QlikView: Datenanalyse neu gedacht und visualisiert

Category: Online-Marketing
geschrieben von Tobias Hager | 6. Februar 2026



QlikView: Datenanalyse neu gedacht und visualisiert

Excel ist tot, BI-Tools sind die neuen Könige – und QlikView sitzt auf dem Thron der Datenvisualisierung. Während andere noch Pivot-Tabellen basteln, liefern dir QlikView-Modelle in Sekunden die Antworten, die dein Management sonst in Wochen zusammenspekuliert. In diesem Artikel zerlegen wir QlikView technisch, strategisch und schonungslos – und zeigen dir, wie du aus rohen

Daten echtes Business-Gold machst.

- Was QlikView ist – und warum es mehr als nur ein BI-Tool ist
- Wie die Assoziative Datenbank-Engine funktioniert – und warum das wichtig ist
- QlikView vs. Excel, Tableau & Power BI – der ehrliche Vergleich
- Technische Architektur: In-Memory, Scripting, QVDs und Data Models
- Wie du Datenmodelle aufbaust, die tatsächlich funktionieren
- Visualisierung mit QlikView: Dashboards, KPIs und Storytelling
- Deployment, Sicherheit und Skalierung in Enterprise-Umgebungen
- Best Practices für Performance, Datenkonsistenz und Governance
- Warum QlikView kein Einsteiger-Tool ist – aber ein strategischer Gamechanger
- Fazit: QlikView als BI-Werkzeug für alle, die Daten wirklich verstehen wollen

QlikView erklärt: Was ist QlikView und warum ist es nicht “nur ein BI-Tool”?

QlikView ist ein Business Intelligence Tool – aber halt eines von der Sorte, die nicht nur hübsche Diagramme malt, sondern tatsächlich einen neuen Denkansatz in die Datenanalyse bringt. Während klassische BI-Werkzeuge auf starre OLAP-Cubes und SQL-Abfragen setzen, arbeitet QlikView mit einer sogenannten assoziativen Datenmodellierung. Das bedeutet: Es denkt nicht in Tabellen, sondern in Beziehungen. Daten sind nicht mehr linear, sondern interaktiv miteinander verbunden.

Das klingt theoretisch? Ist es auch – aber nur bis du es einmal gesehen hast. Mit QlikView kannst du Daten aus verschiedenen Quellen – ERP, CRM, Excel, SQL, APIs – in einem einzigen Modell zusammenführen und dann über Dimensionen navigieren, ohne dich mit JOINs oder Tabellenverknüpfungen zu quälen. Die Engine erkennt Zusammenhänge automatisch. Das ist kein Zauber, sondern verdammt clevere Architektur.

Was QlikView ebenfalls von klassischen Tools unterscheidet: Es ist kein Frontend-Klicki-Bunti für Controller, sondern ein extrem mächtiges Werkzeug für Datenarchitekten, Analysten und IT-Profis. Mit Scripting, Variablen, QVDs (QlikView Data Files) und komplexen Transformationen baust du dir genau das Datenuniversum, das du brauchst – ohne dich an die Limitierungen von SQL oder Excel binden zu müssen.

Und ja, QlikView ist alt – aber das ist kein Bug, sondern ein Feature. Denn es ist ausgereift, performant und in der Unternehmenswelt tief verwurzelt. Wer mit QlikView arbeitet, braucht keine Spielereien – sondern Ergebnisse. Und die liefert es zuverlässig, skalierbar und blitzschnell.

Die assoziative Engine von QlikView: Warum SQL hier keine Rolle mehr spielt

Im Herzen von QlikView schlägt die sogenannte Associative Data Engine. Sie ist der Grund, warum QlikView nicht wie klassische BI-Systeme funktioniert – und warum es so verdammt schnell ist. Statt Daten in relationalen Tabellen mit harten JOINs zu verknüpfen, lädt QlikView alle relevanten Daten in den RAM – also In-Memory – und erstellt daraus eine assoziative Matrix.

