

RÉPUBLIQUE DU CAMEROUN

Paix-Travail-Patrie

MINISTÈRE DE L'ENSEIGNEMENT
SUPÉRIEUR

Université de Yaoundé I

Institut Saint Jean



RÉPUBLIQUE DU CAMEROUN

Paix-Travail-Patrie

MINISTÈRE DES FINANCES

Secrétariat général

Centre National de Développement de
l'Informatique



RAPPORT DE STAGE ACADÉMIQUE

THÈME : CLOUD PRIVÉ ET CONTENEURISATION DES COMPOSANTS APPLICATIFS

Stage effectué du 21 Septembre 2020 au 12 Février 2021

**AU CENTRE NATIONAL DE DÉVELOPEMENT DE
L'INFORMATIQUE (CENADI)**

En vue de la validation du projet de stage

Option : **Système Réseaux**

Rédigé et Présenté par :

NDOMOUO NANA CARELLE

Etudiante en **Cycle Ingénieur quatrième année**

Sous l'encadrement

Académique de :

Dr MOUAFO Laetitia

Enseignante ISJ

Professionnel de :

MR NGENE Goddy

Chef DEL du CENADI

DÉDICACE

Je dédie ce travail à mes parents Mr et Mme NGUESSI.

REMERCIEMENTS

Nous remercions :

- ✚ Le Centre National de Développement de l'Informatique (CENADI) plus précisément sa Directrice de nous avoir accordé l'opportunité d'effectuer le stage dans ses locaux ;
- ✚ L'Institut Saint Jean de nous avoir donné la possibilité d'effectuer ce stage ;
- ✚ Mr EPIE NGENE GODDY notre encadreur professionnel pour le temps et la patience qu'il nous a accordé et pour la transmission de la passion ;
- ✚ Dr MOUAFO Laetitia notre encadreur académique pour le soutien ;
- ✚ Mr HOUSSEINI MOUSSA, Mr BILONG Hermann et Mr BELLO Bertrand pour l'accompagnement et les enseignements ;
- ✚ Tout le personnel du CENADI pour leurs disponibilités ;
- ✚ Notre Famille pour l'encadrement tout au long de ce stage.

SOMMAIRE

DÉDICACE.....	i
REMERCIEMENTS	ii
SOMMAIRE	iii
LISTE DES TABLEAUX.....	v
LISTE DES SCHÉMAS.....	vi
LISTE DES IMAGES ET DES FIGURES.....	vii
LISTE DES ABRÉVIATIONS.....	viii
AVANT-PROPOS.....	ix
INTRODUCTION GÉNÉRALE.....	1
CHAPITRE 1 : INSERTION DANS L'ENTREPRISE	2
INTRODUCTION.....	3
I- PRÉSENTATION DE L'ENTREPRISE	3
1- Les Services.....	4
2- Organisations	5
3- Infrastructures et réalisations.....	10
II- DÉROULEMENT DU STAGE.....	15
1- Imprégnation.....	15
a- Visite guidée	16
b- Apprentissage IBM.....	16
i- Définition.....	16
ii- Caractéristiques Physiques	17
iii- Systèmes Supportés	17
iii.1- z/OS	17
iii.2- z/VM.....	18
iii.3- z/VSE.....	19
iii.4- Linux for zSeries	19
iii.5- z/TPF.....	20

iv-Les spécificités du mainframe	20
2-Les activités.....	22
a- Les séances « master the mainframe »	22
b- Les Séances de POC.....	23
CONCLUSION	24
CHAPITRE 2 : LE PROBLÈME POSÉ	25
INTRODUCTION.....	26
I- ÉTUDE DU PROBLÈME.....	26
1- Les problèmes rencontrés.....	26
2- Étude de la solution	27
a- Cloud privé.....	27
b- Conteneurisation.....	29
II- LES OUTILS PRÉSENTS SUR LE MARCHÉ	32
CONCLUSION	37
CHAPITRE 3 : MÉTHODOLOGIE DE RÉOLUTION DU PROBLÈME	38
INTRODUCTION.....	39
I- LA SOLUTION ADOPTÉE	39
1- Présentation d’Openshift.....	39
2- Étape d’implémentation	43
II- LE RÉSULTAT.....	46
CONCLUSION	48
CONCLUSION GÉNÉRALE.....	49
LEXIQUE.....	50
SOURCES	51
ANNEXES	52

LISTE DES TABLEAUX

<i>TABLEAU 1- HISTORIQUE.....</i>	<i>4</i>
<i>TABLEAU 2- RÉALISATIONS N°1.....</i>	<i>12</i>
<i>TABLEAU 3- RÉALISATIONS N°2.....</i>	<i>13</i>
<i>TABLEAU 4- RÉALISATIONS N°3.....</i>	<i>14</i>
<i>TABLEAU 5- RÉALISATIONS N°4.....</i>	<i>15</i>
<i>TABLEAU 6- RÉALISATIONS N°5.....</i>	<i>15</i>
<i>TABLEAU 7- ACTIVITÉS DU POC.....</i>	<i>24</i>

LISTE DES SCHÉMAS

<i>SCHÉMA 1- ORGANIGRAMME CENADI.....</i>	<i>6</i>
<i>SCHÉMA 2- DIFFÉRENCES ENTRE LES SERVICES.....</i>	<i>29</i>
<i>SCHÉMA 3- DIFFÉRENCES ENTRE CONTENEURS ET MACHINES VIRTUELLES.....</i>	<i>31</i>
<i>SCHÉMA 4- FONCTIONNEMENT DE KUBERNETES.....</i>	<i>31</i>
<i>SCHÉMA 5- ARCHITECTURE FOUNDRY.....</i>	<i>34</i>
<i>SCHÉMA 6- ARCHITECTURE WSO2.....</i>	<i>34</i>
<i>SCHÉMA 7- ARCHITECTURE CLOUDIFY.....</i>	<i>35</i>
<i>SCHÉMA 8- ARCHITECTURE STACKATO.....</i>	<i>36</i>
<i>SCHÉMA 9- ARCHITECTURE OCP.....</i>	<i>42</i>

LISTE DES IMAGES ET DES FIGURES

<i>IMAGE 1- CONSOLE WEB OCP.....</i>	<i>35</i>
<i>IMAGE 2- CONSOLE WEB TSURU.....</i>	<i>36</i>
<i>IMAGE 3- CONSOLE WEB ALIBABA.....</i>	<i>37</i>

<i>FIGURE 1- STRUCTURE OCP.....</i>	<i>41</i>
-------------------------------------	-----------

LISTE DES ABRÉVIATIONS

API : Application Programming Interface

Apps : Applications

BIOS : Basic Input Output System

CENADI : Centre Nationale de Développement de l'Informatique

CLI : Command Line Interface

DEL : Division de l'Exploitation et des Logiciels

DNS : Domain Name System

GUI : Graphical User Interface

IBM : International Business Machine

Lpar : Logical Partition

MINFI : Ministère des Finances

OS : Operating System

POC : Proof Of Concept

RACF : Resource Access Control Facility

SAF : System Authentication System

SE : Système d'Exploitation

VM : Virtual Machine

VPN : Virtual Private Network

AVANT-PROPOS

L'institut saint jean est une école de formation professionnelle qui offre à ses étudiants trois cycles à savoir un cycle ingénierie sur cinq (5) ans, un cycle de licence sur trois (3) ans et un cycle Master sur deux (02) ans. Créé en 2016 par les pionniers de Prépa-Vogt, il a pour objectif de :

- ✚ Favoriser le développement et le plein potentiel de tous les jeunes selon leurs habiletés ;
- ✚ Répondre au désir de formations techniques professionnelles et d'ingénierie de haut niveau sans devoir quitter le Cameroun ;
- ✚ Développer chez les jeunes l'autonomie, le sens des responsabilités et l'amour d'un travail éthique.

Dans le but d'atteindre leurs objectifs, l'Institut Saint Jean a mis en place un système organisé incluant les stages académiques, offrant ainsi à ses étudiants la possibilité de se familiariser au monde professionnel. En cycle ingénieur quatrième année, il est instauré un stage académique de six (06) mois qui représente le premier semestre annuel. C'est dans le cadre de l'accomplissement de notre stage que nous rédigeons ce rapport.

INTRODUCTION GÉNÉRALE

Disposer d'une infrastructure résiliente et hautement sécurisée est un défi pour les organisations qui hébergent des données et des systèmes critiques. Les différentes avancées technologiques aussi bien sur les aspects matériels (hardware), logiciel (OS et apps), d'architecture et des opérations, ont concouru au développement des technologies sécurisées, garantissant des opérations continu qui ont permis de repousser les limites de l'indisponibilité desdits systèmes. Comprendre les défis de ces organisations aujourd'hui suppose une rétrospective sur les technologies antérieures et les motivations ayant entraîné les transitions subséquentes.

Le Centre National de Développement de l'Informatique qui est l'hébergeur attitré des grands systèmes de l'Etat du Cameroun n'a pas été exempt des mutations technologiques successives qui ont forgé la pratique de son métier et déterminé certaines orientations propres à son ADN qui est le traitement des données. C'est fort des défis susmentionnés et d'une analyse conjoncturelle de son marché que le Centre entreprendra de migrer son infrastructure vers le Cloud. Cependant, la nature sensible des données manipulées par le Centre et la nécessité d'appuyer techniquement les administrations publiques et parapubliques, a contraint le Centre à conditionner son choix aux exigences de confidentialité de son infrastructure.

Face aux problèmes cités plus haut, le CENADI nous a proposé le choix d'un Cloud privé couplé à une conteneurisation des applications comme solution offerte aux tiers (administrations publiques et para), pour un déploiement hautement sécurisé et continu de leurs applications, garantissant une disponibilité même dans l'hypothèse d'une maintenance active.

Ce rapport fera objet de restitution de ces six (06) mois d'apprentissage autour du thème : Cloud privé et conteneurisation des composants applicatifs suivants trois (03) principaux axes qui sont : Insertion dans l'Entreprise, Problème Posé et Méthodologie de résolution du problème.

CHAPITRE 1 : INSERTION DANS L'ENTREPRISE

INTRODUCTION

Pour les étudiants, le stage est l'occasion parfaite pour s'imprégner du monde professionnel, mettre en pratique les connaissances apprises en cours. Dans notre cas, le stage de quatrième (4^{ème}) année qui s'est déroulé au CENADI sur une période approximative de six (06) mois, nous a surtout permis de découvrir des secteurs peu connus voir pas connus du monde informatique que sont le système z et le déploiement d'applications. Il sera question de présenter l'entreprise qui nous a accueillies, les activités que nous avons menées et une appréciation du stage, tout au long de cette partie.

I- PRÉSENTATION DE L'ENTREPRISE

Le CENADI est une structure rattachée au Ministère des Finances (MINFI), qui a pour mission la mise en œuvre de la politique du gouvernement dans le domaine de l'informatique et de la téléinformatique, ainsi que le développement des méthodes informatiques, téléinformatiques dans tous les secteurs de la vie nationale. En outre, il veille au secret, à la confidentialité et à l'intégrité des données et des informations. Il est ainsi le conseil du gouvernement, des administrations publiques et parapubliques, des collectivités locales et même des entreprises privées, en matière d'informatique.

Il a été institué le 12 août 1988 au Ministère de l'Enseignement Supérieur de l'Informatique et de la Recherche Scientifique (MESIRES), mis sous tutelle du MINFI le 10 mai 1993, et maintenu rattaché au MINFI le 09 décembre 2011.

Le tableau ci-après récapitule les grandes lignes de l'historique du CENADI.

APPELATIONS	ORGANISME DE RATTACHEMENT	ANNÉE
Service Central de la Mécanographie (SCM)	Rattaché au Secrétariat d'Etat au Plan	1960
Service Central de l'Informatique (SCI)	Rattaché au Secrétariat Général (SG) de la Présidence de la République (PR)	1967

Direction Centrale de l'Informatique et de la Télématic (DCIT)	Rattachée au Secrétariat Général (SG) de la Présidence de la République (PR)	1976
Direction de l'Informatique et de la Télématic (DIT)	Ministère de l'Informatique et des Marchés Publics (MINMAP).	1985
Centre National de Développement de l'Informatique (CENADI)	Sous tutelle Ministère de l'Enseignement Supérieur, de l'Informatique et de la Recherche Scientifique (MESIRES).	1988
CENADI	Sous la tutelle du Ministère de la Recherche Scientifique et Technique (MINREST).	1991
CENADI	Sous la tutelle du Ministère des Finances	1993
CENADI	Rattaché au Ministère des Finances.	2007
CENADI	Maintenu rattaché au Ministère des Finances.	2011

Tableau 1- Historique

1- Les Services

Le CENADI offre les services suivant :

- ✚ Conception, mise en place et exploitation des systèmes d'information ou de conduite de processus; destiné à la gestion, à la production des biens et services, à l'enseignement, aux opérations de toute nature impliquant l'usage immédiat ou futur de l'Informatique et de la Télématic ;
- ✚ La négociation des protocoles d'accord avec les constructeurs et les fournisseurs de matériel et de logiciels informatiques ;
- ✚ Définition et gestion de façon exclusive des banques d'informations et les réseaux de transmission de données implantés dans l'administration, dans les organismes parapublics et éventuellement dans le privé ;
- ✚ Supervision de l'exploitation des ordinateurs installés dans les secteurs publics, parapublics et éventuellement privés ;

- ✚ Mise en disposition de la clientèle toute ou une partie de ses ressources en conseils, matériels, logiciels ou fichiers de données ;
- ✚ Participation à l'application des soumissions pour les appels d'offres de marchés d'informatisation des opérations de l'État ;
- ✚ Contrôle sur les opérations d'automatisation ou d'organisation effectuées par les sociétés de services ou les utilisateurs eux-mêmes sur demande du gouvernement.

2- Organisations

La structure organisationnelle du CENADI comprend :

- ✚ MINFI ;
- ✚ Un Conseil de Direction ;
- ✚ Une Direction ;
- ✚ Un Secrétariat ;
- ✚ La Division des Etudes et Projets (DEP) ;
- ✚ La Division de l'Exploitation et des Logiciels (DEL) ;
- ✚ La Division de la Téléinformatique et de la Bureautique (DTB) ;
- ✚ La Division de l'Informatique appliquée à la Recherche et à l'Enseignement (DIRE) ;
- ✚ La Division des Affaires Administratives et Financières (DAAF).

