



**SIGGRAPH** THINK  
BEYOND  
2020 S2020.SIGGRAPH.ORG

---

**SPATIOTEMPORAL RESERVOIR RESAMPLING FOR  
REAL-TIME RAY TRACING WITH DYNAMIC DIRECT LIGHTING**

Benedikt Bitterli<sup>1</sup>  
Peter Shirley<sup>2</sup>

Chris Wyman<sup>2</sup>  
Aaron Lefohn<sup>2</sup>

Matt Pharr<sup>2</sup>  
Wojciech Jarosz<sup>1</sup>

<sup>1</sup>Dartmouth College

<sup>2</sup>NVIDIA



# Goals

# Goals

- Focus on (but not limited to) *direct lighting*

# Goals

- Focus on (but not limited to) *direct lighting*
- Goal:
  - 1'000'000+ dynamic area light sources
  - Correct shadowing and BRDF
  - Large scenes with complex visibility



# Direct Lighting is hard

## Bayesian online regression for adaptive direct illumination sampling, Vévoda et al. 2018

## The Design and Evolution of Disney's Hyperion Renderer, Burley et al. 2018

## Importance Sampling of Many Lights with Adaptive Tree Splitting, Estevez and Kulla 2018

### The Design and Evolution of Disney's Hyperion Renderer

BRENT BURLEY, DAVID ADLER, MATT JEN-YUAN CHIANG, HANK DRISKILL, RALF HABEL, PATRICK KELLY, PETER KUTZ, YINING KARL LI, and DANIEL TEECE, Walt Disney Animation Studios



Fig. 1. Production frames from *Rig Hero 6* (upper left), *Zootopia* (upper right), *Moana* (bottom left), and *Olaf's Frozen Adventure* (bottom right), all rendered using Disney's Hyperion Renderer.

Walt Disney Animation Studios has transitioned to path-traced global illumination as part of a progression of brute-force physically based rendering in the name of artist efficiency. To achieve this without compromising our geometric or shading complexity, we built our Hyperion renderer based on a novel architecture that extracts traversal and shading coherence from large, sorted ray batches. In this article, we describe our architecture and discuss our design decisions. We also explain how we are able to provide artistic control in a physically based renderer, and we demonstrate through case studies how we have benefited from having a proprietary renderer that can evolve with production needs.

CCS Concepts • **Computing methodologies** → **Rendering**; **Ray tracing**; Additional Key Words and Phrases: Production rendering, physically based rendering, path tracing

© 2018 Copyright held by the owner/author(s). Publication rights licensed to ACM. This is the author's version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published in ACM Transactions on Graphics, <https://doi.org/10.1145/3182159>.

**ACM Reference format:**  
Brent Burley, David Adler, Matt Jen-Yuan Chiang, Hank Driskill, Ralf Habel, Patrick Kelly, Peter Kutz, Yining Karl Li, and Daniel Teece. 2018. The Design and Evolution of Disney's Hyperion Renderer. *ACM Trans. Graph.* 37, 3, Article 33 (July 2018), 22 pages. <https://doi.org/10.1145/3182159>

### 1 INTRODUCTION

Since the early 1990s, rendering of computer graphics (CG) imagery at Walt Disney Animation Studios had been accomplished using the Reyes algorithm (Cook et al. 1987) in Pixar's RenderMan. More recently, ray-traced global illumination (GI) promised artists significant productivity gains by providing more immediate feedback during rendering and removing the significant data management burden associated with shadow maps and point clouds. However, initial attempts to render our existing production scenes with ray-traced GI were unsuccessful; incoherent access of texture maps inhibited shading of indirect ray hits, and we had difficulty fitting our scenes in memory as required by existing ray-traced renderers.

To overcome these limitations, we created a new rendering architecture which traces and shades rays in large batches, first sorting each batch for geometric coherence during scene traversal, then sorting ray hits for texture coherence during shading (Eisenacher et al. 2013). This streaming architecture is at the heart of Hyperion, our proprietary renderer, and allows us to render scenes using ray-traced GI without compromising geometric

ACM Transactions on Graphics, Vol. 37, No. 3, Article 33. Publication date: July 2018.

### Bayesian online regression for adaptive direct illumination sampling

PETR VÉVODA<sup>1</sup>, Charles University, Prague and Render Legion, a.s.  
IVO KONDAPANENI<sup>1</sup>, Charles University, Prague  
JAROSLAV KRIVÁNEK<sup>1</sup>, Charles University, Prague and Render Legion, a.s.



Fig. 1. Equal-time comparison (60 s) of path-traced global illumination solutions computed using our learning-based direct illumination sampling method (right) and a baseline sampling method without learning (left). While both methods start off by sampling lights proportionally to rough estimates of their unoccluded contribution, our method progressively incorporates information about their actual contributions, including visibility, dramatically reducing image variance.

Direct illumination calculation is an important component of any physically-based renderer with a substantial impact on the overall performance. We present a novel adaptive solution for unbiased Monte Carlo direct illumination sampling, based on online learning of the light selection probability distributions. Our main contribution is a formulation of the learning process as Bayesian regression, based on a new, specifically designed statistical model of direct illumination. The net result is a set of regularization strategies to prevent over-fitting and ensure robustness even in early stages of calculation, when the observed information is sparse. The regression model captures spatial variations of illumination, which enables aggregating statistics over relatively large scene regions and, in turn, ensures a fast learning rate. We make the method scalable by adopting a light clustering strategy from the Lightcuts method, and further reduce variance through the use of control variates. As a main design feature, the resulting algorithm is virtually free of any preprocessing, which enables its use for interactive progressive rendering, while the online learning still enables super-linear convergence.

CCS Concepts • **Computing methodologies** → **Rendering**; **Visibility**; **Machine learning**.

**Author's address:** Petr Vévoda, Charles University, Prague, Malostranské náměstí 25, Prague, 11800, Render Legion, a.s., Karlovů náměstí 288/17, Prague, 12000, petrovevoda@seznam.cz, Ivo Kondapaneni, Charles University, Prague, Malostranské náměstí 25, Prague, 11800, ivokondapaneni@gmail.com, Jaroslav Krivánek, Charles University, Prague, Malostranské náměstí 25, Prague, 11800, Render Legion, a.s., Karlovů náměstí 288/17, Prague, 12000, jaroslav.krivaneck@rll.com.cz.

<sup>1</sup>Petr Vévoda and Ivo Kondapaneni share the first authorship of this work. Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions.acm.org](http://permissions.acm.org).  
© 2018 Association for Computing Machinery.  
0730-0301/2018/ART25-\$15.00  
<https://doi.org/10.1145/3197517.3201340>

**Additional Key Words and Phrases:** direct illumination, adaptive sampling, visibility, learning

**ACM Reference Format:**  
Petr Vévoda, Ivo Kondapaneni, and Jaroslav Krivánek. 2018. Bayesian online regression for adaptive direct illumination sampling. *ACM Trans. Graph.* 37, 4, Article 125 (August 2018), 12 pages. <https://doi.org/10.1145/3197517.3201340>

### 1 INTRODUCTION

Realistic rendering today is almost entirely based on Monte Carlo (MC) methods. The indirect illumination component has traditionally been held responsible for the undesirable image noise produced by such algorithms, which is probably why the direct illumination has received disproportionately less attention in research. However, many scenes in digital production feature complex lighting setups, and practical experience shows that it is often direct illumination that is responsible for the majority of image noise.

In this paper, we aim at unbiased direct illumination estimation for MC renderers. Specifically, we address the problem of randomly choosing an appropriate light source for a given scene location, so that variance of the direct illumination estimator is minimized. This could be achieved by choosing lights with probability proportional to their respective contributions, but these are unknown at the outset, they are costly to evaluate and difficult to predict. This is true especially due to the visibility, since it can be discontinuous and its evaluation involves expensive ray casting.

One possible solution would involve constructing the light sampling distributions in a preprocessing step (Georgiev et al. 2012). However, long preprocessing disqualifies any form of interactive rendering – a crucial feature of any modern progressive renderer, a feature that we consider a hard constraint in our work. Such preprocessing can be avoided by learning from the observed samples

ACM Transactions on Graphics, Vol. 37, No. 4, Article 125. Publication date: August 2018.

### Importance Sampling of Many Lights with Adaptive Tree Splitting

ALEJANDRO CONTY ESTEVEZ, Sony Pictures Imageworks  
CHRISTOPHER KULLA, Sony Pictures Imageworks



Fig. 1. A procedural city with 363,036 lights, one GI bounce and participating media. Rendered with 16 samples per pixel, each shading point takes an average of 7 shadow rays (45 for the volume integral). We shoot an average of 1700 rays per pixel. The image rendered in 20 minutes on a quad core Intel i7.

We present a technique to importance sample large collections of lights (including mesh lights as collections of small emitters) in the context of Monte-Carlo path tracing. A bounding volume hierarchy over all emitters is traversed at each shading point using a single random number in a way that importance samples their predicted contribution. The tree aggregates energy, spatial and orientation information from the emitters to enable accurate prediction of the effect of a cluster of lights on any given shading point. We further improve the performance of the algorithm by forcing splitting until the importance of a cluster is sufficiently representative of its contents.

CCS Concepts • **Computing methodologies** → **Ray tracing**;

**Additional Key Words and Phrases:** illumination, ray tracing, many lights

**ACM Reference Format:**  
Alejandro Conty Estevez and Christopher Kulla. 2018. Importance Sampling of Many Lights with Adaptive Tree Splitting. *Proc. ACM Comput. Graph. Interact. Tech.* 1, 2, Article 25 (August 2018), 17 pages. <https://doi.org/10.1145/3233305>

### 1 INTRODUCTION

Direct lighting calculations are a critical part of modern path tracing renderers with next event estimation. While sampling from simple light shapes [Shirley et al. 1996] is well understood, relatively little attention has been devoted to the problem of efficiently sampling from large collections of such shapes. In production renderers, this problem appears both in the form of scenes containing many distinct lights (Figure 1), and scenes with meshes acting as emitters (sometimes

**Authors' addresses:** Alejandro Conty Estevez, Sony Pictures Imageworks; Christopher Kulla, Sony Pictures Imageworks.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions.acm.org](http://permissions.acm.org).

© 2018 Copyright held by the owner/author(s). Publication rights licensed to ACM.  
2577-6193/2018/8-ART25-\$15.00  
<https://doi.org/10.1145/3233305>

Proc. ACM Comput. Graph. Interact. Tech., Vol. 1, No. 2, Article 25. Publication date: August 2018.



# Real-Time is harder





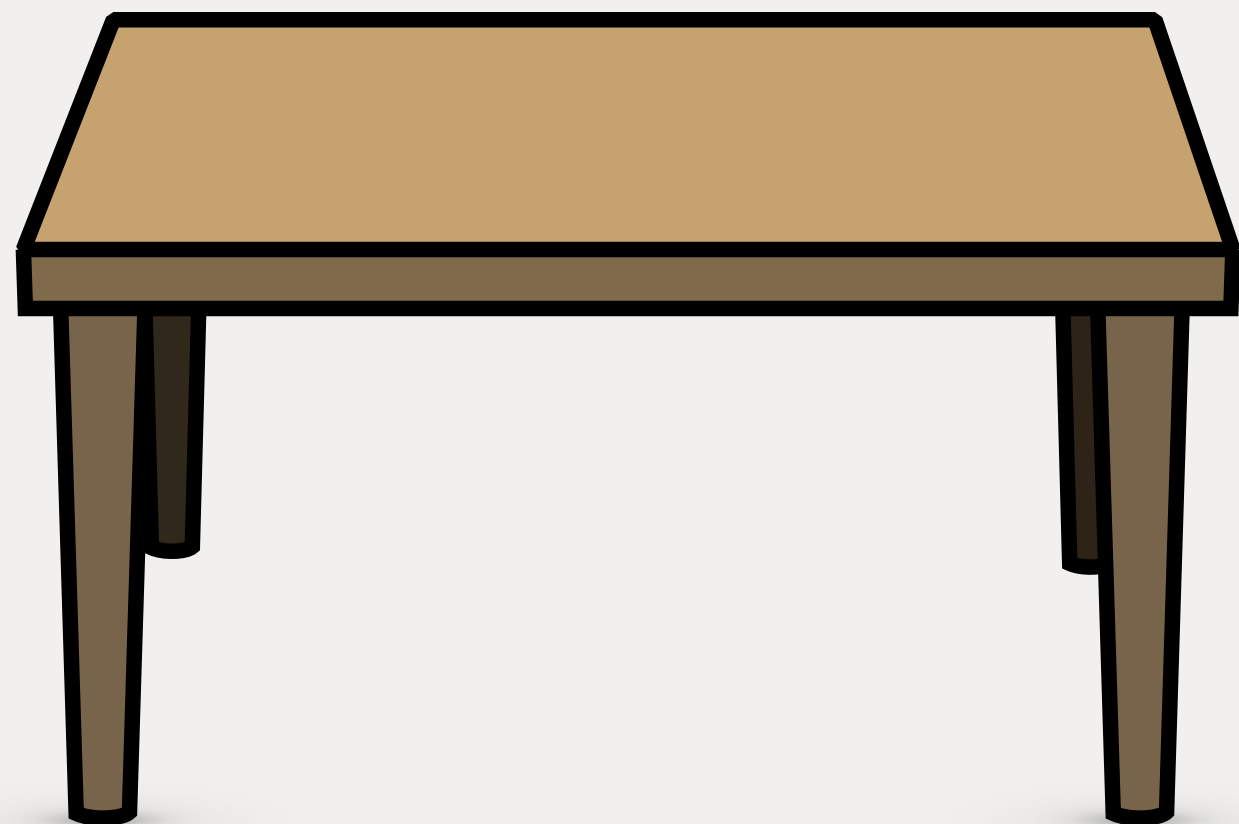
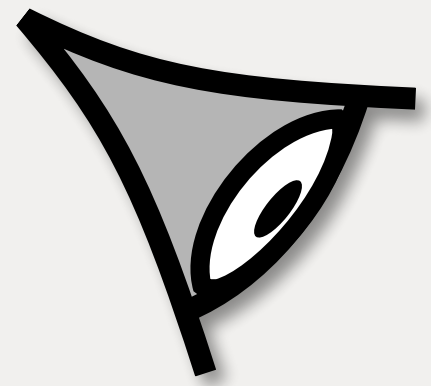
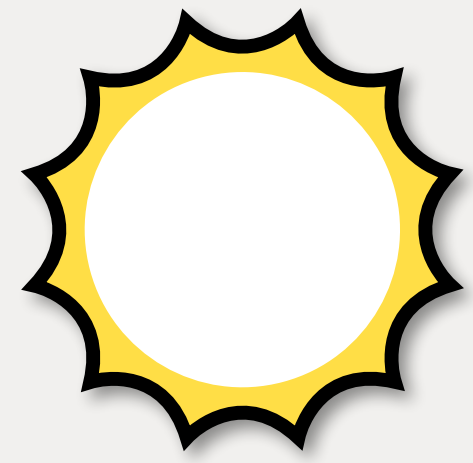
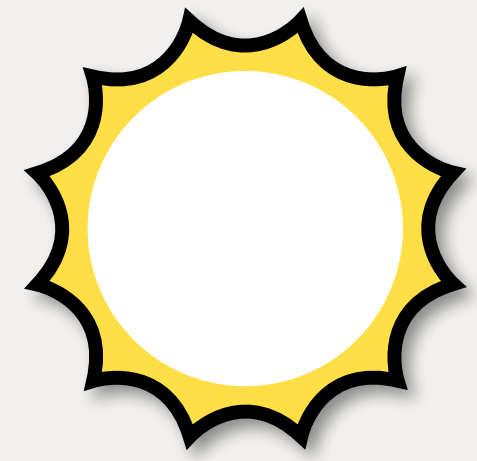
# Goals

- Good quality at low sample count ( $\leq 4$  spp)
- *Everything* is dynamic
- Massively parallel

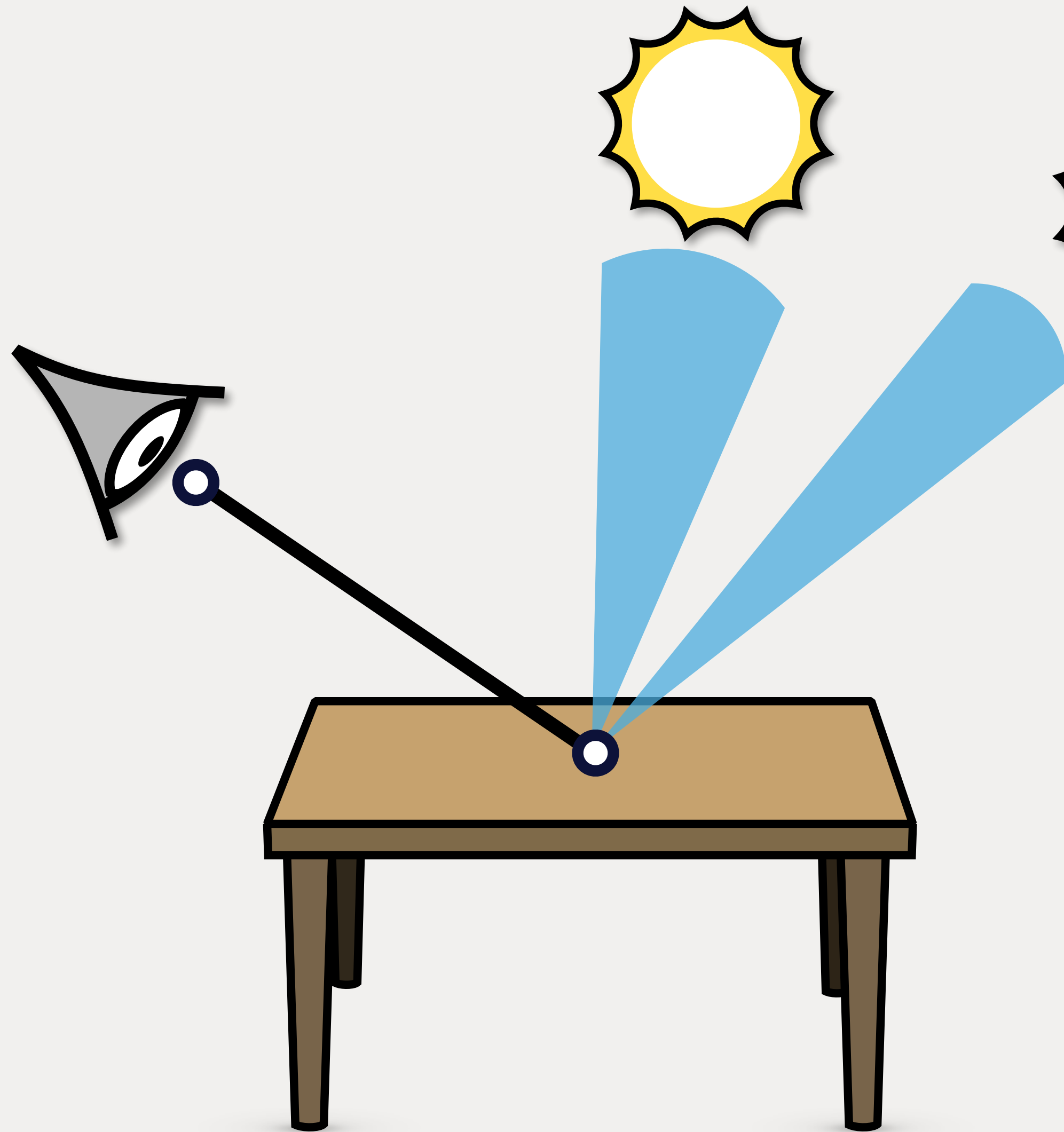
# Problem Description



# Problem Description



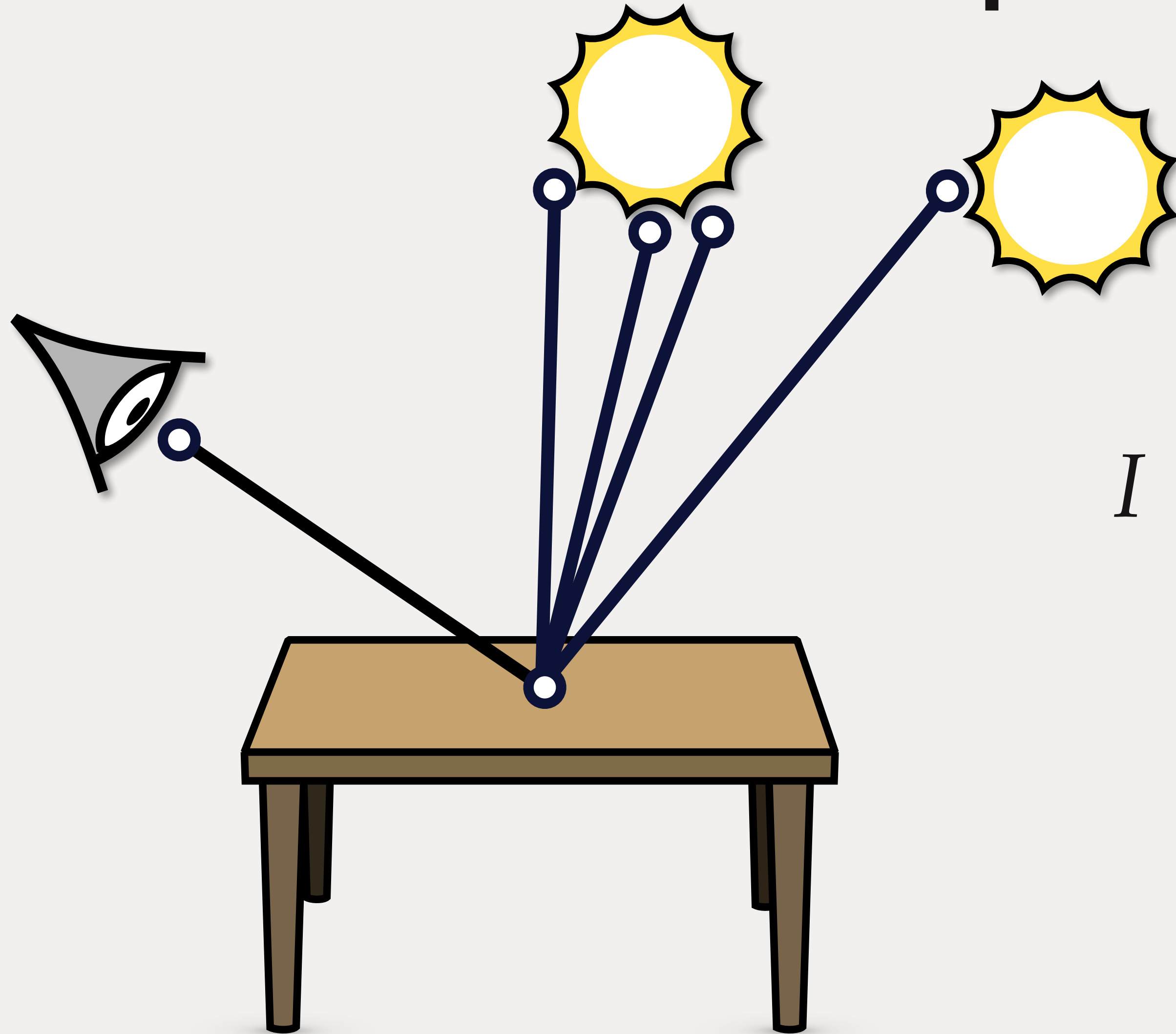
# Problem Description



$$I = \int_{\Omega} f(x) dx$$

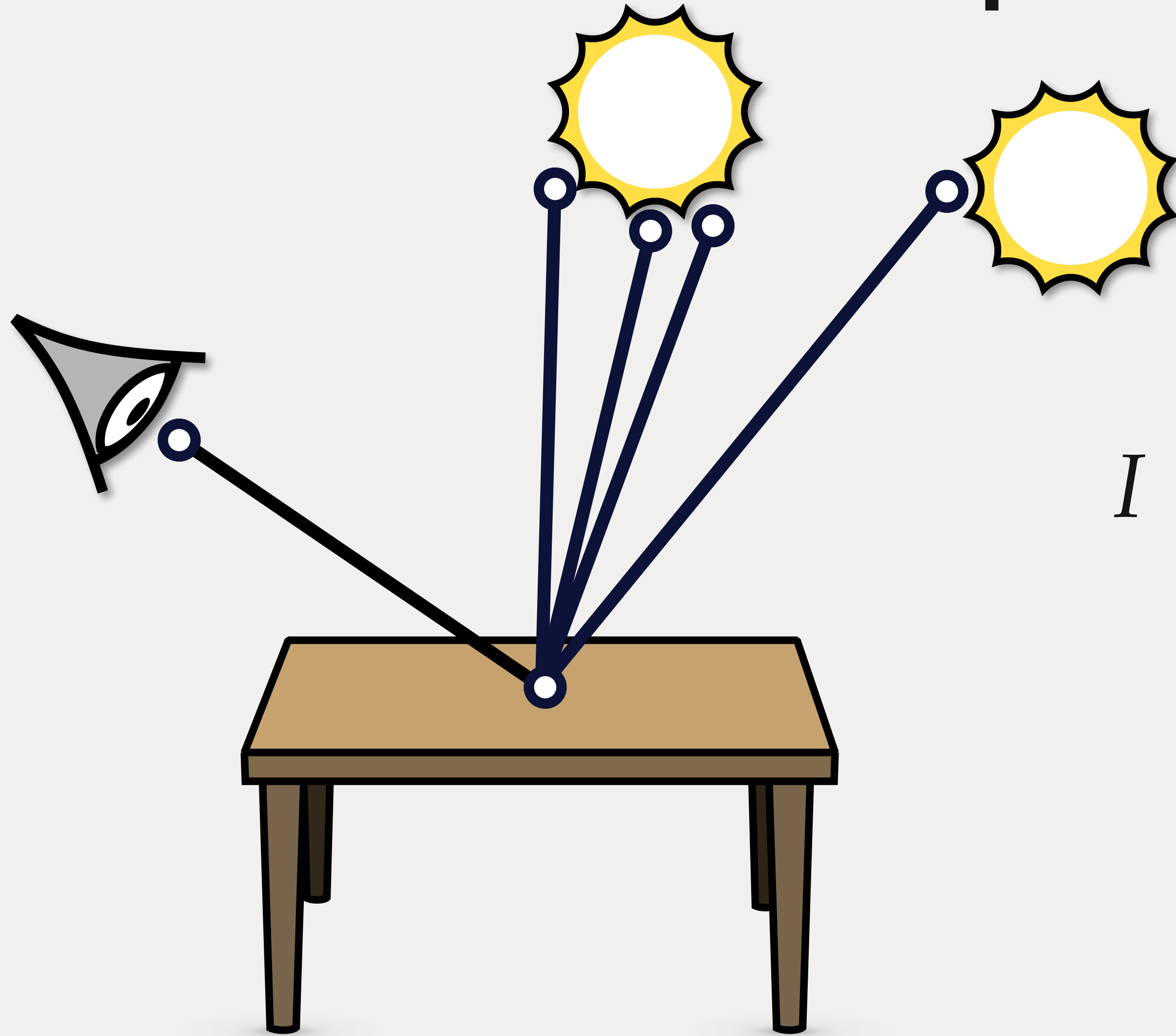


# Problem Description



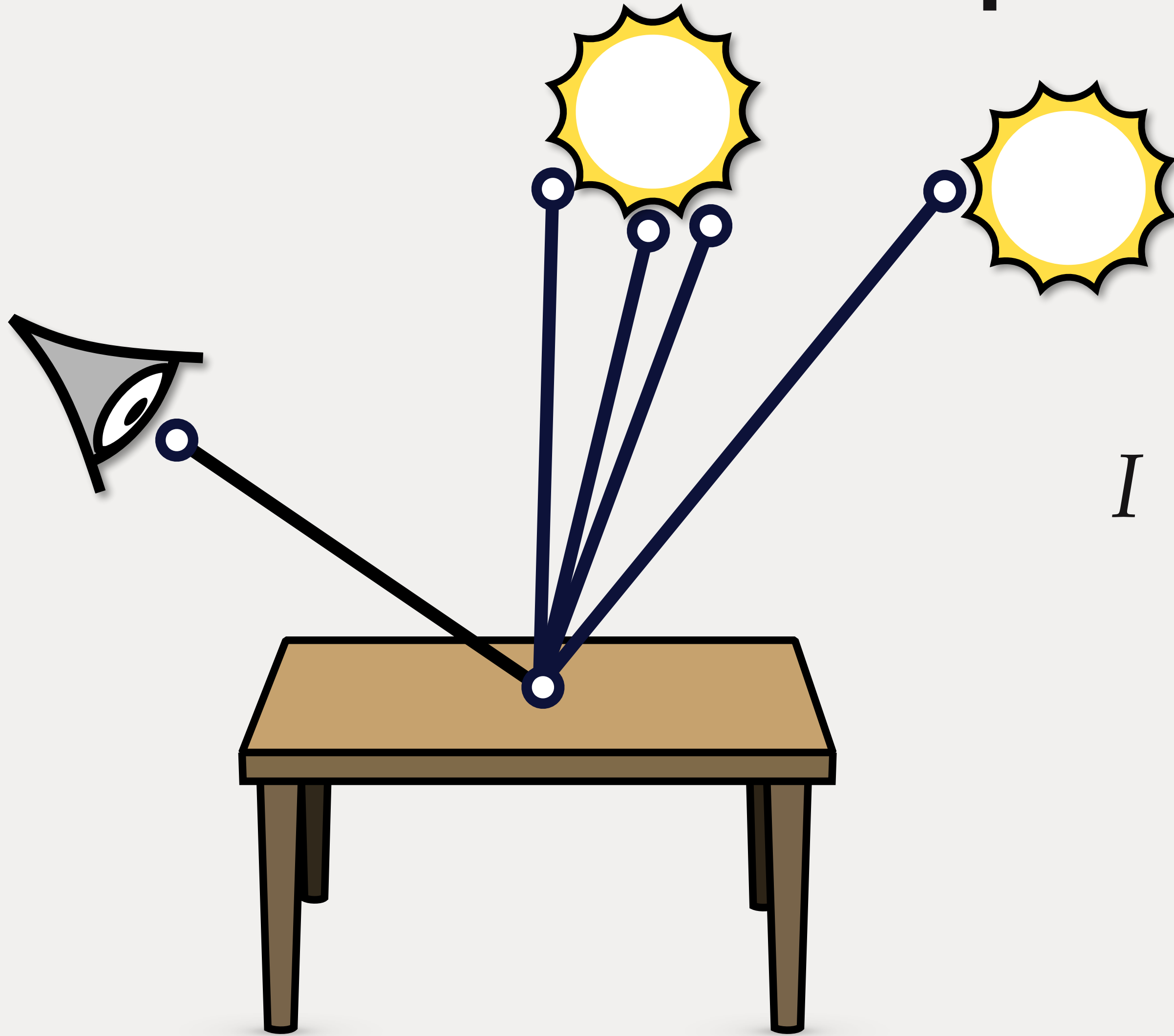
$$I = \int_{\Omega} f(x) dx \approx \frac{1}{N} \sum_i^N \frac{f(x_i)}{p(x_i)}$$

# Problem Description



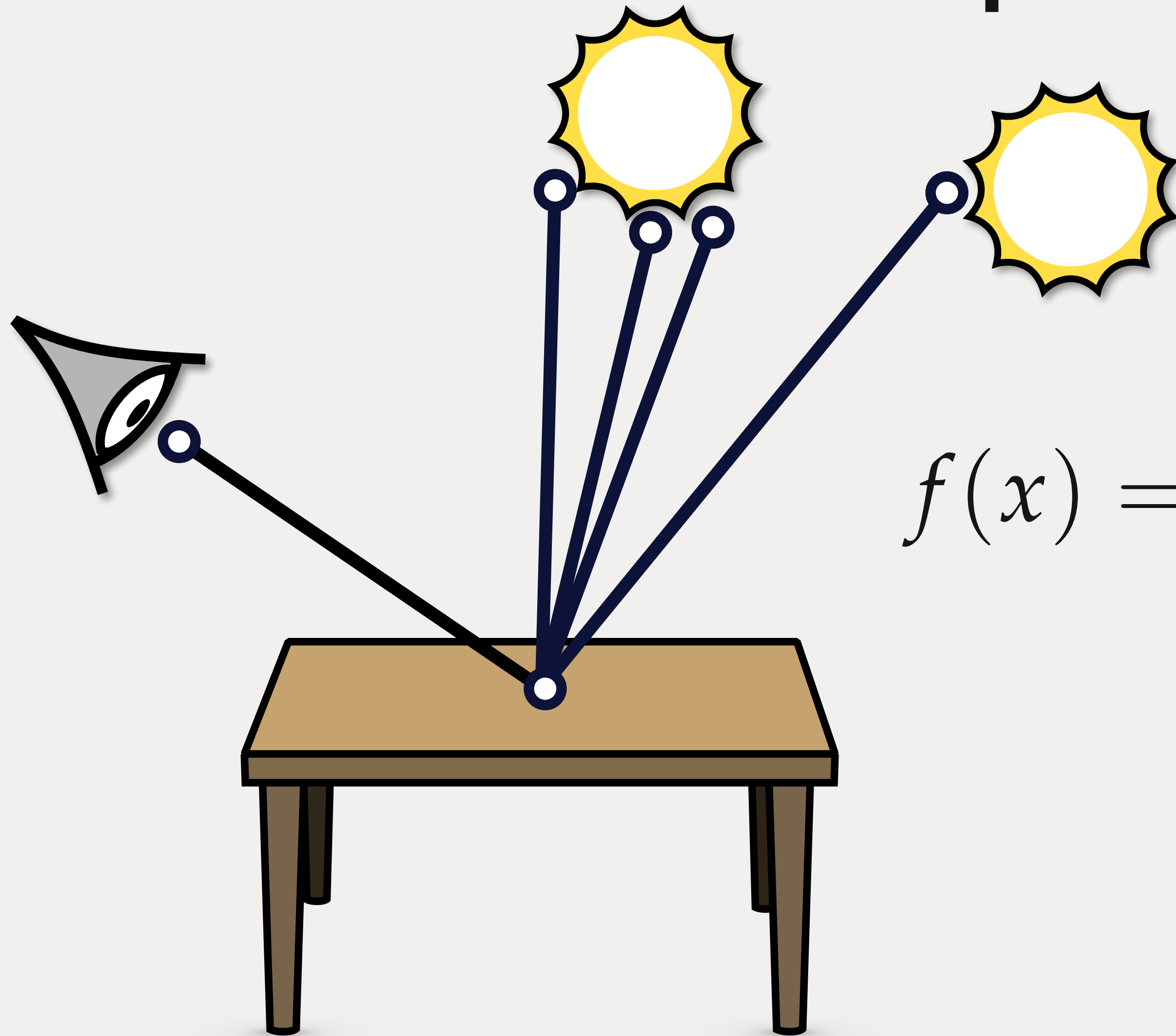
$$I = \int_{\Omega} f(x) dx \approx \frac{1}{N} \sum_i^N \frac{f(x_i)}{p(x_i)}$$

# Problem Description



$$I = \int_{\Omega} f(x) dx \approx \frac{1}{N} \sum_i^N \frac{f(x_i)}{p(x_i)}$$

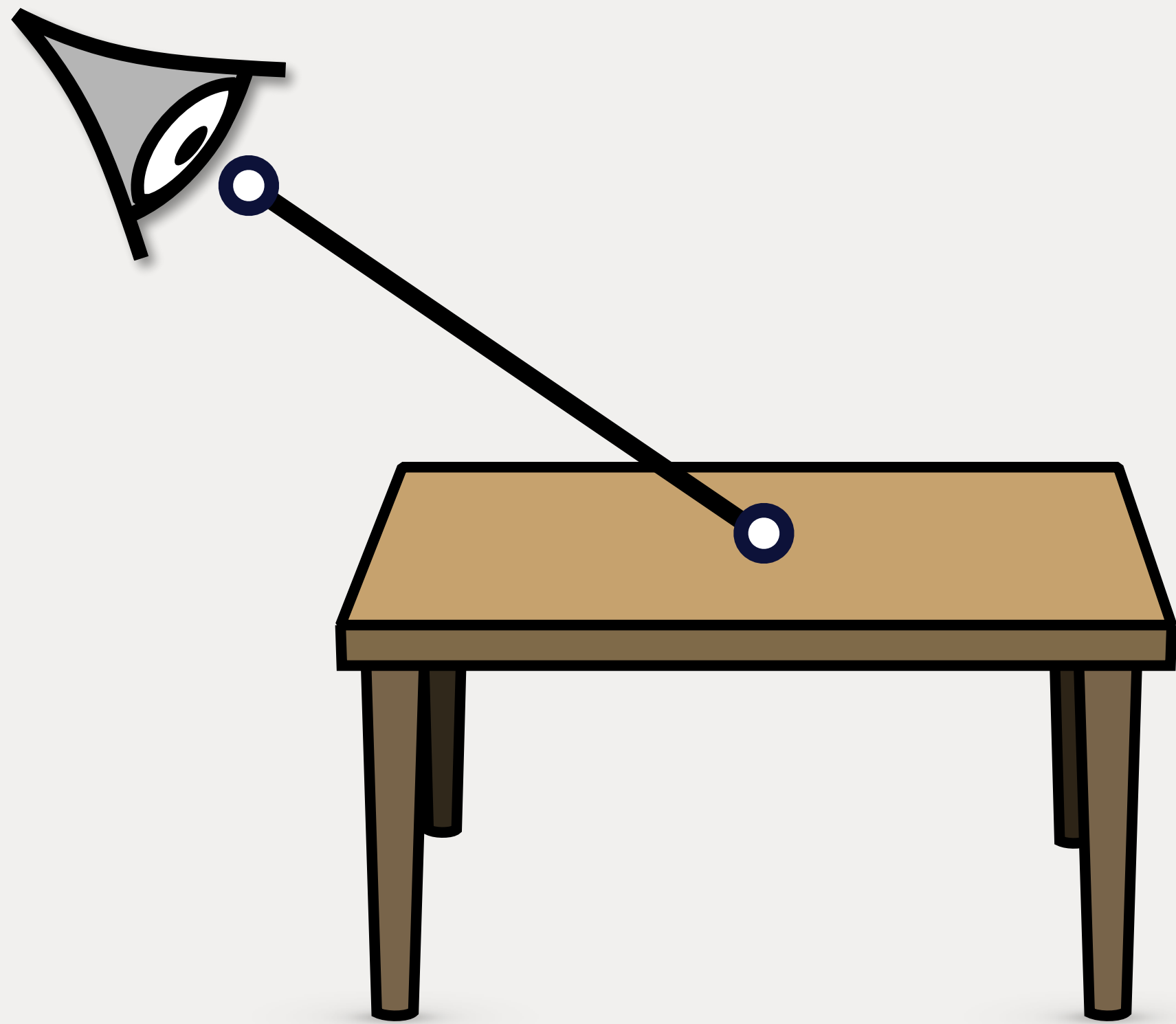
# Problem Description



$$f(x) = L_e(x) \rho(x) V(x) \frac{\cos \theta_i \cos \theta_o}{r^2}$$

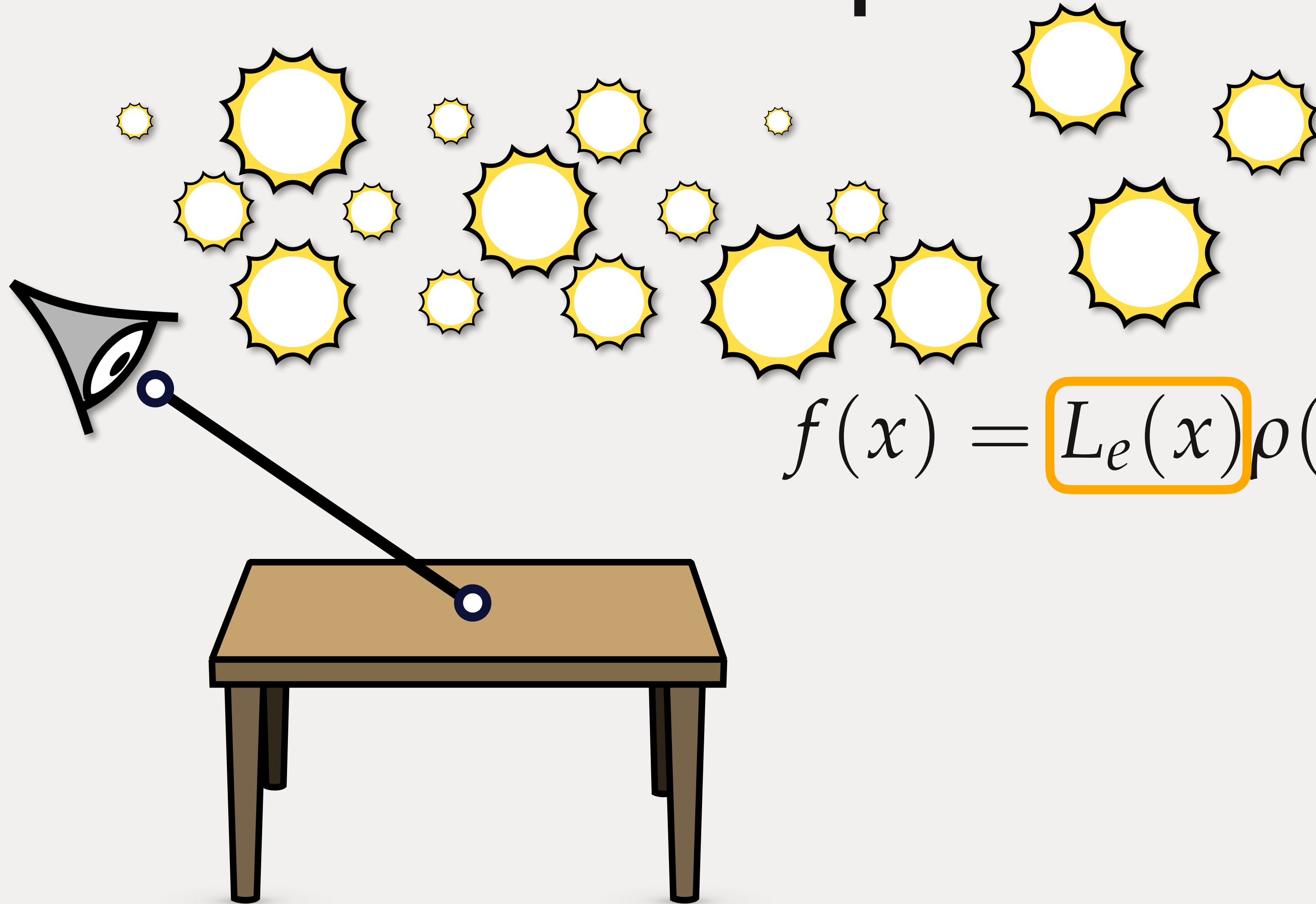


# Problem Description



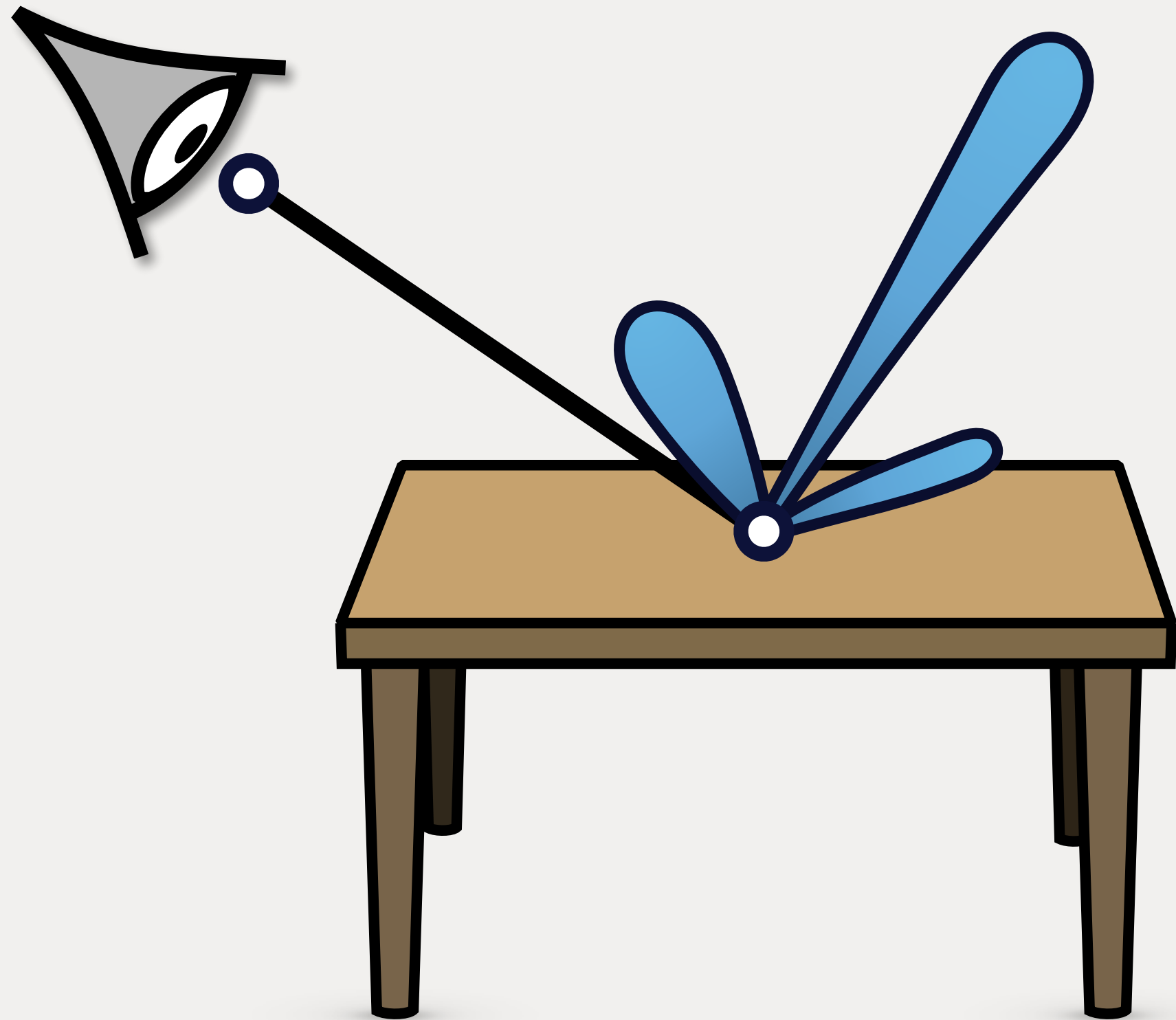
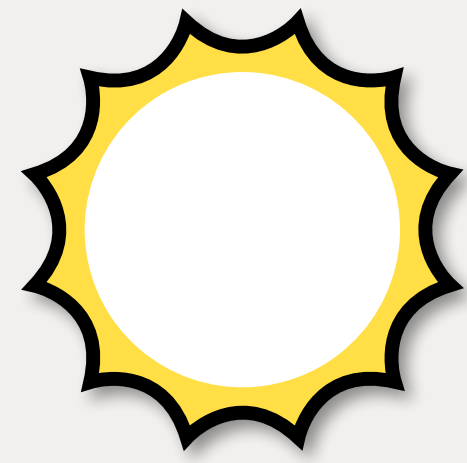
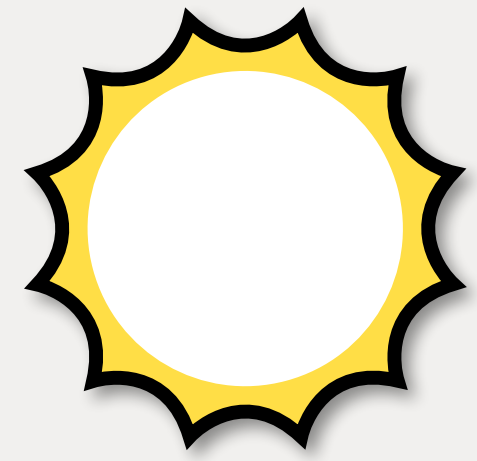
$$f(x) = L_e(x) \rho(x) V(x) \frac{\cos \theta_i \cos \theta_o}{r^2}$$

