



QoS Provisioning Framework for OSD-Based Storage System

Yingping Lu, David Du, Thomas Ruwart
DTC Intelligent Storage Consortium (DISC)
University of Minnesota

MSST 2005



Outline

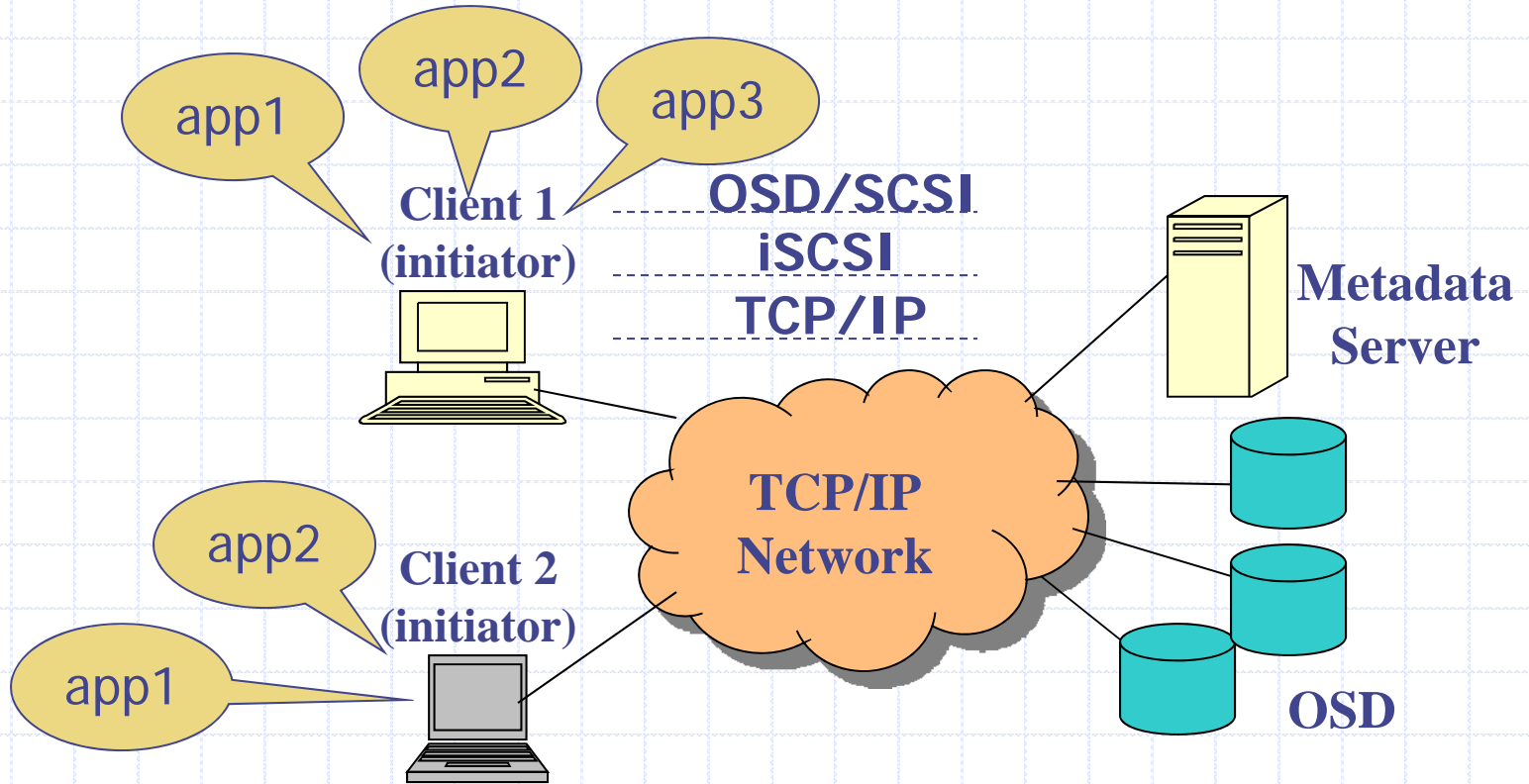
- ◆ Motivation
- ◆ QoS Requirements in OSD
- ◆ QoS Specification
- ◆ QoS Enforcement
- ◆ On-going Work
- ◆ Summary



Motivation

- ◆ OSD is an emerging storage technology
- ◆ Different applications require different Performance guarantees: bandwidth, response time, throughput, etc.
- ◆ OSD has its new features:
 - Objects in OSD carries application semantics
 - OSD has knowledge of its storage condition
 - OSD can probe network conditions

A Typical OSD-Based Storage





Why QoS in OSD?

