

Agilistas are killing the planet

is Agile, Scrum, kanban or lean,
really solving our problems?

or are they just helping us build
the wrong thing righter?

rather than achieving success
from hitting 'key milestones', the
Agilistas were delivering
'working software'

in fact, all they did was push
more through the system faster

on time and on budget are
meaningless if you build the
wrong thing, at the wrong time,
for the wrong people.

agile uses a machine gun

you take more shots, and increase
your odds of hitting the target

but how many innocent bystanders
get hit along the way, making it hard
to know *which* target was the right
one?

I just delivered 35 features on time
and on budget!

that's great but did anyone use it?

Did you make more money, help
more people? And why 35

features? Could you have gotten
the same results with 3?

development accounts for 30% of the cost. Are we expending 70% of our effort for only 30% reward?

stop building the wrong thing
righter

build the right thing

build
outcomes over outputs

Target Outcomes Framework

outcomes are the destination
you want to get to

options are the potential ways
you get there

how to get started

understand what's *really* going on

gather data;
it lies (a little) less than people

become the method actor of
user experience

never ask your users what they
want. never ask the developers
what they *think* the users want

“If I had asked people what they wanted, they would have said faster horses” Henry Ford

get out of your chair and go find
out what they *need*

we create empathy, we build
personas to remind us that real
people use our products

each user has a goal; a job to get
done

create strategies for getting to the
right people at the right time

once you know where you are,
figure out where you want to be

impact mapping;
choose the right outcomes by
measuring the severity of the
problem or size of the
opportunity

set target outcomes

target outcomes are the poke
yoke of product development

they make it easy to build the
right thing, and hard to build the
wrong thing

target outcomes need to be clear
and measurable

if you don't measure how will you
know when you get there?

target outcomes set a direction, and
there are many ways to achieve it

when you plan based on outputs,
you commit to a certain future

stop pretending you can see the
future

create options

keep your options open, be
ready to execute rapidly

It's about arrival not survival of the
fittest

*survival is the slowest form of
suicide*

outcomes and options are fractal

each one leads to another

simple right?

now simply go and build the right
thing

but how do we know what the
'right' thing is?

we don't. we have no [naughty
word] clue.

stop guessing and start testing

run multiple options fast;
make the unknowns known

set based design;
run options concurrently to
increase speed to market

arrival of the fittest, not the
survival of the fittest

who adapts first not fastest, fast
doesn't help if you get to the
wrong place

get in front of your users early
and often

seeing is believing, research is too
important to leave up to the
researchers

ignore your mother,
be fast, dirty and cheap.

features are not the experience

incremental feature farming
leads to Frankenbuilds

remove the inessential so the
essential can speak

the game is to hit the target by firing as *few* bullets as possible

build the minimal viable, not
maximum possible

measure as you go

setup automated metrics
frameworks

your metrics lag, you lag

the customer should be able to
reach in and grab what they need
when they need it

on demand;
15 minute builds, from concept to
cash

why test with 5 people in a room
when you can have thousands test
real time

transparency is key;
share the data with the entire
development team

progress indicators tell you if you
are on track

target outcomes validate the
investment

how do we know we have
succeeded?

we get to where we are going, if we
are not hitting the target outcomes,
adapt

sounds good but does it really
work?

which sounds better:

‘our velocity is 46 points and we
have delivered 100 new features’

or:

‘we just implemented one feature in three days which will save us £12 million pounds over the next year’

what industries can do this?

we are currently using it with a large
energy trading company, a video
gaming company and a VOIP telco

it's not just changing development,
it's fundamentally changing the
systems and structures around us

even our legal contracts need to
move to a target outcomes model

remember where we started?
'Agilistas are killing the planet'

when you use the machine gun
approach you build more
features faster

the more you features you build
the more resources you consume

save the features,
save the planet

Just because doesn't mean it has
value

Be careful not to focus on
throughput, this changes the game

Every option is an experiment

Transparency; letting people
know what the numbers mean

be lazy

Keep your options open

4. Deliver and adapt

Build less, build the right thing.

If you are building the wrong thing
faster sure you fail faster, but you
are still failing.

Target Outcomes are driven by
complexity theory

Survive and breathe or evolve and
fly.

Outcomes communicate and
align

the more you build, the more
resources you use

Outcomes over outputs

prove it don't plea it

Stop focusing on the process,
focus on the product.

let the data talk for you

User research is too important to
leave to the researchers

You create target outcomes by
measuring the starting point, then
setting a target

build

options and run experiments

Goal mapping – different people's
views of what's important on the
journey

don't waste time building a
ferrari

prototype; means you *can* throw
it away

by setting the constraints, we know
where we need to get to, no more,
no less

The Goldilocks principle:

Build what you need as you need it,
no more and no less – just right.