# Problem Description



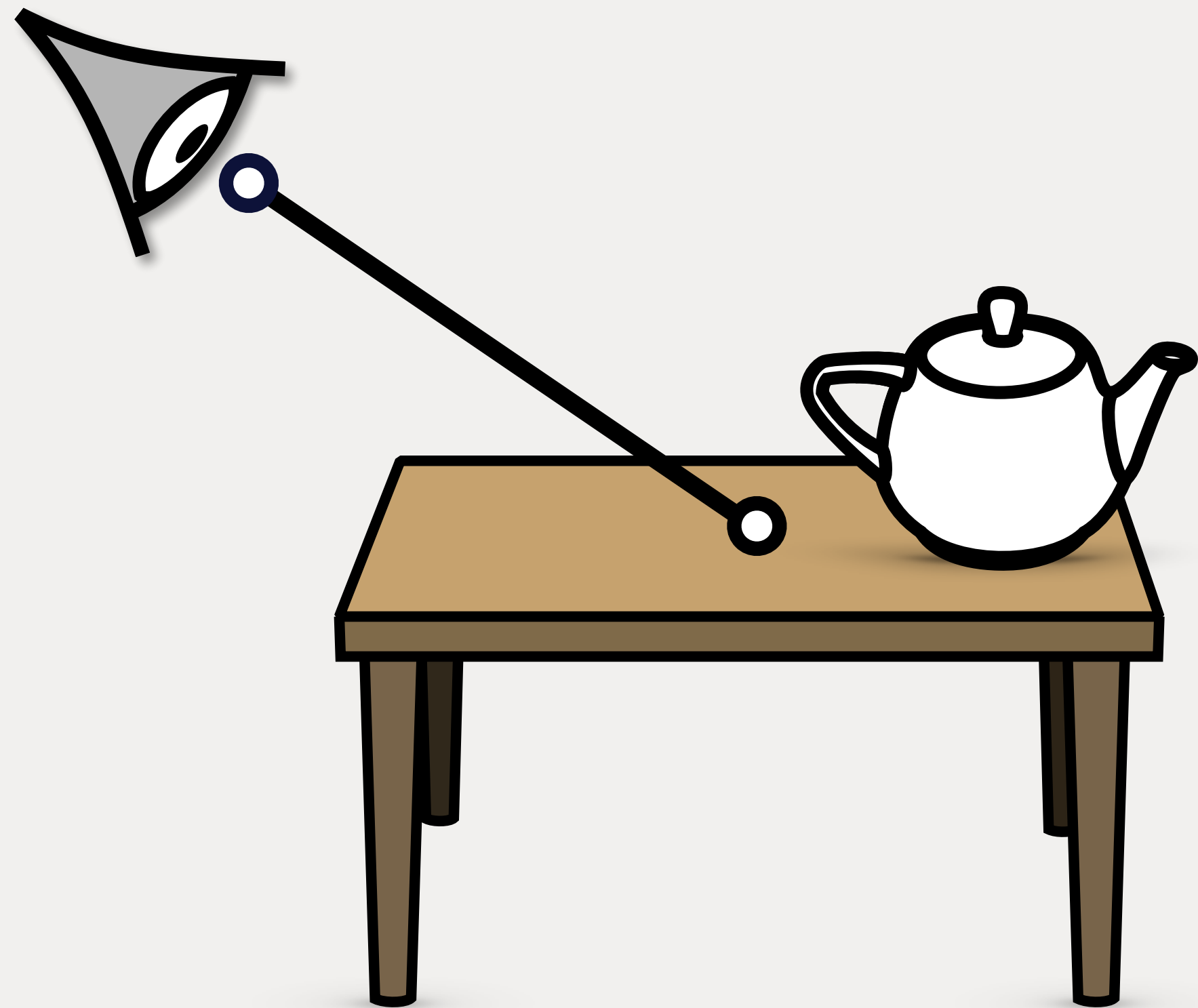
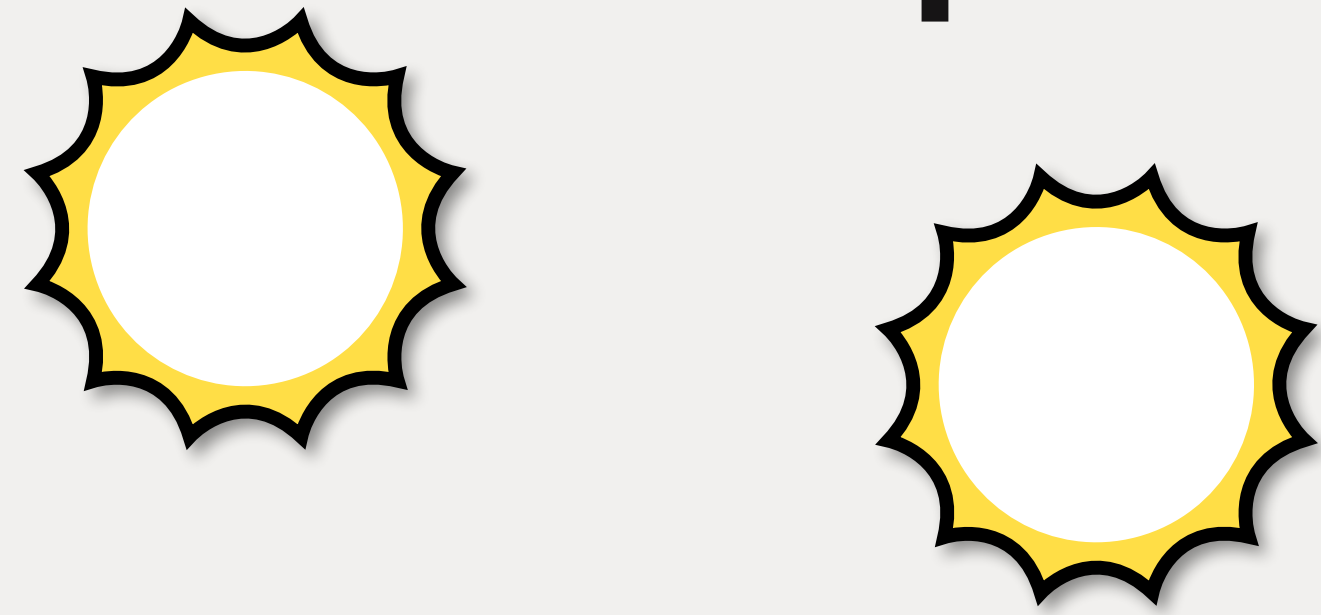
$$f(x) = L_e(x) \rho(x) V(x) \frac{\cos \theta_i \cos \theta_o}{r^2}$$

# Problem Description



$$f(x) = L_e(x) \rho(x) V(x) \frac{\cos \theta_i \cos \theta_o}{r^2}$$

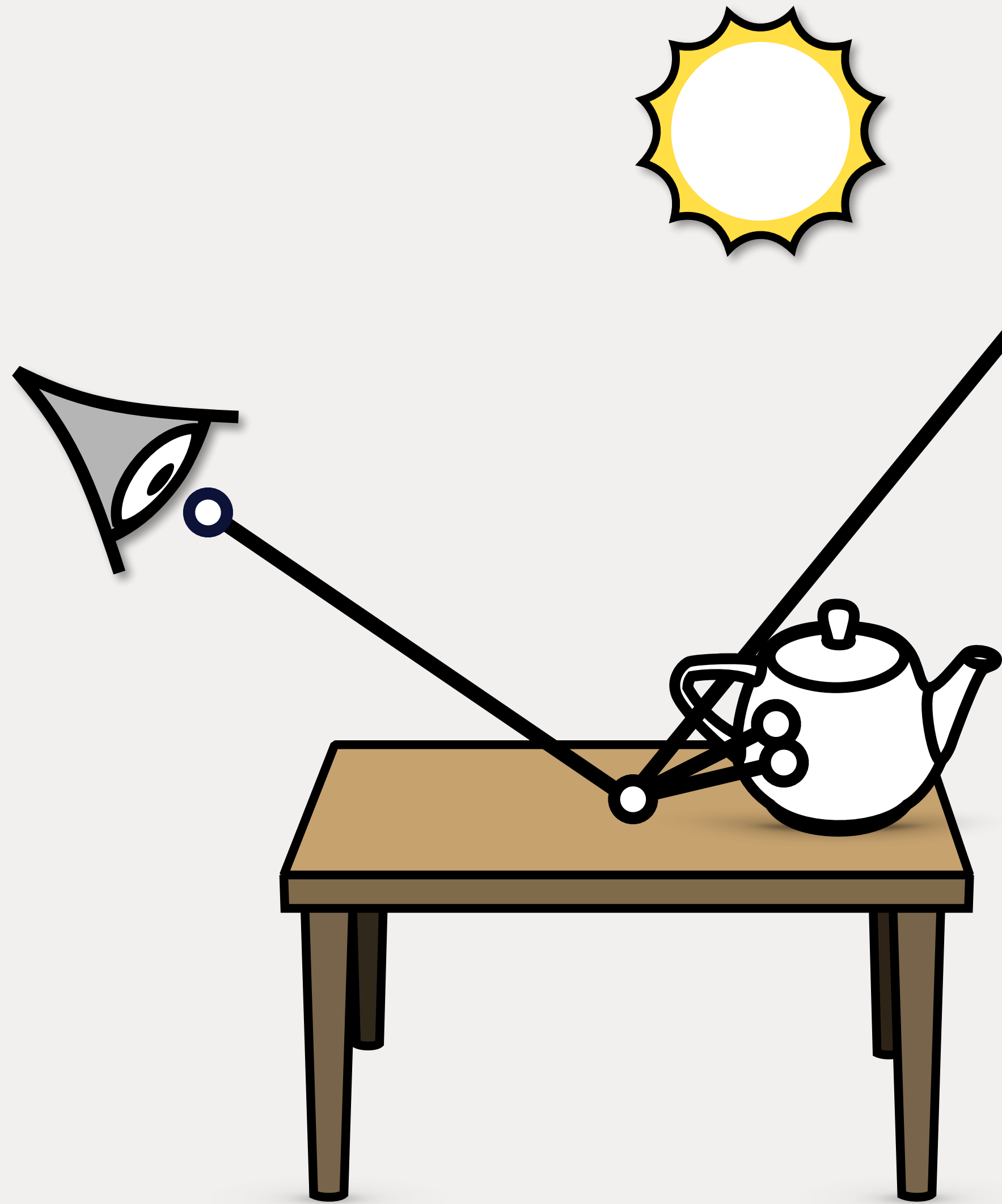
# Problem Description



$$f(x) = L_e(x) \rho(x) \boxed{V(x)} \frac{\cos \theta_i \cos \theta_o}{r^2}$$

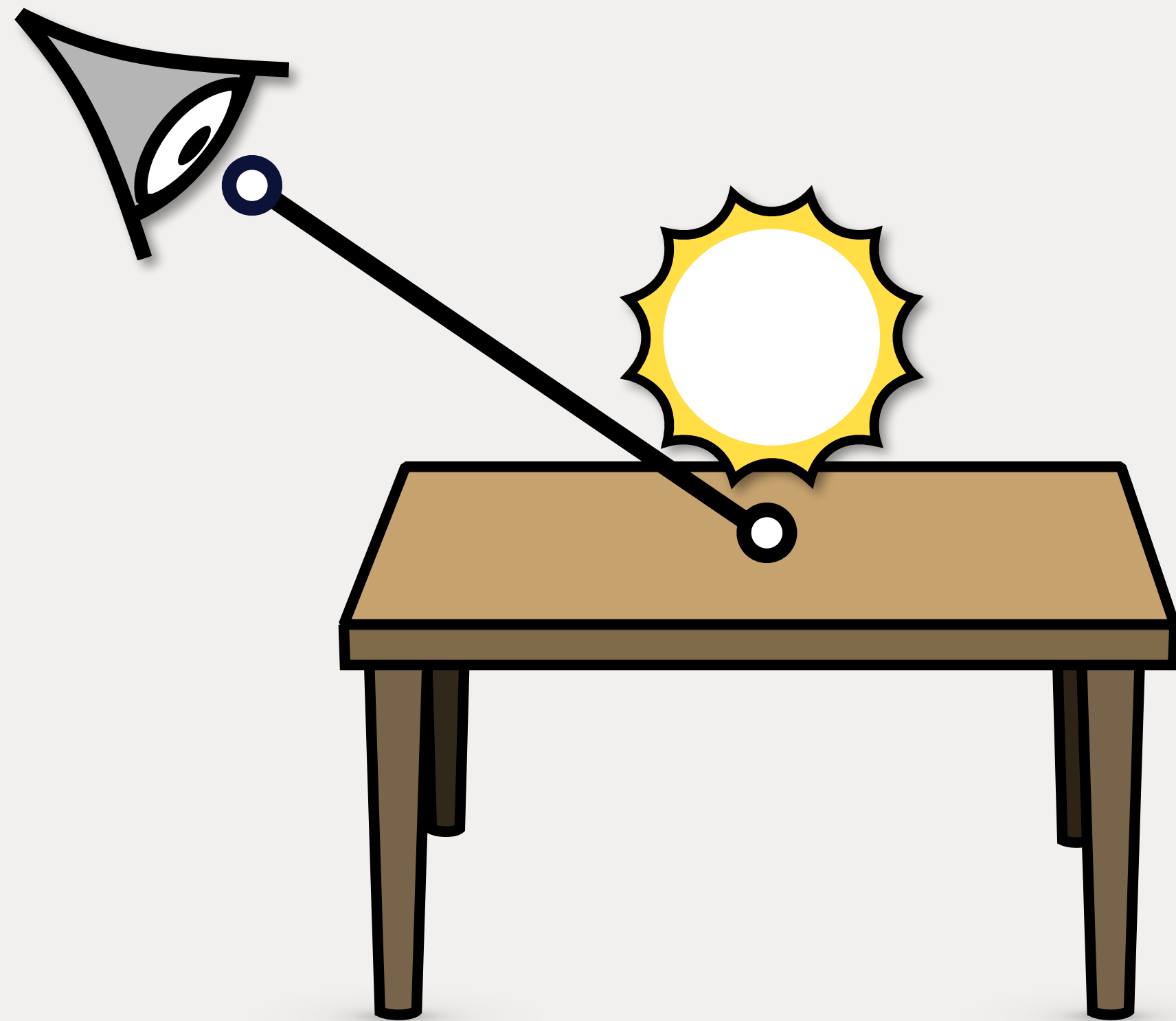
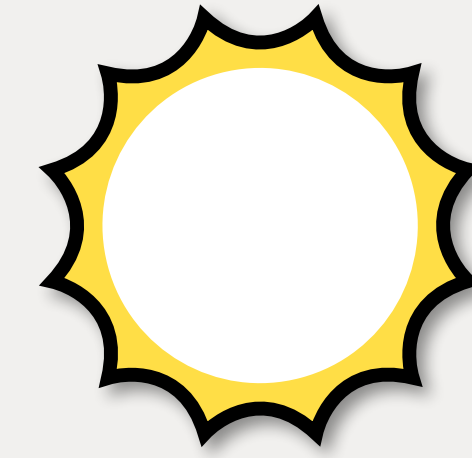


# Problem Description



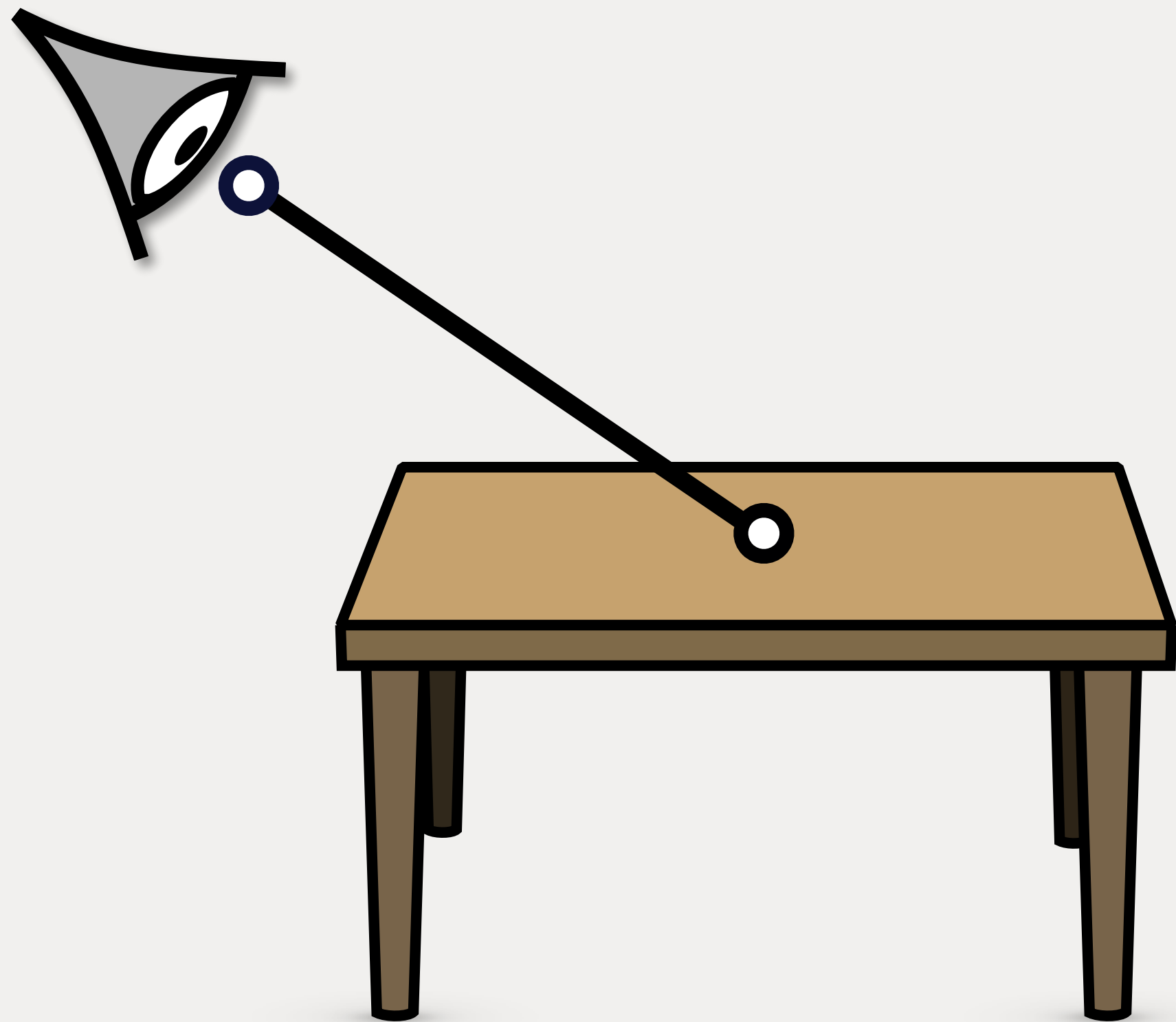
$$f(x) = L_e(x) \rho(x) \boxed{V(x)} \frac{\cos \theta_i \cos \theta_o}{r^2}$$

# Problem Description



$$f(x) = L_e(x) \rho(x) V(x) \frac{\cos \theta_i \cos \theta_o}{r^2}$$

# Problem Description



$$f(x) = L_e(x) \rho(x) V(x) \frac{\cos \theta_i \cos \theta_o}{r^2}$$

- Want to sample *product*
- This is hard!



# Related Work

BRDF

---

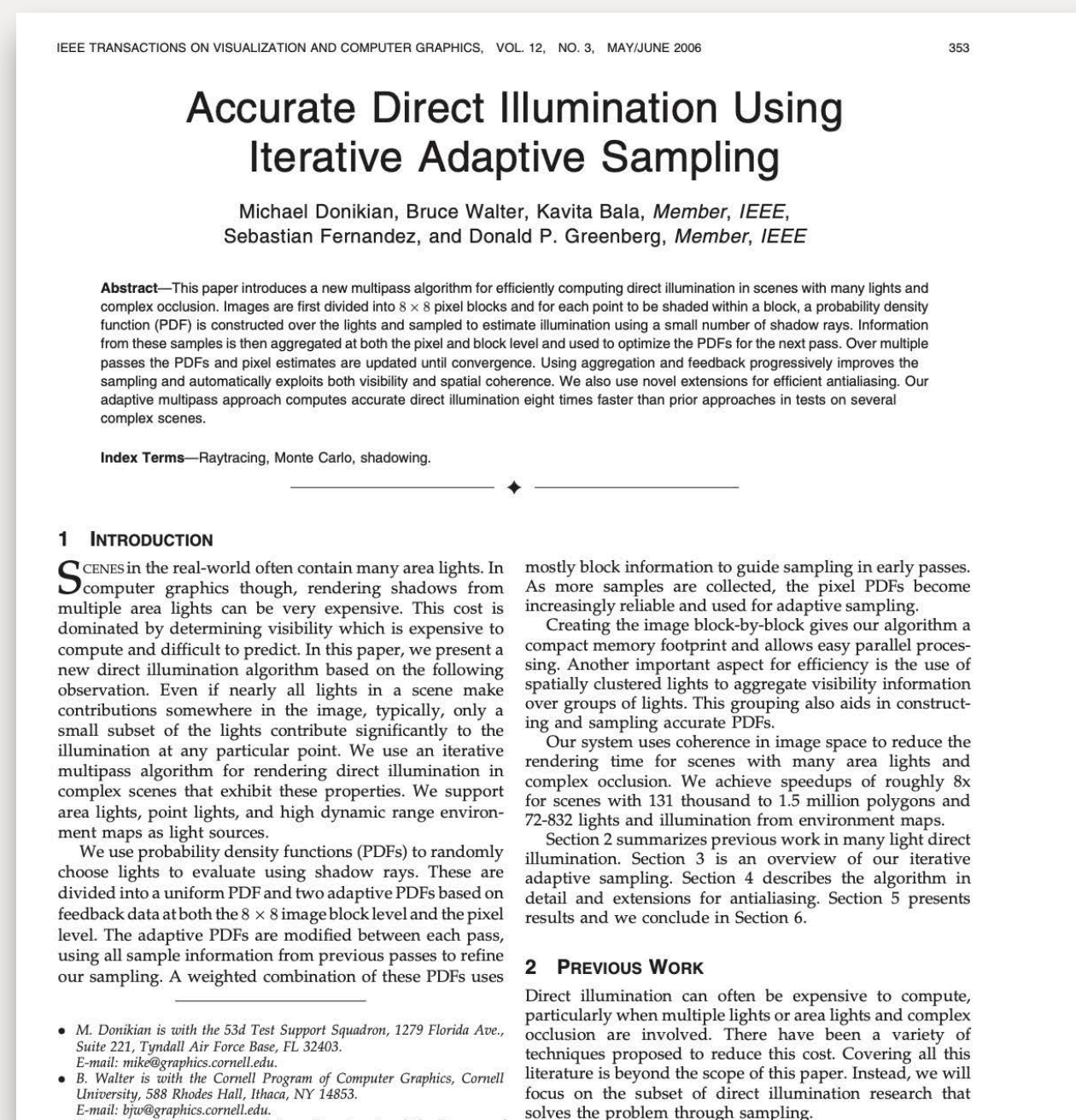
Visibility

---

Geometry

# Related Work

## Donikian et al. 2006



BRDF



Visibility



Geometry





# Related Work

## Donikian et al. 2006

IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS, VOL. 12, NO. 3, MAY/JUNE 2006 353

### Accurate Direct Illumination Using Iterative Adaptive Sampling

Michael Donikian, Bruce Walter, Kavita Bala, *Member, IEEE*, Sebastian Fernandez, and Donald P. Greenberg, *Member, IEEE*

**Abstract**—This paper introduces a new multipass algorithm for efficiently computing direct illumination in scenes with many lights and complex occlusion. Images are first divided into  $8 \times 8$  pixel blocks and for each point to be shaded within a block, a probability density function (PDF) is constructed over the lights and sampled to estimate illumination using a small number of shadow rays. Information from these samples is then aggregated at both the pixel and block level and used to optimize the PDFs for the next pass. Over multiple passes the PDFs and pixel estimates are updated until convergence. Using aggregation and feedback progressively improves the sampling and automatically exploits both visibility and spatial coherence. We also use novel extensions for efficient antialiasing. Our adaptive multipass approach computes accurate direct illumination eight times faster than prior approaches in tests on several complex scenes.

**Index Terms**—Raytracing, Monte Carlo, shadowing.

#### 1 INTRODUCTION

SCENES in the real-world often contain many area lights. In computer graphics though, rendering shadows from multiple area lights can be very expensive. This cost is dominated by determining visibility which is expensive to compute and difficult to predict. In this paper, we present a new direct illumination algorithm based on the following observation. Even if nearly all lights in a scene make contributions somewhere in the image, typically, only a small subset of the lights contribute significantly to the illumination at any particular point. We use an iterative multipass algorithm for rendering direct illumination in complex scenes that exhibit these properties. We support area lights, point lights, and high dynamic range environment maps as light sources.

We use probability density functions (PDFs) to randomly choose lights to evaluate using shadow rays. These are divided into a uniform PDF and two adaptive PDFs based on feedback data at both the  $8 \times 8$  image block level and the pixel level. The adaptive PDFs are modified between each pass, using all sample information from previous passes to refine our sampling. A weighted combination of these PDFs uses

mostly block information to guide sampling in early passes. As more samples are collected, the pixel PDFs become increasingly reliable and used for adaptive sampling. Creating the image block-by-block gives our algorithm a compact memory footprint and allows easy parallel processing. Another important aspect for efficiency is the use of spatially clustered lights to aggregate visibility information over groups of lights. This grouping also aids in constructing and sampling accurate PDFs.

Our system uses coherence in image space to reduce the rendering time for scenes with many area lights and complex occlusion. We achieve speedups of roughly 8x for scenes with 131 thousand to 1.5 million polygons and 72-832 lights and illumination from environment maps.

Section 2 summarizes previous work in many light direct illumination. Section 3 is an overview of our iterative adaptive sampling. Section 4 describes the algorithm in detail and extensions for antialiasing. Section 5 presents results and we conclude in Section 6.

#### 2 PREVIOUS WORK

Direct illumination can often be expensive to compute, particularly when multiple lights or area lights and complex occlusion are involved. There have been a variety of techniques proposed to reduce this cost. Covering all this literature is beyond the scope of this paper. Instead, we will focus on the subset of direct illumination research that solves the problem through sampling.

• M. Donikian is with the 53d Test Support Squadron, 1279 Florida Ave., Suite 221, Tyndall Air Force Base, FL 32403. E-mail: mike@graphics.cornell.edu.  
• B. Walter is with the Cornell Program of Computer Graphics, Cornell University, 588 Rhodes Hall, Ithaca, NY 14853. E-mail: bjw@graphics.cornell.edu.  
• K. Bala is with the Computer Science Department and the Program of

## Burley et al. 2018

### The Design and Evolution of Disney's Hyperion Renderer

BRENT BURLEY, DAVID ADLER, MATT JEN-YUAN CHIANG, HANK DRISKILL, RALF HABEL, PATRICK KELLY, PETER KUTZ, YINING KARL LI, and DANIEL TEECE, Walt Disney Animation Studios




Fig. 1. Production frames from *Big Hero 6* (upper left), *Zootopia* (upper right), *Moana* (bottom left), and *Olaf's Frozen Adventure* (bottom right), all rendered using Disney's Hyperion Renderer.

Walt Disney Animation Studios has transitioned to path-traced global illumination as part of a progression of brute-force physically based rendering in the name of artist efficiency. To achieve this without compromising our geometric or shading complexity, we built our Hyperion renderer based on a novel architecture that extracts traversal and shading coherence from large, sorted ray batches. In this article, we describe our architecture and discuss our design decisions. We also explain how we are able to provide artistic control in a physically based renderer, and we demonstrate through case studies how we have benefited from having a proprietary renderer that can evolve with production needs.

CCS Concepts • **Computing methodologies** → **Rendering**; *Ray tracing*;  
Additional Key Words and Phrases: Production rendering, physically based rendering, path tracing

© 2018 Copyright held by the owner/author(s). Publication rights licensed to ACM. This is the author's version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published in *ACM Transactions on Graphics*, <https://doi.org/10.1145/3182159>.

**ACM Reference format:**  
Brent Burley, David Adler, Matt Jen-Yuan Chiang, Hank Driskill, Ralf Habel, Patrick Kelly, Peter Kutz, Yining Karl Li, and Daniel Teece. 2018. The Design and Evolution of Disney's Hyperion Renderer. *ACM Trans. Graph.* 37, 3, Article 33 (July 2018), 22 pages. <https://doi.org/10.1145/3182159>

#### 1 INTRODUCTION

Since the early 1990s, rendering of computer graphics (CG) imagery at Walt Disney Animation Studios had been accomplished using the Reyes algorithm (Cook et al. 1987) in Pixar's RenderMan. More recently, ray-traced global illumination (GI) promised artists significant productivity gains by providing more immediate feedback during rendering and removing the significant data management burden associated with shadow maps and point clouds. However, initial attempts to render our existing production scenes with ray-traced GI were unsuccessful; incoherent access of texture maps inhibited shading of indirect ray hits, and we had difficulty fitting our scenes in memory as required by existing ray-traced renderers.

To overcome these limitations, we created a new rendering architecture which traces and shades rays in large batches, first sorting each batch for geometric coherence during scene traversal.

BRDF  
Visibility  
Geometry





# Related Work

## Donikian et al. 2006

IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS, VOL. 12, NO. 3, MAY/JUNE 2006 353

### Accurate Direct Illumination Using Iterative Adaptive Sampling

Michael Donikian, Bruce Walter, Kavita Bala, *Member, IEEE*, Sebastian Fernandez, and Donald P. Greenberg, *Member, IEEE*

**Abstract**—This paper introduces a new multipass algorithm for efficiently computing direct illumination in scenes with many lights and complex occlusion. Images are first divided into  $8 \times 8$  pixel blocks and for each point to be shaded within a block, a probability density function (PDF) is constructed over the lights and sampled to estimate illumination using a small number of shadow rays. Information from these samples is then aggregated at both the pixel and block level and used to optimize the PDFs for the next pass. Over multiple passes the PDFs and pixel estimates are updated until convergence. Using aggregation and feedback progressively improves the sampling and automatically exploits both visibility and spatial coherence. We also use novel extensions for efficient antialiasing. Our adaptive multipass approach computes accurate direct illumination eight times faster than prior approaches in tests on several complex scenes.

**Index Terms**—Raytracing, Monte Carlo, shadowing.

#### 1 INTRODUCTION

SCENES in the real-world often contain many area lights. In computer graphics though, rendering shadows from multiple area lights can be very expensive. This cost is dominated by determining visibility which is expensive to compute and difficult to predict. In this paper, we present a new direct illumination algorithm based on the following observation. Even if nearly all lights in a scene make contributions somewhere in the image, typically, only a small subset of the lights contribute significantly to the illumination at any particular point. We use an iterative multipass algorithm for rendering direct illumination in complex scenes that exhibit these properties. We support area lights, point lights, and high dynamic range environment maps as light sources.

We use probability density functions (PDFs) to randomly choose lights to evaluate using shadow rays. These are divided into a uniform PDF and two adaptive PDFs based on feedback data at both the  $8 \times 8$  image block level and the pixel level. The adaptive PDFs are modified between each pass, using all sample information from previous passes to refine our sampling. A weighted combination of these PDFs uses

mostly block information to guide sampling in early passes. As more samples are collected, the pixel PDFs become increasingly reliable and used for adaptive sampling. Creating the image block-by-block gives our algorithm a compact memory footprint and allows easy parallel processing. Another important aspect for efficiency is the use of spatially clustered lights to aggregate visibility information over groups of lights. This grouping also aids in constructing and sampling accurate PDFs.

Our system uses coherence in image space to reduce the rendering time for scenes with many area lights and complex occlusion. We achieve speedups of roughly 8x for scenes with 131 thousand to 1.5 million polygons and 72-832 lights and illumination from environment maps.

Section 2 summarizes previous work in many light direct illumination. Section 3 is an overview of our iterative adaptive sampling. Section 4 describes the algorithm in detail and extensions for antialiasing. Section 5 presents results and we conclude in Section 6.

#### 2 PREVIOUS WORK

Direct illumination can often be expensive to compute, particularly when multiple lights or area lights and complex occlusion are involved. There have been a variety of techniques proposed to reduce this cost. Covering all this literature is beyond the scope of this paper. Instead, we will focus on the subset of direct illumination research that solves the problem through sampling.

• M. Donikian is with the 53d Test Support Squadron, 1279 Florida Ave., Suite 221, Tyn dall Air Force Base, FL 32403. E-mail: mike@graphics.cornell.edu.  
 • B. Walter is with the Cornell Program of Computer Graphics, Cornell University, 588 Rhodes Hall, Ithaca, NY 14853. E-mail: bjw@graphics.cornell.edu.  
 • K. Bala is with the Computer Science Department and the Program of

## Burley et al. 2018

### The Design and Evolution of Disney's Hyperion Renderer

BRENT BURLEY, DAVID ADLER, MATT JEN-YUAN CHIANG, HANK DRISKILL, RALF HABEL, PATRICK KELLY, PETER KUTZ, YINING KARL LI, and DANIEL TEECE, Walt Disney Animation Studios




Fig. 1. Production frames from *Big Hero 6* (upper left), *Zootopia* (upper right), *Moana* (bottom left), and *Olaf's Frozen Adventure* (bottom right), all rendered using Disney's Hyperion Renderer.

Walt Disney Animation Studios has transitioned to path-traced global illumination as part of a progression of brute-force physically based rendering in the name of artist efficiency. To achieve this without compromising our geometric or shading complexity, we built our Hyperion renderer based on a novel architecture that extracts traversal and shading coherence from large, sorted ray batches. In this article, we describe our architecture and discuss our design decisions. We also explain how we are able to provide artistic control in a physically based renderer, and we demonstrate through case studies how we have benefited from having a proprietary renderer that can evolve with production needs.

CCS Concepts • **Computing methodologies** → **Rendering**; *Ray tracing*;  
 Additional Key Words and Phrases: Production rendering, physically based rendering, path tracing

© 2018 Copyright held by the owner/author(s). Publication rights licensed to ACM. This is the author's version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published in *ACM Transactions on Graphics*, <https://doi.org/10.1145/3182159>.

**ACM Reference format:**  
 Brent Burley, David Adler, Matt Jen-Yuan Chiang, Hank Driskill, Ralf Habel, Patrick Kelly, Peter Kutz, Yining Karl Li, and Daniel Teece. 2018. The Design and Evolution of Disney's Hyperion Renderer. *ACM Trans. Graph.* 37, 3, Article 33 (July 2018), 22 pages. <https://doi.org/10.1145/3182159>

#### 1 INTRODUCTION

Since the early 1990s, rendering of computer graphics (CG) imagery at Walt Disney Animation Studios had been accomplished using the Reyes algorithm (Cook et al. 1987) in Pixar's RenderMan. More recently, ray-traced global illumination (GI) promised artists significant productivity gains by providing more immediate feedback during rendering and removing the significant data management burden associated with shadow maps and point clouds. However, initial attempts to render our existing production scenes with ray-traced GI were unsuccessful; incoherent access of texture maps inhibited shading of indirect ray hits, and we had difficulty fitting our scenes in memory as required by existing ray-traced renderers.

To overcome these limitations, we created a new rendering architecture which traces and shades rays in large batches, first sorting each batch for geometric coherence during scene traversal.

## Müller et al. 2017

Eurographics Symposium on Rendering 2017  
 P. Sander and M. Zwicker (Guest Editors)  
 Volume 36 (2017), Number 4

### Practical Path Guiding for Efficient Light-Transport Simulation

Thomas Müller<sup>1,2</sup> Markus Gross<sup>1,2</sup> Jan Novák<sup>2</sup>  
<sup>1</sup>ETH Zürich  
<sup>2</sup>Disney Research

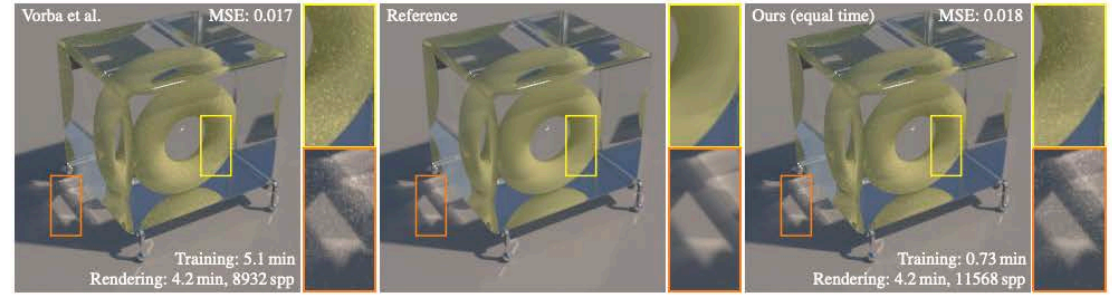


Figure 1: Our method allows efficient guiding of path-tracing algorithms as demonstrated in the TORUS scene. We compare equal-time (4.2 min) renderings of our method (right) to the current state-of-the-art [VKs'14, VK16] (left). Our algorithm automatically estimates how much training time is optimal, displays a rendering preview during training, and requires no parameter tuning. Despite being fully unidirectional, our method achieves similar MSE values compared to Vorba et al.'s method, which trains bidirectionally.

**Abstract**  
 We present a robust, unbiased technique for intelligent light-path construction in path-tracing algorithms. Inspired by existing path-guiding algorithms, our method learns an approximate representation of the scene's spatio-directional radiance field in an unbiased and iterative manner. To that end, we propose an adaptive spatio-directional hybrid data structure, referred to as SD-tree, for storing and sampling incident radiance. The SD-tree consists of an upper part—a binary tree that partitions the 3D spatial domain of the light field—and a lower part—a quadtree that partitions the 2D directional domain. We further present a principled way to automatically budget training and rendering computations to minimize the variance of the final image. Our method does not require tuning hyperparameters, although we allow limiting the memory footprint of the SD-tree. The aforementioned properties, its ease of implementation, and its stable performance make our method compatible with production environments. We demonstrate the merits of our method on scenes with difficult visibility, detailed geometry, and complex specular-glossy light transport, achieving better performance than previous state-of-the-art algorithms.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—; I.3.3 [Computer Graphics]: Picture/Image Generation—

BRDF  
 Visibility  
 Geometry

✗  
 ✗  
 ✓

✗  
 ✓  
 ✓

✓  
 ✓  
 ✓



# Related Work

BRDF

---

Visibility

---

Geometry

# Related Work

## Vévoda et al. 2018

**Bayesian online regression for adaptive direct illumination sampling**

PETR VÉVODA\*, Charles University, Prague and Render Legion, a. s.  
IVO KONDAPANENI\*, Charles University, Prague  
JAROSLAV KRIVÁNEK, Charles University, Prague and Render Legion, a. s.



No learning  
RMSE 0.0031 (0.0014 for DI only)

Our method  
RMSE 0.0012, 6.7x speedup (0.000062, 510x for DI only)

No learning  
Our method

Fig. 1. Equal-time comparison (60 s) of path-traced global illumination solutions computed using our learning-based direct illumination sampling method (right) and a baseline sampling method without learning (left). While both methods start off by sampling lights proportionally to rough estimates of their unoccluded contribution, our method progressively incorporates information about their actual contributions, including visibility, dramatically reducing image variance.

Direct illumination calculation is an important component of any physically-based renderer with a substantial impact on the overall performance. We present a novel adaptive solution for unbiased Monte Carlo direct illumination sampling, based on online learning of the light selection probability distributions. Our main contribution is a formulation of the learning process as Bayesian regression, based on a new, specifically designed statistical model of direct illumination. The net result is a set of regularization strategies to prevent over-fitting and ensure robustness even in early stages of calculation, when the observed information is sparse. The regression model captures spatial variation of illumination, which enables aggregating statistics over relatively large scene regions and, in turn, ensures a fast learning rate. We make the method scalable by adopting a light clustering strategy from the Lightcuts method, and further reduce variance through the use of control variates. As a main design feature, the resulting algorithm is virtually free of any preprocessing, which enables its use for interactive progressive rendering, while the online learning still enables super-linear convergence.

Additional Key Words and Phrases: direct illumination, adaptive sampling, visibility, learning.

**ACM Reference Format:**  
Petr Vévoda, Ivo Kondapaneni, and Jaroslav Krivánek. 2018. Bayesian online regression for adaptive direct illumination sampling. *ACM Trans. Graph.* 37, 4, Article 125 (August 2018), 12 pages. <https://doi.org/10.1145/3197517.3201340>

### 1 INTRODUCTION

Realistic rendering today is almost entirely based on Monte Carlo (MC) methods. The *indirect* illumination component has traditionally been held responsible for the undesirable image noise produced by such algorithms, which is probably why the *direct* illumination has received disproportionately less attention in research. However, many scenes in digital production feature complex lighting setups, and practical experience shows that it is often direct illumination that is responsible for the majority of image noise.

In this paper, we aim at unbiased direct illumination estimation for MC renderers. Specifically, we address the problem of randomly choosing an appropriate light source for a given scene location, so that variance of the direct illumination estimator is minimized. This could be achieved by choosing lights with probability proportional to their respective contributions, but these are unknown at the outset, they are costly to evaluate and difficult to predict. This is true especially due to the visibility, since it can be discontinuous and its evaluation involves expensive ray casting.

One possible solution would involve constructing the light sampling distributions in a preprocessing step [Georgiev et al. 2012]. However, long preprocessing disqualifies any form of *interactive*

CCS Concepts • **Computing methodologies** → **Rendering**; **Visibility**; **Machine learning**.

Authors' addresses: Petr Vévoda, Charles University, Prague, Malostranské náměstí 25, Prague, 11800, Render Legion, a. s. Karlovo náměstí 288/17, Prague, 12000, petrvevoda@seznam.cz; Ivo Kondapaneni, Charles University, Prague, Malostranské náměstí 25, Prague, 11800, ivo.kondapaneni@gmail.com; Jaroslav Krivánek, Charles University, Prague, Malostranské náměstí 25, Prague, 11800, Render Legion, a. s. Karlovo náměstí 288/17, Prague, 12000, jaroslav.krivank@mf.cuni.cz.

\*Petr Vévoda and Ivo Kondapaneni share the first authorship of this work.  
Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a

BRDF

Visibility

Geometry





# Related Work

## Vévoda et al. 2018

**Bayesian online regression for adaptive direct illumination sampling**

PETR VÉVODA\*, Charles University, Prague and Render Legion, a. s.  
 IVO KONDAPANENI\*, Charles University, Prague  
 JAROSLAV KRIVÁNEK, Charles University, Prague and Render Legion, a. s.

Fig. 1. Equal-time comparison (60 s) of path-traced global illumination solutions computed using our learning-based direct illumination sampling method (right) and a baseline sampling method without learning (left). While both methods start off by sampling lights proportionally to rough estimates of their unoccluded contribution, our method progressively incorporates information about their actual contributions, including visibility, dramatically reducing image variance.

Direct illumination calculation is an important component of any physically-based renderer with a substantial impact on the overall performance. We present a novel adaptive solution for unbiased Monte Carlo direct illumination sampling, based on online learning of the light selection probability distributions. Our main contribution is a formulation of the learning process as Bayesian regression, based on a new, specifically designed statistical model of direct illumination. The net result is a set of regularization strategies to prevent over-fitting and ensure robustness even in early stages of calculation, when the observed information is sparse. The regression model captures spatial variation of illumination, which enables aggregating statistics over relatively large scene regions and, in turn, ensures a fast learning rate. We make the method scalable by adopting a light clustering strategy from the Lightcuts method, and further reduce variance through the use of control variates. As a main design feature, the resulting algorithm is virtually free of any preprocessing, which enables its use for interactive progressive rendering, while the online learning still enables super-linear convergence.

