Objectivity

301B East Evelyn Avenue
Mountain View, CA 94041
1-650-254-7100

-

MRD for Container to File Mapping.

Approval

Department	*Name*	*Date*
Development		
Marketing		
Sales		
Support		

Modification History

Version	*Date*	*Author*	*Comments*
1.0	17-Apr-02	Brian Clark	Initial Draft

Feature Requirements

Description of the Problem

Objectivity claims scalability as one of its distinguishing features. While this is logically true with the 64 bit OID, there is a physical limitation. We can have 2^16 – 1 databases in a federation, 2^15 – 1 containers in a database, 2^16 – 1 logical pages in a container, and 2^16 – 1 slots in a page. Taking the maximum page size of 64Kb this gives a total address space of 64K (databases) * 32K (containers) * 64K pages * 64K (page size) = 8 Exabytes.

However each database file would have to be 128 Terabytes. Nobody knows how to handle a single file this size, not even 128 Gigabytes!

We need a lower level of granularity to map to a single file.

Additional benefits:

- Finer granularity of distribution.
- Finer granularity of replication.
- Finer granularity of security.
- Performance improvements.
    - Increased throughput if multiple updaters are working on different containers in a database.
    - Could enable virtual memory mapping of container files
- Implements original intent of the architecture.

Description of the requested features

- Currently Objectivity databases map to operating system files. This request is to optionally map a container to file.

Part of an optional feature or does it require another feature? If so, which one?

- Standard Feature

How is this problem being solved now, and why isn't that acceptable?

Currently Objectivity databases map to operating system files. This has the physical size limits detailed above. It also limits the granularity of any operating system assisted security mechanism.

Which languages must support this capability?

This is a kernel change.

Which platforms must be supported?

All.

Do any competitors already have this capability?

No.

Benefit Category:

Performance, scalability, security and usability.

Customers who require this capability:

All.

Revenue at risk or which could be won:

Future VLDB projects would benefit from this improvement.

When is this required?

R8.0

Additional Notes

The facility should be designed to incorporate multiple files per database with multiple (one or more whole) containers per file. This is not essential for the first release as it will require some additional tools.

Review

Feature Sizing

Efforts	*Size*
Development	
QA	
Documentation	

Scheduled for Objy Release
Assigned Engineering Group