



# Projectile

Version 6

Die integrierte Gesamtlösung für  
Dienstleister: Projekte, Prozesse, Wissen.

## Kurzdokumentation, Reportgenerator+

### Inhaltsverzeichnis

1 Reportgenerator und Reportgenerator+.....	2
2 Dashboard-Builder.....	2
3 Kombinierte Abfragen.....	7
3.1 Komplexe Abfragen in einfache Anfragen aufteilen.....	7
3.2 Mehrere Listen zu einer Liste zusammenfügen.....	8

# 1 Reportgenerator und Reportgenerator+

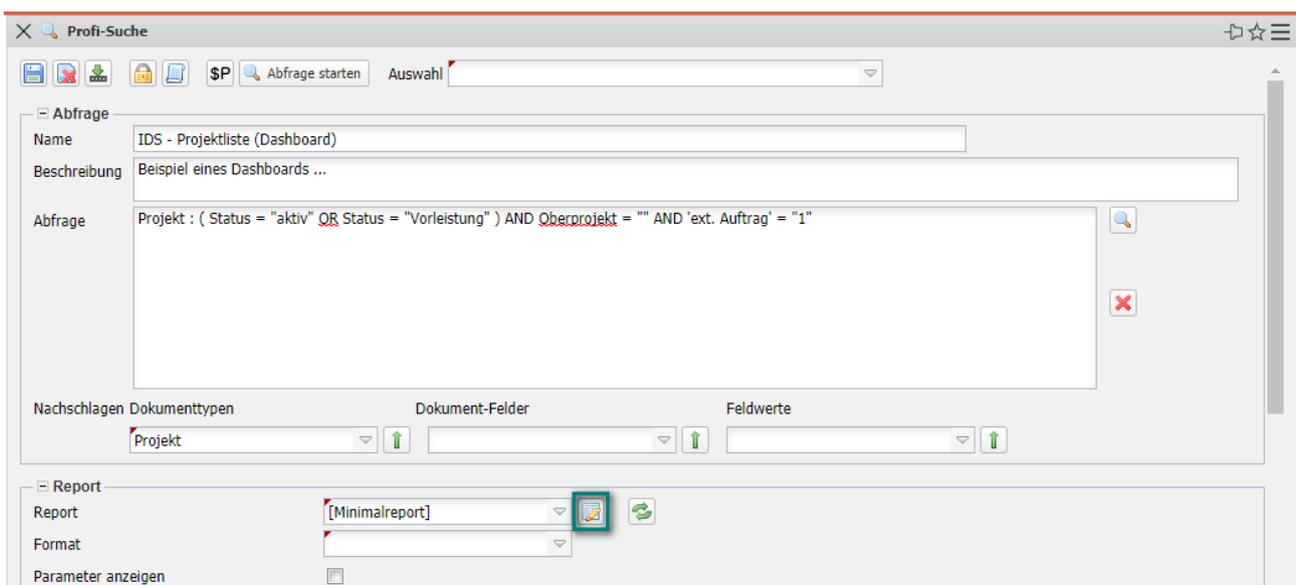
Ein **Reportgenerator** ist eine Anwendung zur Erstellung von Berichten aus den Daten einer Datenbank. Damit Reports nicht nur von Softwareentwicklern, sondern auch von Endanwendern erstellt werden können, werden die dazu erforderlichen Datenbankabfragen über eine grafische Benutzeroberfläche definiert und die datenbankspezifischen Anweisungen (hier SQL) automatisch generiert. In Projectile besteht der Reportgenerator aus dem **Abfrage-Manager** und dem **Report-Designer**. Der Reportgenerator in Projectile bietet, über die Standardreports hinaus, eine weitere Möglichkeit den Datenbestand von Projectile auszuwerten und bestehende Reports anzupassen.

Die Abfragen in Projectile können bspw. in den Portalen oder den Standardauswertungen verwendet werden.

Der **Reportgenerator+** ist eine kostenpflichtige Erweiterung des integrierten Reportgenerators und besteht aus den **Dashboard-Builder** und den **kombinierten Abfragen**.