Le CENADI a également les branches extérieures telle que :

- ✚ Le Centre Informatique de Douala (CID) ;
- ✚ Le Centre Informatique de Garoua (CIG) ;
- ✚ Le Centre Informatique de Bafoussam (CIB).

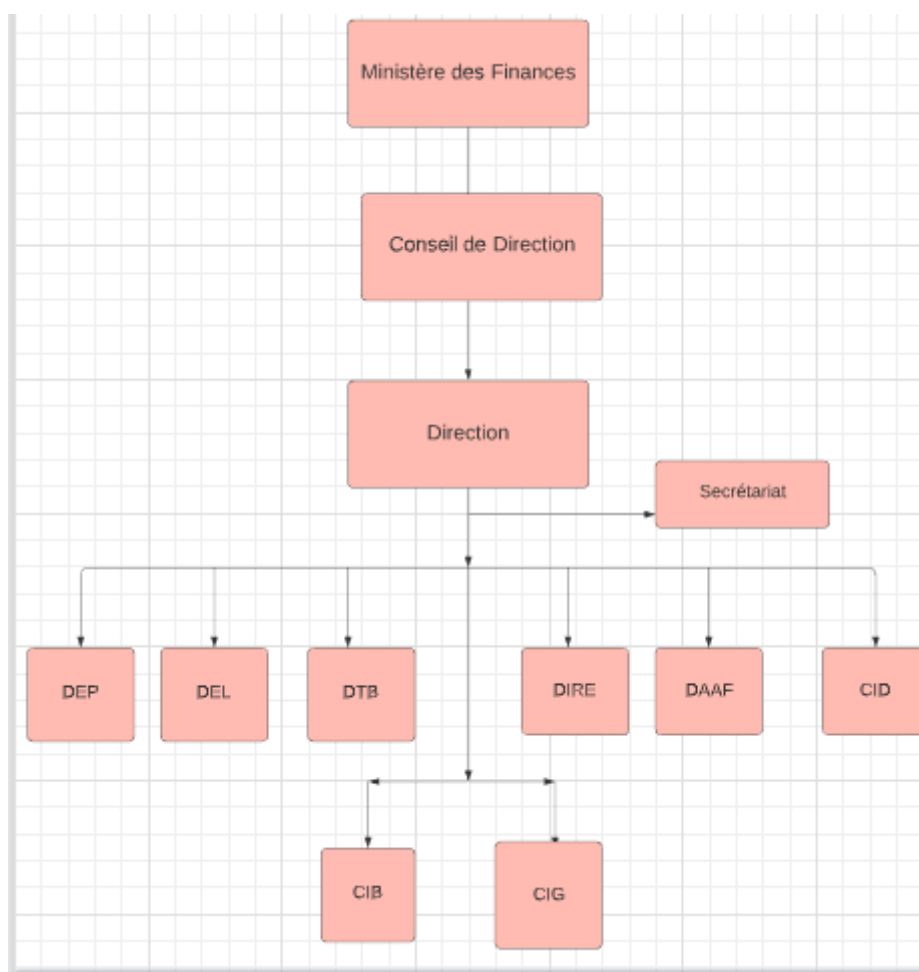


Schéma 1- Organigramme CENADI

CONSEIL DE DIRECTION : le Conseil a les pouvoirs les plus étendus pour l'administration et la gestion du CENADI. Il lui appartient notamment :

- + d'arrêter la politique générale du centre et de contrôler ses activités ;
- + d'adopter le règlement intérieur ;
- + d'approuver le programme d'action et le rapport annuel d'activités ;
- + d'approuver le budget provisionnel et les programmes généraux d'investissement ;
- + d'approuver le rapport financier de l'exercice écoulé ainsi que les comptes et bilans ;
- + de fixer les conditions de passation des marchés relatifs aux travaux et fournitures du centre ;

- ✚ d'autoriser la passation de toute convention relative à la gestion et au fonctionnement du centre ;
- ✚ de décider de l'affectation des résultats d'exploitation.

DIRECTION : Le CENADI est placé sous l'autorité d'un Directeur.

SÉCRÉTARIAT : Il est placé sous l'autorité d'un Chef de Secrétariat.

DIVISION DES ÉTUDES ET PROJETS : Placée sous l'autorité d'un Chef de Division, la DEP est chargée de l'étude des problèmes de développement à moyen et long termes, de l'informatique et plus particulièrement :

- ✚ d'adapter et de finaliser les systèmes informatiques ;
- ✚ de mener à bien l'analyse informatique ;
- ✚ de rechercher de nouvelles opportunités de développement informatique dans les domaines scientifiques et de gestion ;
- ✚ de procéder aux études de faisabilité préalable à toute mécanisation ;
- ✚ de définir les standards d'analyse et de programmation applicables dans les centres informatiques et orientés vers la modularité ;
- ✚ de concevoir des études et des packages généraux destinés à la gestion ;
- ✚ d'assurer la maintenance des doubles des dossiers d'analyse et de programmation de tous les projets ;
- ✚ d'assurer la mise en œuvre technique des différents projets informatiques en veillant à l'harmonisation des codes de nomenclature utilisés dans les traitements ;
- ✚ d'assurer la maintenance des applications en exploitation.

DIVISION DE L'EXPLOITATION ET DES LOGICIELS : Placée sous l'autorité d'un Chef de Division, la DEL est chargée, entre autres, de :

- ✚ d'assurer la maintenance des logiciels fournis par les constructeurs d'ordinateurs ou les sociétés de services ;

- + de veiller à l'optimisation et à l'harmonisation des procédures de sécurité appliquées dans les centres informatiques ;
- + de procéder aux essais de nouveaux logiciels préalablement à leur mise en exploitation ;
- + de définir des normes de sécurité en ce qui concerne l'exploitation des archives magnétiques ;
- + de veiller au secret et à l'intégrité des informations ;
- + de veiller à l'adéquation des programmes-sources conservés par la Division des Etudes et des Projets avec les load-modules en exploitation ;
- + de procéder à l'exécution sur ordinateurs de l'ensemble des travaux dans les délais prévus ;
- + de veiller au bon fonctionnement des matériels installés ;
- + d'effectuer des études en vue d'émettre des propositions, au regard de l'état de l'art, pour une éventuelle modification du parc de matériels installés ;
- + de participer à la réalisation des projets informatiques en ce qui concerne le choix des logiciels et la mise en œuvre technique desdits projets.

DIVISION DE LA TÉLÉINFORMATIQUE ET DE LA BUREAUTIQUE : Placée sous l'autorité d'un Chef de Division, la DTB est chargée de définir et de gérer de façon exclusive des banques d'information implantées dans les services publics, parapublics et éventuellement privés et les organismes parapublics ainsi que les réseaux nationaux de transmission de données. Elle assure notamment :

- + la mise en œuvre de la politique de développement des réseaux conformément aux objectifs fixés par le Gouvernement ;
- + la coordination de l'action commerciale ;
- + les relations fonctionnelles avec les représentants du Ministère des Postes et Télécommunications ;
- + les relations fonctionnelles avec tous raccordements compatibles avec l'exploitation harmonieuse des réseaux ;
- + le développement des applications télématiques ;
- + la gestion et la maintenance des équipements de transmission de données ;
- + la mise en œuvre des serveurs.

DIVISION DE L'INFORMATIQUE APPLIQUÉE A LA RECHERCHE ET À L'ENSEIGNEMENT : Placée sous l'autorité d'un Chef de Division, la DIRE est chargée :

- + de l'organisation des stages de recyclage en informatique ;
- + de l'harmonisation des programmes de formation en Informatique en accord avec les administrations et les organismes concernés ;
- + de la mise en forme, de la conservation et de la diffusion des documents ;
- + de l'encadrement des stagiaires ;
- + de l'information et de la formation des nouvelles recrues ;
- + de l'archivage par micrographie ;
- + de la recherche ;
- + du développement de la micro-informatique et de la Bureautique ;
- + d'assurer la liaison avec les organismes et les centres nationaux de recherche.

DIVISION DES AFFAIRES ADMINISTRATIVES ET FINANCIÈRES : Placée sous l'autorité d'un Chef de Division, la DAAF est chargée :

- + des problèmes de sécurité ;
- + de la gestion du personnel du Centre ;
- + du contrôle de la gestion du personnel des centres informatiques ;
- + du traitement des affaires contentieuses, en liaison avec la cellule juridique ;
- + du contrôle de la gestion financière des centres informatiques ;
- + de la préparation et de l'exécution du budget du Centre ;
- + de la gestion du matériel et des véhicules affectés au Centre ;
- + de la gestion du magasin d'imprimés et fournitures informatiques ;
- + de l'entretien des matériels ;
- + du suivi des problèmes contractuels des fournisseurs ;
- + de la facturation et du recouvrement des créances.

BRANCHES EXTÉRIEURES : Le CENADI dispose de trois Centres Informatiques respectivement situés dans les villes de Douala, Garoua et Bafoussam.

- ✚ Le Centre Informatique de Douala (CID) est placé sous l'autorité d'un Chef de Division;
- ✚ Le Centre Informatique de Garoua (CIG) est placé sous l'autorité d'un Chef de Centre;
- ✚ Le Centre Informatique de Bafoussam (CIB) est placé sous l'autorité d'un Chef de Centre.

Placé sous l'autorité d'un Chef de Centre éventuellement assisté d'un Adjoint. Ils sont chargés, entre autres:

- ✚ de veiller à l'application des instructions, des procédures, des techniques informatiques définies par les instances supérieures ;
- ✚ de mener à bien l'analyse informatique depuis la prise de connaissance de problèmes jusqu'à leur mise en exécution sur ordinateurs ;
- ✚ de procéder aux études de faisabilité préalables à toute mécanisation;
- ✚ d'assurer la maintenance des doubles de dossiers d'analyse et de programmation de tous les projets ;
- ✚ d'assurer la programmation des objectifs trimestriels assignés au personnel ;
- ✚ d'assurer la mise en œuvre technique des différents projets informatiques en veillant à l'harmonisation des codes et nomenclatures utilisés dans les traitements ;
- ✚ d'assurer la maintenance des logiciels fournis par les constructeurs d'ordinateurs ou les sociétés de services ;
- ✚ de veiller à la confidentialité et à l'intégrité des informations traitées.

3- Infrastructures et réalisations

Le CENADI possède de nombreuses plateformes technique constituées de serveurs, machines et technologies telles que :

- ✚ Mainframe IBM Z10
- ✚ Mainframe IBM z14
- ✚ SUN Supercluster/Exadata
- ✚ Des Imprimantes Infoprint 4100 et 4000 IBM

- ✚ des enregistreurs de cassettes magnétiques permettant l'enregistrement des jobs ont remplacés les bandes magnétiques qui ne sont plus utilisés actuellement ;
- ✚ un pupitre chargé de répondre aux demandes des machines et d'envoyer ces demandes à d'autres machines ;
- ✚ une alarme permettant la signalisation de tous les problèmes qui surviennent en salle machines ;
- ✚ des caméras permettant la surveillance visuelle de toute personne accédant à la salle machine ;
- ✚ un système de détection de fumée pour éviter les incendies ;
- ✚ des climatiseurs pour maintenir la salle machine à une température constante ;
- ✚ des routeurs Cisco 2600 séries ;
- ✚ des switch (catalyst 2900 séries...) ;
- ✚ des routeurs (Cisco 7200) ;
- ✚ une liaison spécialisée permettant la transmission de données sensibles ;
- ✚ des imprimantes matricielles (infoprint 6500) ;
- ✚ une baie pour le réseau LAN du CENADI ;
- ✚ une baie pour le réseau d'interconnexion avec les sites SIGIPES ;
- ✚ des serveurs tours et racks qui hébergent les services réseaux (messagerie, web DNS) ;
- ✚ Le courant régulé, injecté dans le réseau de distribution de l'électricité aux équipements électroniques, qui dessert l'ensemble des bureaux et des locaux techniques, est contrôlé par un onduleur central suppléé par un groupe électrogène ;
- ✚ La climatisation de la salle machines et des bureaux est assurée par des dispositifs de systèmes autonomes.


