



## Microsoft Project & TimeTraveler™

*When a baseline is not enough, TimeTraveler™ makes Project useable*

### Introduction

Many software products provide working environments for users that are creating intellectual property of significant value. Often such a person has an advanced degree or significant work experience that distinguishes her from her colleagues. The assets these users create are most often unique, and their uniqueness is in fact what determines their value. Enhancing the productivity of these users often results in significant cost savings. Below we present an example of such an enhancement that can be easily extended into other areas.

### The problem with Microsoft Project (and many other applications)

#### Microsoft Project's structure

**Microsoft Project** (or MSP) is project management software developed and sold by [Microsoft](#) which assists project managers in developing plans, assigning resources to tasks, tracking progress, managing budgets and analyzing workloads. (See [Wikipedia](#)) It does these things by creating and manipulating a complex set of inter-related data structures that represent a Project. The data structures contain all the aforementioned objects (and many more).

#### Typical Usage

The MSP user starts by creating these structures and verifying that they accurately reflect the project she wishes to analyze. Often the project plan created is quite detailed. Next come a series of interactions with the plan, which assist the project manager in understanding and optimizing the plan. These normally include entering actual task completion data into MSP and comparing actual progress against planned progress. To assist in this process MSP allows for the creation of a baseline plan that can be compared to the project's actual progress. The interrelation of the resources (including people and money) in the baseline determines the time (and money) projected to complete the plan. MSP assists the Project Manager in her quest to optimize expenditures and minimize time by providing for the tracking and "leveling" of all resources. This exquisite capability rests on knowing the serial dependencies of all the tasks and the exclusivity properties of all the resources, thus enabling MSP's leveling feature. The Project Manager has to specify what a resource can do in parallel and what it (possibly he or she) must do serially. By pushing the "level now" button the user can direct MSP to insure the relationships of the tasks and resources she has specified are true. The result is a schedule. For the user, the time of completion (or the cost of completion) is probably the interesting part.

Unfortunately, leveling the resources can make significant changes to the times at which tasks are expected to be accomplished and to the resources assigned to do the work. The unfortunate part is that the changes themselves are in data structures that are not easily displayed and comprehended by the Project Manager. This leads to a situation in which she can inadvertently make sweeping changes to her project plan by changing a few attributes of the tasks and resources (pay rate, time to accomplish, etc.) and not realize the consequences. In



fact, the point where she does realize there has been an undesirable result may be sometime later. It would then be very helpful for her to “back up” to a previous version of the MSP data.

## TimeTraveling Back in Time

It is not unheard of that our intrepid project manager trashes her computer and its software in favor of pencil and paper. This happens because significant changes in the project plan are often hidden from view. It may have taken days or weeks before she realizes her dilemma. Possibly days or weeks worth of effort have been expended on wasted work by the project team. (The scenario we describe is often encountered in work in which the user has a deep understanding of the science and technology she is manipulating, and at least a reasonable understanding of the data transformations being effected by the application. Unfortunately, the consequence of self confidence is occasionally a disaster.) In our experience, the next step is always resetting the plan. With MSP, it is most often a variation of the following procedure.

First: identify the cause of the problem. If the project has been proceeding for some time before the problem is recognized, you cannot just tell everyone to go back a week and redo his or her work. Instead, you must first correct the error (change the time to accomplish a task to a reasonable duration) and then re-compute and re-optimize the plan. Fortunately, in most cases it is possible to revert to yesterday’s (or last week’s) plan, fix the error and re-optimize. You will now find a few deviations from the plan, which you handle just like other deviations from plan caused by unanticipated events in the real world. You capture the actual data against the *new* expected plan and manage as always. The reversion, correct, re-plan and fix sequence can only happen if the reversion can actually be accomplished.

## Conclusion

Even when the data and applications users work with are complex and interrelated, it is often possible to recover from errors of omission and commission by reverting to a known-good prior point-in-time and proceeding forward again. It may even be possible, depending on the users level of understanding of the application to revert only a part of the data, perhaps one or two tables, re-execute the application, patch or fix-up the data, and verify the result. Very often, **TimeTraveler™** makes this reversion possible.