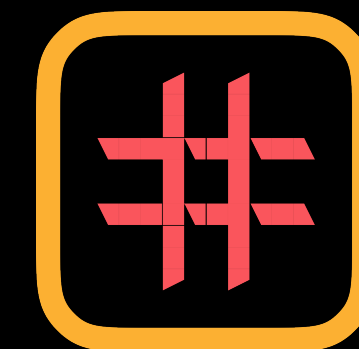


A surprising real world 70x performance jump



PRAGMA
CONFERENCE



Sparkle



Sparkle

Balloon Travel 65% Zoom Device Grid Page Blog Text Layout Block Box Image Add SEO Settings Preview Publish

Pages Layers

Home Sites Packages Book

Balloon Travel

Ballooning Sites

- Savannah, Tennessee**
From vibrant pyrotechnics to an unparalleled view of farm, agriculture, nature, and an unforgettable dining scene, this town...
- Albuquerque, New Mexico**
The artwork and the light that illuminates the desert and Balloon Fiesta grounds, with hundreds of balloons rising to grace...
- Queenstown, New Zealand**
New Zealand's adventure capital capitalizes on its rugged terrain and scenic views to provide a unique and unforgettable experience...

To most people, the sky is the limit. To those who love to fly, it's just the beginning.

Page 1

Ballooning Sites

Style Arrange Page

Image

Bitmap (PNG, JPEG, GIF)

FILENAME
pasted-image.png

TITLE (TOOLTIP)

DESCRIPTION

1834x2362 px
Replace...
Remove

Stretch Fill Fit

1x 2x 3x +

Image resolution is optimal for 3x, element can be made larger if desired.

Edit Background
Note: background removal is optimized for foreground subjects with a good contrast over the background.

Border
 Rounding
 Shadow
 Opacity
 Filters

On Click
Do Nothing

Scroll Effects
Animate While Scrolling



Sparkle

65%
Zoom
Device
Grid
Page
Blog
Text
Layout Block
Box
Image
Add

SEO
Settings
Preview
Publish

Pages
Layers

Style
Arrange
Page

Page 1

Text Wrap

65%

Zoom Device Grid Page Blog Text Layout Block Box Image Add

SEO Settings Preview Publish

Style Arrange Page

Image

Bitmap (PNG, JPEG, GIF)

FILENAME

hot-air-balloon-4761_1920.png

TITLE (TOOLTIP)

DESCRIPTION

1920x1440 px

Replace...
Remove

Stretch
 Fill
 Fit

1x
2x
3x

Image resolution sufficient for 2x, larger image or smaller element required for 3x.

Edit Background

Note: background removal is optimized for foreground subjects with a good contrast over the background.

Border
 Rounding
 Shadow
 Opacity
 Filters

On Click

Do Nothing

Scroll Effects

None

Text

Text Wrap

65%

Zoom Device Grid Page Blog Text Layout Block Box Image Add

SEO Settings Preview Publish

Style Arrange Page

Image

Bitmap (PNG, JPEG, GIF)

FILENAME

hot-air-balloon-4761_1920.png

TITLE (TOOLTIP)

DESCRIPTION

1920x1440 px

Replace...
Remove

Stretch
 Fill
 Fit

1x
2x
3x

Image resolution sufficient for 2x, larger image or smaller element required for 3x.

Edit Background

Note: background removal is optimized for foreground subjects with a good contrast over the background.

Border
 Rounding
 Shadow
 Opacity
 Filters

On Click

Do Nothing

Scroll Effects

None



Sparkle


65%
Zoom
Device
Grid
Page
Blog
Text
Layout Block
Box
Image
Add

SEO
Settings
Preview
Publish

Pages
Layers

Style
Arrange
Page

Page 1



1920x1440 px

Replace...

Remove

Stretch Fill Fit

1x 2x 3x

Image resolution sufficient for 2x, larger image or smaller element required for 3x.

Edit Background

Note: background removal is optimized for foreground subjects with a good contrast over the background.

Border

Rounding

Shadow

Opacity

Filters

On Click

Do Nothing

Scroll Effects

None



Sparkle

Balloon Travel 65% Zoom Device Grid Page Blog Text Layout Block Box Image Add SEO Settings Preview Publish

Pages Layers

Home Sites Packages Book

Balloon Travel

Ballooning Sites

- Savannah, Tennessee**
From vibrant pyrotechnics and fireworks to a variety of hot air balloon rides, Savannah is a hot air balloon capital.
- Albuquerque, New Mexico**
The world's largest hot air balloon festival is held in Albuquerque, New Mexico, each year.
- Queenstown, New Zealand**
New Zealand's adventure capital capital Queenstown is a hot air balloon capital.

To most people, the sky is the limit. To those

Page 1

Ballooning Sites

Do Nothing

Animate While Scrolling

Image

Bitmap (PNG, JPEG, GIF)

FILENAME
pasted-image.png

TITLE (TOOLTIP)

DESCRIPTION

1834x2362 px
Replace...
Remove

Stretch Fill Fit

1x 2x 3x +
Image resolution is optimal for 3x, element can be made larger if desired.

Edit Background
Note: background removal is optimized for foreground subjects with a good contrast over the background.

Border
 Rounding
 Shadow
 Opacity
 Filters

On Click
Do Nothing

Scroll Effects
Animate While Scrolling



Sparkle



balloon-1.png



balloon-2.png



fuji.jpg



Sparkle



balloon-1.png



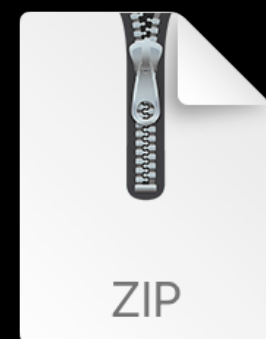
balloon-2.png



fuji.jpg



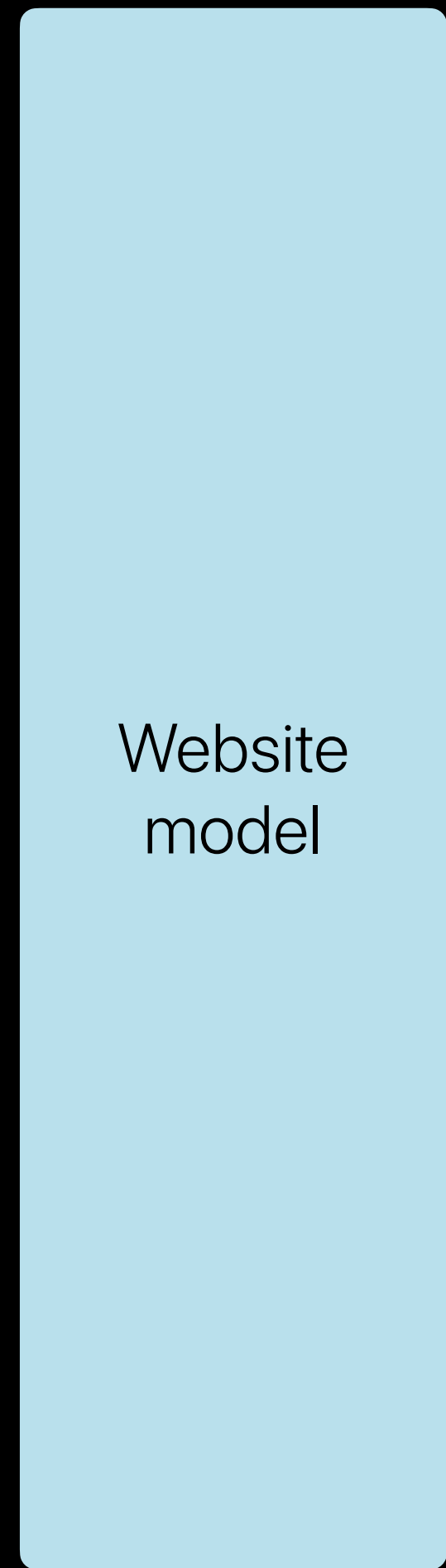
paper.pdf



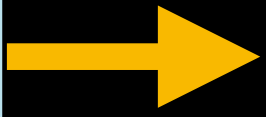
download.zip



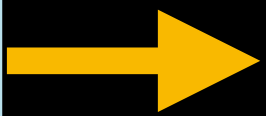
Sparkle



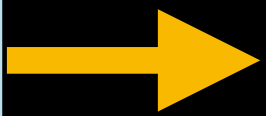
Website
model



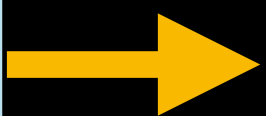
balloon-1.png



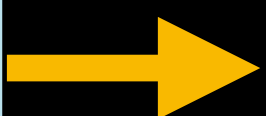
balloon-2.png



fuji.jpg



paper.pdf



download.zip



Sparkle



website.sparkle



Sparkle

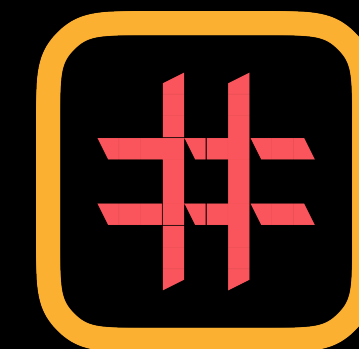


Sparkle

- Objective-C runtime hacks
- Low level filesystem stuff
- Obscure SQL

...for fun & profit!

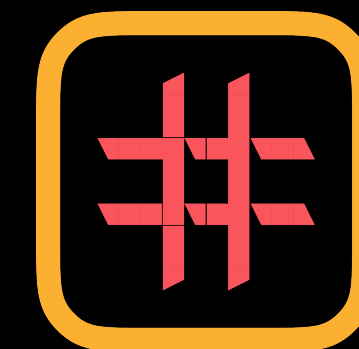
A surprising real world 70x performance jump



PRAGMA
CONFERENCE

**A surprising real world 70x
performance jump**

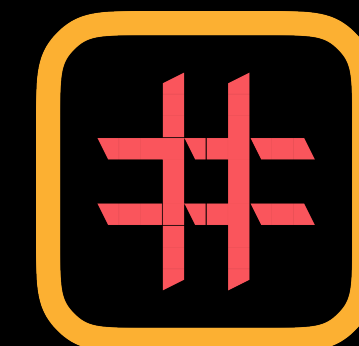
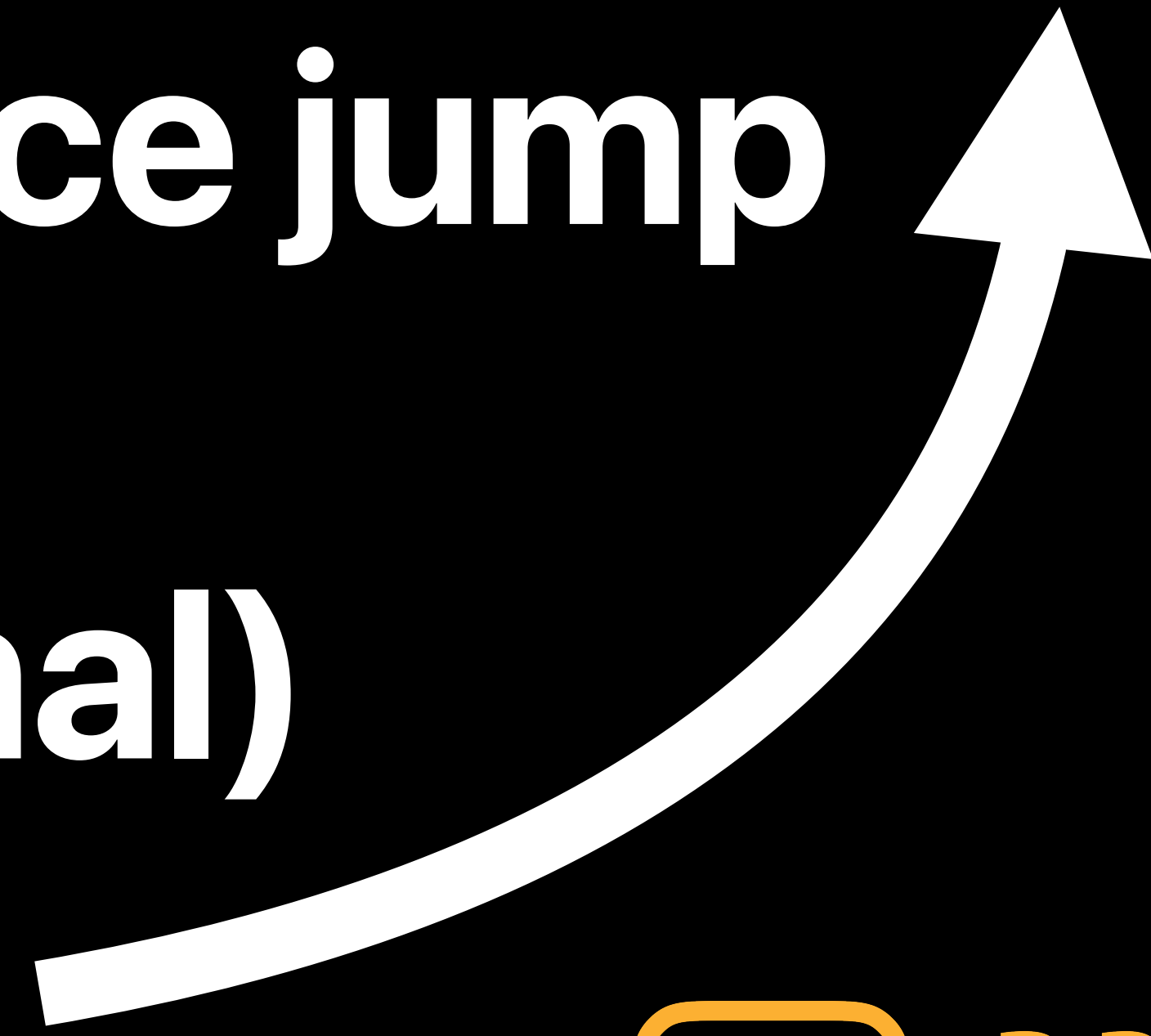
clickbait



**PRAGMA
CONFERENCE**

**A surprising real world 70x
performance jump**

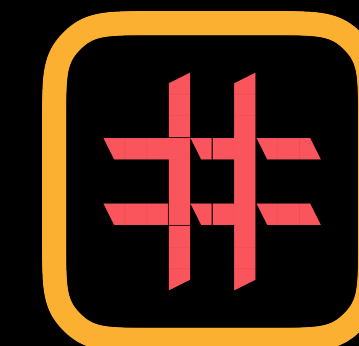
**(unintentional)
clickbait**



**PRAGMA
CONFERENCE**

**A surprising real world ~30x
performance jump**

**(unintentional)
clickbait**



**PRAGMA
CONFERENCE**

The problem



website.sparkle



website.sparkle



website.sparkle

- 35 GB
- 300 pages
- 100k words
- 800 images
- 230 videos
- 400 attached files



large website.sparkle

- 35 GB
- 300 pages
- 100k words
- 800 images
- 230 videos
- 400 attached files



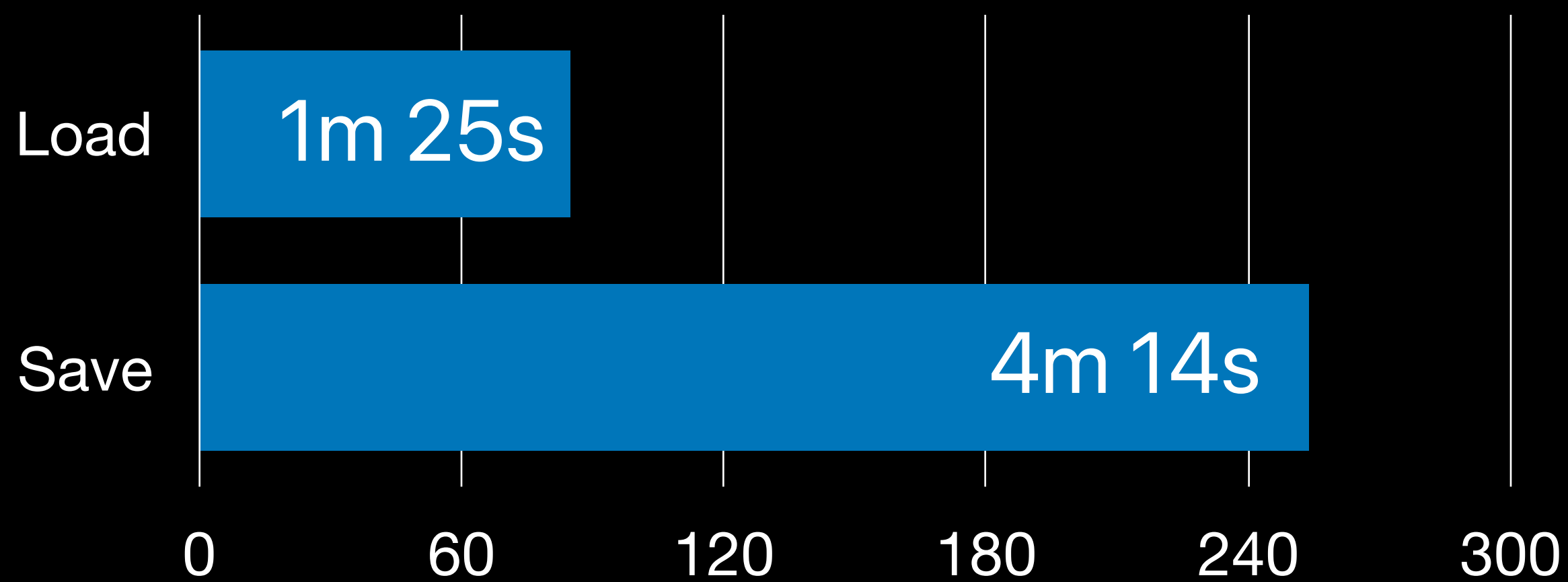
large website.sparkle

- 35 GB
- 300 pages
- 100k words
- 800 images
- 230 videos
- 400 attached files



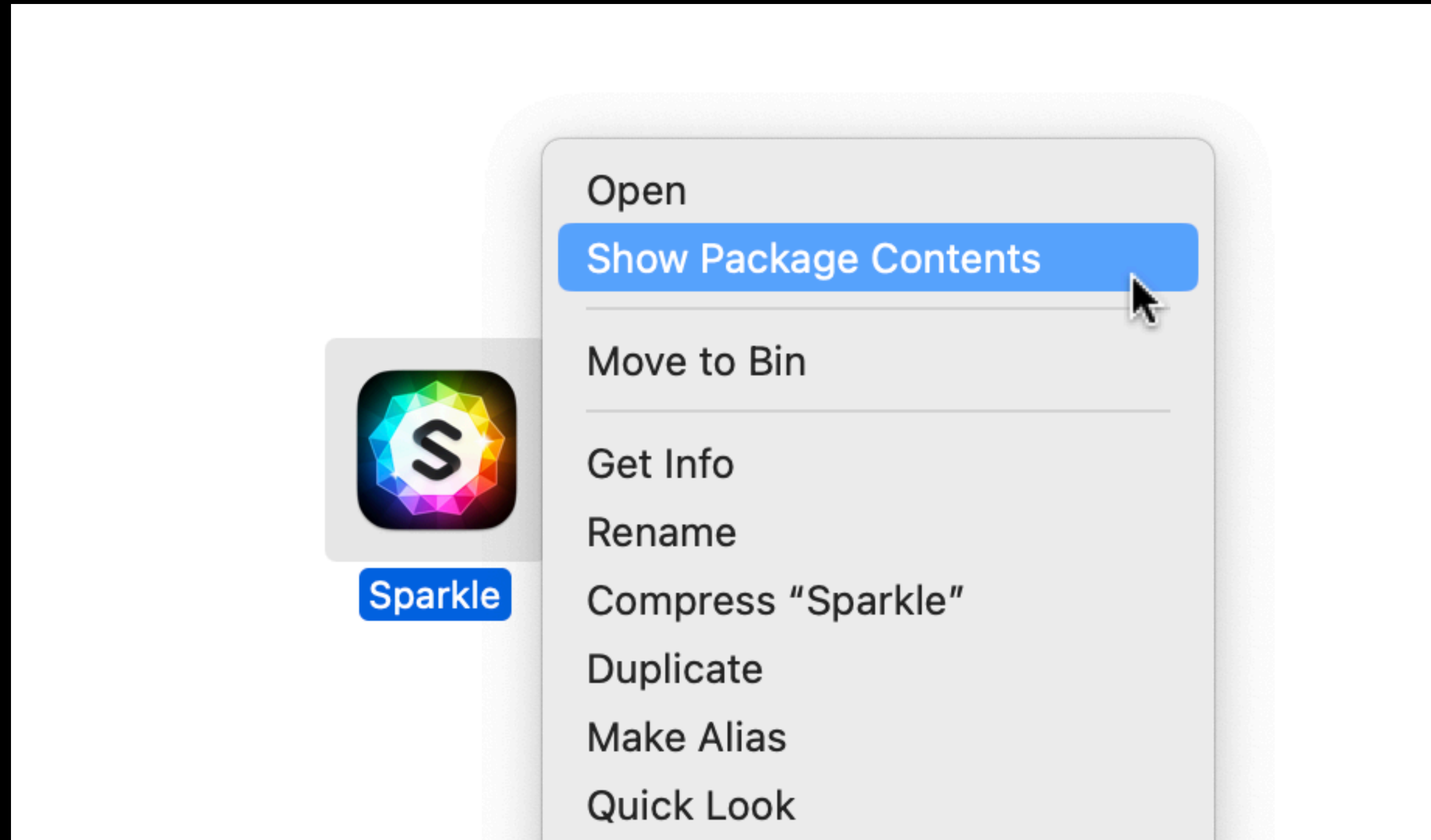
large website.sparkle

- 35 GB
- 300 pages
- 100k words
- 800 images
- 230 videos
- 400 attached files



Why not use a file bundle?

File bundle



File bundle issues

- Hard to share
- Backup issues
- Cloud sync issues

File bundle issues

- Hard to share
- Backup issues
- Cloud sync issues

Procreate Dreams:

“All your assets are bundled into one file made for iCloud synchronization and backups”

% unzip -v test.pages

Archive: test.pages

Length	Method	Size	Cmpr	Date	Time	CRC-32	Name
4749	Stored	4749	0%	03-06-2023	18:19	3431e399	Index/Document.iwa
543	Stored	543	0%	03-06-2023	18:19	fe3e0e70	Index/ViewState.iwa
296	Stored	296	0%	03-06-2023	18:19	b93f544b	Index/CalculationEngine-5037.iwa
22	Stored	22	0%	07-03-2023	15:44	22c83e08	Index/AnnotationAuthorStorage-5036.iwa
40230	Stored	40230	0%	03-06-2023	18:19	ef39583f	Index/DocumentStylesheet-5040.iwa
23	Stored	23	0%	07-03-2023	15:44	22979e87	Index/DocumentMetadata.iwa
11573	Stored	11573	0%	03-06-2023	18:19	19f46a4b	Index/Metadata.iwa
322	Stored	322	0%	03-06-2023	18:19	4ab82032	Metadata/Properties.plist
36	Stored	36	0%	03-06-2023	18:19	99ccdf24	Metadata/DocumentIdentifier
273	Stored	273	0%	03-06-2023	18:19	4e89ec53	Metadata/BuildVersionHistory.plist
13351	Stored	13351	0%	03-06-2023	18:19	2e1610ee	preview.jpg
829	Stored	829	0%	03-06-2023	18:19	3a4909c1	preview-micro.jpg
1425	Stored	1425	0%	03-06-2023	18:19	b4fd7f37	preview-web.jpg
73672		73672	0%				13 files

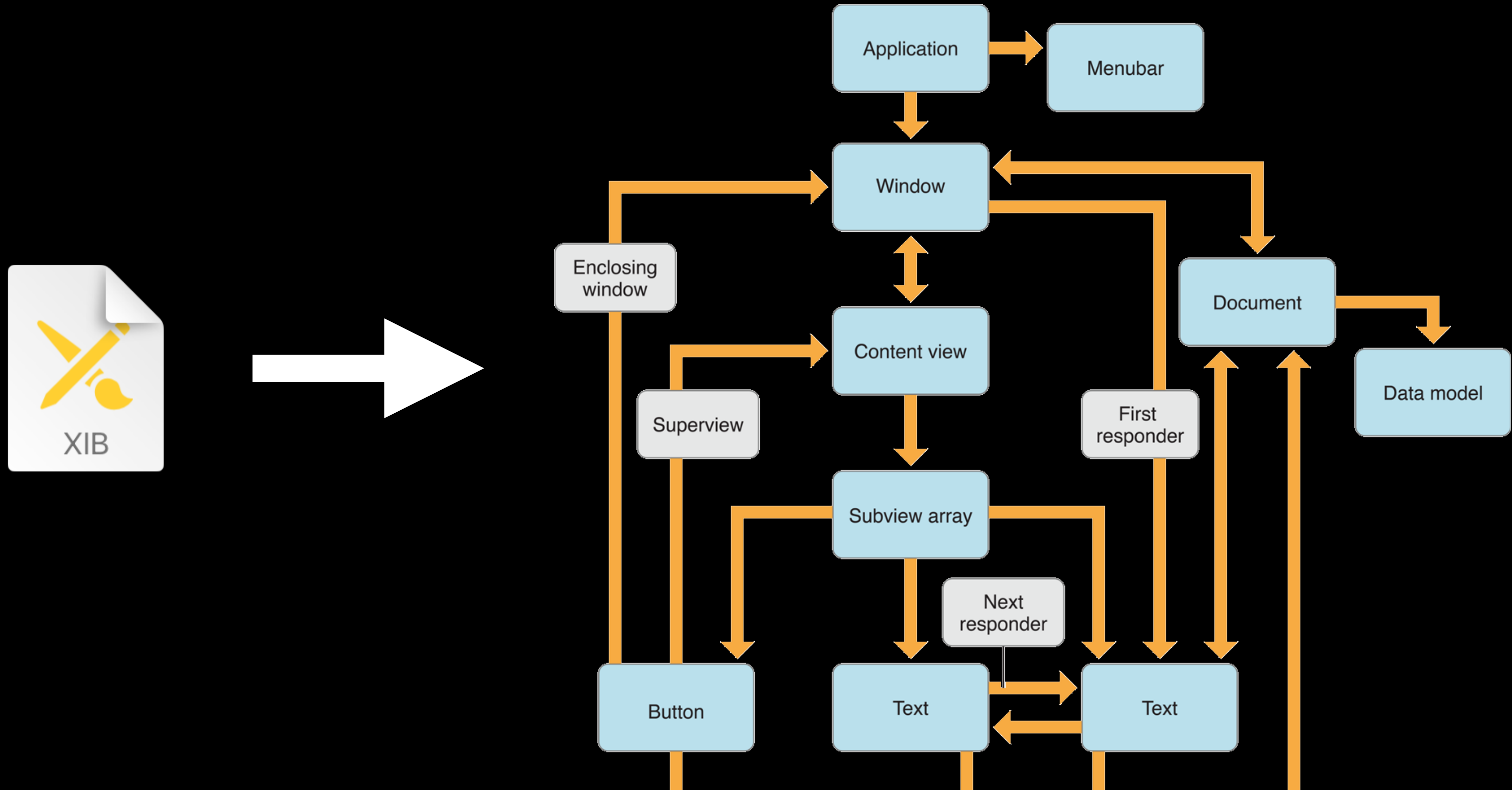
Why not use Core Data?

