The Architecture of Inspur Cloud Storage

2013.7

Contents

Contents

INSPUR 浪潮

What is in the Cloud

Massive Data

create/obtain

Index/identify

search

share/collaborate

store/save

safe/protect

destroy/delete

web

table

electronic document

printout equipment

paper

Images, video

email

Business documents

faxes

instant messaging

data base

INSPUR 浪潮

What the Cloud Storage must support

- Massive Data
    GB, TB, PB, EB
- All Data Format
    Structured Data、Semi-Structured Data、
    Unstructured Data
- Multi Access Mode
    Block Device、S3、Swift、NFS、HDFS
- Data Safety
- Performance
- Advanced Data Management
- Even More…

Contents

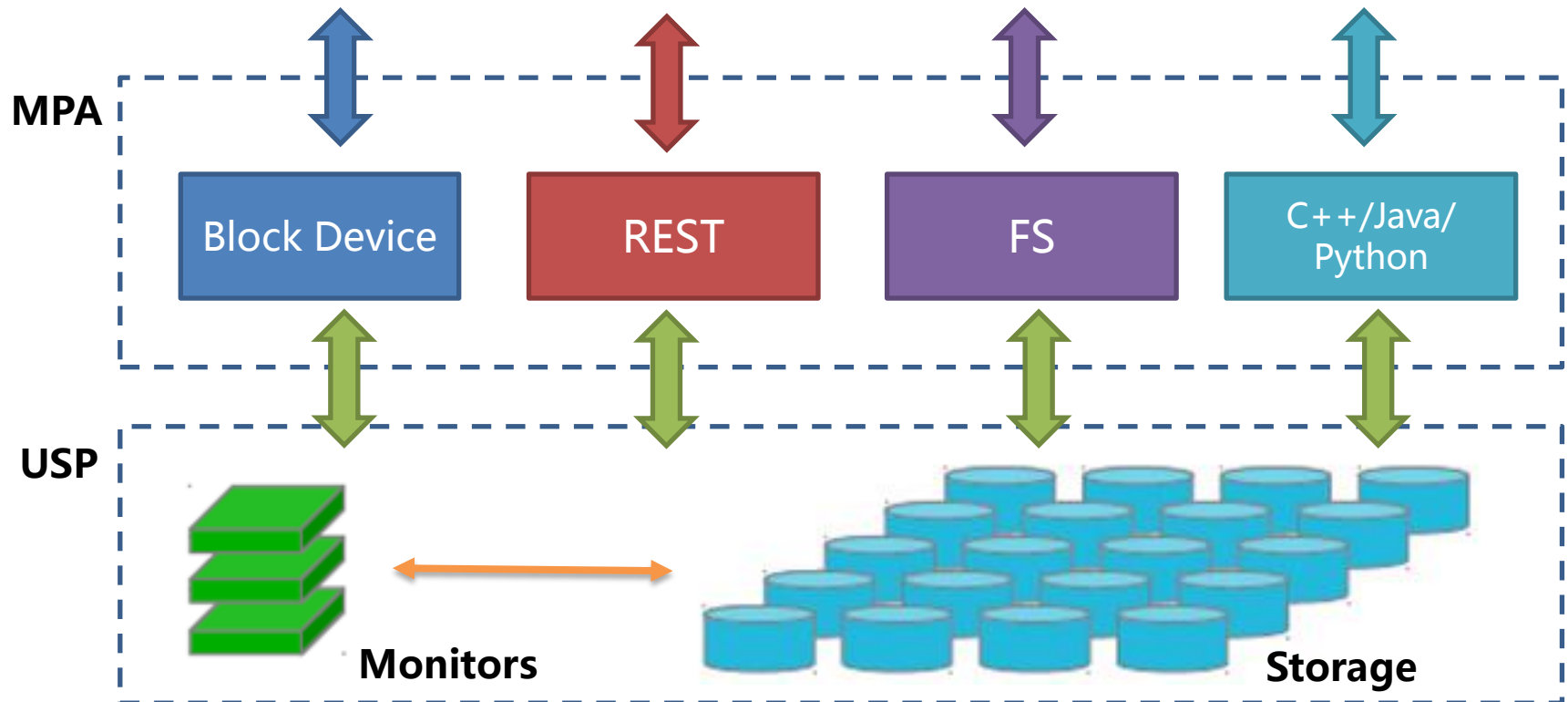*INSPUR* 浪潮

Our Solution

## Inspur Cloud Storage
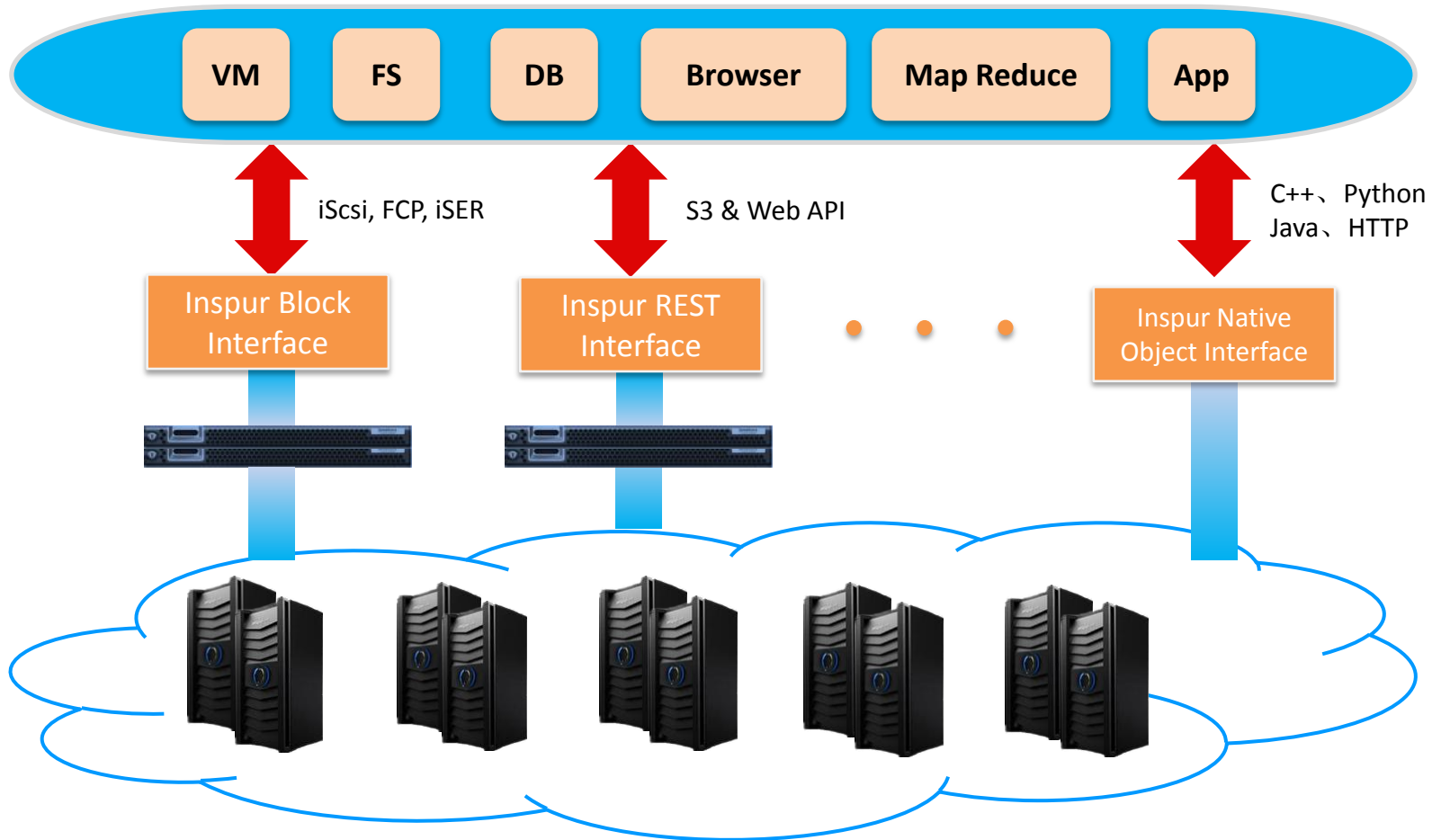
The Architecture

Unified Storage Pool + Multi-Protocol Access

- Unified Storage Pool: Integrate hardware resource
- Multi-Protocol Access: Block Device, REST(S3, Swift), FS, etc...

MPA

| Block Device | REST | FS | C++/Java/Python |

USP

Monitors

Storage

Key Components: Unified Storage Pool

USP provides service by object

- Each object has a unique name （object id）
- Object is located by a Pseudo-Random Consistent Hashing
- No relations between objects and Fully Scale-Out