CCS Concepts • **Computing methodologies** → **Rendering**; **Visibility**; **Machine learning**.

Authors' addresses: Petr Vévoda, Charles University, Prague, Malostranské náměstí 25, Prague, 11800, Render Legion, a. s. Karlovo náměstí 288/17, Prague, 12000, petrvevoda@seznam.cz; Ivo Kondapaneni, Charles University, Prague, Malostranské náměstí 25, Prague, 11800, ivo.kondapaneni@gmail.com; Jaroslav Krivánek, Charles University, Prague, Malostranské náměstí 25, Prague, 11800, Render Legion, a. s. Karlovo náměstí 288/17, Prague, 12000, jaroslav.krivank@mf.cuni.cz.

\*Petr Vévoda and Ivo Kondapaneni share the first authorship of this work. Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Additional Key Words and Phrases: direct illumination, adaptive sampling, visibility, learning.

ACM Reference Format: Petr Vévoda, Ivo Kondapaneni, and Jaroslav Krivánek. 2018. Bayesian online regression for adaptive direct illumination sampling. *ACM Trans. Graph.* 37, 4, Article 125 (August 2018), 12 pages. <https://doi.org/10.1145/3197517.3201340>

### 1 INTRODUCTION

Realistic rendering today is almost entirely based on Monte Carlo (MC) methods. The *indirect* illumination component has traditionally been held responsible for the undesirable image noise produced by such algorithms, which is probably why the *direct* illumination has received disproportionately less attention in research. However, many scenes in digital production feature complex lighting setups, and practical experience shows that it is often direct illumination that is responsible for the majority of image noise.

In this paper, we aim at unbiased direct illumination estimation for MC renderers. Specifically, we address the problem of randomly choosing an appropriate light source for a given scene location, so that variance of the direct illumination estimator is minimized. This could be achieved by choosing lights with probability proportional to their respective contributions, but these are unknown at the outset, they are costly to evaluate and difficult to predict. This is true especially due to the visibility, since it can be discontinuous and its evaluation involves expensive ray casting.

One possible solution would involve constructing the light sampling distributions in a preprocessing step [Georgiev et al. 2012]. However, long preprocessing disqualifies any form of *interactive*

## Estevez & Kulla 2018

**Importance Sampling of Many Lights with Adaptive Tree Splitting**

ALEJANDRO CONTY ESTEVEZ, Sony Pictures Imageworks  
 CHRISTOPHER KULLA, Sony Pictures Imageworks

Fig. 1. A procedural city with 363,036 lights, one GI bounce and participating media. Rendered with 16 samples per pixel, each shading point takes an average of 7 shadow rays (45 for the volume integral). We shoot an average of 1700 rays per pixel. The image rendered in 20 minutes on a quad core Intel i7.

We present a technique to importance sample large collections of lights (including mesh lights as collections of small emitters) in the context of Monte-Carlo path tracing. A bounding volume hierarchy over all emitters is traversed at each shading point using a single random number in a way that importance samples their predicted contribution. The tree aggregates energy, spatial and orientation information from the emitters to enable accurate prediction of the effect of a cluster of lights on any given shading point. We further improve the performance of the algorithm by forcing splitting until the importance of a cluster is sufficiently representative of its contents.

CCS Concepts: • **Computing methodologies** → **Ray tracing**;

Additional Key Words and Phrases: illumination, ray tracing, many lights

ACM Reference Format: Alejandro Conty Estevez and Christopher Kulla. 2018. Importance Sampling of Many Lights with Adaptive Tree Splitting. *Proc. ACM Comput. Graph. Interact. Tech.* 1, 2, Article 25 (August 2018), 17 pages. <https://doi.org/10.1145/3233305>

### 1 INTRODUCTION

Direct lighting calculations are a critical part of modern path tracing renderers with next event estimation. While sampling from simple light shapes [Shirley et al. 1996] is well understood, relatively little attention has been devoted to the problem of efficiently sampling from large collections of such shapes. In production renderers, this problem appears both in the form of scenes

BRDF  
 Visibility  
 Geometry





# Related Work

Vévoda et al. 2018

Estevez & Kulla 2018

Moreau et al. 2019

**Bayesian online regression for adaptive direct illumination sampling**

PETR VÉVODA<sup>\*</sup>, Charles University, Prague and Render Legion, a. s.  
IVO KONDAPANENI<sup>\*</sup>, Charles University, Prague  
JAROSLAV KRIVÁNEK, Charles University, Prague and Render Legion, a. s.

Fig. 1. Equal-time comparison (60 s) of path-traced global illumination solutions computed using our learning-based direct illumination sampling method (right) and a baseline sampling method without learning (left). While both methods start off by sampling lights proportionally to rough estimates of their unoccluded contribution, our method progressively incorporates information about their actual contributions, including visibility, dramatically reducing image variance.

Additional Key Words and Phrases: direct illumination, adaptive sampling, visibility, learning.

ACM Reference Format: Petr Vévoda, Ivo Kondapaneni, and Jaroslav Krivánek. 2018. Bayesian online regression for adaptive direct illumination sampling. *ACM Trans. Graph.* 37, 4, Article 125 (August 2018), 12 pages. <https://doi.org/10.1145/3197517.3201340>

**1 INTRODUCTION**

Realistic rendering today is almost entirely based on Monte Carlo (MC) methods. The *indirect* illumination component has traditionally been held responsible for the undesirable image noise produced by such algorithms, which is probably why the *direct* illumination has received disproportionately less attention in research. However, many scenes in digital production feature complex lighting setups, and practical experience shows that it is often direct illumination that is responsible for the majority of image noise.

In this paper, we aim at unbiased direct illumination estimation for MC renderers. Specifically, we address the problem of randomly choosing an appropriate light source for a given scene location, so that variance of the direct illumination estimator is minimized. This could be achieved by choosing lights with probability proportional to their respective contributions, but these are unknown at the outset, they are costly to evaluate and difficult to predict. This is true especially due to the visibility, since it can be discontinuous and its evaluation involves expensive ray casting.

One possible solution would involve constructing the light sampling distributions in a preprocessing step [Georgiev et al. 2012]. However, long preprocessing disqualifies any form of *interactive*

CCS Concepts • **Computing methodologies** → **Rendering**; **Visibility**; **Machine learning**.

Authors' addresses: Petr Vévoda, Charles University, Prague, Malostranské náměstí 25, Prague, 11800, Render Legion, a. s. Karlovo náměstí 288/17, Prague, 12000, petrvevoda@seznam.cz; Ivo Kondapaneni, Charles University, Prague, Malostranské náměstí 25, Prague, 11800, ivo.kondapaneni@gmail.com; Jaroslav Krivánek, Charles University, Prague, Malostranské náměstí 25, Prague, 11800, Render Legion, a. s. Karlovo náměstí 288/17, Prague, 12000, jaroslav.krivank@imf.cuni.cz.

<sup>\*</sup>Petr Vévoda and Ivo Kondapaneni share the first authorship of this work. Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

**Importance Sampling of Many Lights with Adaptive Tree Splitting**

ALEJANDRO CONTY ESTEVEZ, Sony Pictures Imageworks  
CHRISTOPHER KULLA, Sony Pictures Imageworks

Fig. 1. A procedural city with 363,036 lights, one GI bounce and participating media. Rendered with 16 samples per pixel, each shading point takes an average of 7 shadow rays (45 for the volume integral). We shoot an average of 1700 rays per pixel. The image rendered in 20 minutes on a quad core Intel i7.

We present a technique to importance sample large collections of lights (including mesh lights as collections of small emitters) in the context of Monte-Carlo path tracing. A bounding volume hierarchy over all emitters is traversed at each shading point using a single random number in a way that importance samples their predicted contribution. The tree aggregates energy, spatial and orientation information from the emitters to enable accurate prediction of the effect of a cluster of lights on any given shading point. We further improve the performance of the algorithm by forcing splitting until the importance of a cluster is sufficiently representative of its contents.

CCS Concepts: • **Computing methodologies** → **Ray tracing**.

Additional Key Words and Phrases: illumination, ray tracing, many lights

ACM Reference Format: Alejandro Conty Estevez and Christopher Kulla. 2018. Importance Sampling of Many Lights with Adaptive Tree Splitting. *Proc. ACM Comput. Graph. Interact. Tech.* 1, 2, Article 25 (August 2018), 17 pages. <https://doi.org/10.1145/3233305>

**1 INTRODUCTION**

Direct lighting calculations are a critical part of modern path tracing renderers with next event estimation. While sampling from simple light shapes [Shirley et al. 1996] is well understood, relatively little attention has been devoted to the problem of efficiently sampling from large collections of such shapes. In production renderers, this problem appears both in the form of scenes

**Dynamic Many-Light Sampling for Real-Time Ray Tracing**

P. Moreau<sup>1,2</sup>, M. Pharr<sup>1</sup> and P. Clarberg<sup>1</sup>

<sup>1</sup>NVIDIA  
<sup>2</sup>Lund University, Sweden

**Abstract**

Monte Carlo ray tracing offers the capability of rendering scenes with large numbers of area light sources—lights can be sampled stochastically and shadowing can be accounted for by tracing rays, rather than using shadow maps or other rasterization-based techniques that do not scale to many lights or work well with area lights. Current GPUs only afford the capability of tracing a few rays per pixel at real-time frame rates, making it necessary to focus sampling on important light sources. While state-of-the-art algorithms for offline rendering build hierarchical data structures over the light sources that enable sampling them according to their importance, they lack efficient support for dynamic scenes. We present a new algorithm for maintaining hierarchical light sampling data structures targeting real-time rendering. Our approach is based on a two-level BVH hierarchy that reduces the cost of partial hierarchy updates. Performance is further improved by updating lower-level BVHs via refitting, maintaining their original topology. We show that this approach can give error within 6% of recreating the entire hierarchy from scratch at each frame, while being two orders of magnitude faster, requiring less than 1 ms per frame for hierarchy updates for a scene with thousands of moving light sources on a modern GPU. Further, we show that with spatiotemporal filtering, our approach allows complex scenes with thousands of lights to be rendered with ray-traced shadows in 16.1 ms per frame.

CCS Concepts  
• **Computing methodologies** → **Ray tracing**;

**1. Introduction**

Complex illumination is a critical ingredient for the visual richness of rendered images. Images that include the soft shadows and diffused lighting that is characteristic of large area light sources have a markedly more realistic appearance than images rendered with small numbers of point or directional light sources, which give stark and harsh lighting effects. However, with more than few light sources it is infeasible to shade them all individually, especially under the constraints of real-time rendering. Culling and/or stochastic selection of a subset of lights is necessary. In this work, we focus on stochastic sampling in order to be able to support many contributing light sources and still compute unbiased results.

With this approach, it is necessary to define a discrete probability density function (PDF)  $p_i(\mathbf{x}, i)$  that gives the probability of sampling the  $i$ th light as seen from a point  $\mathbf{x}$  in the scene. The more closely proportional  $p_i(\mathbf{x}, i)$  is to the reflected light at  $\mathbf{x}$  due to the light  $i$ 's emission, the less error will be present in the image. Unfortunately, an accurate  $p_i$  cannot be easily precomputed as there are millions of shading points  $\mathbf{x}$ , a scene may have tens of thousands of lights  $i$ , and generally, the optimal sampling distribution varies

$\mathbf{x}$ , the tree is stochastically traversed [KWR<sup>+</sup>17, CEK18]. At each level of the traversal, the relative contributions of the children nodes are estimated such that the full distribution  $p_i$  is never represented explicitly and only  $\log(n)$  computations per shading point (where  $n$  is the number of lights) are required. This idea is illustrated in Figure 1. For offline rendering, the cost of constructing the light BVH is negligible compared to the rendering time. This is not the case for real-time rendering, where many fewer rays are generally traced per frame and no more than a few milliseconds per frame are available. The goal of this paper is to adapt light BVH methods to be suitable for real-time ray tracing of dynamic scenes. We make the following contributions:

- We organize light sources in multiple bounding volume hierarchies, arranged in a two-level hierarchy.
- We show that refitting light BVHs without modifying their topology can be implemented efficiently on the GPU, and that this approach works well for moderate amounts of light source motion.
- We demonstrate that top-level BVHs can be rebuilt asynchronously to maintain close-to-optimal overall tree topology.

BRDF  
Visibility  
Geometry

✗  
✓  
✓

✗  
✗  
✓

✗  
✗  
✓



# Related Work

- Overhead at low sample counts
- Sample only parts of product
- Maintain complex data structures

# ReSTIR Overview

# ReSTIR Overview

- Approximately sample full product



# ReSTIR Overview

- Approximately sample full product
- No complex data structures

# ReSTIR Overview

- Approximately sample full product
- No complex data structures
- Constant memory

# ReSTIR Overview

- Approximately sample full product
- No complex data structures
- Constant memory
- Fixed computation per frame

# ReSTIR Overview

- Approximately sample full product
- No complex data structures
- Constant memory
- Fixed computation per frame
- *Simple* algorithm



# Method

# Importance Resampling (RIS)

[Talbot et al. '05]

- Approximately sample *arbitrary* target distribution  $\hat{p}(x)$
- Can be unnormalized

# Importance Resampling (RIS)

[Talbot et al. '05]

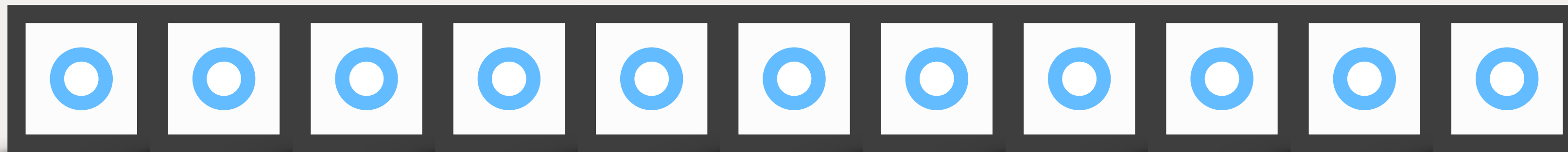
- Start with *candidate* distribution  $p(x)$ ...

# Importance Resampling (RIS)

[Talbot et al. '05]

- Start with *candidate* distribution  $p(x) \dots$

$M$  "candidates"  $x_i \propto p(x_i)$



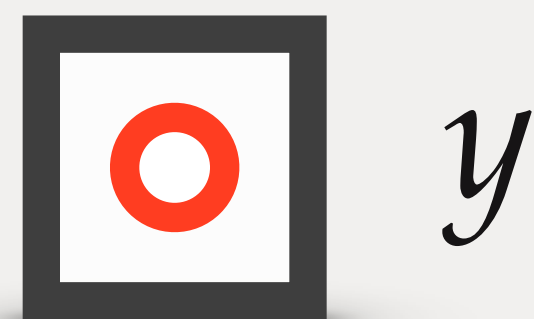
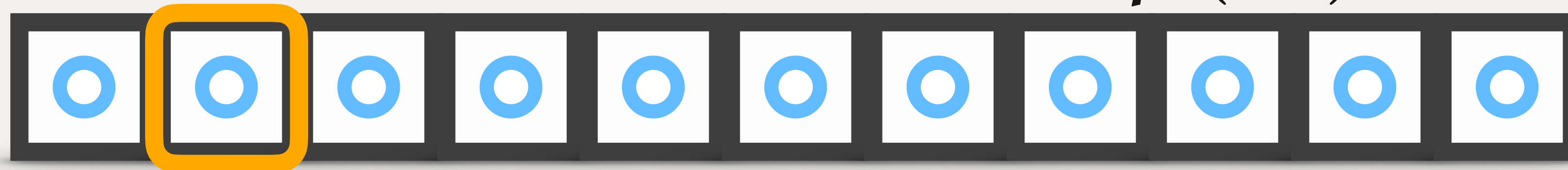


# Importance Resampling (RIS)

[Talbot et al. '05]

- Start with *candidate* distribution  $p(x)$ ...

$M$  "candidates"  $x_i \propto p(x_i)$

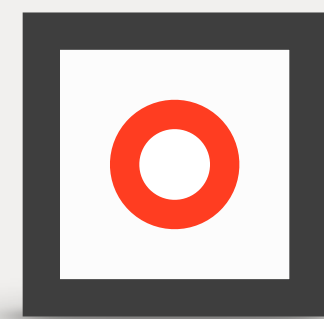
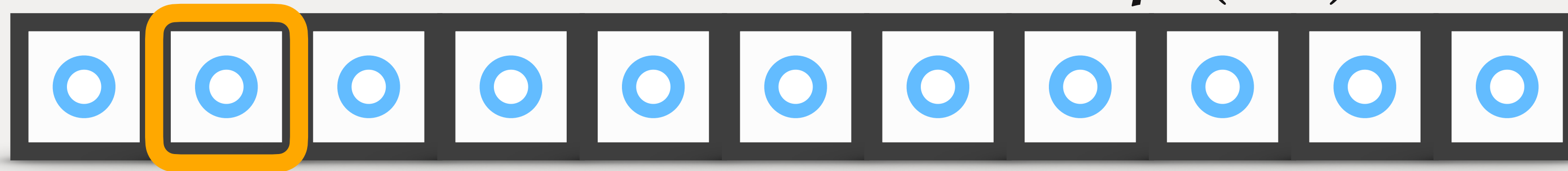


# Importance Resampling (RIS)

[Talbot et al. '05]

- Start with *candidate* distribution  $p(x) \dots$

$M$  "candidates"  $x_i \propto p(x_i)$



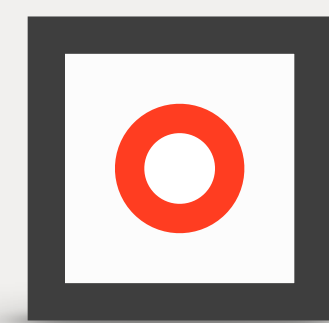
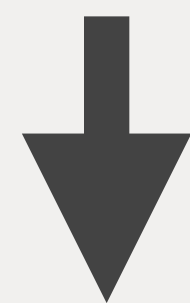
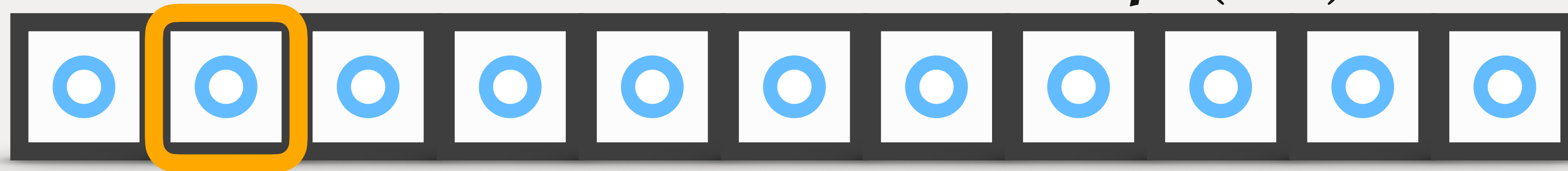
$y \approx \hat{p}(y)$

# Importance Resampling (RIS)

[Talbot et al. '05]

- Start with *candidate* distribution  $p(x)$ ...

$M$  "candidates"  $x_i \propto p(x_i)$

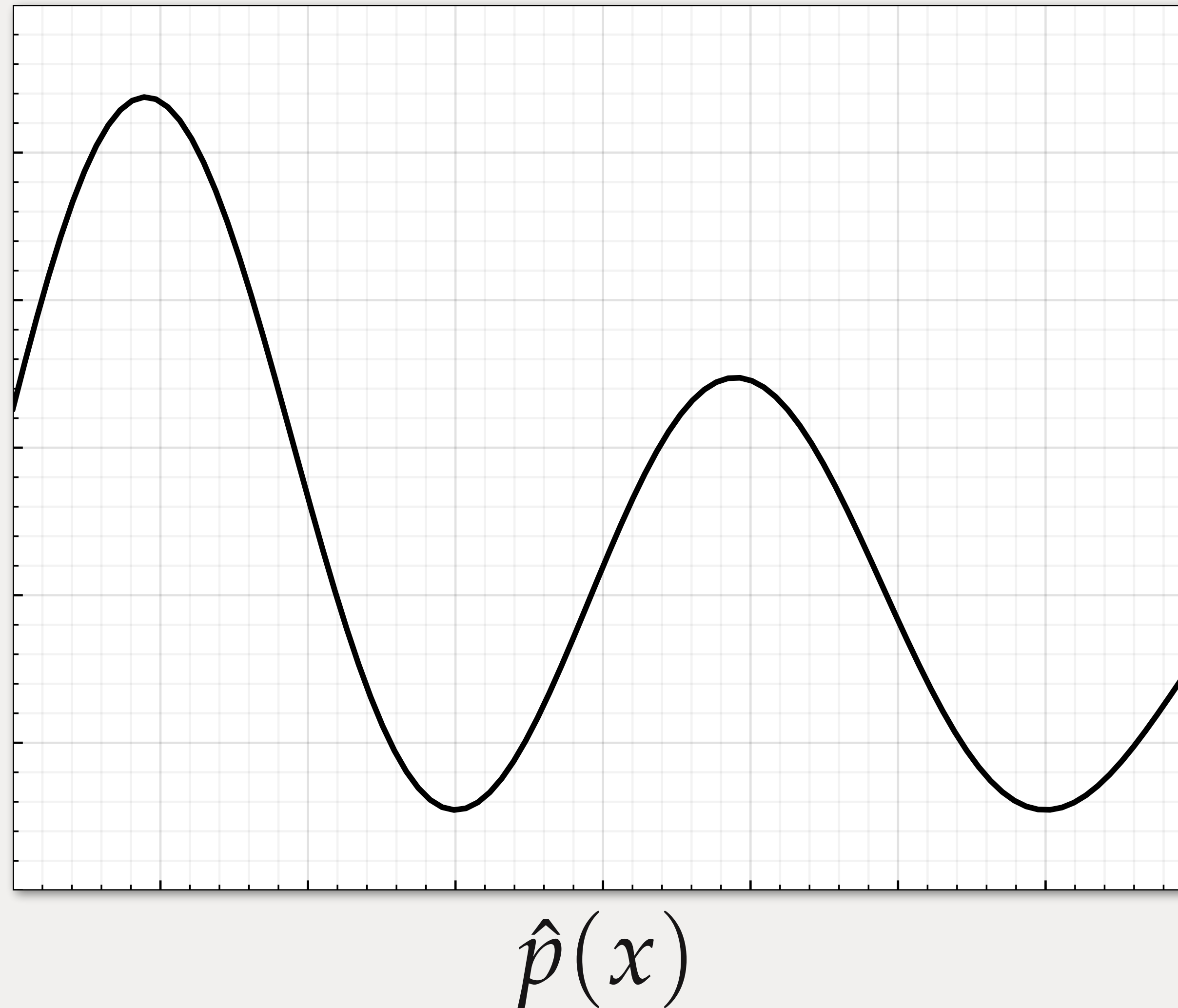


$y \approx \hat{p}(y)$

$$w(x_i) = \frac{\hat{p}(x_i)}{p(x_i)}$$

# Importance Resampling (RIS)

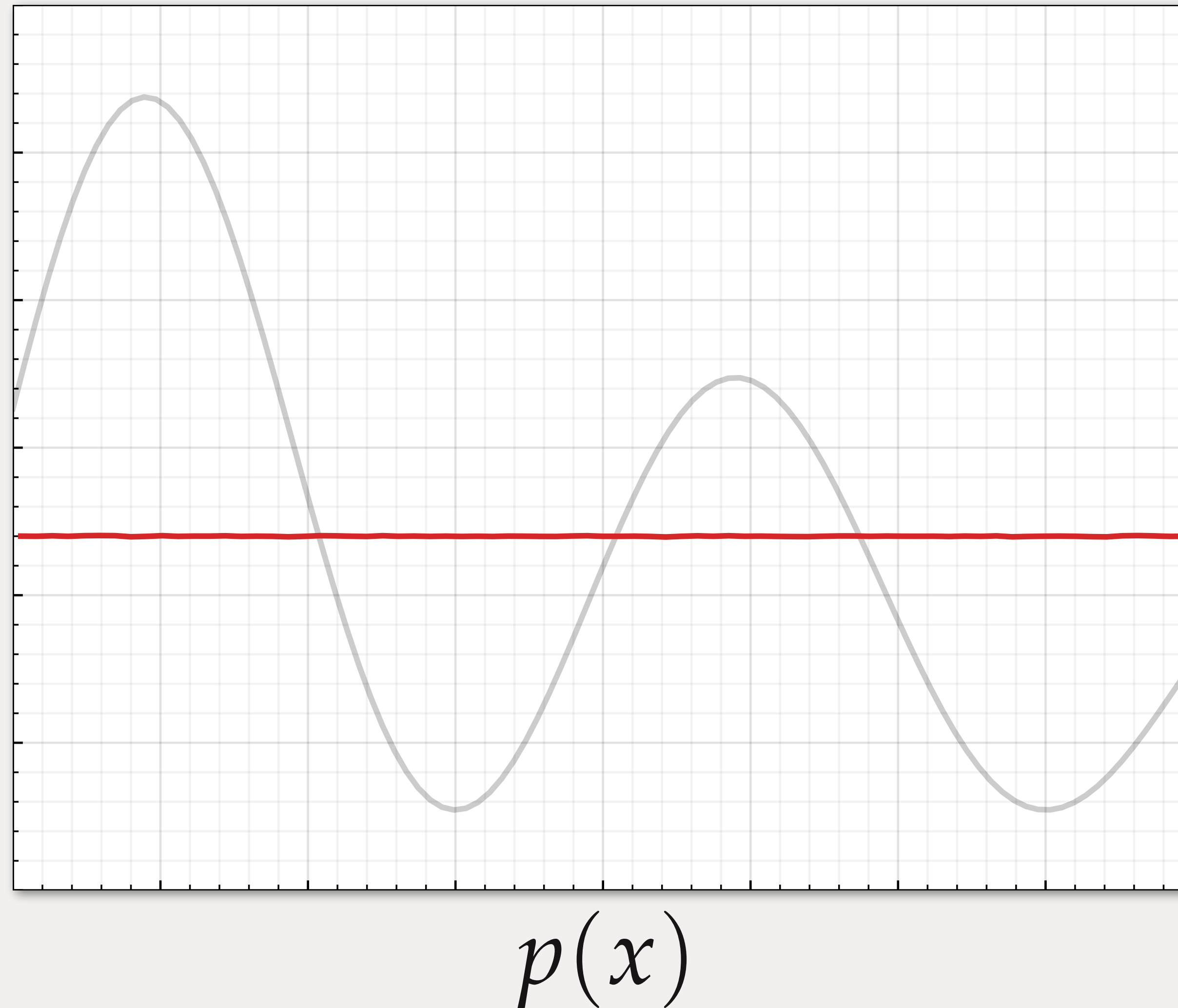
[Talbot et al. '05]





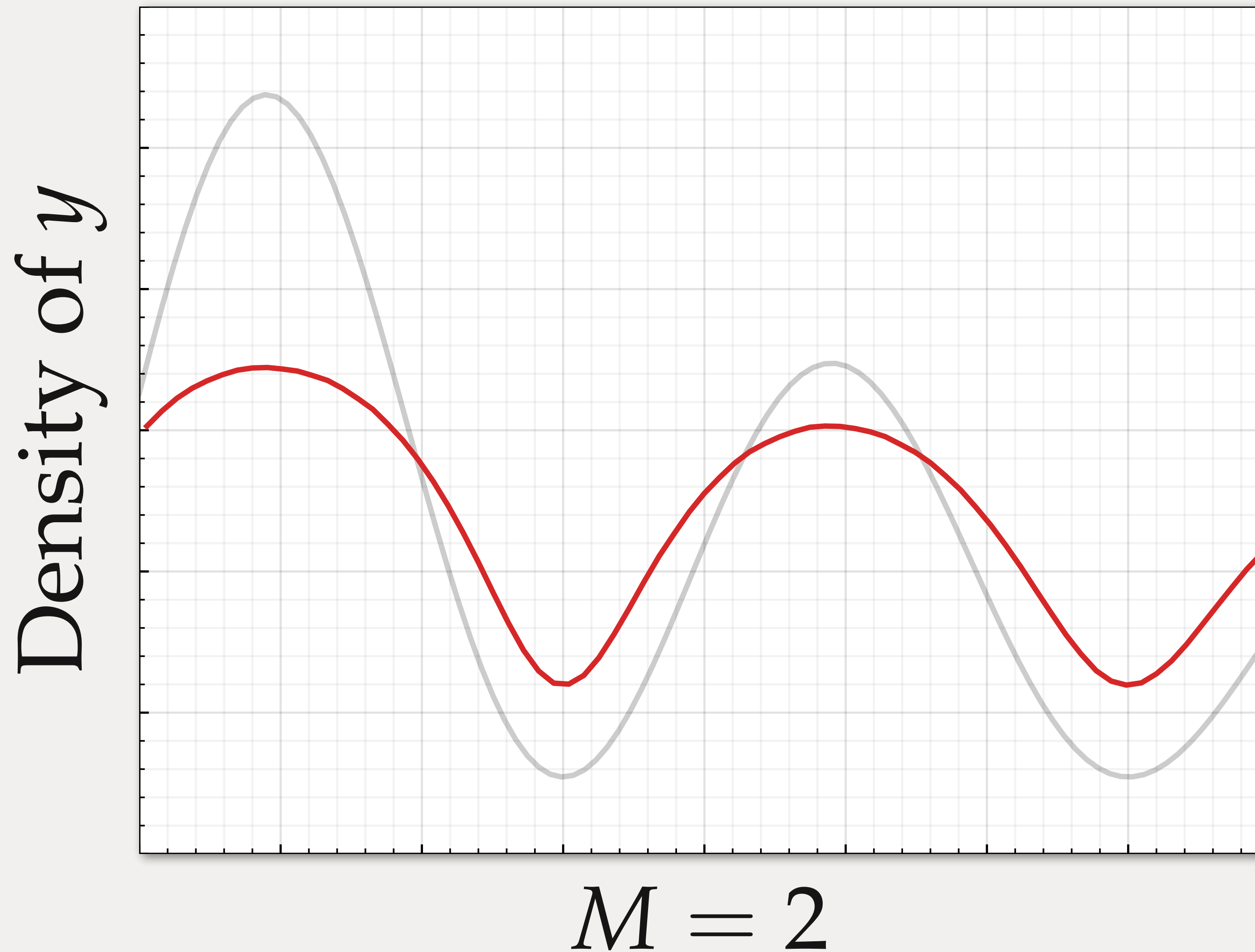
# Importance Resampling (RIS)

[Talbot et al. '05]



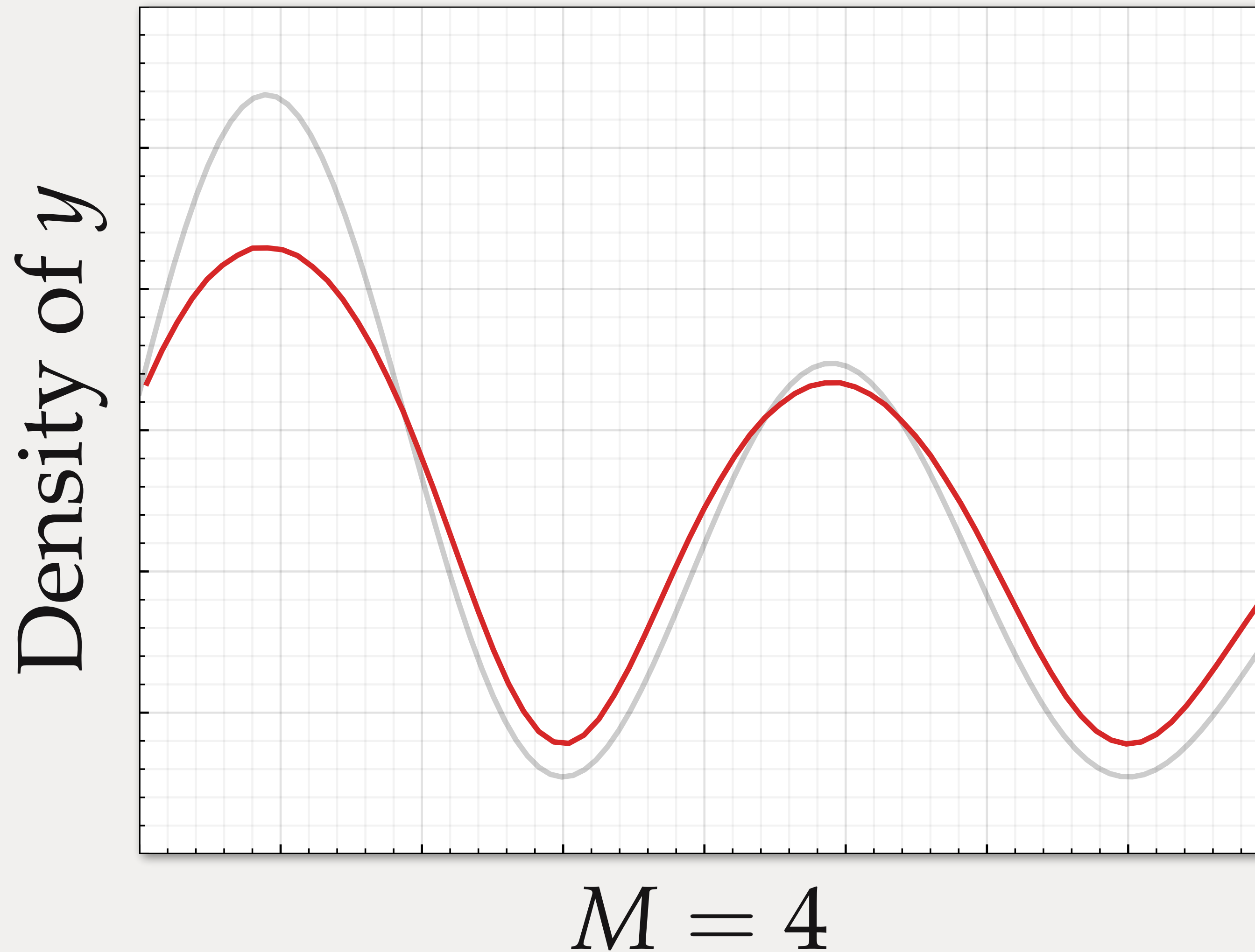
# Importance Resampling (RIS)

[Talbot et al. '05]



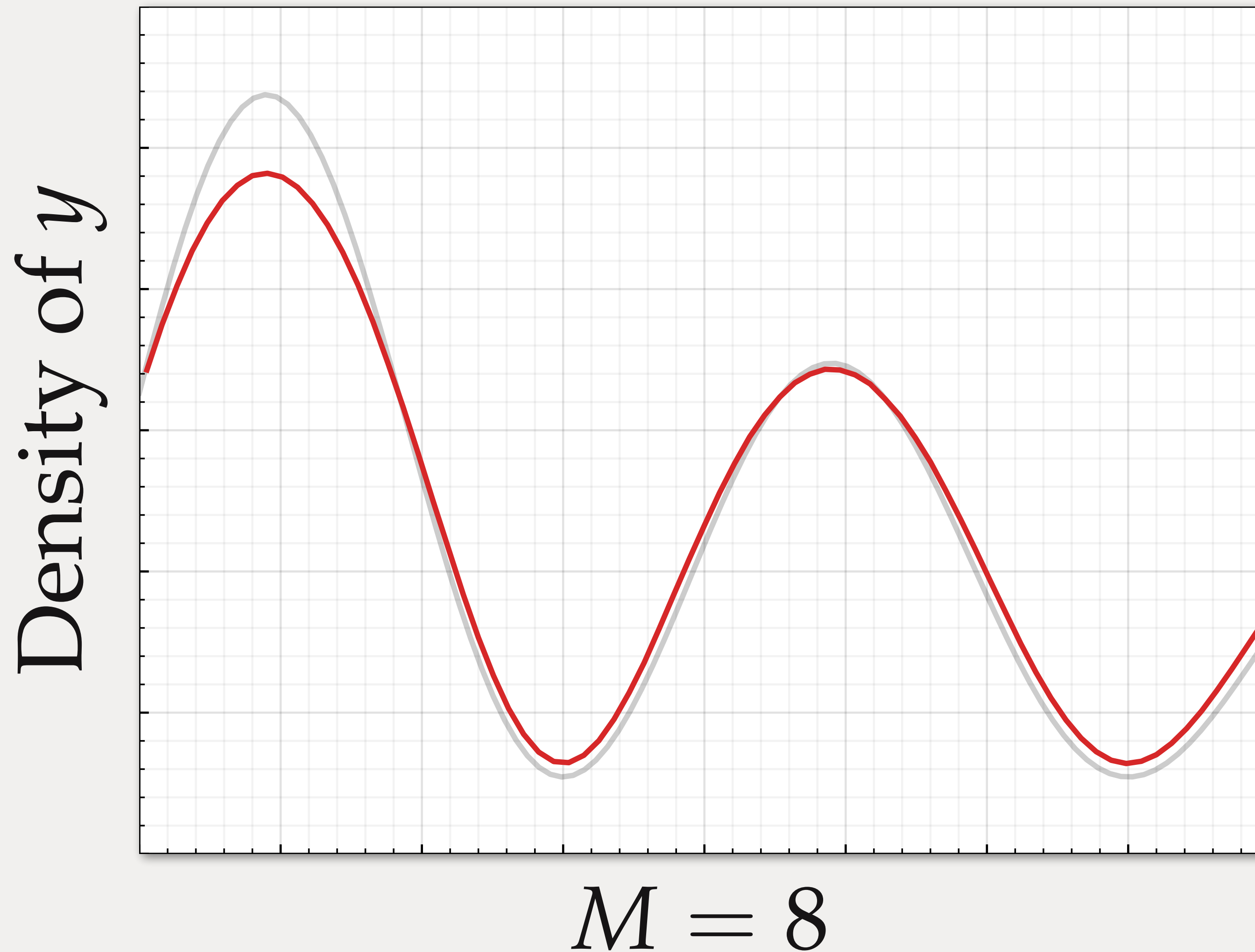
# Importance Resampling (RIS)

[Talbot et al. '05]



# Importance Resampling (RIS)

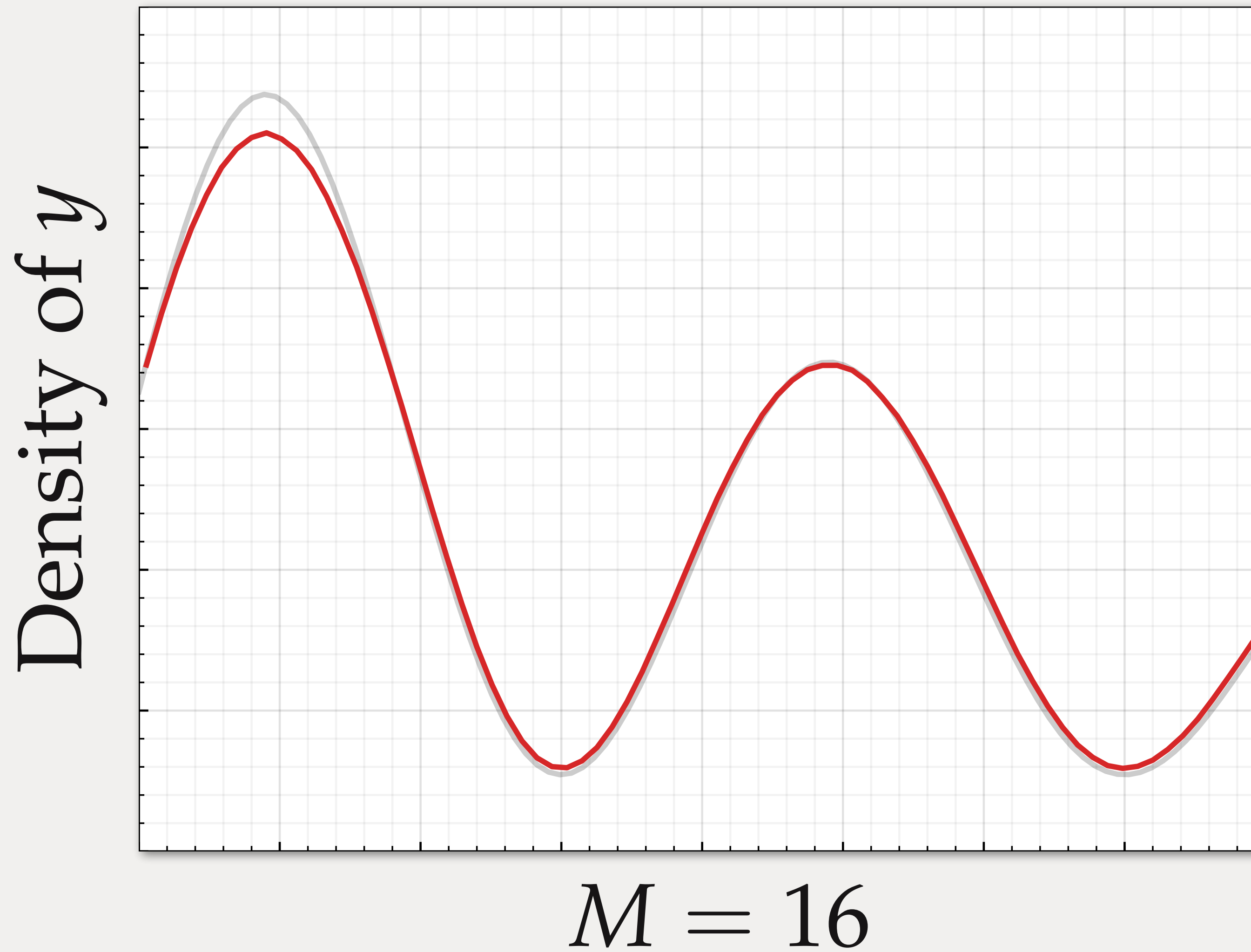
[Talbot et al. '05]





# Importance Resampling (RIS)

[Talbot et al. '05]



# Importance Resampling (RIS)

[Talbot et al. '05]

$$\int_{\Omega} f(x) dx$$

# Importance Resampling (RIS)

[Talbot et al. '05]

$$\int_{\Omega} f(x) dx \approx f(y) \cdot W$$

# Importance Resampling (RIS)

[Talbot et al. '05]

$$\int_{\Omega} f(x) dx \approx f(y) \cdot W$$

$$W = \frac{1}{\hat{p}(y)} \cdot \left( \frac{1}{M} \sum_{i=1}^M w(x_i) \right)$$



# Importance Resampling (RIS)

[Talbot et al. '05]

$$\int_{\Omega} f(x) dx \approx f(y) \cdot W$$

$$W = \frac{1}{\hat{p}(y)} \cdot \left( \frac{1}{M} \sum_{i=1}^M w(x_i) \right)$$

Unbiased if  $p(x) > 0$  where  $\hat{p}(x) > 0$

# Resampled Direct Lighting

$$p(x) \propto L_e(x)$$

# Resampled Direct Lighting

$$p(x) \propto L_e(x)$$

$$\hat{p}(x) = L_e(x) \rho(x) \frac{\cos \theta_i \cos \theta_o}{r^2}$$

# Importance Resampling (RIS)

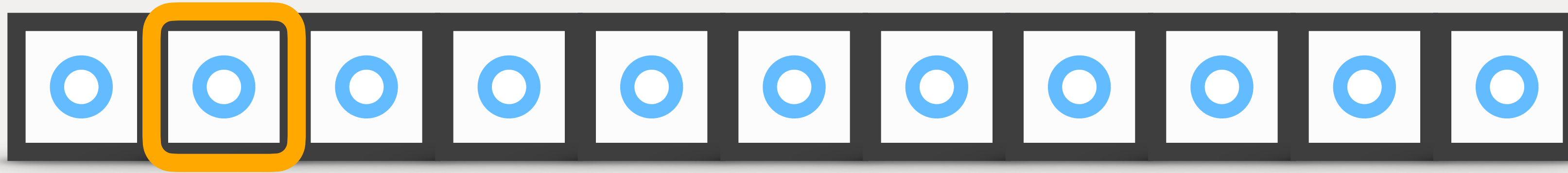
[Talbot et al. '05]



# Importance Resampling (RIS)

[Talbot et al. '05]

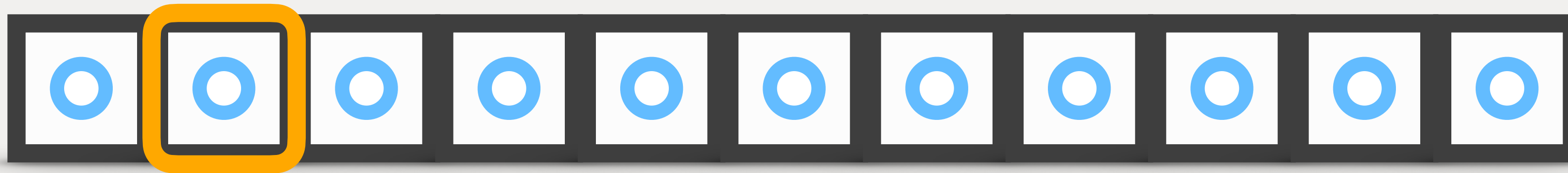
- Sampling candidates requires *storing* them



# Importance Resampling (RIS)

[Talbot et al. '05]

- Sampling candidates requires *storing* them

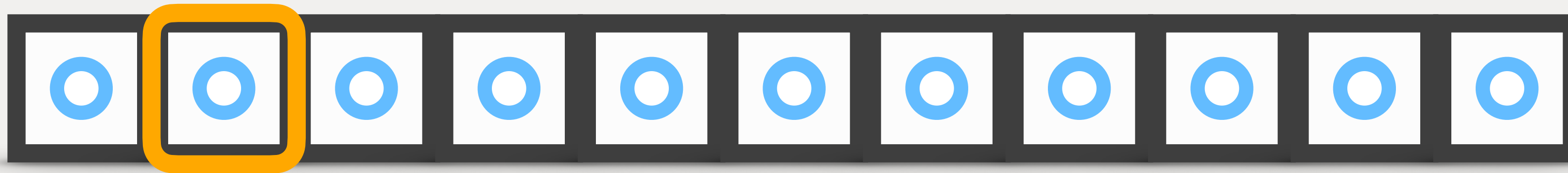


- Not GPU friendly

# Importance Resampling (RIS)

[Talbot et al. '05]

- Sampling candidates requires *storing* them

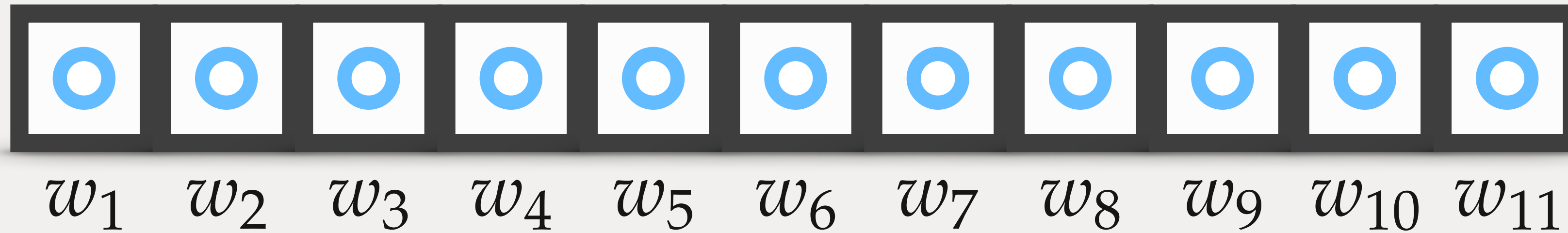


- Not GPU friendly
- Can we do better?

