

Travail de Bachelor 2021

Analyse et développement ABAP simple sur SAP Cloud Platform (SCP) dans un contexte projet avec l'État du Valais



Etudiant-e : Yann Jollien

Professeur : Cassandre Payen

Déposé le : 03.08.2021

Résumé

Avec l'introduction de la SAP Cloud Platform, SAP veut offrir un serveur d'application dans le cloud. Sur celui-ci, les entreprises et ses utilisateurs peuvent déployer et créer des applications d'une façon rapide et agile, sans être dépendant du cœur du système SAP. Avec cette solution homogène, il est désormais possible d'étendre des applications existantes et les associer les unes avec les autres, le tout à un endroit centralisé ; le cloud.

Le but premier de ce travail est de se familiariser avec le nouveau concept et de l'analyser. Pour ce faire, un aperçu général est dans un premier temps nécessaire, évoquant les différentes technologies utilisées pour le sujet. Une analyse plus approfondie du développement ABAP sur le cloud mène à en cristalliser les avantages et les désavantages de la SAP Cloud Platform. Cette partie comporte des aspects comme les outils de développement utilisés, les différences avec du développement ABAP classique, l'utilisation de Git ou l'architecture du nouveau service cloud.

Cette analyse permet ensuite de prendre position par rapport à cette nouvelle technologie et d'évaluer son utilité pour le mandant, répondant ainsi à la question initiale du travail. Finalement, un prototype servira d'illustration pour le travail afin de montrer en pratique les points évoqués durant l'analyse.

Au terme de ce travail, nous pouvons constater que la SAP Cloud Platform propose des solutions intéressantes, la gestion de la partie hardware par SAP facilite le travail aux entreprises, le nouvel écosystème cloud permet de travailler plus rapidement et d'une manière plus efficace et ouvre des nouvelles portes en termes de possibilités de développement. Mais à ce jour, beaucoup de fonctionnalités manquent encore pour pouvoir développer pleinement sur le cloud. Bien que le langage ABAP reste (presque) le même, la migration vers la solution cloud nécessite de nombreux changements et d'adaptations.

Avant-propos

L'introduction de la SAP Cloud Platform par SAP offre des nouvelles possibilités aux utilisateurs de SAP et ouvre de nouvelles portes aux développeurs ABAP. L'objectif de SAP est de créer un écosystème regroupé dans le cloud, permettant un développement « front » – et « backend » dans un même système cloud afin d'éliminer toutes les contraintes liées à un système local et refermé vers l'extérieur.

Le travail de bachelor proposé par l'État du Valais, supervisé par le Professeur Cassandre Payen, consiste à évaluer la possibilité de migrer le développement ABAP en local vers le cloud et d'étudier ses nouvelles possibilités. Un prototype, couvrant deux scénarios différents, permet de montrer les technologies évoquées dans ce travail de bachelor.

L'étude de la nouvelle technologie cloud est dans un premier temps théorique, comportant des aspects analytiques de la nouvelle technologie cloud proposée par SAP. Elle sera suivie par une partie pratique, qui met en œuvre cette dernière pour couvrir une approche pratique.

Remerciements

Par la présente, je tiens à remercier chaleureusement le Professeur Cassandre Payen de m'avoir suivi tout au long de ce travail de bachelor et Monsieur Frédéric Morand, professeur et membre de l'équipe du Centre de Compétences SAP de la HES SO Valais/Wallis. Ils ont été d'une grande disponibilité durant toute la période du bachelor et m'ont répondu de manière très compétente à mes questions.

Mes remerciements sincères s'adressent également à l'État du Valais pour la proposition du sujet et leur disponibilité.

Un grand merci également à ma mère, Madame Marianne Jollien-Jossen, et à Mademoiselle Alison Heraud, qui ont consacré leur temps précieux pour la relecture de ce rapport.

Table des matières

Liste des figures	VII
Liste des abréviations	X
1. Introduction	1
1.1 Personnes impliquées	1
1.2 Problématique	1
2. Technologies	2
2.1 SAP ABAP	2
2.2 ABAP Dictionary	2
2.3 VCS	4
2.4 PaaS	5
3. SAP Cloud Platform	7
3.1 Rebranding de la SAP Cloud Platform	7
3.2 Concept	8
4. ABAP sur la SCP	9
4.1 ABAP Environment	9
4.2 Différence entre ABAP directement dans le cœur du système vs ABAP sur la SCP	10
5. Développement ABAP sur le cloud	11
5.1 Motivations	11
5.2 SAP Cloud Platform Architecture	12
5.3 Exposer avec ABAP dans le cloud	13
5.4 Consommer avec ABAP dans le cloud	15
5.5 Compétences à avoir	17
5.5.1 Méthode agile	17
5.5.2 DevOps	19
5.6 Coûts	20
5.7 Sécurité	21
5.7.1 Utilisation sécurisée	22
5.7.2 Développement sécurisé	24
5.8 abapGit et Git vs Transport	25
5.8.1 SAP Transport	26
5.8.2 abapGit	27
5.8.3 Git-enabled Change and Transport System	27
5.9 Développer et tester	28
5.10 Outils de développement	31
5.10.1 SAP BTP Cockpit	31
5.10.2 Eclipse	31
5.10.3 SAP Business Application Studio	32
5.10.4 GitHub	32
6. Avantages et désavantages du SCP	34
6.1 Avantages	34
6.1.1 Evolutivité	34
6.1.2 Sécurité et conformité	35
6.1.3 Intégration	35
6.1.4 Encourage les méthodes Agiles et DevOps	35
6.2 Désavantages	36
6.2.1 Coûts	36
6.2.2 Connectivité	36
6.2.3 Compatibilité ABAP	36
6.2.4 Restrictions technologiques	37
6.2.5 Complexité	37

7.	Quelle option choisir	38
8.	Prototype projet ABAP cloud	40
8.1	Configuration de la plateforme cloud	40
8.2	Configuration d'Eclipse	41
8.3	Nouveau projet ABAP	43
8.4	Création d'un package	45
8.5	Création de la base de données	47
8.6	Création d'une View	53
8.7	Création d'une projection View	56
8.8	Création d'un OData	57
8.9	Migration d'un projet local vers le cloud	61
8.10	Installation d'abapGIT	62
8.11	Configurer connexion SSL	65
8.12	Création d'un dépôt GIT	71
8.13	Migration code ABAP vers dépôt Git	73
8.14	Ajouter élément ABAP au dépôt	76
8.15	Installer plugin abapGit sur Eclipse	80
8.16	Cloner dépôt Git dans l'SAP Cloud	81
9.	Conclusion	84
9.1	Synthèse pratique	84
9.2	Synthèse générale	84
9.3	Synthèse personnelle	86
	Bibliographie	87
	Annexe I : Script Database Table	93
	Annexe II : Script ABAP Class	93
	Annexe III : Script CDS View	94
	Annexe IV : Script projection View	95
	Annexe V : Script service OData	97
	Annexe VI : Accès SAP Gui	97
	Annexe VII : Cahier des charges	98
	Annexe VIII : Aide au développement avec les outils ADT	99
	Annexe IX : Comparaison directe ABAP SAP GUI vs ABAP cloud	101
	Annexe X : SAP Business Application Studio	102
	Annexe XI : Gestion du travail	102
	Annexe XII : Product Backlog	104
	Annexe XIII : Product Backlog Prototype	105
	Annexe XIV : Détails des heures	106
	Annexe XV : Récapitulatifs des heures	107
	Déclaration sur l'honneur	108

Liste des figures

Figure 1 : Dictionary Object	3
Figure 2 : DVCS	5
Figure 3 : PaaS	6
Figure 4 : SAP Cloud Platform.....	7
Figure 5 : Vital parts of SAP Business Technology Platform (BTP), ABAP environment.....	10
Figure 6 : Architecture SCP ABAP environment.....	13
Figure 7 : RESTful Application Programming Model	14
Figure 8 : Internet Communication Framework	15
Figure 9 : Remote Function Call from ABAP cloud	16
Figure 10 : Méthode agile pour le développement.....	18
Figure 11 : Méthode DevOps pour le développement.....	20
Figure 12 : Prix pour ABAP environment	21
Figure 13 : Prix pour le SAP Cloud Transport Management ¹²	21
Figure 14 : Prix pour le SAP Workflow Management ¹²	21
Figure 15 : Aperçu sécurité sur la SCP	23
Figure 16 : Accès sécurisé à la SCP.....	24
Figure 17 : OAuth avec SCP	25
Figure 18 : Exemple de distribution de tâches	26
Figure 19 : Environnement gCTS	28
Figure 20 : Cycle de développement DevOps	29
Figure 21 : Aperçu environnement de test	30
Figure 22 : Récapitulatif avantages et désavantages	34
Figure 23 : Configuration SCP – Choix région	40
Figure 24 : Configuration SCP - Instance ABAP	41
Figure 25 : Configuration Eclipse - Install New Software	41
Figure 26 : Configuration Eclipse - Ajout des outils ADT	42
Figure 27 : Configuration Eclipse - Certificat.....	42
Figure 28 : Configuration Eclipse : Changement perspective	43
Figure 29 : Configuration Eclipse - Perspective ABAP	43
Figure 30 : Nouveau projet ABAP cloud.....	44
Figure 31 : Nouveau projet ABAP cloud - Service Key.....	44
Figure 32 : Nouveau projet ABAP cloud - Service Key2.....	44
Figure 33 : Nouveau projet ABAP cloud - Login.....	45
Figure 34 : Nouveau projet ABAP cloud - Vérification	45
Figure 35 : Création d'un package - Nouveau package	46
Figure 36 : Création d'un package - ABAP package	46
Figure 37 : Création d'un package - Nom et description du package	46
Figure 38 : Création d'un package - Ordre de transport.....	46
Figure 39 : Création d'un package- Ajout package aux favoris	47
Figure 40 : Création de la base de données - Other ABAP Repository Object.....	47
Figure 41 : Création de la base de données - Database Table	48
Figure 42 : Création d'une base de données - Nom et description.....	48
Figure 43 : Création d'une base de données - Ordre de transport	48
Figure 44 : Création d'une base de données - Script	49
Figure 45 : Création d'une base de données - Activation.....	49
Figure 46 : Création d'une base de données - Preview	50
Figure 47 : Création d'une base de données - Fenêtre de Preview	50
Figure 48 : Création d'une base de données - Nouvelle classe ABAP	50
Figure 49 : Création d'une base de données - Nom et description de la classe	51
Figure 50 : Création d'une base de données - Ordre de transport Class	51
Figure 51 : Création d'une base de données - Script Class	52
Figure 52 : Création d'une base de données - Activer script.....	52

Figure 53 : Création d'une base de données - Exécuter script.....	53
Figure 54 : Création d'une base de données - Validation.....	53
Figure 55 : Création d'une base de données - Vérification insertion	53
Figure 56 : Création d'une View - Other ABAP Repository Object.....	54
Figure 57 : Création d'une View - Data Definition	54
Figure 58 : Création d'une View - Nom et description	54
Figure 59 : Création d'une View - Ordre de transport View	55
Figure 60 : Création d'une View - Script	55
Figure 61 : Création d'une View - Activer script View	56
Figure 62 : Création d'une projection View - Other ABAP Repository Object	56
Figure 63 : Création d'une projection View - Data Definition	56
Figure 64 : Création d'une projection View - Nom et description.....	57
Figure 65 : Création d'une projection View - Ordre de transport projection View.....	57
Figure 66 : Création d'une projection View - Activation	57
Figure 67 : Création d'un OData - New Service Définition	58
Figure 68 : Création d'un OData - Nom et description	58
Figure 69 : Création d'un OData - Ordre de transport OData.....	58
Figure 70 : Création d'un OData - Activation.....	59
Figure 71 : Création d'un OData - Service Binding	59
Figure 72 : Création d'un OData - Nom, Description et Type	59
Figure 73 : Création d'un OData - Ordre de transport binding.....	60
Figure 74 : Création d'un OData - Activation binding	60
Figure 75 : Création d'un OData - Publier binding	60
Figure 76 : Création d'un OData - Publication du binding	61
Figure 77 : Création d'un OData - Aperçu Fiori.....	61
Figure 78 : Installation d'abapGIT - Nouveau programme.....	62
Figure 79 : Installation d'abapGit – Titre, type de programme et variant	62
Figure 80 : Installation d'abapGIT - Ordre de transport.....	63
Figure 81 : Installation d'abapGIT - Latest build.....	63
Figure 82 : Installation d'abapGIT - Activer programme	63
Figure 83 : Installation d'abapGIT - Confirmation d'activation	64
Figure 84 : Installation d'abapGit - Exécuter le programme	64
Figure 85 : Installation d'abapGit - Page d'accueil	64
Figure 86 : Configurer connexion SSL - Sélection profil.....	65
Figure 87 : Configurer connexion SSL - Modifier profil.....	65
Figure 88 : Configurer connexion SSL – Paramètre client_ciphersuites.....	65
Figure 89 : Configurer connexion SSL - Paramètre nom et valeur	66
Figure 90 : Configurer connexion SSL - Copier	66
Figure 91 : Configurer connexion SSL – Paramètre ciphersuites.....	66
Figure 92 : Configurer connexion SSL - Paramètre 2 nom et valeur	67
Figure 93 : Configurer connexion SSL - Copier	67
Figure 94 : Configurer connexion SSL - Sauvegarder.....	67
Figure 95 : Configurer connexion SSL - Sauvegarder certificat.....	68
Figure 96 : Configuration connexion SLL - Aperçu des certificats	69
Figure 97 : Configuration connexion SSL - Certificats téléchargés	69
Figure 98 : Configuration connexion SSL - Change de mode	69
Figure 99 : Configuration connexion SSL - SSL Client.....	70
Figure 100 : Configuration connexion SSL - Ajout des certificats	70
Figure 101 : Configuration connexion SSL - Sélection certificat.....	70
Figure 102 : Configuration connexion SSL- Valider ajout certificat.....	71
Figure 103 : Configuration connexion SSL - Liste des certificats ajoutés	71
Figure 104 : Création d'un dépôt Git - Nouveau dépôt.....	72
Figure 105 : Création d'un dépôt Git - Configuration.....	72
Figure 106 : Création d'un dépôt Git - Initial commit	73

Figure 107 : Migration code ABAP vers dépôt Git - URL.....	73
Figure 108 : Migration code ABAP vers dépôt Git - New Online	73
Figure 109 : Migration code ABAP vers dépôt Git - Create Package	74
Figure 110 : Migration code ABAP vers dépôt Git - Configuration	74
Figure 111 : Migration code ABAP vers dépôt Git - Own Requests	74
Figure 112 : Migration code ABAP vers dépôt Git - Workbench description	75
Figure 113 : Migration code ABAP vers dépôt Git - Valider requête	75
Figure 114 : Migration code ABAP vers dépôt Git - Cloner dépôt.....	75
Figure 115 : Migration code ABAP vers dépôt Git - Dépôt sur SAP.....	76
Figure 116 : Ajouter élément ABAP au dépôt - Nouveau projet.....	76
Figure 117 : Ajout élément ABAP au dépôt - Package	77
Figure 118 : Ajout élément ABAP au dépôt - Nouvelle classe	77
Figure 119 : Ajout élément ABAP au dépôt - Synchroniser	78
Figure 120 : Ajout élément ABAP au dépôt - Nouvelle classe dans le dépôt.....	78
Figure 121 : Ajout élément ABAP au dépôt - Stage	78
Figure 122 : Ajout élément ABAP au dépôt - Add all et commit.....	78
Figure 123 : Ajout élément ABAP au dépôt - Commit.....	79
Figure 124 : Ajout élément ABAP au dépôt - Check.....	79
Figure 125 : Installer plugin abapGit sur Eclipse - Ajouter nouvel outil.....	80
Figure 126 : Installer plugin abapGit sur Eclipse - Ajout URL.....	80
Figure 127 : Installer plugin abapGit sur Eclipse - Changer vue	81
Figure 128 : Cloner dépôt Git dans l'SAP Cloud - Ajout dépôt	81
Figure 129 : Cloner dépôt Git dans l'SAP Cloud - Ajout URL	82
Figure 130 : Cloner dépôt Git dans l'SAP Cloud - Pull.....	82
Figure 131 : Cloner dépôt Git dans l'SAP Cloud - Vérifier.....	83
Figure 132 : Assistance - Informations élément ABAP	99
Figure 133 : Assistance - Informations élément ABAP nouvelle fenêtre	99
Figure 134 : Assistant erreurs.....	100
Figure 135 : Configuration format code ABAP	100
Figure 136 : Comparaison ABAP cloud vs ABAP local.....	101
Figure 137 : SAP Business Application Studio – Code et visuel	102
Figure 138 : Récapitulatifs des heures	107

Liste des abréviations

ABAP	Advanced Business Application Programming
ADT	ABAP Development Tools
API	Application Programming Interface
BTP	Business Technology Platform
CAP	Cloud Application Programming Model
CDS	Core Data Services
ChaRM	Change Request Management
CI	Continuous Integration
CRUD	Create, Read, Update, Delete
DevOps	Development and Operations
DVCS	Distributed Version Control Systems
gCTS	Git-enabled Change and Transport System
GUI	Graphical User Interface
HANA	High Performance Analytic Appliance
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
ICF	Internet Communication Framework
IDE	Integrated Development Environment
ISO	International Organization for Standardization
JRE	Java Runtime Environment
OAuth	Open Authorization
OData	Open Data Protocol
PO	Product Owner
QGM	Quality Gate Management
RAM	Random Access Memory
RAP	RESTful Application Model
REST	Representational State Transfer
RFC	Remote Function Call
SaaP	Software as a Product
SaaS	Software as a Service\$
SAML	Security Assertion Markup Language
SAP	Systems Applications Products
SCP	SAP Cloud Platform
SMSI	Système de management de la sécurité de l'information
SMTP	Simple Mail Transfer Protocol

TMS	Transport Management
URL	Uniform Resource Identifier
VCS	Version Control System

1. Introduction

1.1 Personnes impliquées

Diverses personnes étaient impliquées lors de la rédaction du travail de bachelor.

Rôle	Nom	E-Mail
Auteur	Yann Jollien	yann.jollien@students.hevs.ch
Professeur	Cassandre Payen	cassandre.payen@hevs.ch
Assistant	Frédéric Morand	frederic.morand@hevs.ch
Mandant	Giorgio Orlando	giorgio.orlando@admin.vs.ch
Mandant	Jean-Philippe Salamin	jean-philippe.salamin@admin.vs.ch

1.2 Problématique

L'État du Valais utilise actuellement SAP, développe les outils nécessaires en interne et souhaite transformer son système d'information. Le transfert du développement SAP Fiori sur SAP a déjà eu lieu, ils envisagent à présent de migrer la partie du développement ABAP sur la SAP Cloud Platform.

Afin d'atteindre l'objectif de basculer la partie ABAP vers le cloud, il est nécessaire d'effectuer une analyse des nouvelles possibilités offertes par SAP Cloud Platform pour le développement ABAP par rapport au développement standard ABAP directement dans le cœur du système.

Il est demandé d'expliquer le fonctionnement du développement ABAP sur le cloud ainsi que les Best Practices à adopter pour ce faire. Un autre aspect à aborder est l'utilisation de Git pour le versionnage.

À l'aide d'un prototype qui fait suite à l'analyse, l'utilisation de la nouvelle technologie sera démontrée et soulignera le travail de recherche.

2. Technologies

Dans cette partie, toutes les technologies importantes mentionnées seront détaillées.

2.1 SAP ABAP

Le Advanced Business Application Programming (ABAP) est un langage de programmation développé par la société SAP. ABAP fait partie des langages de quatrième génération, apparus dans les années 1980.

A l'aide du ABAP, chaque client peut développer ses propres interfaces pour divers applications et outils. Aujourd'hui, les programmes SAP sont exclusivement développés en ABAP.

Java et certains programmes C++ démontrent quelques similitudes avec le langage ABAP. Mais contrairement à ceux-ci, ABAP se sert de l'intégralité des informations dans la base de données SAP. Les bases de données ABAP se servent du principe de la Random Access Memory (RAM), les données et informations fréquemment utilisées sont stockées afin d'accéder rapidement à partir de la mémoire locale (Netinbag, 2021).

En ABAP, plusieurs types de programmes peuvent être utilisés. Le type détermine les attributs techniques basiques d'un programme. Il est impératif que le type soit défini lors de la création du programme. La principale différence entre les différents types de programmes est la manière dont les blocs de traitement du programme sont appelés. Lorsqu'un programme ABAP est lancé, le « runtime » lance un processeur (qui dépend du type de programme) qui va exécuter le 1^{er} bloc de traitement ABAP.

Le type de programme détermine par exemple son mode de lancement, via l'utilisateur, en arrière-plan ou via une interface externe (Advanced Business Application Programming, support de cours SAP).

2.2 ABAP Dictionary

Le ABAP Dictionary contient la définition de tous les objets SAP groupés sous forme de dépôt et ainsi transportables via un ordre dit « Workbench ». Le Dictionary permet de créer, modifier et supprimer les éléments d'une base de données SAP. Il est important de comprendre que le développeur n'accède pas directement à la base de données réelle mais uniquement au dictionnaire ABAP (ABAP Dictionary, support de cours SAP).

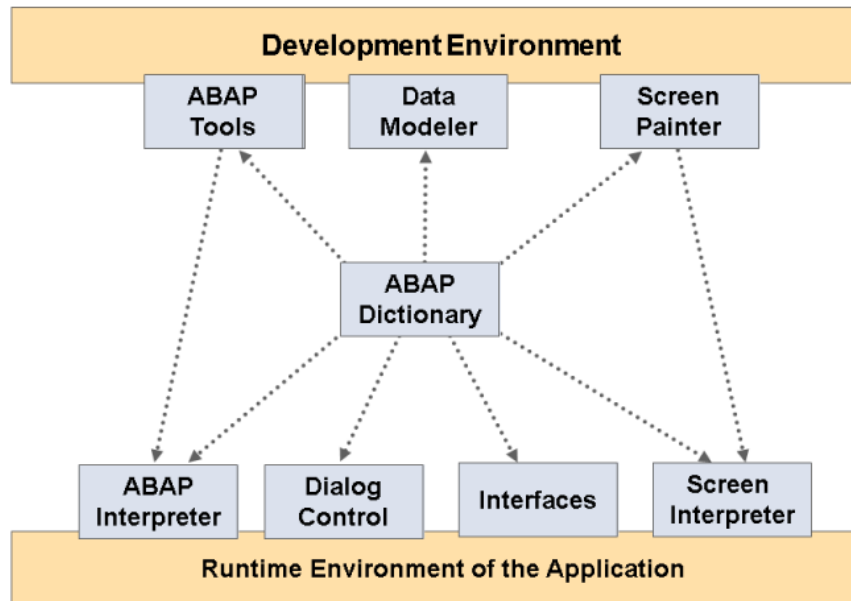


Figure 1 : Dictionary Object¹

Le groupe principal du ABAP Dictionary comporte les types de données. Les données de type élément sont disponibles pour tous les entrepôts et objets. Leurs attributs peuvent être définis sur la base d'un type primitif ou en utilisant le domaine. Les objets « domaine » servent à décrire les propriétés d'un type élément : type de donnée, longueur de donnée, plage de valeurs et description. Ils définissent l'affichage des données « éléments » dans les tableaux, écrans et rapports. La structure est un type ABAP basique mais très importante. Elle est un ensemble de champs prédéfinis qui peut contenir des données structurales et qui peut décrire les zones de travail, les types de table et les tables de la base de données. Enfin, les types de tableaux peuvent décrire des tables internes dans des programmes ABAP. Ils ressemblent aux structures car ils consistent à un ensemble prédéfini de champs, mais ils ont une caractéristique utile en plus : ils peuvent contenir des clés primaires et secondaires ce qui facilite les requêtes.

Tous les objets se trouvant dans le ABAP Dictionary peuvent être réutilisés. Il est possible de créer un nombre infini de tables basées sur les mêmes éléments, un nombre infini d'éléments basés sur le même domaine et cetera. Ceci est un autre aspect important de l'ABAP Dictionary – minimisation des redondances et élimination des routines. Par exemple,

¹ Source : <https://erproof.com/abap/sap-abap-training/sap-abap-data-dictionary/>

il n'est pas nécessaire de recréer une table avec 100 entrées, il est possible de copier la structure à partir d'une table existante et de changer uniquement les paramètres voulus (Astashonak, 2017).

2.3 VCS

Un Version Control System (VCS) se charge de gérer le développement d'un projet évoluant. Tous les changements, opérés par un développeur, y sont enregistrés. Un VCS a plusieurs domaines d'applications et son objectif est de simplifier le développement.

Lors du développement, le code est modifié en permanence, il peut être changé, supprimé ou rajouté. De plus, le code doit souvent être partagé entre développeurs. Effectuer ces opérations en local engendre certains risques, comme la perte de code. C'est ici qu'entre en jeu un système de contrôle qui permet d'offrir une solution aux problèmes mentionnés ci-dessus.

Selon un article de l'université de Malaisie Terengganu, les projets ont aujourd'hui majoritairement adopté la méthode du Distributed Version Control Systems (DVCS). L'avantage de cette méthode est qu'elle enregistre tout l'historique des fichiers localement, tout en synchronisant les fichiers locaux avec le serveur afin de partager les changements avec l'équipe. Il est donc possible de conserver une branche locale sans avoir à passer par un serveur. En cas d'erreur sur le serveur, le projet sur une machine locale peut être copié sur le serveur pour avoir sa dernière version. Git travaille avec cette architecture par exemple.

Distributed VCS

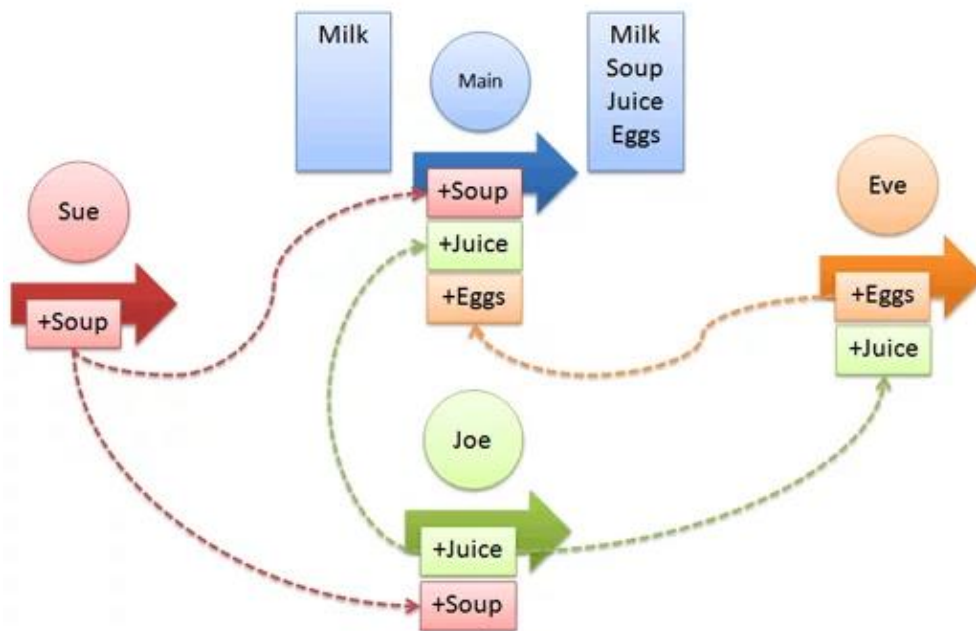


Figure 2 : DVCS²

Les VCS simplifient et accélèrent le développement de software. Le travail en équipe est possible en local, le travail d'une tierce personne peut être modifié et changé, sans perdre le sien. En cas de changements, les développeurs en sont informés afin de décider quelle version maintenir. D'autant plus, il est possible de revenir à un état spécifique du projet, puisque tout l'historique est conservé. Les développeurs sont donc à l'abri de suppressions ou changements accidentels (Nurlisa Zolkipli, Ngah, & Deraman, 2018).

2.4 PaaS

La plateforme en tant que service est une plateforme qu'un client ou utilisateur peut utiliser pour développer, exécuter et gérer des applications ne nécessitant pas une infrastructure locale. Il est donc possible d'accéder à des ressources au niveau hardware, supérieures à celles d'une machine locale. Ce service est hébergé dans le cloud et est majoritairement accessible via un navigateur web.

² Source : <https://betterexplained.com/articles/intro-to-distributed-version-control-illustrated/>

Les avantages principaux des PaaS sont le gain de temps et d'argent. Avec l'utilisation d'un PaaS, la partie maintenance et mise en place d'une infrastructure n'est plus nécessaire. Il n'y a pas besoin d'investir dans de l'hardware dernier cri régulièrement. De plus, il est possible de tester rapidement de nouvelles innovations et technologies, puisqu'il n'est pas nécessaire de s'acquérir physiquement de ces dernières.

En revanche, il ne faut pas oublier qu'en utilisant des PaaS, nous sommes dépendant de ses fournisseurs. Des temps d'arrêts pour des maintenances par exemple pourraient être contrariants. A ne pas négliger la partie sécurité, puisque les données sont souvent stockées sur des serveurs externes (Oracle, 2021).

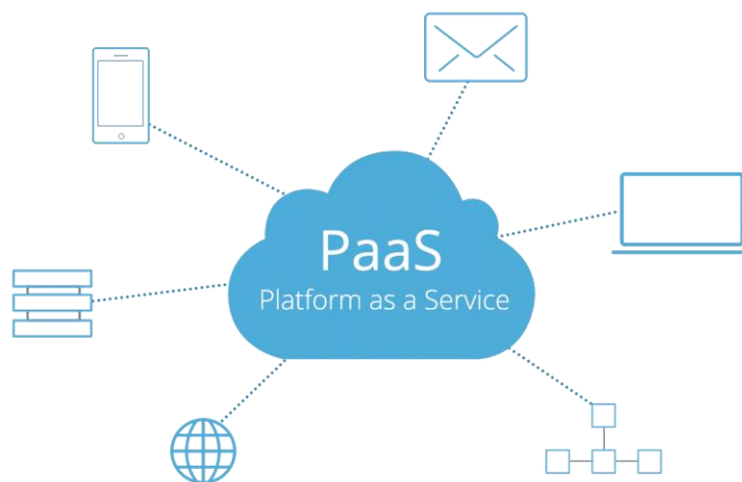


Figure 3 : PaaS³

Pour illustrer plus facilement les PaaS, l'industrie du jeu vidéo s'est récemment penchée sur la thématique. Des grandes entreprises comme Nvidia, Google ou encore Microsoft proposent un service Cloud Gaming.

Les images et le son d'un jeu sont transmis en temps réel sur l'écran de l'utilisateur via internet, sans qu'une console de jeu ne soit nécessaire. Concrètement, le jeu tourne sur le serveur du fournisseur et permet à l'utilisateur de pouvoir jouer à un jeu vidéo avec les graphismes au maximum, sur n'importe quel appareil doté d'un écran et connecté à internet (Bastien, 2019).

³ Source : <https://blog.iron.io/what-is-platform-as-a-service/>

3. SAP Cloud Platform

La SAP Cloud Platform est un service cloud PaaS offert par SAP pour le développement et déploiement d'applications web. SAP se charge entièrement du management de l'infrastructure, de la maintenance du hardware et de l'upgrade des composants. Le service cloud de SAP offre une large palette d'outils comme la gestion de base de données, le stockage et backup de données ainsi que le développement de programmes ou d'applications multi-plateformes. Les utilisateurs de la SCP ont la possibilité de l'utiliser pour la gestion complète du développement de programmes ou alors de façon hybride, avec une partie du développement en local et une autre dans le cloud (Tutorialspoint, 2019).

Avec l'utilisation de la SCP, les entreprises ont la possibilité de réagir rapidement aux changements d'exigences du marché afin de mettre à disposition des solutions dans un bref délai. Le service cloud a été présenté pour la première fois par SAP en 2012 (Luber & Karlstetter, Was is die SAP Cloud Platform?, 2019).

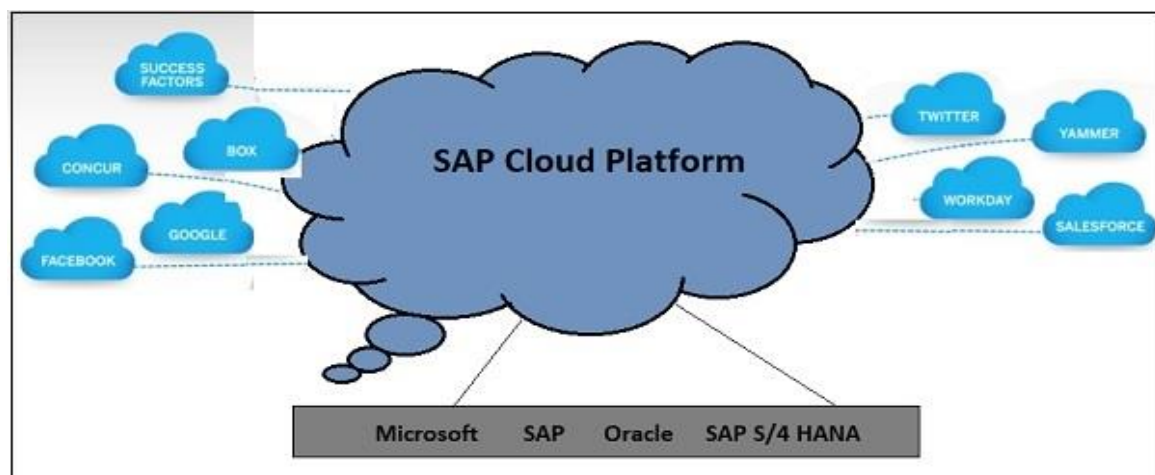


Figure 4 : SAP Cloud Platform⁴

3.1 Rebranding de la SAP Cloud Platform

La SCP a officiellement été retirée par SAP mi-janvier de l'année 2021 afin de se diriger vers une stratégie « One Platform ». La partie communication au niveau de la plateforme

⁴ Source : https://www.tutorialspoint.com/sap_cloud_platform/sap_cloud_platform_tutorial.pdf

devrait se focaliser sur la Business Technology Platform (BTP) qui est la nouvelle plateforme orientée business avec une architecture transformant des données en solutions business. La SAP BTP contient des solutions comme la gestion de base de données, l'analyse, l'intégration et le développement d'applications ainsi qu'une variété de technologies modernes comme le machine learning.

