

Composability is to Software as Compounding Interest is to Finance



Bill Mulligan, Cilium Community

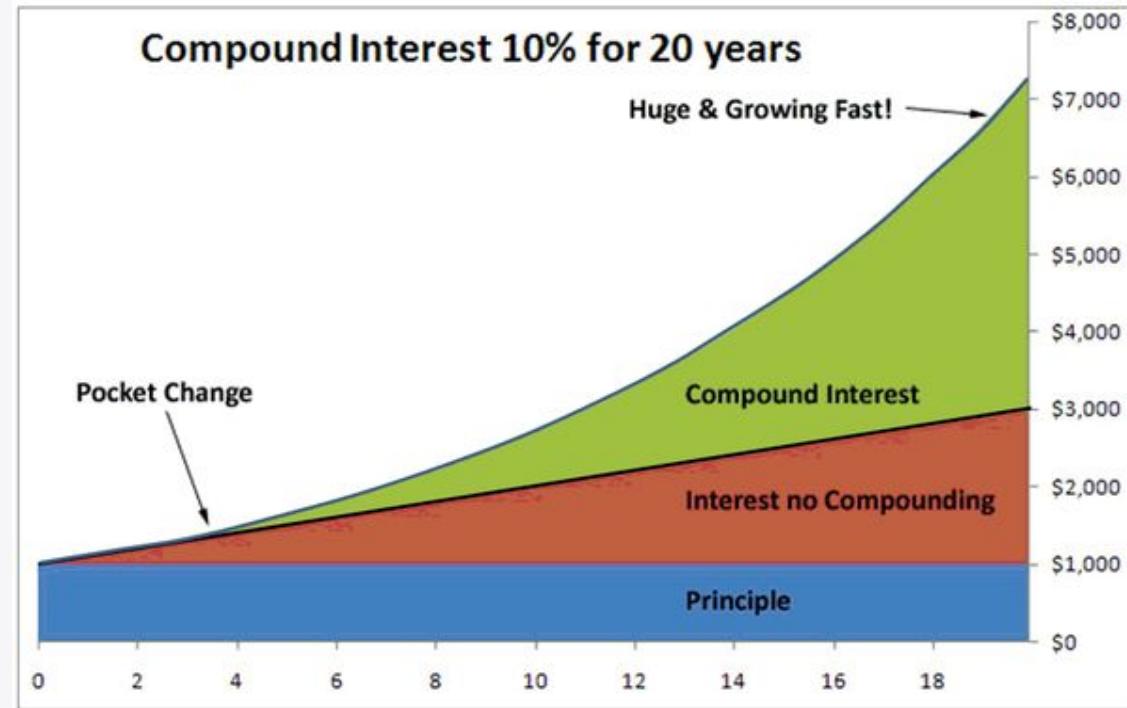
ISOVALENT

“Compound interest is the eighth wonder of the world. He who understands it, earns it; he who doesn't, pays it”

- Einstein

@breakawaybilly





@breakawaybilly



What is compound interest for software?

@breakawaybilly



What is compound interest for software?

Composability

@breakawaybilly



What is composability?

@breakawaybilly



Composability -

a system design principle that deals with the inter-relationships of components. A highly composable system *provides components that can be selected and assembled* in various combinations to satisfy specific user requirements

@breakawaybilly



Do we see composability in the cloud native ecosystem?

@breakawaybilly



Yes! And it's compounding

CLOUD NATIVE COMPUTING

WHERE WE STARTED



HOW IT'S GOING



@breakawaybilly

[datamattsson.tumblr](https://datamattsson.tumblr.com)





And it builds business value - 2016

- Kismatic ➔ Apprenda (May 2016)
- Joyent ➔ Samsung (June 2016)

VM
S

@breakawaybilly



And it builds business value - 2017

- Kismatic ➔ Apprenda (May 2016)
- Joyent ➔ Samsung (June 2016)
- Skipbox ➔ Bitnami (Feb 2017)
- Deis ➔ Microsoft (April 2017)
- Wercker ➔ Oracle (April 2017)
- Wavefront ➔ VMware (May 2017)
- Plexistor ➔ NetApp (September 2017)

VM
S

@breakawaybilly



And it builds business value - 2018

VM
S

- Kismatic ➔ Apprenda (May 2016)
- Joyent ➔ Samsung (June 2016)
- Skipbox ➔ Bitnami (Feb 2017)
- Deis ➔ Microsoft (April 2017)
- Wercker ➔ Oracle (April 2017)
- Wavefront ➔ VMware (May 2017)
- Plexistor ➔ NetApp (September 2017)
- Codeship ➔ CloudBees (February 2018)
- Netsil ➔ Nutanix (March 2018)
- GitHub ➔ Microsoft (June 2018)
- SUSE ➔ Micro Focus (July 2018)
- StackPointCloud ➔ NetApp (Sept 2018)
- Hortonworks ➔ Cloudera (October 2018)
- CoScale ➔ New Relic (October 2018)
- Heptio ➔ VMware (November 2018)
- BlueData ➔ HPE (November 2018)
- MLab ➔ MongoDB (Oct 2018)

@breakawaybilly



And it builds business value - 2019

VM
S

- Kismatic → Apprenda (May 2016)
- Joyent → Samsung (June 2016)
- Skipbox → Bitnami (Feb 2017)
- Deis → Microsoft (April 2017)
- Wercker → Oracle (April 2017)
- Wavefront → VMware (May 2017)
- Plexistor → NetApp (September 2017)
- Codeship → CloudBees (February 2018)
- Netsil → Nutanix (March 2018)
- GitHub → Microsoft (June 2018)
- SUSE ← Micro Focus (July 2018)
- StackPointCloud → NetApp (Sept 2018)
- Hortonworks → Cloudera (October 2018)
- CoScale → New Relic (October 2018)
- Heptio → VMware (November 2018)
- BlueData → HPE (November 2018)
- MLab → MongoDB (Oct 2018)
- Shippable → JFrog (Feb 2019)
- Electric Cloud → CloudBees (April 2019)
- NGINX → F5 (May 2019)
- Bitnami → VMware (May 2019)
- Praqma → Eficode (May 2019)
- Avi Networks → VMware (June 2019)
- Rollout.io → Cloudbees (June 2019)
- Red Hat → IBM (July 2019)
- Twistlock → Palo Alto Networks (July 2019)
- DevSecCon → Snyk (July 2019)
- Intrinsic → VMware (Aug 2019)
- Omniton → Splunk (Sept 2019)
- SignalFx → Splunk (October 2019)
- Streamlio → Splunk (October 2019)
- CloudSploit → Aqua (November 2019)
- Docker Enterprise → Mirantis (Nov 2019)
- Aporeto → Palo Alto Networks (Dec 2019)
- Pivotal → VMware (Dec 2019)

@breakawaybilly



And it builds business value - 2020

- Kismatic ➔ Apprenda (May 2016)
- Joyent ➔ Samsung (June 2016)
- Skipbox ➔ Bitnami (Feb 2017)
- Deis ➔ Microsoft (April 2017)
- Wercker ➔ Oracle (April 2017)
- Wavefront ➔ VMware (May 2017)
- Plexistor ➔ NetApp (September 2017)
- Codeship ➔ CloudBees (February 2018)
- Netsil ➔ Nutanix (March 2018)
- GitHub ➔ Microsoft (June 2018)
- SUSE ➔ Micro Focus (July 2018)
- StackPointCloud ➔ NetApp (Sept 2018)
- Hortonworks ➔ Cloudera (October 2018)
- CoScale ➔ New Relic (October 2018)
- Heptio ➔ VMware (November 2018)
- BlueData ➔ HPE (November 2018)
- MLab ➔ MongoDB (Oct 2018)
- Shippable ➔ JFrog (Feb 2019)
- Electric Cloud ➔ CloudBees (April 2019)
- NGINX ➔ F5 (May 2019)
- Bitnami ➔ VMware (May 2019)
- Praqma ➔ Eficode (May 2019)
- Avi Networks ➔ VMware (June 2019)
- Rollout.io ➔ Cloudbees (June 2019)
- Red Hat ➔ IBM (July 2019)
- Twistlock ➔ Palo Alto Networks (July 2019)
- DevSecCon ➔ Snyk (July 2019)
- Intrinsic ➔ VMware (Aug 2019)
- Omniton ➔ Splunk (Sept 2019)
- SignalFx ➔ Splunk (October 2019)
- Streamlio ➔ Splunk (October 2019)
- CloudSploit ➔ Aqua (November 2019)
- Docker Enterprise ➔ Mirantis (Nov 2019)
- Aporeto ➔ Palo Alto Networks (Dec 2019)
- Pivotal ➔ VMware (Dec 2019)
- Packet ➔ Equinix (Jan 2020)
- Scytale ➔ HPE (February 2020)
- Kontena ➔ Mirantis (February 2020)
- Containership ➔ Hitachi (March 2020)
- Affirmed Networks ➔ Microsoft (March 2020)
- Octarine ➔ VMware (May 2020)
- Jetstack ➔ Venafi (May 2020)
- Spot ➔ NetApp (June 2020)
- Rancher ➔ SUSE (July 2020)
- IGNW ➔ CDW (July 2020)
- Portworx ➔ Pure Storage (September 2020)
- Univa ➔ Altair (September 2020)
- Chef ➔ Progress (September 2020)
- 2ndQuadrant ➔ EnterpriseDB (Sept 2020)
- Portshift ➔ Cisco (October 2020)
- Kasten ➔ Veeam (October 2020)
- Instana ➔ IBM (November 2020)
- Banzai Cloud ➔ Cisco (Nov 2020)
- Flowmill ➔ Splunk (November 2020)

@breakawaybilly



And it builds business value - 2021

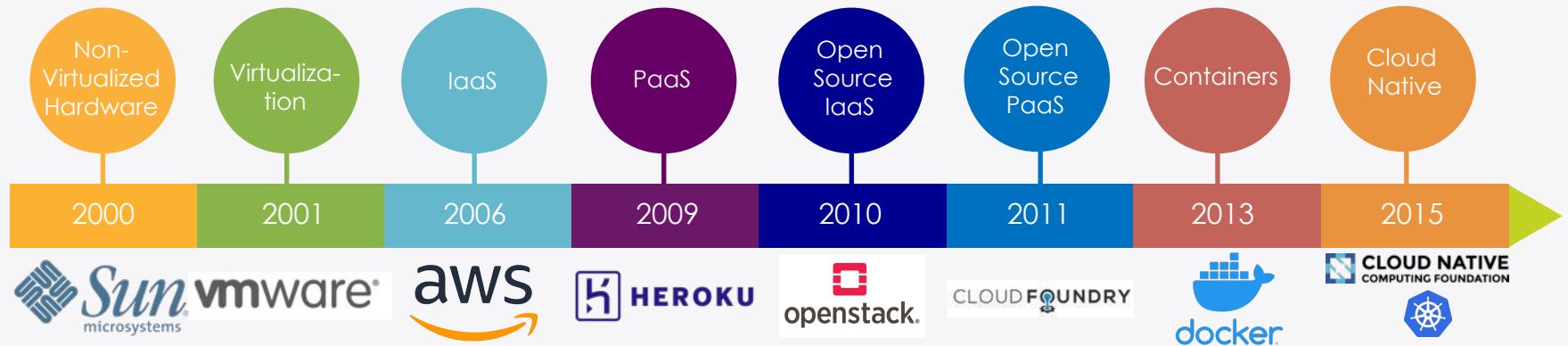
VM
S

- Stackrox → Red Hat (January 2021)
- Volterra → F5 (January 2021)
- EngineerBetter → Container Solutions (Feb 2021)
- Bridgecrew → Palo Alto Networks (Feb 2021)
- Humio → CrowdStrike (Feb 2021)
- Turbonomic → IBM (April 2021)
- Kinvolk → Microsoft (April 2021)
- Sensu → Sumo Logic (June 2021)
- k6 → Grafana Labs (June 2021)
- Zerto → HPE (June 2021)
- Platformer → WSO2 (June 2021)
- Apolicy → Sysdig (July 2021)
- Boxboat → IBM (July 2021)
- Epsagon → Cisco (August 2021)
- Novetta → Accenture (August 2021)
- Threat Stack → F5 (September 2021)
- Accurics → Tenable (October 2021)
- Replex → Cisco (October 2021)
- Neuvector → SUSE(October 2021)
- Mayadata → DataCore (Nov 2021)
- Argon Security → Aqua Security (Nov 2021)
- Wind River → Aptiv (Dec 2021)

@breakawaybilly



Innovation builds on itself



@breakawaybilly



If I have seen further than others, it is
because I was standing on the
shoulders of giants

- Newton

@breakawaybilly



Why is composability compounding?

