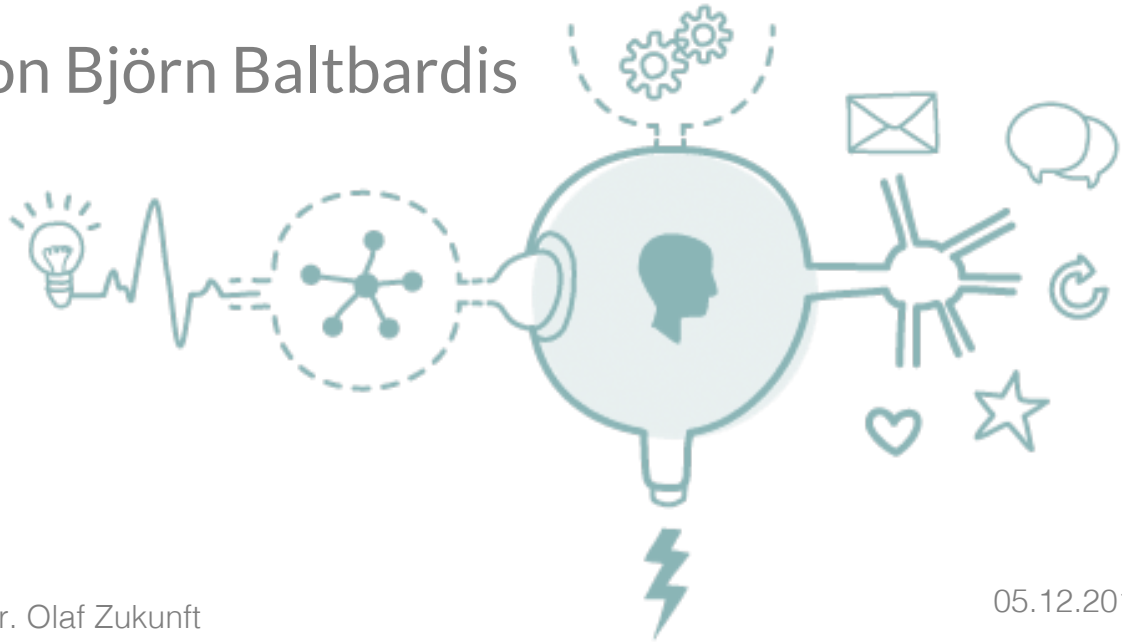
API Monitoring mit Predictive Analytics

von Björn Baltbardis



Inhalt des Vortrags

- Einführung
  - Motivation
  - XING API
- Erkennungsmetriken
- Lösungsstrategien
  - Asynchrone Auswertung
  - Information Flow Processing
- Ziele und Ausblick

Motivation

- XING
  - soziales Netzwerk
  - 14 Mio. Mitglieder [XNG1]
  - ca.14 Mio. Requests / Tag
  - ca. 60-80% des Traffic entfallen auf die API
- Großes Datenaufkommen
  - schwer angemessen zu überwachen
- Probleme
  - Fehler
  - Missbrauch

[Abb. 1]

XING API

XING API

- Öffentliche REST API
- 89 Resources in 15 Kategorien

Table of contents

→ User Profiles
→ Profile Editing
→ Jobs
→ Groups
→ Messages
→ Profile Messages
→ Contacts
→ Contact Requests
→ Contact Path
→ Bookmarks
→ Network Feed
→ Profile Visits
→ Recommendations
→ Invitations
→ Geo Locations

Resource Beispiel - XING API

GET	/v1/users/me	Shows a particular user's profile.

Parameters
fields (OPTIONAL)

Response (JSON)

```
{
  "users": [
    {
      "id": "17425810_c9de17",
      "active_email": "bjoern@baltbardis.de",
      "badges": [
        "PREMIUM"
      ],
      "birth_date": {
        "year": 1992,
        "month": 1,
        "day": 23
      },
    [.....]
}
```

Response (XML)

```
<?xml version="1.0" encoding="UTF-8"?>
<users>
 <user>
  <id>17425810_c9de17</id>
  <active_email>bjoern@baltbardis.de</active_email>
  <badges>
   <badge>PREMIUM</badge>
  </badges>
  <birth_date>
   <year>1992</year>
   <month>1</month>
   <day>23</day>
  </birth_date>
  [.....]
 </user>
</users>
```