## 2 Dashboard-Builder

Mit Hilfe des Dashboard-Builders können Sie Dashboards mit Haupt- und Detailauswertungen erzeugen. Das folgende Beispiel zeigt die Anwendung des Dashboard-Builders. Zunächst wird die Abfrage über die aktiven Kundenprojekte geladen und ...



... dann ein neuer Report erzeugt.

In der Reportdefinition sind jetzt nur noch wenige zusätzliche Schritte erforderlich, um aus einem tabellarischen Report ein Dashboard zu erstellen. In der Spalte Diagramm müssen die Achsen bestimmt werden. Im Beispiel wird die Projektnummer zur x-Achse und die vier Kennzahlen „Plan[h]“, „Ist[h]“, „Differenz[h]“ und Rest[h]“ zur y-Achse.

Daten											Diagramm	
		Feldname	Formula	Aggregatfunktion	Grup	Seite	Sicht	Sortierung	Aufs	Diagramm	Beschriftung	
1	...	Projekt-Nr,		0 - keine	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	2	<input checked="" type="checkbox"/>	Index (x)		
2	...	Bezeichnung		0 - keine	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>		Kunde	
3	...	Auftraggeber		0 - keine	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	1	<input checked="" type="checkbox"/>			
4	...	Status		0 - keine	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>			
5	...	Unit		0 - keine	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>			
6	...	Plan-Beginn		0 - keine	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>			
7	...	Plan-Ende		0 - keine	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>			
8	...	Planaufwand *		1 - Summe	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	Daten (y)	Plan [h]	
9	...	Istaufwand *		1 - Summe	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	Daten (y)	Ist [h]	
10	...	(Formel)	Planaufwand - I...	1 - Summe	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	Daten (y)	Differenz [h]	
11	...	Erwarteter_Res...		1 - Summe	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	Daten (y)	Rest [h]	

Die Bezeichnung ist weniger als X-Achse geeignet, weil dieses Feld zu lang ist. Im Reiter „Diagramm“ wird dann die „Hauptauswertung“ (hier Balkenstapel) und die „Detailauswertung“ (hier Torte) ausgewählt.

Statisches Diagramm

Dynamisches Diagramm

Hauptauswertung: Balkenstapel

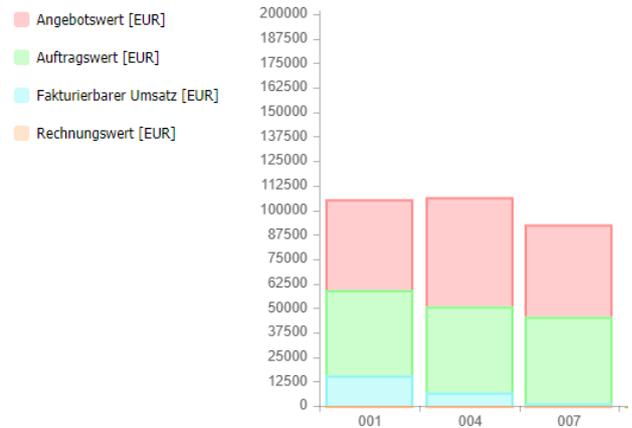
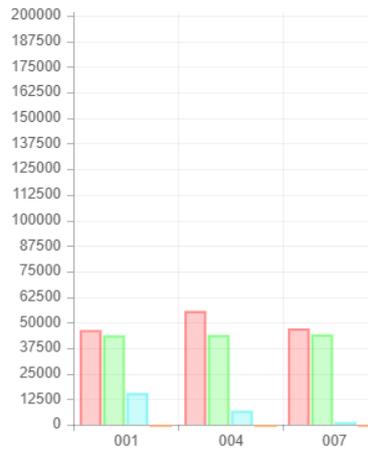
Detailauswertung: Torten

Animation:

Derzeit sind u. a. folgende Diagrammtypen verfügbar und als Hauptauswertung geeignet:

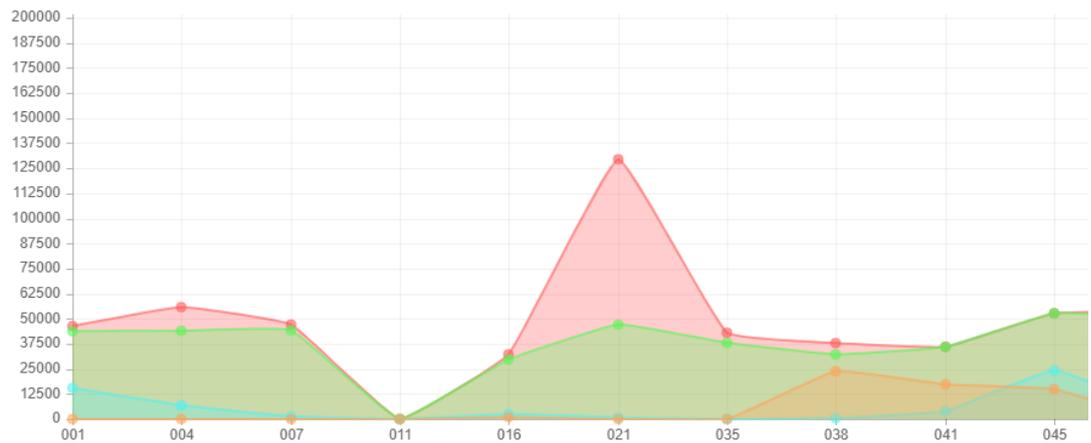
## Balken und Balkenstapel:

- Angebotswert [EUR]
- Auftragswert [EUR]
- Fakturierbarer Umsatz [EUR]
- Rechnungswert [EUR]



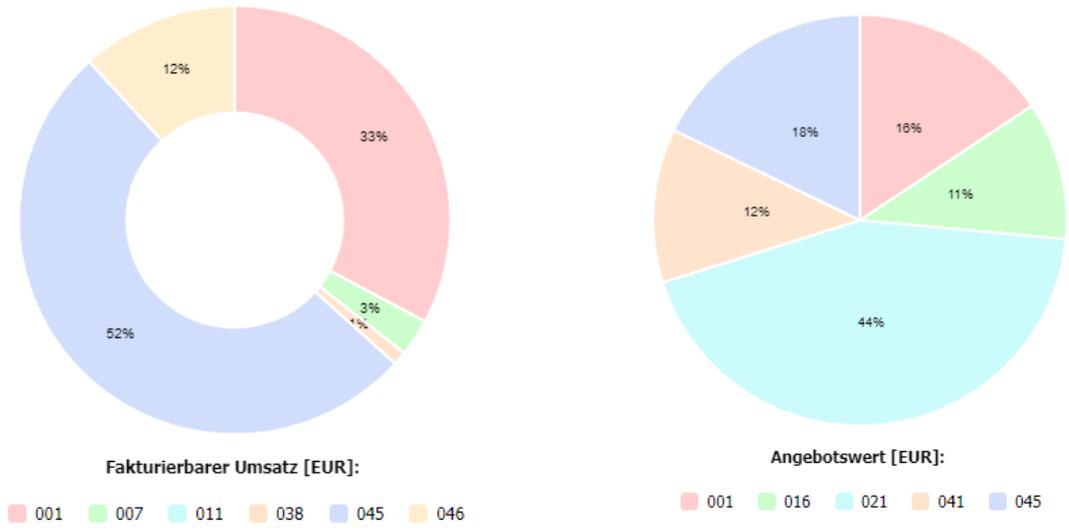
## Liniendiagramm:

- Angebotswert [EUR]
- Auftragswert [EUR]
- Fakturierbarer Umsatz [EUR]
- Rechnungswert [EUR]