@breakawaybilly

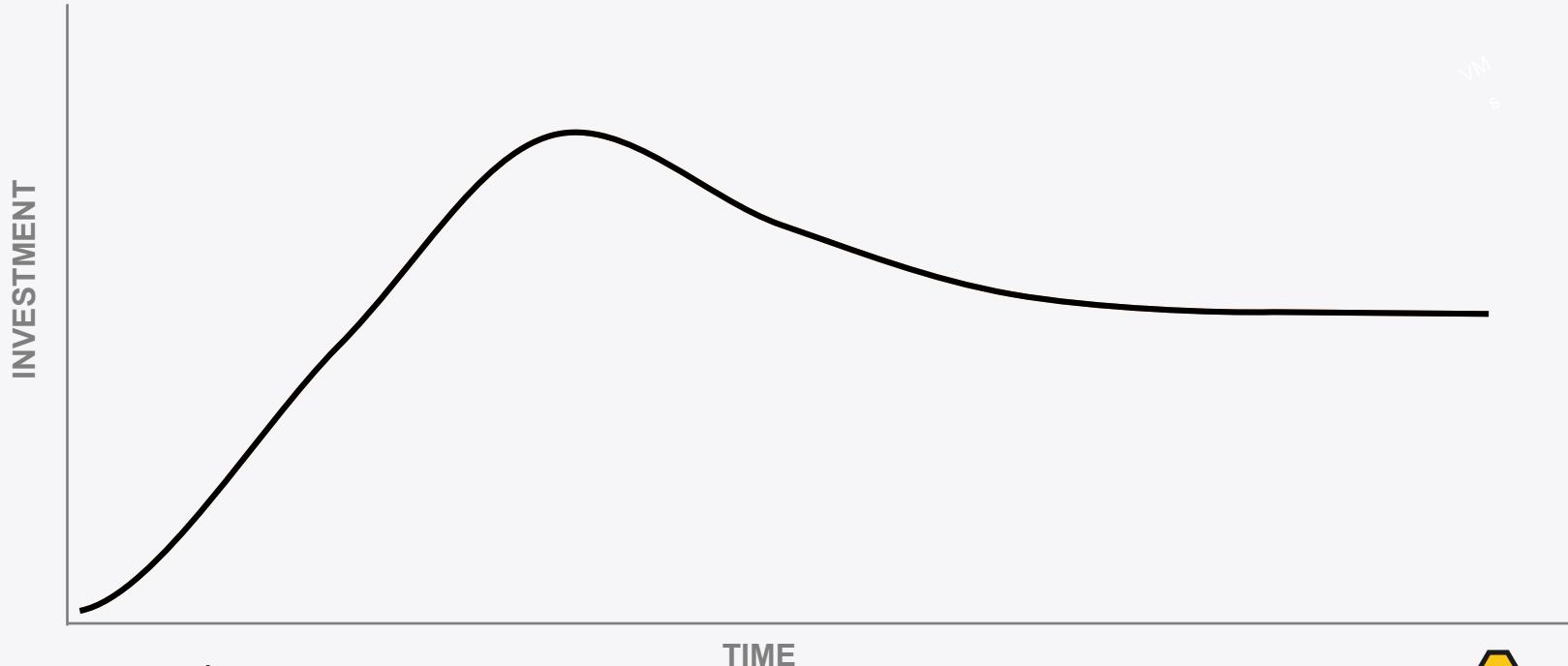


Let's start with one project

@breakawaybilly



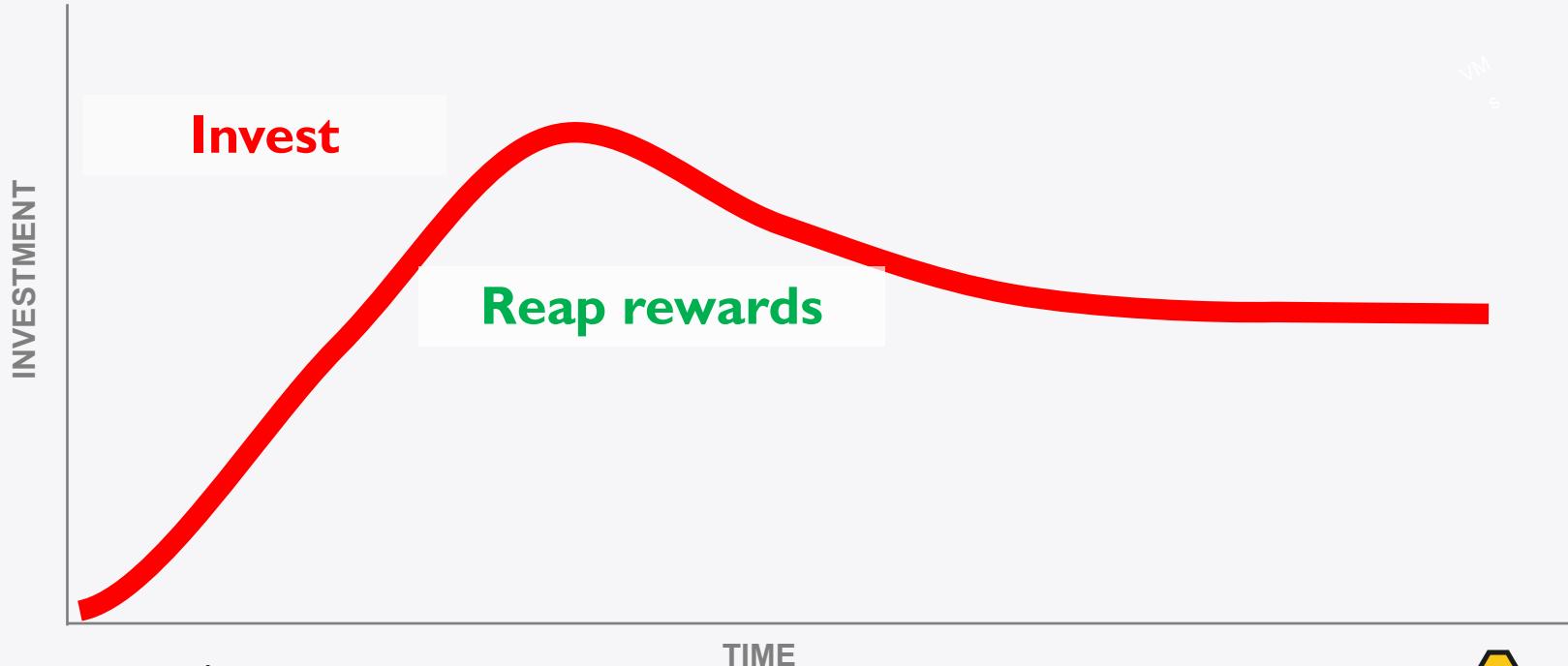
Projects have a life cycle



@breakawaybilly



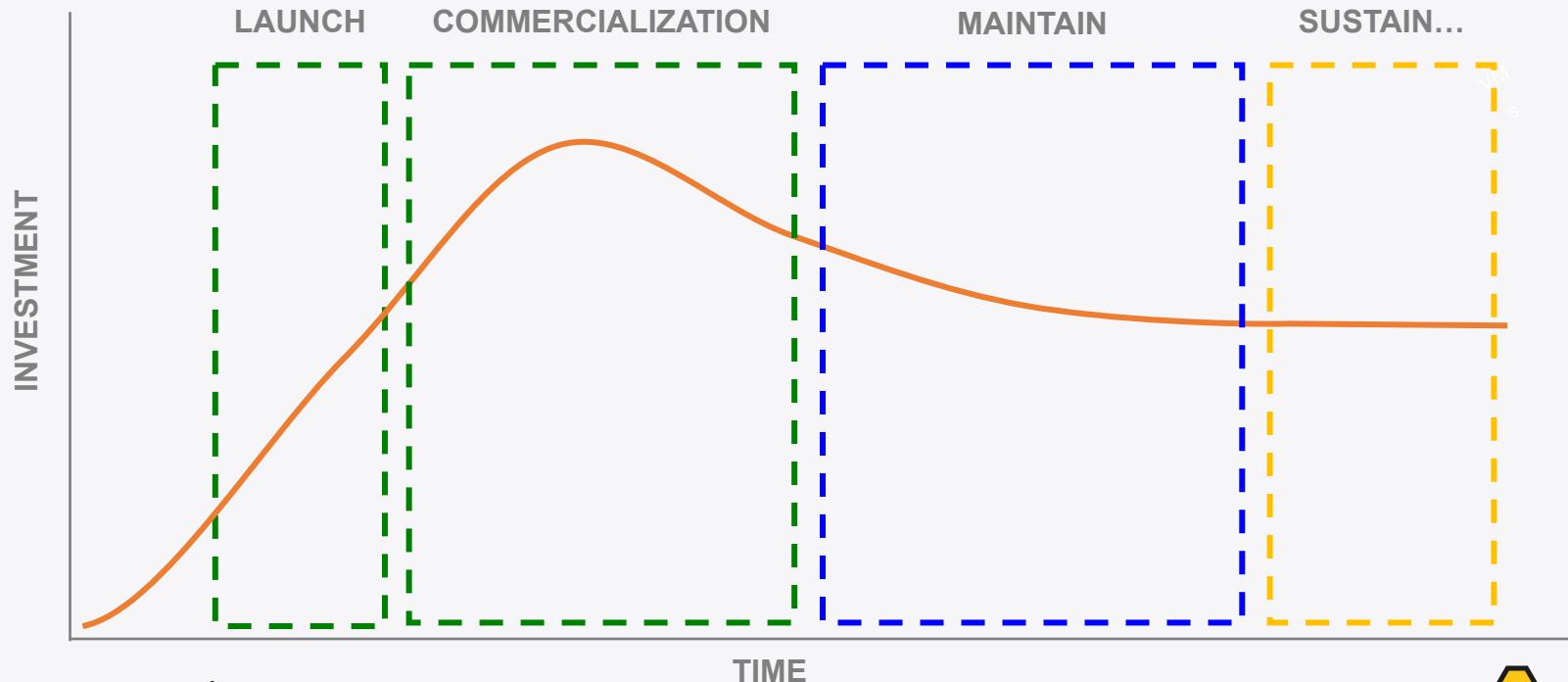
Projects have a life cycle



@breakawaybilly



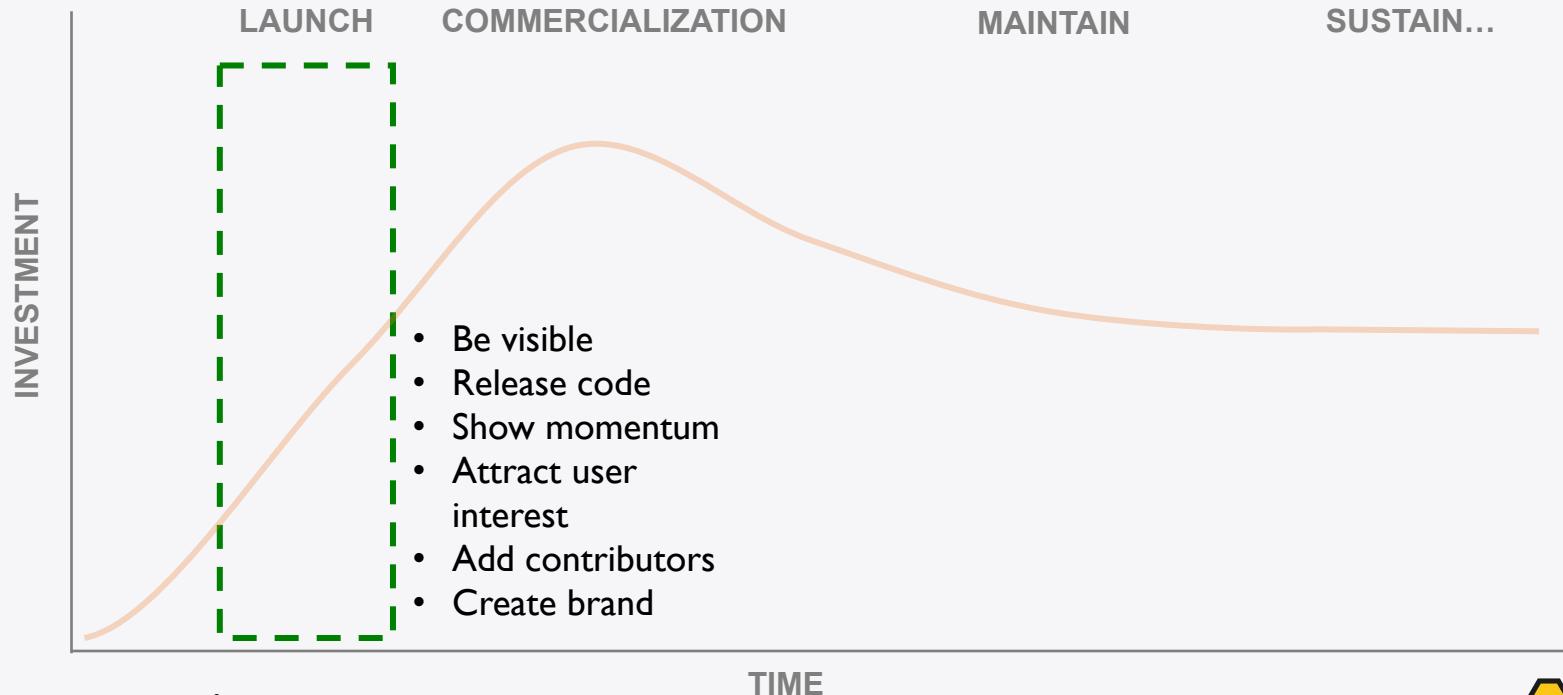
Projects have a life cycle



@breakawaybilly



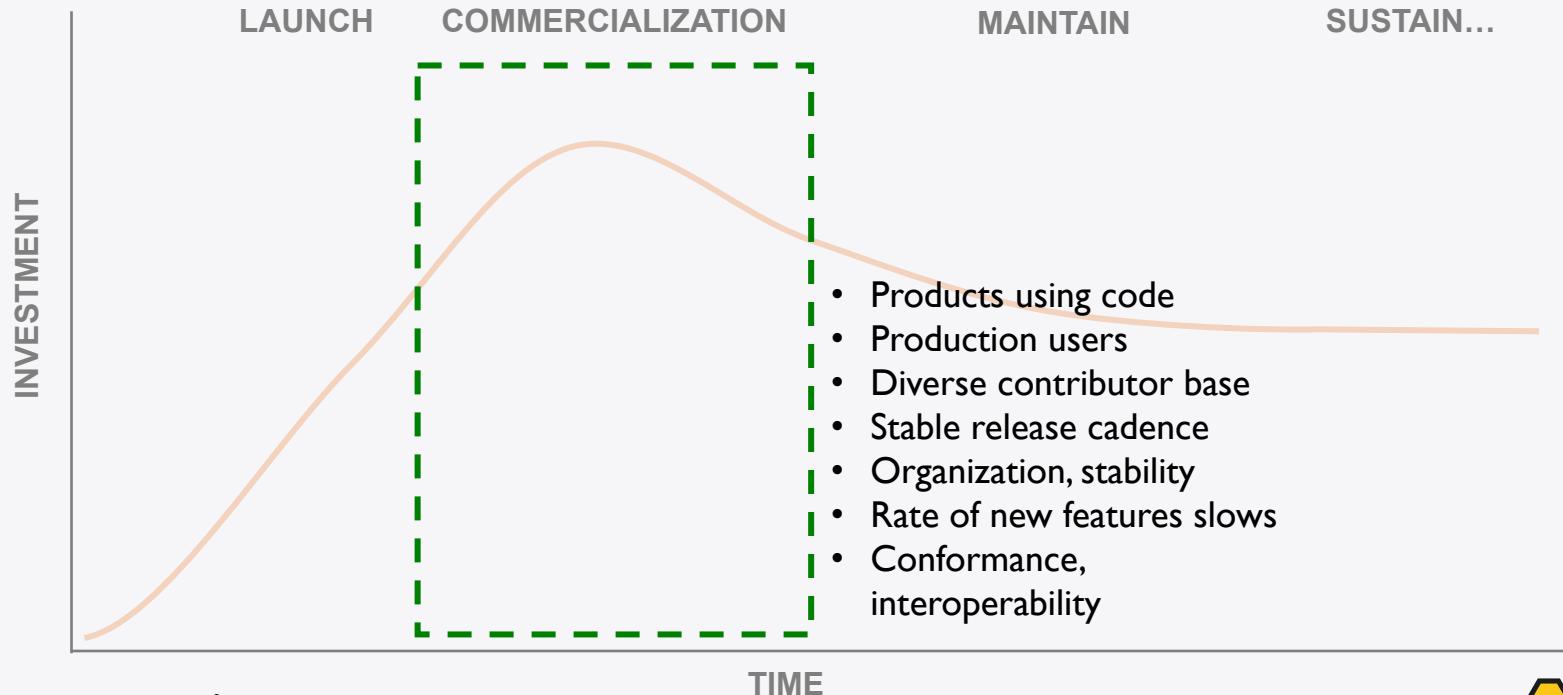
