

Digitale Nachhaltigkeit von Forschungsanwendungen durch Microservices

FORGE 2016, Universität Hamburg
15.9.2016

Michael Haft, Anna Neovesky, Gabriel Reimers
Akademie der Wissenschaften und der Literatur | Mainz
team@digitale-akademie.de



Akademie
der Wissenschaften
und der Literatur
Mainz

Gliederung

1. Projektkontext an der Akademie
2. Nachhaltige Softwareentwicklung
3. Microservices
4. Beispiel: Incipit-Suche in der Gluck-Gesamtausgabe
5. Fazit

Akademie, Digitale Akademie und ihre DH-Projekte

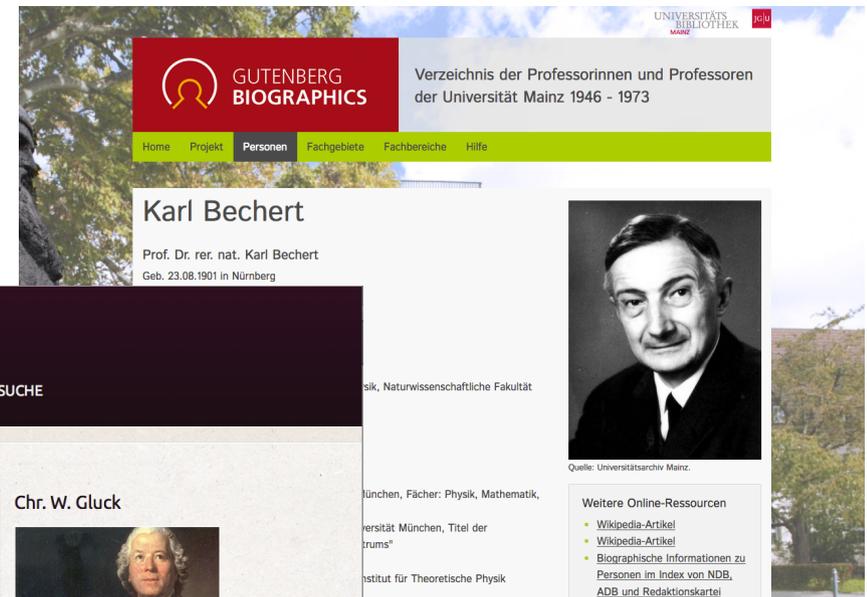
Akademie & Digitale Akademie

- Akademie: "Forschungsvorhaben mit Schwerpunkt langfristige Grundlagenforschung"
- Digitale Akademie: „Forschungseinrichtung für Digitale Geisteswissenschaften im Rahmen der Forschungsvorhaben der Mainzer Akademie.“
- Forschung an Kernfragen der Digitalen Geisteswissenschaften



Projekte der Digitalen Akademie

- Breites Fächerspektrum (Geschichtswissenschaft, Archäologie, Musikwissenschaft, ...)
- davon viele Editionsprojekte



<http://www.inschriften.net>

<http://gutenberg-biographics.ub.uni-mainz.de>

<http://www.gluck-gesamtausgabe.de>

Nachhaltige Softwareentwicklung

Nachhaltige Softwareentwicklung

- Konzeption von generischen Softwarekomponenten und Bezug auf die geisteswissenschaftliche Fragestellungen und Anwendungen ist ein Kerngebiet der DH
- Einen wesentlichen Aspekt moderner und nachhaltiger Software-Entwicklung stellt neben Neuentwicklungen auch die Nachnutzung dar.
- Die Verwendung verbreiteter Standards für die Datenhaltung und für die Programmierschnittstellen (APIs) erlaubt es Daten und Dienste im Sinne einer offenen Wissenschaft zu nutzen und weiterzuführen.

