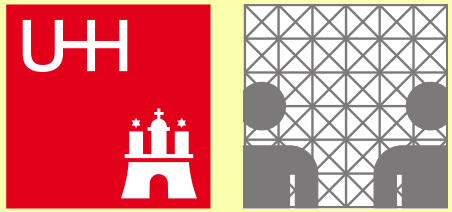
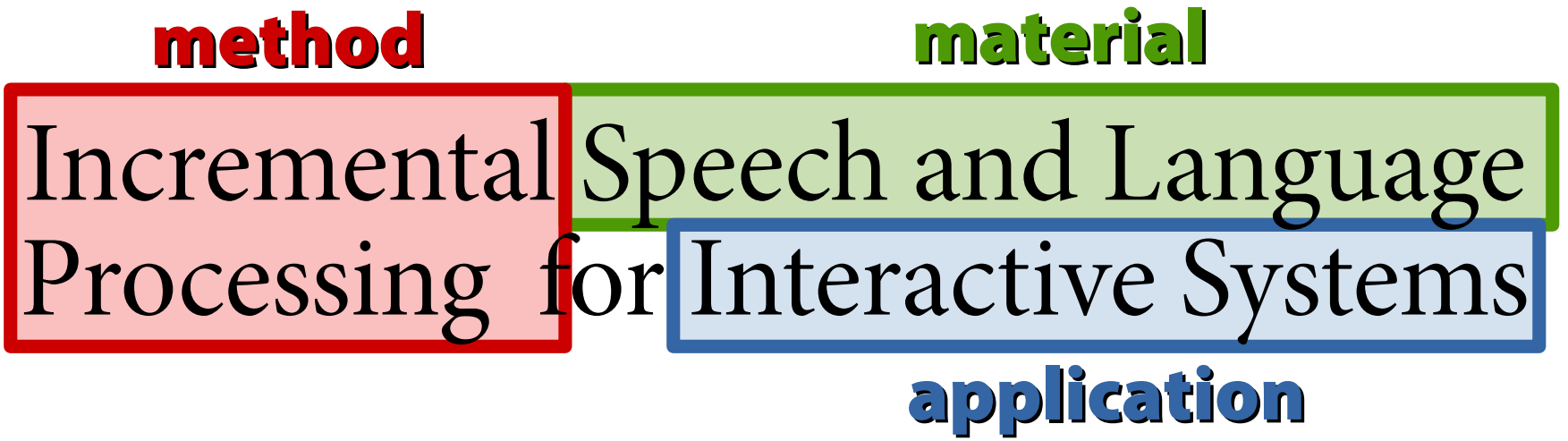


ESSLLI

Incremental Speech and Language Processing for Interactive Systems

Timo Baumann, Arne Köhn,
Universität Hamburg, Informatics Department
Natural Language Systems Division
baumann@informatik.uni-hamburg.de


method
material
Incremental Processing for Speech and Language Interactive Systems
application

Contents of the Course

- today:
  - introduction, major features of incremental processing
- Tuesday:
  - incremental processing for sequence problems
- Wednesday:
  - incremental processing for structured problems
- Thursday:
  - generating output based on structured and partial input
- Friday:
  - wrap-up and outlook, also based on your questions and interests

Contents for today

- What *is* incremental spoken language processing?
- A general model for incremental processing

Motivation:
What *is* incremental processing?

Please take out pen&paper

- write down the next word(s) for each sentence:
  - Ladies and ____.
  - Thank you ____ ____.
  - Yesterday, all my ____ ____ ____ ____ ____.

A few more:

- add as many (likely) words as you deem plausible:
  - My name is Bond, ____ (____) (____) (____) ...
  - A long time ago in ____ (____) (____) (____) ...
  - Have ____ (____) (____) (____) ...
  - The horse raced past the ____ (____) (____) (____) ...

- this time, there was no „future context“, your hypotheses of what comes next are based on incremental processing alone!

A more detailed look

- The horse raced past the barn (fell) (____) (____) ...
- DT NN ~~XXX~~ VBD IN DT NN VBD
  VBN

- note down the (simplified) parts of speech for each word
  - what is a likely next POS? how do you know? are you certain?

- garden-path sentences require re-analysis, or put differently:
- it's impossible to delay analysis until the situation is unambiguous (because ambiguity is resolved non-locally)

What we've done

- predict what's next (and beyond)
- based on an analysis of what was before
- using a variety of information (syntactic, semantic, pragmatic, ...)

  ➔ **this is called incremental processing!**

- the farther we look into the future, the more we speculate
- the more complex our reasoning, the more we speculate
  - we need to be able to re-analyze and re-decide

  ➔ **this is called non-monotonous incremental processing!**

Monotonicity vs. Non-monotonicity

- monotonicity: output that has been made will not be changed in the future (it will only be added to)

- non-monotonicity: previous output can be changed later on
  - allow to „change our mind“ given new evidence

Please take out Pen&Paper

- write down advantages/disadvantages of monotonous/non-monotonous incremental output

Monotonicity vs. Non-monotonicity

- monotonous
  - results are reliable
  - results must be correct (or will remain wrong)
  - easy to process monotonously extended input
- non-monotonous
  - results can later turn out to be wrong
  - results can later be improved (this enables speculation)
  - need to manage changing input

- as monotonously as possible, as non-monotonous as necessary
- have an expectation how likely something is meant to stay

Incremental processing in Interaction

The temporal aspect of interaction

- human speech is surprisingly slow
  - ~10 speech sounds per second
  - typing is not faster either (~1 word per second)

→ this is an impediment to human-computer interaction in conventional (non-incremental) systems, when pauses are used to indicate completion of a user's turn:

U:

S:

Speedings things up

- using slow delivery to fold processing time (waiting time is reduced to threshold)

> *"Every Millisecond Matters!"*
> Arvind Jain (Google)

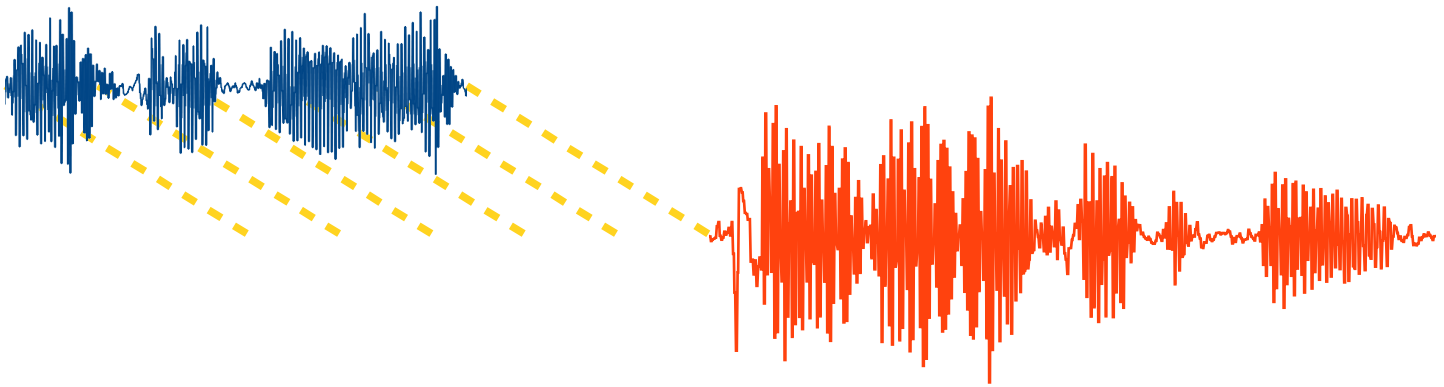
U:

S:

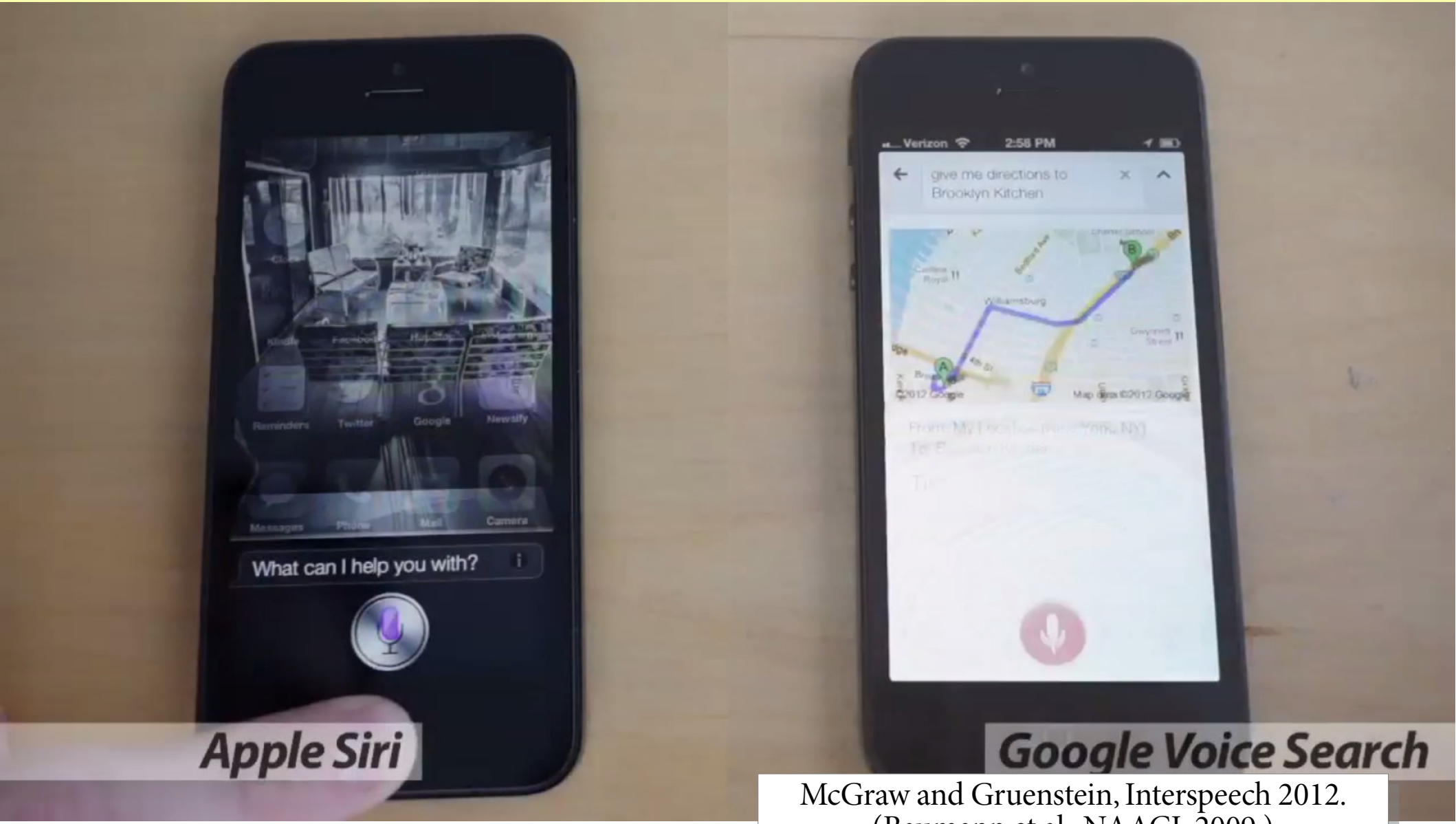
Key ingredients of incremental processing

