



**THEORY OF CONSTRAINTS**  
INTERNATIONAL CERTIFICATION ORGANIZATION

2016 TOCICO International Conference

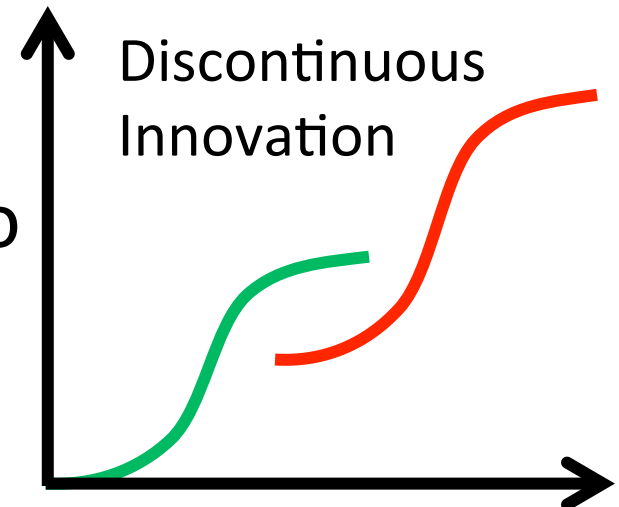
**How “Agile” CCPM enables us to  
successfully complete challenging  
projects for major upgrades of  
existing software products, and  
maintain 100% DDP for years**

Koichi Ujigawa

# About the Project

2016 TOCICO International Conference

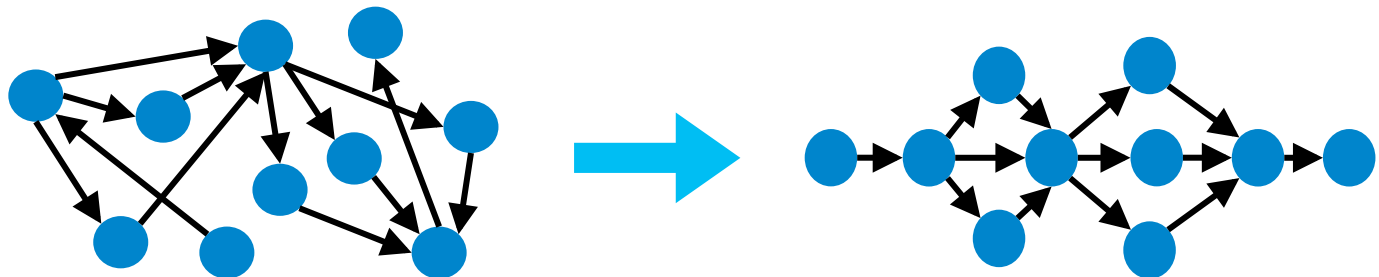
- The project was challenging, because:
  - it was a Major Upgrade of an existing software product
  - it required “Discontinuous Innovation” and state-of-the-art technologies  
(we need to get on the red curve)
- And also, needless to say, due date of the project had to be met and the flow of the system had to be maintained



# About the Project (cont.)

2016 TOCICO International Conference

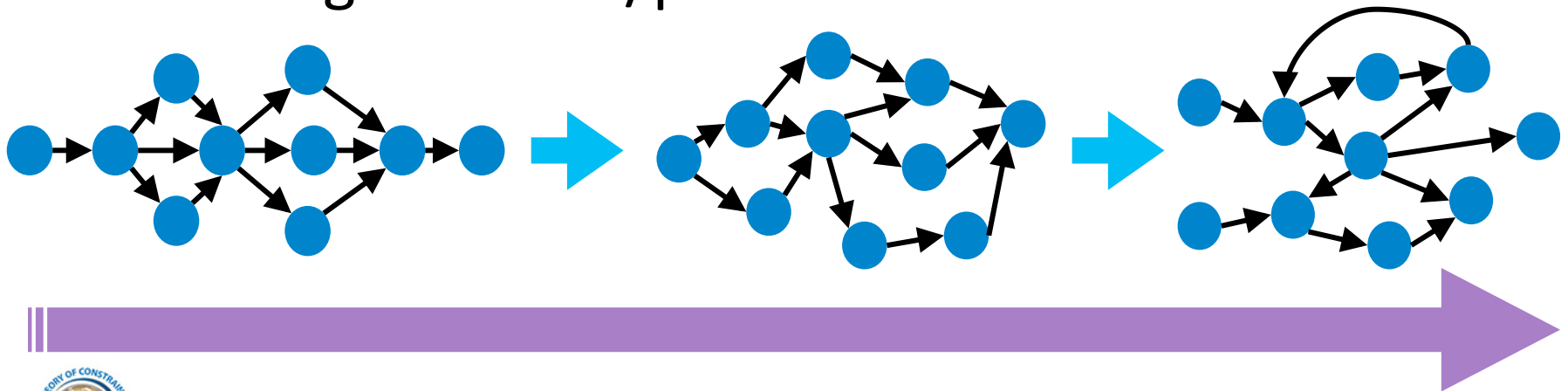
- There are several modalities for recognizing a project's environment, such as “Complicated” and “Complex”
- In the “Complicated” environment, Cause and Effect can be captured in planning phase, even it seems to be (almost) impossible at first sight



# About the Project (cont.)

2016 TOCICO International Conference

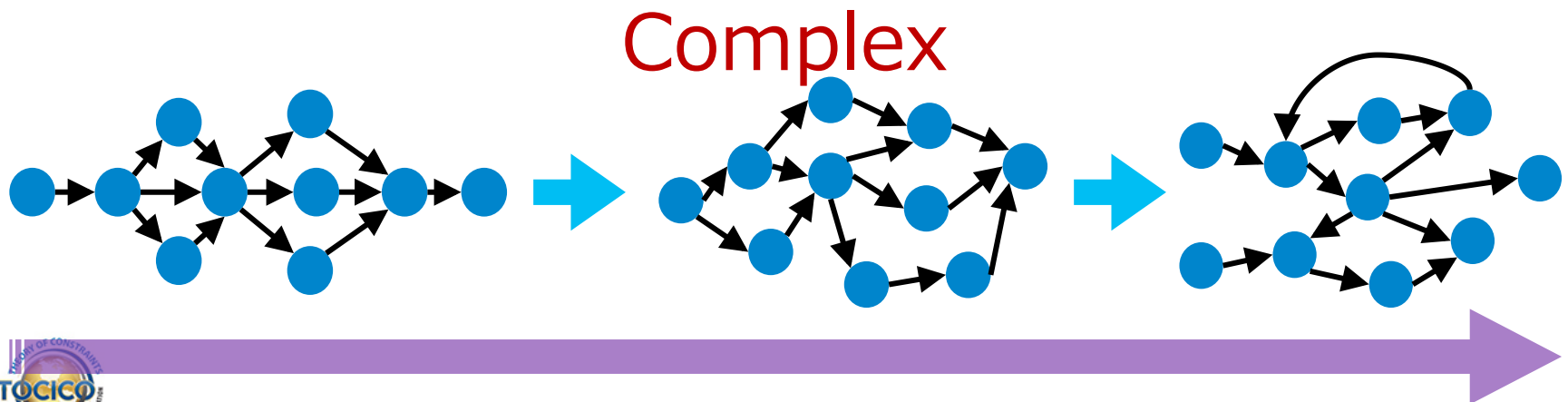
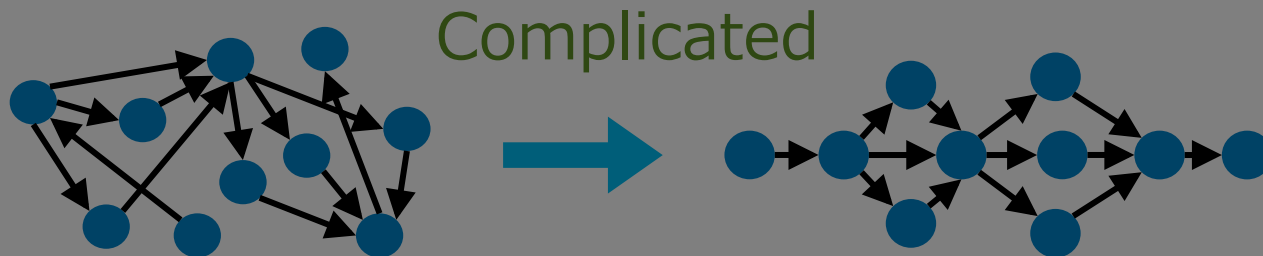
- In the “Complex” environment, Cause and Effect cannot always be captured, even it appears so in retrospect
- It constantly changes according to external conditions
- But a big direction/pattern can be obtained



# About the Project (cont.)

2016 TOCICO International Conference

- The project was challenging, because its environment was recognized as “Complex”



# About the Project (cont.)

2016 TOCICO International Conference

- We believe that the following perceptions are more important than usual in the “Complex” environment
  - “it is better to be approximately right than to be precisely wrong”
  - “the more complex the problem is, the more simple the solution has to be”
- And we believe that we should be more “innovative” and “inventive” than usual



# About the Project (cont.)

2016 TOCICO International Conference

- We knew that more adaptability, responsiveness and autonomy would enhance the intuition of developers during the project
- We decided to apply “Agile” CCPM
  - In our pipeline, projects are managed by either standard CCPM (\*) or “Agile” CCPM

\*The term “standard” CCPM is used to avoid confusion during the presentation



# “Agile” CCPM in a nutshell

2016 TOCICO International Conference

- “Agile” CCPM is an enhancement of CCPM, employing the following for SW development (\*):
  - Modified Network Construction Process
  - Modified Buffer Management Method

\*Conceptually, this can be applied to other environments such as R&D, NPD and organization design



# “Agile” CCPM in a nutshell (cont.)

2016 TOCICO International Conference

- All the benefits and attributes of CCPM are maintained while building in the adaptability, responsiveness and autonomy of some of the practices adopted by Agile
- It is agile, not in the sense that it is a hybridization of Agile and CCPM, but in the sense that it makes standard CCPM more adaptable to change

# “Agile” CCPM in a nutshell (cont.)

2016 TOCICO International Conference

- Although we use slightly different techniques in certain situations, none of the fundamental rules or benefits of CCPM are sacrificed
- For this reason, the application enhancement is truly “CCPM”

# Quick Review of standard CCPM

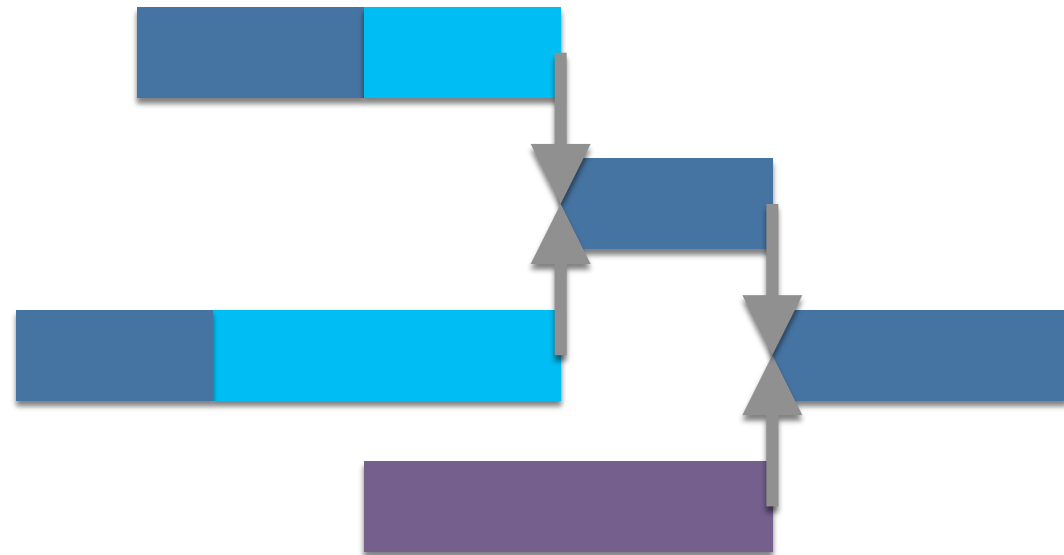
2016 TOCICO International Conference

- Procedure in planning
  - Add tasks and dependencies
  - Level resources
  - Identify the Critical Chain
  - Insert buffers
- Procedure in execution
  - Execute tasks and report RDU
  - Update buffer status
  - Take recovery action as required

# Procedure in Planning - Standard CCPM -

2016 TOCICO International Conference

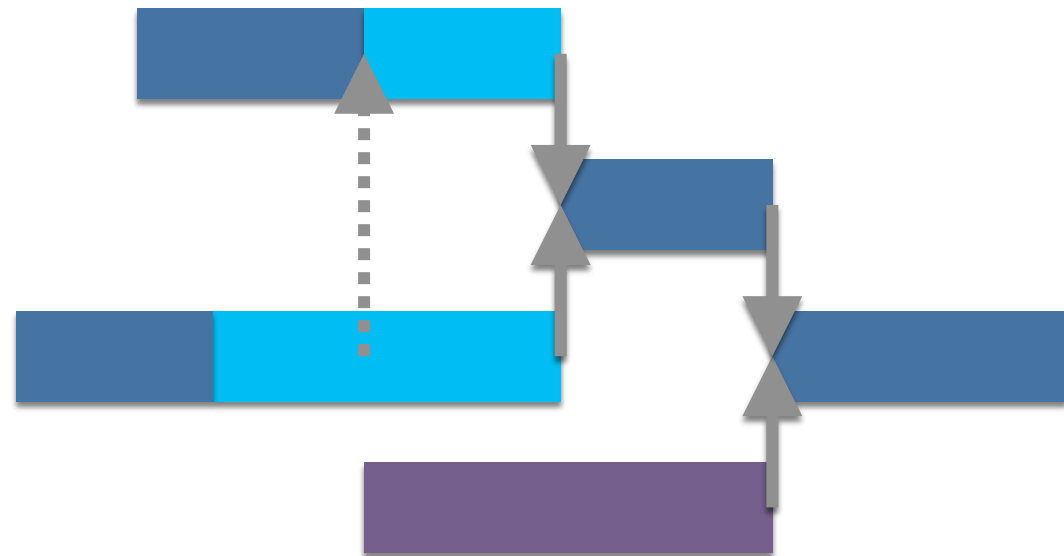
## Add tasks and dependencies



# Procedure in Planning - Standard CCPM -

2016 TOCICO International Conference

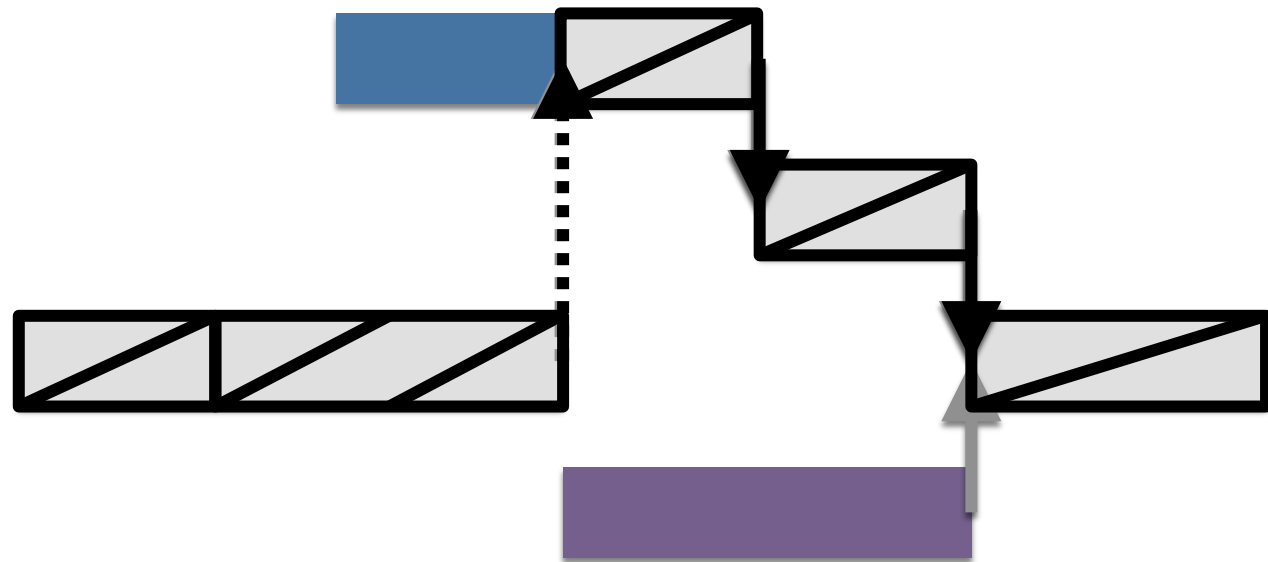
## Level resources



# Procedure in Planning - Standard CCPM -

2016 TOCICO International Conference

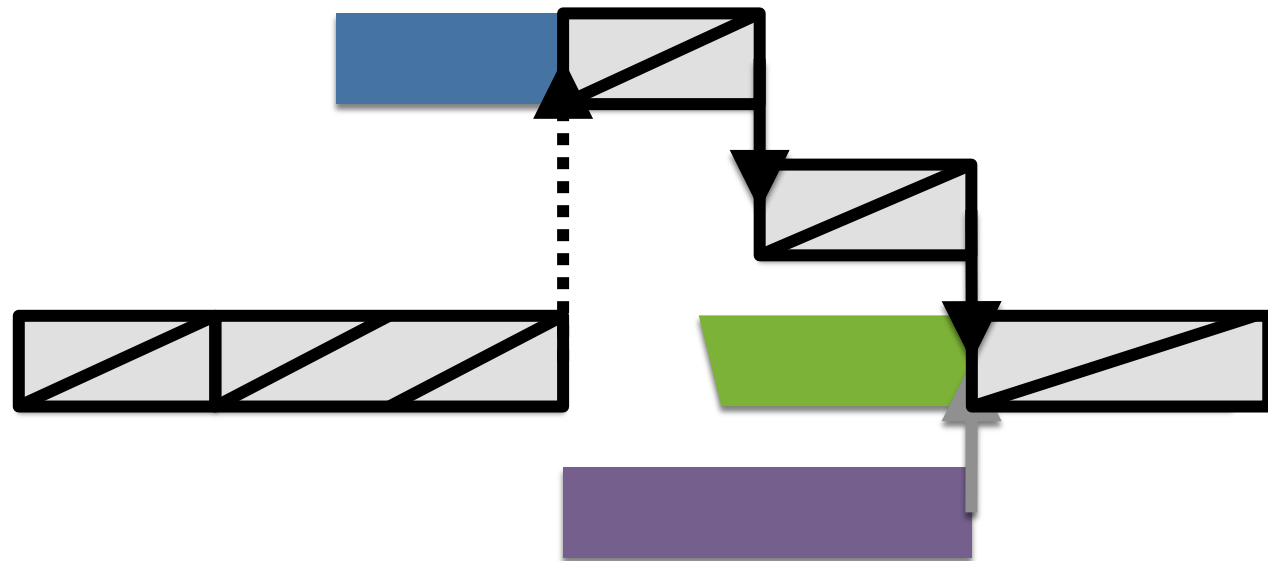
## Identify the Critical Chain



# Procedure in Planning - Standard CCPM -

2016 TOCICO International Conference

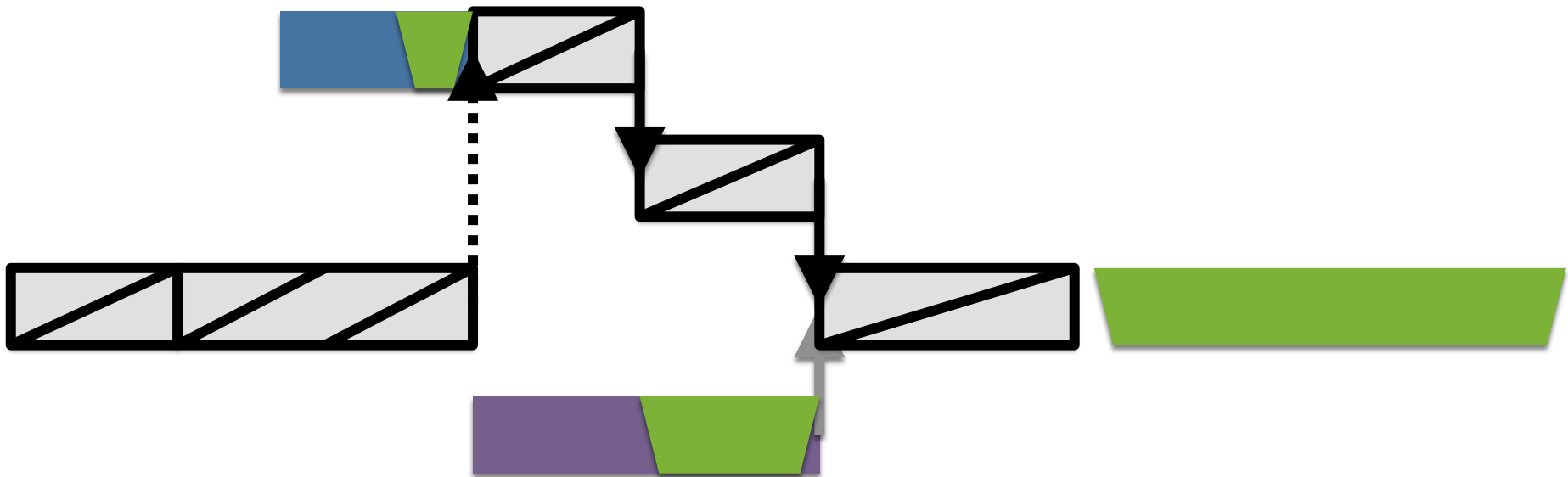
## Insert the project buffer



# Procedure in Planning - Standard CCPM -

2016 TOCICO International Conference

## Insert feeding buffers

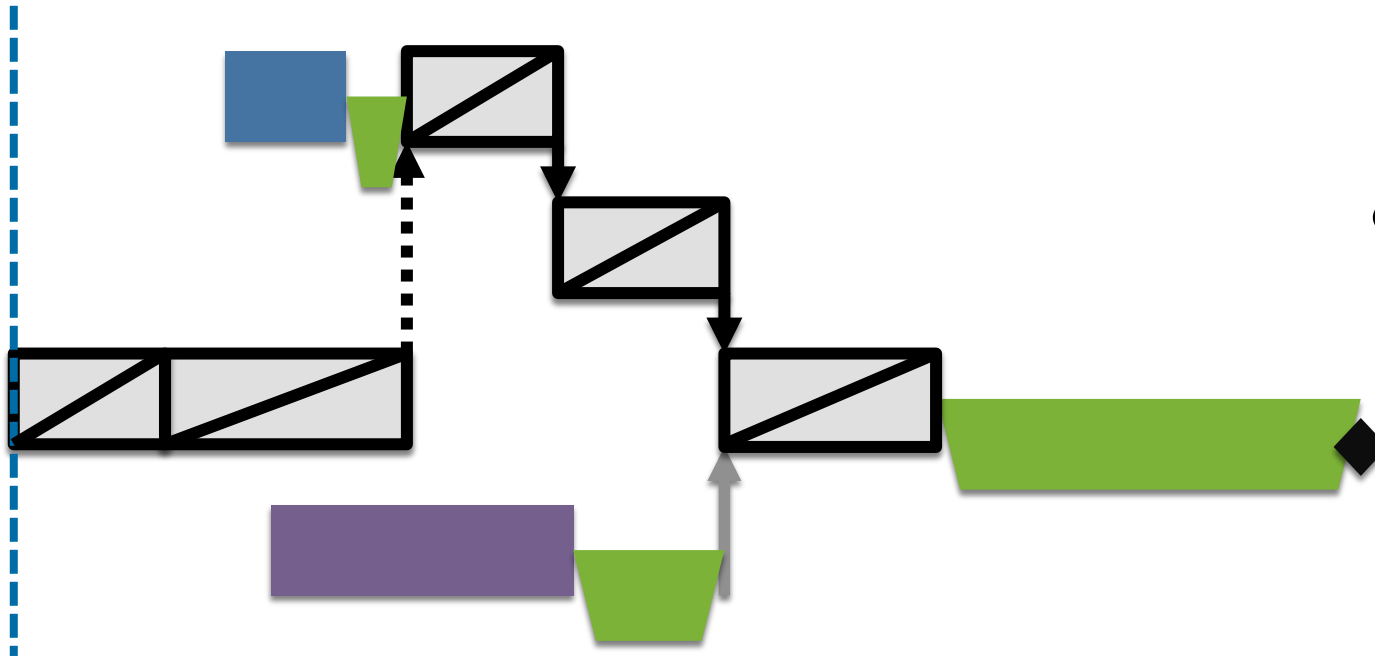




# Procedure in Execution - Standard CCPM -

2016 TOCICO International Conference

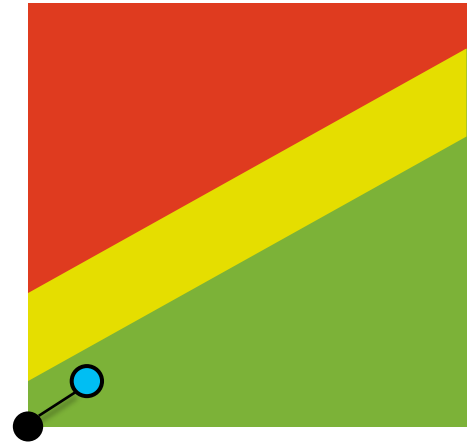
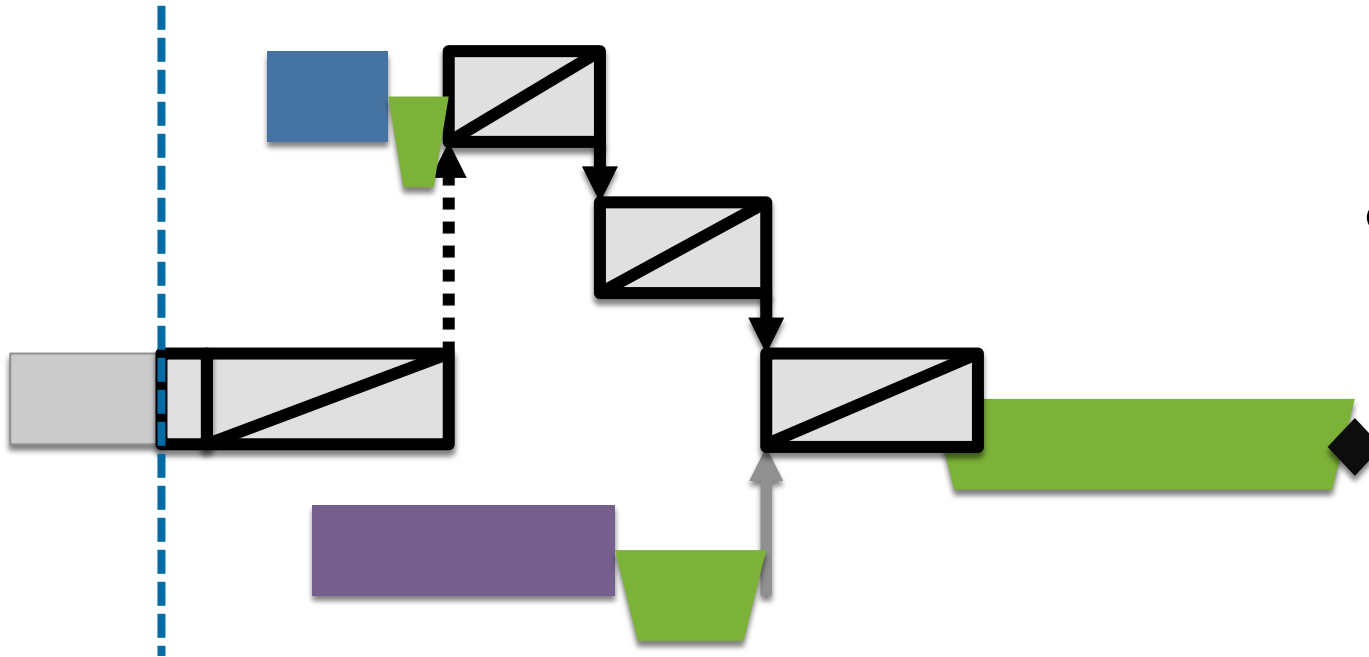
Execute tasks and report RDU  
Update buffer status



# Procedure in Execution - Standard CCPM -

2016 TOCICO International Conference

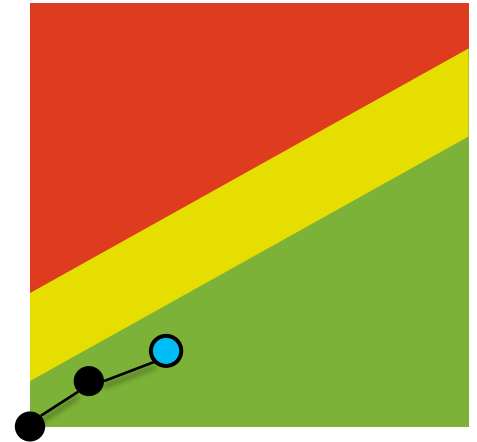
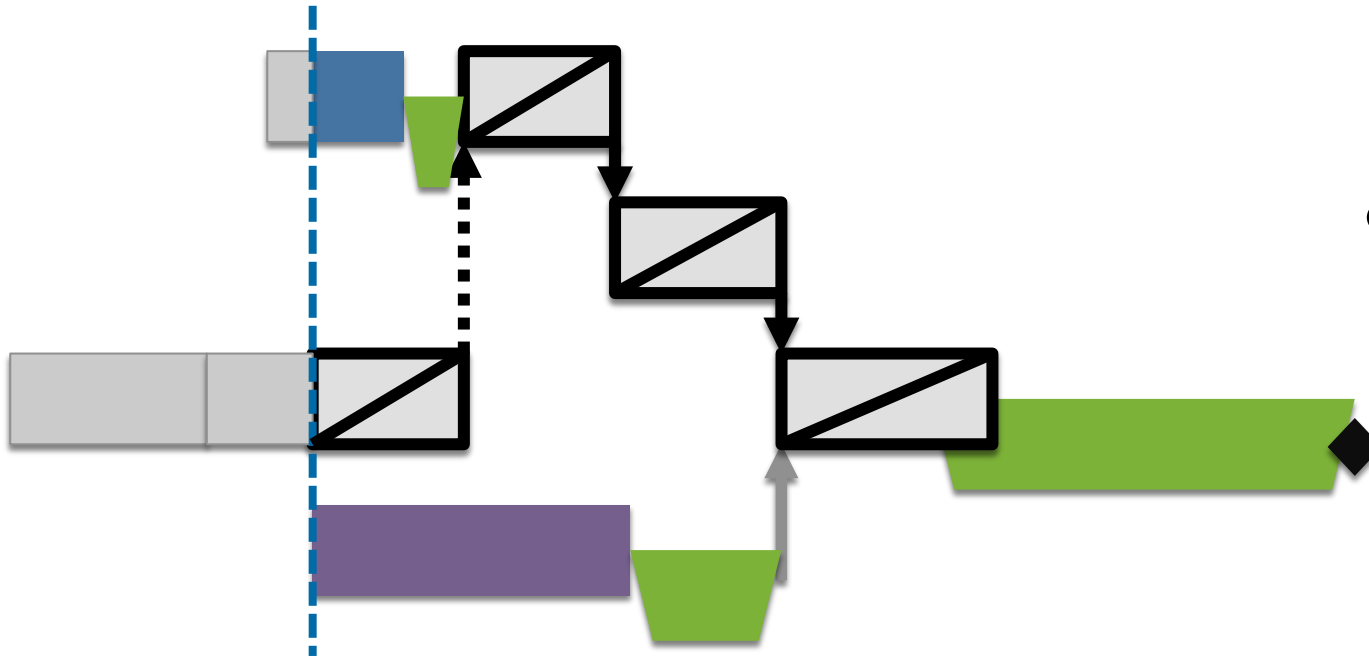
Execute tasks and report RDU  
Update buffer status



