Control

☐ Job_Size_in_Man_Days(t) = Job_Size_in_Man_Days(t - dt) + (Rate_of_Change_in_Job_Size) * dt
INIT Job_Size_in_Man_Days = Development_Man_Days+Testing_Man_Days
  INFLOWS:
  ⌖ Rate_of_Change_in_Job_Size = Rate_of_Adjusting_Job_Size+
    Rate_of_Increase_in_Development_Man_Days+Rate_of_Increase_in_Testing_Man_Days

☐ Perceived_Job_Size(t) = Perceived_Job_Size(t - dt) + (Changes_in_Perceived_Job_Size) * dt
INIT Perceived_Job_Size = Perceived_Job_Size_DSI/DSI_per_Task
  INFLOWS:
  ⌖ Changes_in_Perceived_Job_Size = Rate_of_Incorporating_Discovered_Tasks_in_Project

☐ Tasks_Discovered(t) = Tasks_Discovered(t - dt) + (Rate_of_Discovering_Tasks -
Rate_of_Incorporating_Discovered_Tasks) * dt
INIT Tasks_Discovered = 0
  INFLOWS:
  ⌖ Rate_of_Discovering_Tasks = Undiscovered_Tasks*
    Percent_of_Undiscovered_Tasks_Discovered_per_Day/100
  OUTFLOWS:
  ⌖ Rate_of_Incorporating_Discovered_Tasks = max(Rate_of_Discovering_Tasks-
    Rate_of_Incorporating_Discovered_Tasks_in_Project,0)

☐ Testing_Size_in_Man_Days(t) = Testing_Size_in_Man_Days(t - dt) +
(Rate_of_Change_in_Testing_Size) * dt
INIT Testing_Size_in_Man_Days = Testing_Man_Days
  INFLOWS:
  ⌖ Rate_of_Change_in_Testing_Size = if Fraction_of_Effort_for_System_Testing>.9 then
    Rate_of_Increase_in_Testing_Man_Days else Rate_of_Increase_in_Testing_Man_Days+
    Rate_of_Adjusting_Job_Size

☐ Undiscovered_Tasks(t) = Undiscovered_Tasks(t - dt) + (-Rate_of_Discovering_Tasks) * dt
INIT Undiscovered_Tasks = Real_Job_Size_in_Tasks-Perceived_Job_Size
  OUTFLOWS:
  ⌖ Rate_of_Discovering_Tasks = Undiscovered_Tasks*
    Percent_of_Undiscovered_Tasks_Discovered_per_Day/100

○ Actual_Testing_Productivity = IF Cumulative_Testing_Man_Days = 0 then 0 else
Cumulative_Tasks_Tested/(Cumulative_Testing_Man_Days)

○ Assumed_Devel_Productivity = Projected_Devel_Productivity*Weight_to_Projected_Productivity+
Perceived_Devel_Productivity*(1-Weight_to_Projected_Productivity)

○ Delay_in_Adjusting_Job_Size = 3

○ Delay_in_Incorporating_Tasks = 10

○ Fraction_Value = if Max_Size_of_Additions_Tolerated = 0 then 0 else
Relative_Size_of_Discovered_Tasks/Max_Size_of_Additions_Tolerated

○ Man_Days_Perceived_Needed_to_Rework_Detected_Errors = Detected_Errors*
Perceived_Rework_MP_Needed_per_Error

○ Man_Days_Perceived_Remaining_for_New_Tasks = max(0,Man_Days_Remaining-
Man_Days_Perceived_Needed_to_Rework_Detected_Errors-
Man_Days_Perceived_Still_Needed_for_Testing)

○ Man_Days_Perceived_Still_Needed_for_New_Tasks = Tasks_Perceived_Remaining/
Assumed_Devel_Productivity

○ Man_Days_Perceived_Still_Needed_for_Testing = Tasks_Remaining_to_be_Tested/
Perceived_Testing_Productivity

