Business Model for a New Venture

0728164

Abstract

Spreadsheets are by far the most common tools for end-user development – that is, they are the only Turing-complete artifacts most people will ever "program". Roe, noting that spreadsheets are frequently used as a metaphor and an example of empirical modelling principles, implemented an EDEN-based spreadsheet framework in 2002. This framework was used as the basis to build a business model for a new venture, which the author intends to launch later this year. The same task had also been attempted in Microsoft Excel, giving insights into the relative utility of the two frameworks. The Lean Startup Methodology, a conceptual framework intended to turn entrepreneurship into a true management science by distilling *assumptions* into testable *hypotheses*, was a key cognitive tool used throughout the process. Both the model itself and the task of building it helped solidify and test certain assumptions about the venture, proving the usefulness of EM techniques in computerized decision support.

1 Spreadsheets

Computerised spreadsheets have a long pedigree, having existed since the days of mainframe computing. They came to mass popularity in 1979 with the creation of VisiCalc, a spreashseet application for personal computers. VisiCalc was so succesful that it became known as the "killer app" for PCs – in other words, it proved so useful that it became a reason for people to purchase the new, expensive personal computers.

Modern spreadsheet applications such as Microsoft Excel are Turing complete, and are often the only such environment accessible to most non-programmers. That is, they allow those without programming knowledge the ability to implement simple – or complex – formulas and calculations on sets of data. One explanation for the wide adoption of spreadsheets is that, unlike most programming environments, they allow variables and dependencies between variables to be laid out spatially. This allows humans to use their highly developed intuitions about *spaces*, especially their ability to sort, categorise and spot patterns in data when it is arranged visually.

Another aspect of spreadsheets, perhaps more relevant from an Empirical Modelling perspective, is that they are forgiving, and allow part of a program to work even if other parts are incomplete or broken. This makes it possible for users to develop a spreadsheet model iteratively – that is, the modeller can use the model to test assumptions about the domain, allowing them to improve their own construal of the domain, which then allows them to further improve the computerised model, and so on.

The original idea for this project was to perform such spreadsheet modelling in the EM environment, and explore how the extra capabilities of EDEN made this task easier than in a standard spreadsheet environment such as Excel. For example, Cadence could have facilitated the creation of interesting visualisations and also provided an easy way to construct time-based dependencies. However, with the practical goal of the model in mind, these approaches did not appear to be as useful as might be first assumed – in the end, most of the dependencies in the model were built from simple mathematical relations, such as addition and multiplacation.

In fact, the ability to arrange data in a 2-dimensional grid, seemingly a very simple feature, proved useful in itself. Joel Spolsky, a project manager for Microsoft Excel, had a similar observation (5):

> Round about 1993 a couple of us went on customer visits to see how people were using Excel.
>
> We found a fellow whose entire job consisted of maintaining the "number of injuries this week" spreadsheet for a large, highly-regulated utility.
>
> Once a week, he opened an Excel spreadsheet which listed ten facilities, containing the name of the facilities and

> the number 0, which indicated that were 0 injuries that week. (They never had injuries).
>
> [. . .]
>
> Over the next two weeks we visited dozens of Excel customers, and did not see anyone using Excel to actually perform what you would call calculations. Almost all of them were using Excel because it was a convenient way to create a table.

He went on to note that the data structure provided by Excel – the 2d table – apparently proved more useful than the ability to perform calculations, and that many other succesful applications also provided end-users with such "fancy data structures". Being able to organise information in a logical way can be more important when modelling than the ability to define complex relationships between variables, a fact which was underlined during my own experience of creating the model in this paper.

My model was built on top of the spreadsheet framework created by Roe in 2002. This provides some basic spreadsheet functionality, with cells based on EDEN variables. This allows more complicated dependencies that those available via standard spreadsheet programs, as well as integration with DoNaLD and Scout for creating graphs and visualisations. Of course, many features of programs such Excel are missing from Roe's framework.