Improving load time

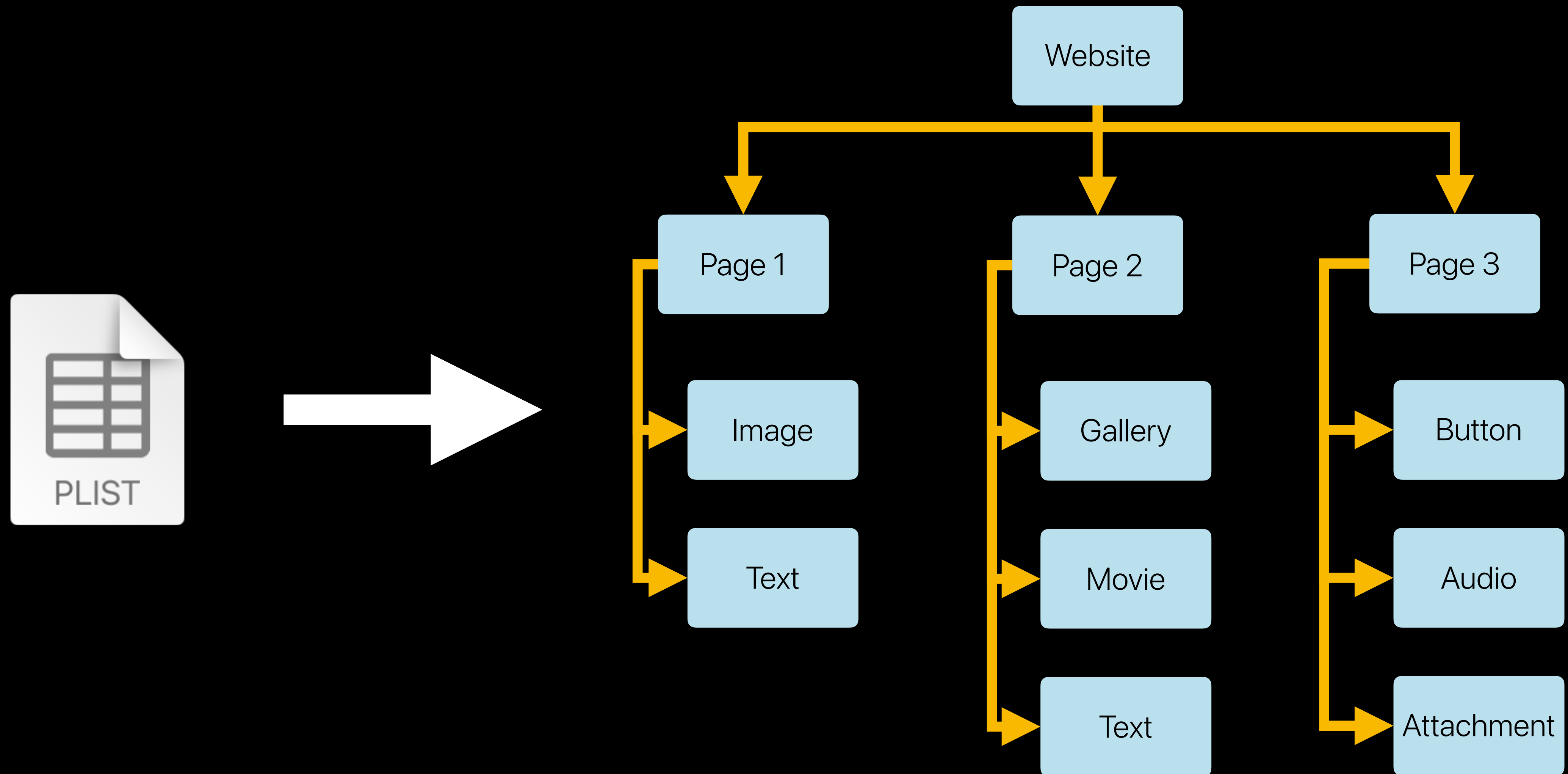
Improving load time

- Don't do unnecessary work

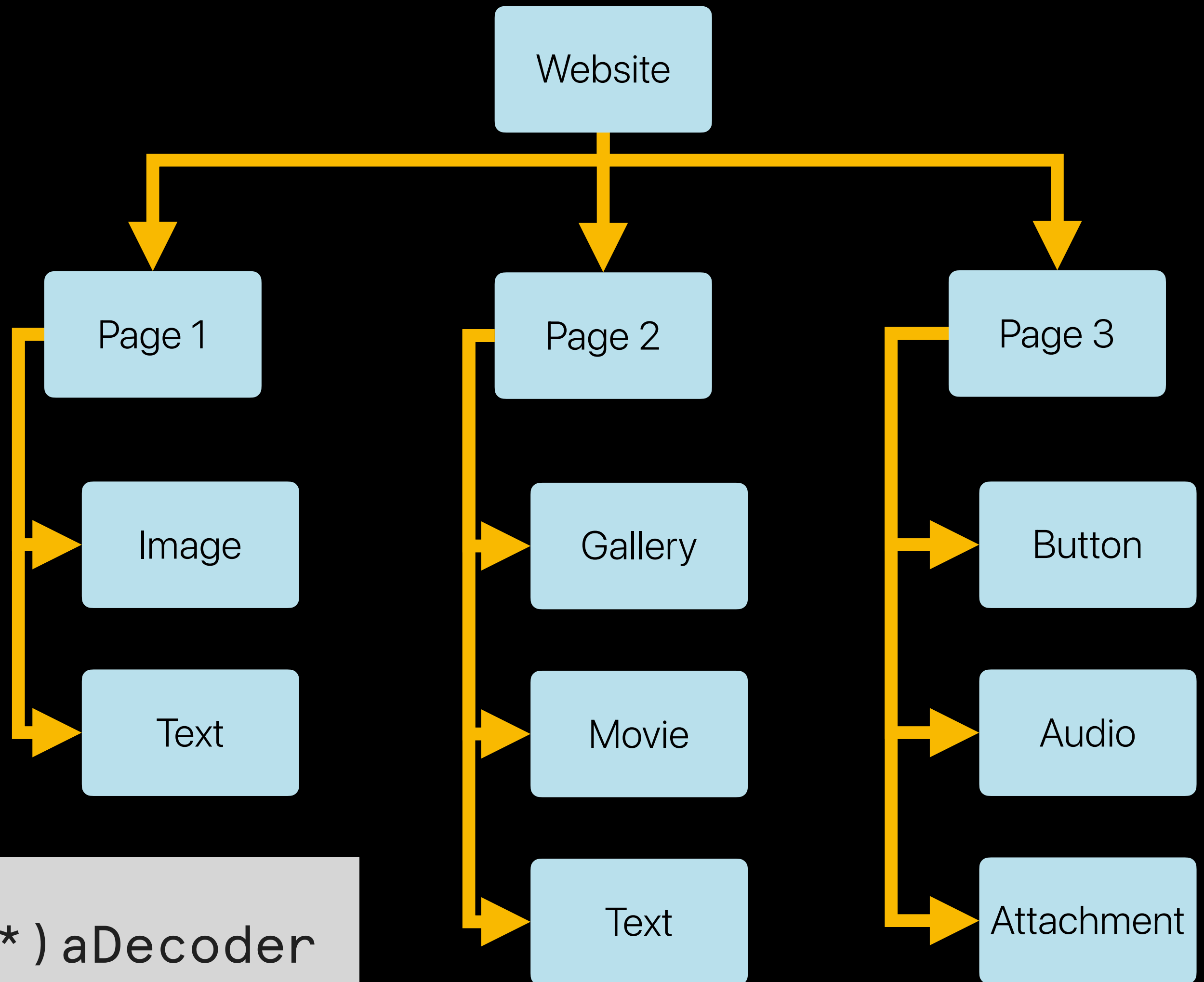
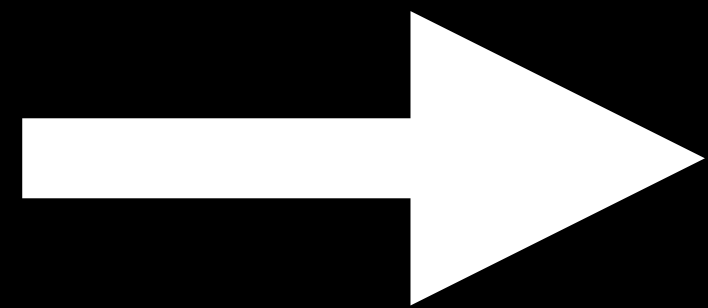
Improving load time



Improving load time



Improving load time



```
- (id)initWithCoder:(NSCoder *) aDecoder
```

Improving load time

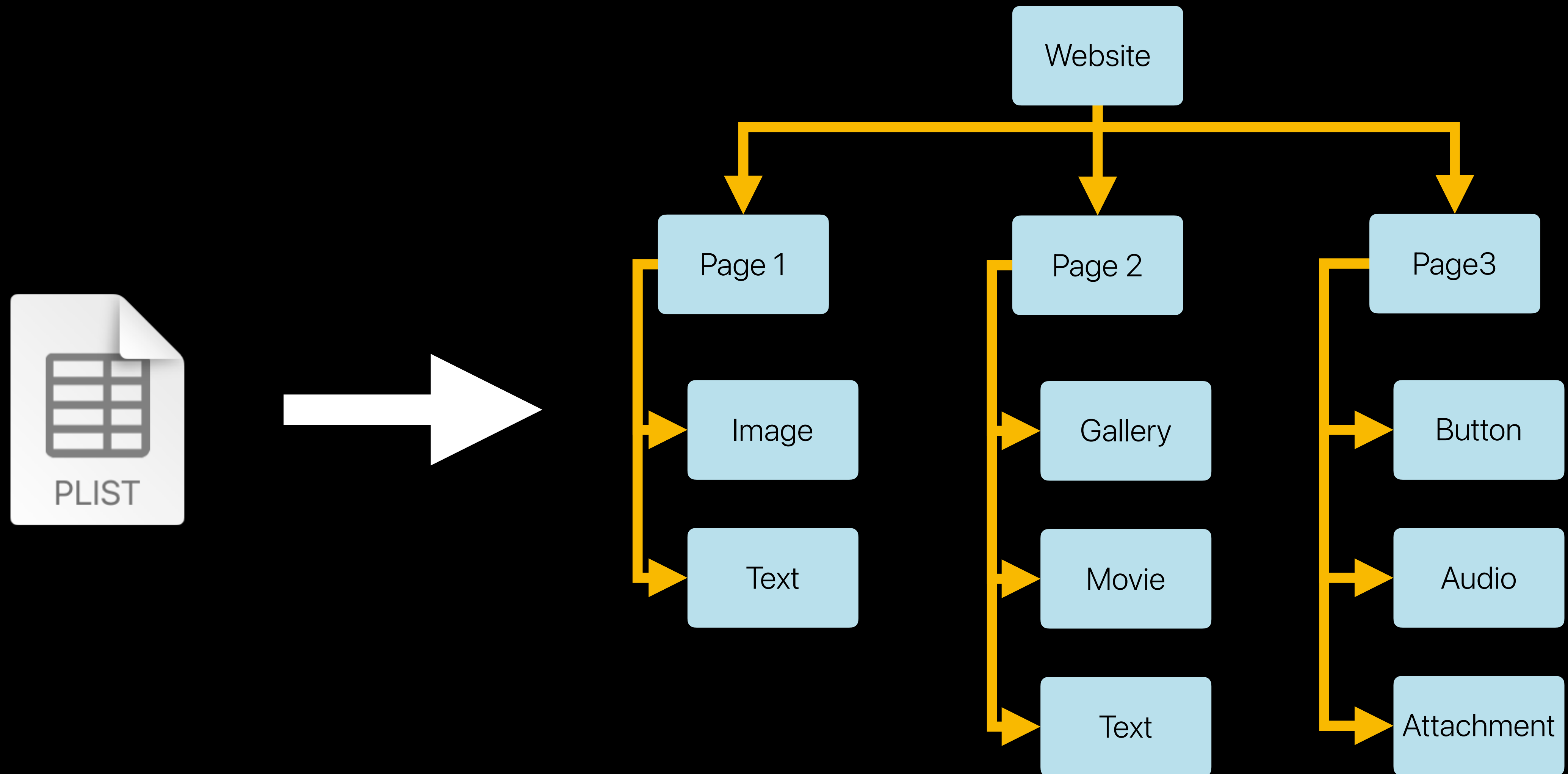
Improving load time

- Don't do unnecessary work

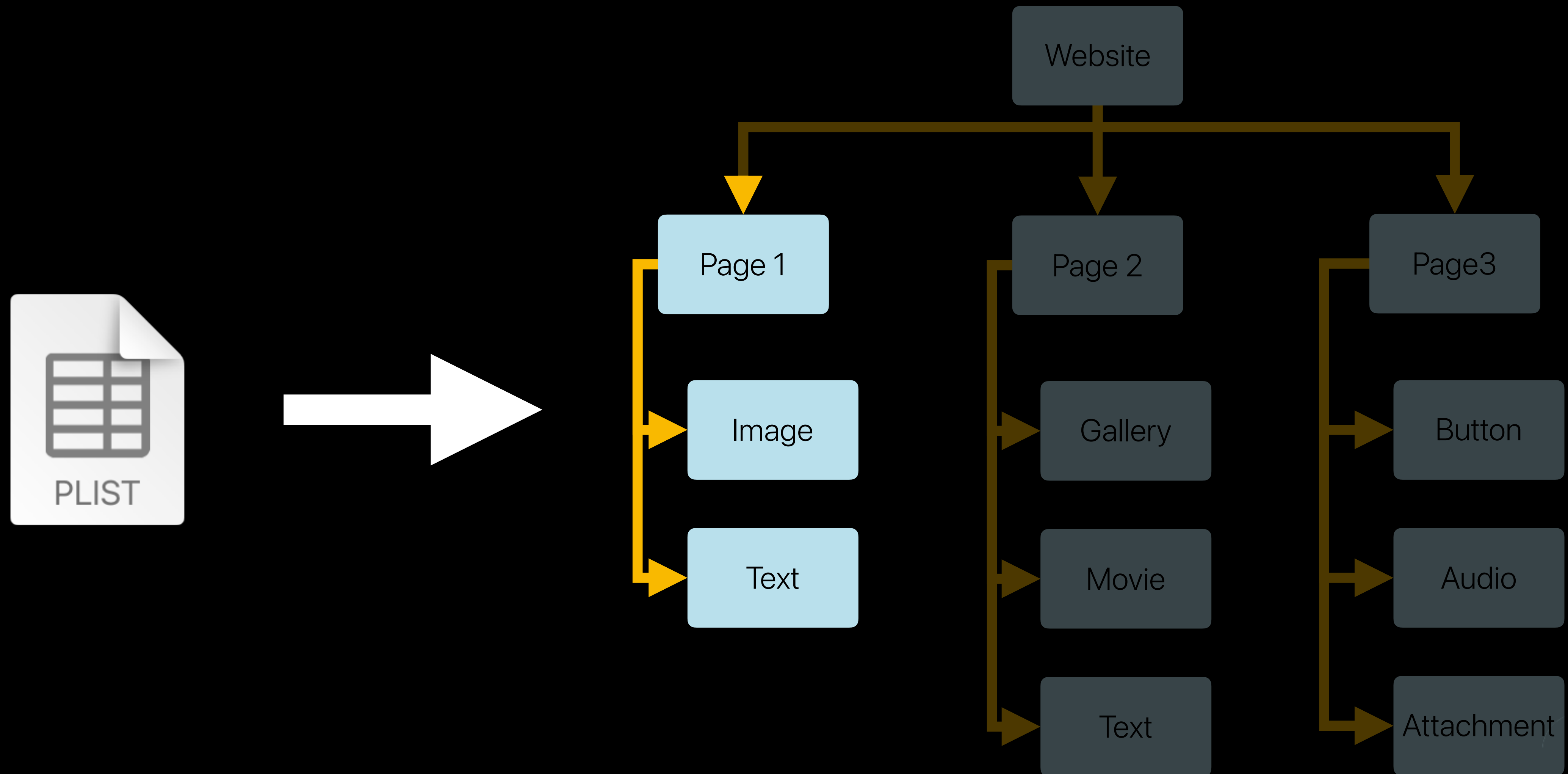
Improving load time

- Don't do unnecessary work
- Only load objects that are used

Improving load time



Improving load time



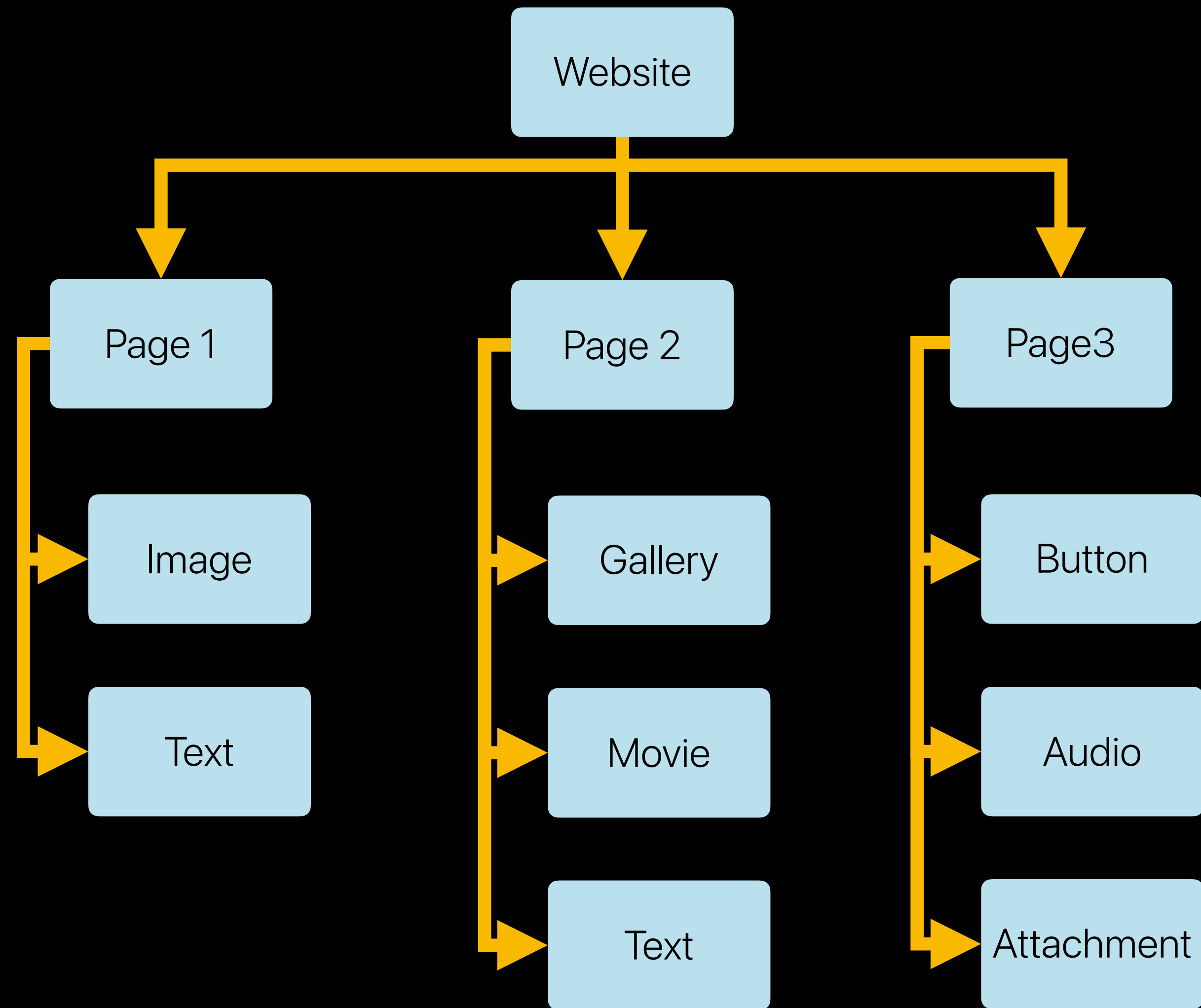
Improving load time

- Don't do unnecessary work
- Only load objects that are used

Improving load time

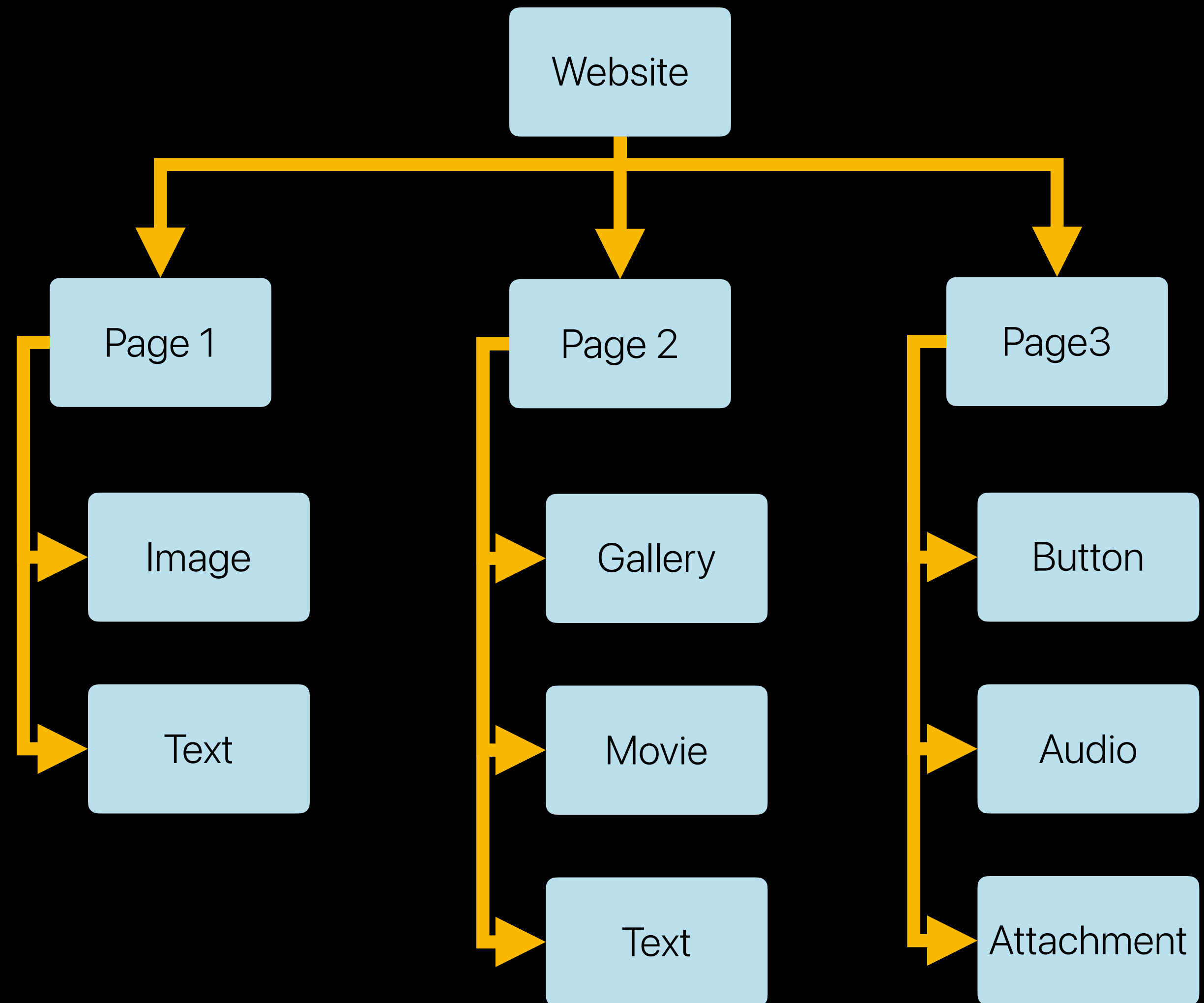
- Don't do unnecessary work
- Only load objects that are used
- Lazy load objects, on demand

Improving load time



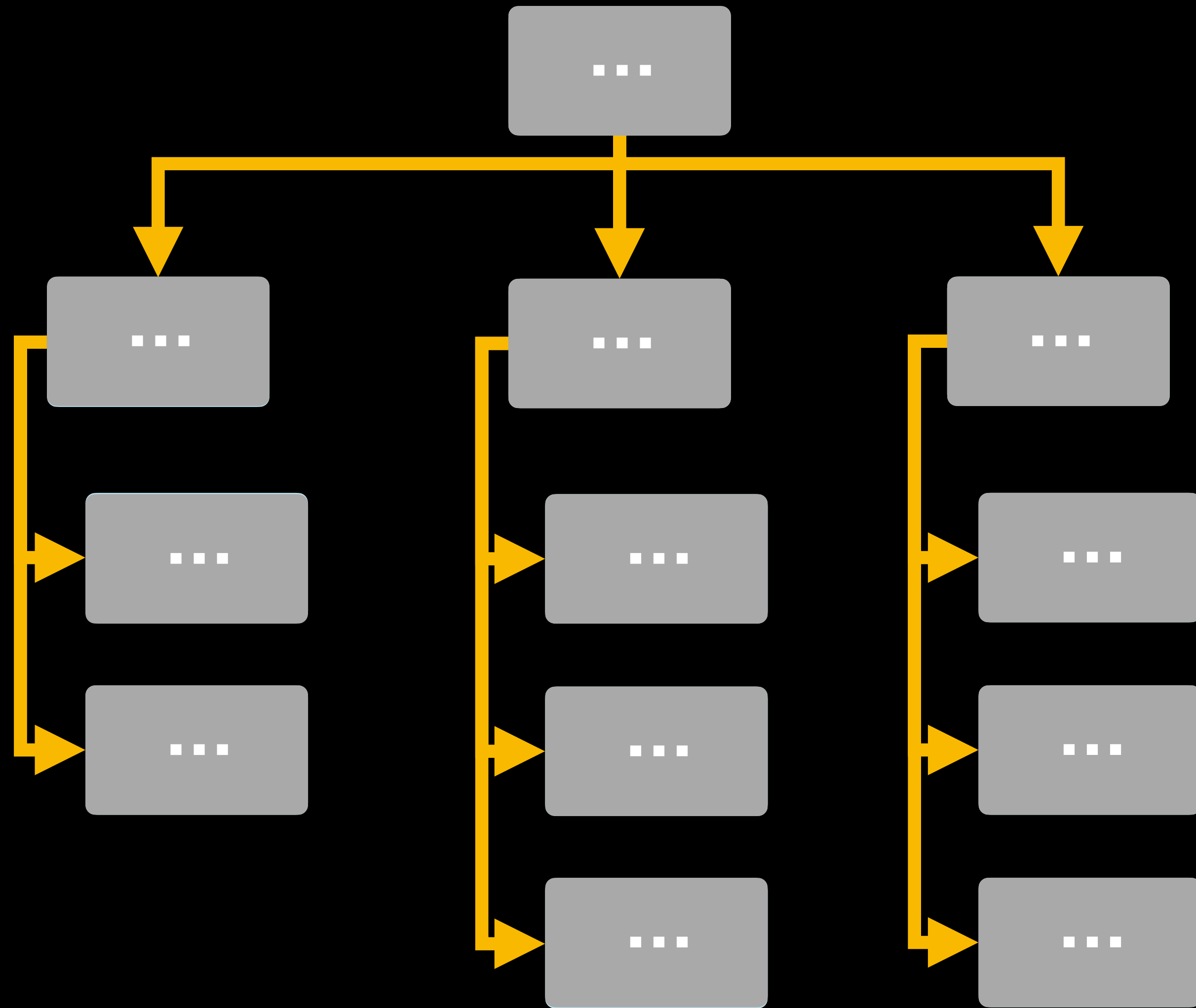
Improving load time

Object Store



Improving load time

Object Store



Improving load time

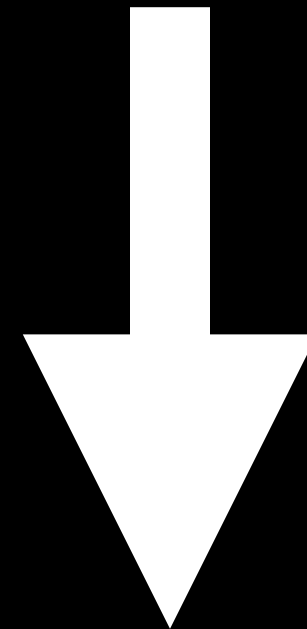
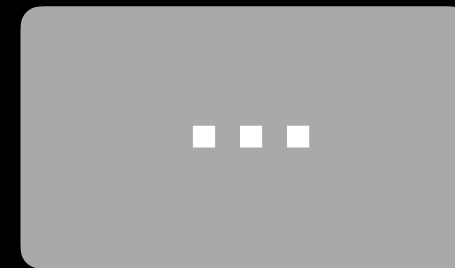
```
// Apple:  
// NSObject is an abstract superclass defining an API  
// for objects that act as stand-ins for other objects  
// or for objects that don't exist yet.
```

```
@interface LazyLoaderProxy : NSObject
```

```
...
```

```
@end
```

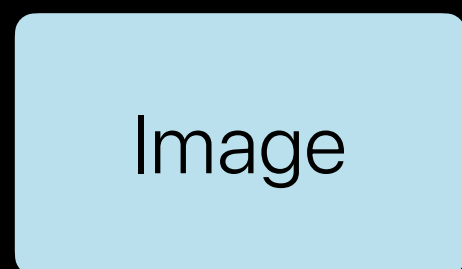
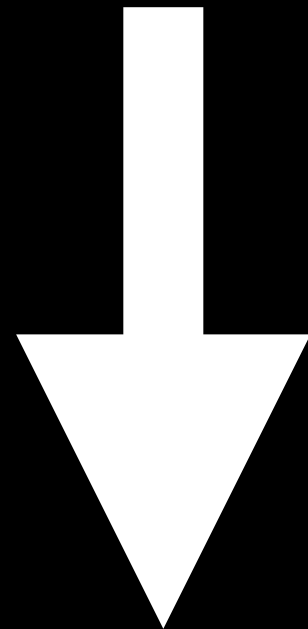
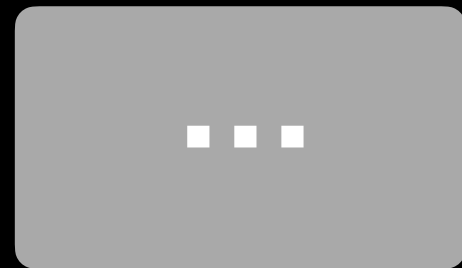
Improving load time



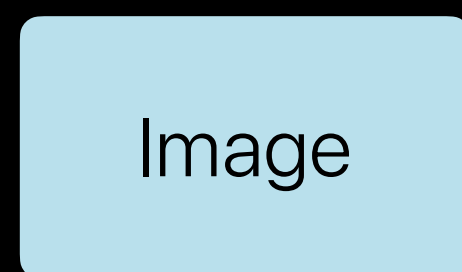
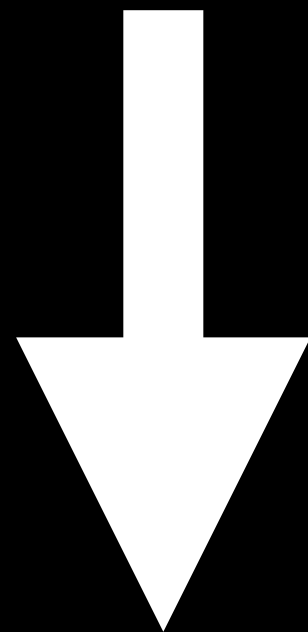
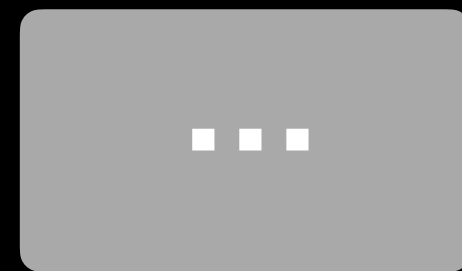
???



