

Open Up! MusIS

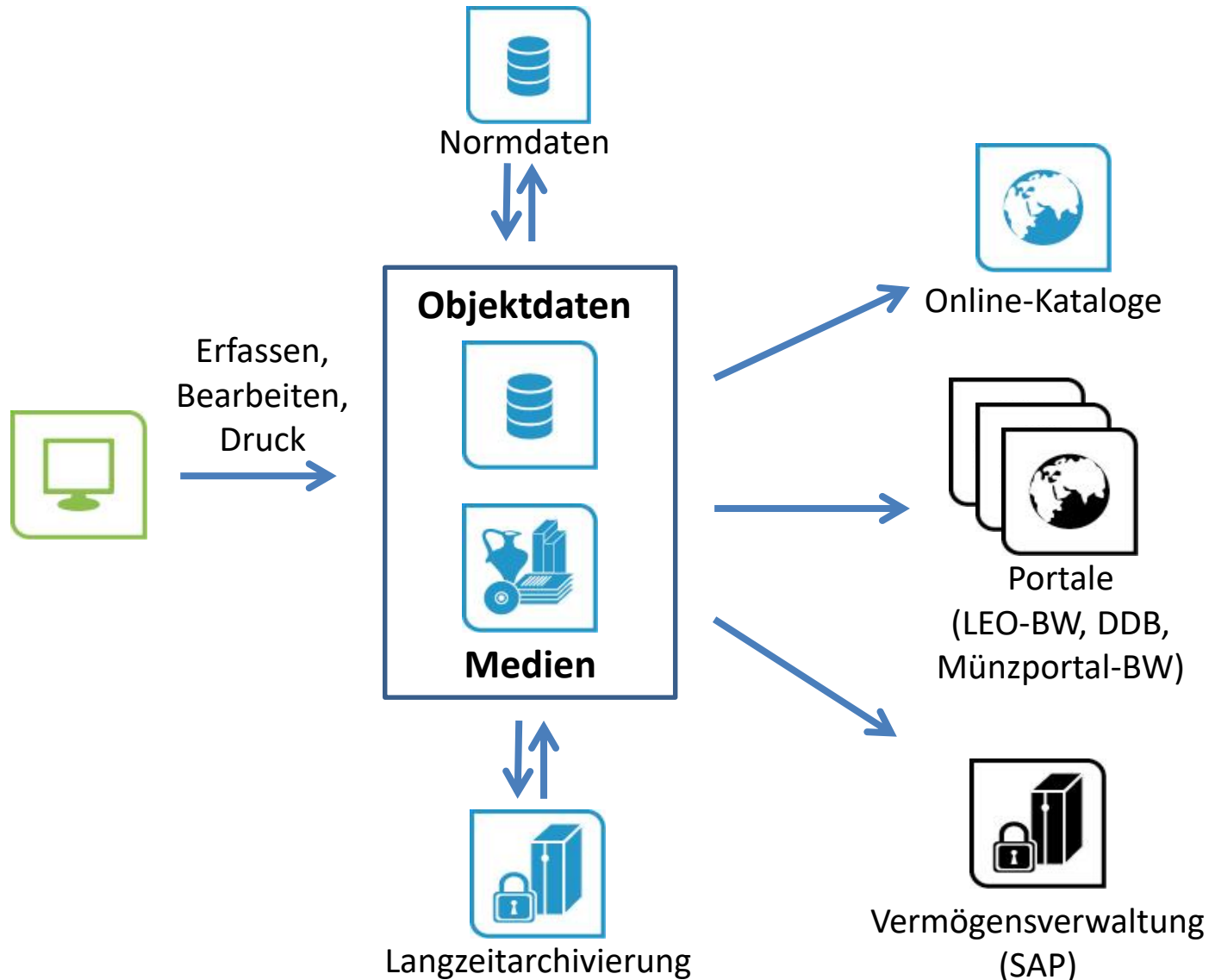
Open Up! Museum

war ein Projekt der MFG Innovationsagentur Medien- und Kreativwirtschaft Baden-Württemberg von 2014 bis 2016 mit dem Ziel, Museen zur digitalen Transformation zu ermutigen.

