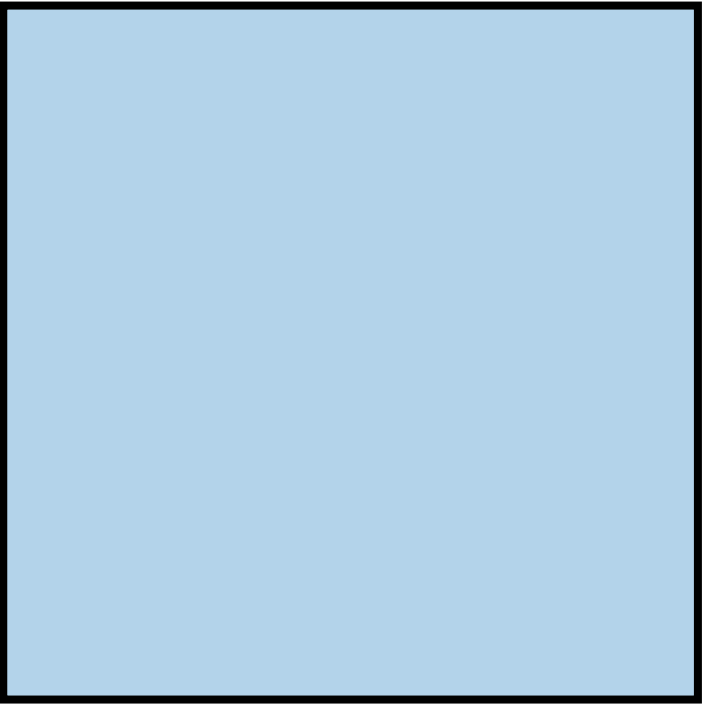
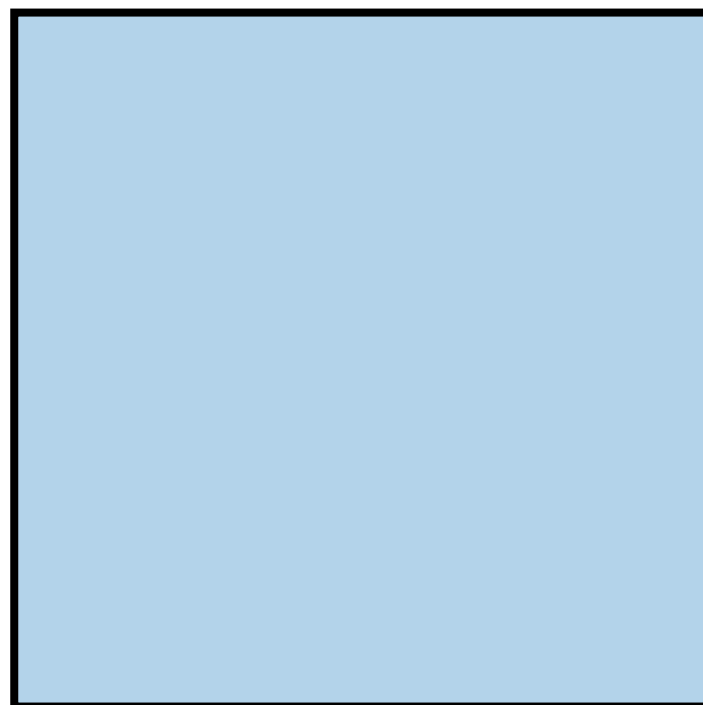
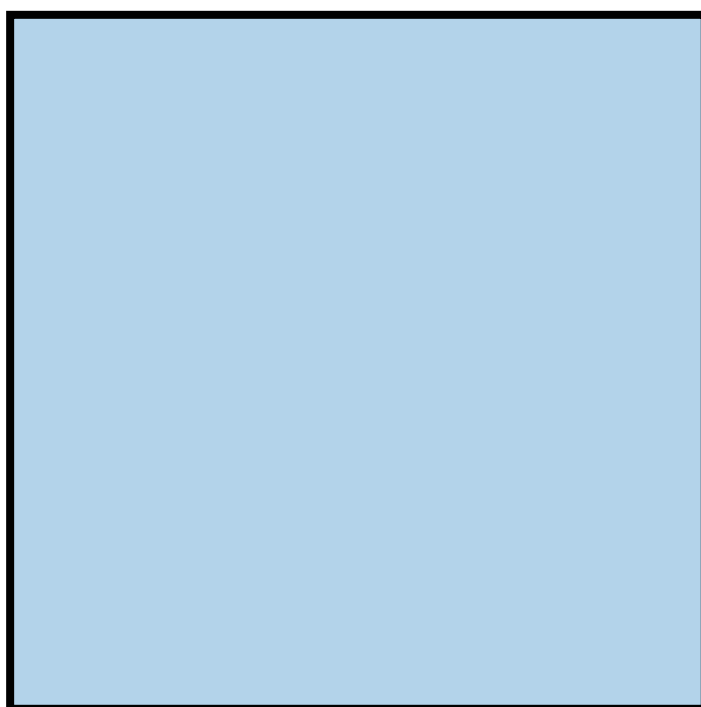


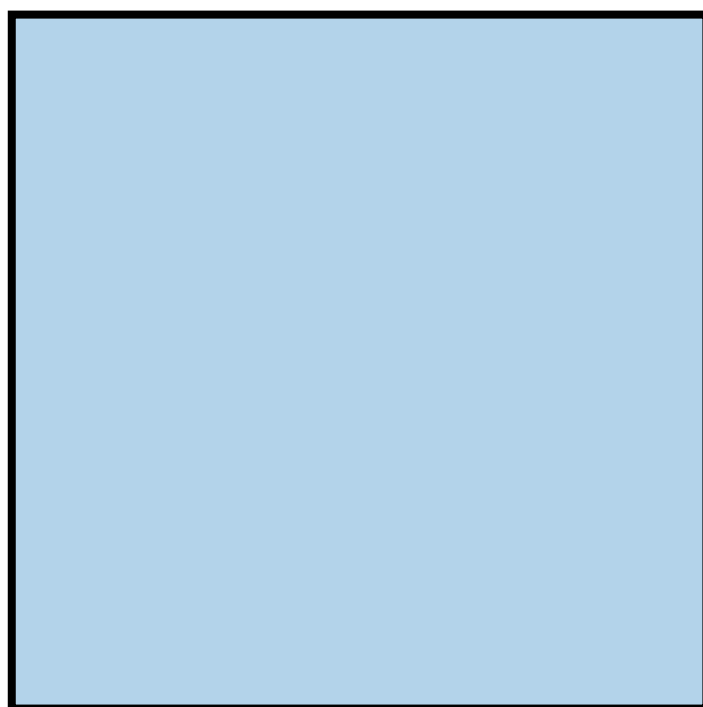
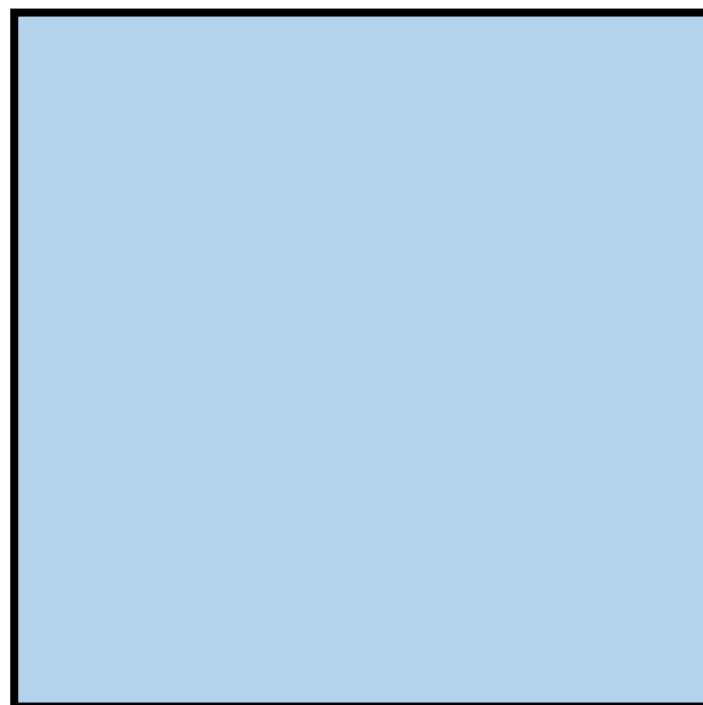
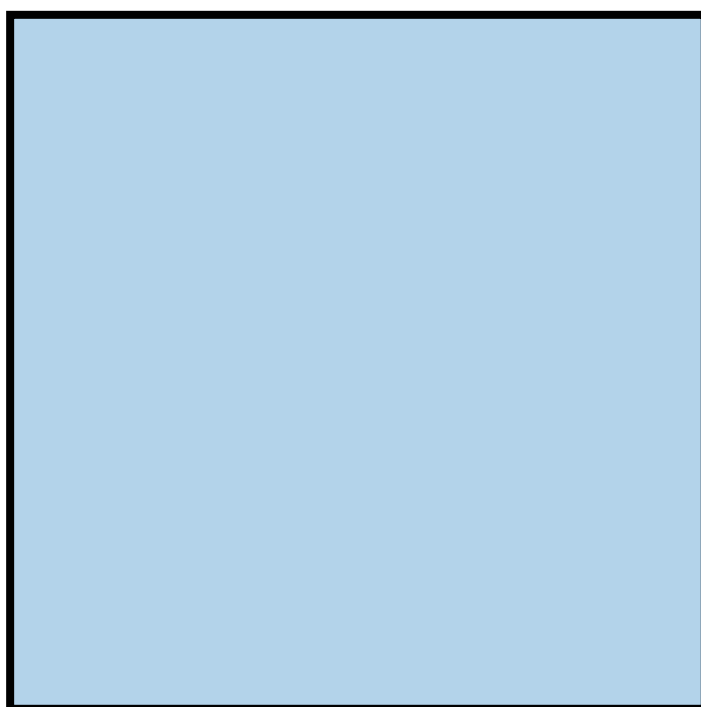
A still from the movie 'The Italian Job' showing Sam Newman (left) with long curly hair and a mustache, wearing a brown jacket over a light green shirt, looking down with a concerned expression. To his right is a bald man with dark hair on the sides, wearing a dark patterned jacket, looking up at him. The background is a dark, rocky, and uneven terrain.

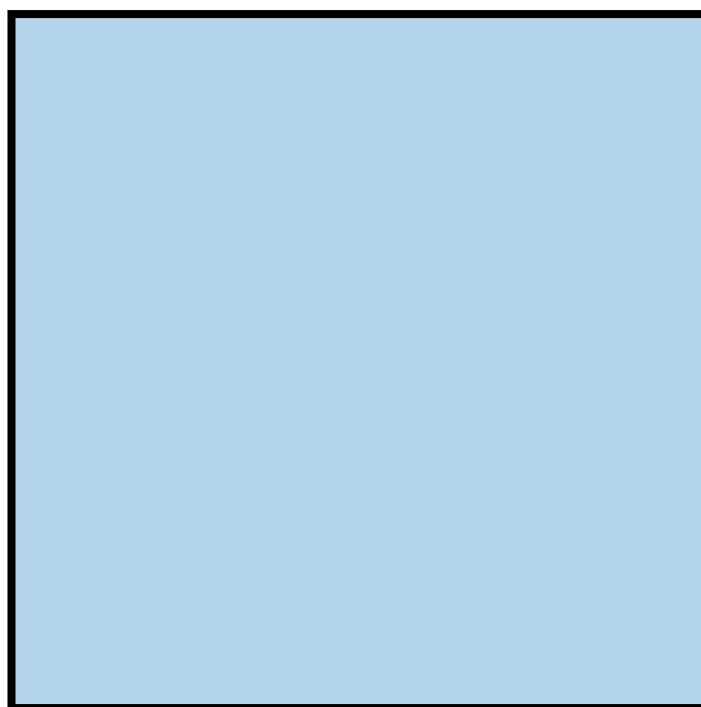
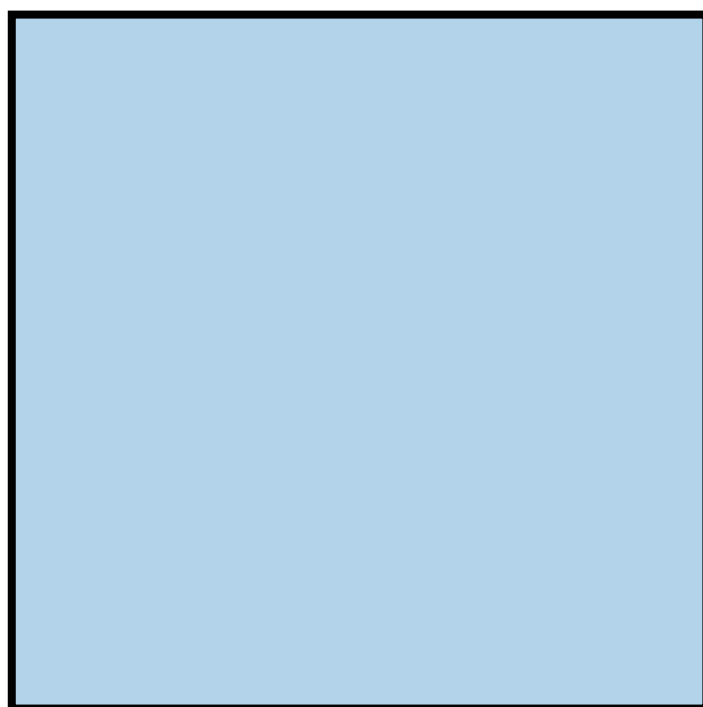
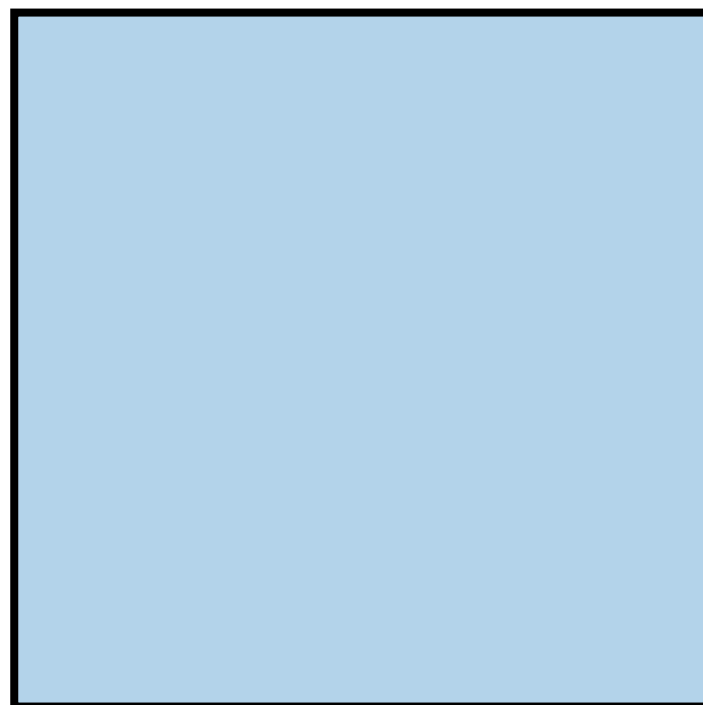
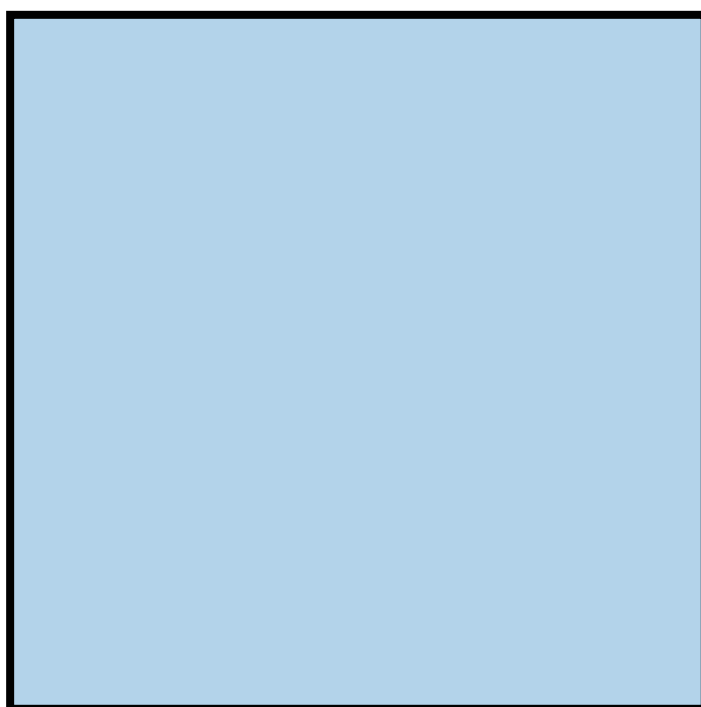
YOU KEEP USING THAT WORD

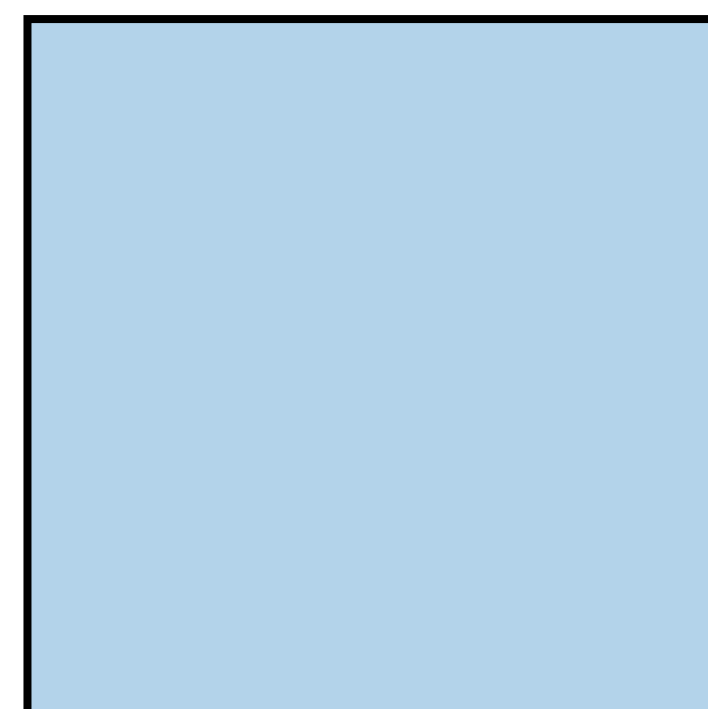
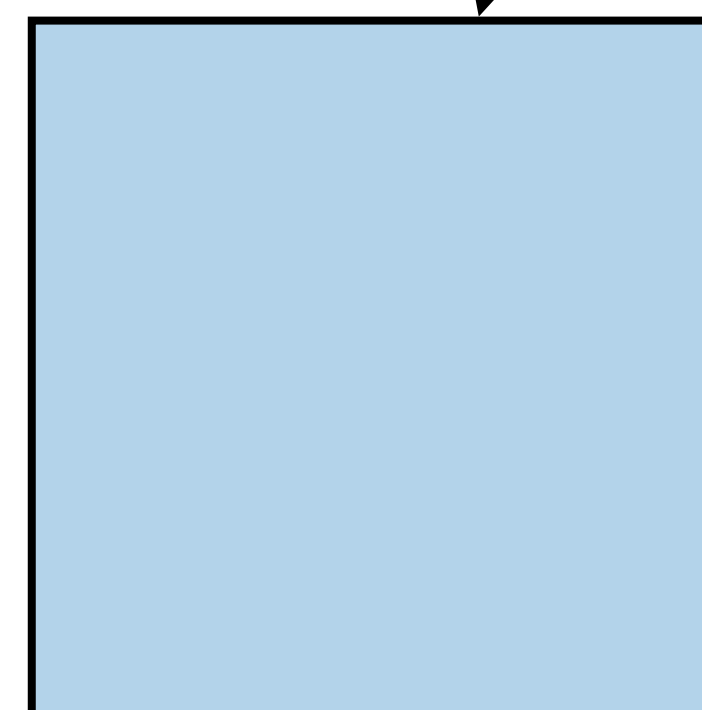
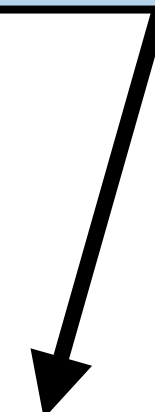
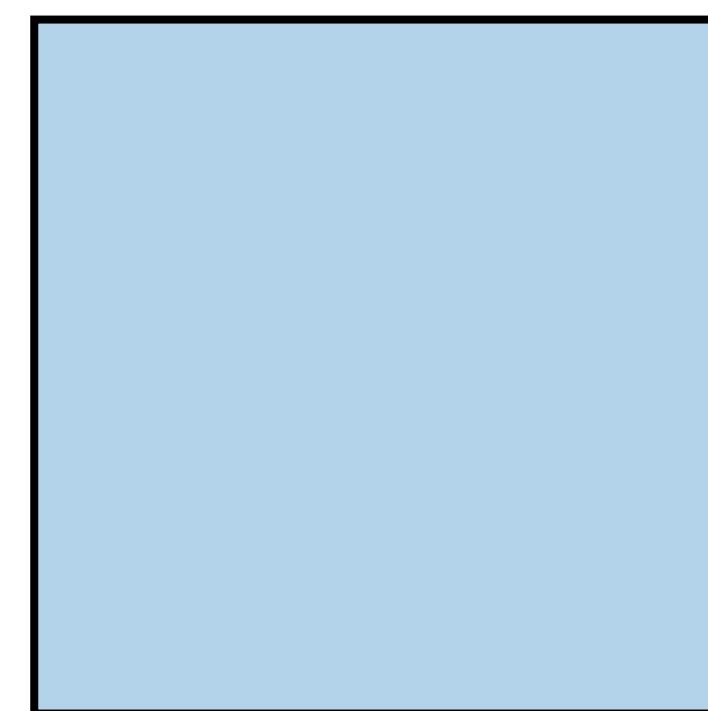
Sam Newman