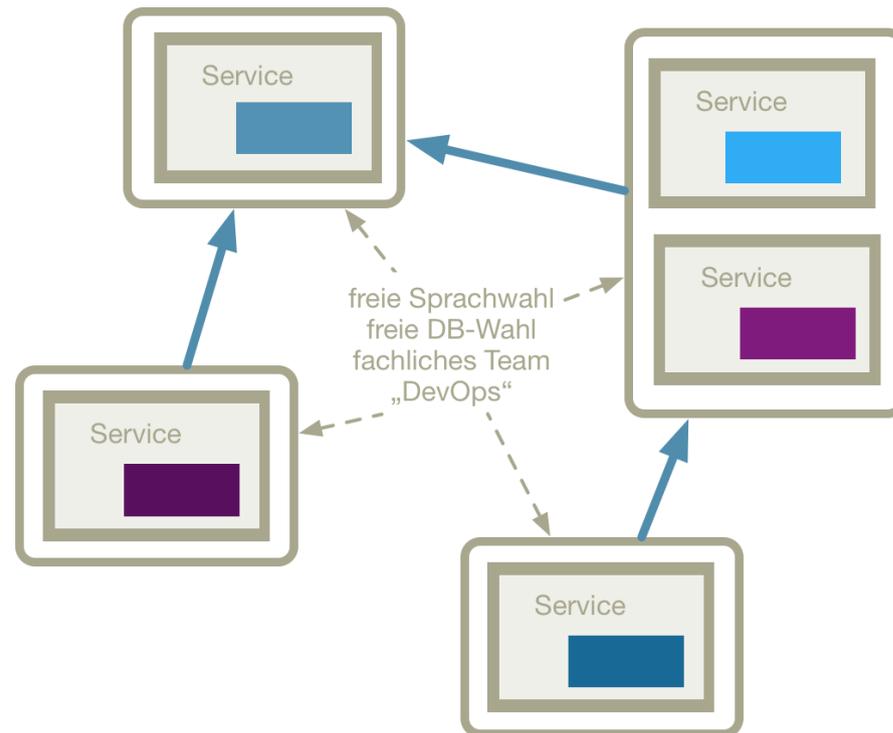
Nachhaltige Softwareentwicklung

- Hierfür sind jedoch nicht nur die lizenzrechtlichen Rahmenbedingungen und die dokumentierte Bereitstellung nötig, sondern auch eine Generalisierung der spezifisch für ein Forschungsvorhaben entwickelten Anwendungen.
- Dies kann am besten erreicht werden indem einzelne Funktionalitäten herausgelöst werden, die dann einzeln genutzt werden können und somit auch in einem breiteren Kontext anwendbar werden.
- Durch Generalisierung und Modularisierung können optimale Voraussetzungen für die Nachnutzbarkeit von Neuentwicklungen geschaffen werden.

Microservices

Funktionsweise

- Aufteilung der Software in Komponenten
- Eine Komponente erfüllt jeweils eine konkrete Aufgabe
- jeweils einzeln voll funktional
- Kommunikation über wohldefinierte Schnittstellen
- skalierbar



Vor- und Nachteile

- Wenige(r) Codezeilen
- Dadurch bessere Wartbarkeit
- Austauschbarkeit
- Unabhängigkeit
- Weiter-/Wiederverwendbarkeit

aber:

- verteilte Architektur \Rightarrow komplexeres System
- größerer Dokumentationsaufwand

Dokumentation

- Dokumentation der Funktionalitäten
- Dokumentation der Schnittstellen
- Paketverwaltung zur Verwaltung mehrerer Microservices

IncipitSearch

A crawler and search interface for musical incipits.

This project was developed to index the incipits of the Gluck-Gesamtausgabe catalog. It crawls the catalog for incipits, processes the entries and indexes them using an Elastic Search Instance. The IncipitSearch provide a convenient search interface to search the incipits for melodies.

Even though it was developed to be used on the Gluck-Gesamtausgabe it is build very modularly and can easily be adjusted to crawl and index any other musical catalog for incipits.

Dependencies

To make use of the IncipitSearch you need two things:

- An [Elastic Search](#) instance
- A catalog of musical incipits with a public interface

The IncipitSearch web-interface uses the virtual keyboard of the [pianoKeyboard](#) project, which is directly integrated in the code.

