Smalltalk-based Speech User Interfaces

The SpexKit Platform

Thomas Brey
University of Regensburg (Information Science)
Speech Experts GmbH

SpexKit ?

- **Framework / Environment for (rapid) development of Speech User Interfaces (SUIs)**

- (Unfortunately?) as Add-On to Visual Smalltalk Enterprise (VSE)

Speech User Interfaces

- Useful for automotive or telephony systems
- **Very Complex if user-friendly**

  (because of required *Mixed-Initiative* and *Natural Language* Capabilities)

=> Hard to Design and Implement

SUIs – A Common Misunderstanding:

Although progress has been made by the many companies and research groups, the following evaluation is still valid: „Comfortable and natural communication in a general setting (no constraints on what you can say and how you can say it) is beyond us for now, posing a problem too difficult to solve "(Peacocke&Graf,1990)

Cited in Shneiderman[98] : 332, similar Shneiderman[00], or Walker[02]:*A Visual Rather Than Verbal Future*

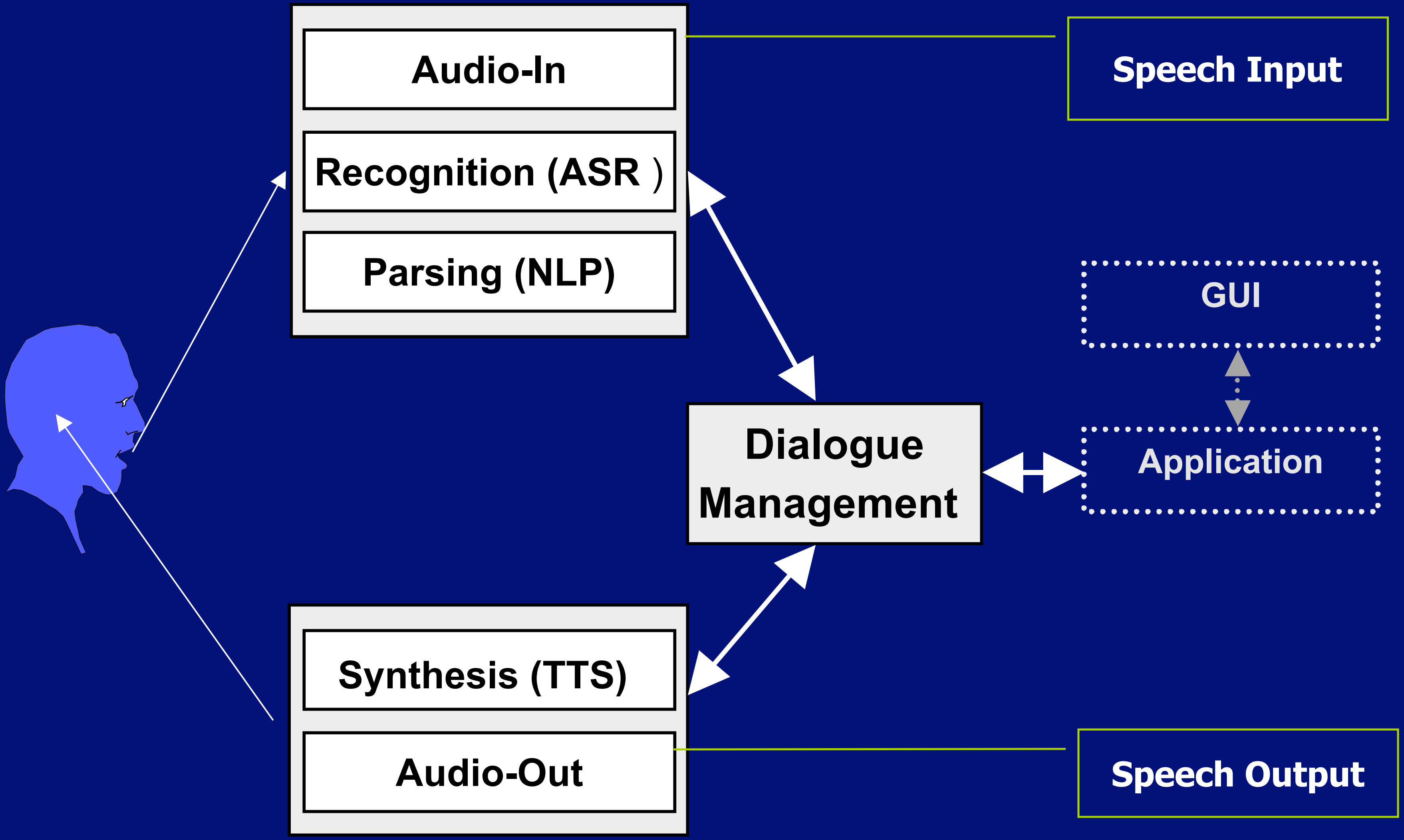
For a given application, a cooperative user with given tasks and a given dialogue history, there always exists a set of constraints for reducing the possibilities of what the user can (reasonably) say.
To identify that set and build a natural and comfortable interface based on these constraints is exactly the task of the interface designer.

