
DEFENCE
R&D
DÉFENSE

State Machine Modeling of TPED and TPPU

Mr. Ron Funk (DRDC)

Mr. Rick Sorensen (Vitech Corp.)

NDIA 8th Sys Engr Conference, San Diego

27 October 2005

Defence Research and Development Canada

Recherche et développement pour la défense Canada

Canada

State Machine Modeling of TPED and TPPU

Mr. Ron Funk (DRDC)

Mr. Rick Sorensen (Vitech Corp.)

NDIA 8th Sys Engr Conference, San Diego

27 October 2005

Defence Research and Development Canada

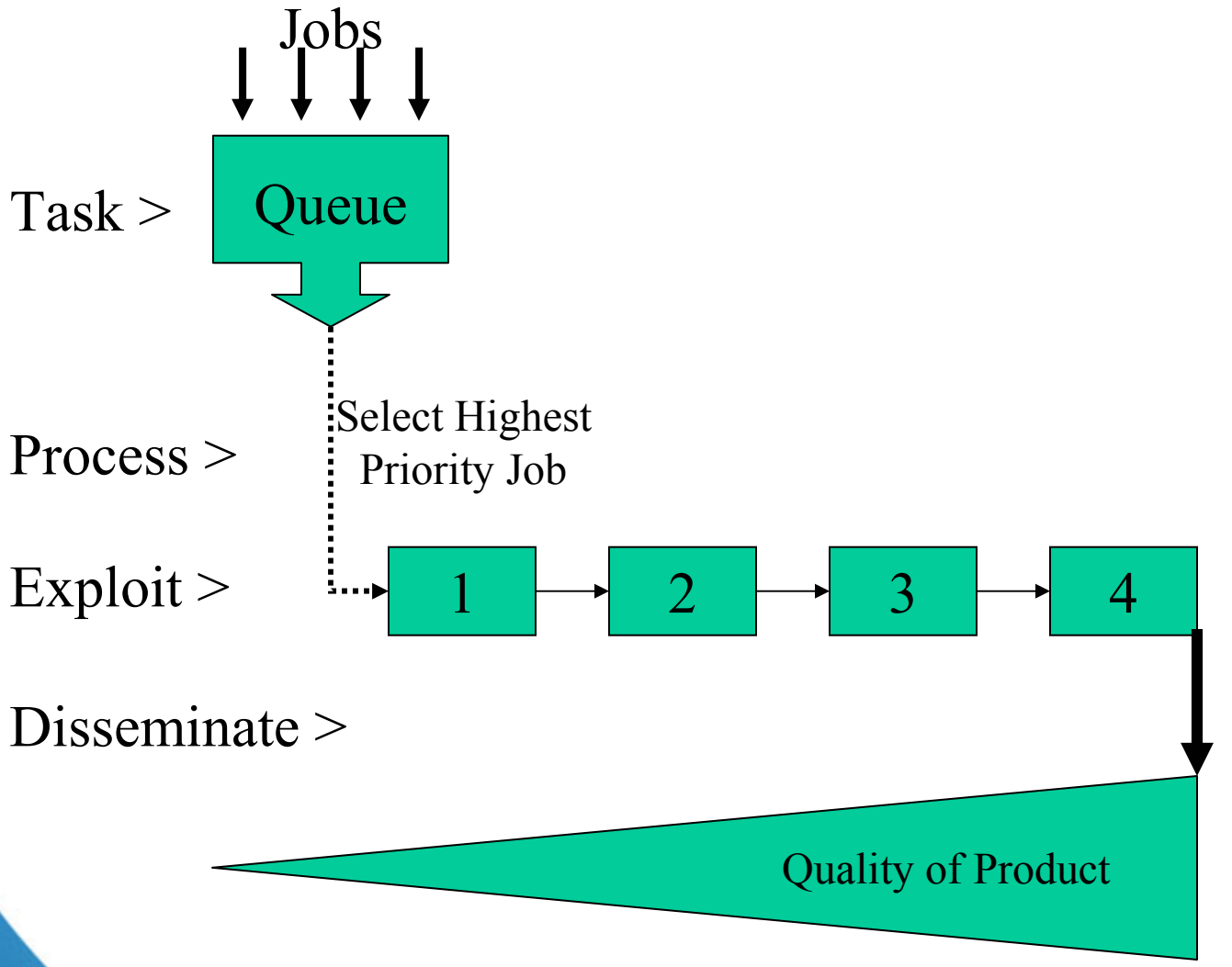
Recherche et développement pour la défense Canada

Canada

Outline of Presentation

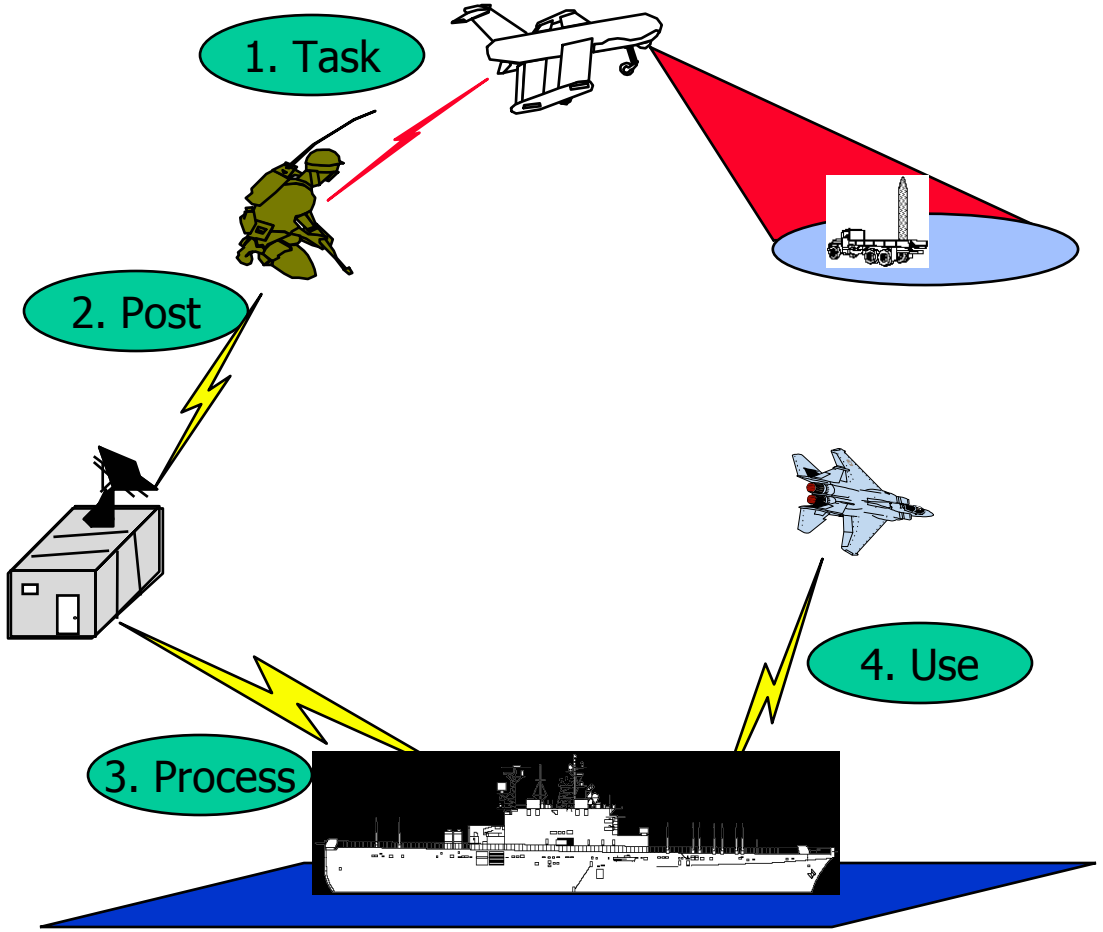
- Operational Activity Modeling
  - Task, Process, Exploit, Disseminate (TPED)
  - Task, Post, Process, Use (TPPU)
- State Machine Concepts
- Description of TPPU State Machine
- Modeling Process
- Examples
- Q&A

How Task, Process, Exploit and Disseminate (TPED) Manages Jobs

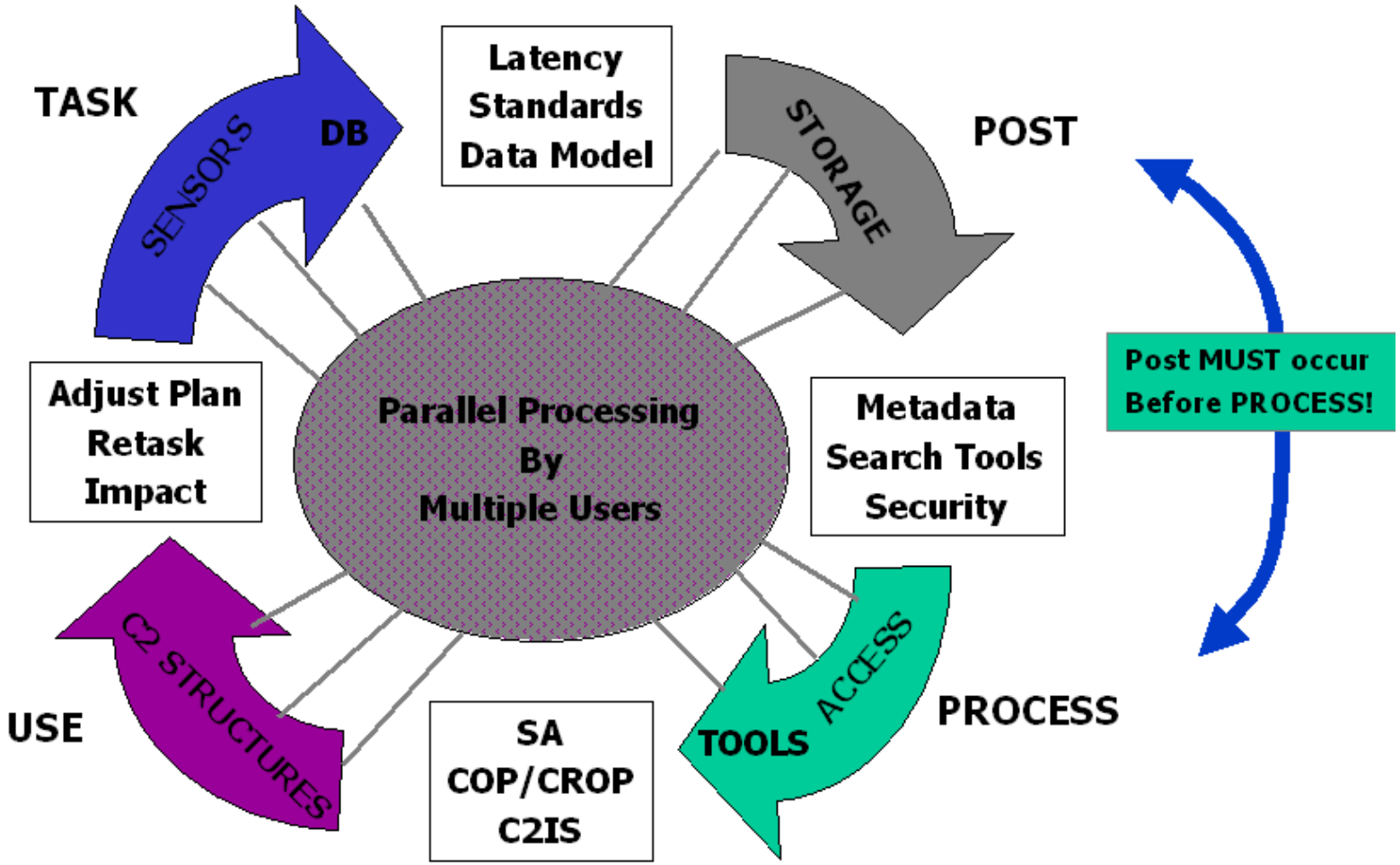


Task, Post, Process and Use (TPPU): Single Iteration

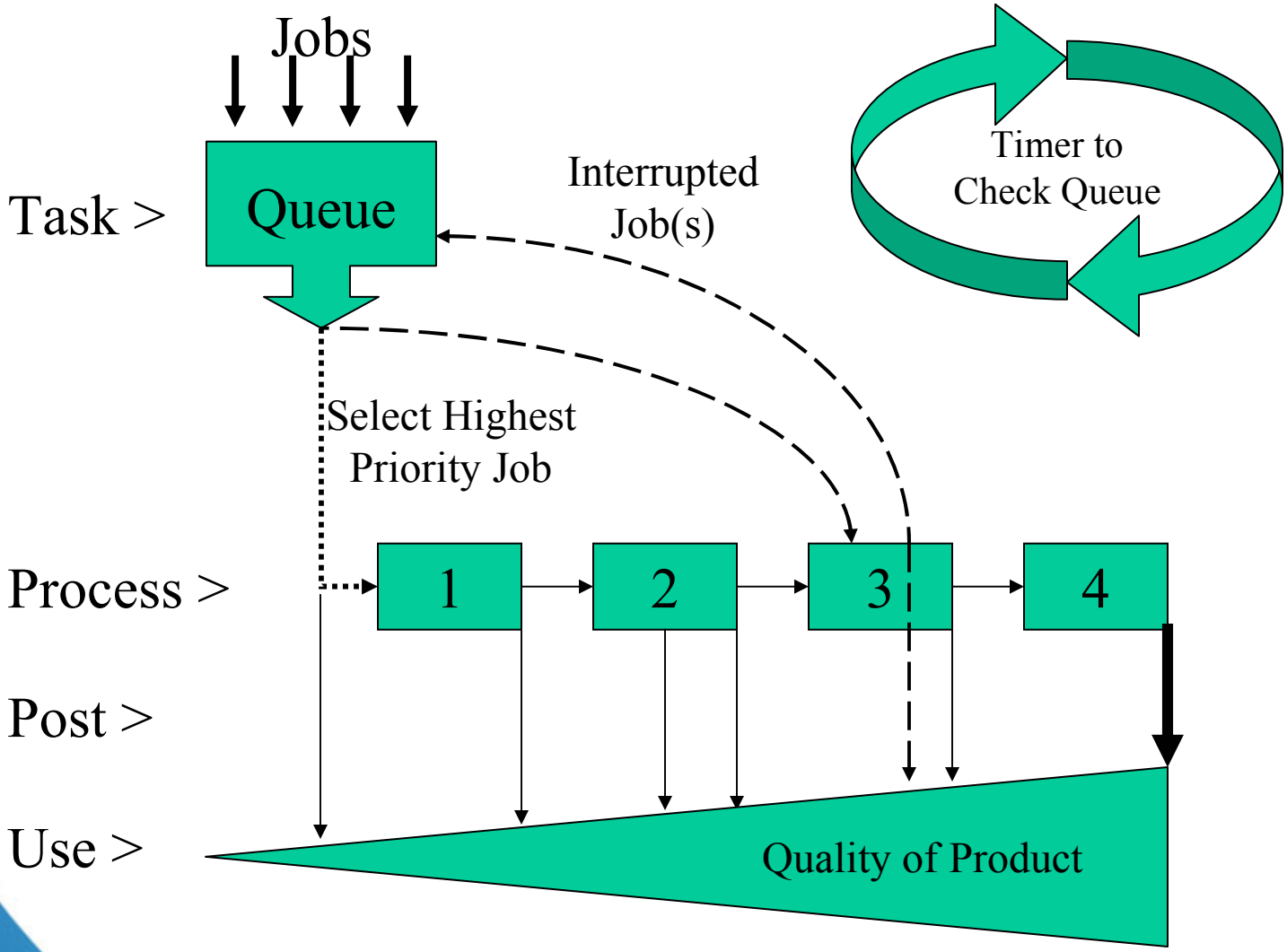
- TPPU
  - Task
  - Post
  - Process
  - Use



TPPU as Repeating Sequence of Activities



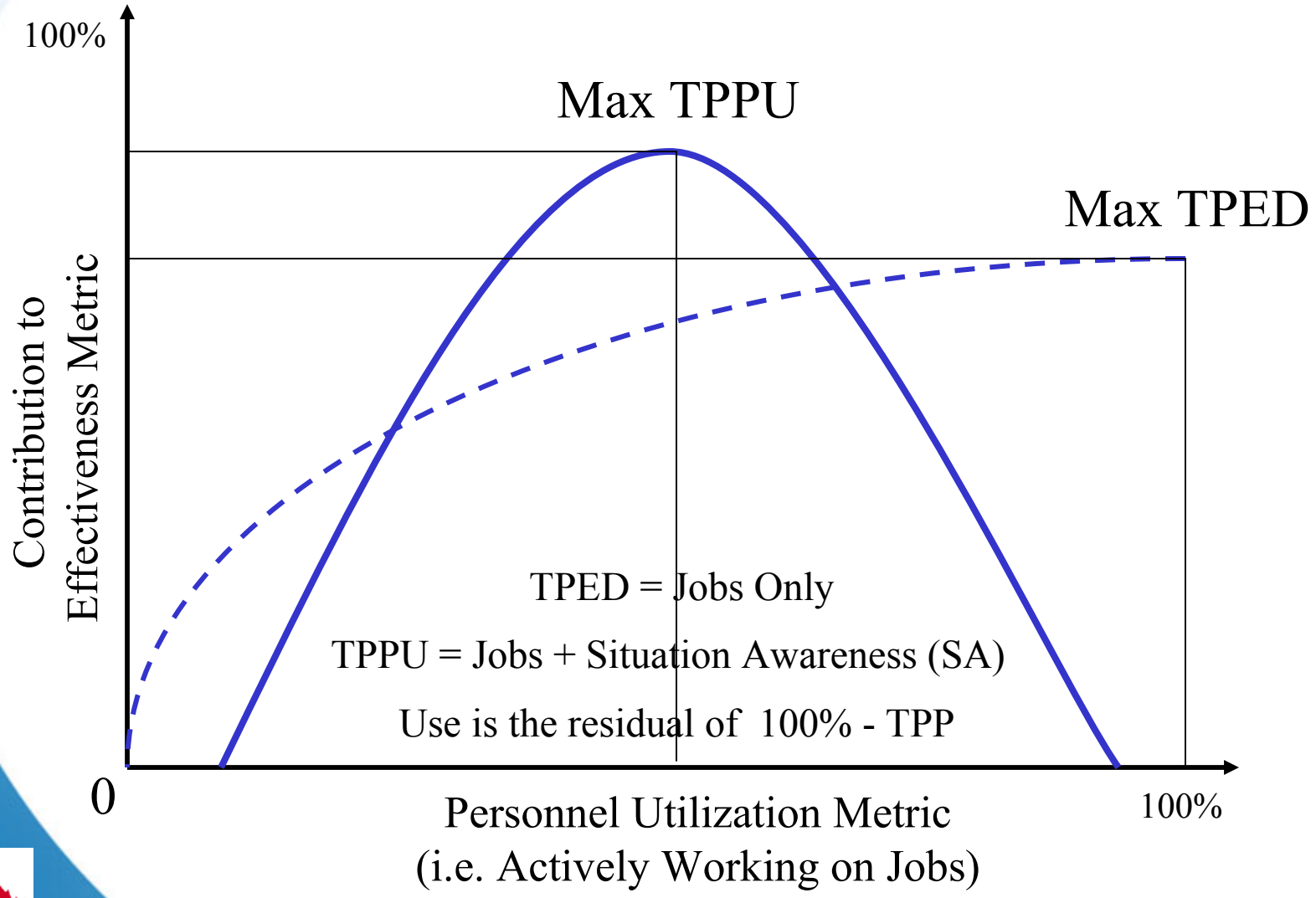
How TPPU Manages Jobs



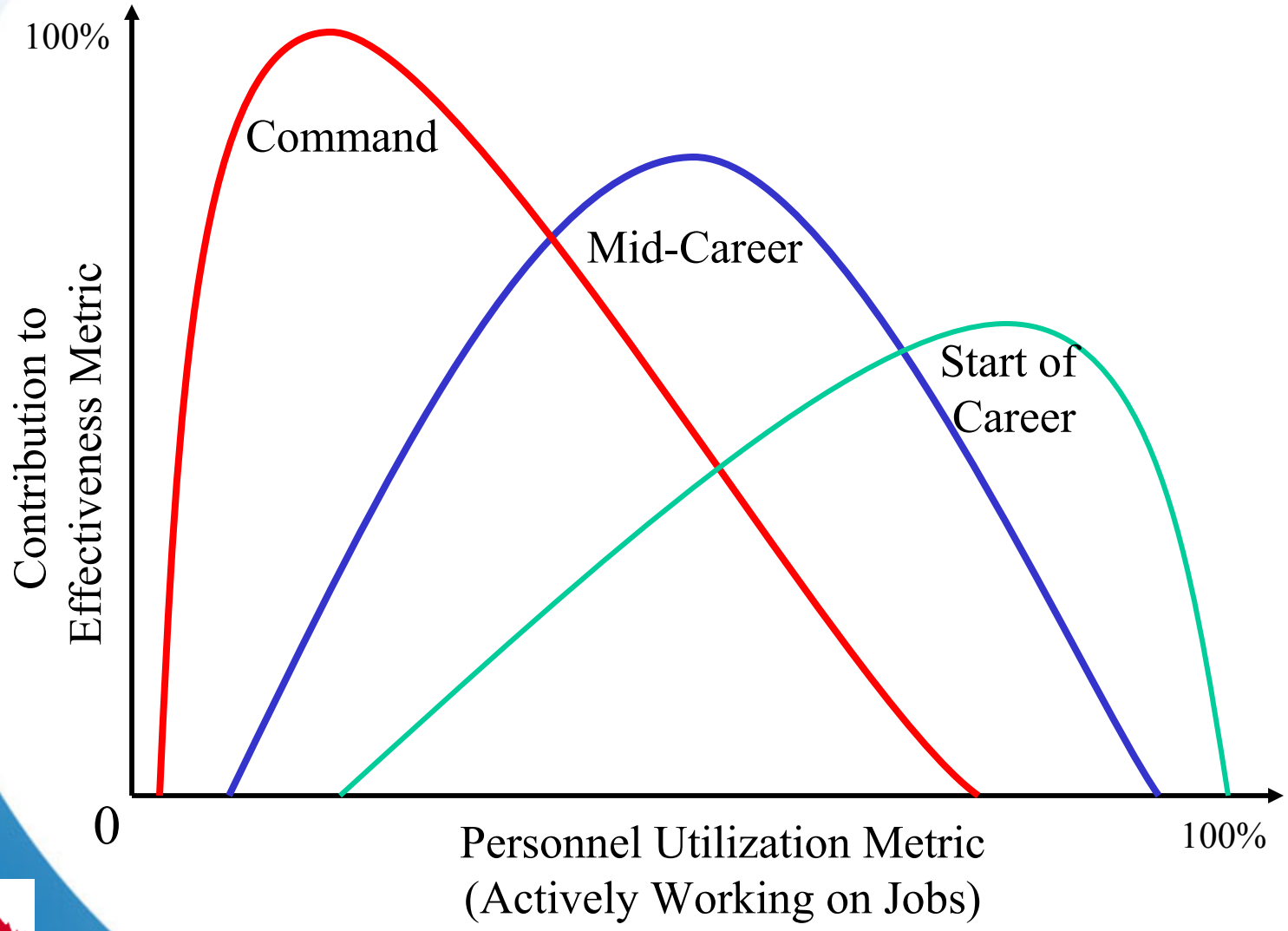
How TPED/TPPU Deal With Situation Awareness (SA)