# Weighted Reservoir Sampling

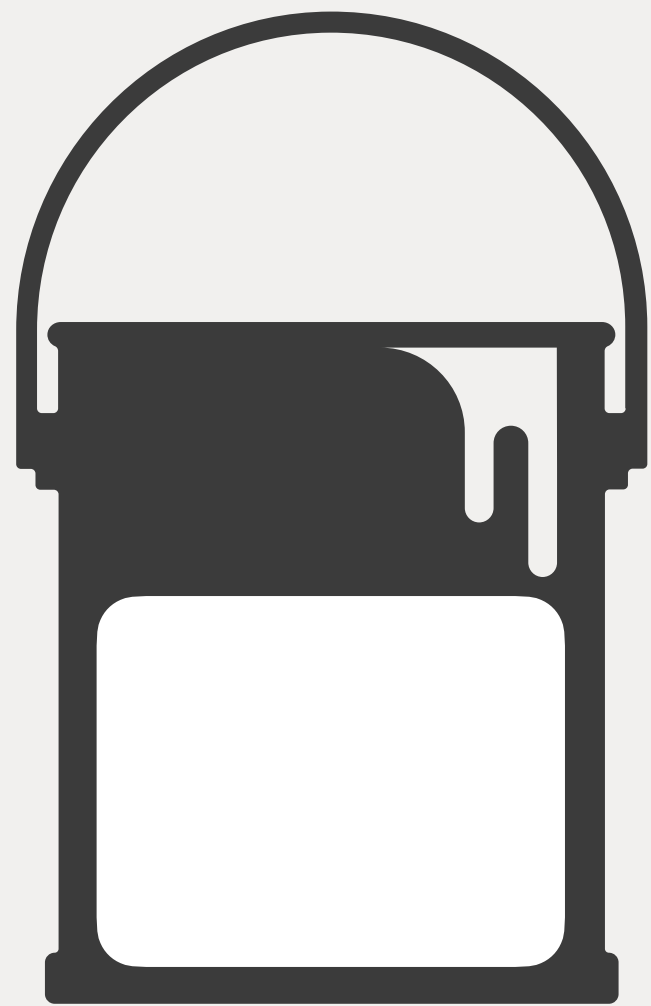
[M. T. Chao 1982]

- Stream-based sampling algorithm



# Weighted Reservoir Sampling

[M. T. Chao 1982]





# Weighted Reservoir Sampling

[M. T. Chao 1982]



$w_1$

# Weighted Reservoir Sampling

[M. T. Chao 1982]



$w_1$

# Weighted Reservoir Sampling

[M. T. Chao 1982]



$w_1$

# Weighted Reservoir Sampling

[M. T. Chao 1982]



$w_1$

# Weighted Reservoir Sampling

[M. T. Chao 1982]



$w_1$



# Weighted Reservoir Sampling

[M. T. Chao 1982]



$w_1$

# Weighted Reservoir Sampling

[M. T. Chao 1982]



$w_4$

# Weighted Reservoir Sampling

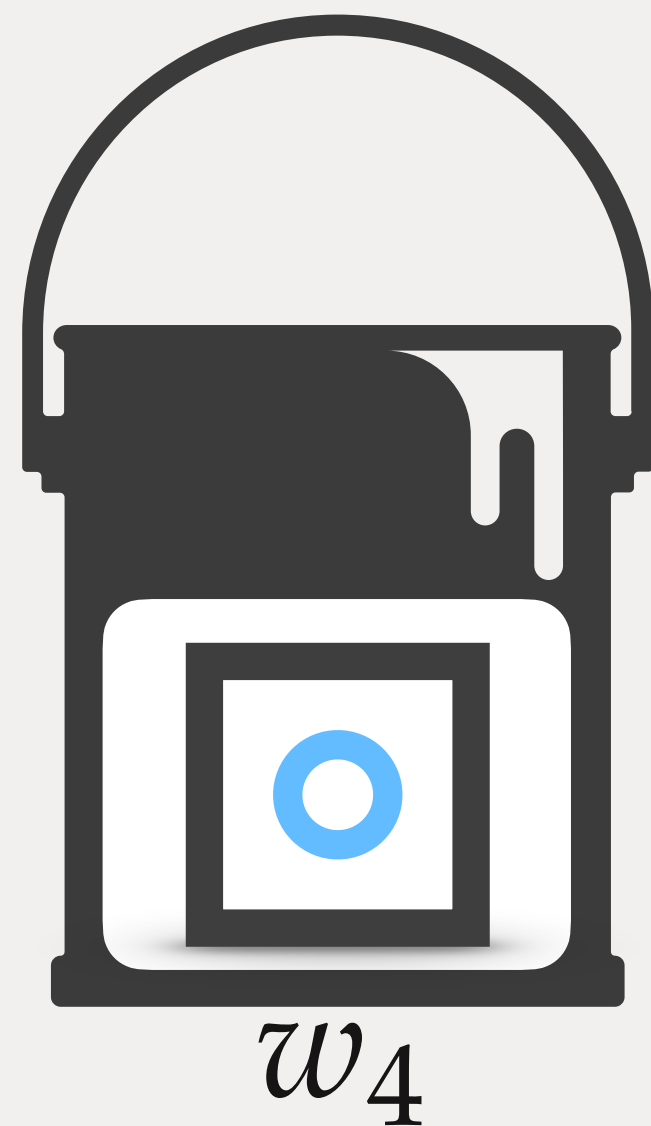
[M. T. Chao 1982]



$w_4$

# Weighted Reservoir Sampling

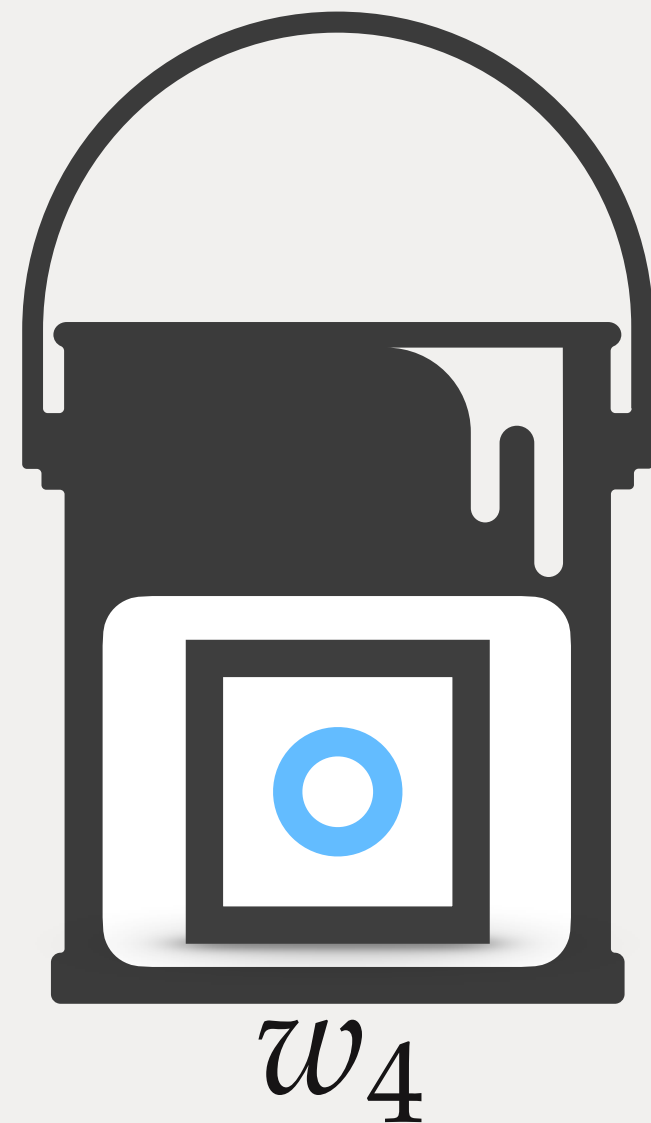
[M. T. Chao 1982]



$$\mathbb{P}[\text{Accept } i] = \frac{w_i}{\sum_j w_j}$$

# Weighted Reservoir Sampling

[M. T. Chao 1982]

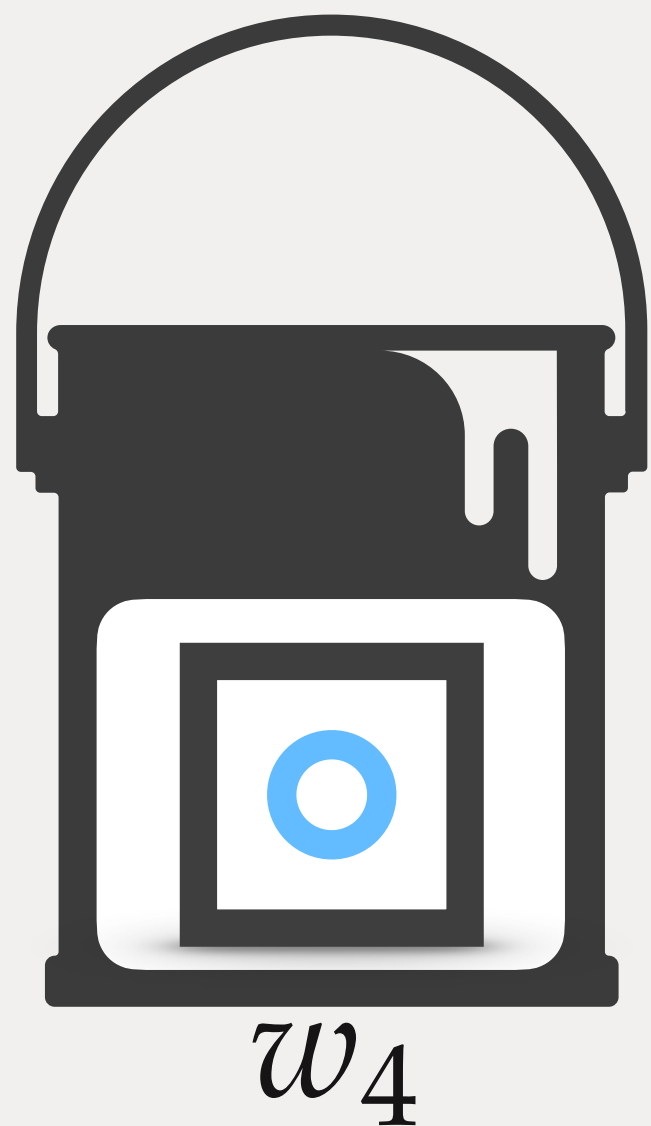


$$\mathbb{P}[\text{Accept } i] = \frac{w_i}{\sum_j w_j}$$



# Weighted Reservoir Sampling

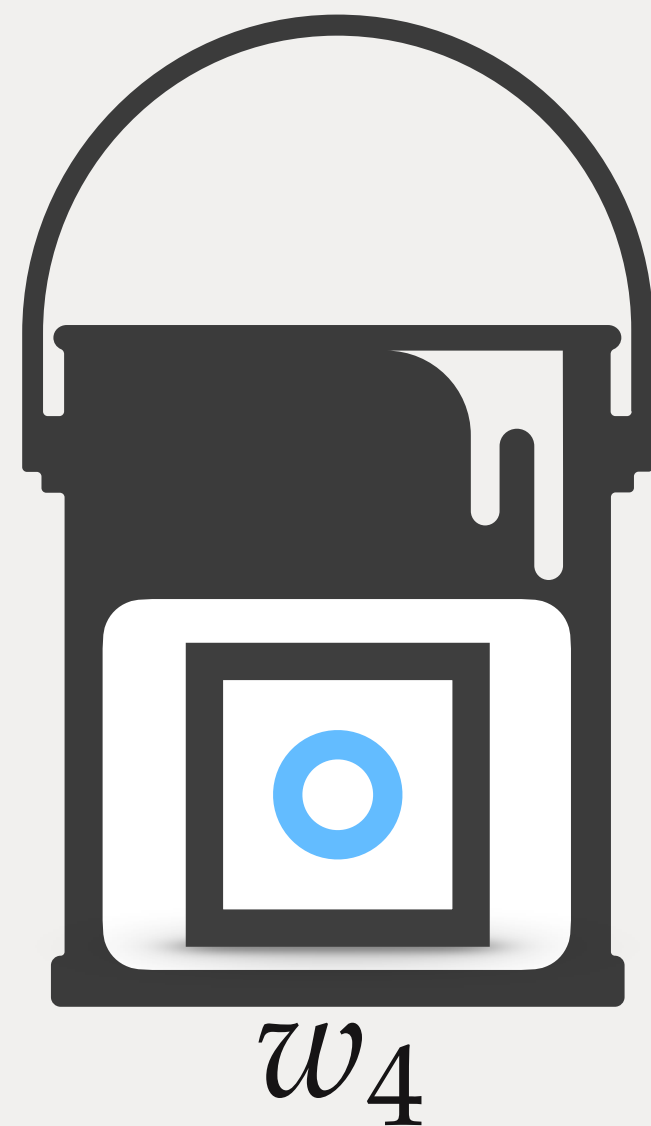
[M. T. Chao 1982]



$$\mathbb{P}[\text{Accept } i] = \frac{w_i}{\sum_j w_j}$$

# Weighted Reservoir Sampling

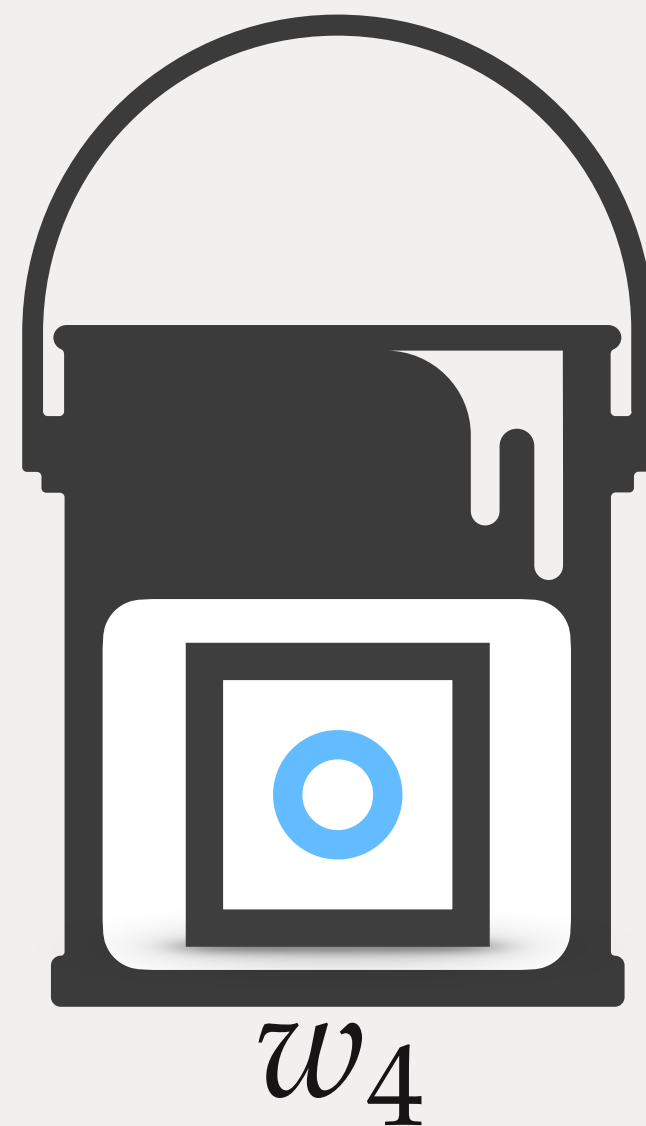
[M. T. Chao 1982]



$$\mathbb{P}[\text{Accept } i] = \frac{w_i}{\sum_j w_j}$$

# Weighted Reservoir Sampling

[M. T. Chao 1982]



$$\mathbb{P}[\text{Accept } i] = \frac{w_i}{\sum_j w_j}$$

# Reservoir Resampling

# Reservoir Resampling

- Constant memory



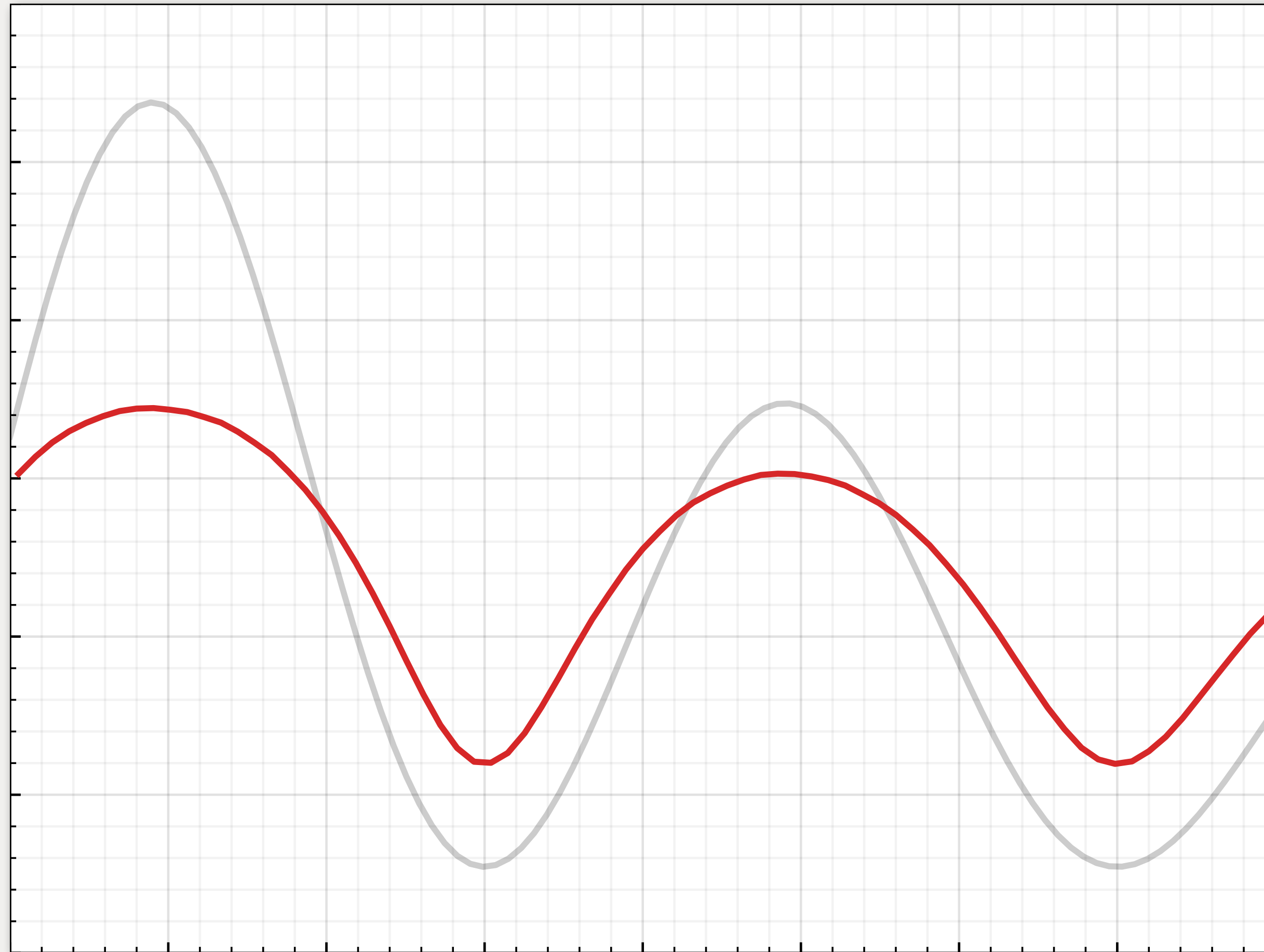
# Reservoir Resampling

- Constant memory
- Computation:  $O(M)$

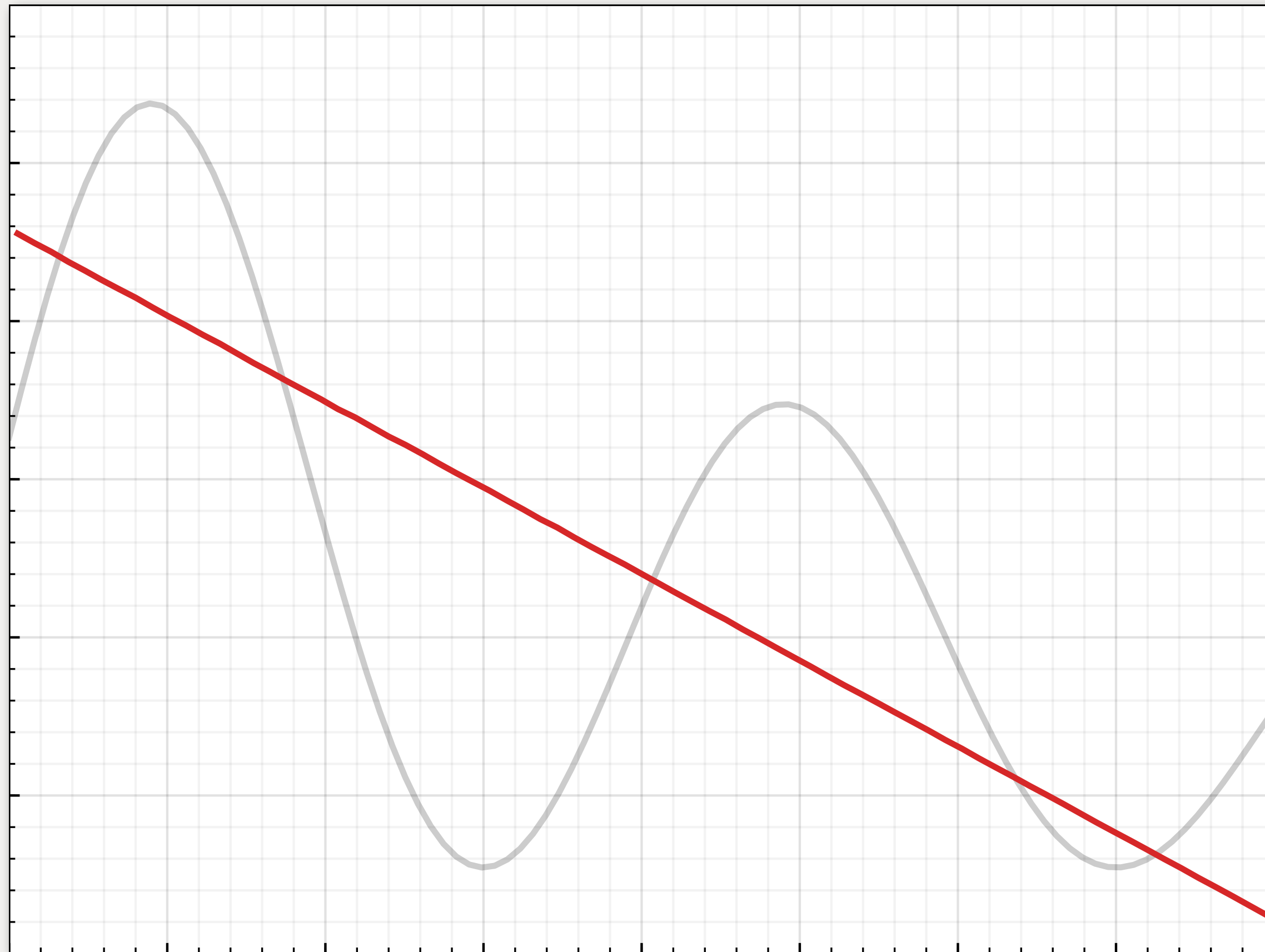
# Reservoir Resampling

- Constant memory
- Computation:  $O(M)$
- Can we do better?

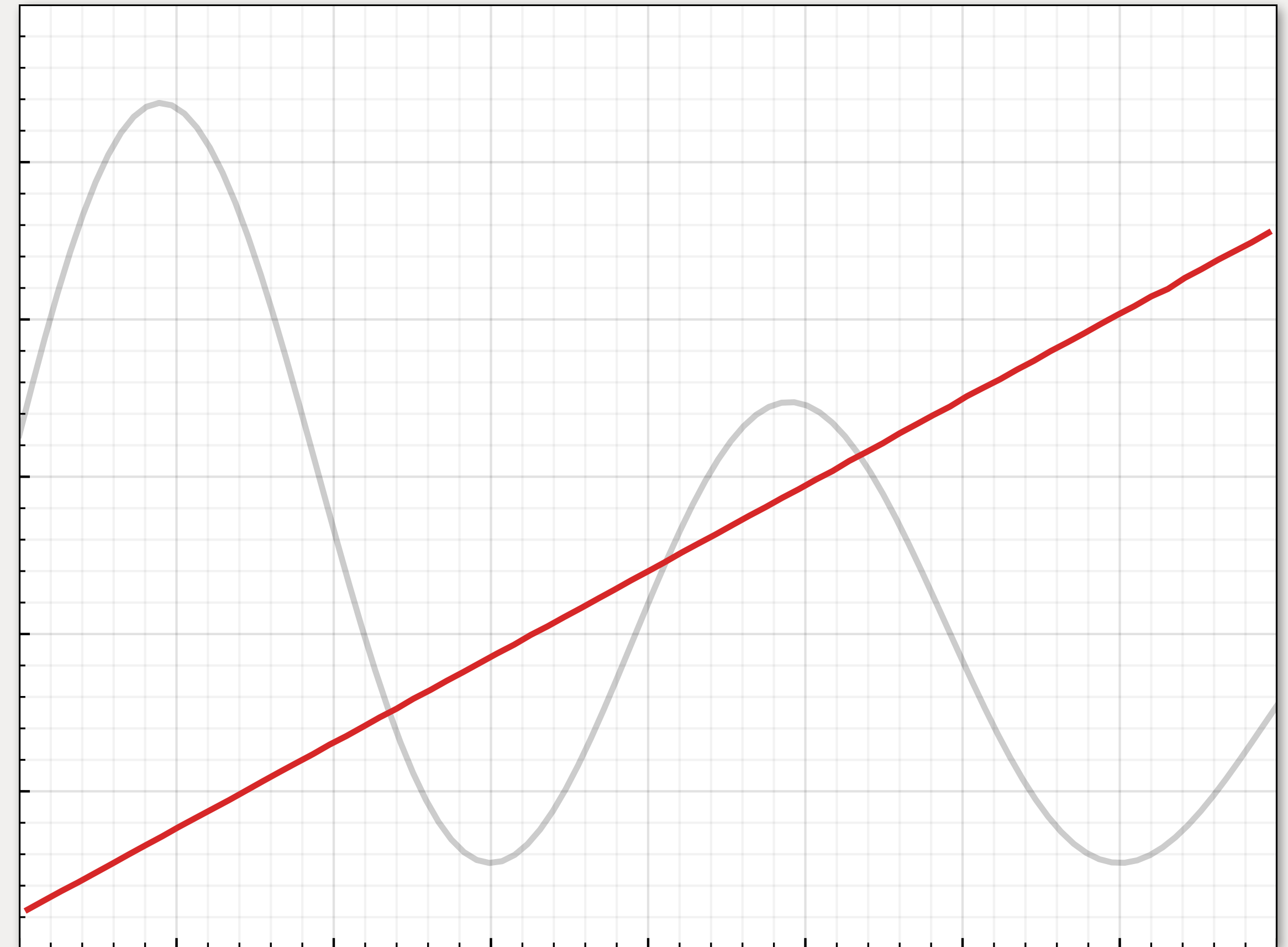
# Reservoir Resampling



# Improved Reservoir Resampling



$p_1(x)$



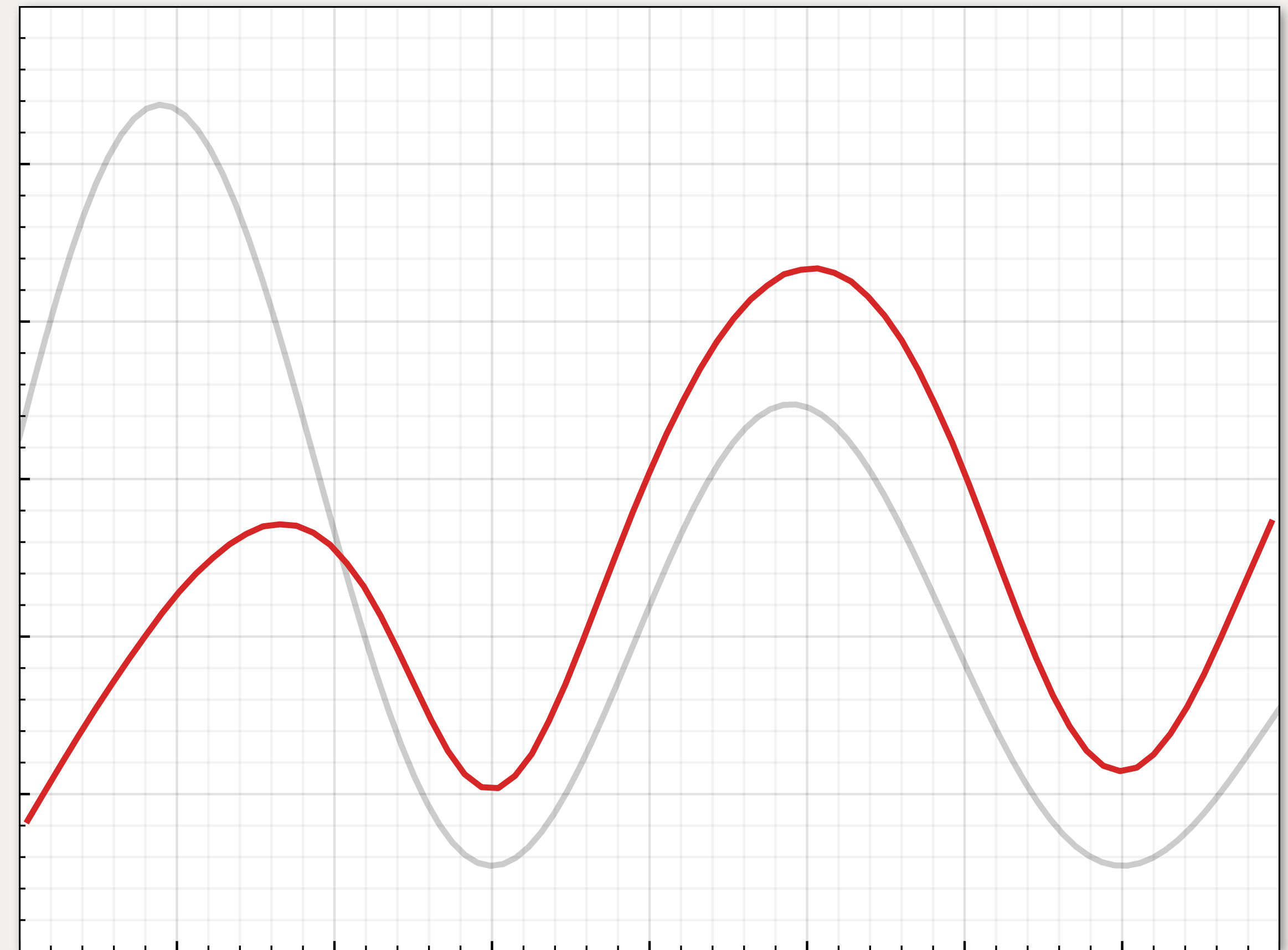
$p_2(x)$

# Improved Reservoir Resampling



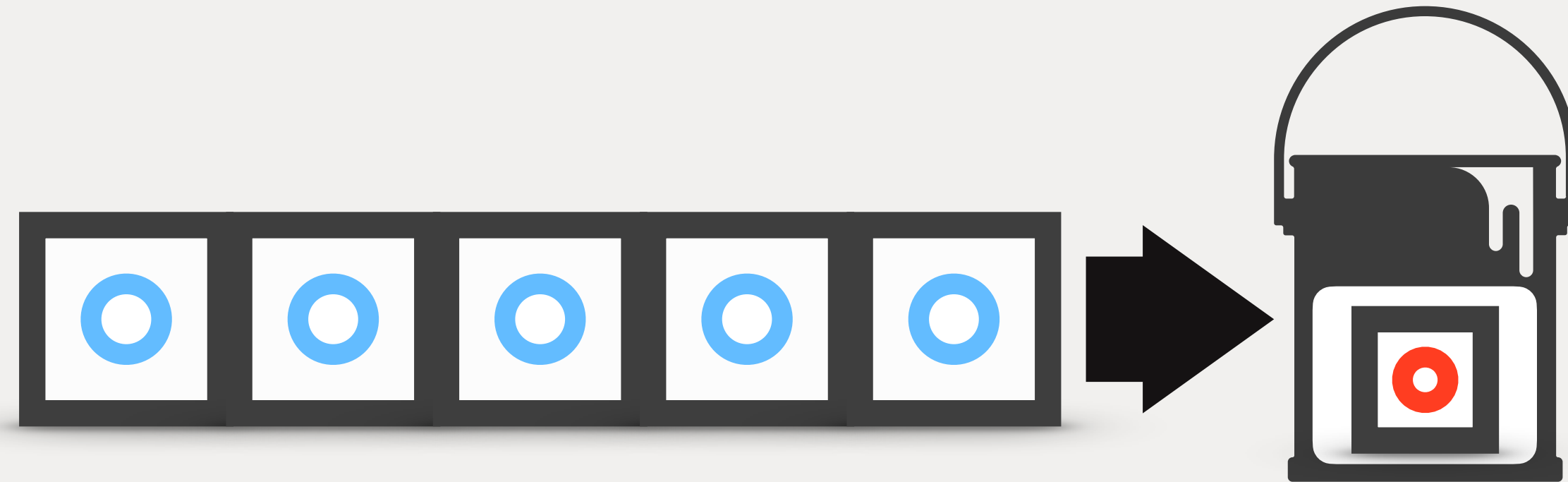
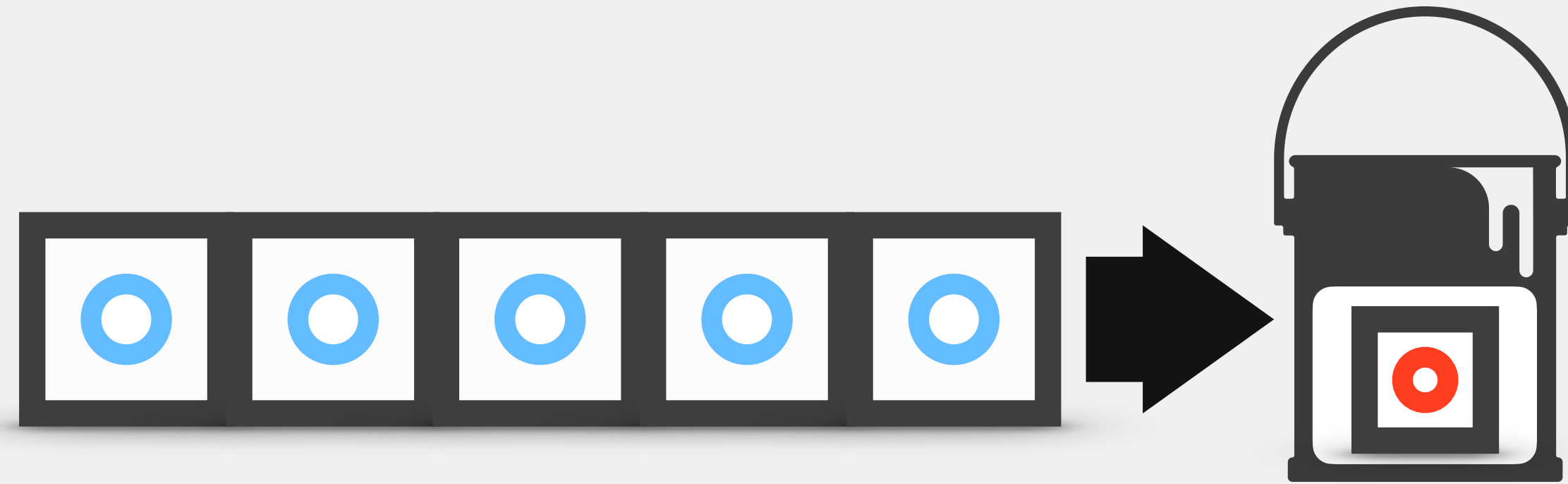
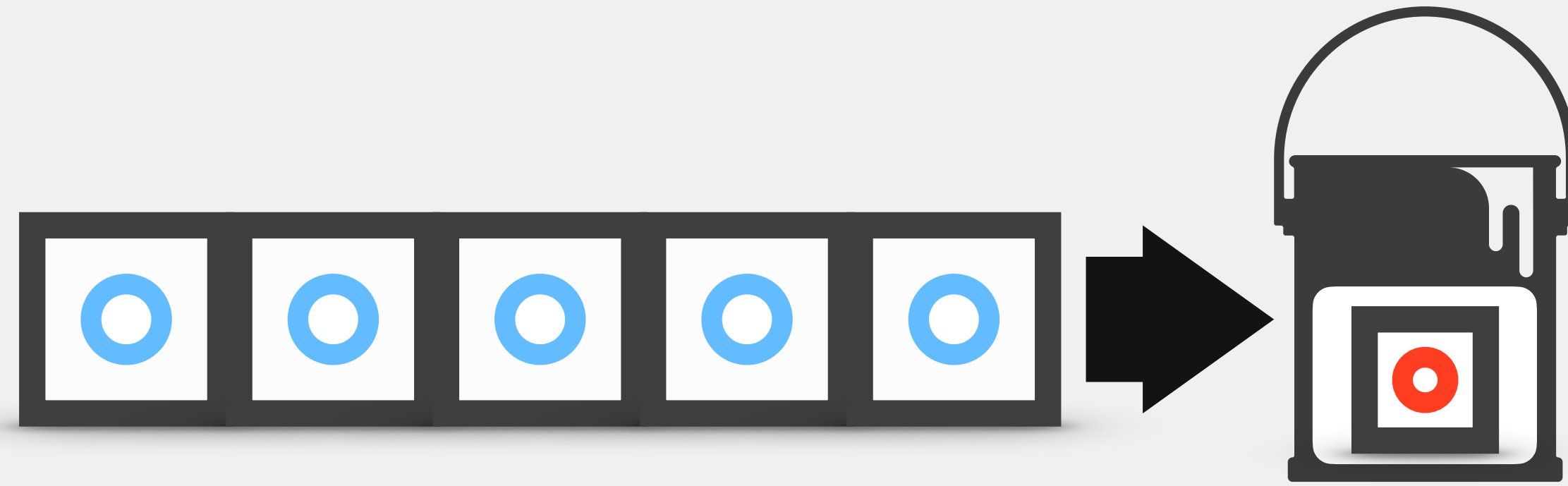
$p_1(x)$

$M = 4$



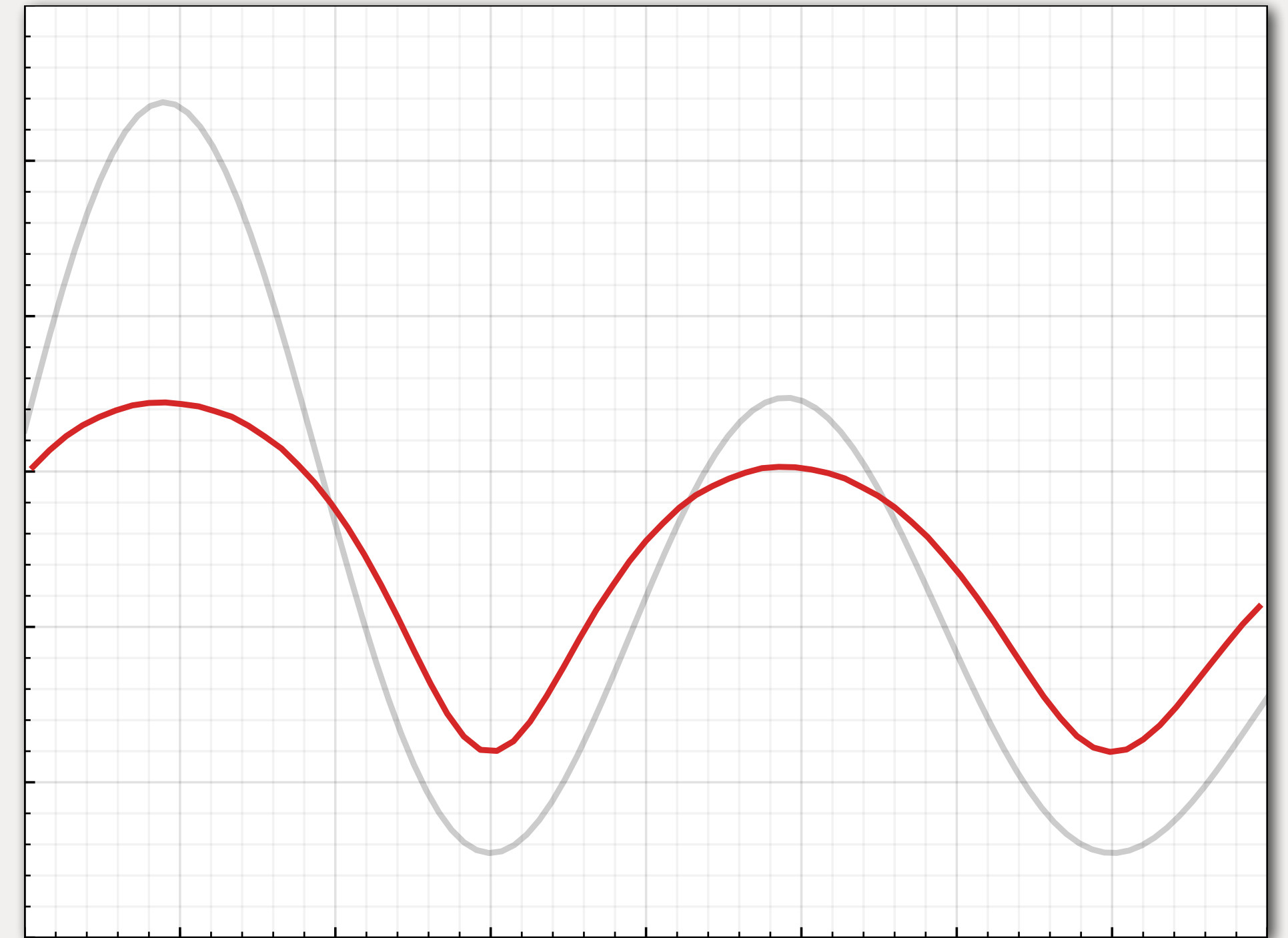
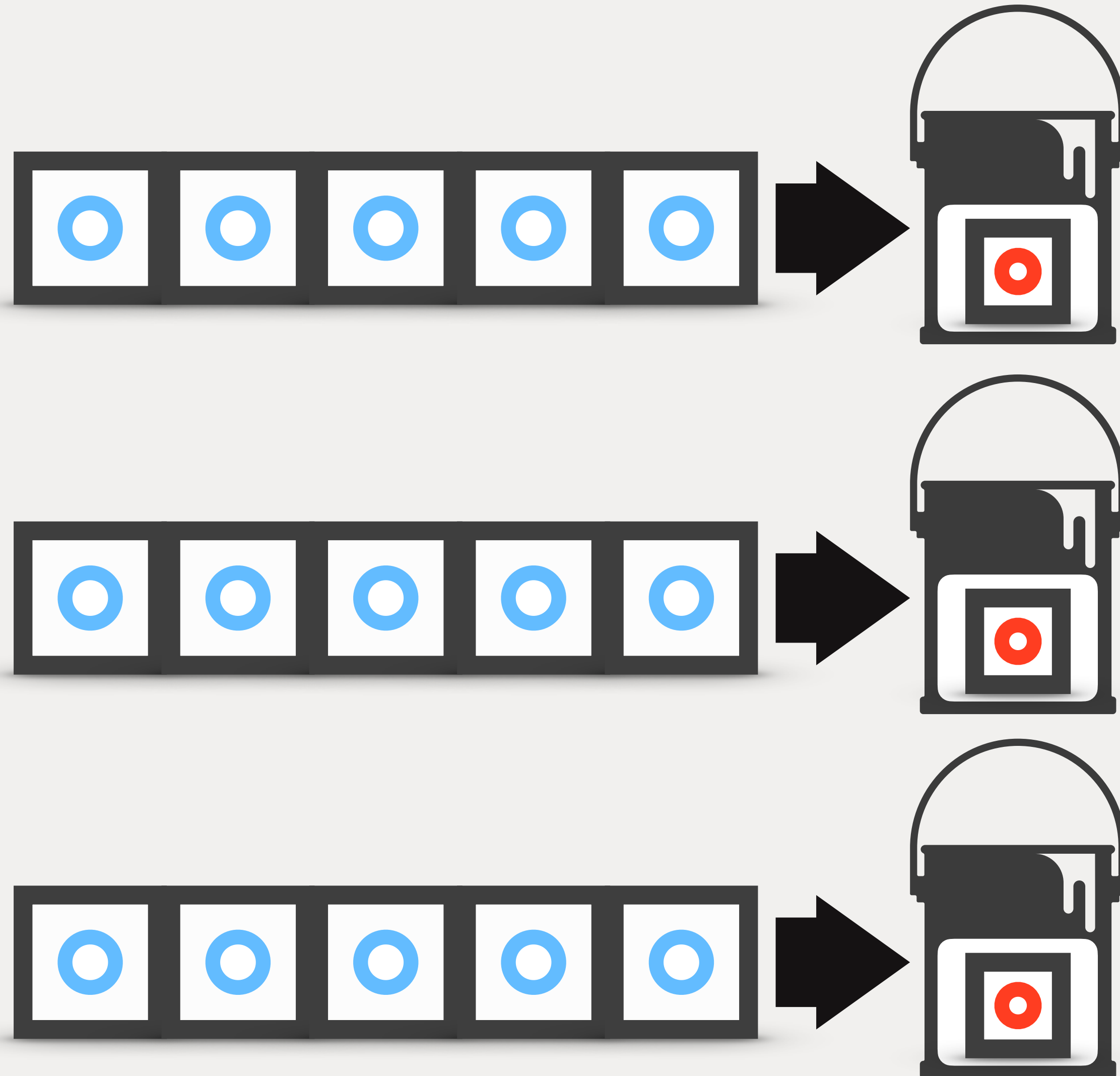
$p_2(x)$