Beyond Design:

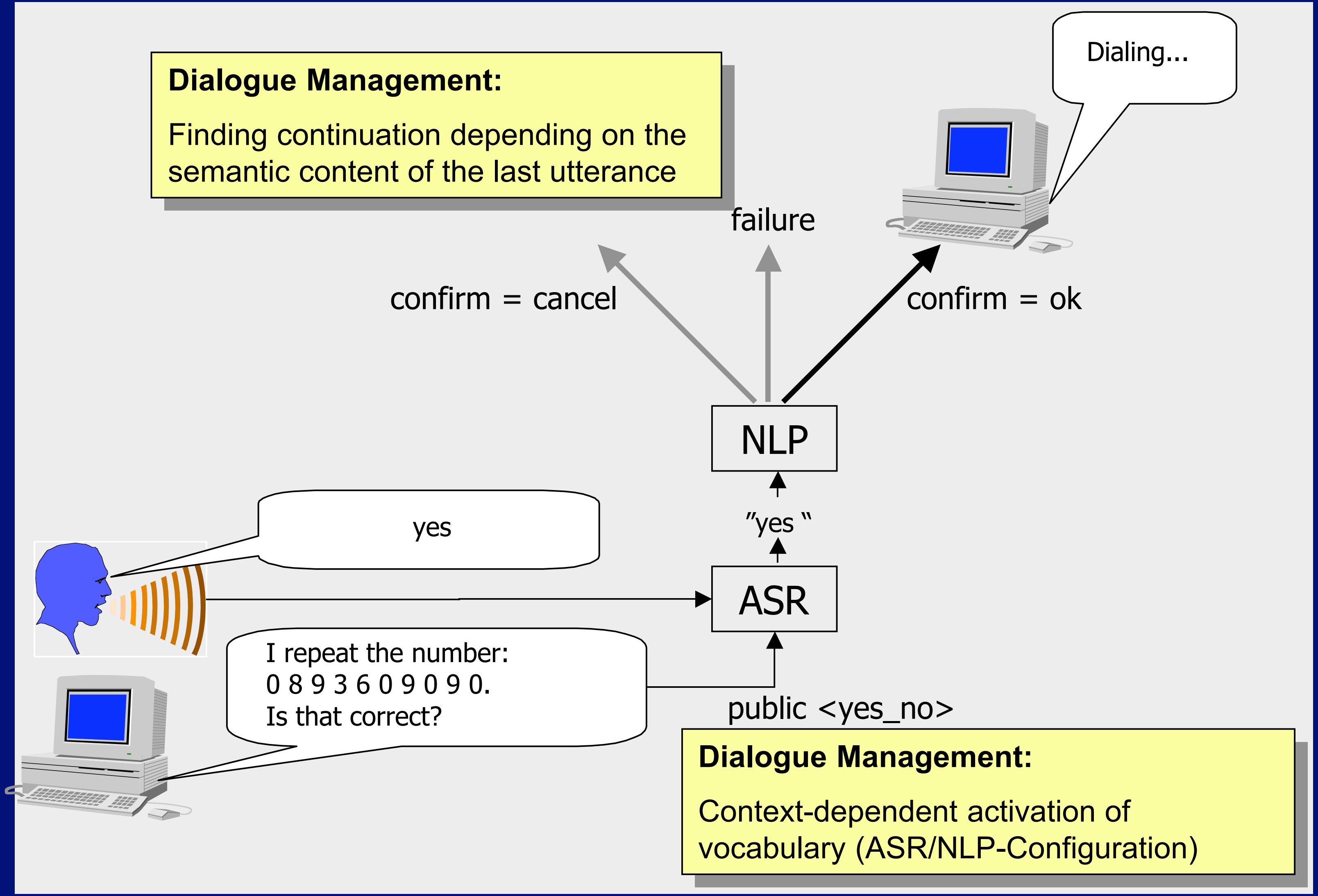
[N]ew modalities will fundamentally change *how* interfaces are developed. For example, to create speech user interfaces today requires learning about vocabularies, parsers, and Hidden-Markov-Models. Tools will be needed that hide all of this complexity and provide an easy-to-use interface to programmers.