Projects have a life cycle



@breakawaybilly



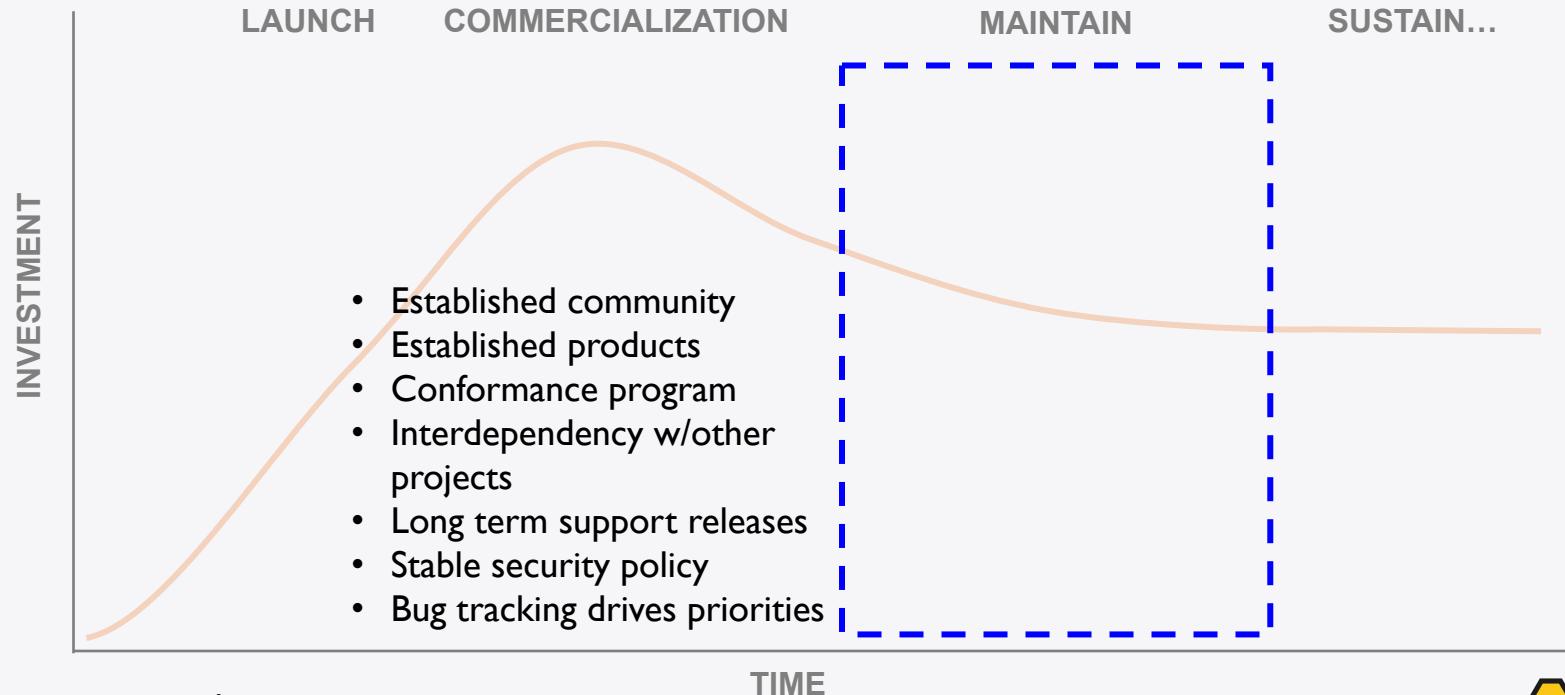
Projects have a life cycle



@breakawaybilly



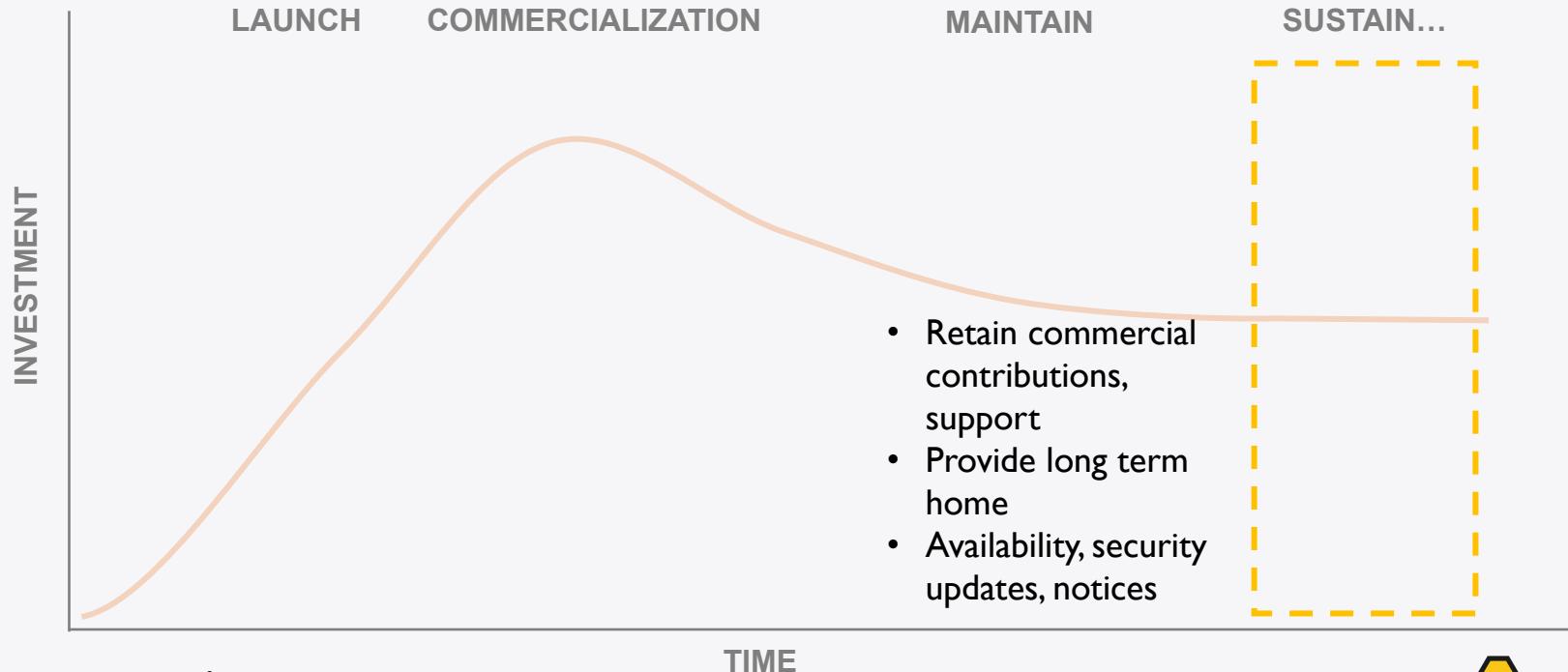
Projects have a life cycle



@breakawaybilly



Projects have a life cycle



@breakawaybilly



ISOVALENT

PROJECT

End User
Driven Open Source

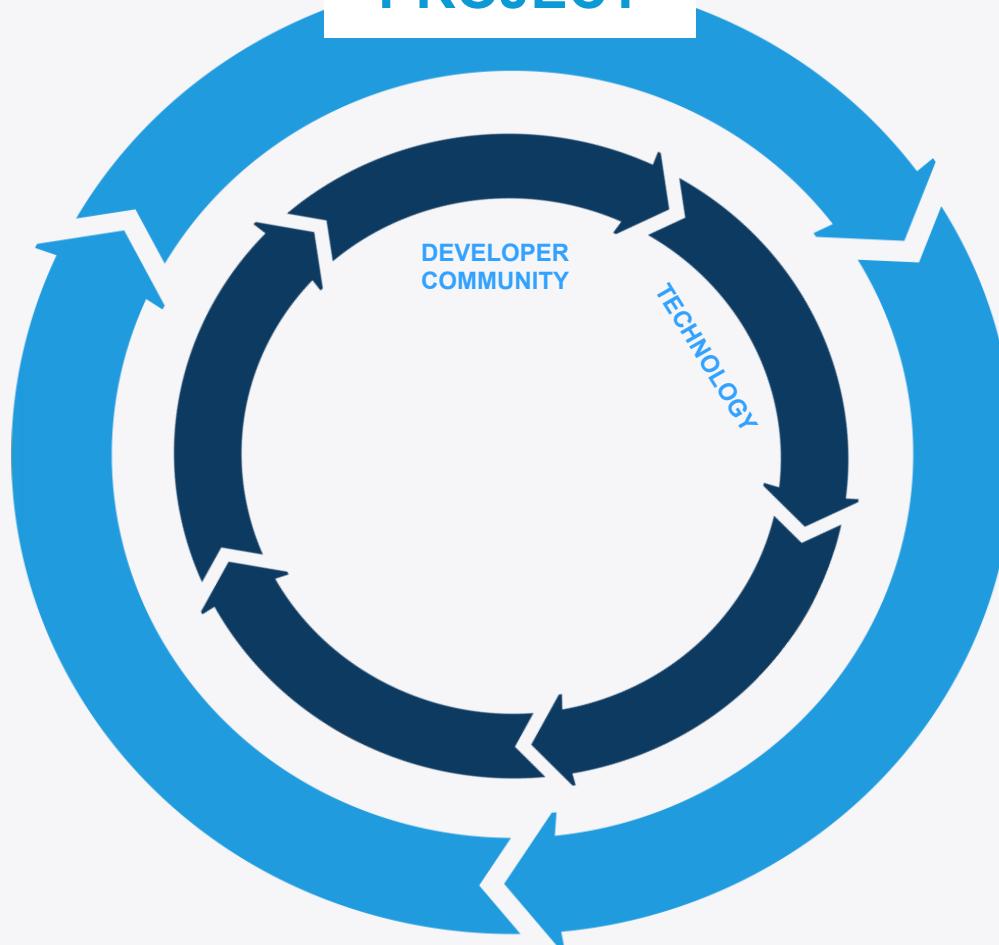
FEEDBACK

END USERS

