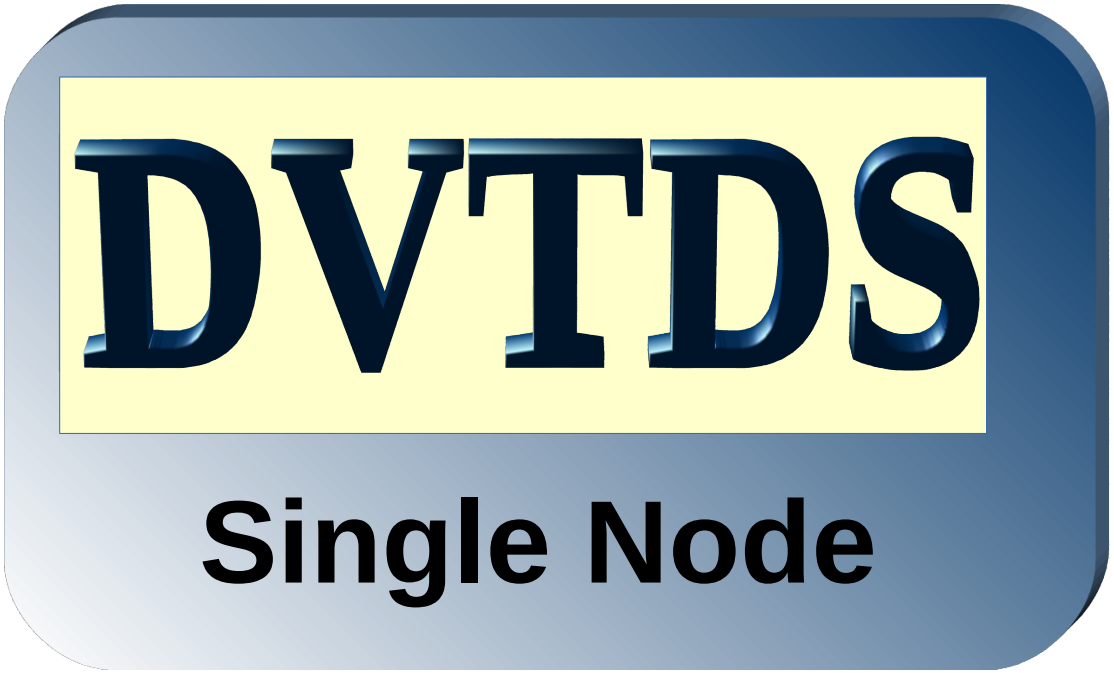
TeraCortex

DVTDS Architecture

Guided Tour

March 2018

You can start with a single data base node

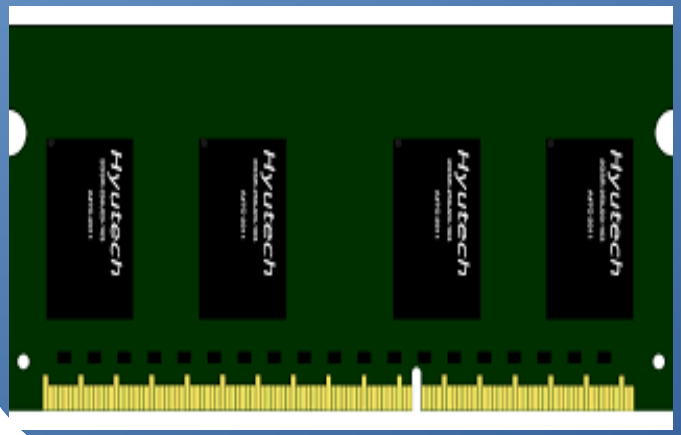


Your interactions are transaction safe

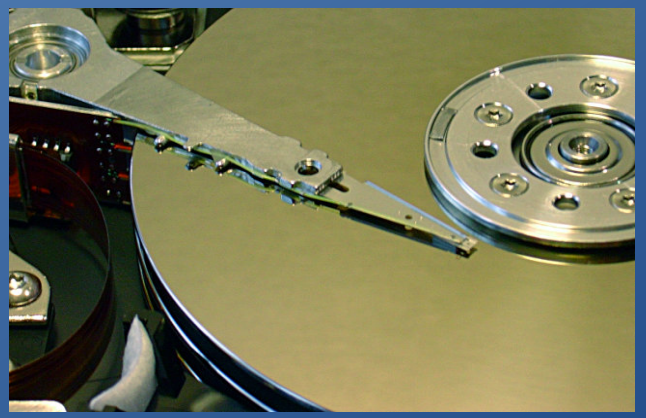
Database

Your node supports hybrid storage

RAM



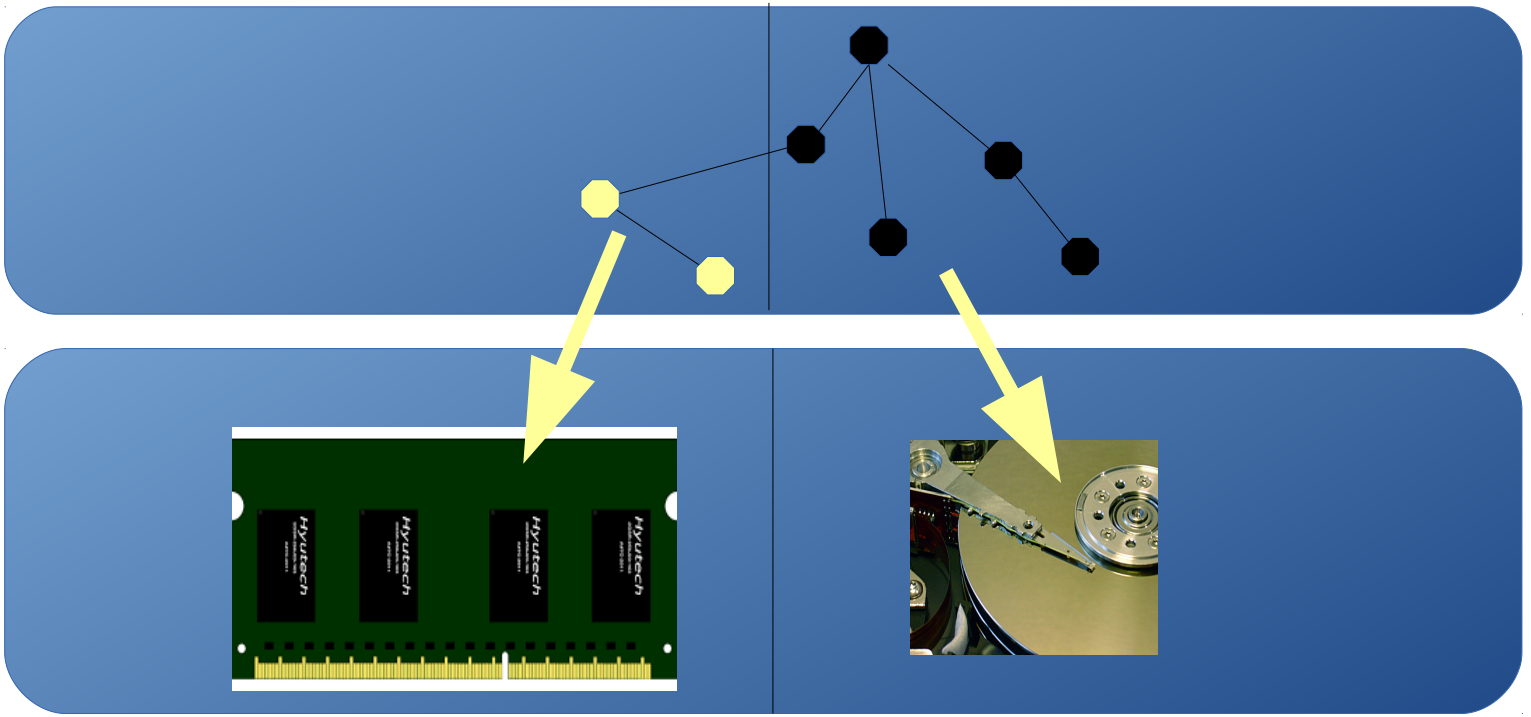
Hard Disk



Your data is stored in object trees

Physical Object Layer

Storage Layer



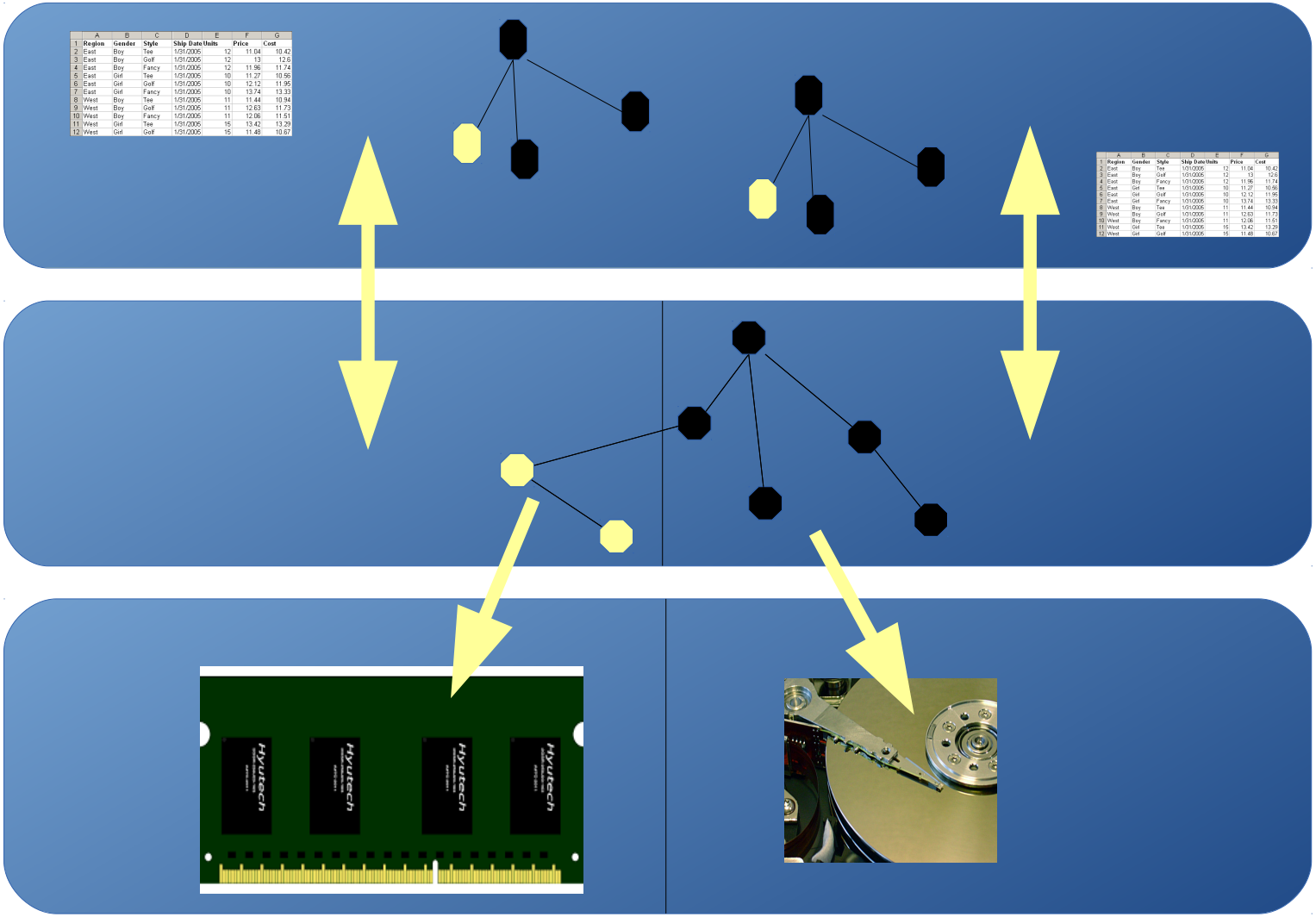
High frequency volatile data is kept in memory. Persistent data is kept on disk. Large volume data (image, video) is stored in file system locations (local or NAS)

