PREMIER MINISTRE

Liberté
Égalité
Fraternité

Secrétariat général du Gouvernement
Direction des services
administratifs et financiers

Annexe 1 au cahier des clauses techniques particulières

-

Extraits du DAT (draft) de la plateforme SIGMAé

Le présent document présente la liste du périmètre applicatifs à prendre en compte pour le marché de TMA

Table des matières

Lexique 4
1.1 Acronymes 4
1.2 Termes du projet 5
2. Introduction 7
2.1 Objet du document 7
2.2 Documents de référence 8
3. Contexte et exigences 9
3.1 Contexte du projet 9
3.2 Populations et canaux 9
3.3 Exigences non fonctionnelles 10
3.4 Contraintes 10
4. Architecture fonctionnelle 12
4.1 Acteurs du système 12
4.1.1 Acteurs humains 12
4.1.2 Acteurs systèmes 13
4.2 Cas d'utilisation du socle transverse 13
4.2.1 Annonceur 13
4.2.2 Gestionnaire et administrateur 14
4.2.3 Application partenaire 14
4.2.4 Administrateur technique 14
4.3 Cas d'utilisation des instances OS-OP et PILA 14
4.3.1 OS-OP : dépôt des comptes des organisations syndicales et professionnelles 15
4.3.2 PILA : production des annonces des associations 16
5. Architecture applicative 17
5.1 Vue d'ensemble 17
5.2 Vue à date 18
5.3 Deux canaux 19
5.4 Composants applicatifs 19
5.4.1 Frontends 19
5.4.2 Passerelles BFF (Backend For Frontend) 20
5.4.3 Serveurs d'autorisation 20
5.4.4 Microservices 20
5.5 Contrats d'interface 22
6. Architecture technique 23
6.1 Zones réseau 23
6.2 Topologie de déploiement 23
6.3 Images de conteneurs livrées 24
6.4 Les deux instances du serveur d'autorisation 24
6.5 Matrice des flux 25
6.6 Haute disponibilité 26
7. Architecture de sécurité 28
7.1 Authentification 28
7.2 Sessions et tokens 29
7.3 Autorisation 30
7.4 Cloisonnement des zones 31
7.5 Gestion des secrets et des clés 31
7.6 Flux machine à machine 31
7.6.1 Applications partenaires 32
7.6.2 Clients internes 32

7.7 Protection active contre les menaces 32
8. Architecture de données 34
8.1 Bases par domaine 34
8.2 Stockage objet 35
8.3 Registres OAuth2 par zone 35
8.4 Schémas, rôles et privilèges 35
8.5 Réplicas de lecture 36
8.6 Migrations 36
9. Exploitation 37
9.1 Démarrage et ordonnancement 37
9.2 Sauvegarde et restauration 37
9.3 Observabilité 37
9.4 Supervision 37

LEXIQUE

Termes et acronymes utilisés dans ce document.

1.1 Acronymes

Sigle	Signification
ANSSI	Agence nationale de la sécurité des systèmes d'information
API	Interface de programmation applicative (Application Programming Interface)
BALO	Bulletin des annonces légales obligatoires
BFF	Backend For Frontend : passerelle dédiée à un frontend, qui porte la session de l'utilisateur et détient ses jetons à la place du navigateur
BO / FO	Back-office (canal des agents) / front-office (canal des usagers)
BOAMP	Bulletin officiel des annonces des marchés publics
BODACC	Bulletin officiel des annonces civiles et commerciales
CCTP	Cahier des clauses techniques particulières
CSP	Content Security Policy : politique de sécurité des contenus déclarée aux navigateurs
DAT	Document d'architecture technique
DGT	Direction générale du travail
DILA	Direction de l'information légale et administrative
DIS	Département des infrastructures et des services (DILA)
DREETS	Direction régionale de l'économie, de l'emploi, du travail et des solidarités
DSFR	Système de Design de l'État Français
DSIEG	Département des systèmes d'informations économiques et de gestion (DILA)
HSTS	HTTP Strict Transport Security : en-tête imposant aux navigateurs l'usage exclusif de HTTPS
IdP	Identity Provider, fournisseur d'identité : système qui authentifie réellement la personne
JOAFE	Journal officiel des associations et fondations d'entreprise
JWT	JSON Web Token : jeton signé, autoporteur, validable sans appel à son émetteur
OIDC	OpenID Connect : protocole d'authentification bâti sur OAuth2
OS-OP	Organisations syndicales et professionnelles ; désigne aussi l'application de dépôt de leurs comptes annuels
PILA	Application de production des annonces des associations, publiées au JOAFE
RGAA	Référentiel général d'amélioration de l'accessibilité
RGPD	Règlement général sur la protection des données
RNA	Répertoire national des associations
RSSI	Responsable de la sécurité des systèmes d'information
SAML	Security Assertion Markup Language : protocole de fédération d'identité, utilisé par le SSO interne de la DILA
SIGMAé	Socle informatique de gestion mutualisée des annonces économiques
SIREN / SIRET	Identifiants des entreprises (unité légale) et de leurs établissements au répertoire de l'INSEE
SLA	Service Level Agreement : niveau de service contractualisé

Sigle	Signification
SRI	Subresource Integrity : contrôle d'intégrité des ressources chargées par une page web
SSO	Single Sign-On : authentification unique
TLS	Transport Layer Security : chiffrement des communications réseau
US	User story : expression d'un besoin fonctionnel du point de vue de l'utilisateur, unité de spécification et de réalisation du projet

1.2 Termes du projet

Terme	Définition
Agent	Personne travaillant pour la DILA ou l'un de ses partenaires institutionnels, qui utilise le back-office depuis le réseau interne. Provisionné par un administrateur, jamais auto-inscrit.
Annonceur	Usager externe, personne physique rattachée à une personne morale, qui dépose et suit ses annonces depuis internet. Entre dans le système par auto-inscription.
Application partenaire	Système tiers qui appelle les API de la plateforme sans utilisateur humain, authentifié en machine à machine par son compte de service OAuth2.
Canal	La voie par laquelle un service est atteint de bout en bout : front-office (usagers, zone internet) ou back-office (agents, zone intranet).
Client OAuth2	Une application déclarée dans un registre, identifiée par un identifiant client et authentifiée par un secret : les BFF, les comptes de service internes et les applications partenaires.
Compte de service	Client OAuth2 utilisé par un système (composant interne ou application partenaire) pour obtenir un jeton technique, sans utilisateur humain.
Fonctionnalité	L'unité d'autorisation fine, du grain de l'action métier. Un profil accorde un ensemble de fonctionnalités ; les API les contrôlent individuellement.
Gestionnaire	Agent qui instruit les dossiers d'un flux d'annonces pour son entité professionnelle. Se décline en administrateur fonctionnel (valide les habilitations de son entité) et gestionnaire opérationnel.
Identité fédérée	Le couple (fournisseur d'identité, identifiant chez ce fournisseur) qui désigne une personne. Plusieurs identités fédérées peuvent être rattachées à un même compte.
Instance applicative	Un type d'annonce instancié sur la plateforme (OS-OP, PILA…), s'appuyant sur les fonctions transverses du socle.
Issuer	L'URL qui identifie l'émetteur d'un jeton. Chaque zone a le sien : un service qui valide les jetons d'un issuer n'accepte pas ceux de l'autre.
Profil	Le rôle métier attribué à un compte (annonceur, gestionnaire, administrateur…). Un profil se traduit en fonctionnalités, elles-mêmes traduites en scopes.
Registre (OAuth2)	La base des clients OAuth2 connus d'une instance du serveur d'autorisation, propre à chaque zone. Ce qu'un registre ne contient pas compte autant que ce qu'il contient : un client absent du registre d'une zone ne peut pas y obtenir de jeton.
Resource server	Microservice qui expose une API protégée et valide le jeton qu'il reçoit. Sans état : aucune session côté serveur.

Terme	Définition
Scope	La porte d'entrée d'un microservice, à raison d'un service pour un scope. Jamais demandé par le client : calculé par le serveur à partir des profils du porteur du jeton.
SecNumCloud	Qualification délivrée par l'ANSSI aux offres d'informatique en nuage de confiance.
Serveur d'autorisation	Le composant qui émet les jetons de la plateforme, une instance par zone. Il joue un rôle de broker d'authentification : il ne stocke aucune identité, délègue l'authentification à un IdP, résout le compte correspondant auprès de l'annuaire, puis émet un jeton interne.
Usager	Personne physique qui utilise les services en ligne depuis internet, côté front-office. Pour la plateforme, l'annonceur est l'usager type.
Zone	Découpage réseau de la plateforme en deux valeurs : internet (zone exposée, usagers) et intranet (zone interne, agents). Chaque zone a sa passerelle, son serveur d'autorisation, son issuer, sa clé de signature et son registre de clients.

2. INTRODUCTION

2.1 Portée et caractère évolutif des informations techniques

L'ensemble des informations techniques présent dans cette annexe sont fournis à titre informatif et reflètent l'état du système d'information à la date de rédaction du dossier de consultation.

Le titulaire est informé que l'environnement technique, les composants logiciels, les infrastructures, les versions des produits, les interfaces ou les modalités d'exploitation peuvent évoluer entre la publication du marché et le démarrage effectif des prestations. En conséquence, certains éléments figurant dans les DAT sont susceptibles d'être modifiés, complétés ou remplacés, y compris de manière substantielle.

Ces évolutions ne remettent pas en cause le périmètre général des prestations attendues. Le titulaire devra prendre en compte les informations et documentations actualisées qui lui seront communiquées lors de la phase de démarrage et, le cas échéant, adapter son organisation et ses méthodes d'intervention en conséquence.
Pour la préparation de son offre, le titulaire est réputé avoir pris en compte le caractère évolutif de l'environnement technique décrit dans les documents contractuels.

2.2 Objet du document

Le présent document constitue le document d'architecture technique (DAT) de la plateforme SIGMAé (Socle Informatique de Gestion Mutualisée des Annonces économiques), réalisée pour le compte de la Direction de l'information légale et administrative (DILA).

La DILA assure la collecte, la transformation et la diffusion des annonces économiques (BODACC, BOAMP, BALO, JOAFE, dépôts de comptes, etc.). Ces missions reposent aujourd'hui sur huit applications développées au fil des besoins, sur des technologies et des logiques fonctionnelles hétérogènes : leur obsolescence technique rend les évolutions coûteuses et complique l'homologation de sécurité. La plateforme SIGMAé est la cible de cette refonte : une plateforme unifiée et modulaire, sur un ensemble technologique cohérent, offrant des fonctions transverses communes sur lesquelles chaque type d'annonce est instancié comme une application de la plateforme, en remplacement progressif des applications existantes.