- ◆ Application requirement
- ◆ Initiator SLA requirement (storage consolidation)
- ◆ Network resource limitation
- ◆ Storage resource limitation



Applications with QoS Needs

- ◆ Real-time applications
 - Mainly in streaming application
 - Guarantee timely delivery
 - Soft vs. hard time guarantee
 - Often retrieve data periodically
- ◆ Interactive applications
 - Requires latency guarantees
- ◆ Best effort applications
 - Throughput, e.g. FTP application



Related Work

- ◆ Network QoS
- ◆ Real-time storage scheduling
- ◆ Feedback-control mechanism
- ◆ Network dynamics study
- ◆ The emerging of object storage device



QoS Framework for OSD Access

- ◆ QoS requirements
- ◆ QoS specification
- ◆ QoS admission control
- ◆ Resource monitoring
- ◆ QoS enforcement



QoS Request Classes

- ◆ Application level
 - Object level
 - Object session level
 - Object operation level
- ◆ Session level
- ◆ Best-effort



Object level

- ◆ Represent inherent QoS requirement of a particular object, e.g. a streaming video object, video surveillance object
- ◆ Define a default QoS requirement for the object
- ◆ May be overridden by explicit QoS specification

Object Session Level

- ◆ Represent the QoS requirement of data access within **a period of time**
- ◆ The specified QoS requirement only applies to the specified time period
- ◆ Application specifies the start and end of the "session"
- ◆ Useful for applications with certain bandwidth requirement at certain period of time, e.g. a multimedia distant learning application.
- ◆ Note: this "session" is different from iSCSI Session



Operation Level

- ◆ Represent QoS requirement for a particular operation
- ◆ QoS requirement (e. g. deadline) only applies to the specific operation
- ◆ Useful for interactive application, e.g. a streaming control operation, or zooming in a point or show a particular data point in scientific study, etc.
- ◆ It can be treated as a special case of object session

Session Level

- ◆ Applies to an initiator session
- ◆ The QoS metrics can be bandwidth, delay. Usually represented by SLO
- ◆ Provides the default QoS requirements for all access across a session.
- ◆ The application-level QoS takes precedence to session level.

Providing QoS Specification

◆ Object-level QoS Attributes Page

Attribute number	Length (bytes)	Attributes	Client settable	OSD logical unit
0h	40	Page ID	No	Yes
1h	20	Bandwidth	Yes	No
2h	20	Delay	Yes	No
3h- FFFFFFE		Reserved	No	