Your clients see data in their own model, tables or trees

Client Data Layer

Physical Object Layer

Storage Layer



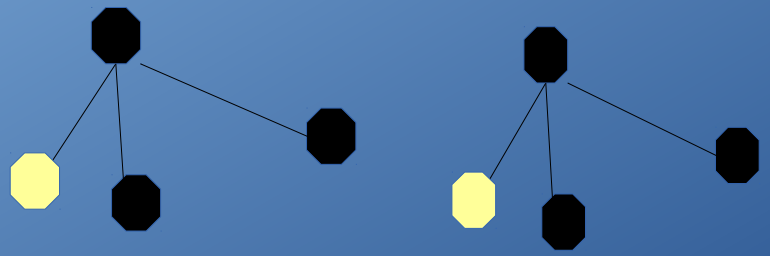
Clients need no knowledge of physical data distribution and storage types. But may decide to engage into it by means of LDAP controls

Your clients connect via interfaces

Protocol Layer

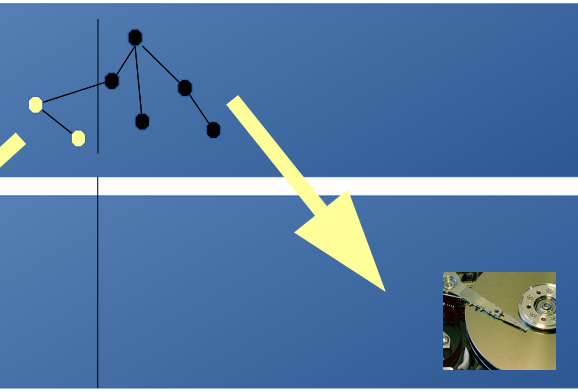
LDAP | LDIF | ASN.1 | HTTP | CSV

Client Data Layer

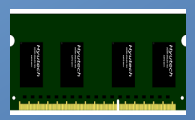


	A	B	C	D	E	F	G
1	Region	Gender	Style	Ship Date	Units	Price	Cost
2	East	Boy	Tee	1/31/2005	12	11.04	10.42
3	East	Boy	Golf	1/31/2005	12	13	12.6
4	East	Boy	Fancy	1/31/2005	12	11.96	11.74
5	East	Girl	Tee	1/31/2005	10	11.27	10.56
6	East	Girl	Golf	1/31/2005	10	12.12	11.95
7	East	Girl	Fancy	1/31/2005	10	13.74	13.33
8	West	Boy	Tee	1/31/2005	11	11.44	10.94
9	West	Boy	Golf	1/31/2005	11	12.63	11.73
10	West	Boy	Fancy	1/31/2005	11	12.06	11.51
11	West	Girl	Tee	1/31/2005	15	13.42	13.29
12	West	Girl	Golf	1/31/2005	15	11.48	10.67

