JM1 - A cosa serve

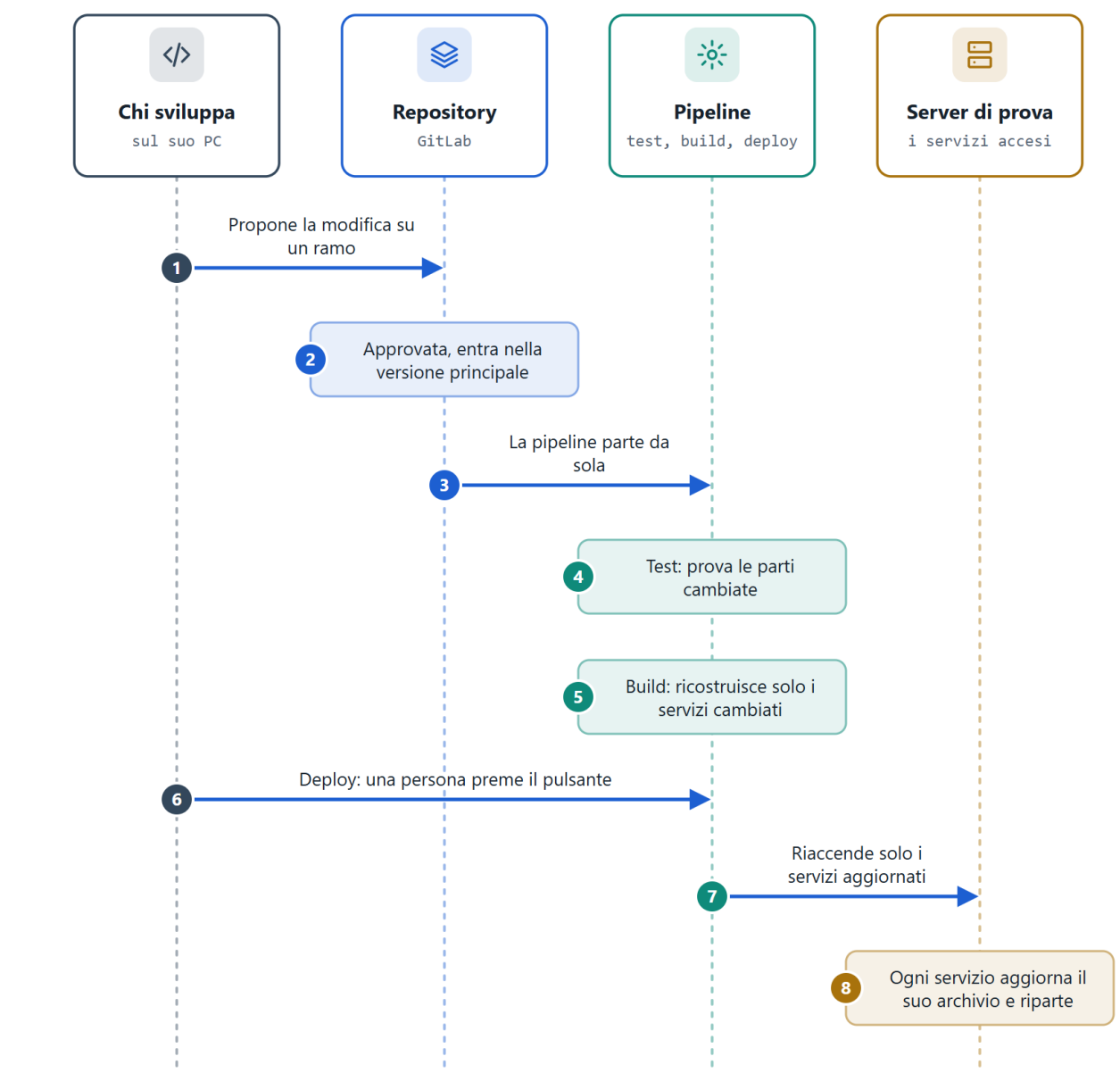
- JM1.1 - Dal codice al server
- JM1.2 - Dove sta nella mappa

JM1.1 - Dal codice al server

In una frase. Una modifica approvata arriva sul server di prova con un clic, e si ricostruiscono solo i servizi che sono cambiati davvero.

