

# An Experiment on Indirect Anaphors in a Programming Language

Sebastian Lohmeier  
Technische Universität Berlin  
sl@monochromata.de

55. StuTS, May 29, 2014

**What? What? Why? Why?**

**Earlier work: Garrod and Terras (2000)**

**Experimental procedure**

**Apparatus**

**Experimental design**

**Hypotheses**

**Questions**

**References**

# What are indirect anaphors?

- 1 Keith was giving a lecture in London.
- 2a He was **taking his car** there overnight.
- 2b He was **driving** there overnight.
- 3 **The car** had recently been overhauled.
- 4 Did Keith go to London by car/train?

(Garrod and Sanford, 1982)

- ▶ An indirect anaphor refers to a referent that is related but not identical to the referent of its previously mentioned anchor.
- ▶ The relation (e.g. thematic role, part, ...) is retrieved from long-term memory instead of being expressed in the text.

# What questions on indirect anaphors are practically relevant ...

... for modelling the comprehension of indirect anaphors?

- ▶ How distant can anchor (e.g. driving) and indirect anaphor (e.g. the car) be?
- ▶ How hard is it for readers to understand indirect anaphors?
- ▶ What does an indirect anaphor mean?

# What questions on indirect anaphors are practically relevant ...

... for modelling the comprehension of indirect anaphors?

- ▶ How distant can anchor (e.g. driving) and indirect anaphor (e.g. the car) be?
- ▶ How hard is it for readers to understand indirect anaphors?
- ▶ What does an indirect anaphor mean?

# Why study indirect anaphors using eye tracking?

Instead of e.g. corpus statistics, reading times or fMRI

- ▶ How distant can anchor (e.g. driving) and indirect anaphor (e.g. the car) be?
- ▶ How hard is it for readers to understand indirect anaphors?
- ▶ What does an indirect anaphor mean?

# Why study indirect anaphors in programming?

- ▶ Sitting in front a computer is ecologically valid for programmers
- ▶ Use knowledge encoded in source code for modelling
- ▶ To implement a cognitive computer model that predicts comprehensibility of indirect anaphors
- ▶ Programming with indirect anaphors might be “better”

# What? What? Why? Why?

- ▶ I want to use eye tracking as an on-line measure of how hard it is to comprehend indirect anaphors in source code.
- ▶ Programming languages will be used to have a “complete” knowledge representation in a computer model.



## Earlier work: Garrod and Terras (2000)

- ▶ eye tracking of direct vs. indirect anaphors
- ▶ 48 subjects, 24 texts (6 per subject), 36 filler texts
- ▶ dominant targets, e.g. WRITE + PEN:  
The teacher ...
  - a She was busy writing a letter of complaint to a parent.
  - b She was busy writing a letter of complaint with a pen.... and the pen dropped on the floor.
- ▶ non-dominant targets, e.g. WRITE + CHALK:  
The teacher ...
  - a She was busy writing an exercise on the blackboard by the door.
  - b She was busy writing an exercise on the blackboard with chalk.... and the chalk dropped on the floor.
- ▶ regression-path duration (initial fixation + regressions to the left) on (in)direct anaphor / following word measured

## Earlier work: Garrod and Terras (2000)

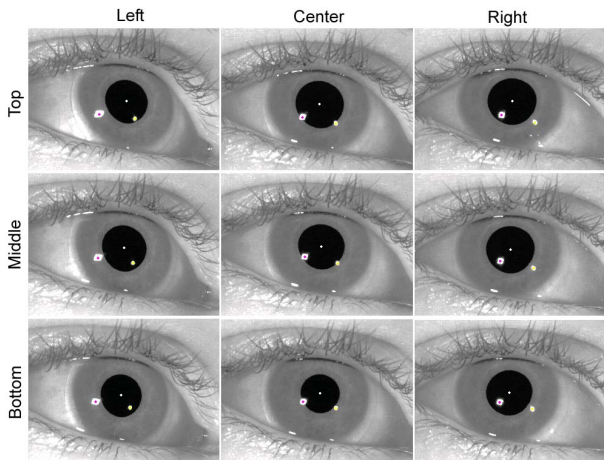
- ▶ dominant targets, e.g. WRITE + PEN:  
The teacher ...
  - a She was busy writing a letter of complaint to a parent.
  - b She was busy writing a letter of complaint with a pen.... and the pen dropped on the floor.
- ▶ non-dominant targets, e.g. WRITE + CHALK:  
The teacher ...
  - a She was busy writing an exercise on the blackboard by the door.
  - b She was busy writing an exercise on the blackboard with chalk.... and the chalk dropped on the floor.
- ▶ no significant difference between dominant instruments (e.g. pen) used as direct vs. indirect anaphor
- ▶ (partly) significant difference of 48ms for non-dominant instrument (e.g. chalk) used as direct vs. indirect anaphor