Le présent DAT est produit dans le cadre du marché de conception, développement, déploiement et maintenance du socle initial de cette plateforme et de ses deux premières instances applicatives (refonte des applications OS-OP et PILA).

Il décrit l'architecture de la plateforme telle qu'elle est conçue et réalisée : composants applicatifs et leurs responsabilités, topologie de déploiement, mécanismes d'authentification et d'autorisation, organisation des données et principes d'exploitation.

Il s'adresse aux architectes et aux équipes techniques de la DILA (notamment DSIEG, DIS et RSSI), aux équipes de réalisation, ainsi qu'à tout intervenant amener à auditer, exploiter, maintenir ou faire évoluer la plateforme, y compris dans le cadre d'une réversibilité.

2.3 Documents de référence

Référence	Document	Version / date
CCTP SIGMAé	Cahier des clauses techniques particulières du marché « Conception, développement, déploiement et maintenance du socle initial de la plateforme SIGMAé et de deux premières instances applicatives »	v1.62 du 29/04/2026
DSFR	Système de Design de l'État Français	version courante
RGAA	Référentiel général d'amélioration de l'accessibilité	version 4
ProConnect	Documentation partenaires ProConnect	en ligne
ANSSI	Guides et recommandations de l'ANSSI (état de l'art sécurité)	en ligne

3. CONTEXTE ET EXIGENCES

3.1 Contexte du projet

La DILA opère aujourd'hui huit applications de collecte, transformation et diffusion des annonces économiques (OS-OP, PILA, DCA, eBALO, Successions, STILA_BODACC, eARP, AMF), représentant environ quatre millions d'annonces publiées par an. Développées au fil des besoins sur des socles hétérogènes, elles présentent une obsolescence technique qui renchérit les évolutions et complique l'homologation de sécurité.

Le projet SIGMAé refond cet ensemble en une plateforme unifiée et modulaire, découpée en quatre lots. Le présent marché porte le lot n°1 : la constitution du socle de la plateforme générique et la refonte des deux premières applications, OS-OP (dépôt des comptes annuels des organisations syndicales et professionnelles) et PILA (production des annonces des associations), dont la mise en service est visée pour début 2027 au regard de contraintes réglementaires. Les lots suivants porteront la refonte progressive des autres applications, sur le même socle.

La conception privilégie une architecture orientée services, implémentée en microservices, dont le découpage technique reste lisible au regard de la modularité fonctionnelle. Deux principes du cahier des charges guident ce découpage : ne pas fragmenter prématurément en applications distinctes, et conserver un stockage propre à chaque type d'annonce plutôt qu'un méta-modèle commun. Le socle décrit dans ce document (authentification, comptes, annuaire, référentiel) est la première concrétisation de cette plateforme ; il est conçu pour servir tous les types d'annonces qui y seront instanciés.

3.2 Populations et canaux

La plateforme sert trois populations, chacune par un canal d'accès distinct :

Population	Canal	Authentification
Annonceurs : usagers externes, personnes physiques rattachées à une personne morale, qui déposent et suivent leurs annonces	Zone internet, via le frontend usager	ProConnect, obligatoire pour les annonceurs disposant d'un SIRET ; compte de plateforme avec second facteur (à définir) pour les autres
Gestionnaires et administrateurs : agents de la DILA et de ses partenaires institutionnels (ministères, DREETS…), qui instruisent les dossiers et administrent la plateforme	Zone intranet, via le frontend agent	ProConnect ou SSO interne de la DILA
Applications partenaires : systèmes externes qui échangent avec la plateforme en machine à machine	Accès direct aux API, sans interface	Compte de service OAuth2 propre à chaque application

Les gestionnaires se déclinent en administrateurs fonctionnels et gestionnaires opérationnels, rattachés à une entité professionnelle ; les administrateurs techniques de la DILA assurent le

support fonctionnel et l'exploitation. La structure de profils, les rôles et les droits sont détaillés au chapitre Architecture fonctionnelle.

3.3 Exigences non fonctionnelles

Sécurité by design. La plateforme est conçue conformément à l'état de l'art de la sécurité des systèmes d'information et aux guides de l'ANSSI. Pour les interfaces web, le cahier des charges impose notamment : TLS systématique avec HSTS recommandé, protection des cookies de session (HttpOnly, Secure, SameSite), politique de sécurité des contenus (CSP, SRI) et en-têtes de sécurité des navigateurs, publication d'un point de contact via /.well-known/security.txt, absence d'adhérence à des modules ou technologies propriétaires côté navigateur. Tout changement d'infrastructure de gestion des données est soumis à une double validation des équipes infrastructure (DIS) et sécurité (RSSI) de la DILA.

Authentification et moindre privilège. L'authentification des annonceurs disposant d'un SIRET passe obligatoirement par ProConnect ; les annonceurs qui n'en disposent pas (une infime minorité) se créent un compte sur la plateforme, protégé par identifiant, mot de passe et un second facteur dont les modalités restent à définir. L'authentification des agents passe par ProConnect ou par le SSO interne de la DILA. Les droits sont attribués selon le principe du moindre privilège, avec une distinction stricte entre les accès des utilisateurs et ceux des comptes techniques.

Traçabilité et audibilité. La plateforme fournit aux administrateurs techniques des fonctions d'audit et de contrôle des accès de l'ensemble des utilisateurs, administrateurs compris. Les journaux correspondants sont protégés, non modifiables, et permettent de retracer une session utilisateur : jalons temporels, fonctionnalités accédées, opérations déclenchées.

Disponibilité. La plateforme reprend a minima les niveaux de service des applications existantes ; le calibrage précis des SLA est mené avec les architectes et le RSSI de la DILA, et sa traduction dans l'architecture est validée par ces mêmes acteurs.

Accessibilité et conception. Les interfaces destinées aux utilisateurs externes sont conformes au DSFR pour la conception et au RGAA pour l'accessibilité. Les interfaces internes peuvent faire l'objet d'une conformité partielle, sur validation de l'équipe projet.

Protection des données personnelles. La plateforme respecte le RGPD : collecte limitée aux données nécessaires, consentement à l'inscription, droit à l'oubli des annonceurs (désactivation du compte et suppression des données personnelles en base active, après validation par un gestionnaire autorisé).

Souveraineté et audibilité des composants. Les composants logiciels retenus, notamment open source ou mis à disposition par l'État, répondent en premier lieu aux exigences de souveraineté, de sécurité et d'audibilité.

3.4 Contraintes

- **Technologies homologuées par la DILA** : Java pour les développements back-end, Angular pour le front-end, PostgreSQL pour les bases de données, Brevo ou le relais SMTP de la DILA pour l'envoi de courriels, solution de signature fournie par la DILA.
- **Hébergement** : les environnements (développement, recette, formation, production) sont mis à disposition et opérés par la DILA sur la plateforme SecNumCloud OUTSCALE. La stratégie de la DILA s'oriente vers la conteneurisation et l'orchestration de conteneurs à horizon 2027 : la plateforme est livrée conteneurisée.

- **Fournisseurs d'identité imposés** : ProConnect pour les annonceurs disposant d'un SIRET et, de façon privilégiée, pour les agents ; le SSO interne de la DILA en alternative pour ces derniers. Ces services sont externes au projet, la plateforme en est consommatrice ; seuls les comptes des annonceurs sans SIRET sont portés par la plateforme elle-même.
- **Cloisonnement réseau** : le SI de la DILA distingue une zone exposée à internet et une zone interne ; les flux initiés depuis l'extérieur vers le SI interne sont interdits. L'architecture de la plateforme reflète ce cloisonnement (une pile d'exposition par zone, détaillée aux chapitres Architecture technique et Architecture de sécurité).
- **Bases de données opérées par la DILA** : les instances PostgreSQL sont fournies et exploitées par la DILA ; la plateforme n'impose ni leur topologie ni leur colocalisation, et chaque domaine de données doit pouvoir être réparti sur des instances distinctes.
- **Calendrier** : la mise en service des applications OS-OP et PILA refondues est fixée à début 2027, précédée d'une période de recette puis de mise en exploitation sécurisée (homologation, audit RGAA, montée en charge, formation).

4. ARCHITECTURE FONCTIONNELLE

Ce chapitre décrit les acteurs du système et leurs cas d'utilisation : ceux du socle transverse (authentification, gestion des comptes, annuaire, référentiel, administration), puis ceux des deux premières instances applicatives, OS-OP et PILA. Le détail des règles de gestion, des écrans et des jalons de traitement relève des US (user stories) du projet.

4.1 Acteurs du système

4.1.1 Acteurs humains

Acteur	Description	Canal
Annonceur	Personne physique rattachée à une personne morale, qui dépose et suit les annonces de celle-ci. Entre dans le système par auto-inscription. Plusieurs annonceurs peuvent être rattachés à une même personne morale ; un annonceur accède aux données déposées par ses prédécesseurs de même profil, pour assurer la continuité de gestion.	Front-office (zone internet)
Gestionnaire opérationnel	Agent, rattaché à une entité professionnelle (DILA, ministère, DREETS…), qui instruit les dossiers des flux d'annonces de son entité.	Back-office (zone intranet)
Administrateur fonctionnel	Agent qui, en plus des capacités d'un gestionnaire opérationnel, gère les habilitations de son entité : il valide la création des gestionnaires et l'attribution de leurs profils. Un administrateur fonctionnel de la DILA peut habiliter les administrateurs fonctionnels des autres entités.	Back-office (zone intranet)
Administrateur technique	Collaborateur de la DILA assurant le support fonctionnel de la plateforme (comptes, paramétrage) ou son exploitation informatique (journaux, supervision, relances).	Back-office et outillage d'exploitation

Deux états particuliers complètent ces acteurs :

- l'utilisateur authentifié non inscrit : une personne qui s'est authentifiée auprès d'un fournisseur d'identité mais n'a pas encore de compte sur la plateforme. Elle n'accède qu'au parcours d'inscription ; aucune donnée n'est écrite tant que l'inscription n'est pas validée ;

- l'agent non habilité : un agent authentifié dont le compte n'est pas provisionné ou ne porte aucun profil. Il est reconnu mais n'accède à aucune fonctionnalité, et est orienté vers une page dédiée l'invitant à se rapprocher de son administrateur.