- processing is triggered by minimal input
  - input must be somehow divisible into smaller units
- processor may predict continuation of ongoing actions
- system may act based on current understanding
- … but actions may have to be revised occasionally

→ promises to speed up processing

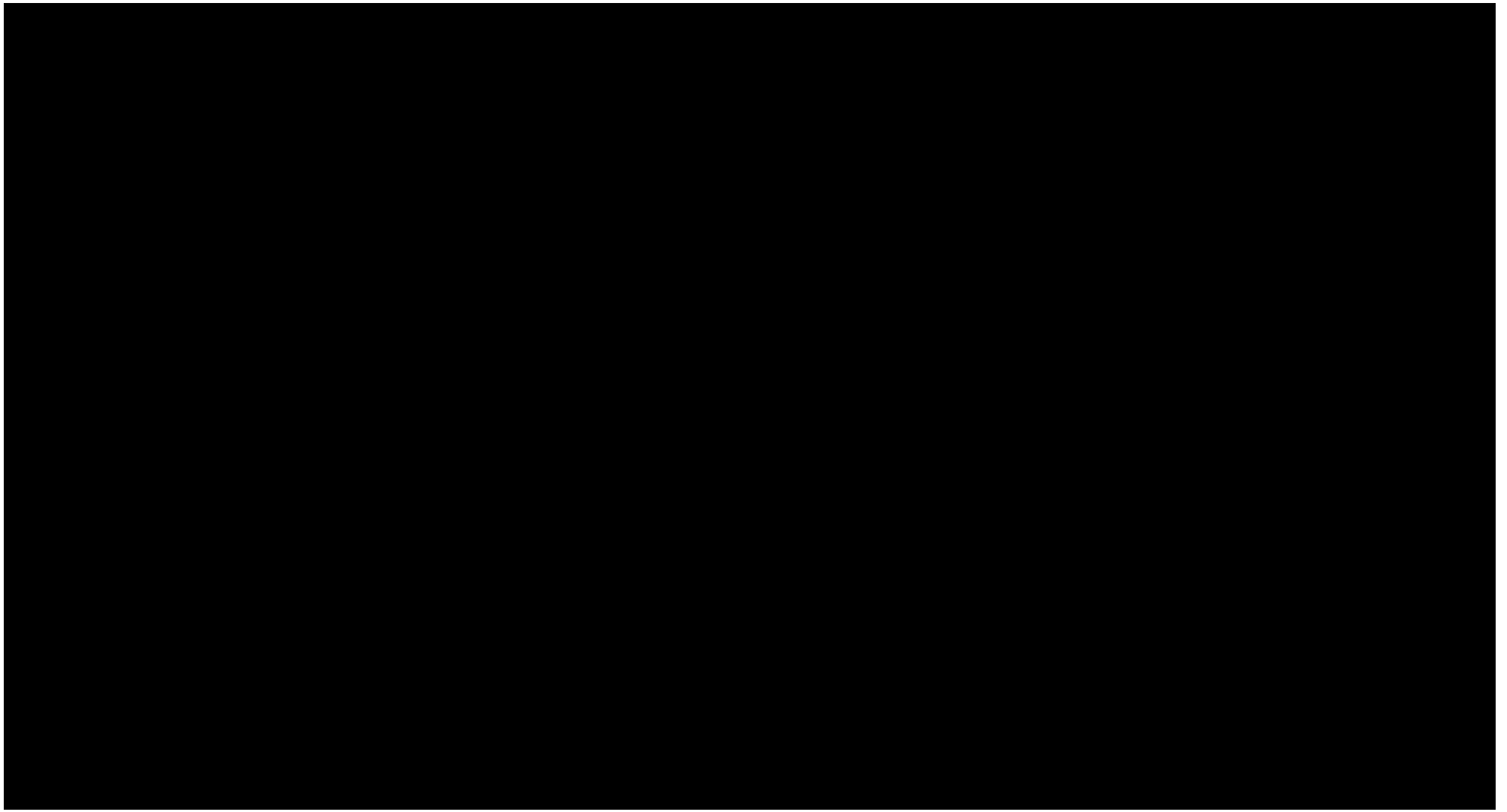


Example: Google™ incremental voice search



McGraw and Gruenstein, Interspeech 2012.
(Baumann et al., NAACL 2009.)

Example: Google incremental voice search



ok, it's faster,
but won't faster computers solve that?

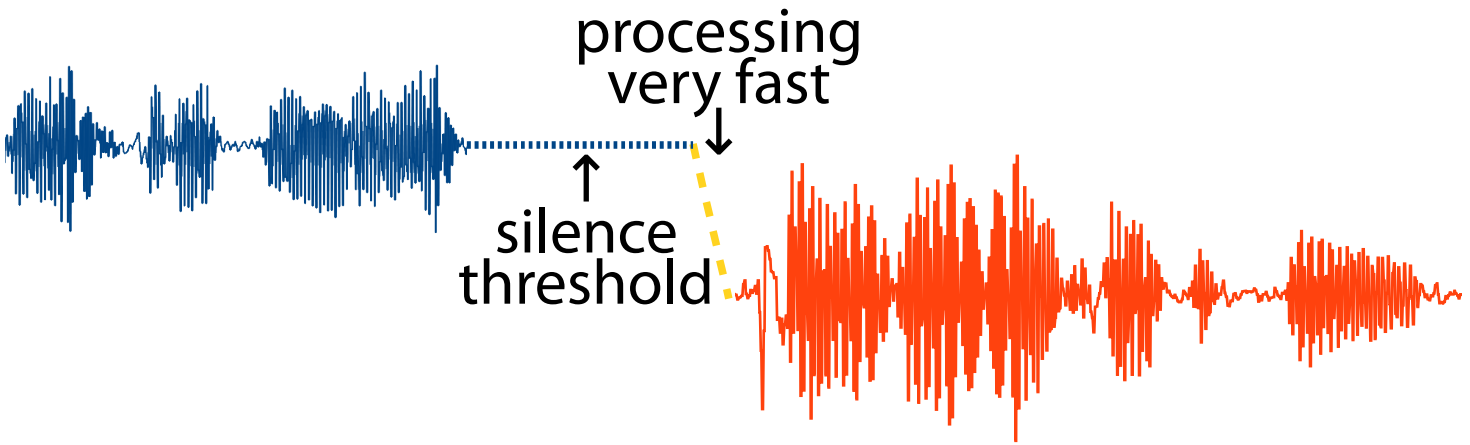
Silence thresholds

- *silence is a bad marker* for speaker-change
  - utterances may contain silences themselves
    - „I need to travel on … uhm … hold on … Sunday“

Sorry, when do you ...

