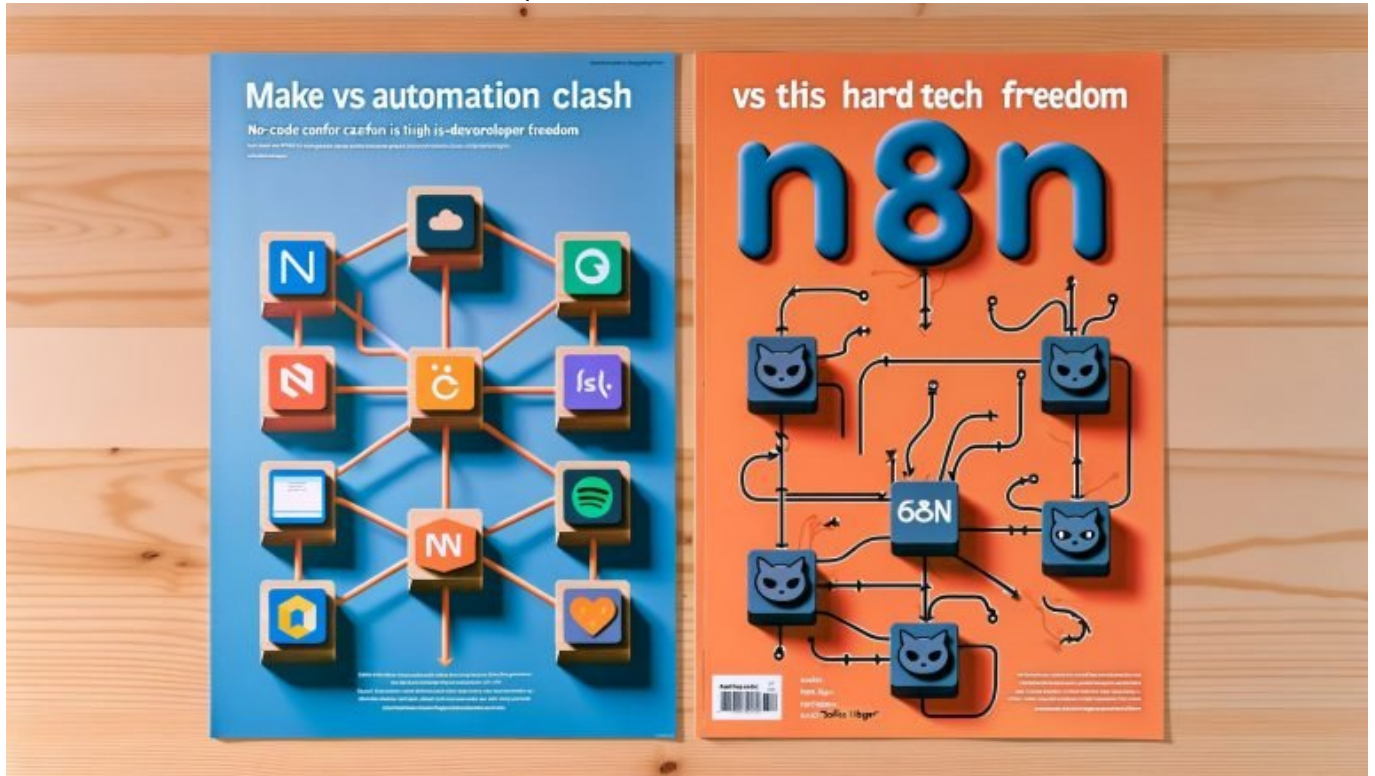


# Make vs n8n: Welches Automatisierungstool passt besser?

Category: Tools

geschrieben von Tobias Hager | 29. September 2025



# Make vs n8n: Welches Automatisierungstool passt besser?

Du willst dein Business automatisieren und hast die Nase voll von Copy-Paste-Hölle, aber weißt nicht, ob du dich für Make oder n8n entscheiden sollst? Willkommen im Dschungel der Automatisierungstools, in dem Marketing-Gurus und Tech-Evangelisten alles versprechen, aber selten ehrlich über Schwächen reden. Schluss mit weichgespülten Vergleichen: Hier bekommst du die knallharte, technisch fundierte Antwort, welches Tool 2024 wirklich liefert – und welches du besser links liegen lässt.

