SCUTTLE Community Guide

REVISION: 2022.10.31

In This Guide:

VISION, VALUES, & PRINCIPALS (A WORK IN PROGRESS)

VISION: SCUTTLE Project

The what:

The vision is to create a multidisciplinary technology space that enables users at all areas (education, research, industry) to create new robotics outcomes more effectively than ever before.

To leverage the open-source revolution for more rapid exchange of new ideas.

To exclusively reside in digital manufacturing space for easier iterations, and higher quality prototyping, and lower-effort development.

Allow hardware development to become more Agile, as software has become. Give access to more spaces, (for example all languages) by digitizing, more time-zones by sharing on self-service platforms.

The how:

Mechanical

- Make mechanical more like software:
  - Parameters are self evident.
  - Digital designs – drive down post-processing by solving problems in the design stage.
  - Unify design and manufacturing
- Source OTS components that are affordable, accessible, and conform to popular standards
- Use DFM strategies and create accessible models: parametric, modular, easy to modify.

Software

- Write functional modules with less interdependency on other elements; allow developers to work on pieces without breaking the system.
- High fidelity nodes: designers take ownership of these nodes and keep them maintained: org makes it worthwhile for the designer.
- Beginner nodes: simpler software, detailed documents, examples provided.

Electronics

- OTS: conform to popular standards, affordable, and compliant with common tools.
- Hi-fi: make designs using industry-standard design tools.
- Educational fidelity: make designs using education-ware. Share designs on established platforms.

Quality

- Quality becomes parametric by absorbing quality into designs.
- Quality becomes automated by pushing design features into machine functions.
- Quality improvement becomes self-implementing by tying performance ceiling to standard-quality machines.

VISION: SCUTTLE PROJECT

We Believe:

Everyone is a learner. Now that technology is evolving at unprecedented rates, the only creators who are relevant are the creators who are learning in parallel with building.

We cannot create technology in a silo, and the disciplines which were once segregated are not interdependent.

Larger value returns will express from creating content that serves the other disciplines than content that is refined only in one space.

Therefore, everyone is a manufacturer. Designers have peak responsibility for manufacturing. Sometimes the manufacturer is just a machine.

