



**KubeCon**



**CloudNativeCon**

North America 2018

# Why data scientists love Kubernetes

**Sophie Watson • [sophie@redhat.com](mailto:sophie@redhat.com)**

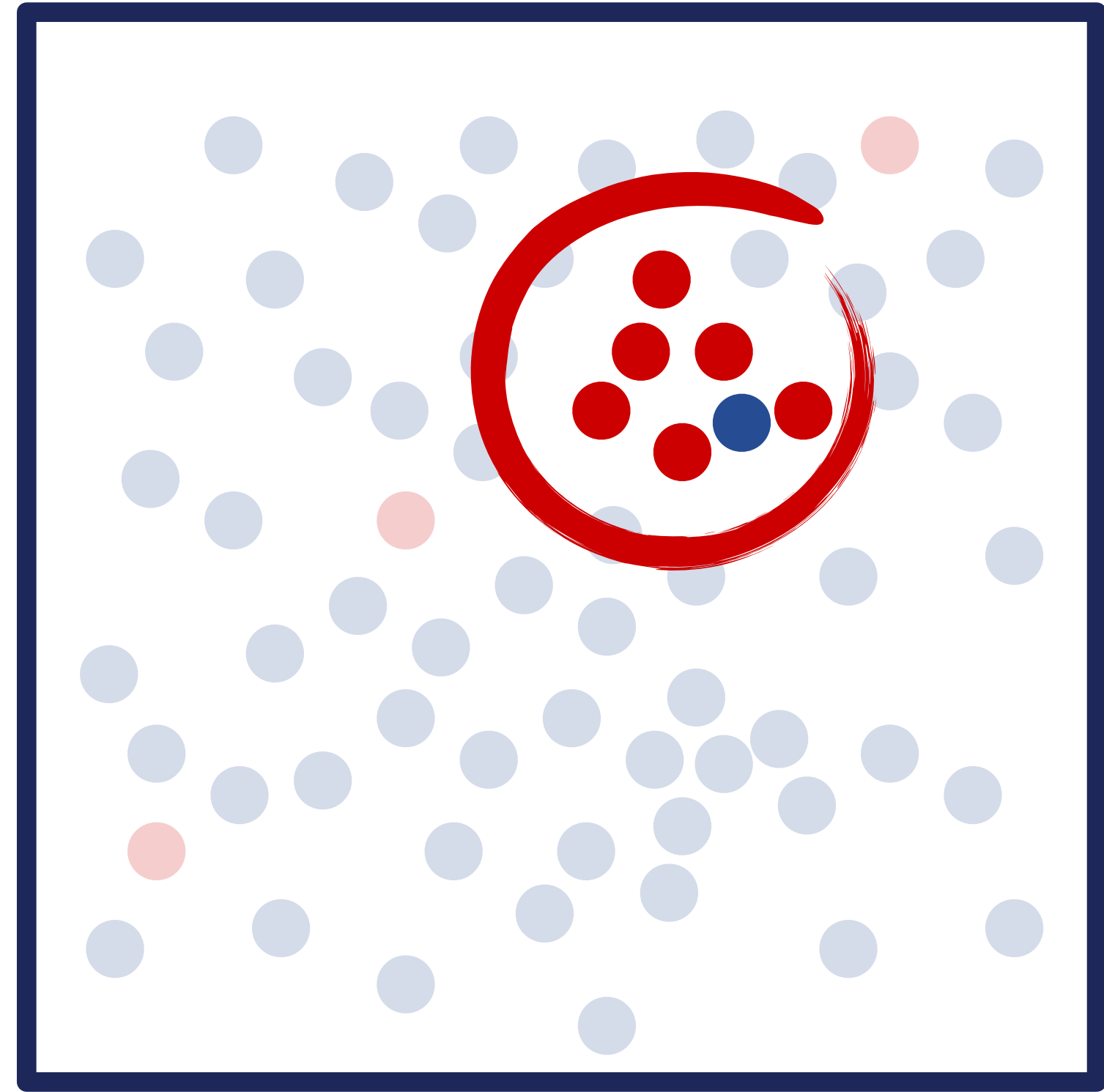
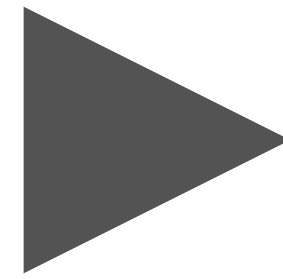
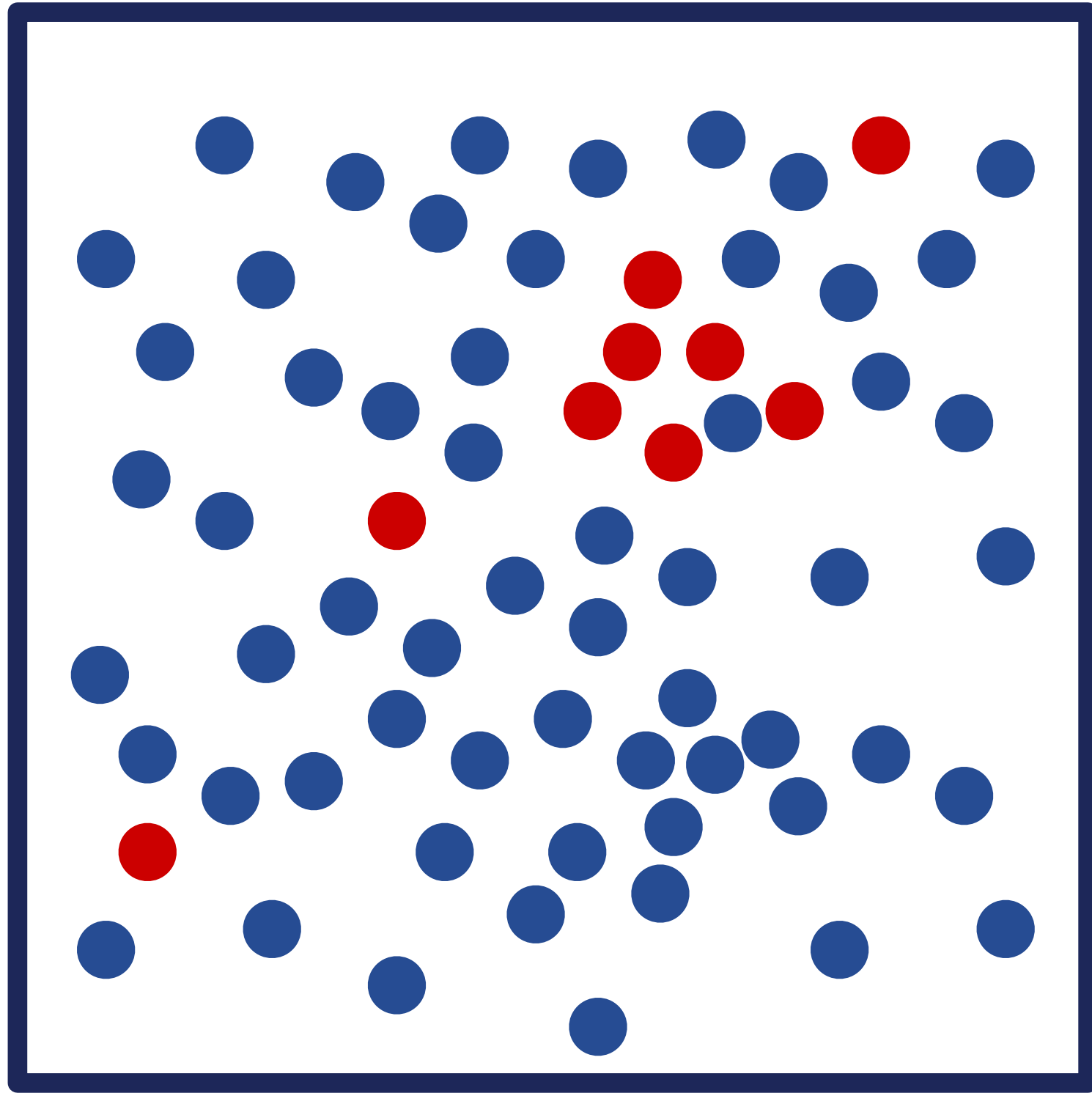
**William Benton • [willb@redhat.com](mailto:willb@redhat.com)**

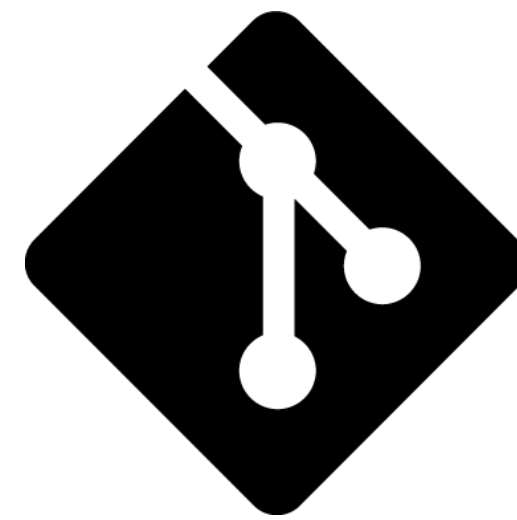
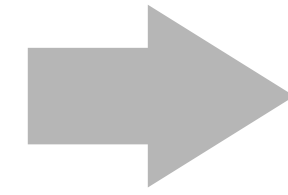


**About us**

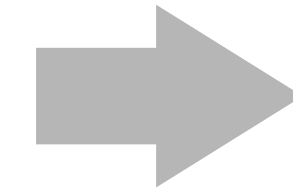
**About you**

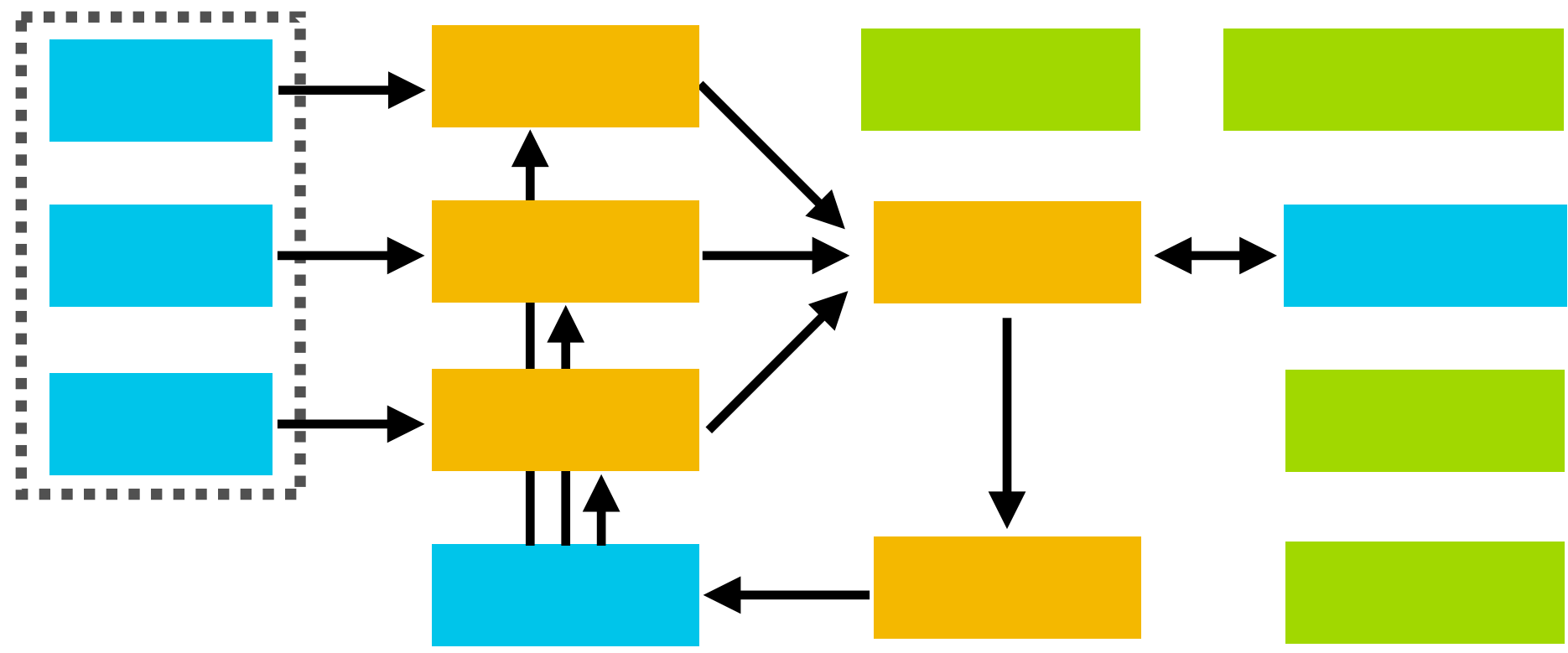
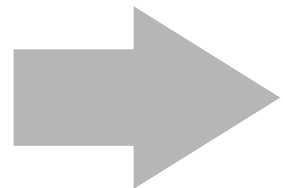
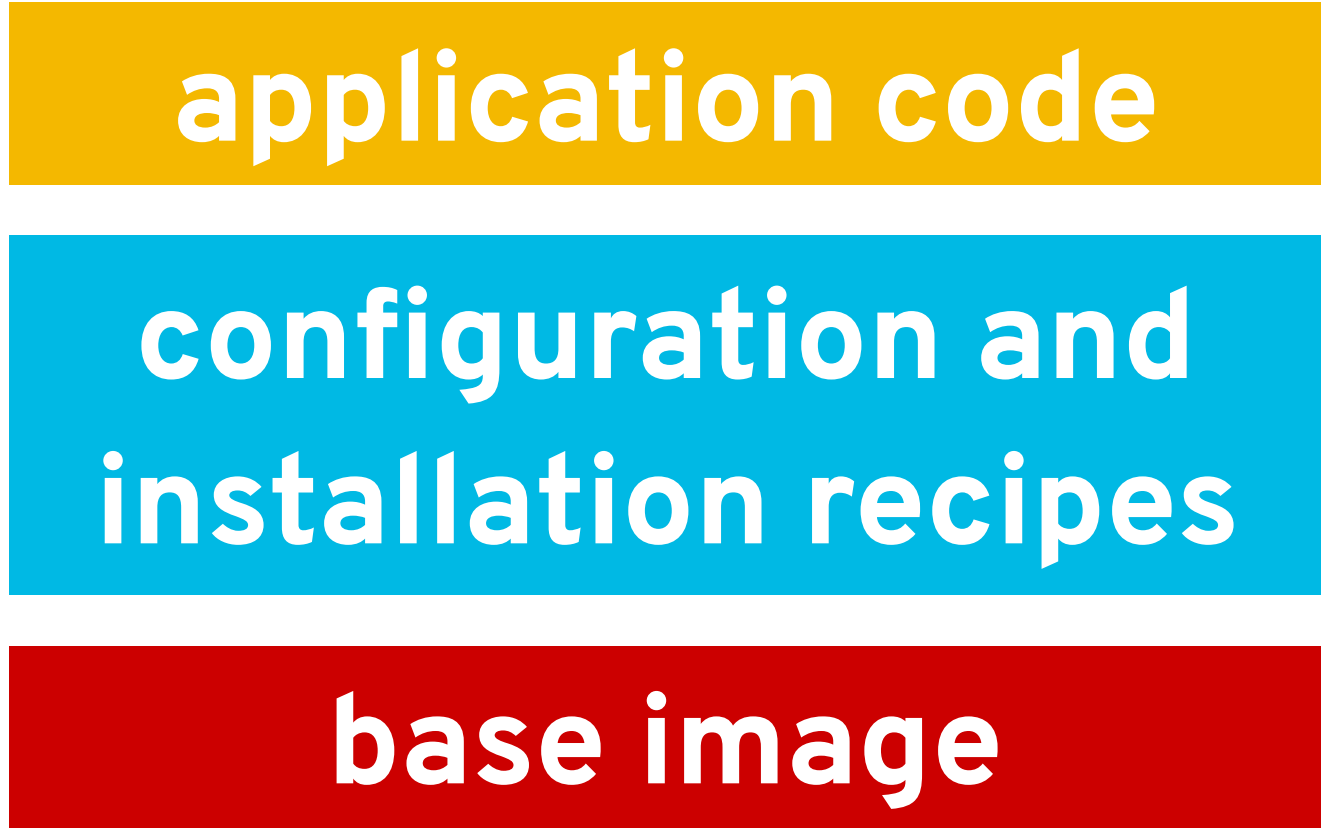
**What we'll talk about today**

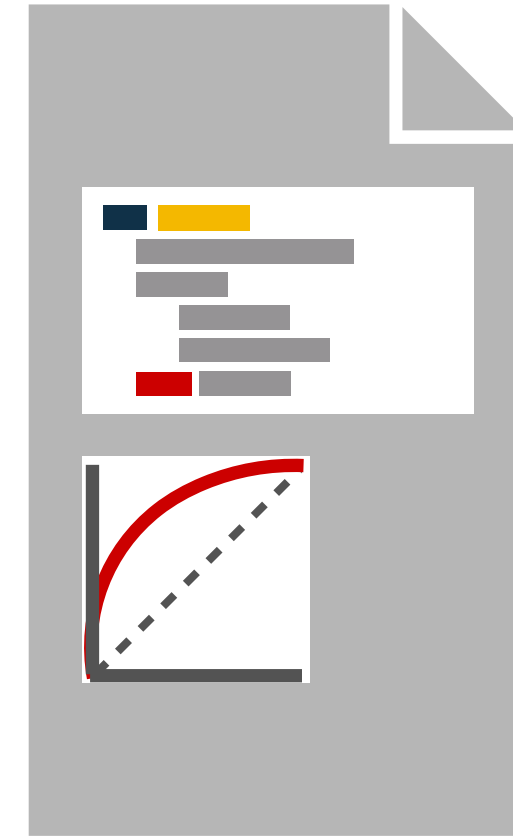
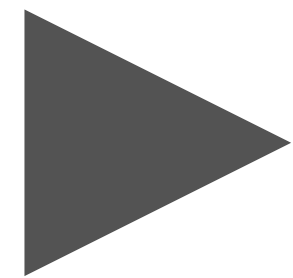
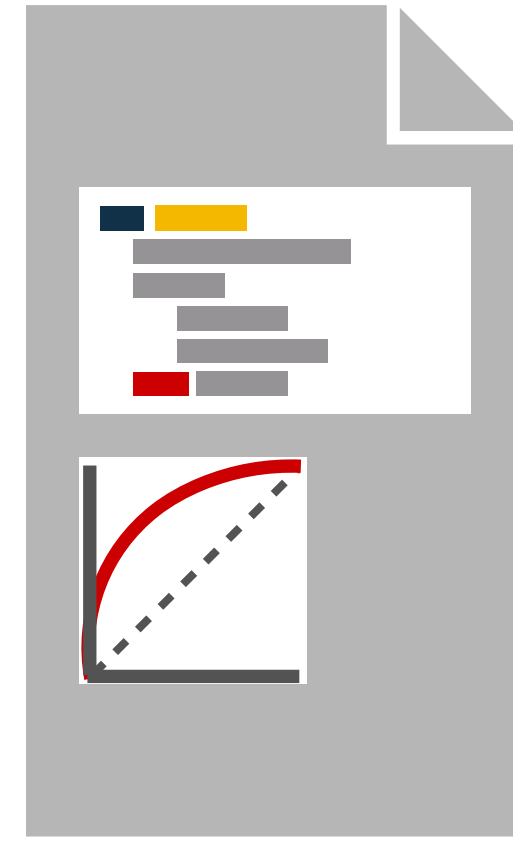
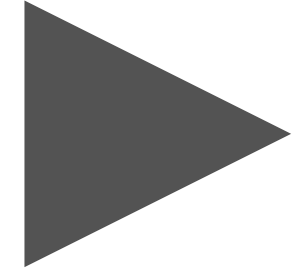
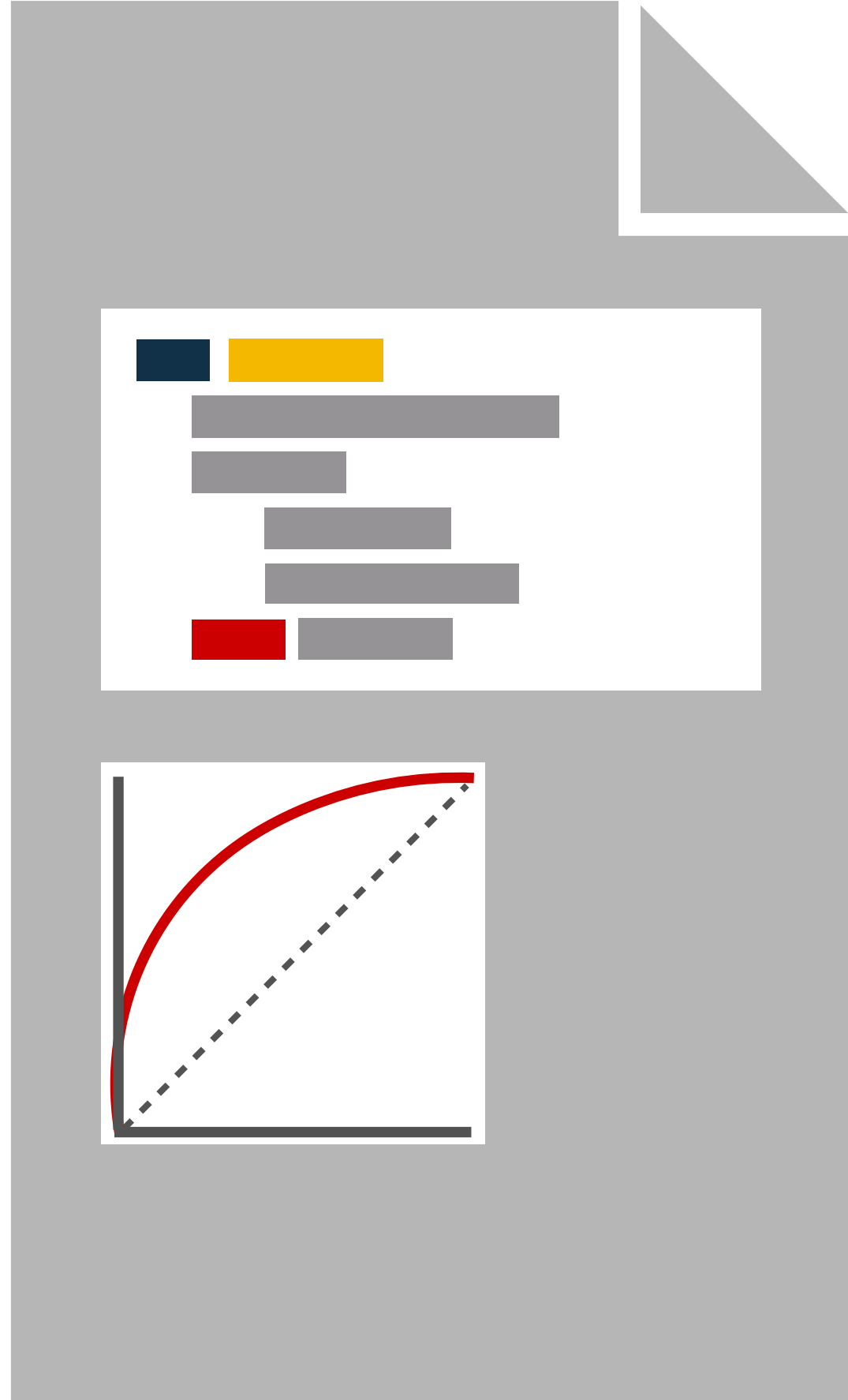




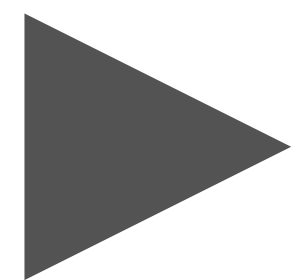
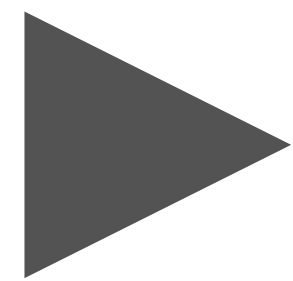
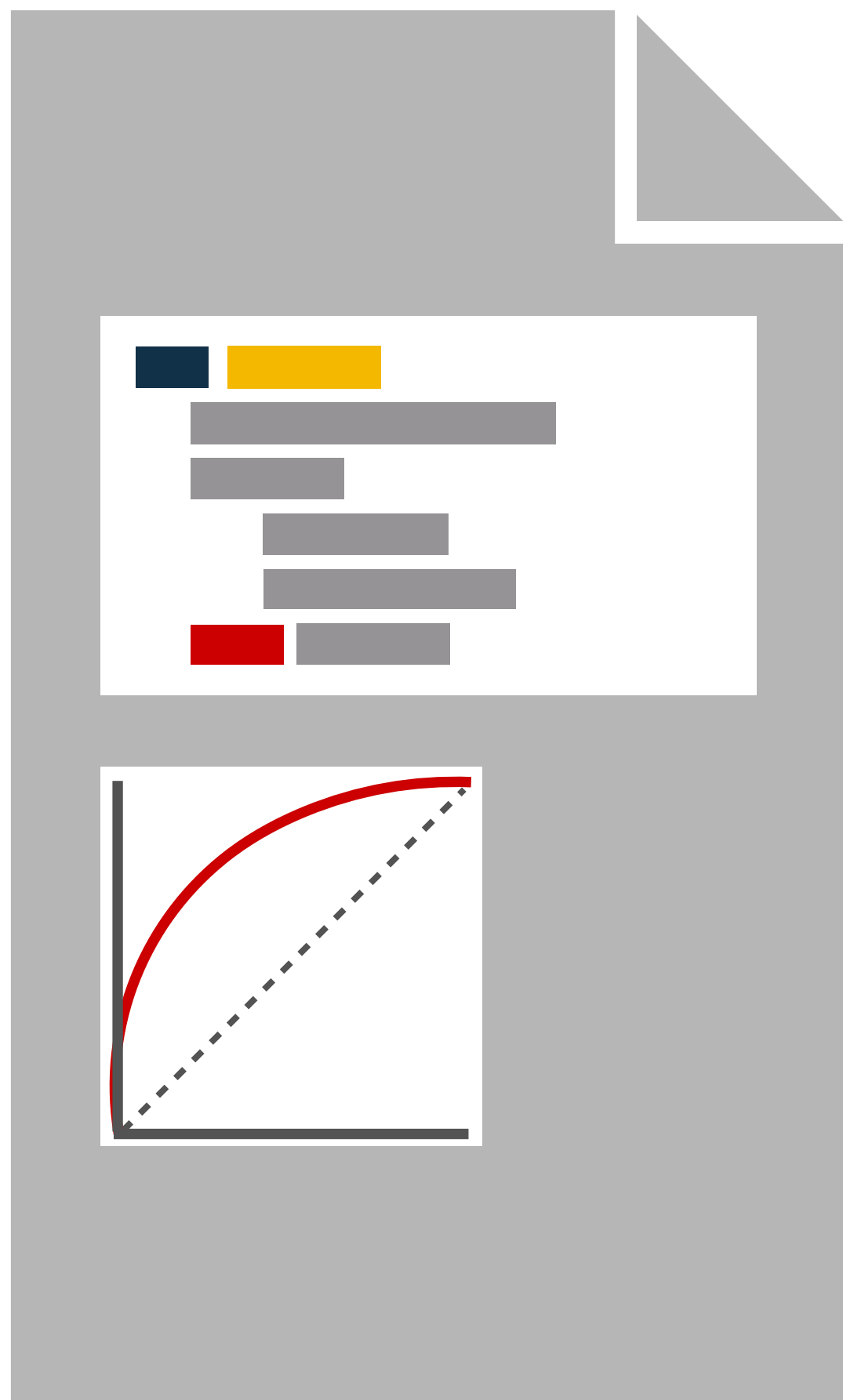
**git**











# Forecast

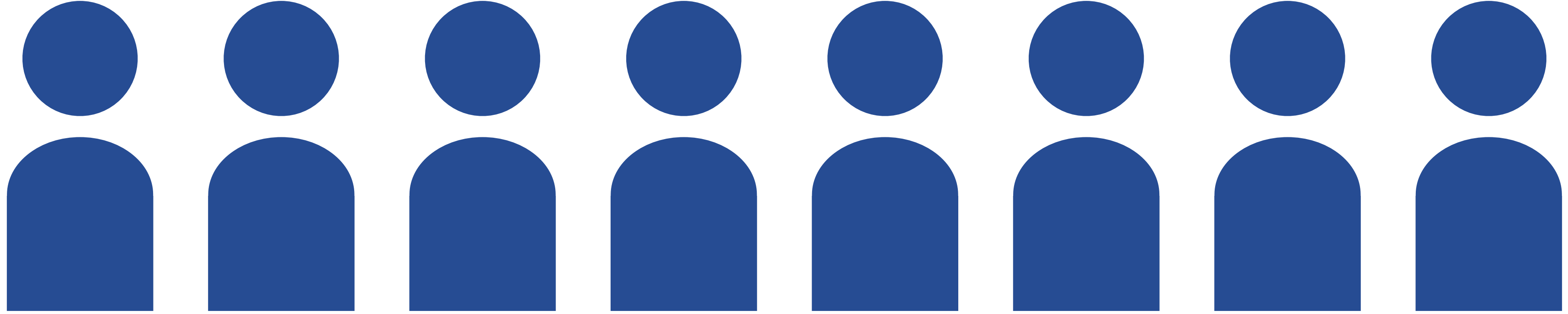
A day in the life of a data scientist

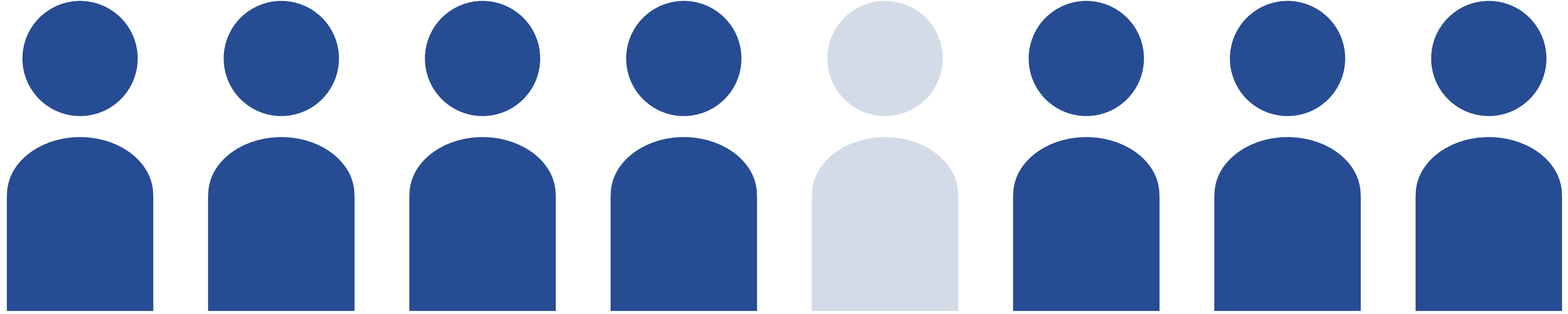
Container workflows for data science

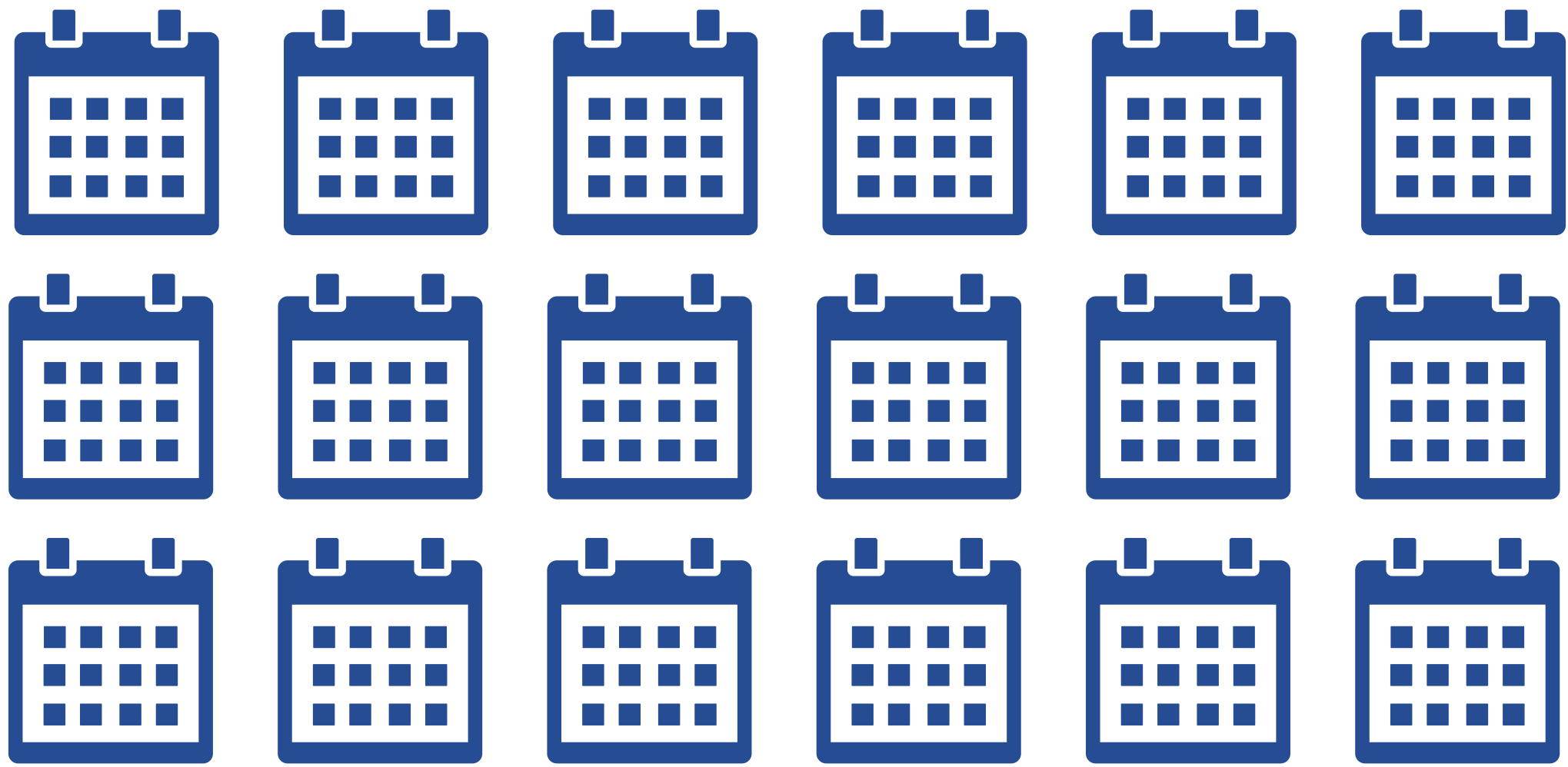
Making the power of Kubernetes accessible  
to data scientists

