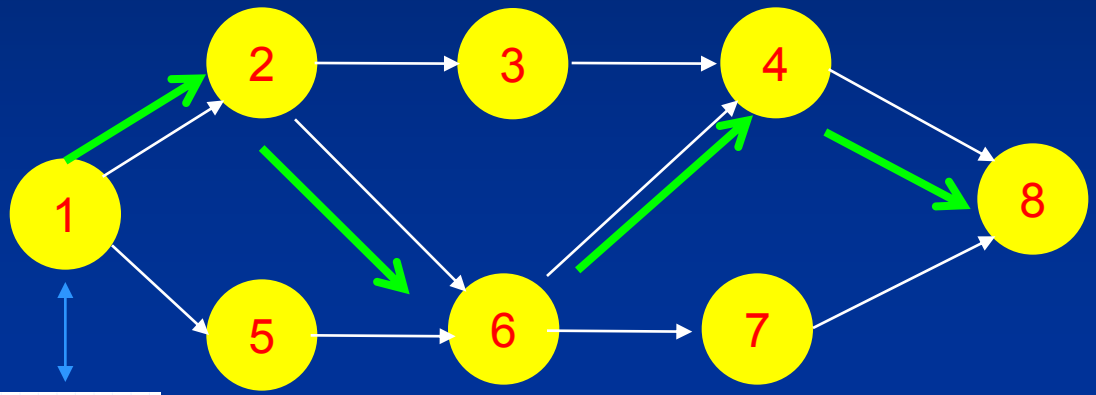
Critical Path (CP)

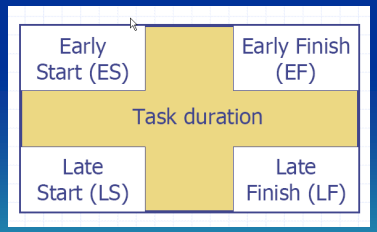
- **Critical Path Method**, abbreviated **CPM**, or **Critical Path Analysis**, is a mathematically based algorithm for scheduling a set of project activities. It is an important tool for effective project management.



Critical Path (CP)

Projet network diagram = any schematic display of the logical relations of the project activities (tasks)

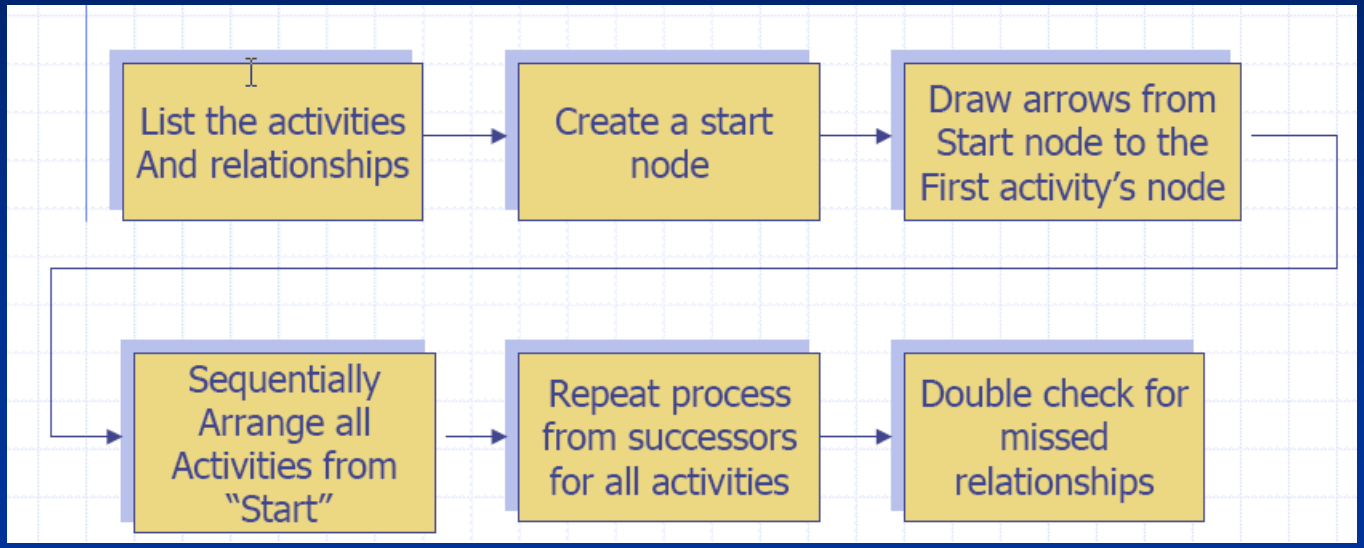




In every node there is both an activity description and an allocated resource

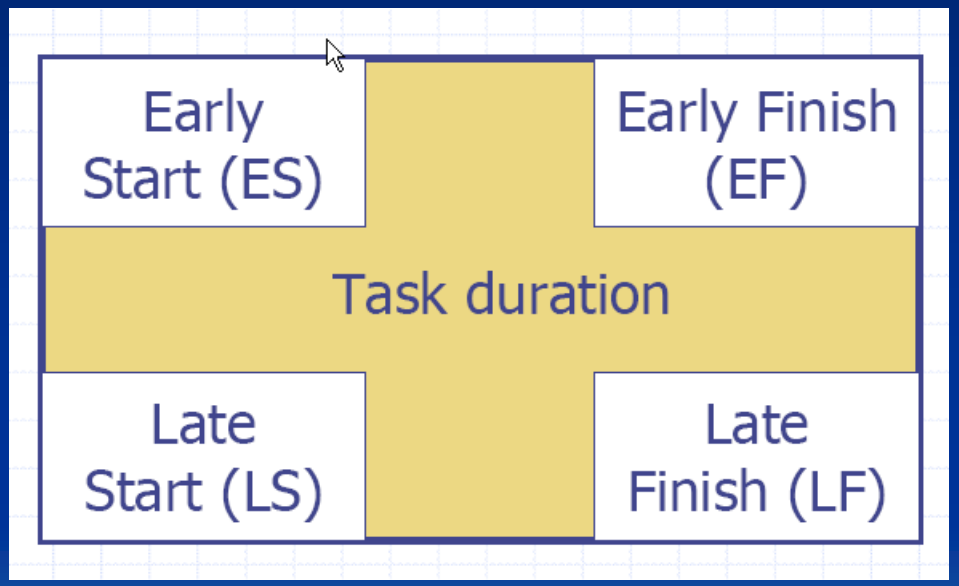
Critical Path (CP)

Building a CP diagram 1



Critical Path (CP)

Building a CP diagram 2



Critical Path (CP)

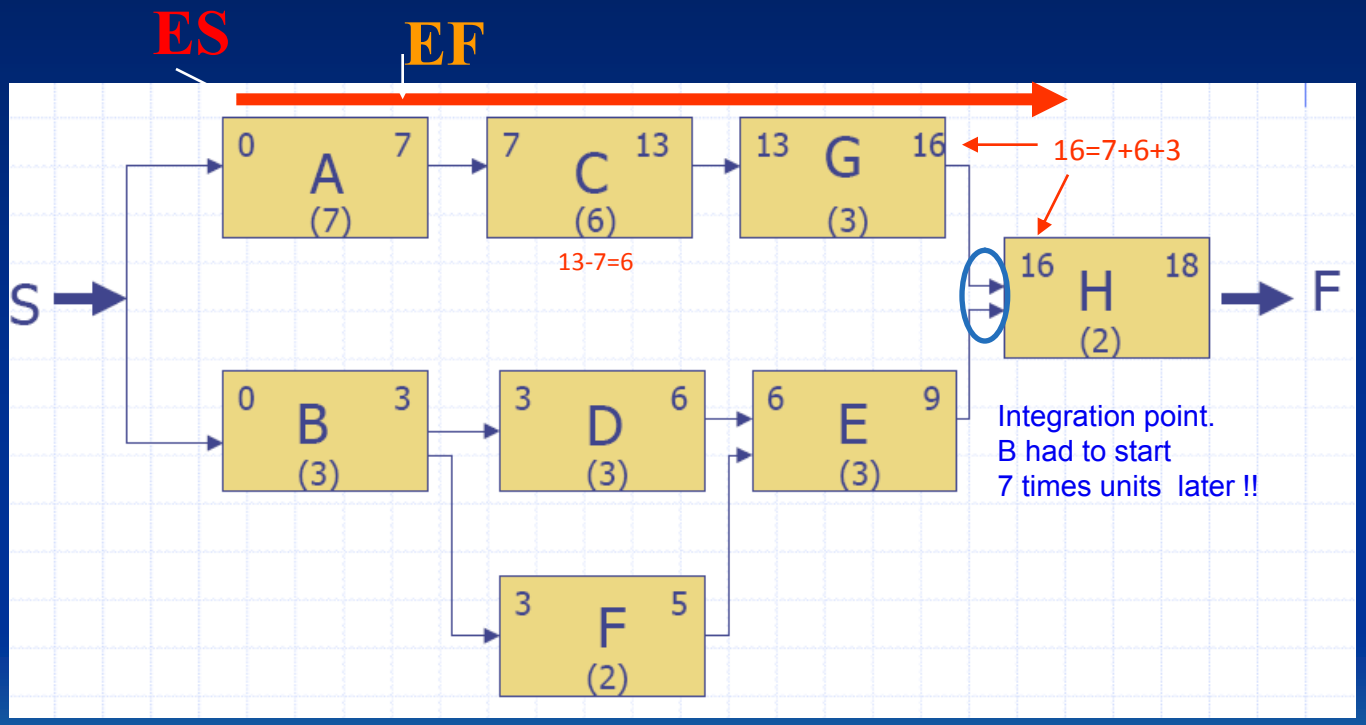
Building a CP diagram 3

Task ID	Duration	Dependency
A	7	
B	3	
C	6	A
D	3	B
E	3	D,F
F	2	B
G	3	C
H	2	E,G

The task in the dependency area influences its successor Task ID

Critical Path (CP)

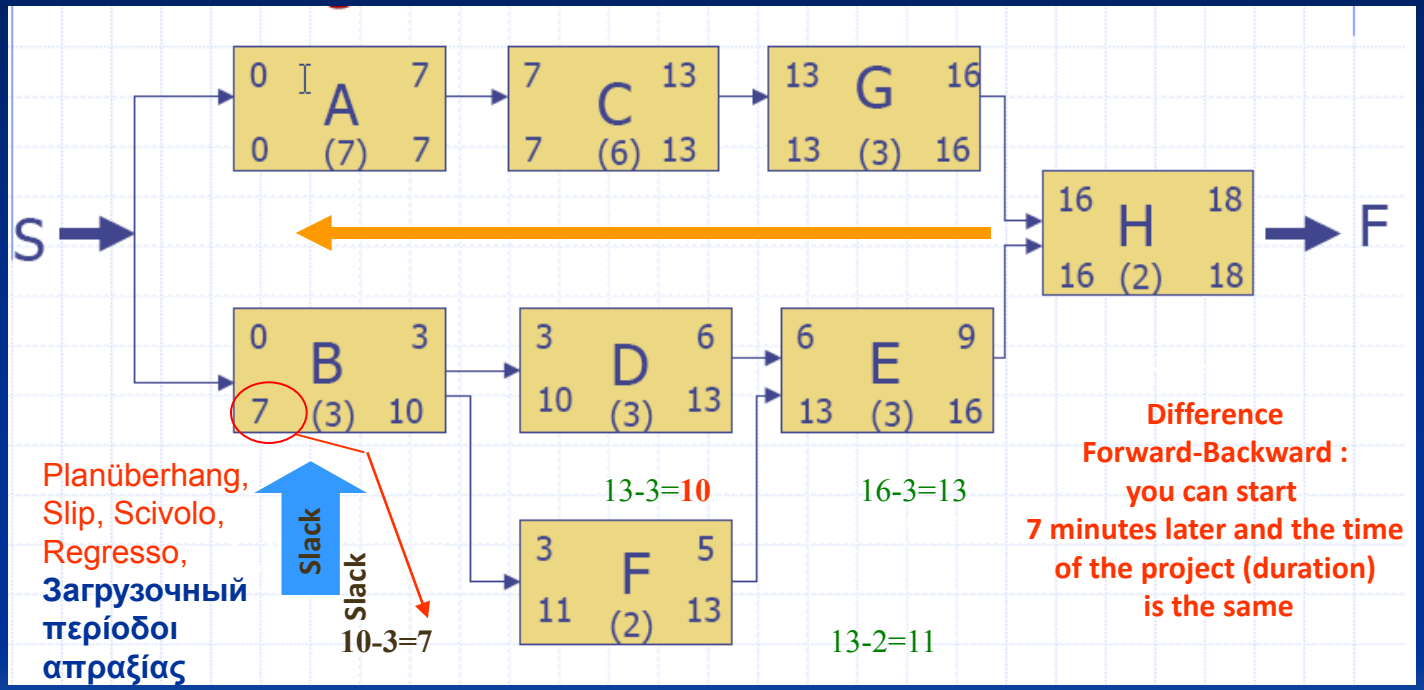
Building a diagram 4 – calculating the **FORWARD PASS**



Early Starts and Early Finishes dates are calculated by means of **Forward Pass**

Critical Path (CP)

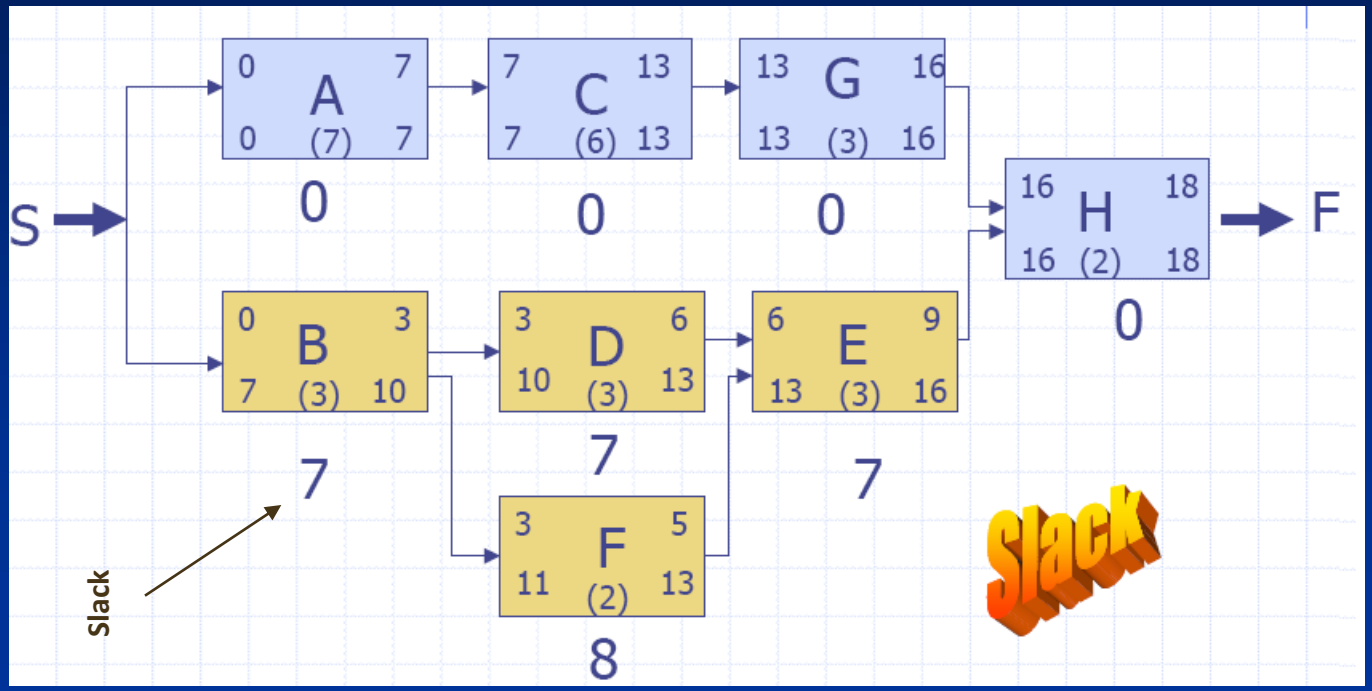
Building a diagram 5 – calculating the **BACKWARD PASS**



