DOCUMENTATION CELLULE-PRO

Architecture Technique

Modèle 4 couches · fédération · MoE sharding

cellule.ai · pro.cellule.ai
2026-05-18

Table des matières

0.1 Cellule PRO — Architecture de référence

Ce document est la source de vérité pour le déploiement Cellule PRO chez un client entreprise.

Le pitch en deux phrases (selon segment)

Pitch générale

Pour toute entreprise avec du compute dormant (laptops, desktops, serveurs idle) :

> Les entreprises possèdent énormément de puissance de calcul non utilisée. Il suffit de l'utiliser et d'apporter des serveurs NPU pour équilibrer l'inférence, permettant d'utiliser les LLM de manière sécurisée et avoir une inférence confortable.

Pitch ciblé studios pro (VFX, design, cinéma, labos scientifiques)

Quand l'interlocuteur a déjà des GPU workstations haut de gamme (RTX 4090/A6000, Radeon Pro W7900, Apple M3 Ultra, Quadro) :

> Vous avez la puissance, nous avons la solution !

L'idée : pas besoin d'investir dans un serveur NPU dédié. Les workstations GPU du studio, idle la nuit et le weekend, fournissent déjà 10-50× plus de capacité qu'un NPU consumer. Cellule PRO supporte CUDA, ROCm, Vulkan, Metal et MLX pour exploiter toute la flotte hétérogène.

