Part Three  
  
ACCELERATE

İçindekiler

[Jargon 2](#_Toc40075766)

[decisions startups face 2](#_Toc40075767)

[clear-cut decision 2](#_Toc40075768)

[release a product 2](#_Toc40075769)

[How often should you release a product? 2](#_Toc40075770)

[Releasing often vs releasing too later 2](#_Toc40075771)

[infrastructure and planning 2](#_Toc40075772)

[*anticipation* of success 2](#_Toc40075773)

[efficiency point of view 2](#_Toc40075774)

[learning 3](#_Toc40075775)

[early success 3](#_Toc40075776)

[cede market leadership 3](#_Toc40075777)

[Fast follower 3](#_Toc40075778)

[combat the extreme uncertainty that is a startup’s chief enemy 3](#_Toc40075779)

[activities create value 3](#_Toc40075780)

[activities create waste 3](#_Toc40075781)

[ultimate value of start-ups 3](#_Toc40075782)

[Ultimate goal of lean start-up 4](#_Toc40075783)

[Ultimate goal of lean start-up 2 4](#_Toc40075784)

[Sustainable growth 4](#_Toc40075785)

[the metrics startups should use 4](#_Toc40075786)

[understand their growth as they add new customers and discover new markets. 4](#_Toc40075787)

[Sustainable growth follows one of three engines of growth: 4](#_Toc40075788)

[adaptive organization 4](#_Toc40075789)

[Five Whys 4](#_Toc40075790)

[bureaucratic 4](#_Toc40075791)

[dysfunctional 4](#_Toc40075792)

[unite 9 4](#_Toc40075793)

[batch 4](#_Toc40075794)

[single -piece flow 5](#_Toc40075795)

[doğru olan 5](#_Toc40075796)

[Yanlış olan 5](#_Toc40075797)

[Batch size 5](#_Toc40075798)

[economies of scale 5](#_Toc40075799)

[large specialized machines vs smaller general-purpose machines 5](#_Toc40075800)

[right part at the right time. 5](#_Toc40075801)

[SMED (Single-Minute Exchange of Die 6](#_Toc40075802)

[smaller batch size (faydası) 6](#_Toc40075803)

[Andon 6](#_Toc40075804)

[allows any worker to ask for help as soon as they notice any problem,  production line if it cannot be corrected immediately. 6](#_Toc40075805)

[validated learning 6](#_Toc40075806)

[Lean Startup goal 6](#_Toc40075807)

[Apple karşı örneği 6](#_Toc40075808)

[our product’s immune system and continuous deployment 7](#_Toc40075809)

[SMALL BATCHES IN ACTION 7](#_Toc40075810)

[large-batch death spiral 9](#_Toc40075811)

[bet the company 9](#_Toc40075812)

[ship the product 9](#_Toc40075813)

[a “hole” in the dealer’s inventory 10](#_Toc40075814)

[pull and just-in-time production 10](#_Toc40075815)

[of just-in-case inventory [called work-in-progress (WIP) inventory] 10](#_Toc40075816)

[start-up WIP 10](#_Toc40075817)

[Lean Startup technique lerin temeli 10](#_Toc40075818)

[hypothesis about the customer 11](#_Toc40075819)

[Leap-of-faith assumptions 11](#_Toc40075820)

[early adopters 11](#_Toc40075821)

[pivot 11](#_Toc40075822)

[customer segment pivots 11](#_Toc40075823)

[GROW 11](#_Toc40075824)

[English 13](#_Toc40075825)

[Clear-cut: 13](#_Toc40075826)

[cede 13](#_Toc40075827)

# Jargon

## Decisions that startups face

## clear-cut decision

doğru seçeneği belirgin karar verme zihindurumu

## release a product

## How often should you release a product?

## Releasing often vs releasing too later

releasing often leaves less time to devote to building the product. However, waiting too long to release can lead to the ultimate waste: making something that nobody wants.

## infrastructure and planning

## anticipation of success

başarı beklentisi

## efficiency point of view

## learning

How much time and energy should companies invest in infrastructure and planning early on in anticipation of success? Spend too much and you waste precious time that could have been spent learning.

## early success

How much time and energy should companies invest in infrastructure and planning early on in anticipation of success? Spend too much and you waste precious time that could have been spent learning. Spend too little and you may fail to take advantage of early success and cede market leadership to a fast follower.

## cede market leadership

## Fast follower

Startup ürünümüzü kopyalamaya çalışanlar

## combat the extreme uncertainty that is a startup’s chief enemy

## activities create value

## activities create waste

## ultimate value of start-ups

that value in a startup is not the creation of stuff, but rather validated learning about how to build a sustainable business.

* What products do customers really want?
* How will our business grow? Who is our customer?
* Which customers should we listen to and which should we ignore?

These are the questions that need answering as quickly as possible to maximize a startup’s chances of success. That is what creates value for a startup.

## Ultimate goal of lean start-up

grow without sacrificing the speed and agility that are the lifeblood of every startup.

