Karlsruher Institut für Technologie

Universal Programmability

How AI Can Help

Walter F. Tichy, Mathias Landhäußer, Sven J. Körner | May 26, 2013

Problem 1

- Almost everyone has one or more programmable devices. There are literally billions of chips out there.
- But only a tiny fraction of the owners of programmable devices can program.
- Programmability, the most fundamental aspect of computers, is inaccessible to almost everyone.

Problem 2

- Shortage of programmers.
  In the EU alone there are estimates that the shortage will grow to 900,000 programmers in ten years.

- Possible fixes:
  - Train more of them.
  - Import them.
  - Select only the best.
  - Match tasks with skill sets.
  - **Deskill programming, so more can program.**
  - (Automate it.)

None of this scales.

A Possible Solution

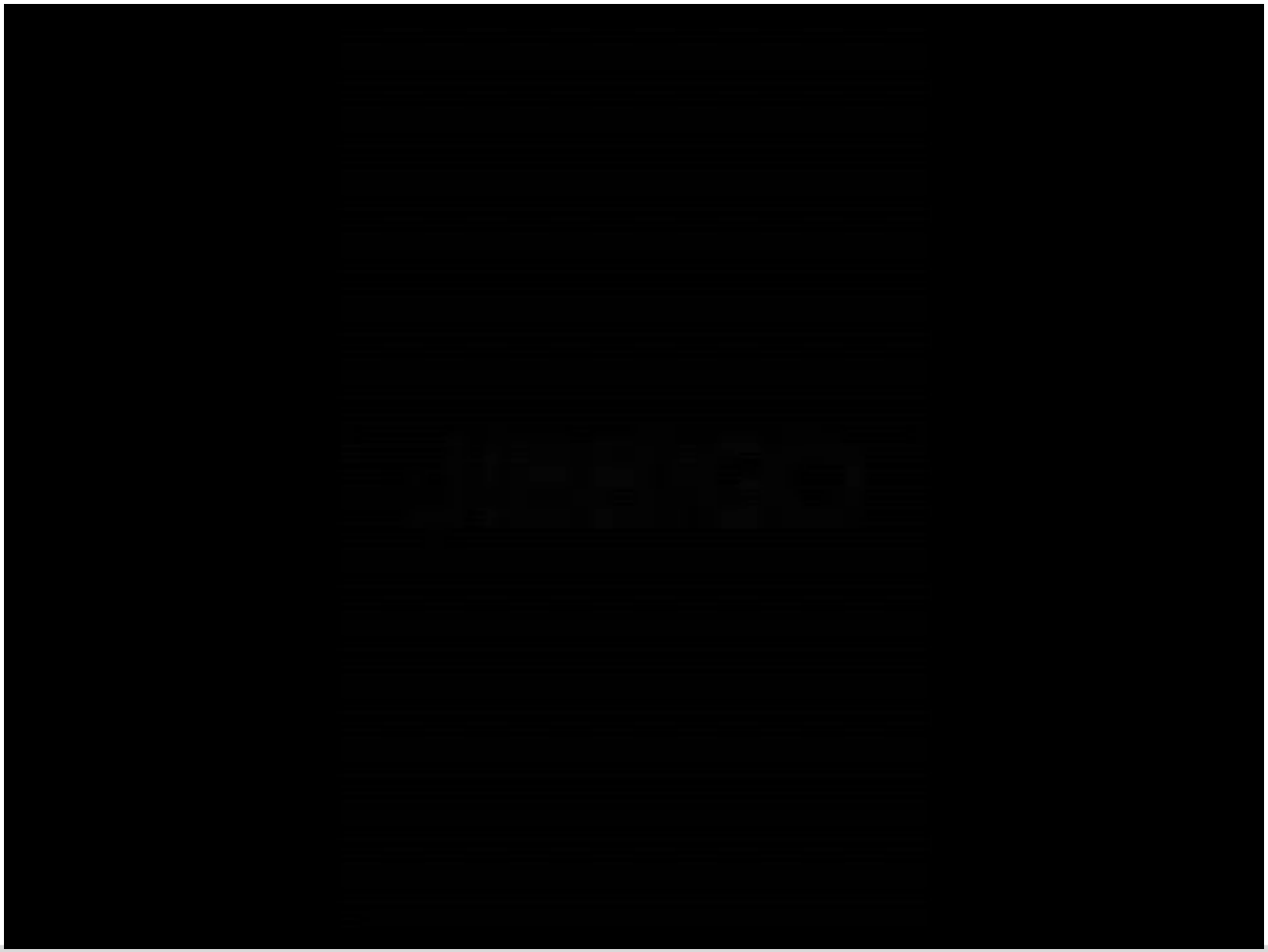
- **Programming in natural language** could be done by anyone.

