

# Open Source Software Business Models Redux

Stephen R. Walli

@stephenrwalli

[stephen.walli@gmail.com](mailto:stephen.walli@gmail.com)

# There is NO Open Source Business Model

Stephen R. Walli

@stephenrwalli

stephen.walli@gmail.com

A Story



INTERIX

[illegible]

1997

~750,000 LoC

~\$10,000,000

Interix  
~\$100,000

Interix + ACT  
~\$140,000

[illegible]



[illegible]

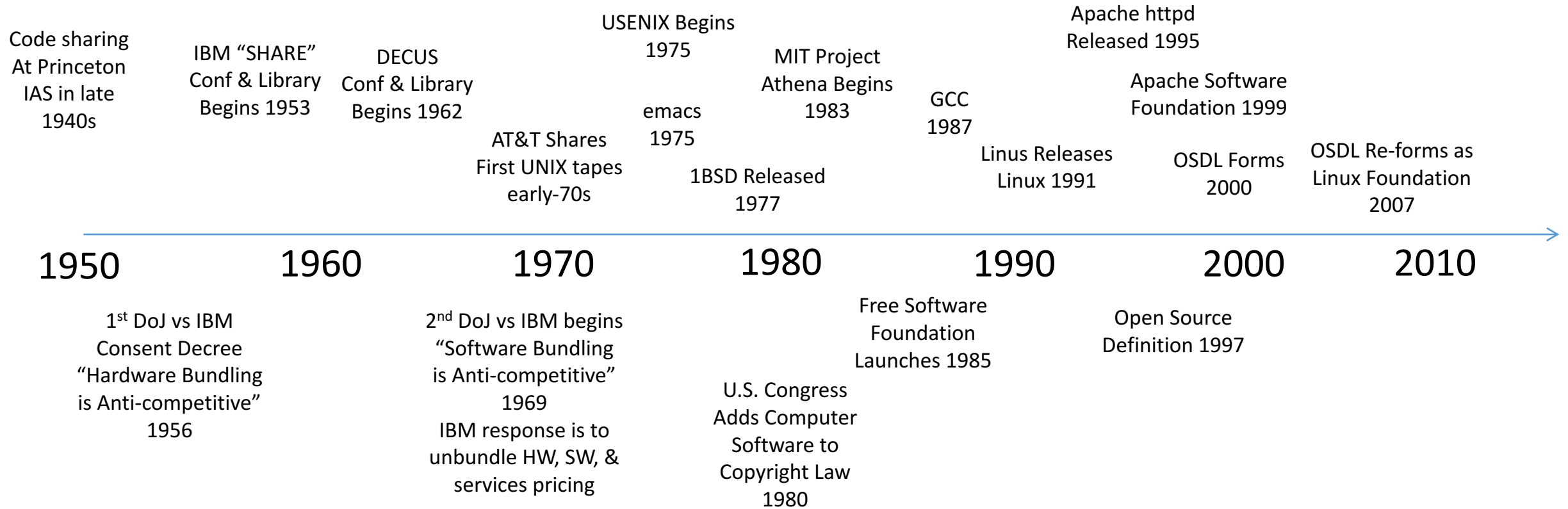
Build vs. Buy

Build vs. Buy vs. Borrow

Build vs. Buy vs. (Borrow & Share)

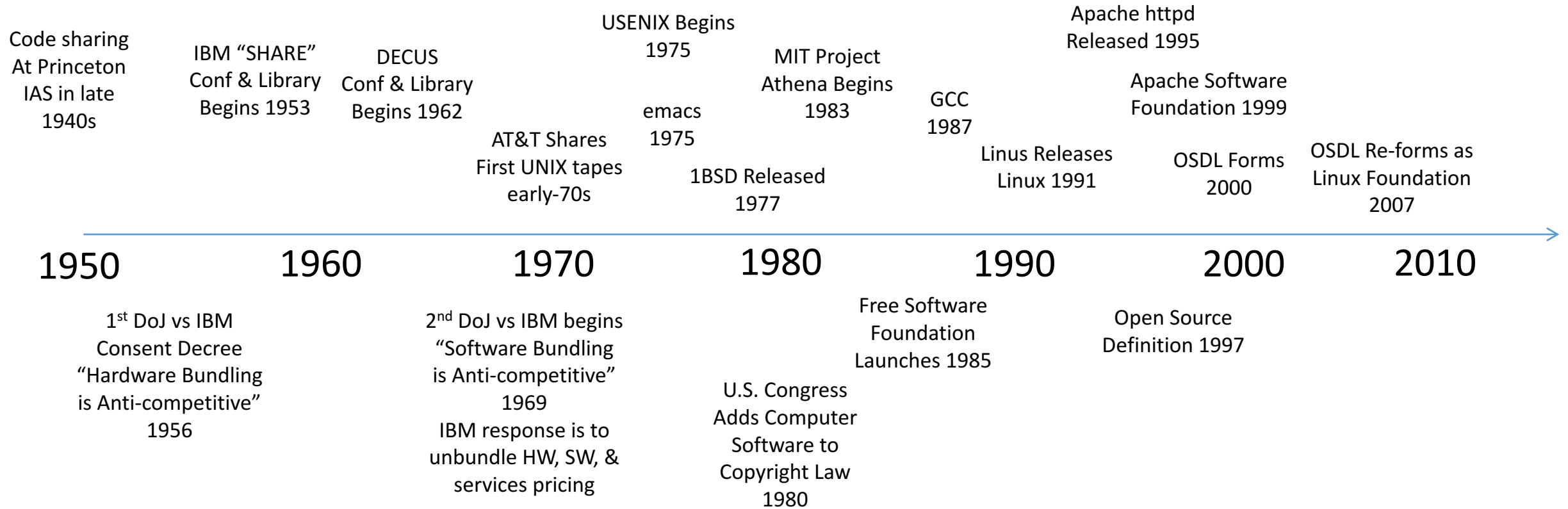
Open Source Software is about Engineering Economics