Late Starts and Late Finishes dates are calculated by means of **Backward Pass**

Critical Path (CP)

Building a diagram 6 – calculating the **FLOAT(SLACK)/CP**



Free Float (Slack): Amount of time a single task **can be delayed without** delaying the early start of any successor task =LS-ES or LF-EF

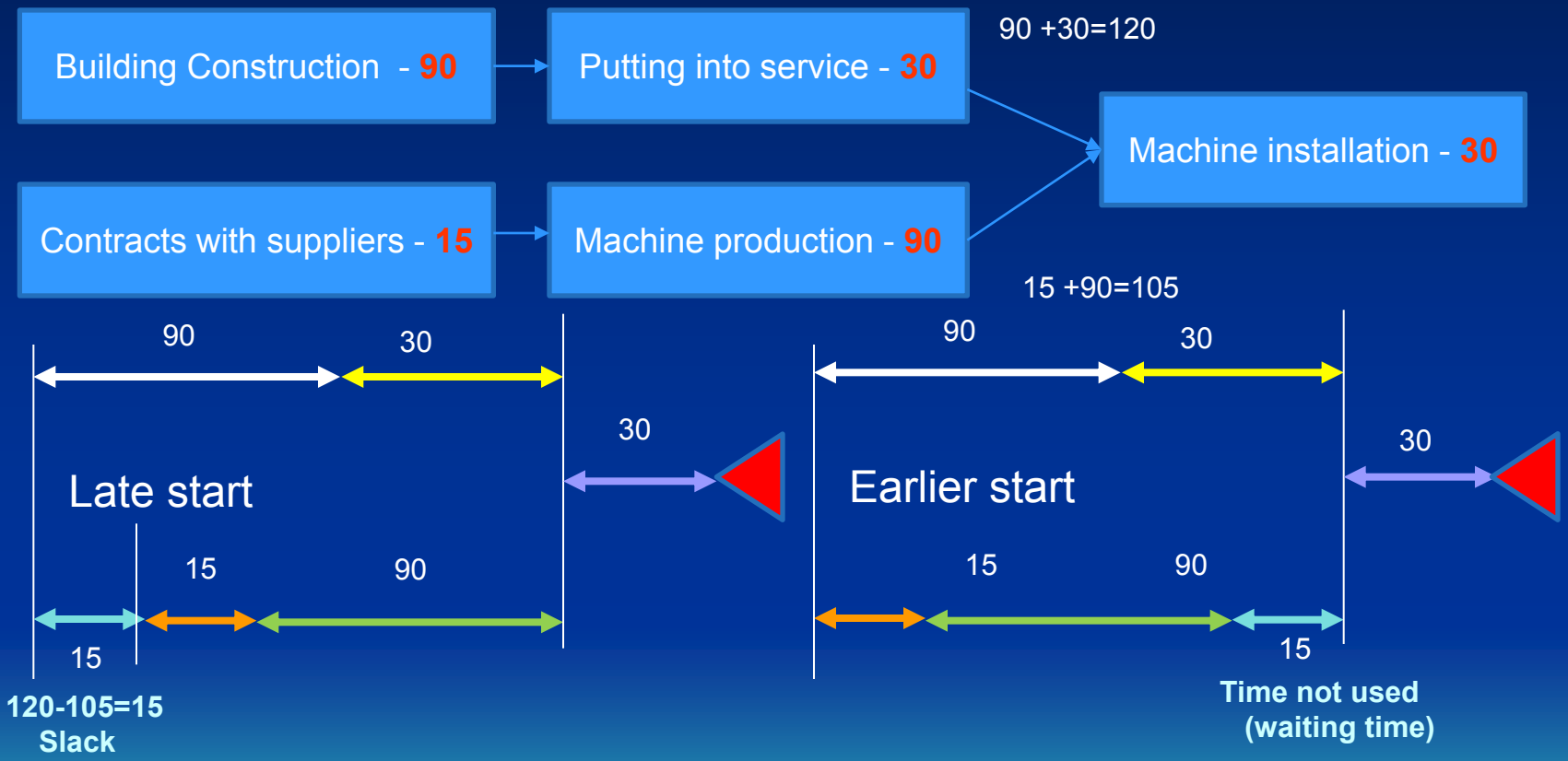
Slack=Float (home study)

Slack, in the context of project management, refers to the amount of time that a task or activity within a **project** **can be delayed without causing a delay to the overall project timeline.** It is also known as **float** or **free float**.

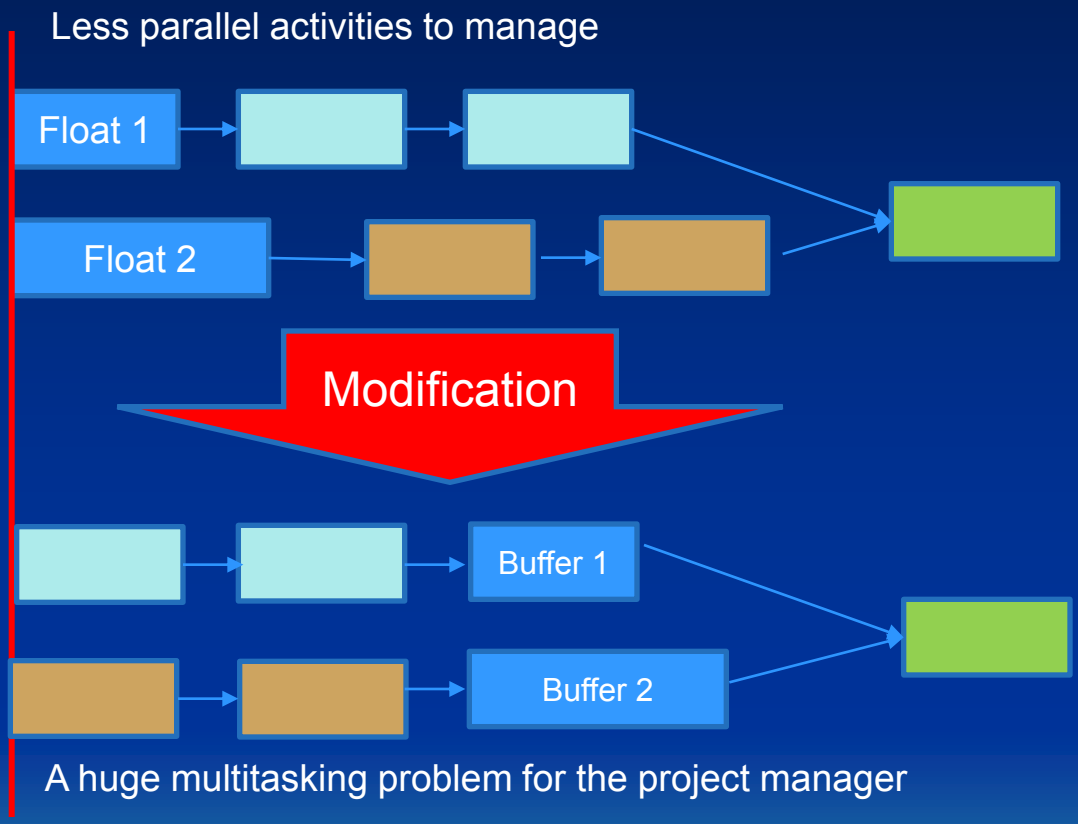
Slack is a critical concept in project scheduling and helps project managers and teams understand the flexibility or free-space they have in managing individual tasks within a project.

Critical Path: The critical path in a project is the sequence of tasks that have **zero total slack**, meaning any delay in these tasks will directly impact the project's completion date. So there is no reserve

Simple project – Construction of the engineering plant



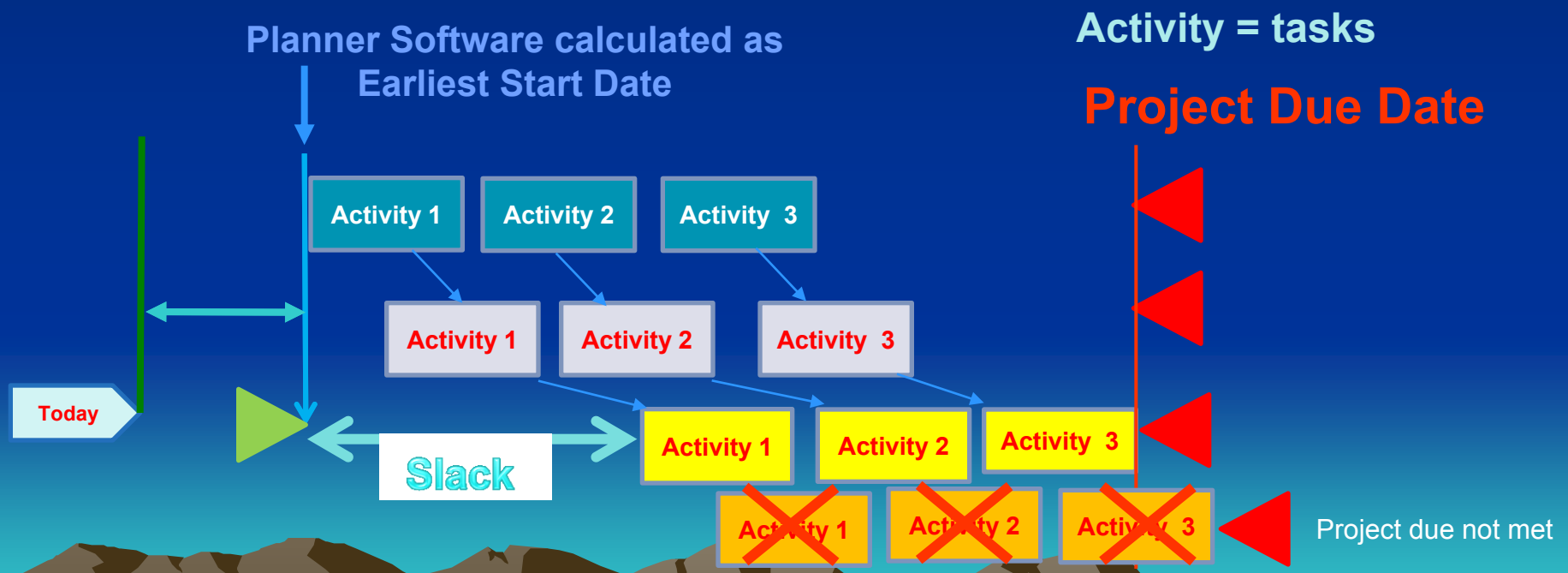
CCPM and Float and Buffers



Today – in order to start ASAP (As Soon As Possible)

Slack=Float (home study)

Accurate description : Slack or **Float** provides flexibility in the project schedule. When used correctly, project managers can shift activities and resources to meet the project objectives and priorities better.
It is the amount of time an activity can be delayed without impacting other activities or the project end date and changes throughout the project implementation.



Critical Path (CP)

CPM is helpful in :

- Project Planning and control.
- **Time-cost trade-offs**.
- Cost-benefit analysis.
- Reducing risk.

Time-Cost Trade-off

The term "trade-off" in project management traditionally refers to the decision to create and maintain a balance between the "time" and "cost" of a project.

The duration of a project can often be reduced by speeding up some of its activities at the cost of additional (higher) resource costs.
There is a relationship between the time to complete a project and its cost.

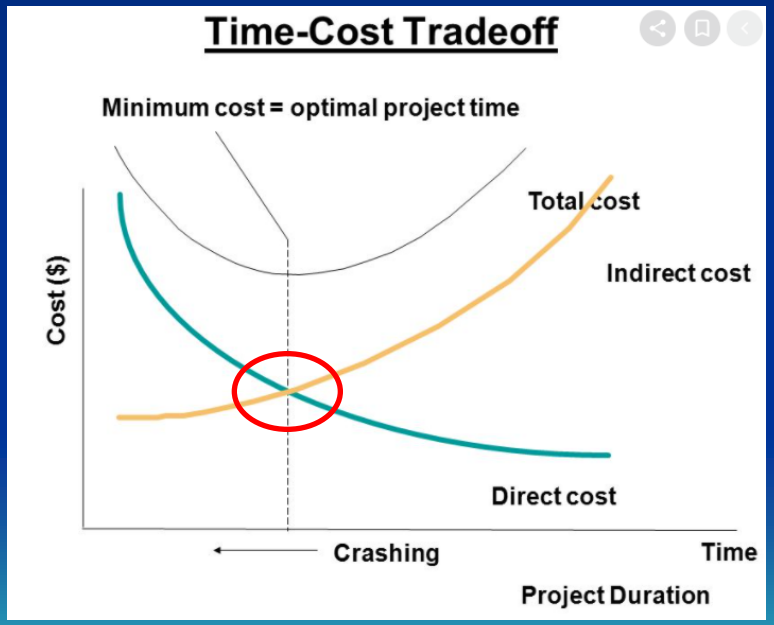
For some types of cost, the time-cost relationship is in direct proportion.

For other types, there is a direct trade-off. The existence of these two types of costs allows setting the optimal pace of the project and thus reducing costs.

Project crashing – reducing project implementation time

Project crashing ： you can achieve by reducing the time of one or more tasks (activities).

However, this is associated with the allocation of additional resources to tasks
It may result in shorter task times than originally planned, but,
of course, it also increases the cost of the whole project.



BUT

Brooks's Law is a principle in software project management that states:
"Adding manpower to a late software project makes it later."

Brook s law

KEY POINTS

- Rump-up Time ：New team members need time to become productive
- Communication Overhead ： number of coomunication channeslk increases
- Limited Divisibility : Some tasks cannot be easily devided among many people

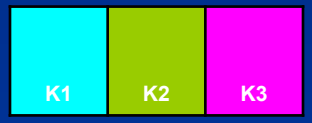
Critical Path (CP)

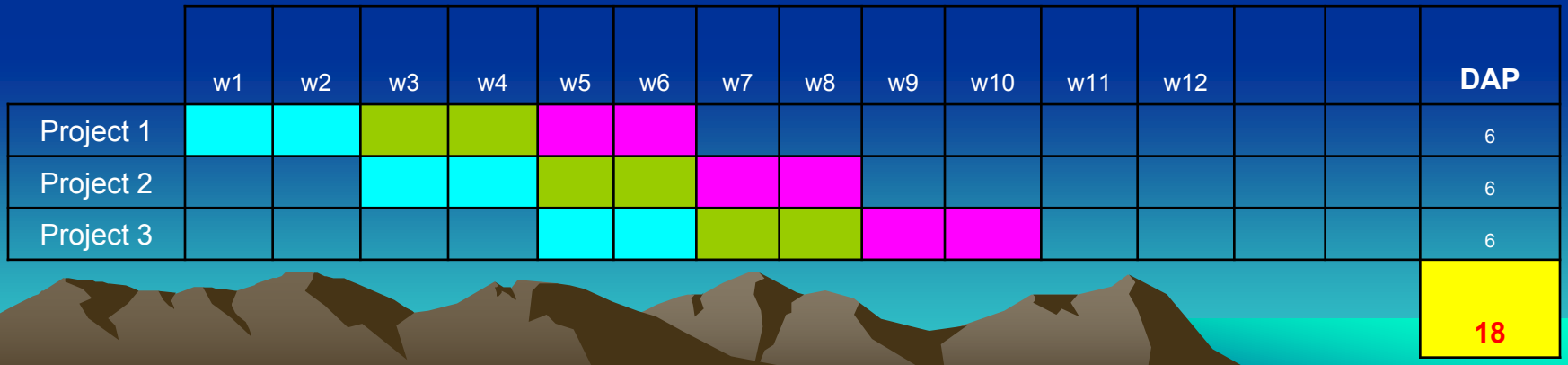
Limitation of CPM :

- Does not consider resource capacities (very bad approach).
- Less efficient use of buffer time.
- Less focus on non critical tasks that can cause risk.
- Based on only deterministic task duration.
- Critical Path can change during execution.