object

$$PRCH(object\ id) = \vec{r}$$

object storage
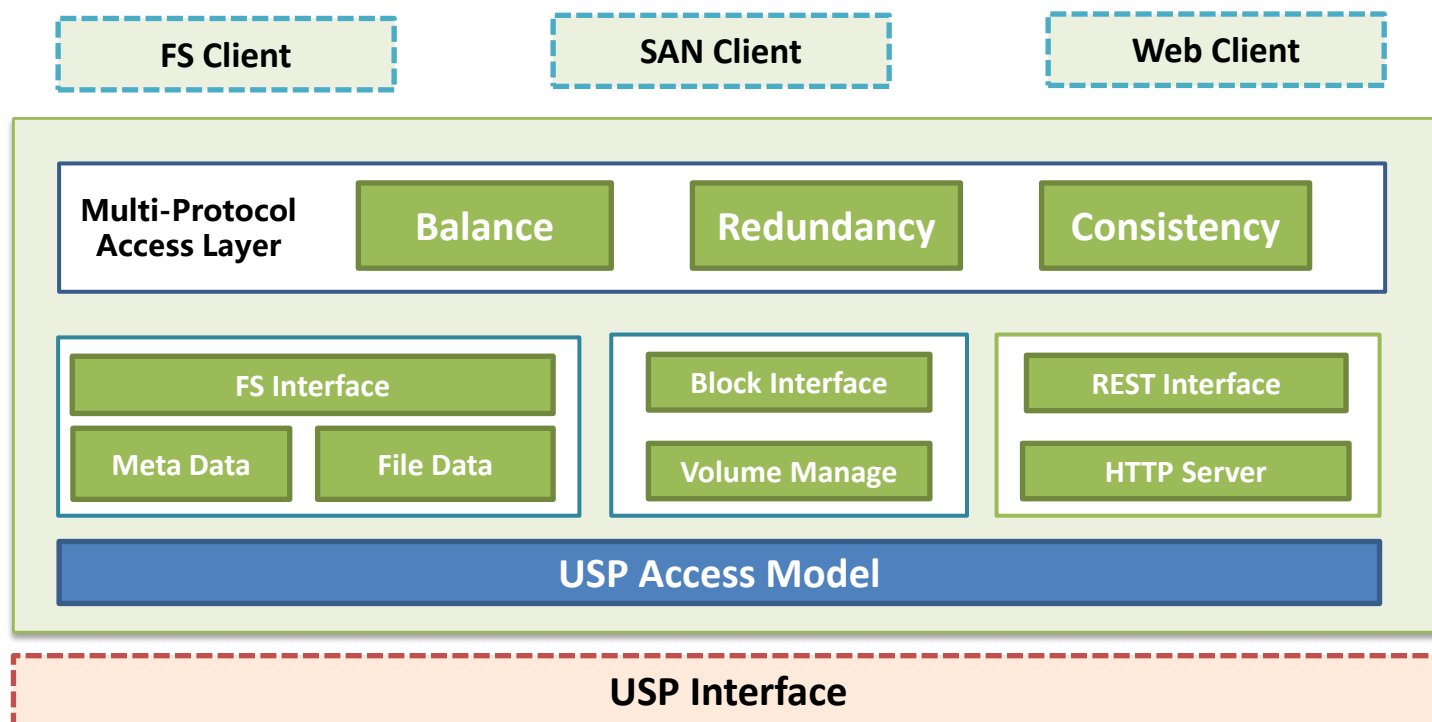devices
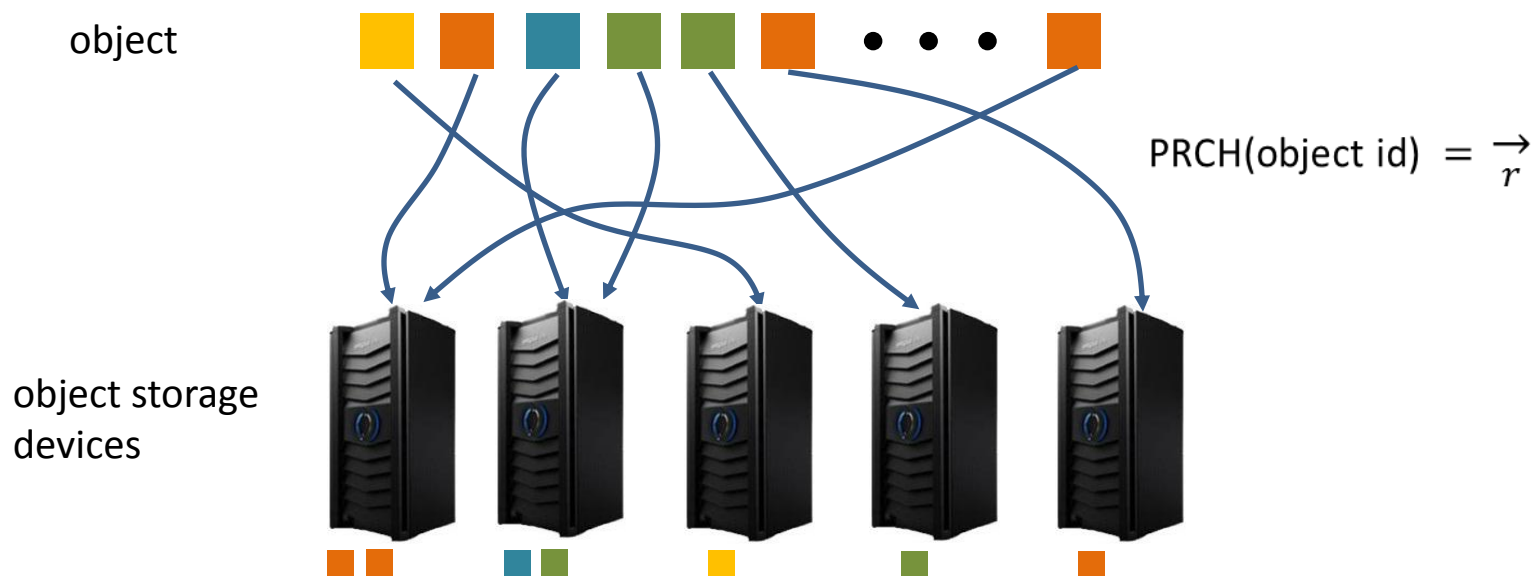
Key Components: Unified Storage Pool

USP distribute data to different rack, node and disk.