# We've shared software since we've written software

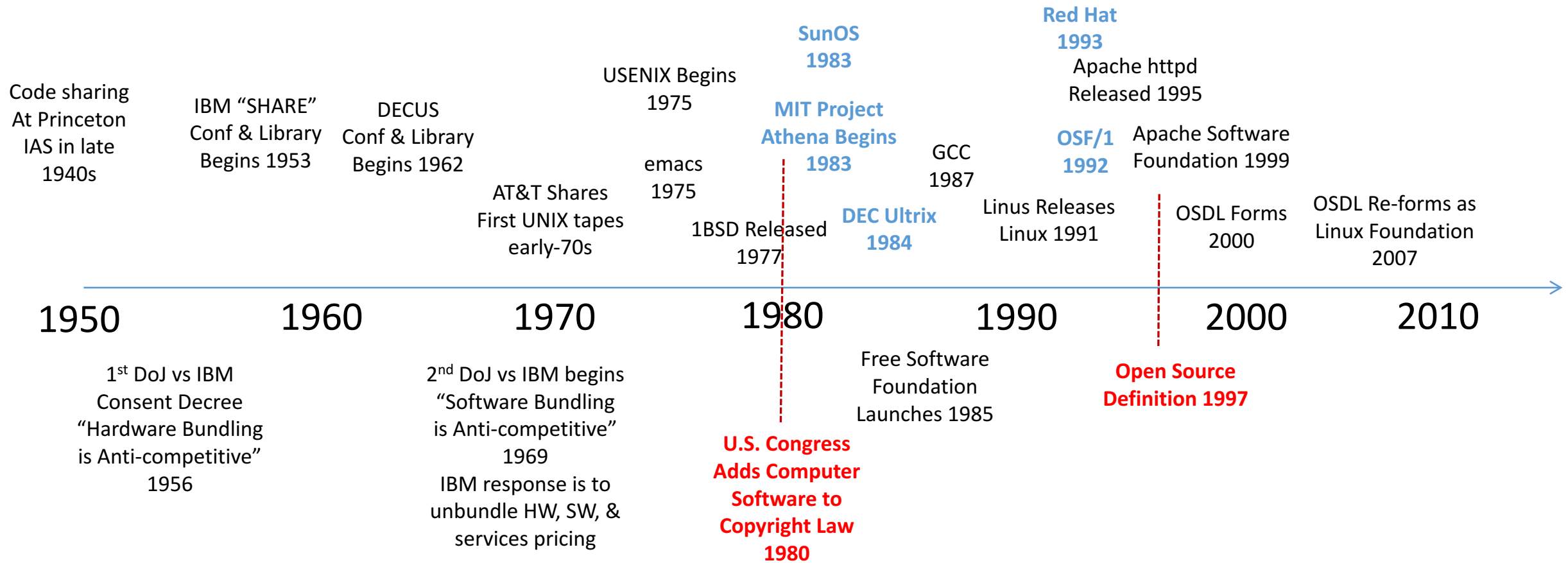


# We've shared software since we've written software

## Writing good software is hard work

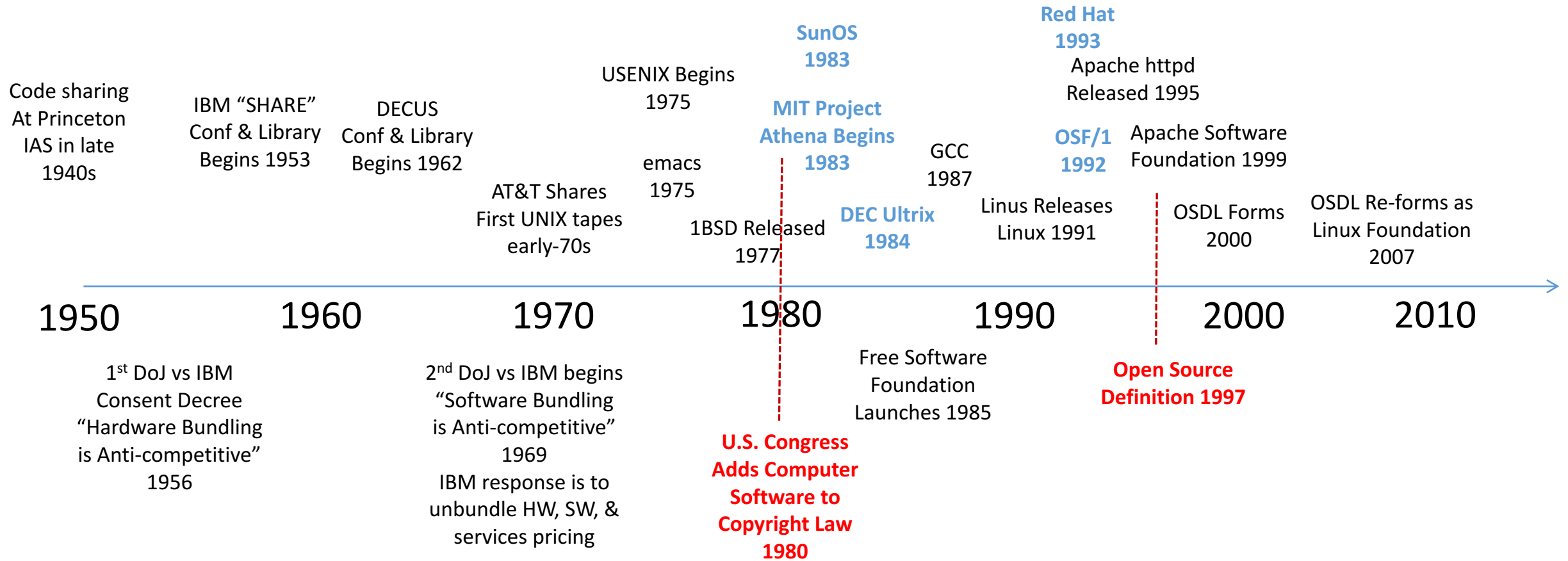


# Companies shared software before we had the OSD





# Companies shared software before we had the OSD (Writing good software is hard work)



Collaboratively-Developed Liberally-Licensed Software is  
about Engineering Economics

Another Story

# A Story of Red Hat in Three CEOs



# A Story of Red Hat in Three CEOs



1994



1999



2008



# A Story of Red Hat in Three CEOs



1994



1999

2008



# A Story of Red Hat in Three CEOs



1994



1999

2008



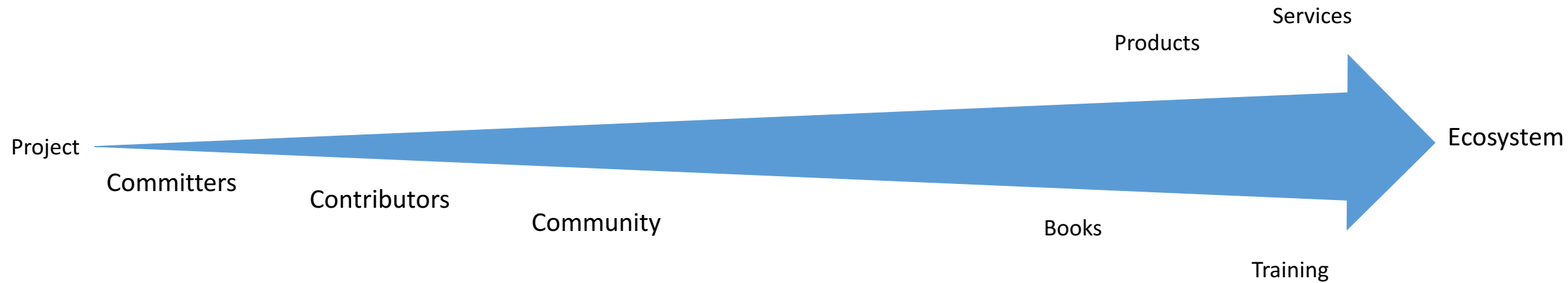
Red Hat is a software company that  
is focused on making its customers successful

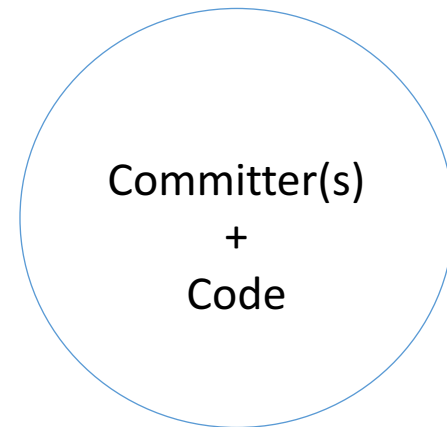
Red Hat is a software company that  
is very good at the engineering economics of open source software

