Berechnete Attribute

- Definition
- Konfiguration
  - Beispiel 1
  - Beispiel 2
- Anwendung

Definition

Anstelle vom Auslesen und Schreiben fester Attributwerte in eine Datenbanktabelle kann ein Attribut seine Inhalte auch aus einer Funktion auf Datenbankebene beziehen. Werte können so datensatzunabhängig zur Laufzeit berechnet und angezeigt werden, ohne dass ein Datensatz dafür gespeichert oder der Vorgang vom Benutzer angetriggert werden muss. Anwendungsfälle sind z.B. Berechnung des Jahresumsatzes eines Artikels, Anzeige von berechneten Datumswerten, Druckerstatus usw.

Berechnete Attribute verhalten sich für den Benutzer fast genauso wie "normale" Attribute. Sie können in Layouts und Suchergebnislisten angezeigt werden. Da es sich bei diesen Attributen aber um eine dynamische Berechnung handelt, wird der Wert nicht in der Datenbank gespeichert sondern immer zur Laufzeit neu berechnet und angezeigt. So ist es z.B. nicht möglich, den Wert über eine Regel zu setzen !

Die Berechnung von Attributen des Typs Referenzfeld ist nicht möglich.

> Berechnete Attribute sollten sparsam verwendet werden, da ihre Berechnung unter Umständen sehr zeitintensiv werden kann, vor allem bei Suchen und der Anzeige in der Suchergebnisliste.

Per Derfault ist die Sortierfunktion bei berechneten Attributen in der Suchergebnisliste deaktiviert. Um diese zu aktivieren gibt es einen Parameter SORT_CALCULATED_ATTRIBUTES.

Konfiguration

Es gibt zwei Wege für die Erstellung eines berechneten Attributes.

Über ein Datenbankobjekt

In diesem Fall wird eine Datenbankfunktion im Bereich Datenbankobjekte angelegt. Die Datenbankfunktion muss einen Übergabeparameter enthalten. Dieser stellt die ID des Datensatzes dar, in dem das berechnete Attribut eingeblendet ist. Diese ID muss in der Abfrage entsprechend verarbeitet werden.

Der Rückgabewert der Datenbankfunktion muss den Ergebniswert des berechneten Attributes zurückgeben. Der Datentyp muss mit dem Attribut im BO-Wizard übereinstimmen.

Der Name dieser Funktion muss mit **CA**_ (für Calculated Attribute) beginnen, der Rest des Namens sollte idealer Weise dem Namen des Attributs entsprechen, für das die Funktion verwendet wird.

Über eine Datenquelle

In diesem Fall wird das berechnete Attribut über eine Datenquelle für ein Berechnetes Attribut definiert. Dort findet man auch ein Beispiel.

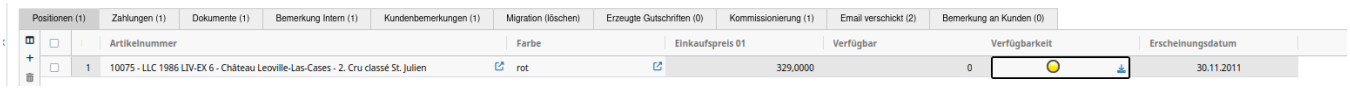
Beispiel 1

Berechnet die Anzahl verfügbarer Artikel. Die Berechnungsgrundlage bilden zwei andere Attribute, deren Werte über eine SELECT Anweisung herangezogen werden. Der Übergabeparameter **id** stellt in die INTID von dem aktuellen Objekt dar. Dieses wird automatisch an die Funktion mit übergeben. Der Rückgabewert muss vom Typ **varchar2** sein, da bei Attributen grundsätzlich das Value als String in der Datenbank abgelegt wird.

```
CREATE OR REPLACE FUNCTION CA_ARTIKELVERFUEGBAR(id IN number) RETURN varchar2 IS
  summe number(12);
  bestand number(12);
  reserviert number(12);
begin
  SELECT strvalue INTO bestand
  FROM t_ud_go_attribute
  WHERE intid_t_md_attribute = (SELECT intid FROM t_md_attribute WHERE strattribute='artikelBestand')
        AND intid_t_ud_genericobject = id;

  SELECT strvalue INTO reserviert
  FROM t_ud_go_attribute
  WHERE intid_t_md_attribute = (SELECT intid FROM t_md_attribute WHERE strattribute='artikelReserviert')
        AND intid_t_ud_genericobject = id;

  summe:=bestand-reserviert;

  RETURN(summe);
end CA_ARTIKELVERFUEGBAR;
```

Beispiel 2

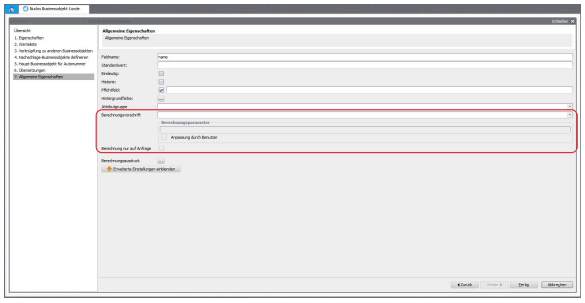
Ein berechnetes Attribut kann auch ein Bild zurückgeben, damit kann man wunderbar Ampeln realisieren:



Die zurückzugebenden Bilder muss man vorher als Ressourcen (unter Konfiguration | Sonstiges | Ressourcen) hochladen.

Beispiel (Postgres) für die Datenbankfunktion des berechneten Attributes, die die Ampel zurückgibt:

```
CREATE OR REPLACE FUNCTION C31G_CA_PROJEKT_VERFUEGBARKEIT(id numeric)
RETURNS bytea AS
$BODY$
DECLARE
   ist numeric(9,2);
   soll numeric(9,2);
   bild bytea;
BEGIN
   select dblfortschritt, dblusedbudget / dblworkingbudget * 100 into ist, soll from c31g_projekt where
intid=id;
   if (ist = 100) then
      bild := null;
   else
      if (ist >= soll) then
         select blbcontent into bild from t_md_resource where strname = 'Ampel grün';
      else
         select blbcontent into bild from t_md_resource where strname = 'Ampel rot';
      end if;
   end if;
   RETURN bild;
END;
$BODY$
  LANGUAGE 'plpgsql' VOLATILE
  COST 100;
```

Anwendung

Im Businessobjekt wird das berechnete Attribut im Schritt 7 der Attributdefinitionen verlinkt.