To display notes on the website, IncipitSearch makes use of the [Verovio Toolkit](#)

Usage

Structure



Incipit-Suche

Hintergrund und Anforderungen

- Suchzugriff über Melodien der Werkteile (Incipits)
- Anzeige von Incipits

The screenshot displays the 'GluckGesamtausgabe' website interface. At the top, a dark purple header contains the site title and navigation links: 'PROJEKT', 'BÄNDE', 'WERKVERZEICHNIS', and 'LITERATURSUCHE'. Below this, a light beige navigation bar includes 'Werksuche', 'Quellensuche', 'Register' (highlighted in red), 'Hinweise', and 'Abkürzungen'. The main content area is divided into two columns. The left column features a 'Zurück zur Liste' button, the title 'GluckWV 1.1 Artaserse', and the section 'Allgemeines'. Under 'Allgemeines', it lists 'Wotquenne-Nummer: A.1', 'Werktyp : Oper', 'Werkbezeichnung : Drama per musica in drei Akten', 'Uraufführungsort : 26. Dezember 1741 , Mailand , Teatro Regio Ducale', and 'Besetzung : Artaserse, principe, e poi re di Persia, amico d'Arbace, ed amante di Semira (Soprano), Mandane, sorella di Artaserse, ed amante d'Arbace (Soprano), Artabano, prefetto delle Guardie Reali, padre di Arbace, e di Semira (Tenore), Arbace, amico d'Artaserse, ed amante di Mandane (Alto), Semira, sorella d'Arbace, ed amante d'Artaserse (Soprano), Megabise, generale dell'armi, e confidente d'Artabano (Tenore); Archi, Cembalo'. The right column has a dark purple 'Allgemein' header with a list of sub-sections: 'Personen', 'Genese', 'Werkteile', 'Übernahmen', 'Quellen', and 'Literatur'. The 'Werkteile' section is highlighted in red.

Kodierung der Musikincipits

- Plaine & Easie Code
- Aufbau: Schlüssel, Vorzeichen, Takt, Noten (Oktaven, rythm. Angaben, Vorzeichen, Pausen ...)



```
@clef:G-2  
@keysig:xF#CGD  
@timesig:3/8  
@data:'6B/{8B+(6B''E'B)}({AFD})/{6.E3G},8B-/{6'EGF)}({FAG)}({GEB})/4F6-
```

Incipit Anzeige

GluckGesamtausgabe

PROJEKT BÄNDE WERKVERZEICHNIS LITERATURSUCHE

Werksuche Quellensuche Incipitsuche **Register** Hinweise Abkürzungen Daten

Zurück zur Liste

GluckWV 1.1

Artaserse

Werkteile

(Sinfonia)

1.1.2 S solo [Timante], c ; D



(Recitativo: "Addio. Sentimi Arbace")

Akt : I

Szene : 1

(Aria: "Conservati fedele")

- Allgemein
- Personen
- Genese
- Werkteile**
- Übernahmen
- Quellen
- Literatur

Die Incipit-Suche

Chrome File Edit View History Bookmarks People Window Help

Gluck Incipit Suche

incipitsearch.local/

Apps http://localhost:808... Gluck Incipit Suche

Gluck Incipit Suche

Geben Sie über die virtuelle Tastatur eine kurze Melodie ein, um nach Werken zu suchen.

