Software Heritage

Contribution à un projet libre dont le but est l'archivage à long terme de notre patrimoire logiciel

Quentin Campos

8 septembre 2016

Introduction

Quentin Campos - M2 Génie Logiciel

Stage de fin d'études réalisé d'Avril à Septembre au sein de **l'Inria**, dans le centre de recherche de Paris.

Institut public de recherche fondé en 1967 dans le cadre du plan calcul.

- Incube de nombreux **projets libres** (OCaml, IRILL).

Software Heritage

- **Archiver** le code source.
- **Protéger** son contenu.
- **Partager** à la demande.

- Projet libre.

L'initiative à été lancée par Roberto Di Cosmo
et Stefano Zacchiroli au sein de l'Inria.

Architecture

Archiver

- **Parcourir** des sources à la recherche de contenu.

Archiver

- **Parcourir** des sources à la recherche de contenu.
- **Télécharger** les contenus dans Software Heritage.

Archiver

- **Parcourir** des sources à la recherche de contenu.
- **Télécharger** les contenus dans Software Heritage.
- **Vérifier** ponctuellement les sources pour mettre à jour Software Heritage.

Protéger

Les fichiers sources sont enregistrés dans un **blob storage** clef-valeur.

sha1(contenu) -> fichier source.

Protéger

Software Heritage conserve **l'historique** des projets logiciels.

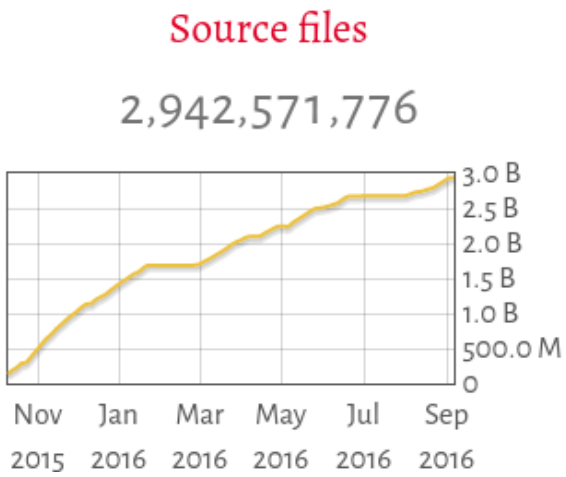
- Les **dépôts** sont accompagnés de leurs **commits**.
- Les **révisions** et **tarballs** sont enregistrées comme des **répertoires**.

Protéger

Ces données sur la **structure** sont stockées dans une **base de données** Postgres.

Préserver

150 TB de fichiers sources

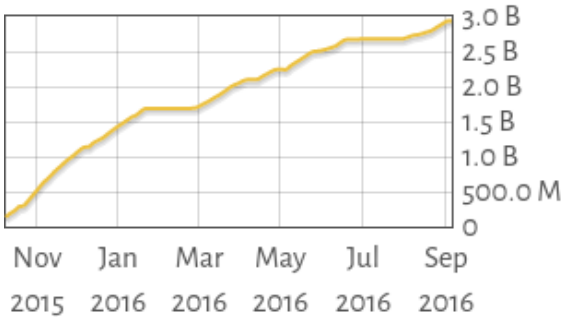


Préserver

5 TB de DB Postgres

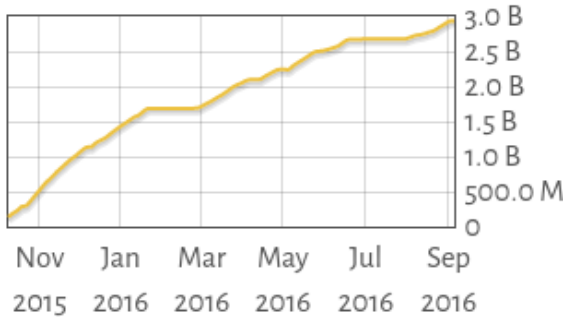
Commits

644,493,312



Projects

25,265,692



Partager

- Le **site web** permet de vérifier si un contenu est **déjà archivé** dans Software Heritage.