Multi-project Management

Bad multitasking causes, that one project will be significantly longer and no other project will be shorter





Multitasking characterization

- people always **overestimate the length of their tasks**
- salesman offers impracticable terms (dates)
- The fight for reserves (capacities) causes, that all saved time is fully wasted (Student s syndrome)
- Reserves (if any) are used badly !!!!!!
- Bad use of reserves causes lack of transparent assignment
- Non transparent priorities are parents of bad multitasking
- Bad multitasking causes longer duration of all activities (tasks) and thus all the projects

CP definition (more in detail)

Critical path is defined as the longest way (meaning time) from the starting point of the project graph to the ending point

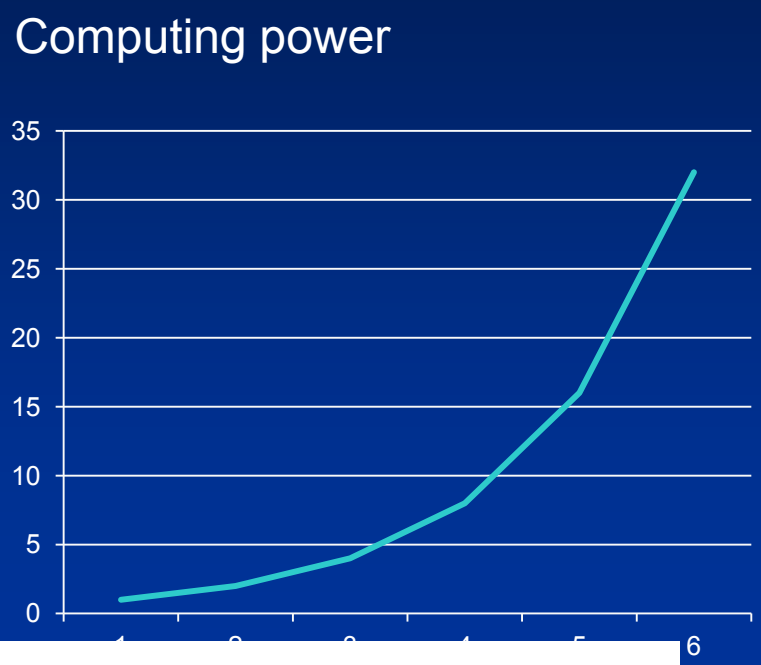
Critical path represents technological dependencies and given times of every task on Critical path inclusive of necessary condition for fulfilment of foregoing tasks (activities) framed by integration points.

Critical Chain

~~Task~~

Resource and capacities

Contemplation I.

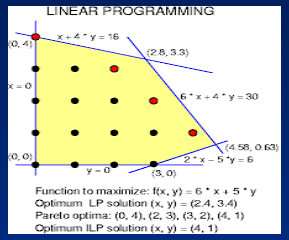


Result ->Stress

Contemplation II.

+

+



E-mails Parallel telephoning Parallel problem solving

a) Am I a multitasker ?

b) NO !! And my IQ went down 15 point due to parallel processing!

Contemplation III.

+

+

Is this the goal of my lifelong efforts?

Maybe not …. I guess
I reached another peak …

Critical chain definition

In TOC the Critical chain is defined as the longest way (meaning time) from the starting point of the project graph (Gantt) to the ending point, which takes into account **technological dependencies** as well as time of the tasks and moreover, **capacities of assigned resources**.

With infinite capacities of resources you can consider **Critical path**=**Critical chain**

Critical Chain

- **Critical Chain** is a project management methodology and scheduling approach developed by Dr. Eliyahu M. Goldratt, known for his work in the Theory of Constraints (TOC).
- **Critical Chain** project management focuses on improving the efficiency and reliability of project execution by addressing common project management challenges.
- The primary components of the Critical Chain include:

See next slides

Project schedule

- **Project Schedule:** In Critical Chain, a project schedule is created by identifying the critical path, the most extended sequence of dependent tasks and activities that determines the project's overall duration.
- Unlike traditional project management, the Critical Chain takes consideres the availability of resources and buffers for the **uncertainty approach**

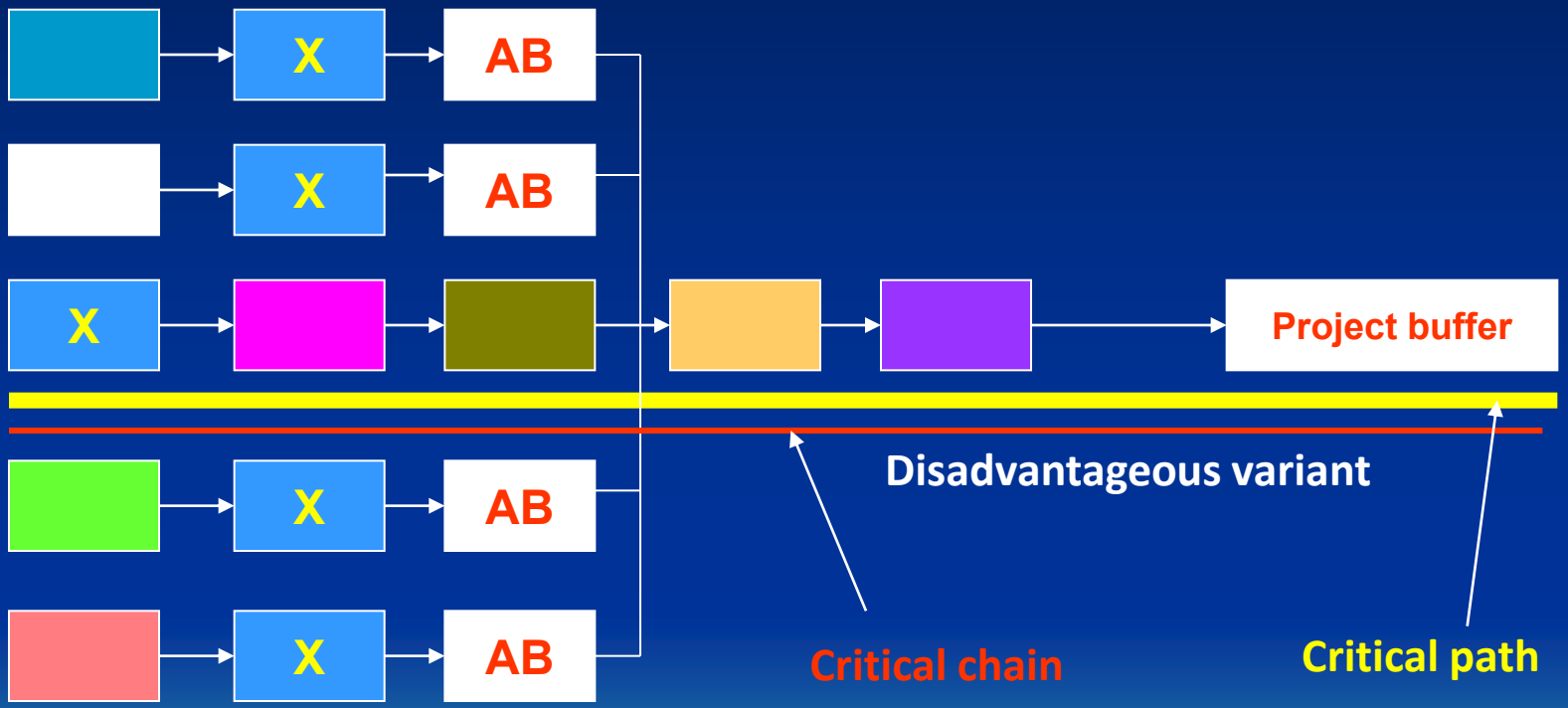
Resource Management

- **Resource Management:** Critical Chain emphasizes the need to effectively manage project resources (human, equipment, materials).
- It suggests that project managers should protect the project schedule by **not overloading resources** and **ensuring that resources are available.**

Buffers

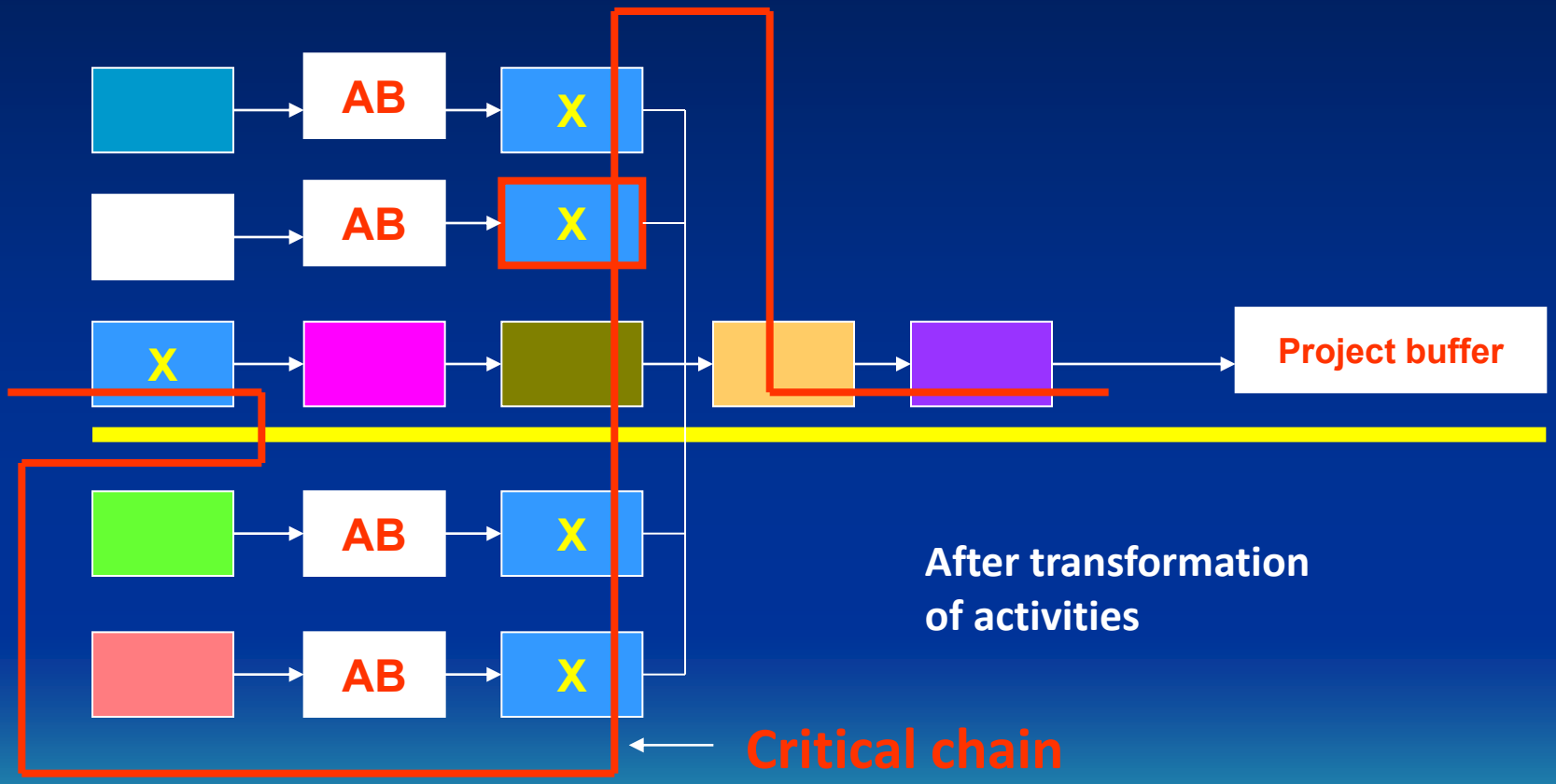
- **Buffers:** Critical Chain introduces the concept of project buffers to manage uncertainties and variability in project execution. There are typically three types of buffers:
- **Project Buffer:** This is placed at the end of the project schedule to protect the project's overall duration. It accounts for uncertainties in task durations and resource availability.
- **Feeding Buffer:** Placed before non-critical tasks. Such a buffer protects the critical path by ensuring, that tasks on the critical path have the necessary resources to proceed.
- **Resource Buffer:** Placed before critical resources, this buffer protects against resource constraints that could impact the project schedule.

Multi-project management and critical resources (CCR) used in more that one project branch



CCR = Capacity Constrained Resource = X

Multi-project management and critical resources (CCR) used in more that one project branch



Project management based on remainimg time in buffers – **Buffer Management**

- Buffers are used for timely warning and that is to say predicting and avoiding future problems related to project deadlines (milestones are not met)

- It is also used as a guideline for corrective actions
- Buffers represent time reserves

Basic metrics showing the project status

- The partial size of Critical chain (CC) fulfilled in days (in %)
- How much of buffer size was used to fulfil above mentioned partial size of Critical Chain?
- Trend of project (buffer consumption graph- see next slide)
- Consumption of the financial buffer –relation to budget
- Priorities – bigger buffer penetration- bigger priority
- Adjoining branches have always lower priorities
- It is not allowed to create bad multitasking

Trends of the project

% use of the buffer

Dangerous zone

Warning zone

Safety zone

Guardian angel

% consumption of the critical chain

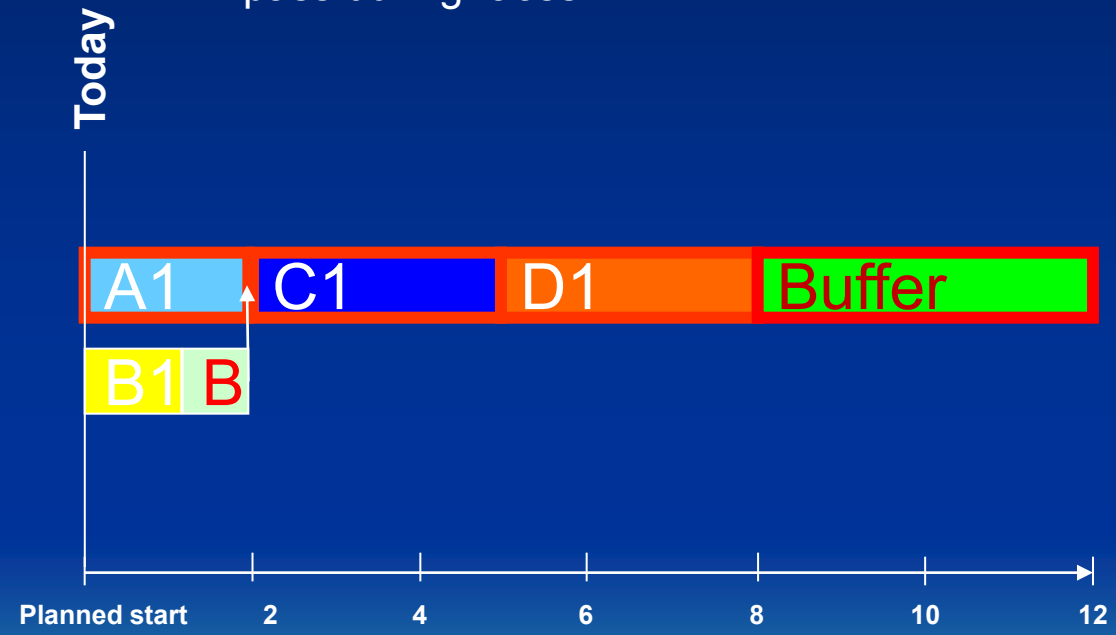
Trend of the project advancement – (another angle of view)

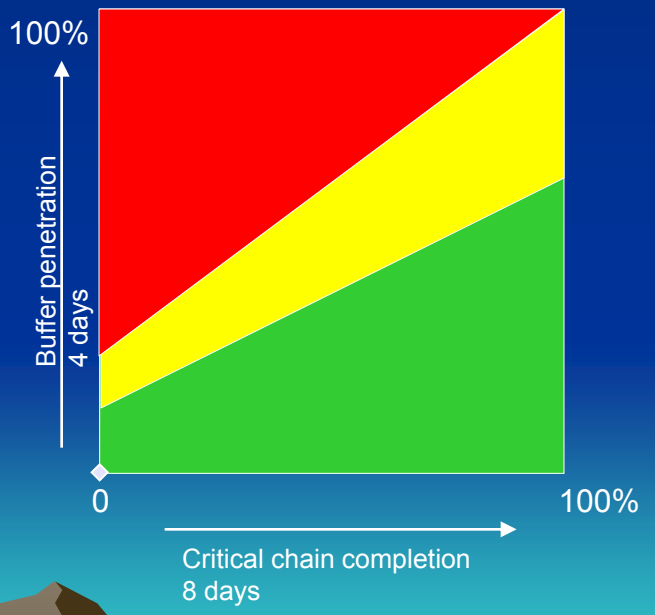
Resource: DP R.Jurka (2006); taken from LEACH, L., P. (2004), s. 12.

