Abstract:
Ruthen-1 established the feasibility of using microsimulation for individuals, to inform their personal strategies for spending down portfolios to finance consumption in retirement, this in the face of uncertainties about mortality, inflation, and investment returns. Simulating millions of independent lifetimes generated distributions of outcomes flowing from the drawdown strategies, revealing major differences across alternative strategies for managing a given retirement portfolio. However, Ruthen-1’s primary purpose was as a feasibility demonstration – establishing an experience base toward the development of a more ambitious, more practical, package in support of financial literacy.

This paper describes the lessons learned, and the design requirements and principles for the next generation “Ruthen-2” modeling package that is currently under construction. In briefest highlight –

- Technically, the new package implements a modeling language that is purpose-built to describe portfolios and the strategies used to manage them. It incorporates a much richer set of building blocks in the form of assets, incomes, loans, and expenses, plus a variety of operations on them. Improved output facilities provide greater convenience, more power, and actively support model validation.

- Substantively, these extensions permit one to address a much broader range of issues, from the treatment of couples, including divorce and survivorship, through retirement timing decisions, to more general objective functions, sensitivity analyses and optimization, and a first pass at the asset accumulation phase. As well, the language-
based approach permits users to model arbitrary tax/transfer systems, including building models for other countries.

- A final section provides a status report on the implementation of Ruthen-2.

Introduction

Perhaps in part as a result of the economic downturn of 2008 - 2009, with investment, debt, employment, and budgetary consequences that continue up to the present, there has been an upsurge of interest in financial literacy. Participants in the discussions have included governments, employers, banks, unions, and the financial management industry, and, of course, individuals and their various associations. Illustrative of these is the report of the Government of Canada’s Task Force on Financial Literacy (2010). Virtually all of the commentators have included prescriptions for increasing the public’s financial literacy so that individuals can make better financial decisions, usually construed as saving more, spending wisely, and avoiding fraud. The several sources have included a variety of definitions of financial literacy, but we’ve noted that they often don’t pay significant attention to individuals’ decisions and outcomes, or to uncertainty.

Our own definition of financial literacy is that “Individuals understand the range of potential courses of action, and appreciate the immediate and longer-term consequences of their financial decisions, including the effects of uncertainty.” Our definition is similar to that of the Task Force on Financial Literacy, but includes a greater attention to the outcomes individuals can expect to experience, and to the impacts of uncertainty. The range of decisions that we consequently include in our definition is broad, much broader than that of most definitions of financial literacy. They include, quite non-comprehensively, decisions about education, work, savings, investment and portfolio management, insurance, the timing of retirement, the drawdown of assets in retirement, and estate planning.

We are sympathetic with the challenges faced by diligent, well-intentioned, experts dealing with financial literacy issues. Given the complexities involved in fostering literacy about a system that includes the immediate and longer-term interactions of individuals’ decisions with complicated systems of tax and benefit programs, combined with a wide variety of risks and uncertainty, it is difficult to provide meaningful, specific advice. Indeed, most expert prescriptions provide general rules of thumb, augmented by the counsel that one should consult one’s financial adviser. The complexity of the task makes it almost impossible to provide personalized advice configured to a particular advisee’s circumstances and objectives. However, based on our reasonably extensive, but informal, experience, citizens are looking for exactly that, personalized analyses that reflect their own situations, and their own personal goals.

Morrison (2007) argues that a microsimulation approach, broadly similar to that of the longitudinal models that governments use to guide their own decisions, offers one promising approach to informing citizens about the consequences of their financial decisions. The focus of this approach is to simulate the range of outcomes likely to result from decisions and strategies, including the impacts of uncertainty, so that individuals and families can make informed decisions. The name for the proposed approach, Ruthen, reflects the underlying methodology, i.e., “Running the numbers.”

By way of exploring the feasibility of this approach, Avery and Morrison (2009) developed a demonstration model, Ruthen-1, to address one component of financial decision making. Specifically, they looked at one of the major problems that individuals and families face, the decision about how to
spend down assets in retirement. Côté (2010) argues that, with the decline of defined benefit pension plans, this decision is crucial to seniors’ financial well-being. Along the same lines, the Society of Actuaries has sponsored a literature review on this topic by MacDonald et al. (2011). Using the Ruthen-1 model, Avery and Morrison (2011) provide a series of illustrative analyses suggesting that traditional rules of thumb for accumulating and decumulating voluntary retirement savings may generally advantage the government sector, financial managers, and prospective heirs, at the expense of those doing the saving. One major conclusion from that work is that a fuller investigation of these issues will require a more powerful methodology than Ruthen-1 provides.

This paper describes a next step in the development of such a more powerful methodology. The Ruthen-2 software currently under construction is an extension of the Ruthen-1 work. It is a tool to support financial literacy in the sense of informing individuals and families about the likely consequences of pursuing particular strategies for saving, investing, spending, etc. Under the Ruthen-2 approach, rather than adopting the usual strategy of telling people what they should be doing, it concentrates on deriving and presenting the likely consequences of various courses of action, in the context in their particular circumstances, and for the objectives they describe. The approach then lets them choose which strategies they prefer. The emphasis lies with the distributions of outcomes and measures of personalized objectives likely to flow from the strategies, along with measures of the central tendencies of those outcomes. These capabilities will also be of interest to financial advisers seeking to tailor their services to the specific circumstances and objectives of their clients.