- The whole cluster is described by a topological diagram which we called NodeMap, while NodeMap is used for object location algorithm.
- We treat hardware resource changes including expansion, reduce and breakdown as change of NodeMap.
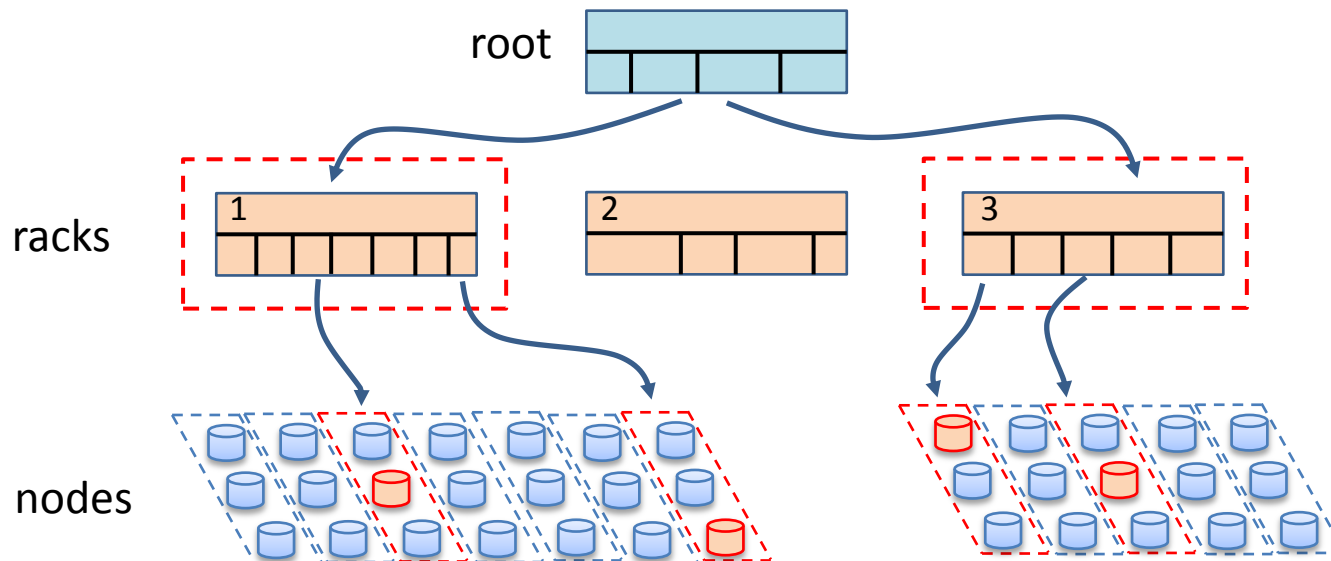
Multi-Protocol Access