- People are good at explaining.
  - We teach our children.
  - Recipes, instruction manuals, assembly manuals, books, etc.
  - Requirements elicitation is explaining the domain.

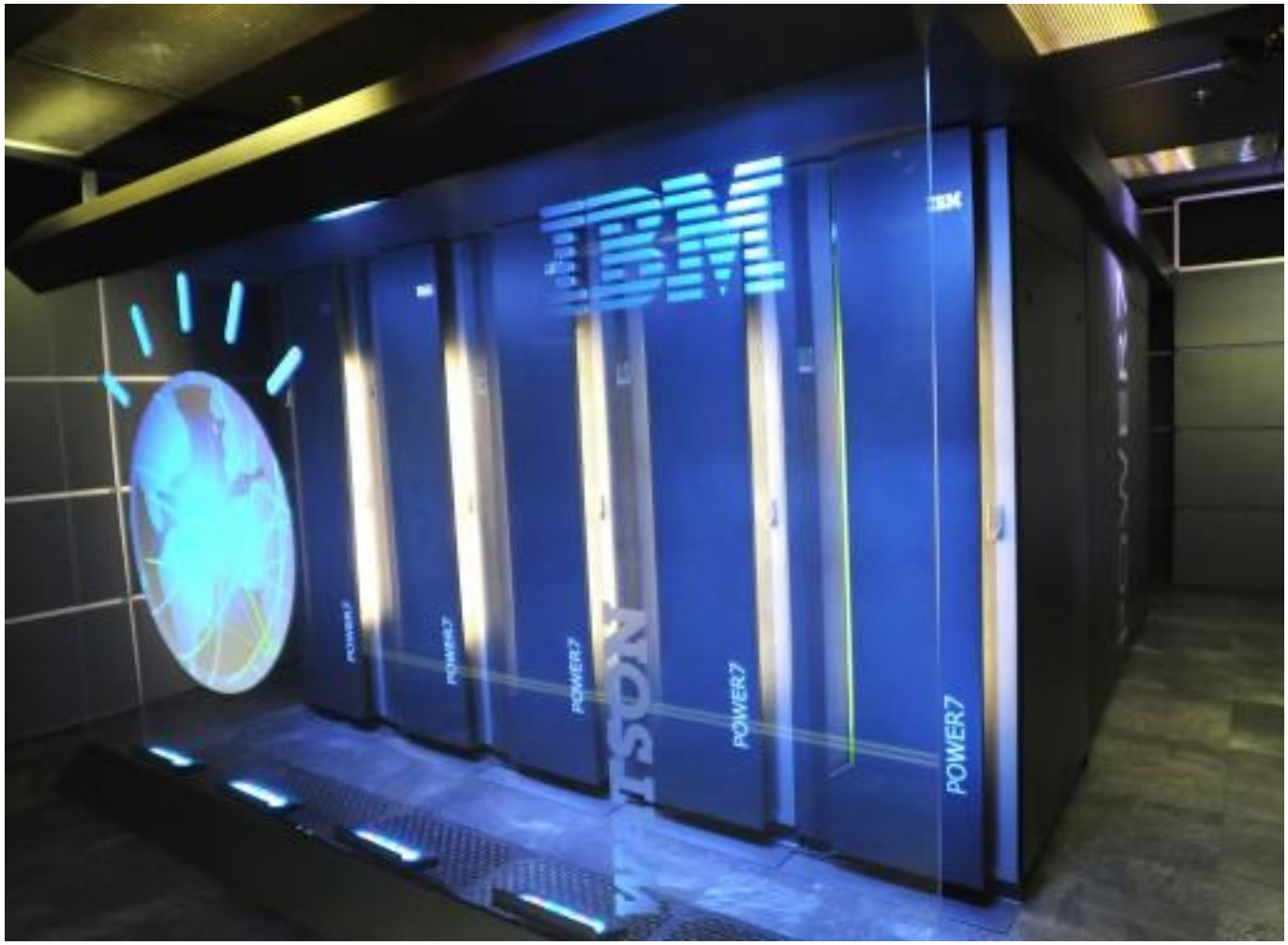
- Some of the challenges of natural language:
  - Ambiguity
  - Implicit context
  - Implicit assumptions
- These challenges need to be met.
  - A smart programmable device can ask for clarification.
  - Programming is not a one-way street.

