

Architektur — Executive Inbox

Für: Jürgen A. C. Wilms (Vorstand, NEG AG · Northgate Enterprises Group AG, Leipzig)

Plattform: n8n · Microsoft 365 (Outlook) · WhatsApp · Teams

Prinzip: Nur Entwürfe. Nichts wird ohne seine Bestätigung gesendet. Minimale Labels, maximal natives n8n.

1. Ziel

Dem Vorstand ein Postfach geben, das auf seinem Microsoft-Stack:

1. **Rauschen filtert** — nur Relevantes erreicht ihn.
2. **Automatisch labelt** — jede E-Mail mit einer Farbe (6 Themen) + einem `Sofort` -Flag.
3. **Threads auf Deutsch zusammenfasst**.
4. **Antworten in seiner Stimme entwirft** (Phase 2), die er per WhatsApp freigibt — nie automatisch gesendet.
5. **Einen Morgen-Digest an Teams sendet**.

Er legt nie ab, wählt nie ein Label, öffnet nie einen Ordner. Die KI taggt; er sieht die Farbe oder filtert nach Tag.

2. Ergebnis des Ponytail-Reviews (was wir gegenüber den Vorlagen gestrichen haben)

Die drei Quell-Vorlagen (Full Inbox Manager, Sally-CSR, Ruby-Sales) basieren auf Gmail + Telegram + GoHighLevel und bauen vieles von Hand nach, das n8n heute nativ kann. Gegen die n8n-Node-Fähigkeiten verifiziert:

Vorlage machte	Wir machen stattdessen	Ersparnis
2 LLM-Durchläufe (Kategorisierung + Wichtigkeit)	1 strukturierter LLM-Node → <code>{topic, sofort}</code>	–1 KI-Aufruf, –2 Nodes
HTTP/Graph-Aufrufe zum Setzen von Labels	Outlook message: Update (Kategorien) + <code>Move</code>	–7 HTTP-Nodes
Eigene Telegram/WhatsApp-Freigabelogik	WhatsApp Send and Wait for Response (nativ)	–1 Sub-Flow
base64-„Reformat Thread“-Code-Node	Outlook liefert dekodierten <code>body</code> + <code>conversationId</code>	–1 Code-Node
Thread-Abruf + Merge + <code>splitInBatches</code> in der Triage	Klassifizierung der letzten Nachricht; Thread nur zum Entwerfen abrufen	–3 Nodes (Phase 1)
GoHighLevel-CRM-Maschinerie	Pipedrive aus Phase 1 entfernt (YAGNI)	–5 Nodes
Gmail + Schedule (Dublekken-Bug)	Outlook Trigger (jede Mail einmal)	Bug entfernt

Ergebnis: ~30-Node-Vorlage → ~9-Node-Phase 1. Wir übernehmen *Muster und Logik*, nicht Nodes.

3. Label-Taxonomie (fixiert)

6 Themen-Tags (eines pro E-Mail) — Outlook-Kategorien, farbcodiert, nichts wird verschoben:

Tag	Farbe	Fängt ab
Investoren & Finanzierung	 Lila	Investoren, Banken, Gesellschafter, Darlehen
Transaktionen	 Blau	An-/Verkauf, LOI, Kaufvertrag, Notar, Makler
Bestand & Mieter	 Grün	Vermietung, Verwaltung, Nebenkosten, Mieter
Bau & Projekte	 Orange	Bauprojekte, Architekten, Genehmigungen, Behörden
Recht	 Rot	Anwälte, Compliance, Streit, Schaden/Haftung
Intern & Backoffice	 Grau	Team, HR, Admin, Protokolle