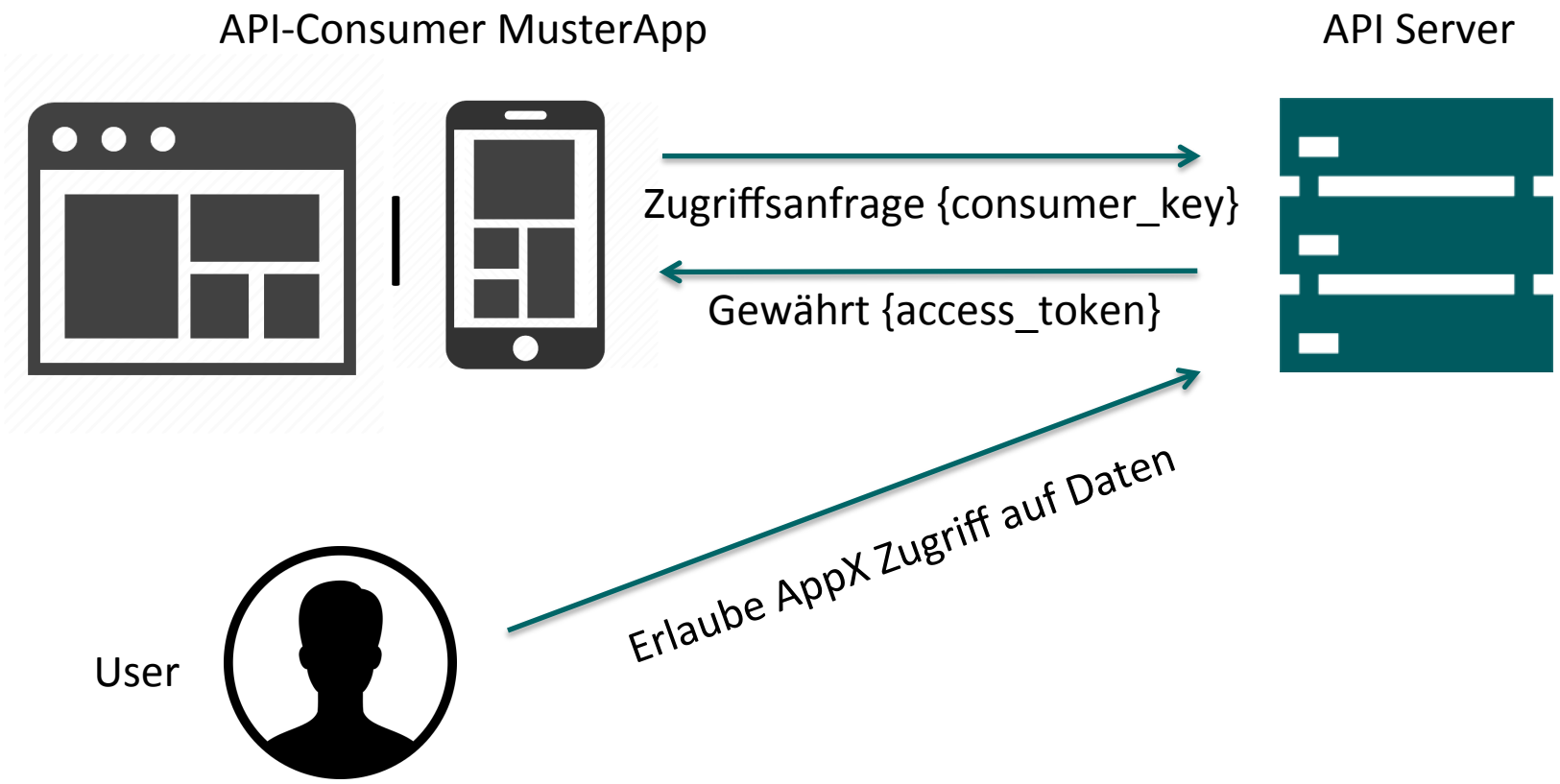
[XNG2]