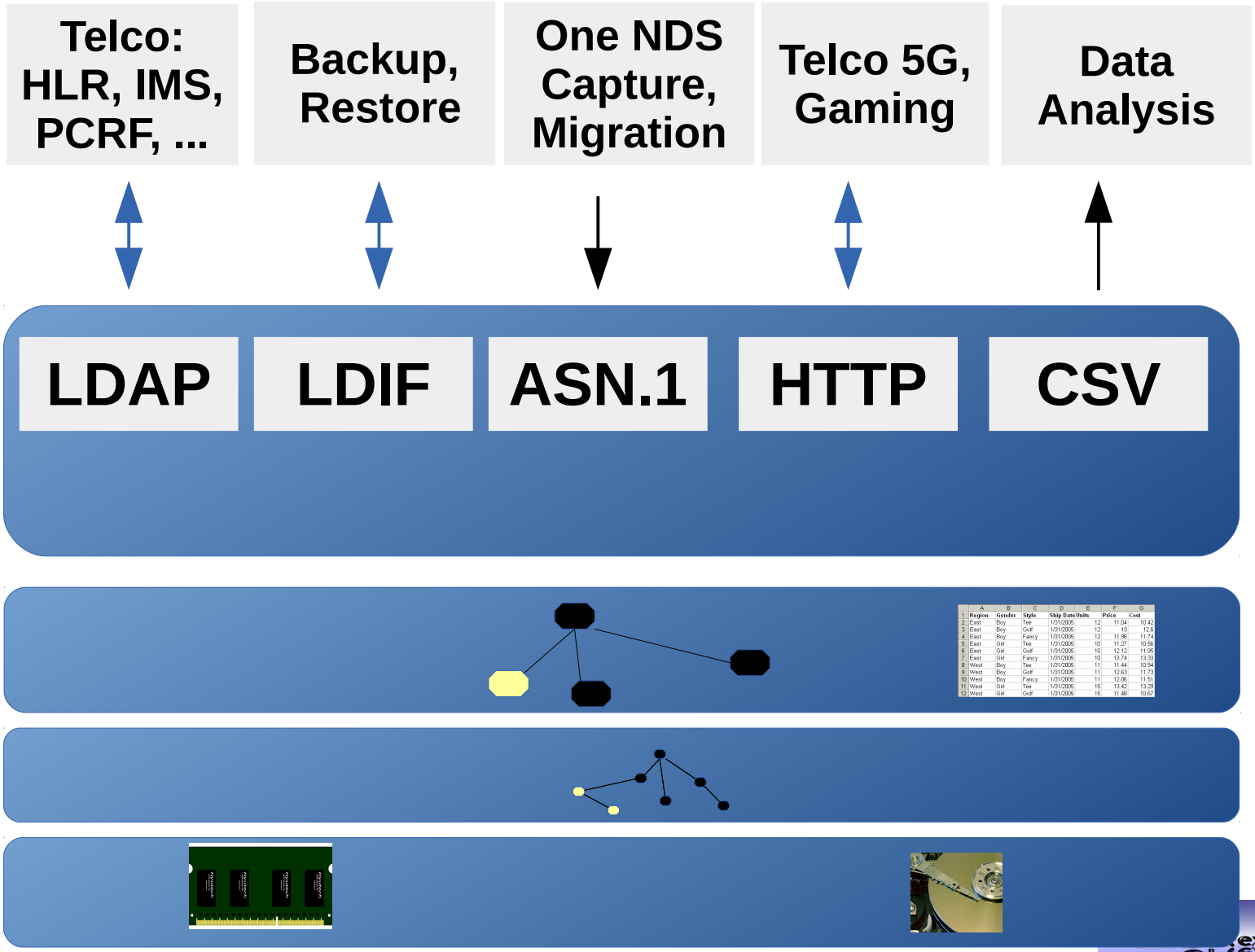
Physical Object Layer



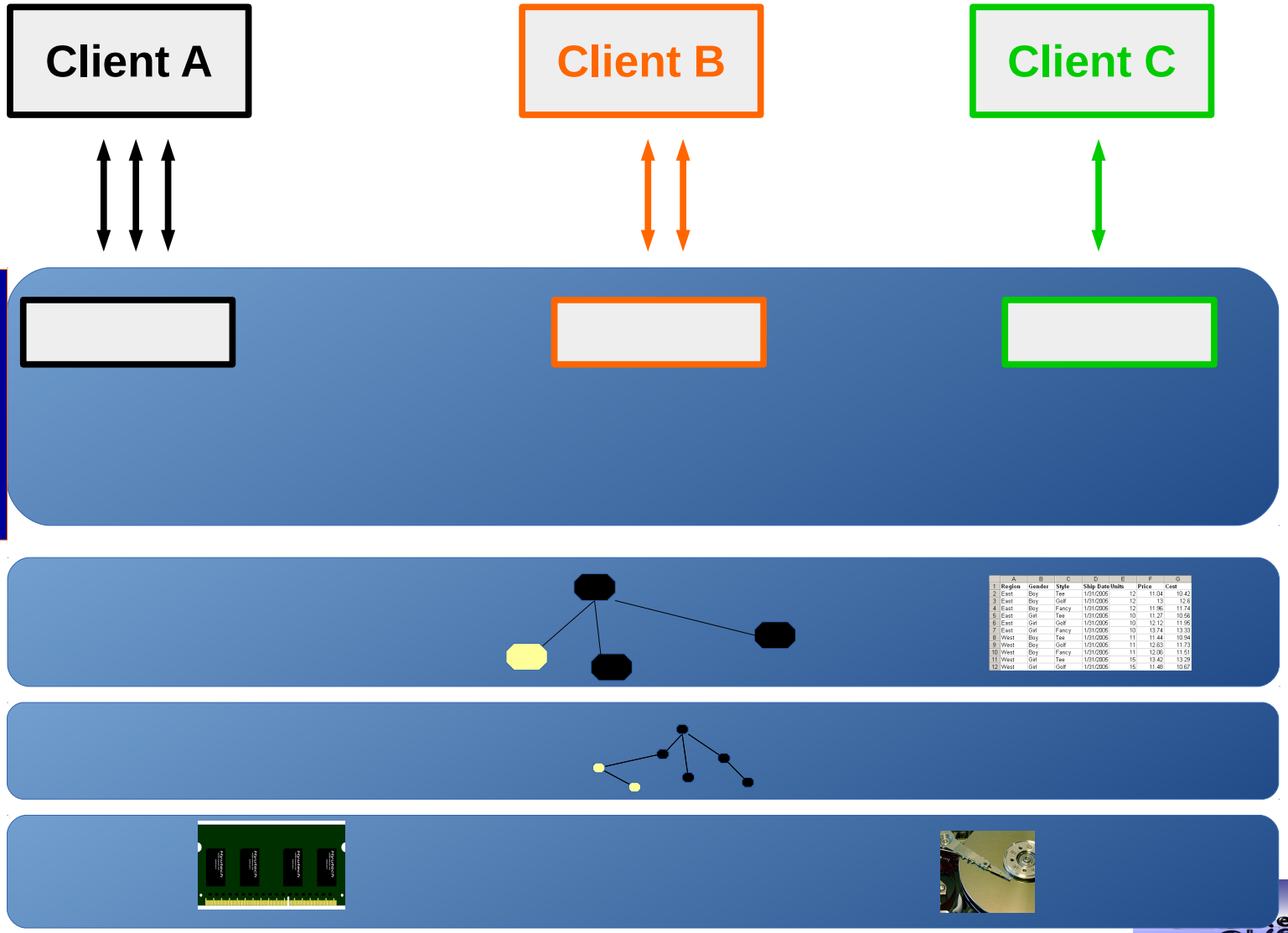
Storage Layer



You run real time applications against the interfaces

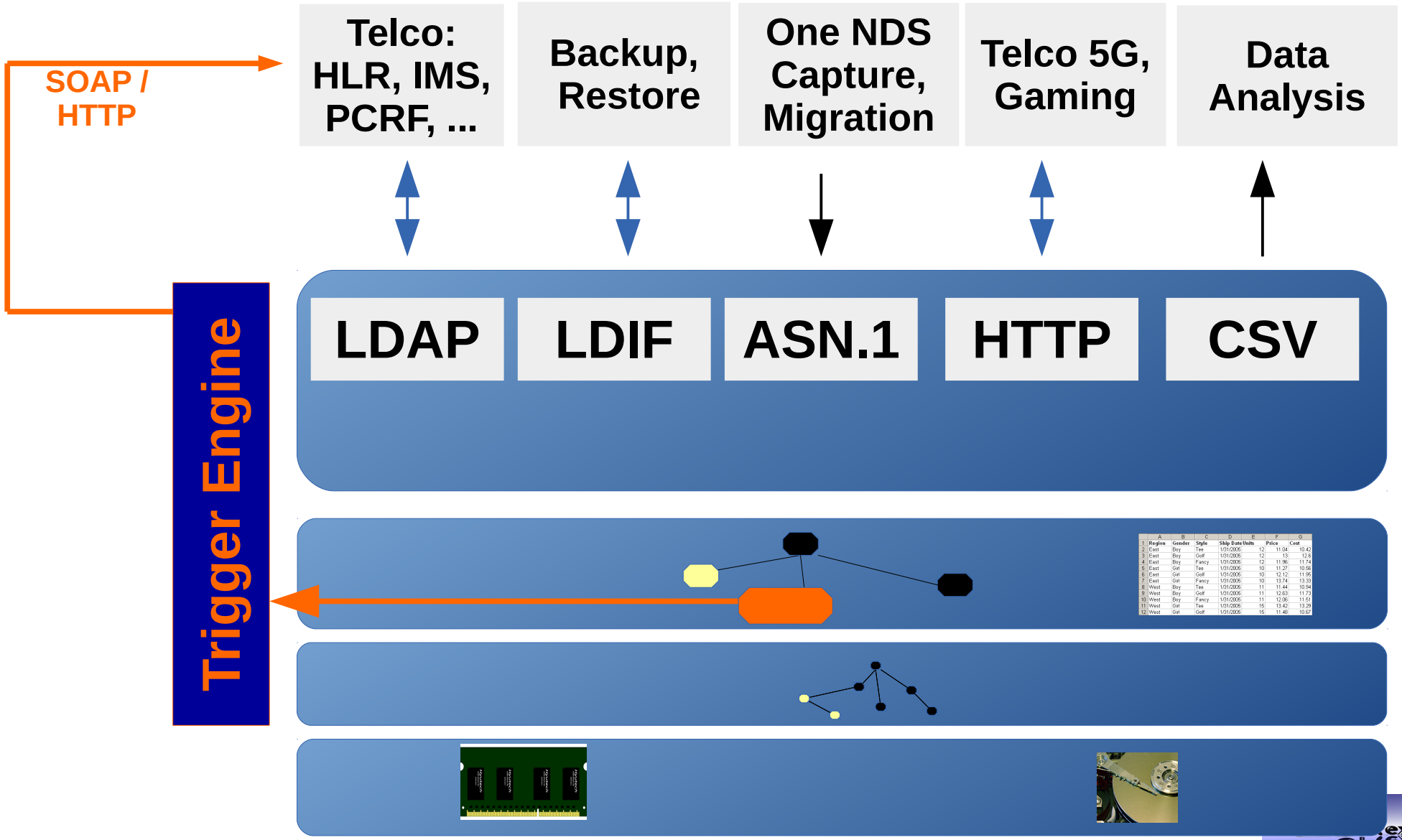


You may assign network categories and QoS to specific clients

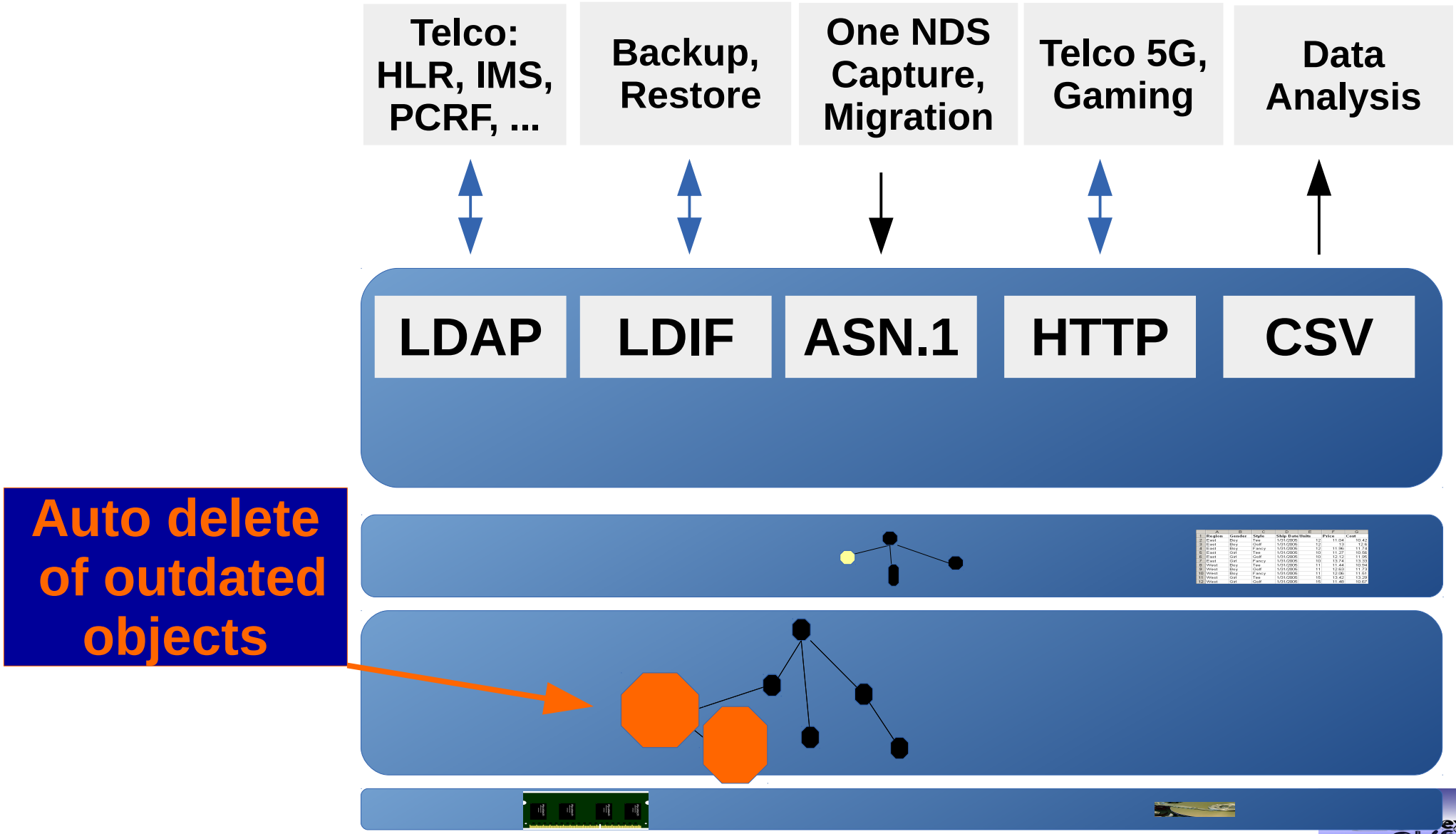


Interfaces with particular network categories can be assigned to specific physical connections

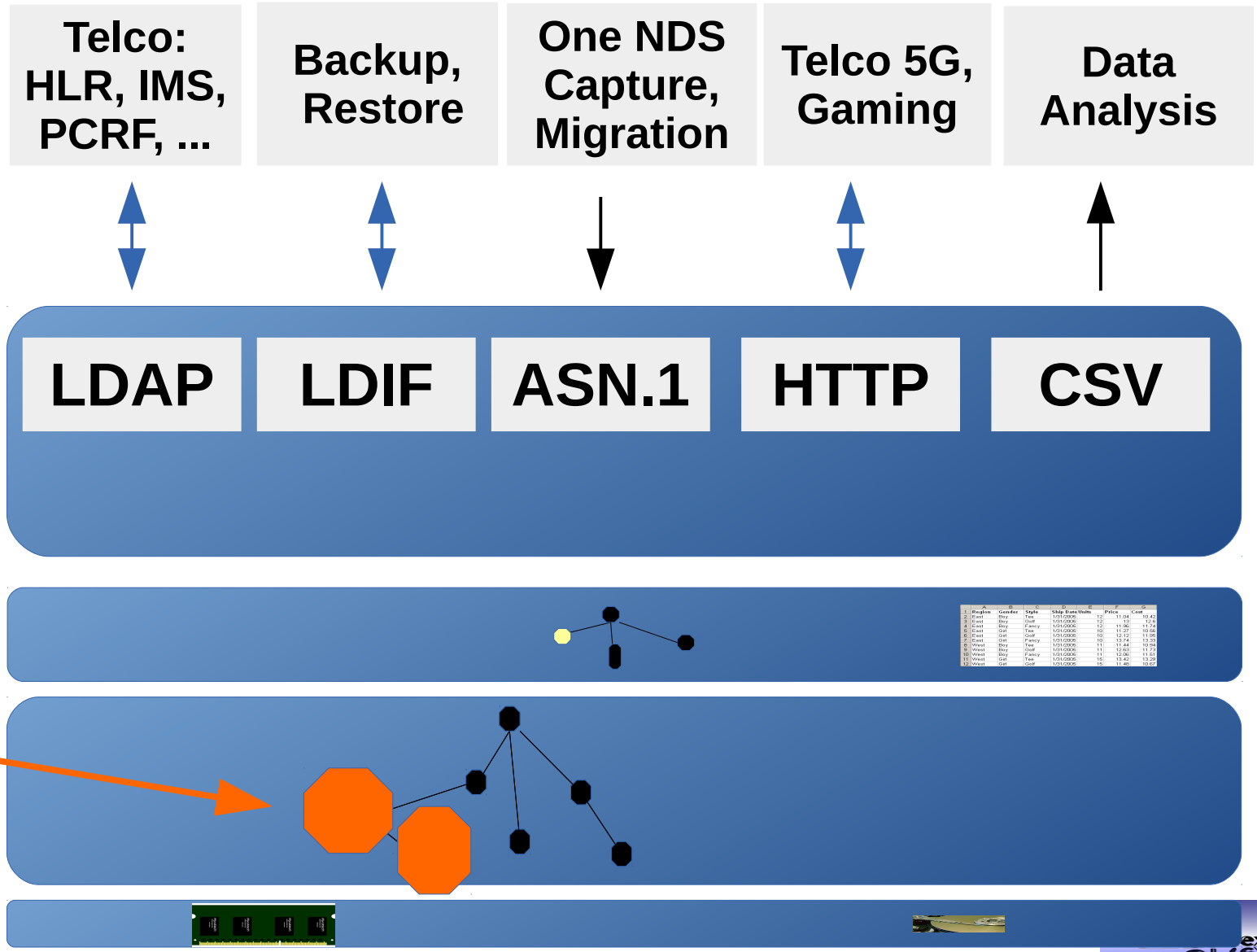
You can notify your applications about specific events



