Perfection & Feedback Loops

or: why worse is better

Marcus Denker

Talk held at ESUG2016

The slides as used for the talk are a bit useless alone. This is a an **annotated version,** grey slides are additions (links, notes…)

- Video: https://youtu.be/LRFLdWG24Mk
- The .key file has all videos embedded

All Files:

http://marcusdenker.de/talks/16ESUG

VIDEO on Youtube:

https://youtu.be/LRFLdWG24Mk

Another Strange Talk

All two years I feel a strange urge to do a strange talk at ESUG…

no idea if that is a good idea…

Like ESUG 2014

At ESUG 2014 I give a similar talk, this one is kind of the same topic, but from another point of view.

See http://www.slideshare.net/MarcusDenker/2014-esugcathedral

2 years ago: Cathedrals


Toilets
FreeFoto.com

Toilets

Today

Perfection

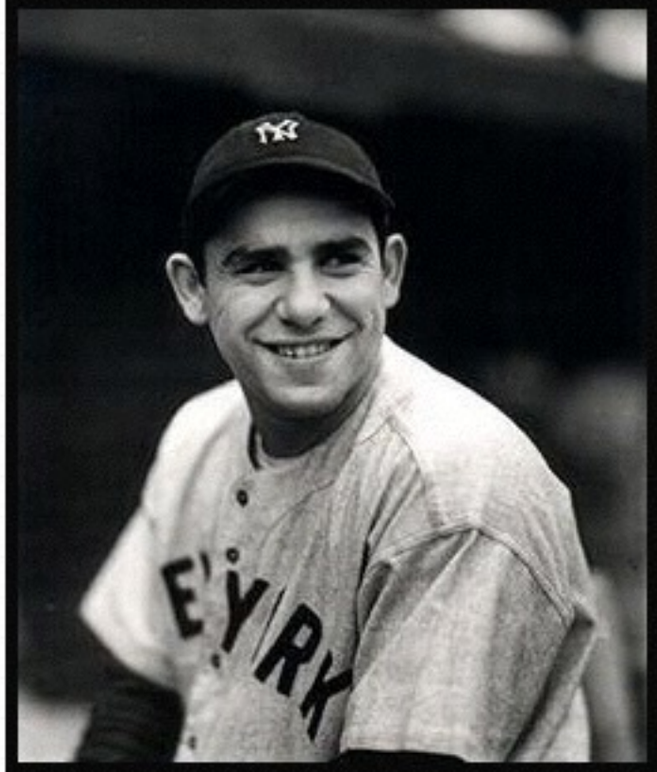
Feedback

Why everything existing sucks

And what to do about it

I have bad news

Perfection does not exist



If the world were perfect, it wouldn't be.

(Yogi Berra)

izquotes.com

Knowing his quotes from the fortune unix command,
I did not know that Yogi Berra was a baseball player…

"I really didn't say everything I said."

https://en.wikipedia.org/wiki/Yogi_Berra

“By the time you’ve arrived at the *perfect* solution, usually the problem has already changed.”

— Jessie Shefrin (@jshefrin)

Quote thanks to Jessie Shefrin
http://artthinking.com

Why is that?

Context

Context changes.

Would the perfect Programming Language and Environment for 1985 be seen as perfect in 2016?

And it goes even deeper than that. Context is everything.

The same solution can be good or bad, depending on the context it is embedded in.