- Man_Days_Remaining = max(.0001,Job_Size_in_Man_Days-Cumul_Man_Days_Expended)
- Man_Days_Reported_Still_Needed = Man_Days_Remaining+ Reported_Shortage_or_Excess_in_Man_Days
- Max_Size_of_Additions_Tolerated = .01
- Perceived_Devel_Productivity = Tasks_Developed/(Cumul_Man_Days_Expended- Cumulative_Testing_Man_Days)
- Perceived_Shortage_in_ManDays = Total_ManDays_Perceived_Still_Needed- Man_Days_Remaining
- Perceived_Size_of_Discovered_Tasks_in_Man_Days = Tasks_Discovered/ Assumed_Devel_Productivity
- Perceived_Testing_Productivity = smth1((if Cumulative_Tasks_Tested=0 then Planned_Testing_Productivity else Actual_Testing_Productivity),Time_to_Smooth_Test_Prod)
- Percent_Devel_Perceived_Complete = smth1(100-((Man_Days_Reported_Still_Needed- Man_Days_Perceived_Still_Needed_for_Testing)/(Job_Size_in_Man_Days- Testing_Size_in_Man_Days)*100),Reporting_Delay)
- Percent_of_Job_Actually_Worked = Tasks_Developed/Real_Job_Size_in_Tasks
- Percent_of_Job_Perceived_Worked = (Tasks_Developed/Perceived_Job_Size)*100
- Percent_Tasks_Reported_Complete = smth1((100-(Man_Days_Reported_Still_Needed/ Job_Size_in_Man_Days)*100),Reporting_Delay)
- Planned_Testing_Productivity = Perceived_Job_Size/Testing_Size_in_Man_Days
- Projected_Devel_Productivity = If Man_Days_Perceived_Remaining_for_New_Tasks = 0 then 0 else Tasks_Perceived_Remaining/(Man_Days_Perceived_Remaining_for_New_Tasks)
- Rate_of_Adjusting_Job_Size = (Man_Days_Reported_Still_Needed+Cumul_Man_Days_Expended- Job_Size_in_Man_Days)/Delay_in_Adjusting_Job_Size
- Rate_of_Incorporating_Discovered_Tasks_in_Project = delay(Rate_of_Discovering_Tasks,Delay_in_Incorporating_Tasks)
- Rate_of_Increase_in_Development_Man_Days = (Rate_of_Incorporating_Discovered_Tasks_in_Project/Assumed_Devel_Productivity)* Fraction_of_Additional_Tasks
- Rate_of_Increase_in_Testing_Man_Days = (Rate_of_Incorporating_Discovered_Tasks_in_Project/ Perceived_Testing_Productivity)*Fraction_of_Additional_Tasks
- Real_Job_Size_in_Tasks = Real_Job_Size_in_DSI/DSI_per_Task
- Relative_Size_of_Discovered_Tasks = if Man_Days_Perceived_Remaining_for_New_Tasks = 0 then 0 else Perceived_Size_of_Discovered_Tasks_in_Man_Days/ (Man_Days_Perceived_Remaining_for_New_Tasks)
- Reported_Shortage_or_Excess_in_Man_Days = Perceived_Shortage_in_ManDays- Handled_ManDays
- Reporting_Delay = 10
- Schedule_Pressure = (Total_ManDays_Perceived_Still_Needed-Man_Days_Remaining)/ Man_Days_Remaining
- Tasks_Perceived_Remaining = Perceived_Job_Size-Tasks_Developed
- Tasks_Remaining_to_be_Tested = Perceived_Job_Size-Cumulative_Tasks_Tested
- Time_to_Smooth_Test_Prod = 50
- Total_ManDays_Perceived_Still_Needed = Man_Days_Perceived_Still_Needed_for_New_Tasks+ Man_Days_Perceived_Needed_to_Rework_Detected_Errors+ Man_Days_Perceived_Still_Needed_for_Testing
- Weight_to_Projected_Productivity = Multiplier_for_Devel*Multiplier_for_Resources

⊘ Fraction_of_Additional_Tasks = GRAPH(Fraction_Value)
(0.00, 0.00), (0.2, 0.00), (0.4, 0.00), (0.6, 0.00), (0.8, 0.00), (1, 0.00), (1.20, 0.7), (1.40, 0.9), (1.60, 0.975), (1.80, 1.00), (2.00, 1.00)

⊘ Multiplier_for_Devel = GRAPH(Percent_of_Job_Perceived_Worked/100)
(0.00, 1.00), (0.1, 1.00), (0.2, 1.00), (0.3, 1.00), (0.4, 1.00), (0.5, 1.00), (0.6, 0.975), (0.7, 0.9), (0.8, 0.75), (0.9, 0.5), (1, 0.00)

⊘ Multiplier_for_Resources = GRAPH((1-Man_Days_Perceived_Remaining_for_New_Tasks/ (Job_Size_in_Man_Days-Testing_Size_in_Man_Days)))
(0.00, 1.00), (0.1, 1.00), (0.2, 1.00), (0.3, 1.00), (0.4, 1.00), (0.5, 1.00), (0.6, 0.975), (0.7, 0.9), (0.8, 0.75), (0.9, 0.5), (1, 0.00)