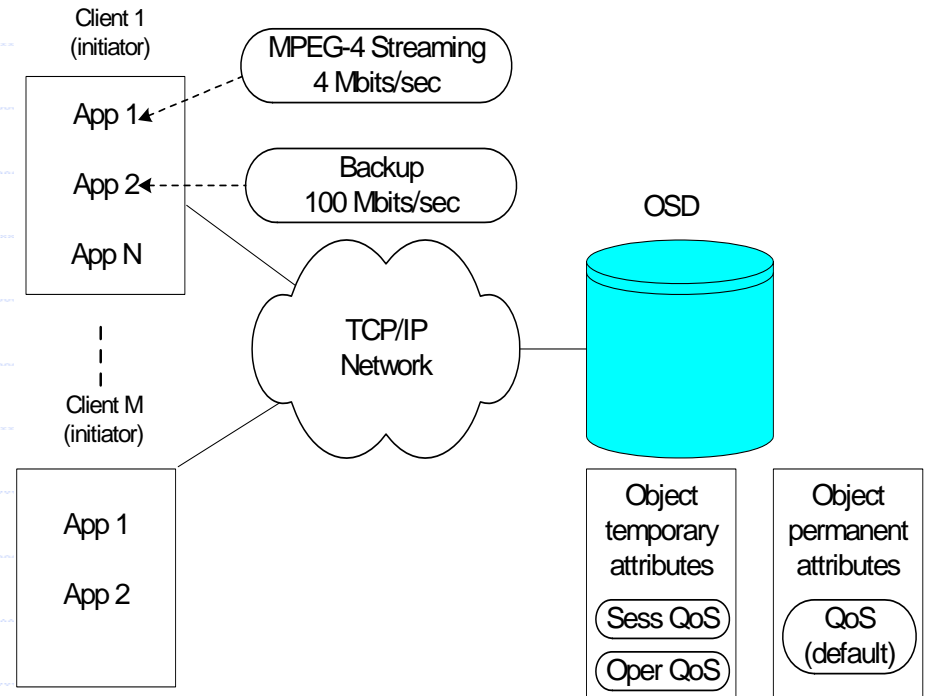
Transient QoS Attributes

- ◆ Transient attributes vs. permanent attributes
 - Transient attributes only affect the current operation or session
- ◆ Object session specification
 - Time to Live (TTL)
 - Operation Count (OC)
 - SessionActive switch

