Feature or Project Growth Assumption Canvas

Time Based Growth

The longer we go the more alterations to original scope get added.

What is time based growth?
The longer between committing to delivery and actual delivery the higher risk feature requirements will change. These changes can be additional ideas to current plans, or reactions to a competitive market change. This type of growth isn't bad, it's just inconvenient from a forecasting perspective. Given that companies we work for need to be agile in a business sense, means that adapting to changes like this is an important part of doing software development.

1. How much more scope?

Release Frequency	Technically Easy	Hard
Cont. – 2 weeks	1x	1.25x
3 – 6 weeks	1.25x	1.5x
7 – 12 weeks	1.5x	1.75x
13 – 26 weeks	1.75x	2x
26+ weeks	2x	4x

Suggested actions -

- Releasing more frequently limits exposure to time growth
- Don't try and eliminate ALL time based growth. Some is healthy, it means recently learnt lessons are headed..
- Bent Flyberg is a good resource of material on Mega-projects and recommends avoiding Mega-projects altogether!

Rate Based Growth

The more work we complete the more we learn about what we need to do to deliver.

What is rate based growth?
This type of growth comes from actually completing work. Defects discovered whilst completing fall into the rate based growth category. Any growth in story count that has a relationship to completed work count falls into this category.

To capture these, have the team brainstorm items that performing work might cause new discovered work. Keeping an ongoing list of items like this from prior projects helps do this more accurately next time.

3.Things we might need to do

Growth due to...	Occurr.	# Stories
E.g. Defects	100%	1-3
E.g. Localization	20-30%	3-4

Suggested actions -

- Create a rate based growth table for your feature and projects
- Consolidate the rate based growth knowledge across other teams, limit to top 10
- Use actual data to refine the occurrence rate estimate and the impact for growth item**s.**

Scale Based Growth

The size of completed work is different than the work in our backlog. E.g. work splits

What is scale based growth?
This is the most overlooked growth of work. It is more properly a correction rather than growth. It is caused when the pace of delivered items is assumed to be the pace remaining backlog items will be completed. Why isn't this true? Commonly work is split into multiple items when the team is analyzing the details just prior to adding them to their sprint or pulling that work into the team. Left un-adjusted, it looks like the team will complete faster than they will.

3a. Low guess: Items in the original backlog become x items in complete?
1 2 3 4 5 other:
(default)

3b. High guess: Items in the original backlog become y items in complete?
1 2 3 4 5 other:
(default)

Ask the team what is the lowest and highest likely split rate.

Suggested actions -

- Be alert to anytime the backlog count is combined with historical data; The result will often be optimistic.
- Start with the estimate range of 1 to 3 times.
- Measure what the actual rate is using historical data or by observation during planning meetings.

Event Based Growth

Feedback or things that go wrong in the approval to release process.

What is event based growth?
Sometimes we have a hint that something might go wrong requiring something else to be done, but can't be sure. For example, sometimes you build a feature but have little insight to how it will perform with thousands of users. You can guess you have the right architectural approach, but until its built and tested under stress, you can't know for certain that it isn't going to need improvement. This is an event based scope increase. And it's my opinion these are the make or break of good software forecasts.

4.Things we might need to do

Risk	Prob.	# Stories
E.g. Performance	50-75%	20-30

Suggested actions -

- Pick the top five. If you have more than this, forecasting is pointless, you are guessing. Don't start!
- There will ALWAYS be a few un-avoidable risks. Do these earlier in the project to avoid late surprises..
- Don't spend all your time on total risk avoidance, accept some will occur and forecast accordingly.

For more probabilistic forecasting resources: FocusedObjective.com Email: troy.magennis@focusedobjective.com