À l'initialisation de la plateforme, les premiers administrateurs sont provisionnés directement, avant toute connexion.

4.1.2 Acteurs systèmes

Acteur	Description
Application partenaire	Système tiers (publicateur, consommateur institutionnel) qui échange avec la plateforme en machine à machine, authentifié par son compte de service OAuth2, sans utilisateur humain. Chaque application est déclarée dans l'annuaire, rattachée à une zone, et porte un profil applicatif qui détermine ses droits.
Fournisseur d'identité	Service externe qui authentifie les personnes pour le compte de la plateforme : ProConnect (annonceurs et agents) et le SSO interne de la DILA (agents). La plateforme en est consommatrice et ne stocke aucun secret d'authentification des personnes.

4.2 Cas d'utilisation du socle transverse

4.2.1 Annonceur

Cas d'utilisation	Description
Se connecter	S'authentifier via ProConnect (obligatoire dès lors que l'annonceur dispose d'un SIRET). Pour les annonceurs sans SIRET, un compte de plateforme avec second facteur est prévu.
S'inscrire	À la première connexion, créer son compte sur la plateforme : consentement au traitement des données personnelles, enregistrement de l'identité fédérée et attribution du profil annonceur.
Créer un compte sans SIREN	Pour l'annonceur dont l'organisation ne dispose pas d'un SIREN, et qui ne peut donc pas s'authentifier via ProConnect : créer un compte porté par la plateforme elle-même (identifiant, mot de passe et second facteur), puis suivre le même parcours d'inscription.
Consulter son compte	Consulter les informations de son compte.
Consulter les référentiels	Accéder aux nomenclatures utilisées dans les formulaires.

4.2.2 Gestionnaire et administrateur

Cas d'utilisation	Description
Se connecter	S'authentifier via ProConnect ou le SSO interne de la DILA, au choix au moment de la connexion.
Consulter et gérer les usagers	Rechercher un usager, consulter son compte, ses identités et ses profils ; désactiver l'usager.
Consulter et gérer les agents	Provisionner un agent par son adresse professionnelle, attribuer ou retirer ses profils ; désactiver l'agent. Le compte est rattaché à l'identité fédérée de l'agent lors de sa première connexion.
Consulter et gérer les applications partenaires	Déclarer une application partenaire (zone, profil applicatif, propriétaire fonctionnel), ce qui crée conjointement son compte de service OAuth2 ; renouveler son secret (communiqué une seule fois, à la création ou à la rotation) ; désactiver l'application.
Consulter son compte	Consulter les informations de son compte.

La distinction entre gestionnaire opérationnel et administrateur fonctionnel se traduit par les profils attribués : la consultation est ouverte aux deux, la gestion (usagers, agents, applications) est réservée aux profils d'administration. Le modèle d'autorisation qui porte cette distinction (profils, fonctionnalités, scopes) est décrit au chapitre Architecture de sécurité.

4.2.3 Application partenaire

Cas d'utilisation	Description
Obtenir un jeton technique	S'authentifier auprès du serveur d'autorisation de sa zone avec son compte de service et obtenir un jeton d'accès technique, porteur de son profil applicatif.
Appeler les API de sa zone	Consommer les API exposées dans sa zone, dans la limite des droits de son profil applicatif.

4.2.4 Administrateur technique

Les cas d'utilisation de l'administrateur technique relèvent du support et de l'exploitation : gestion des comptes et des habilitations (via les cas d'administration ci-dessus), consultation des journaux, supervision et relance des traitements. Les moyens correspondants sont décrits au chapitre Exploitation.

4.3 Cas d'utilisation des instances OS-OP et PILA

Les deux premières instances refondues s'appuient sur les acteurs et les cas du socle : leurs utilisateurs se connectent, s'inscrivent et sont administrés comme décrit ci-dessus. Les cas suivants reprennent le périmètre fonctionnel des applications existantes, reformulé pour la cible ; leur déclinaison détaillée est portée par les US de chaque instance.

4.3.1 OS-OP : dépôt des comptes des organisations syndicales et professionnelles

Les annonceurs sont les représentants des organisations syndicales ou professionnelles, identifiées par leur SIREN ou, pour celles qui n'en ont pas, par un numéro d'ordre attribué par la DGT. Les gestionnaires relèvent des DREETS (instruction), de la DGT (administration du flux, avec les mêmes fonctions que les DREETS) et de la DILA (administration du processus, visibilité sur l'ensemble des dépôts).

Cas d'utilisation de l'annonceur OS-OP :

Cas d'utilisation	**Description**
Déposer les comptes annuels	Renseigner le formulaire de dépôt (exercice comptable, niveau de ressources, type initial ou rectificatif), joindre les documents comptables en PDF, choisir le mode de consultation (libre ou sur demande ; libre obligatoirement au-delà de 230 000 € de ressources) et valider pour publication. Le SIREN de l'organisation est vérifié auprès du répertoire des entreprises de l'INSEE ; la validation du dépôt déclenche une notification par courriel accompagnée d'un justificatif de dépôt.
Rectifier un dépôt	Déposer un dépôt rectificatif corrigeant un dépôt initial erroné.
Suivre ses dépôts	Consulter l'état de ses dépôts et les dépôts antérieurs de son organisation, y compris ceux effectués par ses prédécesseurs ; télécharger les documents déposés et le justificatif de dépôt.
Gérer les données de son organisation	Mettre à jour les informations de l'organisation.

Cas d'utilisation du gestionnaire OS-OP (DREETS, DGT, DILA) :

Cas d'utilisation	**Description**
Suivre les dossiers	Consulter le tableau de bord de l'état des dossiers et des comptes à contrôler ; rechercher un dossier par SIREN ou numéro d'ordre.
Instruire les demandes de consultation	Pour les comptes en consultation sur demande : valider la demande, la valider en anonymisant le dossier, ou la refuser ; y compris pour les demandes reçues au format papier.
Gérer les comptes des organisations	Créer le compte d'une organisation non inscrite, transférer un compte et ses dépôts vers la DREETS compétente ou vers la DILA.
Consulter les statistiques	Consulter les statistiques des dépôts selon divers critères.

Le dépôt des comptes par une DREETS pour le compte d'une organisation, possible dans l'application actuelle, n'est pas reconduit dans la cible.

4.3.2 PILA : production des annonces des associations

L'instance PILA n'a pas de front-office annonceur : les avis publiés au JOAFE proviennent de flux entrants ou de saisies internes. Environ 90 % des avis (associations loi 1901) arrivent du registre national des associations du ministère de l'intérieur, dont la refonte (Référentiel des associations et des fondations) porte les échanges sur une API au format JSON ; le reste provient des téléprocédures dématérialisées (fondations d'entreprise, fonds de dotation, associations syndicales de propriétaires) et de la saisie des avis reçus au format papier. Les gestionnaires sont internes à la DILA.

Cas d'utilisation du gestionnaire PILA :

Cas d'utilisation	**Description**
Saisir un avis papier	Saisir les avis reçus au format papier (fondations partenariales, fonds de pérennité, avis de suspension des fonds de dotation, avis d'annulation loi 1901). L'orientation étudiée est une saisie dans le téléservice « Mes démarches numériques » (demarche.numerique.gouv.fr) plutôt que dans un écran propre à la plateforme, ces avis rejoignant alors le canal de collecte des téléprocédures.
Gérer les avis	Rechercher un avis, consulter son historique et ses propriétés, modifier ses métadonnées, changer son statut (mise en attente, report, refus ou suppression, selon la provenance de l'avis), prévisualiser le PDF signé.
Gérer les parutions	Administrer le calendrier de parution, constituer le bulletin des avis prêts à publier, le publier, le republier ou le reporter.
Consulter les statistiques	Consulter les statistiques des avis selon divers critères.

Flux entrants machine à machine :

Cas d'utilisation	**Description**
Recevoir les avis du registre des associations	Le registre national des associations transmet les avis loi 1901 à la plateforme via son API.
Récupérer les téléprocédures	La plateforme collecte quotidiennement, via l'API des services de démarches en ligne, les avis saisis par les déclarants et, si cette orientation est retenue, les avis papier saisis par les gestionnaires.

Ces systèmes sont des applications partenaires au sens du socle : authentifiés par compte de service, habilités par leur profil applicatif. Symétriquement, des partenaires institutionnels pourront être habilités à consulter par API les données de gestion de leur flux (le ministère de l'intérieur pour PILA, la DGT pour OS-OP).

5. ARCHITECTURE APPLICATIVE

5.1 Vue d'ensemble

Cette vue d'ensemble est là pour donner une trajectoire. Certains composants qui y figurent ne sont pas encore arrêtés, et notre vision s'ajustera de façon itérative, à mesure de l'avancement du projet.

