

ANVIL & AG

There and Back Again...

Chicago, 21 June 2007

Michael Kipp

DFKI

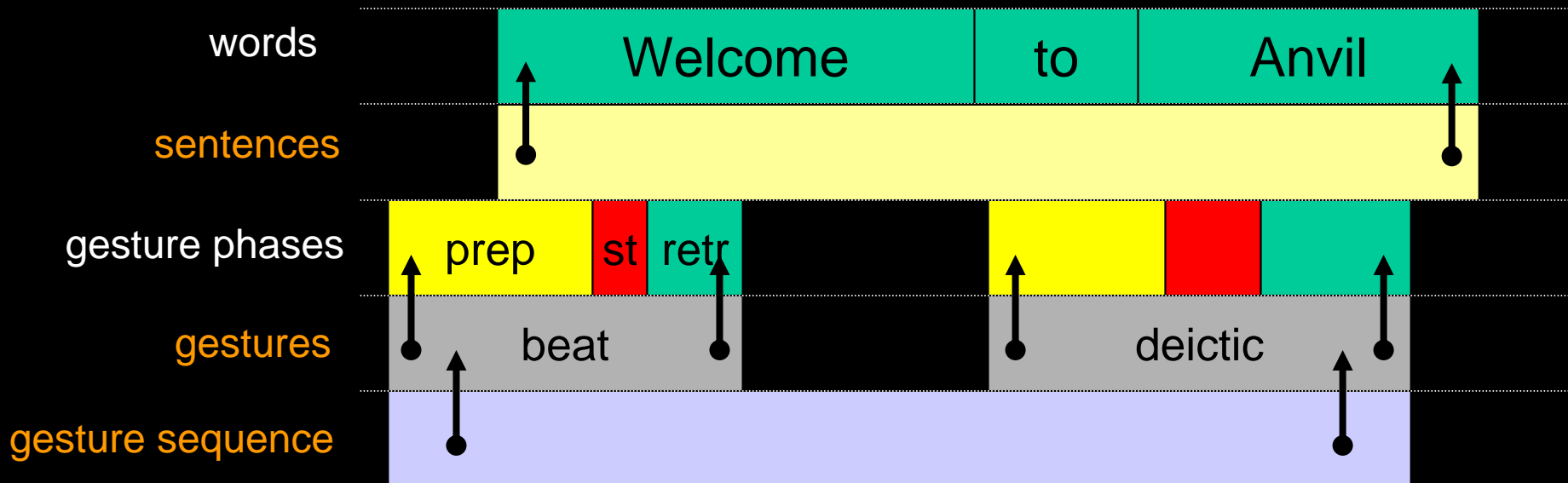
German Research Center for Artificial Intelligence
Saarbrücken, Germany