- TPED
  - Centralized control of all activity
  - Only describes assigned jobs
  - Worker utilization maximized when jobs are 100% of time
  - No way for operators to do SA except as a separate job
- TPPU
  - Decentralized and uncoordinated independent activities
  - Becomes self synchronizing when queue checked frequently
  - Cycle based on doing jobs and maintaining SA
    - Jobs are composed of TPP, but not Use
    - Use is residual time when no jobs (i.e. 100% - TPP)
    - SA automatically distributed everywhere as Use
  - Effectiveness is maximized when job utilization < 100%

How Utilization Metric Differs from TPED to TPPU



How TPPU Utilization Metric Shifts During Career



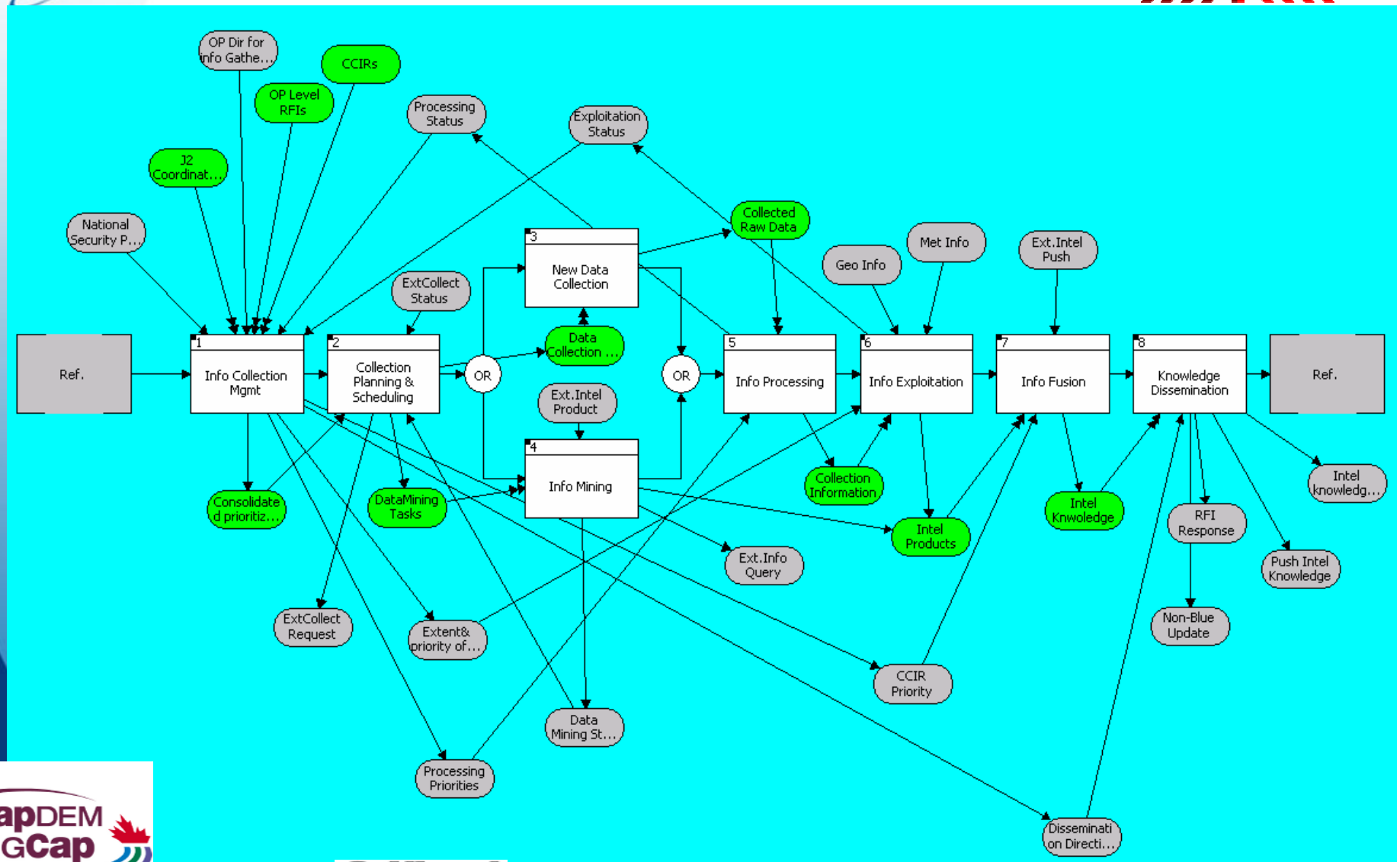
TPPU Business Rules Being Modeled

- Concurrent Independent Cycles are modeled:
  - Cycles automatically repeated
  - Activity logic is easily amended
  - Model is scalable
- Improved realism of business rules by explicitly modeling:
  - Regular checking of queue for highest priority job
  - Pre-empted jobs returned to queue to complete later
    - Credited for work already completed
  - Job status tracked continuously once it enters queue
    - Checks remaining time needed to complete job
    - Jobs abandoned once expected completion time exceeds deadline

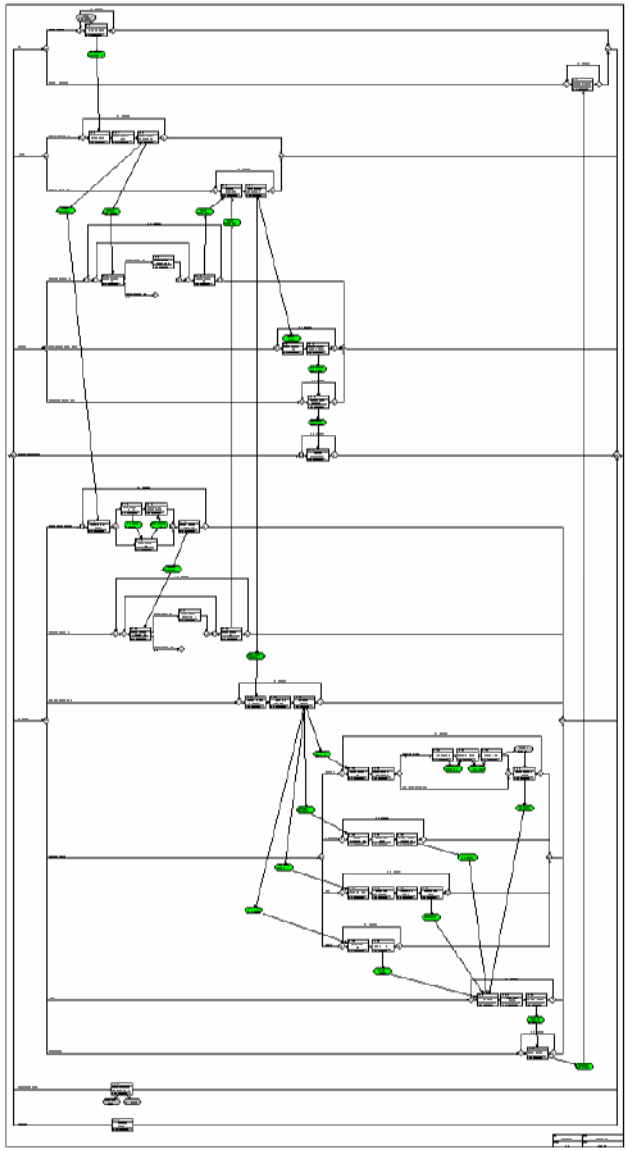
DODAF Operational Activities (OV-5) Modeling Efforts

- Summer 2003
  - Conceptual TPED from first principles
    - Elegant model but too conceptual to actually work
- Spring 2004
  - Data Fusion Using TPED
    - Integrated behavior of basic TPED components
    - Data stovepipes shown but no description of fusion
- Summer 2004
  - Integrated ISR Architecture Examples
    - Modeled sequence of C2 changes for UAV flight
    - TPPU used to describe OPCEN activities
- Spring 2005 to Present
  - Describe TPPU as State Machine
    - Articulate TPPU with robust business rules
    - Model can also handle TPED

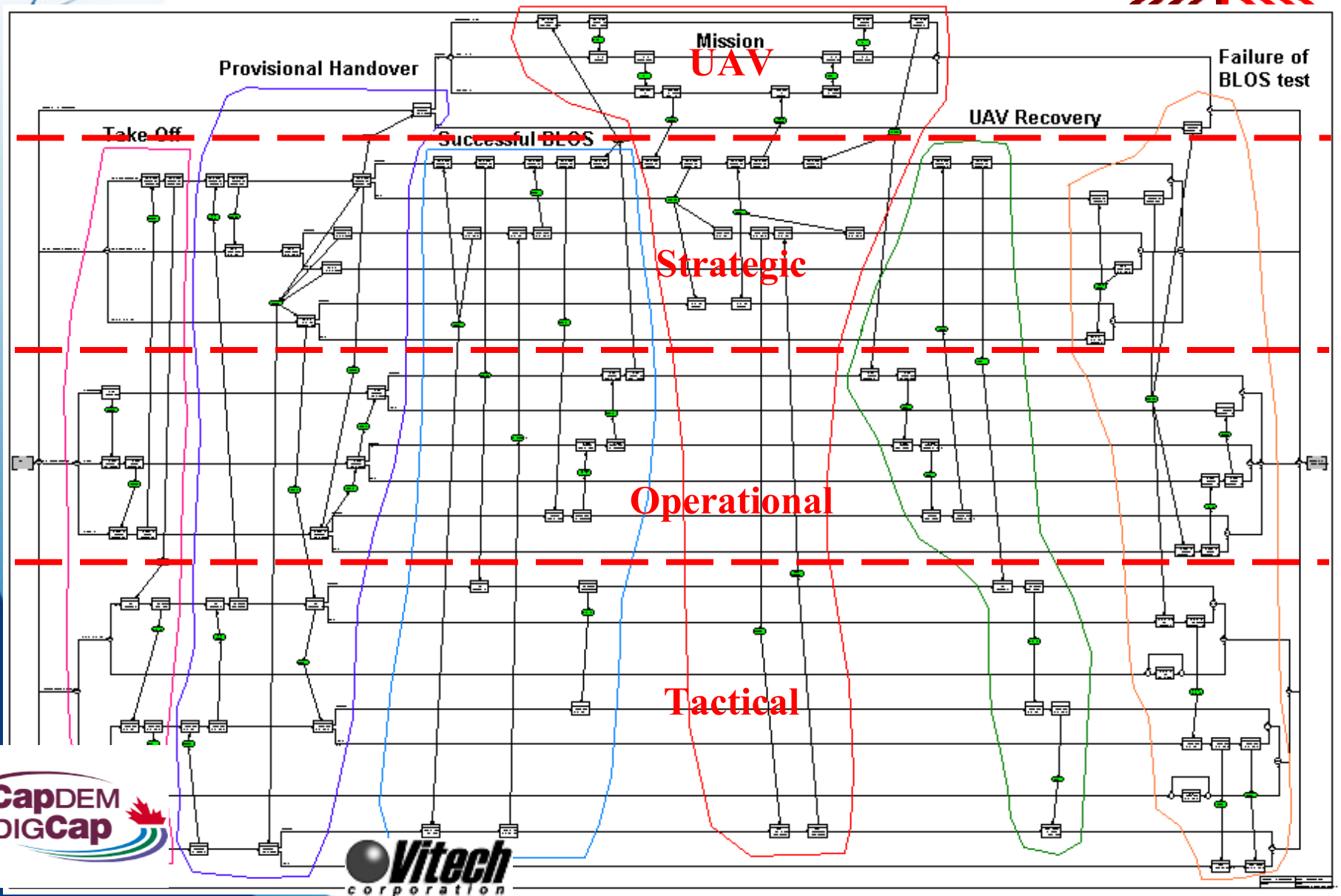
CORE® Model of Conceptual TPED: Summer 2003



CORE Model of TPED Work Flow: Spring 2004



CORE Model of UAV C2 Changes: Summer 2004

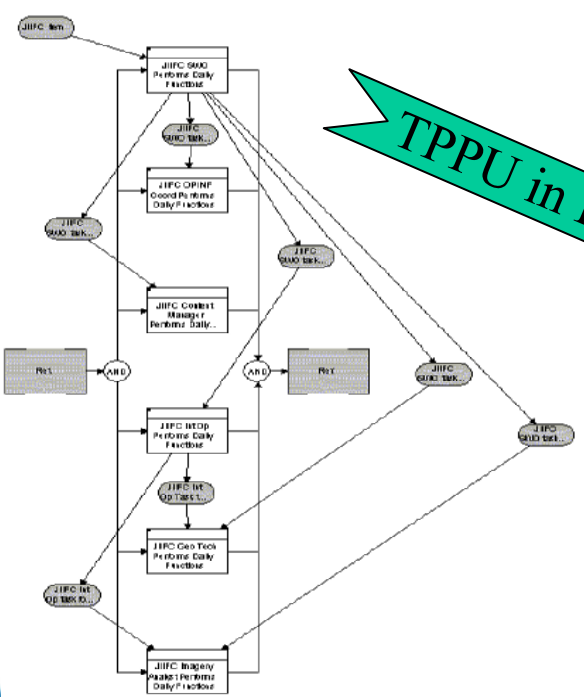


CapDEM DIGCap

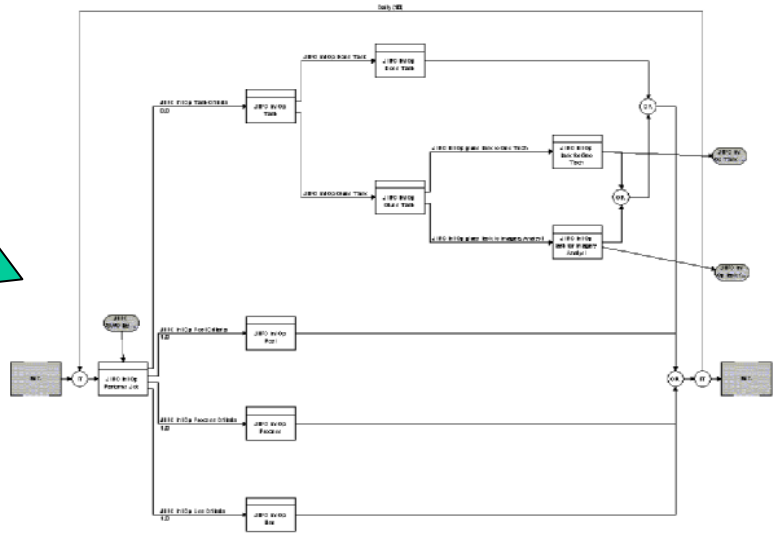
Vitech corporation

Initial CORE Model of TPPU Job Activity

Example of Top-Level Model



TPPU in Block



Example of TPPU within One Block

Modeling the Processes Using CORE

- CORE used to capture behavior models
  - Document known processes
  - Capture timelines, data flow, business rules
  - Expose unknowns for further exploration
- COREsim used to execute models
  - Validate our understanding
  - Exercise to-be processes

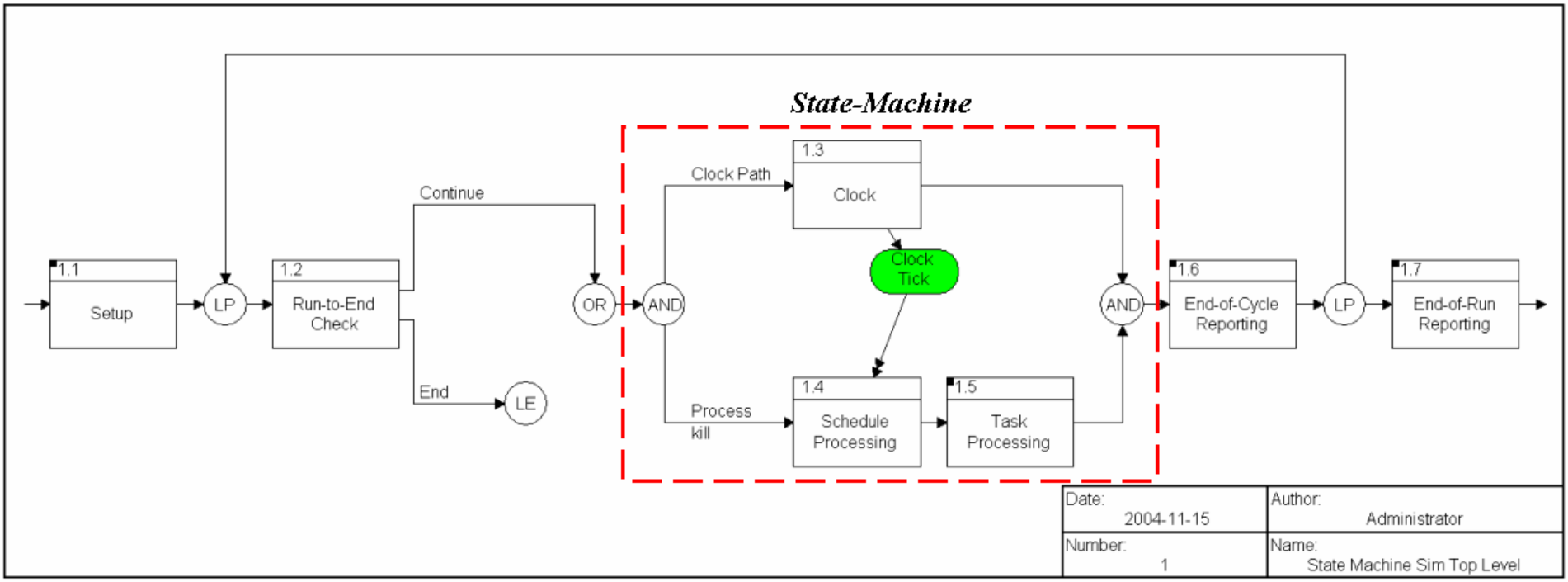
Process to Efficiently Build Behavior & SM Models

1. **Threads** : Work process of each job by an operator
   - Articulates specific job activities for each operator
2. **Integrated** : Common themes between jobs & operators
   - Calculates minimum resource demand
3. **Allocated** : Differentiation of skills
   - Added cost of specialization
   - Determine any offsets when generic work is done during idle time
4. **State Machine** : Schedule and track concurrent activity
   - Impact of Business Rules (i.e. Job priorities & time remaining)
   - Caution: Logic is not as visible to users as usual behavior models

State Machine Characteristics

- State Machine has the following key **characteristics**:
  - Initial state or record of something stored someplace
  - Set of possible input events
  - Set of new states that may result from the input
  - Set of possible actions or output events that result
  - It is composed of the following elements:
    - Description of the initial state;
    - Set of states;
    - Set of input events;
    - Set of output events;
    - Function that maps states and input to output;
    - State transition function.

CORE Model of State Machine: Overall Structure



- Excel spreadsheets are used to store model data for:
  - Schedule of Events (i.e. time and location of next work item)
  - Initial Status of Threads (i.e. result of occurred before)
  - Thread Attributes (i.e. duration of work)
- Event states and schedule updated using COREscript

How Specialization Can Affect Personnel Requirements

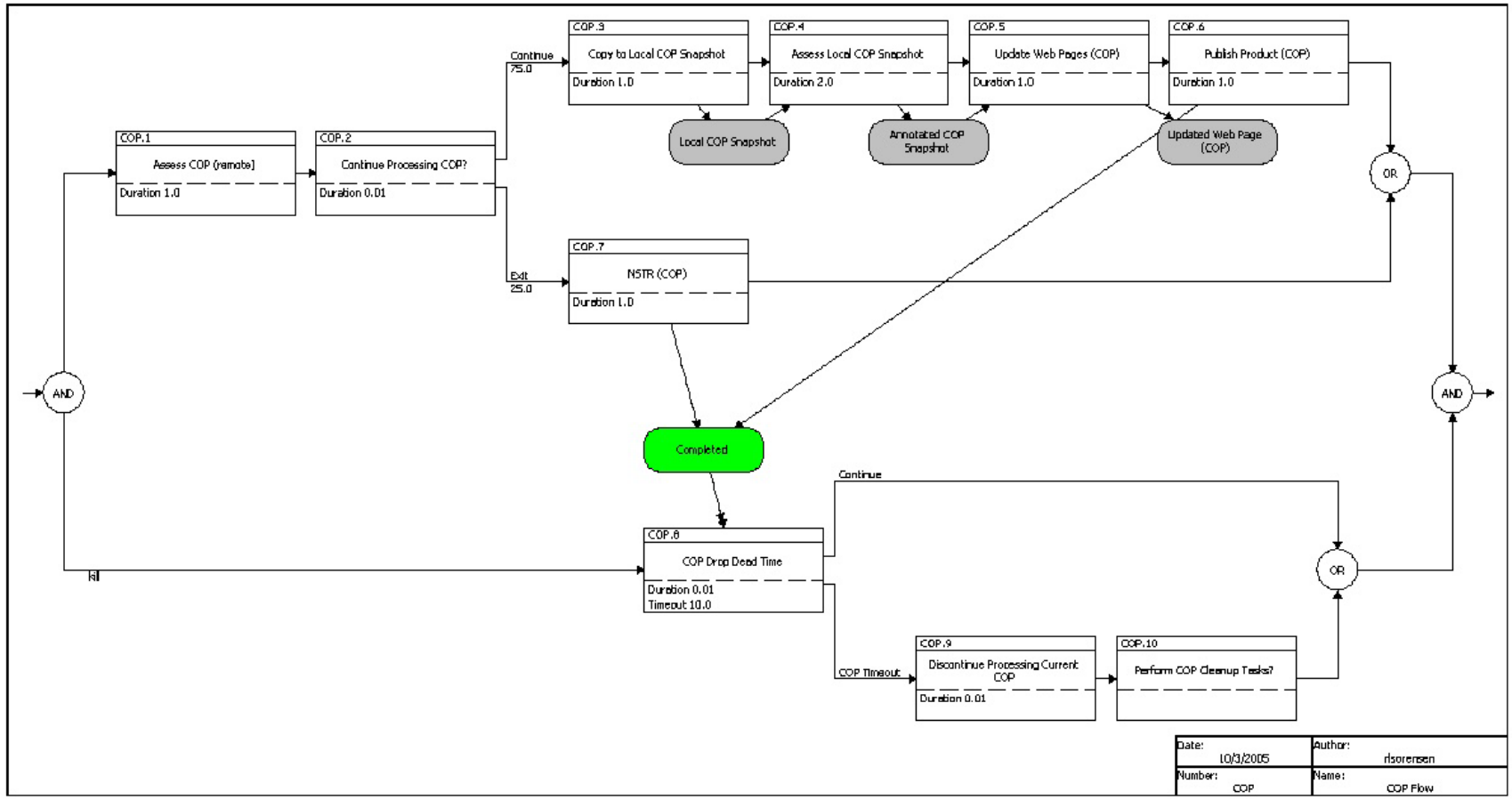
Functionality	Original Personnel Demand (No Transfer)	Specialists Do Generic Work 50% of Idle Time	Personnel Required (if 50% transferred)	Specialists Do Generic Work 100% of Idle Time	Personnel Required (if 100% transferred)
Generic	3.2	-0.7	2.5	-1.4	1.8
Specialist Type 1	0.2	+0.4	0.6	+0.8	1.0
Specialist Type 2	1.4	+0.3	1.7	+0.6	2.0
Bodies Needed	**7**		**6**		**5**

NOTE: Have to round up to account for fractional headcounts.
3.2 becomes 4, 0.2 becomes 1, 1.4 becomes 2, and total equals 7!