Planning - principles

We are working with plan , which takes into account different times of tasks :

- start of the tasks are changed based on termination of preceding tasks
- you have to react in project in such a way , that handover is done as a baton pass during races



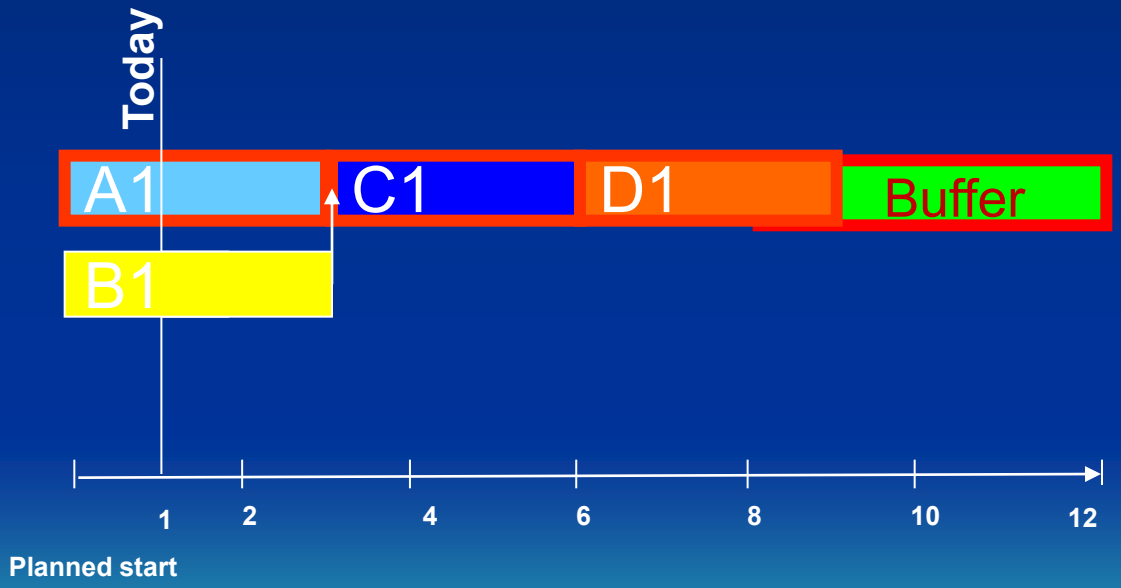


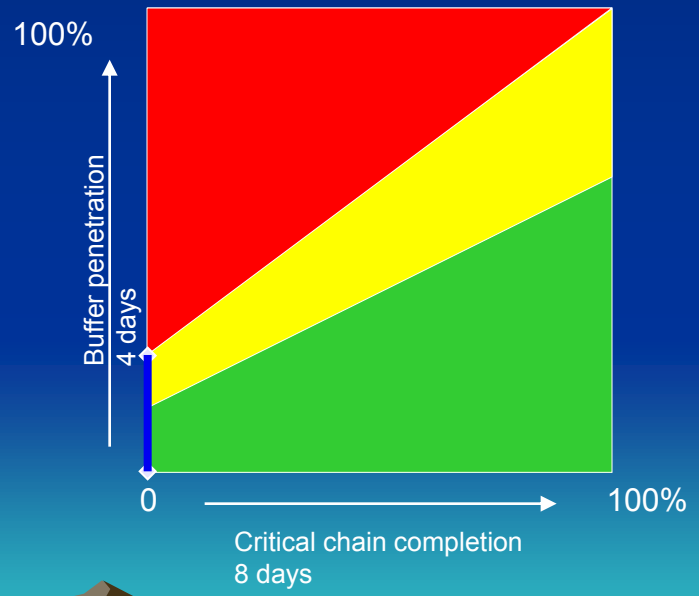
Plan with sharp deadlines with buffers 50% (2+3+3=8 8+4=12)

Planning - principles

A1 did not started yet , because this A1 resource is still working on another order (task), which may be part of another project

B1 already started an for completion will need another two days



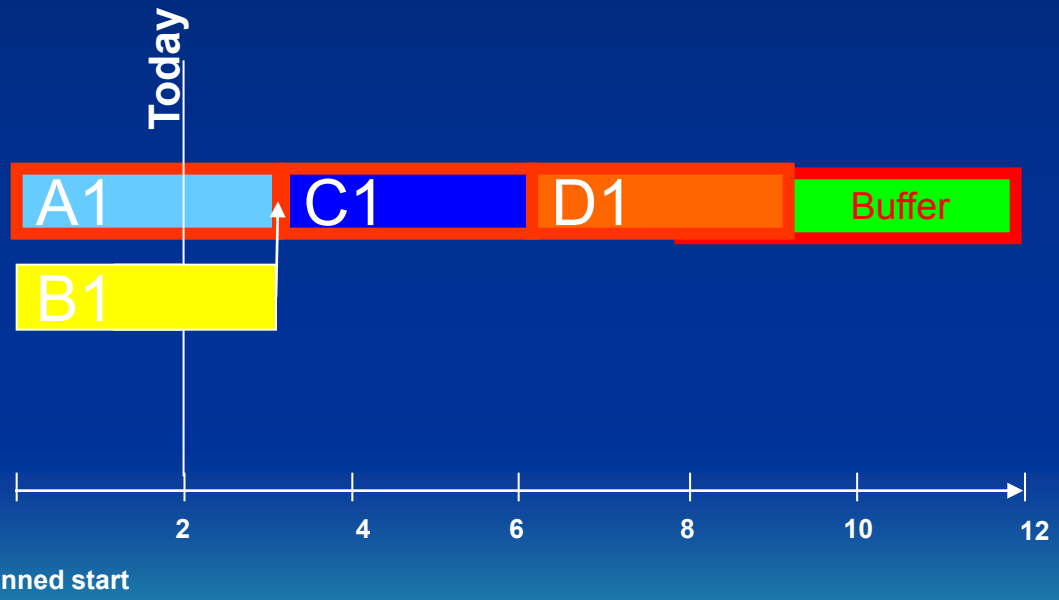


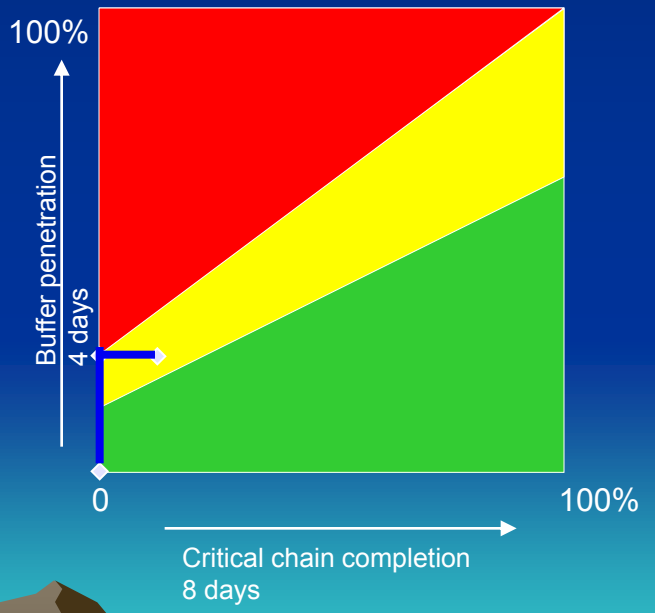
Plan with sharp deadlines with buffers 50% (2+3+3=8 8+4=12)

Plan 2nd day after start

A1 started and will be finished (completed) tomorrow.

B1 will be finished (completed) tomorrow

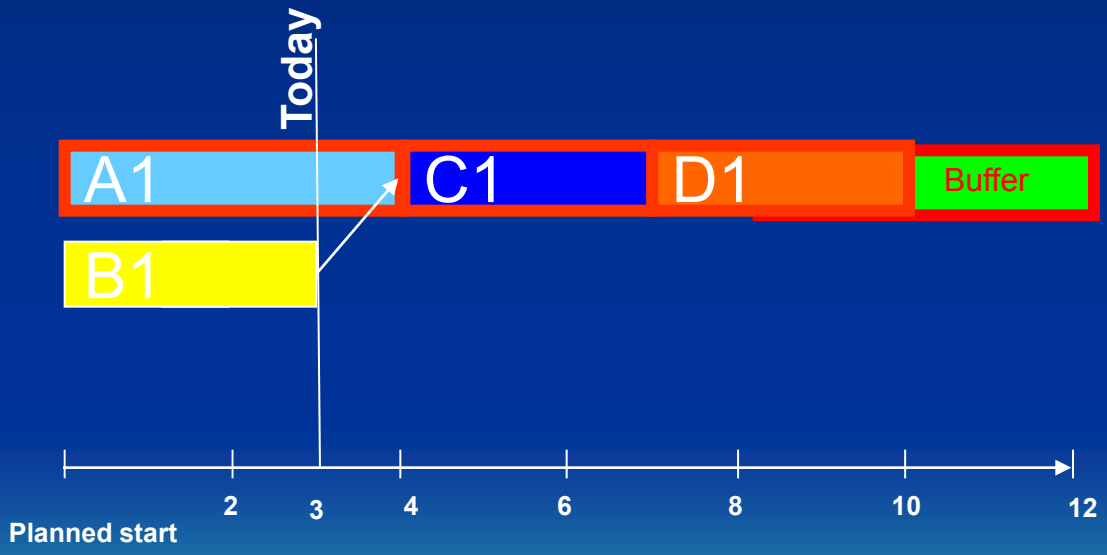


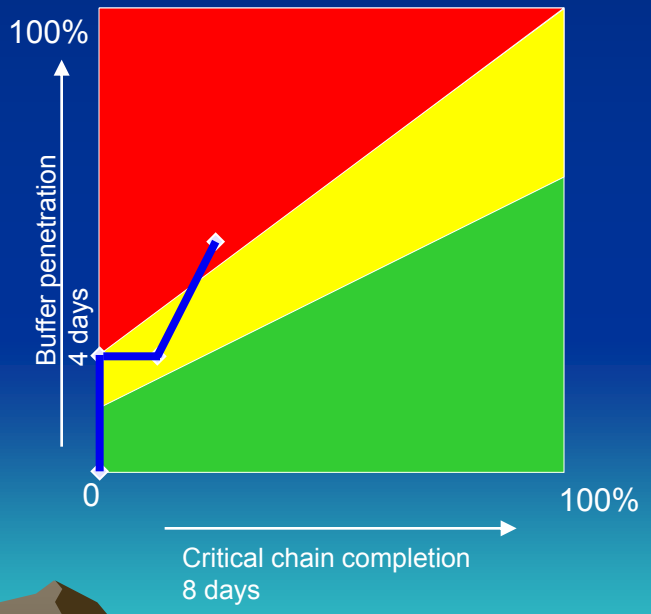


Plan with sharp deadlines with buffers 50% (2+3+3=8 8+4=12)

Plan 3rd day after start

A1 despite all efforts resource A1 needs another day to complete.
B1 has completed his work with 2 days delay

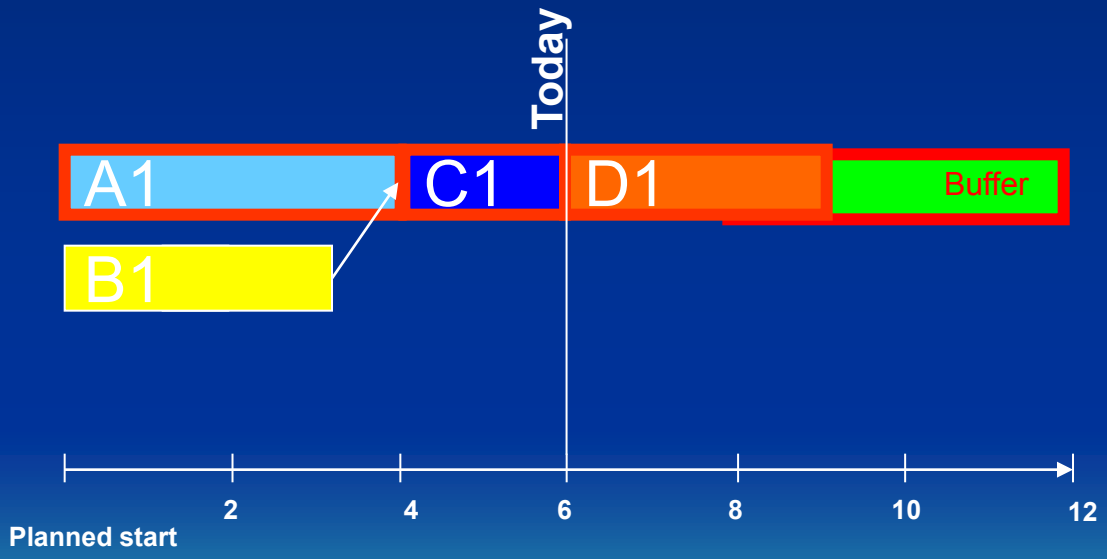


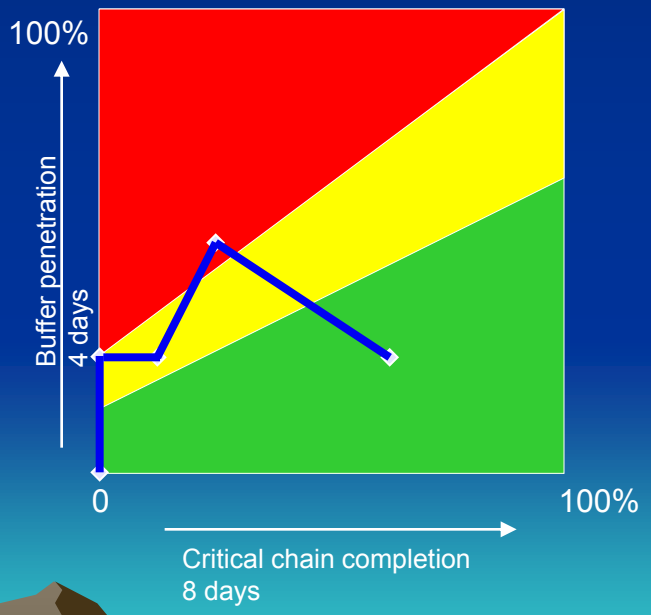


Plan with sharp deadlines with buffers 50% (2+3+3=8 8+4=12)

Plan 6 day after start

A1 completed his task with 2 days delay
B1 completed his task with 2 days delay
C1 completed his task 1 day earlier than expected (planned)
D1 will start to work tomorrow