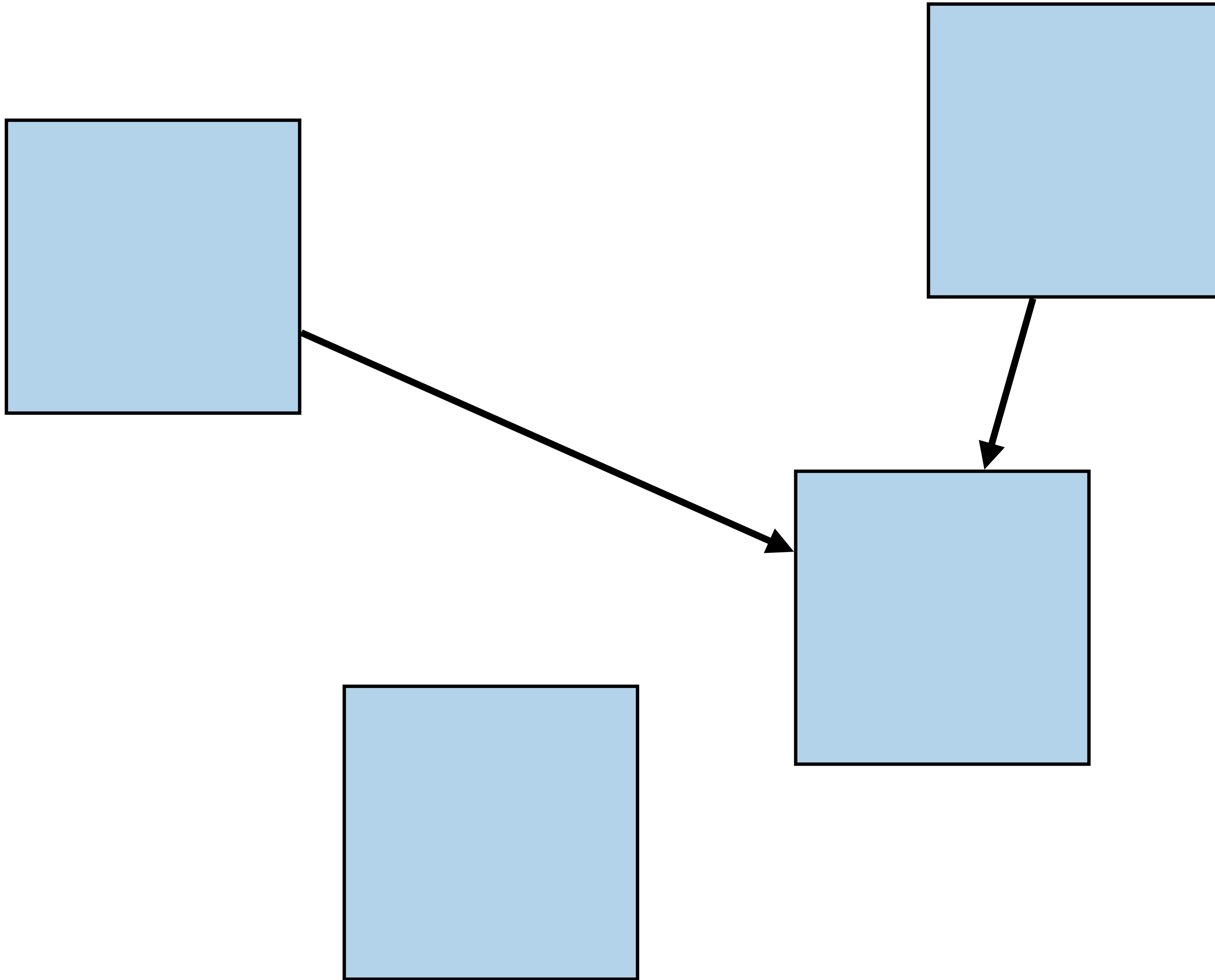


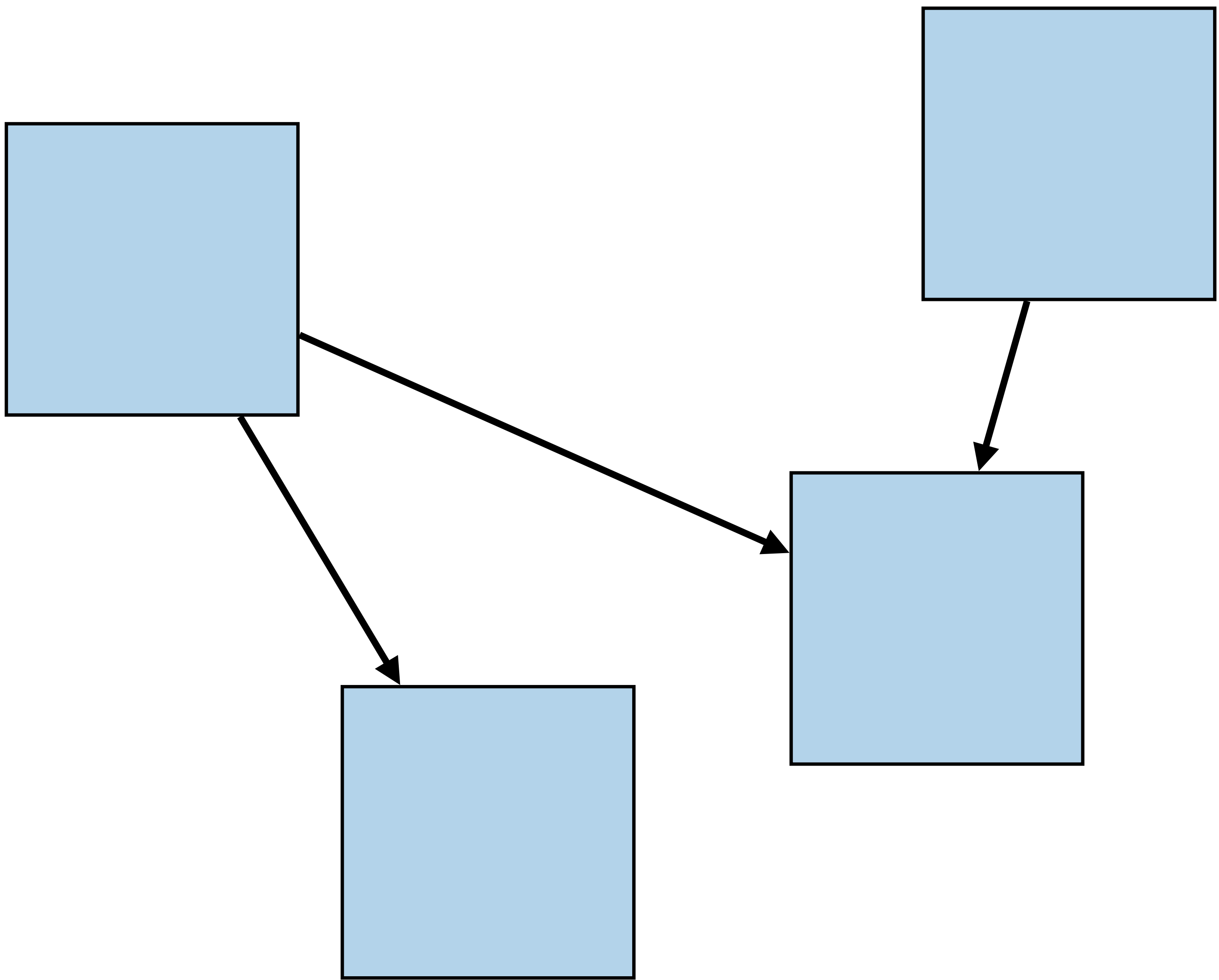


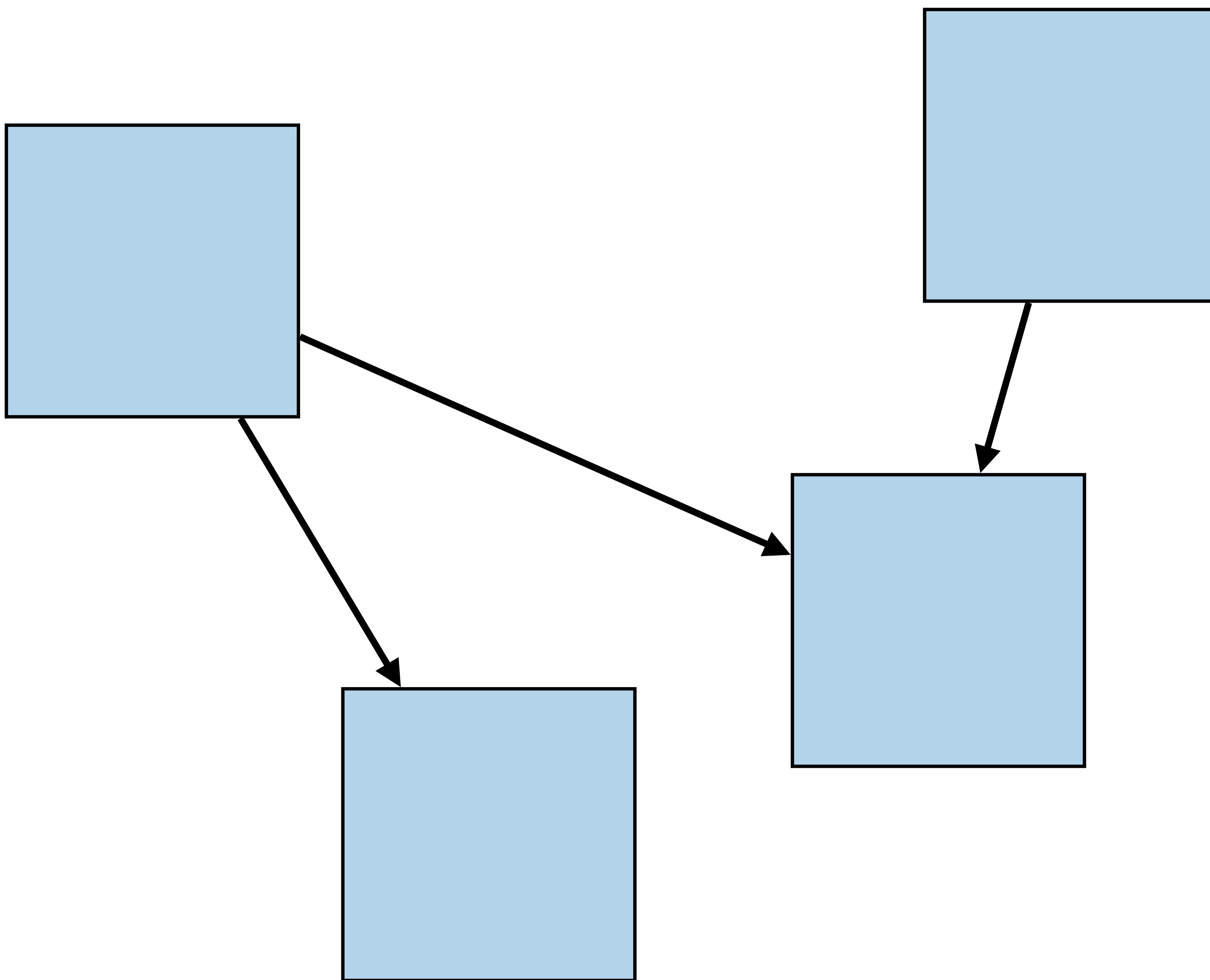


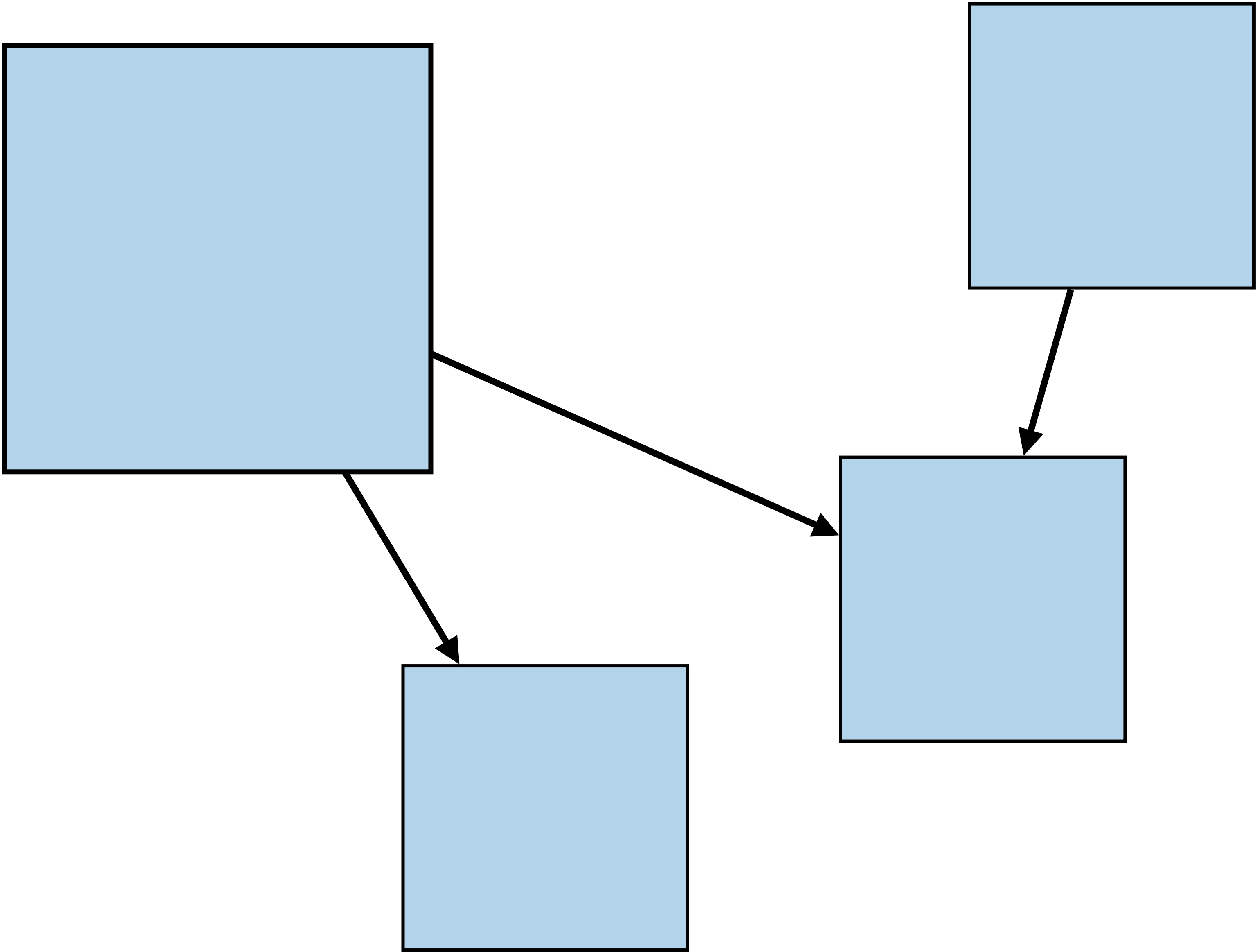


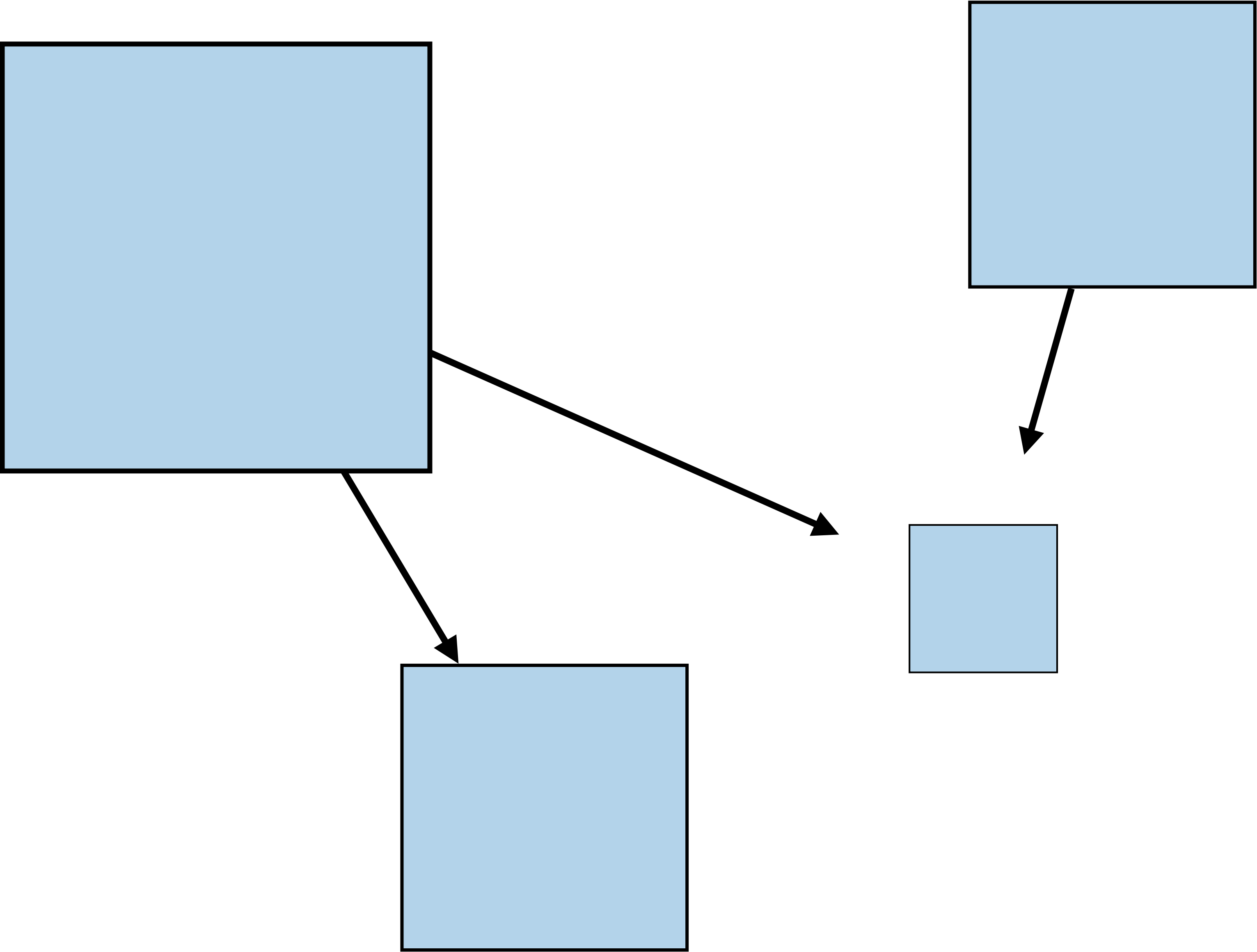


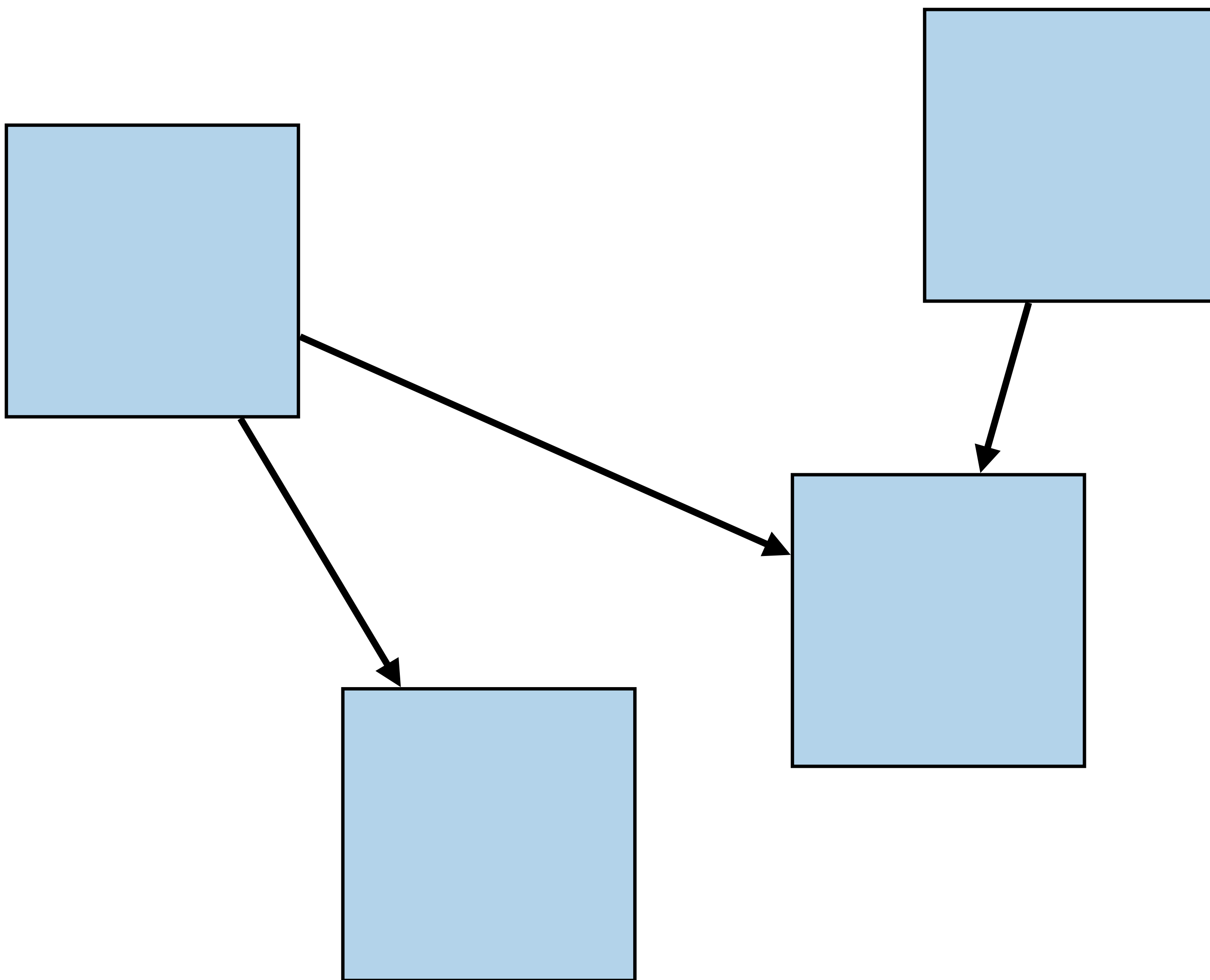


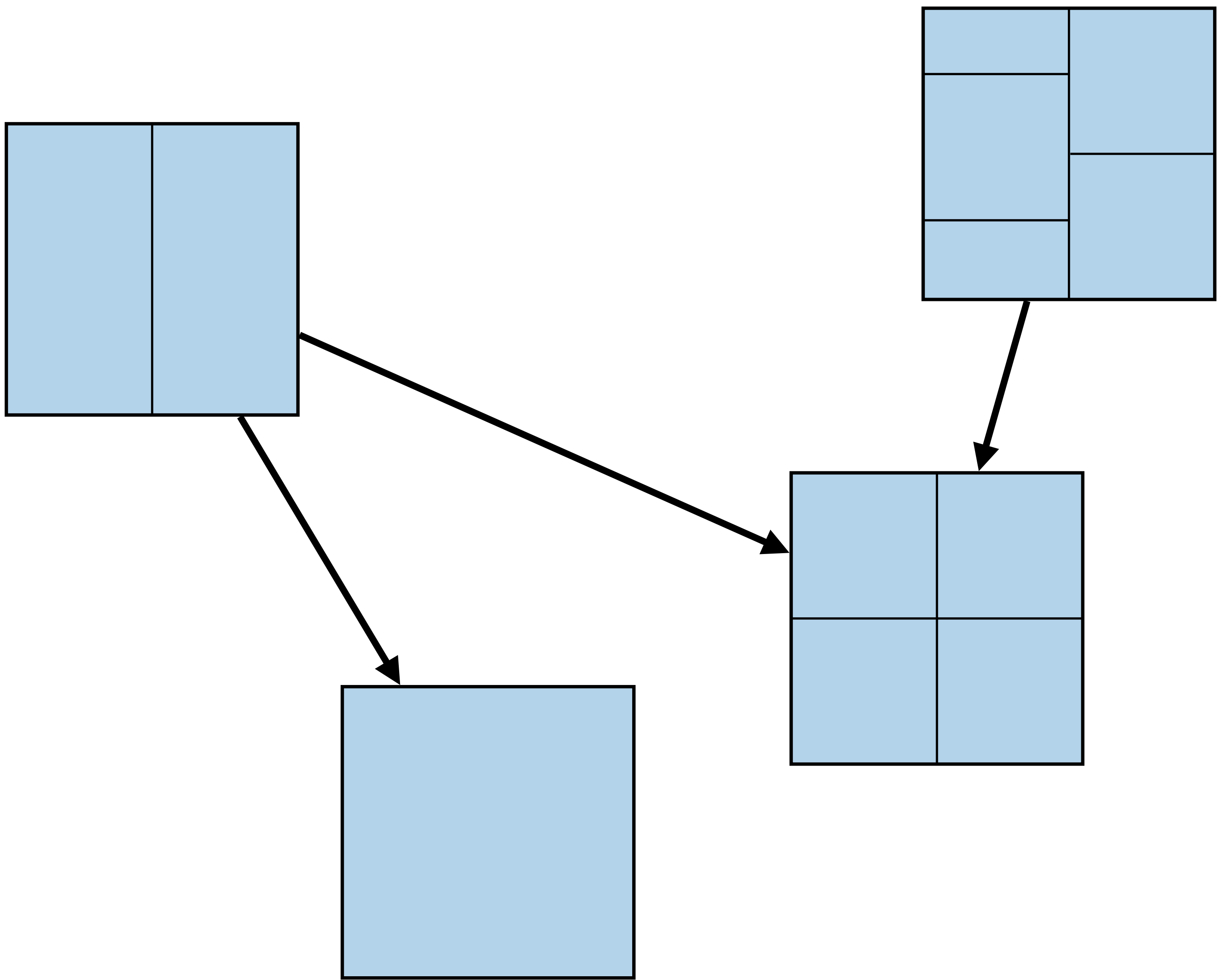


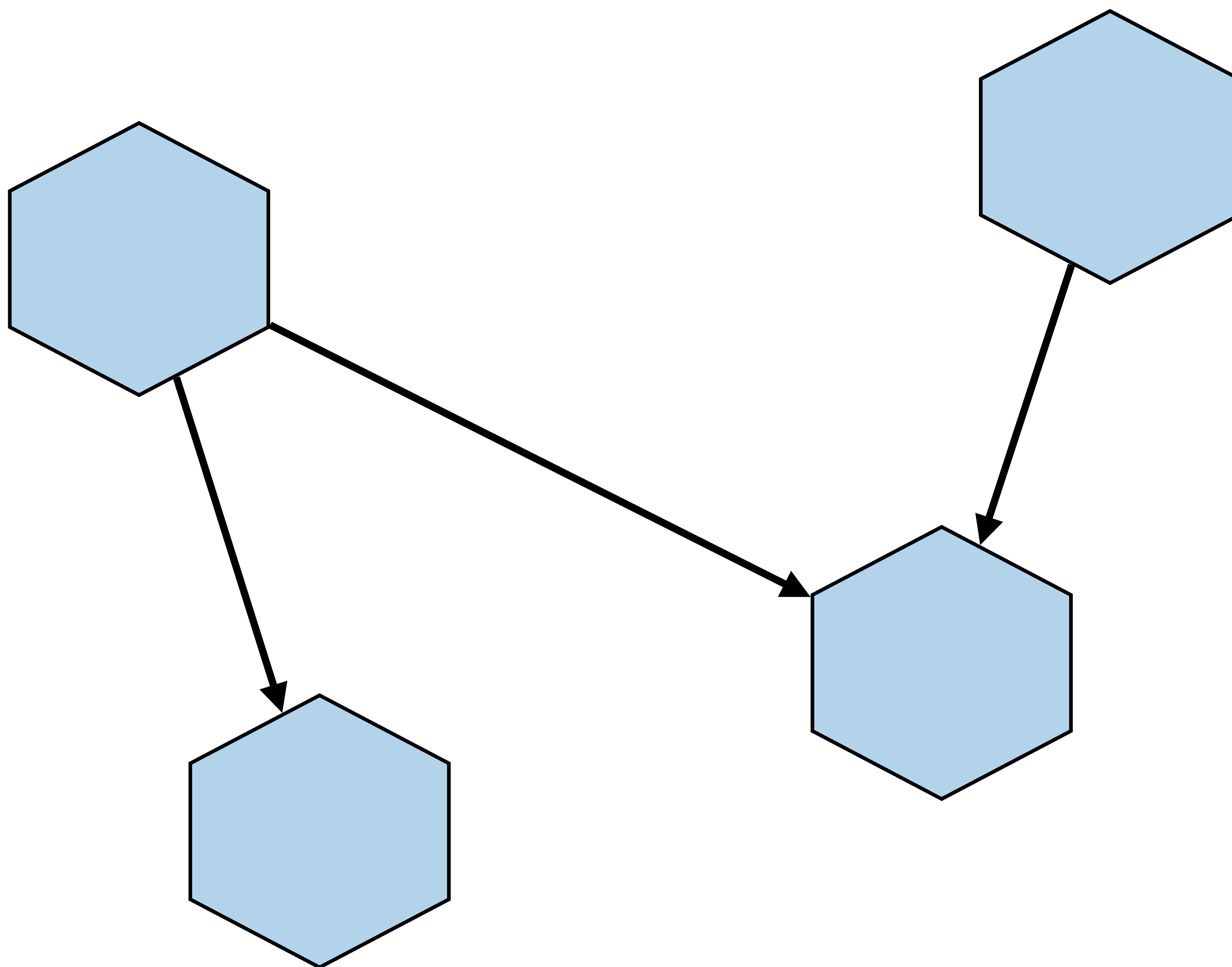


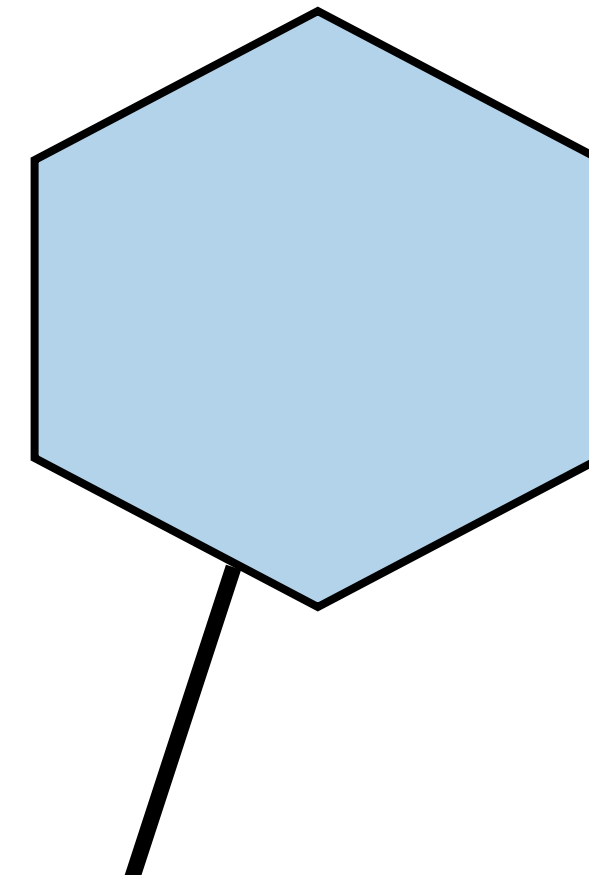
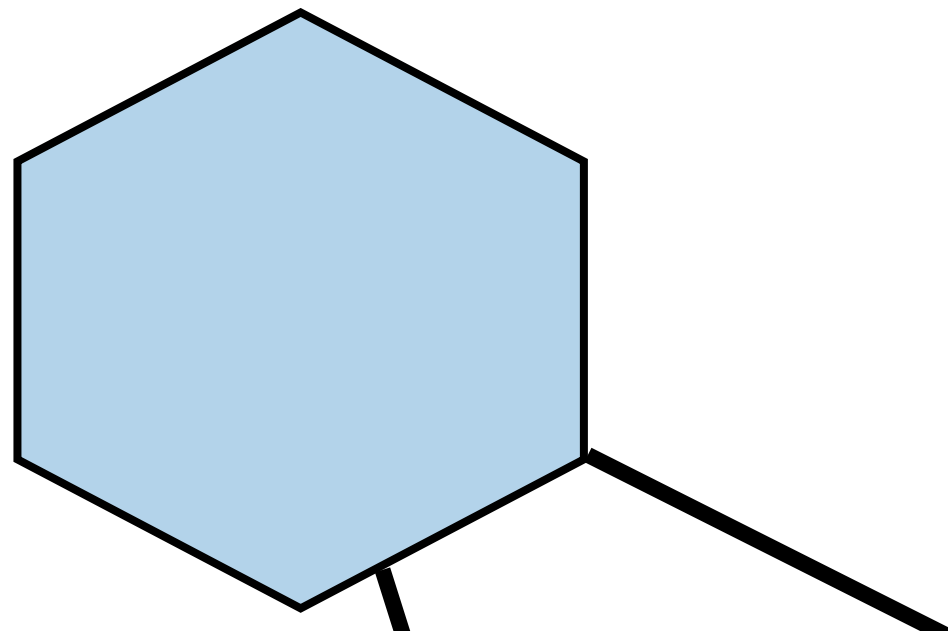




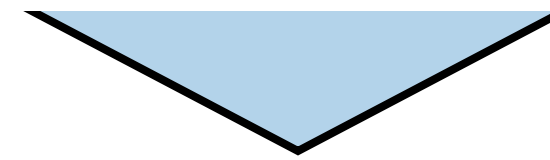
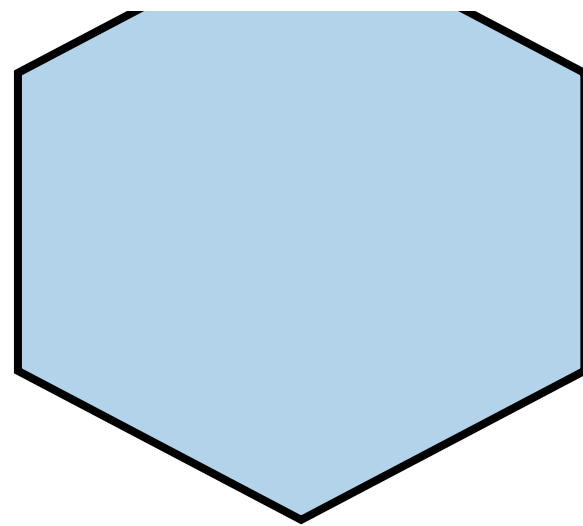