This paper will briefly describe what we learned from Ruthen-1, and the resulting design requirements for Ruthen-2. Subsequent portions provide a technical description of Ruthen-2’s design, and characterize the structure of a Ruthen-2 model. They also describe potential challenges associated with this approach, and our anticipated responses to those challenges. The paper concludes with a technical description of the software, an indication of how it will be able to help address some significant issues in financial planning, and a progress report on the implementation of Ruthen-2.

What We Learned from Ruthen-1

The Ruthen-1 software, presented at the 2009 IMA meetings, was intended, from the beginning, to be a feasibility demonstration to see if it was possible to build a micro-simulation model to help retirees project the consequences, many of them distributional, of adopting alternative prospective strategies for spending down their retirement assets. Although it proved to be surprisingly useful, we sought primarily to confirm whether one could simulate millions of fairly realistic retirement lifetimes in a few minutes, and produce meaningful results. Beyond that, we expected to learn what more was required, so that we could include it in the next version. There were two primary streams for this learning. One was the lessons we learned from the experience of writing and using Ruthen-1, some of them largely confirmations of what we thought we already knew. The second was what we learned when we explained Ruthen to others, essentially representatives of the spectrum of prospective clients. This section summarizes the highlights from both streams, inputs that played an important role in designing Ruthen-2.

Inputs from Others

Our “others” were a mixed group. They varied from financial management professionals and actuaries, to economist and policy analyst colleagues, to members of our respective congregations, family members, friends, and casual acquaintances. The circumstances ranged from formal presentations, to email correspondence, to casual conversations about how our retirements were coming, to, in the extreme,
talking to random seatmates on the bus, and a distant neighbour encountered during a morning walk. We'll term them “respondents.” Their several contributions reinforced or strongly influenced our understanding of what Ruthen-2 should be, and what it should be able to do.

**They really want to have access to this type of software.**

Respondents were enthusiastic about Ruthen, often even from hearing only a casual description of it. They were concerned about managing their assets in retirement, and unhappy with the resources available to them. They wanted assistance in doing it effectively. They were quick to share the various challenges and obstacles they face, particularly in managing assets that are all they'll have now that they are retired. Surprisingly to us, many of the respondents, including ones we didn't know at all well, on hearing of our plans for Ruthen-2, initiated offers to be beta-testers. The professionals among our respondents indicated that something like Ruthen-2 would be helpful to financial advisers.

**They have very different ages, financial situations, approaches to managing their finances, and different levels of financial literacy.**

Our respondents had very different levels of financial literacy. Perhaps because of recent market declines, pension plan failures, and unilateral cuts to retiree services they had been counting on, they were generally well aware of some of the major risks. However, they had quite a range of risk tolerances, different approaches to managing risks, and different approaches to managing their finances generally. But across these differences they were interested in having a tool that could give them an idea of the outcomes likely to flow from their current strategies or alternatives to these strategies.

They had very different financial situations and wealth levels. Some of the respondents had indexed government pensions on which they could live comfortably if necessary, but wanted to manage various registered and non-registered assets to their best advantage. Others had no pensions, and saw their future welfare as largely dependent on the quality of their asset management. Some were not yet retired, and were keenly interested in when they could retire. Others were well into retirement and concerned about the trade-offs between their own consumption and the estates they would be able to leave. Some were extremely concerned about running out of money before they died, even if it meant not enjoying their early retirements as much as they'd like to, or had planned to. Some were all but planning to run out of money, ensuring that they'd get the full benefit of their sacrifices in saving for retirement. Some wanted strategies that would have them spending more money while they were young enough to do things and enjoy them. Others foresaw health care costs increasing steeply with age, and wanted to have enough resources to cover them. Some reported family and friends living “comfortably” on just basic government pensions and benefits, and were confident of their own ability to do so as well. Some of them owned their own homes. Others had sold them to avoid the burden of keeping them up.

**Most of them don't see themselves in isolation.**

Most of the respondents, even those who were single, were not concerned about just themselves. They saw themselves as part of larger groupings. Couples were concerned to adopt strategies that wouldn't leave a surviving spouse in a precarious position upon the death of the other. Some were quite concerned about trade-offs between the size of estate they might leave for disabled children unable to care for themselves, versus the costs of financing their own retirements. Others wanted to strike a balance between donating to a favoured institution during their retirements versus bequests at their deaths. Nearly all of them were interested, for purposes of planning their finances, in the likely outcomes for others as well as for themselves.
They have very different objectives.

As the preceding comments indicate, our respondents were concerned with very different things. Where they are concerned with themselves, the focus seemed to be on something like actual constant-dollar consumption, rather than some measure of income or wealth that might be a rather imperfect approximation for that consumption. Beyond this, several of them expressed an interest in their discretionary consumption after basic necessities have been taken care off. In other respects, their objectives varied widely, as did their attitudes toward risk. Surprisingly to us, attitudes toward health-related concerns and critical care insurance also varied considerably. Some wanted to be sure to have enough money for anticipated later-life health care costs. Others had concluded that if they had assets they'd end up paying those costs, whereas if they didn't, the level of care would be different, but “the system” would pay, so they might as well use the resources, earlier, for something they cared about.

Many of them distrust governments and the operation of existing tax and benefit programs.

Our respondents were generally appreciative of the benefits flowing from the government tax and benefit systems. At the same time, having paid so much in taxes and contributions over their working careers to provide benefits for others, they felt that they were entitled to these benefits in their own turn. They were frustrated with previous government attempts to reduce these benefits, and with anticipated future attempts to reduce them further. One respondent expressed it as, “I'm willing to take responsibility for my affairs, but I don't want to be played for a sucker.” Many of them felt as if they were getting penalized for saving, with the government telling them, in effect, “Mr. X didn't save, so we (you) have to take care of him. You did, so you get to take care of yourself.” They were not confident that governments' urgings for them to work longer and save more were necessarily in their own interests rather than the government’s interest. They want a tool that will give them a better idea of the likely consequences of decisions they may make. They liked the Ruthen approach of looking at the impacts on actors other than themselves, as well as themselves, this as one way of gauging the objectivity of the advice they receive.