Why now?
Because Language Technology has made tremendous strides

- Translation devices: Jibbigo



IBM Watson wins Jeopardy



And the Siri Announcement...

Recent Successes of Applying NLP in SE

- Inferring code contracts from method descriptions
- Detecting incorrect code comments
- Programming Excel by example
- Test script generated from NL requirements

- Our own contributions:
  - Automatic specification improvement (remove ambiguities by checking knowledge sources and asking the writer)
  - Generating UML diagrams from textual specifications.
- Round-trip engineering (if you change the class diagram, the requirements text is updated automatically)

Example: IETF 3912 WHOIS Protocol

A **WHOIS server** listens on **TCP port 43** for requests from **WHOIS clients**.

The WHOIS client makes a **text request** to the WHOIS server, then the WHOIS server replies with text content.

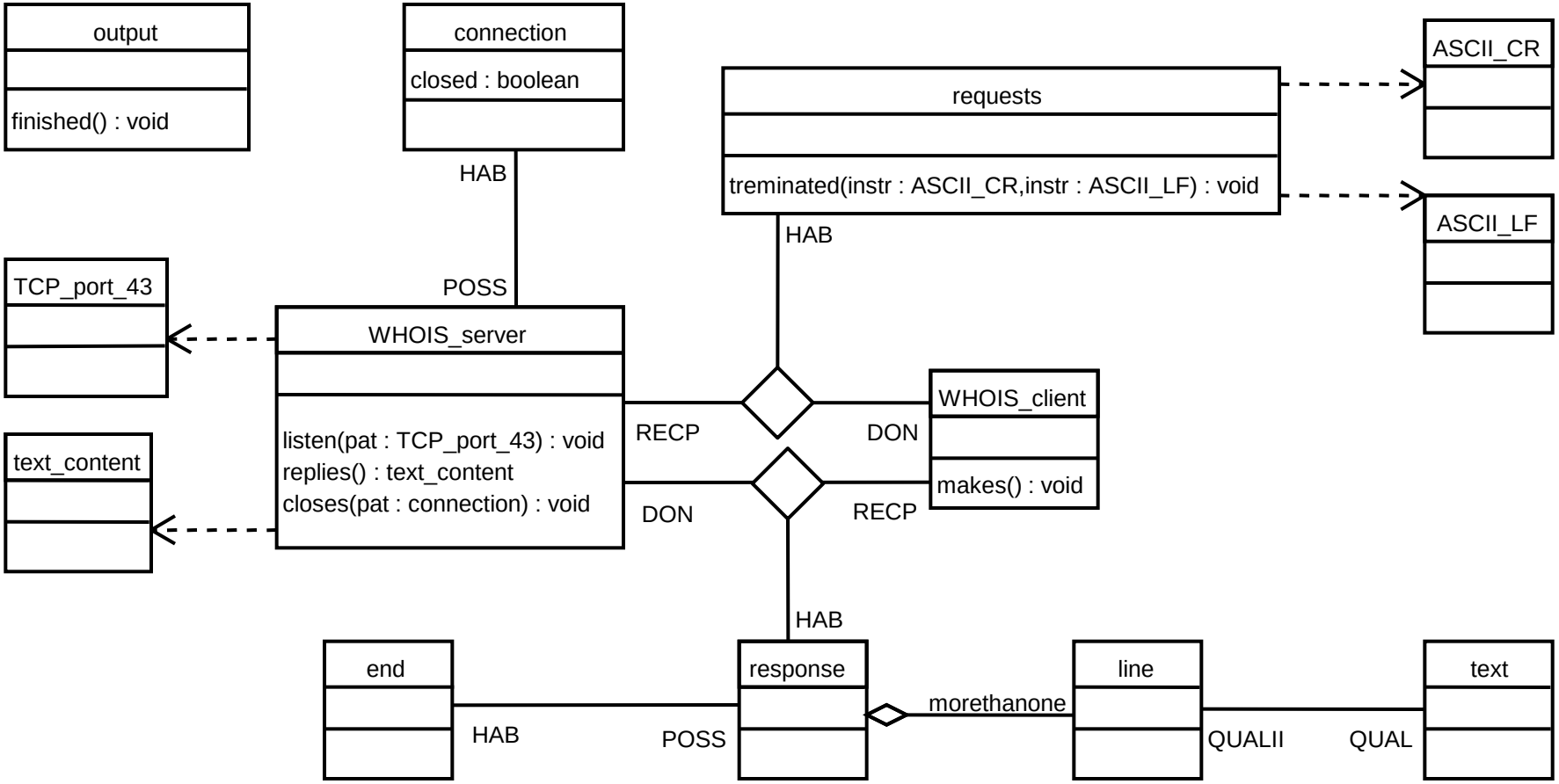
All requests are terminated with ASCII CR and then ASCII LF.