# Reservoir Reuse

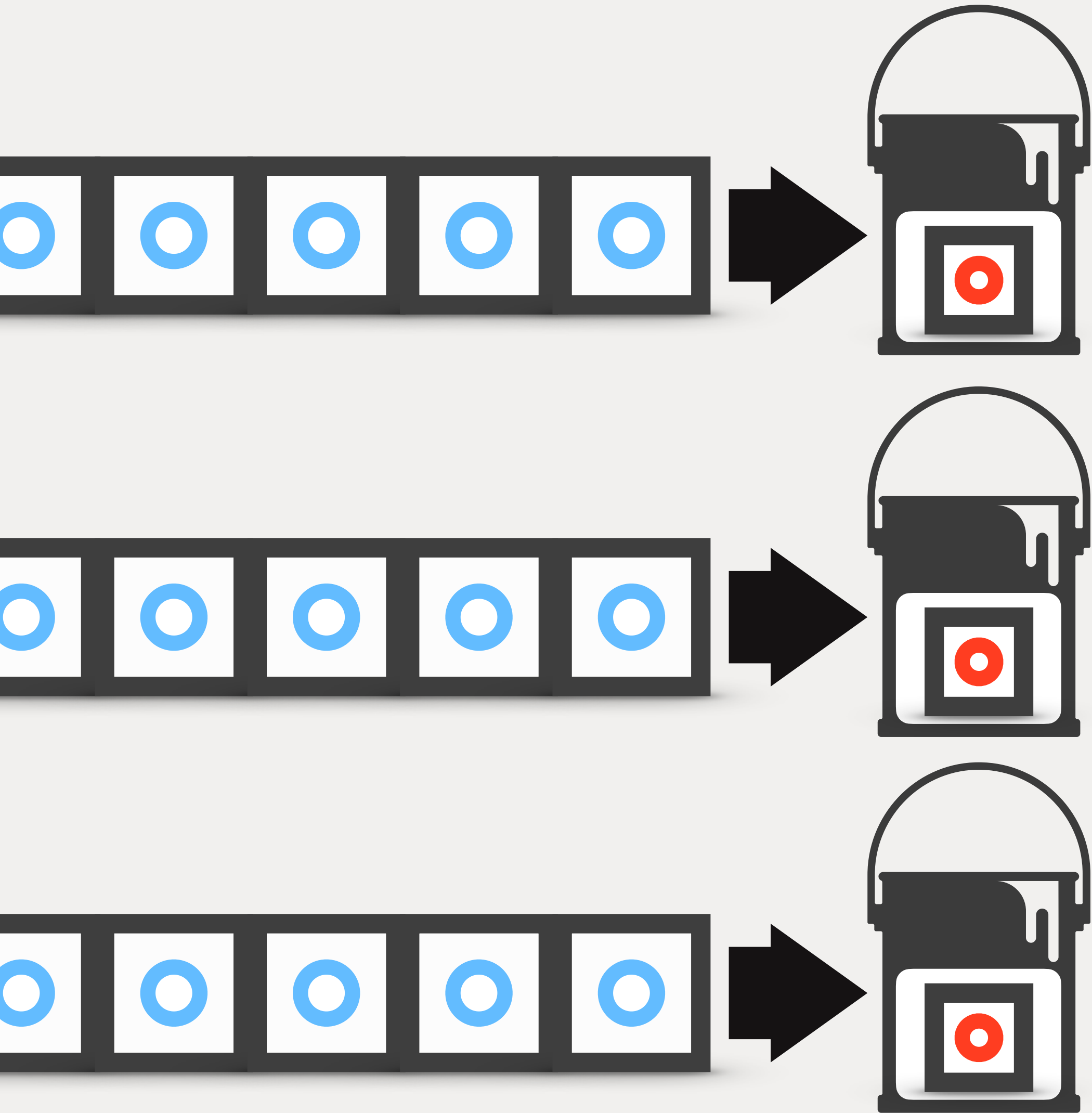




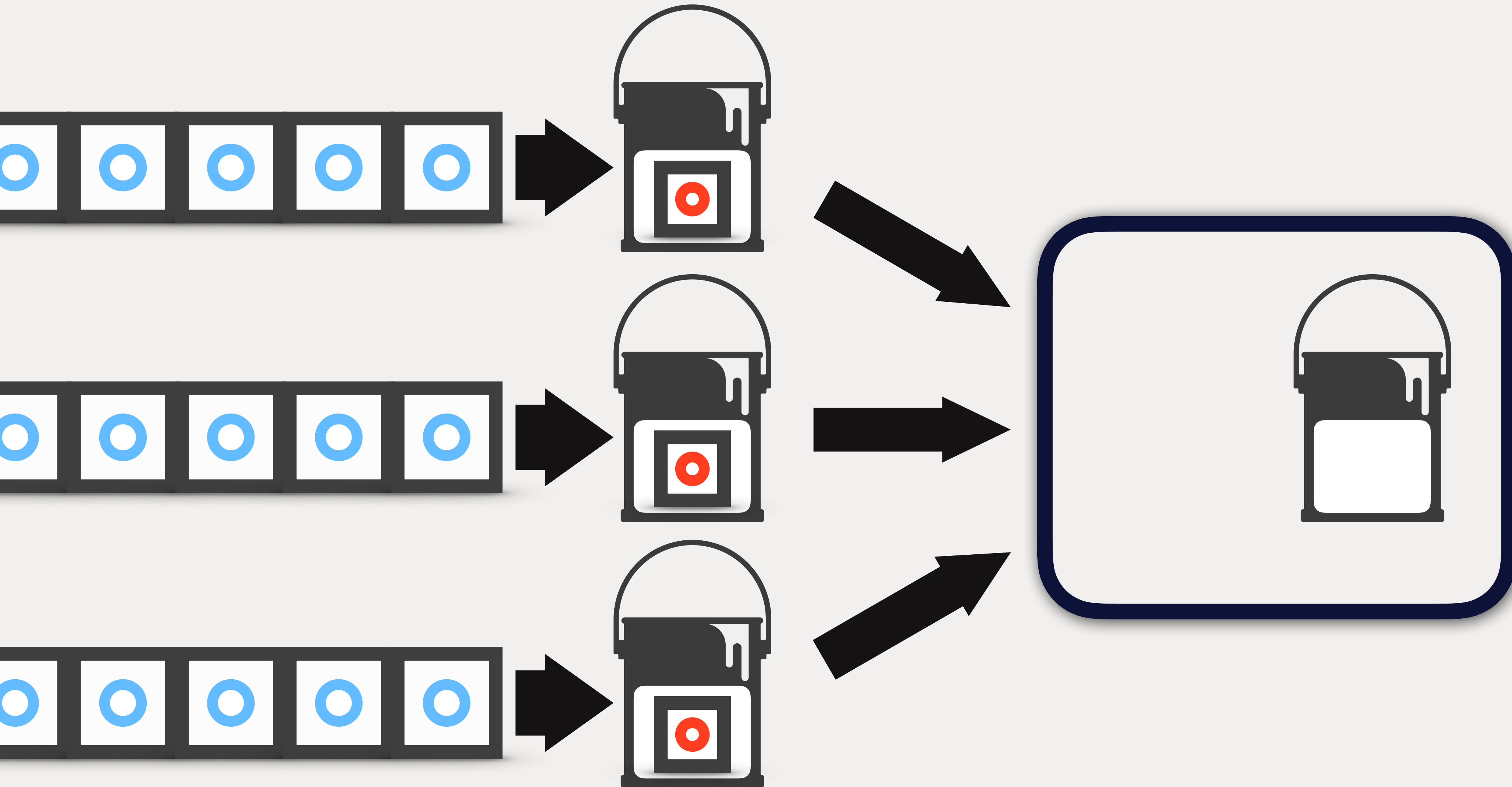
# Reservoir Reuse



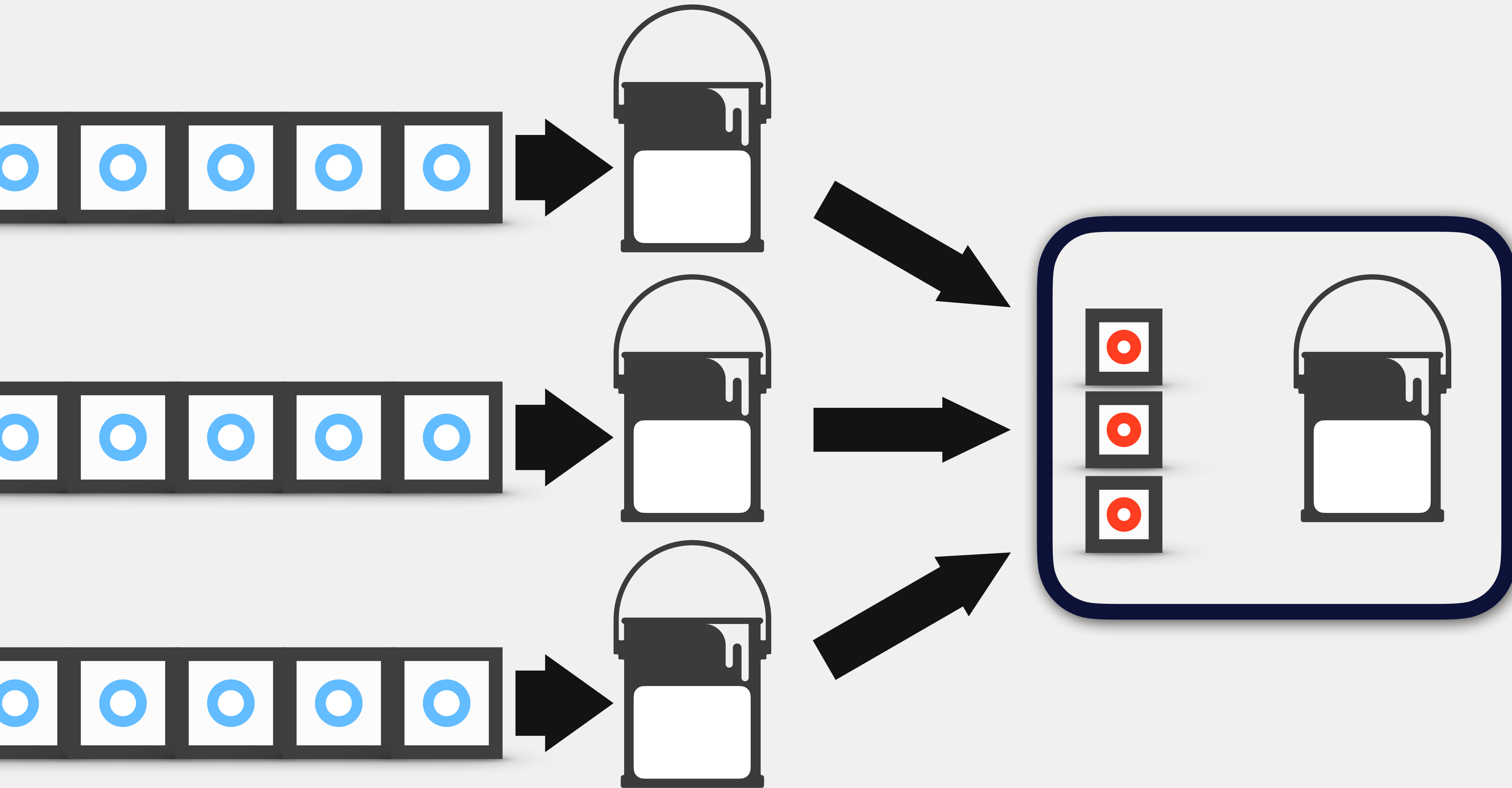
# Reservoir Reuse



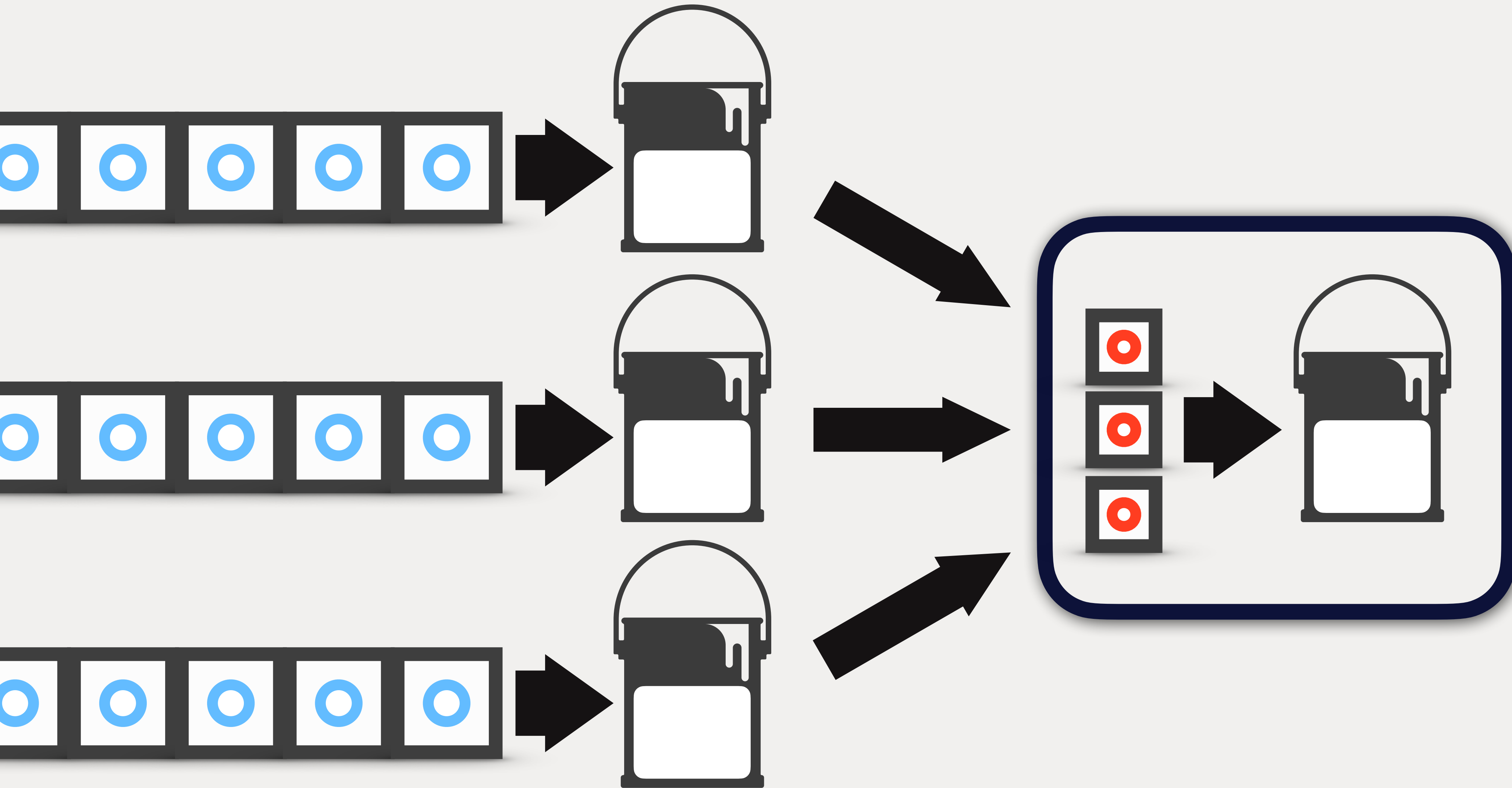
# Reservoir Reuse



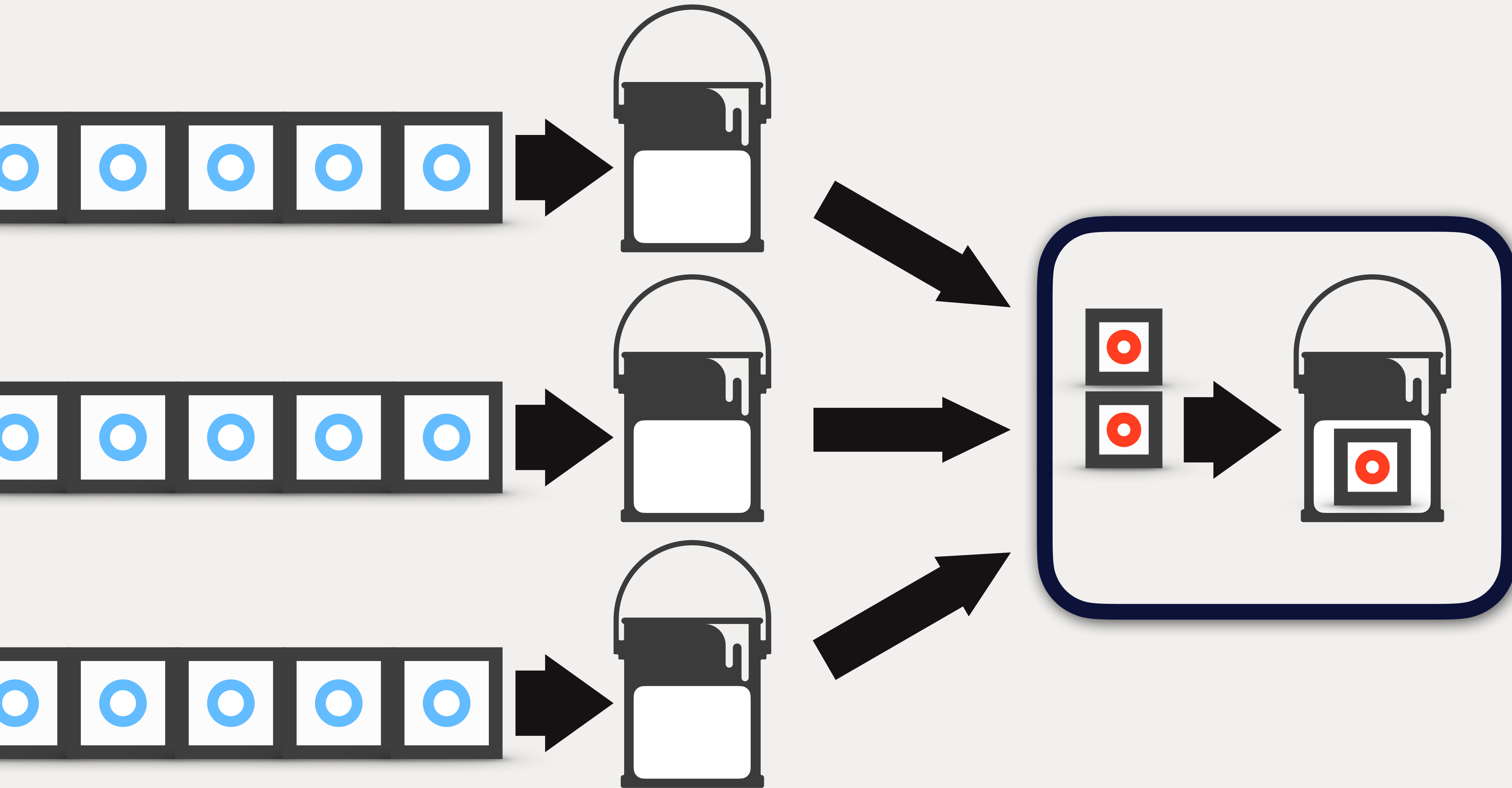
# Reservoir Reuse



# Reservoir Reuse

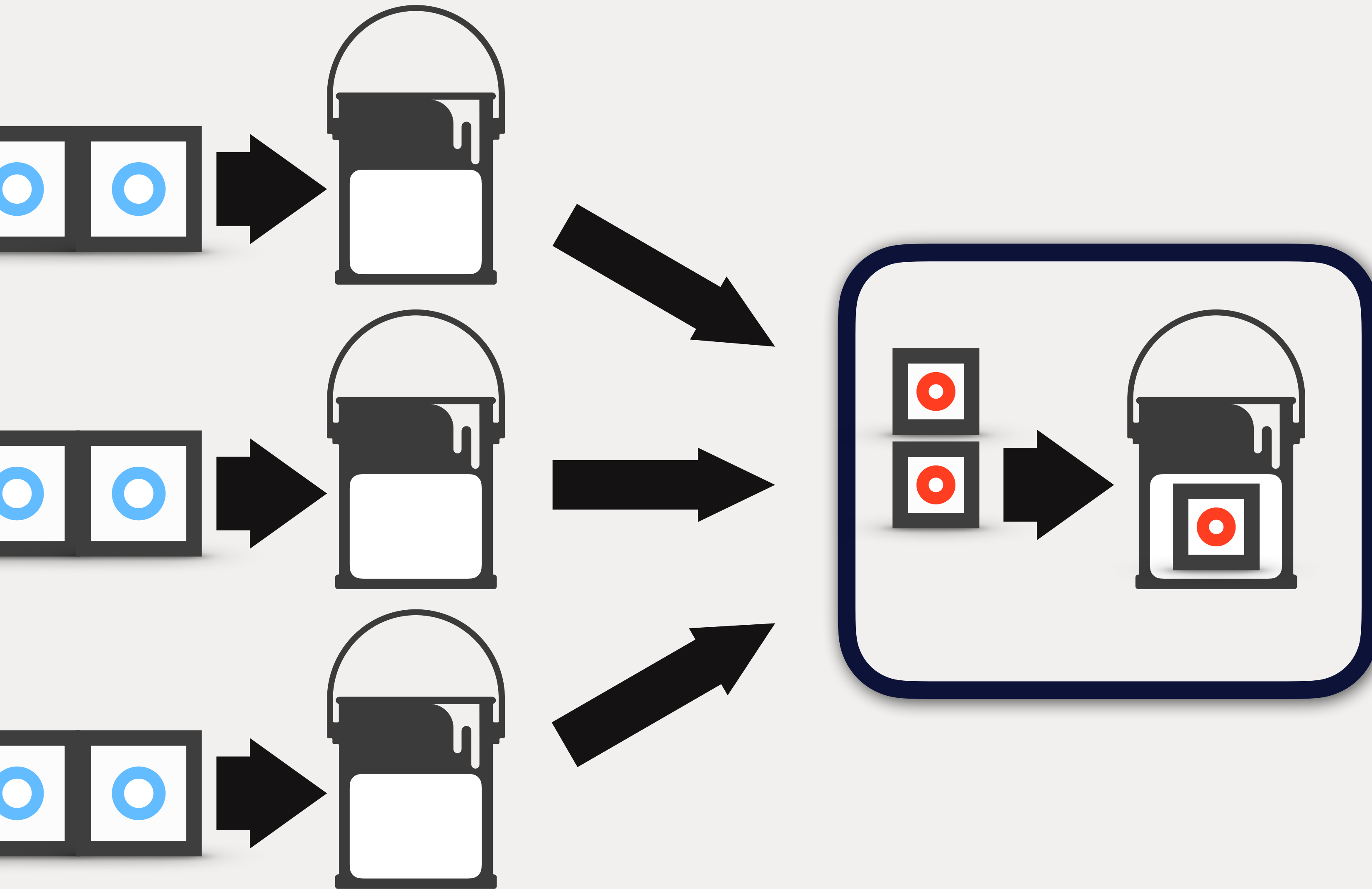


# Reservoir Reuse

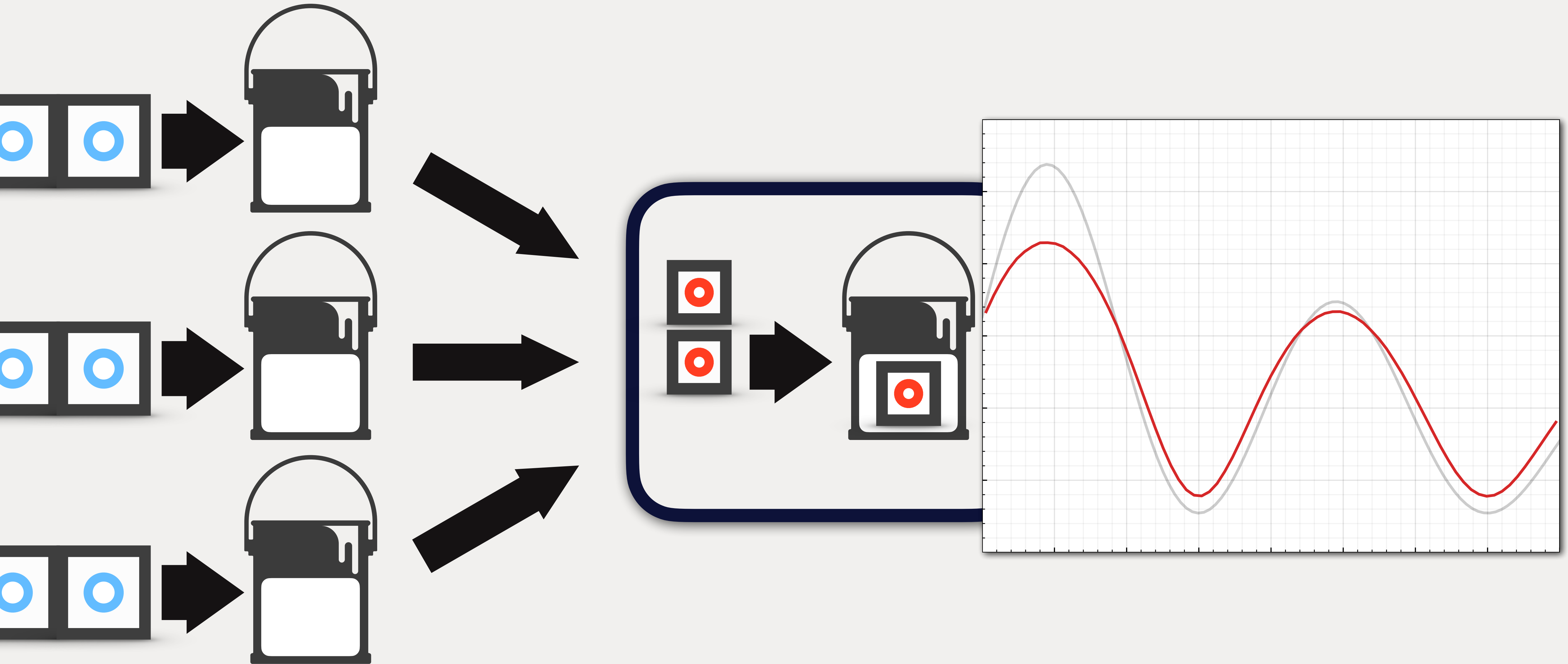




# Reservoir Reuse

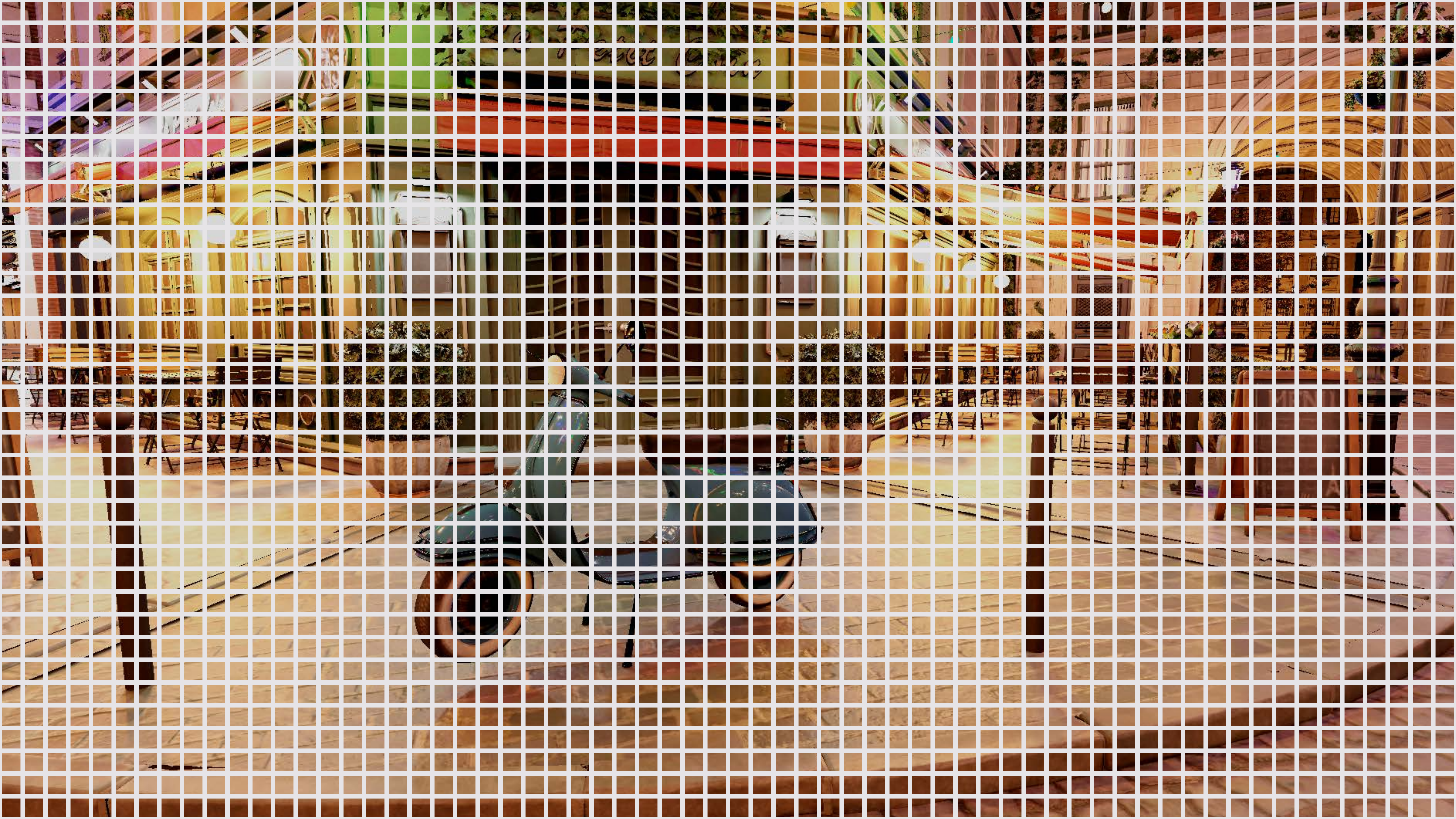


# Reservoir Reuse

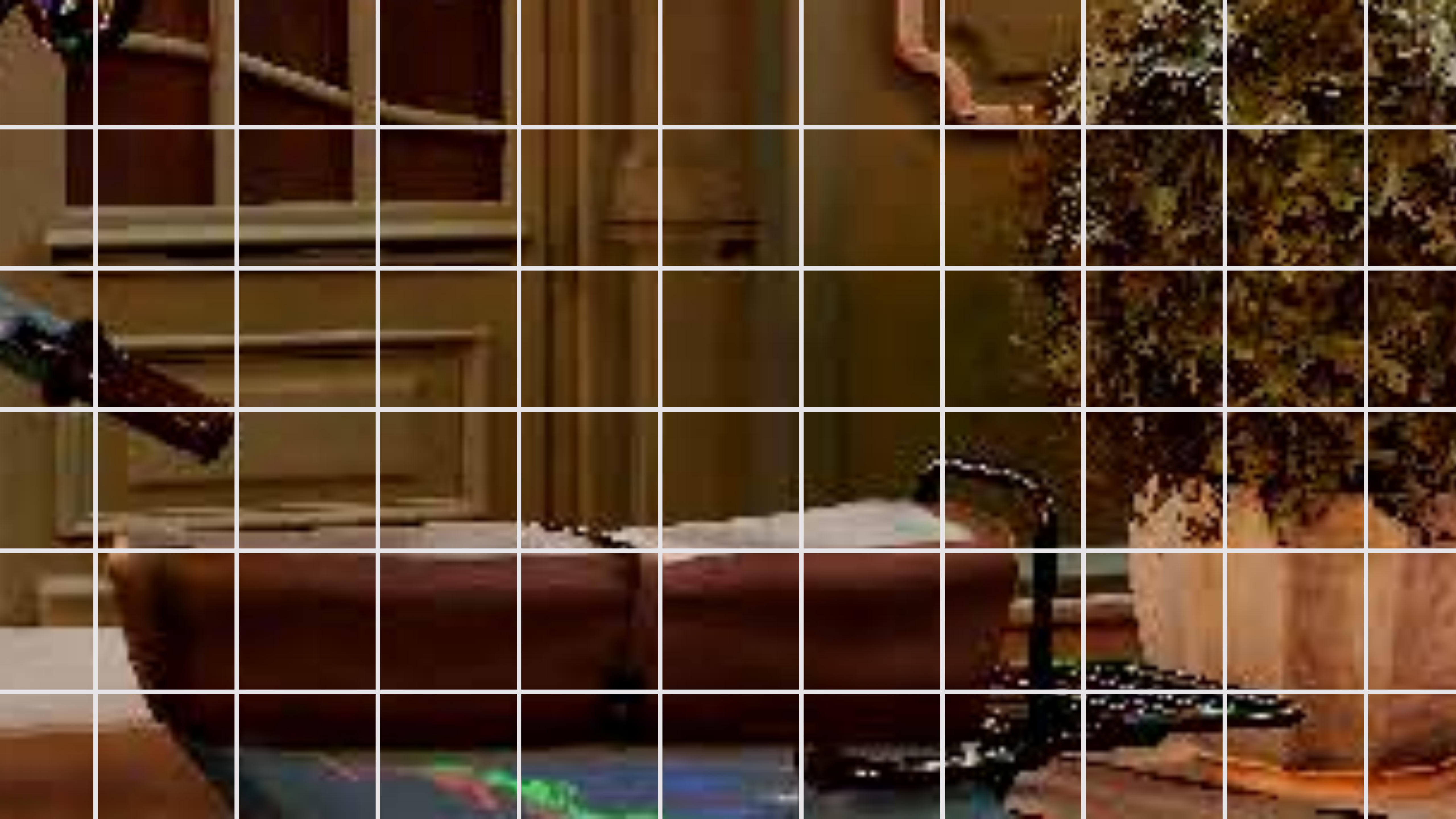


# Spatial Reuse

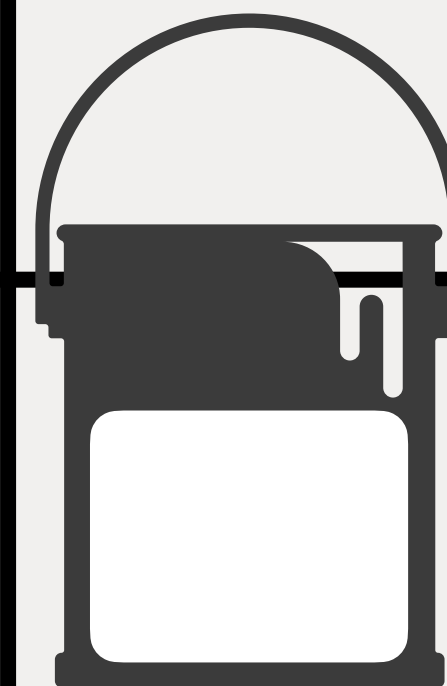
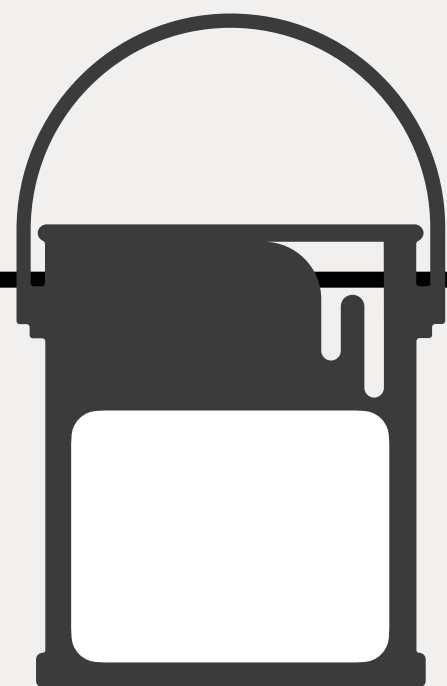
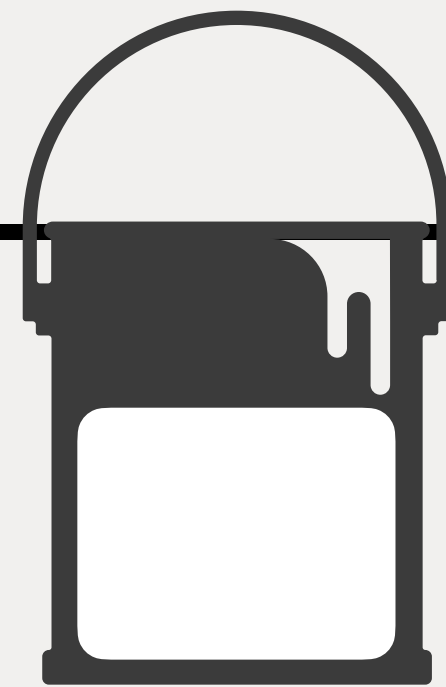
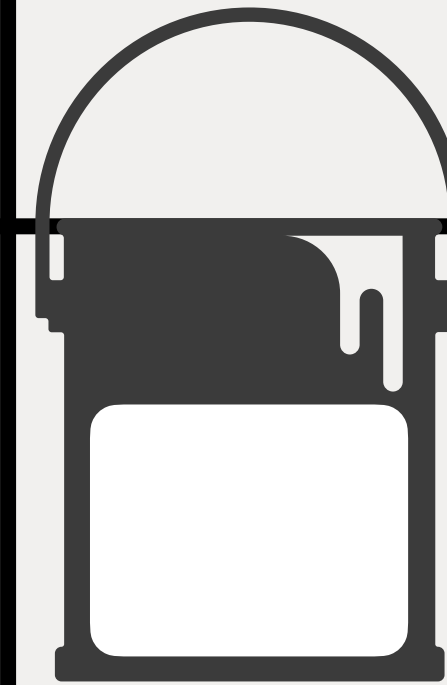
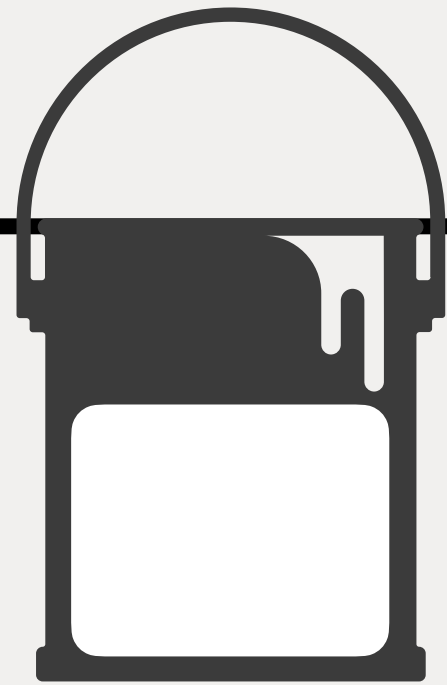




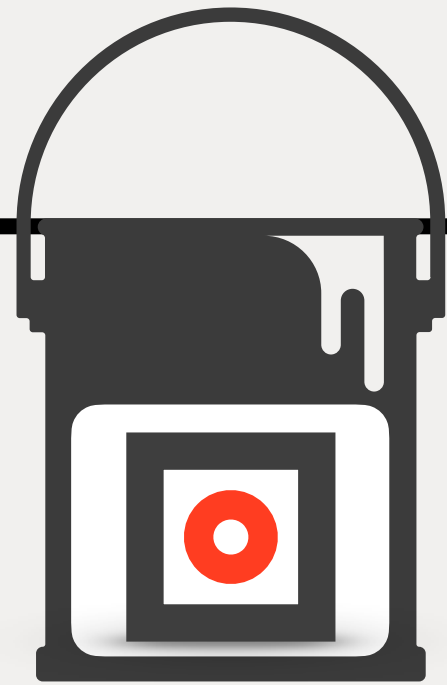


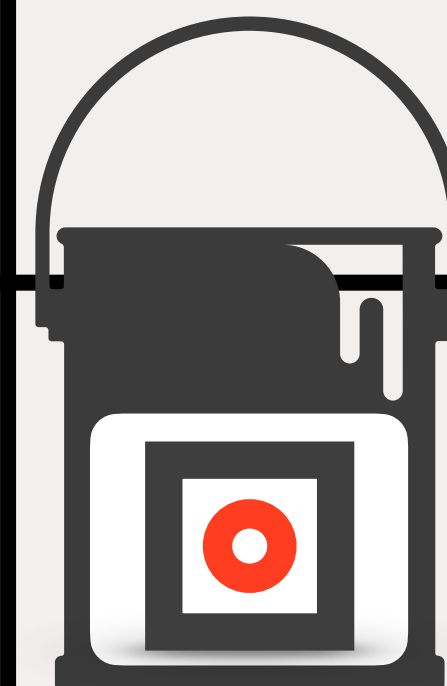
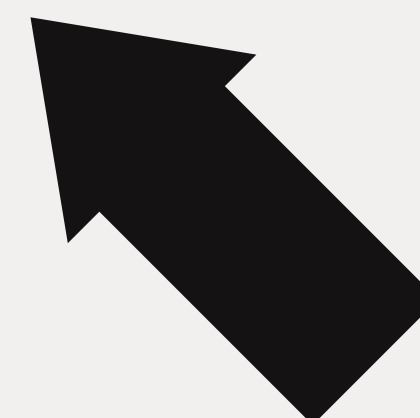
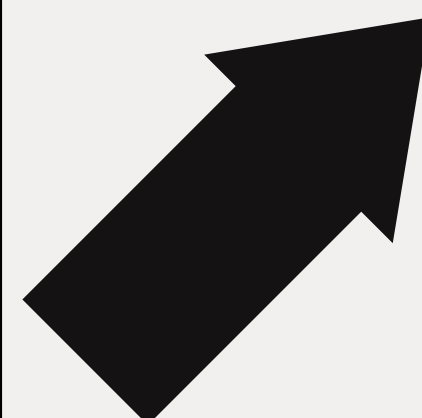
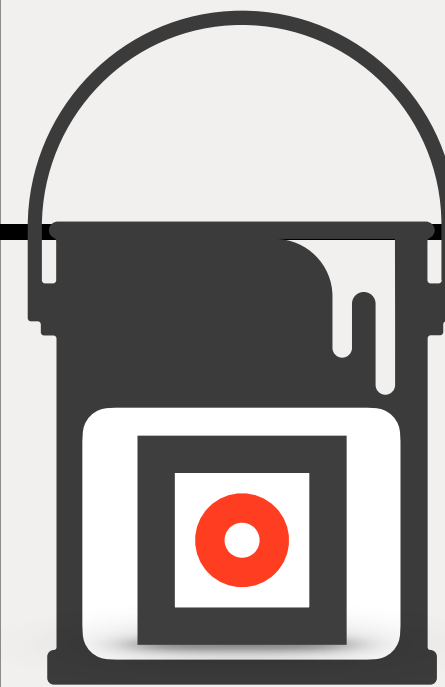
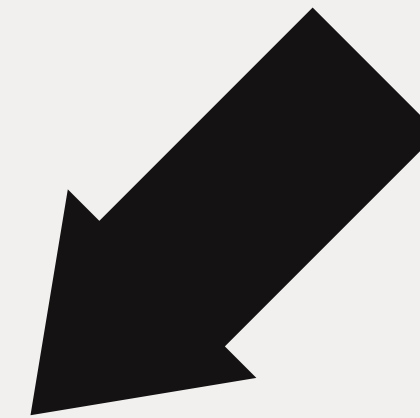
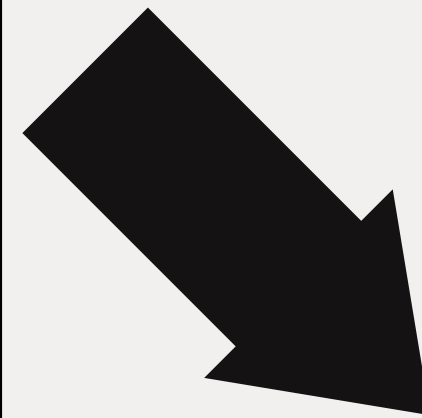