- Make und n8n: Was diese Automatisierungstools wirklich leisten – und was

nicht

- Die wichtigsten Unterschiede im Funktionsumfang, der Integrationstiefe und beim API-Handling
- On-Premise vs. Cloud: Datenschutz, Hosting und Compliance im Realitätscheck
- No-Code, Low-Code, Pro-Code: Für wen Make ein Traum ist – und warum n8n Entwicklern mehr Spielraum gibt
- UI/UX, Usability und die unschönen Wahrheiten der „intuitiven“ Workflows
- Performance, Skalierbarkeit und das böse Erwachen bei größeren Prozessen
- Preisstrukturen, versteckte Kosten und das Märchen vom günstigen Einstieg
- Security, Datenschutz und DSGVO: Wer nimmt's wirklich ernst?
- Konkrete Use Cases und Best Practices für Marketing, E-Commerce und Entwicklung
- Fazit: Die gnadenlos ehrliche Empfehlung, wann Make oder n8n für dich Sinn macht

Wenn du Make oder n8n googelst, bekommst du erst mal eine Lawine aus Buzzwords, heiler Welt und „Jeder kann alles“-Versprechen. Die Wahrheit: Automatisierung ist kein Ponyhof, sondern ein knallharter Wettbewerbsvorteil – wenn du die Technik im Griff hast. Beide Tools – Make (früher Integromat) und n8n – haben ihre Daseinsberechtigung, aber sie verfolgen radikal unterschiedliche Philosophien. Du willst nicht noch eine weitere weichgespülte Vergleichstabelle? Gut so. Hier kommt das technische Duell, das deinem Tech-Stack wirklich auf den Zahn fühlt. Und ja, das kann wehtun. Aber besser jetzt als nach dem dritten abgebrochenen Workflow um 2 Uhr morgens.

## Make vs n8n: Architektur, API-Handling und technische Basis im Vergleich

Wer Make oder n8n nur als „Automatisierungstool“ abstempelt, unterschätzt das Thema gewaltig. Die Architekturfrage ist der erste Stolperstein: Make ist ein strikt Cloud-basiertes SaaS-Produkt, das in tschechischer Pragmatik gebaut wurde. Es setzt auf eine proprietäre Infrastruktur – du hast null Zugriff auf den Quellcode, null Einfluss auf die Server und bist auf das angewiesen, was Make bereitstellt. Das ist bequem, solange du keine Compliance-Hürden nehmen musst oder alles nach Standard abläuft.

n8n spielt in einer ganz anderen Liga. Die Open-Source-Plattform lässt sich on-premise, in der eigenen Cloud, per Docker oder als Managed Service betreiben. Der Quellcode ist offen, die Architektur modular – perfekt für alle, die tief in Prozesse und Security eingreifen wollen. Das API-Handling ist bei n8n deutlich flexibler: Native Nodes für REST, GraphQL, SOAP, HTTP Request – alles da, alles konfigurierbar, keine künstlichen Limitierungen. Wer mit komplexen APIs arbeitet, lacht über die Make-typischen „Connector Constraints“.

Make punktet dafür mit einer riesigen Anzahl an fertigen Modulen („Apps“), die fast jede Mainstream-API abdecken. Der Haken: Sobald du etwas machen willst, das nicht im Baukasten steckt, geht's ans Basteln – und an die (oft undokumentierten) HTTP-Module. Während du bei n8n mit direkt editierbaren JavaScript-Funktionen, Expressions und Data Mapping in den Code tauchen kannst, bleibt Make immer einen Schritt hinter der echten Entwicklerfreiheit zurück.

Ein kurzer Realitätscheck für die API-Nerds:

- n8n: REST, GraphQL, SOAP, Webhooks, OAuth2, JWT – alles nativ, alles konfigurierbar. Eigene Nodes via Community oder Custom Node SDK in wenigen Stunden gebaut.
- Make: API-Integration per HTTP-Request-Modul, aber kein echtes SDK für eigene Module. REST, SOAP, OAuth – ja, aber alles unter der Haube von Make, mit limitierten Einstellungen.

Fazit: Wenn du maximale Freiheit, vollständige Kontrolle und die Option auf Self-Hosting willst, ist n8n dein Werkzeug. Make ist das Tool für alle, die schnell und möglichst ohne technische Detailarbeit Prozesse zusammenschrauben wollen – solange sie im vordefinierten Rahmen bleiben.

