

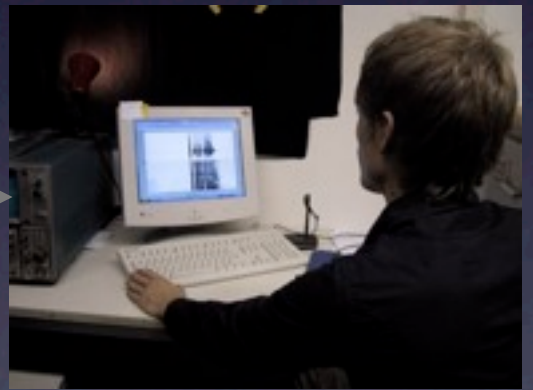
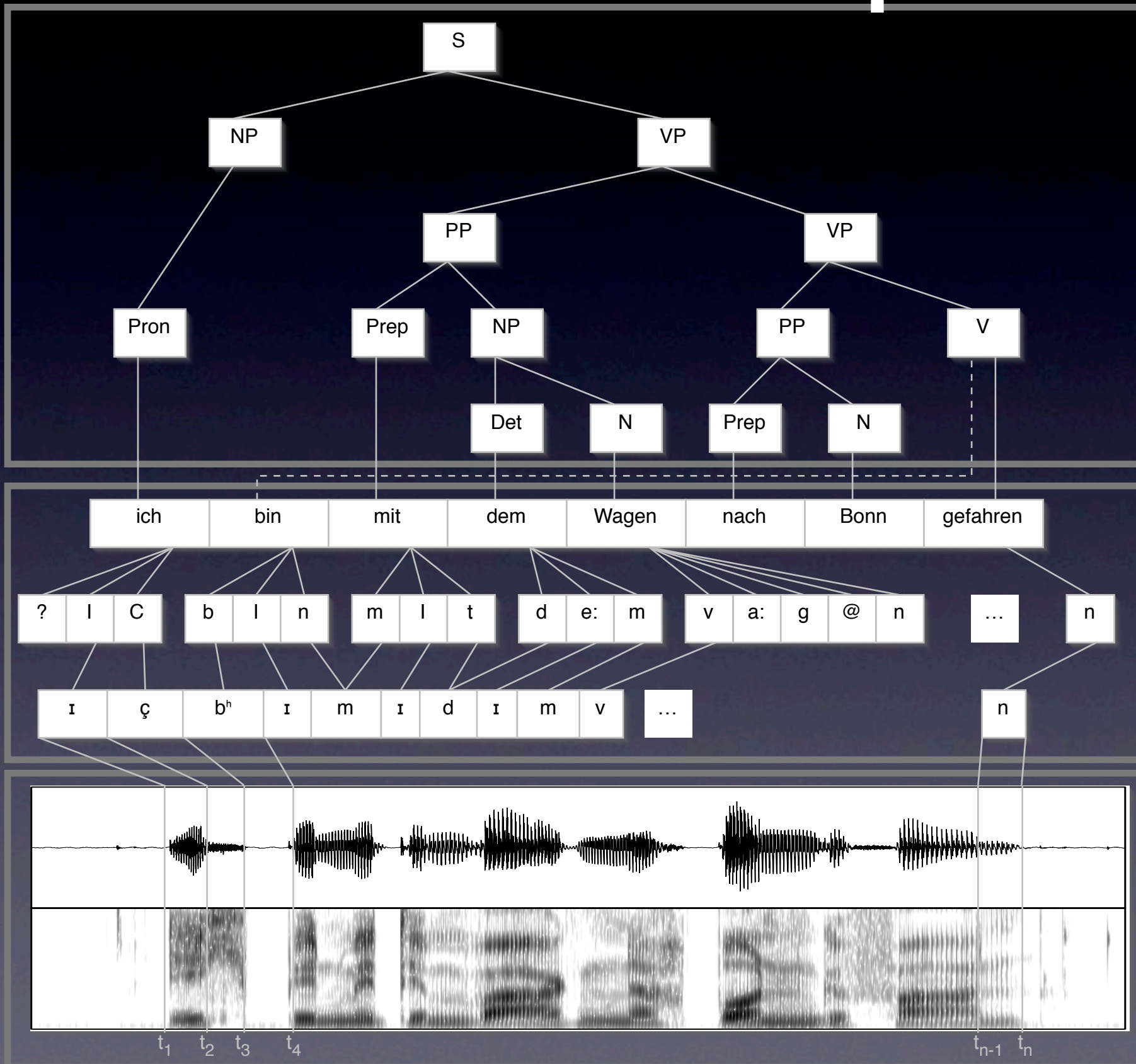
Tools come and go,
the data will stay!