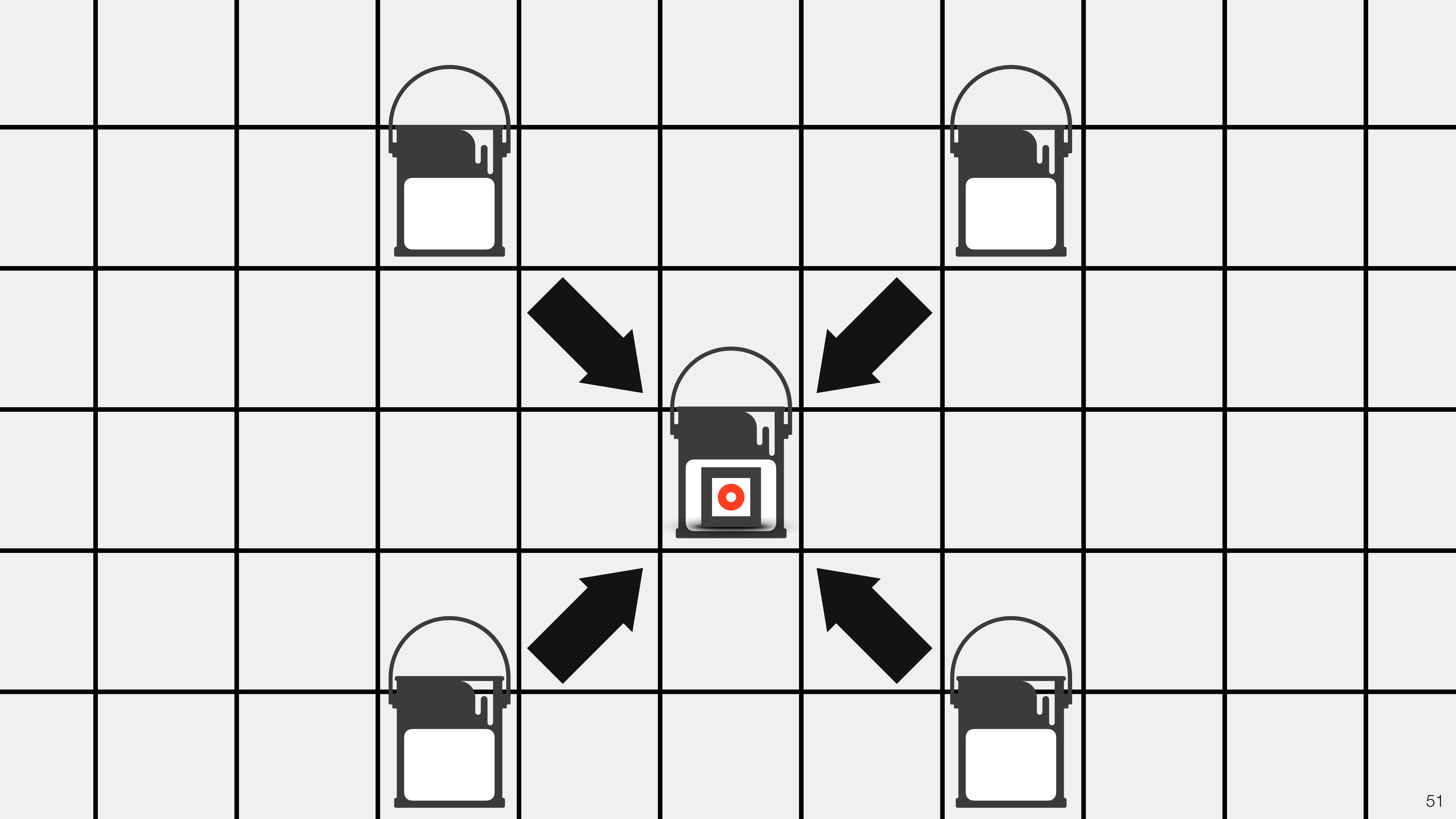


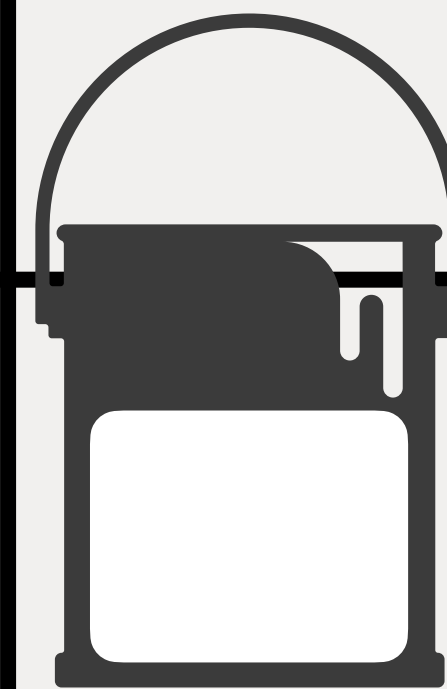
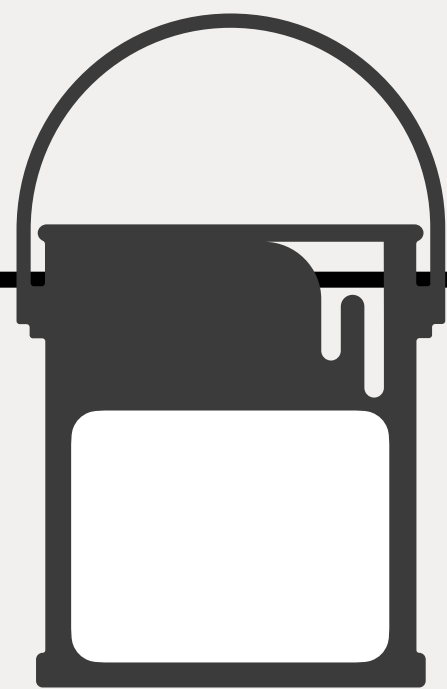
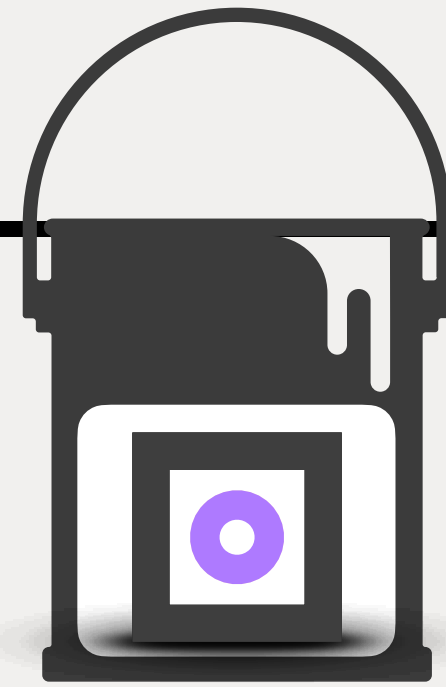
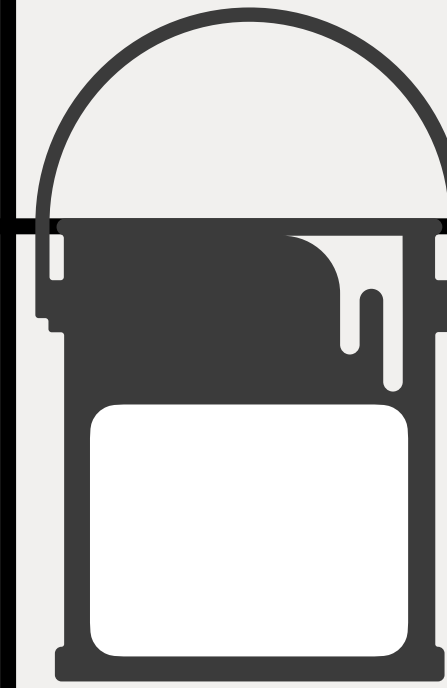
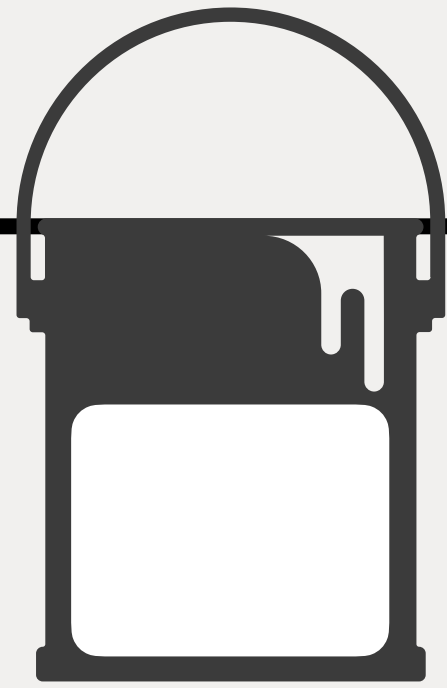




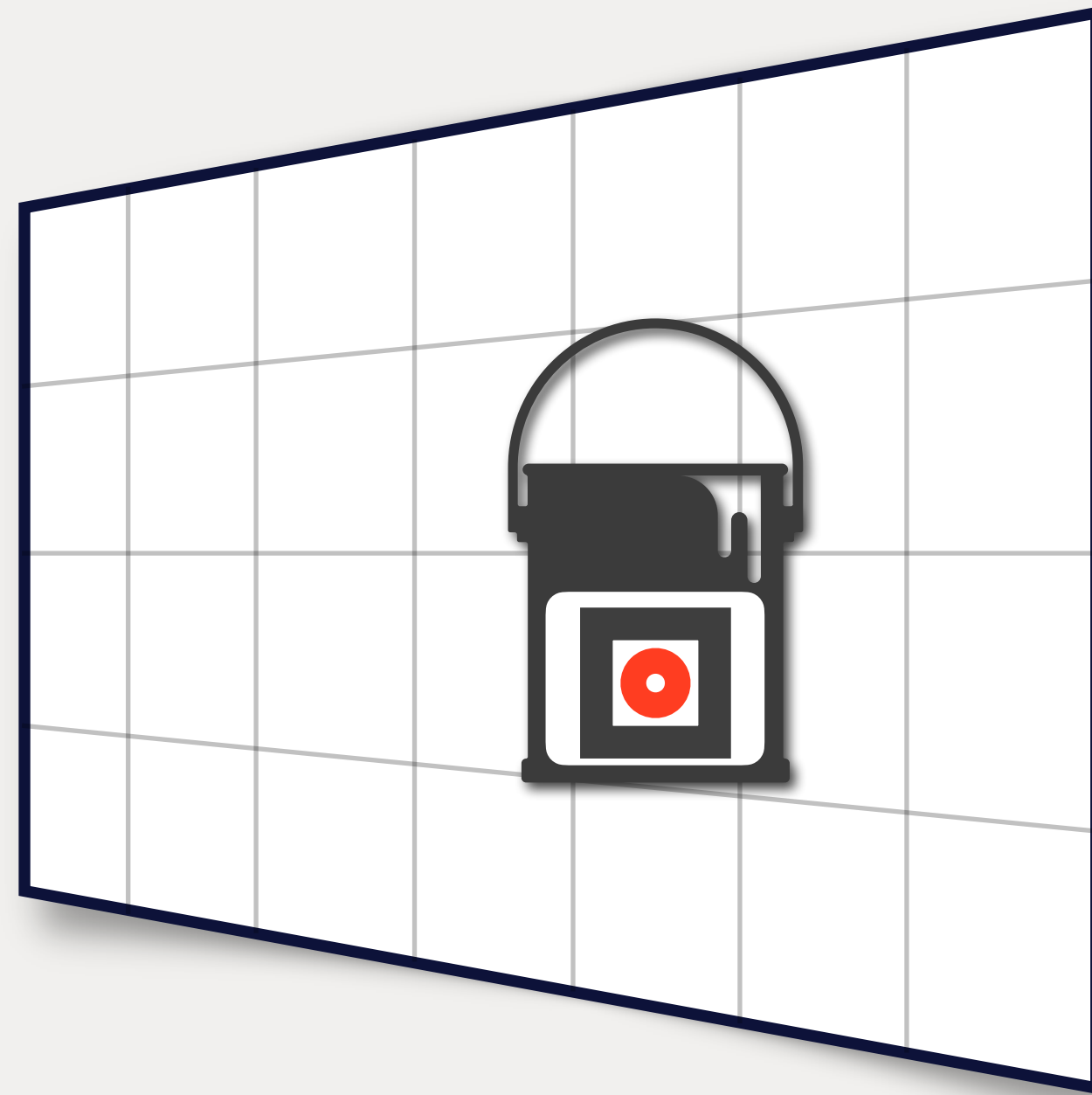






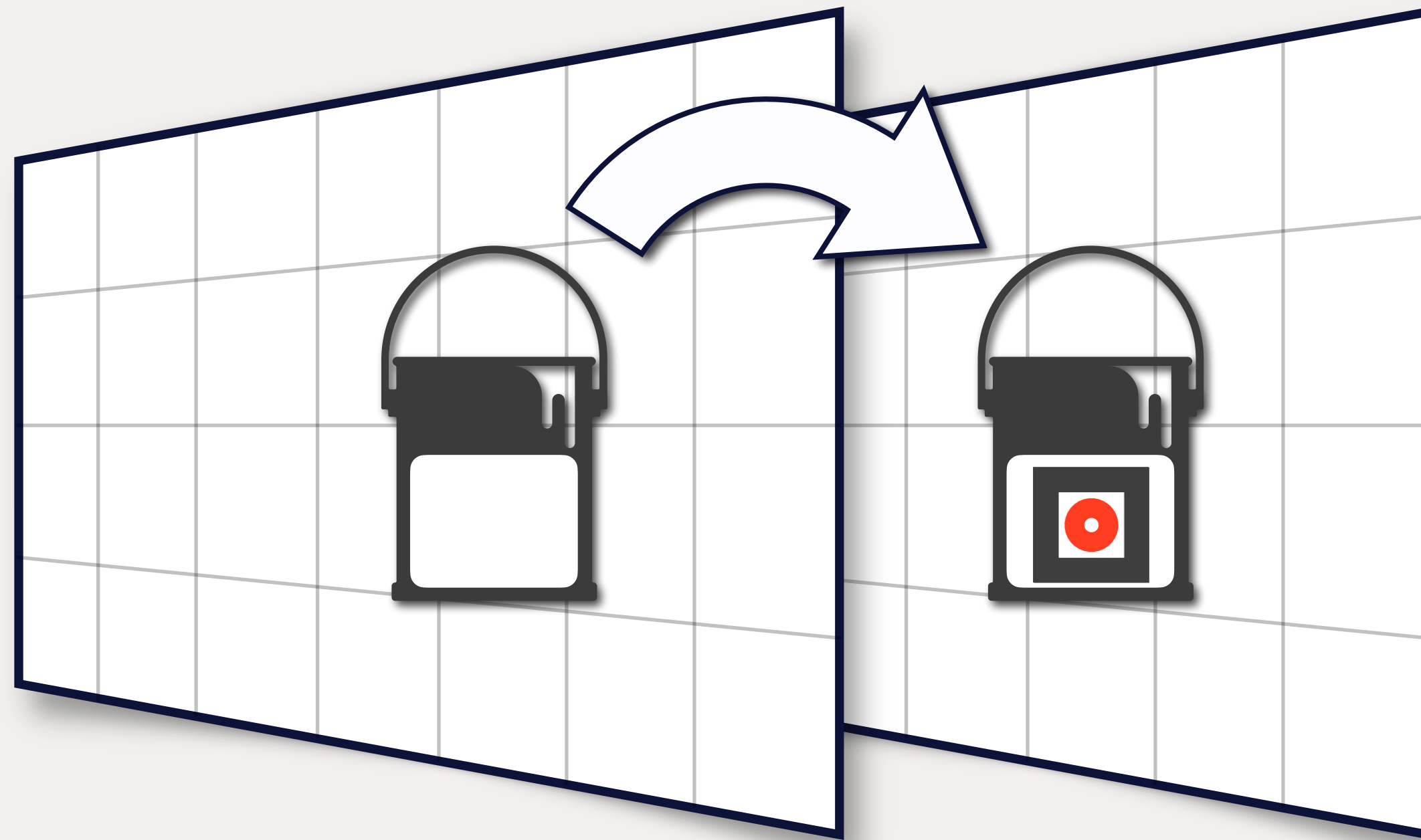


# Temporal Reuse

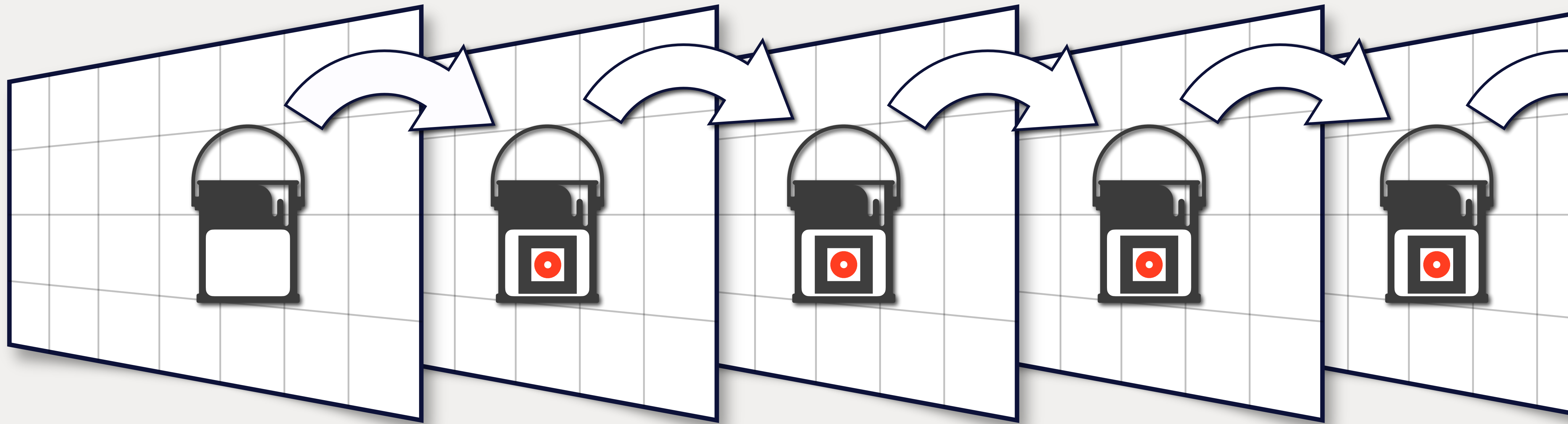




# Temporal Reuse



# Temporal Reuse



# Visibility Reuse

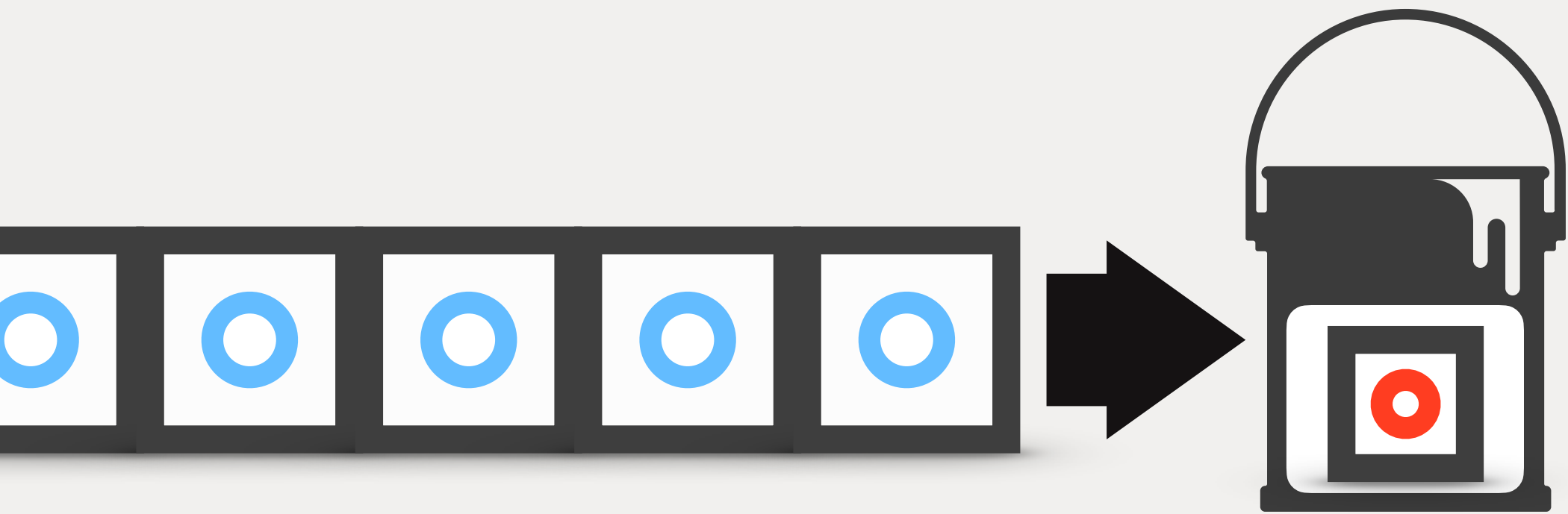
# Visibility Reuse

$$w(x_i) = \frac{\hat{p}(x_i)}{p(x_i)}$$

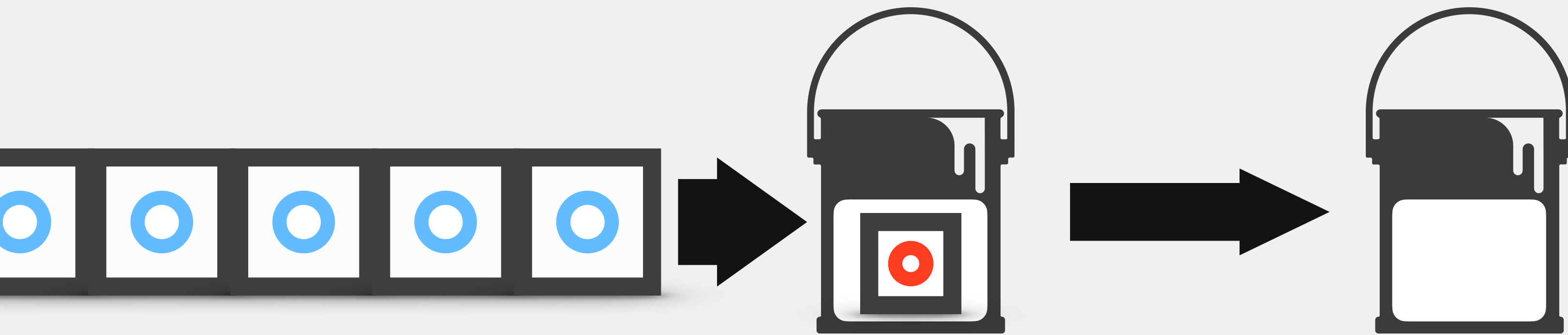
# Visibility Reuse



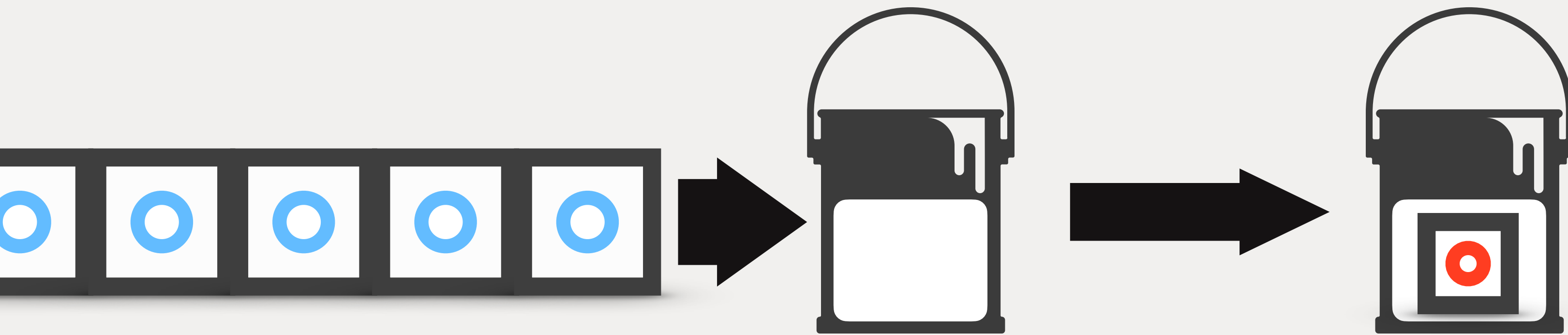
# Visibility Reuse



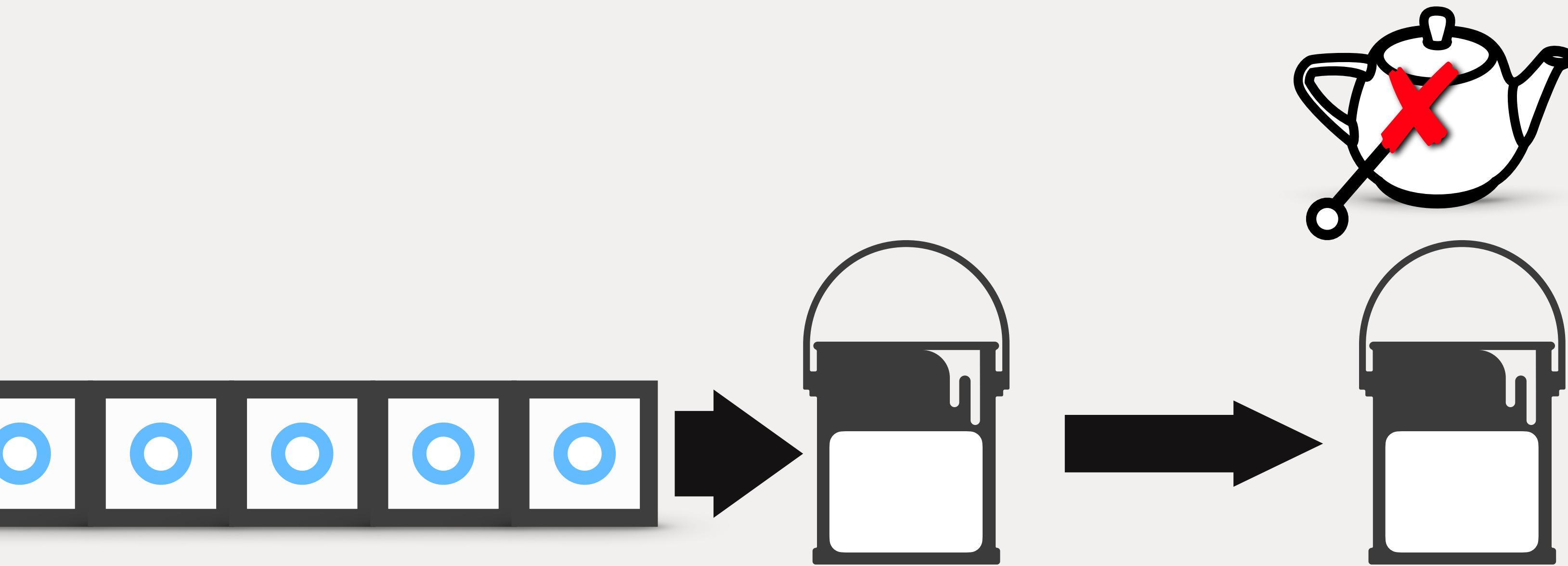
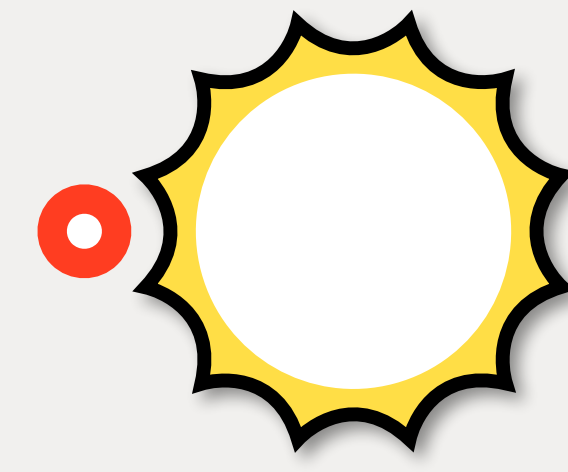
# Visibility Reuse



