

Architectuur & Ontwerpfilosofie

Productvisie, High Level Design en referentiearchitectuur.

Velio — Architectuur & Ontwerpfilosofie

Productvisie, High Level Design en referentiearchitectuur

Status: in ontwikkeling · **Versie platform:** 3.2.1

Voorwoord

Velio is ontstaan uit mijn overtuiging dat professionele communicatie niet ingewikkeld hoeft te zijn.

Tijdens mijn werk zag ik hoe implementaties steeds complexer werden, terwijl de behoefte van veel organisaties juist eenvoudig was: betrouwbaar communiceren, zonder afhankelijk te zijn van één leverancier of een onnodig ingewikkeld platform.

Daarom heb ik Velio bewust anders ontworpen. Niet met zoveel mogelijk functies, maar met een duidelijke architectuur, open standaarden en reproduceerbare deployments als uitgangspunt. Ik geloof dat software begrijpelijk moet blijven — voor beheerders, partners én ontwikkelaars.

Die filosofie zie je terug in iedere ontwerpkeuze. Een nieuwe Velio-omgeving moet geen project van dagen of weken zijn, maar in enkele uren operationeel kunnen zijn. De infrastructuur is reproduceerbaar, de configuratie grotendeels geautomatiseerd en een volledige omgeving kan opnieuw worden opgebouwd zonder complexe handmatige stappen. Daardoor blijven implementaties voorspelbaar, upgrades beheersbaar en herstel na een incident snel en betrouwbaar.

Velio is nog volop in ontwikkeling. Dit document beschrijft de huidige architectuur, de gemaakte ontwerpkeuzes en de richting waarin het platform zich ontwikkelt.

Leeswijzer

Dit document opent met de visie en de ontwerpprincipes, gevolgd door twee delen voor twee soorten lezers:

- **Deel A — Productbeschrijving.** Voor een breed publiek: wat Velio is, voor wie, en waarom je het zou kiezen. Geen technische voorkennis nodig.
- **Deel B — High Level Design.** Voor een technisch publiek (architecten, beheerders, een klant die al Nextcloud draait): referentiearchitectuur, security, koppelvlakken, deployment en de backup/restore-strategie, met schematische weergaven in Mermaid.

Statusmarkering door het document heen:  beschikbaar ·  in ontwikkeling ·  ontworpen (nog niet gebouwd).

1. Waarom bestaat Velio?

Moderne communicatieplatformen worden steeds uitgebreider en daarmee vaak ook complexer om te implementeren en te beheren.

Velio kiest bewust voor eenvoud.

Het platform is gebouwd rond open standaarden en geeft organisaties de vrijheid om zelf keuzes te maken: waar het platform draait, hoe identiteiten worden beheerd en met welke systemen wordt geïntegreerd.

Velio integreert daarom uitstekend met platformen zoals Nextcloud, maar blijft volledig zelfstandig inzetbaar. Integraties zijn een keuze, geen vereiste.

2. Ontwerpprincipes

Begrijpelijke software. Iedere architectuurkeuze wordt afgewogen op eenvoud, onderhoudbaarheid en voorspelbaarheid.

Open standaarden. SIP, WebRTC, OIDC en PostgreSQL vormen de basis.

Reproduceerbare infrastructuur. Nieuwe omgevingen worden geautomatiseerd uitgerold zodat iedere installatie dezelfde kwaliteit heeft.

Deploybaarheid. Een nieuwe klantomgeving moet binnen enkele uren operationeel kunnen zijn. Niet omdat snelheid een doel op zich is, maar omdat een reproduceerbare deployment leidt tot minder fouten, eenvoudiger beheer en sneller herstel.

Single tenant. Iedere klant krijgt een volledig eigen omgeving.

Backend is de bron van waarheid. Businesslogica hoort niet thuis in de PBX.

Security by design. Beveiliging is onderdeel van het ontwerp, niet een uitbreiding achteraf.

Deel A — Productbeschrijving

A1. Wat is Velio?

Velio is een **communicatieplatform**. Of je nu eenvoudig een **receptie / telefooncentrale** wilt, een volwaardig **contactcenter voor je servicedesk** wilt uitrollen, of **beide** — Velio dekt die lijn met één geïntegreerd product.

De kracht van Velio zit in eenvoud, snelle uitrol en een platform dat door een klein team te begrijpen en te beheren blijft. Het is bewust niet gebouwd om contactcenters met honderden agents te bedienen — die keuze houdt het licht, snel en overzichtelijk.