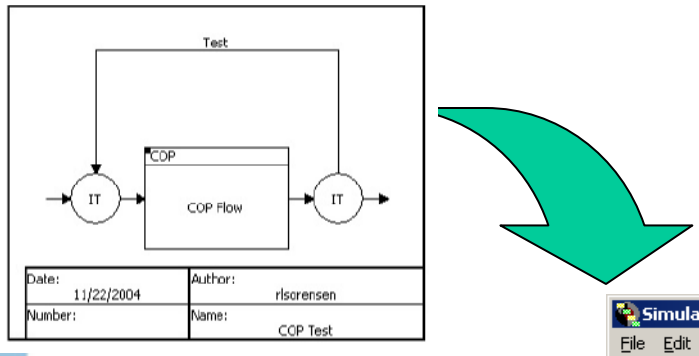
Process to Efficiently Build Behavior & SM Models

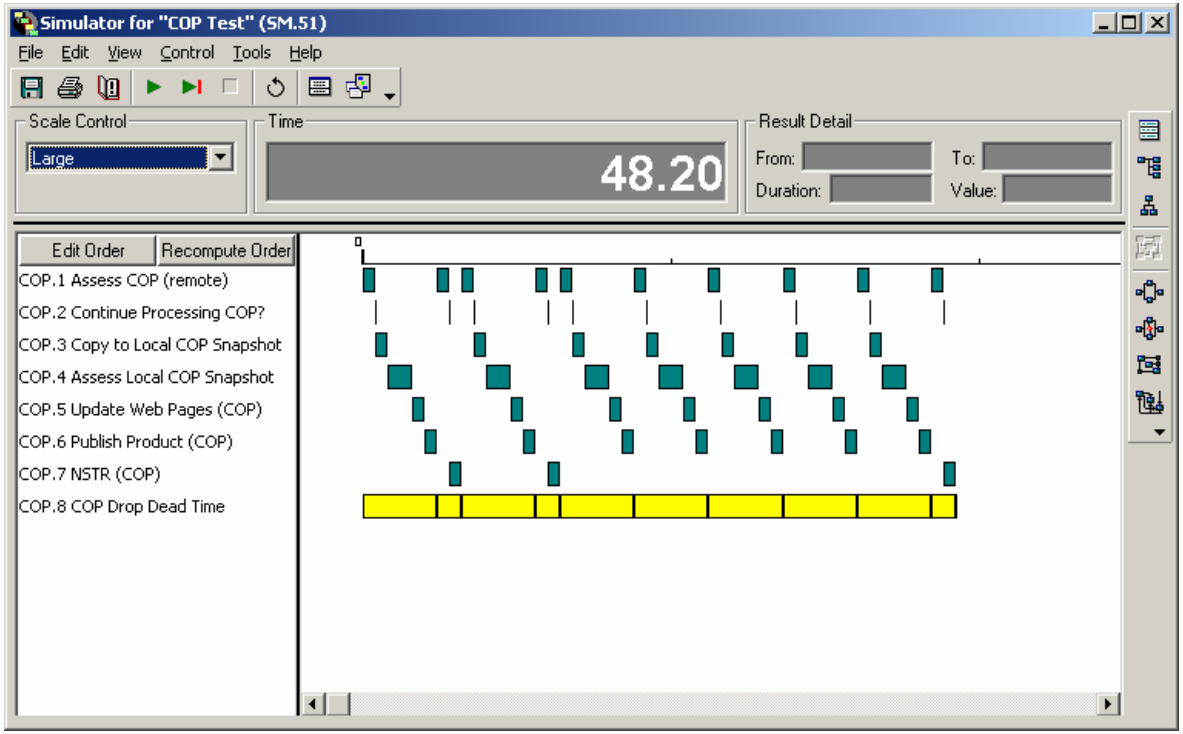
1. **Threads** : Work process of each job by an operator
   - Articulates specific job activities for each operator
2. **Integrated** : Common themes between jobs & operators
   - Calculates minimum resource demand
3. **Allocated** : Differentiation of skills
   - Added cost of specialization
   - Determine any offsets when generic work is done during idle time
4. **State Machine** : Schedule and track concurrent activity
   - Impact of Business Rules (i.e. Job priorities & time remaining)
   - Caution: Logic is not as visible to users as usual behavior models

Example Thread



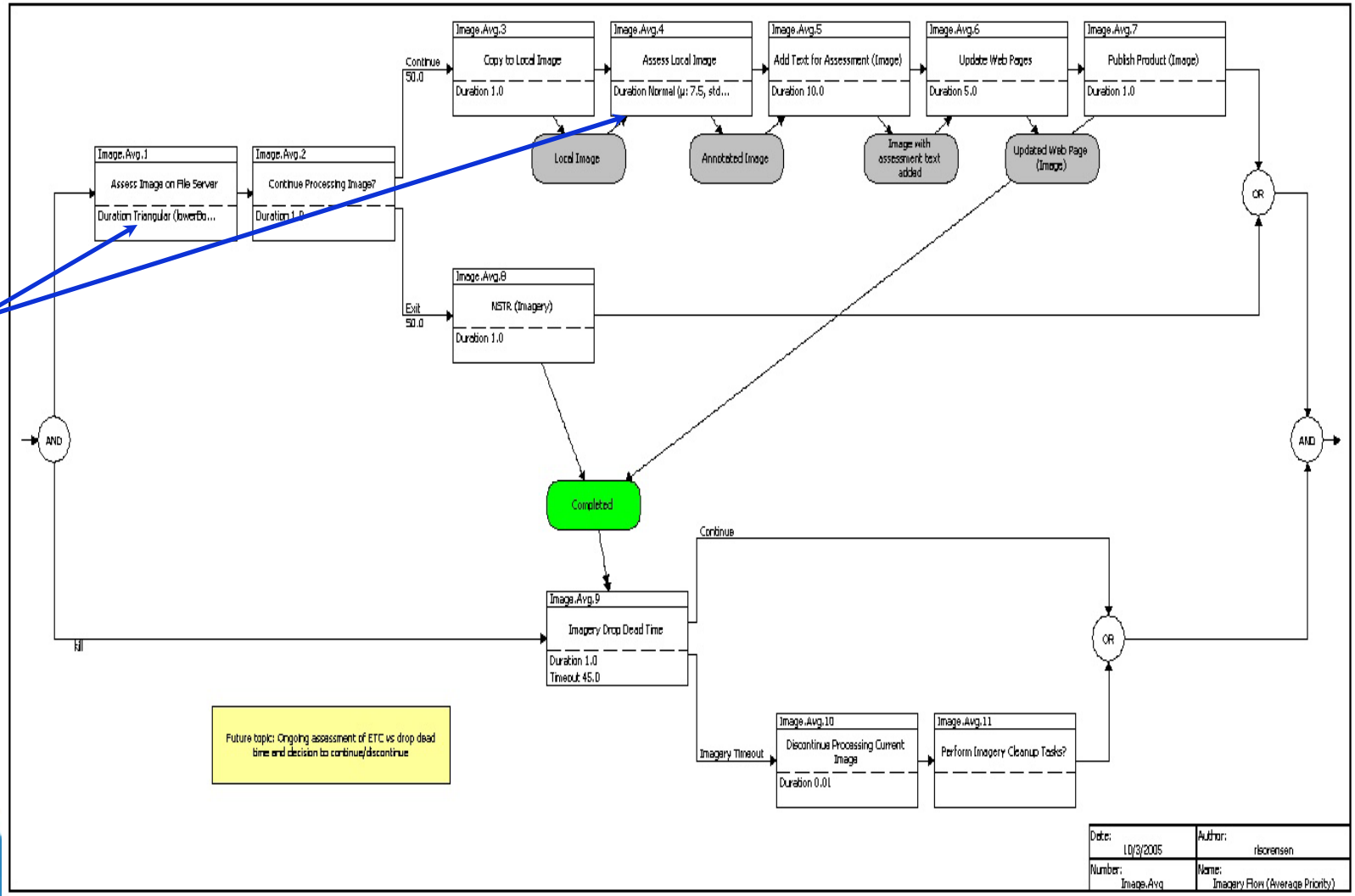
Testing


Test
COP
IT
COP Flow
IT
Date:
11/22/2004
Author:
rlsorensen
Number:
Name:
COP Test

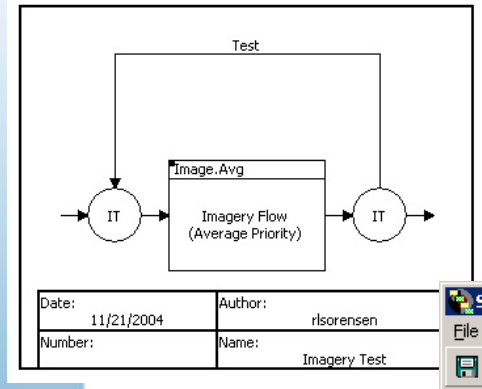

Simulator for "COP Test" (SM.51)
File Edit View Control Tools Help
Scale Control
Large
Time
48.20
Result Detail
From:
To:
Duration:
Value:
Edit Order
Recompute Order
COP.1 Assess COP (remote)
COP.2 Continue Processing COP?
COP.3 Copy to Local COP Snapshot
COP.4 Assess Local COP Snapshot
COP.5 Update Web Pages (COP)
COP.6 Publish Product (COP)
COP.7 NSTR (COP)
COP.8 COP Drop Dead Time

Example Thread [Take 2]

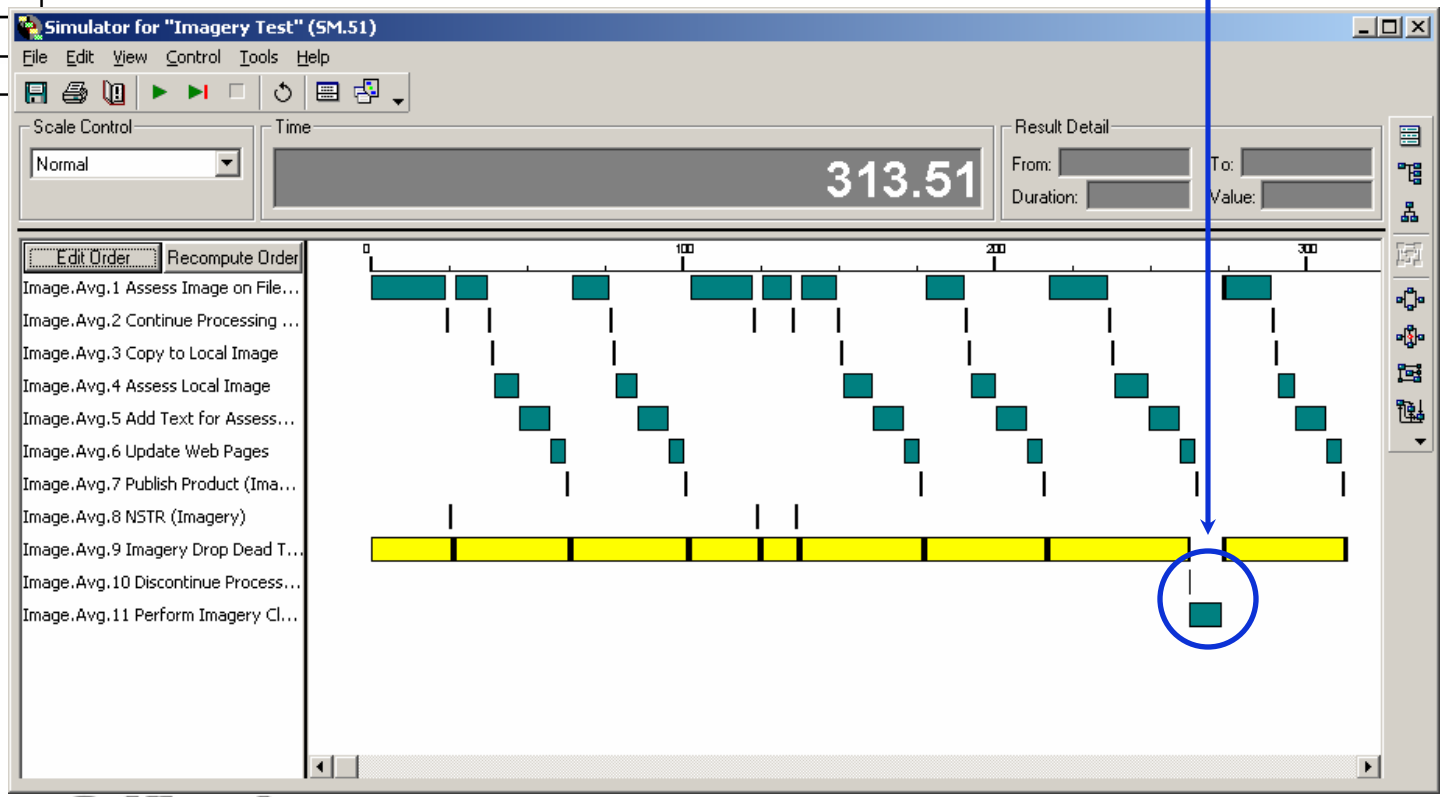
NOTE



Testing [Take 2]



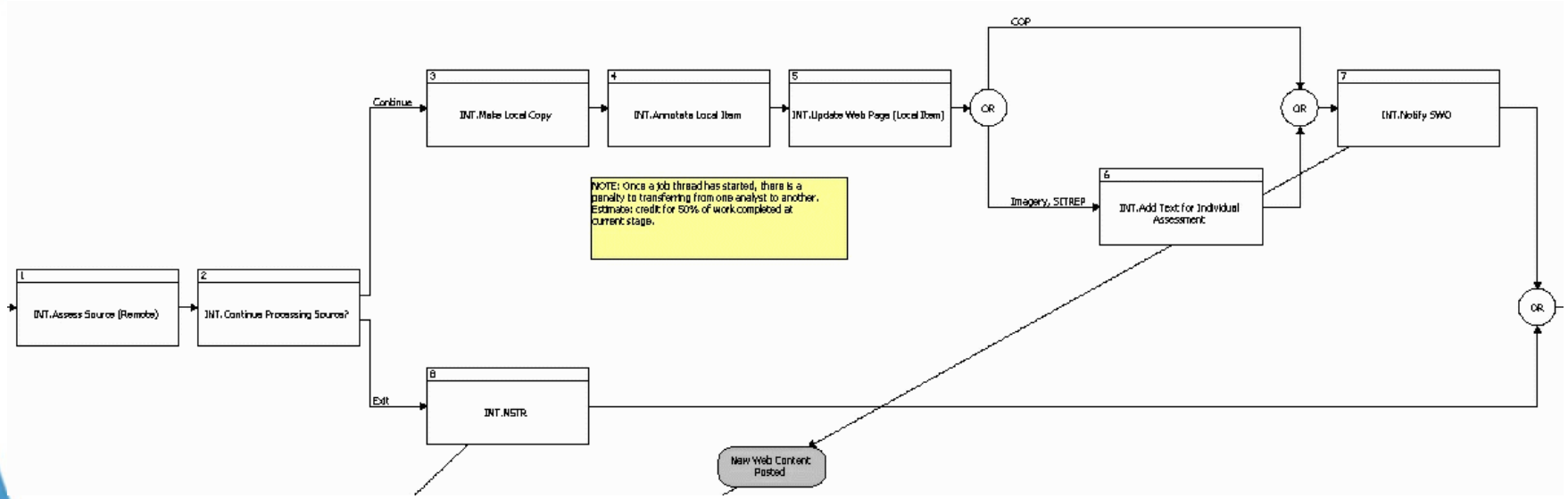
NOTE



Process to Efficiently Build Behavior & SM Models

1. **Threads** : Work process of each job by an operator
   - Articulates specific job activities for each operator
2. **Integrated** : Common themes between jobs & operators
   - Calculates minimum resource demand
3. **Allocated** : Differentiation of skills
   - Added cost of specialization
   - Determine any offsets when generic work is done during idle time
4. **State Machine** : Schedule and track concurrent activity
   - Impact of Business Rules (i.e. Job priorities & time remaining)
   - Caution: Logic is not as visible to users as usual behavior models

Integrated Behavior



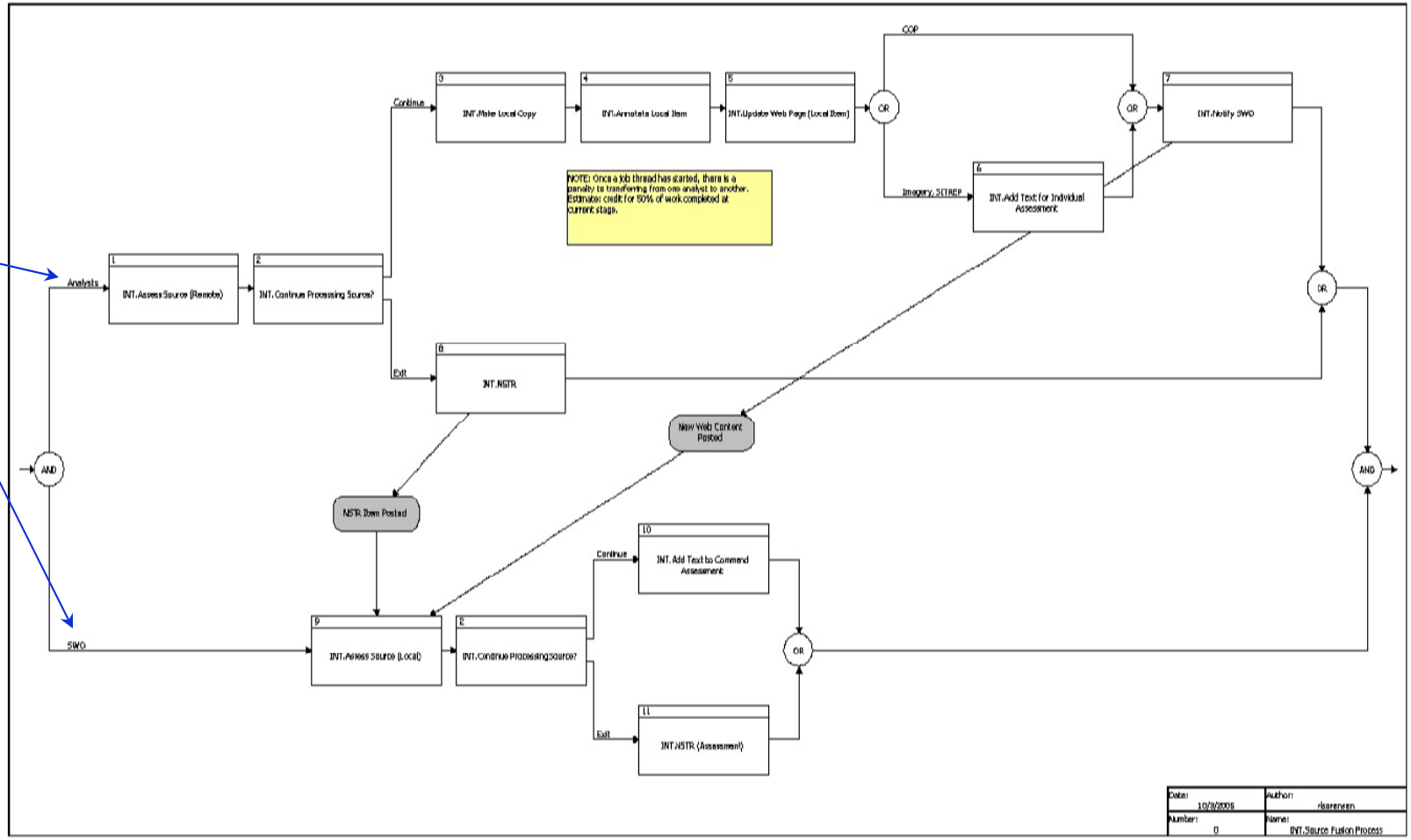
See Allocated Behavior slide

Process to Efficiently Build Behavior & SM Models

1. **Threads** : Work process of each job by an operator
   - Articulates specific job activities for each operator
2. **Integrated** : Common themes between jobs & operators
   - Calculates minimum resource demand
3. **Allocated** : Differentiation of skills
   - Added cost of specialization
   - Determine any offsets when generic work is done during idle time
4. **State Machine** : Schedule and track concurrent activity
   - Impact of Business Rules (i.e. Job priorities & time remaining)
   - Caution: Logic is not as visible to users as usual behavior models

