Agni: An Efficient Dual-access File System over Object Storage

Kunal Lillaney, Vasily Tarasov, David Pease, Randal Burns

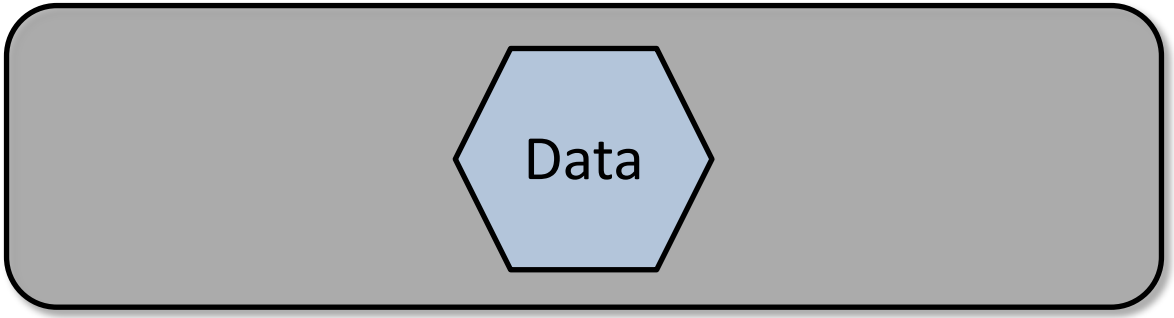
JOHNS HOPKINS UNIVERSITY

IBM Research

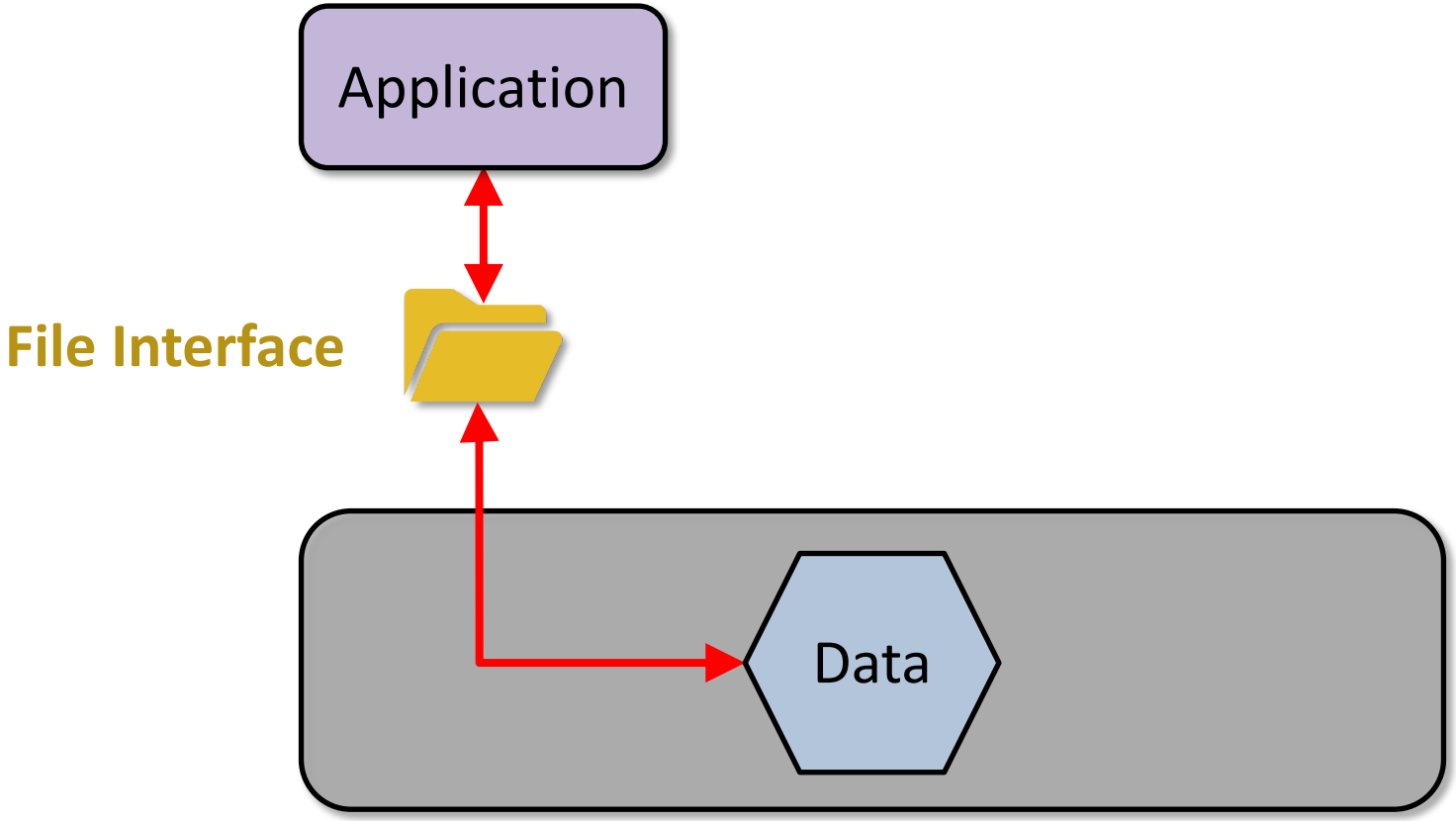
23 November, SoCC'19 - Santa Cruz

What is Dual-Access?

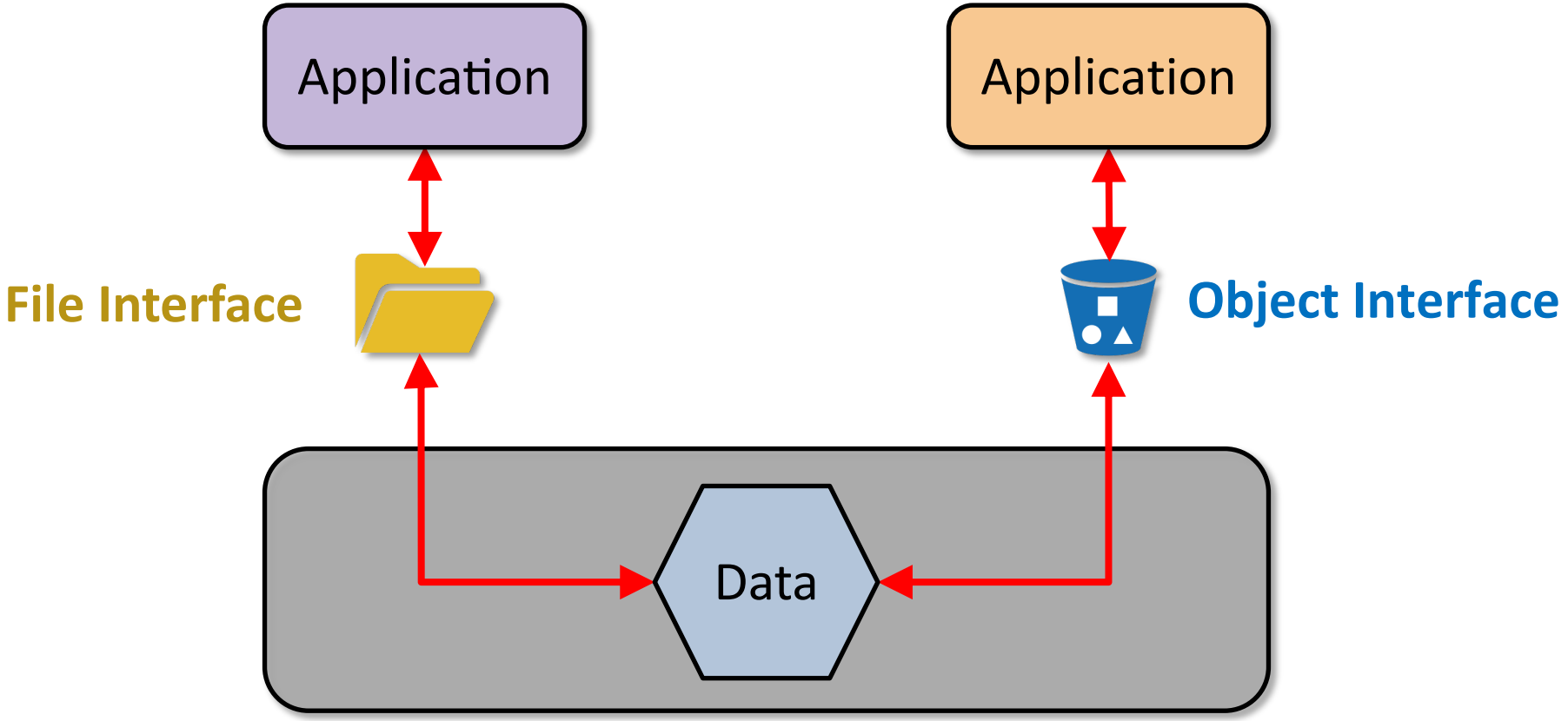
Dual Access



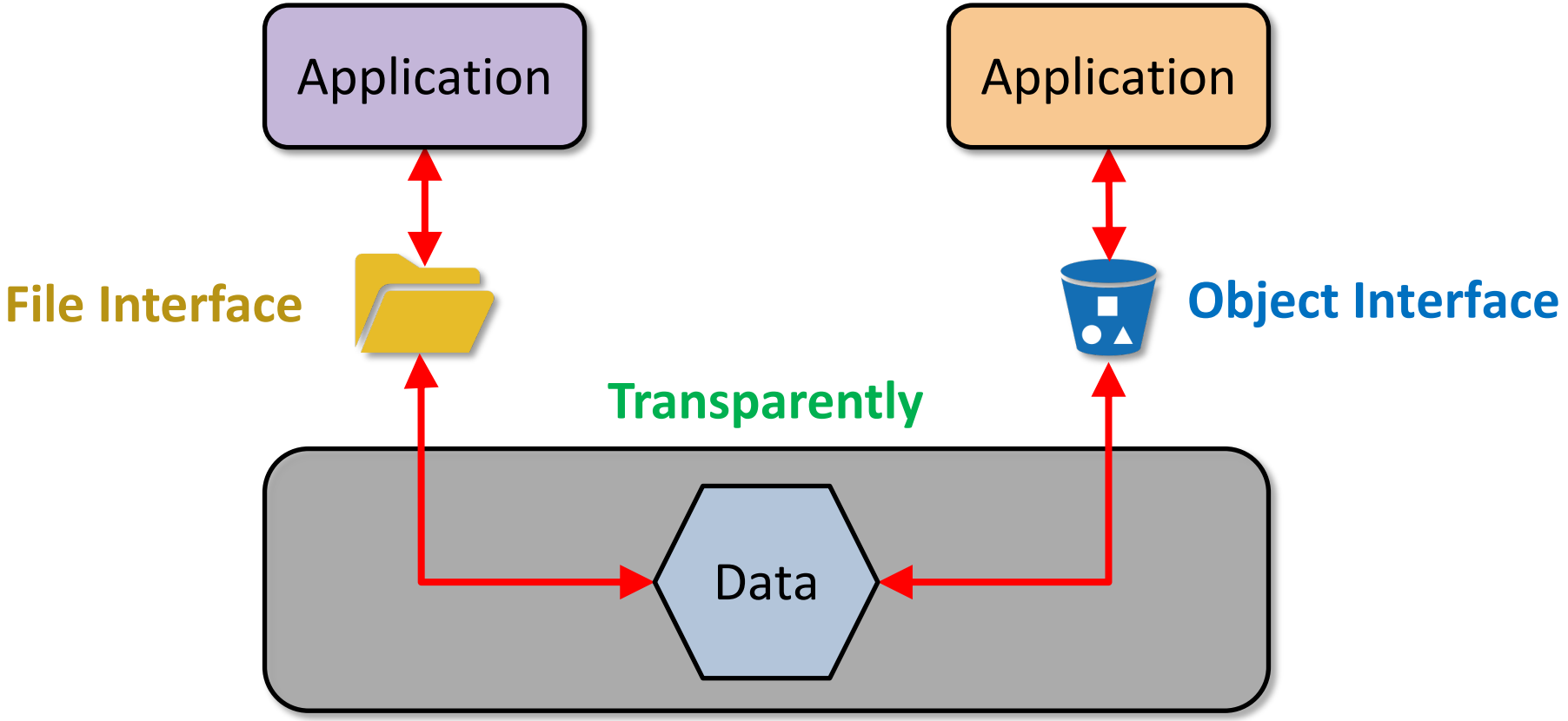
Dual Access

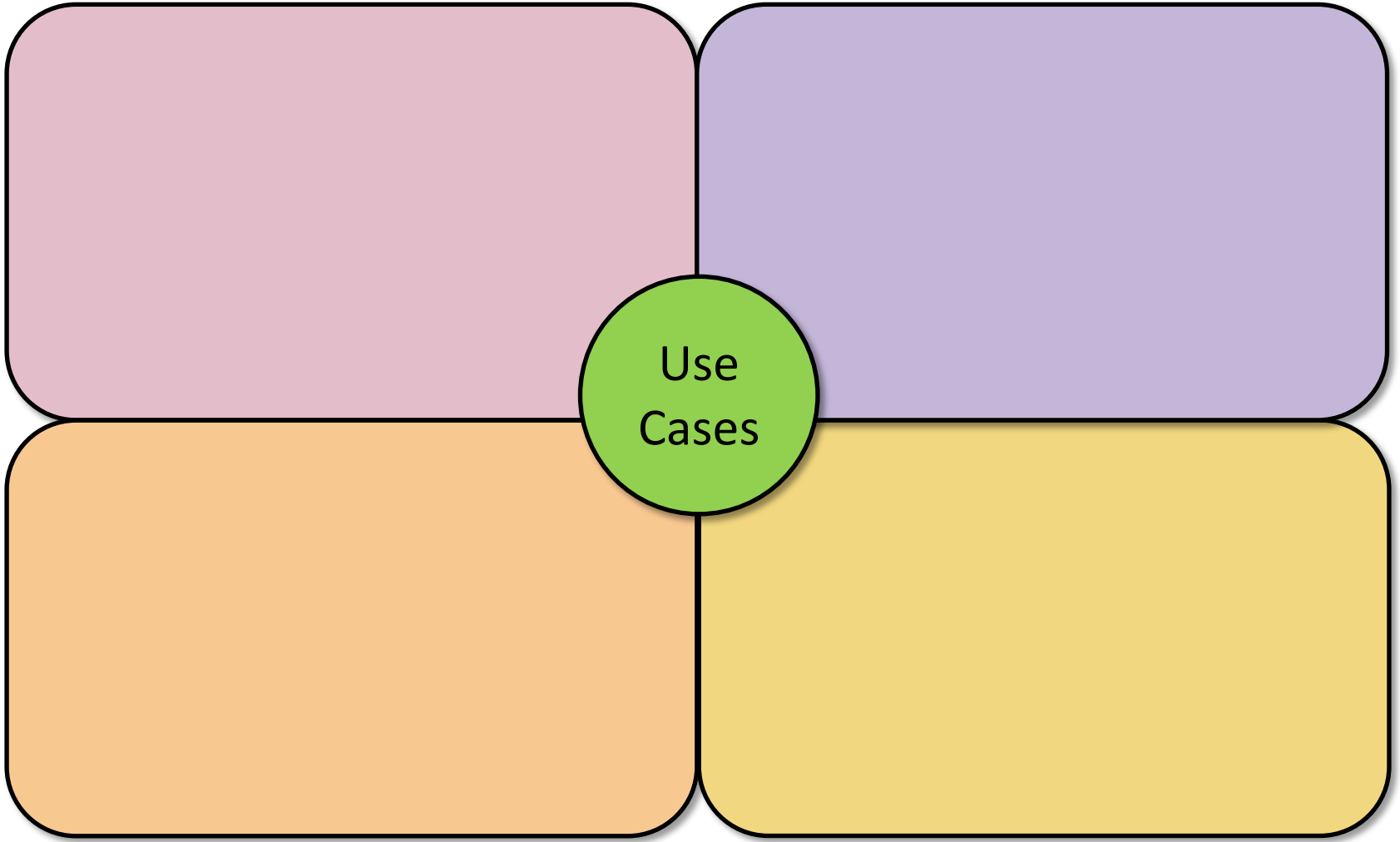


Dual Access


Application
Application
File Interface
Object Interface
Data

Dual Access


Application
Application
File Interface
Object Interface
Transparently
Data


Use
Cases

Media

Use Cases

Media

Use Cases

Life Science

Media

Use Cases

Life Science

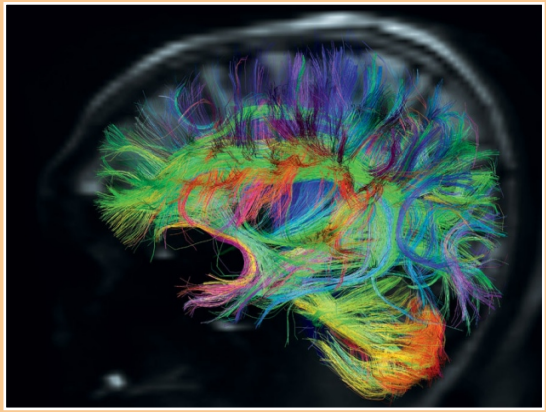
Geo-Informatics

Use Cases

Media

Life Science

Neuroscience



Geo-Informatics

Media Transcoding, Editing, Analytics

Media Transcoding, Editing, Analytics

Object Interface

Media Transcoding, Editing, Analytics

pinnacle

COREL™

FFmpeg

Ps Adobe Photoshop

File Interface

Object Interface

Media Transcoding, Editing, Analytics

pinnacle

COREL™

FFmpeg

Ps Adobe Photoshop

File Interface

Object Interface

Object Interface

File Systems vs Object Storage

File Systems vs Object Storage

Partial Writes

File Systems vs Object Storage

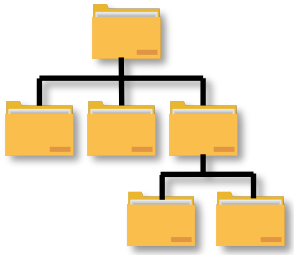
Partial Writes

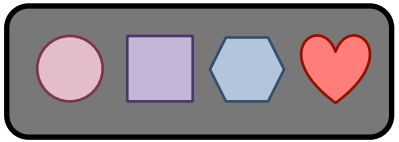
Namespace

File Systems vs Object Storage

Partial Writes

Namespace

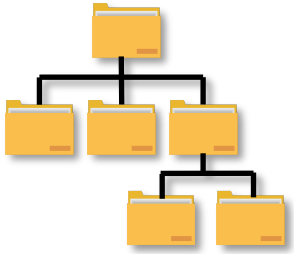


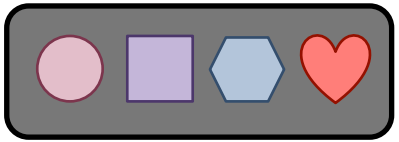


File Systems vs Object Storage

Partial Writes

Namespace



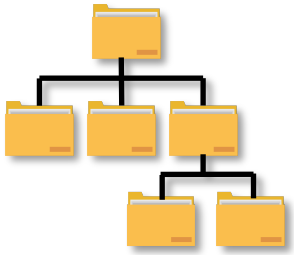


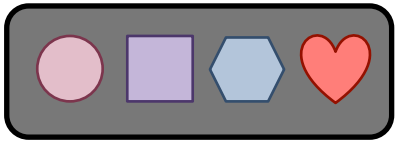
Interfaces

File Systems vs Object Storage

Partial Writes

Namespace





Interfaces

Outline

- Design considerations
- Existing systems
- Agni

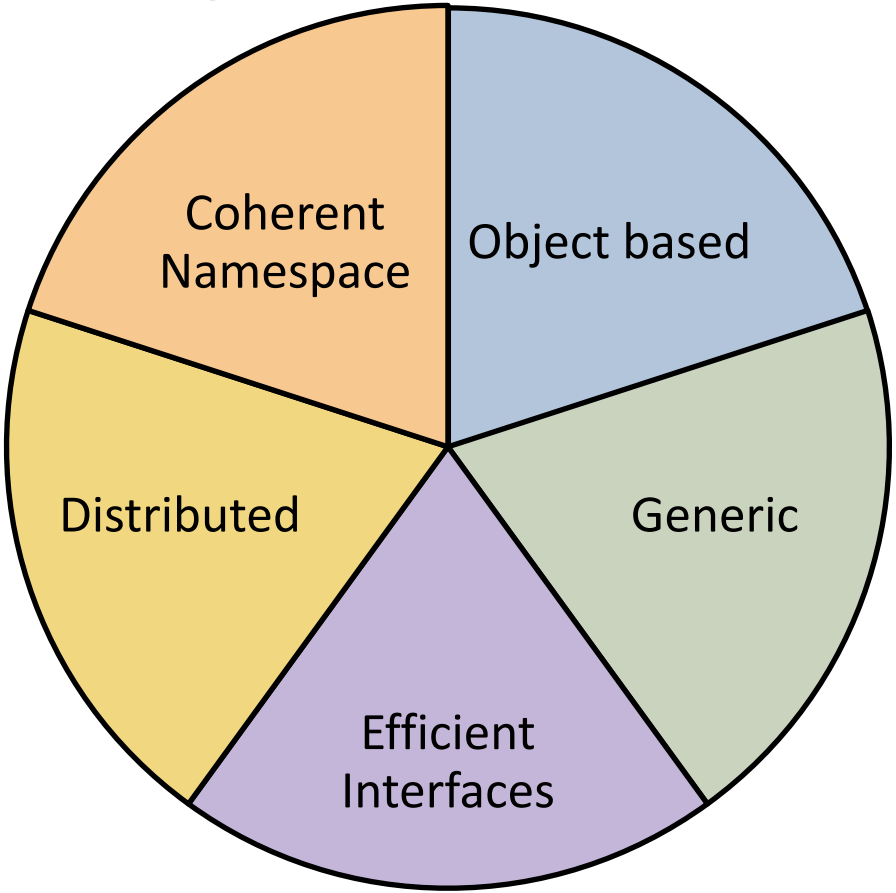
https://www.greenbiz.com/sites/default/files/styles/gbz_article_primary_breakpoints_kalapicture_screenmd_1x/public/images/articles/featured/datacenter_0.jpg?itok=iJm7ezgB&timestamp=1483504030

Outline

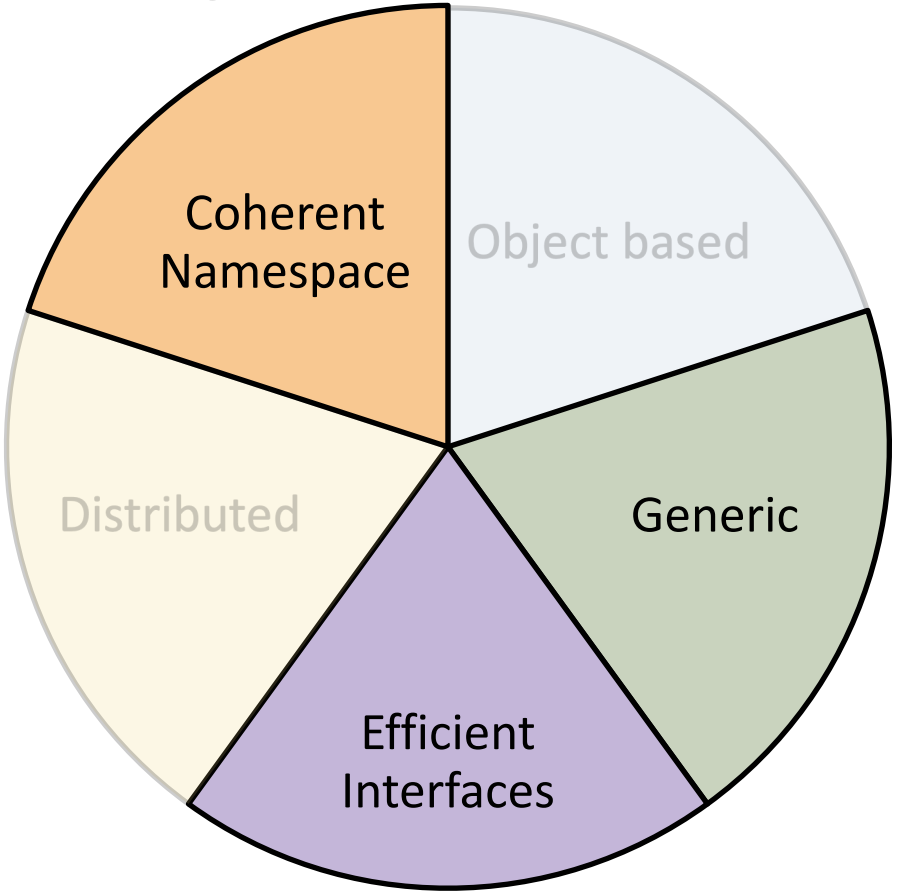
- Design considerations
- Existing systems
- Agni

https://www.greenbiz.com/sites/default/files/styles/gbz_article_primary_breakpoints_kalapicture_screenmd_1x/public/images/articles/featured/datacenter_0.jpg?itok=iJm7ezgB&timestamp=1483504030

