

THE LONE STAR STATE

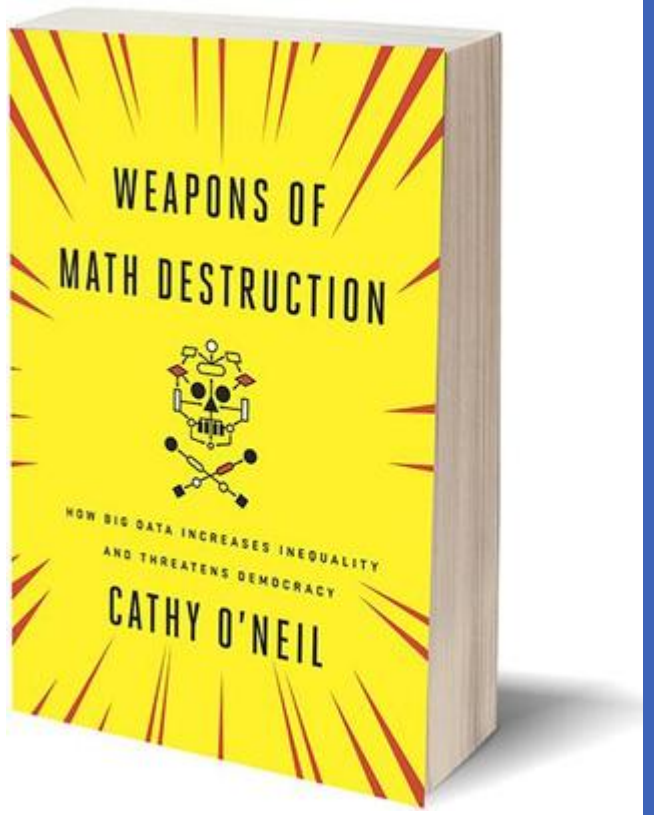
NetPoint+GPM CONFERENCE

AUSTIN, TX | MARCH 1 & 2, 2018



Better Algorithms to Schedule By

Presented by:
Dr. Gui Ponce de Leon, PE, PMP



"O'Neill calls on modelers to take more responsibility for their algorithms…But in the end, it's up to us to become more savvy about the models that govern our lives."

Algorithms have become an integral part of our daily lives. From finance to healthcare to entertainment, we rely on algorithms to filter information and when making important decisions. In project management, algorithms are essential to scheduling, risk analysis, and beyond. In NetPoint®, algorithms compute gaps and drifts/floats/total floats in real-time, author network layout, and generate weather-impacted schedules while preserving total float continuity. NetRisk™ algorithms carry-out a progress simulation and predict *safe floats* with respect to the targeted p-value completion date. In Schedule MD™, algorithms analyze hundreds of schedule metrics to generate schedule reliability scores. Join Dr. Gui Ponce de Leon as he discusses how PMA algorithms improve planning and scheduling and have helped make NetPoint, NetRisk, and Schedule MD indispensable tools in project management.

Meet Dr. Gui



Dr. Gui Ponce de Leon, PE, PMP
Chief Executive Officer

gui@pmaconsultants.com

Dr. Gui is recognized by his peers as one of our nation's foremost planning and scheduling experts. He has pioneered innovations in project management throughout his career. He holds four US patents on his graphical path method (GPM) and has numerous applications pending with the US Patent and Trademark Office. He and a growing number of project planners believe that GPM has the potential to become the dominant network-based planning and scheduling method by 2022.

Better Algorithms to Schedule By



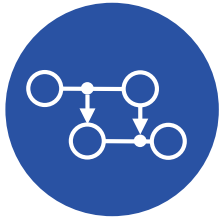
CPM Scheduling Algorithms

1957-1959 vintage, inapt for surface computing and ineffectual for randomized solutions



GPM Algorithms

2004-2009 vintage, better suited for planning and scheduling and for randomized schedule solutions



GPM Scheduling Algorithms

Gap–not date–based, self-healing, total floats left of the data date, push and/or pull planning



GPM Schedule Risk Algorithms

Each iteration is a progress simulation, floating is allowed, early & late completion risk, safe float



Network Layout Authoring Algorithms

Improve layout by utilizing strategies to optimize object locations under layout and aesthetic constraints



What the Future Holds

Countless algorithms genetically suitable for cloud and surface computing and for AI schedule solutions



Graphical Path Method

The graphical path method (GPM) is similar to the critical path method (CPM) but embodies a simpler scheme of thought in ways CPM can’t

CPM scheduling algorithms were developed in **1957-1959** and remain unchanged

GPM algorithms were initially developed in **2004-2009** and continue to be expanded

1960 1970 1980 1990 2000 2010 2020 2030

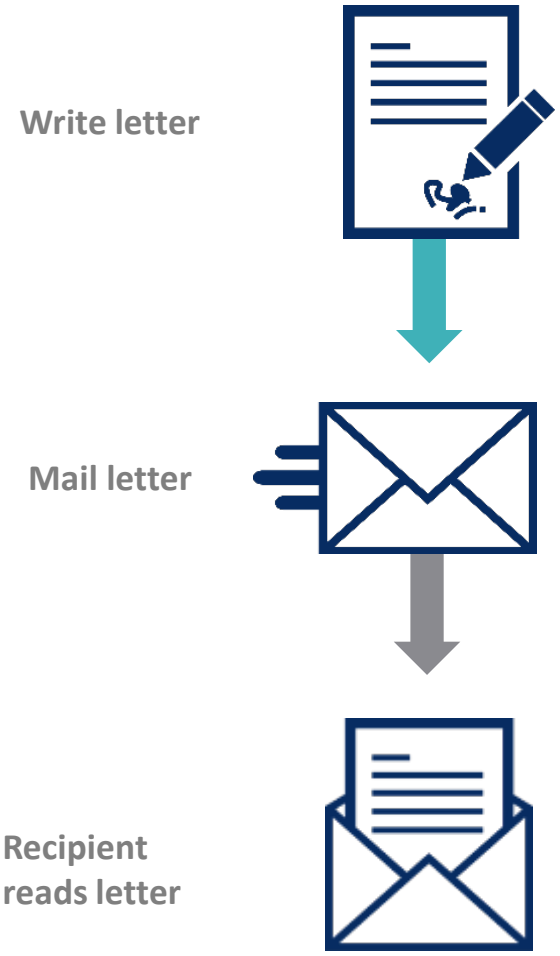
CPM

The method in a nutshell

- Networking method that, following any input for any activity, logic tie, or milestone, requires a forward pass and a backward pass for the entire network as a preceding step to obtain a separate output of the schedule
- Date constraints are required to schedule activities on planned dates
- Neither total floats nor the as-built critical path can be calculated left of the data date

METAPHOR

Written Communication:



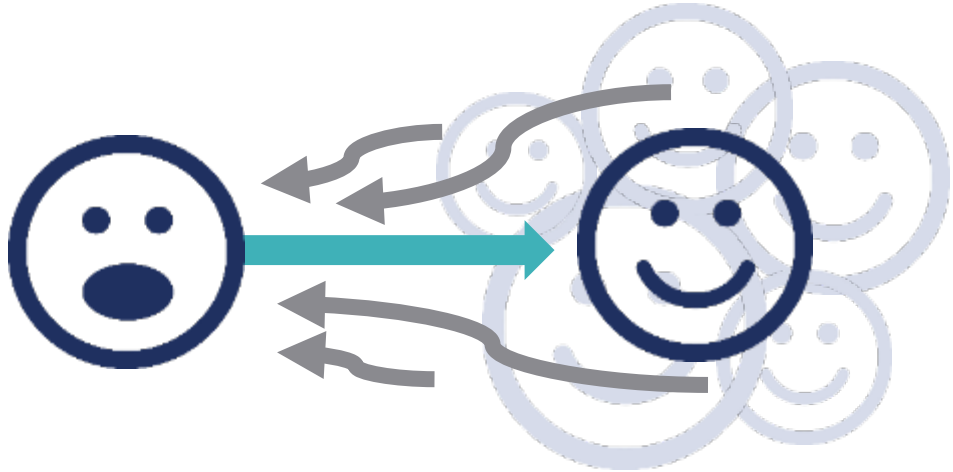
GPM

The method in a nutshell

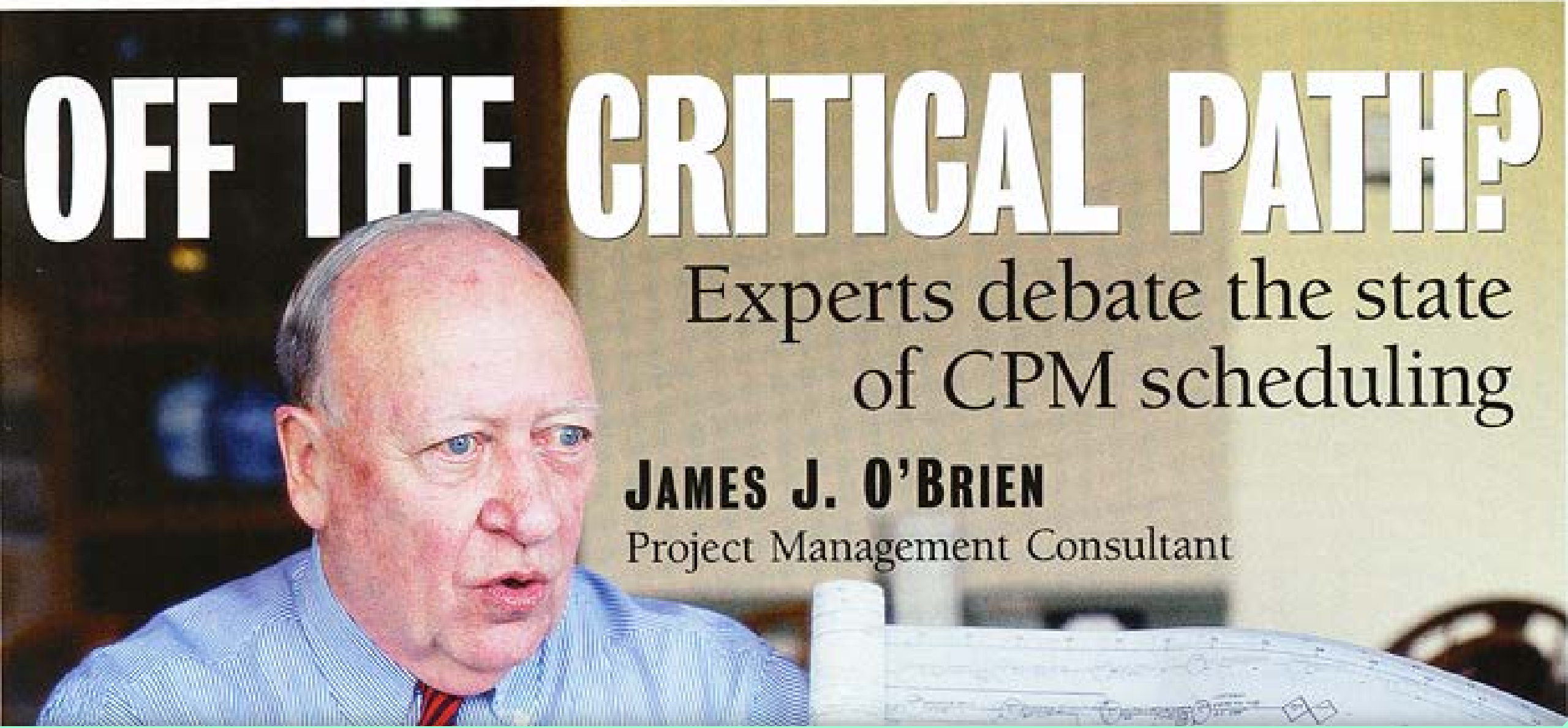
- Graphical & visualization networking method that, without a forward or backward pass, kinetically refreshes the schedule only where impacted while the user adds/deletes/revises/ repositions activities, logic ties, and milestones
- Activities on planned dates may float back (in GPM lexicon, have drift > 0)
- Total floats and the as-built critical path are algorithmically calculated left of the data date and refresh as the schedule is changed right of the data date

METAPHOR

Verbal Communication:



enr.com The McGraw-Hill Companies May 26, 2003 $5

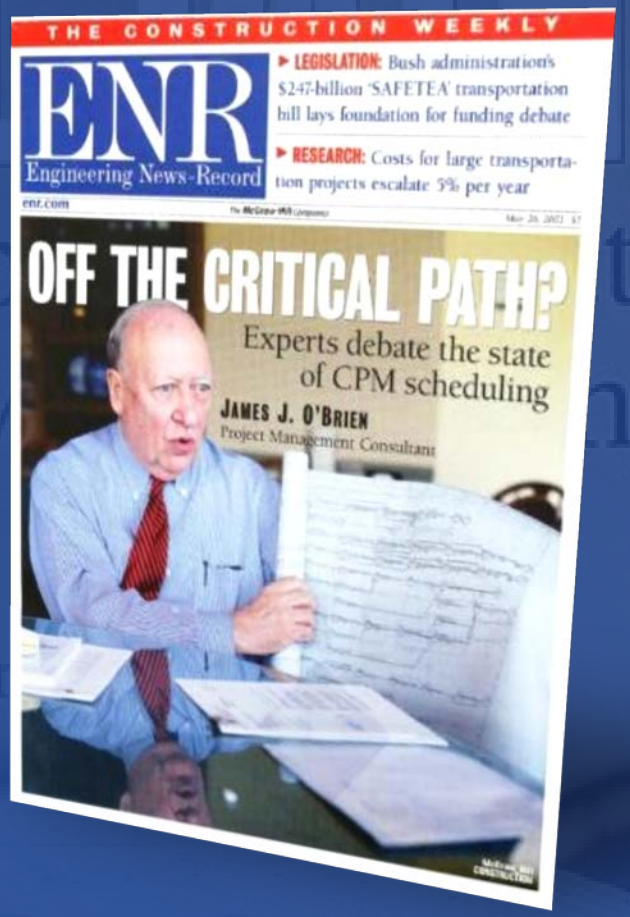


The Fork in the CPM Road

Snippets of how CPM Went off the rails

"What is described as a CPM schedule these days sometimes is not one at all"