2 Business Models & The Lean Startup Methodology

One common use for spreadsheets is crafting *business models*. This is a broad term, which refers to a range of models and frameworks that attempt to capture the important elements of a business and the relations between those elements. One of the most common frameworks is Alexander Osterwalder's Business Model Canvas, which splits a business into nine building blocks, including revenue streams, key partners, marketing channels, customer segments, and so on.

When implementing a business model in a spreadsheet, we are interested in the aspects of the business that can be modelled quantitively. Looking at our purpose in creating a business model can help us work out precisely which elements we want to model. Generally, the computerised business model will be used as a decision support system; for example, to help us decide if a new venture is viable, or the requirements – such as marketing resources – that are needed for a new venture to succeed. According to (2), there are 3 issues that decision support software aids with:

1. Problem identification
2. Developing alternative solutions
3. Selecting solutions

In this case, the act of creating the business model itself helps with the problem identification step, by encouraging the modeller to "drill down" into the input variables for each piece of data. This was certainly my own experience when creating the model.

The second and third steps, developing and selecting solutions, depends on the problem in mind. In my case, I wished to evaluate four different potential products in terms of profitability. Further refinements of the model showed that the means by which products were marketed and the resources available had a great impact on profitability; some products could make money with little investment, wheras some products required significant marketing expenditure to break even, but then would quickly become much more profitable. This is related in more detail below.

Returning to the question of which aspects to model, most spreadsheets only model the revenue streams and cost structure of a business. Obviously, some important, qualitiative aspects of a business, such as customer and partner relationships, are difficult to model with a computer. However, going beyond profits and losses was important; in particular, I wished to incorporate marketing into my model, since marketing has a complicated effect on both revenues and costs.

2.1 The Lean Startup Methodology

The *Lean Startup Methodology* was the conceptual framework I used when creating the model, both as an aid to refining assumptions and in deciding how to model particular aspects of the business.

The idea of the "lean startup" was developed by Steve Blank and Eric Ries (1), and is their attempt to turn entrepreneurship into a valid management science. There is a large body of theory and practice for those attempting to efficiently and effectively run large, established organisations. However, such methods, when applied without modification to startup firms, frequently fail. For example, in established firms where the market is relatively well-understood, it makes sense to spend significant time planning and strategising before spending resources on product development. However, it is extremely common for startup teams to follow the same process, spend months developing their product and refining their plan and strategy, only to launch and find that there is no market demand for what they have developed.

The key insight of the lean startup movement is that entrepreneurs, unlike traditional managers, are operating in conditions of extreme uncertainty – or, in empirical modelling terms, that they have a very vague construal of their target market. With this in mind as the major problem, it becomes clear that writing plans and software are not the most important tasks. The most important task is *reducing uncertainty.* For example, one technique for reducing uncertainty is to launch a simple prototype – the *minimum viable product* – to gauge market demand, before deciding whether to invest resources in developing a fully featured product.

The next step is for entrepreneurs to turn their key assumptions about the businesses into testable hypotheses, which can be disproven by experimentation. Ries states that the two most important hypotheses for a new product are the *value hypothesis* – which states how the product will benefit users – and the *growth hypothesis* – which states how it will gain users.

During the development of my model, I drilled down into these two hypotheses. For the value hypothesis, I looked at the costs and revenues of each product, which depended on a number of different variables. For the growth hypothesis, I looked at the marketing methods for each product. This proved to be more complex, since according to the lean startup there are three distinct marketing models, or *growth engines*, for a startup.

The first is the *paid engine of growth*, which depends on two variables: the cost of acquiring a customer (whether by advertising, direct sales, or otherwise), and the lifetime value of a customer. If LTV is greater than CAC, the startup can profitably invest money in acquiring new customers, and the growth engine is viable.

The second is the *sticky engine of growth*, which depends on a high customer retention rate. If customers are loyal to the product and provide recurring revenues, then the business can profitably grow via word-of-mouth or with low investment in marketing.

The third is the *viral engine of growth*, which depends on customers actively recruting new users. An example is the type of games popular on social networks. The key variable is the *viral coefficient*, the average number of new users recruited by each user. If the viral coefficient is greater than 1, then the business will grow, sometimes very rapidly.