@breakawaybilly



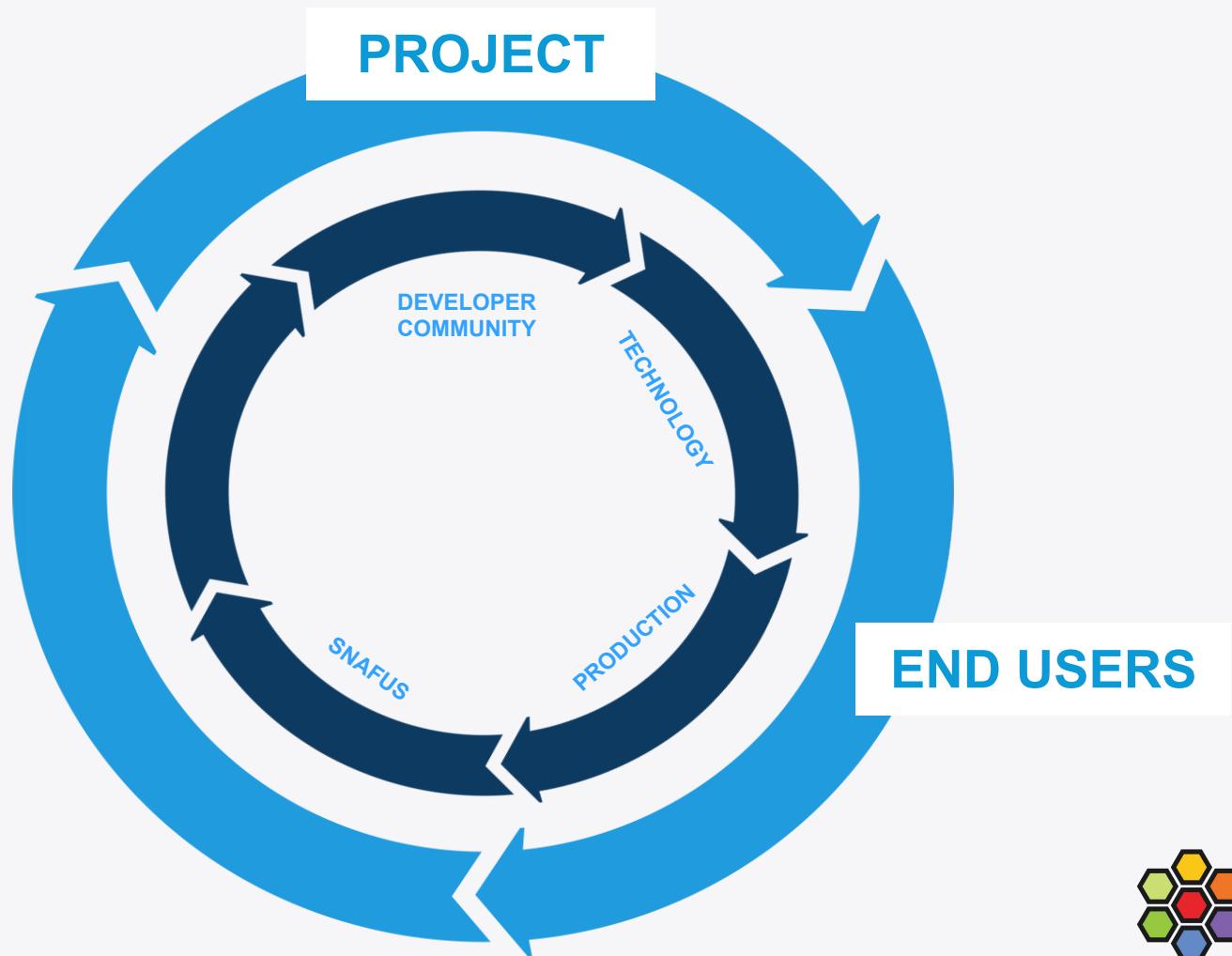
PROJECT



@breakawaybilly



ISOVALENT

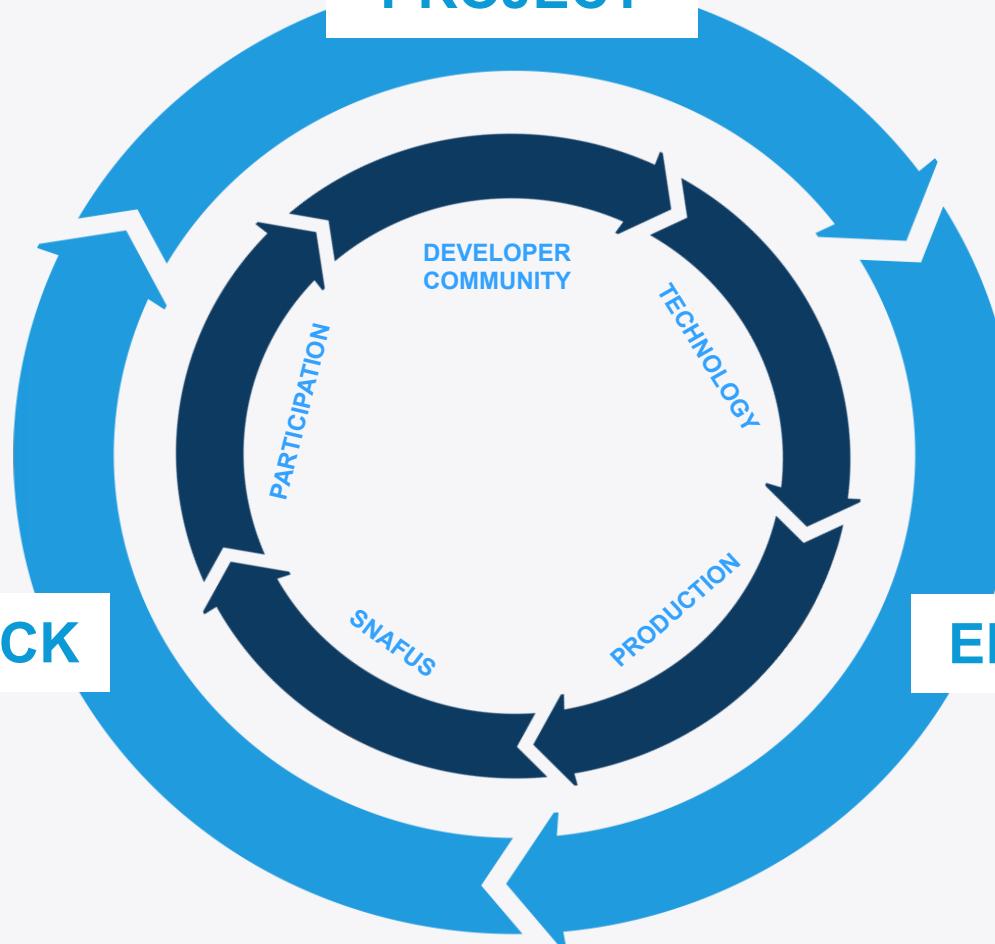


@breakawaybilly



ISOVALENT

PROJECT



@breakawaybilly



Ecosystems, like cloud native, are a collection of lifecycles

@breakawaybilly

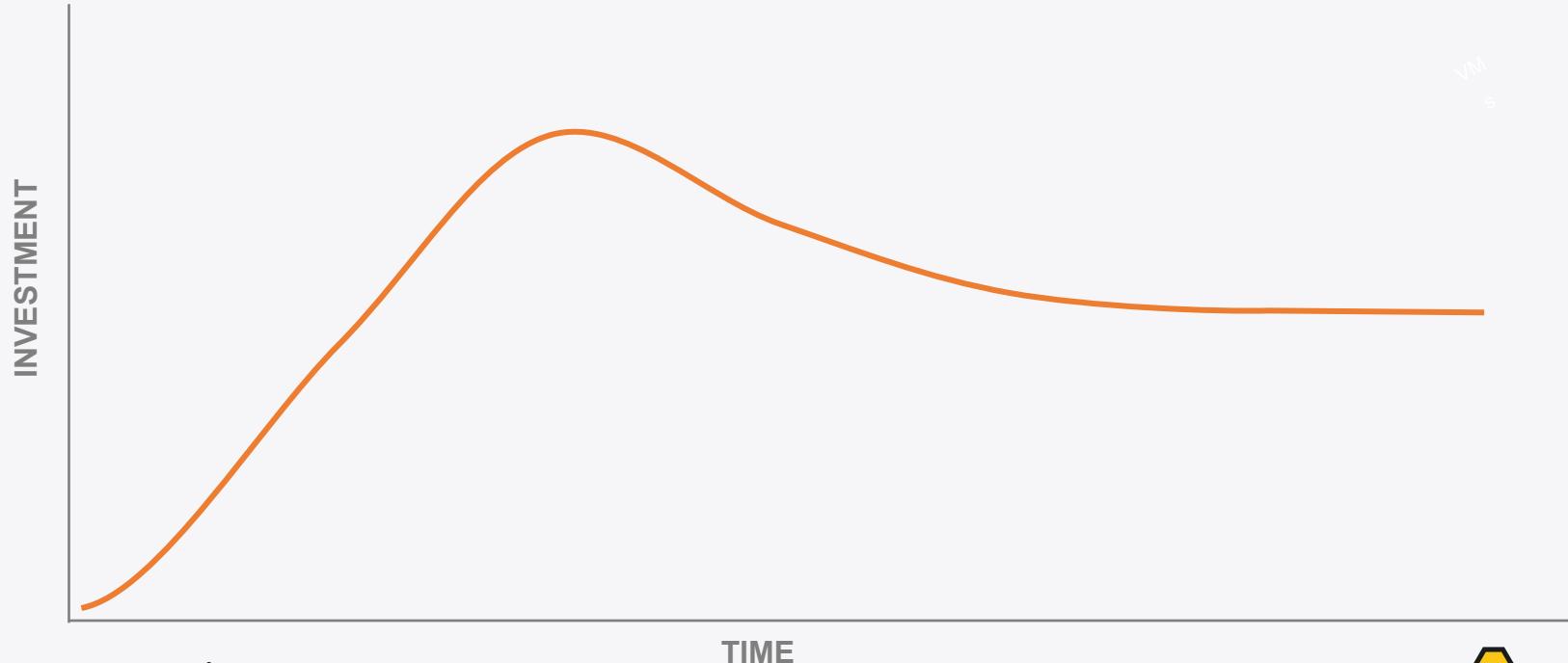


Let's add a second project

@breakawaybilly



Where do we put the curve?



