

Object Storage

Pour les Données non Structurées et Archives

Résumé

1

Pourquoi un système de type Object Storage

4

Notre choix d'implémentation

2

Pour quelles données

5

Scality Ring

3

Quels changements pour les utilisateurs

6

Connecteur S3 & Management

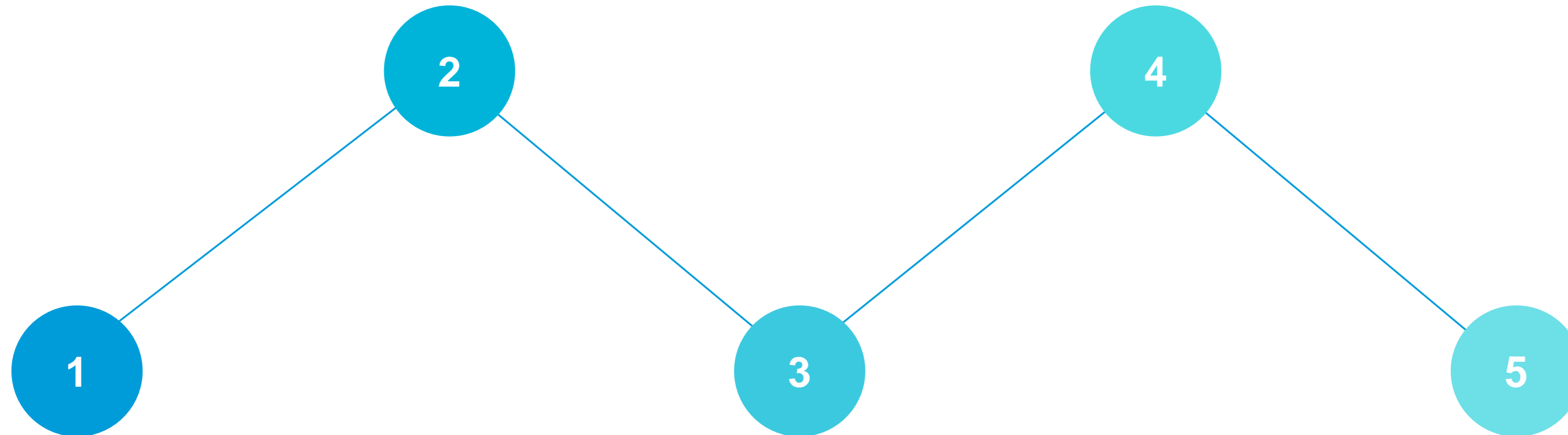
Pourquoi un système d'Object Storage ?

Protection de données

Permet une meilleure protection des données à l'échelle de plusieurs petabytes grâce à la technologie «Erasure coding»

Méta données

Permet d'étendre la richesse de l'information au sujet des objets stockés



Explosion des données

Résout les problèmes liés à l'explosion de la quantité de données non structurées .

(Scale-out & limitless)

Pas d'hierarchie

Permet de se détacher des systèmes de fichiers, leurs hiérarchisations et limites.

Contrôle des coûts

Permet un meilleur contrôle des coûts par rapport au Cloud Public qui peut être très cher en fonction de son utilisation

Object Storage: pour quelle types de données ?



Le stockage des données froides

- En général 85% des données sont dites de type froides dans les stockage traditionnels (NAS & Block)



Consolidation des Données Institutionnelles

- Données de recherche
- Documentations scientifiques
- Bibliothèque et références
- Projet « Master Data Plan » de l'EPFL



Sauvegardes et dépôt

- Remplacement des lecteurs de bandes
- Dépôts pour codes, logiciels, documentations, etc...

Object Storage: changement pour l'utilisateur

Protocole

- Protocole S3 (Simple Storage Services) développé par Amazon
- Devenu le protocole par défaut pour ces systèmes

Stockage des objets

- Dans des conteneurs, «Buckets»,
- Les objets ont tous un identifiant unique, «ObjectID»
- Pas d'hierarchie (Flat structure)

Droits et permissions

- Support pour IAM d'Amazon (Identity and Access Management)
- Active Directory

Nouveaux outils

Pour l'accès aux données en S3 (montage et synchronisation)

- **GUI:** S3 Browser, CloudBerry, Bucket Explorer, S3Fox, etc..
- **CLI:** S3FS, S3 Tools, FUSE, Duplicity, Rclone, etc..