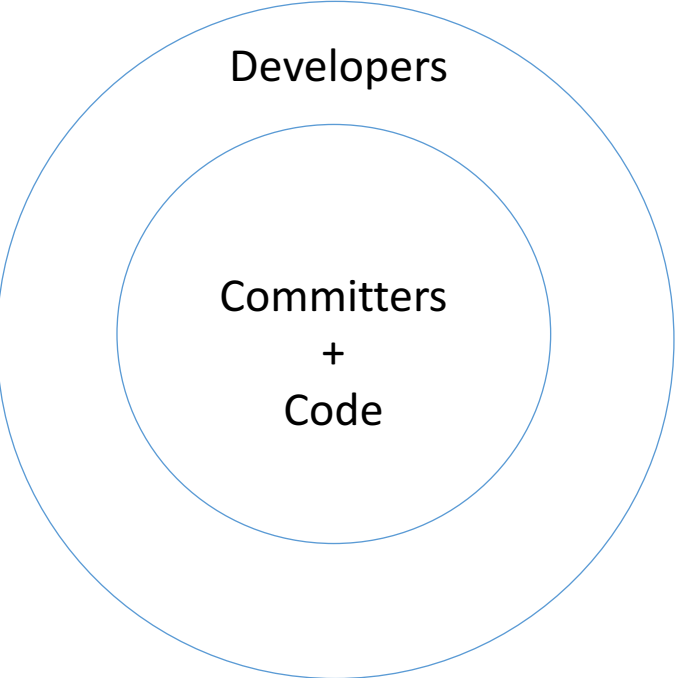


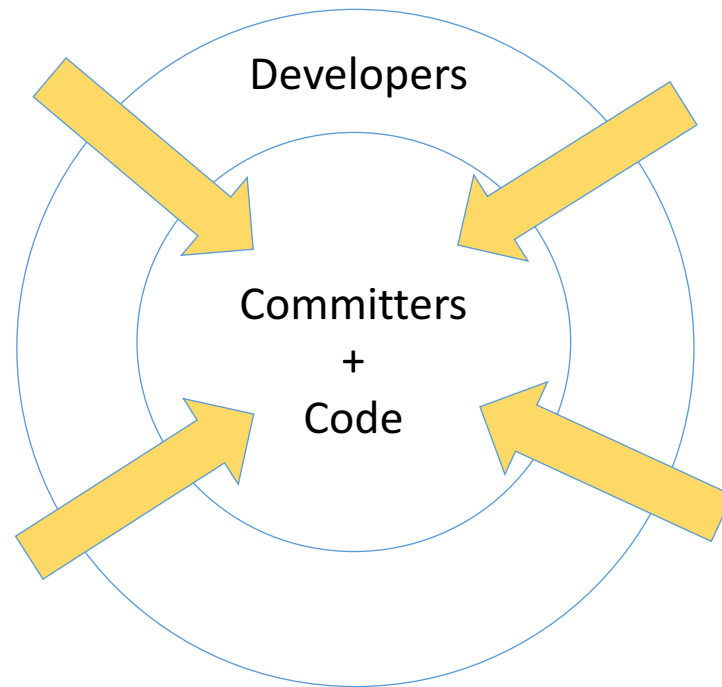
So What About the Business Model?!?!?

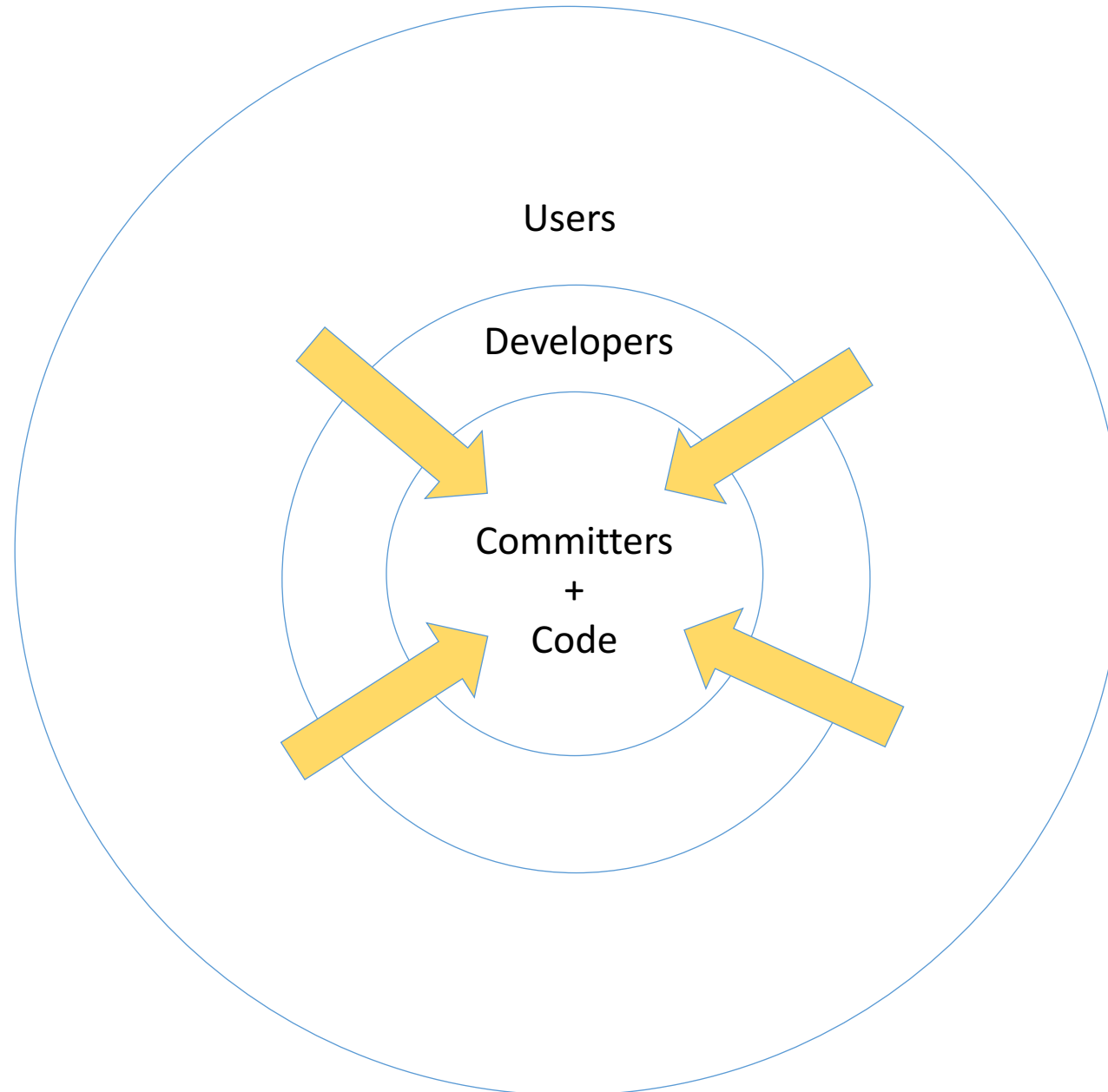
# The Evolution of an Open Source Project