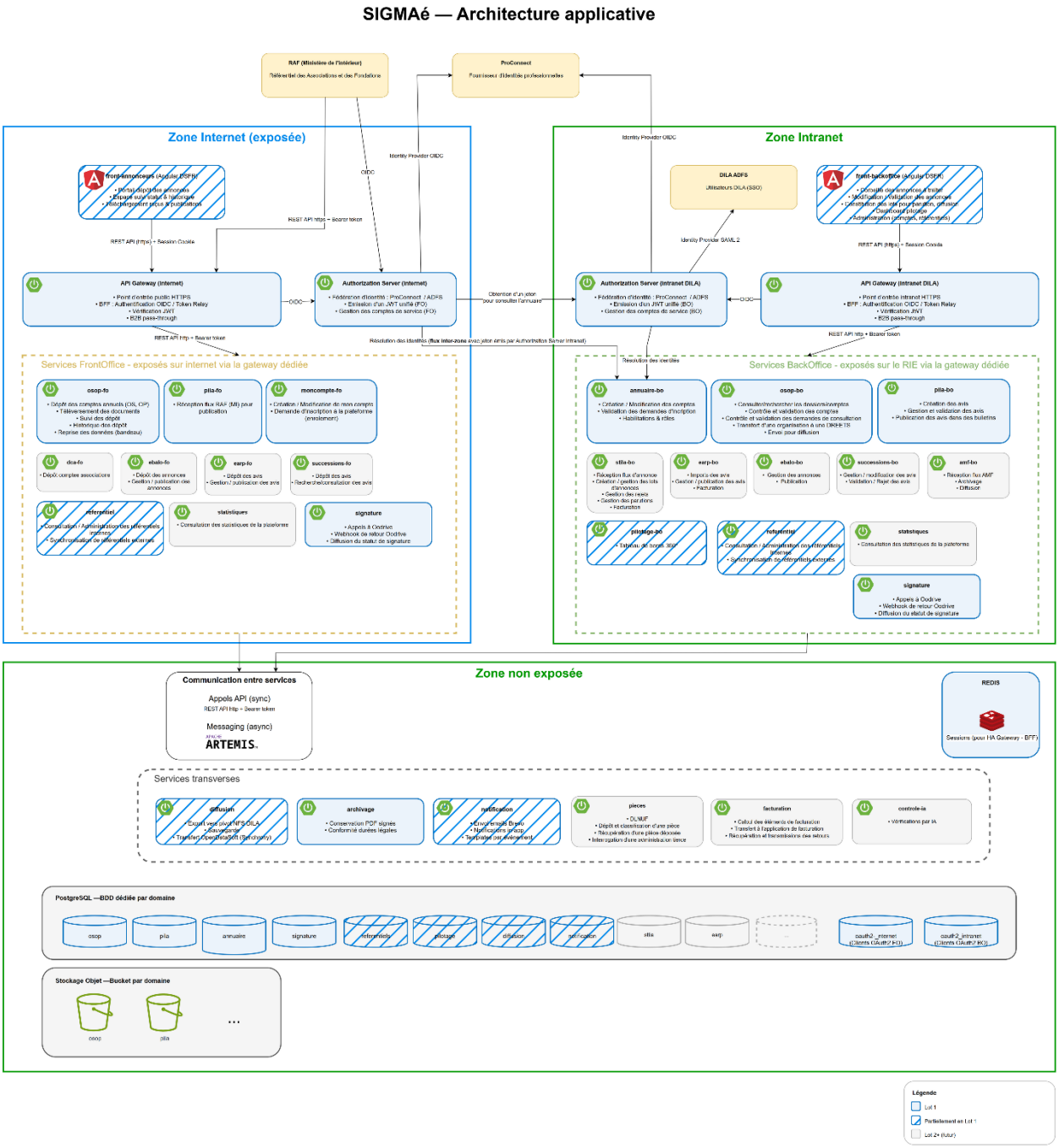


Figure 1 — Architecture applicative cible

5.2 Vue à date

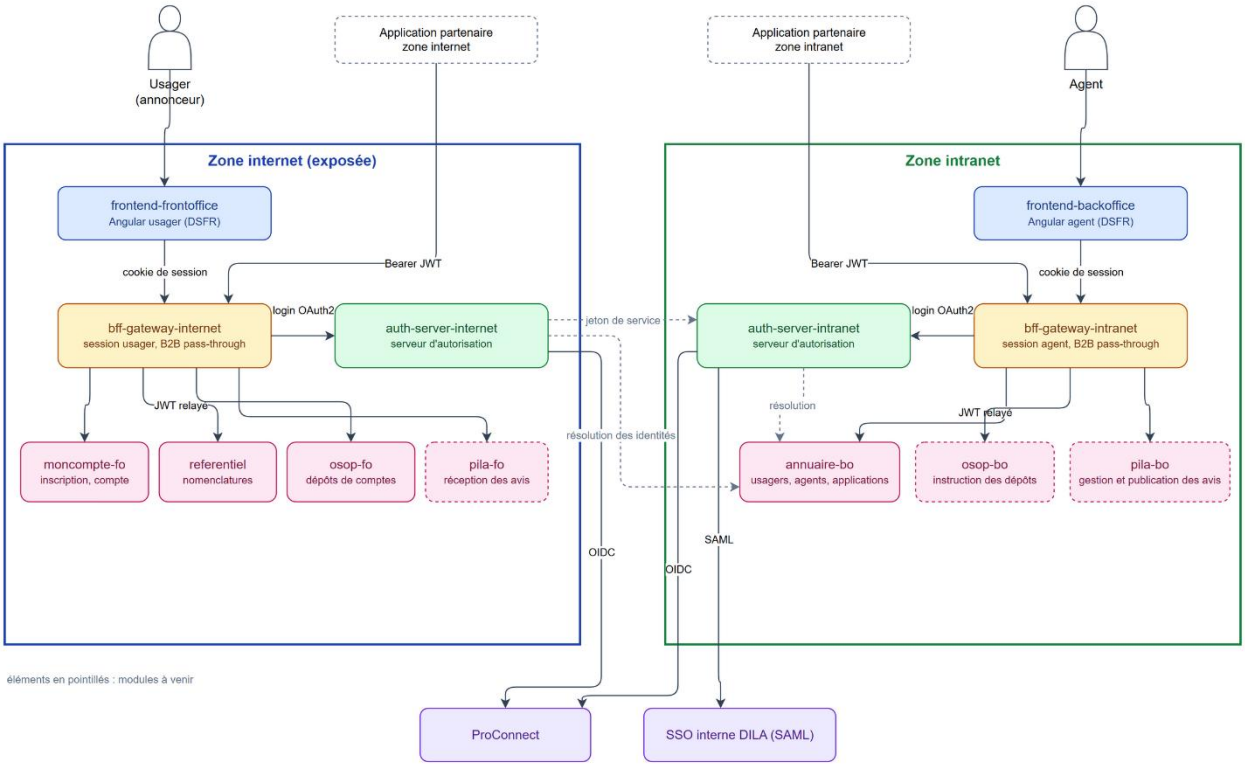


Figure 2 — Vue à date de l'architecture applicative

La plateforme est organisée en cinq couches applicatives :

- **Frontends Angular** : deux applications distinctes, une par population (frontend-frontoffice pour les annonceurs, frontend-backoffice pour les agents). Le navigateur ne détient jamais de jeton : sa seule matérialisation de session est un cookie HTTP.
- **Passerelles BFF (Backend For Frontend)** : une par zone réseau (bff-gateway-internet et bff-gateway-intranet). Chacune porte la session OIDC de sa population, détient les jetons pour le compte du navigateur et les relaie vers les microservices. Elles portent aussi le canal machine à machine : le jeton d'une application partenaire est validé puis relayé tel quel, sans session.
- **Microservices métier**, resource servers sans état : ils exposent des API protégées, valident le JWT reçu et ne tiennent aucune session. Chaque microservice appartient à un domaine fonctionnel et possède ses données.
- **Serveurs d'autorisation** : le module auth-server, déployé en deux instances, une par zone. Chaque instance délègue l'authentification aux fournisseurs d'identité, résout le compte auprès de l'annuaire et émet les JWT internes que valident les microservices de sa zone.
- **Bases PostgreSQL**, une par domaine de données, sans lien structurel entre elles.

La séparation des populations est topologique : chaque zone a sa passerelle, son serveur d'autorisation, son issuer, sa clé de signature et son registre de clients OAuth2. La zone exposée ne peut pas émettre de jeton d'agent. Ce cloisonnement est détaillé aux chapitres Architecture technique et Architecture de sécurité.

5.3 Deux canaux

Canal front-office (usagers, zone internet). L'annonceur charge frontend-frontoffice, qui dialogue exclusivement avec bff-gateway-internet sous couvert d'un cookie de session. Au login, la passerelle engage le flux OAuth2 avec auth-server-internet, qui délègue l'authentification à ProConnect puis résout l'identité auprès de l'annuaire ; en retour de quoi la passerelle détient les jetons de l'usager et les relaie vers les microservices du canal : moncompte-fo (inscription, compte), referentiel (nomenclatures), osop-fo (dépôts de comptes OS-OP) et, à venir, pila-fo (réception des avis pour publication). Les applications partenaires de la zone (le registre des associations pour l'instance PILA, par exemple) appellent les mêmes API en Bearer direct, sans session.

Canal back-office (agents, zone intranet). L'agent charge frontend-backoffice, qui dialogue avec bff-gateway-intranet selon le même patron. Au login, auth-server-intranet propose ProConnect ou le SSO SAML interne, puis résout l'identité côté agent. La passerelle route vers annuaire-bo (gestion des usagers, des agents et des applications partenaires) et, à venir, vers osop-bo et pila-bo (instruction des dépôts et des avis). Les applications partenaires de la zone intranet disposent du même canal Bearer direct.

Quelques composants servent les deux canaux : annuaire-bo résout les identités pour les deux serveurs d'autorisation, et l'instance intranet du serveur d'autorisation émet les jetons de service des deux zones. Ces flux inter-zones sont limités, unidirectionnels et décrits au chapitre Architecture technique ; aucun flux applicatif n'est initié depuis la zone exposée vers la zone interne en dehors d'eux.

5.4 Composants applicatifs

Cette section détaille les composants de la plateforme : les frontends, les passerelles BFF, les serveurs d'autorisation et les microservices métier.

5.4.1 Frontends

Composant	Responsabilité	Technologie
frontend-frontoffice	Application web des usagers (annonceurs) : connexion, inscription, dépôts et suivi.	Angular 22, DSFR
frontend-backoffice	Application web des agents : gestion des usagers, des agents et des applications partenaires, instruction des dossiers.	Angular 22, DSFR

Les deux applications partagent les mêmes principes :

- **DSFR** : les interfaces sont construites avec les composants du Système de Design de l'État, en visant la conformité RGAA pour le front-office.
- **Aucun jeton dans le navigateur** : pas de stockage de token dans le navigateur ni de flux OAuth2 côté client ; l'authentification est entièrement portée par la passerelle, le navigateur ne détient qu'un cookie de session.
- **Session sondée, droits dérivés du jeton** : l'interface interroge la passerelle pour connaître l'état de session et les droits de l'utilisateur, et conditionne l'affichage à partir de cette réponse.

- **Déploiement** : chaque frontend est servi par un serveur web statique, la configuration d'environnement étant injectée au démarrage du conteneur.