The response might contain more than one line of text, so the presence of **ASCII C**R or **ASCII LF characters** does not indicate the end of the response.

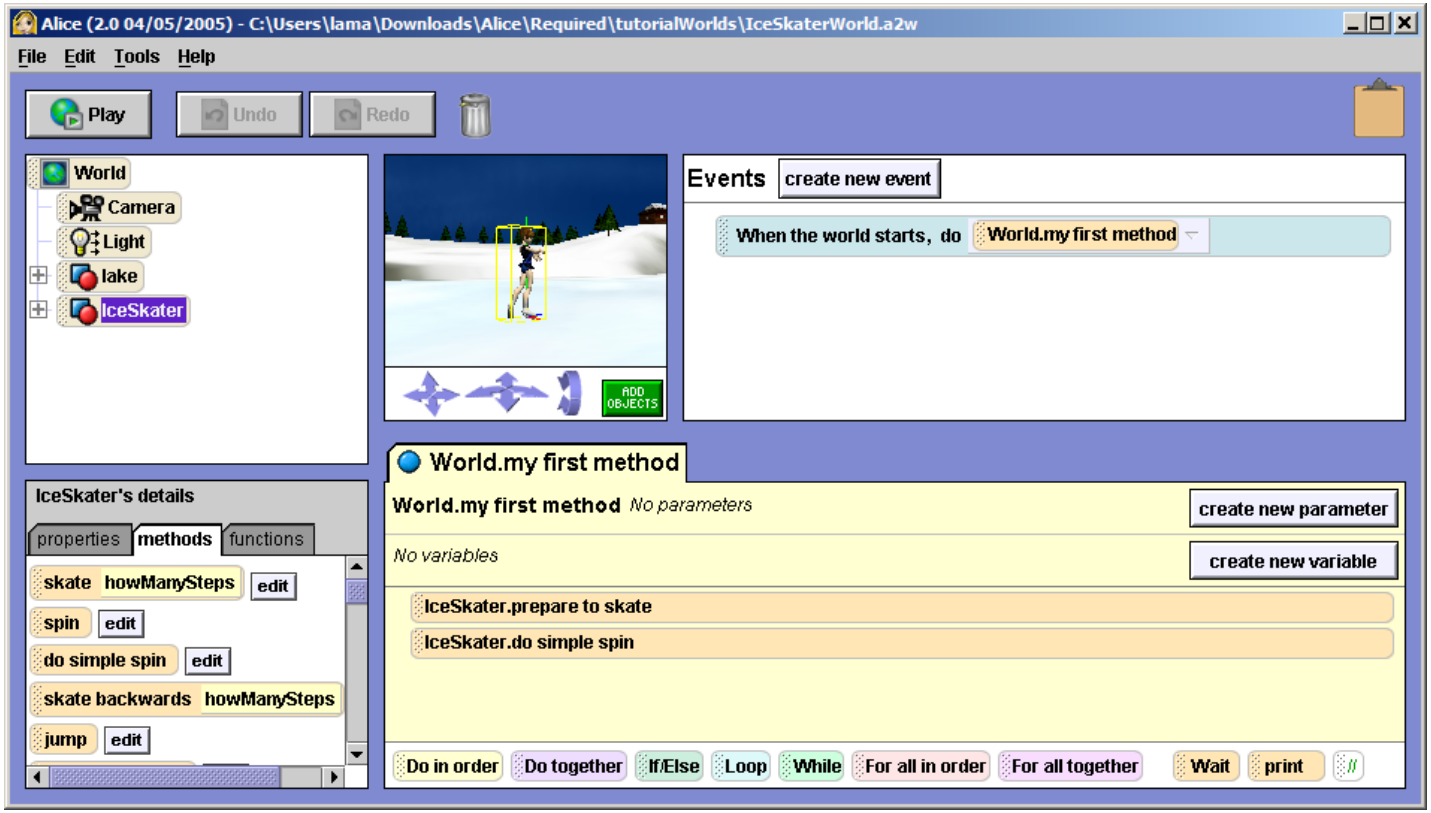
The WHOIS server closes its **connection** as soon as the output is finished.