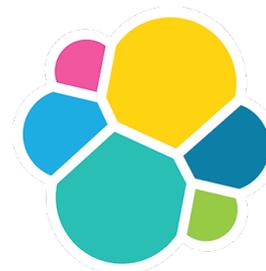
Plaine & Easie Code LÖSCHEN SUCHEN

C4 B #

Technische Umsetzung

Komponenten

- Applikation nutzt verschiedene Daten und Software nach
- für Entwicklung: Kodierte Incipits von RISM (Répertoire International des Sources Musicales)
- Verovio Toolkit
- Searchengine Elasticsearch



elastic



Gluck, Christoph Willibald

[ermittelt]

Paride ed Elena (Excerpts)

TERZETTO FROM THE 5TH ACT OF THE OPERA IN 5 ACTS

▼ Werkinformation

Werkverzeichnis: [WotG](#) 1A.39
Text: [Calzabigi, Ranieri de](#) [ermittelt]
Sprache: Deutsch, Italienisch
Schlagwort: Opern

▼ Quellenbeschreibung

Originaler Titel: *[heading:] Terzetto. I zu Orpheus. I [later added in pencil:] Gluck, Orpheus*
Material:

- score: 22p.
Abschrift: 1800 (1800c); 24,5 x 30,5 cm

▼ Musikincipits

1.1.1 vl 1, 3/8 Andante con portamento; C



1.1.2 S, 3/8 ; C



– Ewig werd' an dir ich hängen; [Sempre a te sarò fedele]

▼ Weitere Angaben und Bemerkungen

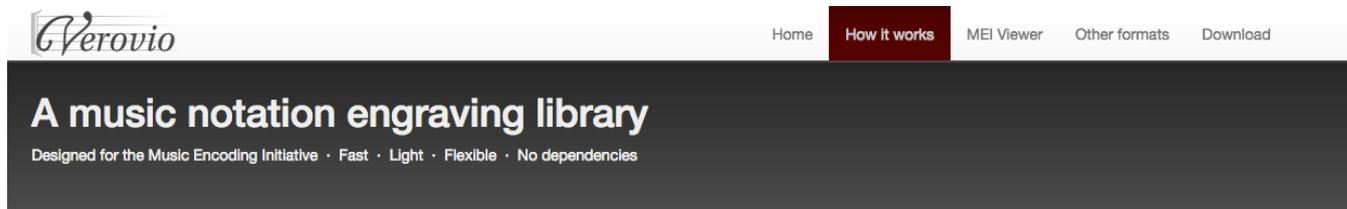
Besetzung: S (3), vl 1, vl 2, vla, vlc, b
Aufführungen : 03.11.1770 Wien, Burgtheater
Digitalisat: [Weiterleitung zu resolver.staatsbibliothek-berlin.de](#)

Mit dieser XML-Datei sind anscheinend keine Style-Informationen verknüpft. Nachfolgend wird die Baum-

```
- <record>
  <leader>00000cdd#a2200000###4500</leader>
  <controlfield tag="001">452020150</controlfield>
  <controlfield tag="003">RISM</controlfield>
  <controlfield tag="005">20110429111111.0</controlfield>
  <controlfield tag="008">140112s1800||||gw ||||||||||||lgerld</controlfield>
- <datafield tag="031" ind1=" " ind2=" " >
  <subfield code="a">1</subfield>
  <subfield code="b">1</subfield>
  <subfield code="c">1</subfield>
  <subfield code="d">Andante con portamento</subfield>
  <subfield code="g">G-2</subfield>
  <subfield code="m">vl 1</subfield>
  <subfield code="o">3/8</subfield>
  <subfield code="p">4G{6AF}/{8FEG}/{GAB}/B</subfield>
  <subfield code="r">C</subfield>
  <subfield code="2">pe</subfield>
  <subfield code="u">30915423</subfield>
</datafield>
- <datafield tag="031" ind1=" " ind2=" " >
  <subfield code="a">1</subfield>
  <subfield code="b">1</subfield>
  <subfield code="c">2</subfield>
  <subfield code="g">C-1</subfield>
  <subfield code="m">S</subfield>
  <subfield code="o">3/8</subfield>
  <subfield code="p">=4/4"E{6FD}/q8D4C8C/4F8F/{6FE}4E/</subfield>
  <subfield code="r">C</subfield>
```

Verovio

- „A music notation engraving library - Designed for the Music Encoding Initiative“
- Entwickelt von RISM Schweiz (LGPLv3 license)
- JavaScript Toolkit zur Anzeige von Musiknoten im Web



The screenshot shows the top part of the Verovio website. On the left is the Verovio logo. On the right is a navigation menu with links for Home, How it works (highlighted in red), MEI Viewer, Other formats, and Download. Below the navigation is a dark grey banner with the text "A music notation engraving library" and "Designed for the Music Encoding Initiative · Fast · Light · Flexible · No dependencies".