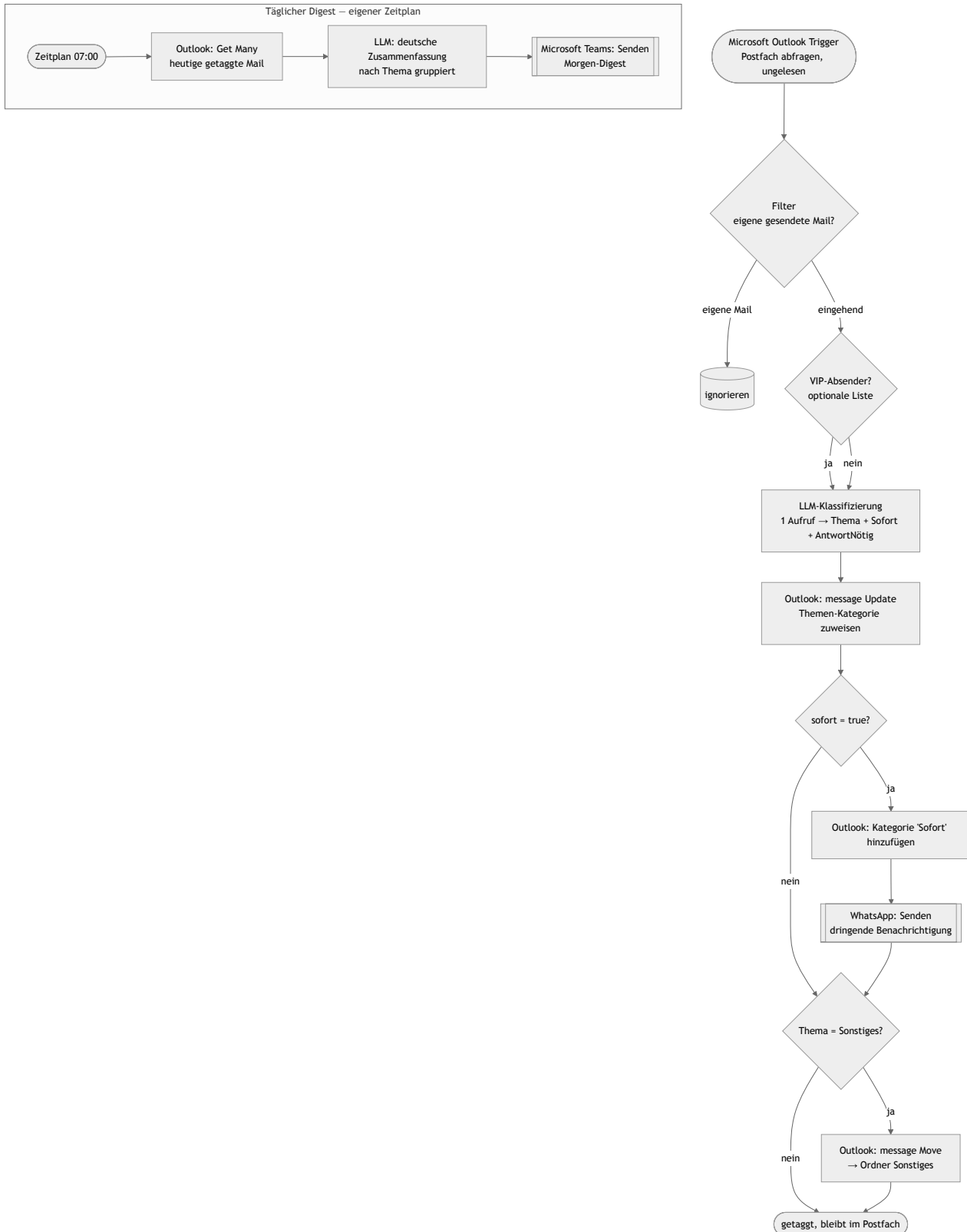
+ **1 Dringlichkeits-Flag** (kombinierbar mit jedem Thema): **Sofort** 

+ **1 Ordner** (die einzige Verschiebung): **Sonstiges** ← Newsletter/Spam werden automatisch aus dem Postfach abgelegt.

Suchordner je Tag vorab angelegt → Seitenleisten-„Ordner“, die eigentlich gespeicherte Filter sind (nichts wird versteckt).