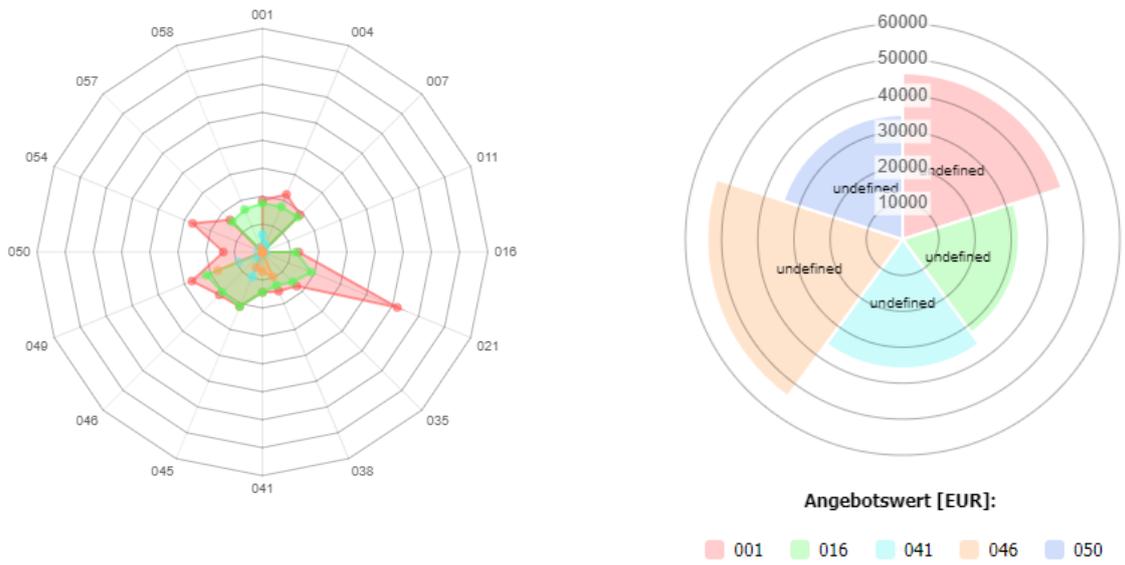


Derzeit sind u. a. folgende Diagrammtypen als Detailauswertung geeignet:

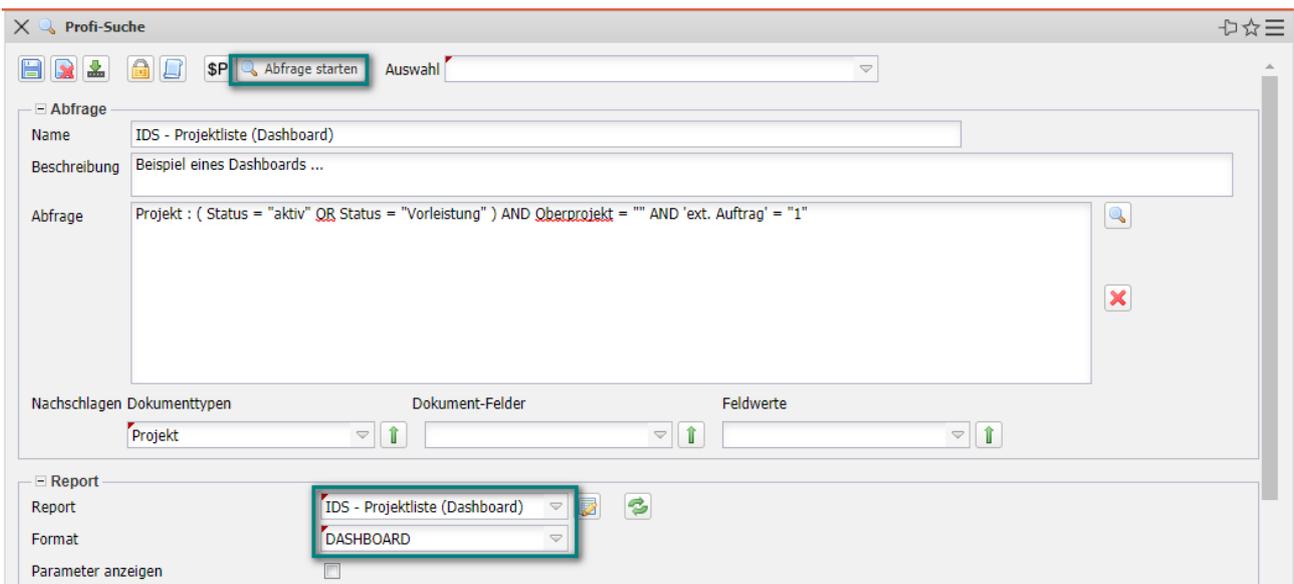
Donut und Torten:



