DIVE into the Event-Based Browsing of Linked Historical Media

Victor de Boer[a,b,*], Johan Oomen[a], Oana Inel[b], Lora Aroyo[b], Elco van Staveren[c], Werner Helmich[d], Dennis de Beurs[d]

[a]*Netherlands Institute for Sound and Vision, Hilversum, the Netherlands*
[b]*Dept. of Computer Science, VU University Amsterdam, the Netherlands*
[c]*Dutch National Library, the Hague, the Netherlands*
[d]*Frontwise, Utrecht, the Netherlands*

Abstract

DIVE is a linked-data digital cultural heritage collection browser. It was developed to provide innovative access to heritage objects from heterogeneous collections, using historical events and narratives as the context for searching, browsing and presenting of individual and group of objects. This paper describes the DIVE Web Demonstrator[1]. We also discuss how the collection metadata the demonstrator uses are enriched through a hybrid workflow. The demonstrator uses semantics from these enriched collections, their vocabularies and linked data vocabularies to establish connections between the collection media objects and the events, people, locations and concepts that are depicted or associated with those objects. The innovative interface combines Web technology and theory of interpretation to allow for browsing this network of data in an intuitive "infinite" fashion. DIVE focuses to support digital humanities scholars in their online explorations and research questions.

Keywords: Digital History, Heterogeneous Data Cloud, Digital Hermeneutics, Historical Events, Crowdsourcing

1. Background

The Web has offered cultural heritage institutions and their public a medium, changing their traditional task from information interpreters to that of information providers [1] and collections are being made digitally available in increasing numbers. Public repositories such as Europeana and the Digital Public Library of America, for instance, offer access to tens of millions of digital artifacts from museums, archives and libraries. This urges cultural heritage institutions to rethink the access provision strategies to their collections to allow the public to interpret and contribute to their collections.

Search and browsing interfaces provide access to both professionals as well as the general public, searching for cultural heritage objects in either a single collection or in multiple collections at the same time. The traditional information access to cultural heritage assumes that experts interpret and curate their collections in such a way that the users of their information systems perform simple or complex keyword search to come to a selection of items matching the query. However, research has shown that many users seek more *exploratory* forms of browsing [2].

Recent technical innovations, which include Linked Data, make it possible to create interactive access to cultural heritage collections not only through direct textual keyword search, but also through structured links between cultural heritage objects and related events, persons, places and concepts. In the Agora project[2], browsing of cultural heritage collections through events and their links to collection object has proved successful for supporting the interpretation of end users, and thus realizing the so-called "digital hermeneutics" [3].

*Corresponding author
Email addresses: `v.de.boer@vu.nl` (Victor de Boer), `joomen@beeldengeluid.nl` (Johan Oomen), `oana.inel@vu.nl` (Oana Inel), `lora.aroyo@vu.nl` (Lora Aroyo), `elco.vanStaveren@kb.nl` (Elco van Staveren), `werner@frontwise.com, dennis@frontwise.com` (Dennis de Beurs)

[2]`http://agora.cs.vu.nl`

2. The DIVE Demonstrator

Building on the event modelling of the Agora project, the DIVE demonstrator presented in this paper implements the event-based browsing of cultural heritage objects from two heterogeneous historical collections. Within DIVE, new interaction concepts for events and event-based narratives have been explored and developed. We explicitly support a diversity of user groups, including Digital Humanities researchers, professional (commercial) users and the general public. The DIVE Web demonstrator is a linked-data digital cultural heritage collection browser. It was developed to provide innovative access to heritage objects from heterogeneous collections, using historical events and narratives as the context for searching, browsing and presentation of individual objects and groups of objects. DIVE provides a novel way to support digital humanities scholars in their online explorations and research questions.

The heterogeneous collections made available through the demonstrator are interlinked in a common linked data network. This interconnected network of events, persons, places and concepts provides context to the cultural heritage objects, which are represented in the same networks. Thus, the objects are contextualized with events and narratives, which is crucial for the findability and hermeneutics. Core contribution is the innovative user interface supporting information interpretation in multimedia collections through dynamic browsing experience with linked data and explicit event representation. DIVE expands on linked media browsers such as NoTube N-screen[3] or HyperTed[4] by supporting exploration of multiple types of linked media and an event-centric approach. This event-based browsing can also be found in tools such as seen.co[5] or Eventify[6].

3. The Collection Data

The DIVE demonstrator allows for browsing of heterogeneous linked datasets as long as they contain media objects (such as images or videos) which are enriched through links with entities such as events, persons and places. currently, content from two cultural heritage institutions are enriched, linked and made available. We here describe the original data and the enrichment process.