⊘ Percent_of_Undiscovered_Tasks_Discovered_per_Day = GRAPH(Percent_of_Job_Perceived_Worked)
(0.00, 0.00), (20.0, 0.4), (40.0, 2.50), (60.0, 5.00), (80.0, 10.0), (100, 100)

Enter Project Parameters
○ Avg_Daily_MP_per_Staff = 1
○ Perceived_Job_Size_DSI = 42880

Estimation
○ Development_Man_Days = .8*Man_Days
○ DSI_per_Task = 60
○ Man_Days = 2.4*19*(Perceived_Job_Size_DSI/1000)^1.05
○ Real_Job_Size_in_DSI = (Perceived_Job_Size_DSI)/Tasks_Underestimation_Fraction
○ Starting_Work_Force = ((Man_Days/TDEV)/Avg_Daily_MP_per_Staff)*.5
○ Tasks_Underestimation_Fraction = .67
○ TDEV = 47.5*(Man_Days/19)^.38
○ Testing_Man_Days = .2*Man_Days

Human Resources Sector
☐ Pros(t) = Pros(t - dt) + (Assimilation_Rate - Quit_Rate - Pro_Transfer_Rate) * dt
INIT Pros = Starting_Work_Force
INFLOWS:
⇥ Assimilation_Rate = Rookies/Assim_Delay
OUTFLOWS:
⇥ Quit_Rate = Pros/Avg_Empl_Time
⇥ Pro_Transfer_Rate = min((max(0,-Work_Force_Gap/Pro_Transfer_Delay))-Rookie_Transfer_Rate, Pros/dt)
☐ Rookies(t) = Rookies(t - dt) + (Hiring_Rate - Assimilation_Rate - Rookie_Transfer_Rate) * dt
INIT Rookies = 0
INFLOWS:
⇥ Hiring_Rate = max(0, Work_Force_Gap/Hiring_Delay)

OUTFLOWS:

    ⬡ Assimilation_Rate = Rookies/Assim_Delay

    ⬡ Rookie_Transfer_Rate = min(max(0,-Work_Force_Gap/Rookie_Transfer_Delay), Rookies/dt)

○ Assim_Delay = 80

○ Avg_Empl_Time = 673

○ Ceiling_on_Hires = FTE_Pros*Most_Hirees_per_FTE

○ Daily_MP_for_Training = Rookies*Trainers_per_Hiree

○ FTE_Pros = Pros*Avg_Daily_MP_per_Staff

○ Full_Time_Equiv_Work_Force = Total_Work_Force*Avg_Daily_MP_per_Staff

○ Hiring_Delay = 40

○ Most_Hirees_per_FTE = 3

○ Pro_Transfer_Delay = 10

○ Rookie_Transfer_Delay = 10

○ Total_Work_Force = Rookies+Pros

○ Trainers_per_Hiree = 0.2

○ Work_Force_Ceiling = Pros+Ceiling_on_Hires

○ Work_Force_Gap = Work_Force_Level_Sought-Total_Work_Force

○ Work_Force_Level_Sought = min(Work_Force_Ceiling,Workforce_Level_Needed)

Manpower Allocation Sector

☐ Cumul_Devel_Man_Days(t) = Cumul_Devel_Man_Days(t - dt) + (Devel_Man_Days_Rate) * dt
INIT Cumul_Devel_Man_Days = 0
  INFLOWS:

    ⬡ Devel_Man_Days_Rate = Daily_MP_for_DevTest*(1-Fraction_of_Effort_for_System_Testing)

☐ Cumul_Man_Days_Expended(t) = Cumul_Man_Days_Expended(t - dt) + (Expended_Rate) * dt
INIT Cumul_Man_Days_Expended = .0001
  INFLOWS:

    ⬡ Expended_Rate = Total_Daily_Manpower

☐ Cumul_QA_Man_Days(t) = Cumul_QA_Man_Days(t - dt) + (QA_MD_Rate) * dt
INIT Cumul_QA_Man_Days = 0
  INFLOWS:

    ⬡ QA_MD_Rate = Daily_MP_for_QA

☐ Cumul_Rework_Man_Days(t) = Cumul_Rework_Man_Days(t - dt) + (Rework_Man_Days_Rate) * dt
INIT Cumul_Rework_Man_Days = 0
  INFLOWS:

    ⬡ Rework_Man_Days_Rate = Daily_MP_for_Rework

○ Actual_Fraction_of_MP_for_QA = Planned_Fraction_of_MP_for_QA*(1+Actual_Fraction_Graph)

○ Daily_MP_Available_After_Training_Overhead = Total_Daily_Manpower-Daily_MP_for_Training