You can assign a time to live to objects

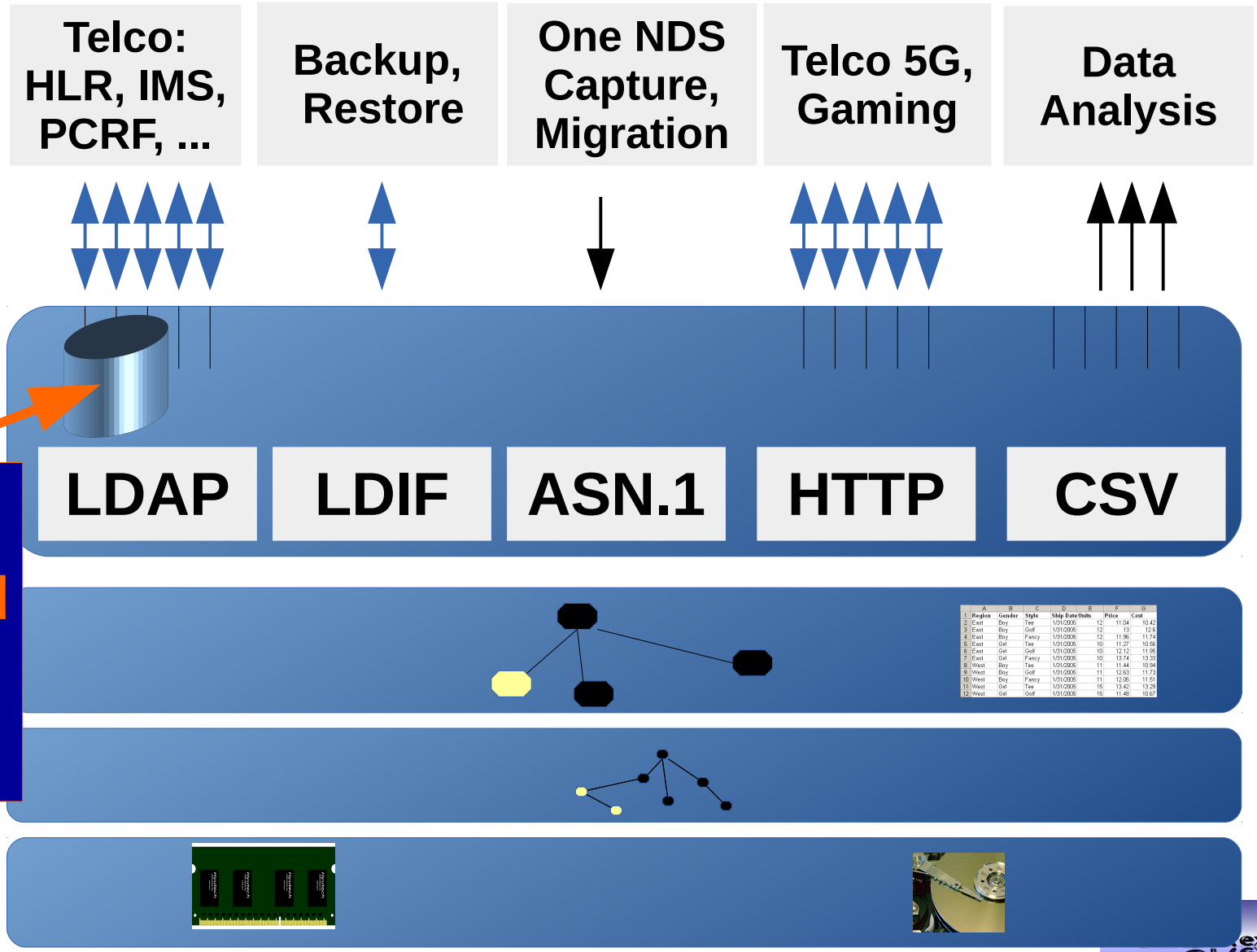


You can advise objects to keep a value level history



Objects keep historic values of attributes

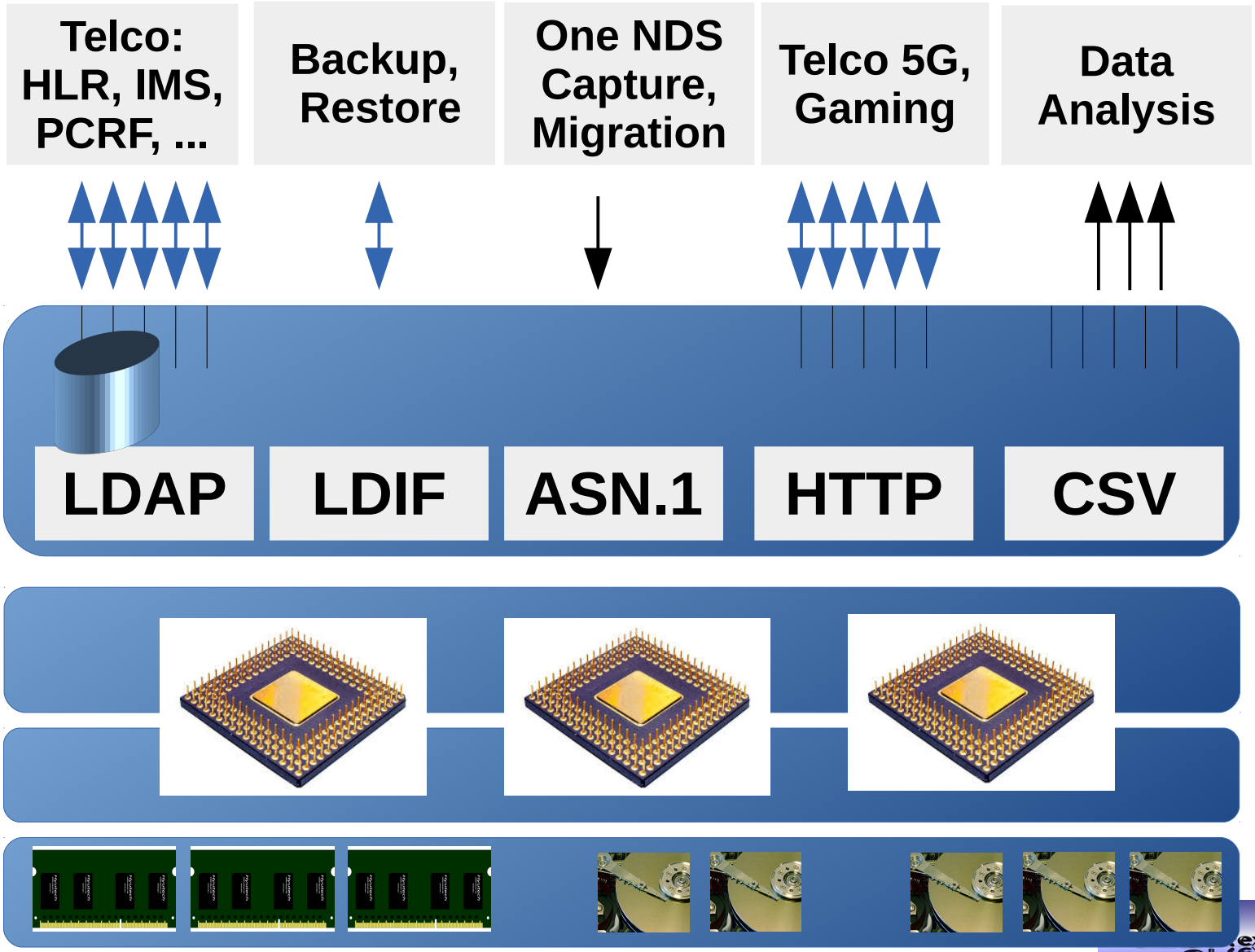
Your clients may use many parallel connections



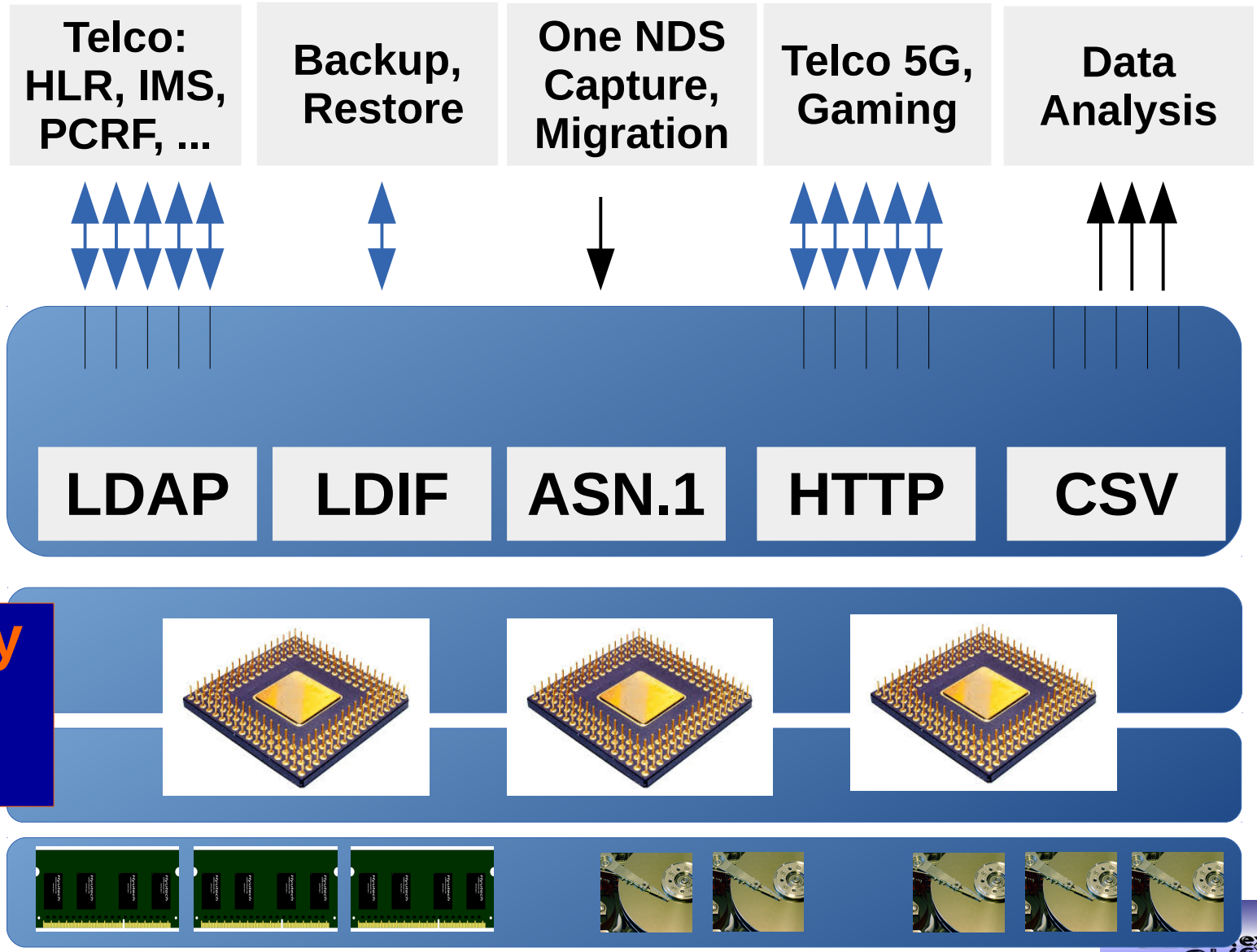
Connections can be queued into worker threads

