

Ecole Supérieure Privée de management de Tunis

Esprit School of Business



RAPPORT DE STAGE

En vue de l'obtention du diplôme de Licence en Business Computing
Parcours Business Intelligence

Soutenu le :
Réalisé par : **Yasmine Ben Larbi**

« Développement d'une solution d'aide à la décision »
01 Mars 2022 – 30 Juillet 2022

Maître de stage : MANSOUR Takwa

Encadrant académique : HADJ KACEM Aymen

	<i>Nom & Prénom</i>	<i>Date et Signature</i>
Maître de stage	<i>Takwa MANSOUR</i>	 28/07/2022
Encadrant académique	A. HajKacem	

Dédicaces

À mes très chers parents,

Tous les mots du monde ne sauraient exprimer l'immense amour que je vous porte, ni la profonde gratitude que je vous témoigne pour votre affection, votre soutien et les valeurs que vous m'inculquez toujours. Je vous remercie infiniment pour les efforts et les sacrifices que vous avez fait à mon égard. Je vous aime tellement et infiniment, que Dieu vous protège.

À mes amis,

Nul mot ne pourra exprimer ma gratitude et ma reconnaissance envers vous. Je vous remercie pour votre soutien et pour vos encouragements. Je vous remercie infiniment et je vous aime. Je vous souhaite un avenir parsemé de réussite, d'amour et de santé.

À ma tante,

Je te remercie pour ton soutien, tes mots si encourageants et tes prières jour et nuit pour que je réussisse mon projet. Je t'aime énormément.

À mes deux sœurs de cœur,

Merci de m'avoir soutenu tout au long de ce projet et merci pour vos encouragements et vos discours motivants quand tout semblait s'écrouler autour de moi. Je vous aime infiniment et j'espère que les liens de notre amitié seront si solides que rien et personne ne pourront les briser.

À une personne spéciale,

Je tiens à te dire merci d'être là dans les bons comme dans les mauvais moments. Dans la vie, on a tous besoin d'une personne sur qui on peut compter et cette personne, c'est toi. Je suis profondément reconnaissante d'avoir rencontré une personne qui me soutient dans tout ce que j'entreprends et qui me remonte le moral quand rien ne va. Je t'aime.

Remerciements

Au terme de ce projet au sein de **RFC**, je tiens à exprimer ma gratitude envers tous ceux qui ont assisté de près ou de loin à sa réalisation.

Je remercie ardemment mon encadrant à **Esprit School of Business**, **Monsieur Mohamed Aymen Haj Kacem** pour la qualité inédite de son suivi et la pertinence de ses conseils ainsi que pour ses suggestions pertinentes et surtout sa disponibilité qui ont permis à maintes reprises, de rediriger mon travail vers le bon chemin.

Je tiens à remercier chaleureusement mon maître de stage, **Madame Takwa Mansour**, architecte à **RFC**, de m'avoir accueillie dans son équipe, prise en charge, confiée des tâches, fait confiance, conseillée et encouragée.

J'adresse mes vifs remerciements à : **Madame Sonia Ben Aissa**, cheffe du département Business Application, et **Madame Faten Kribi**, cheffe du département Finance pour leurs conseils et leur implication dans mon projet.

Je remercie **toute l'équipe** de la Société **RFC** pour leur aide et leurs conseils qui m'ont été très précieux au cours de ces cinq mois de stage.

Un grand merci également à l'ensemble du corps professoral et administratif d'**Esprit School of Business**. Je voudrais, en particulier, remercier **Madame Amel Hajji** et **Monsieur Walid El Ayeb** pour leurs précieux conseils tout au long de mon cursus universitaire.

Table des matières

<i>Introduction générale</i>	16
<i>Chapitre I : Présentation générale du projet</i>	17
Introduction.....	18
1. Présentation de l'organisme d'accueil.....	18
1.1 Présentation de la société.....	18
1.2 Organisation de RFC.....	19
1.3 Services offerts.....	20
2. Cadre général du projet.....	21
2.1 Business Intelligence.....	21
2.1.1 OLAP et OLTP	21
2.1.2 Choix de l'approche de traitement.....	22
2.2 Le Cloud Computing.....	22
2.2.1 Cloud-Computing vs On-Premise.....	22
2.2.2 Les différents types de cloud Computing.....	23
2.2.3 Les services Cloud.....	23
2.3 Exemple de plateforme Cloud : Azure Synapse Analytics.....	24
2.3.1 Services offerts par Azure.....	24
2.3.2 Explication de la plateforme.....	25
2.4 Datamart/Data Warehouse.....	29
2.4.1 Datamart.....	29
2.4.2 Data Warehouse.....	29
2.5 Modélisation multidimensionnelle.....	30
2.6 Processus d'extraction, transformation et chargement de données.....	32
2.7 Le reporting.....	32
3. Cadre spécifique du projet.....	33
3.1 Description du projet.....	33
3.2 Problématique.....	34
3.3 Objectifs.....	35
3.4 Solution proposée.....	35
4. Méthode de gestion du projet.....	35

4.1 Méthode SCRUM.....	36
4.2 Méthode GIMSI.....	37
4.3 Méthodologie de travail adoptée.....	37
<i>Conclusion</i>	38
Chapitre II : L'environnement.....	39
<i>Introduction</i>	40
1. L'environnement de travail.....	40
1.1 L'environnement matériel.....	40
1.2 L'environnement logiciel.....	41
2. Préparation de l'environnement.....	42
2.1 L'abonnement Azure.....	42
2.2 Création de l'espace de travail et du groupe de ressources.....	42
2.3 Configuration du pool SQL dédié.....	44
2.4 Configuration du service lié : Logic Apps.....	44
2.5 Configuration du service lié : Power BI.....	46
<i>Conclusion</i>	48
Chapitre III : Conception de la solution.....	49
<i>Introduction</i>	50
1. Choix de l'approche de modélisation.....	50
1.1 Kimball.....	50
1.2 Inmon.....	50
1.3 Kimball vs Inmon.....	51
2. Modélisation conceptuelle des données.....	52
2.1 La table de faits.....	52
2.1.1 Table “Fact_analyse”.....	52
2.2 Les tables de dimension.....	54
2.2.1 Dimension date.....	54
2.2.2 Dimension client.....	55
2.2.3 Dimension société.....	55
2.2.4 Dimension facture.....	55
2.2.5 Dimension paiement.....	56
<i>Conclusion</i>	57
Chapitre IV : Réalisation du projet.....	58

<i>Introduction</i>	59
1. Construction de l'entrepôt de données.....	59
2. Extraction et automatisation de la génération du fichier source avec Logic Apps.....	62
3. Transformations.....	70
3.1 Création du jeu de données.....	70
3.2 Ajout des identifiants.....	74
3.2.1 Id_Client.....	74
3.2.2 Id_Société.....	77
3.2.3 Id_Paiement.....	80
3.2.4 Id_Date.....	81
3.3 Calcul des objectifs.....	86
3.3.1 Flux RFC.....	89
3.3.2 Flux Compusell.....	106
4. Chargement.....	122
5. Ajout des mesures.....	133
6. Visualisation.....	135
6.1 Connexion entre Azure Synapse et Power BI Desktop.....	135
6.2 Tableaux de bord.....	138
6.3 Publication des tableaux de bord sur Azure Synapse Analytics.....	144
<i>Conclusion</i>	146
<i>Conclusion générale</i>	147
<i>Webographie</i>	148

Liste des abréviations

BI : Business Intelligence

RFC : Réseaux, Formation et Conseil

MCPLS : Microsoft Certified Partner For Learning Solution

PVTC : Pearson Vue Authorized Test Center

IBM : International Business Machine corporation

OLAP : Online Analytical Processing

OLTP : Online Transaction Processing

SQL : Structured Query Language

JSON : JavaScript Object Notation

CSV : Comma Separated Values

DWH : DataWareHouse

ETL : Extraction, Transformation, Loading

KPI : Key Performance Indicator

DM : DataMart

URL : Uniform Ressource Locator

Liste des figures

Figure 1 : Logo de RFC.....	18
Figure 2 : Organigramme de la direction générale.....	19
Figure 3 : Organigramme du pôle “Business Application”.....	19
Figure 4 : Les partenaires de RFC.....	20
Figure 5 : Interface d'accueil Azure Synapse Analytics.....	25
Figure 6 : Hub “Données/Data” (Partie 1).....	26
Figure 7 : Hub “Données/Data” (Partie 2).....	27
Figure 8 : Hub “Développer/Develop”	27
Figure 9 : Hub “Intégrer/Integrate”	28
Figure 10 : Hub “Monitor/Surveillance”	28
Figure 11 : Hub “Gérer/Manage”	29
Figure 12 : DataMart/DataWarehouse.....	30
Figure 13 : Schéma en étoile.....	30
Figure 14 : Schéma en flocon.....	31
Figure 15 : Modèle en constellation.....	31
Figure 16 : Processus ETL.....	32
Figure 17 : Méthode SCRUM.....	36
Figure 18 : Méthode GIMSI.....	37
Figure 19 : L'abonnement Azure.....	42
Figure 20 : Création du groupe de ressources.....	43
Figure 21 : Création de l'espace de travail (Partie 1).....	43
Figure 22 : Création de l'espace de travail (Partie 2).....	44
Figure 23 : Configuration du pool SQL dédié.....	44
Figure 24 : Configuration du service lié : Logic Apps (Partie 1).....	45
Figure 25 : Configuration du service lié : Logic Apps (Partie 2).....	45
Figure 26 : Création du Workspace Power BI en ligne.....	46
Figure 27 : Configuration du service lié : Power BI (Partie 1).....	46
Figure 28 : Configuration du service lié : Power BI (Partie 2).....	47
Figure 29 : Configuration du service lié : Power BI (Partie 3).....	47
Figure 30 : Modélisation conceptuelle du Datawarehouse.....	52
Figure 31 : Création des scripts SQL.....	59

Figure 32 : Création de la table de dimension Dim_client.....	60
Figure 33 : Création de la table de dimension Dim_date.....	60
Figure 34 : Création de la table de dimension Dim_facture.....	60
Figure 35 : Création de la table de dimension Dim_paiement.....	61
Figure 36 : Création de la table de dimension Dim_société.....	61
Figure 37 : Création de la table de dimension Fact_analyse_recouvrement.....	62
Figure 38 : Fichier source dans Microsoft Teams.....	63
Figure 39 : Logic App Designer.....	63
Figure 40 : Logic App (Partie 1).....	64
Figure 41 : Logic App (Partie 2).....	64
Figure 42 : Logic App (Partie 3).....	65
Figure 43 : Logic App (Partie 4).....	65
Figure 44 : Logic App (Partie 5).....	66
Figure 45 : Logic App (Partie 6).....	67
Figure 46 : Logic App (Partie 7).....	67
Figure 47 : Ajout d'un pipeline (Partie 1).....	68
Figure 48 : Ajout d'un pipeline (Partie 2).....	69
Figure 49 : Logic App (Partie 8).....	69
Figure 50 : Ajout d'un pipeline (Partie 3).....	70
Figure 51 : Ajout d'un dataflow.....	71
Figure 52 : Ajout de la source.....	71
Figure 53 : Création du jeu de données (Partie 1).....	72
Figure 54 : Création du jeu de données (Partie 2).....	72
Figure 55 : Création du jeu de données (Partie 3).....	73
Figure 56 : Création du jeu de données (Partie 4).....	73
Figure 57 : Création du jeu de données Dataset 1.....	74
Figure 58 : Select client (Partie 1).....	74
Figure 59 : Select client (Partie 2).....	75
Figure 60 : Agrégation client.....	75
Figure 61 : Agrégation client (Grouper par).....	76
Figure 62 : Agrégation client (Agrégats).....	76
Figure 63 : Ajout de l'id client.....	77
Figure 64 : Select société (Partie 1).....	77

Figure 65 : Select société (Partie 2).....	78
Figure 66 : Agrégation société (Grouper par).....	78
Figure 67 : Agrégation société (Agrégats).....	79
Figure 68 : Ajout de l'id société.....	79
Figure 69 : Ajout de l'id paiement (Partie 1).....	80
Figure 70 : Ajout de l'id paiement (Partie 2).....	80
Figure 71 : Select Date Objectif.....	81
Figure 72 : Agrégation Date Objectif (Grouper par).....	82
Figure 73 : Agrégation Date Objectif (Agrégats).....	82
Figure 74 : Ajout de l'id Date Objectif.....	83
Figure 75 : Select date paiement.....	84
Figure 76 : Agrégation date paiement (Grouper par).....	84
Figure 77 : Agrégation date paiement (Agrégats).....	85
Figure 78 : Ajout de l'id date paiement.....	86
Figure 79 : Unions des id des deux dates.....	86
Figure 80 : Ajout des colonnes.....	88
Figure 81 : Fractionnement conditionnel : Société.....	89
Figure 82 : Fractionnement conditionnel des objectifs de RFC.....	90
Figure 83 : L'agrégation des objectifs non encaissés de RFC (Grouper par).....	91
Figure 84 : L'agrégation des objectifs non encaissés de RFC (Agrégats).....	92
Figure 85 : Jointure Id société.....	93
Figure 86 : Jointure numéro facture.....	93
Figure 87 : Jointure Id client.....	94
Figure 88 : Jointure Id Date Objectif.....	95
Figure 89 : L'agrégation des objectifs encaissés de RFC (Grouper par).....	96
Figure 90 : L'agrégation des objectifs encaissés de RFC (Agrégats).....	97
Figure 91 : Jointure Id société.....	98
Figure 92 : Jointure numéro facture.....	99
Figure 93 : Jointure Id client.....	99
Figure 94 : Jointure Id Date paiement.....	100
Figure 95 : L'agrégation des non objectifs encaissés de RFC (Grouper par).....	101
Figure 96 : L'agrégation des non objectifs encaissés de RFC (Agrégats).....	102
Figure 97 : Jointure Id société.....	103

Figure 98 : Jointure numéro facture.....	104
Figure 99 : Jointure Id client.....	105
Figure 100 : Jointure Id Date paiement.....	105
Figure 101 : Fractionnement conditionnel des objectifs de Compusell.....	106
Figure 102 : L'agrégation des objectifs non encaissés de Compusell (Grouper par).....	107
Figure 103 : L'agrégation des objectifs non encaissés de Compusell (Agrégats).....	108
Figure 104 : Jointure Id société.....	109
Figure 105 : Jointure numéro facture.....	110
Figure 106 : Jointure Id client.....	110
Figure 107 : Jointure Id Date Objectif.....	111
Figure 108 : L'agrégation des objectifs encaissés de Compusell (Grouper par).....	112
Figure 109 : L'agrégation des objectifs encaissés de Compusell (Agrégats).....	113
Figure 110 : Jointure Id société.....	114
Figure 111 : Jointure numéro facture.....	114
Figure 112 : Jointure Id client.....	115
Figure 113 : Jointure Id Date paiement.....	116
Figure 114 : L'agrégation des non objectifs encaissés de Compusell (Grouper par).....	117
Figure 115 : L'agrégation des non objectifs encaissés de Compusell (Agrégats).....	118
Figure 116 : Jointure Id société.....	119
Figure 117: Jointure numéro facture.....	120
Figure 118: Jointure Id client.....	121
Figure 119: Jointure Id Date paiement.....	122
Figure 120 : Jeu de données Dim_client.....	122
Figure 121 : Jeu de données Dim_date.....	123
Figure 122 : Jeu de données Dim_facture.....	123
Figure 123 : Jeu de données Dim_paiement.....	124
Figure 124 : Jeu de données Dim_societe.....	124
Figure 125 : Jeu de données Fact_analyse.....	125
Figure 126 : Récepteur Dim_client.....	126
Figure 127 : Récepteur Dim_societe.....	127
Figure 128 : Récepteur Dim_facture.....	128
Figure 129 : Récepteur Dim_paiement.....	129
Figure 130 : Récepteur Fact_date.....	130

Figure 131 : Récepteur Fact_analyse (Partie 1).....	132
Figure 132 : Récepteur Fact_analyse (Partie 2).....	132
Figure 133 : Ajout des mesures (Partie 1).....	133
Figure 134 : Ajout des mesures (Partie 2).....	133
Figure 135 : Ajout des mesures (Partie 3).....	134
Figure 136 : Connexion entre Azure Synapse et Power BI Desktop (Partie 1).....	135
Figure 137 : Connexion entre Azure Synapse et Power BI Desktop (Partie 2).....	135
Figure 138 : Connexion entre Azure Synapse et Power BI Desktop (Partie 3).....	136
Figure 139 : Connexion entre Azure Synapse et Power BI Desktop (Partie 4).....	136
Figure 140 : Connexion entre Azure Synapse et Power BI Desktop (Partie 5).....	137
Figure 141 : Connexion entre Azure Synapse et Power BI Desktop (Partie 6).....	137
Figure 142 : Premier tableau de bord (Partie 1).....	138
Figure 143 : Premier tableau de bord (Partie 2).....	140
Figure 144 : Deuxième tableau de bord.....	140
Figure 145 : Troisième tableau de bord.....	141
Figure 146 : Quatrième tableau de bord.....	142
Figure 147 : Cinquième tableau de bord.....	143
Figure 148 : Enregistrement des tableaux de bord.....	143
Figure 149 : Publication des tableaux de bord sur Azure Synapse (Partie 1).....	144
Figure 150 : Publication des tableaux de bord sur Azure Synapse (Partie 2).....	145
Figure 151 : Publication des tableaux de bord sur Azure Synapse (Partie 3).....	145
Figure 152 : Publication des tableaux de bord sur Azure Synapse (Partie 4).....	146

Liste des tableaux

Tableau 1 : OLAP vs OLTP	21
Tableau 2 : Cloud Computing vs On Premise Computing.....	22-23
Tableau 3 : L'environnement matériel.....	40
Tableau 4 : L'environnement logiciel.....	41-42
Tableau 5 : Kimball vs Inmon.....	51
Tableau 6 : Dictionnaire de données “Fact_analyse” (Partie 1).....	53
Tableau 7 : Dictionnaire de données “Fact_analyse” (Partie 2).....	54
Tableau 8 : Dictionnaire de données “Dim_date”	54
Tableau 9 : Dictionnaire de données “Dim_client”	55
Tableau 10 : Dictionnaire de données “Dim_société”.....	55
Tableau 11 : Dictionnaire de données “Dim_facture”	55-56
Tableau 12 : Dictionnaire de données “Dim_paiement”	56-57
Tableau 13 : Ajout des colonnes.....	87
Tableau 14 : Fractionnement conditionnel : Société.....	88
Tableau 15 : Fractionnement des objectifs de RFC.....	89
Tableau 16 : L'agrégation des objectifs non encaissés de RFC (Grouper par).....	90
Tableau 17 : L'agrégation des objectifs non encaissés de RFC (Agrégats).....	91
Tableau 18 : Condition de jointure 1.....	92
Tableau 19 : Condition de jointure 2.....	93
Tableau 20 : Condition de jointure 3.....	94
Tableau 21 : Condition de jointure 4.....	94
Tableau 22 : L'agrégation des objectifs encaissés de RFC (Grouper par).....	95
Tableau 23 : L'agrégation des objectifs encaissés de RFC (Agrégats).....	96
Tableau 24 : Condition de jointure 1.....	97
Tableau 25 : Condition de jointure 2.....	98
Tableau 26 : Condition de jointure 3.....	99
Tableau 27 : Condition de jointure 4.....	100
Tableau 28 : L'agrégation des non objectifs encaissés de RFC (Grouper par).....	100-101
Tableau 29 : L'agrégation des non objectifs encaissés de RFC (Agrégats).....	101

Tableau 30 : Condition de jointure 1.....	102
Tableau 31 : Condition de jointure 2.....	103
Tableau 32 : Condition de jointure 3.....	104
Tableau 33 : Condition de jointure 4.....	105
Tableau 34 : Fractionnement des objectifs de Compusell.....	106
Tableau 35 : L'agrégation des objectifs non encaissés de Compusell (Grouper par).....	107
Tableau 36 : L'agrégation des objectifs non encaissés de Compusell (Agrégats).....	107
Tableau 37 : Condition de jointure 1.....	108
Tableau 38 : Condition de jointure 2.....	109
Tableau 39 : Condition de jointure 3.....	110
Tableau 40 : Condition de jointure 4.....	111
Tableau 41 : L'agrégation des objectifs encaissés de Compusell (Grouper par).....	111
Tableau 42 : L'agrégation des objectifs encaissés de Compusell (Agrégats).....	112
Tableau 43 : Condition de jointure 1.....	113
Tableau 44 : Condition de jointure 2.....	114
Tableau 45 : Condition de jointure 3.....	115
Tableau 46 : Condition de jointure 4.....	115
Tableau 47 : L'agrégation des non objectifs encaissés de Compusell (Grouper par).....	116
Tableau 48 : L'agrégation des non objectifs encaissés de Compusell (Agrégats).....	117
Tableau 49 : Condition de jointure 1.....	118
Tableau 50 : Condition de jointure 2.....	119
Tableau 51 : Condition de jointure 3.....	120
Tableau 52 : Condition de jointure 4.....	121
Tableau 53 : Mappage Dim_client.....	125
Tableau 54 : Mappage Dim_société.....	126
Tableau 55 : Mappage Dim_facture.....	127
Tableau 56 : Mappage Dim_paiement.....	128-129
Tableau 57 : Mappage Dim_date.....	129
Tableau 58 : Mappage Fact_analyse.....	130-131-132

Introduction générale

L'évolution de la business intelligence s'explique par plusieurs facteurs principalement liés à l'environnement actuel, qui est devenu de plus en plus nuancé. Par conséquent, les besoins en informatique décisionnelle sont maintenant considérables. À cette fin, la BI occupe une place primordiale dans le domaine de l'analyse de données. Il met à la disposition des décideurs et des managers des traitements de données qui seront visualisés sous forme de tableaux de bord pour les aider à prendre de bonnes décisions commerciales.

Dans le monde du travail, les entreprises peuvent se retrouver dans des difficultés financières, menant parfois à la misère à cause de complications quant au règlement des créances clients. En effet, si une entreprise vend du matériel et des services sans recevoir en contrepartie de l'argent, un écart entre les entrées et les sorties des flux de trésorerie va commencer à croître, ce qui conflictuel. De ce fait, l'objectif de notre projet consiste à développer une solution décisionnelle afin de définir les indicateurs de performance clé « KPI » pour le suivi de la rentabilité des clients, suivre le recouvrement des créances ainsi que les objectifs du mois.

Nous avons effectué un stage au sein de RFC, une société tunisienne spécialisée dans le domaine du conseil et de l'intégration des solutions d'infrastructure informatique. Nos principales missions consistent à valoriser l'entrepôt de données en mettant en place une solution décisionnelle visant l'optimisation de la gestion financière ainsi qu'une élaboration de tableaux de bord contenant tous les indicateurs de performances répondant aux critères établis par l'entreprise.

Ce rapport s'étale sur 4 chapitres. Au niveau du chapitre 1, nous présenterons l'organisme d'accueil, le cadre général et le cadre spécifique du projet ainsi que la méthodologie de gestion du projet adoptée. Dans le chapitre 2, nous présenterons L'environnement logiciel, l'environnement matériel ainsi que la configuration de l'environnement. La méthodologie de conception choisie ainsi que la modélisation conceptuelle des données seront présentées au niveau du chapitre 3. Les étapes de la réalisation du projet ainsi que les visualisations seront détaillées dans le chapitre 4.

Chapitre I : Présentation générale du projet

Introduction

Nous avons effectué un stage au sein de RFC, une société tunisienne spécialisée dans le domaine du conseil et de l'intégration des solutions d'infrastructure informatique.

La mission consiste à développer une solution permettant la prise de décision concernant le recouvrement des créances et permettant de mieux visualiser les objectifs mensuels de l'entreprise. Au niveau de ce chapitre, nous présenterons l'organisme d'accueil, le cadre général et le cadre spécifique du projet ainsi que la méthodologie de gestion du projet adoptée. Ce premier chapitre représente le point de départ de notre projet.

1. Présentation de l'organisme d'accueil

1.1 Présentation de la société

"RFC, aussi appelée Réseaux, Formation et Conseil, a été fondée en 2003 et s'est spécialisée dans les projets à forte valeur ajoutée dans le domaine du conseil et de l'intégration des solutions d'infrastructure informatique. Forte de plus de 15 ans d'expérience, RFC dispose aujourd'hui de compétences et de savoir-faire uniques sur le marché tunisien, et ce, en matière d'infrastructure, sécurité et conseil stratégique pour l'optimisation des infrastructures informatiques.

RFC a développé des activités d'assistance et de services dans les domaines des réseaux, de la formation et du conseil, en nous appuyant sur des collaborateurs certifiés à l'échelle nationale et internationale." [1]



Figure 1 : Logo de RFC

1.2 Organigramme de RFC

L'organigramme de RFC se présente comme suit. Nous avons tout d'abord l'organigramme de la direction générale qui se présente comme suit. La société est présidée par Riadh Fehmi. (Figure 2)

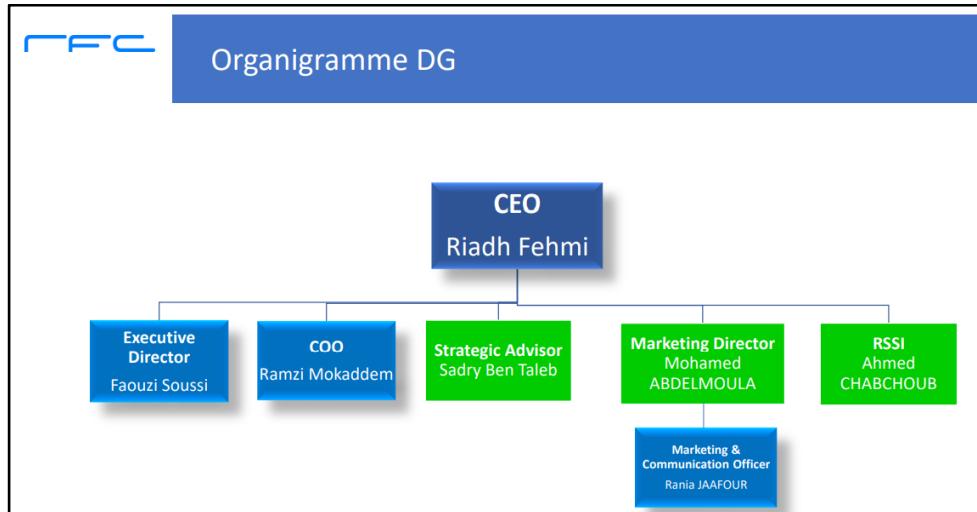


Figure 2 : Organigramme de la direction générale.

Nous avons notamment l'organigramme du pôle au sein duquel nous avons effectué notre stage appelé "Business Application" présidé par M.Sonia Ben Aissa et composé d'une équipe de consultants et architectes dont M. Takwa Mansour qui est mon encadrante professionnelle. Le stage a été effectué en étroite collaboration avec le pôle Finance. (Figure 3)



Figure 3 : Organigramme du pôle "Business Application"

1.3 Services offerts par RFC

RFC offre plusieurs services dont :

- L'infogérance équivaut à pouvoir offrir l'assistance et les aptitudes qu'il faut aux clients au moment où ils en ont besoin. Ceci est un élément concluant pour garantir la disponibilité et la stabilité de leur infrastructure. Pour les clients qui ne disposent pas d'informaticiens dédiés au sein de leurs structures, RFC, leur apporte ses services de maintenance, d'assistance et de conseil nécessaires.
- Support : RFC privilégie les relations avec ses clients par l'écoute, la disponibilité et une grande élasticité. Les clients de RFC jouissent d'un accompagnement de la part de l'équipe informatique du début jusqu'à la fin du projet.
- Conseil : La tâche de conseil de RFC se base avant tout sur une mission d'étude de projets informatiques, indispensable pour identifier et comprendre le besoin des clients, évaluer le projet, proposer des scénarios d'architecture et aider les clients à faire les meilleurs choix, réaliser une plateforme de test et des maquettes de démonstration, établir le budget moyen de la solution et assurer un accompagnement et créer une véritable relation de partenariats et de confiance.
- RFC, est aussi un centre de formation et certification : RFC, certifiée MCPLS (Microsoft Certified Partner For Learning Solutions), délivre des formations basées sur des supports de cours originaux de Microsoft. En tant que PVTC (Pearson Vue Authorized Testing Center), RFC offre un programme de certification permettant aux candidats de valider leurs expertises dans les technologies Microsoft, IBM, Cisco, Oracle, Java...



Figure 4 : Les partenaires de RFC

2. Cadre général du projet

Le présent projet vise à mettre en place une solution d'aide à la décision. Pour mieux comprendre les concepts inhérents à ce domaine, nous étions amenées à mener une étude bibliographique que nous présenterons dans ce qui suit.

2.1 Business Intelligence

“La Business Intelligence, parfois appelée tout simplement «le décisionnel», est un processus d'exploitation des données qui a pour but de faciliter la prise de décisions commerciales par les chefs d'entreprises et managers par le biais de tableaux de bord.”[2] “Dans le monde d'analyse de données, il existe deux modèles : OLAP (Online Analytical Processing) ou OLTP (Online Transaction Processing) qui sont des systèmes d'information spécialisés permettant de traiter les données de l'entreprise de deux manières différentes.”[3]

2.1.1 OLAP et OLTP

“Les systèmes décisionnels OLAP sont les sources d'analyse des données qui conduisent aux décisions, tandis que les systèmes relationnels OLTP sont souvent utilisés comme sources de données.” [4] Une comparaison entre OLAP et OLTP est présentée dans le tableau ci-dessous:

OLAP	OLTP
Système d'extraction et d'analyse de données en ligne	Système de transaction en ligne
Extraction des données multidimensionnelles à des fins d'analyse	Insertion, mise à jour et suppression
Données volumineuses	Données non volumineuses
Requêtes complexes	Requêtes simples

Tableau 1 : OLAP vs OLTP

2.1.2 Choix de l'approche de traitement

Nous avons choisi le traitement analytique OLAP afin d'analyser des données massives. Elle permet le traitement des requêtes complexes.

2.2 Le Cloud Computing

Le cloud computing permet l'usage des services informatiques (tels que le stockage de données, serveurs, les outils d'analyse, mise en réseau, etc..) par le biais d'internet fournis par des fournisseurs tels que Amazon, Microsoft, Google, etc...

2.2.1 Cloud Computing VS On-premise (sur place)

Le tableau ci-dessous présente une étude comparative entre le Cloud Computing et les serveurs sur place et donc ceux qui se trouvent physiquement sur le site de la société.

On-premise	Cloud
Coûteux	Pay as you go
Moins de scalabilité	Dynamique
Nécessite de l'espace	Pas d'espace nécessaire pour installer le serveur
Equipe nécessaire pour la maintenance	Maintenance occupée par le fournisseur
Faible sécurité	Forte sécurité
Pas de recouvrement de données	Données recouvrables
Pas de flexibilité	Accessible et flexible

Partage de données limité	Partage facile de données via internet
Implémentation qui prend du temps	Exécutable dès le début

Tableau 2 : Cloud Computing vs On Premise Computing

2.2.2 Les différents types Cloud

Le cloud dispose trois types différents :

- Le cloud public : “C'est une infrastructure informatique dédiée au grand public pour une utilisation ouverte, par le biais d'internet, disponible dans les locaux des fournisseurs de services Cloud.”[5]
- Le cloud privé : “C'est une infrastructure informatique dédiée à certains utilisateurs ou à une entité prédéfinie, par le biais d'internet, qui existe dans les locaux des fournisseurs de services Cloud. Il est dédié à répondre aux besoins des entreprises qui ont la possibilité d'héberger une infrastructure de cloud privé directement sur leurs sites ou dans le centre de données du fournisseur de services pour assurer la maintenance, l'accessibilité, la flexibilité et des niveaux de sécurité plus élevés.” [5]
- Le cloud hybride : “C'est une infrastructure informatique qui réunit les deux types de cloud précédents. Il donne la possibilité à l'utilisateur de choisir de partitionner une partie de l'opération entre l'environnement privé et public.” [5]

2.2.3. Les services Cloud

Le cloud dispose de 4 services :

- “IaaS (Interface as a Service) qui désigne les solutions matérielles fournies par le fournisseur comme des processeurs de haute performance, des espaces de stockage etc... L'utilisateur a recours, d'une manière virtuelle, à ces instances qui sont distribuées dans des pools de ressources.” [5]
- “PaaS (Platform as a Service) qui désigne les solutions matérielles avec un environnement complet fourni par le fournisseur. Ce service est dédié aux

développeurs qui veulent gérer et développer les applications commerciales qui seront face à une plateforme cloud déjà établie sans avoir à construire et à maintenir l'infrastructure que ces processus de développement de logiciels requièrent généralement.” [5]

- “SaaS (Software as a Service) qui désigne les logiciels fournis par le fournisseur sous forme de service hébergé dans un serveur cloud, accessible à distance avec un abonnement préétabli.” [5]
- “XaaS (Everything as a Service) qui désigne tous les services et les fonctionnalités qui pourraient être fourni par le fournisseur, citons à titre d'exemple Haas (Humain as a service) qui est un service désignant un groupe d'Hommes qui prend en charge le travail à distance via internet.” [5]

2.3 Exemple de plateforme Cloud : Azure Synapse Analytics

Azure Synapse Analytics est une PaaS (Platform as a Service) d'analytique qui regroupe l'insertion et la visualisation des données, le Data Warehousing, le traitement Big data ainsi que la génération de rapports d'analyse dans une seule et unique même plateforme. C'est un service d'analyse d'informations illimité qui sert à l'extraction, la transformation ainsi que le chargement des données afin de régir les besoins en Business Intelligence.

2.3.1 Les services offerts par azure synapse

Azure synapse offre plusieurs services :

- Synapse SQL : est un système de requête distribué. Il donne le droit de manipuler les scénarios de Data warehousing et de la virtualisation et de Machine Learning. Il présente deux modèles de ressources :
 - Pools SQL serverless : pour les charges de travail imprévues ou en rafale.
 - Pools SQL dédiés : Un pool SQL dédié fait référence aux fonctionnalités d'entreposage de données d'entreprise qui sont disponibles dans Azure Synapse Analytics. Il permet d'analyser un volume conséquent de données. C'est une base de données relationnelle dans laquelle vous allez pouvoir

stocker des tables, créer des index et créer des procédures stockées. C'est le pool que nous allons utiliser dans le cadre de notre projet.

- Apache Spark : C'est le moteur de Big Data le plus utilisé dans les domaines d'analyse de données. Il est généralement utilisé dans le traitement de données, data engineering et l'apprentissage automatique.
- Data Lake : permet de stocker les données de format JSON, CSV, TSV.
- Azure Data Factory : qui permet l'insertion et la création des pipelines enrichis à grande échelle ainsi que la création des dataflows (pour les transformations).
- Synapse studio : C'est l'interface d'accueil de Azure Synapse qui centralise tous les services.

2.3.2 Explication de la plateforme

La figure ci-dessous (figure 5) montre l'interface d'accueil d'Azure Synapse Analytics. Nous pouvons y apercevoir les différents services et les hubs qui se trouvent à gauche de l'écran.

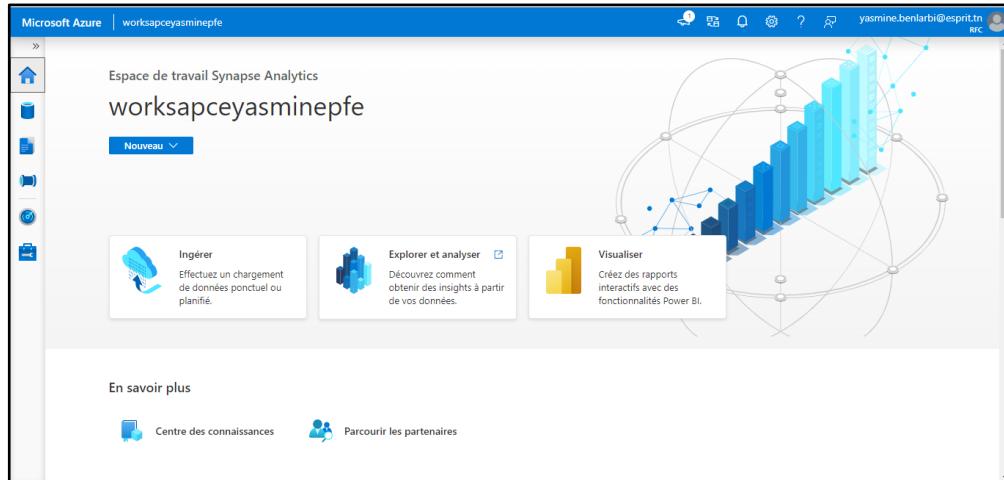


Figure 5 : Interface d'accueil Azure Synapse Analytics

La figure ci-dessous (Figure 6) montre le hub “Données/Data”. Dans “Espace de travail” nous pouvons apercevoir la base de données Lake qui correspond au pool SQL Serverless créé par défaut à la création du Workspace. En dessous, nous pouvons voir la base de données SQL qui correspond au pool SQL dédié où seront stockées les tables créées par le biais de scripts SQL.

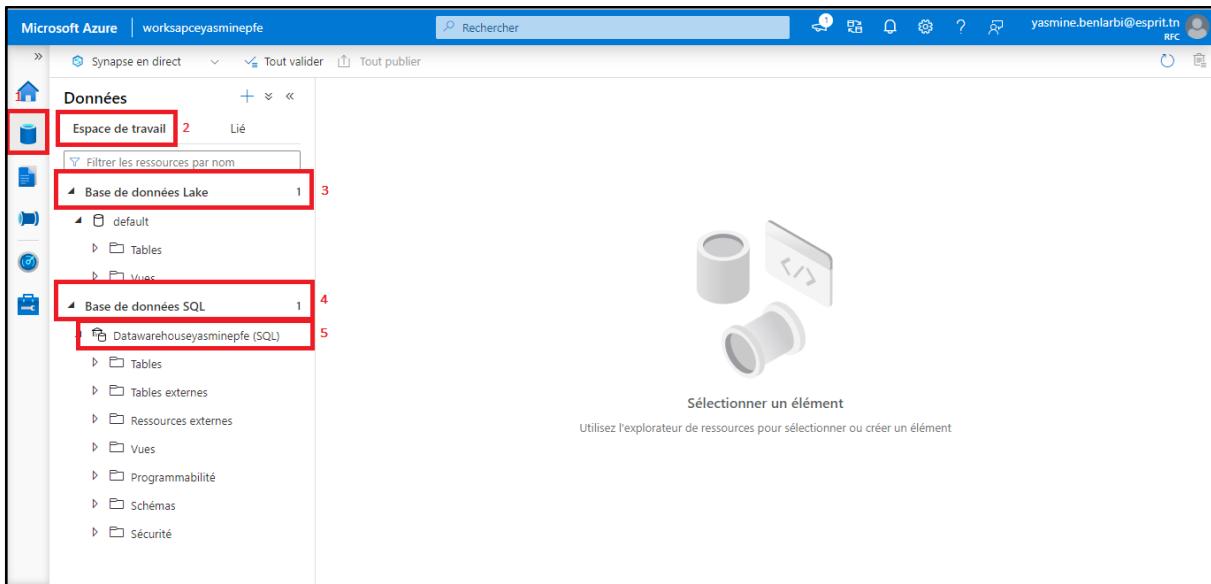
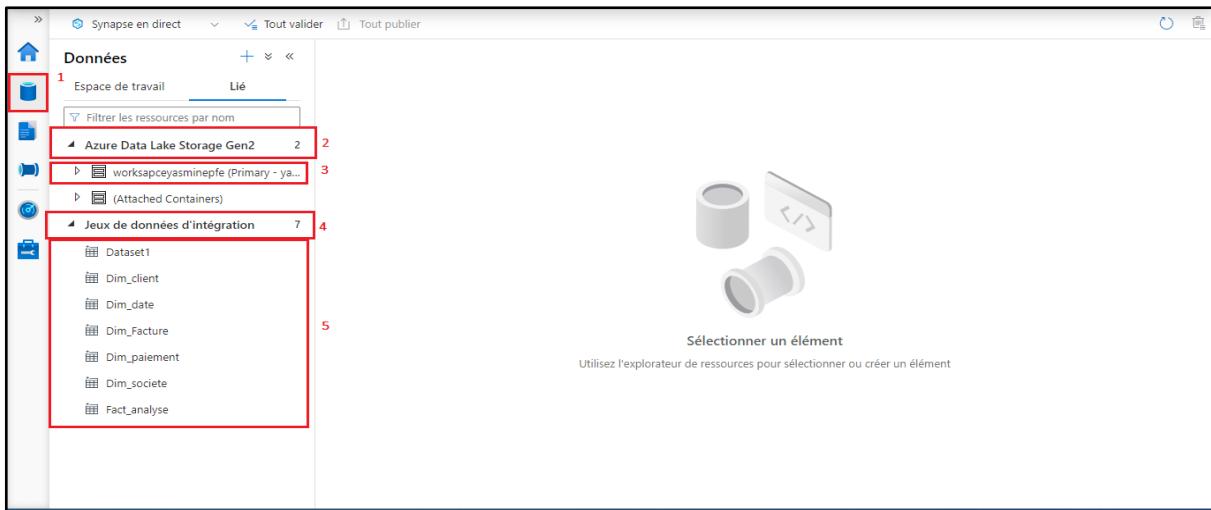
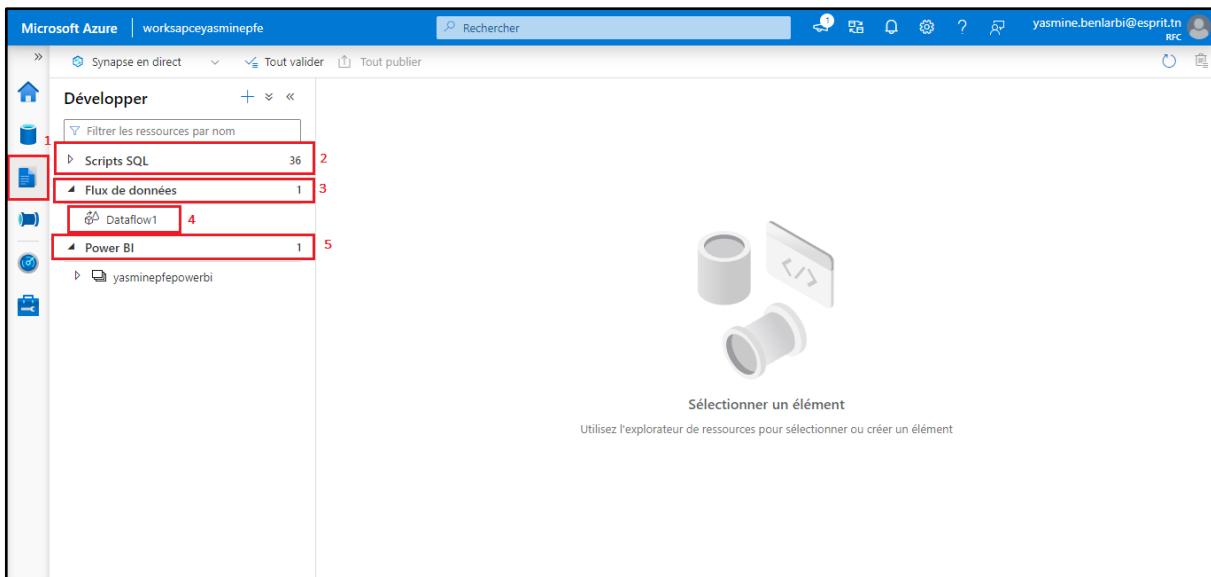


Figure 6 : Hub “Données/Data” (Partie 1)

La figure ci-dessous (Figure 7) montre le hub “Données/Data”. Dans “Lié” nous pouvons apercevoir Azure Data Lake Storage où sera stocké notre fichier source. En dessous, nous pouvons apercevoir les jeux de données d'intégration. En créant un jeu de données Azure, nous créons une référence à l'emplacement de la source de données, ainsi qu'une copie de ses métadonnées.