# Procedure in Execution - Standard CCPM -

2016 TOCICO International Conference

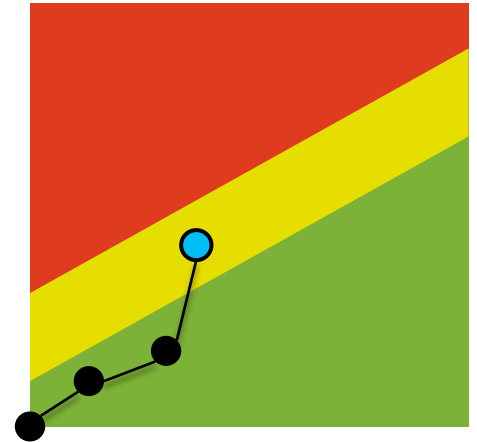
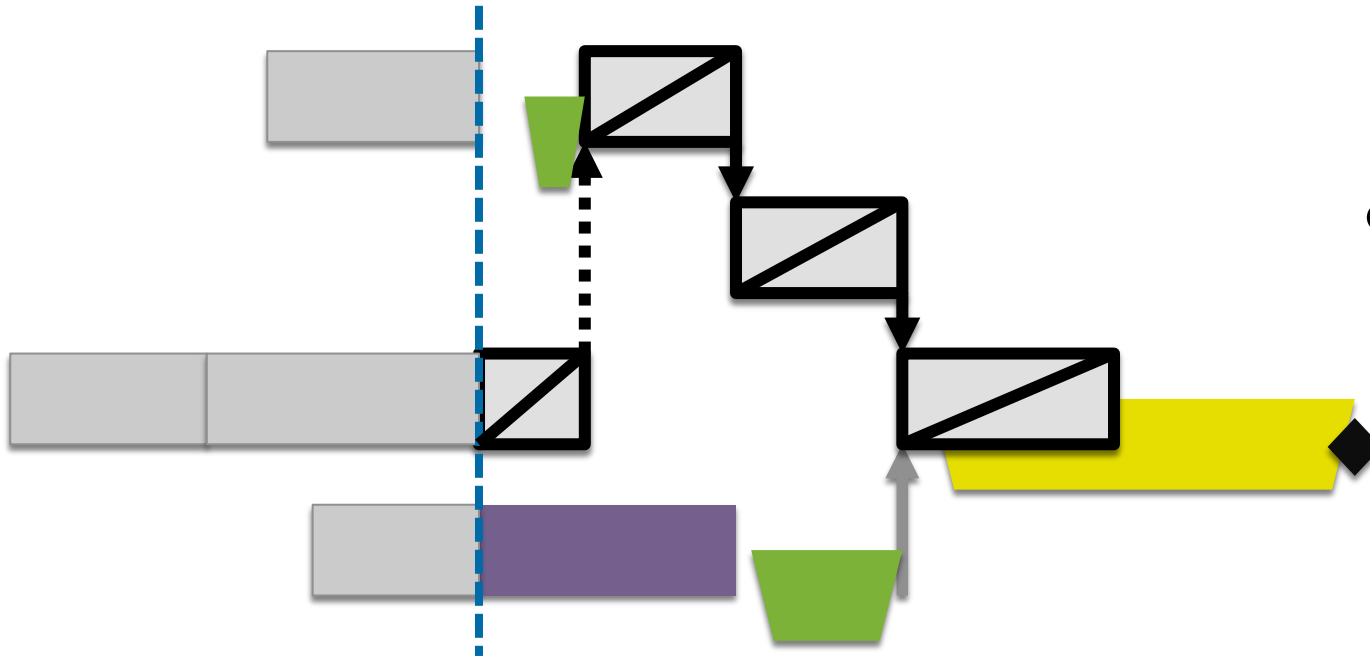
Execute tasks and report RDU  
Update buffer status



# Procedure in Execution - Standard CCPM -

2016 TOCICO International Conference

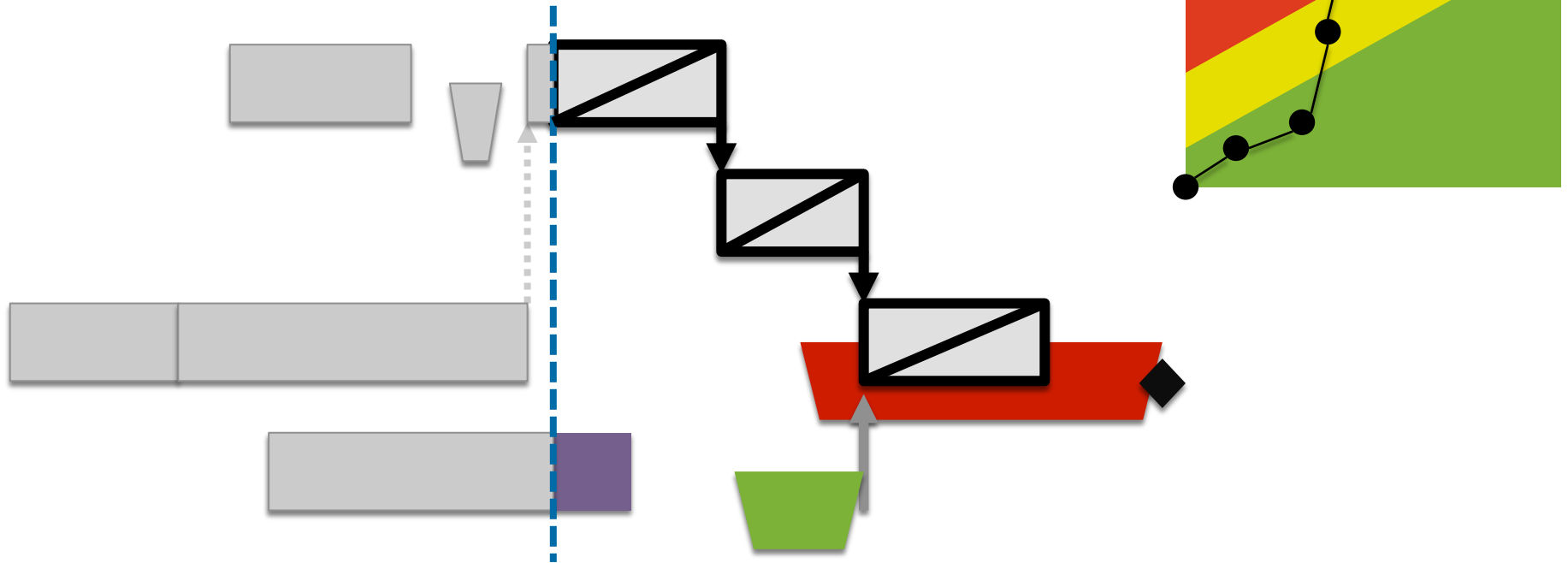
Execute tasks and report RDU  
Update buffer status



# Procedure in Execution - Standard CCPM -

2016 TOCICO International Conference

Take recovery actions as required

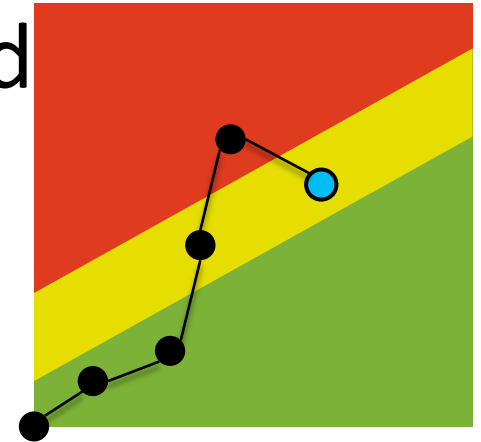
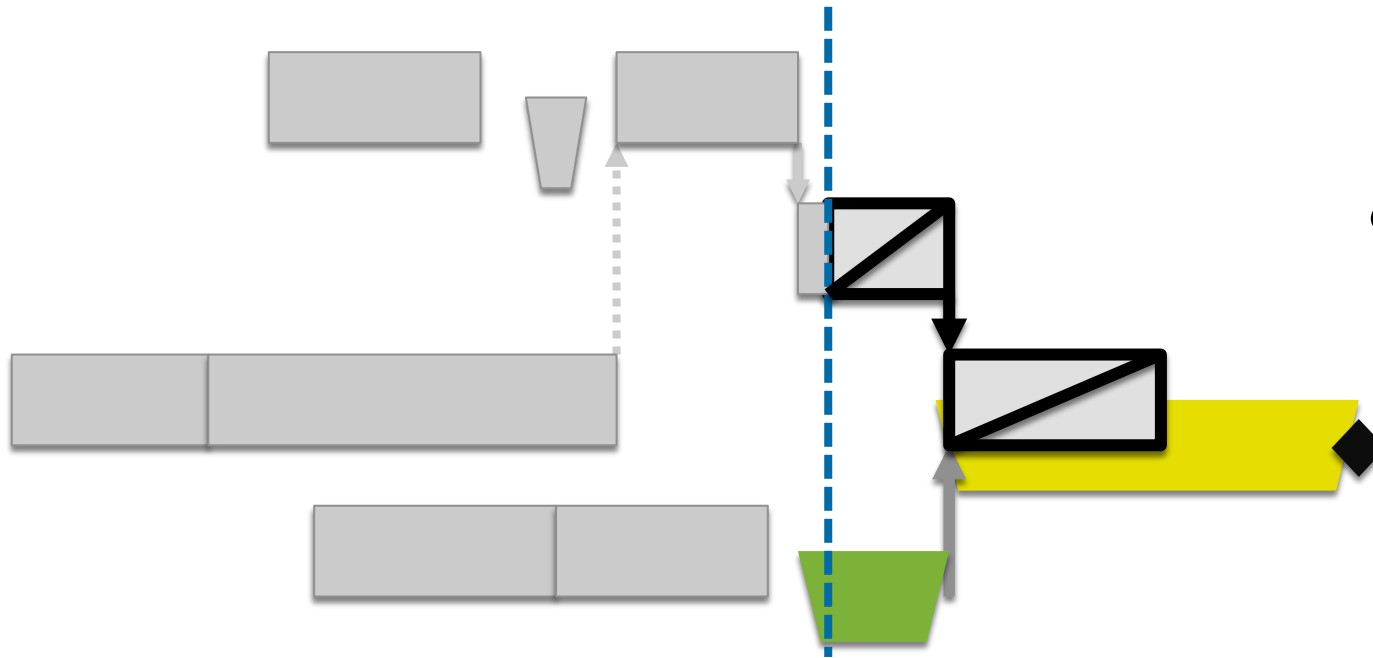


Before a recovery action is taken

# Procedure in Execution - Standard CCPM -

2016 TOCICO International Conference

Take recovery actions as required



After a recovery action is taken

# Overview of “Agile” CCPM

2016 TOCICO International Conference

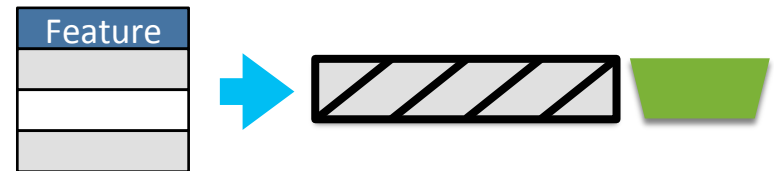
## CCPM

### “Standard”

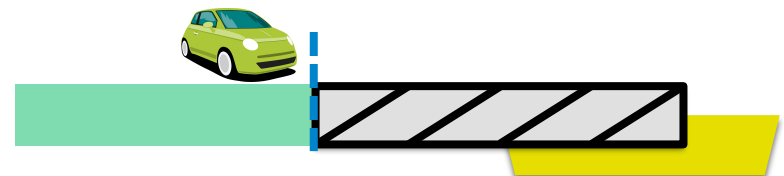
- Procedure in planning
  - Add tasks and dependencies
  - Level resources
  - Identify the Critical Chain
  - Insert buffers
- Procedure in execution
  - Execute tasks and report RDU
  - Update buffer status
  - Take recovery action as required

### “Agile”

#### Modified Network Construction Process



#### Modified Buffer Management Method



# Flow of Modified Network Construction Process

2016 TOCICO International Conference

Car Drive

(The process is similar to a long drive)

Where to Go?

Set the goal/purpose of the drive



How Far?

Get the distance to the goal



How Long?

Estimate time required according to the distance and the average velocity



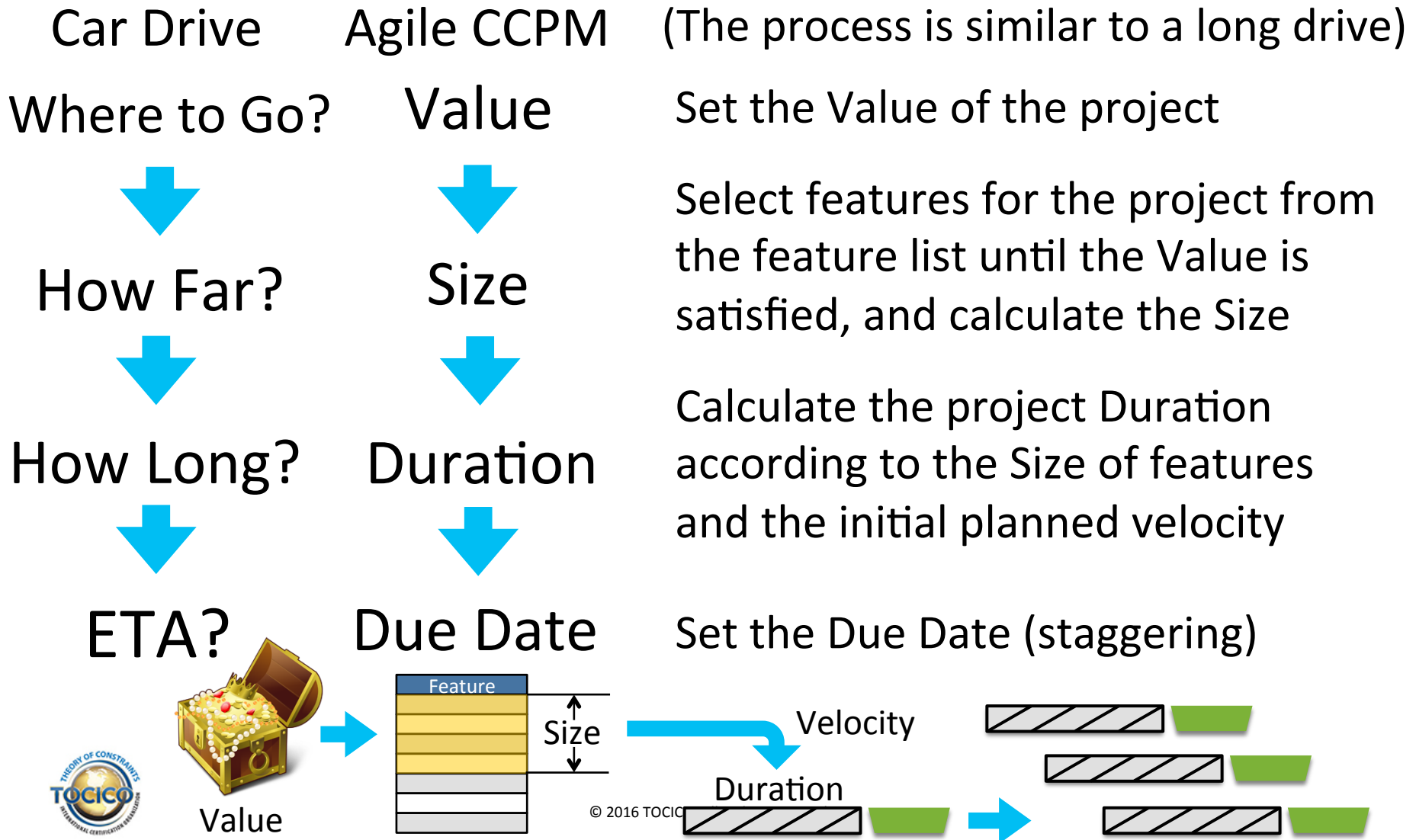
ETA?

Get the estimated time of arrival



# Flow of Modified Network Construction Process (cont.)

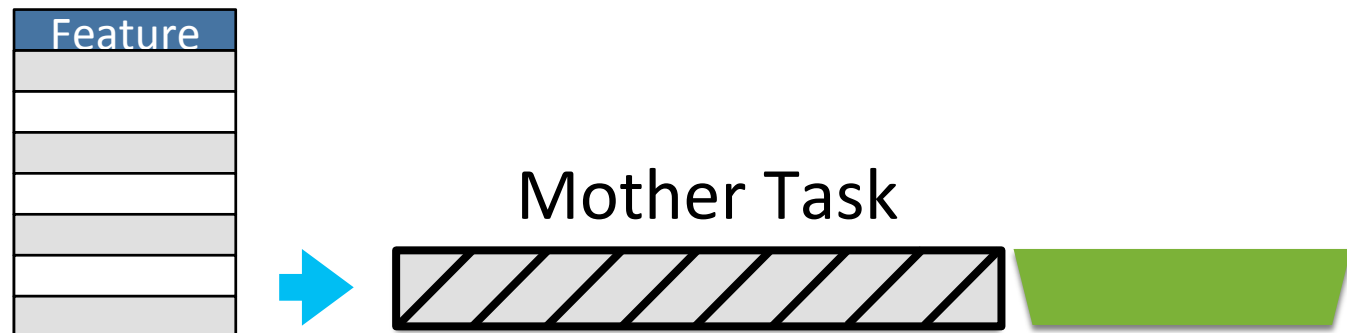
2016 TOCICO International Conference



# Procedure for Network Construction Process in “Agile” CCPM

2016 TOCICO International Conference

- A network is developed from a Feature List
- The structure of resulting network can be very simple, and is necessary and sufficient for performing the Modified Buffer Management Method
- Task(s) may represent multiple features (even all features) and are referred to as the “Mother Task”



# Procedure for Network Construction Process in “Agile” CCPM (cont.)

2016 TOCICO International Conference

No	Feature	Size
1	F001	8
2	F002	5
3	F003	
4	F004	
5	F005	
6	F006	
7	F007	
8	F008	
9	F009	
10	F010	
11	F011	
12	F012	

No	Feature	Size
13	F013	13
14	F014	5
15	F015	8
25	F025	13
26	F026	3
27	F027	5
28	F028	8
29	F029	3
30	F030	13
31	F031	13
32	F032	5

- A Feature List is prepared/ maintained for a product/ service prior to creating the network of a project
- Features are described in user stories
- Features are sorted in

# Procedure for Network Construction Process in “Agile” CCPM (cont.)

2016 TOCICO International Conference

No	Feature	Size
1	F001	8
2	F002	5
3	F003	
4	F004	
5	F005	
6	F006	
7	F007	
8	F008	
9	F009	
10	F010	
11	F011	
12	F012	

- A Feature List is prepared/maintained for a product/service prior to creating the network of a project
- Features are described in

As a pipeline manager, I want to display a multi-project fever chart, because it enables us to see the health of whole pipeline and tells us which project needs help at a glance.

# Procedure for Network Construction Process in “Agile” CCPM (cont.)

2016 TOCICO International Conference

No	Feature	Size
1	F001	8
2	F002	5
3	F003	
4	F004	
5	F005	
6	F006	
7	F007	
8	F008	
9	F009	
10	F010	
11	F011	
12	F012	

No	Feature	Size
13	F013	13
14	F014	5
15	F015	8
25	F025	13
26	F026	3
27	F027	5
28	F028	8
29	F029	3
30	F030	13
31	F031	13
32	F032	5

- A Feature List is prepared/ maintained for a product/ service prior to creating the network of a project
- Features are described in user stories
- Features are sorted in approximate order based on value/importance for the product/service
- The relative size of each feature is estimated

# Procedure for Network Construction Process in “Agile” CCPM (cont.)

2016 TOCICO International Conference



Set the Value of the project

Select features for the project from the feature list until the Value is satisfied

Set a scope buffer

No	Feature	Size
1	F01	8
2	F02	5
3	F03	
4	F04	
5	F05	
6	F06	
7	F07	
8	F08	
9	F09	
10	F10	
11	F11	
12	F012	

No	Feature	Size
13	F01	13
14	F01	5
15	F01	8
16	F01	
17	F01	
18	F01	
19	F01	
20	F01	
21	F01	
22	F01	
23	F01	
24	F01	
25	F01	
26	F02	13
27	F02	3
28	F02	5
29	F02	8
30	F02	3
31	F03	13
32	F03	5

# Procedure for Network Construction Process in “Agile” CCPM (cont.)

2016 TOCICO International Conference

When the feature list consists of similar features  
(requires the same type of skill to execute)

Total story points of all features  
as the Size of the project

Calculate the Duration of the initial  
task bar representing all features by  
dividing the Size by the Initial  
Planned Velocity

No.	Feature	Size
1	F001	8
2	F002	5
3	F003	8
4	F004	
5	F005	
6	F006	
7	F007	
8	F008	
9	F009	
10	F010	
11	F011	
12	F012	

No.	Feature	Size
13	F013	13
14	F014	5
15	F015	8
16	F016	
17	F017	
18	F018	
19	F019	
20	F020	
21	F021	
22	F022	
23	F023	
24	F024	
25	F025	13
26	F026	3
27	F027	5
28	F028	8
29	F029	3
30	F030	13

ex. 220 [sp]

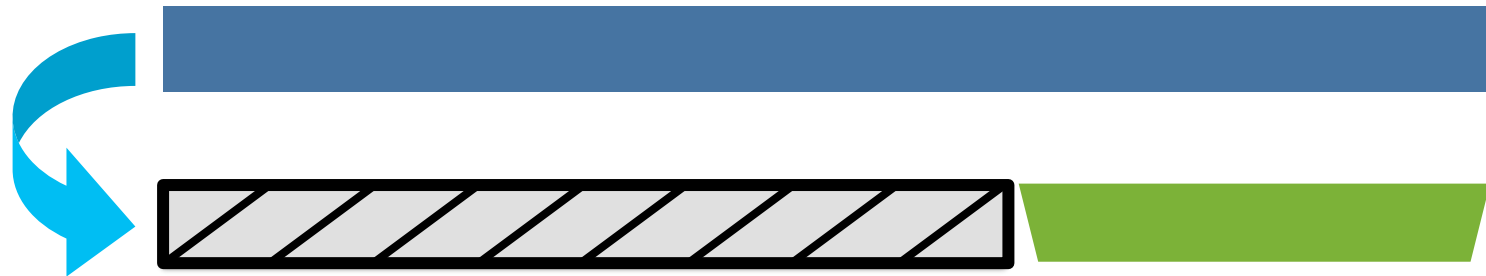
$$\text{ex. } 220 [\text{sp}] / 2 [\text{sp/d}] = 110 [\text{d}]$$

# Procedure for Network Construction Process in “Agile” CCPM (cont.)

2016 TOCICO International Conference

When the feature list consists of similar features  
(requires the same type of skill to execute)

Convert the initial task bar to a Critical Chain schedule consisting of a Mother Task and a Project Buffer



Mother Task

Project Buffer



# Procedure for Network Construction Process in “Agile” CCPM (cont.)

2016 TOCICO International Conference

When the feature list consists of different types of features (requires different skills to execute)

Make groups of features by type (or based on some other rule), and total the story points of each group

ex. **Group #1** : 140 [sp]

**Group #2** : 80 [sp]

The diagram illustrates the grouping of features into two categories, Group #1 and Group #2, based on their size and type. Group #1 features are highlighted in light blue, and Group #2 features are highlighted in light red. The total story points for each group are calculated and shown next to the group name.

No.	Feature	Size
1	F001	8
2	F002	5
3	F003	8
4	F004	8
5	F005	13
6	F006	14
7	F007	15
8	F008	15
9	F009	25
10	F010	26
11	F011	27
12	F012	28

No.	Feature	Size
13	F013	13
14	F014	5
15	F015	8
16	F016	8
17	F017	13
18	F018	13
19	F019	3
20	F020	5
21	F021	8
22	F022	3
23	F023	3
24	F024	13

Group #1 features (light blue): F001, F002, F003, F004, F005, F006, F007, F008, F009, F010, F011, F012, F013, F014, F015, F016, F017, F018, F019, F020, F021, F022, F023, F024. Total: 140 [sp].

Group #2 features (light red): F013, F014, F015, F016, F017, F018, F019, F020, F021, F022, F023, F024. Total: 80 [sp].

ex. 220 [sp]

# Procedure for Network Construction Process in “Agile” CCPM (cont.)

2016 TOCICO International Conference

When the feature list consists of different types of features (requires different skills to execute)

Calculate the Duration of the initial task bar for each group by dividing the Size by the Initial Planned Velocity



ex. **Group #1** : 140 [sp] / 2 [sp/d] = 70 [d]



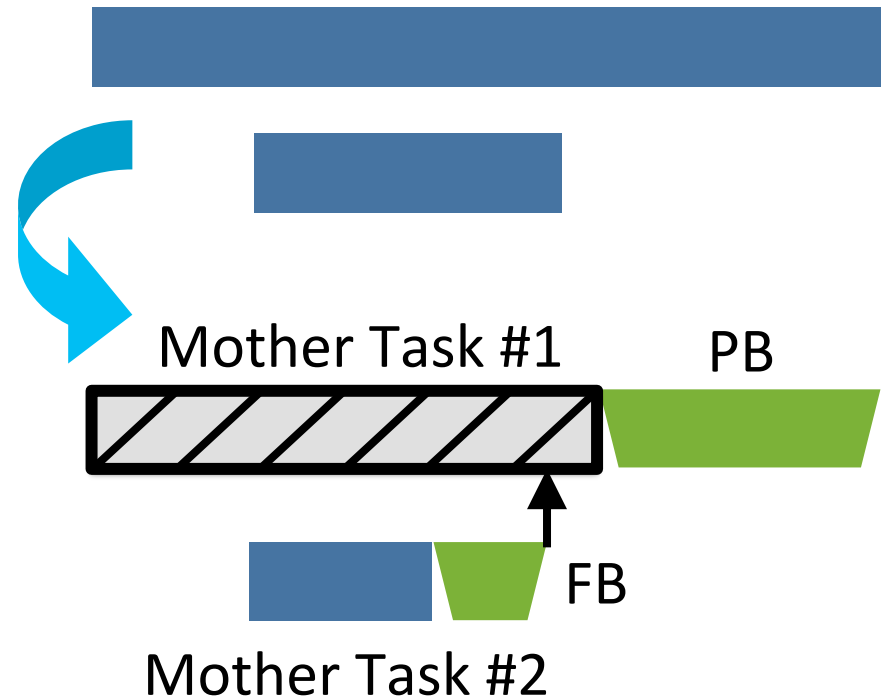
ex. **Group #2** : 80 [sp] / 2 [sp/d] = 40 [d]

# Procedure for Network Construction Process in “Agile” CCPM (cont.)

2016 TOCICO International Conference

When the feature list consists of different types of features (requires different skills to execute)

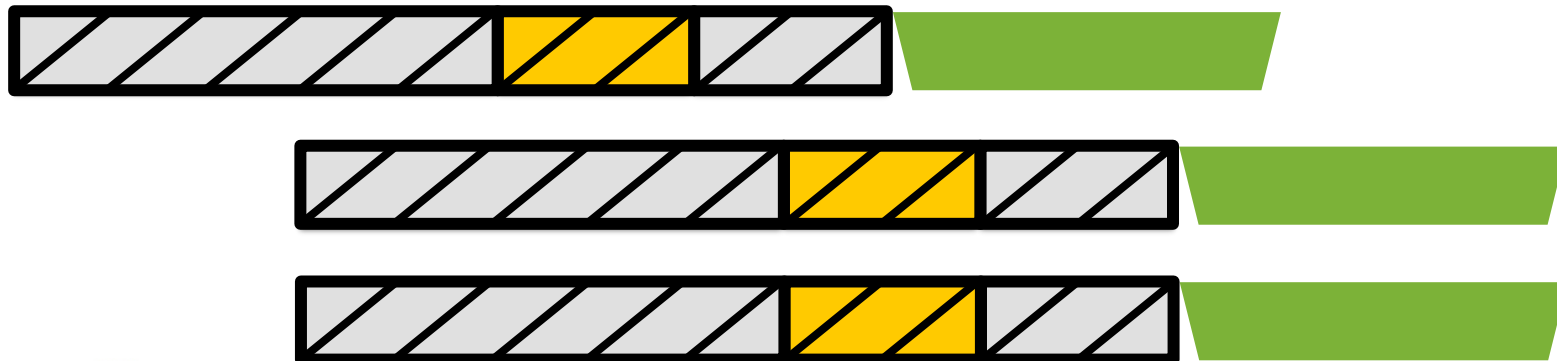
Convert the initial task bars to a Critical Chain schedule consisting of Mother Tasks and Buffers



# Procedure for Network Construction Process in “Agile” CCPM (cont.)

2016 TOCICO International Conference

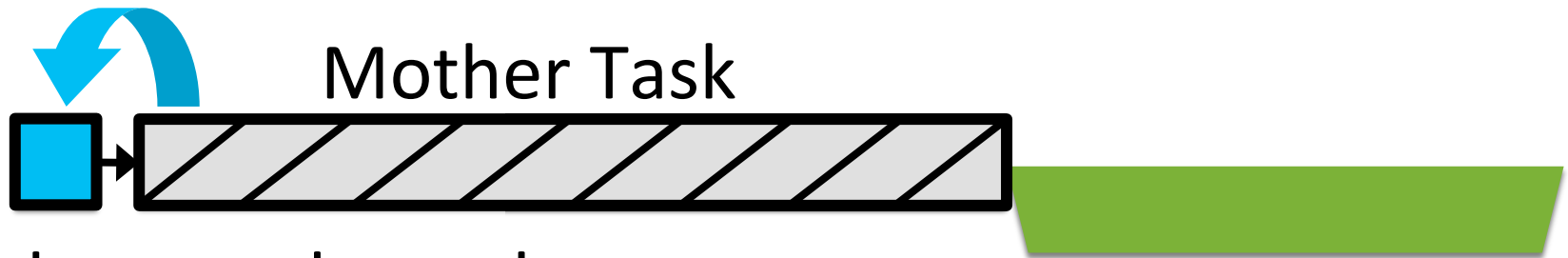
When in a multi-project environment, an integration phase is specified and staggering is performed in the same way as with standard CCPM



# Procedure for Network Construction Process in “Agile” CCPM (cont.)

2016 TOCICO International Conference

Tasks representing each feature(\*) will be broken out from the Mother Task(\*\*) and will be executed during project execution



Task to execute next

\*This doesn't necessarily mean tasks and features always have a one-for-one relationship

\*\*Technically, VBE and VBBM can be accomplished with a single Mother Task, without feature breakdown through project execution

# Tips and Examples for Network Construction Process

2016 TOCICO International Conference

A feature list is prepared

The initial task bar is created from the feature list

The initial task bar is converted to a Critical Chain schedule which consist of mother task(s) and buffer(s)

A next task is broken out from the mother task

Feature List

No.	Feature	Size
1	F001	8
2	F002	5
3	F003	8
4	F004	
5	F005	13
6	F006	14
7	F007	15
8	F008	
9	F009	25
10	F010	26
11	F011	27
12	F012	28

No.	Feature	Size
13	F013	13
14	F014	5
15	F015	8
16		
17		
18		
19		
20		
21		
22		
23		
24		
25	F025	13
26	F026	3
27	F027	5
28	F028	8
29	F029	3
30	F030	13

Initial Task Bar



Mother Task



Next Task

rights Reserved.

# Tips and Examples for Network Construction Process

2016 TOCICO International Conference

A feature list is prepared

Feature List

No.	Feature	Size
1	F001	8
2	F002	5
3	F003	8
4	F004	8
5	F005	13
6	F006	14
7	F007	15
8	F008	25
9	F009	26
10	F010	27
11	F011	28
12	F012	29

No.	Feature	Size
13	F013	13
14	F014	5
15	F015	8
16	F016	13
17	F017	3
18	F018	5
19	F019	8
20	F020	3
21	F021	13

The initial task bar is created from the feature list

Tip #1: How to Make Planning Poker Successful

A next task is broken out from the mother task

Initial Task Bar



Mother Task



Next Task

# Tips and Examples for Network Construction Process

2016 TOCICO International Conference

## Tip #2: Buffer Sizing and Velocity Based Estimation in “Agile” CCPM

The initial task bar is converted to a Critical Chain schedule which consist of mother task(s) and buffer(s)

A next task is broken out from the mother task

10	F010	26	F026	3
11	F011	27	F027	5
12	F012	28	F028	8
		29	F029	3
		30	F030	13

Initial Task Bar



Mother Task



Next Task

rights Reserved.



# Tips and Examples for Network Construction Process

2016 TOCICO International Conference

A feature list is prepared

Feature List

No.	Feature	Size
1	F001	8
2	F002	5

**Tip #3: Available Resources and Velocity Based Estimation**

The  
fro

The

converted to a Critical Chain schedule which consist of mother task(s) and buffer(s)

A next task is broken out from the mother task

Initial Task Bar



Mother Task



Next Task

rights Reserved.

# Tips and Examples for Network Construction Process

2016 TOCICO International Conference

A feature list is prepared

Feature List

No.	Feature	Size
1	F001	8

## Example #1: How Tasks are Broken Out from the Mother Task

converted to a Critical Chain schedule which consist of mother task(s) and buffer(s)

Initial Task Bar



Mother Task



A next task is broken out from the mother task

Next Task

rights Reserved.

# Tips and Examples for Network Construction Process

2016 TOCICO International Conference

A feature list is prepared

Feature List

No.	Feature	Size
1	F001	8
2	F002	5
3	F003	8
4	F004	8
5	F005	13
6	F006	14
7	F007	15
8	F008	25
9	F009	26
10	F010	27
11	F011	28
12	F012	29

No.	Feature	Size
13	F013	13
14	F014	5
15	F015	8
16	F016	13
17	F017	3
18	F018	5
19	F019	8
20	F020	3
21	F021	13

The initial task bar is created from the feature list

**Tip #1: How to Make Planning Poker Successful**

A next task is broken out from the mother task

Initial Task Bar



Mother Task



Next Task

rights Reserved.