Setting QoS Attributes

Get/Set CDBFMT

value	Desc
00b	Operation
01b	Obj session
10b	Get/set value
11b	Get/set list



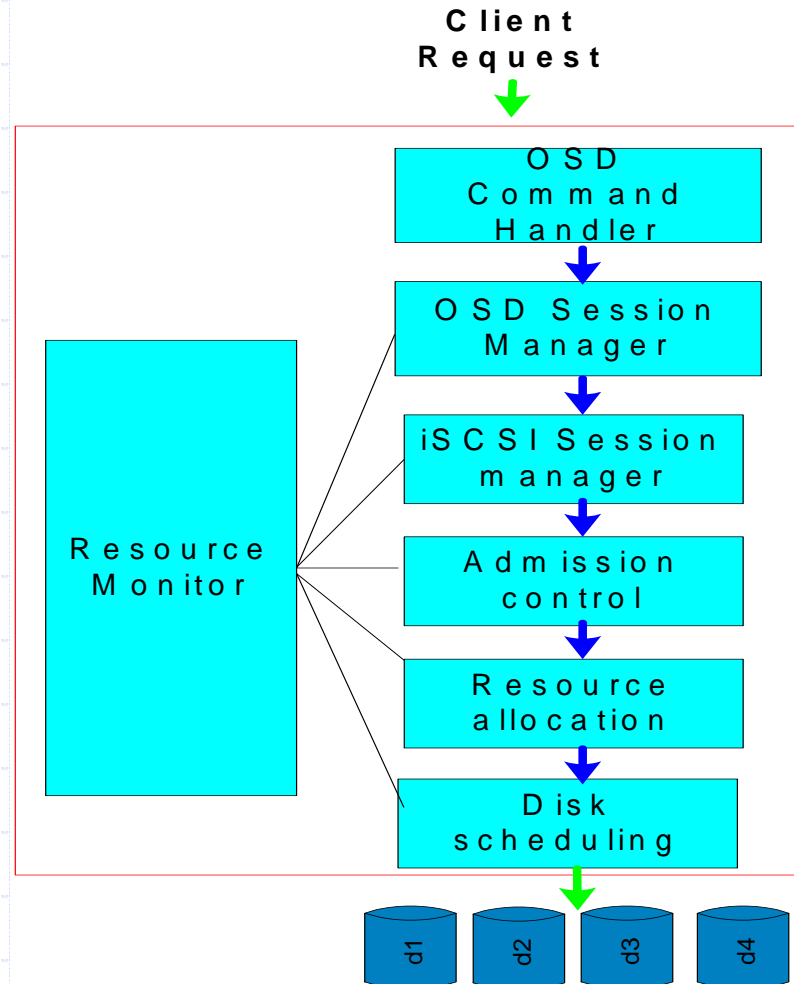


Specifying Session Level QoS

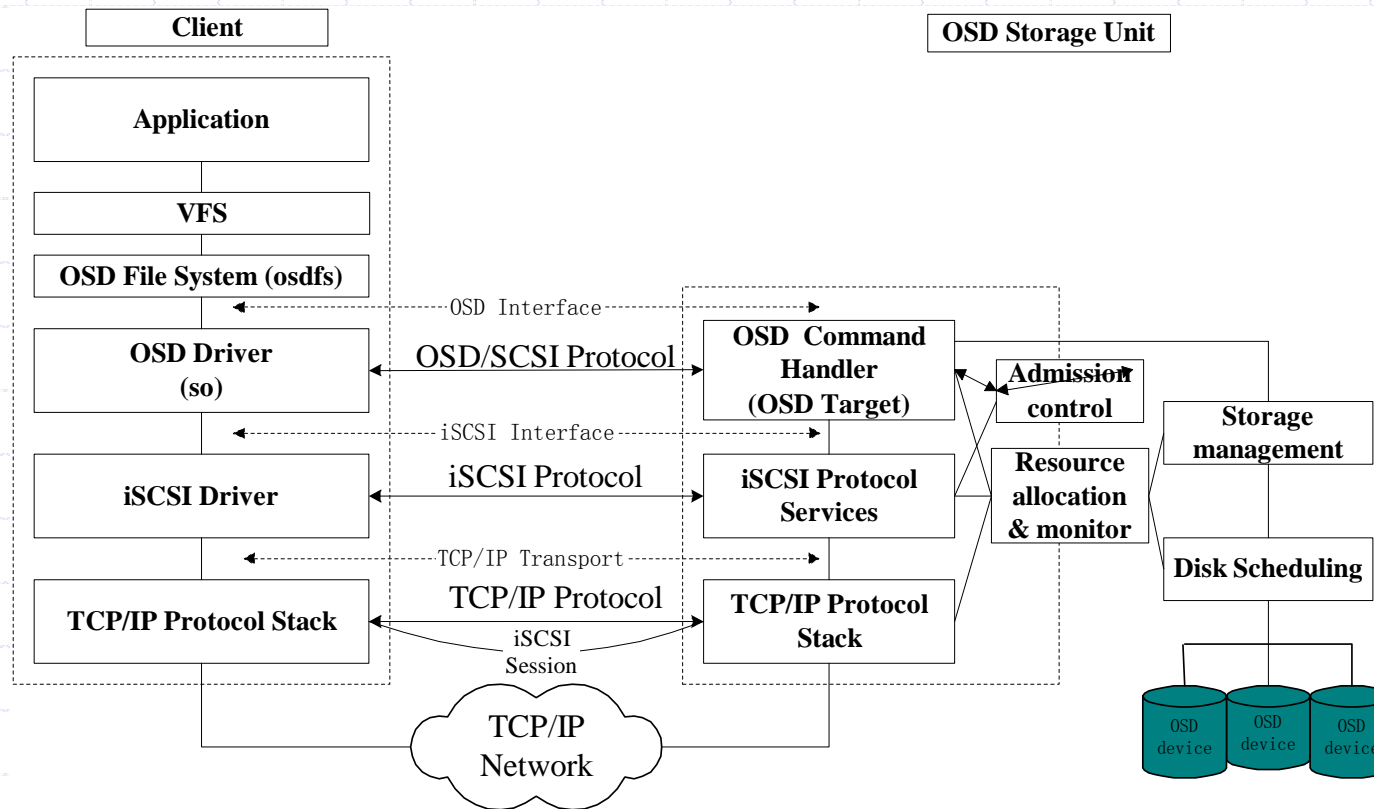
- ◆ Added new QoS parameters in iSCSI session.
- ◆ The QoS parameters include the bandwidth range or delay range.
 - MaxBandwidth, MinBandwidth
 - MaxDelay, MinDelay
- ◆ The QoS parameters are negotiated during session setup.

QoS Enforcement Architecture

- ◆ There are mainly seven components in the target to enforce QoS
- ◆ Disk scheduling ensure the differentiation of different QoS requirements



Reference Implementation





Summary

- ◆ OSD provides new incentive for QoS provisioning
- ◆ Three main QoS specification classes are identified
- ◆ OSD protocol extended to support the specification of QoS requirements
- ◆ Reference implementation is on-going to incorporate the specification and enforcement effort

Questions?



Thanks!