# RecoLeta: A Recommender System for Events for Personalised E-Mail Campaigns

Frank Eichinger, Immanuel Wietreich

CTS Eventim AG & Co. KGaA, Information Science, Hamburg, Germany
{frank.eichinger, immanuel.wietreich}@eventim.de

**Abstract:** We demonstrate the RecoLeta system for event recommendations. It combines two different recommender approaches: one novel approach dedicated to music concert events and one state-of-the-art approach. We also present our big-data architecture for e-mail delivery and recommendation calculation in an in-memory database.

## 1 Introduction

In e-commerce, e-mail marketing is a key tool to attract customers to web pages and to stimulate orders. Apart from a few data-driven companies and early adopters, many companies send out static newsletters, e.g., millions of customers receive the same content. Typically, the content is selected manually by editorial staff and focuses on top-seller products. The reason is that the content should match the interest of a potentially large share of the customers. As this content is popular and frequently well-known, the likelihood that customers buy the respective products is relatively high. The drawback of this is that the mid and long tail of products [LRU14], i.e., the non-top-seller products, will not be advertised and will ultimately be bought even less frequently. Personalised recommendations have the potential to feature the mid and long tail and to recommend only products to customers they are likely to be interested in.

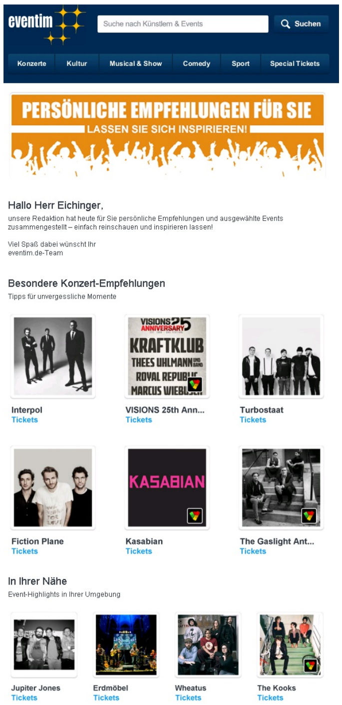


Figure 1: A newsletter containing personalised event recommendations. Contents are dynamically loaded when the e-mail is opened.

Even if there is a huge amount of literature on recommender systems (see, e.g., [RRSK11]) including dedicated recommender systems for music (see, e.g., [KDK11]), there is few literature for the recommendation of events such as music concerts, sport and culture events. Such events are however different than other products like books or CDs, as they have an associated date and location (the recommendation of historic events is not relevant; event venues with a large distance to the users might not always be promising). Furthermore, there are artists doing tours with several associated events which should be taken into account when recommending events to customers.

In this paper, we present the RecoLeta system for event recommendations, which has a focus on music concerts. This system combines a content-based approach [LRU14] (utilising music genre information) for recommending events of artists the users probably like with a state-of-the-art item-to-item collaborative-filtering approach [LSY03]. Furthermore, it incorporates event-specific information such as dates and locations. This approach ensures that customers receive recommendations from music artists which may be new to the customers but still relevant, and that the customers may receive other events than concerts as well. It also covers customers who are not interested in concerts. As input for the recommendations, we use a variety of data ranging from orders to customer demographics. Besides the recommender system, we describe our big-data infrastructure for warehousing, recommendation calculation, campaign management and e-mail (content) delivery.

The RecoLeta system and surrounding infrastructure is a live system which we are currently testing in a continuous-improvement process. Specifically, we send newsletters to millions of real customers (see Figure 1 for an example). First results indicate that dynamic recommendations increase the number of clicks, orders, sold tickets and revenue compared to a static variant with editorial content in mid and long-tail promotions.

## 2 The RecoLeta System for Calculating Event Recommendations

**Content-Based Approach for Music Concerts.** In order to recommend music concerts to customers, we follow a content-based approach which compares music profiles from customers (building on their past interests) to those from artists. This allows us to recommend artists to customers they are likely to be interested in but they might not know so far. Our intention is to inspire the customer and to feature concerts from the mid and long tail.

In order to calculate music similarities, we rely on genre information from the project `www.discogs.com`. This wiki-like web page contains user-generated information for more than five million albums. Each album is assigned with one or more of 15 ‘genres’ and 382 ‘styles’ (sub-genres). Based on this information, we have derived artist profiles. These are vectors where the elements describe the degree to which an artist belongs to a certain music genre. As an example, the artist profile for ‘The Rolling Stones’ is a vector with value 89% for ‘rock’, 4% for ‘blues’, few further fields with small values and many fields with value 0%. Using distance functions [LRU14], we can calculate similarities between two artists. For example, the similarity based on Manhattan distance of ‘The Rolling Stones’ and ‘The Who’ is 81%, while the similarity of ‘The Rolling Stones’ and ‘Helene Fischer’ is 3%. Likewise, we calculate user profiles as vectors containing the aggregated values of all artist profiles (vectors) from artists the customers have showed interest in. Having artist profiles and customer profiles, we calculate raw recommendations as similarities from all customers to all artists we have concert tickets available for. As long as we have an artist profile, this approach does not suffer from any cold-start problems.