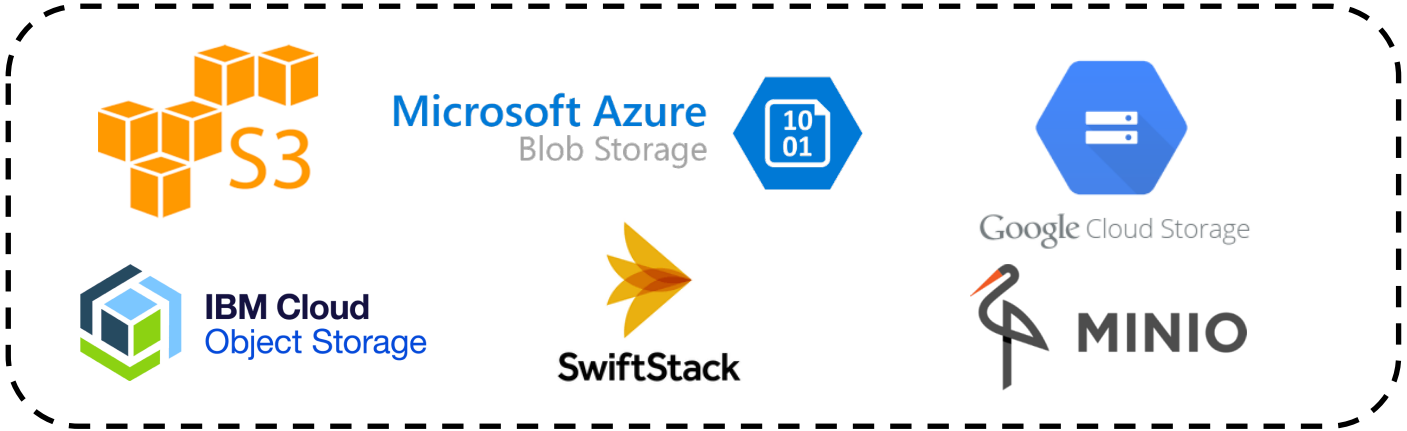
Design Considerations



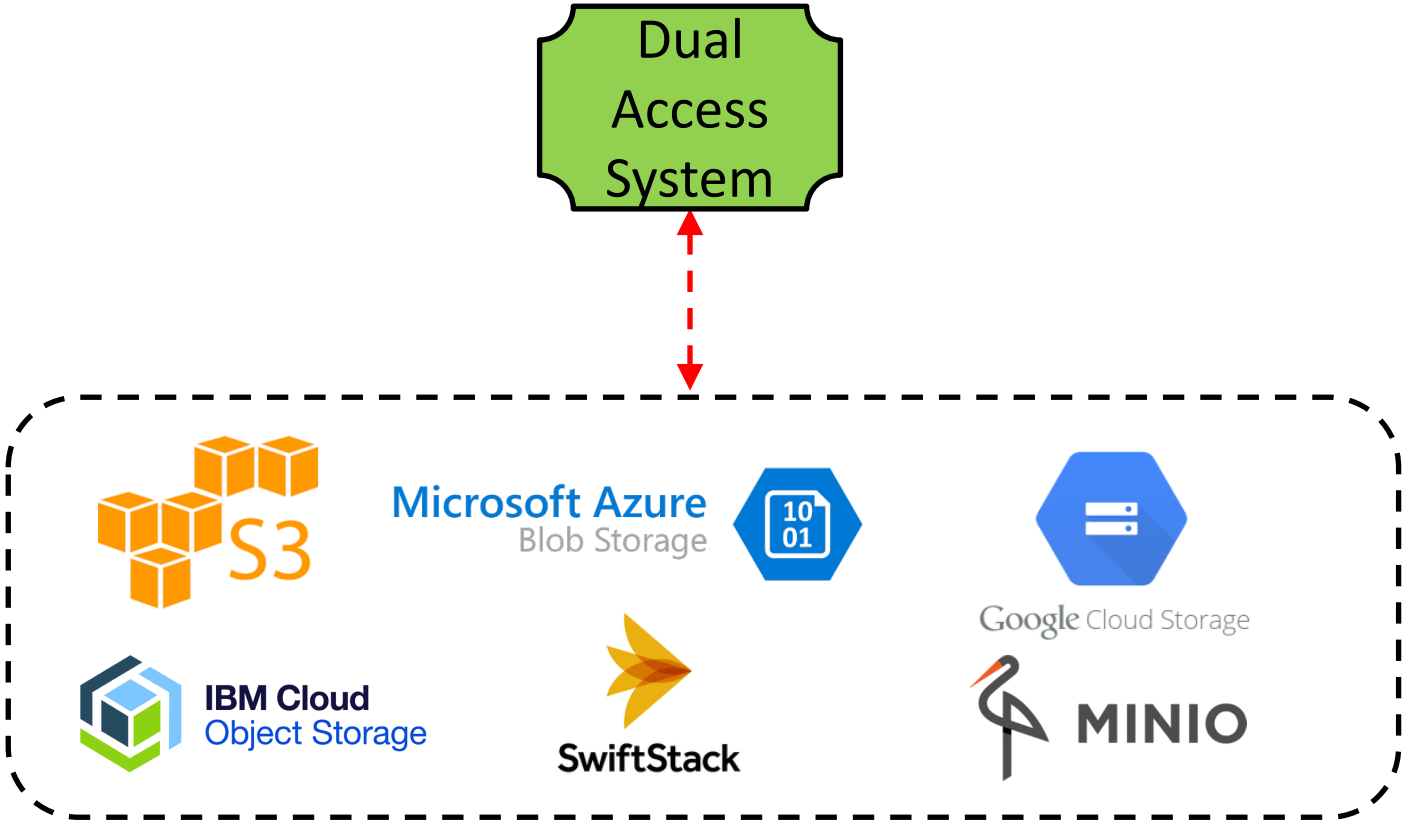
Design Considerations



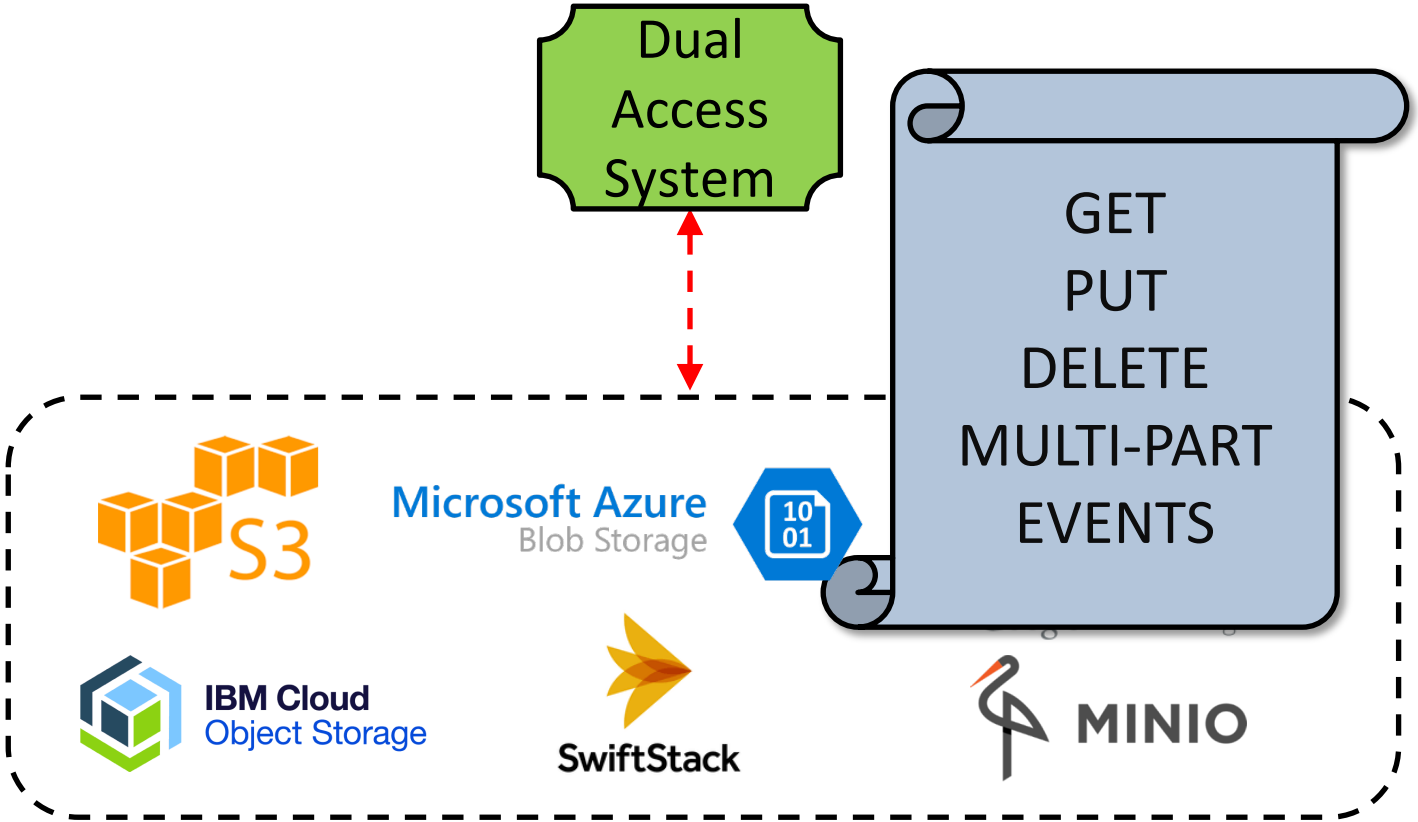
Generic → Object Store Agnostic



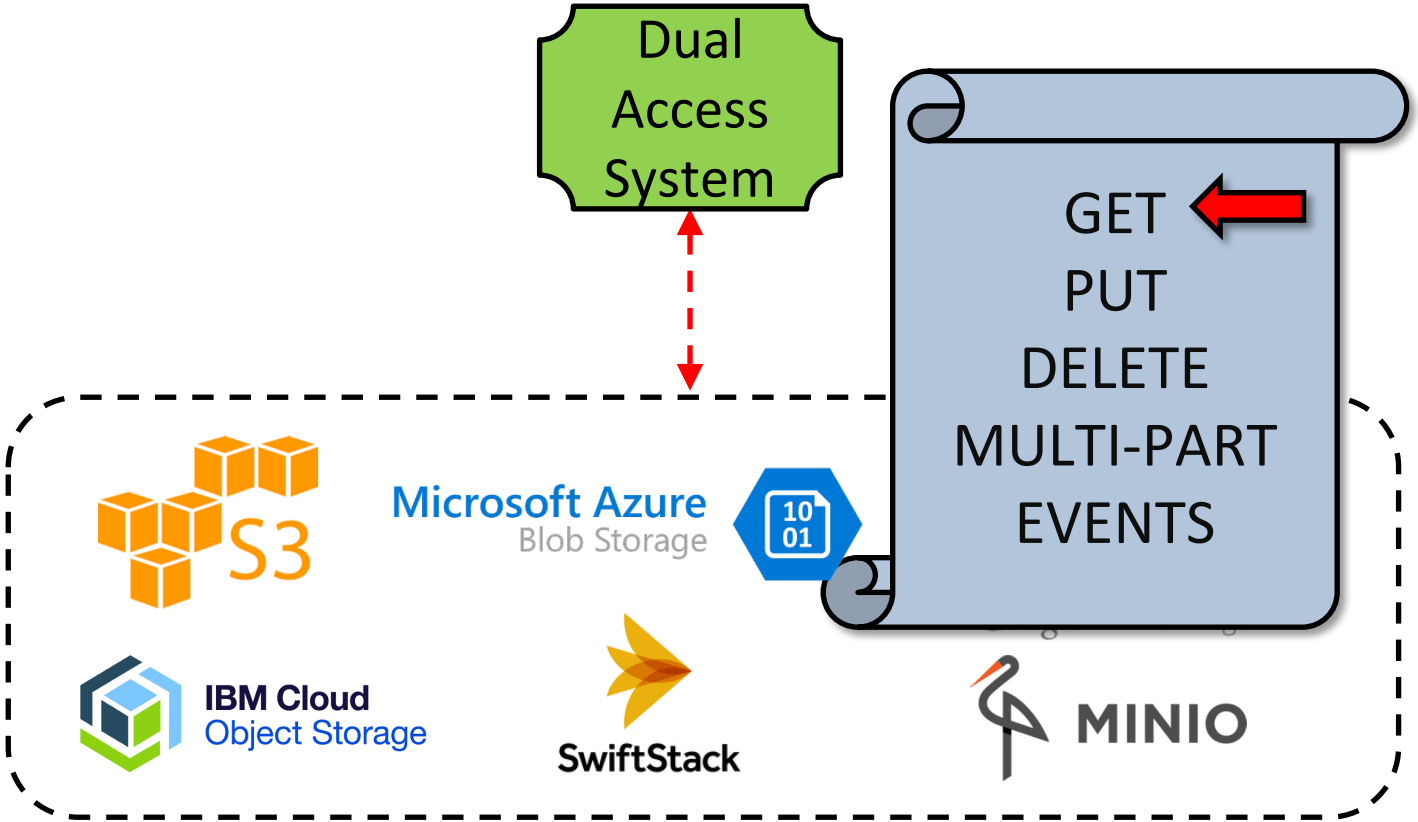
Generic → Object Store Agnostic



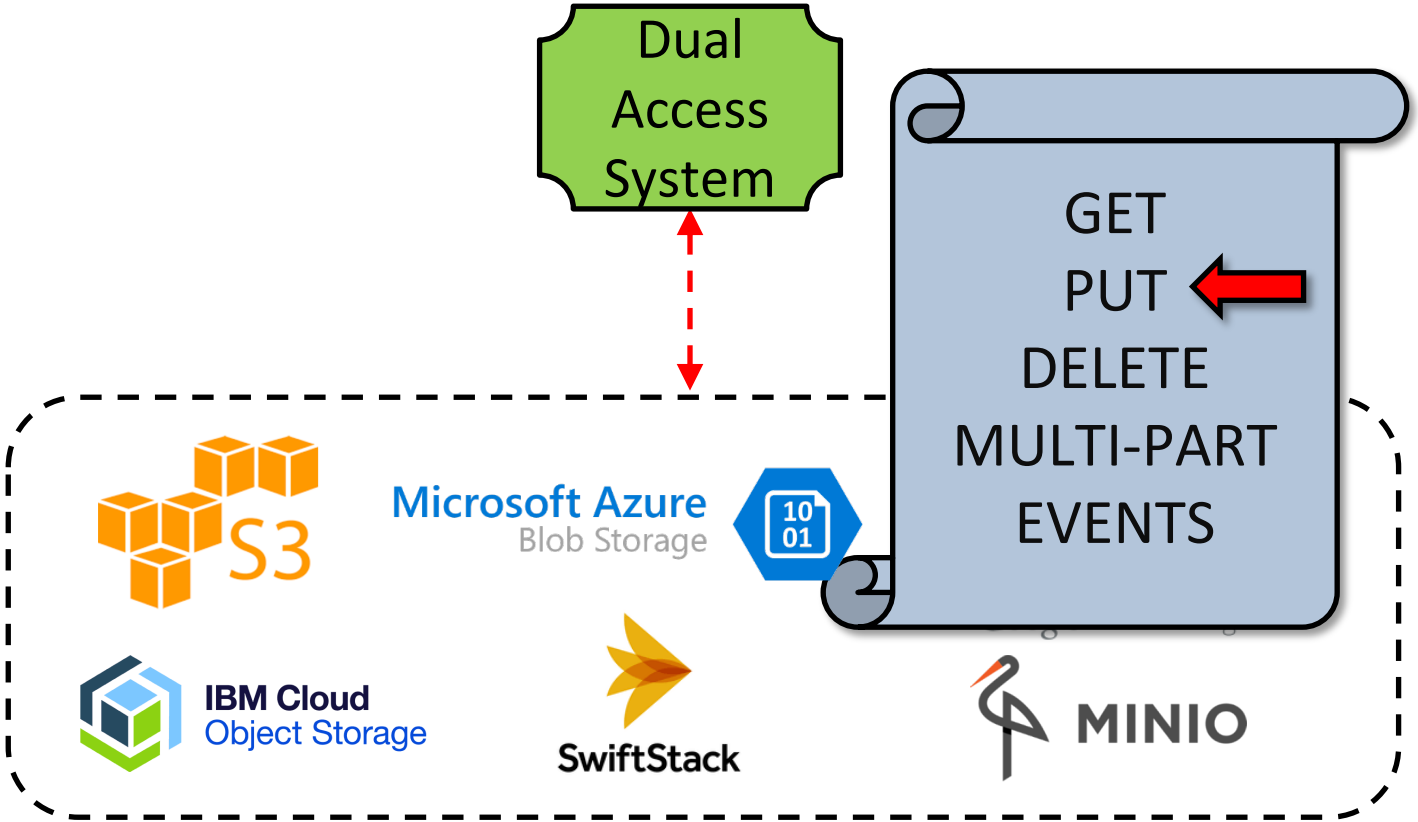
Generic → Object Store Agnostic



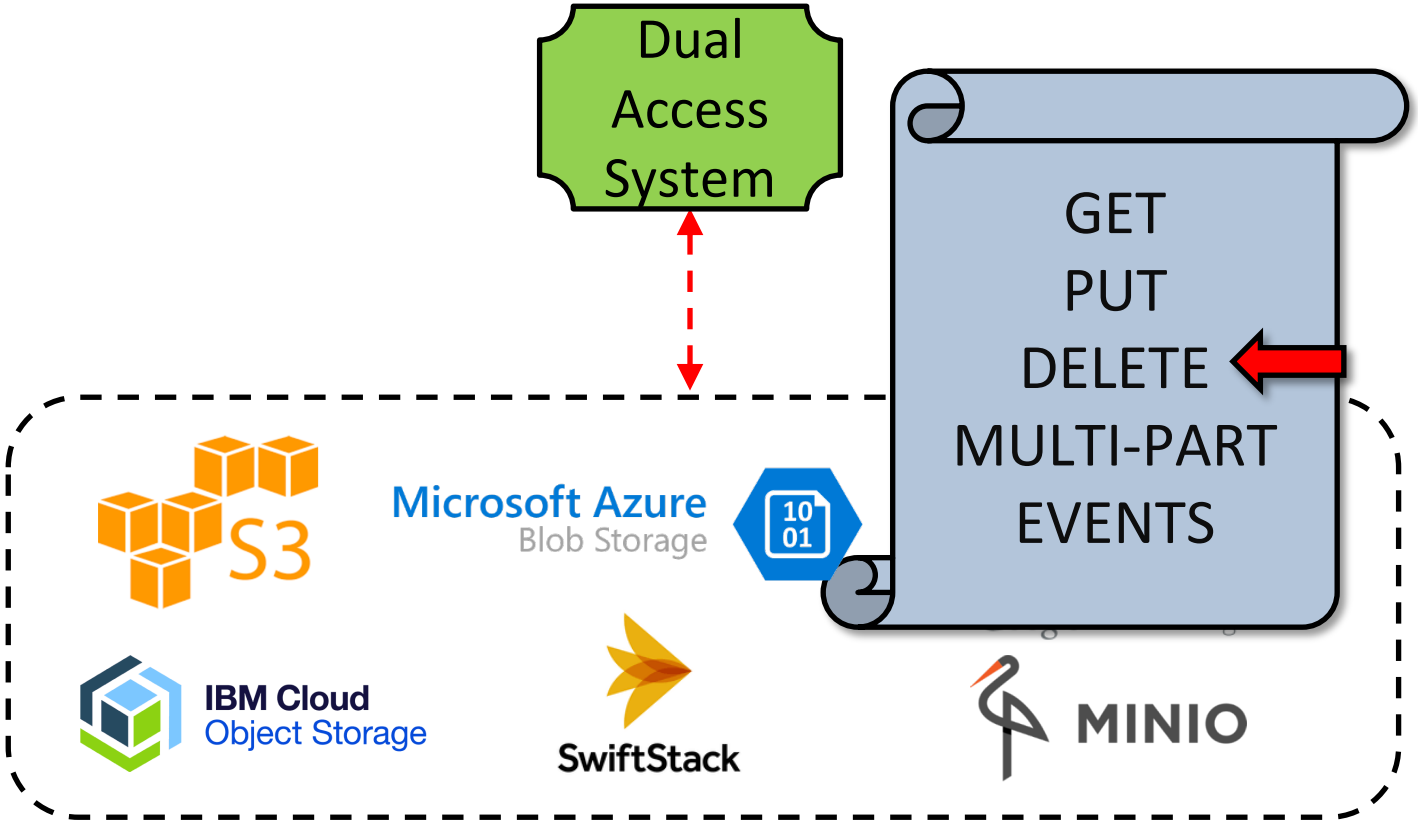
Generic → Object Store Agnostic



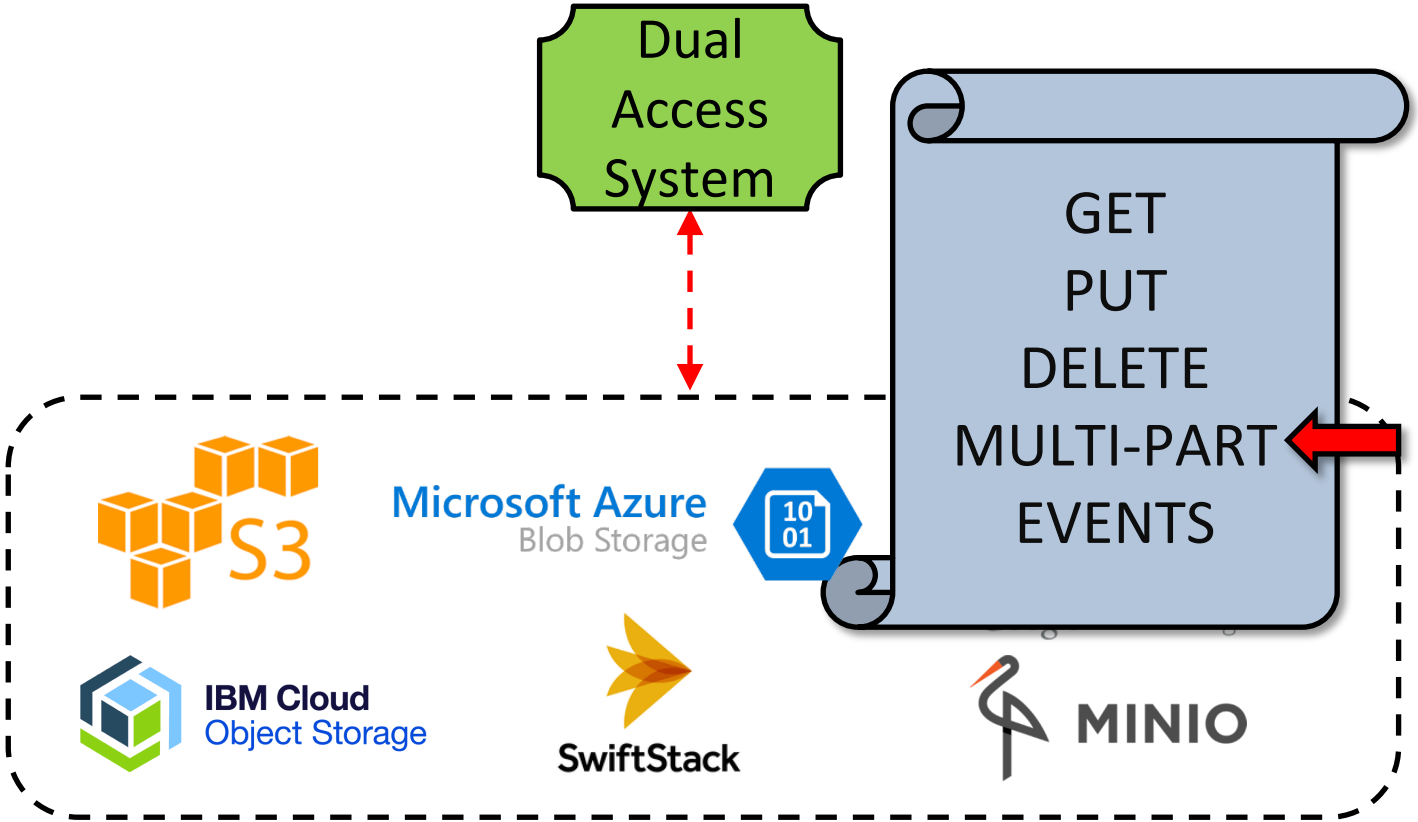
Generic → Object Store Agnostic



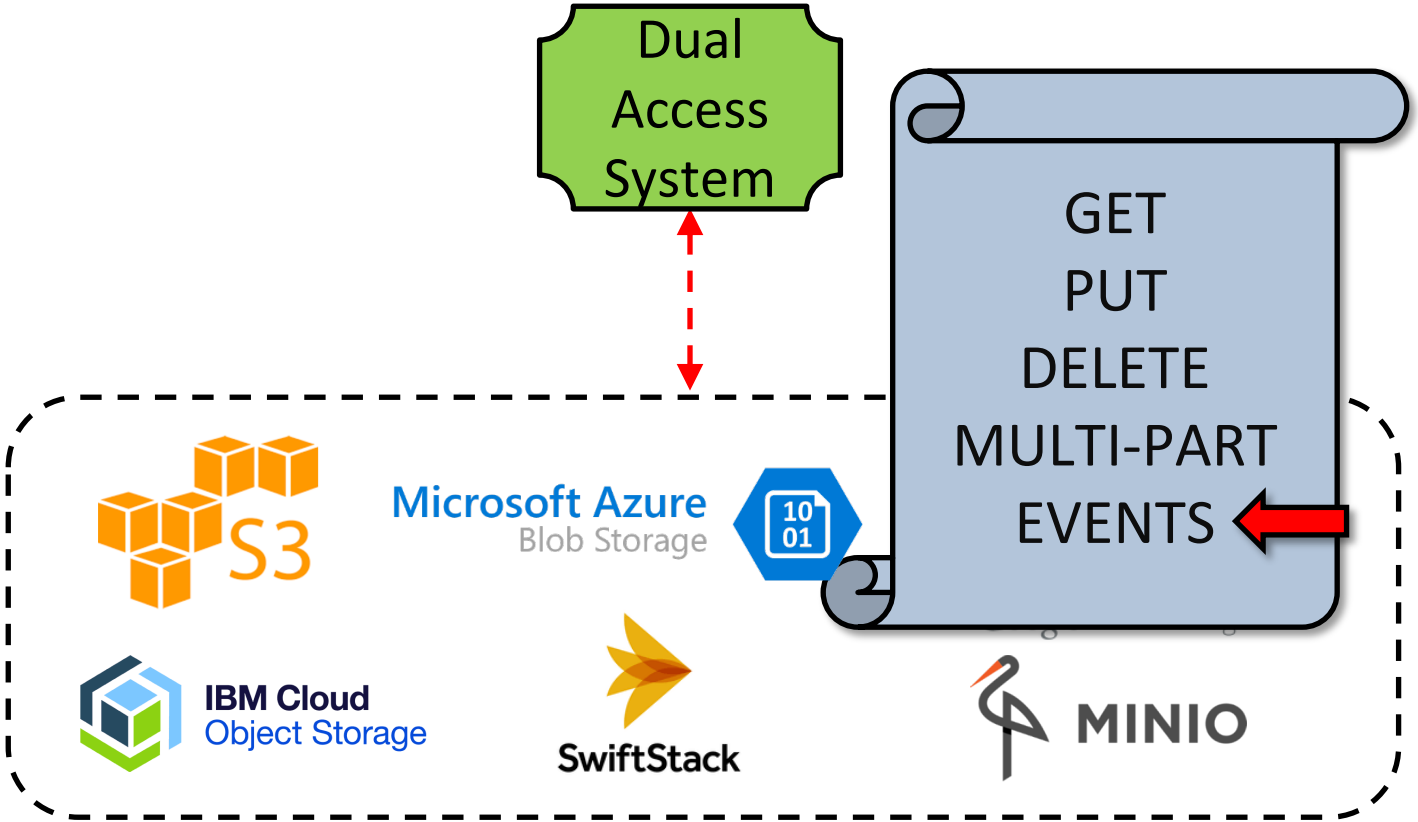
Generic → Object Store Agnostic



Generic → Object Store Agnostic

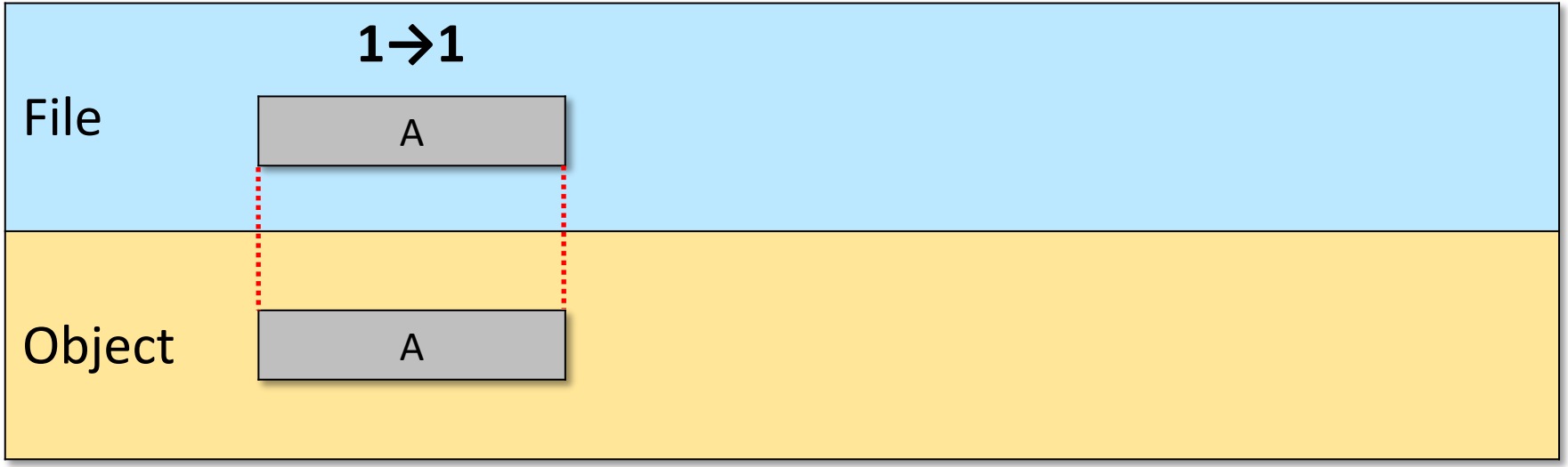


Generic → Object Store Agnostic

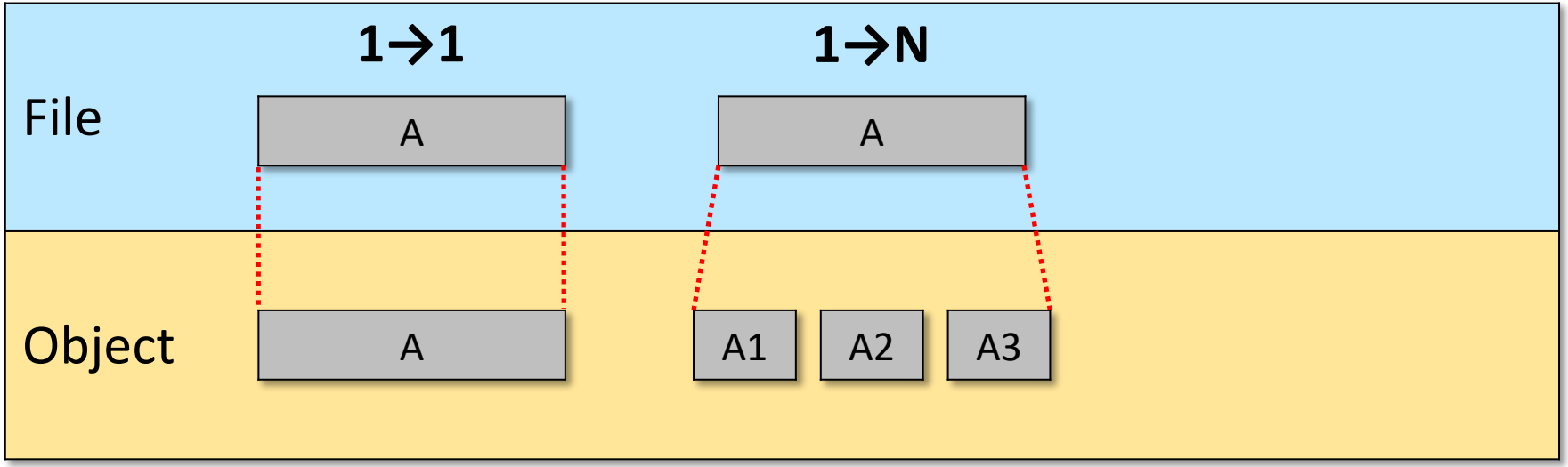


Dual Access: File to Object Mapping

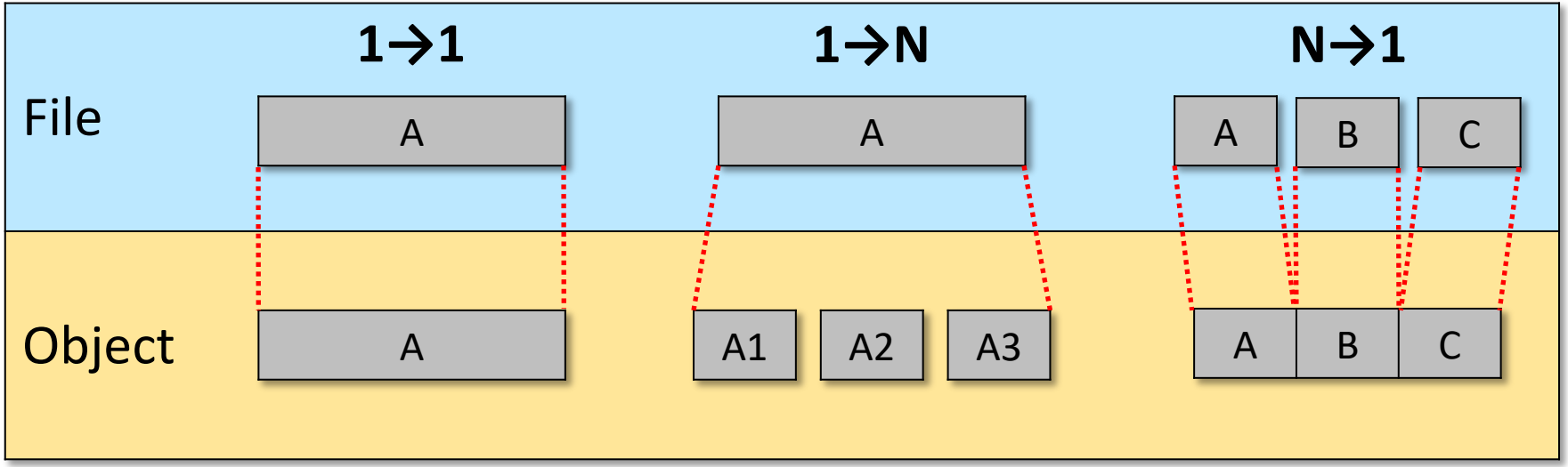
Dual Access: File to Object Mapping



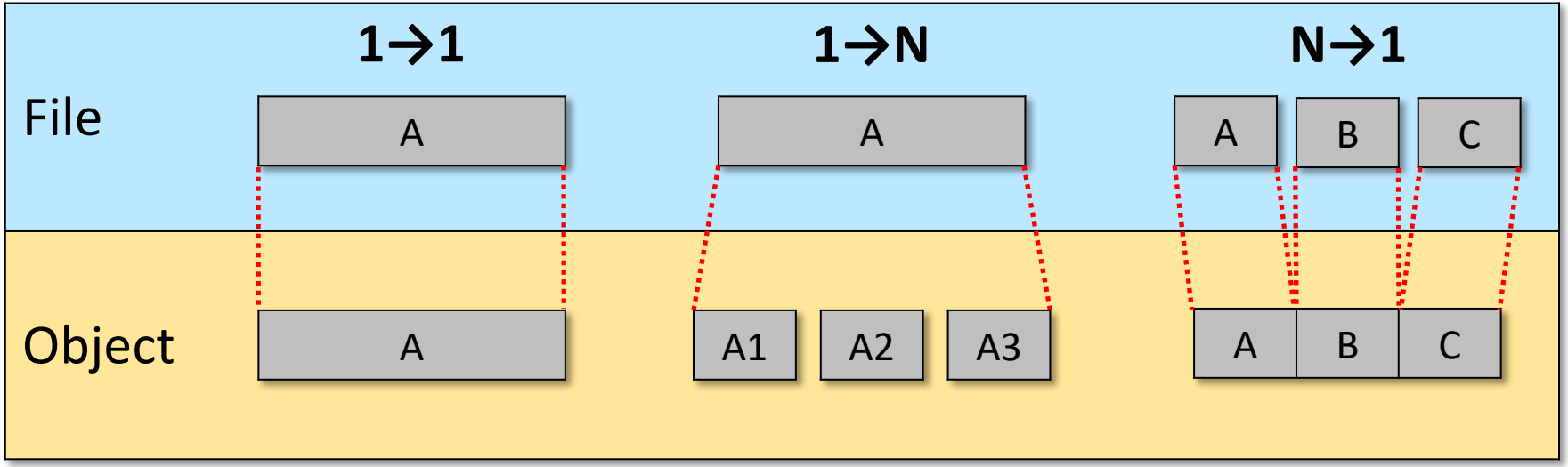
Dual Access: File to Object Mapping



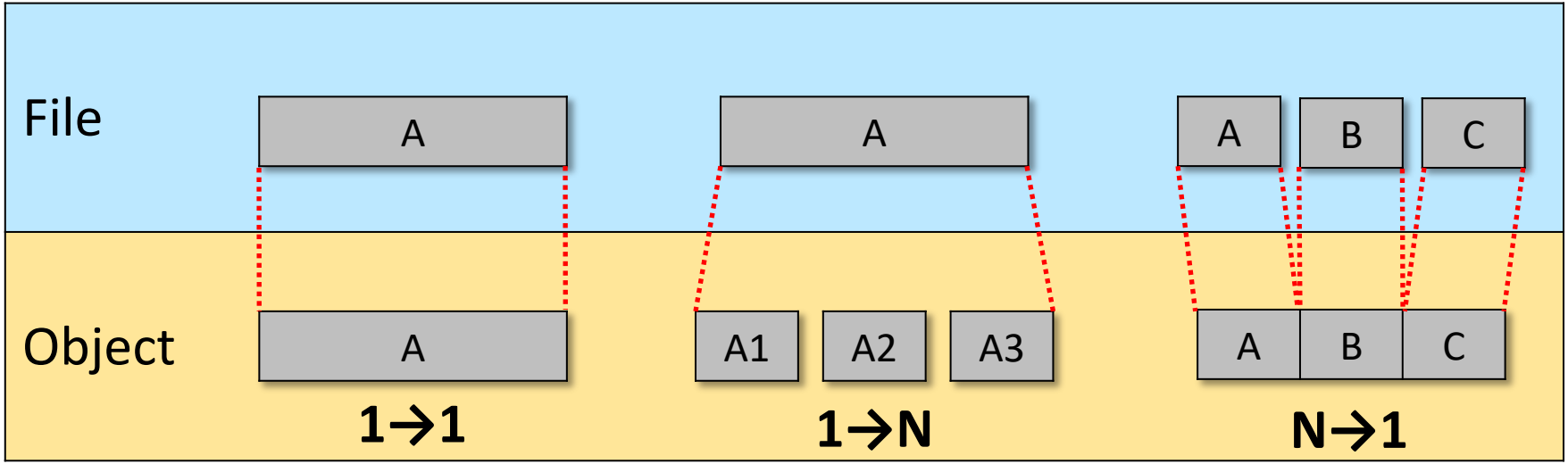
