

Rapport d'activité POC2 et Nuxeo

Stefane Fermigier (Nuxeo)

4 novembre, 2011

Abstract

Ce document est le rapport d'activité autour du POC2 à T0+12. Il constitue également le rapport d'activité de Nuxeo pour la même période.

Contents

1	Objectifs de l'activité	2
2	Progrès réalisés vers ces objectifs	2
2.1	Rédaction de l'expression des besoins	2
2.2	Rédaction des spécifications techniques du PaaS orienté GED multitenant	2
2.3	“OSGi-fication” de Nuxeo EP / Projet Apricot	3
2.4	Nuxeo et le multi-tenancy	3
2.5	Expérimentation du portage de Nuxeo sur des PaaS existants	4
2.5.1	SlapOS	4
2.5.2	Intalio	4
2.5.3	CloudBees	4
2.5.4	CloudFoundry	4
2.5.5	Ensemble / Juju	5
2.6	Nuxeo Cloud Controller	5
2.7	Stockage non-relationnel	5
2.7.1	Stockage S3	5
2.7.2	Stockage Scalify	6
2.7.3	Stockage non relationnel multi-backends	6
2.8	Publication / dissémination	6
3	Travaux en cours	7

1 Objectifs de l'activité

L'objectifs de l'activité "POC2 - Nuxeo" est de réaliser un prototype de services (PaaS et SaaS) de gestion documentaire basés sur un runtime multitenant pour composants OSGi.

Il doit notamment implémenter dans le contexte du cloud des services de:

- stockage de fichiers binaires sous forme de BLOBs
- droits d'accès aux documents (ACLs)
- gestion des métadonnées (i.e. propriétés)
- transformation des documents (ie. rendu de documents en PDF, extraction de vignettes photos / vidéos, etc.)
- indexation et recherche plein texte, par métadonnées, combinées
- relations entre documents

Le standard CMIS¹ fournira un modèle cohérent, commun au principaux éditeurs de solutions de gestion documentaire du marché, permettant d'accélérer l'adoption de la plateforme par les éditeurs tiers.

L'objectif de ce POC est de développer un service de gestion documentaire implémentant le modèle et les API (REST) de CMIS, et qui soit par ailleurs scalable en tirant parti des services mis en place sur le POC1 ainsi que des services offerts par le module PaaS4Dev, le runtime OSGI de CompatibleOne.

2 Progrès réalisés vers ces objectifs

2.1 Rédaction de l'expression des besoins

Le livrable 29.1 ("Expression des besoins et spécifications techniques de la GED en ligne") a été rédigé à T0+4, et a servi de base aux développements qui ont suivi.

2.2 Rédaction des spécifications techniques du PaaS orienté GED multitenant

TODO: le livrable n'est pas fini.

¹http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=cmis

2.3 “OSGi-fication” de Nuxeo EP / Projet Apricot

Afin de porter la plateforme existante Nuxeo sur un environnement PaaS de type OSGi (sur le runtime PaaS4dev de Compatible One, mais aussi potentiellement sur n’importe quel runtime OSGi du marché), un travail important sur l’architecture de la plateforme Nuxeo EP préexistante a été nécessaire.

Pour ne pas pénaliser les utilisateurs et clients existants, mais aussi pour accélérer l’adoption de la nouvelle plateforme par les éditeurs tiers, il a été décidé de *forker* le code existant et d’en faire un projet Eclipse, baptisé dans un premier temps Eclipse ECR² (Enterprise Content Repository) puis Apricot³.

Le passage de Nuxeo à la norme OSGi impose de revoir en profondeur l’architecture de la plateforme, notamment la manière dont sont utilisés les services.

Une formation de l’équipe projet au sein de Nuxeo aux best practices OSGi a été organisée en juin 2011 auprès de Peter Kriens (CTO de l’OSGI Alliance). Il est à noter que la formation n’a pas permis de répondre à toutes les questions qui se posent sur l’intégration d’OSGi et de Java EE. Bien au contraire, elle a confirmé qu’il s’agissait bien de question ouvertes, sur lesquelles des efforts conséquents doivent être consentis pour espérer aboutir à un modèle qui rende compatible les deux approches.

Sur le même sujet, des discussions et des réunions de coordination avec l’équipe de JOnAS (sous-projet PaaS4dev), ainsi qu’avec d’autres projets open source exposés à des problématiques similaires (workshop sur les architectures Java à composants, organisé le 21 septembre 2011 avec le soutien de l’IRILL) ont permis également de creuser le sujet, sans aboutir pour l’instant à des réponses définitives.