5.4.2 Passerelles BFF (Backend For Frontend)

Composant	Responsabilité	Technologie
bff-gateway-internet	Point d'entrée HTTP de la zone internet : porte la session des usagers (login OAuth2, cookie de session, logout), détient leurs jetons et les relaie vers les microservices du canal, contrôle le scope exigé par chaque route, et porte le canal machine à machine (validation puis relais du Bearer des applications partenaires). Sessions déportées dans Redis pour permettre une haute disponibilité (plusieurs instances).	Java 25, Spring Boot 4, Spring Cloud Gateway 5
bff-gateway-intranet	Même rôle pour la zone intranet, au service des agents et des applications partenaires de cette zone.	Java 25, Spring Boot 4, Spring Cloud Gateway 5

5.4.3 Serveurs d'autorisation

Composant	Responsabilité	Technologie
auth-server (deux instances, une par zone)	Délégation de l'authentification aux fournisseurs d'identité (ProConnect en zone internet ; ProConnect et SAML en zone intranet), résolution des comptes auprès de l'annuaire, émission des JWT internes, registre des clients OAuth2 de sa zone, API interne d'administration des clients.	Java 25, Spring Boot 4, Spring Authorization Server 7

Techniquement, le serveur d'autorisation est un microservice comme les autres : même socle, même modèle de construction, de conteneurisation et de déploiement. Il est présenté à part parce que son rôle est transverse : il ne porte aucun domaine métier, mais la sécurité de tous.

5.4.4 Microservices

Composant	Responsabilité	Technologie
annuaire-bo	Annuaire des usagers, des	Java 25, Spring Boot 4

Composant	Responsabilité	Technologie
	agents et des applications partenaires, structuré par population. Orchestre la gestion composite « application partenaire + client OAuth2 associé » (création, rotation de secret, désactivation) en s'appuyant sur l'API d'administration du serveur d'autorisation de la zone concernée. Propriétaire de la base du domaine annuaire.	
moncompte-fo	Volet front-office du domaine compte : inscription des annonceurs, consultation du compte. Consommateur de la base du domaine annuaire.	Java 25, Spring Boot 4
referentiel	Lecture des référentiels transverses aux domaines. Un référentiel propre à un seul domaine a vocation à vivre dans son domaine, pas dans ce microservice. Propriétaire de la base du domaine référentiel.	Java 25, Spring Boot 4
osop-fo	Domaine métier de l'instance OS-OP côté front-office : formulaires et cycle de vie des dépôts de comptes, conservation des documents déposés dans un stockage objet, génération du justificatif. Partage la base du domaine OS-OP.	Java 25, Spring Boot 4
osop-bo (à venir)	Volet back-office de l'instance OS-OP : consultation et recherche des dossiers, contrôle et validation des comptes et des demandes de consultation, transfert d'une organisation entre entités, envoi pour diffusion. Partage la base du domaine OS-OP.	Java 25, Spring Boot 4
pila-fo (à venir)	Volet front-office de l'instance PILA : réception des avis transmis par le registre des associations, pour	Java 25, Spring Boot 4

Composant	Responsabilité	Technologie
	publication. Partage la base du domaine PILA.	
pila-bo (à venir)	Volet back-office de l'instance PILA : création, gestion et validation des avis, publication des avis dans les bulletins. Partage la base du domaine PILA.	Java 25, Spring Boot 4

Convention de nommage : le suffixe -fo/-bo désigne le canal du déployable.

Les responsabilités transverses sont mutualisées dans des socles techniques partagés plutôt que dupliquées : un starter pour les passerelles (filtres de scope, relais conditionnel de jeton, résolution du fournisseur d'identité au login), un starter pour les resource servers (validation JWT et conversion des autorisations, chaîne de sécurité par défaut, gestion d'erreurs normalisée RFC 7807), le référentiel d'autorisation en code partagé (profils, fonctionnalités, scopes), un socle d'exceptions portant la sémantique HTTP, et des clients HTTP de consommation inter-services qui propagent le jeton de l'utilisateur appelant (client du référentiel, client du domaine OS-OP).

5.5 Contrats d'interface

Les API sont conçues en contract-first : le contrat OpenAPI est écrit d'abord, dans un module dédié par service (annuaire-bo-openapi, moncompte-fo-openapi, referentiel-openapi, osop-openapi, auth-server-openapi, etc.). Les interfaces serveur et les objets d'échange sont générés à la construction et ne sont pas versionnés ; côté Angular, les clients d'API sont générés depuis les mêmes contrats. Ce fonctionnement garantit que frontends, microservices et consommateurs machine à machine partagent une définition unique de chaque API.

6. ARCHITECTURE TECHNIQUE

6.1 Zones réseau

La plateforme est découpée en deux zones réseaux :

	Zone internet (exposée)	**Zone intranet (interne)**
Exposition	Internet public	Réseau privé, accès agents
Population	Usagers (front-office)	Agents (back-office)
Frontend	frontend-frontoffice	frontend-backoffice
Passerelle BFF	bff-gateway-internet	bff-gateway-intranet
Serveur d'autorisation	auth-server-internet	auth-server-intranet
Fournisseurs d'identité	ProConnect (OIDC)	ProConnect (OIDC) et SSO interne (SAML)
Microservices	moncompte-fo, referentiel, osop-fo, pila-fo (à venir)	annuaire-bo, osop-bo et pila-bo (à venir)
Registre OAuth2	base sigmae_oauth2_internet	base sigmae_oauth2_intranet

Deux règles gouvernent le placement des services :

- **Un resource server est mono-issuer** : il ne valide que les jetons du serveur d'autorisation de sa zone et n'est routé que par la passerelle de sa zone. Un service n'est donc jamais « à cheval » sur la frontière.
- **Un service consommé par les deux populations se déploie en deux instances**, une par zone, sur le modèle du serveur d'autorisation (un module, une image, un profil de configuration de zone), les deux instances partageant la base de leur domaine.

Conformément à la contrainte de cloisonnement du SI (chapitre Contexte et exigences), aucun flux n'est initié depuis l'extérieur vers la zone interne ; les seuls flux traversant la frontière entre zones sont ceux de la matrice ci-dessous, tous initiés côté plateforme.

6.2 Topologie de déploiement

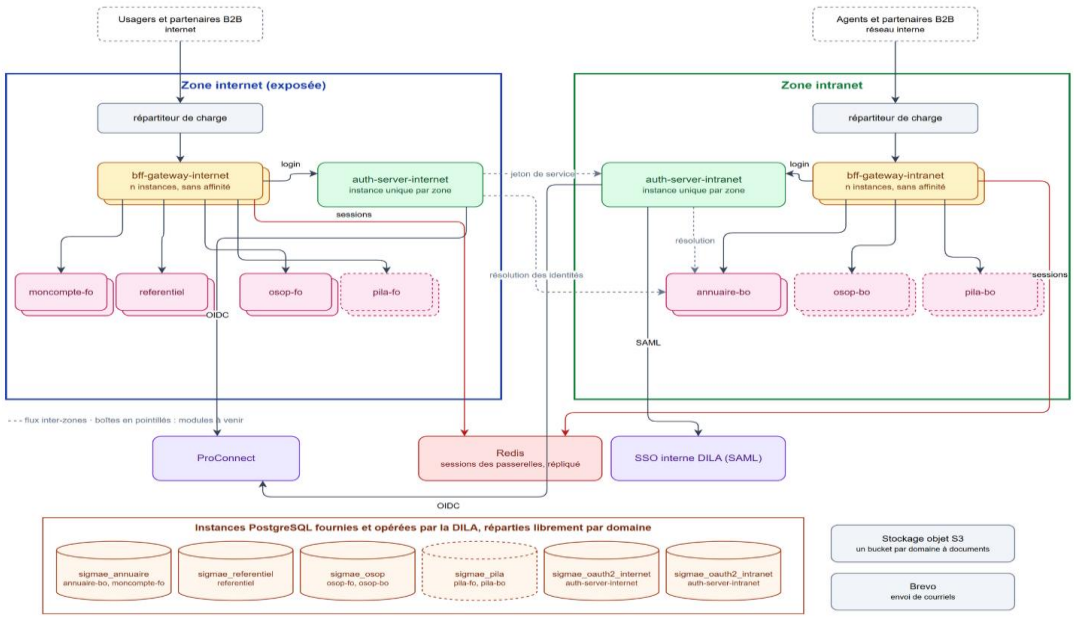


Figure 3 — Topologie de déploiement

- **Conteneurisation** : chaque livrable applicatif (frontends, passerelles, serveurs d'autorisation, microservices) est une image de conteneur autonome ; la plateforme s'inscrit dans la stratégie d'orchestration de conteneurs de la DILA. Les frontends sont des images de serveur web statique dont la configuration d'environnement est injectée au démarrage.
- **Bases de données hors conteneurs** : les instances PostgreSQL sont fournies et opérées par la DILA. La plateforme ne fait aucune hypothèse de colocalisation des bases : chaque domaine peut être hébergé sur une instance distincte.
- **Migrations de schéma** : chaque base est migrée par un job Flyway dédié, exécuté au déploiement avant le démarrage des services du domaine.
- **Session distribuée** : un entrepôt Redis porte les sessions des passerelles (voir « Haute disponibilité »).
- **Ordre de démarrage** : les passerelles chargent les clés publiques de leur serveur d'autorisation au démarrage et échouent immédiatement s'il est injoignable ; le serveur d'autorisation de chaque zone démarre donc avant sa passerelle, et les jobs de migration avant les services de leur domaine. L'instance internet du serveur d'autorisation démarre sans dépendre de l'instance intranet (résolutions paresseuses) ; l'indisponibilité de l'intranet n'empêche pas son démarrage, seulement l'aboutissement des logins.

6.3 Images de conteneurs livrées

Toutes les images sont construites sur des bases Alpine.

Image	Contenu	Socle technique
sigmae/frontend-frontoffice, sigmae/frontend-backoffice	Applications Angular compilées, servies en statique	Nginx 1.29 non privilégié
sigmae/bff-gateway-internet, sigmae/bff-gateway-intranet	Passerelles BFF (Spring Cloud Gateway)	Java 25 (JRE Eclipse Temurin), Spring Boot
sigmae/auth-server	Serveur d'autorisation, une image pour les deux déploiements de zone	Java 25 (JRE Eclipse Temurin), Spring Boot
sigmae/annuaire-bo, sigmae/moncompte-fo, sigmae/referentiel, sigmae/osop-fo	Microservices resource servers	Java 25 (JRE Eclipse Temurin), Spring Boot
sigmae/annuaire-flyway, sigmae/auth-server-flyway, sigmae/referentiel-flyway, sigmae/osop-flyway	Jobs de migration de schéma, exécutés une fois au déploiement	Flyway 11

Les modules à venir (osop-bo, pila-fo, pila-bo et le job de migration du domaine PILA) suivront le même modèle et s'ajouteront à ce catalogue.

