



Oracle Hidden Secrets:

# Bildbearbeitung in der Datenbank

Carsten Czarski, ORACLE Deutschland B.V. & Co. KG

Ein Bild in der Datenbank als Binary Large Object (BLOB) abzuspeichern, ist nichts Besonderes. Aber wem ist bekannt, dass die Datenbank sogar einfache Funktionen zur Bildbearbeitung enthält? Egal ob Drehen, Skalieren, Umrechnen in Graustufen oder das Anwenden eines Wasserzeichens: Das alles ist im Grundumfang der Datenbank enthalten und kann mit wenigen PL/SQL-Codezeilen sofort genutzt werden. Der Artikel zeigt, wie das genau geht.

Bereits mit der Datenbank-Version 8i wurde Oracle-Multimedia eingeführt – damals noch unter dem Namen „InterMedia“. Multimedia besteht aus Datentypen für verschiedene Multimedia-Inhalte und aus Funktionen zum Umgang damit. Zwar wurden seinerzeit mehrere Datentypen für verschiedene Mediatypen eingeführt, praktisch bedeutsam ist jedoch nur der „ORDIMAGE“-Datentyp mit Funktionen zum Umgang mit Bildern. Allerdings ist es nicht nötig, die Datentypen von Tabellenspalten oder PL/SQL-Variablen von „BLOB“ auf „ORDIMAGE“ umzustellen – denn alle ORDIMAGE-Funktionen lassen sich auch mit einem BLOB nutzen.

## Ein Bild skalieren

Die Prozedur „ORDIMAGE.PROCESSCOPY“ dient dem Skalieren eines Bildes. Sie nimmt zwei BLOBs als Parameter entgegen: Der eine enthält das zu skalierende Bild, der zweite das Resultat. Der dritte Parameter „COMMAND“ bestimmt die konkret durchzuführende Operation, denn „PROCESSCOPY“ beherrscht mehrere Bildoperationen:

- „maxScale“ skaliert ein Bild, sodass das Seitenverhältnis erhalten bleibt. Diese Methode ist sehr gut geeignet, um kleinere Vorschaubilder zu generieren.
- „rotate“ dreht Bilder, um sie in die richtige Ausrichtung zu bringen. Hat man

(von einem Smartphone) die Angaben über die linke untere Ecke, so kann dies völlig automatisch geschehen.

- „flip“ und „mirror“ spiegeln Bilder horizontal oder vertikal.

*Listing 1* zeigt, wie „ORDIMAGE.PROCESSCOPY“ mit dem Kommando „maxScale“ funktioniert, um ein Vorschaubild zu erzeugen, das maximal 200x200 Pixel groß ist. Dieser Code ließe sich gut in einen Trigger implementieren, der beim Abspeichern eines neuen Bildes sofort aktiv wird.

Das Rotieren eines Bildes wird analog durchgeführt – allerdings wird dann beispielsweise „rotate 90“ als „COMMAND“

```

declare
  l_src_bild blob;
  l_dst_bild blob;
begin
  -- Bild aus Tabelle holen
  select image_blob into l_src_bild
  from image_table where id = 1;

  -- BLOB für skaliertes Bild initialisieren
  dbms_lob.createtemporary(l_dst_bild, true, DBMS_LOB.SESSION);

  -- Bild skalieren
  ordimage.processcopy(
    imageblob => l_src_bild,
    command   => 'maxScale 200 200',
    dest      => l_dst_bild
  );

  -- Ergebnis abspeichern
  insert into scaled_image_table
  set image_blob = l_dst_bild
  where id = 1;
end;

```

Listing 1: Ein Bild mit PL/SQL-Code in der Datenbank skalieren

```

-- Parameter für das "Wasserzeichen"
props := ordsys.ord_str_list(
  'font_name=Times New Roman',
  'font_style=bold',
  'font_size=40',
  'text_color=red',
  'position=bottomright',
  'transparency=0.6'
);

-- Wasserzeichen hinzufügen
ORDImage.applyWatermark(
  imageblob      => src_blob,
  added_text     => 'HIDDEN SECRET',
  dest           => dst_blob,
  logging        => logging,
  watermark_properties => props
);