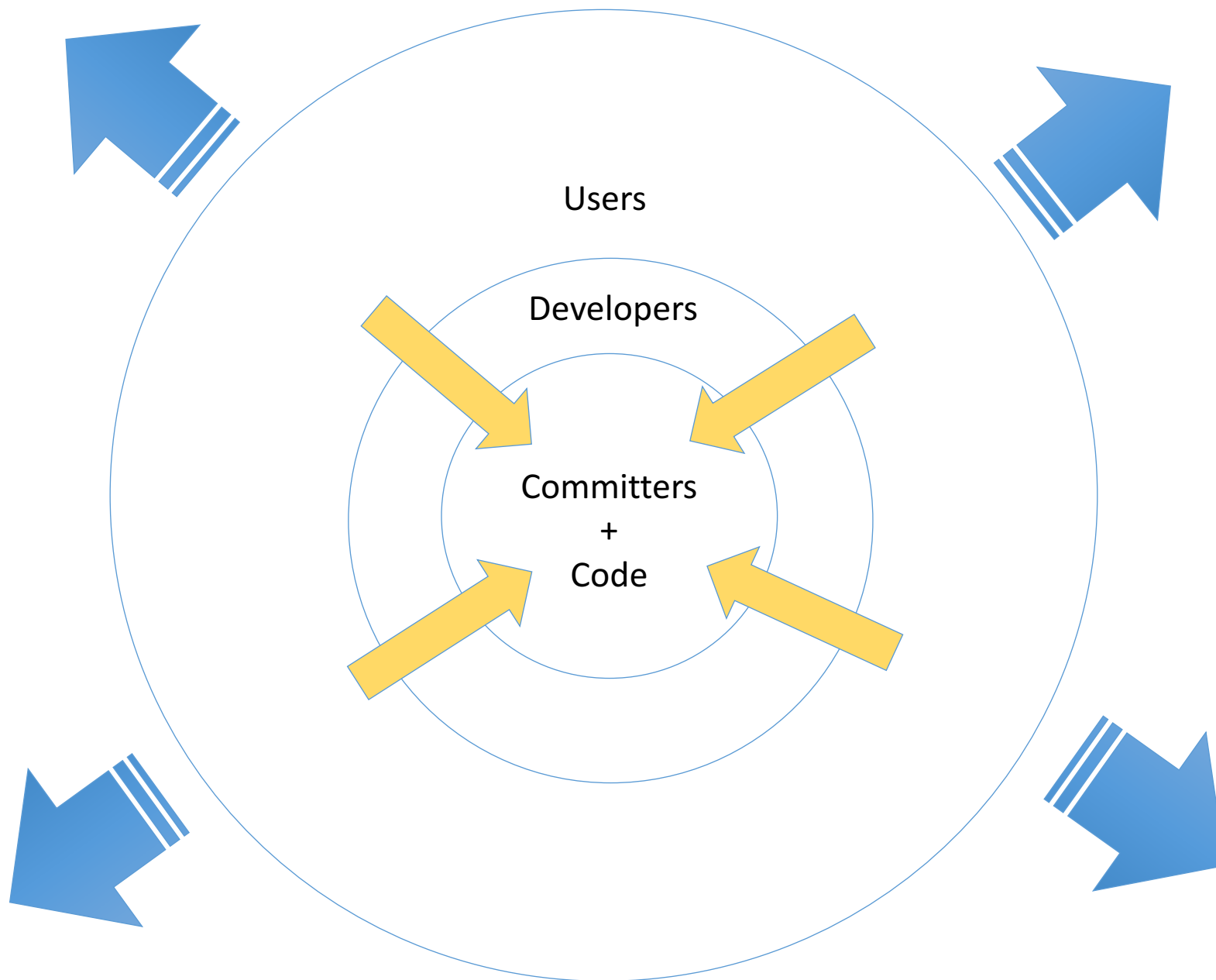




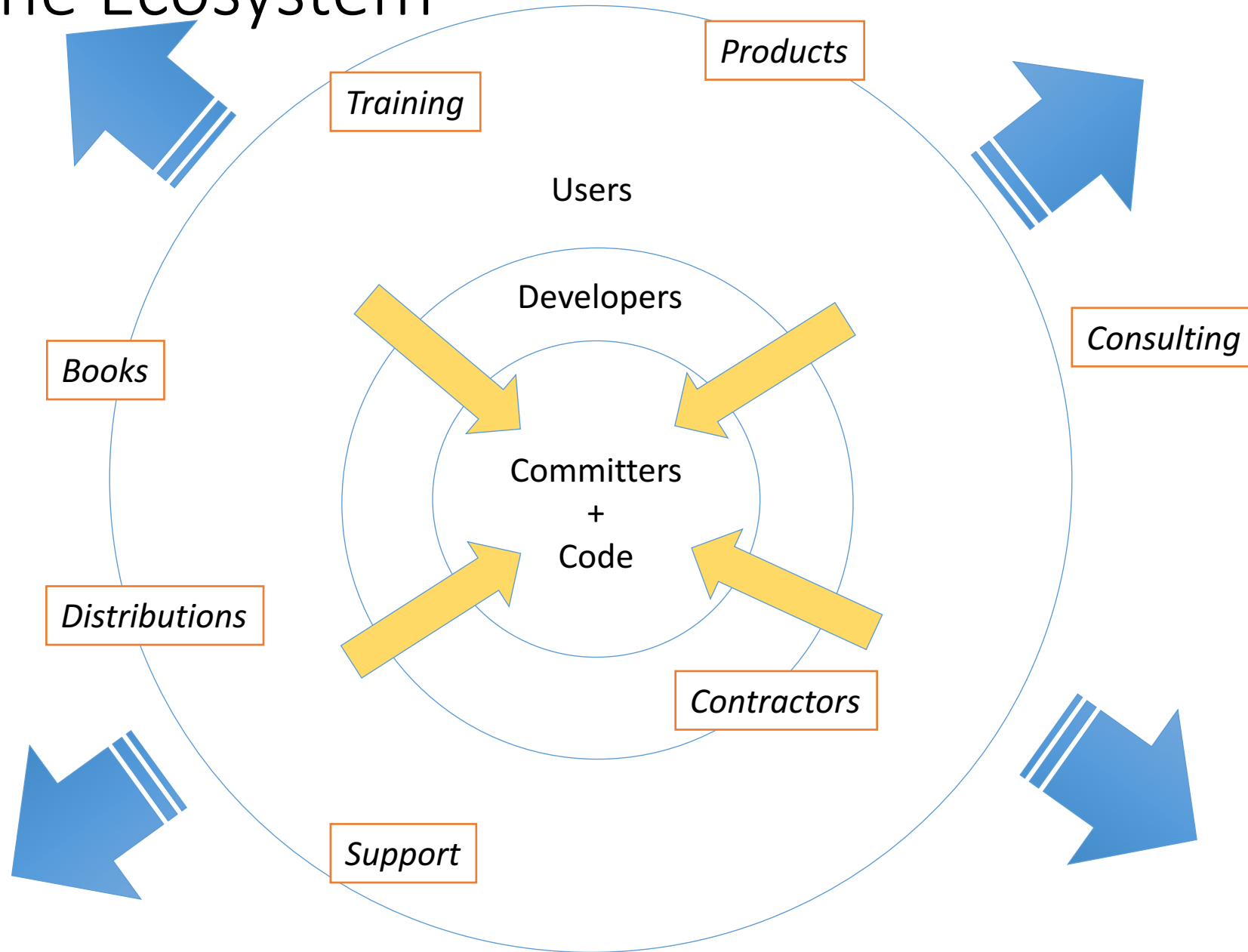








# Building the Ecosystem





**How do you increase your user base?**

**How do you increase your user base?**

(Because that's where you'll find your developers)

**How do you increase your user base?**

(Because that's where you'll find your developers)

**How do you encourage developers?**

(Because these are your future contributors)

**How do you increase your user base?**

(Because that's where you'll find your developers)

**How do you encourage developers?**

(Because these are your future contributors)

**How do you make it easy to contribute?**

(Because this is the growth and success of your community)

## **How do you increase your user base?**

(How do you make it easy to install/configure/use the software?)