- Multi-Protocol Access Layer accepts client requests with responsibility for common functions such as Balance, Line Redundancy and Consistency.
- Different interfaces handle different request format with the same USP Access Model
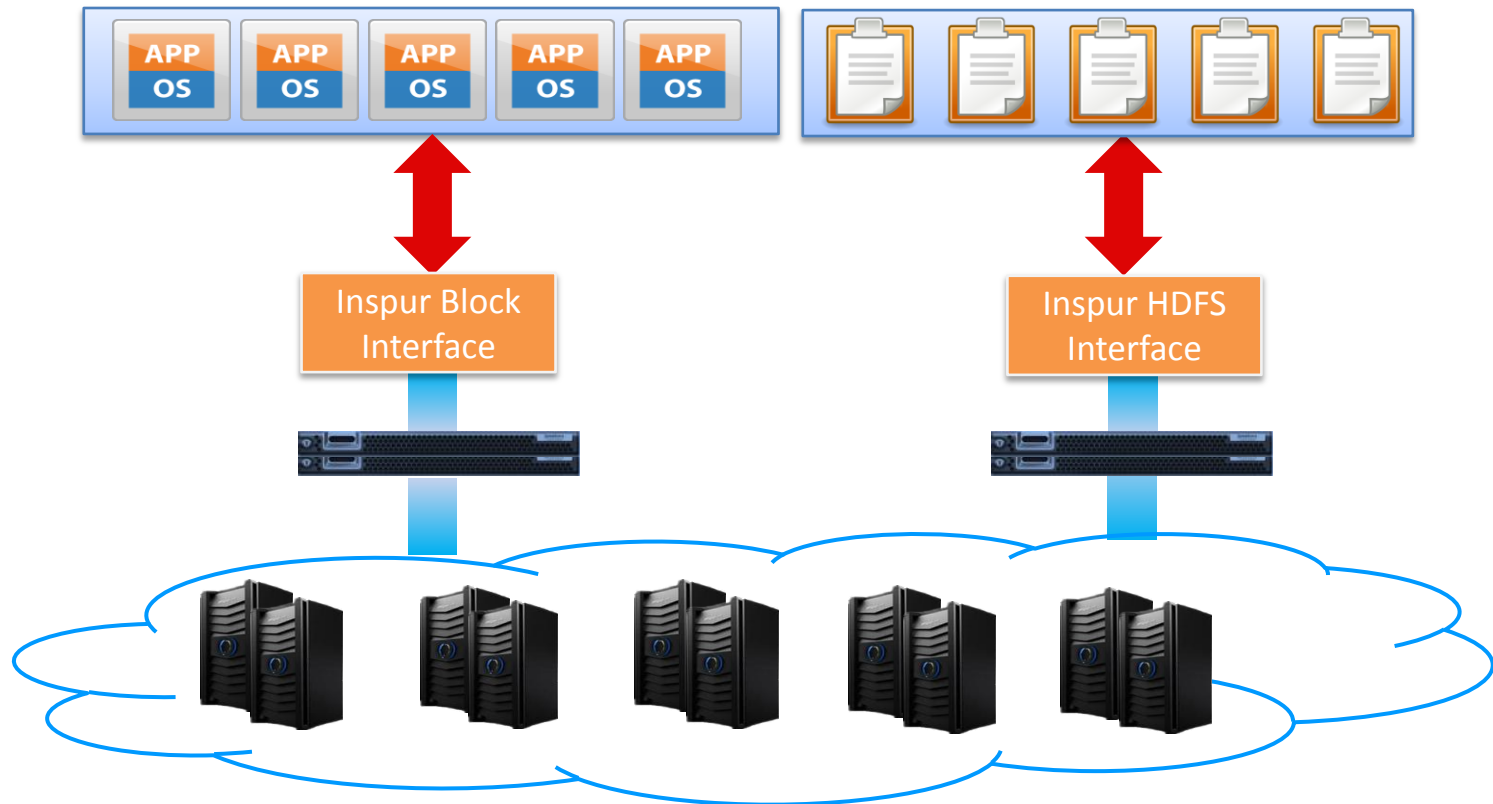
Contents

INSPUR 浪潮

Cloud Computing System Based on Cloud Storage

Two Typical Modes In Cloud Computing

- Virtual Machine(VM Ware, KVM, Xen)

- Big Data Analyze（MapReduce）

Inspur Block Interface

Inspur HDFS Interface
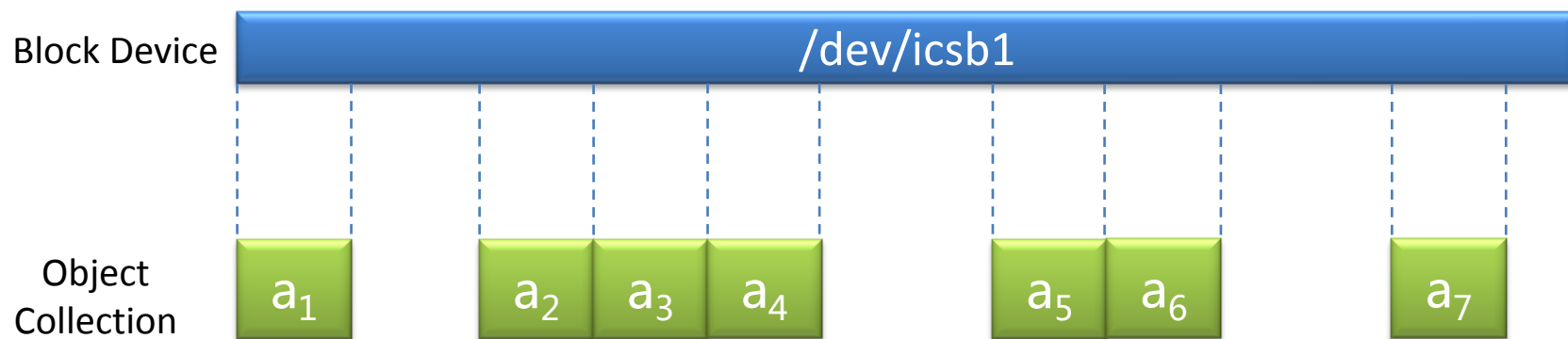
Block Device Access

- Block Device Access Model combines object collection into a standard Linux block device.