Many of them use financial advisers, but don't always have full confidence in them.

Many of our respondents used financial advisers. Few considered themselves experts. Because their own financial literacies varied, their relationships with these advisers varied from working with them, to simply taking an adviser’s advice on matters they did not understand. But even though our respondents generally recognize that there are major uncertainties involved, many of them felt that they weren't receiving the kinds of answers they want. Their impression, and ours, was that they often seemed to be receiving advice based on rules of thumb, insufficiently personalized to their own circumstances and objectives. On the other side of the coin, some of the financial advisers among our respondents expressed frustration about the effort and costs involved in personalizing advice, and the challenges involved in demonstrating to clients the value that their services added.

Lessons from Our Own Experience

Beyond the lessons we learned from others, we learned, as was expected and intended, a great deal from our own experience in developing and applying Ruthen-1. Some of these lessons proved to be confirmations of our prior expectations. Others we should have anticipated, but didn't. Still others were surprises. Much of the detail from this experience will be apparent in the specifications appearing later in this paper. Consequently, this section will touch on only a few highlights.
• Modeling millions of independent retirements or lifetimes is computationally feasible in runs of only a few minutes, even on low-end hardware such as netbooks.
• Ruthen-1’s tax and benefit sophistication and generality are far too limited. Other jurisdictions are necessary, as well as greater detail when appropriate to the decisions and strategies of interest.
• Ruthen-1 can only model a single individual, with a single retirement age. However, many retirees are couples with different ages and different retirement dates. Others are not yet retired, or may plan on a phased retirement that combines work and pension receipt. All of these require access to more general analytic capabilities than are available in Ruthen-1.
• We know that events such as death and divorce will generally have a major impact on a couple’s consumption and objectives. One needs to include these kinds of events, stochastically when appropriate, in the simulated histories, and show their consequences for various financial strategies.
• Ruthen-1 allows for a very limited set of assets, some of which are “hard coded” into the software. Even though Ruthen-1’s asset set is on a par with those in common software applications, we realized that a much broader range of asset types is necessary, with all of them under the “control” of the analyst. These asset types should range from mutual funds and annuities, to listed investments, durable goods, and even household goods. The parameters defining the assets should permit tracking their acquisition costs, maintenance and insurance expenses, depreciation, and the proceeds and tax consequences should the asset be sold.
• We also realized that we need to be able to model a wider range of income streams and entitlements, as well as various debts and ongoing expenses that are part of families' finances. Again, parameters should allow a model to customize these flows as regards their draws on consumption and any tax consequences, e.g., treatment of the interest paid when one borrows for investment purposes.
• For applicability to the broader set of questions that we and our respondents wanted to address, analyses need to include the working/accumulation phase of subjects’ lives. The software needs to be able to simulate more general forms of asset accumulation and drawdown. It must allow for realistic adaptations to job loss, disability, and forced or involuntary retirement. As well, we need to be able to model strategies that are qualitatively and quantitatively conditional on events occurring within the model, from life events such as death of a spouse, disability, or divorce, to portfolio performance.
• Ruthen-1’s output capabilities, though extensive, are nonetheless too limited. Modellers should be able to build output tables and files from any of the variables available within the model, including the convenient tabulation of functions of those variables. Moreover, those same output capabilities should lend themselves naturally to model validation.

Fortunately, our experience has also suggested ways to address these requirements. The remaining sections will indicate the highlights of our intended approach, and our progress to date.

### Design Requirements for Ruthen-2

Based on our experiences with Ruthen-1, the lessons we learned from it, and comments from the respondents, we were able to formulate a set of requirements for the next version of Ruthen:

- A successful micro-simulation will use hundreds or thousands of parameters, ranging from the specification of assets and entitlements, to the performance of the economy, to the values required for tax and benefit calculations, to the parameters specifying how retirement assets are to be drawn down. The modeller must be able to specify or override any of them in Ruthen-2.
- Ruthen-2 must be much more general than Ruthen-1, handling the tax and benefit systems for other
Canadian provinces, and for other countries, in whatever levels of detail are required by the problems thrown at it. Moreover, it must be capable of addressing the evolution of changes in the tax and benefit programs over time. This includes not only changes in tax and benefit rates, and indexation to inflation or wage growth, but more fundamental changes such as tax reform, or the introduction of new benefit programs.