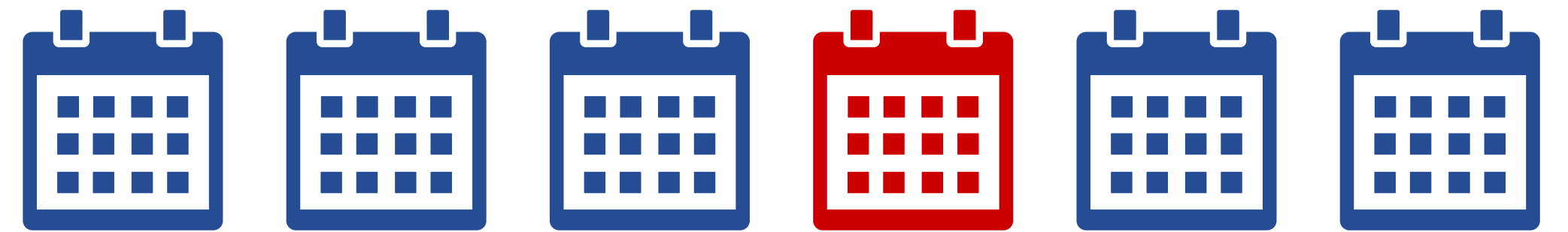
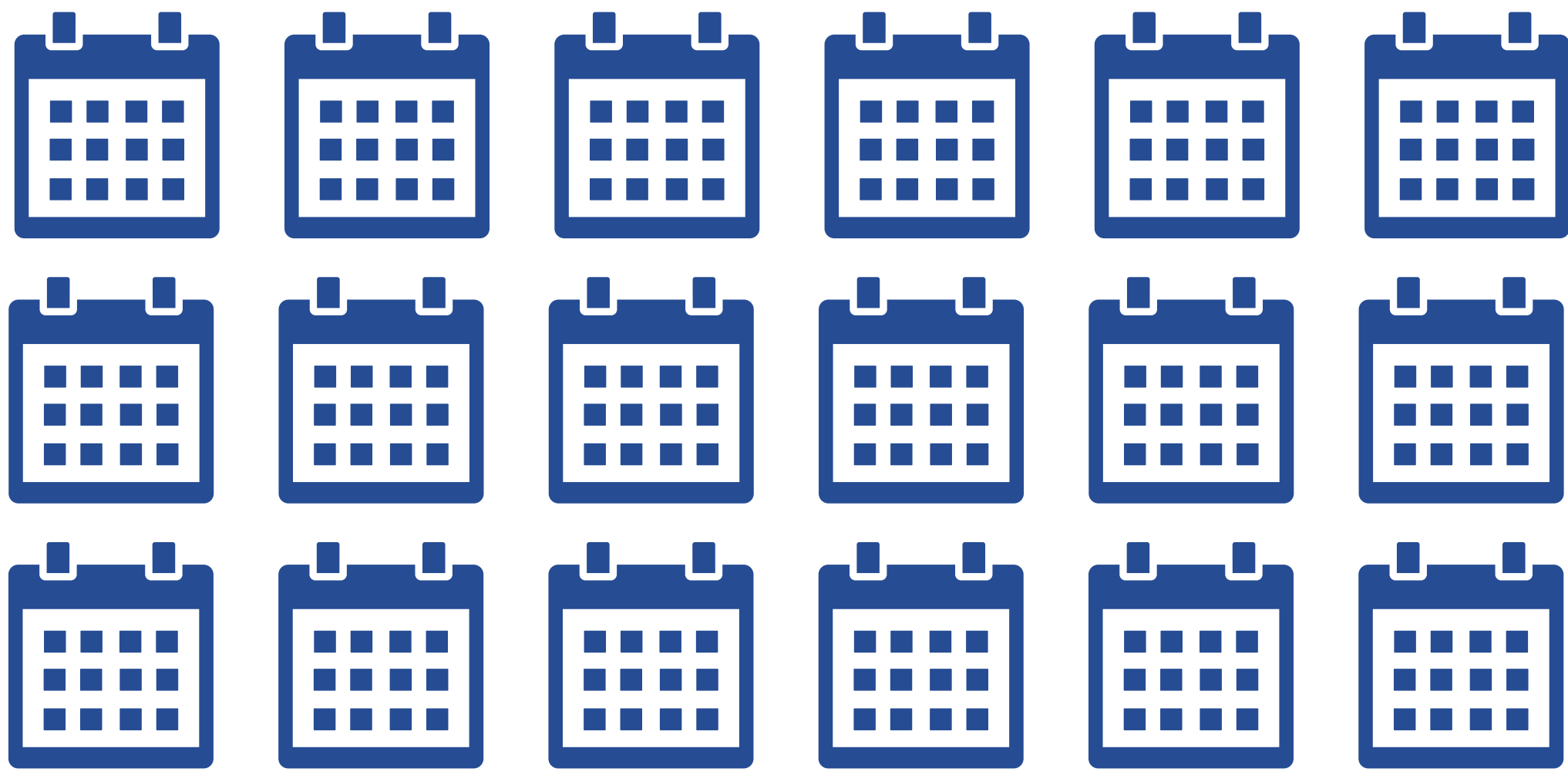
Communities you should know about

**What does a data scientist do?**



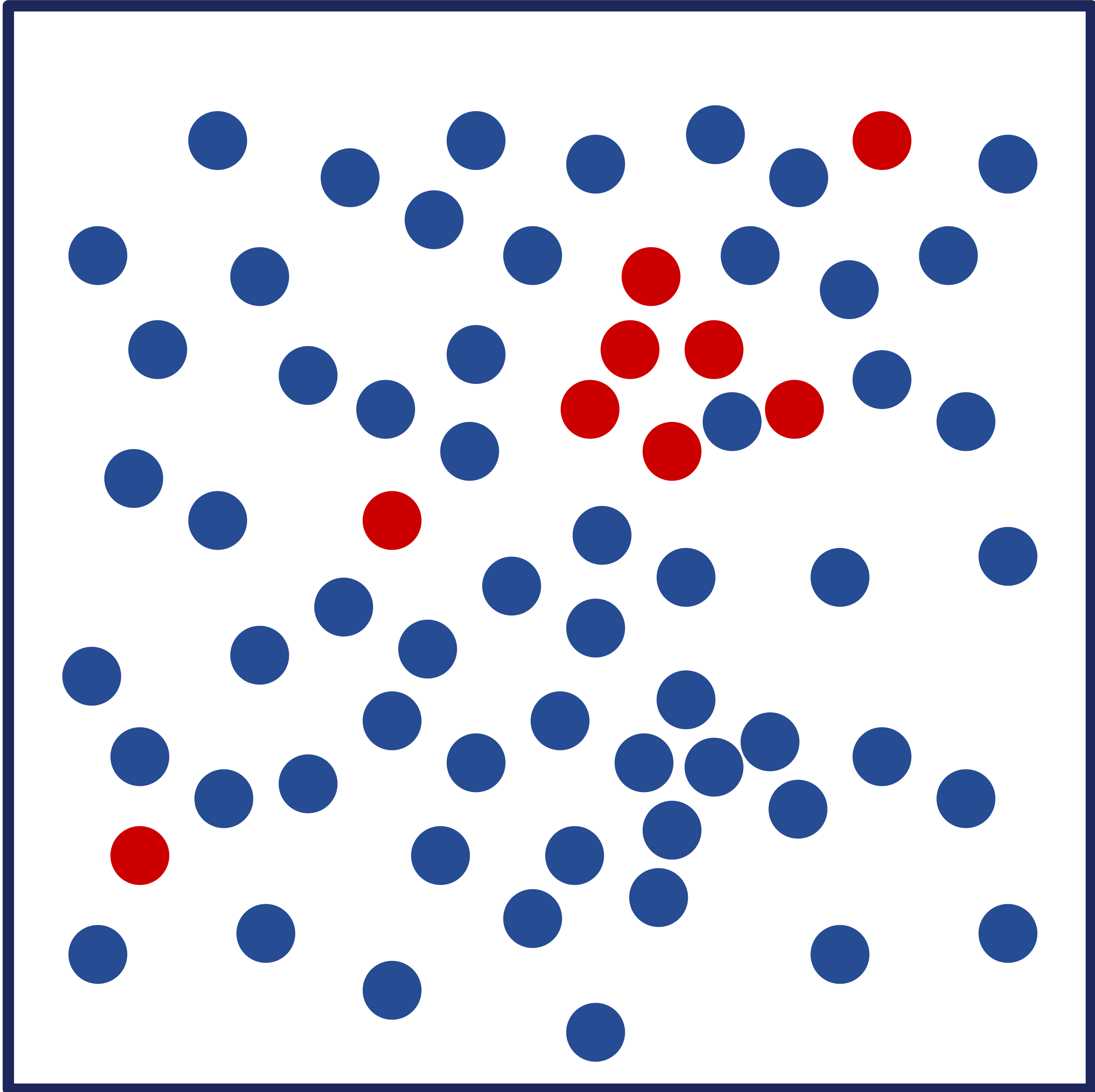


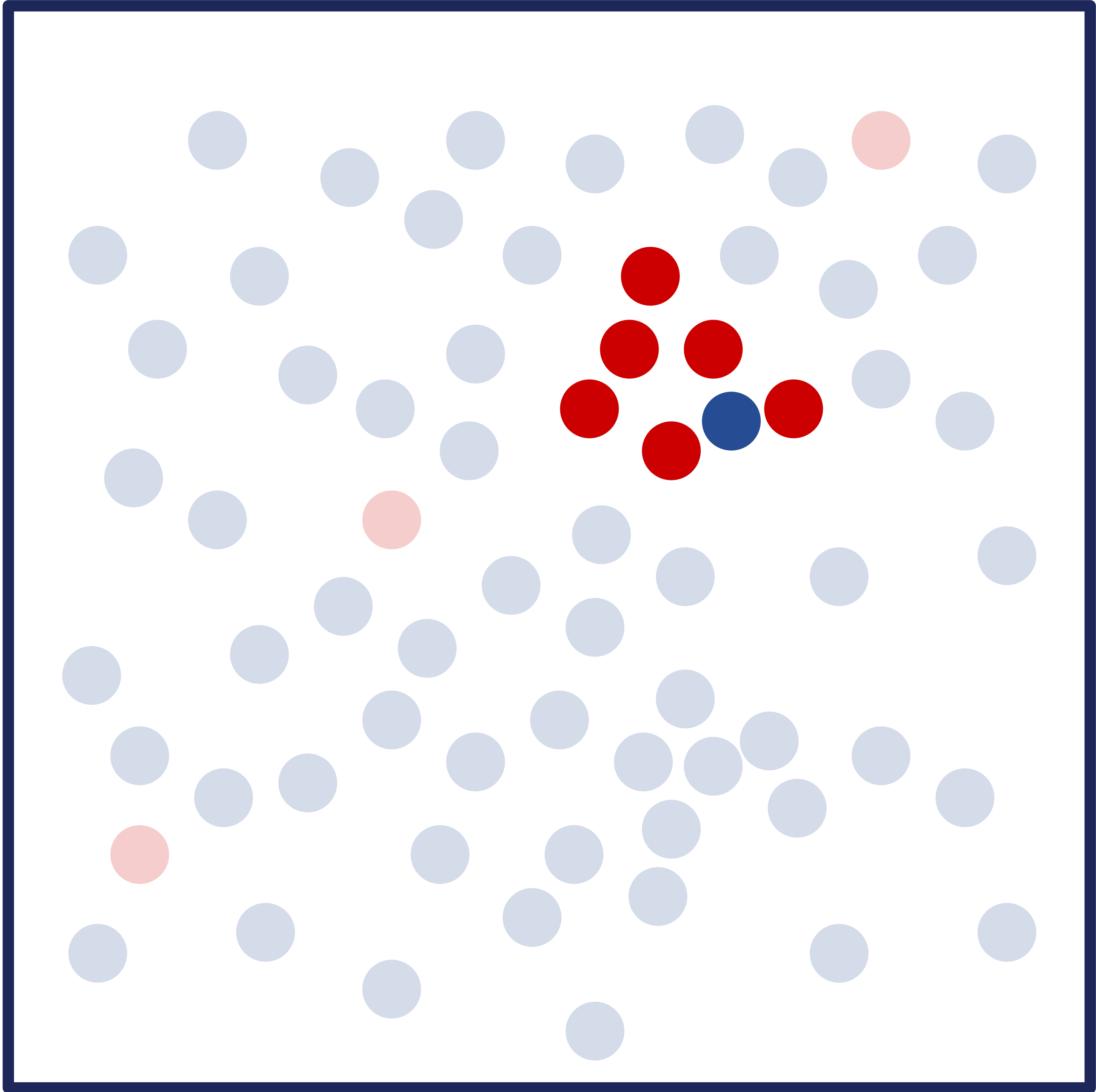


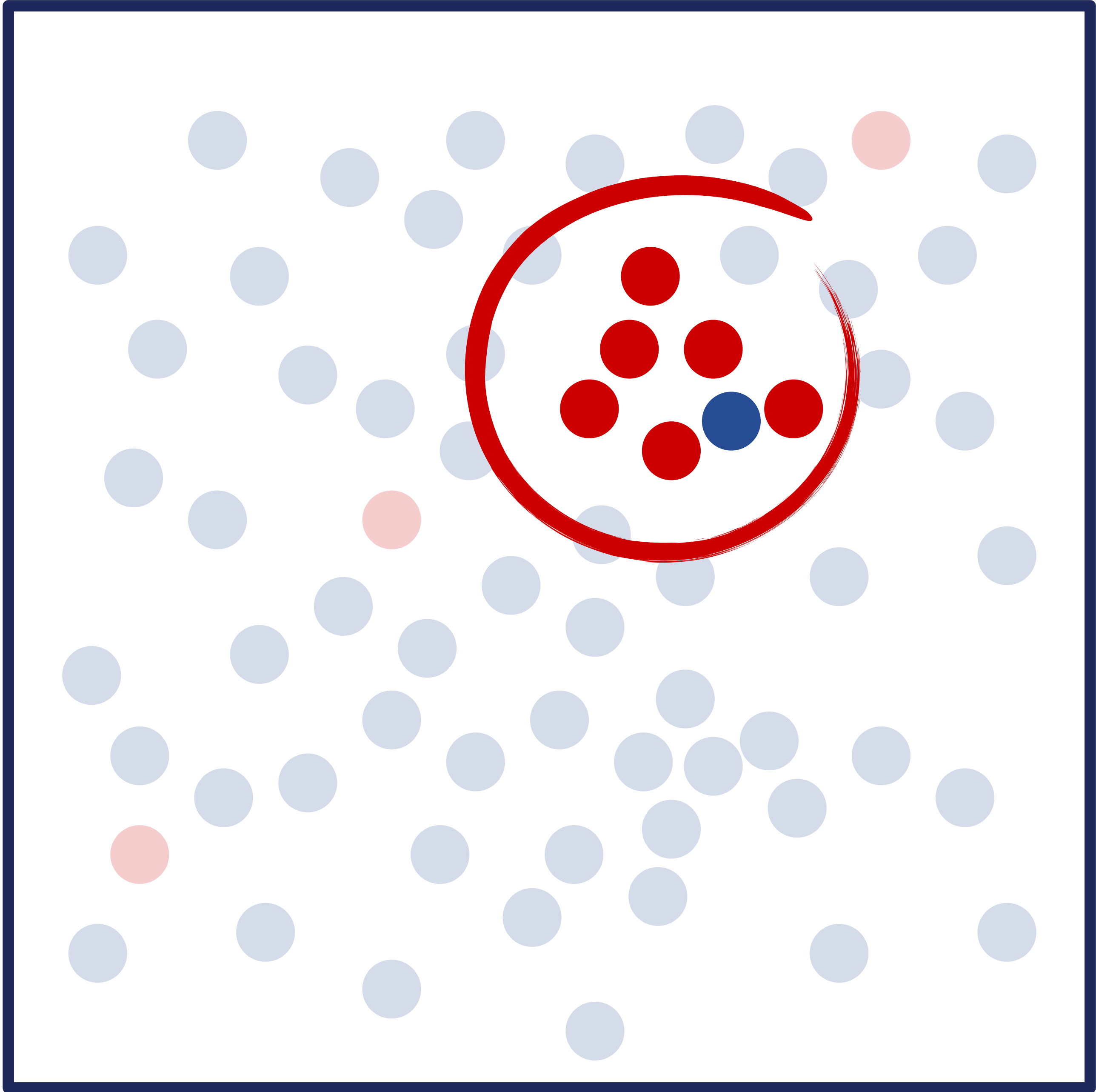








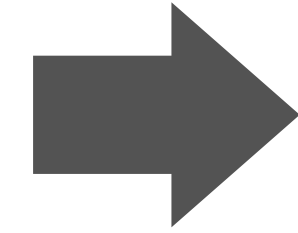
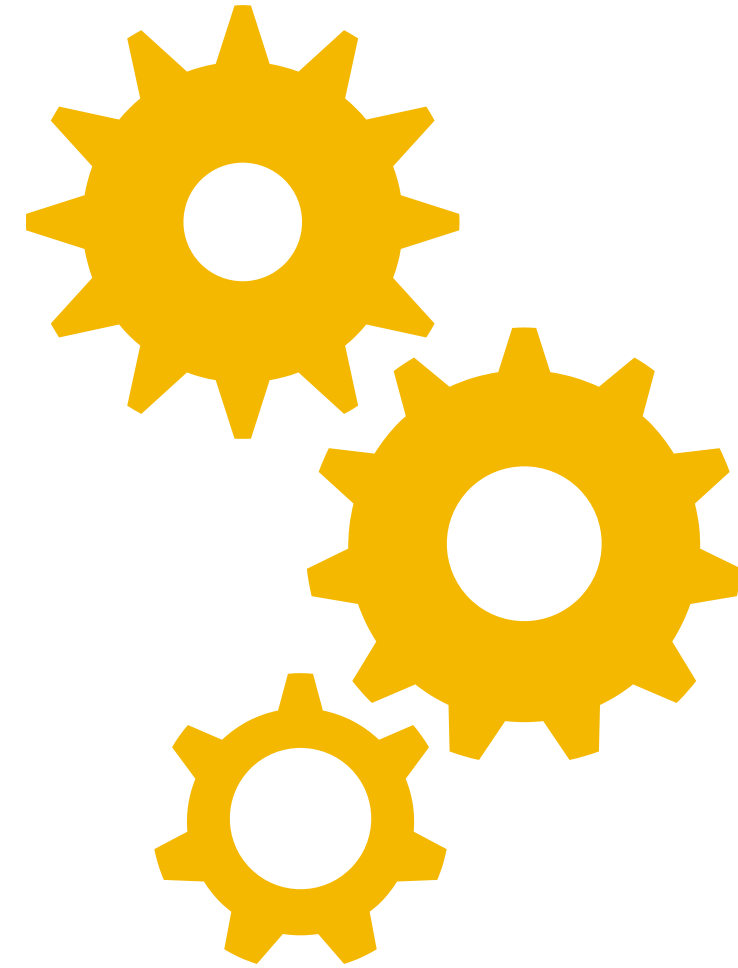
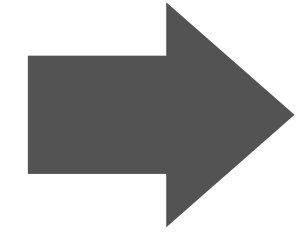
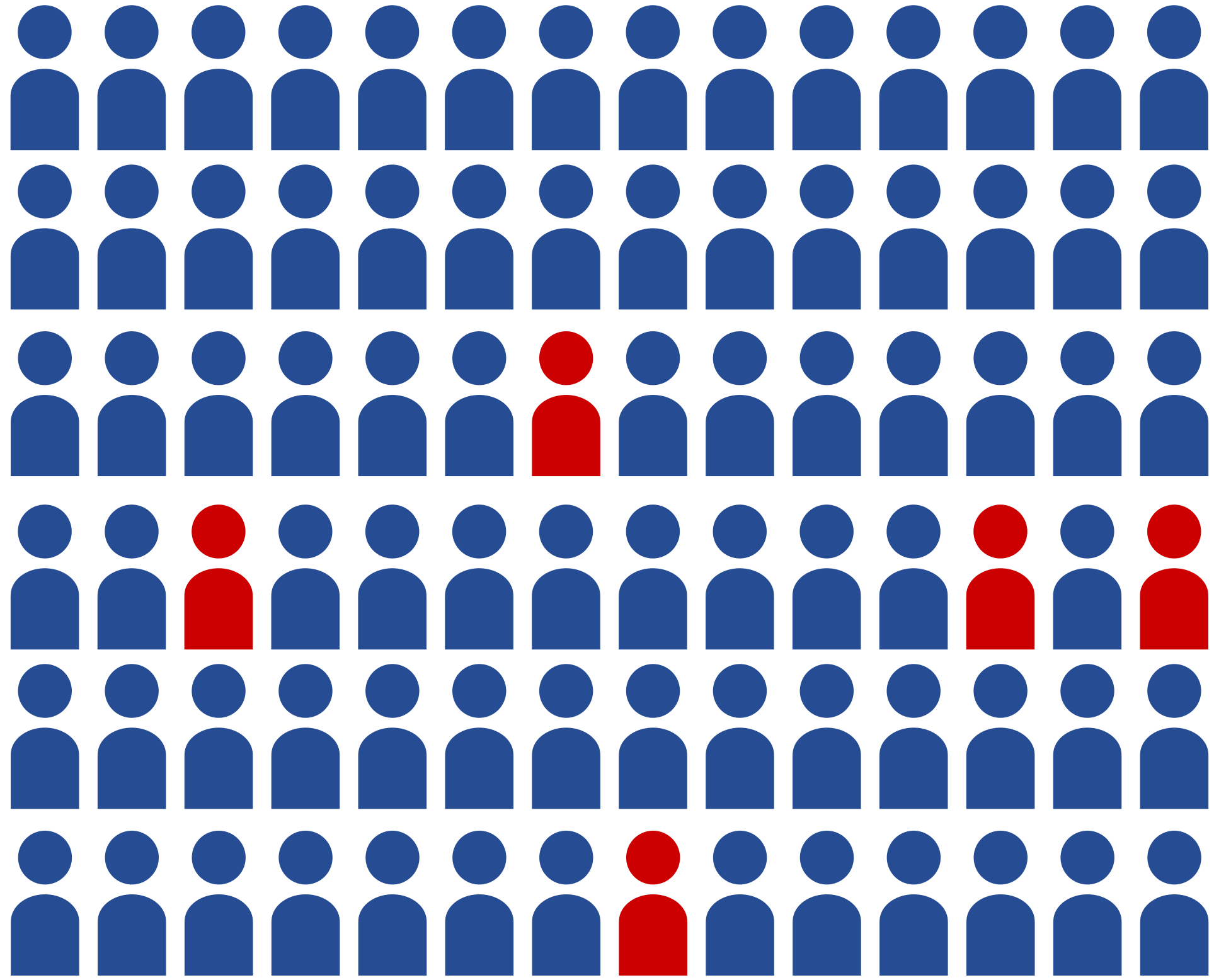


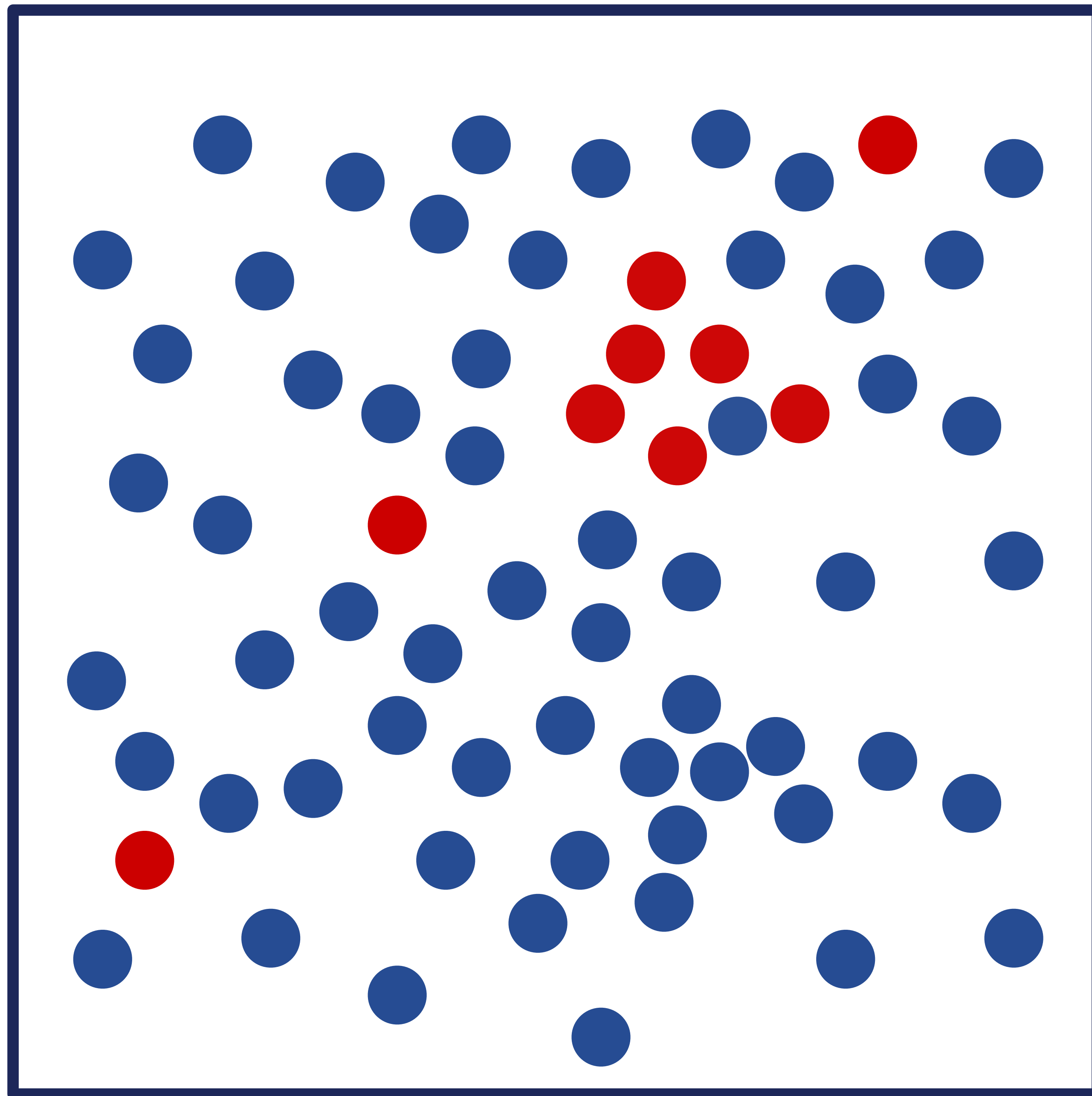


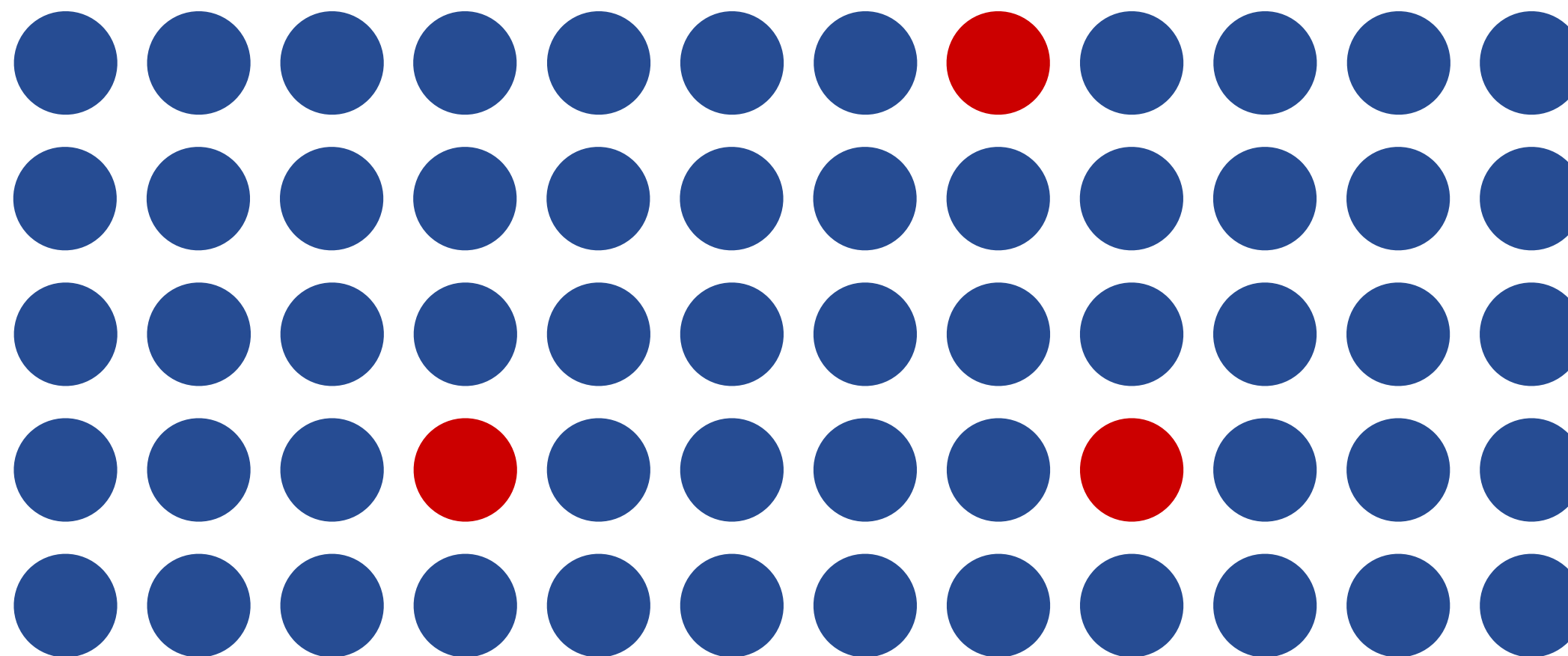
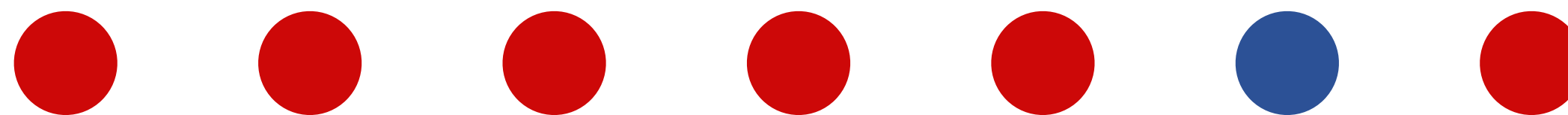
$$f(\text{person}) = [0.67 \quad 0.57 \quad 0.84 \quad \dots \quad 0.08 \quad 0.42 \quad 0.01]$$