  - silence is likely to be interpreted by interlocutor

→ unprincipled use of time in conventional systems



Turn-taking in dialogue

- what we think our turn-taking behaviour is:

  A:

  B:

- what it actually looks like:

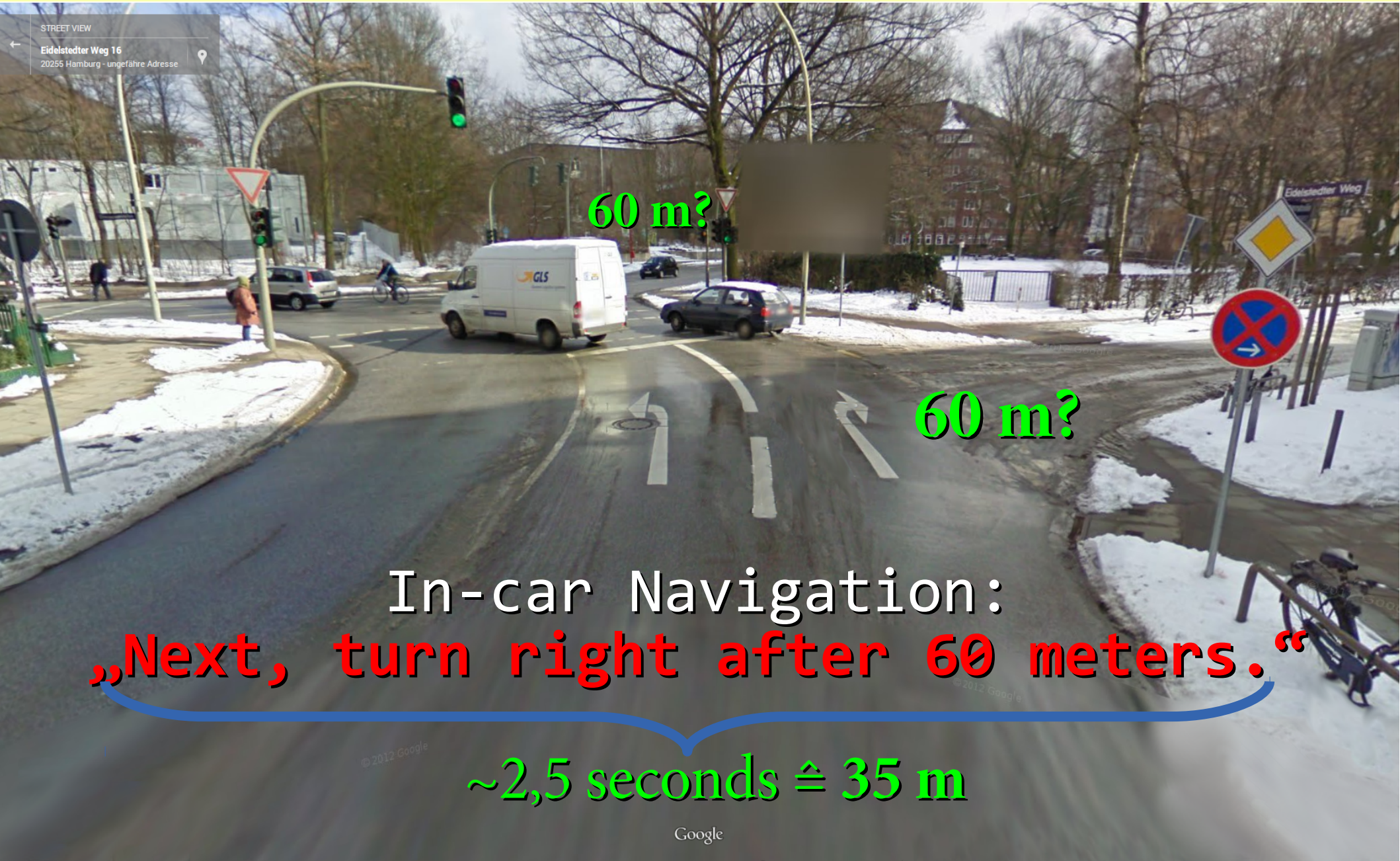
  A:

  B:

"Dialogue interaction is an intricate dance."

Nigel Ward (at SSW 2013)

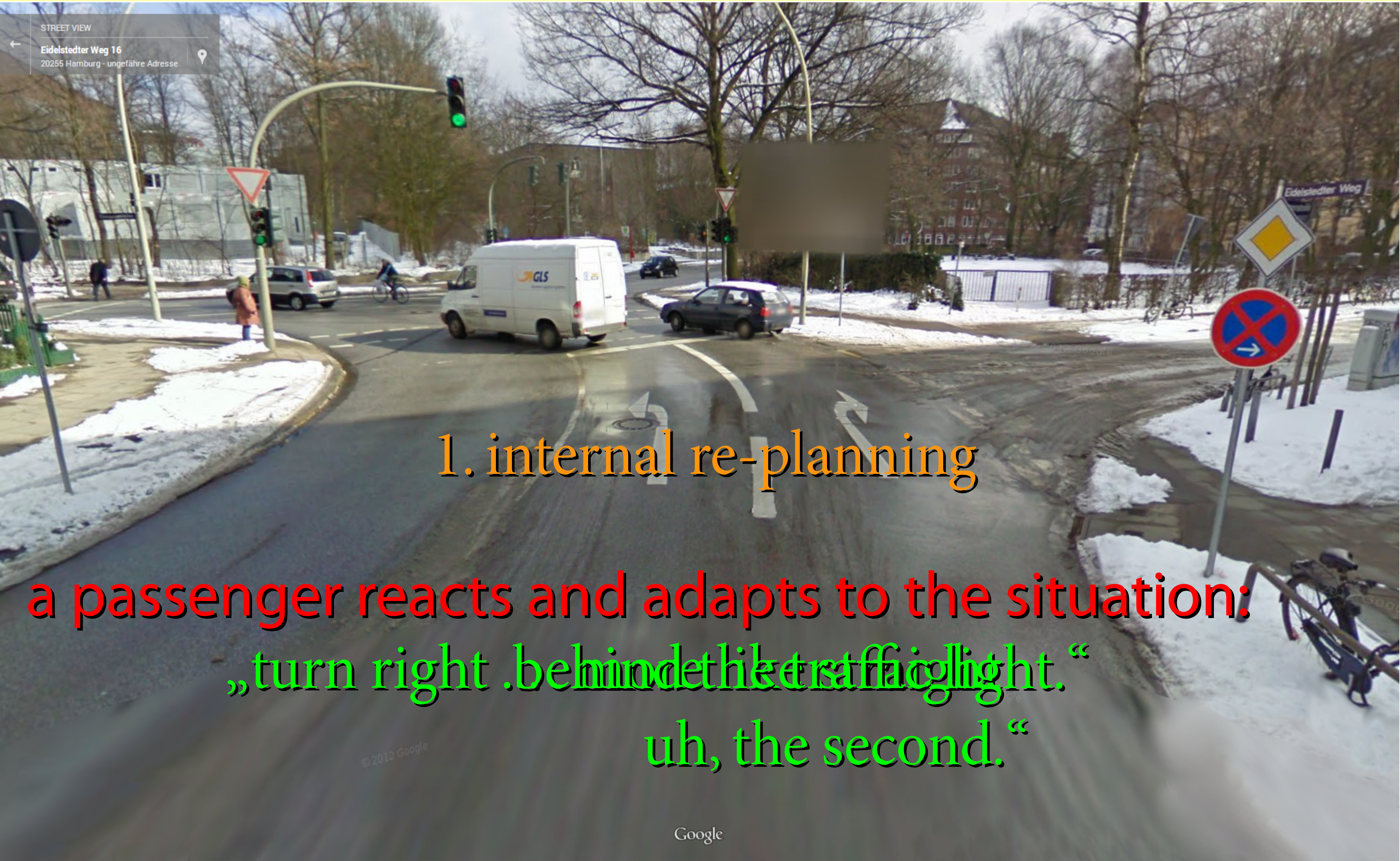
Example


STREET VIEW
Eidelstedter Weg 16
20255 Hamburg - ungefähre Adresse
60 m?
60 m?
Eidelstedter Weg
In-car Navigation:
„Next, turn right after 60 meters.“
~2,5 seconds ≙ 35 m
Google

Example


STREET VIEW
Eidelstedter Weg 16
20255 Hamburg - ungefähre Adresse
Eidelstedter Weg
GLS
Spoken language
unfolds in time
↦ this is both a challenge
and the solution
Google

Human speakers are *responsive*.



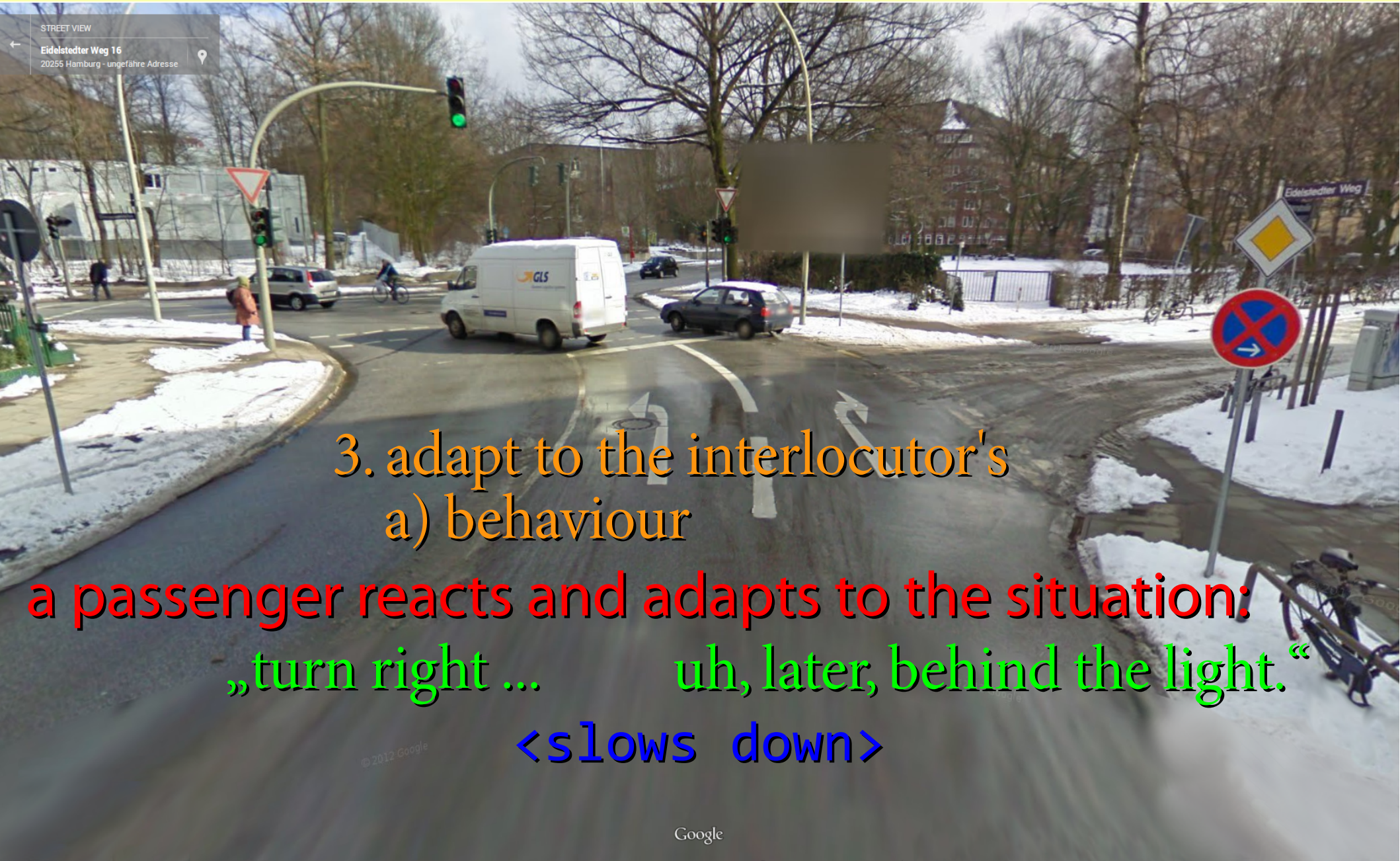
Human speakers are *responsive*.