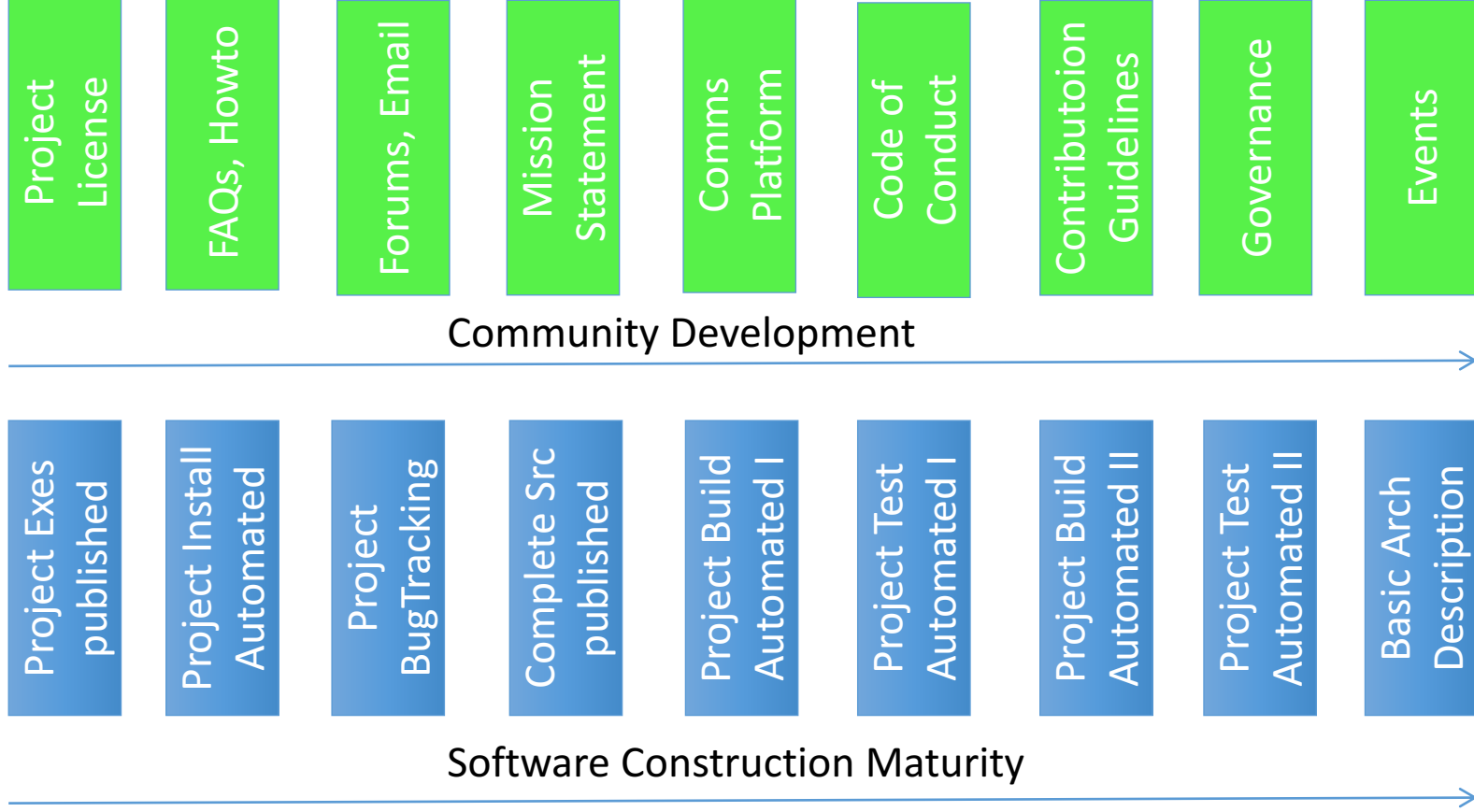
## **How do you encourage developers?**

(How do you make it easy to build/test/experiment?)

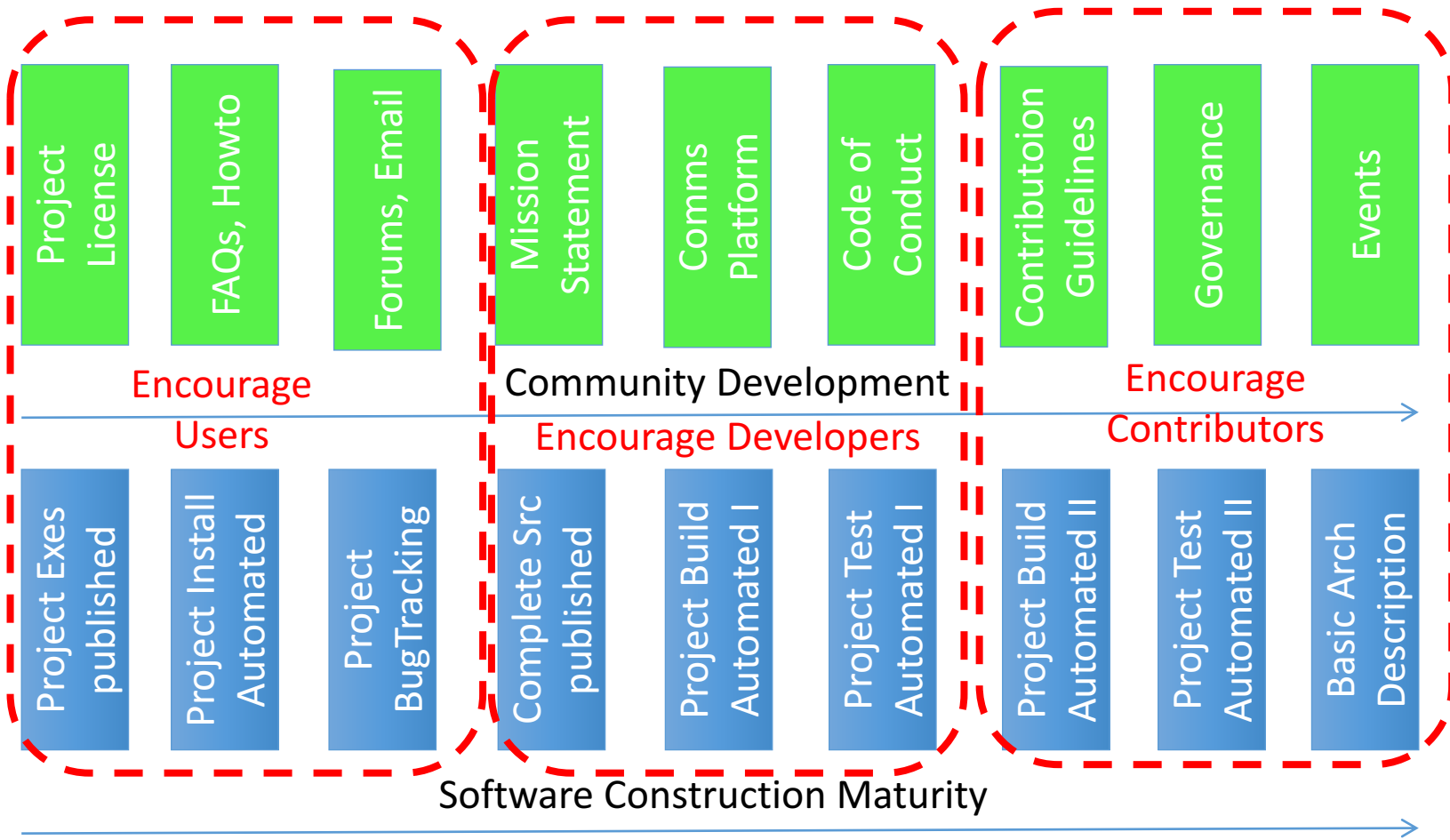
## **How do you make it easy to contribute?**

(What do you communicate to your community)