The closed TCP connection is the indication to the client that the response has been received.

IETF 3912: WHOIS Protocol
Generated Class Diagram



Animation Scripting



- CMU Alice is a framework for building 3D environments, animations, and games.
- It is graphic!
- Program it in English rather than Java.

Animation Scripting: Prerequisites

- Getting to know Alice → Ontology building from Alice.
- What objects are available for programming?
- Determine the features of the objects.
- How can objects and actions be described in natural language? (linking the API to NL)
- Determine method parameters (what if they are not given in the script? or in the wrong type?).
- Handle control flow (e.g. *foreach*).
  How do humans express repetition? They do not spell out a while-loop.

Animation Scripting: Scene Setup

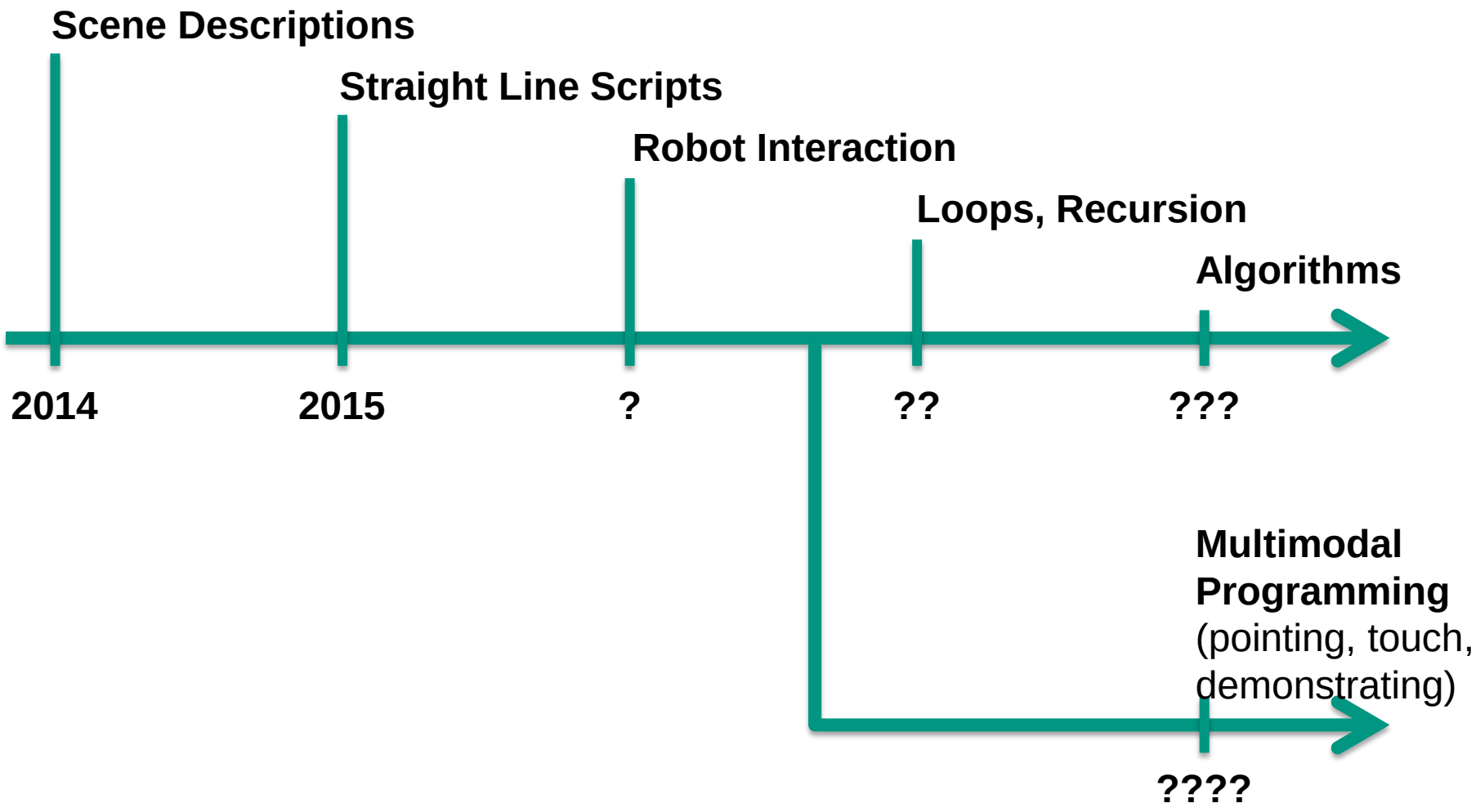
The ground is covered with grass, the sky is blue. In the background on the left hand side there is a palm tree. In the foreground on the left hand side there is a closed Mailbox facing southeast. Right to the mailbox there is a frog facing east. In the foreground on the right hand side there is a bunny facing southwest. In front of the bunny there is a broccoli.