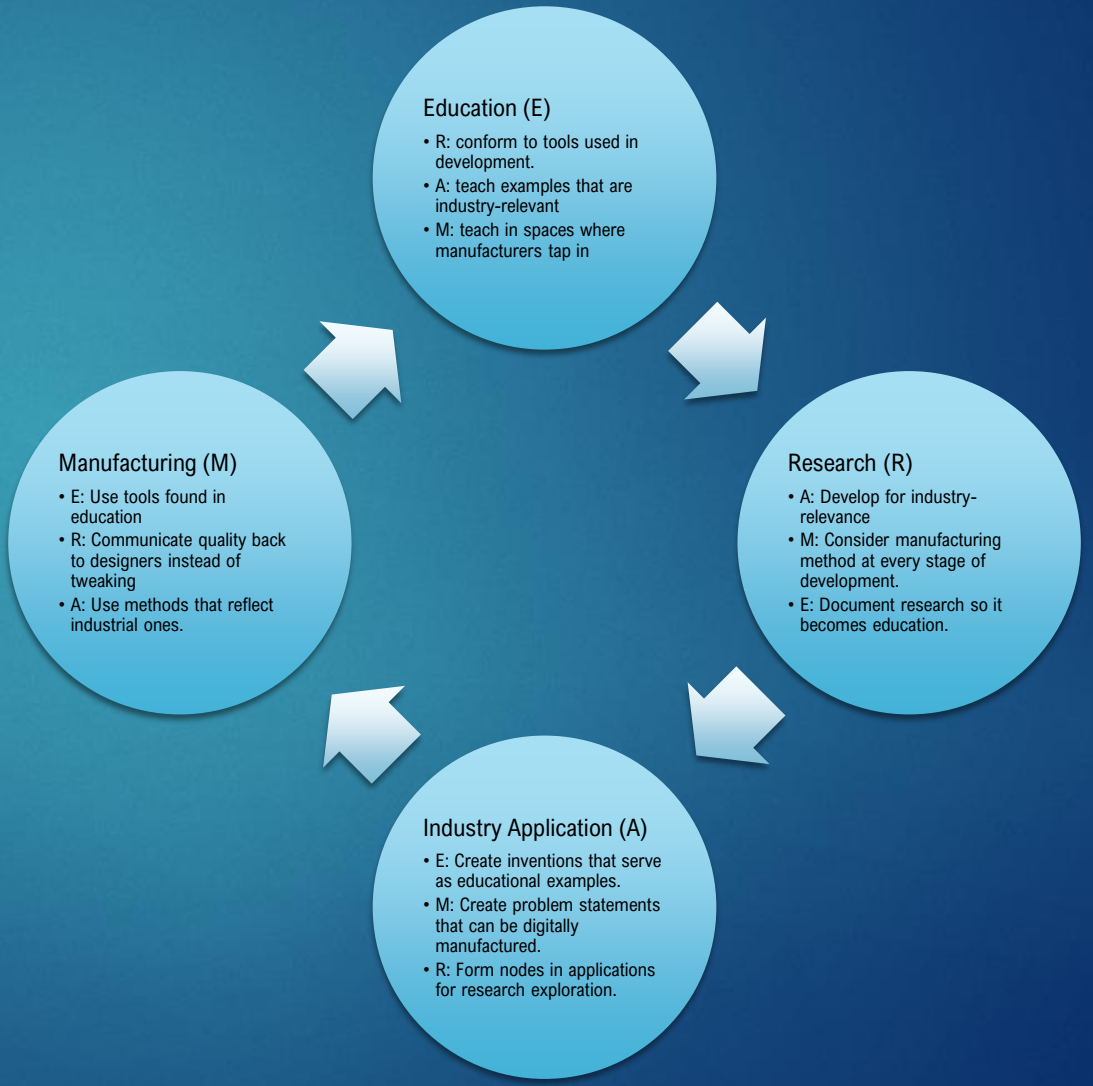
Everyone is a researcher. The least-experienced are becoming the first adopters of latest technologies; they have the most to gain by trial-and-error and they have the least to lose by testing new methods. This means each person with inexperience becomes a researcher.

Open source will ultimately become the only source.

The mission cannot be achieved in one organization; community is necessary.

Application, in time, will be only a byproduct. Within our lifetimes, the needs of consumers and industry will change so frequently that companies will become horizontally integrated across industries. NASA/agriculture will be unified. Healthcare/art will be unified. Construction/software will be unified.

How each facet serves the others:



VISION: Open Source

"Open Source is a secret sauce that repels zero-sum thinkers."

-SCUTTLE Robotics LLC

VISION: Open Source

Challenges of Open Source

Community	• Must have retention • Must have engagement
Contribution	• The community must feedback into the project • Contributions to other projects may be converted into SCUTTLE contributions
Communication	• New information must be organized to perfectly meet needs of the audience • The framing of the offering is everything.

VISION: Open HARDWARE

The Backstory

It started with Software

- Open engineering design started with “source code” when it became free to duplicate designs
- Copy/Paste of software duplicates value and costs nothing.
- It also allowed for broader teams to collaborate.

Stuff became free

- Professionally-made software platforms became free because adoption in high volumes is more beneficial than sales in small volumes.
- Offering a freeware allowed the “little guy” publisher to make software capabilities available freely to other “little guys” – think about freelance designers who don’t have the resources of a full-scale business, but want to create value in the world.

Community adoption

- The overhead of running a company includes technical support; but publishers realized there’s no better support than having a huge number of experts/users helping other users.
- So, forums and channels for sharing and peer-to-peer discussion replaces the overhead of large tech support in organizations.