En conclusion, si à ce jour un travail important a été réalisé pour mettre l’architecture de Nuxeo EP “d’équerre” pour rentrer dans le moule OSGi (modularisation), il reste encore de nombreux axes de travail pour 2012.

2.4 Nuxeo et le multi-tenancy

Nuxeo EP intègre depuis longtemps la notion de “domaine”, des conteneurs de documents qui apparaissent à la racine de la hiérarchie documentaire. L’idée était de faire tourner dans un même serveur des applications aux caractéristiques nettement différentes, par exemple avec des caractéristiques de stockages différentes (ex: un domaine pour les documents vivants, et un ou plusieurs autres pour les documents archivés).

Néanmoins l’implémentation actuelle ne pousse pas la logique jusqu’à son terme, et il reste encore de nombreux points de configuration qui sont

²<http://www.eclipse.org/proposals/rt.ecr/>

³<http://www.eclipse.org/projects/project.php?id=rt.apricot>

globaux au serveur alors qu'il devraient être localisé au niveau de chaque domaine.

Le projet nuxeo-multi-tenant⁴ a donc été démarré pour ajouter différents points de configuration locaux aux domaines.

2.5 Expérimentation du portage de Nuxeo sur des PaaS existants

Il nous est apparu utile d'expérimenter le portage de Nuxeo sur des PaaS préexistants à Compatible One afin de mieux cerner les enjeux et les difficultés d'un tel effort.

2.5.1 SlapOS

Des discussions ont eu lieu avec Nexedi pour comprendre comment déployer Nuxeo dans SlapOS.

Un certain nombre de contraintes opérationnelles (nécessité d'utiliser IPv6, packages qui ne s'installent pas sur Ubuntu...), et le fait que la communauté SlapOS ne soit pas familière des technologies Java, rendent ce travail un peu délicat et difficile à reproduire par des tiers, néanmoins lorsque l'accès à SlapOS aura été rendu plus aisé comme cela semble en prendre la direction.

2.5.2 Intalio

Intalio est un provider de SaaS qui a développé son propre cloud privé sur une technologie Jetty + OSGi. Nous avons discuté avec les développeurs d'Intalio de l'opportunité et des contraintes techniques d'un portage de Nuxeo sur leur plateforme, mais le caractère propriétaire de cette plateforme ne nous a pas permis d'aller plus loin.

2.5.3 CloudBees

CloudBee est un PaaS Java qui permet de déployer des WARs de manière contrôlée par le moteur de PaaS. Il

Des discussions techniques ont eu lieu avec l'équipe de CloudBees pour mieux comprendre ce qui est nécessaire, et un passage de Nuxeo en WAR pur a été réalisé. Néanmoins, il reste encore du travail avant d'espérer porter Nuxeo sur CloudBees.

2.5.4 CloudFoundry

CloudFoundry est un PaaS open source qui a été annoncé par VMWare / SpringSource au début de l'été. Cloudfoundry permet de déployer des

⁴<https://github.com/nuxeo/nuxeo-multi-tenant>

applications de plusieurs types (Ruby, Python, Java). Il paraissait naturel d'envisager un portage de Nuxeo sur cette nouvelle plateforme.

Des expérimentations ont eu lieu à la rentrée 2011. Elles ont montré qu'un tel portage était possible à condition de modifier CloudFoundry, ce qui heureusement est possible du fait que la plateforme est open source.

2.5.5 Ensemble / Juju

Ensemble, qui a été renommé ensuite Juju, est un projet de Canonical qui permet de déployer et de chorégraphier des services dans des machines virtuelles sur Amazon EC2 (pour l'instant).

Nuxeo a expérimenté le déploiement de plateformes multi-VM à l'aide de Juju. La technologie semble prometteuse, mais a besoin d'être industrialisée pour rendre les développements plus aisés.

Le code source du "charm" Juju dédié à Nuxeo est sur GitHub: <https://github.com/nuxeo/nuxeo-juju>

2.6 Nuxeo Cloud Controller

Nuxeo Cloud Controller (NCC) est un POC de moteur de cloud dédié à Nuxeo qui permet d'empiler les instances de Nuxeo dans une seule VM tout en les gérant de manière automatique, et donc d'optimiser l'utilisation des VMs et donc le coût de l'hébergement.

Il s'agit d'un prototype, destiné à évaluer les difficultés inhérentes à l'exercice.