- It must support the modeling of individuals across a broader set of ages than just a single retirement age, including at least a first pass at working careers and accumulating retirement assets, the potential for earnings after retirement, as well as the choice of retirement age, including phased retirement.
- Ruthen-2 should provide for a much broader range of assets, including multiple, distinct instances of each type, but with the potential for differing properties across the instances, e.g., multiple mutual funds with different distributions of returns. Beyond this, it should support a wider range of income streams and entitlements, as well as the various forms of debt and ongoing expenses that are inherent parts of families' finances.
- It must provide for much more general forms of strategies for accumulation and drawdown. Within this, the relevant transactions, e.g., for buying or selling assets, or borrowing or paying down debts, need to be conditional. That is, the strategies must be able prescribe transactions that are conditional on particular events or situations occurring within the model, from extremes in investment returns, to attaining particular ages, to changes in family structures, or to health status.
- It must be able to model systems and strategies using clients' assumptions rather than just sets of values that, say, financial advisers think are reasonable. It must be able, conveniently, to assess the sensitivity of the results to changes in those assumptions.
- Although measures such as real consumption are important, and much more relevant than gross incomes or even disposable incomes, Ruthen-2 must also be able to calculate discretionary consumption, after necessities and ongoing commitments have been taken care of. More generally, it must be able to handle customized objective functions that better reflect what is important to specific families and individuals. Such considerations may include everything from the likelihood of dying after assets have been exhausted, to the smoothness of consumption before and after retirement, to the likelihood of receiving certain income-tested benefits.
- Ruthen-2 must be able to address family units other than individuals, especially couples. It must explicitly treat events such as death and divorce that change family composition. It must have the capacity to track the finances and well-being of the component members after such changes, and include the costs associated with those changes, whether it be the legal costs of a divorce, or the payments to executors or for funeral expenses.
- Well thought out sets of standard outputs are necessary, but far from sufficient. Users will want to be able to conveniently generate complicated customized outputs in a variety of levels of detail, and to export those results in forms convenient for subsequent processing and secondary analyses. These same output capabilities should be designed so that they constructively support model validation.
- Ruthen-2 must be able to perform all these tasks, simulating a million or more independent family lifetime histories, and generating all of the desired outputs, within a reasonable period of time. Informally, we think that a full-fledged run should generally require no longer than a typical coffee break.
Functional Design for Ruthen-2

Overview

Ruthen-1 used an interface that is common to most software. A person runs the program, picks a menu item to either initiate an action or pop up a dialogue box to be filled in. Filling a dialogue box typically entails one or more of: picking an item from a list, ticking check boxes, setting radio buttons, and/or typing in short text entries. Most programs interface with a user the same way, be they Windows’ applications, internet pages, or even iPhone apps.

Initially, we considered having Ruthen-2 follow the same pattern. An analyst would run Ruthen-2, and use various menu items and dialogue boxes to set/override the values of system parameters, describe the actors (i.e., the persons), their assets, incomes, loans and expenses, describe the various drawdown methods to be employed, and describe when, during a lifetime, the various assets, incomes, loans, expenses and drawdown methods should be initiated and cancelled.

Very quickly, it became clear that this type of interface would not be practical. There were too many system parameters, and too many different kinds of assets, incomes, loans and expenses. A significant challenge was to design dialogue boxes to indicate when, during a lifetime, such assets, incomes, loans and expenses should be initiated, cancelled, or changed. The program would need a plethora of different, very complicated, interdependent, dialogue boxes to input the required information. Such an interface would be unwieldy, and difficult to master. Moreover, the compromises involved in forcing potentially complex strategies into these dialogues would compromise the generality we wanted. Even when one wanted a simple model, one would still have to contend with a complex menu structure.

Drawing on our previous experience with MAPSIT, we quickly moved to an alternative approach. Rather than use dialogue boxes to specify a system, we decided to employ a purpose-built modeling language. This turned Ruthen-2, from a model like Ruthen-1, into a model building and execution environment. A properly designed language enables one to specify the system to be simulated (i.e., a model) in a “natural” way, using a “natural” language. It especially makes it possible for another person to look at system being simulated (i.e., the model) and easily understand it. Thus, the functional design of Ruthen-2 became the design of a modelling language. The key was to have the language elements reflect the kinds of objects and actions one needs to simulate financial decisions, and to do so naturally and efficiently.

The Modelling Language

Based on the design requirements, as described earlier, the modelling language must be able to specify:

- one or two people as the modelled subjects (if there are two people, they are married, or are the equivalent of being married – e.g. a common-law union);
- their individual assets, loans, incomes, and expenses;
- groups of assets, loans, incomes, or expenses;
- drawdown methods relevant for each applicable asset or group of assets;
- definitions and initializations for variables;
- overrides for system parameters;
- accumulators that specify the desired analytical and validation outputs;
- output files, the measurements that are to be output, and the format of the output
- prompts that enable the analyst to assign values to certain variables during model execution
transactions involving the assets, loans, incomes, and expenses (including any calculations associated with those transactions); and
conditional instructions for specific transactions to be performed when certain events occur.

Specification of people, assets, incomes, loans, expenses, groups, and accumulators would require arguments to describe the individual characteristics of the item being specified. Examples of arguments include:

- the age of an individual at the beginning of a lifetime being simulated,
- the relative mortality factor of an individual,
- the value of an asset when it is first purchased,
- the annual real growth rate (or return) of the value of a financial asset,
- the age at which an individual starts receiving a benefit,
- the fraction of inflation to use to index an income, and
- the initial principle for a loan.

There are hundreds of such arguments, many of them relevant for several different objects, far too many to list here. One of the design features is that there are reasonable default values for all of the arguments for Ruthen-2 objects, e.g., all of the different types of assets, so that a modeler only has to specify desired exceptions to these default values.

In addition, the persons, assets, loans, incomes, expenses, groups, and accumulators must have attributes that typically change dynamically as execution proceeds. These attributes are readily accessible for any calculations performed in the model. As well, all system parameters must be accessible for any calculations. Examples of these dynamic attributes include:

- the current age of a person,
- total value of a person’s non-registered assets,
- value of an asset, in the current year, in current dollars,
- amount of interest income generated by a financial asset during the current year,
- current year realized capital gains associated with sales of an asset,
- total amount received from an income in the current year, and
- amount of interest charged on a loan’s outstanding principal during the current year.