- Thin-provisioning, Clone, Isolation and Access Control is supported

MPA

Block Device

Thin-provisioning   Clone   Isolation

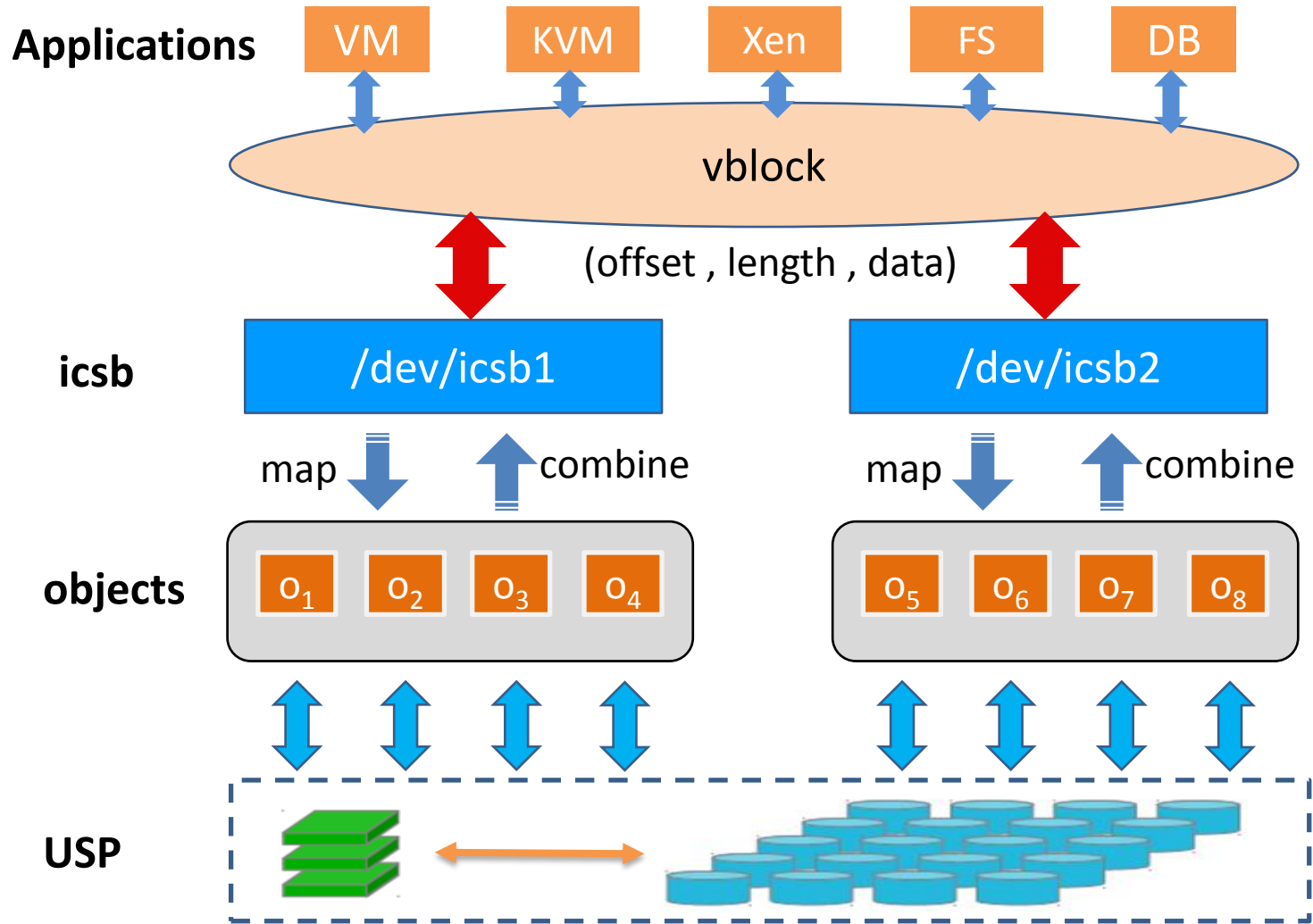**USP**

Monitors