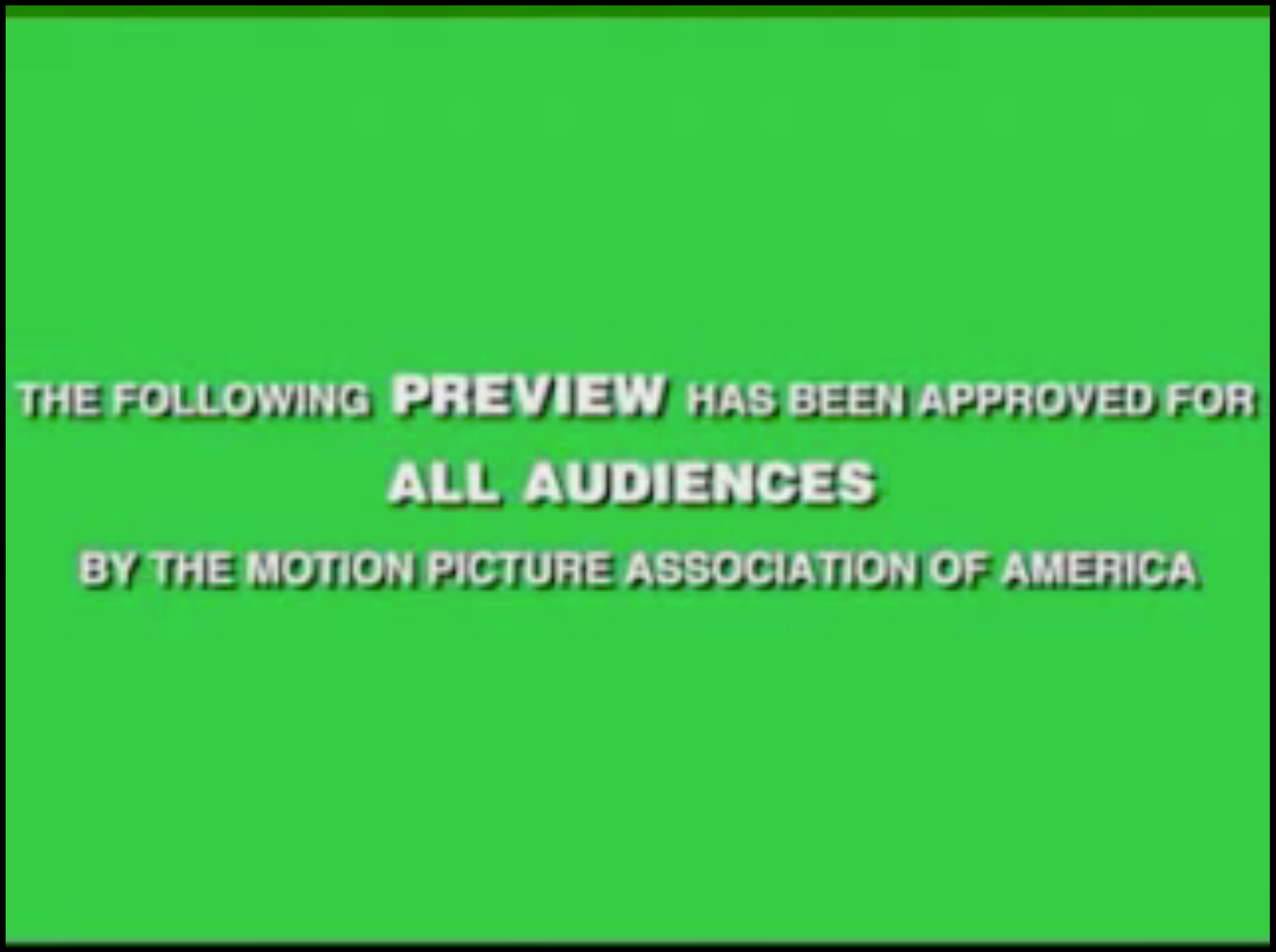
https://www.youtube.com/watch?v=HPPj6viIBmU

One of the first viral videos of the internet.

The video should never have been posted, it is often shown as an example of cyber bullying.

Lots of things can be learned, but what everyone agrees: “Perfection” is the last thing that comes into mind.

So you are seeing this…



https://www.youtube.com/watch?v=3GJOVPjhXMY

... but his context was this. And it is just perfect.

Context is everything.

And context changes,
constantly

You are not alone in the world.

“Everyone, stop, until I created the perfect thing!”

Another problem: You learn while building

When you are ready to finish, it will be obvious to you that what you did is all bad. You could do so much better… easily.

Now it is so obvious…

But the “old you” would love it. The old you did not yet learn what the “new you” knows.

Keep in mind: everyone else but you did not learn what makes you see your work as imperfect.

Everything you can *finish*
will "embody" its own critique.

… but to some extend others (and the old you) will see the problems, though. A finished artefact always embodies its own critique.

(If you can see flaws in everything other people do, maybe it is not because you are oh so clever?)

Ok, no perfection, but better?

So, ok. There is no perfection.

But why can’t things be at least good?

Those who could have done better where busy with building the perfect solution

…now they know how to do it right!
Just wait! It will be perfect!

But…

Yes: there is something else at play…

A force of incomprehensible to mankind

Exponential Growth

“The greatest shortcoming of the human race is our inability to understand the exponential function.”