michael.kipp@dfki.de



ANVIL: Tracks + Track Hierarchy

Elements in one track can be composed of elements in another track



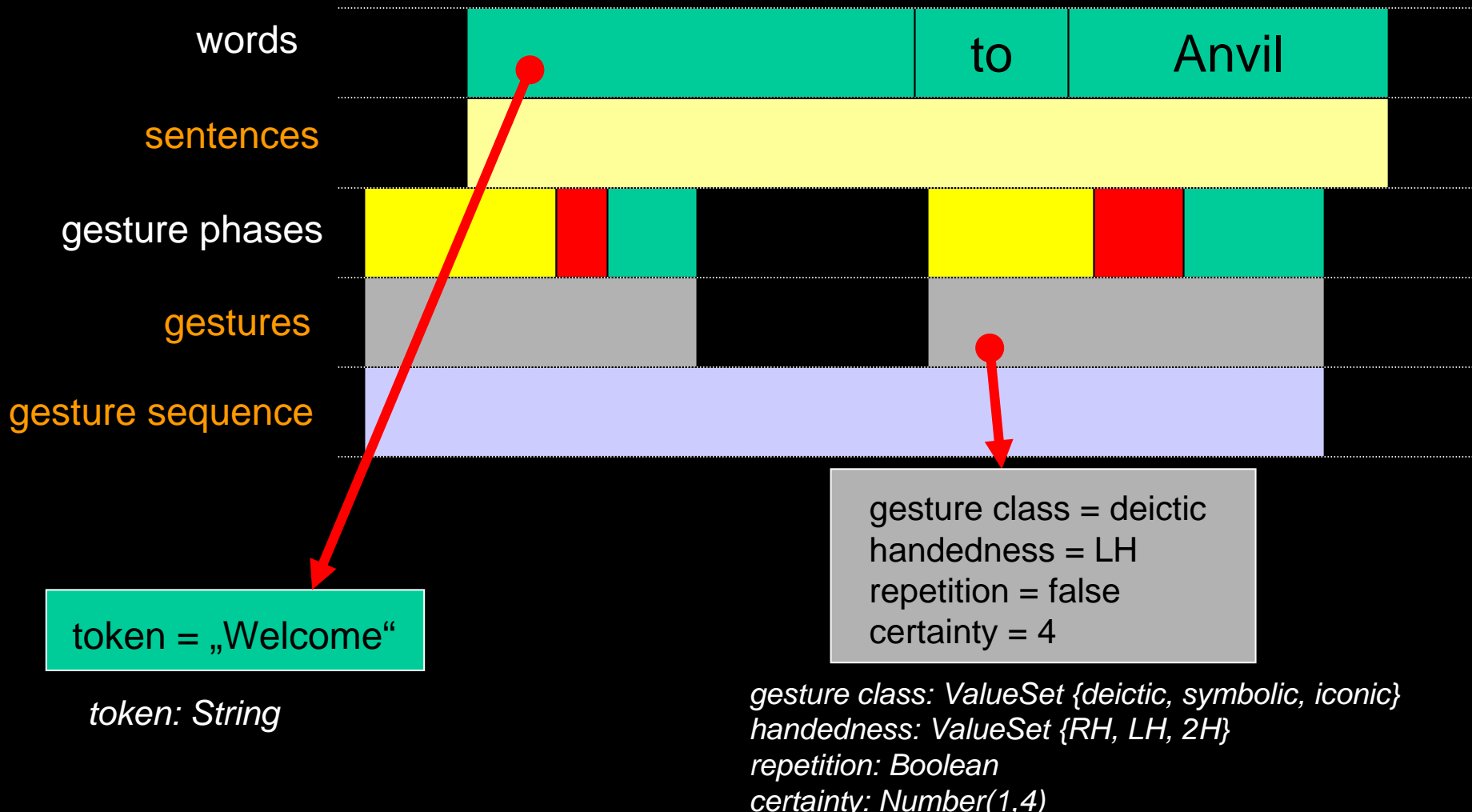
Primary Track: Elements start + end in absolute time

Secondary Track: Start + end determined by "reference" elements

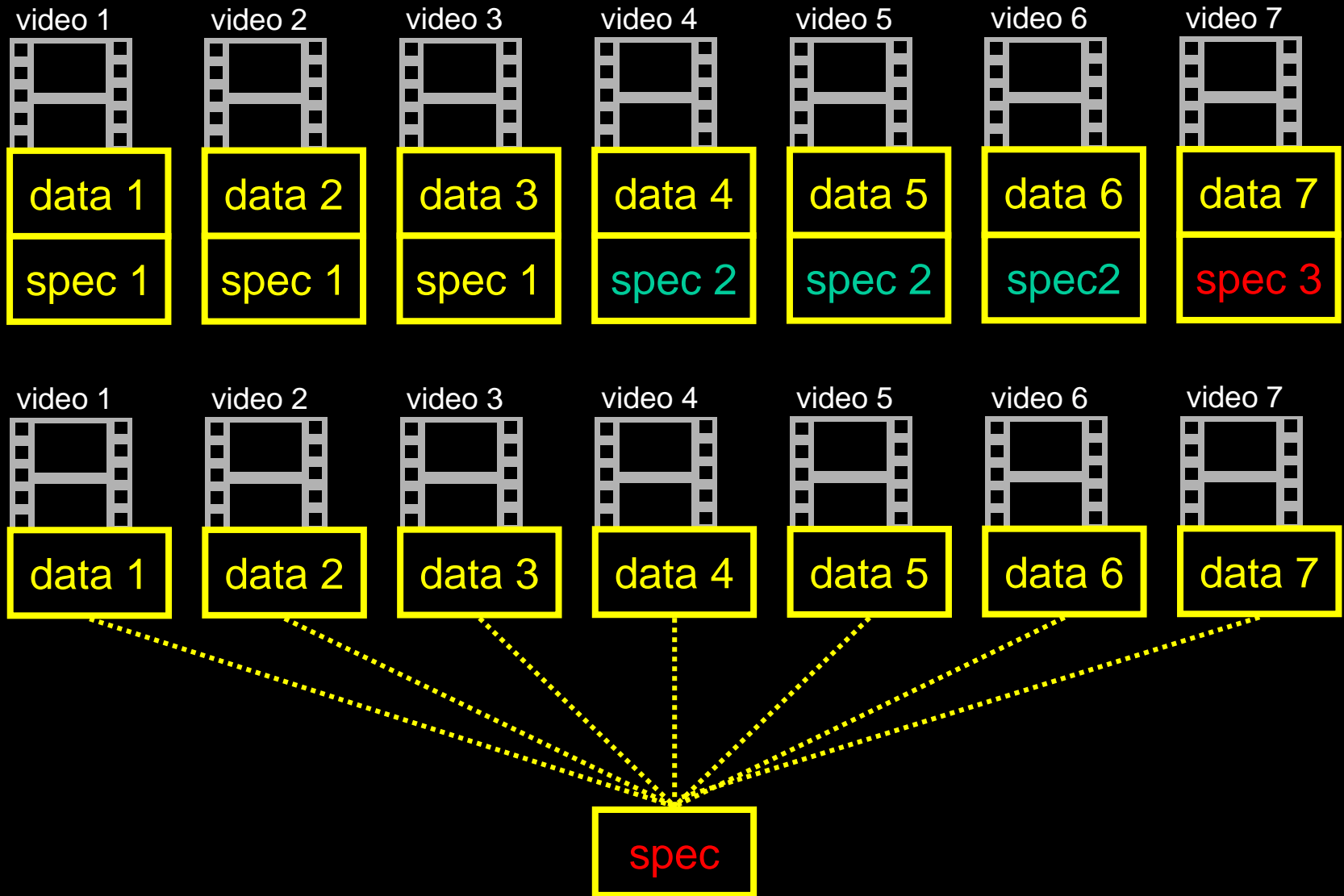
Advantage: ANVIL takes care of alignment, even if elements are moved (saves you work + errors)