ERKENNUNGSMETRIKEN

- **Missbrauchserkennung**
- Fehlerprävention
- Technische Gegebenheiten

OAuth – Missbrauchsrisiko

- Mehrstufiger Authentifizierungsmechanismus

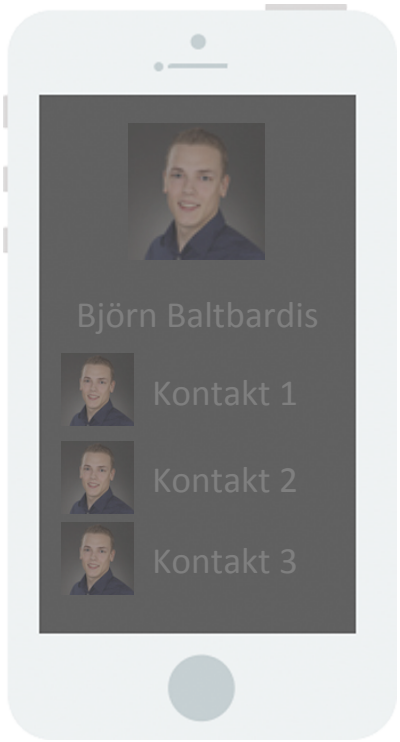


OAuth - Missbrauchsrisiko

- *consumer key* kann gestohlen, missbraucht werden
  - Verschleierung von API Nutzung
  - Datenabruf im falschen Namen
  - Rechtemissbrauch
  - Umgehen von Thresholds beim Throttling
    - od. für andere Apps erhöhen

Missbrauchserkennung

MusterApp



Beispielhafter Ablauf:

1. Profil abrufen

GET `/v1/users/me/`

```
“fields": “display_name, photo_urls“,
"consumer_key": “123XXXXXXXXXXXXXXXX“,
"access_token": “ABCXXXXXXXXXXXXXXXX“
```

2. Kontakte abrufen

GET `/v1/users/me/contacts`

```
"consumer_key": “123XXXXXXXXXXXXXXXX“,
"access_token": “ABCXXXXXXXXXXXXXXXX“
```

3. Kontaktanfragen abrufen

GET `/v1/users/:user_id/contact_requests`

```
"consumer_key": “123XXXXXXXXXXXXXXXX“,
"access_token": “ABCXXXXXXXXXXXXXXXX“
```

➔ Verhalten des consumer keys „123XXXXXXXXXXXXXXXX“

Missbrauchserkennung

BadApp

Mögliche Verdachtsfälle:

Andere Parameter

GET `/v1/users/me/`

```
“fields": “display_name, private_address“,
"consumer_key": “123XXXXXXXXXXXXXXXX“,
"access_token": “XYZXXXXXXXXXXXXXXXX“
```

Bisher ungenutzte Resources

PUT `/v1/users/me/web_profiles/facebook`

```
"consumer_key": “123XXXXXXXXXXXXXXXX“,
"access_token": “XYZXXXXXXXXXXXXXXXX“
```