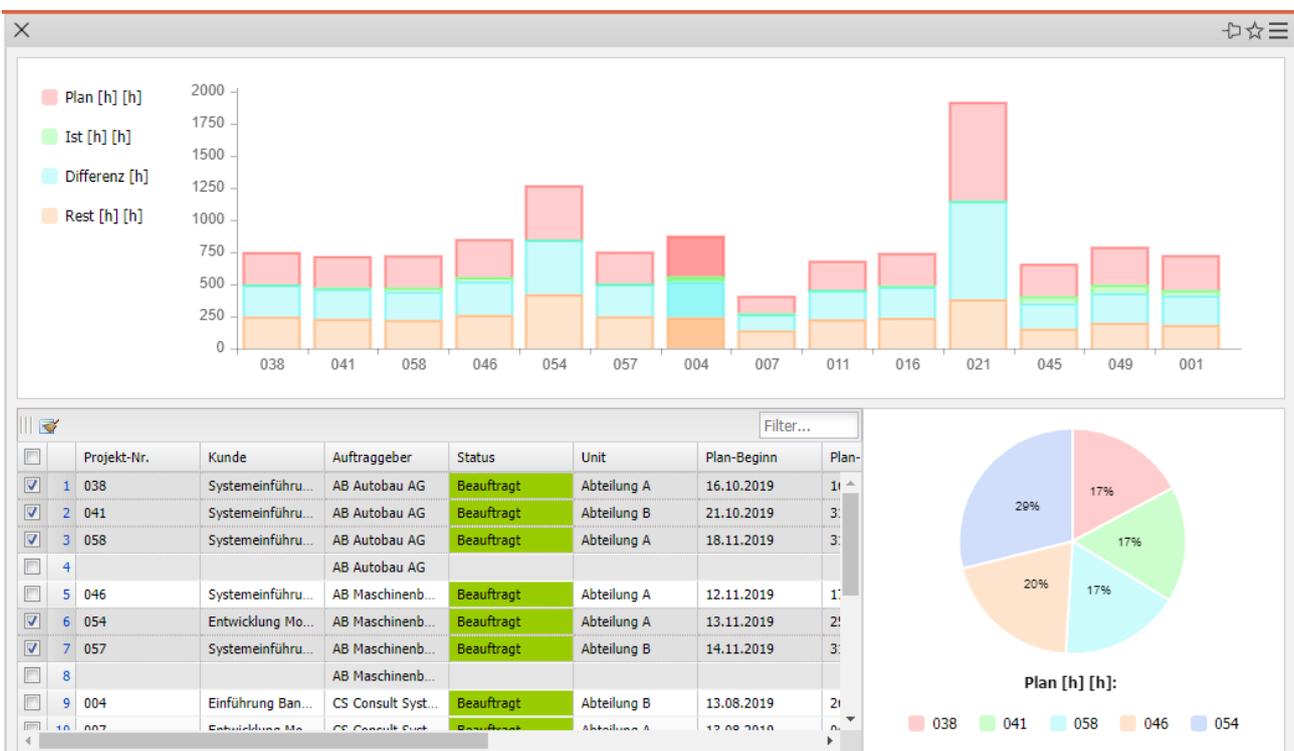
Netz und Polar:



Nach dem Speichern und Schließen des Reports ...



... wird mit „Abfrage starten“ das Dashboard mit den 4 Kennzahlen (Plan, Ist, Differenz, Rest) als Balkenstapel generiert:



## 3 Kombinierte Abfragen

Diese Modul bietet zwei Features zur Kombination von Abfragen. Das erste Feature ermöglicht es, komplexe Abfragen in einfache Abfragen zu zerlegen, die einfach und performant ablaufen. Das zweite Feature erlaubt es, mehrere Listen mit der gleichen Struktur aneinander zu hängen.