Myers et al.[00]:	Past, Present and Future of User Interface Software Tools
In: Carroll (ed):	HCI in the New Millenium

Main Components of SUIs



A simple Use-Case:


Dialogue Management:
Finding continuation depending on the semantic content of the last utterance
Dialing...
failure
confirm = cancel
confirm = ok
NLP
"yes "
yes
ASR
I repeat the number: 0 8 9 3 6 0 9 0 9 0. Is that correct?
public <yes_no>
Dialogue Management:
Context-dependent activation of vocabulary (ASR/NLP-Configuration)

Mixed-Initiative Example

S1	The number please?
U1	Three eight zero four
S2	Three eight zero four, please continue
U2	Five nine one
S3	Nine nine one, please continue
U3	That’s wrong. I said *five nine one*
S4	Correcting, five nine one, please continue
U4	Zero one that’s all
S5	Zero one <Pause> Dialing

Dialogue Management

Tasks:

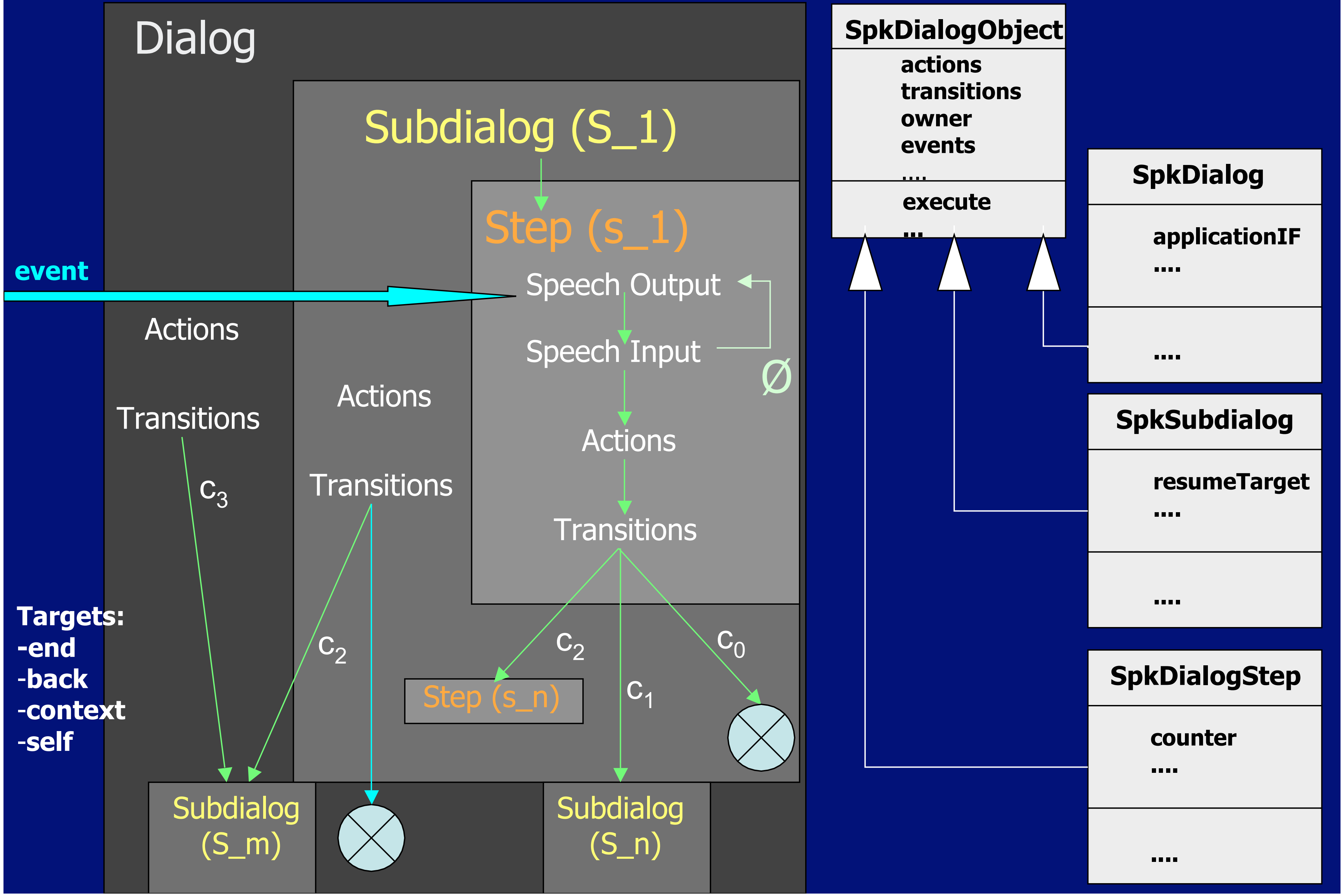
- Configuring Components
- Controlling Interaction

Problems: Complex Interaction, Procedural Parts

SpexKit Solution:

- Augmented Transition Networks with Inheritance
- Distinct Module responsible for procedural parts