Partager

- Le **site web** permet de vérifier si un contenu est **déjà archivé** dans Software Heritage.
- Une **API** publique permet de demander un **dépôt**, une révision ou un répertoire.

Mes contributions

Archiver

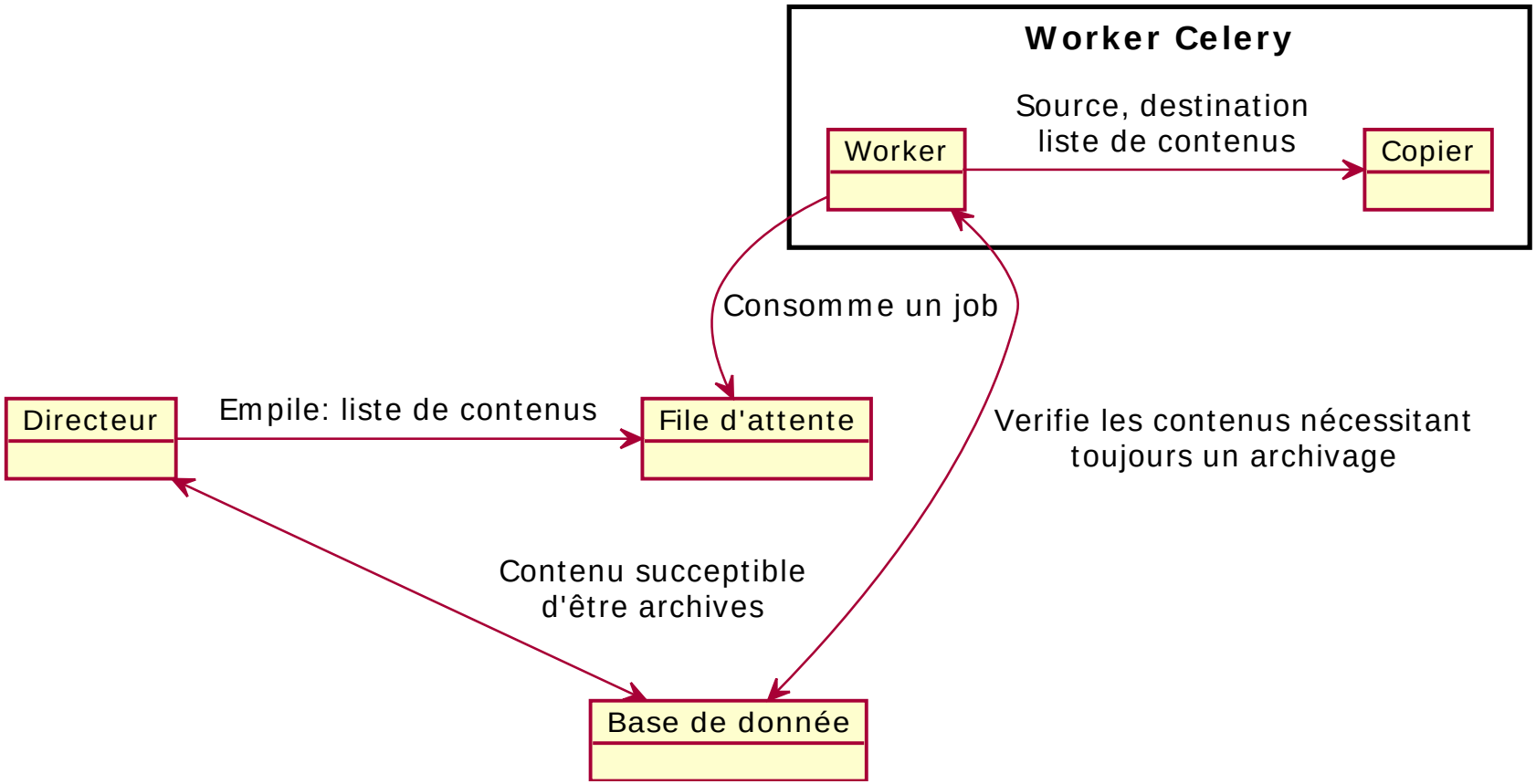
Objectif :

- avoir **plusieurs copies** de chaque fichier source.