Improving load time

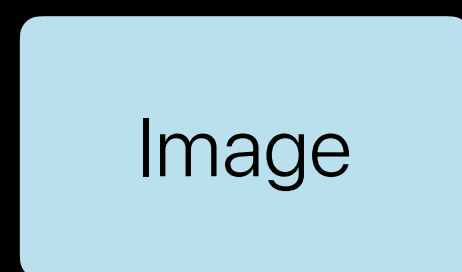
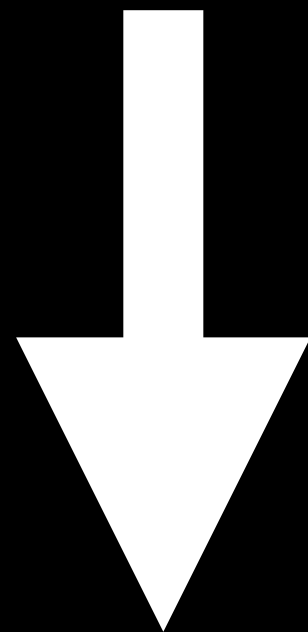
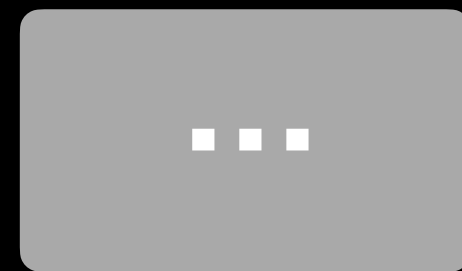


Improving load time



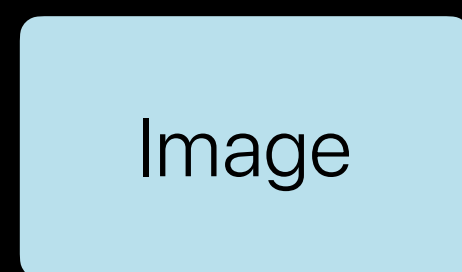
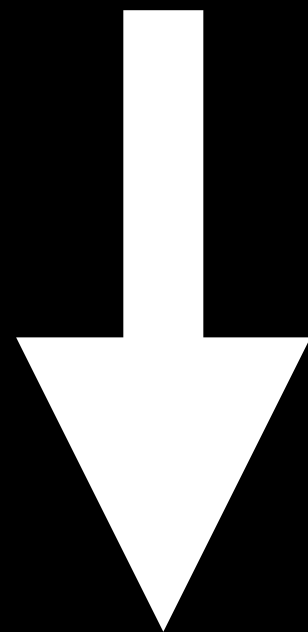
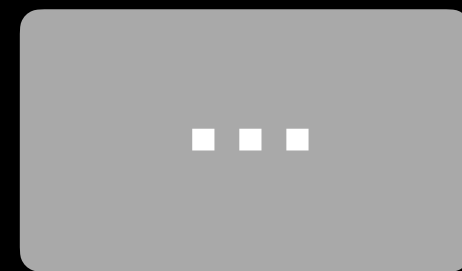
- Instantiate NSProxy

Improving load time



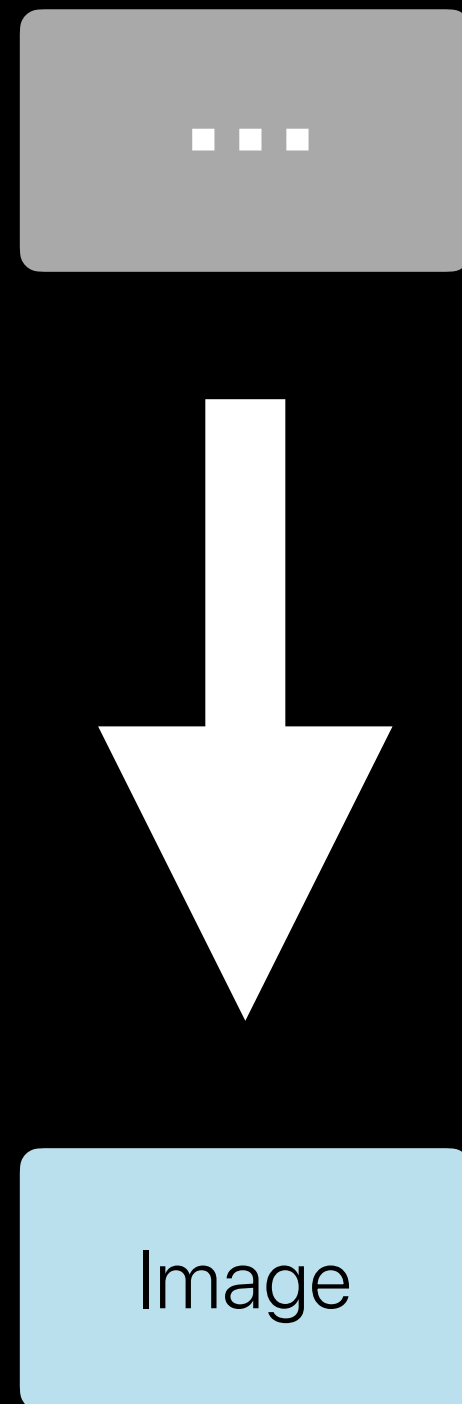
- Instantiate NSProxy
- Detect use of object

Improving load time



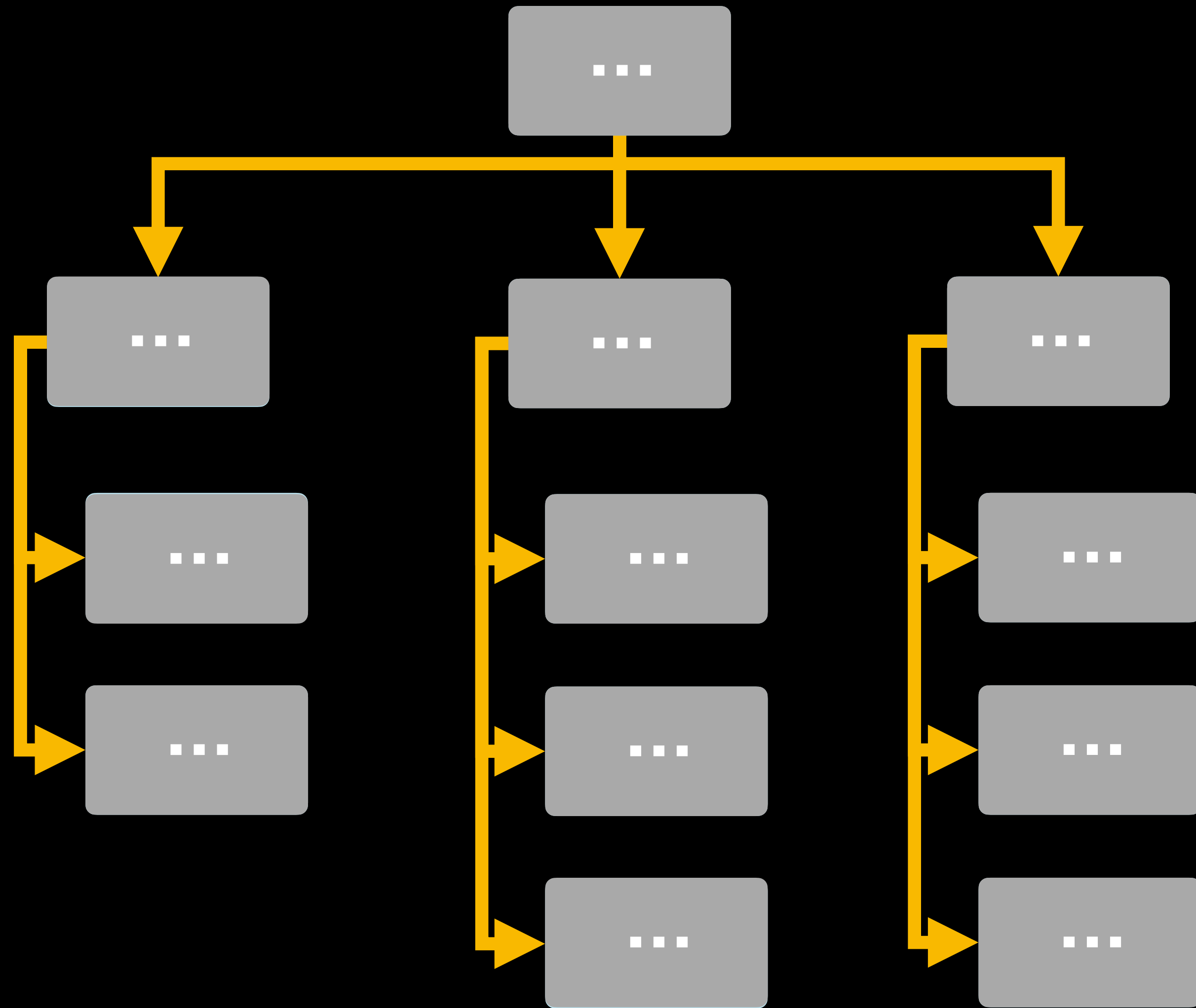
- Instantiate NSProxy
- Detect use of object
- Transform proxy instance into object

Improving load time

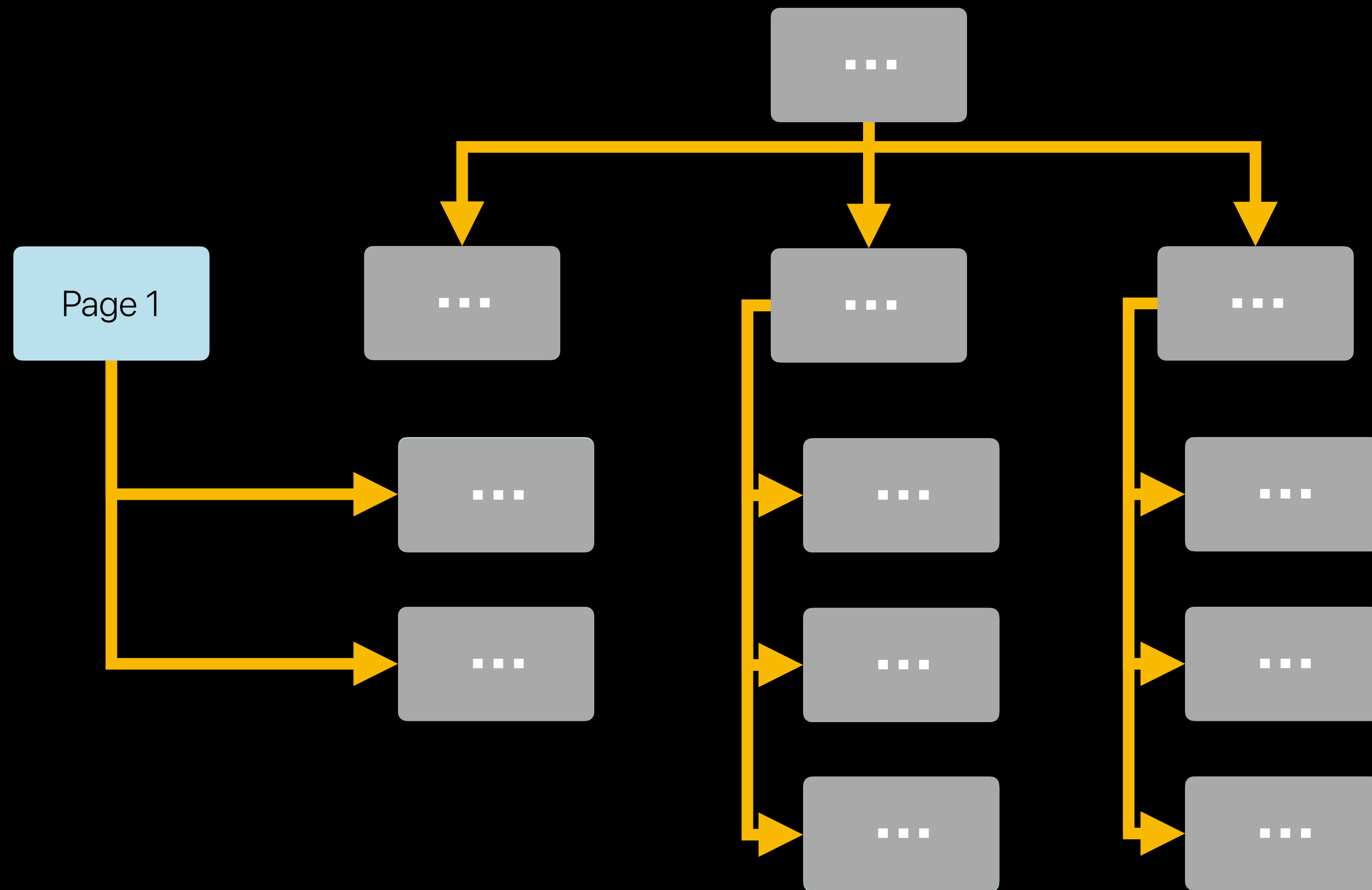


- Instantiate NSProxy
- Detect use of object
- Transform proxy instance into object
- Dearchive object

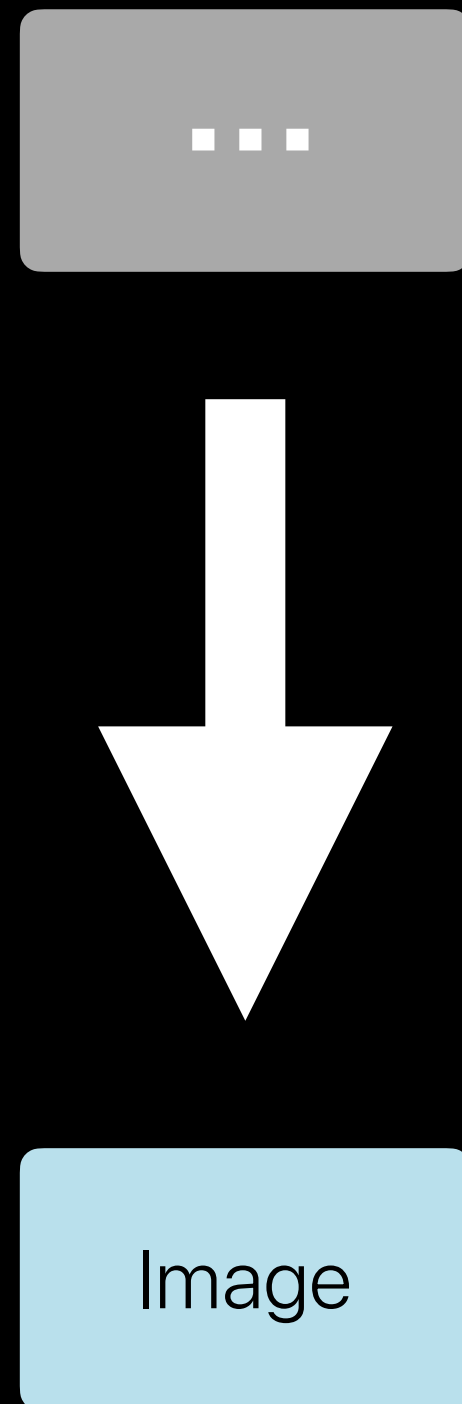
Improving load time



Improving load time



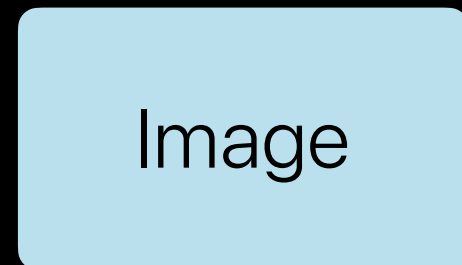
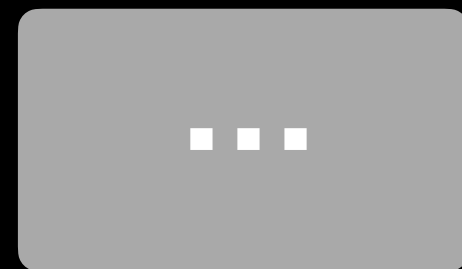
Improving load time



- Instantiate NSProxy
- Detect use of object
- Transform proxy instance into object
- Dearchive object

Improving load time

Memory layout

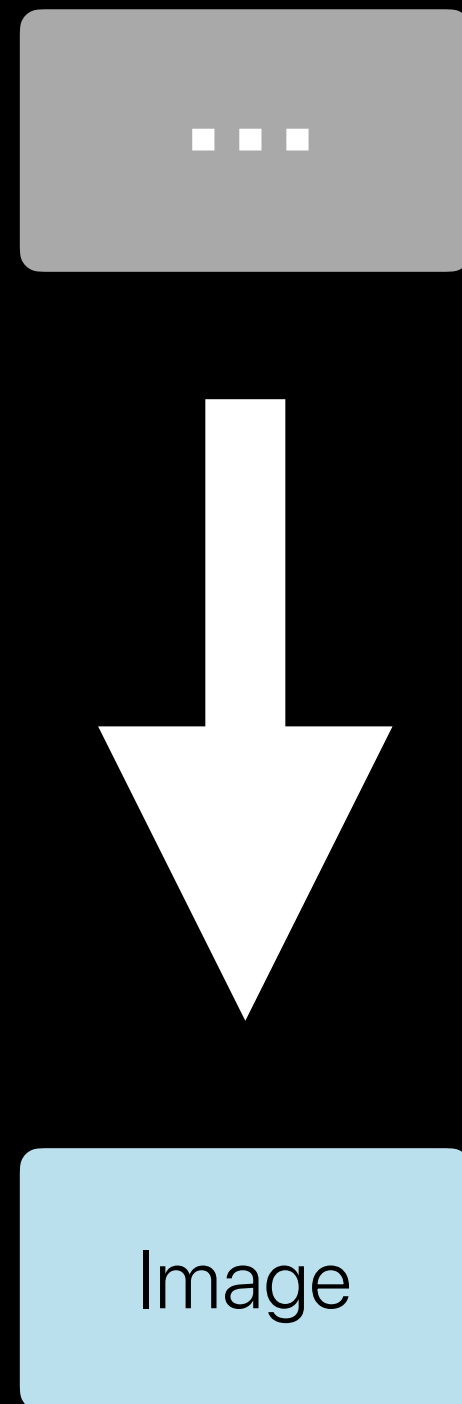


Improving load time

Memory layout

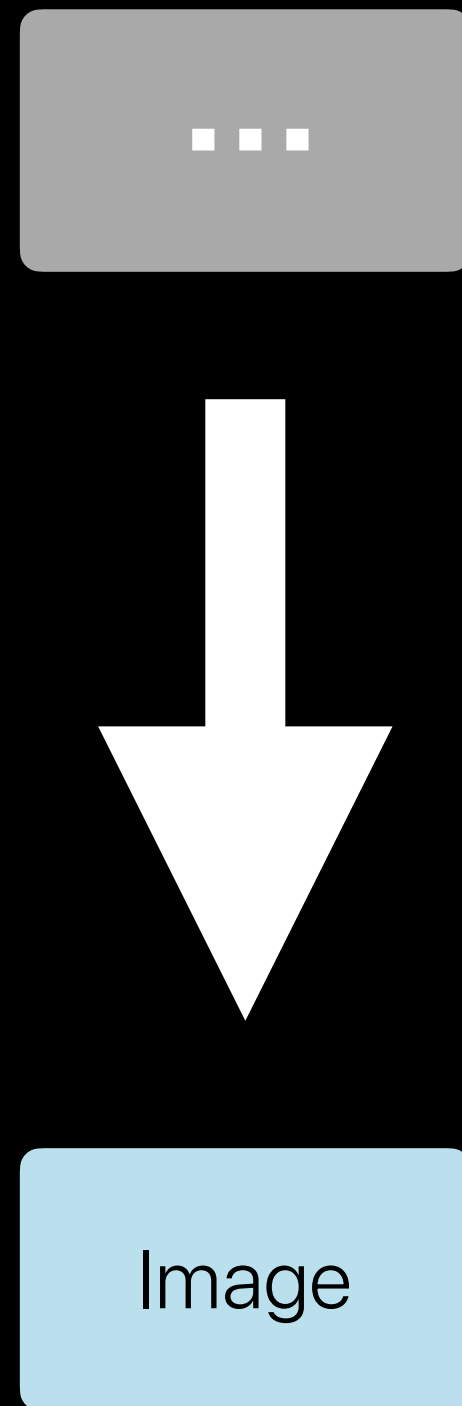


Improving load time



- Instantiate NSProxy
- Detect use of object
- Transform proxy instance into object
- Dearchive object

Improving load time



- Measure object sizes
- Instantiate NSProxy with custom size
- Detect use of object
- Transform proxy instance into object
- Dearchive object

Improving load time

```
NSInteger size1 = class_getInstanceSize(targetObjectClass);
NSInteger size2 = class_getInstanceSize(LazyLoaderProxy.class);
NSInteger allocsize = MAX(size1, size2);

// allocate object memory
void *bytes = calloc(1, allocsize);

// non-ARC code
id obj = objc_constructInstance(LazyLoaderProxy.class, bytes);
```

Improving load time

```
NSInteger size1 = class_getInstanceSize(targetObjectClass);  
NSInteger size2 = class_getInstanceSize(LazyLoaderProxy.class);  
NSInteger allocsize = MAX(size1, size2);
```

```
// allocate object memory  
void *bytes = calloc(1, allocsize);
```

```
// non-ARC code  
id obj = objc_constructInstance(LazyLoaderProxy.class, bytes);
```

Improving load time

```
NSInteger size1 = class_getInstanceSize(targetObjectClass);  
NSInteger size2 = class_getInstanceSize(LazyLoaderProxy.class);  
NSInteger allocsize = MAX(size1, size2);
```

```
// allocate object memory  
void *bytes = calloc(1, allocsize);
```

```
// non-ARC code  
id obj = objc_constructInstance(LazyLoaderProxy.class, bytes);
```

Improving load time

```
NSInteger size1 = class_getInstanceSize(targetObjectClass);
NSInteger size2 = class_getInstanceSize(LazyLoaderProxy.class);
NSInteger allocsize = MAX(size1, size2);

// allocate object memory
void *bytes = calloc(1, allocsize);

// non-ARC code
id obj = objc_constructInstance(LazyLoaderProxy.class, bytes);
```

Improving load time

```
- (NSMethodSignature *)methodSignatureForSelector:(SEL) sel
{
    Method m = class_getInstanceMethod(targetObjectClass, sel);
    return [NSMethodSignature
            signatureWithObjCTypes:method_getTypeEncoding(m)];
}

- (void)forwardInvocation:(NSInvocation *)invocation
{
    // do something
}
```


Improving load time

```
- (NSMethodSignature *)methodSignatureForSelector:(SEL) sel
{
    Method m = class_getInstanceMethod(targetObjectClass, sel);
    return [NSMethodSignature
            signatureWithObjCTypes:method_getTypeEncoding(m)];
}
```

```
- (void)forwardInvocation:(NSInvocation *)invocation
{
    // do something
}
```

Improving load time

```
- (NSMethodSignature *)methodSignatureForSelector:(SEL) sel
{
    Method m = class_getInstanceMethod(targetObjectClass, sel);
    return [NSMethodSignature
            signatureWithObjCTypes:method_getTypeEncoding(m)];
}
```

```
- (void)forwardInvocation:(NSInvocation *)invocation
{
    // do something
}
```

Improving load time

```
+ (BOOL)loadObject:(LazyLoaderProxy *)proxy
{
    memset((void *)proxy, 0, allocsize);
    object_setClass(proxy, targetObjectClass); // non-ARC code
    [our_object_store instantiateIntoObject:proxy];
}

- (void)forwardInvocation:(NSInvocation *)invocation
{
    [LazyLoaderProxy loadObject:self];
    [invocation invokeWithTarget:self];
}
```

Improving load time

```
+ (BOOL)loadObject:(LazyLoaderProxy *)proxy
{
    memset((void *)proxy, 0, allocsize);
    object_setClass(proxy, targetObjectClass); // non-ARC code
    [our_object_store instantiateIntoObject:proxy];
}
```

```
- (void)forwardInvocation:(NSInvocation *)invocation
{
    [LazyLoaderProxy loadObject:self];
    [invocation invokeWithTarget:self];
}
```

Improving load time

```
+ (BOOL)loadObject:(LazyLoaderProxy *)proxy
{
    memset((void *)proxy, 0, alloca_size);
    object_setClass(proxy, targetObjectClass); // non-ARC code
    [our_object_store instantiateIntoObject:proxy];
}
```

```
- (void)forwardInvocation:(NSInvocation *)invocation
{
    [LazyLoaderProxy loadObject:self];
    [invocation invokeWithTarget:self];
}
```

Improving load time



large website.sparkle

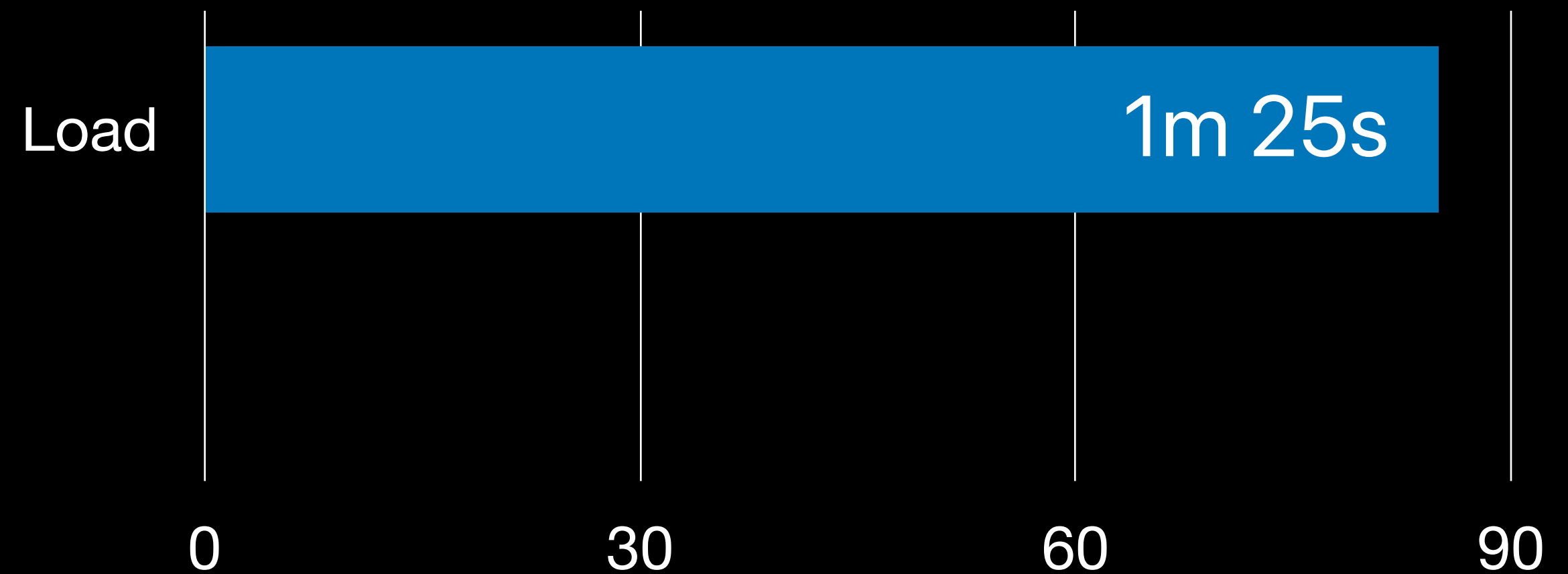
- 35 GB
- 300 pages
- 100k words
- 800 images
- 230 videos
- 400 attached files