Many of these attributes are common across the objects in a model. For example, the current “value” of an asset is available for every asset in the model. This commonality, and the ability to refer to groups of assets as well as each of the assets individually, makes it convenient to specify inputs to calculations. For example, one can use a single expression to gather all the taxable capital gains generated by arbitrarily many assets. As with the arguments for the constructs, there are far too many attributes to list here.

Transactions, which involve the assets, loans, incomes and expenses, represent the actions that an individual (or the model) can take during the course of a simulated lifetime. An example of a transaction would be to change the drawdown strategy of a particular asset when the value of that asset falls below some specified threshold value. Another example of a transaction would be to sell some or all of a particular asset, and to purchase another, when the individual reaches a specific age. Other examples of transactions include evaluating an expression and assigning that result to a variable, updating an accumulator, making a donation, paying down a loan, divorce, writing out a data record to a file, and calculating taxes and benefits.
**The Tax Engine**

Like Ruthen-1, Ruthen-2 is able to automatically calculate all relevant taxes for a basic, parameter-defined tax system. The advantage of this approach is computational efficiency. Tax calculations occur frequently, and are among the more computation intensive components of the simulations. To make Ruthen-2 more versatile, all tax calculations are isolated in a separate module – the tax engine. Thus, it is possible to write a new “tax engine” for another country (e.g., the United States or Great Britain), remove the Canadian “tax engine” from Ruthen-2, and replace it with the new one. There would then exist a version of Ruthen-2 relevant for that country.

However, it may be that an analyst does not like the way Ruthen-2 calculates taxes. What if the analyst does not have the inclination or ability to write an entire tax engine? (Knowledge of C++ and object oriented programming is a necessity for writing a new Ruthen-2 tax engine.) The modelling language includes the capability to “turn off” the tax engine, and is powerful enough to allow the analyst to “roll his/her own” tax calculations within the model itself. It is clear that such calculations would take longer to execute than those for a “tax engine” built into Ruthen-2, but, more importantly, it would be possible, and would allow the inclusion of any “arcane” aspects of taxes that might be relevant for a particular analysis.

**Structure of a Model**

**Overall Model**

It is clear that to execute the various operations required to simulate one million, or so, lifetimes, the model must be divided up into several sections. There must be a section to perform initializations and provide definitions before any simulation begins. In addition, it must be possible to perform initializations and provide definitions prior to the simulation of each separate lifetime. There must be complementary sections to allow for transactions at the end of each separate lifetime, as well as transactions after all lifetimes have been simulated.

Simulating a lifetime means simulating the separate years of an individual’s/couple’s life, from their start age (or ages if two people are being simulated) to their death(s). We elected to allow for transactions to be specified for the beginning of each year, in the middle of the year, and at the end of the year. Thus, a model is made up of seven sections, sequenced in the order below:

1) Model Initializations  
2) Lifetime Initializations  
3) Year Start Transactions  
4) Mid-Year Transactions  
5) Year End Transactions  
6) Lifetime End Transactions  
7) Model End Transactions

Figure 1 shows these sections and how they are related within the flow of a model.
The sections that follow provide a brief description of the substance for each model section, as well as the procedures that Ruthen performs before and after executing each section.

**The Model Initializations Section**
This section is optional. As its title indicates, Ruthen-2 executes it exactly once, at the very beginning of a run. If specified, it contains all of the output file definitions and any overrides to (the default values of) system parameters such as tax rates, mean inflation rate, random number seed, etc. It can also contain
definitions and one-time initializations of variables and accumulators, that is, the initializations that apply for the entire run as opposed to those that apply just within the lifetimes of individuals and couples. Initial values for variables defined in this section can be hard-coded or prompted for.

**The Lifetime Initializations Section**

This section is mandatory and, in fact, is the only mandatory section in a model since it contains the definitions for each person. It also contains the initial sets of assets, incomes, loans and expenses assigned to them, all groups (of assets, incomes, loans or, expenses ) assigned to them, and any initial drawdown methods assigned to the relevant assets and groups (of assets). These people, assets, incomes, loans, expenses, groups, and drawdown methods are defined and initialized at the beginning of the model's execution, and are reinitialized at the beginning of each successive simulated lifetime so that all of the instantiations start out identically. This section also contains definitions and initializations for all of the variables and accumulators that will be reinitialized at the beginning of each lifetime.

There must be a definition for at least one person, but at most two people. If only one person is defined, s/he is single; if two people are defined, they are treated as married, whether as a traditional marriage, or common-law union. Each person can have any number of different kinds of assets, incomes, loans, expenses, and groups (of assets, incomes, loans or, expenses) defined and assigned to them. Usually, each person would have several assets and a few incomes. The numbers of loans and expenses would typically be small.

There is only one kind of loan, and one kind of expense. However, using the arguments associated with specific instances, one can configure them to reflect a variety of different types, from car loans, to mortgages or investment loans, or from a golf club membership to a cell-phone contract.

In contrast, here are several kinds of assets, and several kinds of incomes. Examples of assets include:

- a primary residence
- household goods
- life insurance
- non-registered investments
- registered investments

Examples of incomes include:

- earnings
- proceeds from registered annuities
- proceeds from non-registered annuities
- employer pension(s)
- government pension

Assets can, optionally, have one or more drawdown methods defined and assigned to them. A group of assets can, optionally, have one or more drawdown methods defined and assigned to it.

Drawdown methods are, generally, the mechanisms by which the individuals withdraw monies from assets in order to support consumption, donations, or gifts. They correspond to triggering the sale of all or part of an asset. Arguments associated with the drawdown methods will provide considerable flexibility in specifying how the assets are drawn down during retirement. Note that a model will also be able to
initiate sales of, or withdrawals from, assets as individual transactions, either together with drawdown methods, or instead of them.