**Figure 7 : Hub “Données/Data” (Partie 2)**

La figure ci-dessous (Figure 8) représente le hub “Développer/Develop” où nous pouvons écrire les scripts SQL, créer des data flows qui correspondent aux transformations faites aux données du fichier source. Nous pouvons notamment afficher les tableaux de bord créés par le biais de Power BI.

**Figure 8 : Hub “Développer/Develop”**

La figure ci-dessous (Figure 9) montre le hub “Intégrer/Integrate” où nous pouvons créer des pipelines qui sont des regroupements logiques d'activités qui effectuent une tâche ensemble.

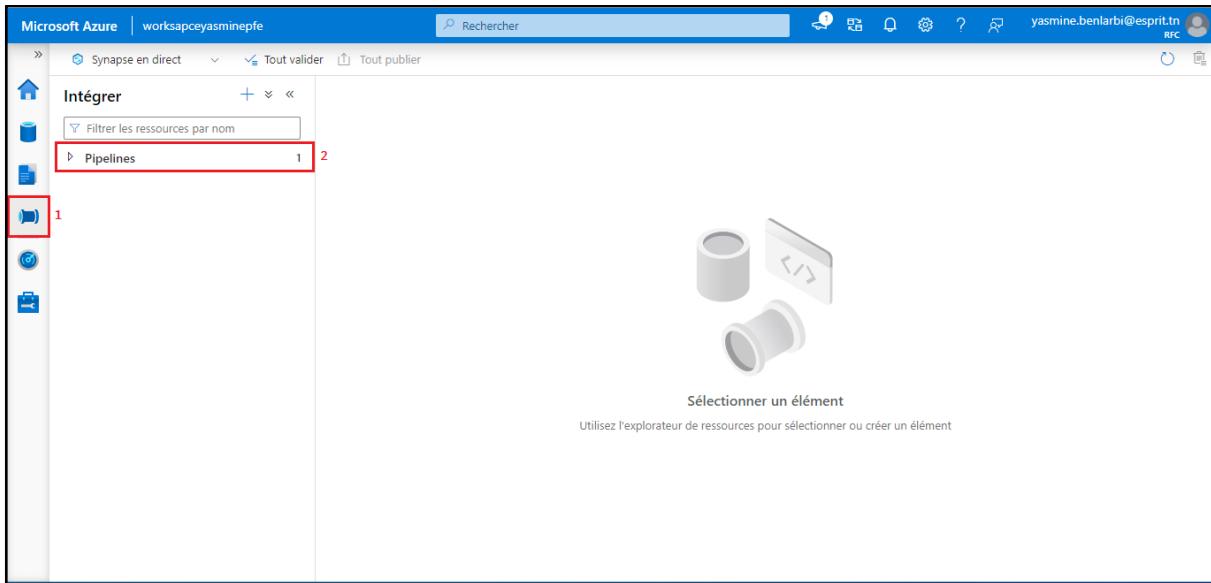


Figure 9 : Hub “Intégrer/Integrate”

Nous avons ici le hub “monitor/surveillance” qui permet de suivre toutes les activités de l'espace de travail Synapse, y compris celles qui sont actives. (Figure 10)

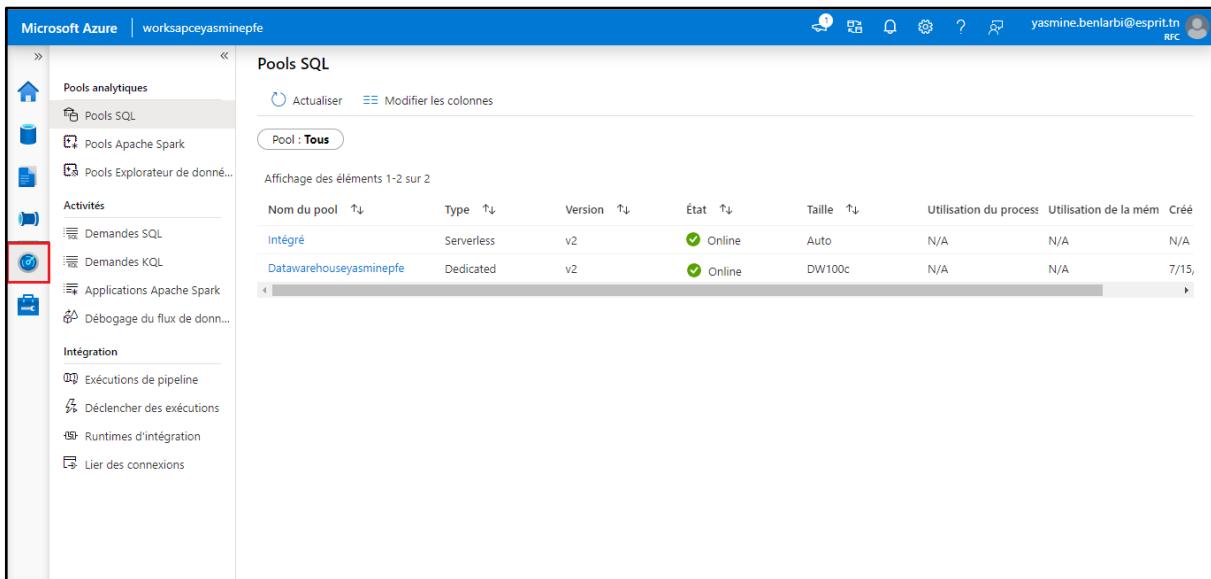


Figure 10 : Hub “Monitor/Surveillance”

Enfin nous avons le hub “Manage/Gérer” qui permet de gérer l'espace de travail dans un seul tableau de bord. Il propose de nombreuses options, de la création de services liés à l'aide de connecteurs tels que Power BI à l'octroi de l'accès à la plateforme. (Figure 11)

Nom	Type	État	Taille
Intégré	Serverless	En ligne	Auto
Datawarehouseyasminepfe	Dédié	NA	DW100c

Figure 11 : Hub “Gérer/Manage”

2.4 Datamart/Data Warehouse

2.4.1 Datamart

“Le Datamart est un groupe de données ciblées, organisées, regroupées et agrégées. C'est un sous-ensemble d'un DW, conçu pour répondre à un besoin métier spécifique. Il est donc consacré à être questionné sur un panel de données restreint à son domaine fonctionnel, selon des paramètres qui auront été précisés à l'avance lors de sa conception.” [6]

2.4.2 Data Warehouse

“Les Data Warehouse dits entrepôt de données sont des bases de données spécialisées utilisées pour stocker des données provenant de bases de données de production à l'aide des outils d'ETL.”[7]

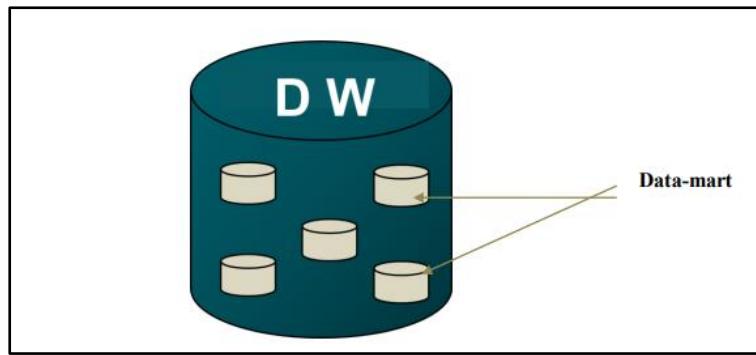


Figure 12 : Datamart/Data Warehouse

2.5 Modélisation multidimensionnelle

Les projets BI incluent souvent la mise en place d'un entrepôt de données. Sa modélisation est un travail important avec une marge d'erreur acceptée presque nulle. La modélisation multidimensionnelle dépend fortement des besoins du client et des objectifs du projet BI. La modélisation du datawarehouse ou du datamart peut suivre n'importe quel modèle de construction de données multidimensionnel tels que le modèle en étoile, flocon de neige ou constellation. Ces modèles impliquent des tables de faits et des tables de dimensions.

En effet, la table de faits contient les faits qui représentent la valeur des mesures (KPI) sur lesquelles va porter l'analyse, quant aux dimensions, elles présentent les axes avec lesquels on veut faire l'analyse.

- Schéma en étoile : “Un modèle en étoile se définit par l’omniprésence de branches autour de la table de fait. C’est une manière de lier les dimensions et les faits dans un entrepôt de données. Dans le schéma en étoile, la table de fait contient les clés primaires des tables de dimensions qui sont en relation avec elle.”[8]

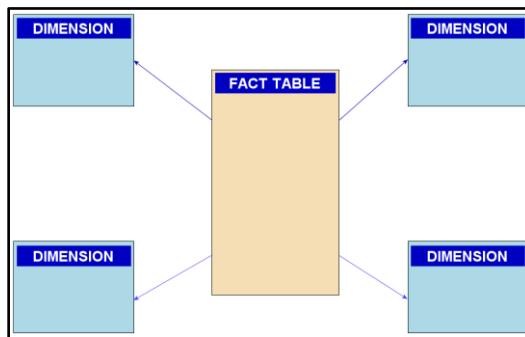


Figure 13 : Schéma en étoile

- Schéma en flocon : "Le principe de ce schéma qu'il peut exister des hiérarchies de dimensions et qu'elles soient reliées à la table de faits. Les dimensions peuvent être décomposées selon leur hiérarchie." [8]

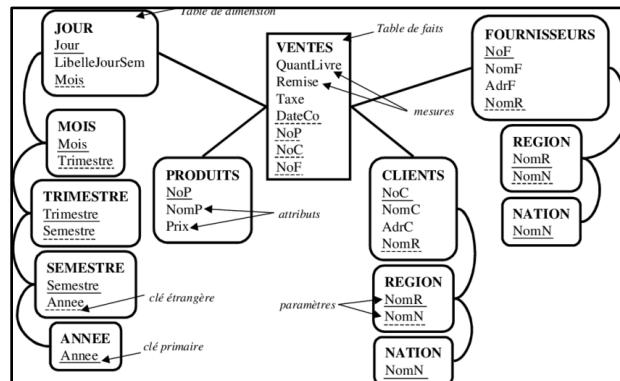


Figure 14 : Schéma en flocon

- Modèle en constellation : "La modélisation en constellation consiste à utiliser plusieurs modèles en étoiles et qui ont des tables de dimensions communes. Un modèle en constellation comprend donc plusieurs tables de faits et des tables de dimensions qui peuvent être communes à ces tables de faits." [8]

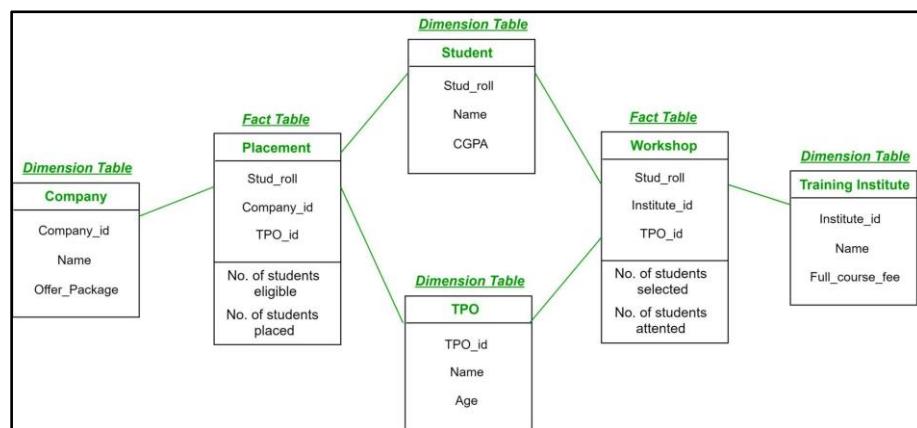


Figure 15 : Modèle en constellation

2.6 Processus d'Extraction, Transformation et Chargement

“Les processus ETL se chargent de récupérer des données depuis des sources opérationnelles existantes et de les charger vers le système décisionnel”[9] :

- Extraction des données des bases de données opérationnelles .
- Transformation de ces données.
- Loading ou chargement des données dans le système décisionnel (Datawarehouse, Datamart, ou Cube).

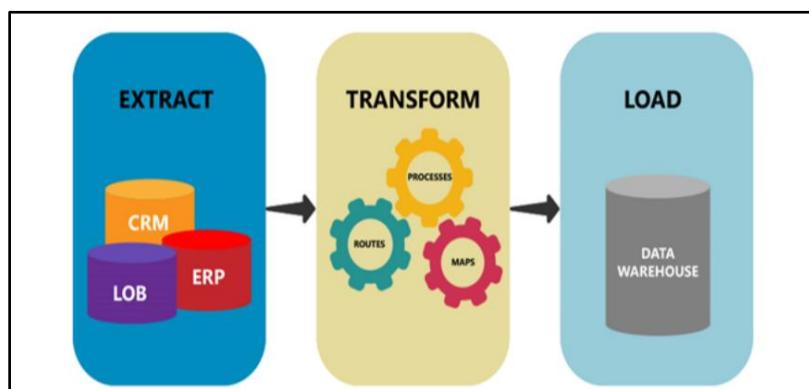


Figure 16 : Processus ETL

2.7 Reporting

Le reporting constitue la dernière étape dans une chaîne de projet BI. Il s'agit d'un processus permettant d'extraire des données depuis un DM/DWH et les représenter sous forme de rapports clairs, lisibles et personnalisés selon la préférence du client ou du lecteur.

Le reporting permet aux managers et aux décideurs de sélectionner, trier mais aussi regrouper et répartir des données relatives selon les critères qu'ils choisissent (période, secteur etc...). Cette technique permet notamment aux chefs d'entreprises et aux gestionnaires de présenter les indicateurs et les objectifs de l'organisme d'une manière synthétique ou détaillée, le plus souvent graphique selon les besoins ou les attentes.

Il existe plusieurs outils de reporting qui permettent de constituer des tableaux de bords et des rapports. Parmi ces outils, nous citons le plus connu : Power BI que nous utiliserons à la fin de ce projet.

3. Cadre spécifique du projet

3.1 Description du projet

RFC offre des services et vend des biens (matériel ou logiciel). Toute vente de bien ou de prestation de service effectuée par un professionnel implique d'émettre une facture. Une facture est un document commercial, comptable et juridique détaillant la nature, la quantité et les conditions des achats et ventes de marchandises ou prestations de services. Elle est établie par toute structure juridique (société, association, micro-entreprise, etc.).

Dans le cadre de notre projet, une facture passe par **quatre** dates clés :

- Date de facturation : C'est la date à laquelle la facture a été créée.
- Date de réception de la facture : C'est la date à laquelle le client reçoit la facture.
- Date d'échéance contractuelle : C'est la date finale à laquelle la facture doit être payée. Elle est calculée en ajoutant le délai contractuel (qui est spécifique à chaque facture) à la date de réception de la facture.
- Date de réception du paiement : C'est la date à laquelle le client effectue le paiement.

Dans le cadre de notre projet, une facture passe par **deux** statuts :

- Encours : Si la facture n'a pas encore été payée.
- Payée : Si la facture a été payée.

Afin de réaliser un bilan annuel et mensuel, chaque société a des objectifs mensuels qui sont définis comme suit :

- Objectif encaissé : Le revenu des factures qui ont été payées au mois et à l'année de l'échéance contractuelle avec les impayés recouvrés (les factures qui ont été payées mais après leur échéance).
- Objectif non encaissé : Le montant des factures qui devaient être payées durant le mois courant mais qui ne l'ont pas été. Ce sont les impayés du mois.
- Non objectif encaissé : Le revenu des factures qui doivent être payées durant les prochains mois mais qui ont été payées durant le mois courant. En d'autres termes, c'est le revenu des factures qui n'ont pas été prévues pour le mois courant.

- Objectif du mois : C'est la somme des encaissements prévus durant le mois courant et donc la somme de l'objectif non encaissé et de l'objectif encaissé.
- KPI : C'est un indicateur de performance qui évalue la réalisation de l'objectif du mois et qui prend en compte l'équation suivante : $(\text{Objectif encaissé}/\text{Objectif du mois}) * 100$
- Chiffre d'affaires : C'est la somme des encaissements réels du mois courant et donc la somme de l'objectif encaissé et du non objectif encaissé.

Dans le cadre de notre projet, chaque facture s'inscrit dans l'un des objectifs suivants :

- Soit elle s'inscrit dans l'objectif encaissé si elle est payée à l'échéance ou après.
- Soit elle s'inscrit dans l'objectif non encaissé si elle est en cours.
- Soit elle s'inscrit dans le non objectif encaissé si elle est payée avant le mois durant lequel elle était sensée être payée. (C'est-à-dire avant le mois de l'échéance).

3.2 Problématique

La créance est une somme que la société perçoit de la part de ses clients dans un délai de paiement précis. Si le client ne paye pas sa créance, elle devient donc un impayé. En effet, si une entreprise vend des matériels ou des services sans aucun encaissement, ceci générera un écart entre les entrées et les sorties des flux de trésorerie ce qui est problématique. Par conséquent, même si leur activité fonctionne bien, l'entreprise peut également se retrouver en situation de détresse financière.

La société gère le suivi de paiement des factures et le recouvrement des créances manuellement et d'une manière non automatisée. Toutes les factures sont gérées seulement par le biais d'un sheet Excel et sans la trace d'une moindre visualisation pour aider à améliorer la gestion du recouvrement des créances et voir les objectifs du mois.

3.3 Objectifs

Notre objectif est donc de mettre en place une solution permettant d'optimiser :

- Le suivi du recouvrement des créances et de la rentabilité des clients dans le but d'améliorer la réalisation des objectifs.
- Automatisation du calcul relatif aux objectifs du mois et à la réalisation de ces objectifs.
- Automatisation de la génération des rapports d'analyse et de la génération de la source de données en cas de modification de celle-ci.

3.4 Solution proposée

RFC a décidé de mettre à la disposition du département finance une solution décisionnelle afin de définir les indicateurs de performance clé « KPI » pour suivre les impayés des clients, optimiser le recouvrement des créances et afin d'automatiser tout le processus. L'objectif de cette solution consiste à valoriser l'entrepôt de données en injectant une solution décisionnelle visant l'amélioration de la gestion financière de l'entreprise. Notre mission consiste à élaborer des tableaux de bord performants, efficents et dynamiques, contenant tous les indicateurs de performances répondant à tous les critères mais aussi de mettre en place une solution d'automatisation du processus afin que toute modification dans le fichier source soit vue dans les tableaux de bord que nous allons mettre en place.

4. Méthode de gestion du projet

La section suivante est consacrée à une comparaison entre différentes méthodes de travail ainsi que la présentation de la méthode choisie dans le cadre de notre projet.

4.1 Méthode Scrum

“Scrum est une méthodologie de développement informatique agile orientée projet avec des ressources régulièrement mises à jour. Comme toute méthode agile, Scrum a tendance à livrer rapidement un prototype, fonctionnant par définition, de manière à pouvoir être évalué par les clients et les membres de l'équipe.” [10]

Il existe cinq concepts Scrum à bien connaître :

- **Scrum master** : On dit souvent qu'un Scrum master est le coach. C'est vrai car son rôle est aussi de faire ressortir le meilleur de chacun et de faire du projet une réussite. Mais un Scrum Master est avant tout un chef de projet. Il s'assure que les principes Scrum fonctionnent comme ils le devraient, il définit les rôles, les horaires et les buts. Les encadrants académiques et professionnels remplissent ce rôle.
- **Product Owner** : Orienté métier, il partagera la vision du produit à réaliser avec l'équipe de développement. Dans le cadre de notre projet, c'est le département Finance.
- **Scrum Board** : C'est le tableau de bord du projet BI. Il doit être bien mis en place pour que chaque membre de l'équipe puisse en profiter. Il permet de suivre la progression du projet.
- **Squad** : C'est une équipe constituée d'une bonne dizaine de développeurs qui doivent être capables de conduire le projet en totale autonomie. Cette équipe est représentée par moi-même dans le cadre de ce projet de fin d'études.
- **Sprint** : C'est la phase la plus primordiale de développement du produit. Cette phase a une limite temporelle qui ne dépasse pas un mois avec une finalité qui est de rendre un produit présentable au client. La méthode Scrum agit par itération.

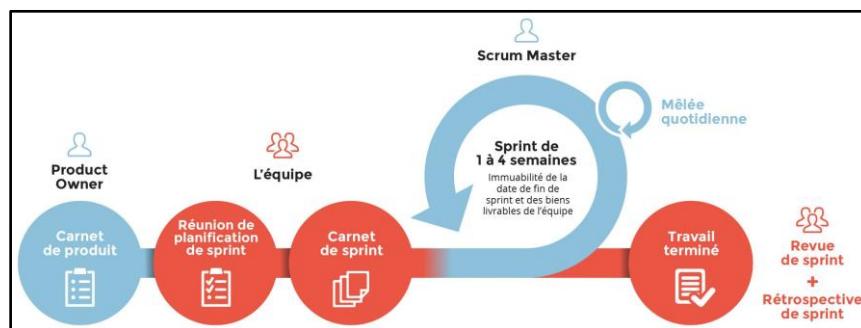


Figure 17 : Méthode SCRUM

4.2 Méthode GIMSI

“GIMSI est une méthode de conception et de réalisation du système de pilotage et d'aide à la décision à base de tableaux de bord, fondée sur la communication. GIMSI se compose de 4 phases régies par dix étapes pour gérer tous les aspects du projet BI depuis l'étude de l'environnement jusqu'à la mise en œuvre en passant par l'accompagnement du changement.”[11]

Elle se résume en quatre phases et dix étapes comme le montre la figure ci-dessous (Figure 18) :

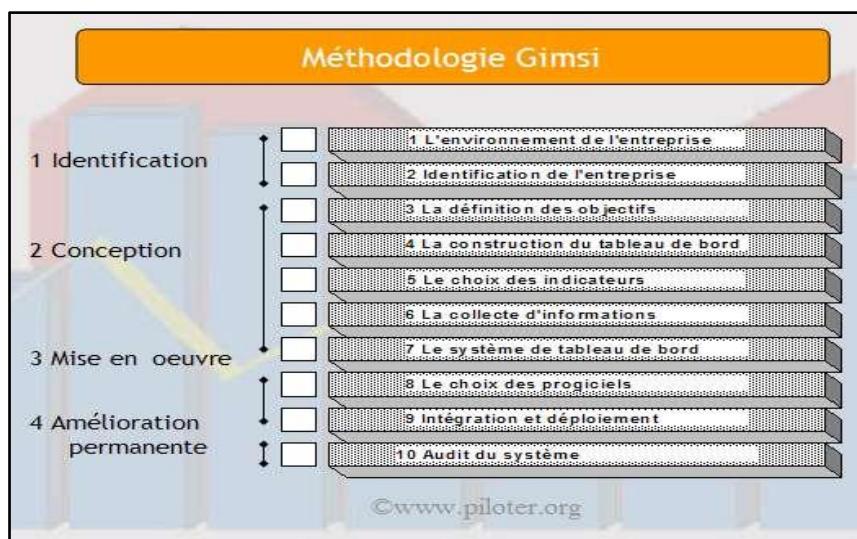


Figure 18 : Méthode GIMSI

4.3 Méthodologie de travail adoptée

Le choix de la méthodologie est une étape nécessaire pour mener le projet à bout avec succès. La méthodologie nous permet d'avoir un travail bien structuré, de la planification à la mise en œuvre du projet.

L'identification de l'entreprise, de ses objectifs et des aspects du projet BI est un pallier capital et représente le point de départ de son élaboration. Cette étape nous permettra d'avoir une perception générale, à commencer par identifier les outils, les contraintes, les risques, l'environnement afin de réaliser le projet. Ensuite, une fois les objectifs définis, nous pouvons déterminer les tâches exactes de chaque étape pour le bon fonctionnement du processus pour ensuite nous lancer dans les phases de conception et mise en œuvre du projet.

Après avoir étudié les divers stades du choix de la méthodologie, l'approche GIMSI a été adoptée étant donné que son processus correspond le plus à notre projet et satisfait nos besoins.

Conclusion :

Dans ce chapitre, nous avons présenté l'organisme d'accueil en détails, le cadre général du projet en expliquant les concepts théoriques utilisés dans le cadre de notre projet. Nous avons notamment mis en lumière le cadre spécifique du projet ainsi que la méthodologie choisie pour mener à bien ce dernier. Le chapitre suivant traitera l'environnement du projet.

Chapitre 2 : L'environnement

Introduction

Au niveau de ce chapitre, nous présenterons l'environnement matériel, l'environnement de logiciel ainsi que la préparation de l'environnement logiciel.

1.L'environnement de travail

1.1 L'environnement matériel

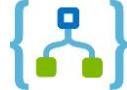
Afin d'exécuter toutes les activités du projet, nous avons utilisé un ordinateur portable qui possède les caractéristiques suivantes :

Marque	Asus
Système d'exploitation	Windows 11 Famille Unilingue
Processeur	Intel(R) Core(TM)
Type de processeur	i7-1065G7 CPU
Fréquence du processeur	1.50 GHz
RAM	8,00 Go

Tableau 3 : L'environnement matériel

1.2 L'environnement logiciel

Au cours de développement de notre solution, nous avons recouru aux logiciels et services ci-dessous :

Logo	Nom	Description
	Microsoft Excel	Affichage des données source.
	Microsoft teams	Organisation des réunions d'encadrement et extraction du fichier source du projet.
	Star UML	Réalisation du schéma de la conception de l'entrepôt de données.
		Logic Apps Extraction du fichier source du Microsoft Teams vers notre plateforme d'analyse et automatisation du processus.
		Azure Synapse Analytics Construction de l'entrepôt de données, réalisation des transformations nécessaires des données et chargement dans la Data Warehouse.
	Power BI Desktop	Réalisation du tableau de bord.
	Portail Power BI en ligne	Afficher les visualisations sur Azure Synapse Analytics

	Google Docs	Rédaction du rapport
---	-------------	----------------------

Tableau 4 : L'environnement logiciel

2. Préparation de l'environnement

Afin de préparer notre environnement, nous avons préparé les configurations présentées ci-dessous :

2.1 L'abonnement Azure

Pour accéder à la plateforme Portail Azure, l'équipe RFC m'a créé un compte Microsoft avec l'adresse email "yasmine.benlarbi@esprit.tn" avec un abonnement "Stagiaire - Yasmine" comme nous pouvons l'apercevoir dans la figure ci-dessous. (Figure 19)



Informations	Valeurs
ID d'abonnement	e0cad13c-21b8-4292-bbcd-f1a95e230e32
Répertoire	RFC (RFCSA.onmicrosoft.com)
Mon rôle	Propriétaire
Offre	MSDN
ID de l'offre	MS-AZR-0029P
Groupe d'administration	7917799a-9669-4e41-b87d-4c14fef7c41
Nom de l'abonnement	Stagiaire - Yasmine
Période de facturation actuelle	Chargement...
Devise	USD
État	Actif
Degré de sécurisation	Indisponible

Figure 19 : L'abonnement Azure

2.2 Création de l'espace de travail et du groupe de ressource

Les ressources, les solutions et les services utilisés de portail azure sont groupés dans un groupe de ressources où se trouve notre espace de travail.

Pour la création du groupe de ressources et de l'espace de travail, nous avons sélectionné l'abonnement créé par RFC et nous avons nommé le groupe de ressources "Pfe-yasmine" comme le montre la figure ci-dessous. (Figure 20)

Détails du projet

Sélectionnez l'abonnement pour gérer les coûts et les ressources déployées. Utilisez les groupes de ressources comme les dossiers pour organiser et gérer toutes vos ressources.

Abonnement * ⓘ Stagiaire - Yasmine

Groupe de ressources * ⓘ (Nouveau) Pfe-yasmine

Groupe de ressources managé ⓘ ressourcepfe-yasmine

Figure 20 : Création du groupe de ressources

Comme le montrent les figures 21 et 22, nous avons ensuite nommé notre espace de travail "worksapceyasminepfe" en choisissant la région "East-US". Le Workspace est la plateforme sur laquelle auront lieu les transformations et où seront affichées nos visualisations à la fin du projet. Nous avons notamment créé un compte de stockage sous le nom "yasminepfestockage" avec un système de fichier nommé "projetpfesys" où sera stocké notre fichier source à chaque fois qu'il sera extrait et généré.

Détails de l'espace de travail

Nommez votre espace de travail, sélectionnez un emplacement et choisissez un système de fichiers Data Lake Storage Gen2 principal comme emplacement par défaut des journaux et de la sortie du travail.

Nom de l'espace de travail * worksapceyasminepfe

Région * East US

Sélectionner Data Lake Storage Gen2 *

À partir de l'abonnement Manuellement via l'URL

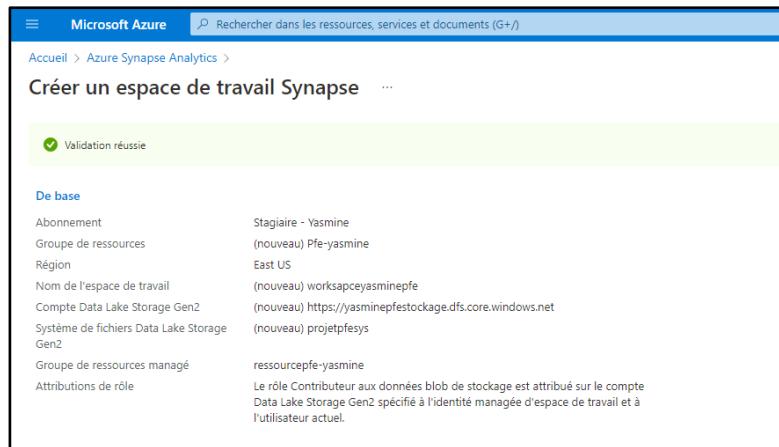
Nom du compte * ⓘ (Nouveau) yasminepfestockage

Nom du système de fichiers * ⓘ (Nouveau) projetpfesys

M'attribuer le rôle Contributeur de données blob de stockage sur le compte Data Lake Storage Gen2 pour l'interroger de manière interactive dans l'espace de travail.

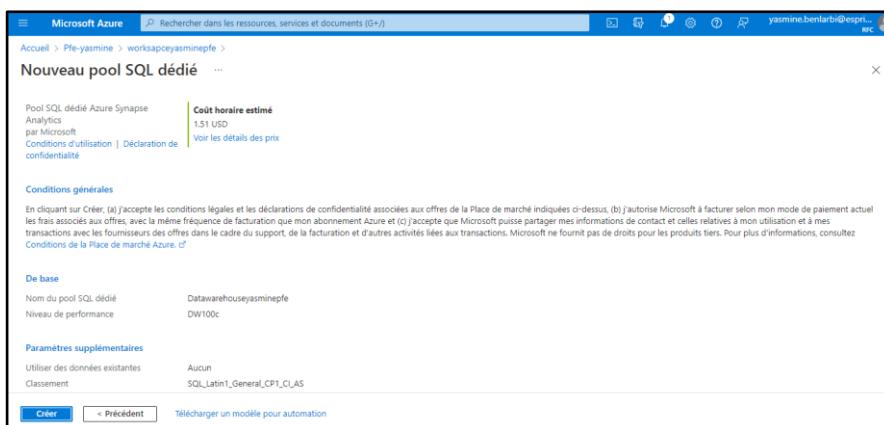
Vérifier + créer < Précédent Suivant : Sécurité >

Figure 21 : Création de l'espace de travail (Partie 1)

**Figure 22 : Création de l'espace de travail (Partie 2)**

2.3 Configuration du pool SQL dédié

Après la configuration du Workspace, nous avons configuré notre pool SQL dédié (nommé SQL DW anciennement). Nous l'avons nommé “Datawarehouseyasminepfe” en choisissant DW100c comme un niveau de performance comme le montre la figure ci-dessous.

**Figure 23 : Configuration du pool SQL dédié**

2.4 Configuration du service lié : Logic Apps

Logic Apps est l'application logique qui nous permettra d'extraire le fichier source mais qui permettra notamment l'automatisation de la génération du fichier source ce qui veut dire que quand une modification touchera le fichier source, cette dernière apparaîtra dans les tableaux de bord automatiquement après l'exécution du pipeline. A l'issue de ces étapes, quand on modifiera le fichier source, le nouveau fichier source modifié sera automatiquement stocké

dans le Data Lake, plus spécifiquement dans « projetpfesys » et on devra ensuite exécuter le pipeline pour que les modifications soient vues au niveau des tableaux de bord.

Pour créer un modèle d'application logique, on écrit “Logic apps” dans la barre de recherche dans portal azure (Figure 24). On clique sur le bouton “ajouter” pour remplir le formulaire comme suivant (Figure 25) :

- Abonnement : “Stagiaire - Yasmine”
- Groupe de ressource : “Pfe-yasmine”
- Nom de l'application logique : “LogicApp”
- Publier : workflow
- Région : East-US
- Type de plan: Standard.

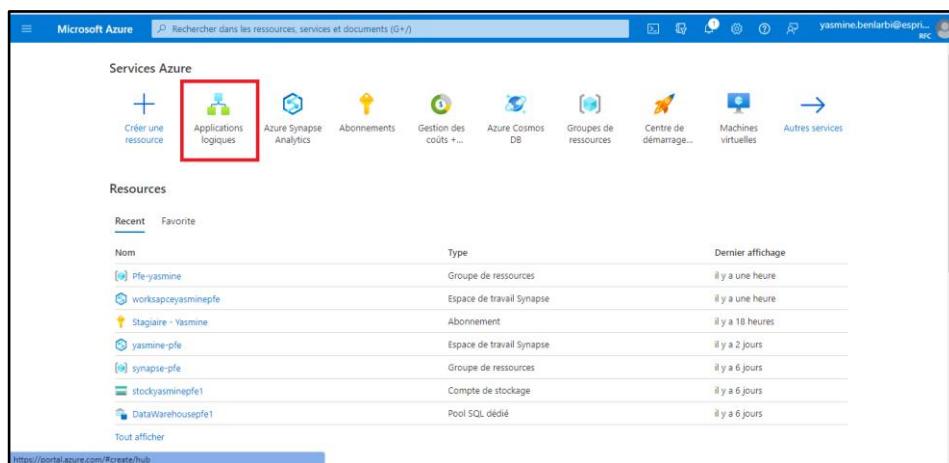


Figure 24 : Configuration du service lié : Logic Apps (Partie 1)

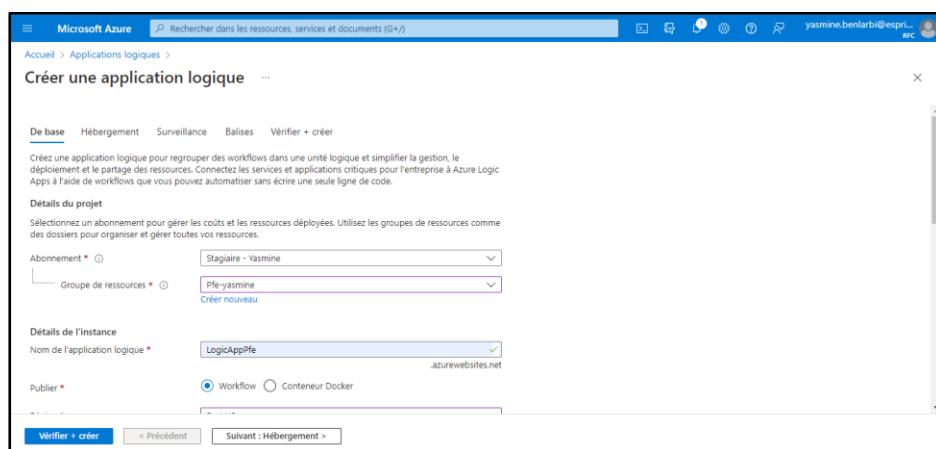


Figure 25 : Configuration du service lié : Logic Apps (Partie 2)

2.5 Configuration du service lié : Power BI

Afin d'effectuer la liaison entre Power BI et Azure Synapse, nous avons suivi les étapes suivantes :

La société m'a créé un compte Power BI en ligne qui servira à lier Azure Synapse en ligne et Power BI Desktop. Pour faire cela on devra tout d'abord créer un Workspace sur la plateforme en ligne de Power BI qu'on nommera "workspaceyasmine" comme l'indique la figure ci-dessous. (Figure 26)

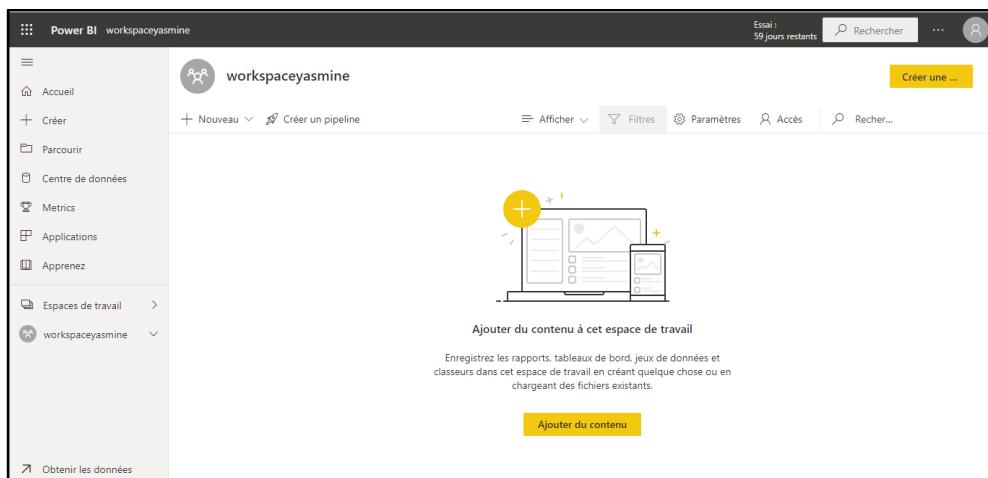


Figure 26 : Création du workspace Power BI en ligne

Dans Azure Synapse, on clique sur le hub "gérer" dans le menu à gauche puis on clique sur les services liés ensuite sur "Nouveau/New" comme le montre la figure 27.

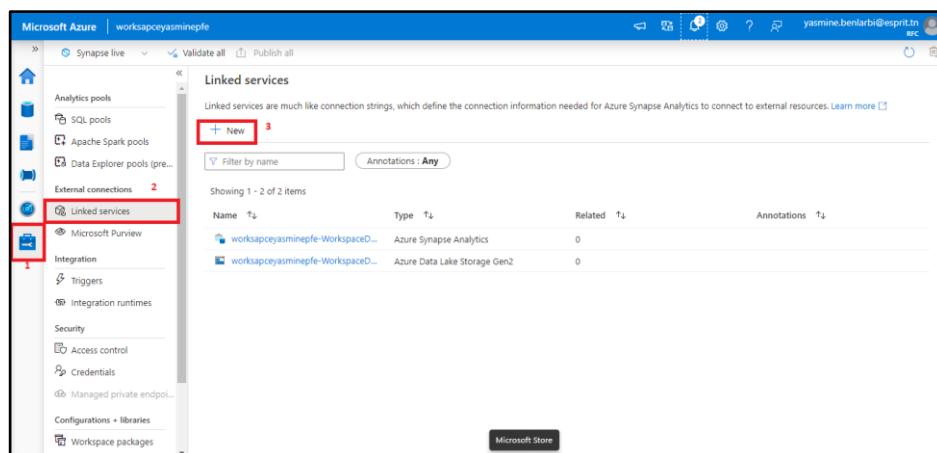


Figure 27 : Configuration du service lié : Power BI (Partie 1)

Dans le menu ouvert, on tape “Power BI” dans la barre de recherche et on choisit “Power BI” puis on clique sur continuer comme le montre la figure 28.

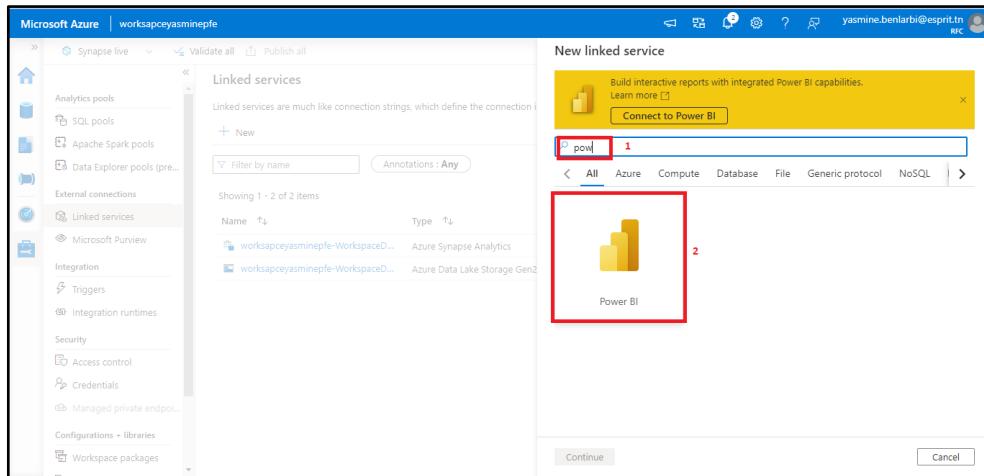


Figure 28 : Configuration du service lié : Power BI (Partie 2)

On nomme notre service lié “yasminepfpowerbi” et on sélectionne notre espace de travail “workspaceyasmine” qui se trouve sur le site power BI puis on clique sur créer. (Figure 29)

A screenshot of the 'Modifier le service lié' (Edit linked service) dialog for Power BI. The dialog has several input fields: 'Nom *' with the value 'yasminepfpowerbi' (1), 'Locataire' dropdown with the value 'esprit.tn (513486ec-6643-4f17-a508-76478311be42)' (2), and 'Nom de l'espace de travail *' dropdown with the value 'workspaceyasmine (62cfdf5-f4e6-42e0-8f13-5bb4841d9ab5)' (3). There are also 'Annotations' and 'Avancé' buttons. At the bottom, there are 'Enregistrer' (4) and 'Annuler' buttons, with the 'Enregistrer' button highlighted by a red box.