— Al Bartlett

Watch on Youtube:
The Most IMPORTANT Video You'll Ever See
8 parts, 10 min each:

https://www.youtube.com/watch?v=F-QA2rkpBSY

... you “know” it in theory

(you might “know” everything in the this talk…
theoretically)

By the time that the fifth square is reached on the chessboard, the board contains a total of 31 grains of wheat

How big can it get? There are just 64 fields…

https://en.wikipedia.org/wiki/Wheat_and_chessboard_problem

… round 1,000 times the global production of rice in 2010
(464,000,000 metric tons)

Lily Pond

In a lake, there is a patch of lily pads. First day there is one, second two. After long 47 days it is half full.

How long does it take to cover the whole lake?

47 long days to do half.. will take a while for sure...

Compound interest

S&P 500, invest $100 monthly.
Start: 1983

https://dqydj.com/sp-500-dividend-reinvestment-and-periodic-investment-calculator/

not including taxes, but take fees into account.

Payed: 39700

Today: ?

We payed in not even 40K. How much is it now?

Payed: 39700

Today: **241036**

not including taxes, but take fees into account.

Interesting:

— It’s not about doubling. ~8%
(doubles every 70/8 years)

— Most value is created by reinvesting (feeding back) interest earned.

Examples for Feedback

Science

New theories and models makes ideas thinkable that are just not thinkable without.

Open Source

Example: Linux vs. Minix.

Minix was clearly better. But there was no feedback loop for Minix.

Linux was a *process*, while Minix was a finished artefact.

Processors

The first microprocessor was designed with paper + pen.

You could not design a current one without having already a computer.

(Going back to paper+pen… why not go back to sticks+stones?)

Another Example: LAM builds machines that build processors. These machines *contain* processors themselves.

Agile

Early results lead to fast feedback cycles…

Lean Startups, Minimum Viable Product…

A startup just does not have resources for building the perfect solution to a problem nobody cares about.

Very hard to explain to our profession… spawned a whole “self help book” section for Programmers.

Perfection and Feedback

Feedback loops do not care about perfection

The system needs only to be good enough to sustain the next step

If just barely good enough to sustain it, feedback will happen within technically horrible solutions.

“It will be perfect when it is finished”
vs.
Feedback Loop Now

But just wait till my perfect system is ready.. it will be so much better.

Really? Who will need it?

What can we we do?

1. Goal vs. System

From self help books: do not focus on the goal ("I want to loose weight”), but build instead a system that has your goal as its result eventually. It is *much* easier, the steps are smaller and manageable and even automatic…

We should understand our programming goals the same way…

http://www.iwillteachyoutoberich.com/blog/goal-setting-is-dead-do-this-instead/

"Do try to tackle goals. Build systems instead"

Understand your Artefact as a Feedback Loop

Every artefact exist in a loop. The most basic one can see when looking at economics: if you can sell a simple, imperfect, early version, you can invest all the result back in your product.
Compare that with working in your spare time…

Works especially well for

Meta Systems

Especially systems that are used to create other systems benefit from feedback loops.
Improve your IDE and you will be more productive.

Reflective Systems even more: They are implemented in themselves, so improving the system leads to a natural cycle.

Smalltalk can be feedback loop

Smalltalk is an example for a system where it is very obvious.

Yet most Smalltalkers argue to never change anything, to just build on top…

If you set the development and community up correctly

An open source smalltalk ignoring all community contributions, just as an example.

2. Accept Imperfection

Remember: You just need to drive the next iteration!

DONE IS BETTER THAN PERFECT

POSTER BROUGHT TO YOU BY YOUR FRIENDS AT THE FACEBOOK ANALOG RESEARCH LABORATORY

Facebook poster. What they mean is that inside a feedback loop, you reach perfection when you reach the point where the next iteration can build on it (aka DONE).