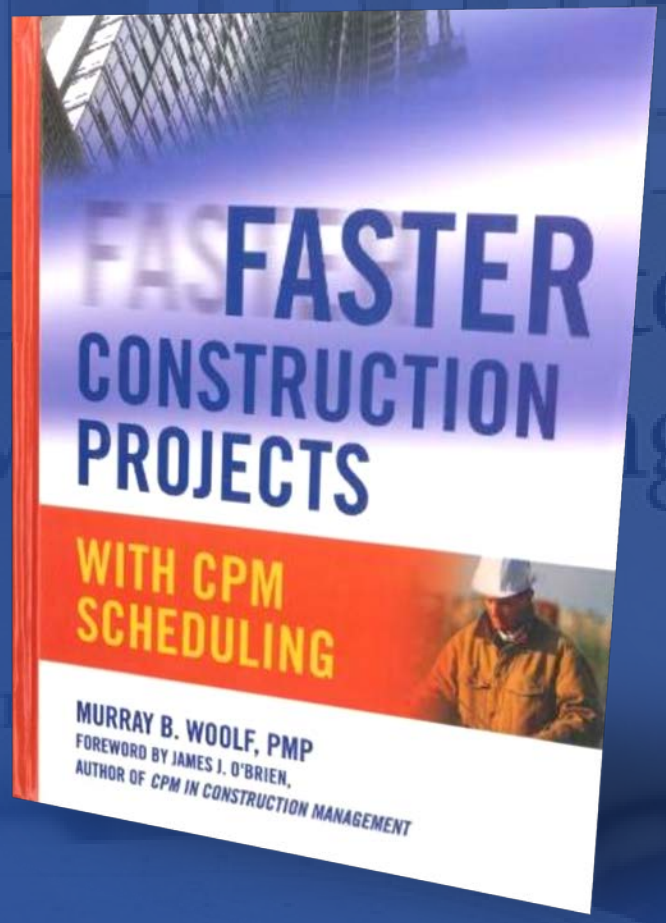
"They say they see widespread abuses of powerful software to produce badly flawed or deliberately deceptive schedules that look good but lack mathematical coherence or common sense about how the industry works"



More on how CPM Went off the rails

“we have collectively evolved the profession to where planning is no longer the essential first step in the scheduling process”

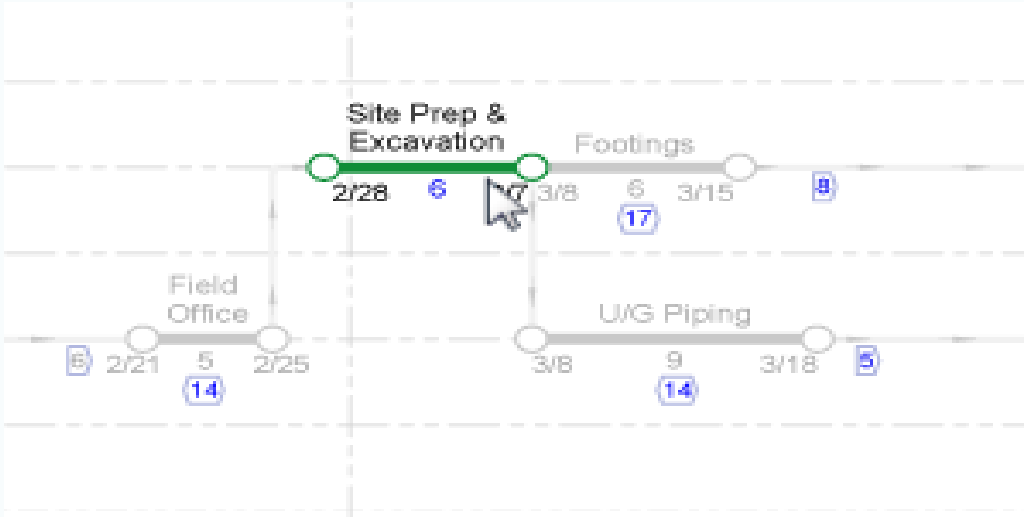
“Among the young guys, computers have made it easy to slap together something that looks right, but there is a thought process that must be involved, and it is hard to tell in many contemporary schedules if the thinking happened or not”



GPM

as applied in planning/ scheduling software

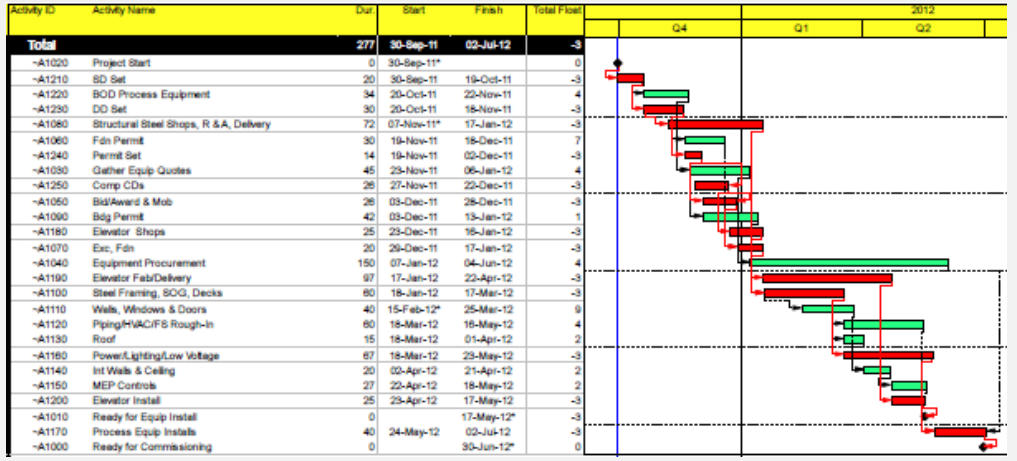
Graphical & visualization apps that use GPM self-healing and object encapsulation/message passing algorithms to synchronously cause the schedule to return to its correct mathematical state while stakeholders working on the display surface add/delete/revise/reposition activities, links, milestones, and benchmarks



CPM

as applied in scheduling software

Textual apps that use a stand-by CPM engine to carry out schedule calculations (i.e. forward pass and backward pass) and another stand-by output engine to displays schedule results, generally as a textual Gantt chart, all in a succession of multiple batch modes

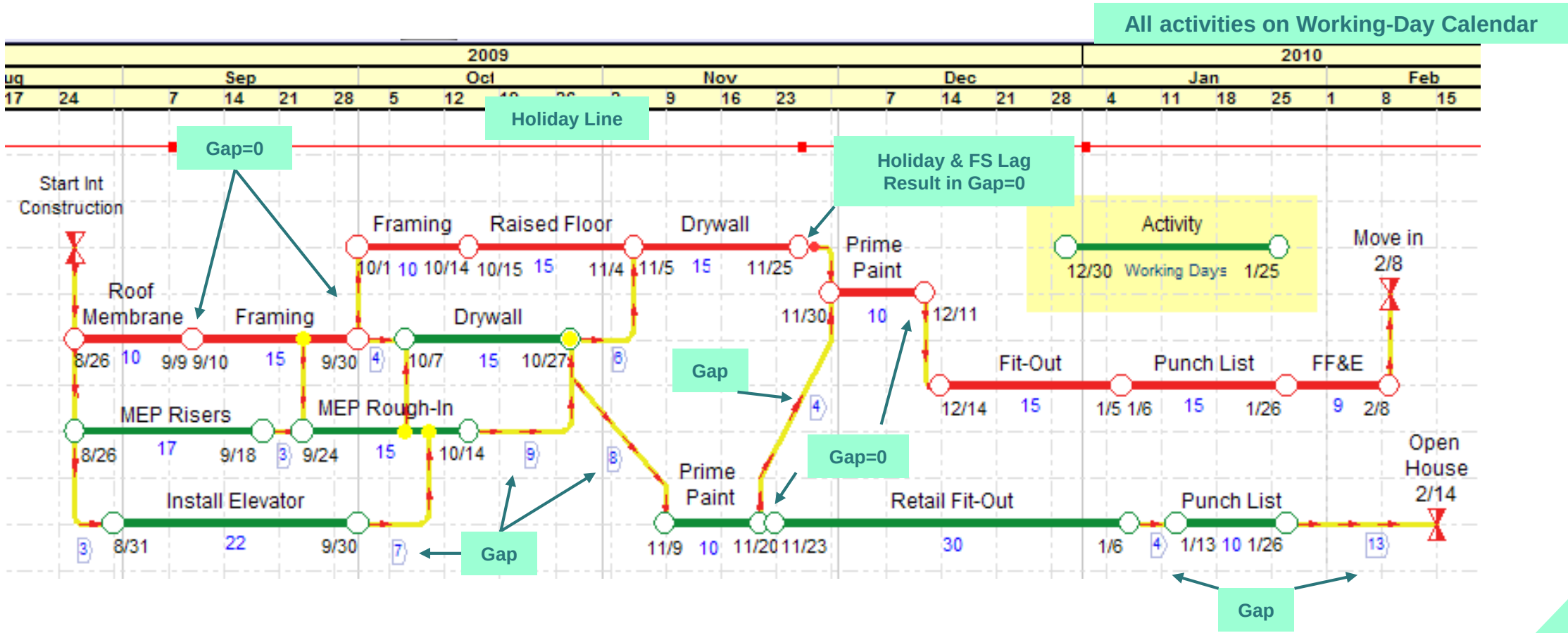


Activity ID	Activity Name	Dur	Start	Finish	Total Float
Total		277	30-Sep-11	02-Jul-12	-3
A1020	Project Start	0	30-Sep-11*		0
A1210	SD Set	20	30-Sep-11	19-Oct-11	-3
A1220	BOD Process Equipment	34	20-Oct-11	22-Nov-11	4
A1230	DD Set	30	20-Oct-11	18-Nov-11	-3
A1080	Structural Steel Shops, R &A, Delivery	72	07-Nov-11*	17-Jan-12	-3
A1060	Fdn Permit	30	19-Nov-11	18-Dec-11	7
A1240	Permit Set	14	19-Nov-11	02-Dec-11	-3
A1030	Gather Equip Quotes	45	23-Nov-11	06-Jan-12	4
A1250	Comp CDs	26	27-Nov-11	22-Dec-11	-3
A1050	Bid/Award & Mob	26	03-Dec-11	28-Dec-11	-3
A1090	Bdg Permit	42	03-Dec-11	13-Jan-12	1
A1180	Elevator Shops	25	23-Dec-11	16-Jan-12	-3
A1070	Exc, Fdn	20	29-Dec-11	17-Jan-12	-3
A1040	Equipment Procurement	150	07-Jan-12	04-Jun-12	4
A1190	Elevator Fab/Delivery	97	17-Jan-12	22-Apr-12	-3
A1100	Steel Framing, SOG, Decks	60	18-Jan-12	17-Mar-12	-3
A1110	Walls, Windows & Doors	40	15-Feb-12*	25-Mar-12	9
A1120	Piping/HVAC/FS Rough-In	60	18-Mar-12	18-May-12	4
A1130	Roof	15	18-Mar-12	01-Apr-12	2
A1160	Power/Lighting/Low Voltage	67	18-Mar-12	23-May-12	-3
A1140	Int Walls & Ceiling	20	02-Apr-12	21-Apr-12	2
A1150	MEP Controls	27	22-Apr-12	18-May-12	2
A1200	Elevator Install	25	23-Apr-12	17-May-12	-3
A1010	Ready for Equip Install	0		17-May-12*	-3
A1170	Process Equip Installs	40	24-May-12	02-Jul-12	-3
A1000	Ready for Commissioning	0		30-Jun-12*	0

Link *GAP*

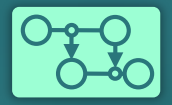
The Crux in GPM Scheduling Algorithms

Gap – for an activity-to-activity link, gap measures the extent the activity may be delayed *beyond its position* and not delay the successor beyond its position; gap values reflect the type of PDM relationship conveyed by a link, and drive all drifts, floats and therefore total floats

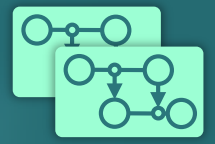


GPM Scheduling Algorithms–*With Optional, Drift-Preserving Planned Dates*

gap $_{I,J}$ = relevant date $_{j}$ – relevant date $_{I}$	measured using the predecessor calendar
drift-gap $_{I,H}$ = relevant date $_{I}$ – relevant date $_{H}$	measured using the successor calendar



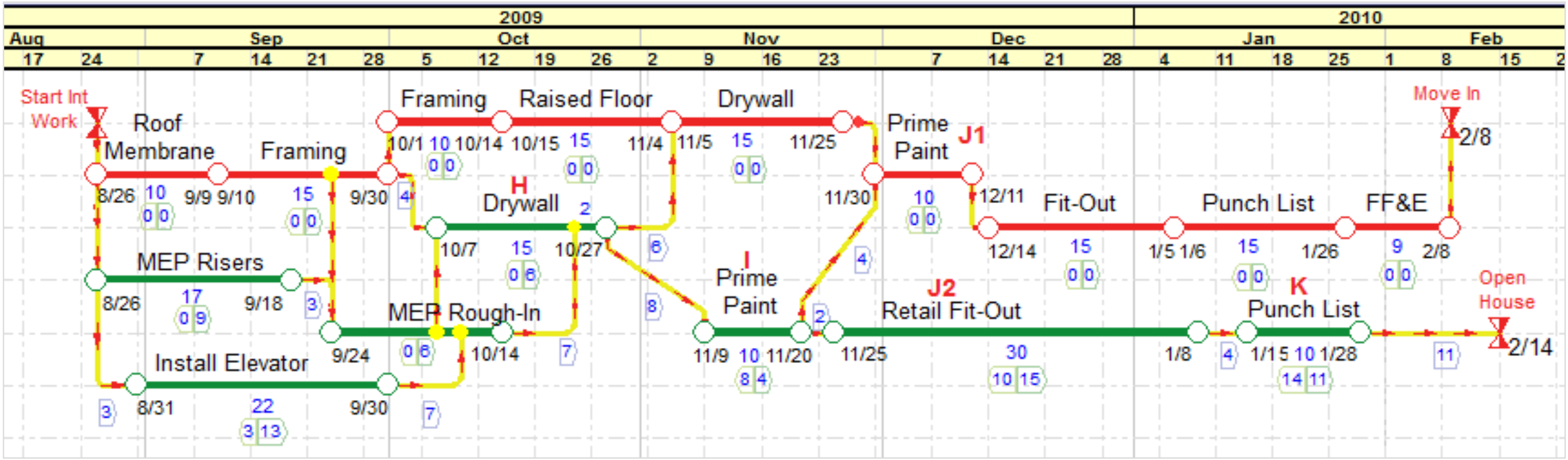
Where one calendar applies to all activities, d**rift-gap = gap**



With multiple calendars, drift-gap exists “under the hood” and its sole purpose is for calculating drift, and both the float and drift formulae are applied after fully floating and fully drifting the activity, again, only under the hood