## Integrationen, Konnektoren und Workflow-Design: Wo Make glänzt, wo n8n aufholt

Automatisierung lebt und stirbt mit Integrationen. Make wirbt mit über 1.500 fertigen Apps – von Google Workspace bis Shopify, von Slack bis Facebook Ads. Der Einrichtungsprozess ist tatsächlich idiotensicher: Account verbinden, Trigger setzen, Aktionen klicken, fertig. Für Standard-Workflows ist das unschlagbar schnell. Marketing-Teams, die nicht tiefer in die Technik einsteigen wollen, sind hier goldrichtig – solange sie sich mit den Make-Grenzen arrangieren.

n8n war lange Zeit Integrations-Außenseiter, hat aber spätestens mit Version 1.x und dem Node-Marktplatz massiv aufgeholt. Die Zahl der nativen Nodes wächst rasant, und über die Open-Source-Community kommen ständig neue hinzu. Was fehlt, baust du mit wenigen Zeilen JavaScript oder TypeScript selbst. Kein Bock auf Coden? Dann helfen dir vorgefertigte Vorlagen („Workflows“) und Community-Templates. Der entscheidende Unterschied: n8n lässt dich auch dort weiterarbeiten, wo Make dich ausbremst. Du kannst JSON-Strukturen vollständig manipulieren, Variablen dynamisch setzen, verschachtelte Loops und Bedingungen einbauen – alles, was für komplexe Automatisierungen nötig ist.

Das Workflow-Design ist bei Make optisch cleaner: Drag & Drop, alles hübsch linear, jeder Step ein Modul. Aber wehe, du willst Schleifen, verschachtelte If/Else-Logik oder dynamische Branches – dann wird's schnell hässlich. n8n setzt auf ein Node-basiertes System, das zwar anfangs gewöhnungsbedürftig

ist, aber echte Programmierer-Freiheit bietet. Workflows können beliebig verzweigen, rekursiv aufgerufen werden und sind dank Expressions und Code Nodes maximal flexibel. Das ist komplexer – aber für Wachstum und echte Prozessautomatisierung unverzichtbar.

Die bittere Wahrheit: Wer einfache Social-Media-Posts oder Newsletter-Trigger automatisieren will, ist mit Make glücklich. Wer Marketing, E-Commerce oder SaaS wirklich skaliert automatisieren will, stößt bei Make schnell an Grenzen. n8n ist weniger hübsch, aber technisch das bessere Fundament.

## No-Code vs. Low-Code: Usability, Erweiterbarkeit und die Realität im Alltag

Make verkauft sich als No-Code-Plattform. Und ja, für klassische Office-Automatisierung, Marketing-Standardaufgaben und „Wenn dies, dann das“-Workflows geht das auf. Das UI ist selbsterklärend, die Dokumentation solide, und für die meisten Aufgaben brauchst du kein Codeverständnis. Aber: Sobald du von der Norm abweichst – eigene Authentifizierung, komplexe Datenstrukturen, asynchrone Prozesse – wird No-Code zur Sackgasse. Das HTTP-Modul ist ein Witz gegen das, was Entwickler brauchen. Fehlerhandling? Eher rudimentär.

n8n positioniert sich als Low-Code-Plattform. Das ist ehrlich. Du kannst alles per Drag & Drop zusammenbauen, aber du kannst jederzeit mit JavaScript eingreifen, Expressions nutzen und eigene Nodes entwickeln. Das ist nicht für jeden – aber für Teams, die wachsen oder anspruchsvolle Automatisierungen brauchen, ein Muss. Das UI ist nicht ganz so fancy wie bei Make, aber nach kurzer Einarbeitung produktiver. Die Lernkurve ist steiler, die Möglichkeiten aber auch.

Ein paar harte Fakten aus dem Alltag:

- Make: Schnell, solange du im Baukasten bleibst. Erweiterbarkeit gleich null. Fehlerhandling und Debugging nur mit Workarounds praktikabel.
- n8n: Steilere Lernkurve, aber alles möglich, was technisch geht. Fehlerhandling, Logging, Branching, Conditional Logic, Webhooks – alles granular steuerbar.

Wer auf No-Code hofft und nie eigene Prozesse braucht, bleibt bei Make. Wer Low-Code akzeptiert und wachsen will, kommt an n8n nicht vorbei.