Features
SMuFL
Command-line tool
JavaScript toolkit
Python toolkit
Structure

JavaScript toolkit

Verovio can be compiled to JavaScript using the [Emscripten](#) LLVM-to-JavaScript compiler. In this case, it behaves similarly to the command-line tool but in the web-browser. The SVG output of Verovio can be directly fed to HTML objects for display. This approach is particularly interesting because it makes Verovio an in-browser music Plain and Easy or MEI typesetter. The [MEI Viewer](#) illustrates its possible use.

Basic usage

The JavaScript version of Verovio is available as a toolkit that makes it very easy to integrate.

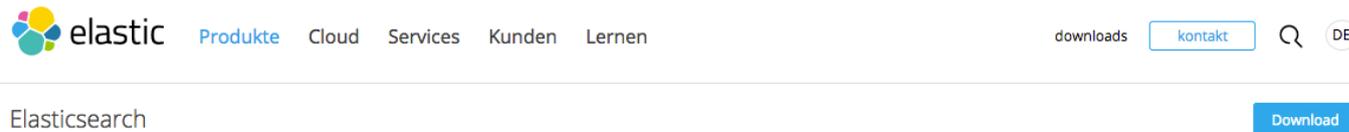
```
<script src="http://www.verovio.org/javascript/latest/verovio-toolkit.js"></script>
```

Once included, it can be used to render music notation on the fly:

```
!- The div where we are going to insert the SVG ->  
<div id="output"/>  
<script type="text/javascript">  
  /* The Plain and Easy code to be rendered */  
  var data = "@clef:G-2\n\  
             @keysig:xFGD\n\  
             @timesig:3/8\n"
```