Beide Features erlauben es, nicht nur zwei sondern mehrere Abfragen zu einem Endergebnis zu kombinieren.

### 3.1 Komplexe Abfragen in einfache Anfragen aufteilen

Manchmal kann es nötig sein, Abfragen über mehrere Dokumenttypen zu benutzen. Zum Beispiel kann man Projekte suchen, deren Team-Mitglieder bestimmten Abteilungen angehören und deren Auftraggeber bestimmte Bedingungen erfüllen.

Versucht man eine solche Abfrage mit normalen Mitteln umzusetzen, entstehen sehr komplexe Abfragen, die schwer verständlich sind und deren Performance schlecht sein kann.

Zusammengesetzte Abfragen lösen das Problem so, dass man die einzelnen Dokumenttypen jeweils mit separaten, einfachen Abfragen sucht und die Abfragen dann zu einem Endergebnis kombiniert.

Hierzu sind folgende Schritte nötig:

1. Abfrage mit einem Parameter erstellen
2. Abfrage erstellen, die Dokumente für den Parameter liefert
3. Abfrage 1 soll Parameter aus Abfrage 2 laden

#### **Abfrage mit einem Parameter erstellen**

Der Abfragemanager unterstützt Parameter in dieser Form:

**Projekt: Projektteam CONTAINS \$Mitarbeiter**

Führt man diese Abfrage aus, wird Projectile den Benutzer zur Eingabe eines Mitarbeiters auffordern:

✕ \$P Abfrageparameter

**Bitte geben Sie die fehlenden Parameter ein**

Wildcard-Möglichkeiten: \*, \*abc, abc\* und \*abc\* (alle, alle auf abc endende, alle mit abc beginnende, alle abc enthaltende)

Mitarbeiter

Anzahl Suchergebnisse begrenzen auf

## Abfrage erstellen, die Dokumente für den Parameter liefert

Statt die Eingabe des Parameters vom Nutzer zu fordern, kann man auch Mitarbeiter aus einer anderen Abfrage liefern:

**Mitarbeiter : 'Aktueller Standort' = "Baden-Württemberg"**

## Abfrage 1 soll Parameter aus Abfrage 2 laden

Jetzt kann man in Abfrage 1 für den Parameter \$Mitarbeiter aus Abfrage 2 laden:

☐ Kombination

	Parameter		andere Abfrage
Parameter aus anderer Abfrage laden	Mitarbeiter		Mitarbeiter

Wenn man die Abfrage jetzt ausführt, wird im Hintergrund zunächst die Abfrage „Mitarbeiter“ ausgeführt und danach für jeden Mitarbeiter, der als Ergebnis geliefert wird, die zweite Auswertung einmal aufgerufen. Die Ergebnisse werden in einer gemeinsamen Liste angezeigt.

## 3.2 Mehrere Listen zu einer Liste zusammenfügen

An eine Liste, die von einer Abfrage mit einem Report erzeugt wird, kann man eine weitere Liste unten anhängen.

Hierzu muss man eine weitere Abfrage definieren, die Dokumente des gleichen Typs liefert. Für die Ausgabe wird auch der Report der ersten Abfrage benutzt.

Ein Beispiel:

An eine Liste von Projekten aus Deutschland sollen die Projekte aus Großbritannien angehängt werden:

❌ 🔍 **Profi-Suche**

[-] **Abfrage**

Name

Beschreibung

Abfrage

❌ 🔍 **Profi-Suche**

[-] **Abfrage**

Name

Beschreibung

Abfrage

Nachschlagen Dokumenttypen  Dokument-Felder  Feldwe

[+] **Report**

[-] **Kombination**

Parameter  andere Abfrage

Parameter aus anderer Abfrage laden

Ergebnis anderer Abfragen hinzufügen

<input type="checkbox"/>	<input type="text" value="Bezeichnung"/>
<input type="checkbox"/>	1 Projekte UK