```

Listing 2: PL/SQL-Code für Image Watermarking

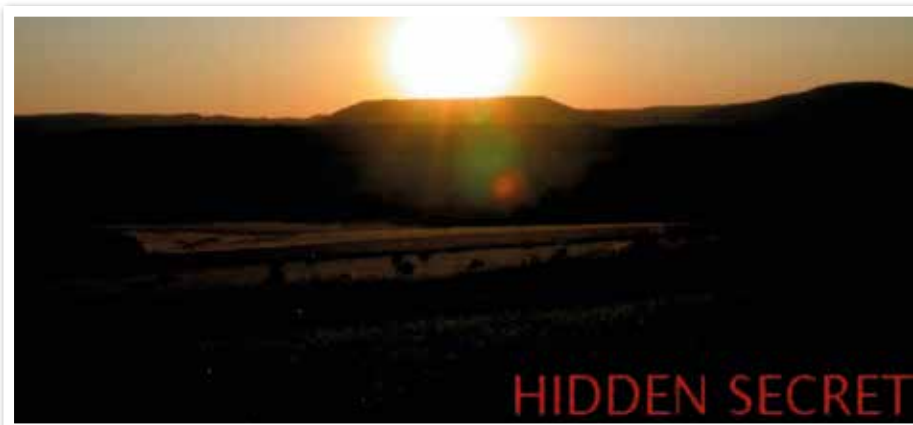


Abbildung 1: Image Watermarking in Aktion

an „PROCESSCOPY“ übergeben. Daneben gibt es in „ORDIMAGE“ auch „PROCESS“ – diese Funktion speichert das Resultat allerdings nicht in einen neuen BLOB, sondern überschreibt die Quelldaten.

## Wasserzeichen anwenden

Einige Funktionen werden nicht über „PROCESS“ beziehungsweise „PROCESSCOPY“ angesprochen. „ORDIMAGE.APPLYWATERMARK“ schreibt ein Wasserzeichen in ein Bild – das kann entweder ein anderes Bild oder einfacher Text sein. *Listing 2* zeigt ein Code-Schnipsel, das den Text „HIDDEN SECRET“ in roter Schrift in ein Bild schreibt. Das Resultat ist in *Abbildung 1* zu sehen.

## Metadaten mit SQL und PL/SQL auslesen

ORDIMAGE.GETMETADATA dient schließlich dem Auslesen der Metadaten, die binär im Bild gespeichert sind. Besonders interessant sind hier die EXIF-Metadaten, die jede handelsübliche Kamera oder jedes Smartphone ins Bild schreibt. Waren beim Smartphone auch noch die Ortungsdienste aktiv, so wird es besonders spannend, denn nun enthalten die Metadaten zusätzlich die Position des Bildes als Längen- und Breitengrad (*siehe Abbildung 2*).

„ORDIMAGE.GETMETADATA“ liefert die Metadaten als „XMLTYPE“ zurück. Mit den XML-Funktionen der Datenbank können die GPS-Daten dann extrahiert und weiterverwendet werden (*siehe Listing 3*).

Die Funktionalität von „ORDIMAGE“ ist zwar sehr mächtig, die Benutzung der Funktionen ist jedoch alles andere als intuitiv. Auf dem Blog des Autors (*siehe „Weitere Informationen“*) findet sich daher das PL/SQL-Paket „PKG\_ORDIMAGE“, das eine einfache Schnittstelle für die verschiedenen „ORDIMAGE“-Funktionen bereitstellt – die Arbeit mit den Bildfunktionen ist damit wesentlich einfacher. *Listing 4* zeigt das Skalieren eines Bildes mit „PKG\_ORDIMAGE“ – der Code ist deutlich besser lesbar als in *Listing 1*. *Listing 5* extrahiert die EXIF-Metadaten als „XMLTYPE“, *Listing 6* die Längen- und Breitengrad direkt als SDO\_GEOMETRY-Datentyp, der anschließend mit einem räumlichen Index versehen werden kann.

```

<FlashpixVersion tag="40960">0100</FlashpixVersion>
<ColorSpace tag="40961">sRGB</ColorSpace>
<PixelXDimension tag="40962">3264</PixelXDimension>
<PixelYDimension tag="40963">2448</PixelYDimension>
<SensingMethod tag="41495">One-chip color area</SensingMethod>
<ExposureMode tag="41986">Auto exposure</ExposureMode>
<WhiteBalance tag="41987">Auto</WhiteBalance>
<FocalLengthIn35mmFilm tag="41989">35</FocalLengthIn35mmFilm>
<SceneCaptureType tag="41990">Standard</SceneCaptureType>
</ExifIfd>
<GpsIfd tag="34853">
  <GPSLatitudeRef tag="1">North latitude</GPSLatitudeRef>
  <GPSLatitude tag="2">48.215332</GPSLatitude>
  <GPSLongitudeRef tag="3">East longitude</GPSLongitudeRef>
  <GPSLongitude tag="4">16.365</GPSLongitude>
  <GPSAltitudeRef tag="5">Sea level</GPSAltitudeRef>
  <GPSAltitude tag="6">201.0</GPSAltitude>
  <GPSTimeStamp tag="7">07:42:08.230</GPSTimeStamp>
  <GPSImgDirectionRef tag="16">True direction</GPSImgDirectionRef>
  <GPSImgDirection tag="17">106.34764</GPSImgDirection>
</GpsIfd>
</exifMetadata>