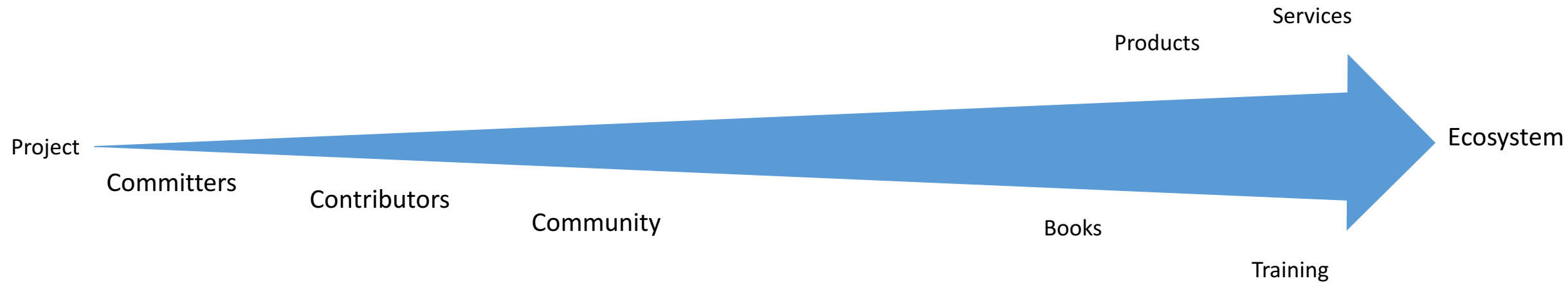
# Open Source Community Practices



# Open Source Community Patterns

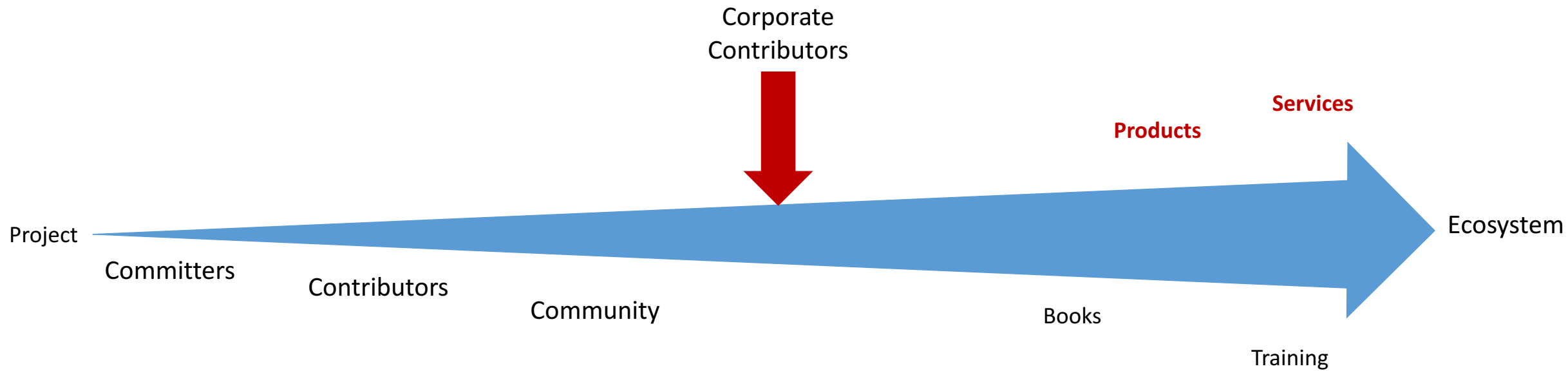


# The Evolution of an Open Source Project

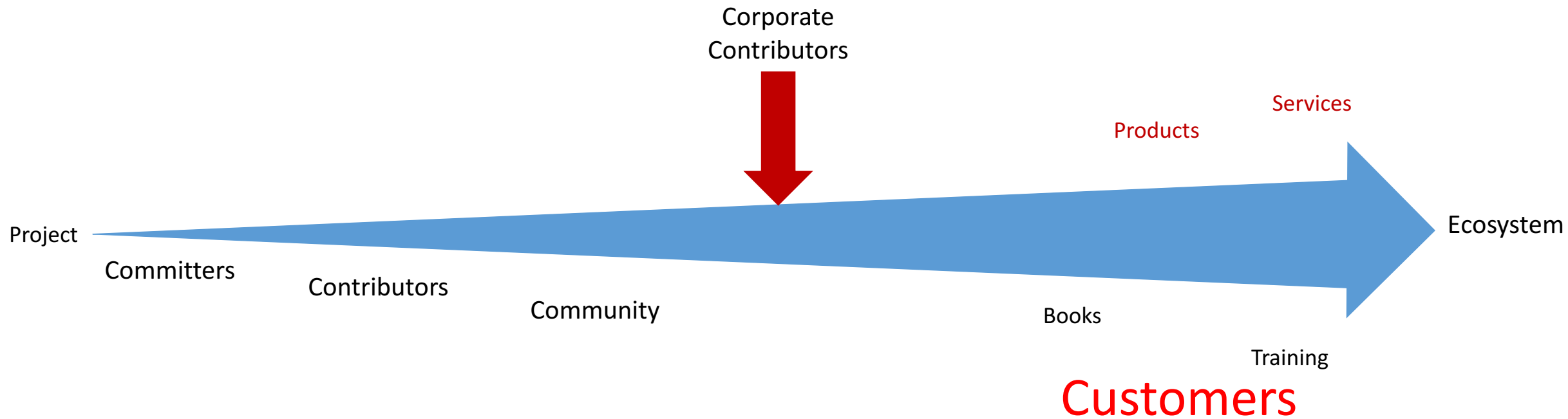




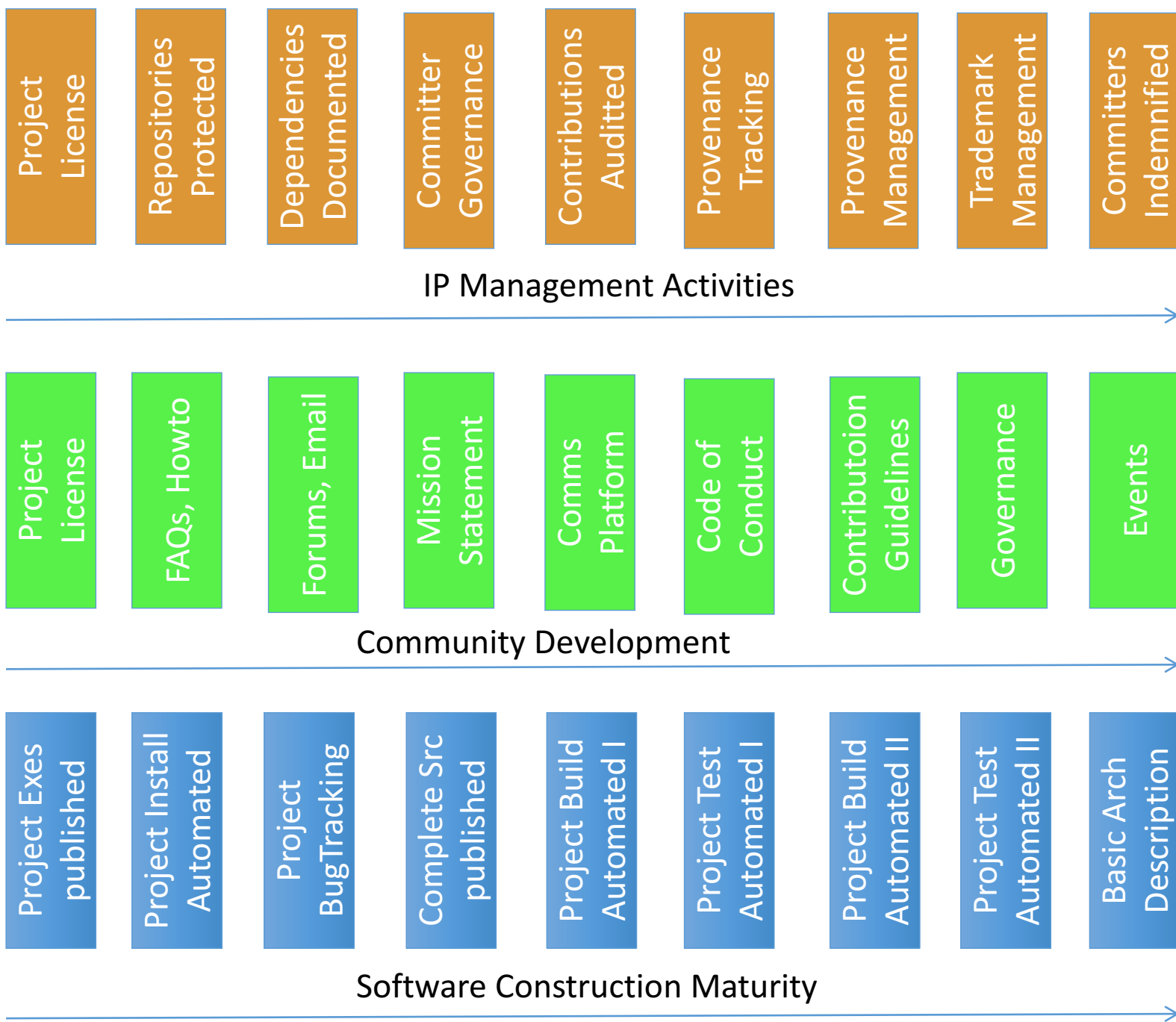
# The Evolution of an Open Source Project