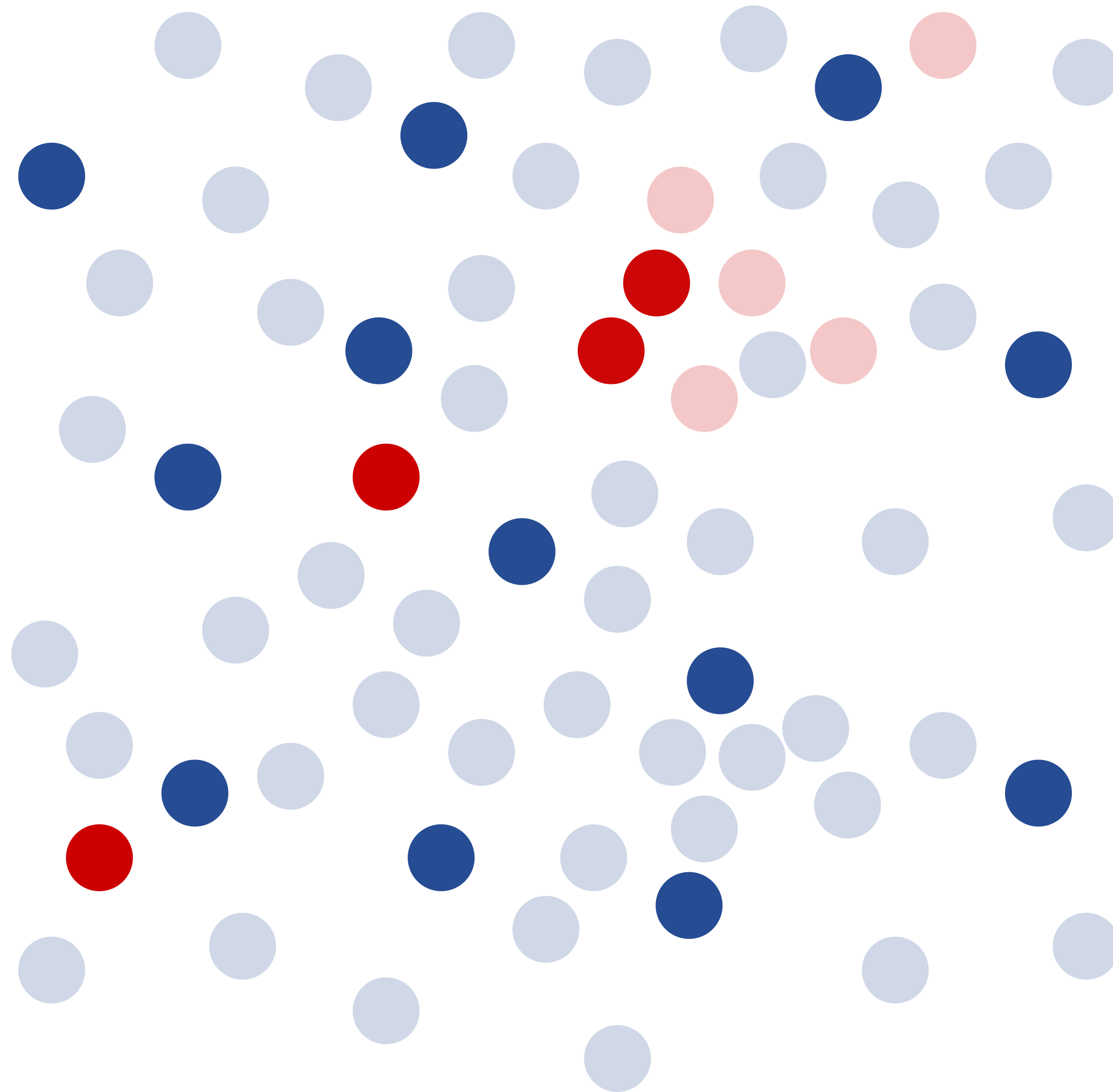
The image shows a function  $f$  applied to a person icon, resulting in a sequence of numerical values. The values are displayed in a grid format, with an ellipsis indicating that the sequence continues.

0.67	0.57	0.84	...	0.08	0.42	0.01
------	------	------	-----	------	------	------

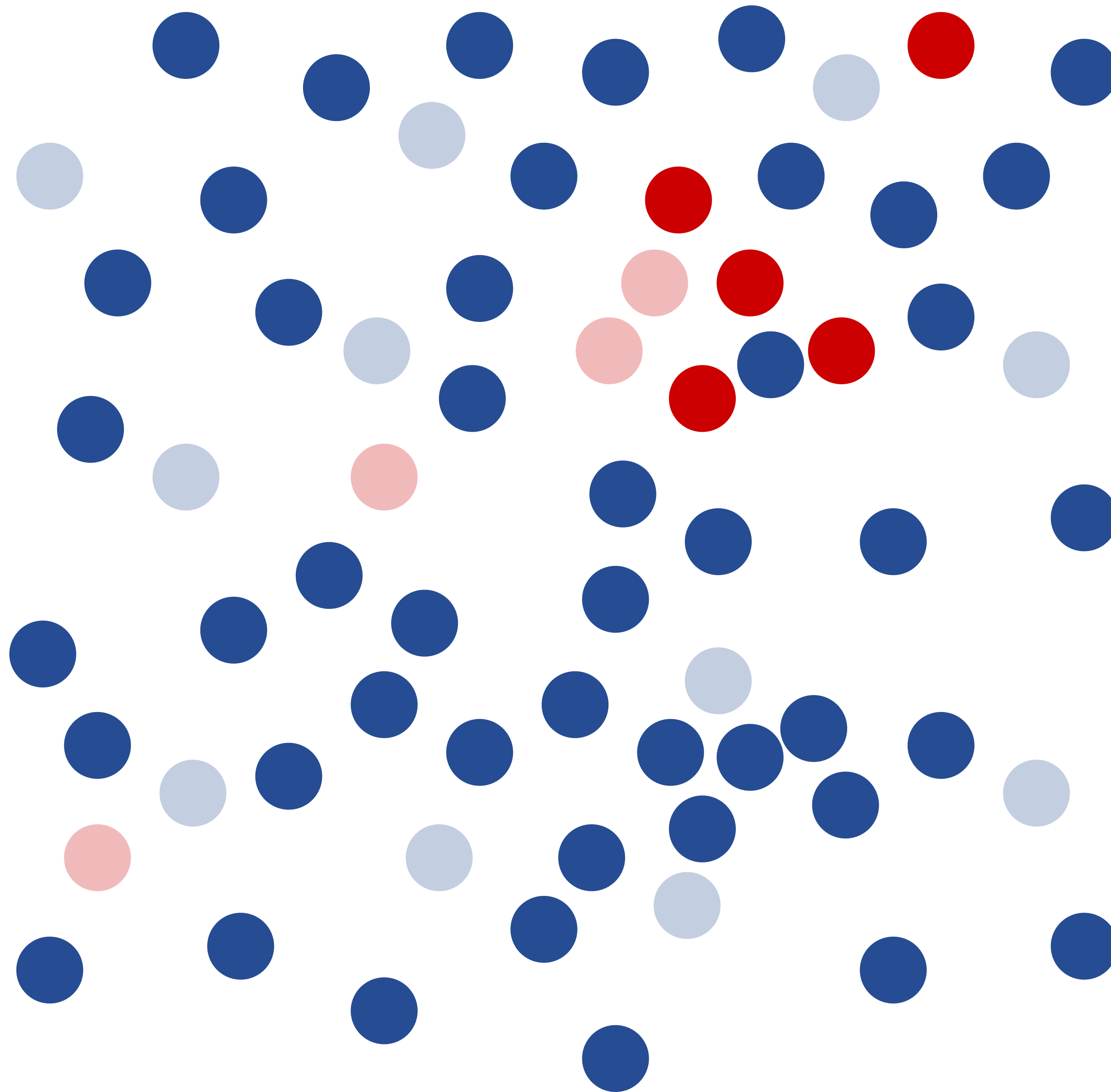


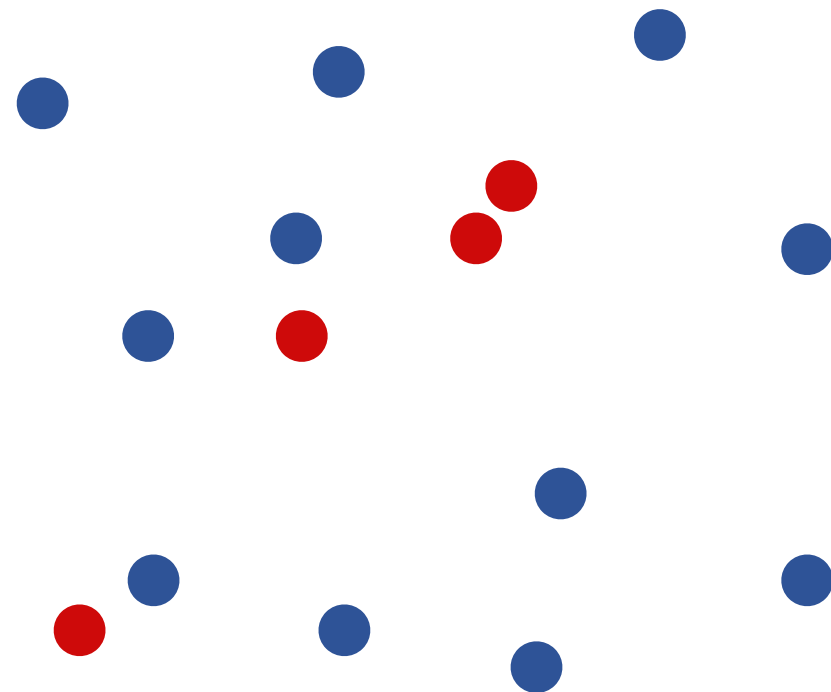
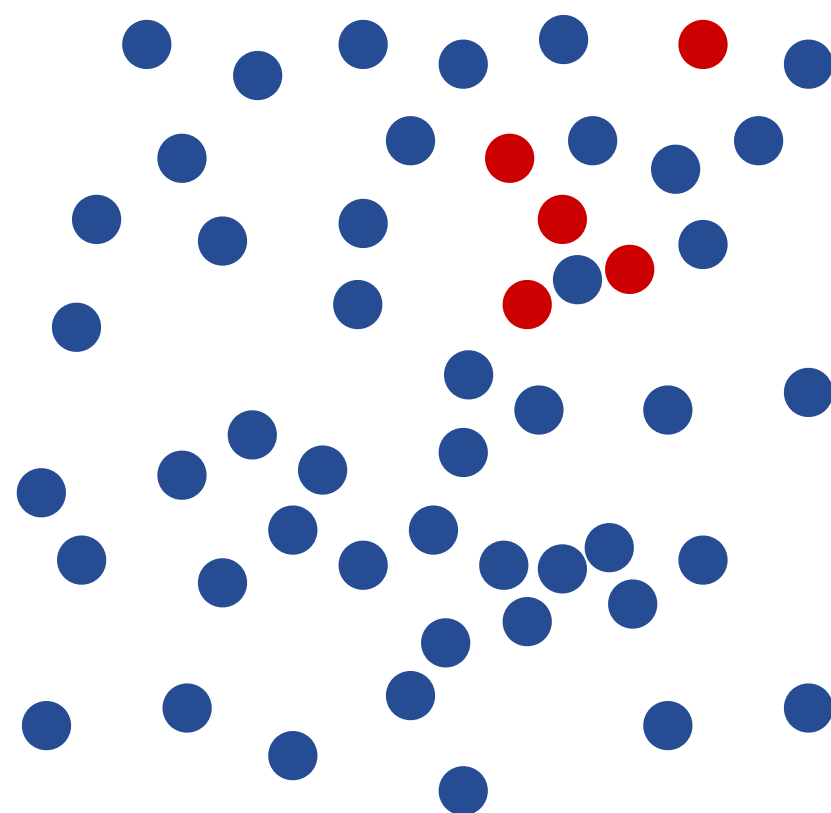


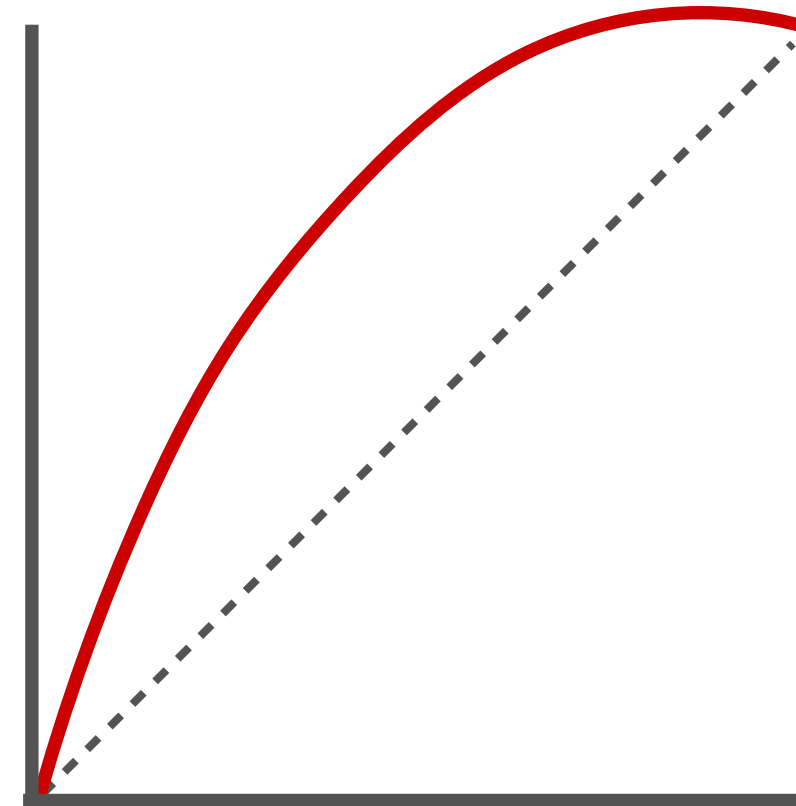
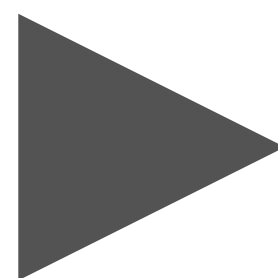
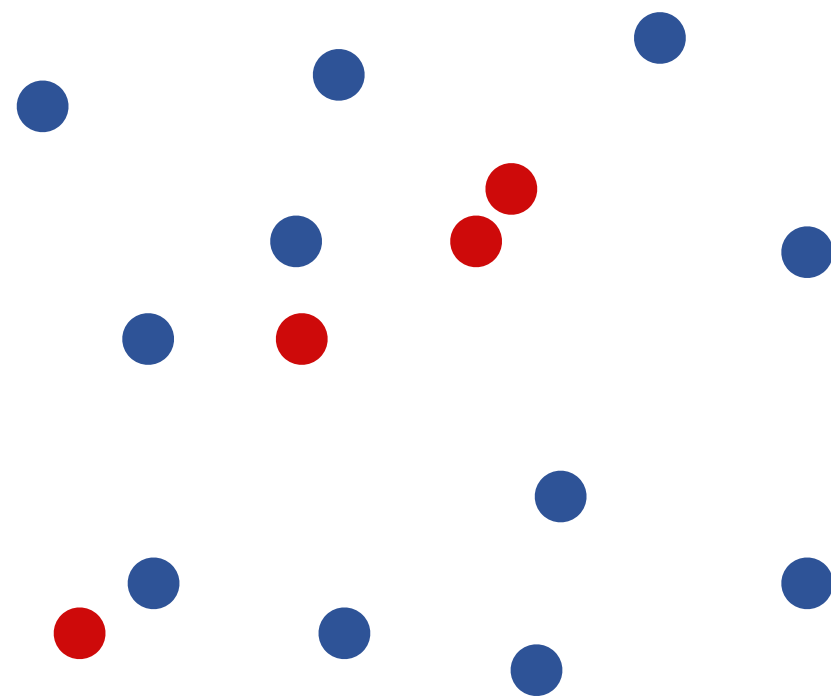
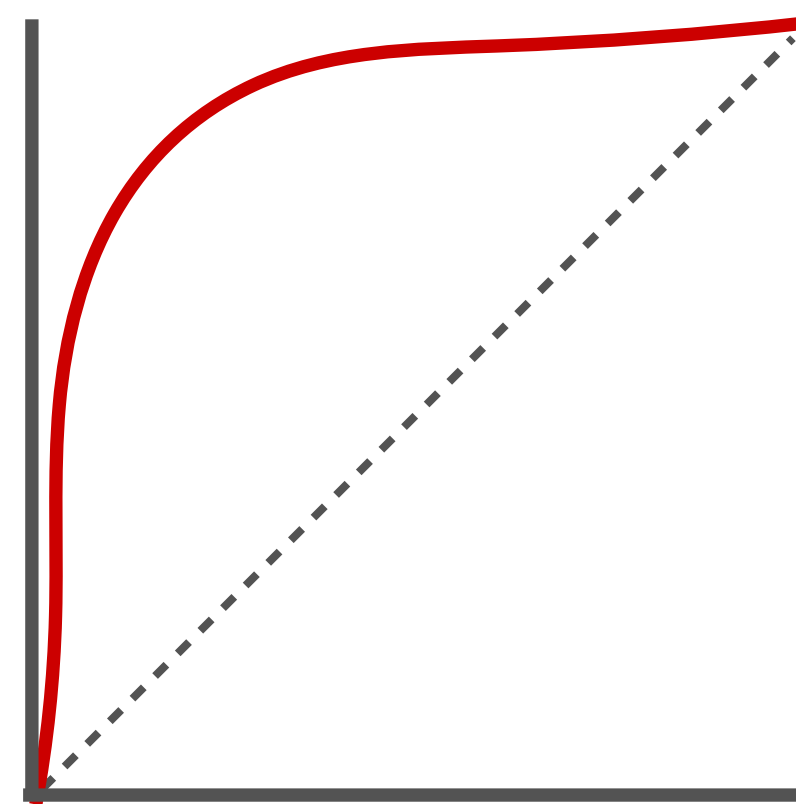
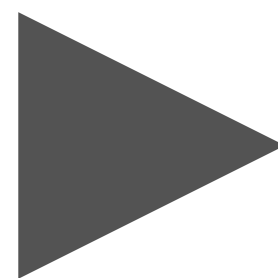
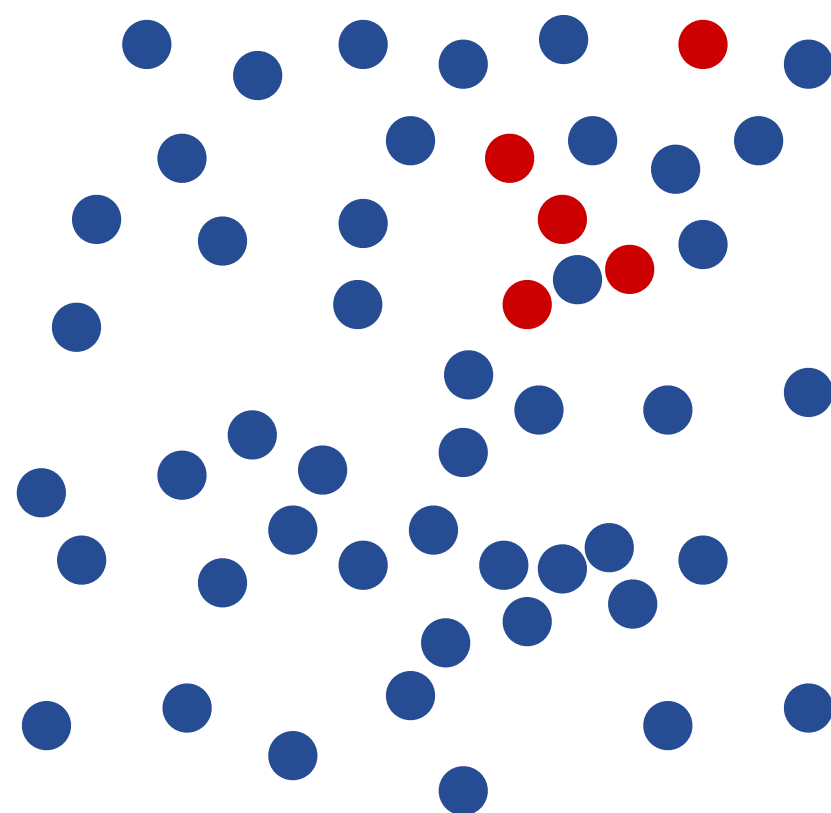


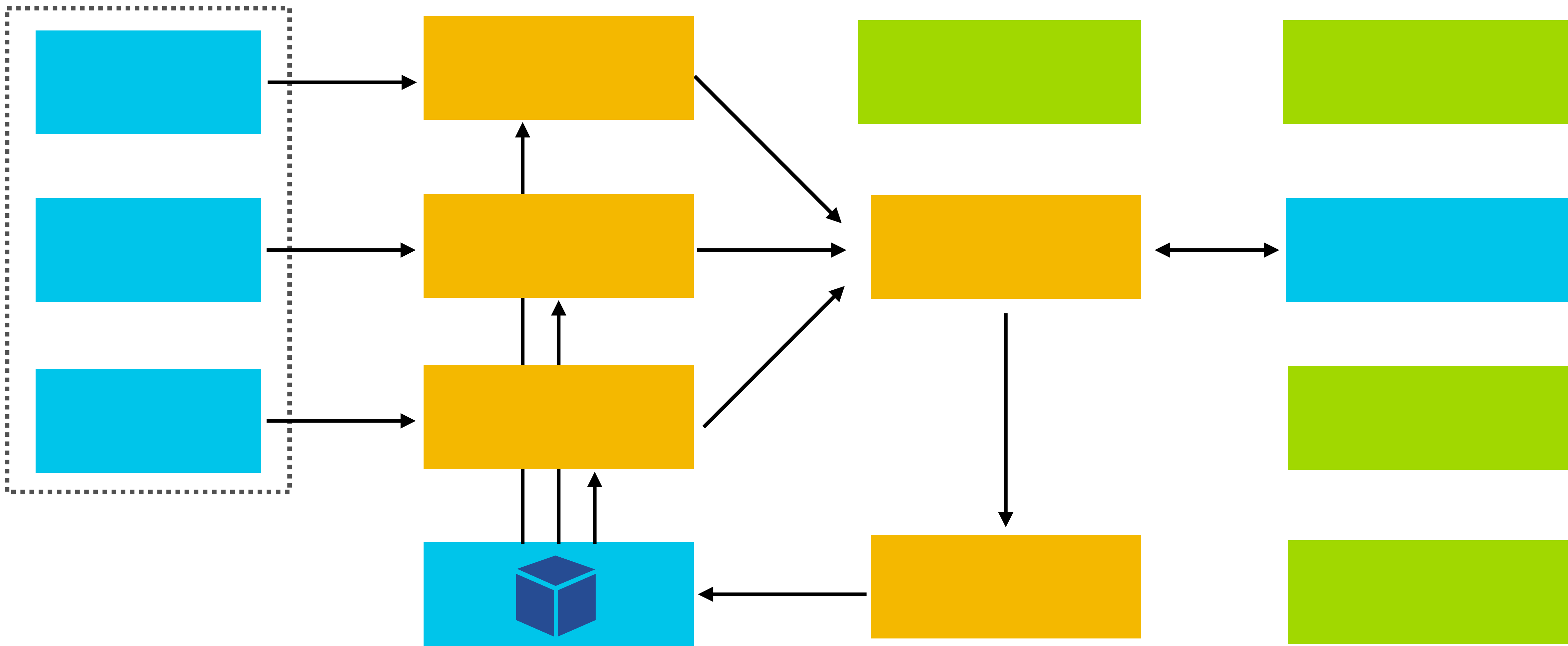


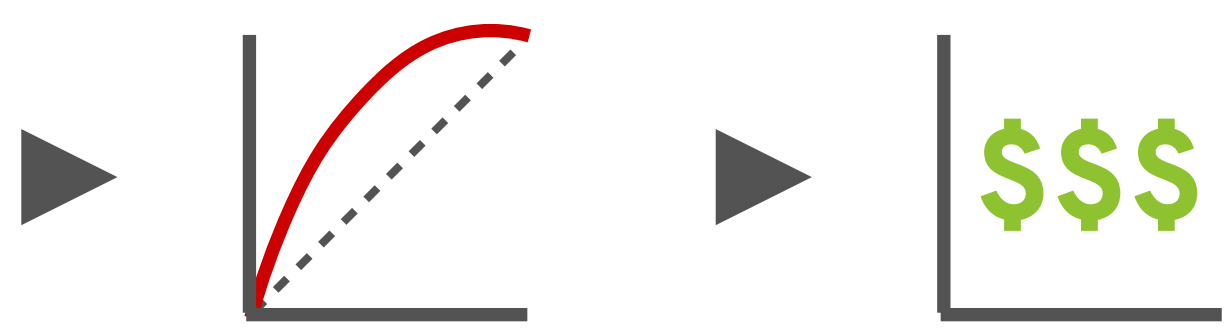
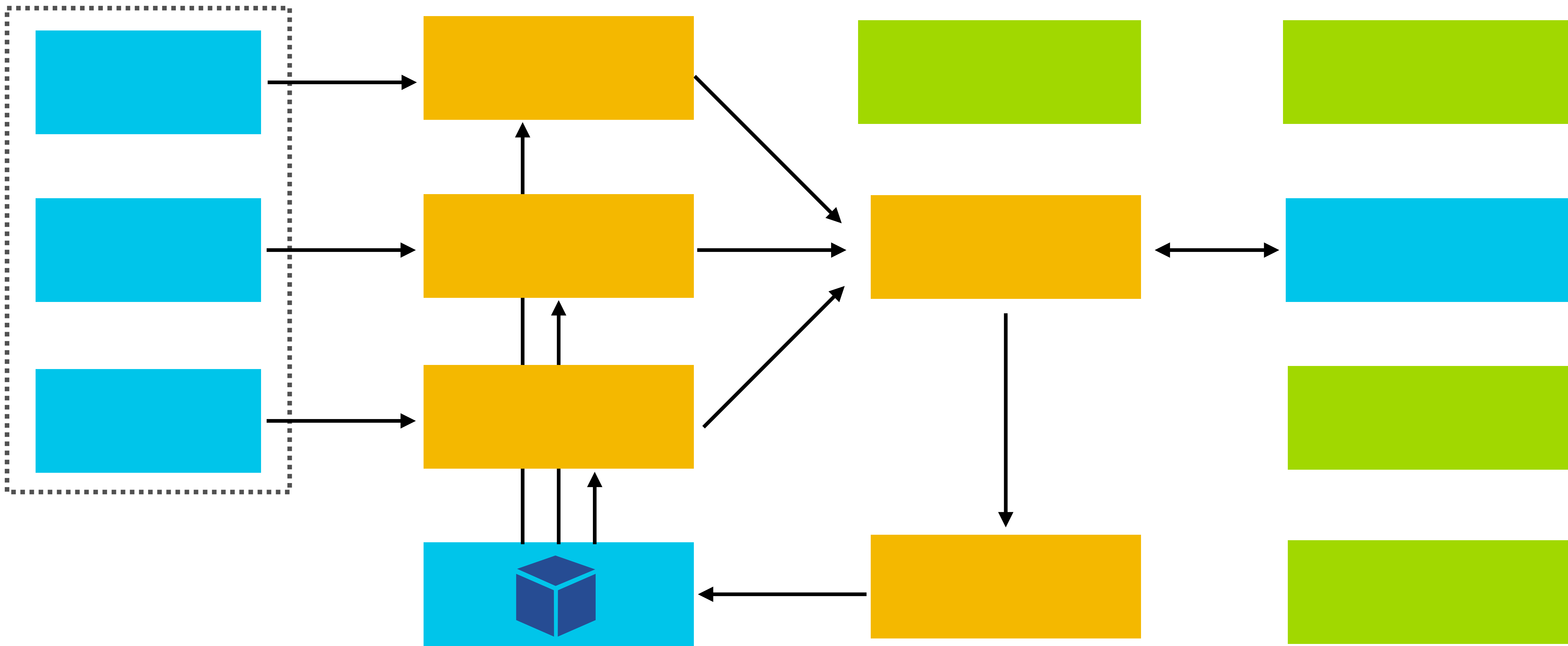