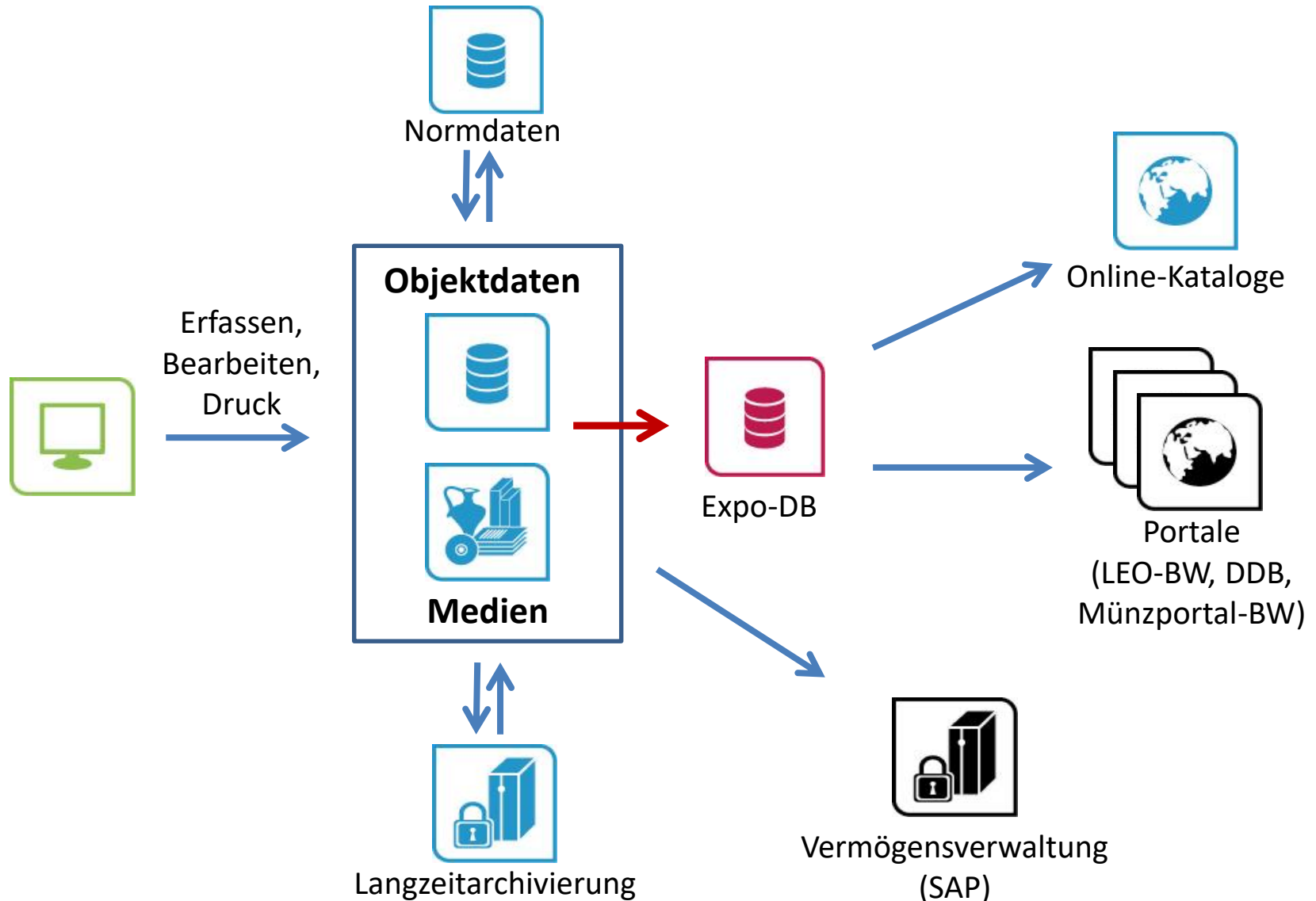
Ein wichtiger Treibstoff dafür ist die **Museumsdokumentation**