@breakawaybilly



Where do we put the curve?

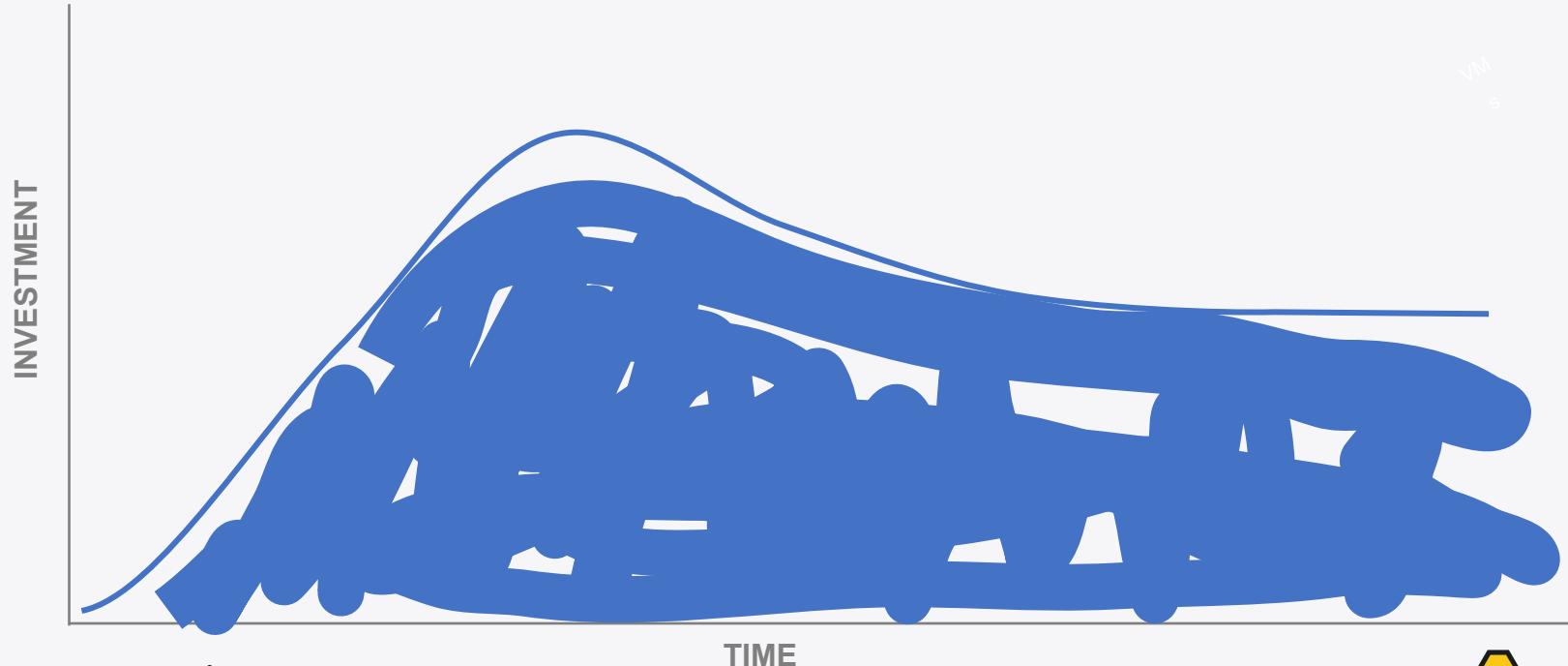


@breakawaybilly



ISOVALENT

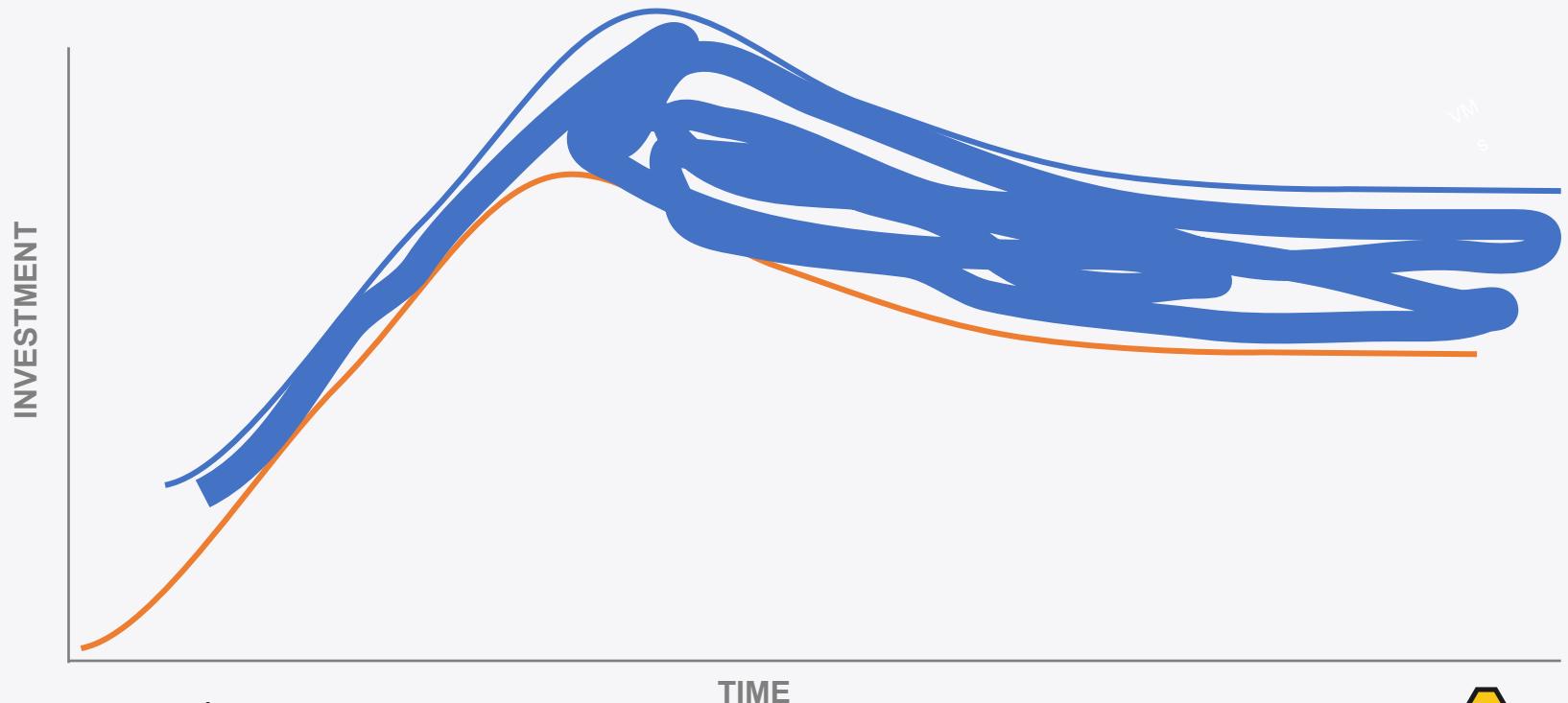
It's not the same investment twice



@breakawaybilly



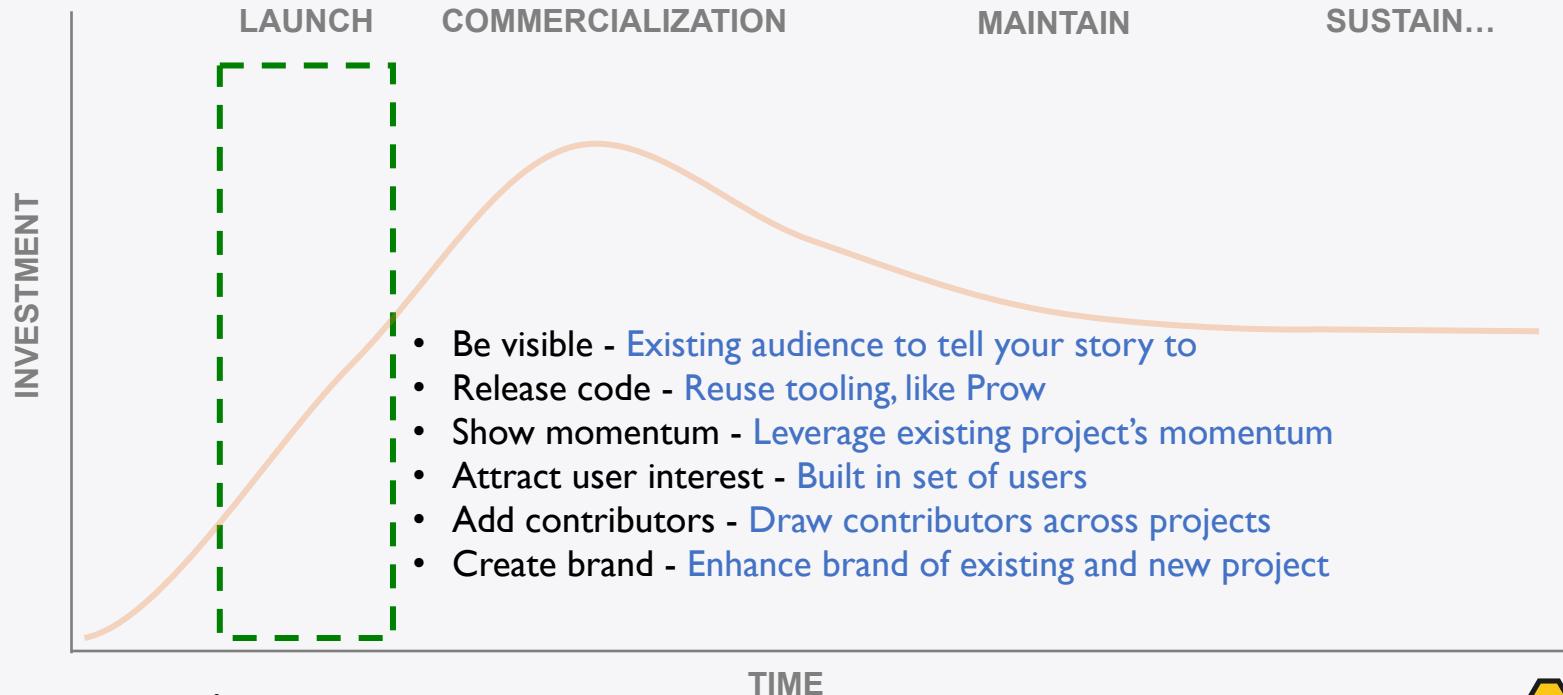
Area under the curve, not addition



@breakawaybilly



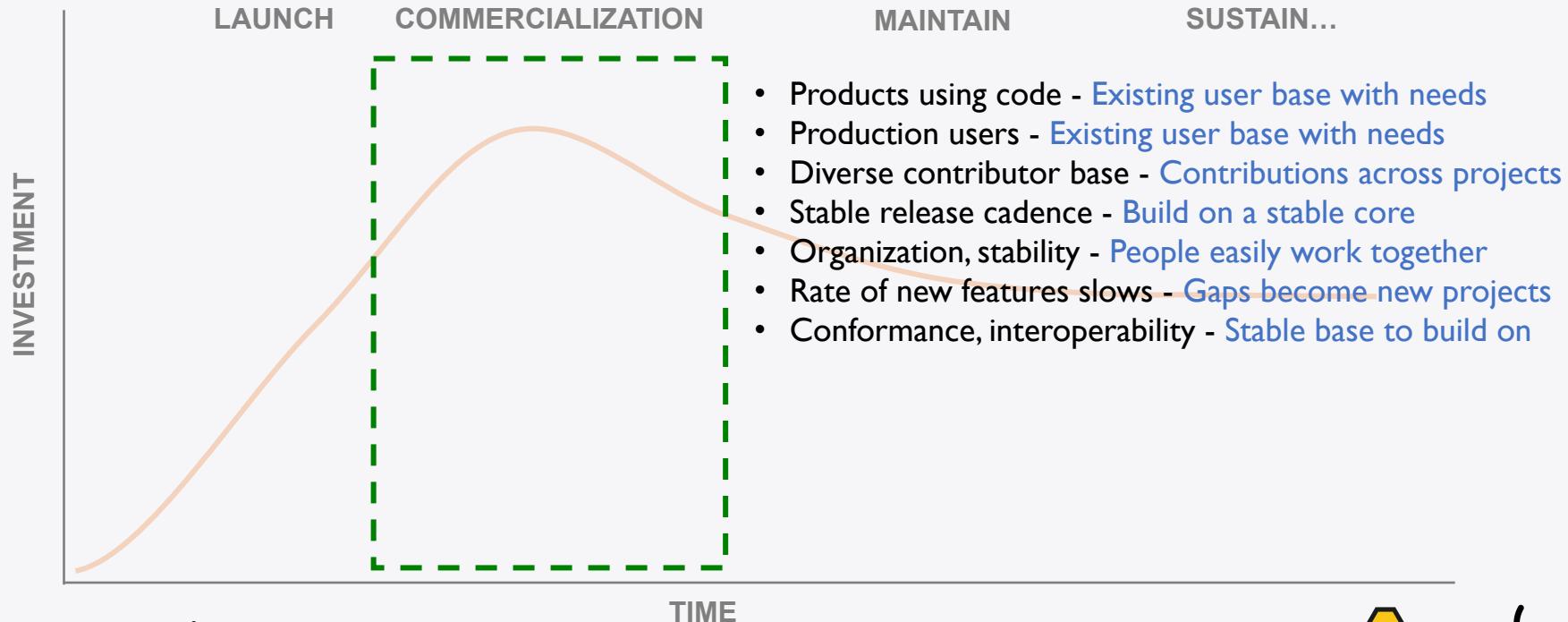
Composability of projects



@breakawaybilly



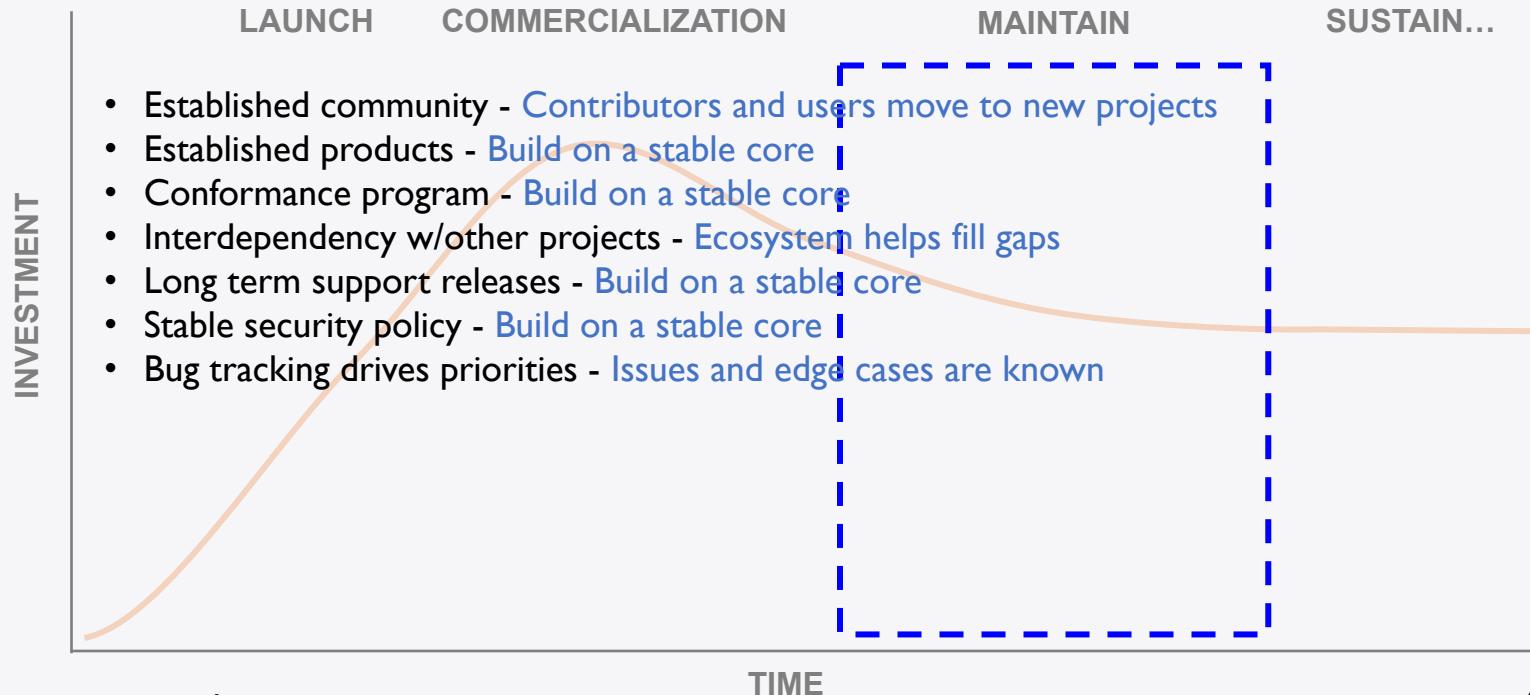
Composability of projects



@breakawaybilly



Composability of projects



@breakawaybilly



Composability Compounds

Ideas and Identity

- Be visible
- Create brand
- Show momentum
- Diverse contributor base



@breakawaybilly



Composability Compounds

People and Community

- Show momentum
- Attract user interest
- Add contributors
- Diverse contributor base
- Organization, stability
- Established community



@breakawaybilly



Composability Compounds

Tools and Integrations

- Release code
- Stable release cadence
- Rate of new features slows
- Conformance, interoperability
- Interdependency w/other projects
- Stable security policy



@breakawaybilly



Composability Compounds

Companies and Profits

- Products using code