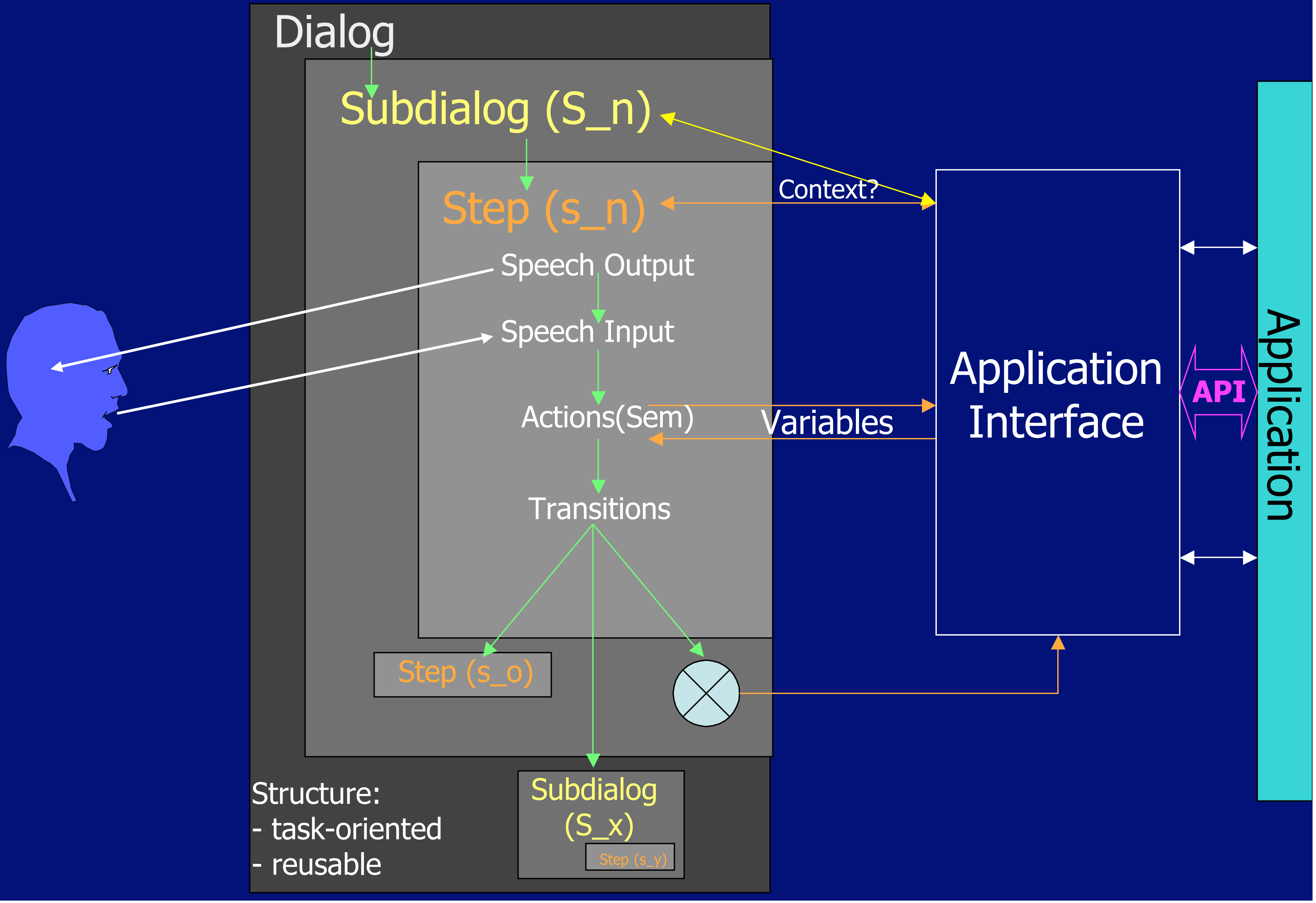
SDML and ST-Counterparts



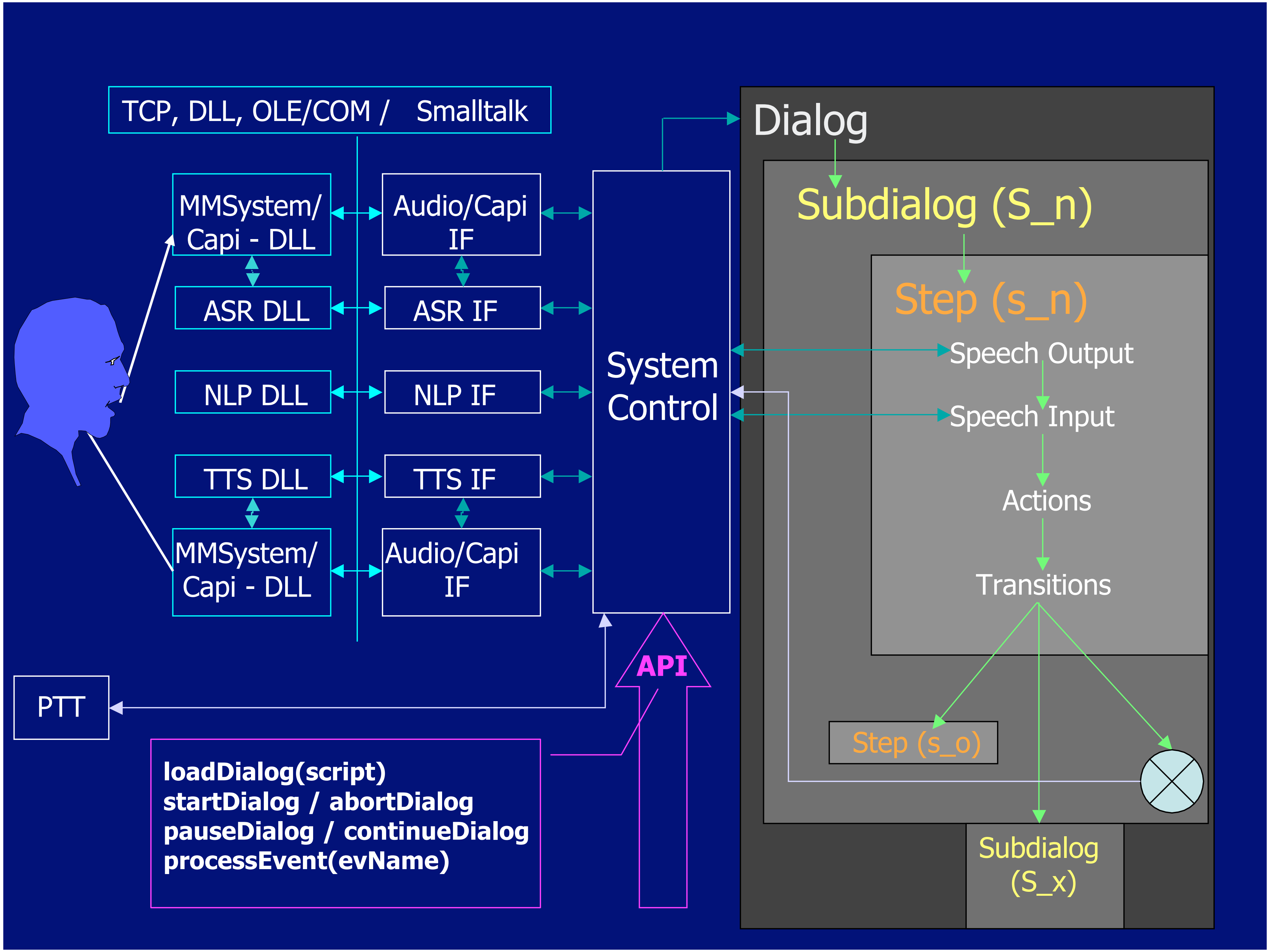
Example SDML-Script

```
<Subdialog Name="S_VoiceDialing">
    <Step Name="s_vd_start">
        <SpeechOutput>
            <PlayList Prompts="p_vd_req_no_1"/>
            <PlayList Prompts="p_sorry p_main_opt p_vd_req_no_2"/>
        </SpeechOutput>
        <SpeechInput Subgrammars="digit_entry"/>
        <ActionSequence Condition="digits = ANY">
            <Action String="store v_lastBlock"/>
            <Action String="request v_number"/>
        </ActionSequence>
        <Transition Condition="digits = ANY" Target="s_vd_continue"/>
        <Transition Condition="digits = ANY, cmd=end" Target="s_vd_confirm"/>
        <Transition Condition="cmd = help" Target="s_vd_num_help"/>
    </Step>
    <Step Name="s_vd_continue">
        <SpeechOutput>
            <PlayList Prompts="v_lastBlock, p_continue"/>
    ...
```