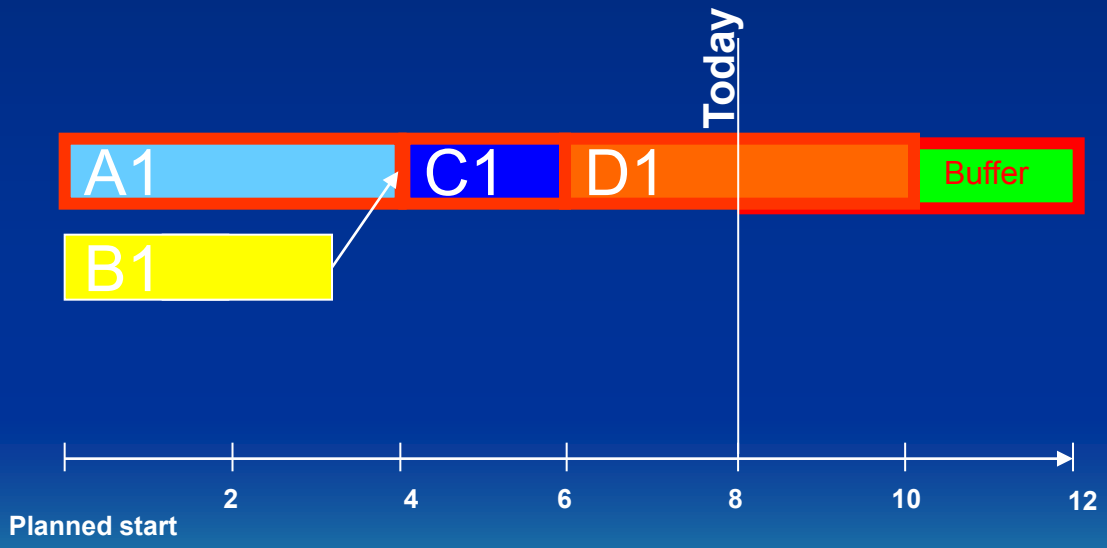


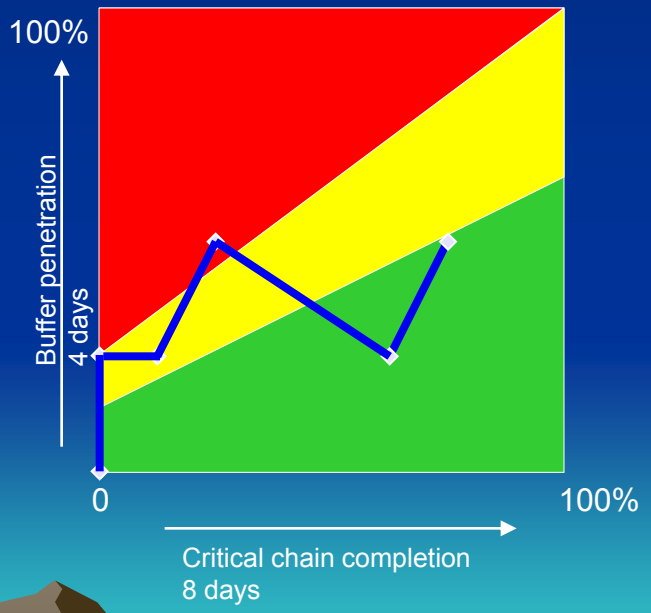


Plan with sharp deadlines with buffers 50% (2+3+3=8 8+4=12)

Plan 8 day after start

A1 completed his task with 2 days delay
B1 completed his task with 2 days delay
C1 completed his task 1 day earlier than expected (planned)
D1 needs one day more to complete

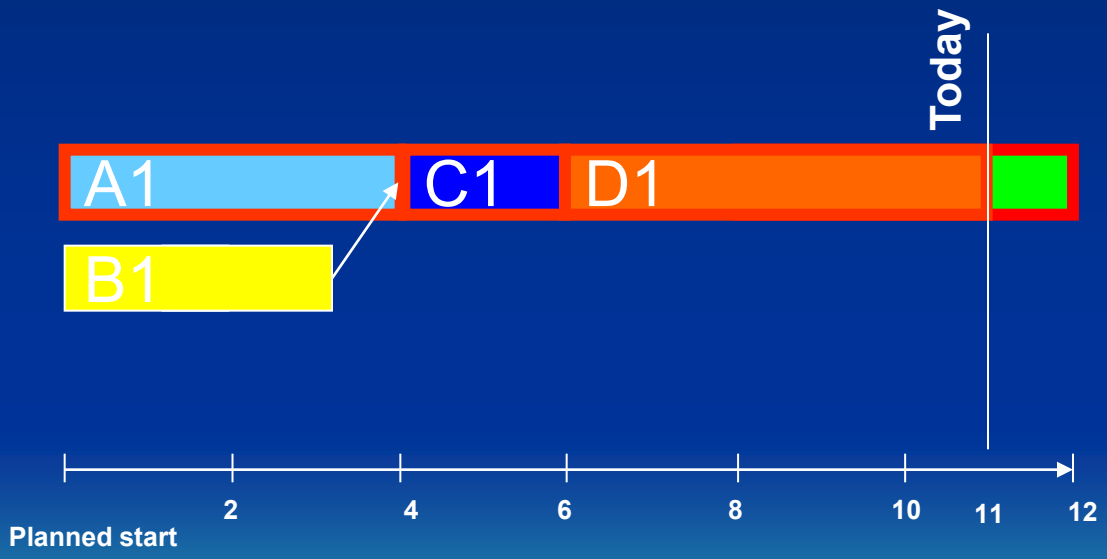


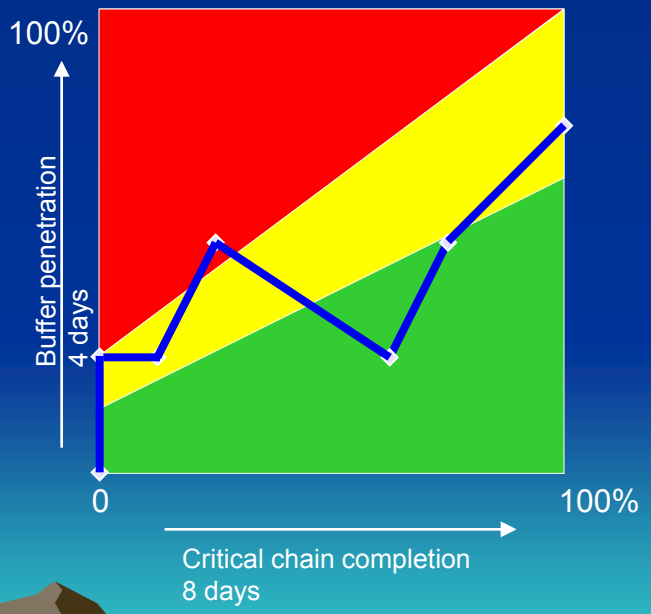


Plan with sharp deadlines with buffers 50% (2+3+3=8 8+4=12)

Plan 11 day after start

A1 completed his task with 2 days delay
B1 completed his task with 2 days delay
C1 completed his task 1 day earlier than expected (planned)
D1 completed his task with 2 days delay



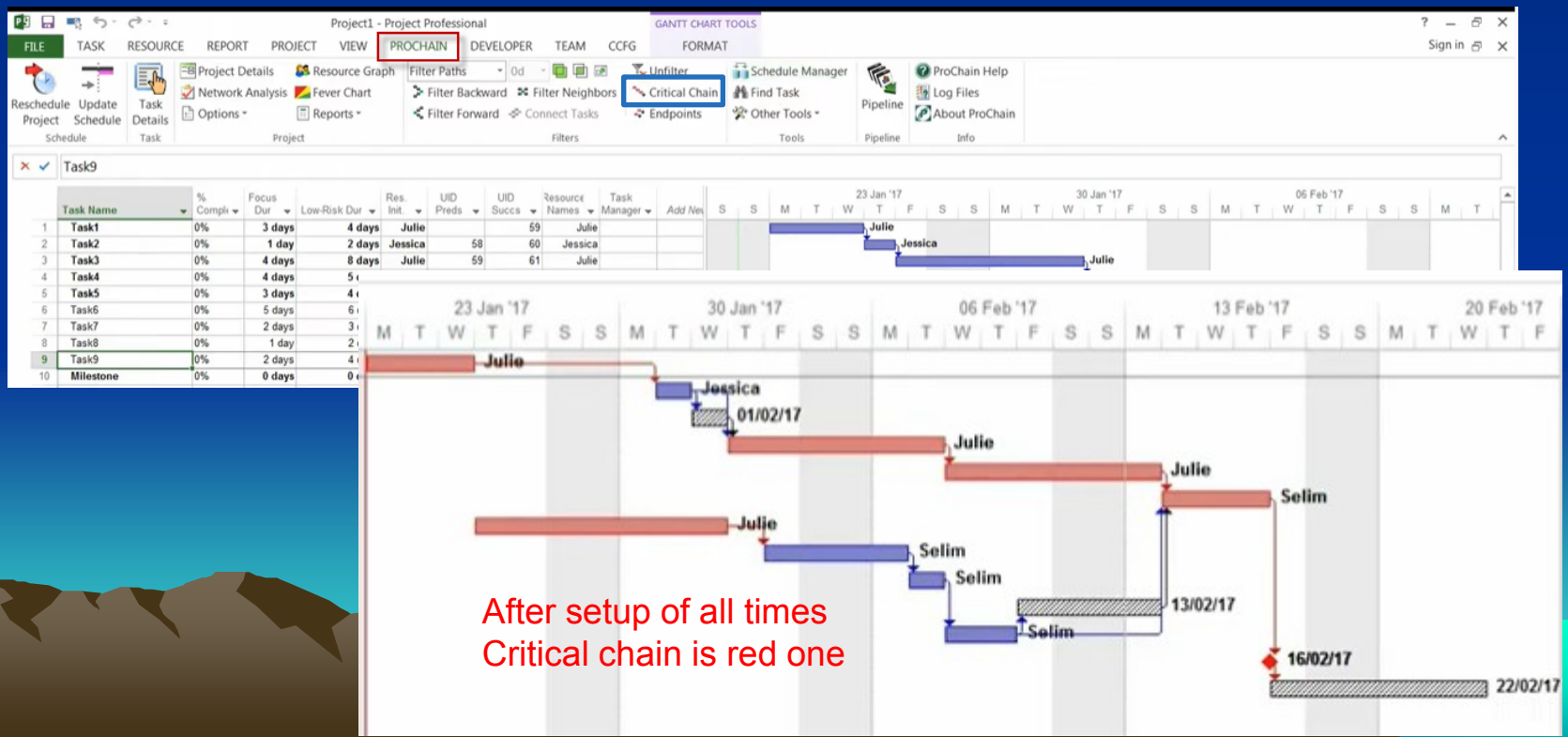


Plan with sharp deadlines with buffers 50% (2+3+3=8 8+4=12)

Example of using real SW package to control project by CCPM methodology (home study)

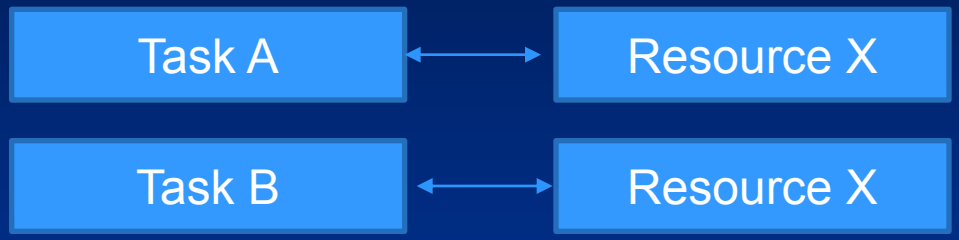
https://www.youtube.com/watch?v=4ARI1qIG1vA

Use of Add-on Prochain to MS Project Application

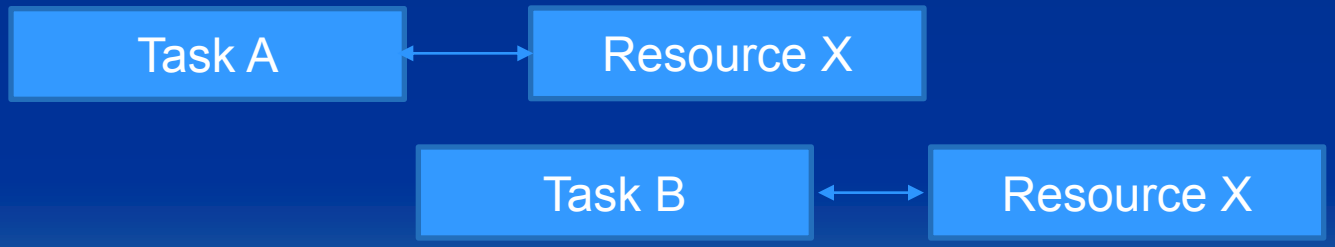


Levelling (balancing)

Before - > tasks and assigned (scheduled) resources

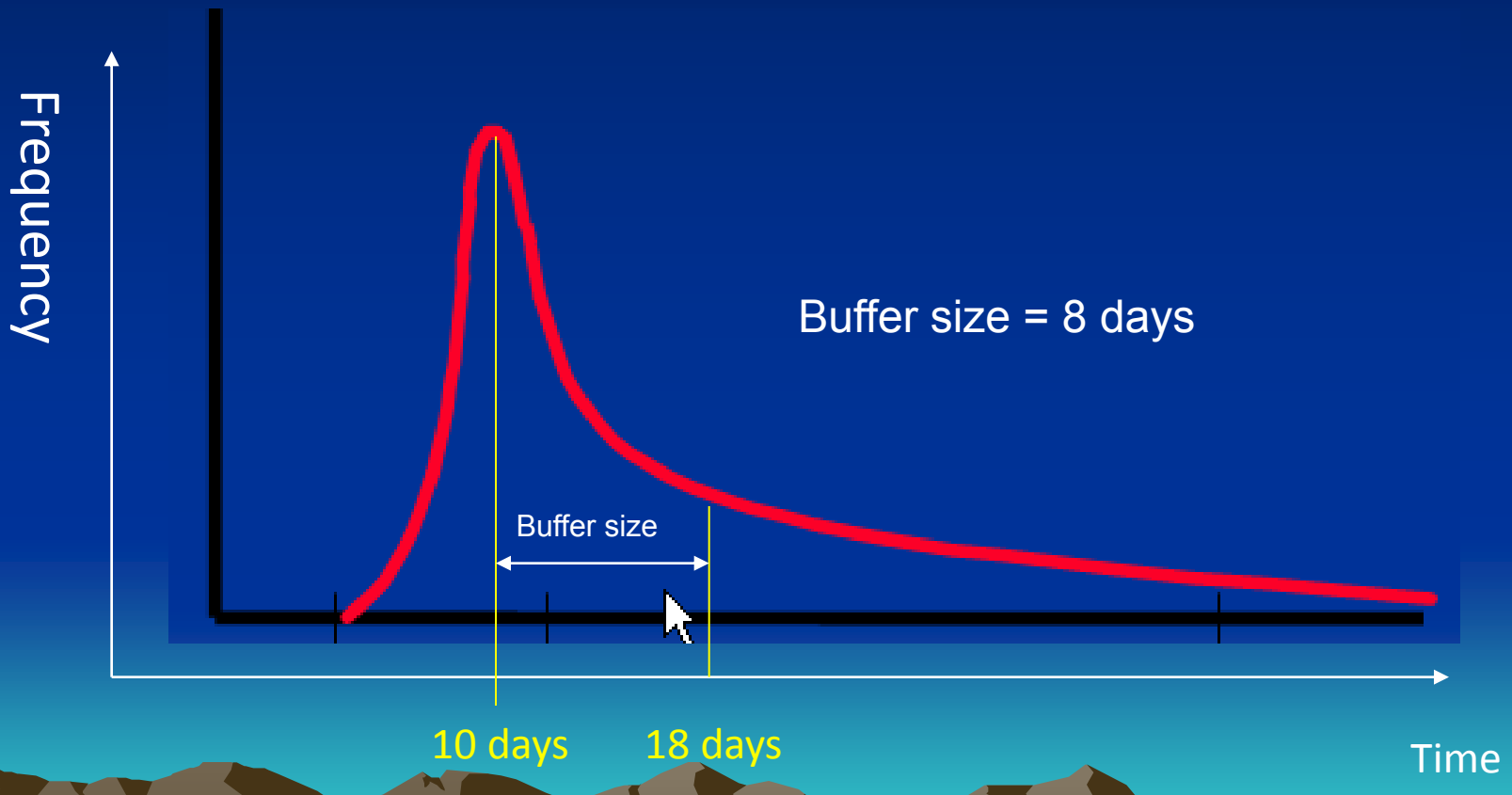


After



A resource cannot perform two actions at the same time !!!