○ Daily_MP_for_DevTest = Daily_MP_for_SW_Prod-Daily_MP_for_Rework

○ Daily_MP_for_QA = min((Actual_Fraction_of_MP_for_QA*Total_Daily_Manpower),0.9*Daily_MP_Available_After_Training_Overhead)

○ Daily_MP_for_Rework = min((Desired_Error_Correction_Rate*Perceived_Rework_MP_Needed_per_Error),Daily_MP_for_SW_Prod)

○ Daily_MP_for_SW_Prod = Daily_MP_Available_After_Training_Overhead-Daily_MP_for_QA

○ Desired_Error_Correction_Rate = Detected_Errors/Desired_Rework_Delay

- ○ Desired_Rework_Delay = 15
- ○ Perceived_Rework_MP_Needed_per_Error = smth1(Actual_Rework_Manpower_Needed_per_Error,10)
- ○ Planned_Fraction_of_MP_for_QA = Planned_Fraction_Graph*(1+Quality_Objective/100)
- ○ Quality_Objective = 0
- ○ Total_Daily_Manpower = Total_Work_Force*Avg_Daily_MP_per_Staff
- ⊘ Actual_Fraction_Graph = GRAPH(Schedule_Pressure)

(0.00, 0.00), (0.1, -0.025), (0.2, -0.15), (0.3, -0.35), (0.4, -0.475), (0.5, -0.5)

- ⊘ Planned_Fraction_Graph = GRAPH(Percent_of_Job_Actually_Worked)

(0.00, 0.15), (0.1, 0.15), (0.2, 0.15), (0.3, 0.15), (0.4, 0.15), (0.5, 0.15), (0.6, 0.15), (0.7, 0.15), (0.8, 0.15), (0.9, 0.15), (1, 0.00)

Planning Sector

- ☐ Scheduled_Completion_Date(t) = Scheduled_Completion_Date(t - dt) + (Rate_of_Adjusting_Flow) * dt

  INIT Scheduled_Completion_Date = TDEV

  INFLOWS:
  - ⇴ Rate_of_Adjusting_Flow = (Indicated_Completion_Date-Scheduled_Completion_Date)/ Schedule_Adjustment_Time
- ○ Indicated_Completion_Date = time+Time_Perceived_Still_Remaining
- ○ Indicated_Work_Force_Level = (Man_Days_Remaining/(Time_Remaining+0.001))/ Avg_Daily_MP_per_Staff
- ○ Max_Tolerable_Completion_Date = 1e7*TDEV
- ○ Schedule_Adjustment_Time = 5
- ○ Time_Perceived_Still_Remaining = Man_Days_Remaining/(Work_Force_Level_Sought* Avg_Daily_MP_per_Staff)
- ○ Time_Remaining = max(Scheduled_Completion_Date-time, 0)
- ○ Willingness_to_Change_Workforce = max(WCWF1,WCWF2)
- ○ Workforce_Level_Needed = min((Willingness_to_Change_Workforce* Indicated_Work_Force_Level+(1-Willingness_to_Change_Workforce)* Total_Work_Force),Indicated_Work_Force_Level)
- ⊘ WCWF1 = GRAPH(Time_Remaining/(Hiring_Delay+Assim_Delay))

(0.00, 0.00), (0.3, 0.00), (0.6, 0.1), (0.9, 0.4), (1.20, 0.85), (1.50, 1.00), (1.80, 1.00), (2.10, 1.00), (2.40, 1.00), (2.70, 1.00), (3.00, 1.00)

- ⊘ WCWF2 = GRAPH(Scheduled_Completion_Date/Max_Tolerable_Completion_Date)

(0.86, 0.00), (0.88, 0.1), (0.9, 0.2), (0.92, 0.35), (0.94, 0.6), (0.96, 0.7), (0.98, 0.77), (1.00, 0.8)

Quality Assurance

☐ Cumulative_Detected_Errors(t) = Cumulative_Detected_Errors(t - dt) + (Count_Detected_Errors) * dt
INIT Cumulative_Detected_Errors = 0

