

## “So what exactly is this #NoEstimates movement?”

**April 2015**

### **Scope of this Report**

Estimation is one of the lightning rod issues in software development and maintenance. Over the past few years the concept of #NoEstimates has emerged and has become a movement within the Agile community. Due to its newness, #NoEstimates has several camps revolving around a central concept of not generating task level estimates. The newness of the movement also means there are no (or very few) large example projects that can be used as references<sup>1</sup>. Finally there are no published quantitative studies of results comparing the results of work performed using #NoEstimates techniques to other methods. In order to have a conversation we need to begin by establishing a shared context and language across the gamut of estimating ideas whether Agile or not. Without a shared language that includes #NoEstimates we will not be able to compare the concept to classical estimation concepts.

### **Context**

#### **#NoEstimates Context:**

There are two main groups or camps of thought leaders in the #NoEstimate movement (the two camps probably reflect more of a continuum of ideas rather than absolutes). The first camp argues that a team should break work down into small chunks and then immediately begin completing those small chunks (doing the highest value first). The chunks would build up quickly to a minimum viable product (MVP) that can generate feedback, so the team can hone its ability to deliver value. This camp leverages continuous feedback and re-planning to guide work, and luminaries like Woody Zuill often champion this camp. A second camp begins in a similar manner – by breaking the work into small pieces, prioritizing on value (and perhaps risk), delivering against a MVP to generate feedback – but they measure throughput. Throughput is a measure of how many units of work (e.g. stories or widgets) a team can deliver in a specific period of time. Continuously measuring the throughput of the team provides a tool to understand when work needs to start in order for it to be delivered within a period time. Average throughput is used to provide the team and other stakeholders with a forecast of the future. This is very similar to throughput measured used in Kanban. People like Vasco Duarte champion the second camp who practice #NoEstimates from a lean or Kanban perspective. We recently heard David Anderson, the

Kanban visionary, discuss a similar #NoEstimates position using throughput as a forecasting tool. Both camps in the #NoEstimates movement eschew developing story- or task-level estimates. The major difference is on the use of throughput to provide forecasting which is central to bottom-up estimating and planning at the lowest level of the classic estimation continuum.

### **Classic Estimation Context:**

Estimation as a topic is often a synthesis of three related, but different concepts. The three concepts are budgeting, estimation and planning. Because these three concepts are often conflated it is important to understand the relationship between the three. These are typical in a normal commercial organization, however these concepts might be called different things depending on your business model.

An estimate is a finite approximation of cost, effort and/or duration based on some basis of knowledge (this is known as a basis of estimation). The flow of activity conflated as estimation often runs from budget, to project estimation to planning. In most organizations, the act of generating a finite approximation typically begins as a form of portfolio management in order to generate a budget for a department or group.

The budgeting process helps make decisions about which pieces of work are to be done. Most organizations have a portfolio of work that is larger than they can accomplish, therefore they need a mechanism to prioritize. Most portfolio managers, whether proponents of an Agile or a classic approach, would defend using value as a key determinant of prioritization. Value requires having some type of forecast of cost and benefit of the project over some timeframe. Once a project enters a pipeline in a classic organization, an estimate is typically generated. The estimate is generally believed to be more accurate than the original budget due to the information gathered as the project is groomed to begin.

Plans breakdown stories into tasks often with personnel assigned, an estimate of effort generated at the task level and sum the estimates into higher-level estimates. Any of these steps can (but should not) be called estimation. The three -level process described above, if misused, can cause several team and organizational issues. Proponents of the #NoEstimates movement often classify these issues as estimation pathologies. Jim Benson, author of *Personal Kanban*, established a taxonomy of estimation pathologies<sup>ii</sup> that includes:

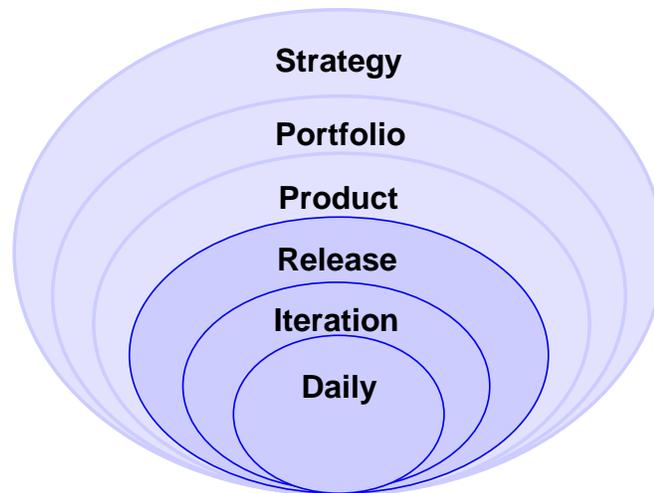
1. Guarantism – a belief that an estimate is actually correct.
2. Swami-itis – a belief that an estimate is a basis for sound decision making.
3. Craftosis – an assumption that estimates can be done better.

4. Reality Blindness – an insistence that estimates are prima facie implementable.
5. Promosoriality – a belief that estimates are possible (planning facility)

Estimates by definition are imprecise and can only be accurate within a range of confidence however these facts are often “forgotten” in lieu of the single number contract. Acting as if any of these pathologies are true has generated the anger and frustration needed to fuel the #NoEstimates movement.