# Tip #1: How to Make Planning Poker Successful

2016 TOCICO International Conference

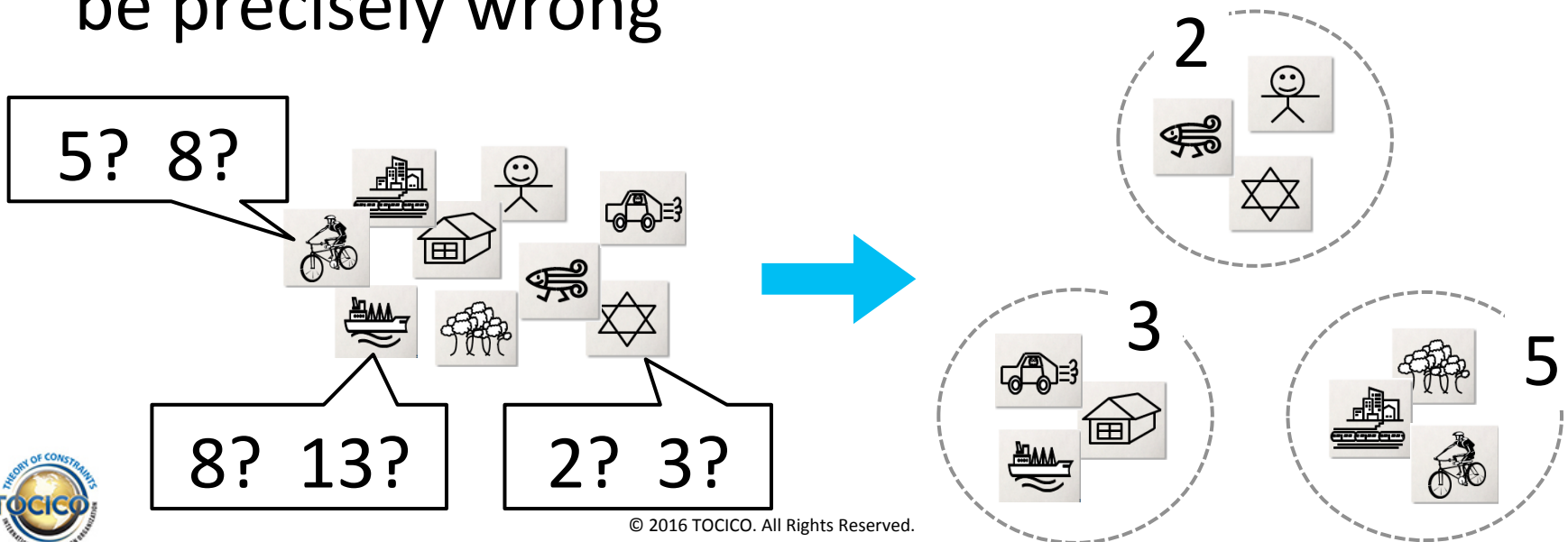
- Planning poker is a practical and consensus-based technique for estimating the relative size of features, while intuition among members is fully utilized.
  1. Estimate by playing numbered cards face-down on the table, instead of speaking them aloud
  2. The cards are revealed, and the estimates are then discussed



# Tip #1: How to Make Planning Poker Successful (cont.)

2016 TOCICO International Conference

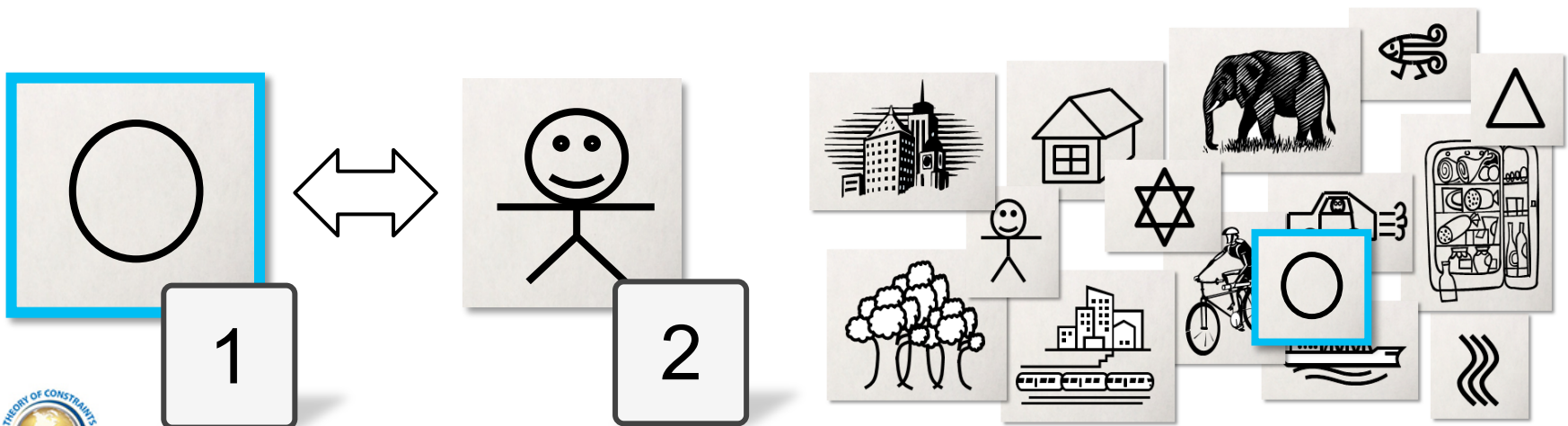
- Play with a feeling that you are creating groups of features by Relative Size, instead of estimating each feature as a specific numerical value
- “it is better to be approximately right than to be precisely wrong”



# Tip #1: How to Make Planning Poker Successful (cont.)

2016 TOCICO International Conference

- If it is your first time and you are struggling to use Planning Poker, it is recommended you:
  1. Find an easy enough feature and set its size as “1”
  2. Estimate remaining features vs. easier ones, by comparison to the first/preceding features

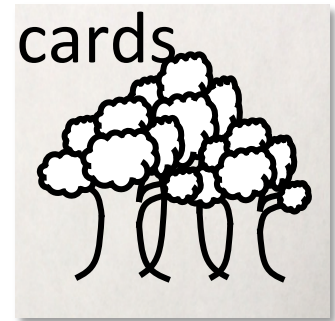
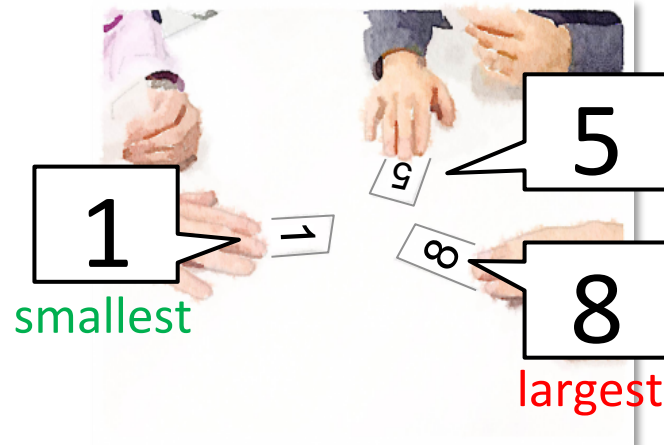


# Tip #1: How to Make Planning Poker Successful (cont.)

2016 TOCICO International Conference

- When there is a big gap in estimation among members, the following can help you find out what is causing the gap
  1. Find both the smallest and the largest cards
  2. Ask for a discussion between the players who played the smallest and the largest cards

“Since, it seems that most of parts can be duplicated easily...”



“Because there are many parts...”



# Tips and Examples for Network Construction Process

2016 TOCICO International Conference

## Tip #2: Buffer Sizing and Velocity Based Estimation in “Agile” CCPM

The initial task bar is converted to a Critical Chain schedule which consist of mother task(s) and buffer(s)

A next task is broken out from the mother task

10	F010	26	F026	3
11	F011	27	F027	5
12	F012	28	F028	8
		29	F029	3
		30	F030	13

Initial Task Bar



Mother Task



Next Task

rights Reserved.



# Tip #2: Buffer Sizing and Velocity Based Estimation in “Agile” CCPM

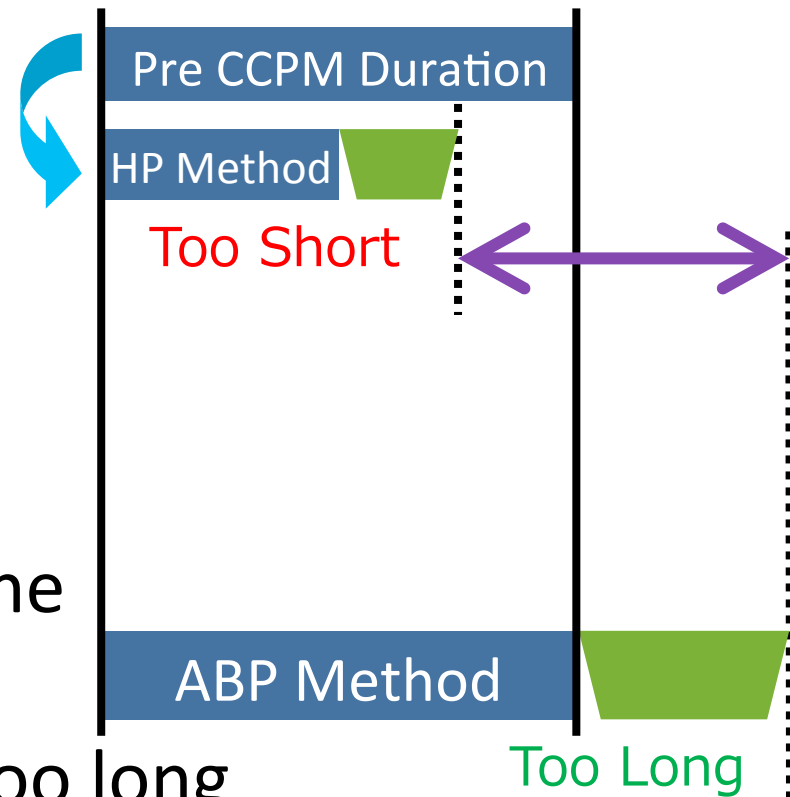
2016 TOCICO International Conference

- In “Agile” CCPM, task durations are recognized as something in between “Aggressive But Possible”(ABP : 50% chance) and “Highly Possible”(HP : 90% chance)
- They are calculated by dividing Story Points of completed features by the Actual Velocity obtained from the past results, including both best and worst cases

# Tip #2: Buffer Sizing and Velocity Based Estimation in “Agile” CCPM (cont.)

2016 TOCICO International Conference

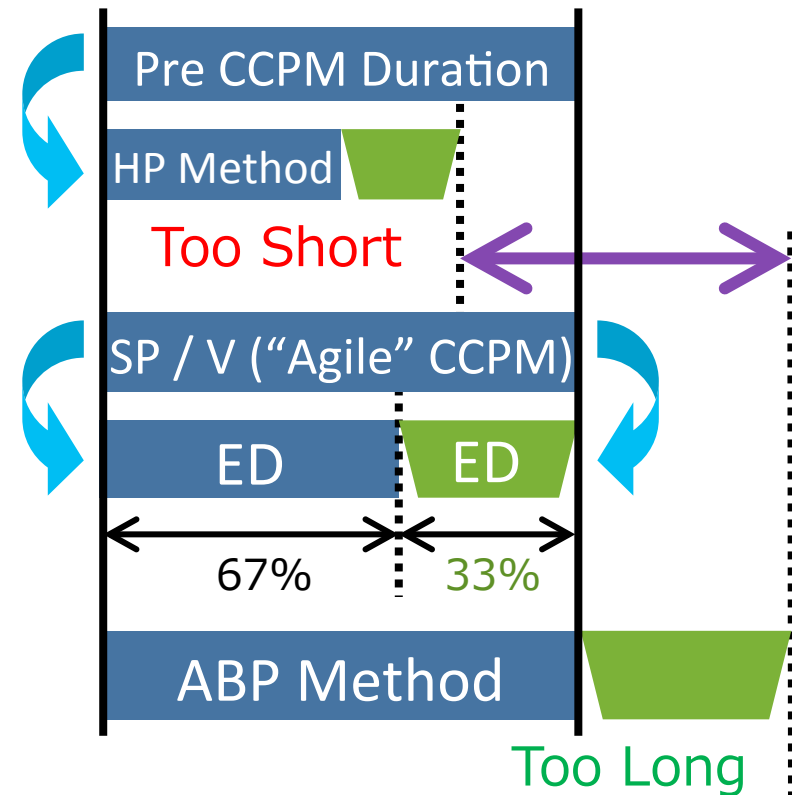
- Our experience has shown that if task duration is treated by the “HP Method”, the resulting project period estimates are too short
- If task duration is treated by the “ABP Method”, the resulting project period estimates are too long
- Due dates need to be calculated in-between, by a certain simple method



# Tip #2: Buffer Sizing and Velocity Based Estimation in “Agile” CCPM (cont.)

2016 TOCICO International Conference

- The “1/3 rule” is applied by default for buffer sizing and the Velocity Based Estimation
- This can be done using the following formulas [33% is set to “Cb” (Buffer Coefficient)]
  - Buffer Part :  $ED = SP / V \times Cb$   
 $= SP / V \times 33\%$
  - Leg Part :  $ED = SP / V \times (1 - Cb)$   
 $= SP / V \times 67\%$



\*To simplify the explanation, a short version of the formula “ $ED = SP / V$ ” (Cb is omitted) is applied for the Velocity Based Estimation in this presentation

# Tips and Examples for Network Construction Process

2016 TOCICO International Conference

A feature list is prepared

Feature List

No.	Feature	Size
1	F001	8
2	F002	5

**Tip #3: Available Resources and Velocity Based Estimation**

The  
from

The  
converted to a Critical Chain  
schedule which consist of  
mother task(s) and buffer(s)

A next task is broken out  
from the mother task

Initial Task Bar



Mother Task



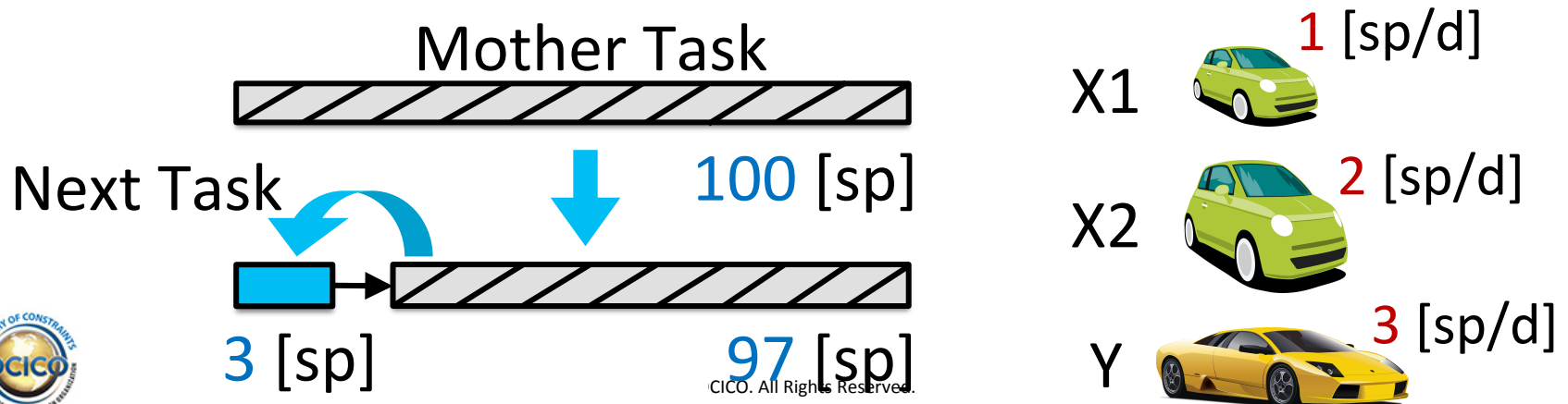
Next Task

rights Reserved.

# Tip #3: Available Resources and Velocity Based Estimation

2016 TOCICO International Conference

- Initially, there is a single Mother Task (100 [sp])
- There are 3 resources “X1”, “X2” and “Y” available for this project
- A task (3 [sp]) is broken out from the Mother Task and the size of the Mother Task is decreased to 97 [sp] as a result



# Tip #3: Available Resources and Velocity Based Estimation - Case #1

2016 TOCICO International Conference

Here only “X1” <sup>1</sup> is available



$$ED = 3 / 1 = 3 \text{ [d]}$$



$$ED = 97 / 1 = 97 \text{ [d]}$$

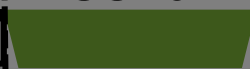
Next Task



Mother Task

← 67% → ← 33% →

“1/3 rule”



$$\frac{3}{1} \times \frac{67\%}{1-C_b} = 2 \text{ [d]}$$

$$\frac{97}{1} \times \frac{67\%}{1-C_b} \approx 65 \text{ [d]}$$

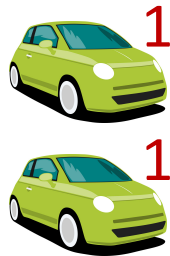
\*To be precise, the formula should be “ED = SP / V x (1 – Cb)”, but the short version “ED = SP / V” is applied in this presentation

# Tip #3: Available Resources and Velocity Based Estimation - Case #2

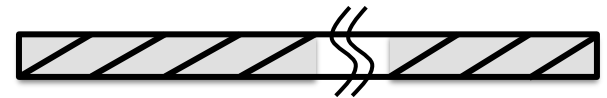
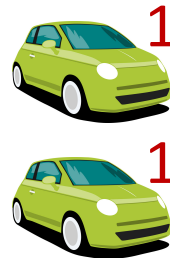
2016 TOCICO International Conference

Here two “X1”s   are available

Next Task  →  Mother Task



$$ED = 3 / 2 = 1.5 \text{ [d]}$$





$$ED = 97 / 2 = 48.5 \text{ [d]}$$

# Tip #3: Available Resources and Velocity Based Estimation - Case #3

2016 TOCICO International Conference



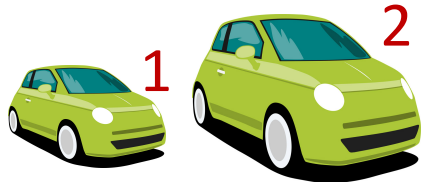
$$ED = 3 / 2 = 1.5 \text{ [d]}$$

Here “X1”<sup>1</sup> and “X2”<sup>2</sup> are available (“X1” and “X2” are the same in skill type, and are different in velocity)

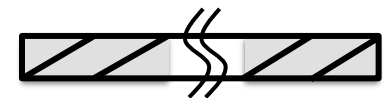
Next Task



Mother Task



$$ED = 3 / 3 = 1 \text{ [d]}$$



$$ED = 97 / 3 \approx 32 \text{ [d]}$$



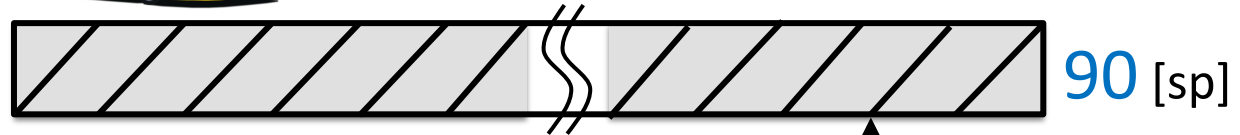
# Tip #3: Available Resources and Velocity Based Estimation - Case #4

2016 TOCICO International Conference

Here “X1” <sup>1</sup> and “Y” <sup>3</sup> are available

Velocity of “Y” : 3 [sp/d]

Resource “Y”



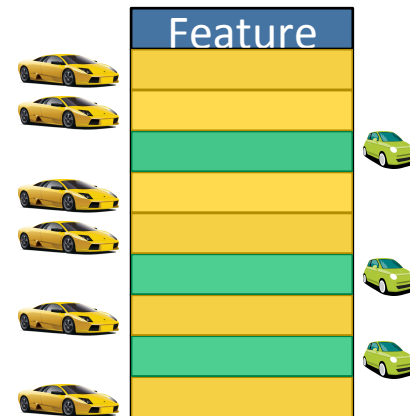
Mother Task for “Y”

Velocity of “X1” : 1 [sp/d]

Resource “X1”



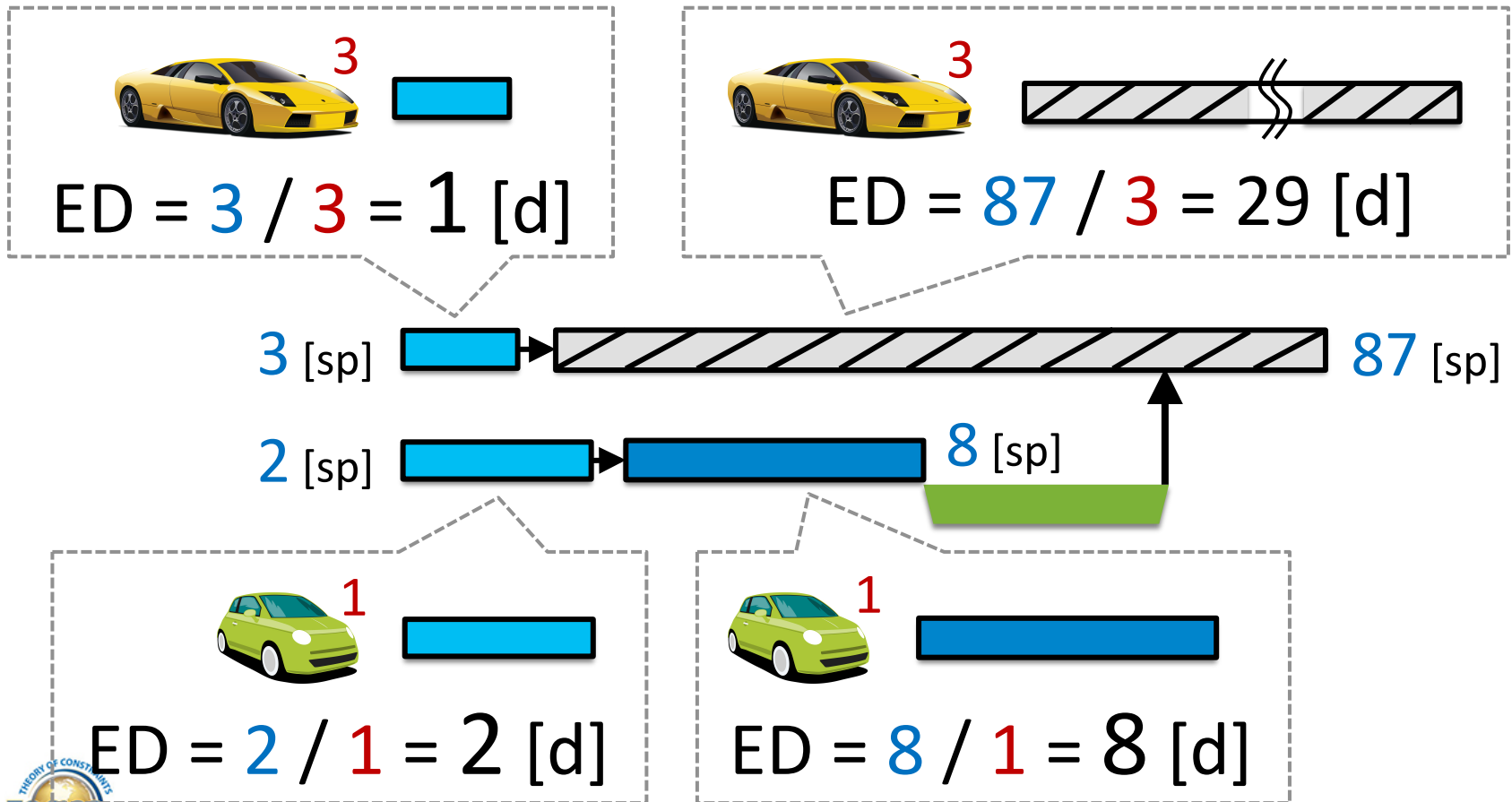
Mother Task for “X1”



# Tip #3: Available Resources and Velocity Based Estimation - Case #4 (cont.)

2016 TOCICO International Conference

Here “X1” and “Y” are available



# Tip #3: Available Resources and Velocity Based Estimation - All Cases on a Page

2016 TOCICO International Conference

## Case #1



Only a single "X1" is available

$$ED = 97 / 1 = 97 \text{ [d]}$$



$$ED = 3 / 1 = 3 \text{ [d]}$$

## Case #2



Two "X1"s are available

$$ED = 97 / 2 = 48.5 \text{ [d]}$$



$$ED = 3 / 2 = 1.5 \text{ [d]}$$

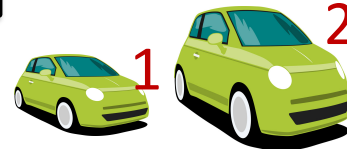
## Case #3



"X1" and "X2"  
are available

$$ED = 3 / 2 = 1.5 \text{ [d]}$$

$$ED = 97 / 3 \approx 32 \text{ [d]}$$



$$ED = 3 / 3 = 1 \text{ [d]}$$

## Case #4



"X1" and "Y" are available

$$ED = 87 / 3 = 29 \text{ [d]}$$



$$ED = 3 / 3 = 1 \text{ [d]}$$

$$ED = 8 / 1 = 8 \text{ [d]}$$



$$ED = 2 / 1 = 2 \text{ [d]}$$



# Tips and Examples for Network Construction Process

2016 TOCICO International Conference

A feature list is prepared

Feature List

No.	Feature	Size
1	F001	8

## Example #1: How Tasks are Broken Out from the Mother Task

converted to a Critical Chain schedule which consist of mother task(s) and buffer(s)

Initial Task Bar



Mother Task



A next task is broken out from the mother task

Next Task

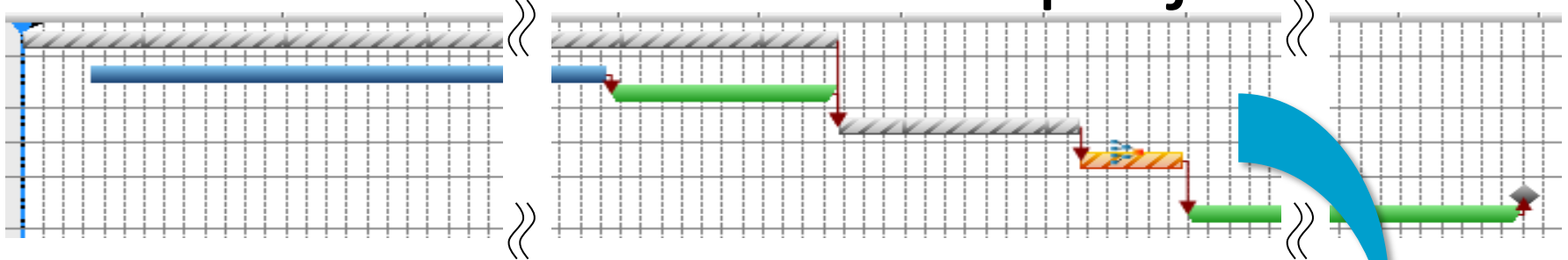


rights Reserved.

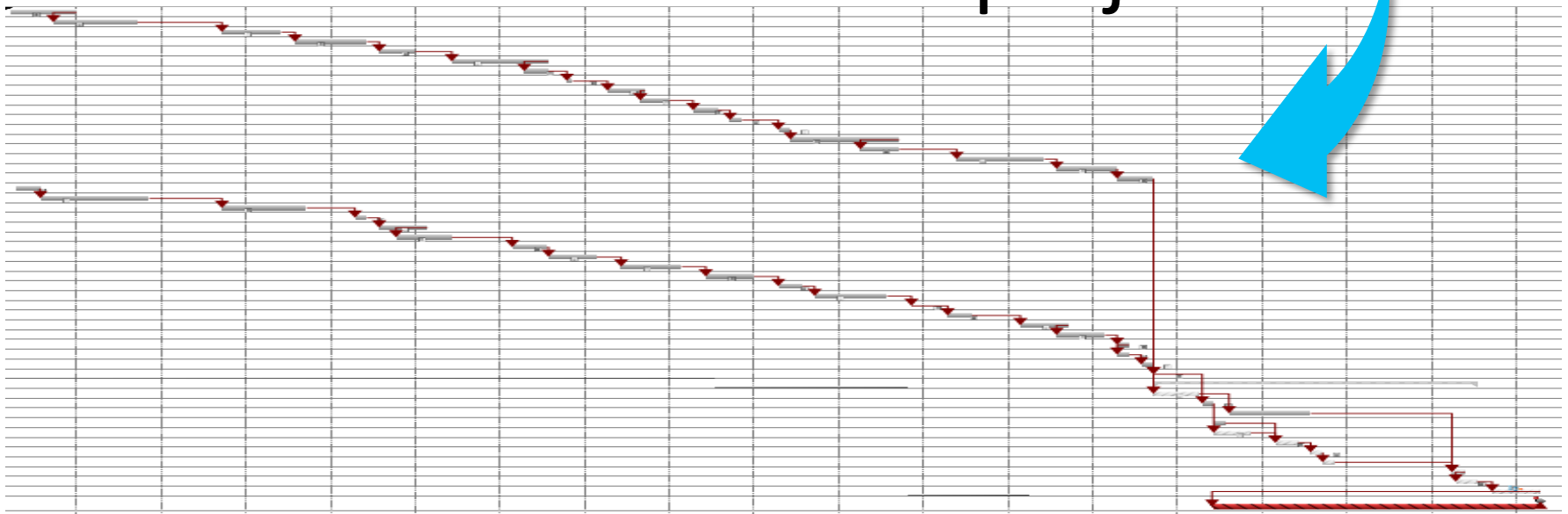
# Example #1: How Tasks are Broken Out from the Mother Task

2016 TOCICO International Conference

At the kick-off of the project

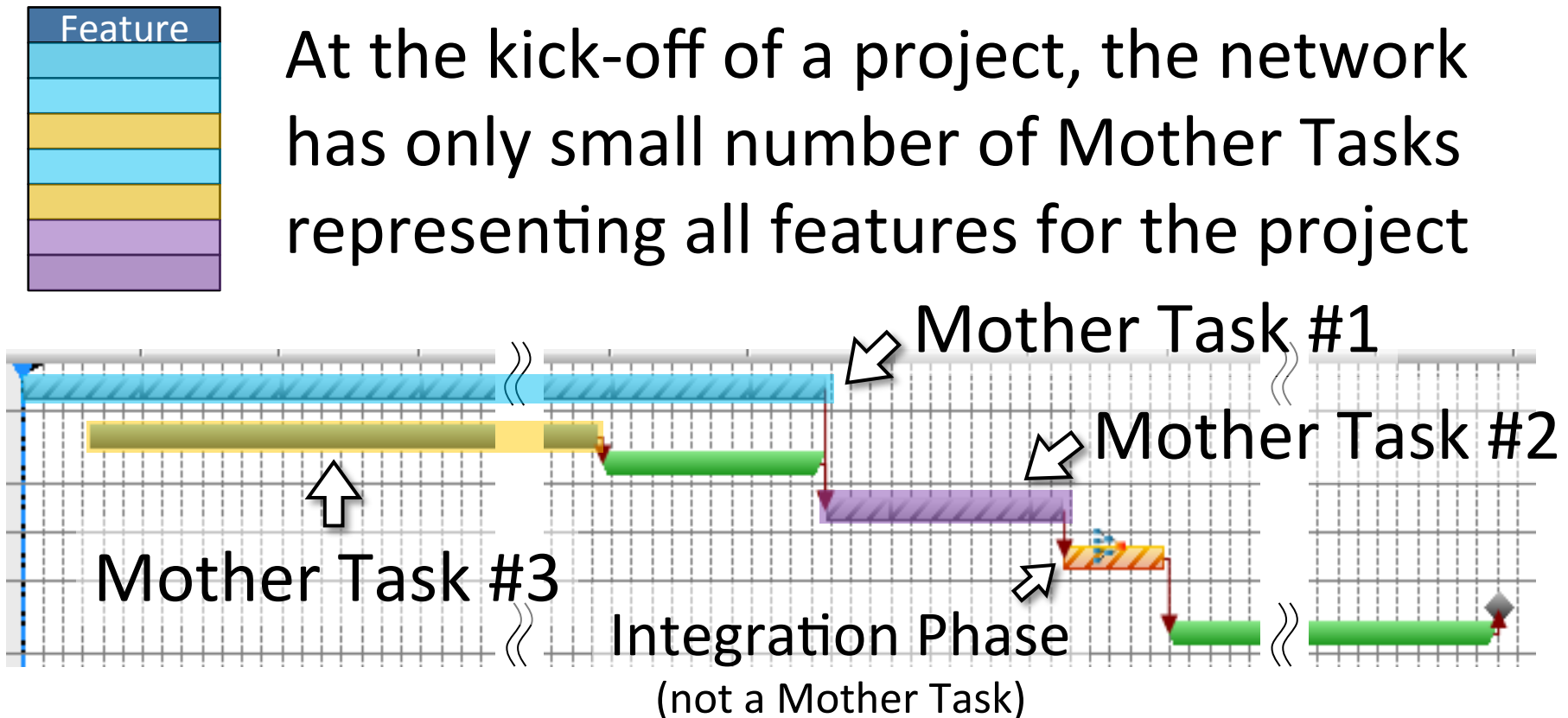


At the end of the project



# Example #1: How Tasks are Broken Out from the Mother Task (cont.)

2016 TOCICO International Conference



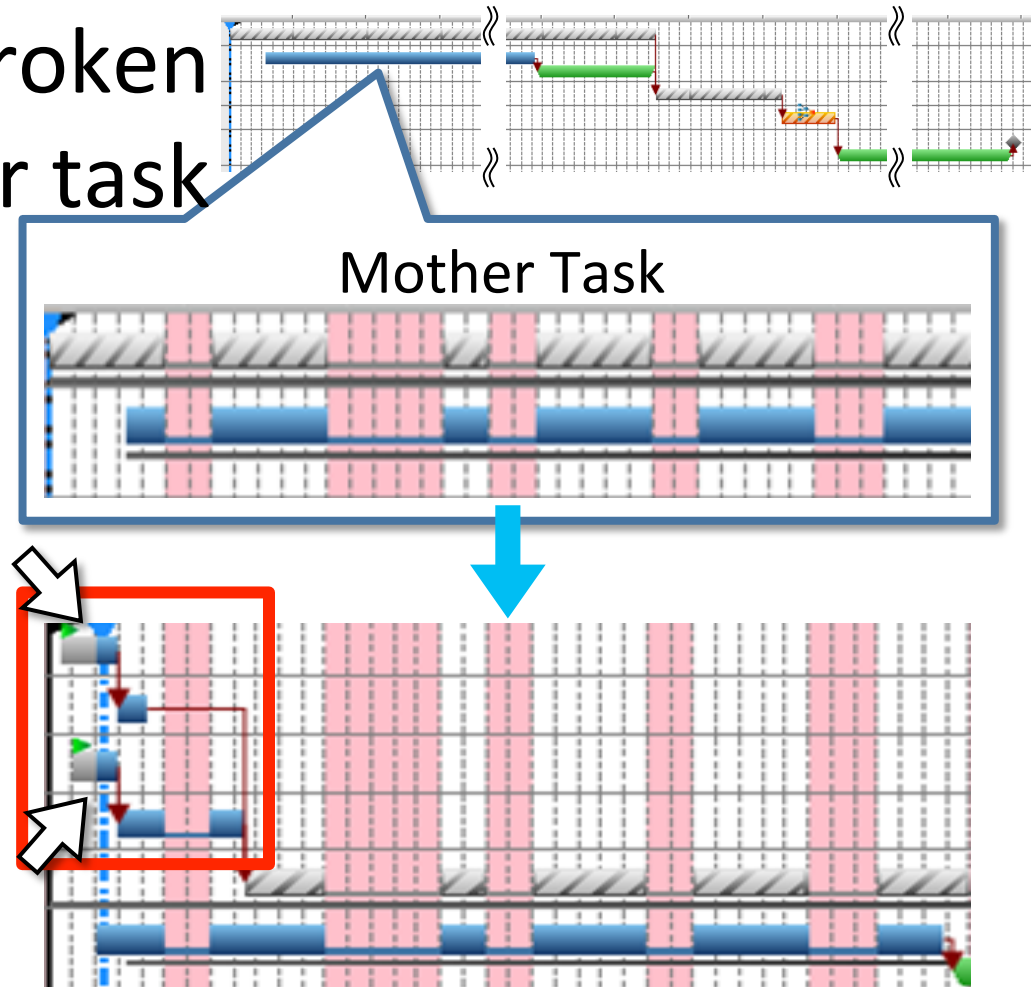
The integration phase is specified so that the project can be staggered in the pipeline by the Virtual Drum, but is not considered a Mother Task in our case