Figure 29 : Configuration du service lié : Power BI (Partie 3)

Conclusion

Nous avons introduit l'environnement de travail de notre projet et nous avons préparé l'environnement logiciel pour ensuite démarrer le projet sur de bonnes bases.

Cependant, une conception préétablie doit être présentée avant la réalisation de l'entrepôt. Au niveau du chapitre suivant, nous choisirons l'approche de modélisation et nous présenterons la modélisation de l'entrepôt de données.

Chapitre 3 : Conception de la solution

Introduction

Tout projet BI nécessite une phase de conception qui nécessite à son tour une méthodologie de conception et de modélisation du DW/DM.

Dans ce chapitre, nous présentons la méthodologie choisie pour la conception et la modélisation du Data Warehouse/Datamart ainsi que les étapes d'une partie nécessaire du volet décisionnel de notre projet. Il s'agit de la conception et la réalisation d'un Datamart pour l'activité du recouvrement des créances du département Finance.

1. Choix de l'approche de modélisation

1.1 Kimball

La méthodologie Kimball est une méthode dédiée à la conception et la réalisation d'architecture d'un entrepôt de données. Elle est appelée aussi 'BOTTOM-UP' puisque c'est une approche ascendante où les datamarts sont construits en fonction des besoins de l'entreprise.

1.2 Inmon

La méthodologie Inmon est une méthode dédiée à la conception et la réalisation d'architecture d'un entrepôt de données. Elle est appelée aussi 'TOP-DOWN' puisque c'est une approche descendante où les Datamarts sont créés à partir d'un entrepôt de données centralisé. Cette méthodologie consiste à réaliser la conception du l'entrepôt d'une façon intégrale. Les dimensions et les faits doivent être connues à l'avance.

1.3 Kimball vs Inmon

Le tableau ci-dessous présente la différence entre la méthodologie Kimball et Inmon

Paramètres	Kimball	Inmon
Présenté par	Ralph Kimball	Bill Inmon
Approche	Ascendante	Descendante
Intégration de données	Concentration sur les domaines d'activité individuels	Concentration sur les domaines à l'échelle de l'entreprise
Temps de construction	Simple et moins de temps	Complexé et plus de temps
Coût	Rentable	Coût initial énorme et coût de développement faible
Compétences requises	Pas de compétences requises	Compétences spécialisées
Maintenance	Difficile	Facile
Modèle de données	Préfère que les données soient dans un modèle dénormalisé	Préfère que les données soient dans un modèle normalisé
Système de stockage de données	Stables	Taux de changement élevé

Tableau 5 : Kimball vs Inmon

Après l'étude des deux méthodologies, nous avons décidé que la méthodologie Kimball était la plus adéquate pour la réalisation de la modélisation de notre entrepôt de données.

2. Modélisation conceptuelle des données

Nous avons décidé de modéliser nos données en utilisant le schéma en flocon comme le montre la figure ci-dessous (Figure 30).

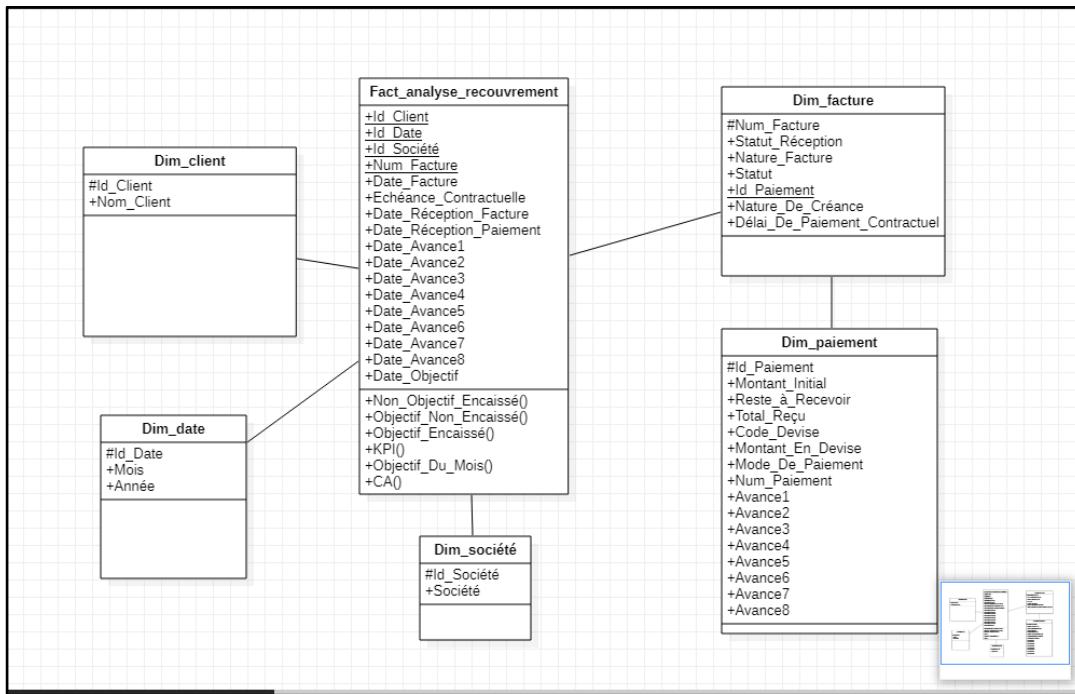


Figure 30 : Modélisation conceptuelle du Data Warehouse

Pour chaque table de notre entrepôt de données/Datamart, nous avons mis en place un tableau synthétisant les attributs utilisés, les types de ces attributs ainsi qu'une description de ces derniers.

2.1 La table de faits

2.1.1 Table de faits “Fact_analyse_recouvrement”

Le tableau ci-dessous représente le dictionnaire de données des attributs de la table de faits “Fact_analyse_recouvrement”.

Attributs	Type	Description
Id_Client	Integer	Clé étrangère qui fait référence à l'attribut Id_Client de la table Dim_client.
Id_Date	Integer	Clé étrangère qui fait références à l'attribut Id_date de la table Dim_date.
Id_Société	Integer	Clé étrangère qui fait référence à l'attribut Id_Société de la table Dim_société.
Num_Facture	String	Clé étrangère qui fait référence à l'attribut Num_Facture de la table Dim_facture.
Date_Facture	Date	Date de facturation.
Echéance_Contractuelle	Date	Date de l'échéance contractuelle.
Date_Réception_Facture	Date	Date de réception de la facture.
Date_Réception_Paiement	Date	Date de réception du paiement.
Date_Avance1	Date	Date de la première avance.
Date_Avance2	Date	Date de la deuxième avance.
Date_Avance3	Date	Date de la troisième avance.
Date_Avance4	Date	Date de la quatrième avance.
Date_Avance5	Date	Date de la cinquième avance.
Date_Avance6	Date	Date de la sixième avance.
Date_Avance7	Date	Date de la septième avance.
Date_Avance8	Date	Date de la huitième avance.
Date_Objectif	Date	Date qui sert à répartir les objectifs chaque mois.

Tableau 6 : Dictionnaire des données “Fact_analyse”

Le tableau ci-dessous représente le dictionnaire de données des mesures de la table de faits “Fact_analyse_recouvrement”.

Mesures	Types	Description
Non_Objectif_Encaissé	Float	Montant des revenus qui n'ont pas été prévus pour le mois courant.
Objectif_Non_Encaissé	Float	Montant des impayés du mois.
Objectif_Encaissé	Float	Correspond aux revenus des factures à l'échéance et des impayés recouvrés.
KPI	Float	Pourcentage qui correspond à l'indicateur clé de performance de chaque mois.
Objectif_Du_Mois	Float	Montant qui doit être atteint chaque mois.
CA	Float	Chiffre d'affaires réalisé chaque mois.

Tableau 7 : Dictionnaire des données “Fact_analyse”

2.2 Les tables de dimension

2.2.1 Table de dimension “Dim_date”

Le tableau ci-dessous représente le dictionnaire de données des attributs de la table de dimension “Dim_date”.

Attributs	Type	Description
Id_Date	Date	Clé primaire de la table qui fait référence à l'id des dates à laquelle les objectifs doivent être pris en compte. Les factures qui font référence aux objectifs encaissés et les non objectifs encaissés sont comptabilisés grâce à la date de paiement tandis que les factures qui font référence objectifs non encaissés sont pris en compte grâce à la date objectif car ils n'ont pas de date de paiement.
Mois	Integer	Mois des objectifs auquel les factures doivent être pris en compte.
Année	Integer	Année des objectifs à laquelle les factures doivent être prises en compte.

Tableau 8 : Dictionnaire des données “Dim_date”

2.2.2 Table de dimension “Dim_client”

Le tableau ci-dessous représente le dictionnaire de données des attributs de la table de dimension “Dim_client”.

Attributs	Type	Description
Id_Client	Integer	Clé primaire de la table qui fait référence à l'id de chaque client.
Nom_Client	String	Nom des clients.

Tableau 9 : Dictionnaire des données “Dim_client”

2.2.3 Table de dimension “Dim_société”

Le tableau ci-dessous représente le dictionnaire de données des attributs de la table de dimension “Dim_société”.

Attributs	Type	Description
Id_Société	Integer	Clé primaire de la table qui fait référence à l'id de chaque société.
Société	String	Nom de la société

Tableau 10: Dictionnaire des données “Dim_société”

2.2.4 Table de dimension “Dim_facture”

Le tableau ci-dessous représente le dictionnaire de données des attributs de la table de dimension “Dim_facture”.

Attributs	Type	Description
Num_Facture	String	Clé primaire qui fait référence au numéro de la facture.
Statut_Réception	String	Statut de réception de la facture.
Nature_Facture	String	Nature de la facture (Contrat/matériel/service/formation).

Statut	String	Statut de la facture (payé ou en cours).
Id_Paiement	Integer	Clé étrangère qui fait référence à l'attribut Id_Paiement de la table Dim_paiement.
Nature_De_Créance	String	Nature de la créance (douteuse/normale/problème administratif).
Délai_De_Paiement_Contractuel	Integer	Délai qu'un client a pour payer sa facture.

Tableau 11 : Dictionnaire des données “Dim_facture”

2.2.5 Table de dimension “Dim_paiement”

Le tableau ci-dessous représente le dictionnaire de données des attributs de la table de dimension “Dim_paiement”.

Attributs	Type	Description
Id_Paiement	Integer	Clé primaire qui fait référence à l'id de chaque paiement qui a été fait.
Montant_Initial	Float	Montant qui doit être payé avant d'appliquer le taux de change.
Reste_à_Recevoir	Float	Montant encore non payé par le client.
Total_Reçu	Float	Montant qui a été reçu par le client.
Code_Devise	Float	Taux de change.
Montant_En_Devise	Float	Montant qui doit être payé après application du taux de change.
Mode_De_Paiement	String	Mode de paiement.
Num_Paiement	String	Numéro du paiement.
Avance1	Float	Première avance effectuée par le client.
Avance2	Float	Deuxième avance effectuée par le client.
Avance3	Float	Troisième avance effectuée par le client.
Avance4	Float	Quatrième avance effectuée par le client.

Avance5	Float	Cinquième avance effectuée par le client.
Avance6	Float	Sixième avance effectuée par le client.
Avance7	Float	Septième avance effectuée par le client.
Avance8	Float	Huitième avance effectuée par le client.

Tableau 12 : Dictionnaire des données “Dim_paiement”

Conclusion

Nous avons choisi l'approche de modélisation et nous avons modélisé notre entrepôt de données pour ensuite travailler convenablement. Au niveau du chapitre suivant, nous présenterons les étapes de réalisation du projet.

Chapitre 4 : Réalisation du projet

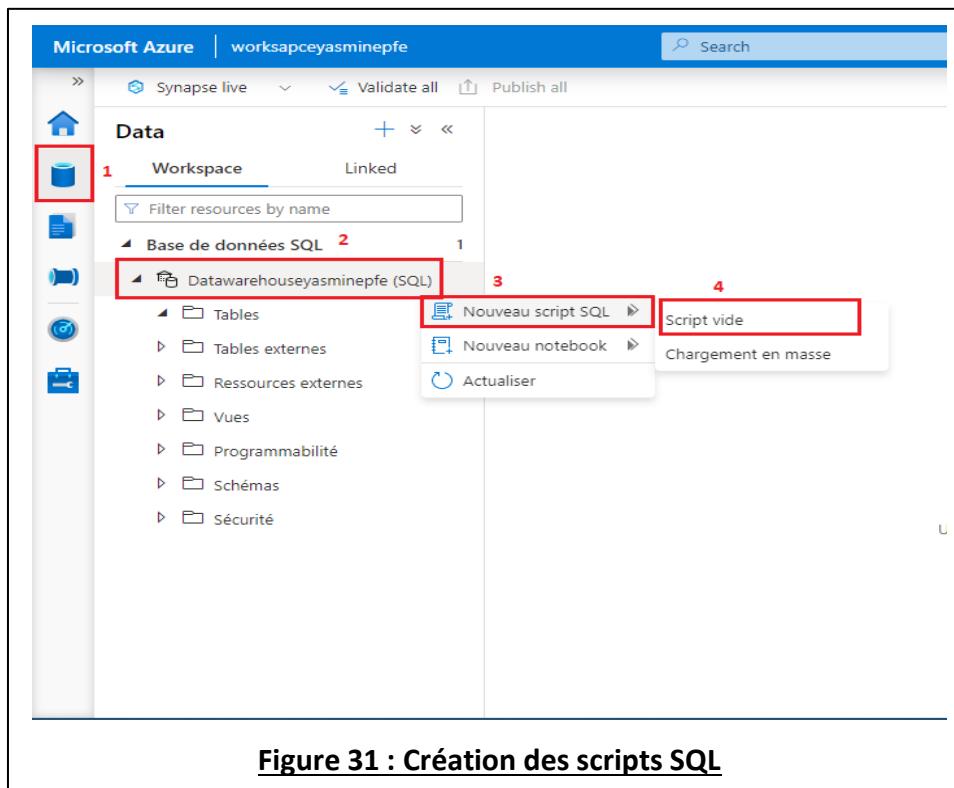
Introduction

En se basant sur la méthodologie Kimball, nous présenterons au niveau de ce chapitre les étapes de la réalisation de la solution proposée. Pour cela, nous construirons l'entrepôt de données à partir de la modélisation établie suivie par l'extraction et l'automatisation de la génération du fichier source par le biais de Logic Apps ensuite nous passerons par les transformations nécessaires pour ensuite continuer avec le chargement des données dans le datawarehouse pour conclure avec la visualisation des données.

1. Construction de l'entrepôt de données

Après la configuration du pool SQL dédié (Chapitre 2), nous pouvons créer nos tables tout en nous basant sur la conception faite avec StarUML. Pour cela, on doit suivre les étapes suivantes :

On sélectionne le hub “Données” dans le menu à gauche. On fait un clic droit sur Datawarehouseyasminepfe et on clique sur nouveau script SQL et ensuite script vide pour créer les tables de dimension et la table de faits. (Figure 31)



Nous procémons maintenant à la création des différentes tables.

La figure ci-dessous (Figure 32) montre la création de la table de dimension Dim_client.

```
1  --Création de la table de dimension Dim_client
2  CREATE TABLE [dbo].[Dim_client]
3  (
4    Id_Client INT,
5    Nom_Client VARCHAR(100) NOT NULL
6  )
7  GO
```

Figure 32 : Création de la table de dimension Dim_client

La figure ci-dessous (Figure 33) montre la création de la table de dimension Dim_date.

```
1  --Création de la table de dimension Dim_date
2  CREATE TABLE [dbo].[Dim_date]
3  (
4    Id_Date INT,
5    Mois INT,
6    Année INT
7  )
8  GO
9
```

Figure 33 : Création de la table de dimension Dim_date

La figure ci-dessous (Figure 34) montre la création de la table de dimension Dim_facture.

```
1  --Création de la table Dim_facture
2  CREATE TABLE [dbo].[Dim_facture]
3  (
4    Num_Facture VARCHAR(100),
5    Statut_Réception VARCHAR(100),
6    Nature_Facture VARCHAR(100),
7    Nature_De_Créance VARCHAR(100),
8    Id_Paiement INT,
9    Délai_De_Paiement_Contractuel INT,
10   Statut VARCHAR(100)
11  )
12  GO
13
```

Figure 34 : Création de la table de dimension Dim_facture

La figure ci-dessous (Figure 35) montre la création de la table de dimension Dim_paiement.

```
1  -- Création de la table Dim_paiement
2  CREATE TABLE [dbo].[Dim_paiement]
3  (
4      Id_Paiement INT,
5      Montant_Initial FLOAT,
6      Reste_à_Recevoir FLOAT,
7      Total_Reçu FLOAT,
8      Code_Devise FLOAT,
9      Montant_En_Devise FLOAT,
10     Num_Paiement VARCHAR(100),
11     Mode_De_Paiement VARCHAR(100),
12     Avance1 FLOAT,
13     Avance2 FLOAT,
14     Avance3 FLOAT,
15     Avance4 FLOAT,
16     Avance5 FLOAT,
17     Avance6 FLOAT,
18     Avance7 FLOAT,
19     Avance8 FLOAT
20 )
21 GO
```

Figure 35 : Crédation de la table de dimension Dim_paiement

La figure ci-dessous (Figure 36) montre la création de la table de dimension Dim_société.

```
--Création de la table Dim_société
CREATE TABLE [dbo].[Dim_société]
(
    Id_Société INT,
    Société VARCHAR(50)
)
```

Figure 36 : Crédation de la table de dimension Dim_société

La figure ci-dessous (Figure 37) montre la création de la table de faits Fact_analyse_recouvrement.

```
1  --Création de la table Fact_analyse_recouvrement
2  CREATE TABLE [dbo].[Fact_analyse_recouvrement]
3  (
4      Id_Client INT,
5      Id_Date INT,
6      Id_Société INT,
7      Num_Facture VARCHAR(100),
8      Date_Facture DATE,
9      Echéance_Contractuelle DATE,
10     Date_Réception_Facture DATE,
11     Date_Réception_Paiement DATE,
12     Date_Avance1 DATE,
13     Date_Avance2 DATE,
14     Date_Avance3 DATE,
15     Date_Avance4 DATE,
16     Date_Avance5 DATE,
17     Date_Avance6 DATE,
18     Date_Avance7 DATE,
19     Date_Avance8 DATE,
20     Date_Objectif VARCHAR(40),
21     Non_Objectif_Encaissé FLOAT,
22     Objectif_Encaissé FLOAT,
23     Objectif_Non_Encaissé FLOAT,
24     KPI FLOAT,
25     Objectif_Du_Mois FLOAT,
26     CA FLOAT
27 )
28 GO
29
```

Figure 37: Crédit à la création de la table de faits Fact_analyse_recouvrement

2. Logic Apps : Extraction et automatisation de la génération du fichier source

Logic Apps” qui est un iPaaS (Integration Platform as a Service) qui simplifie et automatise les solutions d'intégrations et les workflows tout en intégrant nos processus, applications et données. L'objectif de Logic Apps est l'extraction pour héberger le fichier source vers notre espace de travail ainsi que l'automatisation de la génération du fichier source pour que toute modification qui touche le fichier source soit détectée dans le data Lake et plus précisément dans le fichier “projetpfesys”.

Pour commencer cette partie, nous commencerons par extraire le fichier source de Teams qui est partagé et accessible par les membres qui sont concernés par ce dernier. Le fichier Excel est nommé Reporting recouvrement RFC_Compusell(1) et les données qui s'y trouvent sont des données réelles. Le fichier se trouve dans un canal appelé Test et partagé avec les membres de l'équipe Test comme le montre la figure ci-dessous. (Figure 38)

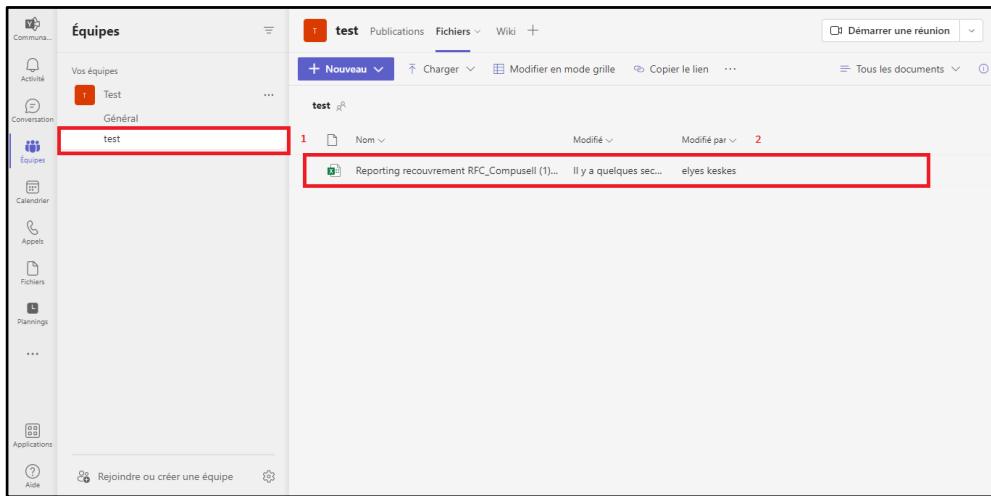


Figure 38 : Fichier source dans Microsoft Teams

Nous avons ensuite ouvert l'application que nous avons nommée LogicApp pour ouvrir le Logic app designer qui est l'outil de conception de la solution d'extraction et d'automatisation comme le montre la figure 39.

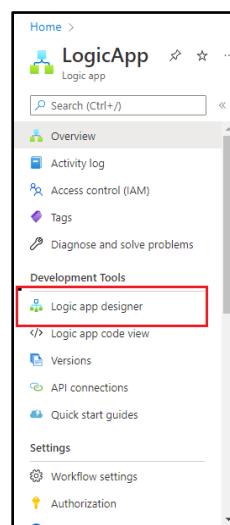


Figure 39 : Logic App Designer

On sélectionne “when a https request is received” sans remplir aucun champ ensuite on sélectionne “Choose an operation” et on tape dans la barre de recherche “list folder share point” puis on choisit “list folder” comme le montre la figure 40.

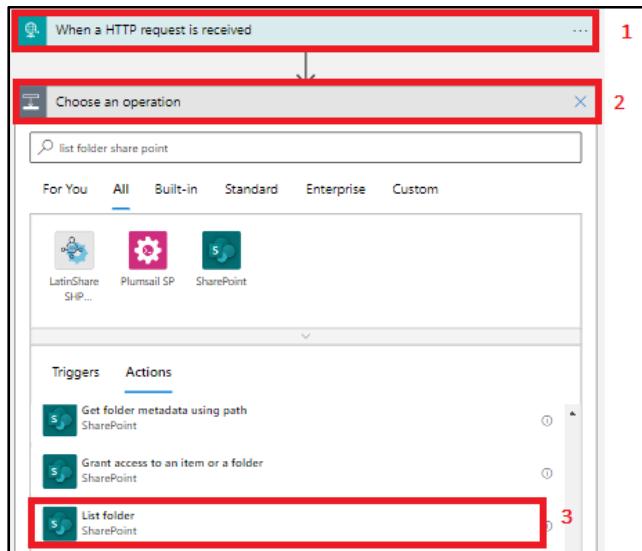


Figure 40 : Logic App (Partie 1)

Dans Microsoft Teams, on fait un clic droit sur le fichier source puis on sélectionne “ouvrir dans SharePoint” comme indiqué dans la figure 41.

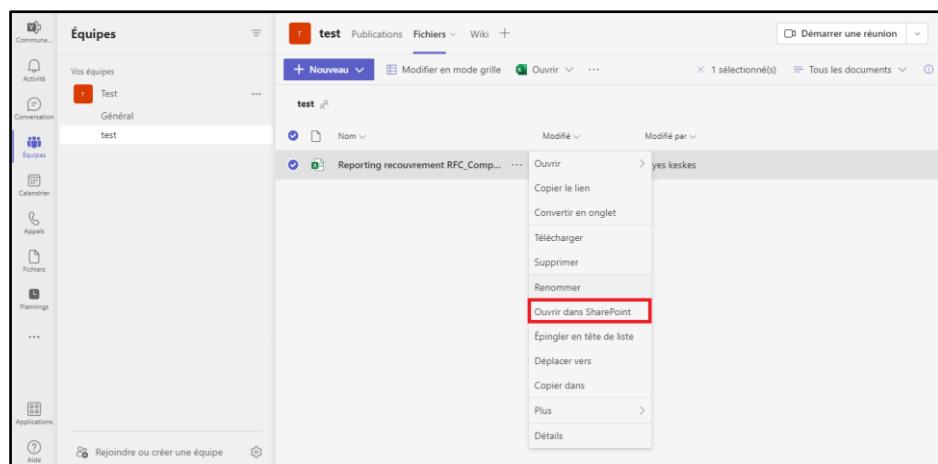


Figure 41: Logic App (Partie 2)

Nous pouvons maintenant apercevoir la plateforme de SharePoint de la société, on refait un clic droit sur le fichier et on copie le lien. (Figure 42)

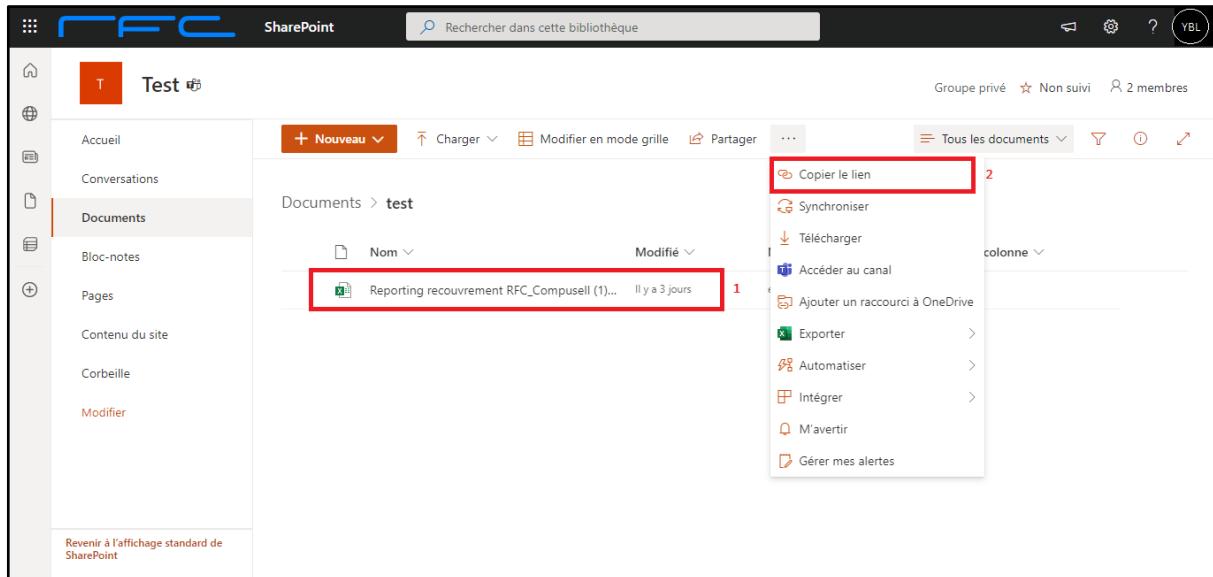


Figure 42 : Logic App (Partie 3)

On retourne au Logic app designer, on colle le lien du fichier affiché dans le SharePoint dans la barre d'adresse du site en identifiant l'identificateur de fichier qui est en d'autres termes le chemin du fichier. (Figure 43)

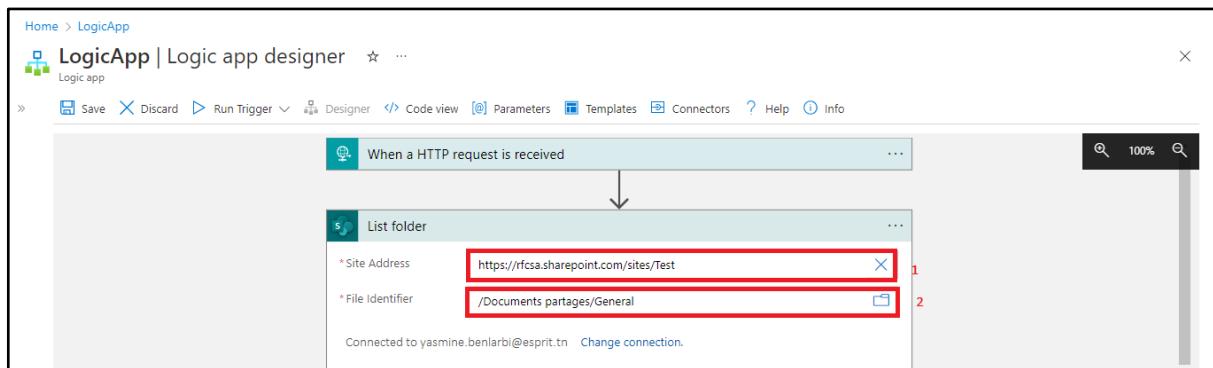


Figure 43 : Logic App (Partie 4)

On ajoute une nouvelle étape et on sélectionne le commutateur “for each”. (Figure 44)

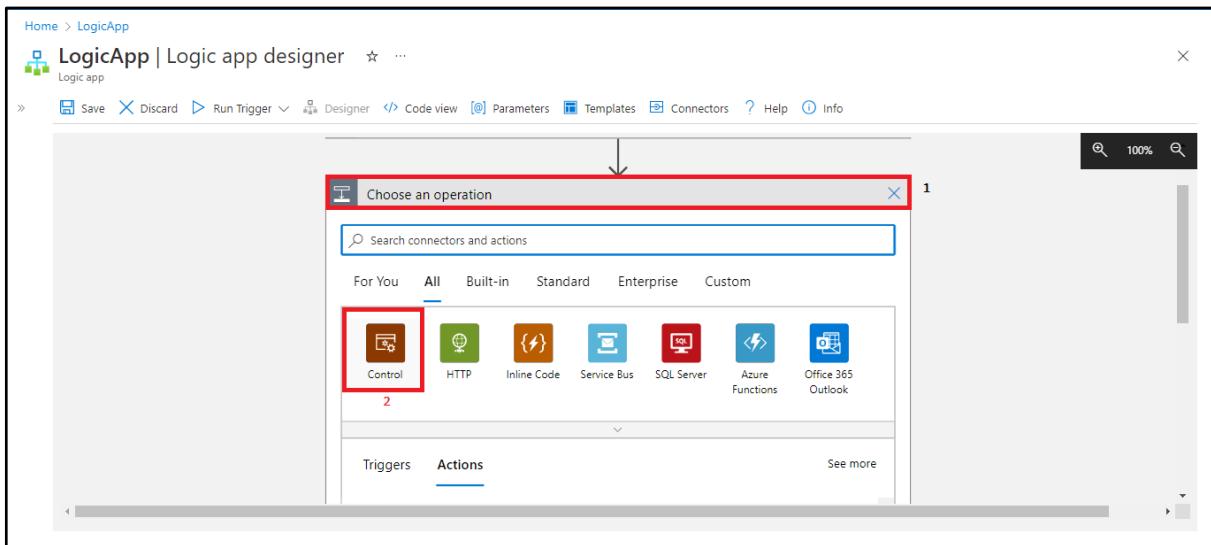
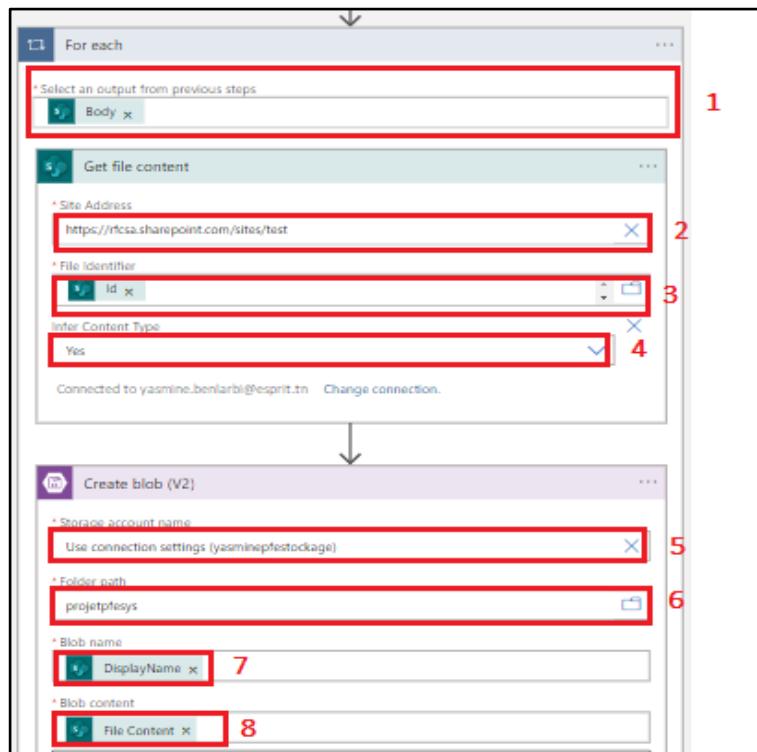


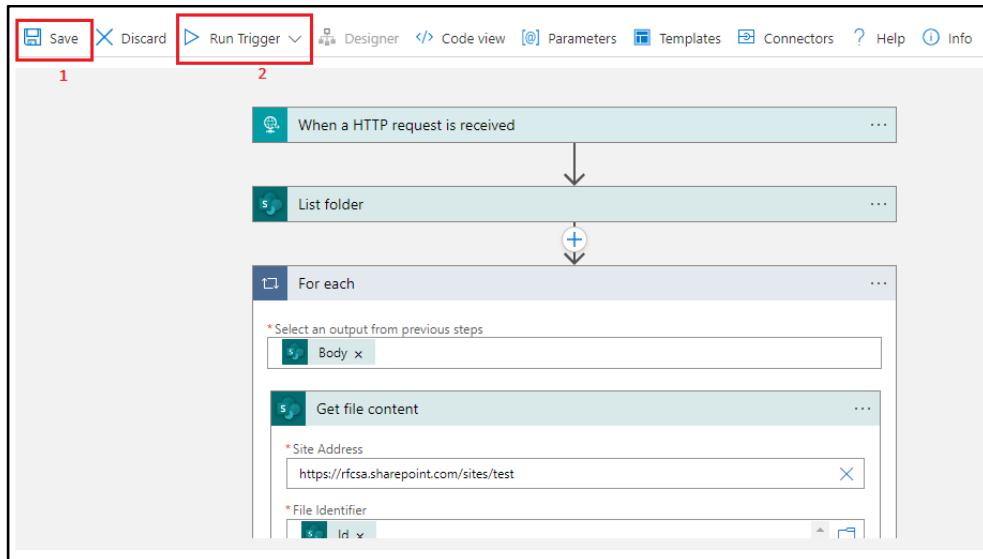
Figure 44 : Logic App (Partie 5)

Comme le montre la figure 45, dans le champ “sélectionnez un résultat”, on sélectionne “Body”. On colle le lien du SharePoint dans l’adresse du site et on sélectionne Id dans l’identificateur du fichier.

Dans le champ déduire le type de contenu, on sélectionne “oui”. On ajoute une action et on choisit “Create Blob v2”. On le connecte avec notre compte de stockage “yasminepfestockage” en indiquant le chemin d'accès, ici “projetpfesys”, du fichier dans lequel nous voulons stocker le fichier dans azure synapse. Dans le champ “Nom de l'objet Blob”, on sélectionne “DisplayName”. Dans le champ “Contenu de l'objet Blob”, on choisit “File Content”

**Figure 45 : Logic App (Partie 6)**

Ensuite, on enregistre et on débogue pour voir si toutes les étapes mises en place sont fonctionnelles comme l'indique la figure 46. A chaque fois que le fichier source sera modifié, une nouvelle copie sera placée dans le Data Lake dans Azure Synapse.

**Figure 46 : Logic App (Partie 7)**

Après l'enregistrement de l'application logique, nous devons créer un pipeline qui est un groupe logique d'activités. Ces activités vont être exécutées ensemble pour avoir en output une solution finale et fonctionnelle. Nous allons ajouter ce que nous avons fait précédemment grâce à Logic Apps au pipeline pour ensuite lier cela au reste des transformations que nous ferons.

Dans Azure synapse, on clique sur l'icône “intégrer” dans le menu à gauche. On ajoute ensuite un pipeline comme le montre la figure 47.

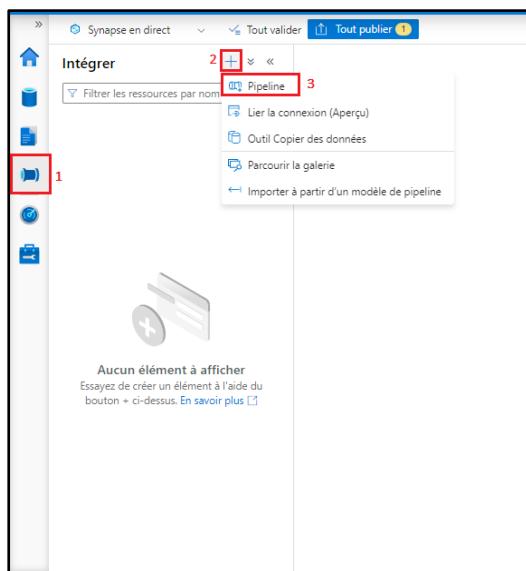
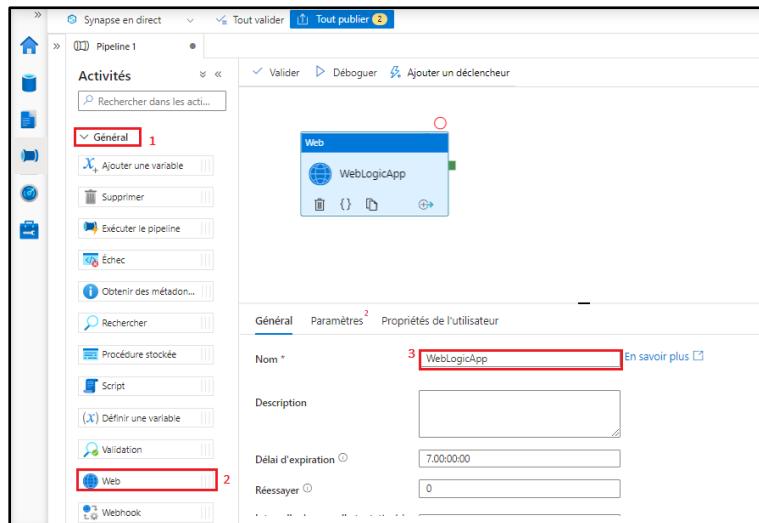
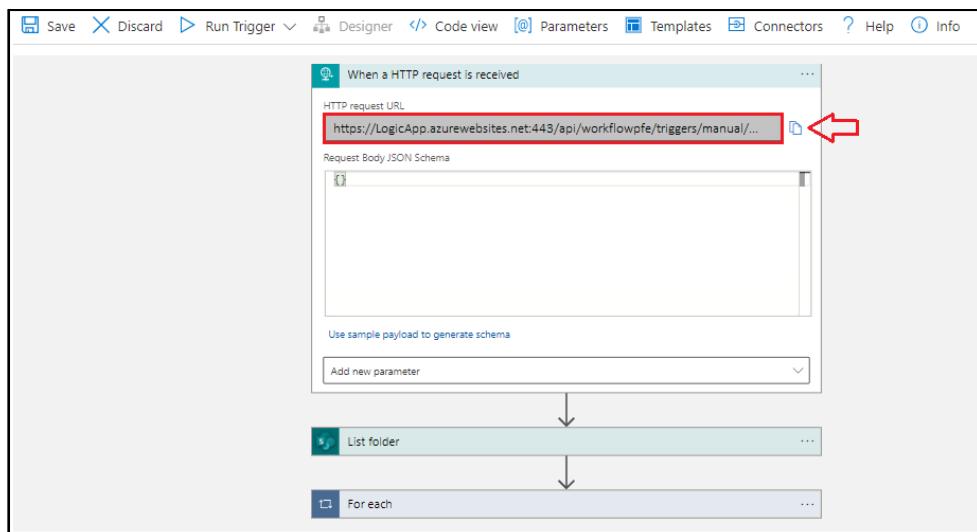


Figure 47 : Ajout d'un pipeline (Partie 1)

Dans les activités, on clique sur “Général” et on choisit “Web”. Dans les paramètres de l'activité Web, on le nomme “WebLogicApp” comme le montre la figure 48.

**Figure 48 : Ajout d'un pipeline (Partie 2)**

On retourne vers le Logic app designer et on copie URL généré par la requête HTTP puis on copie le lien dans le champ URL des paramètres de “WebLogicApp”. On choisit “POST” comme méthode puis dans le champ Corps, on écrit la commande suivant “@concat('{},')” comme le montrent les figure 49 et 50.