   INFLOWS:
      ⇨ Count_Detected_Errors = Error_Detection_Rate
☐ Cumulative_Errors(t) = Cumulative_Errors(t - dt) + (Generation_Rate) * dt
INIT Cumulative_Errors = 0
   INFLOWS:
      ⇨ Generation_Rate = Error_Generation_Rate
☐ Detected_Errors(t) = Detected_Errors(t - dt) + (Error_Detection_Rate - Rework_Rate) * dt
INIT Detected_Errors = 0
   INFLOWS:
      ⇨ Error_Detection_Rate = min(Potential_Error_Detection_Rate,Potentially_Detectable_Errors/
         dt)
   OUTFLOWS:
      ⇨ Rework_Rate = Daily_MP_for_Rework/Actual_Rework_Manpower_Needed_per_Error
☐ Escaped_Errors(t) = Escaped_Errors(t - dt) + (Error_Escape_Rate) * dt
INIT Escaped_Errors = 0
   INFLOWS:
      ⇨ Error_Escape_Rate = QA_Rate*Average_#_Errors_per_Task
☐ Potentially_Detectable_Errors(t) = Potentially_Detectable_Errors(t - dt) + (Error_Generation_Rate -
Error_Detection_Rate - Error_Escape_Rate) * dt
INIT Potentially_Detectable_Errors = 0
   INFLOWS:
      ⇨ Error_Generation_Rate = Software_Development_Rate*
         Nominal_Errors_Committed_per_Task*Multiplier_Due_to_Schedule_Pressure*
         Multiplier_Due_to_Workforce_Mix
   OUTFLOWS:
      ⇨ Error_Detection_Rate = min(Potential_Error_Detection_Rate,Potentially_Detectable_Errors/
         dt)
      ⇨ Error_Escape_Rate = QA_Rate*Average_#_Errors_per_Task
☐ Reworked_Errors_During_Devel(t) = Reworked_Errors_During_Devel(t - dt) + (Rework_Rate) * dt
INIT Reworked_Errors_During_Devel = 0
   INFLOWS:
      ⇨ Rework_Rate = Daily_MP_for_Rework/Actual_Rework_Manpower_Needed_per_Error
☐ Tasks_Worked(t) = Tasks_Worked(t - dt) + (SD_RATE - QA__RATE) * dt
INIT Tasks_Worked = 0
   INFLOWS:
   ⇨ SD_RATE = Software_Development_Rate
   OUTFLOWS:
      ⇨ QA__RATE = QA_Rate
○ Actual_Rework_Manpower_Needed_per_Error = Nominal_Rework_Manpower_Needed_per_Error/
Multiplier_for_Losses
○ Average_#_Errors_per_Task = If Tasks_Worked = 0 then 0 else max(Potentially_Detectable_Errors/
(Tasks_Worked),0)
○ Average_QA_Delay = 10
○ Error_Density = Average_#_Errors_per_Task*(1000/DSI_per_Task)

- ○ Nominal_Errors_Committed_per_Task = Nominal_Errors_Committed_per_DSI* DSI_per_Task/ 1000
- ○ Percent_Errors_Detected = 100*Cumulative_Detected_Errors/(Cumulative_Errors+.001)
- ○ Potential_Error_Detection_Rate = Daily_MP_for_QA/QA_Manpower_Needed_to_Detect_an_Error
- ○ QA_Manpower_Needed_to_Detect_an_Error = Nominal_QA_Manpower_Needed_per_Error*(1/ Multiplier_for_Losses)*Multiplier_Due_to_Error_Density
- ○ QA_Rate = DELAY(Software_Development_Rate,Average_QA_Delay, 0)
- ⊘ Multiplier_Due_to_Error_Density = GRAPH(Error_Density)
  (0.00, 50.0), (1.00, 36.0), (2.00, 26.0), (3.00, 17.5), (4.00, 10.0), (5.00, 4.00), (6.00, 1.75), (7.00, 1.20), (8.00, 1.00), (9.00, 1.00), (10.0, 1.00)

- ⊘ Multiplier_Due_to_Schedule_Pressure = GRAPH(Schedule_Pressure)
  (-0.4, 0.9), (-0.2, 0.94), (-5.55e-17, 1.00), (0.2, 1.05), (0.4, 1.14), (0.6, 1.24), (0.8, 1.36), (1, 1.50)

- ⊘ Multiplier_Due_to_Workforce_Mix = GRAPH(Ratio_of_Pros_to_Rookies)
  (0.00, 2.00), (0.2, 1.80), (0.4, 1.60), (0.6, 1.40), (0.8, 1.20), (1, 1.00)

- ⊘ Nominal_Errors_Committed_per_DSI = GRAPH(Percent_of_Job_Actually_Worked)
  (0.00, 25.0), (0.2, 23.9), (0.4, 21.6), (0.6, 15.9), (0.8, 13.6), (1, 12.5)

- ⊘ Nominal_QA_Manpower_Needed_per_Error = GRAPH(Percent_of_Job_Actually_Worked)
  (0.00, 0.4), (0.1, 0.4), (0.2, 0.39), (0.3, 0.375), (0.4, 0.35), (0.5, 0.3), (0.6, 0.25), (0.7, 0.225), (0.8, 0.21), (0.9, 0.2), (1, 0.2)

