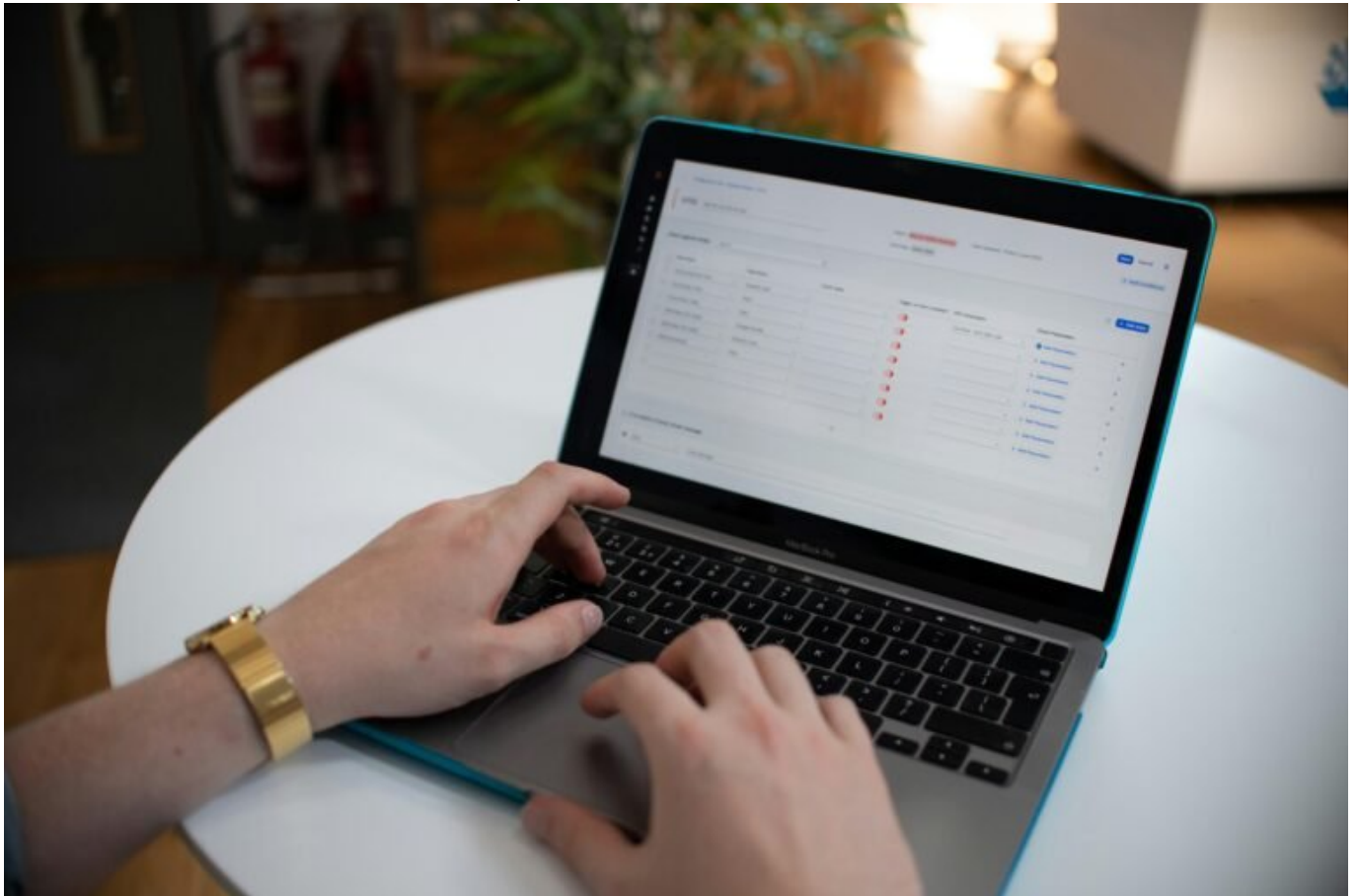


Make: Mehr als nur ein Befehl im Online-Marketing

Category: Online-Marketing

geschrieben von Tobias Hager | 7. Februar 2026



Make: Mehr als nur ein Befehl im Online-Marketing

Mach mal schnell 'ne Kampagne. Mach mal SEO. Mach mal Conversion. Willkommen in der Welt des "Make" – dem wohl am häufigsten missverstandenen Befehl im Online-Marketing. Was ursprünglich aus der Entwicklerwelt stammt, ist längst zum Buzzword mutiert – und wird von selbsternannten Digitalhelden inflationär missbraucht. Aber Make ist mehr als ein Tool, mehr als Automatisierung, mehr

als ein Baukasten für faule Marketer. Es ist eine Philosophie, ein technisches Framework – und ein potenzieller Gamechanger. Vorausgesetzt, man weiß, was man da eigentlich tut.