3.1. Data Sources

Figure 1 shows examples of the media objects in the two collections included in the current version of the DIVE demonstrator.

- The Netherlands Institute for Sound and Vision (NISV)[7] archives Dutch broadcasting content, including television and radio content. A part of the NISV collection of broadcast video was published as Open Data on the Openimages.eu platform[8]. Within the DIVE project, a subset of this collection of 100 randomly selected videos was ingested through the OAI-PMH protocol[9]. This collection consists mainly of Dutch news broadcasts items from the period. These videos have a typical duration of between a 1 and 10 minutes. For these videos descriptive metadata is available including free-text content description.

- The Dutch National Library (KB)[10] provides access to a number of historical datasets. In the DIVE demonstrator, we use the KB ANP Radio News Bulletin dataset[11]. This dataset is made up of digitized typoscripts (radio news scripts, to be read during news broadcasts) from the period 1937-1984. These have been made public through a Web interface and API. Here, the scanned images, OCRed content and descriptive metadata is available. The original data and metadata are available in Dutch. We ingested 2210 transcripts into the DIVE demonstrator. These were selected from roughly the same period and topics as the NISV dataset to ensure that links between the collections could be established.

3.2. Data Conversion and Enrichment

The textual descriptions and descriptive metadata for both collections are retrieved and are converted to RDF. From the textual descriptions we

[3] `http://nscreen.notu.be/`
[4] `http://linkedtv.eurecom.fr/Hyperted/`
[5] `http://seen.co`
[6] `https://eventify.it/`
[7] `http://www.beeldengeluid.nl`
[8] `http://openimages.eu`
[9] We are currently expanding this small subset
[10] `http://www.kb.nl`
[11] `http://radiobulletins.delpher.nl/`

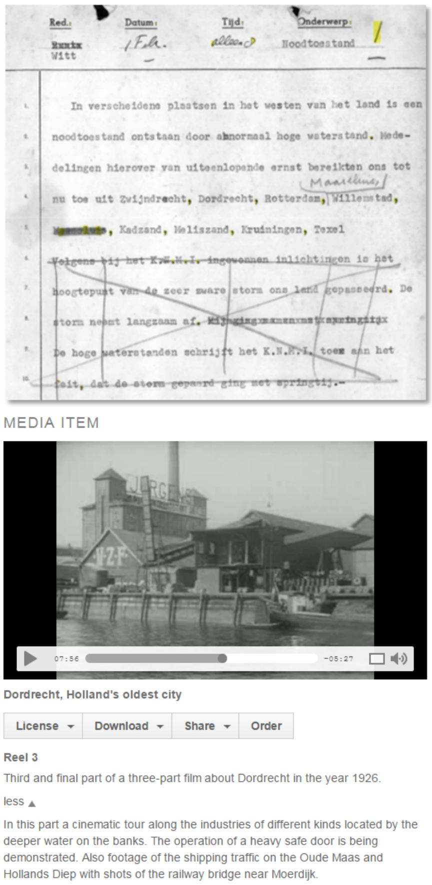

Figure 1: Examples of the media objects from the two collections: the top shows one typoscript from the KB dataset. Not only the textual content but also the annotations are of interest. The second image shows a video item from the OpenImages collection of NISV with the descriptive metadata. The items share a mention of a geographical location (the city of Dordrecht).

extract *events* as well as places, persons and concepts linked to those events which in turn are depicted by the cultural heritage objects (videos or news bulletins). For this extraction, we employ an ensemble of methods. The DIVE system incorporates a hybrid workflow for event enrichment in video collections. In this hybrid workflow the machines and the crowd collaborate in the process of extracting relevant events and event-related concepts in video collections. In the first stage Named Entity Recognition (NER) and Event extraction tools for Dutch text are used in order to retrieve a set of relevant concepts from the video descriptions. In a second stage, crowdsourcing through the CrowdTruth platform[12] is employed to have human-recognized entities and to refine the results from Natural Language Processing. Section 4 presents in detail this workflow. For the extraction of the News Bulletins, we also use the results of the NER employed by the KB, which is based on the Stanford parser, optimized for Dutch texts.

The results from the different tools and the crowdsourcing are consolidated to RDF. The data is modeled using the Simple Event Model (SEM) [4]. This model allows for the representation of events, actors, locations and temporal descriptions. We extend SEM with other Linked Data schemas, e.g. DC, SKOS, OpenAnnotation and FOAF to represent other types of resources linked to the media objects[13]. Links are established between the different datasets using the Amalgame alignment tool [5]. Amalgame allows for interactive alignment of terms, using a variety of filtering and (exact and fuzzy) text matching algorithms. In this case, we employed exact matching with minimal preprocessing of text labels. Amalgame allows for the output of different types of relations (including for example `skos:exactMatch` or `skos:closeMatch`), in this case, we opted to include the found correspondences as `owl:sameAs` relations[14]. Links are also established to external sources, including Wikipedia and DBpedia. The resulting dataset is stored in an RDF Triple store, which provides a SPARQL endpoint[15].

