

;login:

THE MAGAZINE OF USENIX & SAGE

April 2002 • Volume 27 • Number 2

inside:

THE WORKPLACE

Planning a Model Retirement
BY RAY SWARTZ



USENIX & SAGE

The Advanced Computing Systems Association &
The System Administrators Guild

planning a model retirement

by Ray Swartz

Ray Swartz started financial planning in 1988. Thanks to good luck, a good plan, and hard work he enjoyed an extended vacation for all of 2001. He doesn't know how long this will last, but his model tells him not to worry about the near future. He hopes the model is right.



ray@trainingonline.net

In my last column, “When Do You Plan To Retire?” I stated that “to have a nice retirement you have to plan to have a nice retirement.” As such, I suggested you collect data on your expenses, income, and investments, and begin designing your future.

What are you supposed to do after collecting all this information? Just generating the numbers and stating your retirement plans is an extremely valuable exercise, even if that is all you do. However, once you have the numbers, it would be nice to know how close you are to achieving your retirement goals and what specific steps you should take now and in the future to get where you want to go financially.

One option is to bundle all this information up and deliver it to your financial advisor. From it, your advisor will generate a large notebook full of charts, graphs, and an analysis of your ability to finance the retirement you've designed. The most important part of this report is the recommendations that you should follow to reach your retirement goals. Typically, these “financial plans” cost a few hundred dollars.

These reports often take into account estate planning, taxes, investment returns, inflation rates, college costs, and other financial issues. Generally, they are comprehensive and very useful. This is the first step I took when I started my financial planning back in 1988.

While my report was quite informative, it had certain limitations. Many assumptions are needed to complete the planning process, and as time went on some of these assumptions didn't turn out to be accurate. For example, the planning process required me to make estimates of my future income. But, since I worked for myself there was no way to accurately predict what my income would be in coming years. Thus, the recommendations became less useful as the years went by and my income diverged from my predictions.

As my plan became out-of-date, I wondered what I should do? While I could have had another plan prepared, financial reports cost money. I didn't want to have to generate a new plan every few years just to keep updating my income and expense numbers.

What's more, the generated financial plan was completely static. There was no way to get answers to “what if” questions: for example, if I have some extra cash at the end of the year should I make extra mortgage payments, invest the money, or donate to charity?

Planning Your Own Future

After reviewing my planning report, I realized that I could create a personalized version of it in a spreadsheet program. I started out with a simple model that began with my current holdings and expenses and then calculated what happened over time using this equation:

$$\text{cash next year} = \text{cash this year} + \text{investment gains} + \text{income} - \text{expenses} - \text{taxes}.$$

This simplified set of calculations allowed me to see where I was headed if I kept to my earning and spending habits. I could also make a few changes and instantly see the result. As things arose, I fiddled more with the sheet.

Over time, the sophistication of the model grew. I added a way to calculate federal and state taxes using tax tables instead of estimating it. I split my investments into taxable and non-taxable (stocks, IRA, life insurance, etc.). I added a calculation for my house

mortgage. I separated my expenses into those that were tax deductible and those that weren't and further refined them to those that would be subject to the inflation rate. In fact, every time a new financial twist entered my life, some new columns were added to the spreadsheet.

Now, whenever I wanted to perform "what if" analysis, I simply changed a few numbers and saw what happened. This greatly assisted many decisions that I had to make about how to handle my money.

As the model progressed, it involved more interaction among the data, and the limitations of the spreadsheet paradigm began to get in the way. Handling complex logical tests was difficult, error-prone, and cumbersome. For example, when I ran out of cash, where did the money to pay my expenses come from? Was it from my IRA, insurance, after-tax investments, or home-equity line of credit? There are various tax consequences depending on which one is used, and writing a single-cell logical test with variable names like A23 got to be too much.

In addition, financial calculations involve auto-correlated data. That is, a year's financial numbers depend on one another. For example, the amount of cash available to pay for expenses depends on this year's income minus this year's expenses. But, suppose that after calculating how much money is required to pay expenses, you discover that you need more cash than you have. Getting access to additional cash will change the expense amount due to interest on borrowed money or taxes from sold investments. Unless I was very careful, my model would get out of phase and give me bogus answers. I began spending lots of time reality testing my, now quite large, planning spreadsheet.