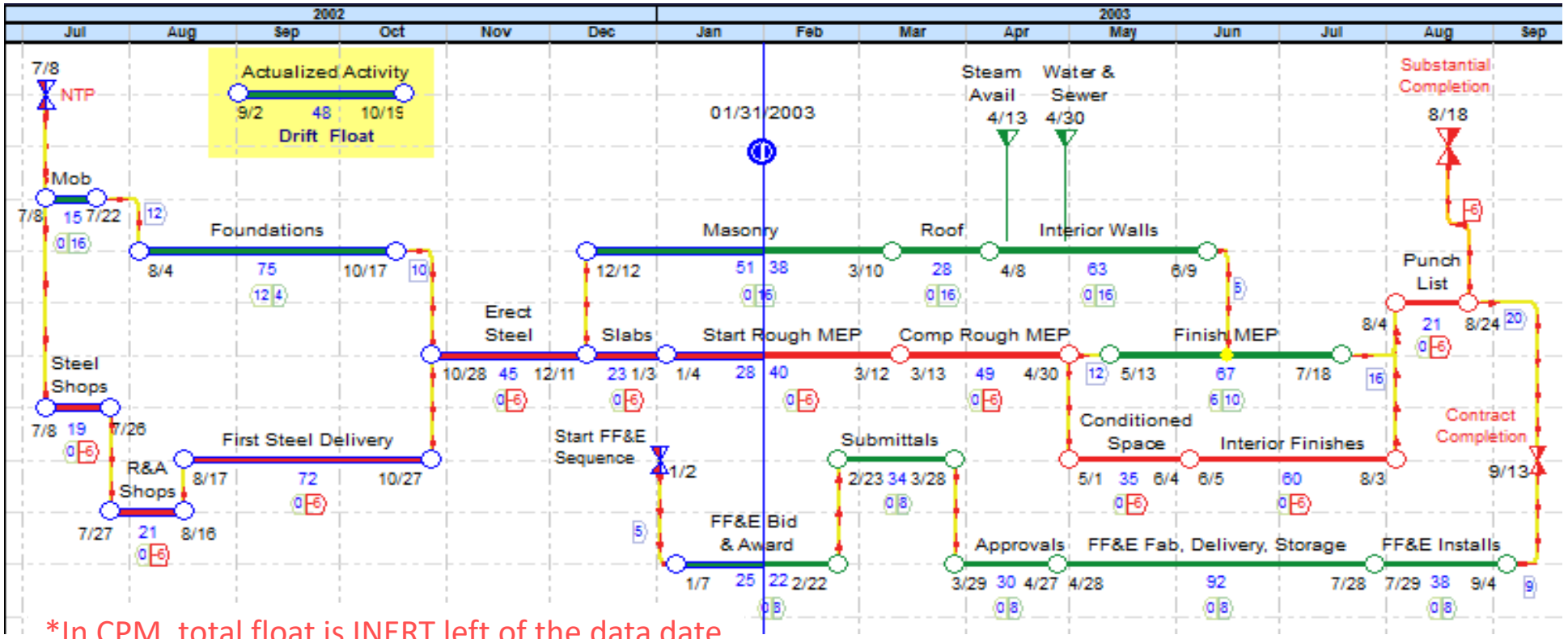
GPM Scheduling Algorithms–*With Optional, Drift-Preserving Planned Dates (cont'd)*

float I = min [gap I,J + float J]	for all J activities that are successors to activity I
drift I = min [drift-gap H,I + drift H]	for all H activities that are predecessors to activity I
total float I = float I + drift I	



The Concept of *Forensic* Total Floats in GPM*

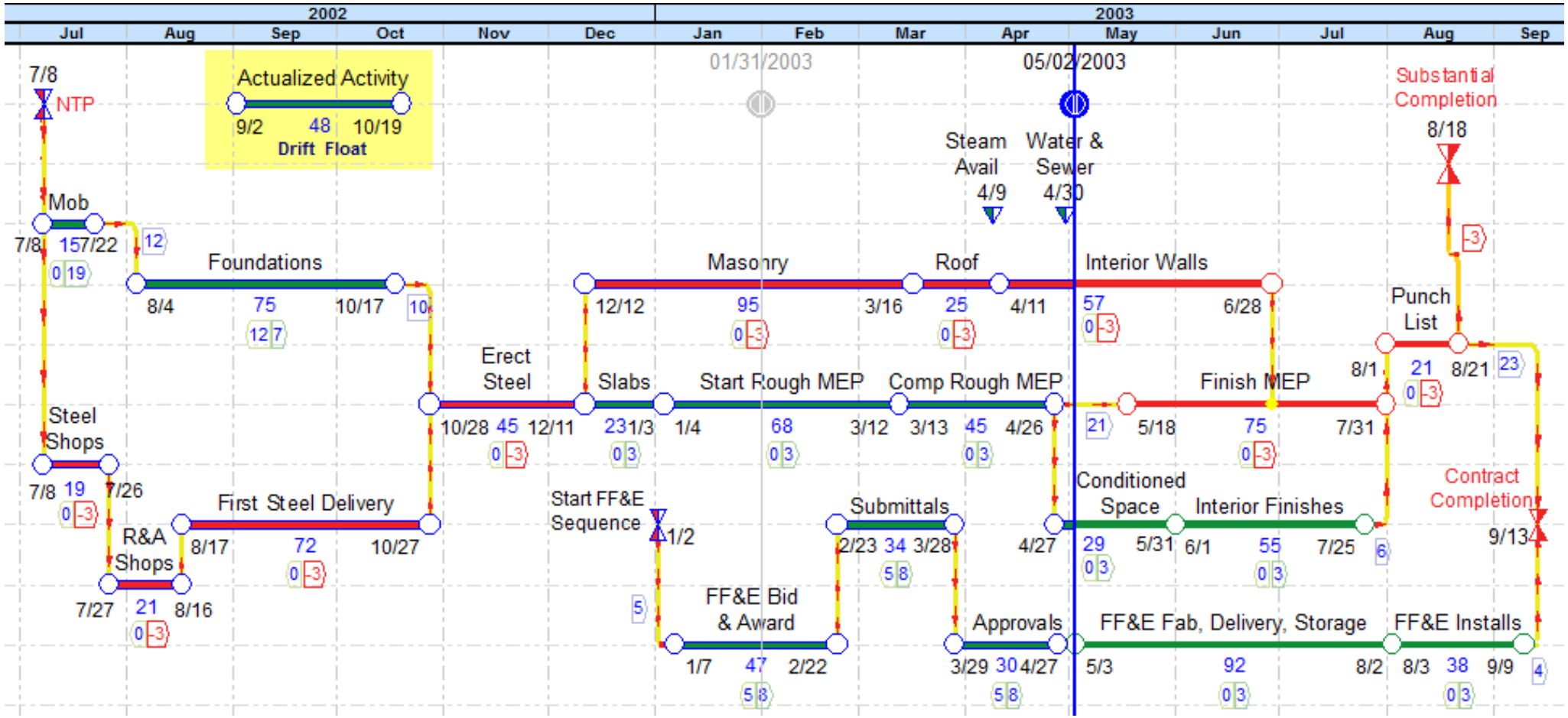
- Using gaps ***both sides*** of the data date, GPM calculates:
  - Forensic float & forensic drift for fully-actualized activities;
  - Forensic drift + forensic float = forensic total float
- For a data date (DD), the *then-current* critical path left of the DD is the activity/logic chain with the least forensic total float



*In CPM, total float is INERT left of the data date

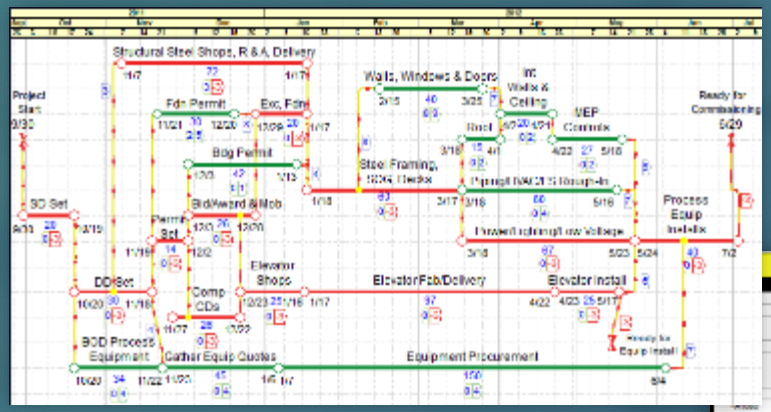
The Concept of *Forensic* Total Floats (*cont'd)*

- For any update, forensic total floats reflect current activity durations and logic right of the DD
  - Because floats/drifts/total floats left of the DD respond to activity/logic revisions right of the DD, the critical path(s) left of the DD may vary from update to update

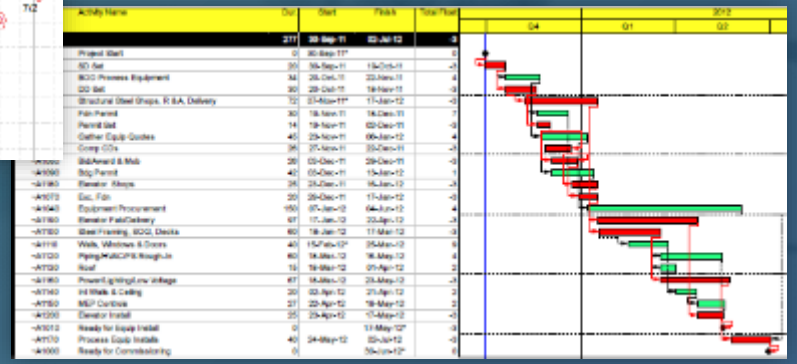


The Case for Constraint-Based Network Layout Authoring Algorithms

Stakeholders are better able to comprehend schedule information when presented as a rheonomic (time-scaled) activity network vs. in a water-fall Gantt chart, textual form



Activity Network



Gantt Chart

Why #1

Network properties such as activity fragnets, logic chain flow, total float continuity, open and orphan ends, logic malpractice, etc. are far more readily apparent in a network than in a textual Gantt chart

Why #2

When activity clustering into fragnets, logic chain flow, total float continuity, and defective logic ties are better visualized, stakeholders can more readily grasp how the network reacts to changes

Constraint-based diagram layout authoring algorithms lay out objects on the graphic display by utilizing strategies to optimize object location under predetermined layout constraints (e.g., NO overlays) and aesthetic constraints (e.g., penalize crossings)

NetPoint's AutoGRAPH™ Network Diagram Layout Authoring Strategy

Global or swim-lane network layout on the canvass grid with coordinates X = time, Y = grid

Identify and enqueue objects by clusters or individually, a cluster comprising a chain of activities

Prioritize enqueuing of clusters and avoid any overlap between activity/milestone/benchmark

Locate optimal position for each object that is NOT X-Y LOCKED by using constraints and penalties that result in layout scores

Optimize layout scores to reduce drawing area, link overlaps, activity-link crossings and link-link crossings

Fine-tuning includes link shape and routing adjustments and deleting or hiding redundant links and hiding large-gap links

AutoGRAPH™ Constraint-Based Layout Authoring Engine

Deploys an enqueuing algorithm (e.g., breadth-first search*) to find all feasible Y coordinates that satisfy layout constraints, whether by cluster or object by object

- Evaluates routing options for links based on aesthetic constraints
- Manipulates object into Y coordinates that best meet selected aesthetic constraints

Clusters are either float chains or every logic chain from start to end node; object-by-object search is based on logic steps

- Logic step of an activity is one step higher than the highest logic step of all its predecessors (Fondahl 1962)

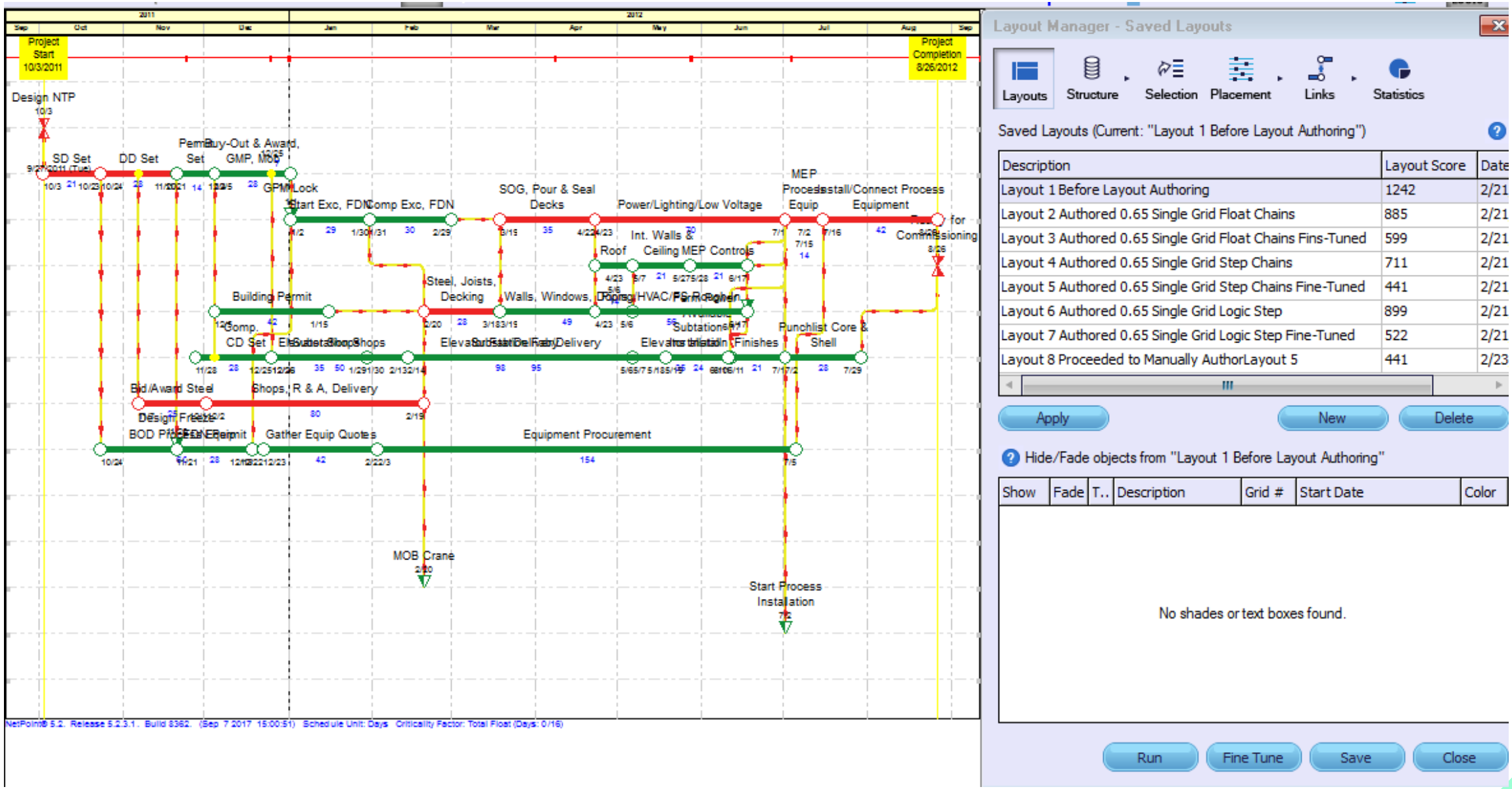
Object overlapping is precluded by foregoing any position on the canvass which is already occupied (0/1 condition)