Christof Mainberger, Gotha, 20. März 2017

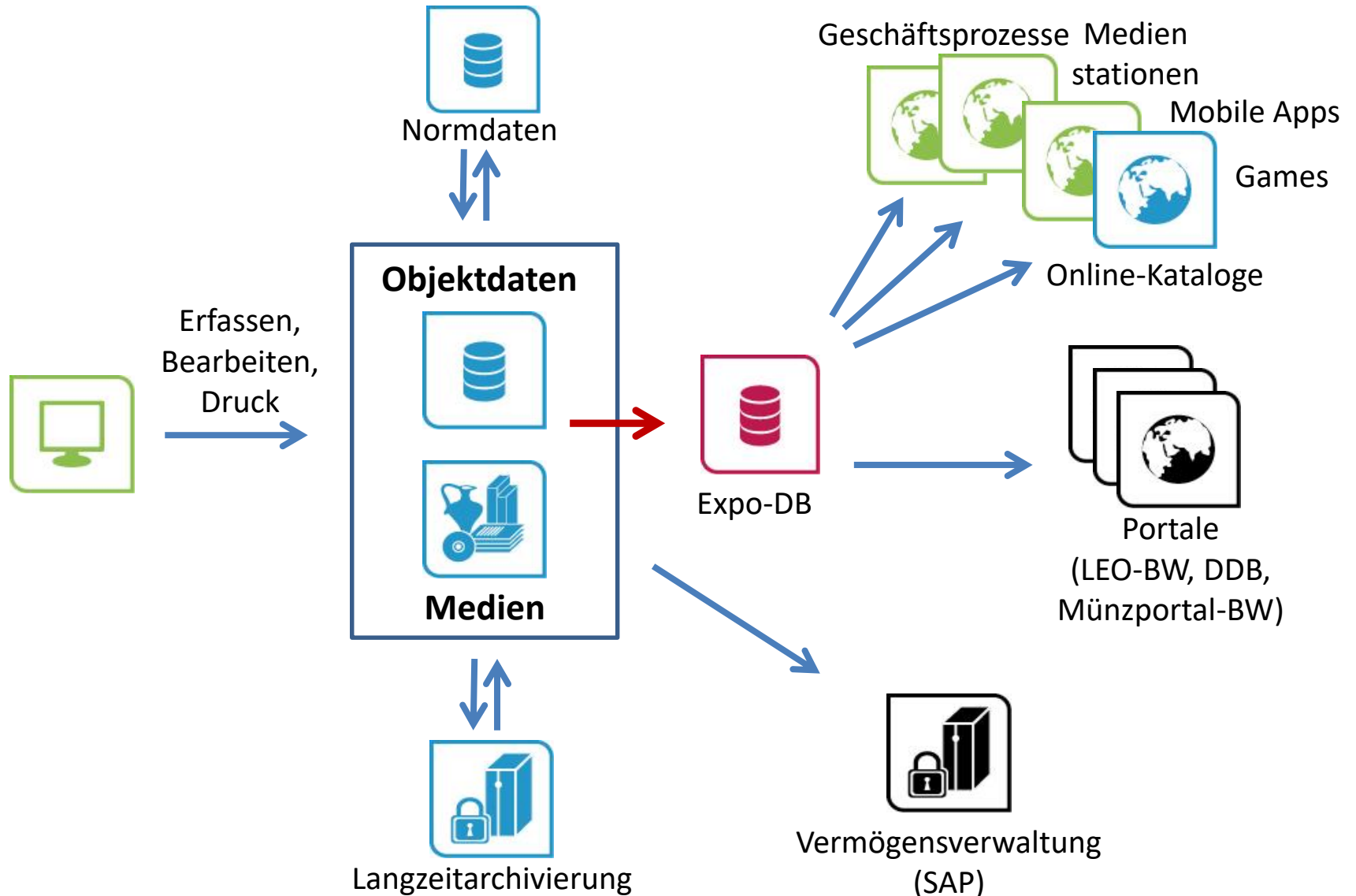
Datenflüsse rund um Imdas Pro in MusIS



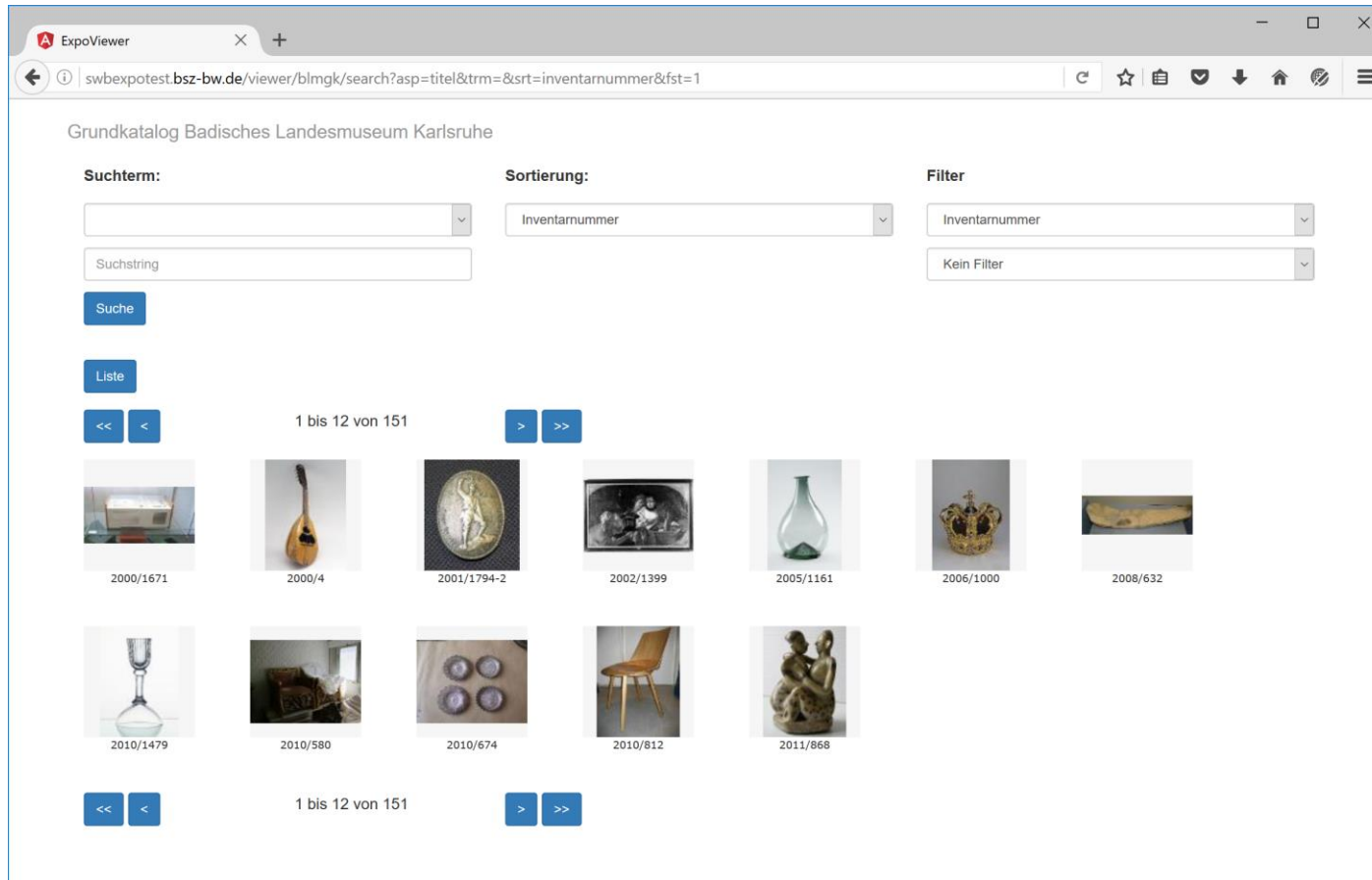
Eine Expo-DB für nicht-sensible Daten



Vielfältige Workflows auf Basis der Expo-DB



Beispiel 1: Digitaler Katalog



- Digitaler Katalog zum Grundkatalog des BLM