A2. Voor wie

- **Receptie & operator** — gesprekken aannemen, doorverbinden, overleg, parkeren.
- **Klantcontact & servicedesk** — wachtrijen, keuzemenu's, openingstijden, terugkoppeling.
- **Omnichannel contactcenter** — voice én webchat in één werkstroom en één rapportage.

A3. Waarom Velio

- **Eenvoud.** Eén samenhangend product in plaats van een wildgroei aan modules en SKU's.
- **Snelle restore & uitrol.** De hele omgeving wordt reproduceerbaar uitgerold; bij een storing herstel je door opnieuw te deployen en je data terug te zetten, niet door een dure schaduwomgeving te onderhouden. Dit is een kernbelofte — zie B7.
- **Open standaarden.** SIP, WebRTC, OIDC — geen vendor lock-in, geen gesloten protocol.
- **Volledig in de browser.** Agents bellen via een WebRTC-softphone; niets te installeren.
- **Omnichannel.** Voice en webchat delen wachtrij, presence en rapportage.
- **Nextcloud-vriendelijk.** Identiteit, directory en presence via Nextcloud/OIDC; spraak naar Talk-collega's via de eigen Velio-Talk-brug ✔.
- **Privacy & security by design.** NL-hosting, versleuteling, MFA, onveranderbare audittrail.

A4. Functioneel overzicht

Domein	Functionaliteit
Telefonie ✓	Inbound routing, wachtrijen, keuzemenu's (IVR), openingstijden/vakantie/ad-hoc, blind & consult transfer, parkeren, hold met music-on-hold
Agent-werkplek (/advisor) ✓	WebRTC-softphone, presence, wrap-up (gesprekscodering), nawerktijd, directory & zoeken
Beheer (/manage) ✓	Teams, telefoonnummers, contact centers (Lite Builder-wizard), agents, beveiliging, resources
Webchat ✓	Insluitbare widget, triage, gedeelde conversatie met voice, anti-abuse
Rapportage ✓	Realtime dashboard + publieke wallboard, historische rapportage, kanaal-bewust (voice/chat)
Identiteit & toegang ✓	OIDC (Nextcloud), RBAC (admin/supervisor/agent), MFA met herstelcodes
Quality monitoring & opnames 🔄	Gespreksopnames en QM als beheerd, privacygevoelig RBAC-object

A5. Licentierichting

Eenvoudig en verkoopbaar: **named seats** (gelicenseerde gebruikers/agents) plus een grens op het aantal **gelijktijdige inkomende gesprekken**. De grens telt op callniveau, niet op PBX-interne call legs of channels.

A6. Positionering

Velio onderscheidt zich niet op feature-volume maar op **eenvoud en snelle herstel-/uitroltijden**, op **open standaarden** en op naadloze aansluiting bij het **Nextcloud-ecosysteem** — gehost in Nederland/EU. Het is een platform voor organisaties die professioneel klantcontact willen zonder de complexiteit en lock-in van de grote suites.