# Experimental procedure

1. Program comprehension skill questionnaire
2. Introduction to anaphors
3. Anaphors test
4. 40 tasks: read source code, answer yes-no question
5. 20 comprehension questions
6. 5 minutes to write short summary of the code
7. Post-test questions and de-briefing



# Apparatus



## Material #10: control and test condition (simplified)

```
public class LookupLocator {
    public ServiceRegistrar getRegistrar() {
        int timeout = 60 * 1000;
        return getRegistrar(timeout);
    }
}
```

Q: Does LookupLocator provide a getRegistrar() method?

2 materials later ...

## Material #13: control vs. test condition (simplified)

```
LookupLocator locator = new LookupLocator();
ServiceRegistrar registrar =
    locator.getRegistrar();
Log.log(Level.INFO, "Found registrar:
    "+registrar);
```

vs.

```
new LookupLocator();
Log.log(Level.INFO, "Found registrar:
    "+.ServiceRegistrar);
```



1 material later ...

## Material #15: control and test condition (simplified)

```
public class UuidFactory {  
    private static SecureRandom secureRandom  
        = new SecureRandom();  
    public static Uuid generate() {  
        long bits0 = secureRandom.nextLong();  
        long bits1 = secureRandom.nextLong();  
        return new Uuid(bits0, bits1);  
    }  
}
```

Q: Does UuidFactory declare a field of type SecureRandom?

1 material later ...

## Material #17: control vs. test condition (simplified)

```
Uuid topUuid = UuidFactory.generate();
Log.log(Level.INFO, "Uuid: "+topUuid);
spaceProxy = new SpaceProxy(ourRemoteRef,
    topUuid, maxServerQueryTimeout);
```

vs.

```
UuidFactory.generate();
Log.log(Level.INFO, "Uuid: "+.Uuid);
spaceProxy = new SpaceProxy(ourRemoteRef,
    .Uuid, maxServerQueryTimeout);
```

# Experimental design

- ▶ 4 Groups with different material configurations
  1. T:01-20 + C:21-40
  2. C:01-20 + T:21-40
  3. T:21-40 + C:01-20
  4. C:21-40 + C:01-20
- ▶ 4 independent vars
  - ▶ condition (T: with vs. C: without indirect anaphors),
  - ▶ program comprehension skill (high vs. low score),
  - ▶ activation of relation used for indirect anaphors (high or low, manipulated via task sequence),
  - ▶ question type (text-based or inference questions in comprehension questionnaire)
- ▶ 3 dependent vars:
  - ▶ error rate in comprehension questions,
  - ▶ regression-path duration for target word, i.e. word following anaphor
  - ▶ task duration

# Hypotheses

- A** Regression-path duration on target word will be shorter, the more active – i.e. more recently and frequently presented – the underspecified relation.
- B** Regression-path durations on target words in control and test group will be identical for highly activated relations.
- C** For highly activated relations there will be fewer errors in comprehension question for the test condition with indirect anaphors than for the control condition without anaphors.
- D1** Task duration could be lower for the test group with indirect anaphors than for the control group without them because under-specification reduces the amount of text to be read.
- D2** Alternatively, task duration could be higher for the test group than for the control group, if indirect anaphors are generally harder to understand than local variables.

# Questions

What? What? Why? Why?

# References I

- Garrod, S. and Sanford, A. J. (1982). Bridging inferences and the extended domain of reference. In Baddeley, A. and Long, J., editors, *Attention and Performance*, volume XI, pages 331–346. Erlbaum, Hillsdale, NJ.
- Garrod, S. and Terras, M. (2000). The contribution of lexical and situational knowledge to resolving discourse roles: Bonding and resolution. *Journal of Memory and Language*, 42:526–544.