*Bewusst aus seinem Postfach ausgeschlossen: Rechnungen/DATEV (bereits über die **rechnung@neg.ag** -Pipeline abgewickelt) und Vertrieb/Marketing (nur Digest).*

4. Architektur — Phase 1 (Triage · Label · Digest)

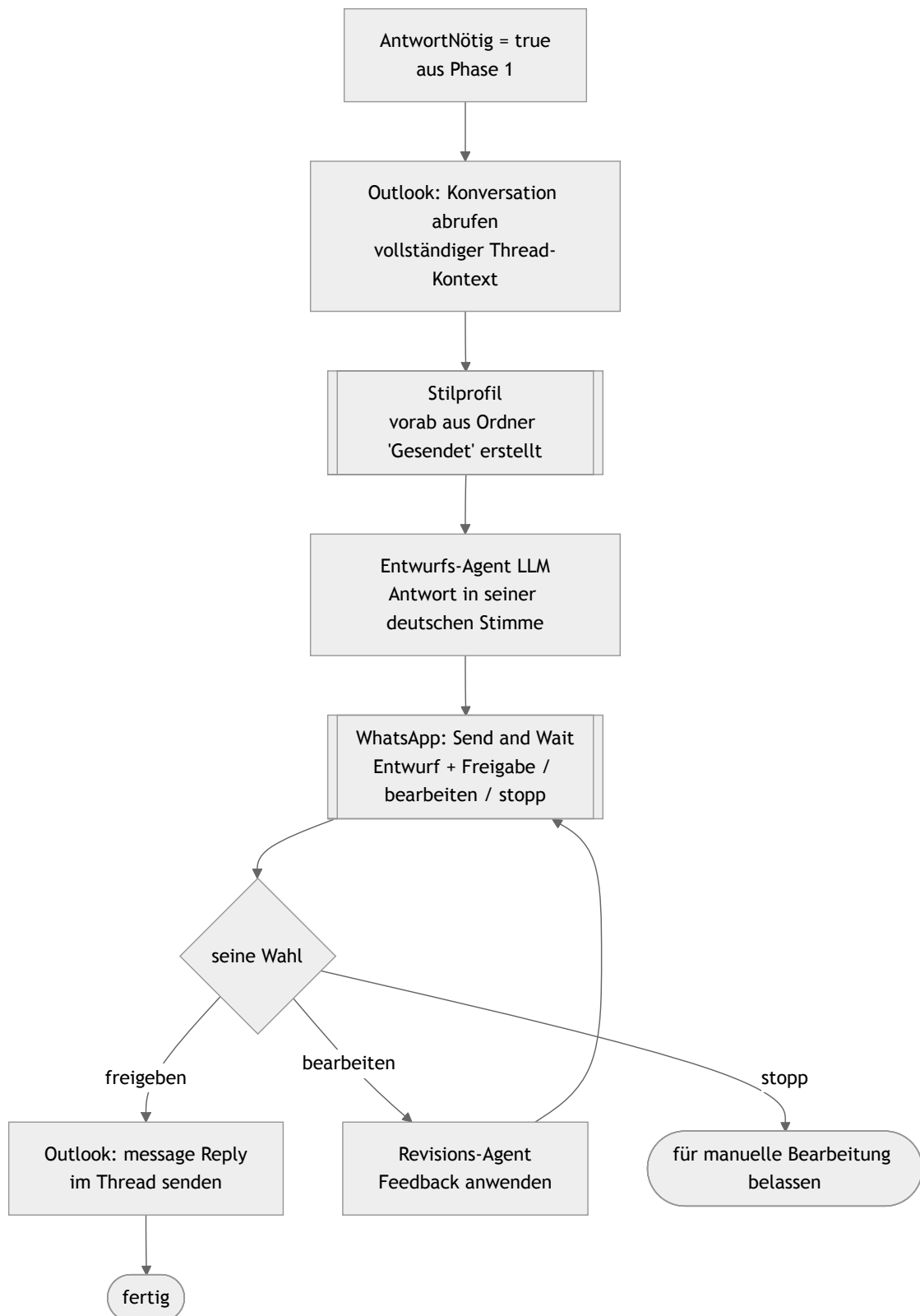


Node-Liste Phase 1 (~9 + Digest):

1. `microsoftOutlookTrigger` — Postfach abfragen (ungelesen), jede Mail einmal
2. `filter` — eigene Absenderadresse verwerfen
3. *(optional)* `if` — VIP-Allowlist als Schnellpfad