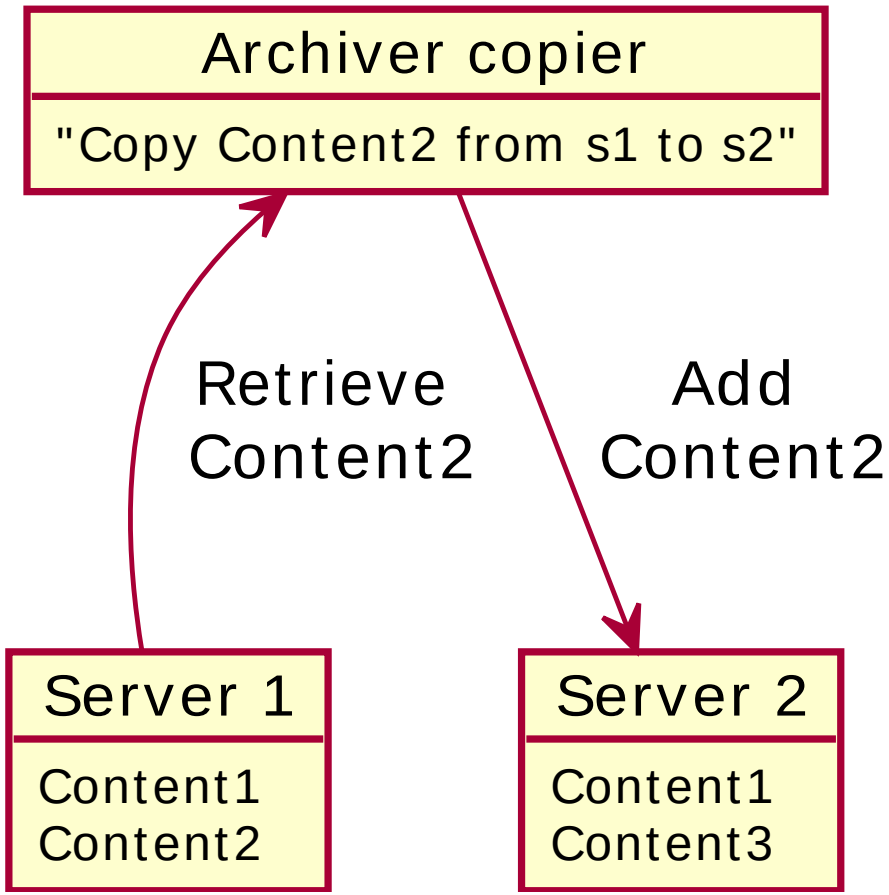
Difficulté :

- tenir **à jour** la copie de manière asynchrone.

Archiver


Worker Celery
Source, destination
liste de contenus
Worker
Copier
Consomme un job
Directeur
Empile: liste de contenus
File d'attente
Verifie les contenus nécessitant
toujours un archivage
Contenu succeptible
d'être archives
Base de donnée

Archiver


Archiver copier
"Copy Content2 from s1 to s2"
Retrieve
Content2
Add
Content2
Server 1
Content1
Content2
Server 2
Content1
Content3

Content Integrity Checker

Vérifier **l'intégrité** des fichiers sources dans le storage.

Séléctionne **au hasard** des objets à tester.

Content Integrity Checker

Vérifier **l'intégrité** des fichiers sources dans le storage.

Séléctionne **au hasard** des objets à tester.