**Year Start Transactions Section**

This section consists of all the specified transactions that (conditionally) occur at the beginning of each year of life, but only after all the automatic drawdowns (via the defined drawdown methods) have been executed, and after all incomes have been received.

In addition, prior to executing this section, Ruthen-2 (randomly) determines if anyone dies during this simulation year, and, if so, stores the information regarding the date of death. If there are two people defined and it is determined that both of them die during the current year, to keep computations simpler, Ruthen-2 currently “allows” just one of them to die but “curses” the other such that s/he dies the next year. This mortality information is accessible (via a person’s attributes) for any calculations that the analyst may want to perform in the model. Ruthen-2 permits modelers to override mortality, and associated “death” attributes, should that be required for special analyses, e. g. assessing the capacity of a particular drawdown strategy to provide satisfactorily for a surviving spouse when the first-to-die spouse dies at different ages.

**Mid-year Transactions Section**

After executing all the modeler-specified transactions in the previous section, Ruthen-2 grows all the assets (i.e., it applies the investment returns using growth rates drawn from the several distributions, including their means, variances, cross-correlations, and longitudinal correlations), performs any automatic loan pay downs, and pays all expenses. After completing these tasks, Ruthen-2 turns to the Mid-year Transactions section. This section consists of all specified transactions that can occur after all assets have been grown, loans paid down, and expenses paid, but before taxes are calculated. Typically, this section is used only to perform death-related transactions in a year that someone dies (e.g., funeral costs and probate taxes, transfer of certain assets to a surviving spouse (if any), and moving of other assets into the deceased's estate). The survivor (if any) may then rebalance assets and initiate or cancel withdrawals to reflect his/her new status as an unattached individual.

Note that the deceased individual’s assets would grow on a part year basis up to the point of death, affecting his/her final tax return. Then, after the mid-year transactions, any portions now owned by the surviving spouse, including any bequests from the deceased spouse, would grow by the remaining-year part-year growth associated with the relevant asset.

**Year End Transactions Section**

After executing the mid-year transactions (if any), Ruthen-2 calculates all taxes, final incomes, and the associated consumption. The transactions in the Year End Transactions section would typically consist of updating accumulators, writing records to output files and implementing any “reactive” transactions (e.g., making charitable donations or perhaps reinvesting excess monies rather than spending them). However, if the model implements these kinds of “reactive” transactions, then the analyst is able to, and must, force Ruthen-2 to recalculate all taxes, final incomes and consumption.

The analyst may also want to perform asset re-assignment in this section, based on events that occurred during the past year, e.g., whether investment returns that have been lower or higher than average, or
random events have resulted in unanticipated external incomes or losses, or stochastically generated expenses such as repairs on a house, or the purchase of new car.

With the completion of this section, a year of simulated life is now over for the individual or couple. This section may, therefore, also contain final analyst-specified transactions such as updating accumulators, updating any analyst-defined variables that may be needed later, and output of records to certain data files and/or validation/debugging files.

**Lifetime End Transactions Section**

This section consists of all the transactions that can occur at the end of a lifetime after all of the persons defined in the modelled family have died. Typically, the transactions in this section would just consist of updating accumulators, e.g., for the desired lifetime summary measures, or the output of records to validation/debugging files. Typically, bequests would have been handled in the Mid-year Transactions section of the model.

**Model End Transactions Section**

Ruthen-2 executes this section exactly once, after all of the simulated lifetimes are completed. It consists of all the transactions that occur at the end of the model's run. This section would probably consist only of output transactions. However, there could be several conditional transactions, since the analyst may only want to output some records under certain conditions, such as when a “debug mode” is turned on.

**Illustrative Model Features**

We anticipate that the primary use of Ruthen-2 will be to project the consequences, as distributions and their central tendencies, of alternative financial strategies, personalized for individual circumstances and objectives. We have designed the language to make this possible. At the same time, drawing on the financial planning literature, we have designed the Ruthen language to be able, conveniently, to execute a number of more specialized analyses. In this section we mention just a few of them.

**Timing of Death:** In what we expect to be a typical Ruthen application, the analyst will choose a mortality schedule for its subject individuals, perhaps something like the standard Canada Life Tables, but adjusted for assumed family genetics. Ruthen will inherently be able to project results for the widowed partner upon the death of the first spouse to die under the typical stochastic approach to mortality. However, for some purposes one may want to look at survivorship from the perspective of the age at which that first death occurs, and to ensure that there are adequate sample sizes across the range of those ages. Ruthen provides for an analyst to override the built-in mortality tables and control mortality directly. The focus might be, for example, to ensure that an apparently good financial strategy did not lead to vulnerabilities for the survivor if the spouse died within a particular age range. We estimate that it would require fewer than twenty additional statements to customize a typical Ruthen model to yield equal samples for different ages at death of a specified spouse. These additional statements include not only the actual control of the mortality, but all of the specifications for the customized outputs across this range, and their output to appropriate files or windows.

**Quantitative Measures of the Significance of Uncertainty:** The financial literature is replete with indications of the importance of risk and uncertainty. However, that same literature offers little in the way of quantitative analyses of that importance, other than some very esoteric studies, usually conducted on a
pre-tax basis, that are not directed to financial managers or their clients. Because Ruthen specifies its models’ uncertainty explicitly, analysts can easily generate projections of outcomes for differing levels of uncertainty, e.g., different distributions of inflation and returns on investments, and examine the resulting impacts on the outcomes flowing from the strategies. They can generate deterministic models by turning off the uncertainty altogether. These capacities allow a user to assess how much difference the uncertainty makes to the results. We anticipate that some strategies, e.g. those that involve leveraged investments, will be particularly sensitive to these stochastic elements.