Application Area: Programming Robots

- ARMAR-III is a humanoid robot.
- Designed to be a household help.

- Example tasks:
  - „Set the table for three persons.“
  - „Put cereal and milk on the table.“
  - “Clear the table and put the dishes in the dishwasher.”
- You may want to explain a “subroutine”, such as “clearing the table means pick up all the objects on the table and put them on the counter”
- Why would you want to instruct the robot any other way?

The NL Programming Grand Challenge



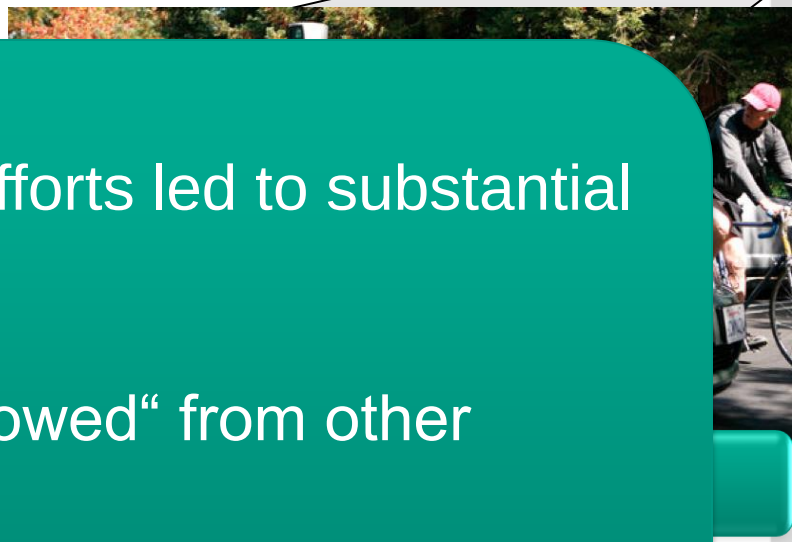
Autonomous Vehicles: DARPA Grand Challenge

Google autonomous vehicle 2010

2007 DARPA
Urban Challenge

2004, 2005 DARPA
Grand Challenge

Autonome Fahrzeuge: DARPA Herausforderungen

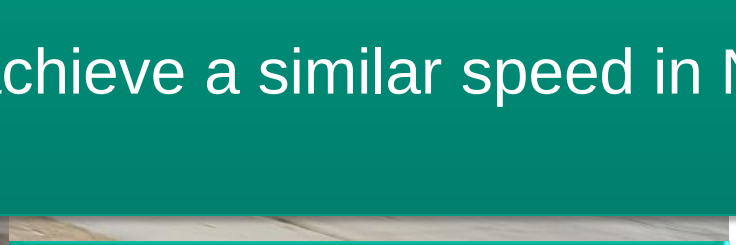


In all these cases, concentrated efforts led to substantial progress in a short time.

Successful techniques were „borrowed“ from other teams and improved.

Could we achieve a similar speed in NL programming?

2004, 2005 DARPA
Grand Challenge



2007 DARPA
Urban Challenge

Grand Challenge: Enabling anyone to program

- Which approaches will enable billions of users to program?
- To unleash the potential of programmable devices, programming should be accessible to anyone.
- Many fields have to work together.