Link penalty concept is used to optimize link routing

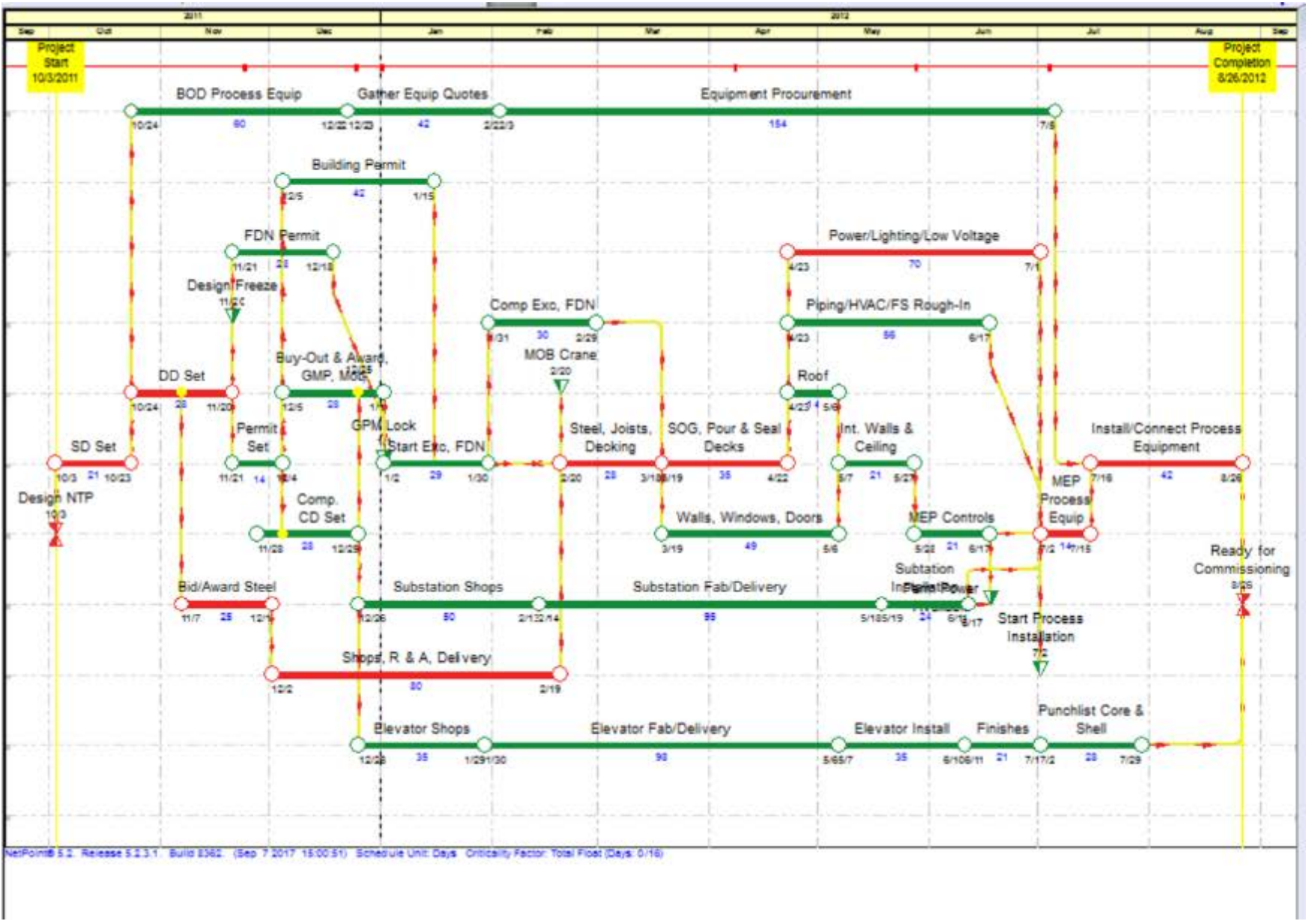
- Customizable link penalties score seven types of link layout inefficiencies, including, but not limited to link-activity crossings and link-link crossings, opposing link leg overlaps , and coinciding link leg overlaps

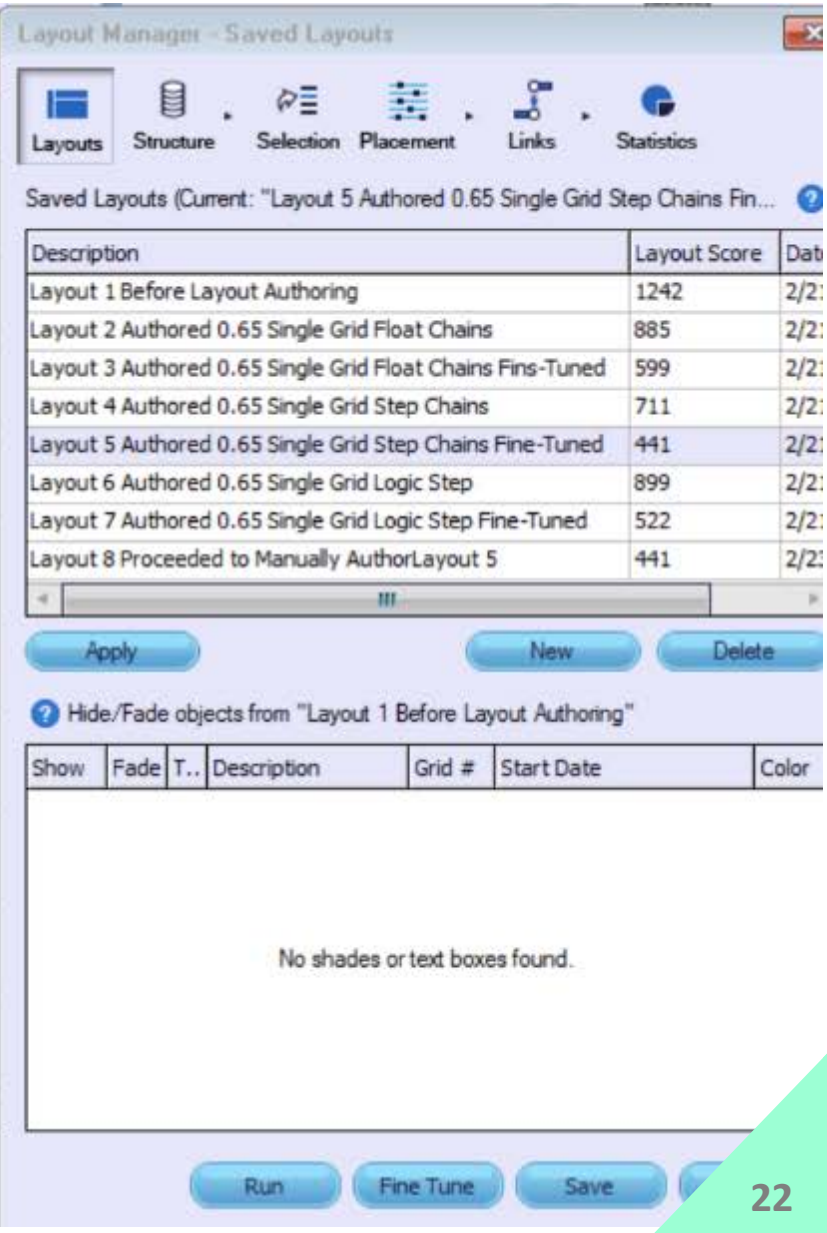
*Breadth-first search is a search strategy in a graph that, after enqueuing an object, gains access to objects that neighbor the enqueued object.

Imported Network Before Layout Authoring

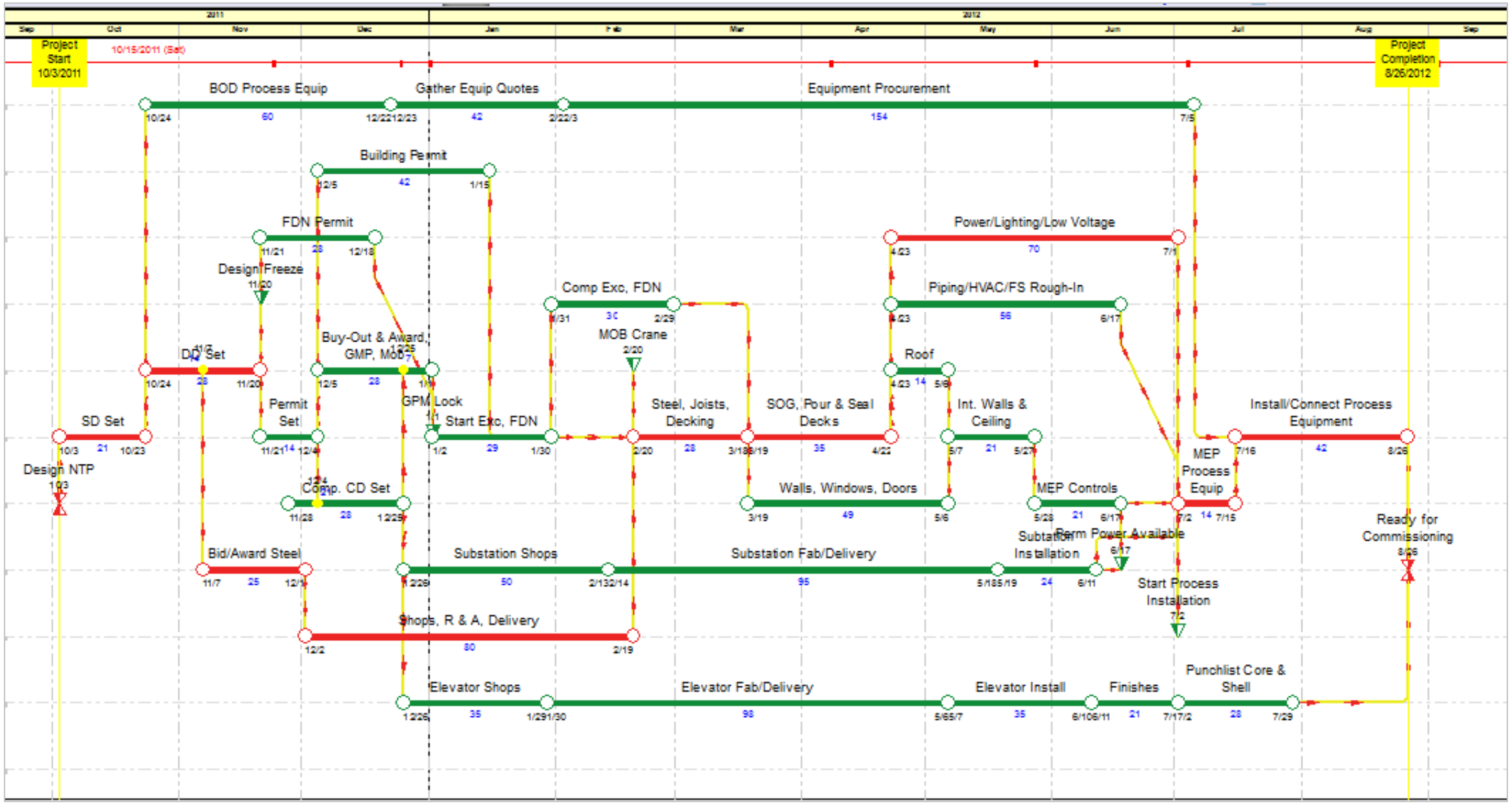


Lowest Layout Score Network Layout Authored by AutoGRAPH™

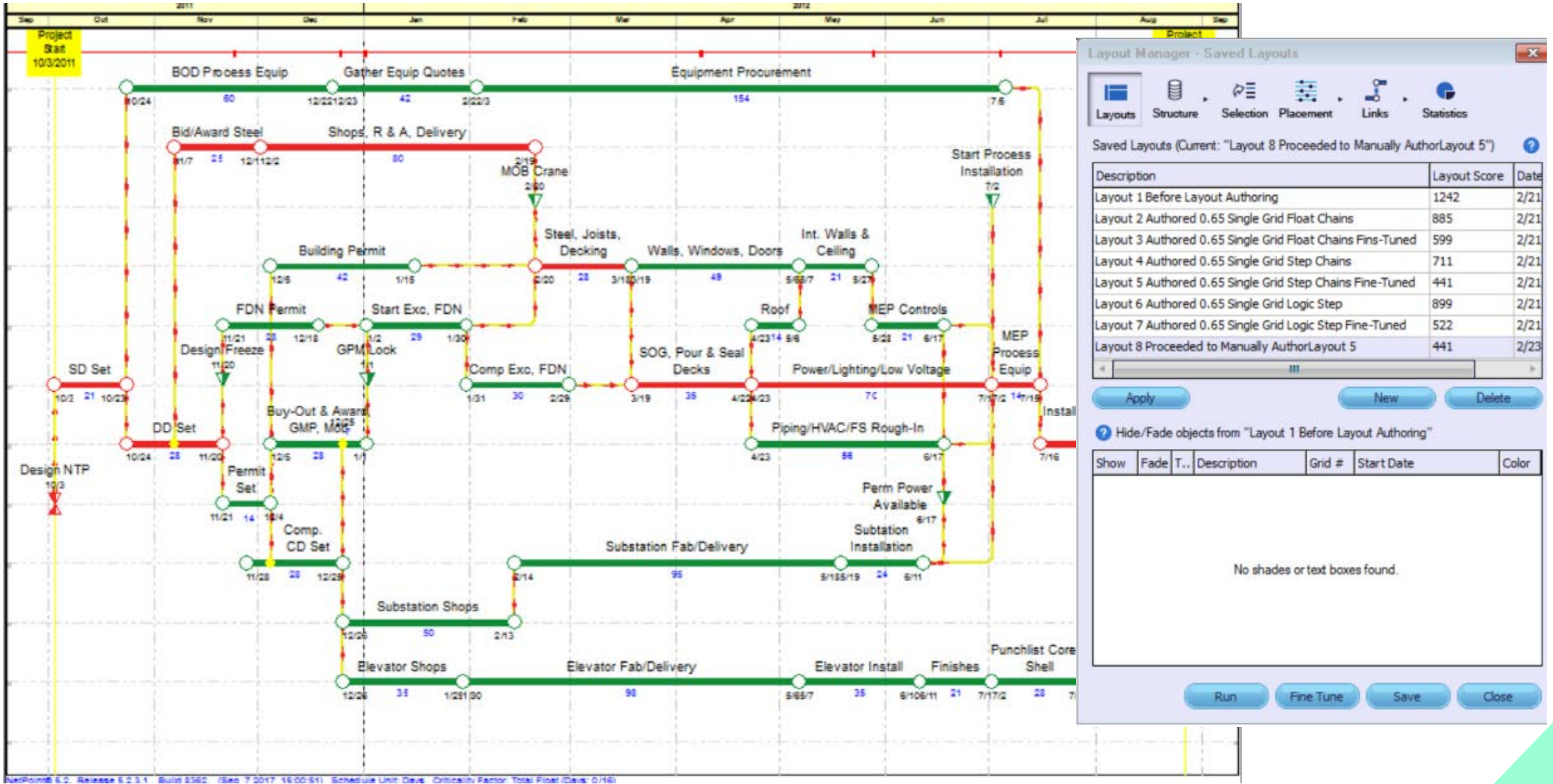




Algorithmically-Authored Network Layout Step Chains + Fine-Tuning



I Could Only Tie Layout 5 441 Score When I Manually Tried to Improve Layout 5

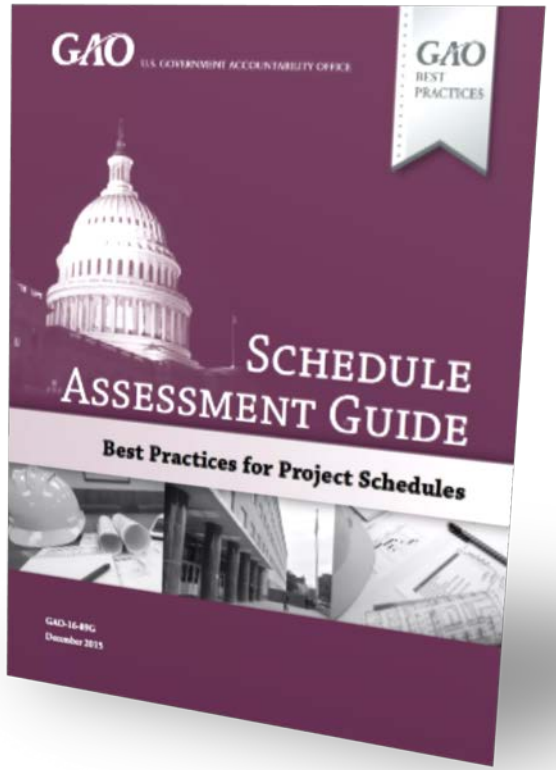


Redundant Logic Algorithm

Redundant logic represents unnecessary logic ties between activities

Redundant Link:

Activity-successor link that cannot unilaterally drive the successor under any combination of durations and lags, because the successor is always necessarily driven by another successor to the activity



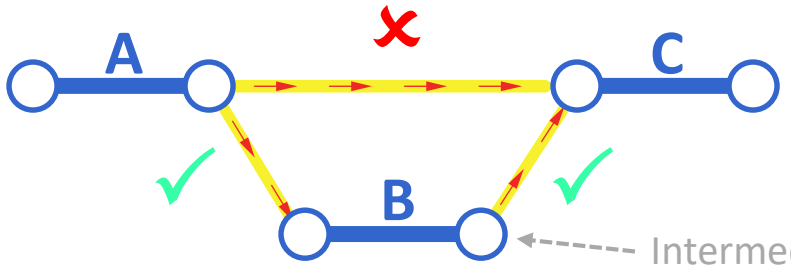
Quote from the GAO Guide (2015 Edition, p. 31)

"The network should be clear of redundant logic"

Redundant Logic Example

Redundant links cause fictive parallel relationships or path convergence and needlessly obfuscate the display of the schedule

EXAMPLE



The classic and indeed most obvious case of redundant logic comprises Activity A preceding Activity B on FS logic, Activity B preceding Activity C on FS logic, making–if included–an FS link between Activity A and Activity C unnecessary and thus redundant

The Redundancy Conundrum, Activity-to-Activity Links

Algorithms to detect and eliminate redundant links are not trivial in a PDM network with 4 relationship types

12 out of 64 possible logic patterns in a PDM network with 4 relationship types are redundant*

NetPoint®

- **12** are redundant
- **52** are valid

** If SF logic is excluded, out of 9 possible logic patterns, 3 are redundant*

Deltek | ACUMEN

- Circa 2015 version
- Finds redundancy in **only 3** of the true **12** redundant logic patterns
- Finds redundancy in **13** valid logic patterns

CPM vs. GPM Algorithms for Randomized Schedule Solutions

53 Years lost in the Schedule Simulation Weeds

In **1963**, Van Slyke introduced a CPM simulation method to provide a schedule randomized solution that corrects for the PERT underestimation from merge bias

In **2016**, Ponce de Leon and Puri proved that CPM simulation underestimates completion risk and introduced a simulation that relies on statistical sampling the as-built schedule

Van Slyke's sampling method applies the CPM forward & backward pass in every iteration, thus ignoring the fact that early dates are unrealistic execution dates

*GPM risk algorithms determine early **and** late completion distributions, and predict safe float−extent of total float that may be used and not delay the targeted p-value date*

Van Slyke's Randomized Schedule Solution circa 1963

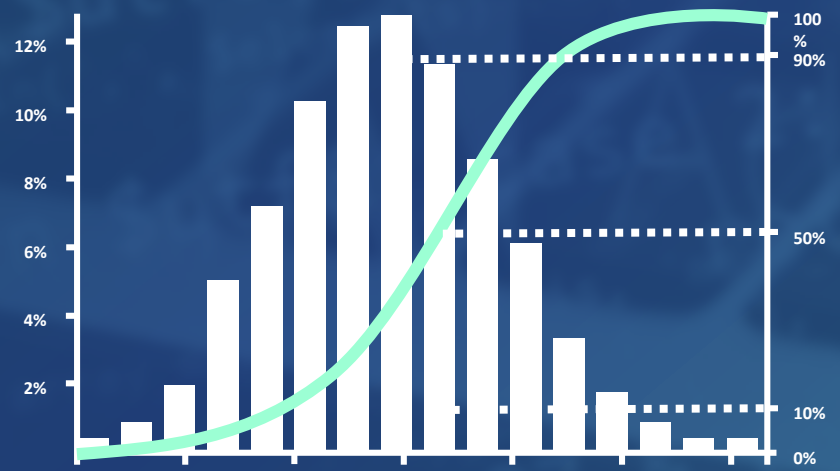


Benefits Over PERT

- Uses generic distributions
- Corrects for the PERT merge bias by creating multiple *early-schedule* occurrences of the project schedule
- Aggregates results from sampled occurrences to compute mean and variance
- Ability to model risks and correlations

Early-Dates Bias in CPM Schedule Risk Analysis

By scheduling activities on early starts in every iteration, CPM risk analysis (Van Slyke's method) does not include the impact of delayed and late starts (impact of float use) on completion:

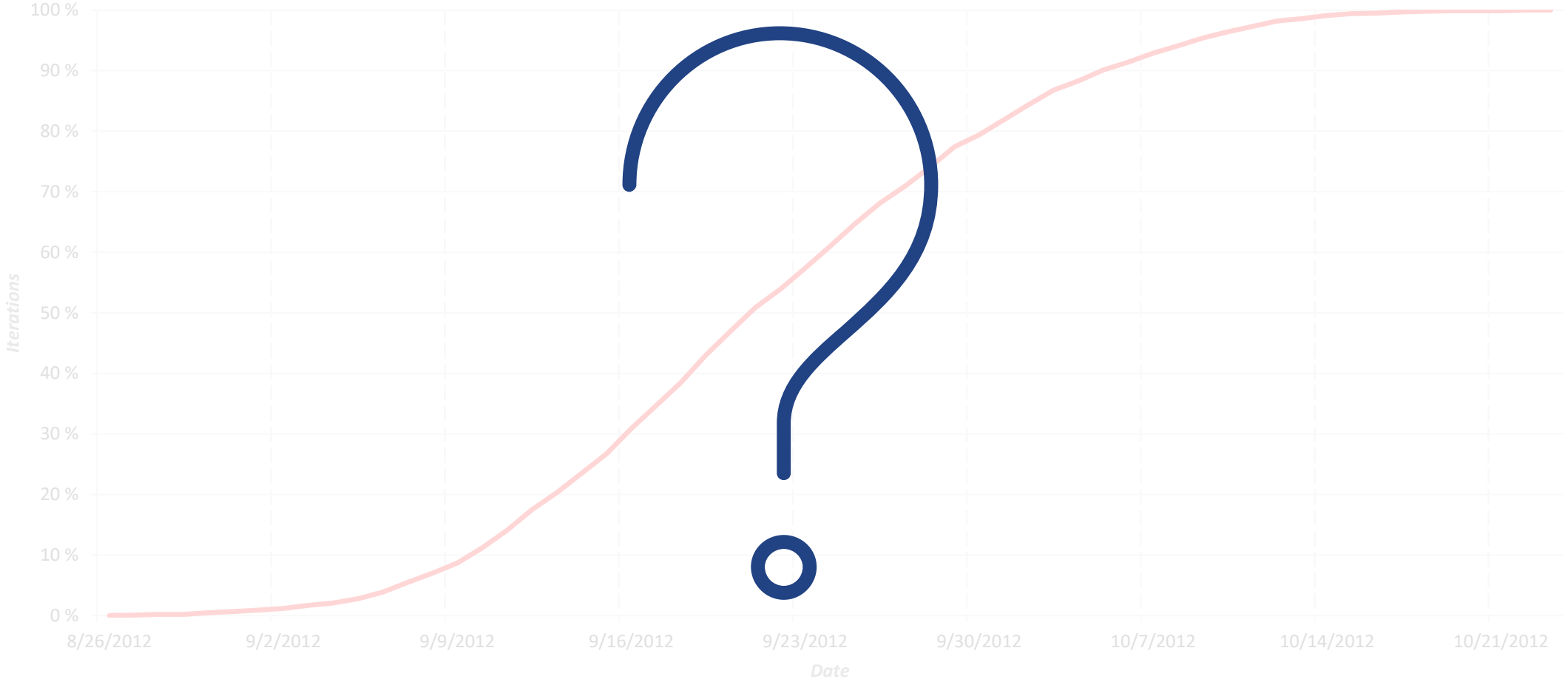


Van Slyke's method

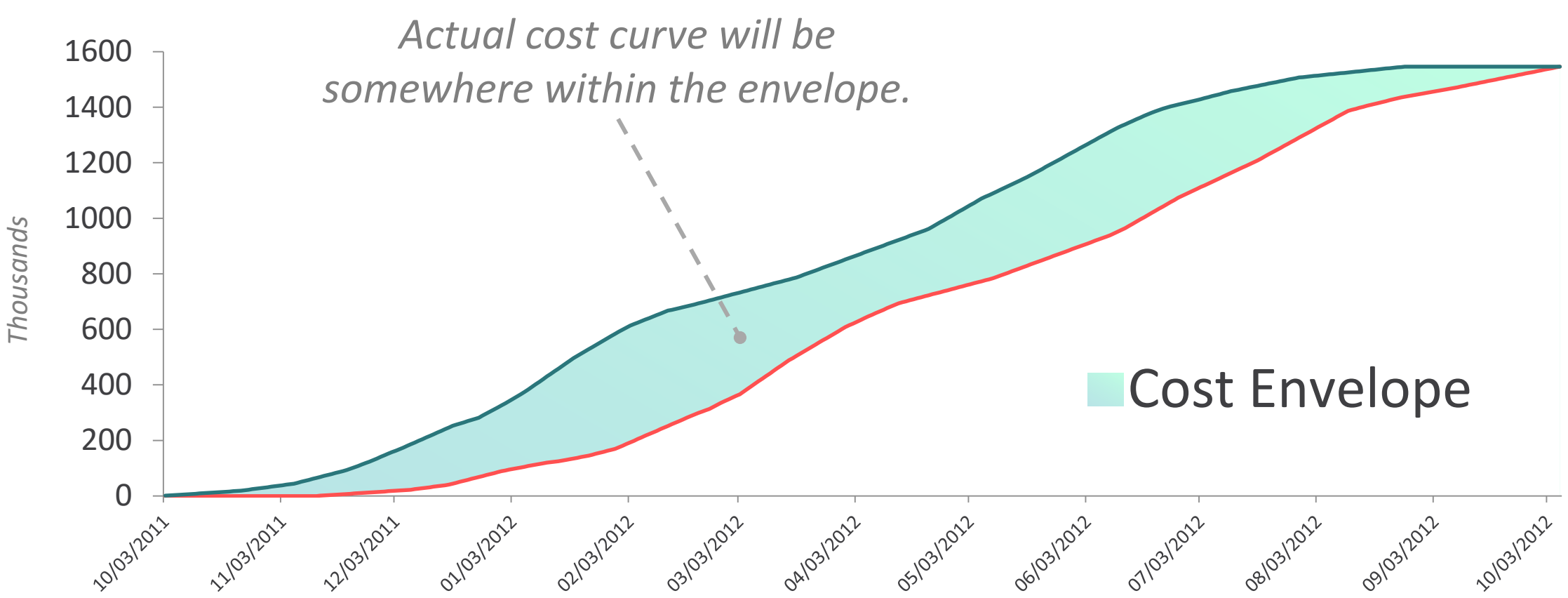
The early-dates limitation is a deviation from the actual system modeled; on actual projects, activities off the critical are often *floated* and start later, making use of total float

By not modeling *floating risk* in any iteration, CPM risk analysis merely develops the early completion risk function because decisions to use float have a likelihood of causing critical path delay and affect the project completion date

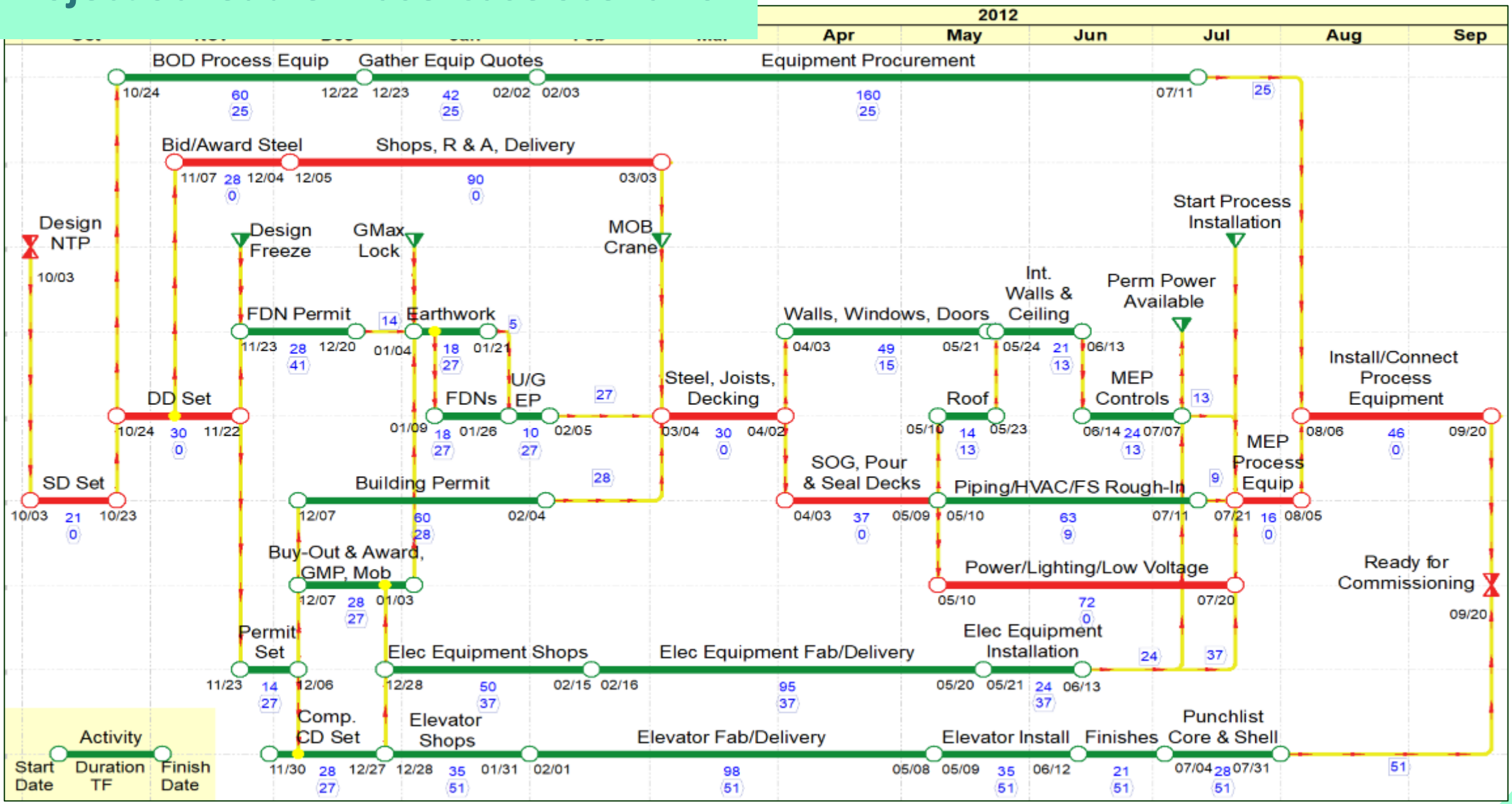
CPM Schedule Risk Analysis Yields One Completion Distribution Curve