4. `lmChatOpenAi` + strukturierte Ausgabe — **ein Aufruf** → `{ topic, sofort, needsReply }` (*alt.: `nativer textClassifier`, falls die Dringlichkeits-Achse entfällt*)
5. `microsoftOutlook` `message:Update` — Themen-Kategorie zuweisen
6. `if` + `message:Update` — `Sofort` -Flag bei Dringlichkeit hinzufügen
7. `whatsApp` `message:Send` — dringende Benachrichtigung
8. `if` + `message:Move` — `Sonstiges` → Ordner
9. **Digest (separat):** `scheduleTrigger` → `microsoftOutlook` `message:Get Many` → `lmChatOpenAi`
Zusammenfassung → `microsoftTeams` senden

5. Architektur — Phase 2 (Entwurf in seiner Stimme · Freigabe per WhatsApp)



Aus Sally/Ruby übernommen (Muster): der Entwurfs-Agent + strukturierte Ausgabe, die Freigabe-/Bearbeiten-/Stopp-Schleife, der Revisions-Agent. **Ausgetauscht:** Gmail-Reply → Outlook message: Reply ; Telegram → WhatsApp Send and Wait ; Kamexa-RAG → **entfernt**; ergänzt um **Stilprofil** (einmalige Destillation von ~150 gesendeten E-Mails → wiederverwendbarer Prompt-Block, monatlich aktualisiert).

Noch schlankere Alternative: die WhatsApp-Schleife überspringen und Outlook `draft: Create` nutzen — die KI-Antwort landet in seinem **Entwürfe-Ordner**, er bearbeitet/sendet nativ in Outlook. (Beides anbieten; WhatsApp ist schneller für einen Phone-First-Vorstand.)

6. Node- & Credential-Inventar

Fähigkeit	Node	Native Operation	Credential
Eingang	<code>microsoftOutlookTrigger</code>	poll	M365 OAuth2
Lesen / labeln / verschieben / antworten / entwerfen	<code>microsoftOutlook</code>	Update · Move · Reply · Draft:Create · Get Many	M365 OAuth2
Klassifizieren	<code>lmChatOpenAi</code> (+ strukturierte Ausgabe)	—	OpenRouter
Dringende Benachrichtigung + Freigaben	<code>whatsApp</code>	Send · Send and Wait	WhatsApp Business Cloud
Digest	<code>microsoftTeams</code>	channelMessage / Send and Wait	M365
CRM (nur Phase 3)	<code>pipedrive</code>	Search	Pipedrive API

7. Einmalige Einrichtung (skriptgesteuert)

- 6 Postfach-**Kategorien** mit den obigen Farben + `Sofort` anlegen.
- Den Ordner `Sonstiges` anlegen.
- 6 **Suchordner** — einen je Thema — für die Seitenleiste anlegen.
- Das **Stilprofil** aus seinem Ordner „Gesendet“ erstellen (Phase 2).
- Credentials (M365, OpenRouter, WhatsApp, Teams) in n8n hinterlegen.

8. Build-Reihenfolge

- **Phase 1 — Triage + Digest.** Kein Entwerfen. Er vertraut zuerst der Filterung + Farbcodierung. ~9 Nodes.
- **Phase 2 — Stimm-Entwürfe + WhatsApp-Freigabe.** Ergänzt um Stilprofil + Entwurf-/Freigabe-Schleife.
- **Phase 3 — Feinschliff.** Leichter Pipedrive-Kontext, VIP-Listen-Tuning, Snooze/FYI-Regeln.

9. Offene Entscheidungen

- **Klassifizierer:** ein strukturierter LLM-Aufruf (Thema + sofort, empfohlen) vs. nativer `textClassifier` (nur Thema, Regel für `Sofort` ergänzen).
- **Entwurfs-Freigabe:** WhatsApp `Send and Wait` (empfohlen, phone-first) vs. Outlook `draft: Create` (Prüfung in Outlook).

- **Digest-Kanal:** Teams-Kanal vs. Teams-DM vs. E-Mail an sich selbst.
-

Alles oben gegen die Live-Fähigkeiten der n8n-Nodes via n8n-mcp validiert.