Your throughput scales with parallel resources

Either worker or connection runs in parallel threads



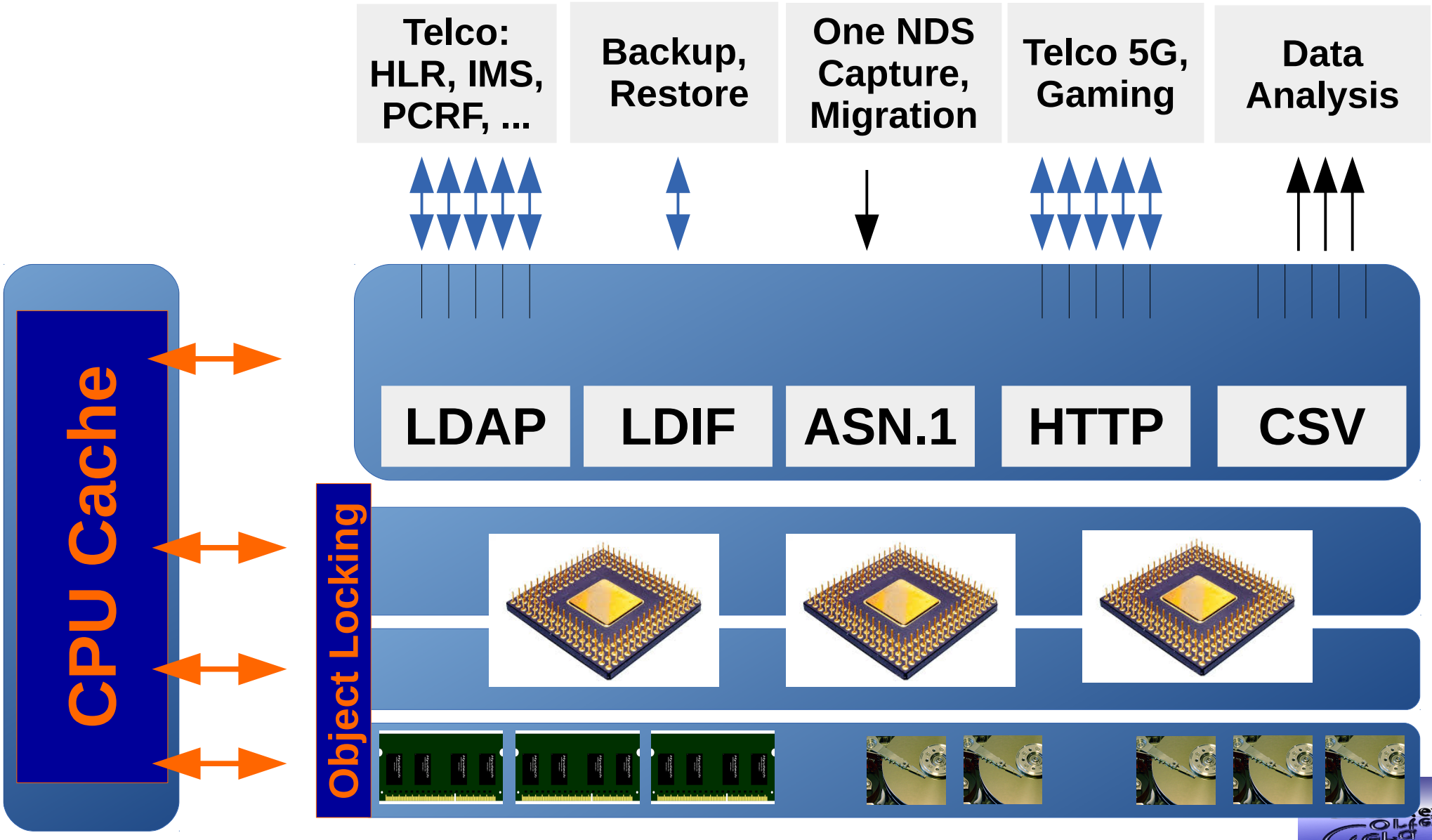
Concurrent access is consistent by object level locking



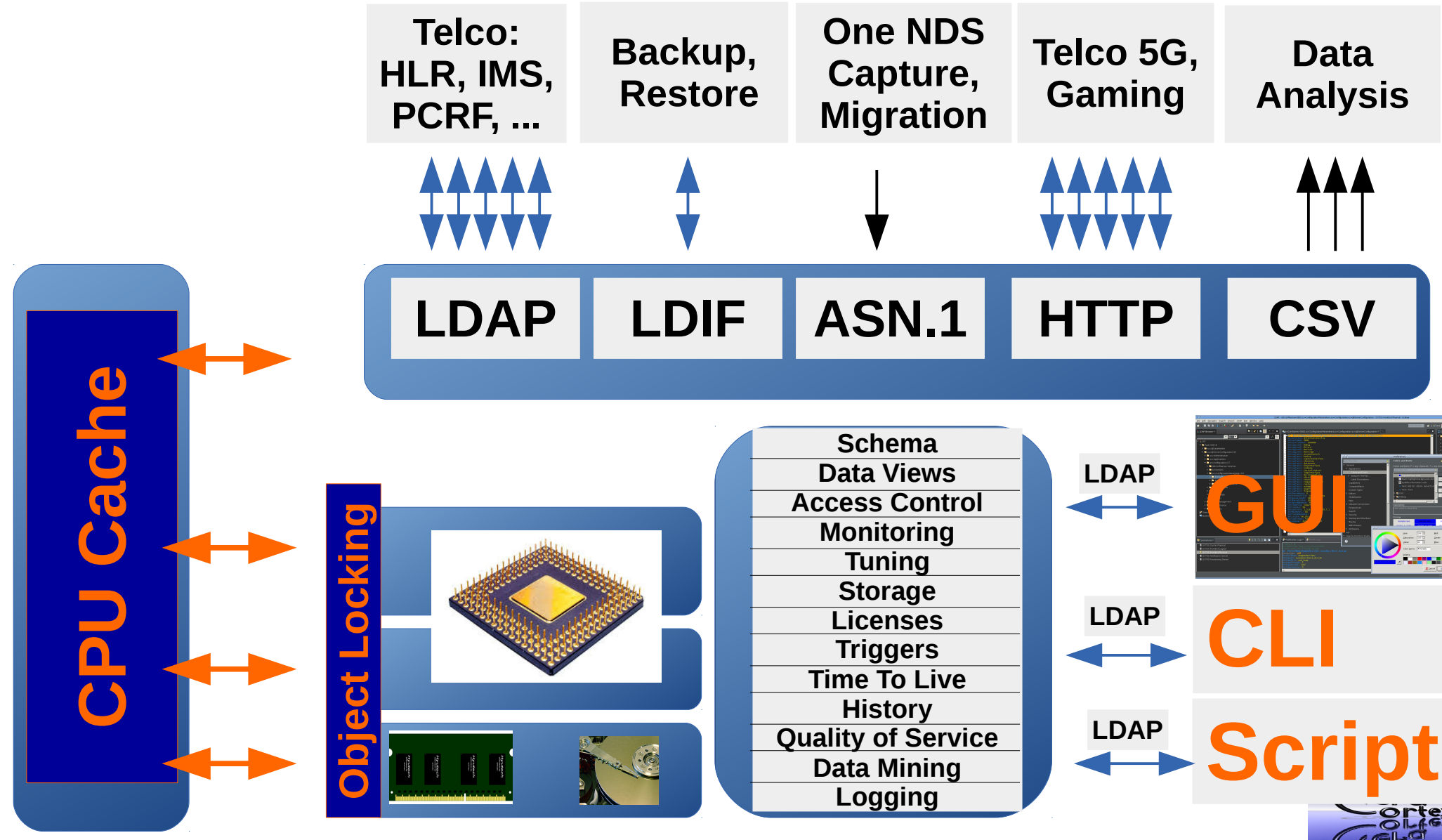
Consistency by object locking

You get micro second responses:

All layers communicate via CPU cache



You use online administration without service interruption



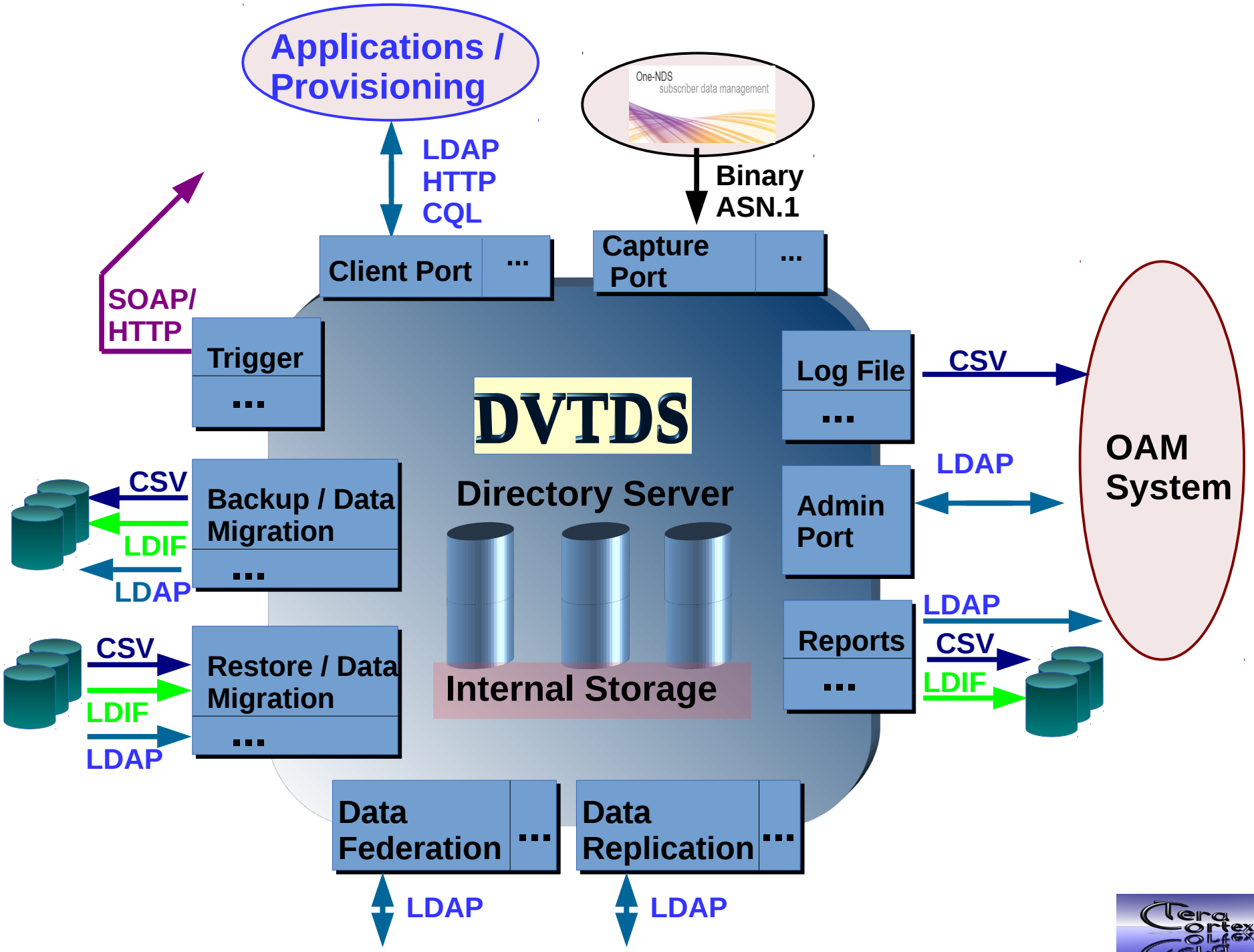
You get these benefits ...