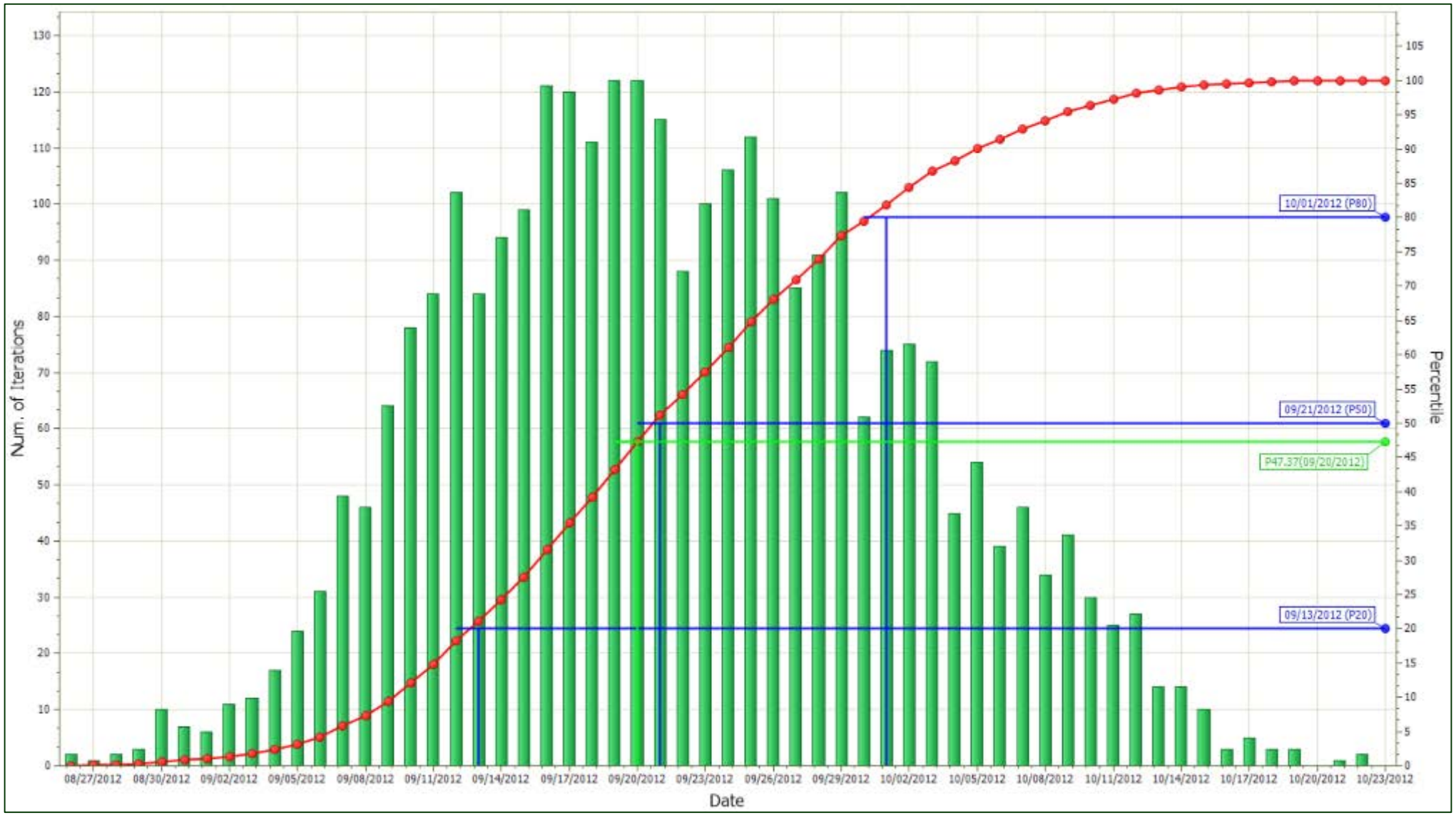
Conventional Network Analysis Early and Late Curves



Project Schedule–Base-Case Scenario



CPM Simulation aka Early-Dates Scenario *Completion Distribution*



GPM Schedule Simulation Algorithms

Core Tenet #1

Each iteration begins with the base-case scenario and unfolds as a simulated update of the base-case scenario activity by activity according to network logic

- Each iteration is a simulated progression of time rather than a batch replacement of activity duration

Core Tenet #2

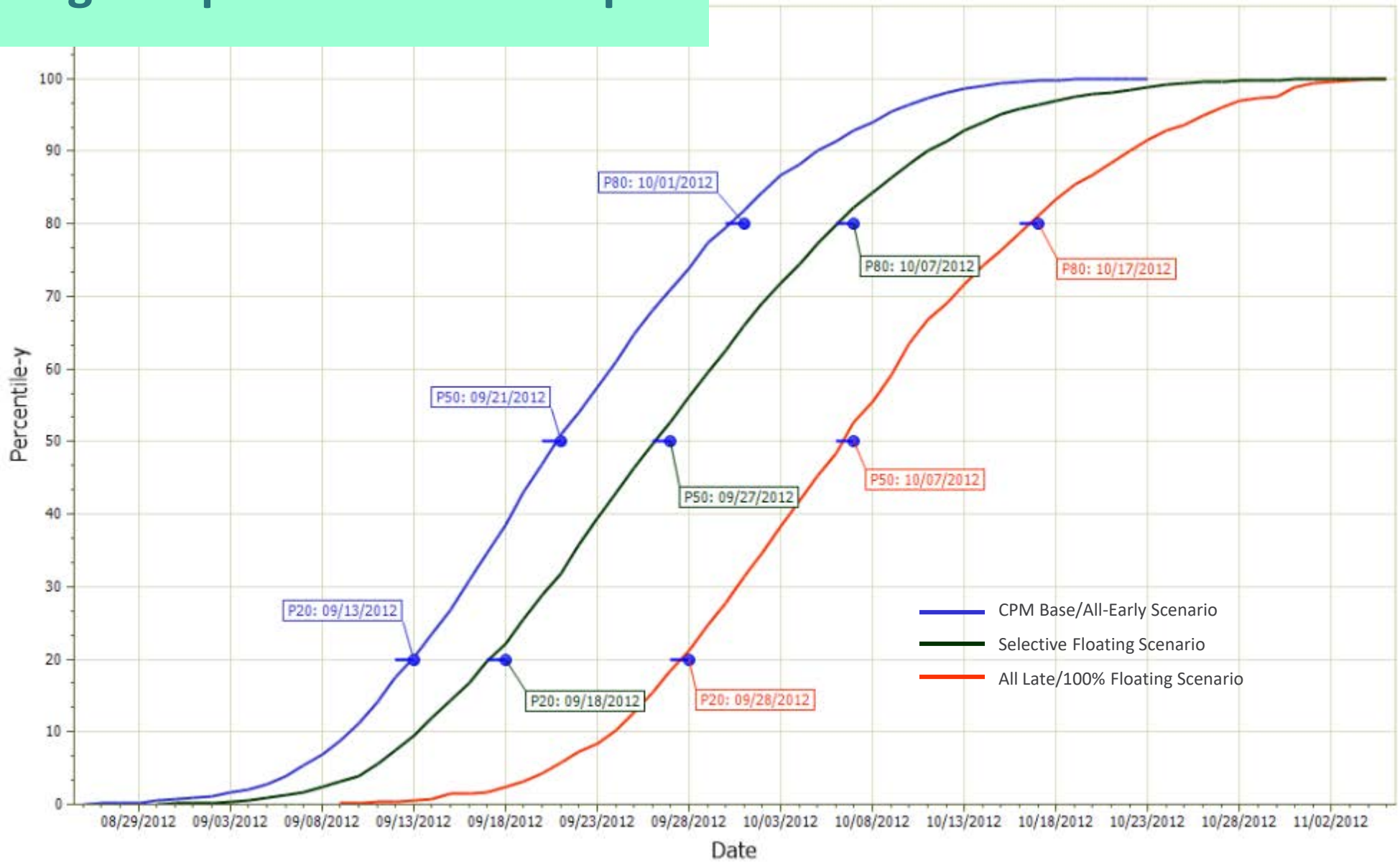
In each iteration, activities may be scheduled on GPM planned dates, either because

- GPM planned dates are present in the base case, or
- Positive-float activities, through sampling, are allowed to float or pace by using a portion or all of then-available float

Core Tenet #3

In each iteration, the longest path has the least total float; late finish date of the last activity on the longest path equals its early finish date or the project finish date

Bounding Completion Risk Envelope



Safe Float Relative to Safe Completion

Extent an activity can be delayed from base-case early start date without delaying the targeted (p-value) project completion date aka safe completion

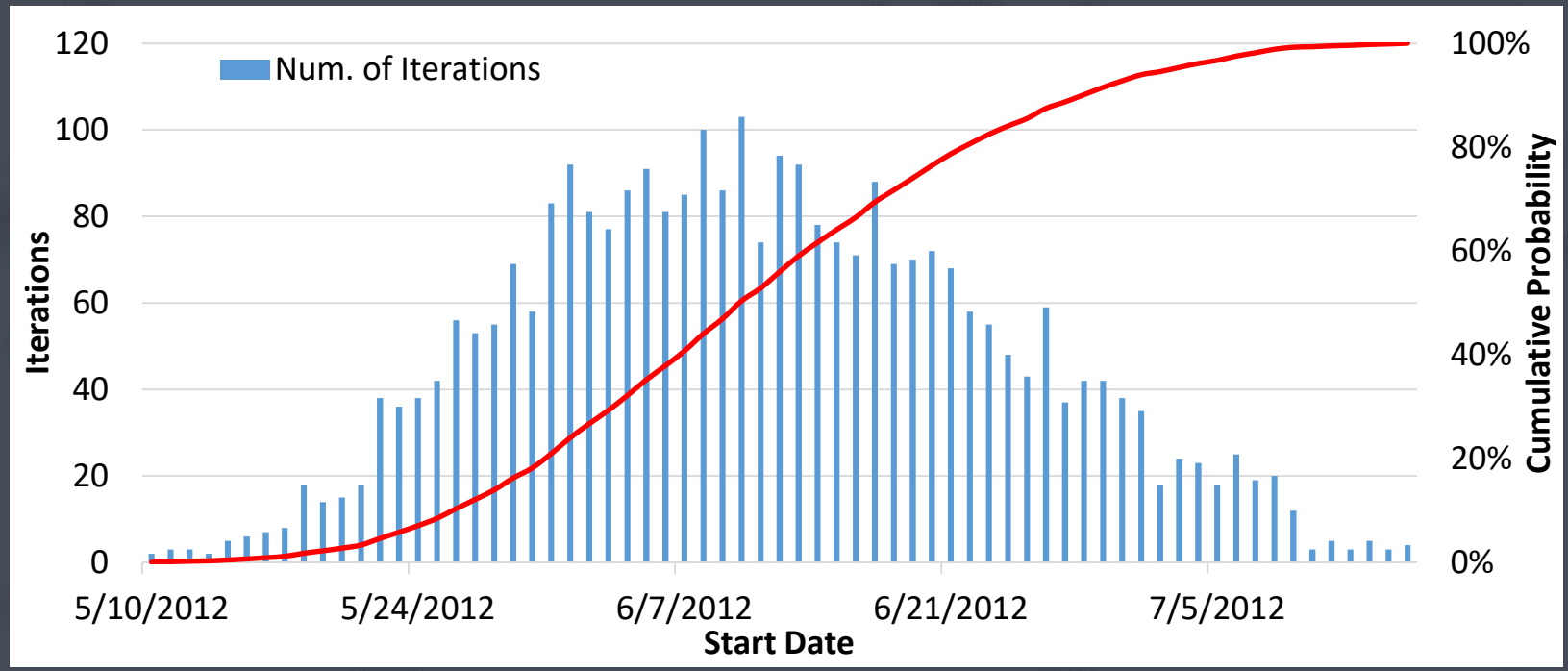
Dependent on the modeled uncertainty

Illustrating approach for calculating safe float for activity “Elec. Equipment Installation”