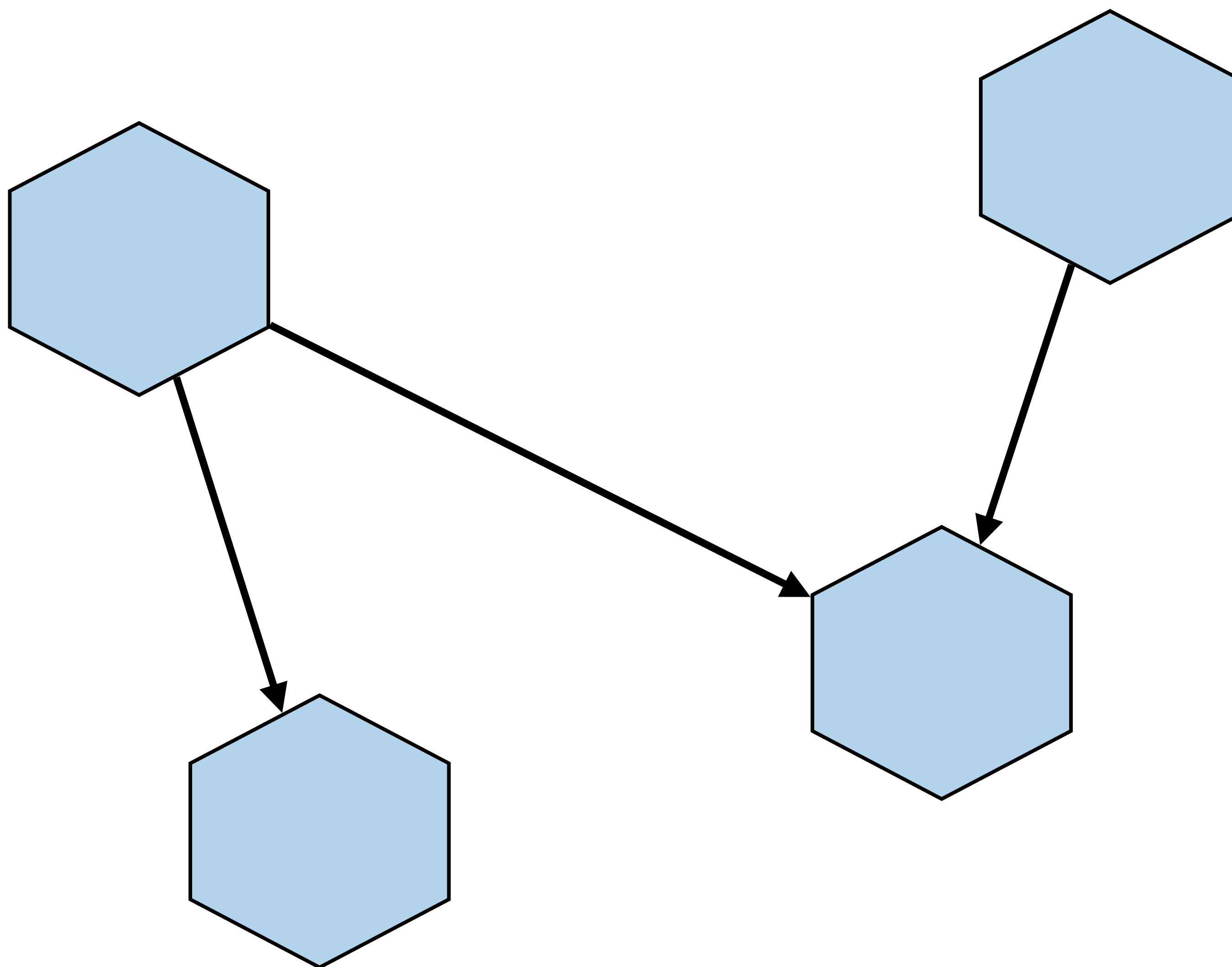


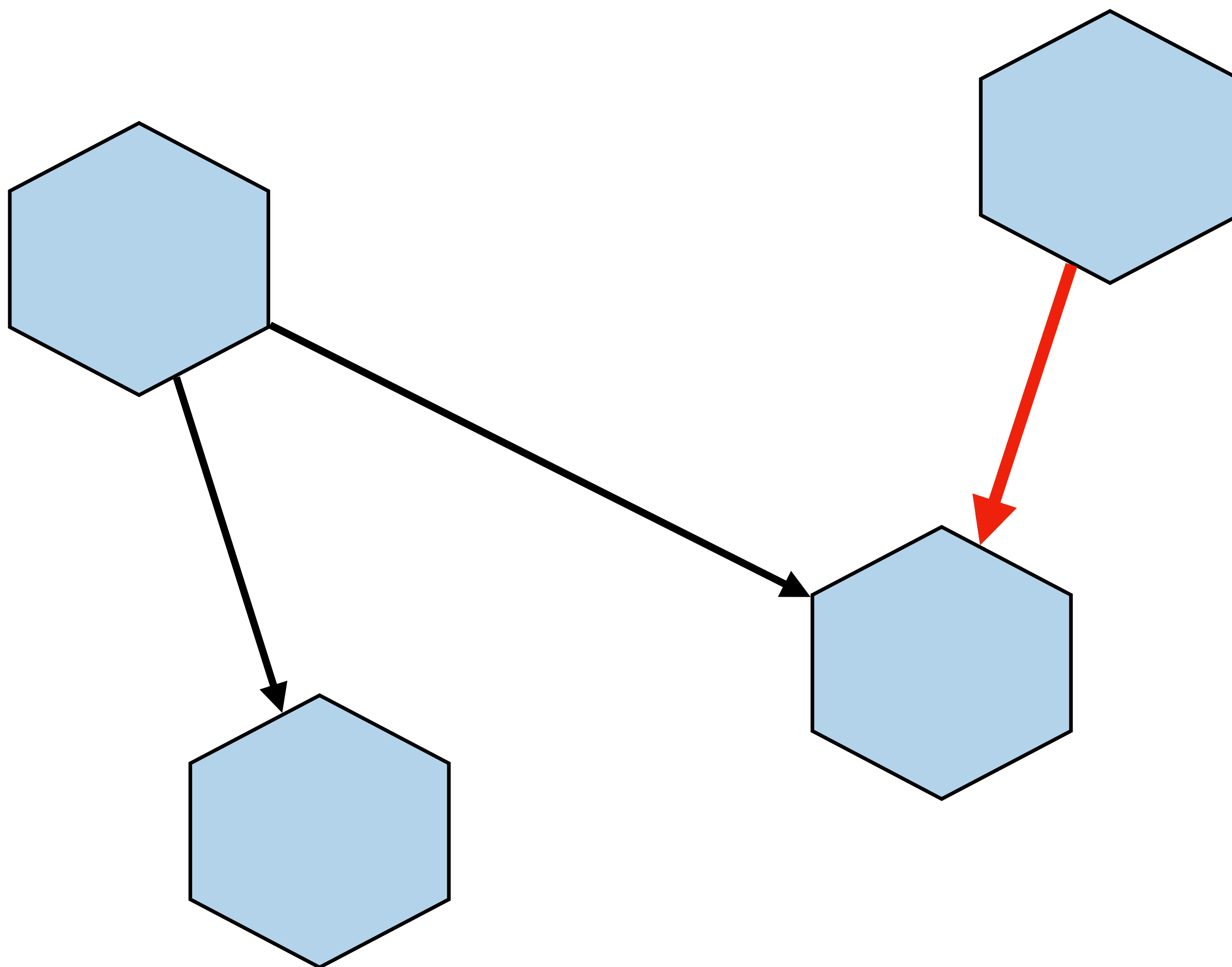


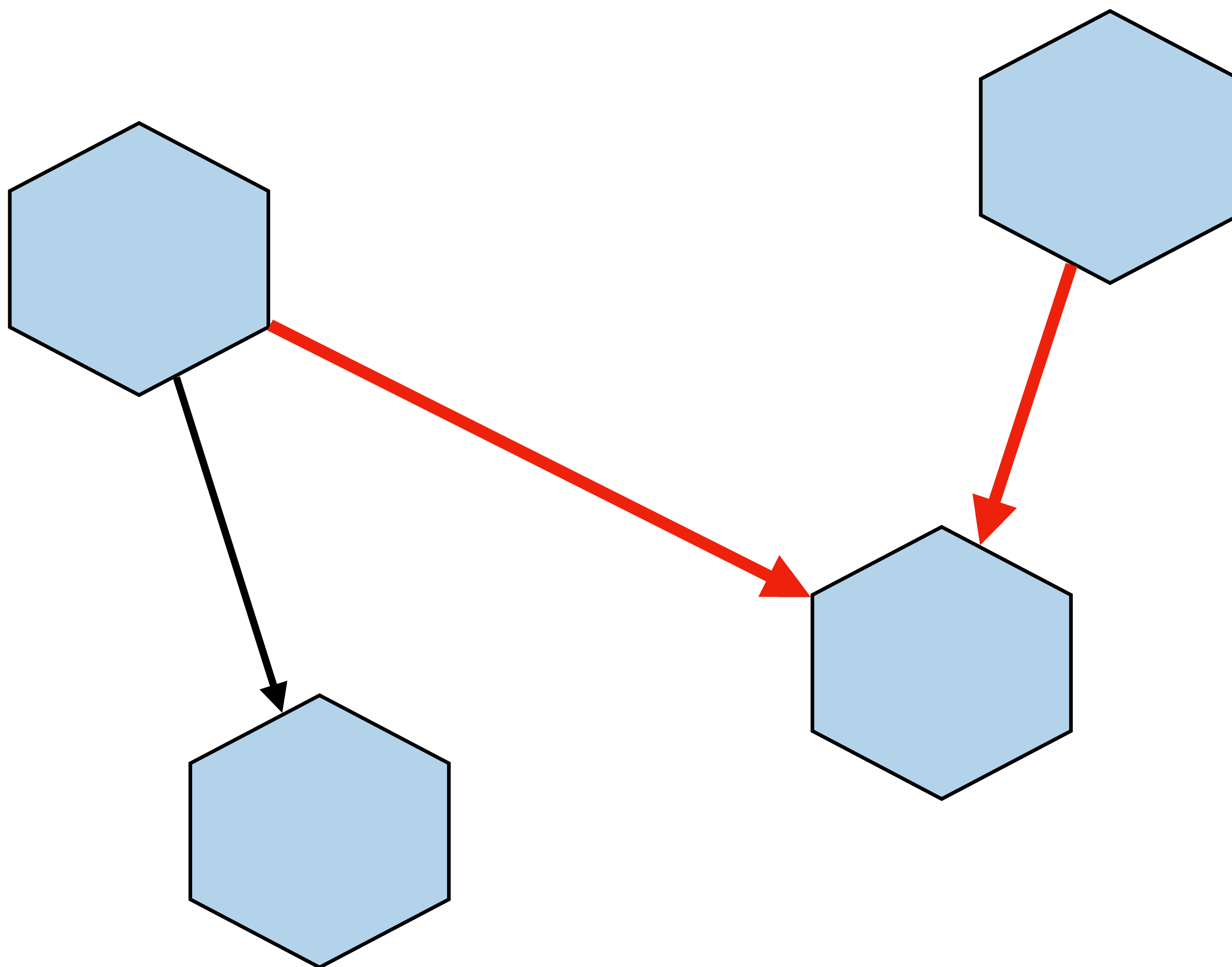


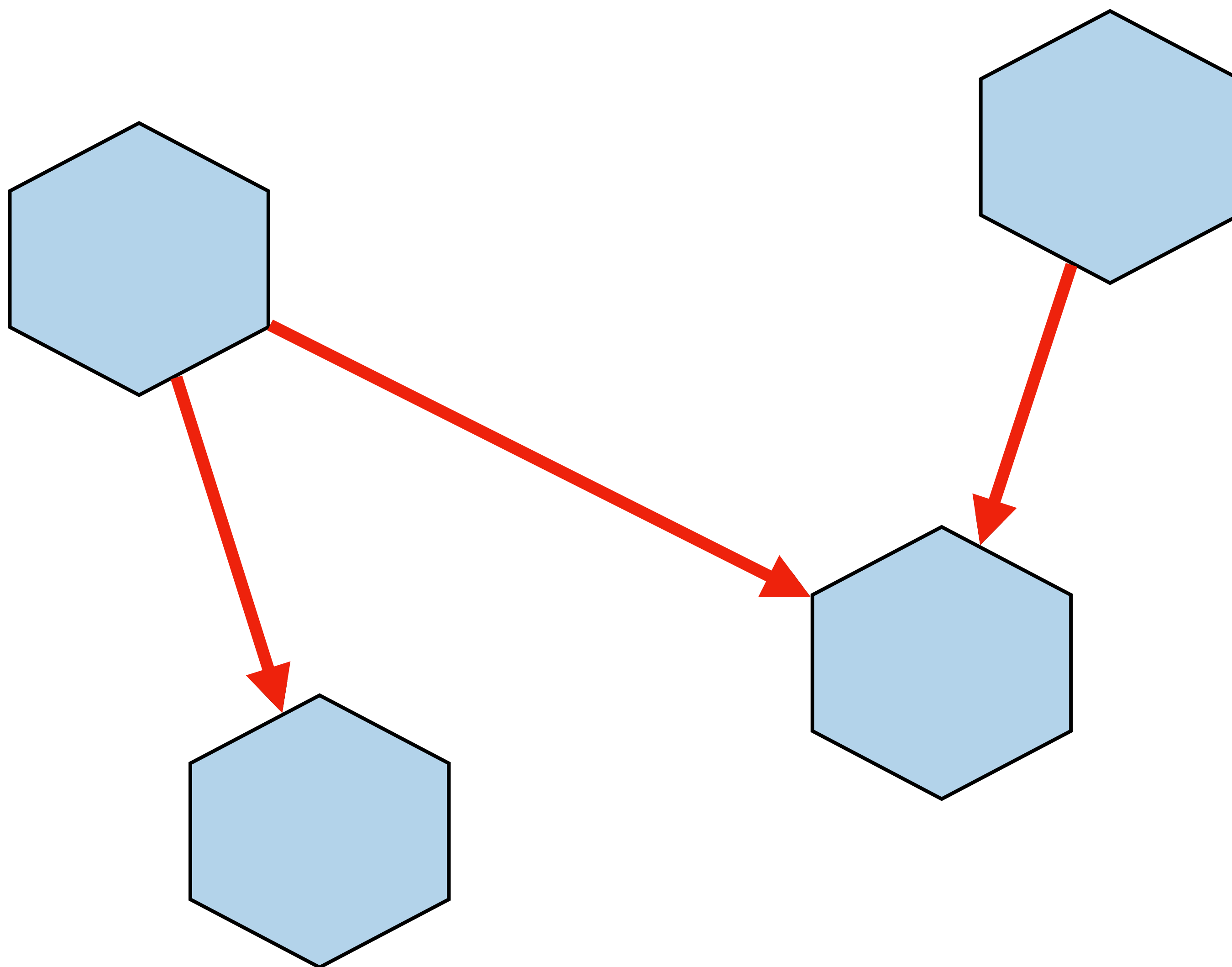
#TeamHexagon

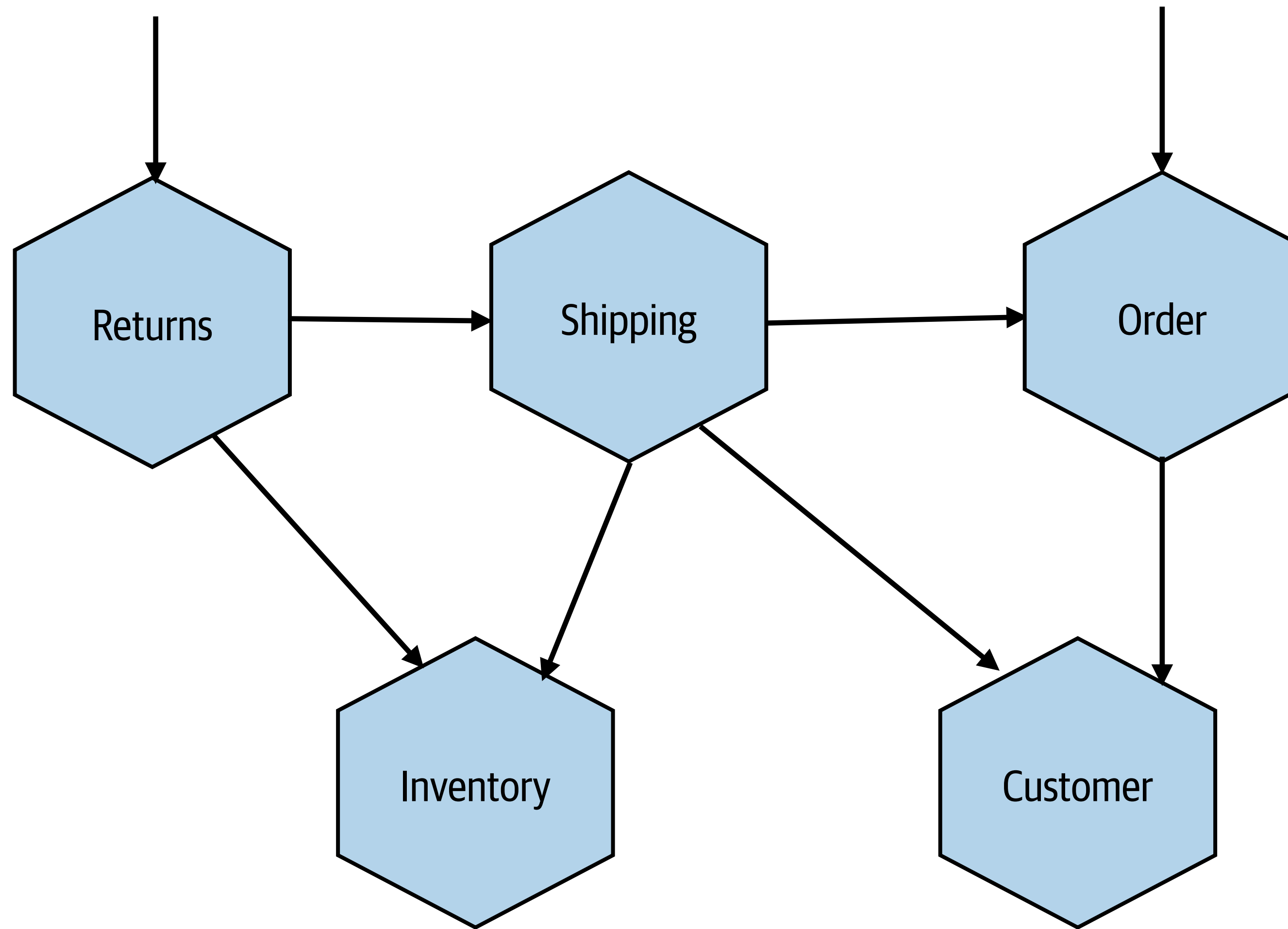


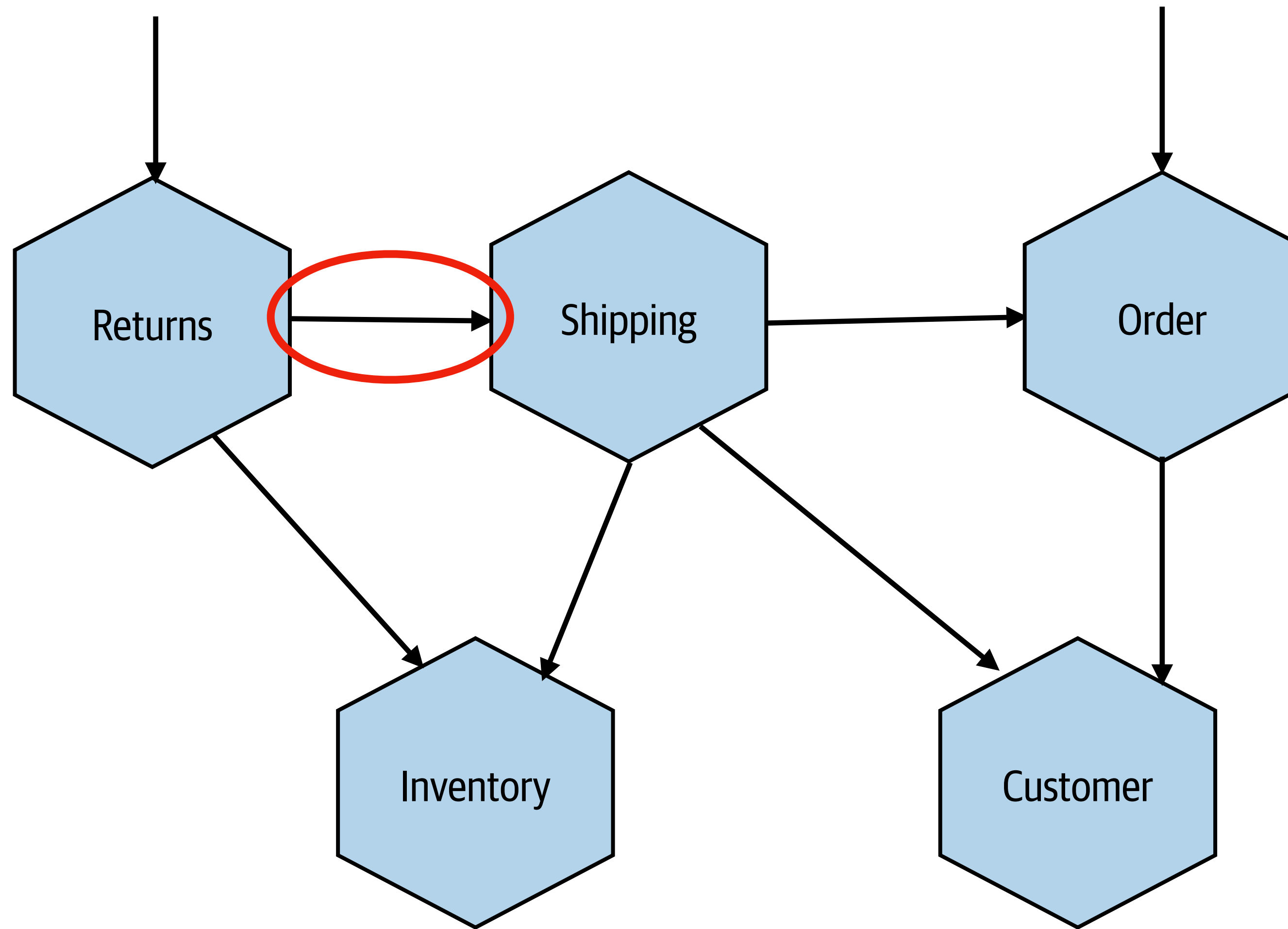


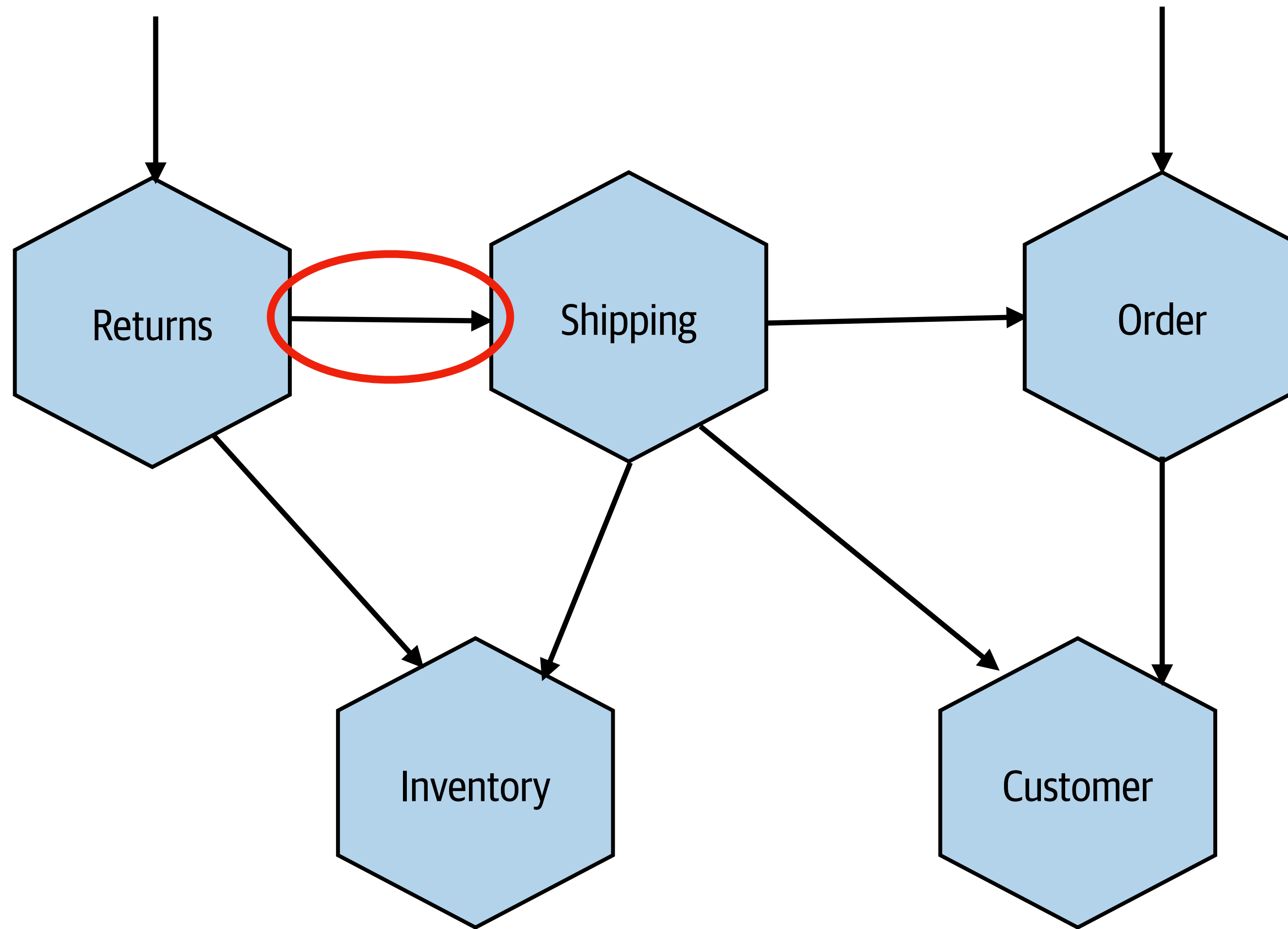




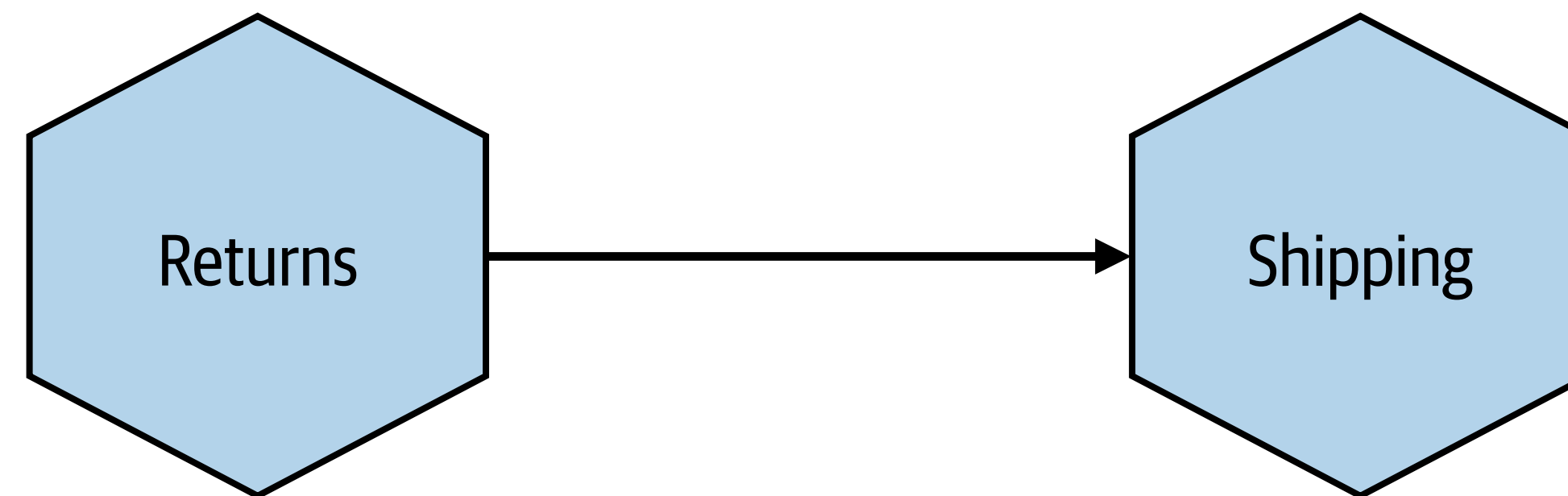


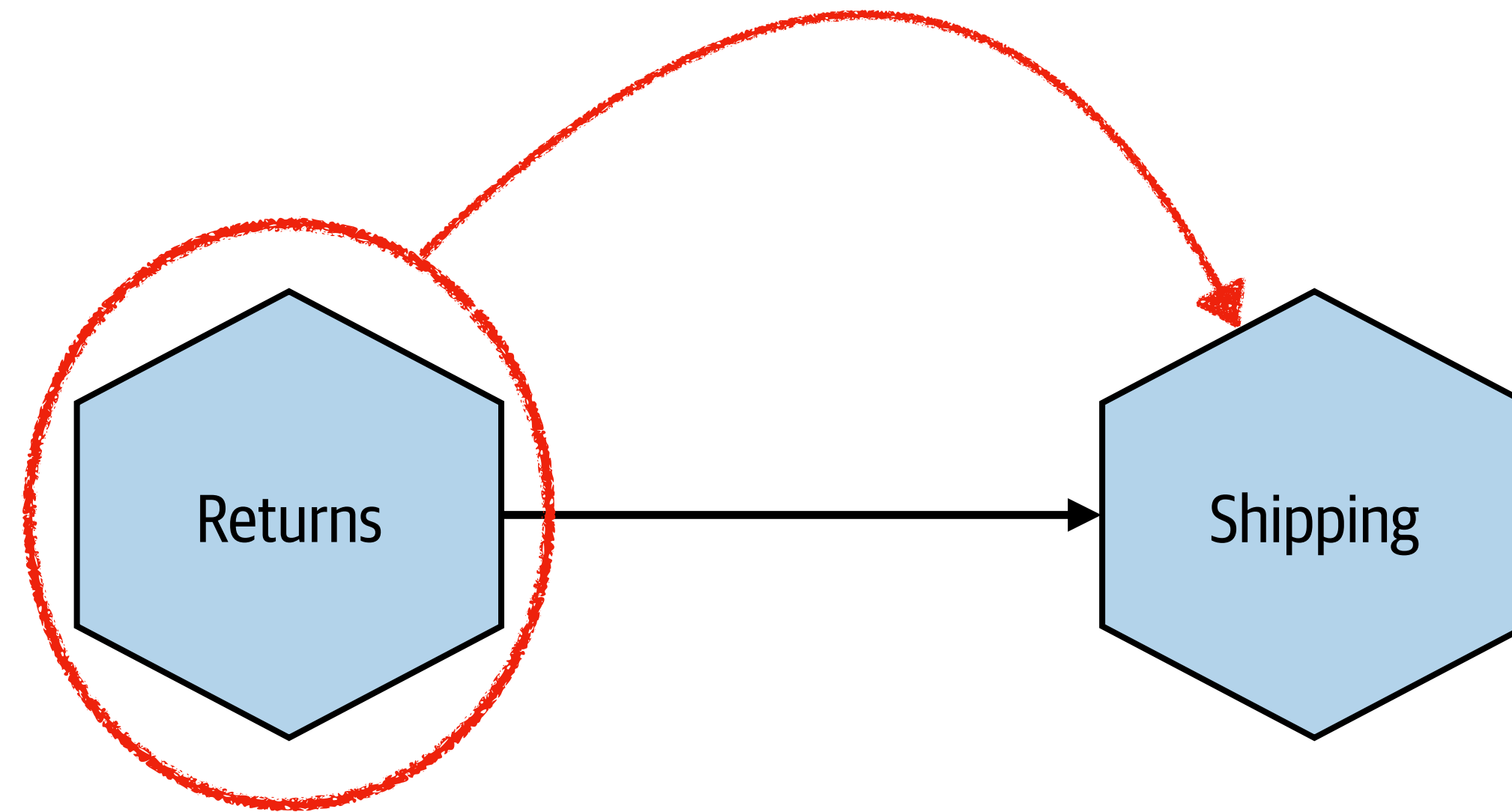


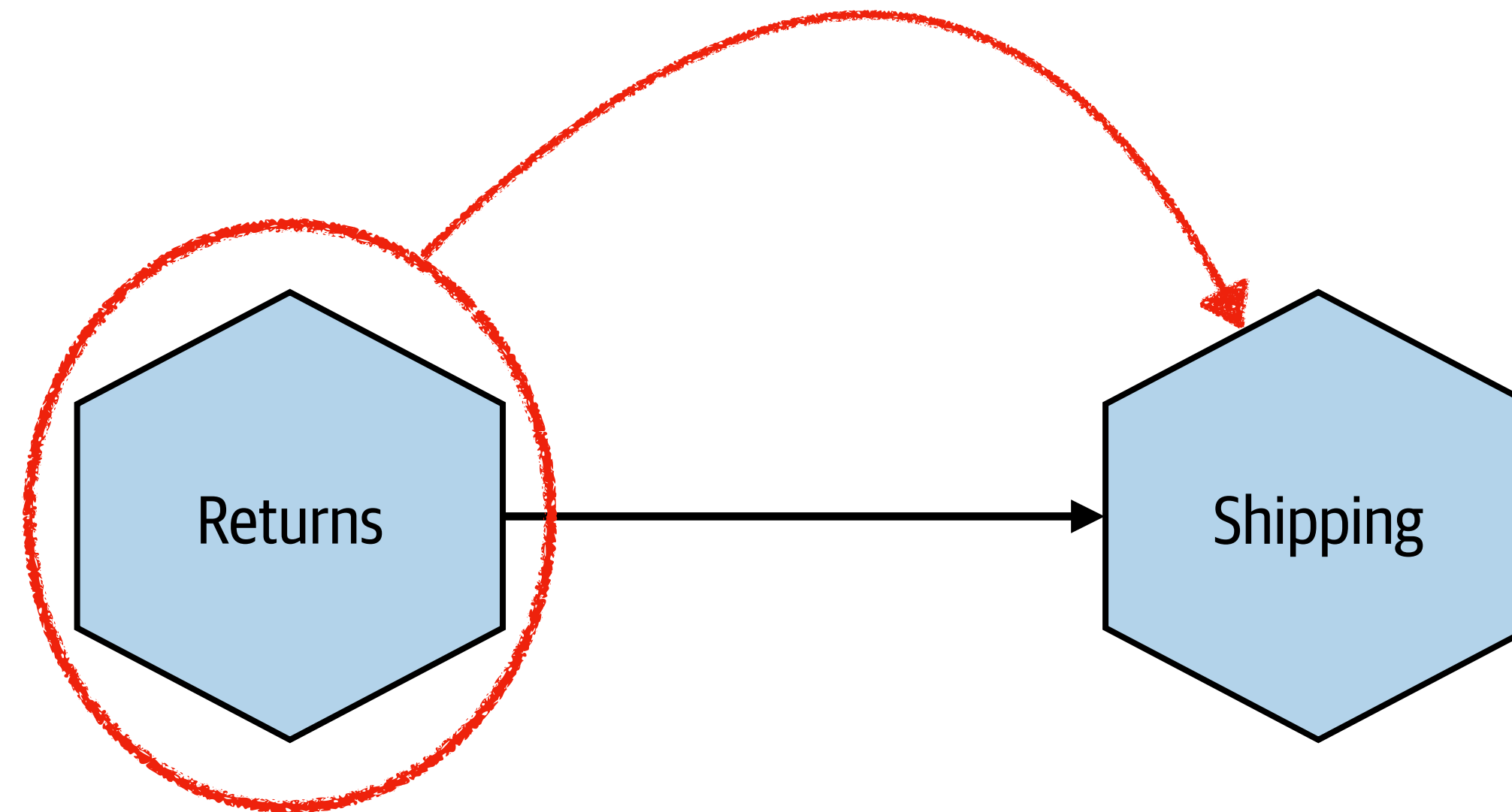




Logical dependency







The Returns microservice depends on some functionality which Shipping provides

Synchronous

Synchronous vs Asynchronous

MAIN STYLES OF COMMUNICATION

MAIN STYLES OF COMMUNICATION

Request/Response

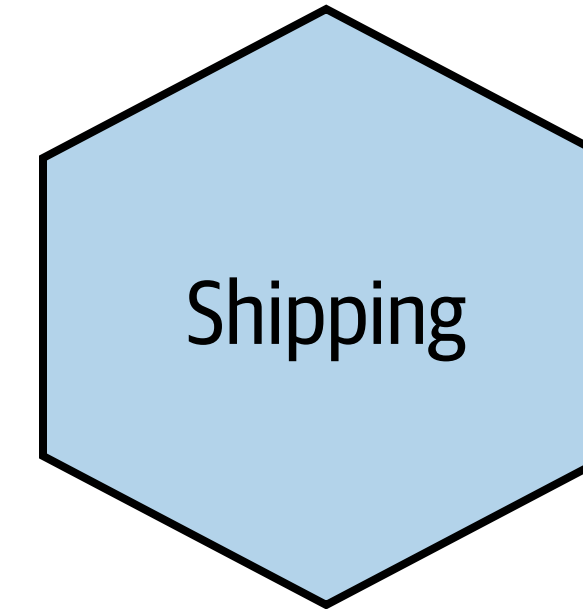
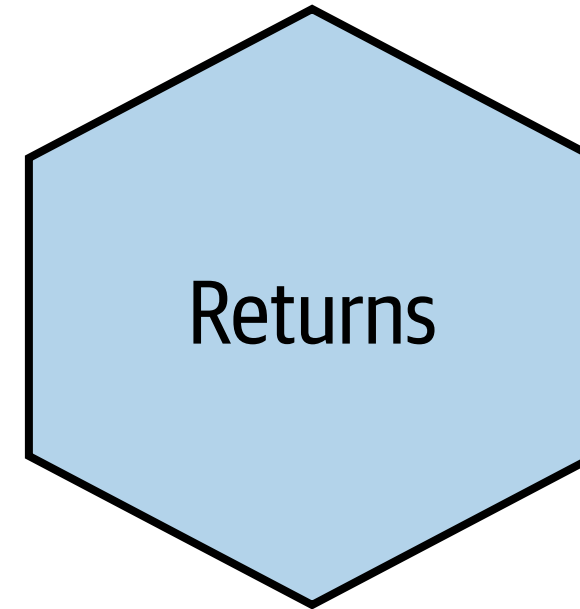
MAIN STYLES OF COMMUNICATION

Request/Response

Event-Driven

REQUEST-RESPONSE

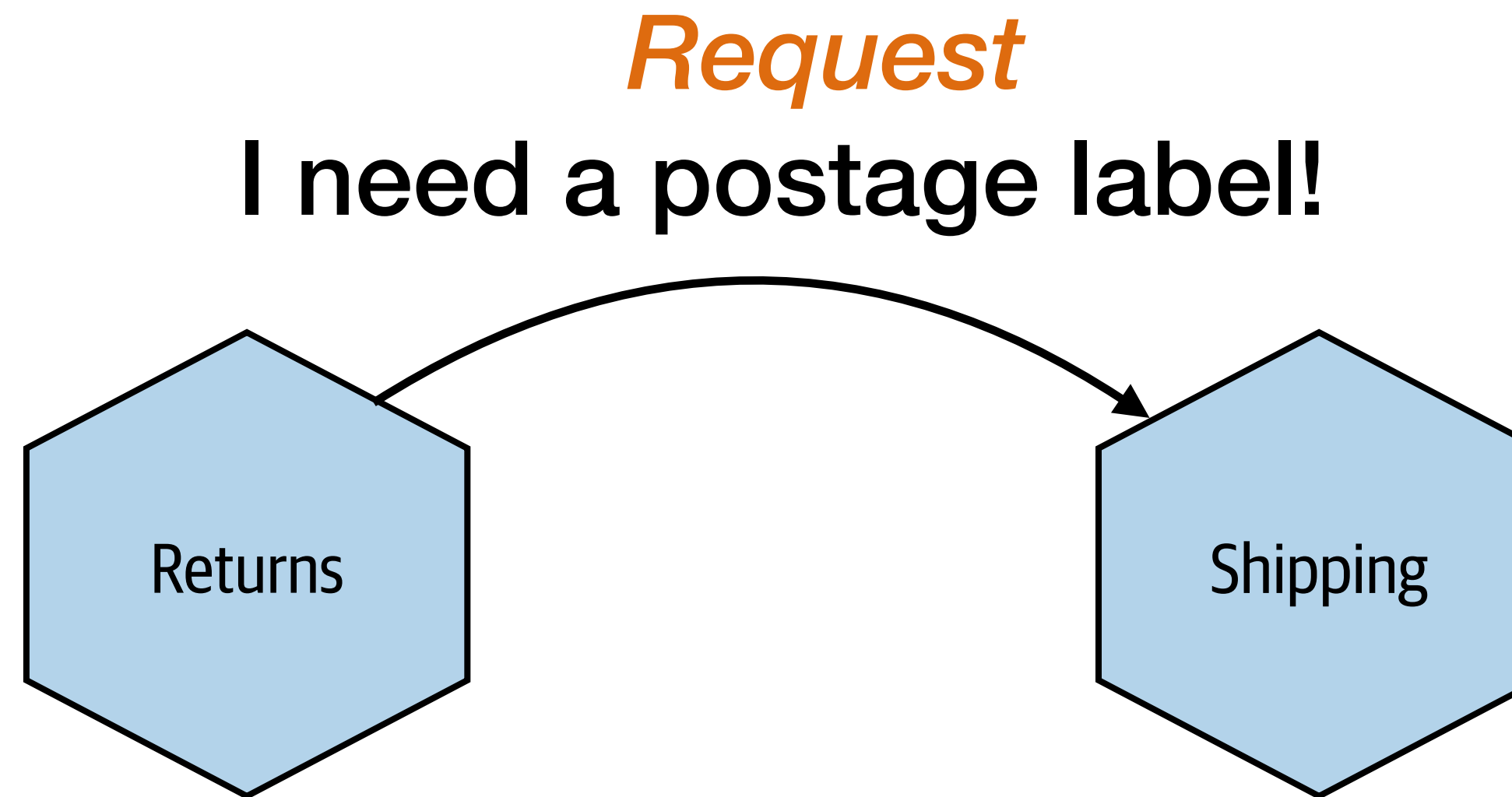
The consumer is *asking*
the microservice for
something



REQUEST-RESPONSE

The consumer is *asking*
the microservice for
something

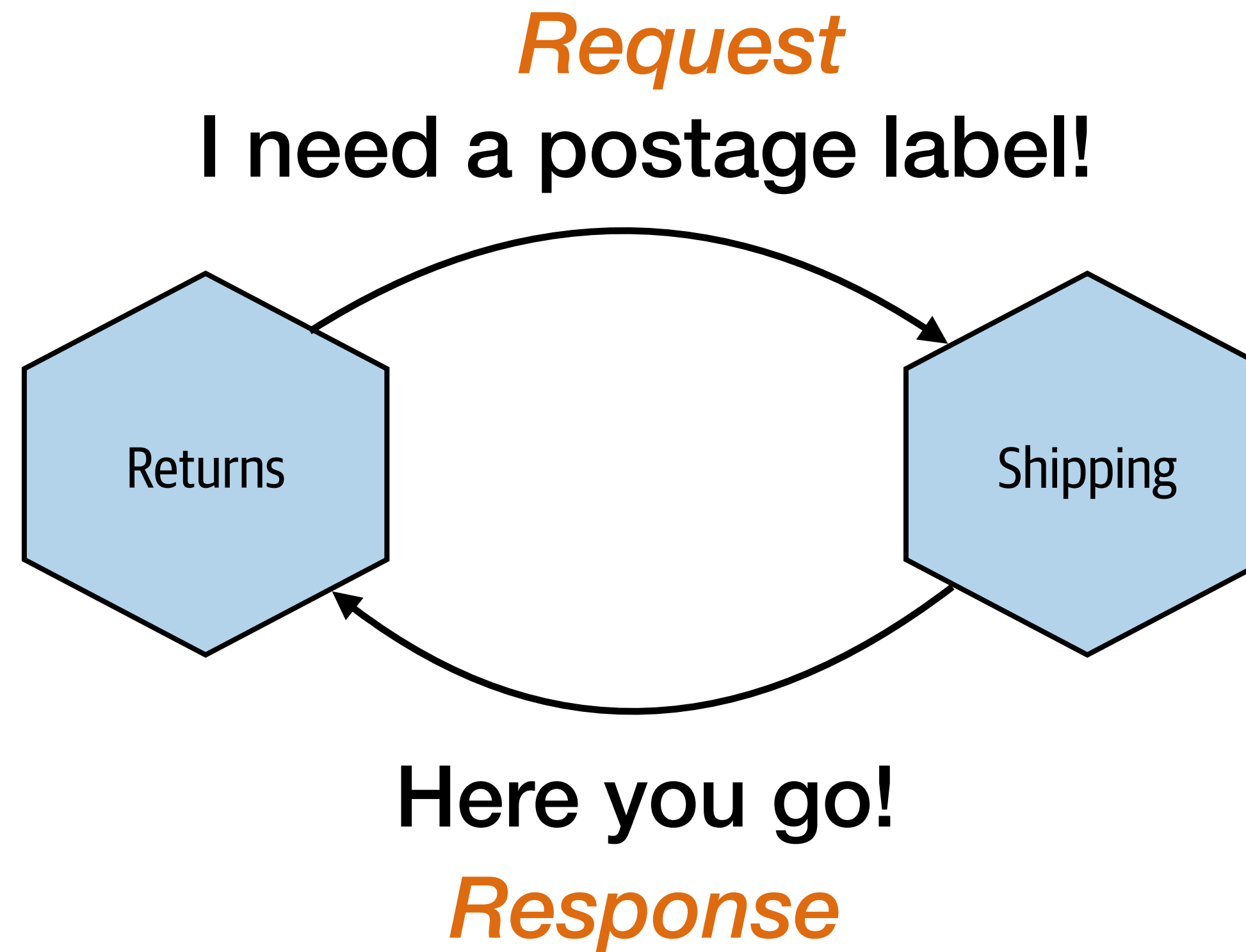
Consumer



REQUEST-RESPONSE

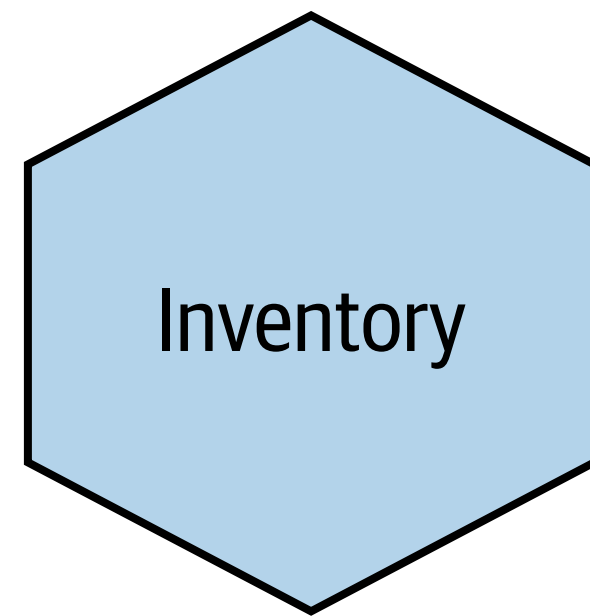
The consumer is *asking*
the microservice for
something

Consumer



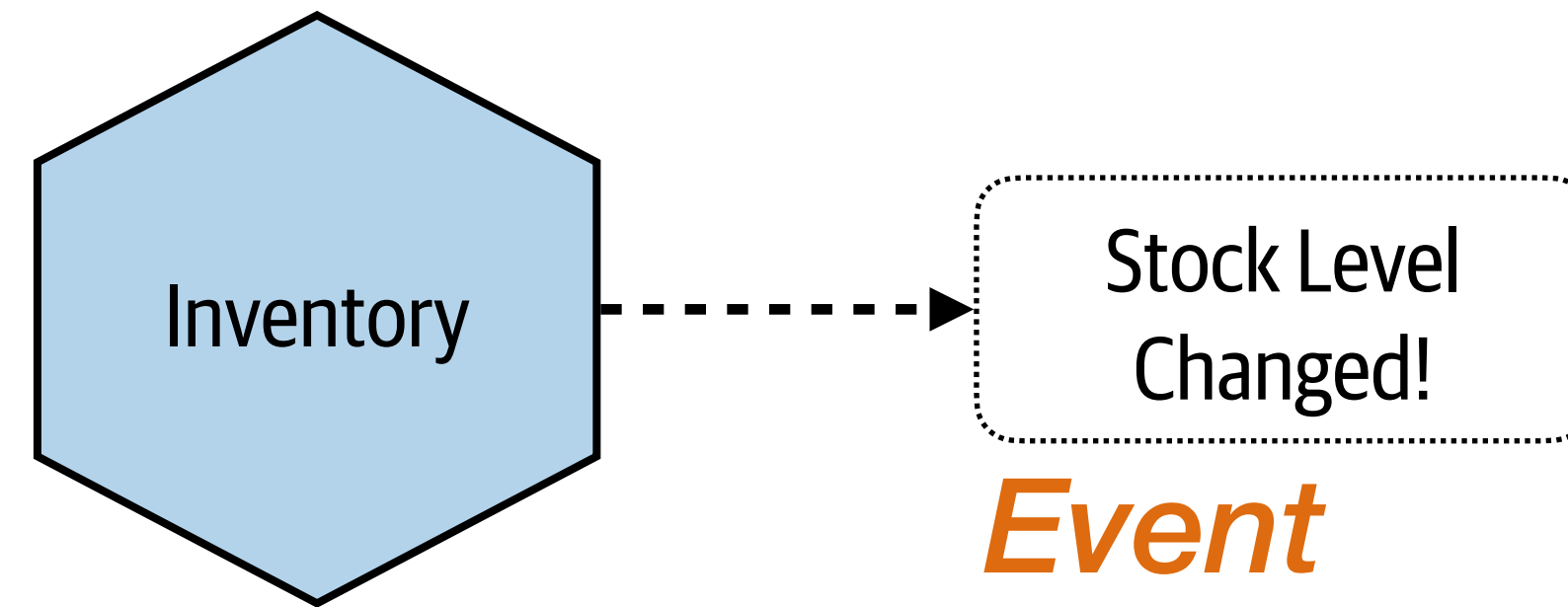
EVENT-DRIVEN

An event (fact) is broadcast



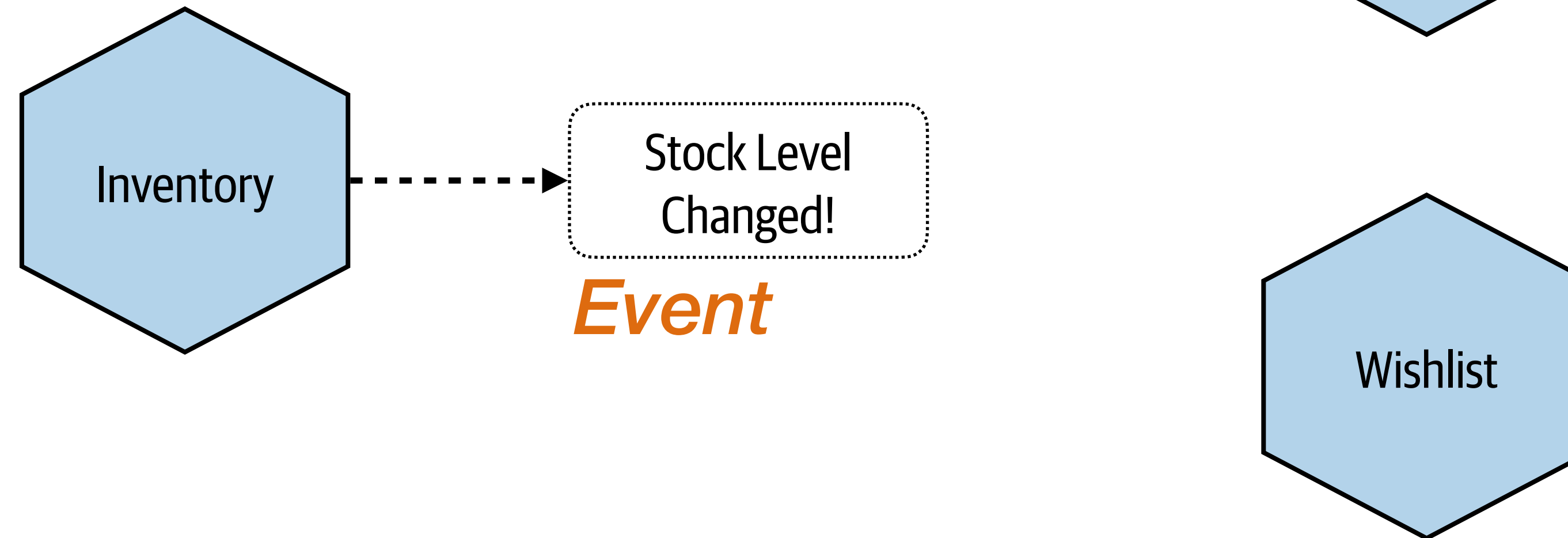
EVENT-DRIVEN

An event (fact) is broadcast



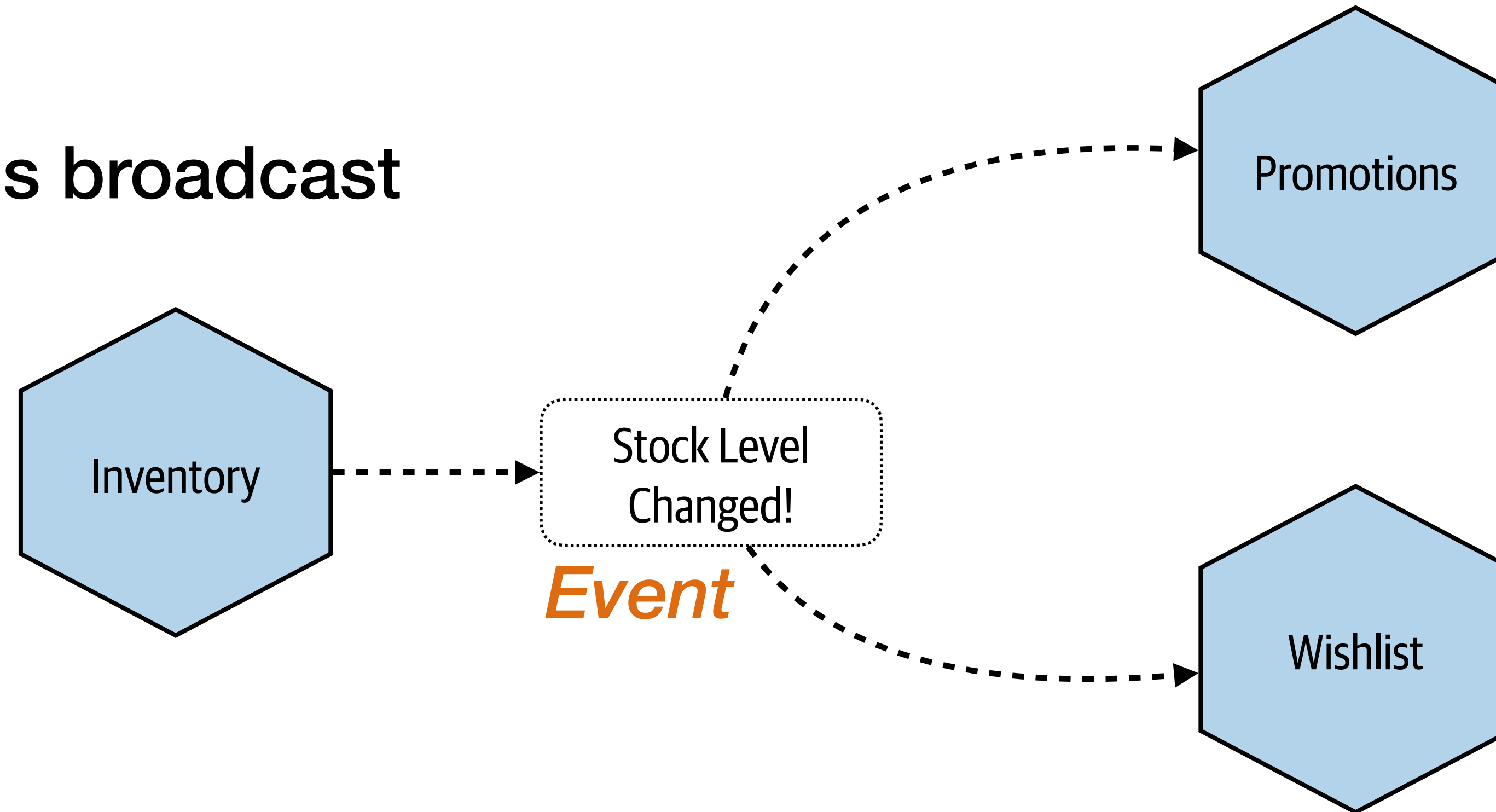
EVENT-DRIVEN

An event (fact) is broadcast



EVENT-DRIVEN

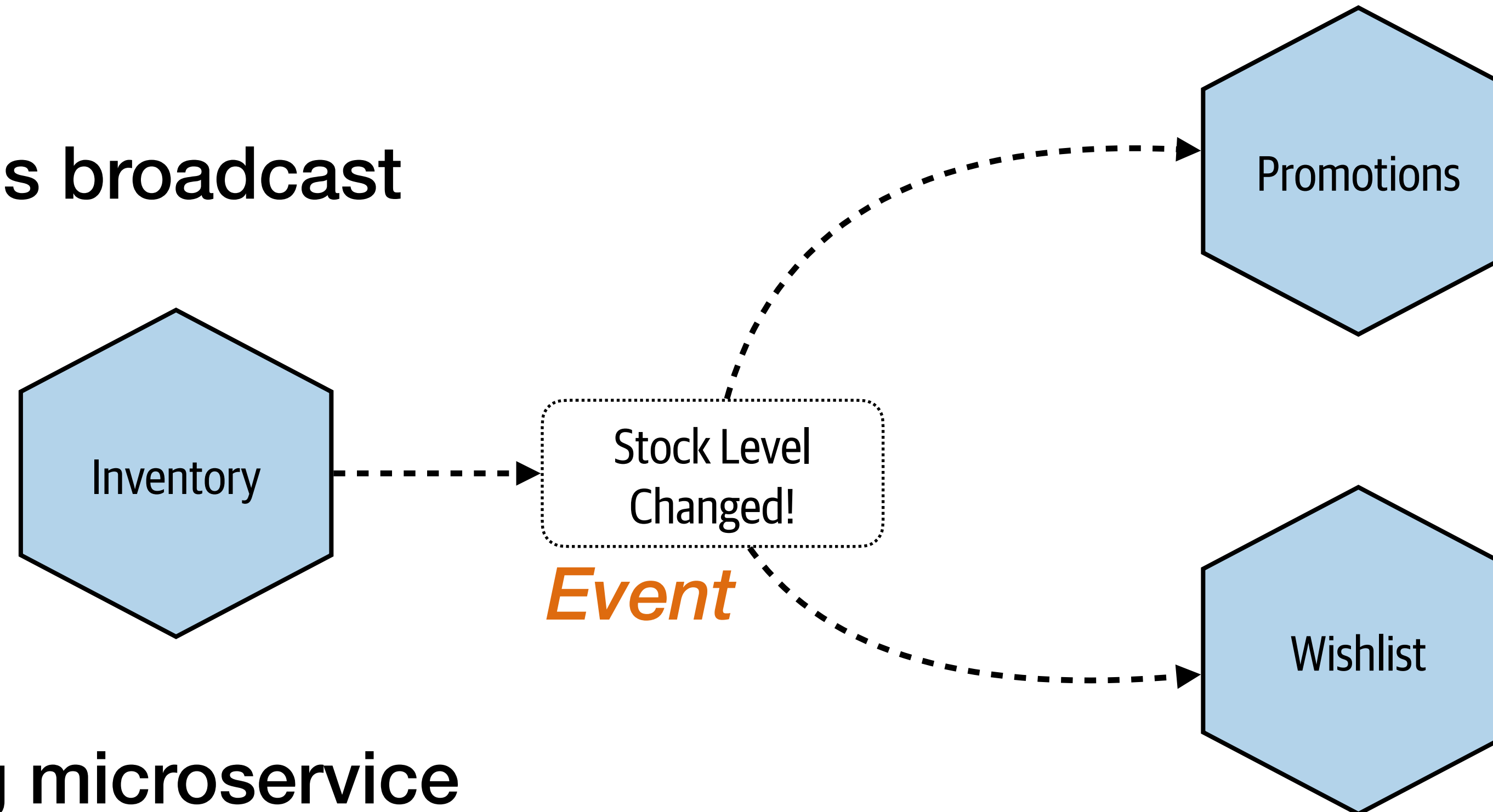
An event (fact) is broadcast



Interested parties receive the event and react accordingly

EVENT-DRIVEN

An event (fact) is broadcast

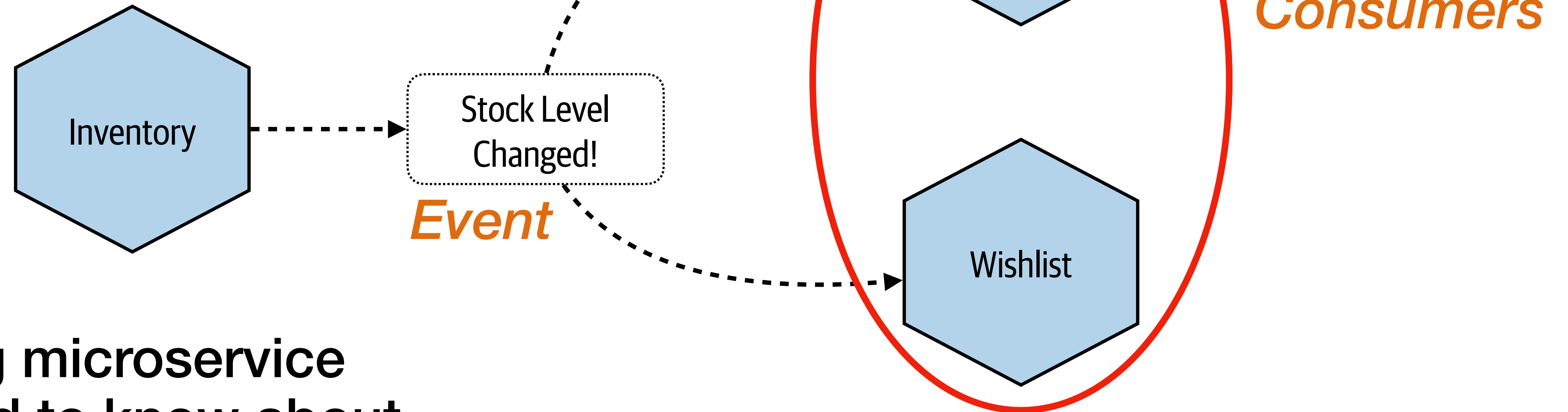


The emitting microservice doesn't need to know about consumers

Interested parties receive the event and react accordingly

EVENT-DRIVEN

An event (fact) is broadcast



The emitting microservice doesn't need to know about consumers

Interested parties receive the event and react accordingly

Synchronous vs Asynchronous

Request/Response

Event-Driven

Request/Response
Synchronous or Asynchronous

Event-Driven

Request/Response

Synchronous or Asynchronous

Event-Driven

Asynchronous

**I don't think there is any consistent
definition of what asynchronous
communication means**

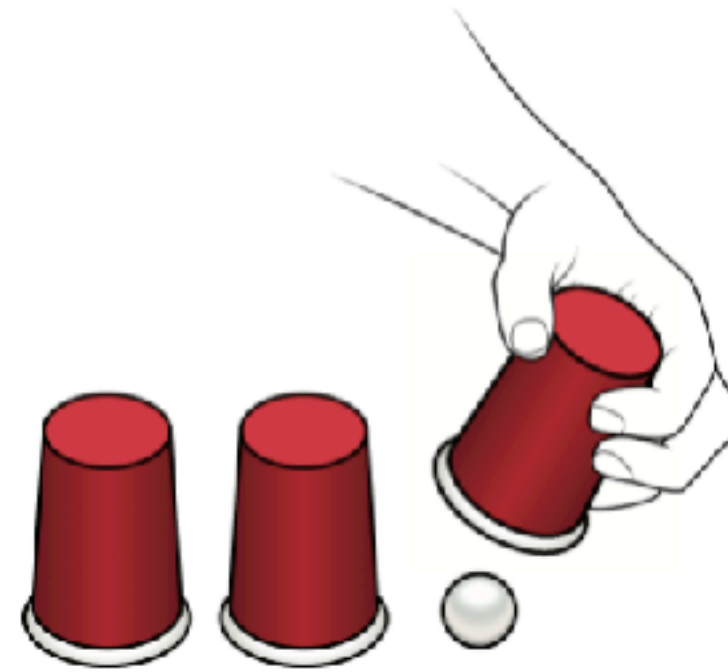
Don't Get Stuck in the CON Game (V3)

Consistency, convergence, and confluence are not the same! Eventual consistency and eventual convergence aren't the same as confluence, either.



Pat Helland

Jul 11 · 3 · 0



This is the 3rd version of “Don’t Get Stuck in the CON Game” that I’ve blogged.

Like many others, I’ve fallen victim to using the phrase *eventual consistency*. It’s a popular phrase, even though its meaning is fuzzy. Different communities within computer science use the word *consistency* in varying ways. Even within these different communities, people are inconsistent in their use of *consistency*. That fuzziness has gotten many of us tangled up in misunderstanding.

It turns out that there are other terms, *convergence* and *confluence*, that have crisper definitions and are more easily understood than *consistency*.

<https://pathelland.substack.com/p/dont-get-stuck-in-the-con-game-v3>



Sam Newman ✓

@samnewman



So I think the distinction for many people between synchronous and asynchronous inter-service communication to be **really** fuzzy. What does async comms mean to you?

11:29 AM · Jul 13, 2021

<https://twitter.com/samnewman/status/1414894650125586434>

What I found might shock you!



Darren Hobbs @darrenhobbs · Jul 13, 2021



Replying to @samnewman

Sending an email vs having a phone call



<https://twitter.com/darrenhobbs/status/1414920239142342662>

**This implies that asynchronous
communication will be slower!**

But this **feels like a useful distinction**

REACTIVE MANIFESTO

The Reactive Manifesto

Published on September 16 2014. (v2.0)

Organisations working in disparate domains are independently discovering patterns for building software that look the same. These systems are more robust, more resilient, more flexible and better positioned to meet modern demands.

These changes are happening because application requirements have changed dramatically in recent years. Only a few years ago a large application had tens of servers, seconds of response time, hours of offline maintenance and gigabytes of data. Today applications are deployed on everything from mobile devices to cloud-based clusters running thousands of multi-core processors. Users expect millisecond response times and 100% uptime. Data is measured in Petabytes. Today's demands are simply not met by yesterday's software architectures.

We believe that a coherent approach to systems architecture is needed, and we believe that all necessary aspects are already recognised individually: we want systems that are Responsive, Resilient, Elastic and Message Driven. We call these Reactive Systems.

Systems built as Reactive Systems are more flexible, loosely-coupled and [scalable](#). This makes them easier to develop and amenable to change. They are significantly more tolerant of failure and when [failure](#) does occur they meet it with elegance rather than disaster. Reactive Systems are highly responsive, giving [users](#) effective interactive feedback.

<https://www.reactivemanifesto.org/>

REACTIVE MANIFESTO?

🔗 Asynchronous

The Oxford Dictionary defines asynchronous as “not existing or occurring at the same time”. In the context of this manifesto we mean that the processing of a request occurs at an arbitrary point in time, sometime after it has been transmitted from client to service. The client cannot directly observe, or synchronize with, the execution that occurs within the service. This is the antonym of synchronous processing which implies that the client only resumes its own execution once the service has processed the request.

<https://www.reactivemanifesto.org/glossary#Asynchronous>

“...the processing of a request occurs at an arbitrary time, sometime after it has been transmitted from client to server”

<https://www.reactivemanifesto.org/glossary#Asynchronous>

Wat?

“...the processing of a request occurs at an arbitrary time, sometime after it has been transmitted from client to server”

<https://www.reactivemanifesto.org/glossary#Asynchronous>

As opposed to the server processing a request before it is sent?



REACTIVE MANIFESTO (CONT)

*“This is the [opposite] of synchronous processing which implies that **the client only resumes its own execution once the service has processed the request.**”*

<https://www.reactivemanifesto.org/glossary#Asynchronous>



Steve Smith @SteveSmith_Tech · Jul 13, 2021



Replying to [@samnewman](#)

TCP connection open for the duration of a communication



1



2



Tip

https://twitter.com/SteveSmith_Tech/status/1414906580542312450



Steve Smith @SteveSmith_Tech · Jul 13, 2021



Replying to @samnewman

TCP connection open for the duration of a communication



1



2



Tip



Sam Newman  @samnewman · Jul 13, 2021



Replying to @SteveSmith_Tech

So if I use QUIC or UDP, what does that mean?



1



1





Tip

https://twitter.com/SteveSmith_Tech/status/1414906580542312450





Steve Smith @SteveSmith_Tech · Jul 13, 2021
Replying to @samnewman
TCP connection open for the duration of a communication

  1   2   Tip






Sam Newman  @samnewman · Jul 13, 2021
Replying to @SteveSmith_Tech
So if I use QUIC or UDP, what does that mean?

