TITLE: "MANAGING CRUNCH TIME"

by Tim Bryce
Managing Director
M. Bryce & Associates (MBA)
P.O. Box 1637
Palm Harbor, FL 34682-1637
United States
Tel: 727/786-4567
E-Mail: timb001@attglobal.net
WWW: http://www.phmainstreet.com/mba/
Since 1971: *"Software for the finest computer - the Mind"*

"All of your hard work, regardless of how well it is intended, is for naught if it results in a pile of rubbage."
- Bryce's Law

INTRODUCTION

Okay, you are under the gun to produce something by a given date; you do not have a lot of time for a robust methodology, nor are you interested in being encumbered with a lot of bureaucracy; you want to get the job done quickly and you want few problems; its "Crunch Time."

This dilemma is faced by software development organizations every day. You are required to move heaven and earth in a short period of time with minimal resources. How you found yourself in this predicament is irrelevant. You can point fingers later, but right now you have a deadline to meet. Now is not the time to lose your cool but, rather, work your way through the problem with a little creativity and a lot of resourcefulness.

What to do? What are the bare essentials for survival? There are seven points to be considered:

1. GET ORGANIZED

First, develop a project scope specifying EXACTLY what is to be accomplished. Determine the minimum to succeed, yet be ambitious enough to aim a little higher. Articulate and document the project scope as accurately as possible. This will be invaluable for conveying the project objectives to the project team.

Next, take stock of your strengths and weaknesses, particularly in terms of available human resources. Now is the time to recruit suitable personnel, either internally or outside contractors, to work on the project. If available, reference the company's skills inventory to locate suitable personnel to perform the work. If special equipment is required, order it now.

Since you may need to offer incentives to employees to motivate them, check with management to see what will be allowed, be it money, time-off, or some other perk.

Determine the working hours during the tenure of the project; for example:

• Will it be necessary to work overtime and on weekends?

• Will it be necessary to cancel vacations?

It is also not uncommon for managers to rent out hotel rooms close to the office to minimize distractions and keep the staff close to their project work.

Get this straight now so there is no confusion later on.

2. PLANNING

Now, more than ever, you have to do some planning. Whether you do it on your feet or on paper (the latter, of course, is preferred), you better get a battle-plan in place otherwise you will certainly fail.

First, prepare a rough design of the programs to be produced. This should require the participation of a senior programmer(s). For each program, consideration is given to:

• Inputs and outputs (with basic layouts).
• Files to be used (an overall data base layout).
• Data interfaces between programs (transaction files, temporary files, etc.)
• Basic processing logic.
• Method of implementation - recommended language, along with design tools and techniques. Also, consideration is given to packaged software.

Second, layout a plan and assign responsibilities. This is performed by defining a simple Work Breakdown Structure (WBS) reflecting the parts of the product to be built. For example:

PROGRAM-A
PROGRAM-B
PROGRAM-C
PROGRAM-D
etc.

Keep your WBS as simple as possible. Talk in terms of

(continued from page 1)

whole phases of work as opposed to detailed activities or tasks. As an aside, for software development projects, it is strongly recommended you separate programming phases from testing phases. Depending on the nature of the application, you may want to also establish a phase to develop the Data Base design. For more information on WBS, see:

http://www.phmainstreet.com/mba/pride/pm10.htm#wbs

When the WBS is defined, assign people to the various phases of work. Assign one person to each phase thereby delegating responsibility for the completion of the work to a specific worker. If it is necessary for more than one person to work on a given phase, assign a person to be primarily responsible for the phase. By delegating authority of a phase of work to an individual, you are empowering the worker and holding them accountable for their actions.

Inevitably, you will not have sufficient resources to do everything in parallel. Because of this, consider resource allocations and establish precedent relationships between phases of work. For more information on precedent relationships, see:

http://www.phmainstreet.com/mba/pride/pm10.htm#precedent

After the WBS has been defined, devise basic standards by which the work is going to be produced; e.g., design standards along with the acceptance criteria of the deliverable (form and content).

A couple of other planning considerations:

- Plan on sharing and reusing everything possible, be it design templates, code, data, etc. Now is not the time to fall prey to the "Not Invented Here" (NIH) syndrome whereby you will not consider alternatives as proposed by outsiders. Now is the time to be as resourceful as possible.