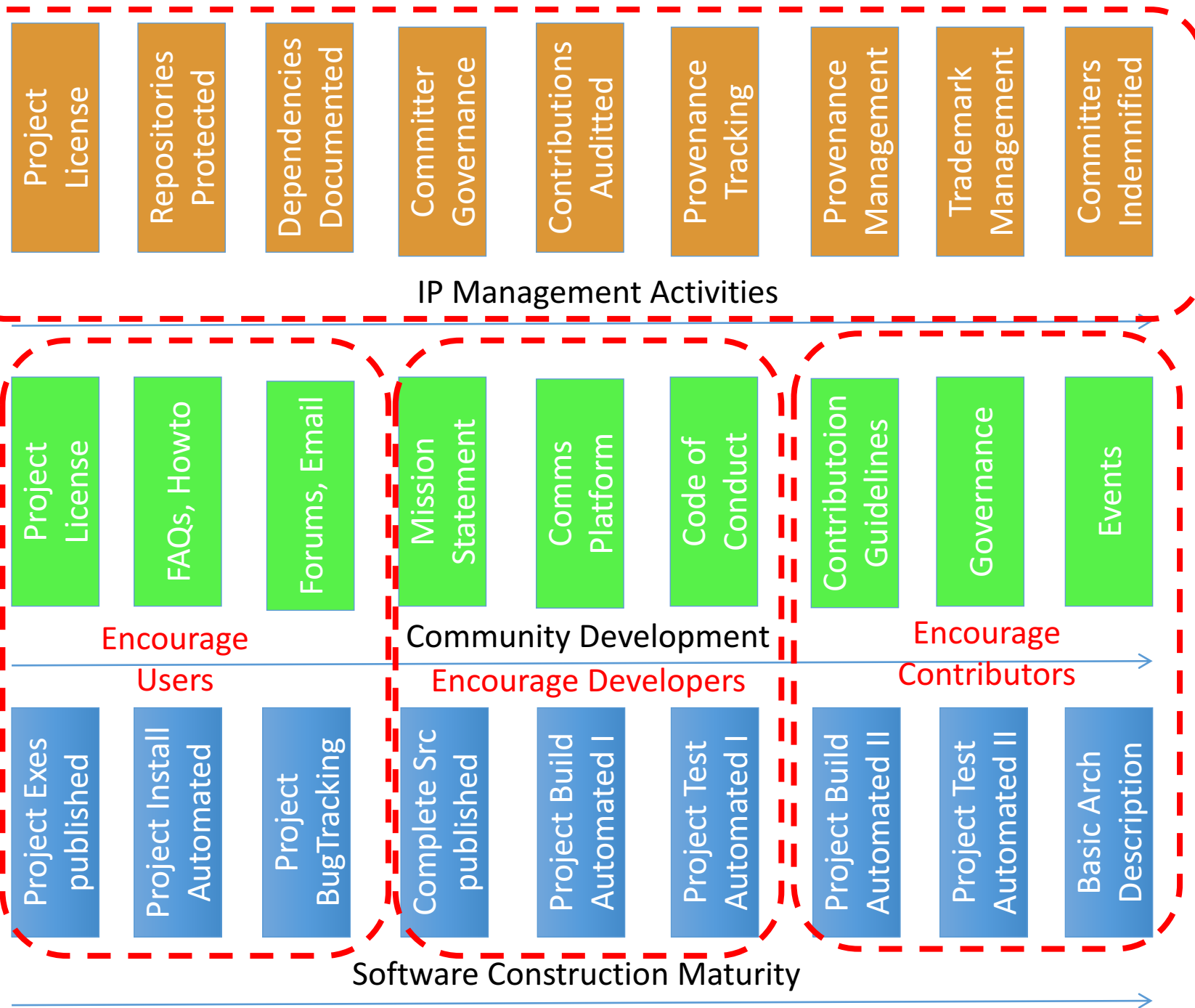
# The Evolution of an Open Source Project



# Open Source Community Practices

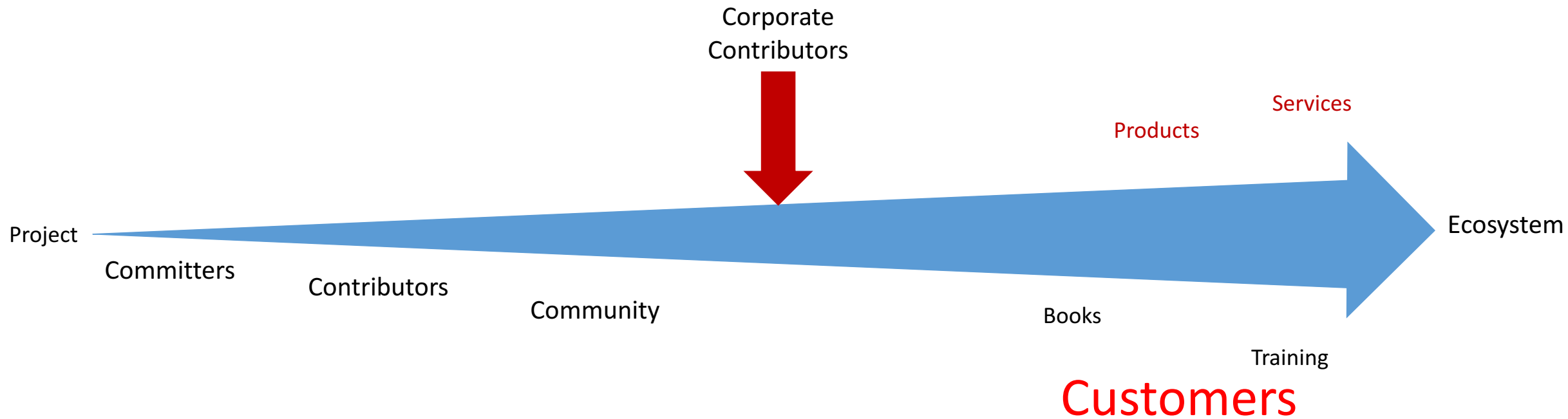


# Open Source Community Patterns



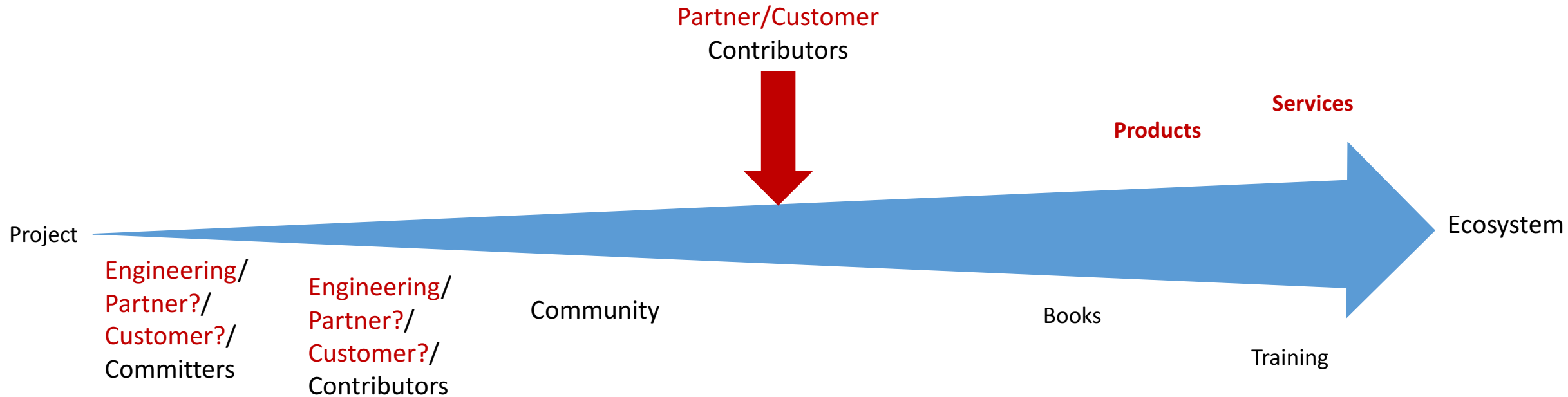
Encourage/  
Manage Corp  
Contributions