En ce qui concerne des projets réalisés depuis sa création peut se résumer en ces tableaux :

✚ De 1982 à 1988

N°	LIBELLÉ DU PROJET	ANNÉE
1	Cameroon Packet Switching Network (CAMPAC)	1982

2	Procédures Automatisées de Gestion des Opérations de la Douane et Commerce Extérieur (PAGODE)	1984
3	Informatique du Budget, de l'Intendance et des Services de l'État (IBIS)	1985
4	Application Nationale pour le Traitement Informatique et Logistique du Personnel de l'État (ANTILOPE)	1986
5	Traitement Informatisé des Impôts et Taxes de l'État (TRINITE)	1988

Tableau 2- Réalisations n°1

 De 1989 à 2008

N°	LIBELLÉ DU PROJET	ANNÉE
1	Plan National Informatique (PNI)	1990
2	Programme Appliqué au Traitement Informatique des Opérations du Trésor (PATRIOT)	1992
3	système Informatisé de Gestion Intégrée du Personnel de l'Etat et de la Solde (SIGIPES)	1994
4	Passage à L'an 2000	1999
5	Application ORDRE (Ordonnancement Rationnel et Diligent des Règlements de l'Etat) au Ministères des Finances (MINFI)	1999
6	Etude et Réalisation des Interfaces entre les Systèmes Informatiques du Ministère des Finances et du Budget	2000
7	Système Douanier Automatisé (SYDONIA)	2003
8	Audit du site Internet du CENADI	2006
9	Evaluation des deux applications de gestion des ressources humaines et de la solde (SIGIPES et ANTILOPE)	2007
10	Constitution du Club DSI-Cameroun	2008

11	Elaboration de la Charte Nationale des Investissements au Cameroun, volet TIC	2008
----	---	------

Tableau 3- Réalisations n°2

 De 2009 à 2011

N°	LIBELLÉ DU PROJET	ANNÉE
1	Stratégie Intégrée de Gestion Informatisée des Finances Publiques (SIGIFiP)	2009
2	Mise en place d'un Centre de Données au CENADI	2009
3	Elaboration des textes sur les communications électroniques, la cybersécurité et la cybercriminalité	2009
4	Réalisation des études relatives à la mise en œuvre du programme d'appui à la composante technologique et professionnelle de l'Enseignement Supérieur	2009
5	Relecture du statut particulier du corps des Fonctionnaires de l'Informatique et de la Télématic	2009
6	Comité de Suivi-Evaluation en charge de la Modernisation de l'Administration Economique et de la Gouvernance Electronique	2010
7	Conception et réalisation du portail web des Cinquantenaires de l'Indépendance et de la Réunification du Cameroun et du streaming des radios et des télévisions	2010
8	Stratégie e-Government du Cameroun avec le MINPOSTEL	2010
9	Négociation des accords-cadres avec les éditeurs des logiciels (gestion des licences globales) pour l'Administration Publique	2010
10	Evaluation du système informatique de l'HGOPY	2010
11	Mise en place d'un logiciel de gestion à l'HRB	2010

12	Plateforme informatique intégrée de mobilisation des ressources extérieures non génératrices d'endettement	2011
13	Schéma Directeur Informatique (SDI) du MINFI	2011
14	Appui à l'organisation de l'élection présidentielle du 09 octobre 2011	2011

Tableau 4- Réalisations n°3

De 2012 à 2017

N°	LIBELLÉ DU PROJET	ANNÉE
1	Phase pilote de la refonte de l'immatriculation des contribuables au Cameroun	2012
2	Audit du Système d'Information et la mise à jour du Schéma Directeur Informatique à la SOPECAM	2012
3	Comité chargé de faire la lumière sur l'affaire des factures en instance du Cabinet JYD & Consultants SARL	2012
4	Acquisition du matériel informatique, déploiement du SIGIPES à l'Université de Buea et acquisition des équipements et logiciels pour la gestion des équivalences académiques au MINESUP	2013
5	Sécurisation des Systèmes d'Information d'INTERPOL	2013
6	Fourniture, livraison, installation, mise en service et mise en service après-vente d'un système intégré de gestion des ressources humaines et de la solde de l'Etat du Cameroun (SIGIPES II)	2013
7	Informatisation de la Gestion du Suivi, de l'exécution des sanctions du Conseil de Discipline Budgétaire et Financière (CDBF)	2014

8	Etude en vue de l'élaboration du Plan National Informatique du Cameroun	2014
9	Fourniture et mise en service d'un nouveau système d'identification sécuritaire pour la République du Cameroun	2014
10	Etude de faisabilité en vue de l'extension du réseau informatique au niveau des contrôles financiers départementaux (CFD)	2015
11	Conception et réalisation du site web de la DGD	2015
12	Système de gestion de l'e-visa (Visa électronique) au Cameroun	2017
13	Projet d'opérationnalisation des bureaux des Douanes du Port Autonome de Kribi	2017
14	Développement de l'application informatique de gestion des Bourses et Stages	2017

Tableau 5- Réalisations n°4

 De 2018 à 2019

N°	LIBELLÉ DU PROJET	ANNÉE
1	Modernisation et sécurisation de la plateforme technique du CENADI	2018

Tableau 6- Réalisations n°5

II- DÉROULEMENT DU STAGE

Notre stage s'est déroulé du 21 Septembre 2020 au 12 Février 2021, marqué par une période d'imprégnation et celle d'apprentissage.

1- Imprégnation

Le CENADI dans sa mission de protection des données et systèmes sensibles de l'État a en son sein différentes plateformes techniques, dont la plus récente est le serveur Z14.

À notre arrivée nous n'étions pas familiers à d'autres systèmes que les systèmes x86, le Directeur Technique dans son rôle d'encadreur a tenu à nous imprégner au monde du mainframe et à nous présenter les capacités des machines IBM qui garantissent des performances inégalées sur le marché des centraux. Cette imprégnation s'est traduite par une visite des salles machines et une période d'apprentissage des mainframes IBM.

a- Visite guidée

La visite s'est articulée par une présentation de notre personne auprès du personnel de la DEL et ensuite une visite de la salle machine du CENADI constituée de toutes les technologies sus citées (confère I.3- Infrastructures et Réalisations). Cette salle est gardée sous une température ambiante très basse et protégée par des caméras, des détecteurs de fumée, l'accès à la salle des serveurs Z14 et SUN Exadata se fait par empreinte digitale et chaque serveur IBM est couplé à une baie de stockage et une baie réseau. Les mainframes utilisés par le CENADI utilisent les architectures processeurs s390 du fait des besoins en calcul et des exigences de sécurité.

b- Apprentissage IBM

International Business Machine (IBM) est un géant de fabrication de supercalculateur établi depuis le vingtième (20^{ème}) siècle. Il a commencé à se faire un nom avec ses premières machines S360 et S70 des années 1960 et 1970. Au fil des années certains experts ne donnaient pas d'avenir à IBM trouvant leur machines désuètes et encombrantes face aux avancées technologiques sur l'avènement du portable (téléphone, ordinateur), et ils donnèrent même pour nom « dinosaures » aux mainframes. Force est de constater que malgré toutes les prédictions sur la fin des mainframes, ceux-ci sont beaucoup plébiscités par les entreprises (en 2017, 70% des systèmes des grandes entreprises sont gérés par les mainframes). Les raisons pour lesquelles les mainframes n'ont pas disparu et continuent de s'épandre vont être détaillées dans cette partie.

i- Définition

Contrairement à ce qu'on a l'habitude de penser les mainframes ne sont pas des serveurs, du moins c'est ce que les experts IBM expliquent dans leur œuvre Introduction to the New

Mainframe, paru en 2006. Les serveurs suivent un modèle distribué tandis que les mainframes ont un modèle centralisé (définition des concepts en lexique), en d'autres termes les serveurs ne peuvent pas fournir tous les services d'une entreprise tandis que le mainframe à lui seul effectue toutes les tâches de l'entreprise. Par exemple pour une entreprise utilisant ces serveurs, elle aura besoin d'un serveur web, d'un serveur de données, d'un serveur mail, d'un serveur de transactions, etc. une entreprise utilisant un mainframe quant à elle verra tous ses services gérés par le mainframe, les entreprises peuvent l'utiliser pour héberger des données commerciales, des serveurs d'opération et des applications qui demandent un degré de sécurité et de disponibilité supérieur à celles qu'offrent d'autres machines d'où leur appellation de « supercalculateur ».

L'habilité de celui-ci à reconfigurer dynamiquement les ressources matérielle et logicielle (processeur, mémoire et connexions matérielle) pendant que les applications continuent de tourner, la capacité de son système de machines virtuelles (z/VM) à partager toute les ressources système avec des niveaux très élevés d'utilisation des ressources tout en gérant les ressources en tant que pod unique, fait du mainframe l'infrastructure idoine pour implémenter un Cloud privé.

Les mainframes sont utilisés dans les banques, en finance, santé, assurance, par les gouvernements, par des entreprises publiques ou privées. Ces supercalculateurs malgré le fait qu'ils ne soient pas visibles par tous, sont beaucoup utilisés au quotidien.

ii- Caractéristiques Physiques

L'un des derniers mainframes en date d'IBM est le Z14, créé en 2017 avec des caractéristiques physiques difficilement égalables par le système x86 : une capacité de disque de treize (13) à vingt-cinq (25) téraoctets, quarante (40) partitions natives (Lpar), Un (01) téraoctets de mémoire vive extensible à 4 téraoctets, des processeurs centraux, spécialisés.

iii- Systèmes Supportés

La liste est exhaustive mais nous pouvons citer entre autres :

iii.1- z/OS

z/ Operating System est un ensemble de programmes qui gèrent le fonctionnement interne du système informatique du mainframe. Ce système d'exploitation est conçu

pour utiliser au mieux les diverses ressources du supercalculateur et veiller à ce que la quantité maximale de travail soit traitée aussi efficacement que possible. Bien que le z/OS ne puisse pas augmenter la vitesse du mainframe, il peut maximiser son utilisation, ce qui rend la machine plus rapide en lui permettant de travailler plus dans une période de temps donnée. z/OS est conçu pour offrir une solution stable, sécurisée et un environnement disponible en permanence pour les applications exécutées sur le mainframe. Dans la plupart des premiers systèmes d'exploitation, les demandes de charge de travail entraient dans le système une à la fois. Le système d'exploitation traitait chaque demande comme une unité et ne démarrait pas la requête suivante jusqu'à ce que celle en cours de traitement ne soit terminée. Cet arrangement fonctionnait bien lorsqu'un travail pouvait s'exécuter en continu du début à la fin ; cependant dans certains cas un travail devait attendre que des informations soient lues ou écrites sur périphérique tel qu'un lecteur de bande ou une imprimante (car les Input/Output prennent plus de temps que la vitesse électronique du processeur) et pendant qu'un travail attendait les Input/Output le processeur restait inactif. Afin de maintenir le processeur en état de marche pendant qu'une tâche attendue augmenterait la quantité totale de travail que le processeur pourrait supporter sans avoir besoin de matériel, le z/OS a été mis sur pied. Celui-ci divise le travail et donne des portions de travail à divers composants du système et sous-systèmes qui fonctionnent inter dépendamment ; à tout moment un composant ou un autre prend le contrôle du processeur, apporte sa contribution, puis passe le contrôle à un utilisateur, un programme ou un autre composant.

iii.2- z/VM

z/Virtual Machine (z/VM) comprend deux composants de base : un programme de contrôle (CP) et un système d'exploitation mono-utilisateur, CMS. En tant que programme de contrôle, z/VM est un hyperviseur car il exécute d'autres systèmes d'exploitation dans les machines virtuelles qu'il crée. N'importe quel systèmes d'exploitation mainframe IBM tels que, z/OS Linux for zSeries, z/VSE et z/TPF, peuvent être exécutés en tant que systèmes invités sur leurs propres machines virtuelles, et z/VM peut exécuter n'importe quelle combinaison de systèmes invités. Le programme de contrôle crée artificiellement plusieurs machines virtuelles à partir des

ressources matérielles réelles. Le programme de contrôle assure la sécurité des données et des applications parmi les systèmes invités. Pour la plupart des clients, l'utilisation des systèmes invités évite le besoin de configurations matérielles plus importantes. L'autre composant majeur de z/VM est le système de surveillance conversationnelle ou CMS. Ce composant de z/VM s'exécute sur une machine virtuelle et fournit à la fois une interface utilisateur interactive et une interface de programmation d'application générale z/VM.

iii.3- z/VSE

z/ Virtual Storage Extended (z/VSE) est populaire auprès des utilisateurs d'ordinateurs mainframe plus petits. Certains de ces clients finissent par migrer vers z/OS lorsqu'ils dépassent les capacités de z/VSE. Par rapport à z/OS, le système d'exploitation z/VSE fournit une base plus petite et moins complexe pour le traitement des transactions. La conception et la structure de gestion de z/VSE est excellente pour exécuter des charges de travail de production de routine consistant en plusieurs travaux par lots (exécutés en parallèle) et un traitement de transactions traditionnel étendu. Dans la pratique, la plupart des utilisateurs de z/VSE disposent également du système d'exploitation z/VM et l'utilisent comme interface de terminal général pour le développement d'applications z/VSE et la gestion du système. z/VSE était à l'origine connu sous le nom de système d'exploitation de disque (DOS) et était le premier système d'exploitation basé sur disque introduit pour le mainframe System/ 360 des ordinateurs. DOS a été considéré comme une mesure temporaire jusqu'à ce qu'OS/360 soit prêt. Cependant, certains clients mainframe ont apprécié sa simplicité et ont décidé de rester avec lui après la disponibilité d'OS/360.

iii.4- Linux for zSeries

Plusieurs distributions Linux (non IBM) peuvent être utilisées sur un mainframe. Il existe deux noms génériques pour ces distributions : linux pour S/390 (utilise un adressage 31 bits), linux pour zSeries (utilise un adressage 64 bits). L'expression Linux sur zSeries est utilisée pour désigner Linux s'exécutant sur un système S/390 ou zSeries, lorsqu'il n'est pas nécessaire de se référer explicitement à la version 31 ou 64 bits.

iii.5- z/TPF

Le système d'exploitation z/ Transaction Processing Facility (z/TPF) est un système spécial utilisé par les entreprises ayant un volume de transactions très élevé, telles que les sociétés de cartes de crédit et les systèmes de réservation des compagnies aériennes. z/TPF était autrefois conçu pour les compagnies aériennes d'où son ancien nom Airline Control Program (ACP), il est toujours utilisé par les compagnies aériennes mais pas que, d'autres entreprises (à grands systèmes avec des exigences de traitement des transactions à grande vitesse et à volume élevé) l'utilisent également. z/TPF peut être utilisé pour gérer régulièrement des dizaines de milliers de transactions par seconde, tout en bénéficiant d'une disponibilité ininterrompue mesurée en années.

iv- Les spécificités du mainframe

Ce qui fait du mainframe sa notoriété sont entre autres sa fiabilité, sa disponibilité, sa facilité d'entretien, sa sécurité, son évolutivité, sa compatibilité continue.

- ✚ Fiabilité : Les composants matériels du système ont de nombreuses capacités d'autocontrôle et d'auto-récupération. La fiabilité du logiciel du système est le résultat de tests approfondis et de la capacité à faire des mises à jour rapides pour les problèmes détectés.
- ✚ Disponibilité : Le système peut récupérer d'un composant défaillant sans impacter le reste système en cours d'exécution. Ceci s'applique à la récupération matérielle (le remplacement automatique des éléments défectueux avec des pièces de rechange) et la récupération de logiciels (les couches de récupération d'erreur fournis par le système d'exploitation).
- ✚ Facilité d'Entretien : Le système peut déterminer pourquoi une panne s'est produite. Cela permet le remplacement des éléments matériels et logiciels tout en impactant le moins possible le système opérationnel. Ce terme implique également des unités de remplacement bien définies, soit matériel ou logiciel.