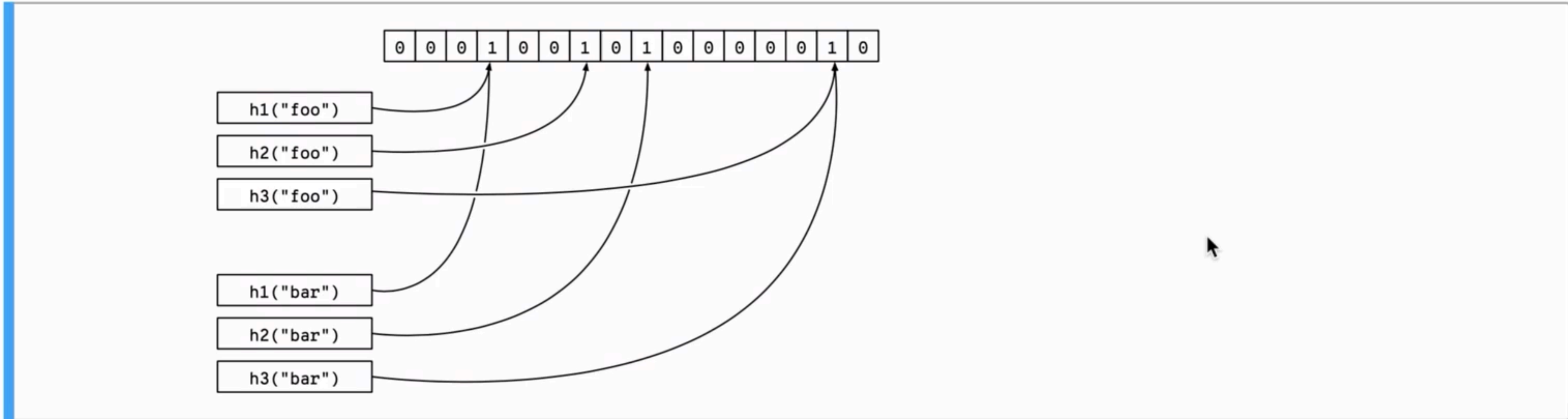








# Bloom filter

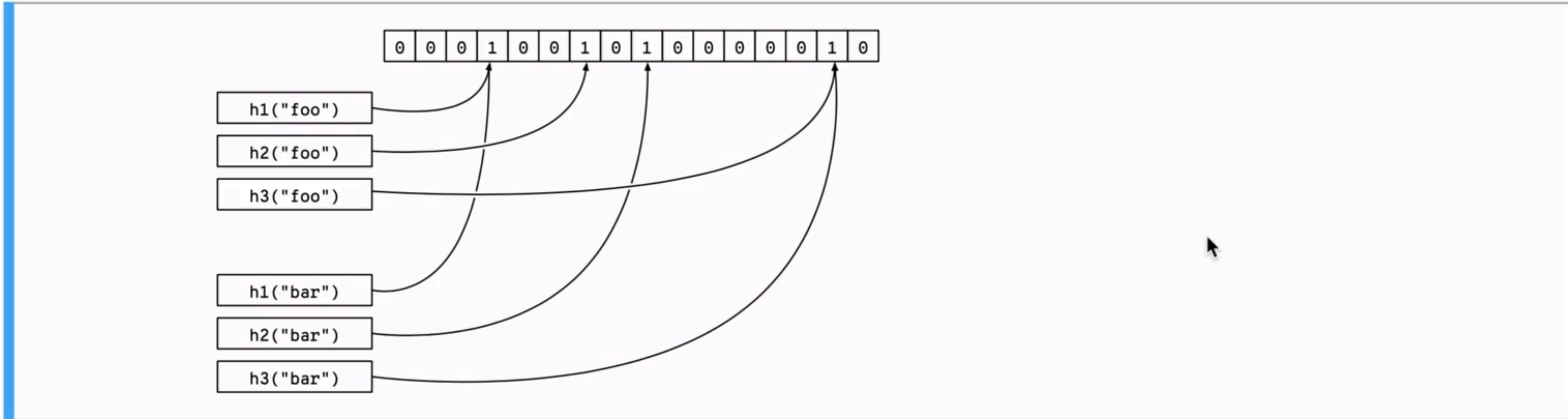


A conventional hash table (or hash table-backed set structure) consists of a series of *buckets*. Hash table insert looks like this:

1. First, use the hash value of the key to identify the index of the bucket that should contain it.
2. If the bucket is empty, update the bucket to contain the key and value (with a trivial value in the case of a hashed set).
3. If the bucket is not empty and the key stored in it is not the one you've hashed, handle this *hash collision*. There are several strategies to handle hash collisions precisely; most involve extra lookups (e.g., having a second hash function or going to the next available bucket) or

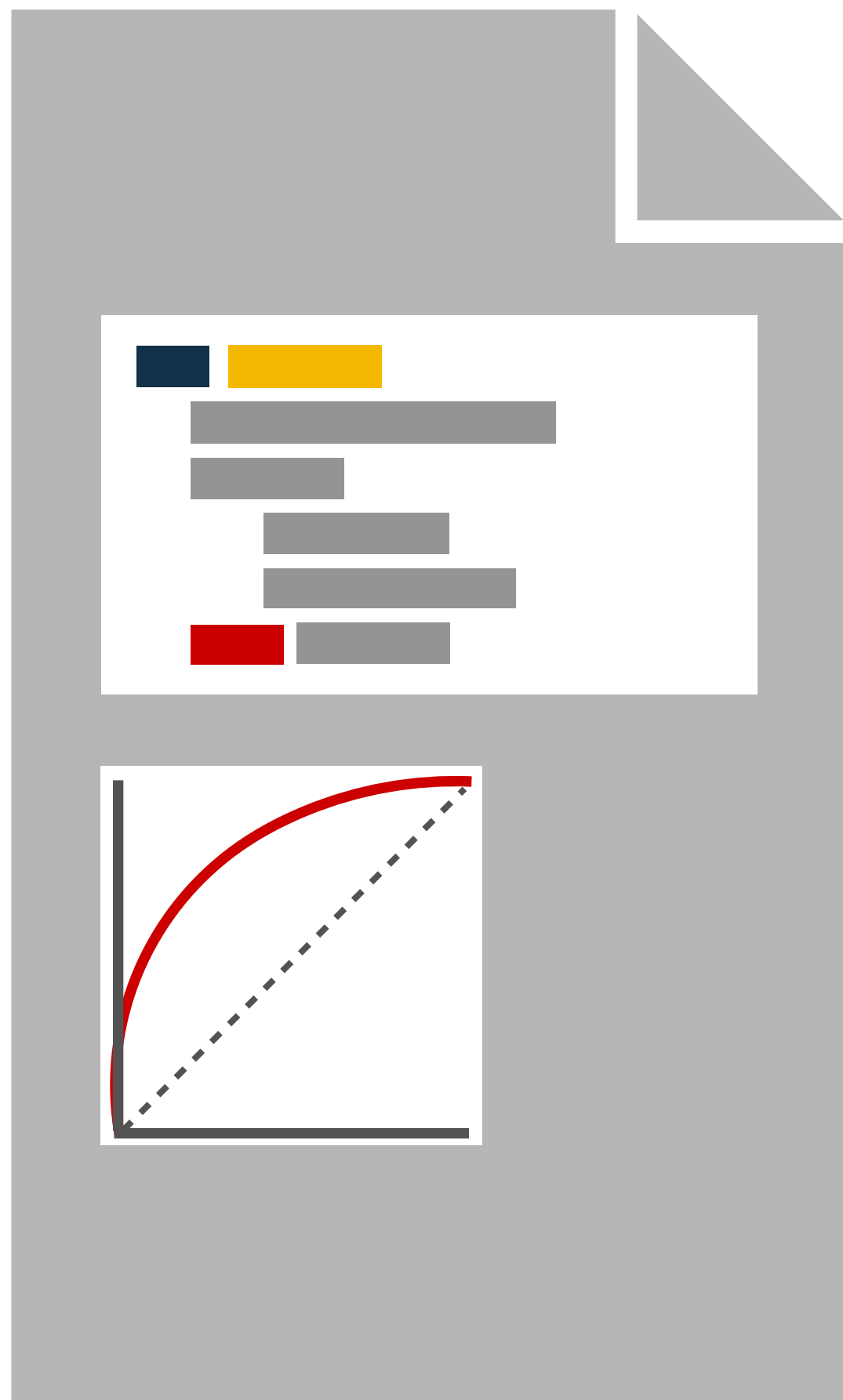


# Bloom filter

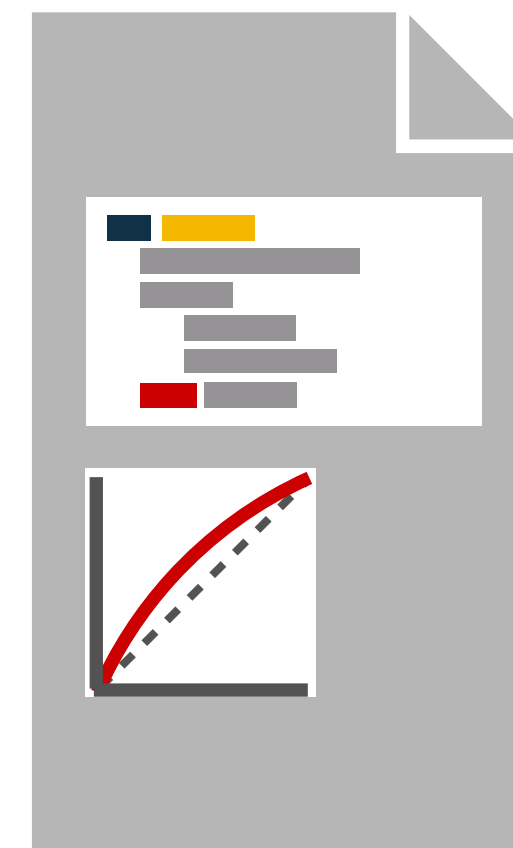
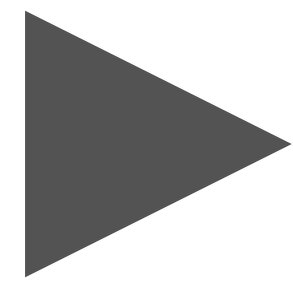
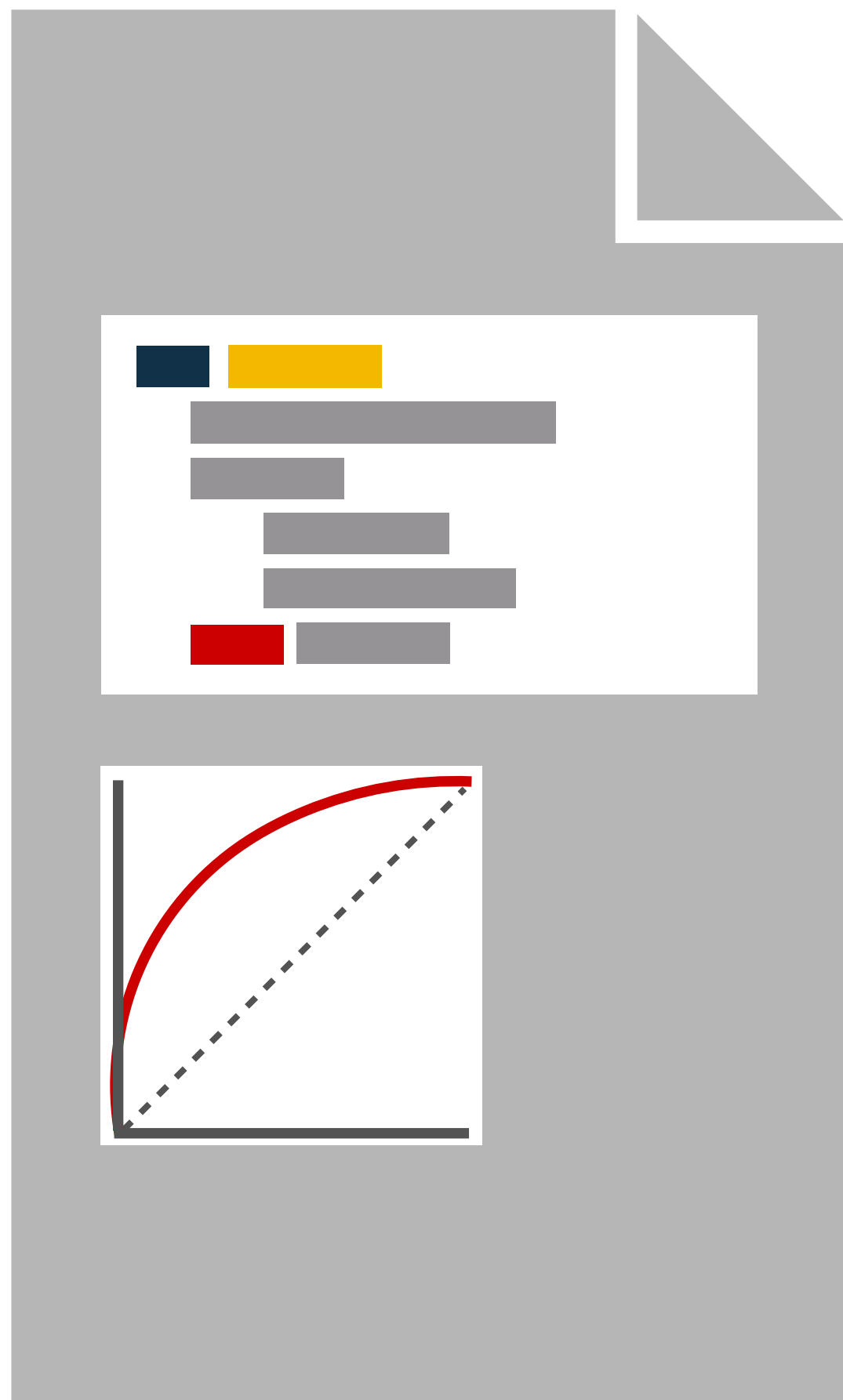


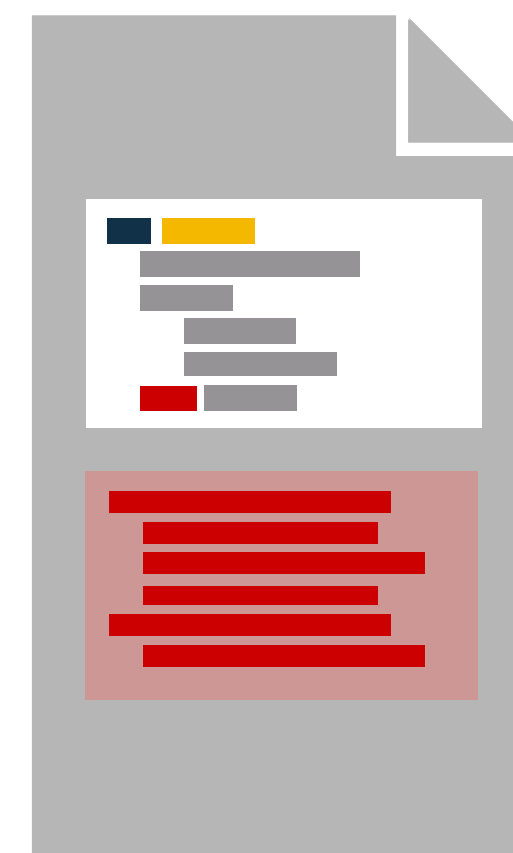
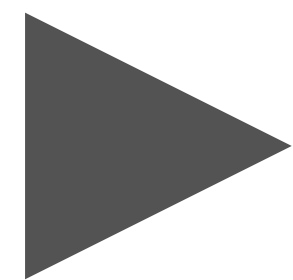
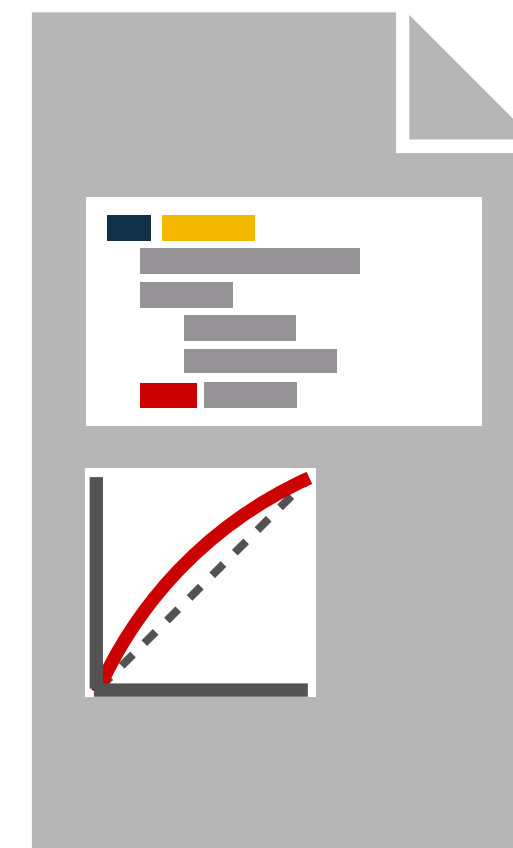
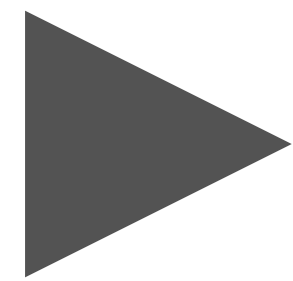
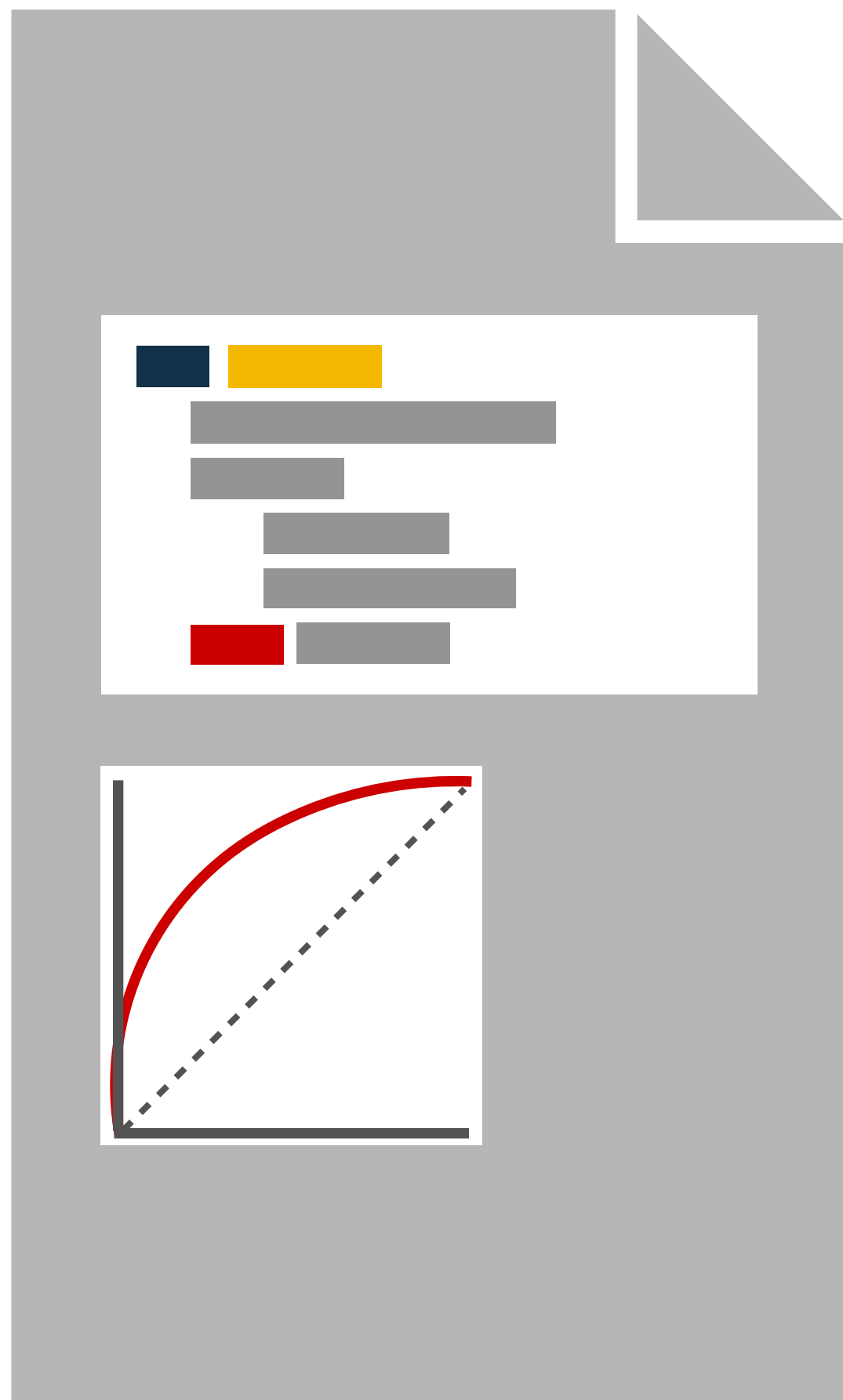
A conventional hash table (or hash table-backed set structure) consists of a series of *buckets*. Hash table insert looks like this:

1. First, use the hash value of the key to identify the index of the bucket that should contain it.
2. If the bucket is empty, update the bucket to contain the key and value (with a trivial value in the case of a hashed set).
3. If the bucket is not empty and the key stored in it is not the one you've hashed, handle this *hash collision*. There are several strategies to handle hash collisions precisely; most involve extra lookups (e.g., having a second hash function or going to the next available bucket) or



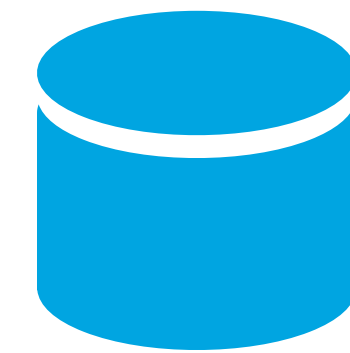


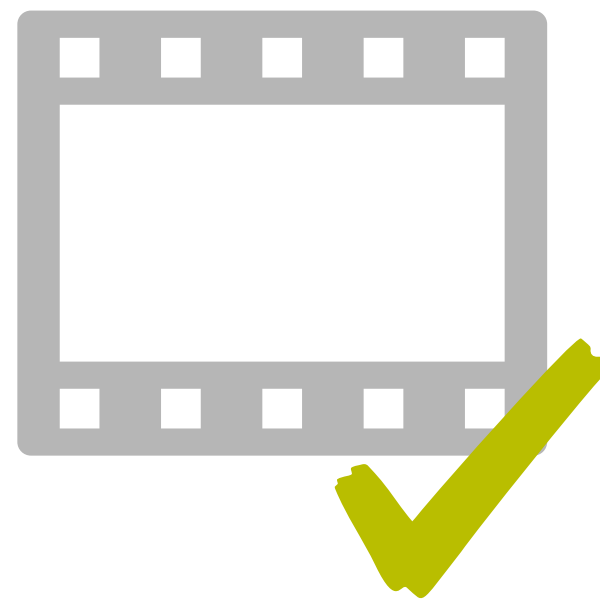




# **Container workflows for data science**



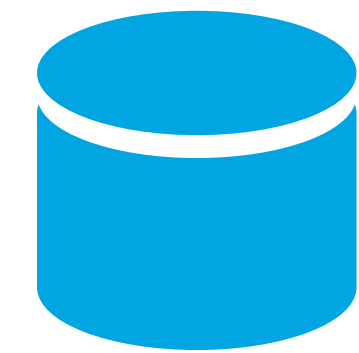
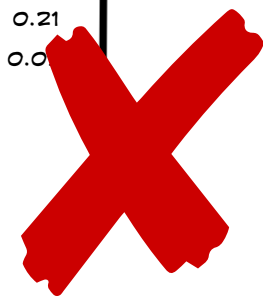


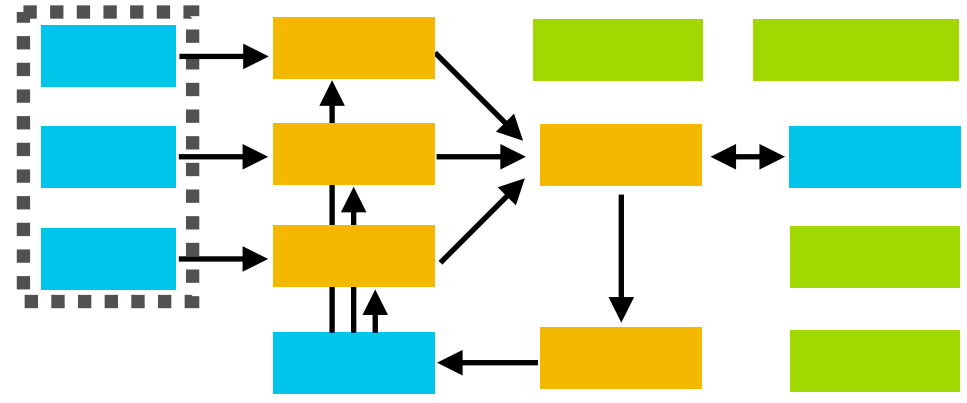


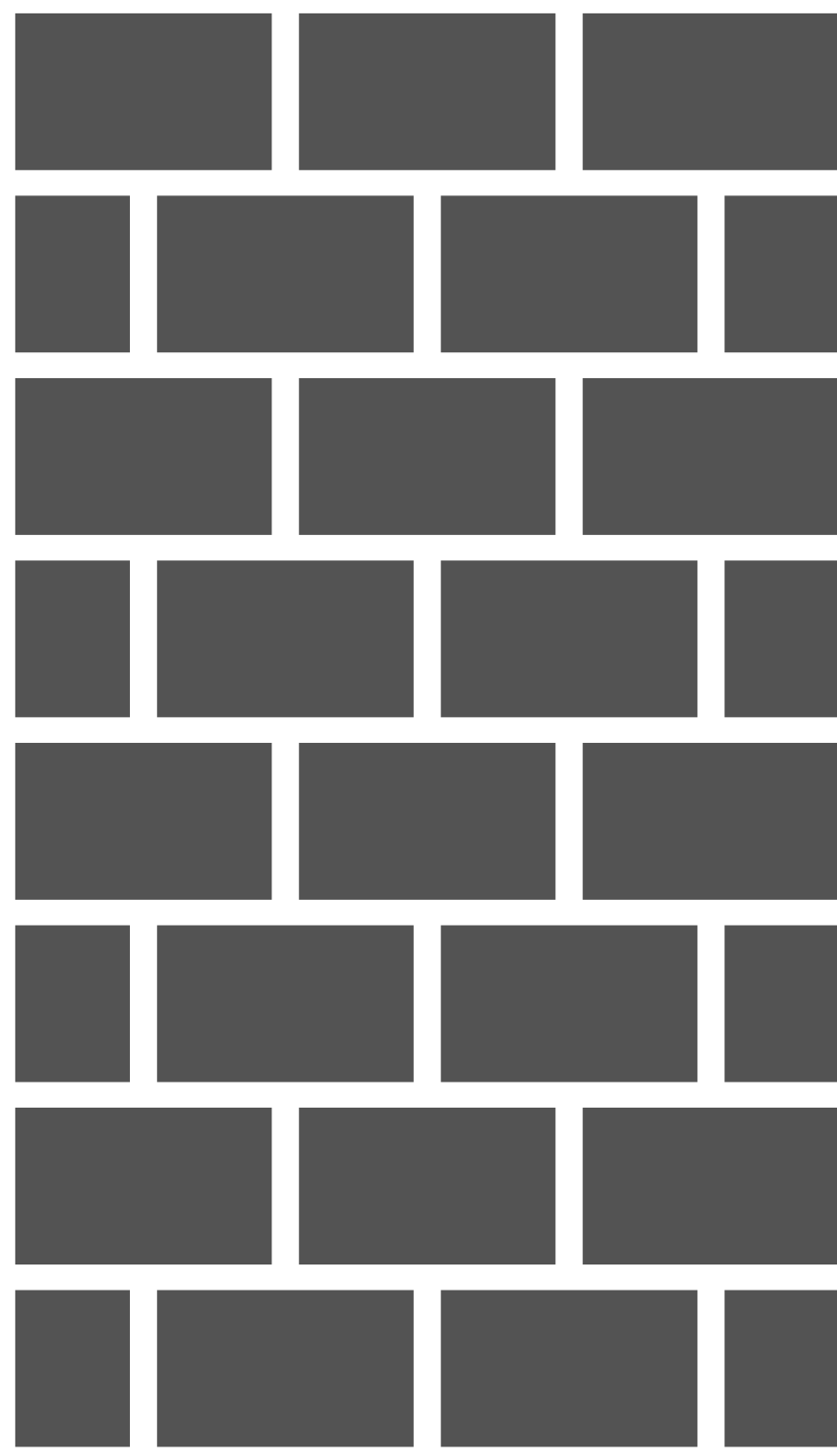
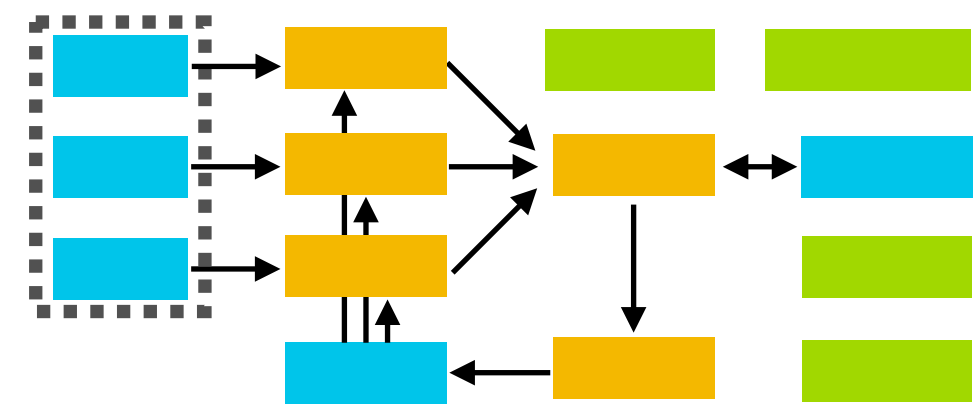
0	0	0	1	1	0	1	0	1	0
0	0	1	0	0	0	1	1	0	0
1	0	1	1	0	1	0	0	0	0
0	0	0	0	0	0	1	1	0	1
0	1	0	0	1	0	0	1	0	0
1	0	0	0	0	1	0	1	1	0
0	0	1	0	1	0	1	0	0	0
0	1	0	0	0	1	0	0	1	1
0	0	0	0	1	0	0	1	0	1
1	1	0	0	0	0	0	0	0	1



0.13	0.13
0.06	0.07
0.07	0.06
0.02	0.08
0.17	0.11
0.11	0.09
0.04	0.18
0.13	0.04
0.13	0.21
0.14	0.0