Storage

INSPUR 浪潮
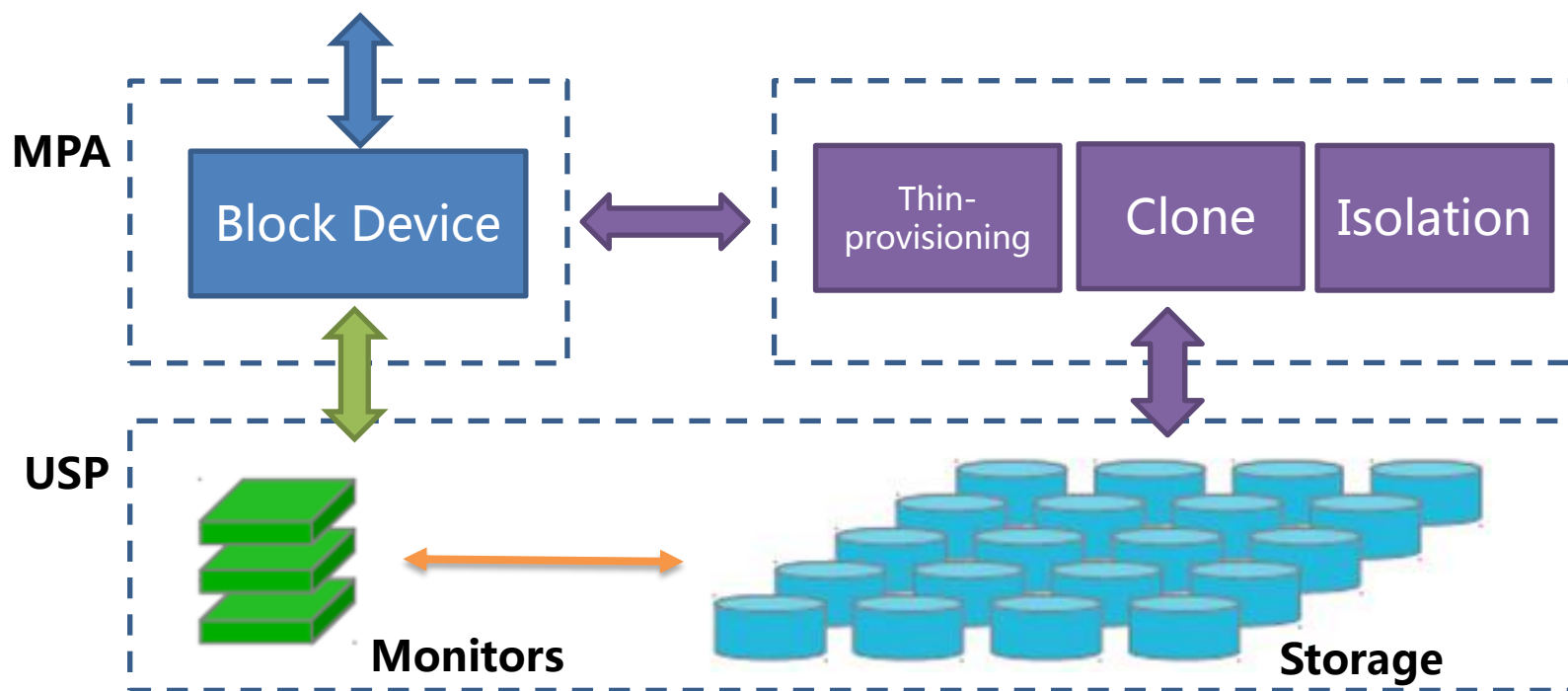
Block Device Access

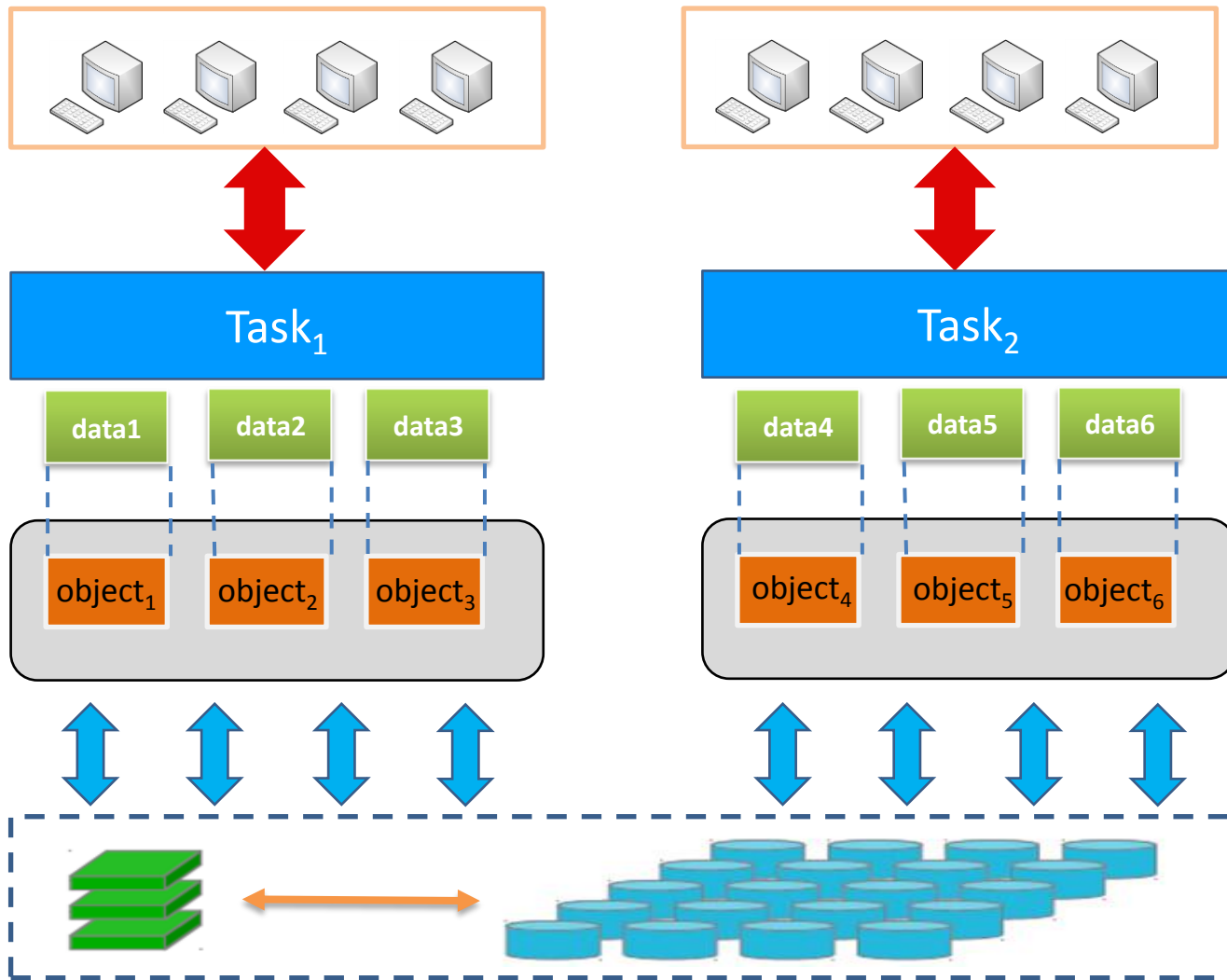The Key of Block Device Access Model is Sparse Data Map Algorithm