# Example #1: How Tasks are Broken Out from the Mother Task (cont.)

2016 TOCICO International Conference

Initially, no task is broken out from the mother task

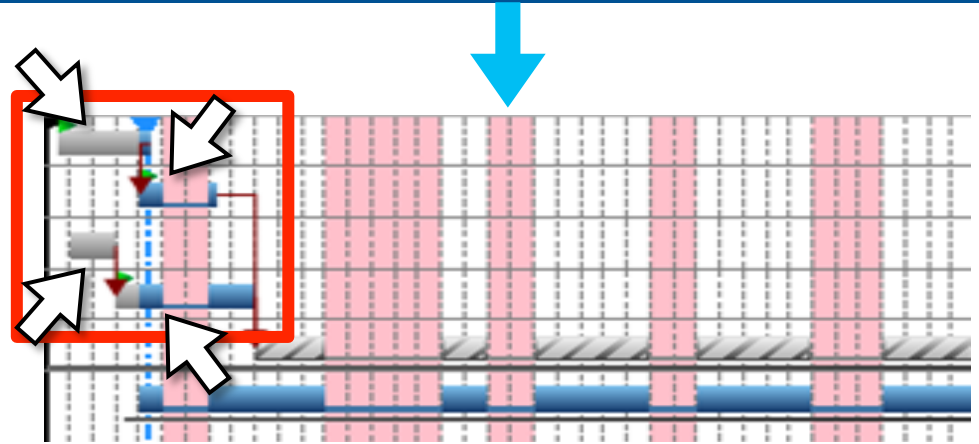
Below, 4 tasks are broken out from the mother task, the first 2 tasks are initiated, and are still in execution



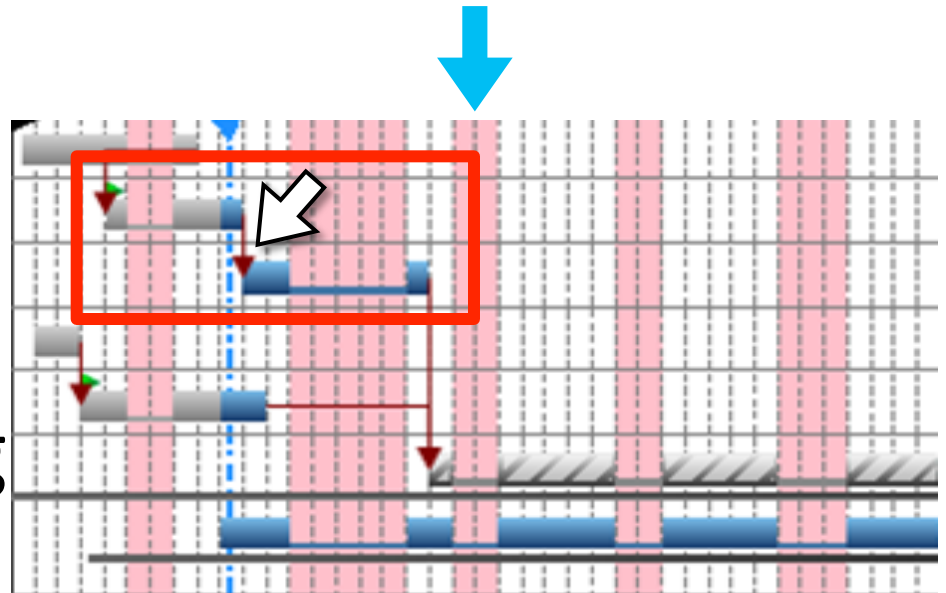
# Example #1: How Tasks are Broken Out from the Mother Task (cont.)

2016 TOCICO International Conference

The first 2 tasks are completed and the next 2 tasks are initiated



Another task is broken out from the mother task just before the preceding task is completed

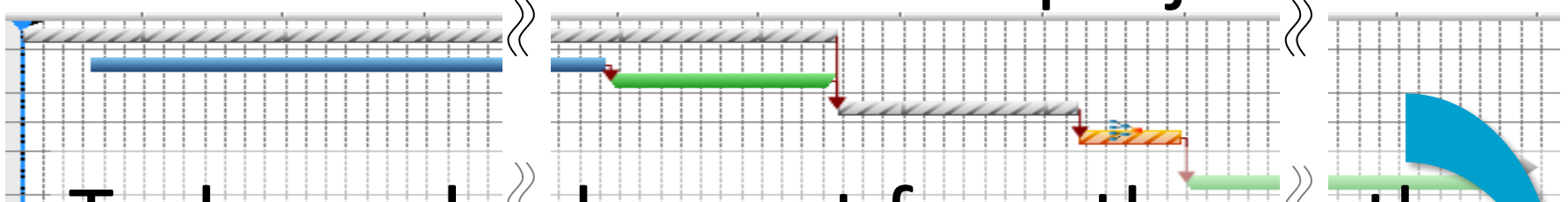




# Example #1: How Tasks are Broken Out from the Mother Task (cont.)

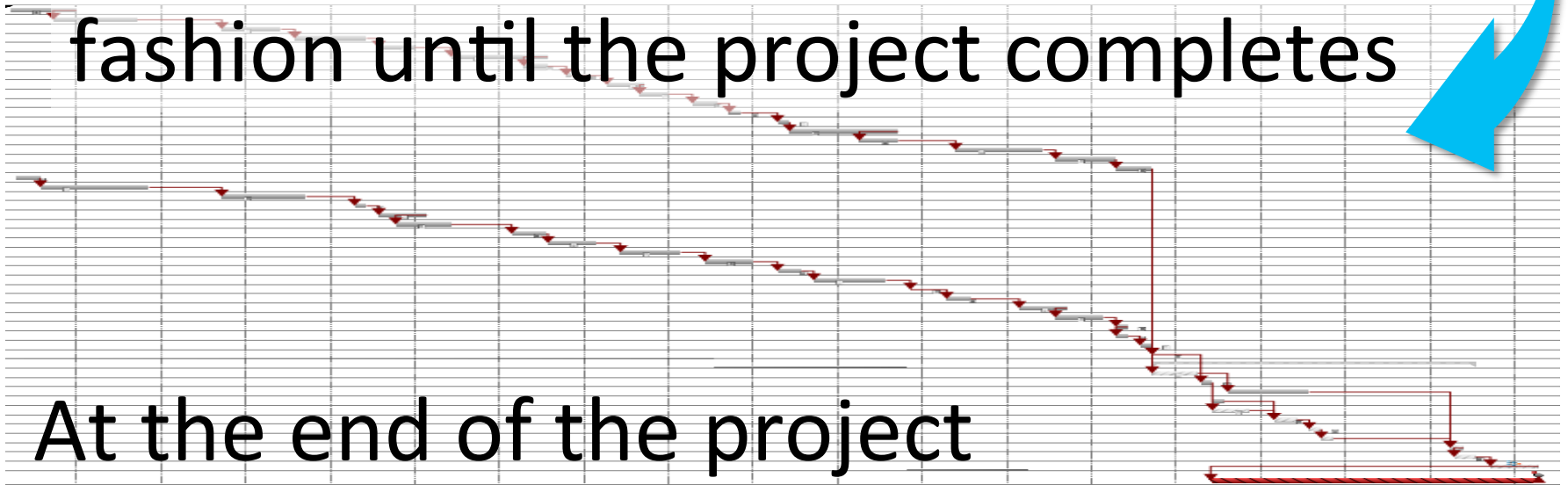
2016 TOCICO International Conference

At the kick-off of the project



Tasks are broken out from the mother task and are executed in a similar fashion until the project completes

At the end of the project



# Overview of “Agile” CCPM

2016 TOCICO International Conference

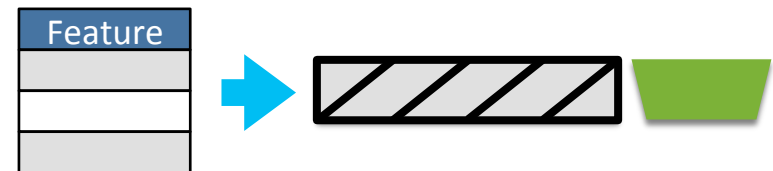
## CCPM

### “Standard”

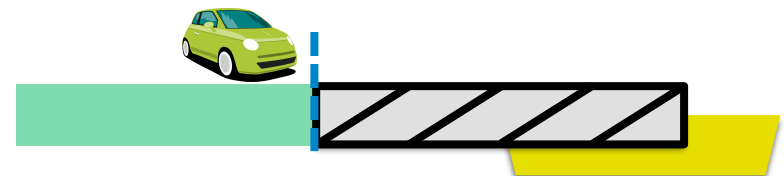
- Procedure in planning
  - Add tasks and dependencies
  - Level resources
  - Identify the Critical Chain
  - Insert buffers
- Procedure in execution
  - Execute tasks and report RDU
  - Update buffer status
  - Take recovery action as required

### “Agile”

#### Modified Network Construction Process



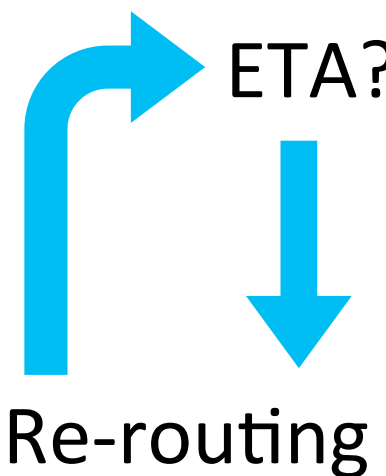
#### Modified Buffer Management Method



# Flow of the Modified Buffer Management Method

2016 TOCICO International Conference

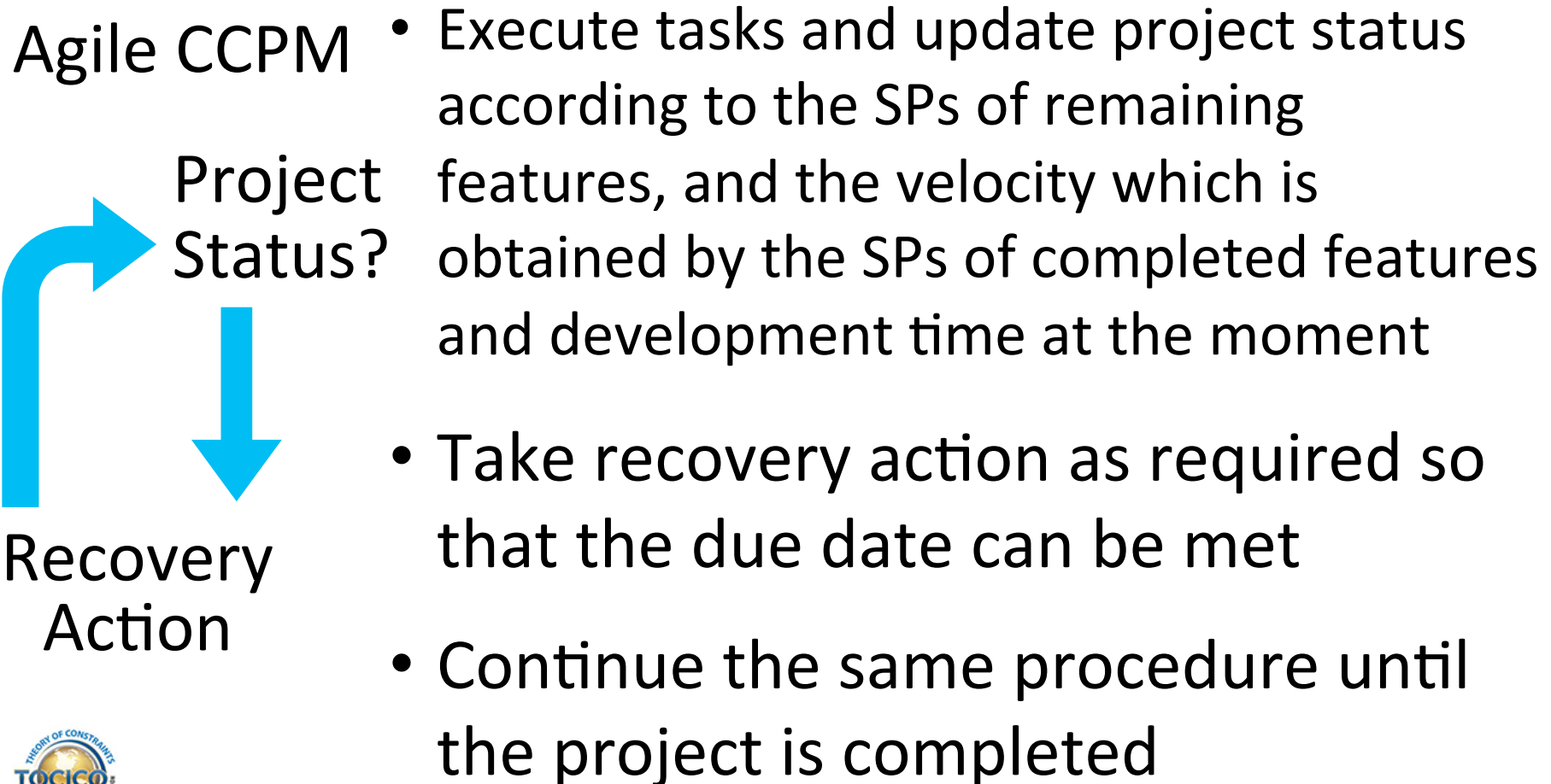
(The process is similar to a long drive)

- Car Drive
- 
- Estimate the arrival time according to the distance to the goal and the velocity which is obtained by travel distance and drive time at the moment
  - Change the route as required so that the initial ETA can be met
  - Continue the same procedure until the car will get to the goal

# Flow of the Modified Buffer Management Method (cont.)

2016 TOCICO International Conference

(The process is similar to a long drive)



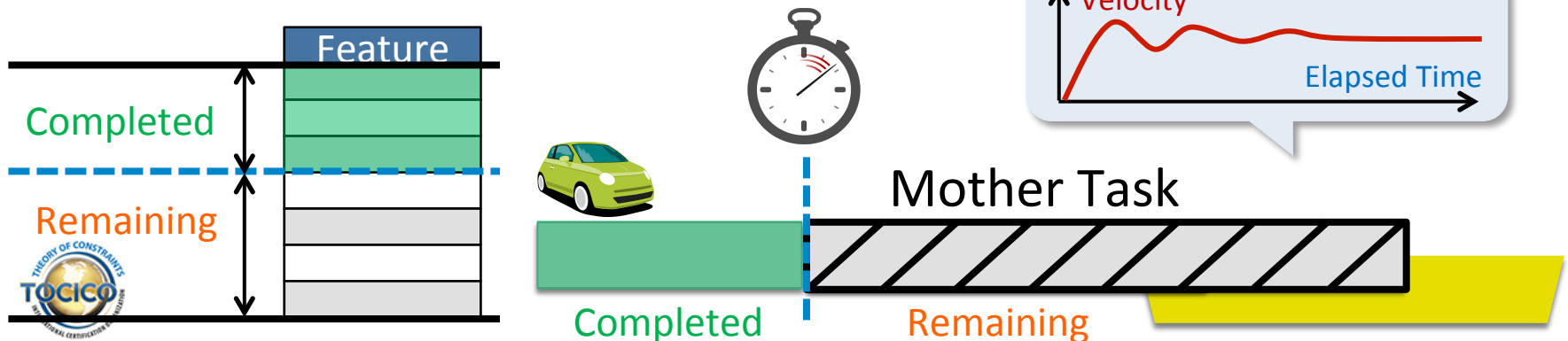
# Procedure for Buffer Management in “Agile” CCPM

2016 TOCICO International Conference

- The procedure is referred to as the “Velocity-Based Buffer Management”(VBBM)
- Estimated duration (ED) of the Mother Task is adjusted by the “Velocity Based Estimation”(VBE) according to the following formulas:

$$\text{Velocity} = \frac{\text{Completed SP}}{\text{Elapsed Time}}$$

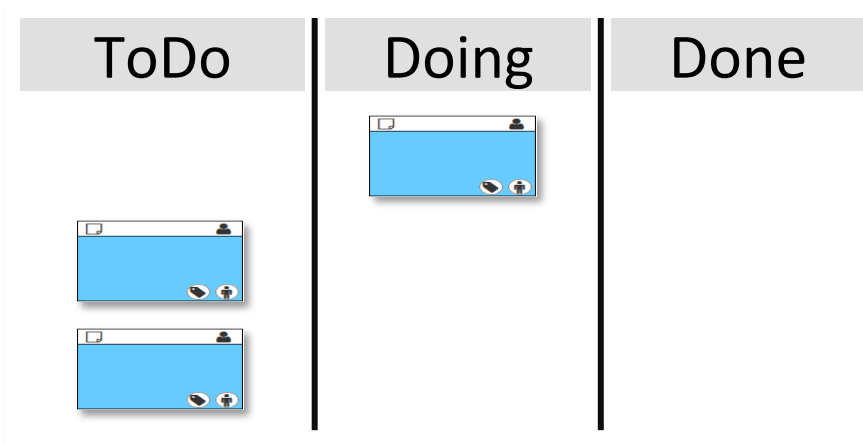
$$\text{ED} = \frac{\text{Remaining SP}}{\text{Velocity}}$$



# Procedure for Buffer Management in “Agile” CCPM (cont.)

2016 TOCICO International Conference

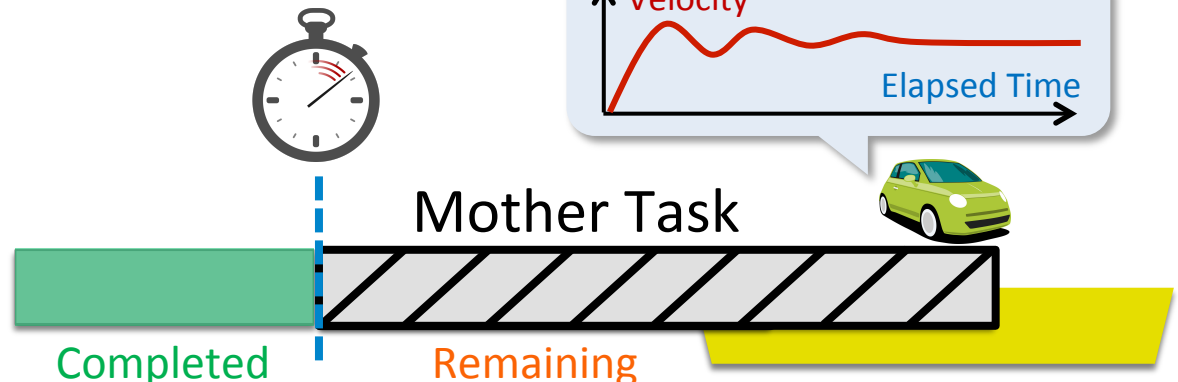
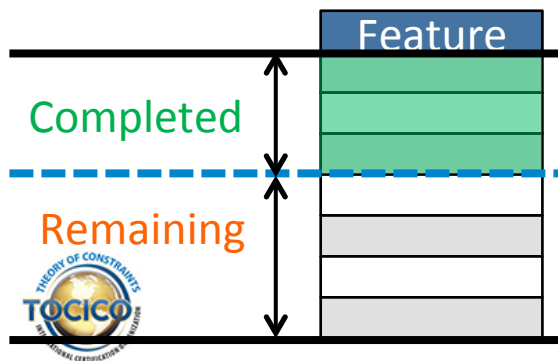
## Task Board



Let's take a look at the concept and the procedure through a Task Board so that we can see it more intuitively and simply

$$\text{Velocity} = \frac{\text{Completed SP}}{\text{Elapsed Time}}$$

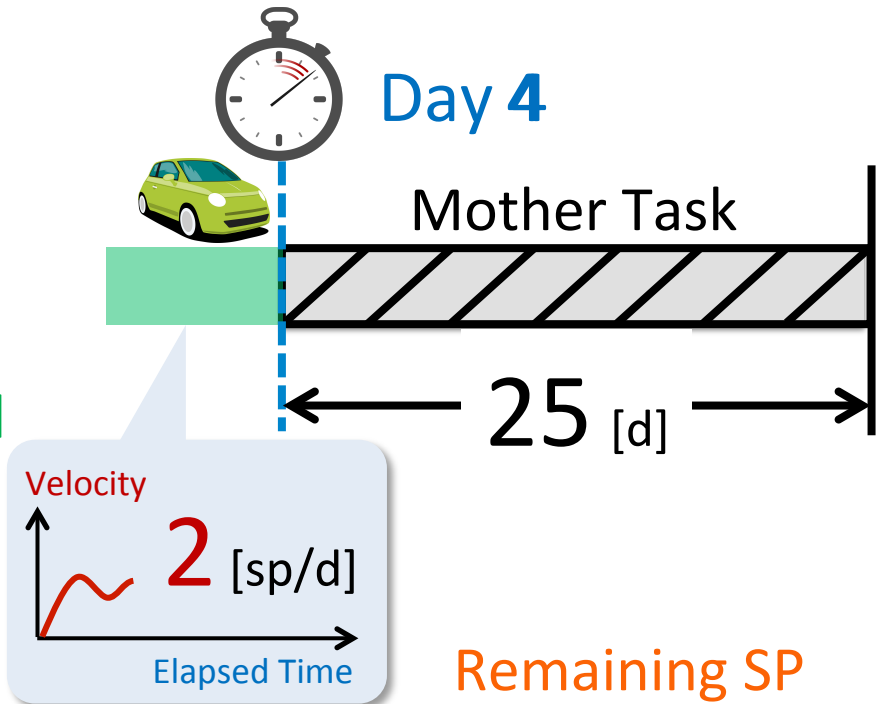
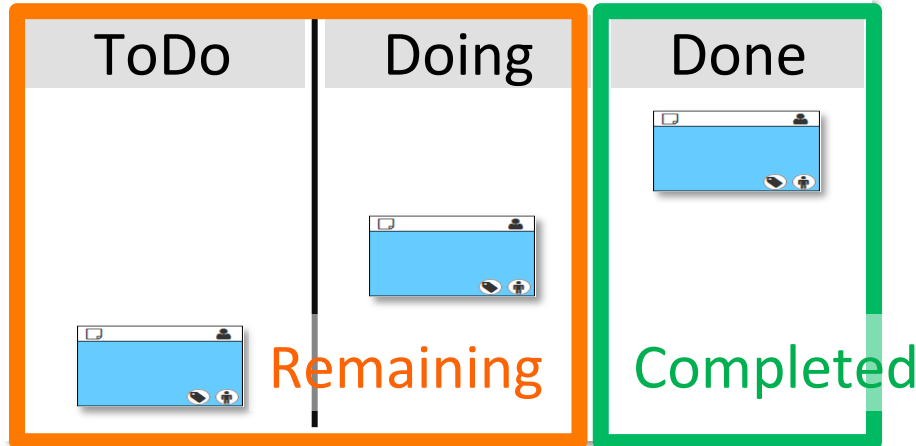
$$\text{ED} = \frac{\text{Remaining SP}}{\text{Velocity}}$$



# Procedure for Buffer Management in “Agile” CCPM (cont.)

2016 TOCICO International Conference

## Task Board



$$\text{Velocity} = \frac{8 \text{ [sp]}}{4 \text{ [d]}}$$

Completed SP

Elapsed Time

$$= 2 \text{ [sp/d]}$$

$$\text{ED} = \frac{50 \text{ [sp]}}{2 \text{ [sp/d]}}$$

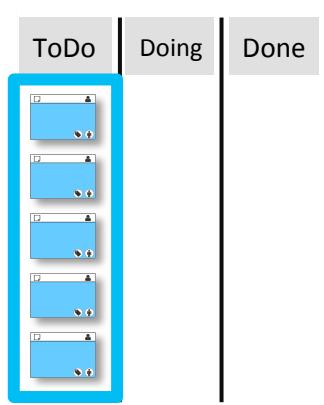
Remaining SP

Velocity

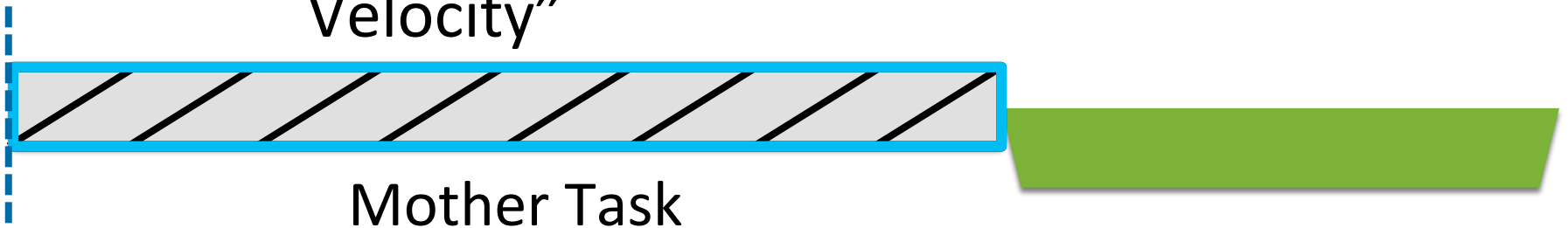
$$= 25 \text{ [d]}$$

# Procedure for Buffer Management in “Agile” CCPM (cont.)

2016 TOCICO International Conference



Initially, there is only a  
“Mother Task” representing all  
target features for the project.  
Its length is calculated based  
on the “Initial Planned  
Velocity”

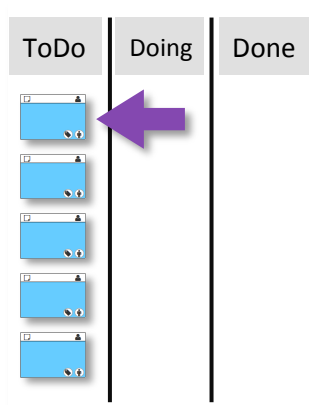


As-of Date



# Procedure for Buffer Management in “Agile” CCPM (cont.)

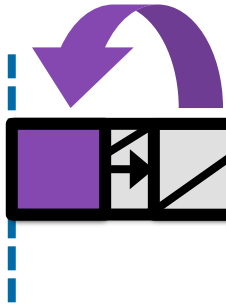
2016 TOCICO International Conference



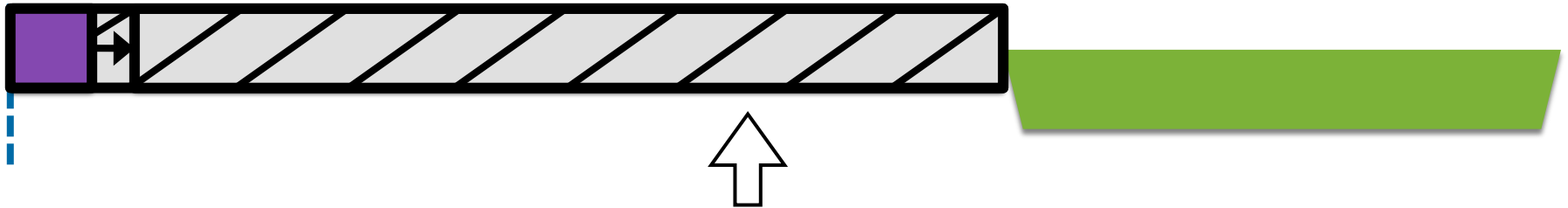
The 1st task (**Task 1**), corresponding to the highest priority feature is broken out from the Mother Task



Task 1



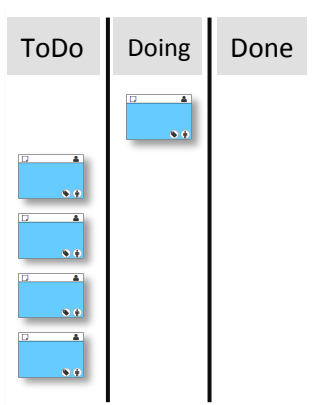
Mother Task



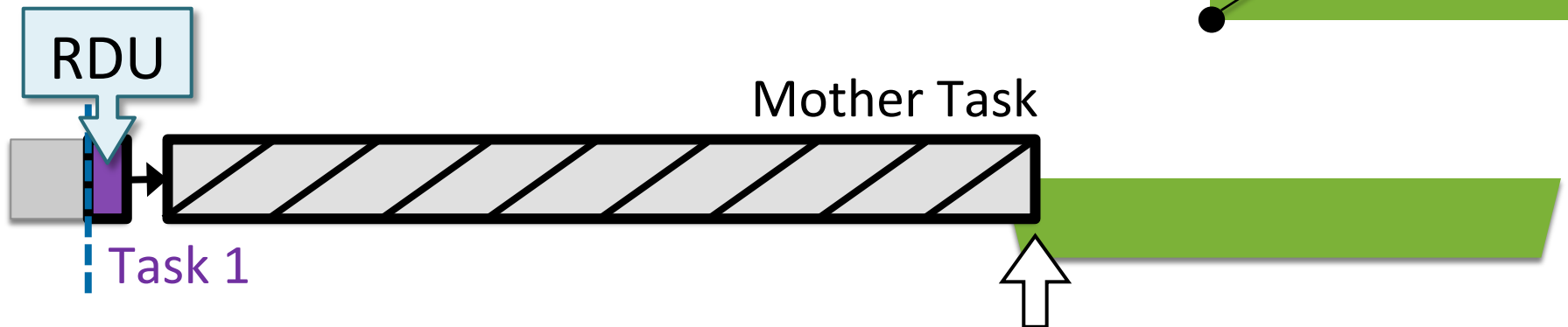
When a task is broken out, the duration of the Mother Task is adjusted as well

# Procedure for Buffer Management in “Agile” CCPM (cont.)

2016 TOCICO International Conference



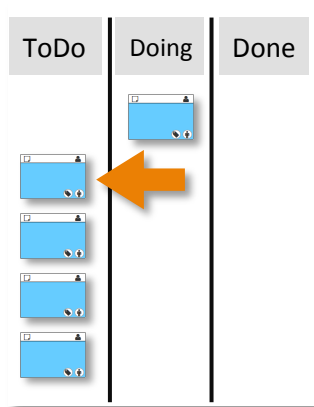
**Task 1** is started and remaining duration (RDU) is reported in the same way as in standard CCPM