Répare immédiatement depuis une autre copie

Planifie l'archiver pour écraser le fichier corrompu

Self-healing

Checker

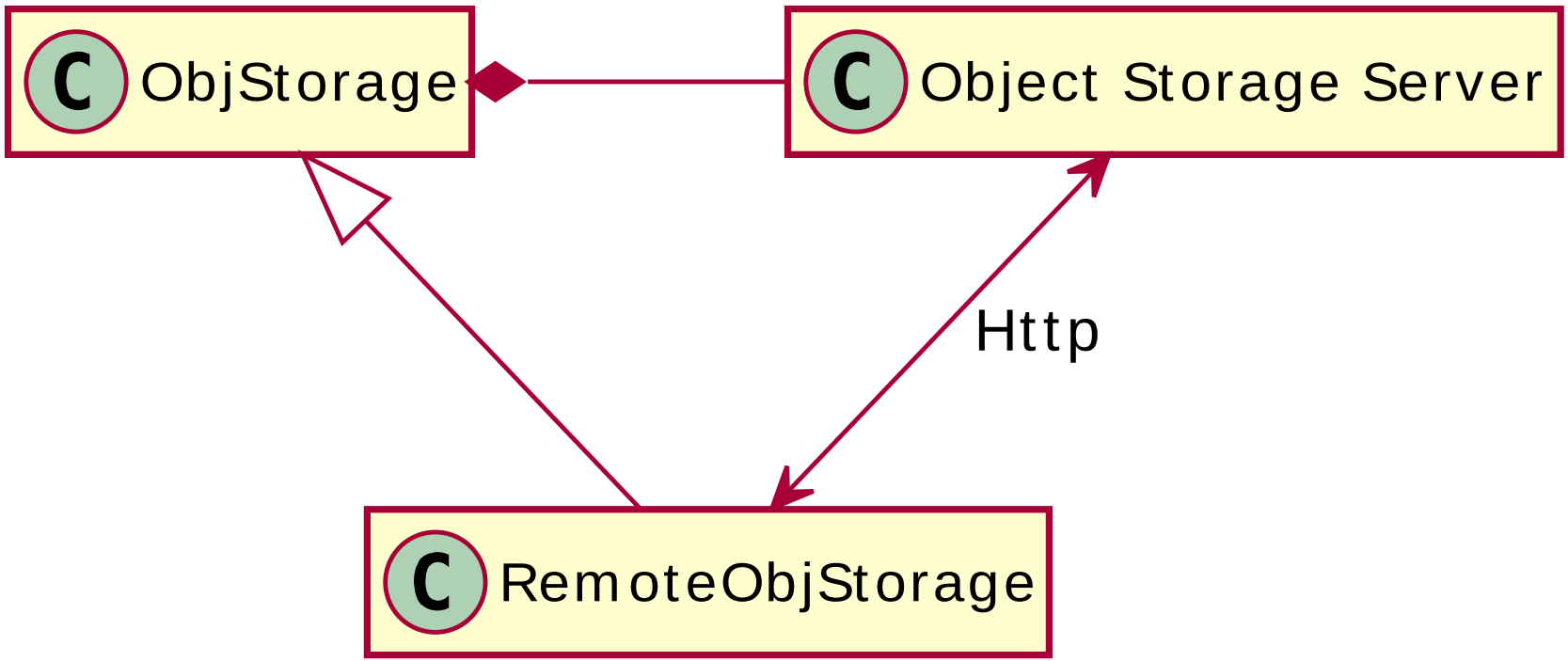
- Vérifie l'integrité en continu
- Invalide ou répare les fichiers corrompus