Il est important de comprendre que la SAP BTP ne remplace pas la SAP Cloud Platform. En effet, la partie SCP représente uniquement le développement d'applications du SAP BTP. La SCP est organisée et divisée de la manière suivante :

- SAP Integration Suite – services et capacités capables de connecter les applications, processus, utilisateurs et appareils. Partie intégrante de la SAP BTP, elle simplifie l'intégration et harmonise les processus.
- SAP Extension Suite – services et capacités capables de simplifier le développement d'applications et extensions. Partie intégrante de la SAP BTP, elle aide les développeurs à réaliser des solutions à haute valeur.

Comme il s'agit d'un changement stratégique au niveau de la marque et non pas de son fonctionnement, les fonctionnalités de la SCP ne changent pas et peuvent être utilisées de la même façon qu'avant (SAP, Rebranding of SAP Cloud Platform, 2021).

Afin d'éviter toute confusion durant la réalisation du travail de bachelor, nous allons continuer à utiliser le terme « SAP Cloud Platform » comme il a été utilisé dans le document « Données du travail de bachelor ».

3.2 Concept

Les trois points clés et motivations de la SCP sont la méthodologie agile, l'optimisation et la digitalisation.

Le concept agile permet la personnalisation et extension rapide d'applications cloud ou locales. L'extension et l'ajout de nouvelles fonctionnalités à des applications existantes se font d'une façon fluide et sont adaptées aux besoins du client. La transformation d'applications locales en applications interconnectées cloud pour les employées, utilisateurs et partenaires est également une partie importante de l'agile (SAP, SAP Cloud Platform, 2017).

« *Achieve business agility by leveraging and extending your current applications* » (SAP, SAP Cloud Platform, 2017)

L'optimisation signifie la connexion d'applications, données et processus, peu importe la source et l'endroit de déploiement de ces dernières. Il est également plus simple de transmettre les bonnes informations aux bonnes personnes, au bon endroit et au bon moment (SAP, SAP Cloud Platform, 2017).

« *Optimize business processes by reducing data silos and operational inefficiencies* » (SAP, SAP Cloud Platform, 2017)

Le dernier point à évoquer est celui de la digitalisation. Pour créer des applications innovantes, il est important d'utiliser des technologies modernes. De plus, il est possible d'utiliser des standards comme JAVA, Python, Node.js grâce à l'intégration de technologies OpenStack sur le cloud (SAP, SAP Cloud Platform, 2017).

« *Innovate and build digital cloud apps using open standards* » (SAP, SAP Cloud Platform, 2017)

4. ABAP sur la SCP

Selon Karl Kessler, Vice President of Product Management ABAP Platform chez SAP, le développement ABAP sur le cloud est une opportunité pour les entreprises de pousser leurs stratégies digitales vers leur propre environnement SAP. Une approche entièrement digitale garantit non seulement une haute efficacité, mais aussi une flexibilité et stabilité durant le développement. Des solutions cloud permettent une continuité des processus sans interruption tout en offrant des mises à jour régulières. Avec le développement ABAP sur le cloud, ces capacités s'étendent au développement d'applications pour le système SAP (Kessler, Make the Move to SAP Cloud Platform ABAP Environment with Confidence, 2020).

4.1 ABAP Environment

L'environnement ABAP est une plateforme innovante, complètement gérée par SAP, leur permettant de « push » les dernières nouveautés dans l'environnement ABAP. Avec l'environnement ABAP, les utilisateurs et partenaires n'ont plus à gérer les mises à jour, « patches », « hot fix », et cetera. Suite à cela, les utilisateurs et partenaires ont la possibilité de travailler en permanence avec la dernière version de la plateforme.

L'environnement ABAP fait partie de la SAP BTP et est accessible via le cockpit BTP. D'un point de vue développeur, les outils de développement ABAP dans Eclipse ont été mis à jour et adaptés au nouvel environnement. Ces changements signifient la fin du SAP Graphical User Interface (GUI) pour des tâches de développement comme le SE80 (code de transaction pour

le navigateur d'objets) par exemple. L'objectif principal de SAP est de rendre l'environnement ABAP compatible avec toute la plateforme BTP. La connexion avec l'environnement ABAP est un prérequis pour le travail sur le cloud. Les utilisateurs et partenaires ont la possibilité d'utiliser un répertoire Git pour transporter le code entre les divers systèmes d'environnement ABAP. Par exemple, il est possible avec Git de migrer un système local vers le cloud (Wahl, 2020).

L'environnement ABAP tourne sur la base de données SAP High Performance Analytic Appliance (HANA) avec un accès aux objets de la base de données. Les options de connexion couvrent deux scénarios différents : la connexion entre les systèmes locaux ou sur le cloud (Press, 2021).

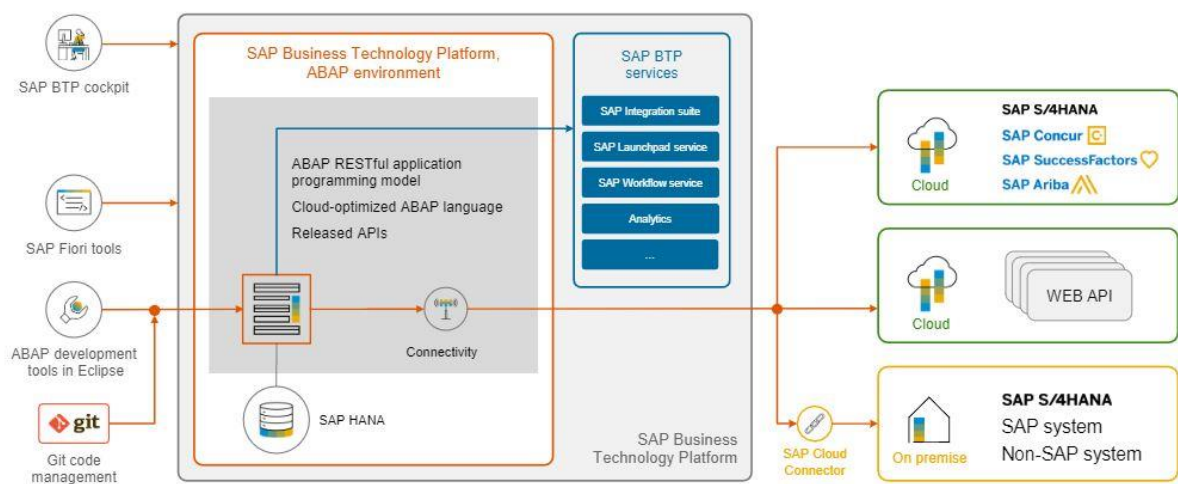


Figure 5 : Vital parts of SAP Business Technology Platform (BTP), ABAP environment⁵

4.2 Différence entre ABAP directement dans le cœur du système vs ABAP sur la SCP

L'une des premières différences concerne l'infrastructure. SAP gère le fonctionnement de l'infrastructure cloud. Le support est également garanti par SAP sur le SCP et l'investissement financier touche uniquement le service avec le choix du développement sur cloud pendant que le développement, dans le cœur du système, requiert une infrastructure telle que le hardware (serveur) et software (Allen, 2018). Suite à cela, la programmation avec ABAP sur le cloud se fait toujours automatiquement avec la dernière version du ABAP qui est constamment élargie

⁵ Source : <https://www.sap.com/documents/2018/08/7a62516c-157d-0010-87a3-c30de2ffd8ff.html>

et qui évolue avec le temps. En local, chaque utilisateur est responsable de son infrastructure et les nouvelles versions doivent être chargées manuellement.

Avec le développement en cloud, chaque utilisateur dispose d'une instance propre à lui sur la plateforme cloud. Il est donc plus simple d'intégrer un système VCS comme Git.

L'extension et le développement pour des systèmes SAP en local ou des systèmes S/4HANA peuvent être développés en « Side-by-Side Extensions ». Le fait de développer des programmes sur la plateforme cloud avec une connexion à S/4HANA est appelé « Side-by-Side Extension » (Nebeling, 2020).

Grâce au développement sur SCP, les accès aux différents services et Application Programming Interfaces (API) de SAP sont plus simples et rapides. L'utilisation d'API avec ABAP requiert une certaine préparation et démontre une certaine complexité.

Au niveau de l'interface utilisateur, les services de la plateforme cloud ABAP Environment sont mis à disposition via le Open Data Protocol (OData) ou via Hypertext Transfer Protocol (HTTP). Ces derniers peuvent donc être utilisés par une interface Fiori ou par un autre Framework UI (User interface). Les technologies UI classiques du ABAP utilisées en local comme le SAP GUI ou Web GUI ne sont plus disponibles.

Le ABAP PaaS utilise une version spéciale cloud pour le langage de programmation ABAP. Des instructions qui influencent le développement cloud ou qui ne sont pas compatibles avec celui-ci sont bannis. De plus, des instructions obsolètes comme « CALL SCREEN » ne sont plus supportées sur le développement cloud. Il en va de même pour les technologies Dynpro (standard SAP UI technology pour créer des interfaces avec le ABAP Workbench) (Lordieck, 2019).

5. Développement ABAP sur le cloud

Dans cette partie nous allons entrer plus en détail sur le développement ABAP dans le cloud et ses capacités.

5.1 Motivations

- Intégration de solutions existantes et systèmes hybrides : en développant du ABAP sur le cloud, l'interaction entre le cœur du système et les différentes « Software-as-a-Service » (SaaS) est fortement simplifiée (Schmidt, 2018). Le service SAP Fieldglass par exemple, qui est une application qui permet de gérer le processus de recrutement au sein de l'entreprise, peut facilement être intégré (Morrison, 2018).

- Utilisation rapide d'applications et services préfabriqués : la SAP Cloud Platform met à disposition un grand nombre de services préconfigurés. Ces services peuvent être utilisés comme base pour développer des applications individuelles d'une façon plus rapide avec ABAP. De plus, les utilisateurs profitent d'un écosystème SAP qui évolue, grandit constamment et met régulièrement à disposition de nouveaux services. Il est non seulement possible d'accéder à des services et solutions propres à SAP mais également à ceux des partenaires de SAP. L'accès à un grand nombre d'applications open cloud pour développer des applications aux besoins de l'entreprise avec ABAP est donc possible. Ces services et solutions sont mis à disposition dans le SAP Store. Celui-ci compte actuellement plus de 1500 services (Schmidt, 2018).
- Développement simple : le développement sur le cloud met à disposition des développeurs plusieurs outils de développement. Les programmeurs peuvent utiliser leurs compétences et connaissances du développement ABAP sans restriction afin de développer de nouvelles applications avec les méthodes agiles et Development and Operations (DevOps) (Schmidt, 2018).

5.2 SAP Cloud Platform Architecture

Selon Karl Kessler, Vice President of Product Management ABAP Platform, l'architecture SAP Cloud Platform ABAP Environment ressemble beaucoup à l'architecture du SAP S/4HANA Cloud. Les administrateurs et les utilisateurs avec les autorisations nécessaires sur le SCP cockpit peuvent créer une nouvelle instance ABAP. Une fois l'instance créée, l'environnement ABAP sur le cloud est fonctionnel et peut être accédé par les outils de développement ABAP sur Eclipse.

Les applications ABAP sont connectées au « backend » via des protocoles standards tels que HTTP et le OData. Il est également possible de connecter ces applications à un « backend » SAP S/4HANA Cloud et dans un futur proche, la possibilité de connexion à un « backend » local via un connecteur cloud devrait également être envisageable.

Avec le SCP ABAP environment, SAP Fiori est la seule technologie pour les interfaces graphiques disponible. SAP GUI et Web GUI sont obsolètes puisque la création d'écrans n'est plus possible avec le cloud.

Le SCP ABAP environment fait partie de l'écosystème Cloud Foundry, ce qui permet aux applications ABAP d'utiliser d'autres services tel que SAP Leonardo. D'un point de vue logique, le mécanisme de transport et les ordres de transports sont maintenus, permettant de développer des applications ABAP dans un système puis de le transporter vers un système

de test par exemple. Contrairement au développement local, les transports ne sont pas enregistrés dans le dictionnaire local de l'entreprise, mais exportés dans un dépôt Git. Avec cette approche, des outils supportant l'intégration continue peuvent être intégrés au développement cloud (Kessler, Develop New Cloud Applications in ABAP with SAP Cloud Platform ABAP Environment , 2018).

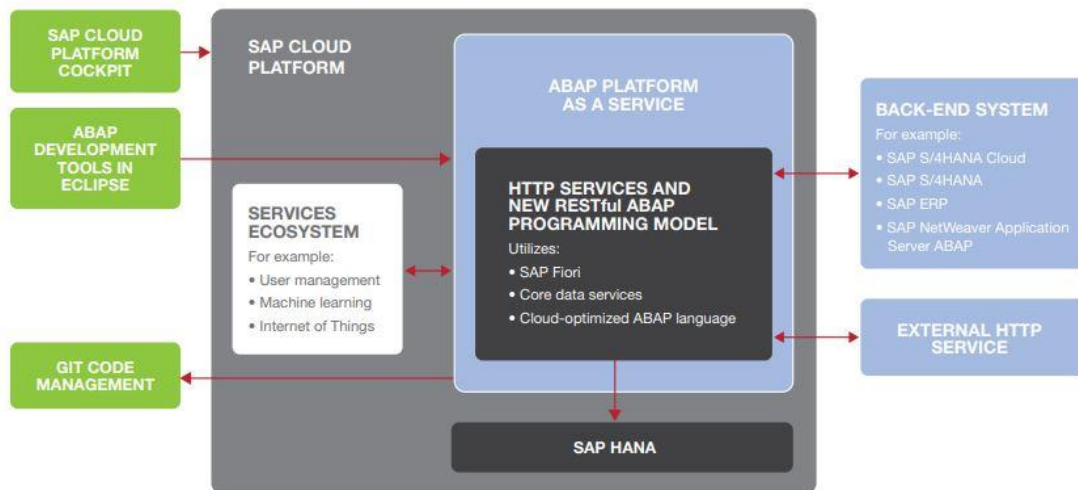


Figure 6 : Architecture SCP ABAP environment⁶

5.3 Exposer avec ABAP dans le cloud

L'utilisation de l'environnement ABAP nous permet d'exposer deux types de services :

- **OData services** : le OData est un protocole standard construit sur la base des protocoles existants HTTP et Representational State Transfer (REST) supportant les opérations « Create, Read, Update, Delete » (CRUD) afin de créer et utiliser des données d'API (Nair, 2013). Le ABAP RESTful Application Model (RAP) définit l'architecture pour le développement « end-to-end » efficace pour des services OData comme des applications Fiori dans l'environnement ABAP. L'architecture permet le support de tous les types d'applications Fiori mais également la publication d'APIs Web. Elle est basée sur des technologies et frameworks comme les Core Data Services (CDS) pour la définition des modèles de données (Djomo, 2019).

⁶ Source : <https://assets.cdn.sap.com/sapcom/docs/2018/09/06c97f15-1e7d-0010-87a3-c30de2ffd8ff.pdf>

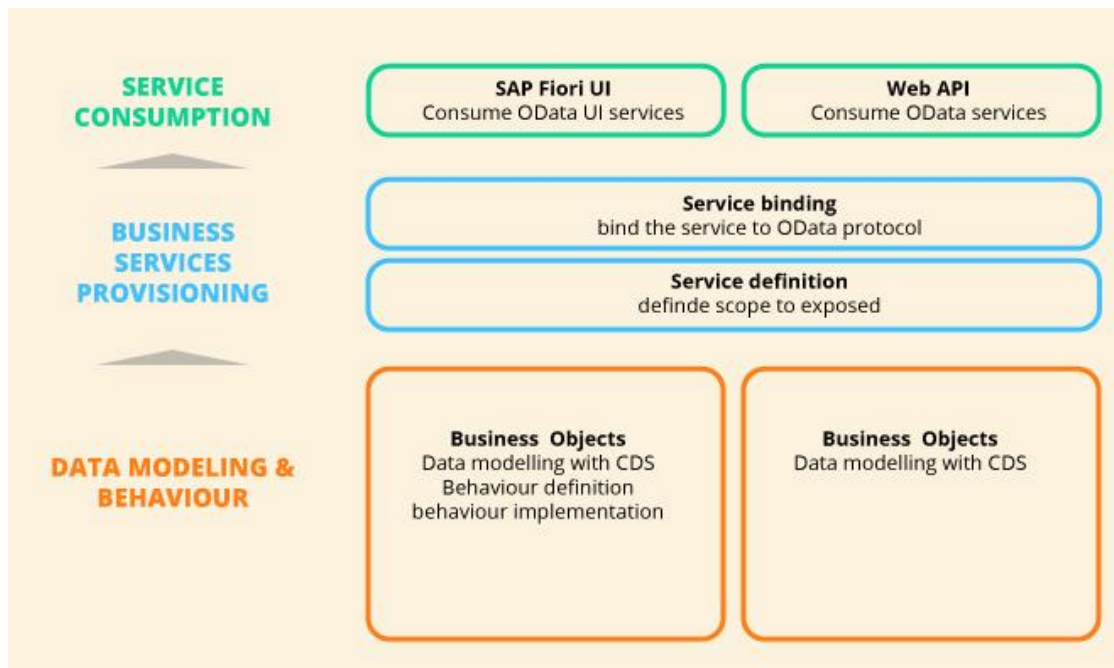


Figure 7 : RESTful Application Programming Model⁷

- HTTP services : HTTP services permet aux applications ABAP d'être affichées dans le browser. Le service est basé sur le ABAP Internet Communication Framework (ICF). L'API ICF est composé d'interfaces et de classes et soutient les protocoles internet HTTP, Hypertext Transfer Protocol Secure (HTTPS) et Simple Mail Transfer Protocol (SMTP) (SAP, ICF - Internet Communication Framework, 2016).

⁷ Source : <https://erlebe-software.de/knowhow/abap-restful-programming-model/>

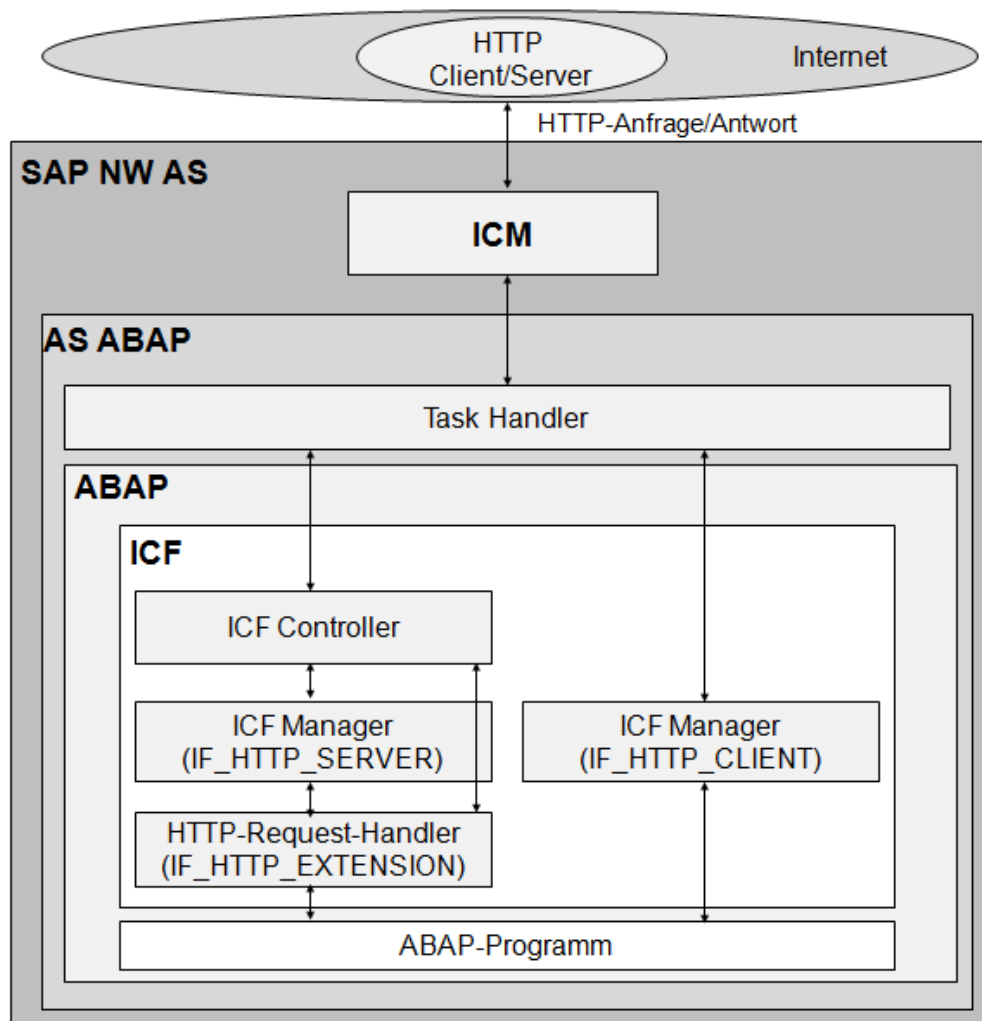


Figure 8 : Internet Communication Framework⁸

5.4 Consommer avec ABAP dans le cloud

L'utilisation de l'environnement ABAP nous permet de consommer quatre types de services :

- **HTTP services (client HTTP) :** on peut utiliser le protocole HTTP pour établir une connexion entre notre code ABAP. Le client HTTP permet de communiquer avec S/4HANA cloud, des systèmes locaux ou tout autre service HTTP exposé à internet (SAP, Enable HTTP Communication in Your ABAP Code, 2021).

⁸ Source : https://help.sap.com/doc/abapdocu_750_index_htm/7.50/de-de/abenicf.htm

- OData services (modèle de consommation) : basé sur le service OData à distance, nous pouvons créer un nouveau service qui améliore le OData avec des informations supplémentaires. Nous pouvons par exemple développer notre propre OData qui utilise un Web API et une base de données de notre système. Cet API représente un service à distance duquel nous allons récupérer les données afin de compléter notre OData. Pour pouvoir créer une nouvelle application Fiori par exemple, il faudra créer une entité CDS dans la classe ABAP qui gère l'appel des différentes sources (SAP, Developing a UI Service with Access to a Remote Service, s.d.).
- Remote Function Call (RFC) : RFC est un mécanisme qui permet aux applications de communiquer et échanger des données avec d'autres systèmes. Il existe deux scénarios dans SAP ; l'appel d'une interface pour des programmes ABAP ou non-SAP (guru99, Remote Function Call (RFC) in SAP Tutorial, 2021). Le RFC sert donc à communiquer avec des systèmes ABAP locaux afin de récupérer des données de ces derniers. D'un point de vue technique, la connexion est établie via un Cloud Connector qui est exécuté dans le « backend » et qui va créer le lien entre l'environnement ABAP et le système en local (Kessler, New Features for SAP Cloud Platform ABAP Environment, 2019).

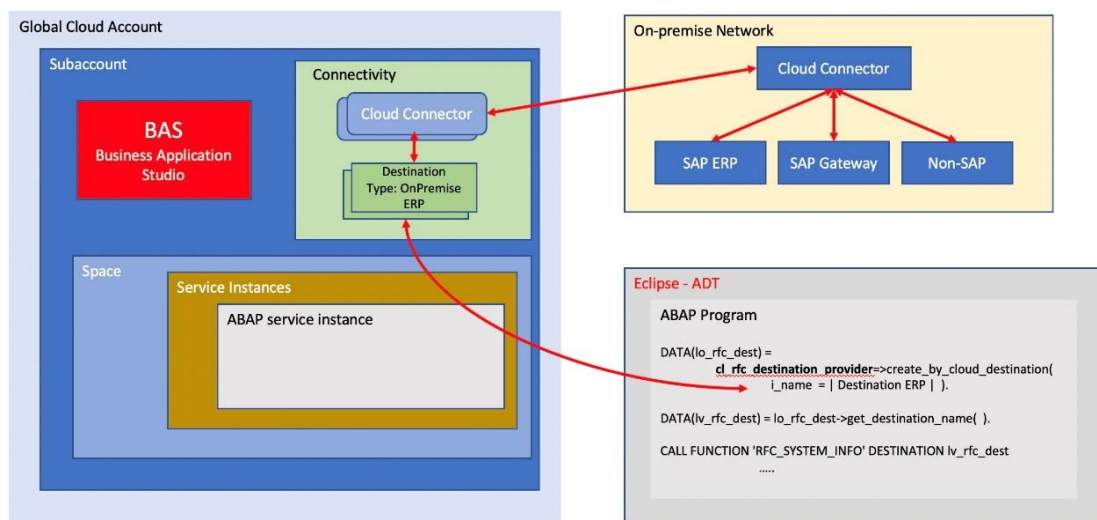


Figure 9 : Remote Function Call from ABAP cloud⁹

⁹ Source : <https://blogs.sap.com/2020/09/03/call-a-remote-function-module-rfc-from-sap-cloud-platform-abap-environment-update-09-2020/>

- External services : utiliser des services externes représente le cœur de toutes les applications dans le cloud et est l'une des motivations principales pour passer à un système cloud. Les services externes proposent une réutilisabilité, garantissent une équité, évitent de devoir enregistrer des données sensibles dans une base de données et intègrent les accès aux outils de développement. Ces services externes sont configurés et gérés dans le Cloud Foundry Environment (Obert, 2018).

5.5 Compétences à avoir

Selon Daniel Lahl, Global Vice President of Product Marketing chez SAP, l'environnement de développement pour les développeurs ABAP et les méthodologies de programmation ABAP ne changent pas avec la migration vers le développement en cloud. Il est néanmoins important d'apporter quelques connaissances nouvelles afin de profiter pleinement de l'environnement cloud (Parizo, 2019).

Pour le développement classique d'applications SAP, le modèle de cascade est le plus souvent utilisé (Parizo, 2019). Ce modèle est un processus linéaire qui consiste à franchir les étapes de développement, de sa conception à la phase de production. On ne peut passer à la prochaine étape que si toutes les étapes précédentes ont été validées. Voici les phases : planification, définition, conception, mise en œuvre, validation et mise en service (Augsten, 2018). En outre, le développement dans le cloud exige par sa nature des compétences en méthodes agiles et DevOps. Ces dernières sont nécessaires pour favoriser une livraison et intégration continue des services.

5.5.1 Méthode agile

La première phase de la méthodologie agile consiste à définir les rôles. Nous retrouvons en premier lieu l'utilisateur ou le client qui va utiliser le programme final, puis le Product Owner (PO) qui est le représentant des clients. Il va évaluer les idées et les connaissances afin d'avoir une vision du produit.

Le PO est responsable de la définition du projet et travaille en collaboration avec l'équipe de développement pour atteindre les objectifs du projet. Dans cette collaboration, le PO découpe l'intégralité du projet en plusieurs parties, selon son ampleur. À l'intérieur de ces parties, appelées itérations ou sprints, on retrouve la tâche à effectuer, l'objectif à atteindre en réalisant la tâche et sa priorité. Tous ces points doivent être discutés avec l'équipe de développement pour éviter toute confusion. À la fin de chaque itération, une séance est organisée, durant laquelle le PO décide si les différentes tâches sont acceptées ou non.

En agile, le rôle de l'équipe de développement et de ses membres est lié à d'avantage de responsabilités par rapport à d'autres méthodes de travail plus classiques. En effet, les équipes sont multidisciplinaires et composées de personnes avec des connaissances différentes. Puisque la priorité est mise sur le déploiement de software fonctionnel, l'équipe doit réaliser des logiciels full stack (« backend » et « frontend »). La gestion de base de données, la business logique et l'interface graphique font partie du développement au sein de l'équipe de développement. Pour pouvoir coordonner toutes ces parties, la communication est primordiale au sein de l'équipe. Des meetings réguliers sont nécessaires afin de s'assurer que toute l'équipe s'aligne sur le même objectif, de déterminer les tâches de chacun et définir la manière dont le logiciel est développé. Une équipe de développeurs agiles peut comporter des ingénieurs de contrôle de qualité, des ingénieurs de base de données, des designers, des analystes et bien d'autres – tout dépend du type de logiciel (Sacolick, 2020).

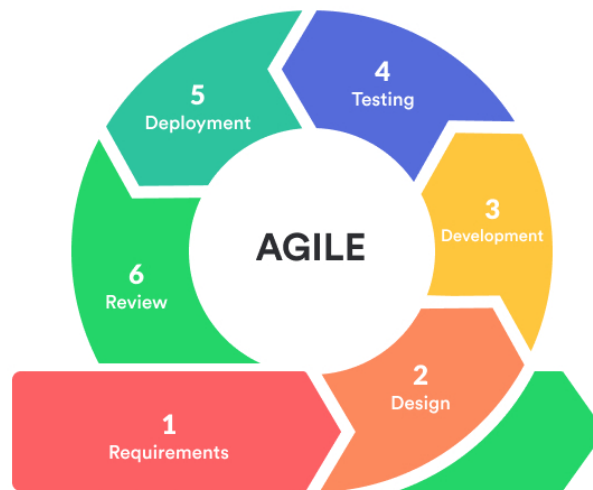


Figure 10 : Méthode agile pour le développement¹⁰

¹⁰ Source : <https://www.infoworld.com/article/3237508/what-is-agile-methodology-modern-software-development-explained.html>

5.5.2 DevOps

Le mot DevOps est composé du mot développement et opérations, signifiant la collaboration et l'approche commune des développeurs et de l'équipe de gestion des opérations. Dans un sens plus large, le DevOps est une philosophie qui promeut une meilleure communication et collaboration entre les différentes équipes travaillant sur un projet.

DevOps n'est pas une technologie et son environnement est généralement composé de méthodologies communes. Parmi ces dernières nous retrouvons des outils de Continuous Integration (CI), des outils supportant le DevOps comme des systèmes de monitoring en direct, des systèmes de management d'erreurs ou encore des outils de management et de gestion des collaborateurs. Enfin, nous avons le « cloud computing », les micro-services et différents conteneurs implémentés en parallèle avec la méthodologie DevOps.

DevOps mène à améliorer le travail tout au long du cycle de développement. Le processus du DevOps peut être vu comme une boucle infinie comprenant les étapes suivantes : planification, réalisation, assemblage, test, versionnage, déploiement, opérationnalisation, surveillance. Idéalement avec la méthodologie DevOps, une équipe de développement code un programme qui répond parfaitement aux attentes du client, qui peut être déployé sans perte de temps et qui fonctionne de manière optimale dès son premier lancement. Pour aligner le rendu aux attentes du client, les développeurs et le client communiquent régulièrement afin de s'assurer que tout se passe comme prévu. Pour éviter des pertes de temps, les équipes de développement utilisent des pipelines ou d'autres systèmes d'automatisation permettant de déplacer le code d'une étape de développement et déploiement à une autre. Les équipes de développement supervisent immédiatement les changements et s'assurent que les versions déployées correspondent aux attentes du client.

Afin de déployer du bon code, les adhérents DevOps utilisent des conteneurs ou d'autres méthodes pour s'assurer que le programme se comporte d'une façon homogène durant toutes les phases : du développement aux tests et jusqu'à la production. Dans toutes les étapes, les changements sont déployés individuellement pour pouvoir retracer d'éventuels problèmes (Courtemanche, Mell, & Gillis, 2020).

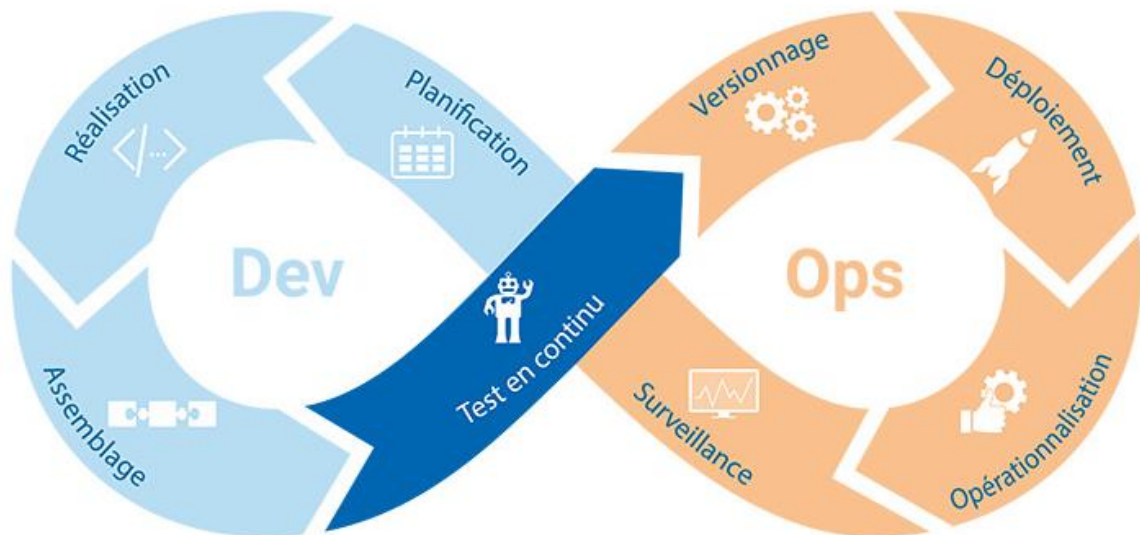


Figure 11 : Méthode DevOps pour le développement¹¹

5.6 Coûts

SAP met à disposition de ses clients un outil permettant d'estimer les frais liés à l'utilisation de la SCP.