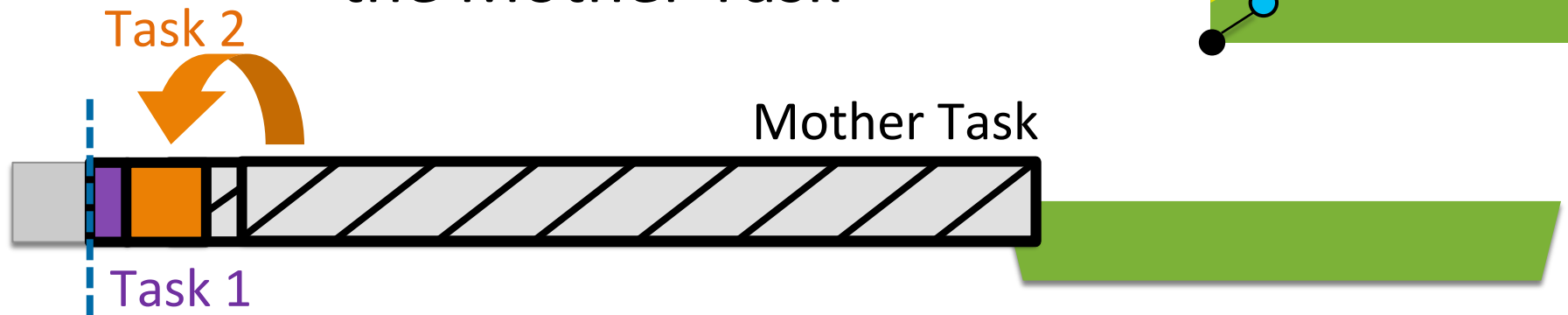
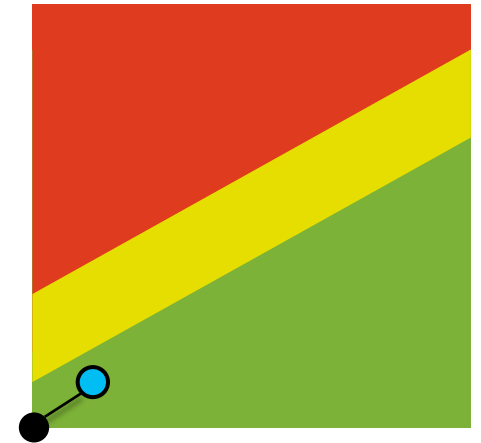
The project buffer is consumed in the same way as in standard CCPM

# Procedure for Buffer Management in “Agile” CCPM (cont.)

2016 TOCICO International Conference

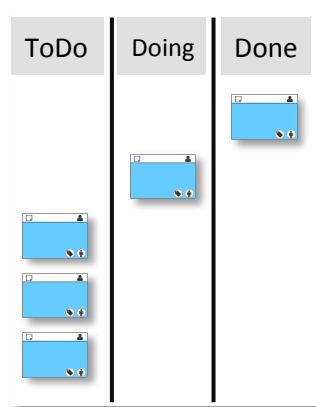


The 2nd task (**Task 2**) for the next highest priority feature is broken out from the Mother Task

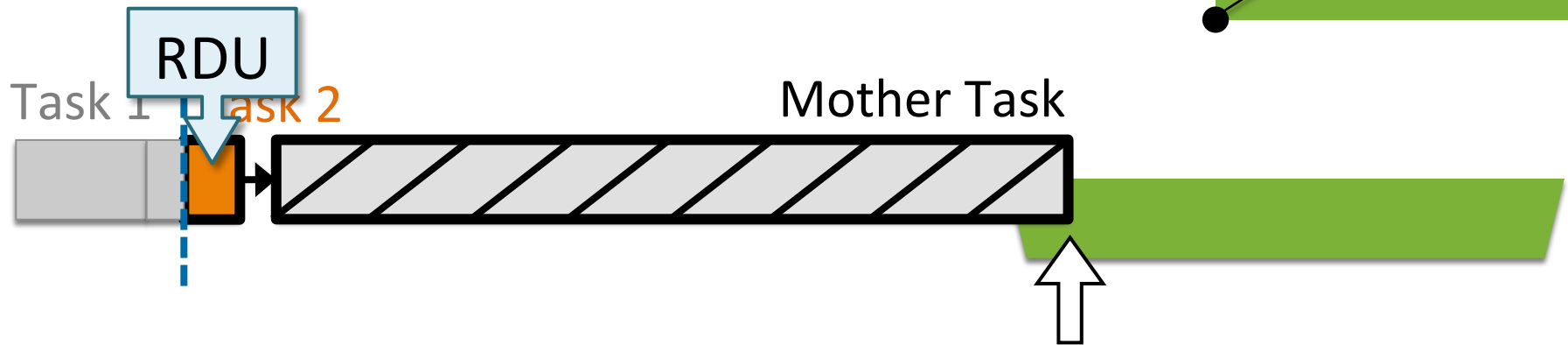
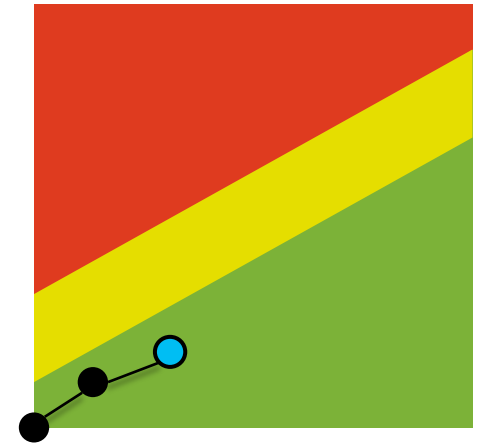


# Procedure for Buffer Management in “Agile” CCPM (cont.)

2016 TOCICO International Conference



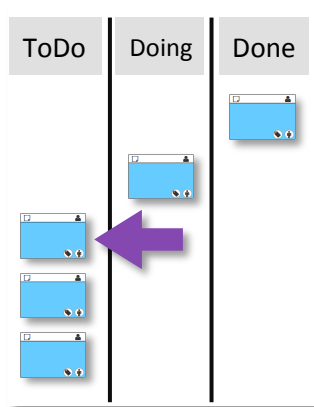
Task 2 is started after Task 1 is completed, and the RDU of Task 2 is reported



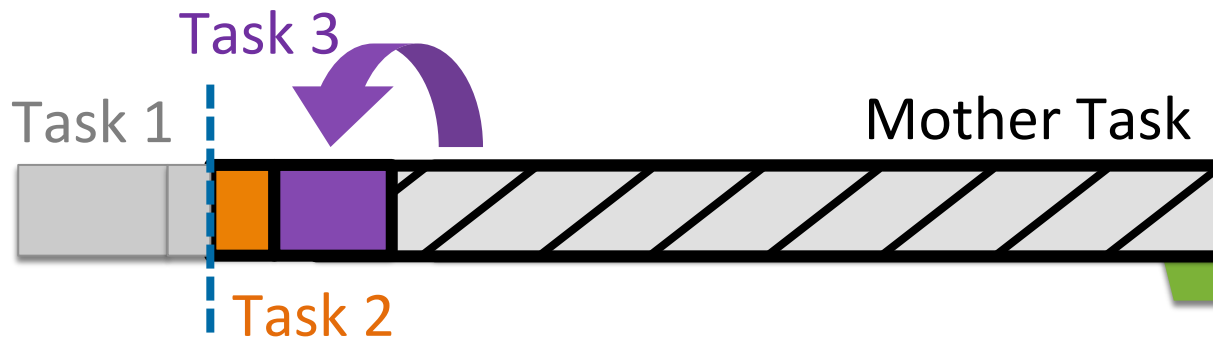
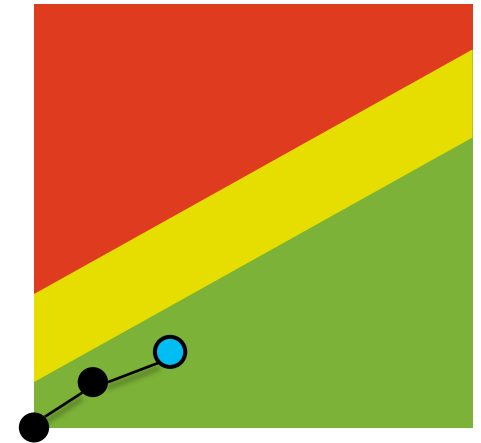
Project buffer is consumed

# Procedure for Buffer Management in “Agile” CCPM (cont.)

2016 TOCICO International Conference

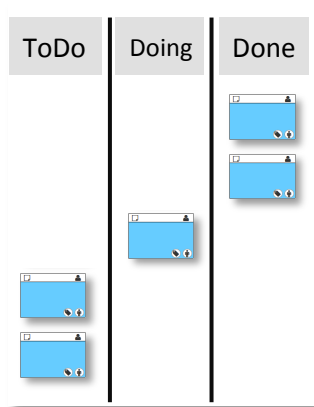


The 3rd task (**Task 3**) for the next highest priority feature is broken out from the Mother Task

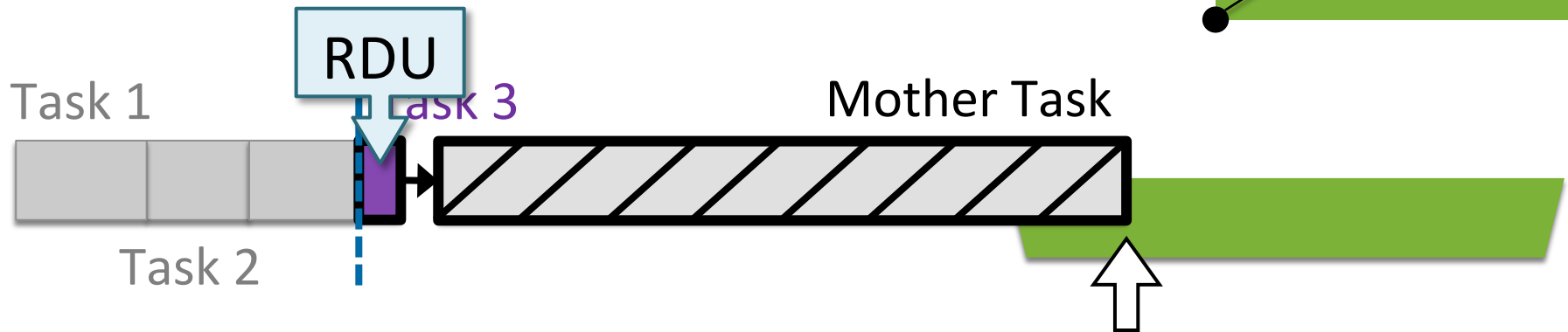
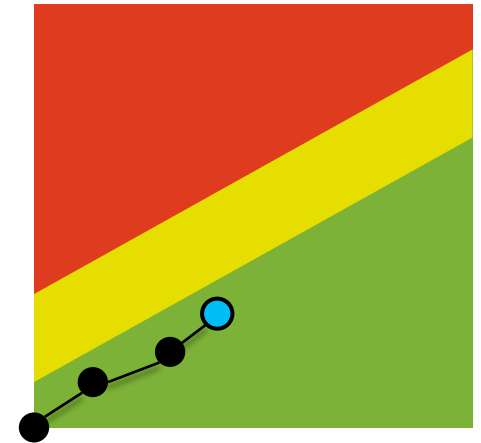


# Procedure for Buffer Management in “Agile” CCPM (cont.)

2016 TOCICO International Conference



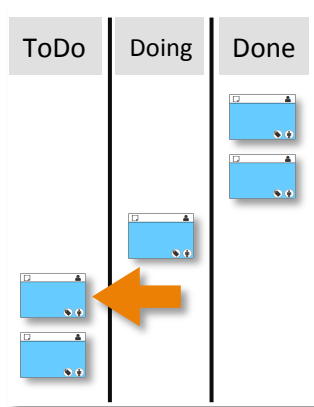
Task 3 is started after Task 2 is completed, and the RDU of Task 3 is reported



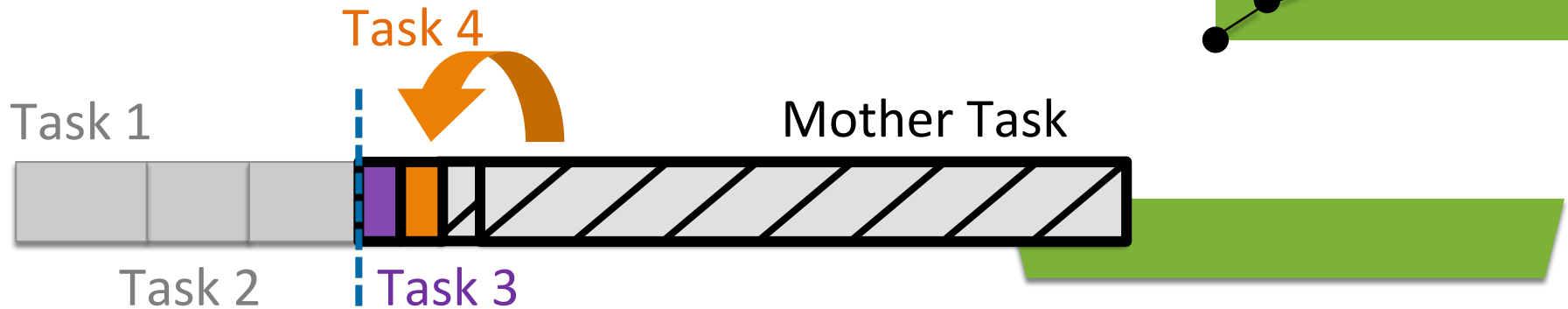
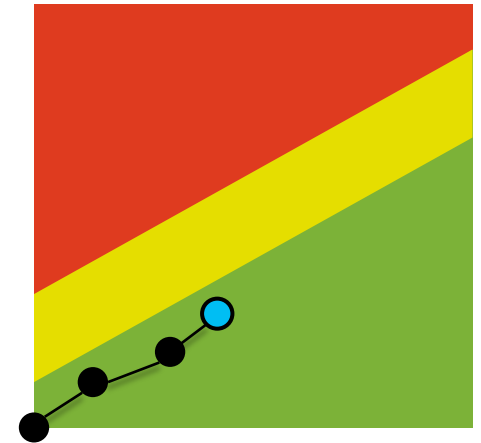
Project buffer is consumed

# Procedure for Buffer Management in “Agile” CCPM (cont.)

2016 TOCICO International Conference

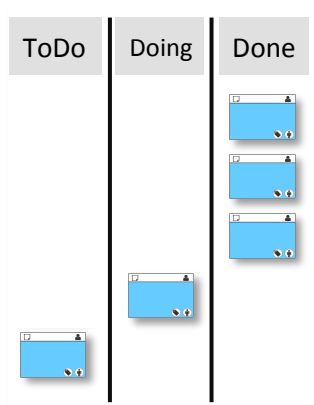


The 4th task (**Task 4**) for the next highest priority feature is broken out from the Mother Task

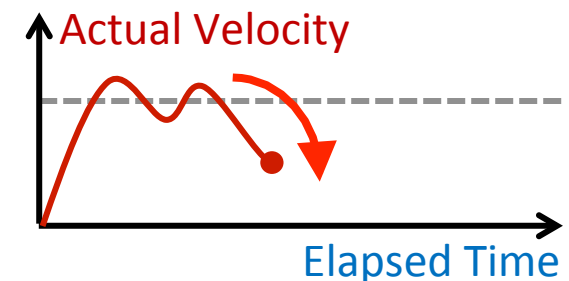
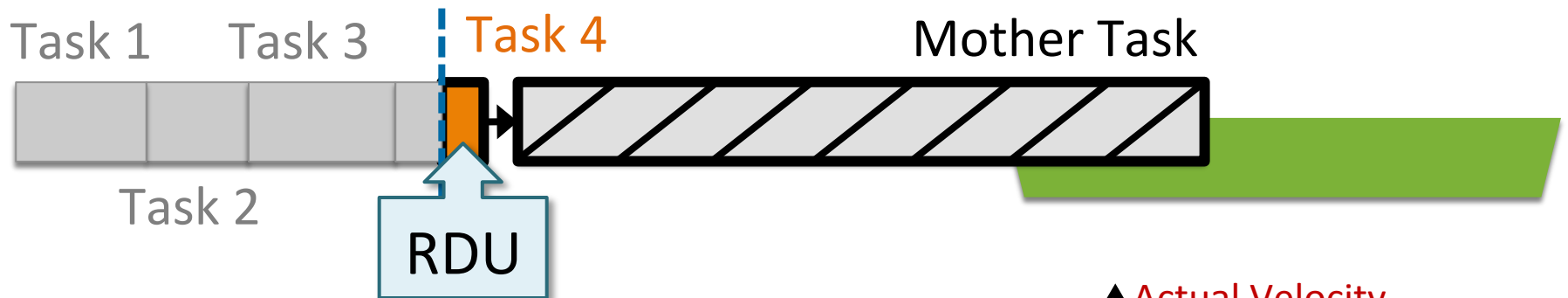
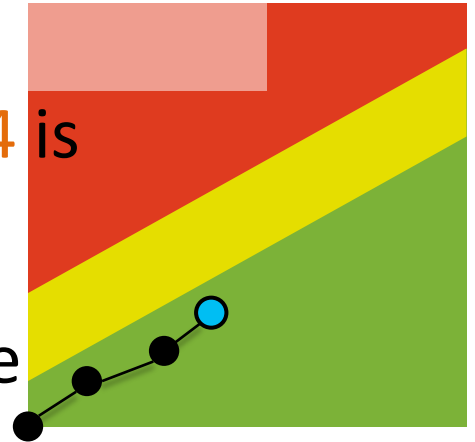


# Procedure for Buffer Management in “Agile” CCPM (cont.)

2016 TOCICO International Conference



Task 4 is started after Task 3 is completed, and the RDU of Task 4 is reported. It is indicated that the Actual Velocity is decreasing at this stage.

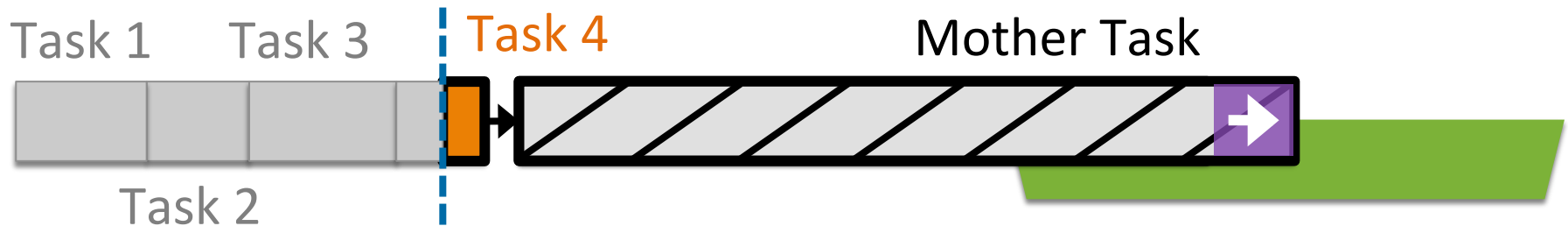
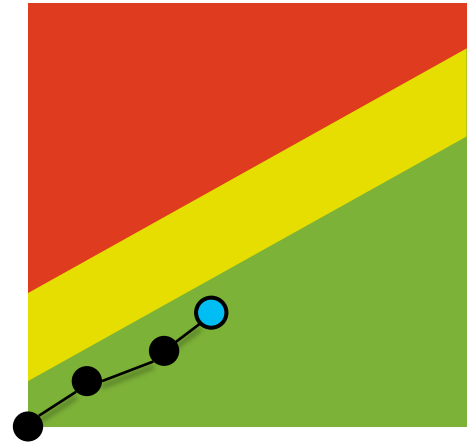




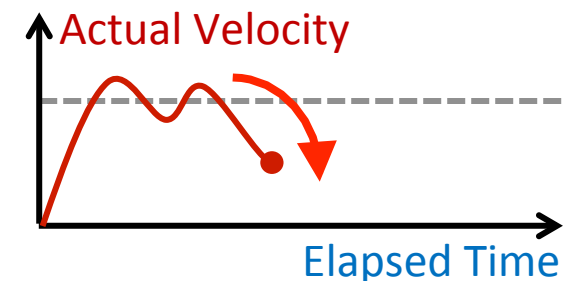
# Procedure for Buffer Management in “Agile” CCPM (cont.)

2016 TOCICO International Conference

The duration of the Mother Task is adjusted by the “Velocity Based Estimation”, according to the **Actual Velocity**



$$ED = \frac{\text{Remaining SP}}{\text{Velocity}}$$

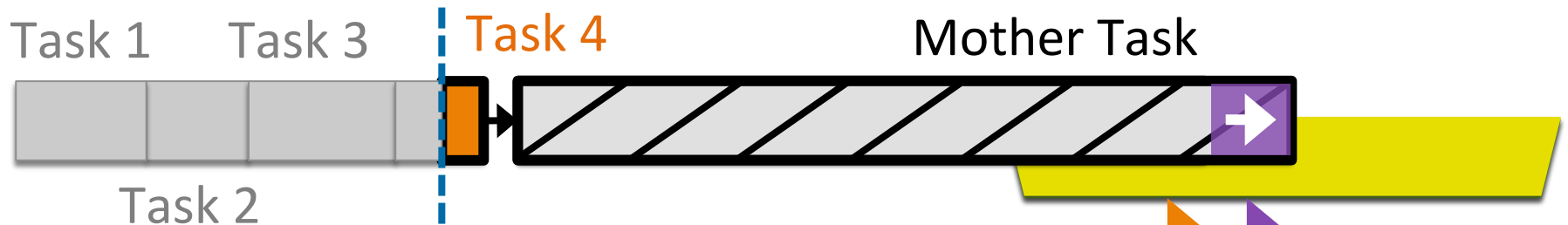
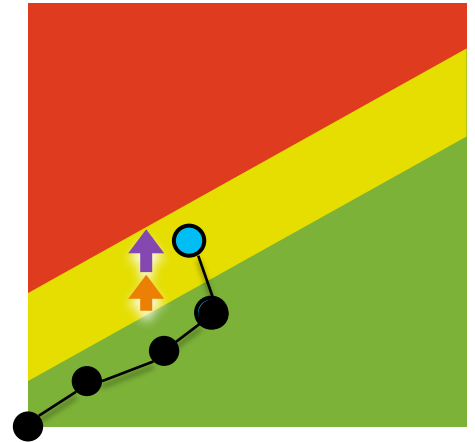


# Procedure for Buffer Management in “Agile” CCPM (cont.)

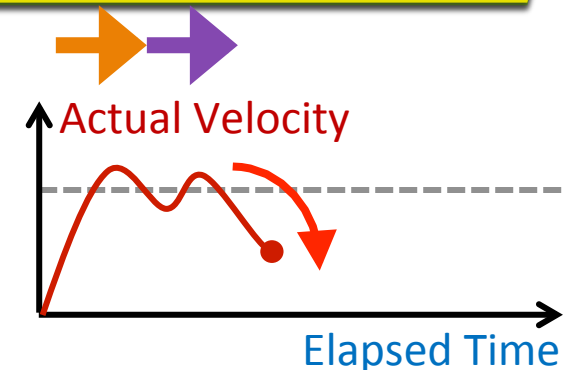
2016 TOCICO International Conference

For this reason, we refer to this as “**Velocity Based** Buffer Management”

The duration of the Mother Task is adjusted by “Velocity Based Estimation”, according to the **Actual Velocity**



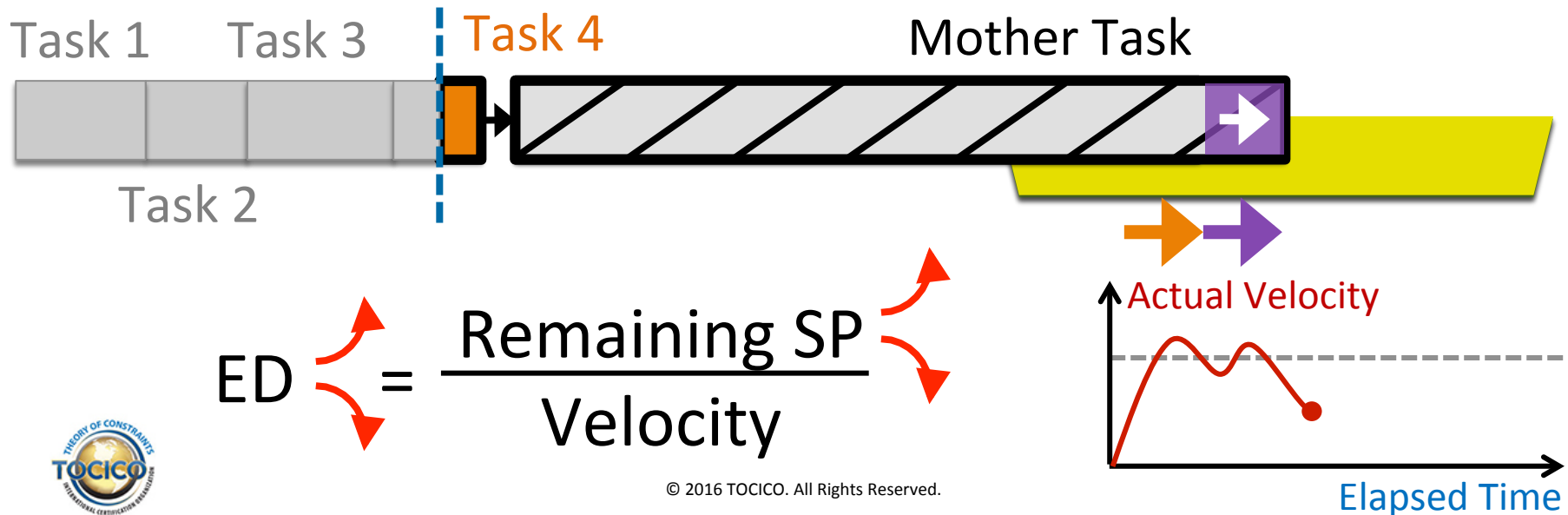
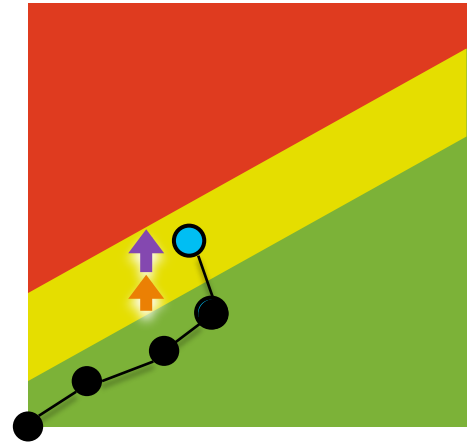
The project buffer is consumed by both the RDU of **Task 4** and the duration of the adjusted Mother Task



# Procedure for Buffer Management in “Agile” CCPM (cont.)

2016 TOCICO International Conference

“Velocity Based Estimation” can be applied even if the Velocity doesn’t change, when it is recognized that the story points of features have changed (for any reason), so the Mother Task shows relevant duration



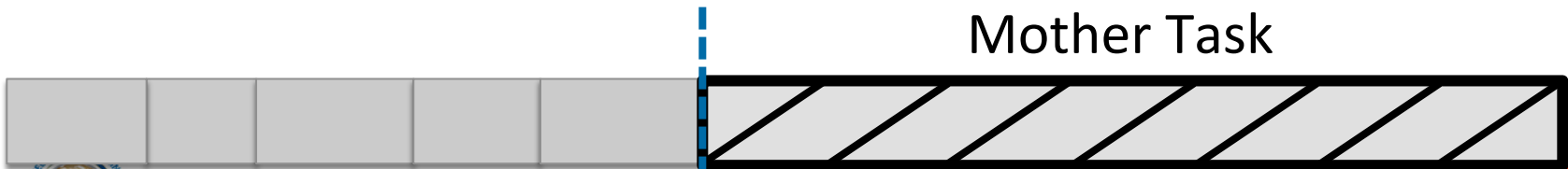
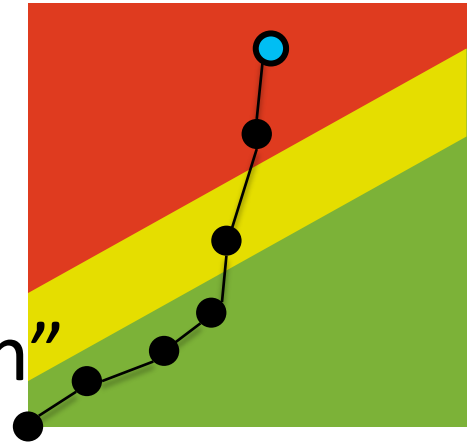
# Procedure for Buffer Management in “Agile” CCPM (cont.)