Another problem involved the model's time frame. I had chosen years as my time unit. In essence, my model transacted all my financial business once a year. This meant that money I spent in January wasn't deducted until December 31, and I was allocated interest on it for the entire year, even though in reality, that money had already been spent. Moving the model to a monthly frame to eliminate this foolishness was simply too hard to do.

Like the paper report, the spreadsheet model was static. The spreadsheet had to have all the numbers before it could do the calculations. That meant that I had to supply data for future values that were not knowable. Examples are the inflation rate, investment returns, and interest rates. I took the best guesses I could, but there was no way to factor in the uncertainty inherent in them. If I depended on an 8% return and the market tanked, I could be in serious trouble.

Even with all these shortcomings, I found the spreadsheet model very useful. I based many financial decisions on its predictions. What's more, I learned a great deal about financial planning from designing, debugging, writing, and maintaining it.

Creating a Custom Plan

As some major personal and financial decisions loomed and I began to depend more and more on this model, I was concerned that its shortcomings were beginning to skew my planning. The only way to create a model of the world as I saw it was to program the whole thing from scratch to react just the way I thought it should. I started this effort in mid-2000.

By this time I had a long list of requirements necessary to make the programming effort worthwhile. I wanted this new model to:

- Enable me to keep track of my investments and cash balances.
- Update investment values from the Internet on a daily basis.
- Enable inflation, investment returns, and interest rates to vary over time.
- Produce different scenarios that represented the worst, average, and best futures I could expect.

- Enable me to explicitly state how my assets were to be consumed as they ran out.
- Enable me to apply taxes accurately over time.
- Enable me to have a monthly time frame.
- Enable me to change data as easily as I did in the spreadsheet model.

I considered several different ways to write this program. To ease input, I thought about writing it with a Web front end to handle input and a Perl back end to do all the calculations. While this would have worked, it required a suite of programs to generate and process all the Web pages necessary to hold and transfer all the data. It also required me to install a Web server on my PC, which I didn't want to do.

Another thought was to write the code using programming tools connected to an SQL database. I decided against this to minimize the execution requirements of the code, which might prevent me from using this code in a different location or distributing it to friends.

Finally, I decided to use Java and write the interface with Swing. The resulting program, which I call FinPlan uses a tabbed main window (See Figure 1).

Each tab represents a multi-column table that serves as both display and input. Those tables holding accumulating values keep running totals at the bottom. Figure 2 shows the yearly expense table. Note the running total at the bottom of the right-hand column.

These data windows allow me to see at a glance the current assumptions I am making and the value of my current holdings, to update the data in these tables interactively, and to quickly modify something and see what effect it might have on my financial situation. Note that the values in these tables are what make up my financial plan.

How Random Can You Get?

I wrote FinPlan to be a dynamic model. That is, I didn't want to get just the one answer the spreadsheet could give me. Instead, I wanted to randomly choose values for the unknowable variables – investment return, interest rate, inflation rate – and then see what effect they had as they moved in the specified range. Figure 3 shows the distributions table. Here, I am using a Gaussian (normal) distribution and I've specified the mean and standard deviation for investment return, interest, and inflation rates.