  1   1   Tip



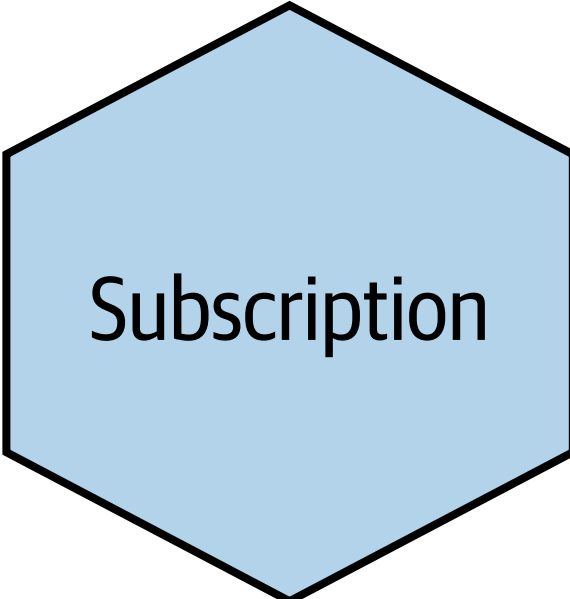
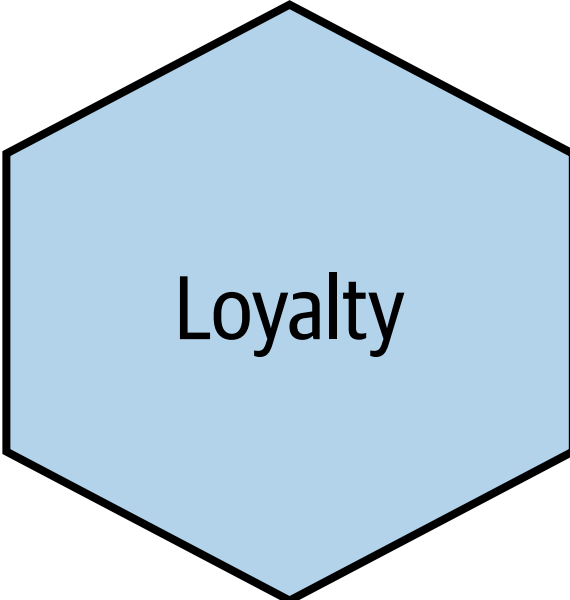
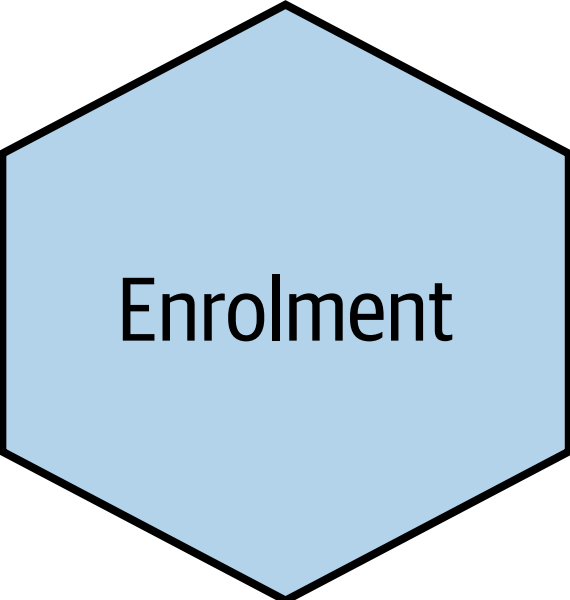
Steve Smith @SteveSmith_Tech · Jul 13, 2021
Replying to @samnewman
I assume you're not blocking for a response

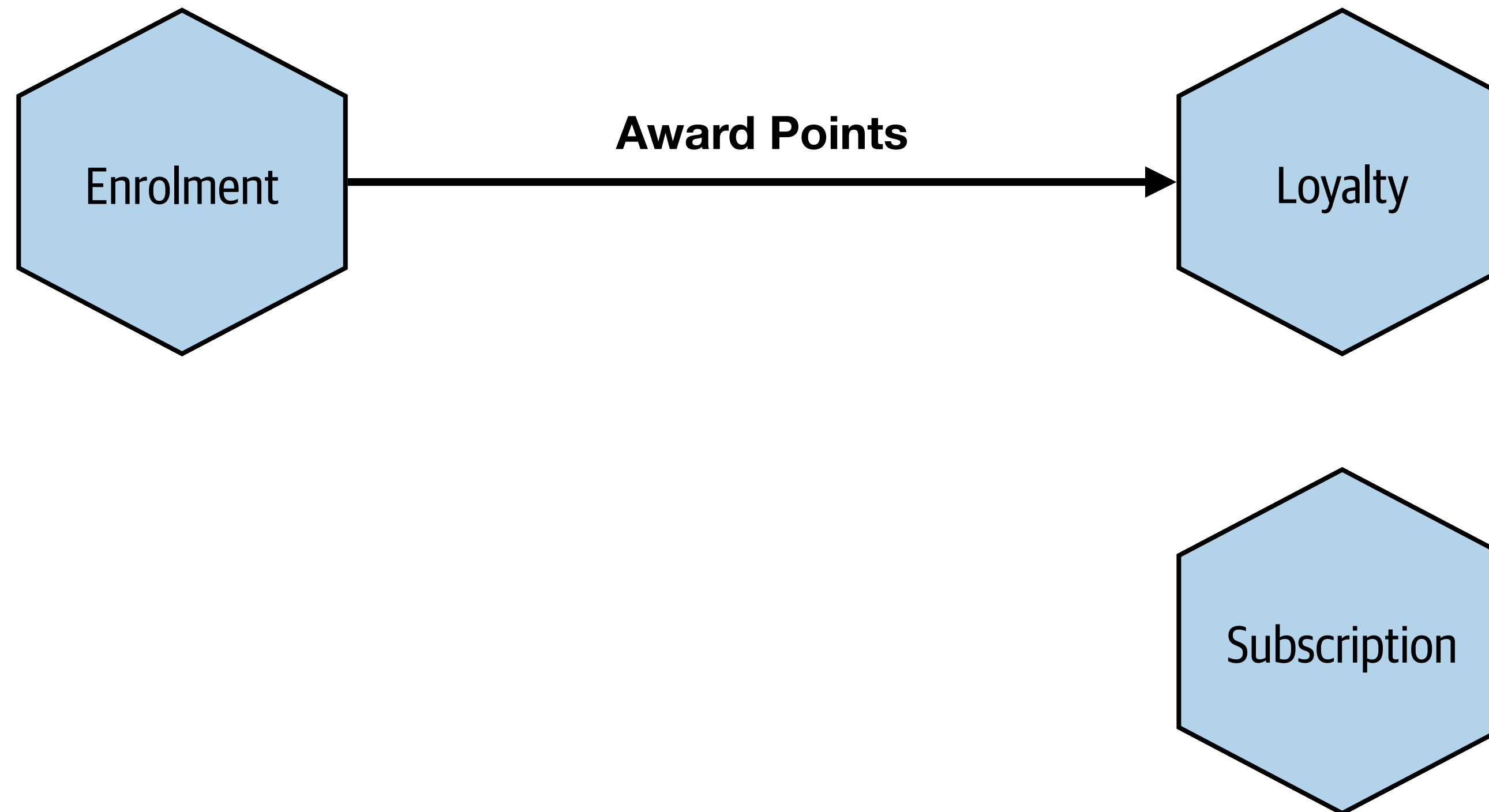
    1   Tip

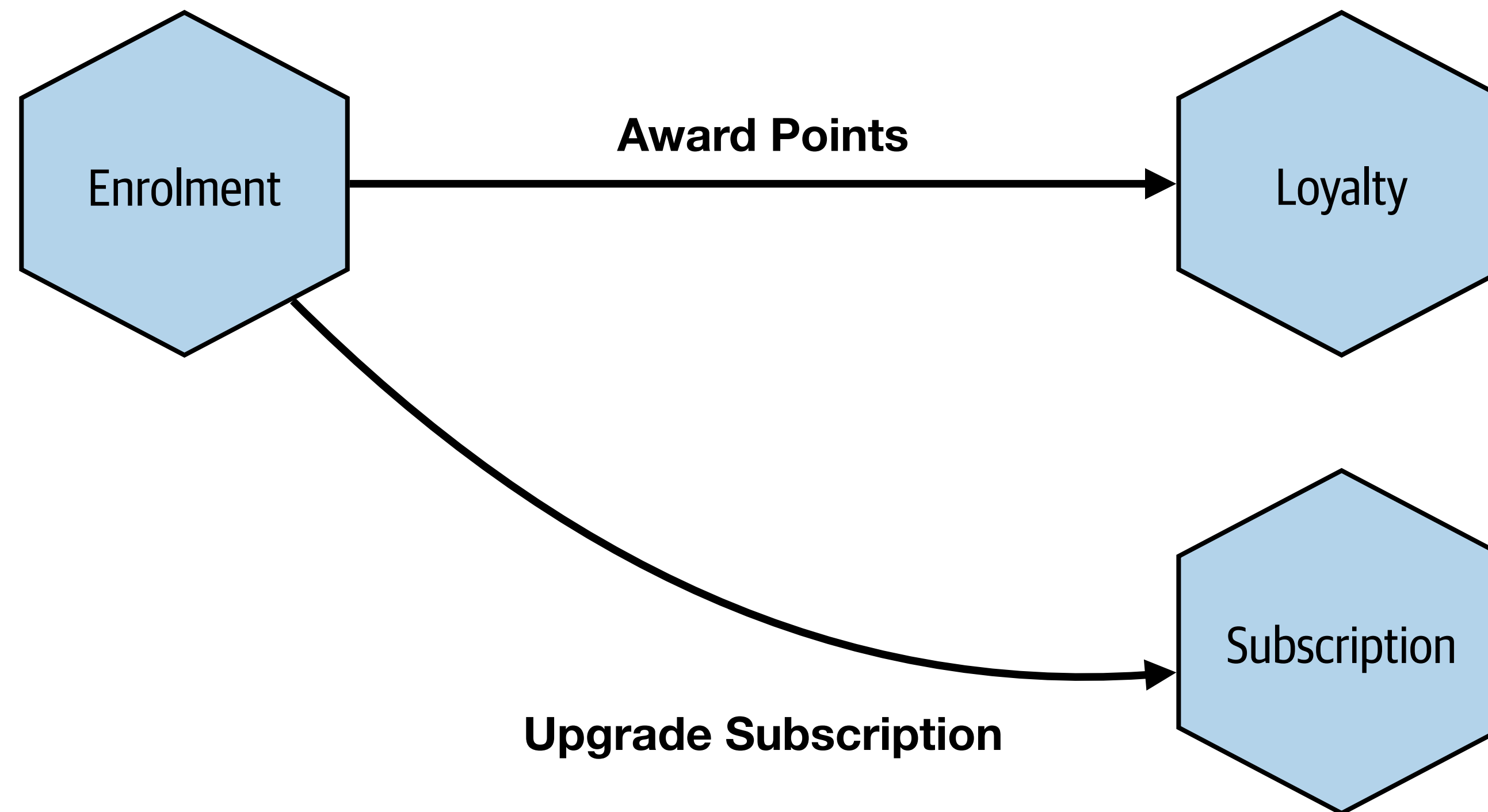
https://twitter.com/SteveSmith_Tech/status/1414906580542312450

**What this really is about is
non-blocking**

**Let's explore the concept of non-
blocking calls**

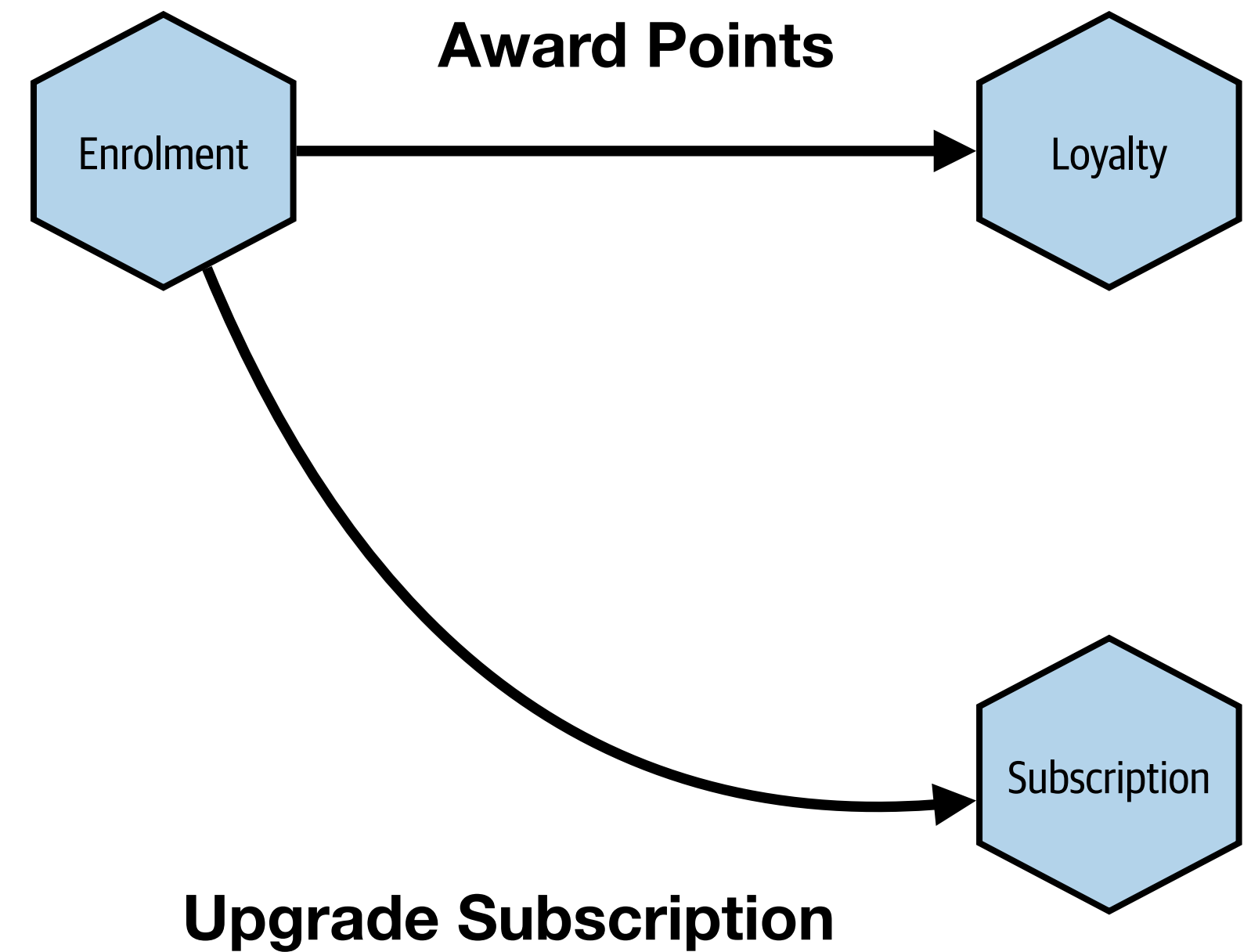




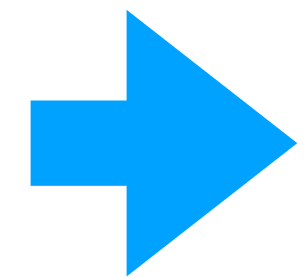


BLOCKING CALLS

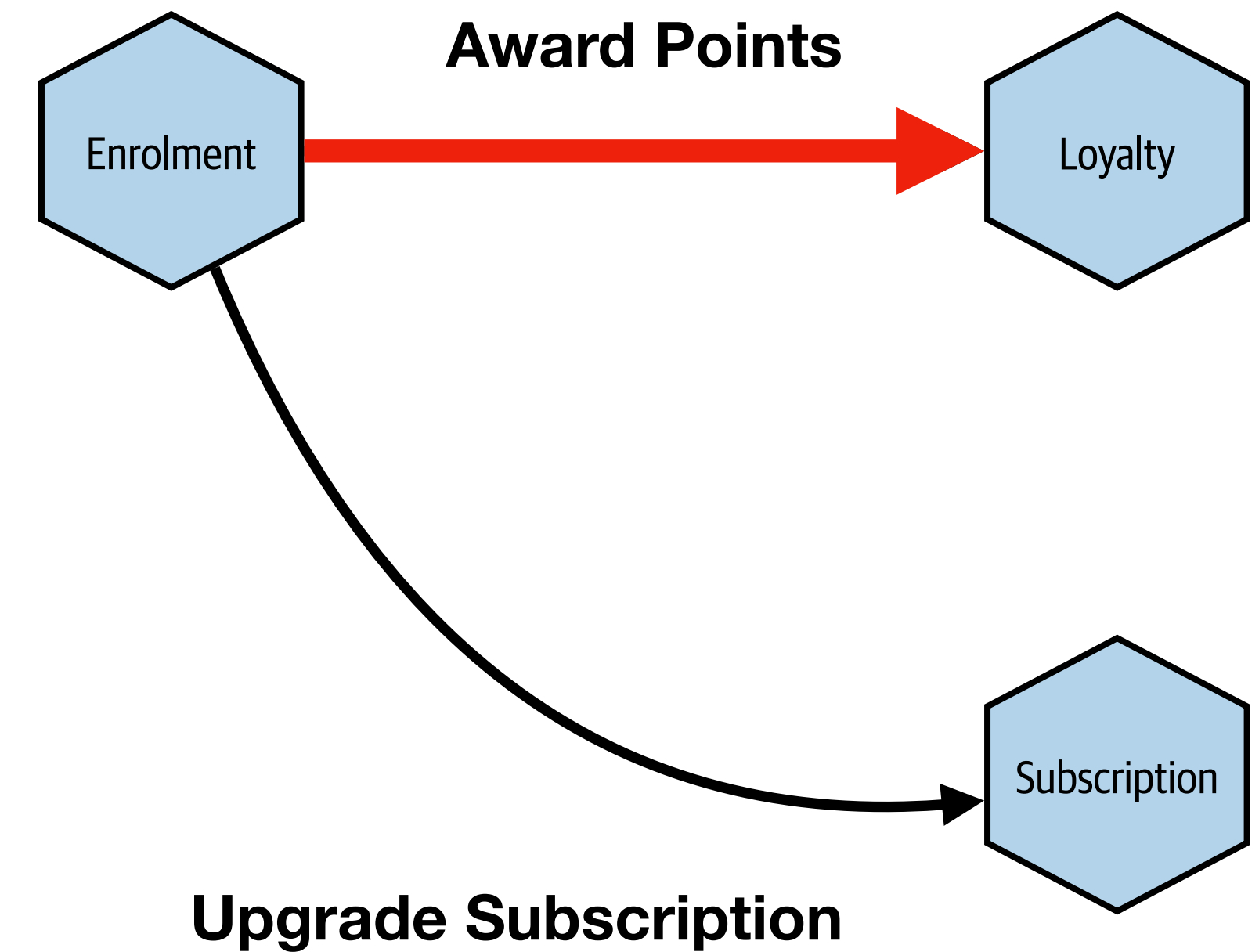
```
res1 = loyaltyService.awardPoints(custID, 200)  
res2 = subsService.upgradeSub(custID, PLATINUM)
```



BLOCKING CALLS

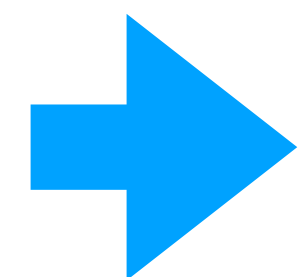


```
res1 = loyaltyService.awardPoints(custID, 200)  
res2 = subsService.upgradeSub(custID, PLATINUM)
```



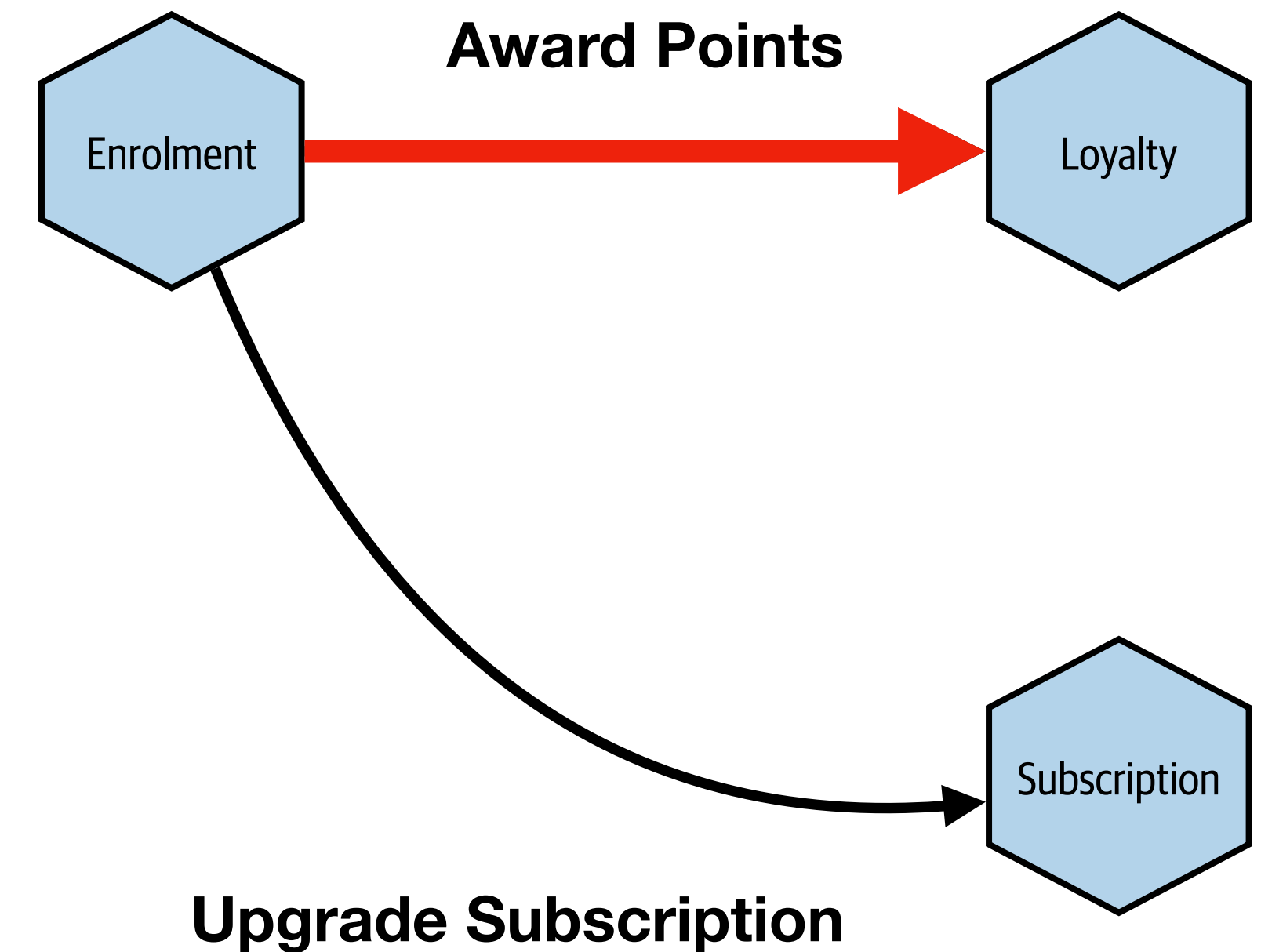
BLOCKING CALLS

We then wait until we either get a response, or we give up



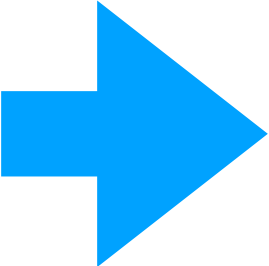
```
res1 = loyaltyService.awardPoints(custID, 200)  
res2 = subsService.upgradeSub(custID, PLATINUM)
```

The first line of code is circled in red.

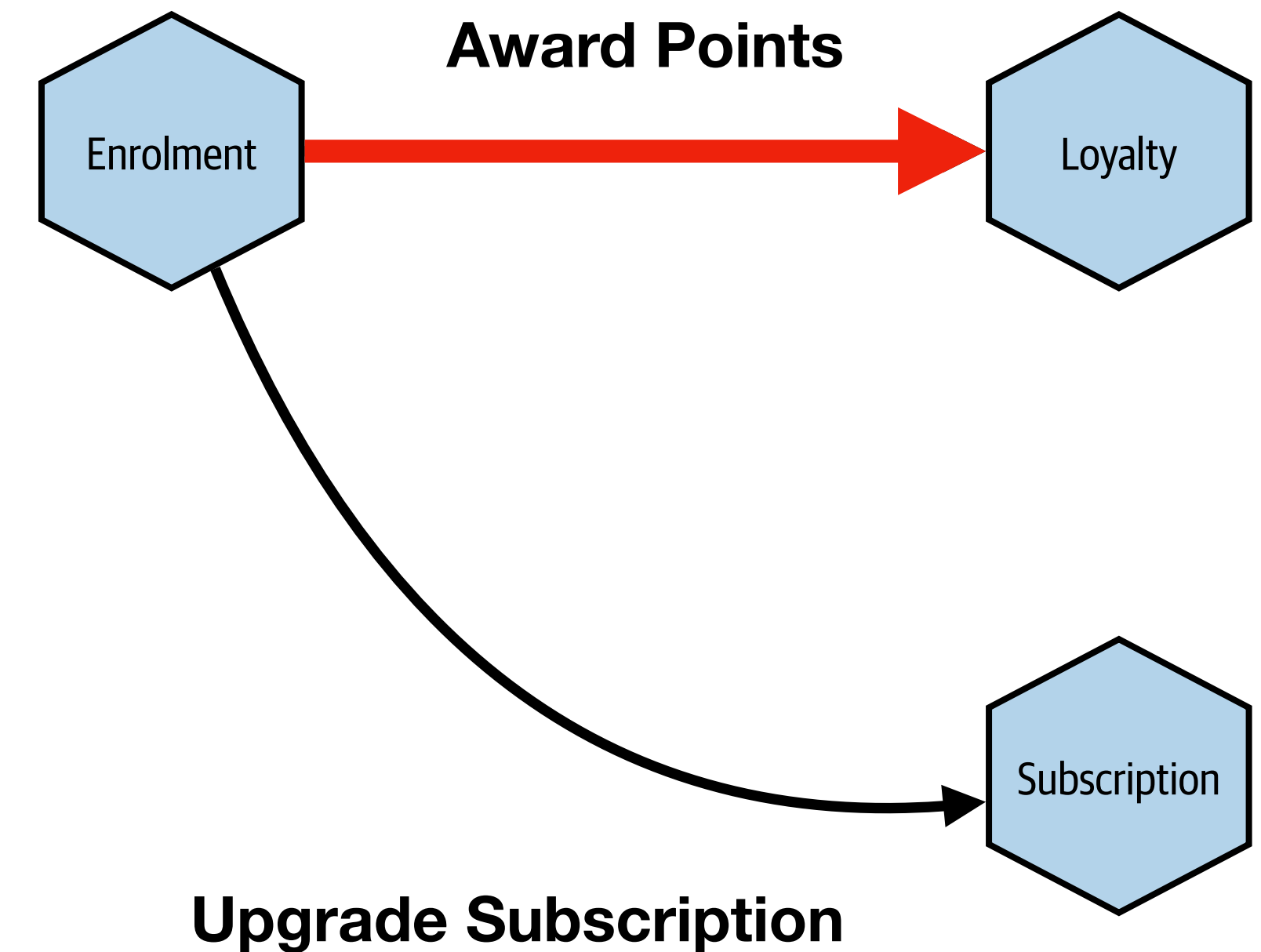


BLOCKING CALLS

We then wait until we either get a response, or we give up

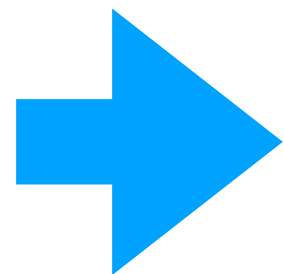


```
res1 = loyaltyService.awardPoints(custID, 200)
res2 = subsService.upgradeSub(custID, PLATINUM)
```



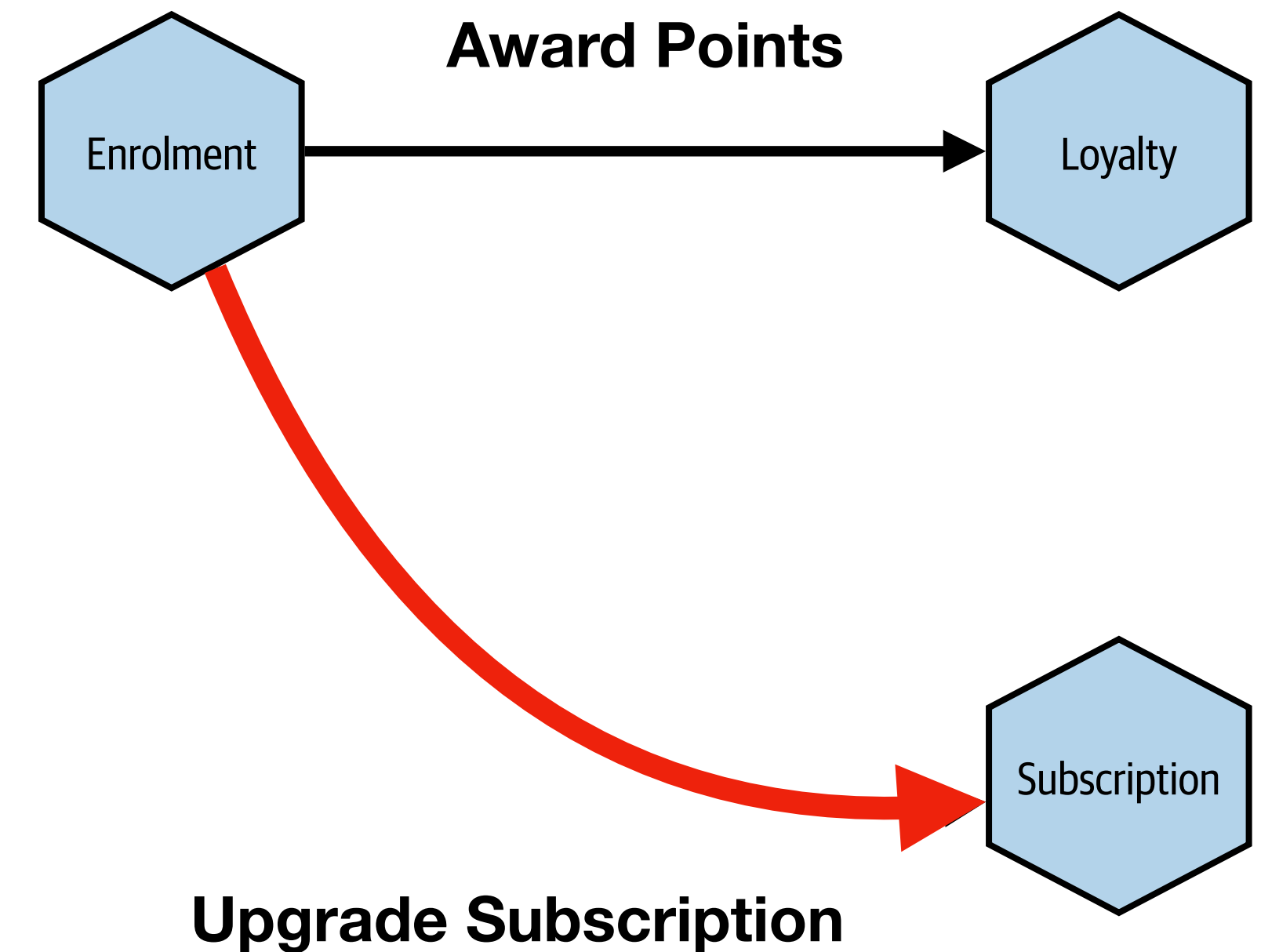
BLOCKING CALLS

We then wait until we either get a response, or we give up

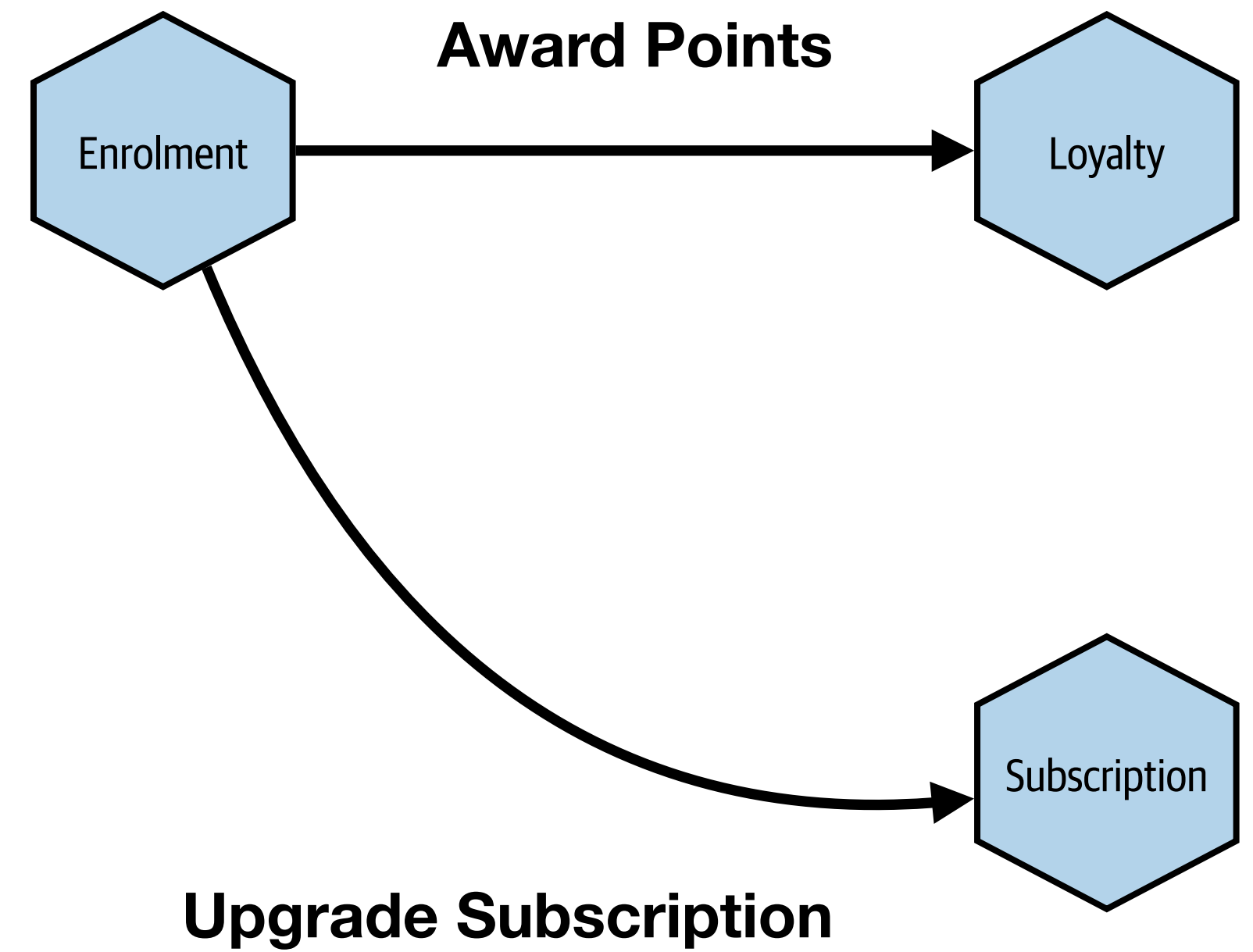


```
res1 = loyaltyService.awardPoints(custID, 200)
res2 = subsService.upgradeSub(custID, PLATINUM)
```

Once the response to loyalty has completed,
then we can call Subscription

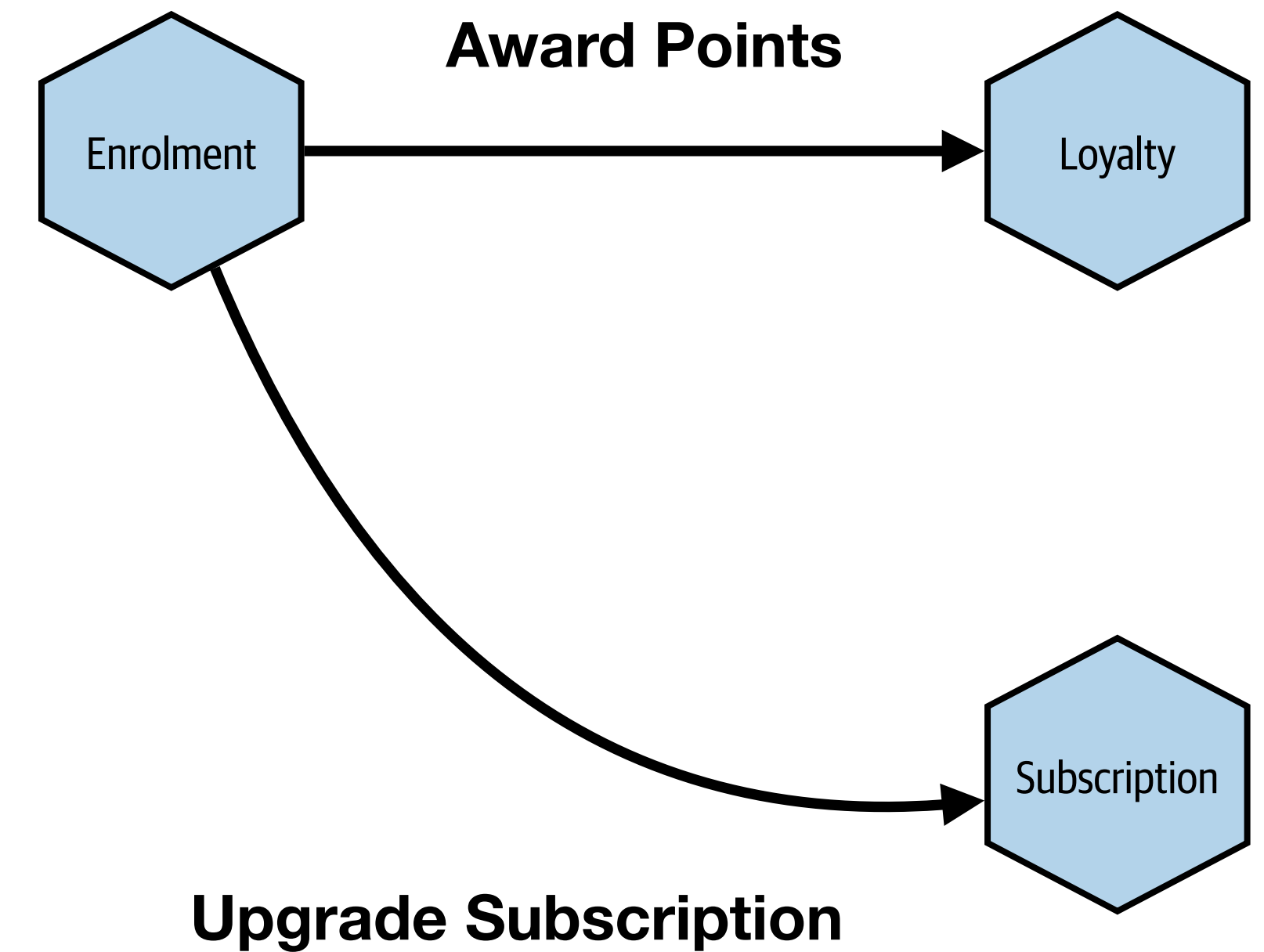


PROBLEMS WITH BLOCKING CALLS?



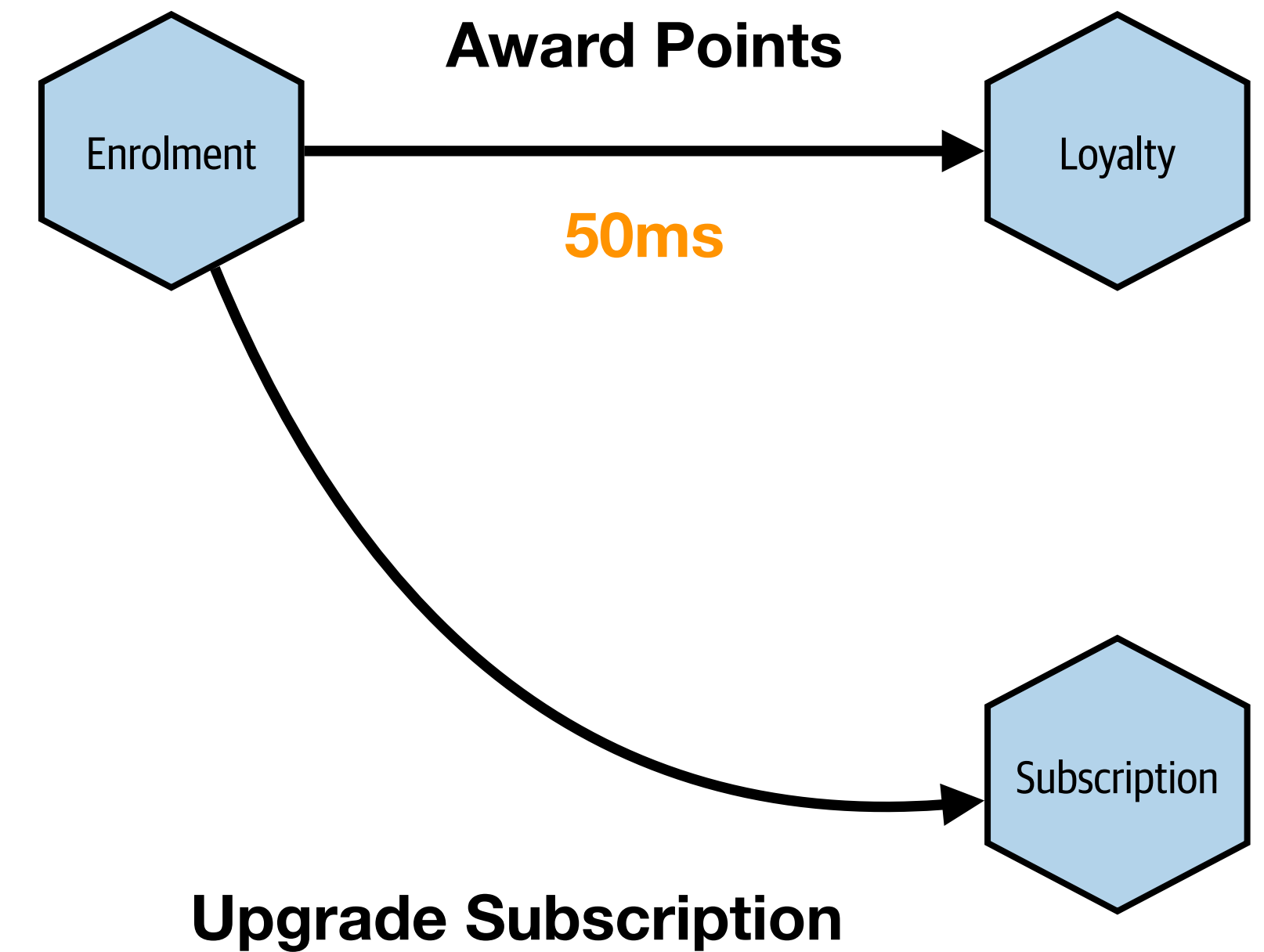
PROBLEMS WITH BLOCKING CALLS?