- Production users
- Conformance, interoperability
- Established products
- Long term support releases
- Bug tracking drives priorities



@breakawaybilly



Composability Compounds

Ideas and Identity



People and Community



Tools and Integrations



Companies and Profits

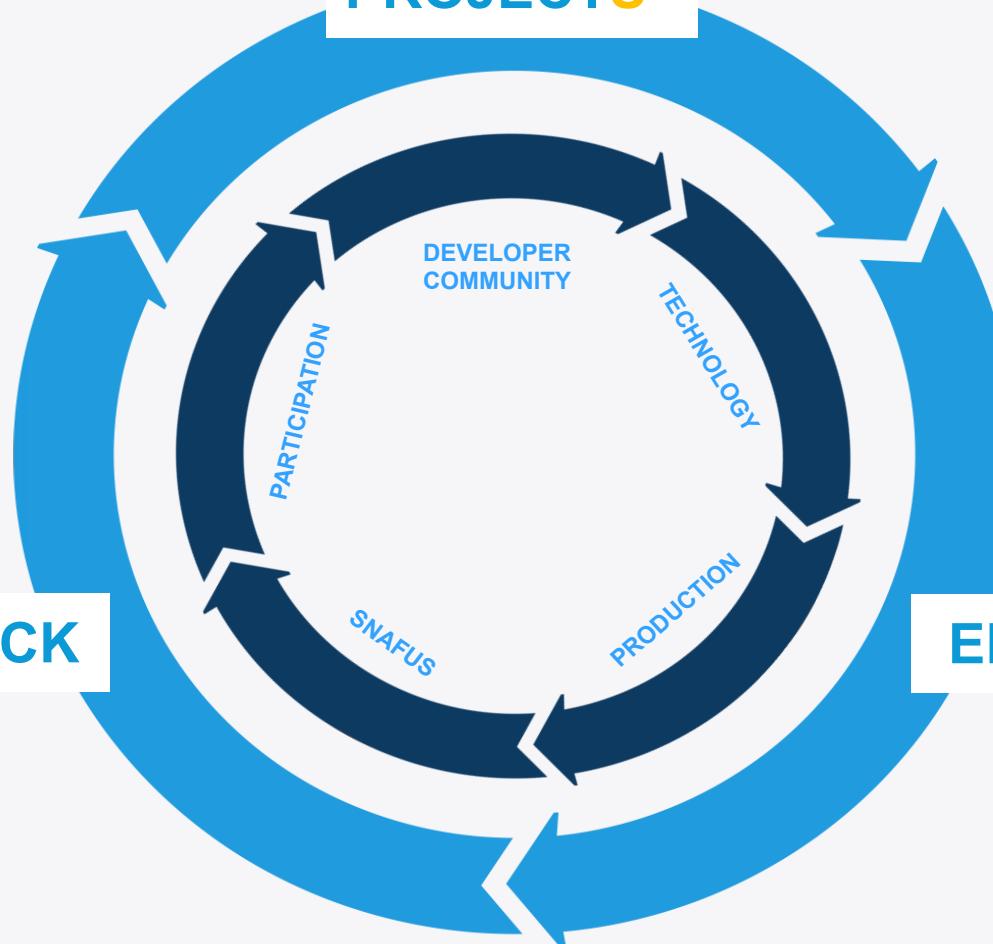


@breakawaybilly



ISOVALENT

PROJECTS



@breakawaybilly



**Solid building blocks allow innovation
to move up the stack**

@breakawaybilly



But what does this mean in the real world?

@breakawaybilly



Then

KubeCon 2015 — The Kubernetes Community Conference

📅 November 9th–11th, 2015 🗺 The Palace Hotel in San Francisco, California



KubeCon 2015 is the first inaugural Kubernetes conference. We will have a variety of expert technical talks designed to spark creativity and promote Kubernetes education.

There will be events planned outside the main conference track to allow attendees to make connections with other Kubernetes enthusiasts. If you are passionate about building highly scalable microservices, make use of Go for your distributed application stack and/or are just getting started with Kubernetes, this is an event you won't want to miss!