- ✚ Sécurité : Un système informatique sécurisé empêche les utilisateurs d'accéder ou de modifier la moindre donnée sur le système y compris les données utilisateur, sauf via l'interface qui applique les règles d'autorité. Le Z14 peut fournir un système sécurisé pour traiter un grand nombre d'applications hétérogènes qui gardent des données sensibles, ceci est possible grâce à son serveur Resource Access Control Facility (RACF) et sa fonction d'autorisation du système (SAF).

RACF : il protège les ressources en accordant l'accès uniquement aux utilisateurs autorisés d'accéder aux ressources protégées. RACF conserve les informations sur les utilisateurs, les ressources et les autorités d'accès spécialement des structures ayant un profil de base de données, et fait référence à ces profils pour décider quels utilisateurs sont autorisés à accéder aux données sensibles du système. En résumé les fonctions de RACF sont : identifier et authentifier les utilisateurs, autoriser les utilisateurs à accéder aux ressources protégées, enregistrer et signaler diverses tentatives d'accès non autorisé, contrôler les moyens d'accès aux ressources.

SAF : ce système fait partie du système d'exploitation z/OS et fournit les interfaces aux services disponibles pour effectuer l'authentification, l'autorisation et l'identification. Le routeur SAF fournit un point focal commun pour tous les produits fournissant un contrôle des ressources. Ce point focal encourage l'utilisation d'un contrôle commun des fonctions partagées entre les produits et les systèmes. La gestion des ressources sous-système appellent le routeur z/OS dans le cadre de certaines fonctions décisionnelles, telles que la vérification du contrôle d'accès et contrôle lié aux autorisations ; Ces points sont appelés points de contrôle. SAF dirige conditionnellement les commandes de contrôle vers RACF (si celui-ci est présent), ou à un sous-programme de traitement fourni par l'utilisateur, ou aux deux.

- ✚ Évolutivité : Il s'agit de la capacité du matériel, du logiciel ou un système de continuer à fonctionner correctement lorsqu'il change de taille ou de volume ; par exemple la capacité de conserver les niveaux de performances lors de l'ajout de processeurs, de mémoire et de stockage. Un système évolutif peut s'adapter efficacement au travail avec des grands ou petits réseaux exécutant des tâches de complexité variable. Les mainframes présentent des caractéristiques d'évolutivité à la fois matérielles et logicielles, avec la possibilité d'exécuter plusieurs copies du logiciel du système d'exploitation en une seule entité appelée système complexe ou sysplex.

- ✚ Compatibilité continue : Capacité du système à pouvoir à la fois exécuter un logiciel exigeant de nouvelles instructions matérielles et lancer un logiciel plus ancien exigeant des instructions matérielles plus anciennes. Cependant la compatibilité absolue à travers des décennies de changements et d'améliorations n'est pas possible, raison pour laquelle lorsqu'une incompatibilité est inévitable, la maison IBM prévient les utilisateurs au moins un an en avance des modifications logicielles nécessaires.

2- Les activités

Après imprégnation, à défaut de pouvoir travailler sur les mainframes du CENADI, (ceux-ci étant d'une sensibilité extrême), il nous a été proposé de travailler sur des mainframes distants à partitions virtuelles sur la plateforme « master the mainframe » d'une part et d'assister au POC tenu par le garage IBM de Montpellier d'autre part.

a- Les séances « master the mainframe »

« Master The Mainframe » est une plateforme IBM permettant aux apprentis de s'exercer de manière ludique sur les mainframes sans nécessité d'avoir des connaissances sur le domaine au préalable. Une fois inscrits, les utilisateurs sont redirigés vers un autre onglet qui contient 3 niveaux d'apprentissage : *Level 1*, *Level 2* et *Level 3*. Dans chaque niveau d'apprentissage les utilisateurs gagnent des badges numériques après avoir passé les exercices du niveau en question. Ces badges numériques étant la preuve du niveau de compréhension d'un utilisateur, peuvent servir à trouver des opportunités d'emploi. Pour les Étudiants spécialement, les séances de travail master the mainframe permettent d'acquérir les compétences uniques requises pour les systèmes d'entreprise, apprendre à coder et à créer des innovations à l'aide du mainframe, travailler sur la mise en réseau, les analyses, les bases de données, les langages de programmation approfondie et les API.

En ce qui nous concerne, nous n'avons pas pu atteindre le *Level 3* du fait des problèmes de connexion à distance sur notre partition mainframe. Malgré cela nous avons terminé le *Level 1* qui comportait quatre (04) exercices pour l'obtention de trois (03) badges. Ces exercices avaient pour but :

- ✚ d'introduire l'apprenant à la notion de LPar, qui a été définie comme des partitions logiques natives du mainframe ;
- ✚ la connexion à un mainframe distant, l'outil utilisé est Visual Studio Code où on y installe les extensions Zowe Explorer version 1.11.1, IBM Z Open Editor version 1.1.0, indent-rainbow version 7.4.0.
- ✚ La création des Dataset et leur modification.

Le niveau 2 quant à lui comportait onze (11) exercices et nous n'avons pu faire que deux (02) sur les onze et obtenir qu'un seul badge. Ces exercices consistaient à créer et régler les *High Level Qualifier (HLQ) filter*

Malgré l'intervention d'un membre du personnel familial à la plateforme, nous n'avons pu régler notre soucis d'accès au mainframe distant, et nous ne pouvons jusqu'à ce jour émettre la raison de cet échec de connexion.

b- Les Séances de POC

Les séances de POC entre le personnel du CENADI et le garage IBM de Montpellier se sont passées par visioconférence qui portait sur RedHat Openshift : sa présentation générale, la création d'une image et son déploiement sur un serveur IBM. Afin d'asseoir les notions apprises, six (6) lab étaient à reproduire sur les consoles web d'Openshift de Montpellier.

Après connexion au serveur de Montpellier via un VPN et création d'un compte, nous avons six tutoriels à reproduire, les activités se résument en ce tableau.

N°Lab	DÉSIGNATION	ACTIVITÉS
1	Exploring the Openshift GUI	<ul style="list-style-type: none"> ✚ Les fonctions Administrateur ; ✚ Les fonctions Développeur ; ✚ Création d'un projet ; ✚ Déploiement d'un conteneur d'image.
2	Using the Openshift Command Line (oc)	✚ Connexion à l'invite de commande d'Openshift ;

		<ul style="list-style-type: none"> ✚ Commandes fréquentes du CLI ; ✚ Déploiement d'un conteneur d'image depuis l'oc ; ✚ Commande d'administration CLI.
3	Deploying an Application from Source Code	Déploiement d'une application JavaScript.
4	Monitoring, Metrics and Alerting with Openshift	Monitoring des workloads Kubernetes avec les outils Grafana, Alertmanager et Prometheus,
5	Using Persistent Storage	Déploiement de MongoDB comme conteneur d'image, ici il s'agit de conteneuriser la base de données de l'application déployée au 4 ^{ème} Lab.

Tableau 7- Activités du POC

Ne disposant pas d'une longue durée d'accès au serveur de Montpellier, nous n'avons pas pu comprendre réellement les enjeux du dernier lab qui portait sur le déploiement d'une base de données comme image.

CONCLUSION

En somme, de ce chapitre nous retenons que le CENADI, data centre de l'État, nous a offert un cadre de travail idéal et un personnel attentionné pour nous aider à la réalisation de notre projet de stage. Durant ce stage, nous avons appris les caractéristiques d'un mainframe et en quoi celles-ci consistent, nous avons pu travailler sur des aspects techniques de la manipulation d'un mainframe même si cela fut court, et nous avons participé à des échanges portant sur Openshift, technologie que le CENADI souhaite implémenter. À présent, il est important de comprendre la problématique de la structure qui l'a poussée à nous donner ce thème.

CHAPITRE 2 : LE PROBLÈME POSÉ

INTRODUCTION

Le CENADI, gérant national des données et applications sensibles de l'État, est confronté à une exigence de haute disponibilité et continuité de service pour les systèmes qu'il héberge ; c'est pour cette raison qu'il s'est muni de super calculateurs onéreux à la pointe de la technologie. Cependant, malgré l'achat de ces infrastructures puissantes, il n'arrive pas à rentabiliser au même titre que les dépenses du fait de l'utilisation rigide des ressources de leurs infrastructures. C'est donc dans l'optique de combler ce problème qu'il se tourne vers des solutions en vogue telles que le cloud privé et la conteneurisation. *En quoi ces technologies représentent-elles une solution face aux problèmes que rencontre le CENADI ?* Répondre à cette question revient à faire une étude du problème posé puis une étude des différentes solutions disponibles pour pallier à cette difficulté.

I- ÉTUDE DU PROBLÈME

Dans cette partie nous aborderons les difficultés que rencontrent le CENADI et l'étude de la solution qui palliera à ces difficultés.

1- Les problèmes rencontrés

Les problèmes rencontrés par le CENADI lors de l'exercice de leurs fonctions sur leurs architectures sont :

✚ **La rigidité matérielle** : Les mainframes viennent avec des partitions logiques natives (Lpar), dans le cas du Z14 du CENADI ils sont au nombre de 04 extensible. Lorsqu'on alloue des ressources à une partition, il est impossible que ces ressources puissent être utilisées par d'autres partitions, à moins de supprimer la partition pour avoir de nouveau les ressources disponibles ;

✚ **Le gaspillage de ressources** : Ce problème découle naturellement de la rigidité matérielle, les ressources ne pouvant pas être modifiées, certaines partitions se retrouvent avec des ressources énormes pour des petites applications et d'autres avec des applications demandant plus de ressources que celles disponibles. Dans le cas du

CENADI par exemple, l'application SYAMPE demande constamment d'énormes ressources pourtant ANTILOPE n'utilise ses ressources que les 17,18 et 19 du mois, le reste du mois la partition hébergeant l'application se retrouve avec des ressources inutilisées et inutilisables.

✚ **La continuité de service** : Ce problème fait référence au maintien de l'uptime à 99,999 %, les applications doivent demeurer disponible 24h/24 peu importe les pannes, les incidents ou les maintenances.

✚ **La portabilité et La nécessité d'hébergement des applications.**

On distingue ces problèmes sur différents plans :

- Il survient lors des mises à jour, les applications souffrent de lenteur conséquente pour s'arrimer aux modifications faites par les développeurs et les administrateurs ;
- Les effets de bords, lorsque les applications sont déployées hors de leur environnement natif, en l'occurrence au CENADI lorsque des applications x86 sont déployées dans le système ZOS ou Z/Linux.

2- Étude de la solution

L'étude consistera à définir les termes de cloud privé et de conteneurisation.

a- Cloud privé

Le Cloud privé ou Cloud d'entreprise fournit en partie les mêmes services qu'un Cloud public à la seule différence que l'infrastructure et le réseau peuvent être internes à l'entreprise. On en distingue trois sous-types : Cloud privé géré, dédié et communautaire. Dans le premier type l'infrastructure informatique est gérée par le fournisseur du matériel, ce style d'hébergement est bénéfique pour les sociétés qui ont juste besoin d'un Cloud privé mais ne peuvent pas s'occuper de la maintenance de la machine. Le deuxième type concerne un Cloud implémenté dans un autre Cloud. Et enfin le dernier type est utile pour relier plusieurs entreprises dans un

réseau interne; en plus des avantages du Cloud computing en général (redimensionnement rapide, un accès large bande), le cloud privé offre un contrôle complet des données de l'entreprise et une sécurisation plus renforcée.

Le Cloud privé offre également les mêmes services qu'un Cloud public à savoir : l'Infrastructure As A Service, le Software As A Service, la Platform As A Service.

- ✚ L'infrastructure as a service (IaaS) : offre des services de location des moyens de calcul et de stockage, des capacités réseau. Les clients gèrent le middleware des serveurs, et surtout les logiciels applicatifs ; l'entreprise qui détient son Cloud gère le matériel serveur, les couches de virtualisation, le stockage et les réseaux. En cas de panne ou d'un quelconque dysfonctionnement, c'est l'entreprise qui se charge des réparations. Le CENADI est en processus de création de leur IaaS et pour cela utilise Openstack.
- ✚ Le software as a service (SaaS) : dans ce cas de figure, l'entreprise détenant son Cloud héberge des applications qu'elle rend disponible à sa clientèle via internet. *Dans le cas du CENADI, l'application ANTILOPE qui gère la solde de l'État est un SaaS.*
- ✚ Platform as a service (PaaS) : il est le type de service qui sera utilisé pour la réalisation de notre projet. Les clients doivent se charger eux même de la gestion de l'application exécutée sur la plateforme, ainsi que des données dont l'application dépend. D'abord destiné aux développeurs et aux programmeurs, PaaS offre à l'utilisateur une plateforme Cloud partagée pour le développeur et l'administrateur (un point important de DevOps). Dans le cadre de notre thème nous allons nous mettre d'un point de vue administrateur uniquement.

Le schéma 2 présente le(s) niveau(x) de gestion d'un hébergeur et d'un client.