Deel B — High Level Design

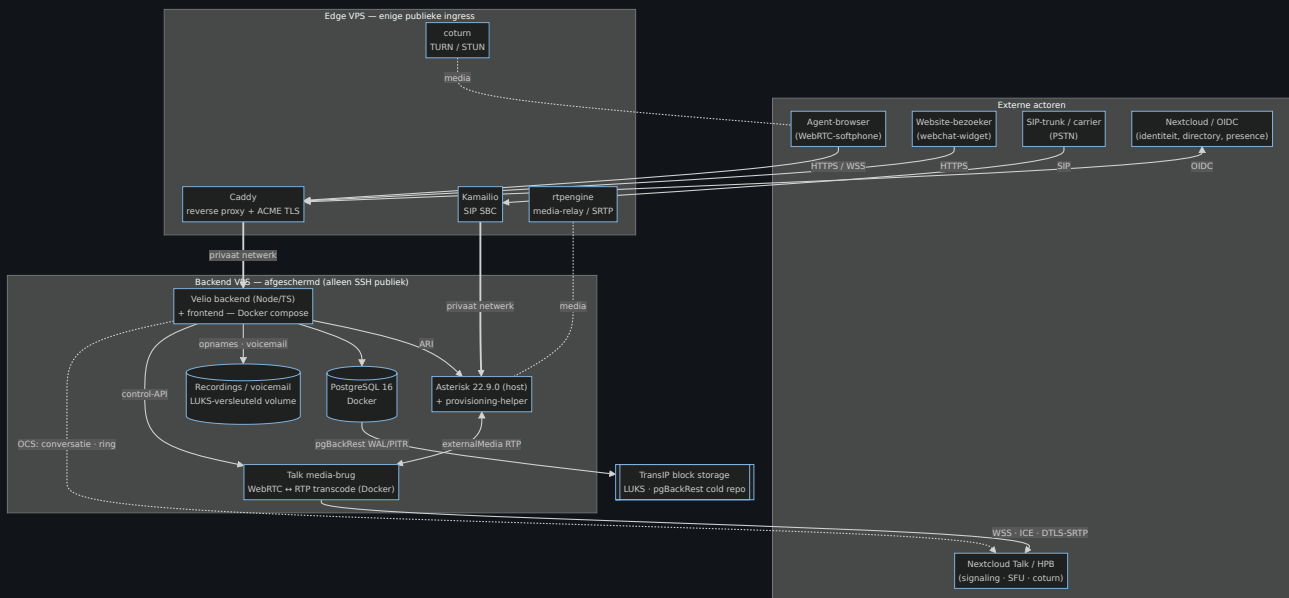
B1. Architectuurprincipes

De ontwerpprincipes uit hoofdstuk 2 vertalen zich technisch naar de volgende keuzes:

- **Backend is de bron van waarheid.** Call state, agent state, routing, queueing, business rules, audit en events leven in de Velio-core. De PBX levert signalering en media, niet de businesslogica.
- **Asterisk-first, achter een adapterboundary.** Asterisk is de vaste execution engine; een PBX-adapter (`PbxProvider`) vertaalt een **canonical telephony model** naar/van ARI. De core spreekt geen vendor-concepten.
- **Event-driven & realtime.** PBX → adapter → core → frontend, met WebSocket/server-push naar de UI; polling alleen als expliciete fallback.
- **Open standaarden voor media.** WebRTC, SIP, RTP/SRTP, SDP, ICE/STUN/TURN.
- **Single-tenant per klant.** Elke klant draait een eigen, geïsoleerde Velio-omgeving.
- **Voorspelbaarheid boven complexiteit.** Geen actieve HA-clustering; de omgeving wordt via Ansible identiek en herhaalbaar opgebouwd. Dat resulteert in reproduceerbare deployments in plaats van complexe HA-oplossingen — herstel verloopt via re-deploy + data-restore (B7), niet via failover.

B2. Referentiearchitectuur (2 VPS)

Een klantomgeving bestaat uit twee VPS'en, verbonden via een **privaat netwerk** (`vpriv0`). De **edge** is de enige publieke ingress; de **backend** (app, data, telefonie) is afgeschermd en alleen via SSH publiek bereikbaar.



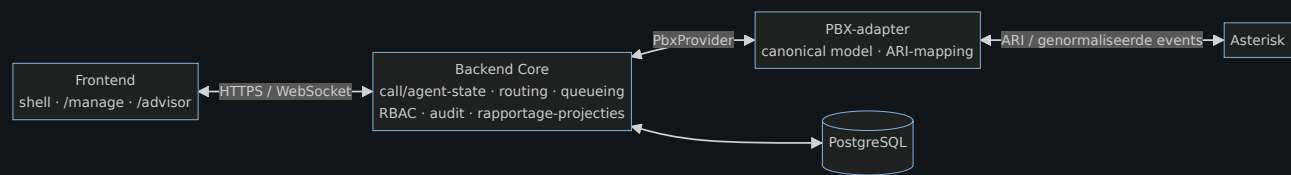
Edge VPS (firewall: 80/443 publiek, SIP 5080/udp + media-ranges, SSH beperkt):

- **Caddy** — reverse proxy met automatische TLS (ACME) voor de web-UI, API en WebSocket.
- **Kamailio** — SIP-SBC voor carrier-trunks (signaleringsgrens naar buiten).
- **rtpengine** — media-relay / SRTP-terminatie.
- **coturn** — TURN/STUN voor WebRTC NAT-traversal vanaf de agent-browser.