**Figure 49 : Logic App (Partie 8)**

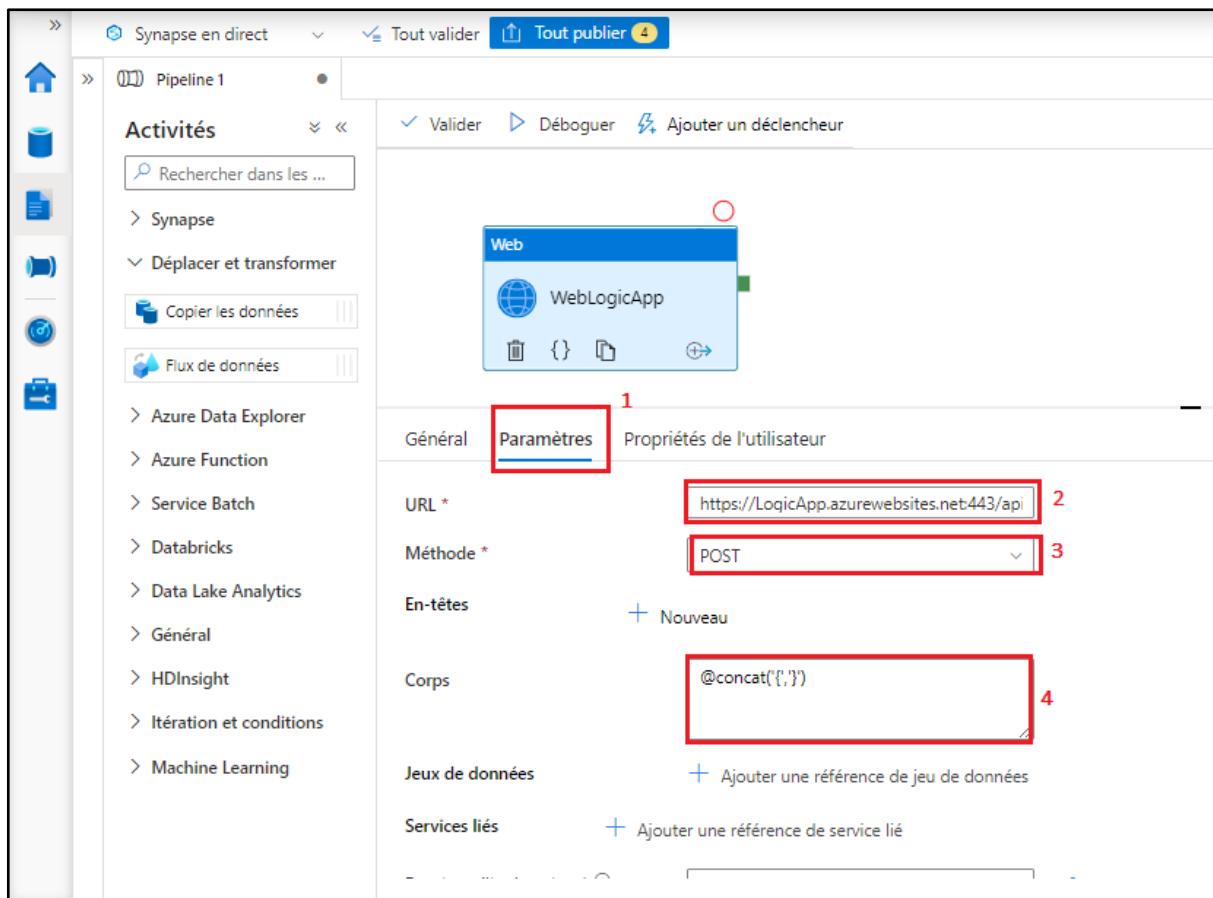


Figure 50 : Ajout d'un pipeline (Partie 2)

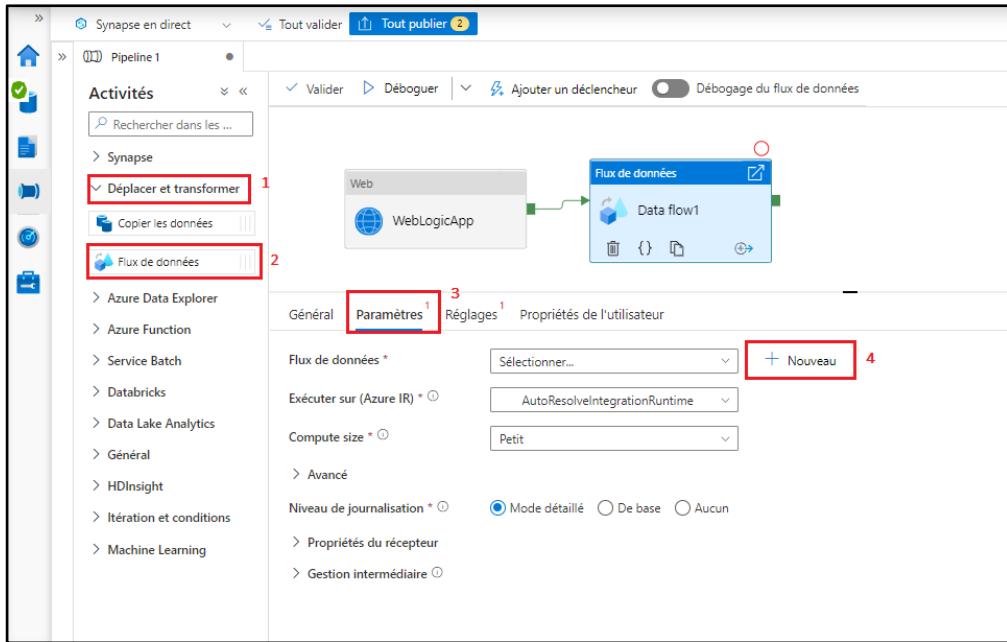
3. Transformations

3.1 Crédation du dataset

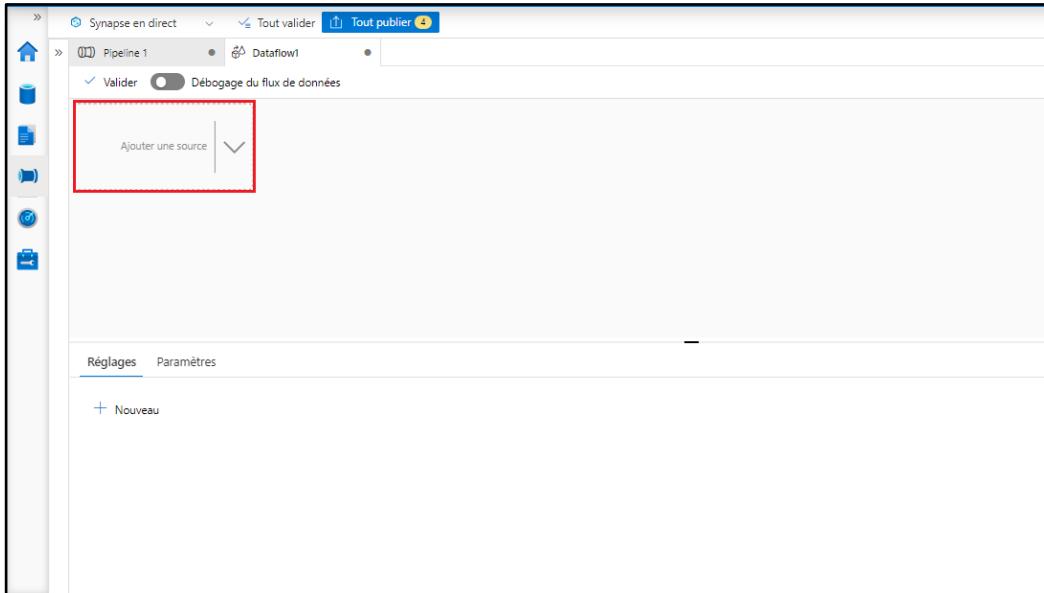
Pour la transformation des données, il faut tout d'abord construire un flux de données/Dataflow. Les flux de données dans Azure Data Factory permettent de réaliser les transformations nécessaires sur nos données.

Pour cela, il faut suivre les étapes suivantes :

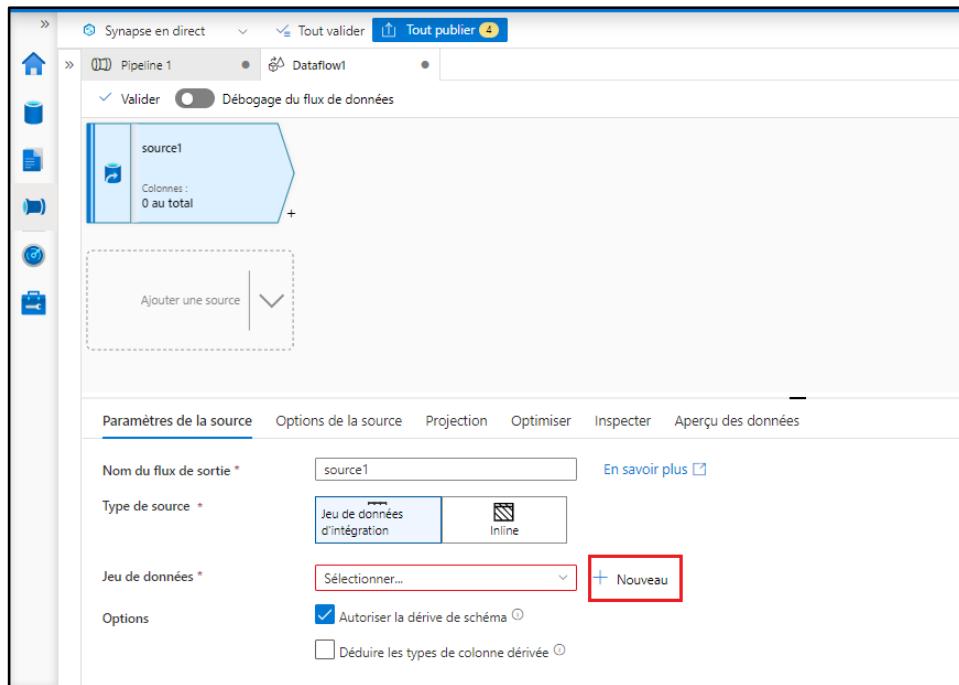
Dans les activités de notre pipeline, créée auparavant, on sélectionne "Flux de données" et on le lie à « WebLogicApp ». Dans les paramètres, on clique sur nouveau. (Figure 51)

**Figure 51 : Ajout d'un dataflow**

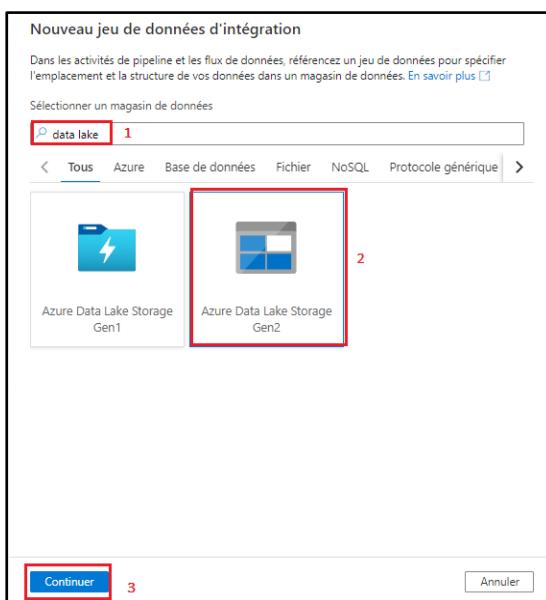
Dans la nouvelle fenêtre ouverte, on clique sur “ajouter une source”. (Figure 52)

**Figure 52: Ajout de la source**

Dans les paramètres de la source, on clique sur Nouveau pour ajouter un jeu de données et donc le fichier extrait de Teams et stocké dans “projetcfesys”. (Figure 53)

**Figure 53 : Création du jeu de données (Partie 1)**

Dans la barre de recherche, on tape “data Lake”, on sélectionne “Azure Data Lake storage Gen2” puis on clique sur “Continuer”. (Figure 54)

**Figure 54 : Création du jeu de données (Partie 2)**

On sélectionne le format Excel. (Figure 55)

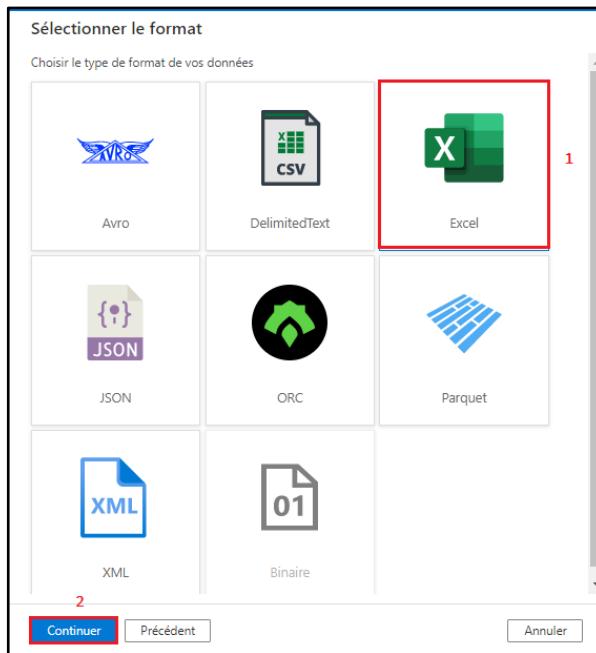


Figure 55 : Crédit du jeu de données (Partie 3)

On nomme notre jeu d'intégration Dataset1. On lie ce dernier à notre espace de travail. Le chemin d'accès va correspondre à "projetpfesys/Directory/ Reporting recouvrement RFC_Compusell (1).xlsx". On sélectionne ensuite "Encours Clt RFC" comme nom de la feuille dont on a besoin et on coche sur "première ligne comme en-tête" puis on clique sur "OK". (Figure 56)

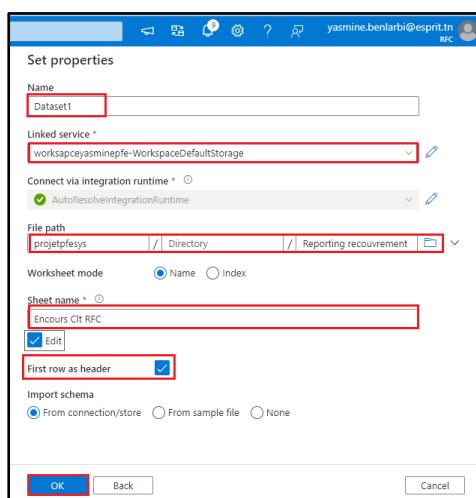


Figure 56 : Crédit du jeu de données (Partie 4)

Voici notre dataset (Figure 57)

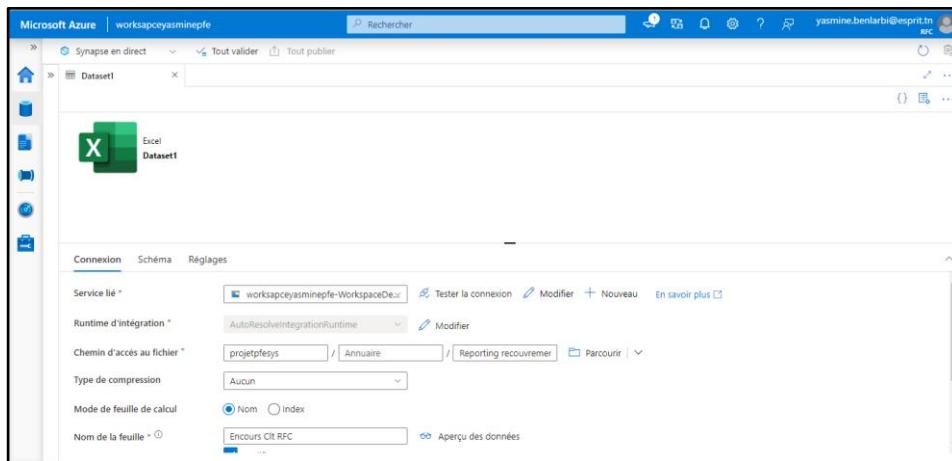


Figure 57 : Jeu de données : Dataset1

3.2 Ajout des Id

3.2.1 Id_client

Pour insérer un identifiant unique à chaque client il faut suivre les étapes suivantes :

Nous devons sélectionner la colonne Client. Pour cela, on clique sur l'icône '+' à côté de la source et dans le modificateur de schéma, on clique sur "sélectionner". (Figure 58)

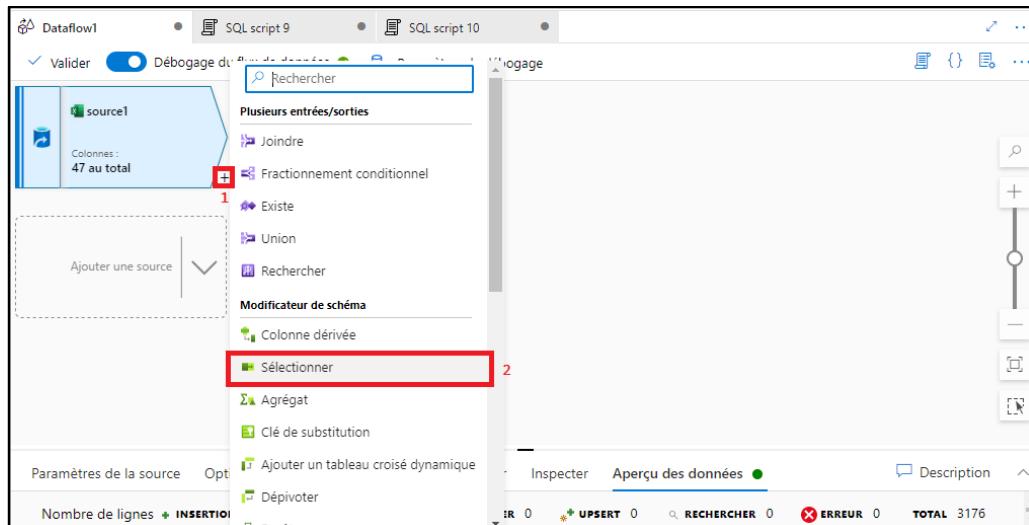


Figure 58 : Select Client (Partie 1)

Dans les paramètres, on met “Selectclient” comme un nom du flux de sortie et “source1” comme flux entrant. Dans la partie de mappage, on sélectionne seulement Client pour n'afficher que la colonne Client. (Figure 59)

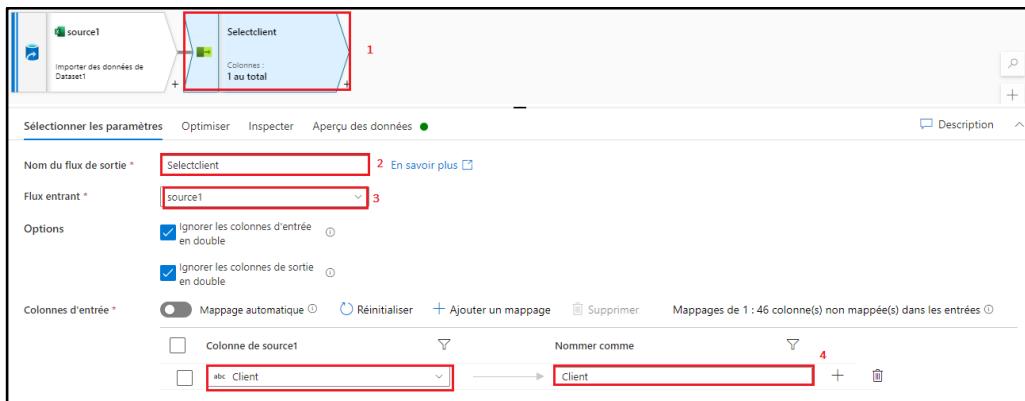


Figure 59: Select Client (Partie 2)

On clique sur l'icône ‘+’ et on clique sur “Agrégat”.(Figure 60)



Figure 60 : Agrégation Client

Dans les paramètres d'agrégation, on met “aggregate1” comme un nom du flux de sortie, “Selectclient” comme flux entrant et on sélectionne la colonne Client dans “grouper par” pour grouper les clients par leur nom. (Figure 61)

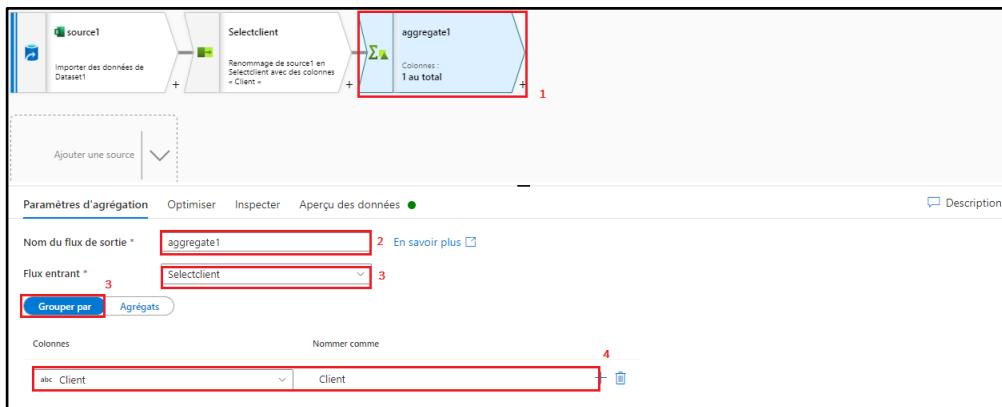


Figure 61 : Agrégation Client (Grouper par)

Dans “agrégats”, on ajoute un modèle de colonne et on met “name==’client’” dans le champ “Chaque colonne correspondant à” avec comme conditions “\$\$” pour la case à gauche et “first (\$\$)” pour la case à droite pour faire la distinction entre les différents clients et accorder un identifiant unique à chaque client même si le nom du client se répète plusieurs fois dans la colonne. (Figure 62)

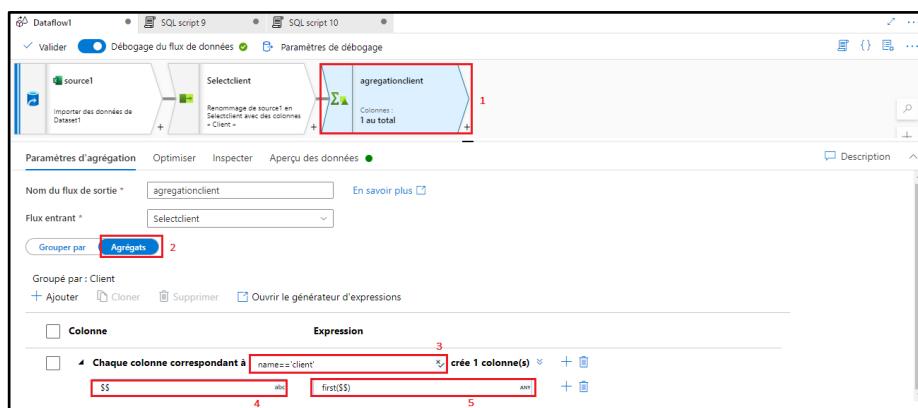
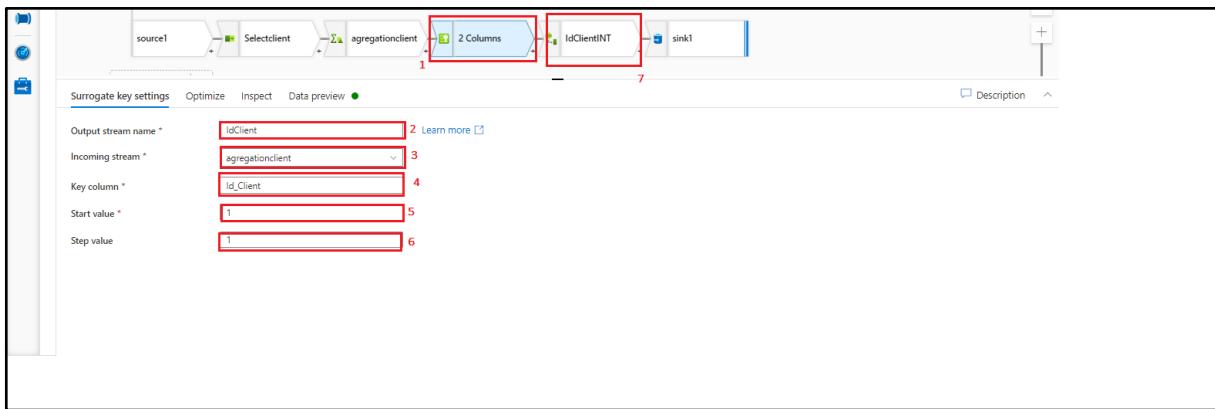


Figure 62: Agrégation Client (Agrégats)

Finalement, pour ajouter un identifiant à chaque client, on ajoute une clé de substitution. On nomme cette clé de substitution “IdClient” et on choisit “Id_Client” comme colonne clé avec “1” comme valeur de départ et “1” comme valeur de pas. On ajoute après une colonne dérivée pour modifier le type des identifiants à INT. Chaque client aura, après ces étapes, un identifiant propre à lui. (Figure 63)

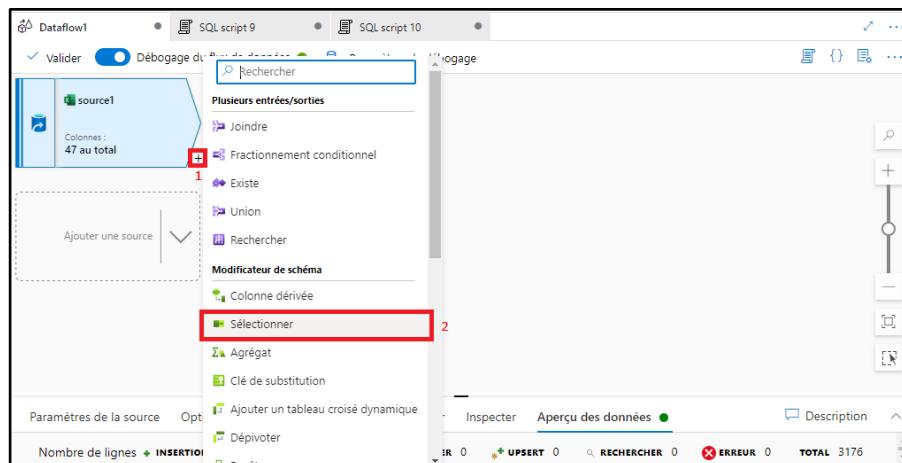
**Figure 63 : Ajout de l'id client**

3.2.2 Id_Société

RFC émet des factures sous deux noms ; RFC, qui offre des services de conseil et d'accompagnement, et Compusell, qui est directement affiliée à RFC et qui vend du matériel informatique.

Pour insérer un identifiant unique à chaque société, il faut suivre les étapes suivantes :

Nous devons sélectionner la colonne “Societe”. Pour cela, on clique sur l'icône ‘+’ à côté de la source et dans le modificateur de schéma, on clique sur “sélectionner”.(Figure 64)

**Figure 64: Select Société (Partie 1)**

Dans les paramètres, on met “Selectsociete” comme un nom du flux de sortie et “source1” comme flux entrant. Dans la partie de mappage, on sélectionne seulement “Societe” pour n'afficher que la colonne “Societe”.(Figure 65)

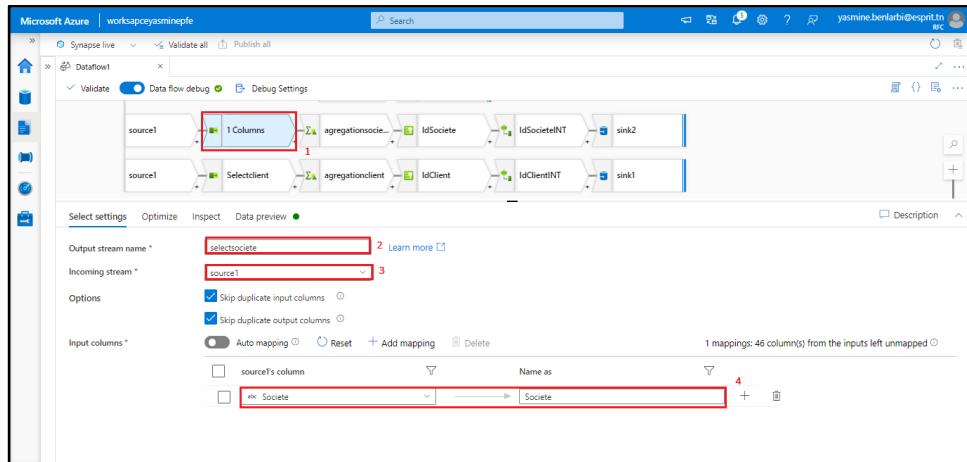


Figure 65: Select Société (Partie 2)

On clique sur l'icône ‘+’ et on clique sur “Agrégat”. Dans les paramètres d'agrégation, on met “aggregationsociete” comme un nom du flux de sortie, “Selectsociete” comme flux entrant et on sélectionne la colonne “Societe” dans “grouper par” pour grouper les sociétés par leur nom. (Figure 66)



Figure 66: Agrégation société (Grouper par)

Dans “agrégats”, on ajoute un modèle de colonne et on met “ name==‘societe’; ” dans le champ “Chaque colonne correspondant à” avec comme conditions “\$\$” pour la case à gauche et “first (\$\$)” pour la case à droite pour faire la distinction entre les deux sociétés accorder un

identifiant unique à chaque société même si le nom de la société se répète plusieurs fois dans la colonne. (Figure 67)

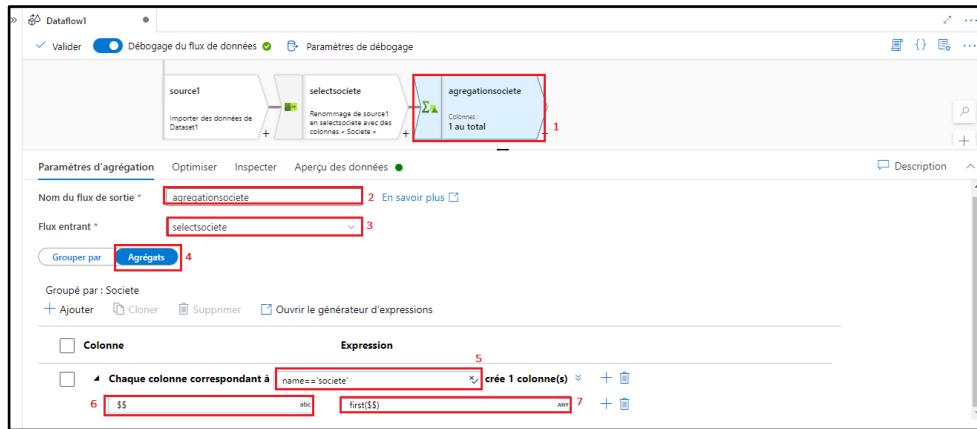


Figure 67: Agrégation société (Agrégats)

Finalement, pour ajouter un identifiant à chaque société, on ajoute une clé de substitution. On nomme cette clé de substitution “IdSociete” et on choisit “Id_Société” comme colonne clé avec “1” comme valeur de départ et “1” comme valeur de pas. On ajoute après une colonne dérivée pour modifier le type des identifiants à INT. Après cette étape, RFC aura l'identifiant numéro 1 et Compusell aura l'identifiant numéro 2. (Figure 68)

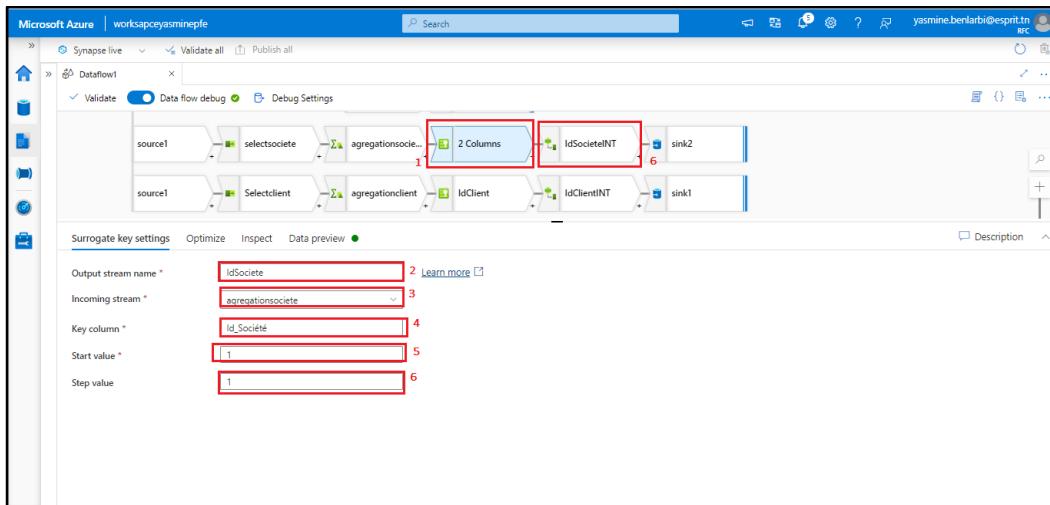


Figure 68: Ajout de l'id société

3.2.3 Id_Paiement

Pour insérer un identifiant unique à chaque paiement, il faut suivre les étapes suivantes :

On clique sur clé de substitution après le fichier source. On nomme cette clé de substitution “Id_Paiement” et on choisit “Id_paiement” comme colonne clé avec “1” comme valeur de départ et “1” comme valeur de pas. On ajoute après une colonne dérivée pour modifier le type des identifiants à INT. Chaque paiement aura un identifiant qui ira de 1 à 3176 puisqu'il existe 3176 factures dans le fichier source. (Figure 69 et 70)

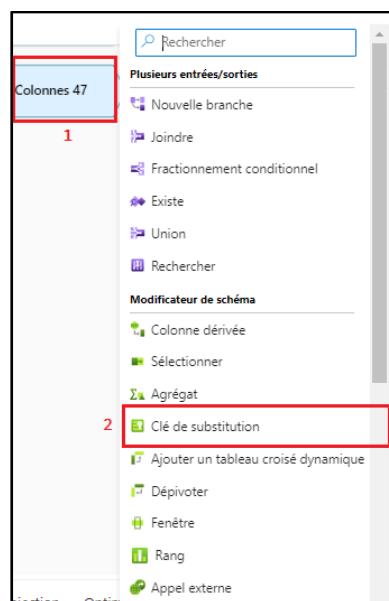


Figure 69: Ajout de l'id paiement (Partie 1)

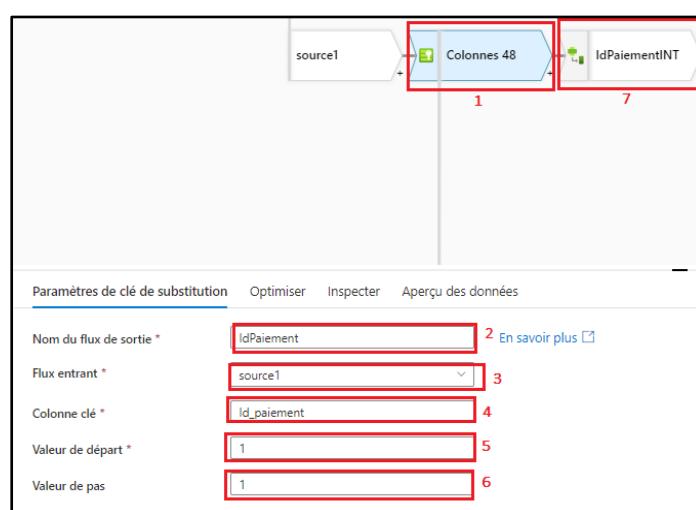


Figure 70: Ajout de l'id paiement (Partie 2)

3.2.4 Id_Date

Chaque facture s'inscrit dans le cadre d'un objectif. Les factures qui entrent dans l'objectif non encaissé n'ont pas de date de paiement car elles n'ont pas encore été encaissées ce qui veut dire que ces factures sont prises en compte grâce à la « date objectif » qui est le mois et l'année de la date d'échéance. Les factures qui entrent dans l'objectif encaissé et le non objectif encaissé ont une date de paiement et une « date objectif » mais ces factures sont comptabilisées grâce à la date à laquelle les factures ont été payées. Nous allons donc unir la datepaiement et la Date_Objectif et donner à chaque date un identifiant unique.

- **Date_Objectif**

Pour insérer un identifiant unique à chaque Date Objectif, il faut suivre les étapes suivantes :

Nous devons sélectionner les colonnes Date_Objectif, Année et Mois pour n'afficher que ces colonnes. Pour cela, on clique sur l'icône ‘+’ à côté de la source et dans le modificateur de schéma, on clique sur “sélectionner”. Dans les paramètres, on met “Selectdateobjectif” comme un nom du flux de sortie et “DateObjectif” comme flux entrant. Dans la partie de mappage, on sélectionne seulement Date_Objectif, Année et Mois. (Figure 71)

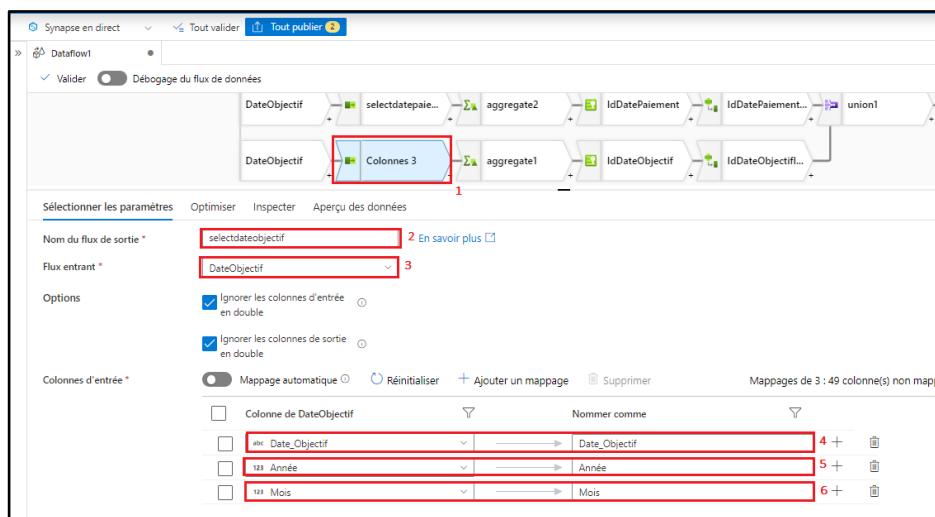


Figure 71 : Select Date Objectif

On clique sur l'icône ‘+’ et on clique sur “Agrégat”. Dans les paramètres d'agrégation, on met “aggregate1” comme un nom du flux de sortie, “Selectdateobjectif” comme flux entrant et

on sélectionne les colonnes Date_Objectif, Année et Mois dans "grouper par" pour grouper par date, année et mois. (Figure 72)

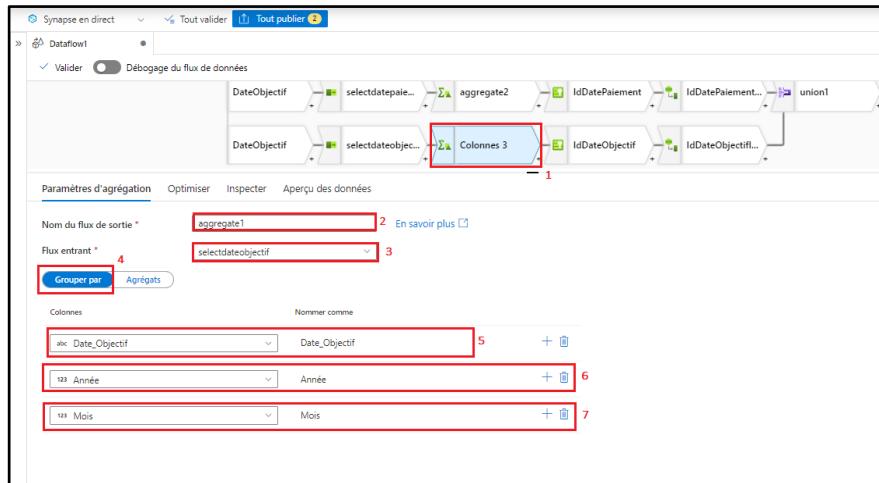


Figure 72 : Agrégation Date Objectif (Grouper par)

Dans "agrégats", on ajoute un modèle de colonne et on met "name=='date_objectif'" dans le champ "Chaque colonne correspondant à" avec comme conditions "\$\$" pour la case à gauche et "first (\$\$)" pour la case à droite pour faire la distinction entre les différentes dates et accorder un identifiant unique à chaque date même si la date se répète plusieurs fois dans la colonne. (Figure 73)

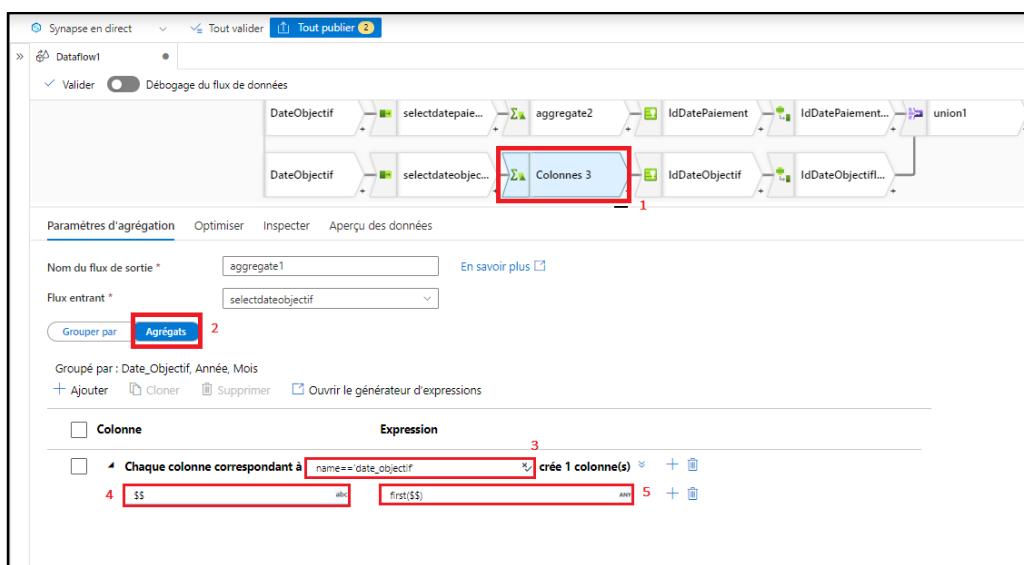


Figure 73: Agrégation Date Objectif (Agrégats)

Finalement, pour ajouter un identifiant à chaque date, on ajoute une clé de substitution. On nomme cette clé de substitution “IdDateObjectif” et on choisit “id_date_objectif” comme colonne clé avec “1” comme valeur de départ et “1” comme valeur de pas. On ajoute après une colonne dérivée pour modifier le type des identifiants à INT. Chaque date aura, après ces étapes, un identifiant propre à elle. (Figure 74)