Sens de lecture

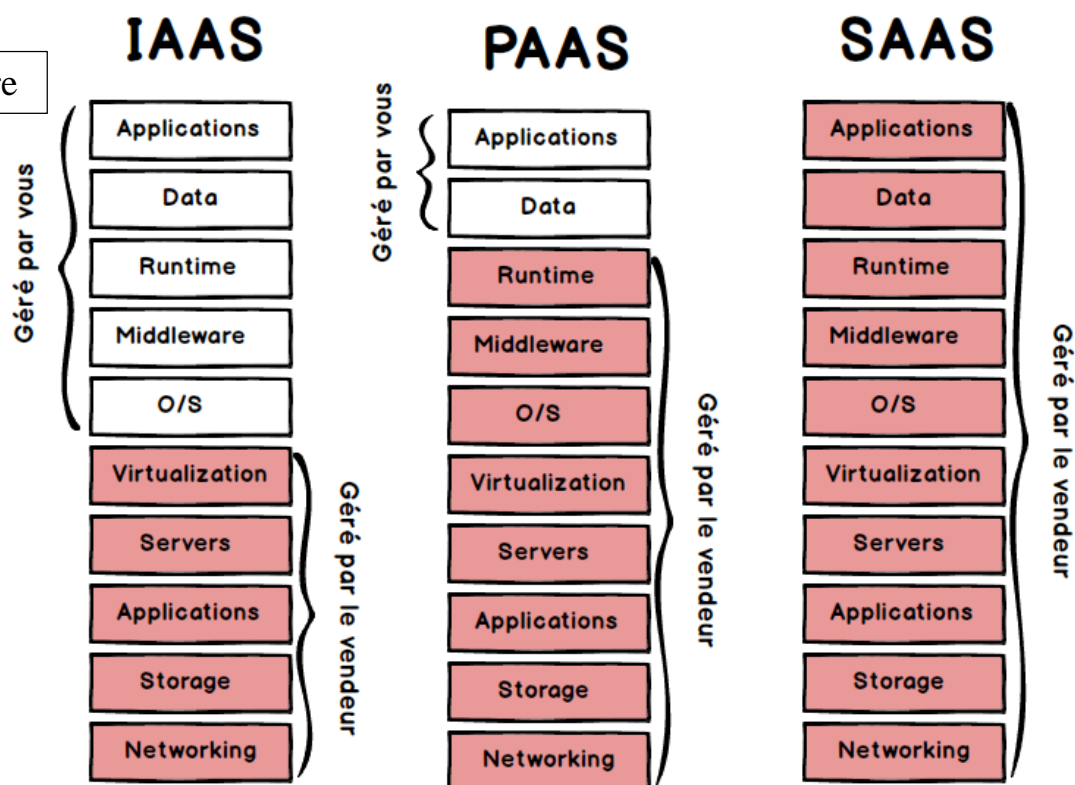


Schéma 2- Différences entre les services

Le Cloud privé se détachant le plus de la couche matérielle, comble le problème de la rigidité des infrastructures cité plus haut.

b- Conteneurisation

La Conteneurisation créer plusieurs instances utilisateurs isolées sur un même noyau, ces instances sont appelées conteneur. Elle favorise le développement et la livraison des applications en continu; les conteneurs n'ayant pas besoin d'hyperviseur de type 2 pour être déployés, ceux-ci comblent le problème de compatibilité rencontré dans la virtualisation classique. Les avantages de la conteneurisation sont entre autres :

- ✚ **Légèreté** : les conteneurs occupent moins d'espace que les machines virtuelles sur le serveur ;
- ✚ **Élasticité** : pas besoin d'allouer une quantité donnée des ressources, les conteneurs font une utilisation dynamique des ressources du serveur. Lorsque la demande sur un

conteneur diminue, les ressources supplémentaires sont libérées pour être utilisées par d'autres conteneurs ;

✚ **Densité** : les conteneurs n'ont pas besoin d'héberger leur propre système d'exploitation et de ce fait se crée des environnements denses où y sont des ressources du serveur hôte pleinement utilisées ;

✚ **Maintenance plus facile** : avec un seul noyau de système d'exploitation, les mises à jour ou les correctifs au niveau du système d'exploitation ne doivent être effectuées qu'une seule fois pour que les modifications prennent effet dans tous les conteneurs, cela rend la maintenance plus efficace.

Les outils utilisés pour conteneuriser sont **Docker** et **Kubernetes**.

Docker : Docker est une plateforme libre sous licence Apache 2 basée sur la technologie de virtualisation en conteneurs logiciels, dédiée principalement aux administrateurs et aux développeurs pour le développement, les tests, la livraison et l'exécution d'applications distribuées. Avec Docker les conteneurs partagent le même système d'exploitation contrairement à la para-virtualisation où chaque machine virtuelle a son système d'exploitation, ce qui peut occasionner plus tard les problèmes de portabilité. Les conteneurs permettent de faire tourner des environnements isolés les uns des autres mais tournant dans un même noyau, Docker virtualise seulement l'environnement d'exécution (Processeur, mémoire vive ou système de fichiers) contrairement à la virtualisation classique qui virtualise tout la machine ce qui rend les machines virtuelles plus lourdes que les conteneurs ; ainsi pour une même quantité de ressources on peut créer plus de conteneurs que de machines virtuelles.

Le Schéma 3 présente les différences entre les conteneurs et les machines virtuelles.

Kubernetes : Kubernetes est un orchestrateur de conteneurs open source développé tout d'abord par Google en 2014, la première version a été réalisée en juillet 2015. Kubernetes s'est inspiré d'une autre technologie de Google, Borg, qui permet d'exécuter plus de 7000 conteneurs par seconde. Pour faire simple, une fois que les conteneurs ont été créés en utilisant Docker, la gestion de ces conteneurs demandaient qu'on s'occupe d'une machine à la fois par exemple si nous avions créés mille (1000) conteneurs, la gestion se ferait d'un conteneur à un autre mille fois raison pour laquelle l'on utilise maintenant Kubernetes pour pouvoir manager ces conteneurs en une fois.

Le schéma 4 présente le fonctionnement de Kubernetes

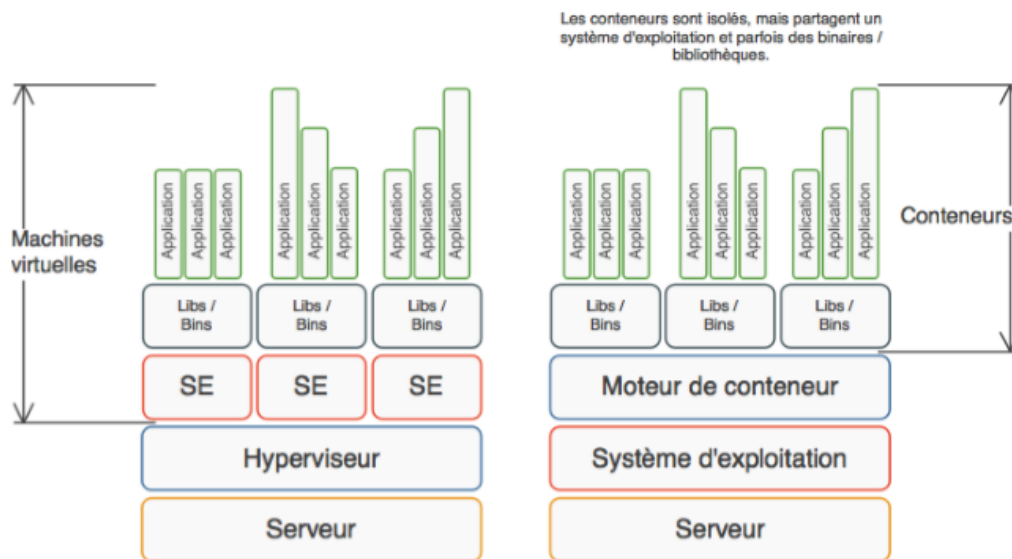


Schéma 3- Différences entre Conteneurs et Machines Virtuelles

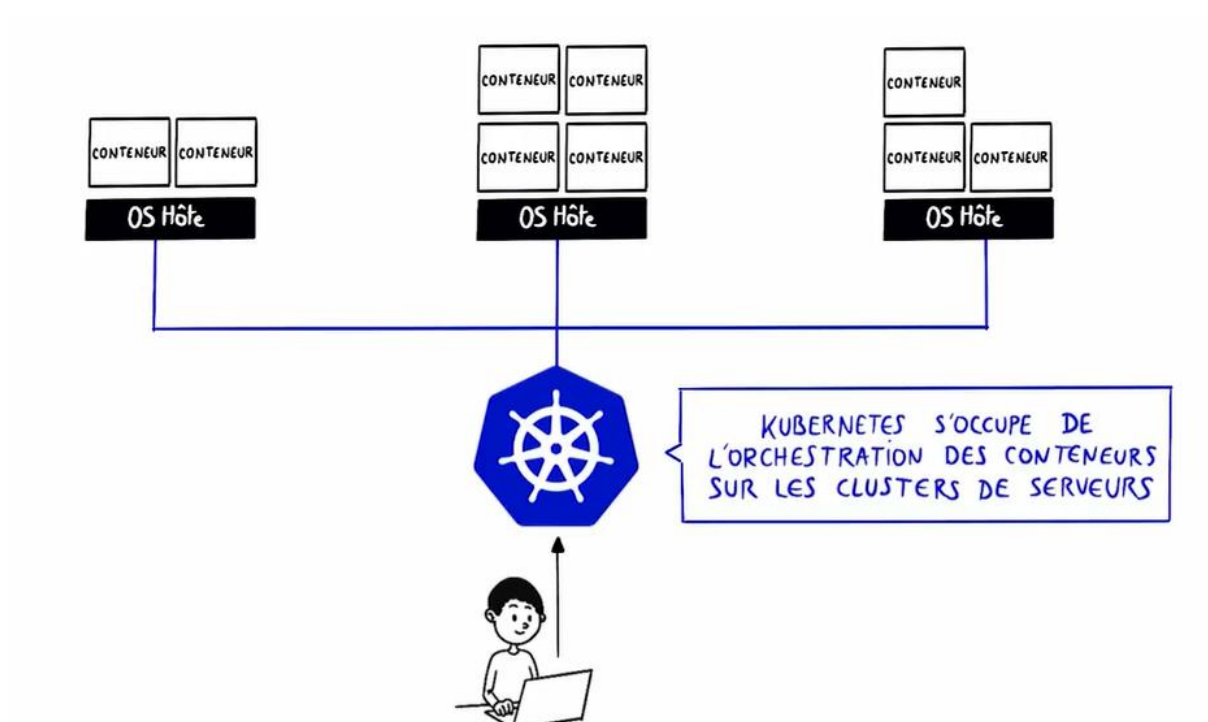


Schéma 4- Fonctionnement de Kubernetes

La conteneurisation, se détachant le plus du système d'exploitation et allouant dynamiquement les ressources, résout les problèmes de compatibilité et de gaspillage de ressources. Elle permet

également l'exécution rapide des lignes de code d'un programme, la possibilité de créer plusieurs instances d'un conteneur favorise la continuité de service car en cas de panne d'un conteneur, l'une des instances prend directement le relai.

Pour réaliser le projet, nous avons besoin d'un outil PaaS qui nous permettra d'avoir à la fois un espace Cloud privé et de conteneurisation. Il existe plusieurs outils sur le marché pour faire cette tâche.

II- LES OUTILS PRÉSENTS SUR LE MARCHÉ

Les solutions PaaS sont généralement des plateformes qui regroupent plusieurs outils de Cloud computing et de Conteneurisation, ils peuvent être assimilables à des boîtes à outils pour implémenter des PaaS. Les plus connus sont :

Foundry

Cloud Foundry est un PaaS open source qui permet de créer, de déployer et de faire évoluer des applications sur des modèles Cloud public et Cloud privé. Foundry a été créé par VMware mais appartient désormais à Pivotal Software. Son support de langage comprend entre autres Python, Ruby, PHP, Java, et Go.

Le schéma 5 présente l'architecture de Foundry.

WSO2

Solution open source offrant des solutions modulaires qui s'intègrent facilement à d'autres composants existants pour fonctionner les uns avec les autres. Il appartient à l'un des fondateurs du protocole Simple Object Access Protocol (SOAP) du serveur Apache. Il est utilisé la plus part du temps par les entreprises devant gérer de nombreuses API internes et externes. WSO2 a justement intégré son API Manager, capable de gérer l'intégralité du cycle de vie de l'API.

Le schéma 6 présente l'architecture de WSO2.

Cloudify

L'une des spécificités de Cloudify est la modélisation de plans basée sur TOSCA ; cette innovation permet aux développeurs d'utiliser YAML pour créer des plans des

topologies d'une application. YAML permet d'encoder des données structurées de façon à la fois lisible par l'homme et analysable sans ambiguïté par une machine.

Le schéma 7 présente l'architecture de Cloudify.

Openshift Container Platform (OCP)

Cette plateforme est particulièrement adaptée pour les déploiements de grande envergure. Il est propriétaire mais a également une version open source qu'est minishift plus adapté pour les débutants qui veulent s'exercer à la conteneurisation sur Openshift.

L'image 1 présente la console web d'OCP.

Tsuru

Propriété de Rede Globo, deuxième plus grand réseau de télévision commerciale au monde, cette solution est Open Source et prend en charge un grand nombre de langages de programmation disponibles ce qui donne plus de liberté aux développeurs.

L'image 2 présente la console web de Tsuru.

Stackato

C'est une solution paas basé sur Cloud Foundry.

Le schéma 8 présente l'architecture de Stackato.

Alibaba

Cloud Alibaba a conquis 50% du marché chinois du Cloud public et commence à s'ouvrir à des marchés extérieurs à la chine.