2. external events

a passenger reacts and adapts to the situation:

„turn right ... following the blue compact.“

Human speakers are *responsive*.



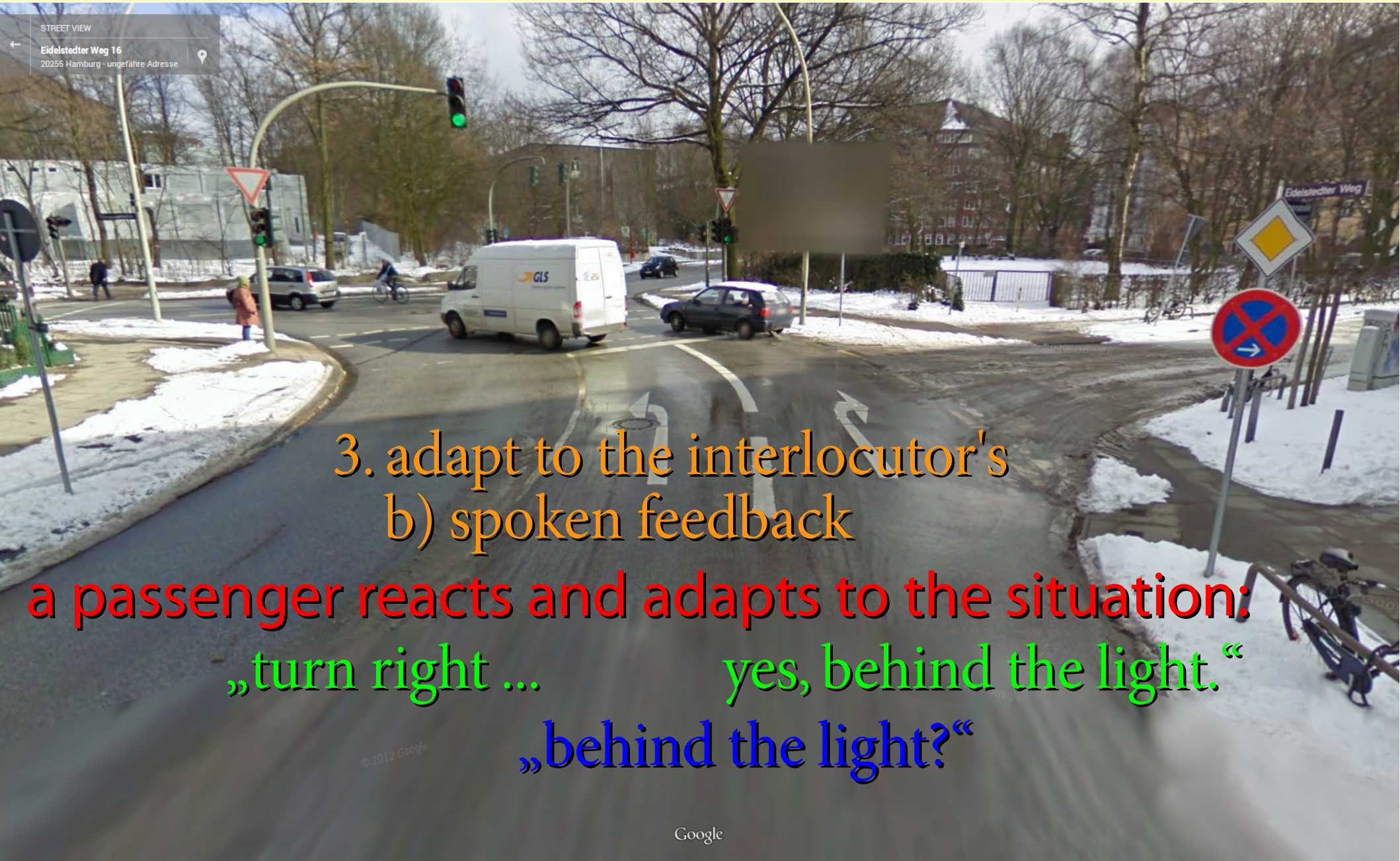
3. adapt to the interlocutor's
a) behaviour

a passenger reacts and adapts to the situation:

„turn right ... uh, later, behind the light.“

<slows down>

Human speakers are *responsive*.



3. adapt to the interlocutor's
b) spoken feedback

a passenger reacts and adapts to the situation:

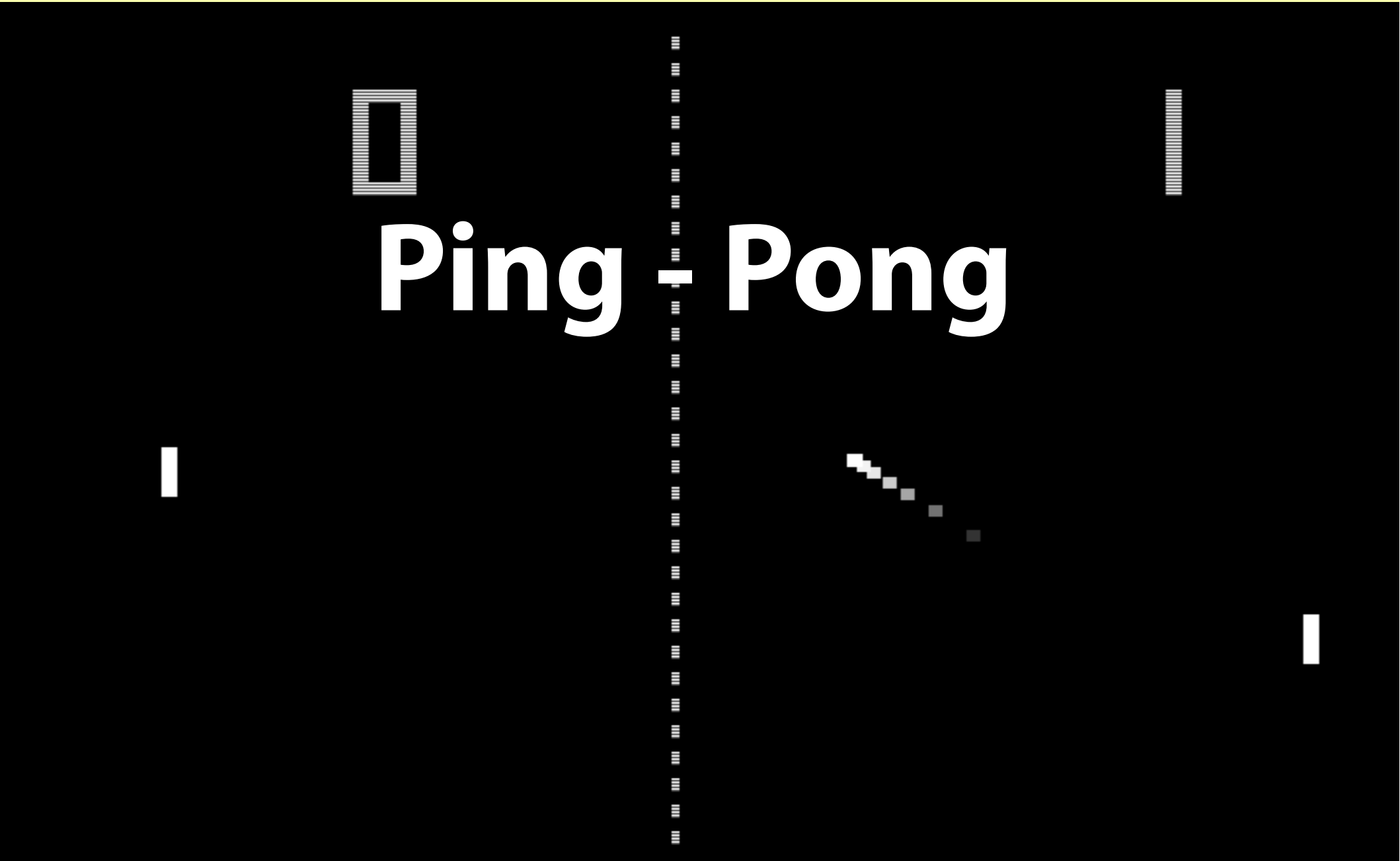
„turn right ... yes, behind the light.“

„behind the light?“

Final example: Human language processing

- „I never know beforehand how I will end my sentences.“
- „I often know how someone else will end their sentence.“
- humans …
  - produce speech incrementally (Levelt 1989)
  - perceive speech incrementally (Tanenhaus et al. 1995)
  - collaborate incrementally in dialogue (Clark 1996)
- current systems …