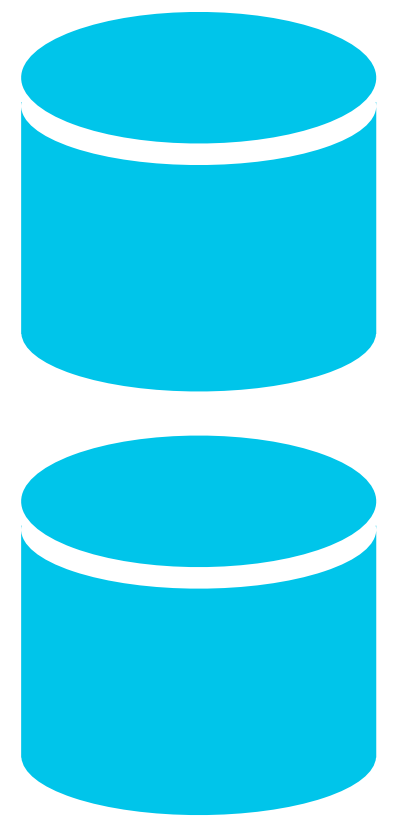
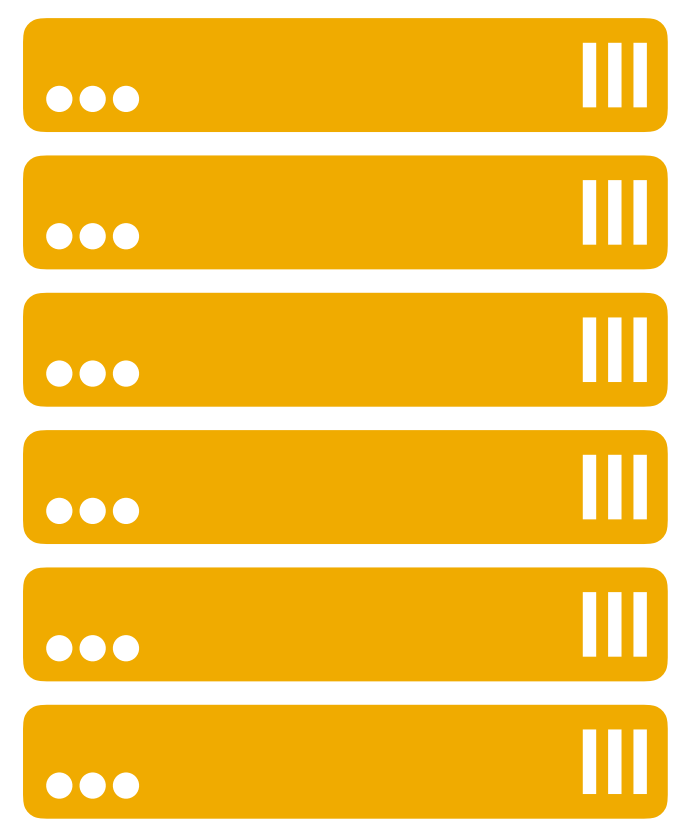






more CPUs

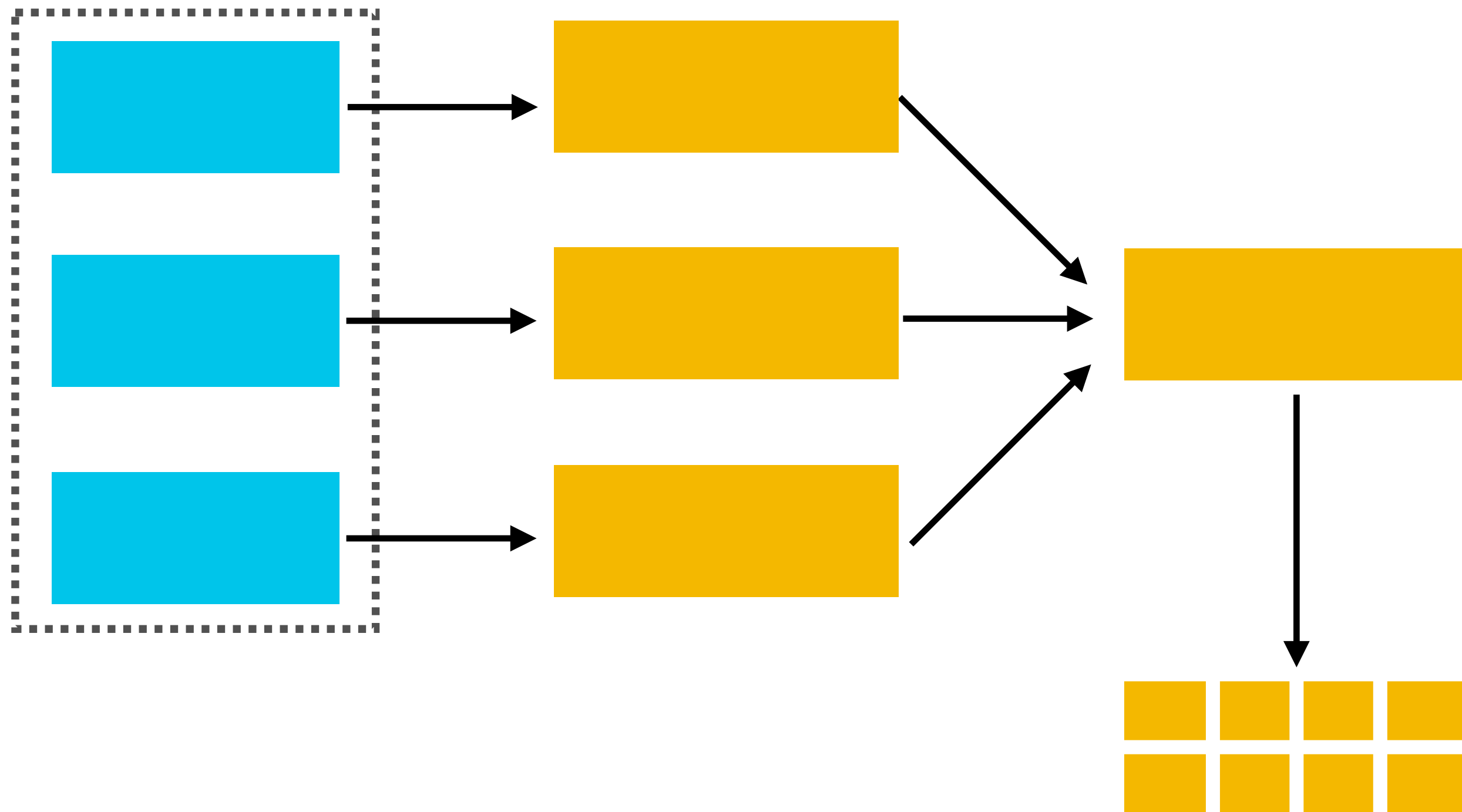
better GPUs

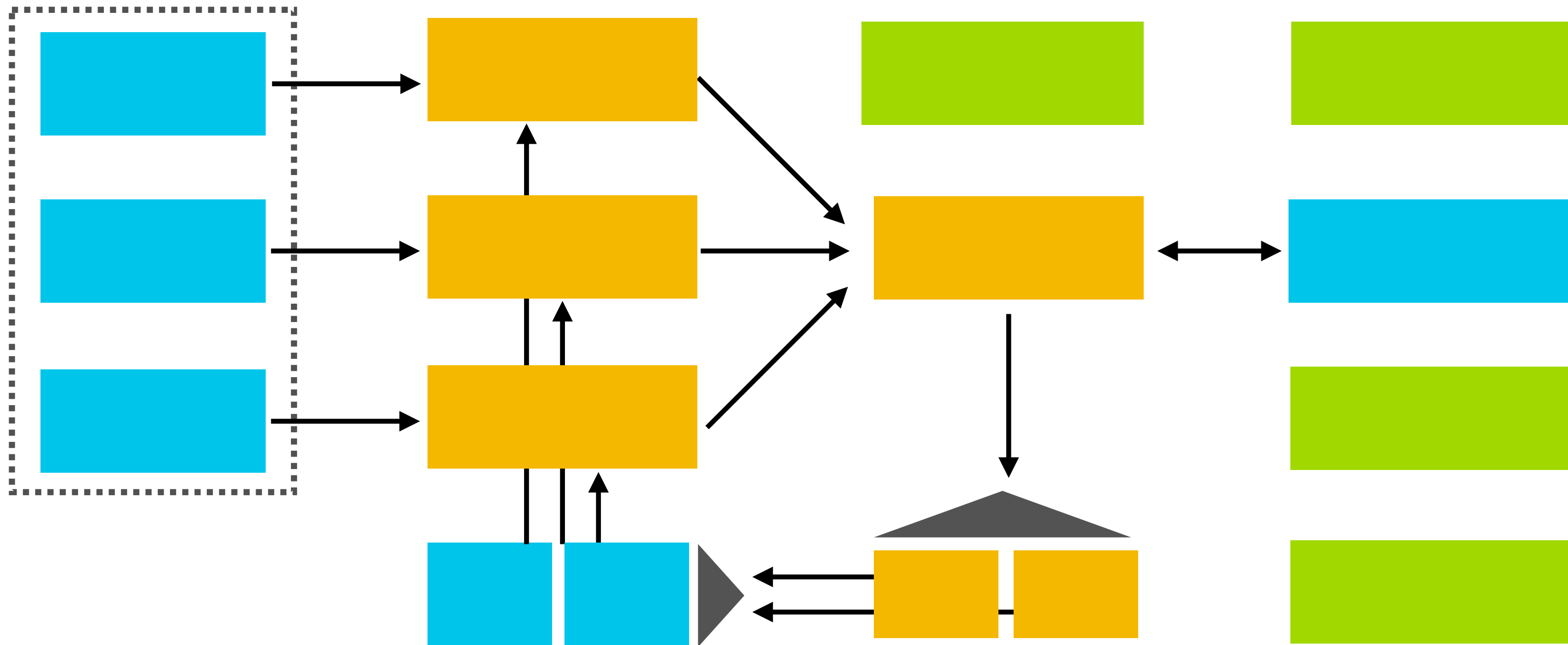


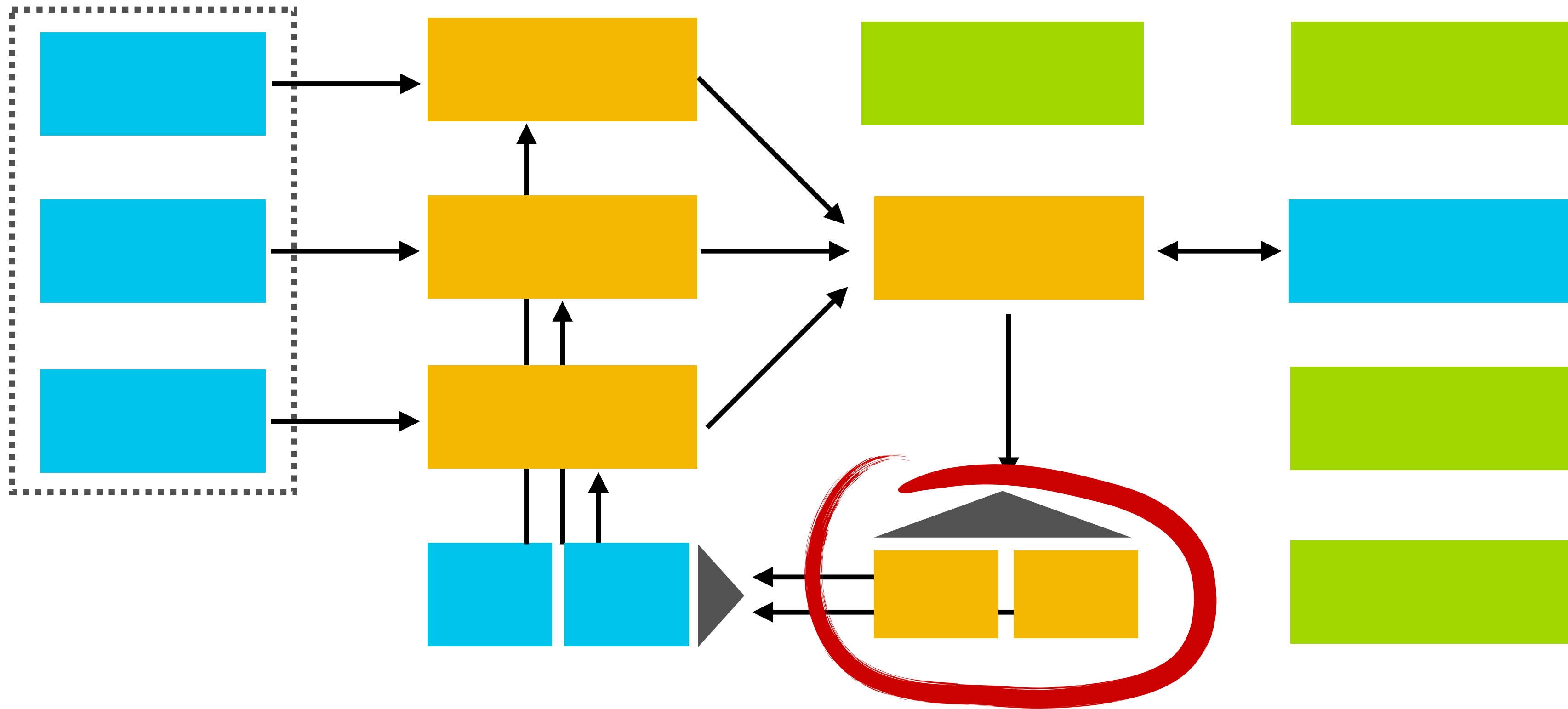
more storage

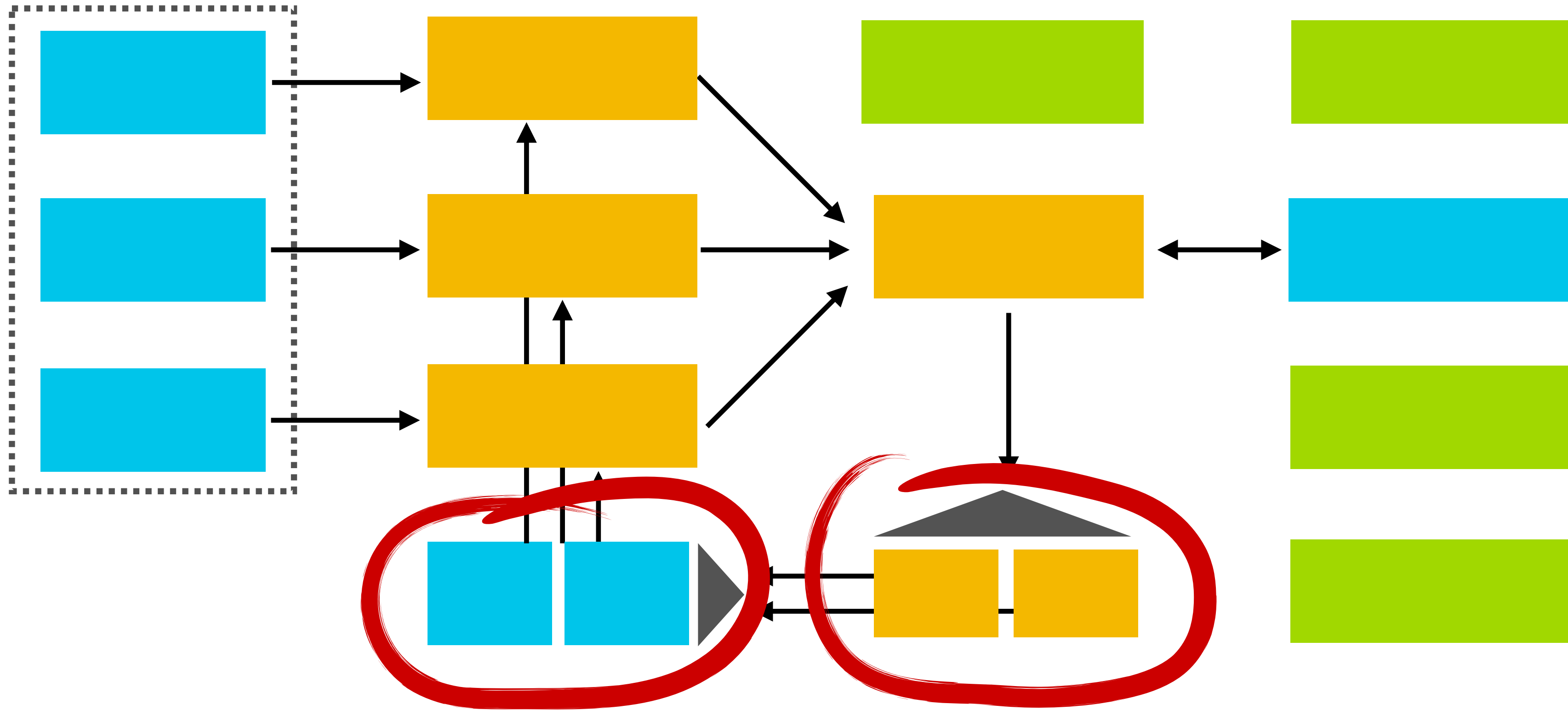
sensitive data



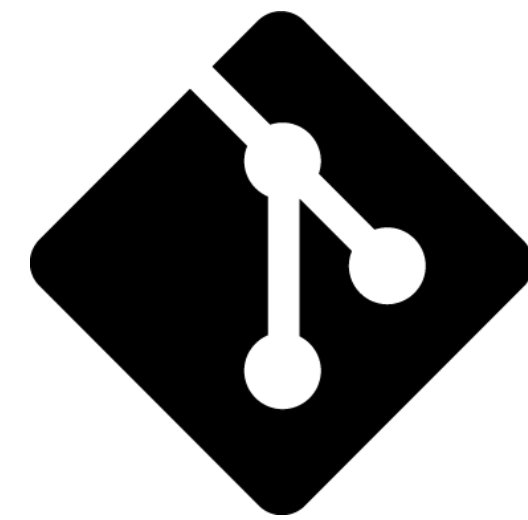
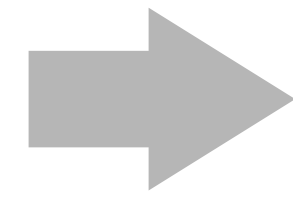




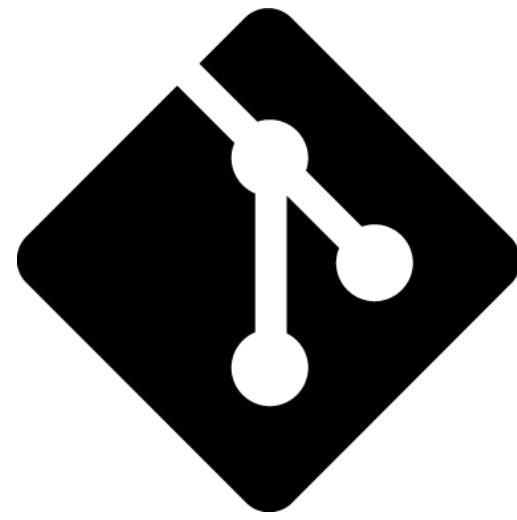
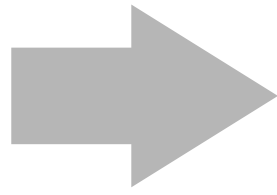




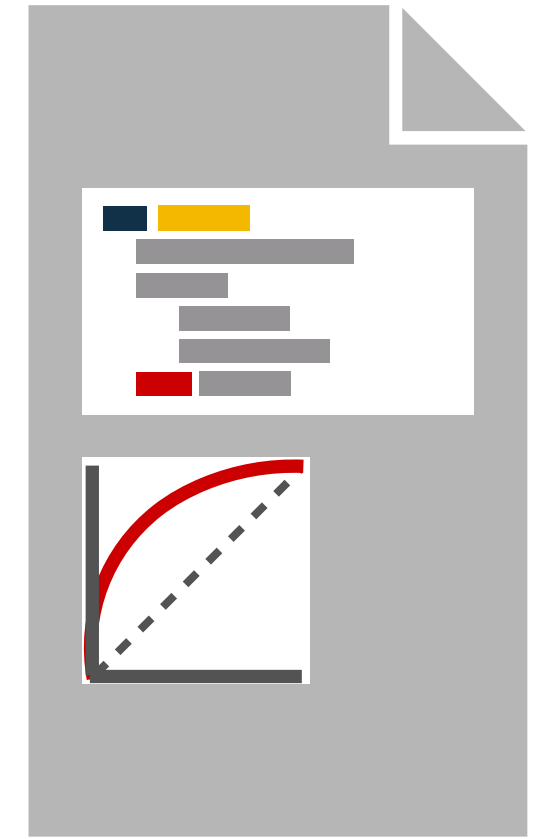
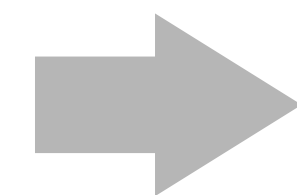


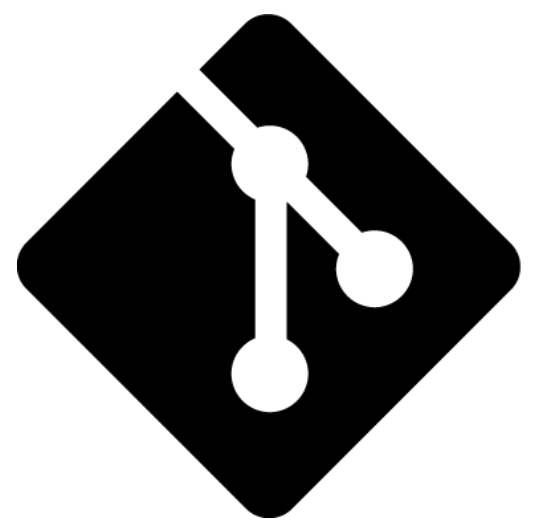
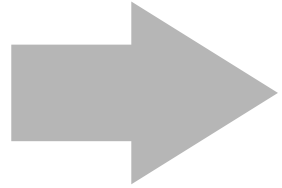


**git**

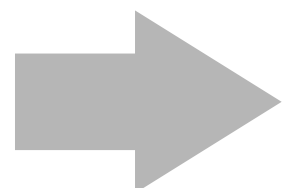


**git**

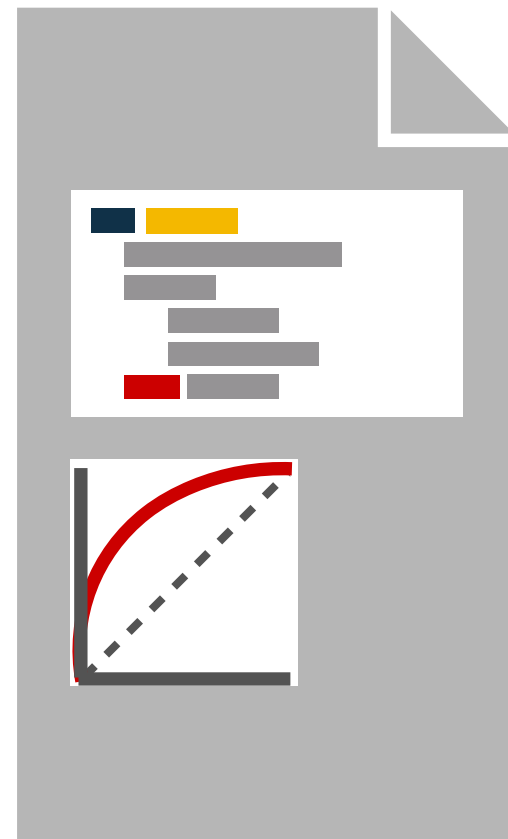




**git**







`requirements.txt`

application code

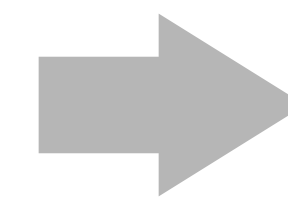
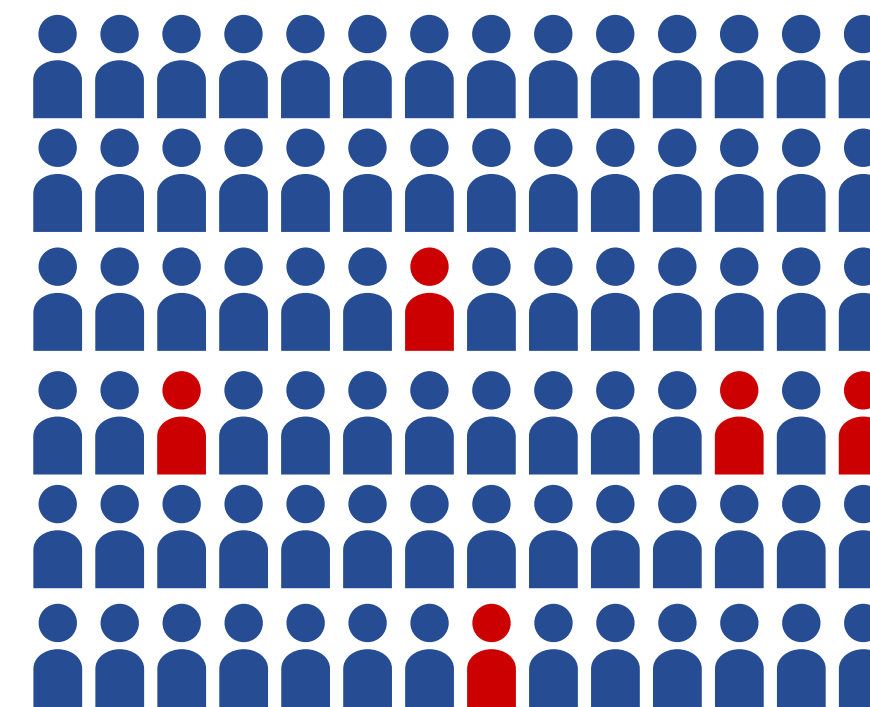
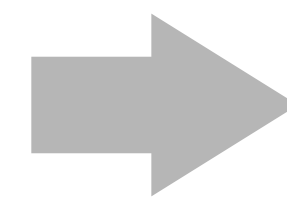
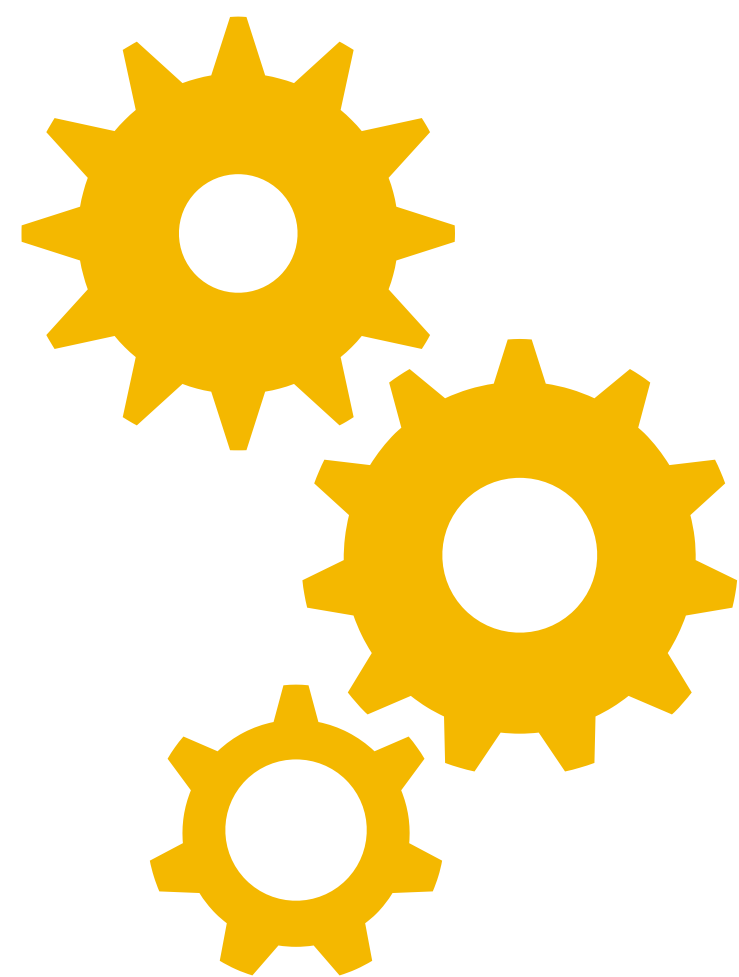
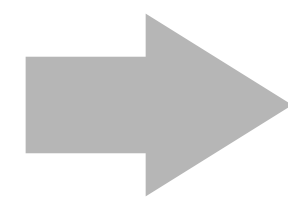
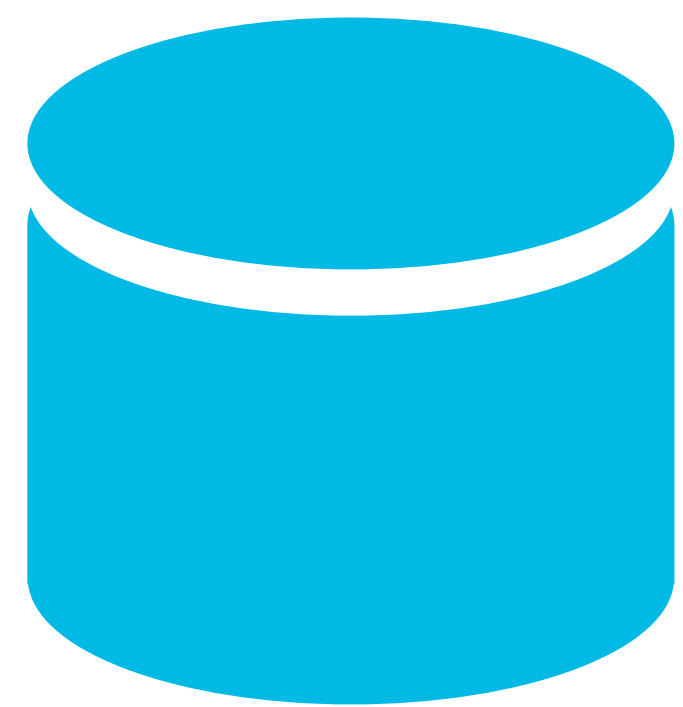
configuration and  
installation recipes