Notre choix d'implémentation à l'EPFL

1. De type Software-defined & Multi-tenant
2. Indépendant du matériel
3. Une capacité de 4.5 petabytes utilisable dès le premier jour
4. Méthode de licence perpétuelle
5. Un plan de croissance de 1 à 2 petabytes par année
6. Nœuds de stockage de type x86
 1. Haute densité (minimum 720TB raw par nœud)
 2. Aucun point unique de défaillance
 3. Réseau d'interconnexion et flux des données à 40Gb/sec
7. Répartition de l'infrastructure sur 3 sites permettant la perte d'un site complet et plus un noeud sans perte de service (*aka* Stretched RING)

Scality Ring: fonctionnalités

1

Software-defined

2

Architecture distribuée (peer-to-peer share nothing)

3

Protocoles

- **Nous implémentons que le protocole S3**

4

Scale-out

5

Protection des données distribuée (réplication & erasure coding 7+5)

- Auto-guérison
- Protection des données basée sur des polices

6

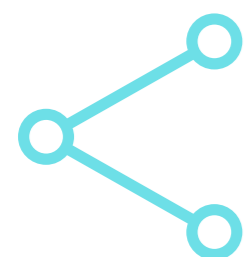
Distribution géographique natif

- Stretched RING
- 3 Sites
- Storage Overhead 70%

Scality Ring: architecture en déploiement étiré

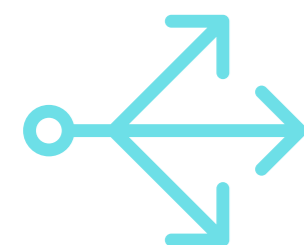


4.5 PB initial



Distribué sur trois sites

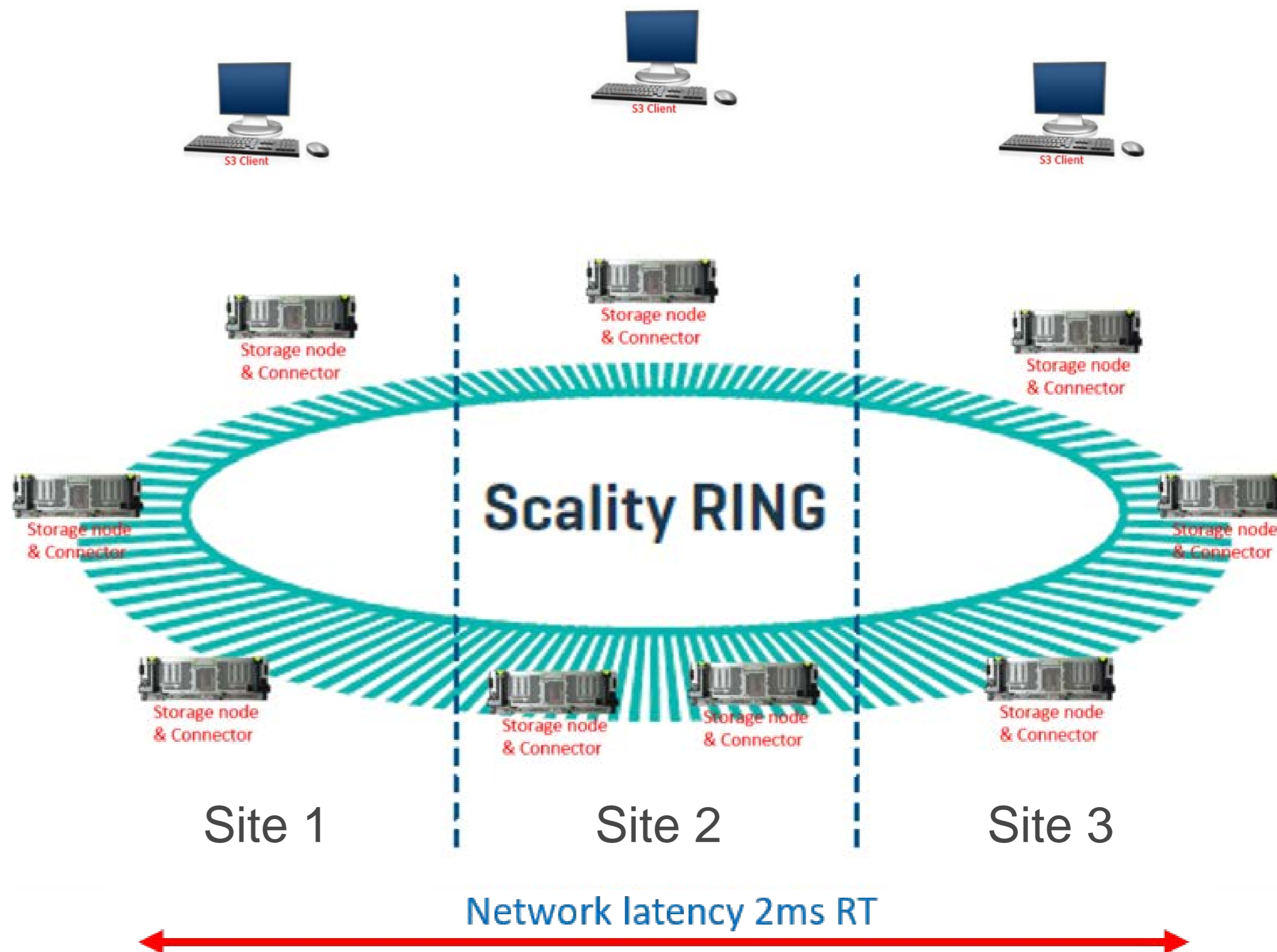
- Stretched Erasure coding 7+5 (sync)
- 70% Overhead
- Triple mirror for small objects
- Durability loss of a site + 1 storage node