Dual Access: File to Object Mapping



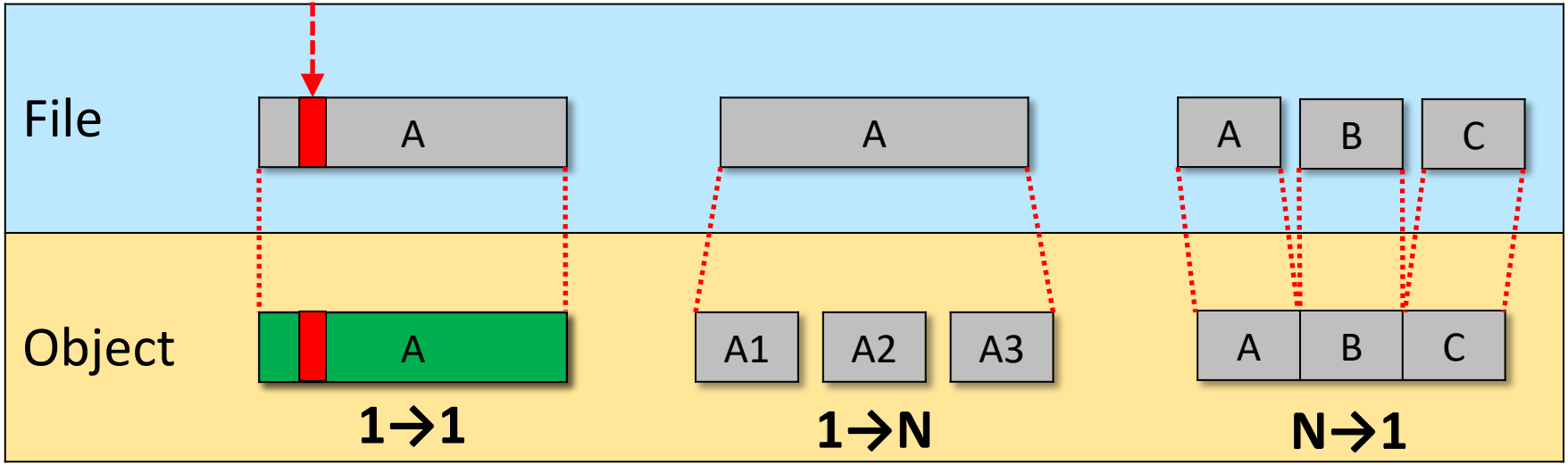
Dual Access: File to Object Mapping



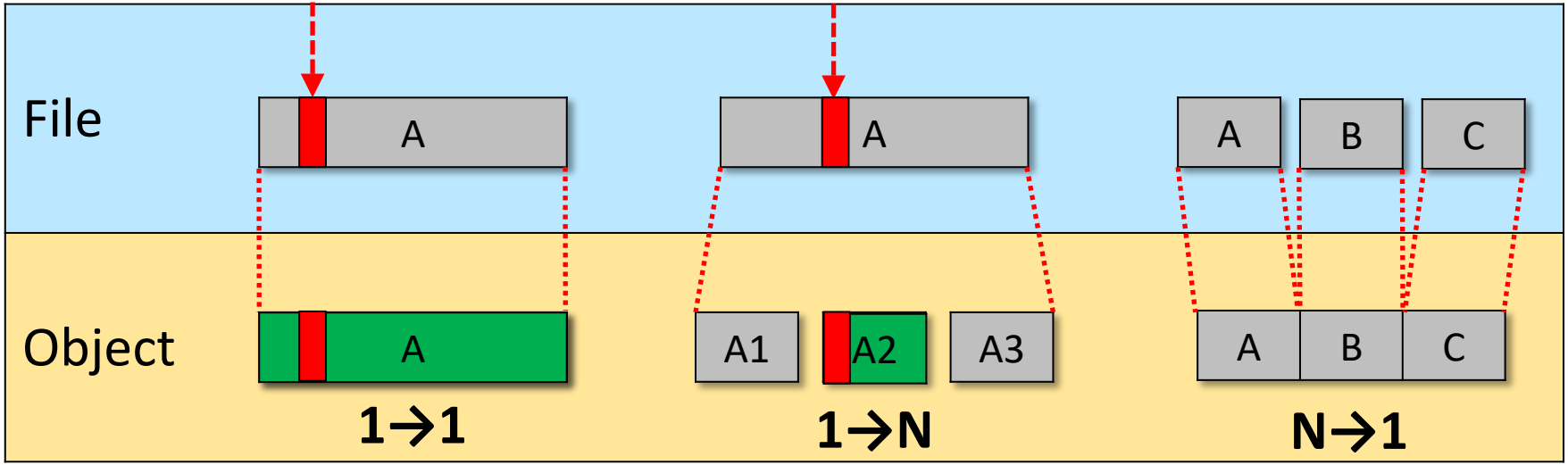
Efficiency: File to Object Mapping



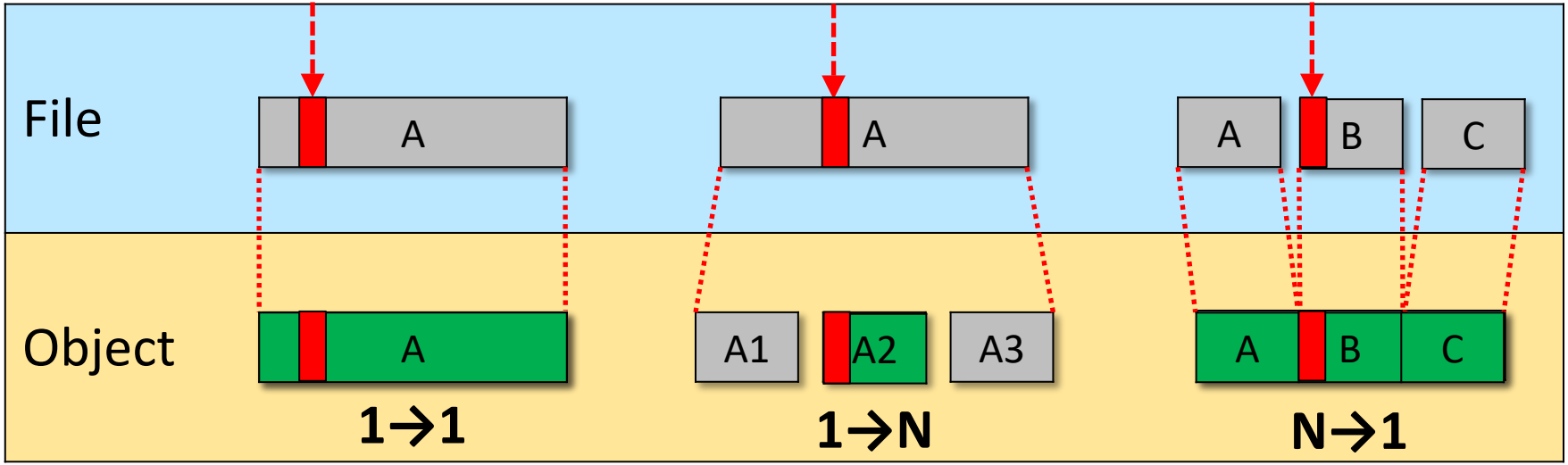
Efficiency: File to Object Mapping



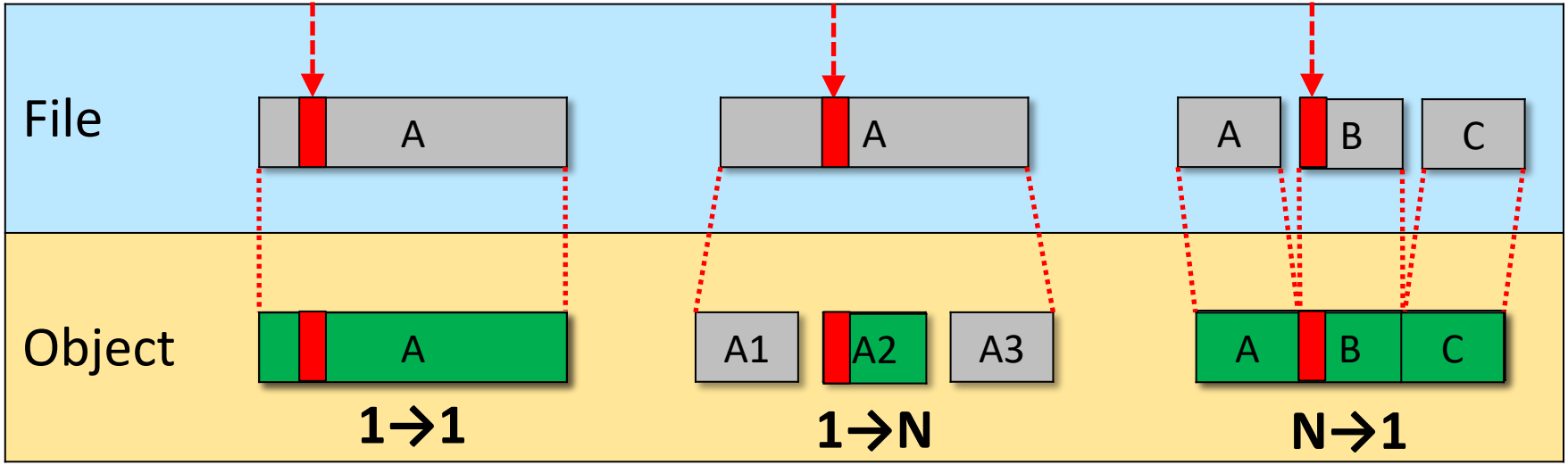
Efficiency: File to Object Mapping



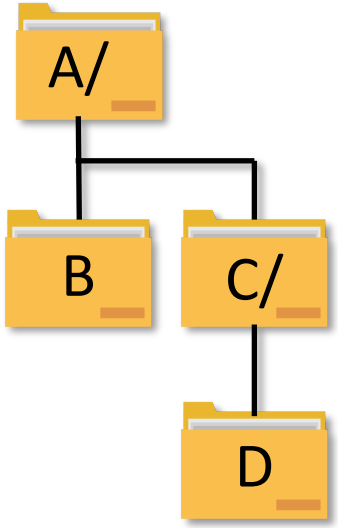
Efficiency: File to Object Mapping



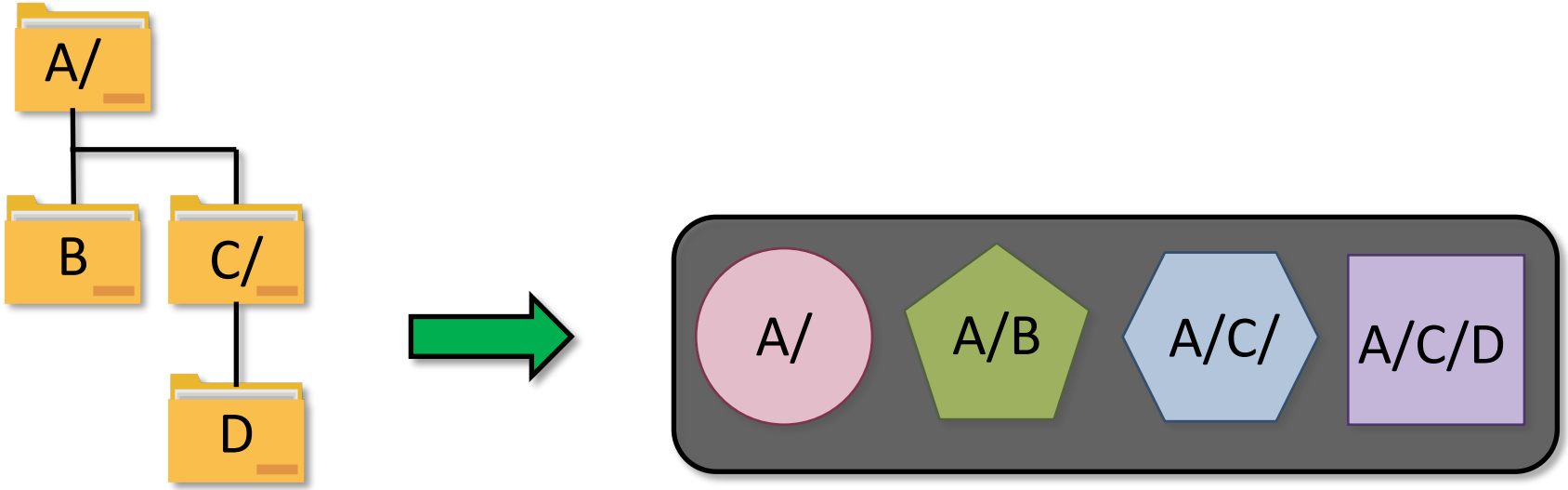
Efficiency: File to Object Mapping



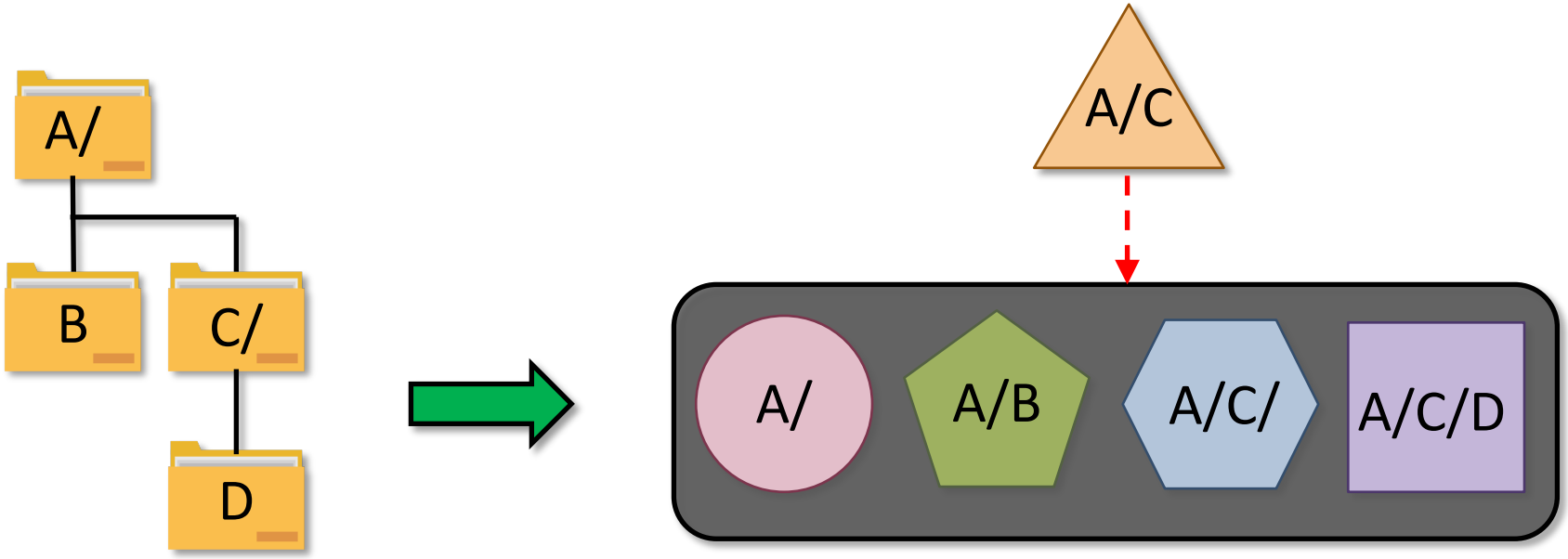
Namespace: Example of Incoherency



Namespace: Example of Incoherency



Namespace: Example of Incoherency

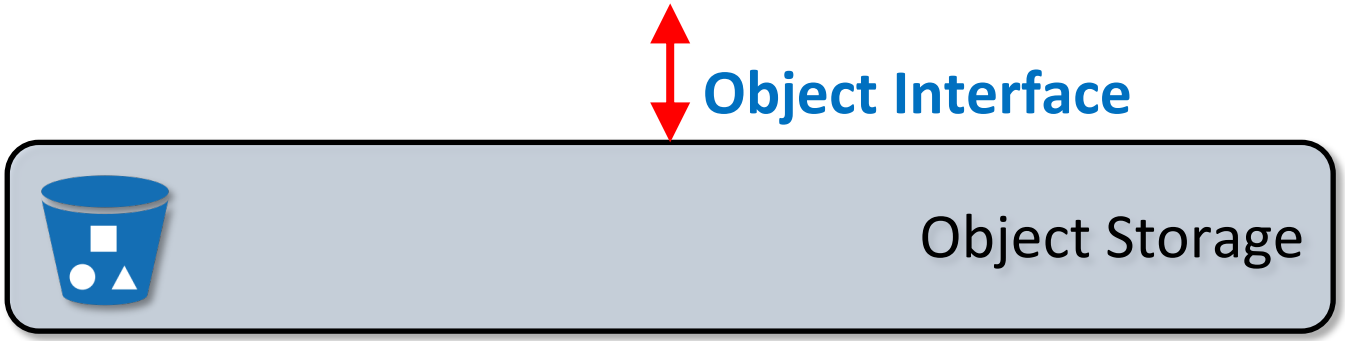


Outline

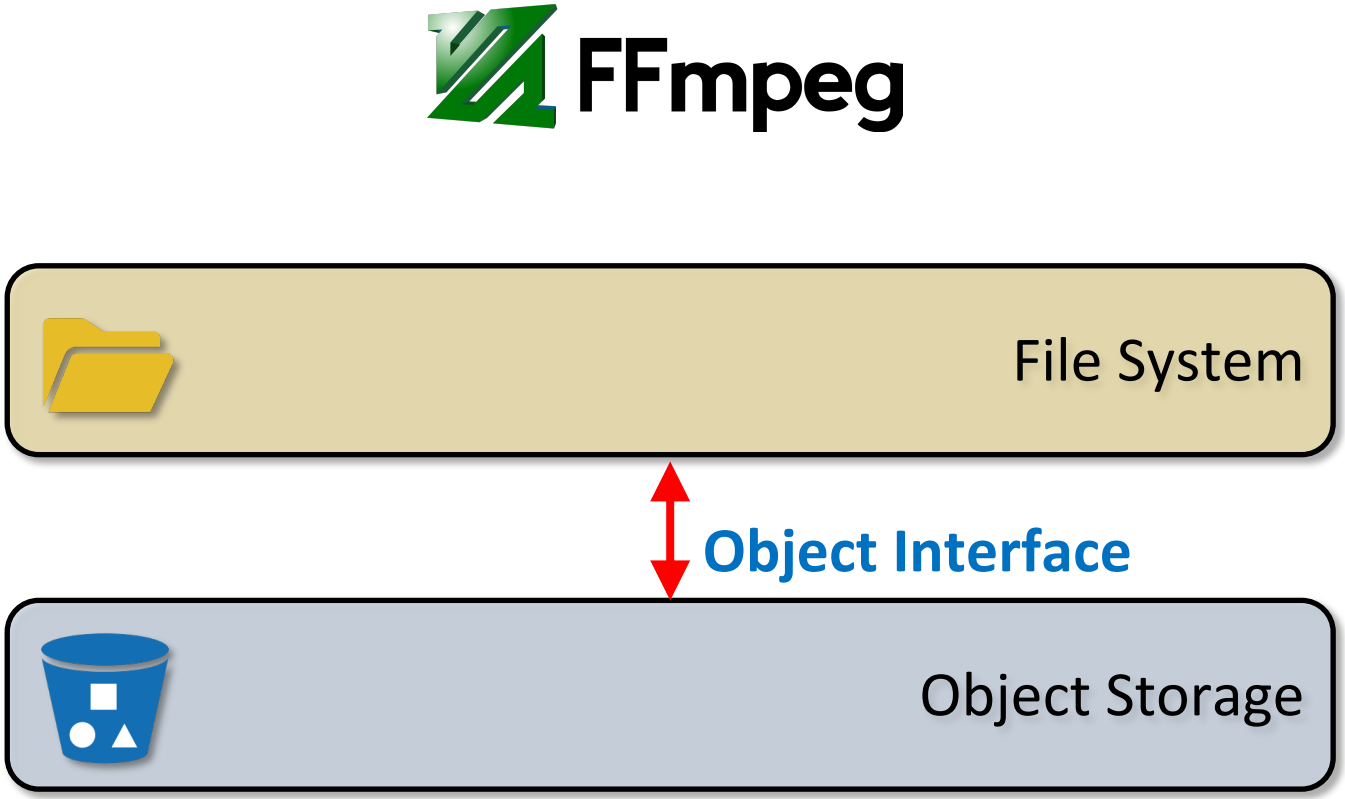
- Design considerations
- Existing systems
- Agni

https://www.greenbiz.com/sites/default/files/styles/gbz_article_primary_breakpoints_kalapicture_screenmd_1x/public/images/articles/featured/datacenter_0.jpg?itok=iJm7ezgB&timestamp=1483504030