Son architecture est similaire à celle de CloudFoundry, mais beaucoup plus simple du fait qu'elle se limite au cas de Nuxeo.

Le code source de NCC est sur GitHub: <https://github.com/compatible/nuxeo-cloud-controller>

2.7 Stockage non-relationnel

Afin de permettre le stockage à coût raisonnable de quantités massives de documents (plusieurs dizaines de téra-octets), l'approche distribuée popularisée par Amazon S3 semble prometteuse⁵.

Afin de mettre en oeuvre cette approche, sans pour autant se retrouver pied et poings liés à Amazon, les expérimentations suivantes ont été réalisées.

2.7.1 Stockage S3

Un "binary manager" pour Nuxeo basé sur S3 a été réalisé.

Ses sources sont sur GitHub: <https://github.com/nuxeo/cloudbinarymanager>.

⁵technologiquement l'approche S3 est probablement pertinente, en revanche elle est très onéreuse, cf. <http://www.backblaze.com/petabytes-on-a-budget-how-to-build-cheap-cloud-storage.html>.

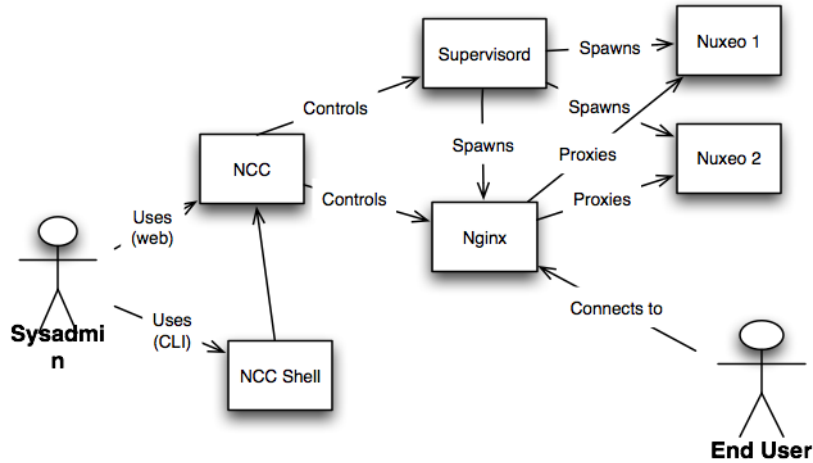


Figure 1: Architecture de NCC

2.7.2 Stockage Scality

Scality⁶ est une startup franco-américaine qui propose du “stockage organique” similaire à S3.

Le Binary Manager Nuxeo S3 a été subséquemment adapté pour permettre un stockage des BLOBs également dans Scality.

2.7.3 Stockage non relationnel multi-backends

L’objectif final est de permettre à Nuxeo de stocker ses BLOBs dans n’importe quel système de stockage à base de BLOBs, en se basant sur la librairie JClouds⁷, et de supporter tous les backends que supporte JClouds: S3 bien sûr, mais aussi: Eucalyptus, Nova et Swift (OpenStack), Cloudfile, ou bien un filesystem local pour faciliter les tests.

2.8 Publication / dissémination

Les travaux de Nuxeo relatifs au cloud ont été présentés lors de Nuxeo World par Stefane Fermigier⁸.

Un talk sur le projet a été donné par Bogdan Stefanescu et Florent Guillaume lors de l’EclipseCon Europ 2011 le 2 novembre 2011⁹.

⁶<http://www.scality.com/>

⁷<http://www.jclouds.org/>

⁸<http://www.slideshare.net/sfermigier/nuxeo-on-the-cloud-nuxeo-world-2011>

⁹<http://eclipsecon.org/sessions/your-first-content-application-apricot-project-presented-nuxeo>

Un blog post expliquant l'articulation de ces activités de R&D avec les objectifs de Nuxeo a été publié le 4 novembre 2011¹⁰.

3 Travaux en cours

Le travail continue sur les aspects suivants:

- OSGi-fication de Nuxeo, avec comme objectif de déployer Nuxeo dans un runtime distribué basé sur JOnAS.
- Packaging de Nuxeo en WAR pour tourner sur un PaaS Java non-distribué.
- Stockage de BLOBs distribué multi-backends via JClouds et benchmarking des solutions.
- Gestion fine du déploiement dans un PaaS de type CloudFoundry et/ou Juju.
- TODO

¹⁰<http://www.contentgeeks.net/2011/11/03/content-management-platforms-in-the-cloud-what-it-really-m>