Figure 2 shows the general setup of the project, the data ingestion and enrichment pipeline as well as the interface layers.

4. Hybrid Event Enrichment Workflow

This section presents the hybrid workflow (Figure 3) used for enriching the video collection with events and event-related concepts. The workflow takes advantage of both machines output and crowd judgments in order to support collection interpretation and navigation through event-based research and exploration. The central idea of this approach is harnessing the disagreement among machine and crowd annotators in order to optimize the extraction of events and their role fillers, mainly from

[12]`http://crowdtruth.org/`

[13]The DIVE datamodel is visualized at `https://github.com/biktorrr/dive/blob/master/imgs/dive_model_v3.pdf`

[14]For more information, we refer the reader to the Amalgame website at `http://semanticweb.cs.vu.nl/amalgame/`

[15]The ClioPatria triple store can be accessed at `http://semanticweb.cs.vu.nl/dive/`

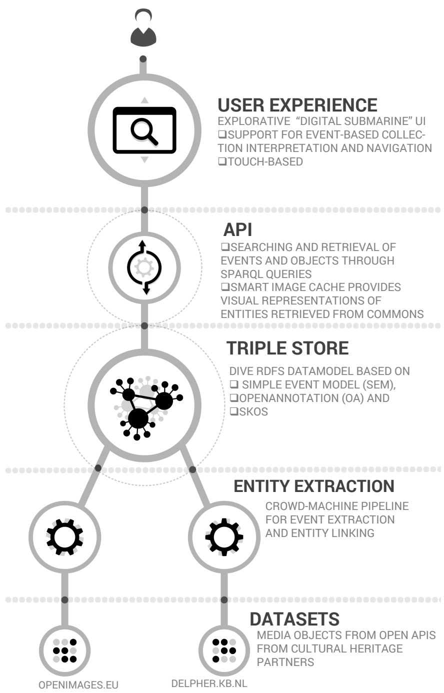

Figure 2: An overview of the DIVE project and demonstrator: cultural heritage content is collected and enriched through the event generators. The resulting graph representation is stored in a triple store, which is queried by the Web interface layers.

video descriptions. The entire workflow was developed on the CrowdTruth platform.

4.1. Machine Annotation

Machine annotation is applied at the level of both video content and video description. The video pre-processing consists of keyframes extraction and video segmentation through the open source FFmpeg[16] multimedia framework. The aim of these transformations is to adjust the media item in such a way that it captures enough event-related features, while keeping a suitable amount of details to visualize these features. The keyframes are main video components that intend to depict single various scenes, people, groups, locations, while the segments cover a broader range of event-related aspects. Thus, the aim of the keyframes and video segments is to guide and support the user experience in the event-based browsing.

In order to detect depicted or associated video concepts, we process the video description. We apply a set of six NLP tools to identify the named entities (NE) present in the video description: NERD[17], TextRazor[18], THD[19], DBpediaSpotlight[20], Lupedia[21] and SemiTags[22]. Even though some of these extractors are already incorporated in the NERD framework, we observed that when NERD is used with the so-called *combined* strategy (*i.e.*, it enables to launch all extractors with a conflict resolution) it might fail in recognizing the correct entity type. Most of the misclassifications appear between Location and Organization types. Each method receives a description as input and returns for each recognized named entity the following: (i) entity name; (ii) start offset; (iii) end offset; (iv) entity types; (v) entity resource.

The state-of-the-art NE extractors (for a review see [6]) provide good results, but it is extremely difficult to find one extractor that performs well on heterogeneous topics such as the TV-news broadcasts from NISV. On the one hand, each extractor uses different algorithms and training data and is targeted on recognizing only some specific named entities. On the other hand, other extractors are being more reliable on different document types such as newspapers or medical articles [7]. The idea of aggregating multiple machine extractors output has been developed in [8, 9] as a majority vote strategy. However, we build our approach on the disagreement notion. By evaluating the disagreement among multiple extractors we can potentially improve our results as an extractor can return: (i) missed entities by other extractors; (ii) alternative entity spans, where a span consists of a contiguous segment of the video description that is being analyzed; (iii) missed or misclassified entity types and resources by other extractors. Furthermore, we do not remove entities with low confidence or relevance scores because these values are most of the time computed by black-box metrics.