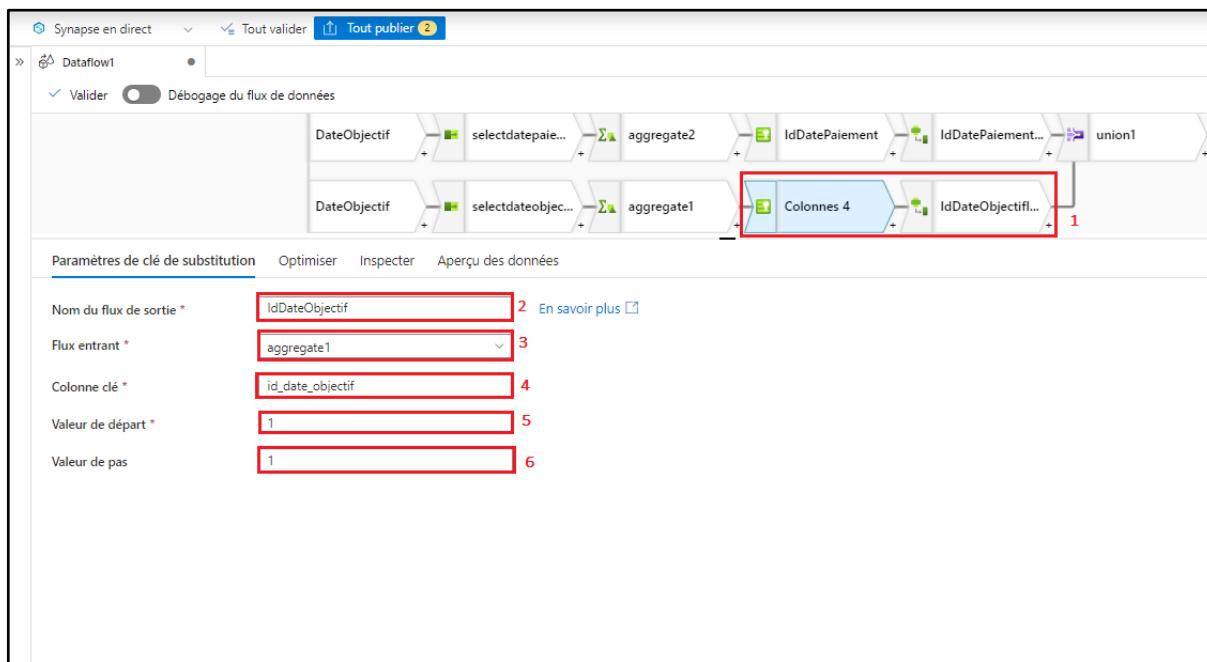


Figure 74: Ajout de l'id Date Objectif

● Datepaiement

Pour insérer un identifiant unique à chaque Datepaiement, il faut suivre les étapes suivantes :

Nous devons sélectionner les colonnes datepaiement, Année de paiement réelle et Mois de paiement réelle pour n'afficher que ces colonnes. Pour cela, on clique sur l'icône '+' à côté de la source et dans le modificateur de schéma, on clique sur "sélectionner". Dans les paramètres, on met "Selectdatepaiement" comme un nom du flux de sortie et "DateObjectif" comme flux entrant. Dans la partie de mappage, on sélectionne Année de paiement réelle et Mois de paiement réelle.(Figure 75)

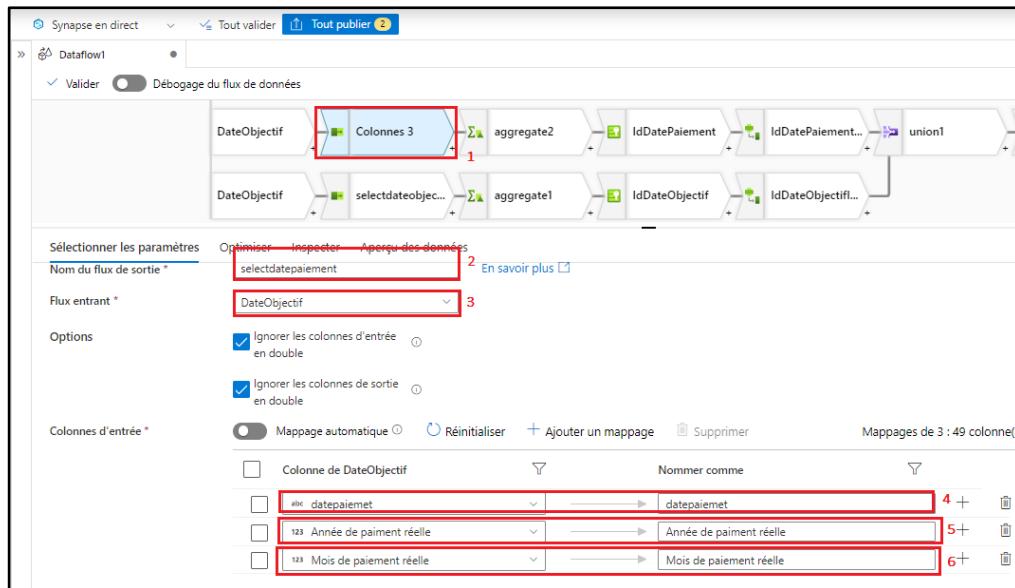


Figure 75: Select Date paiement

On clique sur l'icône ‘+’ et on clique sur “Agrégat”. Dans les paramètres d'agrégation, on met “aggregate2” comme un nom du flux de sortie, “Selectdatepaiement” comme flux entrant. et on sélectionne les colonne Année de paiement réelle et Mois de paiement réelle dans “grouper par” pour grouper par date, année et mois. (Figure 76)

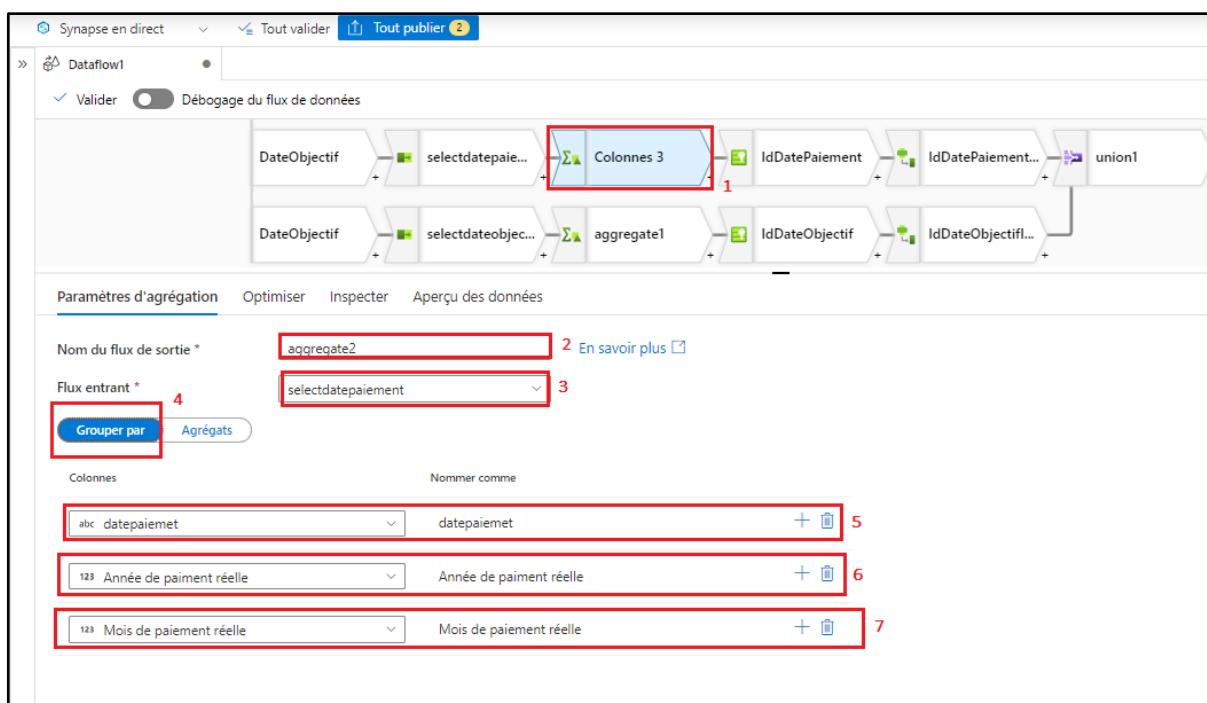


Figure 76: Agrégation Date paiement (Grouper par)

Dans "agrégats", on ajoute un modèle de colonne et on met "name=='date_paiement';" dans le champ "Chaque colonne correspondant à" avec comme conditions " \$\$" pour la case à gauche et "first (\$\$)" pour la case à droite pour faire la distinction entre les différentes dates et accorder un identifiant unique à chaque date même si la date se répète plusieurs fois dans la colonne. (Figure 77)

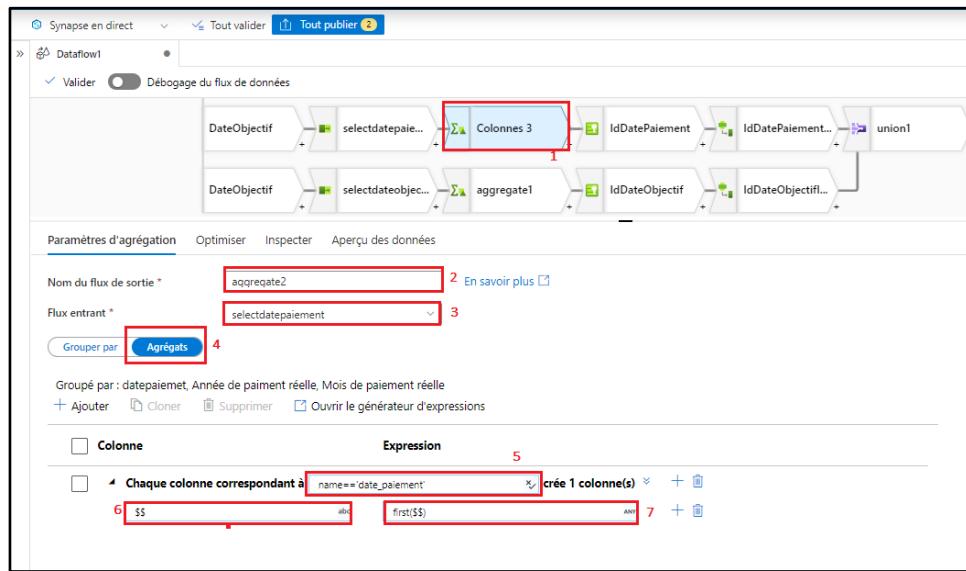


Figure 77 : Agrégation Date paiement (Agrégats)

Finalement, pour ajouter un identifiant à chaque date, on ajoute une clé de substitution. On nomme cette clé de substitution "IdDatePaiement" et on choisit "id_date_paiement" comme colonne clé avec "1" comme valeur de départ et "60" comme valeur de pas car l'id de la date objectif s'est arrêté à 59. On ajoute après une colonne dérivée pour modifier le type des identifiants à INT. Chaque date aura, après ces étapes, un identifiant propre à elle. (Figure 78)

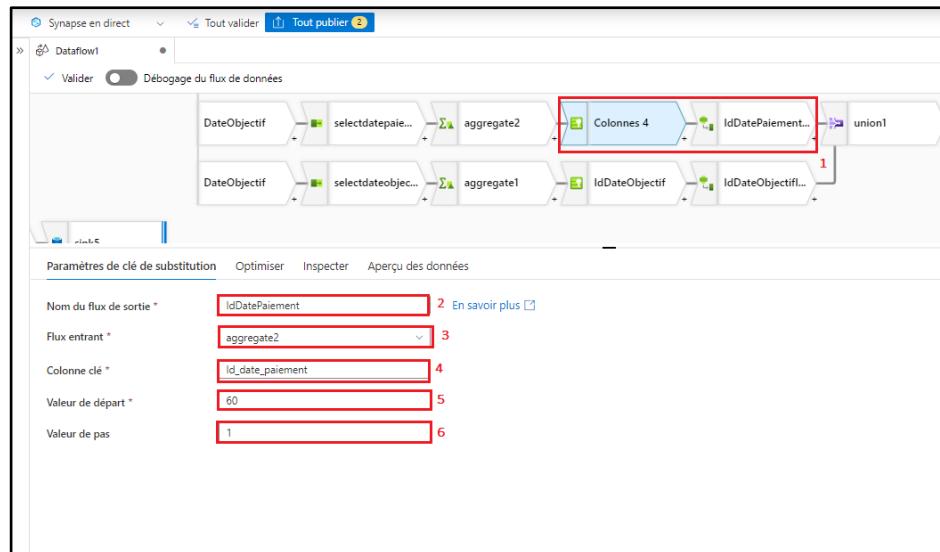


Figure 78 : Ajout de l'id Date paiement

Nous allons ensuite réunir les deux dates dans un seul et unique tableau pour pouvoir ensuite charger les dates dans la dimension date. (Figure 79)

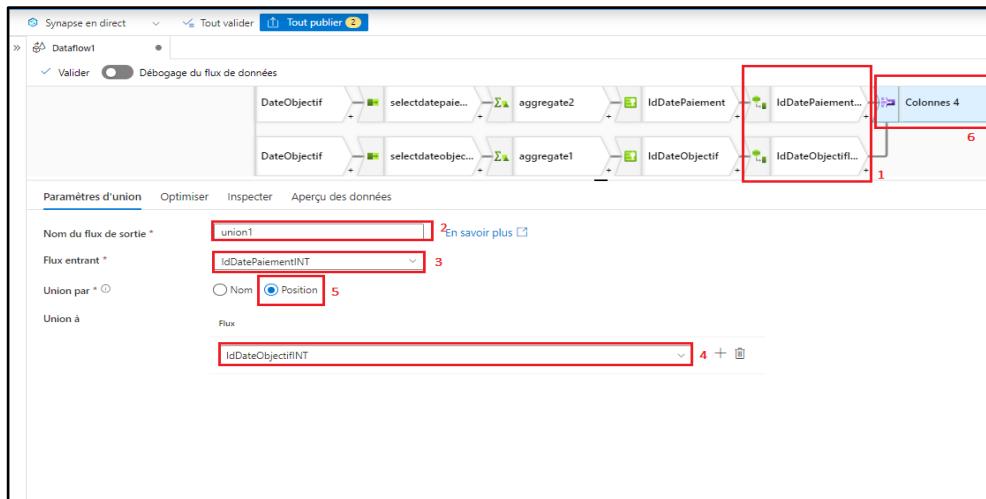


Figure 79 : Union des id des deux dates

3.3 Calcul des objectifs

À chaque facture correspond un objectif. Pour calculer nos objectifs encaissés et non-encaissés et non-objectifs encaissés, il faut tout d'abord ajouter une colonne pour la « date objectif » qui correspond au mois et à l'année à laquelle chaque facture devrait normalement être comptabilisée et ensuite partitionner les objectifs sur nos deux sociétés (RFC et Compusell). Pour cela, nous allons suivre les étapes suivantes :

Après la clé de substitution qui désigne l'identifiant de chaque paiement, on ajoute des colonnes dérivées sous le nom du flux “DateObjectif” avec les conditions suivantes (Figure 80):

Colonne	Expression	Explication
Année	year({Échéance Contractuelle})	Nous allons prendre l'année de la date d'échéance.
Mois	month({Échéance Contractuelle})	Nous allons prendre le mois de la date d'échéance.
Date_Objectif	concat(toString(year({Échéance Contractuelle})), "-" ,(toString(month({Échéance Contractuelle}))))	Nous allons concaténer le mois et l'année en une seule date “Date_Objectif”. La « date objectif » permet de comptabiliser les factures qui font référence aux objectifs non encaissés du mois puisque les objectifs encaissés n'ont pas de date de paiement.
datepaiemet	concat(toString({Année de paiement réelle}), "-",(toString({Mois de paiement réelle})))	Nous allons concaténer le mois de paiement réelle et l'année de paiement réelle en une seule date “datepaiement”. La datepaiement permet de prendre en compte les factures qui entrent dans les objectifs encaissés du mois et les non objectifs encaissés du mois.

Tableau 13 : Ajout des colonnes

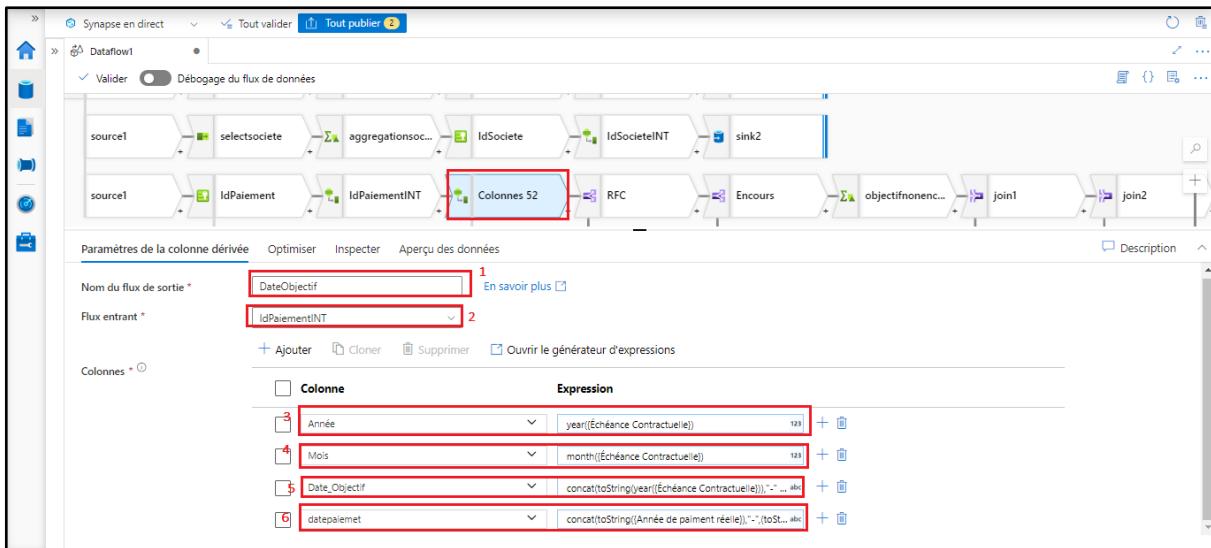
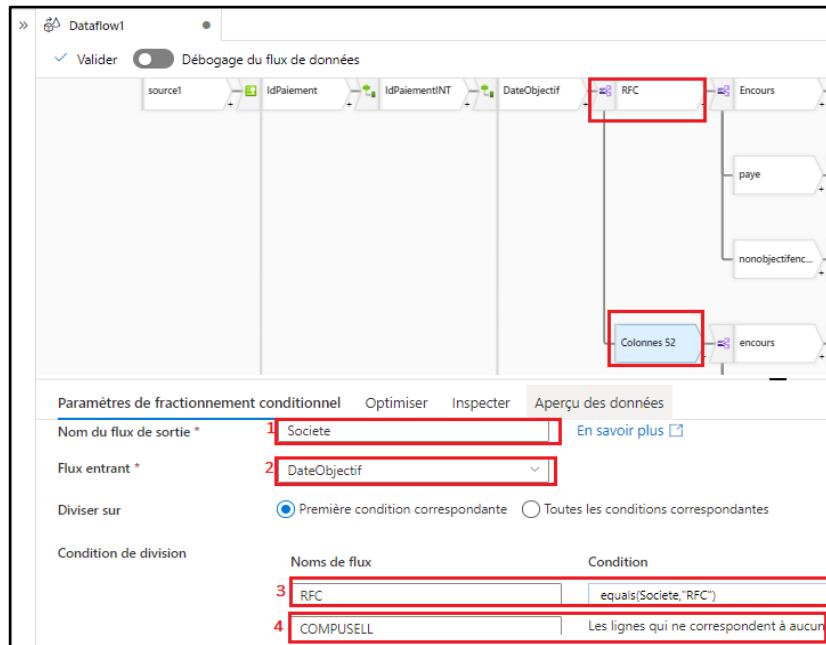


Figure 80 : Ajout des colonnes

RFC et Compusell peuvent avoir le même numéro de facture mais pas la même facture. Pour éviter des erreurs liées à la comptabilisation des factures, on ajoute un fractionnement conditionnel sous le nom Société avec les conditions suivantes pour partitionner RFC et Compusell (Figure 81) :

Noms de flux	Condition
RFC	equals(Societe,"RFC")
Compusell	Vide

Tableau 14 : Fractionnement conditionnel : Société

**Figure 81 : Fractionnement conditionnel : Société**

3.3.1 Flux RFC

Pour le flux de RFC, on ajoute un autre fractionnement conditionnel sous le nom “objectifrfc” avec les conditions suivantes pour répartir les objectifs sur trois comme l'a demandé le département Finance (Figure 82) :

Noms de flux	Condition	Explication
Encours	equals({Statut }, 'Encours')	Le flux encours aura les factures qui seront encore en cours de paiement.
paye	equals({Statut }, 'Payé')&&equals({Année de paiement réelle}, {Année})&&greaterOrEqual({Mois de paiement réelle}, Mois) equals({Statut }, 'Payé')&&greater({Année de paiement réelle}, {Année})	Ce flux correspond aux objectifs encaissés ce qui veut que ce flux aura les factures qui ont été payées à l'échéance ou après l'échéance (les impayés recouvrés).
nonobjectifencasse	Vide	Ce flux aura les factures restantes et donc les factures payées avant le mois de leur échéance.

Tableau 15 : Fractionnement des objectifs de RFC

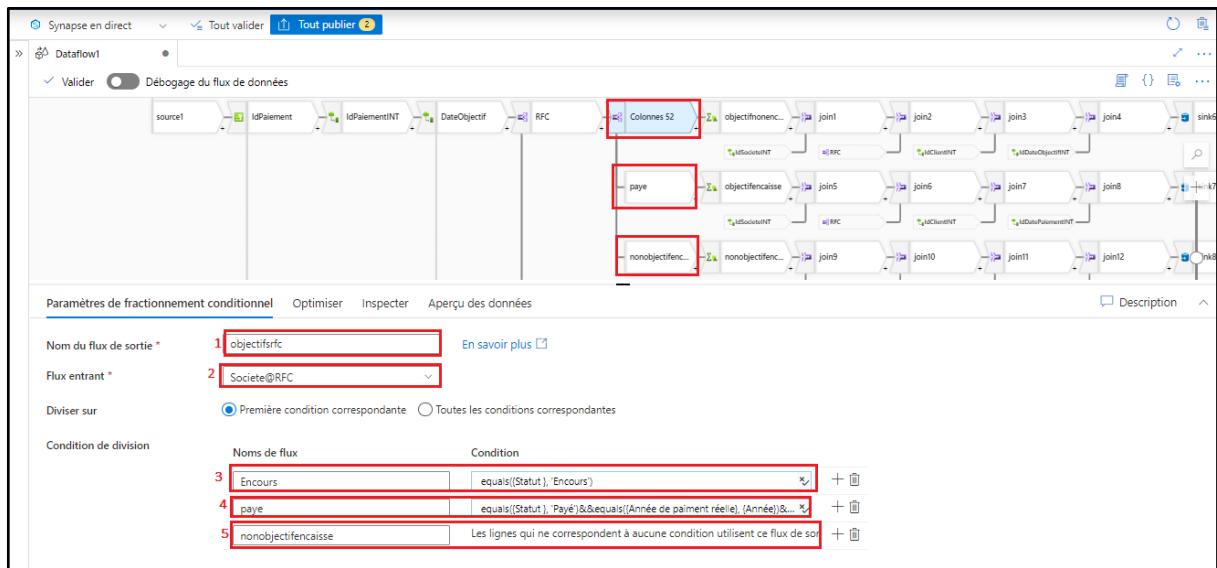


Figure 82: Fractionnement des objectifs de RFC

- Pour le flux encours, on ajoute une agrégation sous le nom “objectifnonencaisse” avec les conditions suivantes :
 - Grouper par pour afficher les colonnes suivantes (Figure 83) :

Colonnes	Nommer comme
Societe	Societe
N°Factures	N°Factures
Date_Objectif	Date_Objectif

Tableau 16 : L'agrégation des objectifs non-encaissés de RFC (Grouper par)

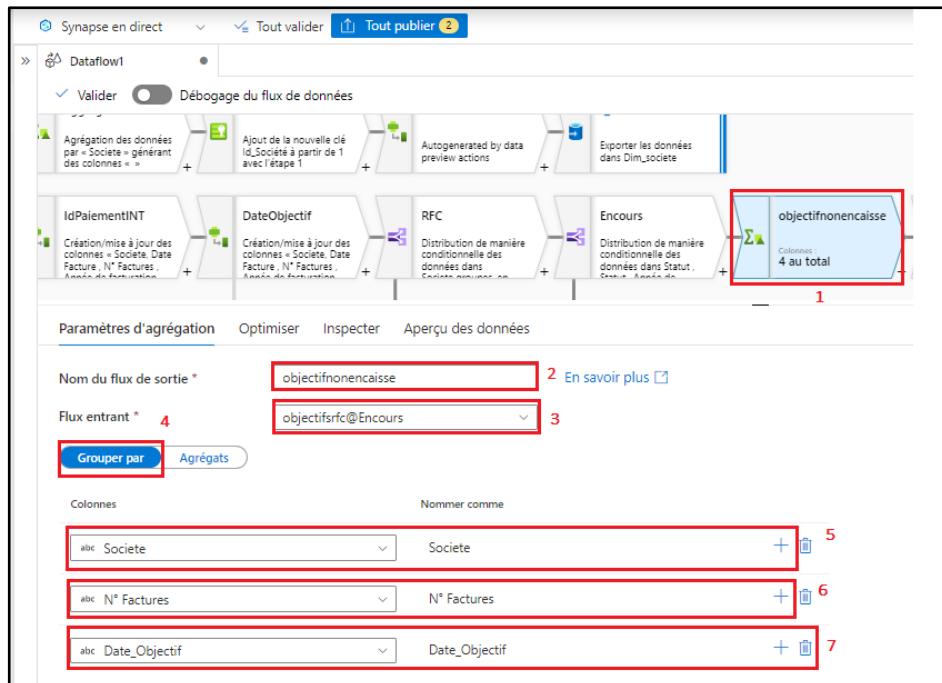


Figure 83 : L'agrégation des objectifs non-encaissés de RFC (Grouper par)

➤ Agrégats (Figure 84) :

Colonne	Conditions
objectifnonencaisse	sum({<Reste à recevoir>})

Tableau 17 : L'agrégation des objectifs non-encaissés de RFC (Agrégats)

Cette condition sert à prendre les montants qui n'ont pas encore été payés et de les mettre dans la colonne objectif non encaissé. Les montants relatifs aux objectifs seront répartis par facture.

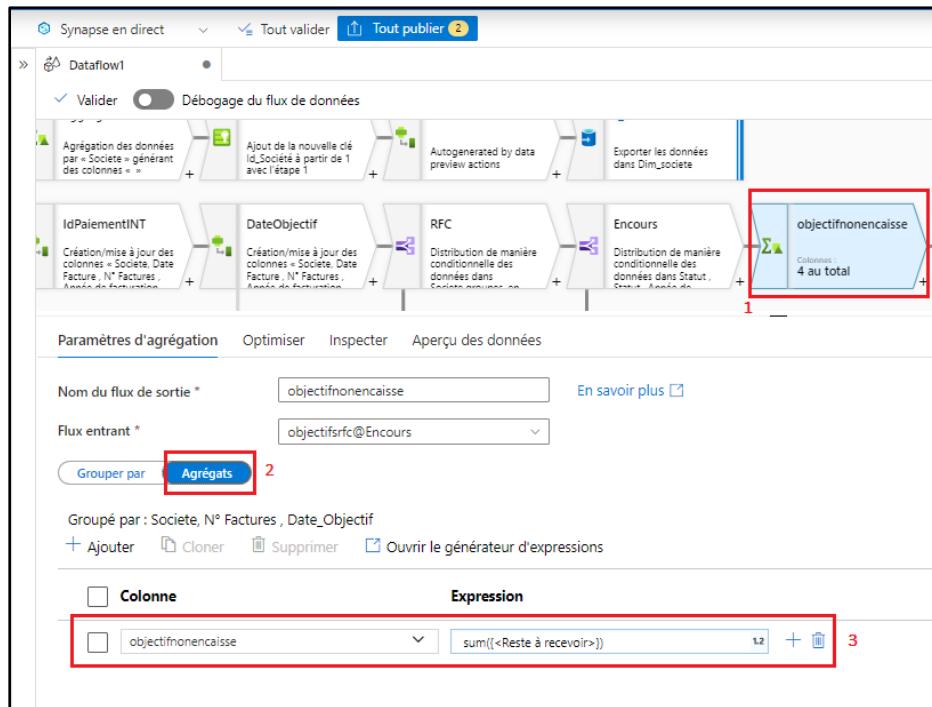


Figure 84 : L'agrégation des objectifs non-encaissés de RFC (Agrégats)

On va ensuite ajouter une jointure entre l'agrégation précédente et le flux "IdSocieteINT" pour ajouter à chaque facture l'id de RFC avec comme condition de jointure (Figure 85) :

Colonne de objectifnonencaisse		Colonne de IdSocieteINT
Societe	==	Societe

Tableau 18 : Condition de jointure 1

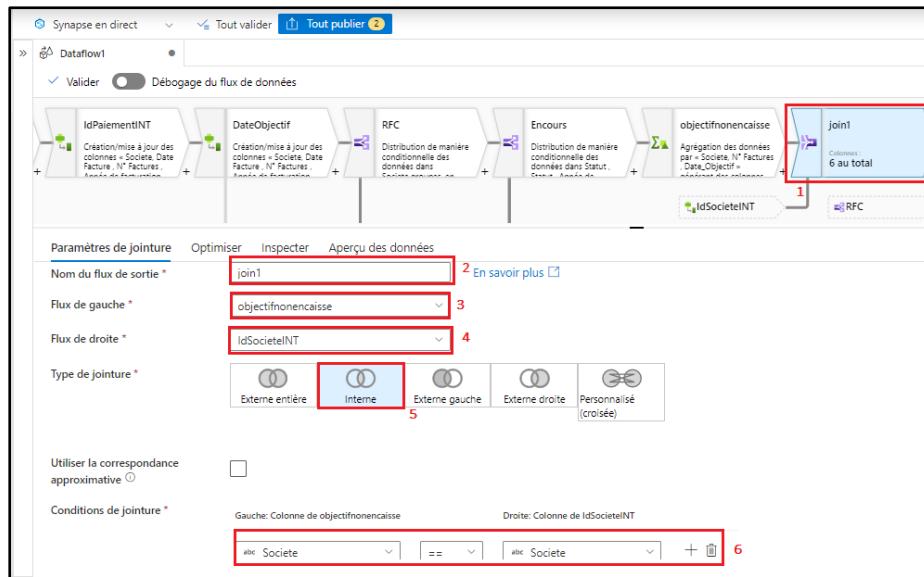


Figure 85 : Jointure Id Société

On va ensuite ajouter une jointure entre la jointure précédente et le flux Societe@RFC pour ajouter le numéro de facture de chaque facture correspondante à l'objectif non encaissé avec comme condition de jointure (Figure 86) :

Colonne de join1		Colonne de Societe@RFC
N°Factures	==	N°Factures

Tableau 19 : Condition de jointure 2

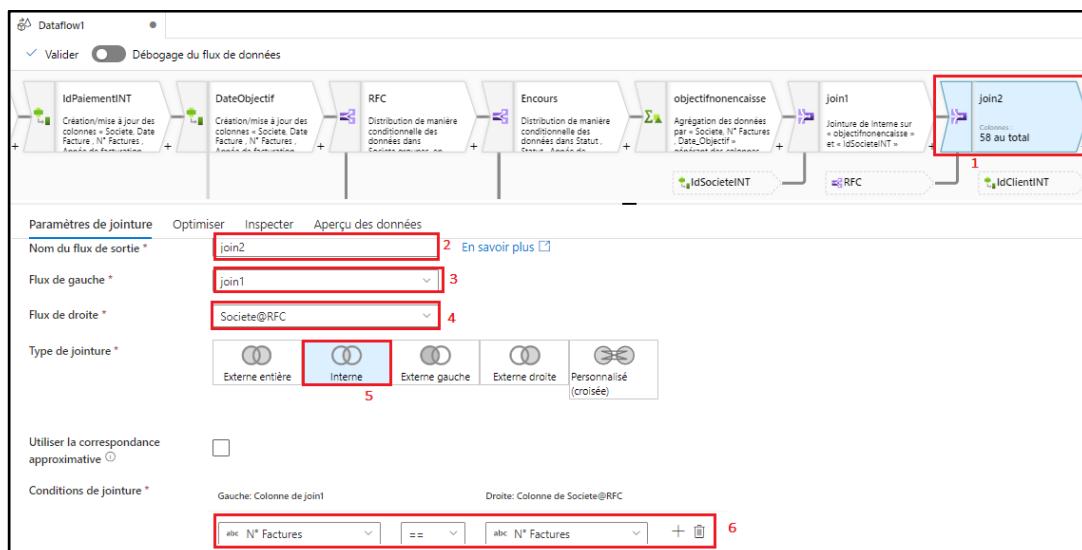


Figure 86 : Jointure numéro facture

On va ensuite ajouter une jointure entre la jointure précédente et le IdClientINT pour ajouter l'id du client de chaque facture avec comme condition de jointure (Figure 87) :

Colonne de join2		Colonne de IdClientINT
Client	==	Client

Tableau 20 : Condition de jointure 3

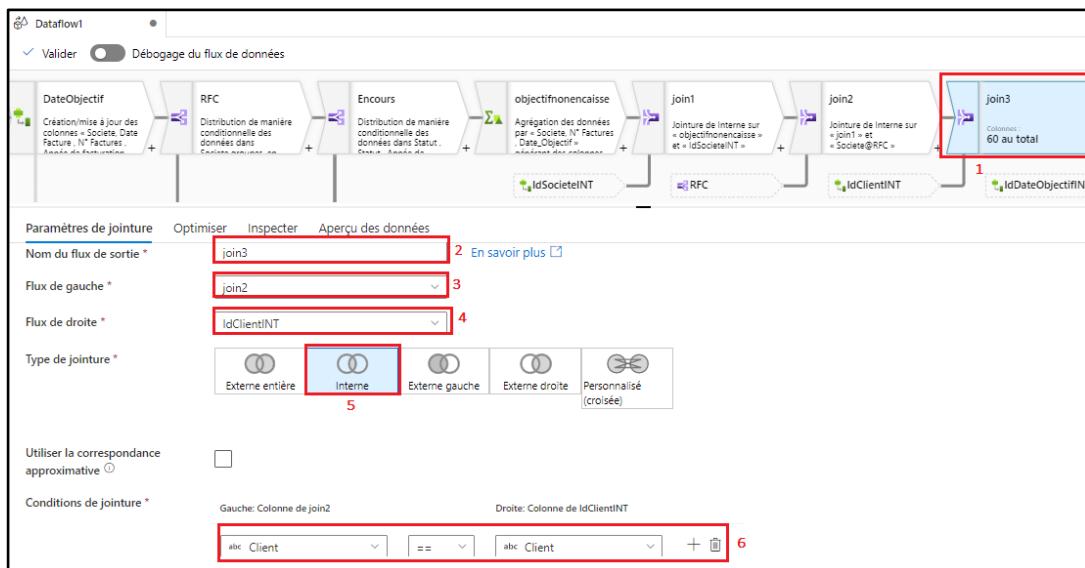


Figure 87 : Jointure id client

On va ensuite ajouter une jointure entre la jointure précédente et le IdDateObjectifINT pour ajouter l'id de la date à laquelle chaque facture correspondante à un objectif doit être comptabilisée avec comme condition de jointure (Figure 88) :

Colonne de join3		Colonne de IdDateObjectifINT
objectifnonencaisse@Date_Objectif	==	Date_Objectif

Tableau 21: Condition de jointure 4

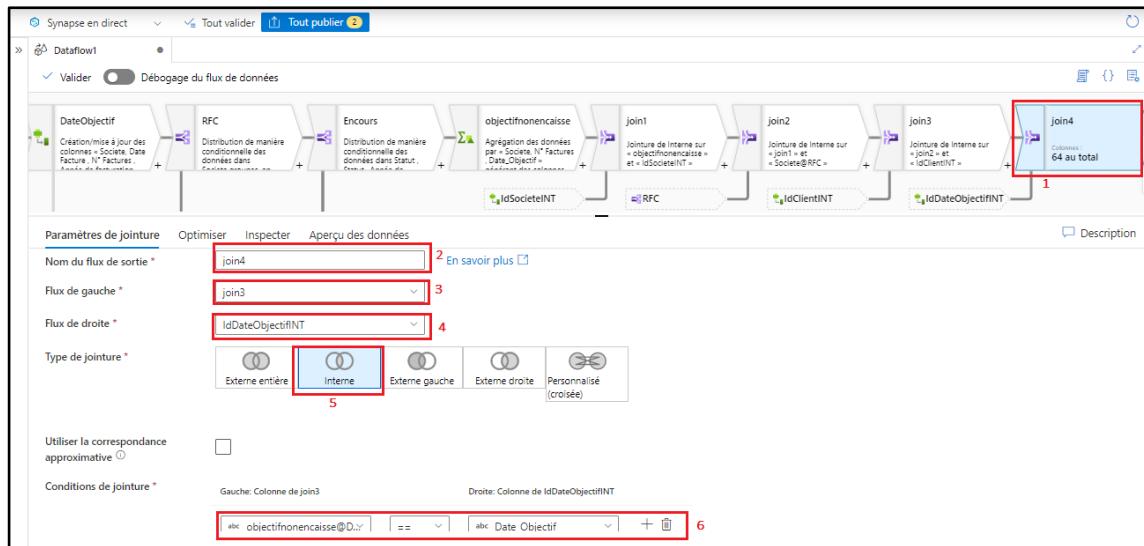
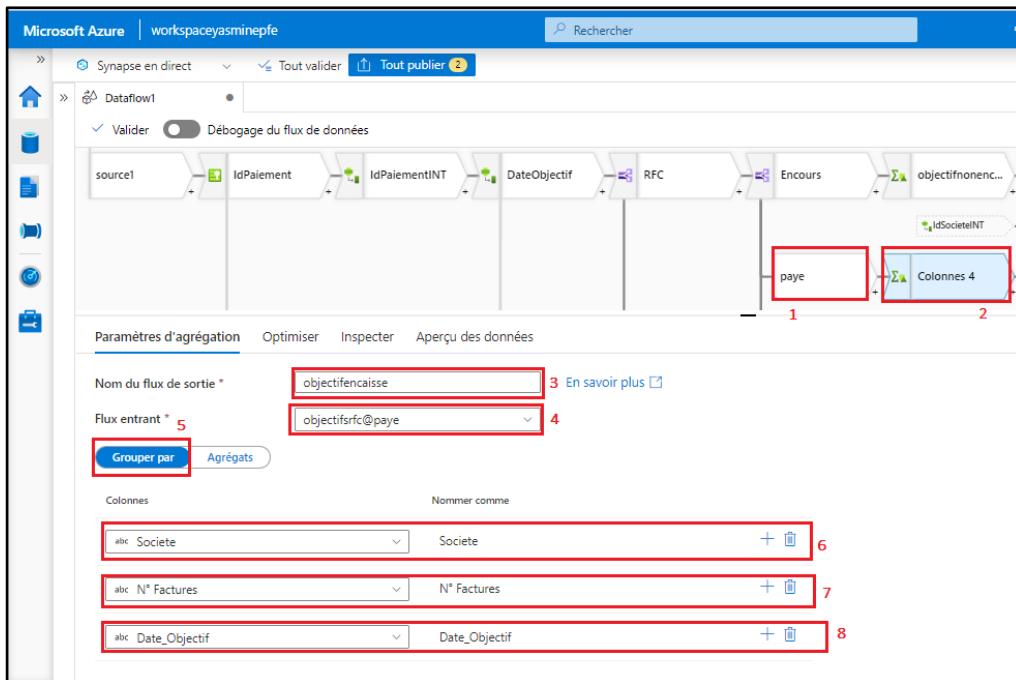


Figure 88 : Jointure id date objectif

- Pour le flux “paye”, on ajoute une agrégation sous le nom “objectifnonencaisse” avec les conditions suivantes :
- Grouper par pour afficher les colonnes suivantes (Figure 89) :

Colonnes	Nommer comme
Societe	Societe
N°Factures	N°Factures
Date_Objectif	Date_Objectif

Tableau 22 : L'agrégation des objectifs encaissés de RFC (Grouper par)

**Figure 89 : L'agrégation des objectifs encaissés de RFC (Grouper par)**

➤ Agrégats (Figure 90) :

Colonne	Conditions
objectifecaisse	sum({<Montant reçu>})

Tableau 23: L'agrégation des objectifs encaissés de RFC (Agrégats)

Cette condition sert à prendre les montants qui ont été payés, qui respectent la condition du flux "objectifecaisse" et de les mettre dans la colonne objectif encaissé. Chaque facture de ce flux aura un montant dans la colonne objectif encaissé.