2016 TOCICO International Conference

No.	Feature	Size
1	F001	8
Scope Buffer		
25	F025	13
28	F028	8
29	F029	3
30	F030	13

$$ED = \frac{\text{Remaining SP}}{\text{Velocity}}$$

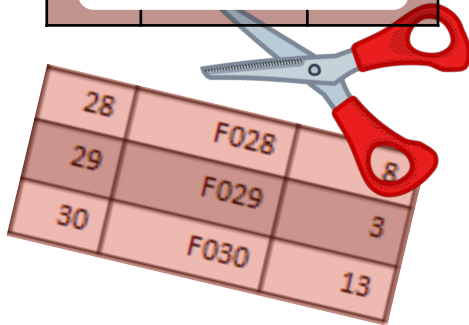
“Velocity Based Estimation” is also applied when the Scope Buffer is invoked



# Procedure for Buffer Management in “Agile” CCPM (cont.)

2016 TOCICO International Conference

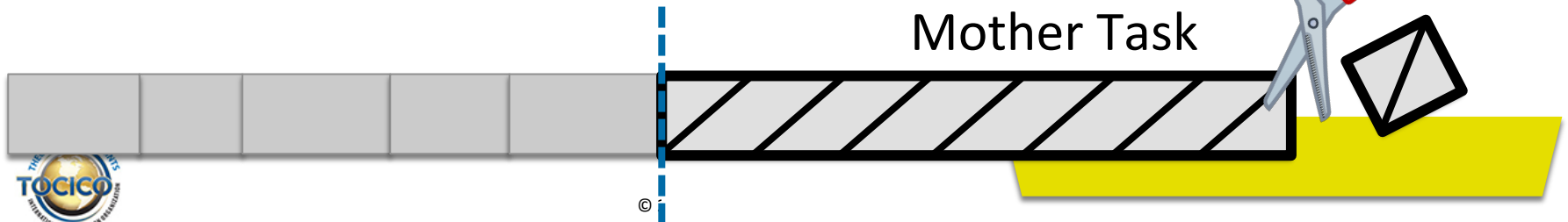
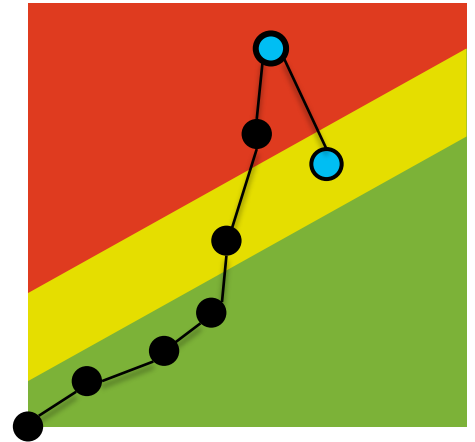
No.	Feature	Size
1	F001	8
~~~~~		
25	F025	13
Scope Buffer		
		3
		5



$$ED = \frac{\text{Remaining SP}}{\text{Velocity}}$$

Remaining SPs decrease by cutting some features in the scope buffer

Estimated duration of the Mother Task is reduced and buffer status is recovered as a result



# Tips and Examples for Buffer Management in “Agile” CCPM

2016 TOCICO International Conference

The next task is broken out from the mother task

The next task is executed and the RDU is reported

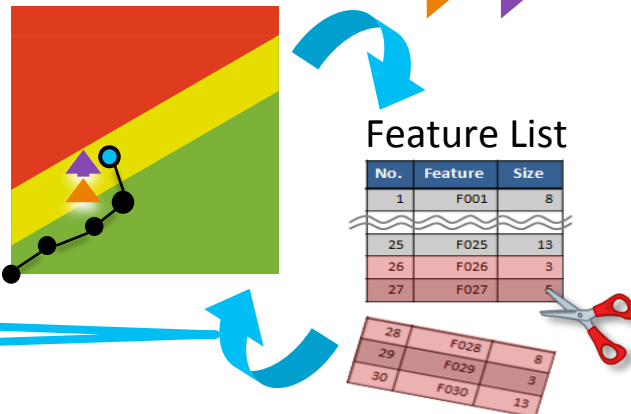
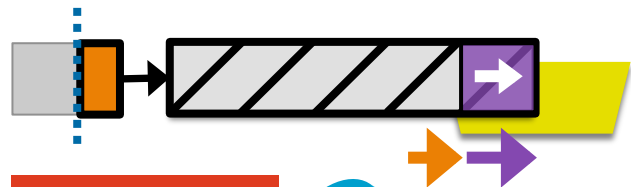
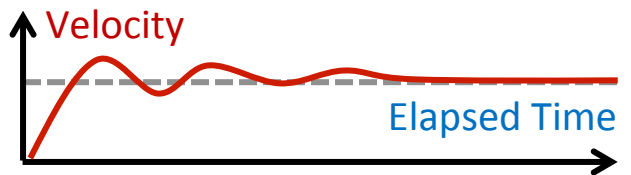
Actual velocity is calculated and the applied velocity is updated as required

The mother task is adjusted by the latest applied velocity as required

Project status is updated according to both the RDU of the current task and the mother task

Scope buffer is invoked as required

Next Task



# Tips and Examples for Buffer Management in “Agile” CCPM

2016 TOCICO International Conference

The next task is broken out from the mother task

The next task is executed and the RDU is reported

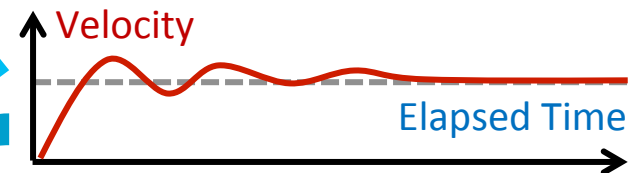
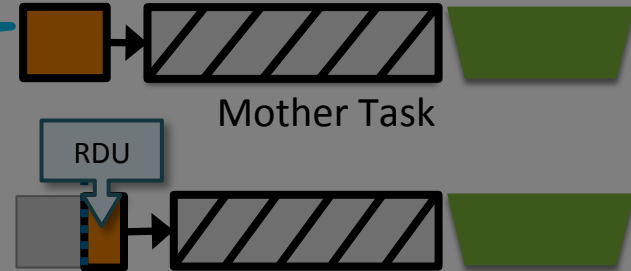
Actual velocity is calculated and the applied velocity is updated as required

The mother task is adjusted by the latest

Proje  
both  
the m

Scope buffer is invoked as required

Next Task



**Tip #4: How Actual Velocity is recognized**

e List

Size
8
13
3
5

29	F029	8
30	F030	13

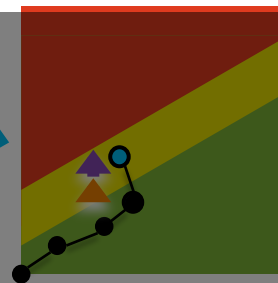
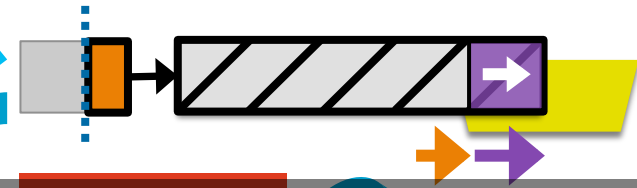
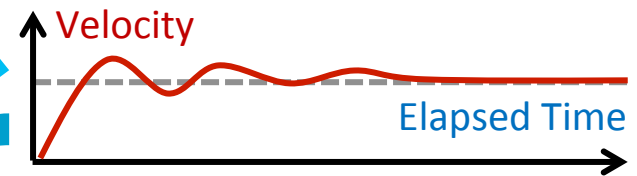
# Tip #5: How to Specify the Velocity Applied to Velocity Based Estimation

Actual velocity is calculated and the applied velocity is updated as required

The mother task is adjusted by the latest applied velocity as required

Project status is updated according to both the RDU of the current task and the mother task

Scope buffer is invoked as required



Feature List

No.	Feature	Size
1	F001	8
25	F025	13
26	F026	3
27	F027	5

28	F028	8
29	F029	3
30	F030	13





# Tips and Examples for Buffer Management in “Agile” CCPM

2016 TOCICO International Conference

The next task is broken out from the mother task

Next Task

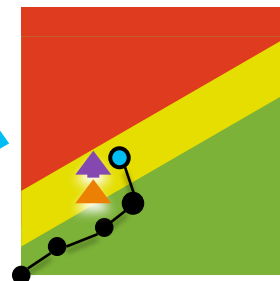


## Tip #6: Using a Scope Buffer or Expanding the Project Buffer?

The mother task is adjusted by the latest applied velocity as required

Project status is updated according to both the RDU of the current task and the mother task

Scope buffer is invoked as required



Feature List

No.	Feature	Size
1	F001	8
25	F025	13
26	F026	3
27	F027	5

28	F028	8
29	F029	3
30	F030	13



## Example #2: How VBBM Works

The next task is broken out from the mother task

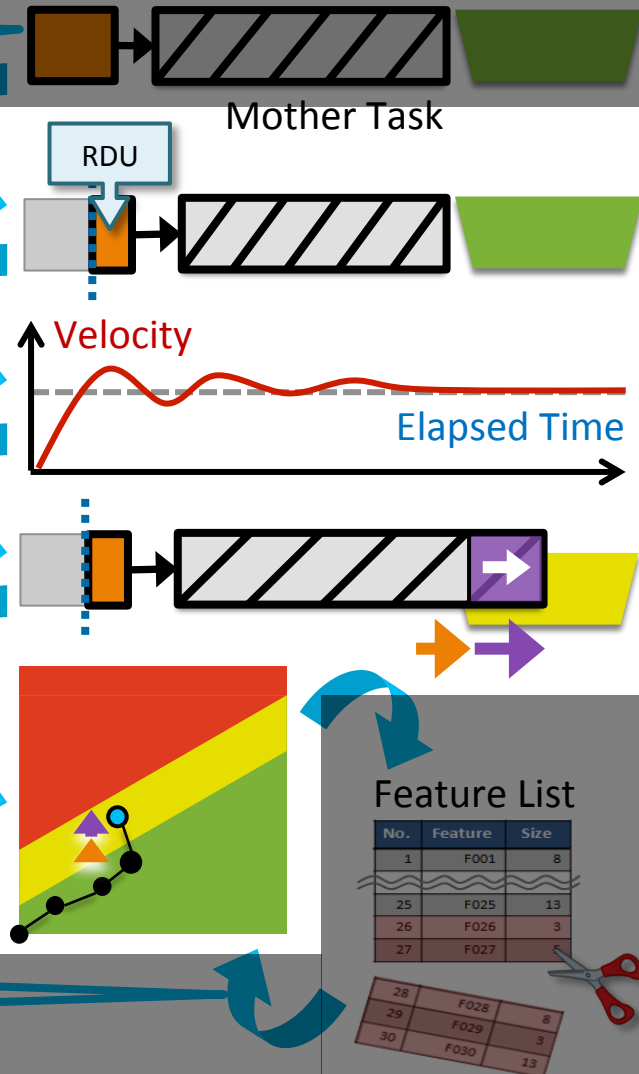
The next task is executed and the RDU is reported

Actual velocity is calculated and the applied velocity is updated as required

The mother task is adjusted by the latest applied velocity as required

Project status is updated according to both the RDU of the current task and the mother task

Scope buffer is invoked as required



# Tips and Examples for Buffer Management in “Agile” CCPM

2016 TOCICO International Conference

The next task is broken out from the mother task

The next task is executed and the RDU is reported

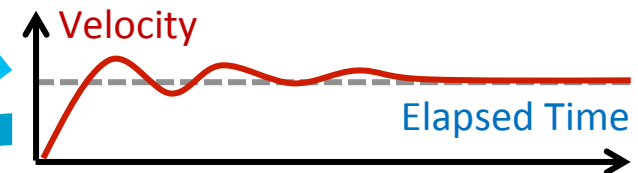
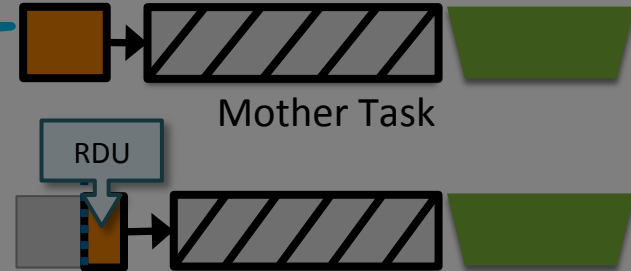
Actual velocity is calculated and the applied velocity is updated as required

The mother task is adjusted by the latest

Proje  
both  
the m

Scope buffer is invoked as required

Next Task



**Tip #4: How Actual Velocity is recognized**

e List

Size
8
13
3
5

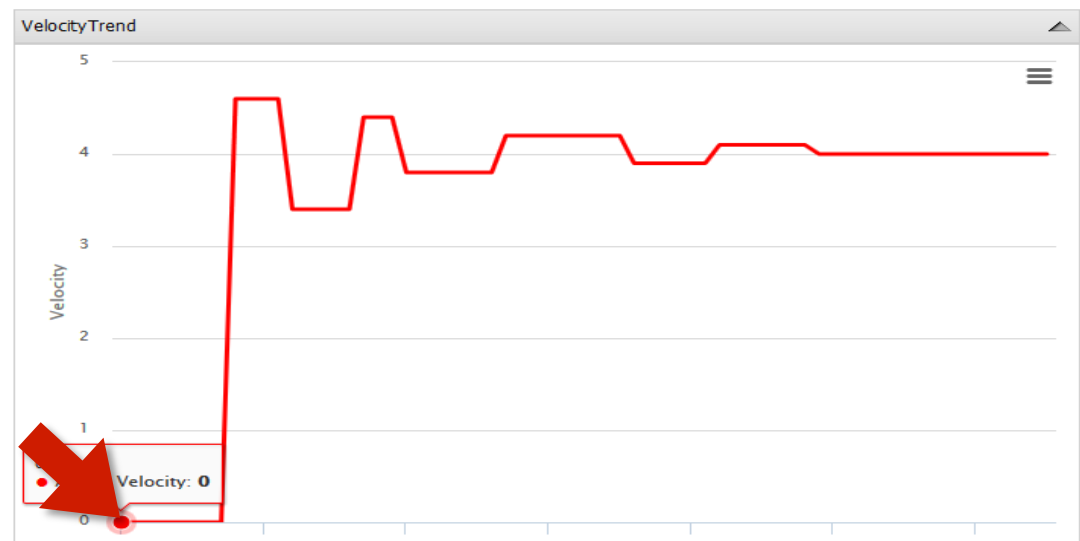
29	F029	8
30	F030	13

# Tip #4: How Actual Velocity is recognized

2016 TOCICO International Conference

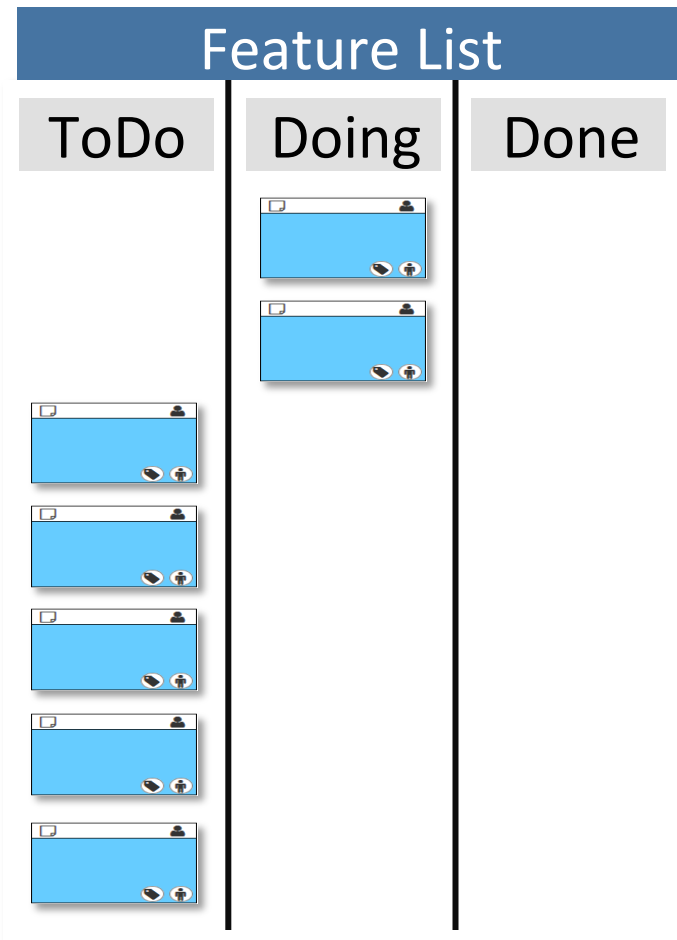


At the kick-off of a project, all tasks are placed in the “ToDo” lane (not started) and the Actual Velocity is recognized as “0”

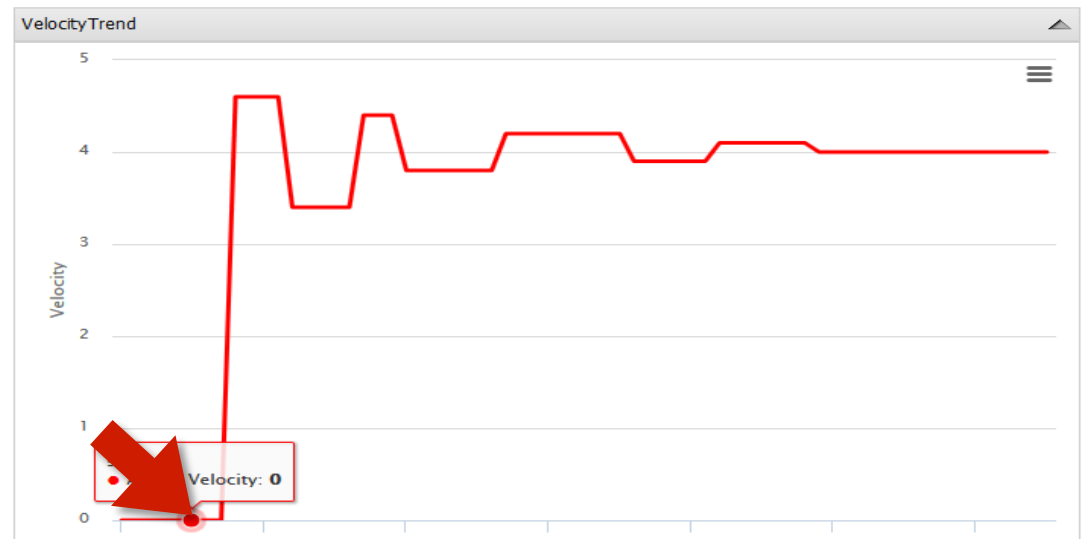


# Tip #4: How Actual Velocity is recognized (cont.)

2016 TOCICO International Conference

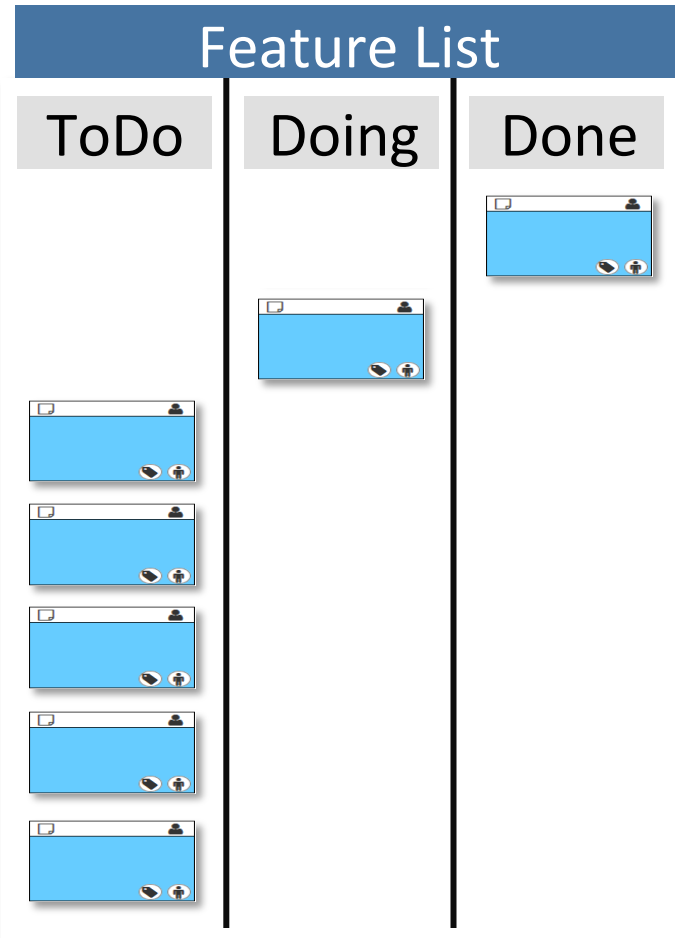


A few days later, no task is yet completed, although some tasks are in execution (placed in the “Doing” lane), and the Actual Velocity is still recognized as “0”

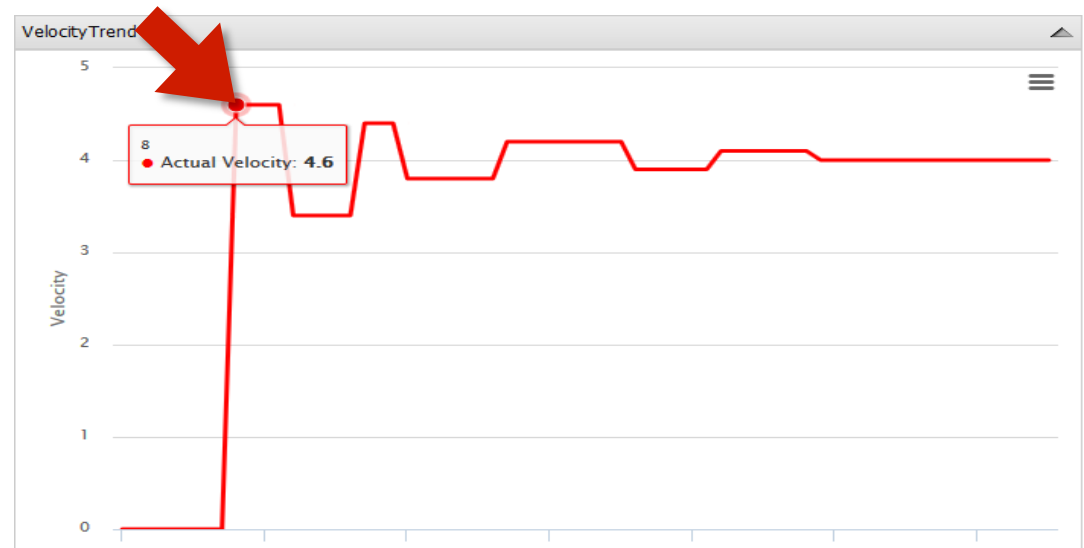


# Tip #4: How Actual Velocity is recognized (cont.)

2016 TOCICO International Conference

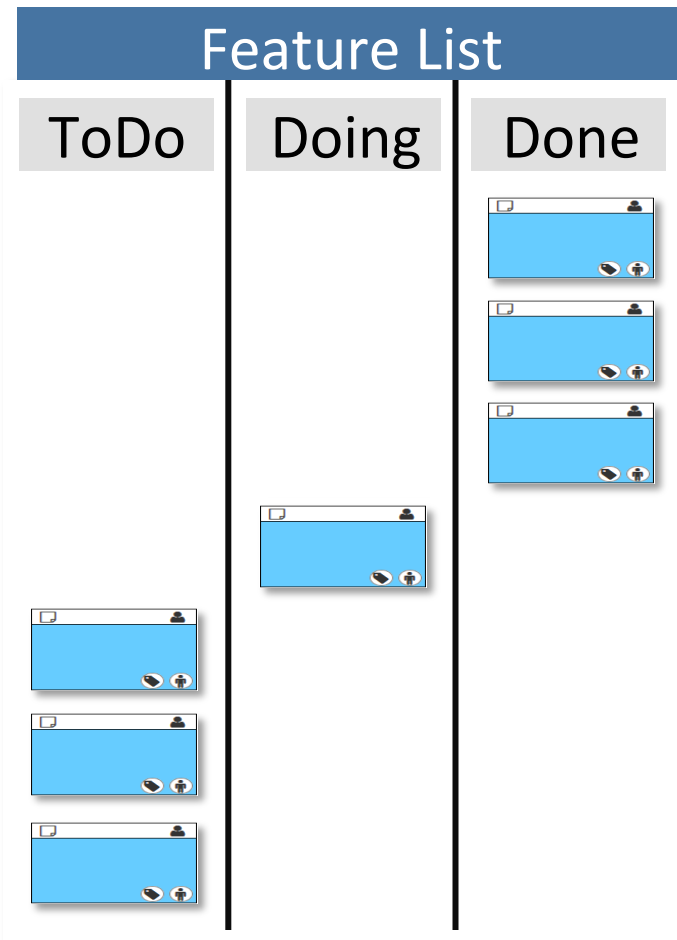


When the first task is completed (and placed in the “Done” lane), the Actual Velocity can be calculated according to the size of the task and elapsed time

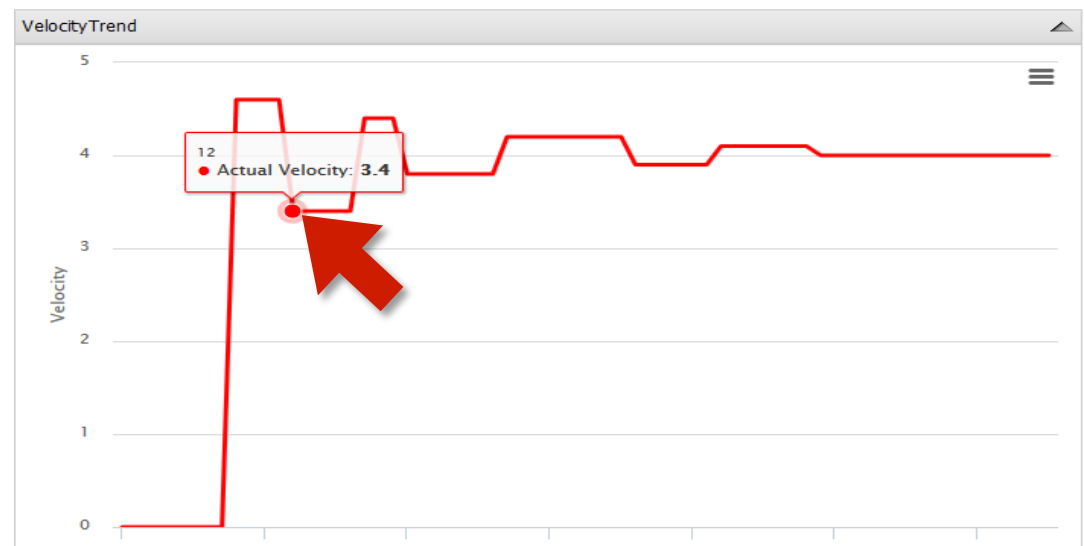


# Tip #4: How Actual Velocity is recognized (cont.)

2016 TOCICO International Conference