Backend VPS (firewall: alleen SSH publiek, rest op loopback/bridge/privaat):

- **Velo backend + frontend** — Node/TS-runtime en statische frontend in Docker compose.
- **PostgreSQL 16** — in Docker; de bron van waarheid voor product- en operationele data.
- **Asterisk 22.9.0** — op de host (gepinde baseline), met een beperkte provisioning-helper.
- **Talk media-brug** — los Docker-component (WebRTC ↔ RTP-transcode) dat een Asterisk-kanaal met een Nextcloud Talk-gesprek koppelt; praat over loopback met Asterisk en enkel **uitgaand** (WSS + ICE/TURN) met de Nextcloud-HPB. Altijd meegedeployd, alleen actief als de Talk-brug aanstaat (B4).
- **Schijfversleuteling (LUKS2)** op de dedicated data-volumes (pgBackRest-repo én recordings/voicemail); sleutel = de Ansible-vault-key per klant, DR-herstelbaar (B5).
- **Healthcheck** — `/health`, pjsip-transport, ARI-status, edge-bereikbaarheid.

B3. Logische lagen



De frontend is dun en PBX-agnostisch (een browser-shell met aparte `/manage` - en `/advisor` - modules). De core bezit het domein; de adapter isoleert alle Asterisk-specifieke uitvoering. Asterisk-provisioning (nummers, agent-endpoints) is een **synchronisatiepad**, geen beschikbaarheidsvoorwaarde: valt Asterisk tijdelijk weg, dan blijft de beheerlaag bruikbaar.

B4. Koppelvlakken

Koppelvlak	Protocol / techniek	Status
Carrier-trunks	SIP via Kamailio SBC	✓
Agent-media	WebRTC (DTLS-SRTP, ICE, coturn/rtpengine)	✓
Identiteit & directory	OIDC tegen Nextcloud; presence	✓
Webchat	Insluitbare widget over HTTPS/WSS	✓
Spraak naar Talk-collega's	Eigen Velio-Talk-brug (headless WebRTC); officiële SIP-bridge (Enterprise) als gereserveerd pad	✓
Rapportage-/CDR-export	Publieke Velio Core API	✓
Backup-opslag	pgBackRest → TransIP block storage (LUKS)	✓
Provisioning / fleet-beheer	Partner Portal (Ansible, TransIP-API)	⦿

Randvoorwaarde — E2EE-calls. Doorverbinden vanuit Velio naar een Nextcloud Talk-deelnemer vereist dat de betrokken conversatie **niet end-to-end versleuteld** is: een gateway die audio transcodeert (naar Asterisk/PSTN) moet de media kunnen ontsleutelen, wat per definitie onverenigbaar is met E2EE. Dit is geen Velio-beperking maar geldt voor **elke** Talk↔telefonie-koppeling — inclusief de officiële **SIP-bridge** (E2EE-calls ondersteunen geen SIP dial-in/-out) en elke PSTN-partij (een telefoontoestel is nooit een E2EE-deelnemer). E2EE **per conversatie** (opt-in) coëxisteert prima: de transfer-conversatie is dan simpelweg niet-E2EE, terwijl interne Talk↔Talk-calls wél E2EE kunnen zijn. Alleen **globaal afgedwongen E2EE** is een **niet-ondersteunde configuratie** voor telefonie-integratie.

B5. Security-architectuur

Netwerk

- De **edge is de enige publieke ingress**; de backend-VPS is afgeschermd (alleen SSH publiek), intern verkeer loopt over het private netwerk.
- **nftables** met default-deny op beide hosts; **fail2ban**, automatische security-updates en een gepinde timezone via de `common`-hardening.
- **Caddy** regelt TLS end-to-end met automatische certificaten.

Identiteit & toegang

- **OIDC** (Nextcloud) voor login; **RBAC** afgedwongen op zowel API- als UI-niveau met rollen `admin` / `supervisor` / `agent` en teamscope.
- **Single-session policy** per account (een tweede login neemt de sessie gecontroleerd over).
- **MFA** voor lokale accounts: TOTP met eenmalige **herstelcodes**; MFA uitzetten vereist **step-up** met een actuele code. Extern beheerde (OIDC) accounts vallen hierbuiten — hun beveiliging loopt via de IdP.

Data & secrets

- **Wachtwoorden** hash-only met **Argon2id** (nooit terugleesbaar).
- **Technische secrets** (SIP/provisioning, MFA-secrets) **AES-256-GCM encrypted-at-rest**, met een deployment-key **buiten** de database, in een afgeschermd secret-store-boundary — niet in brede productconfig, niet in de UI, niet in logs.
- **Geparametriseerde queries** overal; **onveranderbare audittrail** voor beheermutaties.