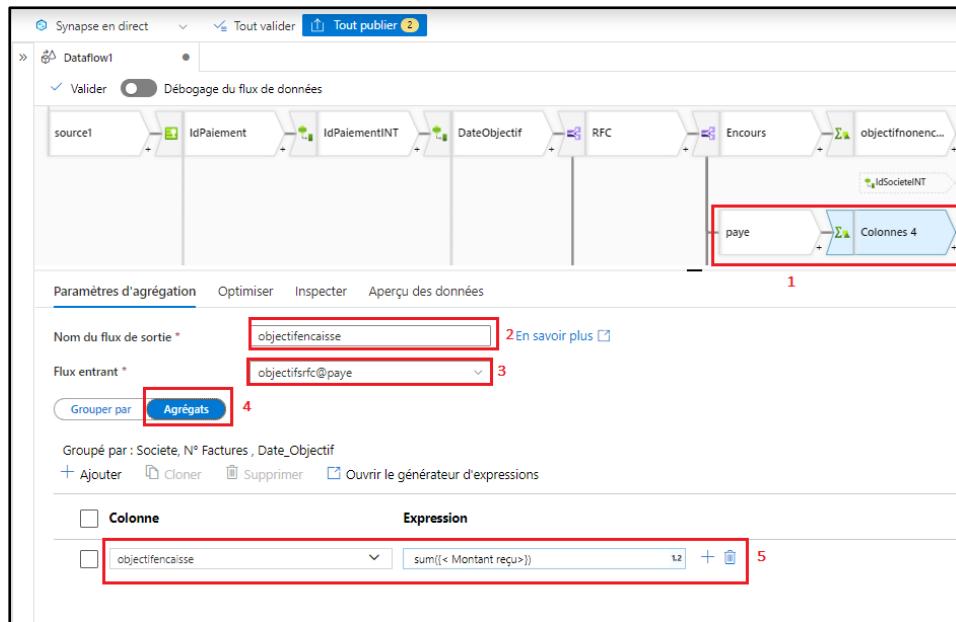
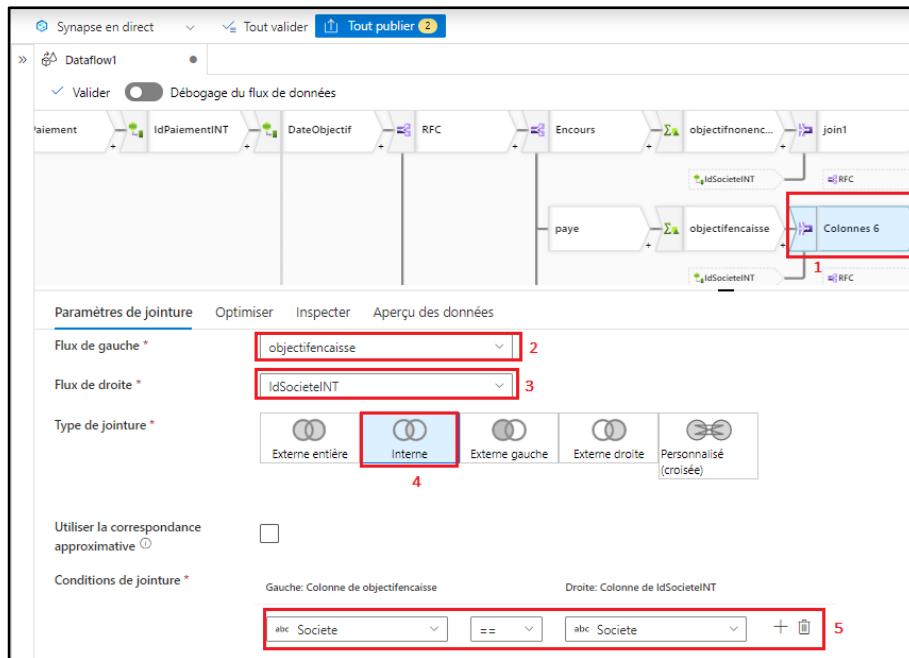


Figure 90 : L'agrégation des objectifs encaissés de RFC (Agrégats)

On va ensuite ajouter une jointure entre l'agrégation précédente et le flux “IdSocieteINT” pour ajouter à chaque facture correspondante à l'objectif encaissé l'id de RFC avec comme condition de jointure (Figure 91) :

Colonne de objectifencaisse		Colonne de IdSocieteINT
Societe	==	Societe

Tableau 24 : Condition de jointure 1

**Figure 91 : Jointure Id Société**

On va ensuite ajouter une jointure entre la jointure précédente et le flux Societe@RFC pour ajouter le numéro de facture de chaque facture correspondante à l'objectif non encaissé avec comme condition de jointure (Figure 92) :

Colonne de join5		Colonne de Societe@RFC
N°Factures	=	N°Factures

Tableau 25 : Condition de jointure 2

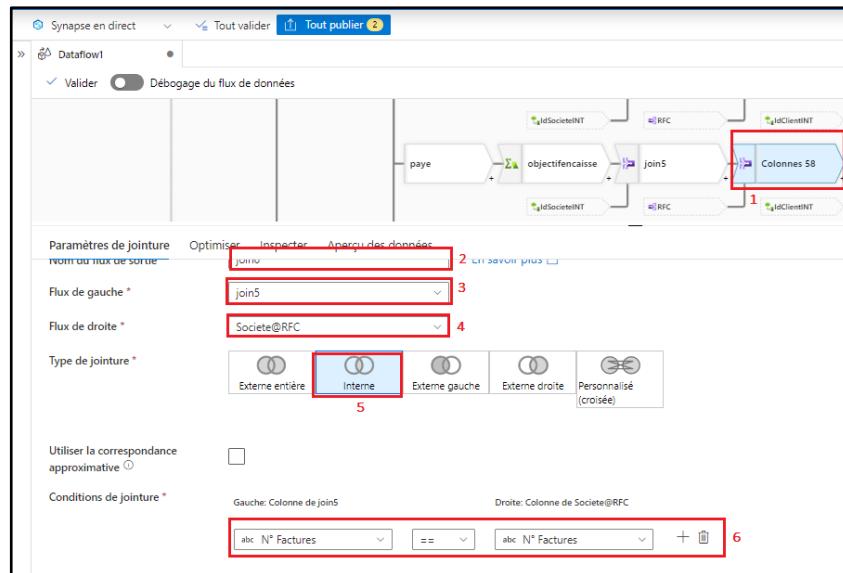


Figure 92 : Jointure numéro facture

On va ensuite ajouter une jointure entre la jointure précédente et le IdClientINT pour ajouter l'id du client de chaque facture avec comme condition de jointure (Figure 93) :

Colonne de join6		Colonne de IdClientINT
Client	==	Client

Tableau 26 : Condition de jointure 3

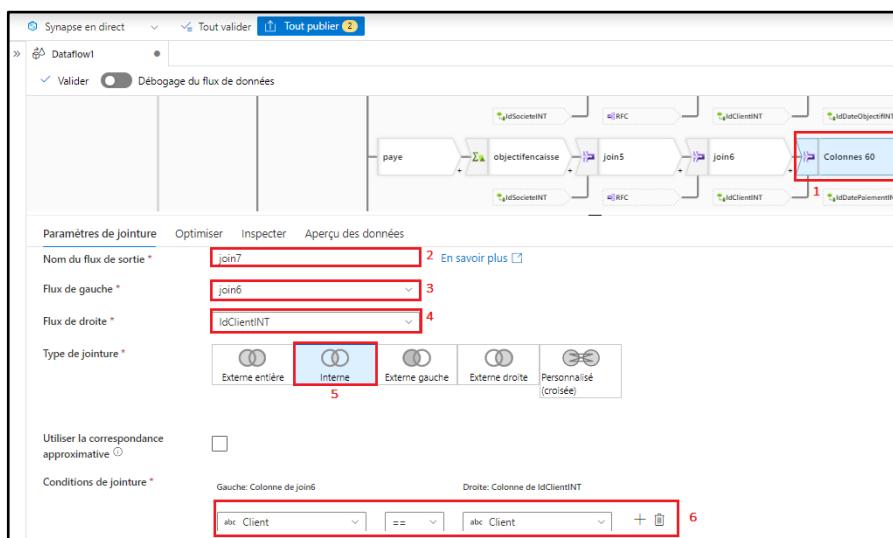


Figure 93 : Jointure id client

On va ensuite ajouter une jointure entre la jointure précédente et le IdDatePaiementINT pour ajouter l'id de la date à laquelle chaque facture correspondante à un objectif doit être comptabilisée avec comme condition de jointure (Figure 94) :

Colonne de join7		Colonne de IdDatePaiementINT
datepaiement	==	datepaiement

Tableau 27: Condition de jointure 4

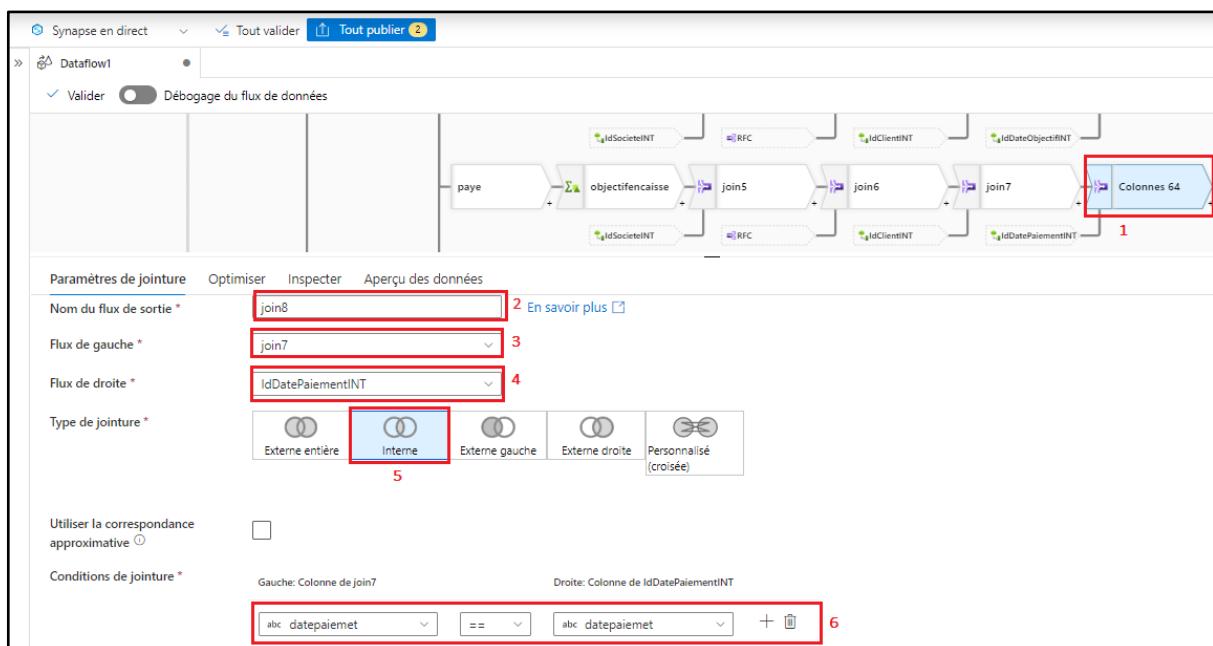


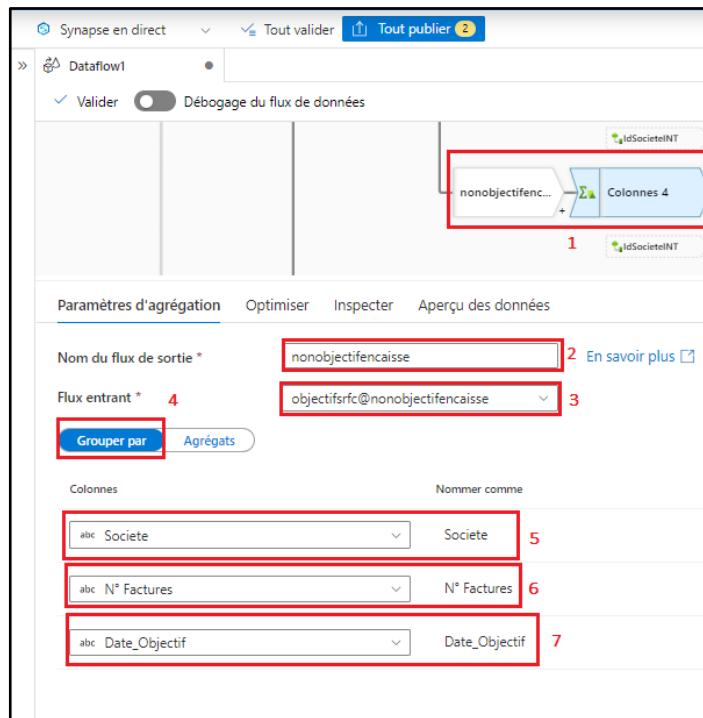
Figure 94 : Jointure id date paiement

Pour le flux “nonobjectifencasse”, on ajoute une agrégation sous le nom “nonobjectifencasse” avec les conditions suivantes :

- Grouper par pour afficher les colonnes suivantes (Figure 95) :

Colonnes	Nommer comme
Societe	Societe

N°Factures	N°Factures
Date_Objectif	Date_Objectif

Tableau 28 : L'agrégation des non objectifs encaissés de RFC (Grouper par)**Figure 95 : L'agrégation des non objectifs encaissés de RFC (Grouper par)**

➤ Agrégats (Figure 96) :

Colonne	Conditions
nonobjectifencaisse	sum({<Montant reçu>})

Tableau 29 : L'agrégation des non objectifs encaissés de RFC (Agrégats)

Cette condition sert à prendre les montants qui ont été payés, qui respectent la condition du flux “nonobjectifencaisse” et de les mettre dans la colonne Non objectif encaissé. Chaque facture de ce flux aura un montant dans la colonne Non objectif encaissé.

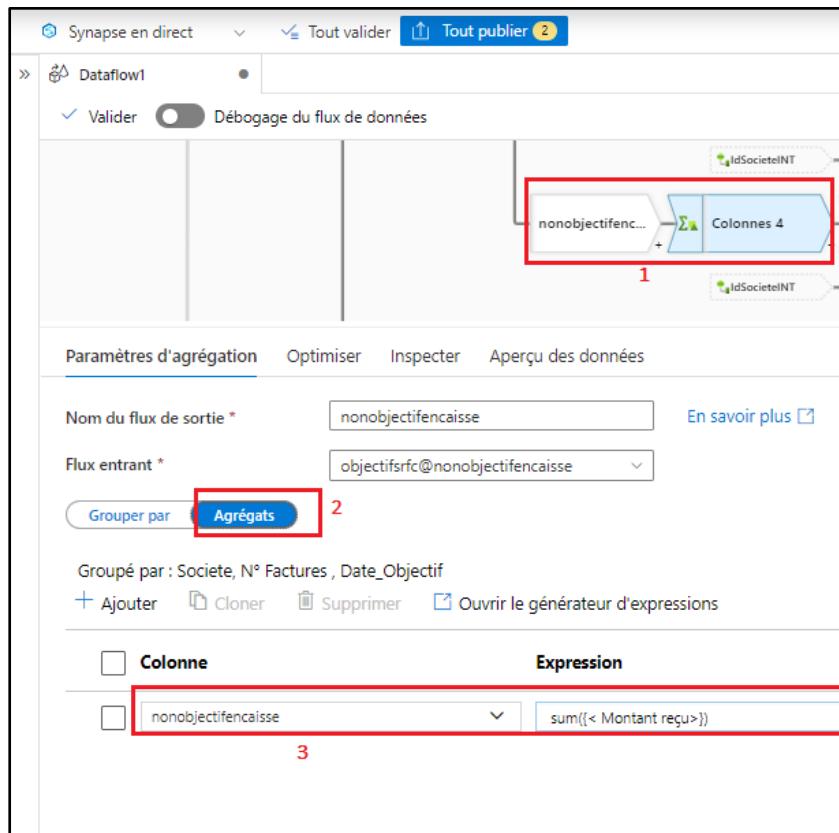
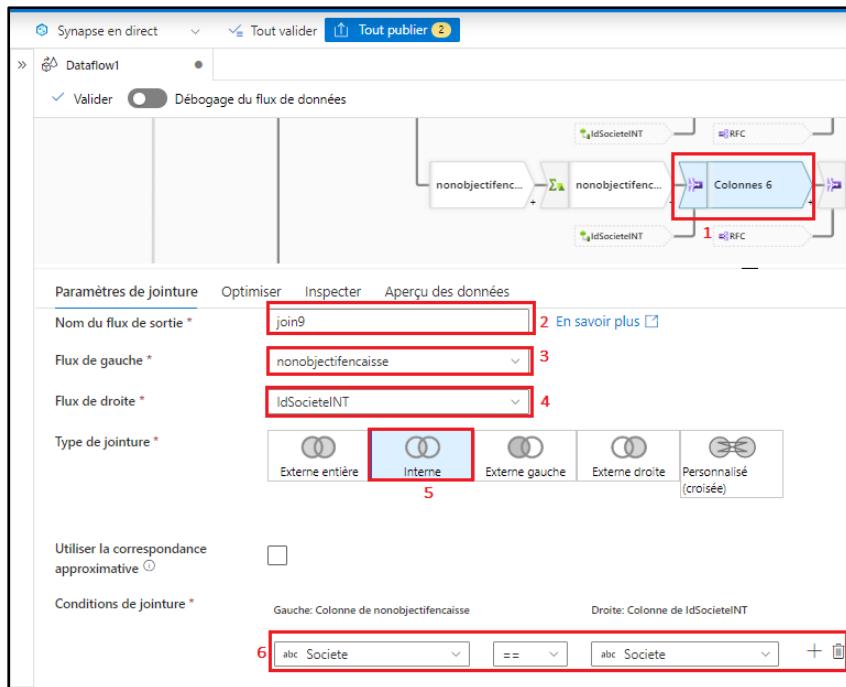


Figure 96 : L'agrégation des non objectifs encaissés de RFC (Agrégats)

On va ensuite ajouter une jointure entre l'agrégation précédente et le flux “IdSocieteINT” pour ajouter à chaque facture correspondante à l'objectif encaissé l'id de RFC avec comme condition de jointure (Figure 97) :

Colonne de nonobjectifencaisse		Colonne de IdSocieteINT
Societe	==	Societe

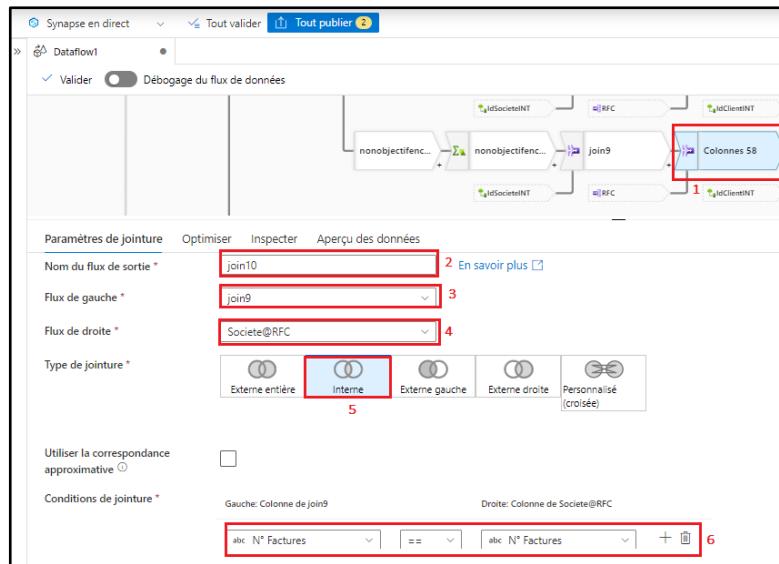
Tableau 30 : Condition de jointure 1

**Figure 97 : Jointure Id Société**

On va ensuite ajouter une jointure entre la jointure précédente et le flux Societe@RFC pour ajouter le numéro de facture de chaque facture correspondante au non objectif encaissé avec comme condition de jointure (Figure 98) :

Colonne de join9		Colonne de Societe@RFC
N°Factures	==	N°Factures

Tableau 31 : Condition de jointure 2

**Figure 98 : Jointure numéro facture**

On va ensuite ajouter une jointure entre la jointure précédente et le IdClientINT pour ajouter l'id du client de chaque facture avec comme condition de jointure (Figure 99) :

Colonne de join10		Colonne de IdClientINT
Client	==	Client

Tableau 32 : Condition de jointure 3

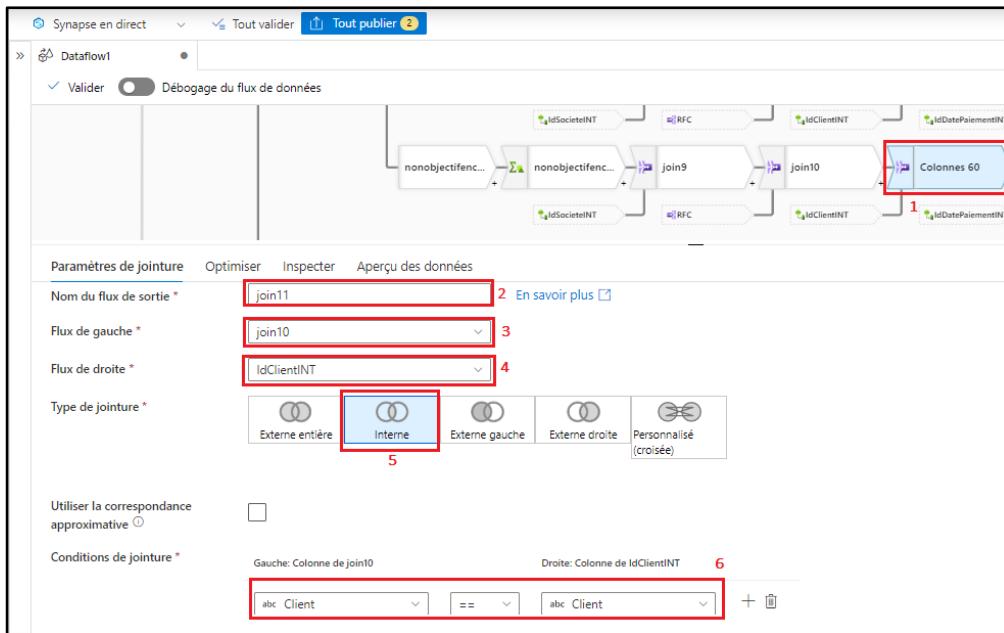


Figure 99 : Jointure id client

On va ensuite ajouter une jointure entre la jointure précédente et le IdDatePaiementINT pour ajouter l'id de la date à laquelle chaque facture correspondante à un objectif doit être comptabilisée avec comme condition de jointure (Figure 100) :

Colonne de join11		Colonne de IdDatePaiementINT
datepaiement	==	datepaiement

Tableau 33 : Condition de jointure 4

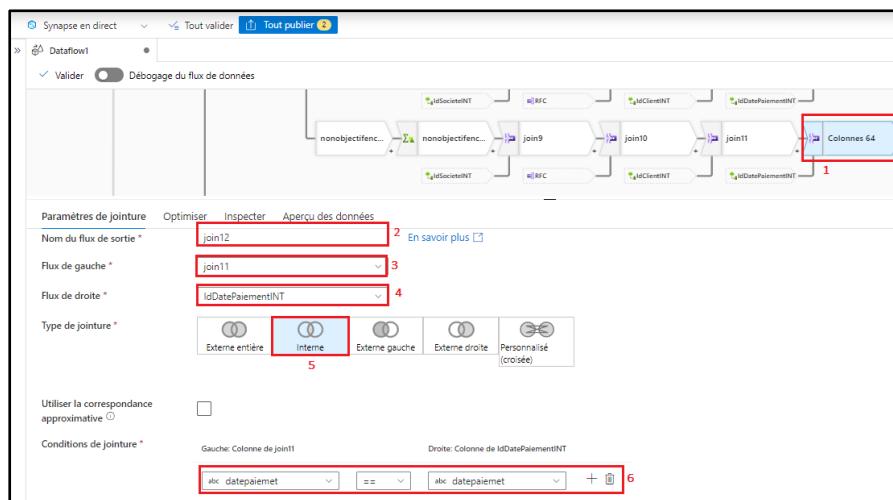


Figure 100 : Jointure id date paiement

3.3.2 Flux Compusell

Pour le flux de Compusell, on ajoute un autre fractionnement conditionnel sous le nom "objectifrfc" avec les conditions suivantes pour répartir les objectifs sur trois comme l'a demandé le département Finance (Figure 101) :

Noms de flux	Condition	Explication
Encours	equals({Statut }, 'Encours')	Le flux encours aura les factures qui seront encore en cours de paiement.
Paye	equals({Statut }, 'Payé')&&equals({Année de paiement réelle}, {Année})&&greaterOrEqual({Mois de paiement réelle}, Mois) equals({Statut }, 'Payé')&&greater({Année de paiement réelle}, {Année})	Ce flux correspond aux objectifs encaissés ce qui veut que ce flux aura les factures qui ont été payées à l'échéance ou après l'échéance (les impayés recouvrés).
Nonobjectifencaisse	Vide	Ce flux aura les factures restantes et donc les factures payées avant le mois de leur échéance.

Tableau 34 : Fractionnement des objectifs de Compusell

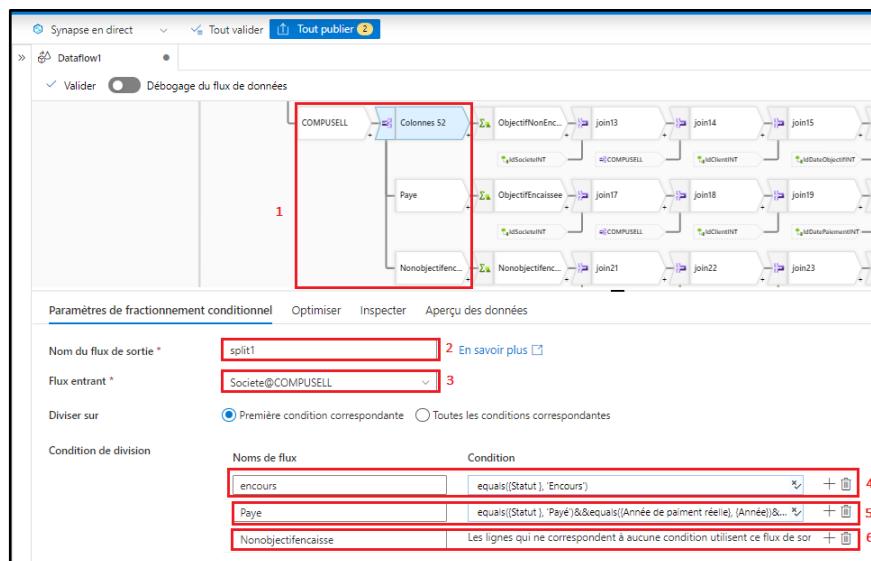


Figure 101 : Fractionnement des objectifs de Compusell

Pour le flux encours, on ajoute une agrégation sous le nom “ObjectifNonEncaisse” avec les conditions suivantes :

- Grouper par pour afficher les colonnes suivantes (Figure 102) :

Colonnes	Nommer comme
Societe	Societe
N°Factures	N°Factures
Date_Objectif	Date_Objectif

Tableau 35 : L'agrégation des objectifs non-encaissés de Compusell (Grouper par)

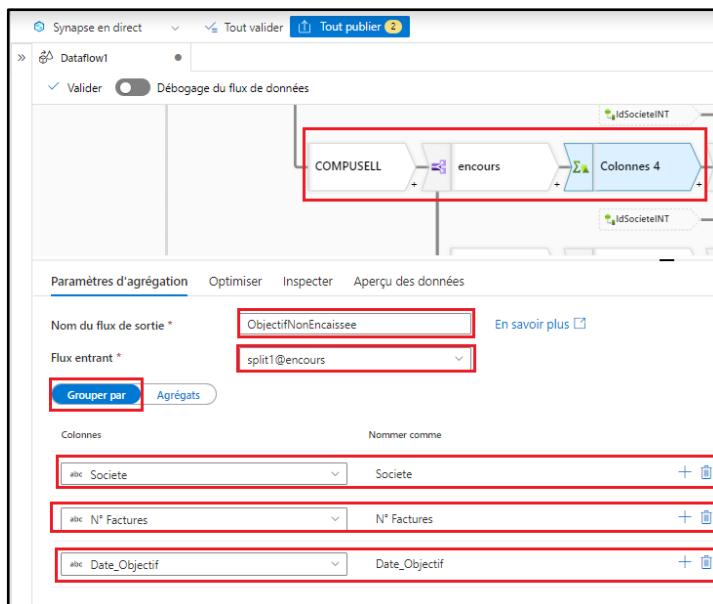


Figure 102 : L'agrégation des objectifs non-encaissés de Compusell (Grouper par)

- Agrégats (Figure 103) :

Colonne	Conditions
objectifnonencaissee	sum({<Reste à recevoir>})

Tableau 36 : L'agrégation des objectifs non-encaissés de Compusell (Agrégats)

Cette condition sert à prendre les montants qui n'ont pas encore été payés et de les mettre dans la colonne objectif non encaissé. Les montants relatifs aux objectifs seront répartis par facture.

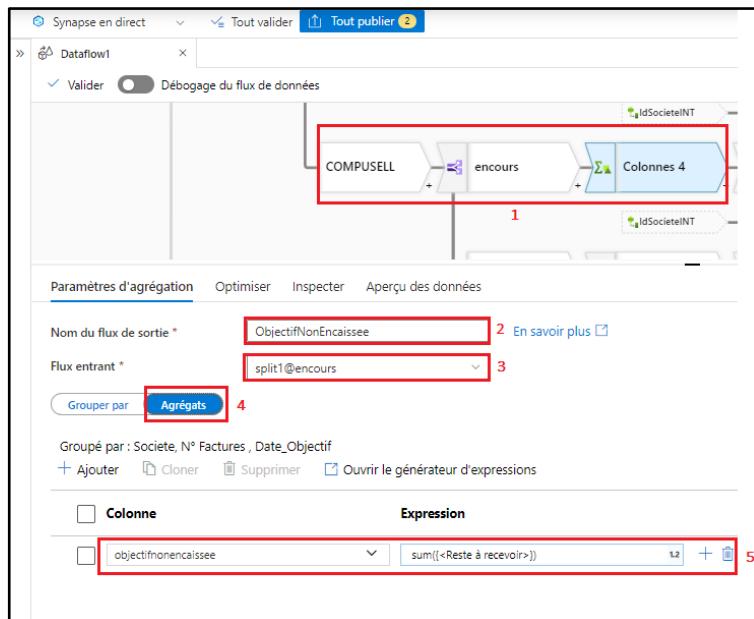
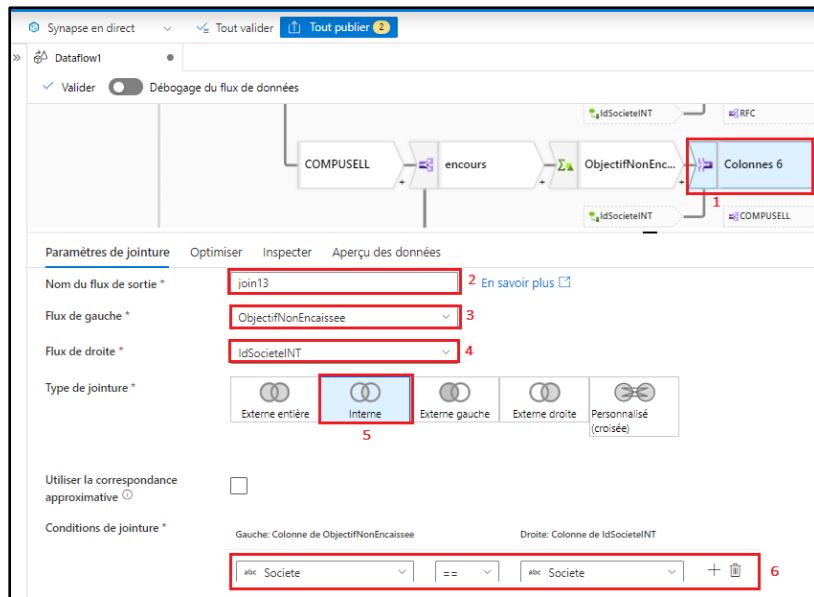


Figure 103 : L'agrégation des objectifs non-encaissés de Compusell (Agrégats)

On va ensuite ajouter une jointure entre l'agrégation précédente et le flux “IdSocieteINT” pour ajouter à chaque facture l'id de RFC avec comme condition de jointure (Figure 104) :

Colonne de objectifNonEncaisee		Colonne de IdSocieteINT
Societe	==	Societe

Tableau 37 : Condition de jointure 1

**Figure 104: Jointure Id Société**

On va ensuite ajouter une jointure entre la jointure précédente et le flux Societe@RFC pour ajouter le numéro de facture de chaque facture correspondante à l'objectif non encaissé avec comme condition de jointure (Figure 105) :

Colonne de join1		Colonne de Societe@RFC
N°Factures	==	N°Factures

Tableau 38 : Condition de jointure 2

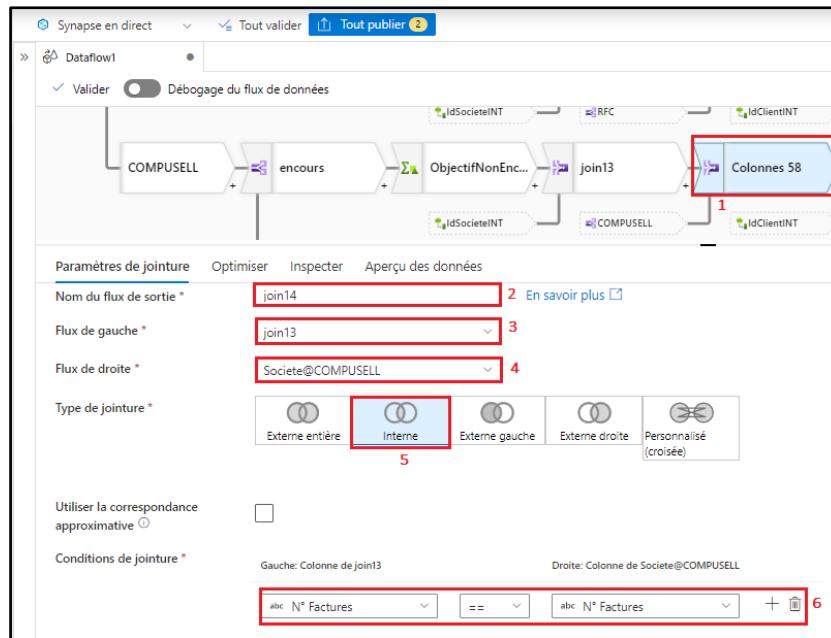


Figure 105 : Jointure numéro facture

On va ensuite ajouter une jointure entre la jointure précédente et le IdClientINT pour ajouter l'id du client de chaque facture avec comme condition de jointure (Figure 106) :

Colonne de join2		Colonne de IdClientINT
Client	==	Client

Tableau 39 : Condition de jointure 3

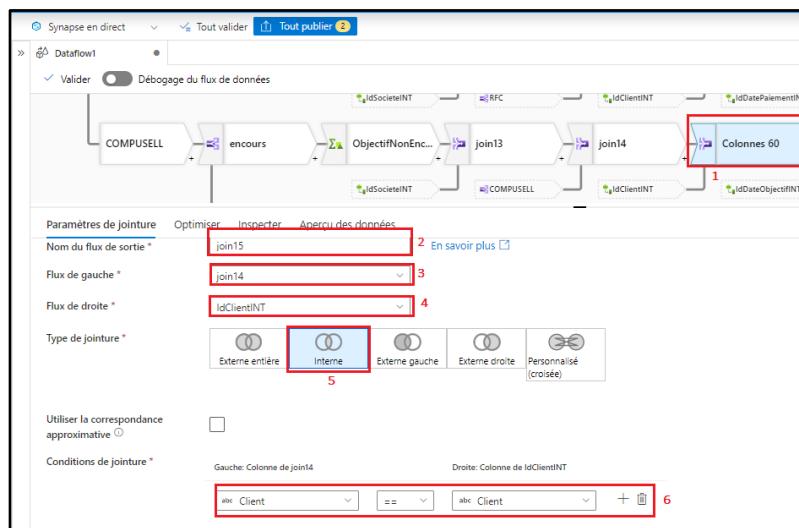


Figure 106 : Jointure id client

On va ensuite ajouter une jointure entre la jointure précédente et le IdDateObjectifINT pour ajouter l'id de la date à laquelle chaque facture correspondante à un objectif doit être comptabilisée avec comme condition de jointure (Figure 107) :

Colonne de join3		Colonne de IdDateObjectifINT
objectifnonencaisse@Date_Objectif	==	Date_Objectif

Tableau 40 : Condition de jointure 4

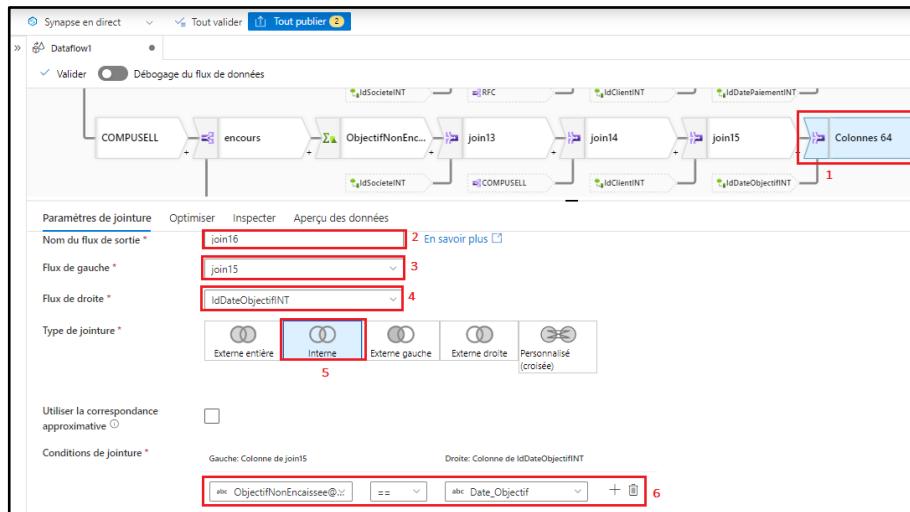


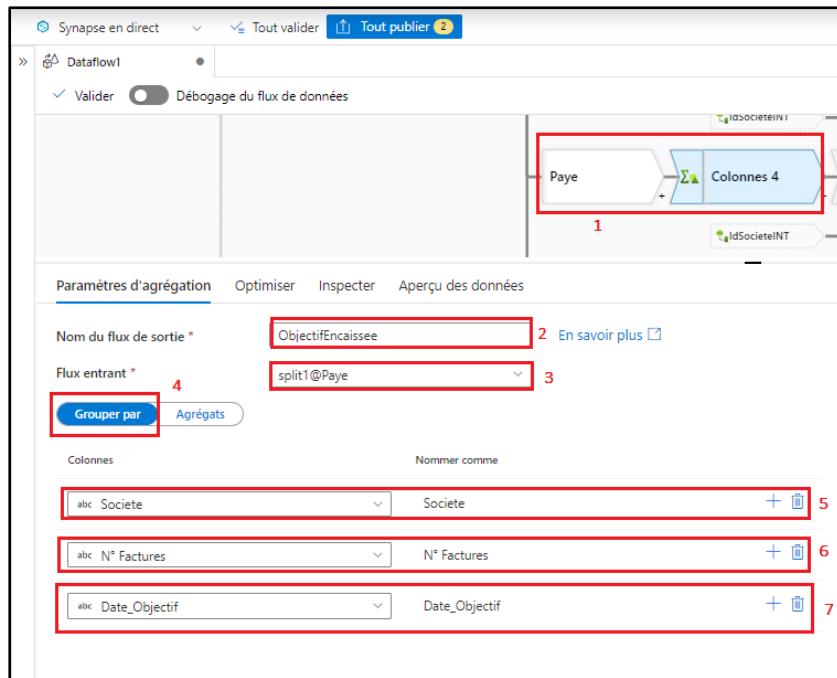
Figure 107 : Jointure id date objectif

Pour le flux paye, on ajoute une agrégation sous le nom “ObjectifEncaissée” avec les conditions suivantes :

- Grouper par pour afficher les colonnes suivantes (Figure 108) :

Colonnes	Nommer comme
Societe	Societe
N°Factures	N°Factures
Date_Objectif	Date_Objectif

Tableau 41 : L'agrégation des objectifs encaissés de Compusell (Grouper par)

**Figure 108 : L'agrégation des objectifs encaissés de Compusell (Grouper par)**

➤ Agrégats (Figure 109) :

Colonne	
objectifecaisse	sum({<Montant reçu>})

Tableau 42 : L'agrégation des objectifs encaissés de Compusell (Agrégats)

Cette condition sert à prendre les montants qui ont été payés, qui respectent la condition du flux "ObjectifEncaissee" et de les mettre dans la colonne objectif encaissé. Chaque facture de ce flux aura un montant dans la colonne objectif encaissé.