- Cut 90% of hardware cost by extreme efficiency
- Data consolidation and consistency by data view based sharing
- Up to 100000 requests per second *per CPU core*
- Down to 20 µs response times
- Complete freedom to structure your data
- Flexible storage type based on category of data
- Transaction safety by ACID compliance
- Any data: Text, numbers, binary, images, video

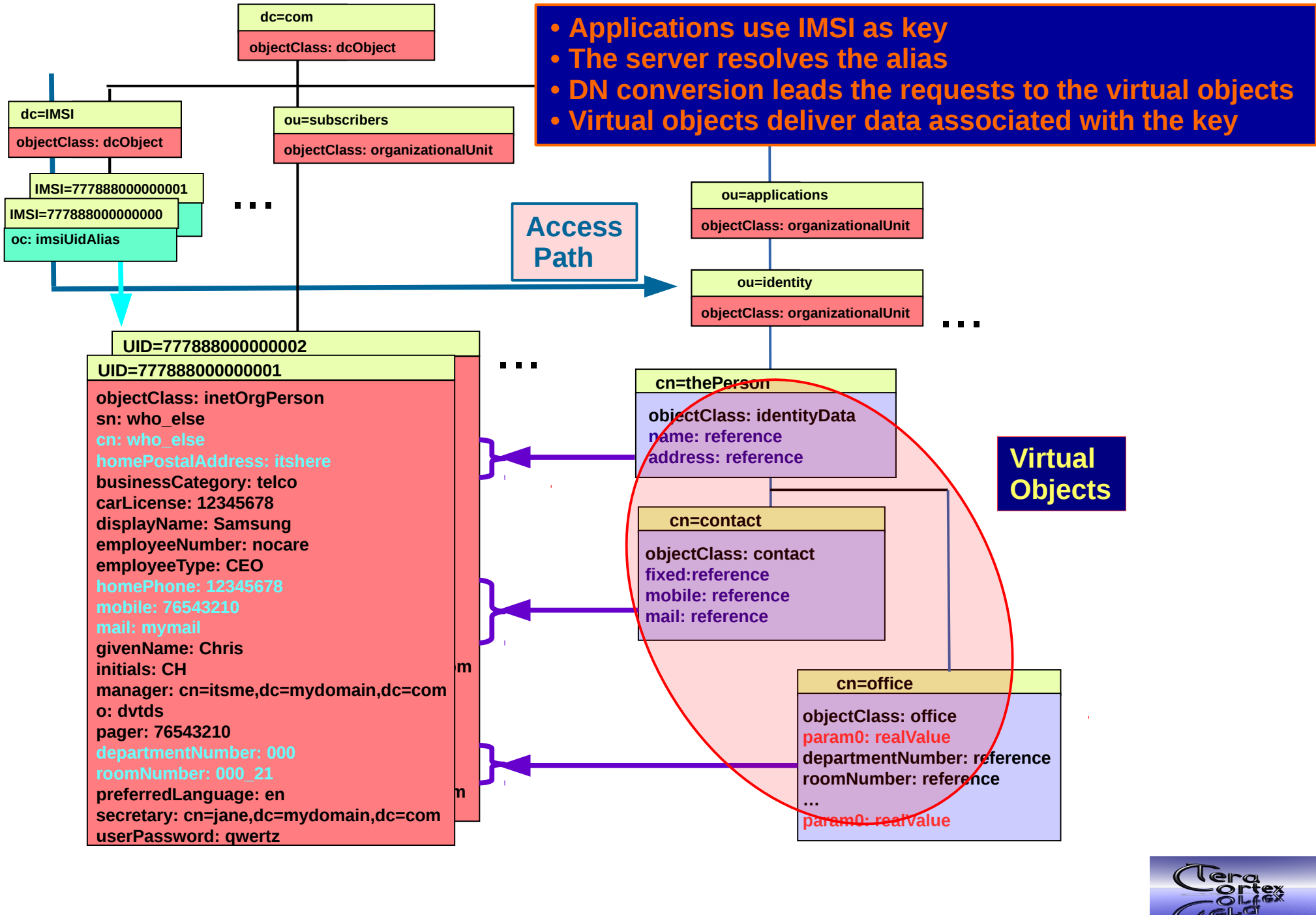
… and many more configurable features

- Automatic deletion of objects having time to live property
- Automatic tracking of historic values of attributes
- High speed parallel logging
- Sophisticated trigger mechanisms
- Quality of service and traffic budget for specific clients
- Physical separation of different categories of traffic
- Fine grain access control and security concept
- Feature activation by license management system
- Resource monitoring

Summary of interfaces

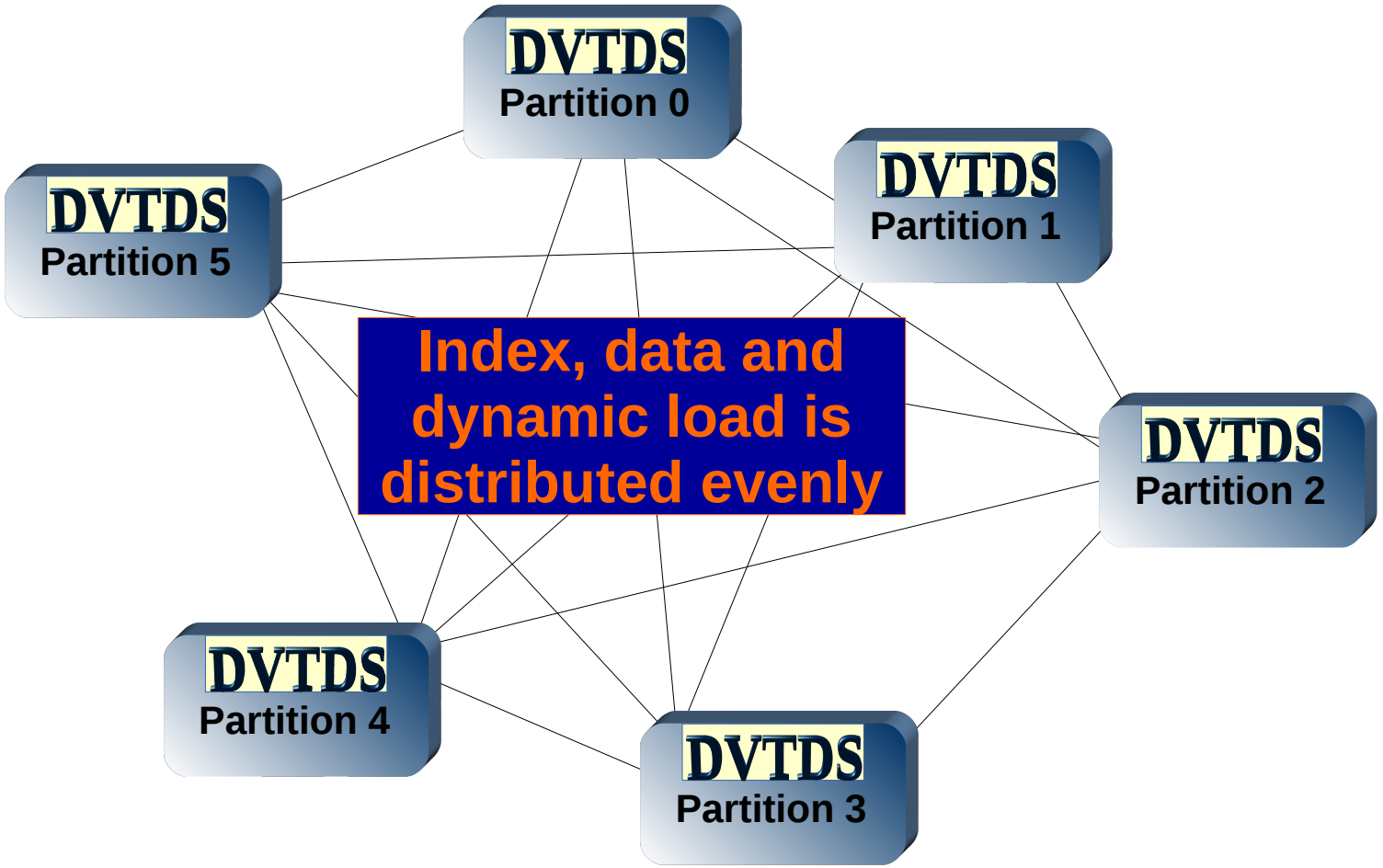


Data view example



You need more throughput?