En image 3 nous avons la console web d'Alibaba

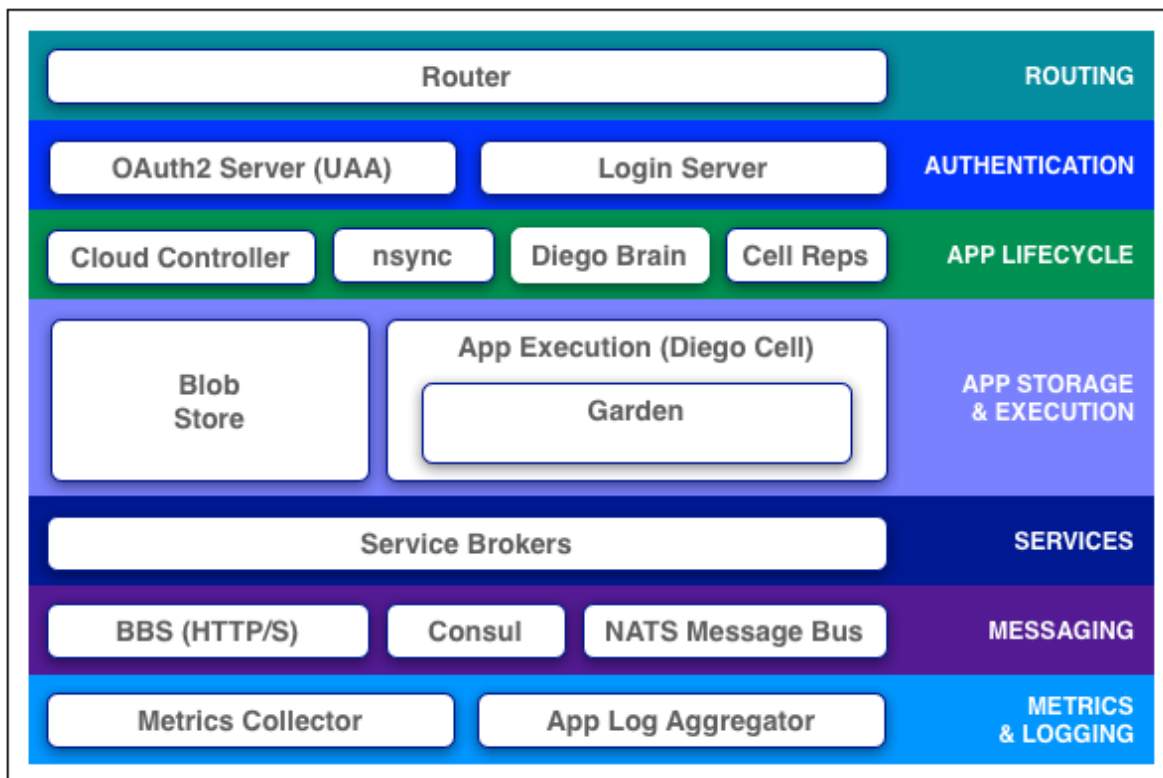


Schéma 5- Architecture Foundry

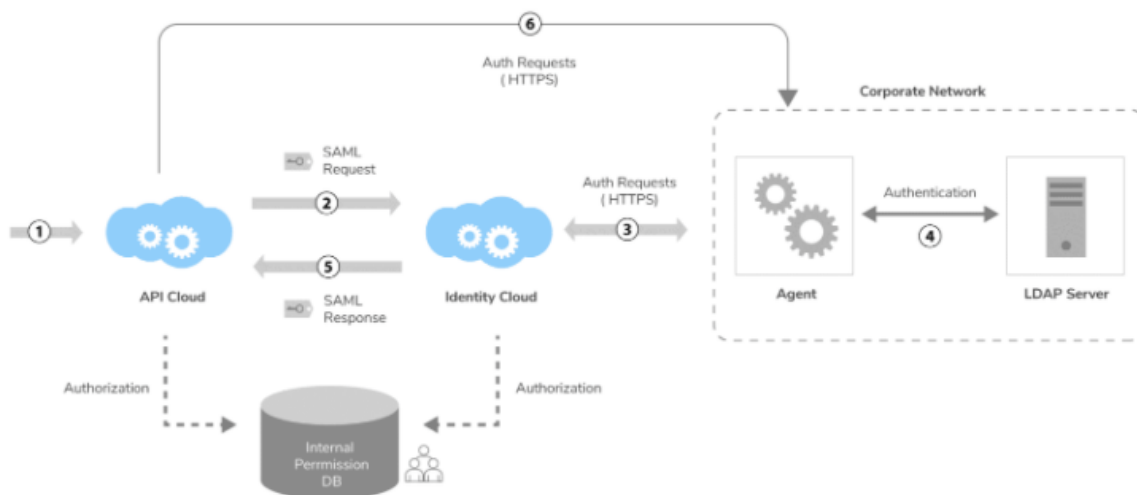


Schéma 6- Architecture WSO2

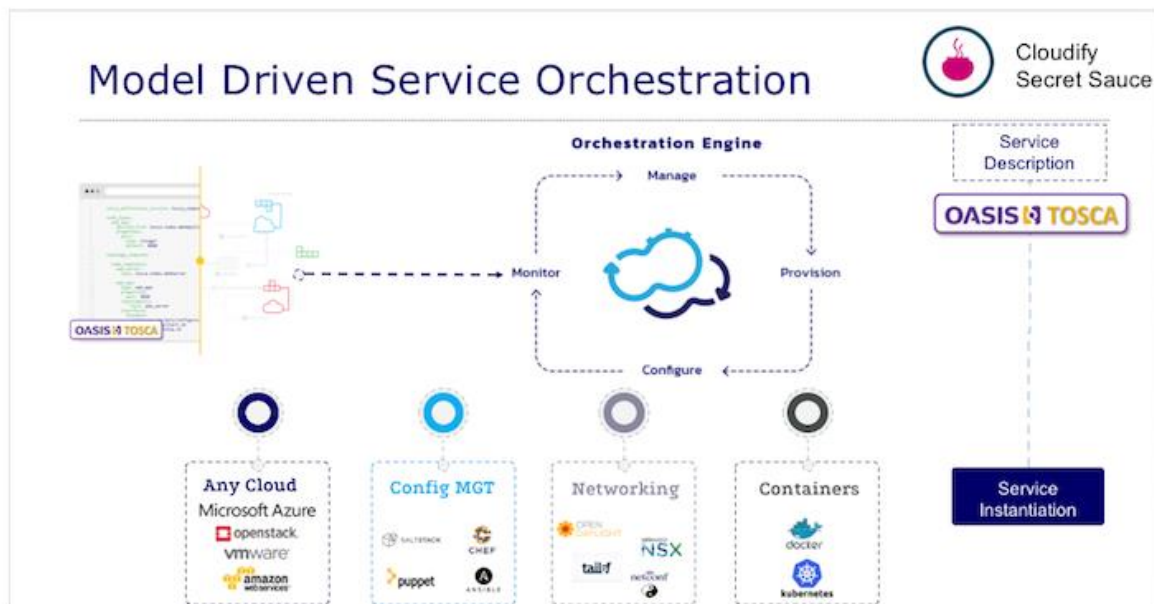


Schéma 7- Architecture Cloudify

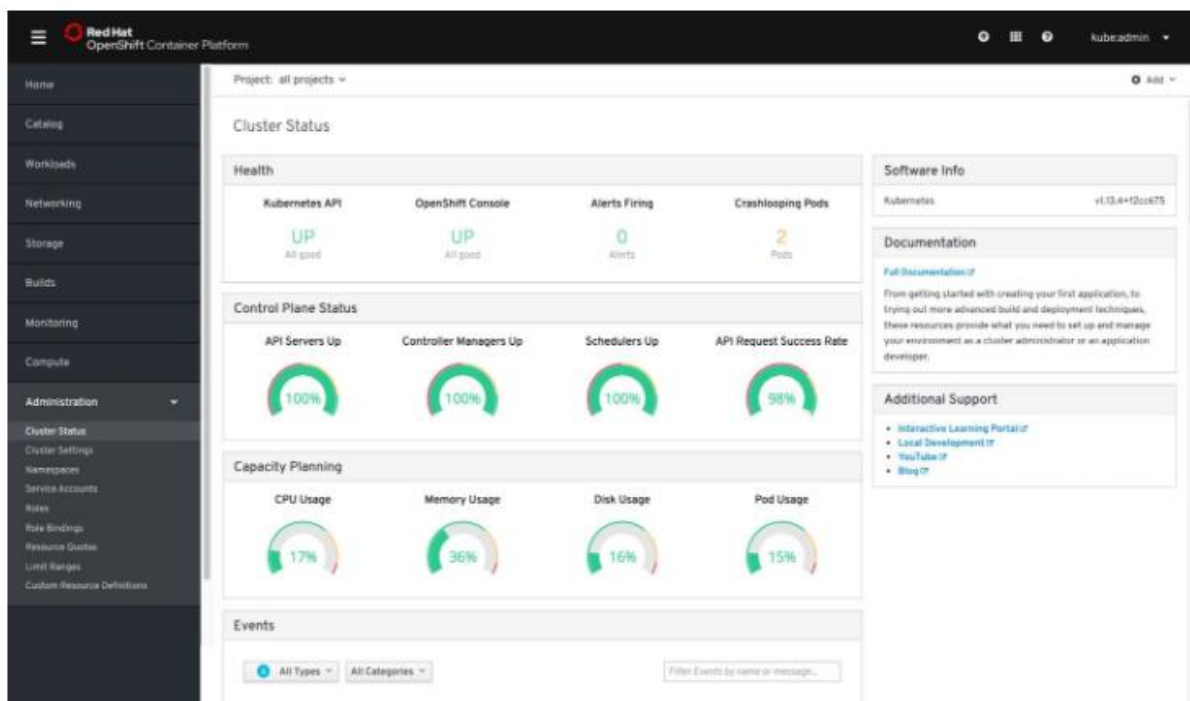


Image 1- Console Web OCP

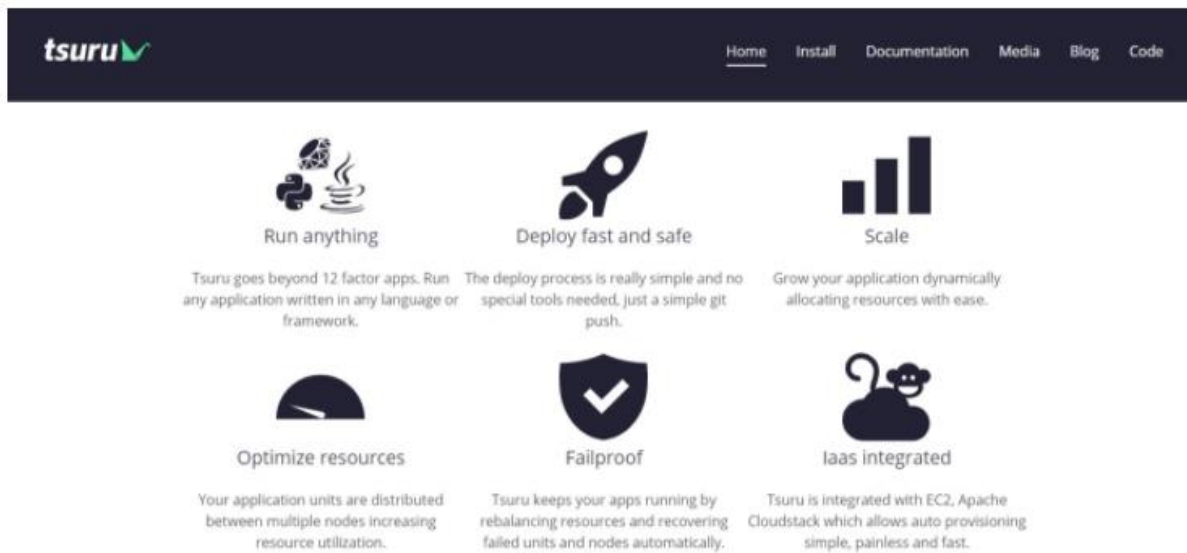


Image 2- Console web Tsuru

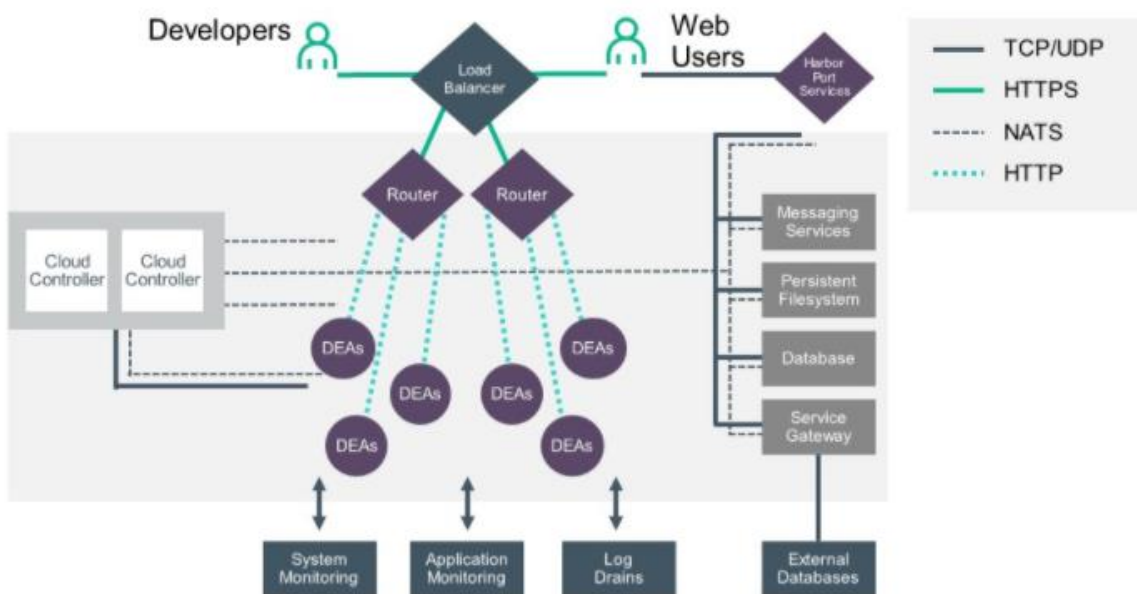


Schéma 8- Architecture Stackato

CHAPITRE 3 : MÉTHODOLOGIE DE RÉOLUTION DU PROBLÈME

INTRODUCTION

Une fois les difficultés rencontrées par le CENADI citées, nous introduisons la concrétisation de ces quasis six (06) mois d'apprentissage. De toutes les solutions énumérées au chapitre 2, nous avons opté pour Openshift Container Platform (OCP) qui tourne sur un système Linux RedHat. Dans ce chapitre, il sera question de vous présenter ce que c'est OCP, pourquoi OCP, les étapes de l'implémentation de la solution et enfin le résultat.

I- LA SOLUTION ADOPTÉE

Le CENADI voulant utiliser le Z14 pour leur solution PAAS, a souhaité que notre choix d'outil soit similaire à ce qu'il aurait choisi pour implémenter la solution ; de ce fait les raisons qui nous ont poussées à adopter OCP sont :

- ✚ Openshift est l'un des leaders sur le marché ;
- ✚ Openshift permet d'utiliser à la fois les piliers Docker et Kubernetes ;
- ✚ Openshift offre des formations en ligne et des exercices d'impregnation ;
- ✚ Le Z14 utilise Linux comme système d'exploitation dans certaines partitions ;
- ✚ Le CENADI a un partenariat avec CFAO qui utilise RedHat.