## On-Premise vs. Cloud:

# Datenschutz, Skalierbarkeit und DSGVO im Realitätscheck

Spätestens beim Thema Datenschutz wird's ernst. Make läuft ausschließlich in der Cloud, primär auf europäischen Servern (laut Anbieter), aber du hast null Kontrolle darüber, wo und wie deine Daten verarbeitet werden. Für viele Startups und KMUs kein Problem, für Konzerne oder Unternehmen mit echten Compliance-Anforderungen ein Showstopper. DSGVO? Make gibt sich Mühe, aber du bist und bleibst abhängig vom Anbieter.

n8n ist die Antwort für alle, die Datenhoheit und Datenschutz wirklich ernst nehmen. On-Premise-Installation? Kein Problem. Eigenes Hosting in der AWS, Azure oder auf dem Blech unter dem Schreibtisch? Klar. Du bestimmst, was gespeichert, geloggt oder verarbeitet wird. Die DSGVO-Konformität liegt komplett in deiner Hand (und Verantwortung) – das ist ein Vorteil für Profis, aber eine Hürde für alle, die keinen eigenen Admin haben.

Skalierbarkeit? Bei Make bezahlt man für Szenarien, Runs und Datenvolumen – und landet bei echten Business-Prozessen schnell im vierstelligen Bereich. n8n skaliert in der eigenen Umgebung, ohne künstliche Limits. Klar, du brauchst Know-how für Load Balancing, Horizontal Scaling und Monitoring – aber das ist Tech-Luxus für alle, die ernsthaft skalieren wollen.

Ein kurzer Datenschutz-Realitätscheck:

- Make: Cloud only, keine On-Premise-Option, DSGVO-konform laut Eigenangabe, aber Kontrolle bleibt beim Anbieter.
- n8n: Vollständige Datenhoheit, Self-Hosting, Compliance nach eigenen Standards, volle Auditierbarkeit.

Wer Compliance als Floskel sieht, kann Make nehmen. Wer Datenschutz leben muss, kommt an n8n nicht vorbei.

## Performance, Preisstrukturen und das böse Erwachen bei echten Workloads

Der Marketing-Sprech verspricht „unbegrenzte Automatisierung“ für kleines Geld. Die Realität: Make ist günstig, solange du ein paar hundert Runs pro Monat hast. Aber wehe, du wächst. Jedes Szenario, jeder Run, jede API-Aktion kostet Geld. Bei komplexen Prozessen oder saisonalen Peaks explodieren die Kosten – und das Monitoring ist eher Alibi als echtes Controlling.

n8n ist im Basispaket kostenlos (Open Source). Die Cloud-Version kostet – aber du kannst jederzeit auf eigenes Hosting ausweichen. Du zahlst für zusätzliche Features (z.B. SSO, Team-Management) im Enterprise-Bereich, aber

nicht für Runs oder Szenarien. Performance hängt von deiner Infrastruktur ab: Ein Kubernetes-Cluster skaliert endlos, eine VM mit 2GB RAM hat irgendwann Feierabend – aber du hast die Wahl.

Was oft verschwiegen wird: Make hat Limits bei Payload-Größe, Ausführungszeit und API-Calls. Komplexe Workflows werden künstlich gebremst oder laufen gar nicht durch. n8n ist limitiert nur durch deine Hardware – und dein technisches Know-how. Dafür musst du dich aber auch um Updates, Security-Patches und Monitoring selbst kümmern.

Preisfalle Make:

- Starter-Tarif: Kaum nutzbar für mehr als ein Hobbyprojekt
- Business-Tarif: Schnell bei 100–300 €/Monat, wenn du wirklich automatisierst
- Enterprise: Individuell – sprich, teuer und undurchsichtig

n8n:

- Open Source: 0 € – solange du selbst hostest und administrierst
- Cloud: Ab 20 €/Monat, keine Begrenzung bei Workflows, aber Features für Teams kosten extra
- Enterprise: Transparente Preise, volles Feature-Set, Self-Hosting oder Managed

Fazit: Wer klein bleibt und keine Kontrolle braucht, zahlt bei Make wenig. Wer wächst, zahlt drauf. n8n ist für Profis günstiger – rechnet sich aber nur, wenn du Admin-Know-how hast.