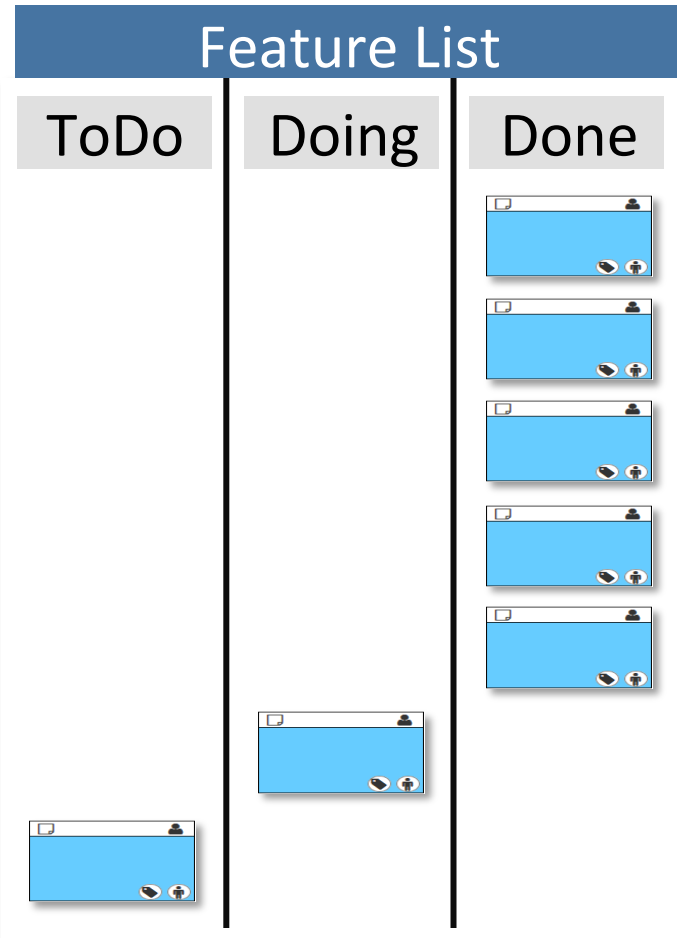


A few weeks later, 2 more tasks have been completed, but they have taken longer than the first task, and the Actual Velocity is decreased

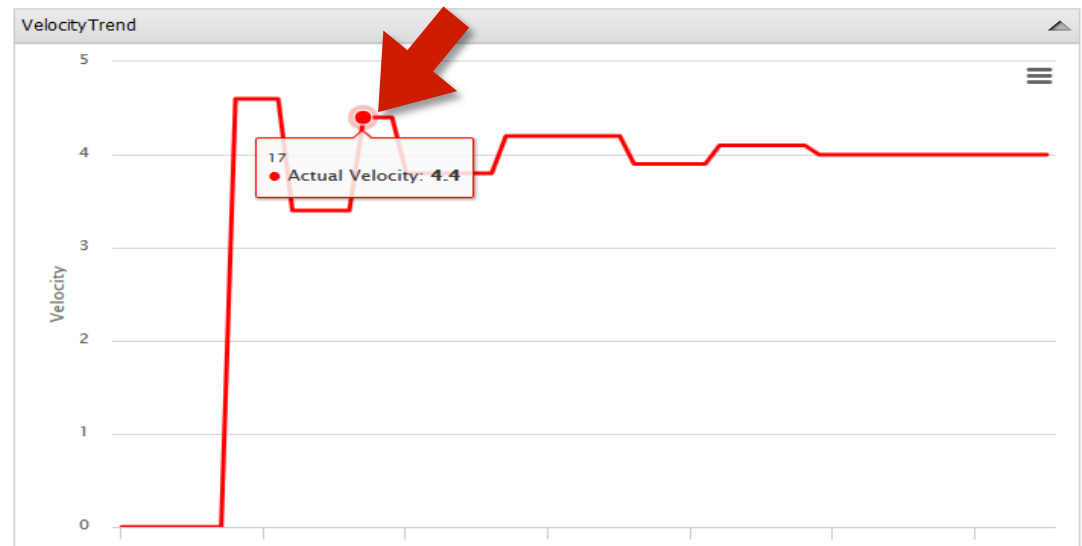


# Tip #4: How Actual Velocity is recognized (cont.)

2016 TOCICO International Conference



Another a few weeks later, some more tasks have completed faster, and the Actual Velocity has slightly recovered





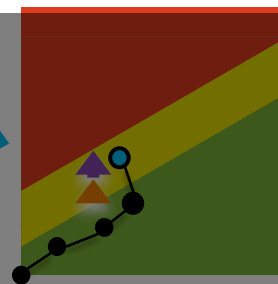
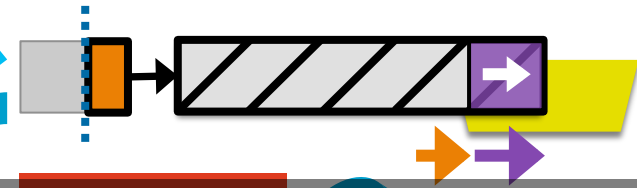
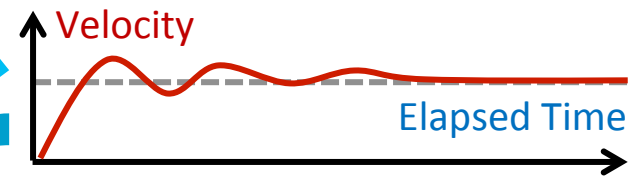
# Tip #5: How to Specify the Velocity Applied to Velocity Based Estimation

Actual velocity is calculated and the applied velocity is updated as required

The mother task is adjusted by the latest applied velocity as required

Project status is updated according to both the RDU of the current task and the mother task

Scope buffer is invoked as required



Feature List

No.	Feature	Size
1	F001	8
25	F025	13
26	F026	3
27	F027	5

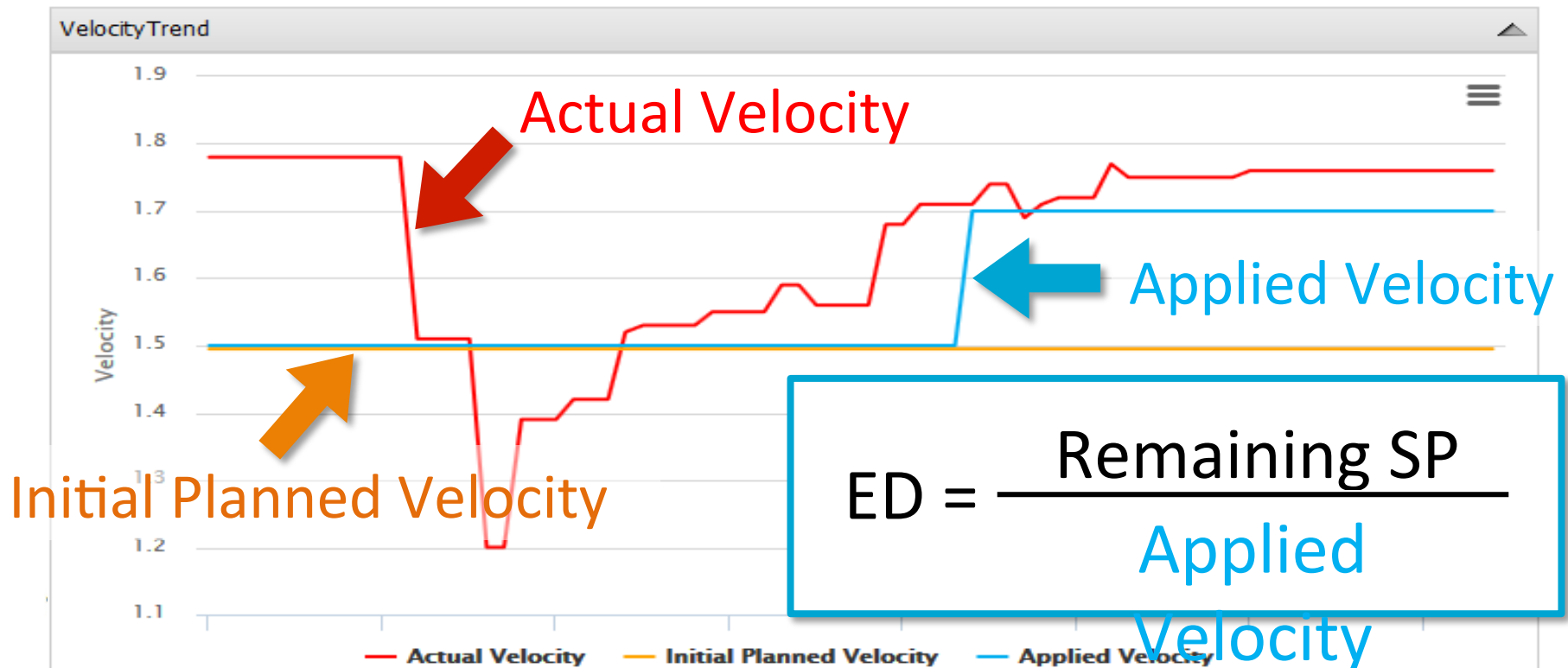
28	F028	8
29	F029	3
30	F030	13



# Tip #5: How to Specify the Velocity Applied to VBE

2016 TOCICO International Conference

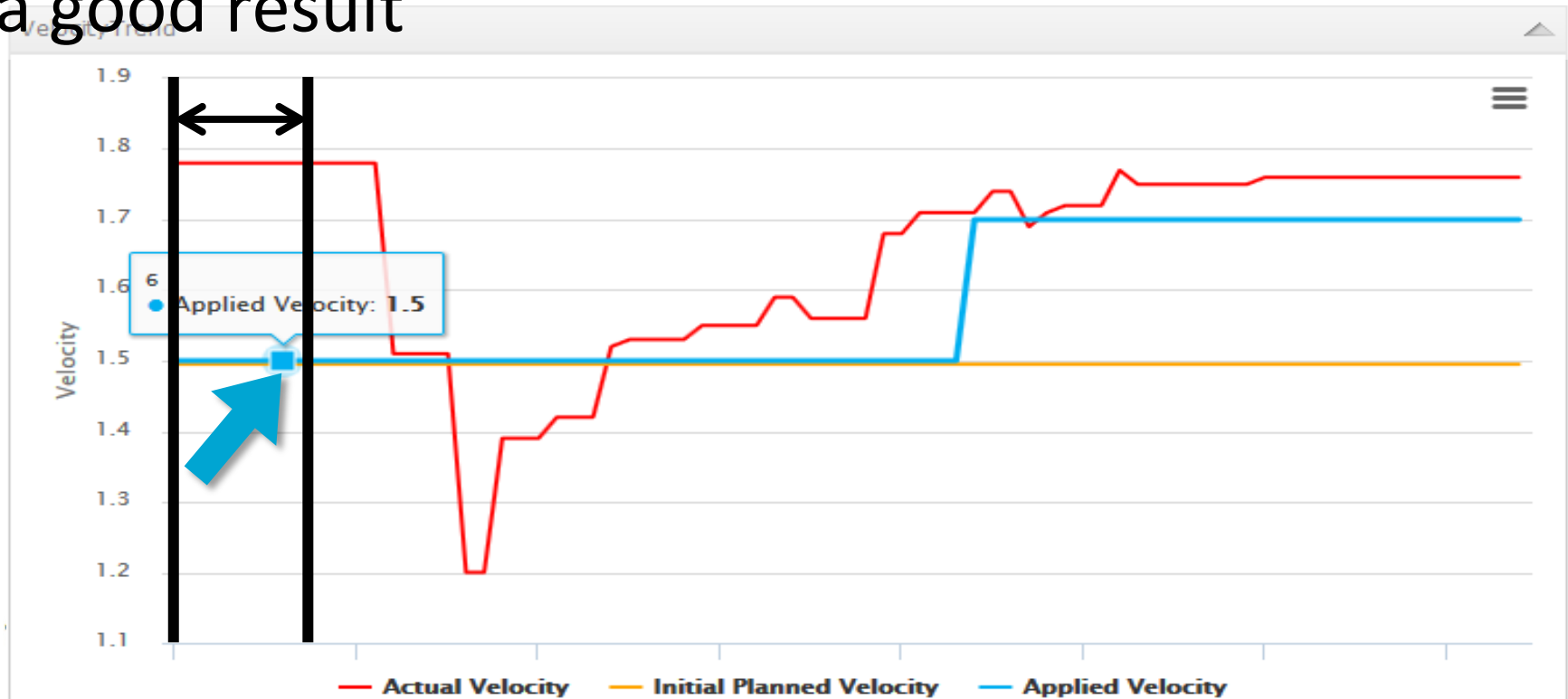
To be precise, Estimated Duration is calculated according to the **Applied Velocity** instead of the **Initial Planned Velocity** or the **Actual Velocity**



# Tip #5: How to Specify the Velocity Applied to VBE (cont.)

2016 TOCICO International Conference

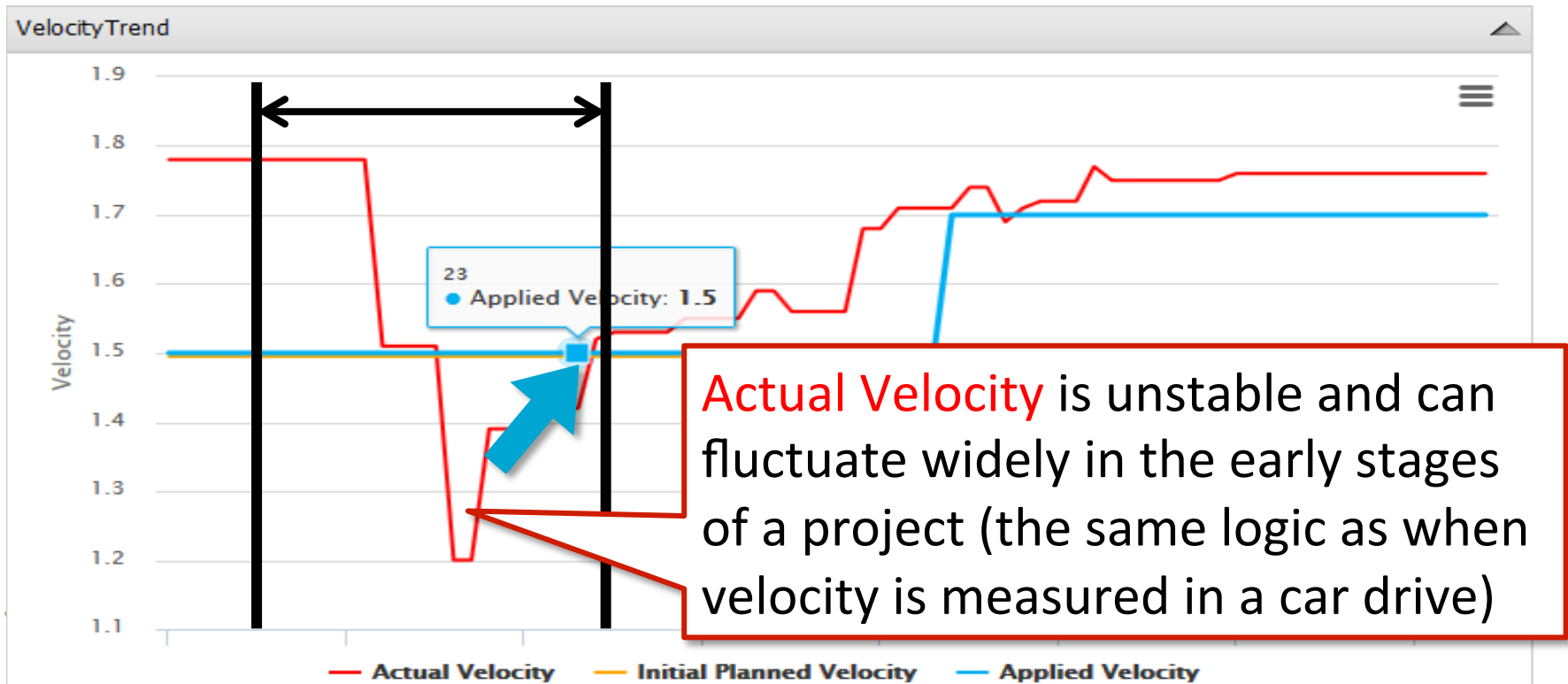
At the beginning of this project, the **Initial Planned Velocity** (“1.5” [sp/d]) was used as the **Applied Velocity**, even as the **Actual Velocity** was showing a good result



# Tip #5: How to Specify the Velocity Applied to VBE (cont.)

2016 TOCICO International Conference

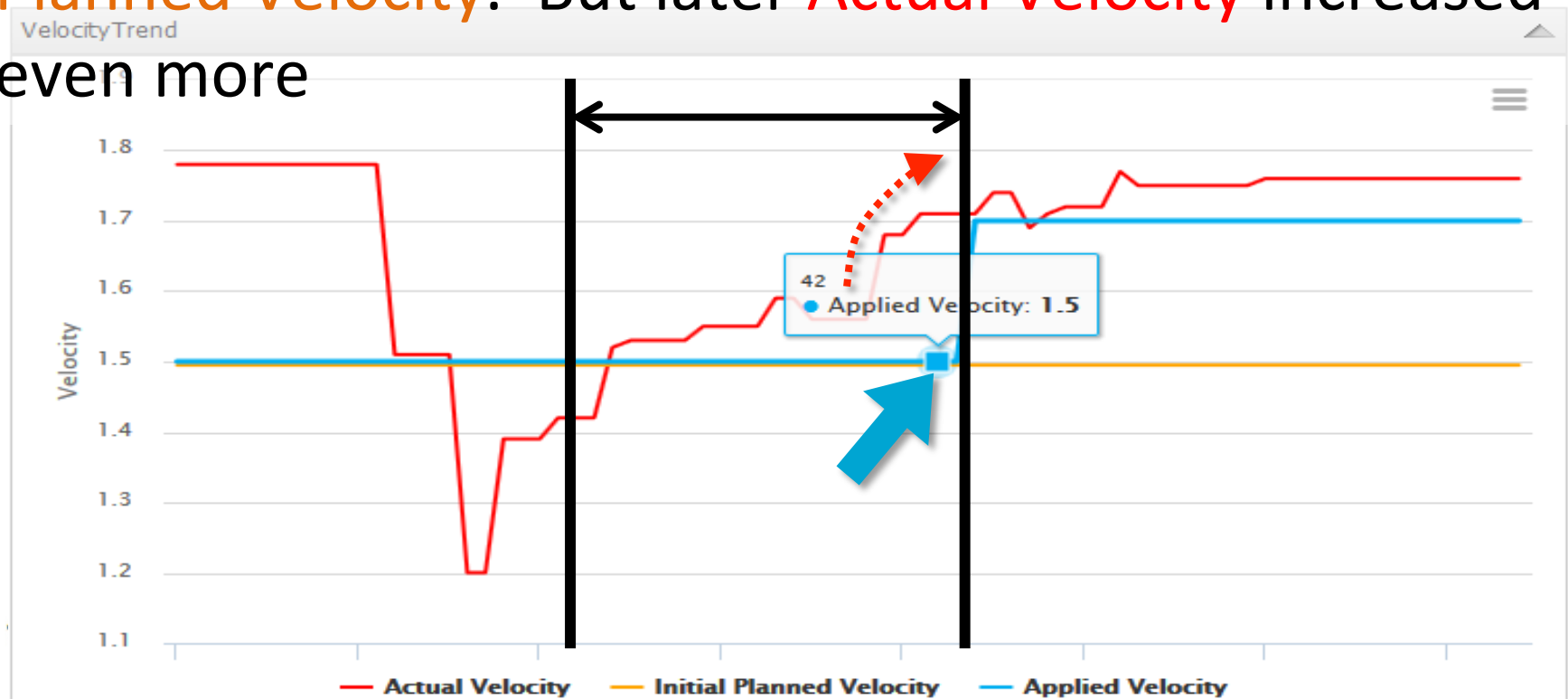
The **Initial Planned Velocity** was still used as the **Applied Velocity** since the **Actual Velocity** had been fluctuating



# Tip #5: How to Specify the Velocity Applied to VBE (cont.)

2016 TOCICO International Conference

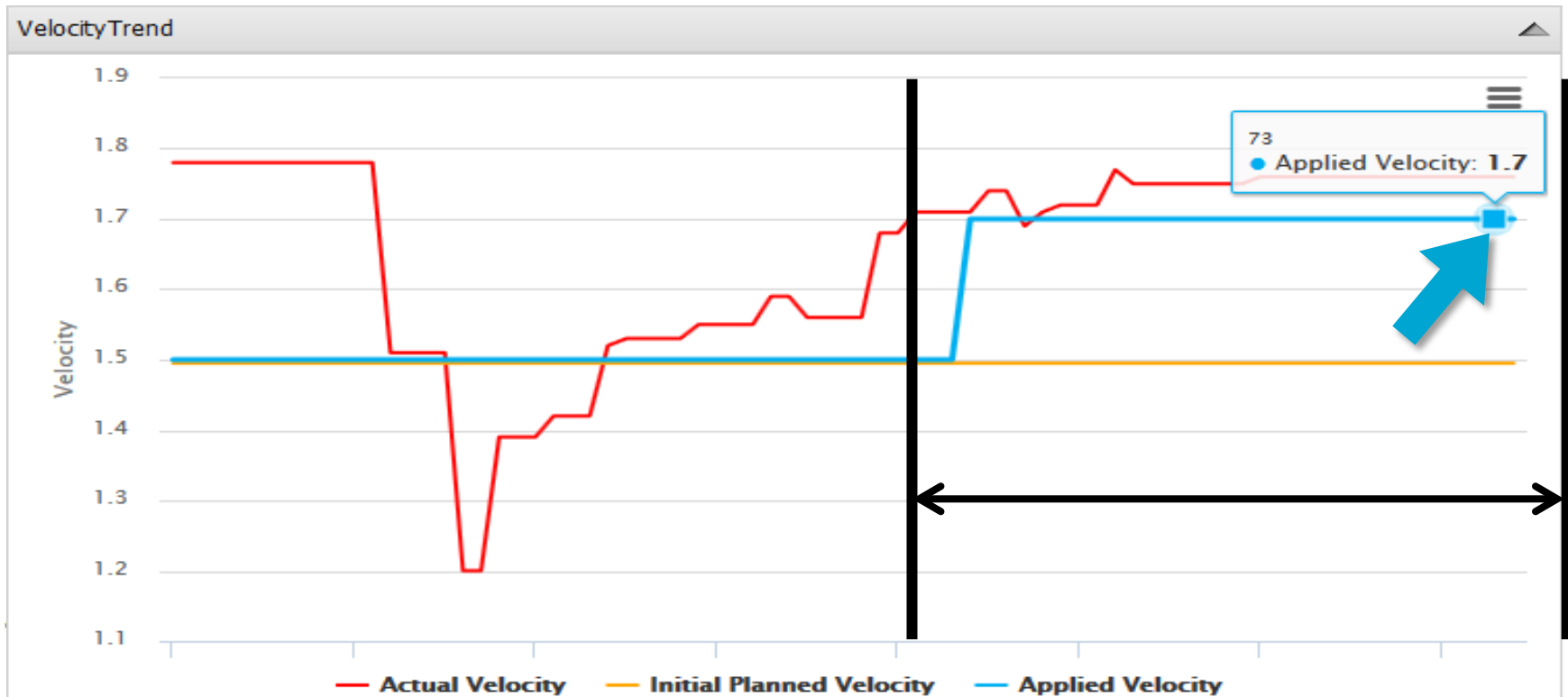
When it seemed that **Actual Velocity** had stabilized, **Initial Planned Velocity** was still maintained for a little while, because **Actual Velocity** stayed close to **Initial Planned Velocity**. But later **Actual Velocity** increased even more



# Tip #5: How to Specify the Velocity Applied to VBE (cont.)

2016 TOCICO International Conference

It was decided that the **Applied Velocity** should be increased to “1.7” [sp/d] (as indicated by the trend of **Actual Velocity**)



# Tips and Examples for Buffer Management in “Agile” CCPM

2016 TOCICO International Conference

The next task is broken out from the mother task

Next Task

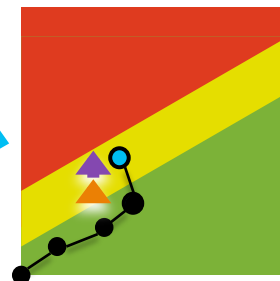


## Tip #6: Using a Scope Buffer or Expanding the Project Buffer?

The mother task is adjusted by the latest applied velocity as required

Project status is updated according to both the RDU of the current task and the mother task

Scope buffer is invoked as required



Feature List

No.	Feature	Size
1	F001	8
25	F025	13
26	F026	3
27	F027	5

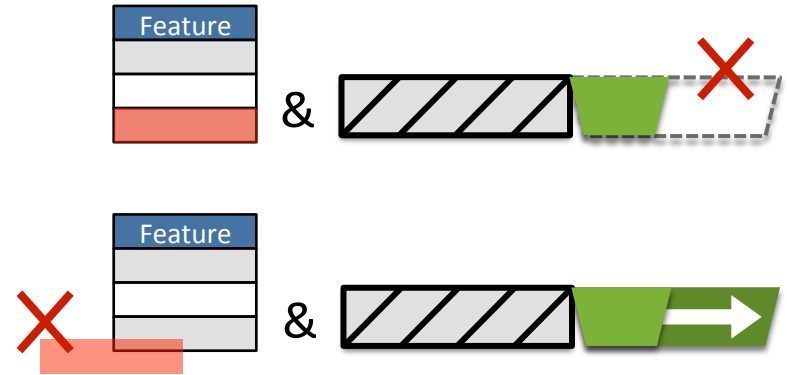
28	F028	8
29	F029	3
30	F030	13



# Tip #6: Using a Scope Buffer or Expanding the Project Buffer?

2016 TOCICO International Conference

- Why a Scope Buffer?
  - Scope Buffer without extending Project Buffer?
  - Extending Project Buffer without Scope Buffer?



- Using a Scope Buffer instead of extending the Project Buffer is beneficial when you need to avoid the following:
  - Longer projects and resulting later completion dates
  - Increase in WIP



## Example #2: How VBBM Works

The next task is broken out from the mother task

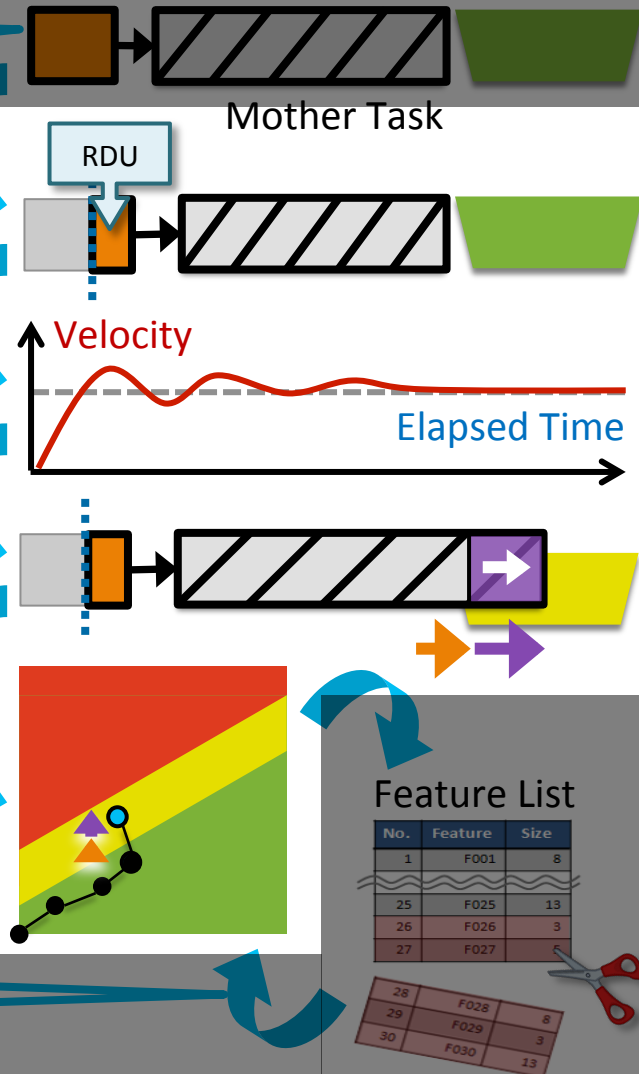
The next task is executed and the RDU is reported

Actual velocity is calculated and the applied velocity is updated as required

The mother task is adjusted by the latest applied velocity as required

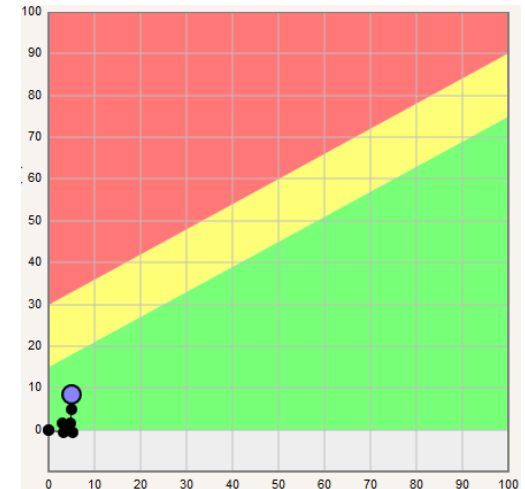
Project status is updated according to both the RDU of the current task and the mother task

Scope buffer is invoked as required



## 2016 TOCICO International Conference

- 

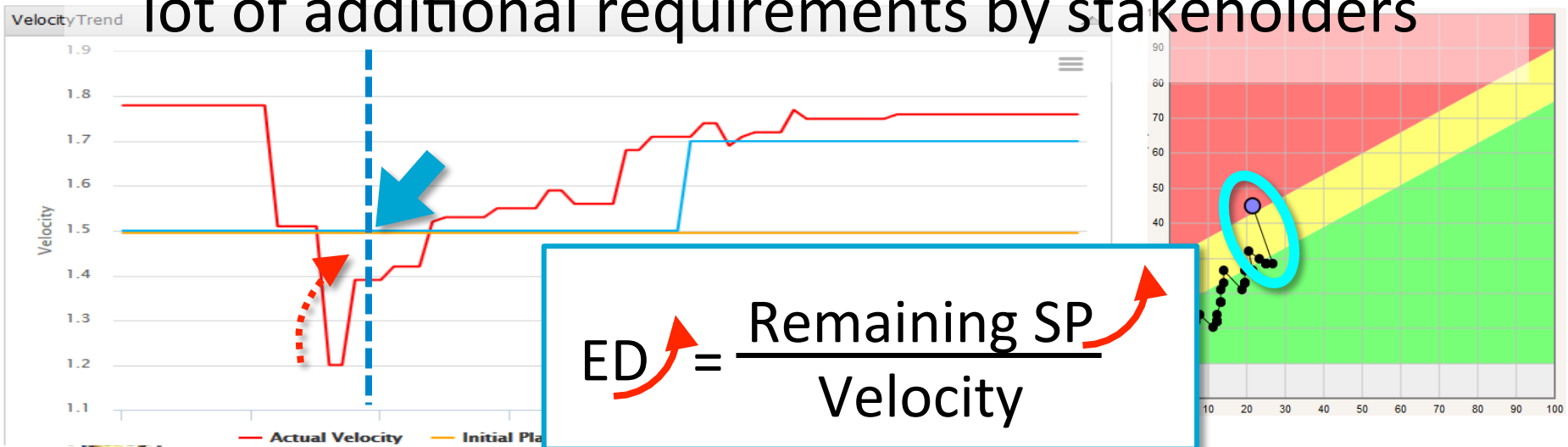


# Example #2: How VBBM Works

## - % Chain Complete : 35% -

2016 TOCICO International Conference

- The **Initial Planned Velocity** is still used as the **Applied Velocity**, even while the **Actual Velocity** is getting better, since developer's skill for the new technologies is increasing
- Buffer status is getting worse because there are a lot of additional requirements by stakeholders

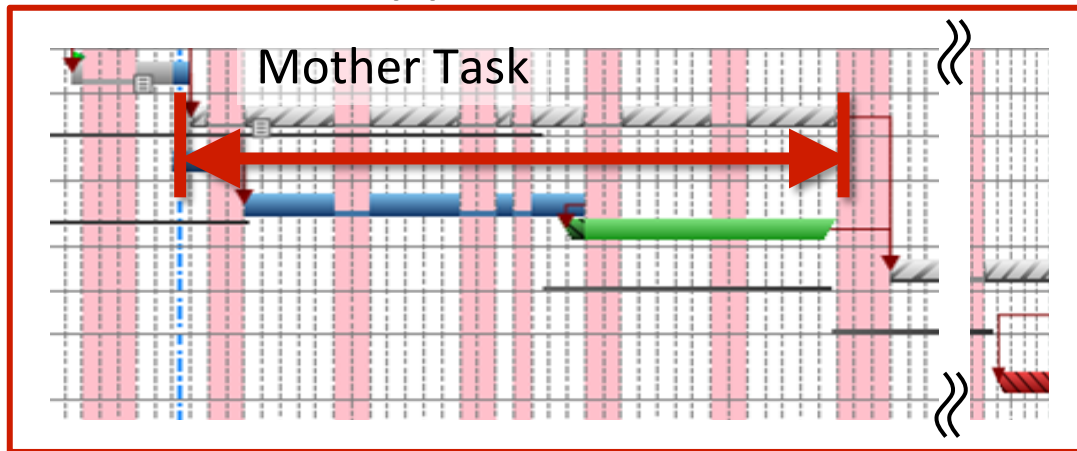


# Example #2: How VBBM Works

## - % Chain Complete : 35% - (cont.)

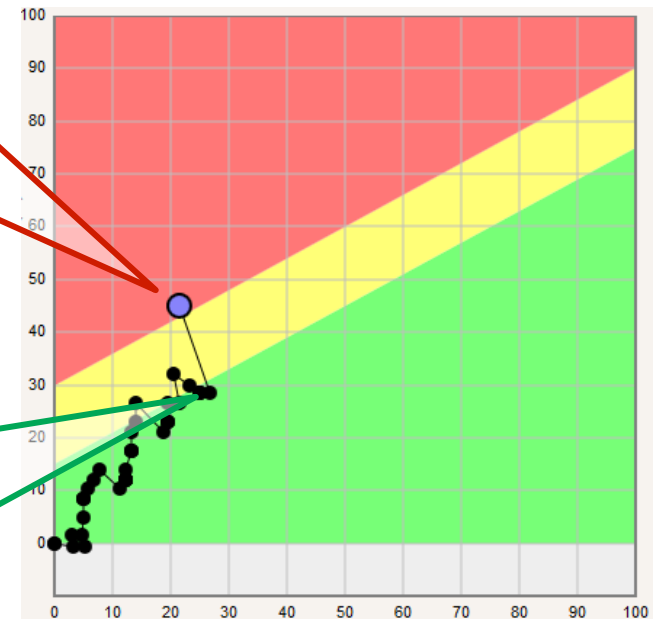
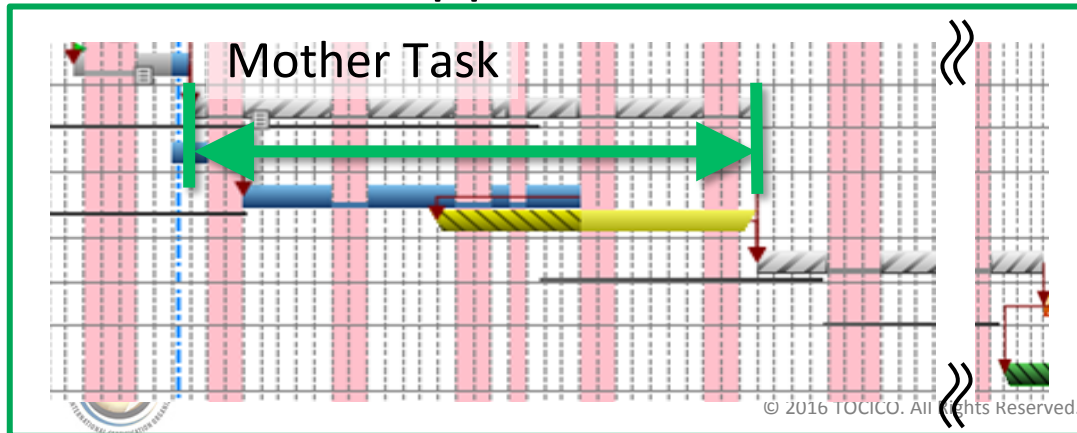
2016 TOCICO International Conference

After VBE is applied



$$ED = \frac{\text{Remaining SP}}{\text{Velocity}}$$

Before VBE is applied

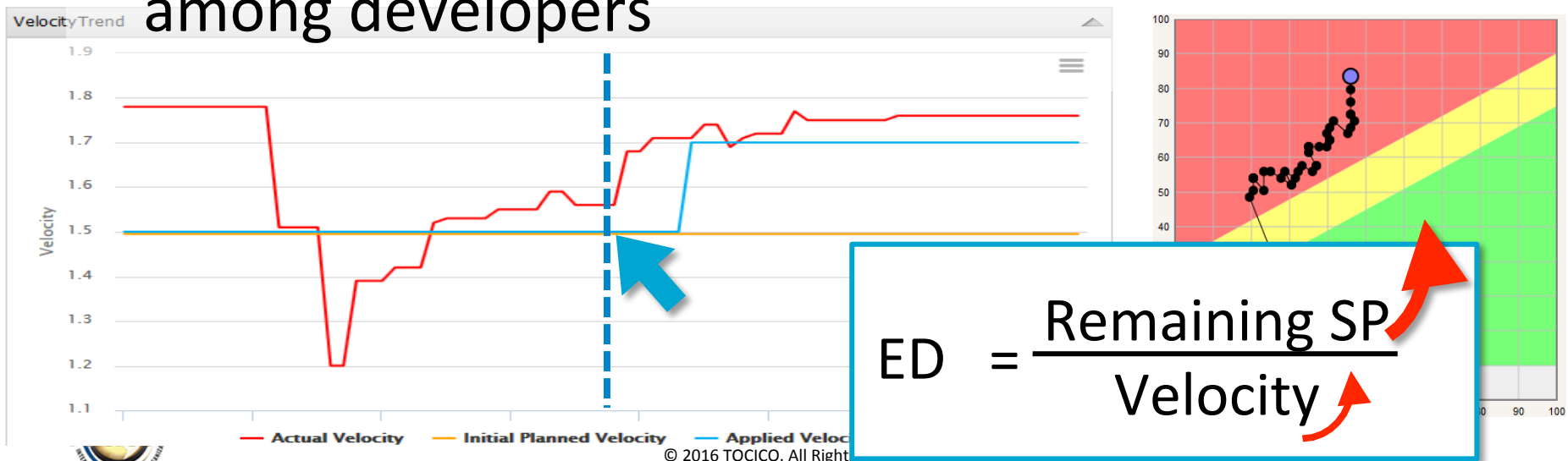


# Example #2: How VBBM Works

## - % Chain Complete : 45% -

2016 TOCICO International Conference

- The **Initial Planned Velocity** is still used as the **Applied Velocity** since it is almost the same as the **Actual Velocity**