Pour faire une estimation approximative, nous avons ajouté trois outils pour le développement ABAP : le ABAP environment pour le développement, le SAP Cloud Transport Management (TMS) pour gérer les transports et le SAP Workflow Management ou SAP Cloud Application Programming Model nécessaire pour le développement Agile ou DevOps.

Pour le ABAP environment, plusieurs options peuvent être paramétrées ; les heures de travail estimées sur l'environnement, la performance en Gigabytes et le nombre de « blocs » de performance. Le prix est indiqué pour une licence et par mois.

¹¹ Source : <https://appsconsultant.com/index.php/blog/107-blogue-francais/340-c-est-quoi-le-devops>

Price Details				
Metric	Unit Price per Hour	Hours	Quantity	Amount
Runtime in 16 GB Blocks	EUR 1,923	210	1	EUR 403,83
Persistence in 16 GB Blocks	EUR 0,549	210	4	EUR 461,16
				EUR 864,99 /MONTH

Figure 12 : Prix pour ABAP environment¹²

Pour le SAP Transport Management, la rapidité exprimée en Gigabytes peut être paramétrée.

				EUR 18,00 /MONTH
--	--	--	--	------------------

Figure 13 : Prix pour le SAP Cloud Transport Management¹²

Pour le SAP Workflow Management, le nombre d'utilisateurs doit être indiqué.

Price Details				
Metric	Ranges	Unit Price per Month	Fixed Fee	Amount
Users	Up to 100	EUR 300,00	✓	EUR 300,00
	Up to 500	EUR 3,00		
	Up to 1000	EUR 2,70		
	Up to 5000	EUR 2,30		
	Up to 20000	EUR 2,00		
	Up to 50000	EUR 1,75		
	Up to 100000	EUR 1,50		
	From 100000	EUR 1,25		

Figure 14 : Prix pour le SAP Workflow Management¹²

Attention : pour l'utilisation du SAP Workflow Management, trois outils supplémentaires sont nécessaires ; la SAP BTP Cloud Foundry Runtime (70 EUR / mois), le SAP Business Application Studio (13 EUR / mois) et le SAP Launchpad Service (125 EUR / mois).

Au total, pour un environnement de travail cloud pour un utilisateur, il faut compter au minimum 1400 EUR par mois. Ce montant ne représente pas une estimation concrète, elle donne une idée générale des coûts liés au SCP.

5.7 Sécurité

De nos jours, la protection des données et la sécurité des infrastructures informatiques jouent un rôle important. Avec l'arrivée de systèmes cloud PaaS, l'aspect sécuritaire a joué un

¹² Source : <https://www.sap.com/swiss/products/business-technology-platform/estimator-tool.html>

rôle majeur dans l'élaboration de ces services.

Lors de l'utilisation de services cloud, la responsabilité est partagée entre l'entreprise qui met à disposition le service et le client. En effet, le client est responsable pour la sécurité des applications, des données et des accès utilisateurs. L'entreprise du service de son côté, est responsable pour la sécurité du service et de l'infrastructure technique (McAfee, 2021).

SAP étant une entreprise d'une grande envergure sur le marché, la sécurité est prise très au sérieux. Qu'il s'agisse d'exécuter un programme à travers les centres de données SAP ou de développer des programmes en toute sécurité, les clients de SAP sont assurés de toujours pouvoir compter sur les meilleurs standards de sécurité (SAP, SAP Cloud Platform Security: Safeguarding Your Business, 2017).

Selon Gerold Hübner, Chief Security and Compliance Officer pour SAP Cloud Platform, la sécurité des systèmes SAP est une variable dynamique, qui demande à être ajustée constamment (Hübner, 2017). La stratégie de sécurité chez SAP est fondée sur trois piliers principaux :

- « Secure products » : le focus se trouve sur l'intégration de sécurité dans les applications tout en proposant la plus haute protection possible pour le contenu des applications et les transactions (Hübner, 2017).
- « Secure operations » : SAP veut proposer un framework de sécurité simple d'utilisation permettant de protéger les données et ses accès, la maintenance de « patch » de sécurité, la surveillance de la sécurité et la gestion d'incidents (Hübner, 2017).
- « Secure company » : en offrant tous les standards de sécurité possibles à ses clients, SAP développe une culture de protection sans faille (Hübner, 2017).

5.7.1 Utilisation sécurisée

Avec la migration vers la SCP, les programmes ne tournent plus en local et les données utilisées sont stockées dans des centres de données SAP distribuées dans le monde entier. Ces centres de données sont soumis à des hauts standards de sécurité industrielle, certifiés ISO (International Organization for Standardization) 27001 et conformes au SOC1 et SOC2 (SAP, SAP Cloud Platform Security: Safeguarding Your Business, 2017). L'ISO 27001 est un certificat international pour la sécurité des informations valable pour des entreprises privées ou publiques. Il décrit les normes pour la mise en place, l'élaboration, le fonctionnement et l'optimisation d'un Système de Management de la Sécurité de l'Information (SMSI) qui est obligatoire pour obtenir la certification. Une fois le certificat remis à l'entreprise, cette dernière

est régulièrement soumise à des contrôles (Luber & Schmitz, Was ist ISO 27001?, 2017). Les backups de données sont physiquement séparés mais co-localisés dans la même région que le centre de données SAP primaire certifié. Les données des entreprises sont toujours maintenues séparément des données de ses clients, l'accès administrateur aux données via l'extérieur est fortement limité. Toutes les données, « au repos » ou en utilisation, sont cryptées avec les plus hauts standards possibles (SAP, SAP Cloud Platform Security: Safeguarding Your Business, 2017). Tous les accès par des employés de SAP sur des données de leurs clients sont retenus sans exception dans un protocole. À savoir que ces accès sont très limités et les accès aux applications et aux infrastructures sont interdits. Le client garde à tout moment le contrôle sur ses données, les protocoles mentionnés auparavant sont disponibles et peuvent être consultés librement. SAP mise sur une haute transparence afin d'établir une relation de confiance avec ses clients (Cesljar, 2019).

SAP offre différents mécanismes de sécurité pour s'assurer qu'uniquement des utilisateurs approuvés ont accès aux applications de la SCP.

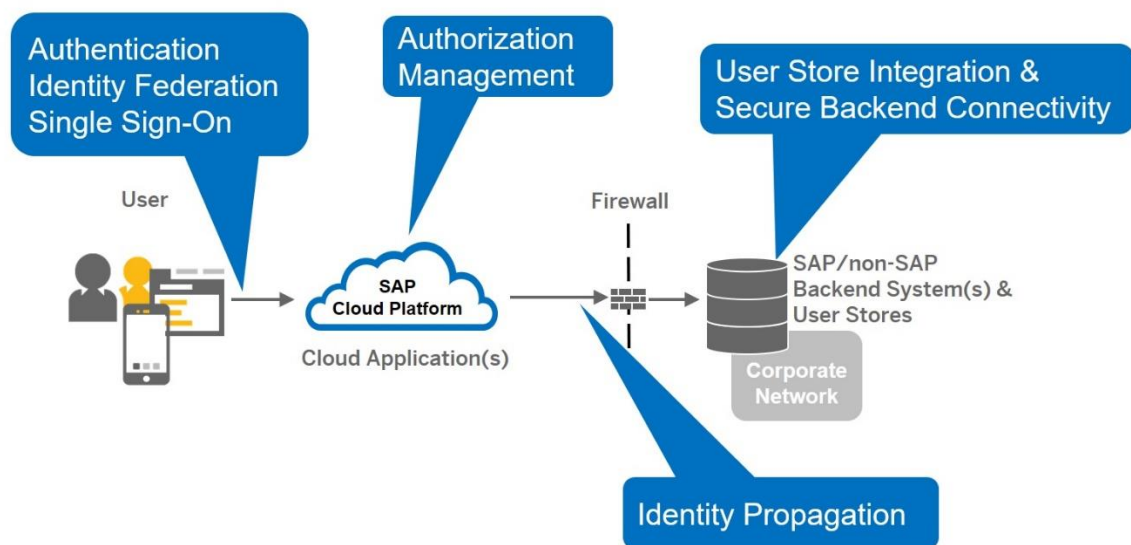


Figure 15 : Aperçu sécurité sur la SCP¹³

¹³ Source : <https://blogs.sap.com/2017/08/04/sap-cloud-platform-security-safeguarding-your-business/>

5.7.2 Développement sécurisé

Avec le service d'authentification de la SCP, les entreprises peuvent fournir à leurs employés, clients et partenaires des accès simples et sécurisés au cloud pour pouvoir gérer les processus, les applications et les données qu'ils requièrent. Cette authentification à double facteur, qui, en plus du mot de passe habituel, exige un code de sécurité unique pouvant être reçu par e-mail, SMS ou via un logiciel d'authentification, garantit un haut standard d'authentification.

Le SAP Cloud Platform Identity Provisioning offre une approche simple et à bas prix au management d'identités dans le cloud. En plus de rendre accessible le cloud aux utilisateurs locaux, il est possible de gérer les accès aux services cloud et aux fonctions de développement de façon individuelle. Ces accès sont gérés par des polices de sécurité, sont mappés entre les groupes d'utilisateurs et ses utilisateurs en local et les rôles dans le cloud. La SCP supporte également l'affectation des rôles de manière dynamique. Quand les rôles en local ou dans le cloud changent, le Security Assertion Markup Language (SAML) change automatiquement. Cette pratique garantit que les accès sur les programmes et les données dans le cloud sont propres aux rôles et aux utilisateurs. Il est également possible de programmer des affectations de rôles via des API.

Source Systems

On-premise:

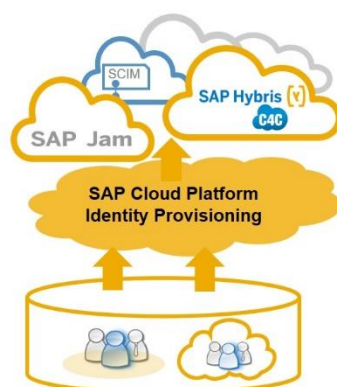
- SAP NetWeaver Application Server for ABAP
- Microsoft Active Directory

Cloud:

- SAP SuccessFactors
- SAP Cloud Platform Identity Authentication

Generic:

- SCIM-enabled solution
- LDAP Server



Target Systems

- SAP Cloud Platform
- SAP Cloud Platform Identity Authentication
- SAP Hybris Cloud for Customer
- SAP Jam
- SAP Concur
- Google G Suite
- SCIM-enabled solution
- Cloud Foundry UAA

Figure 16 : Accès sécurisé à la SCP¹⁴

¹⁴ Source : <https://blogs.sap.com/2017/08/04/sap-cloud-platform-security-safeguarding-your-business/>

En plus de tous les standards et mesures de sécurité mentionnés, SAP mise également sur l'Open Authorization (OAuth), qui est une mesure préventive de sécurité remplaçant le nom d'utilisateur et le mot de passe par un « token ». La SAP Cloud Platform utilise le protocole OAuth 2.0 qui est un moyen fiable de protéger des API et d'autres ressources. Le OAuth communique avec le SAML qui authentifie l'utilisateur et qui fournit un code d'identification. Le client se connecte avec le service OAuth SAP Cloud Platform où le « token » d'accès est généré. Depuis là, l'authentification est évaluée puis validée (SAP, SAP Cloud Platform Security: Safeguarding Your Business, 2017).

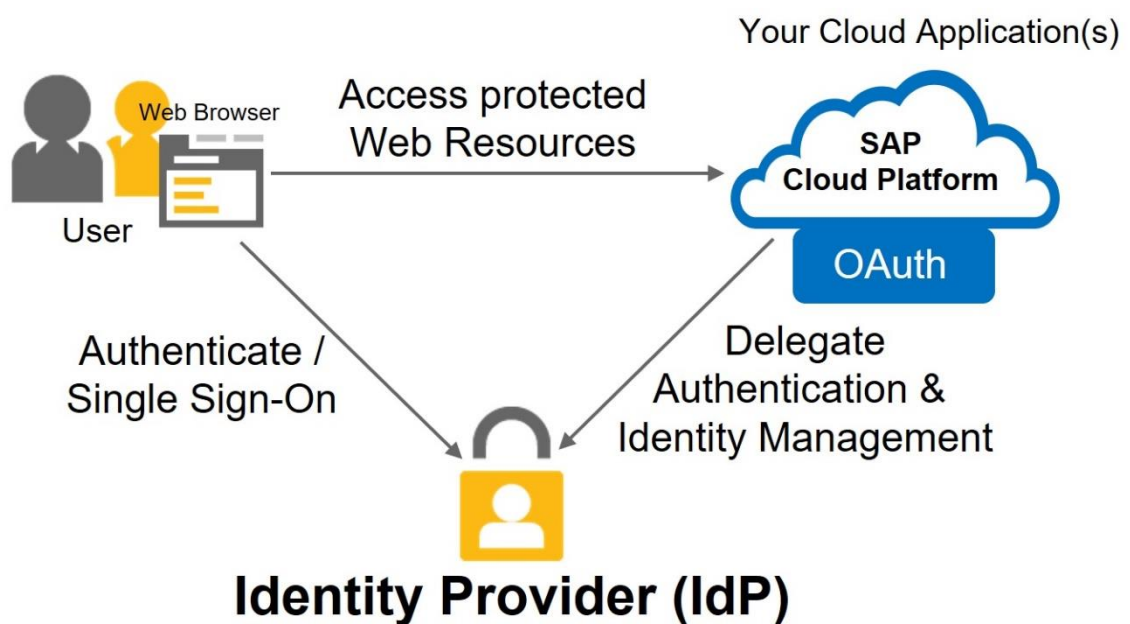


Figure 17 : OAuth avec SCP¹⁵

5.8 abapGit et Git vs Transport

Comme base d'abapGit la plateforme open-source GitHub a été utilisée. Elle permet aux développeurs de gérer leur code, de le partager, modifier et publier. La simplicité et la collaboration améliorée sont au cœur de cette technologie.

Contrairement à d'autres langages de programmation, il faut un Serveur SAP pour pouvoir exécuter des programmes codés en ABAP. Lors du développement en local, il n'est donc pas

¹⁵ Source : <https://blogs.sap.com/2017/08/04/sap-cloud-platform-security-safeguarding-your-business/>

nécessaire d'avoir un système comme abapGit puisque tous les développeurs ont accès aux programmes en local (Wulff, 2018). Mais quel est donc l'intérêt d'abapGit ?

5.8.1 SAP Transport

Les transports sont intégrés au système SAP et permettent de transporter des données d'un système de développement SAP vers un autre.

Les ordres de transport sont des « collections » de changements effectués dans le système de développement. Ils contiennent également les informations concernant le type de changement, la raison du transport, la catégorie ou encore le système ciblé concerné. Les ordres de transport sont composés de tâches. Ces tâches sont enregistrées dans l'ordre de transport comme des fichiers dans un dossier. L'ordre de transport peut uniquement être transporté quand toutes les tâches à l'intérieur ont été complétées, supprimées ou libérées. Les tâches de changement représentent une liste d'objets qui ont été changés par un utilisateur. Chaque tâche peut être assignée à un seul utilisateur. Les tâches ne peuvent pas être transportées ou alors seulement dans leur intégralité à l'intérieur d'un ordre de transport.

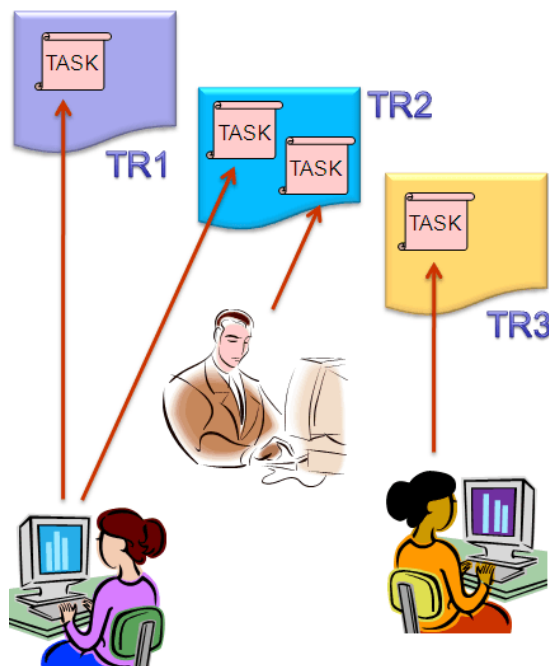


Figure 18 : Exemple de distribution de tâches¹⁶

¹⁶ Source : <https://www.guru99.com/what-is-transport-request-how-to-importexport-it-check-logs.html>

C'est le responsable du projet qui crée un ordre de transport et qui assigne les différentes tâches aux membres du projet. De plus, il est le propriétaire de l'ordre de transport et il a le contrôle sur tous les changements qui sont enregistrés dans celui-ci. Il est donc le seul à pouvoir transporter l'ordre (guru99, What is SAP Transport Request? How to Import/Export TR, 2021).

5.8.2 abapGit

Il faut savoir qu'abapGit a spécialement été développé pour le développement ABAP. AbapGit est donc adapté aux besoins du développement ABAP. Avec abapGit, une connexion entre la gestion de versions Git et le serveur d'applications ABAP est fortement simplifiée. Tous les objets de développement locaux ABAP sont déposés dans un dépôt Git, sans avoir besoin d'un serveur Git (Kriegbaum, 2020). Dans le contexte de notre travail de bachelor, ce dépôt et cette pratique nous permettent de migrer le développement ABAP local vers le cloud.

L'installation d'abapGit est très simple et rapide, elle ne laisse pas de traces dans le développement et offre plusieurs avantages :

- Il existe à tout moment une version à jour du code source sur la branche master.
- Il est possible à tout moment de revenir sur une version plus ancienne.
- La collaboration est simplifiée, plusieurs développeurs indépendants les uns des autres peuvent travailler sur le même code source et ils ont la possibilité de documenter leur code directement sur Git.
- Il est possible d'utiliser le « Continuous Integration » avec ses pipelines, qui permettent d'automatiser le transfert de code pour des tests par exemple.

5.8.3 Git-enabled Change and Transport System

Pour adapter le processus de développement au cloud et aux méthodes agiles ou DevOps, SAP a présenté sa solution Git-enabled Change and Transport System (gCTS). L'abapGit fait partie de cette solution. Les développeurs continuent à travailler dans le ABAP Workbench (SE80) ou dans le ABAP environnement dans le cloud, continuent à enregistrer leurs changements avec des tâches et à libérer les tâches et des ordres de transport. La libération des ordres de transport ne veut pas dire que l'ordre de transport est ajouté aux systèmes de transport de notre serveur local. Libérer un ordre de transport avec le gCTS est en réalité un « commit » vers notre dépôt Git. Les objets qui ont été créés ou changés sont ajoutés comme fichier au dépôt Git. Plusieurs « runtimes » ABAP peuvent être connectés au même dépôt, permettant de fusionner des programmes ABAP (Spiegel, 2019).

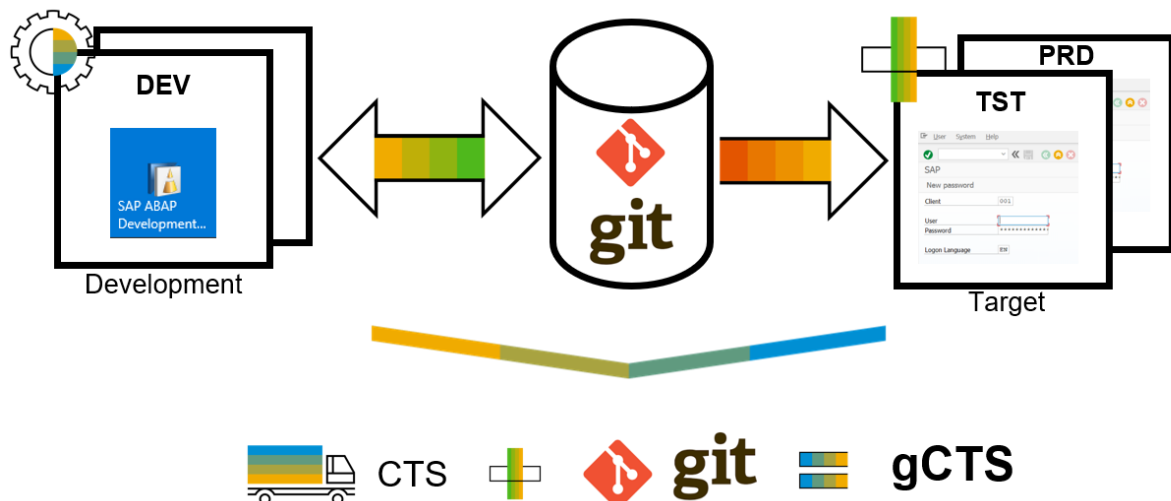


Figure 19 : Environnement gCTS¹⁷

5.9 Développer et tester

Avec l'outil SAP Cloud Application Programming Model (CAP), le développeur est guidé à travers le développement d'applications. Un « golden path » avec des « best practices » et des outils de développement sont recommandés au développeur. Il offre un modèle de programmation « end-to-end » cohérent et comporte des librairies et des APIs pour le développement « fullstack ». Par exemple, lorsqu'un CDS est défini, un framework peut gérer automatiquement la connexion avec SAP HANA.

En suivant le modèle, l'environnement de test peut être utilisé pour effectuer des tests automatiques ou manuels. Cette pratique a pour objectif de remplacer le SAP Change Request Management (ChaRM), utilisé pour administrer les activités comme le design ou les tests en parallèle du développement ABAP local (Zarske, 2019).

Cet outil a spécialement été développé pour être compatible avec DevOps.

¹⁷ Source : <https://blogs.sap.com/2019/11/14/gcts-is-here/>

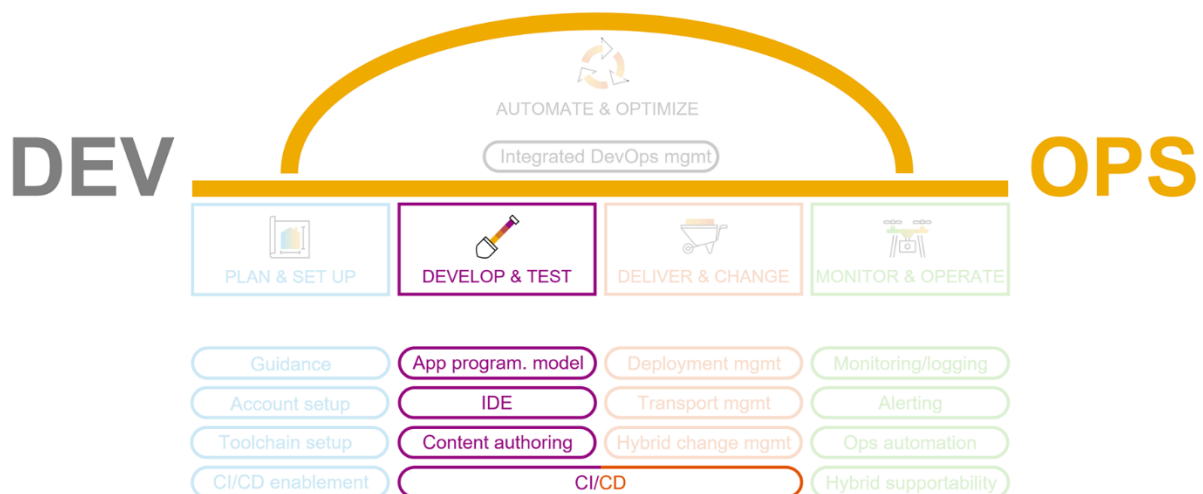


Figure 20 : Cycle de développement DevOps¹⁸

La possibilité de continuer l'utilisation de SAP ChaRM reste. Nous pouvons donc intégrer le SAP Cloud Platform TMS dans le SAP ChaRM ou Quality Gate Management (QGM). L'utilisation du TMS est plutôt simple puisque l'utilisateur est guidé via un « Wizard » pour l'exportation. Mais pour que ce scénario fonctionne, quelques critères doivent être remplis (Stevens, 2020) :

- SAP Solution Manager 7.2 avec SPS10 implémenté
- ChaRM ou QGM doivent être disponibles et configurés

L'intégration du TMS dans le système ChaRM se fait en plusieurs étapes :

1. Préparation du SAP Cloud Platform Transport Management. Il faut entre autres créer une instance sur service avec une clé d'accès.
2. Préparation dans le SAP Solution Manager. Il faut créer une connexion RFC qui permet au SAP Solution Manager de communiquer avec le Transport Management. Cette connexion est établie grâce à la clé générée dans le SAP Cloud Platform Transport Management.
3. Création d'un cycle de changements dans le TMS. C'est ici que vont apparaître tous les changements et que l'on peut apercevoir le cycle de développement d'une application.

¹⁸ Source : <https://blogs.sap.com/2019/12/12/devops-with-sap-cloud-platform-develop-test/>

4. Création d'un document dans le cycle de changements. Ce document va comporter toutes les informations relatives à un programme ABAP.
5. Création de contenu pour le transport. Il existe plusieurs possibilités d'ajouter du contenu à un document. La première option est celle d'automatiser l'import du code ABAP dans le document via CI/CD pipelines. Une autre option est d'exporter le code directement depuis le compte de développement cloud en utilisant le Solution Export Wizard. Cette option permet d'exporter le code directement depuis l'environnement ABAP sur la SCP.
6. Ajouter un ordre de transport au document. Cela permet de regrouper les changements et son code dans un ordre de transport.
7. Le programme est maintenant disponible sur SAP ChaRM et prêt à parcourir les différentes étapes jusqu'à la production.

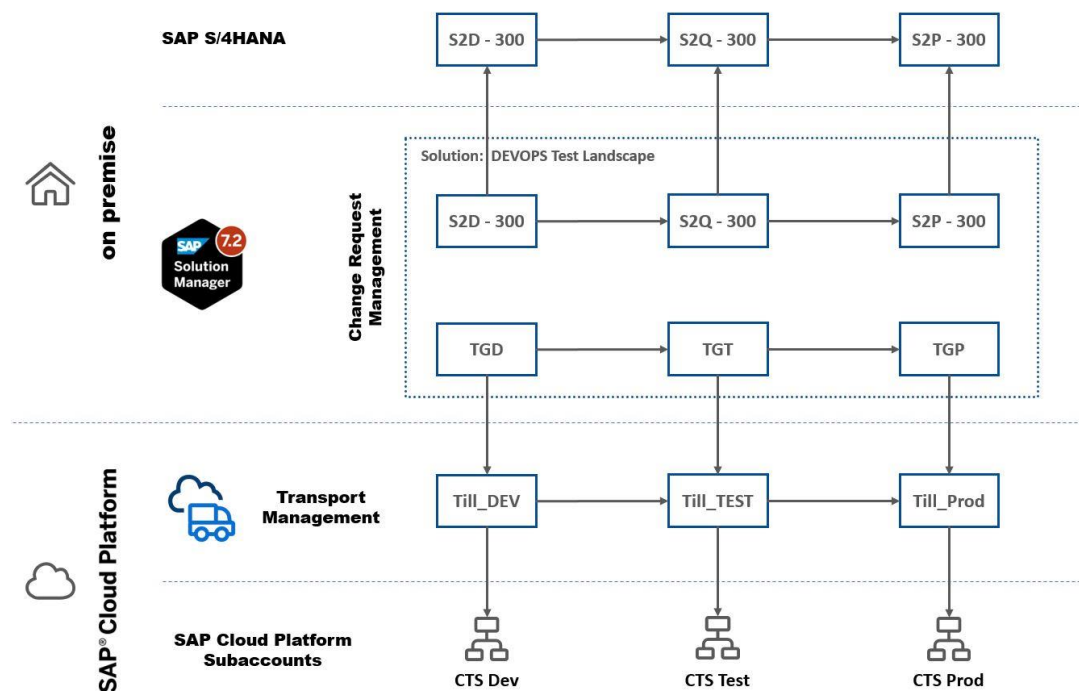


Figure 21 : Aperçu environnement de test¹⁹

¹⁹ Source : <https://blogs.sap.com/2020/04/27/how-to-use-the-integration-of-sap-cloud-platform-transport-management-into-sap-solution-manager-change-request-management-and-quality-gate-management/>

5.10 Outils de développement

5.10.1 SAP BTP Cockpit

SAP BTP Cockpit est une interface d'administration en ligne permettant l'accès à de nombreuses fonctionnalités pour configurer et gérer les applications, services et comptes. Cet outil administratif facilite la gestion des ressources, des services, de la sécurité et nous permet d'administrer les applications et services cloud.

Avec le cockpit, une vue générale sur les applications et services SCP est possible et la gestion des comptes est également dirigée à cet endroit.

Pour pouvoir utiliser le cockpit, la création d'un compte est nécessaire. Pour la vérification de ce dernier, un téléphone mobile doit être utilisé pour la double authentification. Le cockpit peut être paramétré selon les besoins du client. Il est notamment possible de changer le langage, le thème ou encore les notifications (SAP, SAP BTP Cockpit, 2021).

L'utilisation du cockpit est faisable sur Chrome, Mozilla Firefox, Microsoft Internet Explorer v11 (fin de support sur ce browser en octobre 2021), Microsoft Edge et Safari (SAP, Feature Scope Description for SAP BTP, 2021).

5.10.2 Eclipse

Eclipse est un Integrated Development Environment (IDE) crée par IBM pour du développement en JAVA. Eclipse supporte aujourd'hui une grande variété de langages de programmation comme C/C++, PHP, JavaScript, Python, Rust, et cetera. Eclipse est capable de tourner sur une grande partie des systèmes d'exploitation comme Windows, Mac ou encore Linux. L'outil est constamment mis à jour et bénéficie d'un soutien de grandes compagnies comme Oracle, SAP, Google, Airbus ou encore BMW. L'IDE peut être téléchargé gratuitement (Minh, 2020).

Pour pouvoir utiliser Eclipse, une version récente de Java doit être installée sur la machine. Java est un langage de programmation utilisé pour développer des programmes complets et les programmes développés en Java nécessitent l'installation de Java comme c'est le cas pour Eclipse.

Pour le développement ABAP sur Eclipse, la version 2020-12 (4.18) ou 2021-03 (4.19) est requise. Pour la version Java, le Java Runtime Environment (JRE) version 11 (64-Bit) ou plus est exigé et pour le système d'opération Windows, « Microsoft Visual C++ 2013 (x64) » doit être installé pour la communication avec le « backend » (SAP, ABAP Development Tools,

2021).

Les ABAP Development Tools (ADT) permettent aux utilisateurs d'Eclipse de pouvoir profiter pleinement d'un développement avec des transactions traditionnelles du SAP GUI mais aussi de fonctionnalités et le confort d'une plateforme de développement moderne. L'intégration de ces outils dans Eclipse permettent pour l'instant d'avoir une alternative gratuite au SAP Business Application Studio, qui est une solution propre à SAP et qui permet de développer dans le cloud. Tout comme le Web IDE propre à SAP qui se trouve encore dans sa phase de maturité. Les ADT sont parfaitement stables et constamment mis à jour. Les ADT permettent donc de développer des applications SAP sur une autre plateforme que le SAP GUI (Ritzau, 2021).

5.10.3 SAP Business Application Studio

Le SAP Business Application Studio est un service de la SCP qui offre un environnement de développement moderne, spécialement conçu pour la création d'applications SAP. Accessible via le cloud, il propose une interface similaire à d'autres IDE et permet de développer des applications ABAP, des interfaces SAP Fiori, des applications mobiles et bien plus.

Il est possible d'ajouter un ou plusieurs environnements de travail, permettant à plusieurs utilisateurs de travailler sur le même studio. Il est compatible avec Git et propose un environnement de test intégré (SAP, What is SAP Business Application Studio?, 2021).

Contrairement à Eclipse par exemple, cette solution est payante et actuellement encore en phase de développement. Raison pour laquelle beaucoup de développeurs optent plutôt pour l'utilisation d'Eclipse avec les outils ADT.

5.10.4 GitHub

GitHub est un service d'hébergement Git, offrant des fonctionnalités supplémentaires au Git classique. Contrairement à Git, dont l'utilisation se fait uniquement via un outil de commandes, GitHub propose une interface graphique web. D'autres outils comme la gestion d'accès sont également disponibles sur la plateforme (Finley, 2012).

L'intérêt principal de l'utilisation de GitHub est de pouvoir créer des dépôts appelés « repository ». C'est un espace de stockage où les projets sont déposés. Le format de fichier pouvant être déposé peut contenir du code, du texte brut, des images et bien plus. À noter qu'un dépôt peut être public ou privé. Dans le premier cas, son contenu est visible par tous. Inversement, le dépôt privé a quant à lui un accès restreint et il est transmis aux personnes invitées seulement. Cette seconde alternative est payante.

La notion de « branching » permet aux développeurs de travailler sur différentes versions d'un projet en même temps. L'ajout d'une nouvelle fonctionnalité par exemple peut se faire sur une nouvelle branche afin d'éviter d'intervenir sur la branche principale, appelée « Master Branch ».

Le « commit » permet de sauvegarder des changements et il est accompagné d'un message. Ce dernier permet de décrire la nature du changement et son contenu, afin de créer un historique des changements aidant les autres collaborateurs à comprendre les modifications.