Anomalien gegenüber bisherigen Abfolgen/Algorithmen

1. **GET** `/v1/users/me/contacts`
2. **GET** `/v1/users/me/`

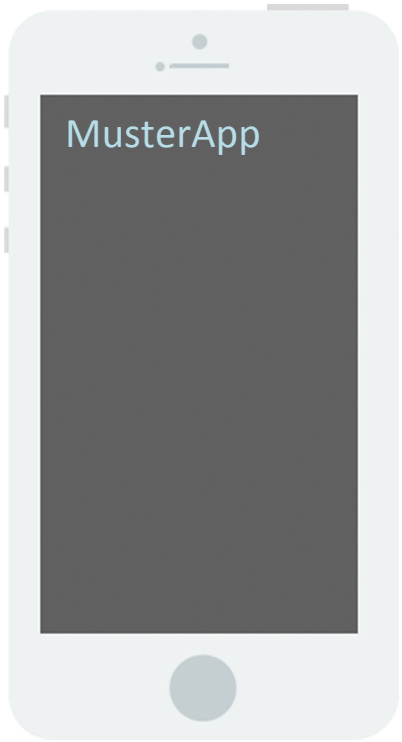
ERKENNUNGSMETRIKEN

- Missbrauchserkennung
- **Fehlerprävention**
- Technische Gegebenheiten

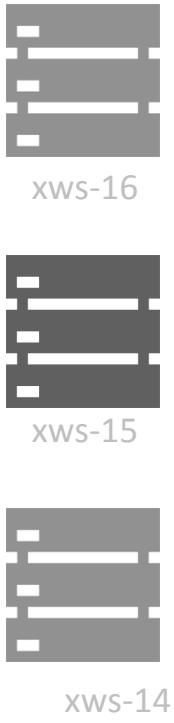
Fehler Prävention

- Was ist ein Fehler?
  - Server gibt 400er oder 500er Status-Code zurück
- Wiederkehrende Muster bei Fehlern:
  - Ähnliche Abfolge
  - Ähnliche Parameter
  - Häufung bei bestimmten *consumer_key's*
- Vorteile
  - Server-Fehler frühzeitig erkennen
    - häufig Sicherheitsrelevant
    - frühzeitige Berichtigung experimenteller Features
  - Detaillierte Fehleranalyse Tools für *consumer*

Technische Gegebenheiten



API Server



RabbitMQ Message

```
{
    "duration" : 0.314159265359,
    "recorded_at" : "2014-12-05T12:48:31+02:00",
    "method": "GET",
    "path_info": "/v1/users/me/contacts",
    "host": "xws-15.xxx.com",
    "params": {
      "limit": "100",
      "offset": "0",
    },
    "status": 200,
    "size": 1234,
    "ip_address": ”XXX.XXX.XXX.XXX",
    "consumer_key": "123XXXXXXXXXXXXXXXX",
    "access_token": "ABCXXXXXXXXXXXXXXXX"
}
```

```
GET /v1/users/me/contacts
    "consumer_key": “123XXXXXXXXXXXXXXXX“,
    "access_token": “ABCXXXXXXXXXXXXXXXX“
```

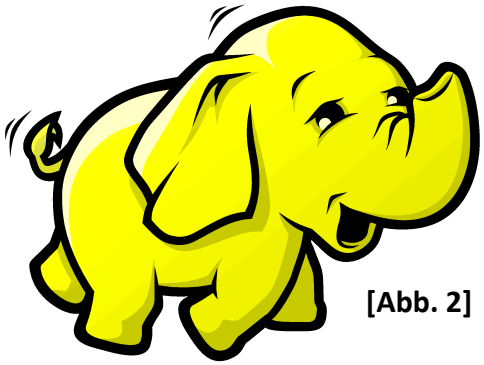
Analyse Server (Master)

LÖSUNGSSTRATEGIEN

- **Asynchrone Auswertung**
- Information Flow Processing

Asynchrone Auswertung

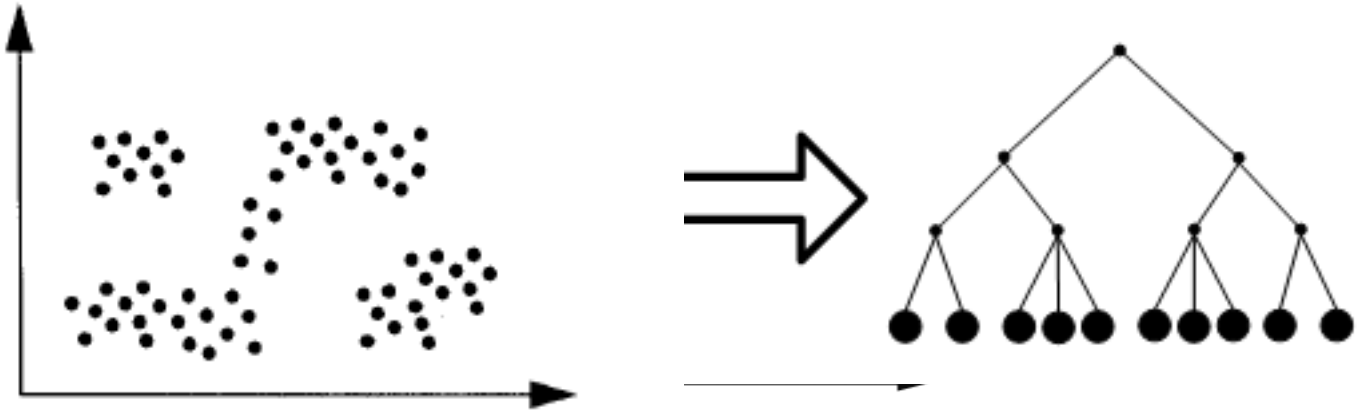
- Im Cluster aus Performancegründen
- Hadoop - de facto Standard
  - Eigenes, verteiltes Dateisystem
  - MapReduce
- Viele Erweiterungen:
  - **Hive**, SQL ähnliche Abfragesprache
  - **Pig**, Erweitert MapReduce Fähigkeiten
  - **Drill**, Analyse und Query Engine
  - **Mahout**, Verteiltes *machine learning* Modul