Allocated Behavior

Allocations

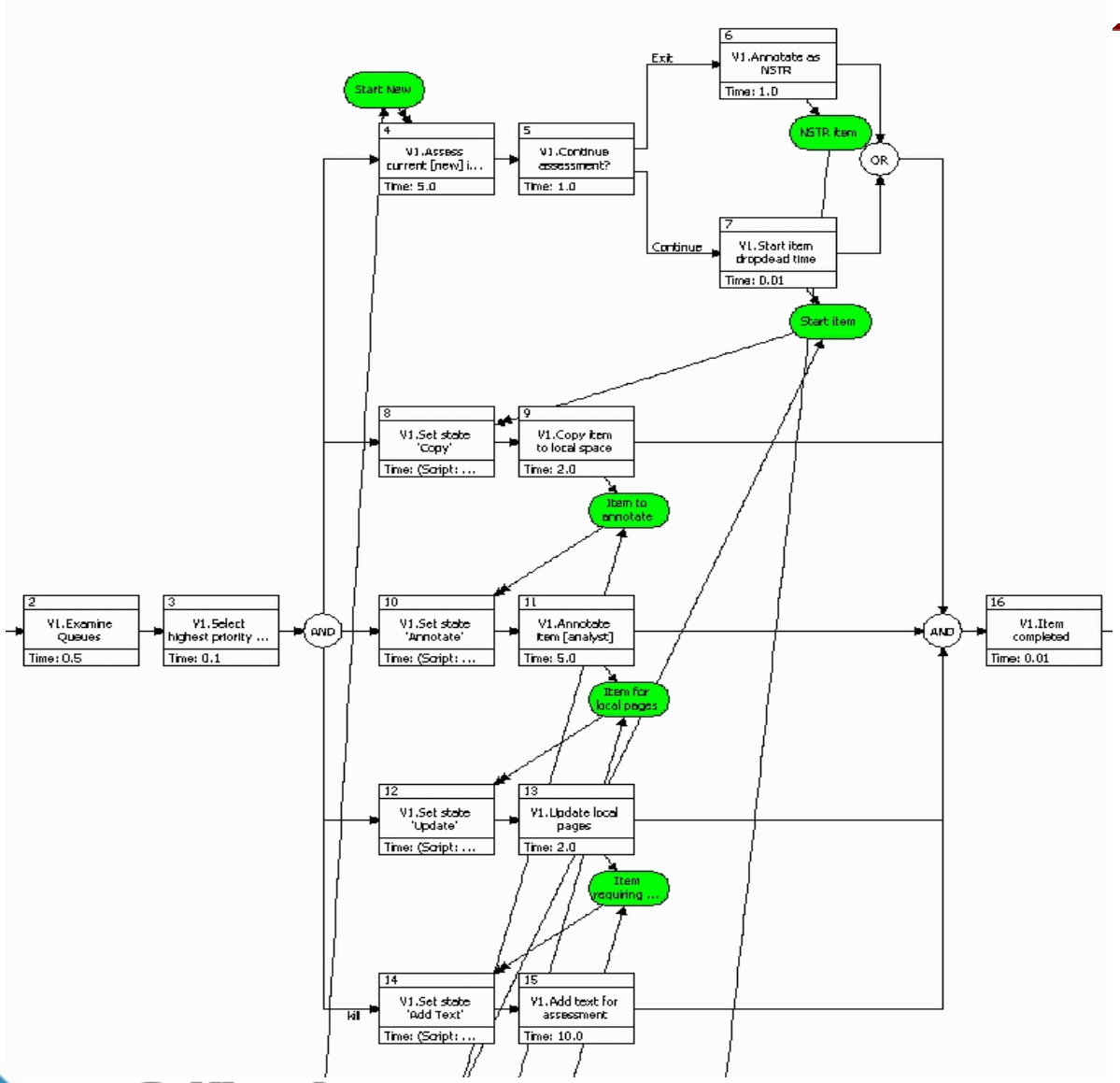


Other Aspects to Explore

- Our target environment has clear notion of interruptible processes. Need to explore:
  - Impacts of interrupts on timelines
  - Business rules for interrupt handling
- Our target environment exhibits queuing of products. Need to explore:
  - Queuing logic