Time distribution

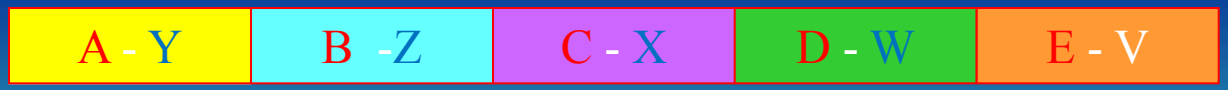


Project Quick, resources A-E and activities X,Z,X,W, and V

Resource and Activity	Median of the required time
A - Y	10 days

Activity = Task

You can say, that 50 % of any activities finish earlier, and other **50 %** will be delayed, meaning, that **10 days** represents **50 %** of the estimated time for chosen aktivity (task).
Project managers decided, that activity ends if **90 %** of estimated time will be consumed. It means, that they add for the safety reasons a time buffer of **8 days**
10 d= 50%, 20d=100%, 2d=10%, 20d-2d=18d=90%, 18d-10d=**8days**



(90 %)

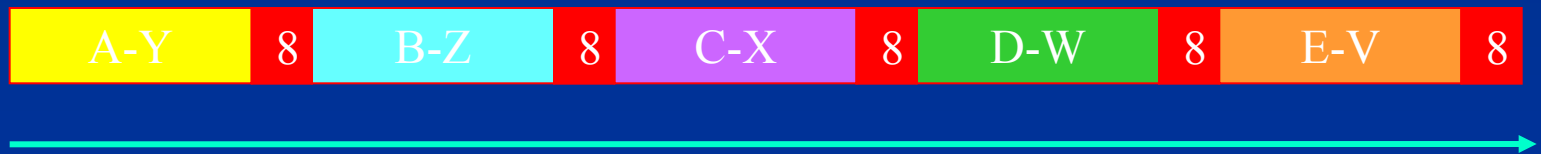
5 x 10 days=50 days (50 %)

5 x 18 days= 5 x (10+**8**)=90 days

Five activities (tasks) and applied modifications

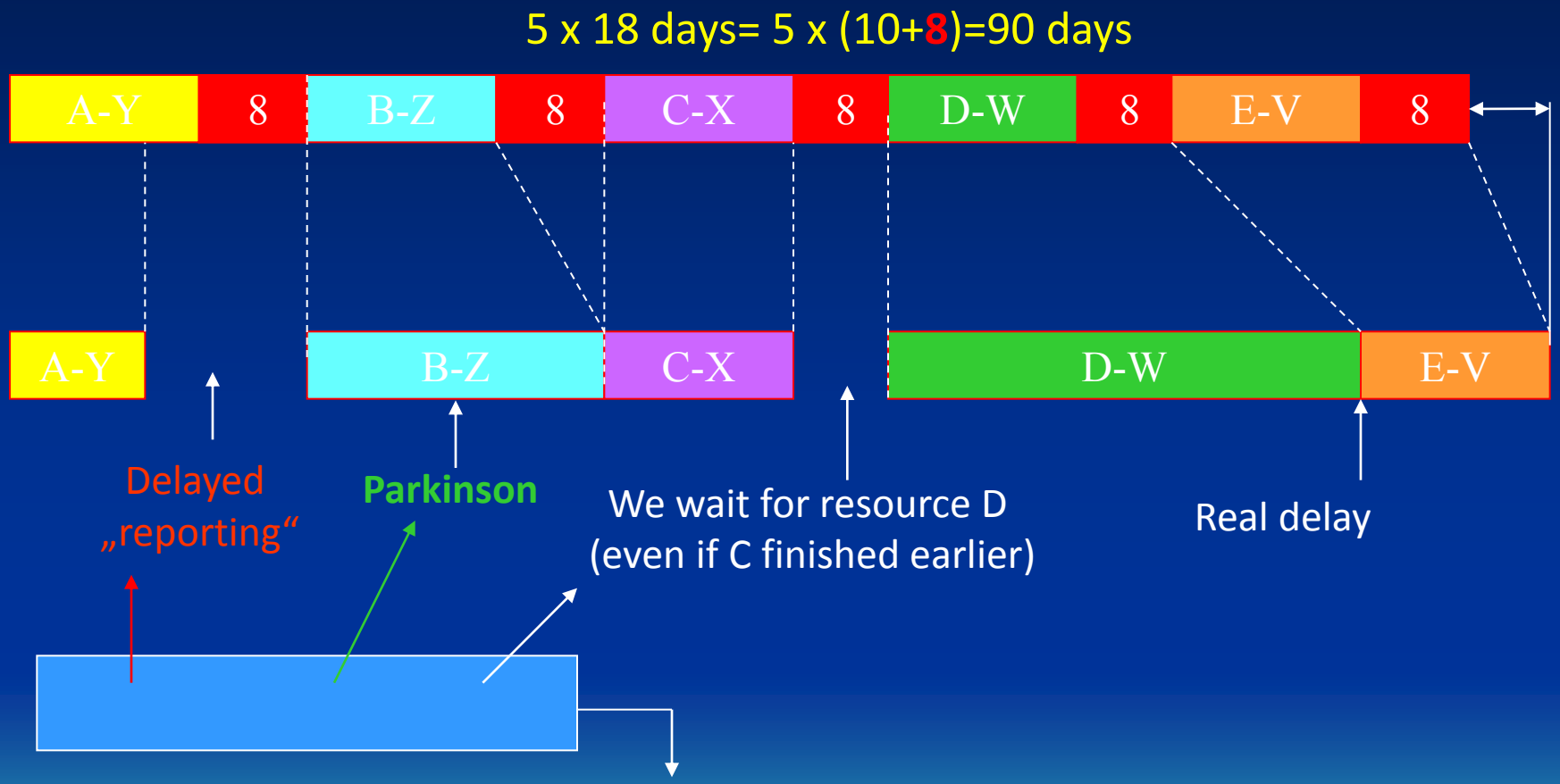
- If we consider for every activity time buffer 8 days we will get :

5 x 10 days= 50 days



5 x 18 days= 5 x (10+8)=90 days

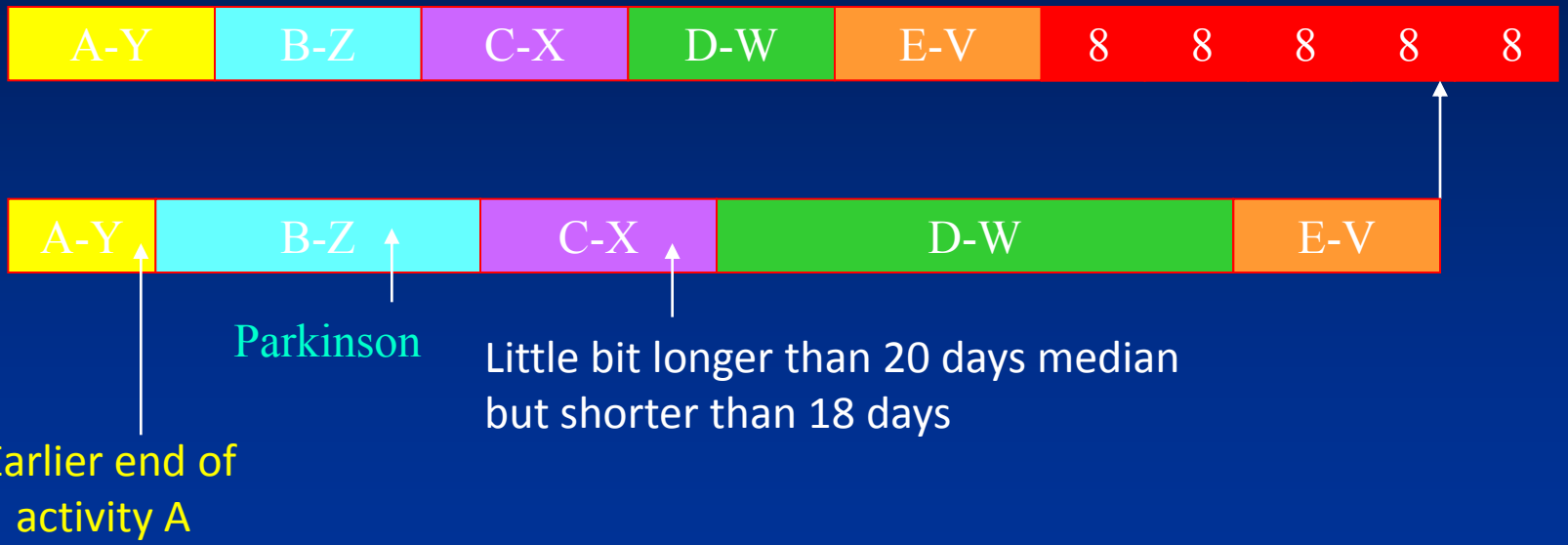
Five activities and modifications (added buffers) and four types of troubles



No one trouble causes project delay taking into consideration planned delivery date (agreed date of the project).

Dissipation of acquired time reserves was caused by company strategy saying strictly stick to the planned project schedule (example of rigid management)

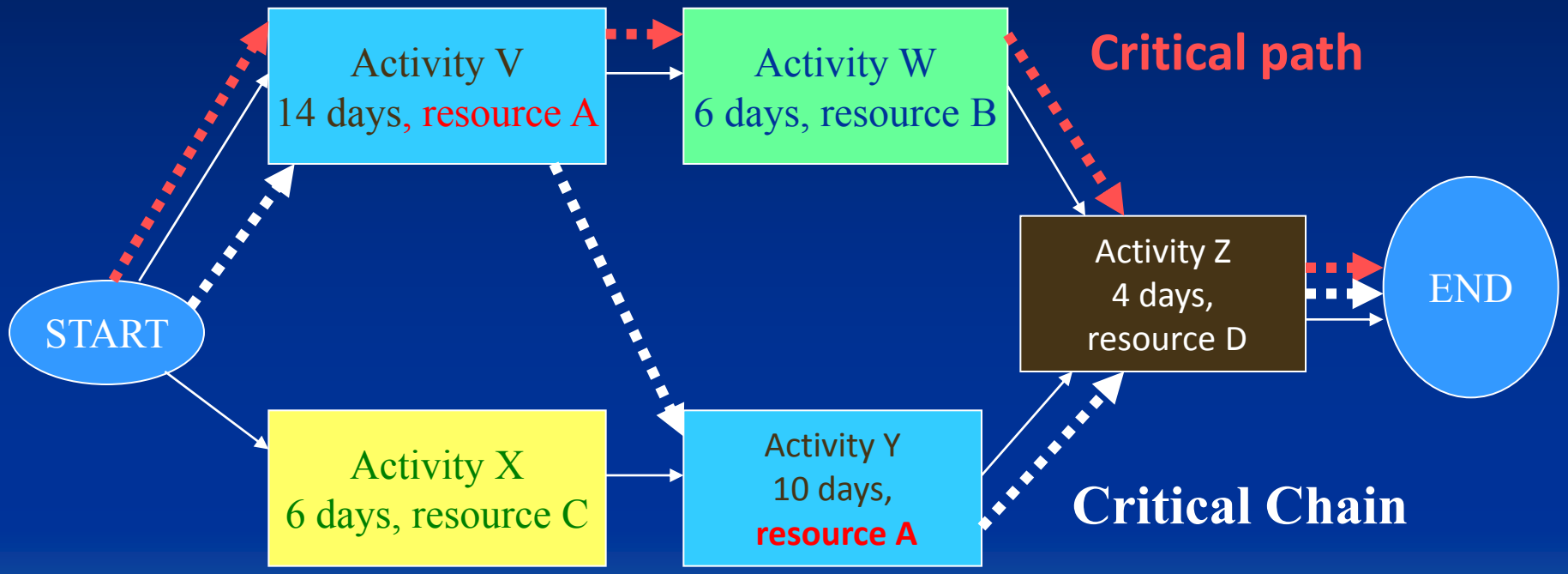
Five projects after modification (buffers united to one and placed to the end of the project)



= CPB=current project buffer = 40 days

= new buffer = 50 % out of CPB, which makes CPB/2

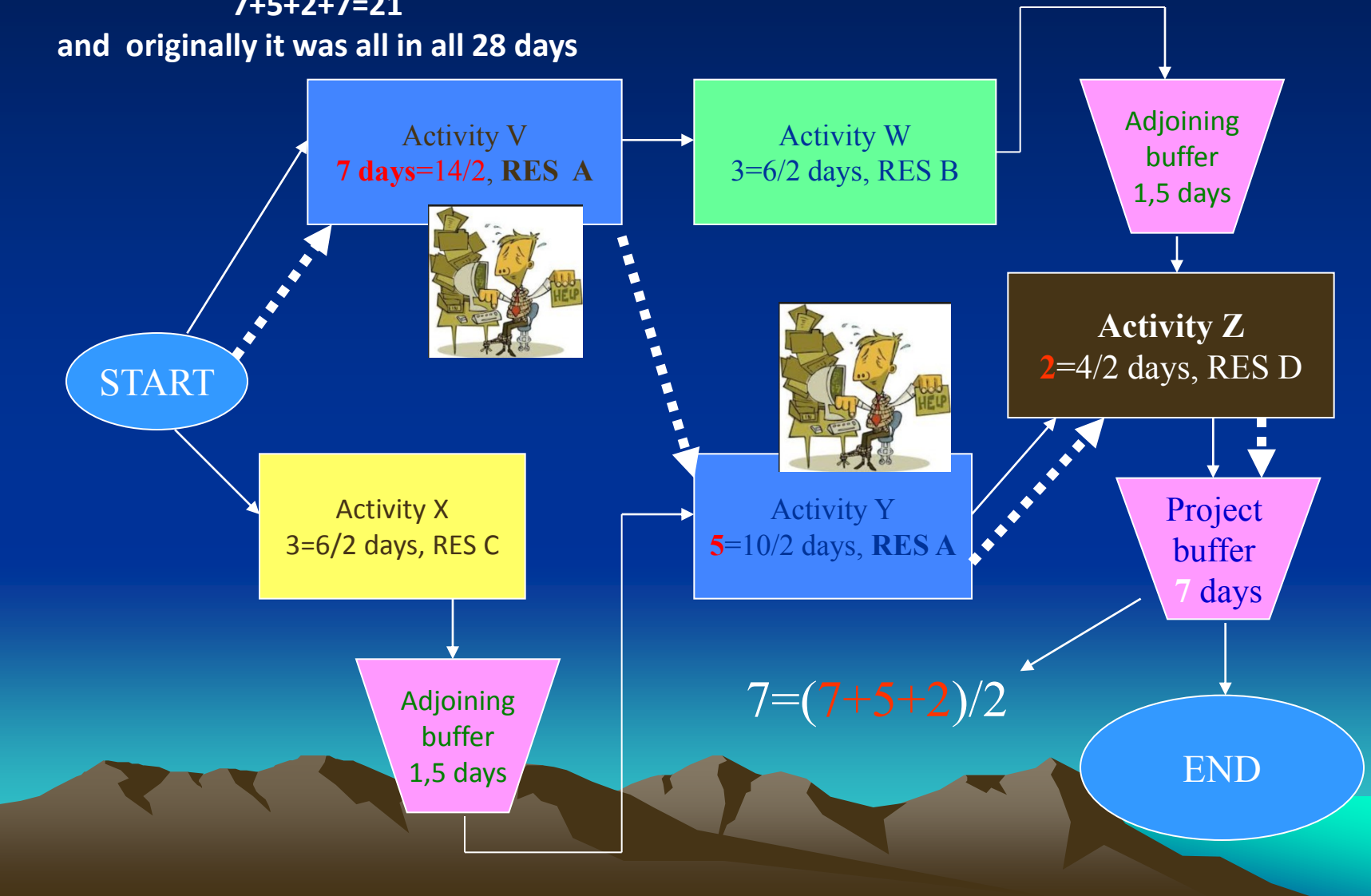
Critical path- Critical chain



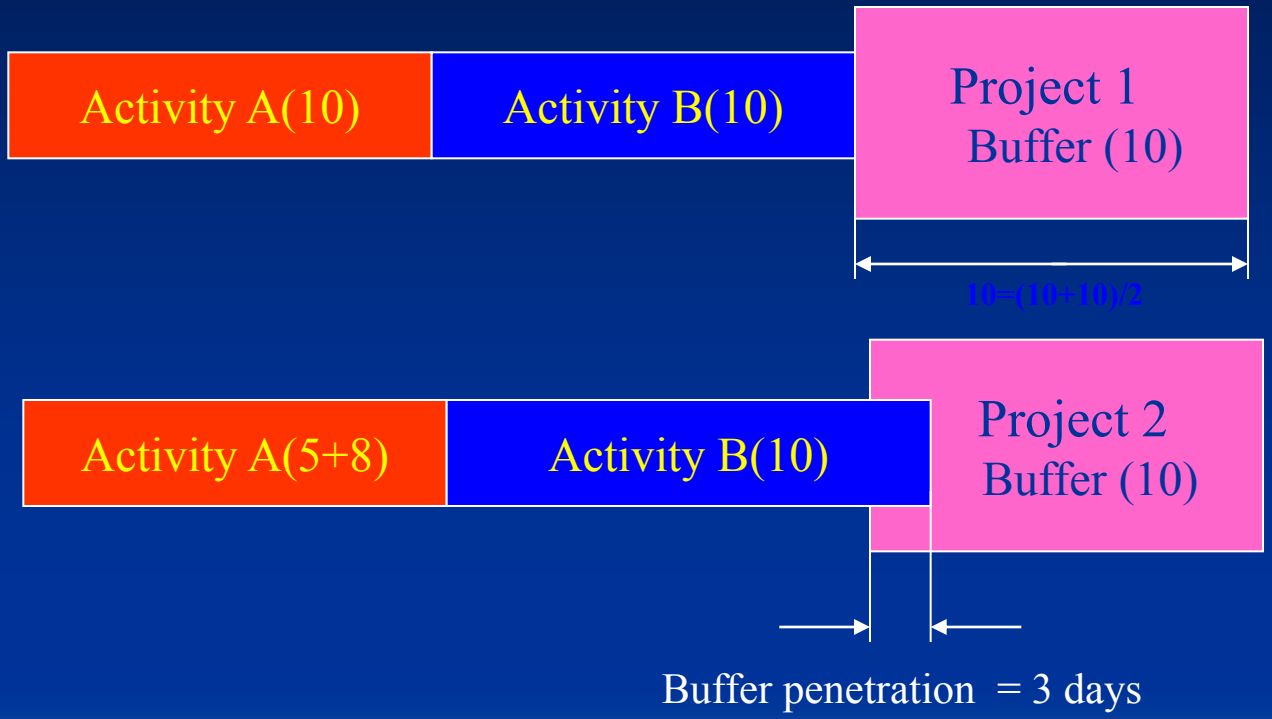
Project is considered as successful if is finished in expected time and financial budget is not exceeded