6.4 Les deux instances du serveur d'autorisation

Le serveur d'autorisation est un module unique produisant une image unique, déployée en deux instances. Les divergences entre zones passent par un profil de configuration :

- **issuer, clé de signature et registre de clients propres à chaque zone** : leur séparation rend le cloisonnement structurel (chapitre Architecture de sécurité) ;

- l'instance internet ne déclare que ProConnect ; l'instance intranet déclare ProConnect et le SP SAML du SSO interne ;
- l'instance intranet émet les jetons de service des deux instances : l'instance auth-server internet est un client client_credentials ordinaire de l'auth-server intranet, si bien que les autorisations de la zone exposée sont décidées en zone interne ;
- chaque instance expose une API interne d'administration des clients de son registre, hors passerelle, dont le seul consommateur est annuaire-bo ; l'instance internet n'accorde sa confiance sur cette API qu'aux jetons émis par l'issuer intranet, jamais aux siens.

6.5 Matrice des flux

Le protocole indiqué est celui du flux tel qu'il circule : TLS est terminé par le répartiteur de charge en frontal des composants exposés (frontends, passerelles, serveurs d'autorisation), les échanges internes à la plateforme circulent en HTTP, à l'exception des appels aux serveurs d'autorisation qui restent en HTTPS de bout en bout. Les flux marqués « inter-zones » traversent la frontière réseau et sont les seuls à le faire.

Source	Destination	Protocole	Objet
Navigateur usager	frontend-frontoffice puis bff-gateway-internet	HTTPS	Chargement de l'application, puis API sous cookie de session
Navigateur agent	frontend-backoffice puis bff-gateway-intranet	HTTPS	Idem, canal agent
Navigateur usager	auth-server-internet	HTTPS	Redirections du login OAuth2 (autorisation, retour du fournisseur d'identité)
Navigateur agent	auth-server-intranet	HTTPS	Idem, canal agent
Application partenaire	passerelle de sa zone	HTTPS	API en Bearer JWT, sans session
bff-gateway-internet	auth-server-internet	HTTPS	Échange du code contre les jetons ; clés publiques (JWKS) au démarrage
bff-gateway-internet	moncompte-fo, referentiel, osop-fo, pila-fo (à venir)	HTTP	Relais des appels API, JWT en Bearer
bff-gateway-intranet	auth-server-intranet	HTTPS	Échange du code contre les jetons ; JWKS au démarrage
bff-gateway-intranet	annuaire-bo, osop-bo et pila-bo (à venir)	HTTP	Relais des appels API, JWT en Bearer
Passerelles	Redis	TCP (protocole Redis)	Lecture/écriture des sessions
auth-server-internet	ProConnect	HTTPS	Délégation d'authentification OIDC

Source	Destination	Protocole	Objet
auth-server-intranet	ProConnect, SSO interne	HTTPS	Délégation d'authentification OIDC et SAML
auth-server-internet	annuaire-bo (inter-zones)	HTTP	Résolution des identités usagers (jeton de service)
auth-server-internet	auth-server-intranet (inter-zones)	HTTPS	Obtention de son jeton de service ; JWKS intranet pour la confiance de son API d'administration
auth-server-intranet	annuaire-bo	HTTP	Résolution des identités agents
annuaire-bo	auth-server-internet (inter-zones)	HTTPS	Administration des clients OAuth2 de la zone internet (jeton de l'agent propagé)
annuaire-bo	auth-server-intranet	HTTPS	Administration des clients OAuth2 de la zone intranet
Microservices et serveurs d'autorisation	base PostgreSQL de leur domaine	TCP (protocole PostgreSQL)	Persistance
osop-fo	stockage objet S3	HTTPS (API S3)	Documents déposés et justificatifs
Microservices	service d'envoi de courriels (Brevo ou relais SMTP de la DILA)	HTTPS (API REST Brevo) ou SMTP/TLS	Notifications aux annonceurs

6.6 Haute disponibilité

- **Passerelles** : bien qu'elles portent les sessions des utilisateurs, les passerelles ne conservent aucun état local. Le contexte de sécurité et les jetons OAuth2 de chaque session sont persistés dans Redis via Spring Session : plusieurs instances d'une même passerelle peuvent tourner derrière un répartiteur de charge, sans affinité de session, avec bascule transparente en cas de perte d'une instance.
- **Resource servers** : sans état par construction (validation de JWT, aucune session), ils se répliquent librement.
- **Serveurs d'autorisation** : l'état durable (registre des clients, codes et jetons en cours de validité) est en base, et la clé de signature est montée par l'infrastructure avec un identifiant de clé stable entre redémarrages. En revanche, la cinétique de login tient un état de session en mémoire locale le temps de la délégation au fournisseur d'identité (requête d'autorisation sauvegardée, state et nonce OIDC, requête SAML) : la réplication d'un serveur d'autorisation imposerait une affinité de session au répartiteur de charge ou l'externalisation de cette session dans un entrepôt partagé, non retenues à ce stade. Une instance unique par zone suffit.

- **Redis** : éphémère et mono-nœud dans les environnements de développement ; en production, déploiement répliqué (Sentinel ou Cluster), authentifié et chiffré. La perte totale de Redis ne compromet aucune donnée métier : elle invalide les sessions en cours, les utilisateurs se reconnectent.
- **PostgreSQL** : la disponibilité des bases relève de l'infrastructure de la DILA ; la plateforme n'impose ni topologie ni mécanisme de réplication.

7. ARCHITECTURE DE SECURITE

7.1 Authentification

Principe. La plateforme n'authentifie pas elle-même les personnes : elle délègue aux fournisseurs d'identité (ProConnect en OIDC, SSO interne de la DILA en SAML) et ne manipule aucun secret d'authentification humain. Seule exception à venir : le compte de plateforme des annonceurs sans SIREN (chapitre Contexte et exigences), dont les modalités restent à définir. Dans chaque zone, le serveur d'autorisation orchestre cette délégation, jouant un rôle de broker d'authentification : il reçoit la demande de login de la passerelle, redirige vers le fournisseur d'identité choisi, résout le compte correspondant auprès de l'annuaire, puis émet le JWT interne (OIDC) que valideront les microservices de la zone.

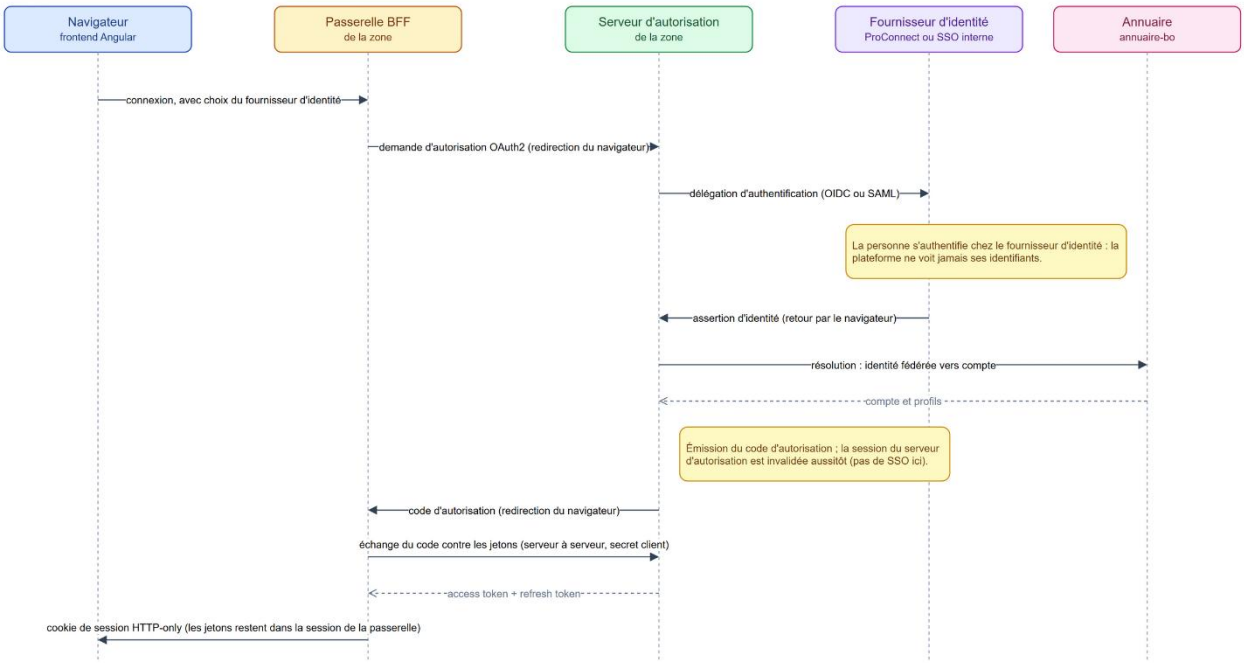


Figure 4 — Séquence de login

Sélection du fournisseur d'identité. Chaque passerelle ne connaît qu'un seul serveur d'autorisation : celui de sa zone. Le choix du fournisseur d'identité (ProConnect ou SSO interne) est porté par un paramètre transmis au moment du login, propagé de l'interface jusqu'au serveur d'autorisation, qui redirige vers le fournisseur choisi, chacun avec son protocole.

Résolution des comptes. Le fournisseur d'identité ne fait qu'authentifier : il atteste qu'une personne est bien celle qu'elle prétend être, mais ne sait rien de son existence dans SIGMAé. Les comptes, leurs identités fédérées et leurs profils vivent dans l'annuaire (annuaire-bo). Une fois l'authentification acquise, le serveur d'autorisation interroge donc l'annuaire pour transformer l'identité fédérée reçue (fournisseur + identifiant chez ce fournisseur) en compte de la plateforme : c'est la résolution.

L'annuaire tient deux populations de comptes séparées, usagers et agents, aux cycles de vie distincts (auto-inscription d'un côté, provisionnement par un administrateur de l'autre). Le serveur d'autorisation déduit la population par configuration (la passerelle front-office donne un usager, la passerelle back-office un agent) et la résolution n'interroge que les API de cette population.