Das bedeutet: Jede Auswahl, die du in einer QlikView-Anwendung triffst, beeinflusst sofort alle anderen Objekte. Du klickst auf ein Produkt – und alle Diagramme, KPIs und Tabellen filtern sich automatisch und in Echtzeit. Ohne Ladezeiten, ohne Abfrage an den Server. Das ist nicht nur UX-Gold, sondern auch analytisch ein enormer Vorteil. Denn du kannst explorativ arbeiten, Hypothesen testen und dich durch die Daten "klicken", ohne vorher alles definieren zu müssen.

Technisch basiert das auf einem sogenannten "field-value pairing system", bei dem jede Dimension und jeder Wert in einer Art Hashmap gespeichert wird. Die Engine indexiert alle Felder und hält sie im Speicher – das ist Speicherintensiv, aber rasend schnell. Deshalb ist RAM-Größe bei QlikView-Servern übrigens wichtiger als CPU-Power.

Und noch ein Killerfeature: Assoziativ bedeutet auch, dass du "graue Werte" siehst – also Daten, die nicht zu deiner aktuellen Auswahl passen. Damit erkennst du sofort Ausreißer, Lücken oder Anomalien. Klassische SQL-Systeme würden dir diese Informationen einfach ausblenden. QlikView zeigt dir, was du nicht siehst – und das ist oft genau das, was du wissen musst.

QlikView vs. Tableau vs. Power BI: Der ehrliche Vergleich

Tableau ist hübsch, Power BI ist günstig – aber QlikView ist brutal effizient. Der große Unterschied liegt in der Architektur. Während Tableau und Power BI stark auf visuelle Drag-and-Drop-Interfaces und SQL-basierte Datenquellen setzen, geht QlikView einen anderen Weg: Es zwingt dich zur Modellierung. Und das ist auch gut so – denn ein sauberes Datenmodell ist die Voraussetzung für jede sinnvolle Analyse.

In Tableau kannst du schnell etwas zusammenklicken, aber komplexe Logiken oder konsistente KPIs über mehrere Datenquellen hinweg? Viel Spaß. In Power BI bist du auf DAX angewiesen – eine Sprache, die man entweder liebt oder hasst. QlikView dagegen nutzt eine eigene Skriptsprache im Load Script – damit baust du dir genau die Datenpipelines, die du brauchst. Ohne

Kompromisse.

Visualisierung? Klar, da hat Tableau die Nase vorn, was Design-Sexiness betrifft. Power BI punktet mit Integration in die Microsoft-Welt. Aber wenn es um Speed, Skalierbarkeit und komplexe Datenlogiken geht, schlägt QlikView beide. Vor allem im Enterprise-Bereich, wo Datenkonsistenz über Abteilungen hinweg entscheidend ist.

Und was ist mit Qlik Sense? Das ist der moderne Nachfolger von QlikView – browserbasiert, responsive, hübscher. Aber: QlikView ist präziser, kontrollierter und bietet Features, die Qlik Sense (noch) nicht kann. Wer maximale Kontrolle über Daten und Logik will, bleibt bei QlikView – oder nutzt beide Tools parallel, was dank Qlik's Shared Data Architecture problemlos möglich ist.

Technische Architektur: Wie QlikView mit QVDs, In-Memory und Scripting skaliert

QlikView ist kein Tool, das du mal eben installierst und loslegst. Es ist ein System. Eine Plattform. Und wie bei jedem komplexen System ist eine saubere Architektur entscheidend. Die Basis ist das In-Memory-Modell: Alle Daten werden beim Laden in den RAM geschrieben – kein Nachladen, keine Live-Abfragen. Das macht QlikView so schnell, aber auch RAM-hungrig.