Opslag

- **Schijfversleuteling standaard (LUKS2)** op alle dedicated data-volumes — de pgBackRest-repo en het recordings/voicemail-volume. De sleutel is de **Ansible-vault-key per klant** (`/etc/velio/disk.key` , root-only); per constructie ge-escrowed, dus **DR-herstelbaar**. TransIP's off-site kopie is block-level en kopieert dus de LUKS-ciphertext — geen onversleuteld tussenstation.

Webchat anti-abuse

- Proof-of-work pre-queue-gating, rate-limits, concurrency-caps, idle/abandon-timeouts, een tijdelijke IP-block ("markeer als spam") en een URL-content-gate. Bewust **WAF-onafhankelijk**; een klant kan desgewenst zelf Cloudflare ervoor zetten.

Media

- WebRTC met **DTLS-SRTP**, ICE en TURN; G.711 (PCMU/PCMA) als interoperabele baseline.

B6. Deployment & operatie

Iedere klantomgeving wordt op identieke wijze uitgerold. Hierdoor zijn implementaties voorspelbaar, upgrades beheersbaar en incidenten eenvoudiger op te lossen.

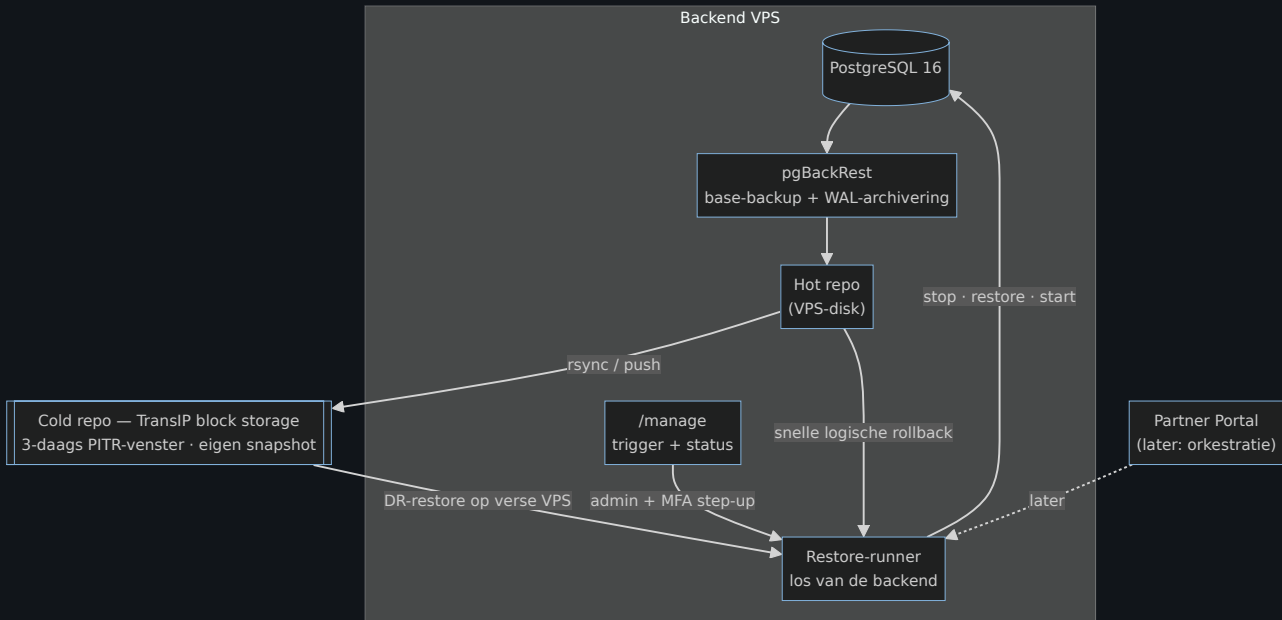
- **Ansible, 2-VPS, idempotent.** `edge.yml` en `backend.yml` (gedeeld via `site.yml`); een intake-wizard (`velio-deploy`) genereert de inventory en draait de apply.
- **Docker** voor backend + frontend (compose) en PostgreSQL; **Asterisk** op de host (gepinde 22.9.0). Releases gaan als rsync'te tar-images naar de backend-VPS.
- **Migraties draaien bij boot** van de backend — geen handmatige databasestappen.
- **Healthcheck-rol** bewaakt `/health` , pjsip-transports, ARI-status en edge-bereikbaarheid.
- **Privaat netwerk** (`vpriv0` , static IP, geen gateway) tussen edge en backend.