- Bezieht Daten und Bilder aus der Expo-DB via JSON

Ich kenne XML... Was aber ist JSON?

```
<data>
  <record id="38B072...2FAAA">
    <invnr>2005/1161</invnr>
    <titel>Flasche</titel>
    <datierung>
      <datum>15.-16. Jh</datum>
    </datierung>
    <schlagworte>
      <schlagwort>
        <term>Tischkultur</term>
        <guid>123453</guid>
      </schlagwort>
      <schlagwort>
        <term>
          Siedlungsarchäologie
        </term>
      </schlagwort>
    </schlagworte>
  </record>
</data>
```

```
{ "data": [
  { "id": "38B072...2FAAA",
    "invnr": "2005/1161",
    "titel": "Flasche",
    "datierung":
      { "datum": "15.-16. Jh." },
    "schlagworte": [
      { "term": "Tischkultur",
        "guid": "123453" },
      { "term":
          "Siedlungsarchäologie" }
    ]
  }
]
```

- JSON ist JavaScript
- JavaScript ist die Sprache der Browser

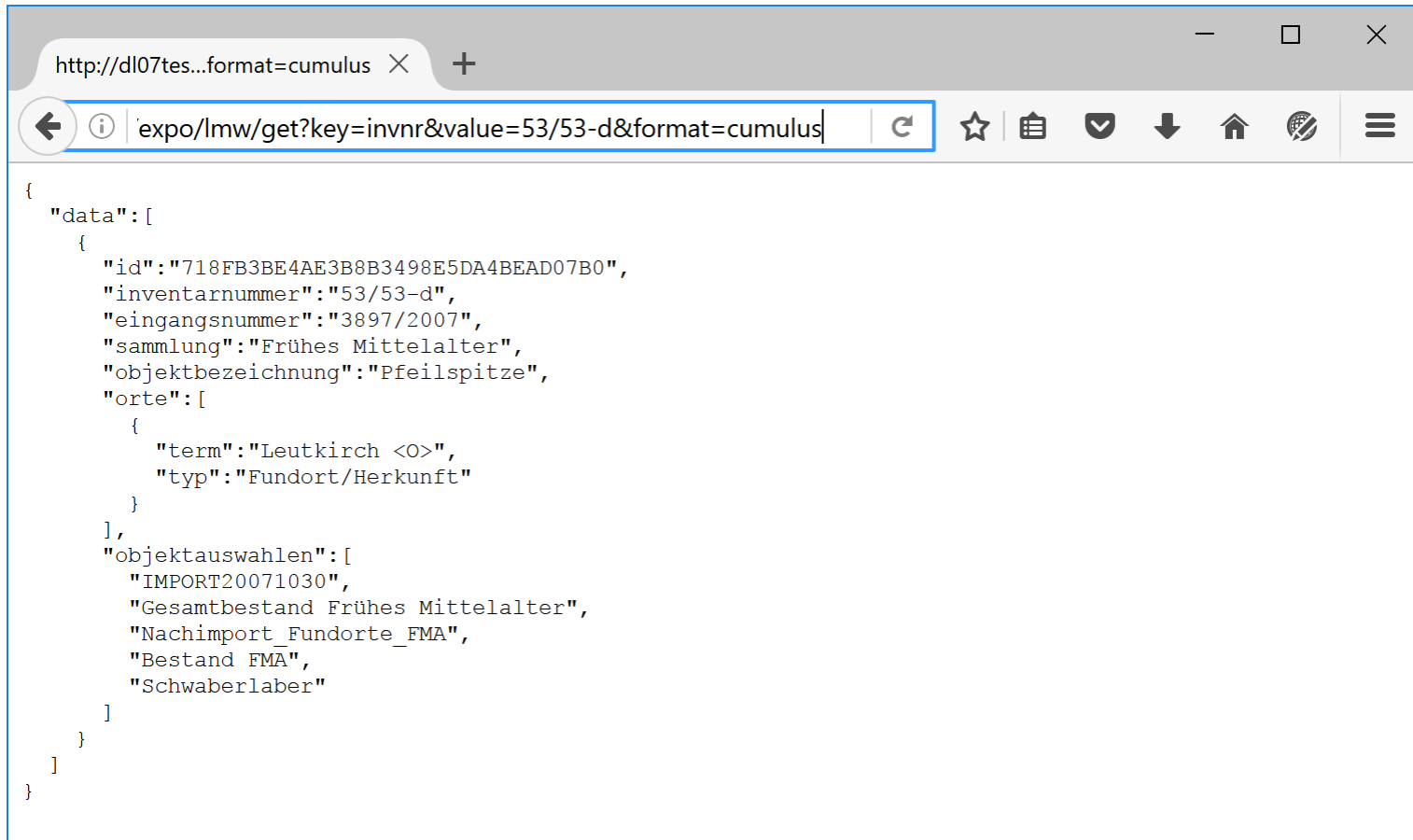
Beispiel 2: Bildanreicherung mit SWBtrafo