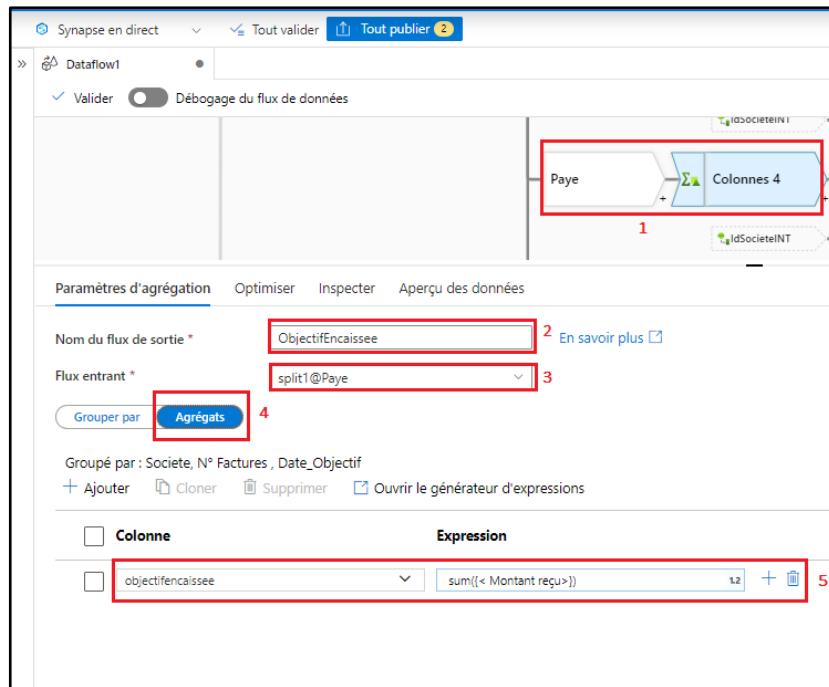


Figure 109 : L'agrégation des objectifs encaissés de Compusell (Agrégats)

On va ensuite ajouter une jointure entre l'agrégation précédente et le flux “IdSocieteINT” pour ajouter à chaque facture correspondante à l'objectif encaissé l'id de RFC avec comme condition de jointure (Figure 110) :

Colonne de ObjectifEncaisse		Colonne de IdSocieteINT
Societe	==	Societe

Tableau 43 : Condition de jointure 1

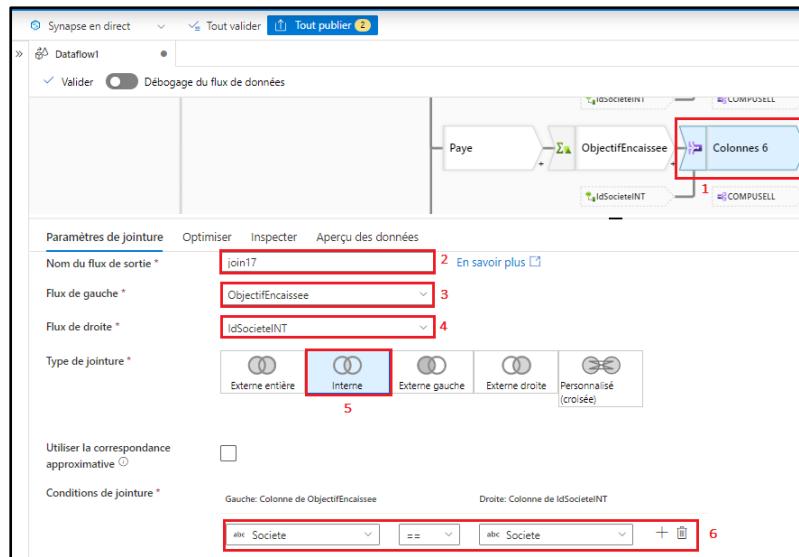


Figure 110 : Jointure Id Société

On va ensuite ajouter une jointure entre la jointure précédente et le flux Societe@RFC pour ajouter le numéro de facture de chaque facture correspondante à l'objectif encaissé avec comme condition de jointure (Figure 111) :

Colonne de join5		Colonne de Societe@RFC
N°Factures	==	N°Factures

Tableau 44 : Condition de jointure 2

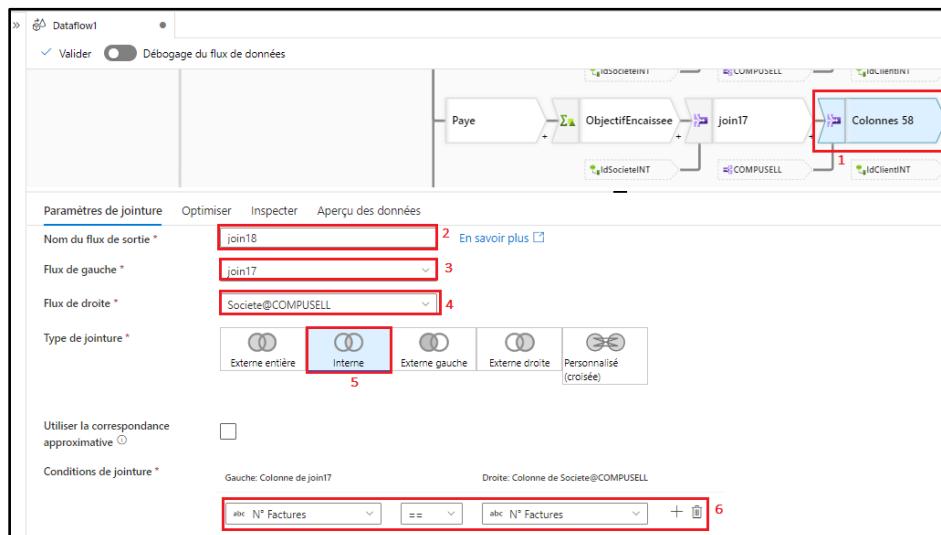


Figure 111 : Jointure numéro facture

On va ensuite ajouter une jointure entre la jointure précédente et le IdClientINT pour ajouter l'id du client de chaque facture avec comme condition de jointure (Figure 112) :

Colonne de join6		Colonne de IdClientINT
Client	==	Client

Tableau 45 : Condition de jointure 3

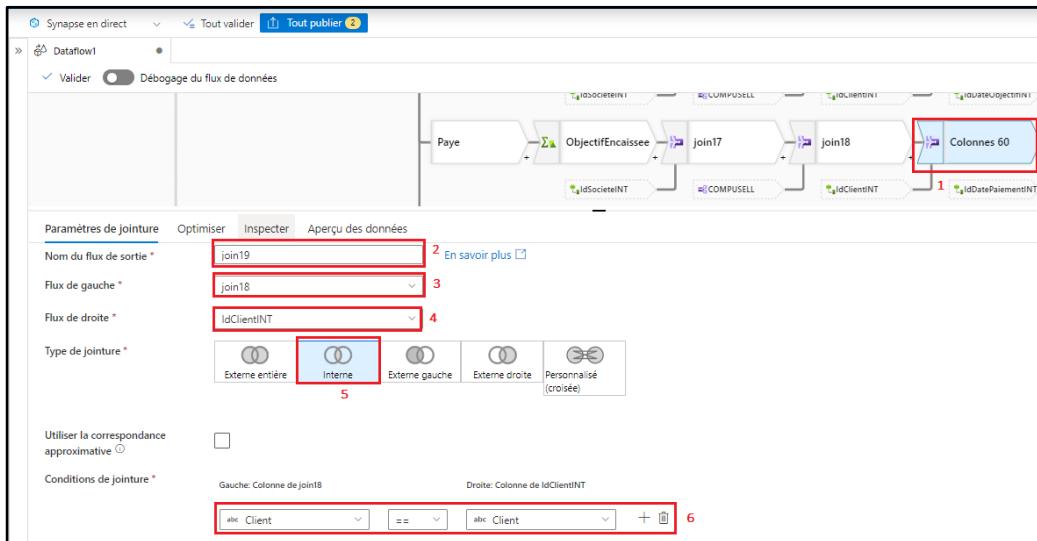
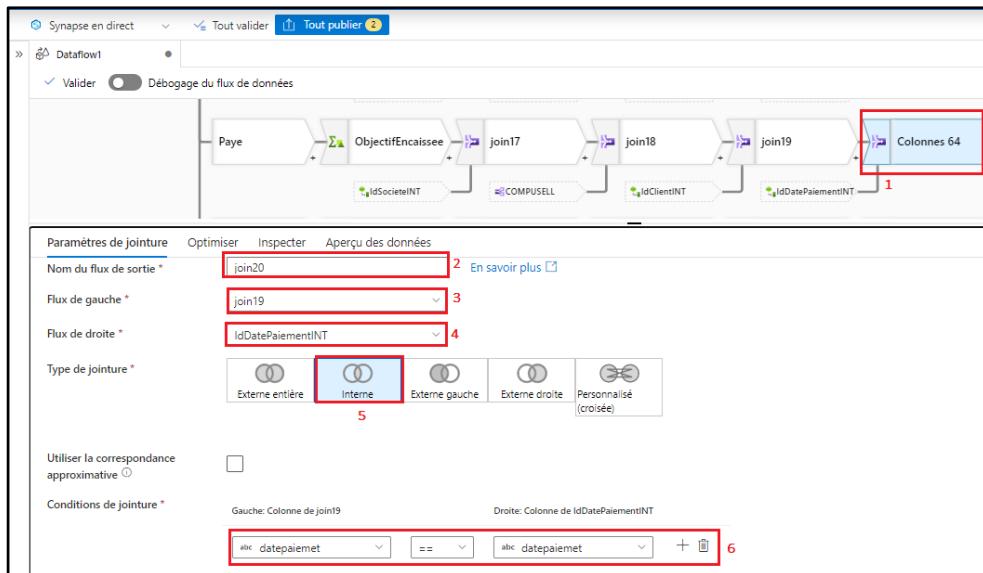


Figure 112 : Jointure id client

On va ensuite ajouter une jointure entre la jointure précédente et le IdDatePaiementINT pour ajouter l'id de la date à laquelle chaque facture correspondante à un objectif doit être comptabilisée avec comme condition de jointure (Figure 113) :

Colonne de join7		Colonne de IdDatePaiementfINT
datepaiement	==	datepaiement

Tableau 46 : Condition de jointure 4

**Figure 113 : Jointure id date paiement**

Pour le flux Nonobjectifencaissee, on ajoute une agrégation sous le nom “Nonobjectifencaissee” avec les conditions suivantes :

- Grouper par pour afficher les colonnes suivantes (Figure 114) :

Colonnes	Nommer comme
Societe	Societe
N°Factures	N°Factures
Date_Objectif	Date_Objectif

Tableau 47 : L'agrégation des non objectifs encaissés de Compusell (Grouper par)

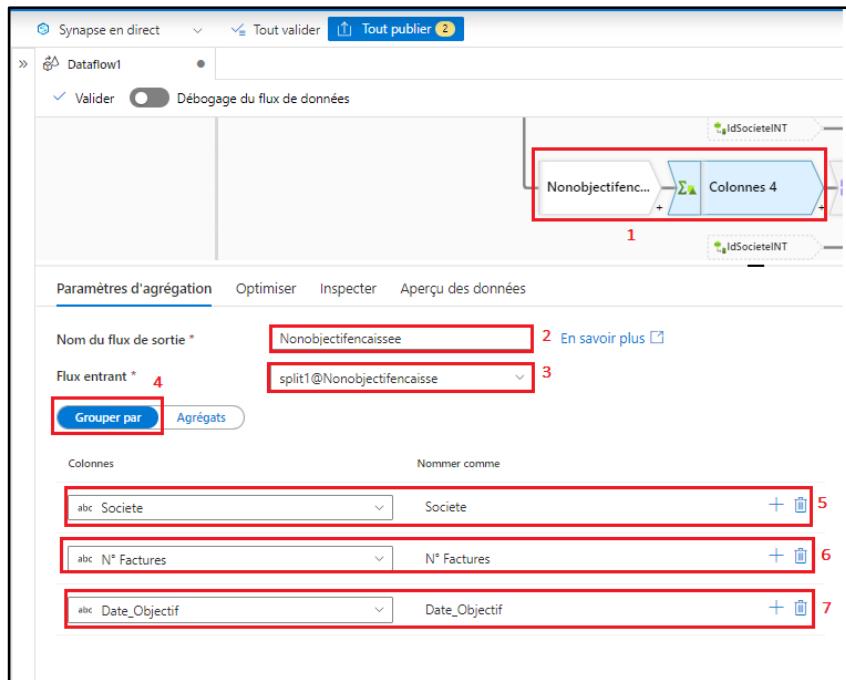


Figure 114 : L'agrégation des non objectifs encaissés de Compusell (Grouper par)

➤ Agrégats (Figure 115) :

Colonne	Conditions
nonobjectifencaissee	sum({<Montant reçu>})

Tableau 48 : L'agrégation des non objectifs encaissés de Compusell (Agrégats)

Cette condition sert à prendre les montants qui ont été payés, qui respectent la condition du flux "Nonobjectifencaissee" et de les mettre dans la colonne Non objectif encaissé. Chaque facture de ce flux aura un montant dans la colonne Non objectif encaissé.

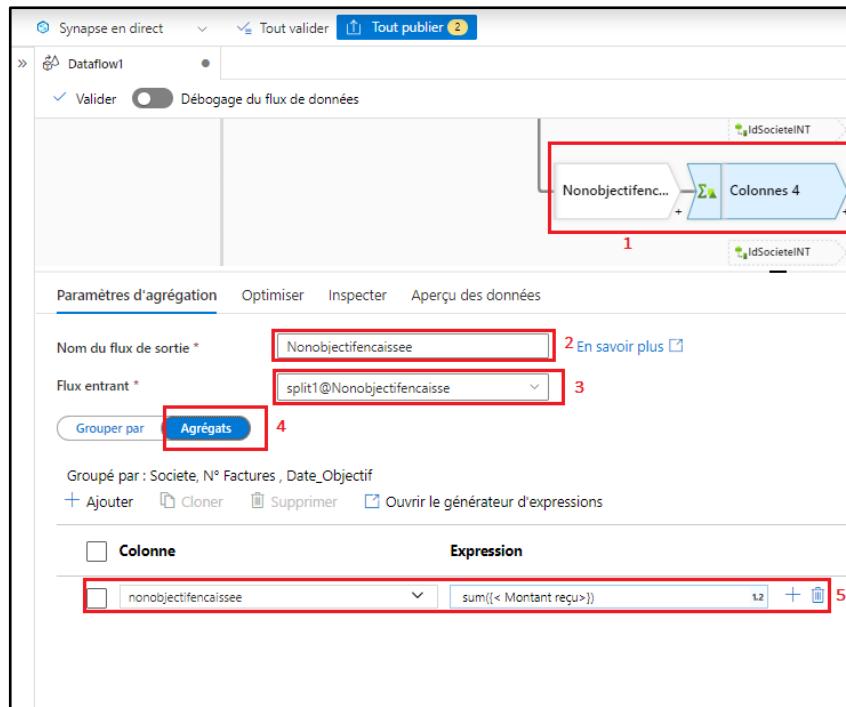
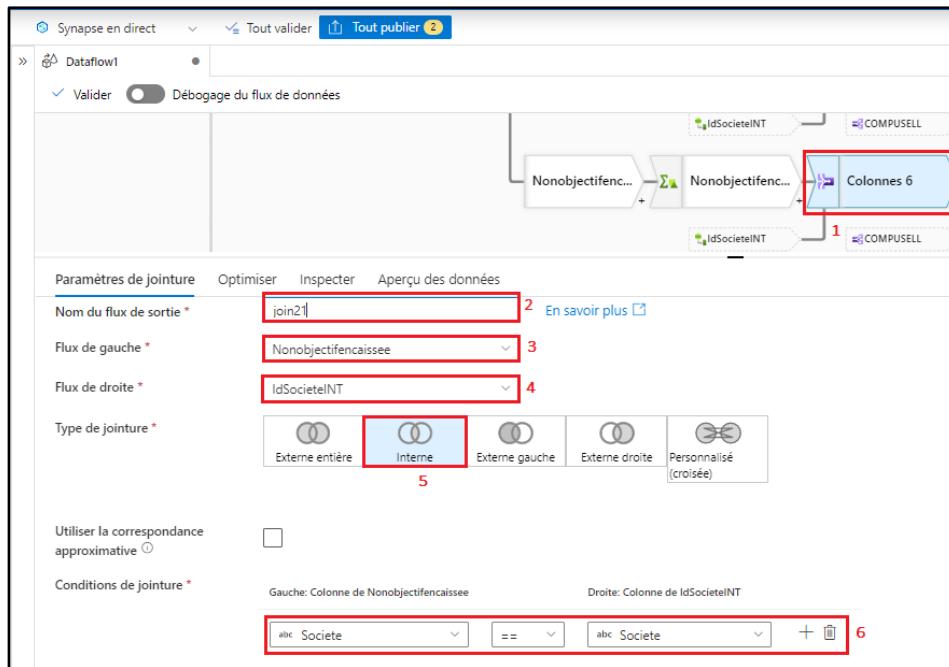


Figure 115 : L'agrégation des non objectifs encaissés de Compusell (Agrégats)

On va ensuite ajouter une jointure entre l'agrégation précédente et le flux “IdSocieteINT” pour ajouter à chaque facture correspondante à l'objectif encaissé l'id de RFC avec comme condition de jointure (Figure 116) :

Colonne de Nonobjectifencaissee		Colonne de IdSocieteINT
Societe	==	Societe

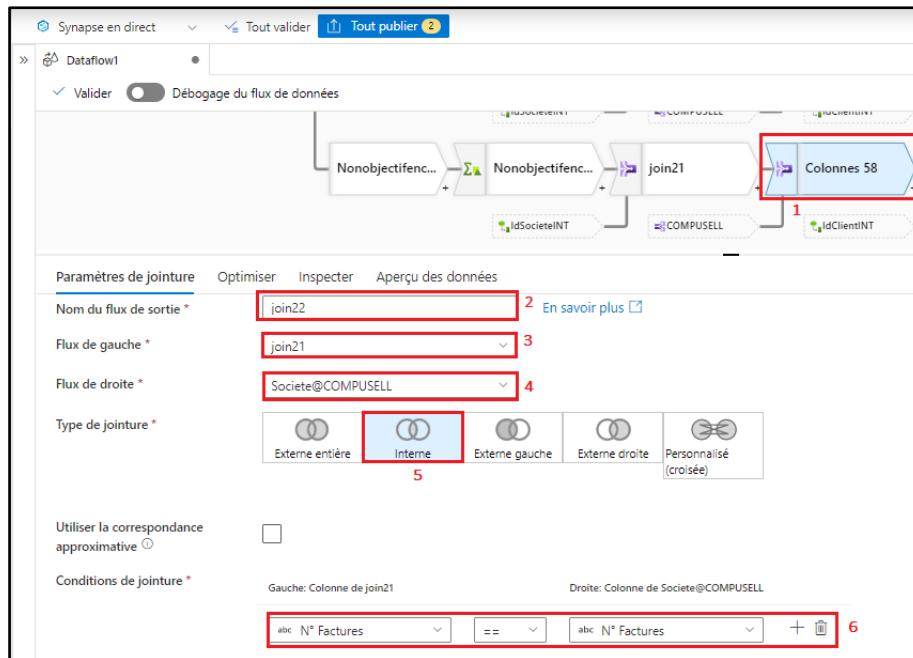
Tableau 49 : Condition de jointure 1

**Figure 116 : Jointure Id Société**

On va ensuite ajouter une jointure entre la jointure précédente et le flux Societe@RFC pour ajouter le numéro de facture de chaque facture correspondante au non objectif encaissé avec comme condition de jointure (Figure 117) :

Colonne de join9		Colonne de Societe@RFC
N°Factures	==	N°Factures

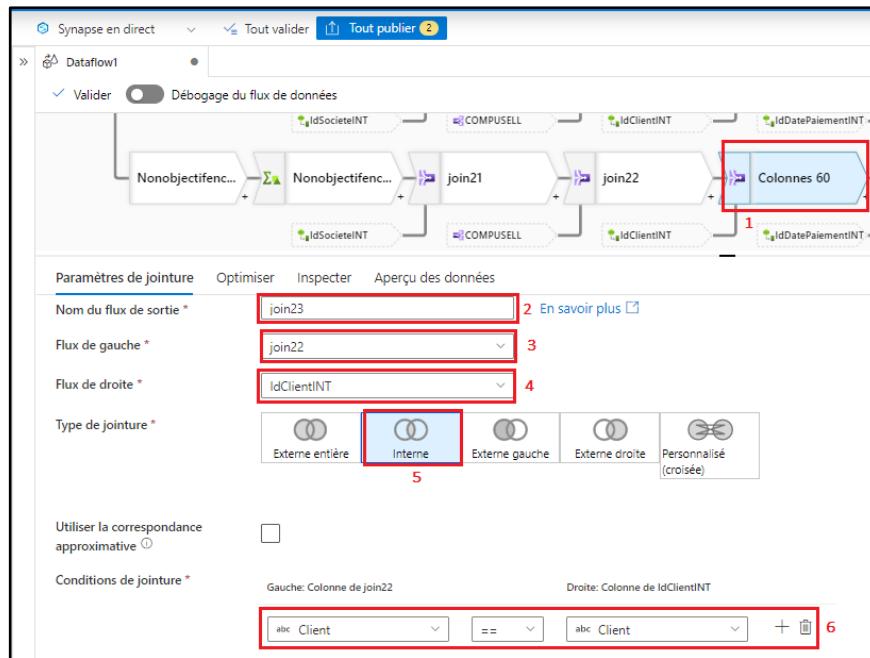
Tableau 50 : Condition de jointure 2

**Figure 117 : Jointure numéro facture**

On va ensuite ajouter une jointure entre la jointure précédente et le IdClientINT pour ajouter l'id du client de chaque facture avec comme condition de jointure (Figure 118) :

Colonne de join10		Colonne de IdClientINT
Client	==	Client

Tableau 51 : Condition de jointure 3

**Figure 118 : Jointure id client**

On va ensuite ajouter une jointure entre la jointure précédente et le IdDatePaiementINT pour ajouter l'id de la date à laquelle chaque facture correspondante à un objectif doit être comptabilisée avec comme condition de jointure (Figure 119) :

Colonne de join11		Colonne de IdDatePaiementfINT
datepaiement	==	datepaiement

Tableau 52 : Condition de jointure 4

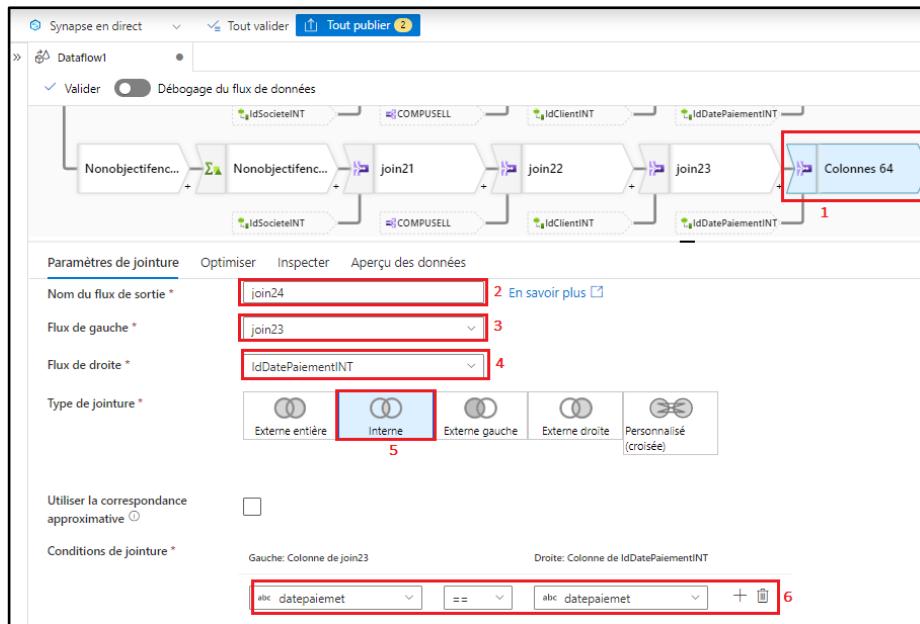


Figure 119 : Jointure id date paiement

4. Chargement

Pour le chargement de données, nous avons tout d'abord créé des jeux de données d'intégration à partir des tables créées dans "datawarehouseyasminepfe".

Voici le jeu de données qui fait référence à la table de dimension Dim_client et qui permettra de charger les données dans le Datawarehouse :

The screenshot shows the Azure Data Factory dataset configuration for the Dim_client table. The table is linked to a service named 'workspaceyasminepfe-WorkspaceDe...' with a database name 'datawarehouseyasminepfe'. The runtime integration is set to 'AutoResolveIntegrationRuntime' and the table is 'dbo.Dim_client'.

Figure 120 : Jeu de données Dim_client

Voici le jeu de données qui fait référence à la table de dimension Dim_date et qui permettra de charger les données dans le Datawarehouse :

The screenshot shows the Azure Synapse Analytics Data Catalog interface. At the top, there's a navigation bar with icons for Home, Datasets, Pipelines, and more. Below it, a search bar and a breadcrumb trail: 'Azure Synapse Analytics / Dim_date'. The main area is titled 'Connexion' (Connection). It shows a linked service named 'workspaceyasminepfe-WorkspaceDe...' with a 'DBName' set to 'datawarehouseyasminepfe'. Under 'Runtime d'intégration' (Integration Runtime), it's set to 'AutoResolveIntegrationRuntime'. The 'Table' dropdown is set to 'dbo.Dim_date'. There are buttons for 'Actualiser' (Refresh) and 'Aperçu des données' (Preview data).

Figure 121 : Jeu de données Dim_date

Voici le jeu de données qui fait référence à la table de dimension Dim_facture et qui permettra de charger les données dans le Datawarehouse :

This screenshot is identical to Figure 121, showing the configuration for the 'Dim_facture' dataset. The connection details, including the linked service 'workspaceyasminepfe-WorkspaceDe...', the integration runtime 'AutoResolveIntegrationRuntime', and the table 'dbo.Dim_facture', are all the same as in Figure 121.

Figure 122 : Jeu de données Dim_facture

Voici le jeu de données qui fait référence à la table de dimension Dim_paiement et qui permettra de charger les données dans le Datawarehouse :

The screenshot shows the Azure Synapse Analytics Data Catalog interface. At the top, there's a navigation bar with icons for Home, Datasets, Pipelines, and more. Below it, a search bar and a breadcrumb trail: 'Azure Synapse Analytics / Dim_paiement'. The main area is titled 'Connexion' (Connection). It shows a linked service named 'workspaceyasminepfe-WorkspaceDe...', a database named 'datawarehouseyasminepfe', and a table named 'dbo.Dim_paiement'. There are buttons for 'Tester la connexion' (Test connection), 'Modifier' (Edit), 'Nouveau' (New), and 'En savoir plus' (Learn more). A 'Runtime d'intégration' (Integration runtime) dropdown is set to 'AutoResolveIntegrationRuntime'. Below the connection settings, there are buttons for 'Actualiser' (Refresh) and 'Aperçu des données' (Preview data). A 'Modifier' (Edit) checkbox is also present.

Figure 123 : Jeu de données Dim_paiement

Voici le jeu de données qui fait référence à la table de dimension Dim_societe et qui permettra de charger les données dans le Datawarehouse :

The screenshot shows the Azure Synapse Analytics Data Catalog interface, similar to Figure 123. At the top, there's a navigation bar with icons for Home, Datasets, Pipelines, and more. Below it, a search bar and a breadcrumb trail: 'Azure Synapse Analytics / Dim_societe'. The main area is titled 'Connexion' (Connection). It shows a linked service named 'workspaceyasminepfe-WorkspaceDe...', a database named 'datawarehouseyasminepfe', and a table named 'dbo.Dim_societe'. There are buttons for 'Tester la connexion' (Test connection), 'Modifier' (Edit), 'Nouveau' (New), and 'En savoir plus' (Learn more). A 'Runtime d'intégration' (Integration runtime) dropdown is set to 'AutoResolveIntegrationRuntime'. Below the connection settings, there are buttons for 'Actualiser' (Refresh) and 'Aperçu des données' (Preview data). A 'Modifier' (Edit) checkbox is also present.

Figure 124 : Jeu de données Dim_societe

Voici le jeu de données qui fait référence à la table de faits Fact_analyse et qui permettra de charger les données dans le Datawarehouse.

The screenshot shows the 'Fact_analyse' dataset configuration in Azure Synapse Analytics. The top navigation bar includes icons for 'Azure Synapse Analytics', 'Fact_analyse', and three dots for more options. Below the navigation is a toolbar with 'Connexion', 'Schéma', and 'Réglages'. The 'Connexion' tab is selected, showing a linked service named 'workspaceyasminepef-WorkspaceDe...' with a 'Test connection' button, a 'Modify' button, a 'New' button, and a 'More information' link. Under 'Propriétés du service lié', there is a 'Nom' (Name) field set to 'datawarehouseyasminepef' and a 'Type' field set to 'String'. The 'Runtime d'intégration' section shows 'AutoResolveIntegrationRuntime' selected. The 'Table' section shows 'dbo.Fact_analyse_recouvrement' selected, with 'Actualiser' (Refresh) and 'Aperçu des données' (Preview data) buttons, and a 'Modifier' (Modify) checkbox. The bottom right corner of the interface has a small 'Up' arrow icon.

Figure 125 : Jeu de données Fact_analyse

Nous avons ensuite utilisé les flux de données pour ajouter un récepteur pour chaque transformation avec les mappages suivants :

- **Dim_client**

Colonne d'entrée	Colonne de sortie
Id_client	Id_Client
Client	Nom_Client

Tableau 53 : Mappage Dim_client

Voici le récepteur de la table Dim_client :

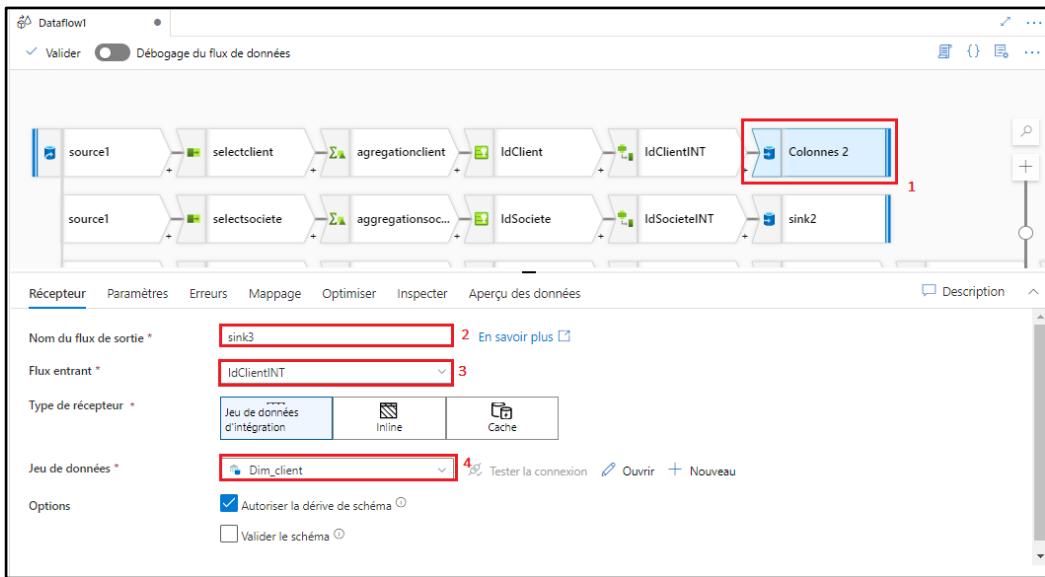


Figure 126 : Récepteur Dim_client

- **Dim_societe**

Colonne d'entrée	Colonne de sortie
Societe	Société
Id_Société	Id_Société

Tableau 54 : Mappage Dim_société

Voici le récepteur de la table Dim_societe :

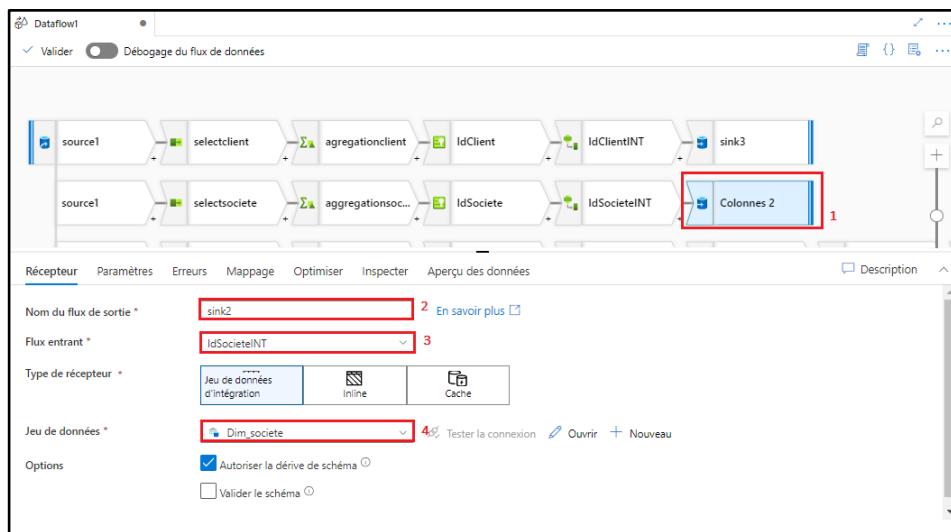


Figure 127 : Récepteur Dim_societe

- **Dim_facture :**

Colonne d'entrée	Colonne de sortie
Id_paiement	Id_Paiement
N° Factures	Num_Facture
Délais de paiement Contractuelle	Délai_De_Paiement_Contractuel
Statut Réception de facture	Statut_Réception
Statut	Statut
Nature de facturation	Nature_Facture
Nature de la créance	Nature_De_Créance

Tableau 55 : Mappage Dim_facture

Voici le récepteur de la table Dim_facture :

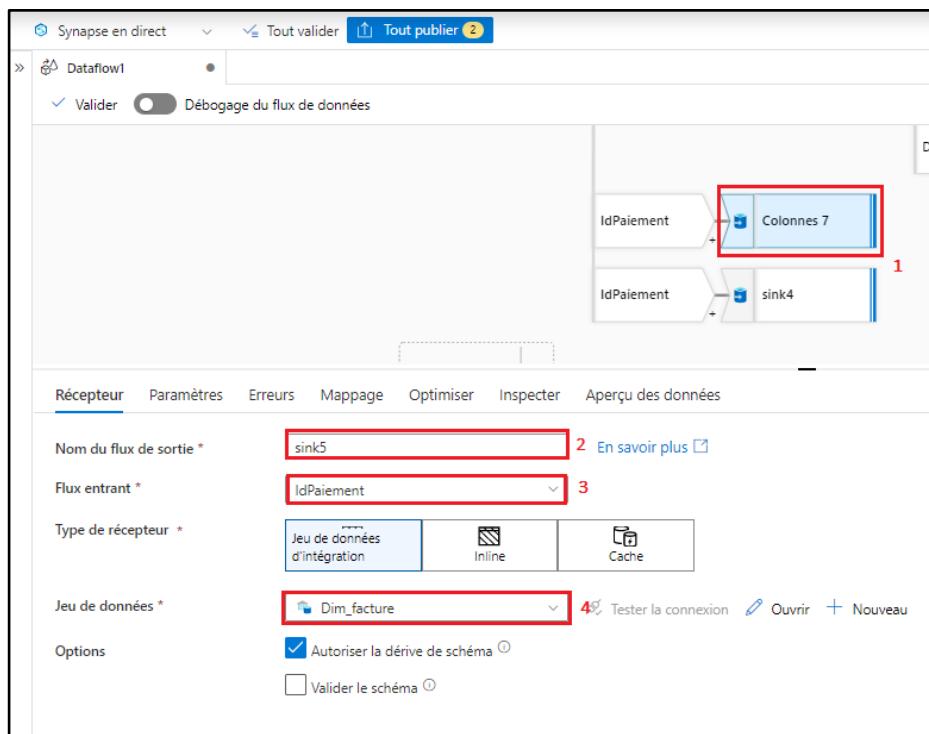


Figure 128 : Récepteur Dim_facture

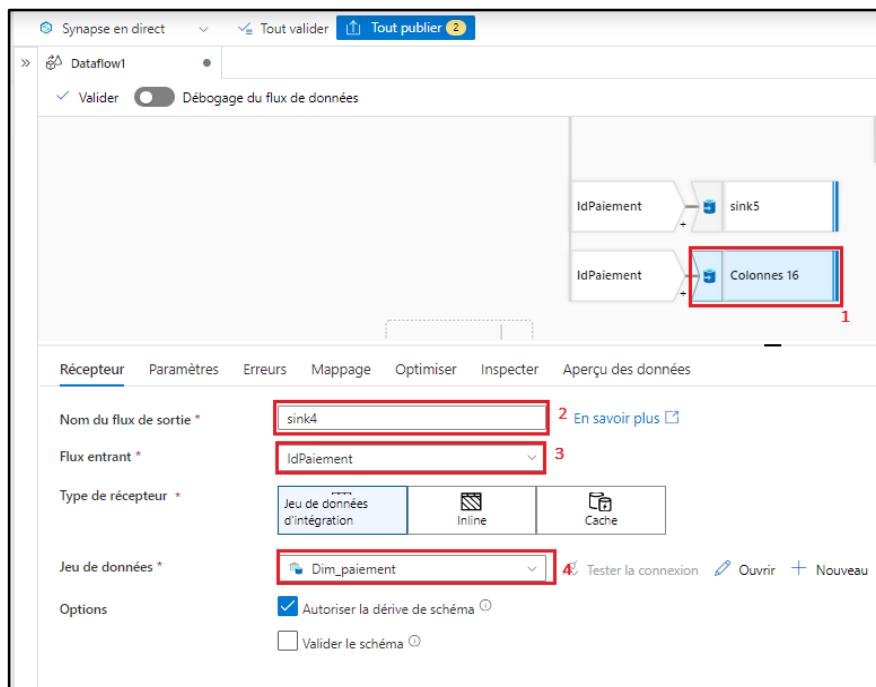
- **Dim_paiement :**

Colonne d'entrée	Colonne de sortie
Montant Factures en devise	Montant_En_Devise
Code devise	Code_Devise
Montant initial TND	Montant_Initial
< Avance 1>	Avance1
< Avance 2>	Avance2
< Avance 3>	Avance3
Avance 4	Avance4
< Avance 5>	Avance5
< Avance 6>	Avance6
Avance 8	Avance8

< Avance 7>	Avance7
< Montant reçu>	Total_Reçu
<Reste à recevoir>	Reste_à_Recevoir
N° Chèque/Traite	Num_Paiement
Mode de paiement	Mode_De_Paiement
Id_paiement	Id_Paiement

Tableau 56 : Mappage Dim_paiement

Voici le récepteur de la table Dim_paiement :

**Figure 129 : Récepteur Dim_paiement**

- **Dim_date**

Colonne d'entrée	Colonne de sortie
Année de paiement réelle	Année
Mois de paiement réelle	Mois
Id_date_paiement	Id_Date

Tableau 57 : Mappage Dim_date

Voici le récepteur de la table Dim_date:

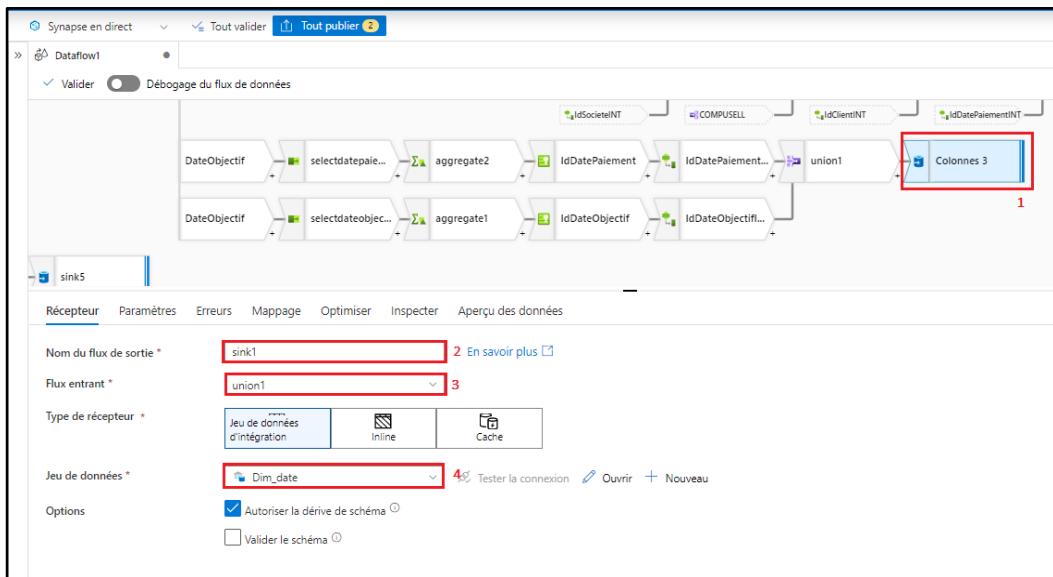


Figure 130 : Récepteur Dim_date

- **Fact_analyse**

Colonne d'entrée	Colonne de sortie
Id_Société	Id_Société
Nonobjectifencaissee@Date_Objectif	Date_Objectif
objectifnonencaisse@Date_Objectif	
objectifencaisse@Date_Objectif	
nonobjectifencaisse@Date_Objectif	
ObjectifNonEncaissee@Date_Objectif	
ObjectifEncaissee@Date_Objectif	
nonobjectifencaissee	Non_Objectif_Encaissé
nonobjectifencaisse	
objectifnonencaisse	Objectif_Non_Encaissé

objectifnonencaissee	
objectifencaisse	Objectif_Encaissé
objectifencaissee	
Nonobjectifencaissee@N° Factures	Num_Facture
objectifnonencaisse@N° Factures	
objectifencaisse@N° Factures	
nonobjectifencaisse@N° Factures	
ObjectifNonEncaissee@N° Factures	
ObjectifEncaissee@N° Factures	
Date Facture	Date_Facture
Id_client	Id_Client
Date de réception facture	Date_Réception_Facture
Date avance 1	Date_Avance1
Date avance 2	Date_Avance2
Date avance 3	Date_Avance3
Date avance 4	Date_Avance4
Date avance 5	Date_Avance5
Date avance 6	Date_Avance6
Date avance 7	Date_Avance7
Date avance 8	Date_Avance8
Date réception de paiement	Date_Réception_Paiement
Échéance Contractuelle	Echéance_Contractuelle
Id_date_paiement	Id_Date

id_date_objectif

Tableau 58 : Mappage Fact_analyse

Voici les récepteurs de la table Fact_analyse :

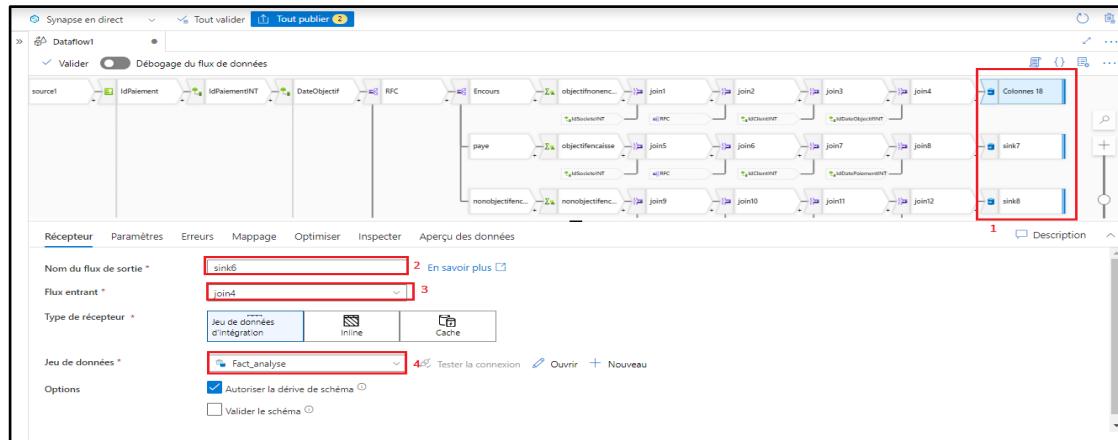


Figure 131 : Récepteur Fact_analyse (Partie 1)

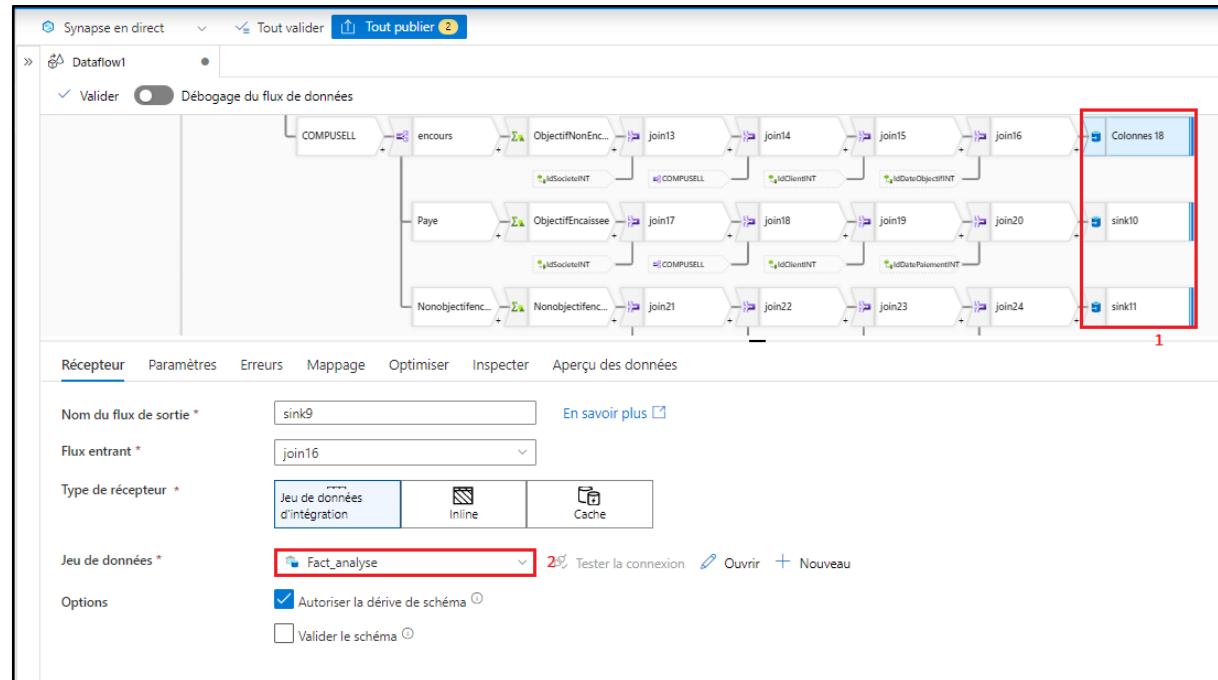


Figure 132 : Récepteur Fact_analyse (Partie 2)

5. Ajout des mesures

Pour ajouter les mesures à la table de faits, nous avons suivi les étapes suivantes :

Dans le hub "données", nous avons cliqué sur Base de données SQL, ensuite sur datawarehouseyasminepfe ensuite sur programmabilité => procédure stockée =>Nouveau script SQL=>Nouvelle procédure stockée où on va écrire le script permettant d'ajouter les mesures. (Figure 133)

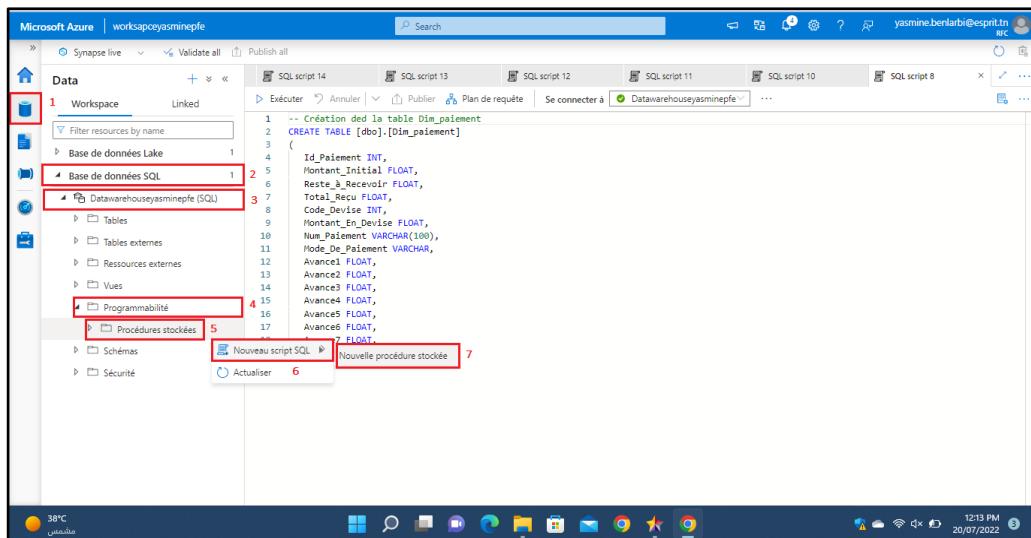


Figure 133 : Ajout des mesures (Partie 1)

On écrit la procédure [dbo].[Mesure] ensuite on l'exécute puis on actualise le dossier procédures stockées. Nous pouvons remarquer que la procédure a été ajoutée. Ensuite, nous l'ajoutons au pipeline créé auparavant comme le montre la figure 134.