- BDAM is implemented as a Linux driver(icsb) and the virtual block device meets the vBlock interface

- Sparse Data Map Algorithm maps the vBlock request(offset, length) to Object request(object id, offset, length)

Block Device Access

Map Reduce

Contents

1 **What is the problem**

2 **What is our solution**

The Architecture

Key Components
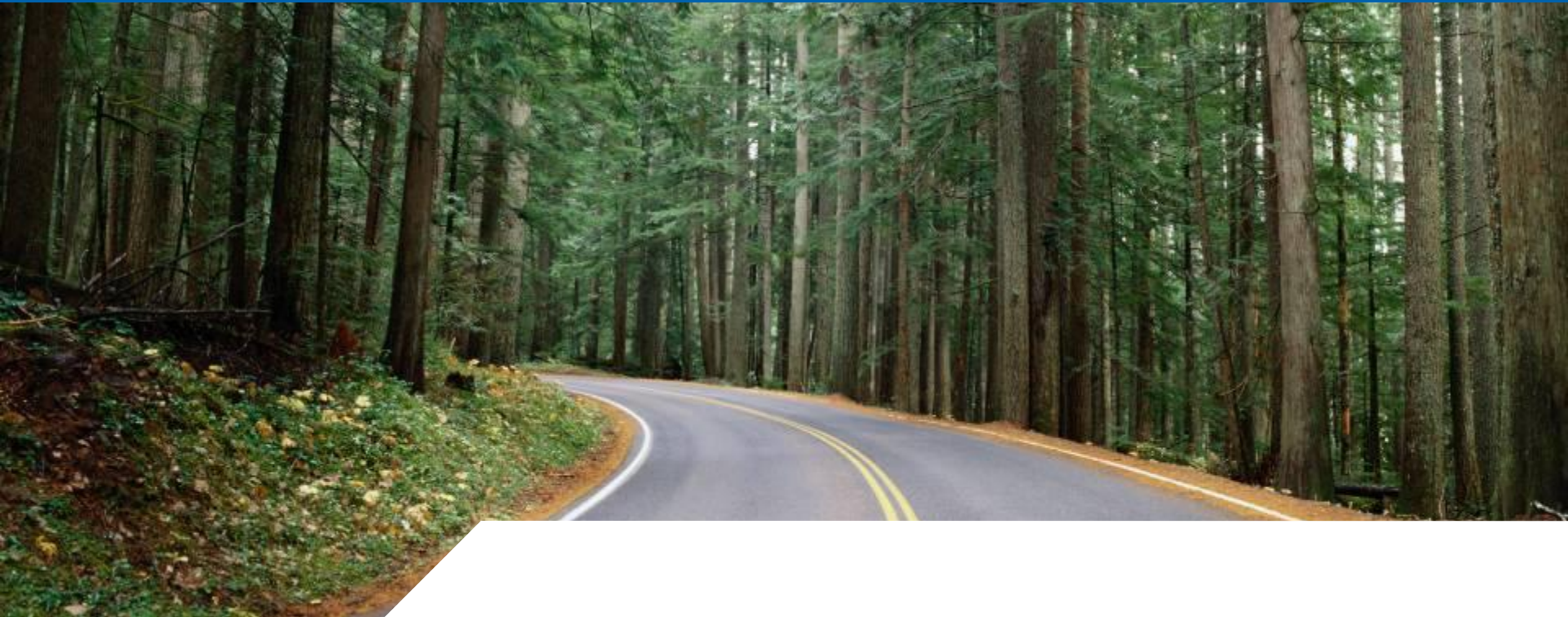
3 **Conclusion**

INSPUR 浪潮

Conclusion

Key Points

- Unified Storage Pool + Multi-Protocol Access
- USP: Object is Located by Pseudo-Random Consistent Hash
- MPA: Different interfaces with the same USP Access Model
- Cloud Computing: Block Device Access & HDFS API

Capability

- Exabyte Data or Even More
- TB/s Bandwidth
- Millions of Concurrent Request

Thank you