- Plan on running computer backups on a frequent and regular basis. The last thing you can afford now is "downtime" or losing the work of your employees.

3. CREATE A SENSE OF URGENCY

Your next concern is to create a sense of urgency among your staff. It is not sufficient that you alone (the manager) are on the hook for delivery, now is the time to kick your employees into high gear. Simple pep-talks won't hack it. You will need to do more.

First, have a kickoff meeting describing what has to be done and when. Describe the project scope in detail with the staff. They must have a crystal clear understanding of the work to be performed, how it will be performed, and the delivery date. Depending on the situation, describe any pertinent incentives for the employees, be it financial, time-off, job recognition, etc. Job security may be the best incentive of all. However, be wary not to be "doom and gloom" as employees may begin to bail out on you immediately.

Review the WBS, precedent relationships, and assignments. Then, as a group, layout a project schedule with start and end dates. Make sure your employees understand this is their commitment to the project which will be closely followed.

Both the WBS and schedule will require high visibility. To this end, it is recommended you post it in a place where everyone will be cognizant of their commitment. For example, I used to keep a magnetic board in the project team area where I posted a simple Gantt Chart listing the phases of work and the employees assigned to it. For information on Gantt Charts, see:

http://www.phmainstreet.com/mba/pride/pm30.htm#gantt

This chart raises the awareness of the impact each employee has on the project. The MagBoard is nice, but you can also accomplish the same feat using a simple blackboard or flipchart placed strategically in the office. I have also seen managers use such diagrams as the official desktop background on all PC's in a department. Further, it is essential that you, as manager, update the schedule as the project progresses.

Another item that can greatly assist in raising the sense of urgency is the implementation of a time reporting system whereby employees keep a daily record of the time expended on their work. This can be done either through simple PC software or through manual forms. For an easy to use time sheet, see:

http://www.phmainstreet.com/mba/pride/iw019.jpg

Very important, on the Time Sheet the employee should account for not only their project assignments but their interferences as well, such as: meetings, breaks, personal time, etc. From this data, the employee should calculate their "Effectiveness Rate" which is simply an analysis of the amount of time spent on direct work as

(continued from page 2)

opposed to interferences. For more information on "effectiveness rate,"see:

http://www.phmainstreet.com/mba/pride/pmmeth.htm#er

The time sheets should be collected once a week (on the first day of the new work week) and approved by the manager. This simple review materially assists in raising awareness of project responsibilities and promotes a sense of urgency.

4. SUPERVISE

You've heard me frequently say that normally you should *"manage more and supervise less."* Unfortunately, under a "Crunch Time" scenario it will be necessary for you to perform more supervision. Inevitably, a much more "hands on" approach will be necessary where you will actively work side-by-side with your staff.

Now is not the time for any surprises to pop-up and distract your mission. To this end, hold short meetings to review progress and report problem areas. I used to call for an early morning meeting prior to the start of the work day to wake everyone up and get them thinking of their assignments right away. From this meeting, I would also develop a punchlist and action plan to tackle technical problems encountered by the staff. Such meetings should be as brief as possible and to the point.

As supervisor, it is critical you control the staff's operating environment. In particular, you want to minimize distractions and interferences, such as irrelevant telephone calls and e-mails. A good secretary can work wonders for monitoring such activities and tracking the whereabouts of the staff. You might also want to consider having lunch brought in to minimize employee "down time."

Since the staff will be under considerable pressure to produce, look for ways to lighten things up, such as background music and loosening up on dress codes. Such techniques, as simple as they may sound, helps to release pressure.

Carefully study each employee's time sheet/screen, paying particular attention to indirect time (interferences). Where necessary, take action to minimize such distractions. Also, consider the amount of time remaining on a task and recalculate schedules as needed.

5. MANAGE THE CRITICAL PATH

During the project it is incredibly important that both the

manager and the staff STAY FOCUSED on their objectives. This is accomplished by constantly monitoring and updating the critical path of the project. As supervisor, your attention will jump from one part of the project to another as the critical path changes. Also, updating the project schedule (such as with a MagBoard or whatever) is an effective means for communicating to the staff as to what has been accomplished and changes in start and end dates.