Extension de 1 a 2 PB par année

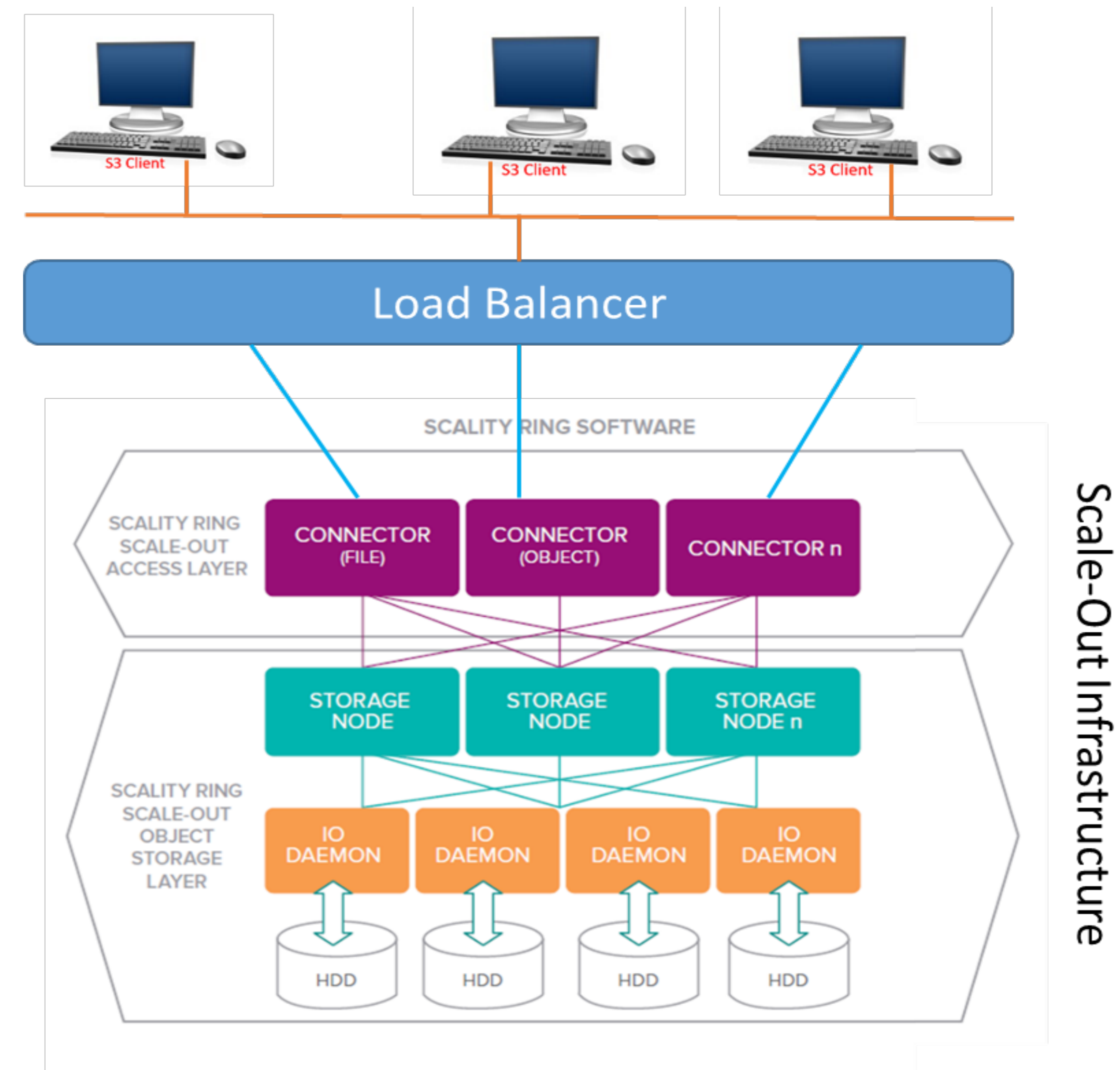


Interface applicatif S3

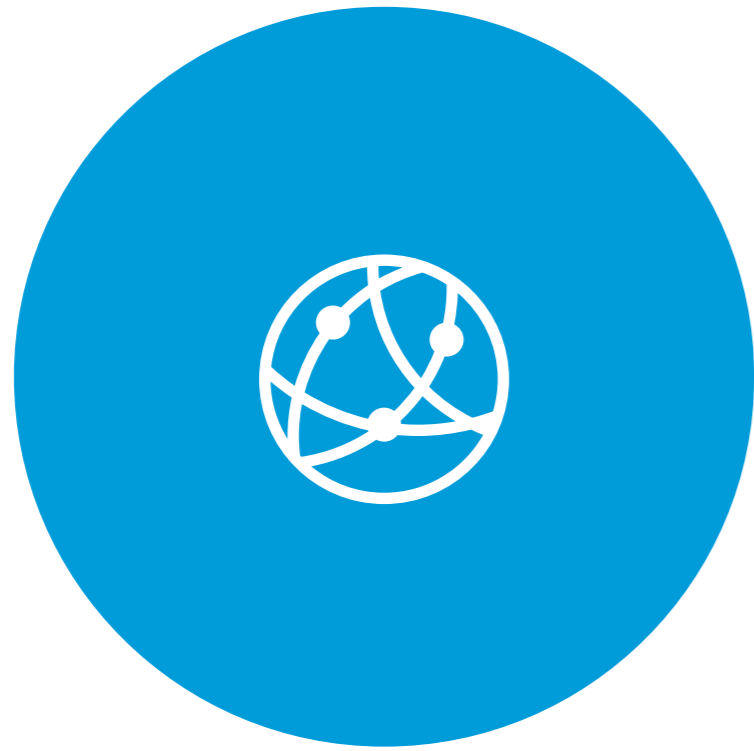


Scality Ring: le flux des données par maillage

- Support de millions de requêtes simultanées
- Les applications clientes peuvent contacter n'importe quel noeud connecteur
- Les connecteurs sont « *stateless* » et peuvent contacter n'importe quel noeud de stockage
- Il n'y a pas de base de données centralisée pour les métadonnées ou les indexes
- Les temps de réponse restent similaires avec de l'extension de l'infrastructure



Rôles de gestion



VPSI

Responsables de l'infrastructure



Facultés

Responsable de l'usage qui est fait de l'infrastructure



Services informatiques de proximité

Responsable du management des données de l'unité