La résolution procède alors comme suit. Le serveur d'autorisation cherche d'abord le compte déjà lié à l'identité fédérée reçue. S'il n'en trouve pas, il cherche un compte portant la même adresse électronique et, le cas échéant, lui rattache cette nouvelle identité : c'est ainsi qu'un agent provisionné entre dans le système à sa première connexion. Si aucun compte ne correspond, l'utilisateur reçoit un profil synthétique, sans écriture en base : « non inscrit » côté usager, qui n'ouvre que le parcours d'inscription ; « non habilité » côté agent, qui n'ouvre aucun droit.
Un compte ou une application désactivés se voient refuser toute émission, y compris au renouvellement de jetons. L'effet d'une désactivation est plafonné par un cache de résolution court (cinq minutes). Si l'annuaire est indisponible, le serveur d'autorisation refuse d'émettre le jeton (comportement fail-closed).

7.2 Sessions et tokens

Quatre couches d'état distinctes participent à « être connecté », chacune avec sa durée propre :

Couche	Détenteur	Durée
Session de la passerelle BFF	passerelle de la zone (cookie HTTP-only, état en Redis)	30 minutes d'inactivité, fenêtre glissante
Session du serveur d'autorisation	serveur d'autorisation de la zone	durée du flux de login uniquement : la session est invalidée dès l'émission du code
Chaîne de jetons OAuth2	détenue par la session BFF	access token 5 min (back-office) / 15 min (front-office) ; refresh token 8 h, plafond absolu de la session
Session du fournisseur d'identité	ProConnect ou SSO interne	selon le fournisseur (ProConnect : 12 h)

Points structurants :

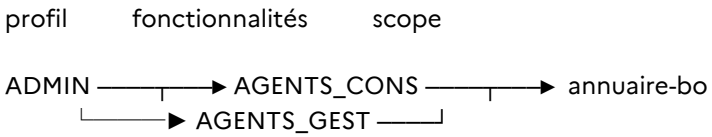
- **Le navigateur ne détient jamais de jeton** : uniquement un cookie de session HTTP-only et un cookie CSRF. Les jetons vivent dans la session de la passerelle, côté serveur, et sont relayés par elle vers les API.
- **Le SSO est porté par les fournisseurs d'identité, pas par le serveur d'autorisation** : sa session étant invalidée dès l'émission du code, chaque re-login rejoue la délégation vers le fournisseur d'identité, qui décide seul d'un éventuel re-login silencieux au vu de sa propre session.
- **Le TTL des access tokens est une politique de sécurité** : un access token est un JWT auto-porteur, que les microservices valident par sa seule signature, sans rappeler le serveur d'autorisation. Conséquence : révoquer un compte ou modifier ses droits ne prend effet qu'à l'expiration des jetons déjà émis, et un jeton qui fuiterait resterait utilisable jusqu'à son expiration. La durée de vie borne donc ces deux risques à la fois, d'où la règle : durée inversement proportionnelle au privilège (5 minutes pour la surface d'administration, 15 minutes pour les usagers).
- **Renouvellement sans rotation du refresh token** : le refresh token vit dans la session de la passerelle, côté serveur ; le navigateur ne le voit jamais. Il est de plus inutilisable seul : à chaque renouvellement, le serveur d'autorisation exige aussi le secret du client BFF (la passerelle est un client OAuth2 confidentiel). La rotation systématique des refresh tokens, recommandée pour les clients publics dont le jeton vit dans le navigateur ou l'appareil de l'utilisateur, apporterait donc ici peu de bénéfice, au prix d'une complexité réelle (renouvellements concurrents) : les

refresh tokens sont réutilisables pendant toute leur durée de validité, un renouvellement n'en émet pas de nouveau.

- **Fin de session** : l'expiration du refresh token (8 h après le login) est le plafond absolu ; l'échec de renouvellement est traité comme une fin de session normale (401 propre, retour à la page de connexion). La déconnexion est un logout local (invalidation de la session BFF) ; la session du fournisseur d'identité lui survit, ce qui est cohérent avec le SSO porté par l'IdP.

7.3 Autorisation

Les trois niveaux du modèle. Le modèle d'autorisation est un graphe profils → fonctionnalités → scopes. Le profil est le rôle métier attribué à un compte ; il accorde un ensemble de fonctionnalités, qui sont le grain fin de l'autorisation, celui de l'action métier. Le scope est la porte d'entrée d'un microservice, à raison d'un scope par service : détenir le scope donne le droit d'atteindre le service, les fonctionnalités décident ensuite de ce que l'on peut y faire. Les scopes ne s'attribuent jamais directement : ils se déduisent des fonctionnalités que portent les profils. Par exemple, pour un administrateur qui gère les agents :



Le profil ADMIN accorde la consultation (AGENTS_CONS) et la gestion (AGENTS_GEST) des agents ; ces deux fonctionnalités étant portées par le microservice annuaire-bo, le jeton de l'administrateur reçoit le scope annuaire-bo, qui lui ouvre ce service.

Un référentiel porté par le code. Ce graphe réside dans le code (module partagé par tous les services), et non en base de données : c'est un choix d'implémentation, qui fait du catalogue des habilitations un élément versionné, revu et testé comme le reste du système. Les profils y sont déclinés par population (usager, agent, application, technique) avec une garde à la compilation : accorder une fonctionnalité d'agent à un profil d'usager ne compile pas.

Ce que porte le JWT. Le jeton contient l'identifiant du compte, sa population (usager ou agent), ses codes profil et ses scopes. Les scopes sont calculés par le serveur d'autorisation au moment de l'émission, à partir des fonctionnalités des profils du porteur ; le client n'en demande jamais. Le graphe profils → fonctionnalités résidant dans le code, les codes profil du jeton suffisent : chaque microservice les expanse localement en fonctionnalités, sans avoir besoin de maintenir un état ou d'interroger l'annuaire. Trois formes d'émission existent : utilisateur humain (profil de sa population), application partenaire (profil applicatif, identifiée comme telle), client interne de la plateforme (jeton technique, profils issus de la configuration).

Défense en profondeur. Chaque appel traverse trois contrôles indépendants : la passerelle exige le scope déclaré sur la route ; le microservice filtre globalement sur son propre scope ; chaque opération contrôle la fonctionnalité requise. L'interface utilisateur ne porte aucune logique d'autorisation propre : elle conditionne l'affichage sur les droits dérivés du jeton, exactement ce que contrôlent les API.

Étanchéité des populations. Le site des usagers (front-office) et celui des agents (back-office) sont deux entrées distinctes, et chacune ne sait connecter que sa propre population. Un agent qui se connecterait via ProConnect par le site des usagers n'obtiendrait donc aucun droit d'agent, et réciproquement.

7.4 Cloisonnement des zones

Le cloisonnement des deux zones repose d'abord sur le réseau, opéré par la DILA. La plateforme y ajoute un cloisonnement applicatif et cryptographique, qui ne repose pas sur le réseau et resterait effectif si un flux normalement interdit devenait possible. Il tient à trois éléments propres à chaque zone, livrés ensemble :

- **un issuer par zone** : un jeton émis pour une zone est rejeté par l'autre à la validation de son émetteur ;
- **une clé de signature par zone** : deux issuers partageant une clé afficheraient un cloisonnement sans effet, la confiance reposant in fine sur la clé ;
- **un registre de clients (OAuth2) par zone** : chaque serveur d'autorisation ne connaît que les clients de sa zone et n'accède qu'à la base de son propre registre. Le registre internet ne contient ni le client de la passerelle back-office ni aucun compte de service : la zone exposée ne peut donc structurellement pas émettre de jeton d'agent, et sa compromission ne donnerait accès ni aux secrets des clients de l'intranet ni aux refresh tokens des agents.

Deux dispositions complètent ce cloisonnement sur les flux entre zones :

- **Les jetons de service sont émis par la zone interne.** Chaque serveur d'autorisation a besoin d'un jeton de service pour interroger l'annuaire. Celui de la zone internet n'émet pas le sien : il l'obtient auprès du serveur intranet, comme n'importe quel client, avec un secret qui lui est propre. Ce que la zone exposée a le droit de faire est donc décidé en zone interne.
- **L'administration du registre internet est réservée à la zone interne.** Chaque serveur d'autorisation expose une API interne d'administration de son registre de clients, dont le seul consommateur est l'annuaire, qui l'appelle en direct. Cette API n'est routée par aucune passerelle : elle n'est pas atteignable depuis l'extérieur, quelle que soit la zone. Sur l'instance internet, cette API n'accepte que les jetons émis par le serveur intranet, jamais les siens : même compromise, la zone exposée ne peut pas administrer son propre registre, ces droits venant toujours de la zone interne.

7.5 Gestion des secrets et des clés

- **Clés de signature JWT** : une paire de clés par zone, dans un keystore PKCS12 monté par l'infrastructure, avec un identifiant de clé stable entre redémarrages. Hors environnement de développement, l'absence de keystore empêche le démarrage (aucune génération de clé à la volée).
- **Clé privée SAML** : portée par la seule instance intranet ; elle ne réside jamais en zone exposée.
- **Secrets des clients OAuth2** : stockés exclusivement en hachage bcrypt (coût 12), jamais en clair en base. Le secret d'une application partenaire est généré côté serveur et communiqué une seule fois, à la création ou à la rotation ; sa rotation est disponible depuis le back-office. Les mutations d'administration sont journalisées, sans jamais inclure le secret.
- **Secrets d'exploitation** (secrets des clients BFF, mots de passe des bases) : injectés par l'environnement d'exécution, jamais présents dans les images ni dans le code source. Aucun jeton n'est écrit dans les journaux.

7.6 Flux machine à machine

Aucun humain n'intervient sur ce canal : le client obtient son jeton par client_credentials directement auprès du serveur d'autorisation de sa zone. Le grant n'émet pas de refresh token :

le client redemande un jeton à l'expiration du précédent, dont la durée de vie est courte (5 minutes). Deux natures de clients empruntent ce canal, et tout client client_credentials qui n'appartient à aucune des deux se voit refuser le jeton.

7.6.1 Applications partenaires

Une application partenaire est un système tiers déclaré dans l'annuaire depuis le back-office. Sa déclaration fixe sa zone (immuable) et son profil applicatif, nécessairement de la même zone (cohérence contrôlée à la création, à la modification et au démarrage), et crée conjointement son compte de service OAuth2 dans le registre de la zone.