```

Abbildung 2: Mit ORDIMAGE.GETMETADATA ausgelesene EXIF-Metadaten

```

alter session set nls_language='AMERICAN';

select lon, lontype, lat, lattype
from table(
  ordimage.getmetadata(xdburitype('/public/Foto.JPG').getblob(),
    'EXIF')
) mx,
xmltable(
  xmlnamespaces(DEFAULT 'http://xmlns.oracle.com/ord/meta/exif'),
  '/exifMetadata/GpsIfd'
  passing mx.object_value
  columns
    lat number path '/GpsIfd/GPSLatitude',
    lon number path '/GpsIfd/GPSLongitude',
    lattype varchar2(30) path '/GpsIfd/GPSLatitudeRef',
    lontype varchar2(30) path '/GpsIfd/GPSLongitudeRef'
) gps;

```

Listing 3: Extraktion der GPS-Koordinaten aus einem Bild

```

select
  pkg_ordimage.thumbnail(image_
    blob,200,200) thumbnail
from image_table
where id = 1;

```

Listing 4: Skalieren eines Bildes mit PKG\_ORDIMAGE

```

select
  pkg_ordimage.get_exif(image_
    blob).getclobval() exif_xml
from image_table
where id = 1;

```

Listing 5: Extraktion der EXIF-Metadaten als XMLTYPE

```

select
  pkg_ordimage.get_location(image_blob) location
from image_table
where id = 1;

```

Listing 6: Direkte Extraktion der GPS-Koordinaten mit PKG\_ORDIMAGE

## Fazit

Die Bildfunktionen der Oracle-Datenbank eignen sich hervorragend dazu, Bilder in der Datenbank automatisiert zu verarbeiten. Sowohl das Generieren eines Vorschaubildes als auch die Extraktion der GPS-Koordinaten sind prädestiniert dafür, in der Datenbank ausgeführt zu werden, denn dort lassen sich die Ergebnisse direkt weiterverarbeiten.

Apex-Nutzer können die Bildfunktionen in einer der mitgelieferten Packaged Applications direkt erleben. Dazu ist lediglich die Packaged Application „Sample Geolocation Showcase“ zu installieren, die ab Version 4.2.5 zur Verfügung steht. Anhand dieser Anwendung lässt sich nachvollziehen, wie die Bildfunktionen konkret in einer Anwendung verwendet werden können.

## Weitere Informationen

1. Blog Posting des Autors, PKG\_ORDIMAGE: <http://sql-plsql-de.blogspot.co.uk/2013/06/image-processing-in-der-datenbank.html>
2. Beschreibung der APEX Packaged Application „Sample Geolocation Showcase“: [https://apex.oracle.com/pls/apex/GERMAN\\_COMMUNITIES.SHOW\\_TIPP?P\\_ID=2321](https://apex.oracle.com/pls/apex/GERMAN_COMMUNITIES.SHOW_TIPP?P_ID=2321)
3. Oracle-Dokumentation, Multimedia Reference: [http://docs.oracle.com/cd/E11882\\_01/app-dev.112/e10776/toc.htm](http://docs.oracle.com/cd/E11882_01/app-dev.112/e10776/toc.htm)



Carsten Czarski

[carsten.czarski@oracle.com](mailto:carsten.czarski@oracle.com)

<http://sql-plsql-de.blogspot.com>

<http://twitter.com/cczarski>