Predominant interaction model of computer-human spoken interaction



„Root causes of lost time and user stress“

- 7 issues detrimental to efficient and pleasant dialogue with spoken dialogue systems
- 3 of those directly related to Ping-Pong interaction and limited interactivity:
  - time-outs, responsiveness, feedback
  - these factors make human-computer-dialogue unnatural
- **there's actually a problem that needs to be solved!**

(Ward et al., Interspeech 2005)

Motivation: Summary

conventional system

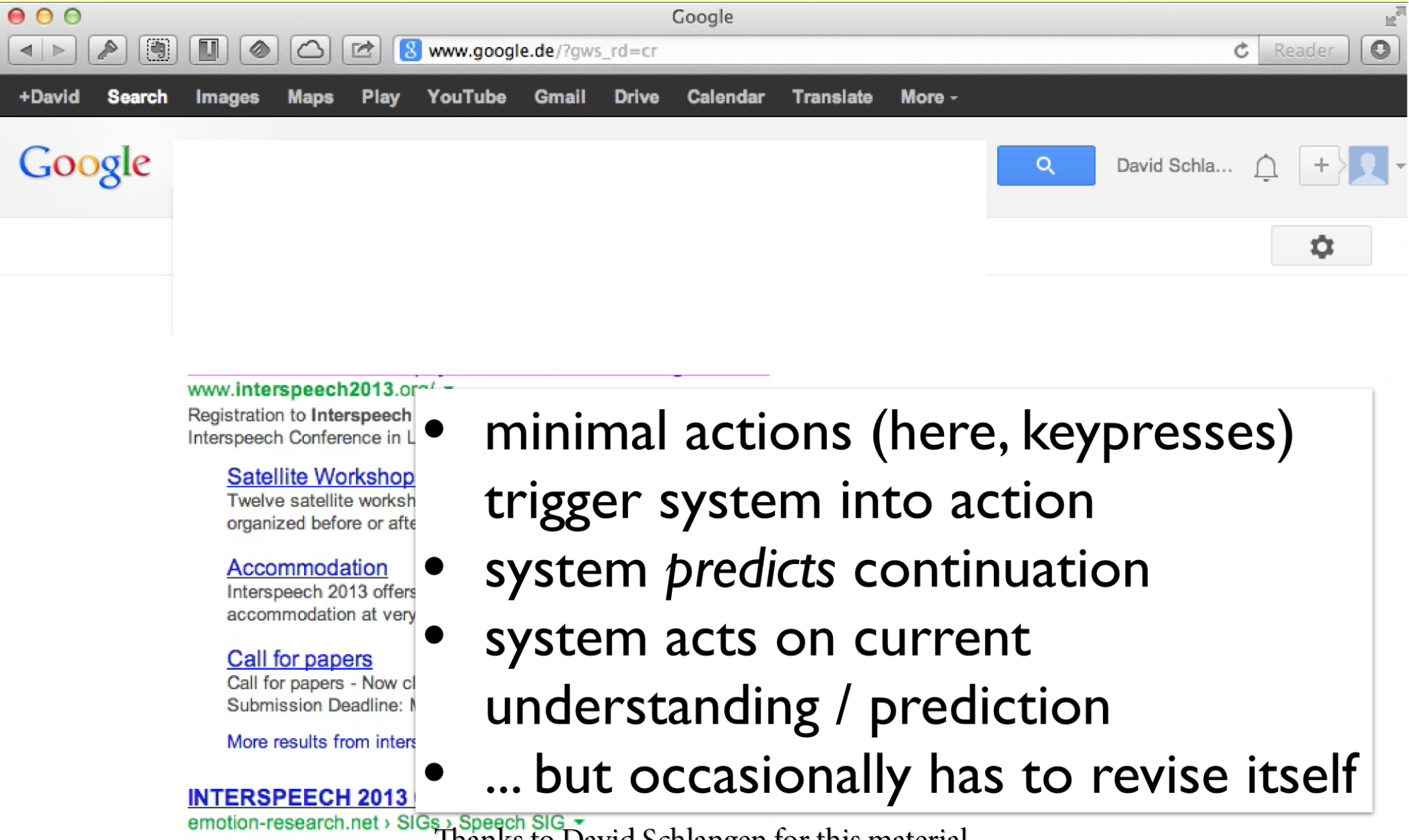
- processing delays *between* the turns
- delays from processing
- purely reactive *Ping-Pong* interactions
- uncontrolled use of time

incremental system

- processing *just-in-time during* turns
- processing folded into delivery
- more flexible: supports feedback, collaboration, *dance*
- timing is part of reasoning

The implications of incremental processing in interaction

Example: Google instant web search



- minimal actions (here, keypresses) trigger system into action
- system *predicts* continuation
- system acts on current understanding / prediction
- ... but occasionally has to revise itself

Thanks to David Schlangen for this material.

Tightening the Feedback Loop

- Incremental processing changes the progression of the interaction:
- observable predictions can influence interaction
  - e.g.: search predictions may become self-fulfilling prophecies
- this is a philosophical issue and ethical issue
  that you MUST be aware of!

Motivation: Challenges

- processing minimal input, creating partial output
  - input must be somehow divisible into smaller units
  - structure for partial output needs to be devised

→ requires re-conceptualisation of information flow

- system acts on current understanding (based on partial information)

→ introduces (even more) uncertainty and room for error

  - does it still pay off?

→ how does this alter the overall interaction?

A general, abstract model for incremental spoken language processing

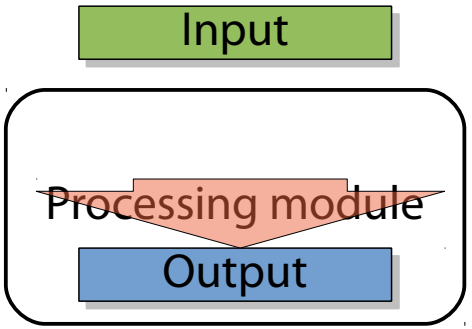
(Schlangen and Skantze EACL 2009, Dialogue & Discourse 2011)

Incremental Processing: a Definition

- an incremental processor consumes input and generates output in a piece-meal fashion.
- (preliminary) output is generated before all input has been consumed (at least in some situations).

Incremental vs. Non-incremental Processing

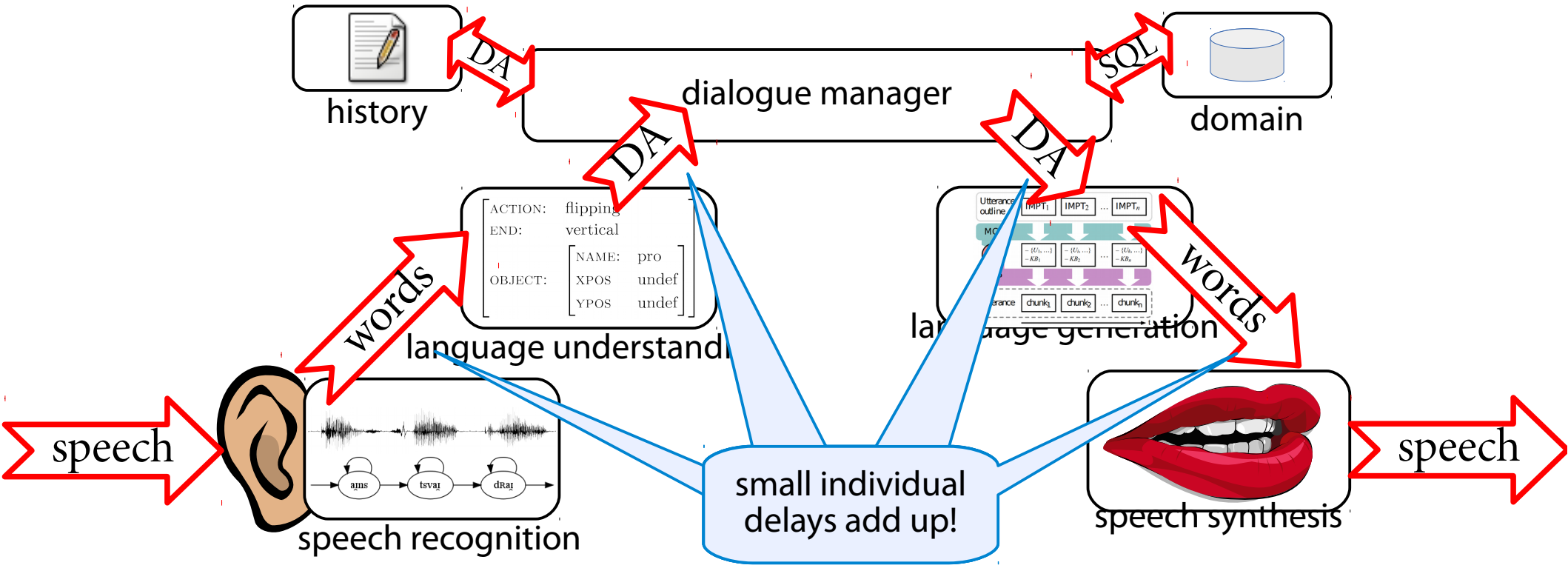
- non-incremental, *decoupled* processing:



- Processing is effected after the input → delay!
  - in a modular system: delays add up

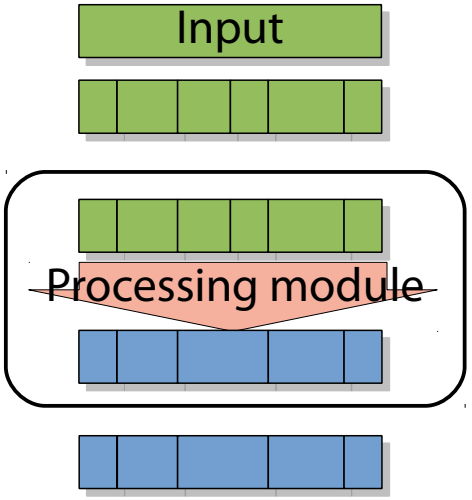
Modelling spoken language processing

- spoken language processing is a complex task, that must be sub-divided and handled by specialized components



output generation starts only after input has ended

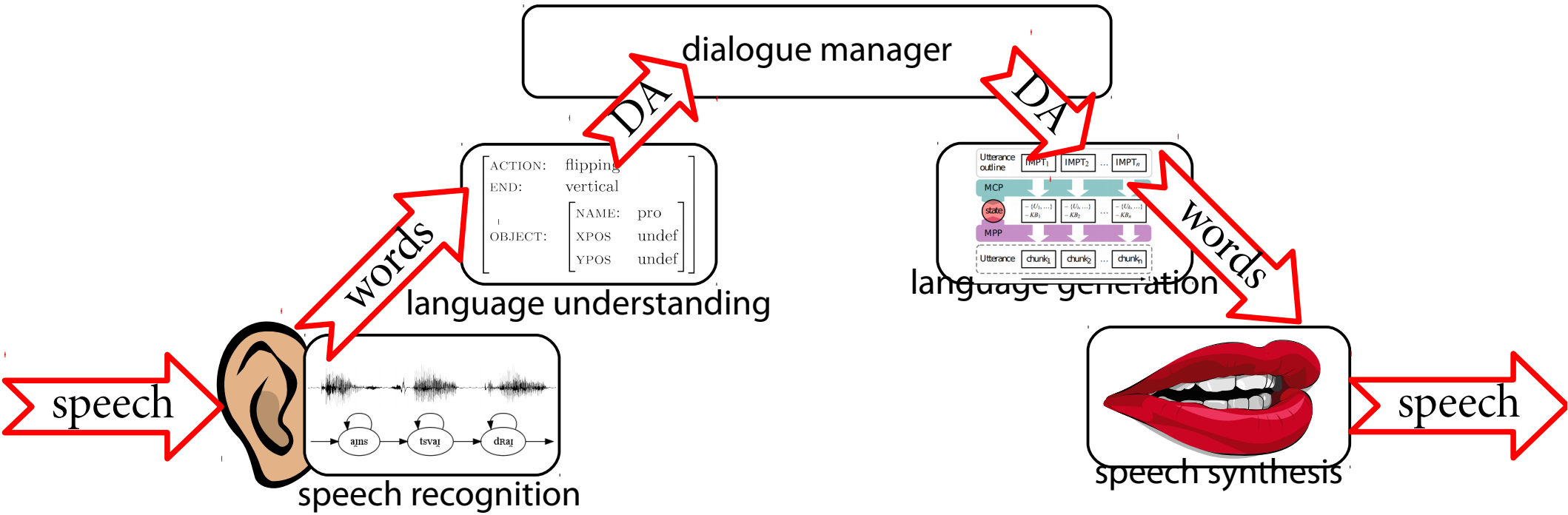
Incremental Processing



- input consists of individual units that are consumed one-by-one (e.g. speech audio, words, ideas, …)
- input is consumed unit-by-unit, and output is generated
- input units may be aggregated to larger units

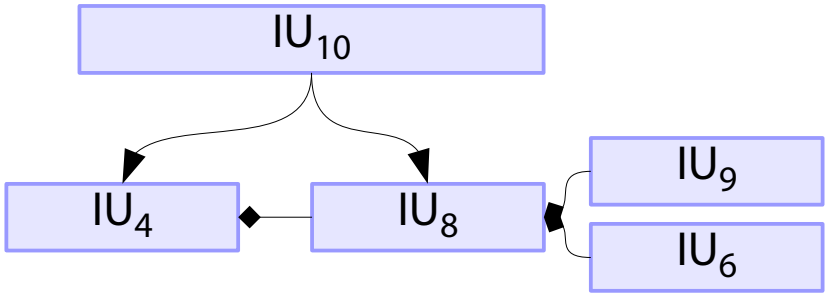
Modelling **incremental** spoken language processing

- components are concurrently processing minimal amounts of input



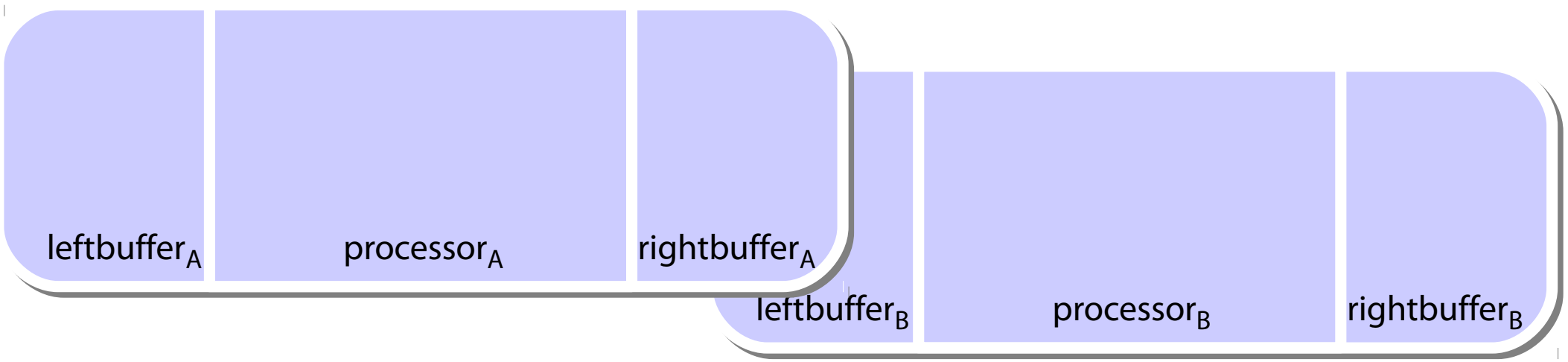
the Incremental Unit

- linked with corresponding unit(s) on the lower/higher levels of abstraction
- linked with neighbouring units on the same level
  - one link pointing backward in time
  - potentially multiple links pointing forward