## Ultimate goal of lean start-up 2

Lean Startups practice just-in-time scalability, conducting product experiments without making massive up-front investments in planning and design.

## Sustainable growth

## the metrics startups should use

## understand their growth as they add new customers and discover new markets.

## Sustainable growth follows one of three engines of growth:

paid, viral, or sticky. By identifying which engine of growth a startup is using, it can then direct energy where it will be most effective in growing the business.

## adaptive organization

## Five Whys

## bureaucratic

## dysfunctional

# unite 9

## batch

## single -piece flow

mektup işleminde diyelim kağıt koyma, katlama , pul yapıştırma gibi 3 adım var.

### doğru olan

bir mektubu al ve hallet sonrakine geç.

### Yanlış olan

Hepsine kağıtkoy, sonra hepsini katla sonra hepsine pul

It works because of the surprising power of small batches. When we do work that proceeds in stages, the “batch size” refers to how much work moves from one stage to the next at a time. For example, if we were stuffing one hundred envelopes, the intuitive way to do it—folding one hundred letters at a time—would have a batch size of one hundred. Single-piece flow is so named because it has a batch size of one.

**Why** does stuffing one envelope at a time get the job done faster even though it seems like it would be slower? Because our intuition doesn’t take into account the **extra time** required to **sort**, **stack**, and **move** around the large piles of half-complete envelopes when it’s done the other way.

## Batch size

## economies of scale

## large specialized machines vs smaller general-purpose machines

Instead of buying large specialized machines that could produce thousands of parts at a time, Toyota used smaller general-purpose machines that could produce a wide variety of parts in small batches.

## right part at the right time.

## SMED (Single-Minute Exchange of Die

A setup reduction and quick changover system, designed to bring the changover time for a machine to less than 10 minutes. This minimizes the time that a process is down while being changed from one product to another.

## smaller batch size (faydası)

Because of its smaller batch size, Toyota was able to produce a much greater diversity of products. The biggest advantage of working in small batches is that quality problems can be identified much sooner.

## Andon

## allows any worker to ask for help as soon as they notice any problem,  production line if it cannot be corrected immediately.

## validated learning

## Lean Startup goal

in the Lean Startup the goal is not to produce more stuff efficiently. It is to—as quickly as possible—learn how to build a sustainable business.

## Apple karşı örneği

Bundled up in that product release are dozens of new features (at the release of iPhone 4, Apple boasted more than 1,500 changes).

Ironically, many high-tech products are manufactured in advanced facilities that follow the latest in lean thinking, including small batches and single-piece flow. However, the process that is used to design the product is stuck in the era of mass production. Think of all the changes that are made to a product such as the iPhone; all 1,500 of them are released to customers in one giant batch.

Behind the scenes, in the development and design of the product itself, large batches are still the rule. The work that goes into the development of a new product proceeds on a virtual assembly line. Product managers figure out what features are likely to please customers; product designers then figure out how those features should look and feel. These designs are passed to engineering, which builds something new or modifies an existing product and, once this is done, hands it off to somebody responsible for verifying that the new product works the way the product managers and designers intended.

# our product’s immune system and continuous deployment

When our immune system detects a problem, a number of things happen immediately:

1. The defective change is removed immediately and automatically.

2. Everyone on the relevant team is notified of the problem.

3. The team is blocked from introducing any further changes, preventing the problem from being compounded by future mistakes …

4. … until the root cause of the problem is found and fixed. (This root cause analysis is discussed in greater detail in Chapter 11.)

## SMALL BATCHES IN ACTION

To see this process in action, let me introduce you to a company in Boise, Idaho, called SGW Designworks. SGW’s specialty is rapid production techniques for physical products. Many of its clients are startups.

SGW Designworks was engaged by a client who had been asked by a military customer to build a complex field x-ray system to detect explosives and other destructive devices at border crossings and in war zones.

Conceptually, the system consisted of an advanced head unit that read x-ray film, multiple x-ray film panels, and the framework to hold the panels while the film was being exposed. The client already had the technology for the x-ray panels and the head unit, but to make the product work in rugged military settings, SGW needed to design and deliver the supporting structure that would make the technology usable in the field. The framework had to be stable to ensure a quality x-ray image, durable enough for use in a war zone, easy to deploy with minimal training, and small enough to collapse into a backpack.

This is precisely the kind of product we are accustomed to thinking takes months or years to develop, yet new techniques are shrinking that time line. SGW immediately began to generate the visual prototypes by using 3D computer-aided design (CAD) software. The 3D models served as a rapid communication tool between the client and the SGW team to make early design decisions.

The team and client settled on a design that used an advanced locking hinge to provide the collapsibility required without compromising stability. The design also integrated a suction cup/pump mechanism to allow for fast, repeatable attachment to the x-ray panels. Sounds complicated, right?

Three days later, the SGW team delivered the first physical prototypes to the client. The prototypes were machined out of aluminum directly from the 3D model, using a technique called computer numerical control (CNC) and were hand assembled by the SGW team.