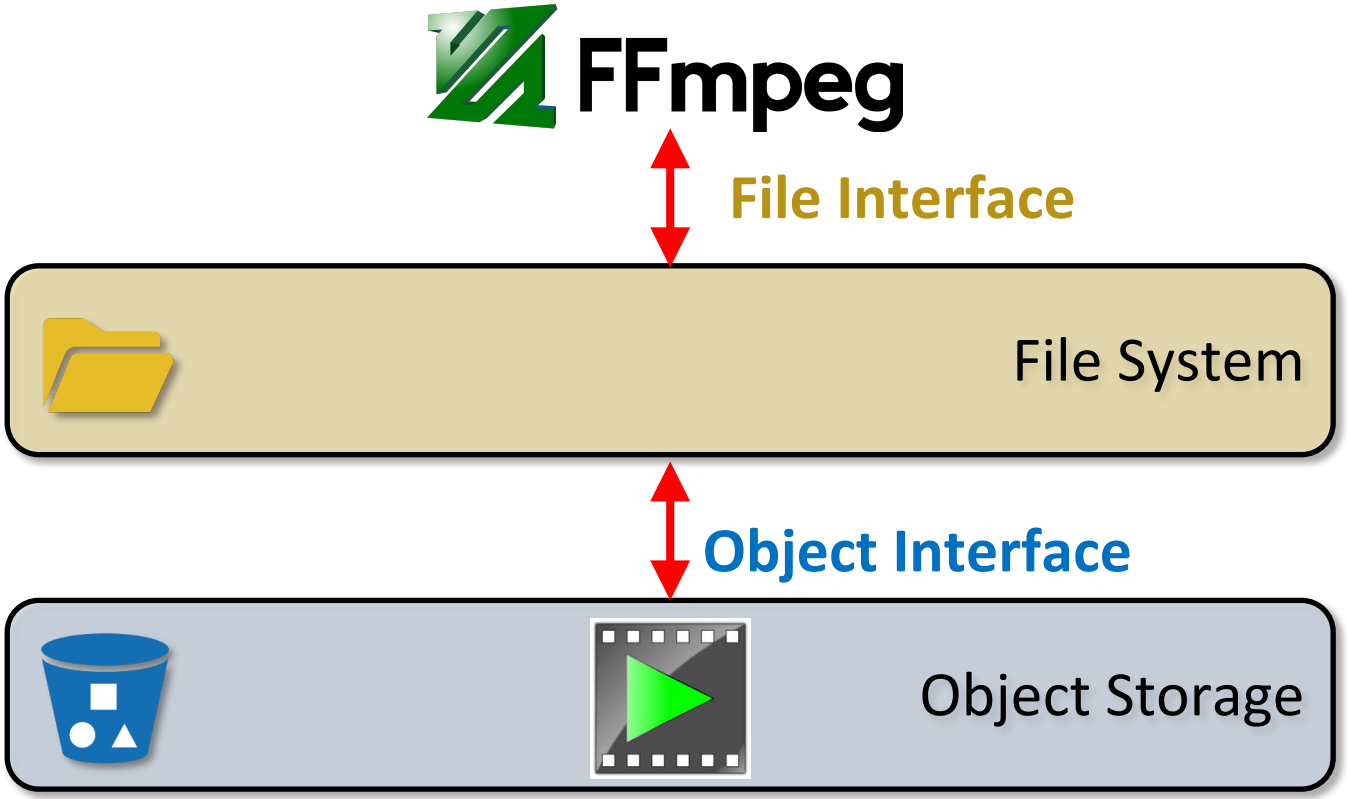
File Systems Paired With Object Storage



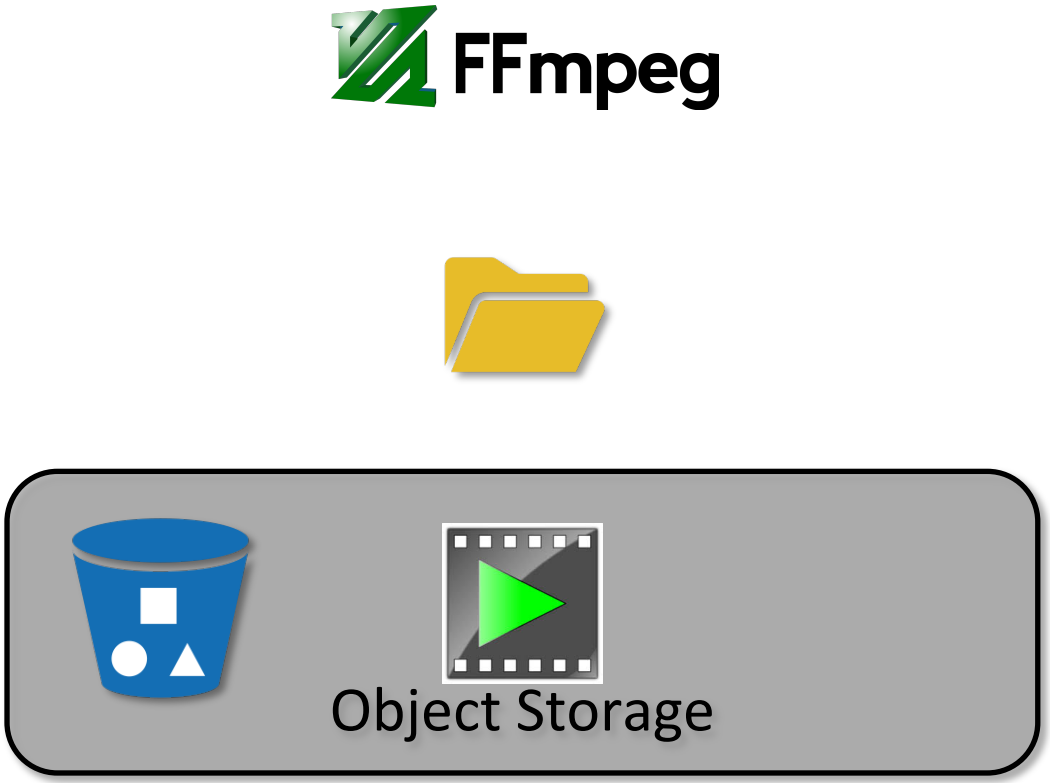
File Systems Paired With Object Storage



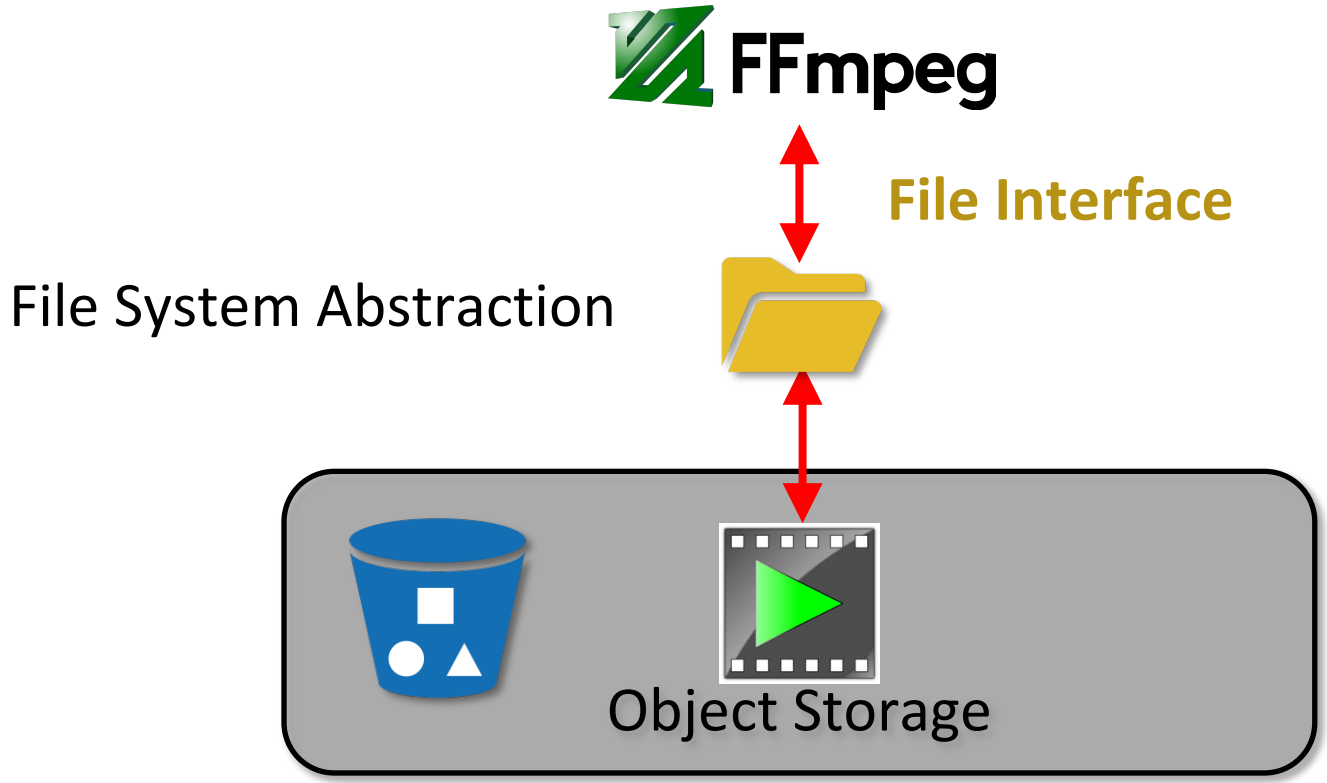
File Systems Paired With Object Storage



Object Storage File System



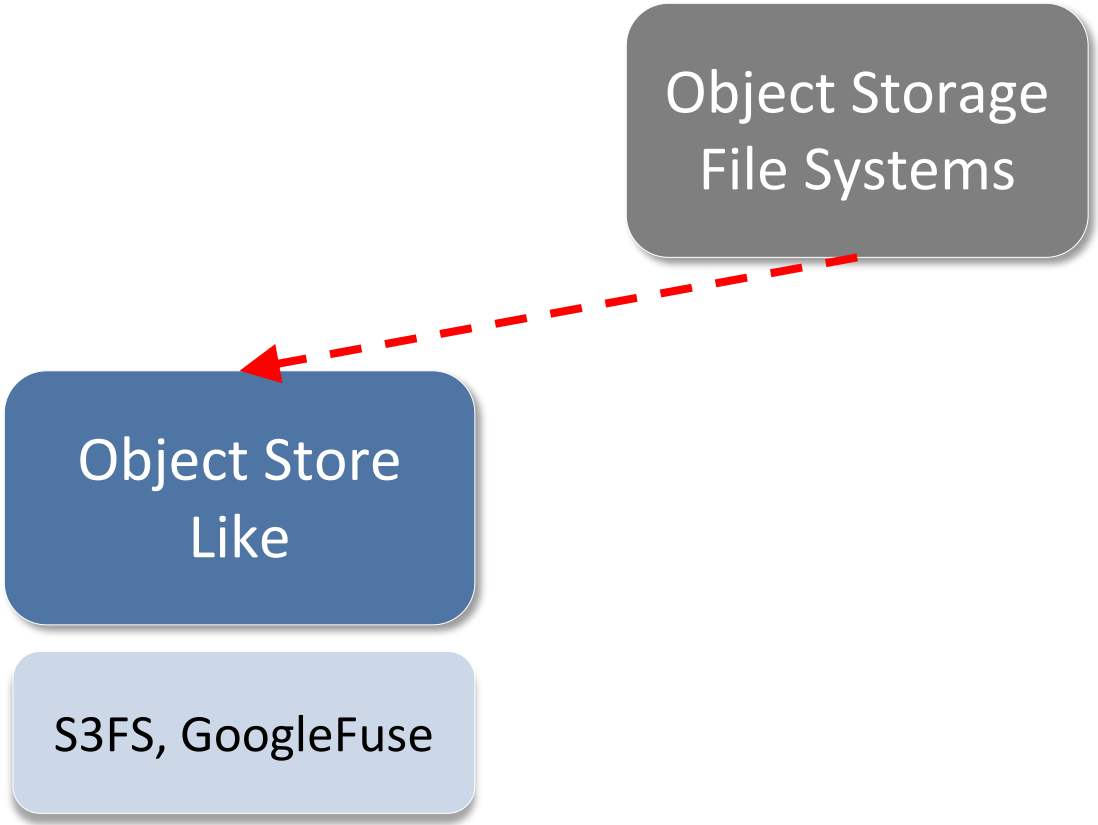
Object Storage File System



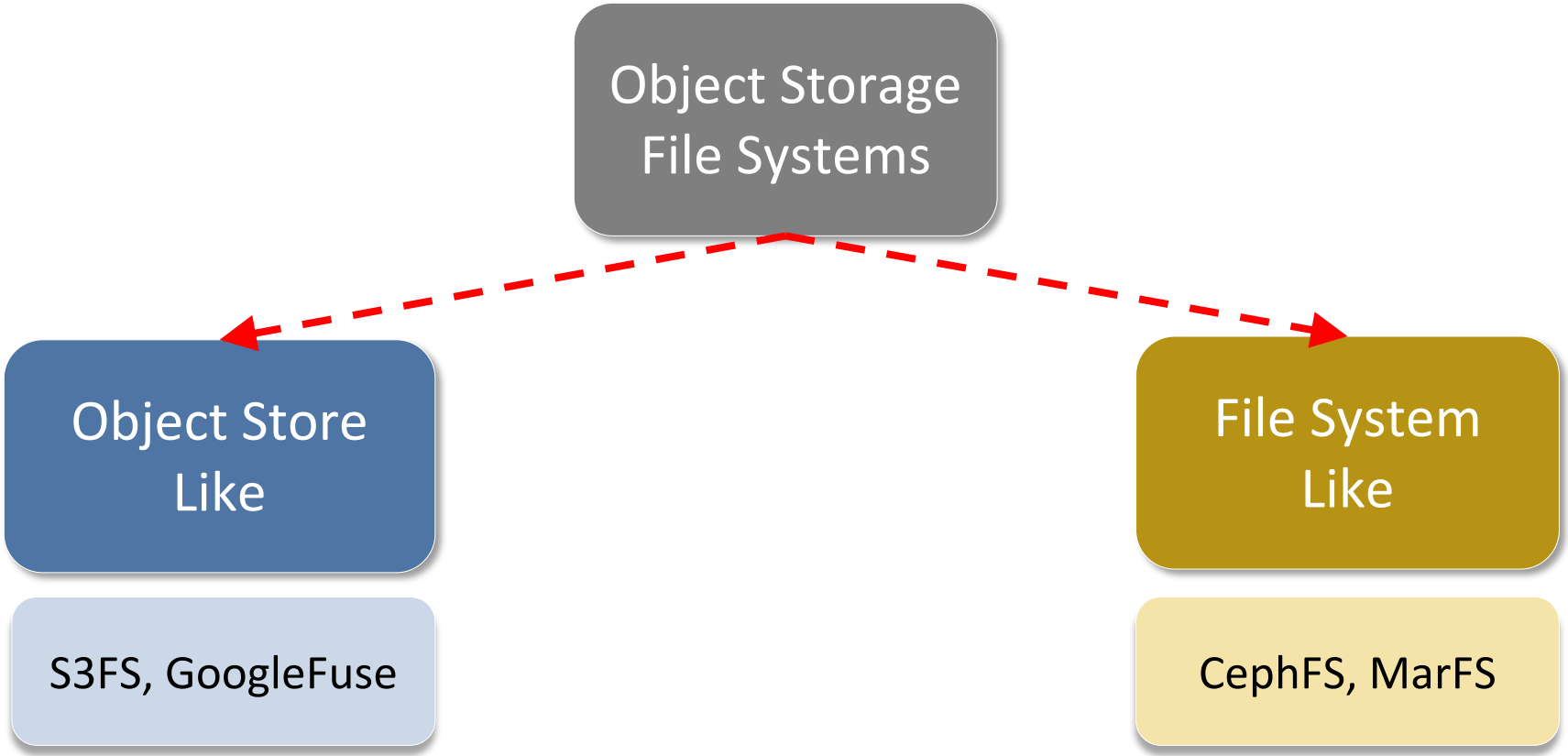
Existing Systems

Object Storage
File Systems

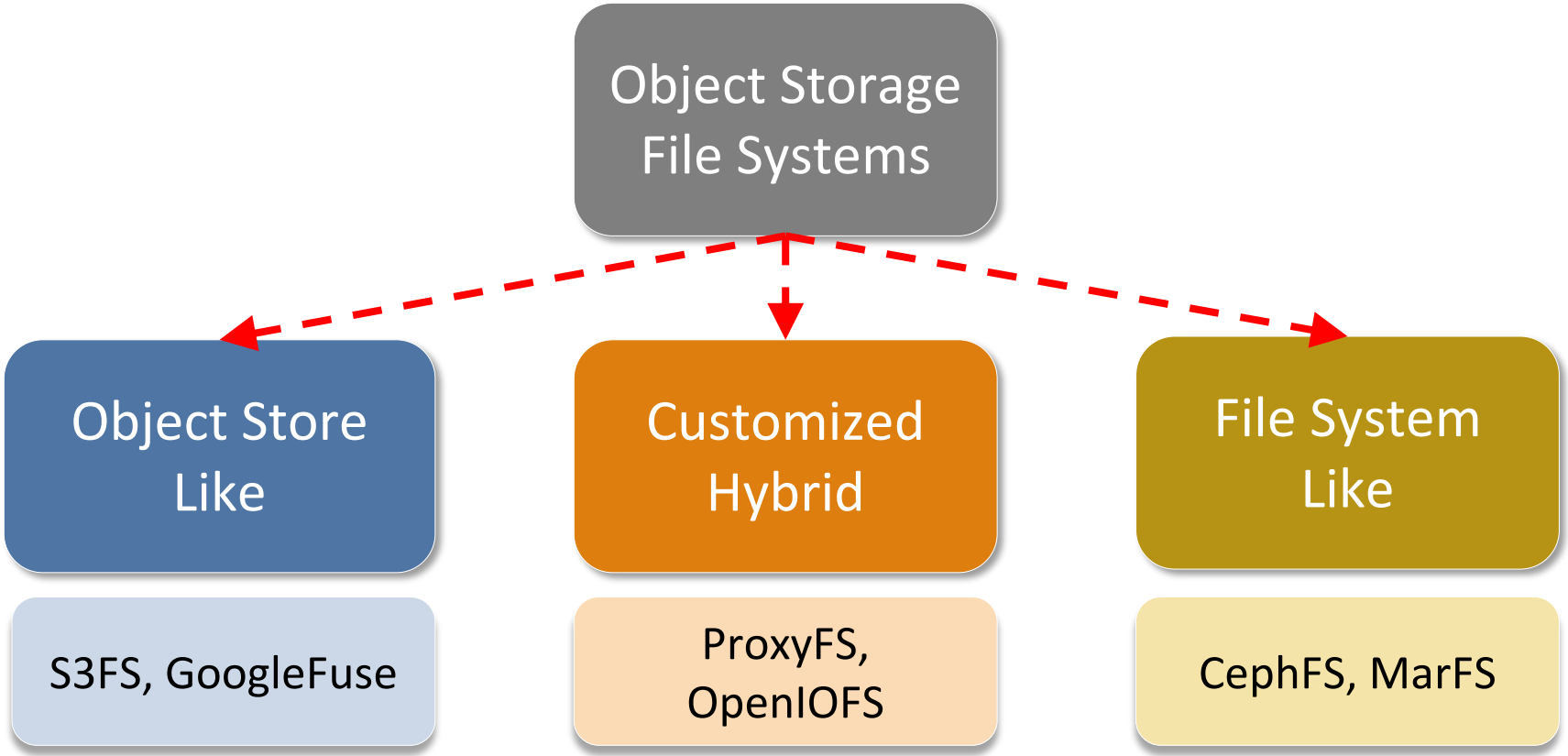
Existing Systems



Existing Systems



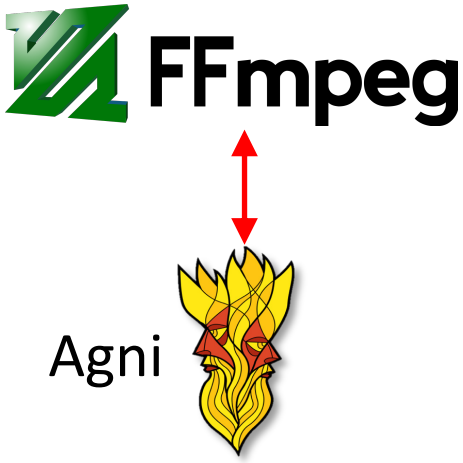
Existing Systems



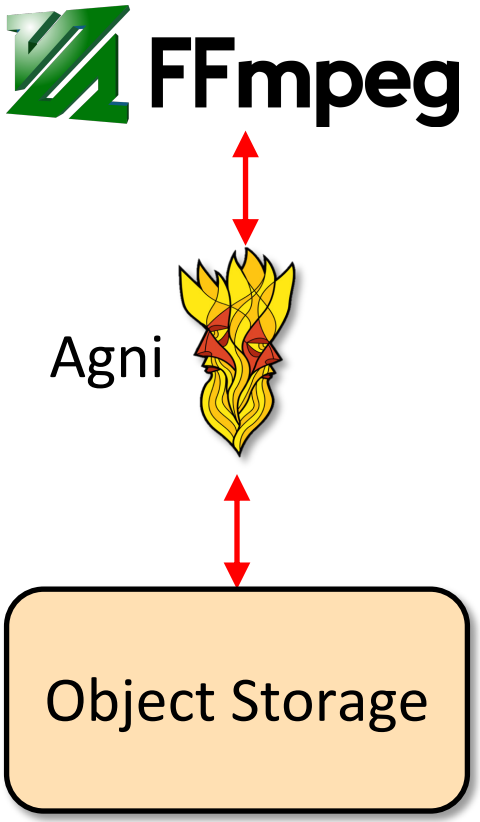