ADSWnext non è un programma solo: è fatto di più servizi, ognuno con il suo compito. Aggiornarli a mano, uno per uno, vorrebbe dire ricordarsi ogni volta l'ordine, i comandi e le impostazioni. Basta una dimenticanza per lasciare il server a metà.

Per questo il lavoro lo fa la **pipeline**: una sequenza di passi automatici che parte ogni volta che una modifica entra nella versione principale del codice. Prova le parti cambiate, prepara i servizi nuovi e poi si ferma ad aspettare. L'ultimo passo, mettere i servizi nuovi al posto dei vecchi, parte solo quando una persona preme un pulsante.



Dal PC di chi sviluppa al server di prova: la pipeline prova, ricostruisce e aspetta il clic

> **Cosa ci guadagni.**
> • **Un aggiornamento è un clic.** Nessuno si collega al server per digitare comandi. La pipeline fa sempre gli stessi passi, nello stesso ordine, e lascia scritto cosa ha fatto.
> • **Si ricostruisce solo ciò che è cambiato.** Se la modifica tocca solo le pagine, si rifà solo l'app web. Una modifica alla sola documentazione non ricostruisce niente.
> • **Chi non è toccato non si ferma.** I servizi rimasti uguali restano accesi. Chi sta lavorando nelle maschere storiche non perde la sessione per un aggiornamento che riguarda le stampe.

Oggi l'installazione che la pipeline tiene aggiornata è una: il **server di prova**, dove si guarda il lavoro prima di portarlo altrove. Una seconda installazione accanto a quella e la distribuzione ai

clienti sono ancora da fare: c'è un piano approvato, non ancora realizzato.

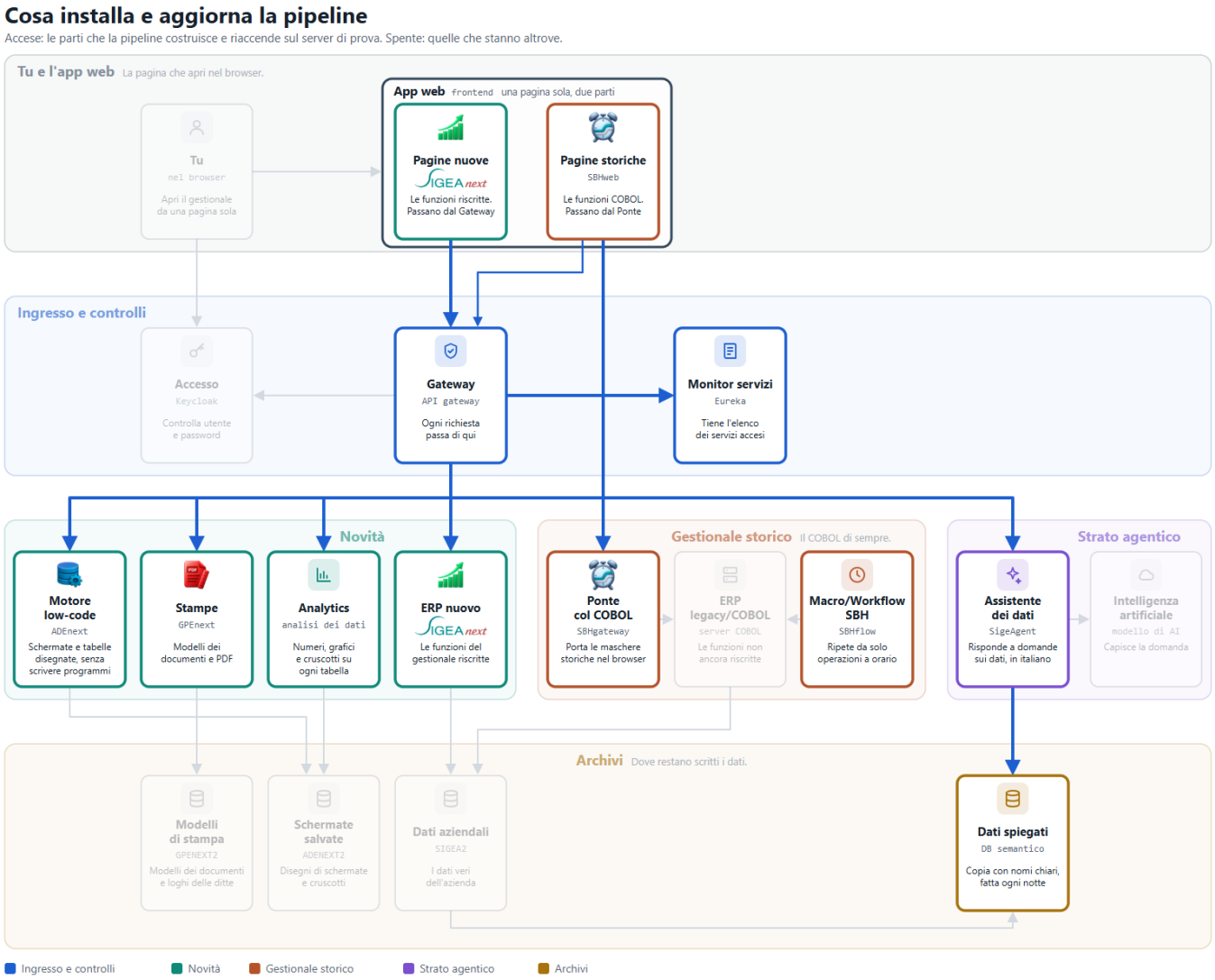
Questo libro racconta cosa gira sul server di prova, come arriva un aggiornamento, come si controlla che sia andato bene e cosa si può fare quando non va.

