

Bot-communicatie: Supabase vs Git — onderzoek + advies

Op verzoek van Jamal · 18 juli 2026 · Sjakie (Fable 5)

Jamals vraag: wij laten de bots met elkaar praten via **Supabase** (een online database die de bots afvragen). Een vriend doet het via **Git**. Is dat beter? En: het stoort dat **Conductor** niks doet met vragen van andere bots als hij bezig is.

Eerst: wat is het échte probleem?

Het knelpunt "Conductor negeert vragen als hij bezig is" is **geen transport-probleem** (Supabase vs Git). Het is een **verwerkings**-probleem: een bot die middenin een taak zit, kijkt pas ná die taak in de wachtrij — ongeacht of die wachtrij in Supabase of Git staat. Overstappen op Git lost dit dus **niet** op.

Vergelijking

	SUPABASE (NU)	GIT
Waarvoor gemaakt	Database / berichten	Code-versiebeheer
Live berichten	❑ Redelijk (polling elke 60s; kan instant met Realtime)	⚠ Traag (pull/push, geen realtime)
Meerdere bots tegelijk schrijven	❑ Prima (database regelt het)	❑ Merge-conflicten
Geschiedenis / audit	❑	❑❑ (Git is hierin sterk)
Offline / robuust	Externe dienst nodig (key)	❑ Lokaal + gedeeld

	SUPABASE (NU)	GIT
"Busy bot negeert" opgelost?	☐ nee	☐ nee (zelfde probleem)

Conclusie: Git is **niet beter** voor live bot-berichten — het is gemaakt voor code, heeft dezelfde wacht-vertraging én extra risico's (conflicten, traagheid). Git is wél goed voor gedeelde **kennis/geheugen** — en dat doen we al (gBrain/Mnemosyne = ons gedeelde brein).

Advies (3 stappen, oplopend)

1. Los het échte probleem op — "busy bot negeert":

- Zet bij elke bot een **altijd-luisterende doorgeefluik** (forwarder) die los van de hoofdsessie draait en urgente berichten direct oppakt. Conductor heeft al een telegram-forwarder — daar kunnen bot-vragen doorheen, ook als de hoofdsessie druk is.
- Of: **prioriteit-berichten** die een bezige bot onderbreken (urgent = nu, rest = later).

2. Maak Supabase instant (i.p.v. elke 60s pollen):

- Supabase heeft **Realtime** (websockets): berichten komen dan meteen binnen i.p.v. na max 60 sec. Kleine ingreep, groot verschil.

3. Git behouden waar 't sterk is:

- Blijf Git/gBrain gebruiken voor gedeeld **geheugen en kennis** (dat werkt goed). Niet voor live chat tussen bots.

Kort samengevat

- **Niet** overstappen naar Git voor de berichten — dat is een stap terug.
- **Wel** het echte probleem aanpakken: een altijd-luisterend doorgeefluik + eventueel Supabase Realtime, zodat een bezige Conductor toch direct reageert.
- Git blijft waardevol voor het gedeelde geheugen.

Vervolg: dit afstemmen met Conductor (hij bouwt de queue-infrastructuur) voordat we iets wijzigen.