The client immediately took the prototypes to its military contact for review. The general concept was accepted with a number of minor design modifications. In the next five days, another full cycle of design iteration, prototyping, and design review was completed by the client and SGW. The first production run of forty completed units was ready for delivery three and a half weeks after the initiation of the development project.

SGW realized that this was a winning model because feedback on design decisions was nearly instantaneous. The team used the same process to design and deliver eight products, serving a wide range of functions, in a twelve-month period. Half of those products are generating revenue today, and the rest are awaiting initial orders, all thanks to the power of working in small batches.

|  |  |
| --- | --- |
| **THE PROJECT TIME LINE** | |
| Design and engineering of the initial virtual prototype | 1 day |
| Production and assembly of initial hard prototypes | 3 days |
| Design iteration: two additional cycles | 5 days |
| Initial production run and assembly of initial forty units | 15 days |

## large-batch death spiral

Consider our hypothetical example. After passing thirty design drawings to engineering, the designer is free to turn his or her attention to the next project. But remember the problems that came up during the envelope-stuffing exercise. What happens when engineering has questions about how the drawings are supposed to work? What if some of the drawings are unclear? What if something goes wrong when engineering attempts to use the drawings?

These problems inevitably turn into interruptions for the designer, and now those interruptions are interfering with the next large batch the designer is supposed to be working on. If the drawings need to be redone, the engineers may become idle while they wait for the rework to be completed. If the designer is not available, the engineers may have to redo the designs themselves. This is why so few products are actually built the way they are designed.

When I work with product managers and designers in companies that use large batches, I often discover that they have to redo their work five or six times for every release. One product manager I worked with was so inundated with interruptions that he took to coming into the office in the middle of the night so that he could work uninterrupted. When I suggested that he try switching the work process from large-batch to single-piece flow, he refused—because that would be inefficient! So strong is the instinct to work in large batches, that even when a large-batch system is malfunctioning, we have a tendency to blame ourselves.

Large batches tend to grow over time. Because moving the batch forward often results in additional work, rework, delays, and interruptions, everyone has an incentive to do work in ever-larger batches, trying to minimize this overhead. This is called the large-batch death spiral because, unlike in manufacturing, there are no physical limits on the maximum size of a batch.

## bet the company

a “bet the company” new version of the product, because the company has taken such a long time since the last release.

## ship the product

## a “hole” in the dealer’s inventory

## pull and just-in-time production

Lean production solves the problem of stockouts with a technique called pull. When you bring a car into the dealership for repair, one blue 2011 Camry bumper gets used. This creates a “hole” in the dealer’s inventory, which automatically causes a signal to be sent to a local restocking facility called the Toyota Parts Distribution Center (PDC). The PDC sends the dealer a new bumper, which creates another hole in inventory. This sends a similar signal to a regional warehouse called the Toyota Parts Redistribution Center (PRC), where all parts suppliers ship their products. That warehouse signals the factory where the bumpers are made to produce one more bumper, which is manufactured and shipped to the PRC.

The ideal goal is to achieve small batches all the way down to single-piece flow along the entire supply chain. Each step in the line pulls the parts it needs from the previous step. This is the famous Toyota just-in-time production method.

## of just-in-case inventory [called work-in-progress (WIP) inventory]

## start-up WIP

Incomplete designs, not-yet-validated assumptions, and most business plans are WIP.

## Lean Startup technique lerin temeli

Almost every Lean Startup technique we’ve discussed so far works its magic in two ways: by converting push methods to pull and reducing batch size. Both have the net effect of reducing WIP.

Lean production vs lean startup

In manufacturing, pull is used primarily to make sure production processes are tuned to levels of customer demand. Without this, factories can wind up making much more—or much less—of a product than customers really want. However, applying this approach to developing new products is not straightforward. Some people misunderstand the Lean Startup model as simply applying pull to customer wants. This assumes that customers could tell us what products to build and that this would act as the pull signal to product development to make them.

As was mentioned earlier, this is not the way the Lean Startup model works, because customers often don’t know what they want. Our goal in building products is to be able to run experiments that will help us learn how to build a sustainable business. Thus, the right way to think about the product development process in a Lean Startup is that it is responding to pull requests in the form of experiments that need to be run.

## hypothesis about the customer

As soon as we formulate a hypothesis that we want to test, the product development team should be engineered to design and run this experiment as quickly as possible, using the smallest batch size that will get the job done. Remember that although we write the feedback loop as Build-Measure-Learn because the activities happen in that order, our planning really works in the reverse order: we figure out what we need to learn and then work backwards to see what product will work as an experiment to get that learning. Thus, it is not the customer, but rather our hypothesis about the customer, that pulls work from product development and other functions. Any other work is waste.

## Leap-of-faith assumptions

Leap-of-faith assumptions are those kind of assumptions that you take for granted, you don’t know if they’re right or wrong, you just go with them.

## early adopters

## pivot

## customer segment pivots

# GROW

# English

## Clear-cut:

kesin belirgin

## cede

devretmek, vazgeçmek