Avec l'aide du « pull », le contenu d'un dépôt, la dernière version, est chargé sur l'ordinateur. Il permet de « télécharger » le contenu d'un dépôt. Le push quant à lui permet de faire l'inverse : déposer le contenu du commit dans le dépôt. Lors de cette opération, il peut y avoir des conflits, par exemple, lorsque deux développeurs ont apporté des modifications au même fichier. Il y a plusieurs possibilités pour régler ce conflit : accepter les modifications d'un des développeurs ou « merger » les fichiers. Le « merge » n'est rien d'autre qu'une fusion du contenu d'un fichier afin de garder toutes les modifications. C'est aux développeurs de décider quelle option choisir lors du « pull » ou du « push » (Johari, 2021).

6. Avantages et désavantages du SCP

Avant de prendre la décision, il est important de prendre en considération les avantages et désavantages d'une migration vers SCP. Avant de faire son choix, il y a un grand nombre de points à évaluer pour un système complexe comme SAP.

Avantages	Désavantages
Evolutivité	Coûts
Sécurité et conformité	Connectivité
Intégration	Comptabilité ABAP
Encourage les méthodes Agiles et DevOps	Restrictions technologiques
	Complexité

Figure 22 : Récapitulatif avantages et désavantages²⁰

6.1 Avantages

6.1.1 Evolutivité

Plusieurs serveurs sont souvent nécessaires pour travailler et développer dans un environnement SAP : des serveurs d'applications, des serveurs de stockage et d'échange de données et des serveurs qui gèrent les bases de données. Ces serveurs nécessitent souvent des changements et des modifications pour s'adapter à l'augmentation des données et des besoins des utilisateurs (Chouffani, 2019). Les serveurs locaux peuvent être améliorés avec du nouveau matériel informatique souvent coûteux et complexe à installer, requérant ainsi des moyens financiers conséquents et de bonnes connaissances en hardware et en gestion de serveurs.

La solution cloud quant à elle se charge de gérer l'infrastructure informatique, les serveurs ne se trouvant plus en local chez l'utilisateur. Les utilisateurs du cloud peuvent utiliser la flexibilité du service pour rapidement ajouter des serveurs et de l'espace de stockage en quelques clics via le cockpit (Chouffani, 2019).

²⁰ Source : Donnée de l'auteur

6.1.2 Sécurité et conformité

Avec l'augmentation significative des cyberattaques ces dernières années, beaucoup d'entreprises ont dû investir dans leur sécurité et la protection des données. La mise en place d'une bonne sécurité informatique peut être coûteuse et il faut faire appel à des professionnels du domaine.

Avec la migration sur le cloud, beaucoup de ces aspects sécuritaires sont couverts directement par SAP. SAP est une entreprise avec une réputation sans faille qui se doit de proposer la meilleure sécurité avec les plus hauts standards à ses clients. Les différentes certifications et labels prouvent cet engagement (Chouffani, 2019).

Selon Erica Kastner, experte en cybersécurité, la sécurité cloud est plus élevée puisque les entreprises qui offrent des services Software as a Product (SaaS) ou SaaS ont des employés qui gèrent uniquement la sécurité de ces services et qui sont donc des experts dans le domaine. De plus, les systèmes cloud possèdent souvent une intelligence artificielle qui permet d'identifier et éliminer des failles potentielles (Kastner, 2020).

6.1.3 Intégration

Un grand nombre d'outils SAP SaaS sont à disposition dans le cloud et peuvent être intégrés rapidement et facilement dans les applications. Parmi les SaaS les plus connus nous retrouvons par exemple S/4HANA Cloud qui est un service très performant et probablement indispensable dans un futur proche pour le développement ABAP.

L'écosystème SAP se dirige vers une uniformisation des services avec des outils centralisés, permettant à l'utilisateur de développer des applications facilement extensibles. De plus, l'intégration de solutions tierces est possible avec des API et permet d'utiliser des services non-SAP.

Avec l'arrivée du SCP, SAP a présenté plusieurs outils d'intelligence artificielle et « machine learning ». Ces derniers peuvent être intégrés dans les applications SAP (Lawton, 2018).

Comme déjà expliqué au préalable, avec SCP il est possible d'utiliser des outils de contrôle de version comme Git, permettant aux équipes de travailler en agile ou DevOps.

6.1.4 Encourage les méthodes Agiles et DevOps

Les méthodologies Agiles et DevOps ont rapidement attiré l'attention de l'industrie informatique et cela pour des bonnes raisons. Avec l'environnement de travail en constante évolution, la demande de solutions toujours plus rapides et l'importance du travail en équipe,

ces méthodologies ont su répondre à ces besoins.

En utilisant ces méthodologies, un déploiement rapide de fonctionnalités permet aux utilisateurs de rapidement entreprendre des changements et des adaptations nécessaires. Les tâches sont clairement définies et la bonne communication dans les groupes de développement permet de supprimer des sources de conflits et d'erreurs. L'automatisation de certaines tâches libère de la place pour de l'innovation et une fois la méthodologie acquise par tout le monde, les cohérences permettent de travailler d'une façon rapide et efficace (Makadia, 2021).

6.2 Désavantages

6.2.1 Coûts

Bien que la réduction des frais de matériel informatique soit un avantage, le passage vers un modèle d'abonnement pour l'utilisation du service cloud peut s'avérer très coûteux si les souscriptions et les besoins de l'entreprise sont mal gérés. Contrairement aux serveurs en local, les serveurs cloud ne sont pas visibles pour l'utilisateur et il est donc important d'avoir un bon « monitoring » afin d'éviter des frais superflus. De plus, il est nécessaire de bien planifier la migration vers le cloud, ce qui peut engendrer des frais supplémentaires considérables.

6.2.2 Connectivité

Les services SCP sont accessibles via une connexion internet. Si cette connexion est interrompue, l'utilisation du cloud devient impossible. L'utilisation du cloud nécessite donc une bonne connexion internet, une mauvaise connexion pouvant entraîner des latences et des ralentissements du système (Chouffani, 2019).

Les clients du SCP dépendent fortement de SAP car avec la technologie, des pannes et des interruptions sont aujourd'hui normales. Lorsque ce scénario arrive, l'entreprise n'aura temporairement pas accès au service.

6.2.3 Compatibilité ABAP

Une grande majorité de solutions ABAP existantes ne peuvent être migrées vers le cloud sans modification. Bien que le langage ABAP en tant que tel est conservé pour le développement dans le cloud, plusieurs facteurs font que le fonctionnement du développement local et cloud nécessite des adaptations pour pouvoir fonctionner.

Plusieurs éléments utilisés dans le développement local ne sont plus supportés dans l'environnement cloud. Même des aspects plutôt simples et basiques requièrent des modifications et des changements. Les tables par exemple ne sont pas incluses dans les objets dit « whitelisted » dans le cloud, signifiant qu'un simple appel ABAP référant vers une table ABAP provoquera des erreurs de syntaxe. Pour éviter ce problème, il est nécessaire d'utiliser une API « whitelisted » afin d'accéder à une table par exemple. Cette pratique ajoute une complexité dans le développement cloud qui n'existe pas en local.

Le langage ABAP reste, son fonctionnement et les différentes approches d'utilisation dans le cloud changent. Les utilisateurs du ABAP dans le cloud doivent donc s'adapter à ce nouveau fonctionnement s'ils veulent utiliser l'environnement cloud (Snapp, 2021).

6.2.4 Restrictions technologiques

Dans l'environnement ABAP cloud, SAP Fiori est la seule technologie supportée pour la création d'interfaces graphiques. Cette restriction peut avoir un impact important puisque le framework Fiori se trouve toujours en phase de développement et n'est pas encore une technologie mature. Plusieurs utilisateurs témoignent de problèmes de performance avec Fiori.

Un autre aspect à ne pas négliger est l'acquisition de compétences. Deux choix s'offrent aux entreprises : recruter des employés formés en SAP Fiori ou faire apprendre la nouvelle technologie aux employés. Les deux options peuvent entraîner une perte de temps considérable.

Dans un même registre, le SAP GUI n'est plus utilisé pour le développement Cloud. Il faut donc s'orienter vers d'autres solutions comme Eclipse ou encore SAP Cloud Applications Studio. Ce dernier se trouve également en phase de développement et ne permet donc pas encore de développer des applications ABAP comme sur le SAP GUI.

Pour des habitués du SAP GUI, la migration vers le cloud signifie des changements dans la manière de travailler et demande l'apprentissage de nouvelles technologies et pratiques (Azmi, 2018).

6.2.5 Complexité

Avec la migration vers cloud beaucoup de nouvelles possibilités s'ouvrent aux développeurs, mais la SCP engendre une certaine complexité. Pour beaucoup de tâches anodines en local, une API ou un outil est essentiel pour le même fonctionnement dans le cloud. Plusieurs outils sont encore en phase de développement et nécessitent une

amélioration afin d'être compatibles avec le système local.

7. Quelle option choisir

Après avoir été intensivement confronté au sujet ces dernières semaines, il est temps pour nous de prendre position et d'évaluer si une migration du développement ABAP sur le cloud a du sens à l'heure actuelle.

Lors de nos premières recherches, la solution cloud de SAP donnait une image innovante et moderne. Avec son écosystème comportant divers services pour tout usage regroupé dans le cloud, SAP veut proposer une solution uniforme à ses utilisateurs. Les possibilités d'intégration de services internes ou externes aux applications ainsi que l'extension de programmes de façon rapide et efficace sont des éléments souvent relevés par SAP. Au lancement de la nouvelle solution, SAP a fortement misé sur une campagne marketing. Les nombreux articles et les vidéos, présentant la nouvelle solution cloud, que l'on peut trouver sur les différentes plateformes de SAP, ont tous un point commun plutôt frappant : le contenu ne montre que rarement des choses concrètes, des interfaces de développement par exemple dans le cloud. Il s'agit souvent d'une énumération de possibilités avec le cloud, accompagnée d'illustrations représentant des architectures d'implémentation. Il y a deux raisons principales pour expliquer ce phénomène ; la SAP Cloud Platform a officiellement été annoncé par SAP en 2012, période à laquelle les services cloud ont gagné en popularité. Néanmoins, les premières présentations des services cloud comme le ABAP environnement par exemple, apparaissent en 2016. Aujourd'hui, en 2021, SAP n'a toujours pas trouvé toutes les solutions nécessaires pour pouvoir entièrement remplacer les systèmes locaux. La SCP et ses outils de développement ne sont pas encore aboutis et se trouvent toujours en phase de développement. Il est donc difficile de montrer du contenu concret. SAP est fort pour le marketing et la séduction de ses clients, avec l'annonce du SAP BTP en 2021, SAP a encore une fois beaucoup montré sans vraiment aller en profondeur.

En approfondissant nos recherches, nous avons constaté que la migration n'allait pas être aussi simple que cela. Même si le ABAP reste, en passant sur le cloud, beaucoup de méthodes de travail et compétences acquises pour les utilisateurs vont changer. Il s'agit souvent de subtilités, mais il faut en tenir compte. L'environnement de travail familier SAP GUI n'est plus utilisé pour le développement dans le cloud. Pour les grands habitués de cette interface, ce changement peut s'avérer compliqué. Pour une migration complète, il faut compter plusieurs semaines de planification et exécution ce qui peut entraîner des perturbations au niveau du système local.

Un service pas abouti, une complexité non-négligeable, l'abandon de pratiques connues et fonctionnelles font qu'aujourd'hui nous ne pouvons pas recommander une migration complète du système local vers la SCP pour le développement ABAP. Le risque de se retrouver avec une solution qui ne fonctionne pas comme elle devrait, est trop élevé. Mais nous encourageons l'utilisation d'un système hybride dans une première étape. Il est possible de travailler avec le cloud et de garder le SAP GUI. En travaillant de cette manière, la migration se fait progressivement, permettant de minimiser les problèmes et de s'habituer au nouvel environnement. Avec le temps, SAP va déployer des solutions complètes et finies qui permettront de profiter pleinement du développement ABAP dans le cloud. Les technologies cloud sont modernes, il est à notre sens important d'aller de l'avant. La position de SAP est claire, ils envisagent de tout miser sur la nouvelle platform cloud, ce qui signifie qu'un jour le SAP GUI n'existera plus.

Nous aimerions également souligner la possibilité de pouvoir travailler en Agile ou DevOps. Le travail dans le cloud étant conçu pour ces méthodologies, il va fortement influencer la manière de travailler. Ces méthodologies modernes sont aujourd'hui un standard pour le développement et apportent plusieurs atouts majeurs. Comme pour la migration, il est également possible de travailler de façon hybride. Il n'est dans un premier temps pas nécessaire d'adopter pleinement une nouvelle méthodologie. Cette dernière peut être instaurée petit à petit, permettant à l'employeur et aux employés de s'y habituer.

L'État du Valais représente une organisation de taille importante, il est donc primordial d'effectuer des études de cas plus approfondies, notamment sur les aspects financiers, mais aussi pratiques, de la nouvelle technologie SCP. Il n'est pas possible dans le cadre de notre travail de bachelor de couvrir tous les aspects d'une migration vers le cloud. Il permet cependant d'avoir un premier aperçu et une vue globale, notamment sur les nouvelles technologies et possibilités.

8. Prototype projet ABAP cloud

Pour la réalisation de ce prototype, la version Eclipse 2021-06 (4.20.0) a été utilisée. Nous allons développer deux scénarios différents, un premier où nous allons créer un nouveau projet ABAP cloud avec différents éléments ABAP (Kempen, 2020) et un deuxième scénario avec une migration d'un programme ABAP sur SAP GUI vers le cloud (Flothkoetter, 2021).

8.1 Configuration de la plateforme cloud

Dans le cadre de ce travail de bachelor, un compte « trial » gratuit a été utilisé, permettant de tester les fonctionnalités de la SCP. La version complète « entreprise » payante permet d'utiliser toutes les fonctionnalités du SCP sans aucune restriction.

1. Se rendre vers le site officiel de SAP BTP pour créer un nouveau compte.
2. Lors de la création de compte, s'assurer que la bonne région est sélectionnée (Europe) ;

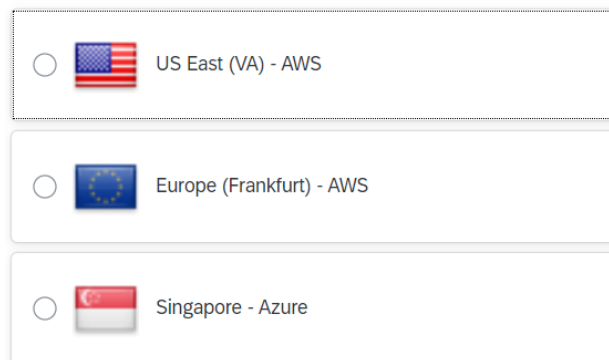


Figure 23 : Configuration SCP – Choix région²¹

3. Sur le cockpit, ajouter l'ABAP environment : *développement (dev)* -> *Service Marketplace > ABAP environment > Create*
4. Entrer les paramètres requis pour la création d'une instance ABAP : nom de l'instance, adresse électronique, puis créer l'instance.
5. Vérifier que l'instance a été créé ;

²¹ Source : Donnée de l'auteur sur SCP cockpit

Instance	Service	Plan	Credentials	Status	
TB - Yann Jollien	ABAP environment	shared		Created	... >

Figure 24 : Configuration SCP - Instance ABAP²²

6. Créer une clé d'accès pour pouvoir accéder à l'instance depuis Eclipse : *Nom de l'instance* > *Service Keys* > *Create* (un fichier JSON est généré)

8.2 Configuration d'Eclipse

Eclipse est l'IDE utilisé pour ce prototype, le développement avec les outils ADT sur Eclipse permet de créer des projet ABAP cloud.

Prérequis :

- Version supportée Eclipse : 2020-12 (4.18), 2021-03 (4.19), 2021-06 (4.20)
- Windows 10 ou Apple Mac OS X 10.14 ou plus
- JRE version 11 (64-Bit, LTS) ou plus (*)

1. Télécharger et installer Eclipse.
2. Une fois Eclipse installée, l'ouvrir et utiliser le menu : *Help* > *Install* > *New Software* ;

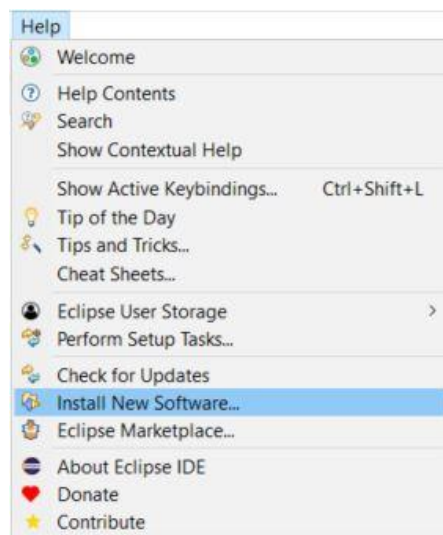


Figure 25 : Configuration Eclipse - Install New Software²³

²² Source : Donnée de l'auteur sur SCP cockpit

²³ Source : Donnée de l'auteur sur Eclipse

3. Insérer **https://tools.hana.ondemand.com/latest** dans la barre de recherche, cocher ABAP Development Tools puis « Next » 2x (accepter les termes de licence) et « Finish » ;

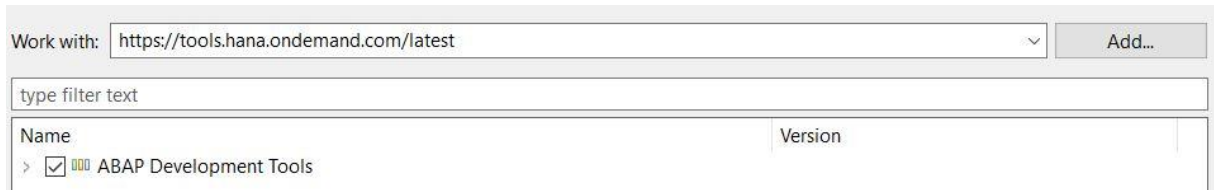


Figure 26 : Configuration Eclipse - Ajout des outils ADT²⁴

4. En cas de message warning, confirmer l'installation et accepter le certificat ;

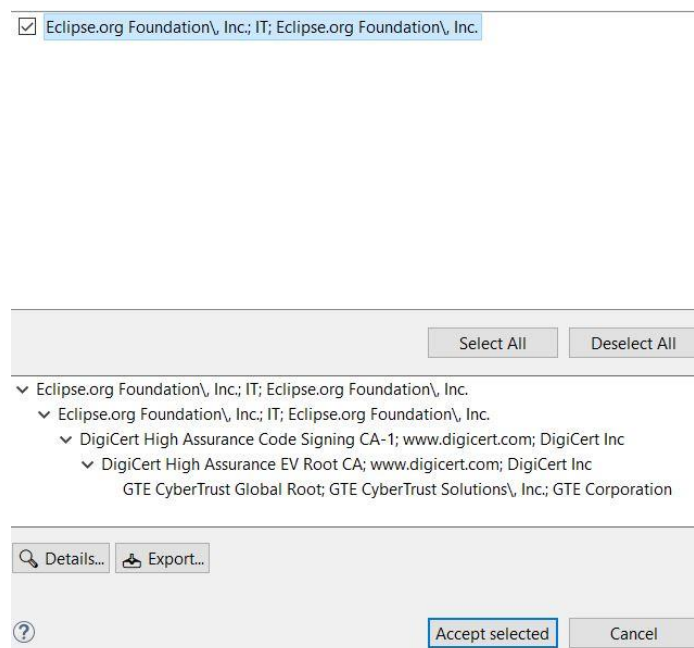


Figure 27 : Configuration Eclipse - Certificat²⁵

5. Une fois l'installation terminée, redémarrer Eclipse.

²⁴ Source : Donnée de l'auteur sur Eclipse

²⁵ Source : Donnée de l'auteur sur Eclipse

8.3 Nouveau projet ABAP

Pour pouvoir créer des projets ABAP sur Eclipse, il est nécessaire de changer la perspective JAVA en ABAP pour pouvoir accéder aux outils de développement.

1. Dans le menu Eclipse : *Window > Perspective > Open Perspective > Other ;*

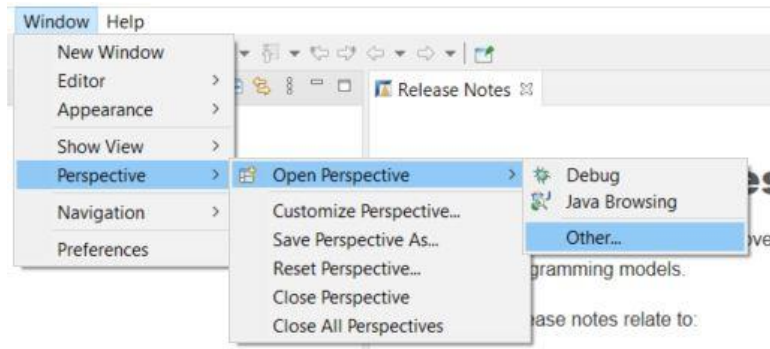


Figure 28 : Configuration Eclipse : Changement perspective²⁶

2. Dans la liste des perspectives, sélectionner « ABAP » ;

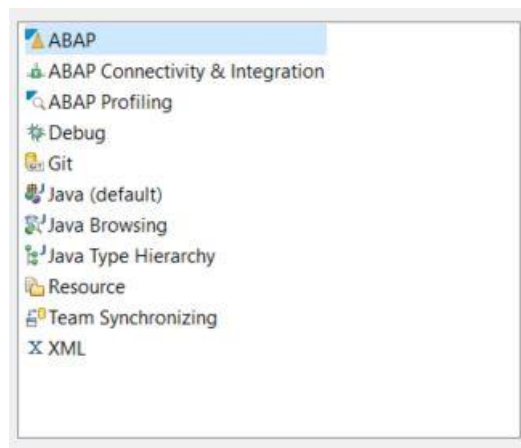


Figure 29 : Configuration Eclipse - Perspective ABAP²⁷

²⁶ Source : Donnée de l'auteur sur Eclipse

²⁷ Source : Donnée de l'auteur sur Eclipse

3. Dans Eclipse, sélectionner « Create an ABAP cloud project » ou : *File > New > ABAP Cloud Project* ;

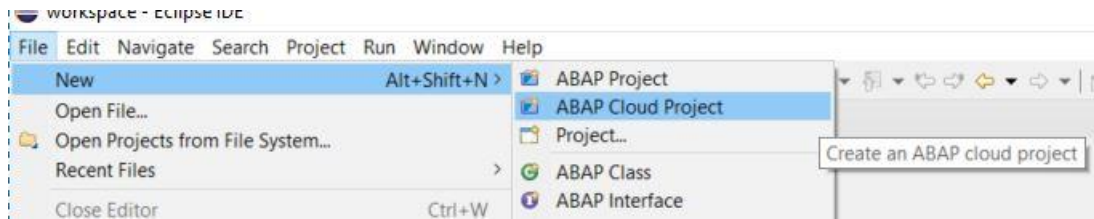


Figure 30 : Nouveau projet ABAP cloud²⁸

4. Sélectionner « Service Key » afin de créer une connexion avec l'ABAP environnement ;

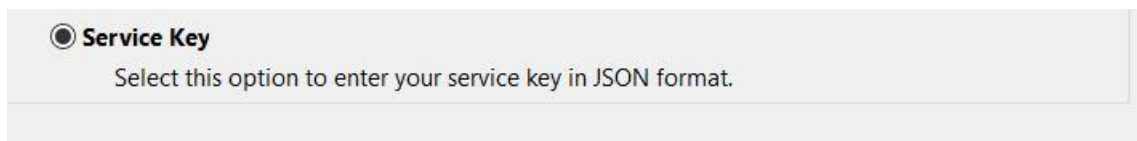


Figure 31 : Nouveau projet ABAP cloud - Service Key²⁹

5. Depuis le SCP cockpit, sélectionner l'instance ABAP, copier le Service Key (format JSON) et coller dans Eclipse ;

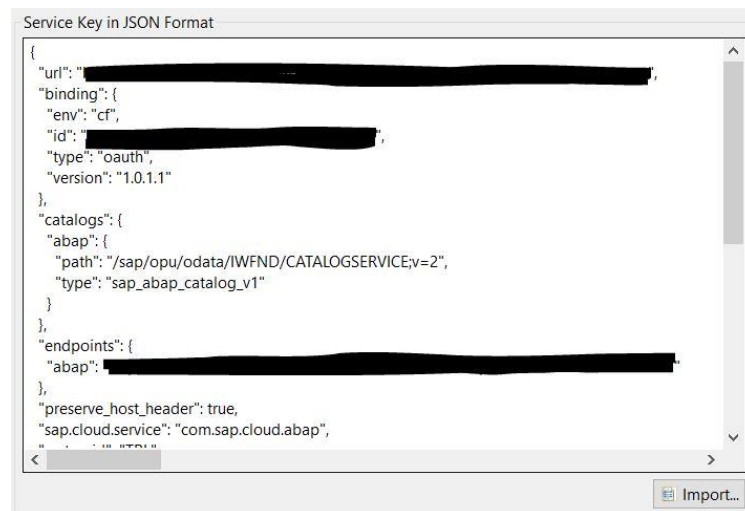


Figure 32 : Nouveau projet ABAP cloud - Service Key³⁰

²⁸ Source : Donnée de l'auteur sur Eclipse

²⁹ Source : Donnée de l'auteur sur Eclipse

³⁰ Source : Donnée de l'auteur sur Eclipse

6. Pour pouvoir s'identifier au SCP, sélectionner « Open Logon Page in Browser » et entrer l'identifiant et le mot de passe du compte cloud ;

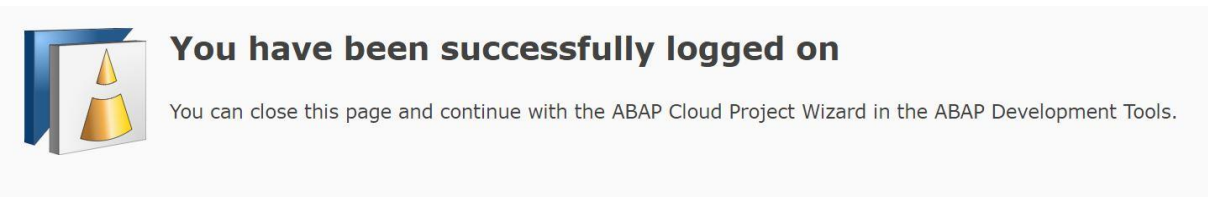


Figure 33 : Nouveau projet ABAP cloud - Login³¹

7. Poursuivre la création du projet en cliquant « Next » puis « Finish ».
8. Dans Eclipse, expandre le projet créé pour vérifier que la création a été faite avec succès : *Projet > Favorite Packages > ZLOCAL* ;

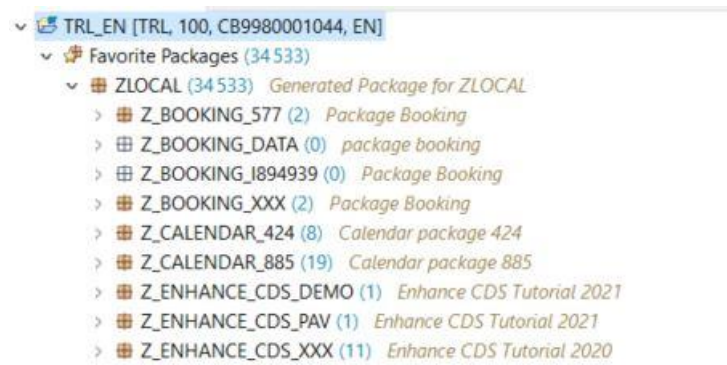


Figure 34 : Nouveau projet ABAP cloud - Vérification³²

Si les packages sont affichés sans erreur, le projet a été créé avec succès.

IMPORTANT : Puisqu'il s'agit d'un compte « trial », les différents packages visibles sur l'illustration représentent des projets d'autres utilisateurs. Il s'agit donc d'un environnement commun et il faut éviter d'utiliser des données sensibles dans le cadre d'un projet cloud trial.

8.4 Création d'un package

Nous allons maintenant créer le package où les différents composants d'un développement ABAP seront créés afin de démontrer comment cela est fait dans Eclipse.

³¹ Source : Donnée de l'auteur sur Eclipse

³² Source : Donnée de l'auteur sur Eclipse

1. Sur Eclipse, créer un nouveau package : *Projet > Favorite Packages > ZLOCAL* ;



Figure 35 : Création d'un package - Nouveau package³³

2. Clic droit sur le package : *New > ABAP Package* ;

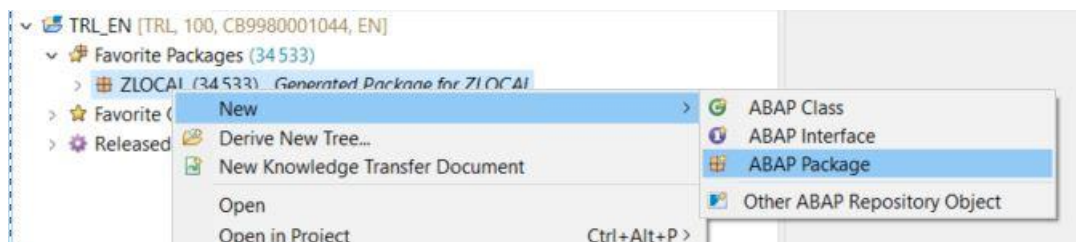


Figure 36 : Création d'un package - ABAP package³⁴

3. Définir un nom et une description ;

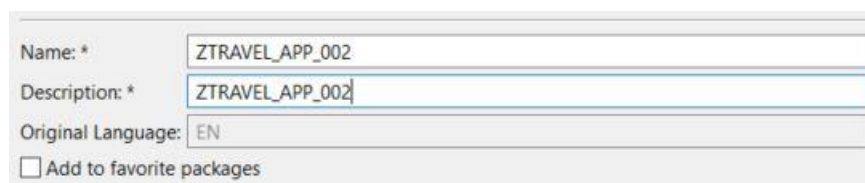


Figure 37 : Création d'un package - Nom et description du package³⁵

4. Création d'un nouvel ordre de transport ;



Figure 38 : Création d'un package - Ordre de transport³⁶

³³ Source : Donnée de l'auteur sur Eclipse

³⁴ Source : Donnée de l'auteur sur Eclipse

³⁵ Source : Donnée de l'auteur sur Eclipse

³⁶ Source : Donnée de l'auteur sur Eclipse

5. Pour faciliter la suite des opérations, ajouter le package crée aux favoris afin de le retrouver plus facilement : *Clic droit package > Add to Favorite Packages* ;

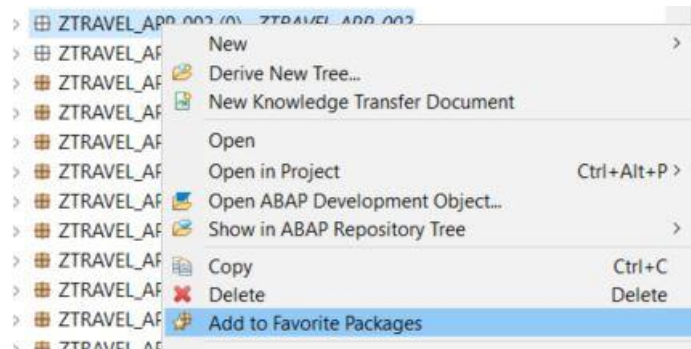


Figure 39 : Création d'un package- Ajout package aux favoris³⁷

8.5 Création de la base de données

Le premier artéfact que nous allons créer pour notre programme ABAP est la base de données. Il s'agit de données mises à disposition par SAP pour des entraînements, elle contenant des données de voyage.

1. Sur Eclipse, créer une nouvelle base de données : *Clic droit package > New > Other ABAP Repository Object* ;

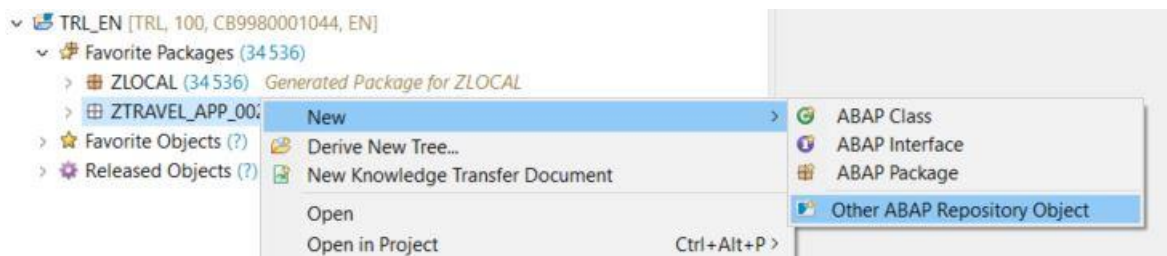


Figure 40 : Création de la base de données - Other ABAP Repository Object³⁸

³⁷ Source : Donnée de l'auteur sur Eclipse

³⁸ Source : Donnée de l'auteur sur Eclipse

2. Sélectionner « Database Table » parmi les différents composants ABAP : *Dossier Dictionary > Database Table* ;

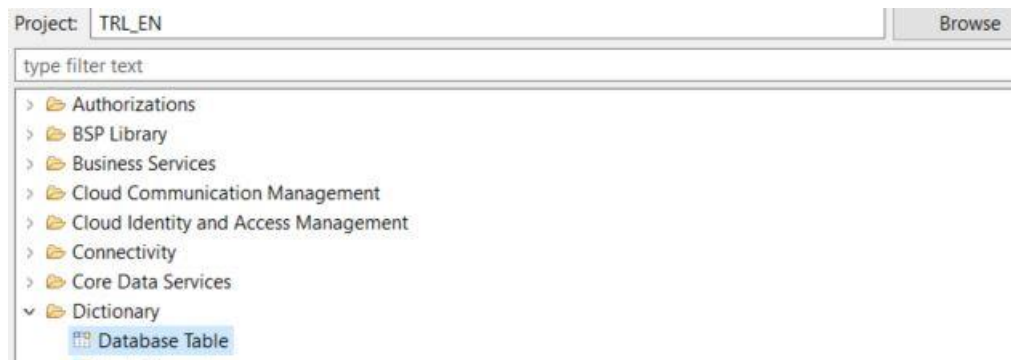


Figure 41 : Création de la base de données - Database Table³⁹

3. Définir un nom et une description ;

The screenshot shows the 'Create Database Table' dialog box in SAP. It has three input fields: 'Name: *' with the value 'ZTRAVEL_002', 'Description: *' with the value 'ZTRAVEL_002', and 'Original Language:' with the value 'EN'.