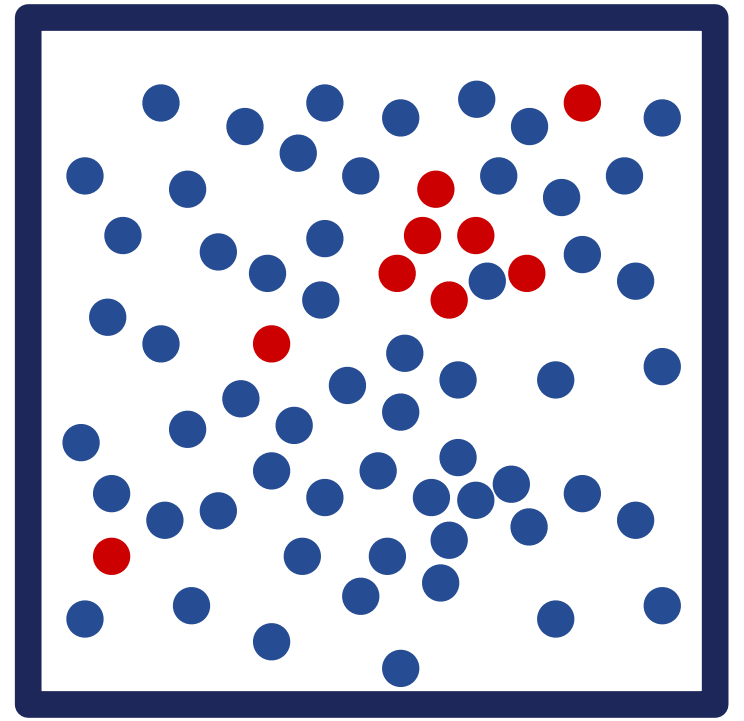
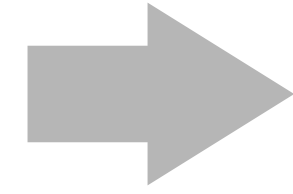
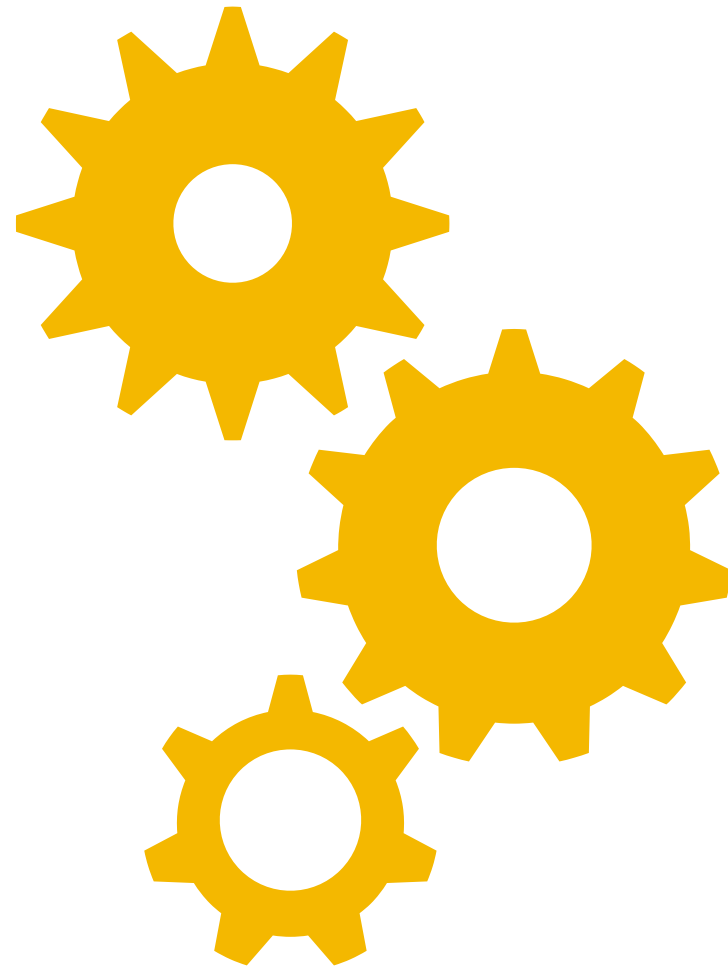
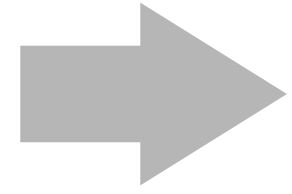
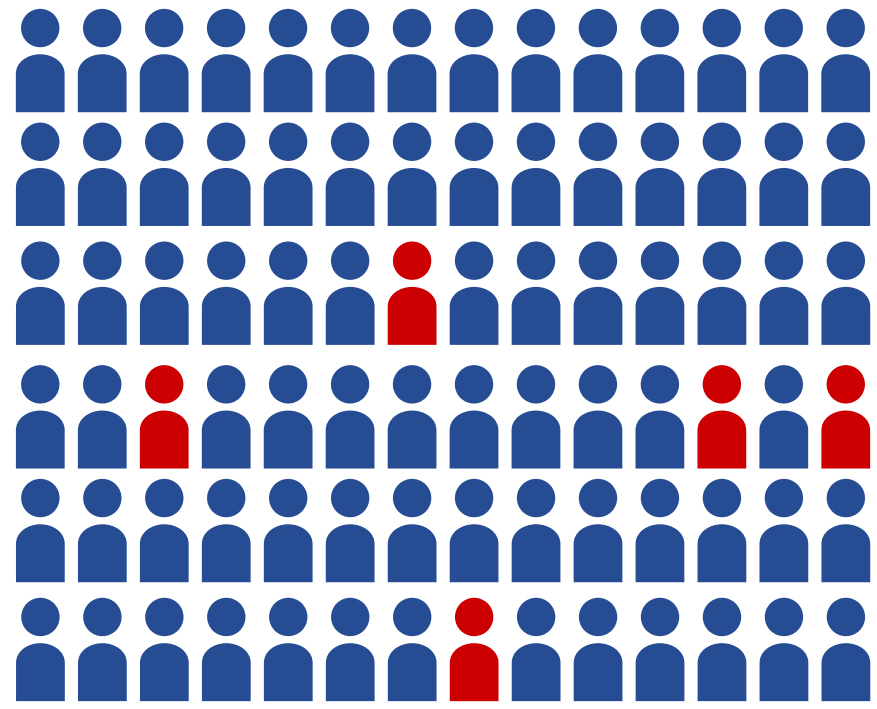
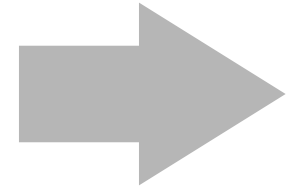
base image

**application code**

**configuration and  
installation recipes**

**base image**





**Making Kubernetes accessible  
to data scientists**

```
FROM centos:centos7
RUN yum install -y \
    python python-pip \
    java java-devel git

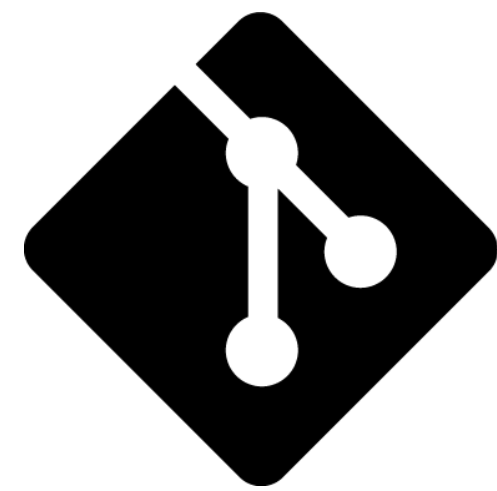
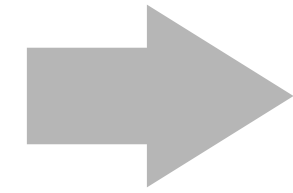
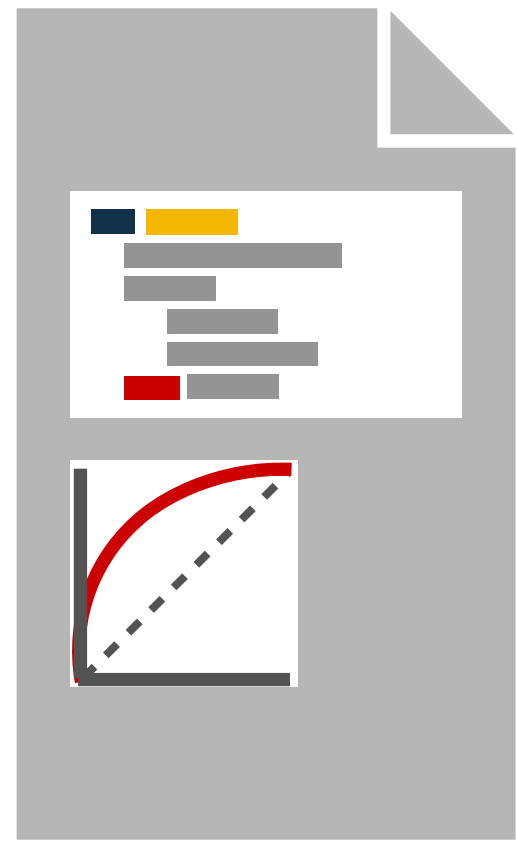
ENTRYPOINT /bin/bash
```



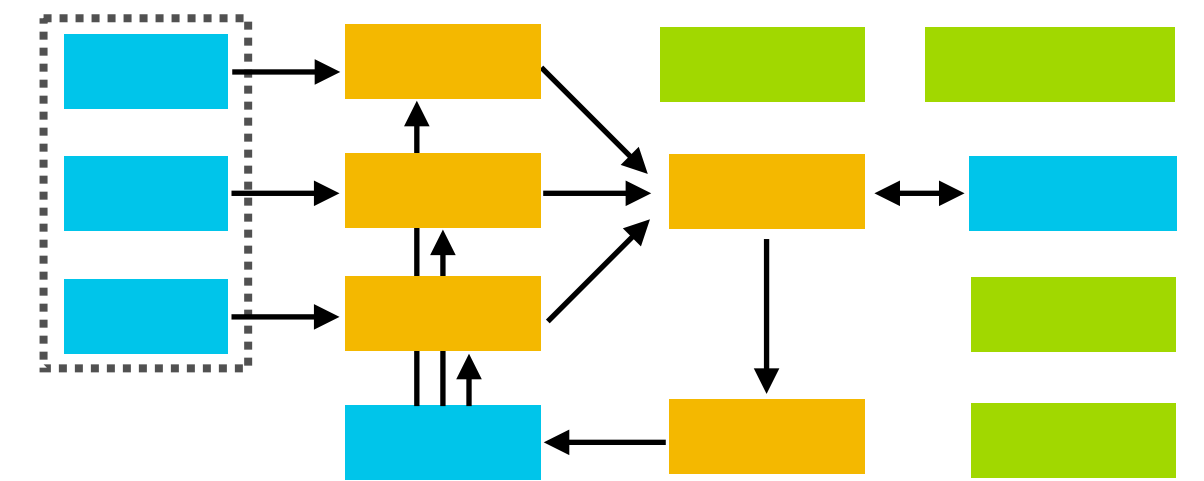
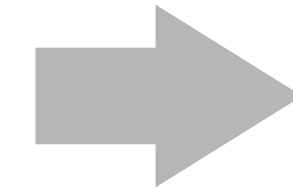
```
FROM centos:centos7
RUN yum install -y \
    python python-pip \
    java java-devel git

ENTRYPOINT /bin/bash
```

# No commitment: [mybinder.org](https://mybinder.org)



**git**









# More flexible: source-to-image

%

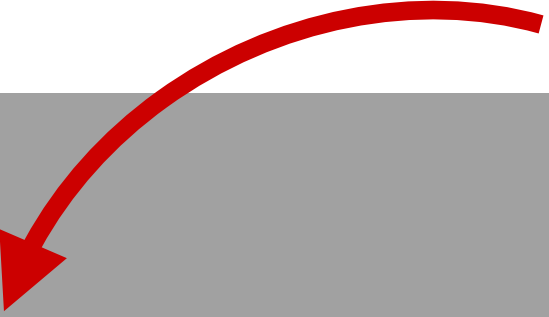
# More flexible: source-to-image

```
% s2i build \  
https://github.com/willb/probabilistic-structures \  
getwarped/s2i-minimal-notebook \  
quay.io/willbenton/probabilistic-structures
```

# More flexible: source-to-image

source repo

```
% s2i build \  
https://github.com/willb/probabilistic-structures \  
getwarped/s2i-minimal-notebook \  
quay.io/willbenton/probabilistic-structures
```



# More flexible: source-to-image

```
% s2i build \  
https://github.com/willb/probabilistic-structures \  
getwarped/s2i-minimal-notebook \  
quay.io/willbenton/probabilistic-structures
```

source repo



builder image



# More flexible: source-to-image

```
% s2i build \  
https://github.com/willb/probabilistic-structures \  
getwarped/s2i-minimal-notebook \  
quay.io/willbenton/probabilistic-structures
```

source repo



builder image



image tag



# More flexible: source-to-image

<https://github.com/openshift/source-to-image>

```
% s2i build \  
https://github.com/willb/probabilistic-structures \  
getwarped/s2i-minimal-notebook \  
quay.io/willbenton/probabilistic-structures
```

source repo



builder image



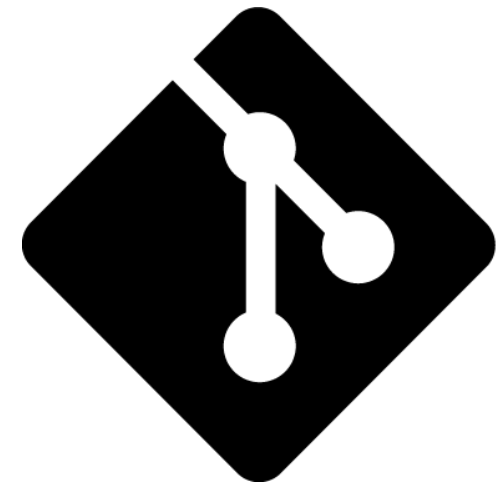
image tag



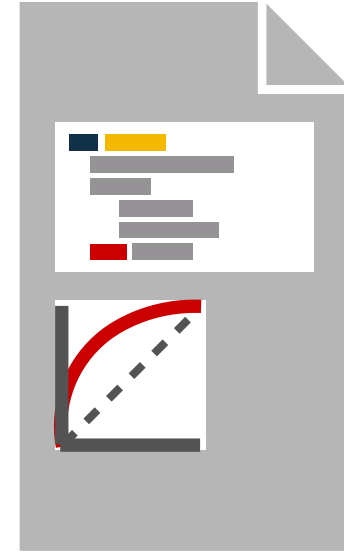
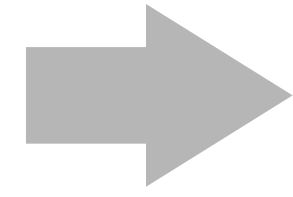


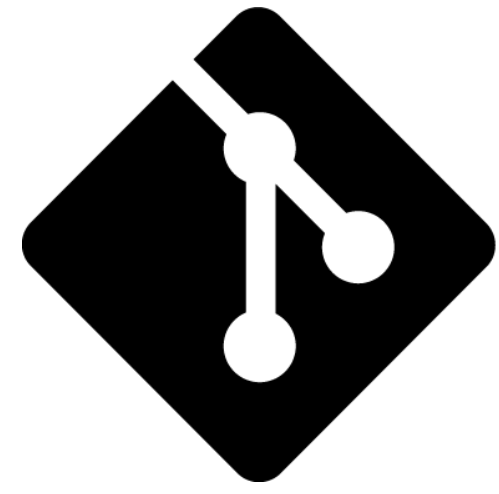




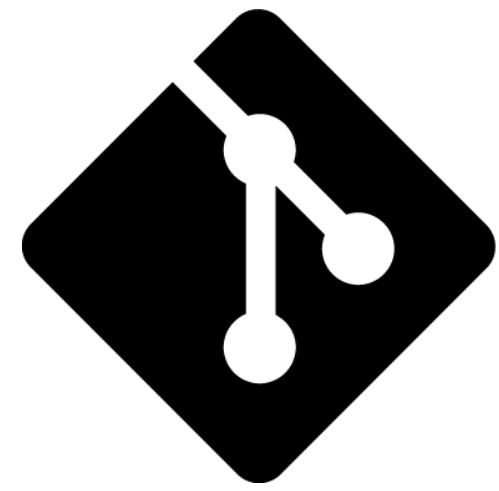
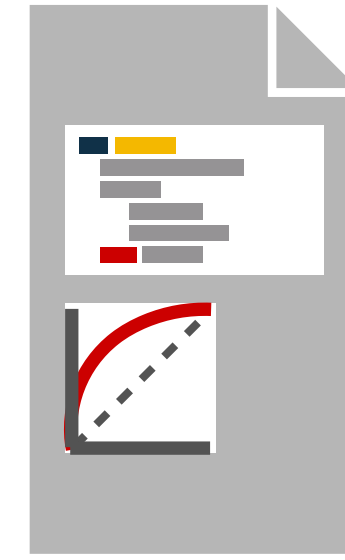
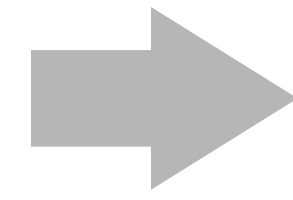


**git**

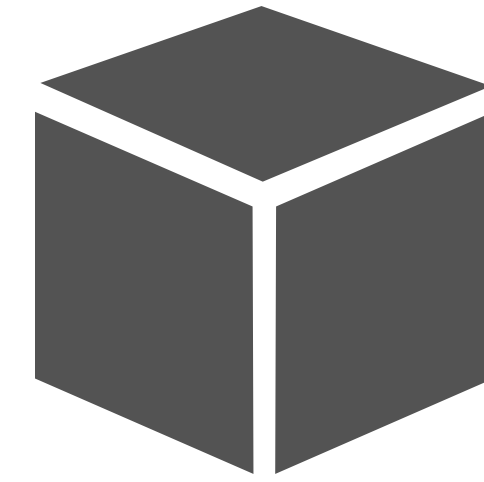
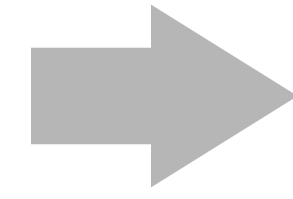


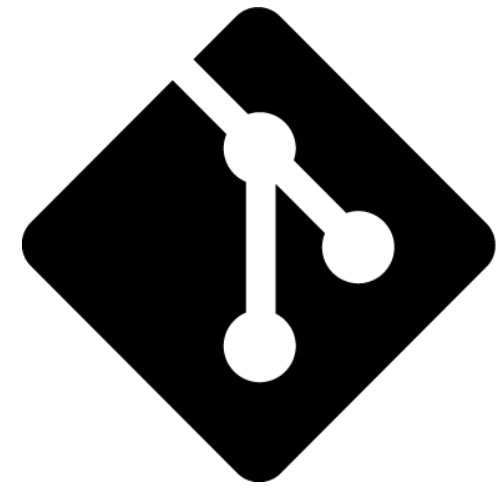


**git**

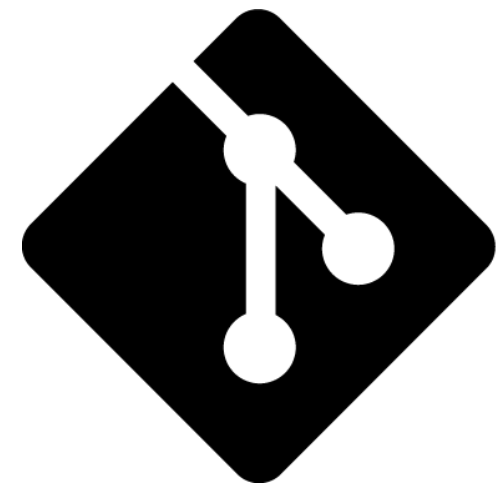
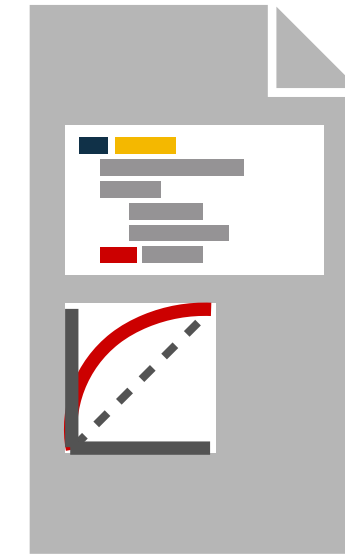
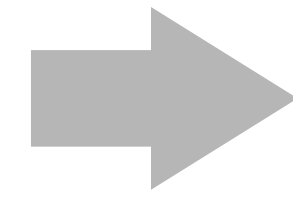


**git**

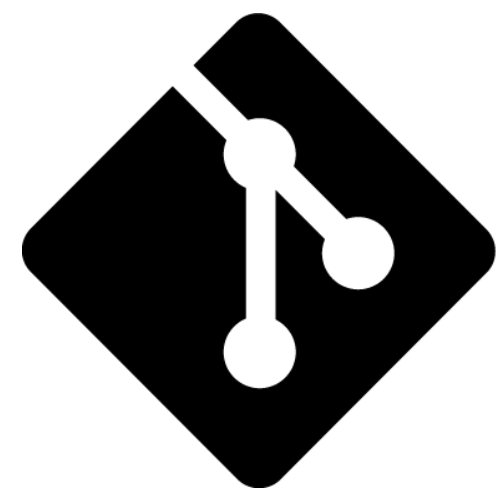
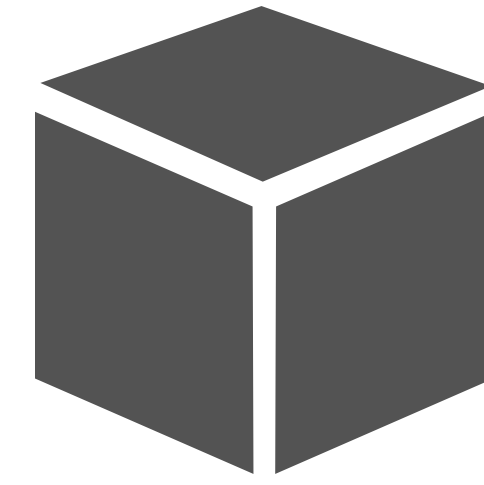
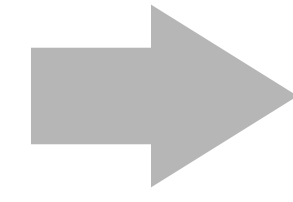




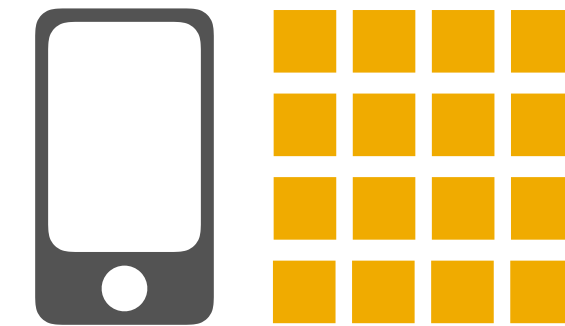
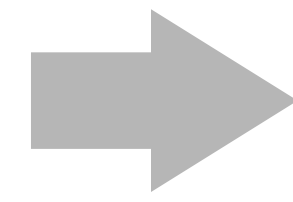
**git**