Archiver

- Vérifie l'intégrité à la copie
- Ecrase l'erreur à un archivage ultérieur

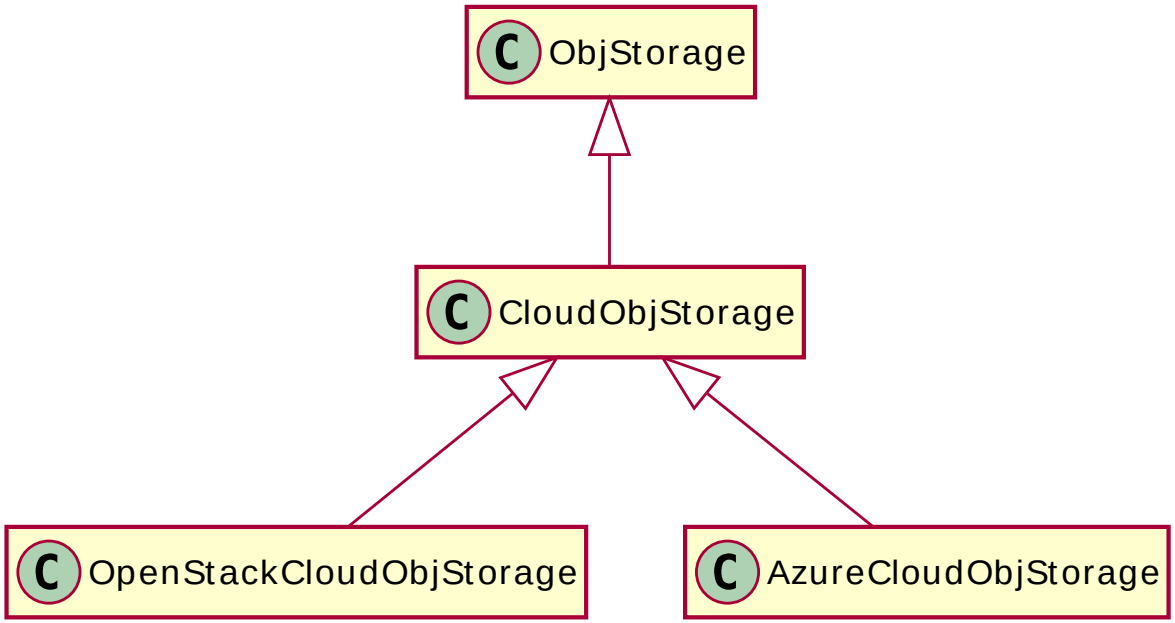
ObjStorage API

API REST pour accéder à un object storage à distance avec la **même interface**.



ObjStorageCloud

Implémentation de ObjStorage qui se connecte à un **cloud** via la librairie **Apache Libcloud**.



Software Heritage Vault

Fournit au téléchargement les objets Software Heritage sous forme d'un *bundle* : leur format original (Dépôt, tarball, . . .).

Software Heritage Vault

Fournit au téléchargement les objets Software Heritage sous forme d'un *bundle* : leur format original (Dépôt, tarball, . . .).

1. Une requête est effectuée en amont

Software Heritage Vault

Fournit au téléchargement les objets Software Heritage sous forme d'un *bundle* : leur format original (Dépôt, tarball, . . .).

1. Une requête est effectuée en amont
2. Le bundle demandé est compilé asynchronement

Software Heritage Vault

Fournit au téléchargement les objets Software Heritage sous forme d'un *bundle* : leur format original (Dépôt, tarball, . . .).

1. Une requête est effectuée en amont
2. Le bundle demandé est compilé asynchronement
3. Le bundle est stocké dans un cache où il est disponible au téléchargement direct

Conclusions

Conclusion

Enrichissant sur le plan **technique** : Python, Gestion de projet.

Plongée dans le monde du **logiciel libre**.

Projet ambitieux dont l'objectif est de devenir une **organisation internationale indépendante**.

Conclusion : Post-stage

Devenir **contributeur** dans de Software Heritage.

Utiliser Software Heritage pour de la **recherche** en Génie Logiciel Empirique.

Collect
Preserve
Share