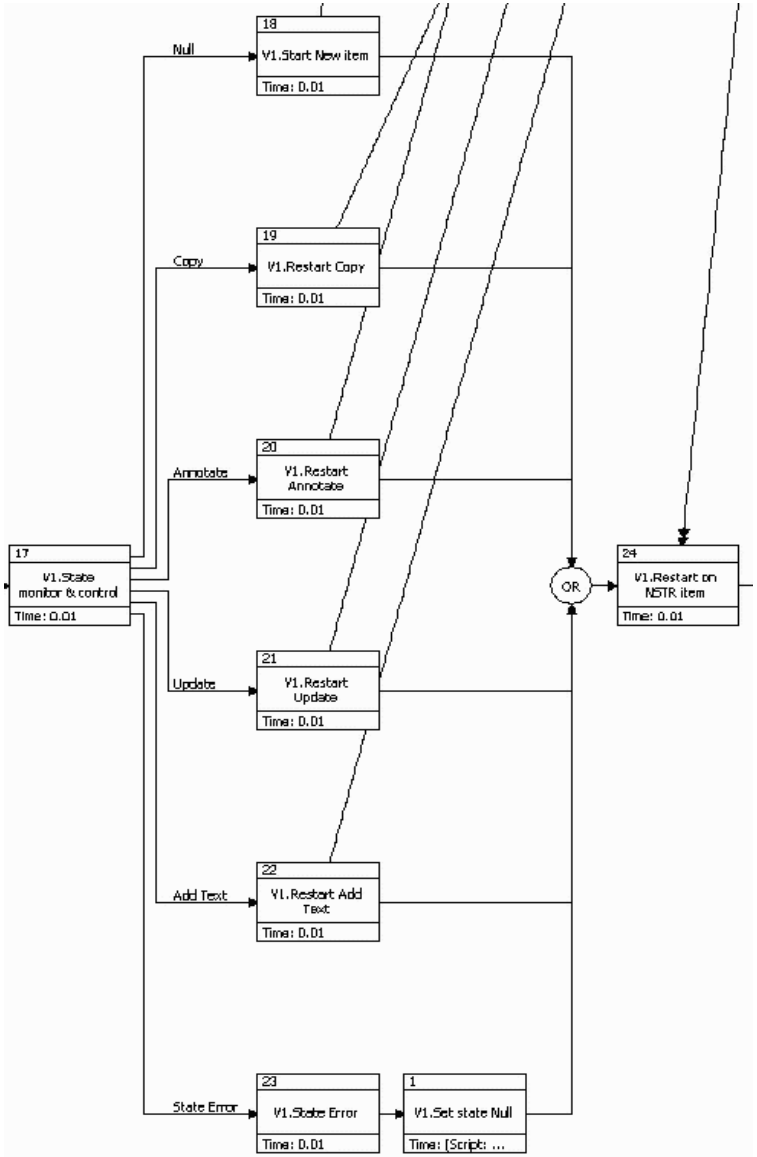
Interrupt Model v1

- Segmenting Tasks for Re-entrant Behavior



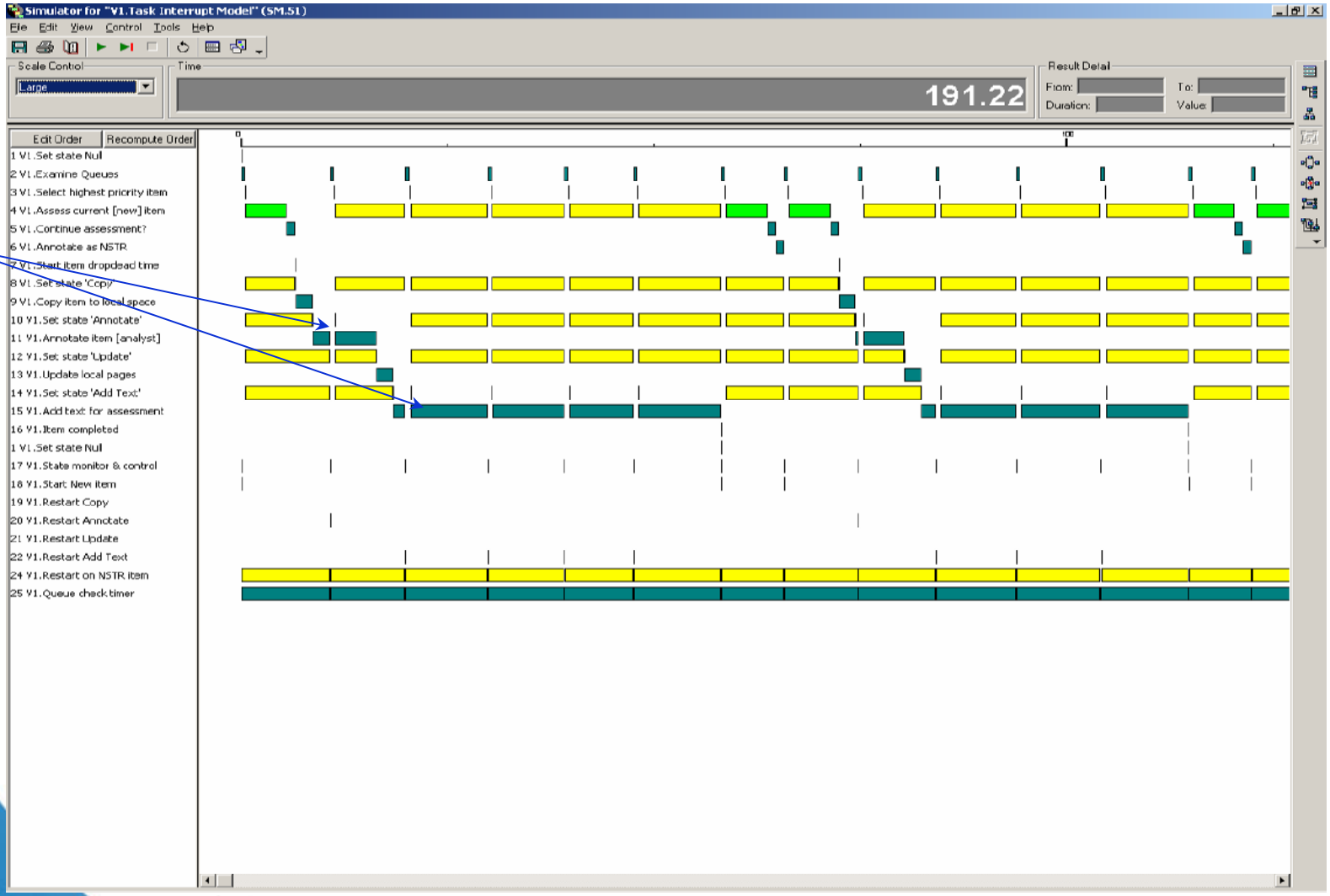
Interrupt Model v1

- Control Logic for Re-entrant Behaviour

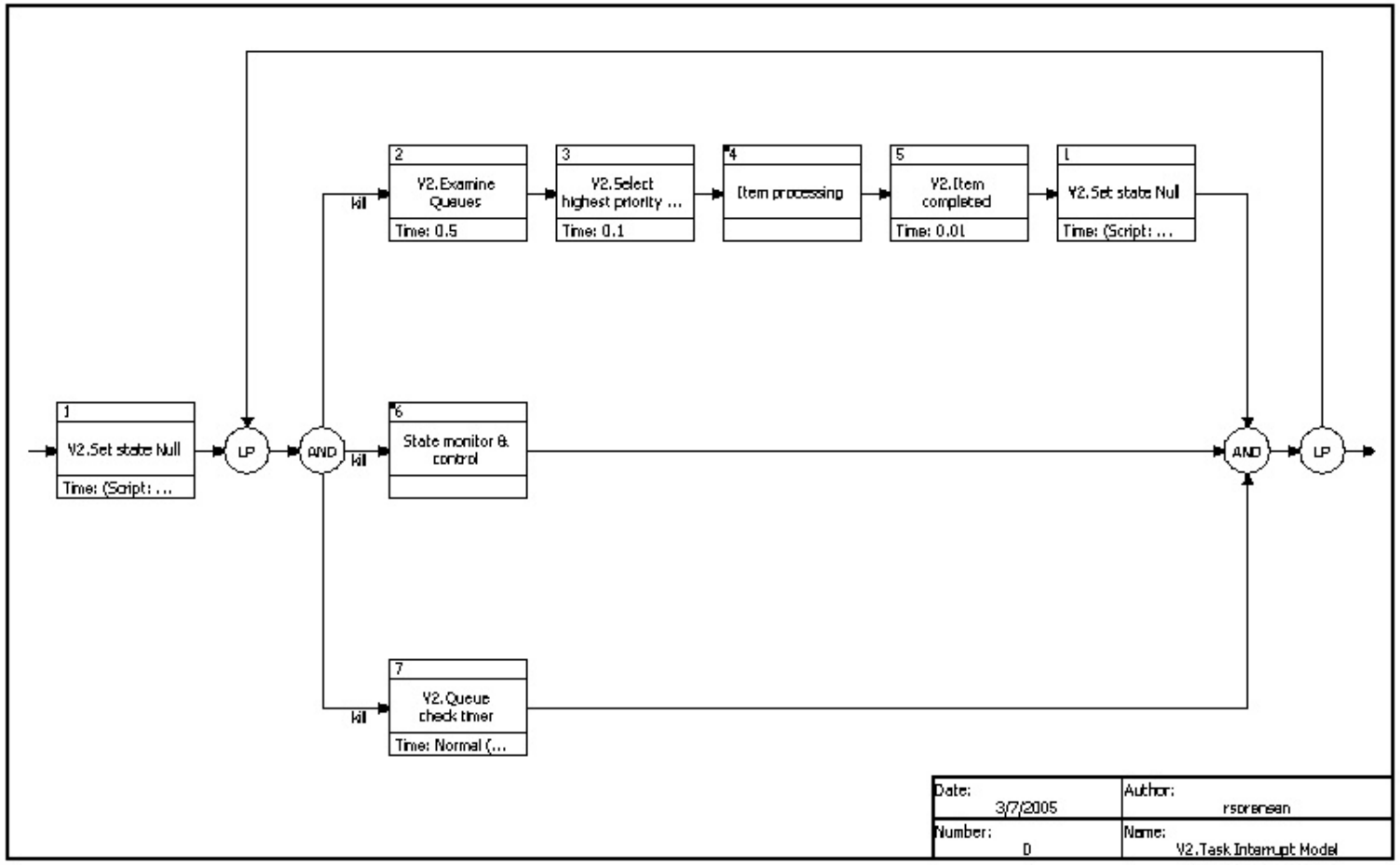


Interrupt Model v1 Execution

Interrupted &
Re-entered tasks

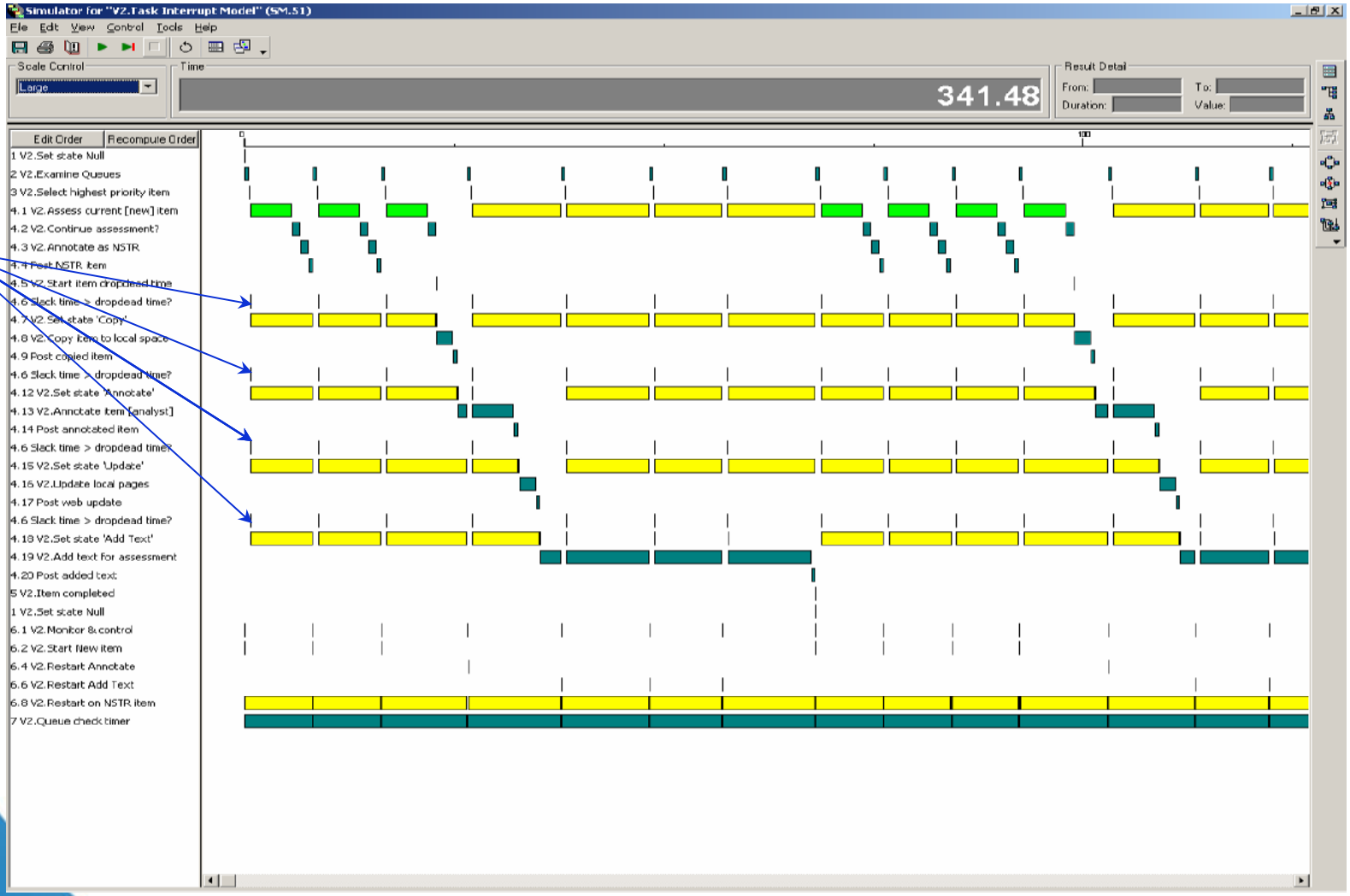


Interrupt Model v2
- Adding Impacts of Queuing

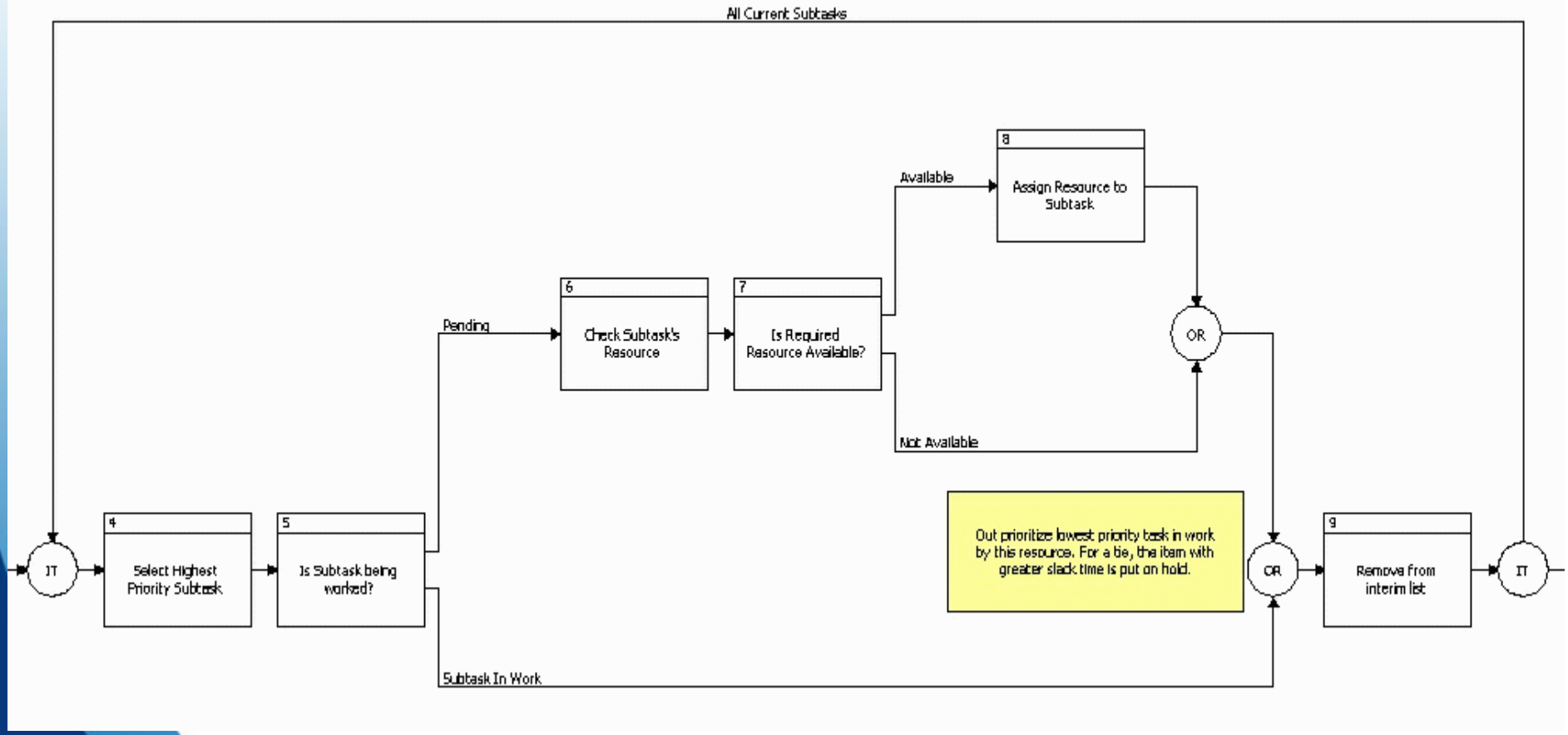


Interrupt Model v2 Execution

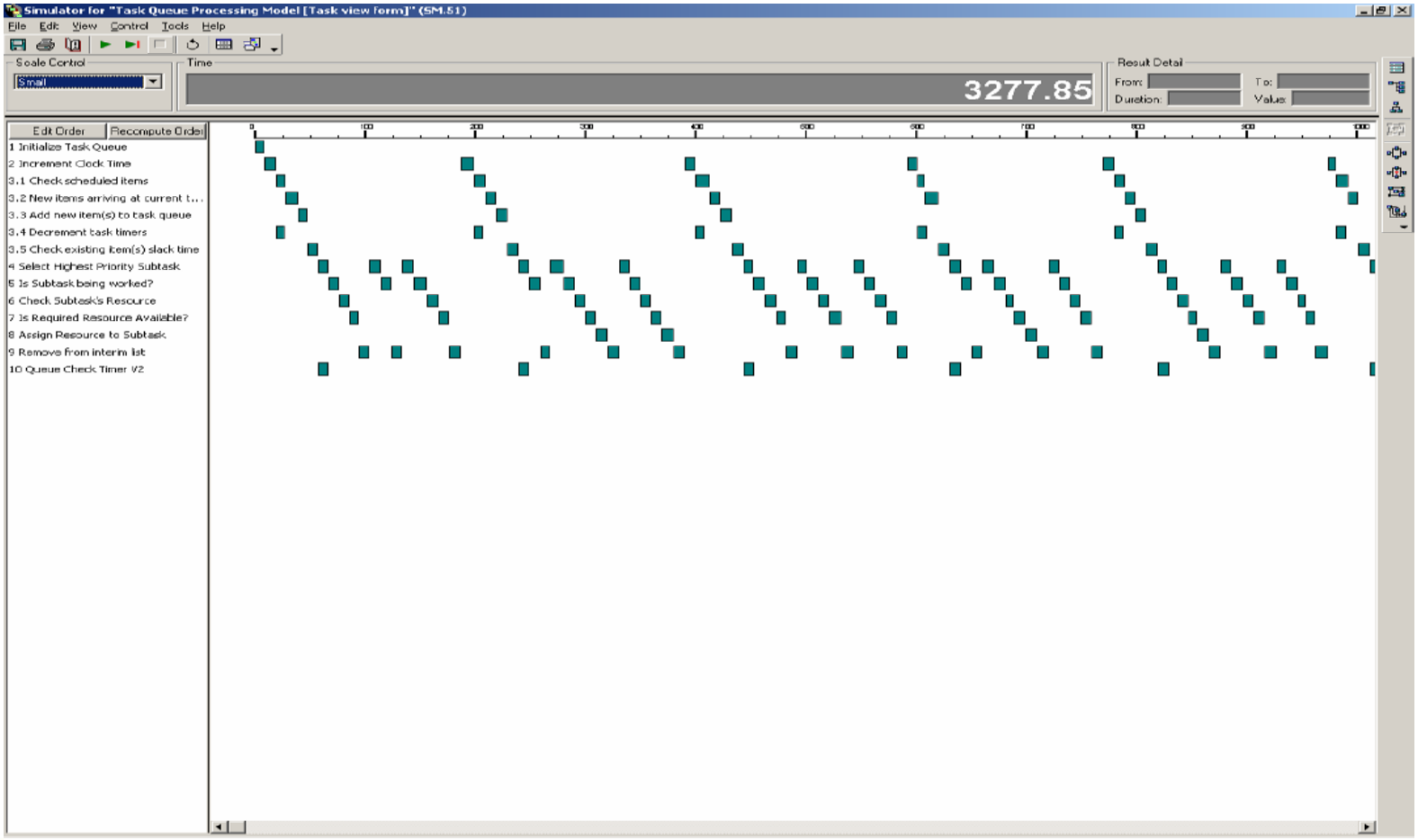
Item processing time checks



Task Queue Processing



Task Queue Processing - Execution



Process to Efficiently Build Behavior & SM Models

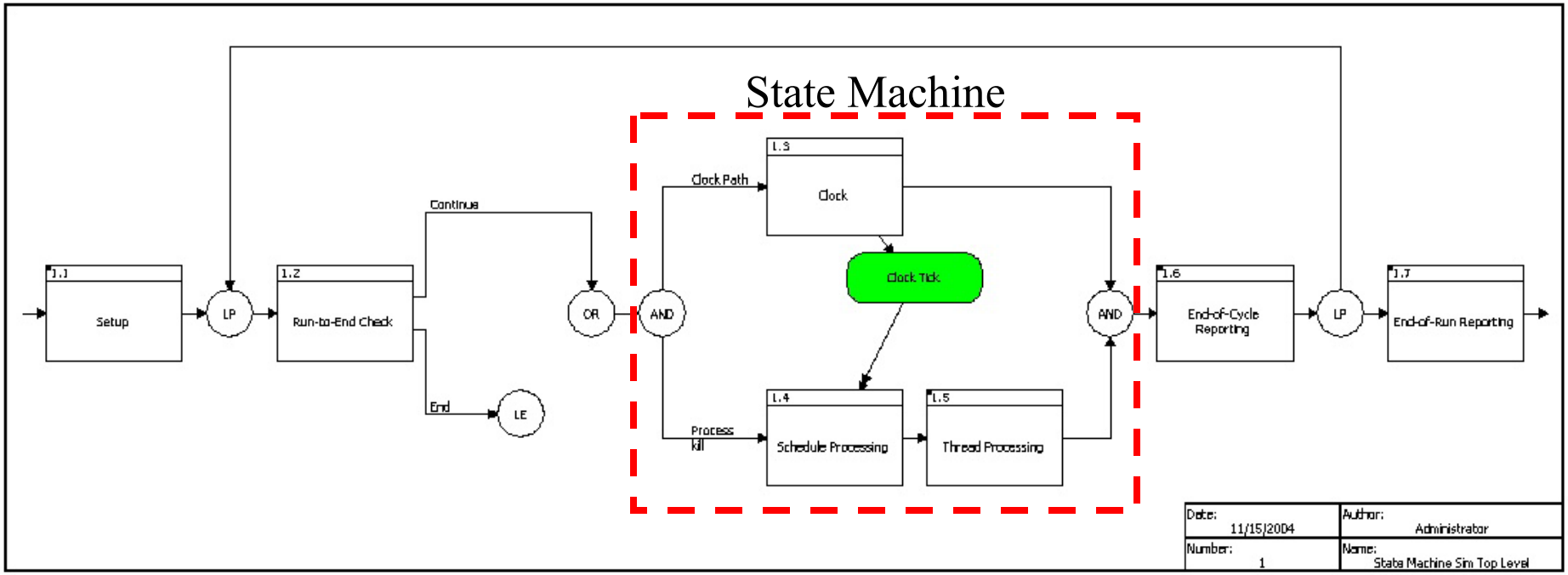
1. **Threads** : Work process of each job by an operator
   - Articulates specific job activities for each operator
2. **Integrated** : Common themes between jobs & operators
   - Calculates minimum resource demand
3. **Allocated** : Differentiation of skills
   - Added cost of specialization
   - Determine any offsets when generic work is done during idle time
4. **State Machine** : Schedule and track concurrent activity
   - Impact of Business Rules (i.e. Job priorities & time remaining)
   - Caution: Logic is not as visible to users as usual behavior models

State Machine Form

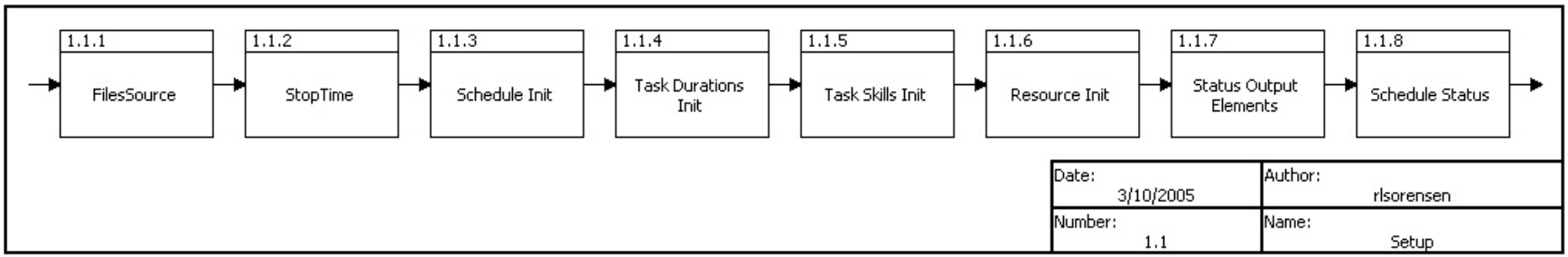
- Orthogonal to FFBD* forms
  - FFBDs are single-threaded, single-instance temporal domains
  - State machine is multi-threaded, multi-instance
- Essentially an 'engine' for processing scenarios
  - Task flow, resource, and queue logic in model
  - Tasks, resources, timelines instantiated at runtime
- Uses pre- and post- processing (Excel)

*FFBD: Functional Flow Block Diagram

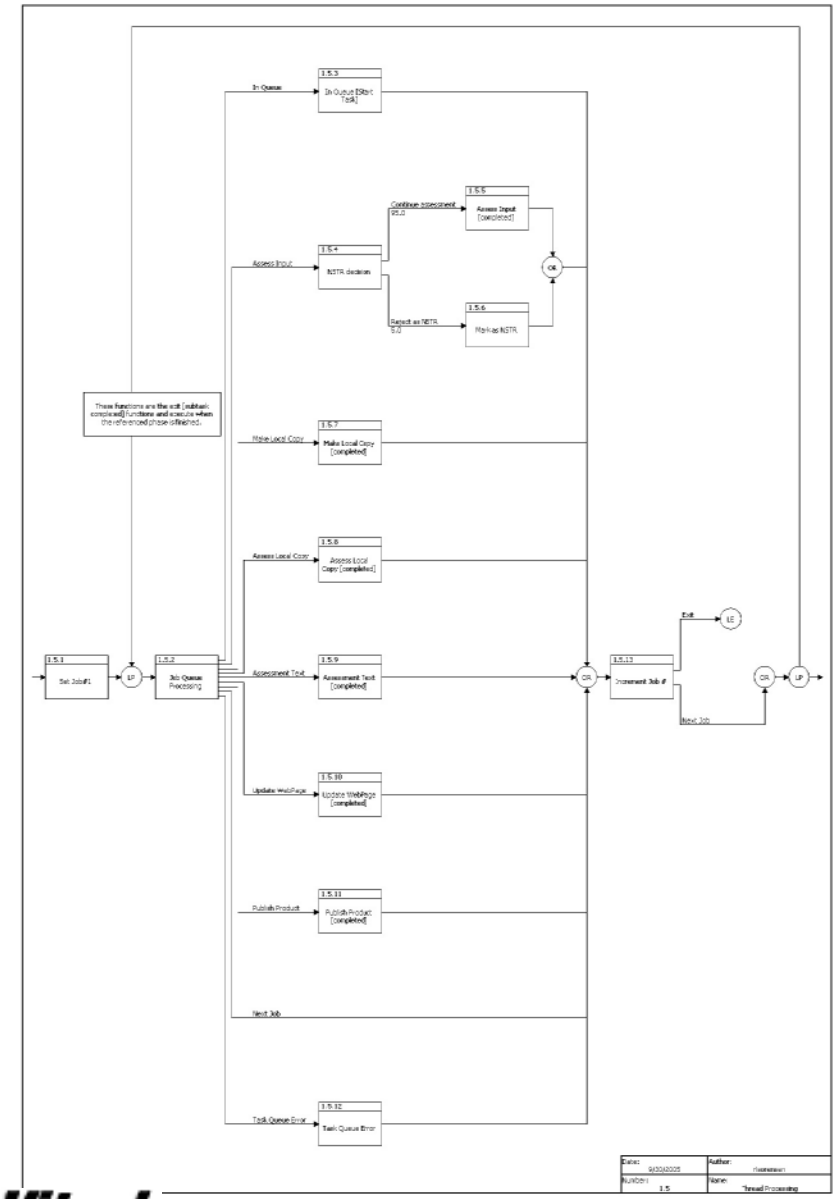
State Machine



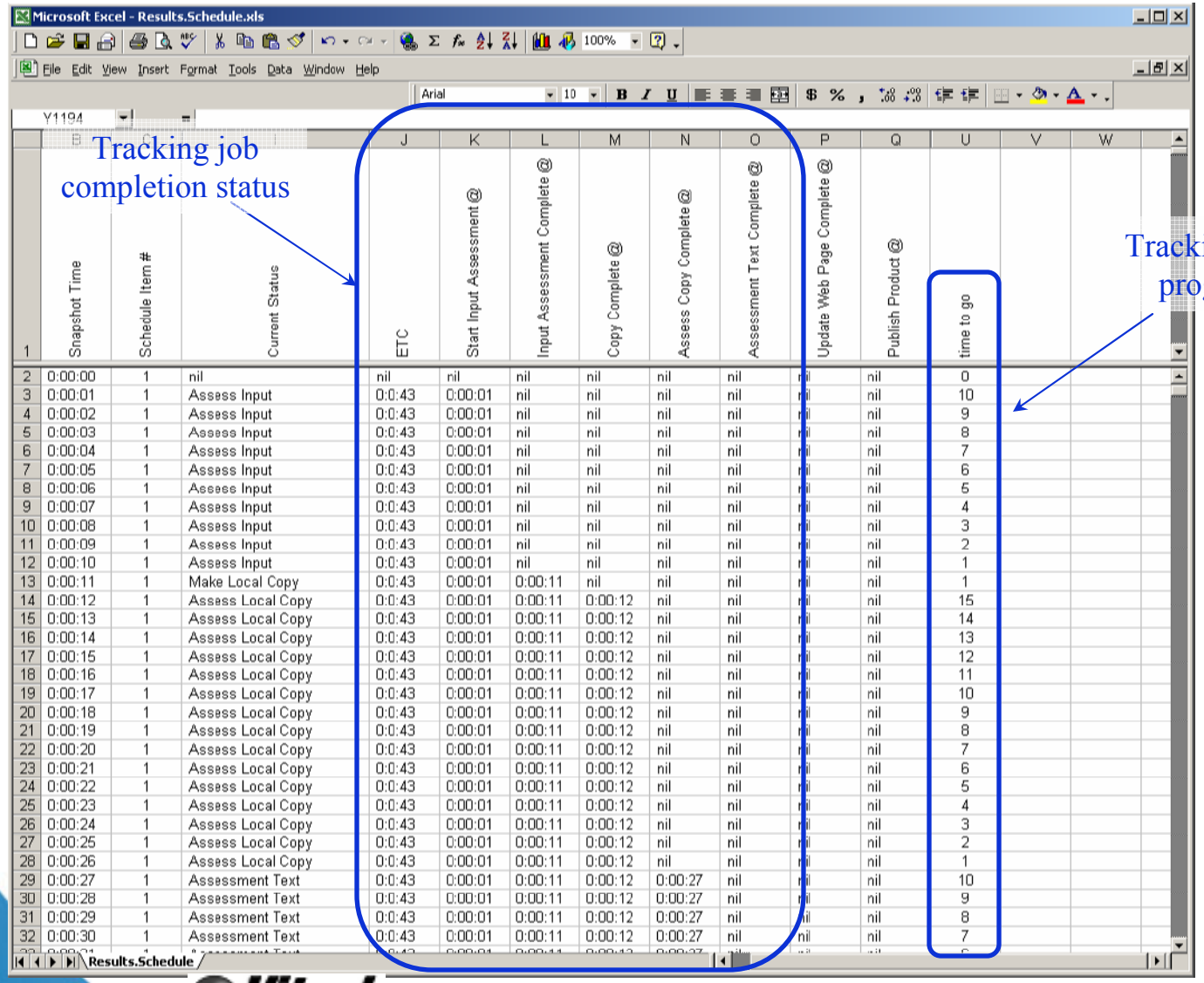
State Machine - Setup



State Machine – Thread Processing



Execution Results – Example

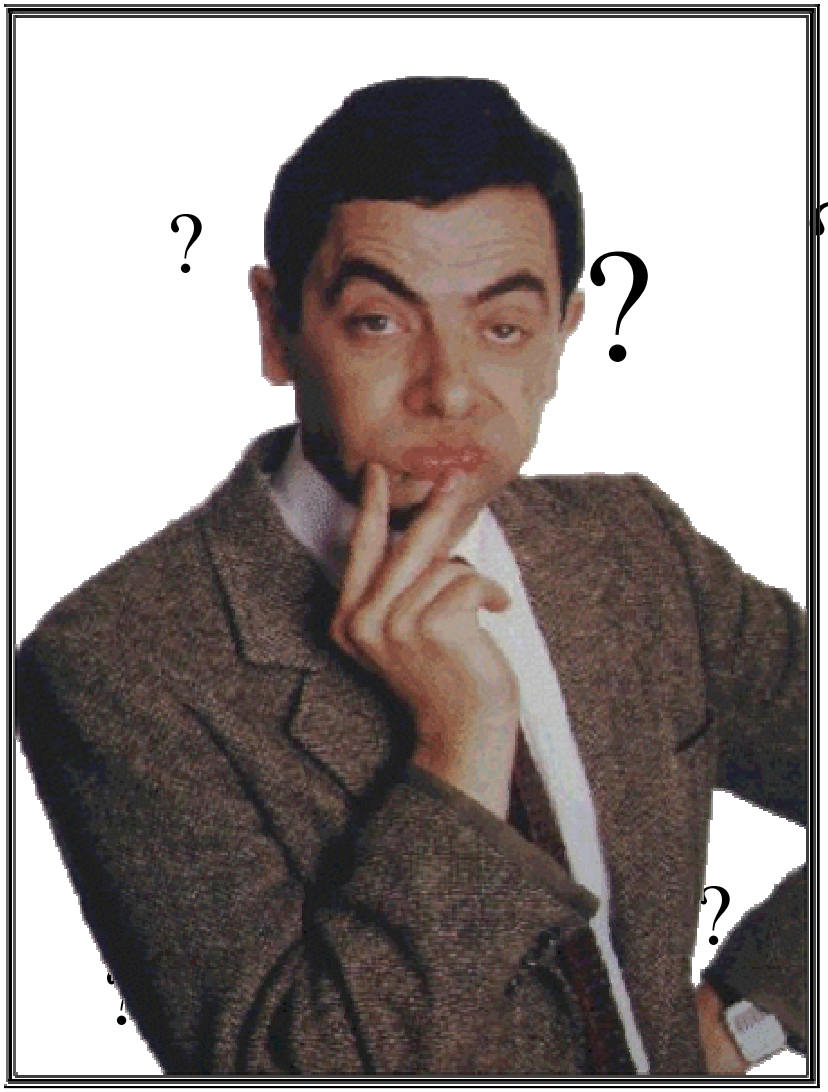


	Snapshot Time	Schedule Item #	Current Status	ETC	Start Input Assessment @	Input Assessment Complete @	Copy Complete @	Assess Copy Complete @	Assessment Text Complete @	Update Web Page Complete @	Publish Product @	time to go
2	0:00:00	1	nil	nil	nil	nil	nil	nil	nil	nil	nil	0
3	0:00:01	1	Assess Input	0:0:43	0:00:01	nil	nil	nil	nil	nil	nil	10
4	0:00:02	1	Assess Input	0:0:43	0:00:01	nil	nil	nil	nil	nil	nil	9
5	0:00:03	1	Assess Input	0:0:43	0:00:01	nil	nil	nil	nil	nil	nil	8
6	0:00:04	1	Assess Input	0:0:43	0:00:01	nil	nil	nil	nil	nil	nil	7
7	0:00:05	1	Assess Input	0:0:43	0:00:01	nil	nil	nil	nil	nil	nil	6
8	0:00:06	1	Assess Input	0:0:43	0:00:01	nil	nil	nil	nil	nil	nil	5
9	0:00:07	1	Assess Input	0:0:43	0:00:01	nil	nil	nil	nil	nil	nil	4
10	0:00:08	1	Assess Input	0:0:43	0:00:01	nil	nil	nil	nil	nil	nil	3
11	0:00:09	1	Assess Input	0:0:43	0:00:01	nil	nil	nil	nil	nil	nil	2
12	0:00:10	1	Assess Input	0:0:43	0:00:01	nil	nil	nil	nil	nil	nil	1
13	0:00:11	1	Make Local Copy	0:0:43	0:00:01	0:00:11	nil	nil	nil	nil	nil	1
14	0:00:12	1	Assess Local Copy	0:0:43	0:00:01	0:00:11	0:00:12	nil	nil	nil	nil	15
15	0:00:13	1	Assess Local Copy	0:0:43	0:00:01	0:00:11	0:00:12	nil	nil	nil	nil	14
16	0:00:14	1	Assess Local Copy	0:0:43	0:00:01	0:00:11	0:00:12	nil	nil	nil	nil	13
17	0:00:15	1	Assess Local Copy	0:0:43	0:00:01	0:00:11	0:00:12	nil	nil	nil	nil	12
18	0:00:16	1	Assess Local Copy	0:0:43	0:00:01	0:00:11	0:00:12	nil	nil	nil	nil	11
19	0:00:17	1	Assess Local Copy	0:0:43	0:00:01	0:00:11	0:00:12	nil	nil	nil	nil	10
20	0:00:18	1	Assess Local Copy	0:0:43	0:00:01	0:00:11	0:00:12	nil	nil	nil	nil	9
21	0:00:19	1	Assess Local Copy	0:0:43	0:00:01	0:00:11	0:00:12	nil	nil	nil	nil	8
22	0:00:20	1	Assess Local Copy	0:0:43	0:00:01	0:00:11	0:00:12	nil	nil	nil	nil	7
23	0:00:21	1	Assess Local Copy	0:0:43	0:00:01	0:00:11	0:00:12	nil	nil	nil	nil	6
24	0:00:22	1	Assess Local Copy	0:0:43	0:00:01	0:00:11	0:00:12	nil	nil	nil	nil	5
25	0:00:23	1	Assess Local Copy	0:0:43	0:00:01	0:00:11	0:00:12	nil	nil	nil	nil	4
26	0:00:24	1	Assess Local Copy	0:0:43	0:00:01	0:00:11	0:00:12	nil	nil	nil	nil	3
27	0:00:25	1	Assess Local Copy	0:0:43	0:00:01	0:00:11	0:00:12	nil	nil	nil	nil	2
28	0:00:26	1	Assess Local Copy	0:0:43	0:00:01	0:00:11	0:00:12	nil	nil	nil	nil	1
29	0:00:27	1	Assessment Text	0:0:43	0:00:01	0:00:11	0:00:12	0:00:27	nil	nil	nil	10
30	0:00:28	1	Assessment Text	0:0:43	0:00:01	0:00:11	0:00:12	0:00:27	nil	nil	nil	9
31	0:00:29	1	Assessment Text	0:0:43	0:00:01	0:00:11	0:00:12	0:00:27	nil	nil	nil	8
32	0:00:30	1	Assessment Text	0:0:43	0:00:01	0:00:11	0:00:12	0:00:27	nil	nil	nil	7