Improving load time



large website.sparkle

- 35 GB
- 300 pages
- 100k words
- 800 images
- 230 videos
- 400 attached files

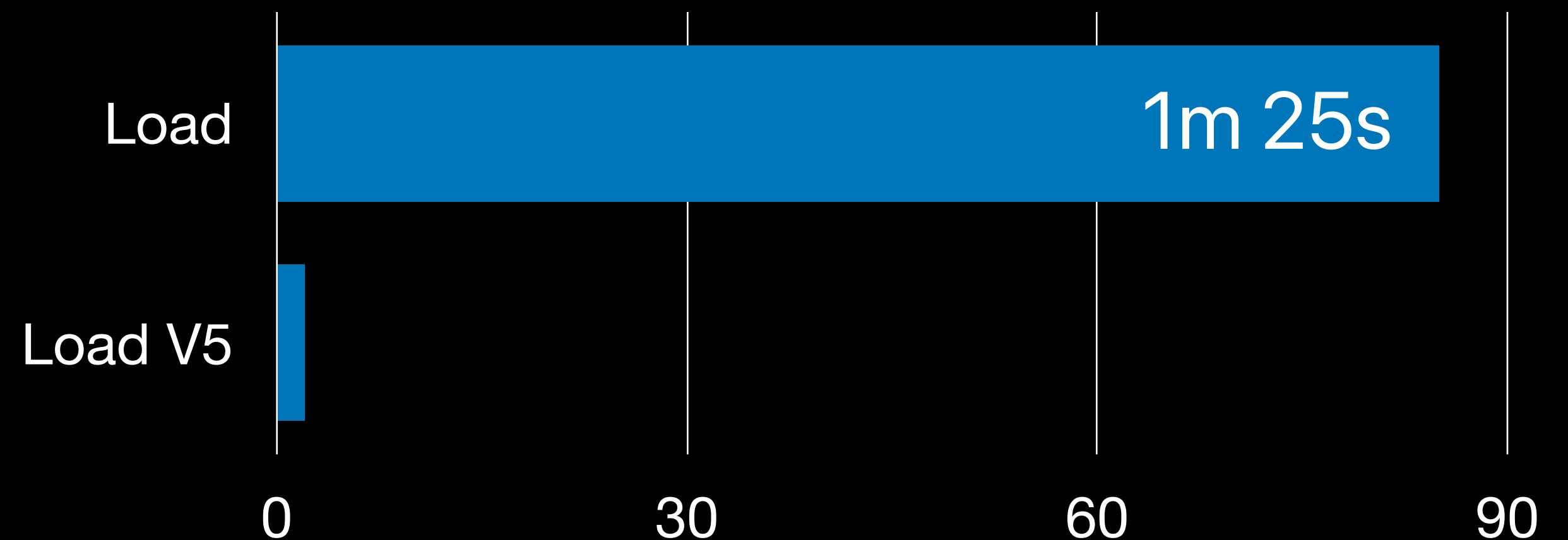


Improving load time



large website.sparkle

- 35 GB
- 300 pages
- 100k words
- 800 images
- 230 videos
- 400 attached files

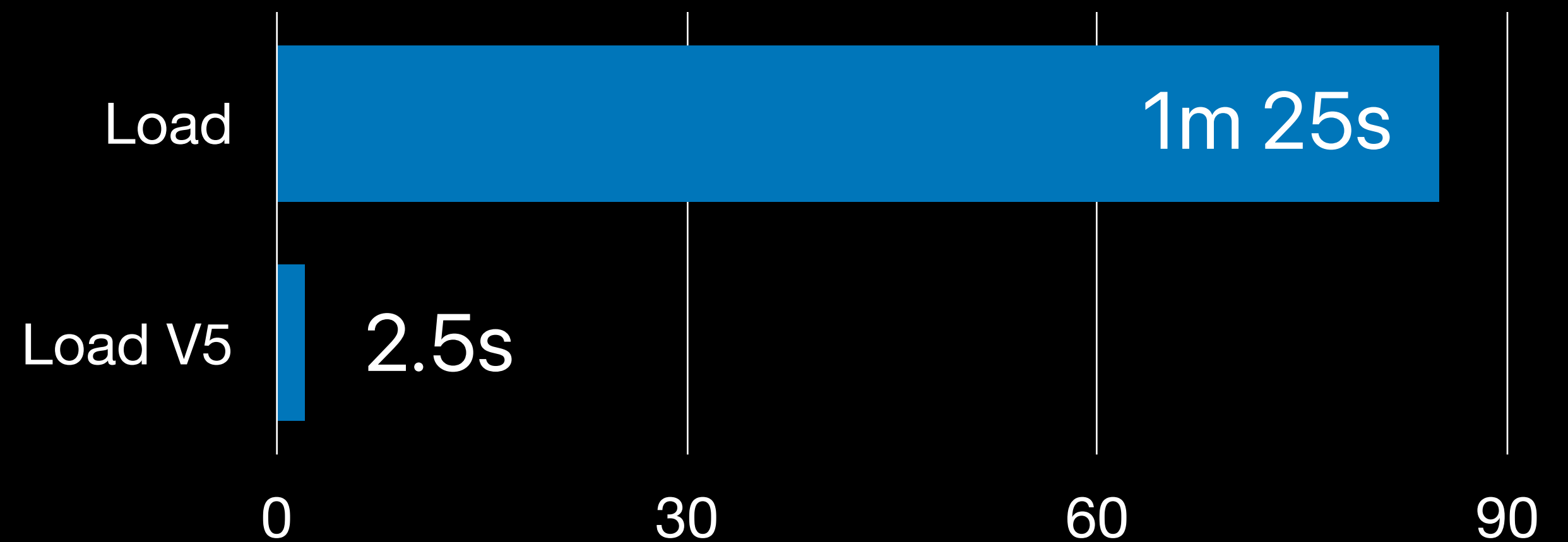


Improving load time



large website.sparkle

- 35 GB
- 300 pages
- 100k words
- 800 images
- 230 videos
- 400 attached files



Improving load time

Improving load time

Pitfalls:

Improving load time

Pitfalls:

- Reinventing the wheel

Improving load time

Pitfalls:

- Reinventing the wheel
- Custom dearchive slower

Improving load time

Pitfalls:

- Reinventing the wheel
- Custom dearchive slower
- Lazy load != never load

Improving load time

Pitfalls:

- Reinventing the wheel
- Custom dearchive slower
- Lazy load != never load
- Trouble with post-load side effects

Improving load time

Pitfalls:

- Reinventing the wheel
- Custom dearchive slower
- Lazy load != never load
- Trouble with post-load side effects
- Not loading = not using memory

Objective-C

Objective-C

- Lazy loading not possible without Objective-C's dynamic runtime

Objective-C

- Lazy loading not possible without Objective-C's dynamic runtime
- Swift? We would have looked for a different solution

Objective-C

- Lazy loading not possible without Objective-C's dynamic runtime
- Swift? We would have looked for a different solution
- There never was a retain or a release in Sparkle

Improving save time

Improving save time

- Don't do unnecessary work

Improving save time

```
% time cp "large website.sparkle" copy.sparkle  
cp "large website.sparkle" copy.sparkle 0.02s user 14.83s system  
19% cpu 1:17.87 total
```

Improving save time

```
% time cp "large website.sparkle" copy.sparkle  
cp "large website.sparkle" copy.sparkle 0.02s user 14.83s system  
19% cpu 1:17.87 total
```


Improving save time

```
% time cp -c "large website.sparkle" clone.sparkle  
cp -c "large website.sparkle" clone.sparkle 0.00s user 0.00s  
system 57% cpu 0.007 total
```

Improving save time

```
% time cp -c "large website.sparkle" clone.sparkle  
cp -c "large website.sparkle" clone.sparkle 0.00s user 0.00s  
system 57% cpu 0.007 total
```

Improving save time

```
cpu 0.007 total
```

Copy

large website.sparkle

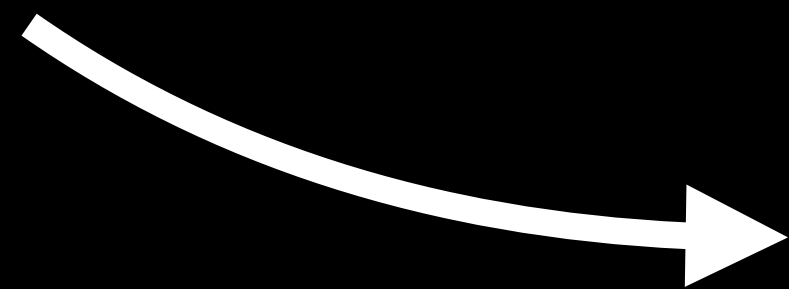


Copy

large website.sparkle

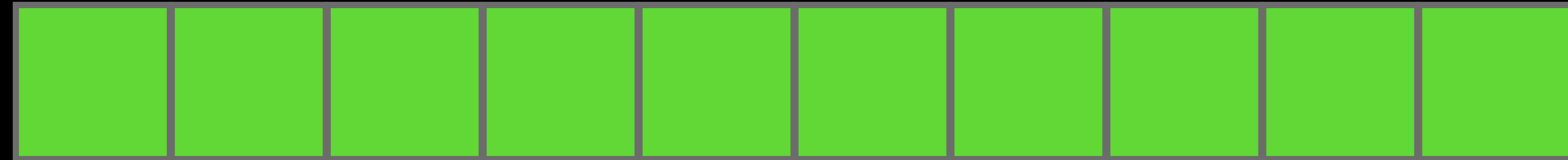


copy.sparkle



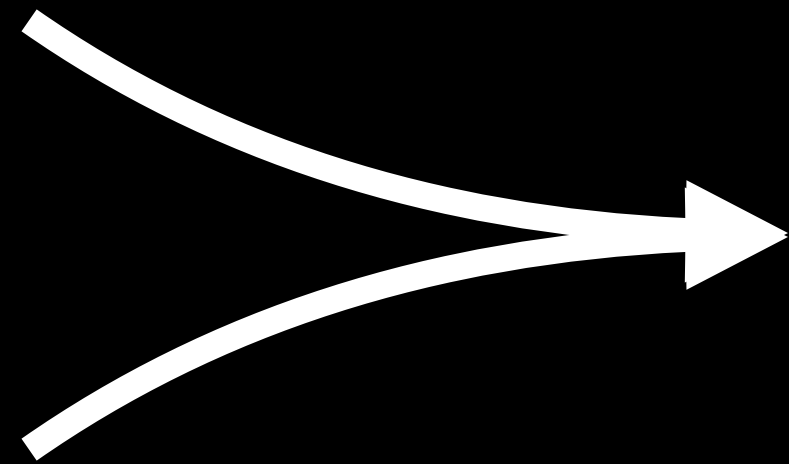
Copy on Write

large website.sparkle

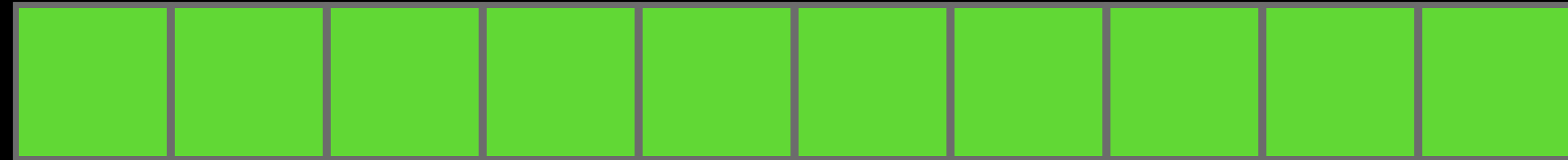


Copy on Write

large website.sparkle

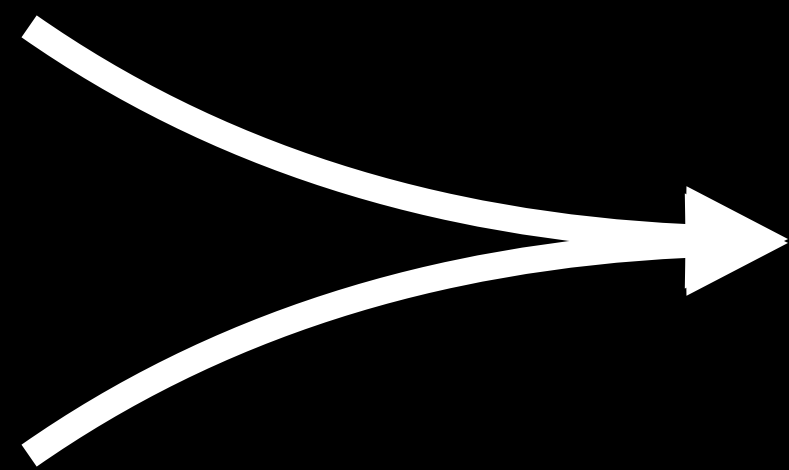


clone.sparkle

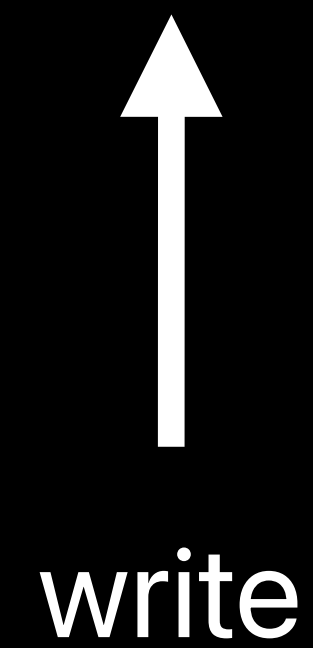


Copy on Write

large website.sparkle



clone.sparkle

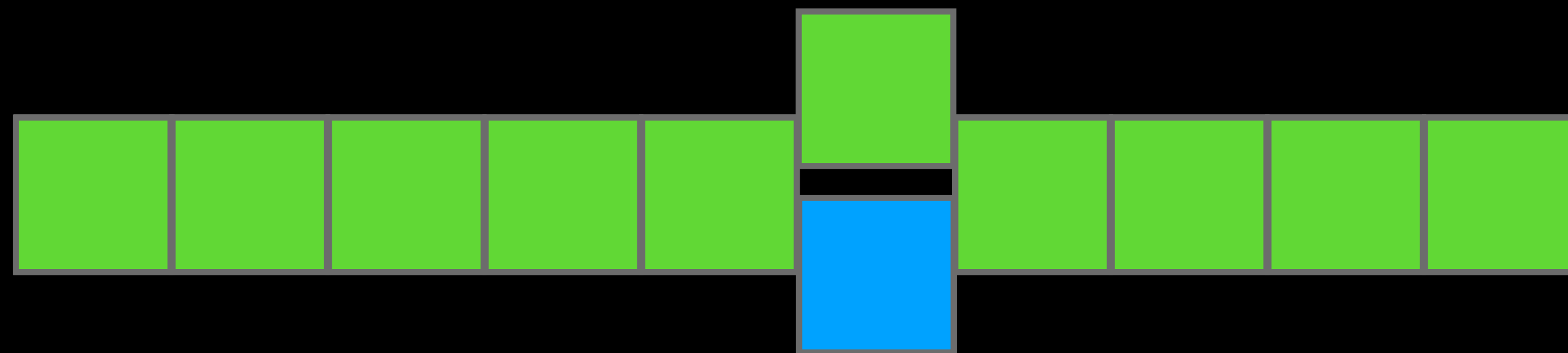
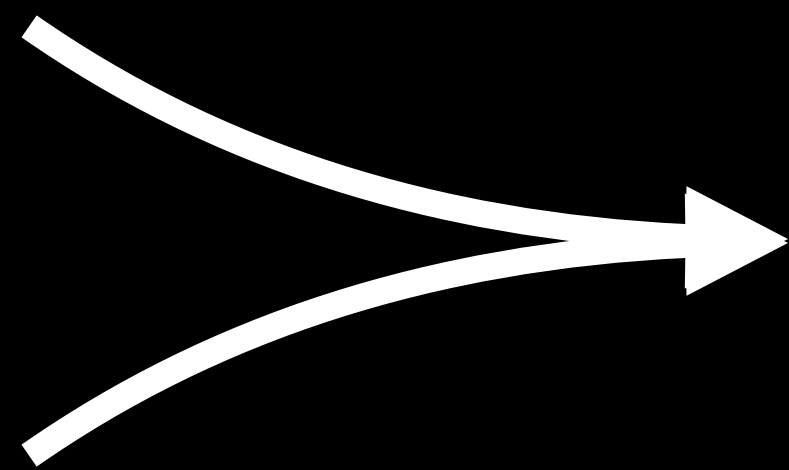


write

Copy on Write

large website.sparkle

clone.sparkle



write

Improving save time

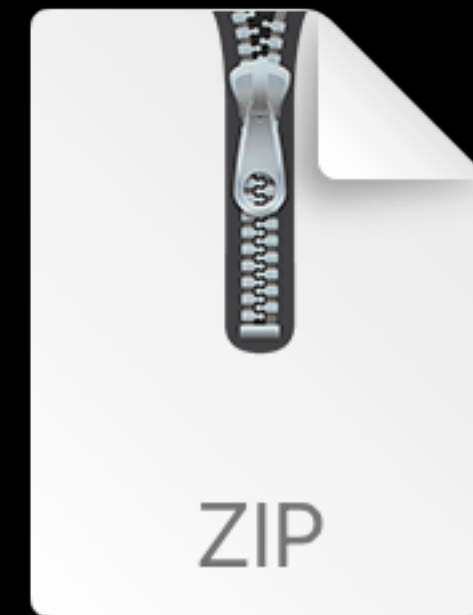
Improving save time



Improving save time



Improving save time



Improving save time



Improving save time



SQLITE

Improving save time



SQLITE

Improving save time



website.sparkle

Improving save time



website.sparkle

Improving save time

Improving save time

Plan:

Improving save time

Plan:

- Clone file

Improving save time

Plan:

- Clone file
- Go through all objects

Improving save time

Plan:

- Clone file
- Go through all objects
 - Archive + save object

Improving save time

Plan:

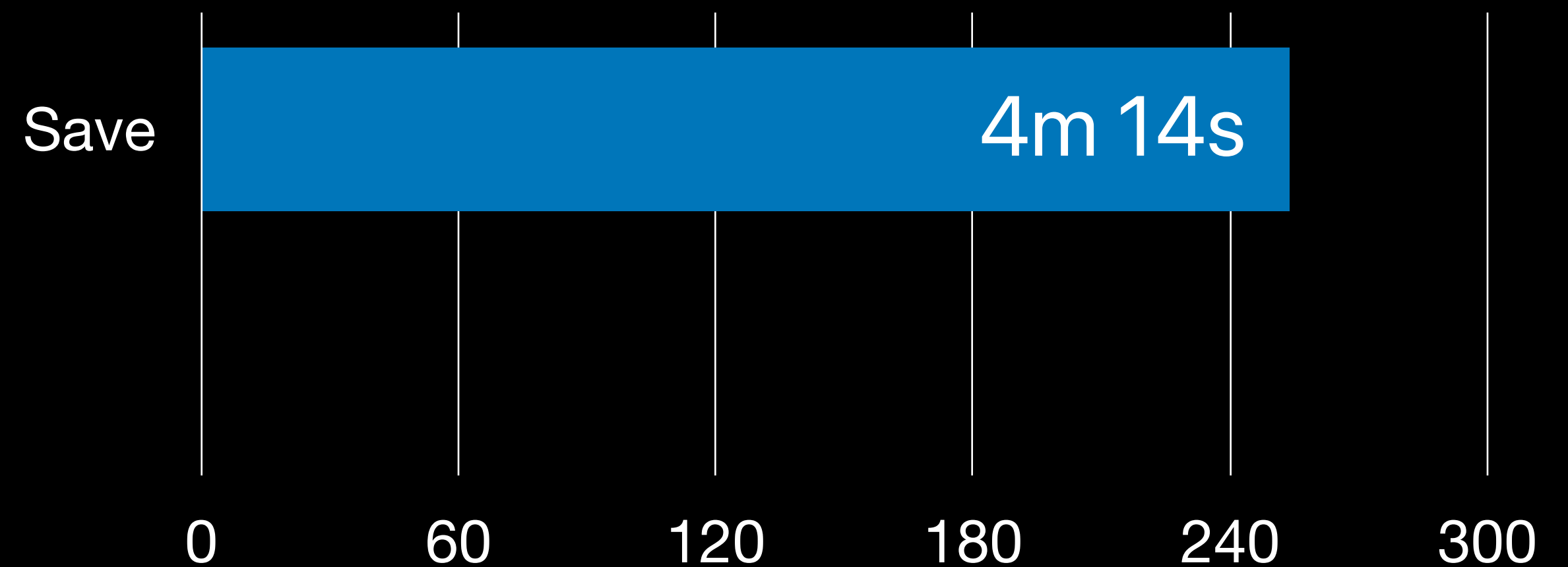
- Clone file
- Go through all objects
 - Archive + save object
- Remove old objects from the clone

Improving save time



large website.sparkle

- 35 GB
- 300 pages
- 100k words
- 800 images
- 230 videos
- 400 attached files

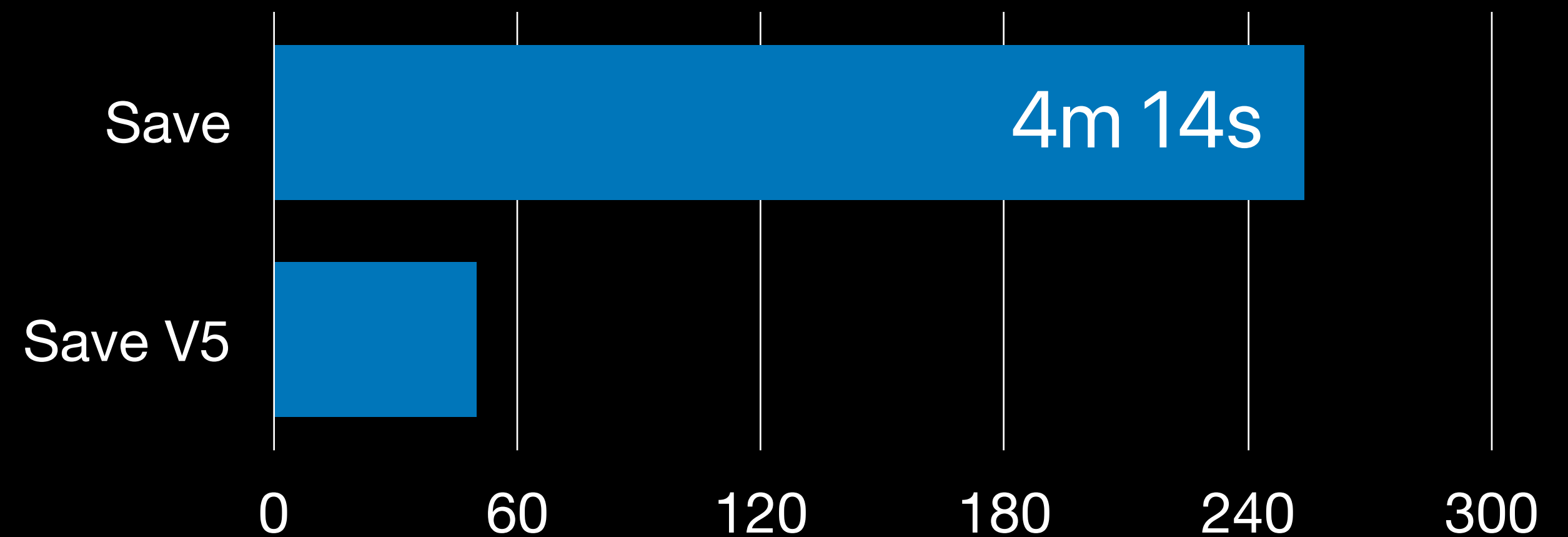


Improving save time



large website.sparkle

- 35 GB
- 300 pages
- 100k words
- 800 images
- 230 videos
- 400 attached files

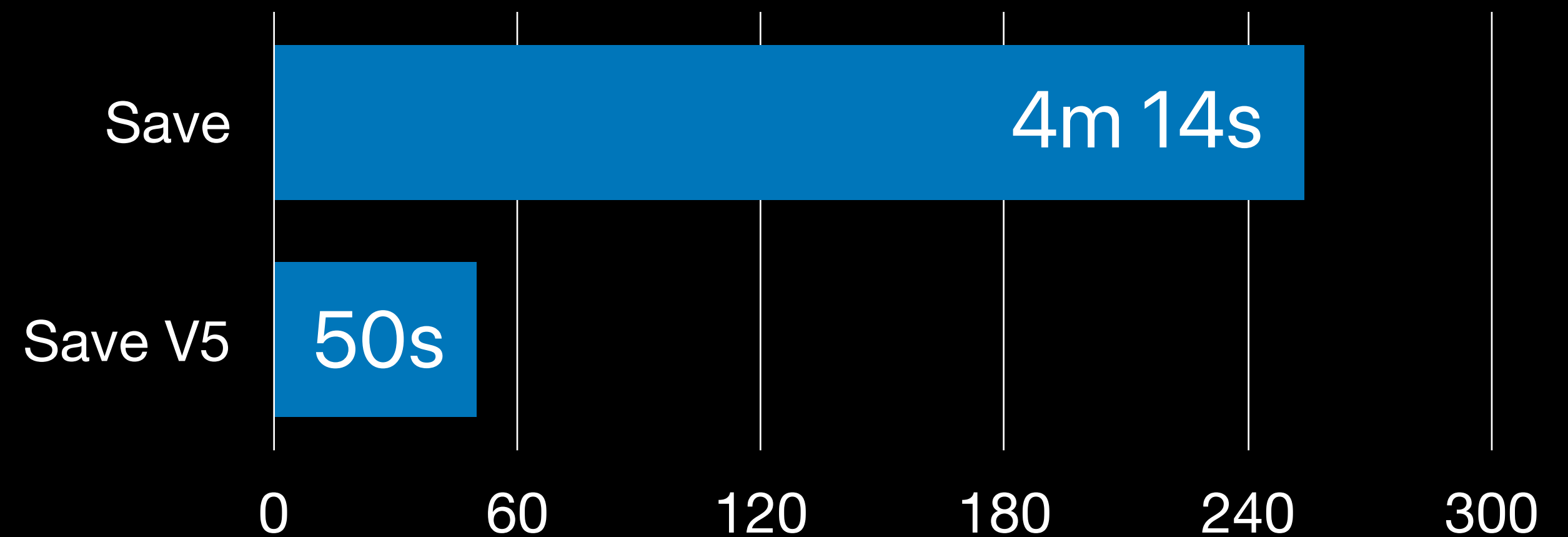


Improving save time



large website.sparkle

- 35 GB
- 300 pages
- 100k words
- 800 images
- 230 videos
- 400 attached files



Improving save time

Improving save time

Pitfalls:

Improving save time

Pitfalls:

- sqlite fragmentation

Improving save time

Pitfalls:

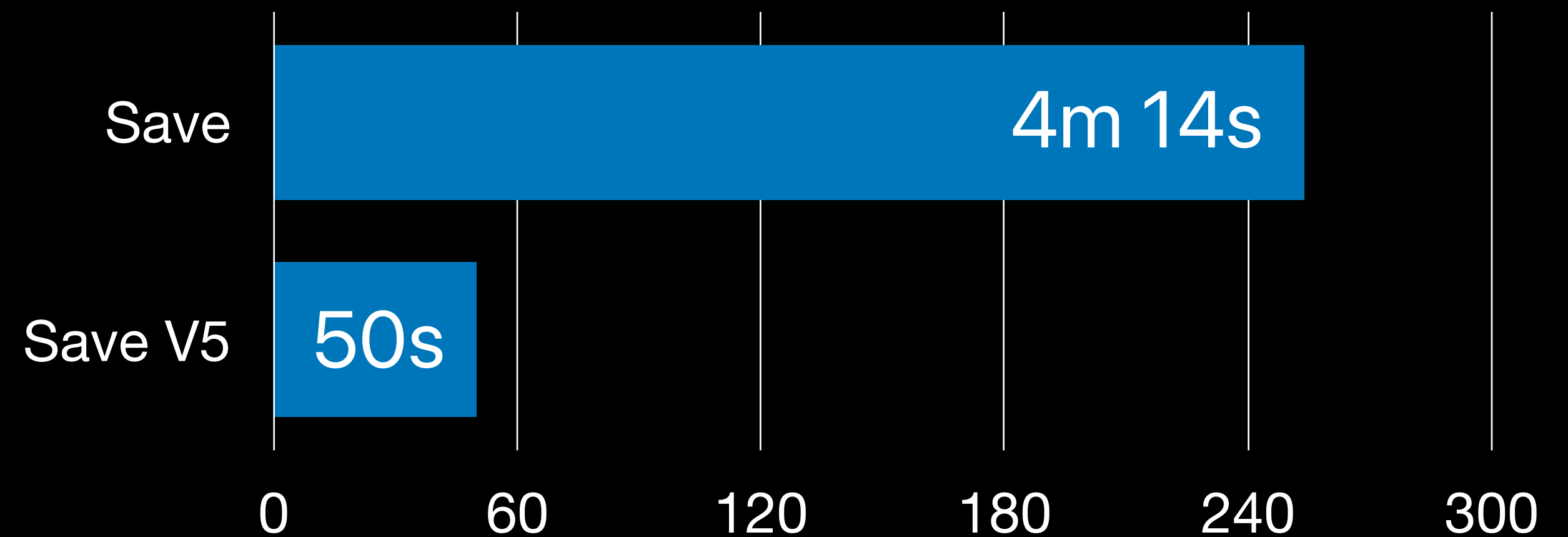
- sqlite fragmentation
- APFS copy-on-write fragmentation

