**DIALOGOS: SIMPLE AND EXTENSIBLE DIALOG MODELING**

**ALEXANDER KOLLER, TIMO BAUMANN, ARNE KÖHN**

---

### ABOUT DIALOGOS

An Open Source graphical dialog modeling tool and dialog management framework that scales from teaching to research and production.

### CORE FEATURES

- graphical finite state automaton-based model
- hierarchical subgraphs help manage complexity
- extensible beyond FSAs via Groovy script
- **multi-platform:** installers for Linux, Windows, MacOS
- built-in open-source ASR & TTS via Sphinx-4 and MaryTTS
- build your (basic) dialog system within minutes → more time to focus on what interests you

### SPEECH INPUT/OUTPUT

- simple keyword-based recognition (with support for garbage words)
- simple interpretation via regular expressions
- support for grammar-based speech recognition
- semantic interpretation nodes in grammar (SISR) and storage in dialog variables

**Flexible speech synthesis** from variables or via scripts (including generation of MaryXML).

---

### APPLICATION IN TEACHING

**Teach dialog systems and interaction design** to students of all levels (middle school through university)

Teach **fundamental CS concepts** to school students in an engaging way: FSAs, context-free grammars, control flow, variables, abstraction

### APPLICATION IN RESEARCH

**I. Scaffolding via DialogOS** for the non-research part of dialog, switch to research system in certain states (e.g. multi-modal incremental processing).

**II. DialogOS as frame-based DM in a large, multi-user distributed system** using externally provided ASR, NLU, NLG and TTS; integration as a plugin.

---

### CONNECTIVITY AND EXTENSIBILITY

- Lego Minstorms interface
- SQL for backend integration
- plugin architecture to integrate further input/output/backend functionality via custom types of nodes
- TCP/IP **client interface** for flexible integration with external components
- DialogOS itself can easily be embedded into other (JVM-based) software
- headless operation (embedded or server)
- future extensions: ROS integration, cloud-based ASR & TTS, more flexible NLU (e.g. via SEMPRE NLU)

---

### OPEN-SOURCE DEVELOPMENT

- everything on Github: modularized code, plugins in separate repos, centralized issue tracking
- gradle-based build system, dependencies via jitpack.io
- example repositories with code for how to write custom extensions/plugins

---

**KOLLER@COLI.UNI-SAARLAND.DE**

**TBAUMANN@CS.CMU.EDU**

**KOENH@UNI-HAMBURG.DE**