Latency is the sum of the calls



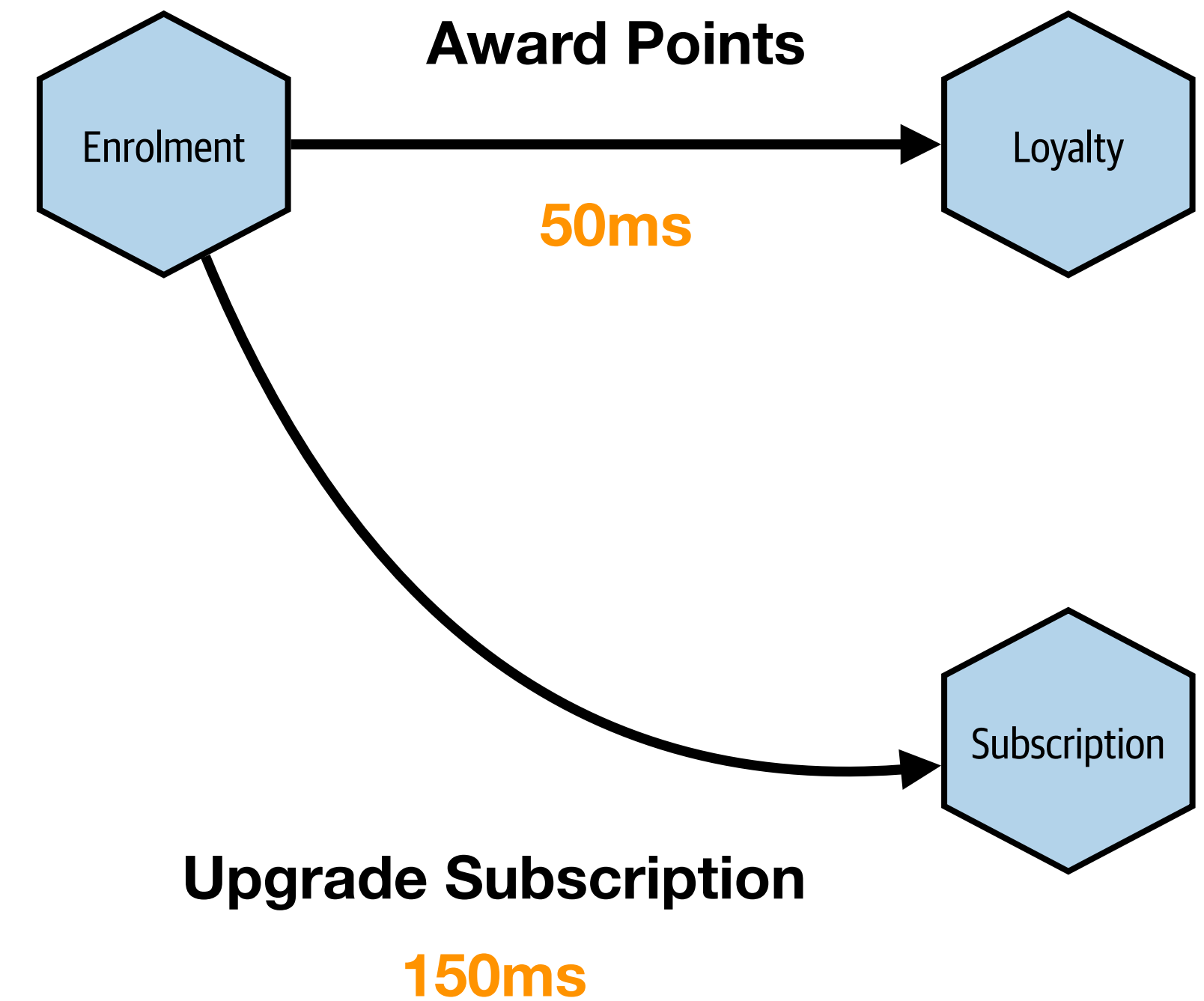
PROBLEMS WITH BLOCKING CALLS?

Latency is the sum of the calls



PROBLEMS WITH BLOCKING CALLS?

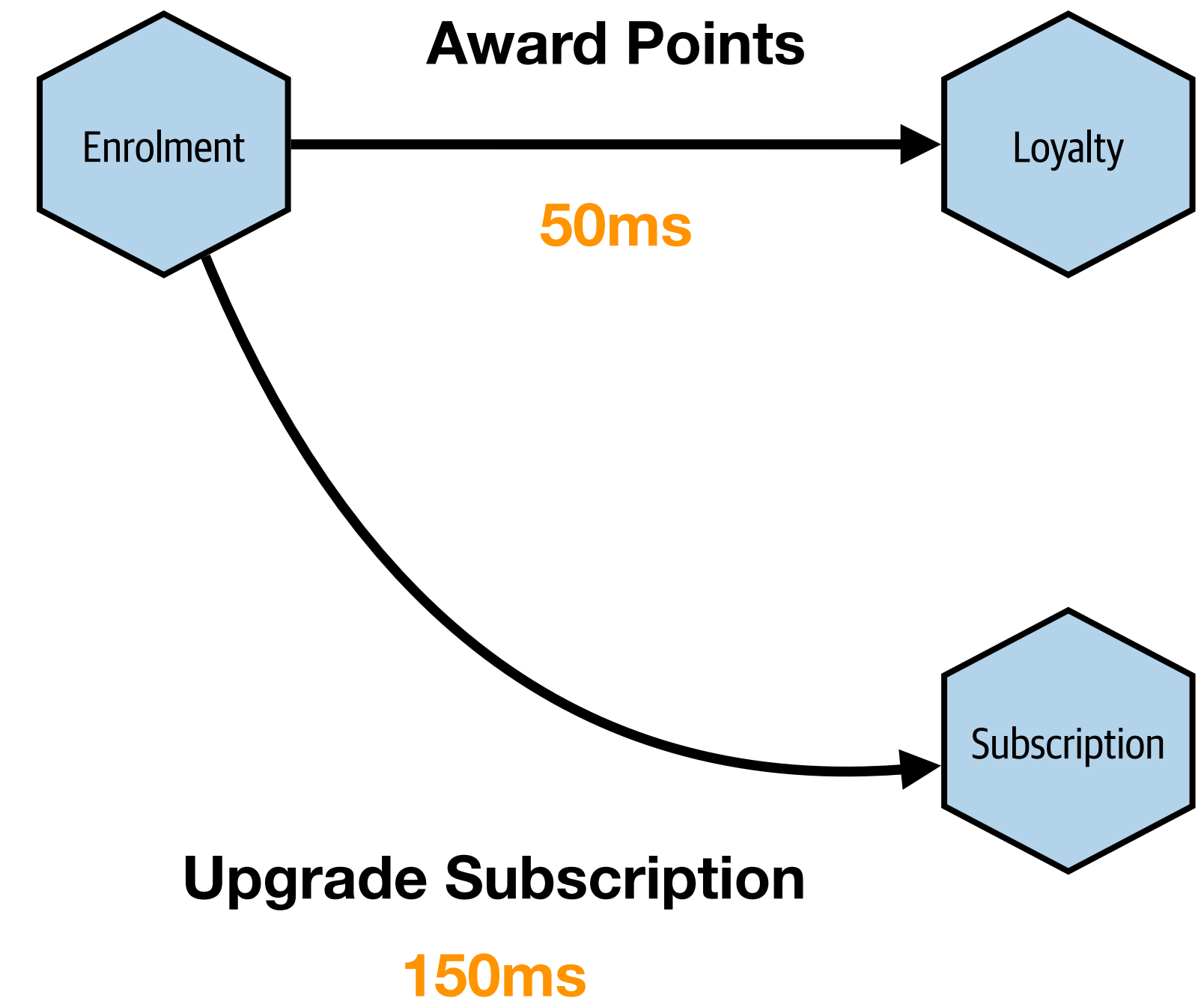
Latency is the sum of the calls



PROBLEMS WITH BLOCKING CALLS?

Latency is the sum of the calls

e.g. 50 + 150ms = 200ms in total

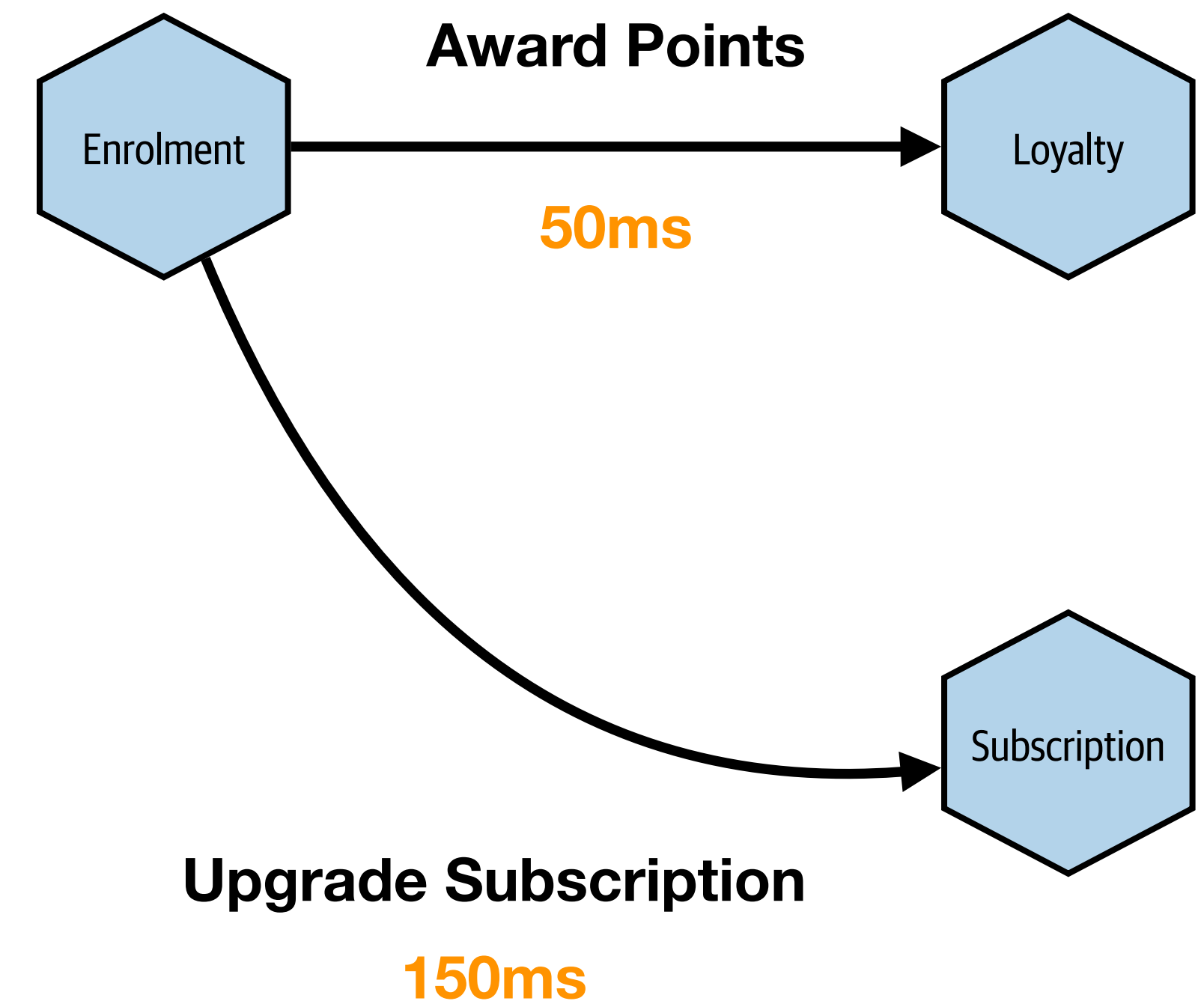


PROBLEMS WITH BLOCKING CALLS?

Latency is the sum of the calls

e.g. $50 + 150\text{ms} = 200\text{ms}$ in total

But remember that latency for an operation is not fixed - it will have a range

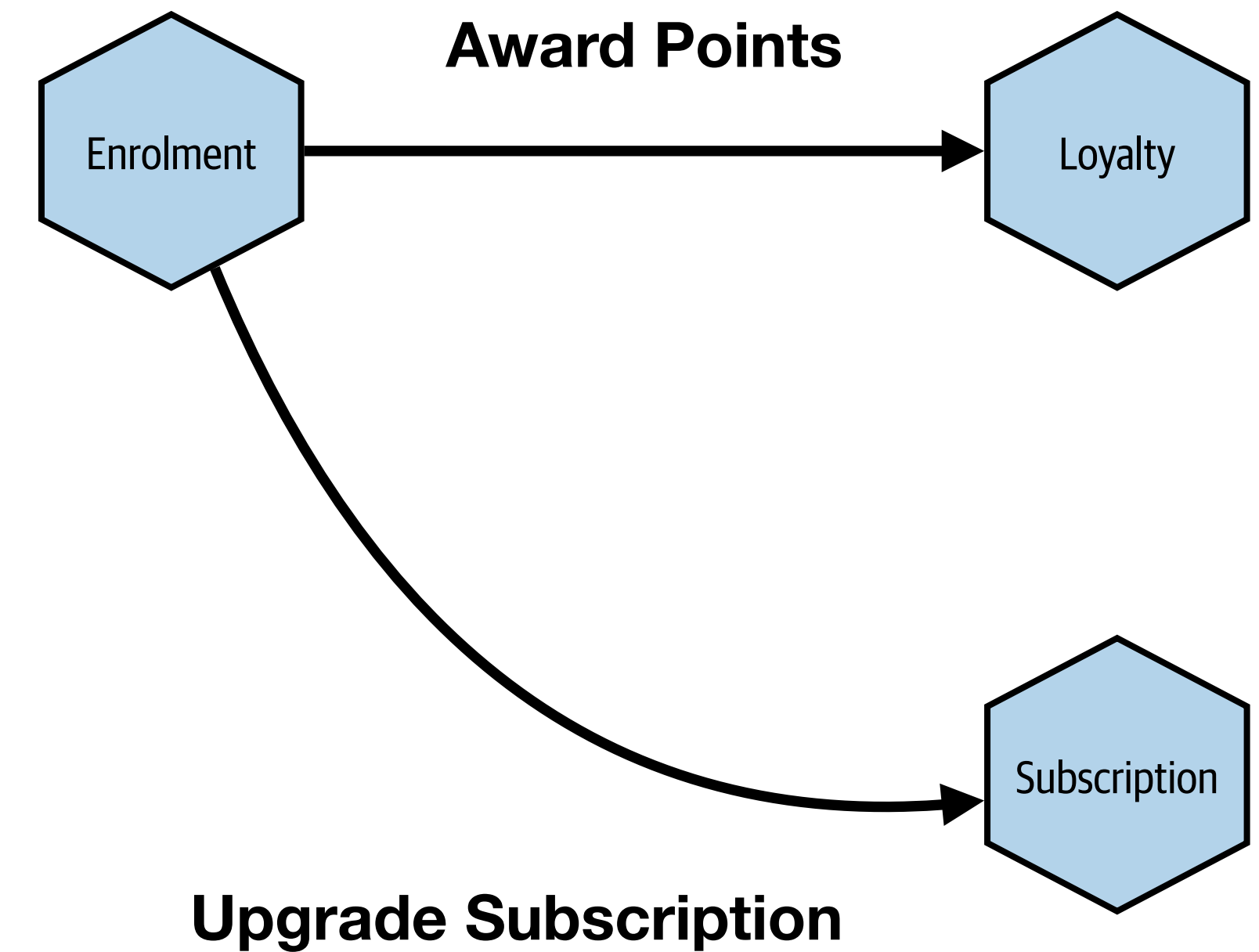


**An improvement would be to
do these two calls in parallel**

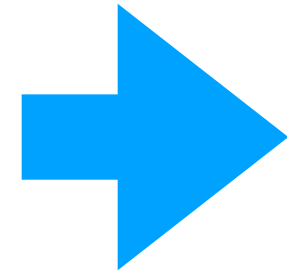
PARALLEL CALLS

```
res1 = loyaltyService.awardPoints(custID, 200)
```

```
res2 = subsService.upgradeSub(custID, PLATINUM)
```

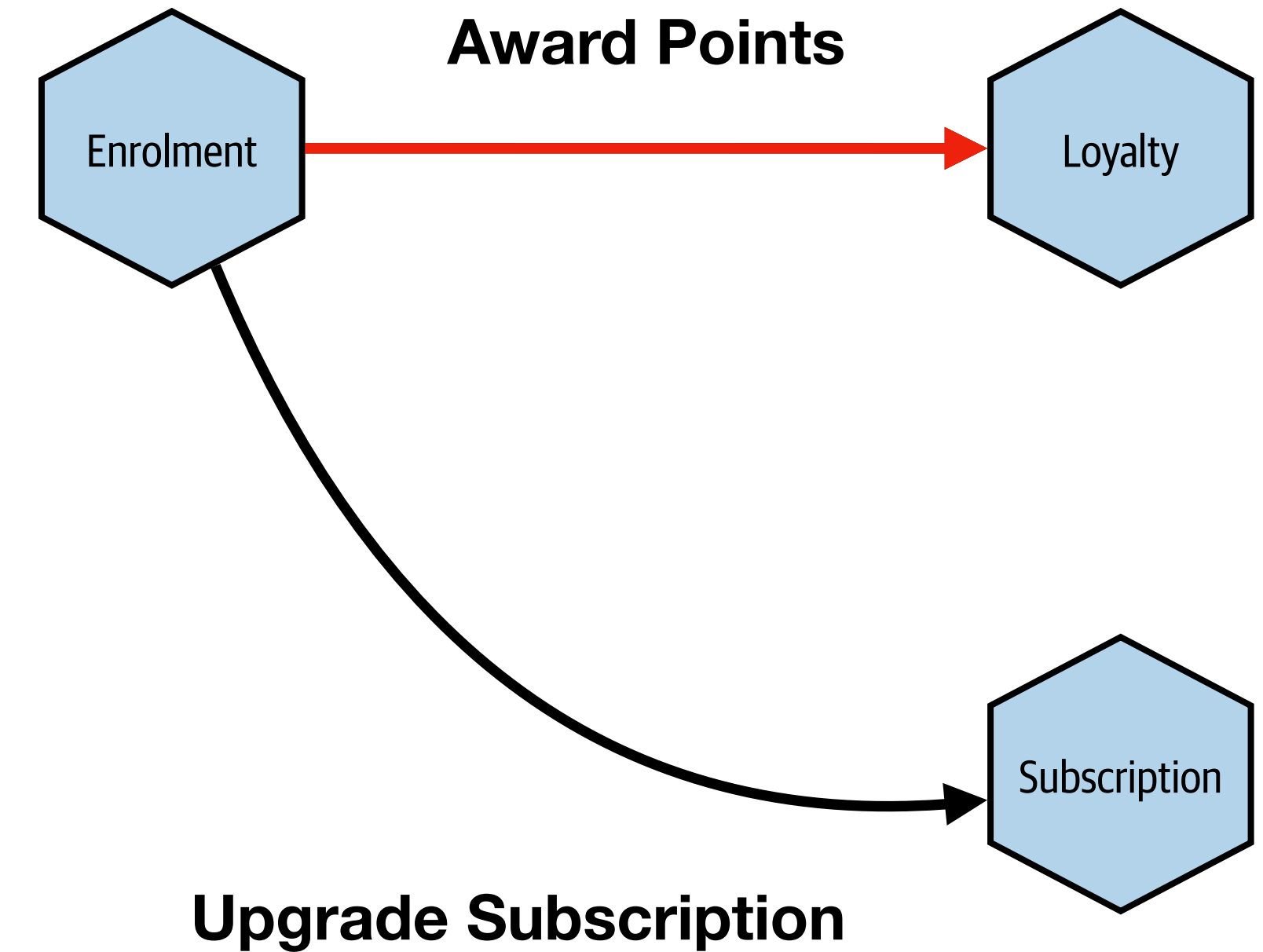


PARALLEL CALLS

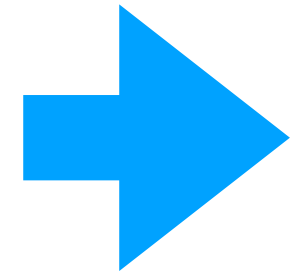


```
res1 = loyaltyService.awardPoints(custID, 200)
```

```
res2 = subsService.upgradeSub(custID, PLATINUM)
```

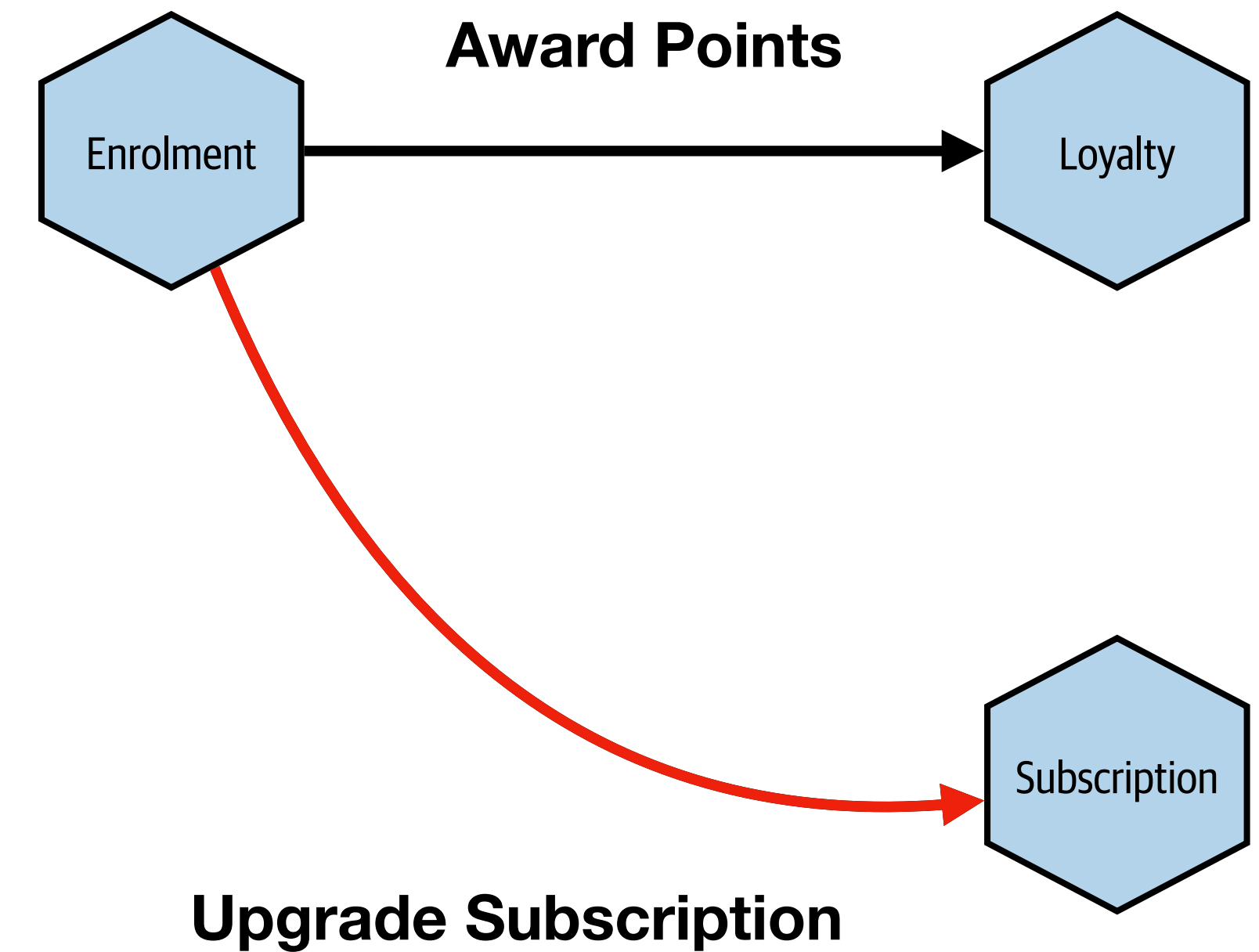


PARALLEL CALLS

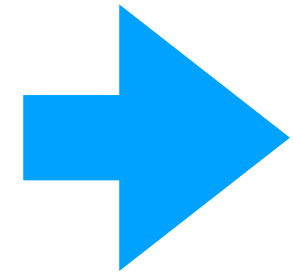


```
res1 = loyaltyService.awardPoints(custID, 200)
```

```
res2 = subsService.upgradeSub(custID, PLATINUM)
```

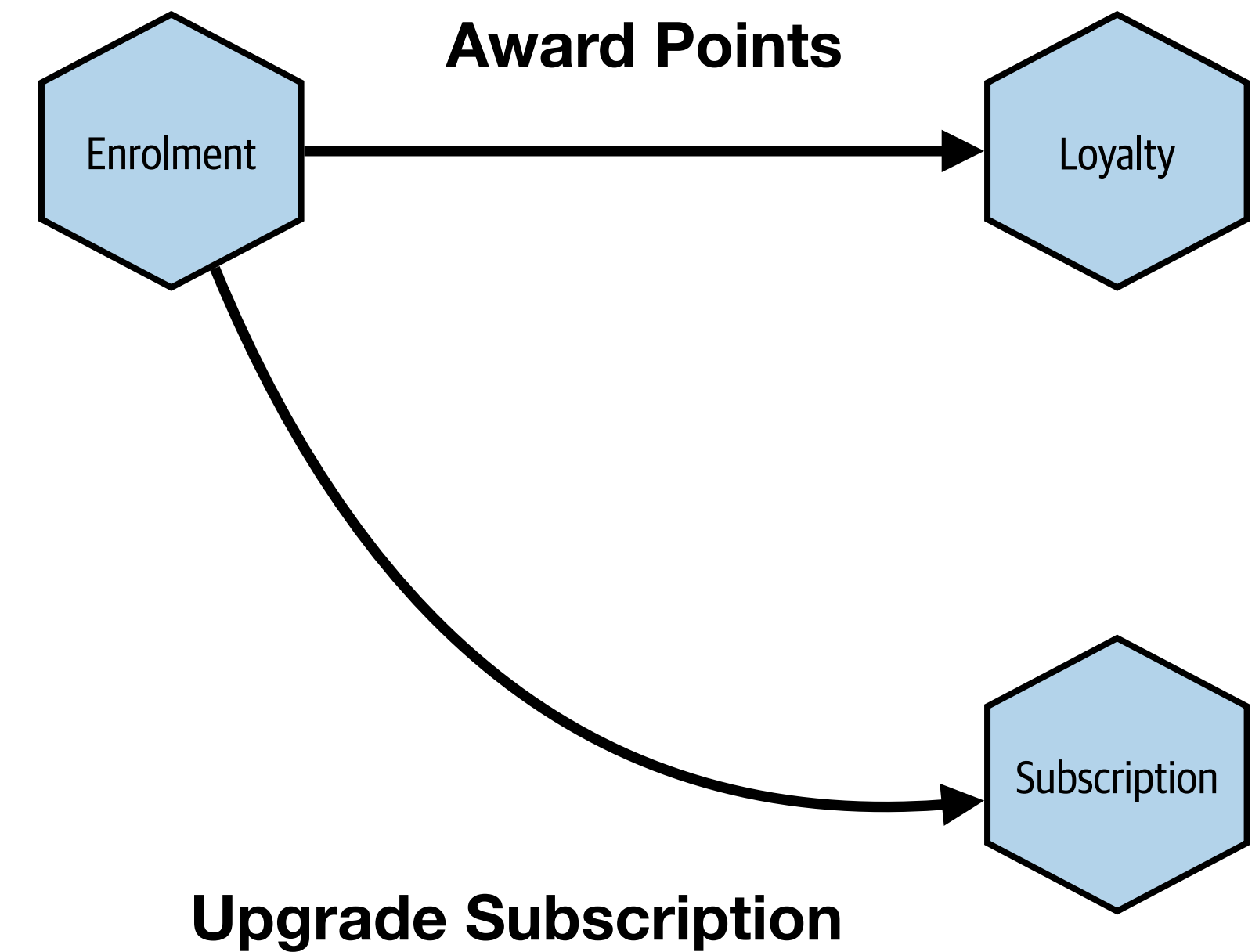


PARALLEL CALLS

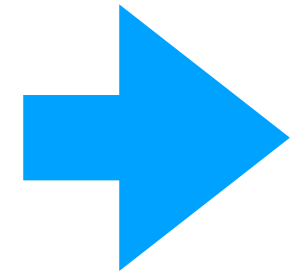


```
res1 = loyaltyService.awardPoints(custID, 200)
```

```
res2 = subsService.upgradeSub(custID, PLATINUM)
```



PARALLEL CALLS

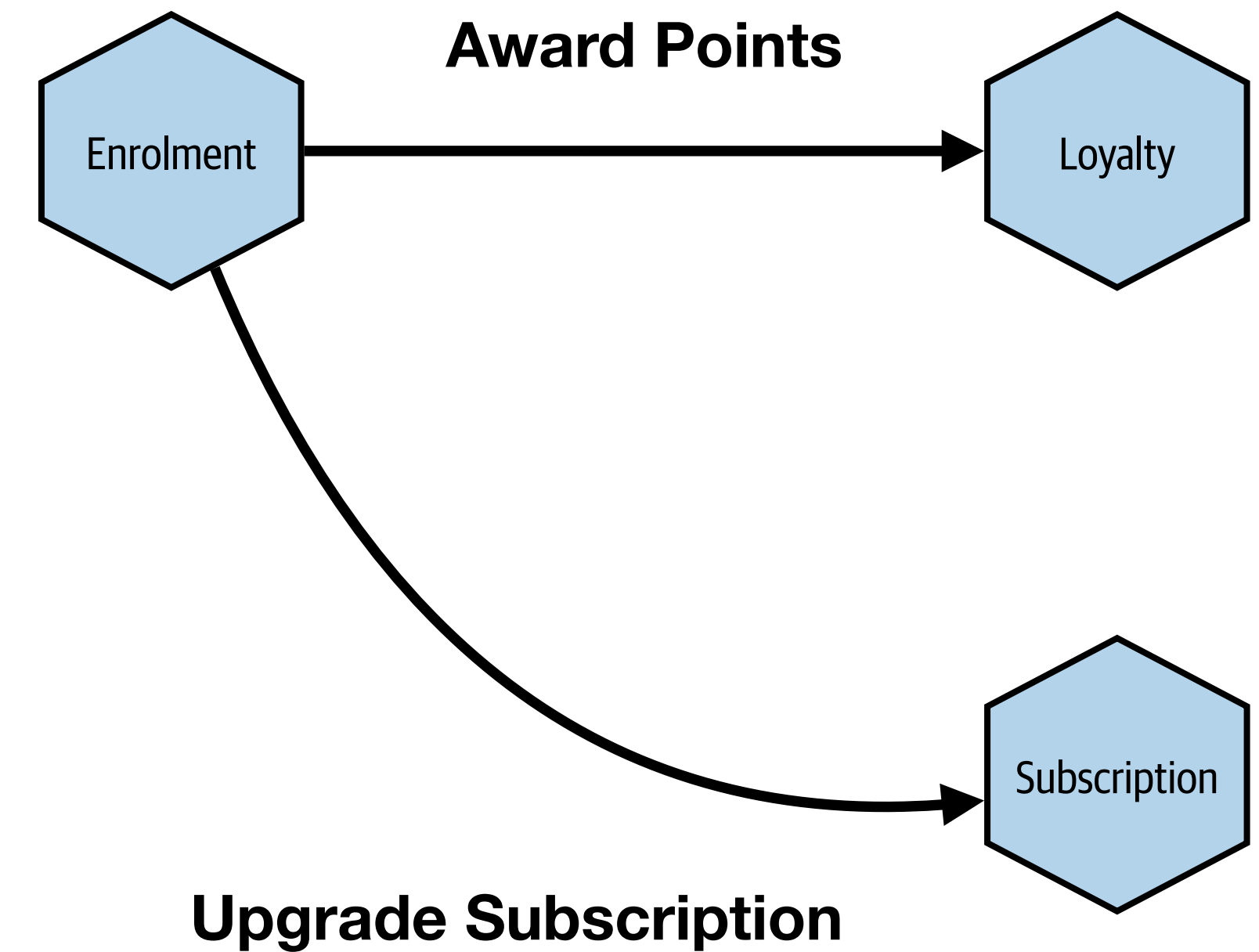


```
res1 = loyaltyService.awardPoints(custID, 200)
```

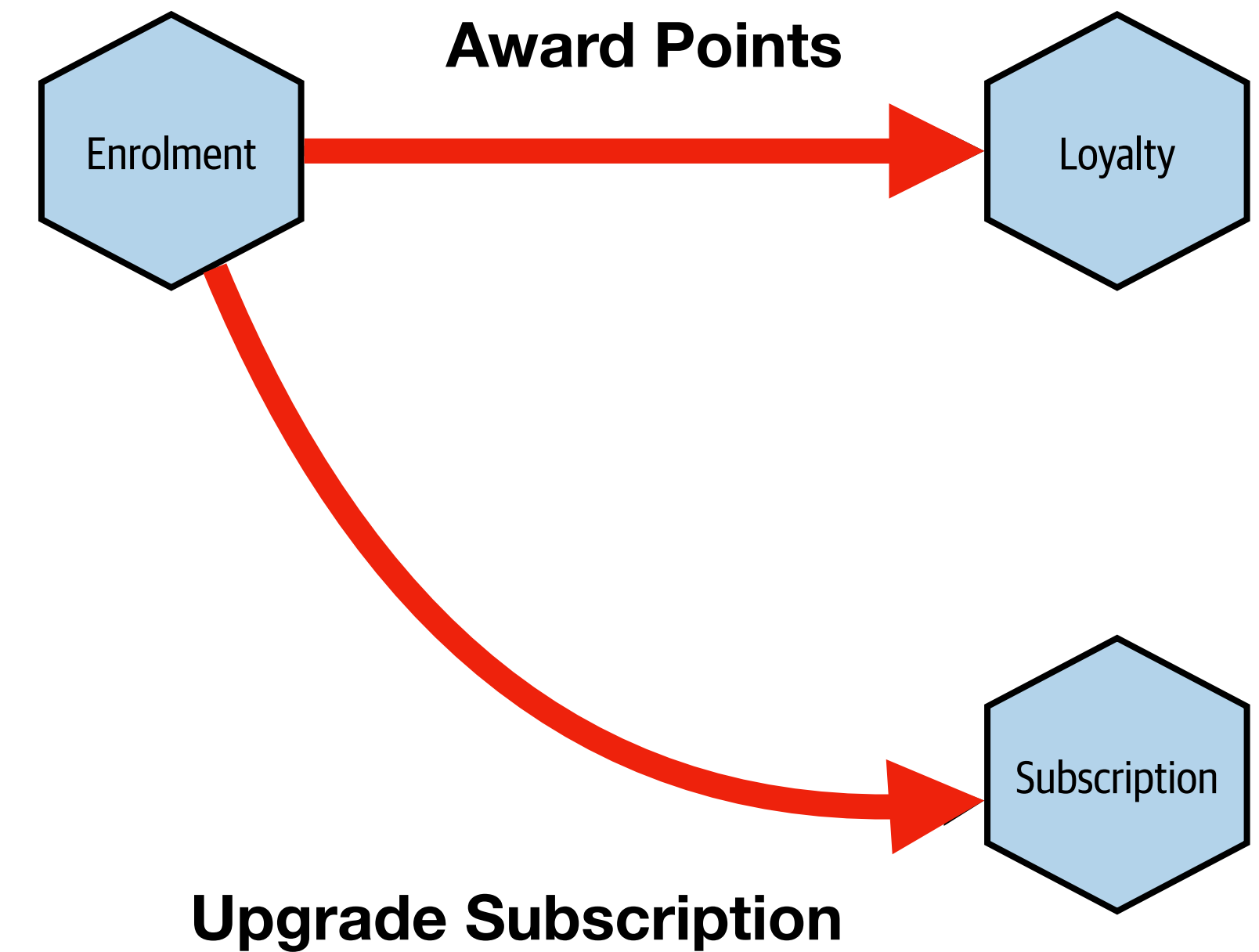
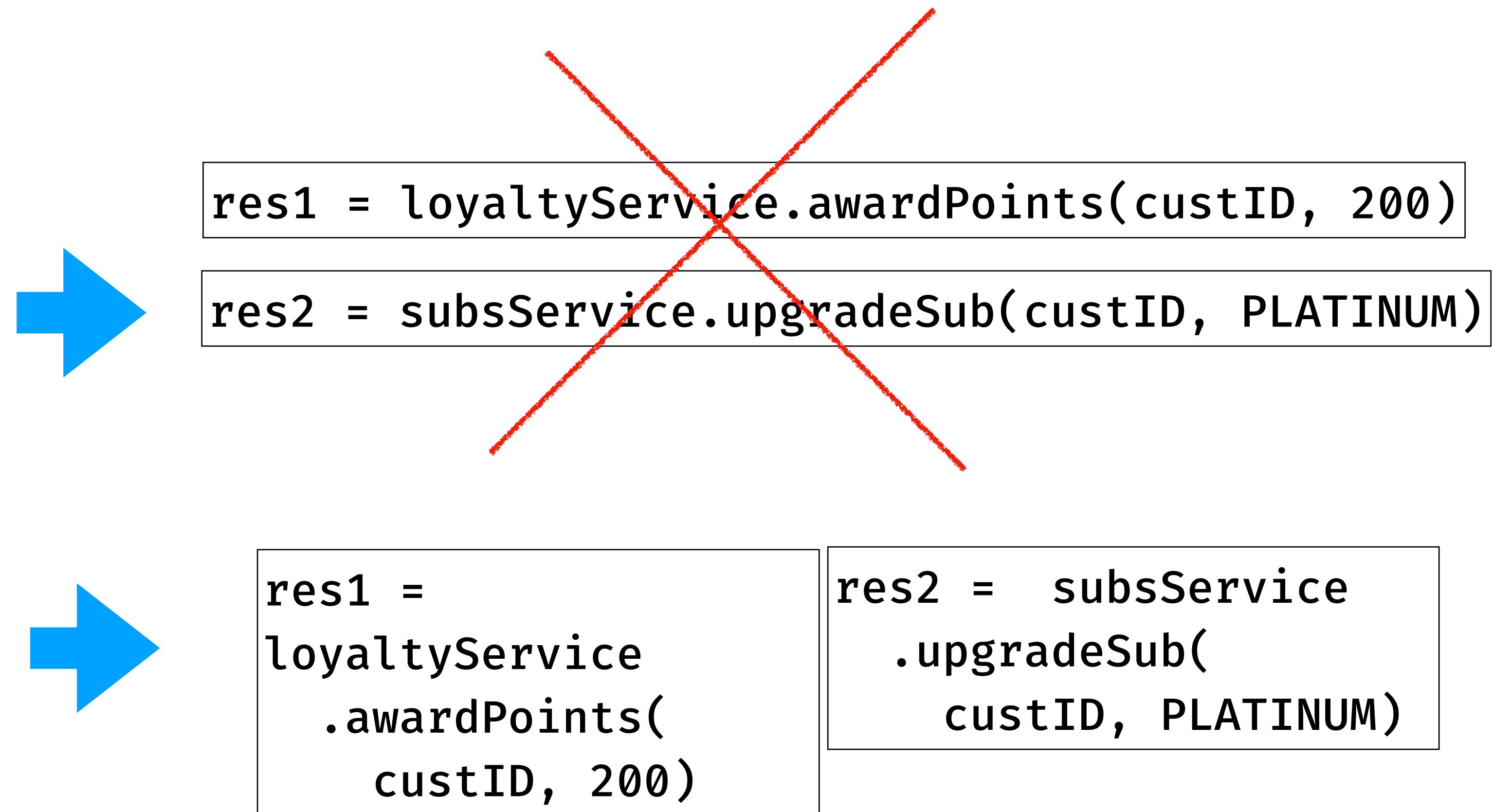
```
res2 = subsService.upgradeSub(custID, PLATINUM)
```

```
res1 =  
loyaltyService  
  .awardPoints(  
    custID, 200)
```

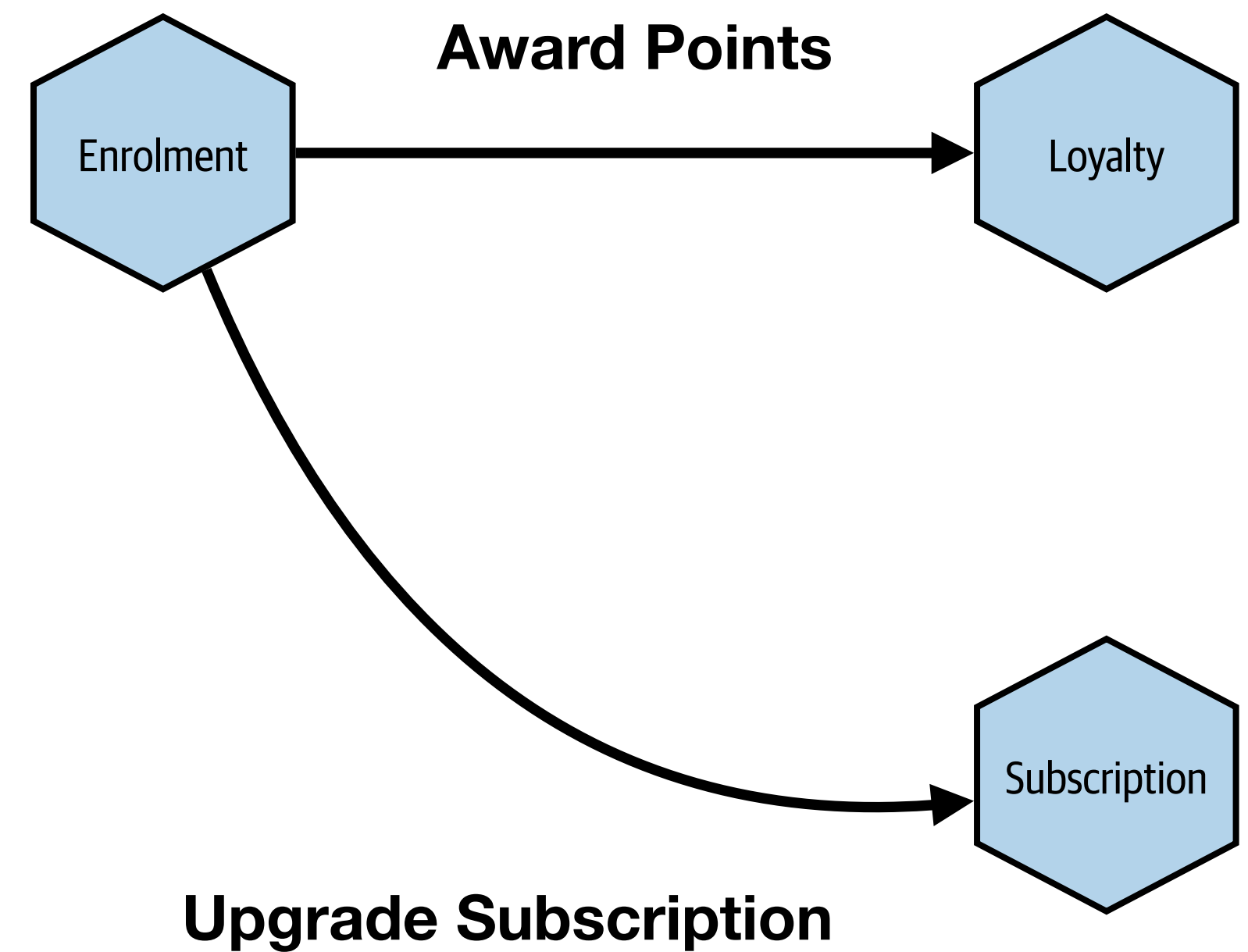
```
res2 = subsService  
  .upgradeSub(  
    custID, PLATINUM)
```