- ⊘ Nominal_Rework_Manpower_Needed_per_Error = GRAPH(Percent_of_Job_Actually_Worked)
  (0.00, 0.6), (0.2, 0.575), (0.4, 0.5), (0.6, 0.4), (0.8, 0.325), (1, 0.3)

Software Development Productivity Subsector

- ▣ Actual_Fraction_of_ManDay_for_Project(t) = Actual_Fraction_of_ManDay_for_Project(t - dt) + (Work_Rate_Adj) * dt
  INIT Actual_Fraction_of_ManDay_for_Project = Nominal_Fraction_of_ManDays_for_Project
  INFLOWS:
  - ⧓ Work_Rate_Adj = (Work_Rate_Adj_Sought - Actual_Fraction_of_ManDay_for_Project) / Work_Rate_Adj_Time
- ▣ Exhaustion_Level(t) = Exhaustion_Level(t - dt) + (Exhaustion_Flow - Exhaustion_Depletion) * dt
  INIT Exhaustion_Level = 0
  INFLOWS:

⊟ Exhaustion_Flow = GRAPH((1-Actual_Fraction_of_ManDay_for_Project)/(1-Nominal_Fraction_of_ManDays_for_Project))
(-0.5, 2.50), (-0.4, 2.20), (-0.3, 1.90), (-0.2, 1.60), (-0.1, 1.30), (-5.55e-17, 1.15), (0.1, 0.9), (0.2, 0.8), (0.3, 0.7), (0.4, 0.6), (0.5, 0.5), (0.6, 0.4), (0.7, 0.3), (0.8, 0.2), (0.9, 0.00), (1, 0.00)

OUTFLOWS:

⊟ Exhaustion_Depletion = if Exhaustion_Flow = 0 then Exhaustion_Level / Exhaustion_Depletion_Delay_Time else 0

○ Avg_Nominal_Potential_Prod = (Ratio_of_Pros_to_Rookies*Nominal_Potential_Prod:_Pros)+(1-Ratio_of_Pros_to_Rookies) * Nominal_Potential_Prod:_Rookies

○ Effect_of_Work_Rate_Sought = if Work_Rate_Adj_Sought>Actual_Fraction_of_ManDay_for_Project then 1 else .75

○ Exhaustion_Depletion_Delay_Time = 20

○ Handled_ManDays = if Perceived_Shortage_in_ManDays > 0 then (min(Perceived_Shortage_in_ManDays, Max_Shortage_ManDays_Handled)) else -(Perceived_Excesses_in_ManDays)

○ Max_Boost_in_ManHours = 1

○ Max_Shortage_ManDays_Handled = (Overwork_Duration_Threshold * Max_Boost_in_ManHours * Full_Time_Equiv_Work_Force) * Willingness_to_Overwork

○ Max_Tolerable_Exhaustion = 50

○ Multiplier_for_Losses = Actual_Fraction_of_ManDay_for_Project*(1-Communication_Overhead)

○ Nominal_Fraction_of_ManDays_for_Project = .6

○ Nominal_Potential_Prod:_Pros = 1

○ Nominal_Potential_Prod:_Rookies = 0.5

○ Overwork_Duration_Threshold = Nominal_Overwork_Duration_Threshold * Multiplier_to_Overwork_Duration_Threshold

○ Perceived_Excesses_in_ManDays = max(0, Slack_Time_Growth * (Man_Days_Remaining - Total_ManDays_Perceived_Still_Needed))

○ Percent_Boost_in_Work_Rate_Sought = if Perceived_Shortage_in_ManDays > 0 then (Handled_ManDays / (Full_Time_Equiv_Work_Force * (Overwork_Duration_Threshold+.0001))) else (Handled_ManDays / (Total_ManDays_Perceived_Still_Needed- Handled_ManDays+.0001))

○ Potential_Productivity = Avg_Nominal_Potential_Prod*Multiplier_Due_to_Learning

○ Ratio_of_Pros_to_Rookies = Pros/(Rookies+Pros)

○ Software_Development_Productivity = Potential_Productivity*Multiplier_for_Losses

○ Willingness_to_Overwork = if Exhaustion_Depletion>0 and Exhaustion_Flow = 0 then 0 else 1

○ Work_Rate_Adj_Sought = (1 + Percent_Boost_in_Work_Rate_Sought) * Nominal_Fraction_of_ManDays_for_Project