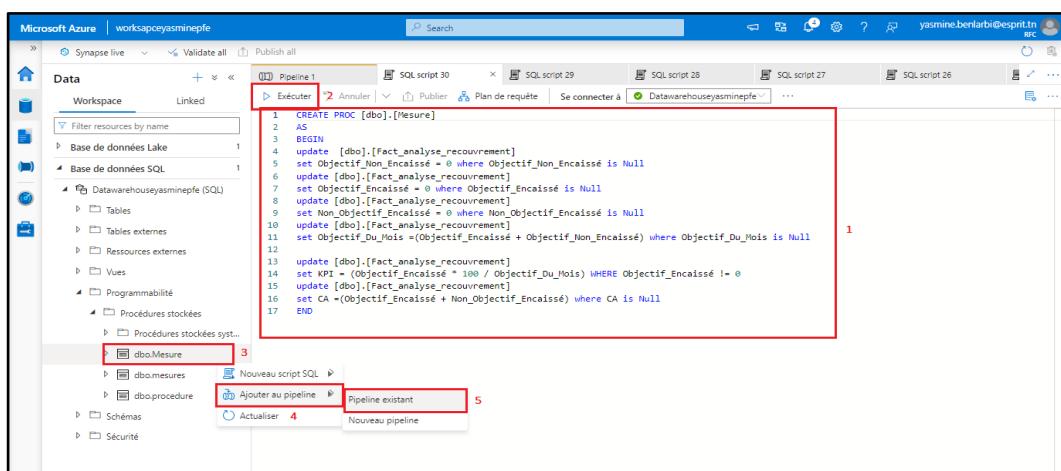


Figure 134 : Ajout des mesures (Partie 2)

On clique sur le bouton déboguer pour exécuter le pipeline pour charger les transformations dans le datawarehouse et ajouter les mesures dans la table de faits. (Figure 135)

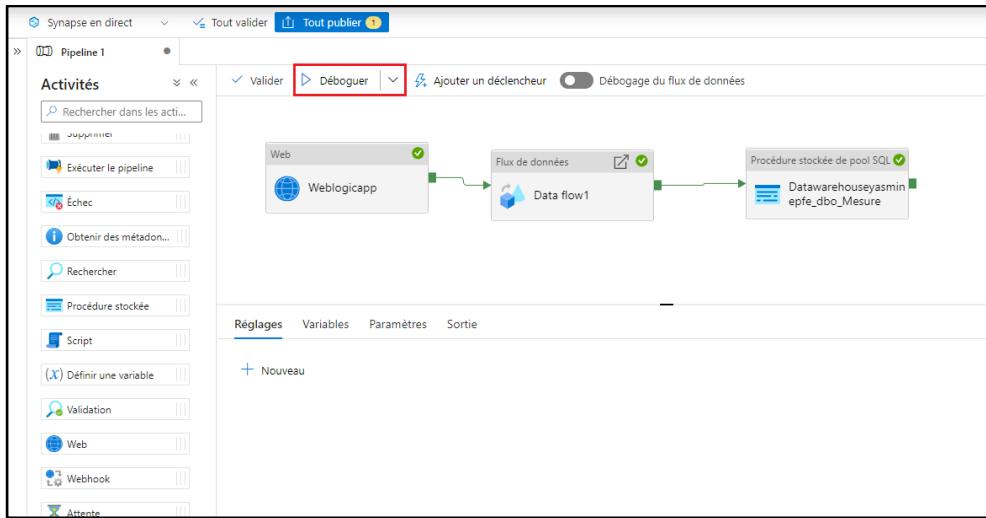


Figure 135 : Ajout des mesures (Partie 3)

Notre pipeline est maintenant complet. A chaque fois qu'une modification est faite au niveau du fichier source, elle atteindra automatiquement le Data Lake. Cependant, nous devrons exécuter le pipeline pour que la modification atteigne le dataflow ainsi que les tableaux de bord.

6. Visualisation

6.1 Connexion entre Azure Synapse Analytics et Power BI Desktop

Nous allons tout d'abord ouvrir Power BI desktop. Nous allons cliquer sur l'accueil puis sur obtenir les données ensuite sur plus. (Figure 136)

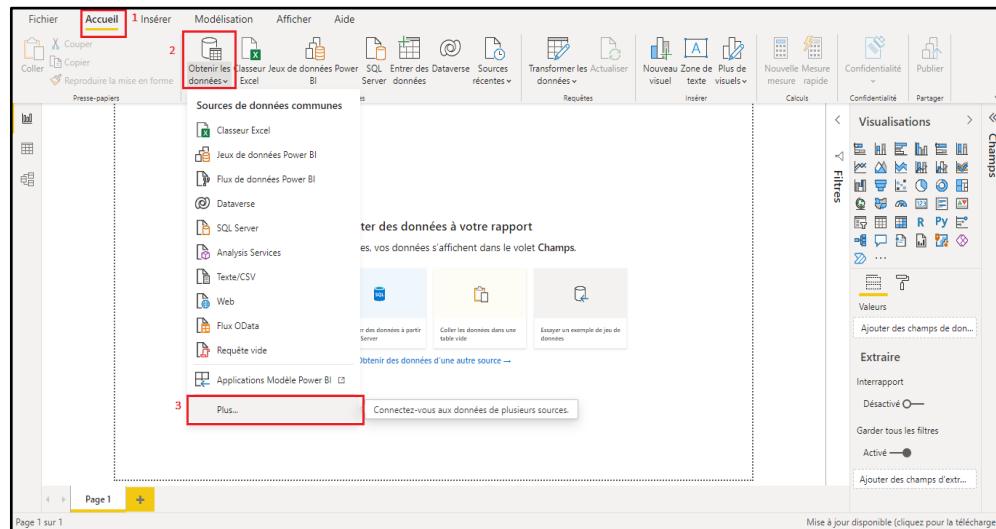


Figure 136 : Connexion entre Azure Synapse Analytics et Power BI Desktop (Partie 1)

Nous allons cliquer sur Azure ensuite Azure Synapse Analytics SQL puis sur se connecter. (Figure 137)

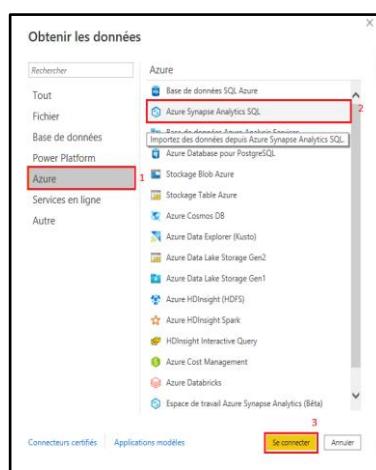


Figure 137 : Connexion entre Azure Synapse Analytics et Power BI Desktop (Partie 2)

Nous allons maintenant entrer le nom du serveur de notre espace de travail et on va se connecter en Direct Query pour accéder en temps réel à nos données transformées et chargées dans le datawarehouse. (Figure 138)



Figure 138 : Connexion entre Azure Synapse Analytics et Power BI Desktop (Partie 3)

On entre le nom d'utilisateur ainsi que le mot de passe fournis automatiquement à la création de l'espace de travail puis on se connecte. (Figure 139)

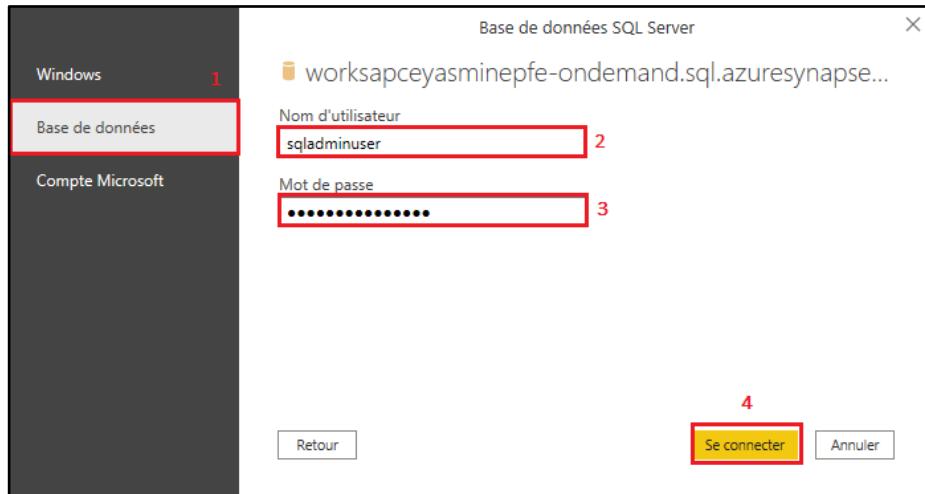


Figure 139 : Connexion entre Azure Synapse Analytics et Power BI Desktop (Partie 4)

On sélectionne toutes les tables créées et on clique sur charger. (Figure 140)

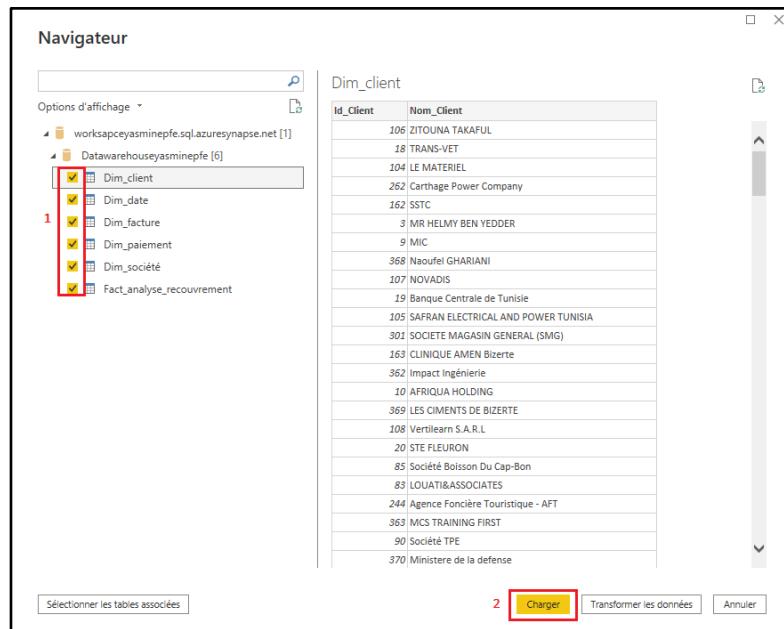


Figure 140 : Connexion entre Azure Synapse Analytics et Power BI Desktop (Partie 5)

Nous établissons ensuite les relations entre la table de faits et les dimensions. (Figure 141)

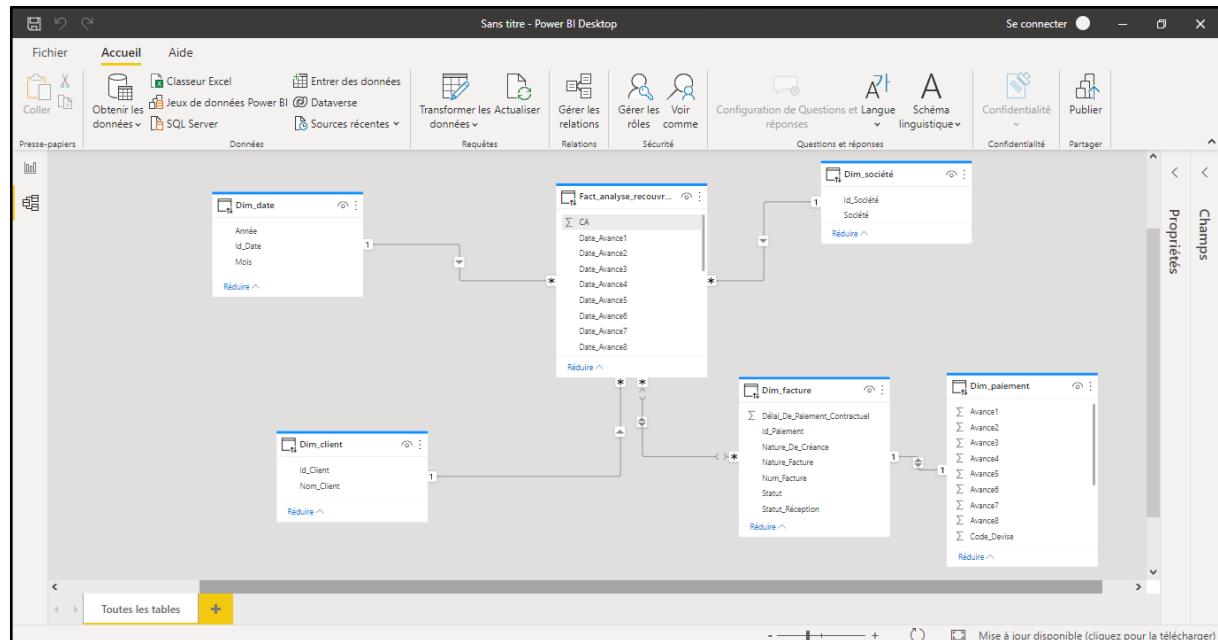


Figure 141 : Connexion entre Azure Synapse Analytics et Power BI Desktop (Partie 6)

6.2 Tableaux de bord

Voici le premier tableau de bord (Figure 142) intitulé "Recouvrement des créances : Indicateurs" demandé par l'entreprise pour visionner les objectifs mensuels. Sur le tableau de bord figurent des jauge ainsi que des indicateurs qui affichent les chiffres clés demandés par RFC. Sur la gauche, nous pouvons voir trois filtres. Le premier filtre permet de sélectionner la société, le deuxième filtre permet de sélectionner le mois et le troisième filtre permet de sélectionner l'année. Nous allons maintenant analyser les graphiques chacun à son tour.

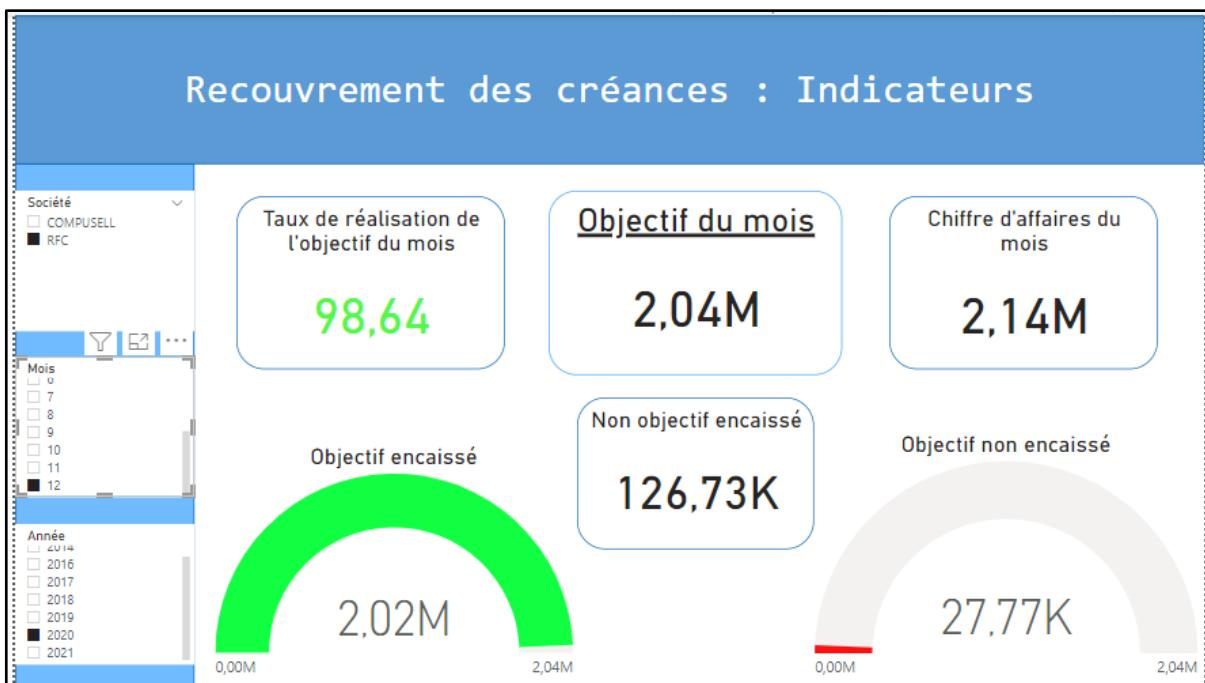


Figure 142 : Premier tableau de bord (Partie 1)

1 - Le premier graphique est une carte (comme nommée sur power BI) qui indique le taux de réalisation de l'objectif du mois. En d'autres termes, c'est la mesure KPI. Ce taux est calculé comme suit : $(\text{Objectif encaissé}/\text{Objectif du mois}) * 100$. Nous avons utilisé la mise en forme conditionnelle. Si le taux est entre 0 et 49, il sera affiché en rouge pour indiquer que ce taux n'est pas convenable. Si le taux est égal ou supérieur à 50, le taux sera affiché

en vert pour indiquer que le taux est convenable. Le taux de réalisation pour le mois de décembre 2020 est de 98,64%. (Figure 143)

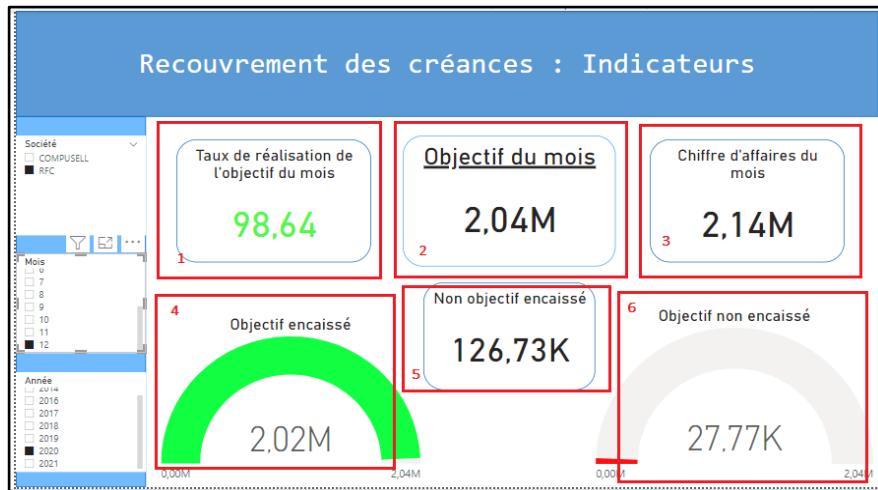
2- Le deuxième graphique est une carte (comme nommée sur power BI) qui montre l'objectif du mois en Millions de dinars. L'objectif du mois est calculé comme suit : Objectif encaissé + Objectif non encaissé. Nous pouvons apercevoir que l'objectif du mois de décembre 2020 est de 2.04 Millions de dinars pour la société RFC. (Figure 143)

3- Le deuxième graphique est une carte (comme nommée sur power BI) qui montre le chiffre d'affaires du mois en Millions de dinars. Le chiffre d'affaires est calculé comme suit : Objectif encaissé + Non Objectif encaissé. Nous pouvons apercevoir que le chiffre d'affaires de décembre 2020 est de 2.14 Millions de dinars pour la société RFC. Le chiffre d'affaires dépasse donc l'objectif du mois mis en place par l'entreprise. (Figure 143)

4- Le quatrième graphique est une jauge qui indique l'objectif encaissé du mois par rapport à l'objectif du mois. Pour le mois de décembre 2020, nous pouvons apercevoir que 2,02 Millions de dinars ont été encaissés par rapport aux 2,04 Millions attendus pour RFC. (Figure 143)

5- Le cinquième graphique est une carte (comme nommée sur power BI) qui montre le non objectif encaissé du mois. Nous pouvons apercevoir que le non objectif encaissé de décembre 2020 est de 126730 dinars pour la société RFC. Nous tenons à rappeler que le non objectif encaissé du mois représente les factures qui doivent être payées durant les mois suivants mais qui ont été payées avant leur mois d'échéance. (Figure 143)

6- Le sixième graphique est une jauge qui indique l'objectif non encaissé du mois par rapport à l'objectif du mois. Pour le mois de décembre 2020, nous pouvons apercevoir que 27770 dinars n'ont pas été encaissés pour la société RFC alors qu'ils auraient dû être encaissés ce mois-là. (Figure 143)

**Figure 143 : Premier tableau de bord (Partie 2)**

Le deuxième tableau de bord (Figure 144) intitulé “Liste des factures payées” n'affiche qu'un seul graphique, une table, qui affiche toutes les factures payées. Existent aussi des filtres pour voir les factures payées par société, mois et année. Nous pouvons ici apercevoir la liste des factures payées de la société RFC pour le mois d'Octobre 2021.

**Figure 144 : Deuxième tableau de bord**

Le troisième tableau de bord (Figure 145) intitulé “Liste des factures impayées” n'affiche qu'un seul graphique, une table, qui affiche toutes les factures impayées. Existent aussi des

filtres pour voir les factures impayées par société, mois et année. Nous pouvons ici apercevoir la liste des factures qui n'ont pas encore été payées de la société RFC pour le mois d'Août 2021.

Liste des factures impayées

Num_Facture	Nom_Client	Société	Statut
FA21/0237	MICROSOFT TUNISIA SARL	RFC	Encours
FA21/0238	MICROSOFT TUNISIA SARL	RFC	Encours
FA21/0239	MICROSOFT TUNISIA SARL	RFC	Encours
FA21/0240	MICROSOFT TUNISIA SARL	RFC	Encours
FA21/0251	DZETA	RFC	Encours
FA21/0354	Banque Centrale de Tunisie	RFC	Encours

Figure 145 : Troisième tableau de bord

Le quatrième tableau de bord intitulé "Chiffre d'affaires : Indicateurs" n'affiche qu'un seul graphique, une table, qui affiche le chiffre d'affaires et le pourcentage du chiffre d'affaires par client. Existent aussi des filtres pour voir le chiffre d'affaires contribué par chaque client par société, mois et année. Nous pouvons apercevoir que pour la société RFC et que pour le mois de Mars 2020, cinq clients ont contribué à plus de 10% au chiffre d'affaires du mois. Zitouna takaful se place en première place avec 15,21% du chiffre d'affaires du mois de Mars 2020 (soit 166730,69 dinars), Société magasin général se positionne en deuxième avec 14,74% (soit 161562,58 dinars) ensuite viennent STIAL Délice danone (13,32% soit 135017,25 dinars), STK (11,46% soit 125642,83 dinars) et Monogros (10,89% soit 119427,90 dinars).

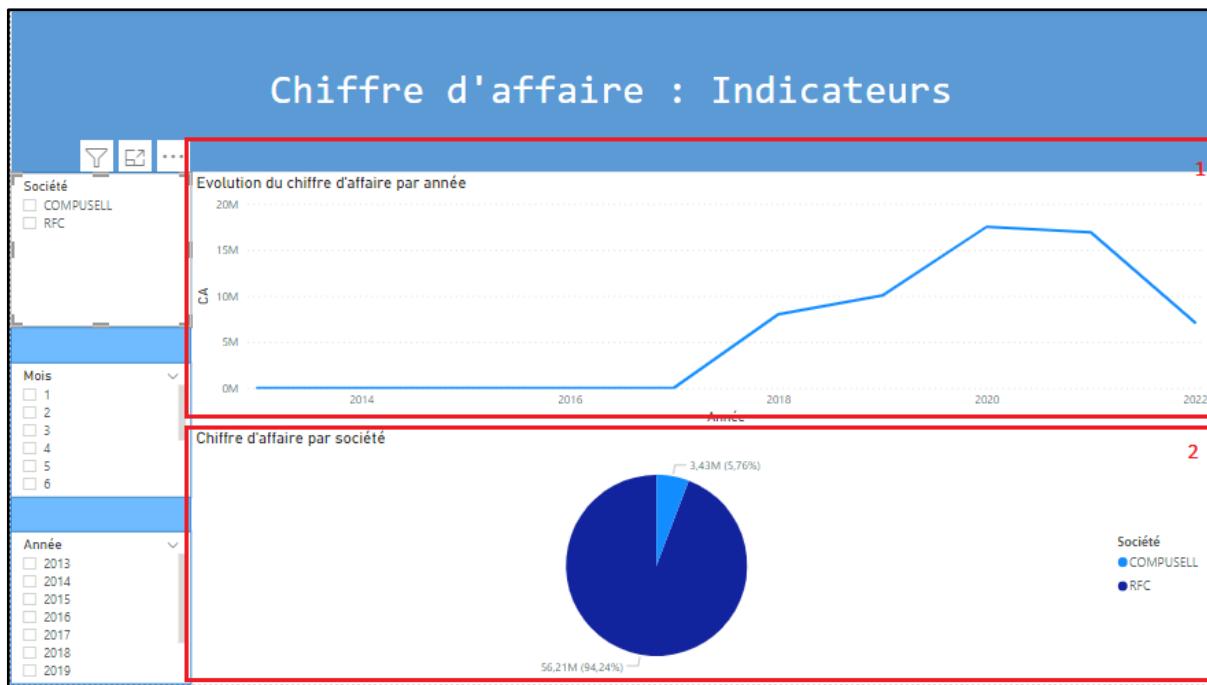
Chiffre d'affaire : Indicateurs			
Société	Nom_Client	CA	%TG CA
COMPUSELL	ZITOUNA TAKAFUL	166 730,69	15,21%
RFC	SOCIETE MAGASIN GENERAL (SMG)	161 562,58	14,74%
	S.T.I.A.L DELICE DANONE	135 017,25	12,32%
	BTK	125 642,83	11,46%
	MONOGROS	119 427,90	10,89%
	PI Services	76 757,95	7,00%
	UNIMED	46 158,73	4,21%
	Bouchamaoui Industrie	45 244,76	4,13%
	ATL	42 388,40	3,87%
	ETANCHÉITE TUNISIENNE	23 090,76	2,11%
	Société MARAM COMPANY	20 449,60	1,87%
	Microsoft Ireland Operations Limited (Ltd)	16 660,00	1,52%
	ECUMED PETROLEUM ZARZIS, LTD	15 165,00	1,38%
	MAGHREB SOLUTIONS	13 970,14	1,27%
	BANQUE ZITOUNA	8 925,60	0,81%
	TPR	8 519,81	0,78%
	AXIS CAPITAL GROUPE BMCE	8 021,10	0,73%
	UAB IARDT	7 500,00	0,68%
	UAB VIE	7 500,00	0,68%
	WIC	7 210,00	0,65%
	Total	1 096 285,89	100,00%

Figure 146 : Quatrième tableau de bord

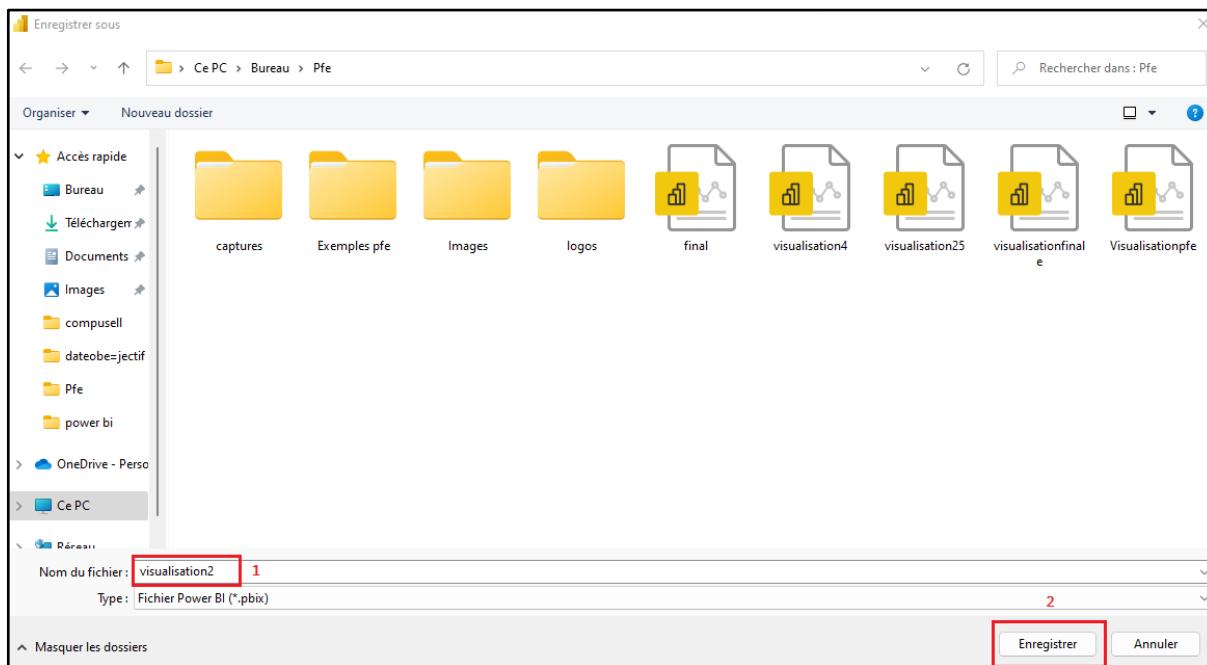
Le cinquième et dernier tableau de bord (Figure 147) aussi intitulé “ Chiffre d'affaires : Indicateurs” affiche deux graphiques et existent à gauche des filtres par société, mois et année.

1- Le premier graphique est un graphique en courbes qui affiche l'évolution du chiffre d'affaires par année. Nous pouvons, grâce au filtre société, voir l'évolution du chiffre d'affaires pour chaque société soit RFC et Compusell. Nous pouvons apercevoir que le chiffre d'affaires total augmente jusqu'à 2020 pour atteindre environ les 17 Millions de dinars pour ensuite baisser en 2021 et encore plus en 2022. La baisse en 2022 s'explique par le fait que l'année n'est pas encore achevée. (Figure 147)

2- Le deuxième graphique est un graphique en secteurs qui affiche le chiffre d'affaires encaissé par société. Existent aussi des filtres pour voir le chiffre d'affaires encaissé par chaque société par mois et année. (Figure 147)

**Figure 147 : Cinquième tableau de bord**

Après avoir complété toutes les visualisations, nous allons enregistrer le fichier sous le nom “visualisation2”. (Figure 148)

**Figure 148 : Enregistrement des tableaux de bord**

6.3 Publication des tableaux de bord sur Azure Synapse

Après avoir connecté Power BI Desktop à notre compte Power BI en ligne grâce à l'adresse mail, nous allons cliquer sur publier, ensuite sur le Workspace "workspaceyasmine" créé auparavant sur la plateforme Power BI en ligne qui est à son tour connectée à Azure Synapse et nous allons cliquer sur sélectionner. (Figure 149)

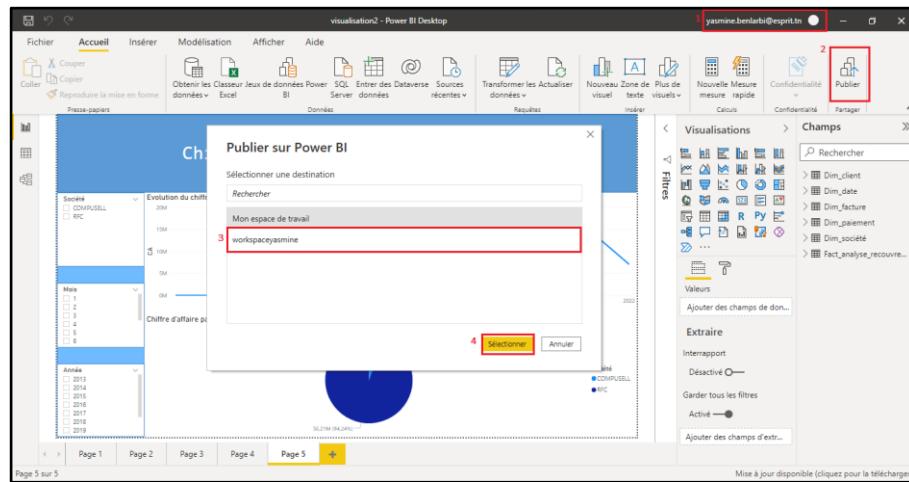


Figure 149 : Publication des tableaux de bord sur Azure Synapse (Partie 1)

Une fenêtre apparaît et nous allons à présent cliquer sur ouvrir “visualisations2.pbix” dans Power BI. (Figure 150)

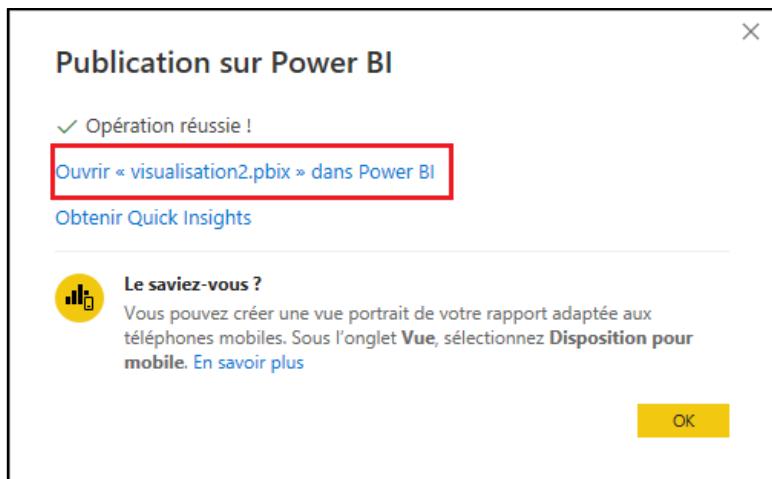


Figure 150 : Publication des tableaux de bord sur Azure Synapse (Partie 2)

Nous pouvons remarquer que notre fichier a été publié dans notre espace de travail sur la plateforme en ligne. (Figure 151)

Nom	Type	Propriétaire	Actualisé	Prochaine actualisation	Approbation
visualisation2	Report	workspaceyasmine	22/07/22 14:05:27	—	—
visualisation2	Jeu de données	workspaceyasmine	22/07/22 14:05:27	Non applicable	—

Figure 151 : Publication des tableaux de bord sur Azure Synapse (Partie 3)

Dans Azure Synapse, on se dirige vers le hub “Développer” puis on clique sur Power BI ensuite sur le service lié yasminepfepowerbi puis on clique sur Rapports Power BI. Nous pouvons remarquer que le fichier “visualisation2” s'y trouve. On clique dessus et nous pouvons maintenant accéder à nos visualisations depuis Azure Synapse Analytics. Toute personne qui

a accès au compte peut à présent apporter les modifications nécessaires qu'elle veut aux tableaux de bord et peut les visionner. (Figure 152)

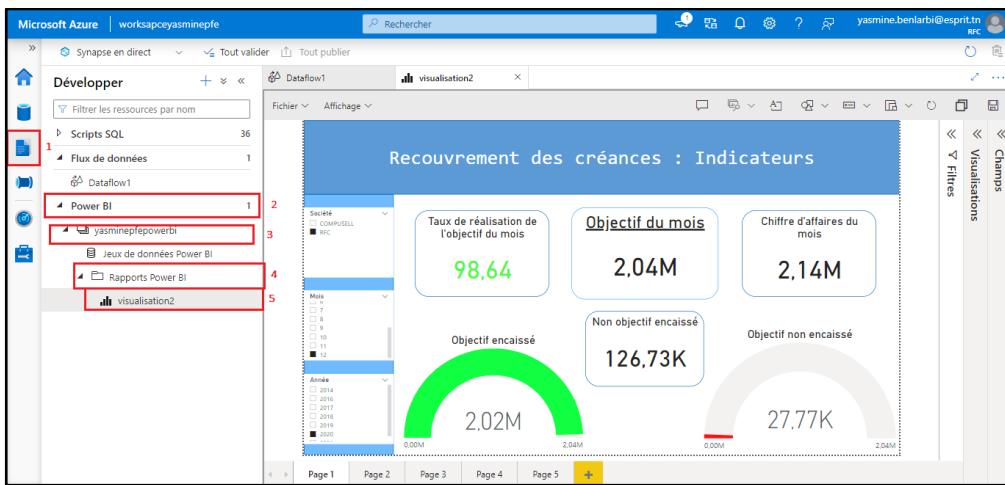


Figure 152 : Publication des tableaux de bord sur Azure Synapse (Partie 4)

Conclusion

Pour conclure, on a utilisé un pool SQL dédié pour la construction de l'entrepôt de données. On s'est également servi de l'application logique pour l'hébergement du fichier source à notre plateforme ainsi que le processus d'automatisation. Nous avons utilisé les dataflows pour les transformations et l'alimentation du Data Warehouse. Concernant les visualisations, nous avons utilisé Power BI pour la réalisation de nos tableaux de bord.

Conclusion générale

Afin d'améliorer la gestion financière de l'entreprise, nous avons réalisé ce projet qui consiste à développer une solution d'aide à la prise de décision. L'objectif est de valoriser l'entrepôt de données en injectant une solution décisionnelle et en élaborant des tableaux de bord dynamiques contenant tous les indicateurs de performances qui répondent à tous les critères pour le suivi de la rentabilité des clients ainsi que le recouvrement des créances mais aussi pour visualiser les objectifs du mois.

De ce fait, nous avons utilisé la méthodologie Kimball. Nous avons défini les besoins de notre organisme d'accueil. Ensuite, nous avons construit à l'aide d'un pool SQL dédié, la table des faits et ses tables de dimensions en se basant sur la conception préétablie de l'entrepôt de données. En utilisant une application logique du Portail Azure, nous avons hébergé le fichier source du Team qui se trouvait sur la plateforme Teams vers Azure synapse qui est la plateforme d'analyse utilisée dans le cadre de notre projet. Nous avons notamment automatisé le processus de génération du fichier source vers la plateforme. Nous avons ensuite transformé les données en ajoutant des identifiants, en calculant les objectifs grâce aux dataflows. Nous avons ensuite alimenté l'entrepôt de données en indiquant les paramètres de mappages nécessaires pour chaque table. Finalement, nous avons créé des tableaux de bord simples et dynamiques.

Grâce à nos tableaux de bord, le département Finance peut maintenant se permettre d'avoir une vue d'ensemble sur la réalisation des objectifs mensuels dans le but d'améliorer les indicateurs de performances de gérer le recouvrement des créances.

La solution développée a répondu aux attentes de la société d'accueil. Cependant, ce projet peut être optimisé et généralisé à d'autres départements avec d'autres paramètres spécifiques à chaque département.

Webographie

Indice	Lien	Date de dernière visite
[1]	https://www.rfc.com.tn/apropos.php	20/06/2022
[2]	https://www.talend.com/fr/resources/guide-business-intelligence/	30/06/2022
[3]	https://www.lebigdata.fr/olap-online-analytical-processing	15/07/2022
[4]	https://www.petite-entreprise.net/P-3081-136-G1-definition-oltp-vs-olap.html	10/07/2022
[5]	https://www.advancia-itsystem.com/site/fr/news_details.php?id_article=19&id_news=139	12/07/2022
[6]	https://www.lebigdata.fr/datamart-definition	22/07/2022
[7]	https://www.lebigdata.fr/data-warehouse-entrepot-donnees-definition	20/07/2022
[8]	https://www.cartelis.com/blog/data-warehouse-modelisation-etoile/	20/07/2022
[9]	https://www.talend.com/fr/resources/guide-etl/	10/07/2022

[10]	https://www.piloter.org/projet/methode/scrum.htm	15/07/2022
[11]	https://www.piloter.org/mesurer/methode/methode-GIMSI-phases.htm	15/07/2022

Résumé :

Afin d'améliorer la gestion financière de l'organisme d'accueil, nous avons réalisé ce projet qui consiste à développer une solution d'aide à la prise de décision. L'objectif est de valoriser l'entrepôt de données en injectant une solution décisionnelle et en élaborant des tableaux de bord dynamiques contenant tous les indicateurs de performances qui répondent à tous les critères pour le suivi du recouvrement des créances. Nous avons utilisé Logic Apps pour automatiser le processus d'extraction des données, Azure Synapse pour les transformations et Power BI pour la mise en place des tableaux de bord.

Mots-clés :

Business Intelligence, tableaux de bord, ETL, Azure Synapse Analytics, Power BI, Modélisation, Datawarehouse, Données, Conception.

Abstract :

In order to improve the financial management of the host organization, we carried out this project, which consists in developing a solution to assist decision-making. The objective is to enhance the data warehouse by injecting a decision-making solution and developing dynamic dashboards containing all the performance indicators that meet all the criteria for the monitoring of debt collection. We used Logic Apps to automate the data extraction process, Azure Synapse for transformations and Power BI for dashboard implementation.

Key words :

Business Intelligence, dashboards, ETL, Azure Synapse Analytics, Power BI, conceptual modelling, Data Warehouse, Data, Conception.