Critical chain with buffers

Length of the Critical Chain (white line):
7+5+2+7=21
and originally it was all in all 28 days



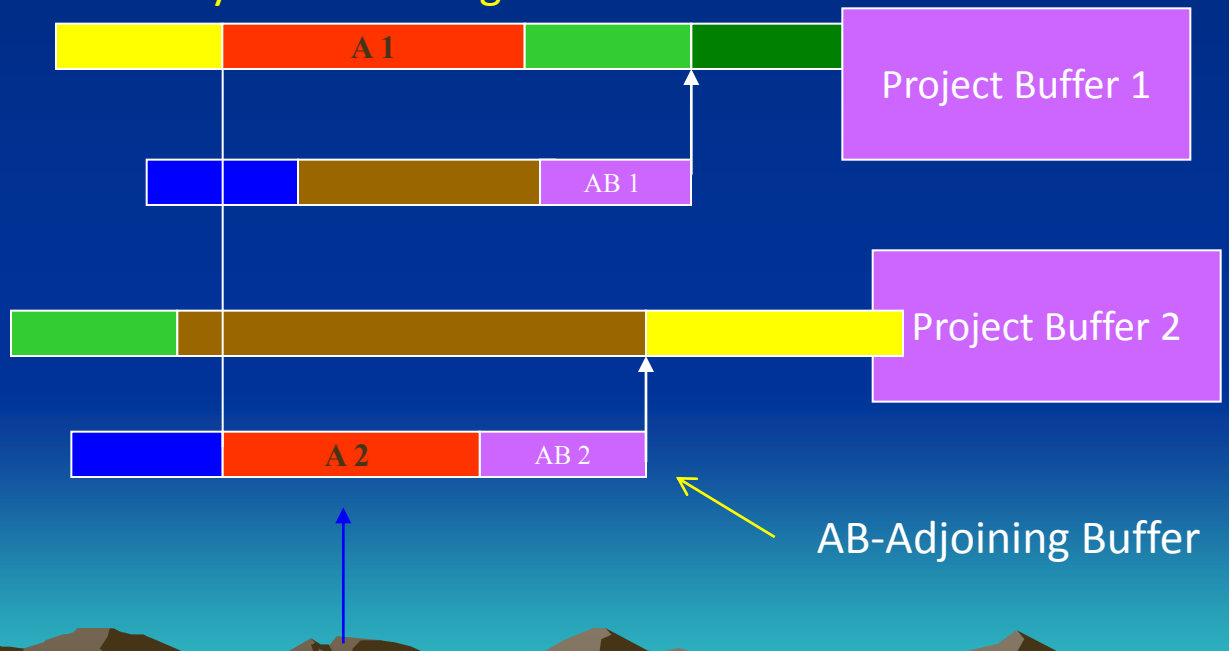
Buffer consumption



Rate of penetration is used to assign priorities to the partial activities

Priorities assigned to resources

- If one resource have (**red A**) to be assigned to two activities starting in the same moment, so the first activity which will start is the one belonging to the project with bigger project buffer penetration (Project 2).
- If none of all project buffers were penetrated with previous activities, so the first starts this activity which belongs to the critical chain.



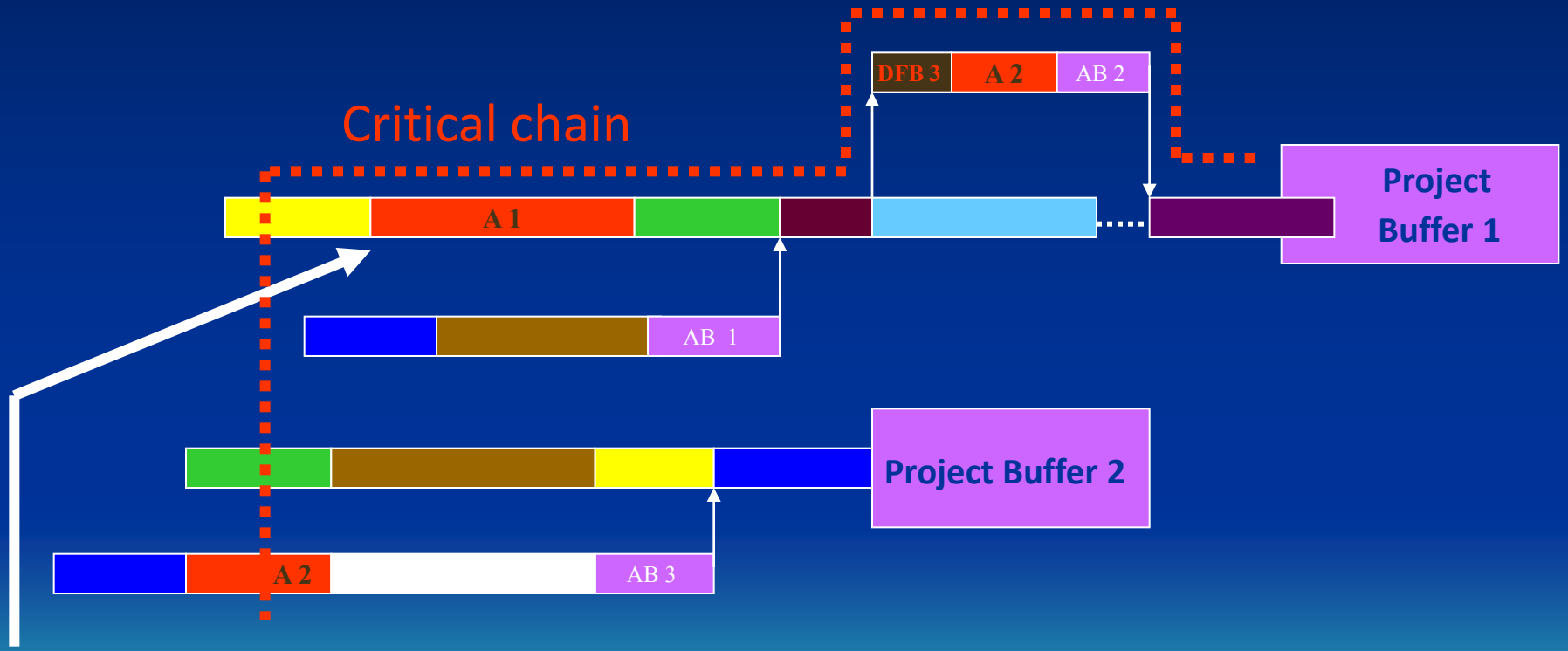
A2 starts firstly because PB 2 is partially consumed (penetrated)

PB-Project Buffer

AB-Adjoining Buffer

Priorities assigned to resources

A1 (P1) -> A2 (P1) -> A2 (P2)



This activity (**A1**) starts first because it is a part of the Critical chain and Project Buffer 1 is penetrated

Main benefits of the Critical Chain (CC) usage

(Home study)

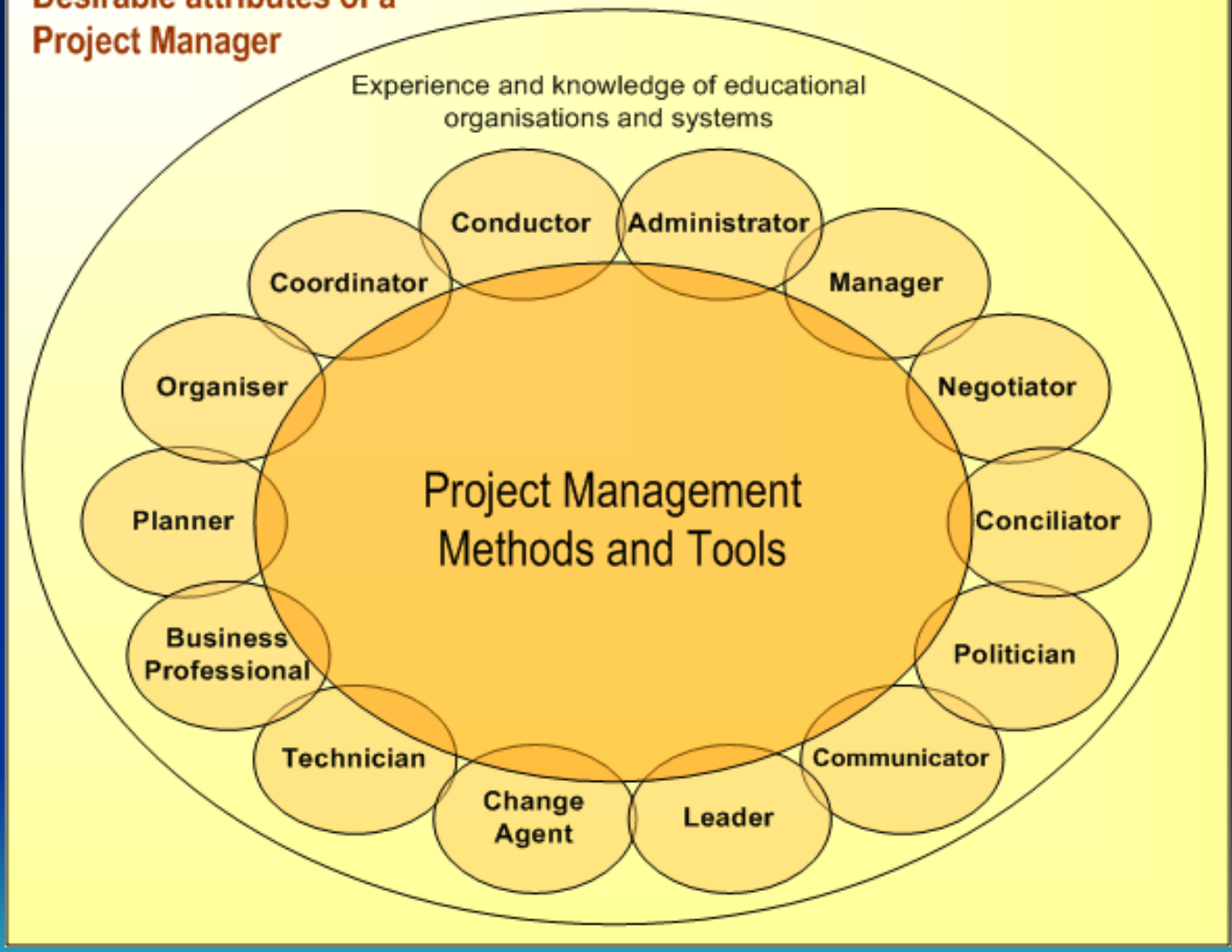
- All projects applying the CC methodology will end earlier than projects being managed by any other method
- Promised delivery times are fulfilled with higher rate of credibility
- You will have more free capacity of all used resources

Main benefits of the Critical Chain (CC) usage

(Home study)

- Better initial estimation about project timing and thus bore **accurate planning**
- You won't encounter any problems when initiating projects, because you always take into considerations the limitations imposed by the **drum type resource**
- Decrease of unfavourable effects such as Student syndrome, Murphy attacks and impacts of Parkinson s laws by redeployment and integration of all buffers to one and only one project buffer at the end of the project
- Utilization of benefits (saved time and capacities of the resources) caused by **earlier ended** activities
-

Desirable attributes of a Project Manager



Different approaches to Project Management

Waterfall methodology–Big design up front, milestones,..**no iterations !!**
It is based on prediction

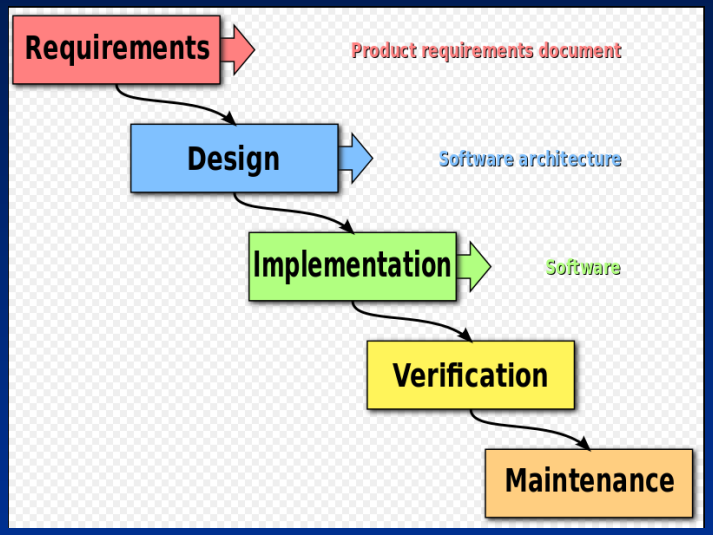
Agile methodology– Scrum (Sprints,..)- **iterations approach**

Critical Chain- based on **TOC** methodology
(see https://www.youtube.com/watch?v=mpc_FdAt75A)

Prince2 - Projects In Controlled Environments (rolling wave planning)

Lean management

Waterfall - predictive approach



Clients may not know exactly what their requirements are before they see working software and so change their requirements, leading to redesign, redevelopment, and retesting, and increased costs

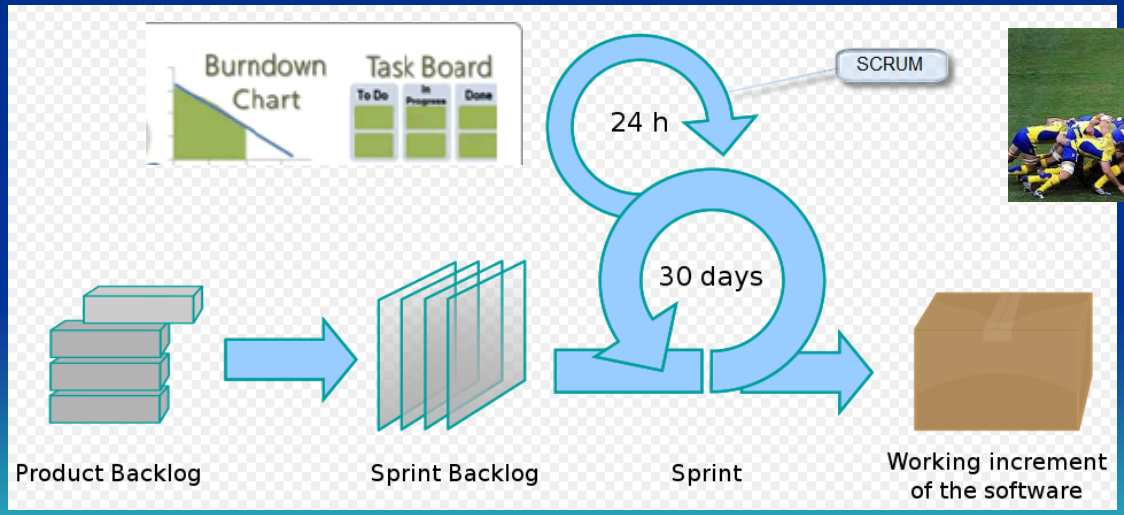
One of the differences between agile software development methods and waterfall is **the approach to quality and testing**. In the waterfall model, there is always a ***separate testing phase*** after a ***build phase***; however, in **agile software development** (see next slide) testing is completed in the same iteration as programming