It stalled after software

- Hardware is less digitized, less free to copy, and is less easy to standardize given that global materials and methods don’t always match up.
- So, the first offerings of open hardware were limited (and still mostly are limited) to simply designs of Printed Circuit Boards (PCBs) which, when industrialized, immediately took a path of fully-digital designs.
- All other products in the world have

The Actions

Digitize all designs:

- Make Hardware just like software: Freely copied, freely distributed, open for modification
- Make manufacturing Digital: Guarantee that the end product is a function of the design, not of unique methods

Robust Methodologies

- Variance in hardware reproduction still contains vastly more variables than software.
- Teach methods of design which reduce inputs such as machining skillset.
- Design that which can be reproduced on the lowest performance equipment and thus builds properly on all equipment.

Standardize communication

- We wish for nonexperts to have access to try their own hacks.
- For a software expert, we must present hardware descriptions with simplicity and accuracy.
- Use templates for designs, posts, and documentation to eliminate variables that complicate communication.

VISION: Educating Others

The inspiration:

How did arrive at standards of producing designs, content, and selected platforms for sharing?

How do characterize our designs and create a culture that supports the mission?

How will folks know "This module was created in the SCUTTLE community?"

How will we keep pace with an evolving industry instead of getting left behind?

Anything worth doing is worth doing well

- designs
- software
- demos

Well-done creations are worth sharing.

- Share CAD
- Share software
- Share demos

Shared creations are worth documenting.

- Models: Parametric
- Code: commented
- Videos: with explanations, links

Documented creations deserve explanation.

- Create entry-points for education
- Describe the design goals, weaknesses, strengths

Good explanations reach the audience.

- short videos to tell how it's done.
- back-links to design docs included in videos.
- authors take ownership, answer questions

VISION: SCUTTLE Project Priorities

Accessible to Everyone

- Source Components
- Instructions to build
- Usable software, distributed openly

Service to Education

- Lessons: The work must be broken down into lessons for proper digestion. Lessons offered by outsiders should be used before authoring new material.
- Ideation: make early advances to help where new makers fall. Channel creativity into productive directions. Channel poor ideas into better ones.
- Inspiration: Escape introverted tendencies and show outsiders why each advancement is special. Never be arrogant. Always question and always share when discoveries are made. Learn from leaders how to inspire.