It should be noted that although most startups will focus on one growth engine, it is possible for them to make use of more than one. In my model, two of the products used paid growth, and two used a combination of paid and sticky growth.

2.2 The Business Idea

The business idea my model is based around is my own, which I intend to launch later this year. The idea is to develop applications for *smart televisions*, a new category of hardware platform being developed by various companies, included Google, Apple, Samsung, Panasonic and others, and which aims to combine televisions with internet connectivity and new interaction modes.

There are four potential products I have considered. Each has distinct revenue streams and marketing models and so the decision of which to develop will have a significant impact on the business; analysing each was the main purpose of the model.

1. A Gaming App, with Paid Downloads
2. A Gaming App, with Free Downloads and In-App Purchases
3. An Entertainment App, with Free Downloads and In-App Advertising
4. Contract App Development for Clients

3 Developing the Model

3.1 Prelinary Attempts

My first attempt at creating a business model was in Excel. I made use of a template provided by

Guy Kawasaki, (3). However, as I attempted to fill out his template with the details of my business, I realised there were some limitations. His template was based around a 12-month period, meaning I had to manually make projections for how the business would grow. I realised that I wanted to encode this information as part of a defined marketing model. In addition, I still had not decided what type of products I would be building – I would need to create a spreadsheet that allowed me to compare the different options side-by-side.

I then found a Google Spreadsheet created by Eric Ries (4), which consisted of a very simple side-by-side analysis of five different ideas. I used this as a starting point for my EDEN-based model. One point worth noting is that Google Spreadsheet is essentially a stripped-down version of Excel. Its major advantage is that it is web-based, which allows for multi-user collaboration and cloud access. Otherwise, the programming models afforded by both are effectively the same.

EDEN can be seen as a generalisation of the spreadsheet model, and so affords a wider range of programming techniques. However, over the course of my project, I made little use of these techniques, and crafted my model using straightforward mathematical operations. Essentially, I used Roe's model as just another spreadsheet application, and consider it one of the weaker aspects of my project that I did not try and make use of the more advanced possibilities of EDEN. However, this underlines the fact that relatively simple relations can be sufficient to model a complex domain.

3.2 EDEN Development

3.2.1 Calculating Profits

The first iteration of my model was a straightforward calculation of revenue, gross profit, and net profit, based on a single month about a year after launching the business.

At first glance, it appeared that video advertising would be the most profitable business model – however, I knew that this was based on very unreliable data, so my next step was to attempt a better estimate for the data. This involved determining the input variables for each product and important variable.

3.2.2 Calculating Number of Purchases

The first variable to be analysed in more depth was the *number of purchases*. For games that featured in-app purchases, the model for number of purchases was (number of users) * (proportion who buy items) * (items bought per item-purchaser). Of course, this model could be simplified to (number of users) * (items bought per user), but this model would ignore the important fact that the majority of users do not purchase virtual items, while those that do tend to purchase a lot.

For entertainment apps that profited from video advertisements, the formula was (number of users) * (hours of interaction / user / month) * (advertisements per hour) / 1000. It was assumed that advertising would be bought in blocks of 1000.

The number of purchases for the other two product types, paid app downloads and client sales, were not analysed in more depth at this stage.

After this iteration, it became clear that in-app purchases could be much more profitable than paid downloads, even if the overall proportion of paying users was low, so long as the overall number of players was high. (This is backed up by the mobile gaming market, where in-app purchases have rapidly become the major revenue model). This point may seem trite when written down, but it was not immediately apparent until to me until I created the model for number of in-app purchases. It also suggests the likely marketing strategy for such a product, which would focus on low-cost ways to get lots of users. Even if such users were not engaged, it would increase the chance of attracting a well-engaged, high-paying user.

3.2.3 Calculating Number of Users

The next iteration looked closer at the *number of users*. This required drilling into the marketing model for each product type. For paid app downloads and client work, it was assumed the paid marketing model was in place, and so the number of users depended on the total marketing budget divided by the cost of acquiring a user.