It is also a good idea to appoint a senior programmer as your troubleshooter to take a SWAT-team approach for conquering technical problems. The staff should learn to turn to this person as a reference point to research problems and find solutions.

6. DELIVER A QUALITY PRODUCT

Due to the shortcuts you are taking in a "Crunch Time" project, in all likelihood you will not deliver the highest quality product as produced using traditional methods. Nonetheless, there is nothing more embarrassing than to produce something that will inevitably fail. All of your hard work, regardless of how well it is intended, is for naught if it results in a pile of rubbage. Test the hell out of everything. This is why I find it advantageous to separate programming phases from testing phases. True, the programmer should test and debug his/her individual program, but ideally, there should be a separate phase where an independent tester shakes the product out thoroughly. Depending on time commitments, this is a job well suited for the manager to perform. By doing so, it makes the manager more intimate with the nuances of the product and assures a level of consistency. When testing in this regard, you should first look for functionality (does the program do what it is intended to do?) and, second, consider the program's ease of use and intuitiveness (is it easy to learn and grasp?). DO NOT look for significant design revisions at this time unless it becomes blatantly obvious you have an unworkable design.

For more information on normal testing, see:
"PRIDE" Special Subject Bulletin No. 41 - ("Testing 1, 2, 3..." - Sep 12, 2005)
http://www.phmainstreet.com/mba/ss050912.pdf

Aside from testing, another technique to consider is "code reviews" whereby the programmer reviews the source code with a team of programmers. You would be surprised how much more carefully a programmer will write code if he knows his peers will be looking through their work. In addition to the basic design of the program and

(continued from page 3)

style of writing, the code reviews are used to determine the maintainability of the code and conformance to standards.

7. REVIEW

After the dust has finally settled and assuming you have satisfactorily delivered the product, you can take a breath. Now is the time to hold a postmortem on the project and determine what went right and what went wrong. Such analysis is invaluable for the next "Crunch Time" project which will inevitably come along, whether you or someone else is charged with implementing it. Also, add up the costs associated with the project and prepare a written report for review by management. When preparing this report, ask yourself, *"If I were to do it all over again, what would I do differently?"*

Finally, it is pay back time for any staff incentives promised at the start of the project. Always keep your promises. If you do not, the staff will certainly not forget it next time. At bear minimum, offer up some sort of celebratory party or luncheon to thank the staff for their hard work.

CONCLUSION

Key to the manager's success in "Crunch Time" projects is his ability to change directions on a dime. Your ability to supervise and control the project will be severely tested for, inevitably, if anything can go wrong, it most certainly will. Think of it as an endurance test. Something that will be watched closely by your superiors as well as your subordinates. Not only will you be evaluated in terms of being able to deliver a product in the required time frame, but you will also be evaluated in terms of how you handle adversary and remain cool under pressure.

Management by "Crunch Time" is no way to operate on a regular basis. Nobody wants to work under helter-skelter conditions with "quick and dirty" solutions for a prolonged period of time. By doing so, you run the risk of burning out your staff and yourself. You need to do a lot more than the above mentioned items to make your department run like a well oiled machine. Your project review should highlight this.

Its interesting. Americans tend to react better to crisis situations than perhaps anyone else on the planet. Our ability to perform under catastrophic conditions is legendary, be it Pearl Harbor, 9-11, or, more recently, Hurricane Katrina. When the chips are down and everyone knows the score, Americans are the most resilient when

it comes to responding to a challenge. Its in our character. As for forethought and planning, forget it.

END

"PRIDE" Special Subject Bulletins can be found at:

http://www.phmainstreet.com/mba/mbass.htm

They are also available through the "PRIDE Methodologies for IRM Discussion Group" at:

http://groups.yahoo.com/group/mbapride/

You are welcome to join this group if you are so inclined.

The "Management Visions" Internet audio broadcast is available at:

http://www.phmainstreet.com/mba/mv.htm

"PRIDE" is the registered trademark of M. Bryce & Associates (MBA) and can be found on the Internet at:

http://www.phmainstreet.com/mba/pride/