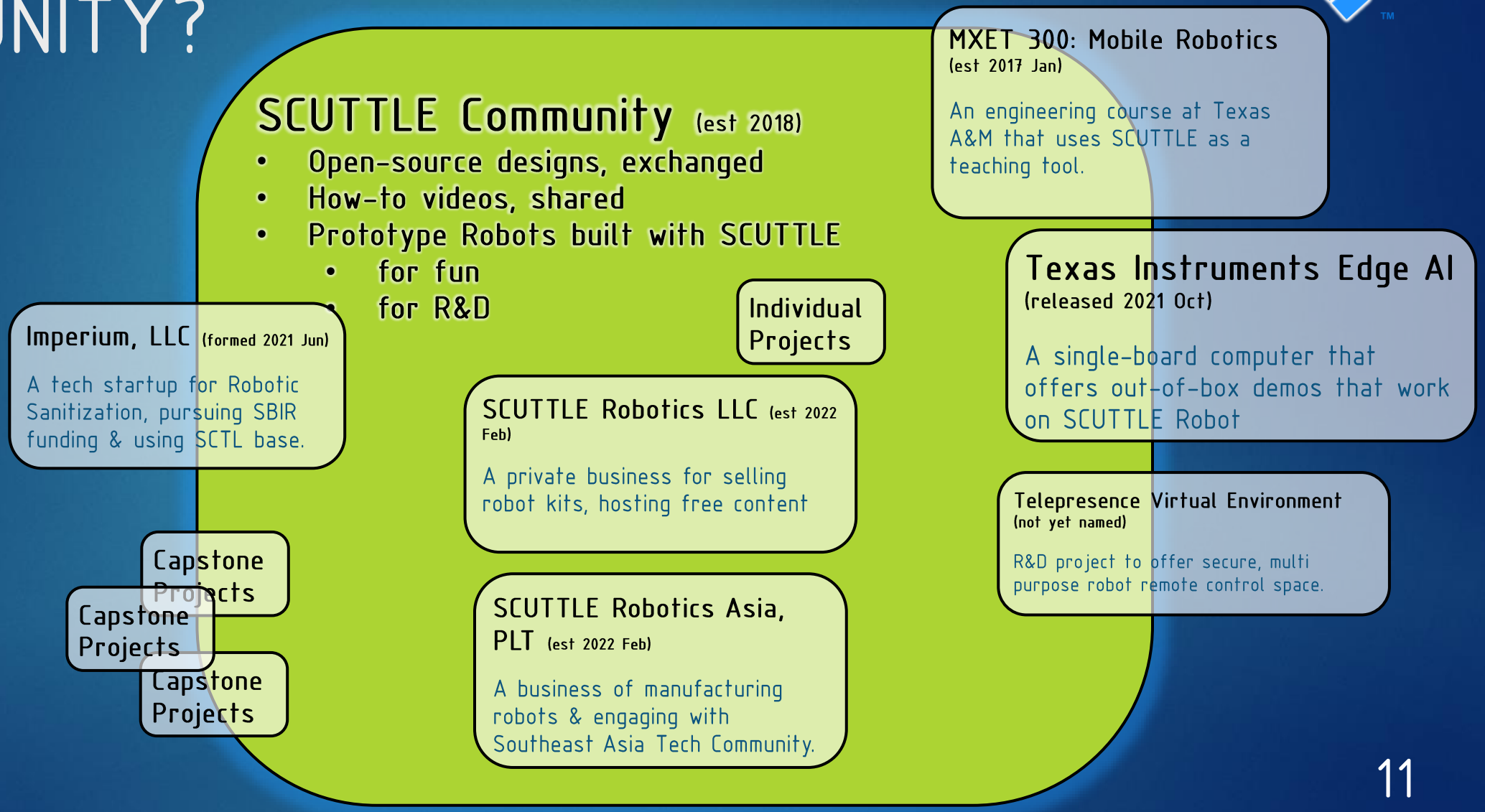
Leveraging State-of-the-Art Technology

- Compatibility: we harness the technology of peripheral industries only when we align our designs with the world's leading experts. Selection of materials, scripting language, communication protocols, all must follow the winners.
- Tied to industry: Among the hundreds of possible developments, we must create solutions that are sought by industrial partners. Concepts will sell themselves only when the buyers imagined the very outcome we build.
- Leading boundaries in:
  - Manufacturing
  - AI
  - Cross-discipline Applications
- Success by failure aversion: Do not allow gaps to creep into the project. If the customers whisper about a need, place a megaphone to their lips.

VISION: SCUTTLE Project

1. SCUTTLE isn't a device – it's a vehicle for advancing people and projects.
    1. We cross-pollinate useful technologies that are siloed
    2. We treat with preference components that are refined, supported, and mass-adopted
    3. As we learn, we teach
    4. As we teach, we listen for inputs from more experienced teams

VISION: WHAT MAKES THE SCUTTLE COMMUNITY?



VALUES: Community Core Values

Learning & discoveries

1.are fed back to the community.

Quality

- above quantity.

Robustness

- above features.

Discovery of value in every step.

- Avoid wastefulness when things don't go as planned.
- Integrity = capturing the value of mistakes

Standardization

- is a means to greater leaps.