1- Présentation d'Openshift

RedHat Openshift Container Platform est une plateforme d'applications conteneurisées qui permet aux entreprises d'accélérer et de simplifier le développement, la distribution des applications dans un Cloud public comme privé. Il existe deux (02) manières de déployer une application sur Openshift : par CLI (Command Line Interface) ou par Console Web, la deuxième manière est la plus avantageuse des deux lors de réalisation de certaines tâches. Peu importe la manière de déployer, les capacités d'Openshift sont les mêmes. D'un point de vue développeur, Openshift permet « d'industrialiser » la conception des applications, c'est-à-dire permet de développer plus rapidement et plus facilement les applications. D'un point de vue Administrateur, Openshift facilite la maintenance de la disponibilité et la mise à l'échelle d'un nouvel hôte. Openshift aidant à la fois le développeur et l'administrateur, favorise une

communication entre les deux métiers qui est un point important dans la nouvelle méthodologie de conception d'applications.

Cas d'un Développeur :

Les rôles d'un développeur sont : écrire des applications, créer des modifications, faire des tests et déployer ces applications dans des clusters ; sur la plateforme un développeur doit créer un projet et une application. Une fois que les applications sont créées, le code source sera poussé vers un Référentiel tel que GitHub ; Openshift va créer un travail Jenkins qui va aider au déploiement de cette application notamment en créant une image Docker hors du code source et en mettant cette image dans un registre privé intégré à Openshift. Une fois l'image intégrée dans un registre, Openshift va pousser tout cela dans un cluster (un cluster peut être constitué d'un ou plusieurs hôtes) de façon à ce qu'il y ait autant d'images que les hôtes présents dans le cluster. Lorsque le développeur fait des modifications dans le code source, Jenkins va démarrer le processus de création d'image en intégrant les modifications faites par le développeur, l'encapsule dans un registre, Openshift change l'ancienne version dans le cluster pour la nouvelle le tout sans aucune latence. Nous remarquons donc que la tâche du développeur est simplifiée et rend le développement plus rapide.

Cas d'un Administrateur :

L'administrateur a pour rôle de maintenir un niveau élevé de disponibilité, de vérifier si l'architecture de l'application est saine, pour cela il utilise la console web d'Openshift pour s'informer si l'application a atteint le niveau de disponibilité souhaité. Openshift aide également un administrateur lors de la mise à échelle d'un nouvel hôte ; pour cela Openshift utilise Ansible Playbooks qui permet d'automatiser les tâches citées plus haut particulièrement l'augmentation du nombre d'hôtes en l'amenant dans le même cluster que les autres hôtes. De plus Openshift offre aux développeurs des templates qui sont des environnements applicatifs standardisés, et ces templates contribuent à la stabilité du système ce qui facilite la tâche à l'Administrateur.

La figure 1 présente la structure de la plateforme.

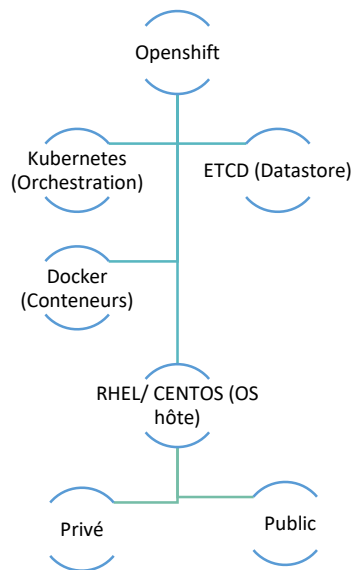


Figure 1- Structure OCP

La couche la plus basse représente le type de Cloud sur lequel sera déployée la plateforme. Dans notre cas il s'agira d'un Cloud privé.

La couche juste au-dessus de la précédente représente le système d'exploitation qui sera installé soit RedHat Enterprise Linux (RHEL) soit CENTOS.

La couche un cran plus haut illustre Docker le créateur de Conteneur, notons que Docker a été récemment ajouté sur la version 3 d'OCP et permet d'implémenter l'isolation des applications.

La couche juste au-dessus de la précédente représente Kubernetes, qui permet d'orchestrer l'ensemble des conteneurs déployés dans Openshift, et ETCD qui est un composant stockant les fichiers de configuration du cluster d'Openshift.

Enfin Openshift est dans la couche la plus élevée de la plateforme, RedHat a ajouté à ce niveau des couches techniques supplémentaires propices à leur offre de PaaS telles que : les configurations spécifiques OCP, les services infrastructures (*nous avons par exemple le service réseau Software Defined Network, SDN, utilisé pour un cloisonnement fort entre conteneurs, par exemple un conteneur d'un pod A ne peut communiquer directement avec un conteneur d'un pod B*), les Runtimes ou Environnements d'exécution, les services UX (fournit des interfaces d'utilisateur et de management) et les services DevOps.

Openshift possède une architecture orientée microservices offrant un découplage important de ses services. On parle également d'indépendance des services.

Les caractéristiques d'un microservice sont :

- ✚ Possibilité d'être construit indépendamment ;
- ✚ Posséder ses propres données ;
- ✚ Etre testé indépendamment ;
- ✚ Etre conçu et développé indépendamment.

Ci-dessous l'architecture d'Openshift Container Platform (OCP)

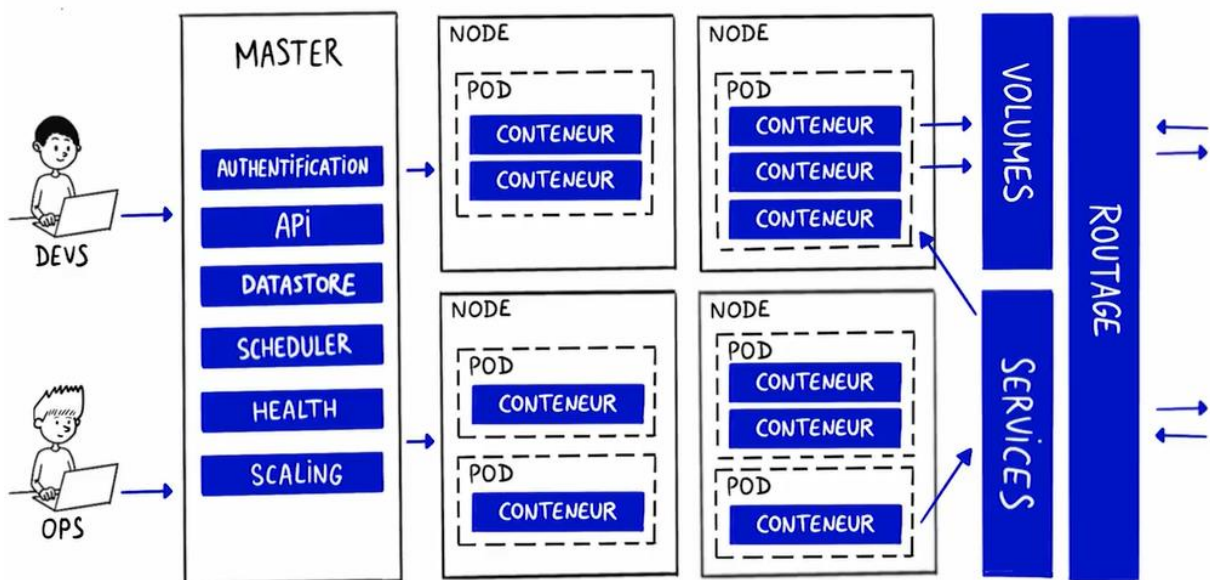


Schéma 9- Architecture OCP

OCP peut s'installer sur machine physique comme sur machine virtuelle ; après installation de la plateforme, OCP met en place des nœuds (nodes), qui sont l'environnement d'exécution d'un ou plusieurs conteneurs, et un Master. Les rôles du master sont :

- ✚ contrôler les nœuds notamment la gestion es nœuds et l'orchestration des Pods ;
- ✚ Authentifier des utilisateurs ;
- ✚ Traiter des requêtes à l'API d'administrateur ;

- + Fournir un registre commun pour les nœuds du cluster et leur permet de partager les données de configurations
- + Recevoir les commandes d'un développeur et d'un administrateur et les relais aux nœuds et le nœud adapté à la tâche alloue les ressources au pod pour qu'il effectue la tâche requise ;
- + Planifier de l'exécution des conteneurs dans le cluster ;
- + Garantir l'intégrité des conteneurs au fil du temps ;
- + Indiquer quel nœud doit tourner en se basant sur la disponibilité ;
- + Gérer l'attribution des ressources, afin de s'assurer que les ressources allouées ne soit pas en excès par rapport à la charge de travail ;
- + Gérer la résilience des pods ;
- + Ordonner à Docker à lancer les conteneurs spécifiés.

Toutes ces fonctionnalités du Master peuvent se résumer en ces services : Authentification, API, Datastore, Scheduler, Health, Scaling.

Les données des conteneurs sont stockées dans le Volume (espace de stockage accessible à tous les conteneurs sur un nœud). Le volume est également nécessaire lorsque des conteneurs ont besoin de se partager des données.

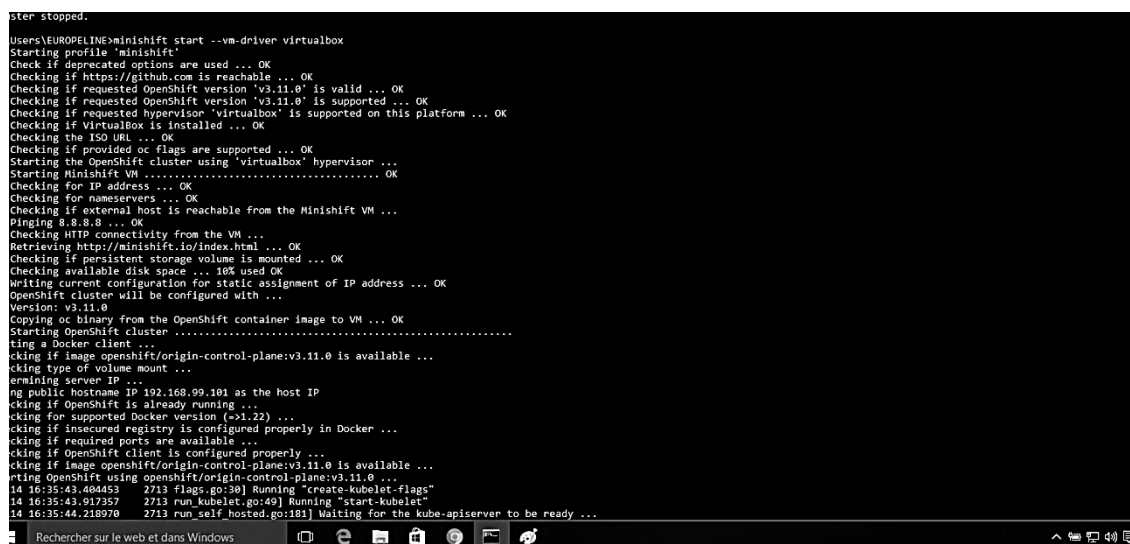
À l'intérieur des nœuds se trouvent les pods qui conservent les conteneurs, un pod possède une adresse IP et des ports réseaux qui lui sont propres et méconnus des autres pods, ainsi les conteneurs d'un même pod peuvent communiquer directement entre eux via le localhost tandis que cela est impossible pour les conteneurs de pods différents. Si des conteneurs de pod différents veulent communiquer, ils doivent passer par le module service. Le service possède sa propre adresse IP et nom DNS, il permet de router les appels d'une application vers l'un de ses pods, le service par défaut est accessible uniquement dans le cluster mais si un client veut accéder au service depuis l'extérieur il devra créer impérativement une route. Une route permet de relier le service d'un pod à une url externe.

2- Étape d'implémentation

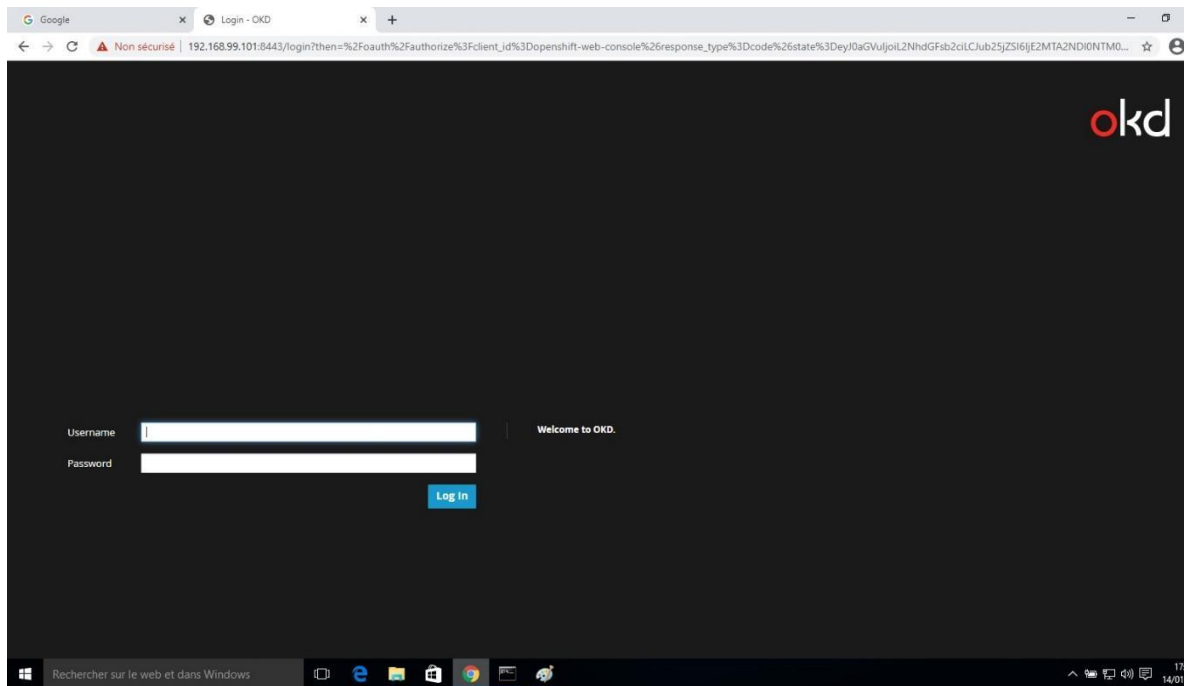
Lors de l'implémentation de notre projet nous n'avons pas pu travailler avec Openshift qui requiert des ressources matérielles que nous n'avons pas (9 Go de RAM), nous avons donc utilisé minishift qui est la version light d'Openshift.