PARALLEL CALLS

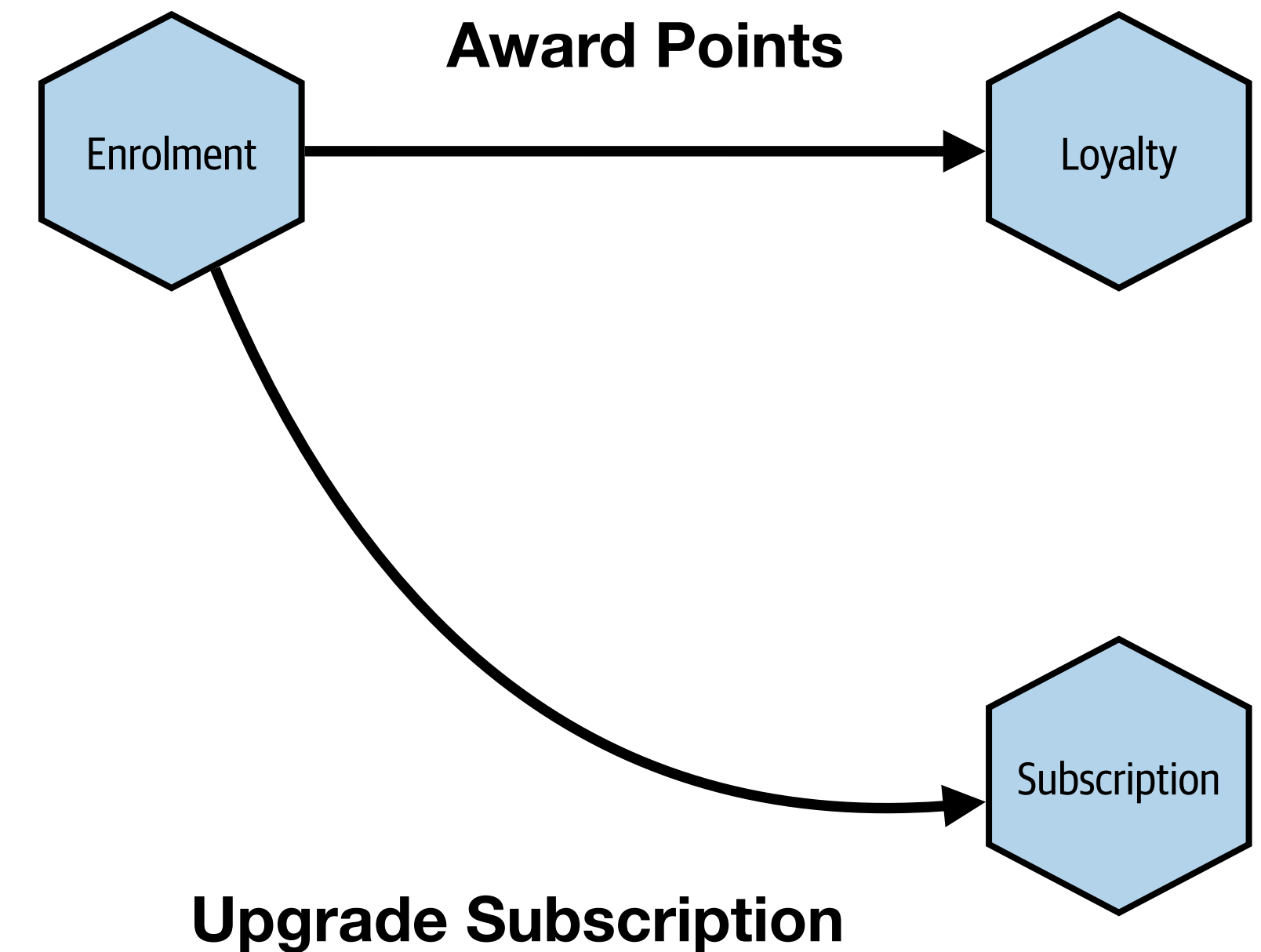


```
fut1 = future(loyaltyService.awardPoints(custID, 200))  
fut2 = future(subsService.upgradeSub(custID, PLATINUM))
```



The future represents a background thread of execution - we can continue while the call is made in the background

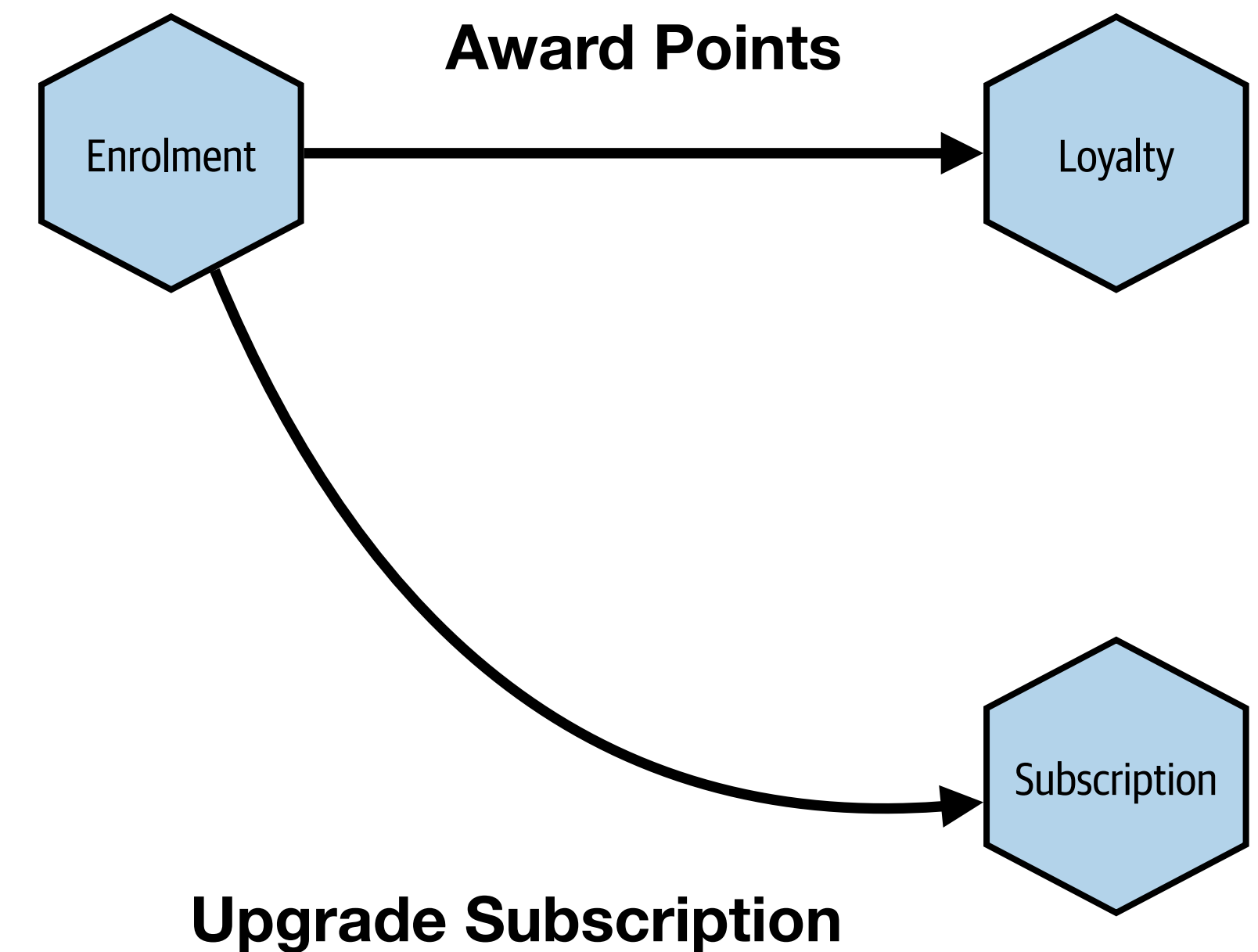
```
fut1 = future(loyaltyService.awardPoints(custID, 200))  
fut2 = future(subsService.upgradeSub(custID, PLATINUM))
```



The future represents a background thread of execution - we can continue while the call is made in the background

```
fut1 = future(loyaltyService.awardPoints(custID, 200))  
fut2 = future(subsService.upgradeSub(custID, PLATINUM))
```

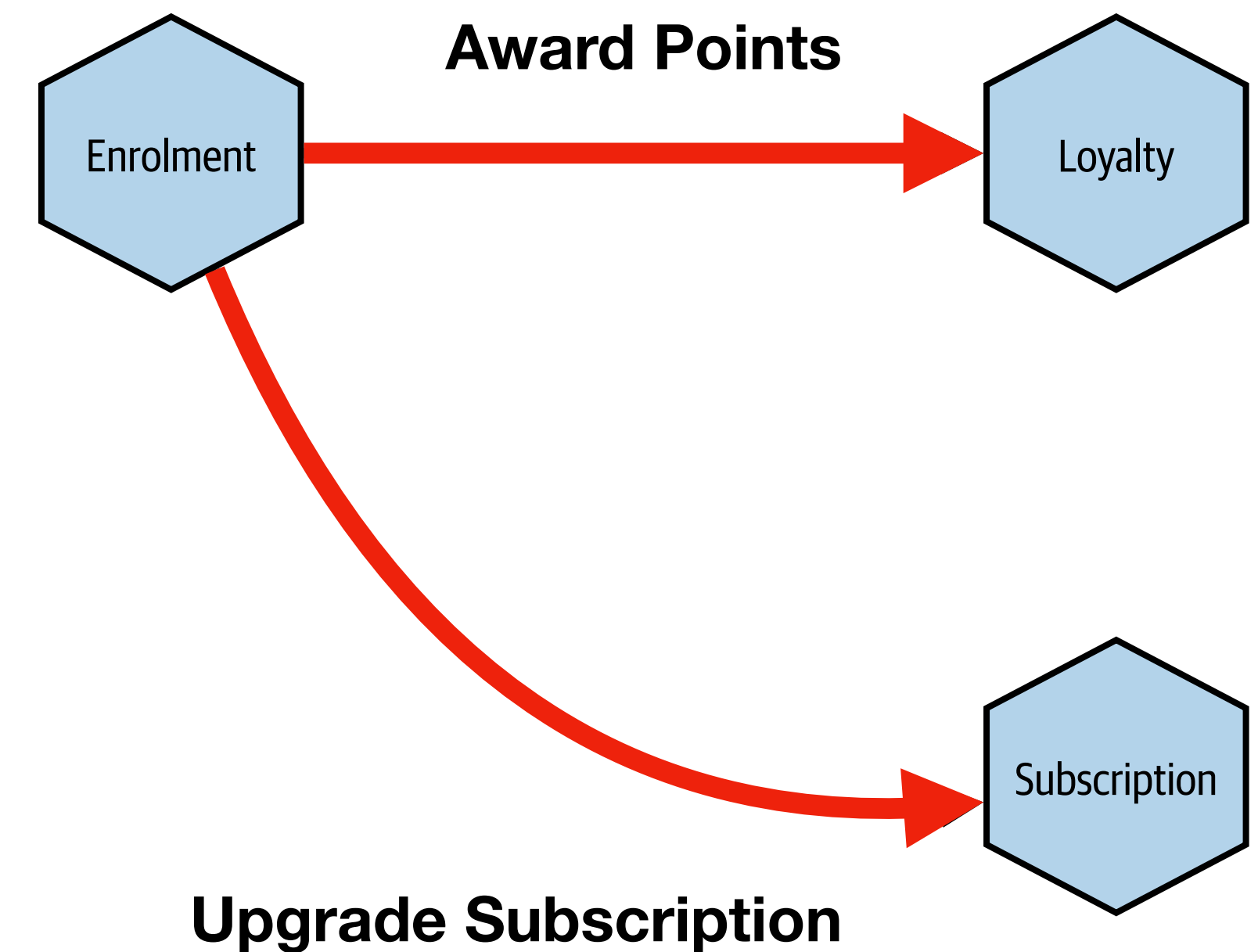
So the calls to Loyalty and Subscription service can now be made in parallel



The future represents a background thread of execution - we can continue while the call is made in the background

```
fut1 = future(loyaltyService.awardPoints(custID, 200))  
fut2 = future(subsService.upgradeSub(custID, PLATINUM))
```

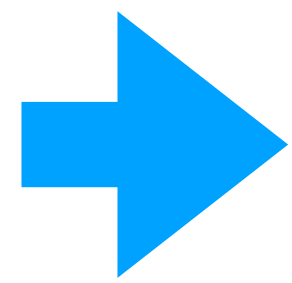
So the calls to Loyalty and Subscription service can now be made in parallel



**But we may actually need the
responses, right?**

```
fut1 = future(loyaltyService.awardPoints(custID, 200))  
fut2 = future(subsService.upgradeSub(custID, PLATINUM))
```

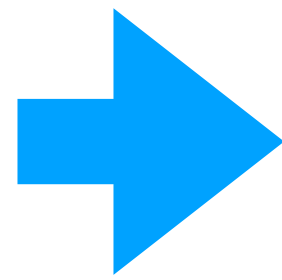
```
fut1 = future(loyaltyService.awardPoints(custID, 200))
```



```
fut2 = future(subsService.upgradeSub(custID, PLATINUM))
```

```
fut1 = future(loyaltyService.awardPoints(custID, 200))
```

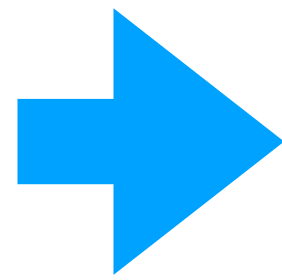
```
fut2 = future(subsService.upgradeSub(custID, PLATINUM))
```



At this point, we don't know if the calls to either service have been processed or not.

```
fut1 = future(loyaltyService.awardPoints(custID, 200))
```

```
fut2 = future(subsService.upgradeSub(custID, PLATINUM))
```

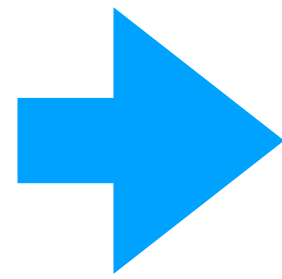


At this point, we don't know if the calls to either service have been processed or not.

We might have to wait for the results!

```
fut1 = future(loyaltyService.awardPoints(custID, 200))
```

```
fut2 = future(subsService.upgradeSub(custID, PLATINUM))
```



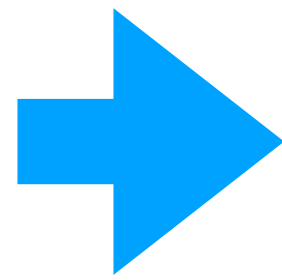
At this point, we don't know if the calls to either service have been processed or not.

We might have to wait for the results!

```
await(fut1, fut2)
```

```
fut1 = future(loyaltyService.awardPoints(custID, 200))
```

```
fut2 = future(subsService.upgradeSub(custID, PLATINUM))
```



At this point, we don't know if the calls to either service have been processed or not.

We might have to wait for the results!

```
await(fut1, fut2)
```

And now we're blocking program execution....

**Even if you have non-blocking calls,
the logic of your processing may
require waiting for things to happen!**

**Even non-blocking calls may
end up blocking**

**But in general if you can use
non-blocking calls and run them
in parallel, that is a good thing.**



Graham Lea @evolvable · Jul 13, 2021



Replying to @samnewman

The definition I've been using is:

“The sending service doesn't wait for completion of the receiving service before continuing and/or completing its own work.”

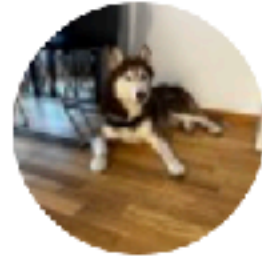
But now that I think about it, I think I also expect a temporal decoupling from the receiving service's availability.



Sam Newman ✓ @samnewman · Jul 13, 2021

So I think the distinction for many people between synchronous and asynchronous inter-service communication to be **really** fuzzy. What does async comms mean to you?

<https://twitter.com/evolvable/status/1414909633597116417>



Benjamin Johnsson

@Benkzxar



Replying to [@samnewman](#)

Sync communication = there is a direct communication between the Sender and the Receiver.

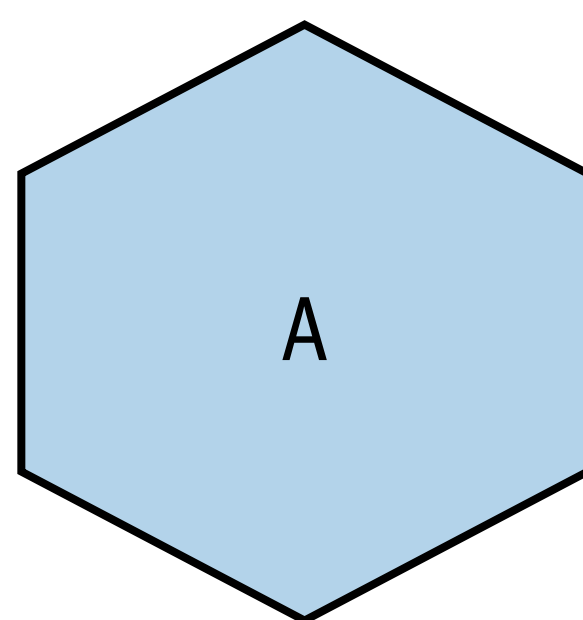
Async = there is an intermediary involved between the sender and the receiver, for example the message broker.

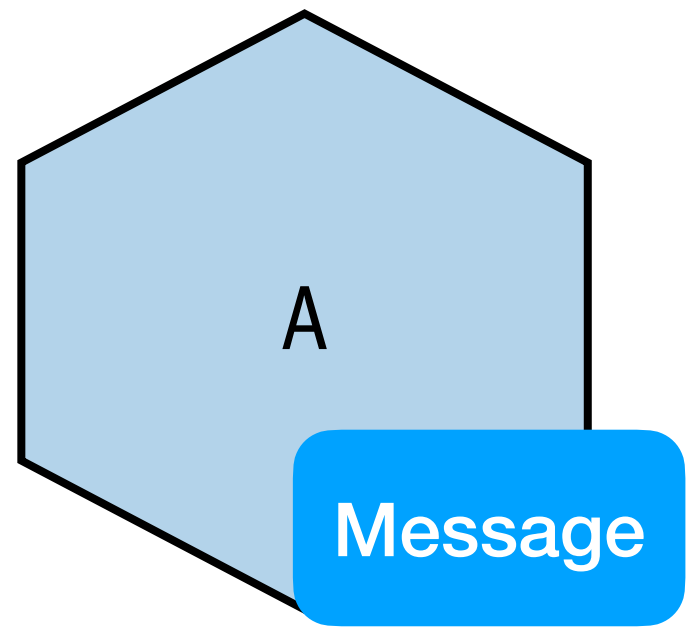
4:09 PM · Jul 13, 2021

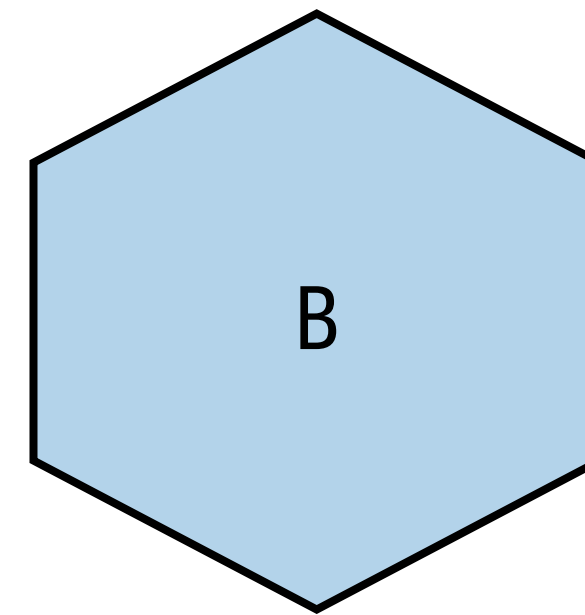
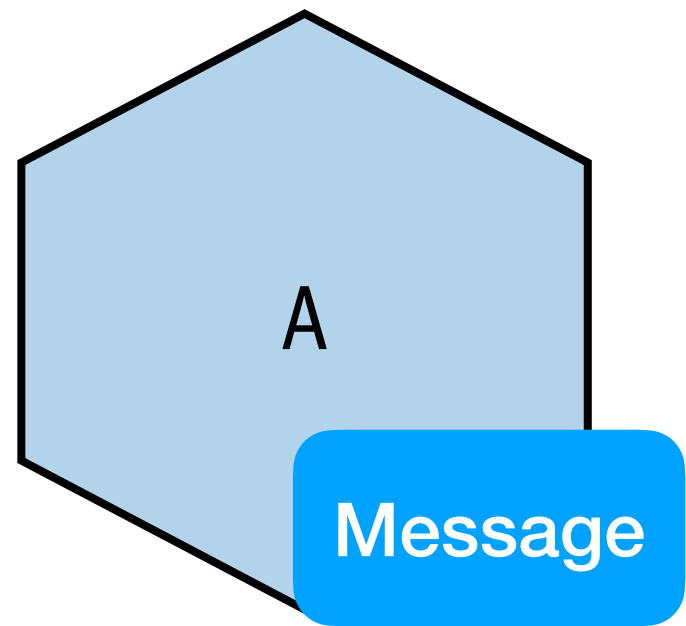
<https://twitter.com/Benkzxar/status/1414965335338496001>

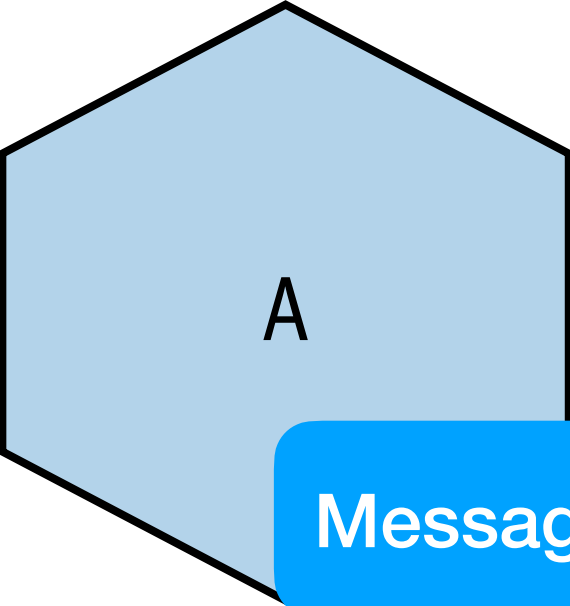
Intermediary?

Broker

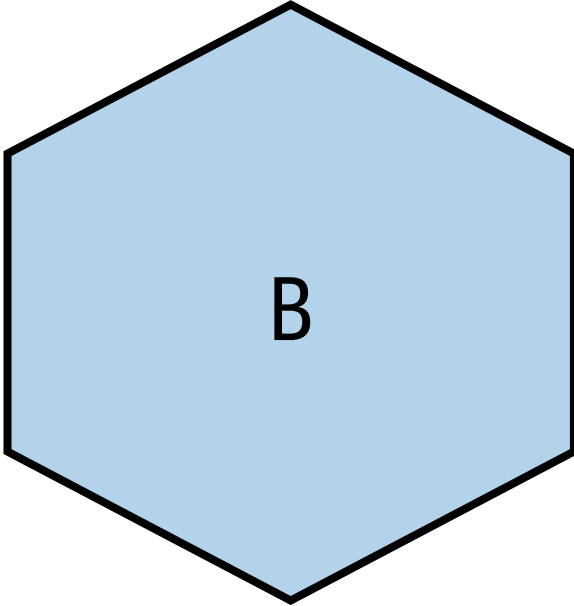
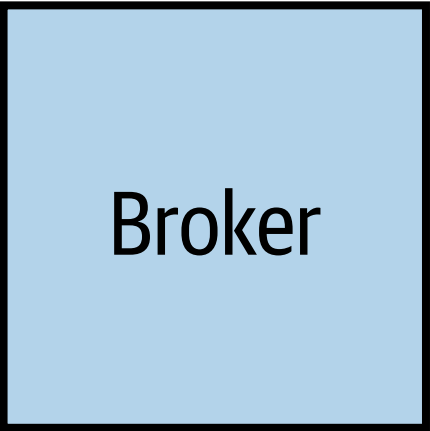


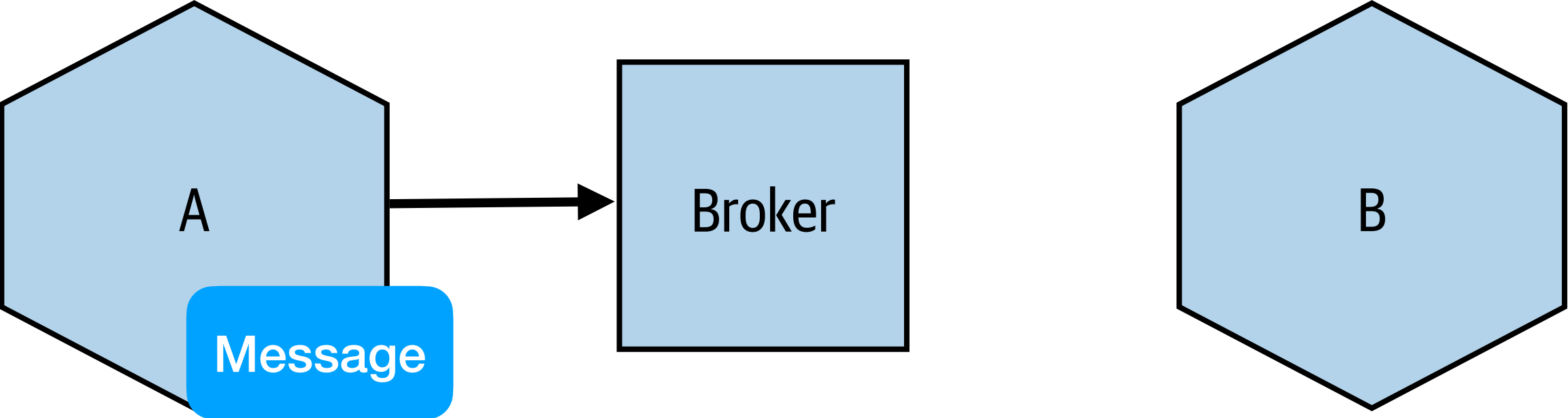


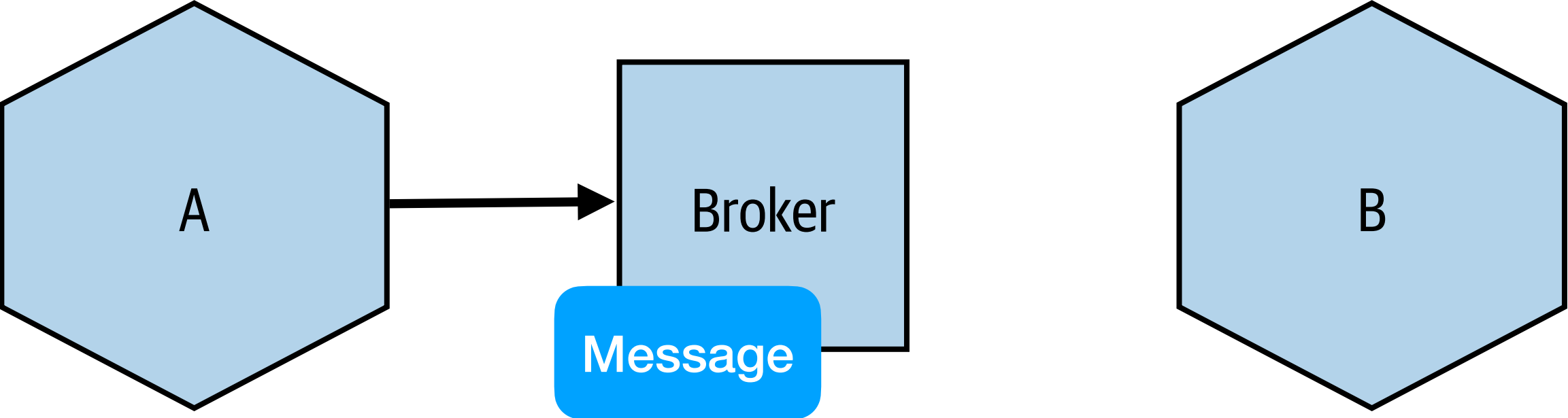


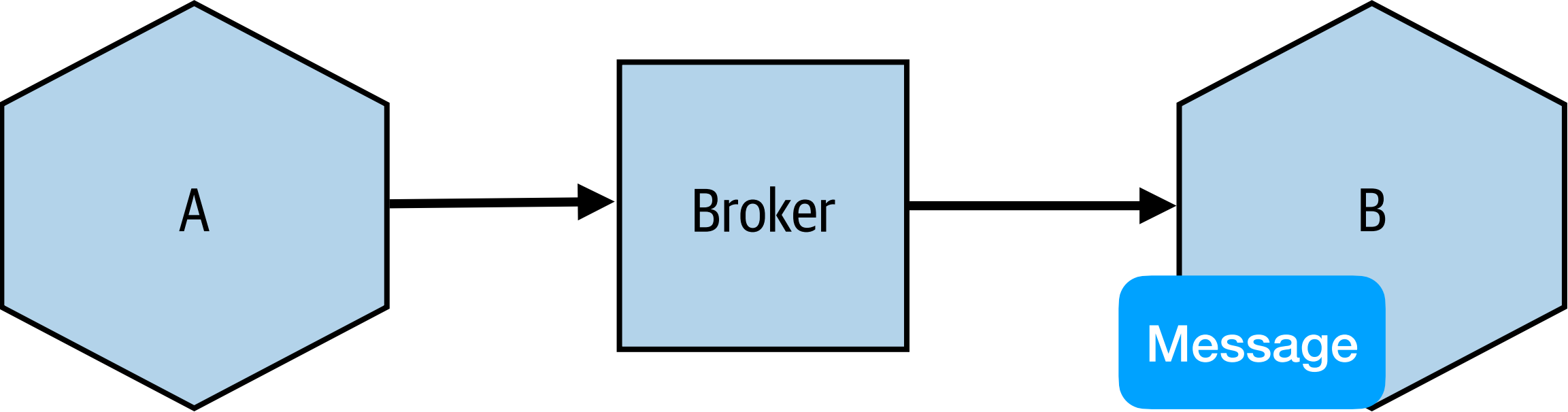


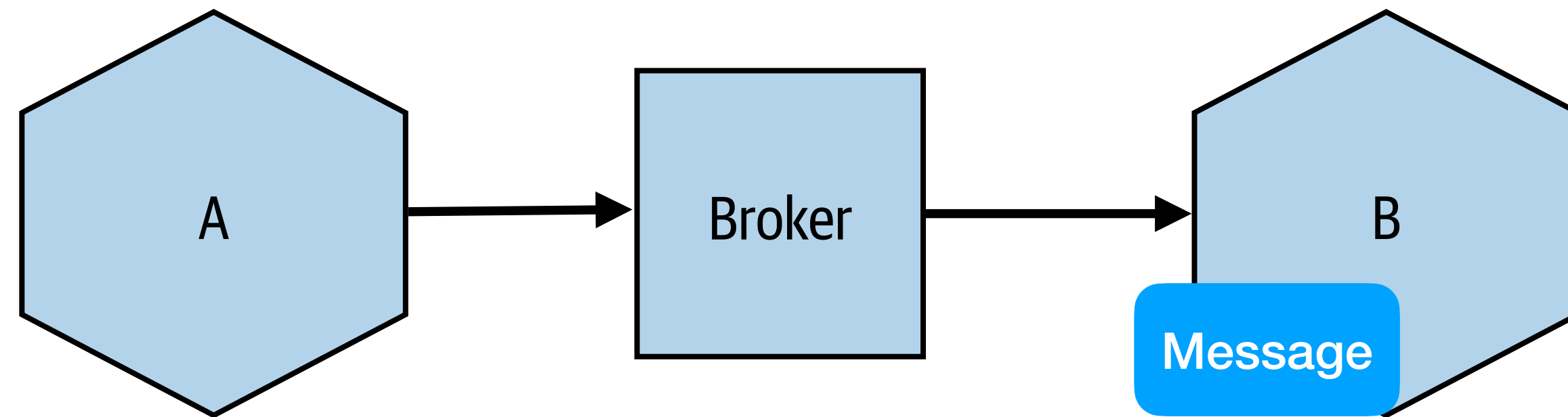
Message











With a broker, we can offload the requirement to guarantee delivery (kinda)



goingkilo @goingkilo · Jul 13, 2021



Replying to [@samnewman](#)

async is like dropping (snail) mail in a (red) postbox then going on with your life

then one day, boom! postman pat stuffs something into your house mailbox, along with the millions of coupons and charity mailers.

<https://twitter.com/goingkilo/status/1414898869075251201>



Post
Office

Royal
Mail

PAT 1



**There are variations on how we can
implement this “inbox” pattern**

**There are variations on how we can
implement this “inbox” pattern**

Database

**There are variations on how we can
implement this “inbox” pattern**

Database

File system

**There are variations on how we can
implement this “inbox” pattern**

Database

File system

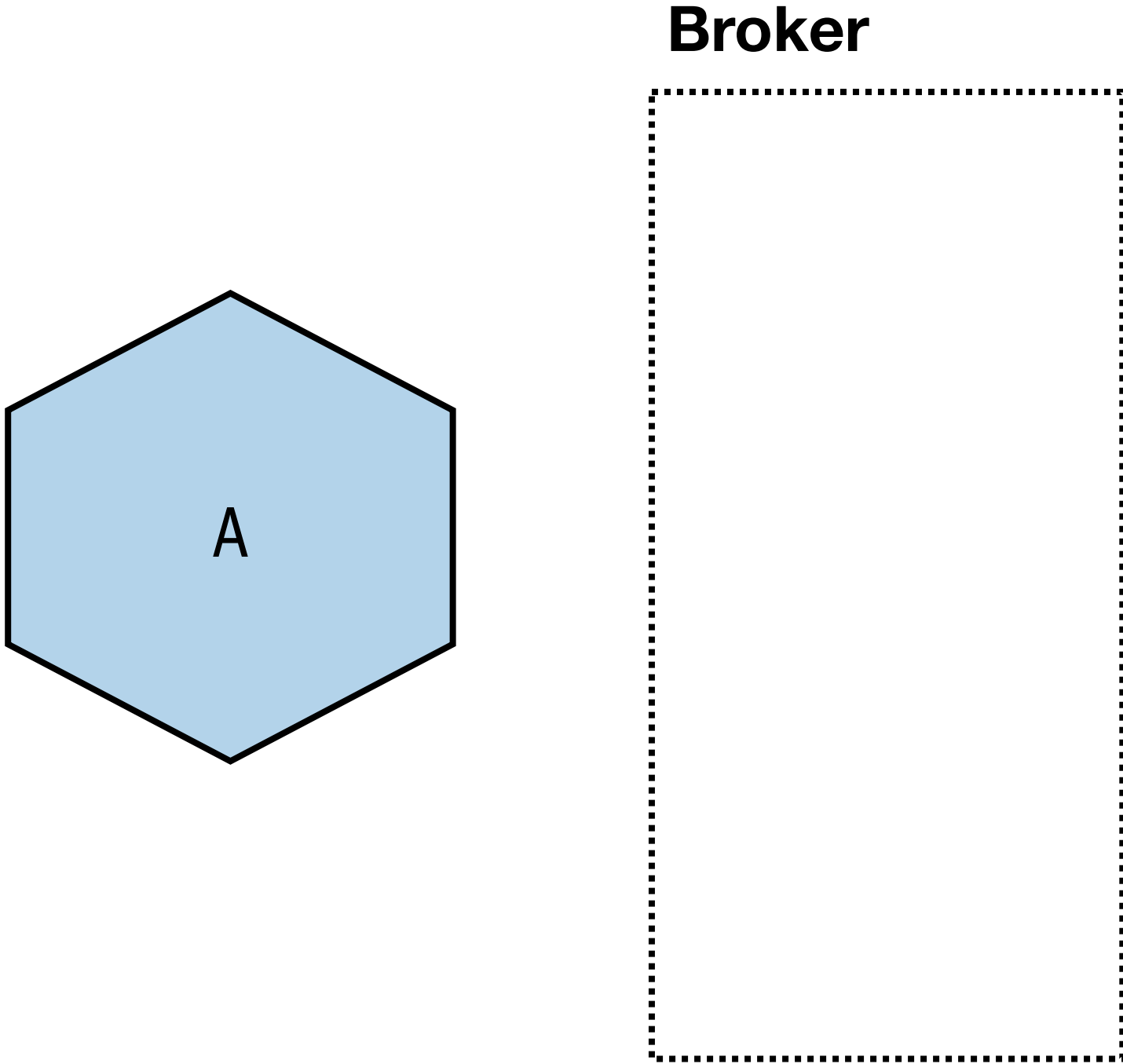
Email!

**The sender of a message doesn't need to
worry about whether or not the recipient is
currently available**

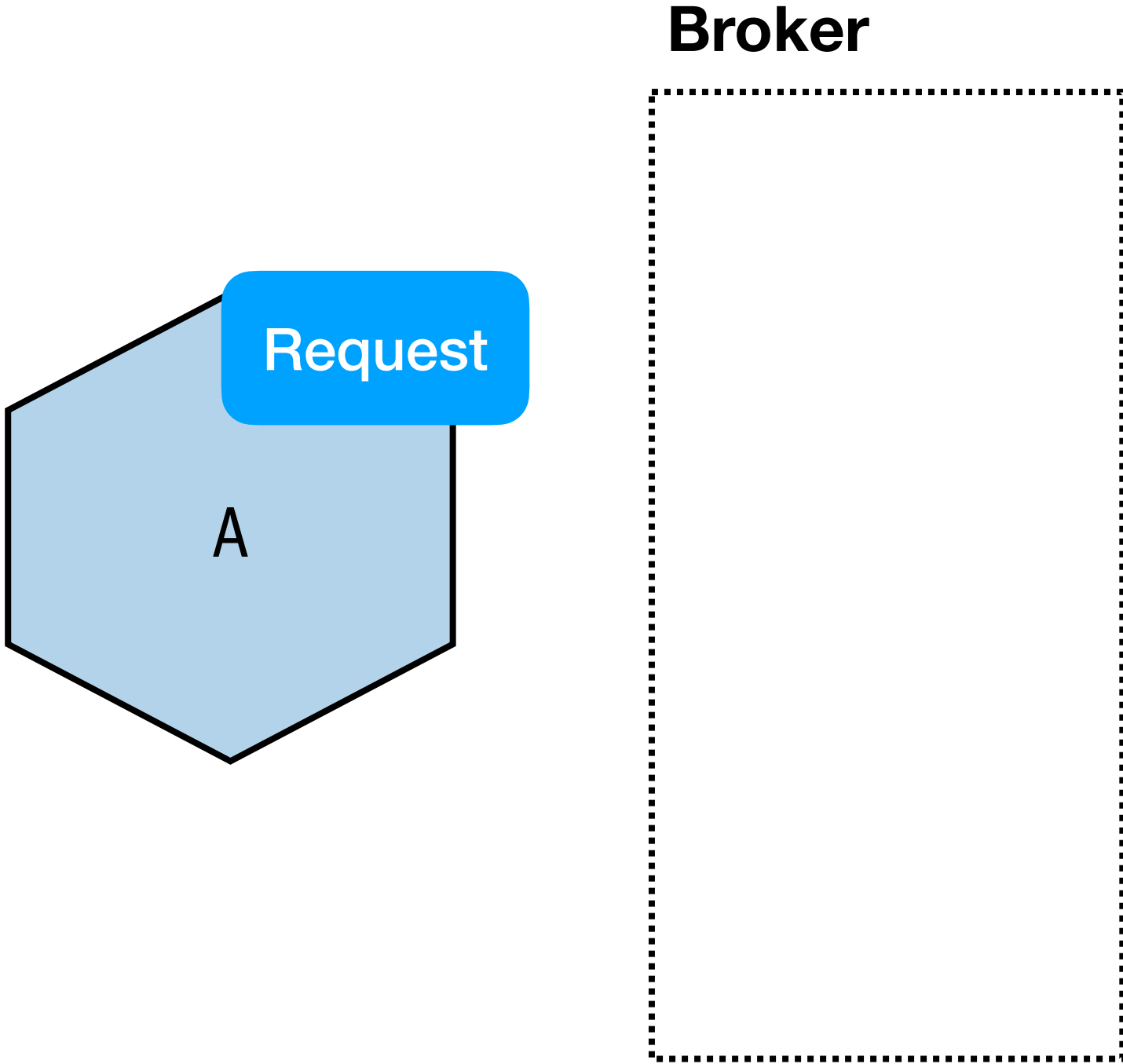
The sender of a message doesn't need to worry about whether or not the recipient is currently available

But we do need to trust the intermediary (the broker), and that does have to be available!