In order to harness the disagreement among the machine annotators, we take each entity that was extracted by one or multiple machine extractors and group it based on its DBpedia class in the following clusters: Locations, People, Time and Other Concepts. In the case where we have overlapping

[16]https://www.ffmpeg.org/

[17]http://nerd.eurecom.fr/
[18]https://www.textrazor.com/
[19]https://ner.vse.cz/thd/
[20]http://dbpedia-spotlight.github.io/demo/
[21]http://www.old.ontotext.com/lupedia
[22]http://ner.vse.cz/SemiTags/app/index

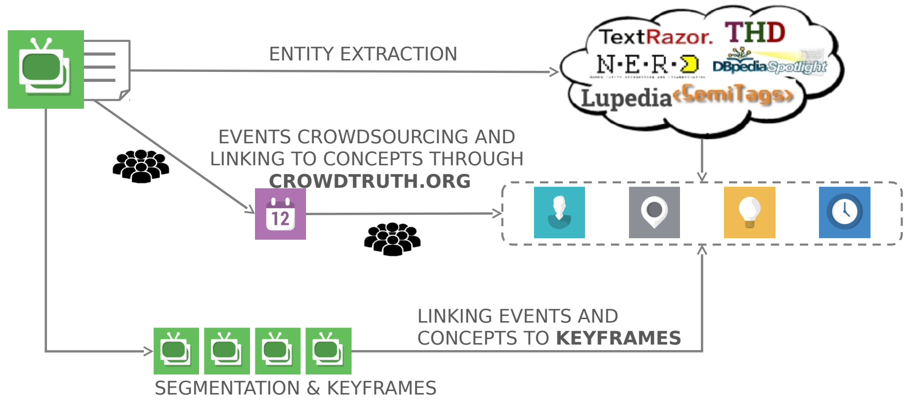

Figure 3: Hybrid machine-crowd event enrichment workflow

entities spans, *i.e.*, multiple spans for the same entity such as "D. van Staveren" and "Staveren" we do not discard any alternative, but keep all of them. For entities where the extractors did not return a type, but they did return a resource URI (either from DBpedia or Wikipedia) we perform various queries to DBpedia in order to retrieve the entity class. Thus, by harnessing the disagreement between extractors, we reject the notion of majority vote and consider that any information from any extractor can be right, even if it was a singular choice. Further, the resulted clusters are used in a series of crowdsourcing experiments aiming to link the events with their role fillers (Participants, Location, Time and other Concepts). Moreover, we extract the timestamps mentioned in the video descriptions and use the keyframes close to these timestamps in order to link the aforementioned concepts with visual parts of the video. Keyframes that are extracted around these timestamps have a high probability of depicting the concepts described in that part of the description. This is mainly desired for the explorative browsing of the collection.

4.2. Crowdsourcing

A crowdsourcing task consists of a unit of work, small and clear enough for crowd workers to provide annotations. The crowdsourcing experiments have as input the integrated output of the machine processing, the clusters, as well as the video description. Their aim is to use human computing in order to resolve the issues that machines are still having problems to clarify, *e.g.* *(i)* extracting events, *(ii)* linking events with their role fillers. Therefore, the first crowdsourcing task aims to extract all the existing events from the video descriptions. Second, we focus on linking the extracted events with their role fillers - participants, location and time concepts.

As mentioned above, the first task (Fig. 4) focuses on determining the events depicted in the video descriptions. Thus, the input received by the crowd contains the text of a video description. The crowd is asked to highlight all the word phrases that could potentially refer to an event. The crowd can verify the selections made and remove the ones that were mistakenly highlighted. Given this open template, we aim at gathering multiple event-granularities, which would further improve the event interpretation. We consider event-granularities the different span alternatives for an event such as "toespraak" or "toespraak tot genodigden". In the current approach we consider events the longest spans annotated by the crowd workers (*i.e.* series of consecutive words that have a high score of being events from the crowd workers).

The second task (Fig. 5) focuses on linking the events from the previous step with the entities classified as participants, location and time from the machines output. The input received by the crowd consists of two text boxes with the video description. The first box contains highlighted events (resulted from the previous crowdsourcing task), while the second box contains highlighted Location/Participants/Time concepts. The crowd is asked to create links between the highlighted events and concepts by clicking the concepts of both text boxes. A link can be removed by clicking on it.