Outline

- Design considerations
- Existing systems
- Agni
- Future work

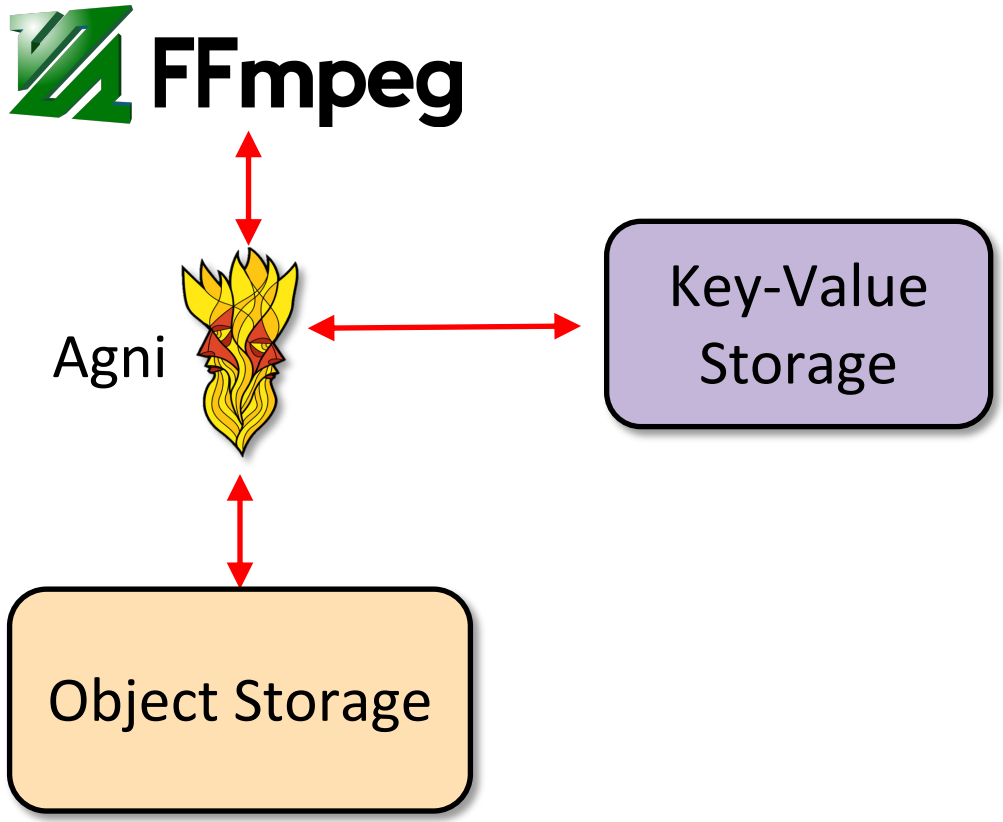
Architecture



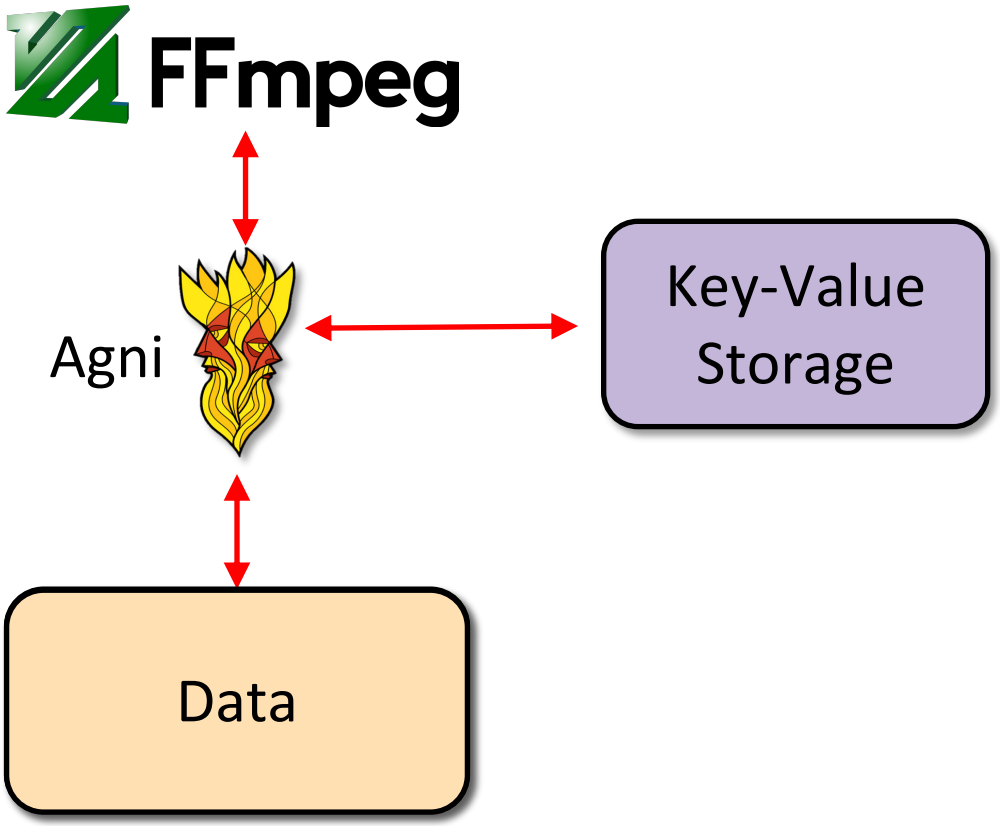
Architecture



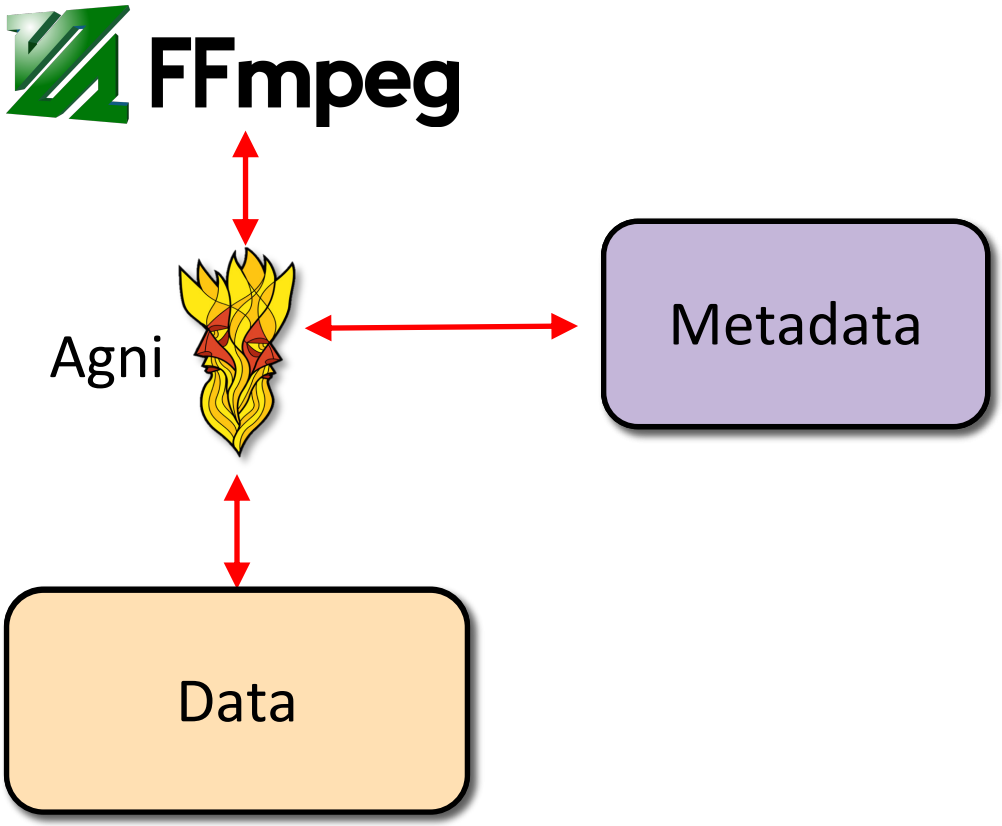
Architecture



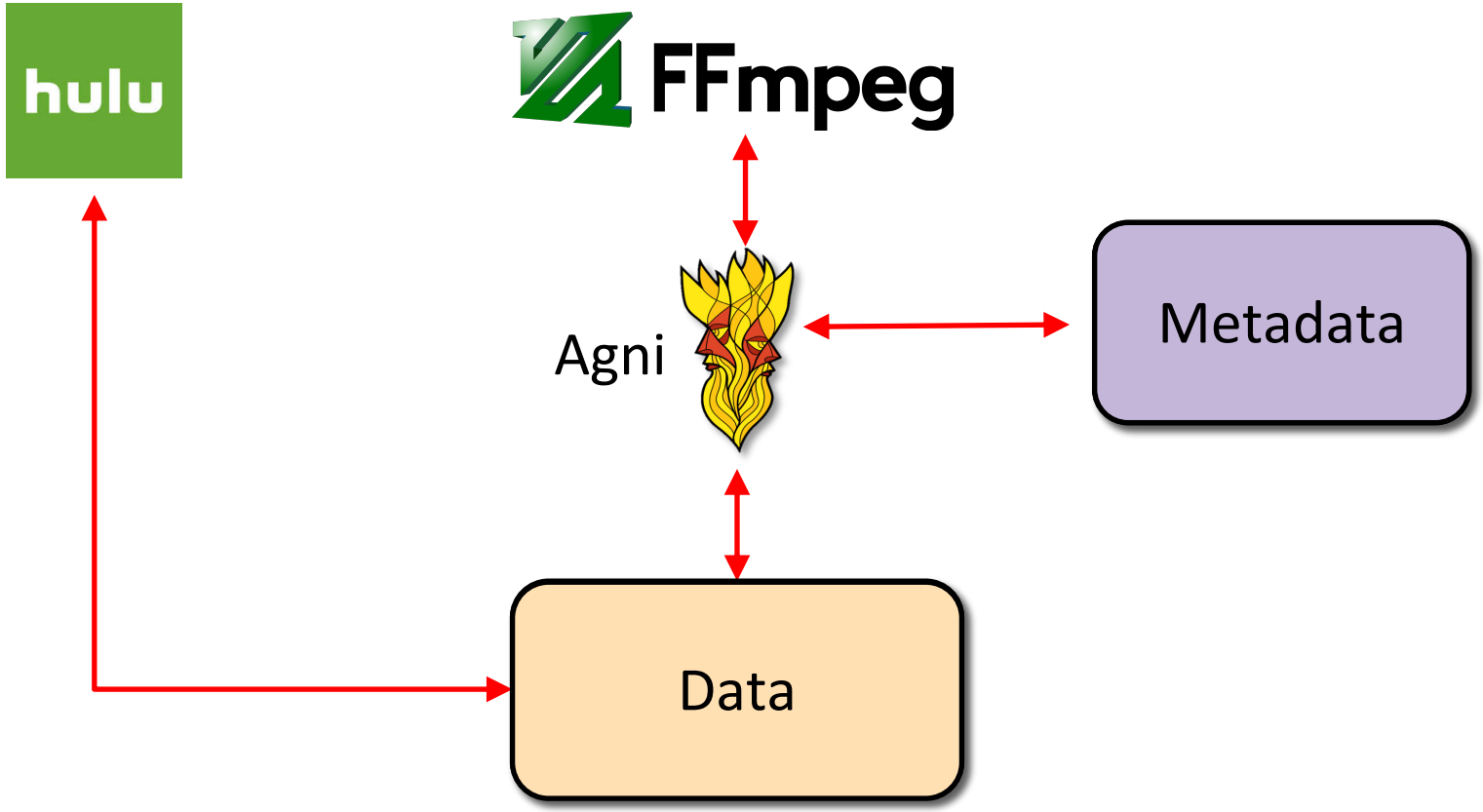
Architecture



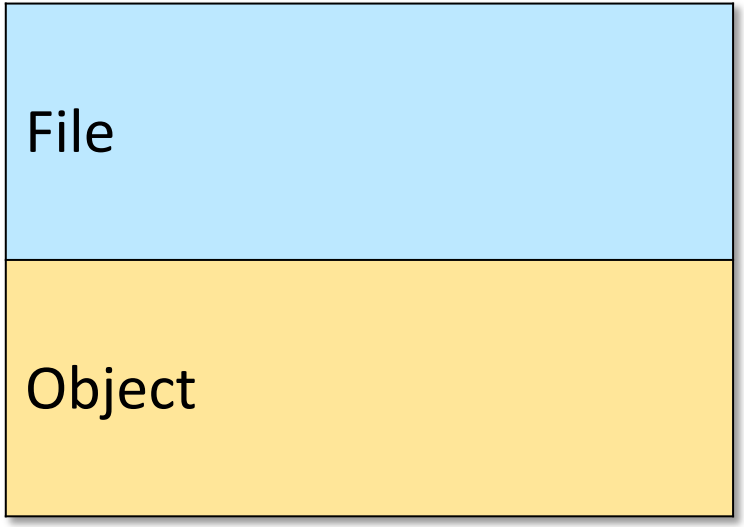
Architecture



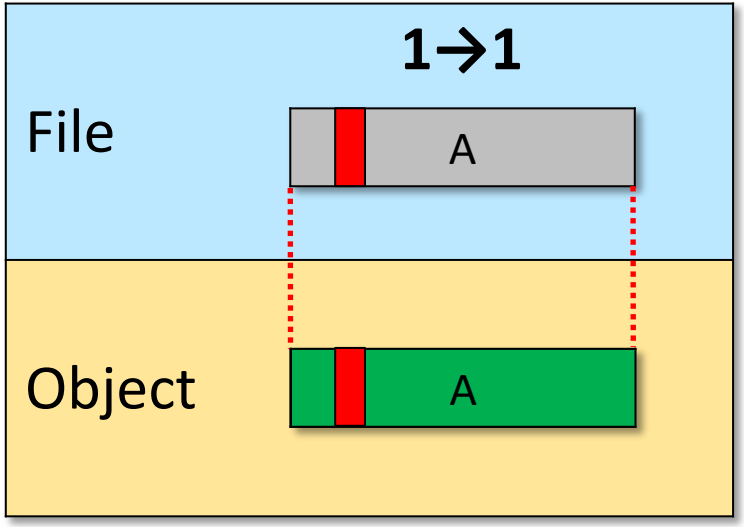
Architecture



Our Design Choices



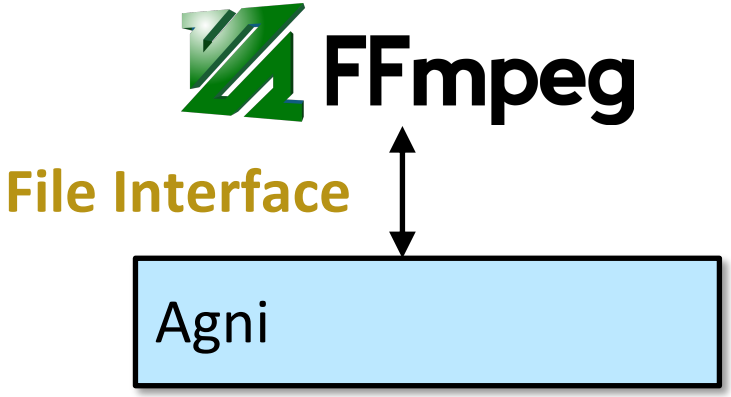
Our Design Choices



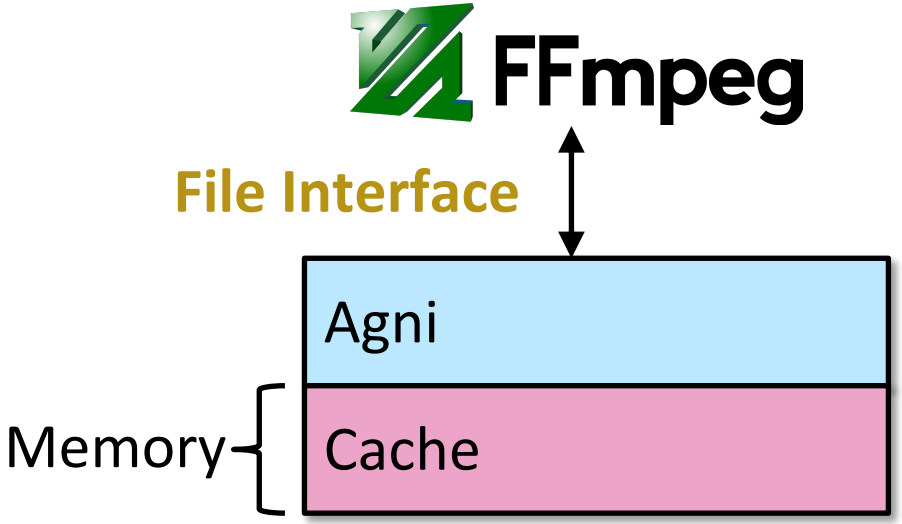
What about inefficient writes to immutable objects ?