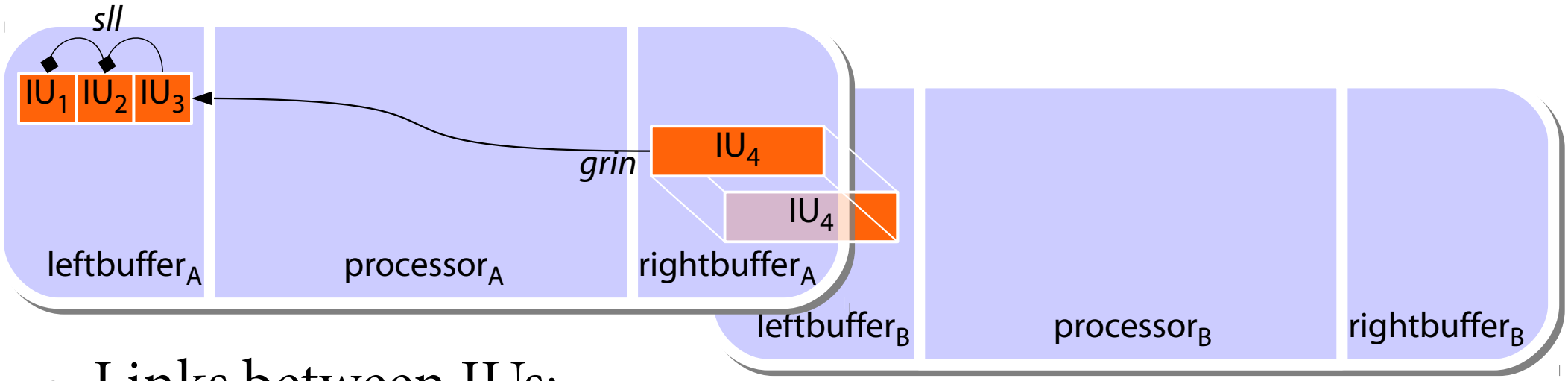
Processing modules

- processing modules are connected via buffers



Processing modules

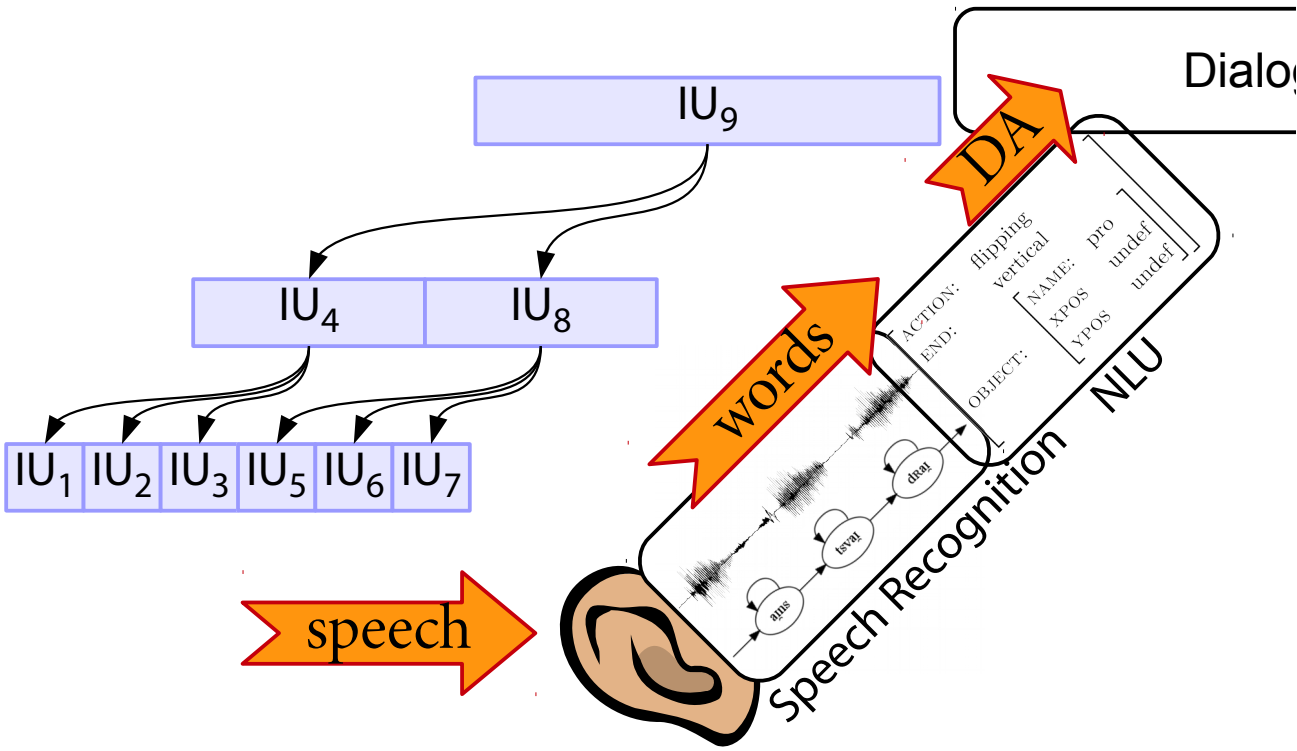
- processing modules are connected via buffers
- buffers contain incremental units (IUs)



- Links between IUs:
  - **grounded-in** links (*grin*) denote ancestry
  - **same-level** links (*sll*) for information of the same type

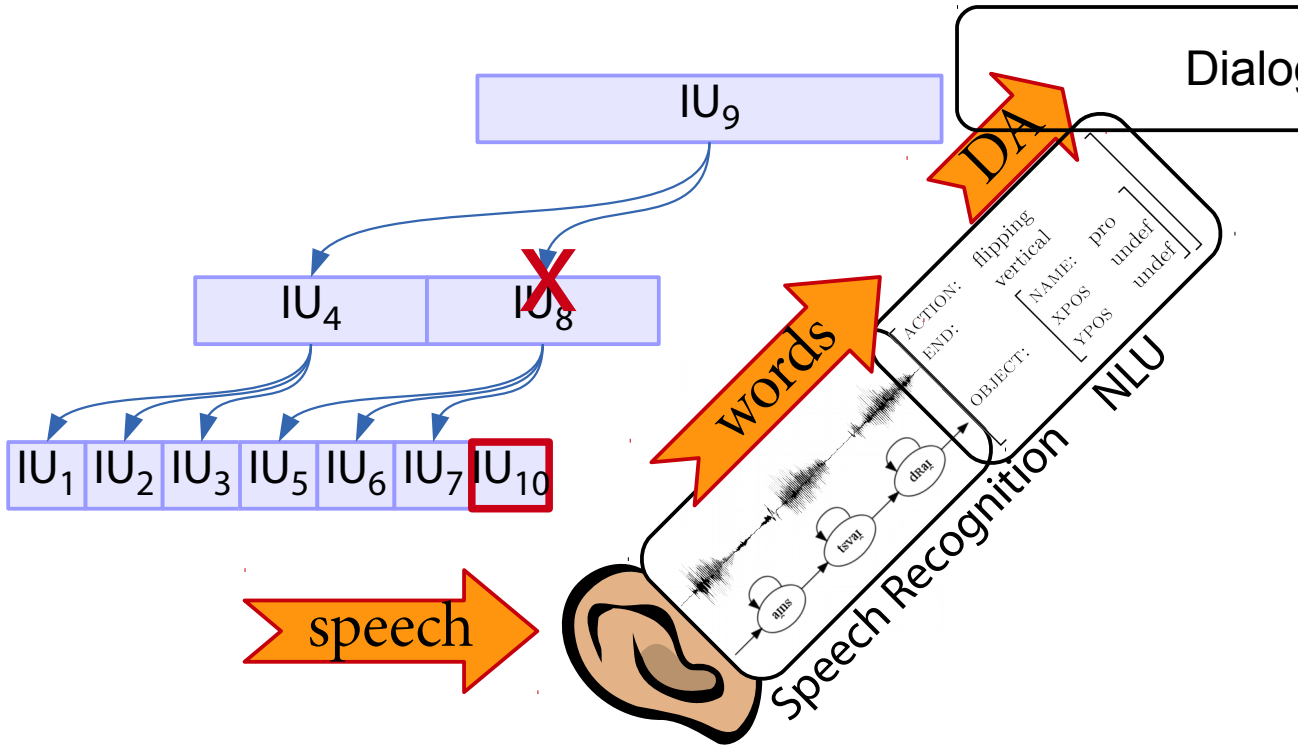
Input Pipeline

- different IU types on different levels
  to denote different kinds of information, e.g.
  - DAs
  - words
  - phonemes



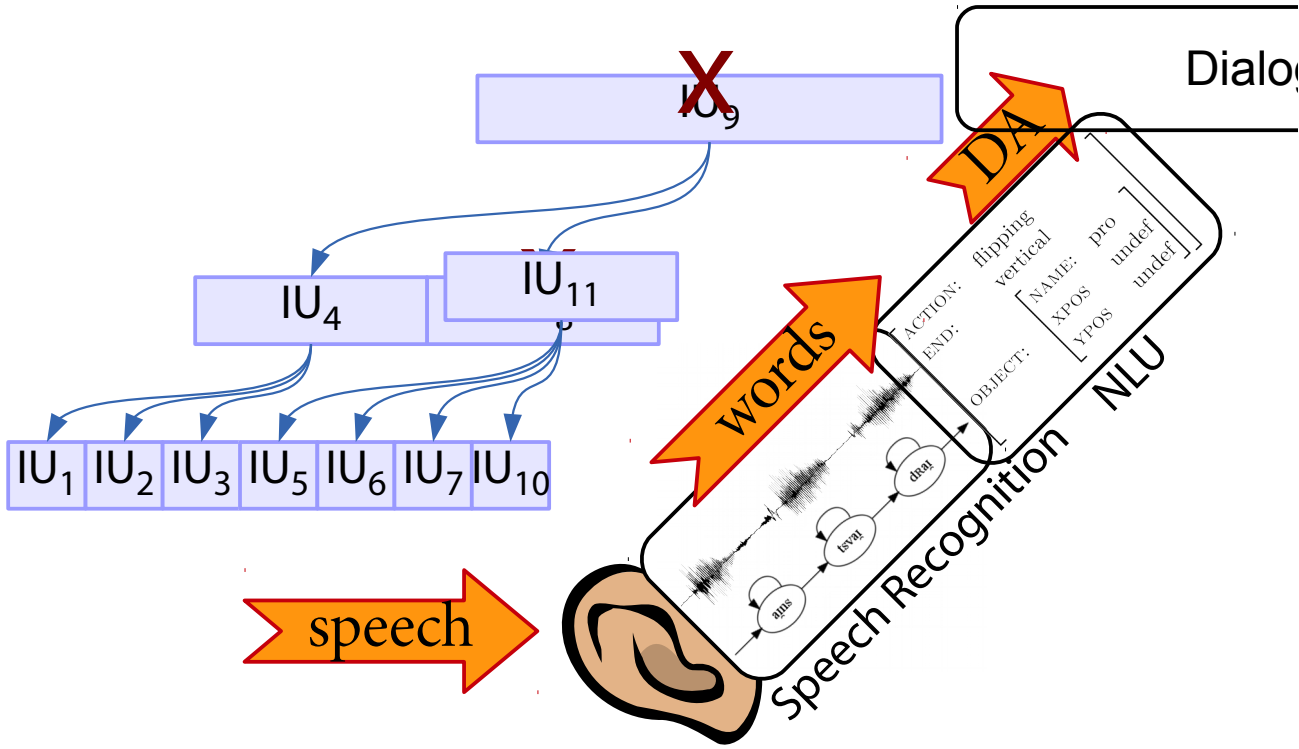
edits as a result of belief changes

- belief changes lead to changes in the network
  - a new frame arrives
  - the word hypothesis is revoked …