Das zentrale Element jeder QlikView-Architektur sind QVDs – proprietäre, spaltenorientierte Datenformate, die komprimiert und extrem performant sind. Sie dienen als Zwischenspeicher, Data Mart oder Cache – und ermöglichen es, große Datenmengen effizient zu verarbeiten, ohne jedes Mal die Quelle anzuzapfen.

Das Laden und Transformieren der Daten erfolgt über das Load Script. Hier definierst du, welche Daten wie geladen, transformiert, aggregiert und verknüpft werden. Du kannst hier SQL nutzen, aber auch QlikView-eigene Funktionen wie ApplyMap, Peek, Previous oder Resident Loads. Wer das Load Script meistert, kontrolliert die gesamte Datenpipeline.

Ein typisches QlikView-Projekt besteht aus mehreren Layern:

- Raw Data Layer: Rohdaten aus verschiedenen Quellen werden extrahiert und als QVD gespeichert.
- Transform Layer: Daten werden gereinigt, kombiniert und modelliert – ebenfalls als QVDs gespeichert.
- Presentation Layer: Die eigentliche QlikView-App lädt die vorbereiteten QVDs und stellt sie visualisiert dar.

Diese Trennung sorgt für Skalierbarkeit, Wiederverwendbarkeit und Performance. Und sie ist der Grund, warum QlikView in großen Konzernen mit Milliarden-Datensätzen problemlos läuft – wenn man es richtig macht.

Visualisierung und User Experience: Dashboards, KPIs und Performance-Design

QlikView ist kein Design-Wunder. Aber es ist funktional. Und genau das macht es stark. Visualisierungen sind nicht “pretty”, sondern präzise. Balken, Linien, Kuchendiagramme? Klar. Aber der wahre Wert liegt in der Interaktivität. Jeder Klick ist ein Filter. Jede Auswahl verändert sofort das gesamte Dashboard. Und genau das erzeugt Erkenntnisse in Echtzeit.

Ein gutes QlikView-Dashboard ist nicht überladen, sondern fokussiert. Es zeigt KPIs, Trends und Details – aber nur das, was gebraucht wird. Performance ist dabei entscheidend. Denn jedes Objekt, jede Formel, jede Aggregation kostet Rechenzeit. Deshalb gilt: So wenig Objekte wie möglich, so viel Nutzen wie nötig.

Technisch kannst du mit Set Analysis, If-Statements, Variablen und Aggregationslogiken extrem komplexe Visualisierungen bauen. Aber: Komplexität ist nicht gleich Qualität. Ein langsames Dashboard, das alles kann, nutzt niemand. Schnelle Reaktionszeit ist UX-Killerfeature Nummer 1.

Auch wichtig: QlikView bietet keine automatische Responsive-Funktion. Dashboards müssen für verschiedene Auflösungen manuell optimiert werden. Wer das ignoriert, liefert schlechte Usability – besonders auf Laptops oder kleineren Bildschirmen.

Und zuletzt: Storytelling ist kein Buzzword. Gute QlikView-Anwendungen führen den Nutzer durch die Daten, helfen bei der Interpretation und machen Ergebnisse nachvollziehbar. Das ist keine Designfrage – das ist Analysepsychologie.

Fazit: QlikView als Werkzeug für echte Datenstrategie

QlikView ist nicht das hippeste Tool auf dem Markt. Es hat keine fancy Animationen, keine Selfie-Funktion und keine Einhorn-Icons. Aber es liefert. Präzise, schnell, skalierbar. Für alle, die Daten nicht nur anschauen, sondern verstehen und nutzen wollen, ist QlikView ein unverzichtbares Werkzeug.

Wer BI ernst nimmt, kommt an QlikView nicht vorbei. Es erfordert technisches Denken, saubere Modellierung und eine klare Architektur. Aber es belohnt dich mit Einsichten, die du mit Tableau, Excel oder Power BI nie findest. QlikView ist kein Tool für Anfänger – aber ein Gamechanger für alle, die Datenanalyse nicht als Visualisierung, sondern als strategische Disziplin begreifen.