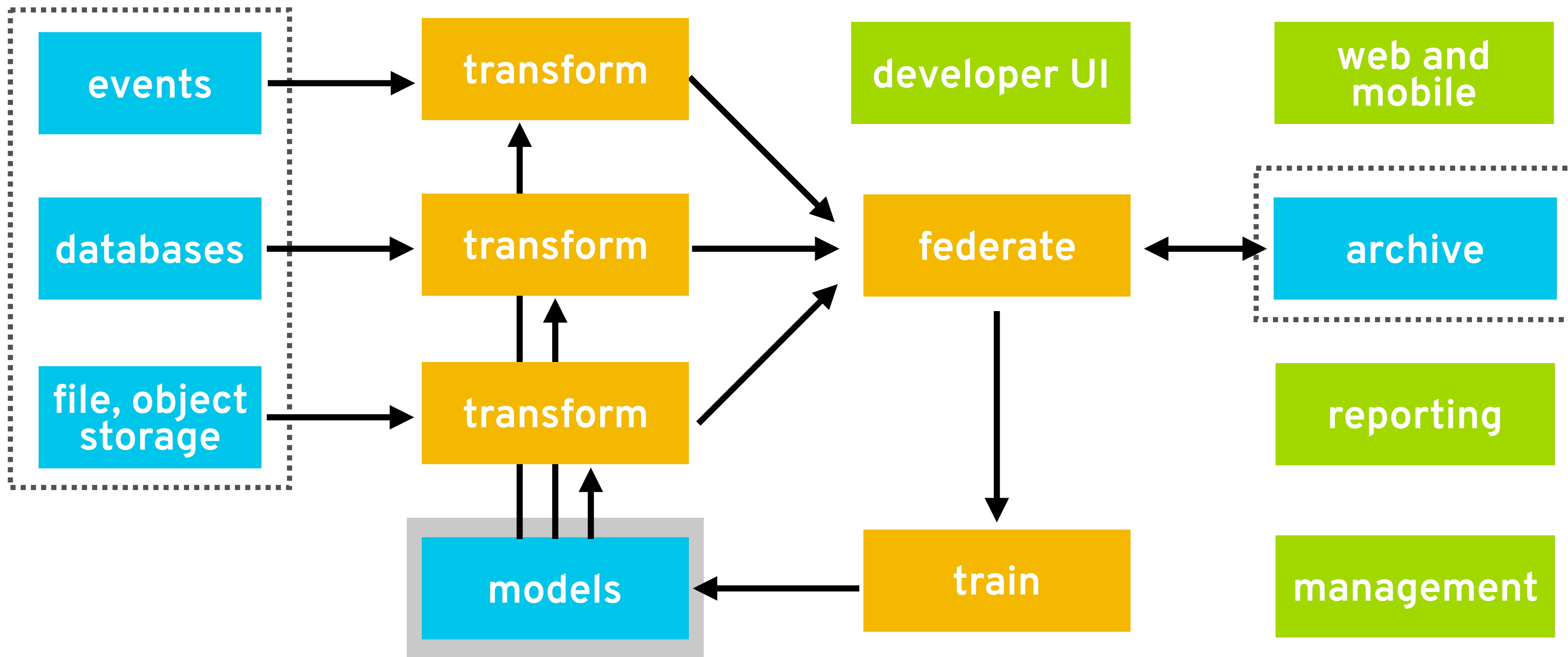


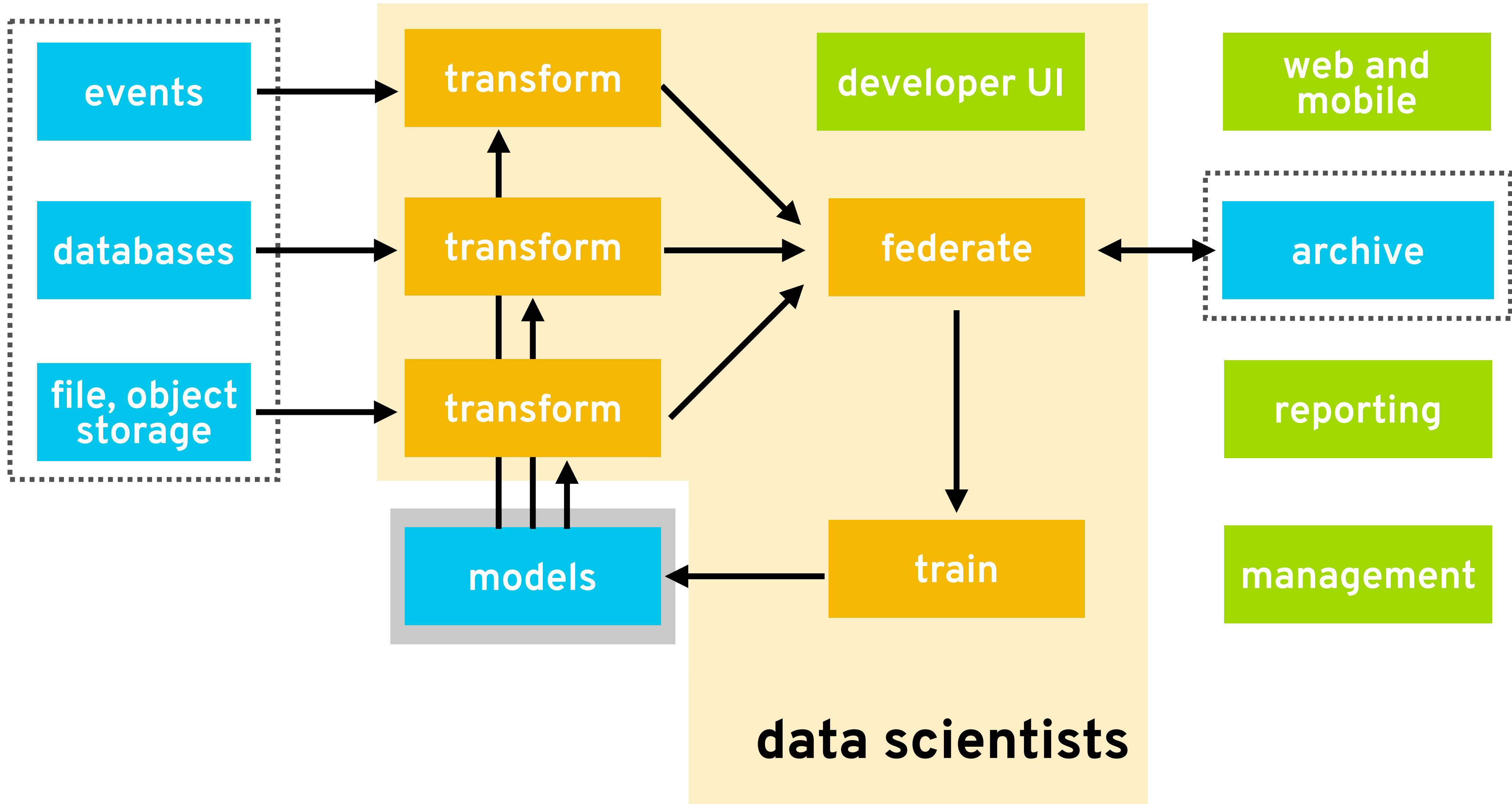
**git**



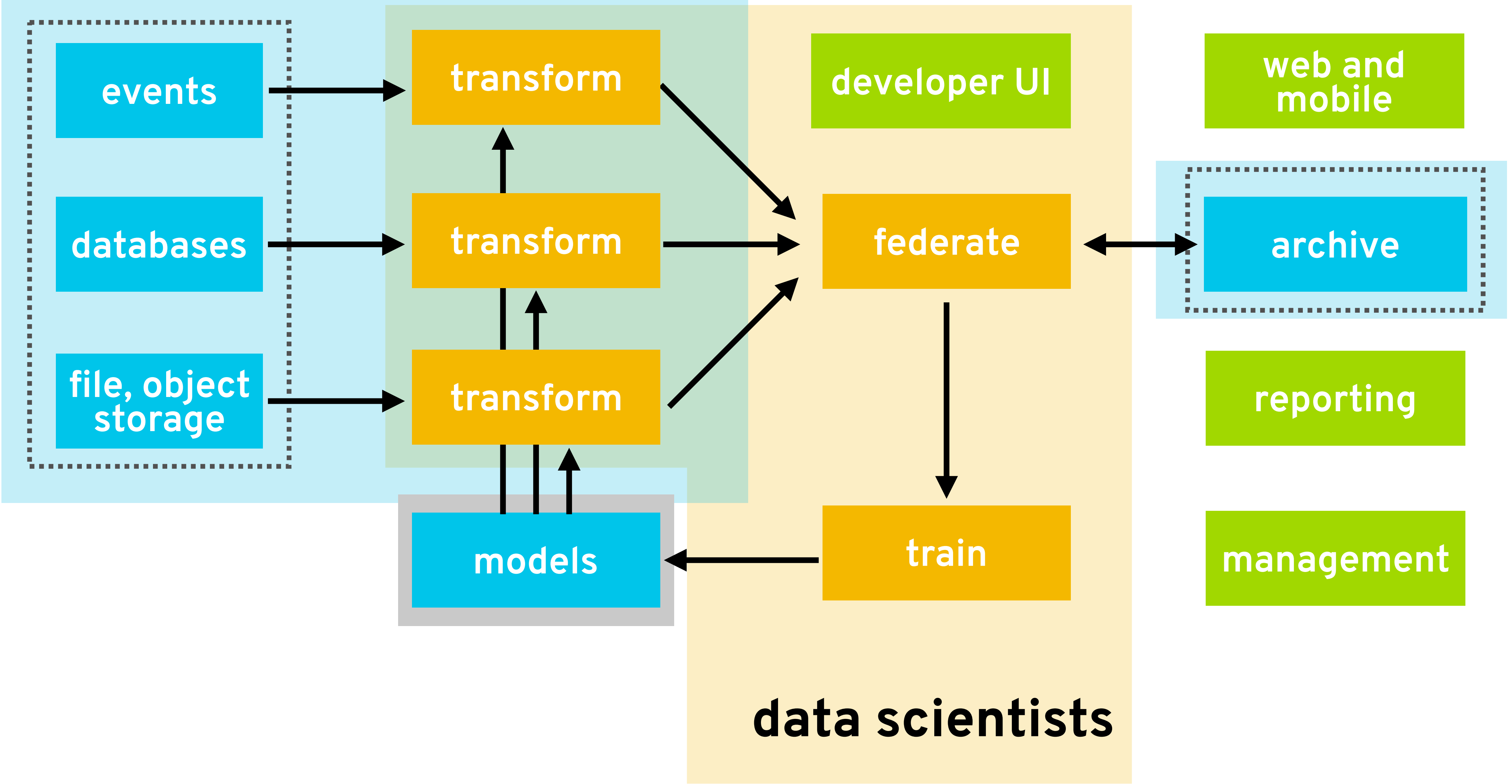
**git**



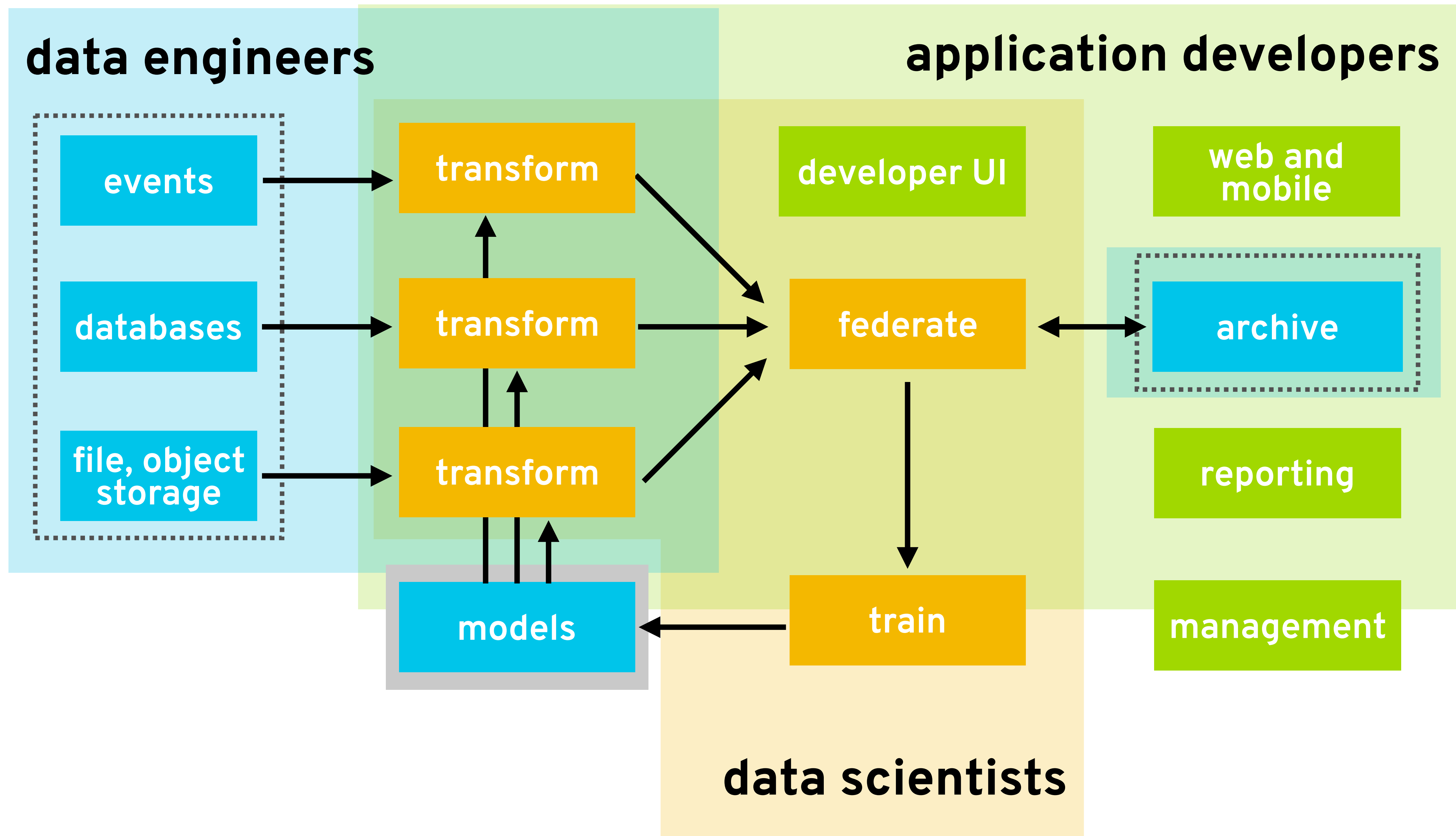




# data engineers

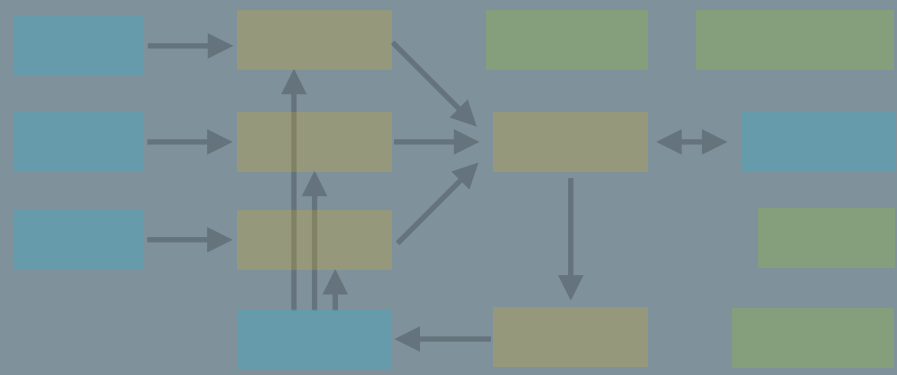




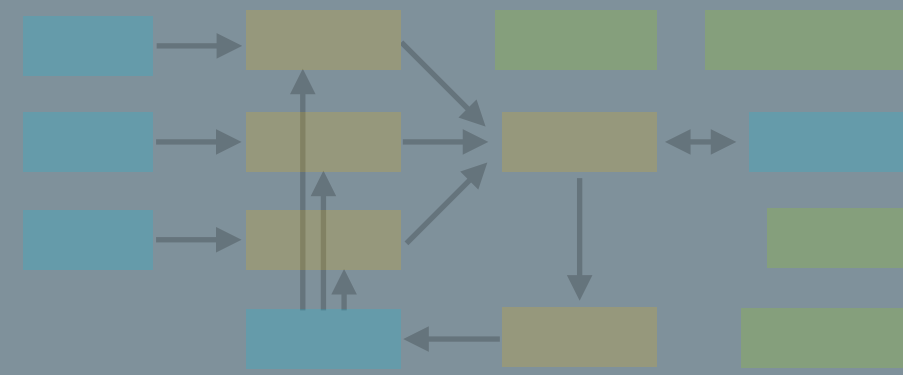


# Resource manager

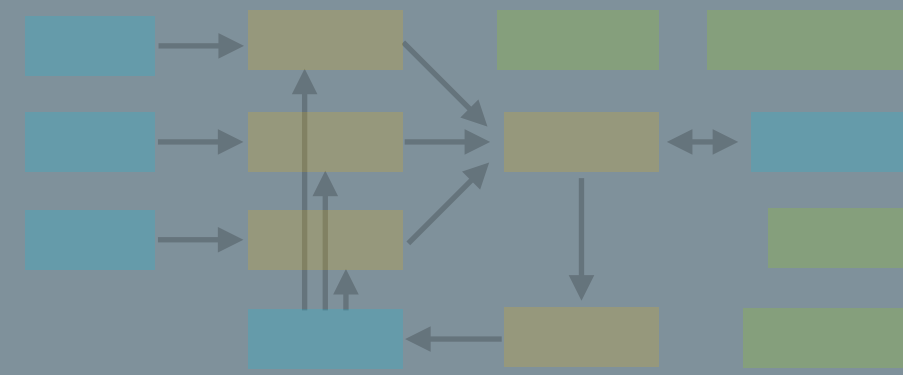
app 1



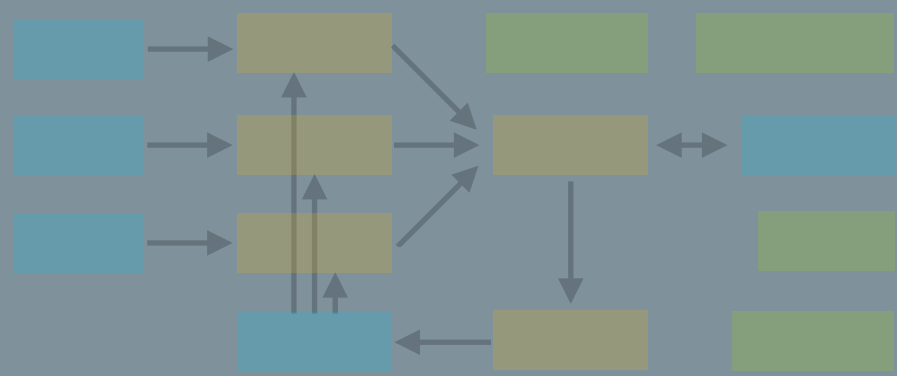
app 2



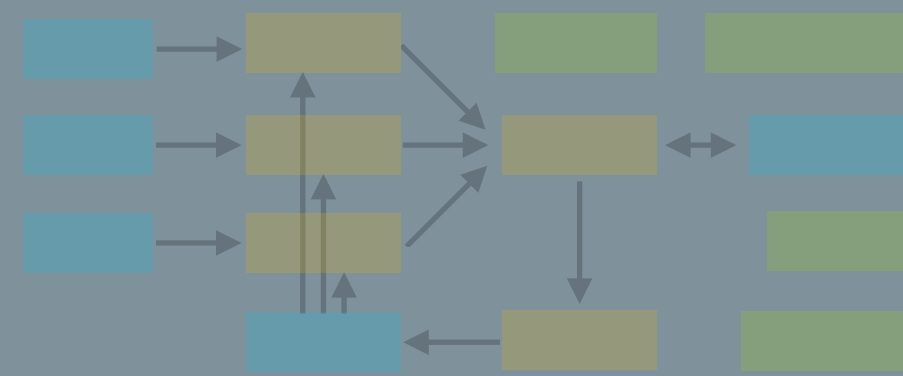
app 3



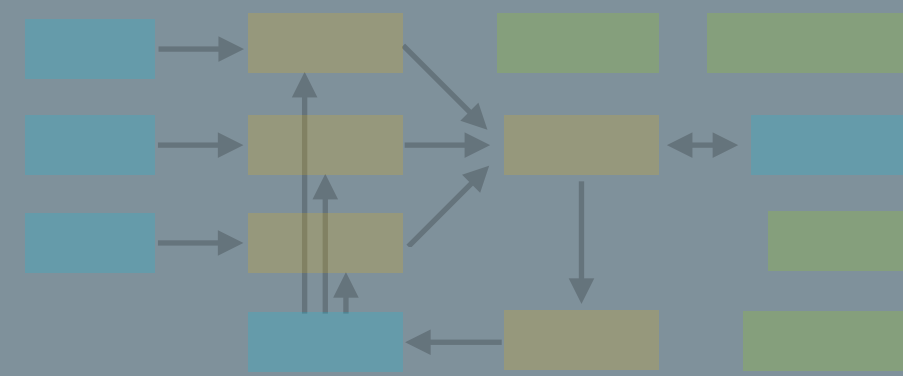
app 4



app 5



app 6

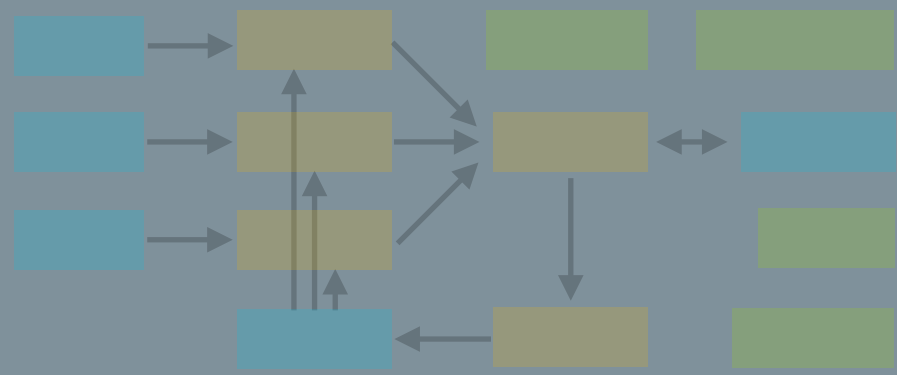


Shared FS /  
object store

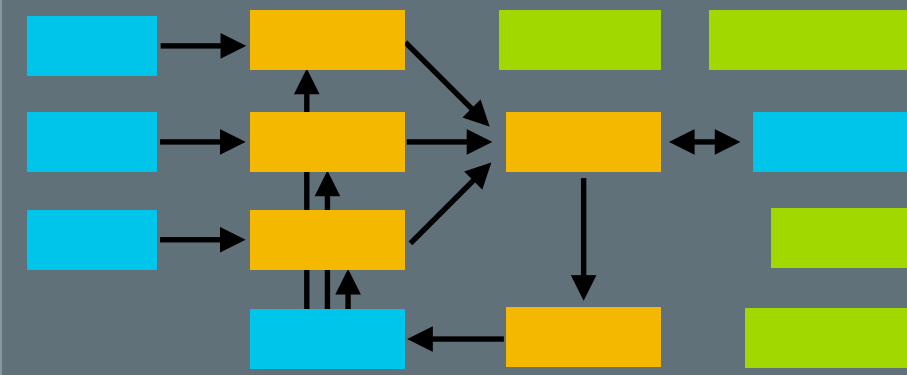
Databases

# Resource manager

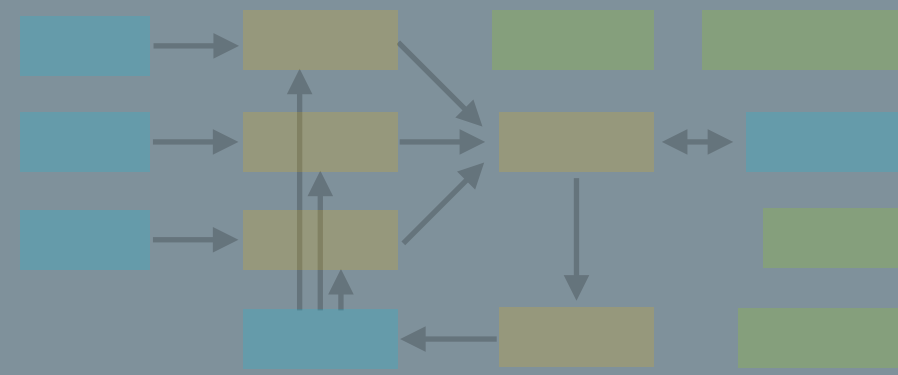
app 1



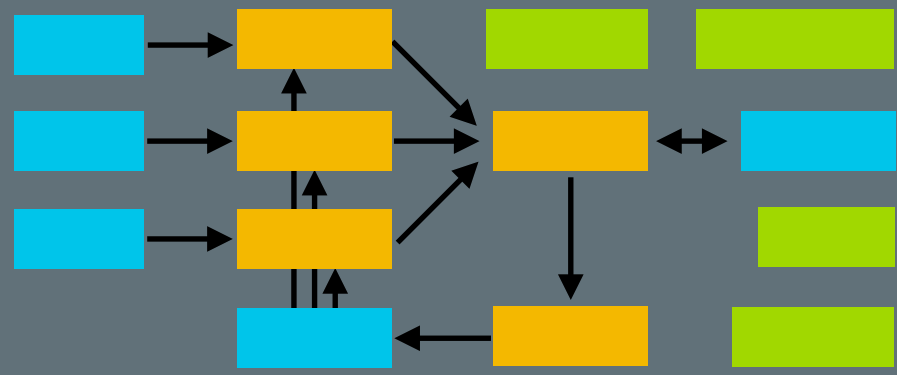
app 2



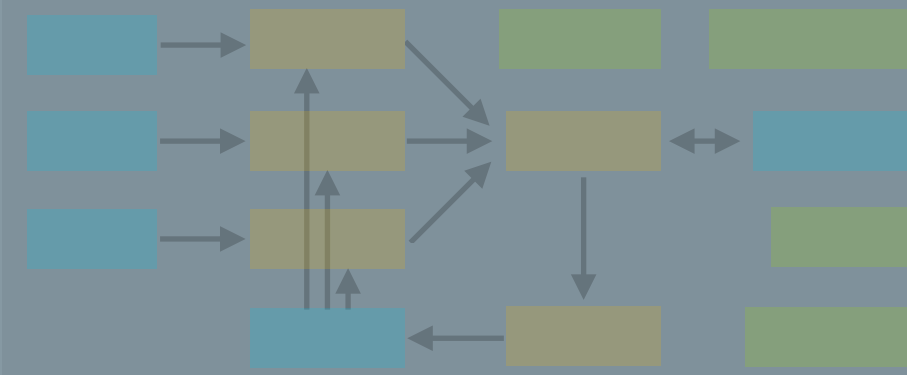
app 3



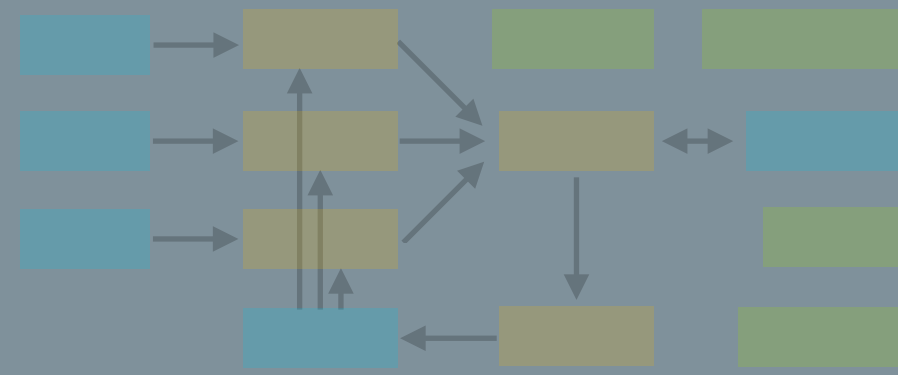
app 4



app 5



app 6



Shared FS /  
object store

Databases

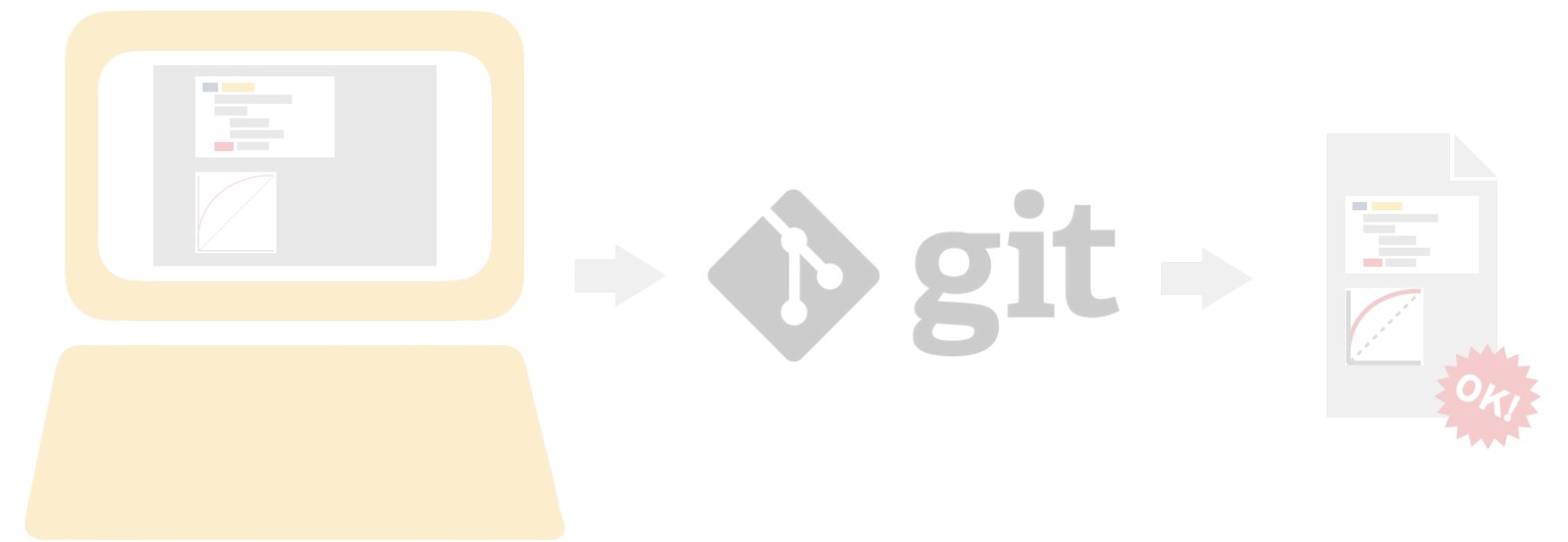
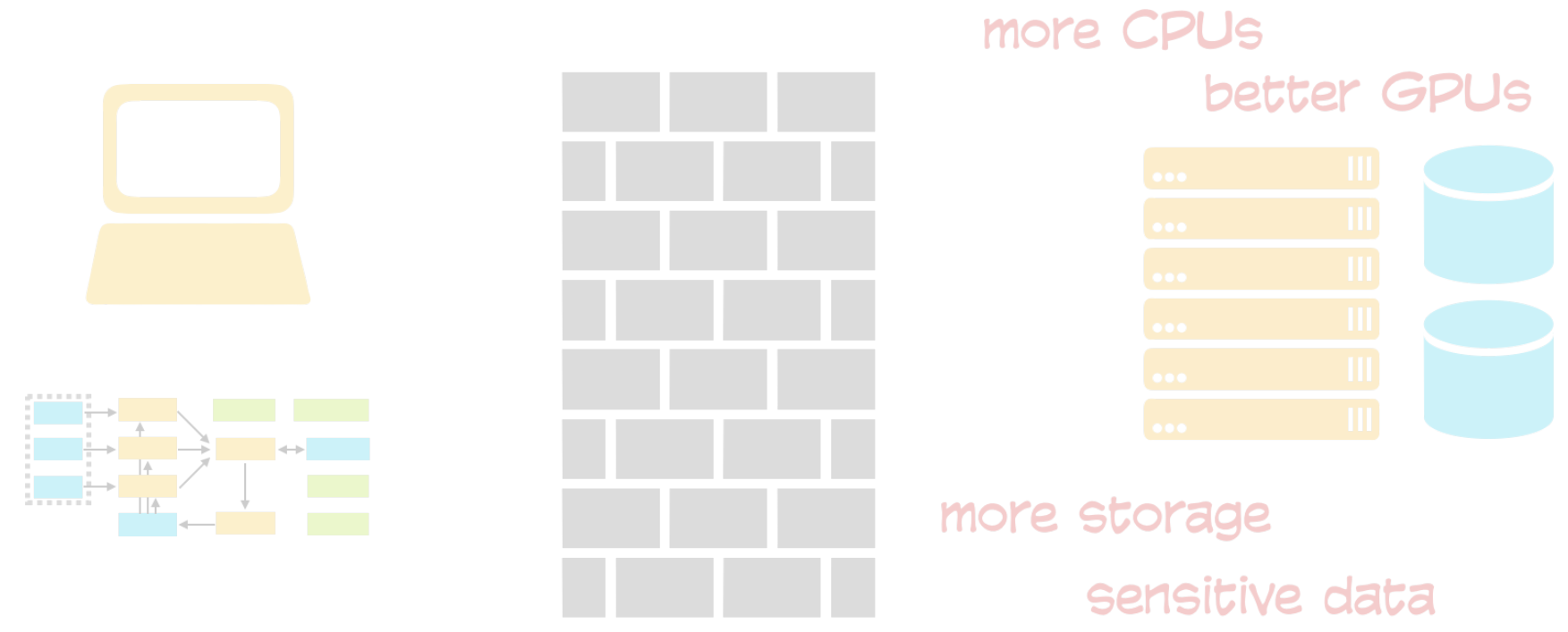
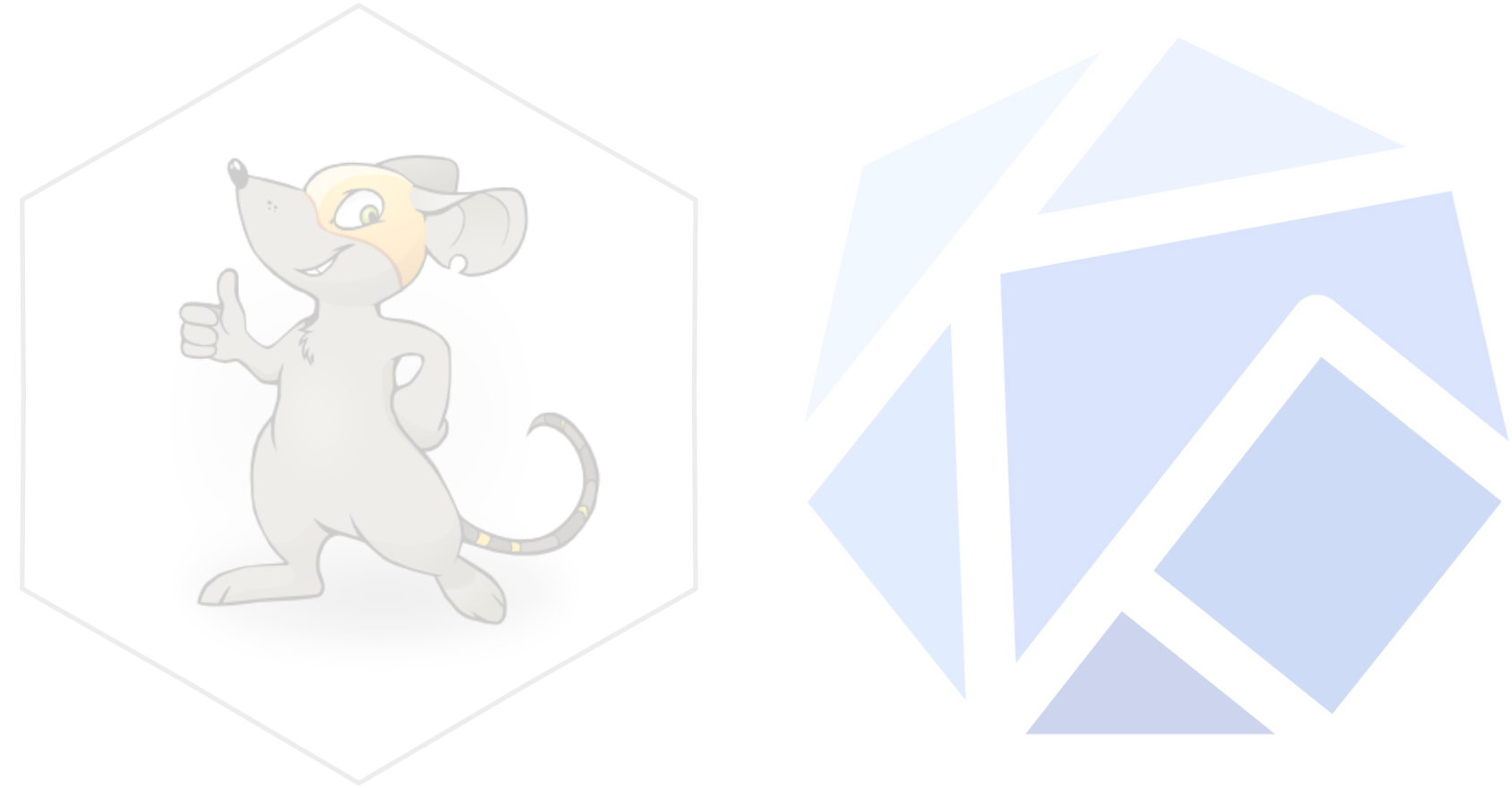
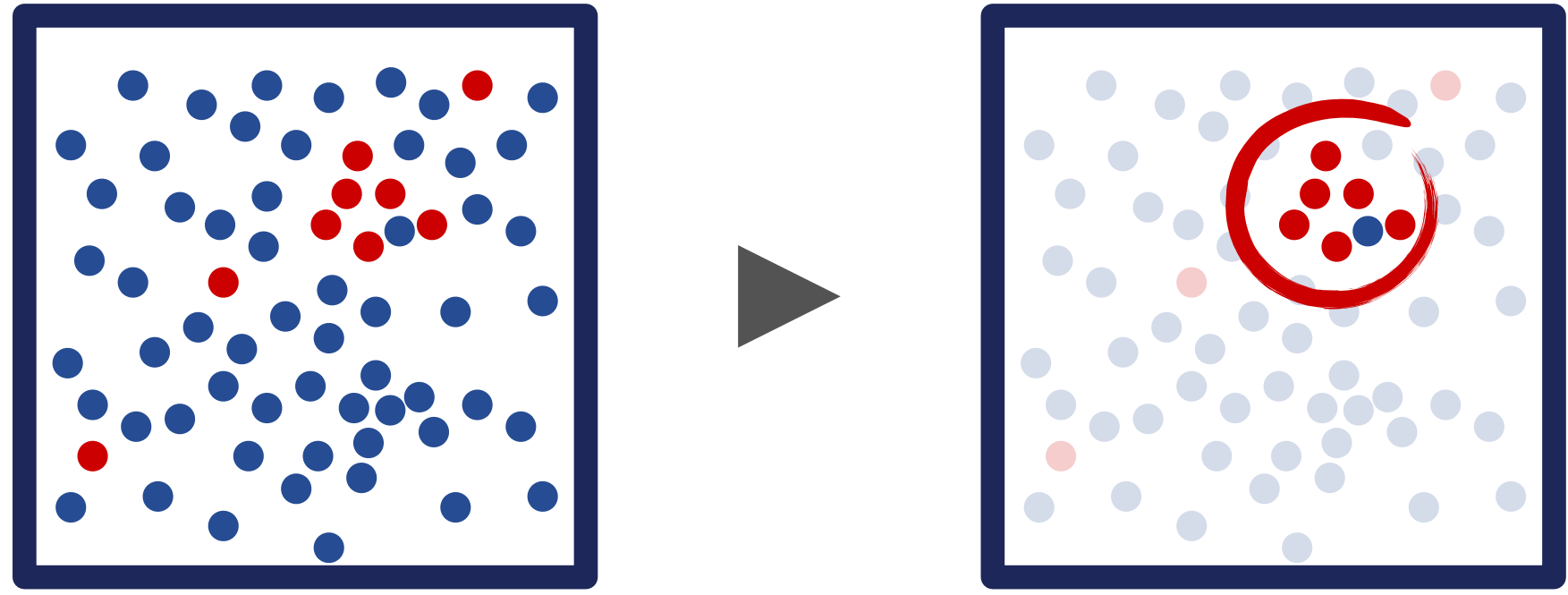