- Was Make (ehemals Integromat) wirklich ist – und warum es kein “No-Code-Spielzeug” ist
- Wie Make die Automatisierung im Online-Marketing auf ein neues Level hebt
- Warum Make kein Ersatz für technisches Verständnis ist – sondern dessen Verstärker
- Die wichtigsten Anwendungsbereiche: von Lead-Generierung bis Reporting
- Best Practices, Patterns und Fallstricke in der Make-Automatisierung
- Die Wahrheit über Make vs. Zapier: Wer gewinnt den Automatisierungskrieg?
- Wie du Make sinnvoll in dein MarTech-Stack integrierst – ohne die Kontrolle zu verlieren
- Fehler, die dir das Marketing-Leben zur Hölle machen (und wie du sie vermeidest)
- Security, API-Limits und Skalierung: Die dunkle Seite der Automatisierung
- Warum “Make it work” nicht reicht – und wie du Make wirklich meisterst

Was ist Make wirklich? Automatisierung für Leute, die wissen, was sie tun

Make, früher bekannt als Integromat, ist ein cloudbasiertes Automatisierungstool, das es dir ermöglicht, komplexe Workflows zwischen verschiedenen Webdiensten zu erstellen – ohne Zeile Code. Klingt nach No-Code? Ist es auch – aber nur auf den ersten Blick. Denn wer glaubt, Make sei ein Tool für Technikverweigerer, hat es entweder nie benutzt oder nie verstanden. Anders als Zapier, das oft mit simplen Trigger-Aktion-Ketten arbeitet, erlaubt Make komplexe Szenarien, HTTP-Requests, Iteratoren, Router, Error-Handler und vollständige API-Integration. Kurz: Make ist kein Spielzeug. Es ist ein Engine Room.

Im Kern basiert Make auf sogenannten Szenarien, die aus Modulen bestehen. Jedes Modul ist ein API-Endpunkt, der Daten sendet, empfängt oder transformiert. Diese Module lassen sich logisch verknüpfen – mit Bedingungen, Schleifen, Variablen und sogar JavaScript-Code. Du willst neue Leads aus einem Typeform-Formular automatisch in dein CRM schicken, ihnen eine personalisierte Mail senden, sie in eine Google-Sheet-Tabelle eintragen und das Ganze in Slack melden? Kein Problem – Make macht's möglich. Und zwar reproduzierbar, skalierbar und transparent.

Doch hier beginnt der Haken: Make ist mächtig, aber nicht fehlertolerant. Wer nicht genau weiß, wie HTTP-Codes, API-Ratenlimits oder JSON-Strukturen funktionieren, fliegt schneller aus der Kurve, als er “Webhook” sagen kann.

“Make” ist also nicht nur ein Tool, sondern ein Commitment: zu technischer Präzision, zu strukturiertem Denken und zu einem Automatisierungsverständnis, das tiefer geht als “wenn A, dann B”.

Make im Online-Marketing: Die Königsdisziplin der Prozessautomatisierung

Online-Marketing ist ein Datensumpf. Leads hier, Metriken dort, Kampagnen in fünf Tools verteilt und der Kunde will “nur schnell mal” wissen, wie’s läuft. Wer versucht, das alles manuell zu orchestrieren, wird wahnsinnig – oder ineffizient. Genau hier kommt Make ins Spiel. Es bringt Struktur in das Chaos, indem es repetitive Aufgaben automatisiert, Tools miteinander reden lässt und Datenflüsse in Echtzeit steuert. Kurz: Make ist das, was Marketing-Teams brauchen, um nicht im eigenen Tech-Stack zu ersaufen.

Ein typisches Beispiel: Lead-Generierung. Nutzer füllt ein Formular aus (z. B. in Typeform oder HubSpot), Make triggert ein Szenario, prüft die Daten auf Vollständigkeit, reichert sie via API mit Zusatzinfos an (z. B. über Clearbit), speichert sie im CRM, verschickt eine Trigger-Mail, legt einen Task im Projekttool an und loggt das Ganze in einer internen Datenbank. Alles ohne einen einzigen menschlichen Klick. Das spart nicht nur Zeit, sondern verhindert auch Fehler – vorausgesetzt, die Logik ist sauber gebaut.

Weitere klassische Use Cases:

- Automatisiertes Reporting via Google Sheets oder Data Studio
- Social-Media-Distribution mit Zeitsteuerung und Asset-Management
- Alert-Systeme bei KPI-Abweichungen oder Datenverlust
- Dynamische Budgetüberwachung von Paid-Kampagnen per API

Make ist damit nicht nur Automatisierung, sondern Qualitätskontrolle, Datenintegrität und strategische Entlastung in einem. Aber nur, wenn man es richtig einsetzt.

Make vs. Zapier: Warum Make die härtere, aber mächtigere Wahl ist

Zapier war der Early Mover im No-Code-Automatisierungsmarkt – und ist bei vielen immer noch der Platzhirsch. Warum? Weil es einfach ist. Trigger, Aktion, fertig. Aber genau das ist das Problem: Zapier stößt viel schneller an seine Grenzen. Keine komplexe Logik, keine echten Schleifen, kaum Debugging. Wer mehr als nur “Wenn jemand Mailchimp abonniert, dann schicke

Slack-Nachricht“ machen will, kommt mit Zapier nicht weit.