When done correctly, both #NoEstimates and classic estimation are tools to generate feedback and create guidance for the organization. In its purest form #NoEstimates uses functionality to generate feedback and to provide guidance about what is possible. The less absolutist “Kanban’er” form of #NoEstimates uses both functional software and throughput measures as feedback and guidance tools. Classic estimation tools use plans and performance to the plan to generate feedback and guidance. The goal is usually the same, it is just that the mechanisms are very different.

## **Budgeting, Estimation, Planning, #NoEstimates and the Agile Planning Onion**



Source: Mike Cohn, [www.mountangoatsoftware.com](http://www.mountangoatsoftware.com)

There are many levels of estimation including budgeting, high-level estimation and task planning (detailed estimation). We can link a more classic view of estimation to the “Agile Planning Onion” popularized by Mike Cohn. In the Agile Planning Onion strategic planning is on the outside of the onion and the planning that occurs in the daily sprint meetings at the core of the onion. Budgeting is

a strategic form of estimation that most corporate and governmental entities perform. Other than in its most extreme form, budget is generally not a practice being eschewed by #NoEstimate proponents. Estimation exists in the middle layers of the Agile Planning Onion (product and release layers). In classic estimation, these estimates are often developed using top down techniques such as analogy or parametric estimation using function points, story points or tee-shirt sizing. #NoEstimate proponents leveraging Kanban techniques perform this level of estimates as forecasts using average flow rates and queuing theory (an application of Little's Law). The resistance at this level has generated the perception that sized based estimation at this level (and later, planning at the task level) generate several pathological behaviors within organizations. The final layers of the planning onion, iteration and daily planning, are generally the areas of highest concern to the #NoEstimates movement. While tasks may be identified, effort is not assigned.

It should be noted that while effort estimates are not done at the planning layers or generally at the estimation layer most teams adopt rules to break work down into predictable units. Rules or guidelines are often established that affect story and task size. The used of rules to govern granularity is one of the reason flow measures can be used to forecast when work needs to begin in order to meet date or dependency requirements. Johanna Rothman stated in her article "The Case for #NoEstimates," that "when you deliver small, valuable chunks of work every day or more often" that you can avoid estimation. The critical words being small and everyday which requires the team to understand how to groom stories to the desired granularity. Whether through the use of rules or feedback using these techniques to groom stories could easily be construed as a crude form of estimation.

## **Scenarios:**

### **Standard Corporate Environments:**

Organizational budgeting (strategy and portfolio): Continuous flow or other #NoEstimates techniques don't answer the central questions most organizations need to answer which include:

1. How much money should I allocate for software development, enhancements and maintenance?
2. Which projects or products should we fund?
3. Which projects will return the greatest amount of value?

While most budgets are scientific guesses there is a need to understand at least some approximation of the size and cost of the work on the overall backlog.

### **High Level Estimation (product and release):**

Release Plans and product road maps could easily be built from forecasts based on teams that have a track record of delivering value on a regular basis. The idea of #NoEstimates can be applied at this level of planning and estimation IF the right conditions are met. Conditions include:

1. Stable Teams
2. Agile Mindset (both team and organizational levels)
3. Well-groomed stories

The classic questions of when, what and how much can be answered in this environment for work done by single teams or by scaled Agile programs.

It should be noted that the example used by Woody Zuill's, which uses the most form of #NoEstimates (start, deliver, get feedback, and then do more) is a reflection of an environment where all of these factors are reflected.

### **Task Level Estimation (iteration and daily):**

Task level planning is the base of #NoEstimates discussions. Stable teams that are able consistently to accept and deliver what is expected do not have any need to plan effort at a task level.

### **Commercial / Contractual Work:**

Raja Bavani, Senior Director at Cognizant Technology Solutions stated in a recent conversation, that he thought that #NoEstimates was a non-starter in a contractual environment.

### **Conclusion**

Estimation is a form of planning. Planning is a considered an important competency in most business environments. Planning activities abound whether planning the corporate picnic to planning the acquisition and implementation of a new customer relationship management system. Most planning activities center on answering a few very basic questions. When will "it" be done? How much will "it" cost? What is "it" that I will actually get? Rarely does the question of how much effort will it take get asked unless as proxy for how much will it cost. As the work progresses the questions shift to whether we are going to meet the date, budget or scope. Answering those questions can be accomplished by any number of techniques. Using #NoEstimates techniques still requires most organizations to budget. Using #NoEstimates techniques requires breaking down stories into manageable, predictable chunks so that teams can predictably deliver value. The ability to predictably deliver value provides organizations with the tool to forecast the delivery. #NoEstimates really isn't not estimating . . . it is just estimating differently.

## Sources

---

<sup>i</sup> The C3 Project was used to hone and prove many of the Agile techniques (eXtreme Programming and WIKIs for example) and acted as a training ground for many luminaries of the early Agile movement.

<sup>ii</sup> [http://herdingcats.typepad.com/my\\_weblog/2015/03/five-estimating-pathologies-and-their-corrective-actions.html](http://herdingcats.typepad.com/my_weblog/2015/03/five-estimating-pathologies-and-their-corrective-actions.html) 4/27/15 or <http://moduscooperandi.com/blog/modus-list-3-our-five-estimate-pathologies/> 4/27/15