# The Evolution of an Open Source Project



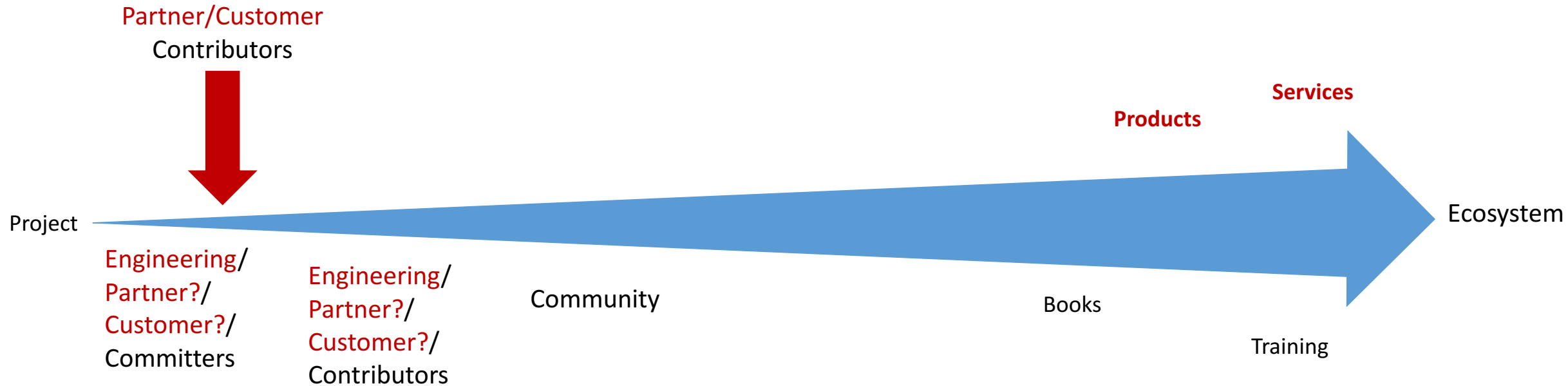
TRANSITION

# The Evolution of a Corporate Open Source Project



Setting Customer and Partner Expectations in Community is **Critical**

# The Reality of a Corporate Open Source Project



Setting Customer and Partner Expectations in Community is **Critical**



## **Customers versus Community**

(Money vs. Time; Expectations are different; Conversations are different)

## **Customers versus Community**

(Money vs. Time; Expectations are different; Conversations are different)

## **Partners versus Community**

(Don't mix business with community)

## **Customers versus Community**

(Money vs. Time; Expectations are different; Conversations are different)

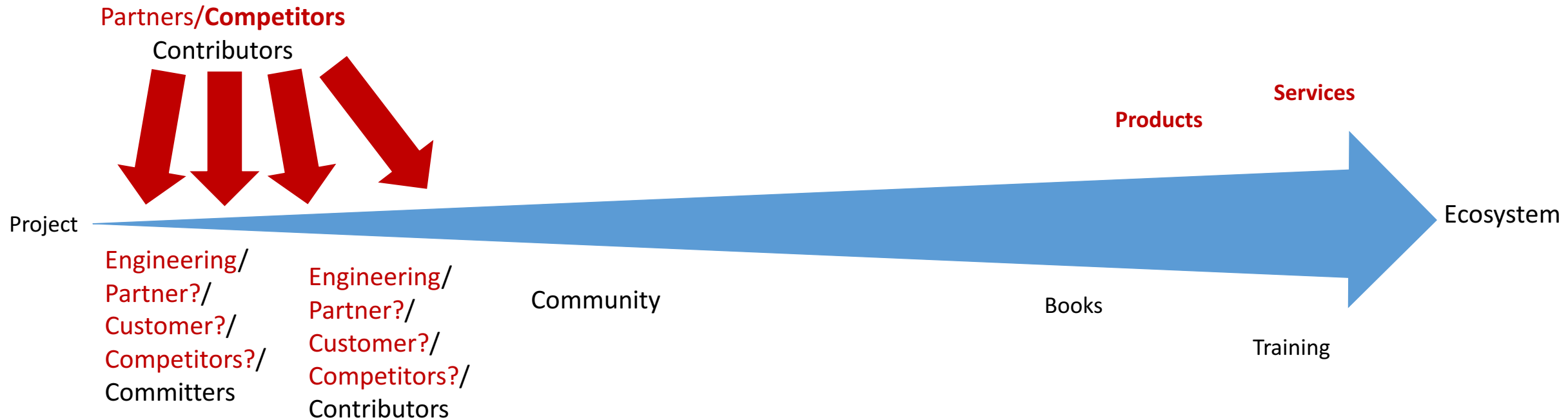
## **Partners versus Community**

(Don't mix business with community)

## **Products versus Projects**

(Success metrics are different; OKRs and KPIs are different)

# The Reality of Corporate Open Source Projects



Setting Customer and Partner Expectations in Community is **Critical**

# **Collaboratively-Developed Liberally-Licensed Software is an Engineering Economic Imperative**

(We've Shared Software Since We've Written Software)

# **Collaboratively-Developed Liberally-Licensed Software is an Engineering Economic Imperative**

(We've Shared Software Since We've Written Software)

**There is no Open Source Business Model**

(Good Companies Solve Customer Problems)

# **Collaboratively-Developed Liberally-Licensed Software is an Engineering Economic Imperative**

(We've Shared Software Since We've Written Software)

## **There is no Open Source Business Model**

(Good Companies Solve Customer Problems)

## **Successful Open Source Projects Follow Well Known Patterns**

(No Scale without Discipline)

# **Collaboratively-Developed Liberally-Licensed Software is an Engineering Economic Imperative**

(We've Shared Software Since We've Written Software)

## **There is no Open Source Business Model**

(Good Companies Solve Customer Problems)

## **Successful Open Source Projects Follow Well Known Patterns**

(No Scale without Discipline)

## **Don't Confuse Customers, Partners, Community, Products & Projects**

(Corporate Open Source Projects Require Different Thinking)



How do you make money when you give away the product for free?

~~How do you make money when you give away the product for free?~~

What problem are we solving for our customers?

What problem are we solving for our customers?  
How do we make our customers successful?

What problem are we solving for our customers?

How do we make our customers successful?

What can we do to make our community healthier?

What problem are we solving for our customers?

How do we make our customers successful?

What can we do to make our community healthier?

How do we make our community successful?

# Questions & Comments

stephen r walli

stephen.walli@gmail.com

@stephenrwalli

<http://stephesblog.blogs.com>

<http://opensource.com>

<https://medium.com/@stephenrwalli/>

**No Scale without Discipline**  
(This is an Attribute of Software)

**Orders of Magnitude**  
(1000:100:10:1)

**Freeloaders are important!**  
(It means you're doing it right)