Improving save time



large website.sparkle

- 35 GB
- 300 pages
- 100k words
- 800 images
- 230 videos
- 400 attached files

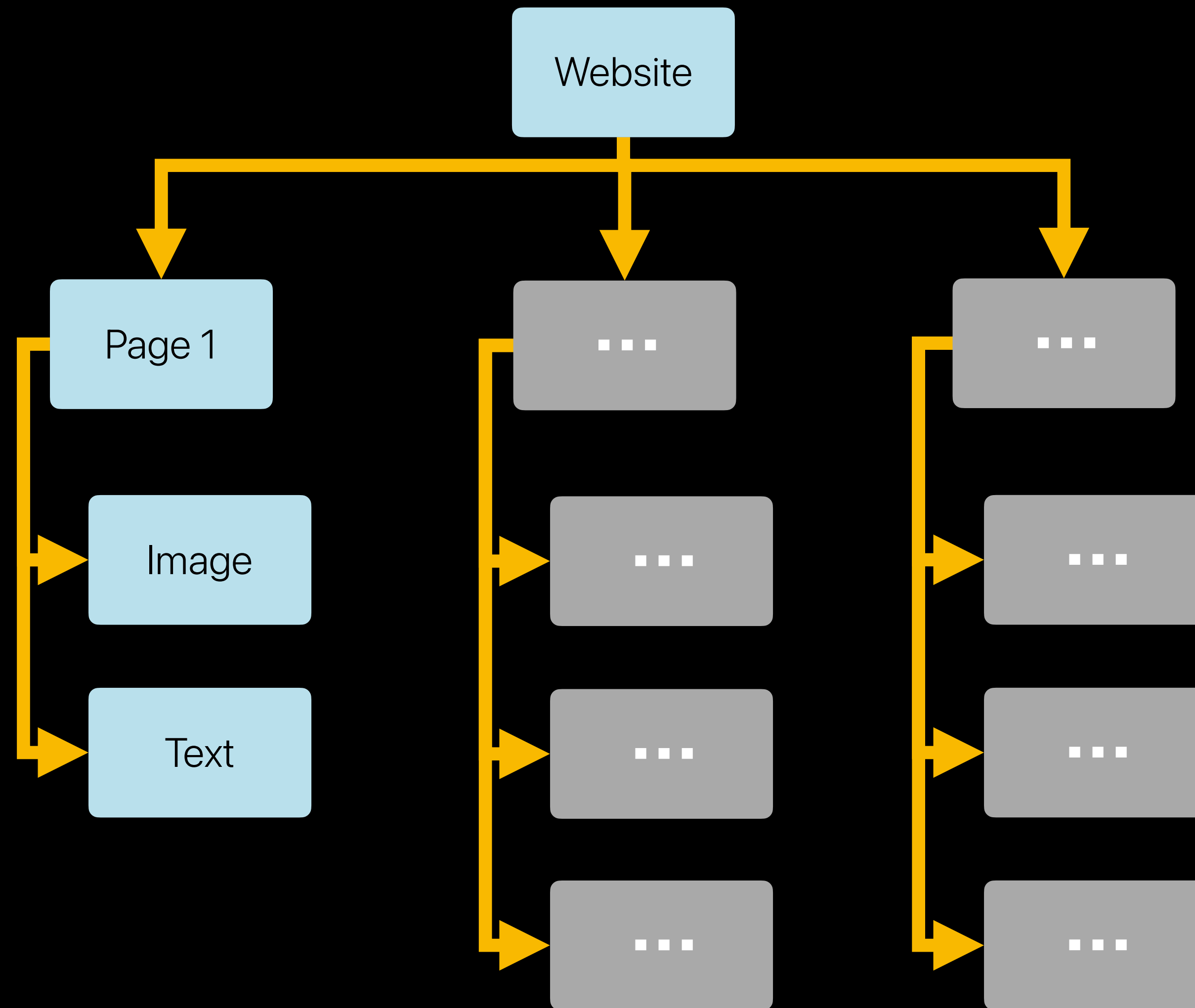


Improving save time

- Don't do unnecessary work

Improving save time

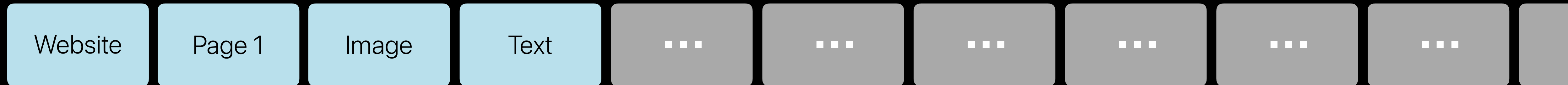
Object Store



Improving save time



website.sparkle



clone.sparkle

Improving save time



website.sparkle

Website

Page 1

Image

Text

...

...

...

...

...

...



clone.sparkle

Improving save time



website.sparkle

Website

Page 1

Image

Text

Page 2

Gallery

Movie

Text

Page3

Button



clone.sparkle

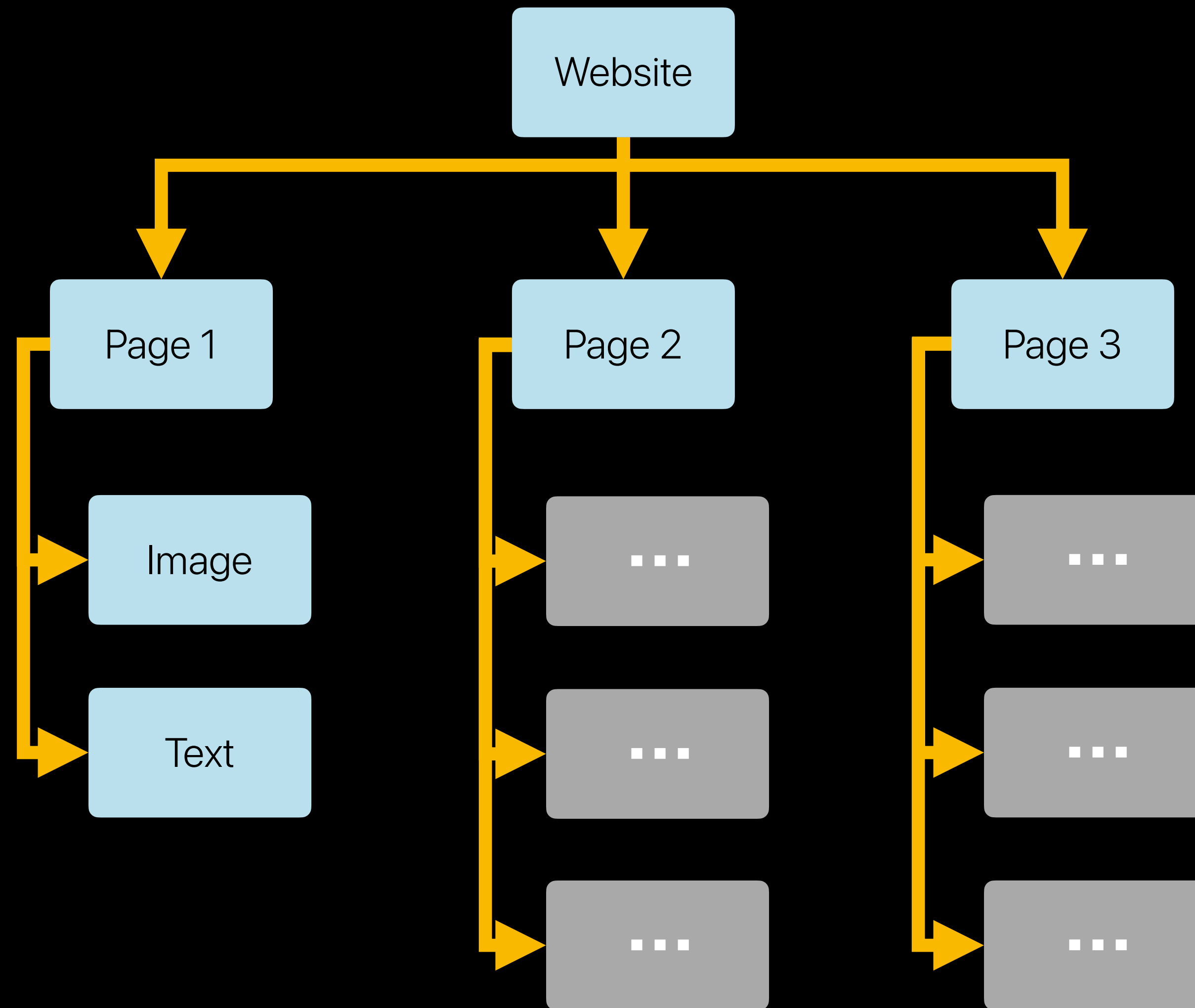
Improving save time

- Don't do unnecessary work
- Don't load objects only to save them

Except...