Search

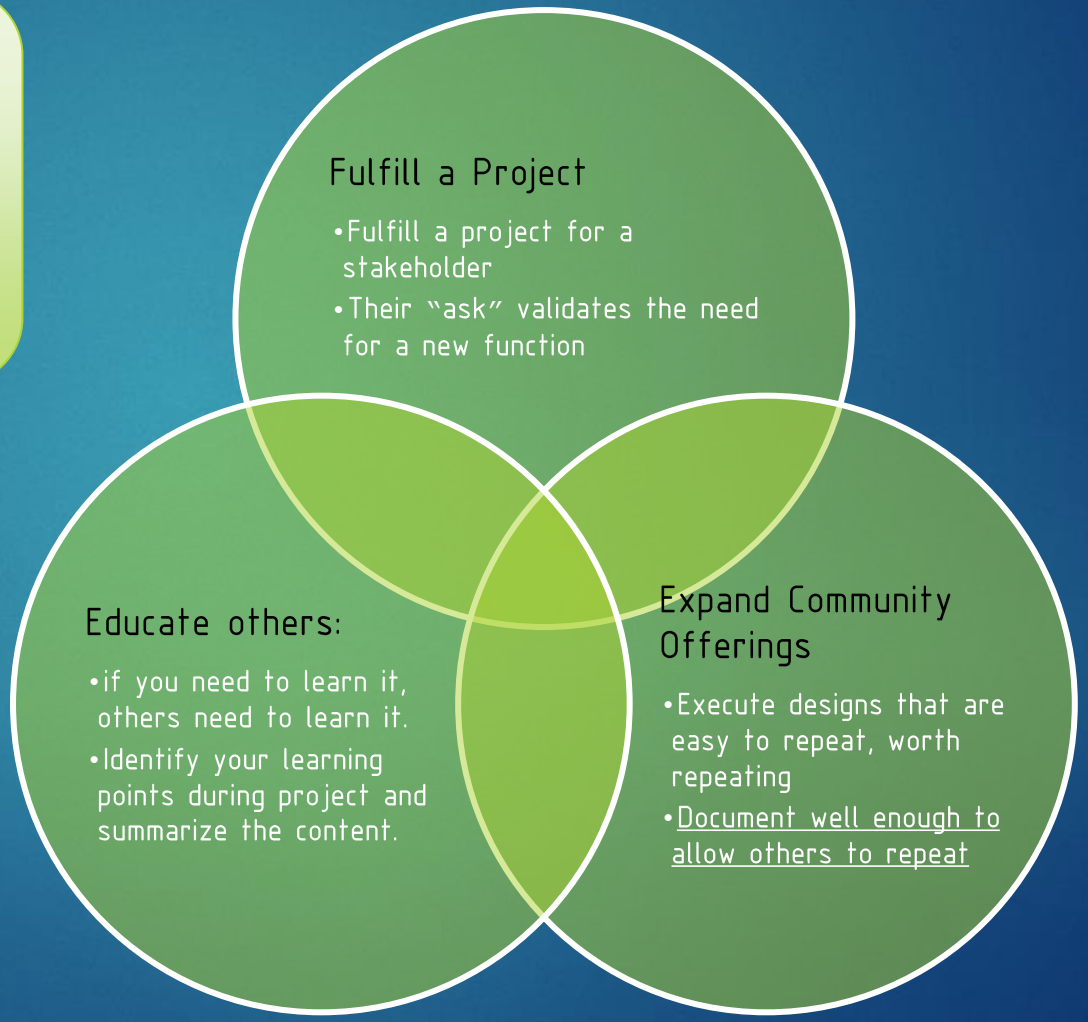
- far and wide for the wheel before reinventing.
- It's out there. Look harder.
- If it isn't out there, the most evolved step preceding the wheel is out there. Study it, then proceed.

Stay out of zero-sum territory.

- If the value output from an action cannot be duplicated ► red flag.
- If an exchange is made which does not build/create ► red flag

PRINCIPAL: Synergize projects by overlapping goals

- How do I select a new project to build?
  - When you select a project and generate ideas, consider how your outcomes can benefit more than one objective simultaneously.