**Distributions of Investment Returns:** Much of the financial planning literature uses a very simple, deterministic approach to returns on investments during the accumulation and decumulation stages. Often they assume a single fixed return rate across the entire lifetime. More sophisticated models may incorporate a gradual reduction in the return to reflect a recommended increase in conservatism as their subjects age. A few will adopt stochastic returns to investments, perhaps even to point of including longitudinal correlation, or sets of assets for which the returns are imperfectly correlated. Because of the mathematical challenges involved, almost none of them attempt an optimization of consumption incorporating the effects of investments interacting with a realistic tax and transfer system. Using Ruthen, all of this comes almost for free. If an analyst is willing to specify the form of the returns, in whatever level of detail s/he deems appropriate, and the accumulation/decumulation strategy, Ruthen does the rest in terms of deriving the distribution of resulting outcomes. One is free to explore the implications of diversification and economies of scale, of portfolio behaviour across business cycles, e.g., the impacts of overshooting during and after downturns, the changing mixes of interest, dividends, and capital gains, or the implications of “fat-tailed” distributions for investment returns.

**Sensitivity Analyses:** Analyzing the likely impacts of strategies is useful. Being able to compare the distributions of outcomes across alternative strategies is better, particularly if one’s goal is to develop good strategies, or to assess the implications of various rules of thumb. We have designed the Ruthen language, and particularly its control structures and its accumulators for outputs, to facilitate sensitivity analyses. Thus, in a single Ruthen run, a user might elect to generate a vector showing the fraction of years in which the subject individual receives income-tested benefits, presented as a function of an input parameter to the drawdown strategy. Or, using simple secondary analysis techniques and standard Ruthen outputs, a user would be able to generate the decile values for the after-tax value of the estate, with these sets of deciles presented as a function of some variable, e.g., real return to the post-retirement portfolio.

**Choosing a Retirement Date:** Although statistics have shown an increased inclination for retirees to adopt phased retirements, and for working after retirement, many individuals are more oriented to “working to live, rather than living to work.” They are interested in when they can, or when they should, retire. Such decisions are typically complex, They involve not only health and mortality considerations, the characteristics of a spouse and his/her finances and retirement goals, and the atmosphere of the work environment, but also the specifics of any pension plans that might be relevant, and the rules for drawing on government pensions and registered savings. Moreover, all of these considerations interact with a complex system of tax and benefit programs. For several of our respondents, in an uncertain economic environment, the retirement timing issues also involved assessing whether they had reached a situation in which, if necessary, they could live at a comfortable level without working longer “just in case something happened.” Ruthen’s capacities for conditional events and sensitivity analyses make it practical, using a single model, to explore the consequences of retiring at different ages, including determining such ages independently for spouses. These kinds of analyses would contribute to improved decision making by quantifying the trade-offs between working longer and retiring.
Incorporating Career Volatility: Choosing when to retire is desirable, when actors can control that decision. However, in the face of downsizing, recessions, mergers, and corporate bankruptcies, choosing when one retires is not always possible. Many seniors and near-seniors have discovered, upon their losing jobs and finding that gaining re-employment is unrealistic, that they have become involuntarily retired. Even in the absence of such a final outcome, it is a given that working careers routinely involve volatility from sources as disparate as disability, major illnesses, or job loss, or changes of employer. Morrison (2000) drawing on administrative data from Canada’s national longitudinal Record of Earnings, shows that earnings are quite volatile. The standard deviation of annual real earnings, divided by the level of those earnings, exceeded 0.4 for more than half of the individuals in that very large representative sample. Beyond suggesting the maintenance of a six-month emergency fund, most financial planning guides do not address how to respond to such volatility, or the priorities that should apply between regular retirement savings and replacing the emergency fund once one has used it. Similarly, traditional financial guides do not indicate the extent to which one should rely on retirement savings if the emergency fund should become exhausted. Ruthen models would allow a user to explore various strategies for saving and spending within an uncertain environment, including earnings volatility and the prospects of becoming involuntarily retired.

Health-conditioned Strategies: Retirement advisers are keenly aware that the health is a major issue for retirees. Health status is an important input into mortality, affecting the size of the portfolio required to sustain any given level of consumption until death. Disabilities before death strongly affect the potential to work and to save for retirement. Disabilities after retirement can affect consumption by limiting retirees’ abilities to participate in planned retirement activities, or incurring additional expenditures to overcome limitations, e.g., moving or adding a wheelchair ramp to one’s dwelling. Indeed, the entire critical care insurance industry exists to respond to one component of these health concerns and linkages. However, in most financial planning, these considerations are only loosely linked to recommended financial strategies, if they are present at all. Using Ruthen, an analyst willing to specify the stochastic evolution of health status variables, and their implications for incomes and expenditures, can incorporate these health considerations into the assessment of alternative financial strategies. We look forward to the opportunities to work with health professionals to explore this type of integration.

Anticipated Challenges, and Prospective Solutions

More Complicated Models
Although a modelling language makes it easier for an analyst to specify a model, such models can become very complicated, though many of them will not be particularly complex. Some knowledge of programming and programming techniques would be a great asset, yet many potential users of Ruthen-2 probably lack this background. To address this challenge, we anticipate that there will be three primary approaches to using Ruthen-2: a “browse” mode; a “normal” mode; and an “advanced” mode.