Multi-Tier Data Structure

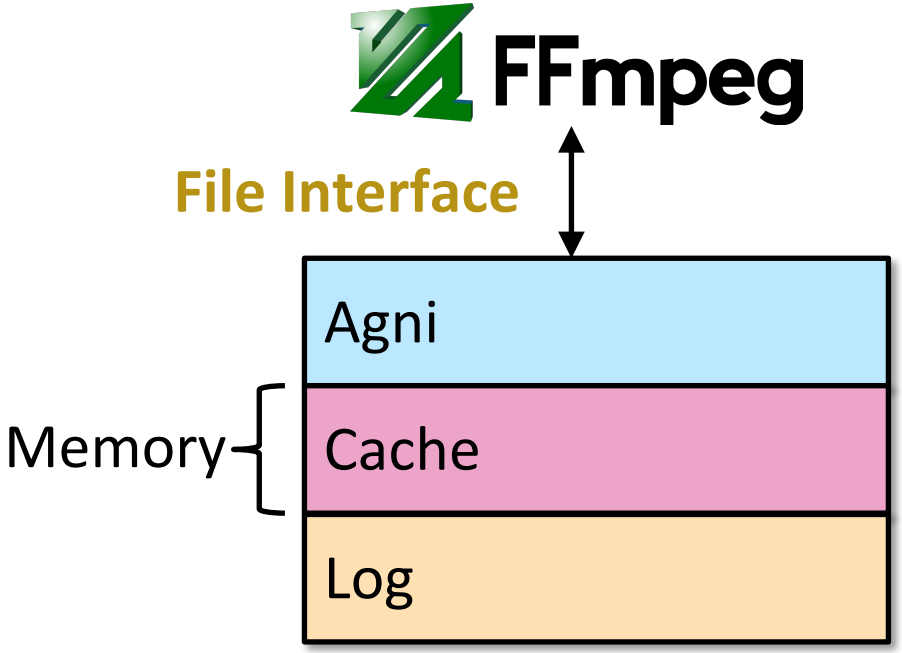
Multi-Tier Data Structure



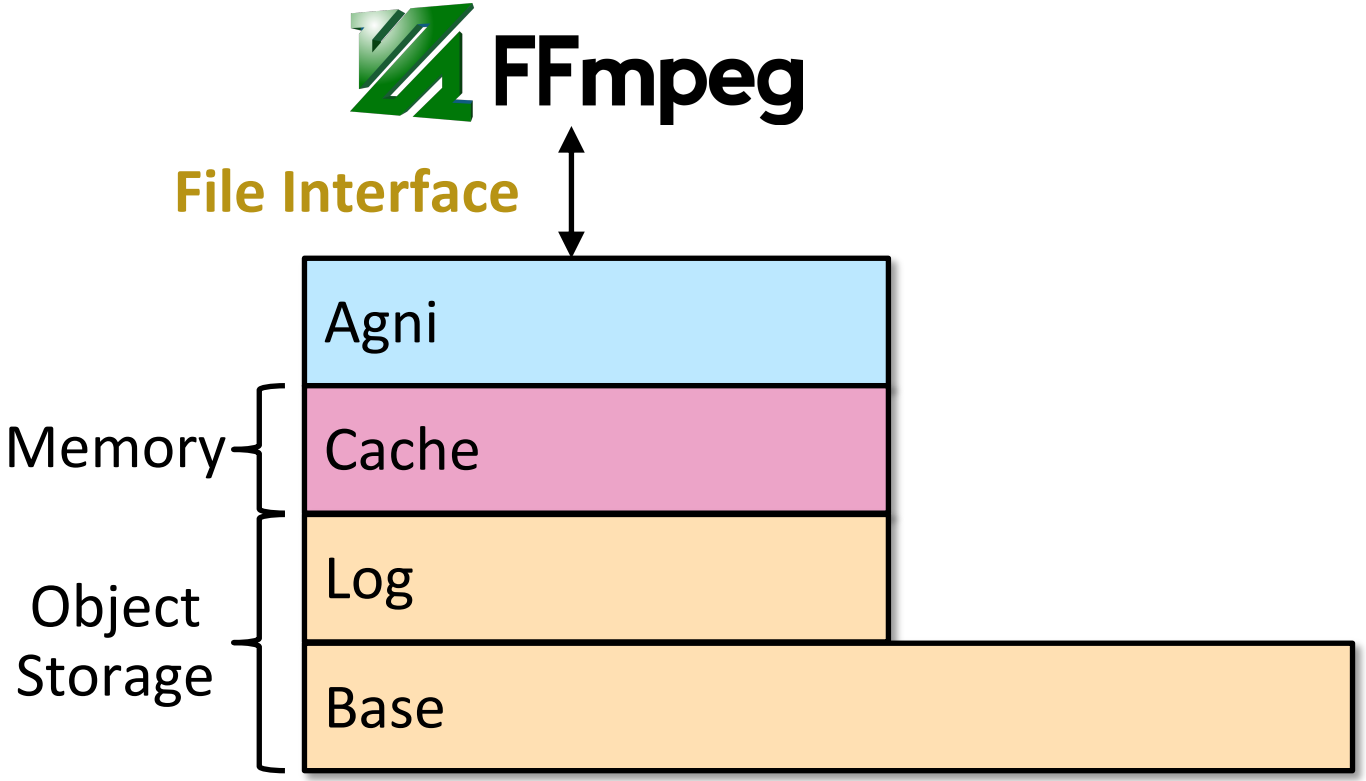
Multi-Tier Data Structure



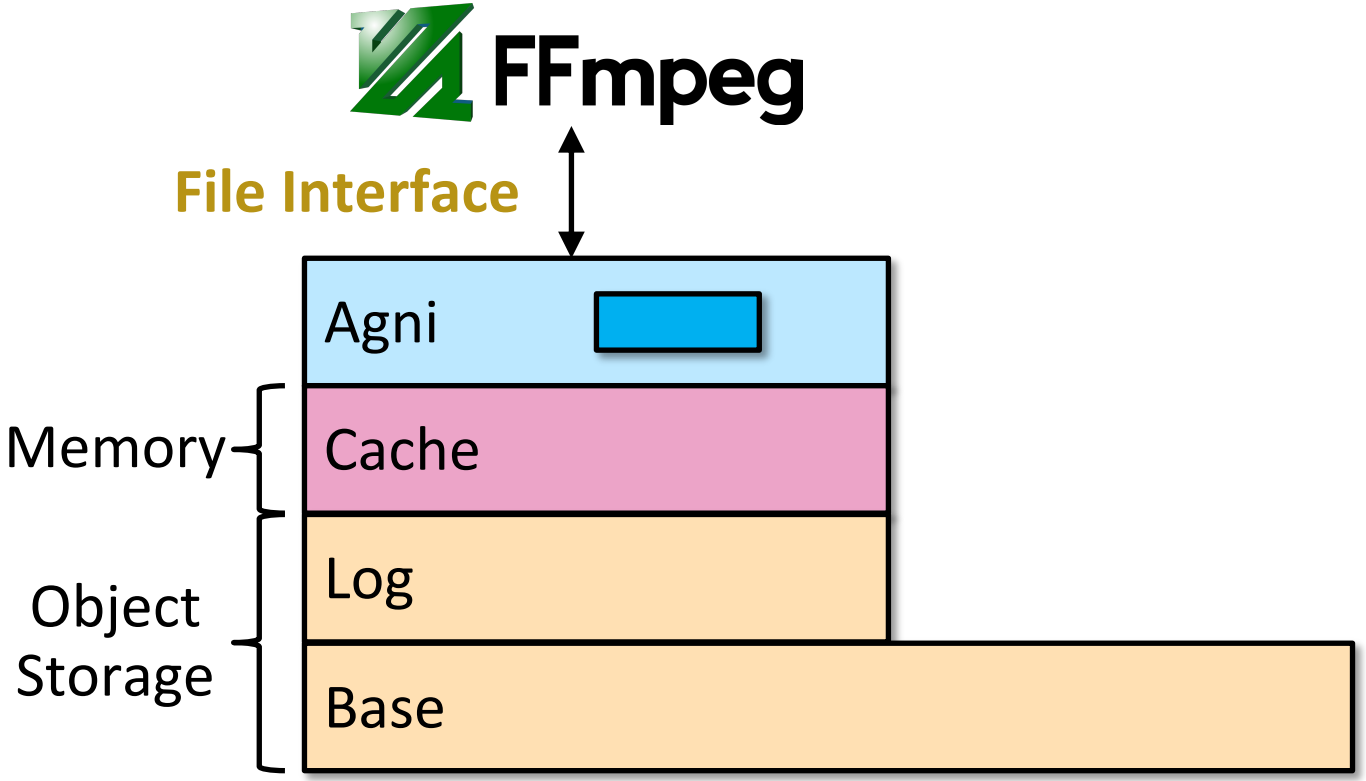
Multi-Tier Data Structure



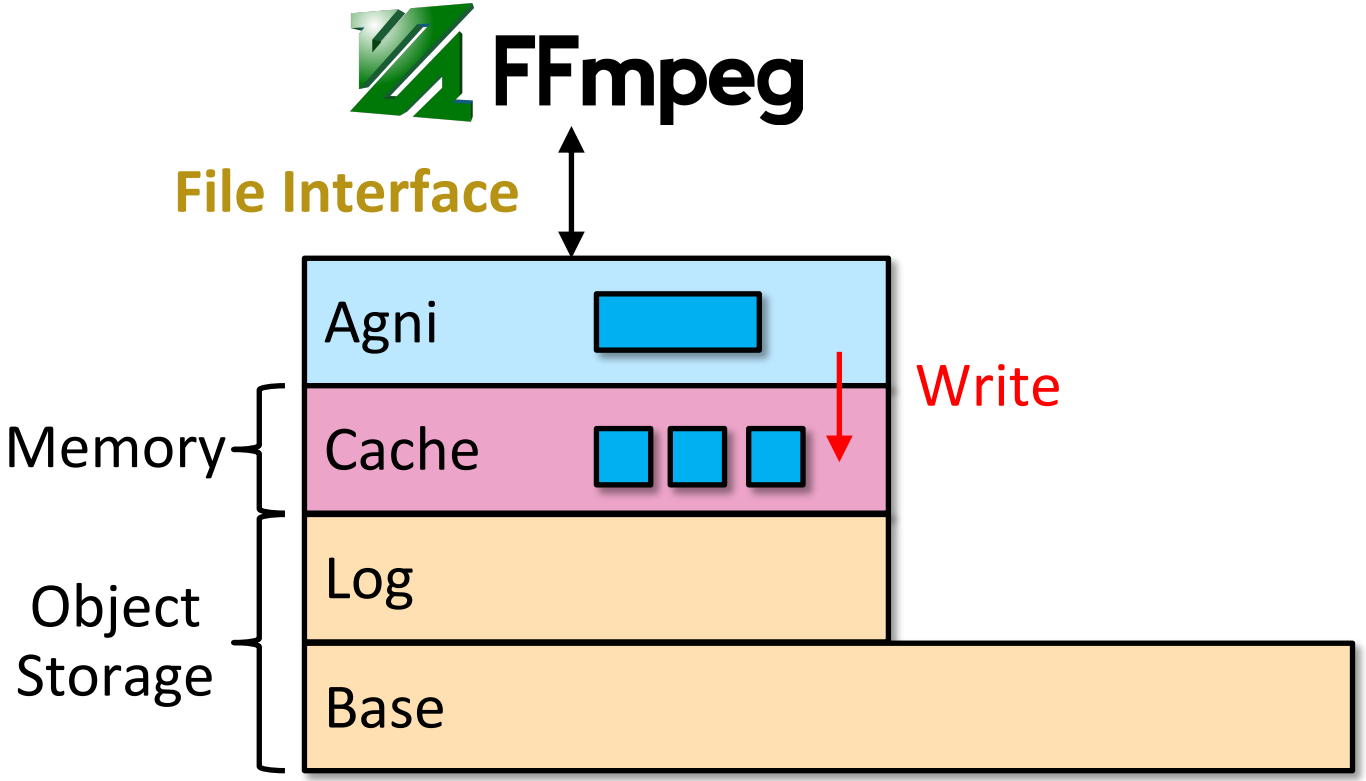
Multi-Tier Data Structure



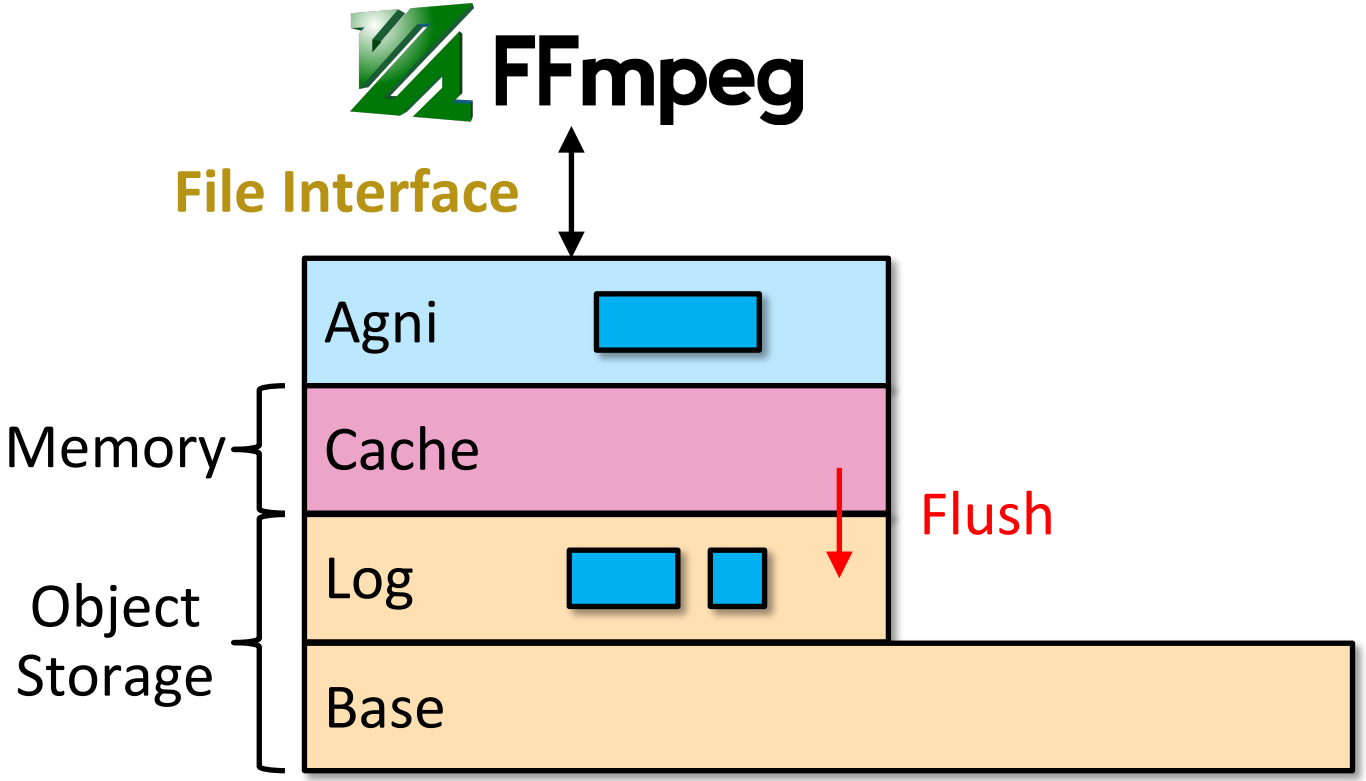
Multi-Tier Data Structure



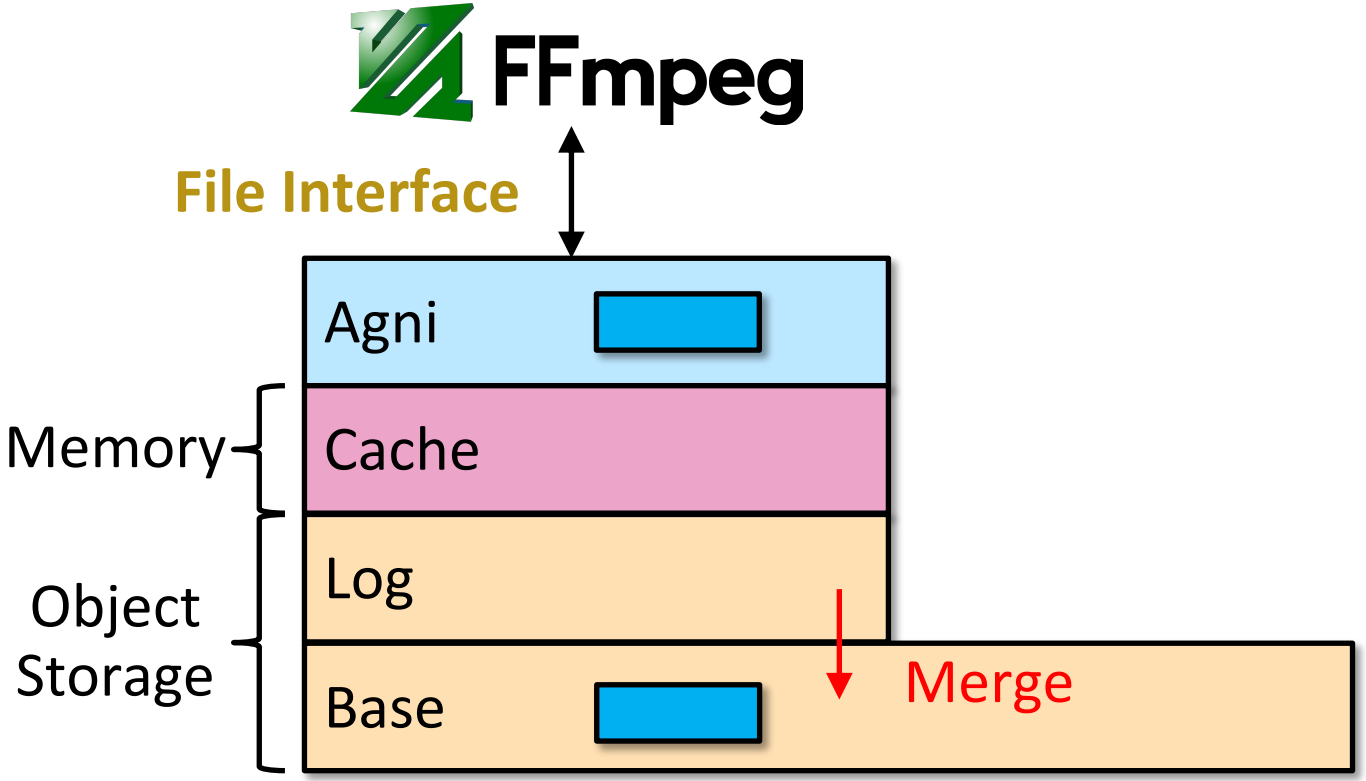
Multi-Tier Data Structure



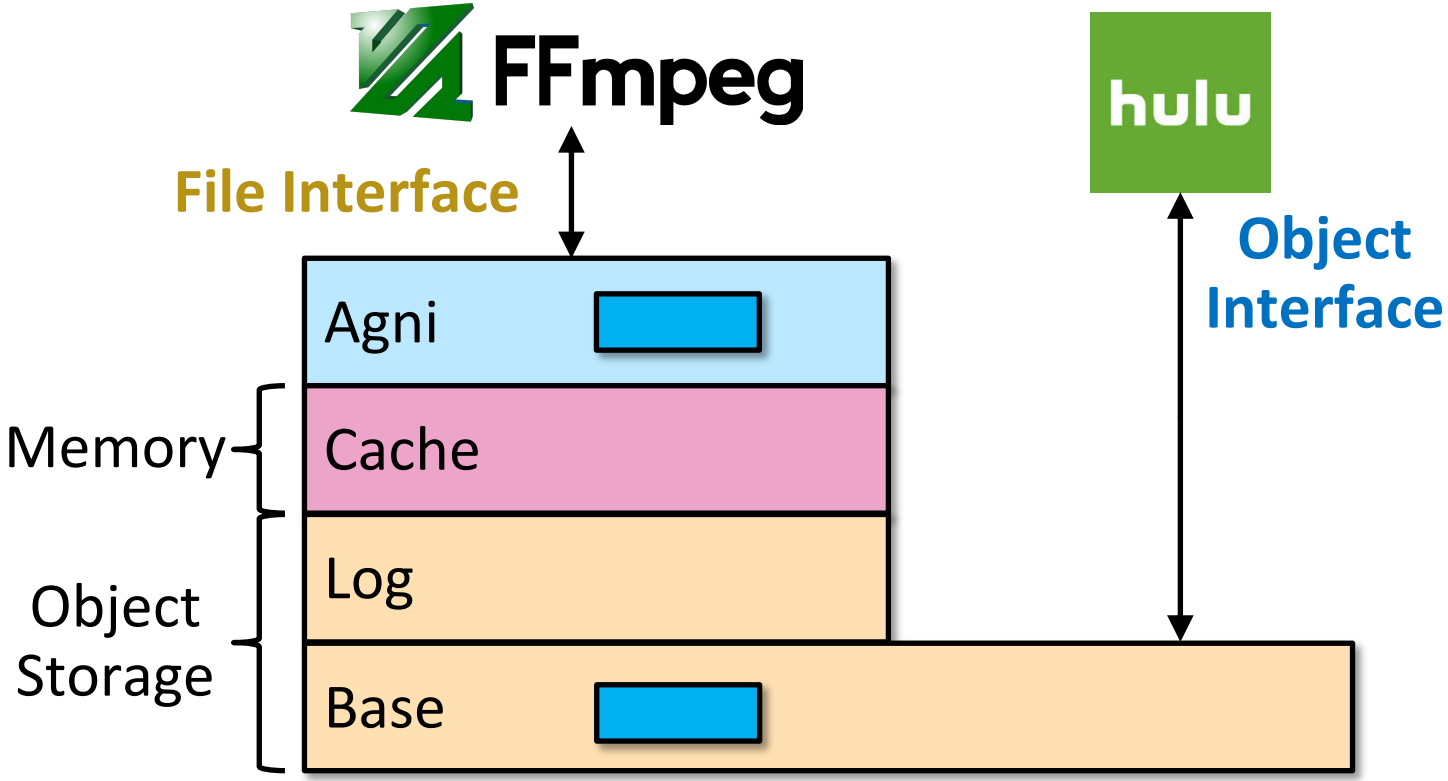
Multi-Tier Data Structure



Multi-Tier Data Structure

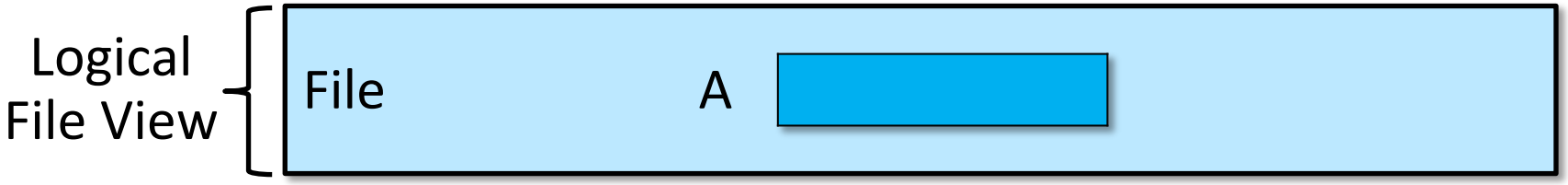


Multi-Tier Data Structure

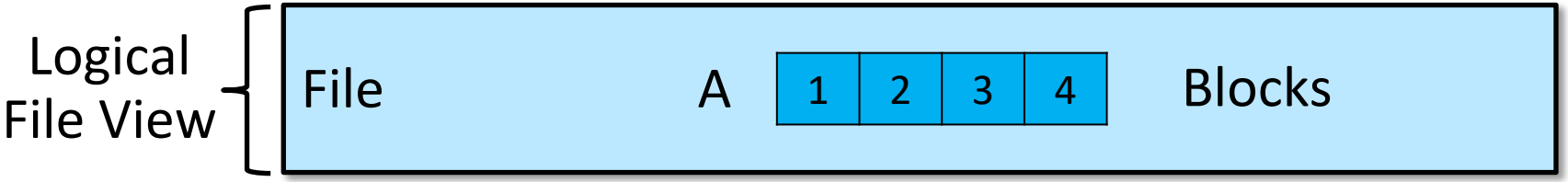


Data Layout

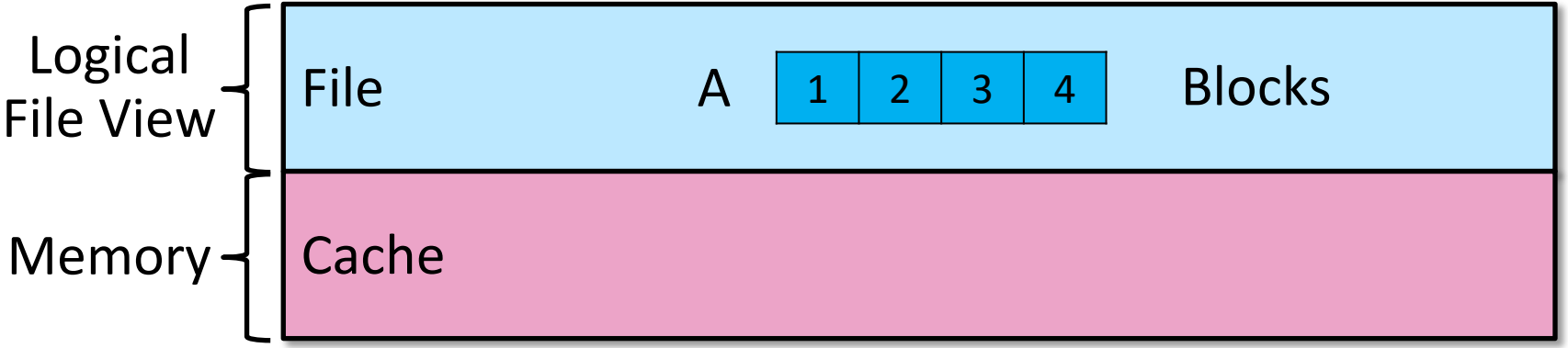
Data Layout



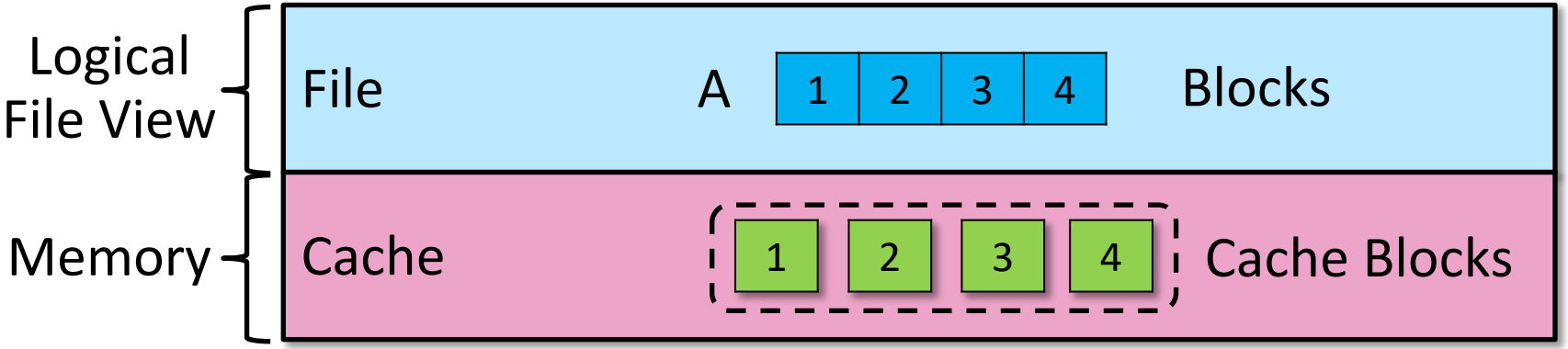
Data Layout



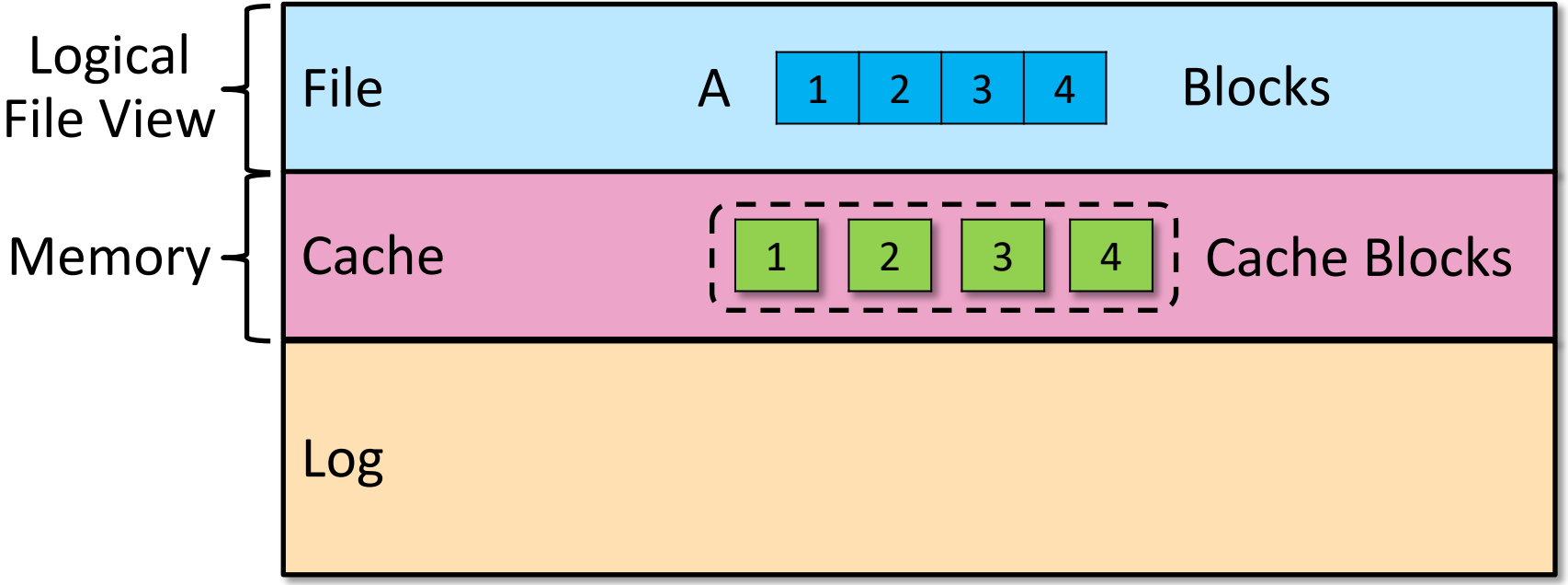
Data Layout



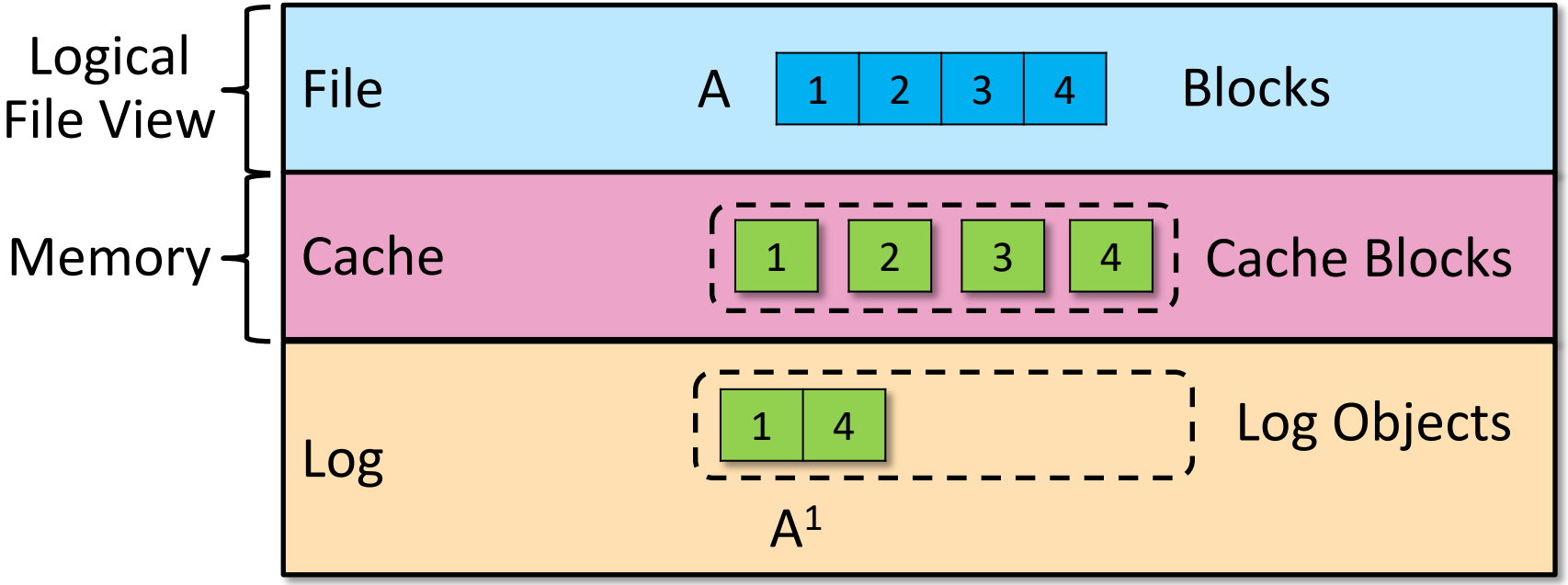
Data Layout


Logical
File View
Memory
File
A
1
2
3
4
Blocks
Cache
1
2
3
4
Cache Blocks

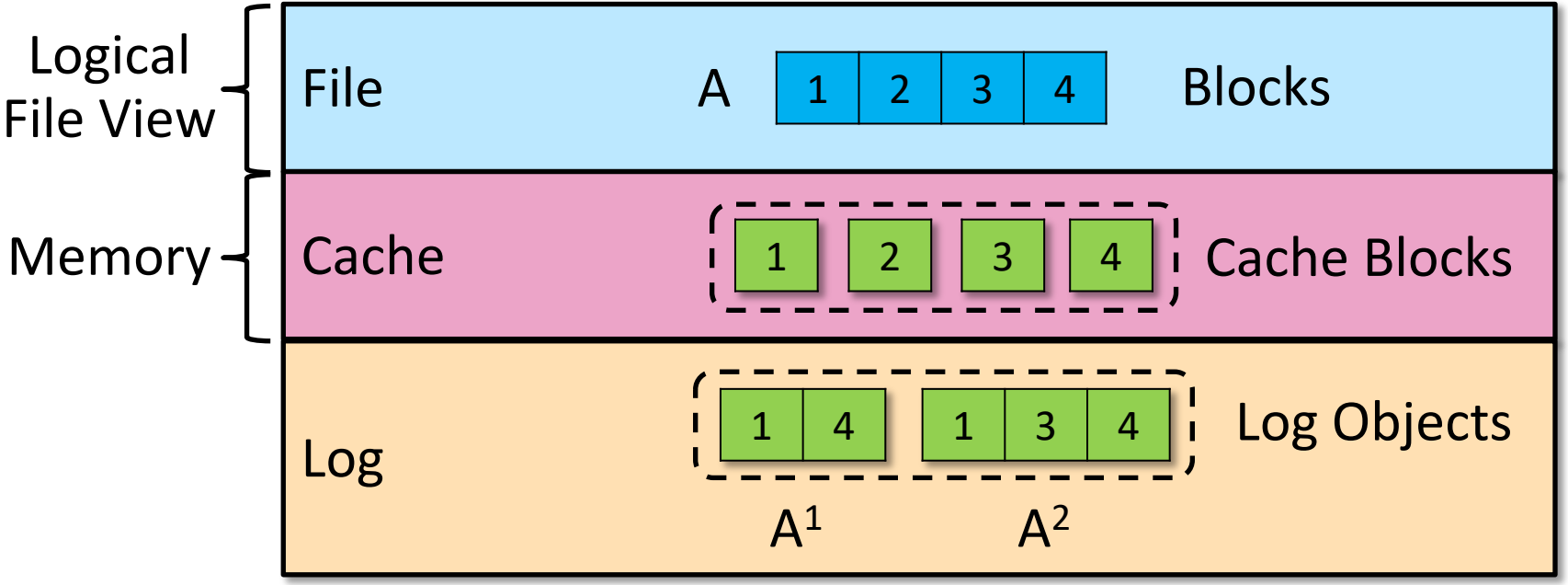
Data Layout


Logical File View
Memory
File
A
1
2
3
4
Blocks
Cache
1
2
3
4
Cache Blocks
Log

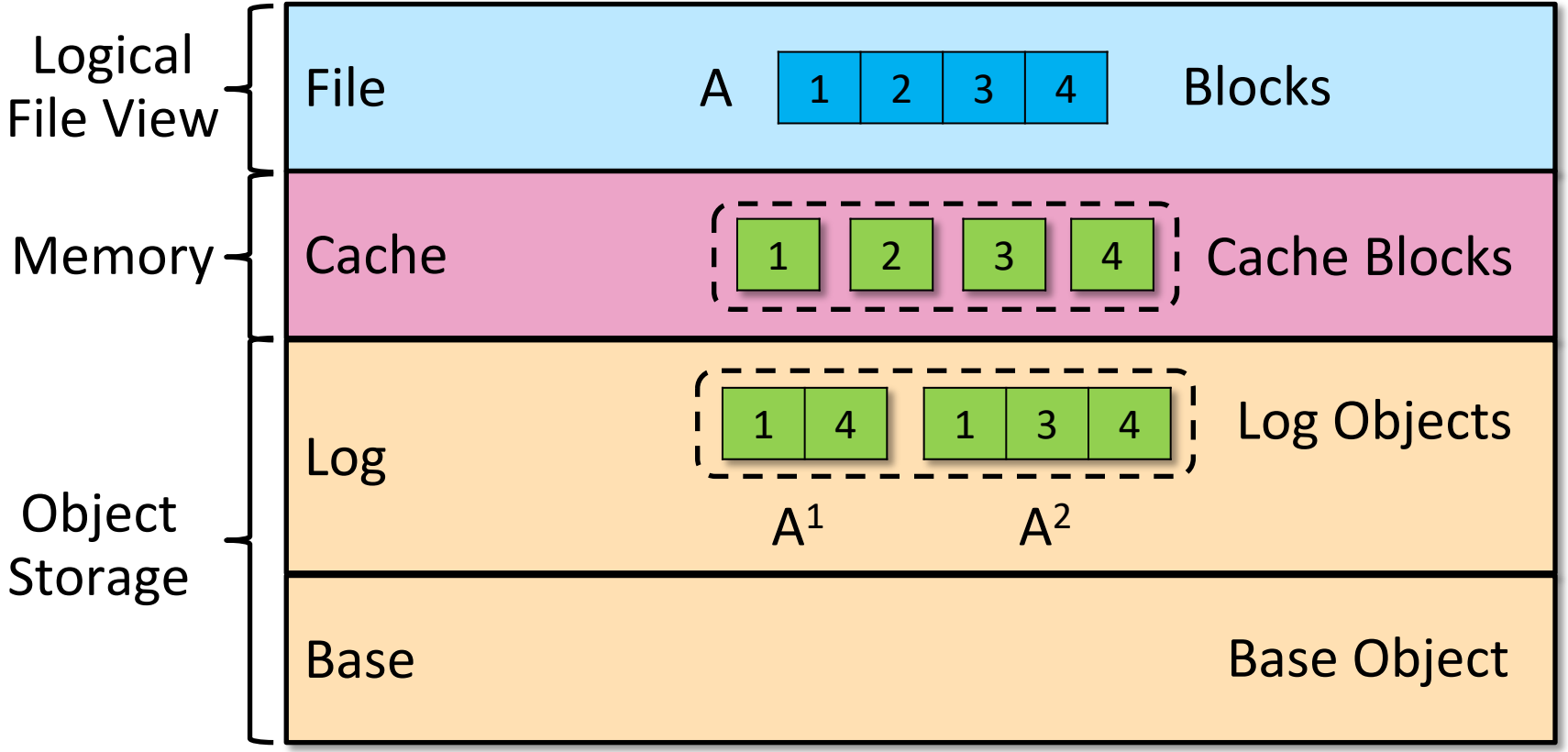
Data Layout


Logical
File View
Memory
File
A
1
2
3
4
Blocks
Cache
1
2
3
4
Cache Blocks
Log
1
4
Log Objects
A1

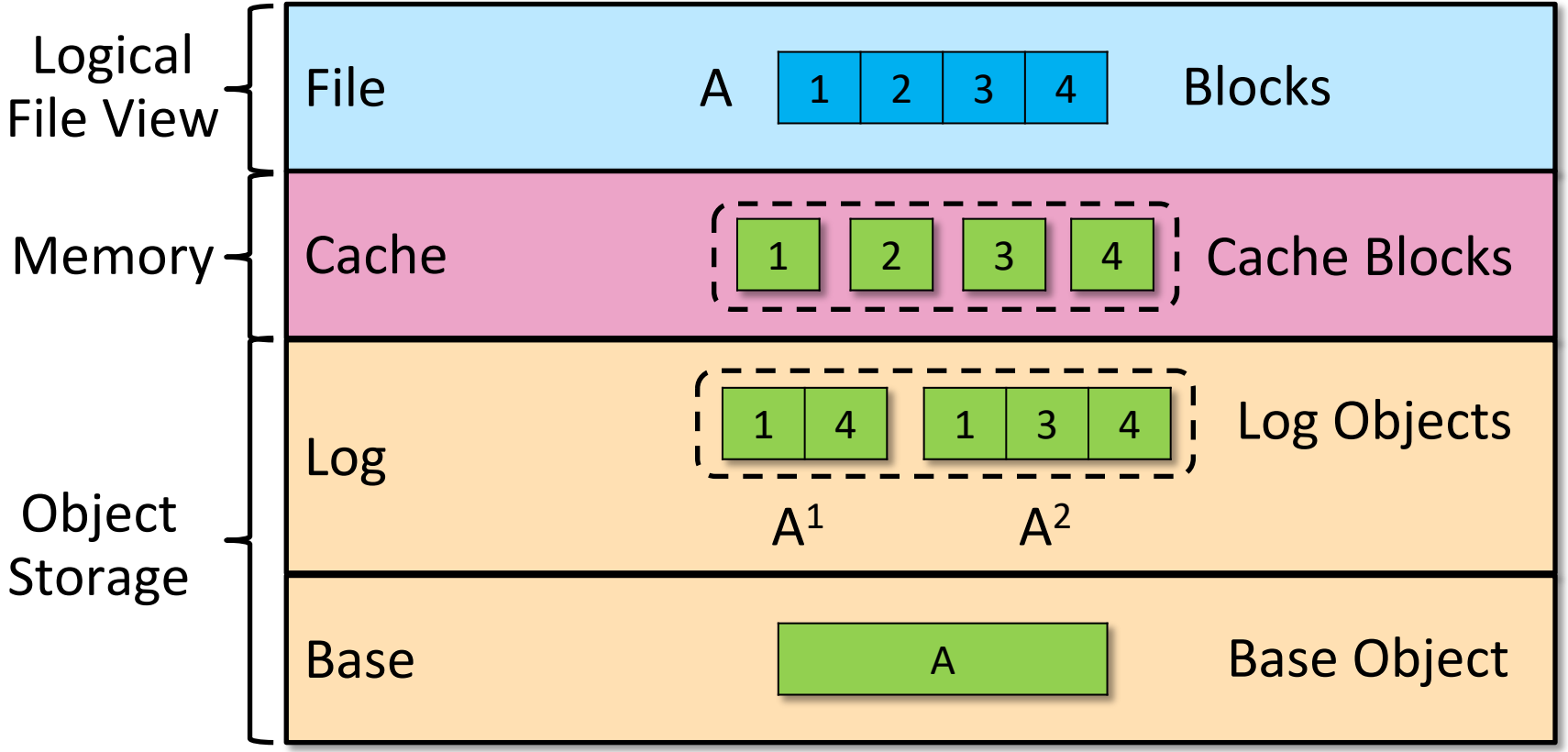
Data Layout


Logical File View
Memory
File
A
1
2
3
4
Blocks
Cache
1
2
3
4
Cache Blocks
Log
1
4
1
3
4
Log Objects
A1
A2

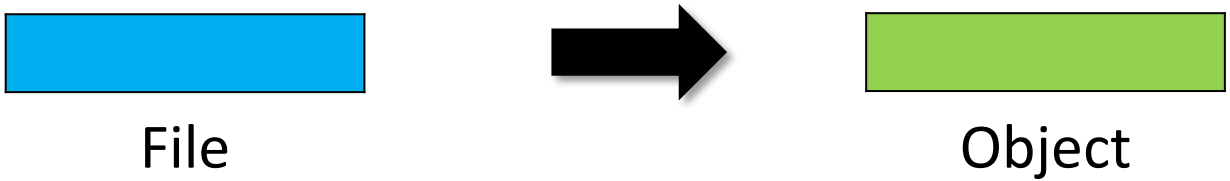
Data Layout


Logical File View
Memory
Object Storage
File
A
1
2
3
4
Blocks
Cache
1
2
3
4
Cache Blocks
Log
1
4
1
3
4
Log Objects
A1
A2
Base
Base Object

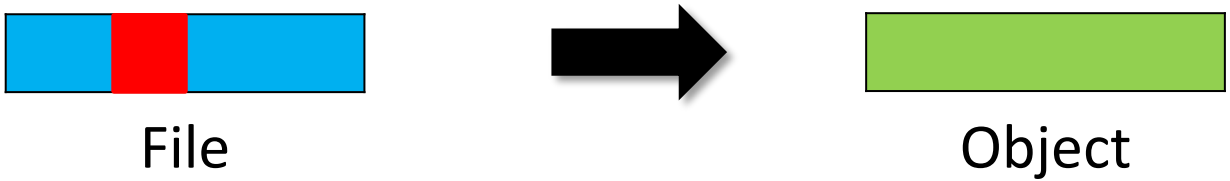
Data Layout


