DiscourseDB

This document refers to the DiscourseDB draft scheme v0.1. The scheme is still at an early design stage and has a particular focus on the overall macro structure. Attributes in the ER diagram are restricted to what is necessary for representing this macro structure. Most of the content attributes are still missing in the ER diagram.

## Conceptual Overview

DiscourseDB represents online discussions from different sources (forums, chats, instant messaging, etc.) in a unified format that allows researchers to perform discourse analyses across sources without having to take the specific properties of each particular source into account.

DiscourseDB represents both the “external discourse structure” (relationships between contributions) as well as the “inner structure” of the contributions (linguistic properties and internal organization of each contribution).

The external structure is represented in a relational database model while the inner structure is modeled with the Unstructured Information Management Architecture (UIMA). To close the gap between the two, UIMA’s Common Analysis Structure (CAS) can be represented in a relational database. This allows to make queries over macro and micro structures.

## Walkthrough

### External Discourse Structure Layer

Discourse entities are containers that represent single discourse units. Depending on the source, these units can be cats, whole forums or Talk pages with multiple discussions.

Discourse entities consist of one or more DiscoursePart entities that can be assigned with a type (e.g. Forum, Subforum, Thread) and can be embedded in each other to form nested structures.

DiscoursePart entities contain Contribution entities which represent the smallest unit of analysis in the external/macro layer of DiscourseDB.

Contribution entities are related to each other by DiscourseRelations, which can be of any arbitrary type (e.g. explicit reply, retweet, implicit reply). Contribution entities are logical units without actual content (i.e. text). They are associated with one or more Revision entities that capture the changes of each contribution over time.

Relationships between entities might change over time. Thus, each relation has a lifetime (startTime/endTime) that exactly defines the time in which the relation was valid. By keeping track of historic relations, it is possible to reconstruct structures at earlier points in time.

### 

### User Layer

User entities represent actors in the discourse. A user can be the author of a Revision entitiy or part of an audience. Furthermore, User entities can be assigned to groups. Group entities represent teams or user groups in the original data source. Currently, User entities are only explicitly associated with Group entities with a member\_of relation. This could be made more generic if necessary.

Audience entities represent the implicit or explicit recipients of contributions in the discourse context. It can be assigned with arbitrary types (e.g. public web audience) and also be associated with Group and User entities.

### UIMA-SQL Layer

The UIMA-SQL layer is based on a framework by Fette, Toepfer and Puppe (2013)[[1]](#footnote-0). UIMA-SQL allows to store UIMA CAS objects in a relational database rather than serializing them to XMI.

In DiscourseDB, a UIMA Collection is represented by a Discourse Entity in the external discourse structure, while a CASDocument is represented by a Revision entity. The tables CASDocument and Revision as well as Collection and Discourse could be merged, since they represent similar concepts.

## Use case “Wikipedia”

Each Wikipedia Talk Page is represented as a Discourse entity. A Wikipedia Discourse has a flat hierarchy of DiscoursePart entities which represent discussions on the page. These parts are not further embedded. A possible DiscoursePart\_Type could have be “Wikipedia Talk Page Discussion”.

Contribution entities represent “turns” as defined by Ferschke (2014).[[2]](#footnote-1) Informally, turns can be thought of as contiguous spans of text created in an “edit session” by a single user. Revisions of this turn are accounted for by the Revision entities that allow to maintain different versions of a turn. It is possible that a turn can be edited by a user who is different from the original author of the turn.

There are multiple DiscourseRelations that can be assigned to pairs of contributions. One type might be a reply relation indicated by indentation. Another relation might be an adjacency relation. This is particularly important, because the sequence of contributions does not necessarily follow a temporal ordering. Contributions on Wikipedia pages are often reorganized over time (e.g. by insertion of messages in the middle of an existing thread). Adjacency relations capture the order of contributions at any given point in time and allow to recover the state of a discussion at earlier points in time.

Contributions in Wikipedia can have a rich and complex internal structure. This structure can be represented on the UIMA-SQL layer by introducing appropriate UIMA-types to represent and annotate the inner structure.

## Use case “Threaded Discussion Forum”

A threaded discussion forum might have multiple nested sub-forums. On the lowest level of this hierarchy, the (sub-)forum contains discussion threads consisting of nested posts.

A Discourse represents the forum as a whole with multiple DiscourseParts each representing the sub-forums. The Embedding relations between the DiscourseParts capture the nested structure of the sub-forums. Discussion threads are represented by Contribution entities and their relations to each other. Each contribution is part of a particular sub-forum (i.e. has a partOf relationship witht the particular DiscoursePart). Furthermore, the explicit reply structure of a thread is represented by DiscourseRelation entities of the “ExplicitReply” type. Since explicit replies in threaded discussions do not necessarily correspond to the actual reply structure from a discourse theoretic point of few, a second DiscourseRelation with an “ImplicitReply” type could be introduced to represent the reply structure on another level based on manual or automatic annotation.

1. http://ceur-ws.org/Vol-1038/paper\_1.pdf [↑](#footnote-ref-0)
2. http://tuprints.ulb.tu-darmstadt.de/4092/ [↑](#footnote-ref-1)