Agile PM approach – for instance SCRUM (relation to RFP)

Request for Proposal

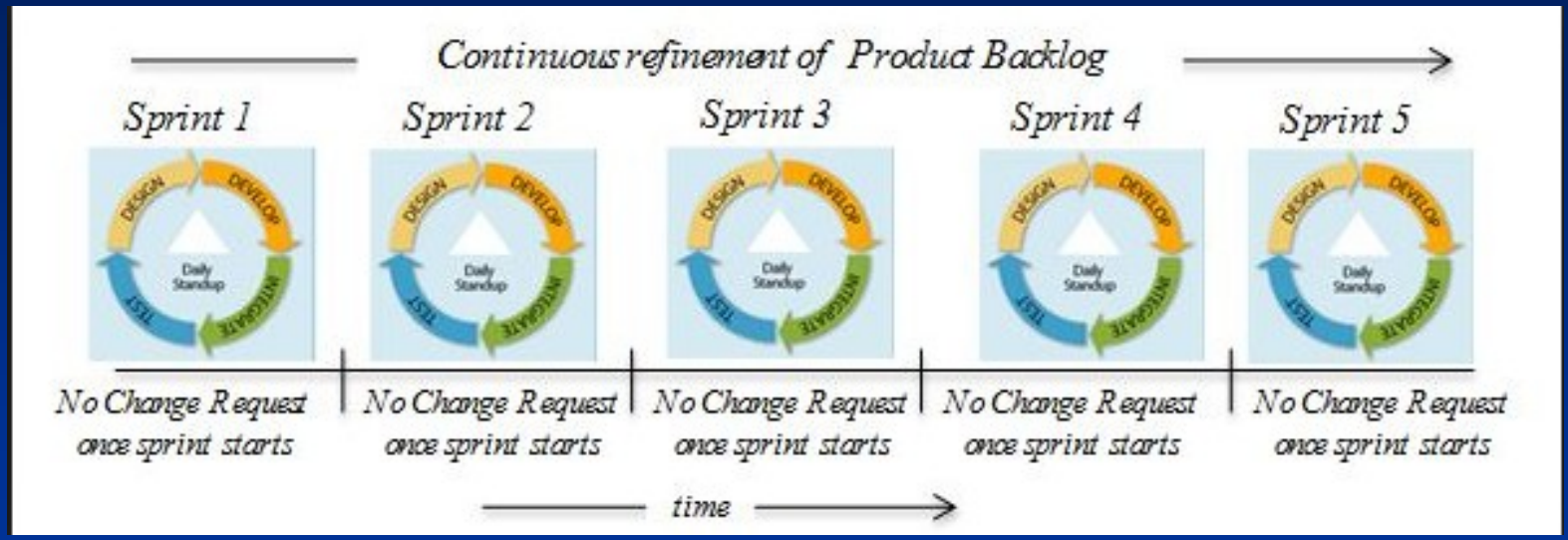
Scrum is an iterative and incremental agile SW development framework for managing product development

A key principle of **Scrum** is the **dual recognition** that customers will change their minds about what they want or need (often called **requirements volatility**) and that there will be unpredictable challenges for which a **predictive** or planned approach is not suited



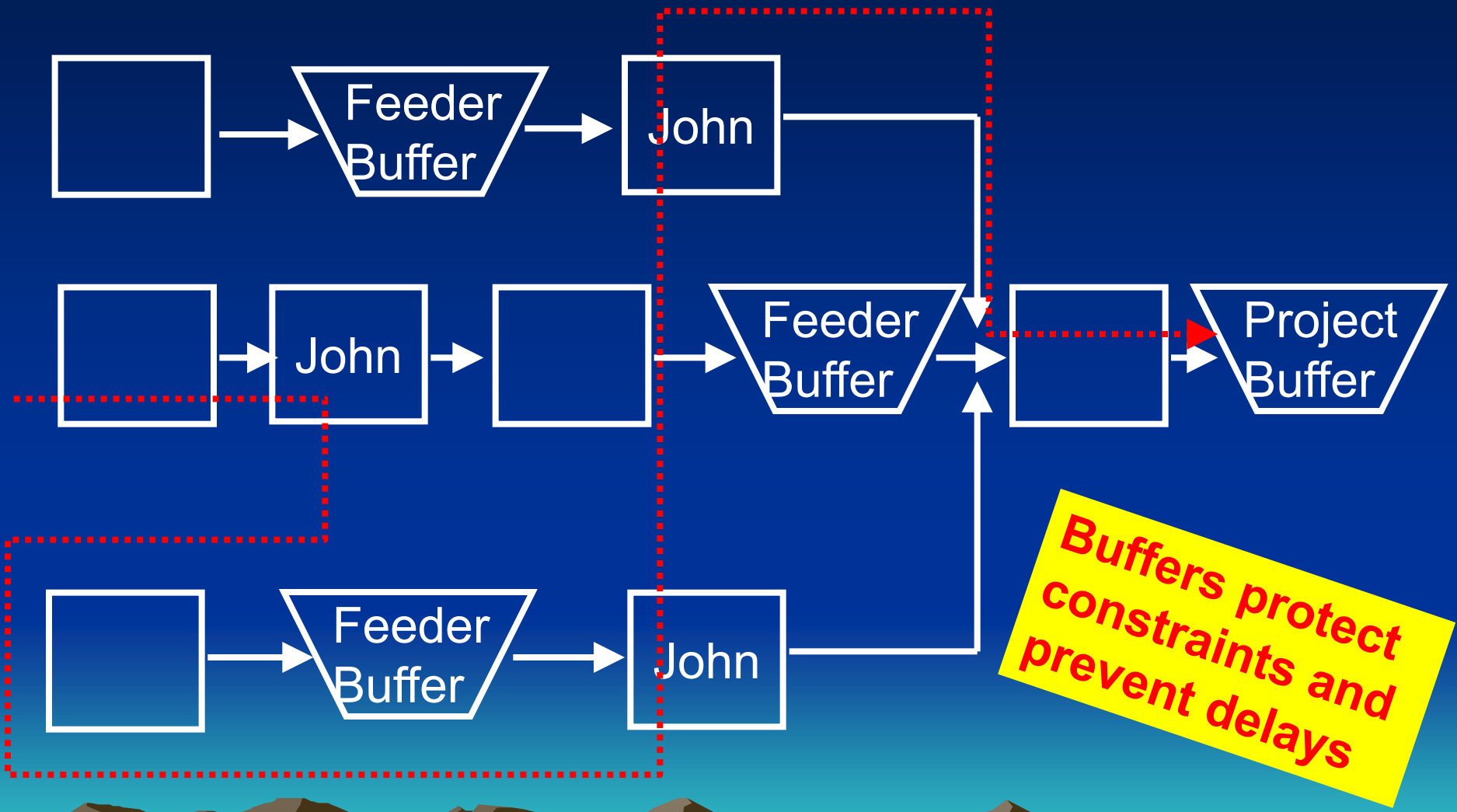
Sprint->Stage
Scrum ->Iteration, daily work

SCRUM


Continuous refinement of Product Backlog
Sprint 1
Sprint 2
Sprint 3
Sprint 4
Sprint 5
DESIGN
DEVELOP
TEST
INTEGRATE
Daily Standup
No Change Request once sprint starts
time

Some slides to complete presentation – final review

Critical Chain Solutions



Drum prinicple- CCPM)

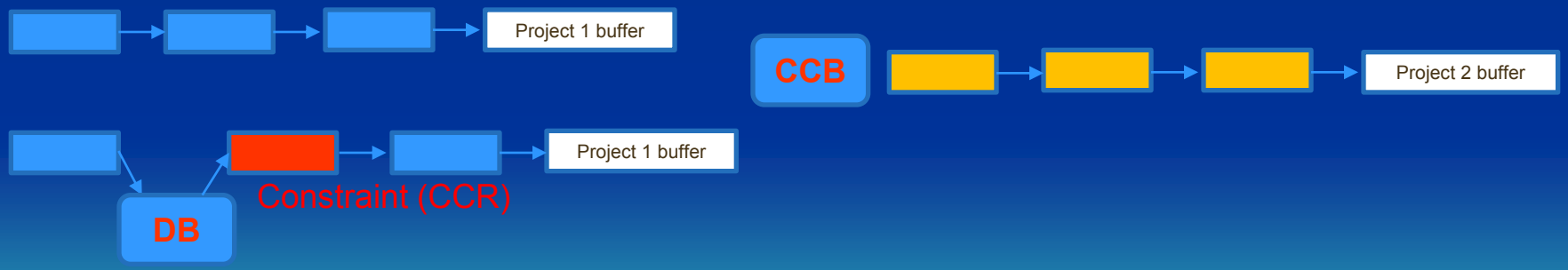
(CC=Critical chain)

Drum – sets the beat (pace, tempo) for the firm's **throughput ->constraint**

Drum – could be person, department, policy, resource

Capacity Constraint Buffer (CCB) – safety margin between projects

Drum buffer (DB) – extra safety before the constraint CCR (DRUM)



Project management and Drum (Capacity Constraint resource= CCR) - principle of schedule projects based on drums

A principle that is very similar to the five steps of TOC, where the term Drum replaces the term Bottleneck

1. Identify the **drum** **(bottleneck)**
2. Exploit the **drum**
   - Prepare a **project schedule** for ech project
   - Determine priority for the **drum**
   - Create the **drum schedule**
3. Subordinate the **project schedules to steps 2 and 3**
4. Elevate the capacity of the **drum**
5. Go back to step 1

Schedule project based on Drum concept ->Designate Critical Chain (CC)->
->Insert Drum buffer before capacity constraint resource (Drum) >Resolve conflicts

Difference Between Buffer and Float (or Slack) - (home study)

People often get confused between **buffer** and **float**. They find these two terms similar, however, they are not.

The following are a few differences between the **float** and **buffer**:

•**Float** (slack) is a **critical path** phenomenon, while **buffer** belongs to the **critical chain**.

•A **float** (slack) is a difference between the duration of the critical path and the non-critical path.

•**A float** is zero on a **critical path.**

•A **buffer** is based on contingencies (unpredictable actions). For example, the project **buffer** is about 50% of the safety time that you have removed from the activity estimate duration. Based on the definition of **buffe**r, it is not zero on a **critical chain** or any other chain.

•A **float** is the same for all activities on a **non-critical path**, any activity can consume it partially or fully, and other activities can utilize balance. There is no further analysis.

•Any activity can also borrow a **buffer** if the activity is delayed. The project manager analyzes the remaining **buffer** to find the status of the project.

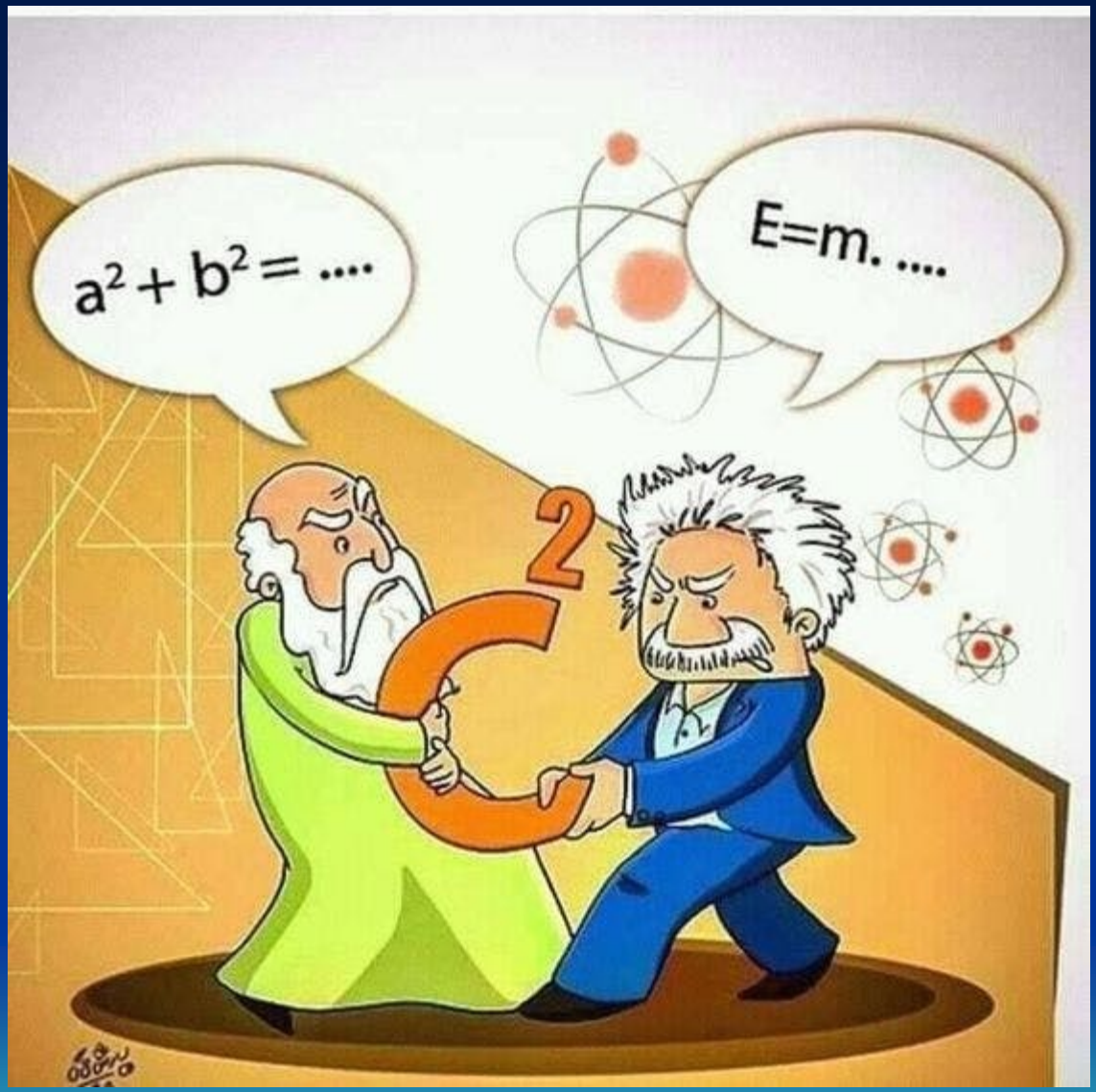
•**Buffers** can be divided into three categories: project buffer, feeding buffer and resource buffer

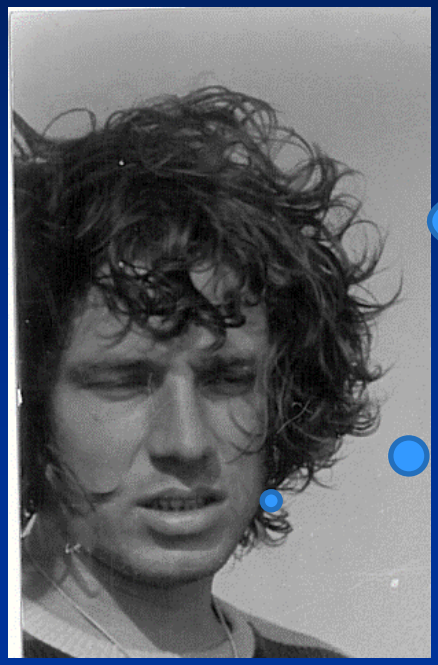
Resources

http://www.fortezzaconsulting.com/blog/5-myths/

https://www.workflowmax.com/blog/choose-your-project-management-methodology-pros-and-cons-of-agile-waterfall-prism-and-more

Very usefull one :
https://www.tutorialspoint.com/management_concepts/pareto_chart_tool.htm


$a^2 + b^2 = ...$
E=m. ....
2



The author of the material thanks you very much for your attention