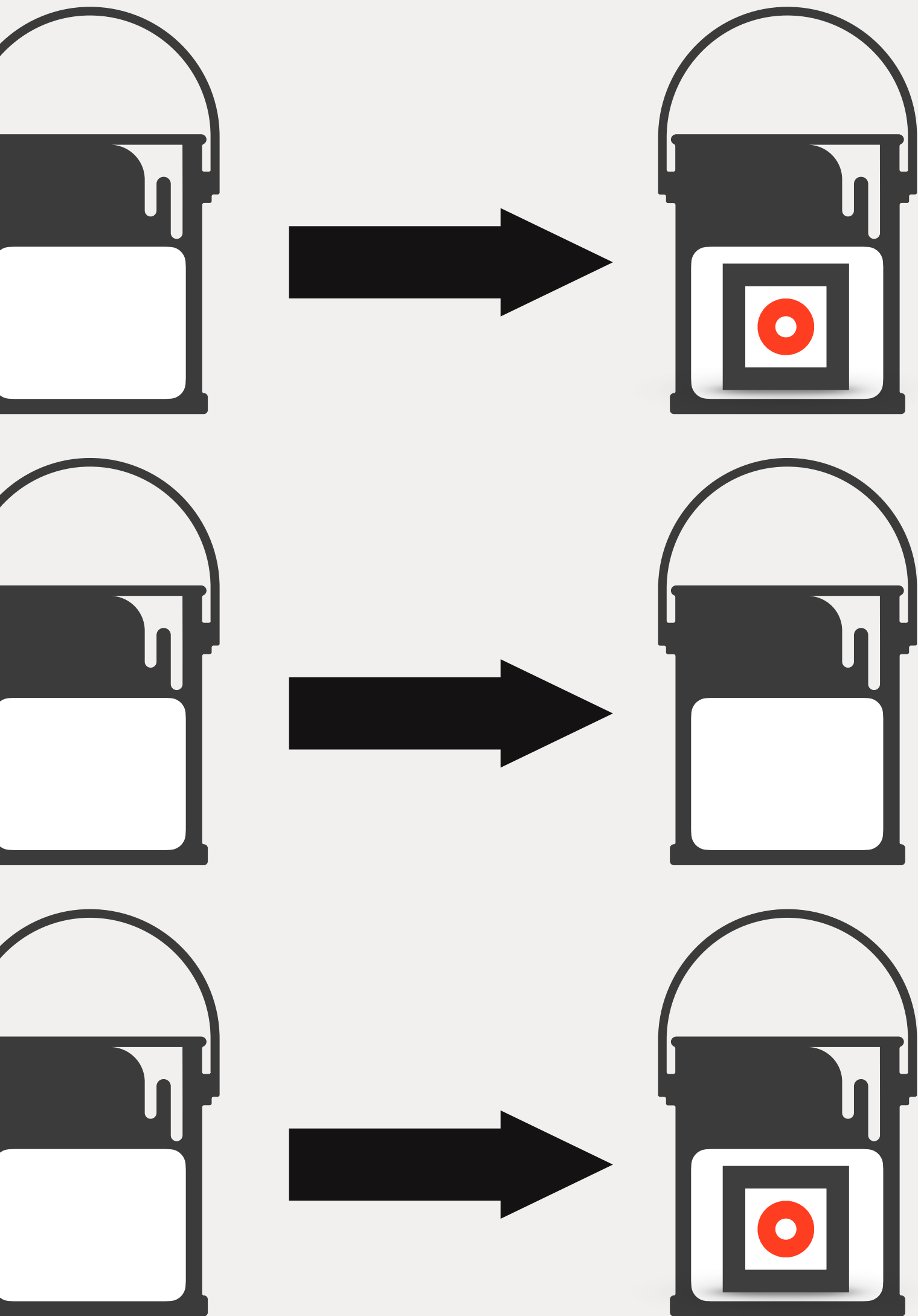
# Visibility Reuse



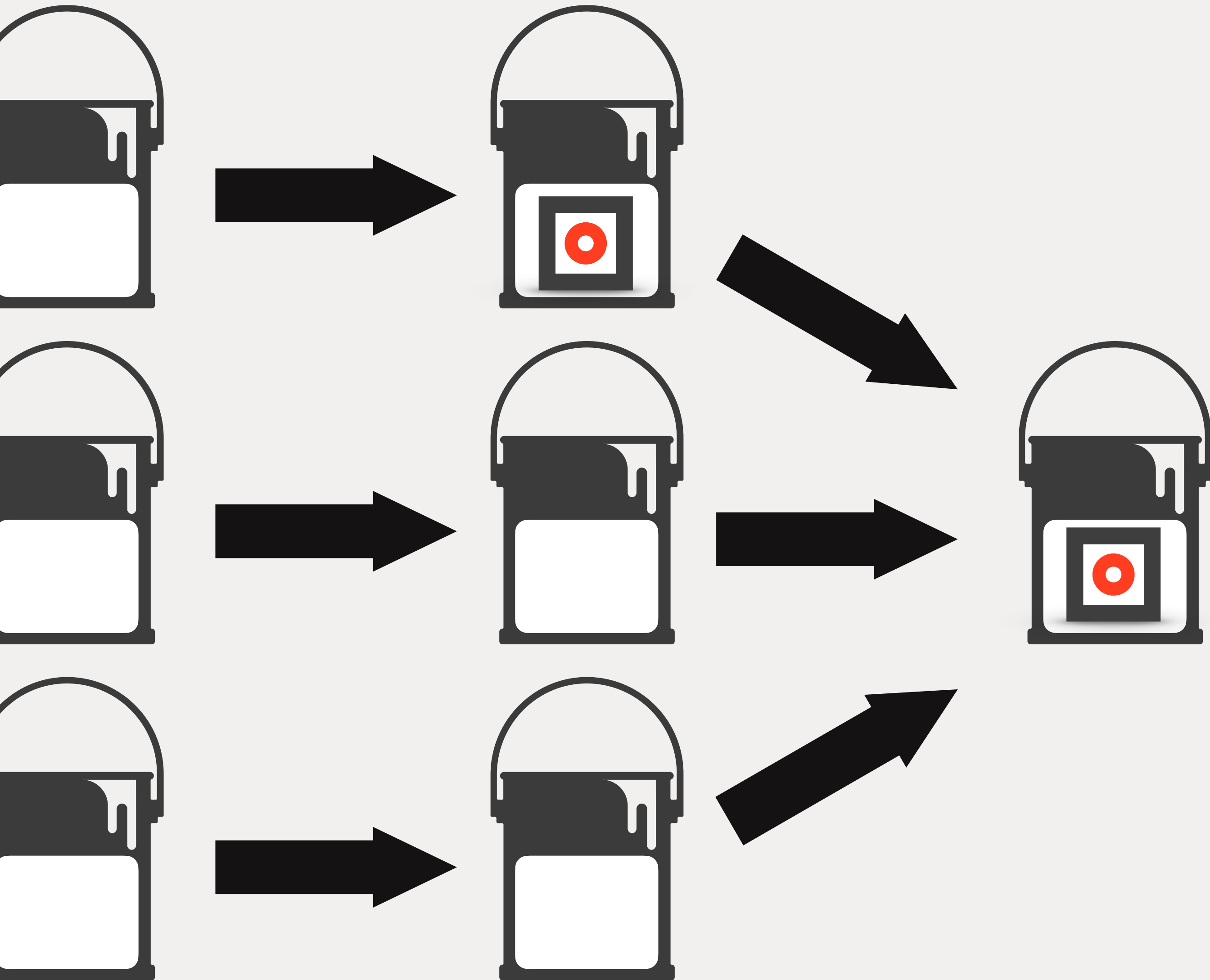
# Visibility Reuse



# Visibility Reuse

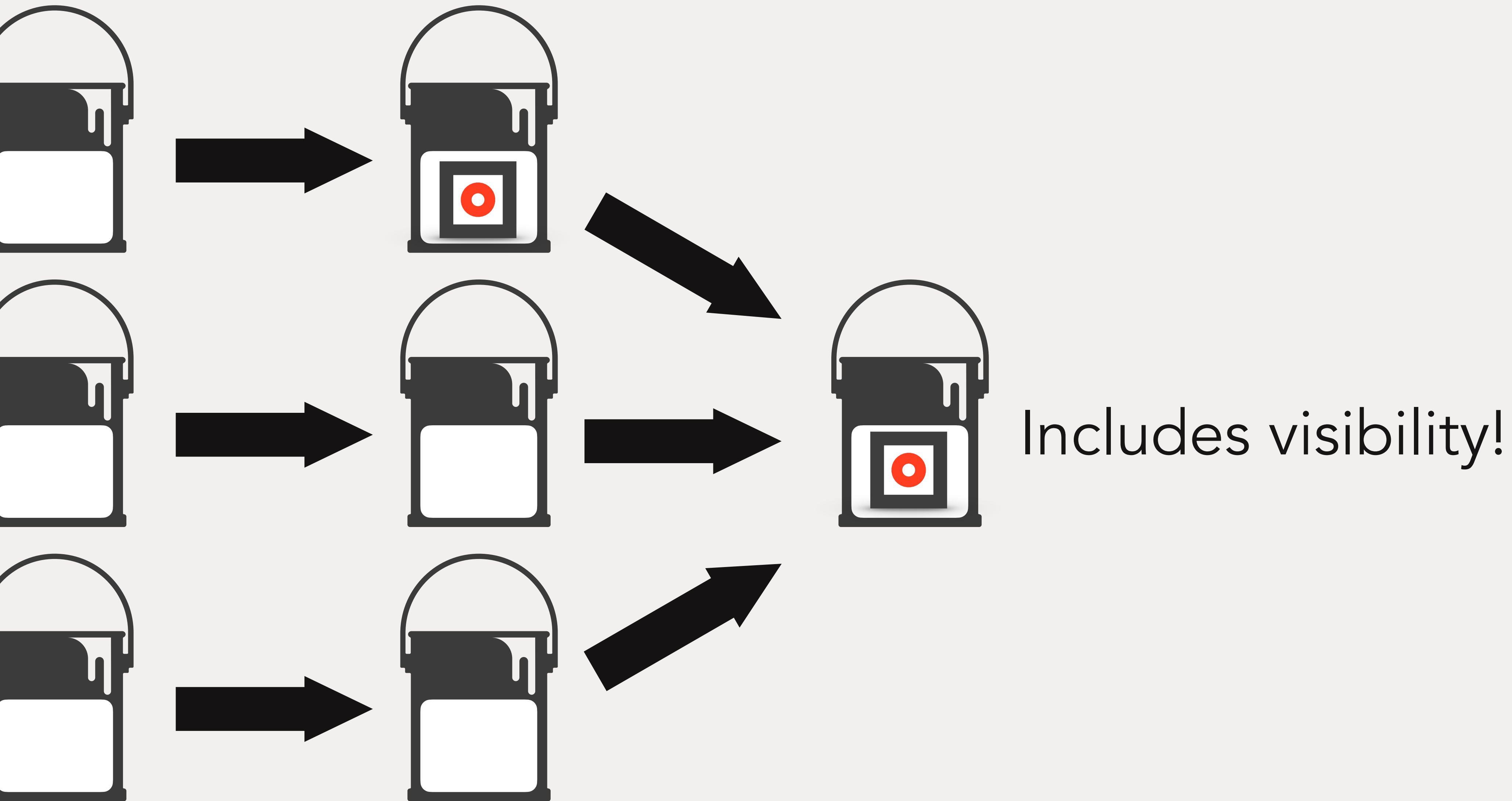


# Visibility Reuse



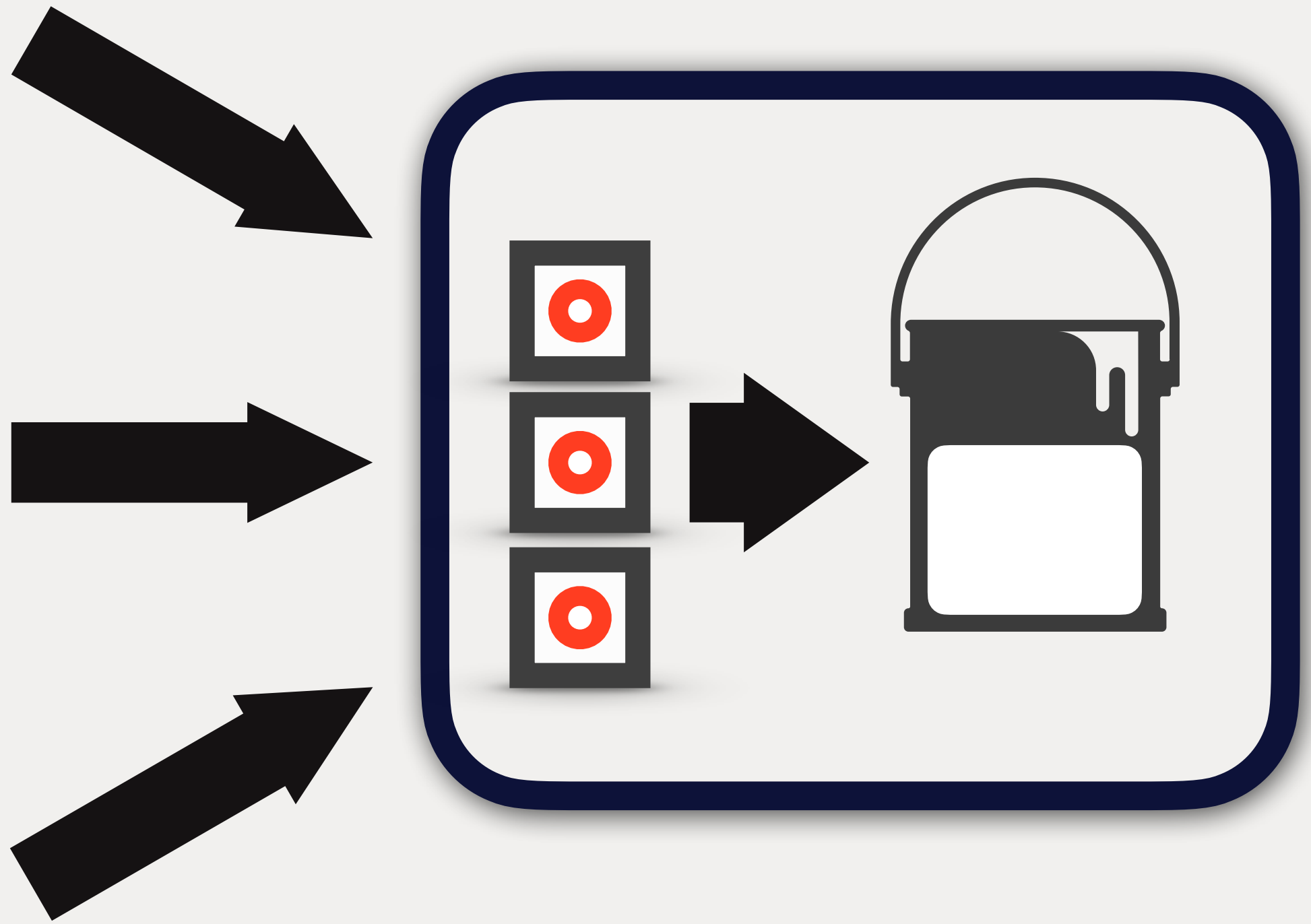


# Visibility Reuse

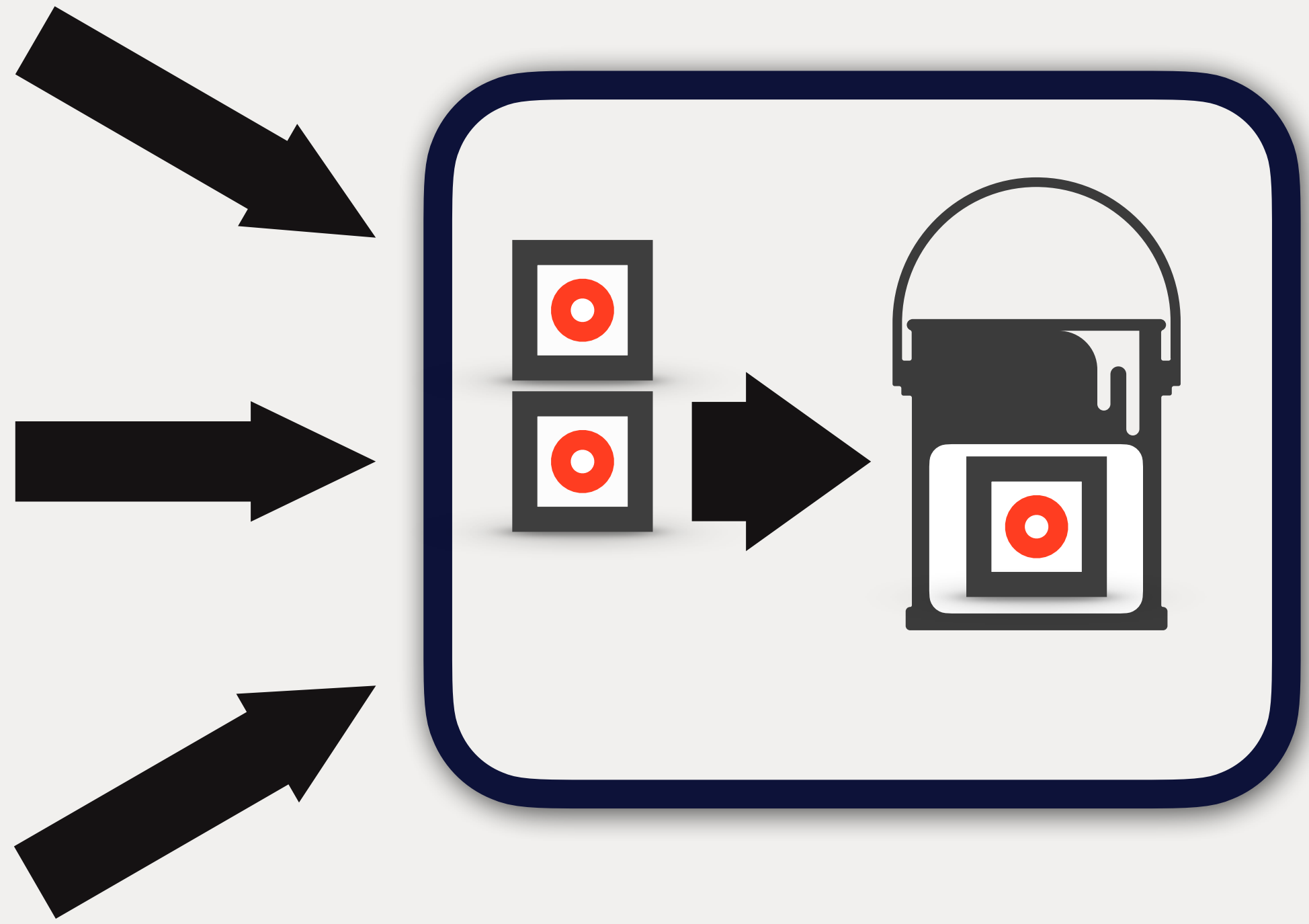


# Unbiased Reuse

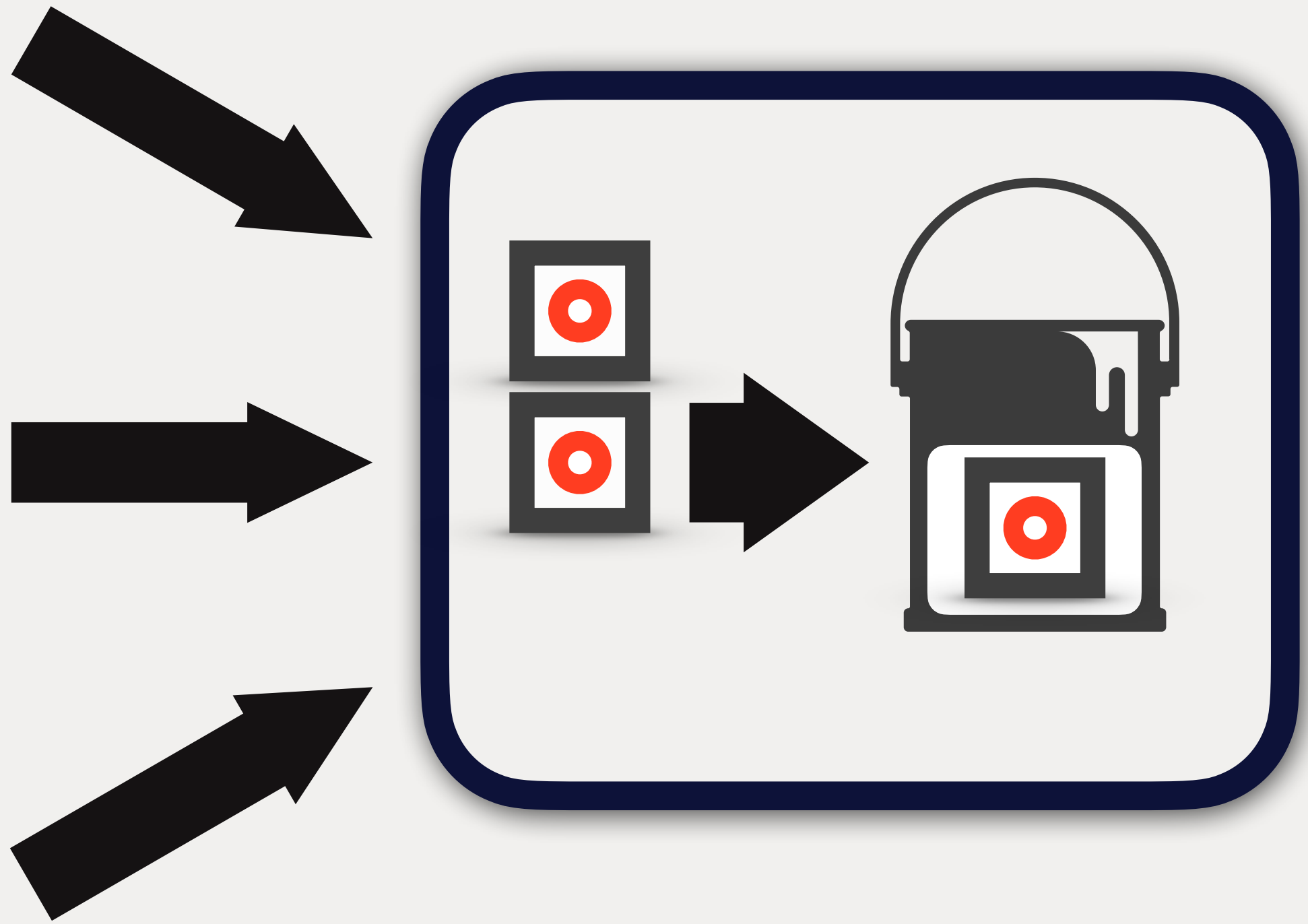
# Unbiased Reuse



# Unbiased Reuse

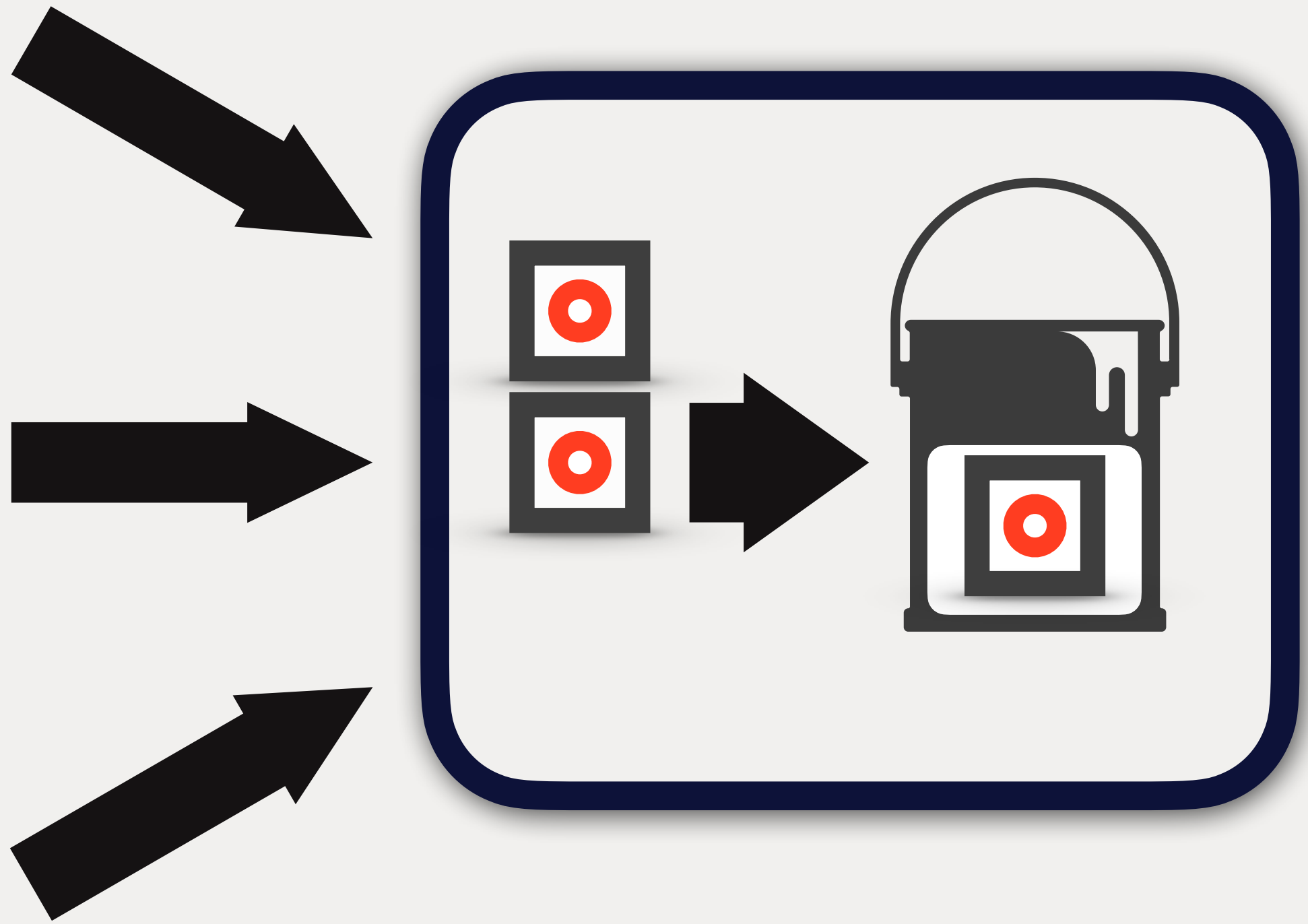


# Unbiased Reuse



$$w(x_i) = \frac{\hat{p}(x_i)}{p(x_i)}$$

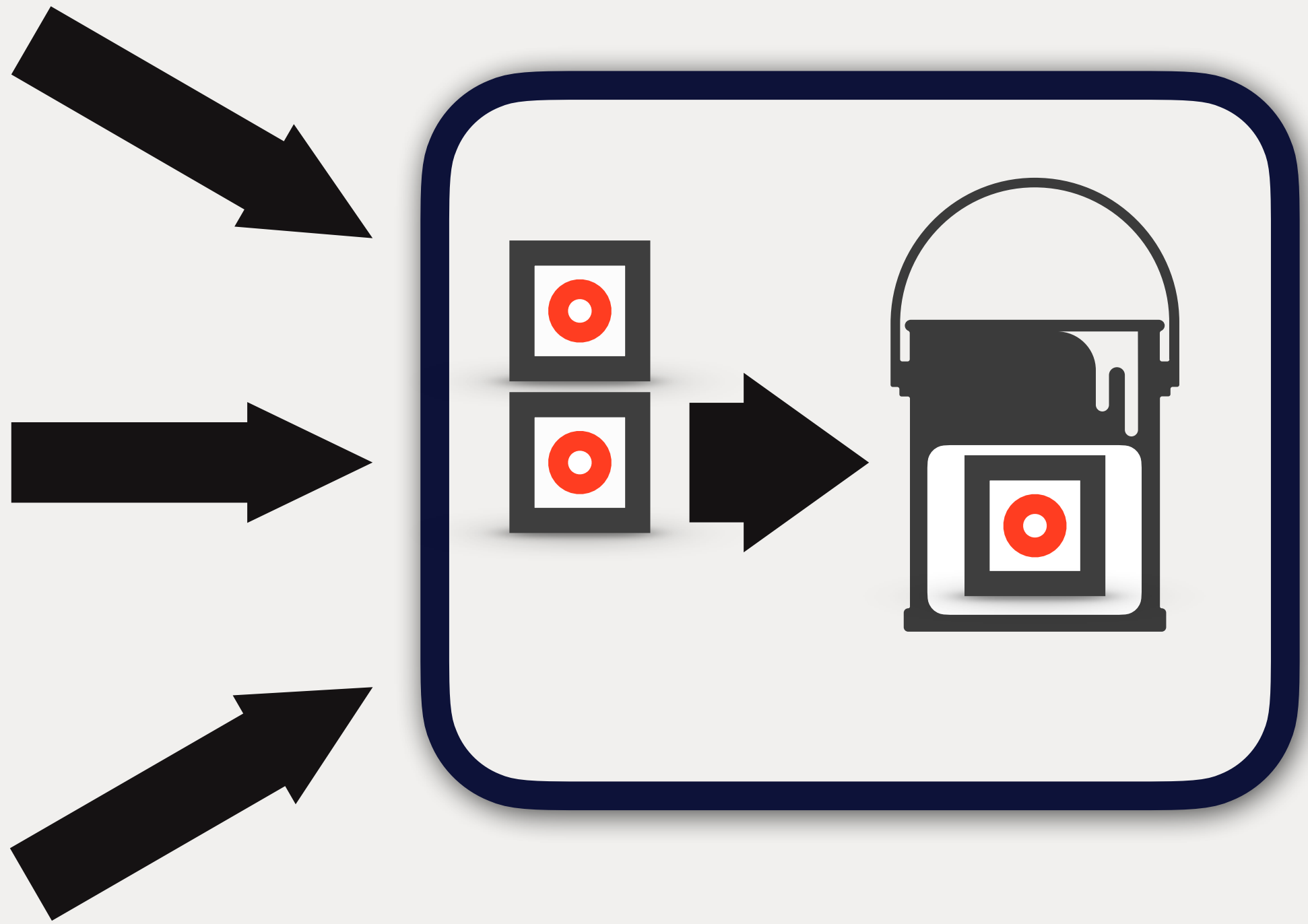
# Unbiased Reuse



$$w(x_i) = \frac{\hat{p}(x_i)}{p_{\text{RIS}}(x_i)}$$

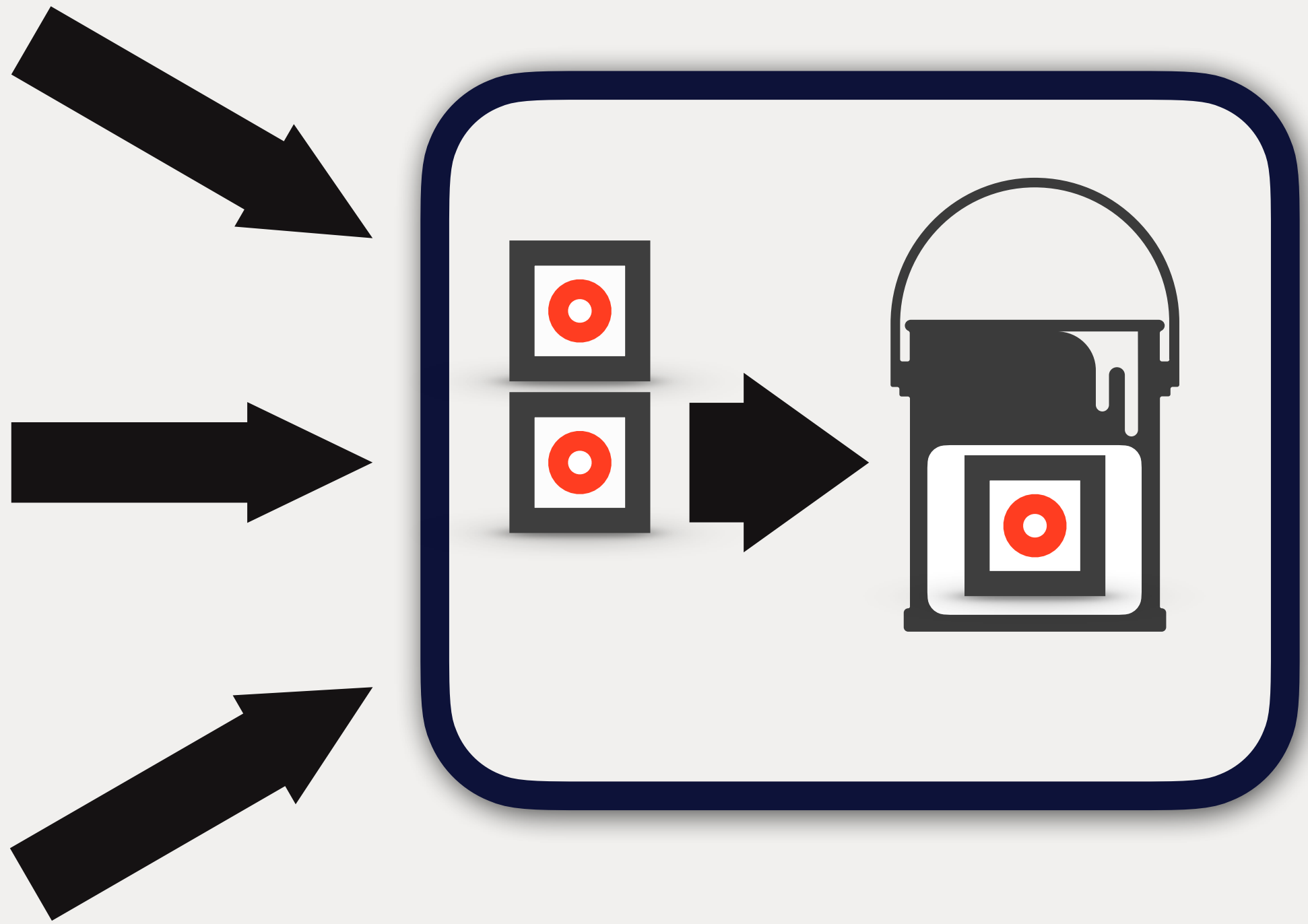


# Unbiased Reuse



$$w(x_i) = \frac{\hat{p}(x_i)}{p_{\text{RIS}}(x_i)}$$

# Unbiased Reuse



$$w(x_i) = \frac{\hat{p}(x_i)}{p_{\text{RIS}}(x_i)}$$

# Unbiased Reuse

$$p_{\text{RIS}}(x)$$

# Unbiased Reuse

$$p_{\text{RIS}}(x) = ?$$

# Unbiased Reuse

$$p_{\text{RIS}}(x) = \sum_{i=1}^M \underbrace{\int \cdots \int}_{M-1 \text{ times}} \left[ \prod_{j=1}^M p(x_j) \right] \frac{w(x_i)}{\sum_{j=1}^M w(x_j)} \underbrace{dx_1 \cdots dx_M}_{M-1 \text{ times}}$$

# Importance Resampling

[Talbot et al. '05]

$$\int_{\Omega} f(x) dx \approx f(y) \cdot W$$

$$W = \frac{1}{\hat{p}(y)} \cdot \left( \frac{1}{M} \sum_{i=1}^M w(x_i) \right)$$



# Importance Resampling

[Talbot et al. '05]

$$\int_{\Omega} f(x) dx \approx f(y) \cdot W$$

$$W = \frac{1}{\hat{p}(y)} \cdot \left( \frac{1}{M} \sum_{i=1}^M w(x_i) \right)$$

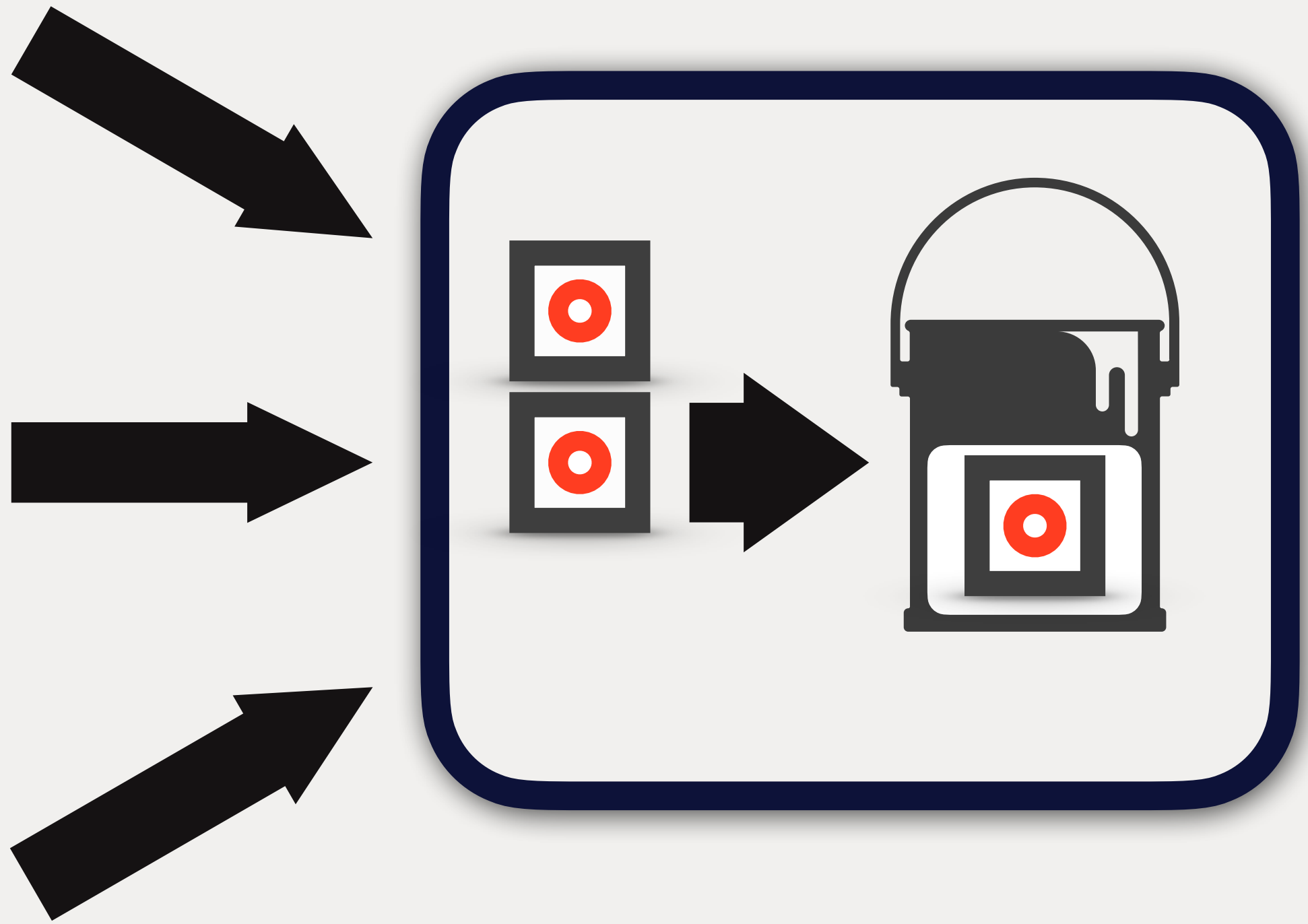
# Importance Resampling

[Talbot et al. '05]

$$\int_{\Omega} f(x) dx \approx f(y) \cdot W$$

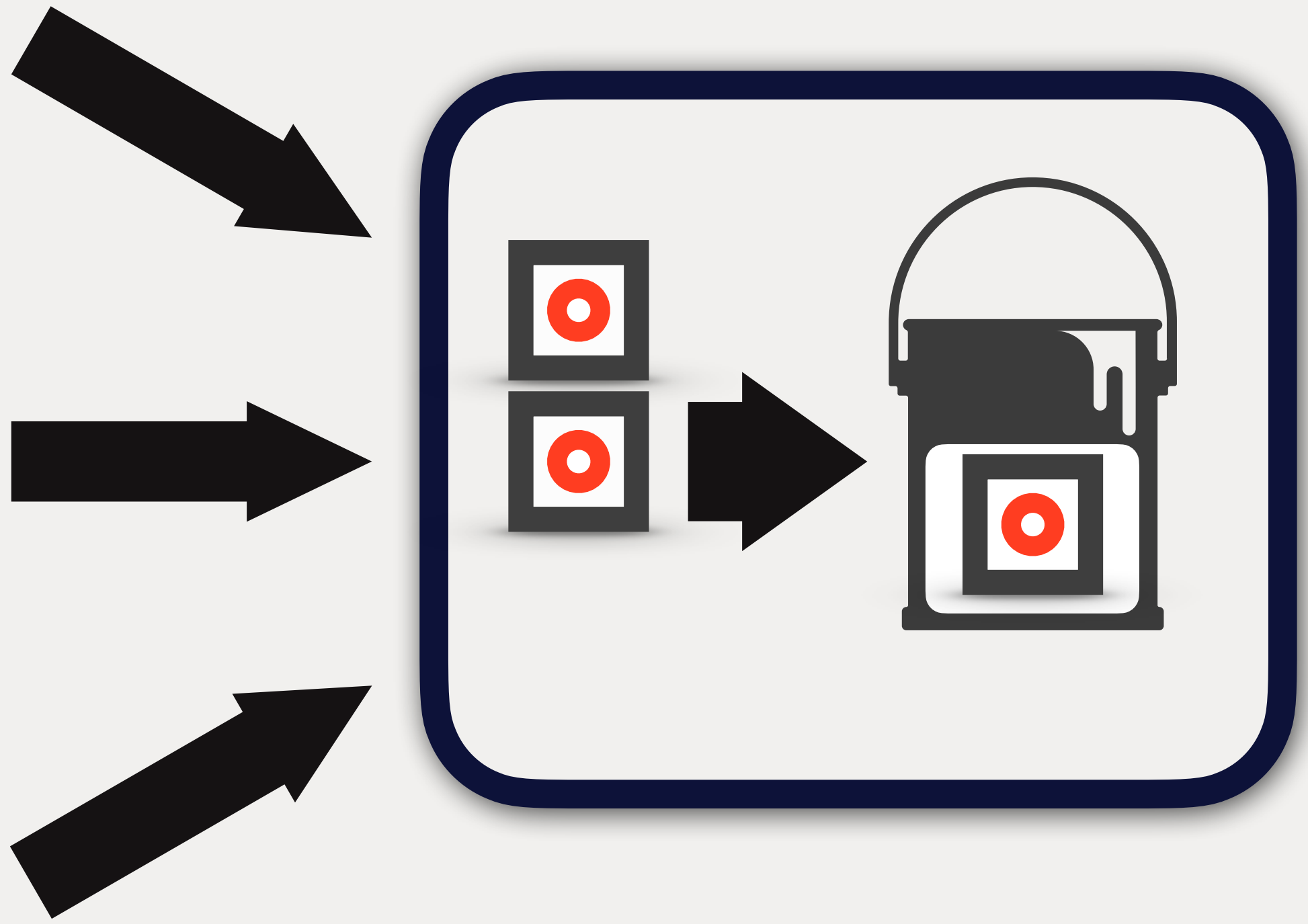
$$\mathbb{E}[W] = \frac{1}{p_{\text{RIS}}(y)}$$

# Unbiased Reuse



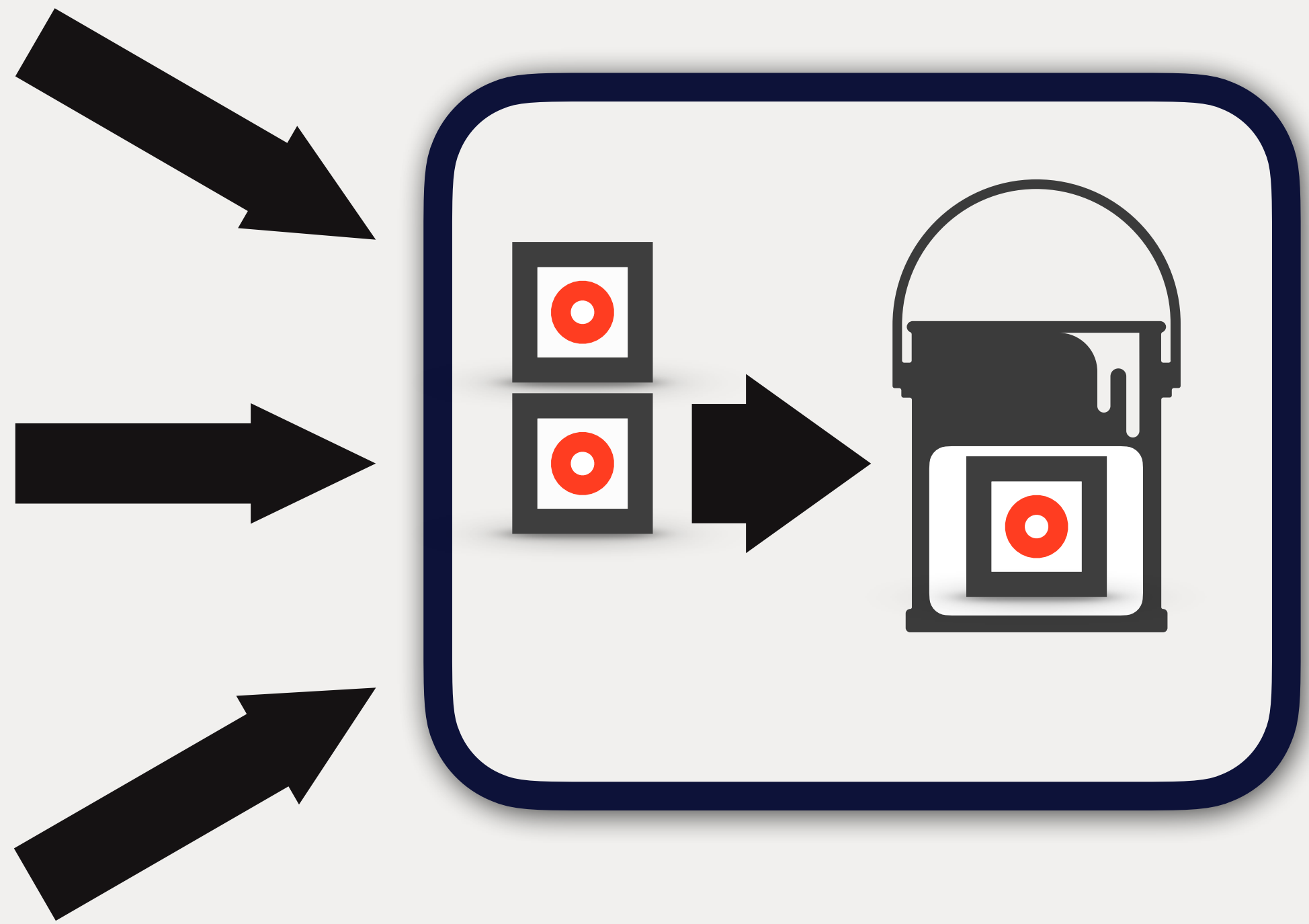
$$w(x_i) = \frac{\hat{p}(x_i)}{p_{\text{RIS}}(x_i)}$$

# Unbiased Reuse



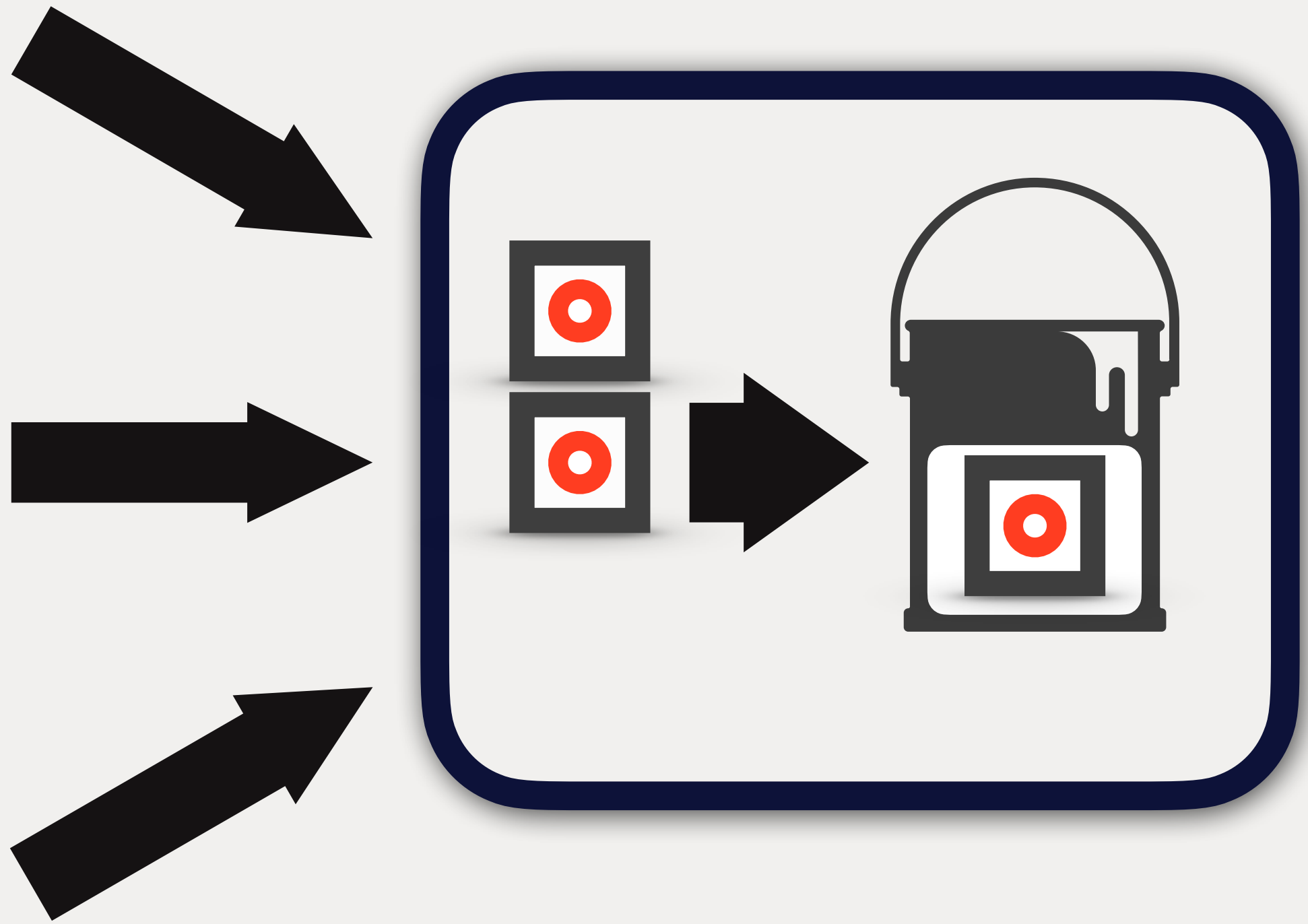
$$w(x_i) = \hat{p}(x_i) \cdot W_i$$

# Unbiased Reuse



# Unbiased Reuse

Resampling unbiased if  $p(x) > 0$  where  $\hat{p}(x) > 0$



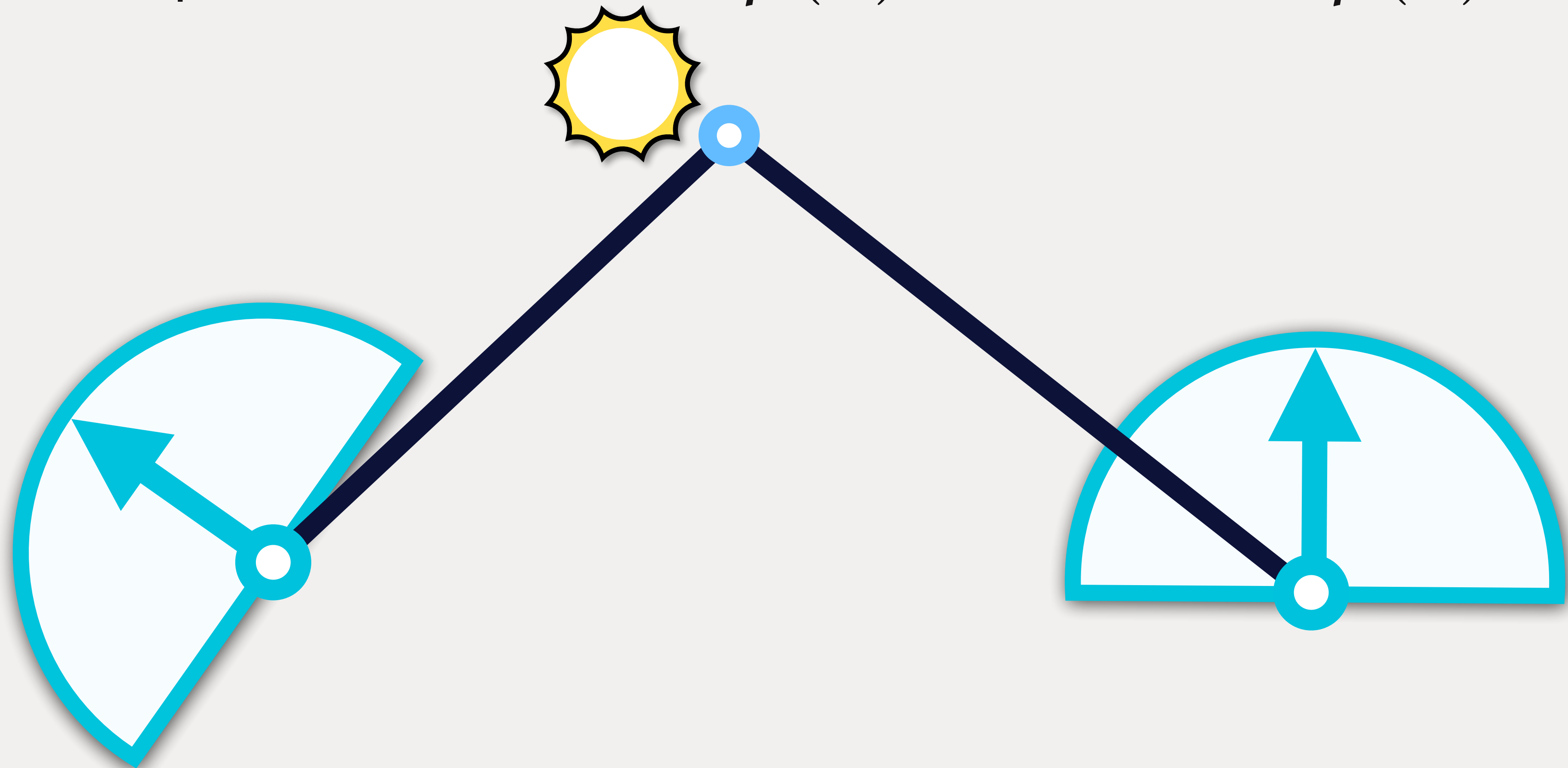


# Unbiased Reuse

Resampling unbiased if  $p(x) > 0$  where  $\hat{p}(x) > 0$

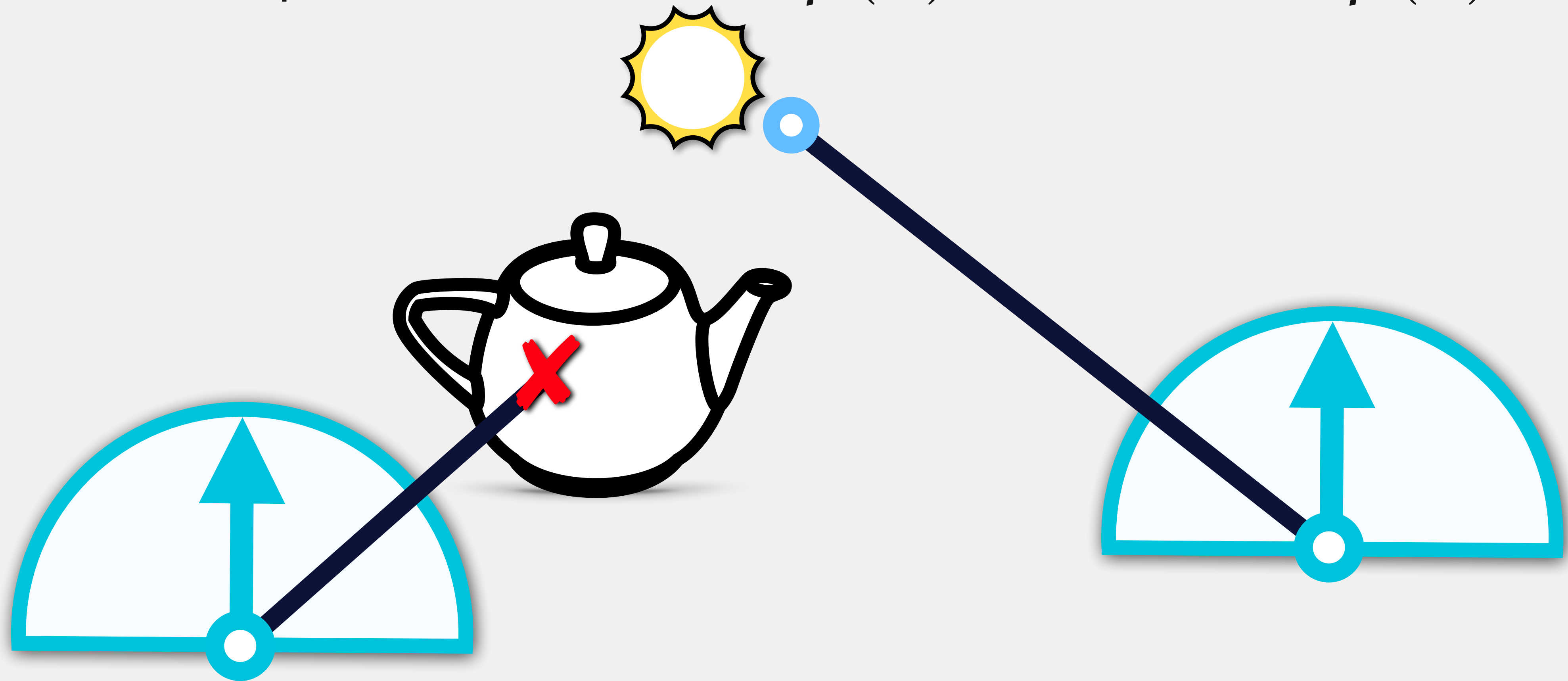
# Unbiased Reuse

Resampling unbiased if  $p(x) > 0$  where  $\hat{p}(x) > 0$



# Unbiased Reuse

Resampling unbiased if  $p(x) > 0$  where  $\hat{p}(x) > 0$



# Unbiased Reuse

$$\mathbb{E}[W] \neq \frac{1}{p_{\text{RIS}}(y)}$$

# Unbiased Reuse

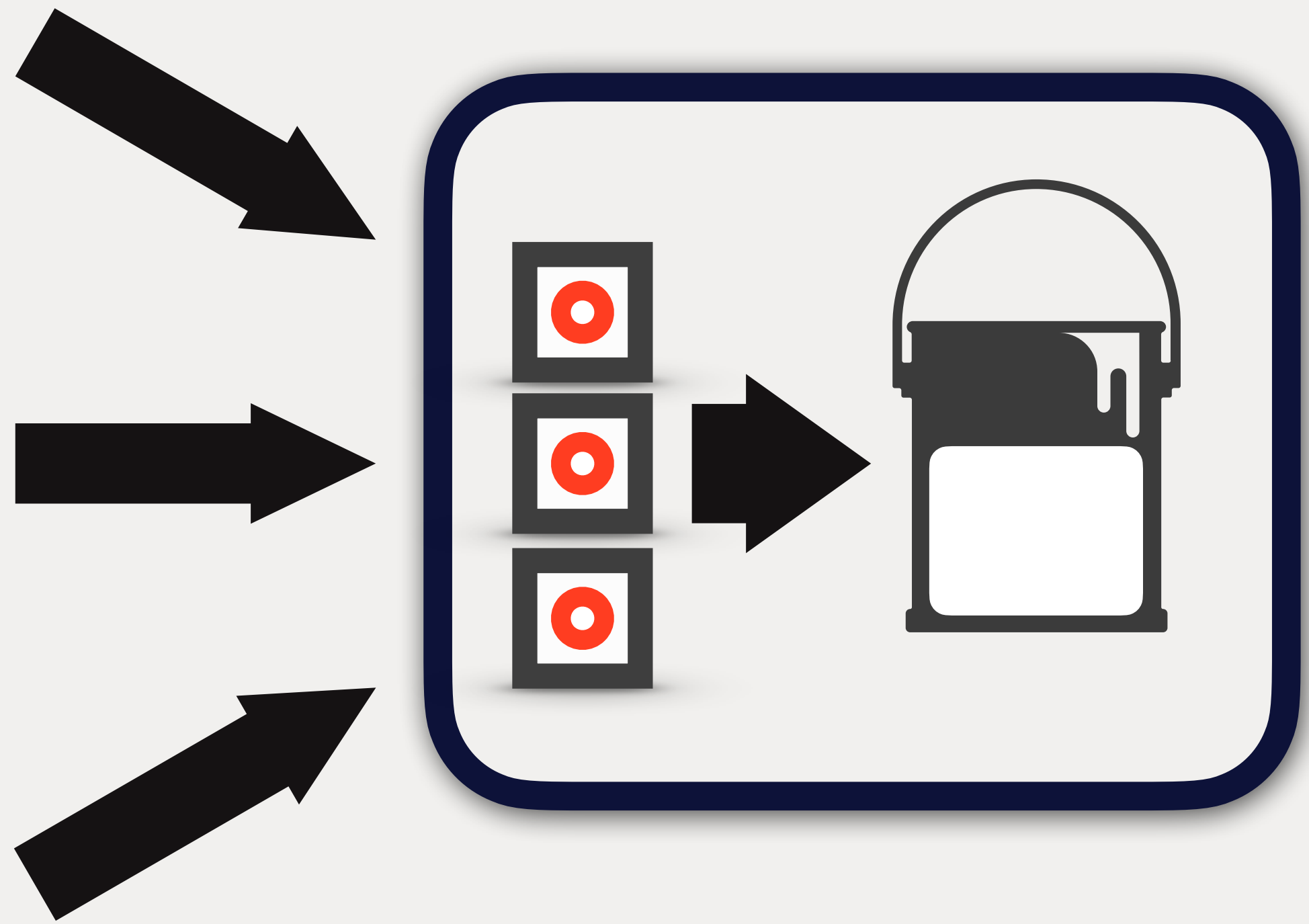
$$\mathbb{E} \left[ \frac{1}{\hat{p}(y)} \cdot \left( \frac{1}{M} \sum_{i=1}^M w(x_i) \right) \right] \neq \frac{1}{p_{\text{RIS}}(y)}$$

# Unbiased Reuse

$$\mathbb{E} \left[ \frac{1}{\hat{p}(y)} \cdot \left( \frac{1}{M} \sum_{i=1}^M w(x_i) \right) \right] \neq \frac{1}{p_{\text{RIS}}(y)}$$