Recap

- Operational Activity Modeling
  - Task, Process, Exploit, Disseminate (TPED)
  - Task, Post, Process, Use (TPPU)
- State Machine Concepts
- Description of TPPU State Machine
- Modeling Process
- Examples
- Way Ahead

Questions

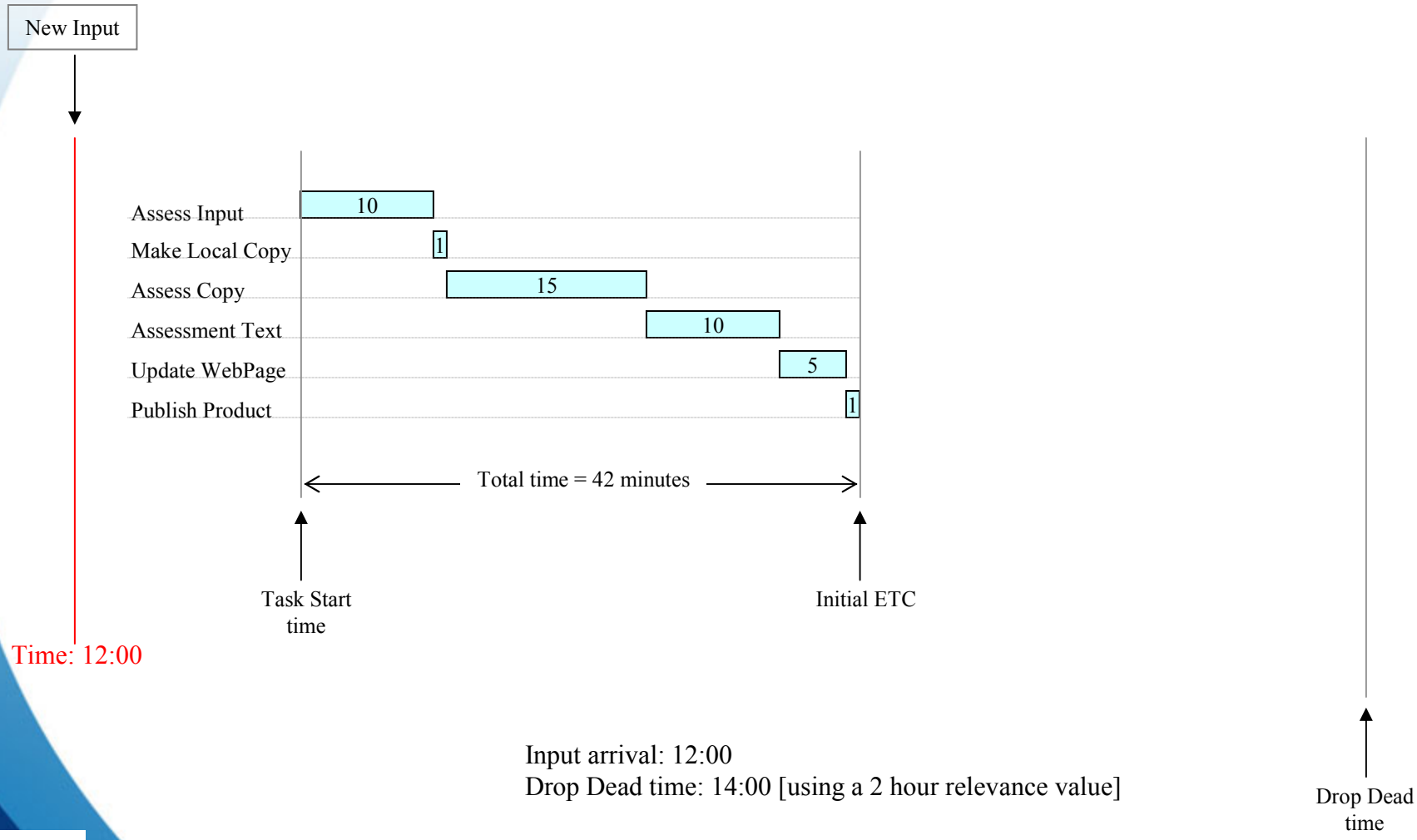


Execution Scenario 1

Normal Completion

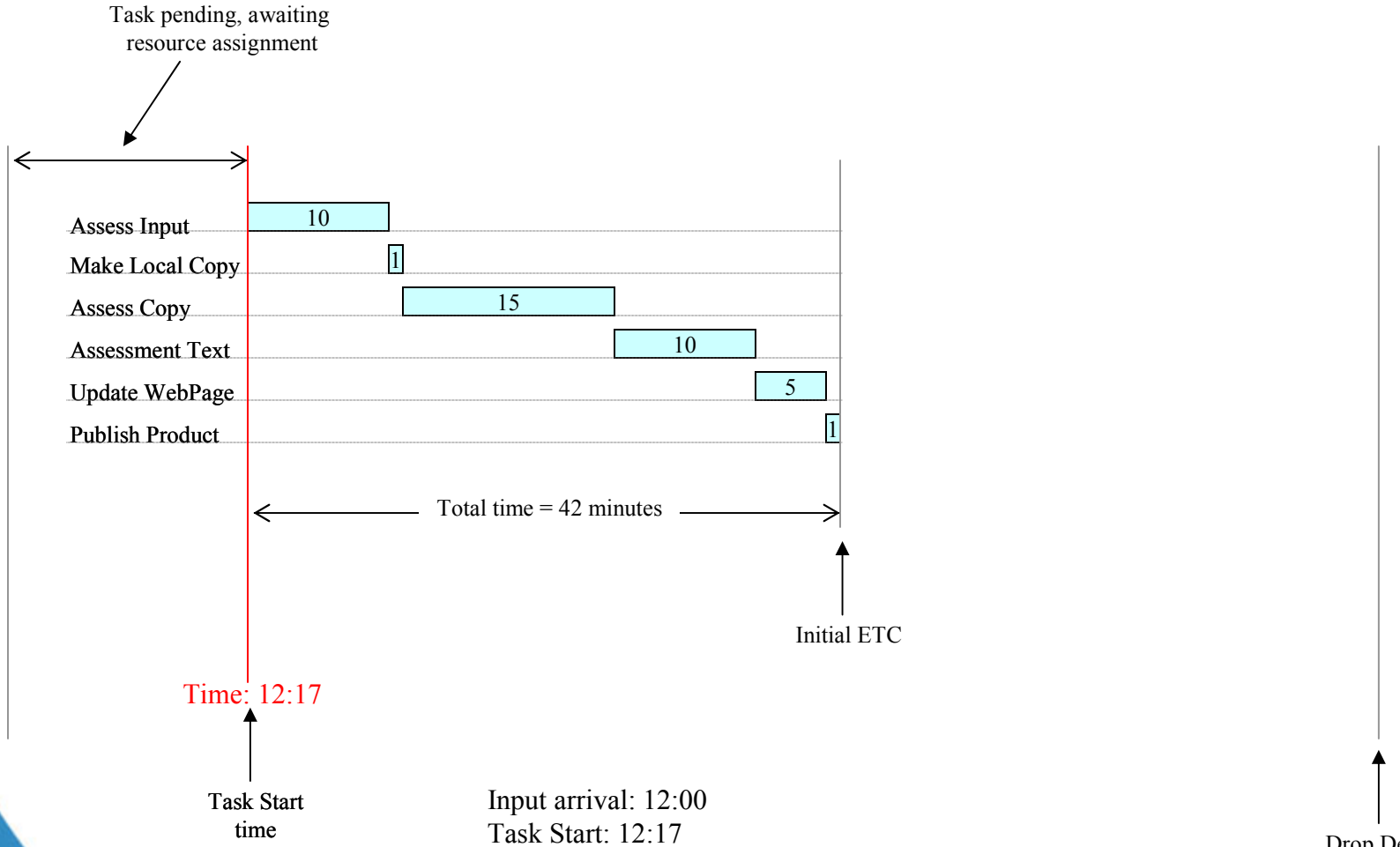
Defence Research and Development Canada
Recherche et développement pour la défense Canada
Canada

Scenario 1-1



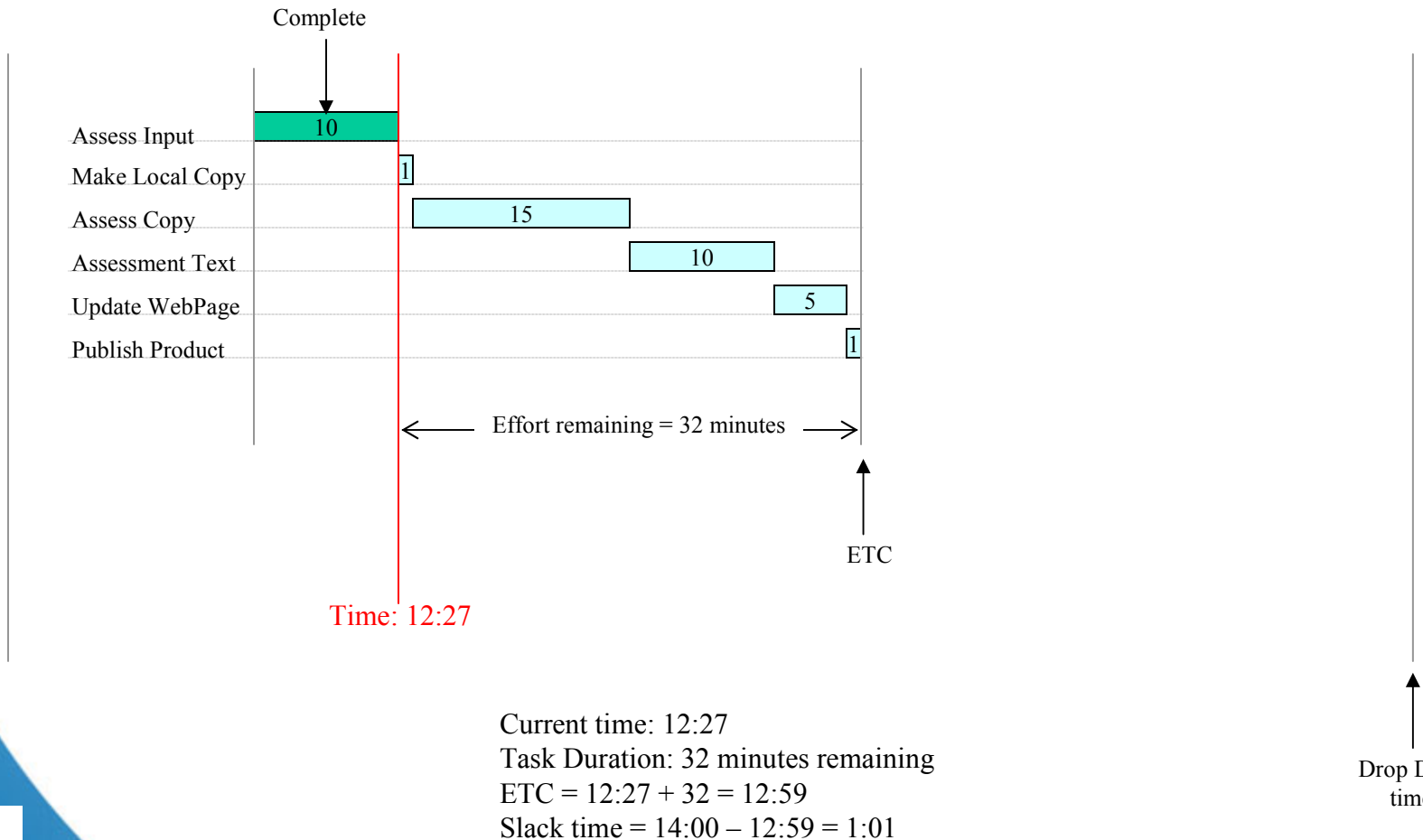
Input arrival: 12:00
Drop Dead time: 14:00 [using a 2 hour relevance value]

Scenario 1-2



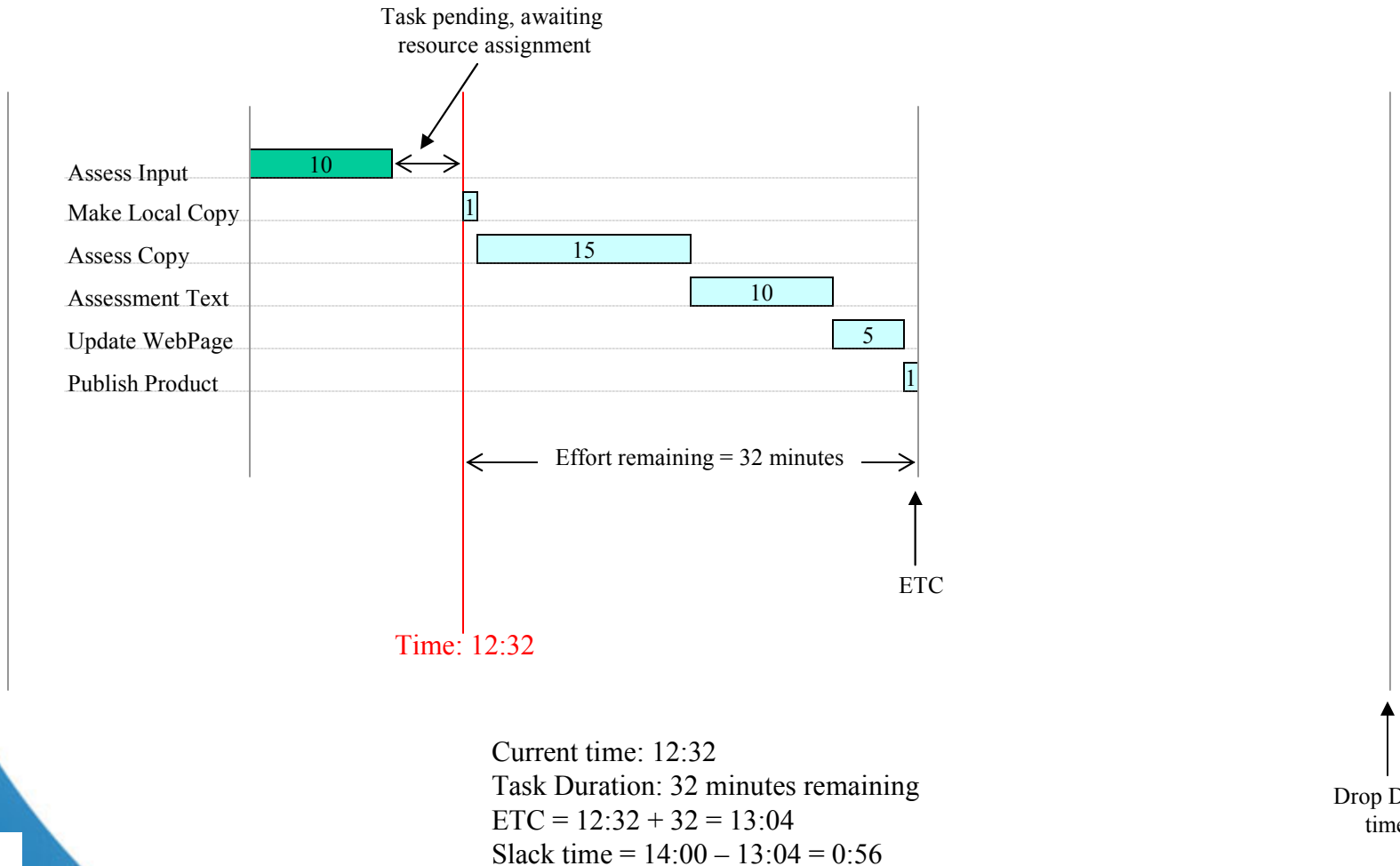
Input arrival: 12:00
Task Start: 12:17
Task Duration: 42 minutes remaining
ETC = 12:17 + 42 = 12:59
Slack time = 14:00 – 12:59 = 1:01

Scenario 1-3



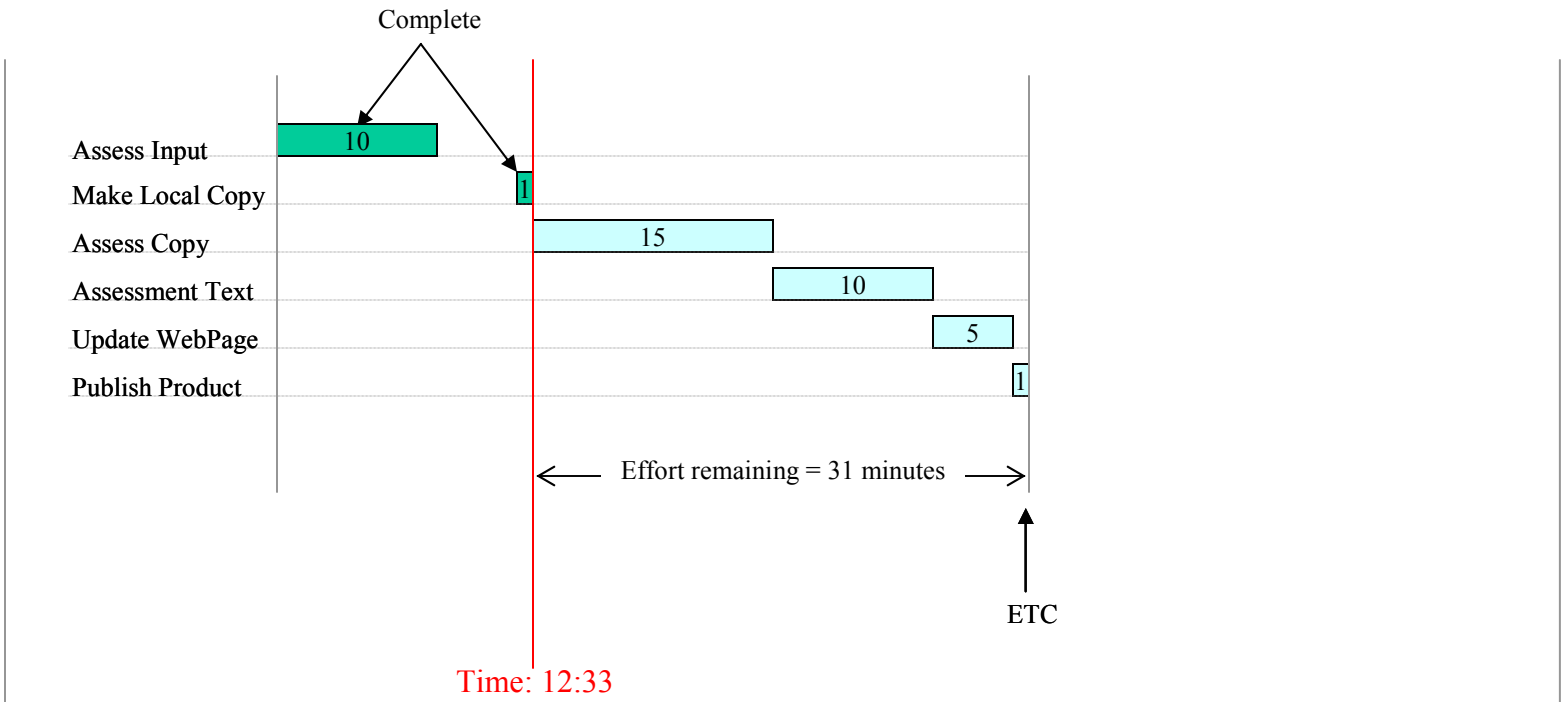
Current time: 12:27
Task Duration: 32 minutes remaining
ETC = 12:27 + 32 = 12:59
Slack time = 14:00 – 12:59 = 1:01

Scenario 1-4



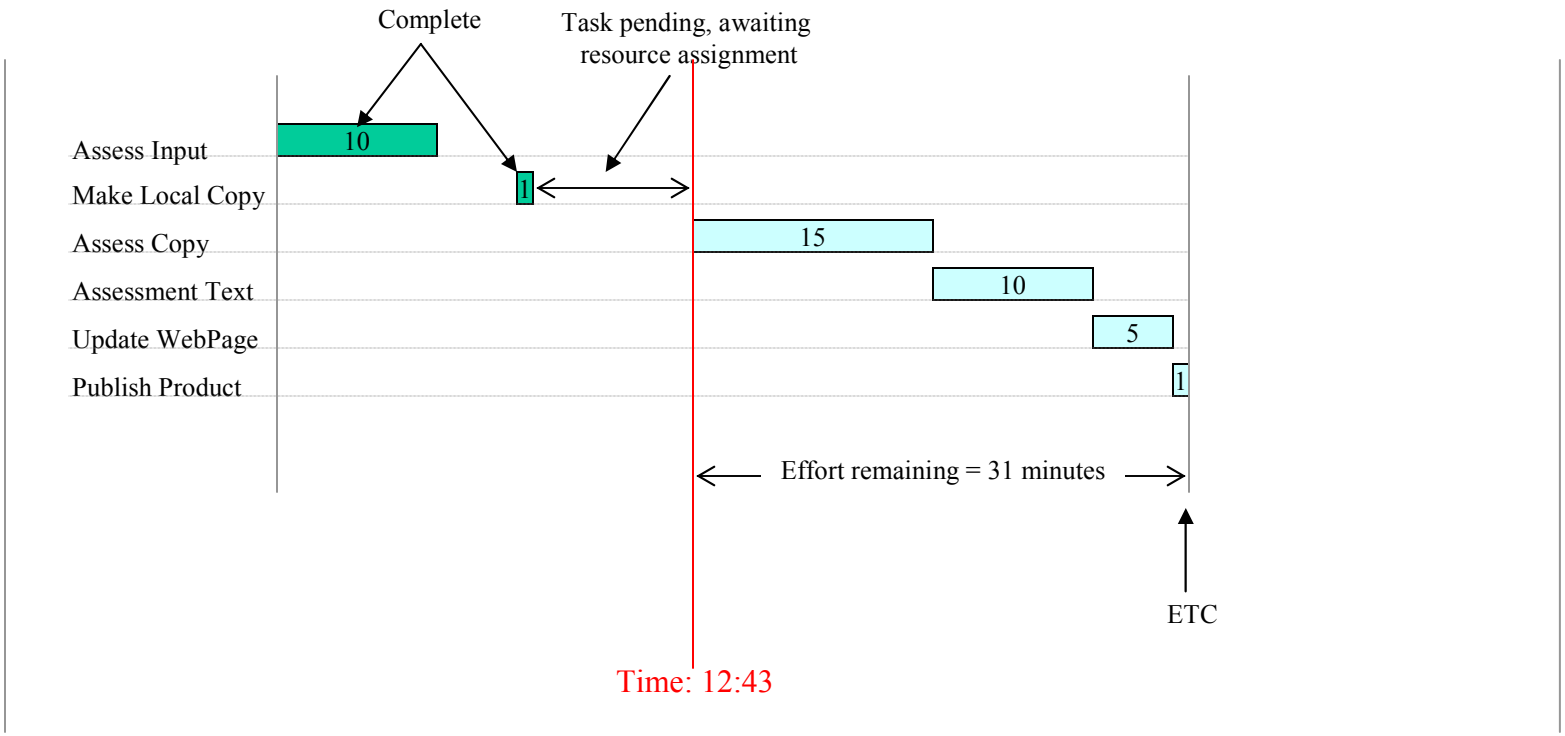
Current time: 12:32
Task Duration: 32 minutes remaining
ETC = 12:32 + 32 = 13:04
Slack time = 14:00 – 13:04 = 0:56