○ Work_Rate_Adj_Time = Normal_Delay*Effect_of_Work_Rate_Sought

⊘ Communication_Overhead = GRAPH(Total_Work_Force)
(0.00, 0.00), (5.00, 0.015), (10.0, 0.06), (15.0, 0.135), (20.0, 0.24), (25.0, 0.375), (30.0, 0.54)

⊘ Multiplier_Due_to_Learning = GRAPH(Percent_of_Job_Actually_Worked)
(0.00, 1.00), (0.1, 1.01), (0.2, 1.03), (0.3, 1.06), (0.4, 1.09), (0.5, 1.15), (0.6, 1.20), (0.7, 1.22), (0.8, 1.25), (0.9, 1.25), (1, 1.25)

⊘ Multiplier_to_Overwork_Duration_Threshold = GRAPH(Exhaustion_Level/
Max_Tolerable_Exhaustion)
(0.00, 1.00), (0.1, 0.9), (0.2, 0.8), (0.3, 0.7), (0.4, 0.6), (0.5, 0.5), (0.6, 0.4), (0.7, 0.3), (0.8, 0.2), (0.9, 0.1), (1, 0.00)

⊘ Nominal_Overwork_Duration_Threshold = GRAPH(Time_Remaining)
(0.00, 0.00), (10.0, 10.0), (20.0, 20.0), (30.0, 30.0), (40.0, 40.0), (50.0, 50.0)

⊘ Normal_Delay = GRAPH(Time_Remaining)
(0.00, 2.00), (5.00, 3.50), (10.0, 5.00), (15.0, 6.50), (20.0, 8.00), (25.0, 9.50), (30.0, 10.0)

⊘ Slack_Time_Growth = GRAPH(Total_ManDays_Perceived_Still_Needed/Man_Days_Remaining)
(0.00, 0.00), (0.1, 0.2), (0.2, 0.4), (0.3, 0.55), (0.4, 0.7), (0.5, 0.8), (0.6, 0.9), (0.7, 0.95), (0.8, 1.00), (0.9, 1.00), (1, 1.00)

Software Development Sector

☐ Tasks_Developed(t) = Tasks_Developed(t - dt) + (Software_Development_Rate) * dt
INIT Tasks_Developed = 0
INFLOWS:
⌁ Software_Development_Rate = min((Daily_MP_for_Development*
Software_Development_Productivity),Tasks_Perceived_Remaining/dt)
○ Daily_MP_for_Development = Daily_MP_for_DevTest*(1-Fraction_of_Effort_for_System_Testing)
○ Daily_MP_for_Testing = Fraction_of_Effort_for_System_Testing*Daily_MP_for_DevTest
⊘ Fraction_of_Effort_for_System_Testing = GRAPH(Tasks_Perceived_Remaining/
Perceived_Job_Size)
(0.00, 1.00), (0.04, 0.5), (0.08, 0.28), (0.12, 0.15), (0.16, 0.05), (0.2, 0.00)

Testing

☐ Cumulative_Errors_Reworked_in_Testing(t) = Cumulative_Errors_Reworked_in_Testing(t - dt) +
(Cumulative_Errors_Rate) * dt
INIT Cumulative_Errors_Reworked_in_Testing = 0
INFLOWS:
⌁ Cumulative_Errors_Rate = Passive_Error_Detection_&_Correction_Rate+
Active_Error_Detection_&_Correction_Rate
☐ Cumulative_Tasks_QA'd(t) = Cumulative_Tasks_QA'd(t - dt) + (QA___RATE - Testing_Rate) * dt
INIT Cumulative_Tasks_QA'd = 0
INFLOWS:
⌁ QA___RATE = QA_Rate
OUTFLOWS:
⌁ Testing_Rate = min(Cumulative_Tasks_QA'd/dt,Daily_MP_for_Testing/
Testing_MP_per_Task)