1^{ère} Étape : Installation de minishift sur Windows

- Télécharger minishift sur le site docs.okd.io et le dézipper ;
- Télécharger et installez Virtualbox (uniquement Virtualbox car minishift ne tourne qu'avec lui comme hyperviseur) ;
- Modifier la variable d'environnement Path par la racine du chemin du dossier de minishift ;
- Aller en invite de commande et assurez-vous d'avoir autorisé les options de Virtualisation dans le BIOS ;
- Une fois les fonctions de virtualisation allouées, tapez la commande minishift start --vm-driver virtualbox ;
- Une fois l'installation de minishift réussie, le lien vers la console web de minishift sera donné, copiez le lien et y accédez. Les informations sur votre nom d'utilisateur, votre mot de passe et votre login en tant qu'administrateur vous seront également données.



```
ster stopped.
Users\EUROPELINE>minishift start --vm-driver virtualbox
Starting profile 'minishift'
Checking if deprecated options are used ... OK
Checking if https://github.com is reachable ... OK
Checking if requested OpenShift version 'v3.11.0' is valid ... OK
Checking if requested OpenShift version 'v3.11.0' is supported ... OK
Checking if requested hypervisor 'virtualbox' is supported on this platform ... OK
Checking if VirtualBox is installed ... OK
Checking the ISO URL ... OK
Checking if provided oc flags are supported ... OK
Starting the OpenShift cluster using 'virtualbox' hypervisor ...
Starting Minishift VM ..... OK
Checking for IP address ... OK
Checking for nameservers ... OK
Checking if external host is reachable from the Minishift VM ...
Pinging 8.8.8.8 ... OK
Checking HTTP connectivity from the VM ...
Retrieving http://minishift.io/index.html ... OK
Checking if persistent storage volume is mounted ... OK
Checking available disk space ... 10% used OK
Writing current configuration for static assignment of IP address ... OK
OpenShift cluster will be configured with ...
Version: v3.11.0
Copying oc binary from the OpenShift container image to VM ... OK
Starting OpenShift cluster .....
Installing a Docker client ...
Checking if image openshift/origin-control-plane:v3.11.0 is available ...
Checking type of volume mount ...
Determining server IP ...
Using public hostname IP 192.168.99.101 as the host IP
Checking if OpenShift is already running ...
Checking for supported Docker version (>=1.22) ...
Checking if insecure registry is configured properly in Docker ...
Checking if required ports are available ...
Checking if OpenShift client is configured properly ...
Checking if image openshift/origin-control-plane:v3.11.0 is available ...
Starting OpenShift using openshift/origin-control-plane:v3.11.0 ...
14 16:35:43.480453 2713 flags.go:30] Running "create-kubelet-flags"
14 16:35:43.917257 2713 run_kubelet.go:49] Running "start-kubelet"
14 16:35:44.218970 2713 run_self_hosted.go:181] Waiting for the kube-apiserver to be ready ...
```



```

quickstart" "sample-templates/nodejs quickstart"
[0114 16:39:39.512504 2713 interface.go:41] Finished installing "openshift-web-console-operator" "centos-imagestreams" "sample-templates" "persistent-volumes" "openshift-image-registry"
"ft-router"
Login to server ...
Creating initial project "myproject" ...
Server Information ...
OpenShift server started.

The server is accessible via web console at:
https://192.168.99.101:8443/console

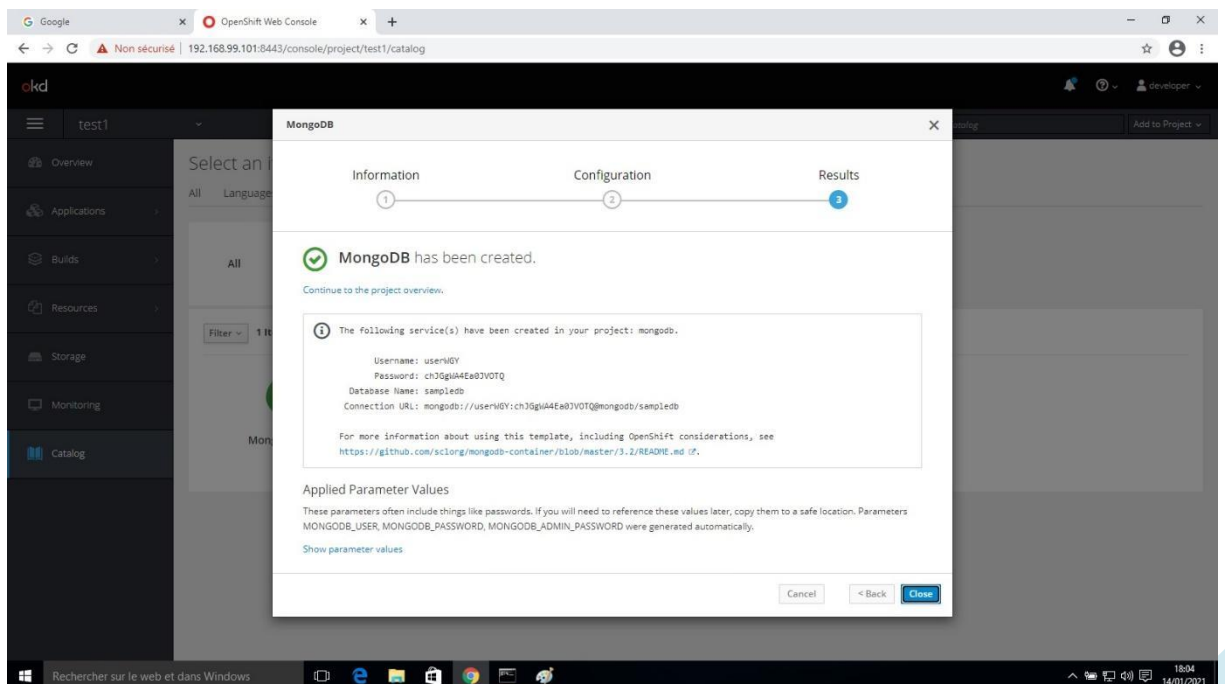
You are logged in as:
User: developer
Password: <any value>

To login as administrator:
oc login -u system:admin

Could not set oc CLI context for 'minishift' profile: Error during setting 'minishift' as active profile: The specified path to the kube config 'C:\Users\EUROPELINE\minishift\machines\mi
beconfig' does not exist

C:\Users\EUROPELINE>https://192.168.99.101:8443/console

```



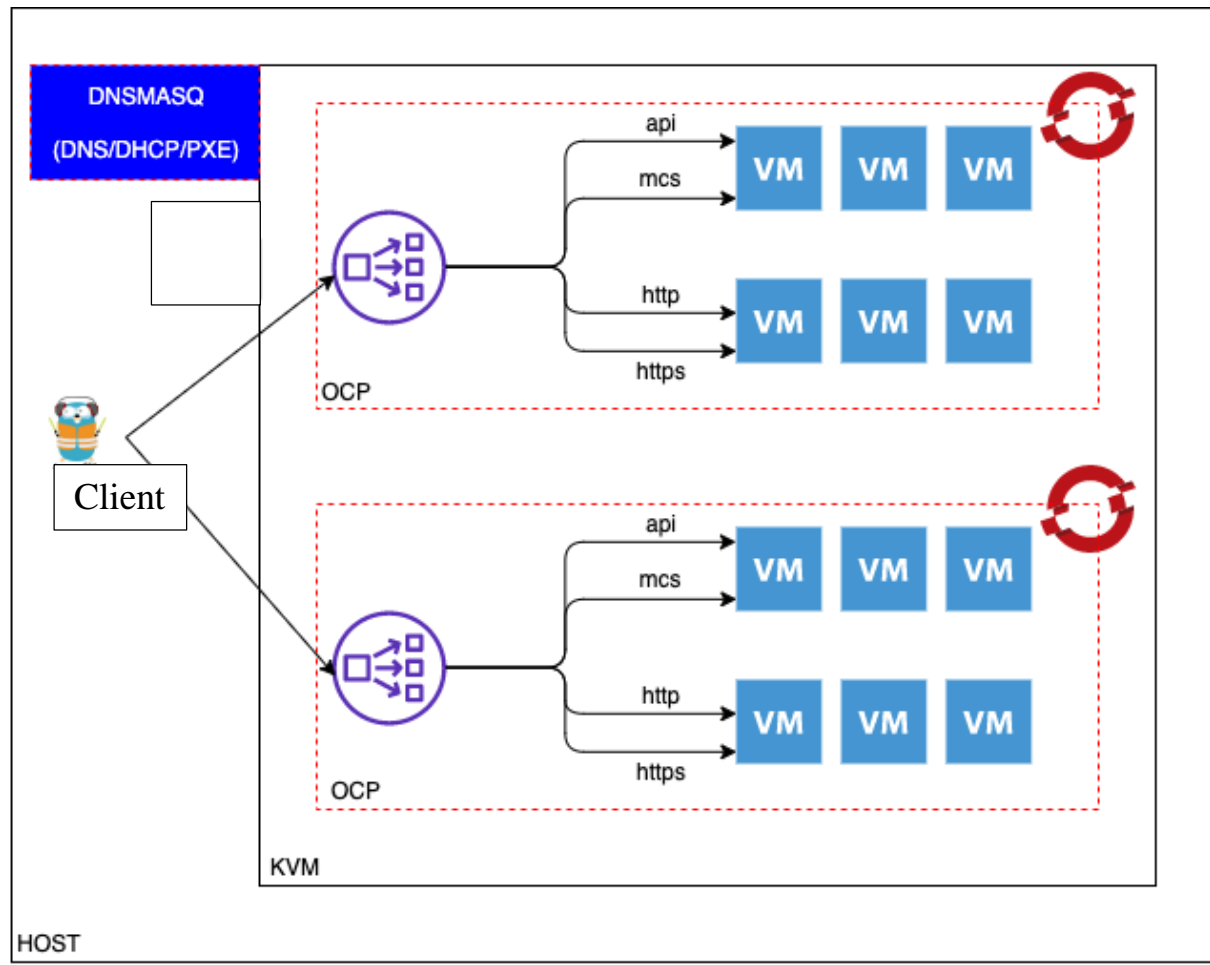
2^{ème} Étape : Générer une Image

Un conteneur est une instance d'une image, donc il nous faut créer une image pour pouvoir avoir des conteneurs. Il existe deux (02) méthodes de déploiement d'images en externe sur Openshift : par GitHub (dans ce cas le docker file est directement généré par Openshift) ou par Docker file. La méthode que nous allons utiliser est GitHub, qui est une plateforme où les développeurs peuvent stocker les codes sources de leurs applications et contrôler les modifications qui lui sont apportées. À travers Source to Image, qui est une spécificité intégrée dans openshift, nous pouvons directement intégrer dans OCP des lignes de codes contenues dans GitHub. La création d'une image par GitHub se fait comme suit :

- + Se mettre en mode développeur ;
- + Cliquer sur *+Add* ;
- + Puis choisir Git ;
- + Donner le lien de votre *repository* sur Github ;
- + Choisir les Options Avancés de Git ;
- + Choisir le langage de votre application pour construire l'image ;
- + Cliquer sur créer ;
- + Donner une route pour votre image.

II- LE RÉSULTAT

Nous obtenons une plateforme où sont stockées des applications sous forme d'images dans des conteneurs. Le développeur comme l'administration peuvent enclencher des modifications sur ces applications via la plateforme d'Openshift.



Architecture de déploiement

CONCLUSION

En somme, après implémentation de la solution PaaS nous nous rendons compte à quel point cette technologie aide les développeurs et les administrateurs au quotidien et favorise le déploiement continu des applications. Openshift qui jouit du soutien des communautés Docker et Kubernetes, ne cesse de se perfectionner et d'actualiser les règles de déploiement suivant les normes DevOps.

CONCLUSION GÉNÉRALE

En définitive, le stage que nous avons effectué avait pour objectifs principaux de nous imprégner au monde professionnel et à mettre en pratique les connaissances individuelles.

Durant ce stage, nous avons pu constater les différents problèmes dont souffre le CENADI au quotidien, ces problèmes peuvent se résumer à l'indisponibilité des services.

Face à ces difficultés, il nous a été proposé d'implémenter une solution PaaS. Notre choix s'est tourné vers Openshift Container Platform, qui comme son nom l'indique est une plateforme permettant de créer des conteneurs qui facilitent le déploiement et la disponibilité des applications.

Notre stage s'est bien déroulé et nous avons beaucoup appris malgré quelques difficultés telles que le manque de temps, notre indisponibilité et la sous information. Toutefois, force est de constater que nous avons apprécié apprendre, échanger, grandir et élargir nos horizons tout au long de ces six (06) mois de stage. Nous sommes satisfaits du travail fourni et de la solution implémentée et nous remercions une fois de plus tous les acteurs qui ont concourus à l'accomplissement de cette expérience professionnelle.

LEXIQUE

CLOUD : accès à des services informatiques via internet ou un fournisseur.

LOAD : technique informatique consistant à répartir uniformément la charge de travail.

MAINFRAME : ordinateur de grande puissance de traitement et qui sert d'unité centrale à un réseau de terminaux.

MODÈLE CENTRALISÉ : c'est le fait qu'Un seul serveur ou Un seul système s'occupe de toutes les tâches.

MODÈLE DISTRIBUÉ : collection de postes ou calculateurs autonomes qui sont connectés à l'aide d'un réseau de communication.

POD : instance unique d'une application.

SOURCES

SITOGRAPHIE

1. www.cenadi.cm
2. <https://blog.wescale.fr/>
3. www.lemagit.fr
4. www.redhat.com
5. <https://Wescale.developpez.com/>
6. www.figer.com
7. www.IONOS.fr
8. <https://fr.wikipedia.org/>
9. <https://hellofuture.orange.com/>
10. <https://geekflare.com/>

BILIOGRAPHIE

1. Introduction to the New Mainframe : z/OS Basics, paru en juillet 2006 écrit par International Business Machine (IBM) Corporation
2. Openshift for Developers, paru en 2015 écrit par Grant Shipley et Graham Dumpleton
3. Documentation POC IBM Montpellier de Juillet 2020

ANNEXES

Situation Géographique