Figure 42 : Création d'une base de données - Nom et description⁴⁰

4. Sélectionner l'ordre de transport créé lors de l'étape précédente « Création d'un package » et valider avec « Finish » ;

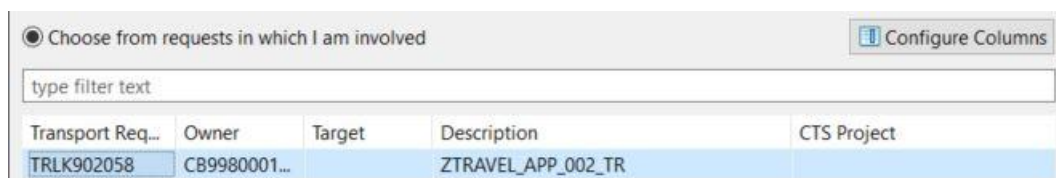


Figure 43 : Création d'une base de données - Ordre de transport⁴¹

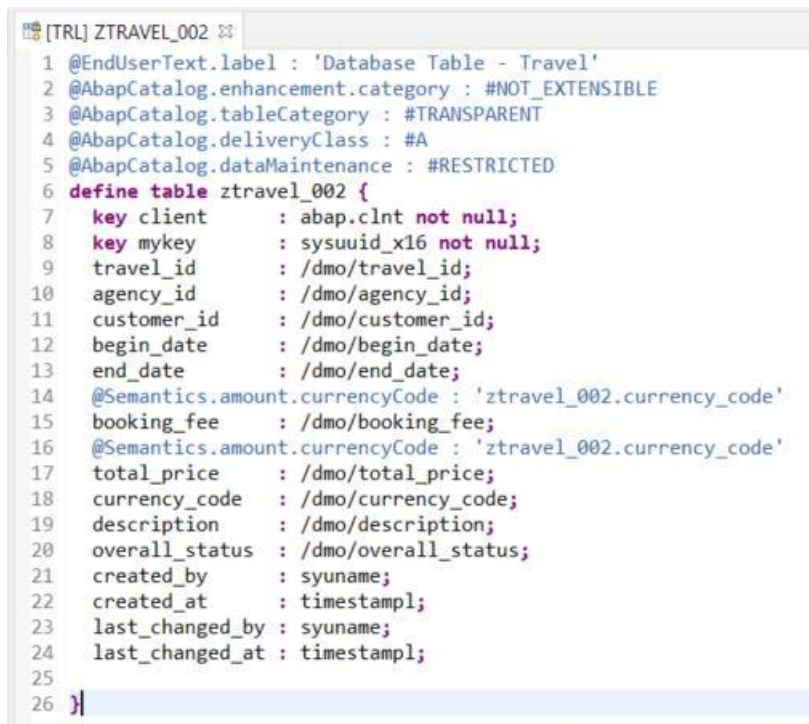
Une nouvelle fenêtre s'ouvre sur Eclipse. Nous avons maintenant la possibilité de créer notre base de données. Pour faciliter cette étape, nous allons utiliser un script prédéfini, fourni par SAP (disponible dans les annexes, *Annexe I : Script Database Table*).

³⁹ Source : Donnée de l'auteur sur Eclipse

⁴⁰ Source : Donnée de l'auteur sur Eclipse

⁴¹ Source : Donnée de l'auteur sur Eclipse

5. Copier et coller le script dans la fenêtre ouverte dans Eclipse ;



```
1 @EndUserText.label : 'Database Table - Travel'
2 @AbapCatalog.enhancement.category : #NOT_EXTENSIBLE
3 @AbapCatalog.tableCategory : #TRANSPARENT
4 @AbapCatalog.deliveryClass : #A
5 @AbapCatalog.dataMaintenance : #RESTRICTED
6 define table ztravel_002 {
7   key client      : abap.clnt not null;
8   key mykey       : sysuuid_x16 not null;
9   travel_id       : /dmo/travel_id;
10  agency_id       : /dmo/agency_id;
11  customer_id      : /dmo/customer_id;
12  begin_date       : /dmo/begin_date;
13  end_date         : /dmo/end_date;
14  @Semantics.amount.currencyCode : 'ztravel_002.currency_code'
15  booking_fee      : /dmo/booking_fee;
16  @Semantics.amount.currencyCode : 'ztravel_002.currency_code'
17  total_price      : /dmo/total_price;
18  currency_code    : /dmo/currency_code;
19  description       : /dmo/description;
20  overall_status   : /dmo/overall_status;
21  created_by       : syuname;
22  created_at       : timestamp1;
23  last_changed_by  : syuname;
24  last_changed_at  : timestamp1;
25
26 }
```

Figure 44 : Création d'une base de données - Script⁴²

6. Sauvegarder et activer le script que nous venons de créer ;



Figure 45 : Création d'une base de données - Activation⁴³

⁴² Source : Donnée de l'auteur sur Eclipse

⁴³ Source : Donnée de l'auteur sur Eclipse

7. Notre base de données apparaît dans le « Dictionary » et nous avons la possibilité de la visualiser : *Clic droit sur la Database Table > Open With > Data Preview* ;

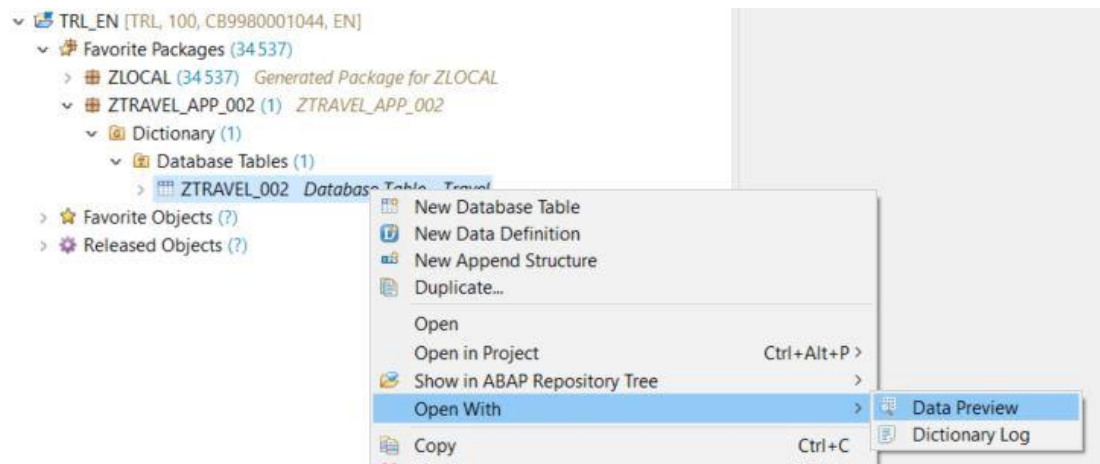


Figure 46 : Création d'une base de données - Preview⁴⁴

Une nouvelle fenêtre s'ouvre avec les différentes colonnes de notre base de données. Elle est pour l'instant vide, ce qui est tout à fait normal ;



Figure 47 : Création d'une base de données - Fenêtre de Preview⁴⁵

8. Insérer des données dans notre base de données : *Clic droit sur le package > New > ABAP Class* ;

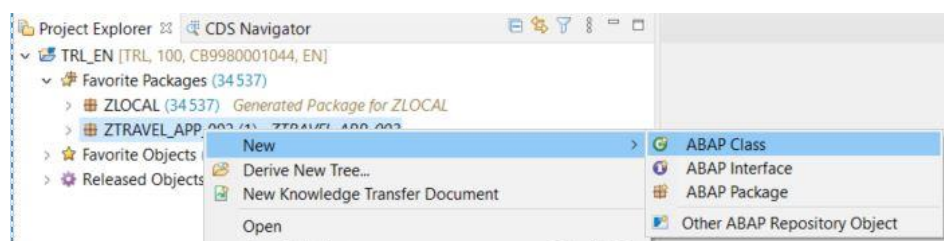


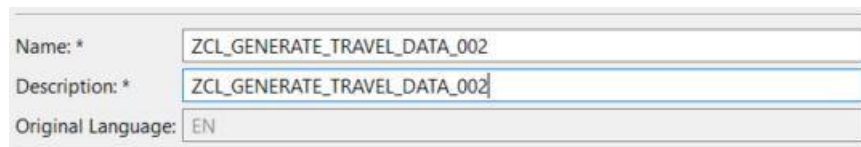
Figure 48 : Création d'une base de données - Nouvelle classe ABAP⁴⁶

⁴⁴ Source : Donnée de l'auteur sur Eclipse

⁴⁵ Source : Donnée de l'auteur sur Eclipse

⁴⁶ Source : Donnée de l'auteur sur Eclipse

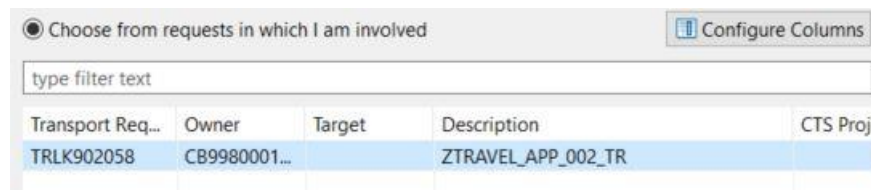
9. Définir un nom et une description ;



Name: *	ZCL_GENERATE_TRAVEL_DATA_002
Description: *	ZCL_GENERATE_TRAVEL_DATA_002
Original Language:	EN

Figure 49 : Création d'une base de données - Nom et description de la classe⁴⁷

10. Sélectionner l'ordre de transport créé lors de l'étape « Création d'un package » et valider avec « Finish » ;



Transport Req...	Owner	Target	Description	CTS Proj
TRLK902058	CB9980001...		ZTRAVEL_APP_002_TR	

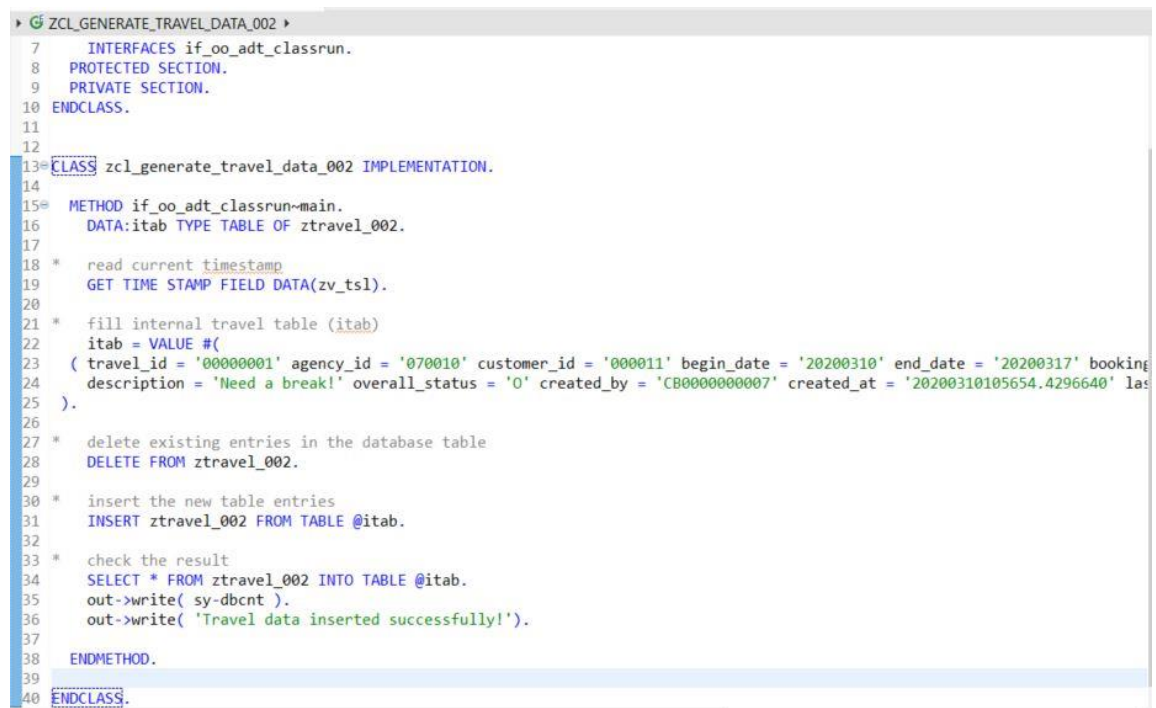
Figure 50 : Création d'une base de données - Ordre de transport Class⁴⁸

Une nouvelle fenêtre s'ouvre sur Eclipse. Nous avons maintenant la possibilité d'insérer des données dans notre base de données. Pour faciliter cette étape, nous allons utiliser un script prédéfini fourni par SAP (disponible dans les annexes, *Annexe II : Script ABAP Class*).

⁴⁷ Source : Donnée de l'auteur sur Eclipse

⁴⁸ Source : Donnée de l'auteur sur Eclipse

11. Copier et coller le script dans la fenêtre ouverte dans Eclipse ;



```
7  INTERFACES if_oo_adt_classrun.
8  PROTECTED SECTION.
9  PRIVATE SECTION.
10 ENDCLASS.
11
12
13 CLASS zcl_generate_travel_data_002 IMPLEMENTATION.
14
15 METHOD if_oo_adt_classrun-main.
16   DATA: itab TYPE TABLE OF ztravel_002.
17
18   * read current timestamp
19   GET TIME STAMP FIELD DATA(zv_tsl).
20
21   * fill internal travel table (itab)
22   itab = VALUE #(
23     ( travel_id = '00000001' agency_id = '070010' customer_id = '000011' begin_date = '20200310' end_date = '20200317' booking
24     description = 'Need a break!' overall_status = '0' created_by = 'CB0000000007' created_at = '20200310105654.4296640' las
25     ).
26
27   * delete existing entries in the database table
28   DELETE FROM ztravel_002.
29
30   * insert the new table entries
31   INSERT ztravel_002 FROM TABLE @itab.
32
33   * check the result
34   SELECT * FROM ztravel_002 INTO TABLE @itab.
35   out->write( sy-dbcnt ).
36   out->write( 'Travel data inserted successfully!' ).
37
38 ENDMETHOD.
39
40 ENDCLASS.
```

Figure 51 : Création d'une base de données - Script Class⁴⁹

12. Sauvegarder et activer le script que nous venons de créer ;



Figure 52 : Création d'une base de données - Activer script⁵⁰

⁴⁹ Source : Donnée de l'auteur sur Eclipse

⁵⁰ Source : Donnée de l'auteur sur Eclipse

13. Exécuter le script pour insérer les données dans la base de données : *Run > Run as > 1 ABAP Application (console)* ;

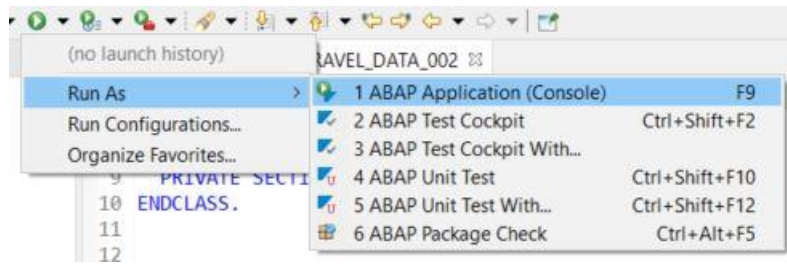


Figure 53 : Création d'une base de données - Exécuter script⁵¹

Une fois le script exécuté, un message est affiché confirmant l'insertion des données ;

```
Field
1 |
Travel data inserted successfully!
```

Figure 54 : Création d'une base de données - Validation⁵²

14. Vérifier l'insertion des données : *Dictionary > Database Tables > Clic droit sur la base de données > Open With > Data Preview* ;

Raw Data											
Filter pattern: 1 rows retrieved - 4 ms											
CLIENT	MYKEY	TRAVEL_ID	AGENCY_ID	CUSTOMER_ID	BEGIN_DATE	END_DATE	BOOKING_FEE	TOTAL_PRICE	CURRENCY_CODE	DESCRIPTION	
100	000000000...	000000001	070010	000011	2020-03-10	2020-03-17	17.00	800.00	EUR	Need a break!	

Figure 55 : Création d'une base de données - Vérification insertion⁵³

8.6 Création d'une View

Le prochain component du développement que nous allons créer est une View CDS. Les Views CDS sont une infrastructure pour définir, modéliser et consommer des « Data Model » sémantiquement riches depuis la base de données.

⁵¹ Source : Donnée de l'auteur sur Eclipse

⁵² Source : Donnée de l'auteur sur Eclipse

⁵³ Source : Donnée de l'auteur sur Eclipse

1. Sur Eclipse, créer une nouvelle base de données : *Clic droit package > New > Other ABAP Repository Object* ;

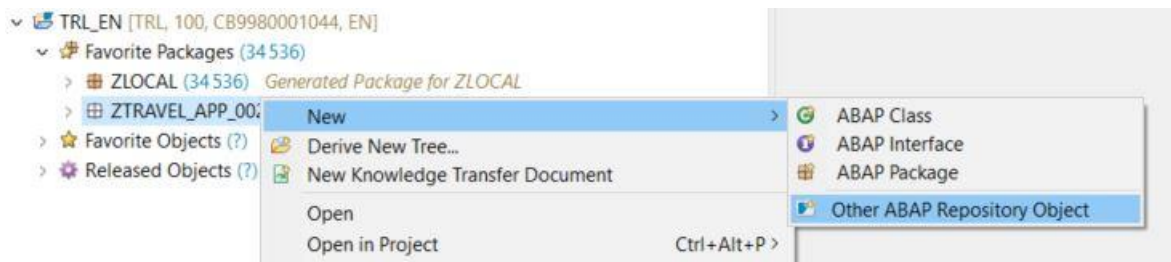


Figure 56 : Création d'une View - Other ABAP Repository Object⁵⁴

2. Sélectionner le dossier « Core Data Services » puis Data Definition ;



Figure 57 : Création d'une View - Data Definition⁵⁵

3. Définir un nom et une description pour la View ;

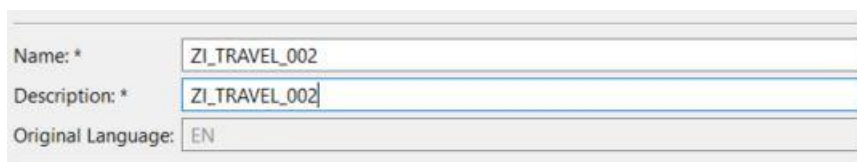


Figure 58 : Création d'une View - Nom et description⁵⁶

⁵⁴ Source : Donnée de l'auteur sur Eclipse

⁵⁵ Source : Donnée de l'auteur sur Eclipse

⁵⁶ Source : Donnée de l'auteur sur Eclipse

4. Sélectionner l'ordre de transport créé lors de l'étape « Création d'un package » et valider avec « Finish » ;

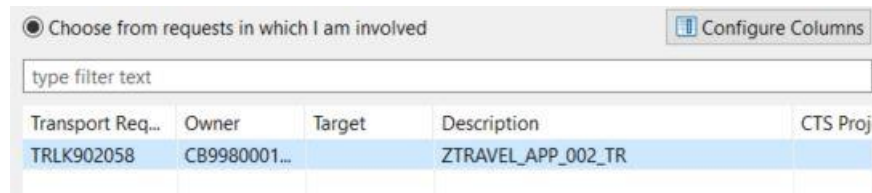


Figure 59 : Création d'une View - Ordre de transport View⁵⁷

Une nouvelle fenêtre s'ouvre sur Eclipse. Pour faciliter cette étape, nous allons utiliser un script prédéfini fourni par SAP (disponible dans les annexes, *Annexe III : Script CDS View*) pour créer la View.

5. Copier et coller le script dans la fenêtre ouverte dans Eclipse ;

```

1@AbapCatalog.sqlViewName: 'ZVI_TRAVEL_002'
2 @AbapCatalog.compiler.compareFilter: true
3 @AbapCatalog.preserveKey: true
4 @AccessControl.authorizationCheck: #CHECK
5 @EndUserText.label: 'Interface View - Travel'
6 define root view ZI_TRAVEL_002
7   as select from ztravel_002
8
9   association [0..1] to /DMO/I_Agency as _Agency on $projection.agency_id = _Agency.AgencyID
10  association [0..1] to /DMO/I_Customer as _Customer on $projection.customer_id = _Customer.CustomerID
11  association [0..1] to I_Currency as _Currency on $projection.currency_code = _Currency.Currency
12
13 {
14   key mykey,
15     travel_id,
16     agency_id,
17     customer_id,
18     begin_date,
19     end_date,
20     @Semantics.amount.currencyCode: 'currency_code'
21     booking_fee,
22     @Semantics.amount.currencyCode: 'currency_code'
23     total_price,
24     @Semantics.currencyCode: true
25     currency_code,
26     overall_status,
27     description,
28
29   /*-- Admin data --*/
30   @Semantics.user.createdBy: true
31   created_by,
32   @Semantics.systemDateTime.createdAt: true
33   created_at,
34   @Semantics.user.lastChangedBy: true
35   last_changed_by,
36   @Semantics.systemDateTime.lastChangedAt: true
37   last_changed_at,
38
39   /* Public associations */
40   _Agency,
41   _Customer,
42   _Currency
43 }

```

Figure 60 : Création d'une View - Script⁵⁸

⁵⁷ Source : Donnée de l'auteur sur Eclipse

⁵⁸ Source : Donnée de l'auteur sur Eclipse

6. Sauvegarder et activer le script que nous venons de créer ;



Figure 61 : Création d'une View - Activer script View⁵⁹

8.7 Création d'une projection View

Le prochain composant du développement que nous allons créer est une projection View CDS. Il s'agit d'une vue de champs définis dans une seule table. Elle projette uniquement les champs assignés à la vue. L'utilisation de projection View permet d'afficher uniquement des données nécessaires et utiles.

1. Sur Eclipse, créer une nouvelle base de données : *Clic droit package > New > Other ABAP Repository Object* ;

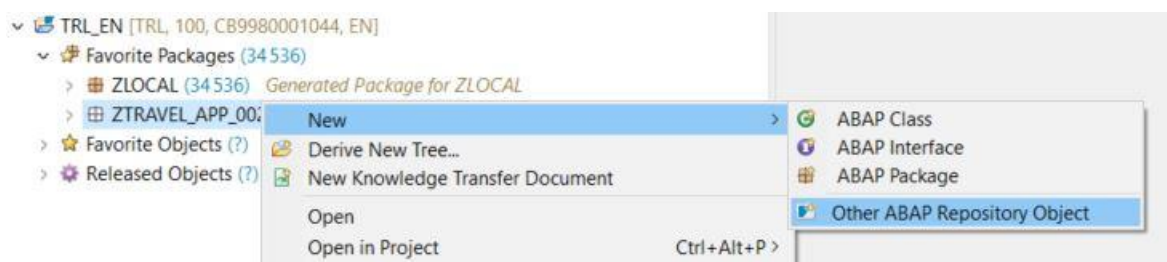


Figure 62 : Création d'une projection View - Other ABAP Repository Object⁶⁰

2. Sélectionner le dossier « Core Data Services » puis Data Definition ;



Figure 63 : Création d'une projection View - Data Definition⁶¹

⁵⁹ Source : Donnée de l'auteur sur Eclipse

⁶⁰ Source : Donnée de l'auteur sur Eclipse

⁶¹ Source : Donnée de l'auteur sur Eclipse

3. Définir un nom et une description pour la projection View ;

Name: *	ZC_TRAVEL_002
Description: *	ZC_TRAVEL_002
Original Language:	EN

Figure 64 : Création d'une projection View - Nom et description⁶²

4. Sélectionner l'ordre de transport créé lors de l'étape « Création d'un package » et valider avec « Finish » ;

● Choose from requests in which I am involved					Configure Columns
type filter text					
Transport Req...	Owner	Target	Description	CTS Proj	
TRLK902058	CB9980001...		ZTRAVEL_APP_002_TR		

Figure 65 : Création d'une projection View - Ordre de transport projection View⁶³

Une nouvelle fenêtre s'ouvre sur Eclipse. Pour faciliter cette étape, nous allons utiliser un script prédéfini fourni par SAP (disponible dans les annexes, *Annexe IV : Script projection View*) pour créer la projection View.

5. Copier et coller le script dans la fenêtre ouverte dans Eclipse.
6. Sauvegarder et activer le script que nous venons de créer ;



Figure 66 : Création d'une projection View - Activation⁶⁴

8.8 Création d'un OData

Le prochain component du développement que nous allons créer est un service OData. L'OData permet d'exposer des données qui peuvent être requêtées par des applications. Il

⁶² Source : Donnée de l'auteur sur Eclipse

⁶³ Source : Donnée de l'auteur sur Eclipse

⁶⁴ Source : Donnée de l'auteur sur Eclipse

s'agit d'un protocole basé sur HTTP et donc accessible via une URL.

1. Créer le service OData basé sur la projection View : *Clic droit sur la projection View > New Service Definition* ;

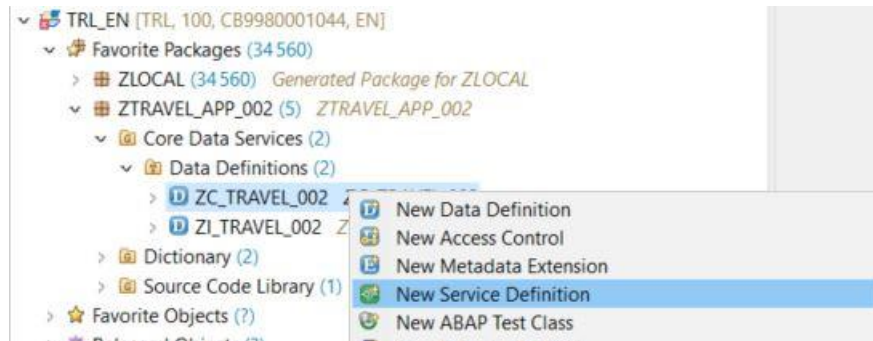


Figure 67 : Création d'un OData - New Service Définition⁶⁵

2. Définir un nom et une description pour le service ;

Name: *	ZSD_TRAVEL_002
Description: *	ZSD_TRAVEL_002
Original Language:	EN

Figure 68 : Création d'un OData - Nom et description⁶⁶

3. Sélectionner l'ordre de transport créé lors de l'étape « Création d'un package » et valider avec « Finish » ;

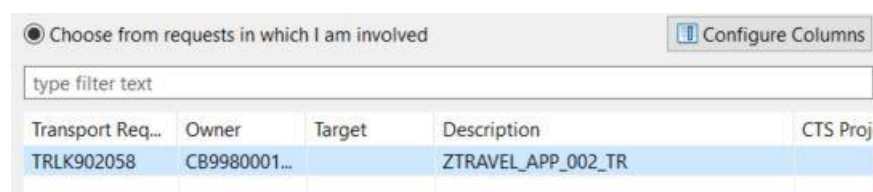


Figure 69 : Création d'un OData - Ordre de transport OData⁶⁷

⁶⁵ Source : Donnée de l'auteur sur Eclipse

⁶⁶ Source : Donnée de l'auteur sur Eclipse

⁶⁷ Source : Donnée de l'auteur sur Eclipse

Une nouvelle fenêtre s'ouvre sur Eclipse. Pour faciliter cette étape, nous allons utiliser un script prédéfini fourni par SAP (disponible dans les annexes, *Annexe V : Script service OData*) pour créer le service OData.

4. Copier et coller le script dans la fenêtre ouverte dans Eclipse.
5. Sauvegarder et activer le script que nous venons de créer ;



Figure 70 : Création d'un OData - Activation⁶⁸

6. Créer un service binding pour notre OData : *Clic droit service OData > New Service Binding* ;

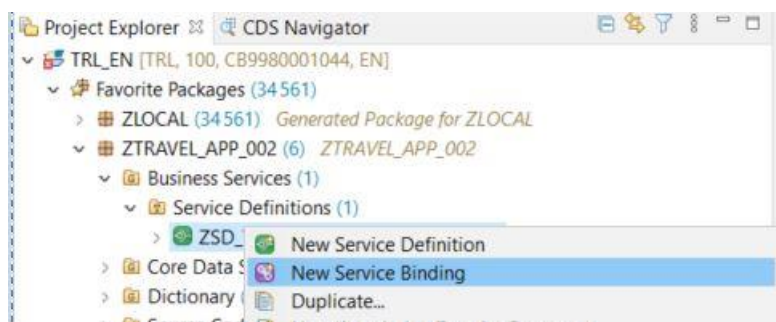


Figure 71 : Création d'un OData - Service Binding⁶⁹

7. Définir un nom et une description pour le service et sélectionner « OData V2 – UI » pour le Binding Type ;

Name: *	ZSB_TRAVEL_002
Description: *	ZSB_TRAVEL_002
Original Language:	EN
Binding Type: *	<div> <div> <div></div> <div>OData V2 - UI</div> <div></div> </div> </div>
Service Definition: *	<div> <div>ZSD_TRAVEL_002</div> <div>Browse...</div> </div>

Figure 72 : Création d'un OData - Nom, Description et Type⁷⁰

⁶⁸ Source : Donnée de l'auteur sur Eclipse

⁶⁹ Source : Donnée de l'auteur sur Eclipse

⁷⁰ Source : Donnée de l'auteur sur Eclipse

8. Sélectionner l'ordre de transport créé lors de l'étape « Création d'un package » et valider avec « Finish » ;

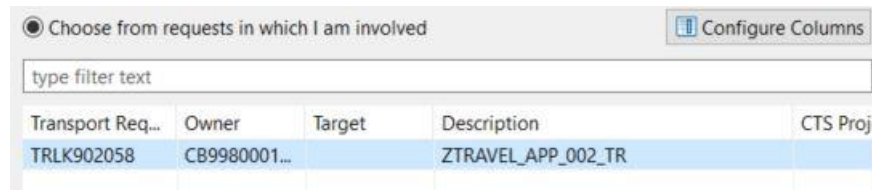


Figure 73 : Création d'un OData - Ordre de transport binding⁷¹

9. Sauvegarder et activer le binding que nous venons de créer ;

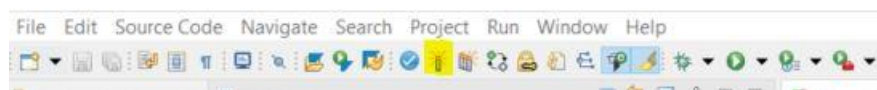


Figure 74 : Création d'un OData - Activation binding⁷²

10. Pour rendre le binding fonctionnel, il faut le publier : *Publish* ;

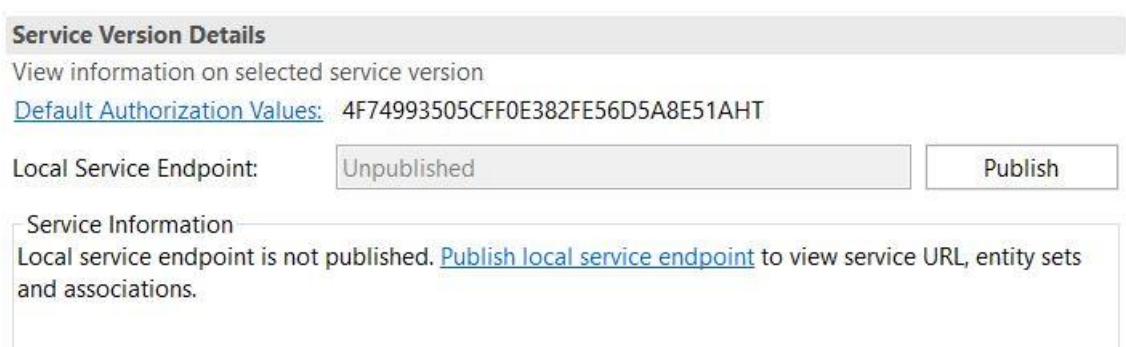


Figure 75 : Création d'un OData - Publier binding⁷³

⁷¹ Source : Donnée de l'auteur sur Eclipse

⁷² Source : Donnée de l'auteur sur Eclipse

⁷³ Source : Donnée de l'auteur sur Eclipse

Le processus de publication peut prendre un instant. Une fois terminé, on peut voir les différentes entités de notre service ;

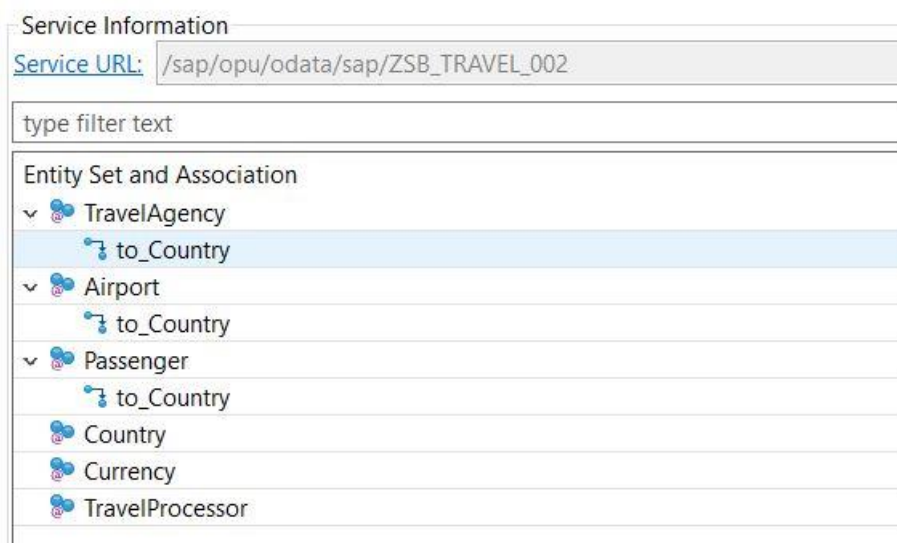


Figure 76 : Création d'un OData - Publication du binding⁷⁴

Avec un double-clic sur « TravelProcessor », nous pouvons avoir un aperçu de l'application FIORI. Il suffit d'entrer notre adresse électronique, le mot de passe du cockpit et nous pouvons visualiser l'application ;

Travel ID	Agency ID	Customer ID	Starting Date	End Date	Total Price	Overall Sta...
1	Travel from Walldorf (70010)	Buchholm (11)	Mar 10, 2020	Mar 17, 2020	800.00 EUR	0

Figure 77 : Création d'un OData - Aperçu Fiori⁷⁵

8.9 Migration d'un projet local vers le cloud

Pour ce deuxième scénario de prototype, nous allons migrer un projet local vers le cloud à l'aide de Git. Comme pour le scénario précédent, nous allons utiliser Eclipse 2021-06 (4.20.0), le SAP GUI (SAP NetWeaver 7.50) ainsi que GIT.