☐ Cumulative_Tasks_Tested(t) = Cumulative_Tasks_Tested(t - dt) + (Testing_Rate) * dt
INIT Cumulative_Tasks_Tested = 0
  INFLOWS:
 ⌇⊙⋗ Testing_Rate = min(Cumulative_Tasks_QA'd/dt,Daily_MP_for_Testing/
     Testing_MP_per_Task)
☐ Cumulative_Testing_Man_Days(t) = Cumulative_Testing_Man_Days(t - dt) +
(Cumulative_Testing_MD_Rate) * dt
INIT Cumulative_Testing_Man_Days = 0
  INFLOWS:
  ⌇⊙⋗ Cumulative_Testing_MD_Rate = Daily_MP_for_Testing
☐ Undetected_Active_Errors(t) = Undetected_Active_Errors(t - dt) + (Active_Error_Gen_Rate +
Active_Error_Regen_Rate - Active_Error_Detection_&_Correction_Rate -
Active_Error_Retirement_Rate) * dt
INIT Undetected_Active_Errors = 0
  INFLOWS:
  ⌇⊙⋗ Active_Error_Gen_Rate = (Error_Escape_Rate+Bad_Fix_Gen_Rate)*
     Fraction_Escaping_Errors_that_will_be_Active
  ⌇⊙⋗ Active_Error_Regen_Rate = Software_Development_Rate*
     SMTH1(Active_Error_Density,Time_to_Smooth_Active_Error_Density)*
     Multiplier_to_Regen_Due_to_Error_Density
  OUTFLOWS:
  ⌇⊙⋗ Active_Error_Detection_&_Correction_Rate = min(Testing_Rate*
     Active_Error_Density,Undetected_Active_Errors/dt)
  ⌇⊙⋗ Active_Error_Retirement_Rate = Undetected_Active_Errors*Active_Errors_Retiring_Fraction
☐ Undetected_Passive_Errors(t) = Undetected_Passive_Errors(t - dt) +
(Active_Error_Retirement_Rate + Passive_Error_Gen_Rate - Passive_Error_Detection_&
_Correction_Rate) * dt
INIT Undetected_Passive_Errors = 0
  INFLOWS:
  ⌇⊙⋗ Active_Error_Retirement_Rate = Undetected_Active_Errors*Active_Errors_Retiring_Fraction
  ⌇⊙⋗ Passive_Error_Gen_Rate = (Error_Escape_Rate+Bad_Fix_Gen_Rate)*(1-
     Fraction_Escaping_Errors_that_will_be_Active)
  OUTFLOWS:
  ⌇⊙⋗ Passive_Error_Detection_&_Correction_Rate = min(Testing_Rate*
     Passive_Error_Density,Undetected_Passive_Errors/dt)
○ Active_Error_Density = if Cumulative_Tasks_QA'd = 0 then 0 else Undetected_Active_Errors/
  (Cumulative_Tasks_QA'd)
○ All_Errors = Potentially_Detectable_Errors+Detected_Errors+Reworked_Errors_During_Devel+
  Undetected_Active_Errors+Undetected_Passive_Errors+Cumulative_Errors_Reworked_in_Testing
○ All_Errors_Reworked = Cumulative_Errors_Reworked_in_Testing+
  Reworked_Errors_During_Devel
○ All_Errors_That_Escaped_or_Were_Generated = Undetected_Active_Errors+
  Undetected_Passive_Errors+Cumulative_Errors_Reworked_in_Testing
○ Bad_Fix_Gen_Rate = Rework_Rate*Percent_Bad_Fixes
○ Passive_Error_Density = IF Cumulative_Tasks_QA'd = 0 then 0 else Undetected_Passive_Errors/
  (Cumulative_Tasks_QA'd)
○ Percent_Bad_Fixes = .075
○ Percent_of_Tasks_Tested = Cumulative_Tasks_Tested/Perceived_Job_Size

○ Testing_Effort_Overhead = 1

○ Testing_MP_per_Error = .15

○ Testing_MP_per_Task = (Testing_Effort_Overhead*DSI_per_Task/1000+Testing_MP_per_Error*
(Passive_Error_Density+Active_Error_Density))/Multiplier_for_Losses

○ Time_to_Smooth_Active_Error_Density = 40

⊘ Active_Errors_Retiring_Fraction = GRAPH(Percent_of_Job_Actually_Worked)
(0.00, 0.00), (0.1, 0.00), (0.2, 0.00), (0.3, 0.00), (0.4, 0.01), (0.5, 0.02), (0.6, 0.03), (0.7, 0.04), (0.8, 0.1), (0.9, 0.3), (1, 1.00)

⊘ Fraction_Escaping_Errors_that_will_be_Active = GRAPH(Percent_of_Job_Actually_Worked)
(0.00, 1.00), (0.1, 1.00), (0.2, 1.00), (0.3, 1.00), (0.4, 0.95), (0.5, 0.85), (0.6, 0.5), (0.7, 0.2), (0.8, 0.075), (0.9, 0.00), (1, 0.00)

⊘ Multiplier_to_Regen_Due_to_Error_Density = GRAPH(smth1(Active_Error_Density*1000/
DSI_per_Task,Time_to_Smooth_Active_Error_Density))
(0.00, 1.00), (10.0, 1.10), (20.0, 1.20), (30.0, 1.32), (40.0, 1.45), (50.0, 1.60), (60.0, 2.00), (70.0, 2.50), (80.0, 3.25), (90.0, 4.35), (100, 6.00)