1. Browse mode would entail providing an analyst with data/results, and documentation, from prior model(s) runs, probably in a spreadsheet format, that s/he could look at and analyse.

2. For normal mode, a number of “standard” models would be available for an analyst to run. These “standard” models would contain run-time prompts that would provide an analyst with some, limited, control over certain aspects of the model’s operation.
3. Finally, advanced mode would entail an analyst editing an existing model, or writing a new model “from scratch,” and then running it, likely with the specification of a variety of custom outputs.

**Execution Speed**

Ruthen-1 is able to simulate a million independent lifetimes, and write out the results, within a few minutes on most, modern day computers. On a reasonably fast off-the-shelf, but far from state-of-the-art, machine, executing a million instantiation run takes a little more than one minute. But, the Ruthen-1 model is far less complicated, with fewer options, than most of the Ruthen-2 models we envisage. In addition, the Ruthen-1 model is executed entirely in machine language, whereas a Ruthen-2 model will be translated into a pseudo-code version, which is then executed.

It is clear that a Ruthen-2 model, even one that is the equivalent of Ruthen-1, will take longer to execute. Unfortunately, we do not know at this stage how much longer that execution will take. We anticipate that most Ruthen-2 models, simulating a million or more lifetimes, will still execute within a reasonable length of time, say 10 or 20 minutes for a run.

If the time requirements prove to be a problem, in that Ruthen-2 is too slow, there are two possible solutions.

1. The first is simply to get a faster computer. This is not very unreasonable since computer speeds are increasing every year.

2. The second solution is to alter Ruthen-2 so that it takes advantage of the multi-core CPUs that exist in almost all current computers. This would mean simulating the lifetimes in parallel. It is expected that such a modification would decrease run times by 50% or more. Similar improvements might be obtained by distributing the processing across multiple machines in a network. However, this enhancement of Ruthen-2 is currently a low priority, and will not be implemented until a future version, perhaps for Ruthen-3.

**A Status Report**

The language and software design for Ruthen-2 have been completed. We have written an internal language document that will form the basis of a future model design, and the starting point for a modelling language reference manual.

Beyond this, the implementation of the software has begun. Ruthen-2 is being written using the C++ programming language (under the Microsoft Windows operating system) and an object-oriented programming paradigm. Thus, the software is made up of a large set of object classes. These classes (and hence, the program itself) can be divided into a number of components:

- Main Program and User Interface Framework
- An Editor - This allows an analyst to build, edit, and maintain a model within the Ruthen-2 environment.
• Lexical Scanner and Language Parser - These classes scan and parse the model written by the analyst, identifying any syntax errors. This component will also generate the internal pseudo-code for model execution.

• Pseudo-Code/Executable - These classes represent the bulk of the model that gets executed, including all of the classes required to perform the model’s calculations.

• Transactions - As indicated in the functional design section above, there are many types of transactions. Each transaction has its own class. In addition, there are special “initialization transactions” that get executed only once, prior to Ruthen-2 beginning its lifetime simulations loop.

• Assets, Incomes, Loans, and Expenses - Each type of asset, income, loan, and expense has its own class.

• Drawdown Methods - Each type of drawdown method has its own class.

• The Tax Engine – Described earlier in this document, the tax engine provides for the efficient calculation of what is typically the most computation-intensive component of a model. Its calculations can be overridden if the analyst requires a different or more detailed set of calculation, though at an increase in the time required to run the model.

• Utilities – These are miscellaneous “helper” classes and routines used by various parts of the program.

The main program and framework have been completed.

A basic editor has been written. More sophisticated functionality will be added later, but this is currently a low priority. However, for example, Ruthen-2 has the ability to connect to and communicate with any external text editor that the analyst chooses to use.

Work is currently progressing on the lexical scanner and parser. We expect that these classes will be completed within the next two months. Afterwards, the pseudo-code classes will be written, followed by the transactions, assets, incomes, loan, expense, and drawdown classes.

Note that it will not be necessary to complete all of the transactions, assets, incomes, loan, expense, and drawdown classes in order to get a preliminary working version of Ruthen-2 up and running. Nor will it be necessary (as indicated in the Functional Design, above) to have completed a tax engine to be able to use Ruthen-2. Given these lack of constraints, we expect to have a preliminary working version of Ruthen-2 by the end of 2011. We hope to complete a full, beta-test version by the summer of 2012.

**Summary**

During our design work on Ruthen we have received advice and support from a variety of sources, from actuaries and financial planning professionals, to government colleagues, friends, and family members. Most of them, even when they dislike or disagree with our preliminary findings, have agreed that microsimulation has something important to contribute toward a practical approach to financial literacy.
We have taken seriously their several suggestions about what a good financial model should do for its users. Their various themes: treatment of uncertainty, the requirement for generality together with the capacity to personalize to individual clients, the variability of individual objectives, the necessity of treating units larger than individuals and to incorporate major life events, the need to validate analyses, and so many others, have challenged and inspired us as we have sought to build a tool that accommodates them. To this point in our development of Ruthen-2 we believe we have made a good start in the exercise. The approach and the language we have described in this paper should provide a reasonable first pass at addressing most of the requirements. Our work to date suggests that models built in the language will be practical in the sense of producing the intended results in a timely manner, in their construction, their execution, and in the communication of their results. We look forward to completing the work, and to sharing the results with the many respondents who have advised us along the way.

References


Côté, Jean-Daniel. “Is Canada ready for the DC spend-down phase?” Benefits Canada (newsletter) November 26, 2010