**Item-to-Item Collaborative-Filtering Approach.** While the content-based approach allows for recommendations of music concerts, item-to-item collaborative filtering is a general approach for recommendations [LSY03]. This allows for both music and non-music recommendations – for customers having shown interest in music and/or other types of events alike. It calculates a similarity matrix of items. On the input side, the items are events (or tours or artists) customers have bought tickets from. On the output side, the items are events (or tours or artists) for which tickets are available. The cells contain the cosine similarity [DK04, LRU14, LSY03] of both events. Roughly speaking, values are high when the number of customers who have bought tickets from the input and the output side is high. We use all events a customer has been interested in as an input to calculate raw recommendations, using matrix-vector multiplication [DK04]. There are usually no cold-start problems, as frequently many tickets are sold on the fist day they are available.

**Selection of Recommendations for a Customer.** When both approaches have calculated a raw set of recommendations for each customer, we merge these sets using information on music-data availability per customer for steering the weight of each approach. This results in a base set of recommendations for each customer. We then filter this set according to several categories we want to feature in a campaign. The example in Figure 1 contains one category for concert recommendations which have the highest similarity values and one category which contains possibly all kinds of events ordered by similarity and physical distance between the customer's place of residence and the event venue.

## 3 Recommendation and Content Delivery Architecture

Figure 2 describes our big-data architecture for recommendation calculation and content delivery. When a customer orders event tickets in the web shop (1), order information is saved in the web-shop database (2). Using ETL processes, we pseudonymise and copy this data and data from other sources to our data warehouse (3). The warehouse is an EXASolution in-memory column-store database cluster, where we also run all recommendation-calculation processes (4; see Section 2) on a daily basis. We also transfer a subset of the data to our campaign-management system DynaCampaign (6), which writes campaign data back to the warehouse (6). This system manages target and control groups and exports e-mail recipients along with salutations to an e-mail service (8). This service sends out e-mails to the customers (9). These e-mails contain only place holders (identified by '?' in the figure) with underlying personalised links rather than actual content like event recommendations. The customer's e-mail client then follows these links (10) which access our e-mail content delivery system. This system has received pre-calculated recommendations for all users from the warehouse (5) in a pseudonymised way. After a ticket-availability check, it redirects the links to the actual image and link URLs of recommended events (11). The e-mail client then accesses the actual images representing teasers for recommended events from a content delivery network (CDN; 12-13; in the figure, five place holders have already been replaced with images). If the events are interesting to the user, she or he clicks them (14) and reaches the home page where the event tickets can be ordered (1).

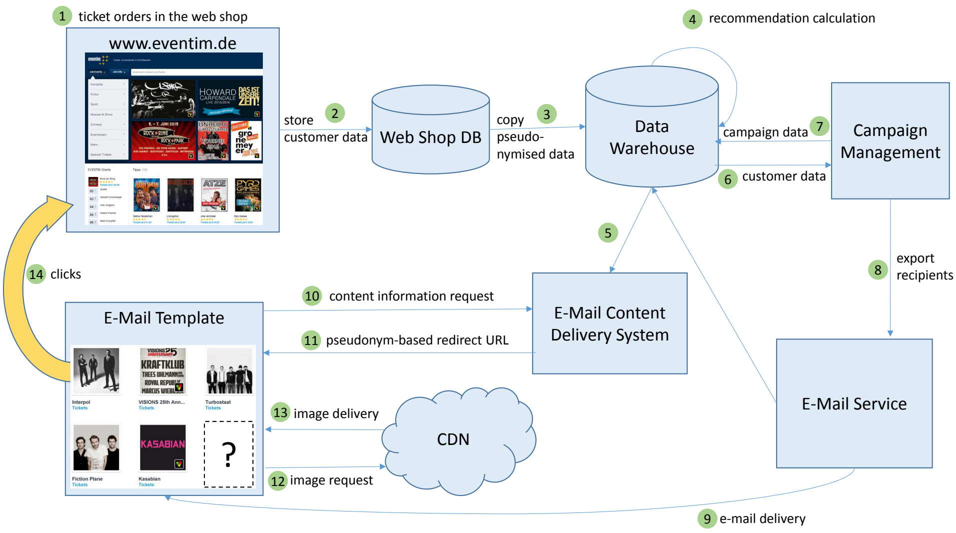


Figure 2: Big-data architecture for recommendation calculation and e-mail delivery.

## 4 Demonstration at the Conference

At the conference, we will demonstrate the RecoLeta recommender system. Conference attendees will have the possibility to receive personalised event recommendations from our live system by entering their preferences into an iPad. As we are currently performing a live test within our direct marketing with millions of real customers over several months, we will also report on the results and experiences of this test at the conference.

## References

[DK04] Mukund Deshpande and George Karypis. Item-based Top-N Recommendation Algorithms. *ACM Transactions on Information Systems*, 22(1):143–177, 2004.

[KDK11] Noam Koenigstein, Gideon Dror, and Yehuda Koren. Yahoo! Music Recommendations: Modeling Music Ratings with Temporal Dynamics and Item Taxonomy. In *Proceedings of the ACM Conference on Recommender Systems (RecSys)*, 2011.

[LRU14] Jure Leskovec, Anand Rajaraman, and Jeffrey D. Ullman. *Mining of Massive Datasets*. Cambridge University Press, 2014.

[LSY03] Greg Linden, Brent Smith, and Jeremy York. amazon.com Recommendations: Item-to-Itemtem Collaborative Filtering. *IEEE Internet Computing*, 7(1):76–80, 2003.

[RRSK11] Francesco Ricci, Lior Rokach, Bracha Shapira, and Paul B. Kantor. *Recommender Systems Handbook*. Springer, 2011.