Logical File View
Memory
Object Storage
File
A
1
2
3
4
Blocks
Cache
1
2
3
4
Cache Blocks
Log
1
4
1
3
4
Log Objects
A1
A2
Base
A
Base Object

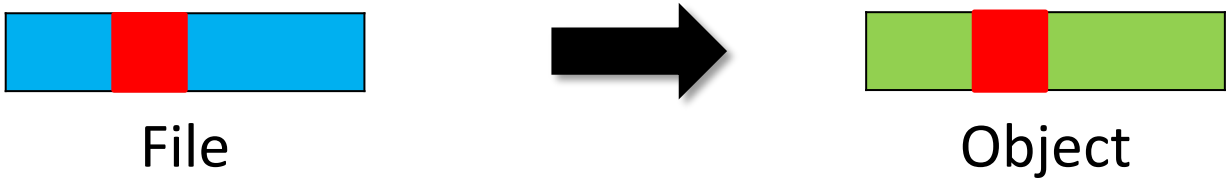
We Call This Approach Eventual 1→1 Mapping



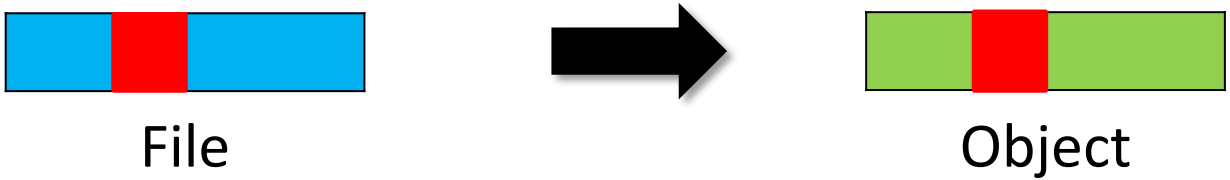
We Call This Approach Eventual 1→1 Mapping



We Call This Approach Eventual 1→1 Mapping

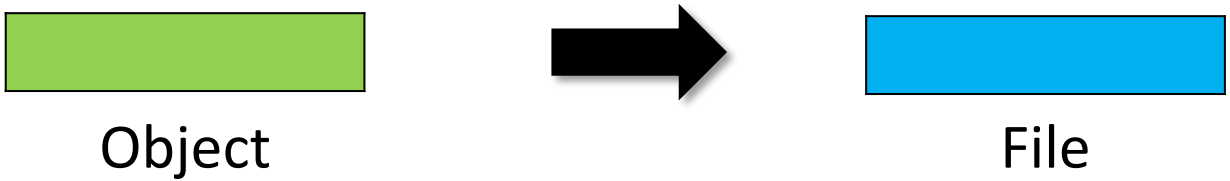


We Call This Approach Eventual 1→1 Mapping

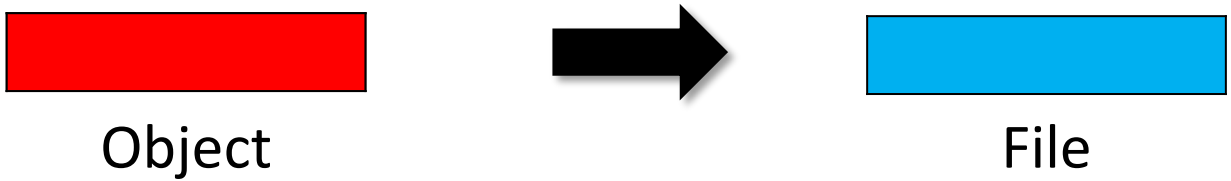


File to Object Visibility Lag

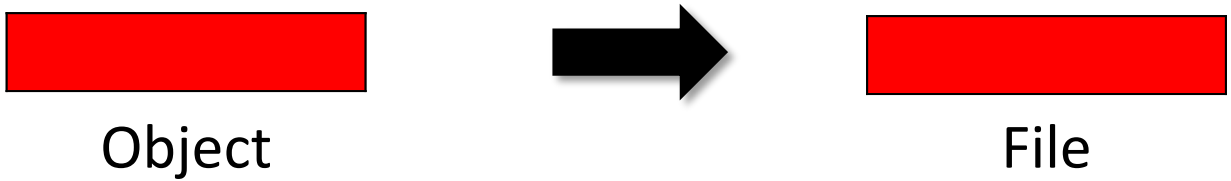
Eventual 1→1 Mapping Works Both Ways



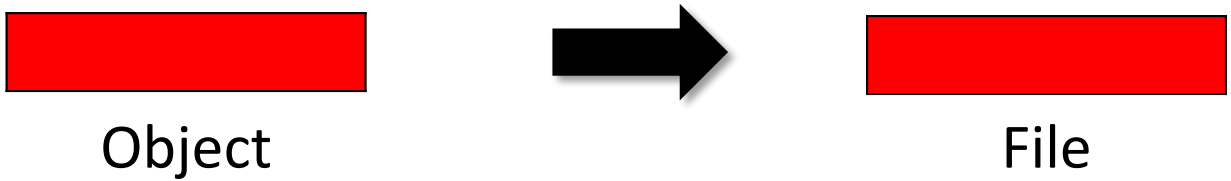
Eventual 1→1 Mapping Works Both Ways



Eventual 1→1 Mapping Works Both Ways

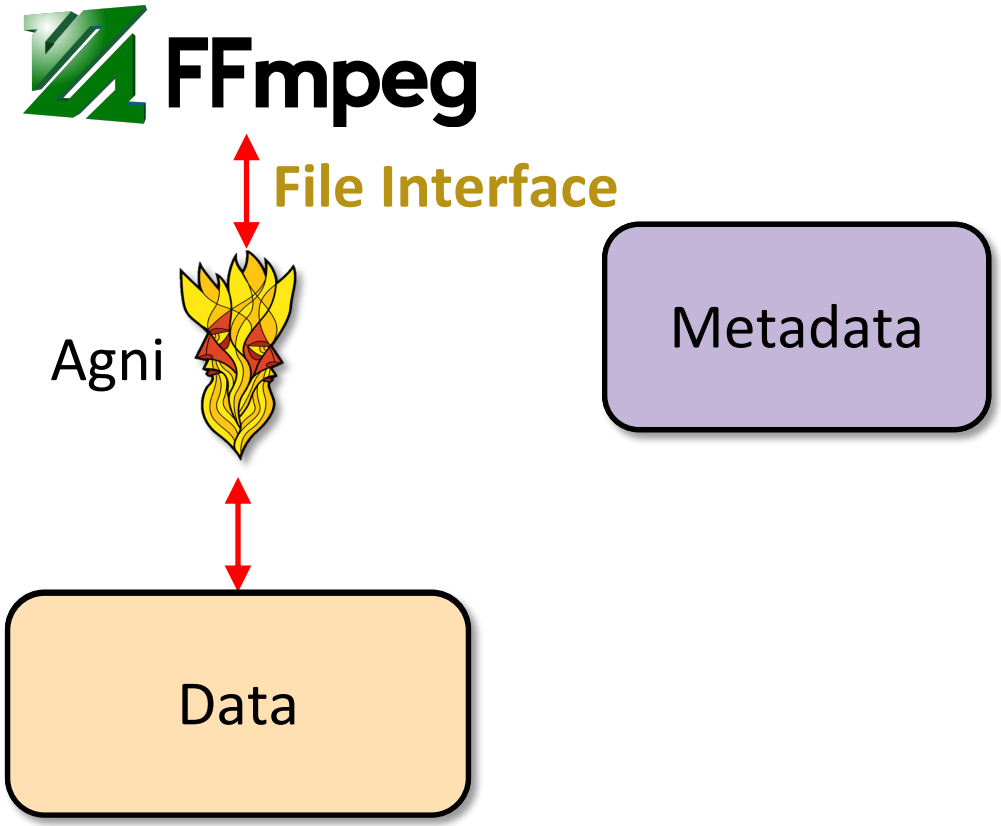


Eventual 1→1 Mapping Works Both Ways



Object to File Visibility Lag

Design: Object to File



Design: Object to File

Object Interface

FFmpeg

File Interface

Agni

Metadata

Design: Object to File

FFmpeg

File Interface

Object Interface

Agni

Visible

Metadata

Notification Processor

Namespace Management

Namespace Management

Namespace Management

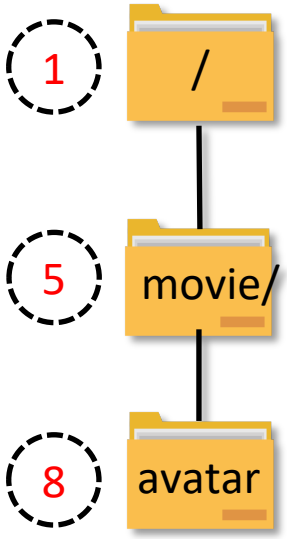
Master index

Key	Value

Namespace Management

Master index

Key	Value

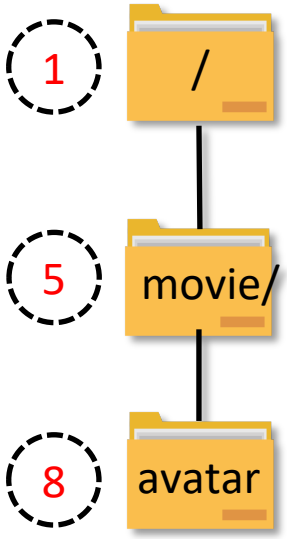


Namespace Management

Master index

Inode key

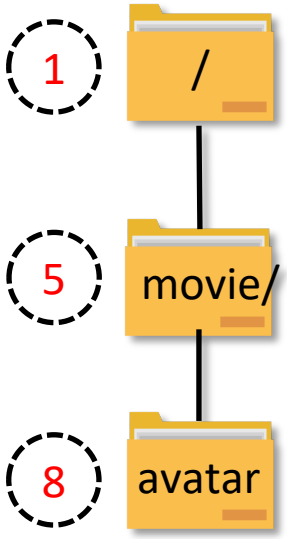
Key	Value
5	inode metadata



Namespace Management

Master index

	Key	Value
Inode key	5	inode metadata
Lookup key	1→movie	5

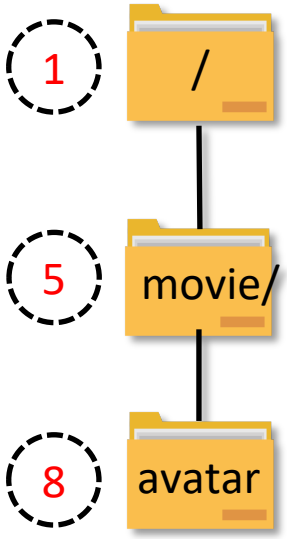


Namespace Management

Master index

	Key	Value
Inode key	5	inode metadata
Lookup key	1→movie	5
Children key	5→Children	[<8,avatar>]