Earliest Stochastic Start: 05/10/2012

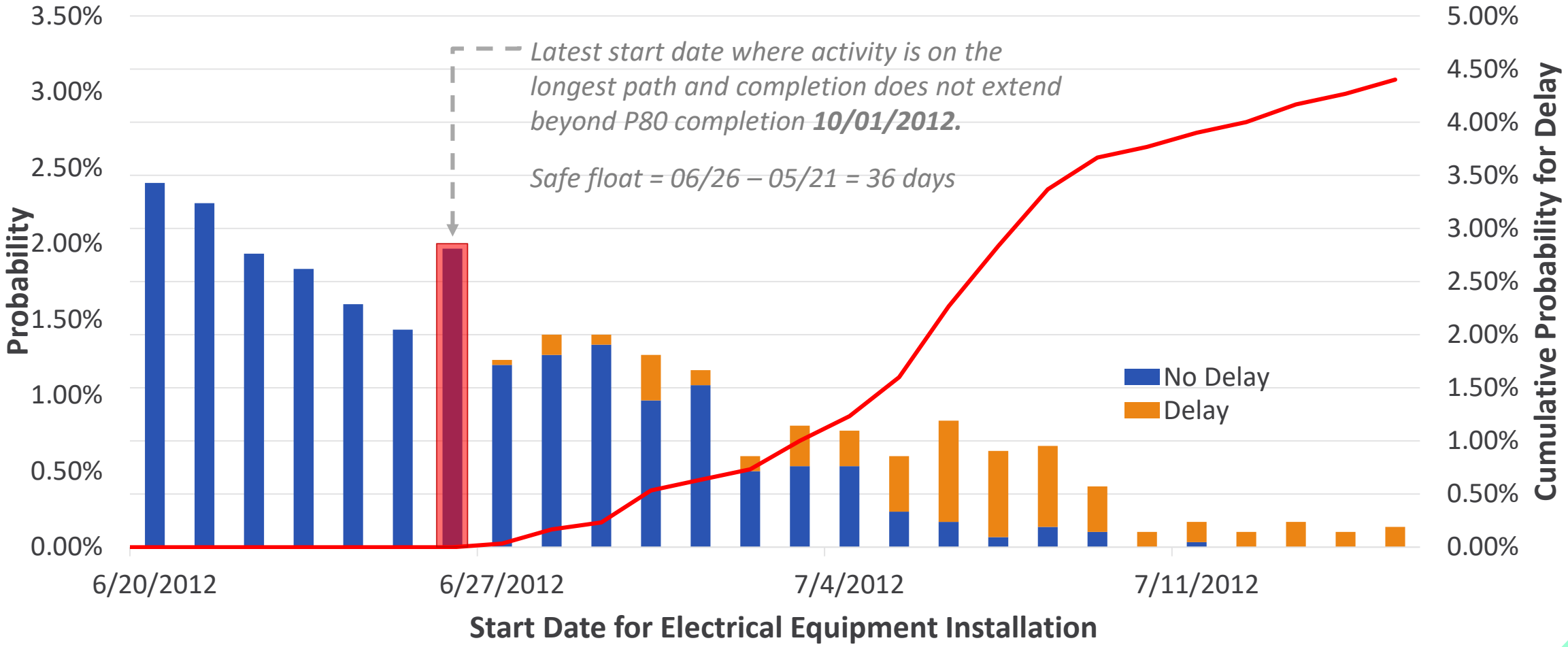
Base-Case Start: 05/21/2012

Latest Stochastic Start: 07/15/2012

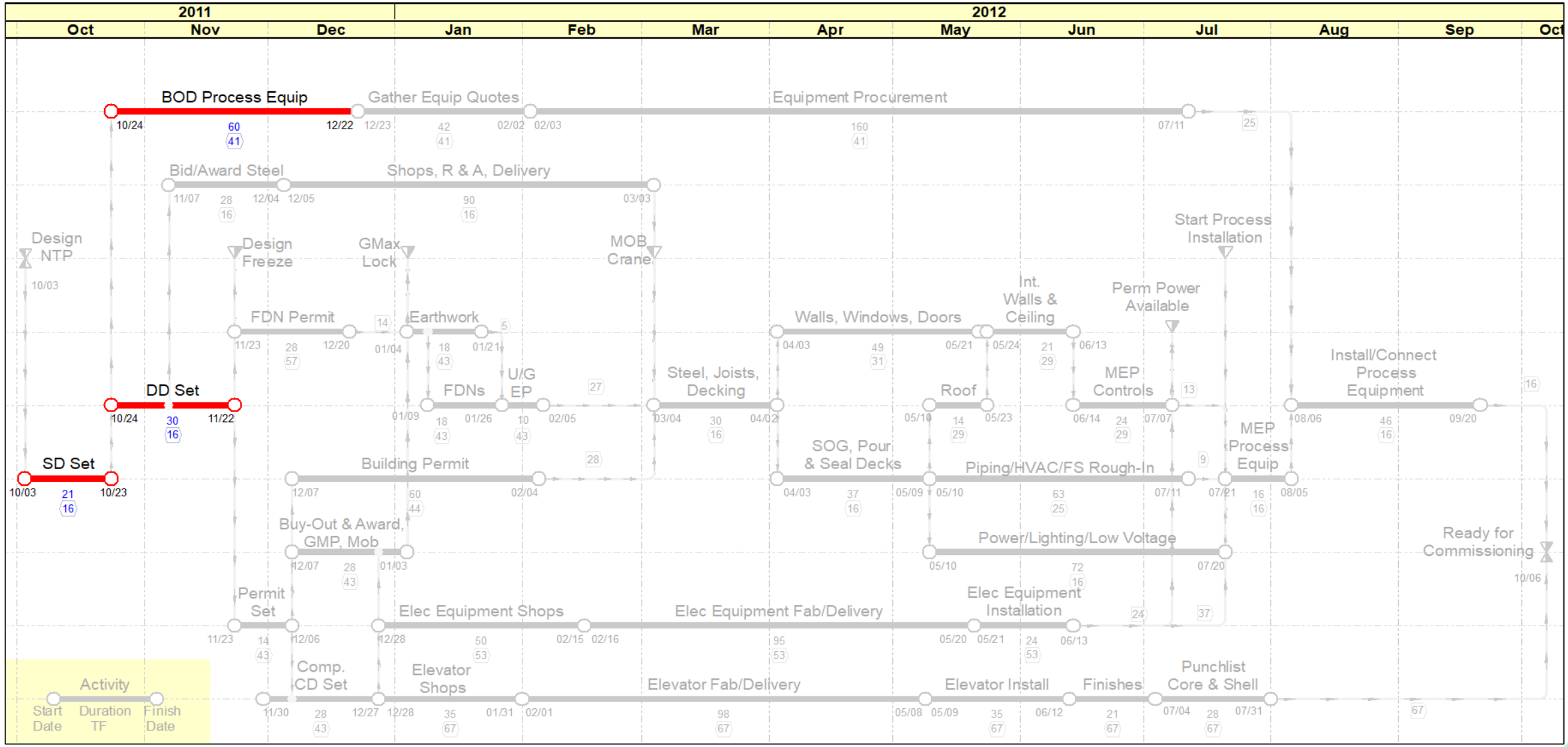


Safe Float Use – Randomized Approach

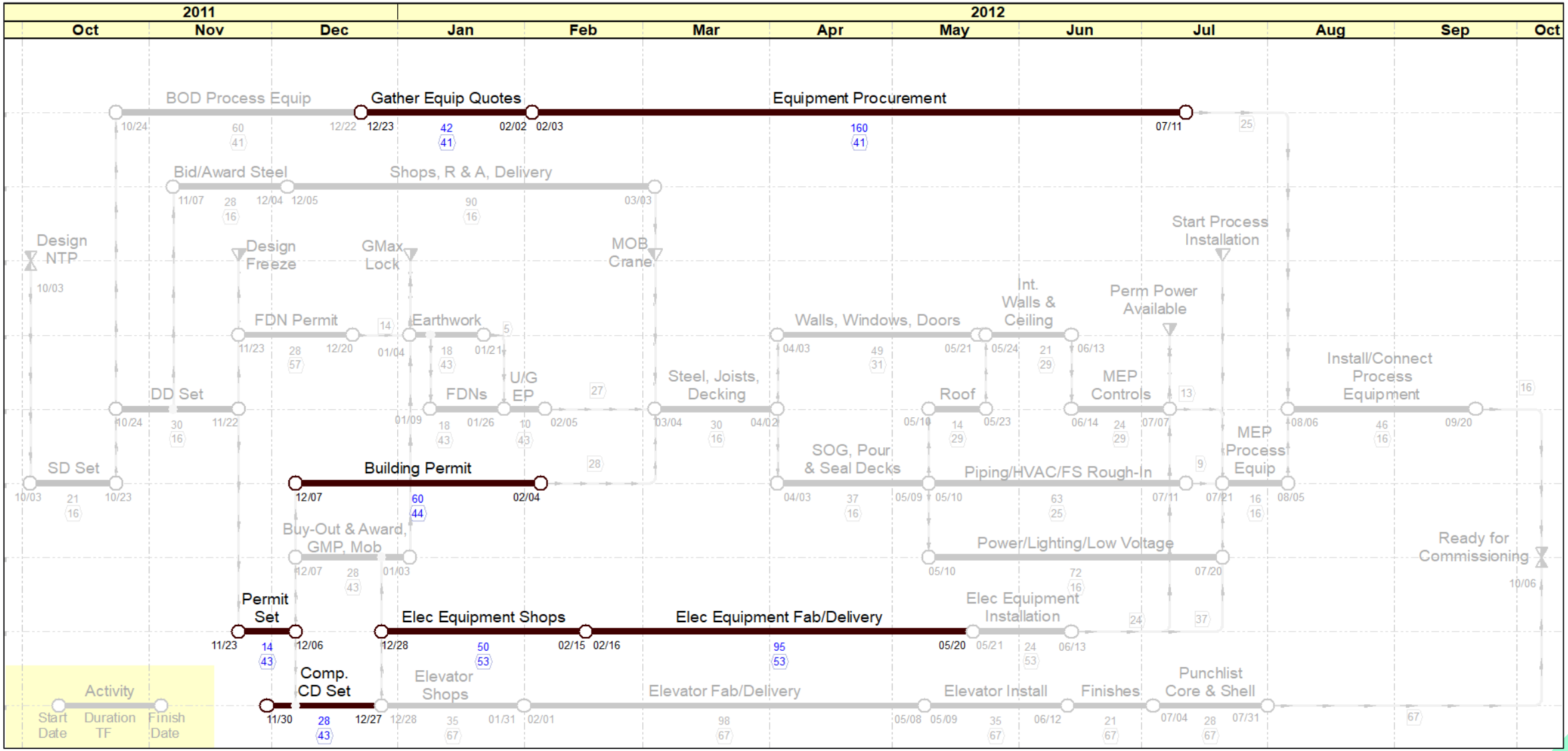
Simulation results analyzed to identify iterations where activity is critical and schedule extends beyond targeted completion date.



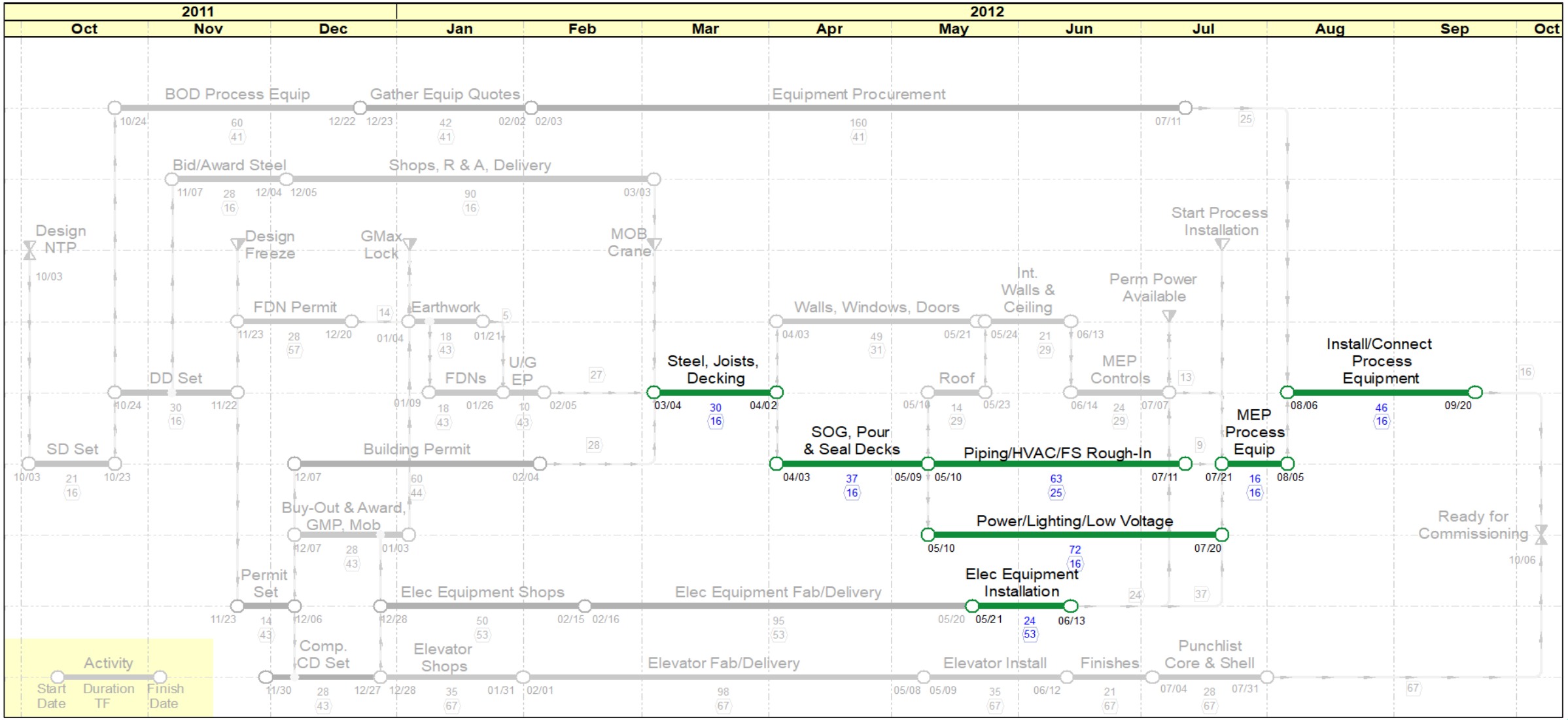
Unsafe Activities–No Possible Start After Stochastic Earliest Start Date



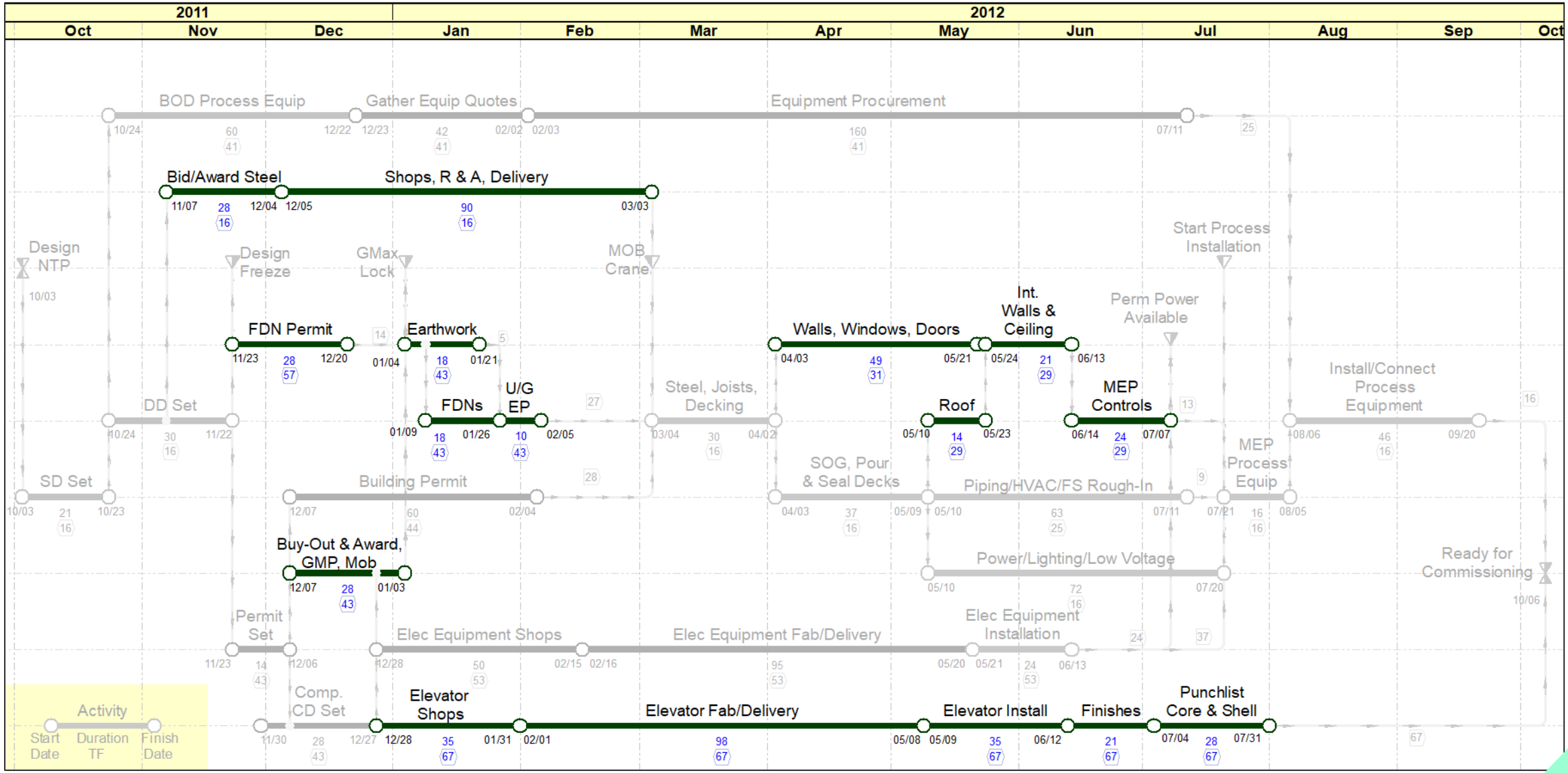
Unsafe Activities–No Possible Start After Base-Case Early Start Date