[Abb. 2]

Clustering

- Grundlage für Anomalieerkennung



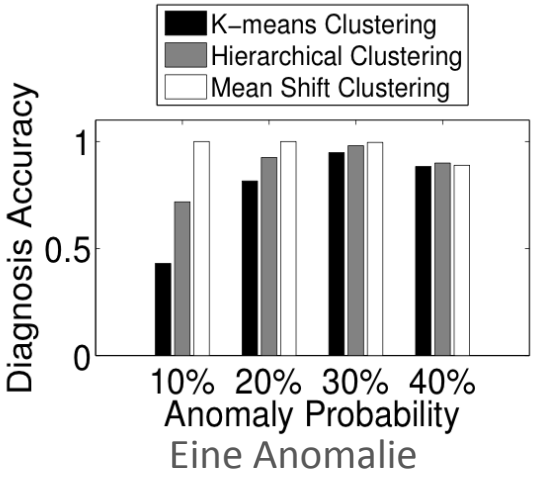
[Abb. 3.1, 3.2, 3.3]

- Typische Verfahren:
  - partitionierendes Clustering (z.B. k-means)
  - hierarchisches Clustering
  - dichtenbasiertes Clustering

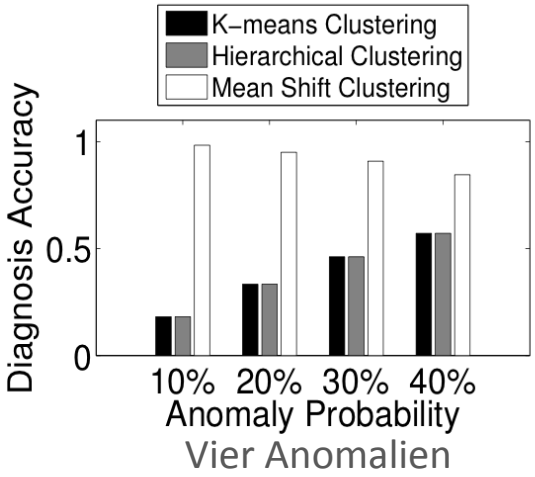
Anomalie-Erkennung mit Hadoop

„A Scalable, Non-Parametric Anomaly Detection Framework for Hadoop“

- Skalierbare Anomalie-Erkennung
- Clustering
  - „non parametric clustering“ (Mean Shift Clustering)



Eine Anomalie



Vier Anomalien

LÖSUNGSSTRATEGIEN

- Asynchrone Auswertung
- **Information Flow Processing**

Information Flow Processing

- Keine Speicherung
- Data Stream Processing (DSP)
  - Aus DBMS entstanden
  - SQL ähnliche, beständige Abfragen
- Complex Event Processing (CEP)
  - Informationen = Events
  - Atomare Events -> höherwertige Events
  - Eigene Sprache zur Definition von Mustern

Vergleich IFP Frameworks

„Processing flows of information: From data stream to complex event processing“

- Gegenüberstellung DSP / CEP
- 35 Sprachen bzw. Frameworks untersucht
- Betrachtet:
  - Verteilung, Benachrichtigung, Zeitmodell, Datenmodell, Regelmodell