⁷⁴ Source : Donnée de l'auteur sur Eclipse

⁷⁵ Source : Donnée de l'auteur sur Eclipse

8.10 Installation d'abapGIT

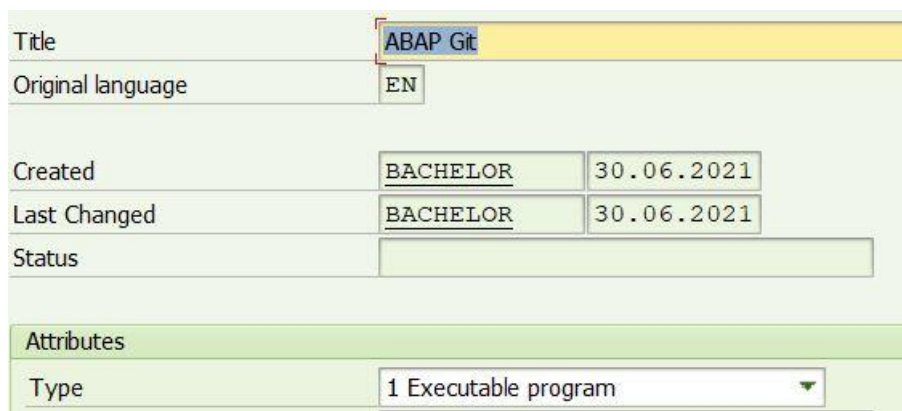
1. Dans le système local, créer un nouveau programme : *SE38 > Entrer le nom du programme > Create* ;



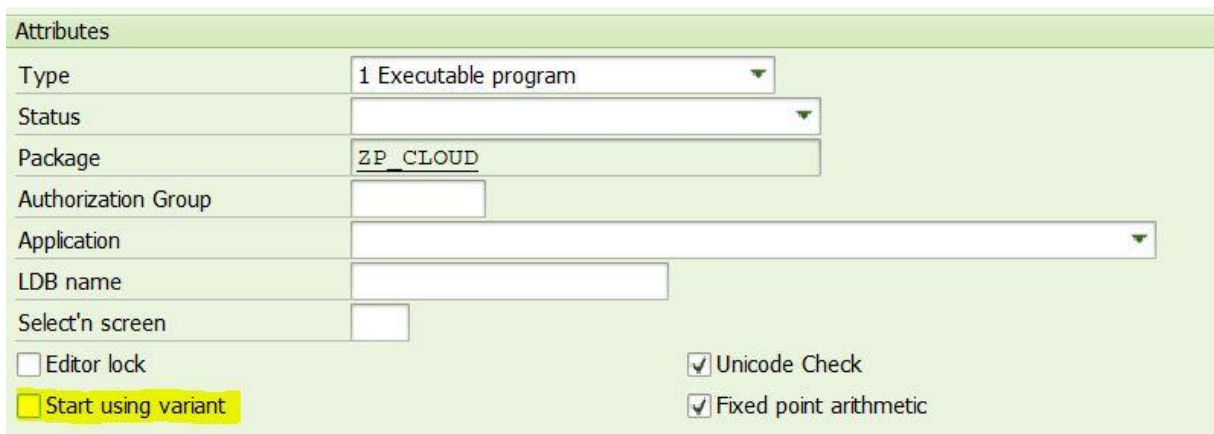
Program	ZABAPGIT	Create
---------	----------	--------

Figure 78 : Installation d'abapGIT - Nouveau programme⁷⁶

2. Dans la nouvelle fenêtre qui s'ouvre, entrer un titre, sous « Attributes », changer le Type en « Executable program » et décocher la case « Start using variant » ;



Title	ABAP Git	
Original language	EN	
Created	BACHELOR	30.06.2021
Last Changed	BACHELOR	30.06.2021
Status		
Attributes		
Type	1 Executable program	



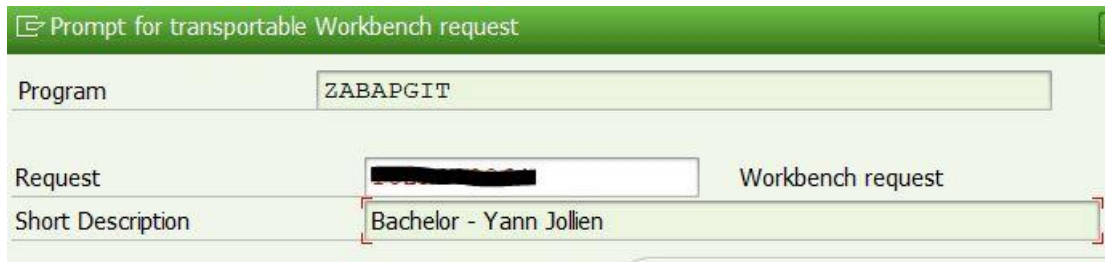
Attributes	
Type	1 Executable program
Status	
Package	ZP_CLOUD
Authorization Group	
Application	
LDB name	
Select'n screen	
<input type="checkbox"/> Editor lock	<input checked="" type="checkbox"/> Unicode Check
<input checked="" type="checkbox"/> Start using variant	<input checked="" type="checkbox"/> Fixed point arithmetic

Figure 79 : Installation d'abapGit – Titre, type de programme et variant⁷⁷

⁷⁶ Source : Donnée de l'auteur sur SAP GUI

⁷⁷ Source : Donnée de l'auteur sur SAP GUI

3. Dans la nouvelle fenêtre, sauvegarder, puis sélectionner l'ordre de transport ou créer le programme sans ordre de transport. Pour ce faire choisir « Local Object » ;



Prompt for transportable Workbench request

Program: ZABAPGIT

Request: [Empty] Workbench request

Short Description: Bachelor - Yann Jollien

Figure 80 : Installation d'abapGIT - Ordre de transport⁷⁸

4. Se rendre sur <https://github.com/abapGit/abapGit> et ouvrir « Latest build : zabapgit.abap » ;



Figure 81 : Installation d'abapGIT - Latest build⁷⁹

5. Copier et coller tout le code ABAP dans le programme « ZABAPGIT » créé auparavant.
6. Sauvegarder et activer le programme (l'activation peut prendre du temps, c'est normal) ;



Figure 82 : Installation d'abapGIT - Activer programme⁸⁰

⁷⁸ Source : Donnée de l'auteur sur SAP GUI

⁷⁹ Source : <https://github.com/abapGit/abapGit>

⁸⁰ Source : Donnée de l'auteur sur SAP GUI

7. S'assurer que l'activation s'est déroulée sans erreur : *Au fond de la fenêtre > Object activated* ;

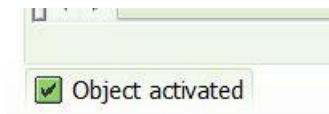


Figure 83 : Installation d'abapGIT - Confirmation d'activation⁸¹

8. Exécuter le programme : *SE38 > Execute* ;



Figure 84 : Installation d'abapGit - Exécuter le programme⁸²

Une nouvelle fenêtre d'accueil d'abapGit s'ouvre, abapGit a été installé avec succès ;

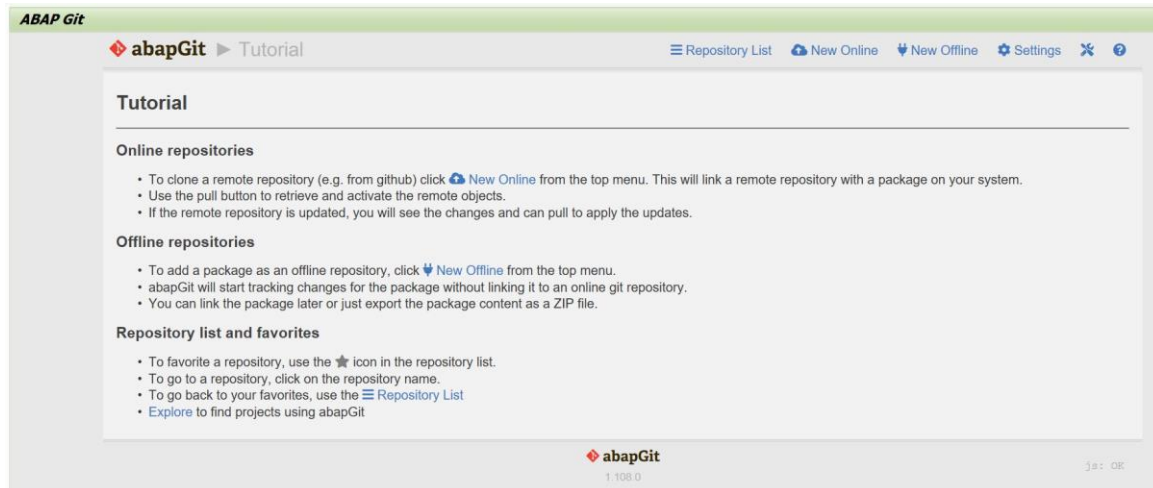


Figure 85 : Installation d'abapGit - Page d'accueil⁸³

⁸¹ Source : Donnée de l'auteur sur SAP GUI

⁸² Source : Donnée de l'auteur sur SAP GUI

⁸³ Source : Donnée de l'auteur sur SAP GUI

8.11 Configurer connexion SSL

Pour que SAP puisse communiquer avec GitHub, nous devons configurer la connexion SSL dans SAP (abapGit, 2021).

1. Ouvrir le gestionnaire de profils dans SAP : Transaction /nrz10 ;
2. Via F4, sélectionner le profile « Default » ;

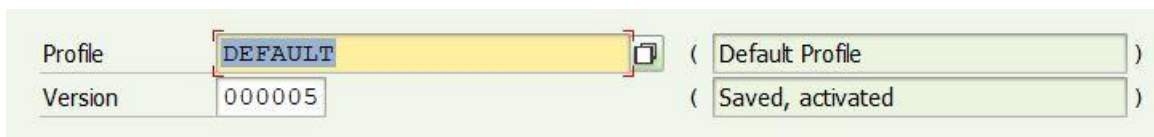


Figure 86 : Configurer connexion SSL - Sélection profil⁸⁴

3. Sous Edit Profile, cocher « Extended maintenance » et ensuite Change ;

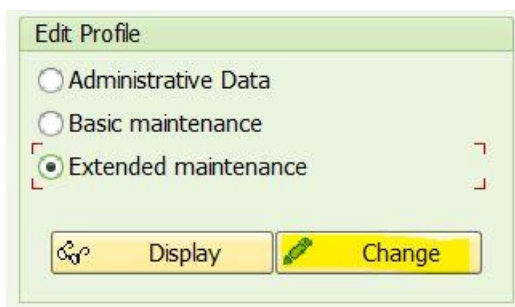


Figure 87 : Configurer connexion SSL - Modifier profil⁸⁵

4. Ouvrir l'option paramètres : *Parameter* ;



Figure 88 : Configurer connexion SSL – Paramètre client_ciphersuites⁸⁶

⁸⁴ Source : Donnée de l'auteur sur SAP GUI

⁸⁵ Source : Donnée de l'auteur sur SAP GUI

⁸⁶ Source : Donnée de l'auteur sur SAP GUI

5. Entrer `ssl/client_ciphersuites` dans le champ « Parameter name » et `150:PFS:HIGH::EC_P256:EC_HIGH` dans le champ « Parameter val. », valider avec « Copy » ;

Maintain Profile 'DEFAULT' Version '000005'

Copy [Icon] [Icon] Line [Icon] Line [Icon] PARAM+ [Icon] PARAM-

Parameter name:

Parameter val.:

Figure 89 : Configurer connexion SSL - Paramètre nom et valeur⁸⁷

6. Une fois la copie effectuée, retourner à l'écran de maintenance via F3 et également effectuer une copie via « Copy » ;

Maintain Profile 'DEFAULT' Version '000005'

Copy [Icon] [Icon] [Icon] [Icon] Parameter [Icon] Parameter [Icon] Parameter [Icon] [Icon]

Figure 90 : Configurer connexion SSL - Copier⁸⁸

7. Ouvrir l'option paramètres : *Parameter* ;

Maintain Profile 'DEFAULT' Version '000005'

Copy [Icon] [Icon] [Icon] [Icon] **Parameter** [Icon] Parameter [Icon] Parameter [Icon] [Icon]

Figure 91 : Configurer connexion SSL – Paramètre ciphersuites⁸⁹

⁸⁷ Source : Donnée de l'auteur sur SAP GUI

⁸⁸ Source : Donnée de l'auteur sur SAP GUI

⁸⁹ Source : Donnée de l'auteur sur SAP GUI

8. Entrer `ssl/ciphersuites` dans le champ « Parameter name » et `135:PFS:HIGH::EC_P256:EC_HIGH` dans le champ « Parameter val. », valider avec « Copy » ;

Maintain Profile 'DEFAULT' Version '000005'

Copy Line Line PARAM+ PARAM-

Parameter name: ssl/ciphersuites Status Active Seq. no. 1

Parameter val.: 135:PFS:HIGH::EC_P256:EC_HIGH

Figure 92 : Configurer connexion SSL - Paramètre 2 nom et valeur⁹⁰

9. Une fois la copie effectuée, retourner à l'écran de maintenance via F3 et effectuer une copie via « Copy » ;

Maintain Profile 'DEFAULT' Version '000005'

Copy Parameter Parameter Parameter

Figure 93 : Configurer connexion SSL - Copier⁹¹

10. Via F3 retourner à l'écran « Edit Profiles » puis : **Profile > Save >** (Valider l'activation quand le popup s'ouvre) ;

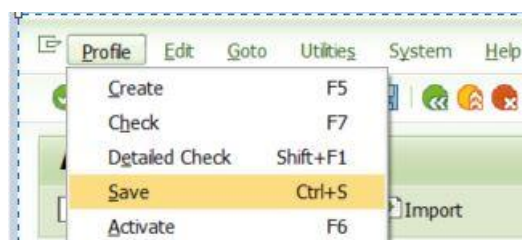


Figure 94 : Configurer connexion SSL - Sauvegarder⁹²

⁹⁰ Source : Donnée de l'auteur sur SAP GUI

⁹¹ Source : Donnée de l'auteur sur SAP GUI

⁹² Source : Donnée de l'auteur sur SAP GUI

11. Pour la prochaine étape, nous conseillons d'utiliser Google Chrome. Ouvrir GitHub, se connecter au compte et copier le certificat de la page : *Clic sur le cadenas devant l'url de la page > Certificate > Details > Copy to File > Enregistrer le certificat sur l'ordinateur ;*

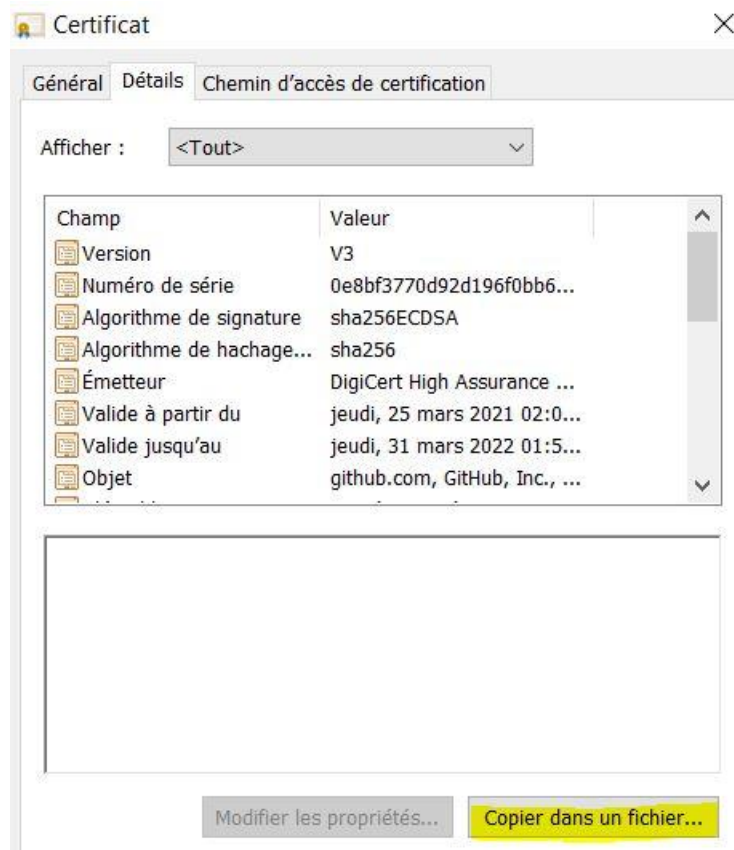


Figure 95 : Configurer connexion SSL - Sauvegarder certificat⁹³

⁹³ Source : Donnée de l'auteur sur Google Chrome

12. Ouvrir l'onglet « Chemin d'accès de certification » et répéter l'étape 11 pour les deux certificats manquants : *Afficher le certificat* ;



Figure 96 : Configuration connexion SLL - Aperçu des certificats⁹⁴

13. Nous devons maintenant avoir trois certificats enregistrés sur notre ordinateur ;



Figure 97 : Configuration connexion SSL - Certificats téléchargés⁹⁵

14. Retourner sur SAP, ouvrir la transaction /nSTRUST et passer en mode modification ;



Figure 98 : Configuration connexion SSL - Change de mode⁹⁶

⁹⁴ Source : Donnée de l'auteur sur Google Chrome

⁹⁵ Source : Donnée de l'auteur sur Ordinateur

⁹⁶ Source : Donnée de l'auteur sur SAP GUI

15. Ouvrir le SSL Client (Anonyme) (Si le dossier est précédé d'un croix rouge, clic droit > create > Enter) ;

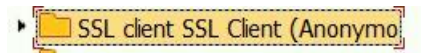


Figure 99 : Configuration connexion SSL - SSL Client⁹⁷

16. Au fond de la fenêtre, ajouter les certificats enregistrés auparavant via le symbole d'ajout ;



Figure 100 : Configuration connexion SSL - Ajout des certificats⁹⁸

17. Sélectionner le certificat et valider ;

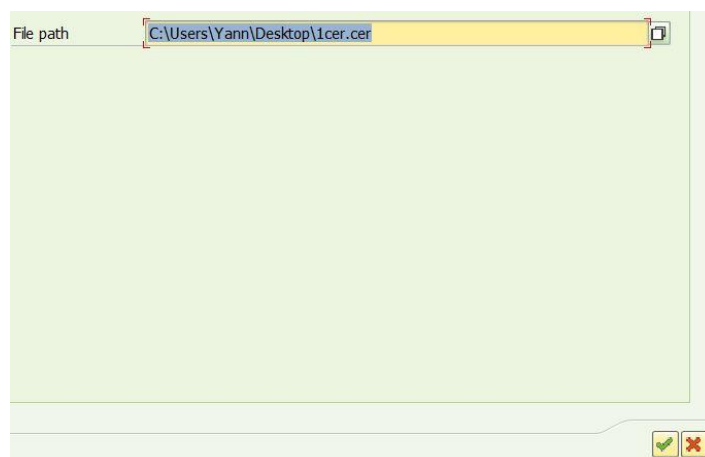


Figure 101 : Configuration connexion SSL - Sélection certificat⁹⁹

⁹⁷ Source : Donnée de l'auteur sur SAP GUI

⁹⁸ Source : Donnée de l'auteur sur SAP GUI

⁹⁹ Source : Donnée de l'auteur sur SAP GUI

18. Valider l'ajout du certificat via « Add to Certificate List » ;

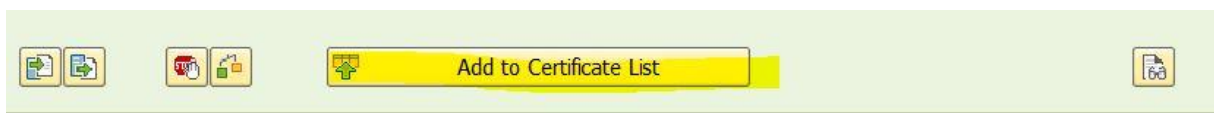


Figure 102 : Configuration connexion SSL- Valider ajout certificat¹⁰⁰

19. Répéter les étapes 16 à 18 pour les deux autres certificats, afin d'avoir finalement trois certificats dans la liste ;

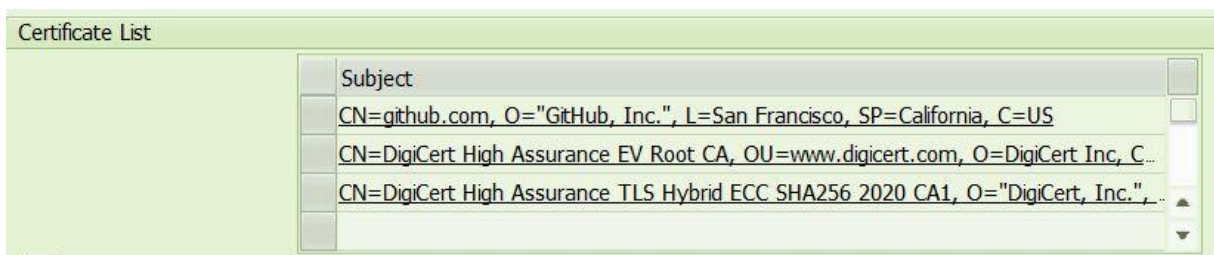


Figure 103 : Configuration connexion SSL - Liste des certificats ajoutés¹⁰¹

20. Redémarrer le système SAP.

8.12 Création d'un dépôt GIT

Pour l'utilisation d'un compte Git, la création d'un compte est nécessaire. Il existe plusieurs outils et plateformes Git. Les plus connus sont GitHub et GitLab. Pour ce prototype, nous allons utiliser un compte GitHub.

¹⁰⁰ Source : Donnée de l'auteur sur SAP GUI

¹⁰¹ Source : Donnée de l'auteurs sur SAP GUI

Sur GitHub, créer un nouveau dépôt : New ;

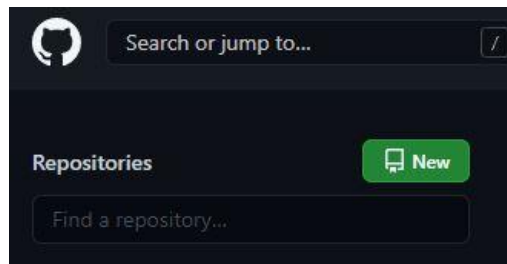


Figure 104 : Création d'un dépôt Git - Nouveau dépôt¹⁰²

1. Configurer le dépôt : Nom du dépôt et cocher « Add a README file » (dans un cadre professionnel il est conseillé de travailler avec l'option « Private » afin d'éviter que des personnes externes aient accès au dépôt) ;

A screenshot of the GitHub 'Create repository' configuration page. The 'Owner' is set to 'YannJollien' and the 'Repository name' is 'abapgit-workflow'. A note suggests repository names should be short and memorable, with an example 'stunning-adventure?'. The 'Description' field is optional. Under 'Visibility', 'Public' is selected, but 'Private' is also shown with a lock icon. The 'Initialize this repository with:' section has three options: 'Add a README file' (checked), 'Add .gitignore', and 'Choose a license'. A note at the bottom states the default branch will be 'main'.

Figure 105 : Création d'un dépôt Git - Configuration¹⁰³

¹⁰² Source : Donnée de l'auteur sur GitHub

¹⁰³ Source : Donnée de l'auteur sur GitHub

Une fois le dépôt créé, nous pouvons voir l'initial commit avec le README.md ;

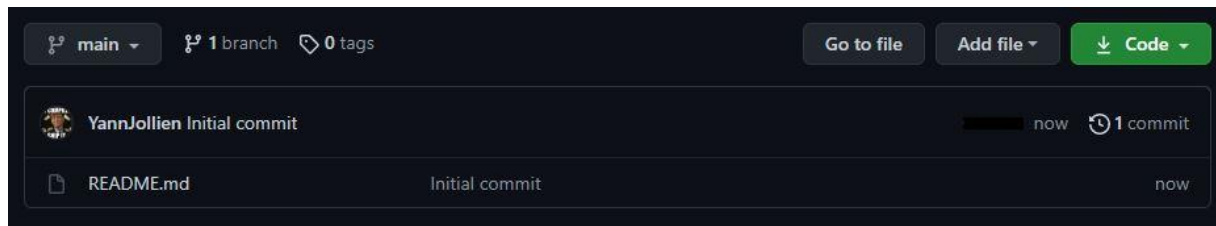


Figure 106 : Création d'un dépôt Git - Initial commit¹⁰⁴

8.13 Migration code ABAP vers dépôt Git

1. Copier l'URL de notre dépôt GitHub : **Code > Copy** ;

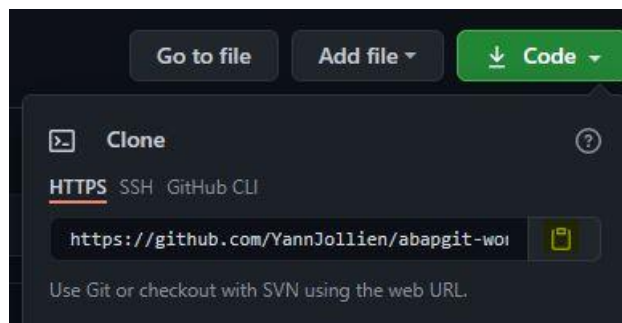


Figure 107 : Migration code ABAP vers dépôt Git - URL¹⁰⁵

2. Ouvrir abapGit en exécutant le programme « ZABAPGIT » et sélectionner « New Online » ;

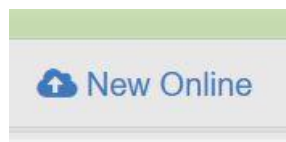


Figure 108 : Migration code ABAP vers dépôt Git - New Online¹⁰⁶

¹⁰⁴ Source : Donnée de l'auteur sur GitHub

¹⁰⁵ Source : Donnée de l'auteur sur GitHub

¹⁰⁶ Source : Donnée de l'auteur sur SAP GUI

3. Coller l'URL dans « Git Repository URL » et valider avec « Create Package » ;

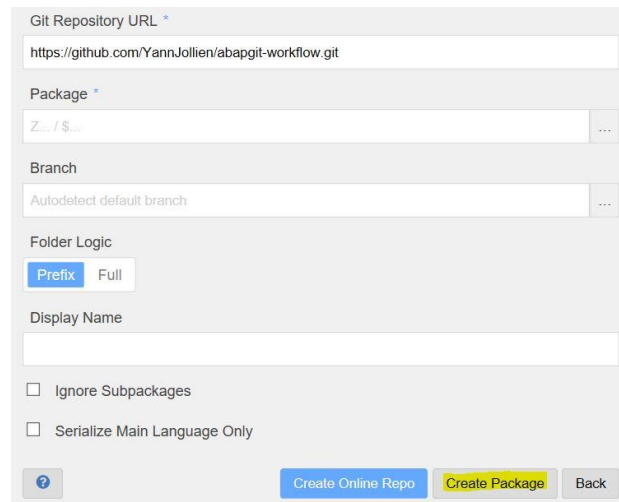


Figure 109 : Migration code ABAP vers dépôt Git - Create Package¹⁰⁷

4. Entrer un nom, une courte description, passer le « Software Component » en LOCAL via F4 et confirmer ;

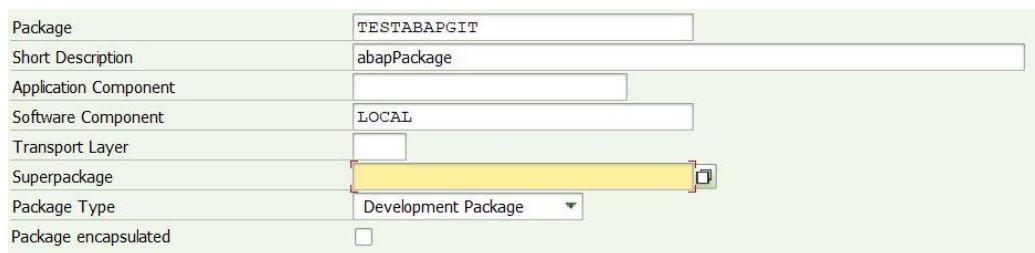


Figure 110 : Migration code ABAP vers dépôt Git - Configuration¹⁰⁸

5. Créer un workbench request: *Own Requests* ;




Figure 111 : Migration code ABAP vers dépôt Git - Own Requests¹⁰⁹

¹⁰⁷ Source : Donnée de l'auteur sur SAP GUI

¹⁰⁸ Source : Donnée de l'auteur sur SAP GUI

¹⁰⁹ Source : Donnée de l'auteur sur SAP GUI

6. Entrer une description et sauvegarder ;

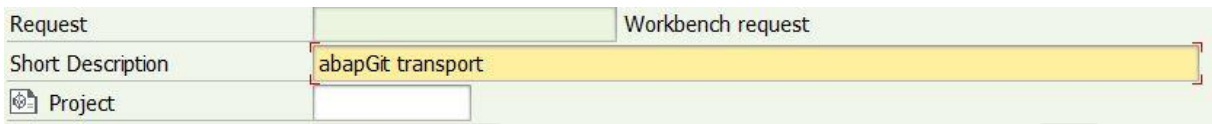


Figure 112 : Migration code ABAP vers dépôt Git - Workbench description¹¹⁰

7. Valider le « Workbench request » ;



Figure 113 : Migration code ABAP vers dépôt Git - Valider requête¹¹¹

8. Le package que nous venons de créer est automatiquement inséré et nous pouvons maintenant cloner le dépôt : *Clone Online Repo* ;

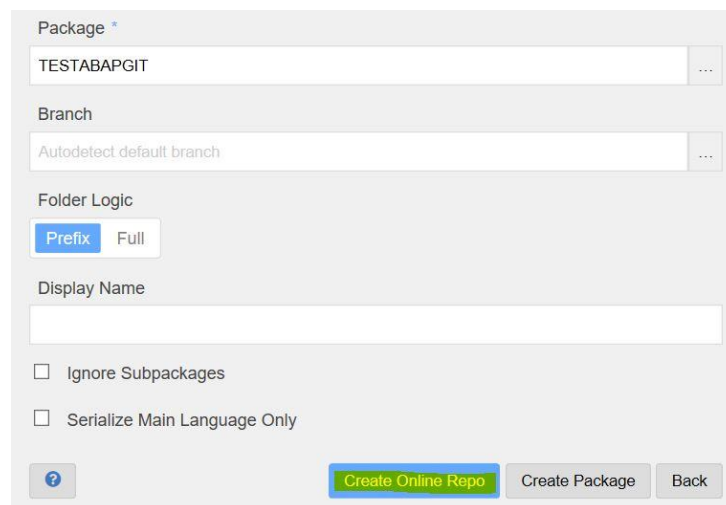


Figure 114 : Migration code ABAP vers dépôt Git - Cloner dépôt¹¹²

¹¹⁰ Source : Donnée de l'auteur sur SAP GUI

¹¹¹ Source : Donnée de l'auteur sur SAP GUI

¹¹² Source : Donnée de l'auteur sur SAP GUI

Une fois le clonage terminé, nous pouvons apercevoir notre dépôt GitHub ;

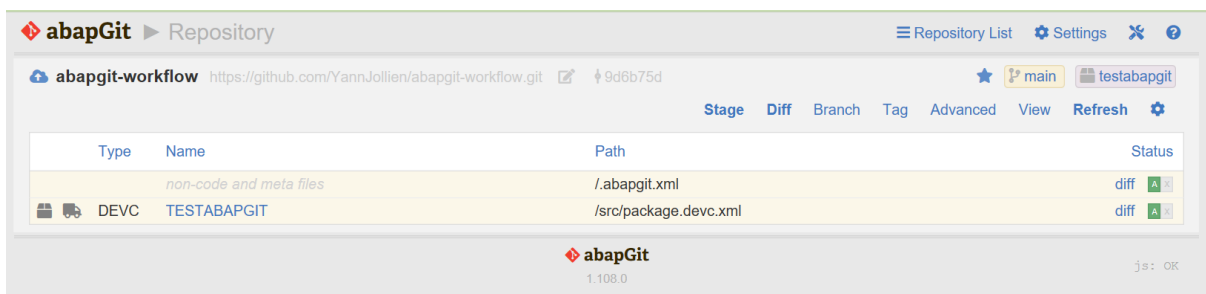


Figure 115 : Migration code ABAP vers dépôt Git - Dépôt sur SAP¹¹³

8.14 Ajouter élément ABAP au dépôt

Pour ajouter du contenu à notre dépôt, nous allons créer une classe.