Partially Safe Activities–Safe Float > 0 Within Modeled Uncertainty



Totally Safe Activities–Free to Use Available Total Float



Safe Float Values, One Unsafe Activity and 7 Partially Safe Activities

Activity Description	Criticality Index	Base-Case Start Date	Safe-Case Start Date	Total Float*	Safe Float
Elec Equipment Fab/Delivery	0.249	2/16/2012	2/16/2012	53	0
Steel, Joists, Decking	0.537	3/4/2012	3/9/2012	16	5
SOG, Pour & Seal Decks	0.537	4/3/2012	4/7/2012	16	4
Power/Lighting/Low Voltage	0.441	5/10/2012	5/14/2012	16	4
Piping/HVAC/FS Rough-In	0.131	5/10/2012	5/16/2012	25	6
MEP Process Equip	0.768	7/21/2012	7/27/2012	16	6
Install/Connect Process Equipment	1.000	8/6/2012	8/13/2012	16	7
Elec Equipment Installation	0.249	5/21/2012	6/26/2012	53	36

*Base-case project completion date 10/06/12

(Partial List) of NetPoint® Algorithms

GPM	***Objectbase*: Data rules and computational algorithms are encapsulated within the network objects**
Gap and Total float	Total float/float (planned float))/drift (backward float) derived from link gaps vs. late & early dates
Crashing/Extension	NetPoint's crashing and extension algorithm in conjunction with planned dates and gaps
Scheduling - Self-Healing	Algorithm that calculates gaps and floats WHILE the user is disrupting the current state (synchronously)
Planned Dates	Planned dates > early dates generate drift > 0 without reducing total float
Backward Planning	Pull planning, building the network and schedule backwards from the completion milestone
Forensic Total Float	Total float is calculated--and the critical path is established--left of the data date
Resource Leveling	Manual and automated leveling heuristics that uses histogram moment as criteria (under development)
Logic Diagramming Method (LDM)	Unique relationship representation using embedded nodes as objects
Redundant Links	Identify links that are redundant
Automated Network Layout	Interactive, constraint-based algorithms for improved layout of activities on the canvass grid
Global Layout	Suitable for level 1 schedules where activity chains naturally organize into a swim lane layout
Swim Lane Layout	Algorithm that visually organizes network objects into horizontal lanes (de rigueur in level 3 & 4 schedules)
Weather Mode	Algorithms that greatly facilitate generating weather-impacted schedules
Weather-Profile Weather Days	Calculates days lost due to selected weather conditions using selected weather station(s) and time horizon
Non-Working Day Algorithm	Auto-generates weather calendars by marking weather days as non-working
Weathered Duration Algorithm	Auto-extends activity durations by impacting them for weather days between start and finish dates
Visual Target & Variances	Development of the target and variance display to aid in what-if scenarios

(Partial List) of NetRisk™ and Schedule MD™ Algorithms

NetRisk

Progress Simulation	Simulation that progresses the schedule to build thousands of iterations of the as-built schedule
Longest Path	Uses minimum float predecessors to identify the longest path
Crashing/Extension	Extends by eating up gap and crashes by holding zero gaps and increasing non-zero gaps
Floating/Pacing	Modeling of floating and pacing scenarios in each iteration
Network Display with p-Durations	Display network using activity durations at selected percentiles and overlay other information
Safe Float	Predicting the amount of total float an activity can use without affecting p-value targeted completion
Project Completion Distribution	Development of bounding completion risk envelope (early-dates and late-dates distributions) with planned dates
Priority Index	Prioritizing activities using criticality and cruciality

Schedule MD

IQ Score	Schedule analytics calculation based on the 20-Trait Protocol
Merge Float and Index	Indicates whether activity has adequate float to mitigate delays in start due to earlier overruns
Schedule Gap Index	Sum of link gaps divided by remaining non-critical durations

In Conclusion:

- Planners are increasingly opting for transparency and collaboration in scheduling
- Graphic & visualization apps are intrinsically transparent and facilitate collaboration
- GPM algorithms make it easier to use schedule graphical & visualization apps
- CPM algorithms are inapt for graphical & visualization scheduling software platforms
- GPM planners that are more savvy about GPM algorithms get more out of GPM apps
- Algorithm-rich modeling is de rigueur in modern network-based planning/scheduling

The self-healing capability in GPM algorithms provides a more cognitively responsive platform for project planning and project team collaboration among multiple users and organizations, where testing what-if scenarios is crucial to achieving project cost/time savings

"I have wondered how, given the diverging routes that planners, schedulers, project managers, owners, and educators are taking, we will ever again bring to projects the kind of synergistic creativity known in the early days."

James J. O'Brien, Foreword to *Faster Construction Projects with CPM Scheduling*, by Murray B. Woolf. (New York: McGraw-Hill, 2007).

Sent: Wednesday, January 23, 2008 7:27 PM
To: Gui Ponce de Leon
Subject: Re: Peer Review of Your Paper for COS 2008

Gui,

I am pleased that our thoughts were on parallel tracks. As to the "rise and fall of CPM" please handle it either way that works for you.

I will admit that the easily said phrase surprised me as I said it. I really believe that optimism for your GPM let me say the unthinkable.

A quote is fine, but giving credit to both of us would also be fine.

To me, the loss of the logic diagram has been the unrecognized tragedy in the evolution of CPM scheduling and your GPM brings it back full circle.

As to the additional GPM treatises, it was obvious that more than two would be necessary and I believe that the list would best come from you who has thought it through.

I agree that LDM would be an excellent term to replace the existing terms ADM and PDM with the best of both of them.

Regards,

Jim

Bibliography

Critical Path Method

Archibald, R.D. and Villoria, R.L. Network-Based Management Systems (PERT/CPM). New York: John Wiley & Sons, Inc., 1968.

Fondahl, J.W. *A Non-Computer Approach to the Critical Path Method for the Construction Industry*, 2nd ed. Palo Alto: Stanford University, 1962.

Ford, L.R. and Fulkerson, D.R. *Flows in Networks*. Princeton: Princeton University Press, 1962.

Kelley, Jr. J. and Walker, M. "Critical Path Planning and Scheduling." *Proceedings of the Eastern Joint Computer Conference: Boston*, 1959.

Kelley, Jr. J. "Critical-Path Planning and Scheduling: Mathematical Basis." *Operations Research* Vol. 9, No. 3, 1961.

Bibliography

Critical Path Method *(cont'd)*

Kelley, Jr. J. and Walker, M. "The Origins of CPM: A Personal History." *PM Network*, 1989.

Kelly, P.M., Nagata, M., Sanders, M.C. and Thorndike, C. "Is CPM Past Its Prime?" Cost Engineering January/February, 2014.

Korman, R. "Critics Can't Find the Logic in Many of Today's CPM Schedules." Engineering News-Record, 26 May 03.

Lamb, E. "How to Fix a Broken Scheduling System." *Engineering News-Record*, 30 Sep 13.

O'Brien, J.J. and Plotnick, F.L. *CPM in Construction Management*. New York: McGraw-Hill, 1965.

O'Brien, J.J. and Plotnick, F.L. *CPM in Construction Management,* 5th ed. New York: McGraw-Hill, 1999.

Bibliography

Critical Path Method *(cont'd)*

Ponce de Leon, G. "Precedence Network-Based CPM – Scheduling Primer: a Monograph." Ann Arbor*:* Townsend & Bottum, 1970.

Ponce de Leon, G. "Extensions to the Solution of Precedence Networks." PhD Dissertation*,* Civil Engineering Department*,* The University of Michigan*,* 1972.

Ponce de Leon, G. and Kedia, S. "Looping Relationships in Precedence Networks." *Theory and Application of Scheduling, Conference Proceedings,* 1976*.*

US Government Accountability Office. *GAO Schedule Assessment Guide.* Washington, DC: Author, 2012.

Woolf, M. *Faster Construction Projects with CPM Scheduling*. New York: McGraw-Hill, 2007.

Bibliography

Graphical Path Method

Ballard, G. *Last Planner™ System of Production Control*. UK: University of Birmingham, 2000.

Kahneman, D. *Thinking, Fast and Slow*. New York: Farrar, Straus & Giroux, 2011.

Mather T. and Liu, L. "An Algorithmic Dissection of Graphical Path Method (GPM®): Float, Drift and Total Float in Comparison to CPM Total Float." *PMICOS Conference Proceedings,* 2011.

Ponce de Leon, G. "Graphical Planning Method (A New Network-Based Scheduling Paradigm)." *PMI College of Scheduling Conference Proceedings,* 2008.

Ponce de Leon, G. "Project Planning Using Logic Diagramming Method." *AACE International Transactions*, 2008.

Ponce de Leon, G. "GPM®: An Objectbase Project Networking Method." *PMI College of Scheduling Conference Proceedings*, 2009.

Bibliography

Graphical Path Method (*cont'd*)

Ponce de Leon, G. "Graphical Planning Method." *Construction Management Association of America*, 2009.

Ponce de Leon, G. et al. *Guide to the Forensic Scheduling Body of Knowledge–Part 1*. Ann Arbor: PMA Consultants, LLC, 2010.

Ponce deLeon, G. "GPM® and Forensic Total Float." *PMI College of Scheduling Conference, Calgary, CAN,* 2010.

Ponce de Leon, G. "Scheduling, Fast & Slow." *2nd NetPoint User Conference*, 2012.

Ponce de Leon, G. "Logic Gantt Chart Requiescat in Pace." *Ann Arbor, MI: PMA Consultants LLC,* 2013.

Ponce de Leon, G. "Mitigating the Planning Fallacy." *3rd Annual GPM & NetPoint User Conference*, 2013.

Bibliography

Graphical Path Method (*cont'd*)

Ponce de Leon, G. *Core Traits of a Reliable Schedule*. Ann Arbor: PMA Consultants, LLC, 2014.

Ponce de Leon, G. and Puri, V. "Delay Principles Based on Completion Risk." *AACE International Transactions*, 2015.

Ponce de Leon, G. "Obsoleting 60 Years of Scheduling Calculus." *3er Congreso de Ingenieria de Costos, Lima*, 2015.

Ponce de Leon, G. "In Pursuit of Reliable Schedules." *6th Annual GPM & NetPoint User Conference,* 2016.

Ponce de Leon, G. "Automate This: Solving the Weather Conundrum." *NetPoint and GPM Conference--Nashville*. Ann Arbor: PMA Consultants, LLC, 2017.

Steiner, C. *Automate This: How Algorithms Came to Rule Our World*. New York: Penguin Group, 2012.

Bibliography

Diagram Layout Authoring

Bohringer, K.F. and Paulisch, F.N. “Using Constraints to Achieve Stability in Automatic Graph Layout Algorithms.” *Proceedings of the CHI Conference on Human Factors in Computing Systems*, 1990.

Di Battista, G., Tamassia, R., Eades, P., and Tollis, I.G. “Algorithms for Drawing Graphs: An Annotated Bibliography.” *Computational Geometry*, Vol. 4, Issue 5, October, 1994.

Dwyer, T., Marriott, K., and Wybrow, M. “Topology Preserving Constrained Graph Layout.” *Proceedings of the Revised Papers From International Symposium on Graph Drawing*, 2008.

Dwyer, T., Marriott, K., and Wybrow, M. “Dunnart: A Constraint-Based Network Diagram Authoring Tool.” *Proceedings of the Revised Papers From International Symposium on Graph Drawing*, 2008.

Zhao, W., Wang, W., Lu, X. “Auto-Layout Algorithm of Project Network Diagram.” *Proceedings of the IEEE Computer Society Workshop on Digital Media and Digital Content Management*, 2011.

Bibliography

Randomized Schedule Solution

AACE International. *RP 57R-09 Integrated Cost and Schedule Risk Analysis Using Monte Carlo Simulation of a CPM Model*. Morgantown: Author, 2011.

AACE International. *CPM Schedule Risk Modeling and Analysis: Special Considerations*. Morgantown: Author, 2012.

Budd, C.I. & Budd, C.S. *Leveraging the Power of Schedule Margin*. Morgantown: AACE International, 2012.

Hulett, D. *Practical Schedule Risk Analysis*. Burlington: Gower Publishing Company, 2008.

MacCrimmon, K.R. and Ryavec, C.A., "An Analytical Study of the PERT Assumptions." *Operations Research*, Vol. 12, No. 1, 1964.

Bibliography

Randomized Schedule Solution (*cont'd*)

Malcolm, D.G. Roseboom, J.H., Clark, C.E. and Fazar, W. "Application of a Technique for Research and Development Program Evaluation." *Operations Research* Vol. 7, No. 5, 1959.

PERT, Program Evaluation Research Task, Phase I Summary Report. Washington, DC: Special Projects Office, Bureau of Ordinance, Department of The Navy, July, 1958.

Ponce de Leon, G. and Schriber, T. "Determination of Criticality Indexes in the PERT Problem." *Third Conference on the Application of Simulation, 1969.*

Van Slyke, R.M. "Monte Carlo Methods and the PERT Problem." *Operations Research*, Vol. 11, No. 5, 1963.

Williams, T. "Criticality in Stochastic Networks." *J Oper Res Soc* 43: 353, 1992.

Q & A Time

Thank you – Any Questions?





THE LONE STAR STATE

NetPoint+GPM CONFERENCE

AUSTIN, TX | MARCH 1 & 2, 2018



Gui Ponce de Leon, PhD, PE, PMP

Chief Executive Officer, PMA Consultants LLC

gui@pmaconsultants.com