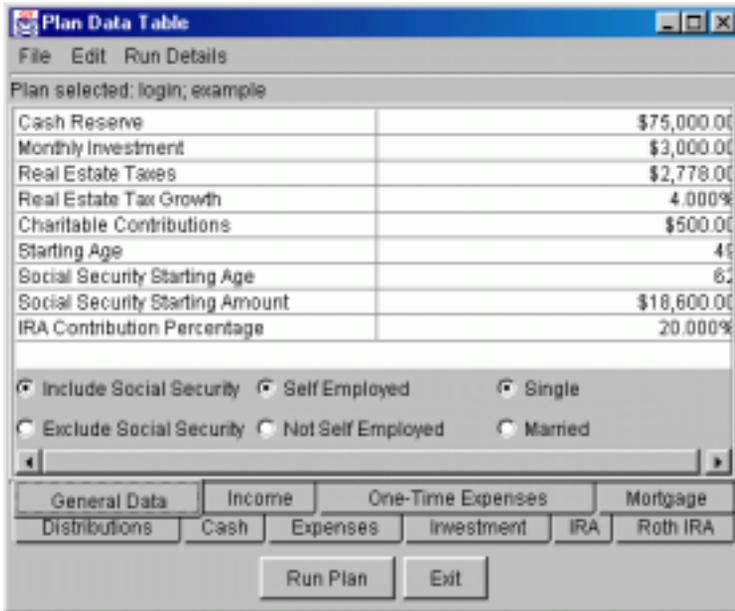


Figure 1: FinPlan General Data

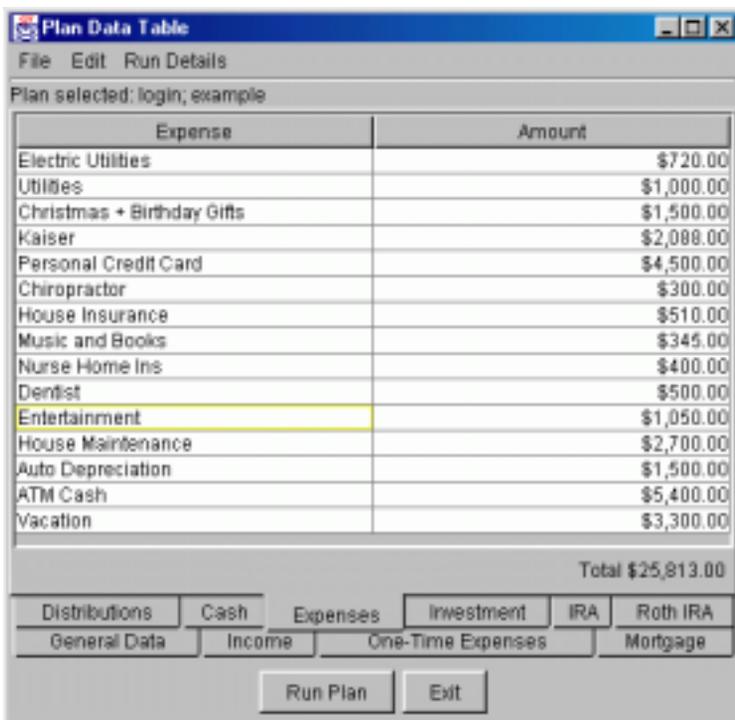


Figure 2: FinPlan Expenses

In order to see the effect of my current plan, I start with the numbers in the tables. Then, on a monthly basis, I factor in income, investment returns, interest, social security income, general expenses (modified by inflation), one-time expenses, mortgage payments, and taxes. I continue this until I run out of all money (cash, investments, IRA, and Roth IRA). I repeat this process 5000 times. The number of iterations is user-specified. An entire run takes about five seconds on my machine (Pentium III PC).

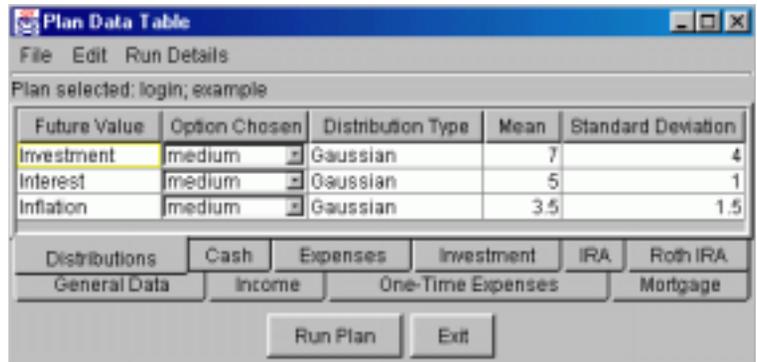


Figure 3: FinPlan Distribution

Figure 4 shows the output data displayed after a run of the model. The output window is a three-column summary of when my money ran out during each of the 5000 iterations. By the way, this plan is completely made up and, aside from my age, the data are not meaningful.