- The size of features keeps increasing at a higher pace against the increase of the proficiency among developers

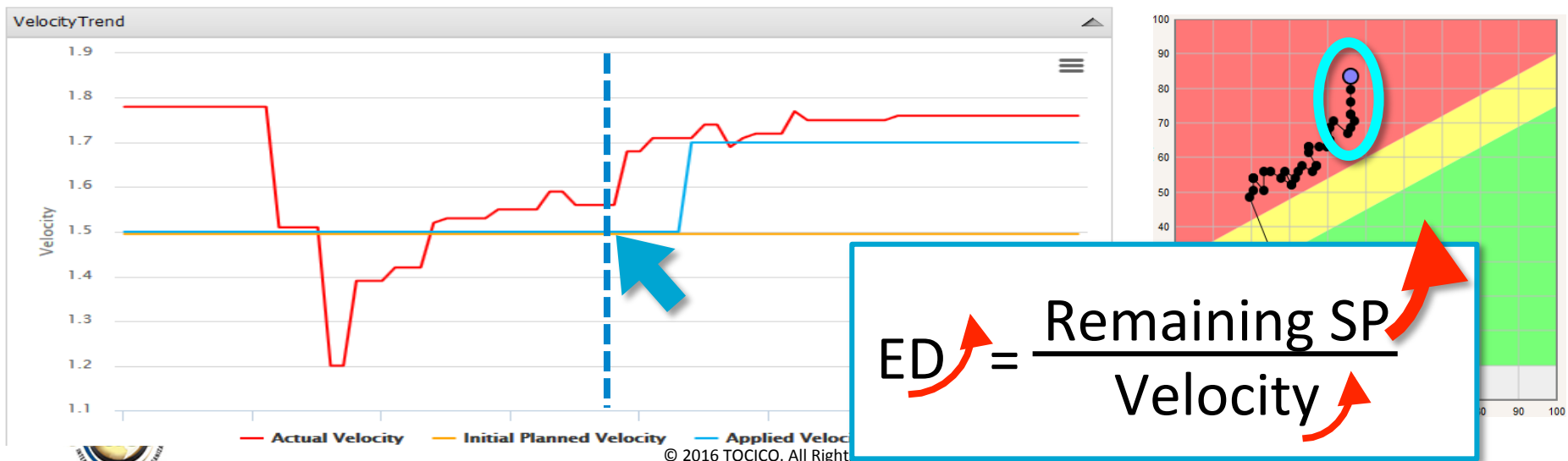
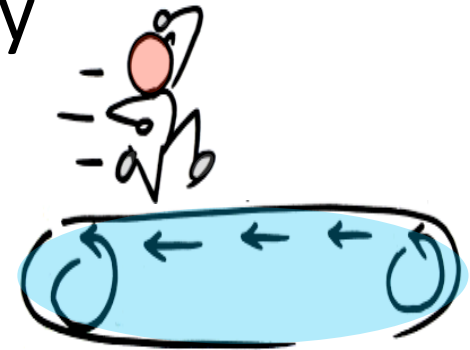


# Example #2: How VBBM Works

## - % Chain Complete : 45% - (cont.)

2016 TOCICO International Conference

- Buffer status is deteriorating sharply because developers are struggling to complete the toughest features in this project (like reverse running a on moving walk!)



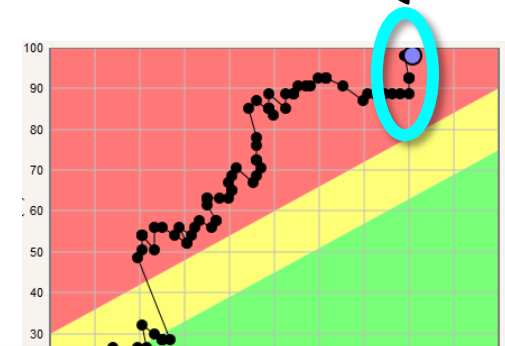
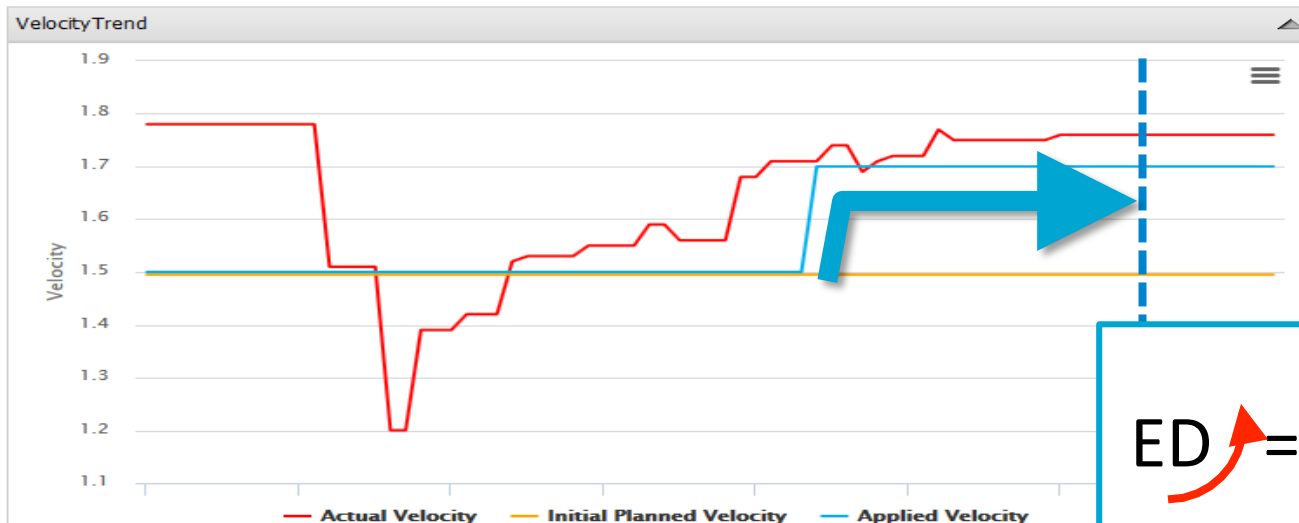
$$ED = \frac{\text{Remaining SP}}{\text{Velocity}}$$

# Example #2: How VBBM Works

## - % Chain Complete : 80% -

2016 TOCICO International Conference

- The 80% mark is achieved just before initiating the integration phase (QA phase)
- Buffer status is still getting worse even as a high level of velocity is maintained, since some small bugs are found and need to be fixed before QA



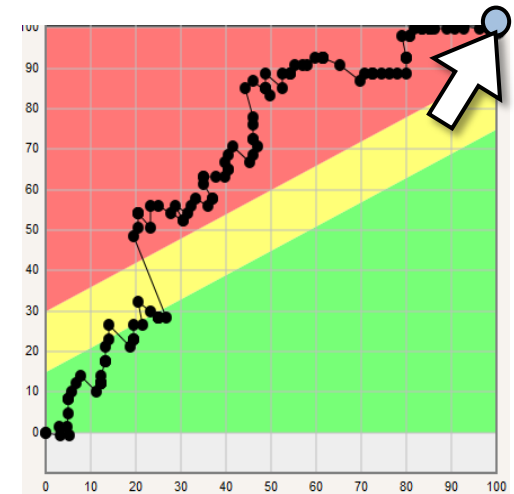
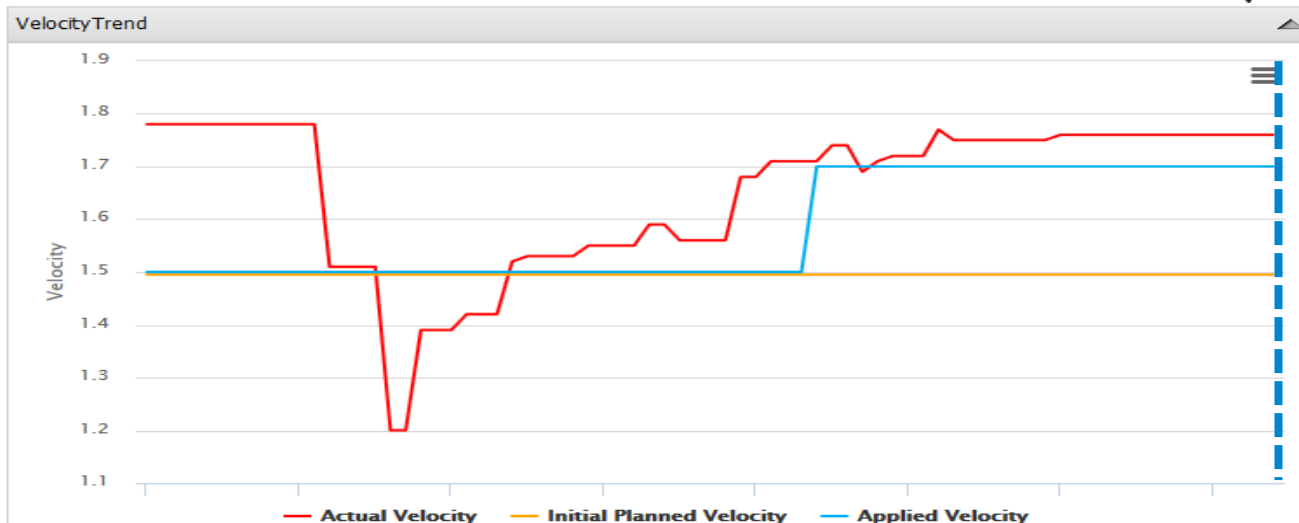
$$ED = \frac{\text{Remaining SP}}{\text{Velocity}}$$

# Example #2: How VBBM Works

## - % Chain Complete : 100% -

2016 TOCICO International Conference

The project completes on time  
(only 24 “min” to the due date!)

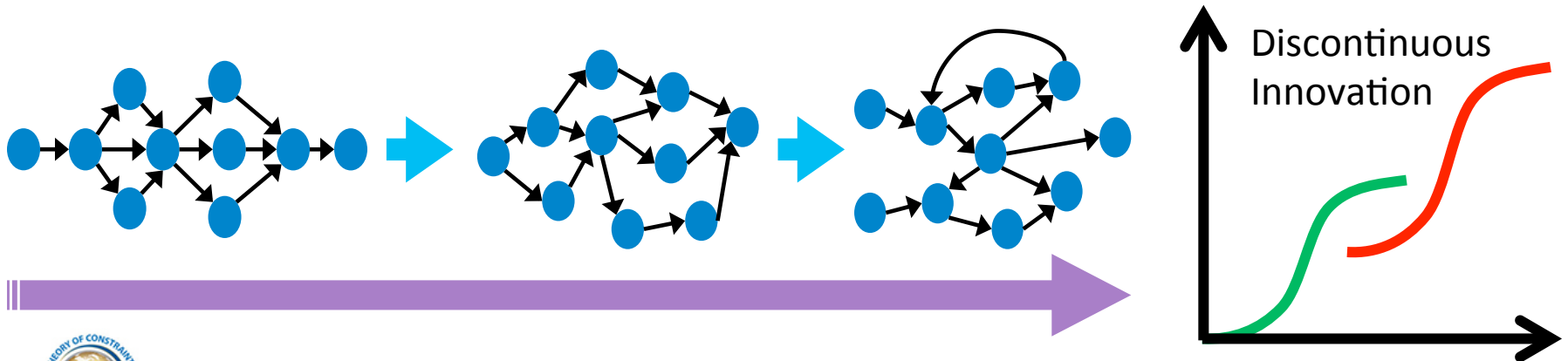




# Conclusions

2016 TOCICO International Conference

- The project was challenging, because:
  - it was a Major Upgrade of existing SW product
  - it required “Discontinuous Innovation” and state-of-the-art technologies
  - its environment was recognized as “Complex”

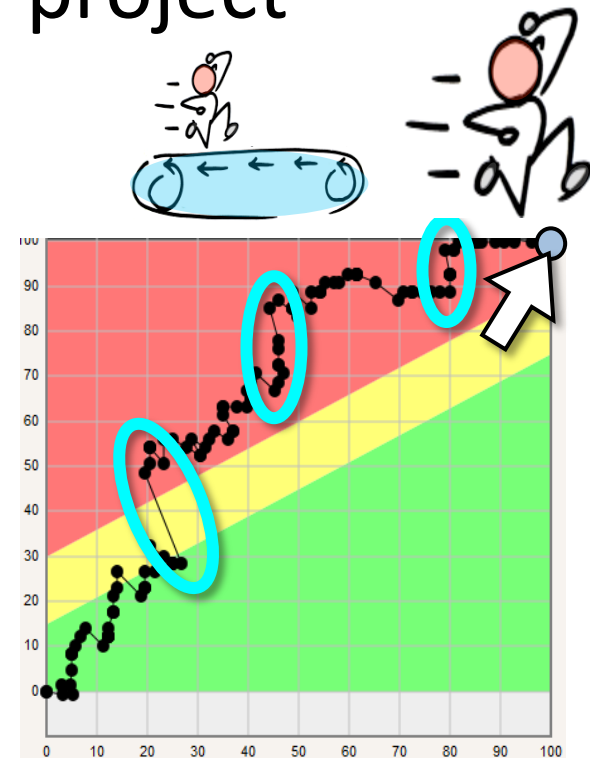


# Conclusions (cont.)

2016 TOCICO International Conference

- Although threatening situations were experienced several times, the project completed on the due date (with only 24 “min” remaining!)
- Relevant recovery actions were taken in a timely manner based on Velocity Based Buffer Management (VBBM)

## VBBM Works!

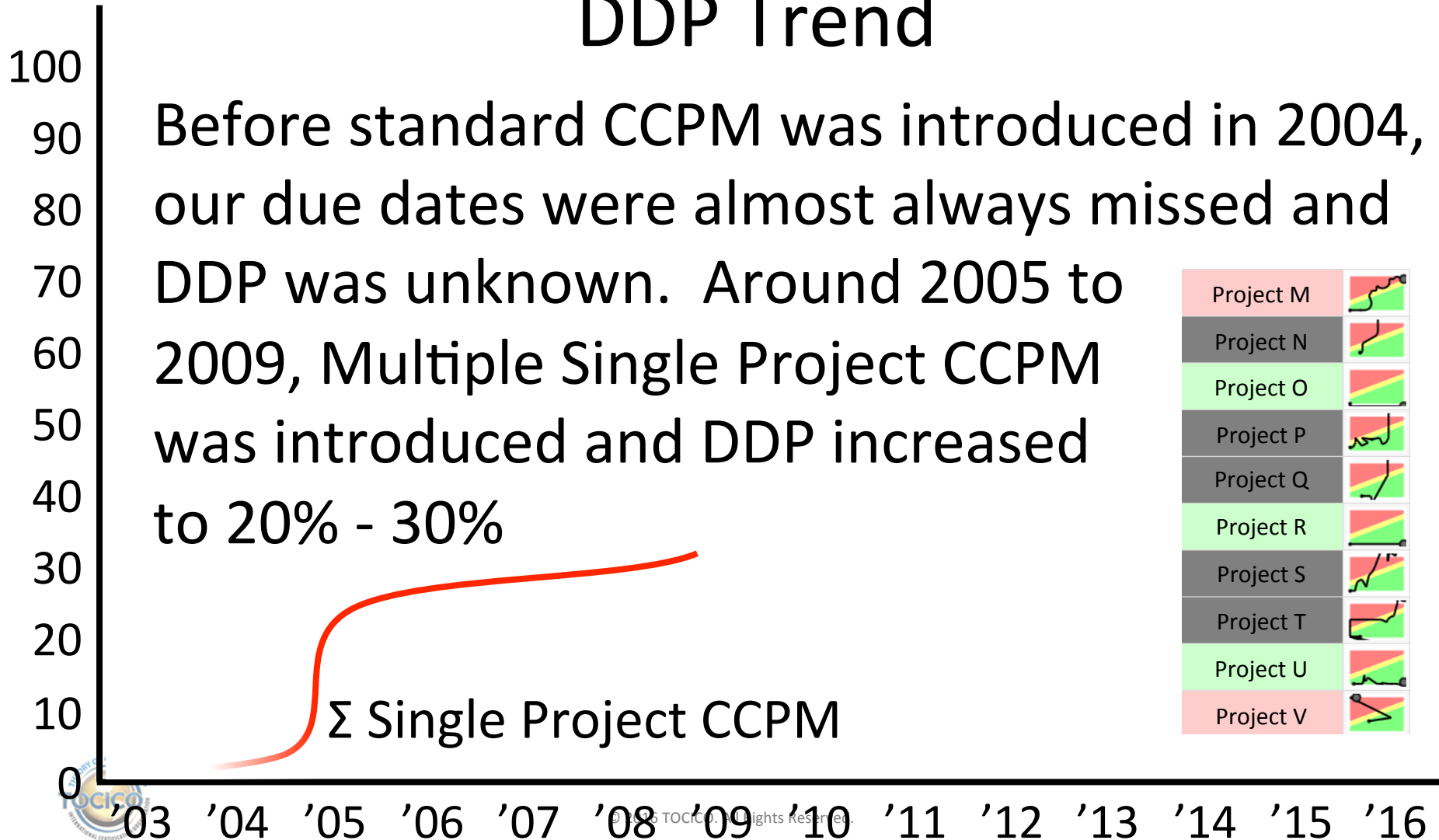


# Conclusions (cont.)

2016 TOCICO International Conference

## DDP Trend

Before standard CCPM was introduced in 2004, our due dates were almost always missed and DDP was unknown. Around 2005 to 2009, Multiple Single Project CCPM was introduced and DDP increased to 20% - 30%

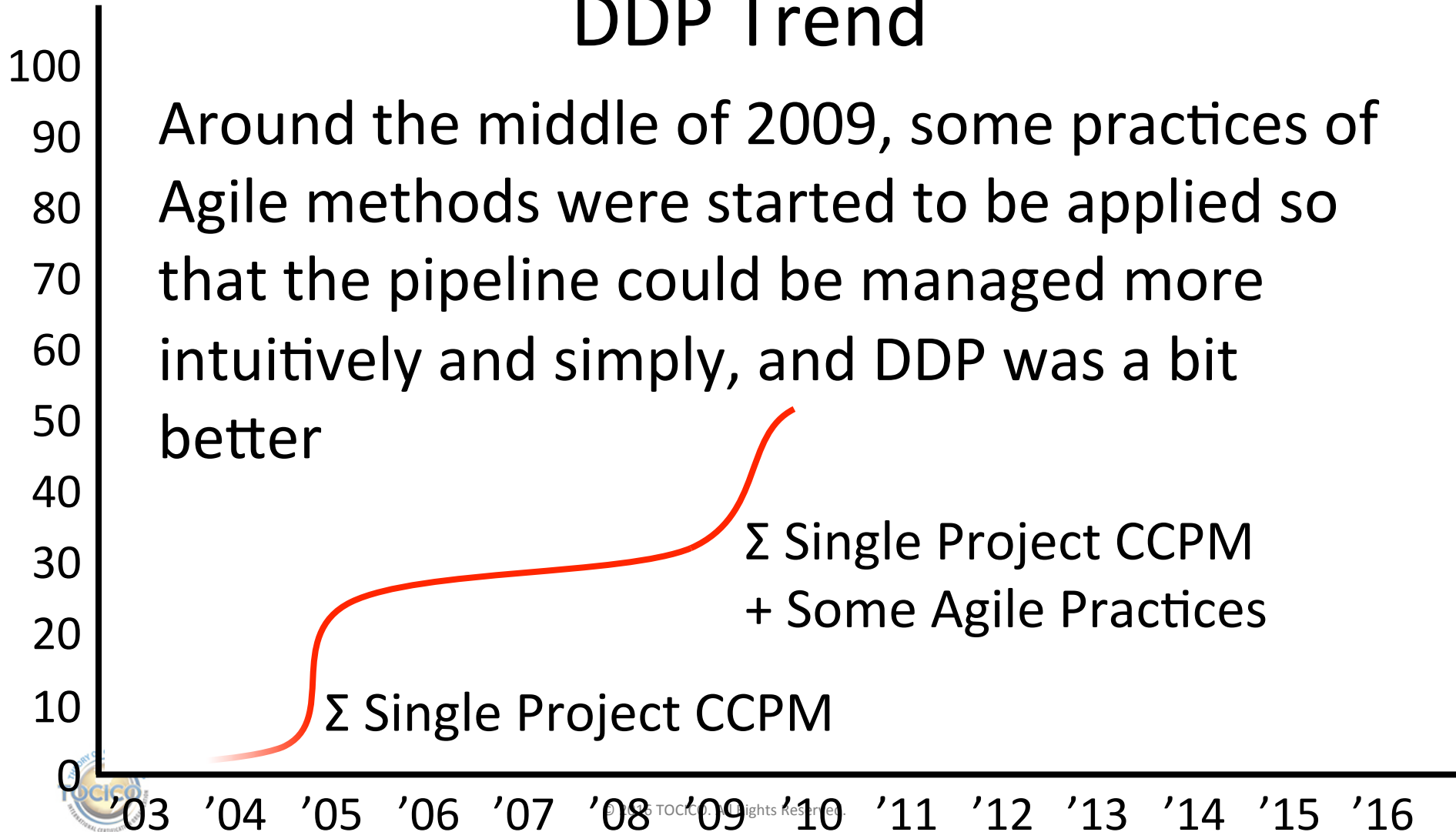


# Conclusions (cont.)

2016 TOCICO International Conference

## DDP Trend

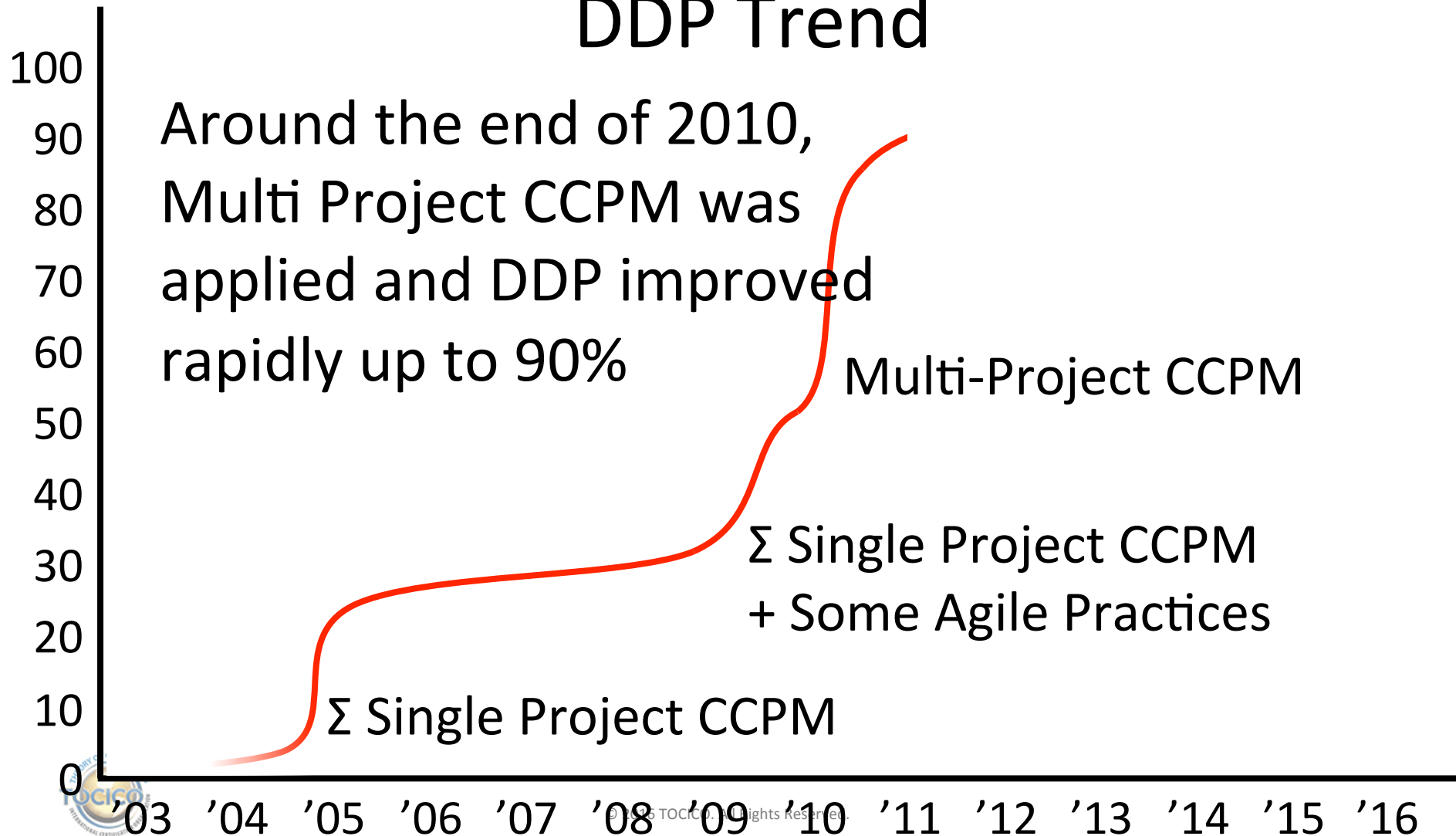
Around the middle of 2009, some practices of Agile methods were started to be applied so that the pipeline could be managed more intuitively and simply, and DDP was a bit better



# Conclusions (cont.)

2016 TOCICO International Conference

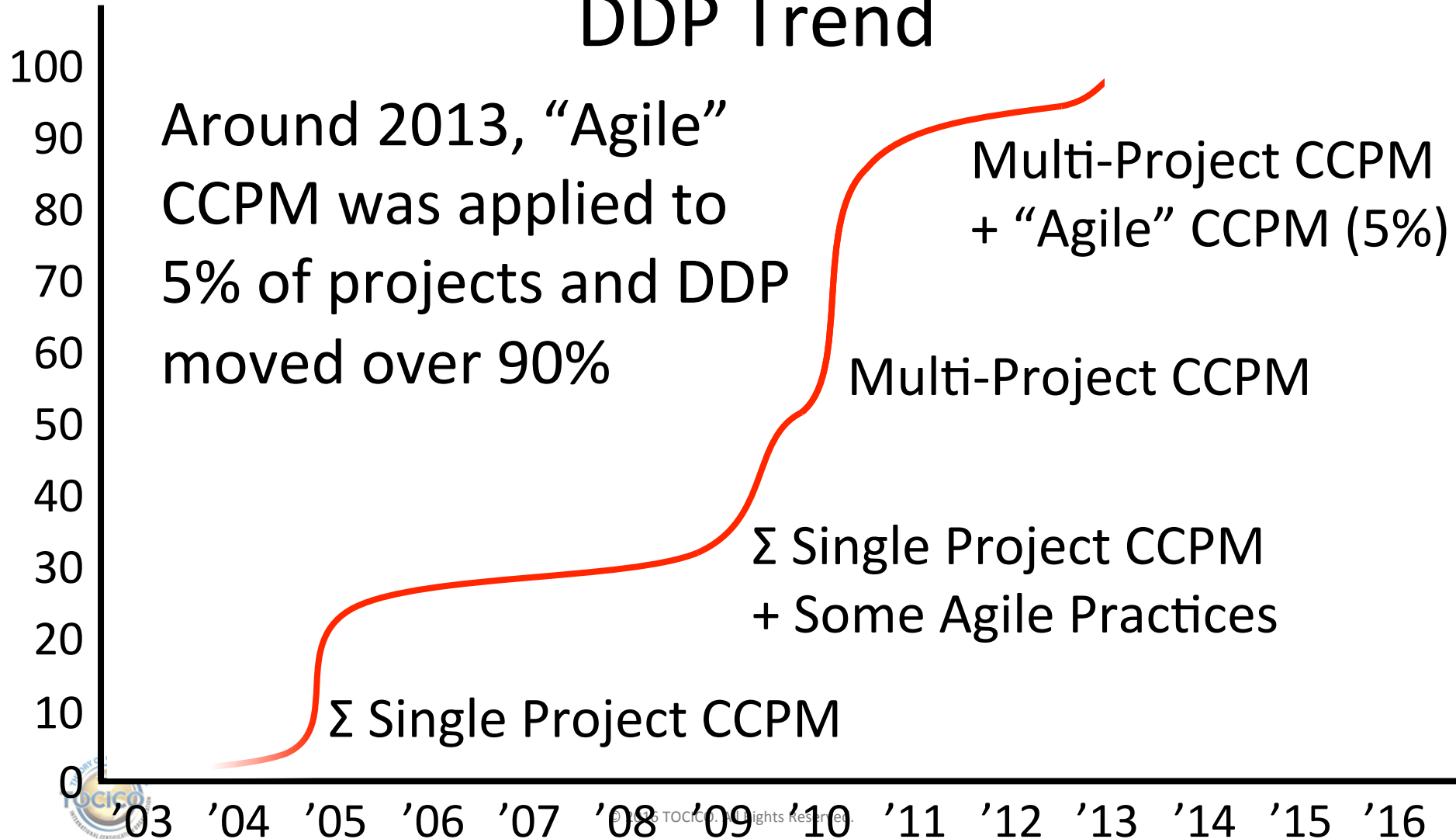
## DDP Trend



# Conclusions (cont.)

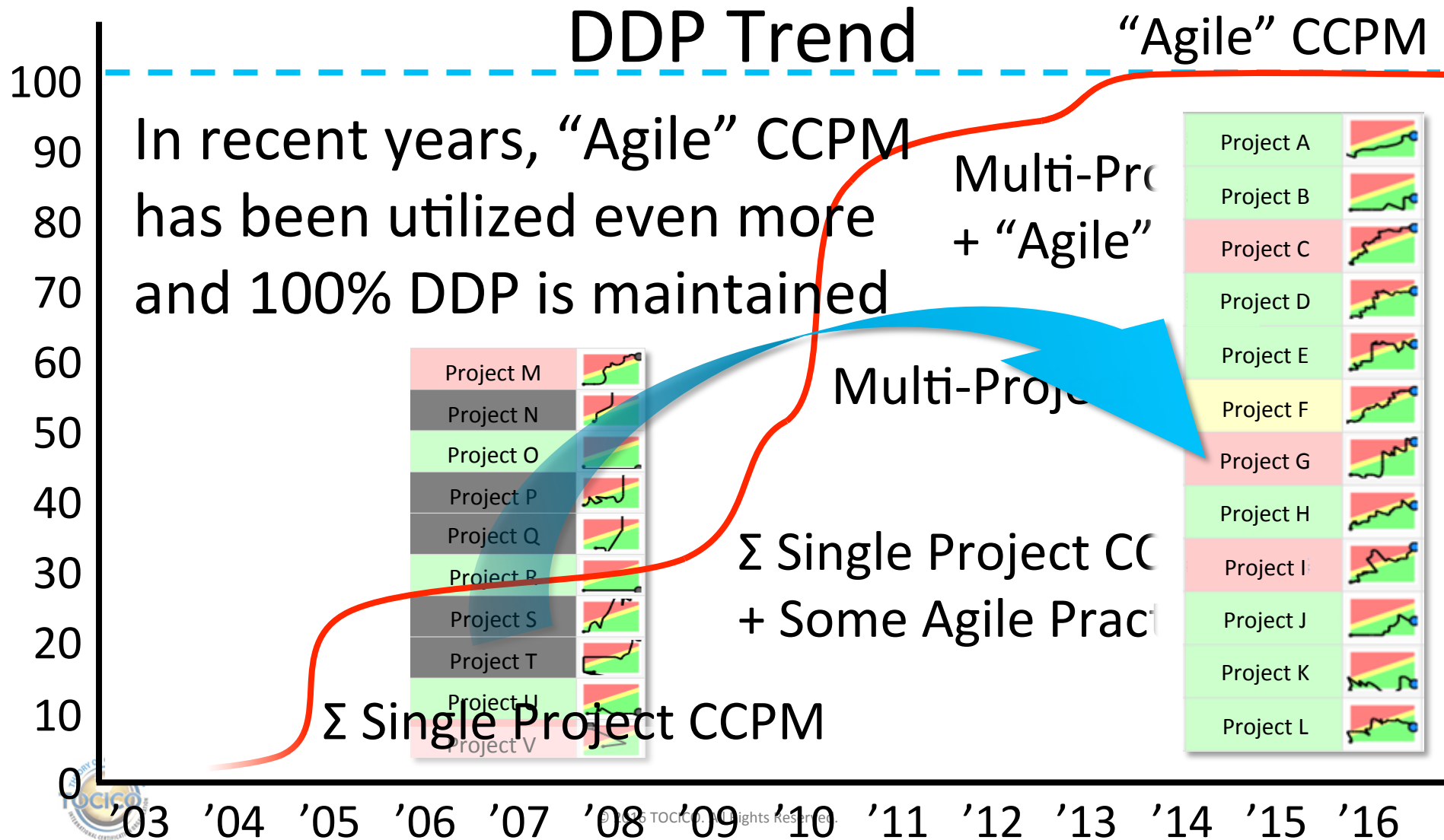
2016 TOCICO International Conference

## DDP Trend



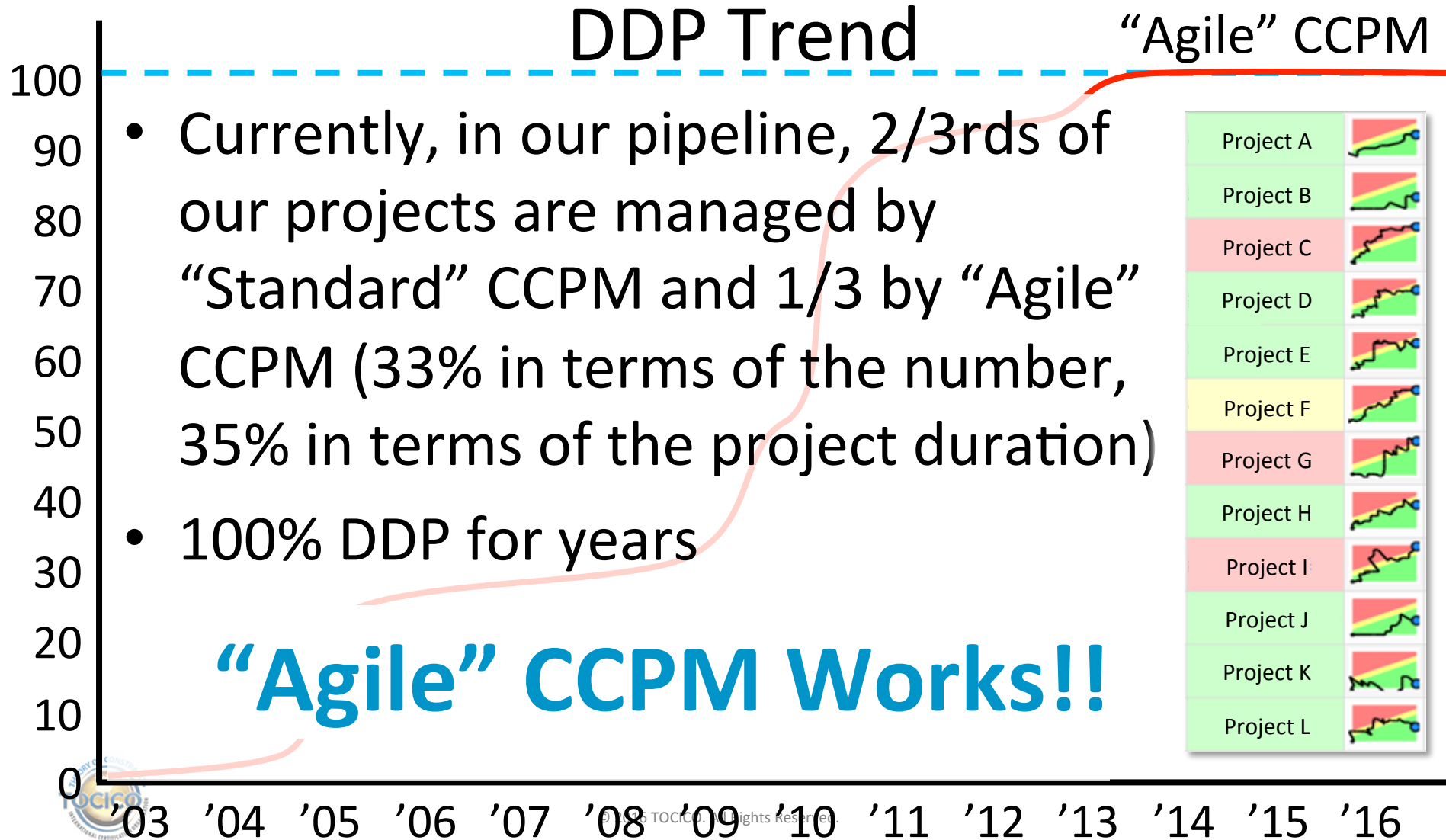
# Conclusions (cont.)

2016 TOCICO International Conference



# Conclusions (cont.)

2016 TOCICO International Conference







# Conclusions (cont.)

2016 TOCICO International Conference

**“Agile” CCPM is** an enhancement of CCPM. Although we use slightly different techniques in certain situations, none of the fundamental rules or benefits of CCPM are sacrificed, and it is **truly “CCPM”**



# Conclusions (cont.)

2016 TOCICO International Conference

Which means that in a software development environment,  
**CCPM** enables us to successfully complete challenging projects, and maintain 100% DDP for years



# Conclusions (cont.)

2016 TOCICO International Conference

If you identify with our approach,  
believe in the power of CCPM and  
believe that there is still more  
room for maximizing the flow of  
the system, I strongly recommend  
you to try **“Agile” CCPM**



# Thank You

2016 TOCICO International Conference

**I hope “Agile” CCPM will  
enable a lot of success for  
you in the near future!**

 [ujigawa@beingcorp.co.jp](mailto:ujigawa@beingcorp.co.jp)

 [www.linkedin.com/in/k1ujigawa](http://www.linkedin.com/in/k1ujigawa)

# About Koichi Ujigawa

2016 TOCICO International Conference



Currently leading a software engineering div. at Being Co., Ltd. to develop TOC solutions that help both TOC consultants and end users implement TOC at the end users' site.

In 1993, joined IHI Corporation in Japan and worked for the shipbuilding division for about 14 years. Engaged in software development for various in-house computer systems, such as a Material Control System, Advanced Planning System, CAD for hull structure, etc.

Involved in various systems' projects for the company, and contributed with strong commitment to successful implementation of Kaizen activities, including productivity improvement of knowledge workers and introduction of TPS (Toyota Production System) to Gemba (manufacturing workplace).

During the course of such activities and continuous efforts on Kaizen, encountered TOC in 2001, then was challenged to make various POOGI plans based on the Thinking Process, implementing both DBR for production lines and CCPM for the shipbuilding design department.

As the consequence of the continuing pursuit of TOC, joined Being Co., Ltd. in 2007.

As new knowledge, provided presentations of “Agile” CCPM at 2012 – 2014 TOCICO conferences.