Make hingegen ist ein Biest. Es erlaubt Multistep-Flows mit parallelen Pfaden, bedingter Logik, Error-Handling, Webhooks, JSON-Parsing, regulären Ausdrücken und sogar eigenen Variablen. Der Editor ist visuell, aber technisch. Wer einmal einen 40-Schritte-Workflow mit mehrfacher Verzweigung gebaut hat, weiß, was gemeint ist. Make ist Zapier auf Steroiden – aber mit Anspruch.

Ein weiterer Unterschied: Die API-Integrationstiefe. Während Zapier oft nur die gängigsten API-Parameter freigibt, erlaubt Make in vielen Modulen den Zugriff auf alle API-Felder – inklusive benutzerdefinierter Header, Query-Strings und Body-Payloads. Das bedeutet: Du kannst praktisch alles tun, was die API hergibt. Keine Gängelung durch UI-Beschränkungen. Kein “geht nicht”, weil das Tool limitiert ist.

Fazit? Wer einfache Workflows will, nimmt Zapier. Wer komplexe Prozesse automatisieren will, nimmt Make. Punkt.

Fehler in Make, die dir das Marketing zerschießen – und wie du sie vermeidest

Make ist mächtig, aber unforgiving. Ein falsch gesetzter Iterator, ein nicht abgefangener API-Fehler, ein vergessener Breakpoint – und dein kompletter Funnel läuft Amok. Deshalb ist Fehlervermeidung kein netter Bonus, sondern Pflichtprogramm. Die häufigsten Fehlerquellen im Überblick:

- Keine Error-Handler: Szenarien ohne Error-Routing laufen bei einem API-Fehler ins Leere – und du merkst es erst, wenn der Kunde fragt, wo sein Lead geblieben ist.
- Übersehenes Rate-Limiting: Viele APIs haben Call-Limits. Wer zu viel auf einmal schickt, wird geblockt – und riskiert Datenverlust.
- Fehlende Validierung: Eingabedaten sollten immer geprüft werden – z. B. auf E-Mail-Format, Pflichtfelder, Dubletten. Sonst landen Müll-Datensätze im CRM.
- Unsaubere Modulkopplung: Wer Daten aus vorherigen Modulen ohne Prüfung weiterverwendet, riskiert Nullwerte, leere Felder oder kaputte Payloads.
- Keine Doku: Szenarien ohne Beschreibung, Tags oder Naming-Convention sind nach zwei Monaten nicht mehr wartbar – auch nicht vom Ersteller selbst.

Die Lösung? Entwickle wie ein Entwickler. Schreibe saubere, kommentierte Szenarien. Benutze Error-Handling. Baue Logging ein. Teste jede Änderung. Und – ganz wichtig – versioniere deine Workflows. Denn ein “kleiner Fix” kann im Livebetrieb schnell zur Katastrophe werden.

Make skalieren: Von der Einzellösung zum strategischen Backbone

Viele starten mit Make als kleine Lösung für ein nerviges Problem. Doch wer das Tool ernst nimmt, kann es zum integralen Bestandteil seiner MarTech-Infrastruktur ausbauen. Das bedeutet: Make wird nicht nur Middleware, sondern Steuerzentrale für Datenflüsse, Trigger, Alerts und Prozessoptimierung. Aber Skalierung braucht Struktur. Und Governance.

Ein skalierbares Make-Setup sollte:

- Namenskonventionen für Szenarien, Variablen und Module definieren
- Ordnerstruktur nach Funktion oder Abteilung einführen
- Versionierung und Änderungsprotokolle pflegen
- Rollen und Berechtigungen klar trennen (besonders bei mehreren Usern)
- Monitoring und Logging-Mechanismen einbauen (z. B. mit internen Slack-Benachrichtigungen)

Außerdem unerlässlich: regelmäßige Reviews. Läuft der Workflow noch korrekt? Hat sich die API geändert? Gibt es neue Anforderungen? Nur wer seine Automationen wie Software betrachtet – mit Tests, Releases und Maintenance – wird Make dauerhaft sinnvoll nutzen können.

Fazit: Make ist kein Befehl. Es ist eine Verantwortung.

Make ist kein Zauberstab. Es ist kein Tool, das deine Marketingprobleme löst, weil du es benutzt. Es ist ein Framework, das dir erlaubt, Prozesse zu automatisieren – wenn du sie verstehst. Wer Make als Shortcut benutzt, landet schnell in der Hölle inkonsistenter Daten, fehlerhafter Workflows und verlorener Leads. Wer es als strategisches Tool begreift, kann sein Online-Marketing auf ein neues Level heben – effizienter, schneller, skalierbarer.

Aber: Make ist nichts für faule Leute. Nichts für Leute, die “einfach mal was ausprobieren” wollen. Es ist für Menschen, die bereit sind, Technik ernst zu nehmen. Die verstehen, dass Automatisierung nur so gut ist wie die Logik, die sie antreibt. “Make” ist kein Befehl. Es ist eine Verpflichtung zu Exzellenz. Und genau deshalb gehört es in den Werkzeugkasten jedes ernsthaften Marketers von 2025.