Since at this level we deal with human annota-

Figure 4: Crowdsourcing Template for Event Extraction

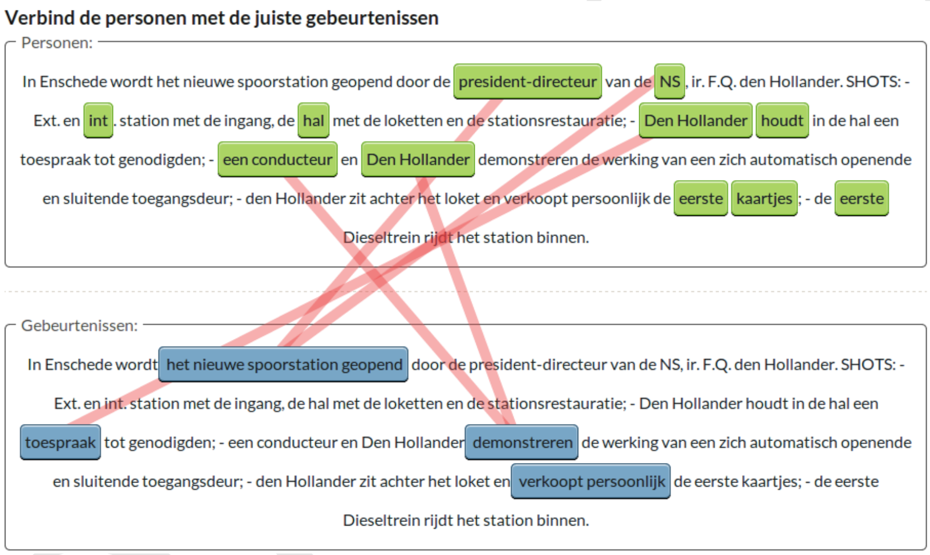

Figure 5: Crowdsourcing Template for Linking Events with Event-related Concepts

tors, it is extremely important to deal with the answers quality. As human annotation is a process of semantic interpretation, often described by the triangle of reference [10], we have since observed that disagreement signals low quality and can be measured at any corner of this triangle. This is the main concept incorporated in CrowdTruth framework and presented through the CrowdTruth metrics [11]. Since the metrics have been presented multiple times [12], we describe here only the adaptation of the annotation vector that is the main concept behind the disagreement metrics.

The result of each worker for the event highlighting task is stored in a vector having the length equal to the total number of words in the video description. Thus, in each worker annotation vector we mark all the highlighted words. The annotation vector for each video description represents the aggregation of all the worker vectors. For the second task, we have a set of events E_i and a set of possible related concepts C_j. The annotation vector contains all the possible combinations E_i-C_j.

For both crowdsourcing tasks we have used the following settings: *(i)* each video description was annotated by 15 workers; *(ii)* each worker was allowed to perform only a maximum of 10 annotations; *(iii)* only the workers from Dutch-speaking countries were allowed to provide annotations. For evaluating and validating the correctness of the CrowdTruth metrics, *i.e.* the spam workers annotations were correctly removed, three people performed manual evaluation for 100 workers, equally distributed between spammers and non-spammers. The performance score given by the evaluators (1-spam, 0-non-spam, 0.5-unclear) was used to compute the precision, accuracy, recall and F1 score. The outcome of the evaluation proved that the spam workers were correctly identified given the high F1 score and accuracy of 0.9, precision of 0.87 and recall of 0.94.

5. User Experience

The DIVE demonstrator is an end-user application, openly available on the Web. The demonstrator's UI is designed for two user groups. The primary user group is that of cultural heritage domain researchers, to browse the integrated collections in an explorative fashion. However, the demonstrator is set up in an intuitive way, allowing the general public to navigate the integrated graph in without the need of academic prior knowledge. It uses reasoning over the graph to display related items to the one that the user is focusing on. Specifically, properties indicating specific relations between entities are modeled as sub-properties of a more generic "is related to" property. Similarly, specific entity types are modeled as sub-classes of a more generic entity class. These generic properties and classes are used in the queries that the UI employs. RDFS reasoning ensures that the specific related entities are retrieved. Furthermore, transitivity of relations is used to display entities that are related indirectly and in the interface module, a number of relation patterns are defined that determine which (directly or indirectly related) entities are shown. These patterns are implemented in the SPARQL queries used by the interface module. This goes well beyond standard information retrieval. The event-related filters allow to identify groups of objects depicting or associated to specific events, related people, locations and times. Additionally, we project all browsing results on an interactive timeline, providing alternative filtering and access to the related objects.

5.1. User Experience Design

5.1.1. Design principles

The core of digital hermeneutics is formed by two components: *object-event relationships* and *event-event relationships*. By making explicit relationships between the objects and events as well as between the events themselves we can facilitate users in their processes of accessing and interpreting objects in online cultural heritage collections. In DIVE we aim at implementing those relations in an intuitive event-centric browsing interface for browsing cultural heritage collections by means of underlying linked data graph.