1. Ouvrir Eclipse et se connecter au système SAP (le même que celui où le clone du dépôt a été créé) : **File > New > ABAP Project** ;

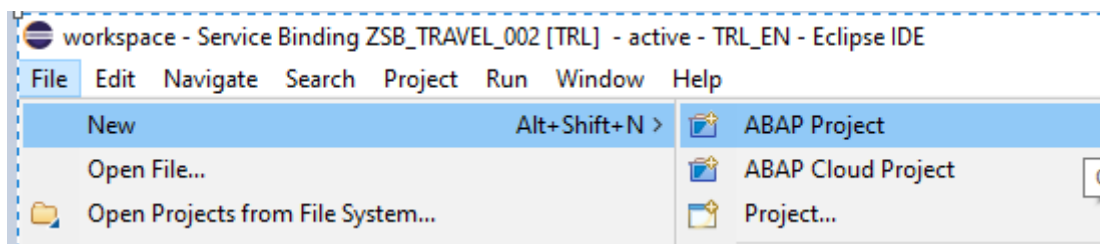


Figure 116 : Ajouter élément ABAP au dépôt - Nouveau projet¹¹⁴

¹¹³ Source : Donnée de l'auteur sur SAP GUI

¹¹⁴ Source : Donnée de l'auteur sur Eclipse

2. Expandre le projet puis nous pouvons voir notre package « TESTABAPGIT » ;

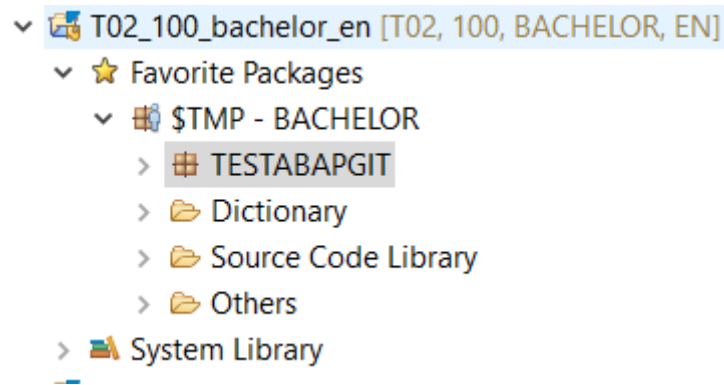


Figure 117 : Ajout élément ABAP au dépôt - Package¹¹⁵

3. Ajouter une classe au package : *Clic droit sur le package > New > Class* ;

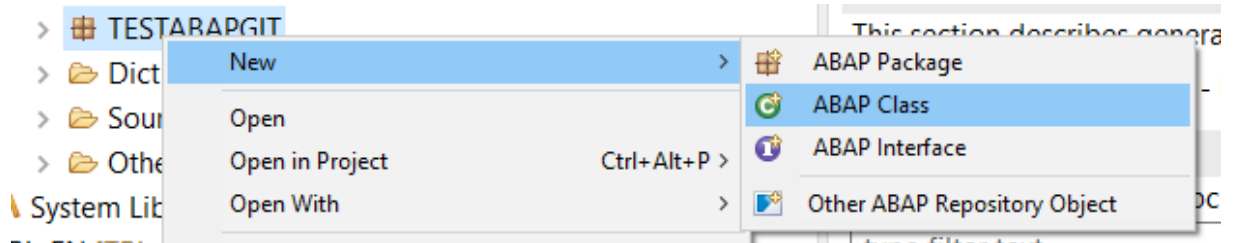


Figure 118 : Ajout élément ABAP au dépôt - Nouvelle classe¹¹⁶

4. Remplir les champs « Name » (Z_CLASS_TEST) et « Description » (Test Class), puis « Next ».
5. Sélectionner l'ordre de transport et valider avec « Finish ».
6. Sauvegarder et activer la nouvelle classe.

¹¹⁵ Source : Donnée de l'auteur sur Eclipse

¹¹⁶ Source : Donnée de l'auteur sur Eclipse

7. Retourner sur SAP, ouvrir le dépôt dans abapGit et synchroniser via « Refresh » ;

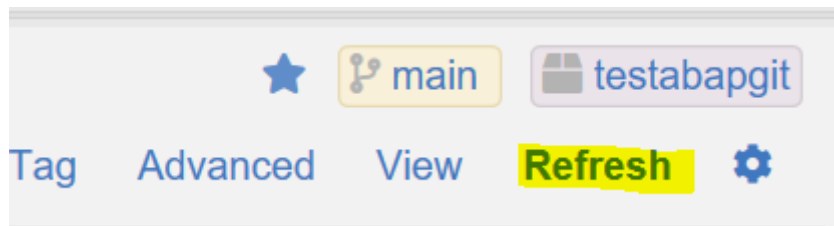


Figure 119 : Ajout élément ABAP au dépôt - Synchroniser¹¹⁷

La nouvelle classe apparaît ;

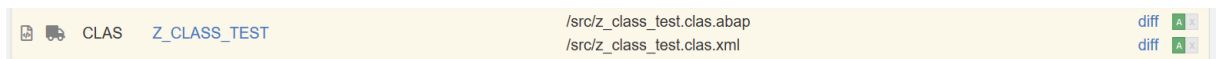


Figure 120 : Ajout élément ABAP au dépôt - Nouvelle classe dans le dépôt¹¹⁸

8. Sélectionner « Stage » ;

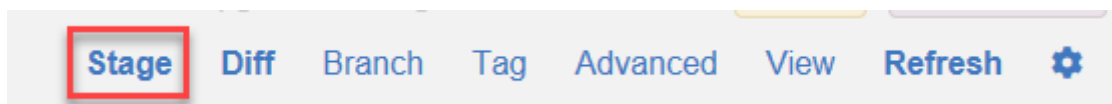


Figure 121 : Ajout élément ABAP au dépôt - Stage¹¹⁹

9. Sélectionner « Add all and commit » ;

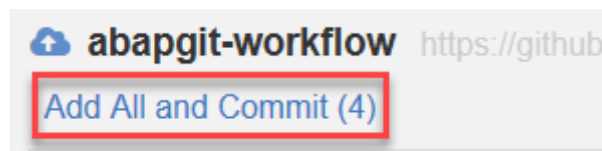


Figure 122 : Ajout élément ABAP au dépôt - Add all et commit¹²⁰

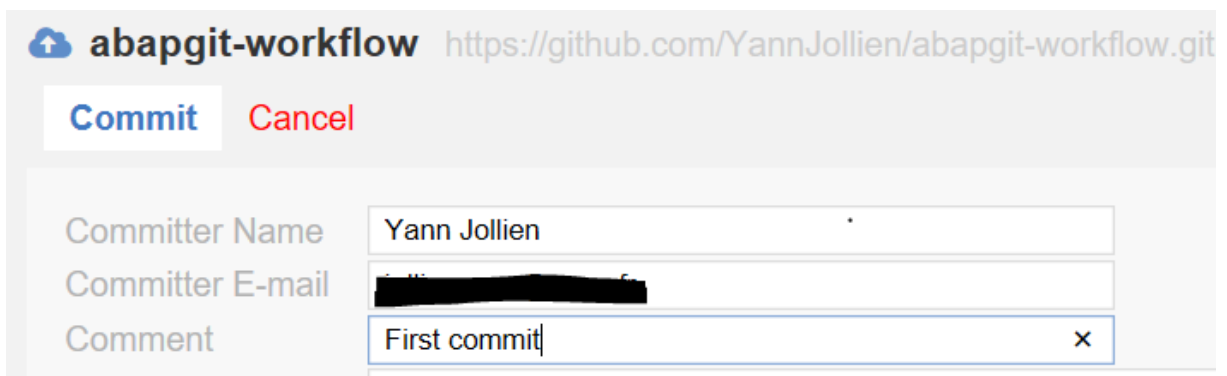
¹¹⁷ Source : Donnée de l'auteur sur SAP GUI

¹¹⁸ Source : Donnée de l'auteur sur SAP GUI

¹¹⁹ Source : Donnée de l'auteur sur SAP GUI

¹²⁰ Source : Donnée de l'auteur sur SAP GUI

10. Remplir les champs « Committer E-mail » et « Comment » et valider avec « Commit » ;



The screenshot shows the GitHub commit page for the repository 'abapgit-workflow' at the URL 'https://github.com/YannJollien/abapgit-workflow.git'. At the top, there are 'Commit' and 'Cancel' buttons. Below, there are three input fields: 'Committer Name' with the value 'Yann Jollien', 'Committer E-mail' with a redacted email address, and 'Comment' with the text 'First commit'. A small 'x' icon is visible in the bottom right corner of the comment field.

Figure 123 : Ajout élément ABAP au dépôt - Commit¹²¹

Entrer les données liées au compte GitHub et ouvrir le dépôt pour vérifier si la classe a été ajoutée ;

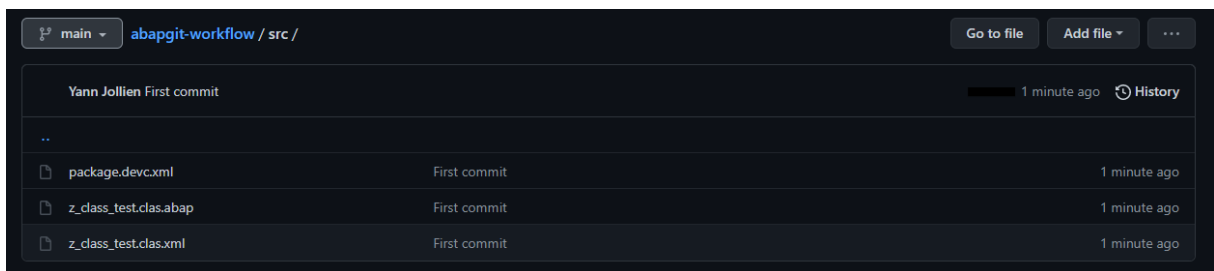


Figure 124 : Ajout élément ABAP au dépôt - Check¹²²

¹²¹ Source : Donnée de l'auteur sur SAP GUI

¹²² Source : Donnée de l'auteur sur GitHub

8.15 Installer plugin abapGit sur Eclipse

1. Sur Eclipse ajouter un nouvel outil : *Help > Install New Software* ;

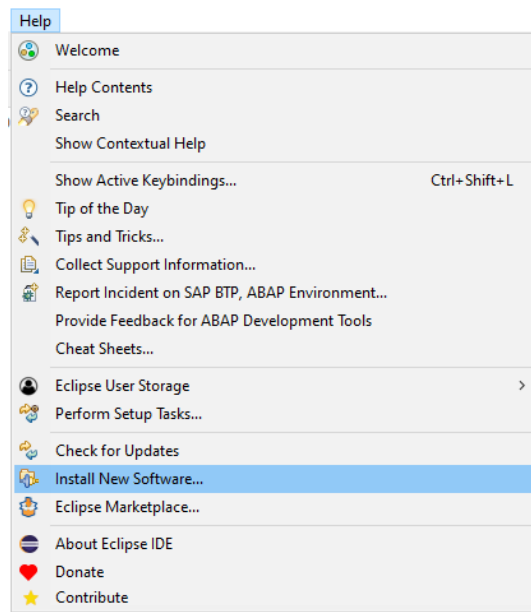


Figure 125 : Installer plugin abapGit sur Eclipse - Ajouter nouvel outil¹²³

2. Ajouter l'URL <http://eclipse.abapgit.org/updatesite/> et cocher « abapGit for ABAP Development Tools (ADT) » ;

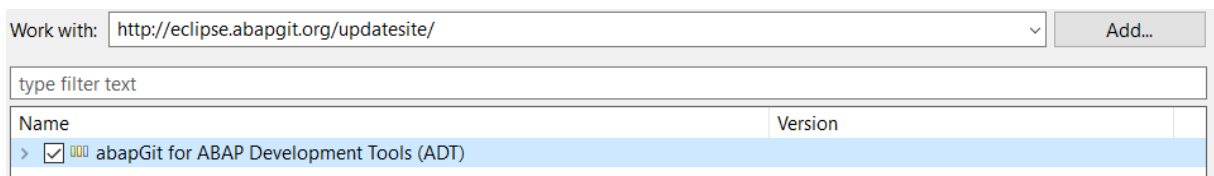


Figure 126 : Installer plugin abapGit sur Eclipse - Ajout URL¹²⁴

3. Valider avec 2x « Next », accepter les conditions de licence et terminer avec « Finish » (Eclipse doit être redémarré après l'opération)

¹²³ Source : Donnée de l'auteur sur Eclipse

¹²⁴ Source : Donnée de l'auteur sur Eclipse

4. Ouvrir le dépôt Git sur Eclipse : *Window > Show > Other > abapGit Repositories* ;

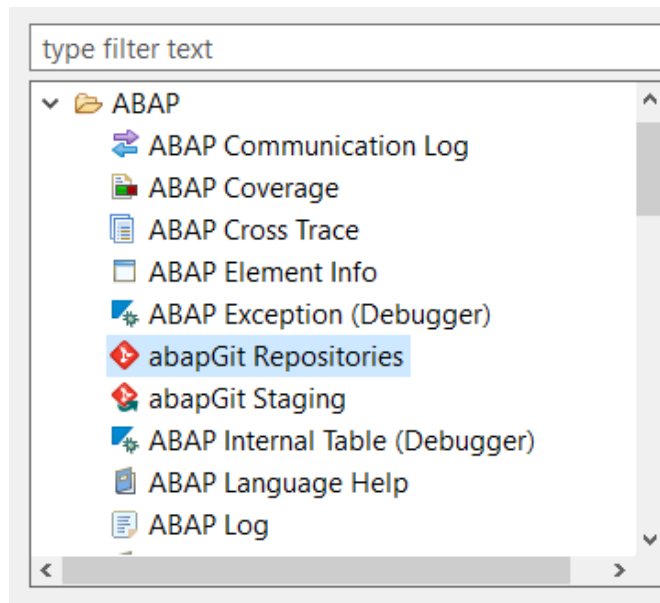


Figure 127 : Installer plugin abapGit sur Eclipse - Changer vue¹²⁵

8.16 Cloner dépôt Git dans l'SAP Cloud

Pour cette étape, deux scénarios sont possibles ; ajouter le code local à un projet ABAP cloud existant ou créer un nouveau projet ABAP cloud. Pour simplifier l'opération, nous allons utiliser le même projet que celui du scénario 1 (ZTRAVEL_APP_002). Dans un environnement professionnel, un nouveau projet ABAP cloud doit être créé pour y ajouter les éléments désirés du système local.

1. Sélectionner le projet cloud et appuyer sur le bouton d'ajout dans abapGit Repositories (Logo croix verte tout à gauche) ;

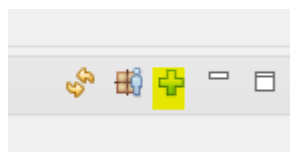


Figure 128 : Cloner dépôt Git dans l'SAP Cloud - Ajout dépôt¹²⁶

¹²⁵ Source : Donnée de l'auteur sur Eclipse

¹²⁶ Source : Donnée de l'auteur sur Eclipse

2. Ajouter l'URL du dépôt GitHub et valider la suite avec « Next » (Ne pas toucher à l'ordre de transport) et finir avec « Finish » ;

Link abapGit Repository

Specify a abapGit repository



URL: *	<input type="text" value="https://github.com/YannJollien/abapgit-workflow.git"/>
--------	--

Figure 129 : Cloner dépôt Git dans l'SAP Cloud - Ajout URL¹²⁷

3. Récupérer la classe pour pouvoir l'ajouter à notre project cloud : Clic droit sur notre dépôt (nous sommes toujours dans la vue abapGit Repositories) > Pull... ;

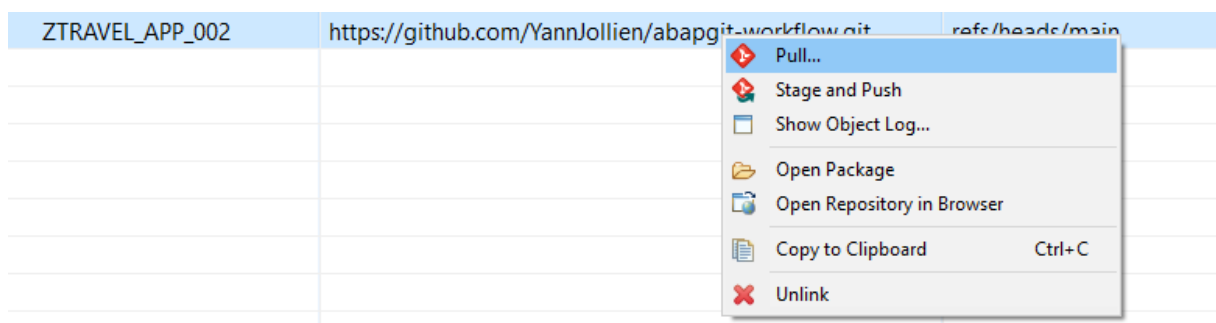


Figure 130 : Cloner dépôt Git dans l'SAP Cloud - Pull¹²⁸

¹²⁷ Source : Donnée de l'auteur sur Eclipse

¹²⁸ Source : Donnée de l'auteur sur Eclipse

4. Vérifier si nous avons pu récupérer la classe : *Expandre le projet cloud > Package ZTRAVEL_APP_002 > Source Code Library > Classes ;*

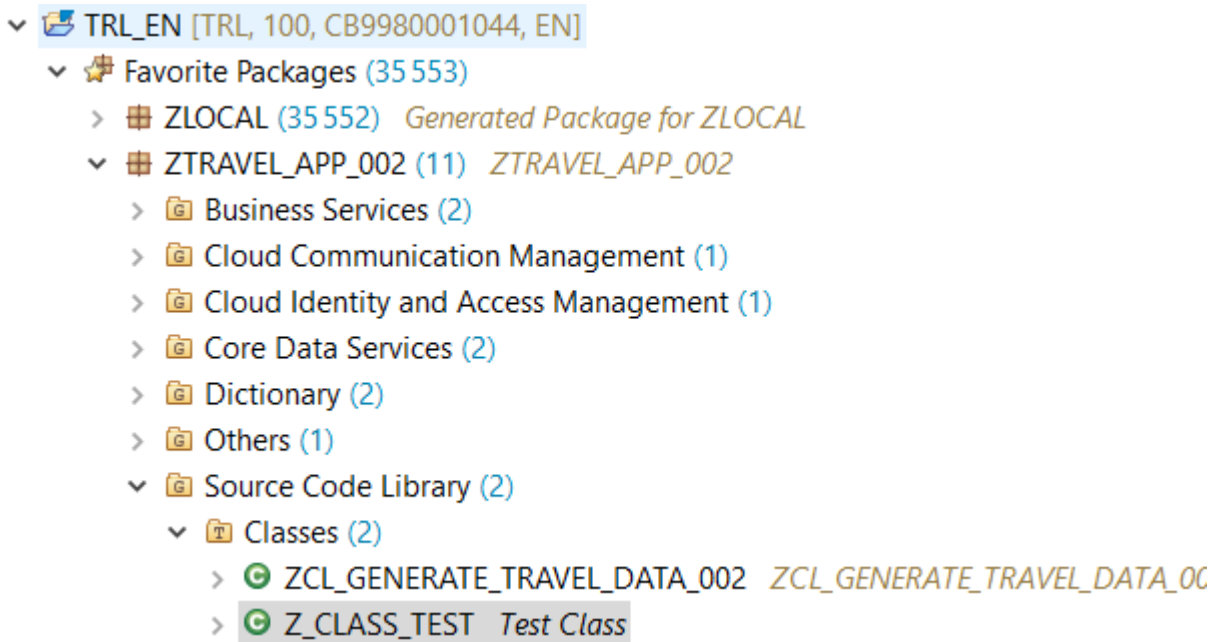


Figure 131 : Cloner dépôt Git dans l'SAP Cloud - Vérifier¹²⁹

Nous avons réussi à migrer une classe SAP depuis le système local vers la SCP.

¹²⁹ Source : Donnée de l'auteur sur Eclipse

9. Conclusion

9.1 Synthèse pratique

Les deux scénarios illustrés à l'aide de deux prototypes nous permettent d'évaluer la partie pratique. D'une manière générale, la création d'un projet cloud et la migration d'un système local vers le cloud se sont déroulés sans problème majeur. Nous étions surpris par la complexité de configuration d'abapGit sur le système local, requérant des configurations diverses autour du protocole SSL et de certificats. Contrairement à nos attentes, l'installation d'abapGit nécessite des interventions plus approfondies dans le système SAP, pouvant provoquer certains problèmes, notamment au niveau des accès des différents utilisateurs. Cette opération liée à la configuration du protocole et à l'ajout des certificats de sécurité est nécessaire sur tous les systèmes utilisant abapGit.

L'inscription à la SCP s'est avérée plus compliquée que prévu. La double authentification requière un numéro de portable pour valider l'inscription avec un code SMS. Pour notre part, SAP n'a pas reconnu notre numéro suisse et nous avons dû contacter le support SAP pour activer notre compte. Il faut prendre en considération qu'il s'agit d'une inscription à un compte « trial » et que ce problème n'est pas forcément lié au numéro suisse. SAP nous a garanti que les numéros Suisses étaient bien supportés pour la double authentification.

Le travail avec Eclipse est plutôt agréable, bien que certains aspects manquent par rapport au système local. Il n'est par exemple pas possible d'afficher des tableaux en double cliquant sur ces derniers, outil très utile disponible sur le SAP GUI. Le plugin abapGit sur Eclipse permet de visualiser les dépôts d'une façon superficielle, sans pouvoir en observer les détails. Les différentes sous-catégories avec les différents éléments ABAP sont bien visibles et structurées d'une manière logique et agréable.

Il s'agit de scénarios simples et basiques, il est donc difficile d'évaluer la performance et la migration de systèmes plus grands. Nous n'avons pas pu couvrir tout le cycle de développement d'une application ABAP, notamment la partie test et production des outils sont disponibles au niveau cloud et il est également possible de conserver l'utilisation du SAP ChaRM.

9.2 Synthèse générale

L'introduction de la SCP est en premier lieu une démarche stratégique de la part de SAP. Elle a comme objectif de créer une base technologique unitaire pour toutes les applications SAP. Le cloud livre un environnement qui permet de centraliser toutes les parties de SAP et

visé à offrir une meilleure possibilité d'intégration et d'avoir un écosystème homogène.

Durant nos recherches, nous nous sommes souvent retrouvés avec des articles mettant en avant les possibilités de la nouvelle technologie cloud, sans vraiment montrer son intérêt et ses avantages par rapport à un système cloud. L'aspect marketing est frappant et les solutions offertes dans le cloud ne sont souvent pas compatibles avec les programmes développés en local ou alors elles demandent des adaptations. Il en est de même pour le développement ABAP sur le cloud. Bien que le langage de base ABAP reste le même, le développement dans le cloud exige des adaptations et des changements dans la manière de faire.

Les ABAP Development Tools nous ont beaucoup plu, car ils offrent une possibilité variée de développer en ABAP sur un IDE comme Eclipse. Ils sont pleinement fonctionnels et nous n'avons pas constaté de lacunes majeures par rapport au développement SAP GUI. Ici aussi, il faut s'attendre à des changements, car certains aspects comme le développement de programmes de type rapport ne sont plus supportés. A notre avis, ceci n'est pas une grosse perte puisque le développement d'interfaces se dirige vers du Fiori. Cependant nous devons en tenir compte.

L'utilisation de la SCP s'adresse aux entreprises souhaitant abandonner la maintenance de leurs systèmes. Gérer ses propres serveurs, mettre à jour le système manuellement, sécuriser les systèmes, tout cela sont des éléments qui disparaissent en utilisant un service cloud. L'aspect financier doit être analysé afin d'avoir une réponse concrète quant à la rentabilité de l'une ou l'autre solution.

La solution SCP a certainement du potentiel d'avenir, mais aujourd'hui, elle représente à notre sens des risques qui ne justifient pas une migration complète. L'écosystème homogène que vise SAP n'est pas encore là où il devrait être et beaucoup d'aspects de développement engendrent trop d'investissements pour arriver à un même résultat que sur le développement en local. Il ne s'agit pas d'un développement similaire sur le cloud par rapport au développement local, mais d'une nouvelle manière de faire qui va donc demander du temps d'adaptation et un temps pour apprendre les nouvelles pratiques à adopter.

Il ne faut pas oublier que ce travail couvre une partie superficielle de la SCP et que des analyses plus précises par rapport aux besoins réels et concrets doivent être effectuées, avant de prendre une décision définitive. Nous restons sur notre position et proposons une migration progressive pour se familiariser avec les nouveaux outils et pratiques. La prise de risque est minime et permet de s'acclimater avec le cloud.

9.3 Synthèse personnelle

Le choix de réaliser mon travail de bachelor sur un sujet SAP a été facile pour moi, puisque je cherchais un sujet orienté technique. Ayant étudié SAP durant mon parcours scolaire, le développement ABAP m'a beaucoup plu et m'a permis de rajouter un langage de programmation particulier à mon répertoire. Bien que ce langage de programmation et la manière de développer sur SAP est bien particulière, elle a toutefois des similitudes avec les langages de programmations plus classiques. En choisissant ce sujet, je voulais développer mes connaissances en SAP et l'aspect du développement dans le cloud était un point attrayant.

Ce travail m'a permis de me familiariser avec la notion du PaaS et de comprendre toute sa complexité ainsi qu'avec l'écosystème SAP. D'un point de vue économique, le travail réalisé m'a fait prendre conscience des enjeux pour une entreprise et la difficulté de faire les bons choix. Une nouvelle solution peut paraître intéressante à première vue, mais il est nécessaire de l'analyser minutieusement avant de prendre une décision.

Les connaissances acquises durant les trois années de formation en gestion de projet m'ont également permis de bien organiser le travail, ce qui était précieux, car je n'ai pas eu de difficultés à respecter les délais imposés. L'utilisation de la méthodologie agile représentait un soutien supplémentaire.

La réalisation de ce travail de bachelor m'a beaucoup plu et m'a permis d'enrichir mes connaissances techniques. Pouvoir réaliser un travail pour un mandant tout en sachant qu'il allait servir de support de décision était une motivation supplémentaire. Il était non seulement question de rédiger un travail sur un sujet mais également de comprendre les besoins du mandant et de pouvoir répondre aux différentes questions de ce dernier. Rendre ce document après plusieurs semaines de travail me remplit de satisfaction et j'espère qu'il en est de même pour le mandant.

Bibliographie

- abapGit. (2021). *SSL setup*. Récupéré sur <https://docs.abapgit.org/guide-ssl-setup.html>
- Allen, B. (2018, 03 08). *On Premise VS Cloud SAP: What to choose among them?* Récupéré sur <https://gocoding.org/sap-on-premise-vs-cloud/>
- Astashonak, P. (2017, 05 18). *SAP ABAP Data Dictionary*. Récupéré sur <https://erproof.com/abap/sap-abap-training/sap-abap-data-dictionary/>
- Augsten, S. (2018, 02 16). *Was ist das Wasserfallmodell?* Récupéré sur <https://www.dev-insider.de/was-ist-das-wasserfallmodell-a-680501/>
- Azmi, A. (2018, 10 06). *Why ABAP will kill SAP Cloud Platform*. Récupéré sur <https://www.linkedin.com/pulse/why-abap-kill-sap-cloud-platform-ahmed-azmi/>
- Bastien. (2019, 06 21). *Cloud Gaming : tout savoir sur la technologie qui révolutionne le jeu vidéo*. Récupéré sur <https://www.lebigdata.fr/cloud-gaming-tout-savoir#:~:text=Le%20Cloud%20Gaming%20est%20une,savoir%20sur%20cette%20nouvelle%20technologie%E2%80%A6>
- Cesljar, D. (2019, 09 05). *Datenschutz und Sicherheit in der Cloud*. Récupéré sur <https://customer-first-cloud.de/customer-data-cloud/datenschutz-und-sicherheit-in-der-cloud/>
- Chouffani, R. (2019, 10 07). *Pros and cons of SAP migration to the cloud*. Récupéré sur <https://searchsap.techtarget.com/tip/Pros-and-cons-of-SAP-migration-to-the-cloud>
- Courtemanche, M., Mell, E., & Gillis, A. (2020, 09). *What is DevOps? The ultimate guide*. Récupéré sur <https://searchitoperations.techtarget.com/definition/DevOps>
- Djomo, C. T. (2019, 10 25). *Getting Started with the ABAP RESTful Application Programming Model (RAP)*. Récupéré sur <https://blogs.sap.com/2019/10/25/getting-started-with-the-abap-restful-programming-model/>
- Finley, K. (2012, 07 14). *What Exactly Is GitHub Anyway?* Récupéré sur <https://techcrunch.com/2012/07/14/what-exactly-is-github-anyway/>
- Flothkoetter, N. (2021, 04 16). *Use abapGit to Transfer Your On-Premise ABAP Source Code to the Cloud*. Récupéré sur <https://developers.sap.com/tutorials/abap-environment-abapgit.html>

- guru99. (2021, 05 10). *Remote Function Call (RFC) in SAP Tutorial*. Récupéré sur <https://www.guru99.com/rfc-remote-function-call.html>
- guru99. (2021, 05 10). *What is SAP Transport Request? How to Import/Export TR*. Récupéré sur <https://www.guru99.com/what-is-transport-request-how-to-importexport-it-check-logs.html>
- Hübner, G. (2017, 11). *Securing the Cloud with SAP Cloud Platform* . Récupéré sur <https://assets.cdn.sap.com/sapcom/docs/2017/11/a837448f-de7c-0010-82c7-eda71af511fa.pdf>
- Johari, A. (2021, 06 01). *How To Use GitHub - Developers Collaboration Using GitHub*. Récupéré sur <https://www.edureka.co/blog/how-to-use-github/>
- Kastner, E. (2020, 01 24). *Cloud Security vs. on-premise security: which is better?* Récupéré sur <https://www.soscanhelp.com/blog/cloud-security-vs-on-premise-security>
- Kempen, D. v. (2020, 03 23). *Getting Started with ABAP in the Cloud - Hands-On*. Récupéré sur <https://blogs.sap.com/2020/03/23/getting-started-with-abap-in-the-cloud-part-ii/>
- Kessler, K. (2018). *Develop New Cloud Applications in ABAP with SAP Cloud Platform ABAP Environment* . Récupéré sur <https://assets.cdn.sap.com/sapcom/docs/2018/09/06c97f15-1e7d-0010-87a3-c30de2ffd8ff.pdf>
- Kessler, K. (2019). *New Features for SAP Cloud Platform ABAP Environment*. Récupéré sur <https://www.sapinsideronline.com/new-features-for-sap-cloud-platform-abap-environment/>
- Kessler, K. (2020, 08 17). *Make the Move to SAP Cloud Platform ABAP Environment with Confidence*. Récupéré sur <https://www.sapinsideronline.com/make-the-move-to-sap-cloud-platform-abap-environment-with-confidence/>
- Kriegbaum, P. (2020, 05 05). *Wie abapGit Ihre SAP-Entwicklung retten kann*. Récupéré sur <https://erlebe-software.de/abap-und-co/wie-abapgit-ihre-sap-entwicklung-retten-kann/>
- Lawton, G. (2018, 09 03). *SAP Cloud Platform: Vor- und Nachteile der Integrations-Tools*. Récupéré sur <https://www.computerweekly.com/de/feature/SAP-Cloud-Platform-Vor-und-Nachteile-der-Integrations-Tools>
- Lordieck, C. (2019, 09 20). *SAP Cloud Platform ABAP Environment*. Récupéré sur

<https://erlebe-software.de/knowhow/sap-cloud-platform-abap-environment/#vorteile-der-sap-cloud-platform-abap-environment>

Luber, S., & Karlstetter, F. (2019, 06 13). *Was ist die SAP Cloud Platform?* Récupéré sur <https://www.cloudcomputing-insider.de/was-ist-die-sap-cloud-platform-a-836321/>

Luber, S., & Schmitz, P. (2017, 07 24). *Was ist ISO 27001?* Récupéré sur <https://www.security-insider.de/was-ist-iso-27001-a-626958/>

Makadia, M. (2021, 03 12). *9 Key Benefits of DevOps*. Récupéré sur <https://www.business2community.com/business-intelligence/9-key-benefits-of-devops-02391855>

McAfee. (2021). *What is PaaS?* Récupéré sur <https://www.mcafee.com/enterprise/de-de/security-awareness/cloud/what-is-paas.html>

Minh, N. H. (2020, 02 16). *What is Eclipse IDE? - for beginners*. Récupéré sur <https://www.codejava.net/ides/eclipse/what-is-eclipse-ide-for-beginner>

Morrison, T. (2018, 09). *Fieldglass*. Récupéré sur <https://searchsap.techtarget.com/definition/Fieldglass>

Nair, S. V. (2013, 10 03). *Let's Talk OData, Shall We? And keep it simple please!* Récupéré sur <https://blogs.sap.com/2013/10/03/lets-talk-odata-shall-we/>

Nebeling, N. (2020, 04 14). *Seite an Seite mit ABAP in der Cloud: Side-by-Side Extensions*. Récupéré sur <https://erlebe-software.de/abap-und-co/seite-an-seite-mit-abap-in-der-cloud-side-by-side-extensions/>

Netinbag. (2021). *Qu'est-ce que ABAP?* Récupéré sur <https://www.netinbag.com/fr/internet/what-is-abap.html>

Nurlisa Zolkipli, N., Ngah, A., & Deraman, A. (2018, 01). *Version Control System: A Review*. Récupéré sur https://www.researchgate.net/publication/327291907_Version_Control_System_A_Review