Scale by adding nodes

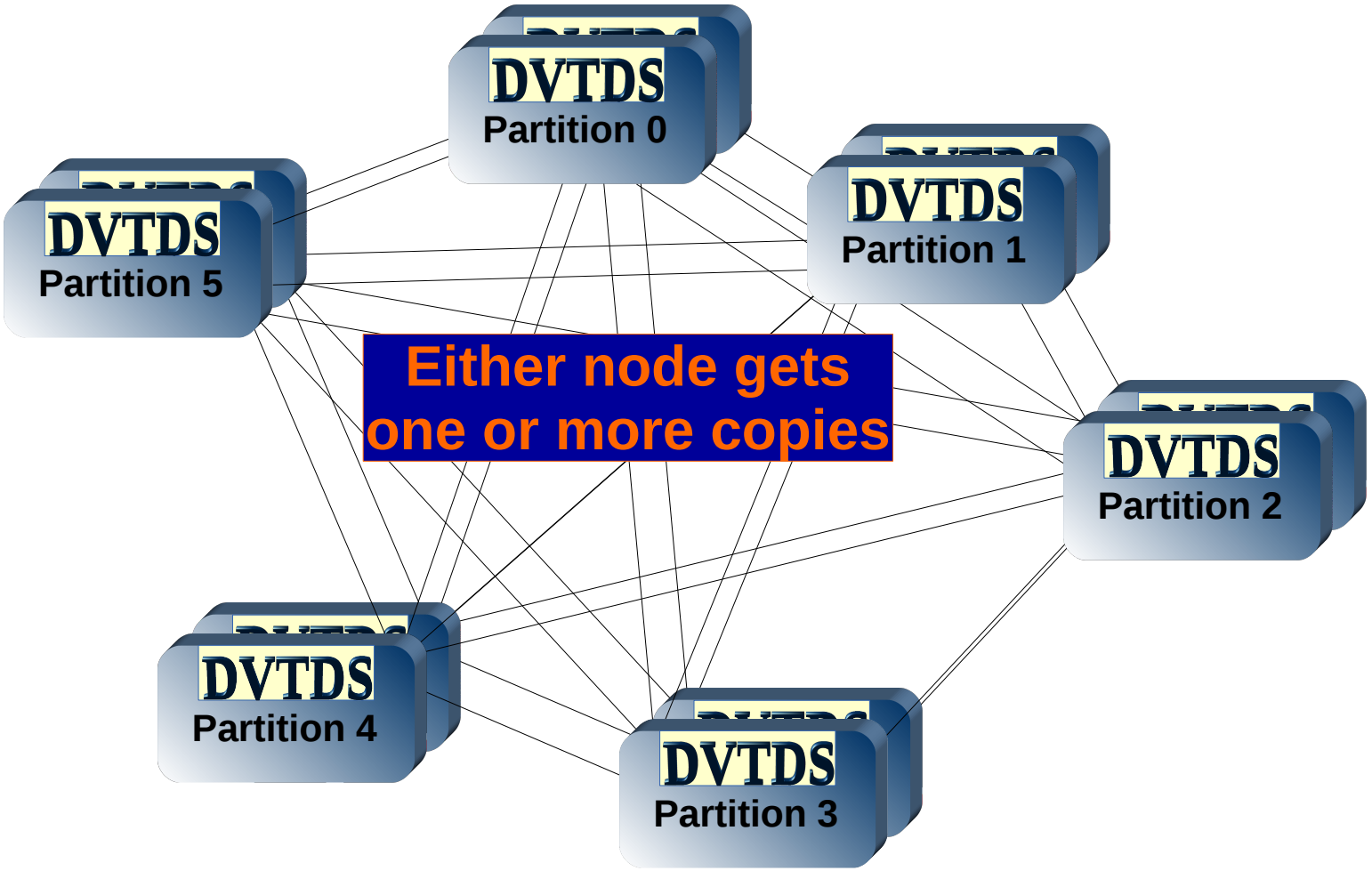


Your throughput scales with the number of nodes

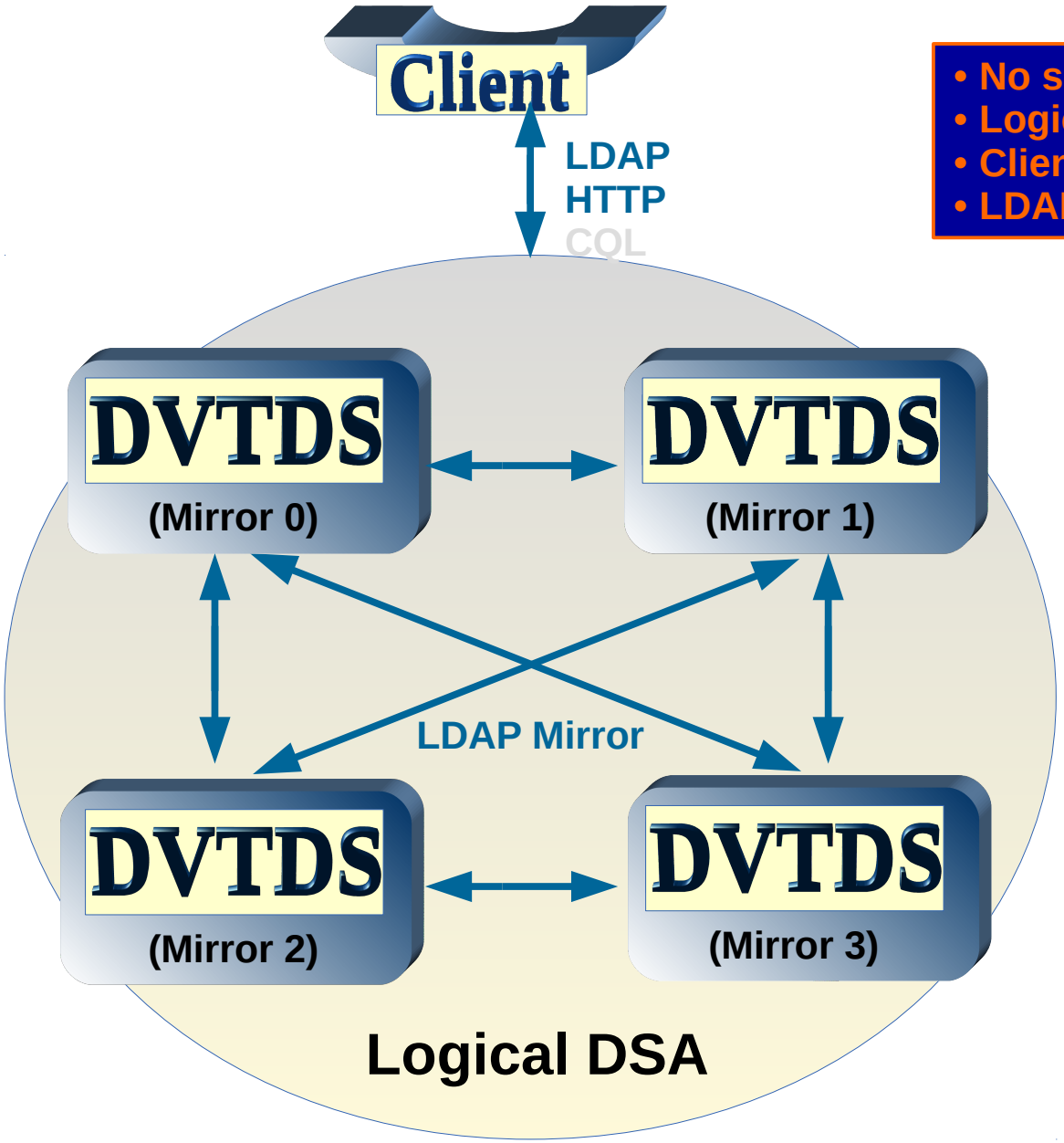
- Linear scaling proven
- Distributed ACID transactions supported
- Benchmark in AWS cloud with 25 million requests / s
- Automatic request routing
- Clients are not aware of physical data distribution
- But can be informed about optimized routes
- Network category / QoS assignments kept across nodes
- Online administration without service interruption

You need high availability?

Add replication



Replication Details

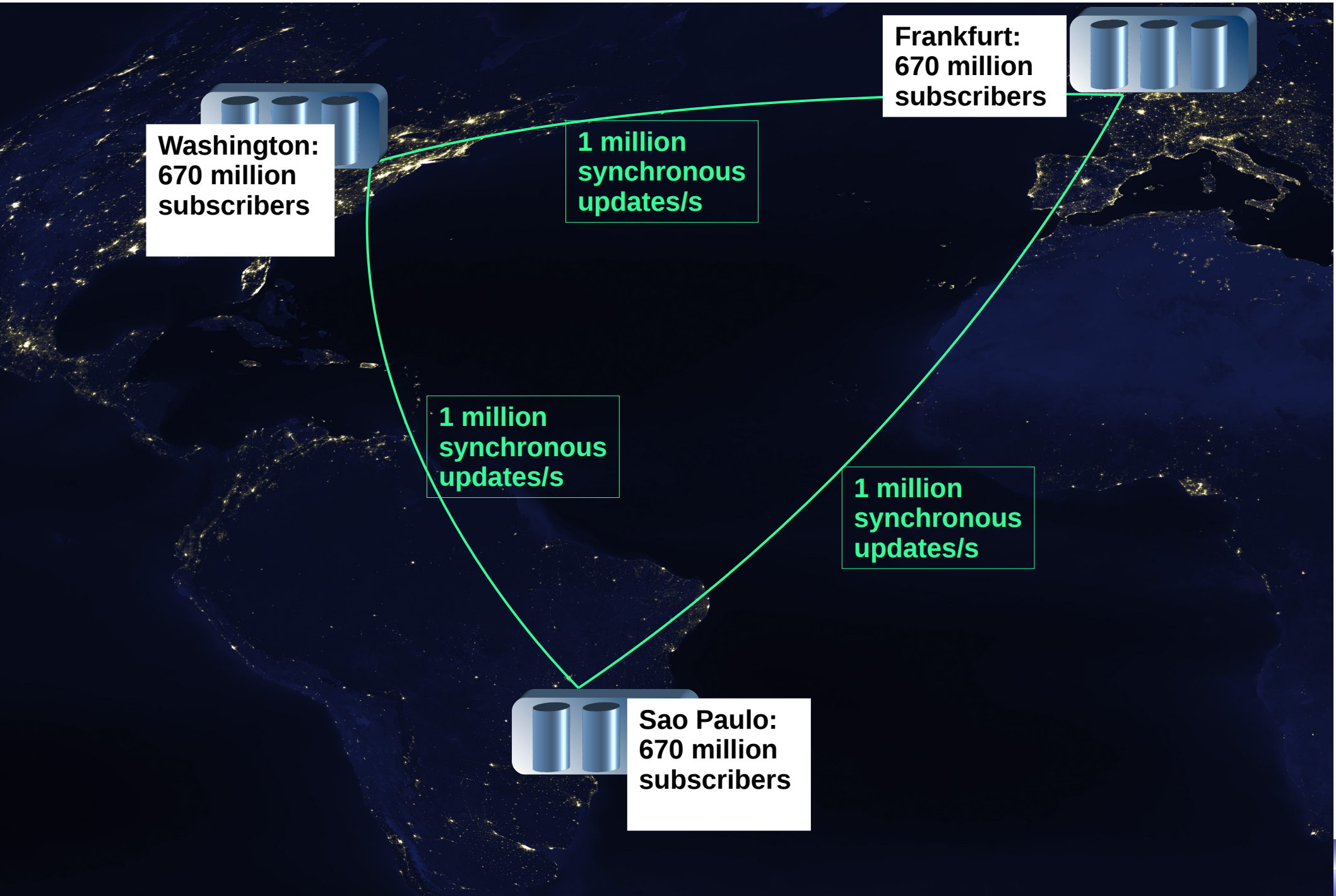


- No single point of failure
- Logical DSA concept
- Clients see the same model in every node
- LDAP protocol used for replication