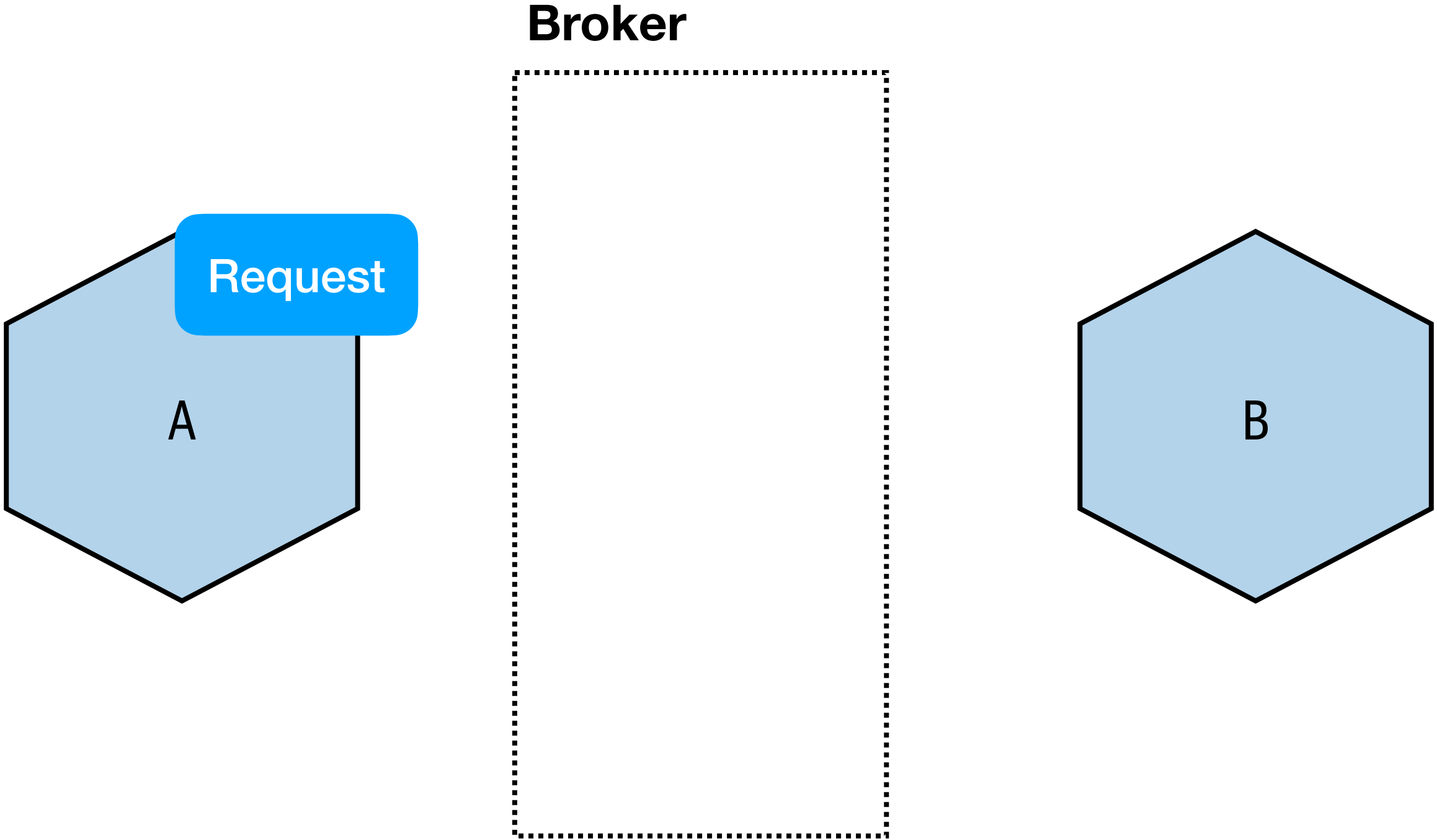
BROKER-BASED REQUEST-RESPONSE



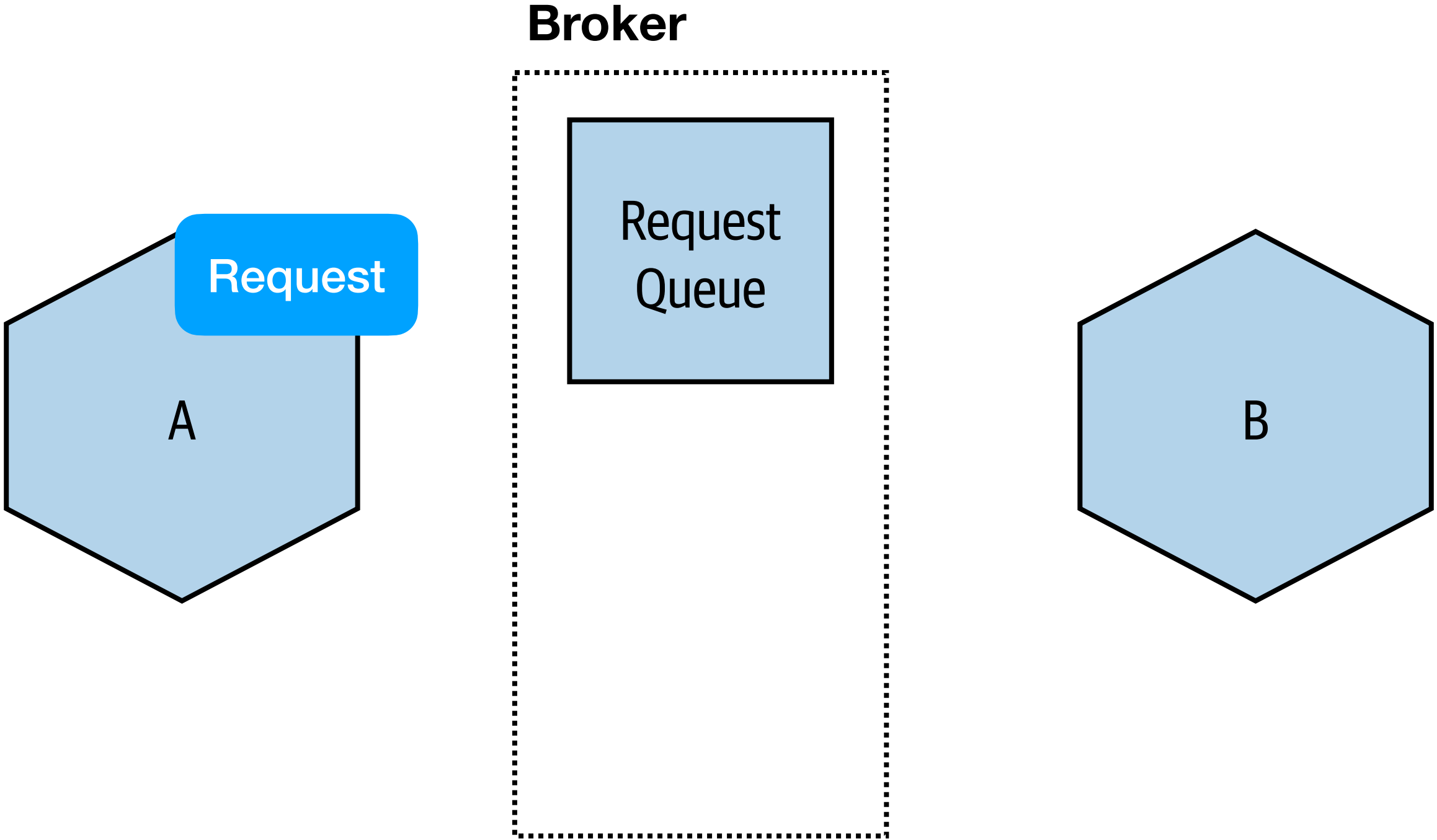
BROKER-BASED REQUEST-RESPONSE



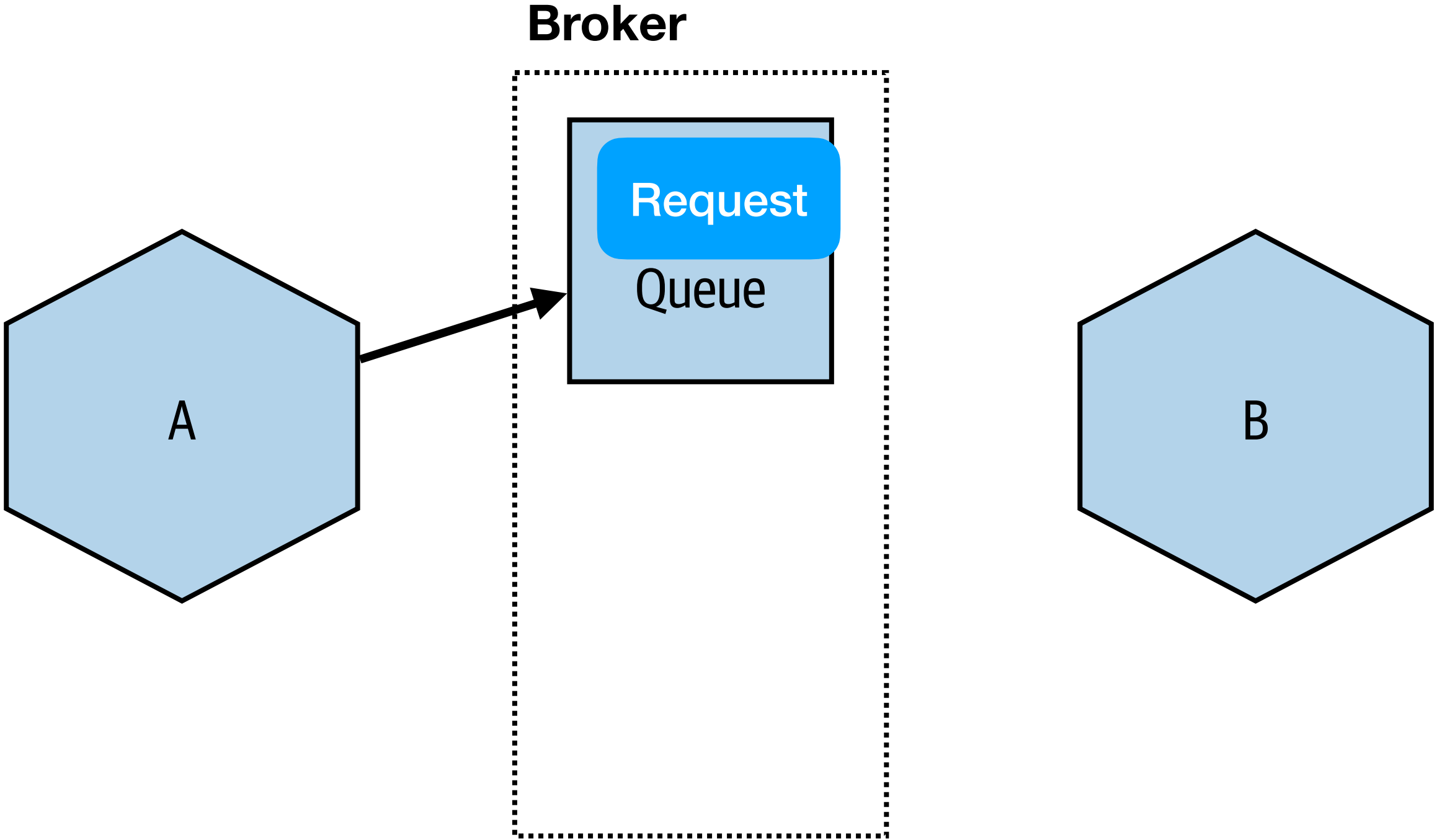
BROKER-BASED REQUEST-RESPONSE



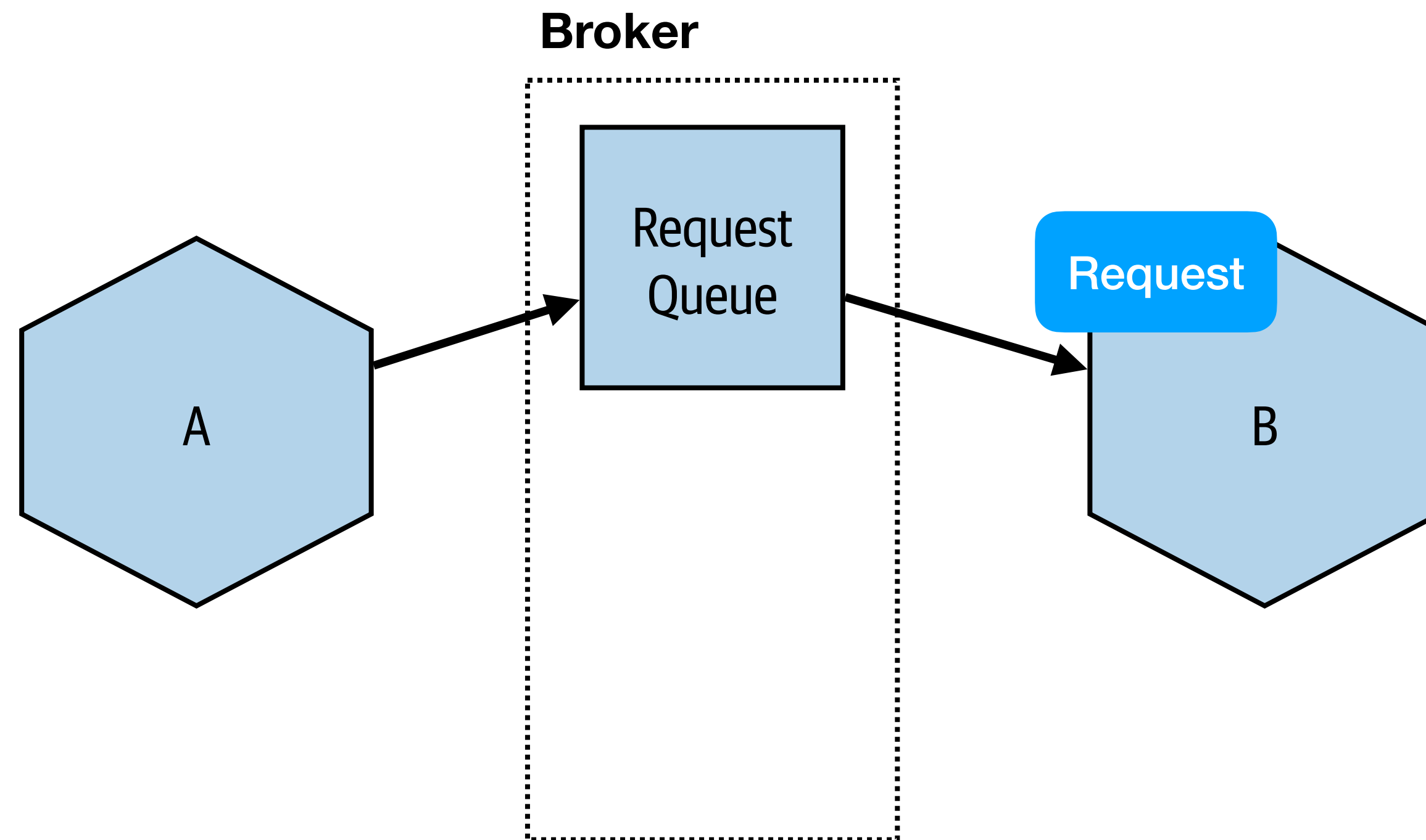
BROKER-BASED REQUEST-RESPONSE



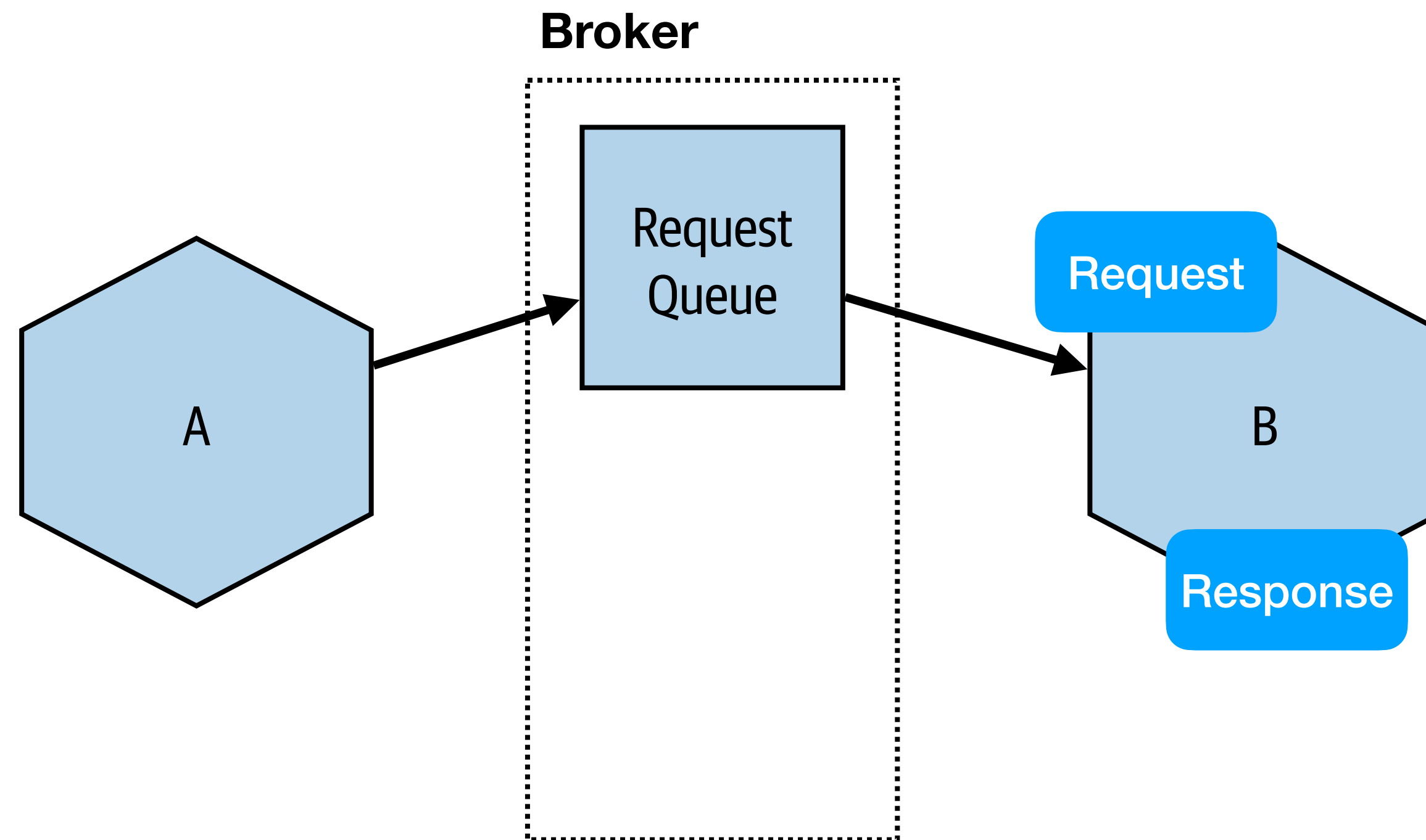
BROKER-BASED REQUEST-RESPONSE



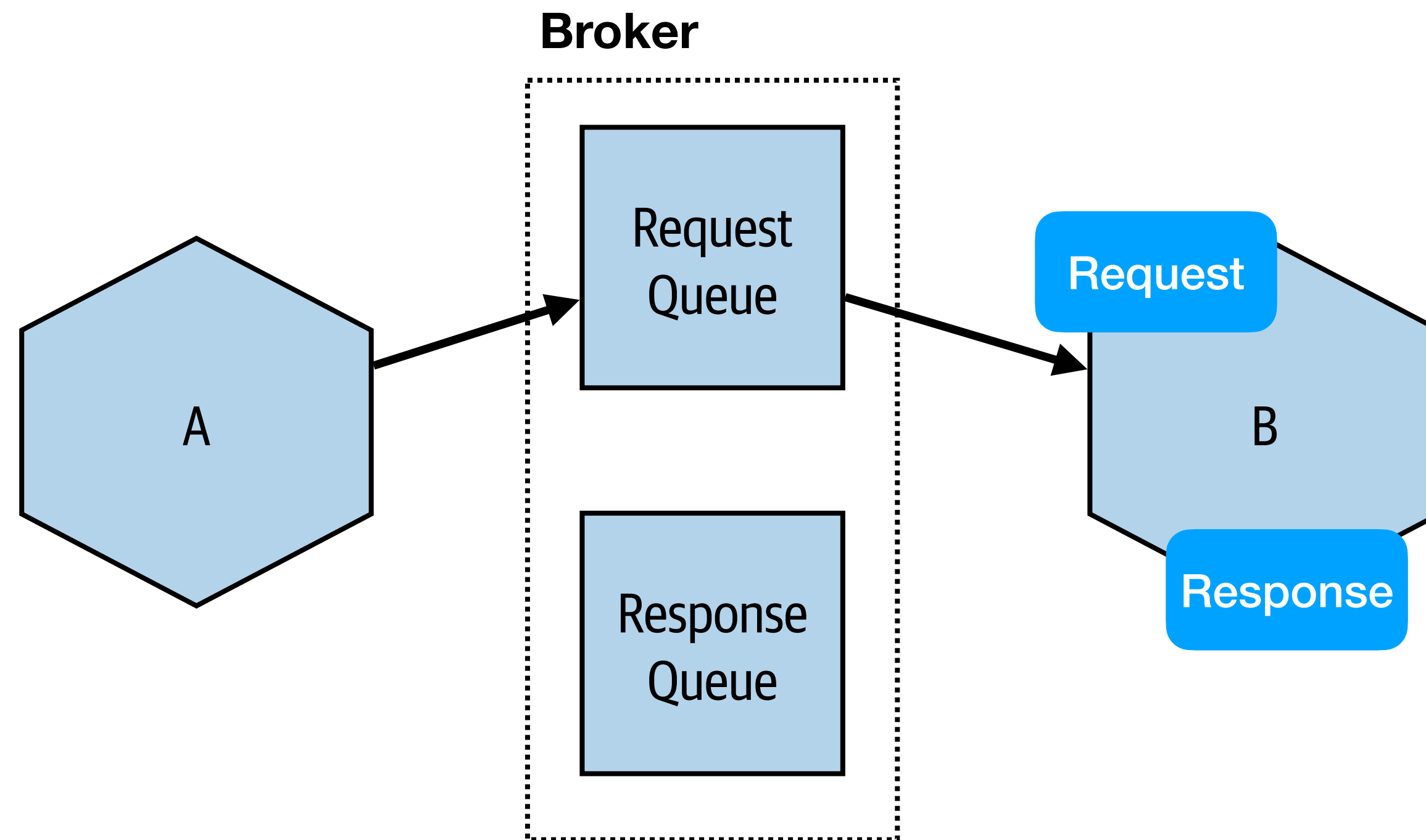
BROKER-BASED REQUEST-RESPONSE



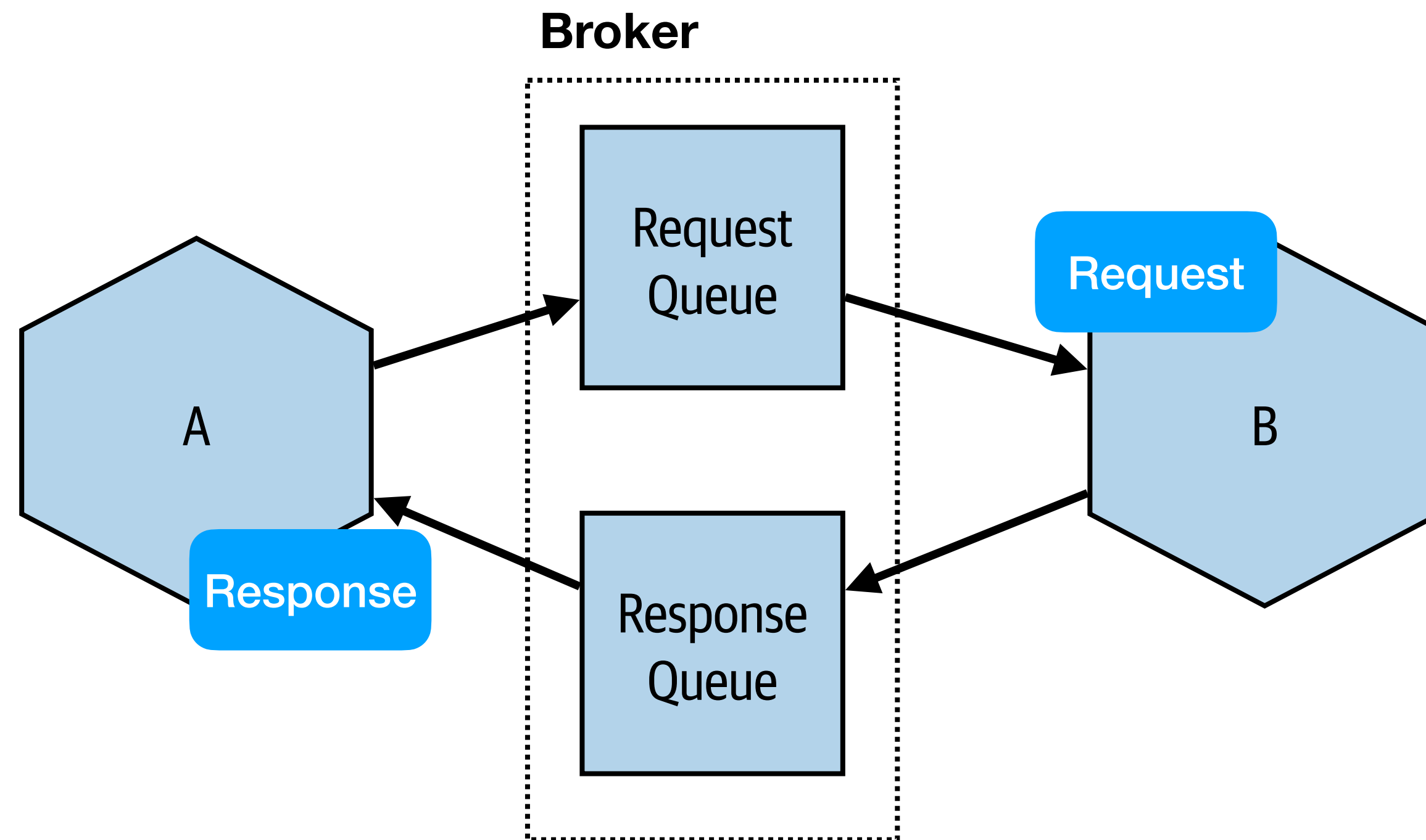
BROKER-BASED REQUEST-RESPONSE

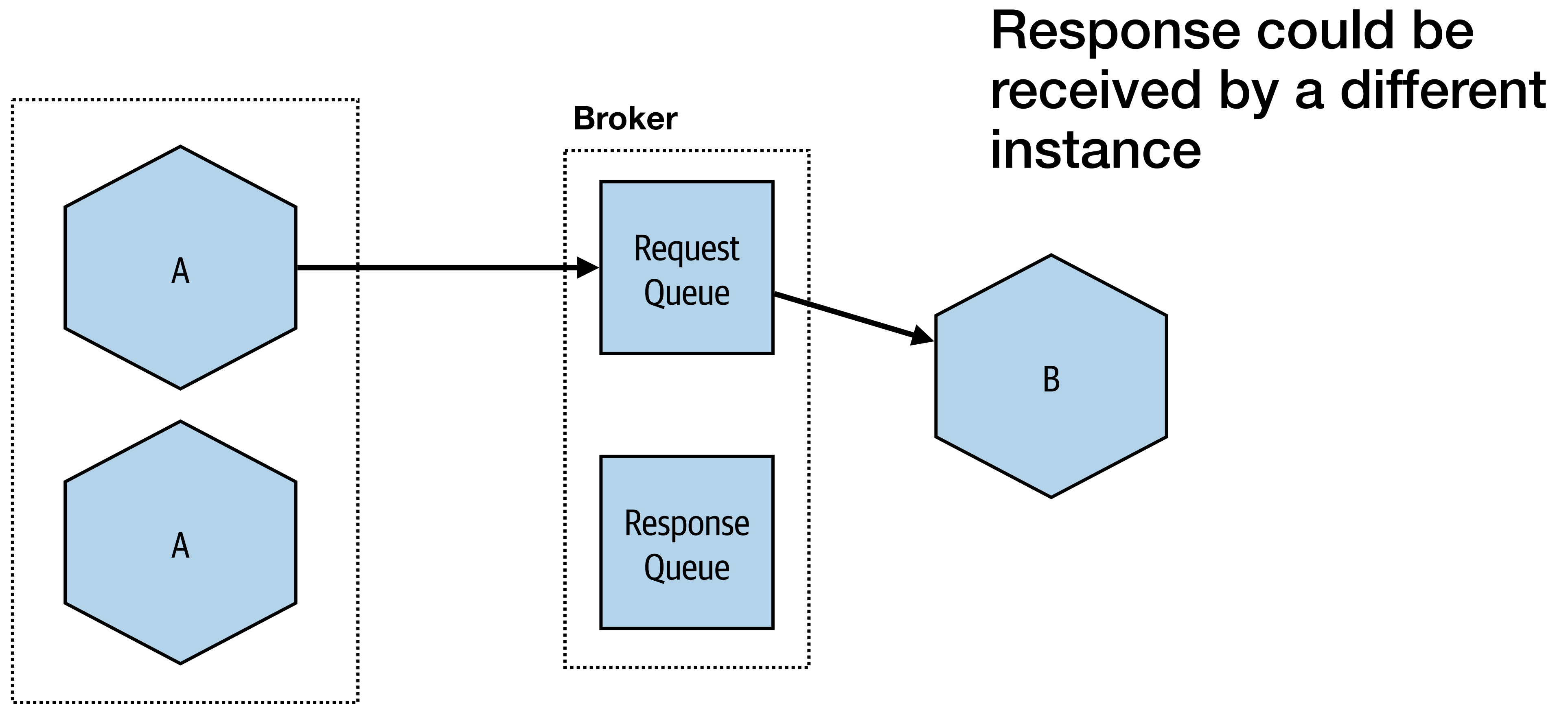


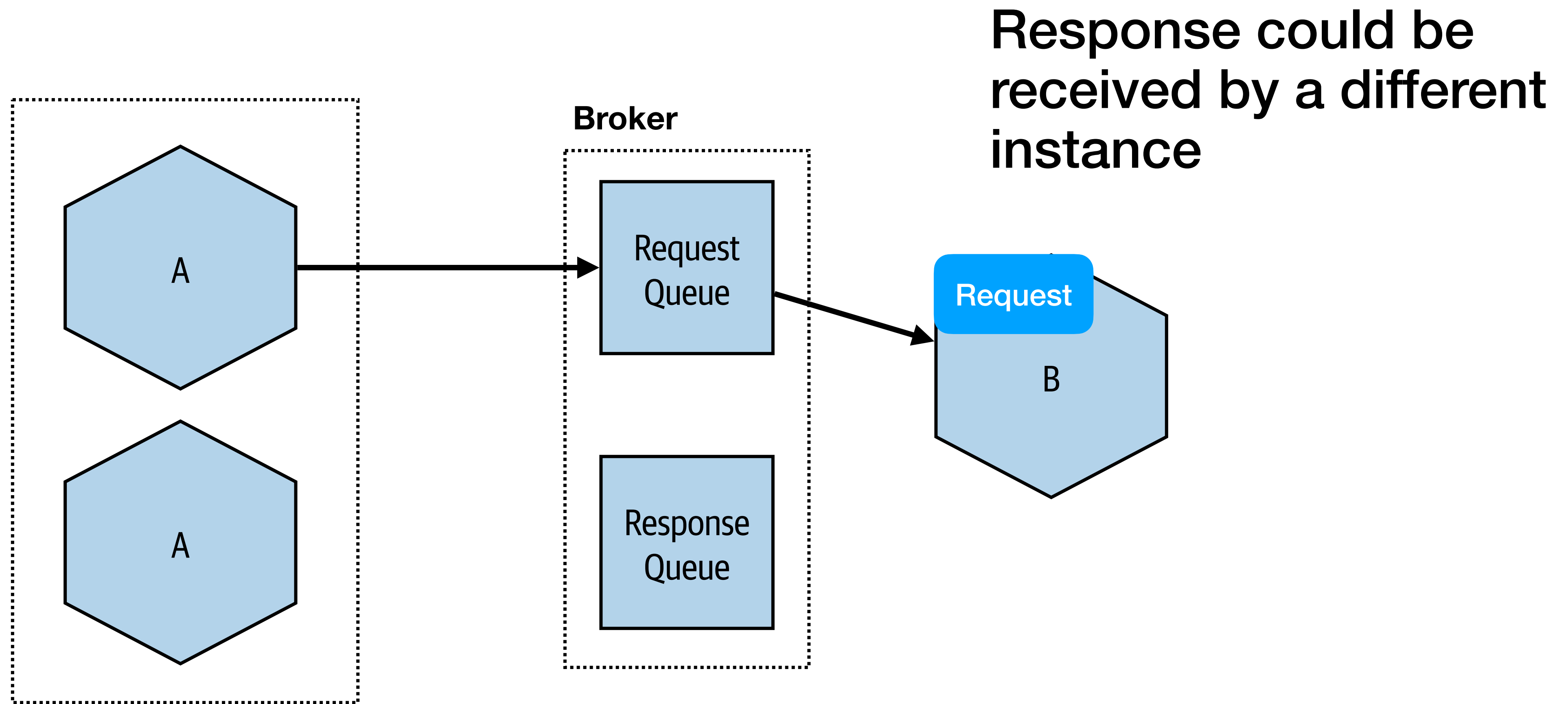
BROKER-BASED REQUEST-RESPONSE

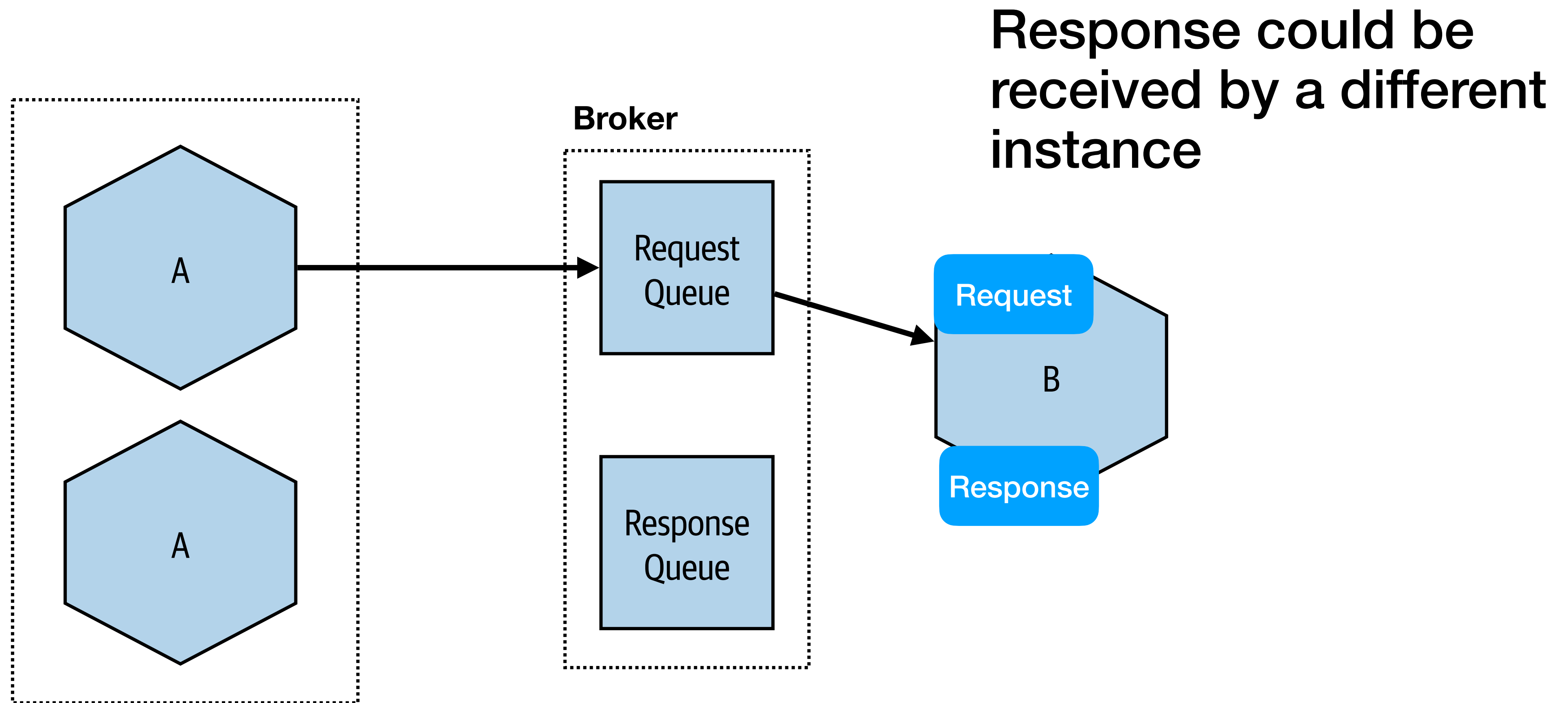


BROKER-BASED REQUEST-RESPONSE

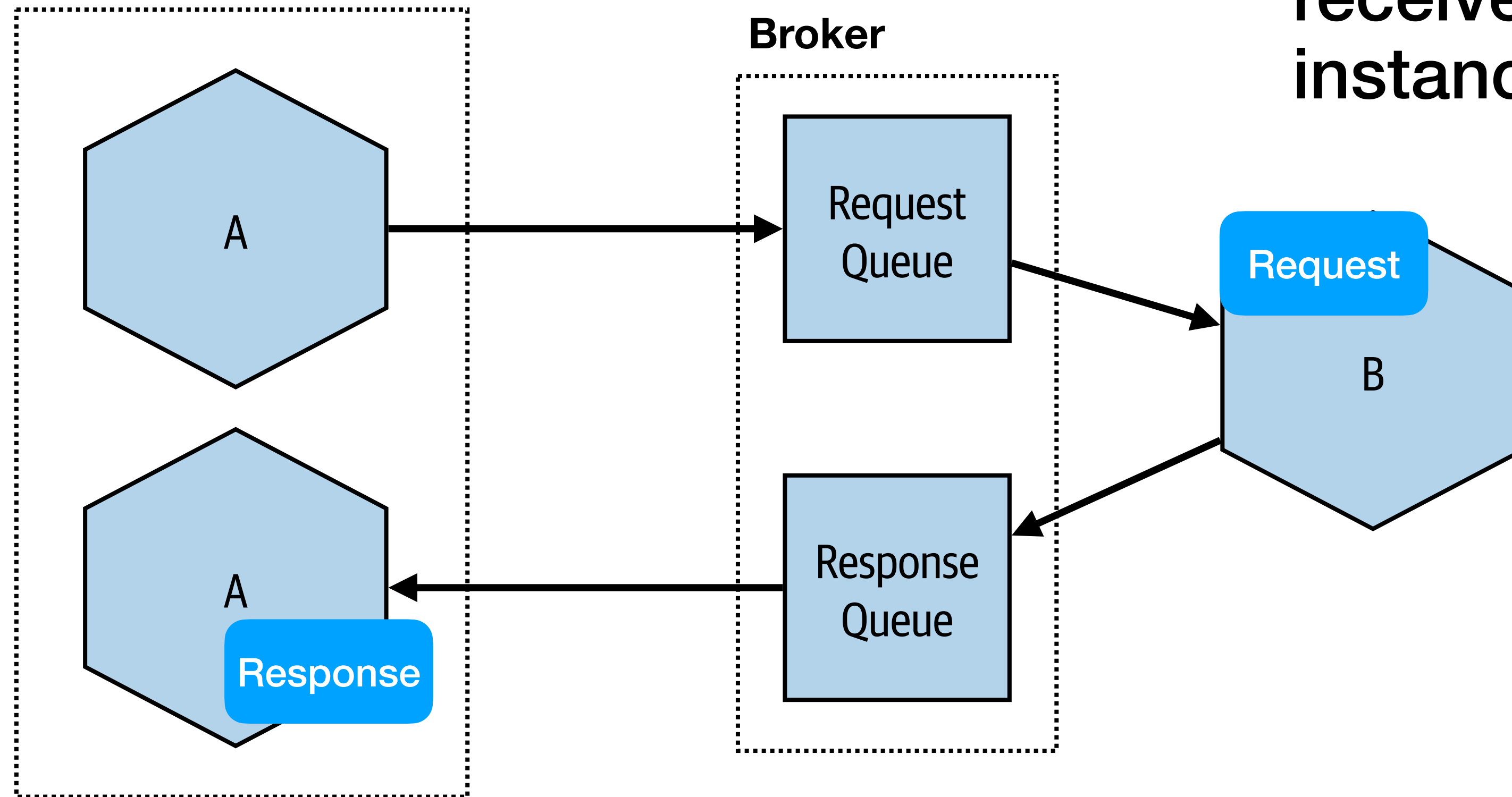




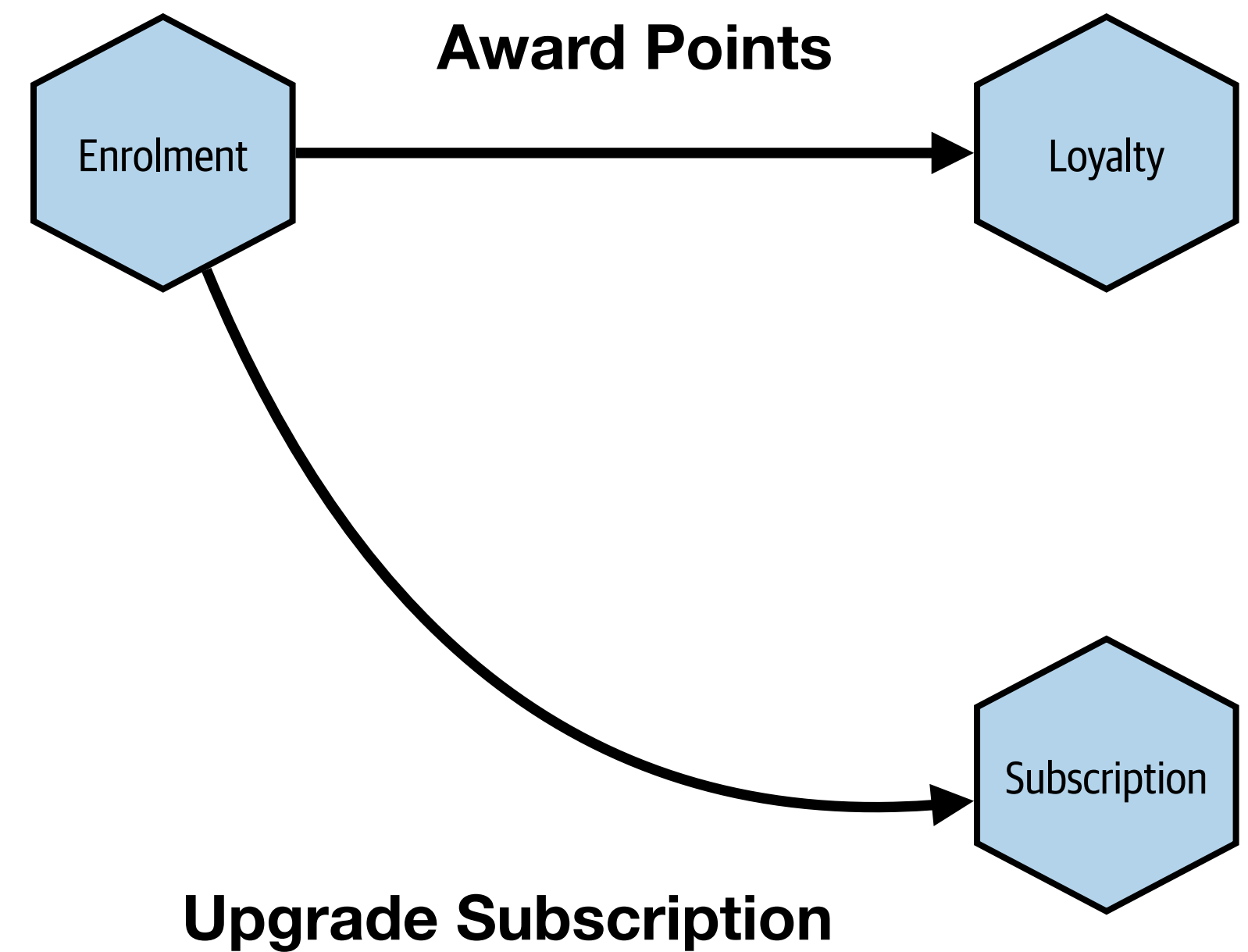




Response could be received by a different instance

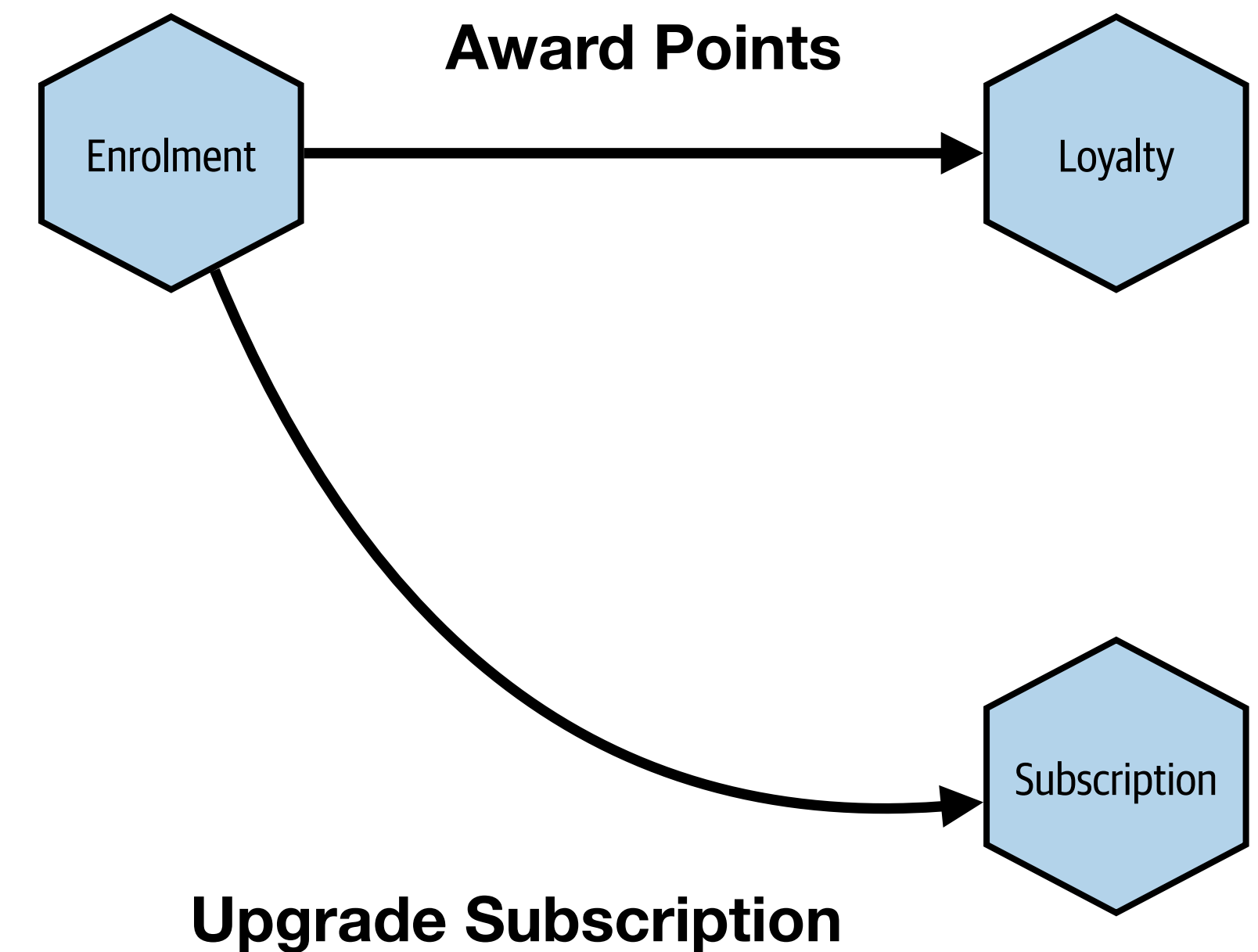


```
fut1 = future(loyaltyService.awardPoints(custID, 200))  
fut2 = future(subsService.upgradeSub(custID, PLATINUM))
```