http://benbarry.com/project/facebook-propaganda-posters

3. Small Change Matters

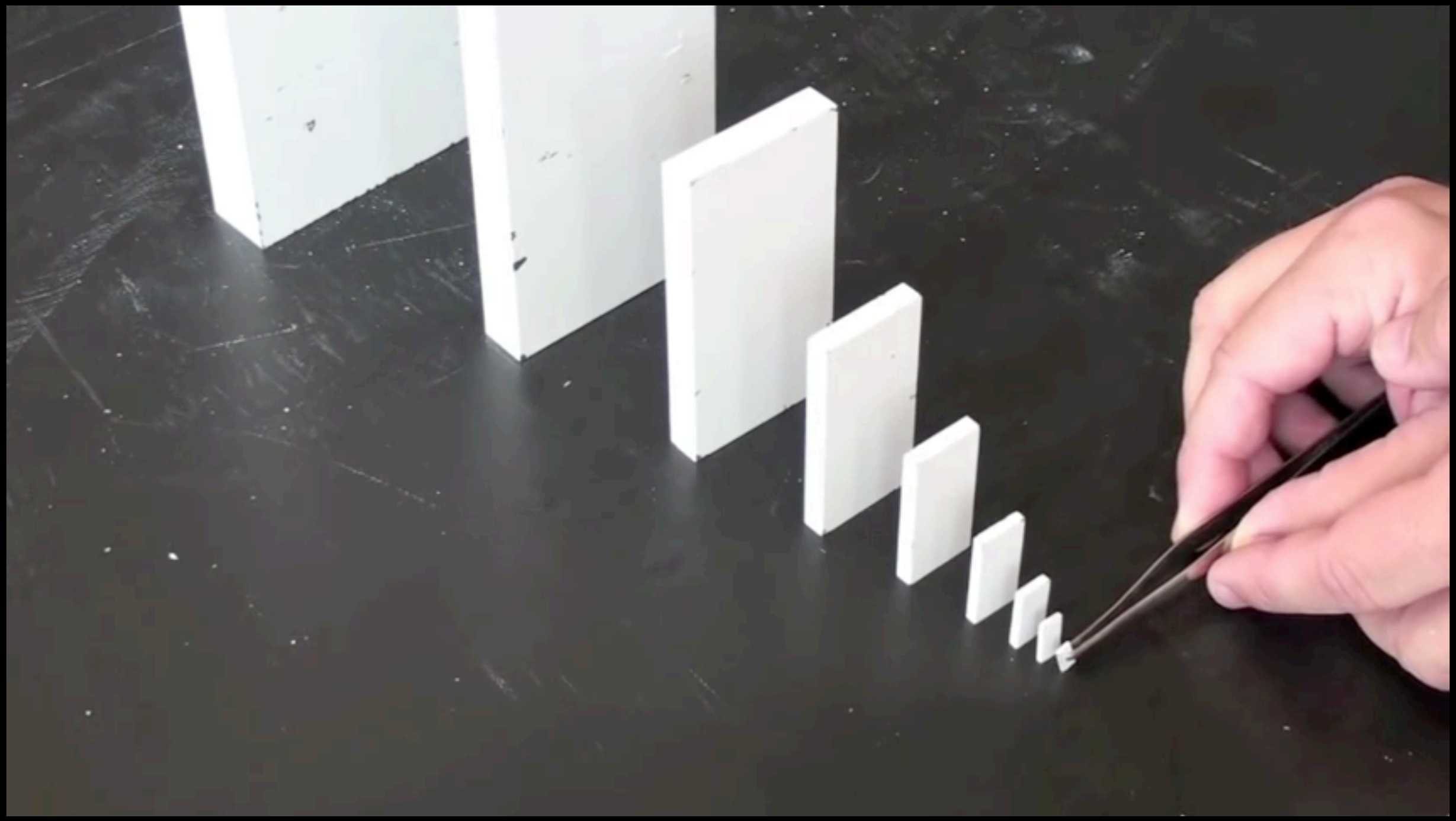
Remember: feeding back creates non-linear growth.

Even though you just get <10% interest, most of your account will be interest.

A small change fed back will have huge payout

That idiotic change will pay for 10, 20, 30 years…

A tiny linear change now would be a **huge** change some iterations ago



https://www.youtube.com/watch?v=y97rBdSYbkg

Why topple the small? To learn and fine-tune the system!

Peter Principle of the Domino

Peter Principle: “that tiny one? I do the next one, more interesting!”

Until you reach the one too large.

Goals vs. Systems:

It's a musical thing

Focus on the Process, not the Goal

Do not postpone your life to the future
“when you are finished”

Life and Music

https://www.youtube.com/watch?v=ERbvKrH-GC4

The video has a slightly different angle…

the one shown in the talk has the middle part cut.

https://www.youtube.com/watch?v=ERbvKrH-GC4

Thank You!

Twitter: @marcusdenker

http://marcusdenker.de