ANVIL: Structured Elements

Each element contains typed attributes instead of a simple string.



Separation of Coding Scheme and Data



Annotation Graph (AG)

type

phase

features

name = prep

features

name = stroke

features

name = retract

time points (anchors)

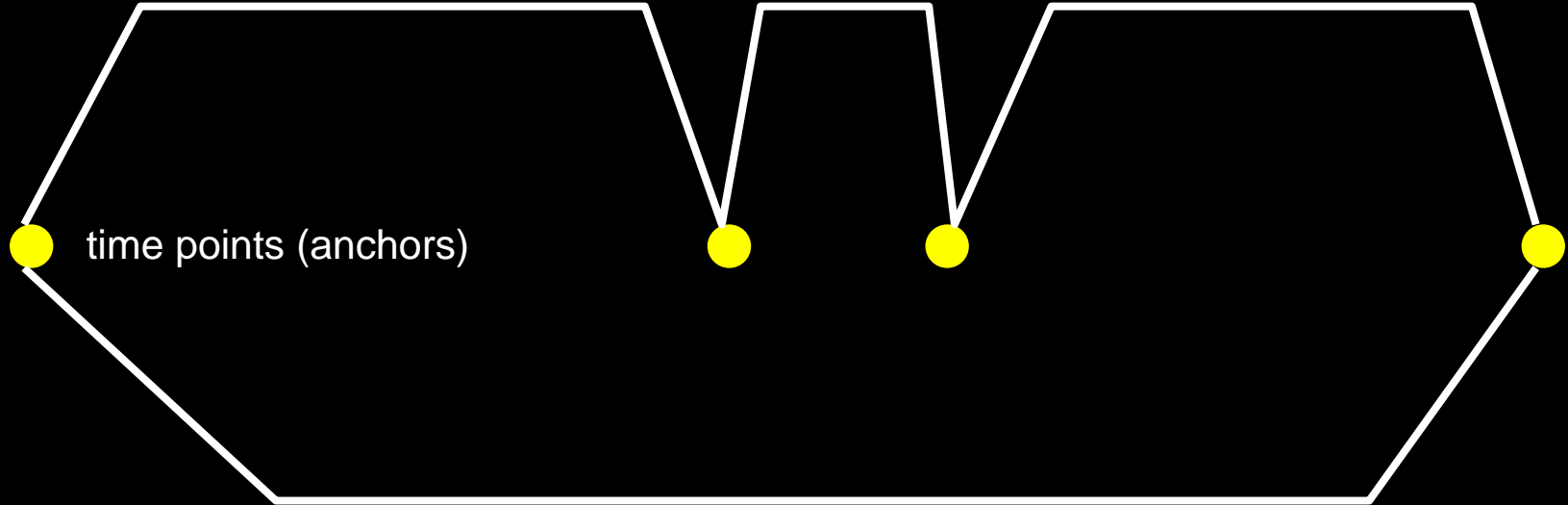
features

class = deictic

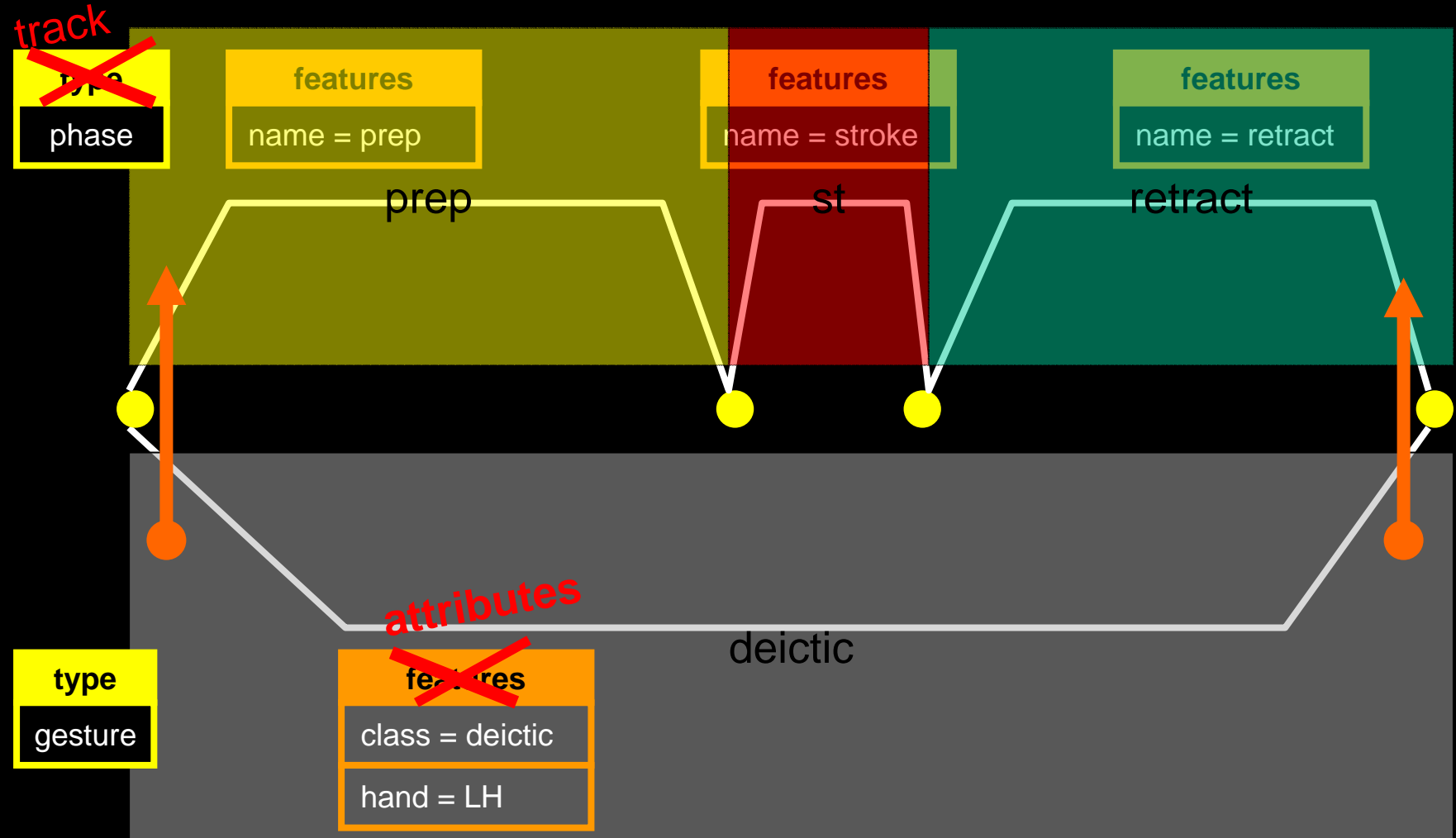
hand = LH

type

gesture



Annotation Graph → ANVIL



ANVIL to AG Algorithm

- For each primary track with name **T**:
step through all elements and...
 - create anchors for start + end time
 - create an arc with type **T**
 - create a feature for each non-empty attribute
- For each secondary track: like above
but re-use existing anchors

AG to ANVIL Algorithm

- Problem: AGs have no structural information
 - what are all possible types?
 - what are all possible features + feature values?
 - no “coding scheme” anywhere – all implicit !!!
- If there is an ANVIL specification → do the previous conversion backwards
- If not, extract a specification by searching through AG:
 - for each new type: create a primary track
 - for each new feature: create an attribute
 - write everything to a new ANVIL specification file
- Drawbacks
 - Error if elements overlap
 - No secondary tracks

Problems & Thoughts

- Why AGs at all?
 - pro: it is already there and favors no tool
 - pro: some AG software exists
 - con: software is buggy
 - con: AGs are very basic (non-AG stuff must be stored)
 - con: AGs graph-based but all tools tier-based
- Challenge
 - Agreement on how to encode non-AG features (see tool matrix)
 - Agreement what an AG Set is (my interpretation: same coding scheme)

Thanks for your attention!

www.dfki.de/~kipp/anvil