Here, the assumption is that the response comes back to the same instance

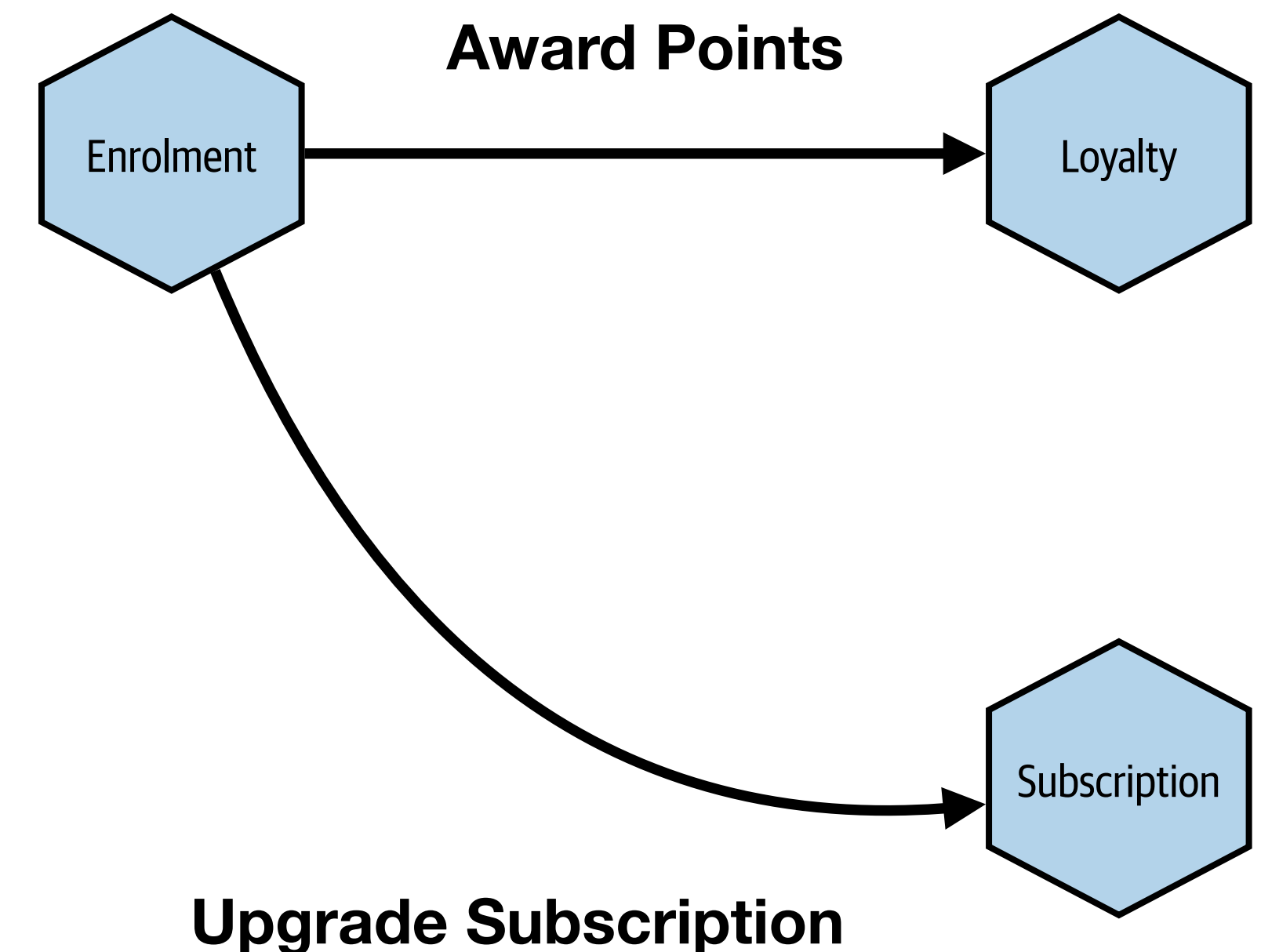
```
fut1 = future(loyaltyService.awardPoints(custID, 200))  
fut2 = future(subsService.upgradeSub(custID, PLATINUM))
```



Here, the assumption is that the response comes back to the same instance

```
fut1 = future(loyaltyService.awardPoints(custID, 200))  
fut2 = future(subsService.upgradeSub(custID, PLATINUM))
```

So we may need to deal with what happens if the consumer dies part way through



It's arguable that the intermediary-based communication encourages stateless processing

COMPETING DEFINITIONS?

COMPETING DEFINITIONS?

Immediacy

COMPETING DEFINITIONS?

Immediacy

Temporal Coupling

COMPETING DEFINITIONS?

Immediacy

Temporal Coupling

Non-blocking

COMPETING DEFINITIONS?

Immediacy

Temporal Coupling

Non-blocking

Intermediary

**Will the real asynchronous
communication please stand up?**

Does this matter?

Words have different meanings

**What words in the English language has
the most dictionary meanings?**

Set

Set

Run

Set

430 meanings

Run

<https://www.guinnessworldrecords.com/world-records/english-word-with-the-most-meanings>

Set

430 meanings

Run

645 meanings

<https://www.guinnessworldrecords.com/world-records/english-word-with-the-most-meanings>

<https://www.nytimes.com/2011/05/29/opinion/29winchester.html>

So we're used to this, right?

**We derive personal understanding
of a word via context**

“The meaning of a word becomes more narrow as we add other words around it”

- Ian Cooper (heavily paraphrased)

run

The issue got worse
the moment the
program was **run**

run

The economy crashed
due to a **run** on the
banks

So, does this matter?

**But we throw around terms like
“asynchronous” without enough
context around the term to derive
shared meaning**

🔗 Asynchronous

The Oxford Dictionary defines asynchronous as “not existing or occurring at the same time”. In the context of this manifesto we mean that the processing of a request occurs at an arbitrary point in time, sometime after it has been transmitted from client to service. The client cannot directly observe, or synchronize with, the execution that occurs within the service. This is the antonym of synchronous processing which implies that the client only resumes its own execution once the service has processed the request.



<https://www.reactivemanifesto.org/glossary#Asynchronous>

asynchronous, *adj.*

Text size: A A

View as: [Outline](#) | [Full entry](#)

Quotations: [Show all](#) | [Hide all](#) [Keywords: On](#) | [Off](#)

Pronunciation: [?] Brit.  /eɪˈsɪŋkrənəs/, U.S.  /eɪˈsɪŋkrənəs/

Frequency (in current use): ●●●●●●●●

Origin: Formed within English, by derivation. **Etymons:** *A-* *prefix*⁶, *SYNCHRONOUS adj.*

Etymology: < *A-* *prefix*⁶ + *SYNCHRONOUS adj.*

With sense 2 compare *SYNCHRONOUS adj.* 2b.

1. Not synchronous; not existing or occurring at the same time, not coinciding in time.

[Thesaurus »](#)

Originally only in medical contexts.

- 1737 tr. F. Nicholls in *Med. Ess. & Observ.* (ed. 2) III. 396 The ascending and descending Aorta are dilated and contracted at different Times, or have asynchronous Motions.
- 1768 *Gentleman's Mag.* May 224/2 The blood enters into these arteries at a time asynchronous to that in which it enters into the other arteries of the body.
- 1845 *Med. Times* 12 244/1 The pupils of the eyes are sluggish, and there is usually asynchronous action present.
- 1877 W. R. GOWERS in J. R. Reynolds *Syst. Med.* IV. 748 Reduplication has been noticed in some cases, and may be due to the asynchronous contraction of the two ventricles.
- 1918 *Mem. Brooklyn Bot. Garden* 1917 1 320 The relative production or not in the wild state of hybrid *Sarracenia*s is almost wholly determined by the synchronous or asynchronous relation of the flowers.
- 1968 *Brain* 91 306 Quick, asynchronous, random jerks of the muscles of the trunk and extremities were also evident.
- 1987 S. BRAND *Media Lab* I. iii. 41 A primary characteristic of interactive television is that it's asynchronous—stuff is broadcast at one time, viewed at another.
- 2001 *Libr. Assoc. Rec.* May 295/2 Asynchronous participation through the discussion board is effective for learner reflection, giving students time to think.

[\(Hide quotations\)](#)

asynchronous, *adj.*

Text size: [A](#) [A](#)

View as: [Outline](#) | [Full entry](#)

Quotations: [Show all](#) | [Hide all](#) Keywords: [On](#) | [Off](#)

Pronunciation: [?] Brit. [▶](#) /eɪˈsɪŋkrənəs/, U.S. [▶](#) /eɪˈsɪŋkrənəs/

Frequency (in current use): ●●●●●●●●

Origin: Formed within English, by derivation. **Etymons:** *A-* [prefix](#)⁶, *SYNCHRONOUS* *adj.*

Etymology: < *A-* [prefix](#)⁶ + *SYNCHRONOUS* *adj.*

With sense 2 compare *SYNCHRONOUS* *adj.* 2b.

1. Not synchronous: not existing or occurring at the same time, not coinciding in time.

[Thesaurus »](#)

Originally only in medical contexts.

- 1737 tr. F. Nicholls in *Med. Ess. & Observ.* (ed. 2) III. 396 The ascending and descending Aorta are dilated and contracted at different Times, or have asynchronous Motions.
- 1768 *Gentleman's Mag.* May 224/2 The blood enters into these arteries at a time asynchronous to that in which it enters into the other arteries of the body.
- 1845 *Med. Times* 12 244/1 The pupils of the eyes are sluggish, and there is usually asynchronous action present.
- 1877 W. R. GOWERS in J. R. Reynolds *Syst. Med.* IV. 748 Reduplication has been noticed in some cases, and may be due to the asynchronous contraction of the two ventricles.
- 1918 *Mem. Brooklyn Bot. Garden* 1917 1 320 The relative production or not in the wild state of hybrid *Sarracenias* is almost wholly determined by the synchronous or asynchronous relation of the flowers.
- 1968 *Brain* 91 306 Quick, asynchronous, random jerks of the muscles of the trunk and extremities were also evident.
- 1987 S. BRAND *Media Lab* I. iii. 41 A primary characteristic of interactive television is that it's asynchronous—stuff is broadcast at one time, viewed at another.
- 2001 *Libr. Assoc. Rec.* May 295/2 Asynchronous participation through the discussion board is effective for learner reflection, giving students time to think.

[\(Hide quotations\)](#)

“Originally only in medical contexts”
- *Oxford English Dictionary*

2. *Electrical Engineering.* Of an alternating-current motor, generator, etc.: able to operate at speeds other than the speed determined by the frequency of the current supply (the synchronous speed); *spec.* designating an induction motor.

Categories »

- 1893 *Electr. Engineer* 22 Sept. 280/1 This plant was practically designed before the advent of a really reliable asynchronous and self-starting alternate-current motor.
- 1925 H. C. BOOTH tr. F. Auerbach *Mod. Magnetism* (U.K. ed.) vii. 176 The most important practical distinction (apart from that between direct, alternating and polyphase current motors) is that between synchronous and asynchronous motors.
- 1991 *Impact of Sci. on Society* (UNESCO) No. 162. 152 The three-phase asynchronous (induction) motor has no commutator.
- 2007 *Observer* 11 Nov. (Review section) 13/4 Railways have changed since Barlow's, let alone Betjeman's, day. It is a world of asynchronous motors, catenaries,...and push-jacks.

(Hide quotations)

3. *Computing.*

a. Of a computer or part of one: not operating in accordance with clock signals; (of operations) beginning when the previous operation finishes, rather than occurring at regular intervals of time.

Categories »

- 1953 *Math. Tables & Other Aids Computation* 7 112 The memory control reports back to the main control when it accepts the main control intervention and when it completes any of the actions described in (a)–(d). In this manner the synchronous memory operation is safely incorporated into the asynchronous arithmetic process.
- 1967 H. JACOBOWITZ *Electronic Computers Made Simple* xii. 261 Unless the computer is asynchronous..all its operations are controlled by a series of equally spaced timing pulses.
- 1979 M. M. MANO *Digital Logic & Computer Design* vi. 210 Asynchronous flip-flops..require an input trigger defined by a change of signal level... Clocked flip-flops are triggered by pulses.
- 2002 *Sci. Amer.* Aug. 48/2 Asynchronous parts of otherwise synchronous systems are also beginning to appear; the UltraSPARC IIIi processor recently introduced by Sun includes some asynchronous circuits.

(Hide quotations)

b. Designating data transmission in which packets of data are sent at irregular intervals, with the start and end of each packet being marked by specific signals; involving such transmission.

Categories »

- 1966 *U.S. Patent 3,249,924* 9 An asynchronous data processing system for transferring data between the main memory of a central processor and a plurality of input-output devices.
- 1976 J. MARTIN *Telecommunications & Computer* (ed. 2) xviii. 360 (*table*) Synchronous data from 2.4 to 1344 Kbps. Asynchronous data up to 1.2 Kbps.
- 1996 *Sunday Tel.* 4 Feb. (Appointments section) 8/2 (*advt.*) A strong datacomms background should include knowledge of asynchronous and synchronous protocols such as RS232.
- 2004 S. MAKOFSKY *Pocket PC Network Programming* vi. 204 PPP supports error correction and is capable of handling either synchronous or asynchronous communications.

(Hide quotations)

2. Electrical Engineering. Of an alternating-current motor, generator, etc.: able to operate at speeds other than the speed determined by the frequency of the current supply (the synchronous speed); *spec.* designating an induction motor.

Categories »

- 1893 *Electr. Engineer* 22 Sept. 280/1 This plant was practically designed before the advent of a really reliable asynchronous and self-starting alternate-current motor.
- 1925 H. C. BOOTH tr. F. Auerbach *Mod. Magnetism* (U.K. ed.) vii. 176 The most important practical distinction (apart from that between direct, alternating and polyphase current motors) is that between synchronous and asynchronous motors.
- 1991 *Impact of Sci. on Society* (UNESCO) No. 162. 152 The three-phase asynchronous (induction) motor has no commutator.
- 2007 *Observer* 11 Nov. (Review section) 13/4 Railways have changed since Barlow's, let alone Betjeman's, day. It is a world of asynchronous motors, catenaries,...and push-jacks.

(Hide quotations)

3. Computing.

a. Of a computer or part of one: not operating in accordance with clock signals; (of operations) beginning when the previous operation finishes, rather than occurring at regular intervals of time.

Categories »

- 1953 *Math. Tables & Other Aids Computation* 7 112 The memory control reports back to the main control when it accepts the main control intervention and when it completes any of the actions described in (a)–(d). In this manner the synchronous memory operation is safely incorporated into the asynchronous arithmetic process.
- 1967 H. JACOBOWITZ *Electronic Computers Made Simple* xii. 261 Unless the computer is asynchronous..all its operations are controlled by a series of equally spaced timing pulses.
- 1979 M. M. MANO *Digital Logic & Computer Design* vi. 210 Asynchronous flip-flops..require an input trigger defined by a change of signal level... Clocked flip-flops are triggered by pulses.
- 2002 *Sci. Amer.* Aug. 48/2 Asynchronous parts of otherwise synchronous systems are also beginning to appear; the UltraSPARC IIIi processor recently introduced by Sun includes some asynchronous circuits.

(Hide quotations)

b. Designating data transmission in which packets of data are sent at irregular intervals, with the start and end of each packet being marked by specific signals; involving such transmission.

Categories »

- 1966 *U.S. Patent 3,249,924* 9 An asynchronous data processing system for transferring data between the main memory of a central processor and a plurality of input-output devices.
- 1976 J. MARTIN *Telecommunications & Computer* (ed. 2) xviii. 360 (table) Synchronous data from 2.4 to 1344 Kbps. Asynchronous data up to 1.2 Kbps.
- 1996 *Sunday Tel.* 4 Feb. (Appointments section) 8/2 (advt.) A strong datacomms background should include knowledge of asynchronous and synchronous protocols such as RS232.
- 2004 S. MAKOFSKY *Pocket PC Network Programming* vi. 204 PPP supports error correction and is capable of handling either synchronous or asynchronous communications.

(Hide quotations)

ASYNCHRONOUS DEFINITION IN A COMPUTING CONTEXT

“Designating data transmissions in which packets of data are sent at irregular intervals, with the start and end of each packet being marked by specific signals involving such transmission”

- *Oxford English Dictionary*

Software is a type of
sociotechnical system

Sociotechnical

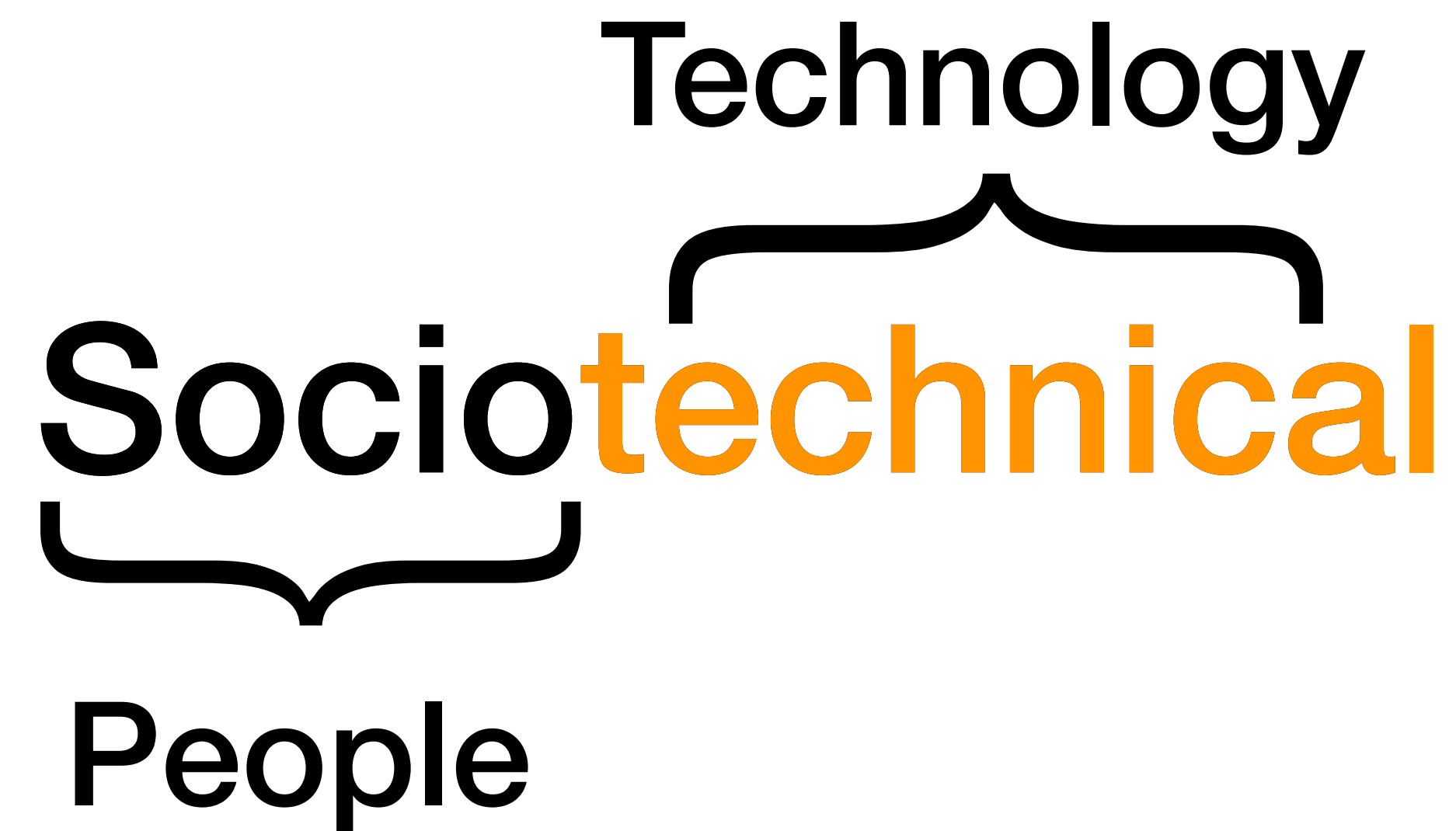
Sociotechnical

People

Technology

Socio**technical**

People

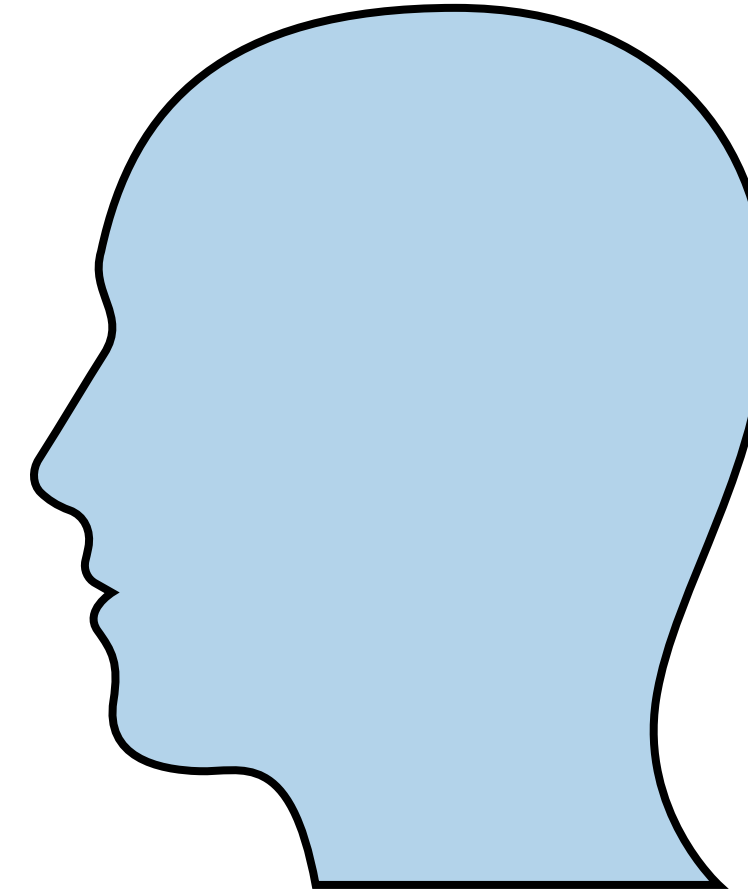
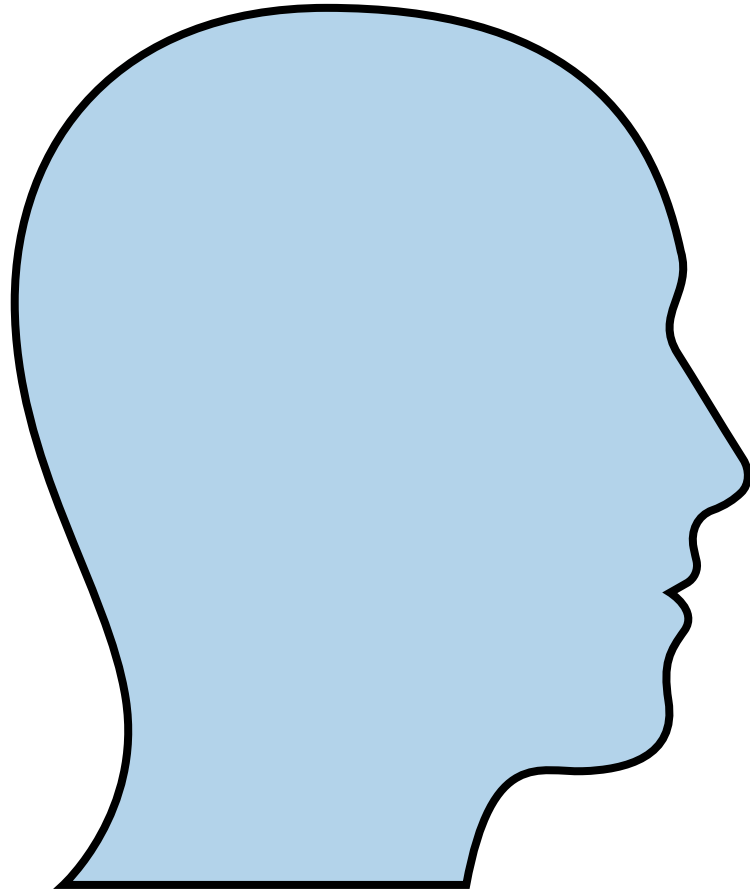
A diagram illustrating the components of a sociotechnical system. The word "Sociotechnical" is centered, with "Socio" in black and "technical" in orange. A black curly brace is positioned above "technical" and points to the word "Technology" above it. Another black curly brace is positioned below "Socio" and points to the word "People" below it.





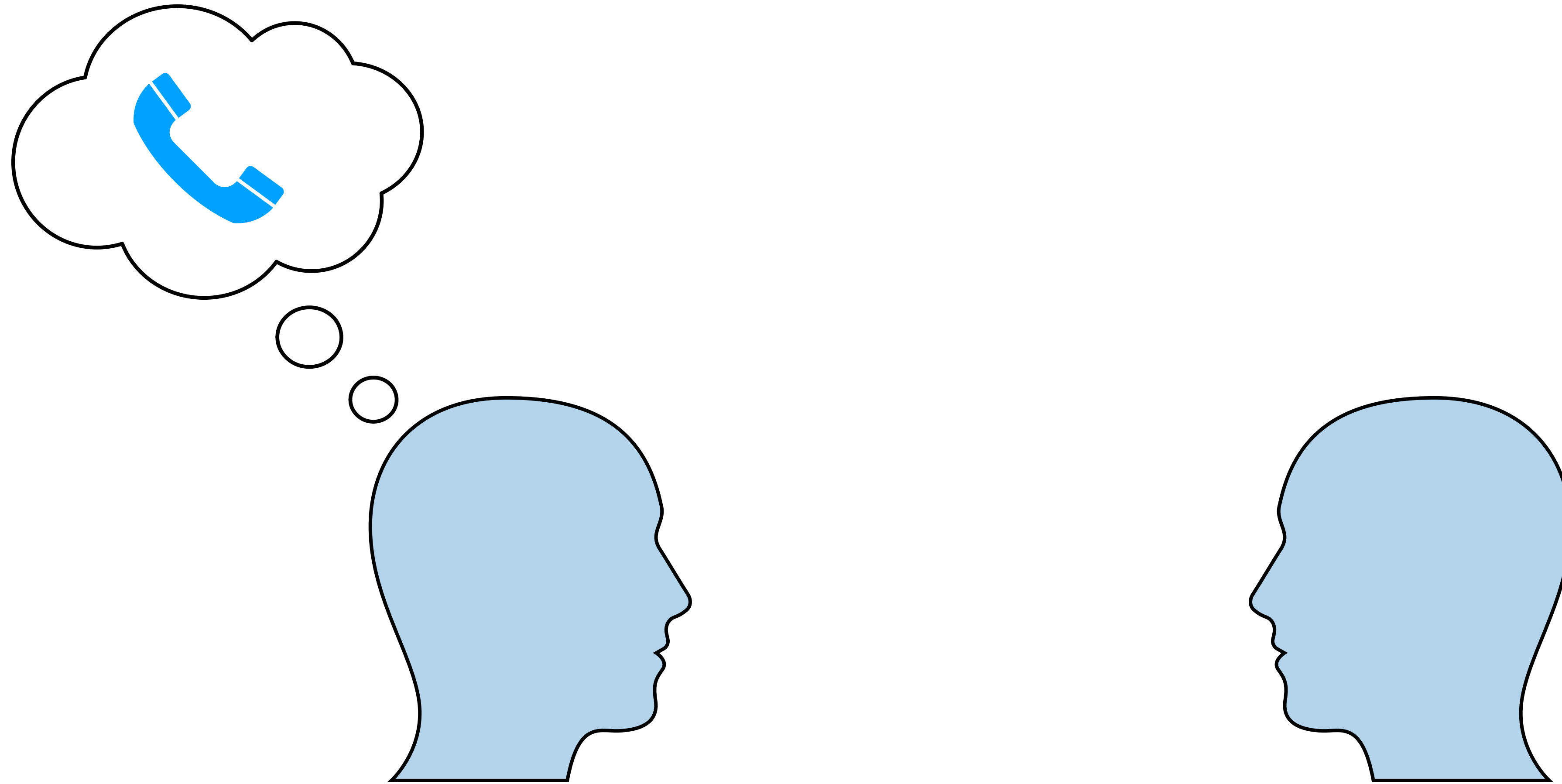
**The amount of useful software created
by individuals is vanishingly rare**

**Bringing groups of people together
to use and use technology requires
good communication**





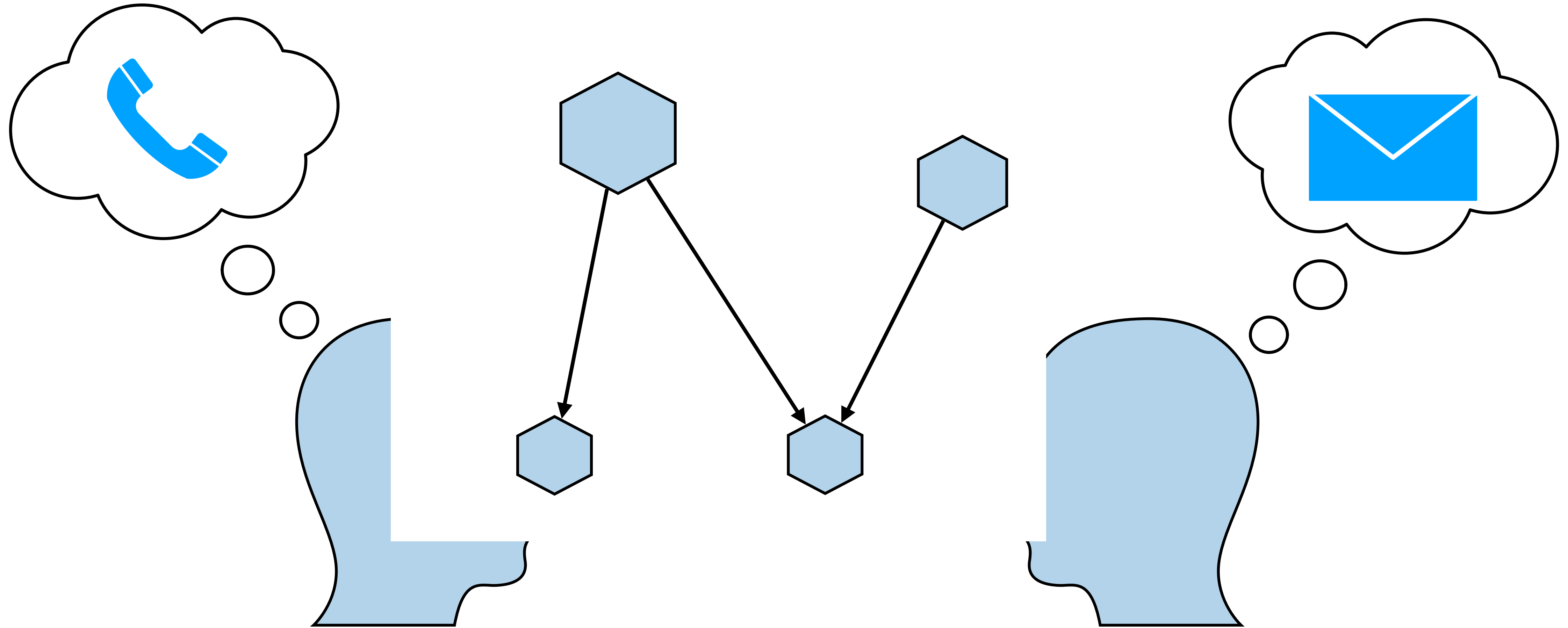
So, we're going to make this asynchronous...



So, we're going to make this asynchronous...



So, we're going to make this asynchronous...



So, we're going to make this asynchronous...



So, we're going to make this asynchronous...



**The term “asynchronous” in the
context of inter-process
communication has so many meanings
that it is effectively meaningless**

So what can we do?

**Understand what your
application needs**

**Understand what your
application needs**

**Describe how it should handle
different situations**

**What should happen when the
server is unreachable?**

**What should happen when the
server is unreachable?**

How fast should it be?

**What should happen when the
server is unreachable?**

How fast should it be?

What if the client crashes?

**Where possible, use more
explicit terms**

Clients should be non-blocking

Clients should be non-blocking

Run operations in parallel

Clients should be non-blocking

Run operations in parallel

Use broker X as an intermediary...

**But try not to make having a
common meaning be about
correcting someone else**



A still from the movie 'The Princess Bride' showing Inigo Montoya (Cary Elwes) on the left and Buttercup (Robin Wright) on the right. Inigo is looking at Buttercup with a serious expression. The background is a stone wall.

"You keep using that word. I do not think it means what you think it means."

— Inigo Montoya, *The Princess Bride*

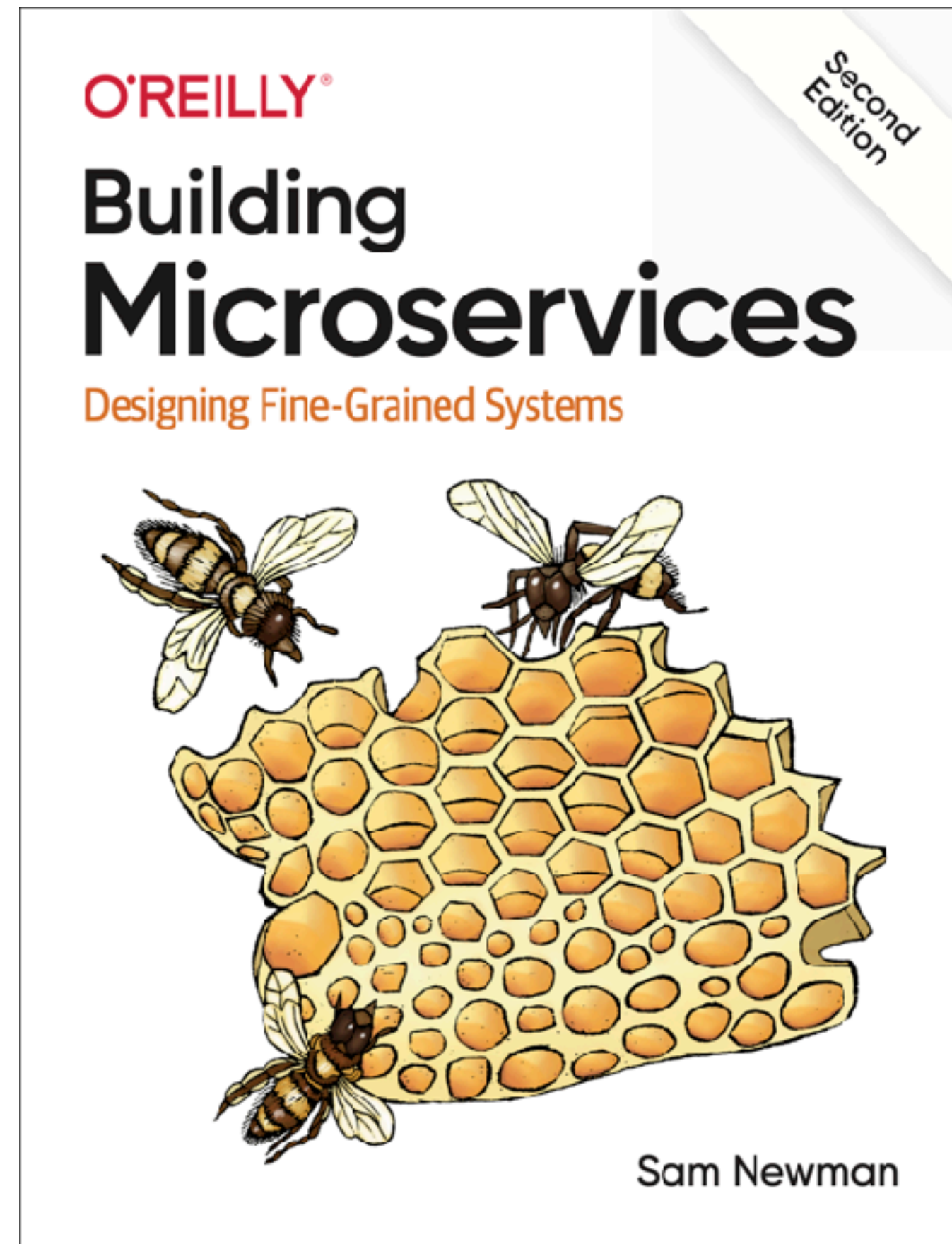
**When someone says “we should
make this asynchronous”...**

**When someone says “we should
make this asynchronous”...**

**Perhaps just say, “what does that
mean to you?”**

What you find might shock you

THANKS!



@samnewman

Sam
Newman
& Associates



Sam Newman.

Home About **Talks** Podcast Writing Contact

Talks & Workshops.

Here are a list of the talks I am currently presenting. On request, I can present different topics or even my older talks. If you want me to present these topics at your conference or company, then please contact me.

You can also see where I'll be speaking next on my [events](#) page.

What Is This Cloud Native Thing Anyway? .

45min Talk

A talk exploring what the hell Cloud Native means

→ [Find Out More](#)

The diagram illustrates the interconnected nature of modern software development practices. It features a central circle labeled 'Cloud' with arrows pointing to it from 'Agile/Lean', 'Continuous Delivery', 'Microservices', and 'DevOps'. Below 'Cloud' is 'Cloud Native'. Arrows also connect 'Agile/Lean' to 'Continuous Delivery', 'Continuous Delivery' to 'DevOps', and 'DevOps' to 'Microservices'.

Feature Branches And Toggles In A Post-GitHub World .

The thumbnail shows a person standing in front of a screen displaying code or a diagram, with a list of topics below.

Book! .

The book cover is identical to the one shown in the first image.

I have written a book called "Building Microservices", which is available now. Want to know more?

→ [Read on...](#)

Video! .

<https://samnewman.io/>

hachyderm.io/@samnewman