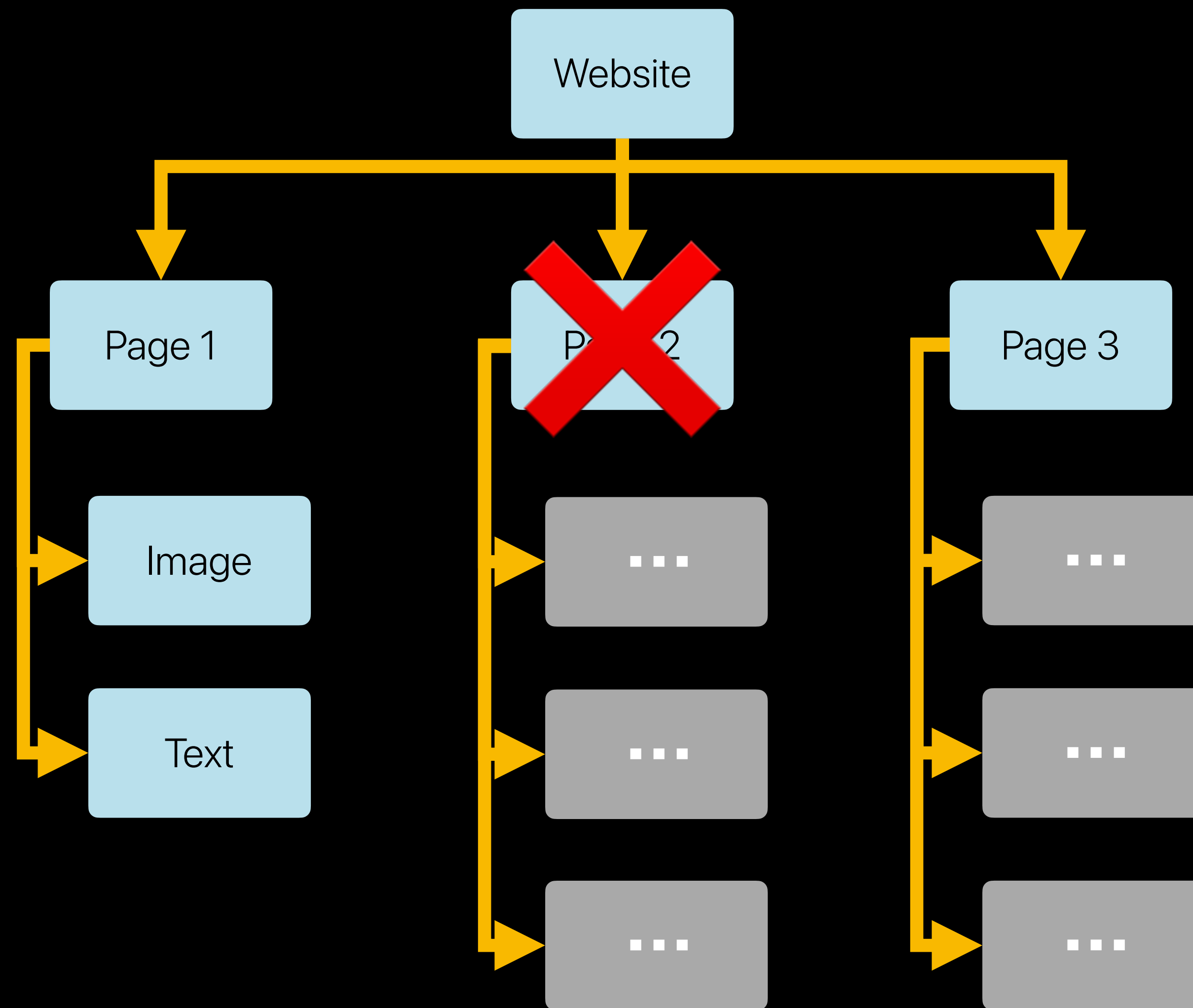
Improving save time

Object Store



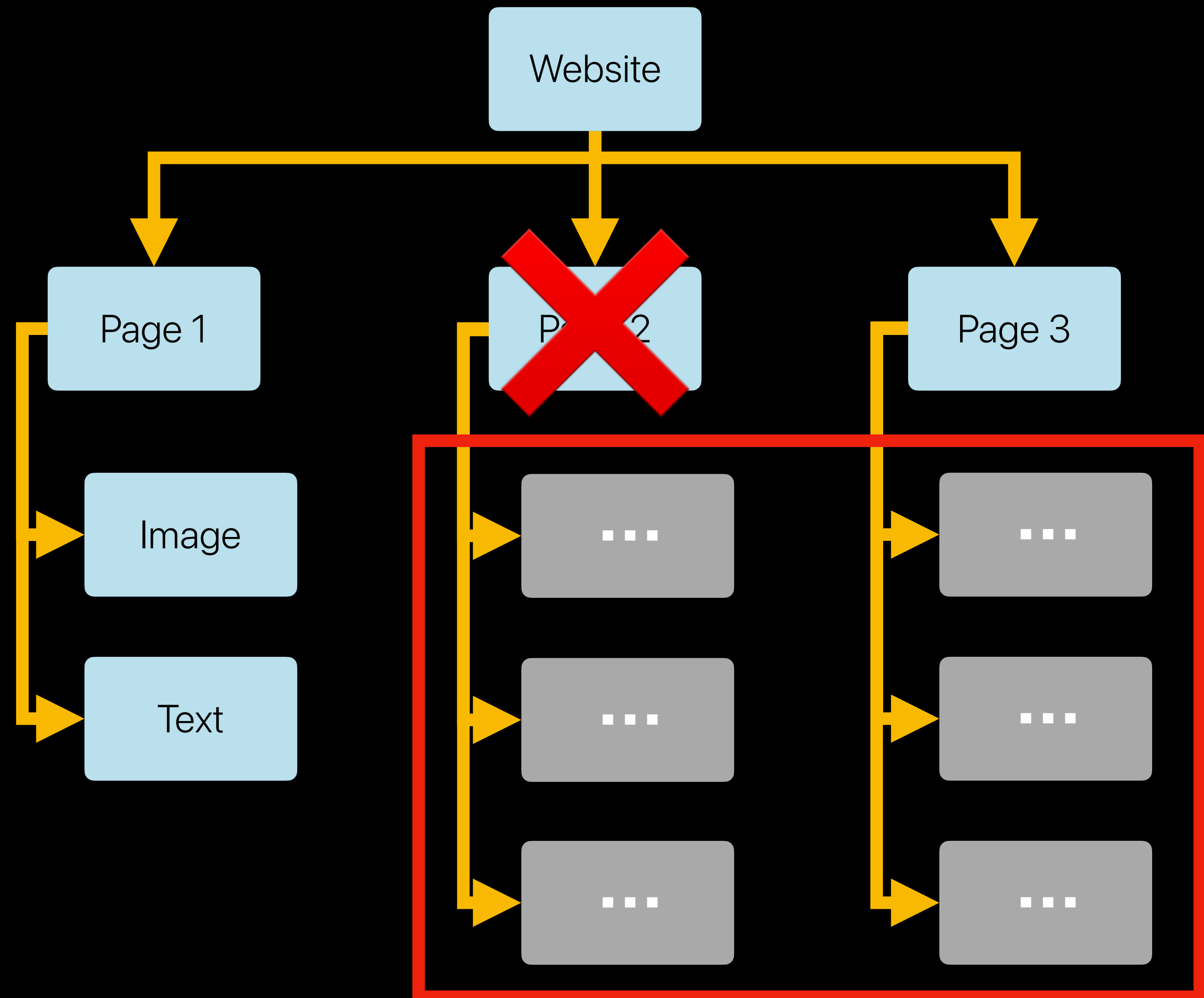
Improving save time

Object Store



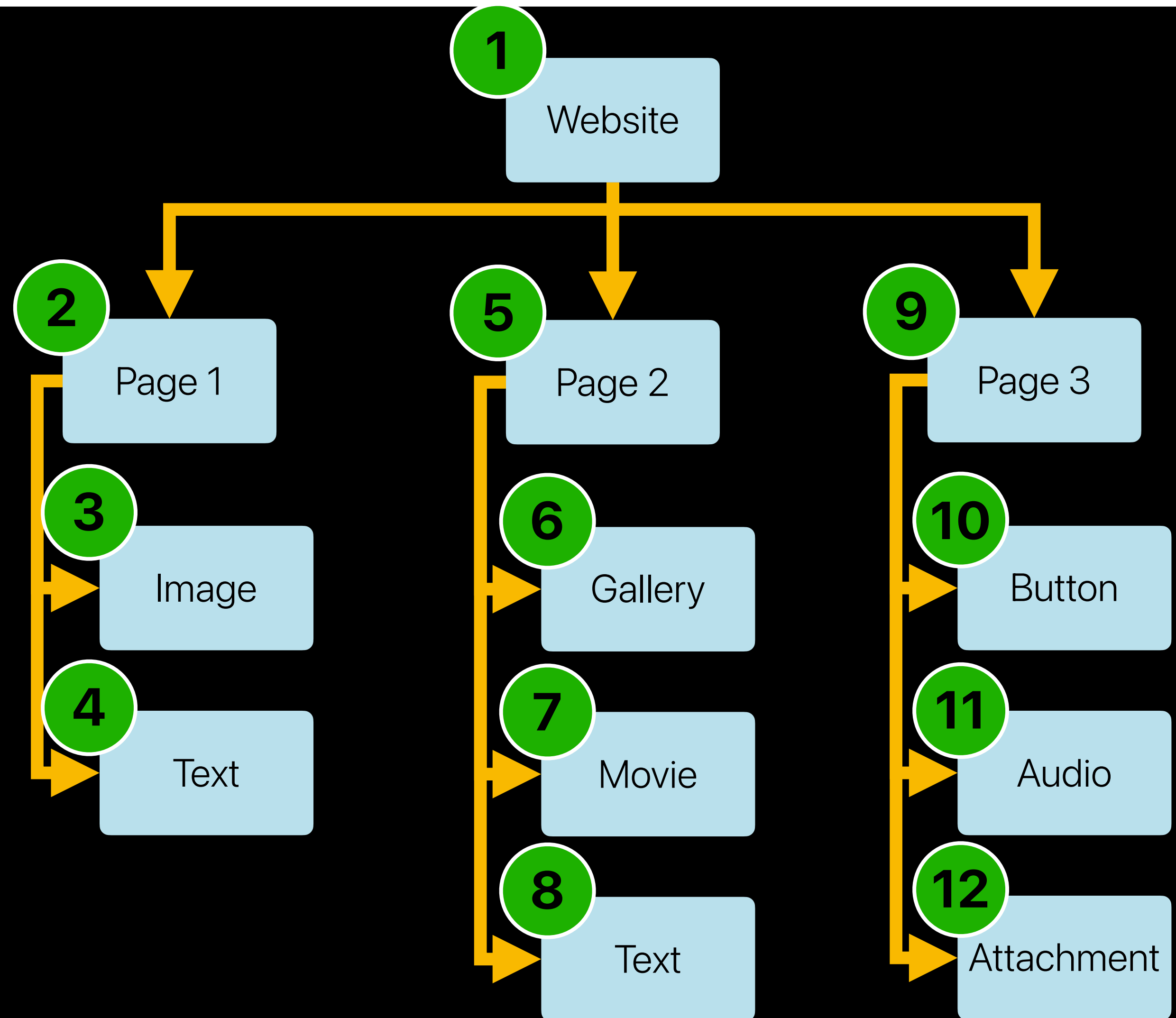
Improving save time

Object Store

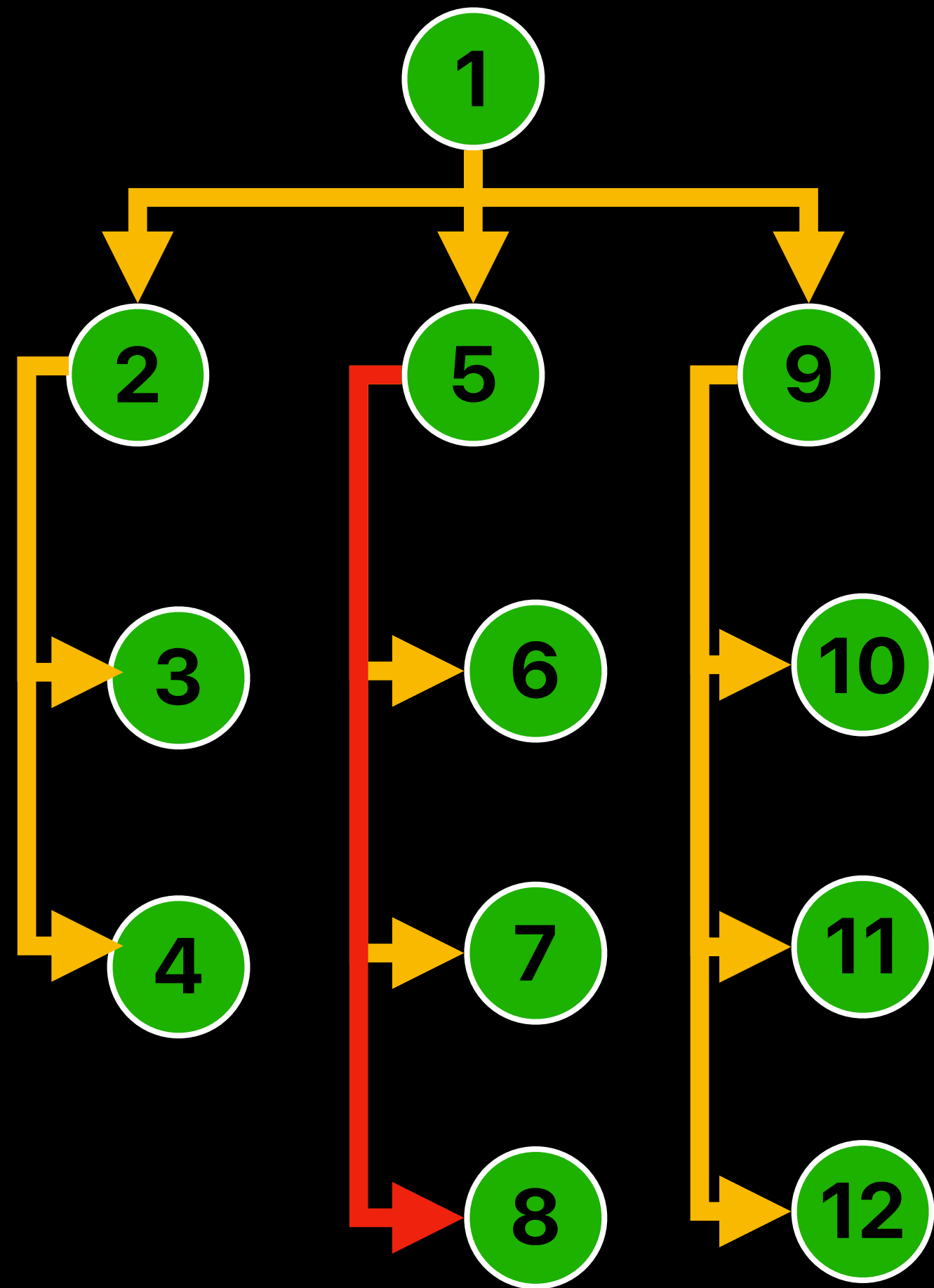


Track live objects

Object Store

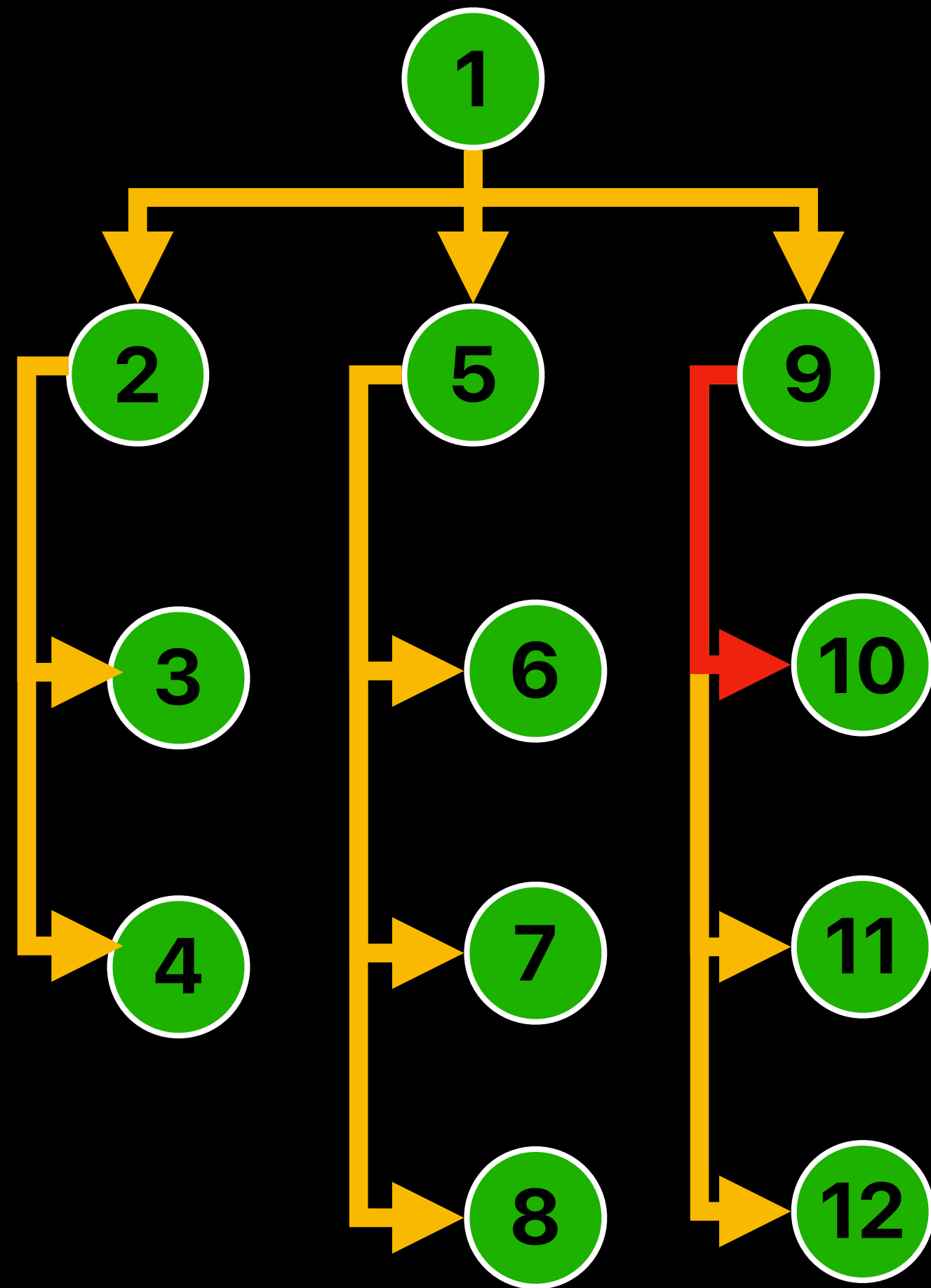


Track live objects



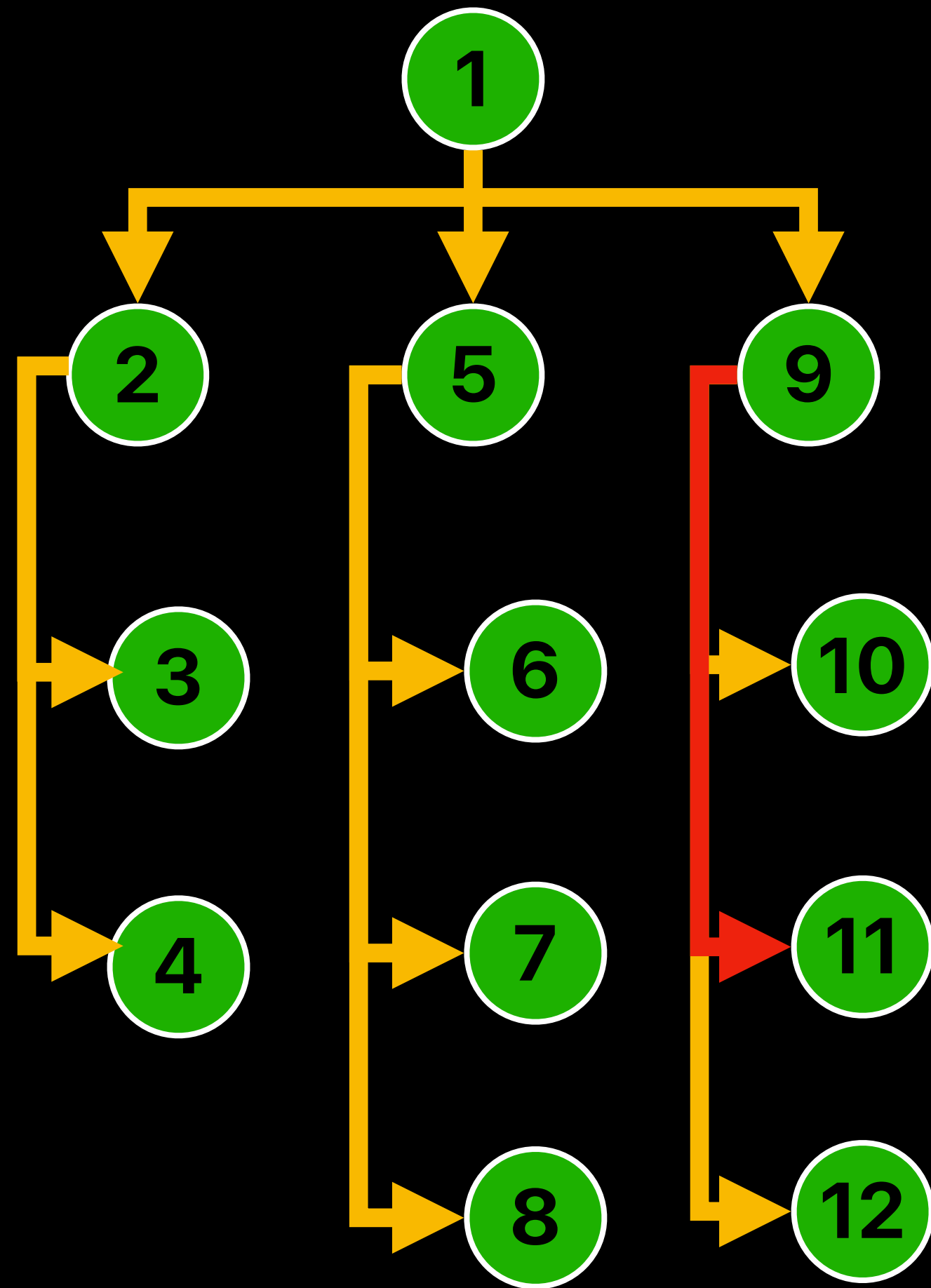
1	2
1	5
1	9
2	3
2	4
5	6
5	7
5	8

Track live objects



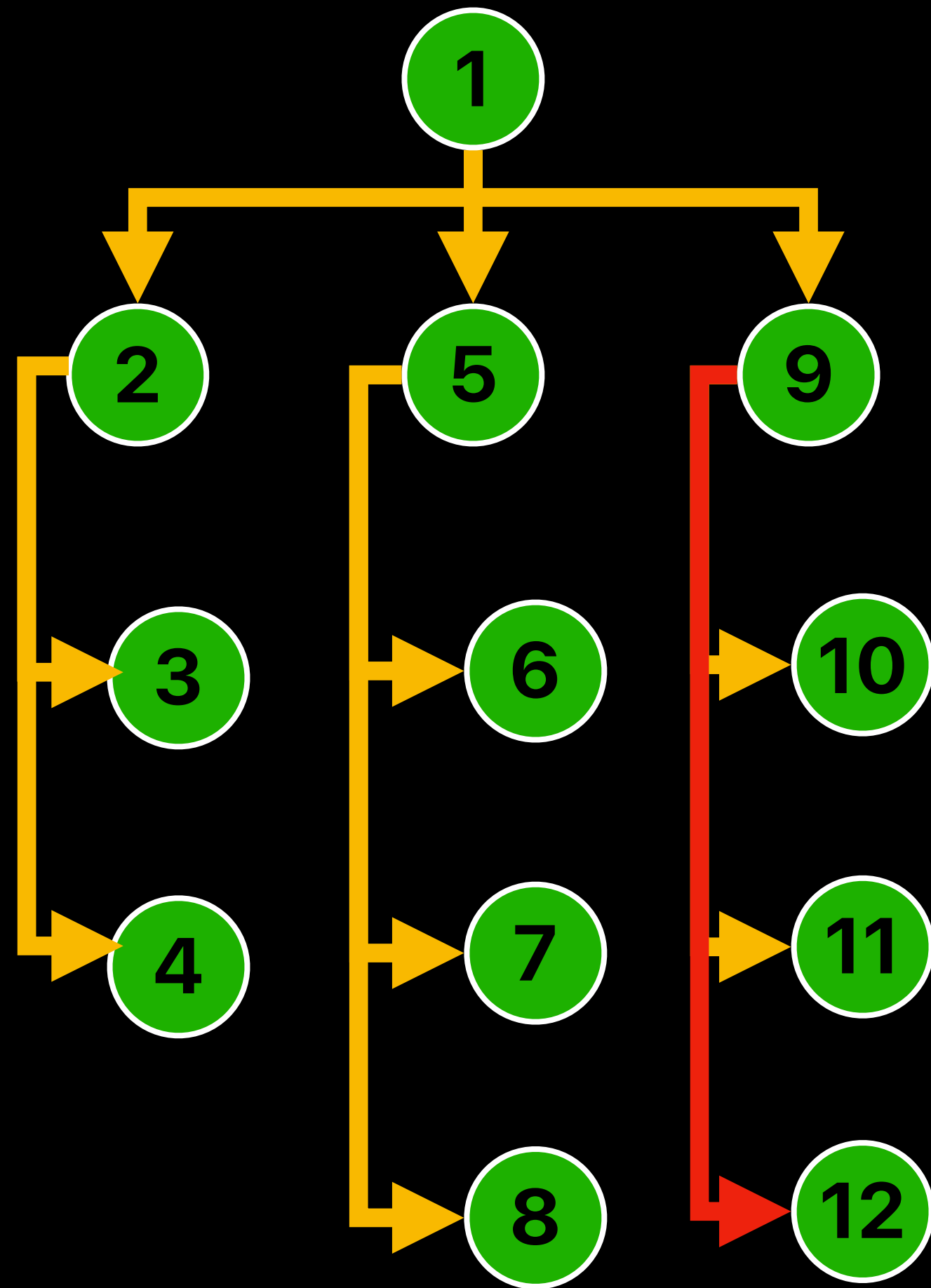
1	2
1	5
1	9
2	3
2	4
5	6
5	7
5	8
9	10

Track live objects



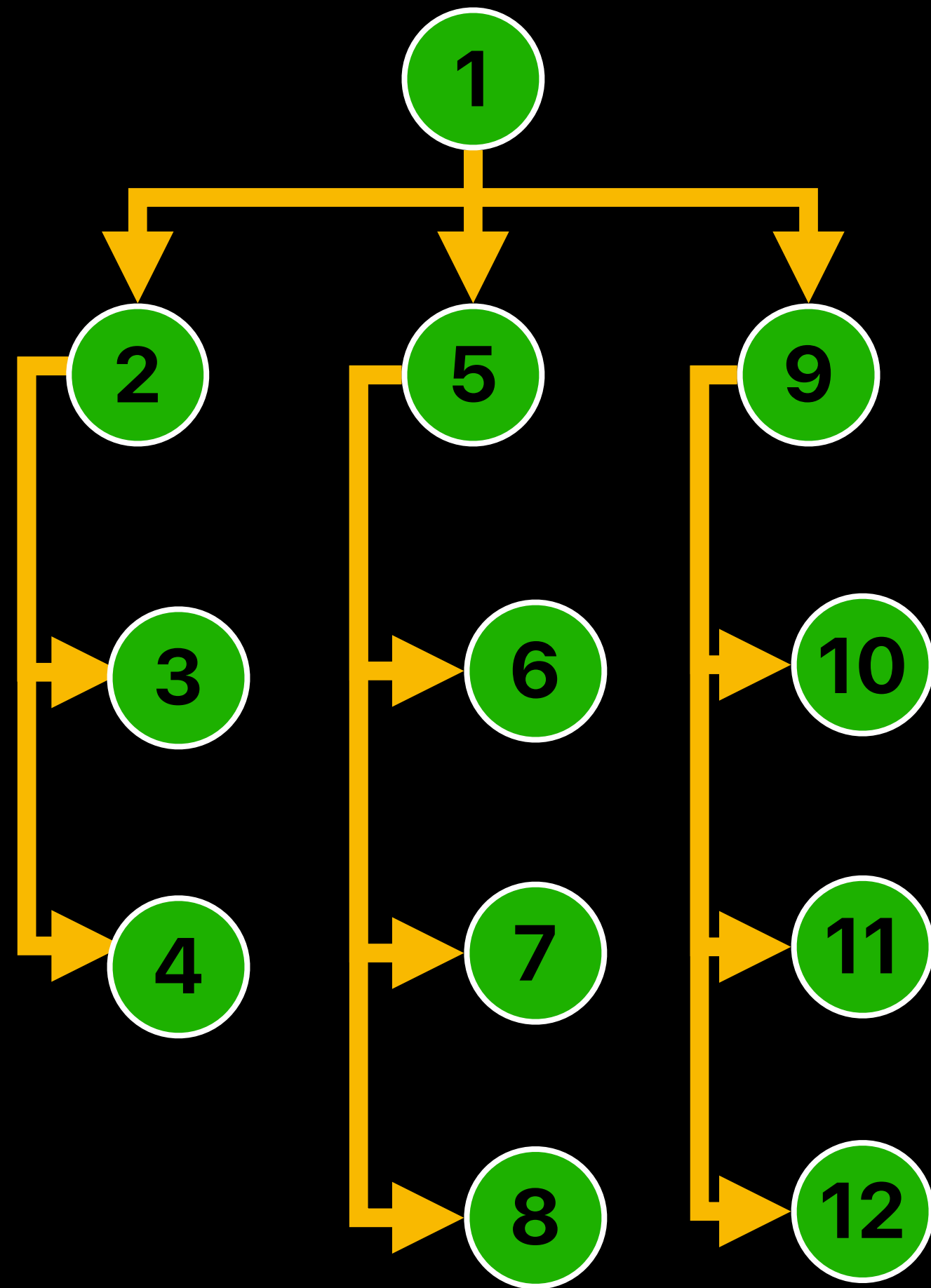
1	2
1	5
1	9
2	3
2	4
5	6
5	7
5	8
9	10
9	11

Track live objects



1	2
1	5
1	9
2	3
2	4
5	6
5	7
5	8
9	10
9	11
9	12

Track live objects



1	2
1	5
1	9
2	3
2	4
5	6
5	7
5	8
9	10
9	11
9	12

Track live objects

1	2
1	5
1	9
2	3
2	4
5	6
5	7
5	8
9	10
9	11
9	12

Track live objects

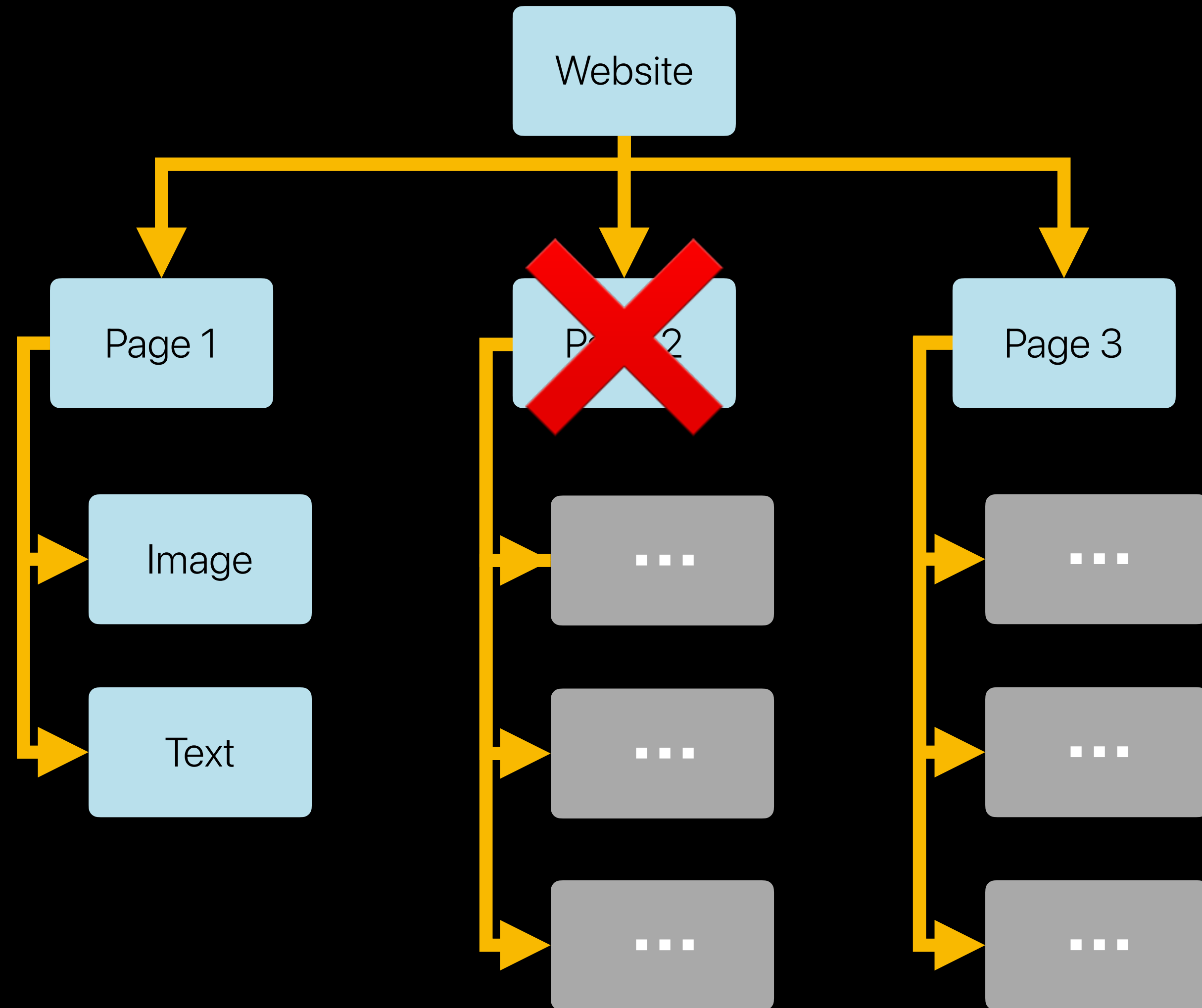
i1	i2	gen
1	2	
1	5	
1	9	
2	3	
2	4	
5	6	
5	7	
5	8	
9	10	
9	11	
9	12	

Track live objects

i1	i2	gen
1	2	41
1	5	41
1	9	41
2	3	41
2	4	41
5	6	41
5	7	41
5	8	41
9	10	41
9	11	41
9	12	41

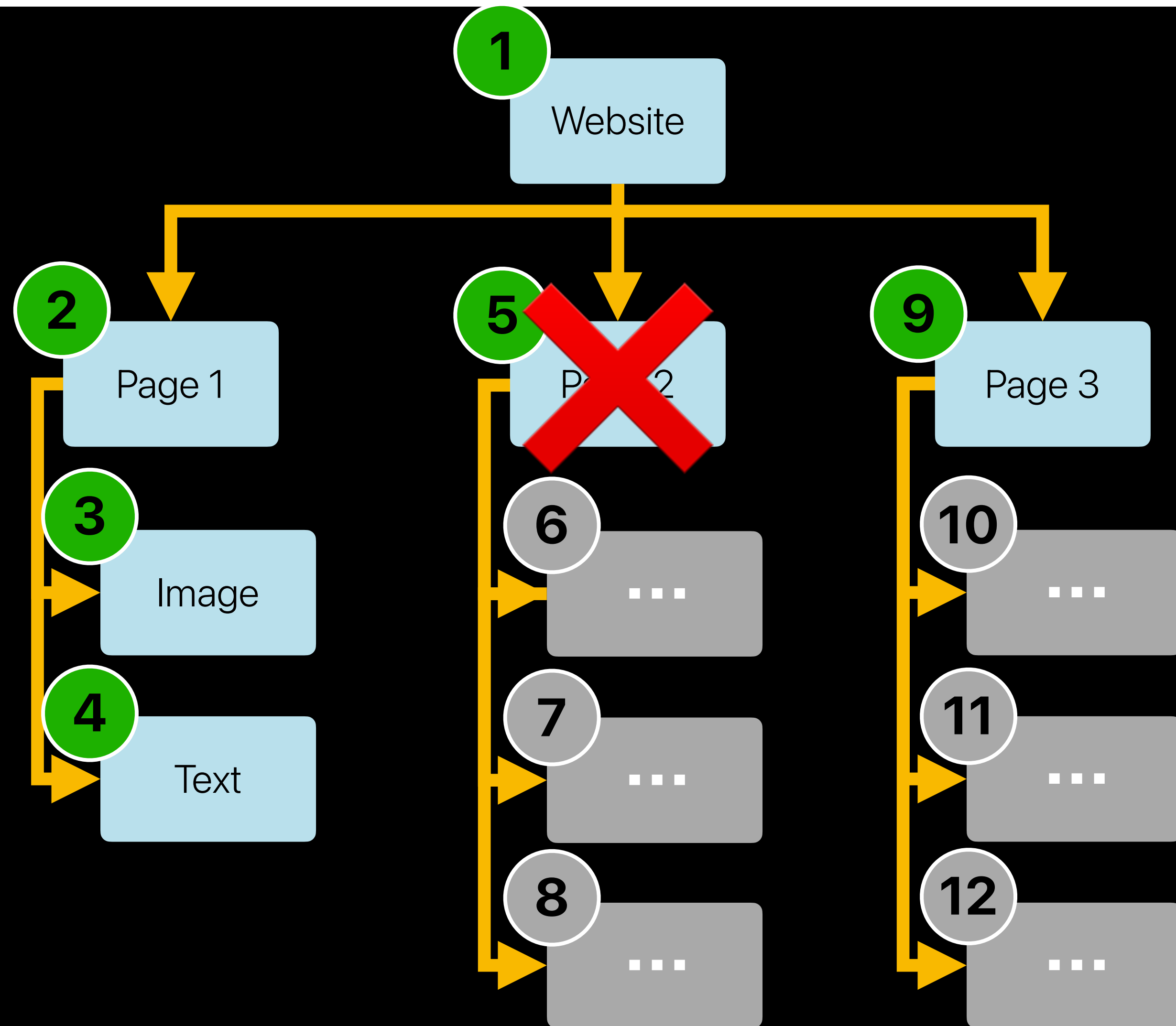
Track live objects

Object Store

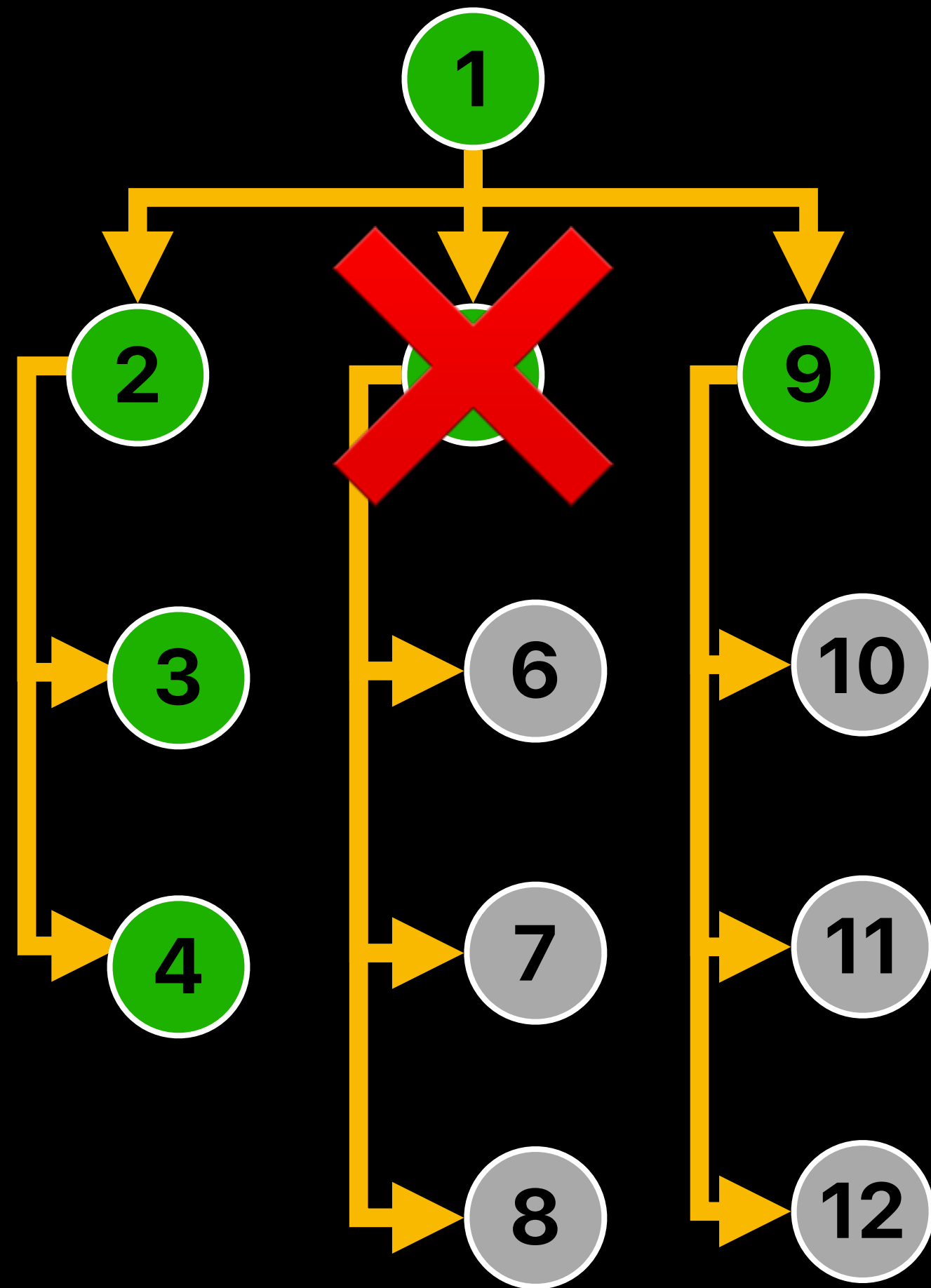


Track live objects

Object Store

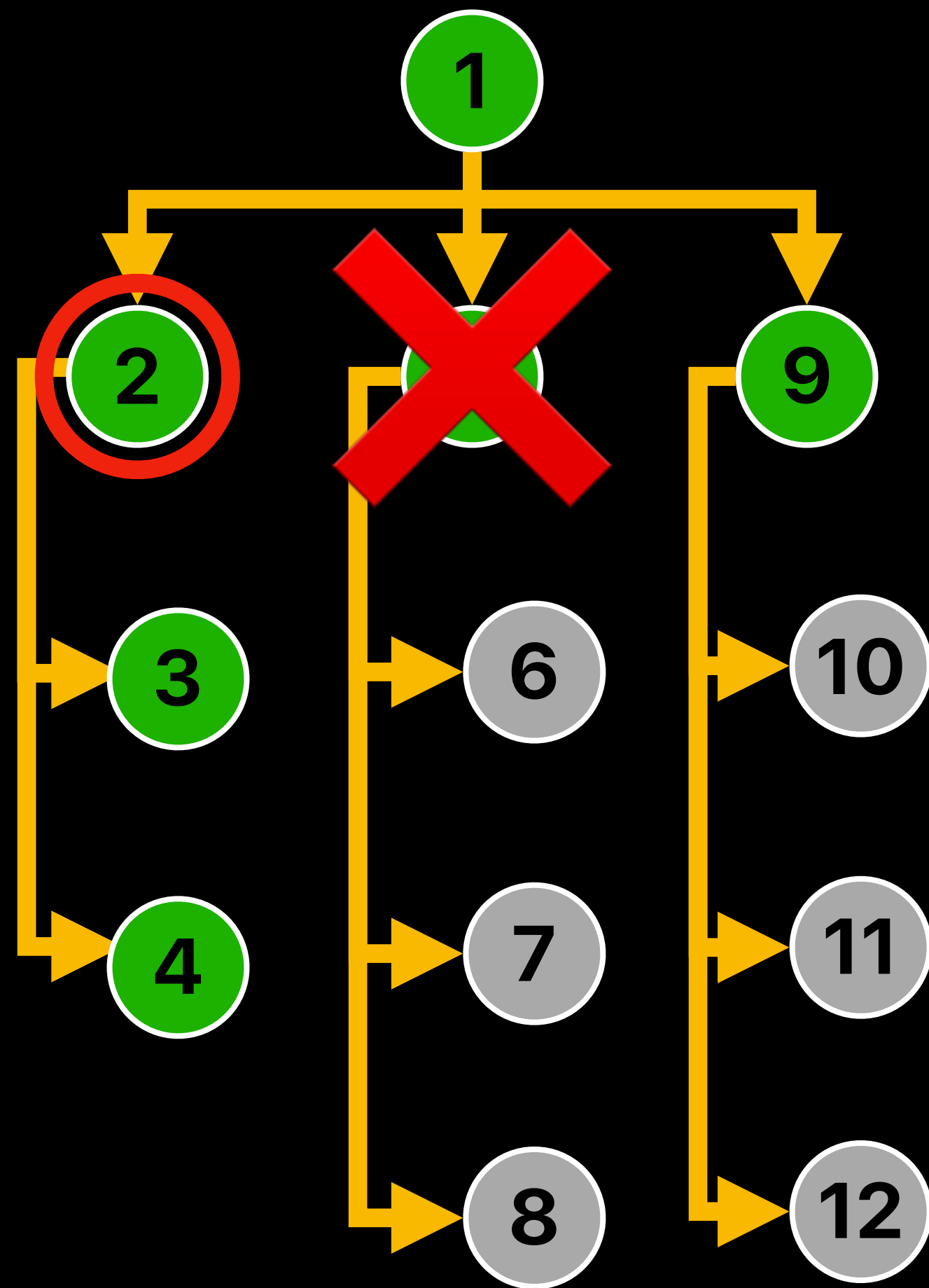


Track live objects



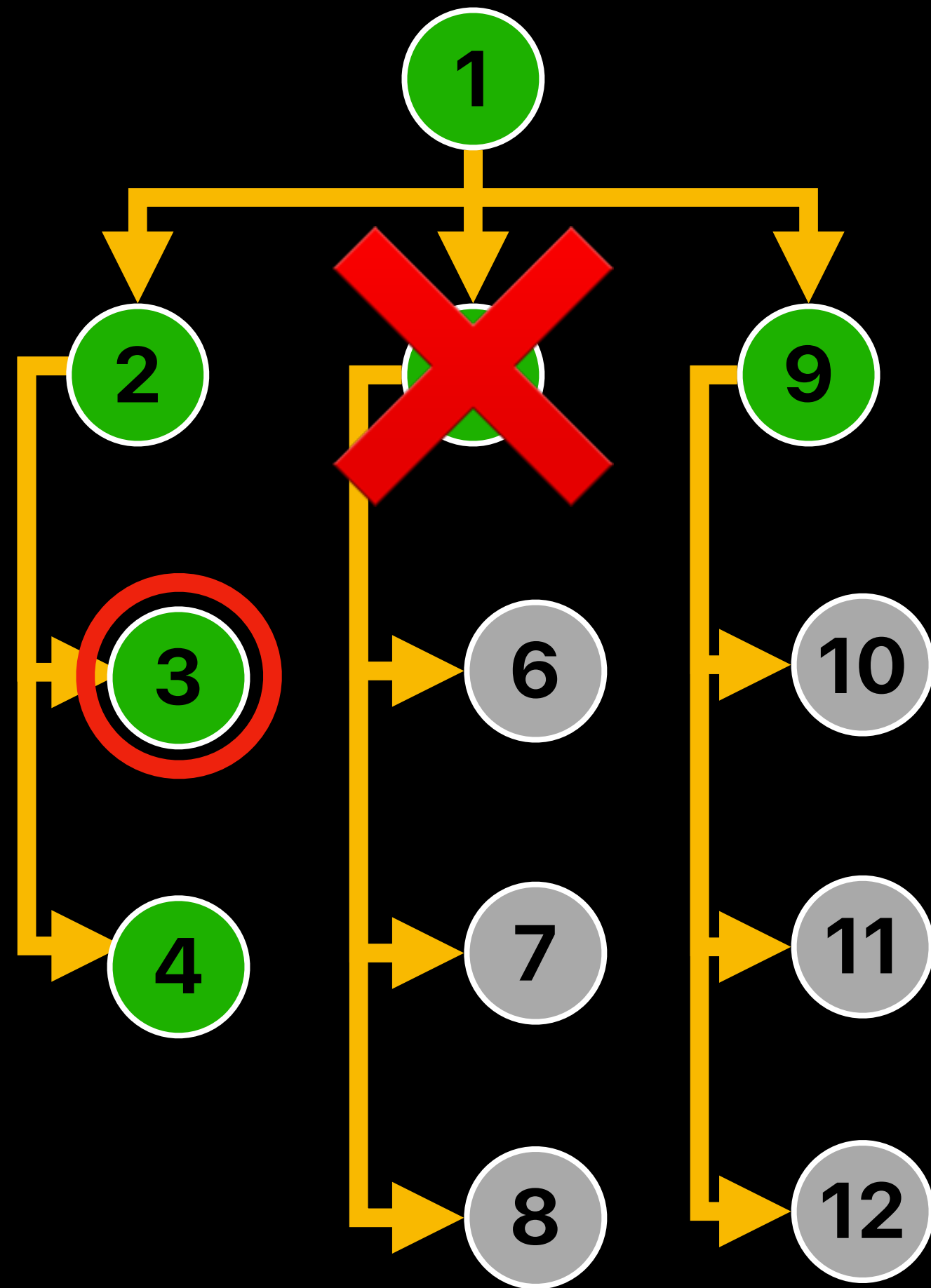
i1	i2	gen
1	2	41
1	5	41
1	9	41
2	3	41
2	4	41
5	6	41
5	7	41
5	8	41
9	10	41
9	11	41
9	12	41