Scenario 1-5



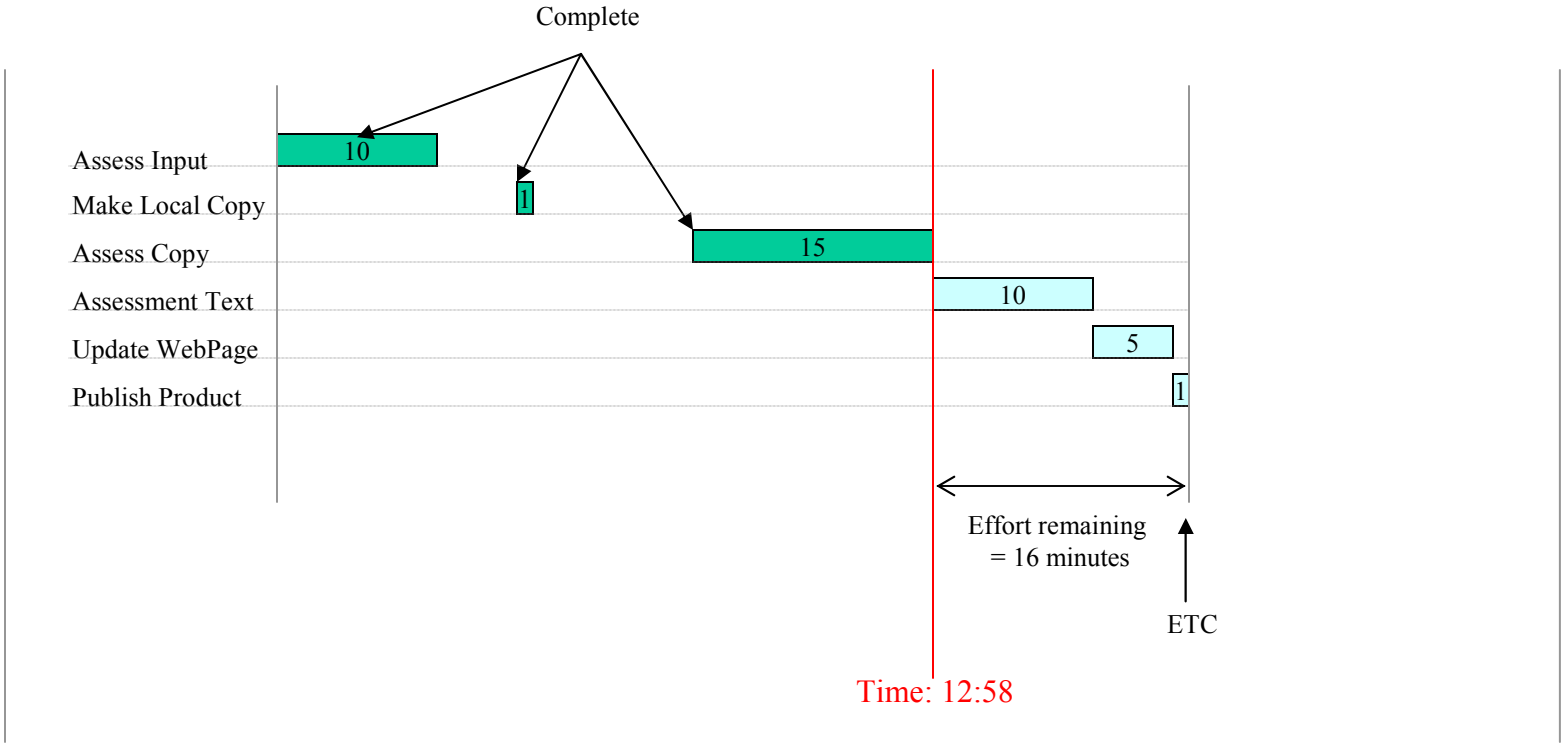
Current time: 12:33
Task Duration: 31 minutes remaining
ETC = 12:33 + 31 = 13:04
Slack time = 14:00 – 13:04 = 0:56

Scenario 1-6



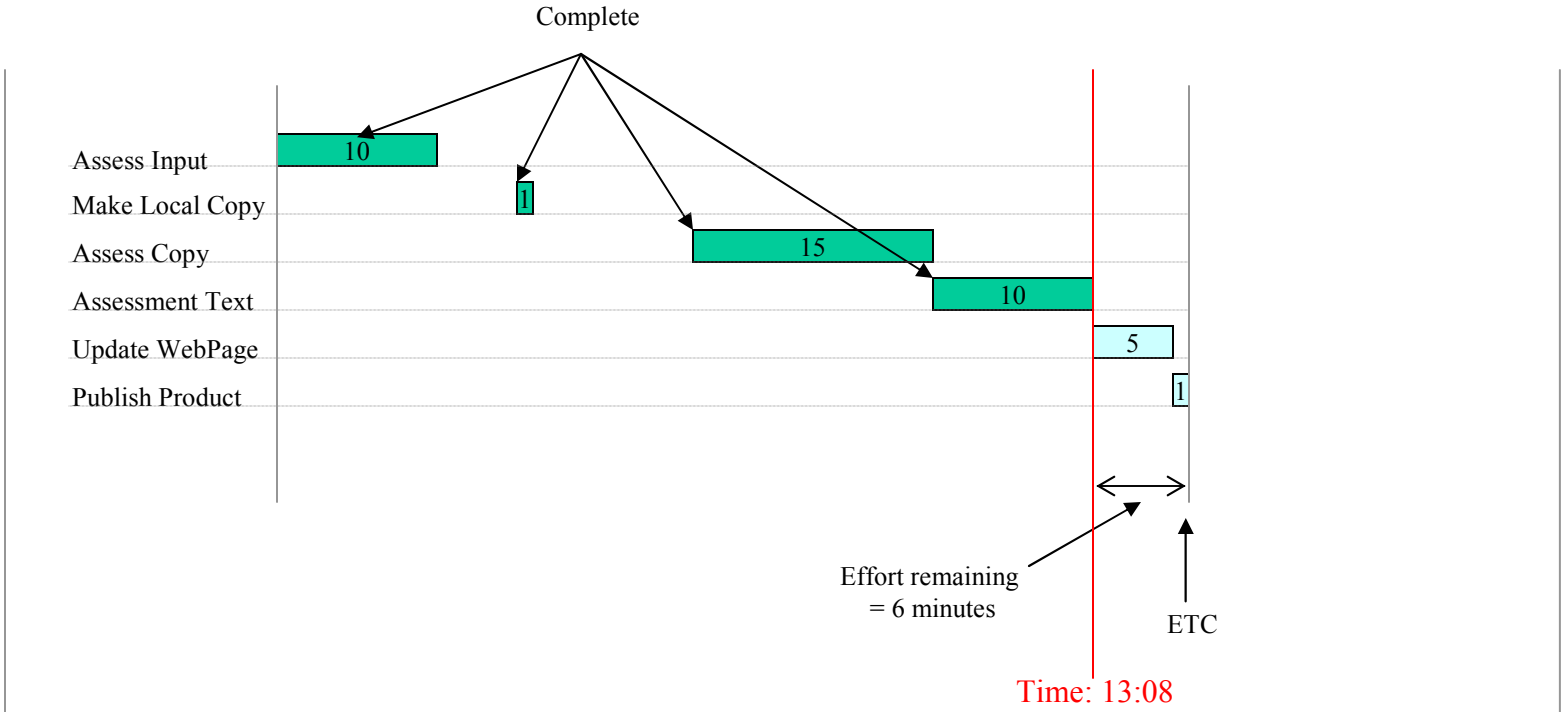
Current time: 12:43
Task Duration: 31 minutes remaining
ETC = 12:43 + 31 = 13:14
Slack time = 14:00 – 13:14 = 0:46

Scenario 1-7



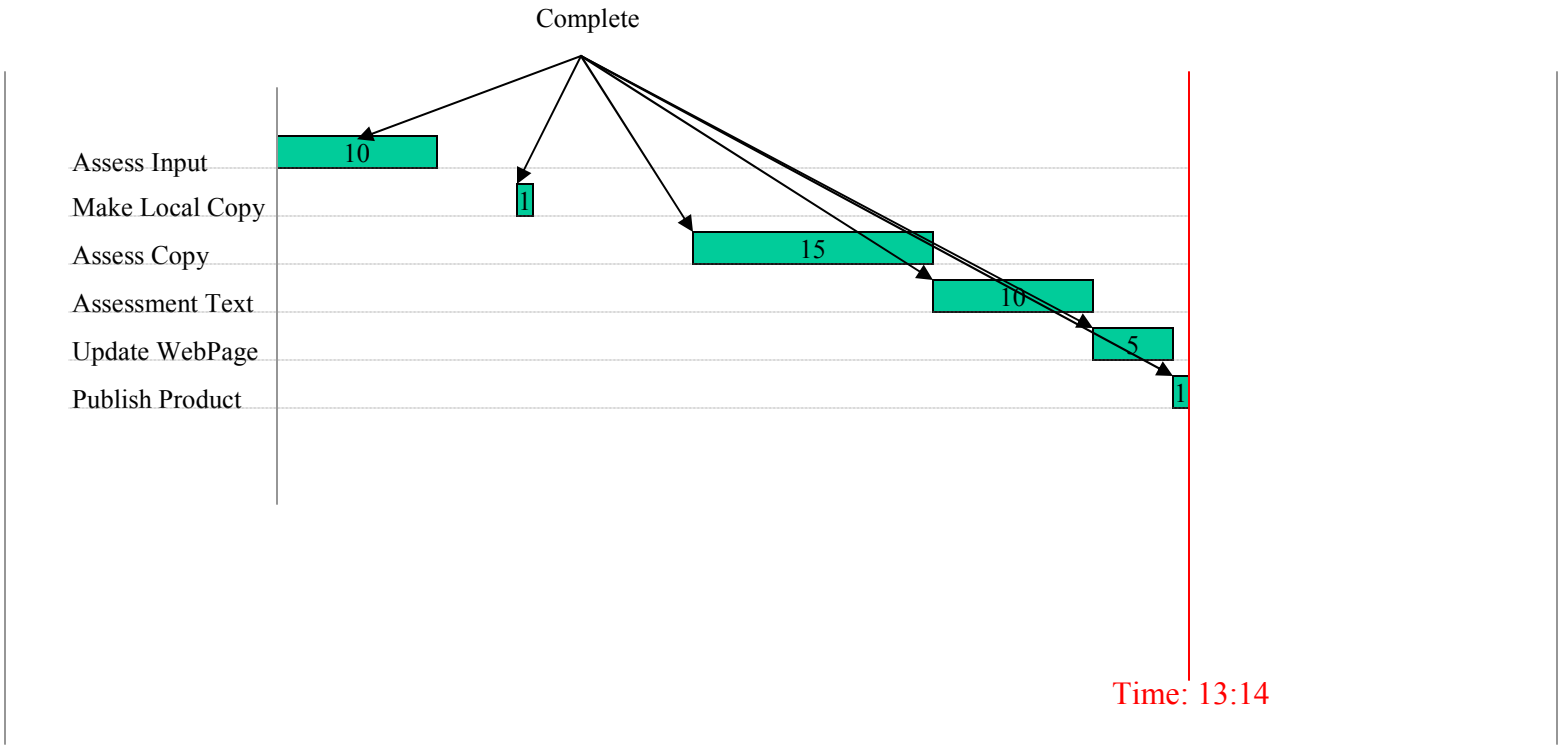
Current time: 12:58
Task Duration: 16 minutes remaining
ETC = 12:58 + 16 = 13:14
Slack time = 14:00 – 13:14 = 0:46

Scenario 1-8



Current time: 13:08
Task Duration: 6 minutes remaining
ETC = 13:08 + 6 = 13:14
Slack time = 14:00 – 13:14 = 0:46

Scenario 1-9



Current time: 13:14
Task Completed
Slack time = 14:00 – 13:14 = 0:46 remaining

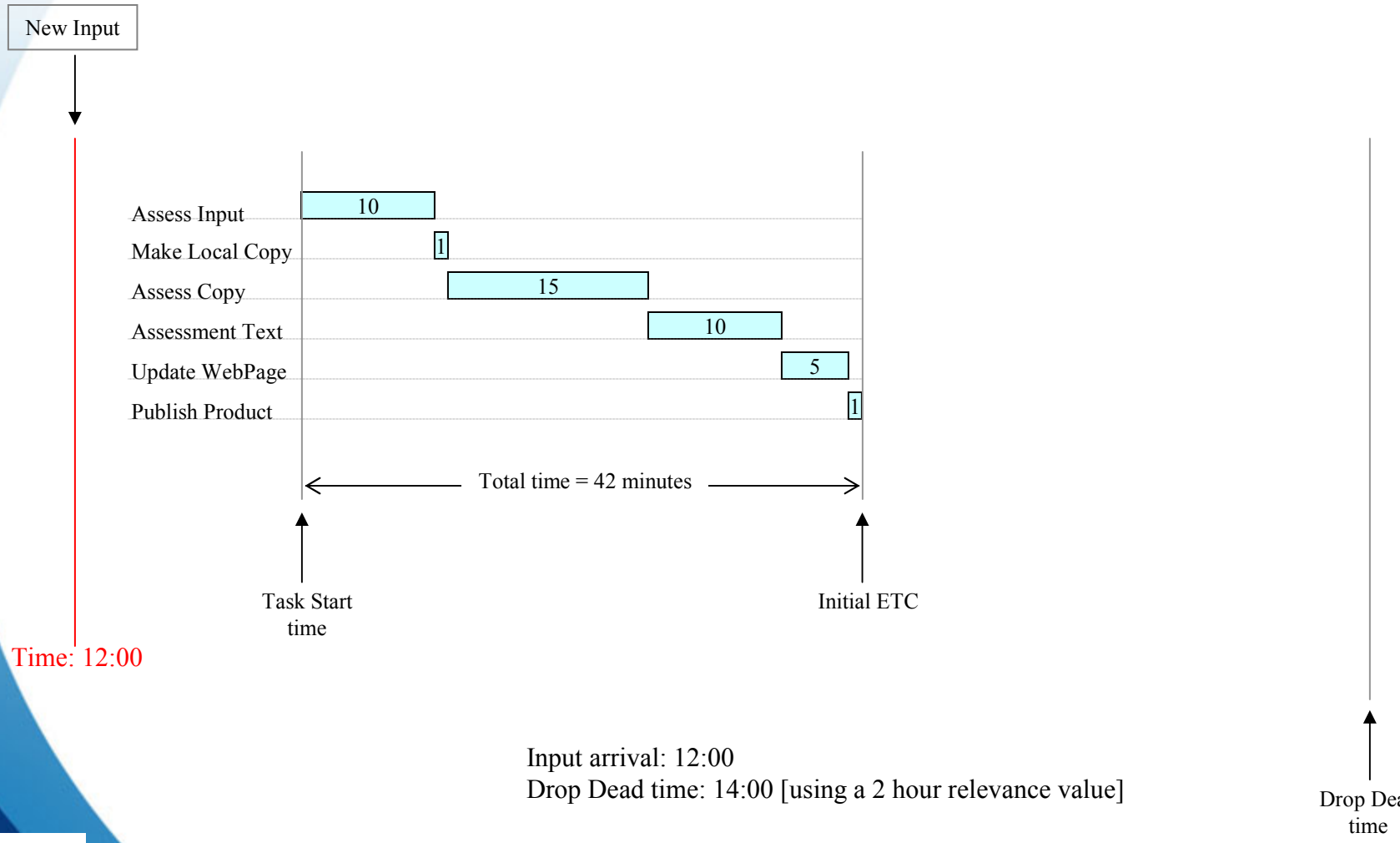
Drop Dead time

Execution Scenario 2

Job Abandoned

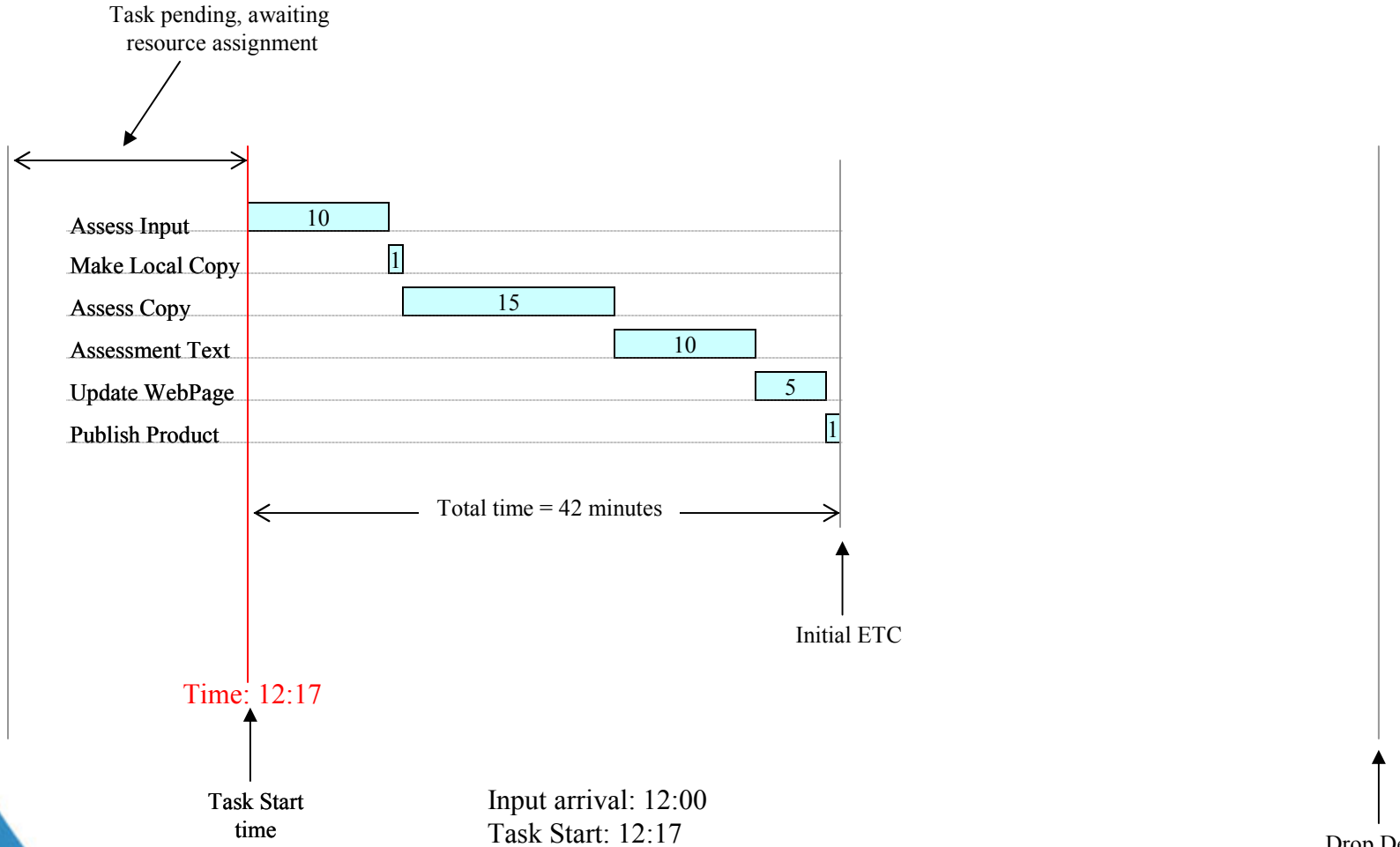
Defence Research and Development Canada
Recherche et développement pour la défense Canada
Canada

Scenario 2-1



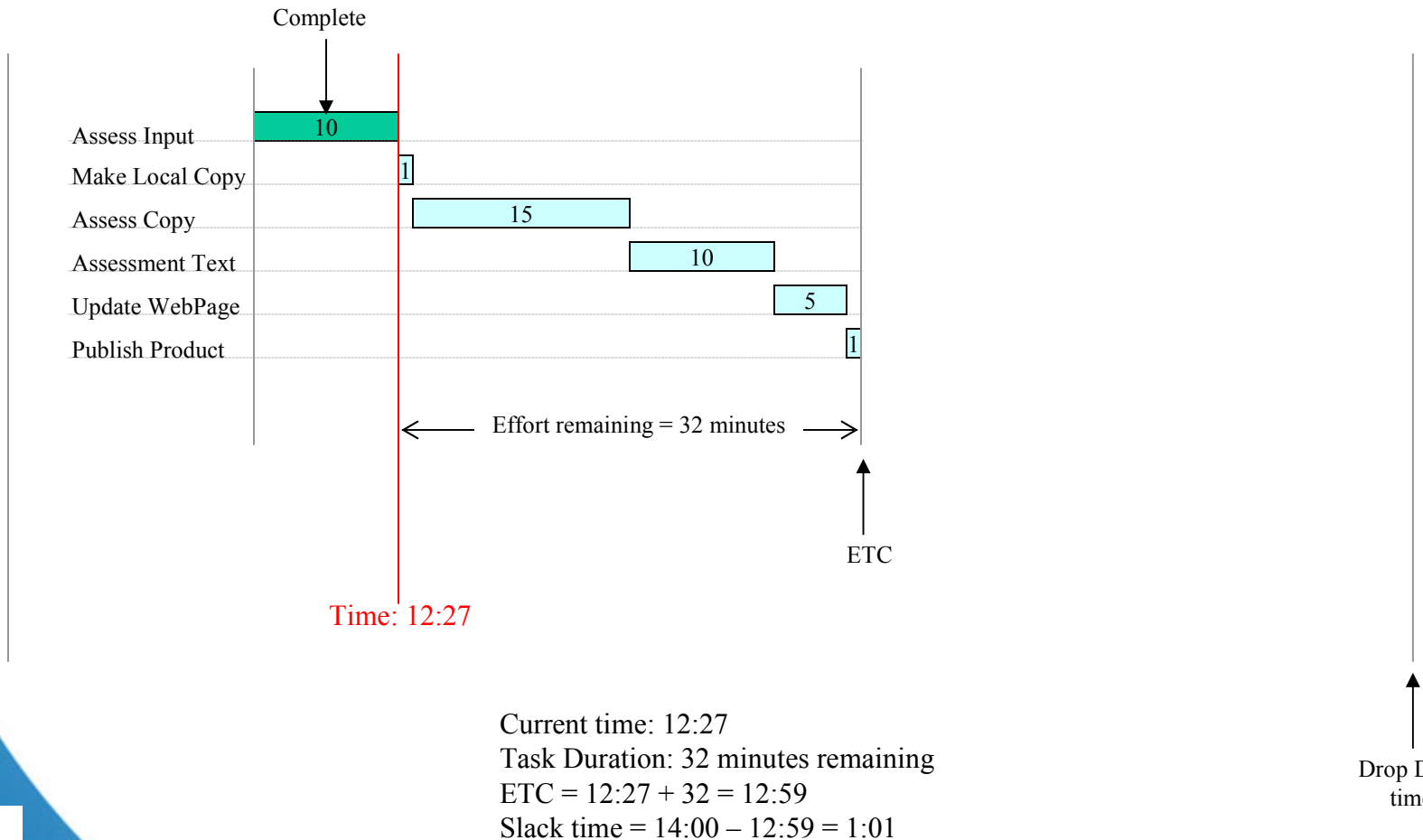
Input arrival: 12:00
Drop Dead time: 14:00 [using a 2 hour relevance value]