The screenshot shows a web browser window with the address bar displaying 'https://swbtrafo.bsz-bw.de'. The page title is 'SWBtrafo - Datenverarbeitung im Web' and the main heading is 'Startseite'. The page content includes a paragraph about data processing via an online interface, a description of the pipeline process, details about data upload formats (XML, CSV), and information about data output (XML, CSV, Solr-Index). A link 'Zum Login und zur Pipelineauswahl' is provided at the bottom.

- Aufbereitung von Importdateien zur Bildverknüpfung in imdas pro
- Bezieht Daten aus der Expo-DB und gleicht Inventarnummern ab

Beispiel 3: Datenlieferung an Cumulus

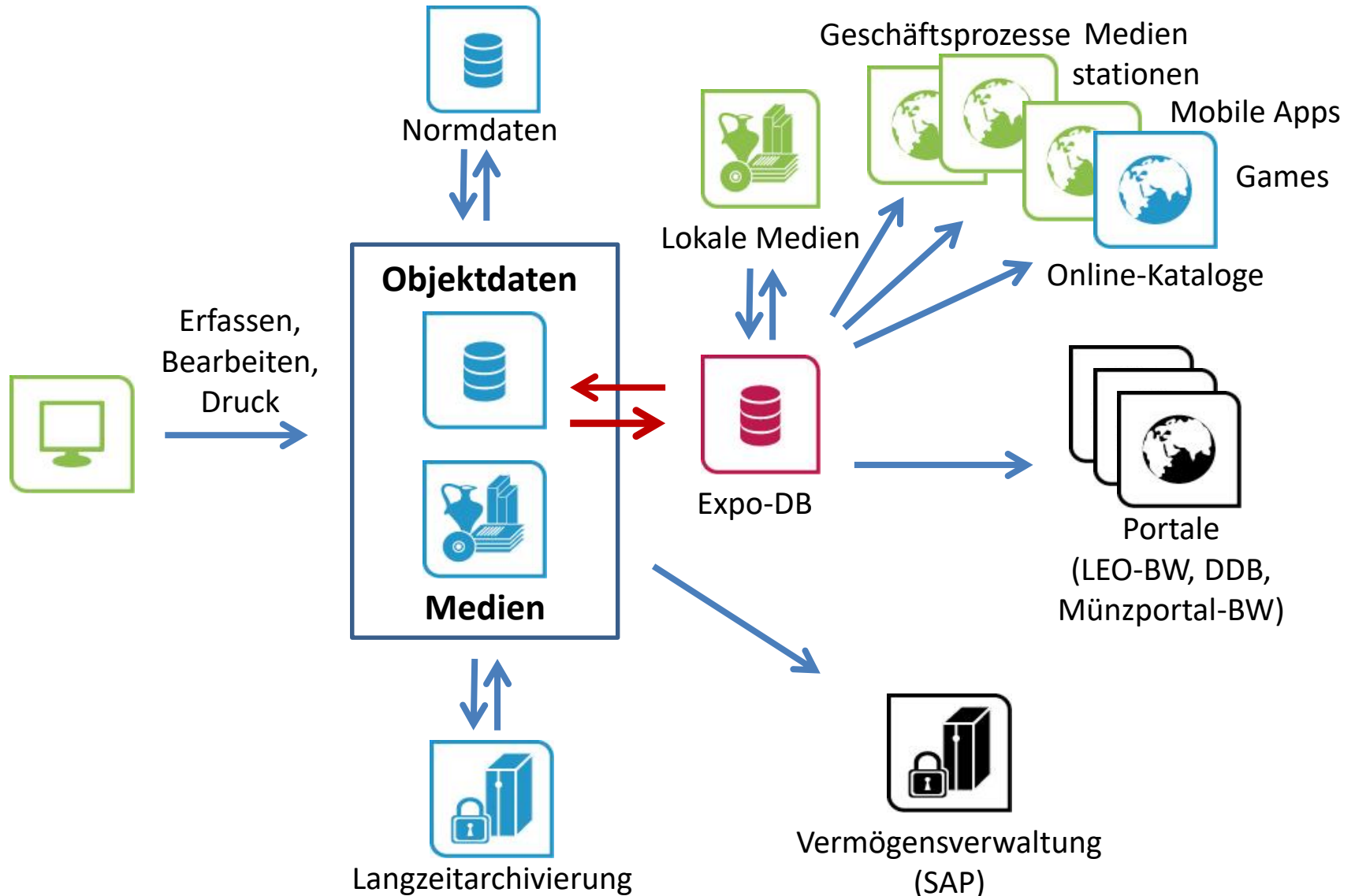


The screenshot shows a web browser window with the address bar containing the URL `http://dl07tes...format=cumulus`. The address bar also shows the full request path: `expo/lmw/get?key=invnr&value=53/53-d&format=cumulus`. The main content area of the browser displays a JSON response:

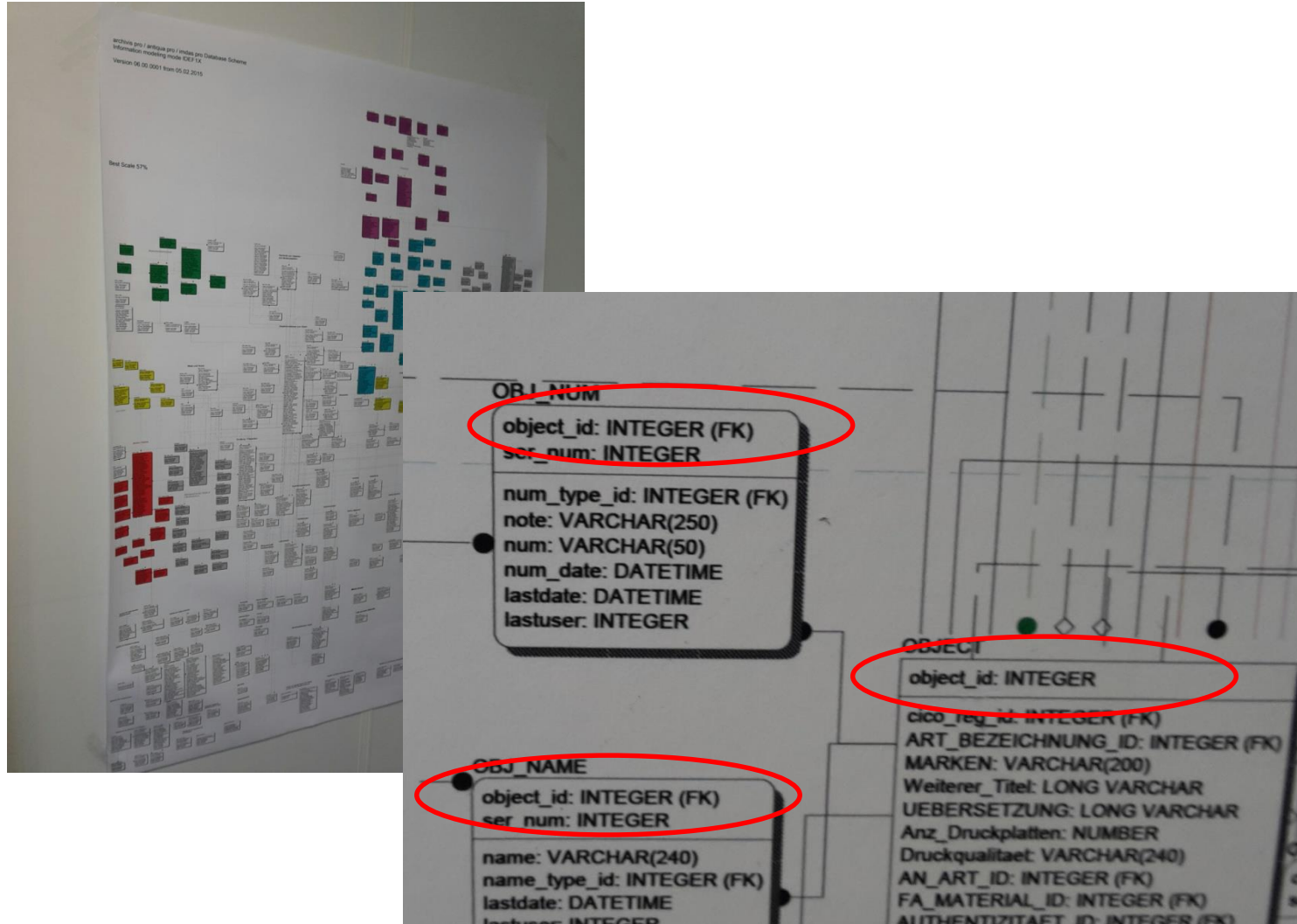
```
{
  "data": [
    {
      "id": "718FB3BE4AE3B8B3498E5DA4BEAD07B0",
      "inventarnummer": "53/53-d",
      "eingangsnummer": "3897/2007",
      "sammlung": "Frühes Mittelalter",
      "objektbezeichnung": "Pfeilspitze",
      "orte": [
        {
          "term": "Leutkirch <O>",
          "typ": "Fundort/Herkunft"
        }
      ],
      "objektauswahlen": [
        "IMPORT20071030",
        "Gesamtbestand Frühes Mittelalter",
        "Nachimport_Fundorte_FMA",
        "Bestand FMA",
        "Schwaberlaber"
      ]
    }
  ]
}
```

- Aufbereitung von Importdateien zur Bildverknüpfung in imdas pro
- Bezieht Daten aus der Expo-DB und gleicht Inventarnummern ab

Daten müssen auch nach Imdas fließen



Importe in die imdas-pro-DB sind komplex



Weiteres Vorgehen

- Datentransfer aus imdas pro automatisieren
- Von Test- auf Produktionsdatenbanken überführen
- Schnittstellen testen und optimieren

... und ein Konzept für das Einspielen von Daten nach imdas pro.

Herzlichen Dank