> **Per chi sviluppa.**
> • La pipeline è il file `.gitlab-ci.yml`, in radice. Gira solo sulla versione principale (`main`), su un runner installato sul server di prova: costruisce ed esegue tutto lì, con Docker.
> • Lo stack è `devops/deploy/docker-compose.deploy.yml`; la spiegazione completa è `devops/deploy/README.md`.
> • Il piano per i clienti è `docs/devops/PIANO-DISTRIBUZIONE-CLIENTI.md`: immagini da un registro, un file di installazione, un comando unico. Non è ancora iniziato.

JM1.2 - Dove sta nella mappa

In una frase. Installare e aggiornare non è una tessera della mappa: è il lavoro che mette sul server quasi tutte le tessere e le tiene alla versione giusta.

Gli altri libri raccontano un componente. Questo racconta come i componenti arrivano sul server. Per questo sulla mappa non c'è un punto da indicare: ci sono le parti che la pipeline costruisce e riaccende, e quelle che non tocca.



La mappa di ADSWnext: accese le parti che la pipeline installa e aggiorna sul server di prova

Cosa installa e aggiorna. L'app web, con le pagine nuove e le pagine storiche. Il Gateway e il Monitor servizi. I servizi delle Novità: ADEnext, che contiene anche Analytics, SIGEAnext e GPEnext. Il Ponte col COBOL e Macro/Workflow SBH. L'Assistente dei dati, con il suo archivio dei Dati spiegati.

Cosa non tocca. L'Accesso, che controlla utente e password, sta su un altro server. L'ERP legacy/COBOL è il gestionale di sempre: resta dov'è. Gli archivi delle Schermate salvate, dei Modelli di stampa e dei Dati aziendali stanno su un database esterno: i servizi ci si collegano, la pipeline non li sposta e non li ricrea.

Chi la fa partire. Chi sviluppa, quando una modifica approvata entra nella versione principale del codice. E chi preme il pulsante dell'ultimo passo.

Cosa vuol dire per te. Se usi il gestionale, di un aggiornamento vedi solo due cose: per circa un minuto qualche pagina può dare un errore, poi trovi le funzioni nuove. I tuoi dati non si muovono: stanno negli archivi, fuori da ciò che viene ricostruito.

> **Per chi sviluppa.**
> • I servizi accesi sul server di prova sono quelli della variabile `PROFILI_CLASSICO` in `.gitlab-ci.yml`.
> • Analytics non è un servizio a parte: sta dentro ADEnext e si aggiorna con lui.
> • Nel compose di installazione non c'è nessun database dei servizi: è esterno. Fa eccezione l'archivio dell'Assistente dei dati, che è un contenitore dello stack con un volume suo.
> • L'Accesso (Keycloak) si raggiunge con gli indirizzi scritti nel file delle impostazioni: vedi il libro JB - Accesso e permessi.