VALUES ► Adopted: Open Source Strategy

Three Pillars of Open Source

Project-Community Fit

Persona: Developers

Measure: GitHub Stars

Product-Market Fit

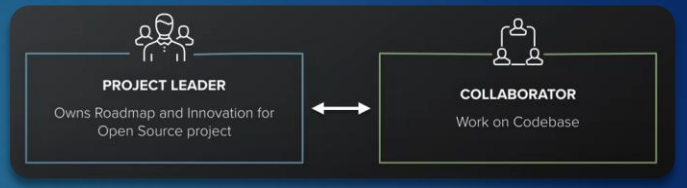
Persona: Users

Measure: Downloads

Value-Market Fit

Persona: Buyers

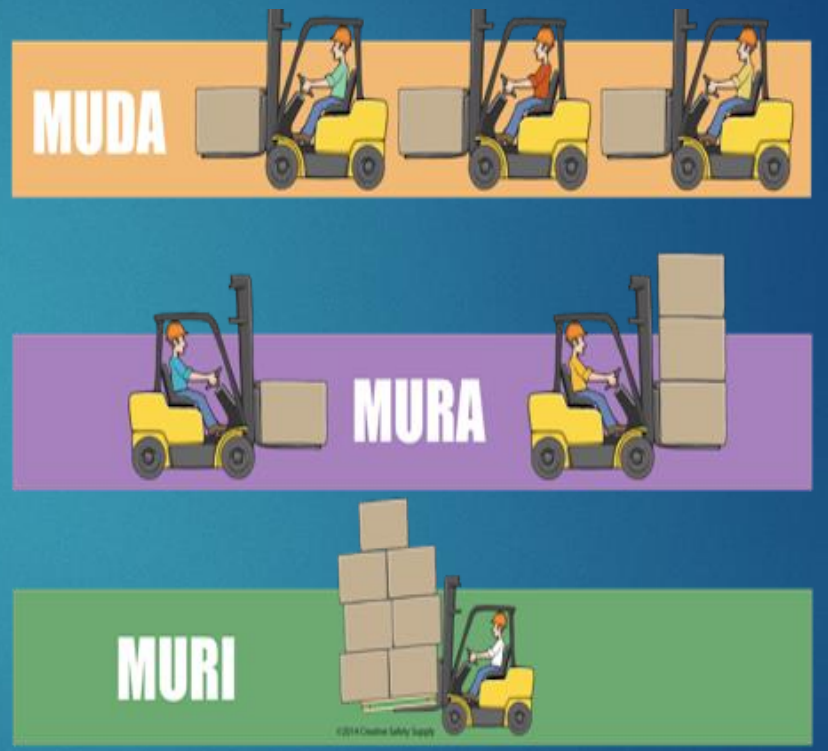
Measure: Revenue



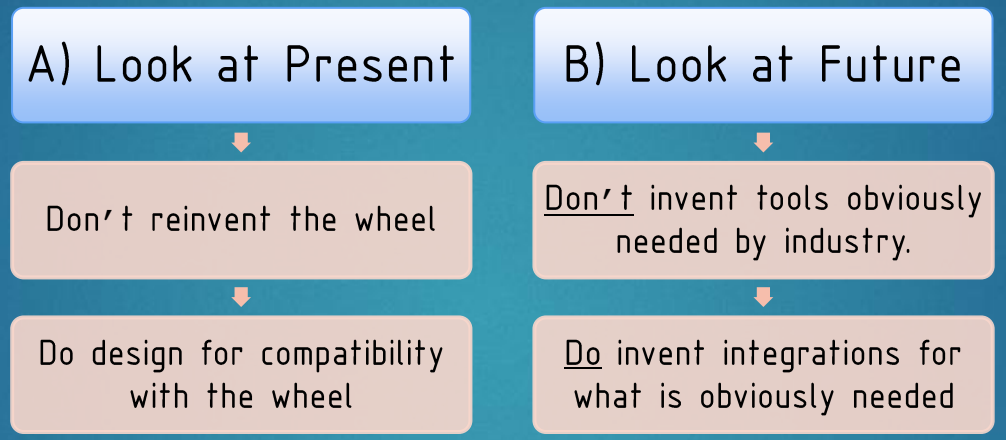
ADOPTED VALUES: Eliminate Muda

3 Kinds of Waste:
From Toyota Production System

- Muda: waste
- Mura: imbalance
- Muri: overload



PRINCIPAL: Don't Reinvent the Wheel



- Explanation: we create the most value by building:
  - content that is (and will be) unaddressed by other parties
  - Designs that "fills the gaps" between disconnected technologies
  - That leverages more than its own weight, using refined tech