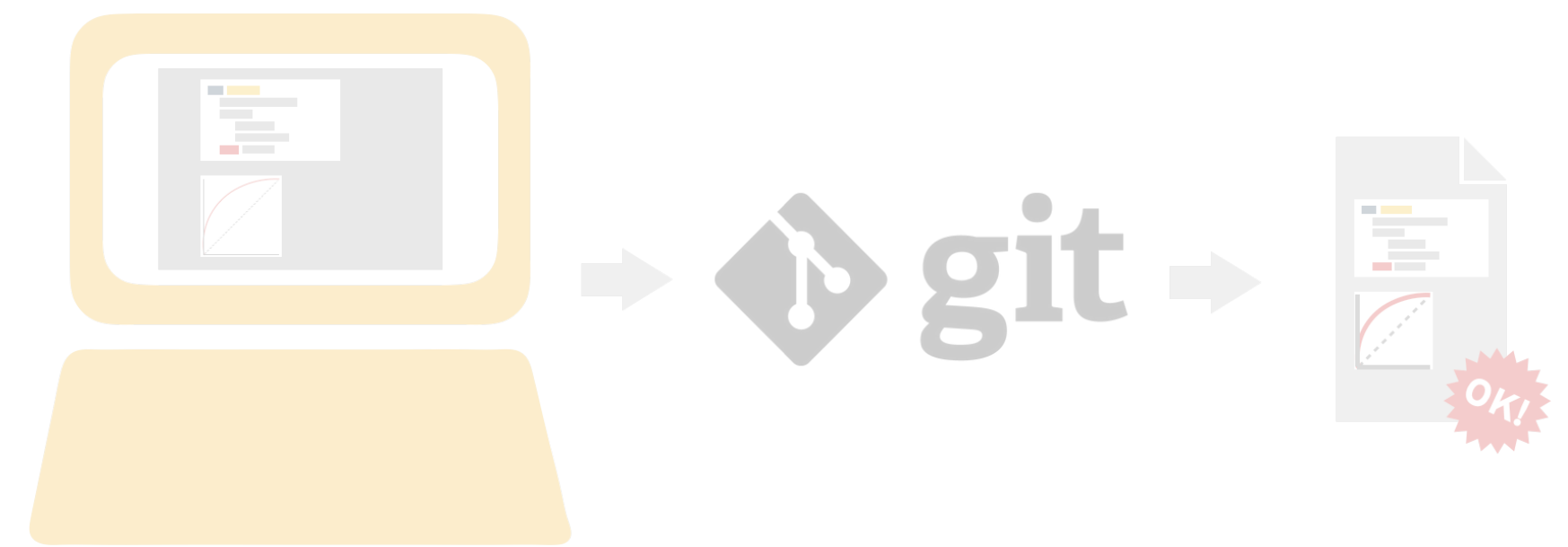
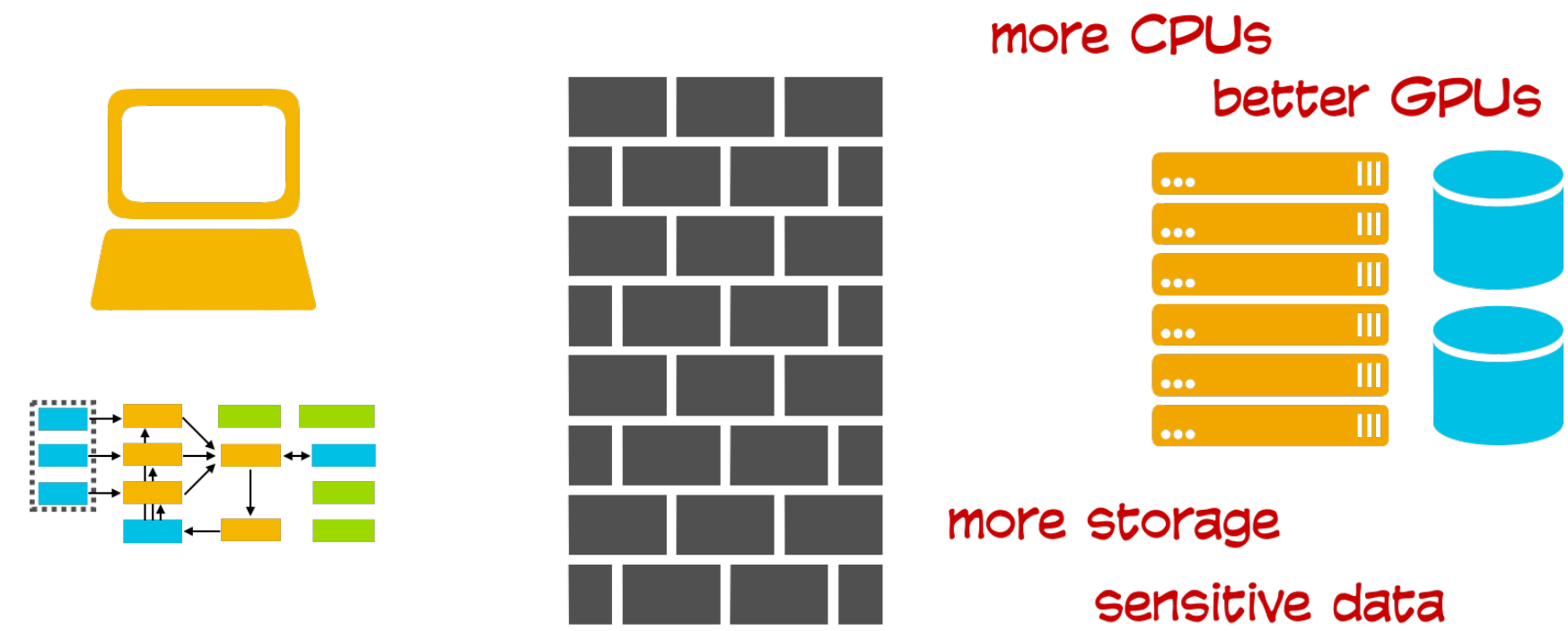
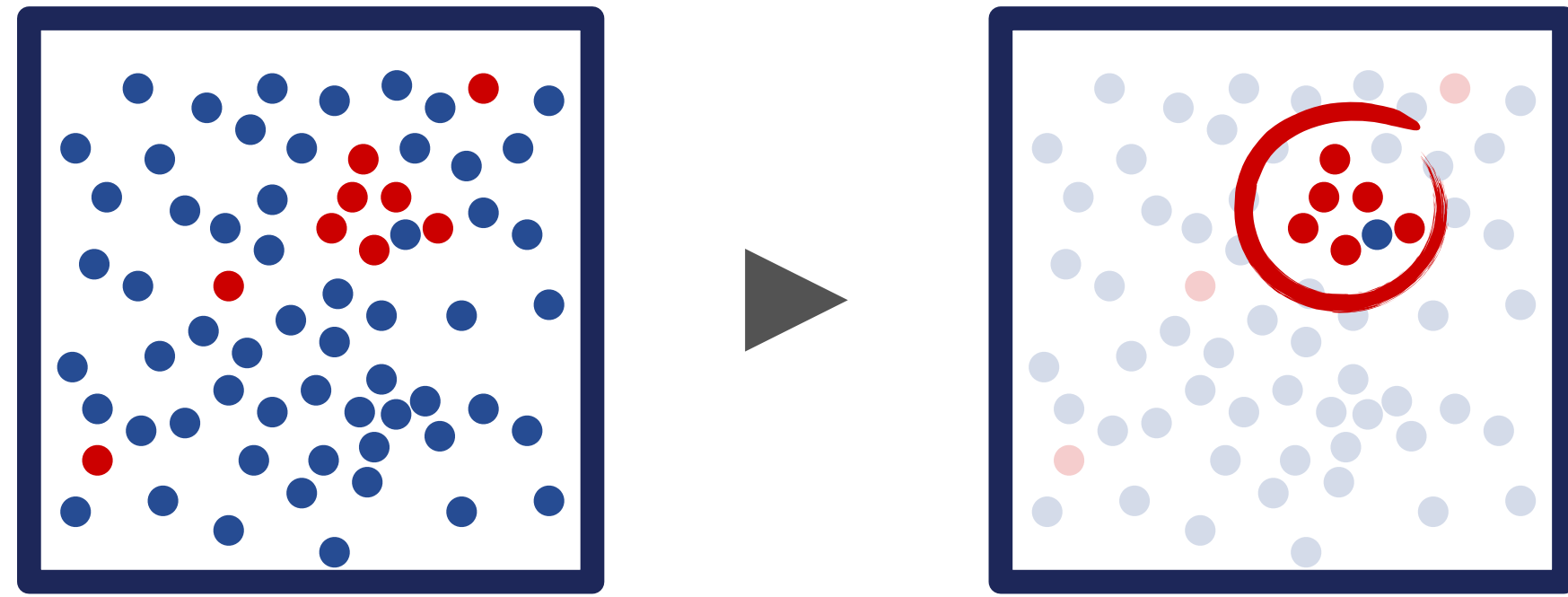
**[radanalytics.io](https://radanalytics.io)**

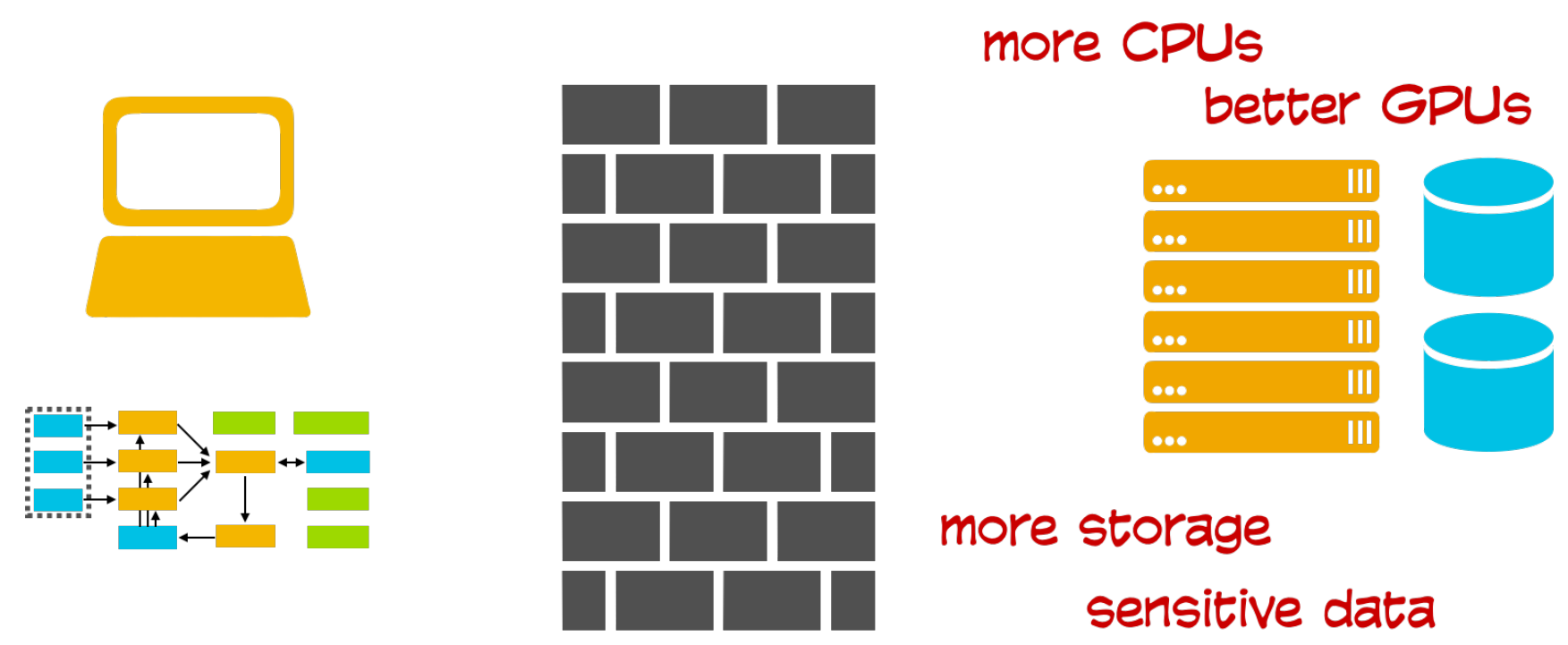
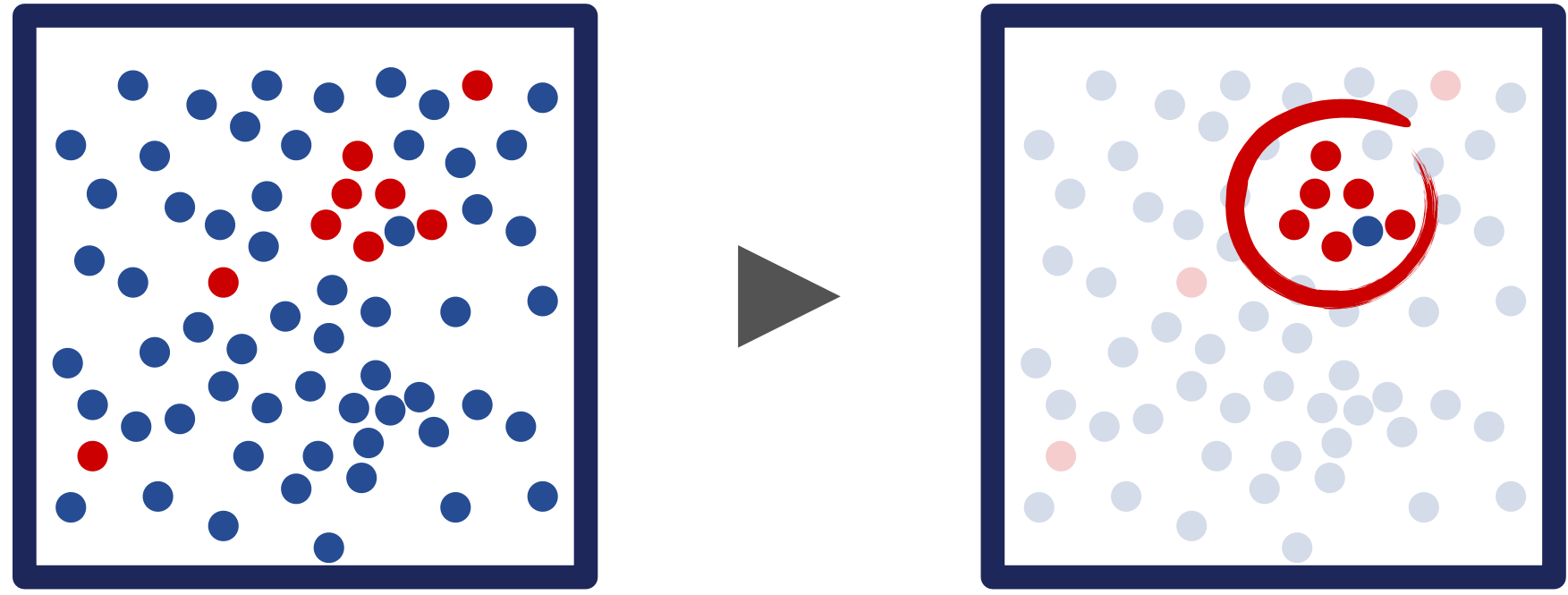


**[kubeflow.org](https://kubeflow.org)**

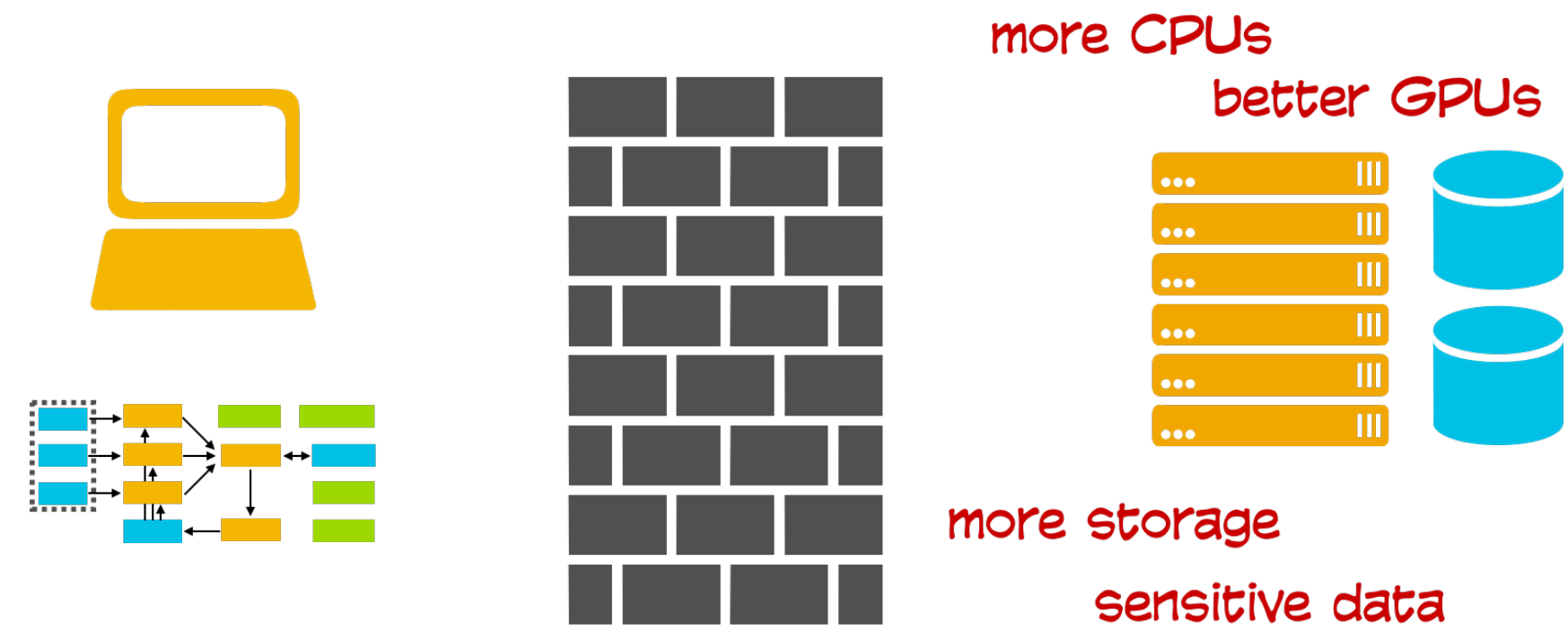
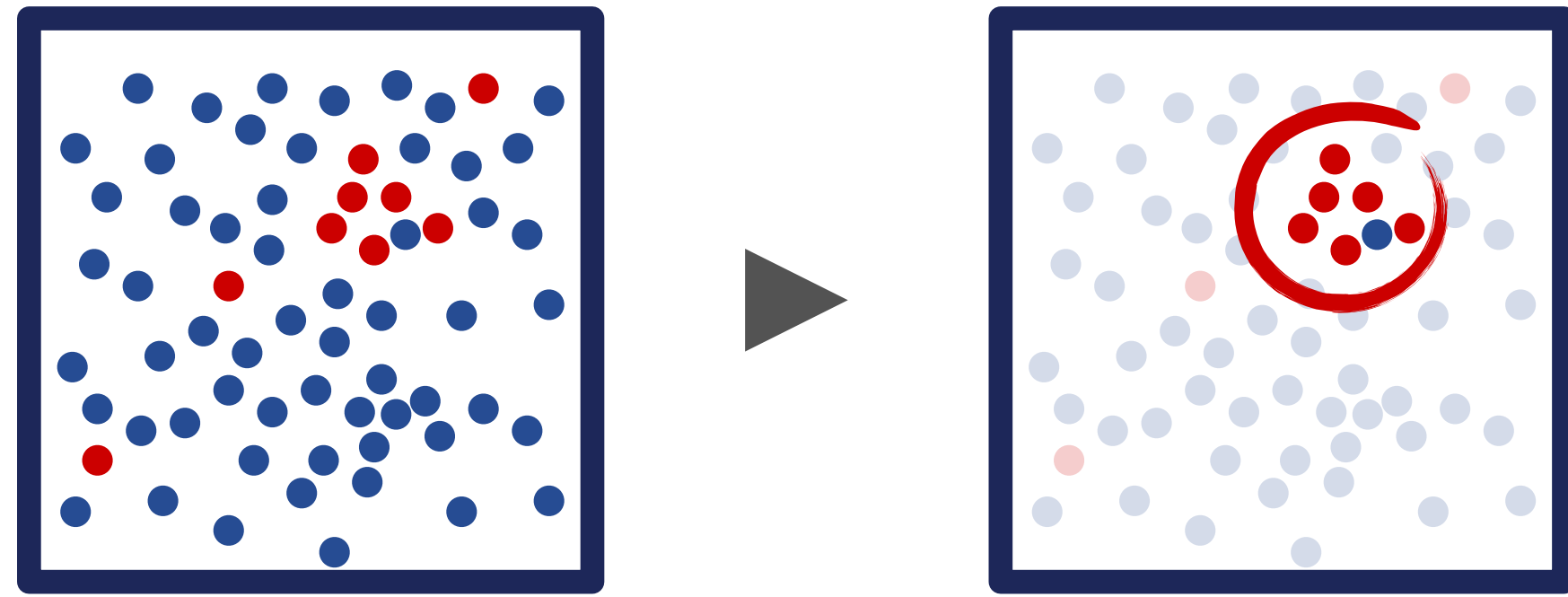
**What did we talk about today?**

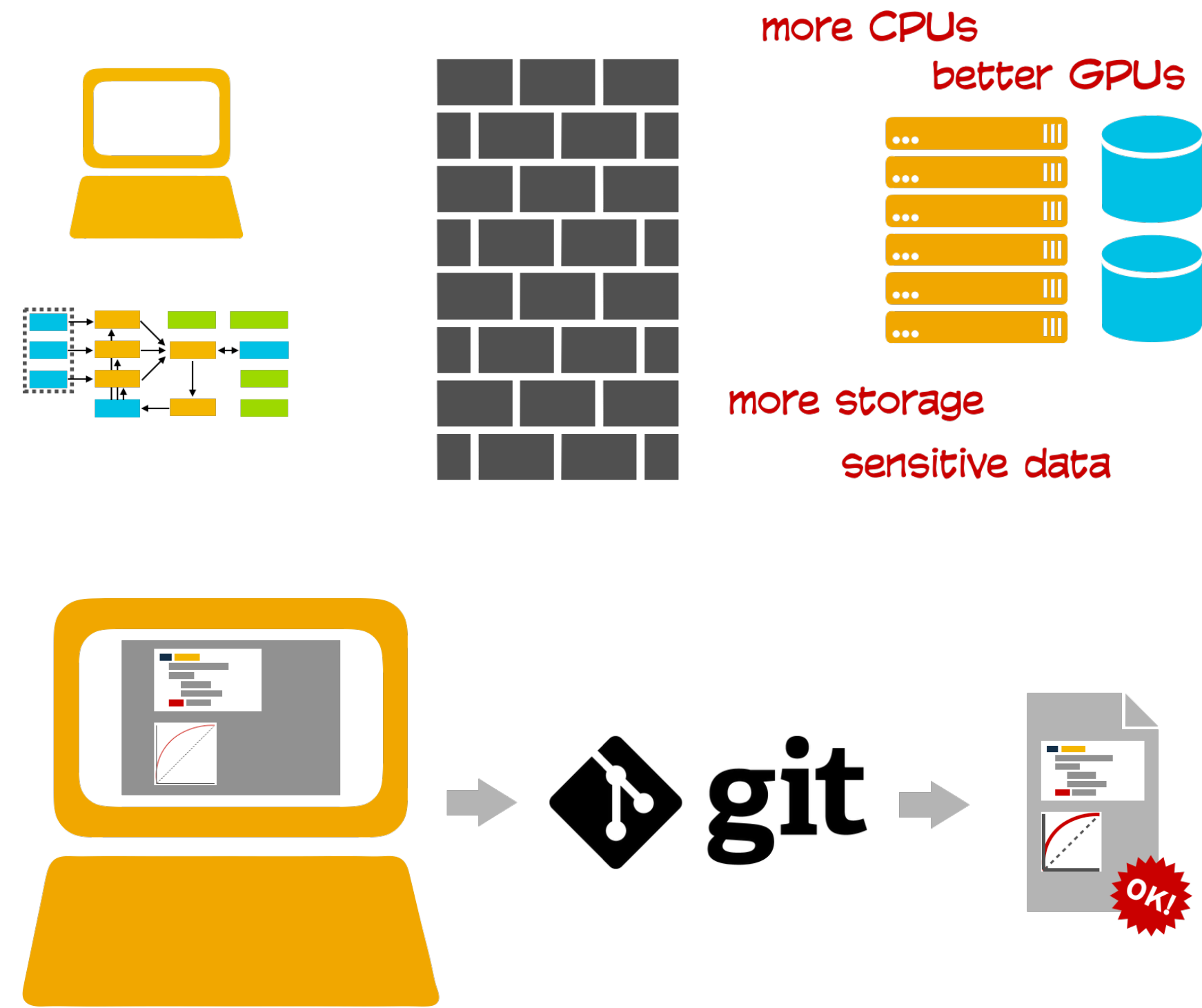
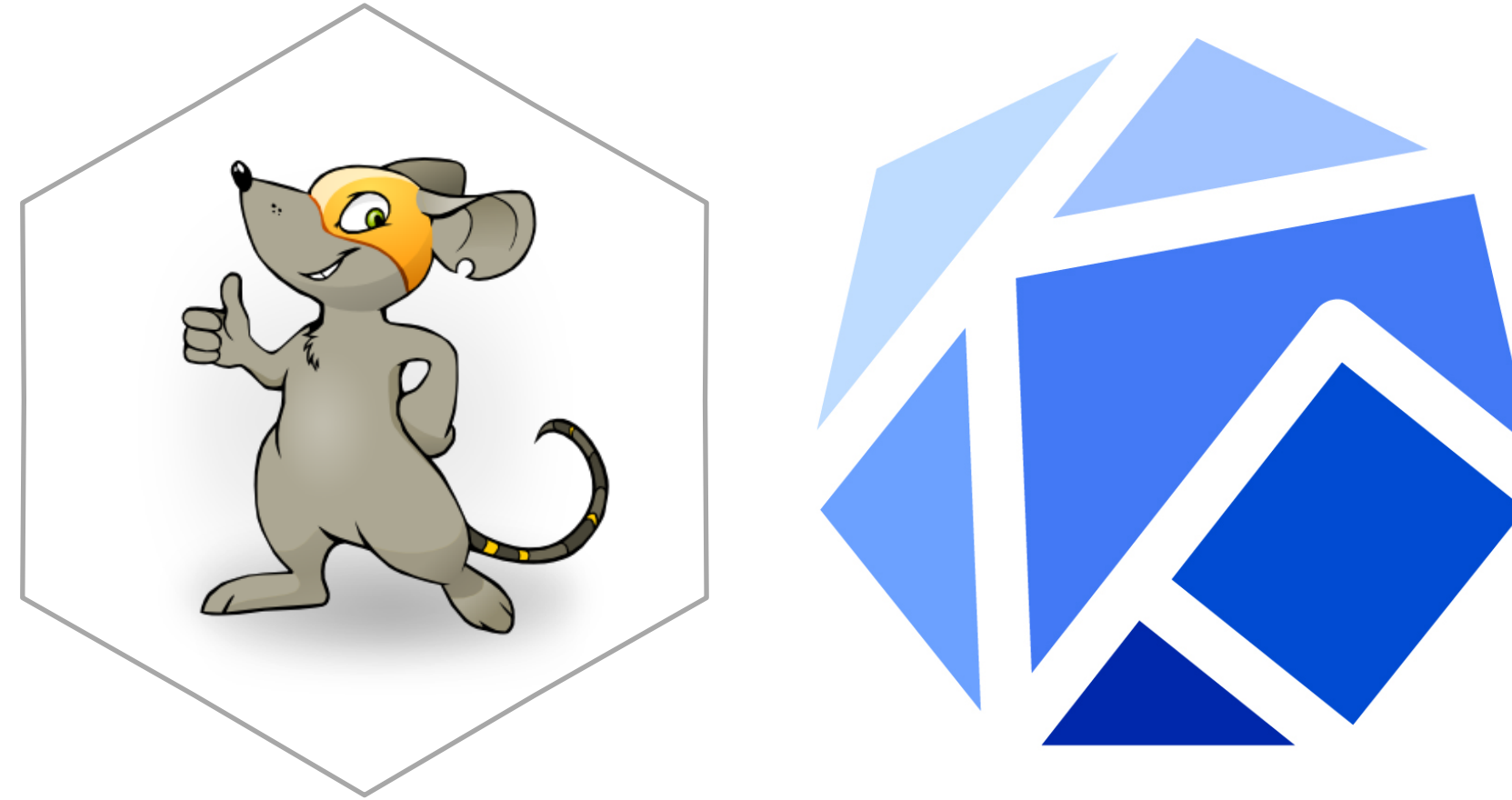
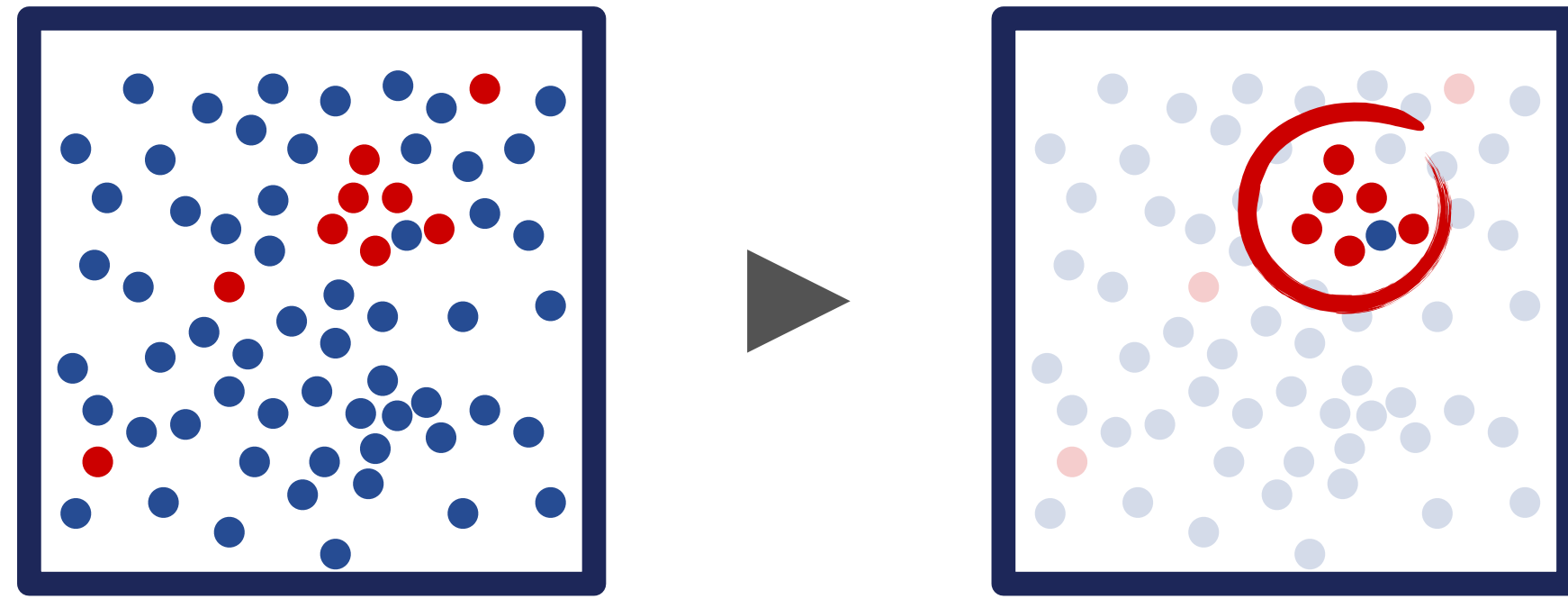












# THANKS!

sophie@redhat.com • @sophwats  
willb@redhat.com • @willb