Track live objects



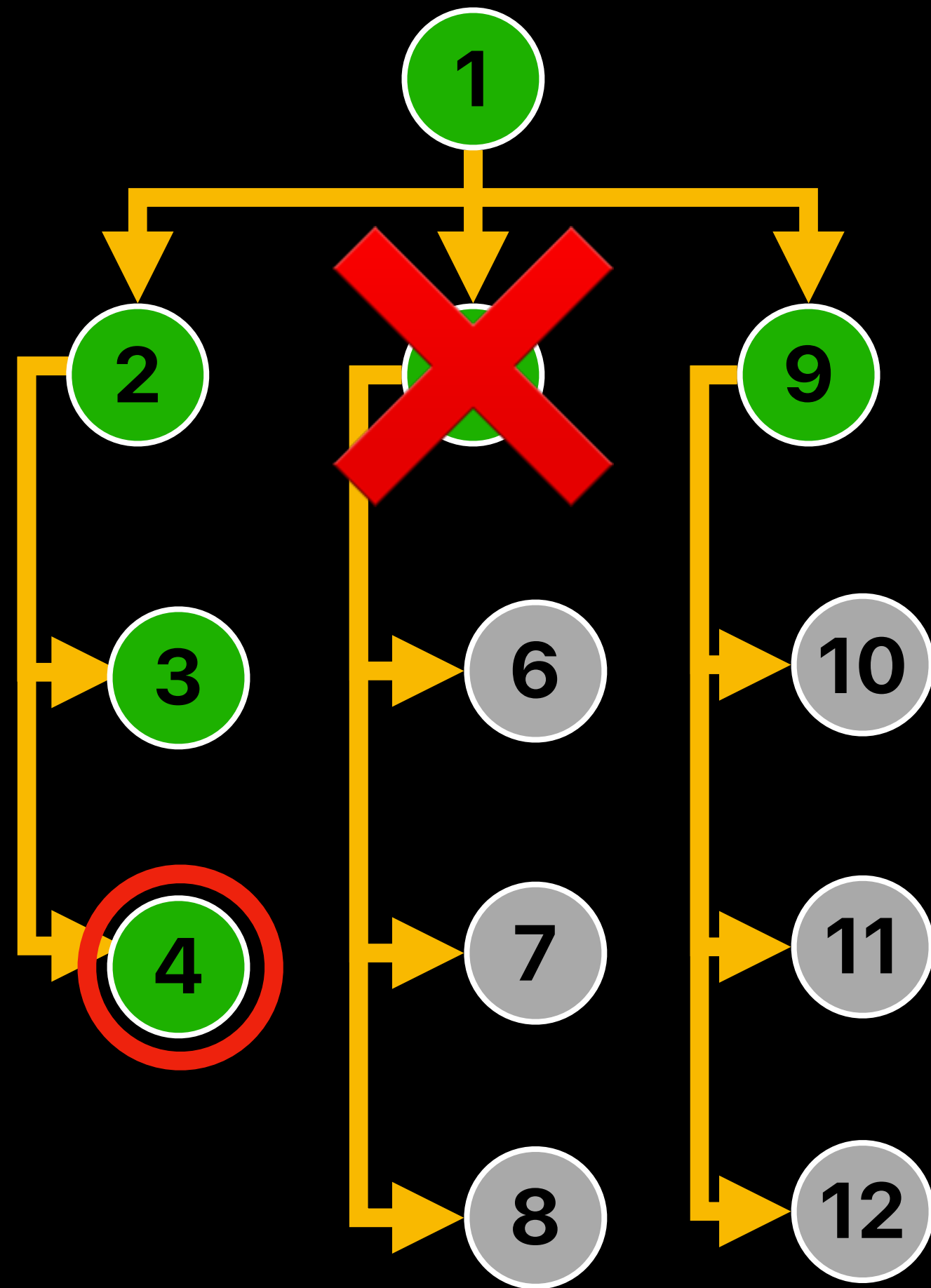
i1	i2	gen
1	2	42
1	5	41
1	9	41
2	3	41
2	4	41
5	6	41
5	7	41
5	8	41
9	10	41
9	11	41
9	12	41

Track live objects



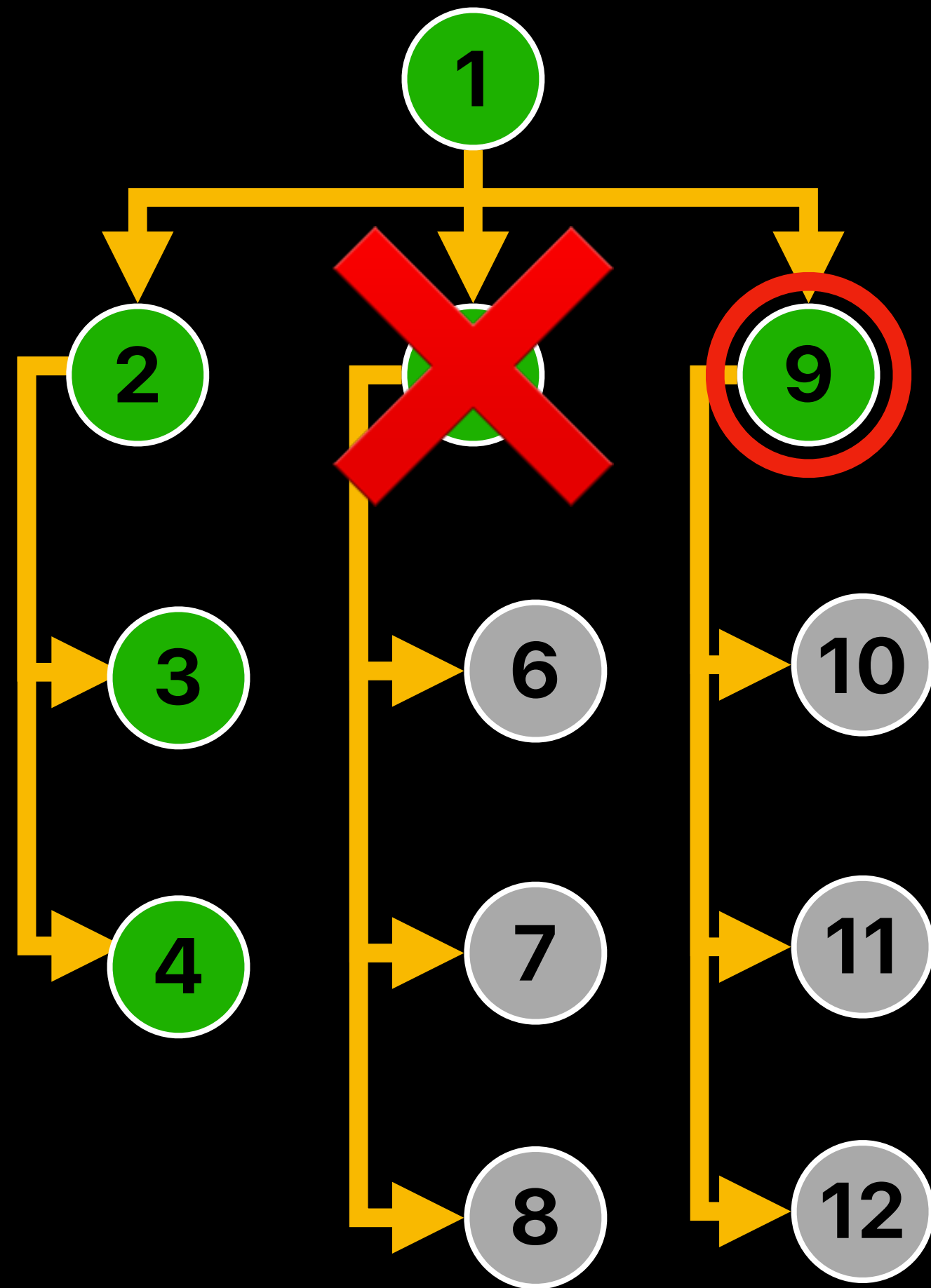
i1	i2	gen
1	2	42
1	5	41
1	9	41
2	3	42
2	4	41
5	6	41
5	7	41
5	8	41
9	10	41
9	11	41
9	12	41

Track live objects



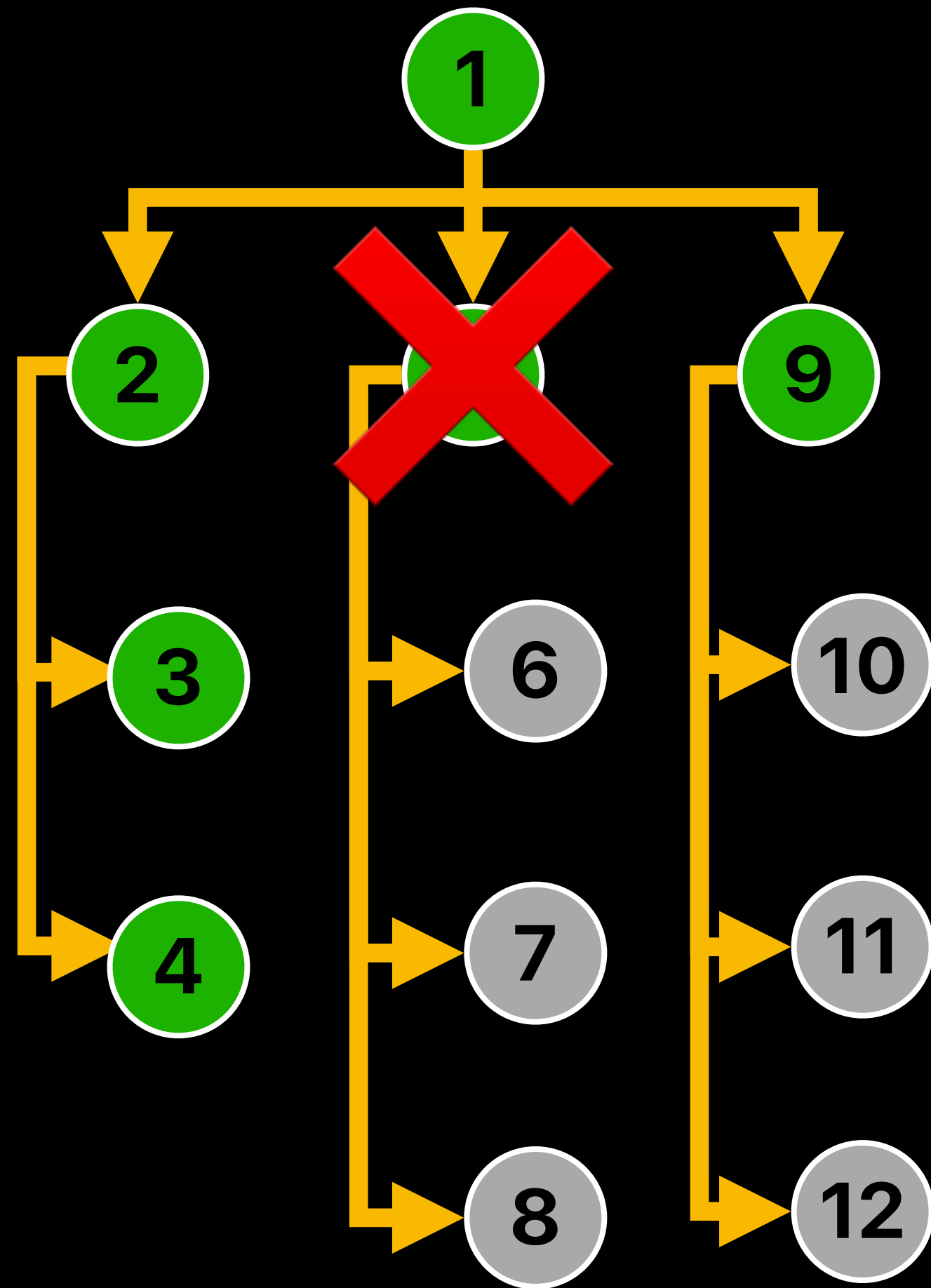
i1	i2	gen
1	2	42
1	5	41
1	9	41
2	3	42
2	4	42
5	6	41
5	7	41
5	8	41
9	10	41
9	11	41
9	12	41

Track live objects



i1	i2	gen
1	2	42
1	5	41
1	9	42
2	3	42
2	4	42
5	6	41
5	7	41
5	8	41
9	10	41
9	11	41
9	12	41

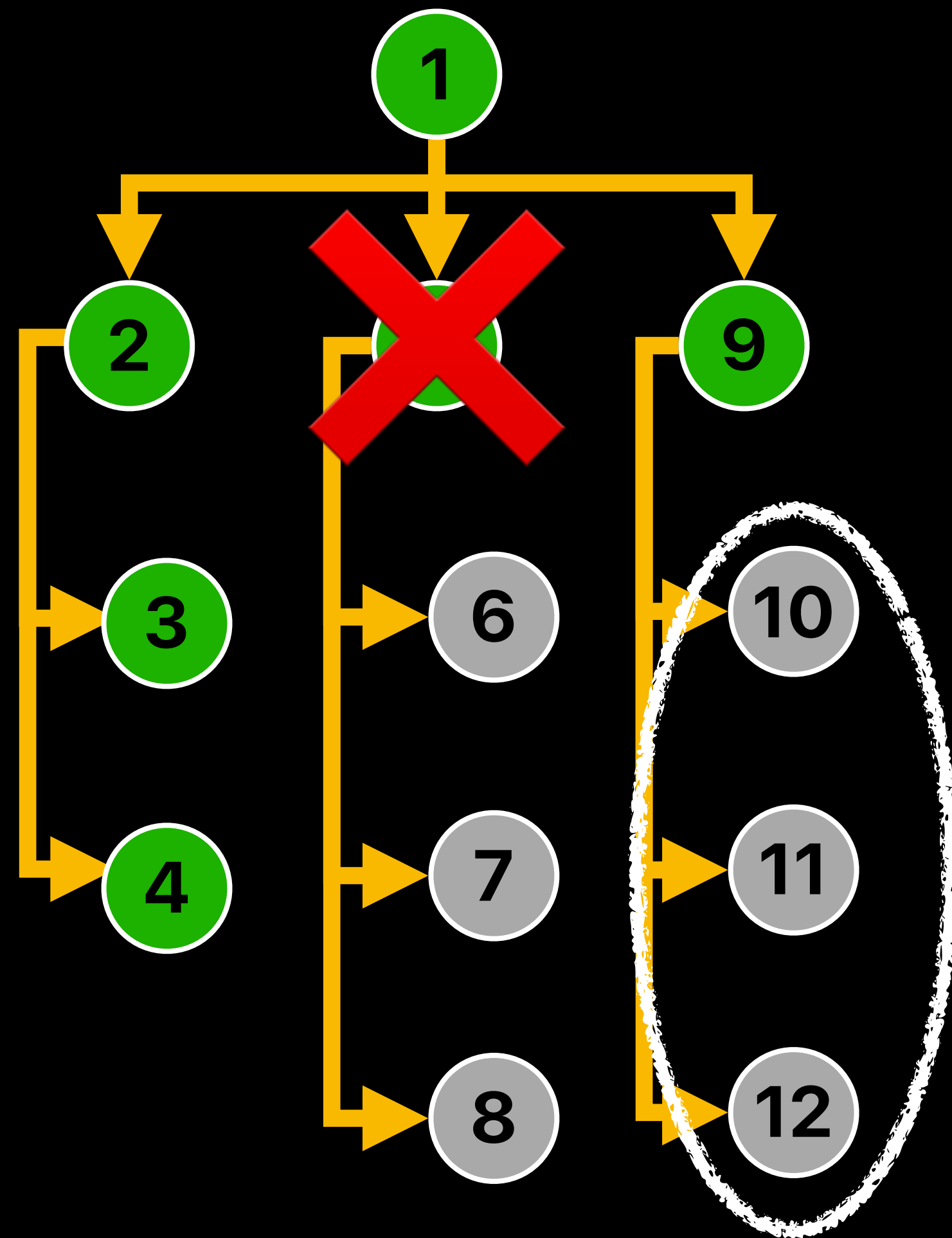
Track live objects



i1	i2	gen
1	2	42
1	5	41
1	9	42
2	3	42
2	4	42
5	6	41
5	7	41
5	8	41
9	10	41
9	11	41
9	12	41



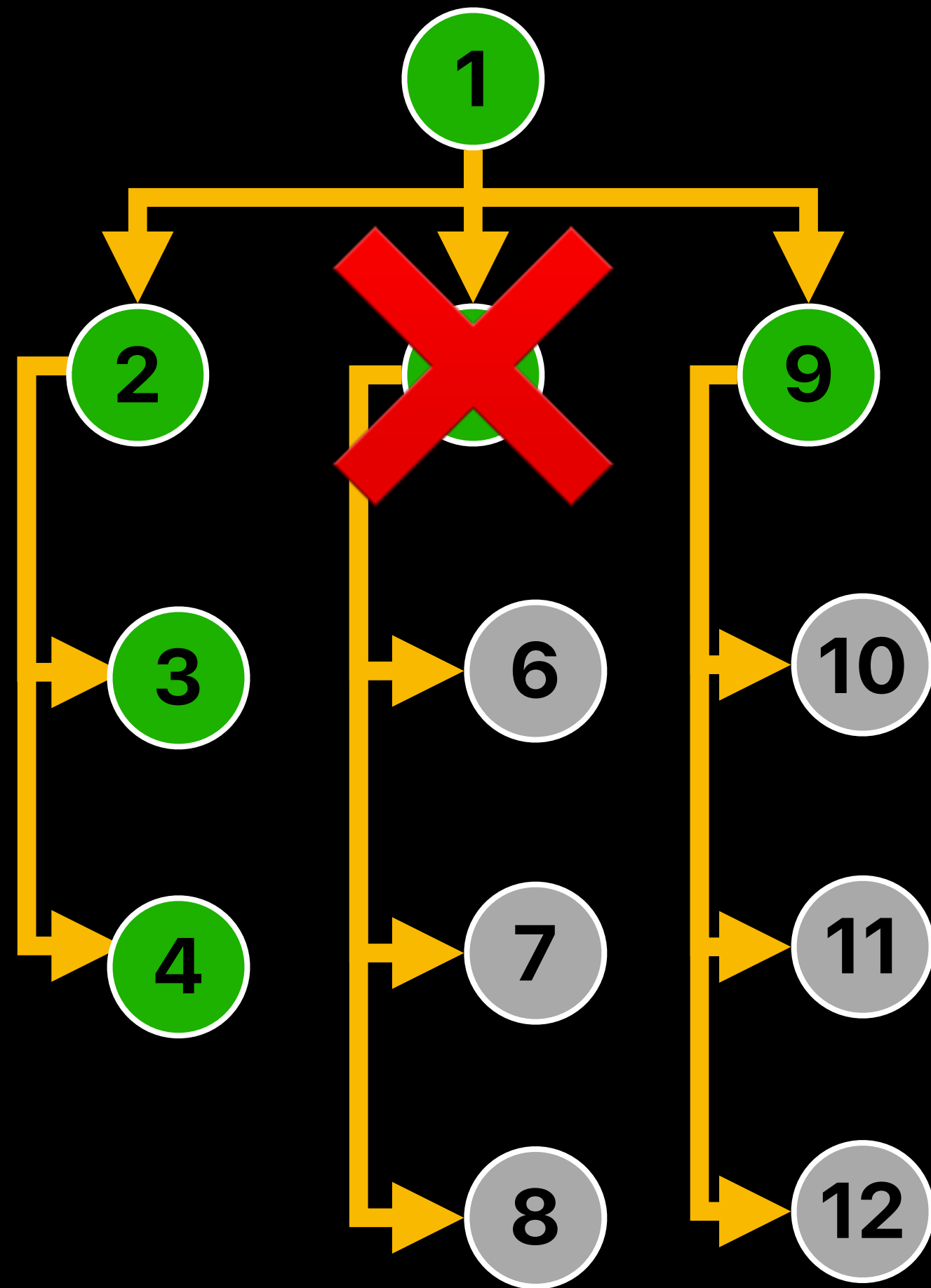
Track live objects



i1	i2	gen
1	2	42
1	5	41
1	9	42
2	3	42
2	4	42
5	6	41
5	7	41
5	8	41
9	10	41
9	11	41
9	12	41

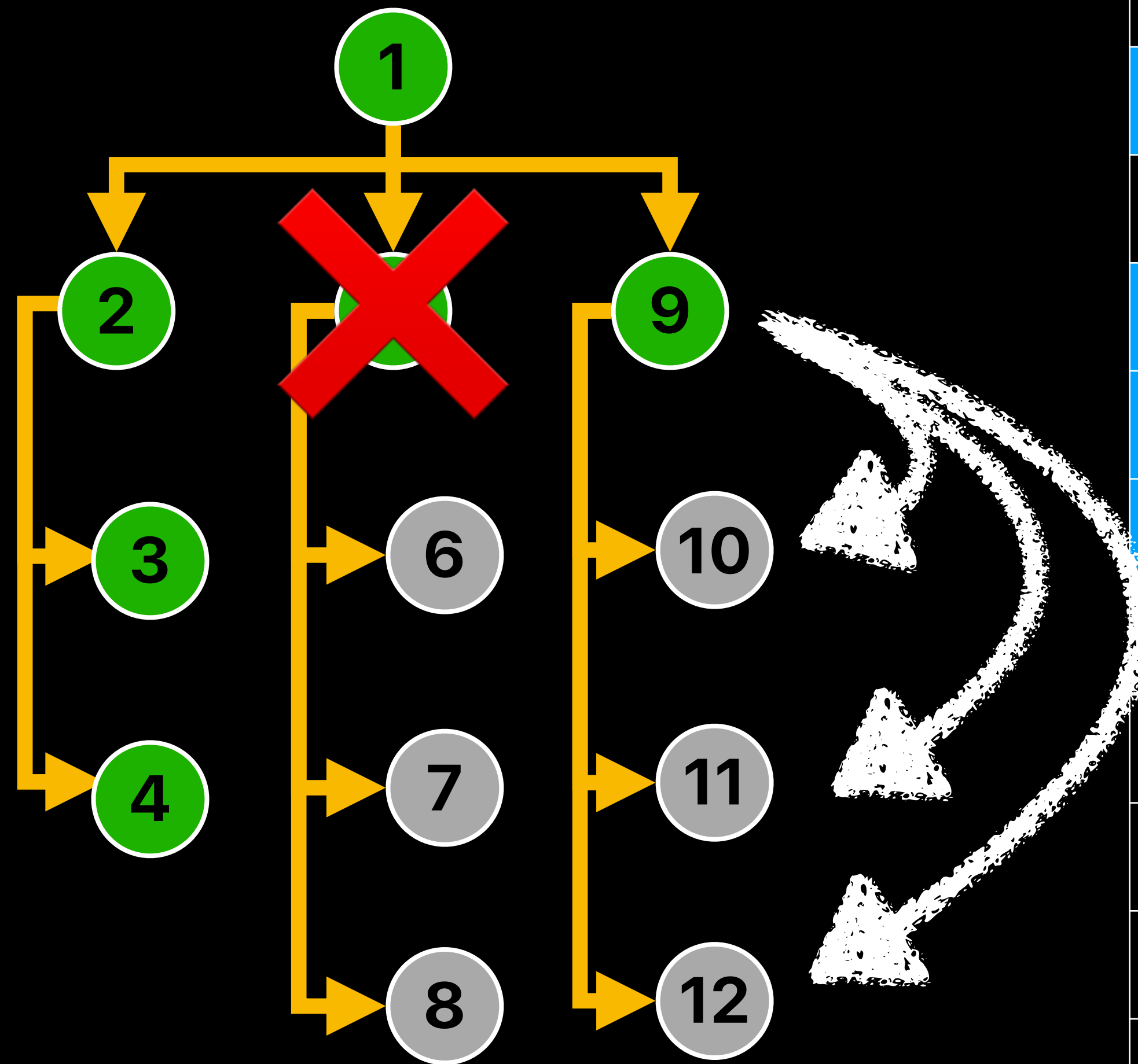


Track live objects



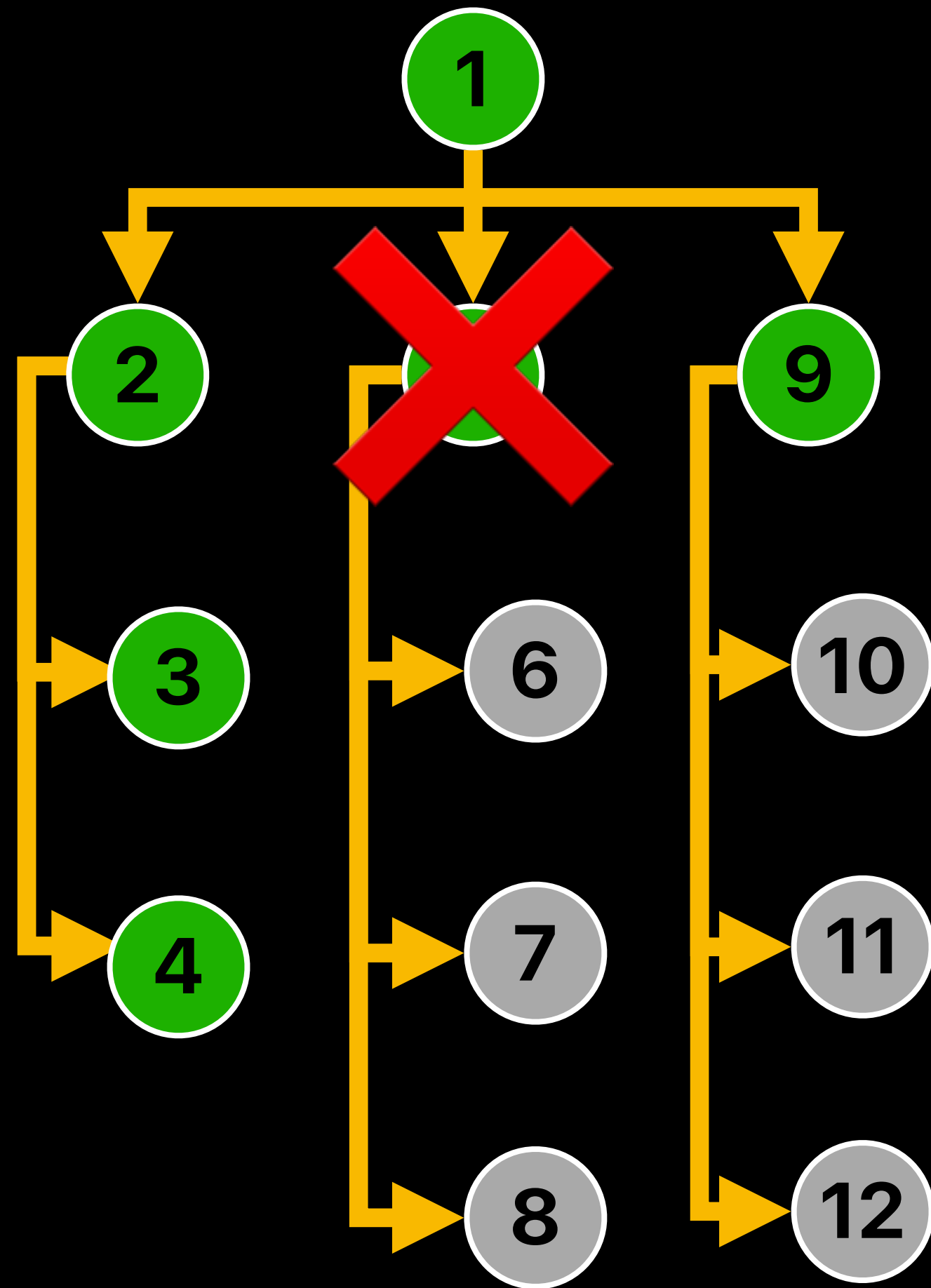
i1	i2	gen
1	2	42
1	5	41
1	9	42
2	3	42
2	4	42
5	6	41
5	7	41
5	8	41
9	10	41
9	11	41
9	12	41

