

# TROVA

This A4 guide is aimed at explaining the main functions of TROVA, an online tool designed for searching (multiple) annotation files.

## Requirements:

- A recent Java runtime environment (at least Java 1.5)
- A working Internet connection
- The IMDI browser at [http://corpus1.mpi.nl/ds/imdi\\_browser/](http://corpus1.mpi.nl/ds/imdi_browser/)

## How to use Trova:

- To start the application right click the node you want to search and select **annotation content search** from the pop-up menu. Trova interface will open, and you will see three types of search:
- **Simple** search – with this type you do not have constraints (apart from the optional ones regarding the results: how to display them, their context size and their font). You just have to type your query in the white search string and click on **Find**. The results will be shown in the bottom half of the page (see figure 1 on the next page).
- **Single Layer** search – this type of search is a bit more complicated than the previous one, since it includes more constraints. The way of making the query is the same as before (typing in the white search string > 'Find'), and so are the the constraints related to the results layout ('Action', 'Context size', 'Font'). Among the new constraints, the most important ones are the following:
  - Additional tier: this option allows you to search for annotations in two tiers at the same time. One of the tiers will be shown in the results (bottom half of the page), whereas the other one will appear in a balloon when hovering over the results. For both of them you have to choose the type (or the name, or the participant, or the annotator), and for the extra one only you also have to decide whether you want it aligned to, or overlapping with, the first one. The two can have either a sibling relationship, or a parent/child relationship.
  - Match: your search string can match either perfectly with the annotation found (**exact match** option), or partially (**substring match** option). Besides these two possibilities, you can also use a regular expression as search string.
- **Multiple Layer** search – this one is the most complex type of search, and the two main reasons for this are that a) you can search several annotations within the same tier, but also several annotations within three tiers at the same time; b) there are many constraints to set. The elements in common with the previous two types of search are the white search strings, the selection of tier type/name/participant/annotator, and the types of query-results match. What is basically new, by comparison, is the following:
  - Time: you can set time constraints to your query, namely a minimal and maximal duration of the annotation(s), and/or a begin and end time.
  - Green fields: between the white search strings you can see green fields. These are drop-down menus from which you select further constraints: either the number of annotations you want between your queries, or the way they have to be aligned.

# TROVA

Simple **Single Layer** Multiple Layer

Types:  EAF (408)  Shoebox (82)  Text (16)  XML (2)  HTML (15)  PDF (341)

Domain: DoBeS archive

Find:

**Ready** Found 72172 hits in 50791 annotations

Action: **Show Concordance View** Page: < > Hit 1 - 15 of 72172 hits

Context Size: 4 Font: **Arial Unicode MS** 12 Show Info Balloons

*near:SPKR that.DIST old -3SG.POSS and then near:SPKR new -3SG.POSS funny.story*  
*go do.thingy Island.Name loan.wor and then OBLIGAT 3SG.SUBJ go Island.Name*  
*OBLIGAT 3SG.SUBJ go Island.Name and then person all.of -3SG.POSS 3PL.SUBJ*  
*go.down OBLIGAT 3PL.SUBJ boaru and then 3SG morning ANA 3SG.SUBJ*  
*NEG 3SG.SUBJ carry -TR and then lamp 3SG.SUBJ carry -TR*  
*lamp 3SG.SUBJ carry -TR and then 3SG.SUBJ go.down -to.SPKR 3PL.SUBJ*  
*OBLIGAT keroseen 1SG.SUBJ touch/make and then 1SG.SUBJ DUP-go.up loan.word*  
*3PL.SUBJ pour CAUS- full and then 3PL.SUBJ talk go 3SG.SUBJ*  
*-1SG.POSS loan.word 1SG.SUBJ get/take and then mind/heart -1SG.POSS 3SG.SUBJ forget*