Considerable effort was put in creating an interface with a clear identity and an engaging user experience that invite users to continue exploring at different levels of detail. Users become explorers diving deeper into the data, like a diver "deeper and deeper into an ocean trench discovering new species". This metaphor makes the interface a "digital submarine", which provides navigation controls as well as supportive and manipulative tools. The design of the interface also forms an innovative "infinite exploration path", which unlocks the potential of touch-based explorative user interfaces.

5.1.2. Design process

The demonstrator's User Interface is the result of a careful design process in which web and interaction design experts, data experts, humanities researchers and end users collaborated. It is both visually attractive as well as functional and makes use of the visual nature of much of the media objects as well as the extracted knowledge. By making explicit relationships between the collection objects and events, and their related properties, e.g. people involved, locations and times we can facilitate users in their access and interpretation processes of objects in online cultural heritage collections.

The extensive design phase in which multiple concepts have been tested resulted in the DIVE "infinity browsing" interface, a combination of two core interaction concepts that involve a *horizontal level* supporting the breadth and a *vertical level* supporting the depth of information exploration and interpretation.

5.1.3. Horizontal browsing

The *horizontal level* displays the result set of objects related to the seed keyword search in a dynamic presentation. At this level, user's exploration is supported by *event-centric filters* making explicit the relation of each object to either the depicted and associate events and their properties, e.g. people, locations and times involved in the events. Consistent *color coding* is used for each property type to allow for a quick discovery of desired type of objects. Large result sets are represented as a colored barcode as an overview of the amount and composition of event properties in the search result. *Objects* are represented by type-color, type-icon, title and an image and associated with a set of *buttons* providing detailed information for each object, e.g. description, source and external links. To allow for active user engagement and sharing of personal perspectives and interpretations, users can add *comments* to each object, share entities and save them in private or shared *collections*. Additionally, we provide a set of related objects from the Europeana collections[23].

Typical interactions at this level are:

- *Pinch or scroll the elements of the color barcode* zooms in on the objects to reveal more information, e.g. image, title, icon of the event-related property (for example, an icon for location indicates that this object depicts a location of the related event).
- *Drag right or left the row of related objects* reveals previous or next object on the horizontal level.
- *Arrows* are used for navigating to previous or next objects in the row.
- *Search option & event filters* allow to show subsets of related objects.

5.1.4. Vertical browsing

The *vertical level* is formed by the user exploration history, as a path of selections on the horizontal level - leading to the last selected object. Each selection of an object results in a new row with related entities loaded under the selected object. Users can scroll back to a previous step, zoom out, choose another object and build a new path from there. This allows for fluid dynamics in collection exploration, discovery of alternative paths, and ultimately supports deep interpretation of cultural heritage collections. Current work includes adding the option to save exploration histories as a collection so that users can revisit or share their browsing experiences.

5.2. Implementation Details

The interface is developed in an iterative integrated process of design and development. It's main building blocks are made in PHP and Symfony2 on the server side, and HTML5, Javascript and CSS3 on the client-side. A number of libraries are used to provide specific functionality: jQuery handles the major part of the functionalities like DOM interaction and manipulation, event handling and AJAX. Velocity.js is used to improve the performance of animations. Hammer.js supports the handling of touch events. Moment.js makes dates manageable.

[23] `http://europeana.eu/`

5.2.1. Optimization and Speed

As the handling and displaying of large amounts of information can be near the feasible limits of the web browsers, optimizations have been made to keep the interface fast and responsive. On the server, query results are cached and served with the correct cache headers by the API. Data is served with the correct cache headers and gzip compression. On the interface side the queuing and serial processing of data calls contributes to fast first responses and gradual loading. Queued calls can be removed before being triggered, in case user interaction (e.g. selecting a different row) makes them obsolete. Other examples of interface optimizations include the gradual buildup of DOM elements[24] and lazyloading of images, stepwise loading and publishing of deeper relation data and eventually limiting visual effects and animations for large collections. When loading the related data, many synchronous calls are made to the API[25]. the On programming level the use of Prototypes improves object creation speed for large amounts of entity and row objects. These efforts results in a smooth and fast browsing experience on the desktop, and makes the interface accessible by tablets.

The interface acquires data from the data layer using the triple store's SPARQL API. Several queries are used to search entities by keyword, get related entities (events, persons, etc.) and get entity details. Key focus of the queries is to return the right data, within an acceptable time frame. `owl:sameAs` relations are included in the query to include entities that are not yet combined is a superclass. With the current dataset/server combination, typical query speed measures about 2000ms

[24] Experiments showed that adding DOM elements only when required reduces the average DOM/class build time for an entity from over 10ms to around 4ms.