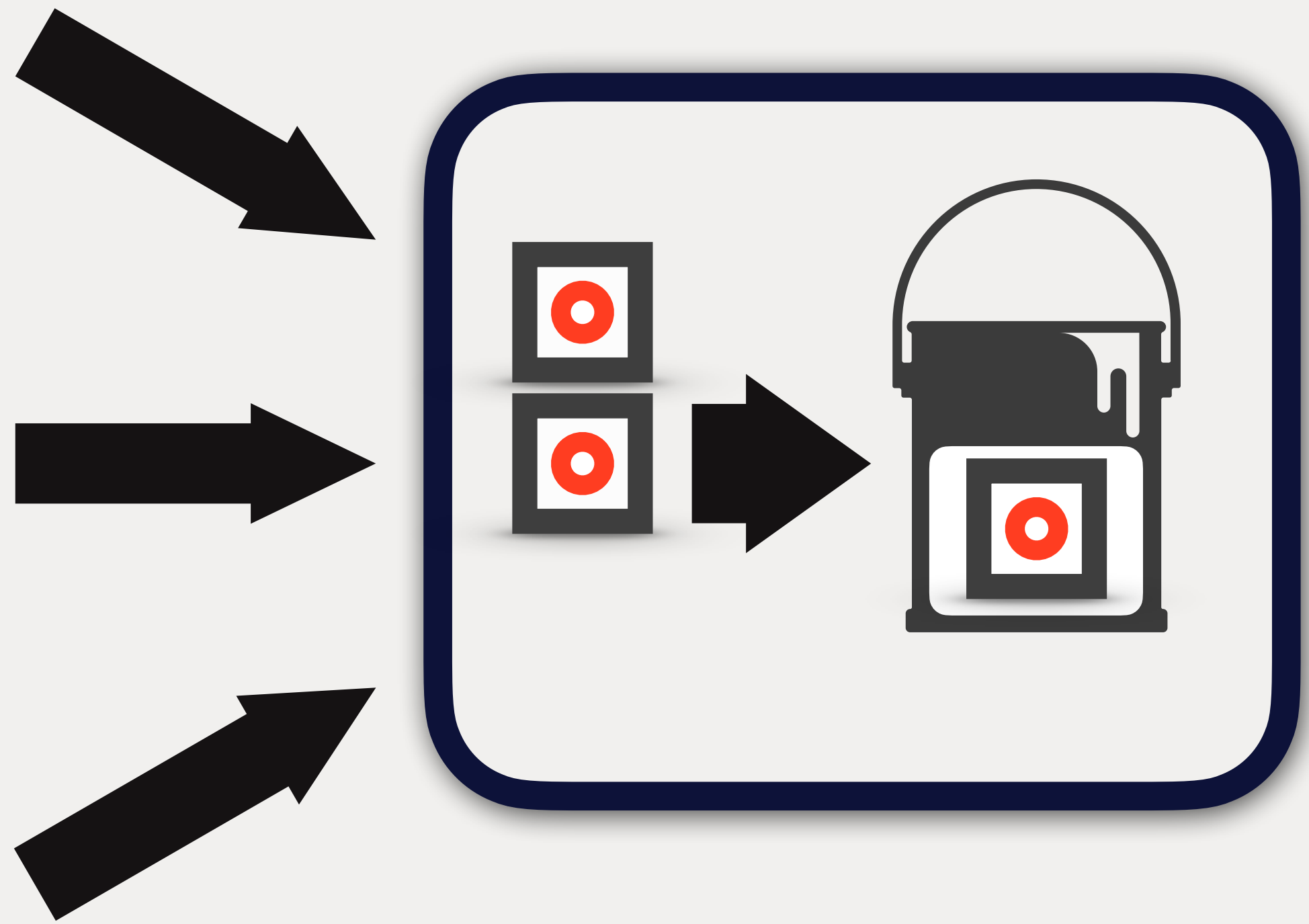


# Unbiased Reuse



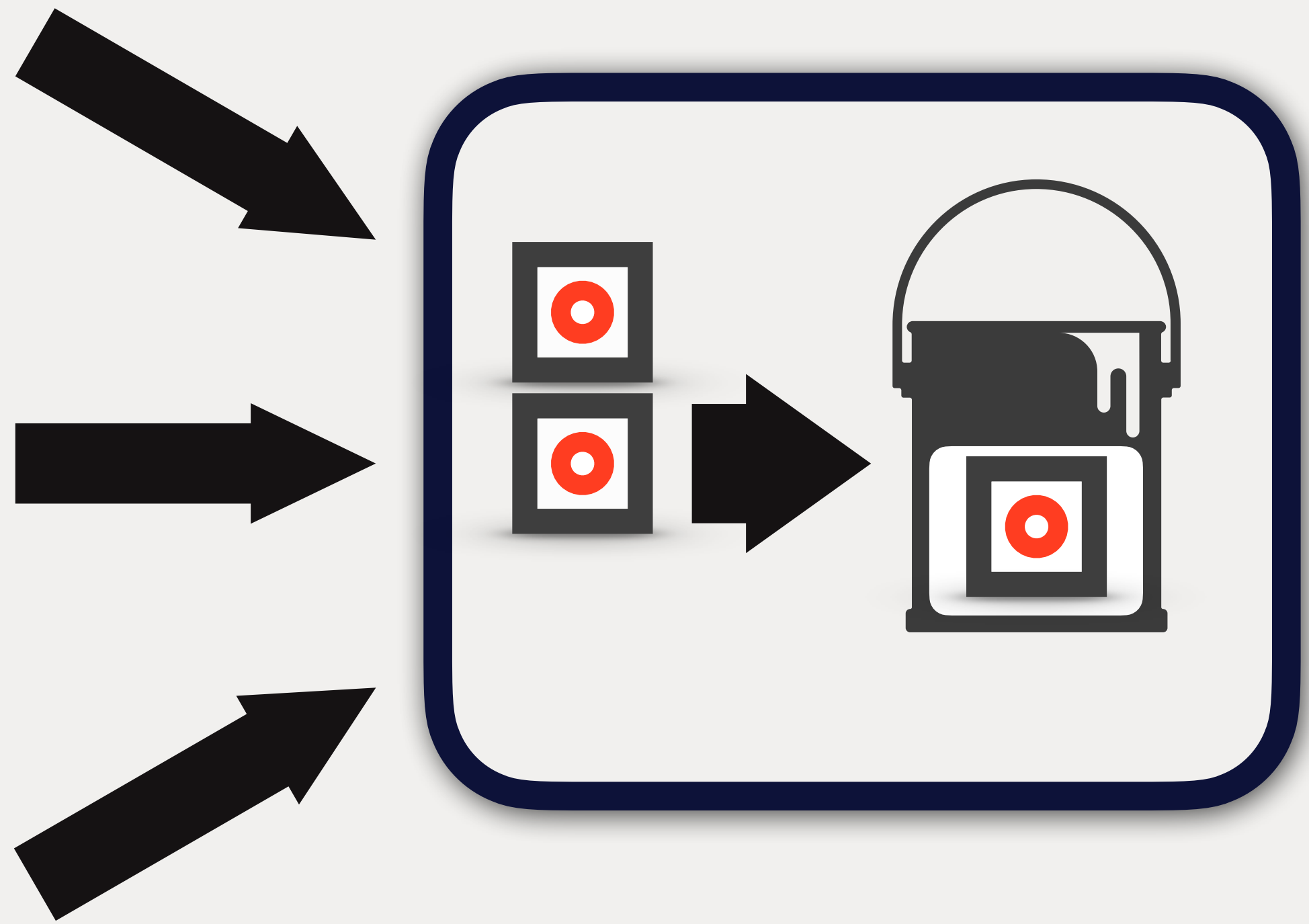
$$\mathbb{E} \left[ \frac{1}{\hat{p}(y)} \cdot \left( \frac{1}{M} \sum_{i=1}^M w(x_i) \right) \right]$$

# Unbiased Reuse



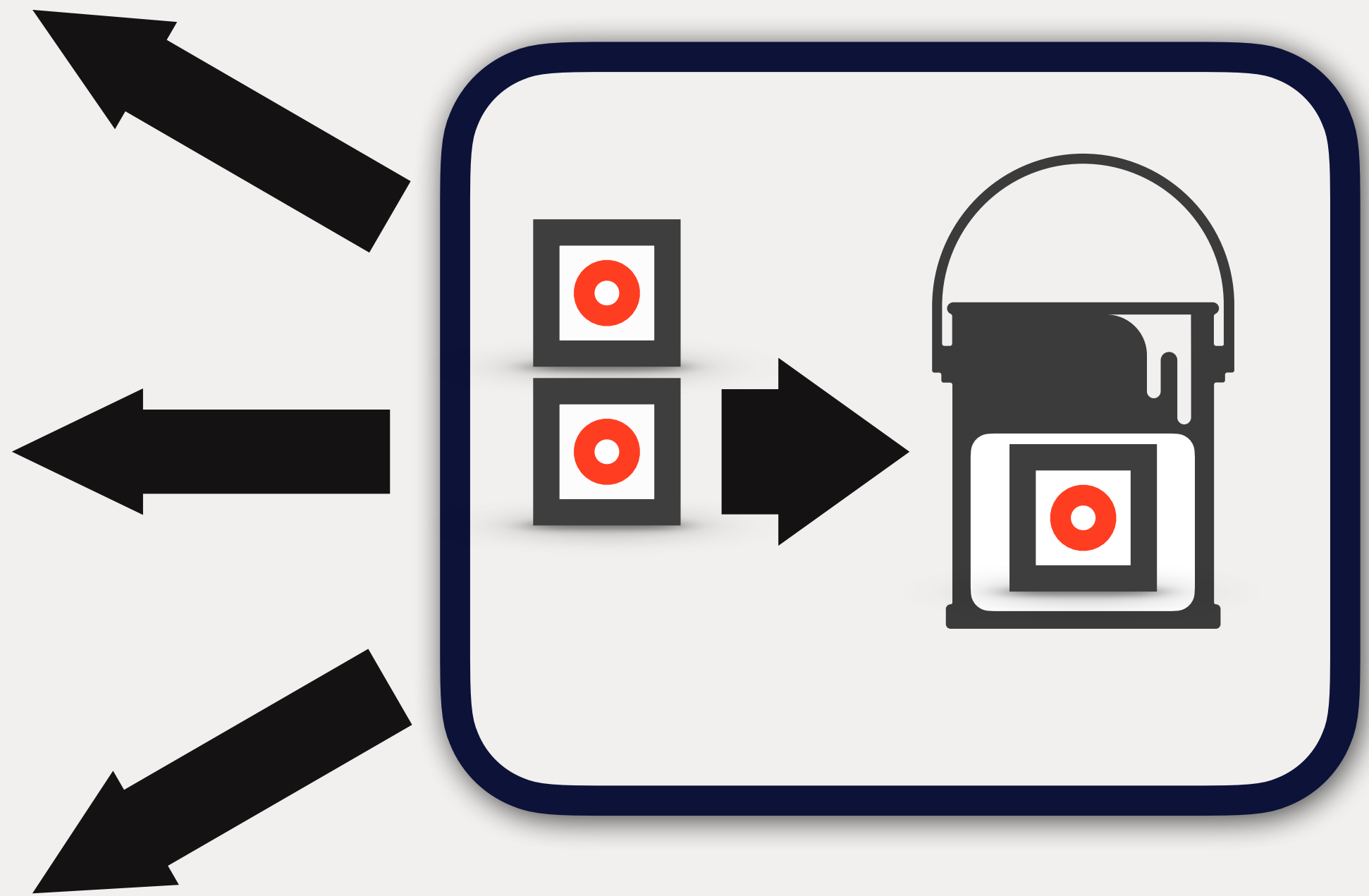
$$\mathbb{E} \left[ \frac{1}{\hat{p}(y)} \cdot \left( \frac{1}{M} \sum_{i=1}^M w(x_i) \right) \right]$$

# Unbiased Reuse



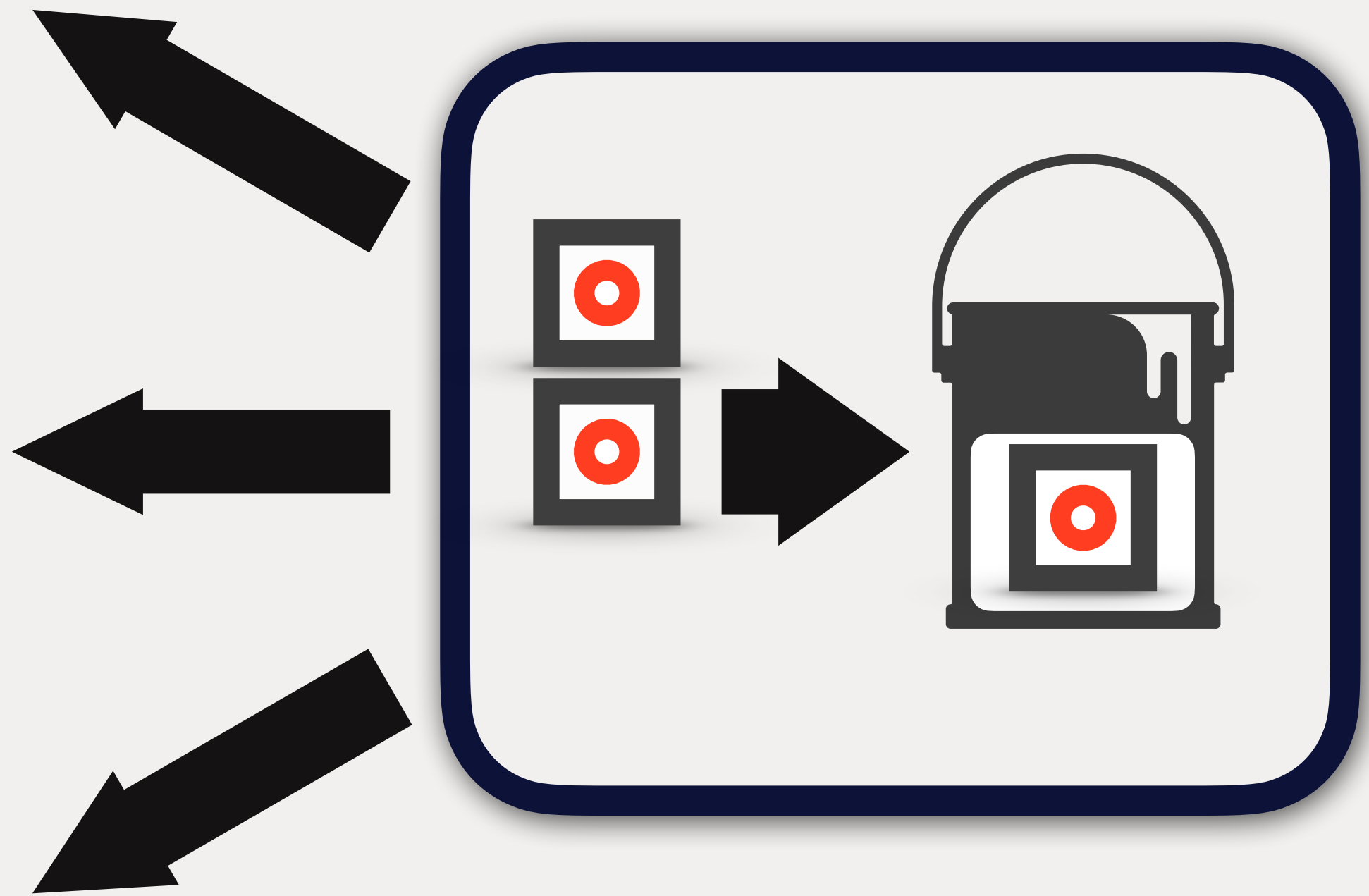
$$\mathbb{E} \left[ \frac{1}{\hat{p}(y)} \cdot \left( \frac{1}{M} \sum_{i=1}^M w(x_i) \right) \right]$$

# Unbiased Reuse



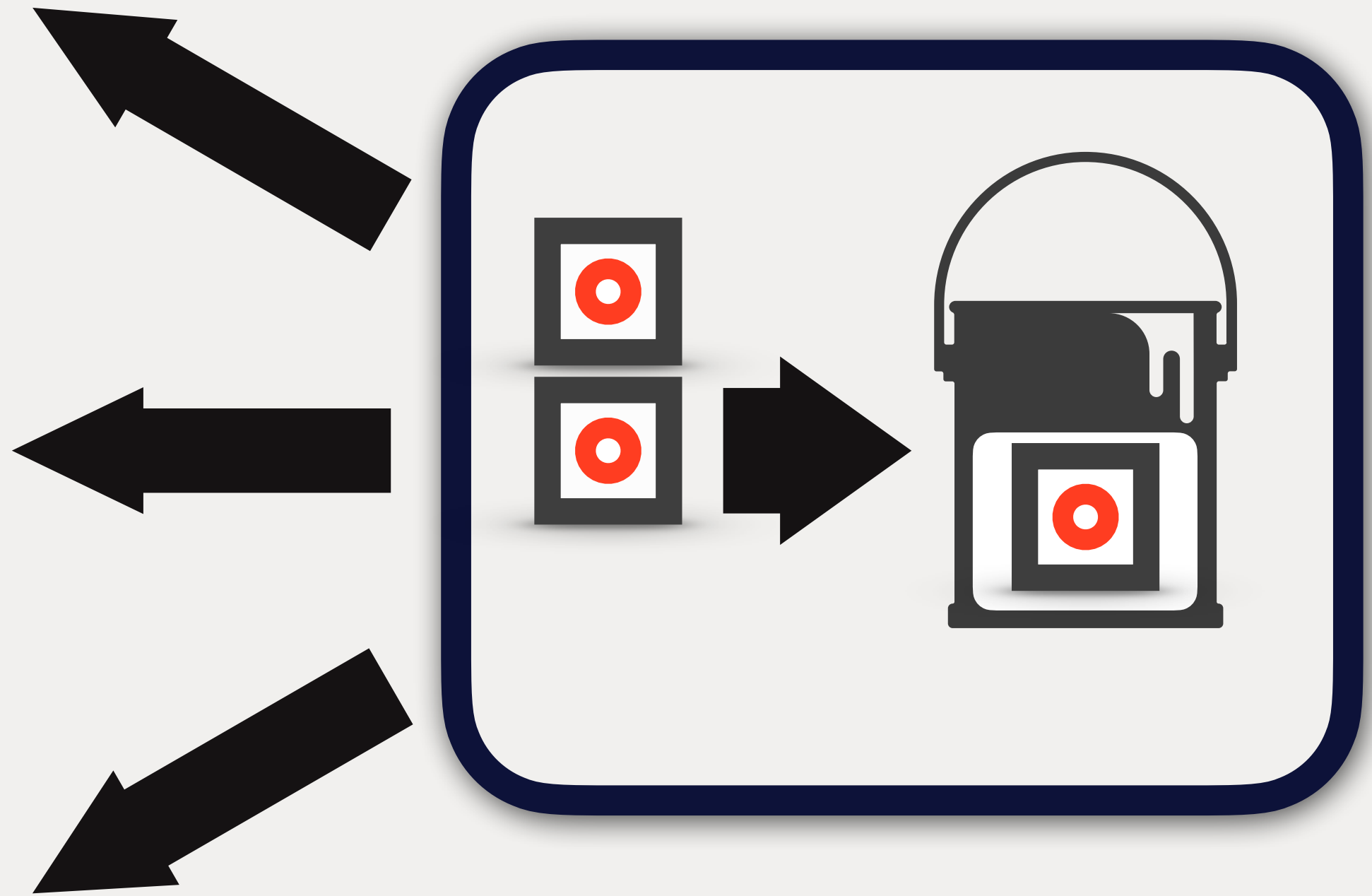
$$\mathbb{E} \left[ \frac{1}{\hat{p}(y)} \cdot \left( \frac{1}{M} \sum_{i=1}^M w(x_i) \right) \right]$$

# Unbiased Reuse



$$\mathbb{E} \left[ \frac{1}{\hat{p}(y)} \cdot \left( \frac{1}{Z} \sum_{i=1}^M w(x_i) \right) \right]$$

# Unbiased Reuse



$$\mathbb{E} \left[ \frac{1}{\hat{p}(y)} \cdot \left( m_i \sum_{i=1}^M w(x_i) \right) \right]$$

# Biased Reuse



# Biased Reuse

- Correcting bias is *expensive*

# Biased Reuse

- Correcting bias is *expensive*
- Can we minimize it instead?

# Biased Reuse

- Correcting bias is *expensive*
- Can we minimize it instead?
  - ⇒ Reject potentially troublesome neighbors

# Biased Reuse

- Correcting bias is *expensive*
- Can we minimize it instead?
  - ⇒ Reject potentially troublesome neighbors
- Compare normals/depth

# Results

# Algorithm Details

# Algorithm Details

- Implemented in Falcor [Benty et al. 2019]



# Algorithm Details

- Implemented in Falcor [Benty et al. 2019]
- Initial Resampling:  $M=32$

# Algorithm Details

- Implemented in Falcor [Benty et al. 2019]
- Initial Resampling:  $M=32$
- Temporal Reuse: Reproject with motion vectors

# Algorithm Details

- Implemented in Falcor [Benty et al. 2019]
- Initial Resampling:  $M=32$
- Temporal Reuse: Reproject with motion vectors
- Spatial Reuse: Pick 5 random neighbors in 30 pixel disk
  - Repeated twice



Reference

20'000 Emitters





[Moreau et al., 2019], 34ms

20'000 Emitters





ReSTIR (unbiased), 30ms

20'000 Emitters





ReSTIR (biased), 24ms

20'000 Emitters





Reference

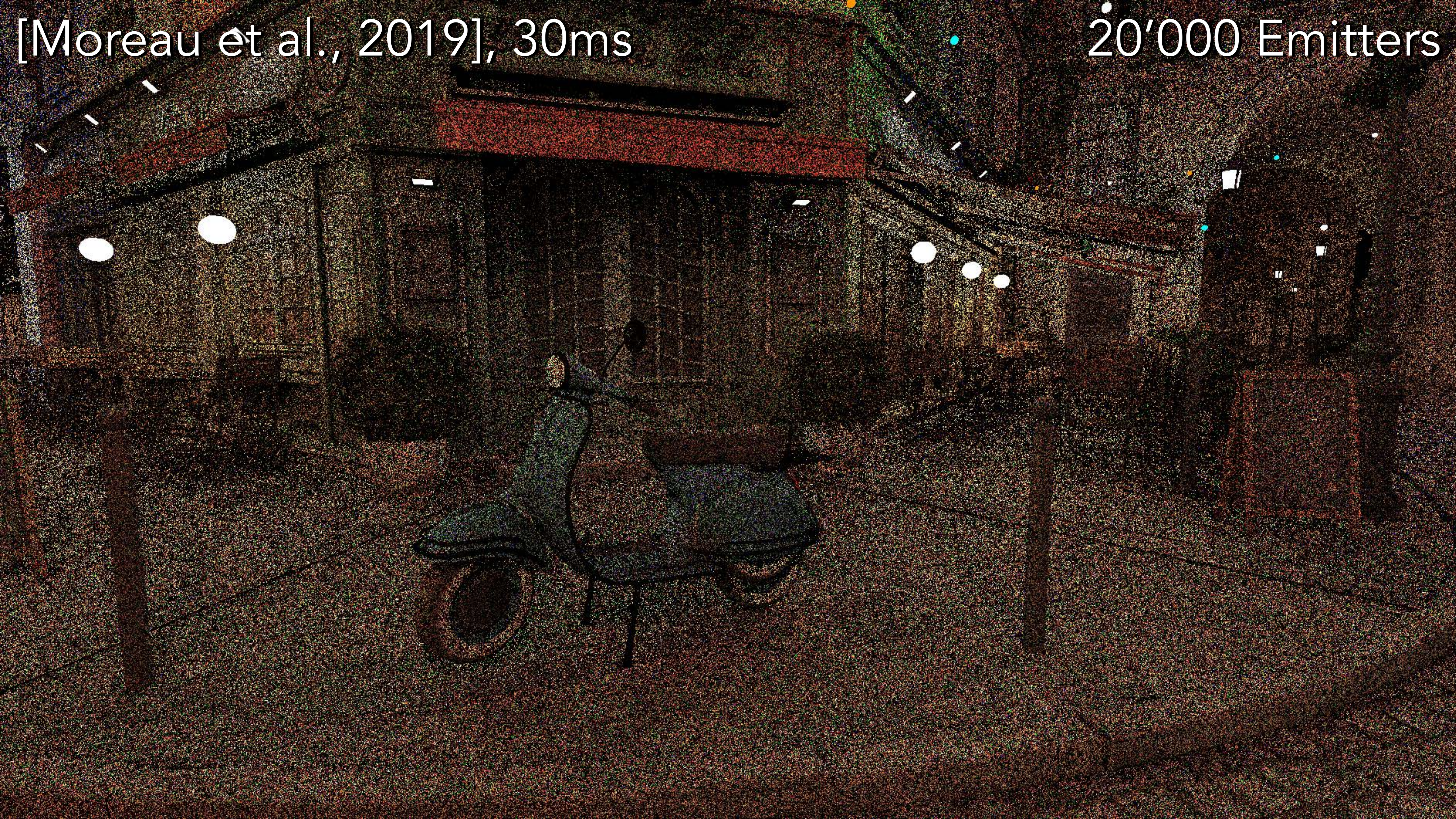
20'000 Emitters





[Moreau et al., 2019], 30ms

20'000 Emitters





ReSTIR (unbiased), 26ms

20'000 Emitters







ReSTIR (biased), 21ms

20'000 Emitters



Reference

23'000 Emitters





[Moreau et al., 2019], 29ms

23'000 Emitters





ReSTIR (unbiased), 17ms

23'000 Emitters





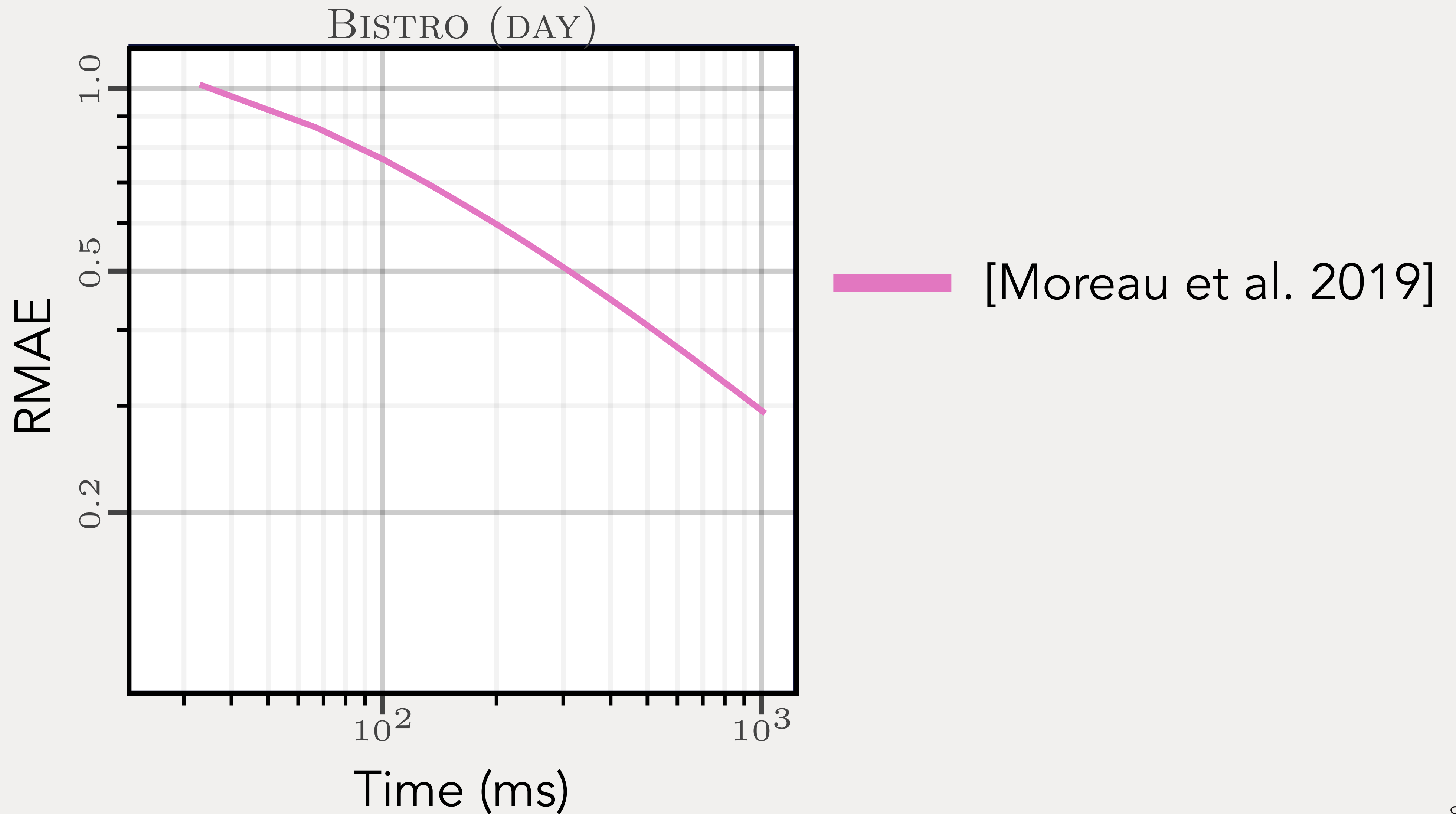
ReSTIR (biased), 16ms

23'000 Emitters



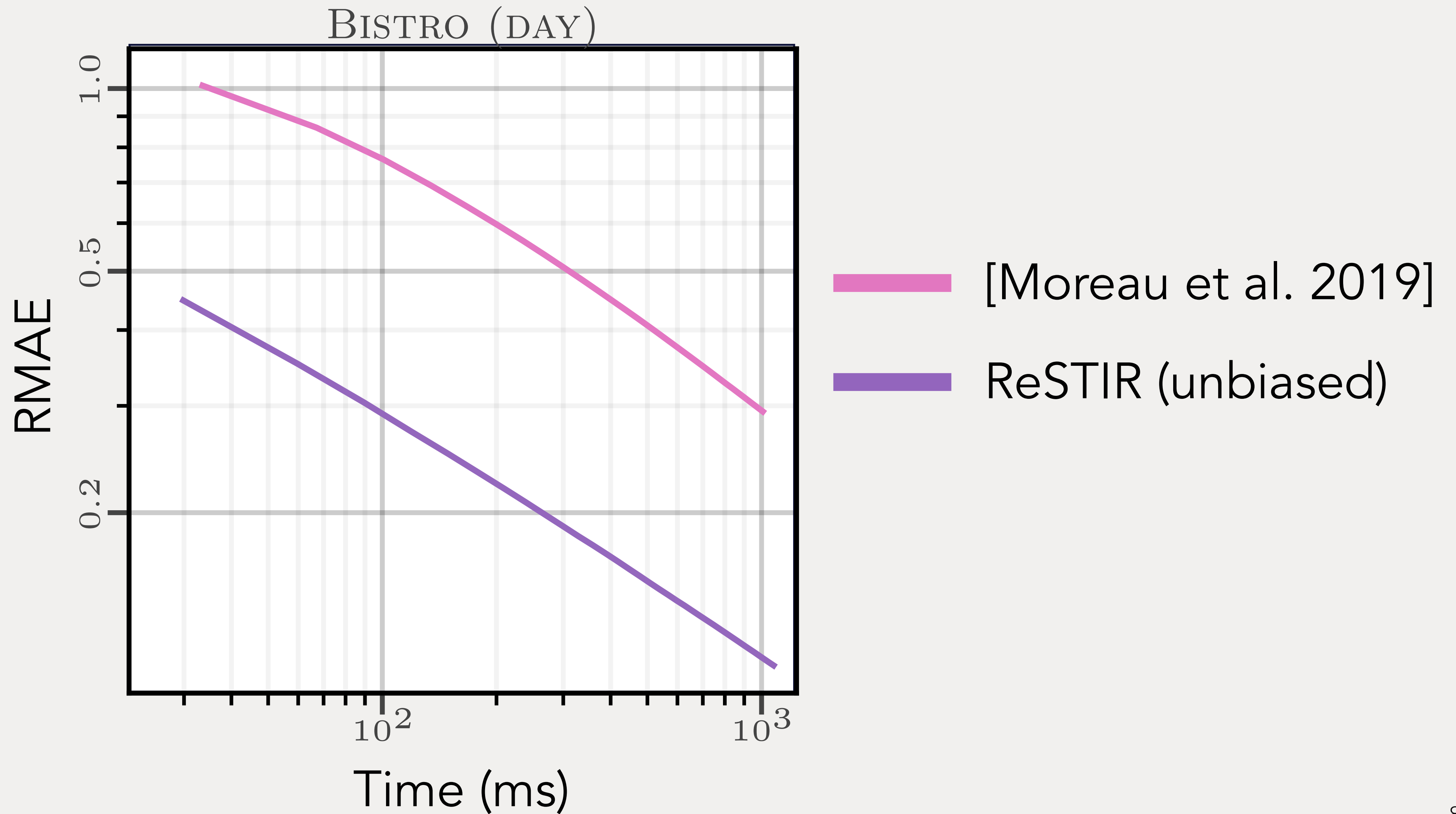


# Convergence Graphs

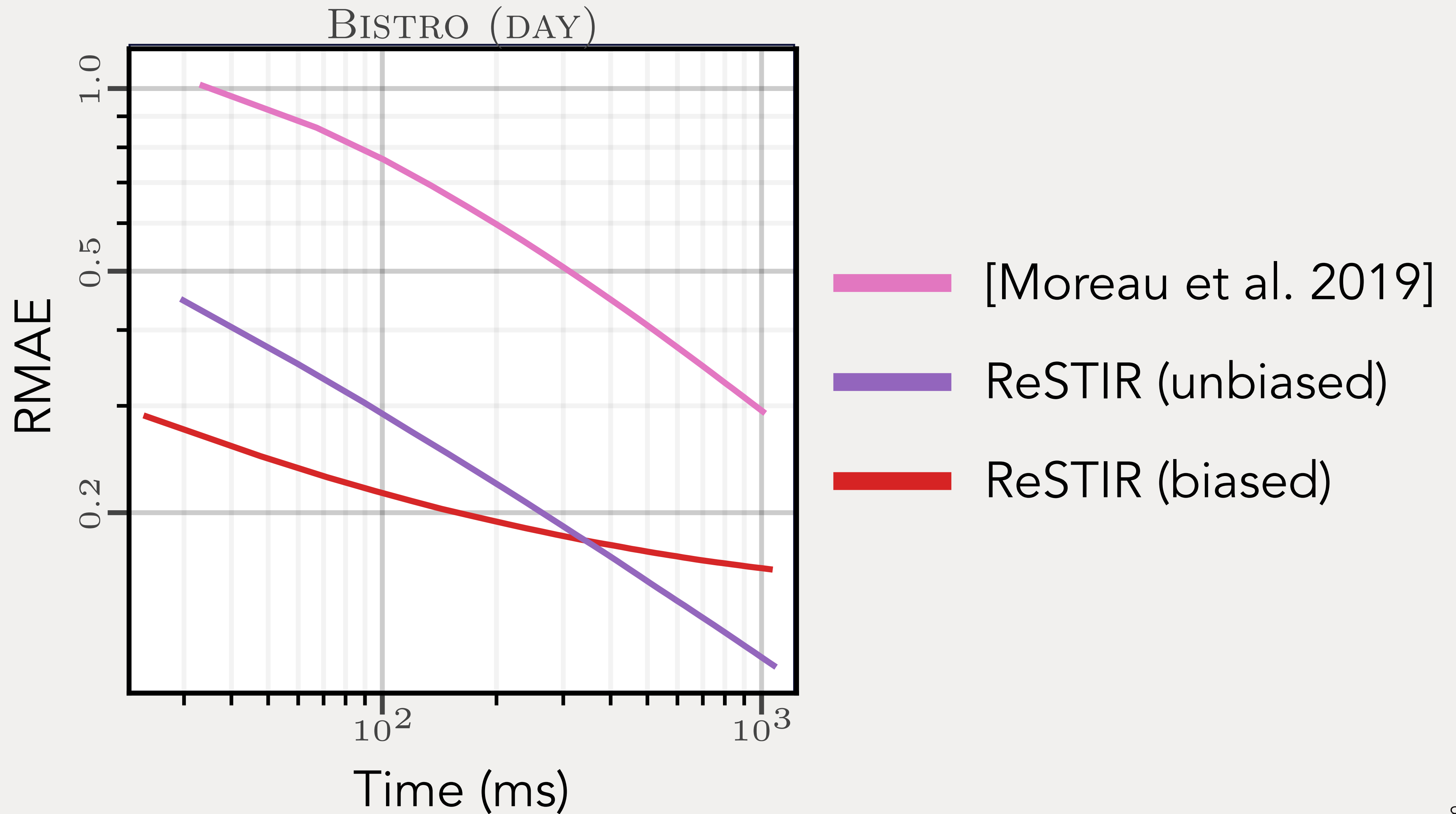




# Convergence Graphs

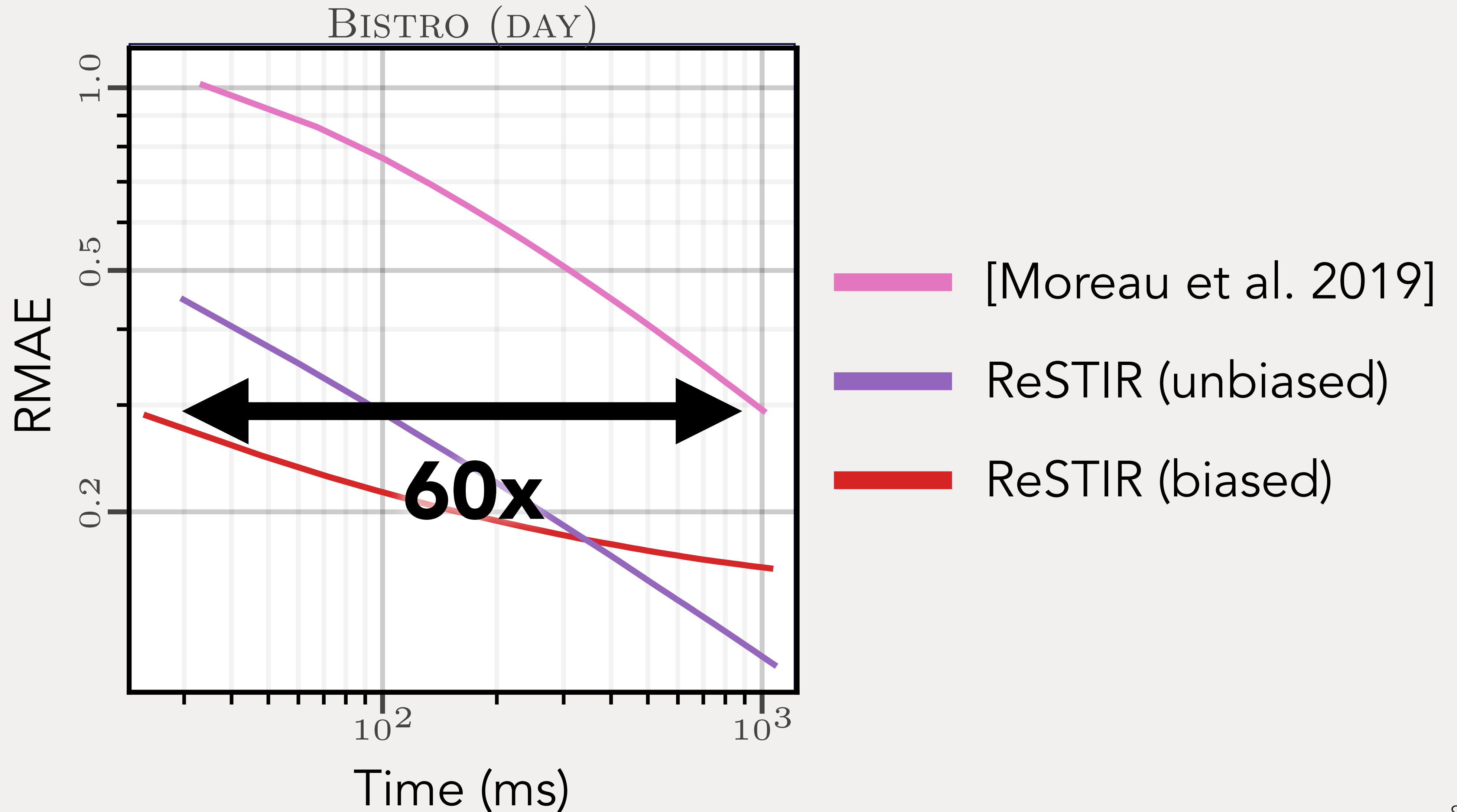


# Convergence Graphs

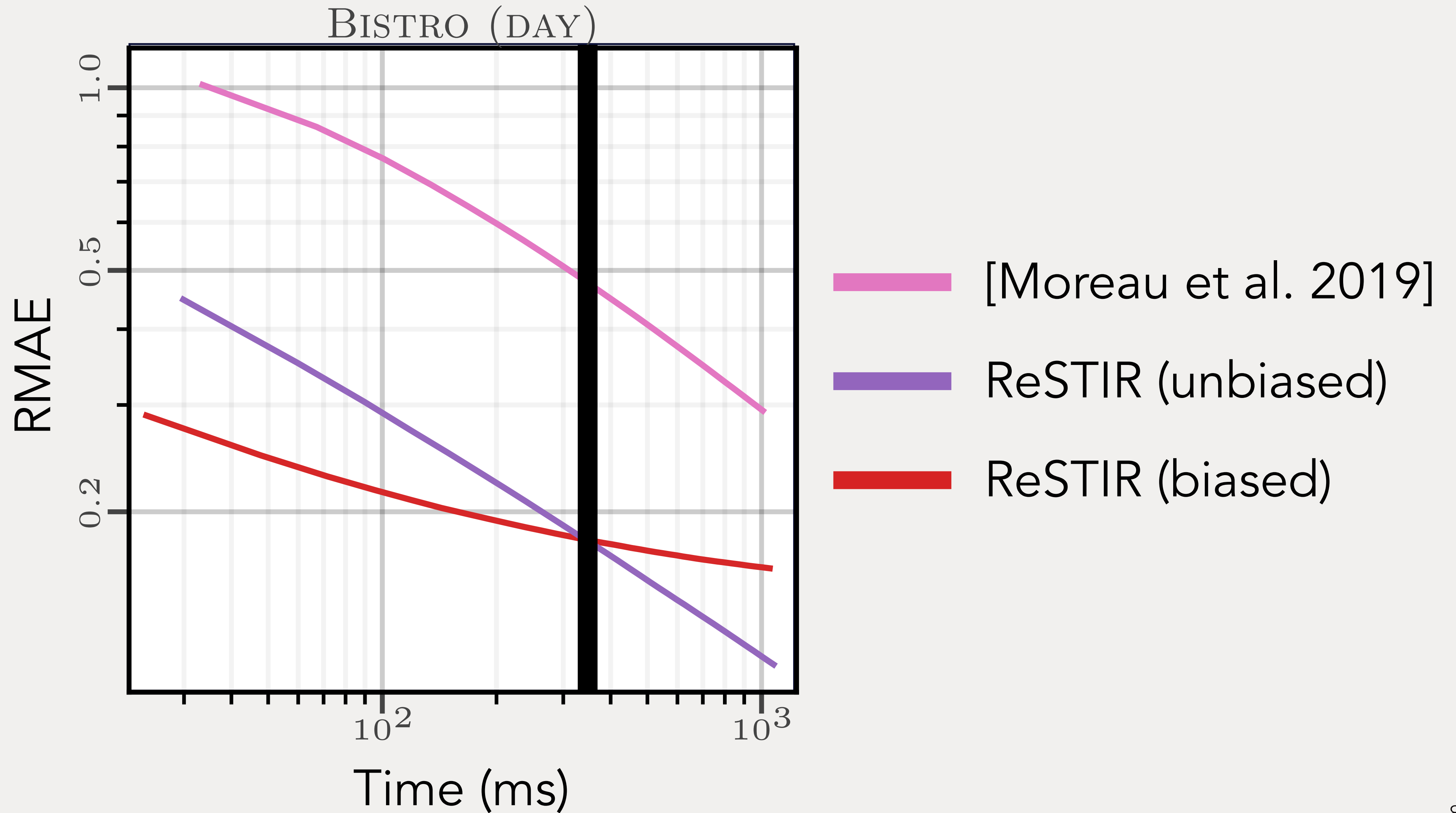




# Convergence Graphs



# Convergence Graphs





# Conclusion

# Conclusion

- Algorithm for sampling arbitrary distributions



# Conclusion

- Algorithm for sampling arbitrary distributions
- Start with “bad” samples, improve them

# Conclusion

- Algorithm for sampling arbitrary distributions
- Start with “bad” samples, improve them
- Reuse bad samples over multiple sample points



# Limitations & Future Work

# Limitations & Future Work

- Higher order bounces?



# Limitations & Future Work

- Higher order bounces?
- World space?

# Limitations & Future Work

- Higher order bounces?
- World space?
- How to best combine candidates from different distributions?



# Limitations & Future Work

- Higher order bounces?
- World space?
- How to best combine candidates from different distributions?
- Disocclusions, visibility boundaries: Reuse limited

# Limitations & Future Work

- Higher order bounces?
- World space?
- How to best combine candidates from different distributions?
- Disocclusions, visibility boundaries: Reuse limited
- BRDF etc. too expensive: Proxy needs to be used









Le Petit Coin

Le Petit Coin

Restaurant

Le Petit Coin

MENU

MENU





Burger Hit



Hit Cols







LIBRAIRIE





LIBRAIRIE