@breakawaybilly





Cloud Native SecurityCon EUROPE

May 16 + 17, 2022 | Valencia, Spain
#CNSecurityCon



CLOUD NATIVE Wasm DAY EUROPE

May 16, 2022 | Valencia, Spain
#CLOUDNATIVEWASM



Vs now



CLOUD NATIVE TELCO DAY

MAY 16, 2022 | VALENCIA, SPAIN
#CNTelcoDay



MAY 16, 2022 | VALENCIA, SPAIN
#FLUENTCON



Kubernetes on EDGE DAY EUROPE

May 17, 2022 | Valencia, Spain
#k8sedge + #k8sedgeday



17 May 2022 | Valencia, Spain
#K8SBATCH #K8SHPC



KnativeCon EUROPE

17 May 2022 | Valencia, Spain
#KNATIVECON



Kubernetes AI DAY EUROPE

May 16, 2022 | Valencia, Spain
#k8sAI + #k8sAlday



ServiceMeshCon EUROPE

MAY 17, 2022 | VALENCIA, SPAIN
#SERVICEMESHCON



PrometheusDay EUROPE

May 17, 2022 | Valencia, Spain
#PROMTHEUSDAY



May 16, 2022 | Valencia, Spain
#CNeBPF



May 17, 2022 | Valencia, Spain
#GITOPSCON



@breakawaybilly

Cloud-Native Storage Situation - 2017 slide

- Fragmented storage integrations are **hurting customer experience and limiting cloud native use cases** for persistent applications
- **Value** in how applications **make use of storage resources** and not how volumes are created, removed, and attached

Kubernetes storage

Volume plugins were “in-tree” means:

- Challenging to add support for new volume plugins
- Vendors wanting must align with K8s release
- Third-party code = reliability and security issues
- Code was difficult to test and maintain

@breakawaybilly



Composability -

a system design principle that deals with the inter-relationships of components. A highly composable system *provides components that can be selected and assembled* in various combinations to satisfy specific user requirements

@breakawaybilly



Goals of libStorage and CSI - 2017 slide

- Ensure **excellent user experience for consumers of storage** with cloud-native platforms
- Simplify storage platform responsibilities through a **single integration method**
- Provide volume orchestration and **common storage operations** to cloud-native platforms

Kubernetes storage ecosystem

Container Storage Interface (CSI):

- Standard for exposing arbitrary storage systems
- Vendors implement and update independently
- Removed third-party code from Kubernetes



@breakawaybilly



Kubernetes storage ecosystem

Results:

- >110 CSI drivers
- Reduced maintainer burden
- Increased security
- Massive expansion in number of potential Kubernetes users 💰



@breakawaybilly



#10: CNI - 2017 slide

- <https://github.com/containernetworking/cni>



- **CNI Specification:** defines an API between runtimes and network plugins for container network setup. No more, no less.
- **Plugins:** provide network setup for a variety of use-cases and serve as reference examples of plugins conforming to the CNI specification
- **Library:** provide a Go implementation of the CNI specification that runtimes can use to more easily consume CNI

Kubernetes CNI

Standardized interface allows users to choose and swap when needed

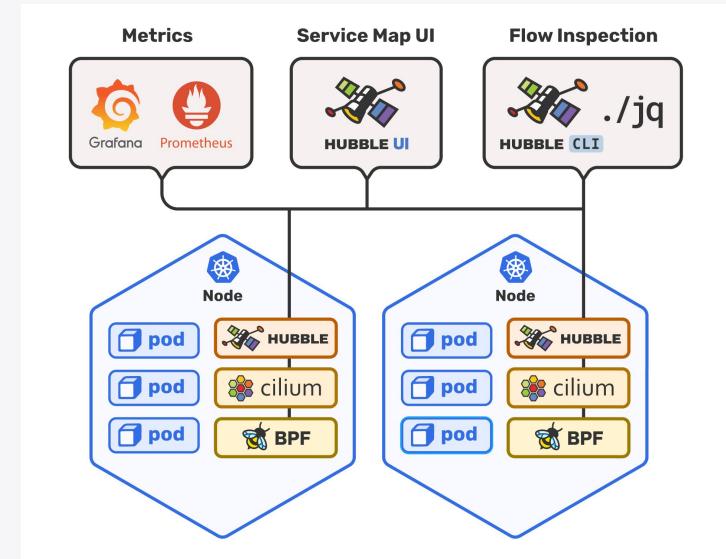
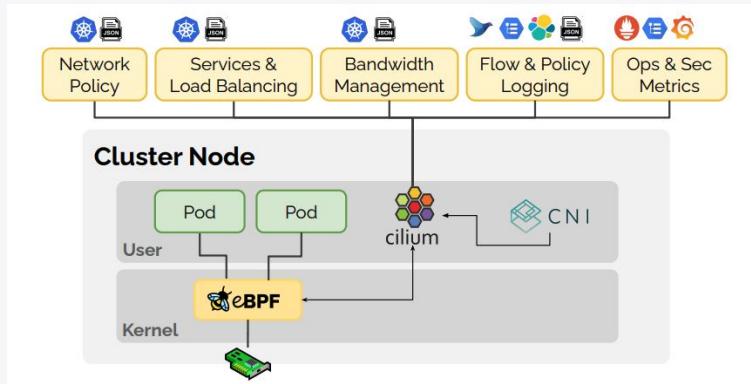


@breakawaybilly



Cilium

Composable projects for observability, networking,
and security

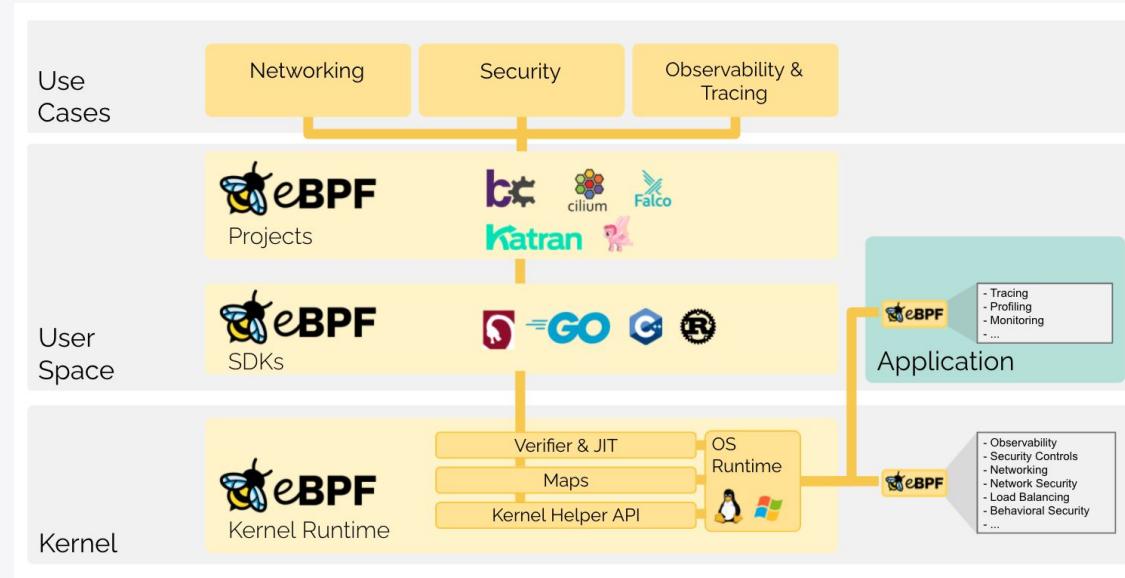


@breakawaybilly



eBPF

Making the Linux kernel composable and programmable



@breakawaybilly



Observability: OTel



OpenCensus



OPENTRACING

OpenCensus and OpenTracing have merged to form [OpenTelemetry](#), which serves as the next major version of OpenCensus and OpenTracing. OpenTelemetry will offer backwards compatibility with existing OpenCensus integrations, and we will continue to make security patches to existing OpenCensus libraries for two years.



OpenTelemetry

@breakawaybilly



What about the business?