[25] By saving/exiting a session, the session lock is released and multiple calls can be handled much faster by the server. This lead to a significant reduction of response times. For example, when retrieving indirectly related entities for "Rotterdam", 1926 entities are found, with the session exit strategy, the response time reduces from around 14000ms to 4500ms.

for the keywords based text search query, 300ms for entity details, and between 200ms and 2000ms for related entities[26]. The last range is heavily related to the amount of related entities. The returning data is handled by an adapter that maps the datafields to an internal format which is used to build the interface representations. This approach relieves the clients of unnecessary data parsing and contributes to compatibility with other data sets.

5.2.2. Smart Image Cache

To provide a visual representation for other entities, the DIVE frontend includes a "smart image cache". This module provides an image lookup based on keywords associated with a DIVE entity and also caches the response because of the potentially long lookup times involved. We consider this cache to be "smart" in the sense that internally it is implemented as calling a specific image file, but the module does a distributed lookup in a number of Open image APIs . Specifically, images are retrieved from the five most relevant Wikipedia searches using the Wikipedia API[27]. If no images are found, another query is made to the OpenCultuurData API[28] which covers an extensive set of Dutch open heritage- and cultural data. This latter API was very useful for our domain specifically as it provided a lot of news related images that match the old videos and radio news bulletins in the DIVE data set. The quality and availability of images through this system is acceptable and provides a powerful way of filling in the (visual) data gaps. These images increase the user experience by supporting the visual navigation through the interface and rememberability and recognizability of individual entities.

5.2.3. Tablet-first

Figure 6 shows the current version of the interface, optimized for tablets and modern web browsers. The demonstrator UI is designed specifically for multiple platforms and features a responsive design. In fact, the UI was designed with a "tablet-first" principle, where touch-actions and gestures are primarily considered for the browsing of the graph. Among the design features specifically made for touch-screen devices are 1) *large touch areas* ; 2) *pinch and swipe navigation* for resizing the result set or zooming in on the specific media object and browsing horizontally and vertically through the graph; 3) *speed optimizations* such as asynchronous loading, code optimizations, limited effects to keep performance up on (generally slower) touch devices.

6. Current Status and Future Work

6.1. Current Status

The DIVE demonstrator has been published at `http://dive.beeldengeluid.nl`. We are continually updating both the data as well as the user interface. Currently, a subset of data from the two institutions is available in the datastore and through the interface. We are scaling this up to more data from the two institutions as it becomes available, as well as (Dutch) historical data from other sources.

6.2. Current Work

In the context of the DIVE+ project[29], we are extending the DIVE demonstrator in four ways. First, we extend the DIVE data with two new collections (i.e. Amsterdam Museum[30] and Tropenmuseum[31]) as well as links to external linked open datasets, such as DBpedia, AAT and ULAN.

Second, we aim at extending the user interface with an intuitive way to deal with event narratives.

Third, we extend the crowdsourcing pipeline discussed in Section 4 for historical events extraction to also image annotation. Through extensions on the CrowdTruth platform, we will be able to deal with diversity of interpretation and perspectives through a joint pipeline for machine and human processing of media and provides a deep analysis of the quality of the crowdsourced data. It will also contribute to mapping various "narratives" and identification of "perspectives" associated with them, to advance research on data representation, structuring and categorizations.

Finally, we propose a set of user studies to test the usability of the approach with a variety of end

[26] Live examples of these three types of queries are maintained at `http://tinyurl.com/dive-query1`, `http://tinyurl.com/dive-query2` and `http://tinyurl.com/dive-query3` respectively

[27] `http://www.mediawiki.org/wiki/API:Main_page`

[28] `http://www.opencultuurdata.nl/api/`

[29] DIVE+ is part of the Netherlands eScience Center program for enabling scientific discovery for digital humanities (`http://www.esciencecenter.nl/`).