You can use up to 16 copies

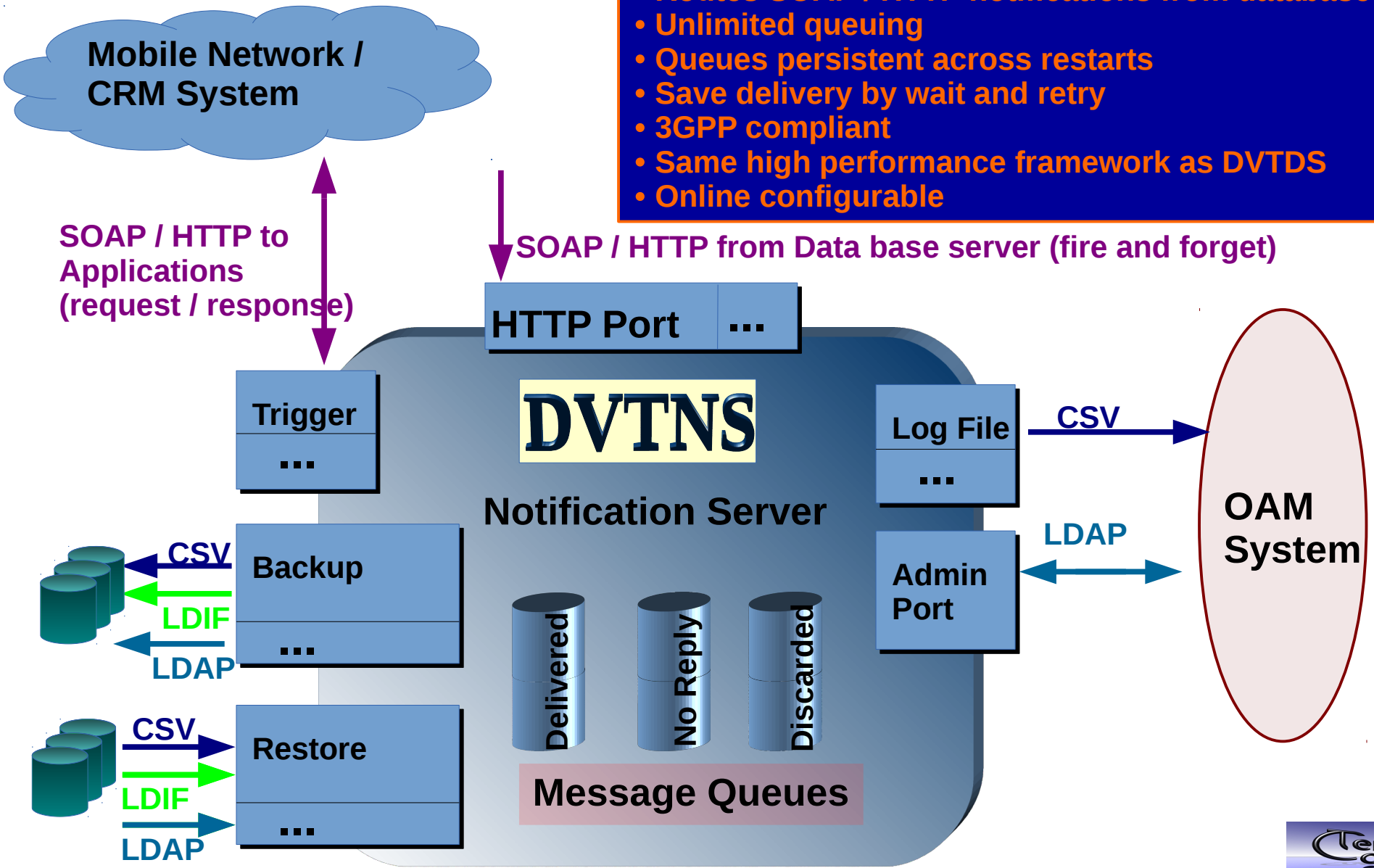
- Symmetric multi master replication
- Real time resolution of distributed conflicts
- Distributed ACID transactions supported
- Automatic replication of administrative actions (schema, ...)
- Consistency types (sync / async) configurable per object class
- Clients may tune consistency type dynamically
- Copy target nodes configurable per object class
- More than 1000000 synchronous updates/s intercontinental

You can count on our benchmarks

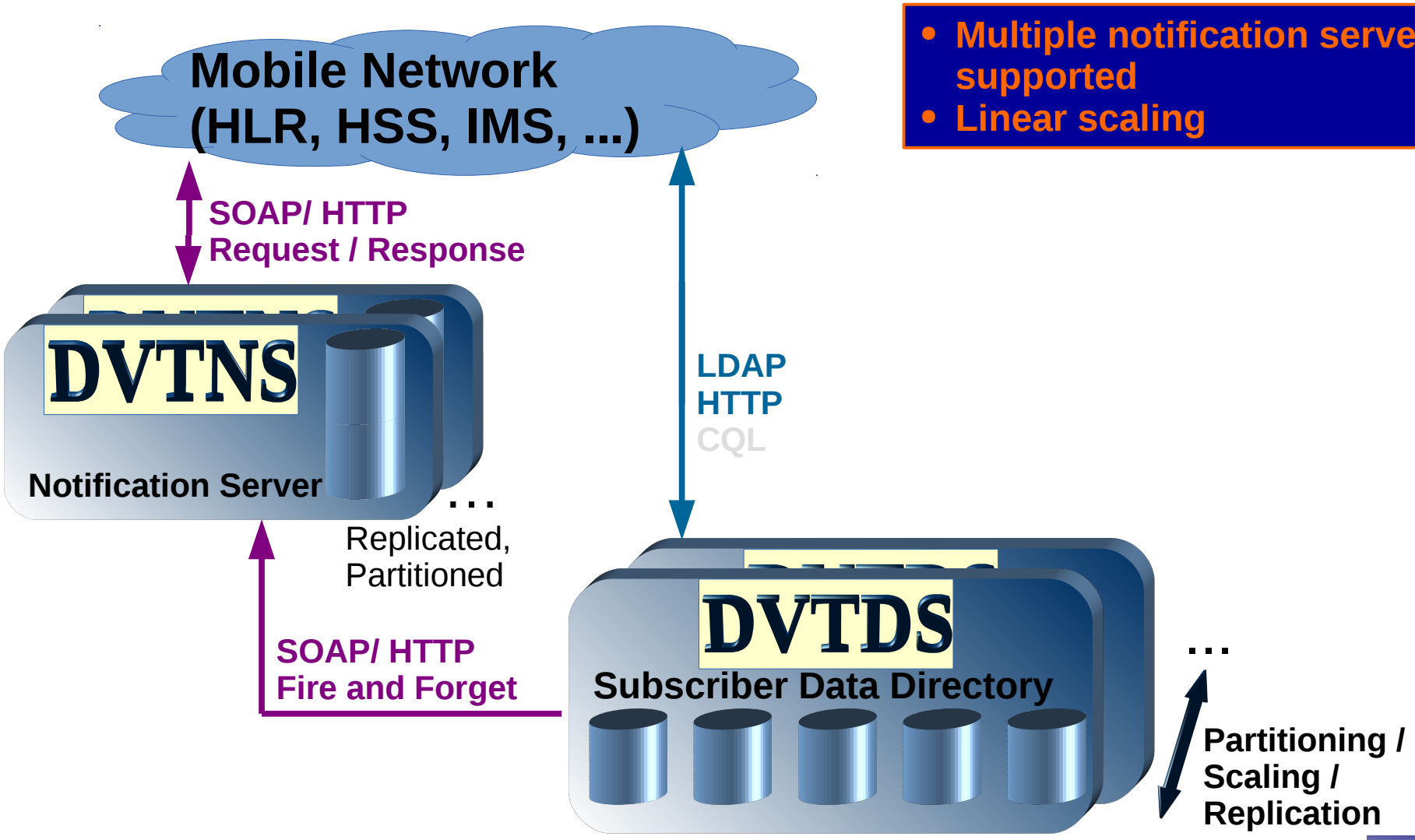


About notification services

- Routes SOAP / HTTP notifications from database
- Unlimited queuing
- Queues persistent across restarts
- Save delivery by wait and retry
- 3GPP compliant
- Same high performance framework as DVTDS
- Online configurable

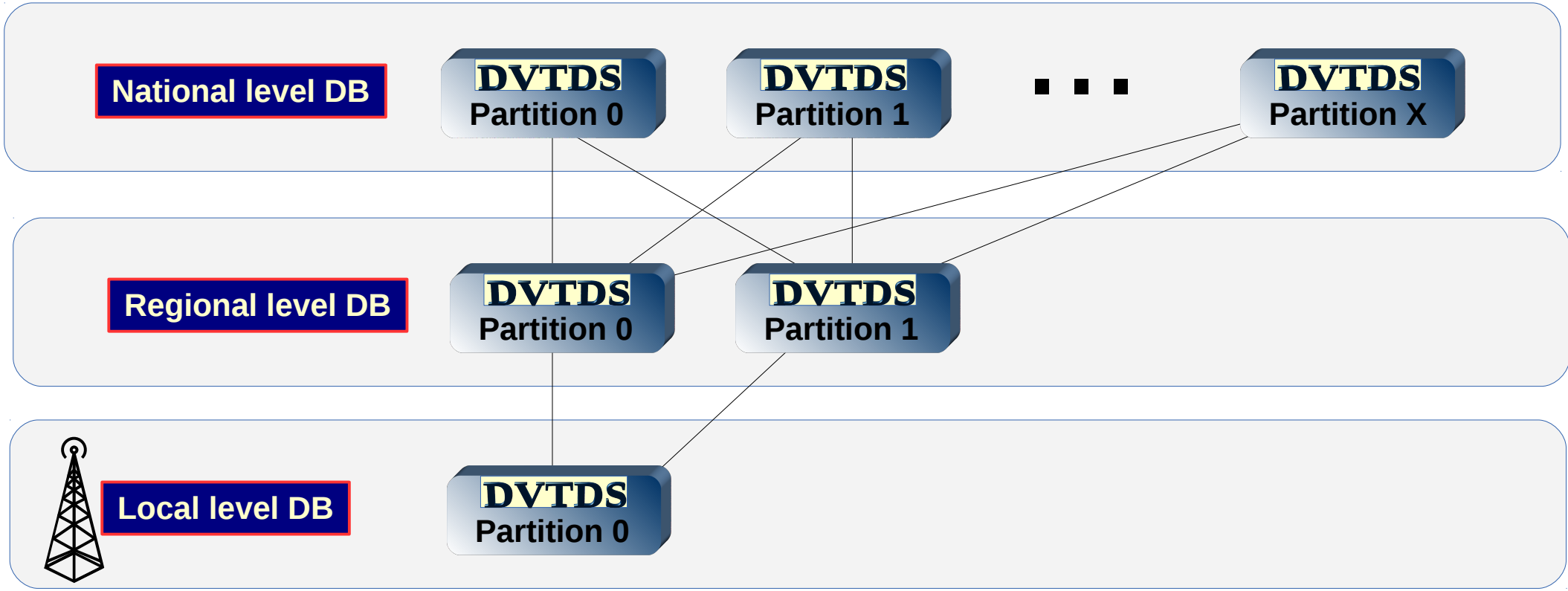


You can also scale the notification environment



- Multiple notification servers supported
- Linear scaling

Mobile Edge: Micro second response times in large networks



Either level only holds the data needed at that level

Your users get superior response times

- Three level deployment in large countries, two *may* suffice
- National DB holds all data
- Local DB fetches data from regional DB when mobiles check in
- Regional DB fetches data from national DB on request
- Applications also deployed at either level
- Search and update DB at their level
- Updates are replicated asynchronously to higher level
- Local and regional DB delete data based on idle timeout (TTL)

Mobile edge properties

- First data fetch as fast as in standard centralized deployment
- All subsequent interactions done locally → no WAN latency
- Standard applications need to be co - located
- When user leaves the cell the idle timeout counters start
- Next cell executes the same process when user checks in
- Idle data deleted from local node when time to live runs out
- Databases may be partitioned and replicated at either level
- DVTDS fully supports mobile edge mechanisms

X.500 style deployment supported for backward compatitibilty