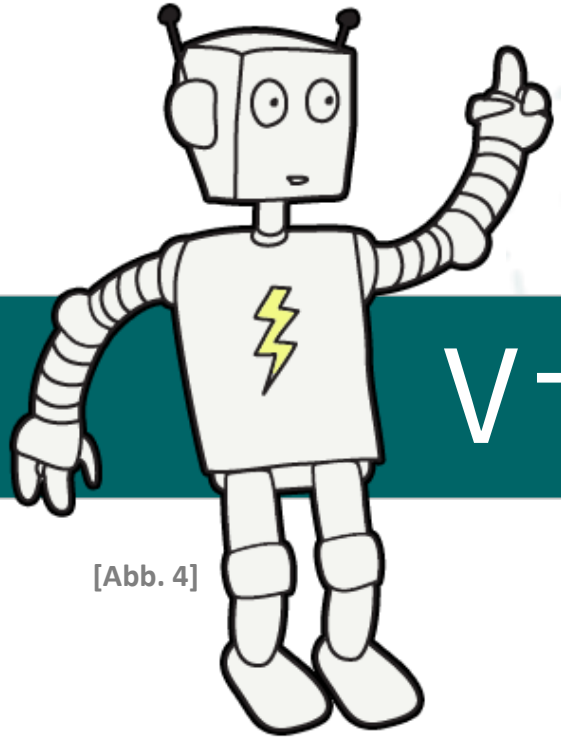
„Scalable hybrid stream and hadoop network analysis system“

- Hadoop + CEP => Storm
- Stream Daten angereichert mit persistenten Daten

ZIELE UND AUSBLICK

Ziele und Ausblick

- Grundprojekt
  - Analyse und Tests der Lösungsstrategien mit Testdaten
    - Generator für Testdaten
  - Vergleich versch. Algorithmen
- Hauptprojekt und Masterthesis
  - Vertiefung
  - Sammeln und Auswerten von Produktivdaten
  - Ausarbeitung von Prozess bei Anomalie



[Abb. 4]

Vielen Dank!

Fragen?

Quellen

- **XNG1**, “XING ist das soziale Netzwerk für berufliche Kontakte.” https://corporate.xing.com/deutsch/unternehmen [Stand 28.11.14]
- **XNG2**, “Resources” https://dev.xing.com/docs/resources [Stand 01.12.14]
- **XNG3**, “Brief information about Oauth” https://dev.xing.com/docs/authentication [Stand 01.12.14]
- **HAD1**, http://hadoop.apache.org/
- **CUG1**, Gianpaolo Cugola and Alessandro Margara. 2012. Processing flows of information: From data stream to complex event processing. ACM Comput. Surv. 44, 3, Article 15 (June 2012)
- **ZHA1**, Haopeng Zhang, Yanlei Diao, and Neil Immerman. 2014. On complexity and optimization of expensive queries in complex event processing. In Proceedings of the 2014 ACM SIGMOD international conference on Management of data (SIGMOD '14). ACM, New York, NY, USA
- **DIA1**, Yanlei Diao, Neil Immerman and Daniel Gyllstrom, SASE+: An Agile Language for Kleene Closure over Event Streams, University of Massachusetts, Amherst
- **BUM1**, Vernon K.C. Bumgardner and Victor W. Marek. 2014. Scalable hybrid stream and hadoop network analysis system. In Proceedings of the 5th ACM/SPEC international conference on Performance engineering (ICPE '14). ACM, New York, NY, USA
- **YU1**, Li Yu and Zhiling Lan. 2013. A scalable, non-parametric anomaly detection framework for Hadoop. In Proceedings of the 2013 ACM Cloud and Autonomic Computing Conference (CAC '13). ACM, New York, NY, USA
- **CLE1**, Jürgen Cleve and Uwe Lämmel, Data Mining, De Gruyter Oldenbourg, 978-3486713916

Abbildungen

- **Abbildung 1** ,Folie 3, XING Logo,
  https://corporate.xing.com/deutsch/presse/bildarchiv/xing-logos/logos-printformat/
- **Abbildung 2** ,Folie 16, Hadoop,
  http://hadoop.apache.org/
- **Abbildung 3.1**, Folie 17, Clustering Beispiel 1
  http://home.deib.polimi.it/matteucc/Clustering/tutorial_html/
- **Abbildung 3.2**, Folie 17, Clustering Beispiel 2
  http://homepages.uni-paderborn.de/pschael/semantisches_clustering/Schael_SemantischesClustering.html
- **Abbildung 3.3**, Folie 17, Clustering Beispiel 3
  http://www.dbs.informatik.uni-muenchen.de/~kailing/Fopras/subspace1.html
- **Abbildung 4** ,Folie 24, Marvin
  https://dev.xing.com/overview