@breakawaybilly



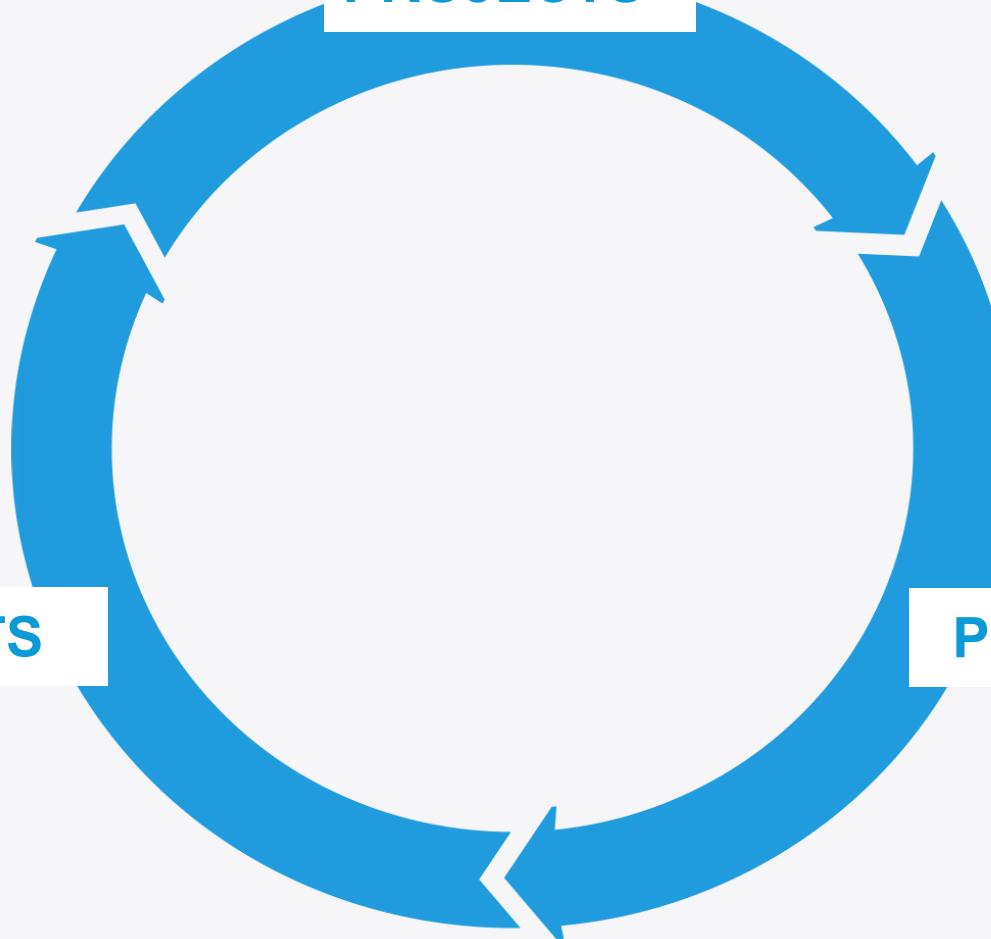
ISOVALENT

PROJECTS

PROFITS

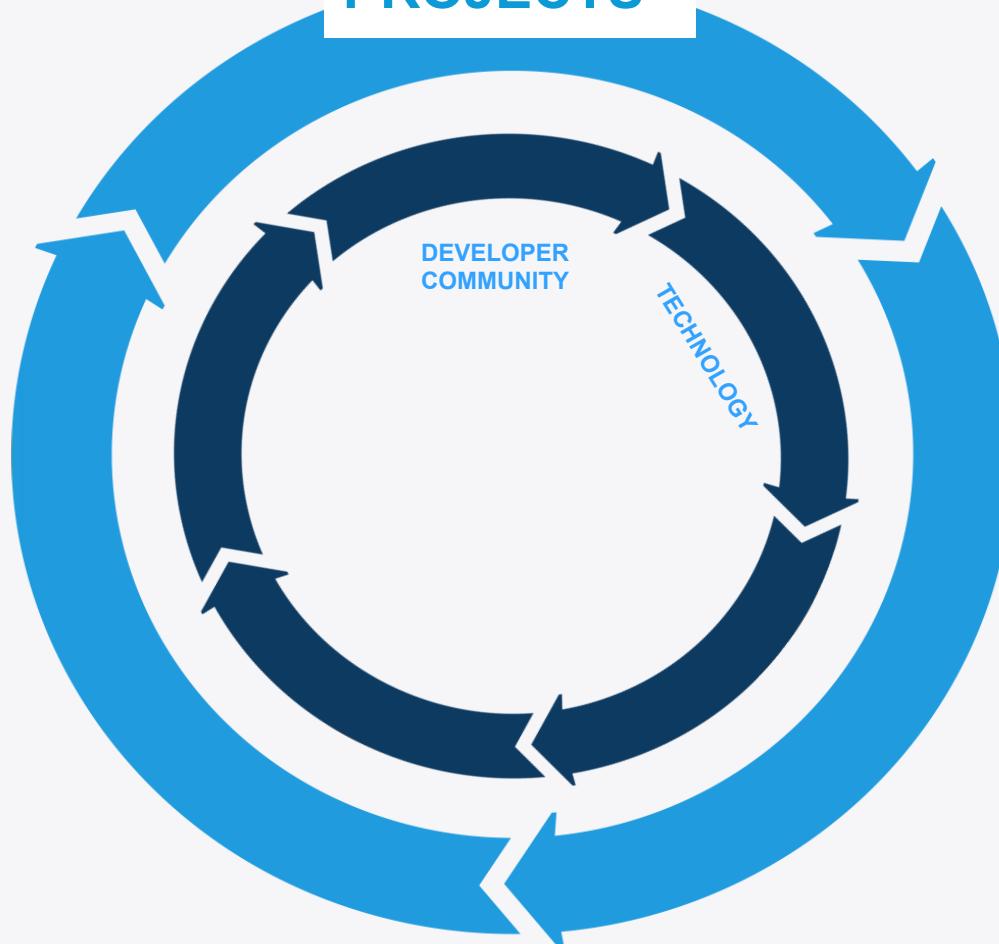
PRODUCTS

@breakawaybilly



ISOVALENT

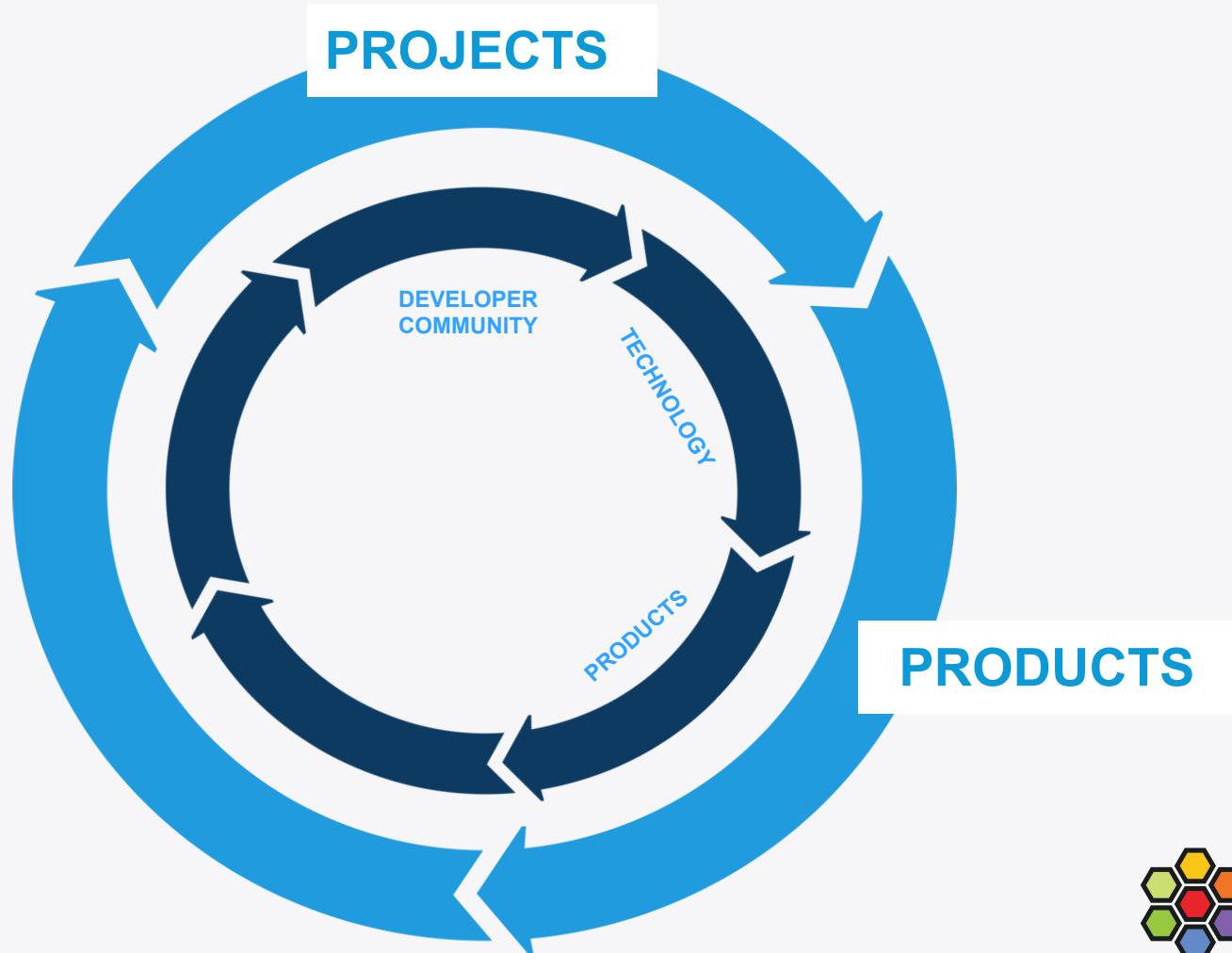
PROJECTS



@breakawaybilly



ISOVALENT

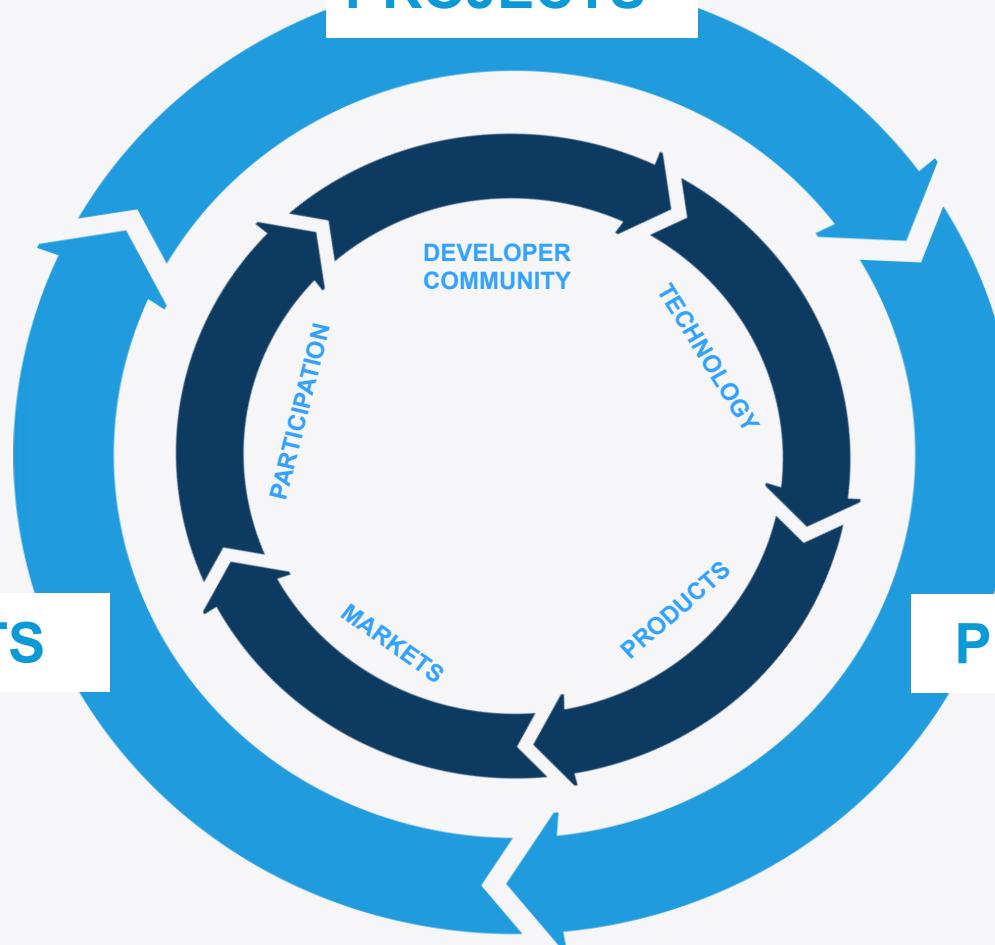


@breakawaybilly



ISOVALENT

PROJECTS



@breakawaybilly



Composability Compounds

Ideas and Identity



People and Community



Tools and Integrations



Companies and Profits



@breakawaybilly



Where to go next?

CNCF CONTRIBUTORS

Maintainers

- GitHub
- Templates
- Recommended
- Reviewing Guide
- Issue Labels

Community

- Community CRM
- Runbook
- Project Health
- Contributor Growth
- Motivation

Project Guidance

GitHub
Tips for managing your GitHub repositories

Community
Tips for managing your project's community

Governance
Define an open governance model

About TAG Contributor Strategy

TAG Contributor Strategy is responsible for contributor experience, sustainability, governance, and openness guidance to help CNCF community groups and projects with their own contributor strategies for a healthy project.

We are a CNCF Special Interest Group that advises the Technical Oversight Committee and CNCF projects on strategies related to building, scaling, and retaining contributor communities, including governance, communications, operations, and tools. We want to help grow flourishing, sustainable communities with smooth journeys throughout their CNCF project lifecycle.

@breakawaybilly



Where to go next?

Container

WHAT IT IS

A container is a running process with resource and capability constraints managed by a computer's operating system. The resources available to the container process are packaged as a container image. Containers run adjacent to each other on the same host machine, but typically the operating system prevents the separate container processes from interacting with each other.

PROBLEM IT ADDRESSES

Before containers were available, separate machines were necessary to run applications. Each machine required its own operating system, which takes CPU, memory, and disk space, all for an individual application to function. Managing, maintaining, upgrading, and starting up an operating system is another significant source of toil.

HOW IT HELPS

Containers share the same operating system and its machine resources, spreading the operating system's resource overhead and creating efficient use of the physical machine. This capability is only possible because containers are typically limited from being able to interact with each other. This allows many more applications to be run on the same physical machine.

There are limitations, however. Since containers share the same operating system, processes can be considered less secure than virtual machines. Containers also require limits on the shared resources. To guarantee resources, administrators must constrain and limit memory and CPU usage so that other applications do not perform poorly.



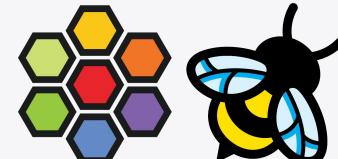
CLOUD NATIVE COMPUTING FOUNDATION

CLOUD NATIVE GLOSSARY

glossary.cncf.io

The [CNCF Cloud Native Glossary Project](#) is intended to be used as a reference for common terms used when talking about cloud native applications.

WORKING WITH DEFINITIONS



Where to go next?

CNCF Landscape Guide

The cloud native landscape guide was initiated by the [CNCF Business Value Subcommittee](#) and [Cartografos group](#). It was authored by [Jason Morgan](#) and [Catherine Paganini](#), edited and reviewed by [Simon Forster](#) and [Ihor Dvoretskyi](#), built by [Jordi Noguera](#) with UX consultation from [Andrea Velázquez](#).

Introduction

If you've researched cloud native applications and technologies, you've probably come across the [CNCF cloud native landscape](#). Unsurprisingly, the sheer scale of it can be overwhelming. So many categories and so many technologies. How do you make sense of it?

As with anything else, if you break it down and analyze it one piece at a time, you'll find it's not that complex and makes a lot of sense. In fact, the map is neatly organized by functionality and, once you understand what each category represents, navigating it becomes a lot easier.

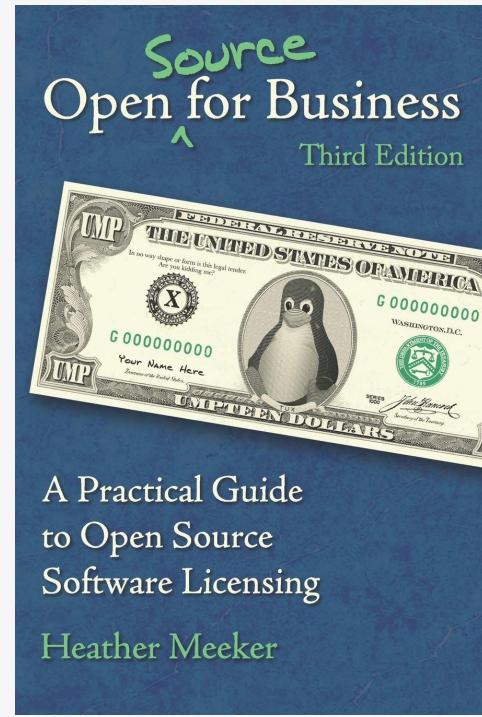
In this guide, we'll break this mammoth landscape down and provide a high-level overview of its layers, columns, and categories.

What is the cloud native landscape?

@CathPaga
@RJasonMorgan
l.cncf.io



Where to go next?



@HeatherMeeker4



Questions?

@breakawaybilly
bill@isovalent.com