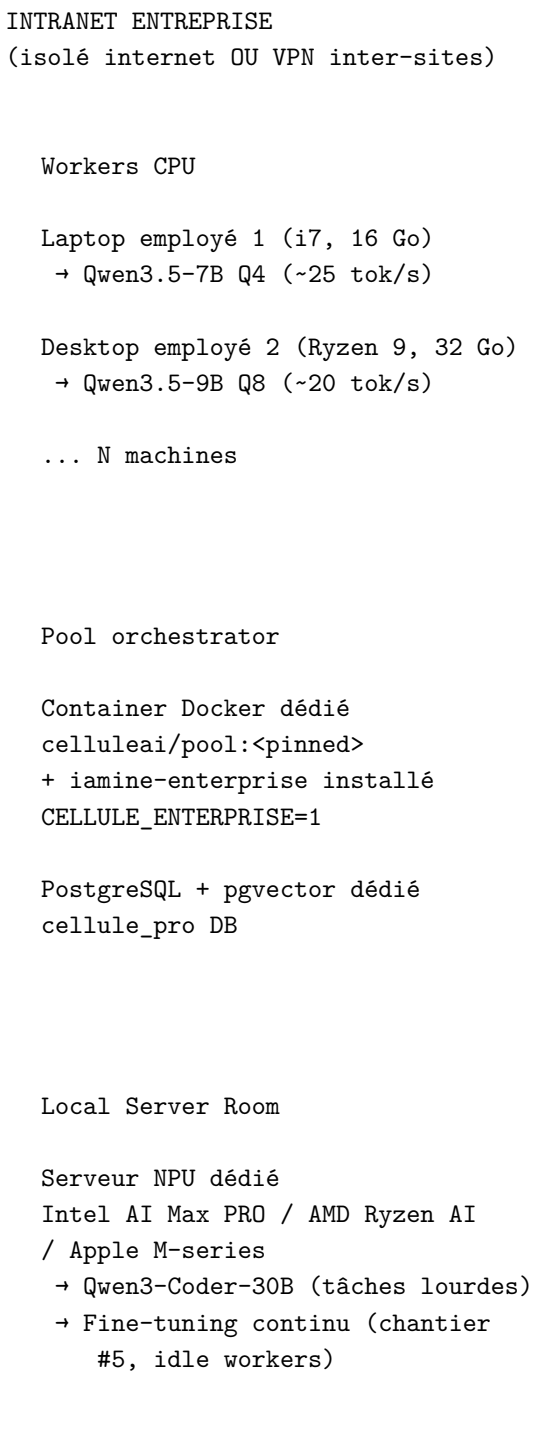
Ce que ça signifie concrètement

1. **Le gisement existant** — dans toute entreprise :
   - Laptops employés inactifs ~18h/jour (nuits, week-ends, pauses)
   - Desktops et workstations CAD/dev qui idlent le soir
   - Serveurs de backup ou de développement sous-utilisés
   - Stations de travail graphique la nuit Cette puissance de calcul est DÉJÀ payée (matériel amorti, électricité, maintenance) mais non valorisée.
2. **L'apport technologique** — un serveur NPU dédié (Intel AI Max PRO, AMD Ryzen AI, Apple M-series, Jetson) installé dans le local serveur client, qui :
   - Sert les tâches d'inférence lourdes (modèle 30B+) qui ne tiennent pas sur un laptop CPU
   - Fait tourner le fine-tuning continu (chantier #5) sur les données de l'entreprise
   - Équilibre la charge entre workers CPU employés et calcul centralisé
3. **Le résultat** — une infrastructure d'inférence LLM :
   - **Sécurisée** : tout reste dans l'intranet client, rien ne sort vers internet
   - **Confortable** : qualité d'inférence équivalente à ChatGPT/Claude grâce au mix CPU + NPU
   - **Économique** : pas d'achat massif de GPU, valorisation de l'existant + 1 seul serveur NPU dédié
   - **Contrôlée** : licence fixe, pas de facturation par token

Argument commercial vs concurrence US

Au lieu d'envoyer vos données à un fournisseur US à 20€/employé/mois (ChatGPT Enterprise, Claude for Business), réveillez vos machines existantes et achetez un seul serveur NPU (5-15k€ one-shot). Gérez tout chez vous. Zéro data leak, coût prévisible, compliance native.

Topologie de déploiement type



```
   Web UI interne      VPN multi-site
(accessible depuis        (bureaux
 n'importe quel poste   additionnels)
  employé authentifié)
```

Composants détaillés

1. Workers CPU — machines employés

Prérequis matériel : - CPU moderne : Intel i5/i7/i9 12th gen+, AMD Ryzen 5/7/9 série 5000+, Apple Silicon M1+ - RAM : 16 Go minimum, 32 Go recommandé - Disque : 30 Go disponibles (modèles GGUF) - OS : Linux / Windows / macOS

Modèle par défaut : Qwen3.5-7B ou 9B en Q4/Q8 selon RAM disponible.

Installation employé : - Service systemd (Linux) / LaunchAgent (macOS) / scheduled task (Windows) - Démarre au boot, s'enregistre auprès du pool orchestrator intranet - Aucune config manuelle (auto-discovery via broadcast mDNS ou config déployée par IT)

Politique de ressources (configurable par l'IT) : - Worker s'exécute à `nice=10` (priorité basse) - Pause si CPU $> 80\%$ pendant 60s (éviter de ralentir le travail de l'employé) - Reprend idle (CPU $< 20\%$)

2. Serveur NPU dédié

Prérequis matériel : - Machine avec NPU intégré (Intel AI Max PRO, AMD Ryzen AI, Apple M3 Ultra, Jetson) - 64 Go RAM minimum, 128 Go recommandé - 500 Go SSD NVMe - Réseau gigabit (10 Gbit si possible)

Modèles servis : - Qwen3-Coder-30B (code, raisonnement complexe) - Qwen3.5-35B-A3B (MoE, tâches étendues) - Optionnel : modèle fine-tuné sur données entreprise (chantier #5)

Rôle dans le pool : - Worker "premium" auto-sélectionné pour les tâches tagguées `complex=True` - Fallback des workers CPU quand latency p95 dépasse SLA - Hôte du fine-tuning continu (nuit/weekend) via idle workers CPU contribuant des gradients

3. Pool orchestrator

Deployment : - Container Docker `celluleai/pool:<pinned-version>` déployé sur une VM Linux dédiée - Package `iamine-enterprise` installé via `pip install` depuis un wheel privé fourni par CELLULE - `CELLULE_ENTERPRISE=1` set dans l'environnement - PostgreSQL + pgvector en container ou VM adjacente

Config différences vs cellule.ai public : - `POOL_URL=http://pool.intranet.client.local:8080` (URL interne) - `FEDERATION_ENABLED=false` par défaut (mono-site) ou `true` + peers intranet uniquement - Pas de connexion sortante vers cellule.ai public - `REGISTRATION_ENDPOINT=disabled` (pas d'inscription publique)

Accès : - Web UI via VPN ou dans l'intranet uniquement - SSO entreprise (LDAP, Azure AD, Okta) — à intégrer dans une itération ultérieure - API tokens gérés par l'admin IT (pas self-serve utilisateur)

4. Base de données

PostgreSQL 16 avec extensions `pgvector` et `pgcrypto`.