Directory metadata

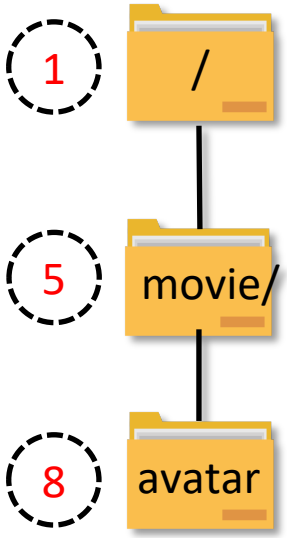


Namespace Management

Master index

	Key	Value
Inode key	5	inode metadata
Lookup key	1→movie	5
Children key	5→Children	[<8,avatar>]
Inode key	8	inode metadata

Directory metadata



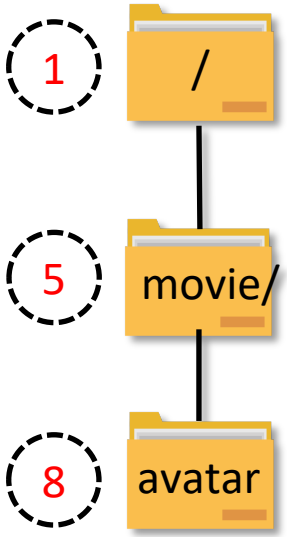
Namespace Management

Master index

	Key	Value
Inode key	5	inode metadata
Lookup key	1→movie	5
Children key	5→Children	[<8,avatar>]
Inode key	8	inode metadata
Lookup key	5→avatar	8

Directory metadata

File metadata



Fragment Map

Master index

Key	Value

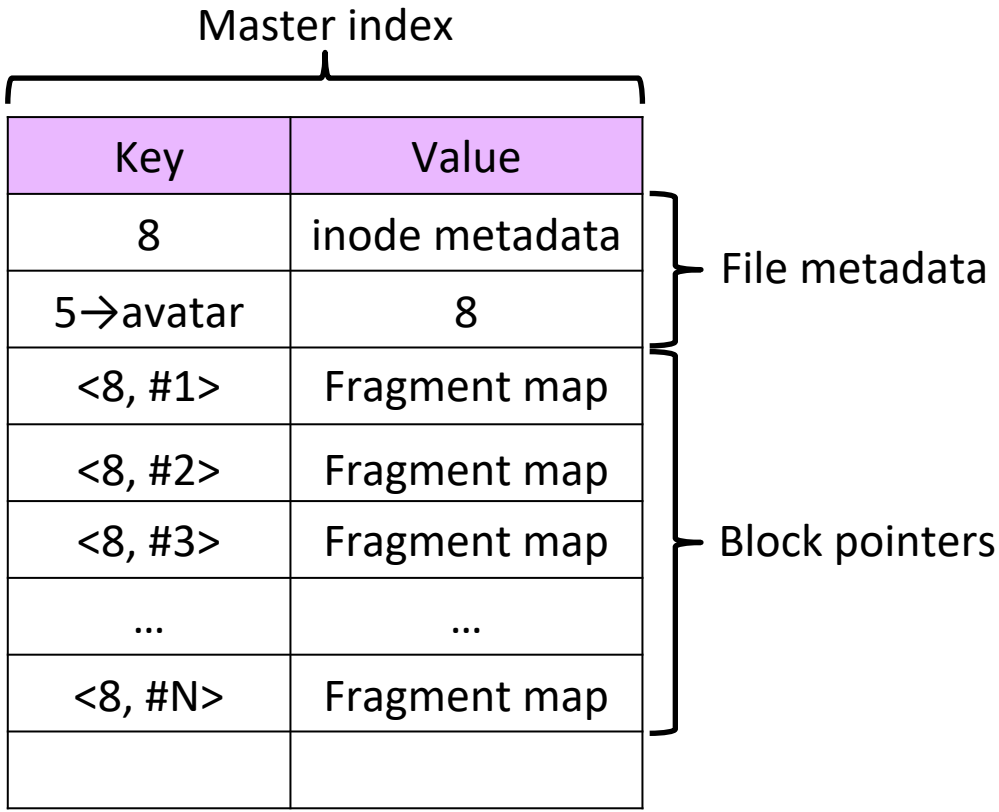
Fragment Map

Master index

Key	Value
8	inode metadata
5→avatar	8

File metadata

Fragment Map

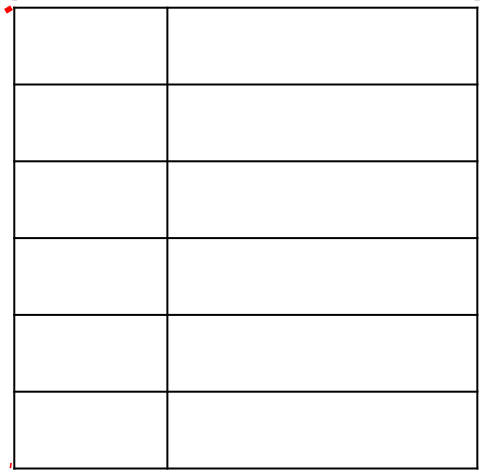


Fragment Map

Master index

Key	Value
8	inode metadata
5→avatar	8
<8, #1>	Fragment map
<8, #2>	Fragment map
<8, #3>	Fragment map
…	…
<8, #N>	Fragment map

Fragment map



Fragment Map

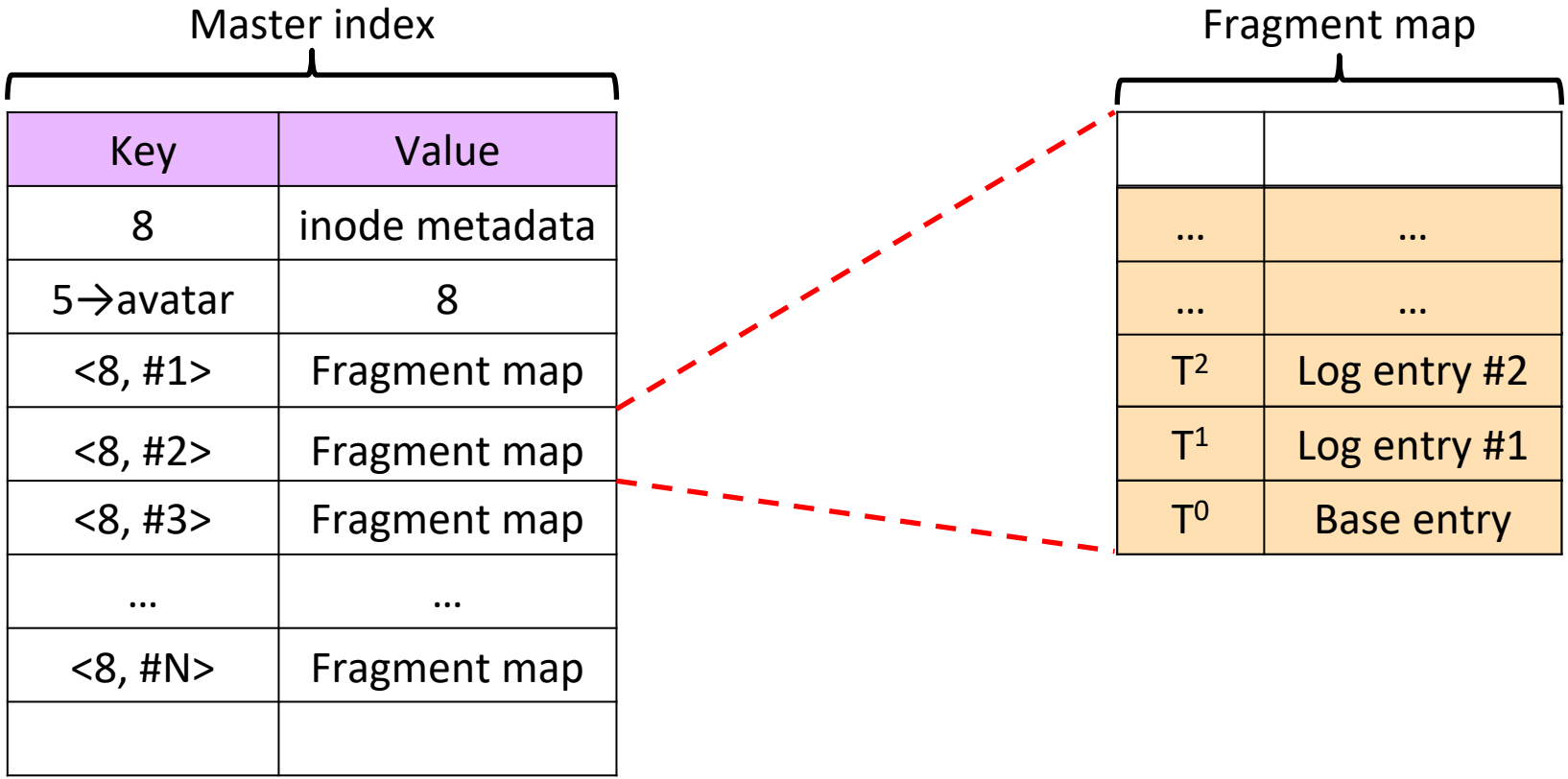
Master index

Key	Value
8	inode metadata
5→avatar	8
<8, #1>	Fragment map
<8, #2>	Fragment map
<8, #3>	Fragment map
…	…
<8, #N>	Fragment map

Fragment map

T^0	Base entry

Fragment Map



Fragment Map

Master index

Key	Value
8	inode metadata
5→avatar	8
<8, #1>	Fragment map
<8, #2>	Fragment map
<8, #3>	Fragment map
…	…
<8, #N>	Fragment map

Fragment map

T^2	Cache entry
…	…
…	…
T^2	Log entry #2
T^1	Log entry #1
T^0	Base entry

Fragment Map

Master index

Key	Value
8	inode metadata
5→avatar	8
<8, #1>	Fragment map
<8, #2>	Fragment map
<8, #3>	Fragment map
…	…
<8, #N>	Fragment map

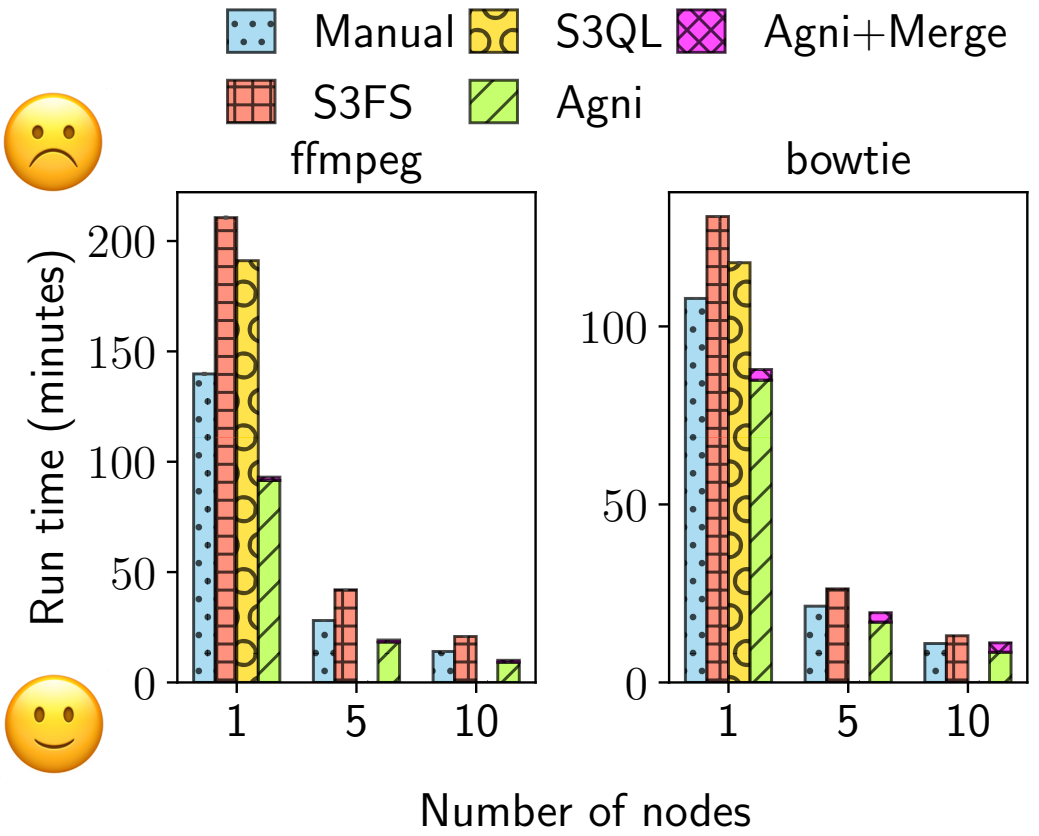
Fragment map

T^2	Cache entry
…	…
…	…
T^2	Log entry #2
T^1	Log entry #1
T^0	Base entry

Temporally indexed

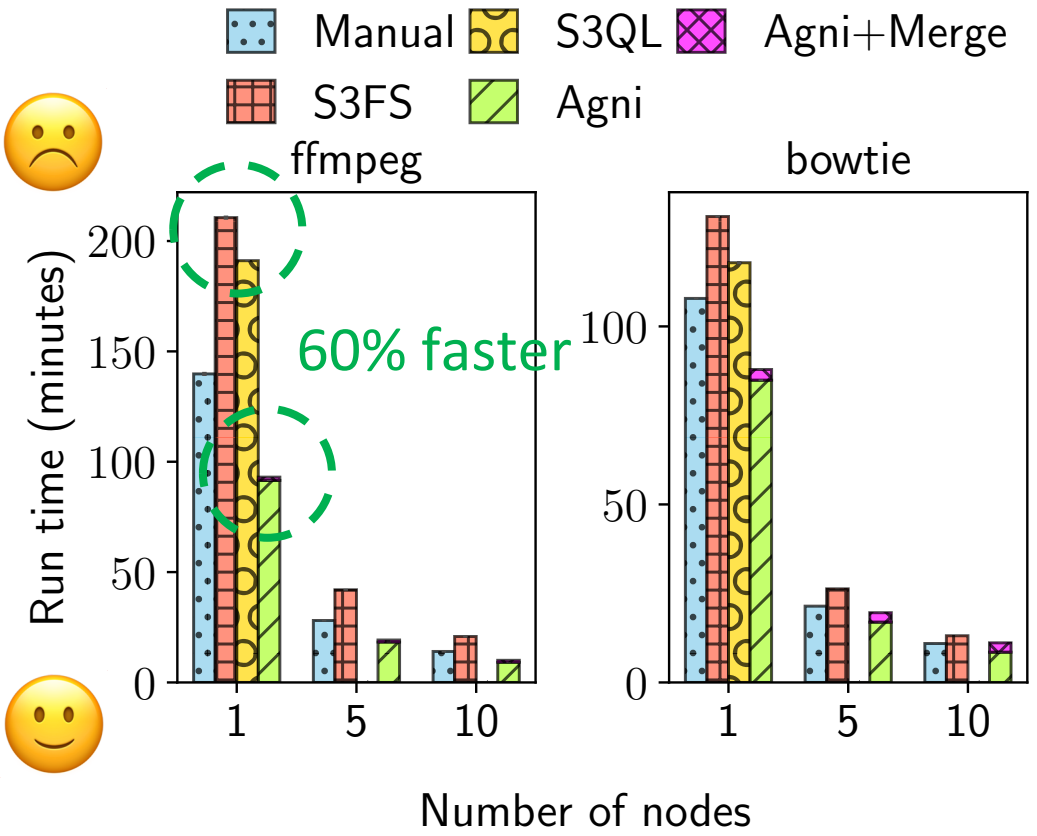
Evaluation: Applications

- ***ffmpeg***: 320 GB of MPEG to MOV files
- ***bowtie***: 80 GB genome files
- Agni+Merge denotes when dual access is enabled



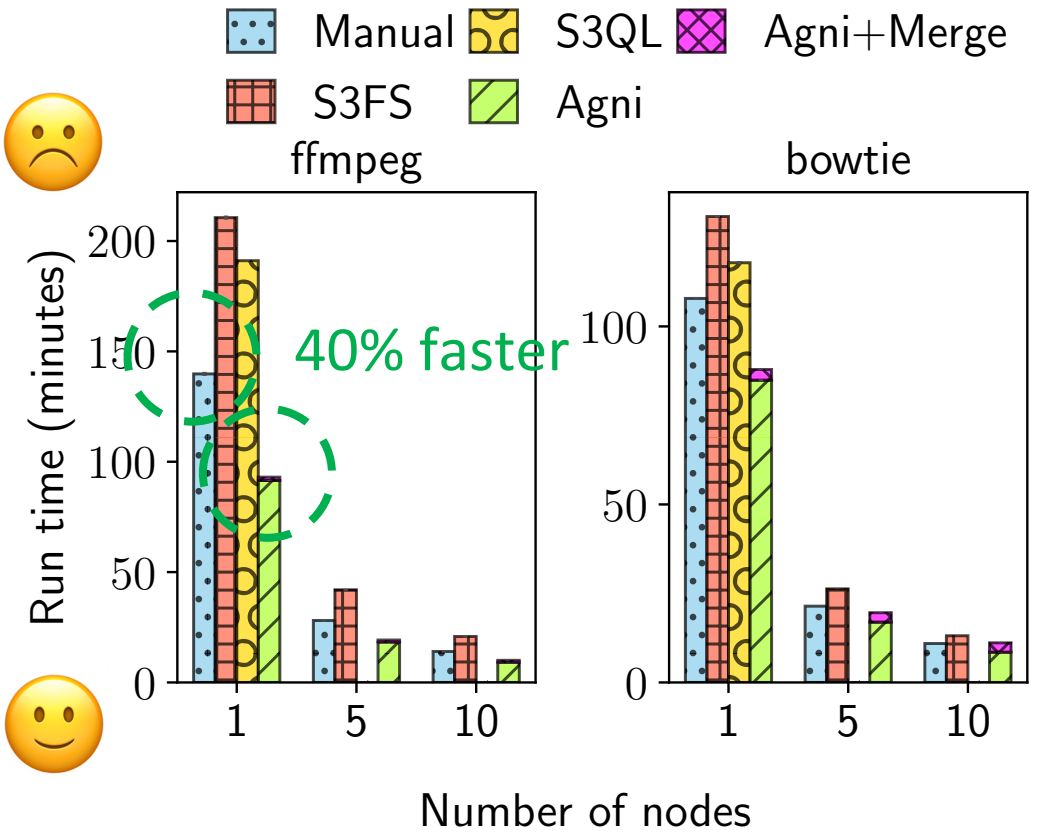
Evaluation: Applications

- ***ffmpeg***: 320 GB of MPEG to MOV files
- ***bowtie***: 80 GB genome files
- Agni+Merge denotes when dual access is enabled



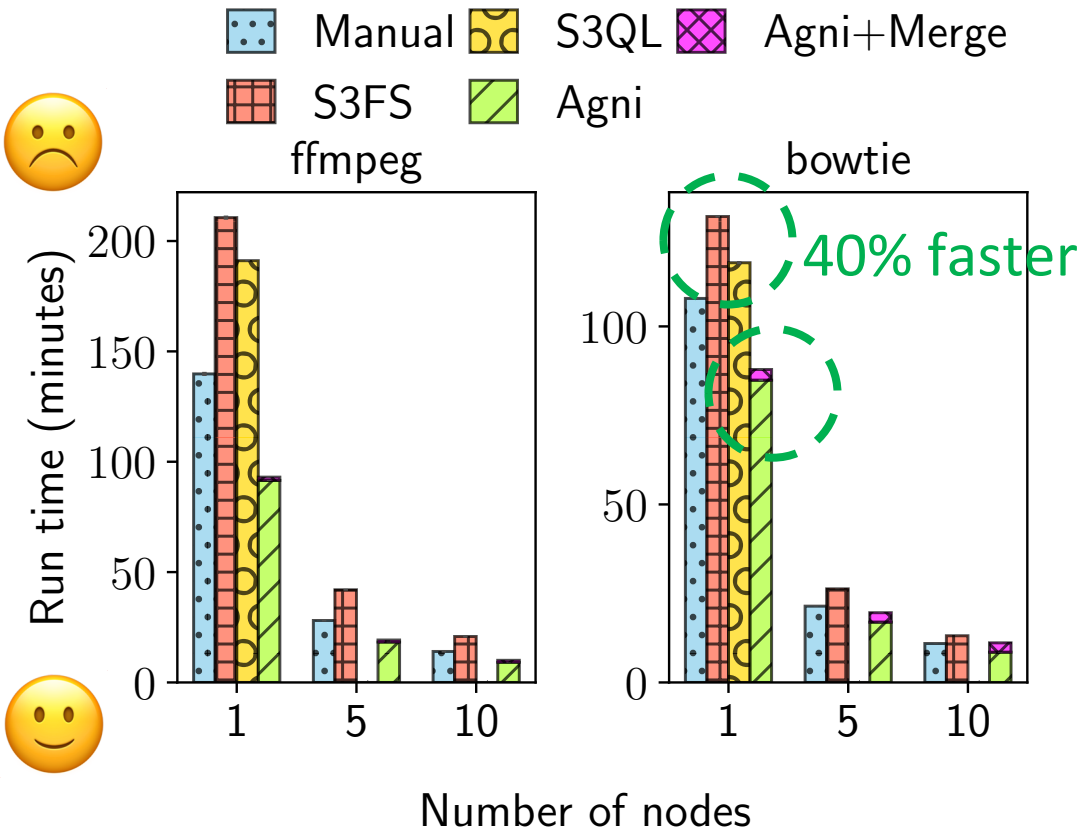
Evaluation: Applications

- ***ffmpeg***: 320 GB of MPEG to MOV files
- ***bowtie***: 80 GB genome files
- Agni+Merge denotes when dual access is enabled



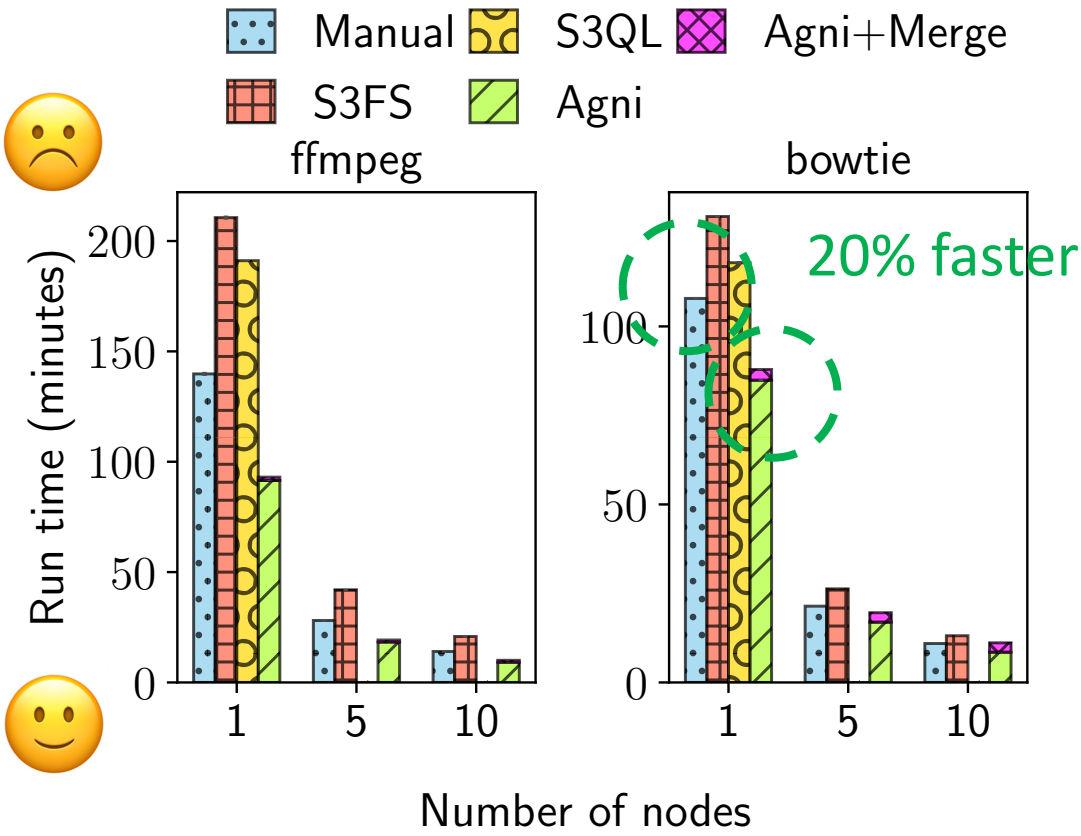
Evaluation: Applications

- ***ffmpeg***: 320 GB of MPEG to MOV files
- ***bowtie***: 80 GB genome files
- Agni+Merge denotes when dual access is enabled



Evaluation: Applications

- ***ffmpeg***: 320 GB of MPEG to MOV files
- ***bowtie***: 80 GB genome files
- Agni+Merge denotes when dual access is enabled



Summary

- Complete dual access with all desired features
- Cloud Neutral
- Outperforms existing dual-access systems
- Adding unified access control on roadmap


Thank you!
https://github.com/objectfs/objectfs
Contact: lillaney@jhu.edu