For in-app purchases and video advertising, it was assumed that users of these products would potentially remain users for a long time, generating returning revenue. Therefore, the sticky marketing model was also incorporated. This required incorporating the users added per month and the monthly retention rate; procedures were created to calculate the total retained users after a set period of time. This was the only point that the more advanced features of EDEN were utilized.

By exploring the model after this iteration, it became clear that the size of the marketing budget had a major impact on profitability, and for some products, a large marketing budget was required before significant profits could be made. Also, for products that relied on stickiness, a change in retention rate from 0.8 to 0.9 might look small, but represents a halving in the number of customers that leave per month, leading to a huge increase in customers over several months.

3.2.4 Calculating Overheads

The next step was to drill into overheads. It was assumed the main overhead would be rent and salary, combined into one variable since it was assumed some sort of per-desk office space could be found. For three of the products, it was assumed that staff numbers and overheads would be static. However, it was assumed that more developers would need to be hired if more contract work was sold, and that was incorporated into the model (Strictly speaking, this extra salary cost should have been incorporated into cost of goods sold). This made it clear that client work could be profitable during the early days of the company, but that developing original products would be more lucrative, provided the resources were available to market them.

At this point, it would have been possible to refine the model further, but I decided that it had reached the point of diminishing returns on complexity.

3.3 Comparison of Frameworks

As mentioned above, I did not make much use of the unique features of EDEN. This would be one possible avenue to extend this project. For example, one could:

- Use DoNaLD and Scout to create visualisations.
- Use Cadence to work on dynamic, time-based models, with animations.
- Extract various peices of logic into reusable functions, making it easy to create future business models.

When comparing Roe's EDEN-based spreadsheet to more polished spreadsheet applications such as Excel, or even Google Spreadsheet, a number of shortcomings become apparent. The ability to define values by clicking in cells, rather than via the EDEN window, would be useful, though would also be easy to implement.

More difficult to implement would be the ability to copy and paste relative references; for example, in Excel, if a cell C5 is equal to (C6 + C7), copying and pasting it to D5 will set that cell to (D6 + D7). This would be difficult to implement in EDEN, though might be possible through judicious use of the "execute()" function.

Another, related, feature from Excel is the ability to specify dependencies on ranges of cells; in particular, the ability to insert rows and columns in these ranges and have the references automatically update. This would likely be very difficult to implement in EDEN.

4 Evaluation

Spreadsheets are often used as an example of an alternative model of computing with similarities to Empirical Modelling, especially in the role dependencies play in both. However, the practical uses of spreadsheets are often fairly simple, and do not really on much "computation" as it might ordinarily be defined.

In attempting to use EDEN for a task where a spreadsheet would often be used, I found I had little motivation to use the more advanced language features of EDEN, and missing some of the more trivial features of spreadsheets. This could be a limitation of both EDEN and my understanding of it; perhaps EDEN is not as good as a traditional spreadsheet for this particular usage, or perhaps my conceptual understanding of the language was not sufficient for me to make full use of the possibilities.

Acknowledgements

I would like to thank Meurig Beynon for graciously allowing me to change the topic of my project midway through. In addition, I would like to thank my sister (aged nine), who gave some useful advice when I claimed to have too much work to do. ("Do your work then").

References

[1] Ries, E, *The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses*. Random House, New York, 1st Edition, 2011.

[2] Beynon, M, Rasmequan, S, Russ, S, A New Paradigm for Computer-Based Decision Support, University of Warwick. [Online] (URL http://www2.warwick.ac.uk/fac/sci/dcs/research/em/publications/papers/downloads/061.pdf)

[3] Kawasaki, G, Financial Tools for Pitching a Business [Online - Spreadsheet] (URL http://blog.guykawasaki.com/2012/01/how-to-create-an-enchanting-financial-forecast-officeandguyk.html)

[4] Ries, E, Business Model Canvas [Online - Spreadsheet] (URL http://leanfounder.com/2011/04/30/startup-week-13-business-model-spreadsheet/)

[5] Spolsky, J, How Trello is Different [Online] (URL http://www.joelonsoftware.com/items/2012/01/06.html)