Données stockées : - Conversations chiffrées Fernet zero-knowledge (par user session) - `user_memories` vectorisées (RF=3 si multi-site via chantier #1) - `worker_reputation` + `inference_verifications` (chantier #4) - `agent_observations`, `agent_episodes` (M13 public)

Backup : snapshots quotidiens, rétention 30 jours, chiffrés offsite (NAS intranet).

Sizing guide — combien de workers pour N employés ?

Hypothèse : usage moyen = 20 requêtes/jour/employé, latency cible p50 < 3s.

Employés actifs	Workers CPU	Serveur NPU	RAM pool orchestrator	Postgres
10-30	5-15 idle machines	1× NPU mid-tier (128 Go)	16 Go	50 Go SSD
30-100	15-50	1× NPU haut de gamme ou 2× mid	32 Go	200 Go SSD
100-300	50-150	2× NPU haut de gamme	64 Go	500 Go SSD NVMe
300+	multi-site fédéré	2+ NPU par site	128 Go	1 To NVMe + read replicas

Un worker CPU moderne sert ~5-10 requêtes/minute en soutien (20-40 tok/s avec modèle 7-9B). Donc 1 worker = ~300-600 requêtes/heure.

Processus d'onboarding client (9 étapes)

1. **Kickoff** — call 1h pour établir : nb employés, contraintes compliance, sites, modèles cibles
2. **Audit infra** — l'IT client liste machines + OS + topologie réseau
3. **Devis + contrat** — licence + setup + support + NDA
4. **Commande matériel si besoin** — serveur NPU si client n'en a pas
5. **Install pool orchestrator** — container docker sur VM Linux fournie par client
6. **Déploiement workers** — script IT pour installer l'agent sur les postes (MDM, GPO, bash)
7. **Formation admin IT** — 2h : monitoring, upgrade, backup, troubleshooting
8. **Formation employés** — 30 min : comment accéder au web UI, où sont leurs conversations
9. **Recette + mise en production** — tests SLA sur 2 semaines avant switch

Temps total : 2-6 semaines selon taille.

Compliance native

- **RGPD art. 32** : chiffrement at-rest (Fernet), chiffrement in-transit (TLS intranet), pseudonymisation (token_hash)
- **HIPAA §164.308(a)(7)** : contingency plan = backup + RTO documented

- **SOC 2 Type II** : audit trail (chantier #4 proof of inference + reputation + deterministic mode max)
- **ISO 27001** : SMSI documenté, contrôles d'accès, backup, incident response

Différences structurelles vs cellule.ai public

Aspect	Cellule communautaire	Cellule PRO
Domaine	cellule.ai	intranet client (ex : ai..local)
Workers	Z2, Gladiator, master.86 (infra David)	Machines employés client
Users	Anon + compte gratuit	Employés authentifiés entreprise
Data location	VPS Contabo Nuremberg	Intranet client
Code	`CELLULEAI/POOL` AGPLv3	`CELLULEAI/POOL` + `iamine-enterprise` privé
Plugins enterprise	Jamais chargés	Tous chargés (CELLULE_ENTERPRISE=1)
Fine-tuning données client	Non	Oui (chantier #5, idle workers)
Pricing	Gratuit	Licence fixe + setup + support
SLA	Best-effort	Contractuel

À retenir

Cellule PRO est une offre produit distincte, pas une version premium de cellule.ai public. Elle partage la fondation technique open-source (AGPLv3) et ajoute une suite de plugins propriétaires sous licence commerciale + NDA.

Le business model est **licence + service**, pas pay-per-token. Le client possède son infrastructure, ses données, son contrôle. CELLULE fournit le logiciel et le support.