Scenario 2-2



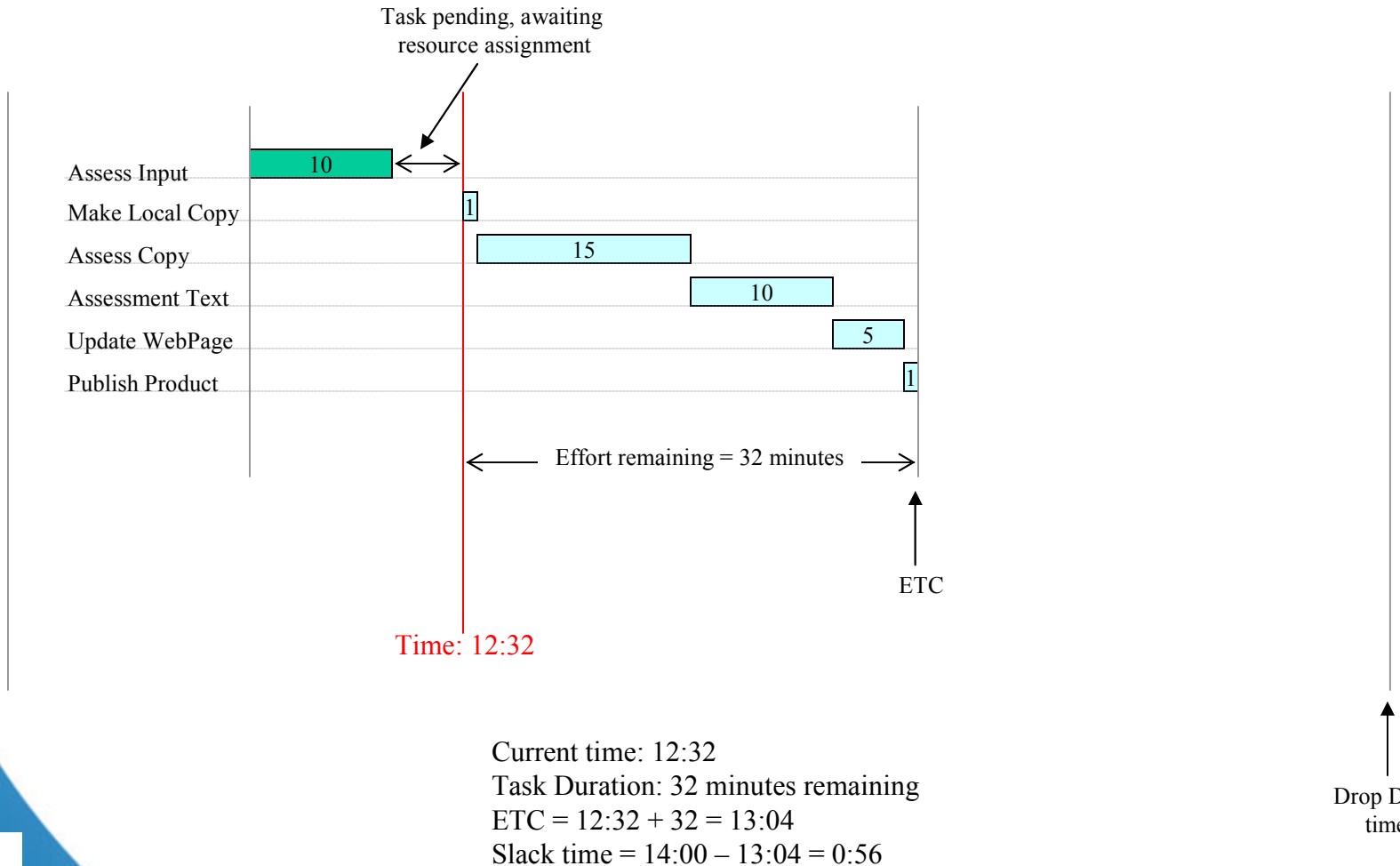
Input arrival: 12:00
Task Start: 12:17
Task Duration: 42 minutes remaining
ETC = 12:17 + 42 = 12:59
Slack time = 14:00 – 12:59 = 1:01

Scenario 2-3



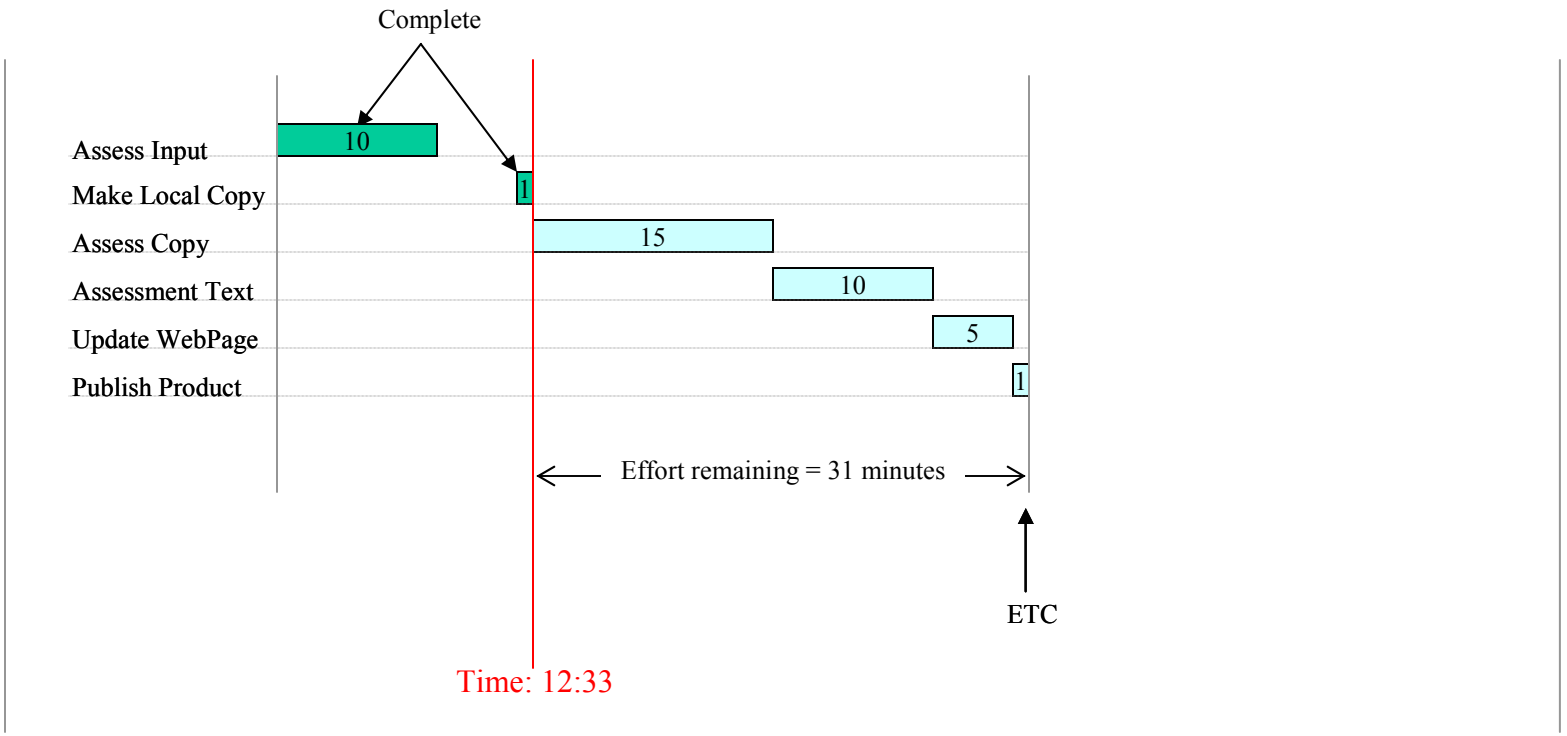
Current time: 12:27
Task Duration: 32 minutes remaining
ETC = 12:27 + 32 = 12:59
Slack time = 14:00 – 12:59 = 1:01

Scenario 2-4



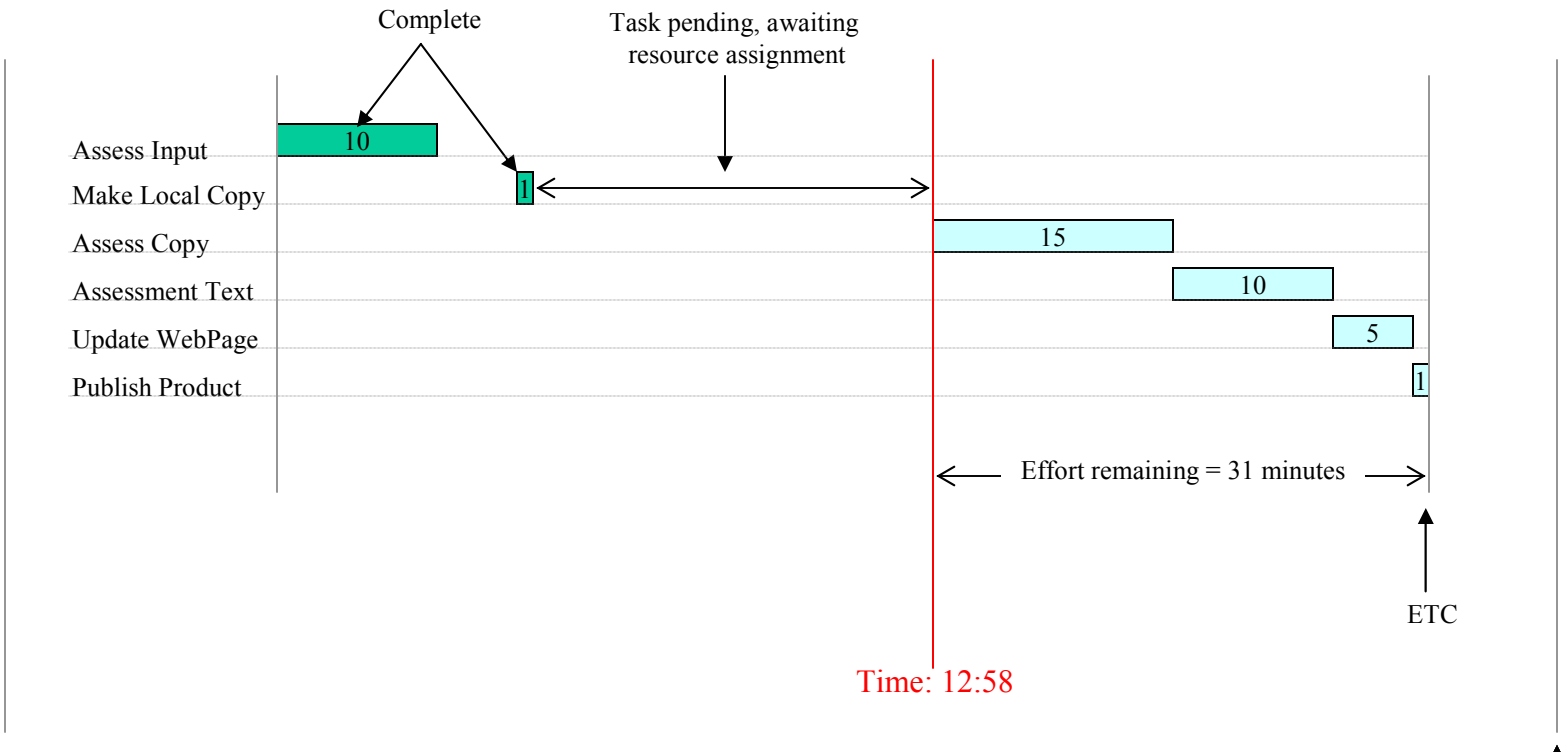
Current time: 12:32
Task Duration: 32 minutes remaining
ETC = 12:32 + 32 = 13:04
Slack time = 14:00 – 13:04 = 0:56

Scenario 2-5



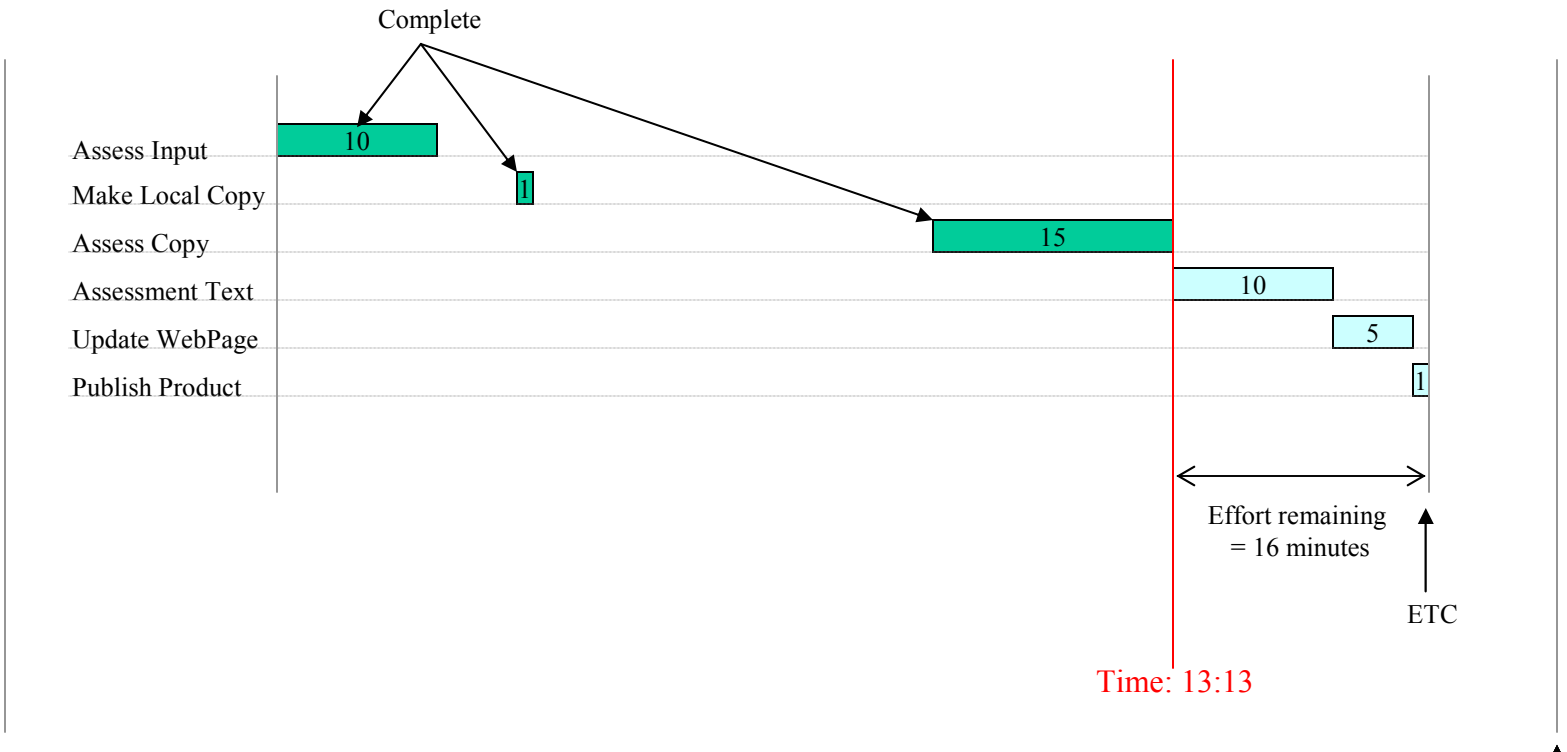
Current time: 12:33
Task Duration: 31 minutes remaining
ETC = 12:33 + 31 = 13:04
Slack time = 14:00 – 13:04 = 0:56

Scenario 2-6



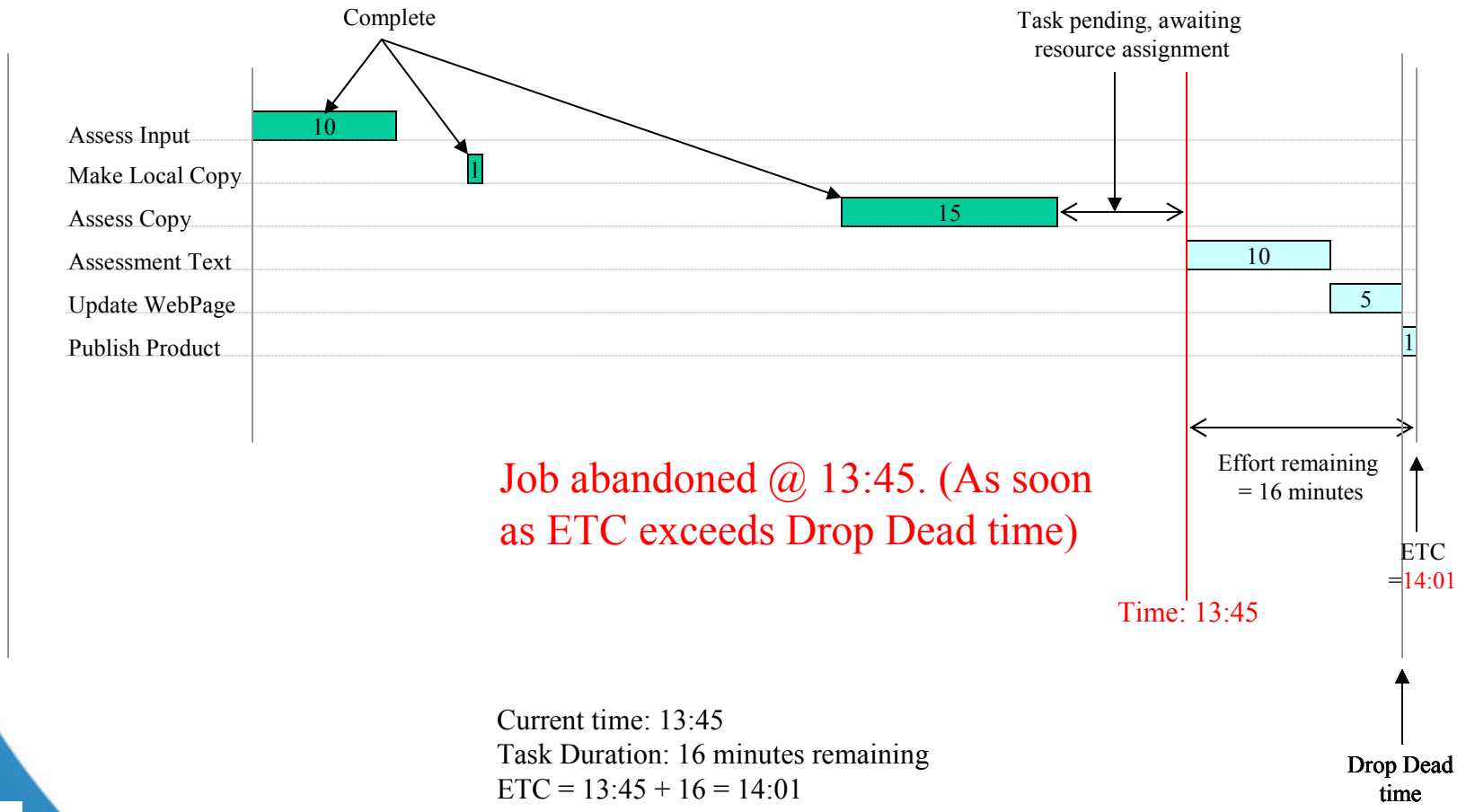
Current time: 12:58
Task Duration: 31 minutes remaining
ETC = 12:58 + 31 = 13:29
Slack time = 14:00 – 13:29 = 0:31

Scenario 2-7



Current time: 13:13
Task Duration: 16 minutes remaining
ETC = 13:13 + 16 = 13:29
Slack time = 14:00 – 13:29 = 0:31

Scenario 2-8



Job abandoned @ 13:45. (As soon as ETC exceeds Drop Dead time)

Current time: 13:45
Task Duration: 16 minutes remaining
ETC = 13:45 + 16 = 14:01
Slack time = 14:00 – 14:01 = -0:01

Contact

Mr Ronald W. Funk

Joint Staff Operational Research Team – Team Leader

DRDC Centre for Operational Research and Analysis

National Defence Headquarters (6CBS)

MGen George R. Pearkes Building

Ottawa Ontario Canada K1A 0K2

(613) 995-6887

Ron.Funk@drdc-rddc.gc.ca

Mr Richard L. Sorensen

Senior Systems Engineering Consultant

Vitech Corporation

2070 Chain Bridge Road, Suite 100

Vienna, VA 22182

(801) 776-5794

rsorensen@vitechcorp.com


DEFENCE
R&D
DÉFENSE