*he wanted to go to Samarai. It was Saturday and he wanted to go to Samarai and all the people went down to board. And as for him, when he got up in the morning all his thoughts were that he wanted to go and sell his goods.*  
*when he thought: 'Eh, I have to get kerosene.' And so he run up (home again). And then he came down but didn't carry the container but carried the lamp. They said: 'That's a lamp!', and he said: 'Eh, I have to get kerosene.'*  
*And so he run up (home again). And then he came down but didn't carry the container but carried the lamp. They said: 'That's a lamp!', and he said: 'Eh, I wanted to get kerosene.'*  
*And so he run up (home again). And then he came down but didn't carry the container but carried the lamp. They said: 'That's a lamp!', and he said: 'Eh, I wanted to get kerosene and I went up to get the cont.*

Figure 1. Simple search

Simple **Single Layer** Multiple Layer

Types:  EAF (408)  Shoebox (82)  Text (16)  XML (2)  HTML (15)  PDF (341)

Domain: DoBeS archive

History: **the Annotation case insensitive substring match in All Tiers**

Mode: **Annotation** case insensitive substring match (Use # as word wildcard in n-gram modes)

Extra: Fully aligned, any tier annotation from All Tiers [4286] in mode 'Annotation + extra tier' info balloons

Find:  in **All Tiers [4286]** tier choice sort

**Ready** Found 72172 hits in 50791 annotations

Action: **Show Concordance View** Page: < > Hit 1 - 15 of 72172 hits

Context Size: 4 Font: **Arial Unicode MS** 12 Show Info Balloons

*near:SPKR that.DIST old -3SG.POSS and then near:SPKR new -3SG.POSS funny.story*  
*go do.thingy Island.Name loan.wor and then OBLIGAT 3SG.SUBJ go Island.Name*  
*OBLIGAT 3SG.SUBJ go Island.Name and then person all.of -3SG.POSS 3PL.SUBJ*  
*go.down OBLIGAT 3PL.SUBJ boaru and then 3SG morning ANA 3SG.SUBJ*  
*NEG 3SG.SUBJ carry -TR and then lamp 3SG.SUBJ carry -TR*  
*lamp 3SG.SUBJ carry -TR and then 3SG.SUBJ go.down -to.SPKR 3PL.SUBJ*  
*OBLIGAT keroseen 1SG.SUBJ touch/make and then 1SG.SUBJ DUP-go.up loan.word*  
*3PL.SUBJ pour CAUS- full and then 3PL.SUBJ talk go 3SG.SUBJ*  
*-1SG.POSS loan.word 1SG.SUBJ get/take and then mind/heart -1SG.POSS 3SG.SUBJ forget*

*he wanted to go to Samarai. It was Saturday and he wanted to go to Samarai and all the people went down to board. And as for him, when he got up in the morning all his thoughts were that he wanted to go and sell his goods.*  
*when he thought: 'Eh, I have to get kerosene.' And so he run up (home again). And then he came down but didn't carry the container but carried the lamp. They said: 'That's a lamp!', and he said: 'Eh, I wanted to get kerosene.'*  
*And so he run up (home again). And then he came down but didn't carry the container but carried the lamp. They said: 'That's a lamp!', and he said: 'Eh, I wanted to get kerosene and I went up to get the cont.*

Figure 2. Single Layer search

TROVA 1.0.16 help

Simple **Single Layer** Multiple Layer

Types:  EAF (408)  Shoebox (82)  Text (16)  XML (2)  HTML (15)  PDF (341)

Domain: DoBeS archive

History:

Mode: case insensitive substring match **Reset form** Unicode input tier choice sort

Time: Minimal Duration Maximal Duration **Begin After** End Before

Find:  **Progress** Found 0 hits in 0 annotations

in **All Tiers [4286]**

in **Must be in same file**

in **All Tiers [4286]**

in **Must be in same file**

in **All Tiers [4286]**

Action: **Show Concordance View** Page: < >

Context Size: 4 Font: **Arial Unicode MS** 12 Show Info Balloons

Figure 3. Multiple Layer search