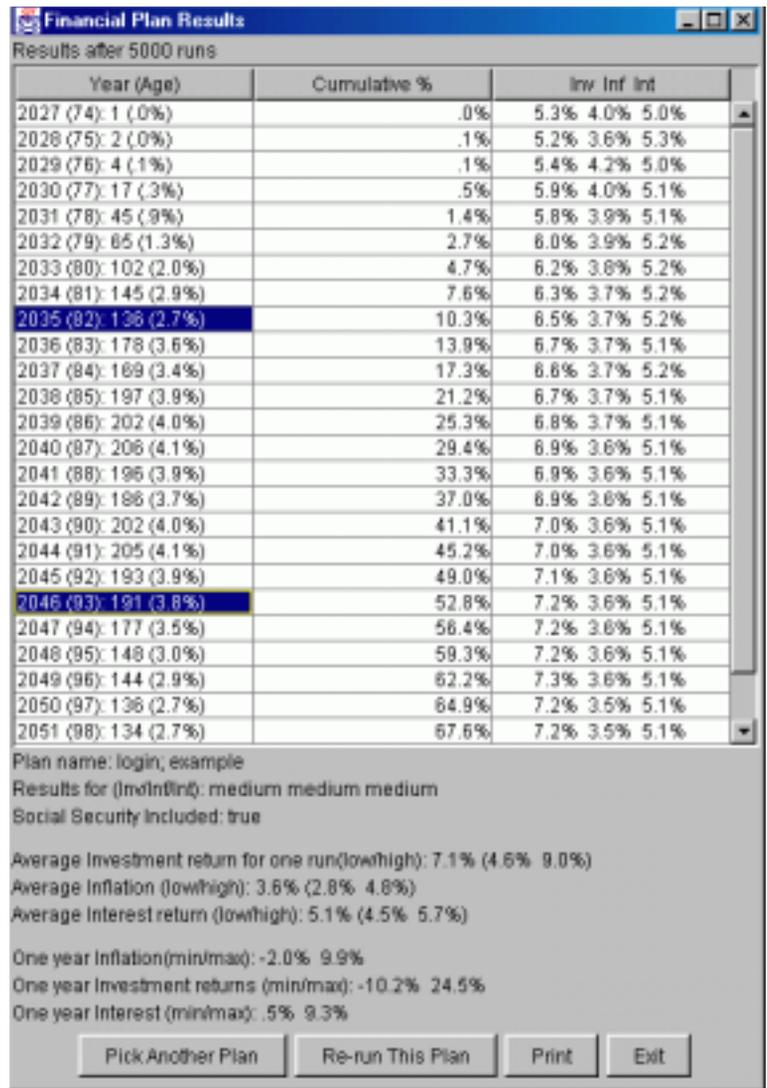


Figure 4: FinPlan Output Summary

The first column shows the year, my age, how many times this was the “money-out” year, and the percentage. The second column is a cumulative percentage. The third column is the average investment return, inflation rate, and interest rate for the plans that ended that year. There is complete run summary data at the bottom. The highlighted years represent 10% and 50% cumulatively. I cut off each plan at 100 years of age.

This run of an example plan tells me that the earliest I ran out of money was at age 74 and that my investment return over the life of that iteration was 5.3% with an inflation rate of 4.0%. This outcome occurred once in 5000 iterations; it is unlikely to happen in real life.

While FinPlan has been an enormous help in making financial decisions, it has drawbacks. First, it treats all investments the same. Thus, I get the same return on all my investments regardless of what they are in. This is not realistic. The same thing is true of cash in various types of accounts. Also, complicated financial instruments, like life insurance, aren't accurately represented.

There is also a personal drawback. With the push of a button, I can get a daily update on how all of my investments are doing and how the future plays out as a result. When the market drops, I get to see its effect on my entire financial future. I've learned that too much information can be just as useless as not enough!

You don't have to go to the effort I have in creating a financial planning application to enjoy the benefits of doing your own financial planning calculations. Whether you hire a financial planner, use a simple spreadsheet, or write a custom program, mapping out how your current plan plays out in the coming years is a powerful way to ensure that you are building the future you want.