draxler@phonetik.uni-muenchen.de

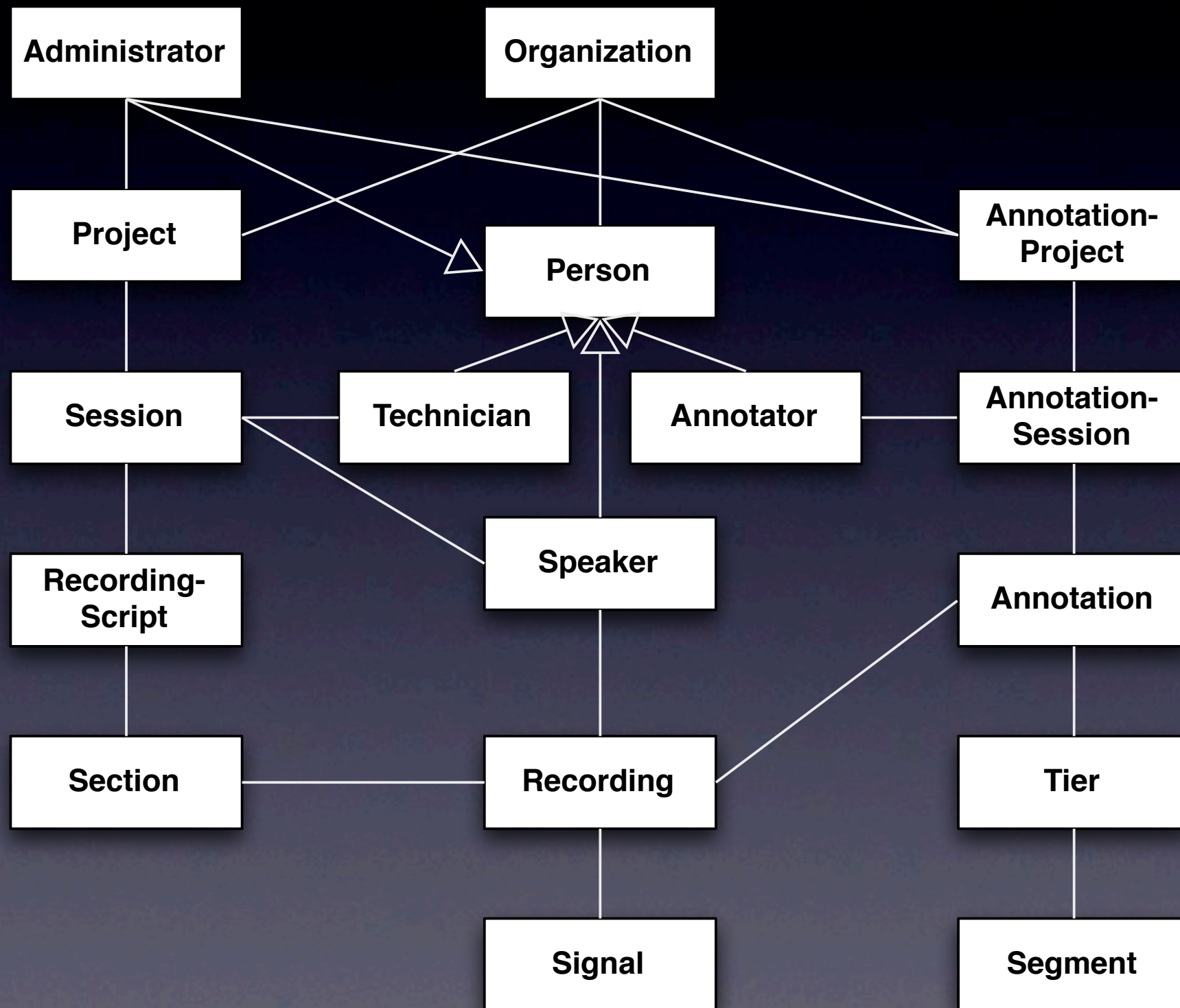
BAS (Munich)

- Speech databases
 - 30+ DBs, lexicons, IMDI-descriptions
- Tools
 - web-based annotation
 - SpeechRecorder, MAUS

Example



A global data model



SQL Query

```
select m.label as phoneme, k.label as canonic, o.label as  
       word, count(f.f1), avg(f.f1)::int as f1, avg(f.f2)::int as f2
```

```
from session ses  
join signalfile sig on ses.session = substring(sig.filename, 3, 9)  
join segment m on sig.id = m.signal_id and m.tier = 'MAU:'  
join segment o on m.signal_id = o.signal_id and  
       m.ref_seg = o.ref_seg and o.tier = 'ORT:'  
join segment k on k.signal_id = o.signal_id and  
       k.ref_seg = m.ref_seg and k.tier = 'KAN:'  
join formant f on m.signal_id = f.signal_id and  
       f.time between (m.begin_seg + (m.dur_seg * 0.2)) and  
       (m.begin_seg + (m.dur_seg * 0.8))  
join speaker spk on ses.session = spk.speaker_code and  
       ses.project = spk.project and ses.project = 'VOYS'
```

```
where m.label = 'E'
```

```
group by m.label, k.label, o.label  
order by m.label, k.label, o.label;
```


Views = Virtual Tables

```
create view voys_data as
select m.label as phoneme, k.label as canonic, o.label as word,
spk.sex as sex, spk.age as age, ses.recording_place as school, f.f1
as f1, f.f2 as f2
```

```
from session ses
join signalfile sig on ses.session = substring(sig.filename, 3, 9)
join segment m on sig.id = m.signal_id and m.tier = 'MAU:'
join segment o on m.signal_id = o.signal_id and
m.ref_seg = o.ref_seg and o.tier = 'ORT:'
join segment k on k.signal_id = o.signal_id and
k.ref_seg = m.ref_seg and k.tier = 'KAN:'
join formant f on m.signal_id = f.signal_id
join speaker spk on ses.session = spk.speaker_code and
ses.project = spk.project
where f.time between (m.begin_seg + (m.dur_seg * 0.2)) and
(m.begin_seg + (m.dur_seg * 0.8))
and ses.project = 'VOYS'
```

Views & Queries

- application-specific, compact

```
select phoneme, avg(f1)::int, avg(f2)::int
from voys_data
where phoneme = 'E'
group by phoneme
```

- SQL still visible
- possibly inefficient