Track live objects



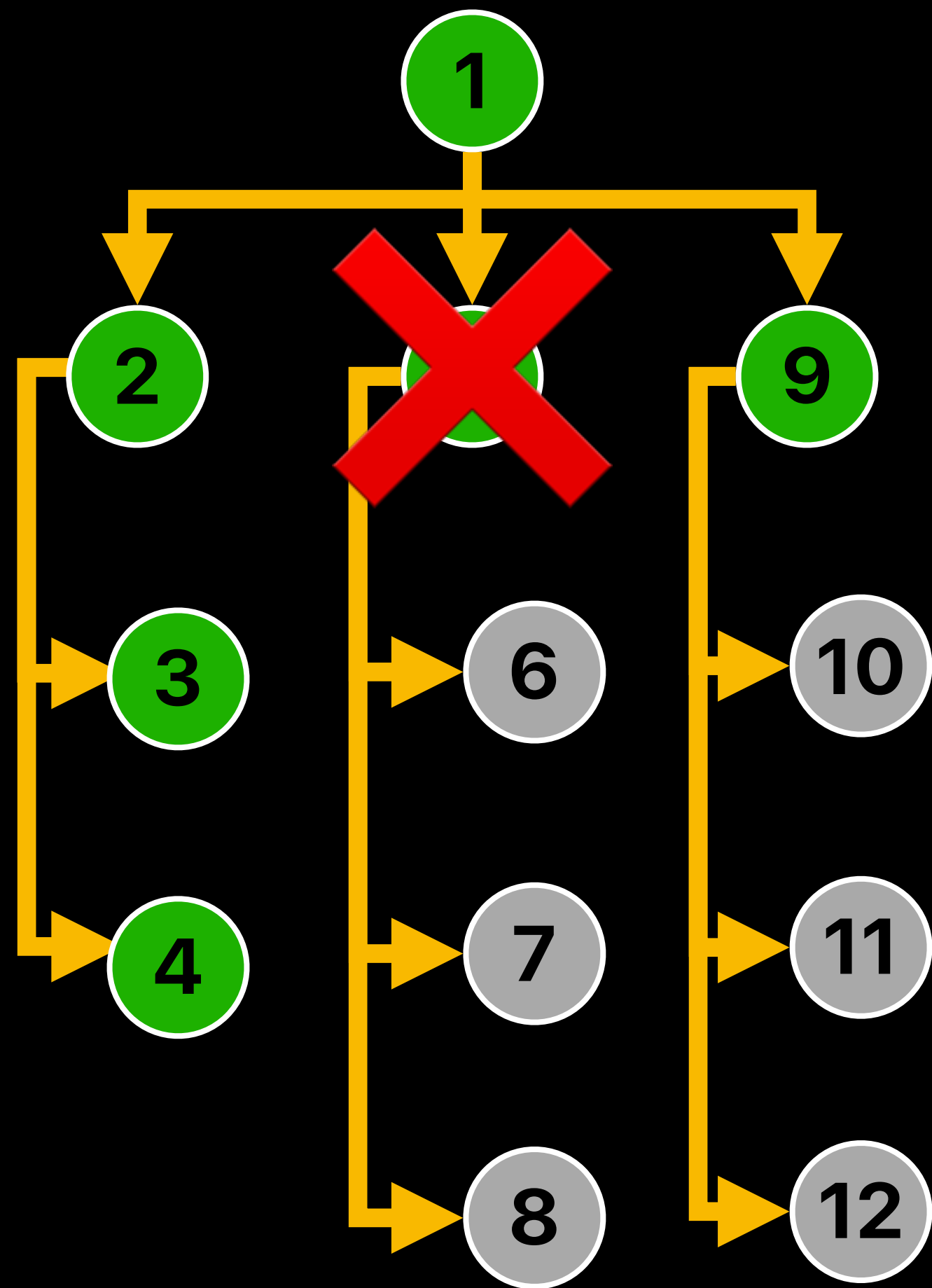
i1	i2	gen
1	2	42
1	5	41
1	9	42
2	3	42
2	4	42
5	6	41
5	7	41
5	8	41
9	10	41
9	11	41
9	12	41

Track live objects



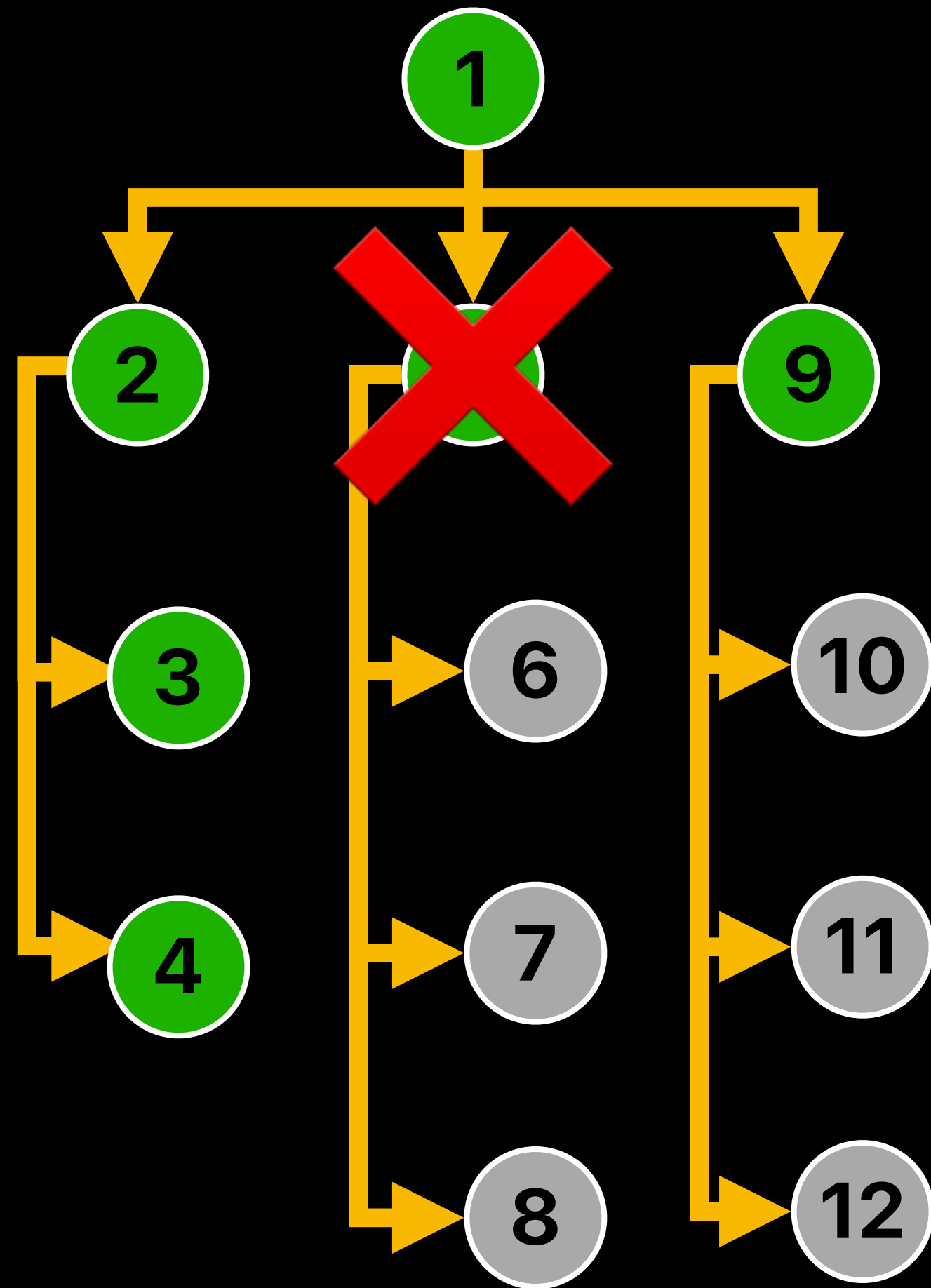
i1	i2	gen
1	2	42
1	5	41
1	9	42
2	3	42
2	4	42
5	6	41
5	7	41
5	8	41
9	10	41
9	11	41
9	12	41

Track live objects



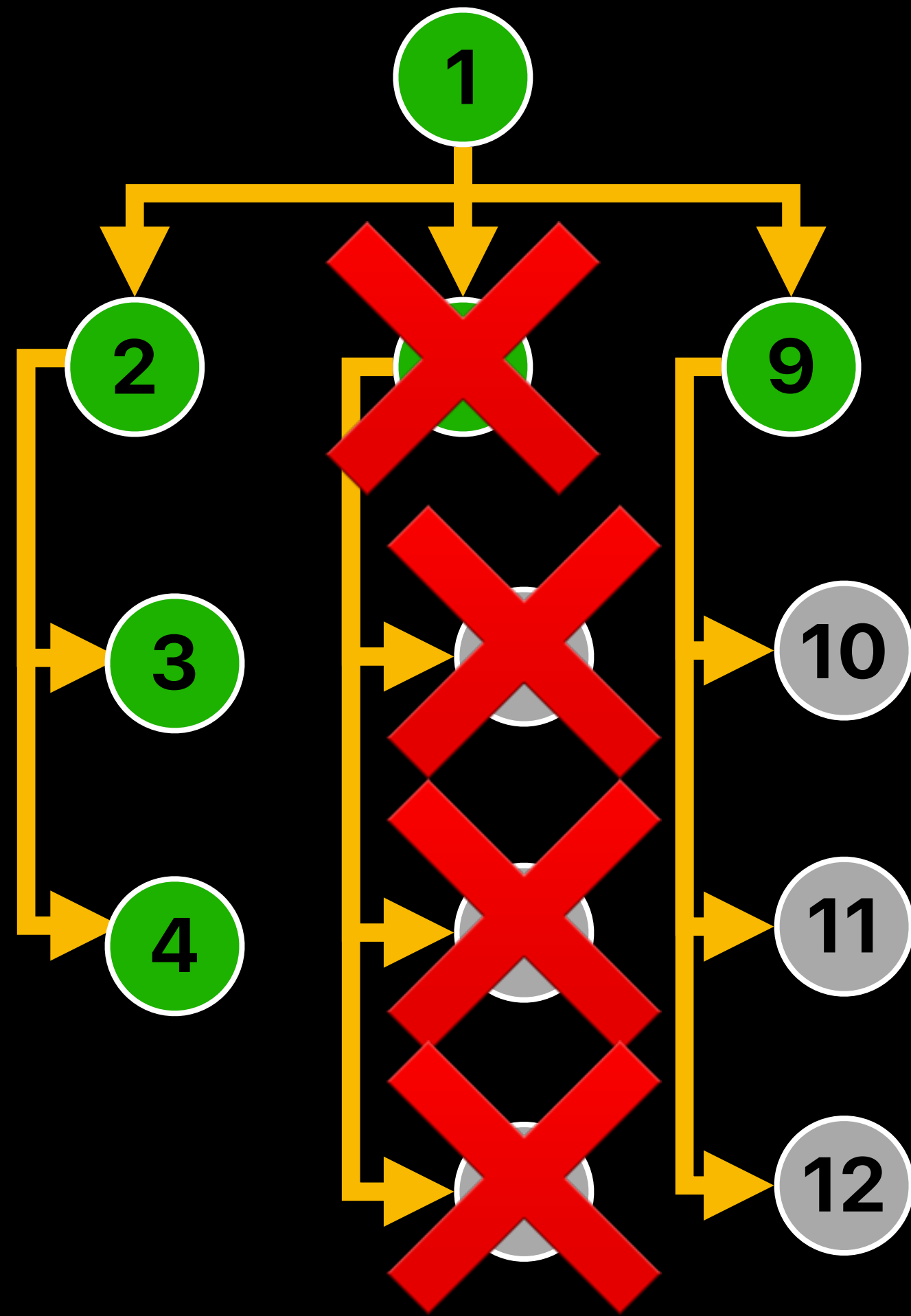
i1	i2	gen
1	2	42
1	5	41
1	9	42
2	3	42
2	4	42
5	6	41
5	7	41
5	8	41
9	10	41
9	11	41
9	12	41

Track live objects



i1	i2	gen
1	2	42
1	5	41
1	9	42
2	3	42
2	4	42
5	6	41
5	7	41
5	8	41
9	10	42
9	11	42
9	12	42

Track live objects



i1	i2	gen
1	2	42
1	5	41
1	9	42
2	3	42
2	4	42
5	6	41
5	7	41
5	8	41
9	10	42
9	11	42
9	12	42

Track live objects

```
CREATE TABLE ooref (i1 INTEGER, i2 INTEGER, gen INTEGER);
```

Track live objects

```
WITH RECURSIVE children(i1, i2) AS (  
    SELECT i1, i2 FROM ooref WHERE i1 = ?1 AND gen != ?2  
    UNION  
    SELECT ooref.i1, ooref.i2 FROM ooref JOIN  
        children ON ooref.i1 = children.i2 AND ooref.gen != ?2  
)  
UPDATE ooref SET gen = ?2  
FROM children  
WHERE ooref.i1 = children.i1 AND ooref.i2 = children.i2;
```

Track live objects

```
WITH RECURSIVE children(i1, i2) AS (  
  SELECT i1, i2 FROM ooref WHERE i1 = ?1 AND gen != ?2  
  UNION  
  SELECT ooref.i1, ooref.i2 FROM ooref JOIN  
    children ON ooref.i1 = children.i2 AND ooref.gen != ?2  
)  
UPDATE ooref SET gen = ?2  
FROM children  
WHERE ooref.i1 = children.i1 AND ooref.i2 = children.i2;
```

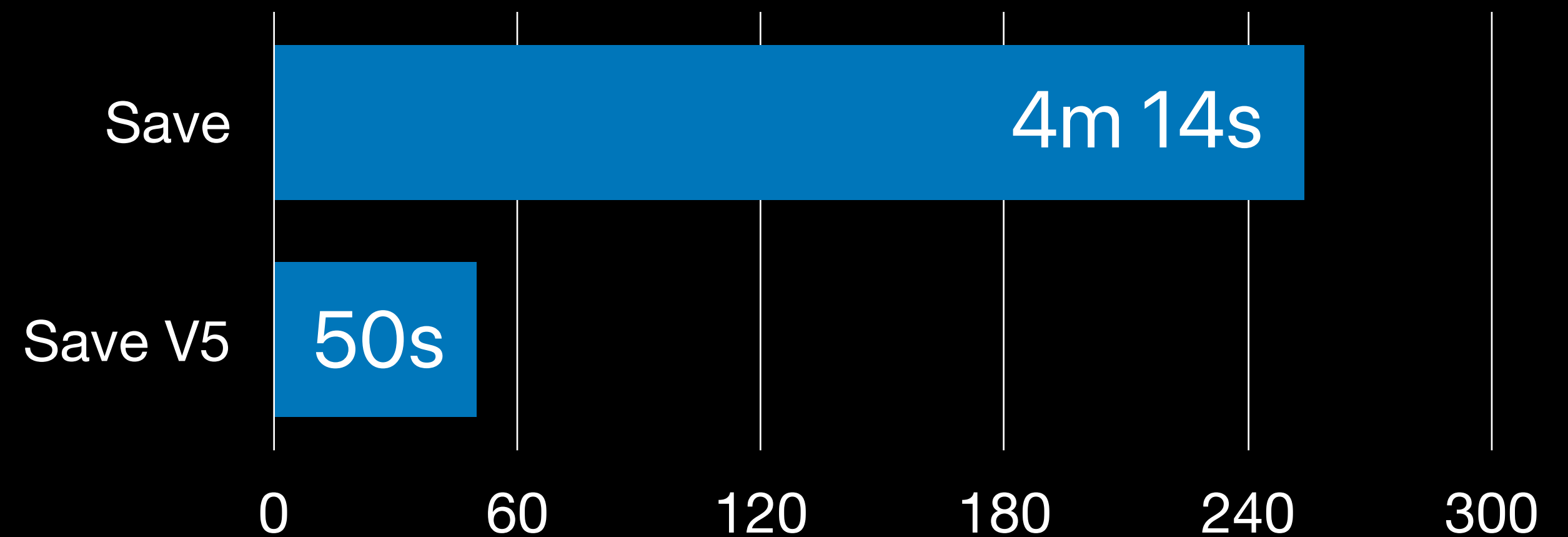
Was it worth it?

Improving save time



large website.sparkle

- 35 GB
- 300 pages
- 100k words
- 800 images
- 230 videos
- 400 attached files

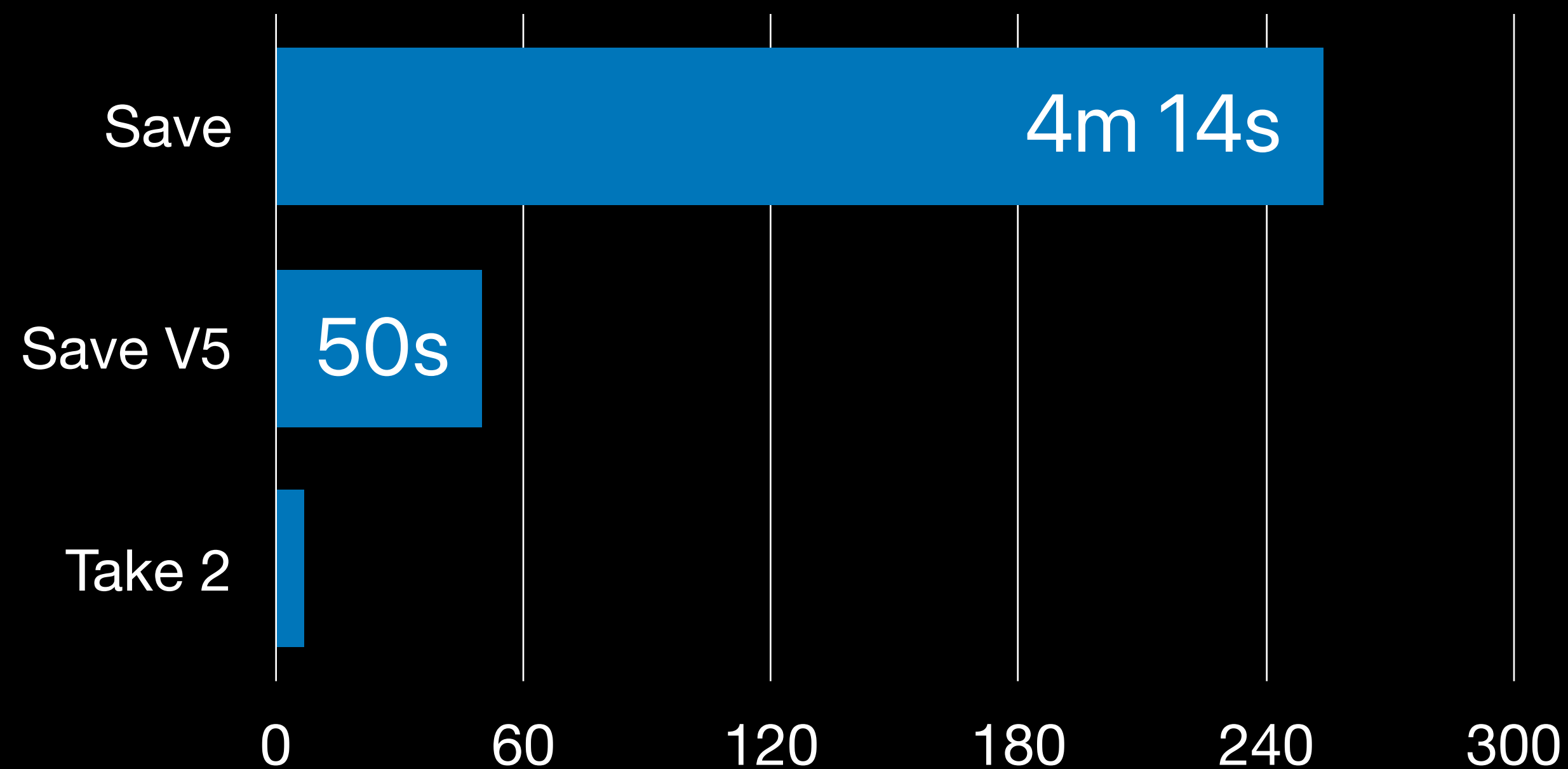


Improving save time



large website.sparkle

- 35 GB
- 300 pages
- 100k words
- 800 images
- 230 videos
- 400 attached files

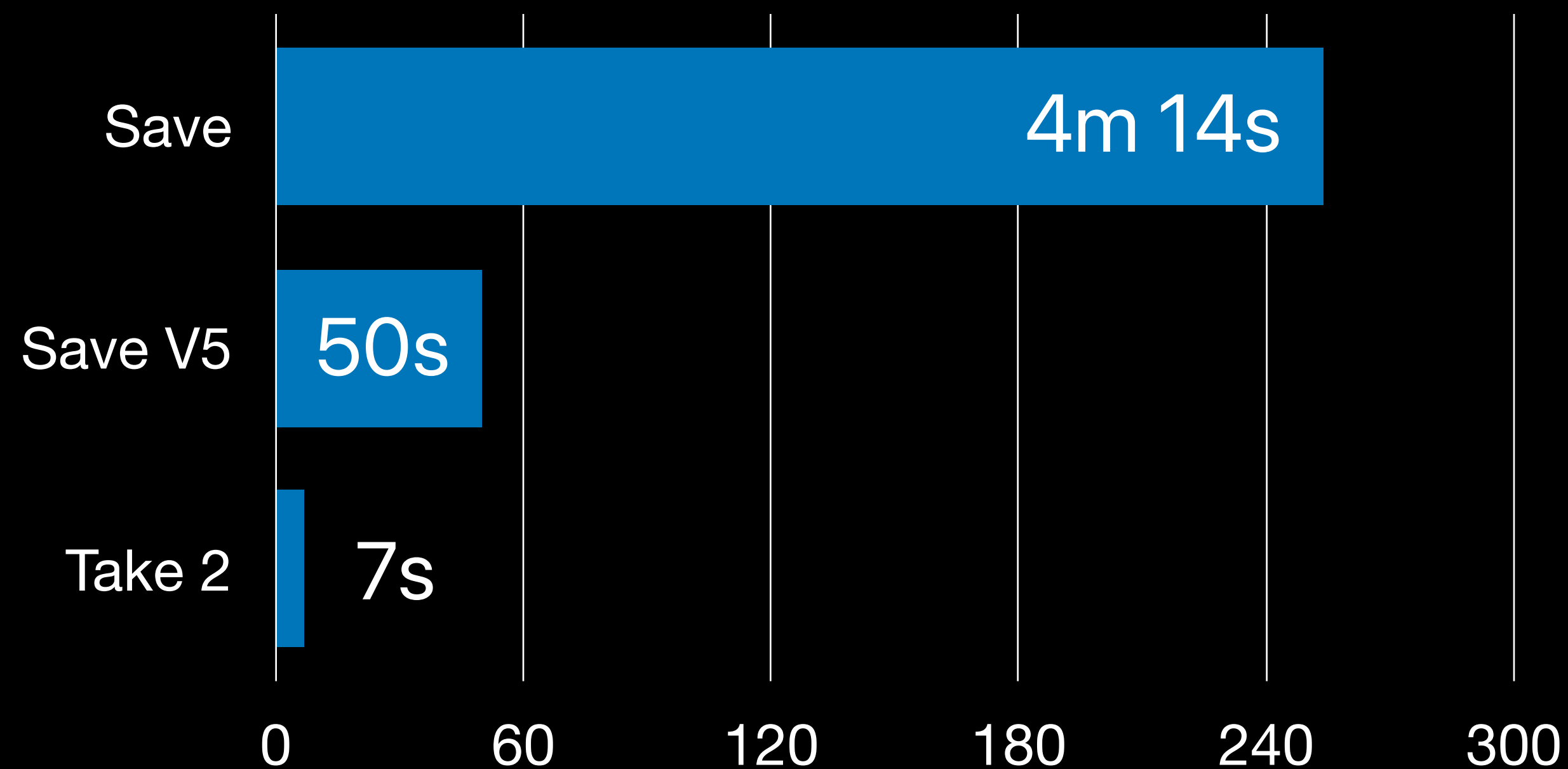


Improving save time



large website.sparkle

- 35 GB
- 300 pages
- 100k words
- 800 images
- 230 videos
- 400 attached files

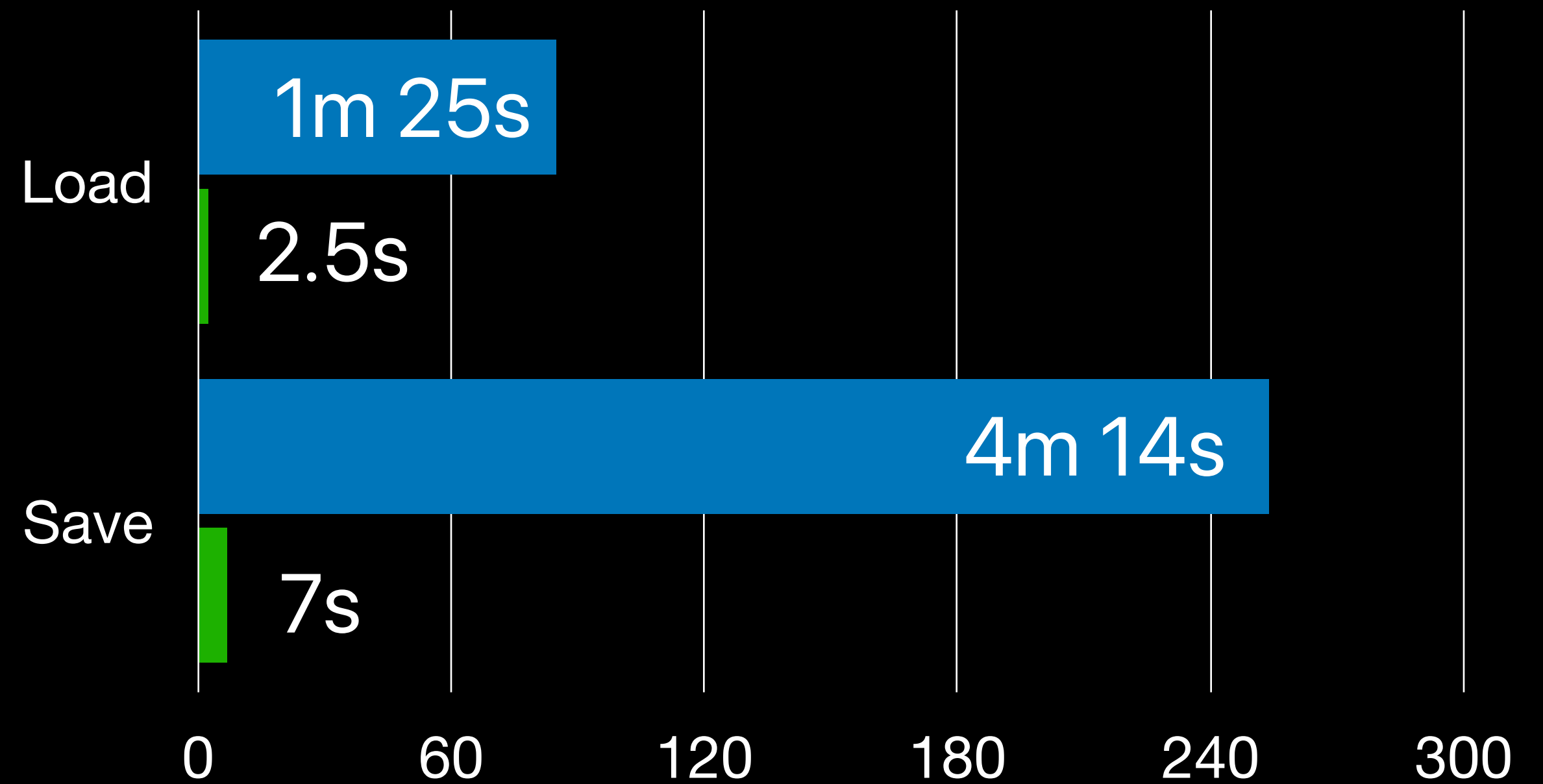


Improving save time



large website.sparkle


- 35 GB
- 300 pages
- 100k words
- 800 images
- 230 videos
- 400 attached files



36000%

Thank You!

My code

A chaotic street scene with a car flying through the air. In the foreground, a man with glasses and a black shirt stands with his arms crossed, looking towards the camera. The background shows a city street with buildings, a white van, a motorcycle, and a car flying through the air. The sky is blue with some clouds.

Me giving
programming advice