ElasticSearch

- Suchmaschine/Suchserver auf Basis von Lucene
- kommuniziert über REST



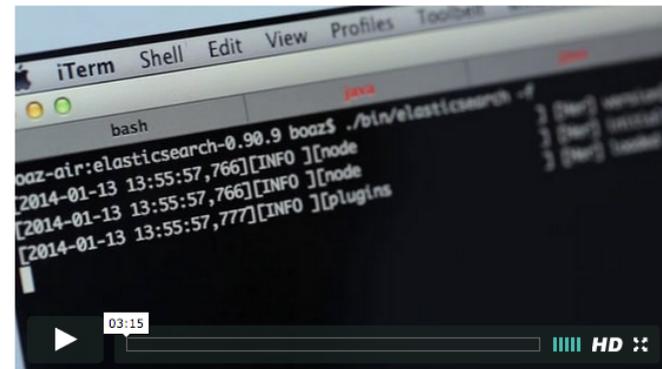
Search & Analyze Data in Real Time

Actionable Insight at Your Fingers

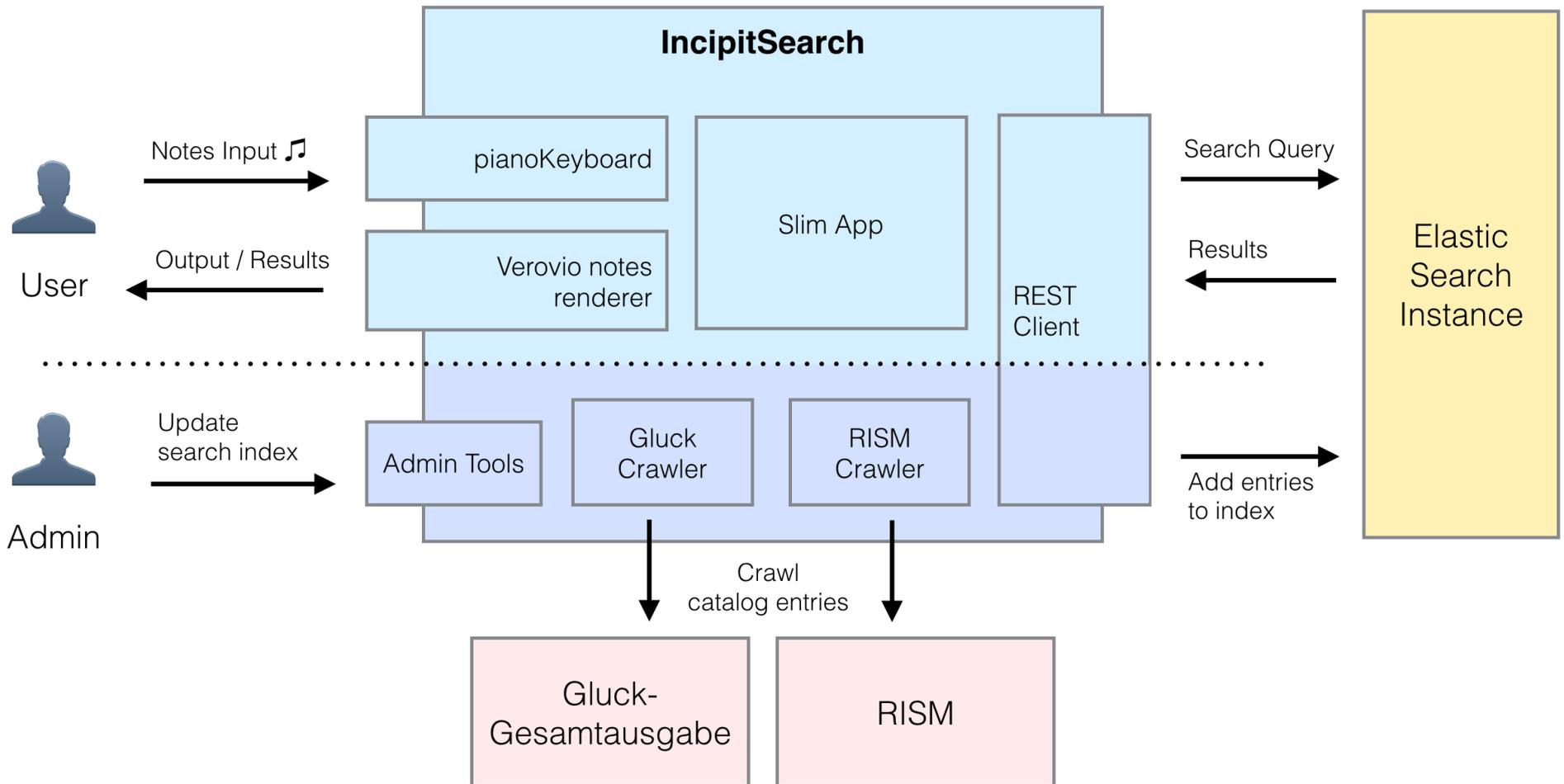
- Distributed, scalable, and highly available
- Real-time search and analytics capabilities
- Sophisticated RESTful API

[Download](#) | [Subscriptions](#)

Watcher (Alarmierung) Übersicht & Demo Video [Jetzt anschauen](#) ▶



Softwarearchitektur



Perspektiven der Nachnutzbarkeit

- einzelne Softwarekomponenten
- Plattform
- Nachnutzung der vollen Funktionalität bei Einbindung einer anderen Plattform

Fazit

- Bessere Möglichkeiten der Nachnutzbarkeit
- Bessere Wartbarkeit

Wenn ein Aspekt der Überarbeitung bedarf, muss nur dieser nachgebessert werden

- Bessere Portierbarkeit und Pflege

⇒ Microservices als Architekturmodell für nachhaltige Softwareentwicklung haben sich als ein praktikabler Ansatz erwiesen und sollten in den DH weiter eruiert, getestet und benutzt werden.

Danke für Ihre Aufmerksamkeit!