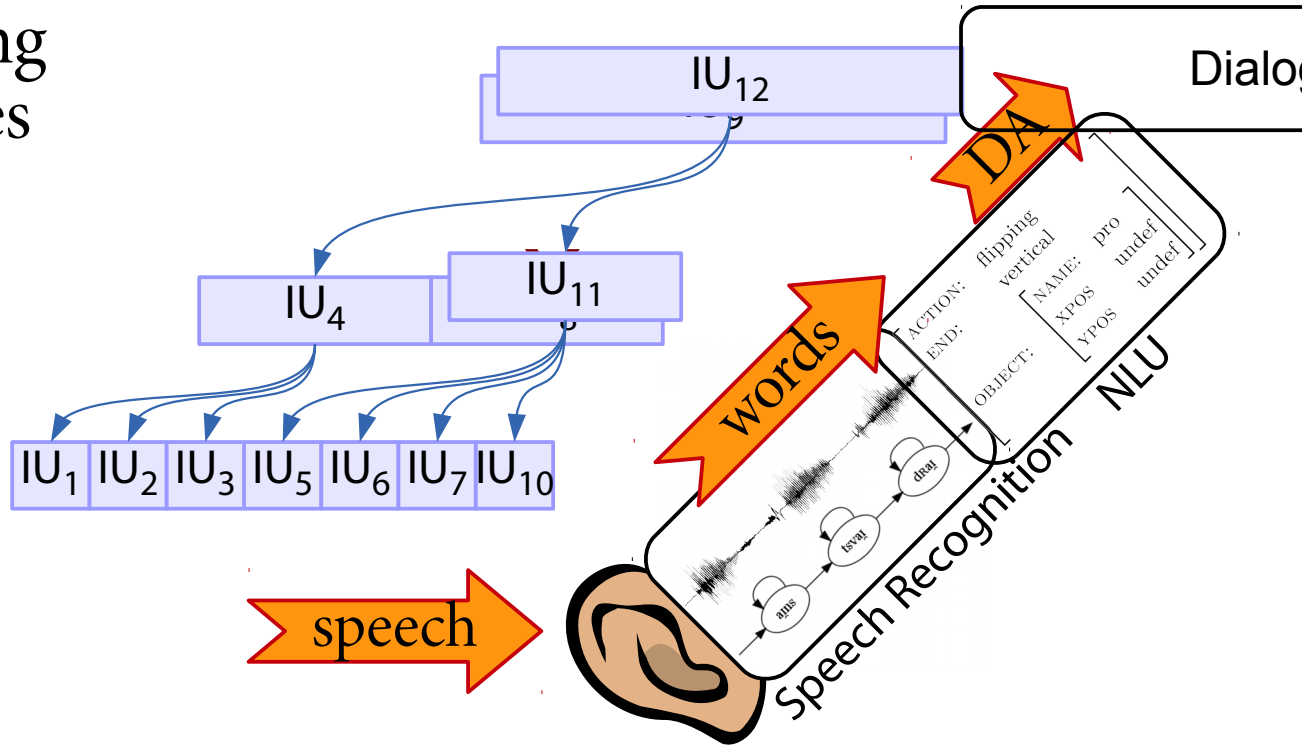
edits as a result of belief changes

- belief changes lead to changes in the network
  - a new frame arrives
  - the word hypothesis is revoked and replaced by a different one



edits as a result of belief changes

- belief changes lead to changes in the network
  - changes trickle up in the system
  - higher-level reasoning might lead to changes trickling down



IU Data Model

- Incremental Units (IUs)
  - encapsulate minimal amounts of information at the current level of abstraction (phones, words, ideas, …)
  - linked to other units on the *same level* to form hypotheses
  - linked to units they are based on to track dependencies
  - network of units stores information states
- Updates to the network reflect changes in understanding:
  - add units when new information becomes available
  - *revoke* units if they turned out to be wrong
  - notify about degree of commitment/certainty to a unit

Schlangen & Skantze (2009, 2011)

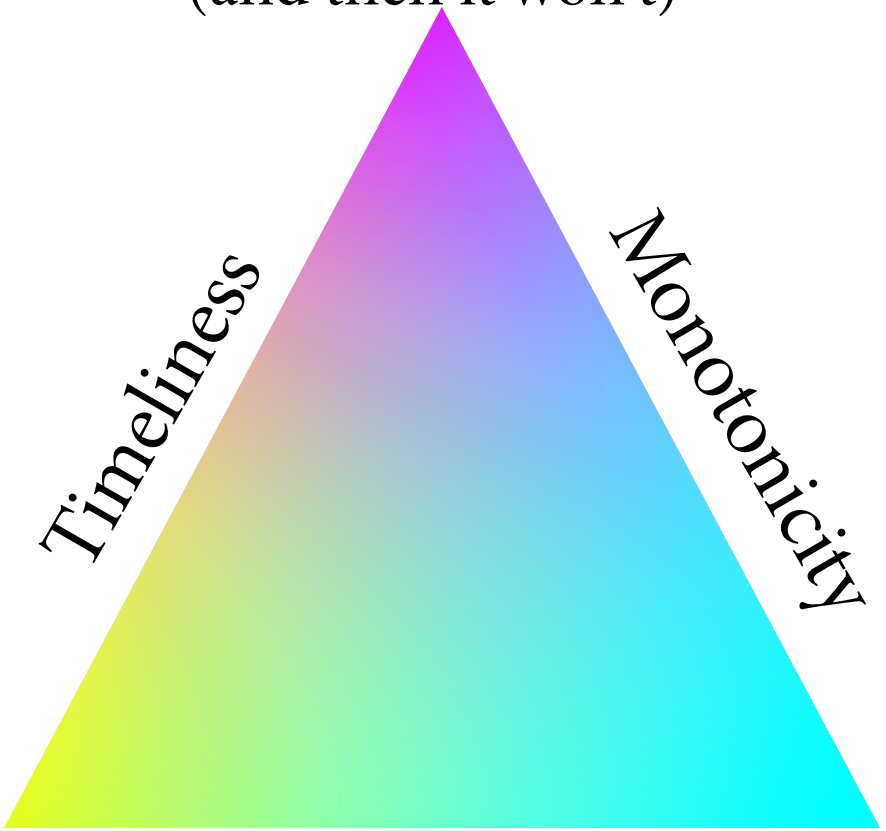
Properties of Incremental Processors

Trade-off on different dimensions:

- **Timeliness**: When you produce output?
- **Accuracy**: How good is your output?
- **Monotonicity**: Commit to the output you made?
  - if you don't commit: aspects of when/how you change your mind

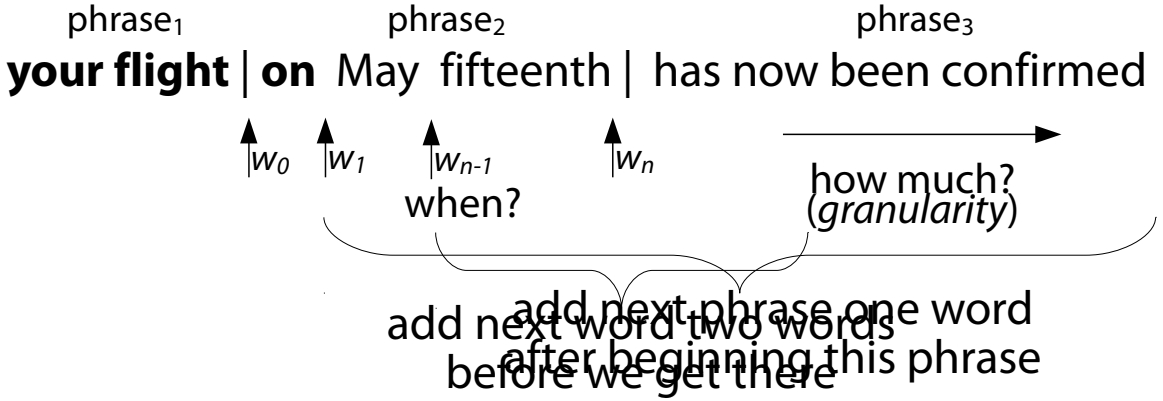
➔ we'll talk about how to evaluate these aspects tomorrow

Properties by Example



Incremental Processing: Important Concepts

- **Lookahead:** the amount of context into the future that a processor needs in order to produce (reasonable) output
- **Granularity:** the size of input that is added at a time

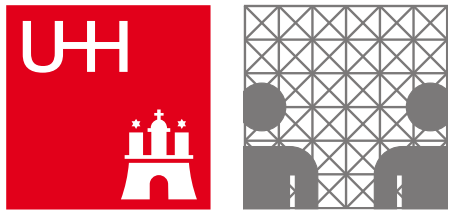


- both lower lookahead and finer granularity help to reduce processing delays

➔ we'll talk about these aspects on Thursday

Finally: Conclusions

- Incremental processing:
  - allows to generate output before input is complete
- monotonicity vs. non-monotonicity
  - neither is ideal;
  - be *as monotonic as possible* and *non-monotonic if necessary*
- IU framework
  - manages non-monotonous incremental processing
  - smallest units are interconnected
  - changes to the network reflect change in belief state
    - monotonic operation: add nodes to the network
    - non-monotonic operation: remove/change nodes in the network



Thank you.

baumann@informatik.uni-hamburg.de
get the code at inprotk.sf.net.

page intentionally left blank

Desired Learning Outcomes

- students understand that incremental processing means to start generating output before input is complete and know the IU model for keeping track of processing dependencies
- students understand that incremental processing comes at additional costs in terms of processing overhead and system complexity
- students know the difference between monotonous and non-monotonous incrementality and understand why the latter is almost always required
- students are aware that incremental processing alters the interaction and may not only solve old problems but also incur new problems