[30] `http://www.amsterdammuseum.nl`

[31] `http://tropenmuseum.nl`

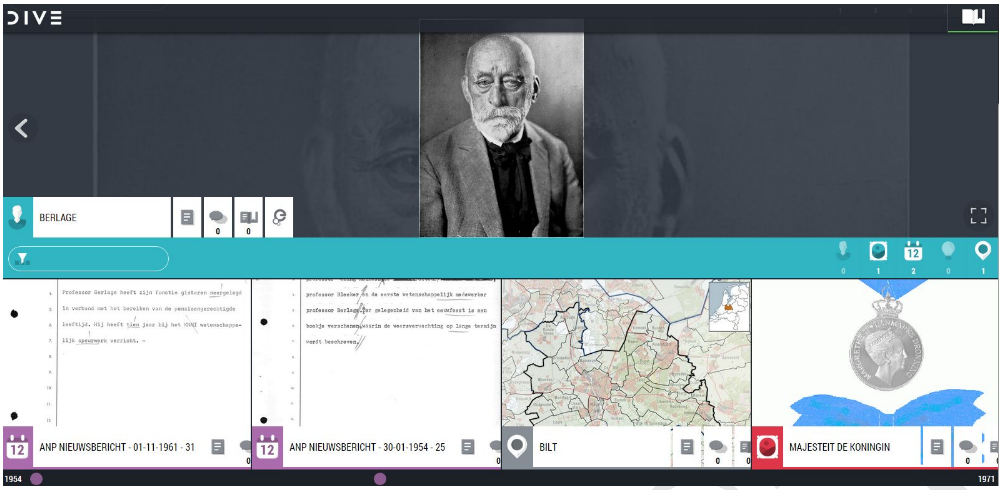

Figure 6: Screenshot of the interface. This page shows a person (Berlage), with two related events and KB media objects associated with them, as well as a related place and person. In the bottom, a timeline is also shown.

users. This will include Information Retrieval-type user evaluations, with a specific aim at measuring the success of explorative search. In this way we will be able to gain insights on the scalability, robustness and reusability of the DIVE digital hermeneutics approach, as well as its usability for history researchers and general audience. We will also log usage by members of the general public as the demonstrator will be made available as part of the websites of NISV and KB.

The aim is therefore to advance the way in which history researchers and general audience interact with digital heritage collections.

Acknowledgements

This work is partially supported by NWO CATCH program (http://www.nwo.nl/catch) and the Dutch Ministry of Culture. The authors would like to thank Theo van Veen and Olaf Janssen for their help with the KB data.

References

[1] K. Mueller, Museums and virtuality, in: In R. Parry, editor, Museums in a Digital Age, chapter 30, pages 295-305 Routledge, 2007.

[2] M. v. Erp, J. Oomen, R. Segers, C. v. d. Akker, L. Aroyo, G. Jacobs, S. Legêne, L. v. d. Meij, J. v. Ossenbruggen, G. Schreiber, Automatic heritage metadata enrichment with historic events, in: Museums and the Web 2011, 2011.

[3] C. van den Akker, S. Legêne, M. van Erp, L. Aroyo, R. Segers, L. van der Meij, J. van Ossenbruggen, G. Schreiber, B. Wielinga, J. Oomen, , G. Jacobs, Digital hermeneutics: Agora and the online understanding of cultural heritage, in: Proc. of the 3rd International Web Science Conference (WebSci '11). ACM, New York, NY, USA, 7 pages., 2011.

[4] W. R. van Hage, V. Malaisé, R. H. Segers, L. Hollink, G. Schreiber, Design and use of the simple event model (sem), Web Semantics: Science, Services and Agents on the World Wide Web 9 (2).
URL http://www.websemanticsjournal.org/index.php/ps/article/view/190

[5] J. van Ossenbruggen, M. Hildebrand, V. de Boer, Interactive vocabulary alignment, in: Research and Advanced Technology for Digital Libraries - International Conference on Theory and Practice of Digital Libraries, TPDL 2011, Berlin, Germany, September 26-28, 2011. Proceedings, 2011, pp. 296–307. doi:10.1007/978-3-642-24469-8_31.

[6] A. Gangemi, A comparison of knowledge extraction tools for the semantic web, in: The Semantic Web: Semantics and Big Data, Springer, 2013, pp. 351–366.

[7] G. Rizzo, M. van Erp, R. Troncy, Benchmarking the extraction and disambiguation of named entities on the semantic web, in: 9th International Conference on Language Resources and Evaluation (LREC'14), 2014.

[8] L. Chen, S. Ortona, G. Orsi, M. Benedikt, Aggregating semantic annotators, Proceedings of the VLDB Endowment 6 (13) (2013) 1486–1497.

[9] S. Hellmann, J. Lehmann, S. Auer, M. Brümmer, Integrating nlp using linked data, in: The Semantic Web–ISWC 2013, Springer, 2013, pp. 98–113.

[10] L. Aroyo, C. Welty, Truth is a lie: 7 myths about human annotation, AI Magazine, 2015.

[11] L. Aroyo, C. Welty, The Three Sides of CrowdTruth, Journal of Human Computation 1 (2014) 31–34.

[12] O. Inel, et al., Crowdtruth: Machine-human computation framework for harnessing disagreement in gathering annotated data, in: The Semantic Web–ISWC 2014, Springer, 2014, pp. 486–504.