Doorlooptijd, realistisch. De Velio-omgeving zelf is doorgaans binnen enkele uren operationeel. De grootste variabele ligt buiten het platform: het **porteren of activeren van telefoonnummers** bij de carrier is een externe afhankelijkheid die de uiteindelijke doorlooptijd kan bepalen.

Richting. Het uiteindelijke doel is een volledig geautomatiseerde deployment, waarbij infrastructuur via de **TransIP-API** wordt ingericht en de complete omgeving zonder handmatige configuratie kan worden uitgerold. Op termijn worden deployment, lifecycle management en tenantmonitoring gecentraliseerd in de **Partner Portal**; iedere Velio-installatie publiceert daarvoor een gestandaardiseerd `/health` -endpoint.

B7. Backup, restore & continuïteit

De "snelle restore"-belofte uit Deel A, technisch onderbouwd. Ansible reconstrueert **code + config**; deze laag dekt de **state**.



Motor. **pgBackRest** (MIT-licentie) met **WAL-archivering** → **point-in-time recovery**, tot op de seconde vóór een incident. Draait als los tool naast PostgreSQL-in-Docker.

Twee tiers.

- **Hot** — base + WAL op de VPS-disk; snelle PITR bij logische fouten ("per ongeluk verwijderd").
- **Cold** — **TransIP block storage**, losgekoppeld van de compute (VPS weg ≠ data weg), optioneel in een ander datacenter, met een eigen ingebouwde snapshot als extra vangnet.

Retentie. PITR-venster van **3 dagen** — recent genoeg voor DR, ruim genoeg om een "pas later ontdekt"-fout over een weekend terug te halen; verder terug voegt niets toe. Eén configuratieregel, later bij te stellen.

Restore-architectuur. De backend praat met dezelfde database die hersteld wordt en kan zijn eigen restore dus niet uitvoeren. **Trigger en uitvoering zijn gescheiden:** `/manage` is de **trigger + status** (alleen `admin`, met **MFA step-up** en expliciete destructieve bevestiging), een **restore-runner** los van de backend doet het werk (maintenance → PostgreSQL stop → `pgbackrest restore` → start → backend terug). De runner is **bron-agnostisch** (hot-lokaal voor rollback, cold-extern voor DR op een verse VPS). De Partner Portal neemt later dezelfde runner over — geen weggegooid werk.

Kritieke koppeling. De data-repo en de **AES-encryptiesleutel** (Ansible-vault) zijn aan elkaar geklonken: een restore tegen een verloren/andere sleutel maakt MFA- en provider-secrets onleesbaar (een stille, halve restore). De restore-routine zet **beide** terug en versioneert ze samen.

Buiten scope. **Gespreksopnames** worden niet geback-upt — die vallen onder data-retentie en verdwijnen met de VPS. Alleen **QM-materiaal** wordt later apart duurzaam bewaard.

Compliance-bewaring van CDR's/audit hoort in de applicatie (rapportage/archiefslag), niet in het backup-venster — zo blijft dat venster kort.

Status: backup, WAL/PITR, de LUKS-repo op block storage en de /manage-observability zijn **gebouwd en productief** ✅ (sinds release 2.0.5, DR-run geoefend). De volledig geautomatiseerde, door de Partner Portal georkestreerde restore-naar-wegwerp blijft de resterende stap 🔄.

B8. Bewuste niet-doelen

- **Geen actieve HA-clustering** — voorspelbaarheid + snelle restore i.p.v. een continu meedraaiende schaduwomgeving (minder onderhoudslast, minder verborgen state).
- **Geen mega-contactcenters** (honderden agents) — die schaal vraagt een ander type platform; Velio kiest bewust voor eenvoud en overzicht.
- **Geen generieke PBX-shell** — Asterisk-first, met een scherpe adapterboundary.
- **Geen proprietary media-SDK's** — uitsluitend open standaarden.

Bijlage — bronnen

- `docs/architectuur/ARCHITECTUUR.md` — uitgebreide interne architectuur (domeinmodel, PBX-adapter, media, telefoniemodel).
- `docs/releases/` — release notes per versie.
- `infra/ansible/` — de 2-VPS-deployment (rollen `edge_*`, `asterisk_host`, `postgres`, `velio_app`, `healthcheck`).