Preface

Why are we writing, and why do our readers need an 8th edition? After our first seven editions, what more may be said about CPM planning and scheduling for the construction and other time-centric industries?

In the 7th edition, we said the mighty ship *USS Scheduling* had come about 180 degrees. Now it is continuing to a full 360-degree circle, perhaps as in the *Caine Mutiny* cutting her own tow line. In its earliest days, limited computing power required many work-arounds and dumbing down of the mathematics behind solving critical path method network logic. Only the "priests" of CPM could understand how to elicit the necessary information from the project team, modify to stay within the limitations of the software, navigate the software, and finally interpret the results.

Some of the priests became enamored of their computers and forgot to consider the needs of their parishioners. Project managers in the field did not find useful these reports that were run by "computer specialists" from headquarters and preferred the use of their "old-fashioned" bar charts. The official CPM prepared of, by, and for these "computer specialists" was submitted, reviewed, and otherwise totally ignored. Unless there was a threat of litigation—then suddenly it was considered an important historic document.

The advent of the personal computer untethered project managers from the priests and computer specialists at headquarters. A plethora of software startups provided products that were simple to use and eminently suitable to the needs of parishioners—or end users—the line managers in charge of projects. In the heat of competition these software products were honed to provide maximum support at minimum effort to managers in responsible charge of bringing a project in within budget and on time.

A few software solutions moved to the forefront and prospered. Others remained for niche uses. All had the need to grow enough to support continued development to harness the growing power of micro-computers. And so those who could not maintain market share faded into the night. The best added capabilities requiring some special training without encumbering the average user of the basic system.

But a funny thing happened on the way to market dominance—the market and average user changed. The field of scheduling includes a variety of uses, only some of which first require careful and detailed planning. A student of operations research may first encounter scheduling to maximize the return on investment of machines in a factory. The goal may be to keep the machines working, perhaps 24/7, except for scheduled maintenance. Perhaps orders (projects) may have an order of priority, but there is rarely any priority in getting a specific order through the system as quickly as possible.

In some cases the focus may be on a machine of iron, in others on one of the flesh and blood of a human resource or computer code writer. In either case the scheduling algorithm differs from that used for construction and other time-centric scheduling—get maximum output from the fixed resource. Continuous use of a resource is more productive and the algorithm will deliberately defer the start of a task to attempt an uninterrupted flow of effort. If our machines are to pump out millions of widgets per month, or our team of code writers each may be assigned to development of a similar module of a dozen different software products, the speed at which the first widget or a selected software product is delivered is of little consequence.

Other forms of scheduling are even less attuned to time. PERT was developed for the research and development (R&D) effort required to launch the US Navy's Polaris missile. The Navy was not concerned with the best possible missile but rather primarily with the speed that the first working model could be developed. However R&D today, in the military and elsewhere, is often looking for the "best" product, and actively considering new features, benefits, and capabilities only first envisioned during the development process. Of course we cannot allow our researchers (or computer code writers) to go on forever, so we need to have a process to regroup and refocus on what we have learned and where we are going on a regular basis, while still promoting ongoing innovation. And thus the newest addition to our stable of scheduling procedures is called Agile scheduling.

Our frontrunner software product developers were not unaware of this burgeoning market. Attendance at the Primavera User Conference swelled from some 250 to approaching 2500, starting with one information technology (IT) track to eventually having only a few construction track sessions at an IT show. Development resources of all major players were devoted to new features desired by this new "average" user—one who is versed in a software code writer vocabulary and not one based on experience in construction. The standard algorithm from the original CPM of 1956, that the early start of an activity be the earliest time possible, was dropped from major software products in favor of that of "the earliest time to support continuous production." Logic network graphics were left to wither while bar charts were supplemented with pie charts, bubble charts, and tornado charts. The consolidation of PCs to a worldwide network (and the attendant need for network administrators setting standards above the level of a project) led

many project managers to leave the preparation of the official schedule to the professionals, while returning to field-prepared bar charts for actually running the project.

A perfect storm ensued in the late 2000s. Previously independent software product providers were acquired by IT companies, exacerbating the shift of product development of, by, and for IT users. A worldwide recession decimated the construction industry and further quieted the voice of those requiring continued support for time-centric scheduling. The continuing shift to the "cloud" or reverting independent and personal computers to being mere terminals further divorced construction professionals from "their" software and even from their data. Indeed, we had turned 180 degrees from the state of affairs of the 1960s and 1970s.

But we are writing and our readers need an 8th edition because our industry is reviving and our demands to meet our needs are starting to be heard. Construction CPM Conference (founded by author Plotnick—more details at www.constructioncpm.com) has replaced the previously shuttered User Conferences attended by our construction professionals. A plethora of new software products have been introduced (or reintroduced) with renewed emphasis on time-centric construction functionality. The major players are also rediscovering the construction market, and all are embracing how more powerful computers may be used to further support the construction industry.

The conference has become a forum for construction project managers and software product developers. A large number of sessions are devoted to new functionalities supporting construction project managers such as building information modeling (BIM) and schedule risk. Product developers, at the request of these authors, have reintroduced the interruptible duration algorithm necessary to speed completion of the project. Lost art (such as distinguishing a "start" restraint from a "begin" restraint) and new art, such as RDM (relationship diagramming method) functionality, are being actively discussed and the necessary critical masses required for funding development and implementation are being sought.

We dedicate this eighth edition to this effort and to the proposition that training in CPM logic networks and scheduling software to support solutions shall have a new birth of freedom—and that construction software of construction professionals, by construction professionals, for construction professionals, shall not perish from the earth.

James J. O'Brien, P.E., PMP
Fredric L. Plotnick, Ph.D., Esq., P.E.