# Security, Fehlerhandling und Community: Die unterschätzten Faktoren

Make wirbt mit Security-Zertifikaten, ISO-Standards, 2FA und verschlüsseltem Datentransport. Alles schön und gut – aber du bist komplett vom Anbieter abhängig. Bugfixes dauern, Feature-Wünsche verpuffen im Support-Chat. Fehlerhandling in den Workflows? Eher rudimentär: Entweder läuft alles, oder du kriegst eine kryptische Fehlermeldung. Rollbacks, Retries, echte Testumgebungen? Fehlanzeige.

n8n lebt von einer aktiven Community. Bugs werden schnell gefixt, Feature-Requests landen auf GitHub, und du kannst selbst Hand anlegen. Fehlerhandling ist granular: Jeder Node kann individuell auf Fehler reagieren, du kannst Try/Catch-Mechanismen aufbauen, Logging und Monitoring selbst konfigurieren. Das ist mehr Aufwand, aber maximale Kontrolle. Security? Liegt in deiner Hand – von HTTPS bis Firewall, von Zugriffskontrolle bis Data Logging. Du hast die Verantwortung – und die Macht.

Ein paar harte Wahrheiten zur Community:

- Make: Support via Ticketsystem, Feature-Requests dauern, Community eher passiv
- n8n: Discord, GitHub, Foren – schnelle Hilfe, Open-Source-Entwicklung, echte Mitgestaltung

Wer auf Vendor-Lock-in und Support-Hotlines steht, bleibt bei Make. Wer mitgestalten, anpassen und wachsen will, ist bei n8n richtig.

# Konkrete Use Cases: Marketing, E-Commerce und Entwicklung im Praxistest

Marketing-Teams lieben Make für die klassische Automatisierung: Social-Media-Posts aus Google Sheets, Newsletter-Subscriber in CRM schieben, einfache Reports per E-Mail. Alles out-of-the-box, alles hübsch klickbar. E-Commerce? Bestellbenachrichtigungen, Lagerupdates, Tracking-Mails – solange die API der Shop-Software unterstützt wird, läuft's. Aber wehe, du willst mehrere Systeme dynamisch verknüpfen, Produktdaten transformieren oder komplexe Bedingungen einbauen: Dann wird's schnell eng.

n8n ist für Entwickler und Tech-affine Teams das Paradies: Du kannst API-Endpoints bauen, eigene Webhooks aufsetzen, Daten transformieren, externe Services anbinden, Slack-Bots entwickeln, Datenbanken synchronisieren, AI-Services integrieren. Wer im SaaS-Bereich arbeitet, Microservices orchestriert oder komplexe Pipelines braucht, kommt an n8n nicht vorbei. Marketing und E-Commerce profitieren, wenn Prozesse über Standard-Workflows hinausgehen – z.B. dynamisches Campaign Management, Multichannel-Tracking oder individualisierte Reportings.

Best Practices für beide Tools – Schritt für Schritt:

- Make: 1. Account anlegen, 2. App verbinden, 3. Szenario auswählen, 4. Trigger setzen, 5. Aktionen definieren, 6. Testen, 7. Aktivieren – fertig.
- n8n: 1. Instanz aufsetzen (Self-Hosted oder Cloud), 2. Node-Auswahl treffen, 3. Workflow modellieren, 4. Expressions und Logik einbauen, 5. Testen und Debuggen, 6. Deployen, 7. Monitoring und Alerting konfigurieren.

Fazit: Für die Masse der Standardaufgaben ist Make schneller. Für alles, was über die Basics hinausgeht, ist n8n technisch überlegen.

## Fazit: Make oder n8n – Wer

# gewinnt das Rennen?

Wer Automatisierung als Click-and-Forget-Spielzeug sieht, wird mit Make glücklich – solange die Prozesse einfach, die Anforderungen überschaubar und die Datenschutz-Fragen egal sind. Make ist der Einstieg in die Welt der Automatisierung: schnell, bequem, aber limitiert. Wer aber Prozesse ernsthaft digitalisieren, wachsen oder skalieren will, stößt bei Make schnell an unsichtbare Wände. n8n ist das Werkzeug für Profis, die Komplexität nicht scheuen, sondern beherrschen wollen. Es ist flexibler, erweiterbarer, datenschutzfreundlicher – aber auch anspruchsvoller in der Einrichtung und Wartung.

Die Wahrheit ist unbequem: Kein Tool ist für alle perfekt. Wer No-Code und schnelle Ergebnisse für Standardaufgaben sucht, bleibt bei Make. Wer Kontrolle, Erweiterbarkeit und Skalierbarkeit will, setzt auf n8n. In 90% aller Enterprise- und Entwickler-Szenarien gewinnt n8n. Im schnellen Marketing-Alltag ohne IT-Support? Punkt für Make. Wer heute für morgen bauen will, kommt an n8n nicht vorbei – alles andere ist Spielerei.