Son autorisation suit exactement le même modèle que celle des humains : le profil applicatif accorde des fonctionnalités, dont se déduisent les scopes. Les profils applicatifs du lot (publicateur PILA, partenaires OSOP et PILA) sont catalogués ; leurs fonctionnalités s'ouvriront avec les premières API partenaires. Par exemple, à l'ouverture de l'API PILA, le profil de publicateur accordera la fonctionnalité de dépôt des avis, portée par le service PILA : le jeton du registre des associations recevra le scope correspondant.

Le jeton d'une application partenaire ressemble donc à celui d'un humain : il porte ses codes de profil et ses scopes calculés à l'émission. À la place de l'identité d'une personne, il porte l'identifiant et le libellé de l'application, ainsi qu'un marqueur de nature (client_type valant application) ; il ne porte jamais d'identité humaine.
Pour consommer les API, l'application présente son jeton en Bearer à la passerelle de sa zone, qui le valide (signature, émetteur, expiration) puis le relaie tel quel au microservice. Aucune session n'est créée, et la protection CSRF, qui protège les sessions à cookie, ne s'applique pas à ce canal.

7.6.2 Clients internes

Les clients internes sont les composants de la plateforme eux-mêmes, livrés avec elle : aujourd'hui, les comptes de service des serveurs d'autorisation, qui leur servent à interroger l'annuaire. Ils sont déclarés par configuration, avec des profils techniques de moindre privilège, et leurs secrets sont configurables par environnement. Leur jeton porte le marqueur client_type valant internal.
Les clients des passerelles BFF (bff-fo, bff-bo) sont eux aussi des clients OAuth2 livrés avec la plateforme, aux secrets configurables par environnement, mais ils n'appartiennent pas à ce canal : clients du canal humain, ils n'obtiennent jamais de jeton technique. Les jetons émis à travers eux sont ceux des utilisateurs connectés.

7.7 Protection active contre les menaces

Mesures contre la falsification de requête intersite (CSRF) :

- le cookie de session porte l'attribut SameSite=Lax : le navigateur ne le joint pas aux requêtes initiées depuis un site tiers ;
- un jeton anti-CSRF dédié est exigé sur toute requête modifiant l'état du système (POST, PUT, PATCH, DELETE) : la passerelle le dépose dans un cookie et l'application Angular le renvoie dans l'en-tête X-XSRF-TOKEN, en s'appuyant sur les mécanismes standards de Spring Security et d'Angular ;
- les méthodes GET, HEAD, OPTIONS et TRACE ne modifient pas, et ne doivent jamais modifier, l'état du système ;
- l'en-tête X-Frame-Options est positionné à DENY : les applications ne peuvent pas être affichées dans un cadre, même d'origine identique.

Mesure contre la fixation de session :

- un nouvel identifiant de session est généré à chaque ouverture de session.

Mesures contre XSS et les autres injections :

- une politique de sécurité des contenus (CSP) est servie avec chaque frontend : contenus, scripts, styles et images restreints à l'origine de l'application, connexions et soumissions de formulaires restreintes à l'application et à sa passerelle ;
- Angular assainit et échappe toutes les valeurs insérées dans le DOM depuis les templates ;
- les accès aux bases de données passent exclusivement par des requêtes paramétrées, ce qui prémunit des injections SQL.

Ces mesures sont complétées par les en-têtes de durcissement servis avec les frontends : X-Content-Type-Options: nosniff, Referrer-Policy, Permissions-Policy restrictive (capteurs, caméra, géolocalisation, paiement… désactivés), Cross-Origin-Opener-Policy: same-origin, et la suppression de la signature du serveur web.
Les mesures portées par la configuration Nginx (CSP, X-Frame-Options et les en-têtes de durcissement ci-dessus) sont livrées dans les images de conteneurs des frontends. En cas de déploiement non conteneurisé des frontends, cette configuration serait à reconstituer sur le serveur web retenu et relèverait de la DILA.

8. ARCHITECTURE DE DONNEES

8.1 Bases par domaine

La maille de découpage des données est le domaine fonctionnel : chaque domaine possède sa base PostgreSQL, dont il est seul maître. La plateforme compte six bases :

Base	Schéma	Contenu	Micro-service
sigmae_annuaire	annuaire	Usagers, agents, identités fédérées, profils, applications partenaires	annuaire-bo, moncompte-fo
sigmae_osop	osop	Dépôts de comptes OS-OP : formulaires, cycle de vie, métadonnées des documents	osop-fo, osop-bo (à venir)
sigmae_pila (à venir)	pila	Avis PILA : réception, gestion, publication dans les bulletins	pila-fo, pila-bo
sigmae_referentiel	referentiel	Nomenclatures transverses aux domaines (référentiels code vers libellé)	referentiel
sigmae_oauth2_internet	oauth2	Registre OAuth2 de la zone internet : clients, autorisations et jetons en cours	auth-server-internet
sigmae_oauth2_intranet	oauth2	Registre OAuth2 de la zone intranet : idem	auth-server-intranet

Règles structurantes :

- **Aucun lien structurel entre les bases** : ni clé étrangère, ni jointure, ni transaction inter-bases. Un rapprochement entre domaines passe par HTTP entre les services.
- **Aucune hypothèse de colocalisation** : les instances PostgreSQL étant fournies et opérées par la DILA, chaque service porte des paramètres de connexion propres à son domaine (hôte, port, nom de base indépendants) ; la répartition des bases sur une ou plusieurs instances est libre.
- **Le domaine prime sur la zone** : un service déployé en deux instances, une par zone, partage la base de son domaine. Le registre OAuth2 est la seule exception (voir ci-dessous).
- **Les documents ne sont pas en base** : la plateforme conserve les documents déposés et les justificatifs dans un stockage objet S3 ; seules les données structurées des dépôts et les métadonnées des documents sont stockées en base.

8.2 Stockage objet

Le stockage objet suit la même maille que les bases : chaque domaine qui conserve des documents dispose de son bucket S3 dédié, configurable par environnement. La base du domaine garde un pointeur sur ce stockage ainsi que des métadonnées. Chaque domaine définit son plan de classement.

8.3 Registres OAuth2 par zone

Le registre OAuth2 est la donnée du serveur d'autorisation : les clients déclarés et les autorisations en cours (codes et jetons émis). C'est un contenu sensible, où l'on trouve notamment les secrets des clients et les refresh tokens des sessions.
C'est pourquoi ce domaine, et lui seul, est dédoublé : une base par zone, chacune accessible uniquement au serveur d'autorisation de sa zone. Même entièrement compromise, la zone internet n'atteint ainsi ni les secrets des clients de l'intranet, ni les refresh tokens des agents.
Les deux bases ont la même structure (même schéma oauth2, mêmes tables, migrées par les mêmes scripts) ; seules leurs données diffèrent, chaque registre contenant les clients de sa zone et rien d'autre (chapitre Architecture de sécurité).

8.4 Schémas, rôles et privilèges

Chaque base contient un unique schéma nommé, celui de son domaine, et a son rôle PostgreSQL applicatif dédié :

Base	Rôle applicatif	Utilisé par	Rôle propriétaire (migrations)
sigmae_annuaire	sigmae_annuaire	annuaire-bo, moncompte-fo	sigmae_install
sigmae_osop	sigmae_osop	osop-fo, osop-bo (à venir)	sigmae_install
sigmae_pila (à venir)	sigmae_pila	pila-fo, pila-bo	sigmae_install
sigmae_referentiel	sigmae_referentiel	referentiel	sigmae_install
sigmae_oauth2_internet	sigmae_oauth2_internet	auth-server-internet	sigmae_install
sigmae_oauth2_intranet	sigmae_oauth2_intranet	auth-server-intranet	sigmae_install

Le rôle sigmae_install est le rôle d'installation : il crée et possède les bases, et c'est avec lui que les jobs de migration Flyway se connectent. Sa granularité est à la discrétion de la DILA (un rôle d'installation par instance PostgreSQL ou un par base) ; la plateforme n'impose que la séparation entre le rôle propriétaire, qui migre les structures, et les rôles applicatifs, qui manipulent les données.

Les privilèges suivent le même modèle pour tous les rôles applicatifs :

- **Connexion** : le droit CONNECT est révoqué au rôle générique PUBLIC sur toutes les bases, puis accordé nominativement à chaque rôle sur sa seule base (LOGIN NOINHERIT). Aucun rôle ne peut joindre la base d'un autre domaine et, pour les registres OAuth2, aucun rôle ne peut joindre la base de l'autre zone.
- **Schéma et données** : USAGE sur le schéma de son domaine ; SELECT, INSERT, UPDATE, DELETE, TRUNCATE sur les tables ; USAGE, SELECT, UPDATE sur les séquences.

- **Aucun droit de définition** : un rôle applicatif ne peut ni créer, ni modifier, ni supprimer une structure. Les schémas et les tables sont créés par les migrations Flyway, exécutées avec le rôle d'installation sigmae_install : lui seul est propriétaire des objets, et c'est lui qui accorde aux rôles applicatifs leurs privilèges de manipulation.

8.5 Réplicas de lecture

Tous les domaines, à l'exception des registres OAuth2 du serveur d'autorisation, suivent ce modèle : une base en lecture-écriture et un réplica en lecture seule. Chaque service maintient deux pools de connexions et route ses transactions selon leur nature : les transactions en lecture-écriture s'exécutent sur l'instance en lecture-écriture, les transactions en lecture seule sur le réplica. La mise en place et l'opération de la réplication relèvent des instances PostgreSQL de la DILA ; la plateforme apporte, elle, la séparation des transactions par nature dans chaque service.

8.6 Migrations

- Chaque base est migrée par Flyway, avec un module de migrations par domaine ; la table d'historique de chaque base vit dans le schéma de son domaine.
- Les migrations sont exécutées par des jobs dédiés au déploiement, avant le démarrage des services du domaine ; aucun service applicatif n'embarque ni n'exécute de migration.

9. EXPLOITATION

9.1 Démarrage et ordonnancement

À rédiger : ordre de démarrage des services et dépendances au boot (serveur d'autorisation avant les gateways), comportement en cas d'indisponibilité d'une dépendance.

9.2 Sauvegarde et restauration

À rédiger : périmètre couvert par le projet (environnements du prestataire) et périmètre relevant de la DILA (production), principe des scripts fournis.

9.3 Observabilité

À rédiger : logs applicatifs, corrélation des requêtes, endpoints de santé et métriques exposés.

9.4 Supervision

À rédiger : points de contrôle recommandés, alertes, indicateurs de bon fonctionnement.