Obert, M. (2018, 10 08). *Using the Destination service in SAP BTP, Cloud Foundry Environment*. Récupéré sur <https://blogs.sap.com/2018/10/08/using-the-destination-service-in-the-cloud-foundry-environment/>

- Oracle. (2021). *Qu'est-ce qu'un PaaS*. Récupéré sur <https://www.oracle.com/fr/cloud/definition-paas/>
- Parizo, C. (2019, 06 05). *SAP Cloud Platform ABAP Environment bringt ABAP in die Cloud*. Récupéré sur <https://www.computerweekly.com/de/feature/SAP-Cloud-Platform-ABAP-Environment-bringt-ABAP-in-die-Cloud>
- Press, S. (2021, 03 29). *ABAP in the Cloud: Getting to Know SAP BTP, ABAP Environment*. Récupéré sur <https://blog.sap-press.com/abap-in-the-cloud-getting-to-know-sap-btp-abap-environment>
- Ritzau, V. (2021, 05 07). *ABAP Development Tools (ADT) in Eclipse: Warum es sich zu lernen lohnt*. Récupéré sur <https://cpro-ips.com/blog/2021/05/07/abap-development-tools-adt-in-eclipse-warum-es-sich-zu-lernen-lohnt/>
- Sacolick, I. (2020, 02 25). *What is agile methodology? Modern software development explained*. Récupéré sur <https://www.infoworld.com/article/3237508/what-is-agile-methodology-modern-software-development-explained.html>
- SAP. (2016). *ICF - Internet Communication Framework*. Récupéré sur https://help.sap.com/doc/abapdocu_750_index_htm/7.50/de-de/abenicf.htm
- SAP. (2017). *SAP Cloud Platform*. Récupéré sur <https://www.sapstore.com/medias/SAP-cloud-platform.pdf?context=bWFzdGVyfHBkZnN8MTQ1MTkzMDF8YXBwbGljYXRpb24vcGRmfHBkZnMvaGU0L2g3Ny84ODg2NjQ2MTQ1MDU0LnBkZnwwZWExNjI5NTc4Y2Q0N2VhNzg0ZGY2OWM5MmQ3ZDQzMWNjMWQ1MmYxMjYyYjE3YTE5ODImZWU5NjM5MzJhMzE4>
- SAP. (2017, 08 04). *SAP Cloud Platform Security: Safeguarding Your Business*. Récupéré sur <https://blogs.sap.com/2017/08/04/sap-cloud-platform-security-safeguarding-your-business/>
- SAP. (2021). *ABAP Development Tools*. Récupéré sur <https://tools.hana.ondemand.com/#abap>
- SAP. (2021, 06 03). *Enable HTTP Communication in Your ABAP Code*. Récupéré sur <https://help.sap.com/viewer/65de2977205c403bbc107264b8eccf4b/Cloud/en-US/cef1ada754154d11b5701ab60e6ab412.html>
- SAP. (2021, 04 12). *Feature Scope Description for SAP BTP*. Récupéré sur

https://help.sap.com/doc/5e8107bf49684962b897217040398007/Cloud/en-US/SAP_BTP_FSD.pdf

- SAP. (2021, 03). *Rebranding of SAP Cloud Platform*. Récupéré sur <https://www.sap.com/swiss/products/cloud-platform.html?pdf-asset=bc275d93-d07d-0010-87a3-c30de2ffd8ff&page=1>
- SAP. (2021, 06 03). *SAP BTP Cockpit*. Récupéré sur https://help.sap.com/doc/d20fc4d5c5d5420387138610c5a648ac/Cloud/en-US/SAP_Cloud_Platform_Neo.pdf
- SAP. (2021). *Tips and Tricks (ABAP Core Tools)*. Récupéré sur <https://help.sap.com/viewer/5371047f1273405bb46725a417f95433/Cloud/en-US/df7d89c16c014323b6bae473058298c3.html>
- SAP. (2021). *What is SAP Business Application Studio?* Récupéré sur <https://help.sap.com/webcomponents/products/SAP%20Business%20Application%20Studio/9d1db9835307451daa8c930fbd9ab264/8f46c6e6f86641cc900871c903761fd4.html?locale=en-US>
- SAP. (s.d.). *Developing a UI Service with Access to a Remote Service*. Récupéré sur <https://help.sap.com/viewer/923180ddb98240829d935862025004d6/Cloud/en-US/f4969e551d3049c59715210cbeb4ef56.html>
- Schmidt, K. (2018, 11 05). *Fünf gute Gründe für SAP Cloud Platform*. Récupéré sur <https://news.sap.com/germany/2018/11/cloud-platform-vorteile-umstieg/>
- Snapp, S. (2021, 03 21). *How to Understand the Problem with ABAP in the Cloud*. Récupéré sur <https://www.brightworkresearch.com/how-to-understand-the-problem-with-abap-in-the-cloud/>
- Spiegel, K. (2019, 11 14). *gCTS is here*. Récupéré sur <https://blogs.sap.com/2019/11/14/gcts-is-here/>
- Stevens, H. (2020, 04 27). *How to use the integration of SAP Cloud Platform Transport Management into SAP ChaRM*. Récupéré sur <https://blogs.sap.com/2020/04/27/how-to-use-the-integration-of-sap-cloud-platform-transport-management-into-sap-solution-manager-change-request-management-and-quality-gate-management/>
- Tutorialspoint. (2019, 03 04). *SAP Cloud Platform*. Récupéré sur https://www.tutorialspoint.com/sap_cloud_platform/sap_cloud_platform_tutorial.pdf

Wahl, F. (2020, 03 19). *SAP Cloud Platform, ABAP Environment: Development Opportunities. SAP TechEd Talk*. Récupéré sur <https://www.youtube.com/watch?v=c3oRSjnxtr8>

Wulff, E. (2018, 01 11). *abapGit*. Récupéré sur <https://tricktresor.de/blog/abapgit/>

Zarske, B. (2019, 12 12). *DevOps with SAP BTP: Develop & Test*. Récupéré sur <https://blogs.sap.com/2019/12/12/devops-with-sap-cloud-platform-develop-test/>

Annexe I : Script Database Table

Database Table: ZTRAVEL_002

```
@EndUserText.label : 'Database Table – Travel'
@AbapCatalog.enhancementCategory : #NOT_EXTENSIBLE
@AbapCatalog.tableCategory : #TRANSPARENT
@AbapCatalog.deliveryClass : #A
@AbapCatalog.dataMaintenance : #LIMITED
define table ztravel_002 {
  key client      : abap.clnt not null;
  key mykey       : sysuuid_x16 not null;
  travel_id       : /dmo/travel_id;
  agency_id       : /dmo/agency_id;
  customer_id     : /dmo/customer_id;
  begin_date      : /dmo/begin_date;
  end_date        : /dmo/end_date;
  @Semantics.amount.currencyCode : 'ztravel_002.currency_code'
  booking_fee     : /dmo/booking_fee;
  @Semantics.amount.currencyCode : 'ztravel_002.currency_code'
  total_price     : /dmo/total_price;
  currency_code   : /dmo/currency_code;
  description     : /dmo/description;
  overall_status  : /dmo/overall_status;
  created_by      : syuname;
  created_at      : timestamp;
  last_changed_by : syuname;
  last_changed_at : timestamp;
}
```

Annexe II : Script ABAP Class

ABAP Class: ZCL_GENERATE_TRAVEL_DATA_002

```
CLASS zcl_generate_travel_data_002 DEFINITION
  PUBLIC
  FINAL
  CREATE PUBLIC .

  PUBLIC SECTION.
    INTERFACES if_oo_adt_classrun.
  PROTECTED SECTION.
  PRIVATE SECTION.
ENDCLASS.
```

```
CLASS zcl_generate_travel_data_002 IMPLEMENTATION.
```

```
  METHOD if_oo_adt_classrun~main.
    DATA: itab TYPE TABLE OF ztravel_002.
```

```
  * read current timestamp
```


GET TIME STAMP FIELD DATA(zv_tsl).

```
* fill internal travel table (itab)
itab = VALUE #(
  ( travel_id = '00000001' agency_id = '070010' customer_id = '000011' begin_date =
'20200310' end_date = '20200317' booking_fee = '17.00' total_price = '800.00' currency_code
= 'EUR'
    description = 'Need a break!' overall_status = 'O' created_by = 'CB00000000007'
created_at   = '20200310105654.4296640' last_changed_by   = 'CB00000000007'
last_changed_at = '20200310111041.2251330' )
).

* delete existing entries in the database table
DELETE FROM ztravel_002.

* insert the new table entries
INSERT ztravel_002 FROM TABLE @itab.

* check the result
SELECT * FROM ztravel_002 INTO TABLE @itab.
Out->write( sy-dbcnt ).
Out->write( 'Travel data inserted successfully!' ).

ENDMETHOD.

ENDCLASS.
```

Annexe III : Script CDS View

Data Definition (Interface View): ZI_TRAVEL_002

```
@AbapCatalog.sqlViewName: 'ZVI_TRAVEL_002'
@AbapCatalog.compiler.compareFilter: true
@AbapCatalog.preserveKey: true
@AccessControl.authorizationCheck: #CHECK
@EndUserText.label: 'Interface View – Travel'
define root view ZI_TRAVEL_002
  as select from ztravel_002

  association [0..1] to /DMO/I_Agency as _Agency on $projection.agency_id =
_Agency.AgencyID
  association [0..1] to /DMO/I_Customer as _Customer on $projection.customer_id =
_Customer.CustomerID
  association [0..1] to I_Currency as _Currency on $projection.currency_code =
_Currency.Currency

{
  key mykey,
    travel_id,
    agency_id,
    customer_id,
    begin_date,
    end_date,
```

```

    @Semantics.amount.currencyCode: 'currency_code'
    booking_fee,
    @Semantics.amount.currencyCode: 'currency_code'
    total_price,
    @Semantics.currencyCode: true
    currency_code,
    overall_status,
    description,

    /*-- Admin data --*/
    @Semantics.user.createdBy: true
    created_by,
    @Semantics.systemDateTime.createdAt: true
    created_at,
    @Semantics.user.lastChangedBy: true
    last_changed_by,
    @Semantics.systemDateTime.lastChangedAt: true
    last_changed_at,

    /* Public associations */
    _Agency,
    _Customer,
    _Currency
}

```

Annexe IV : Script projection View

Data Definition (Projection View): ZC_TRAVEL_002

```

@AccessControl.authorizationCheck: #CHECK
@EndUserText.label: 'Projection View – Travel'

@UI: {
  headerInfo: { typeName: 'Travel', typeNamePlural: 'Travels', title: { type: #STANDARD,
value: 'TravelID' } } }

@Search.searchable: true
define root view entity ZC_TRAVEL_002
as projection on ZI_TRAVEL_002
{

  @UI.facet: [ { id:      'Travel',
                  purpose: #STANDARD,
                  type:    #IDENTIFICATION_REFERENCE,
                  label:   'Travel',
                  position: 10 } ]

  key mykey      as TravelUUID,
  @UI: {
    lineltem:    [ { position: 10, importance: #HIGH } ],
    identification: [ { position: 10, label: 'Travel ID [1,...,99999999]' } ] }
  @Search.defaultSearchElement: true
  travel_id      as TravelID,

```

```

    @UI: {
        lineItem: [ { position: 20, importance: #HIGH } ],
        identification: [ { position: 20 } ],
        selectionField: [ { position: 20 } ] }
    @Consumption.valueHelpDefinition: [{ entity : {name: '/DMO/I_Agency', element:
'AgencyID' } } ]

    @ObjectModel.text.element: ['AgencyName']
    @Search.defaultSearchElement: true
    agency_id      as AgencyID,
    _Agency.Name   as AgencyName,

    @UI: {
        lineItem: [ { position: 30, importance: #HIGH } ],
        identification: [ { position: 30 } ],
        selectionField: [ { position: 30 } ] }
    @Consumption.valueHelpDefinition: [{ entity : {name: '/DMO/I_Customer', element:
'CustomerID' } } ]

    @ObjectModel.text.element: ['CustomerName']
    @Search.defaultSearchElement: true
    customer_id    as CustomerID,

    @UI.hidden: true
    _Customer.LastName as CustomerName,

    @UI: {
        lineItem: [ { position: 40, importance: #MEDIUM } ],
        identification: [ { position: 40 } ] }
    begin_date     as BeginDate,

    @UI: {
        lineItem: [ { position: 41, importance: #MEDIUM } ],
        identification: [ { position: 41 } ] }
    end_date       as EndDate,

    @UI: {
        lineItem: [ { position: 50, importance: #MEDIUM } ],
        identification: [ { position: 50, label: 'Total Price' } ] }
    @Semantics.amount.currencyCode: 'CurrencyCode'
    total_price    as TotalPrice,

    @Consumption.valueHelpDefinition: [{entity: {name: 'I_Currency', element: 'Currency'
}}}
    currency_code  as CurrencyCode,

    @UI: {
        lineItem: [ { position: 60, importance: #HIGH },
                    { type: #FOR_ACTION, dataAction: 'acceptTravel', label: 'Accept Travel'
} ],
        identification: [ { position: 60, label: 'Status [O(Open)|A(Accepted)|X(Canceled)]' } ] }
    overall_status as TravelStatus,

```

```

    @UI.identification: [ { position: 70, label: 'Remarks' } ]
    description      as Description,

    @UI.hidden: true
    last_changed_at  as LastChangedAt

}

```

Annexe V : Script service OData

```

Service Definition: ZSD_TRAVEL_002


---


@EndUserText.label: 'Service Definition – Travel'
define service ZSD_TRAVEL_002 {
    expose ZC_TRAVEL_002 as TravelProcessor;
    expose /DMO/I_Customer as Passenger;
    expose /DMO/I_Agency as TravelAgency;
    expose /DMO/I_Airport as Airport;
    expose I_Currency as Currency;
    expose I_Country as Country;
}

```

Annexe VI : Accès SAP Gui

L'accès au système GUI nous a été fourni par l'école, il a été utilisé pour le deuxième scénario pratique. Une connexion au VPN de la HES-SO doit être établie pour son fonctionnement. Il s'agit d'une connexion locale établie avec le SAP Logon 760 avec les données suivantes ;

SERVER

Host vlhsapt02 (ABAP GBI)

IP Address 153.109.124.148

SAP SID T02

Instance 02

NetWeaver 750

SAP MC <http://153.109.124.148:50213>

SAP User

T02 Client 100

User = BACHELOR

Pwd = sapbachelor

PACKAGE

SE80

- Package = ZP_CLOUD

Transport = T02K900064

Annexe VII : Cahier des charges

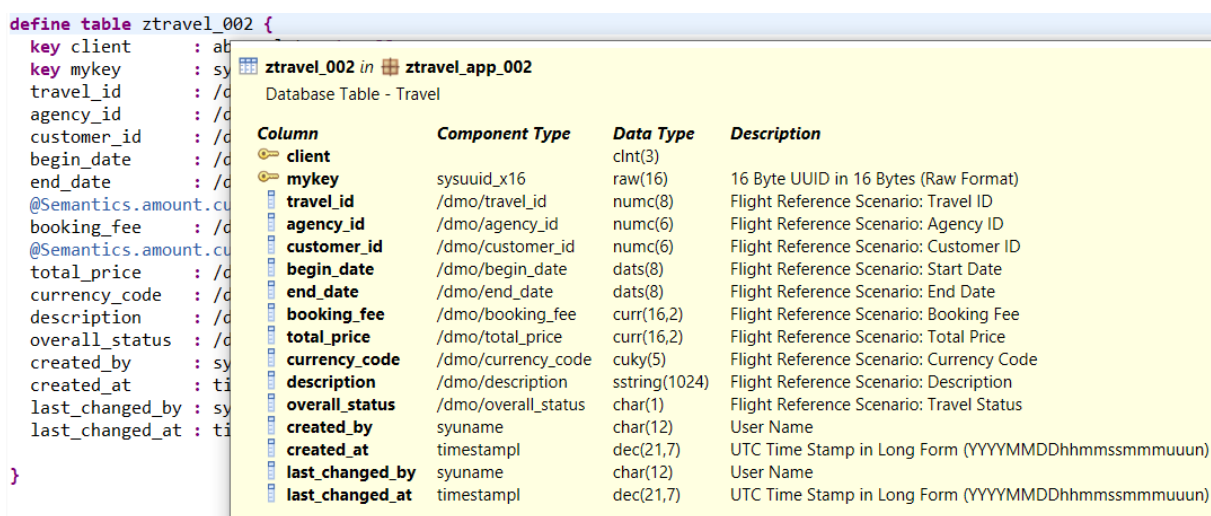
Tâches à effectuer pour le travail de bachelor selon document « Données du travail de Bachelor » et après consultation avec le mandant :

- Explications des technologies utilisées
- Explications générales sur le sujet
- Explications de la SAP Cloud Platform
- Aperçu et explications du développement ABAP sur SCP
- Différences entre ABAP dans le cœur du système et ABAP sur SCP
- Aperçu de l'architecture SCP
- Concept de développement ABAP sur SCP
- Déploiement ABAP sur SCP
- Environnement de test sur SCP
- Outils de développement sur SCP
- Avantages et inconvénients de la SCP
- Prise de position relative à la meilleure solution pour le développement ABAP
- Création d'un prototype

Annexe VIII : Aide au développement avec les outils ADT

Pour faciliter le développement ABAP sur Eclipse, plusieurs outils sont mis à disposition. Voici un aperçu des outils fréquemment utilisés (SAP, Tips and Tricks (ABAP Core Tools), 2021) ;

Avec la touche F2, des informations avancées pour un élément ABAP peuvent être affichées. On peut comparer ceci à un double-clic sur un élément ABAP sur le SAP GUI menant à des informations détaillées ;



```
define table ztravel_002 {
  key client      : abapkey
  key mykey       : sysuid
  travel_id      : /dmo/travel_id
  agency_id      : /dmo/agency_id
  customer_id    : /dmo/customer_id
  begin_date     : /dmo/begin_date
  end_date       : /dmo/end_date
  @Semantics.amount.currency_code : /dmo/amount_currency_code
  booking_fee    : /dmo/booking_fee
  @Semantics.amount.currency_code : /dmo/amount_currency_code
  total_price    : /dmo/total_price
  currency_code  : /dmo/currency_code
  description     : /dmo/description
  overall_status : /dmo/overall_status
  created_by     : syuname
  created_at     : timestamp
  last_changed_by : syuname
  last_changed_at : timestamp
}
```

Column	Component Type	Data Type	Description
client		clnt(3)	
mykey	sysuid_x16	raw(16)	16 Byte UUID in 16 Bytes (Raw Format)
travel_id	/dmo/travel_id	numc(8)	Flight Reference Scenario: Travel ID
agency_id	/dmo/agency_id	numc(6)	Flight Reference Scenario: Agency ID
customer_id	/dmo/customer_id	numc(6)	Flight Reference Scenario: Customer ID
begin_date	/dmo/begin_date	dat(8)	Flight Reference Scenario: Start Date
end_date	/dmo/end_date	dat(8)	Flight Reference Scenario: End Date
booking_fee	/dmo/booking_fee	curr(16,2)	Flight Reference Scenario: Booking Fee
total_price	/dmo/total_price	curr(16,2)	Flight Reference Scenario: Total Price
currency_code	/dmo/currency_code	cuky(5)	Flight Reference Scenario: Currency Code
description	/dmo/description	sstring(1024)	Flight Reference Scenario: Description
overall_status	/dmo/overall_status	char(1)	Flight Reference Scenario: Travel Status
created_by	syuname	char(12)	User Name
created_at	timestamp	dec(21,7)	UTC Time Stamp in Long Form (YYYYMMDDhhmmssmmuuun)
last_changed_by	syuname	char(12)	User Name
last_changed_at	timestamp	dec(21,7)	UTC Time Stamp in Long Form (YYYYMMDDhhmmssmmuuun)

Figure 132 : Assistance - Informations élément ABAP¹³⁰

Possibilité de transférer l'information d'un popup vers une fenêtre à part via le symbole « @ » ;

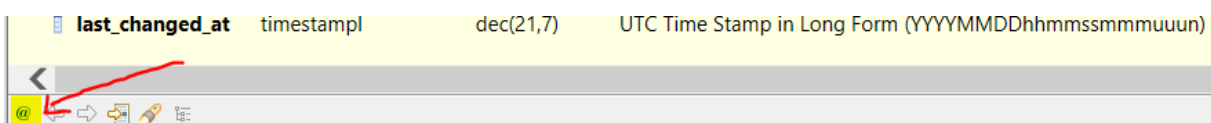


Figure 133 : Assistance - Informations élément ABAP nouvelle fenêtre¹³¹

Un assistant nous propose des solutions en cas d'erreurs via ctrl+1 sur l'erreur ;

¹³⁰ Source : Donnée de l'auteur sur Eclipse

¹³¹ Source : Donnée de l'auteur sur Eclipse

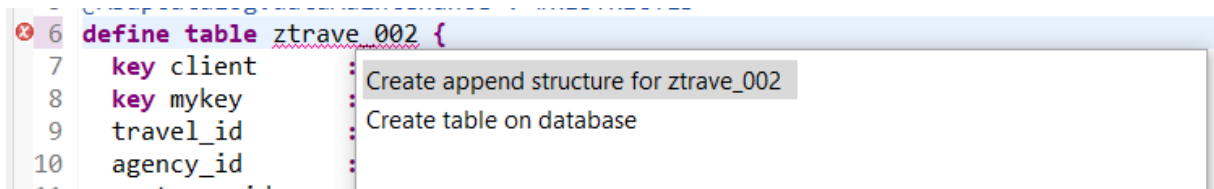


Figure 134 : Assistant erreurs¹³²

Similaire au « Pretty Printer » sur le SAP GUI, le formatage du code peut être configuré sur Eclipse : *Window > Preferences > ABAP Development > Editors > Source Code Editors* ;

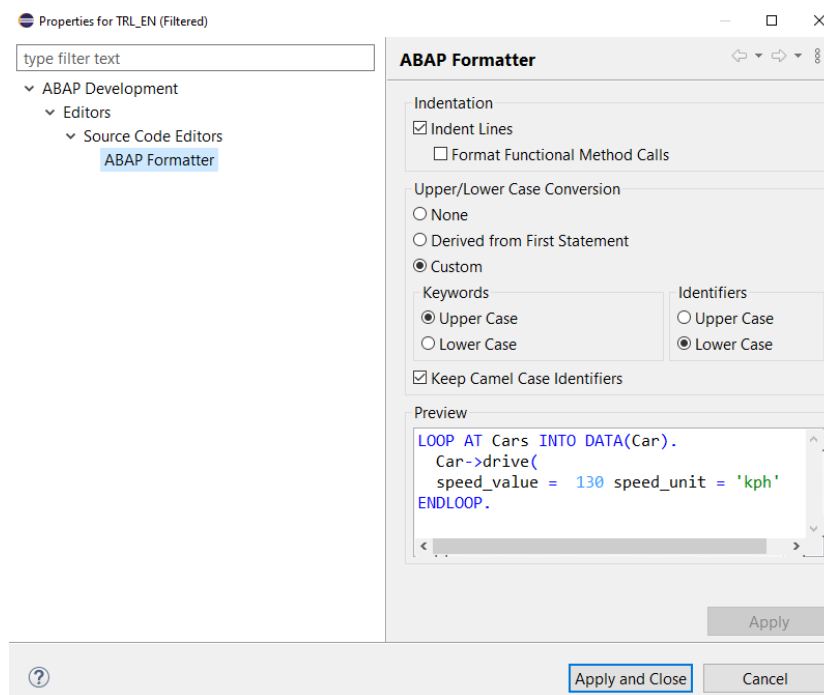


Figure 135 : Configuration format code ABAP¹³³

¹³² Source : Donnée de l'auteur sur Eclipse

¹³³ Source : Donnée de l'auteur sur Eclipse

Annexe IX : Comparaison directe ABAP SAP GUI vs ABAP cloud

Voici un exemple simple du développement ABAP en local vs cloud. A l'aide de l'appel d'un « function module », deux nombres entiers sont multipliés et le résultat est affiché ;

Développement cloud	Développement local
<pre> 1 Z_CLASS_VARIABLE 2 CLASS z_class_variable DEFINITION 3 PUBLIC 4 FINAL 5 CREATE PUBLIC . 6 7 PUBLIC SECTION. 8 INTERFACES: 9 if_oo_adt_classrun. 10 PROTECTED SECTION. 11 PRIVATE SECTION. 12 ENDCLASS. 13 14 15 CLASS z_class_variable IMPLEMENTATION. 16 METHOD if_oo_adt_classrun~main. 17 DATA : number1 TYPE int1, 18 number2 TYPE int1, 19 result TYPE int1 . 20 21 number1 = 10. 22 number2 = 5. 23 24 call function 'ZAS_VARIABLE_TB' 25 EXPORTING 26 iv_number01 = number1 27 iv_number02 = number2 28 IMPORTING 29 ev_result = result 30 . 31 ENDMETHOD. 32 33 ENDCLASS. </pre>	<pre> REPORT Z_CLASS_VARIABLE. DATA : number1 TYPE int1, number2 TYPE int1, result TYPE int1 . number1 = 10. number2 = 5. START-OF-SELECTION . CALL FUNCTION 'ZAS_VARIABLE_TB' EXPORTING iv_number01 = number1 iv_number02 = number2 IMPORTING EV_RESULT = result . WRITE : / result . </pre>
<p>ABAP Console</p> <hr/> <p>Field</p> <p>50</p>	<p>Program Z_CLASS_VARIABLE</p> <hr/> <p>Program Z_CLASS_VARIABLE</p> <hr/> <p>50</p>

Figure 136 : Comparaison ABAP cloud vs ABAP local¹³⁴

¹³⁴ Source : Données de l'auteur sur Eclipse et SAP GUI

Il n'est plus possible de développer des programmes comme sur SAP GUI en utilisant des reports par exemple. La raison est simple : les reports ont un dynpro ui (composants du « Screen programming ») et celui-ci n'est plus supporté par l'ABAP environnement. Il faut donc passer par des classes pour développer des applications console par exemple.

Annexe X : SAP Business Application Studio

Comme mentionné dans le chapitre 9.10 « Outils de développement », l'SAP Business Application Studio offre quelques outils intéressants. Bien que l'IDE n'est pas encore à la hauteur des outils ADT, elle montre des aspects pratiques et utiles pour les développeurs. L'SAP CDS Graphic Modeler par exemple, permet de créer des modèles CDS manuellement. Il est possible de représenter visuellement les modèles et de créer des entités à partir de ceux-ci ;

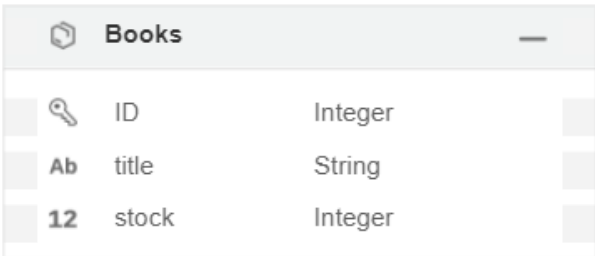
Représentation code	Représentation graphique
<pre>namespace my.namespace; entity Books { key ID : Integer; title : String; stock : Integer; }</pre>	

Figure 137 : SAP Business Application Studio – Code et visuel¹³⁵

Annexe XI : Gestion du travail

La méthodologie agile étudiée et pratiquée durant les études a servi de base pour la réalisation du travail de bachelor. Puisque cette méthodologie cible des groupes de projets, elle a été adaptée pour ce travail.

En début de travail, la création d'un « product backlog » était exigée par l'école. Celui-ci servait de fil rouge pour le travail et posait un certain cadre, définissant les tâches à accomplir et ses délais. Par rapport à un « product backlog » classique, les aspects prioritaires,

¹³⁵ Source : Donnée de l'auteur sur SAP Business Application Studio

estimations de temps et sprints n'ont pas été définis. Le « product backlog » a été divisé en cinq blocs ;

- Premier bloc : préparations administratives, travail de recherche sur le sujet (environ deux semaines)
- Deuxième bloc : analyse et rédaction (environ trois semaines)
- Troisième bloc : pratique, création du prototype (environ deux semaines)
- Quatrième bloc : rédaction (environ deux semaines)
- Cinquième bloc : corrections, relecture, finalisation (environ deux semaines)

Pour chaque bloc, un délai a été défini, évitant ainsi de prendre du retard et garantissant un travail régulier.

Pour le prototype, un « product backlog » a également été mis en place, exclusivement pour la réalisation de la partie pratique. Ce dernier a été divisé en deux parties, un bloc par prototype réalisé ;

- Premier bloc : préparation de l'environnement de travail et réalisation du premier prototype, développement sur le cloud (environ une semaine)
- Deuxième bloc : réalisation du deuxième prototype, migration d'un système local vers le cloud (environ une semaine)

Chaque semaine, un rapport a été établi, énumérant les tâches effectuées et leur durée ainsi que les problèmes rencontrés et leur solution. La durée du travail effectuée a également été inscrite dans un fichier Excel.

Environ toutes les deux semaines, une séance avec le professeur et/ou l'assistant était agendée pour faire une mise au point et discuter d'éventuels problèmes organisationnels ou rédactionnels. Ce contact régulier a permis d'avoir un suivi constant du travail.

Afin de mettre le cadre pour le travail et de définir les attentes, une séance de démarrage a également eu lieu avec le mandant, permettant. J'ai également eu l'occasion de me rendre sur place à Sion pour me faire une image de la situation actuelle et de mieux comprendre les méthodologies de travail mises en place.

Tout au long du travail, Monsieur Frédéric Morand et Madame Cassandra Payen étaient à disposition pour répondre à des questions techniques ou autres.

Annexe XII : Product Backlog

Délai	Thème	Tâches
jusqu'au 4.06.2021	Préparation	Préparer les divers fichiers administratifs ainsi que le fichier principal
	Recherche	Recherche et information sur le sujet
	Rédaction	Explication des technologies
	Rédaction	Explication de la SAP Cloud Platform (SCP)
	Rédaction	Aperçu et explications du ABAP sur SCP
jusqu'au 25.06.2021	Rédaction	Différence entre ABAP directement dans le cœur du système vs ABAP sur SCP
	Rédaction	Développement SAP sur SAP Cloud Platform
	Rédaction	Public cible & compétences à avoir
	Rédaction	Aperçu de l'architecture SAP
	Rédaction	Différences de concept de développement ABAP
	Rédaction	Déploiement ABAP (Git + comparaison avec le SAP Transport)
	Rédaction	Expliquer les outils de développement
	Rédaction	Avantages et inconvénients du SCP
	Rédaction	Suite aux résultats, prise de position relative à la meilleure solution pour le dev. ABAP
jusqu'au 09.07.2021	Préparation	Préparer l'environnement de travail pour la prog
	Programmation	Création d'un prototype
	Programmation	Montrer les Best Practices recommandées en termes de dev. ABAP
	Programmation	Tester le prototype
jusqu'au 23.07.2021	Rédaction	Rédiger l'introduction, les remerciements
jusqu'au 04.08.2021	Finalisation	Mise en forme/page
	Finalisation	Vérification, relecture

Annexe XIII : Product Backlog Prototype

	28.06.2021	-	
Sprint 0	02.07.2021		Préparation environnement de développement
			Création projet ABAP cloud
			Liaison à l'environnement ABAP cloud
			Déploiement sur ABAP cloud
			Tester le programme sur le cloud
	02.07.2021	-	
Sprint 1	07.07.2021		Création projet ABAP sur SE80 (local)
			Mise en place d'ABAP git
			Mise en place du dépôt Git
			Migration du projet local vers le cloud
			Teste le programme sur le cloud

Annexe XIV : Détails des heures

Ci-dessous le détail des heures consacrées à la réalisation de ce travail de bachelor.

HES-SO Valais	Report Summary													
Title	Analyse et Développement ABAP simple sur SCP @ Etat du Valais													
Student	Yann JOLLIEN													
Period	10.05.2021 - 04.08.2021													
	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Week 11	Week 12	Week 13	Total
Analysis & Planification	5,00	2,00	3,00	1,25	2,00	0,00	0,50	3,75	0,00	0,00	1,25	0,00	4,00	22,75
Installation	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	4,00	0,00	0,00	0,00	0,00	4,00
Configuration	0,00	0,00	0,00	0,00	2,00	2,00	0,00	4,00	4,00	0,00	5,00	2,00	0,00	19,00
Programmation	0,00	0,00	0,00	0,00	0,00	0,00	0,00	20,00	15,00	0,00	6,00	2,00	0,00	43,00
Research / Lecture	15,00	14,00	9,00	15,00	7,50	8,00	9,00	5,00	0,00	0,00	0,00	0,00	0,00	82,50
Redaction	0,00	10,00	11,00	12,00	15,50	10,00	15,00	11,00	12,00	24,00	13,00	36,00	0,00	169,50
Working Report Details	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	2,00	14,00
Course	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Total	21,00	27,00	24,00	29,25	28,00	21,00	25,50	44,75	36,00	25,00	26,25	41,00	6,00	354,75

Annexe XV : Récapitulatifs des heures



Figure 138 : Récapitulatifs des heures¹³⁶

¹³⁶ Source : Donnée de l'auteur

Déclaration sur l'honneur

Je déclare, par ce document, que j'ai effectué le travail de Bachelor ci-annexé seul, sans autre aide que celles dûment signalées dans les références, et que je n'ai utilisé que les sources expressément mentionnées. Je ne donnerai aucune copie de ce rapport à un tiers sans l'autorisation conjointe du RF et du professeur chargé du suivi du travail de Bachelor, à l'exception des personnes qui m'ont fourni les principales informations nécessaires à la rédaction de ce travail et que je cite ci-après : Madame Cassandra Payen – Monsieur Frédéric Morand.

Sion, le 03 août 2021

Yann Jollien