Architecture Pt 1



Structure:
- task-oriented
- reusable

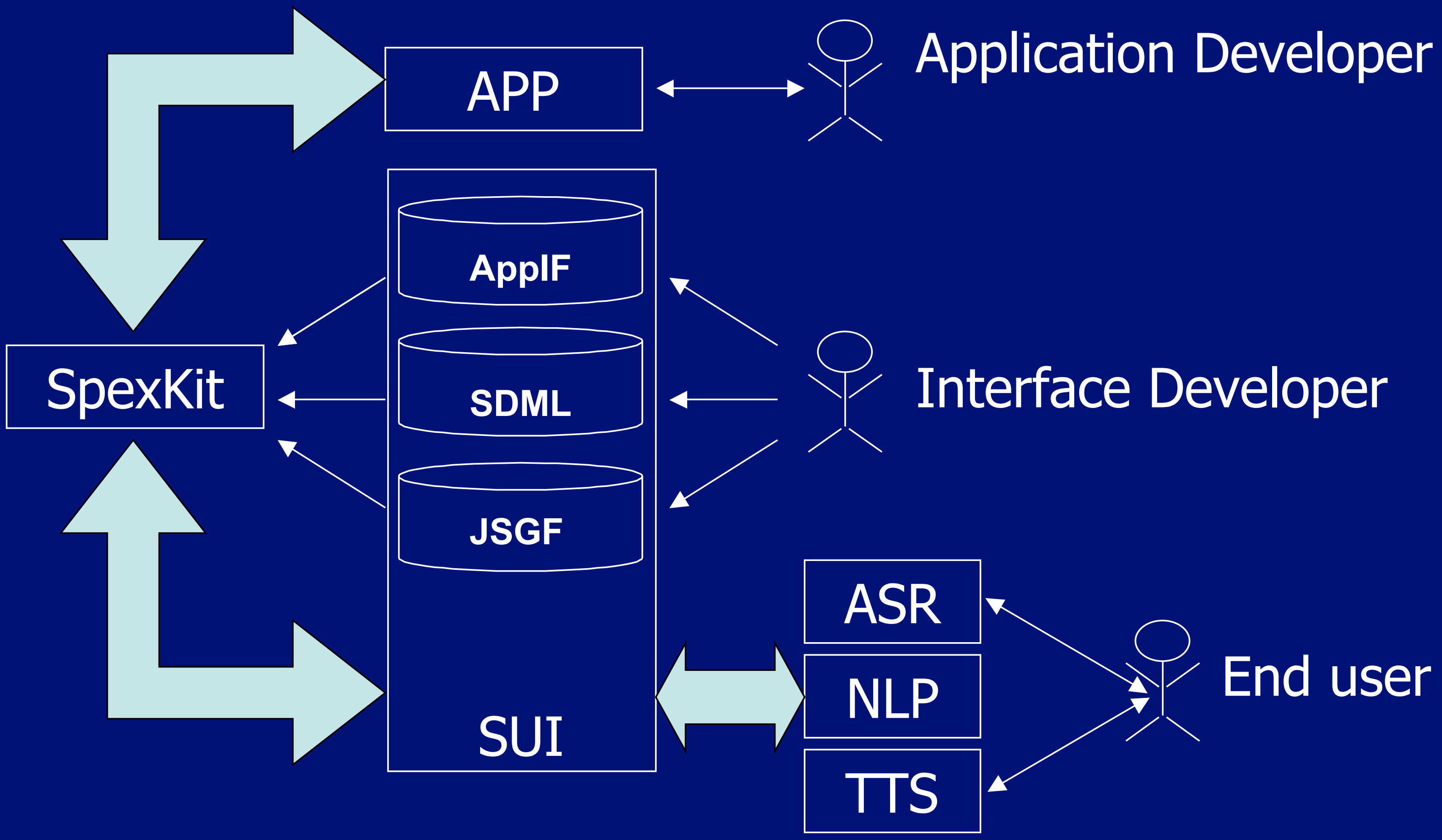


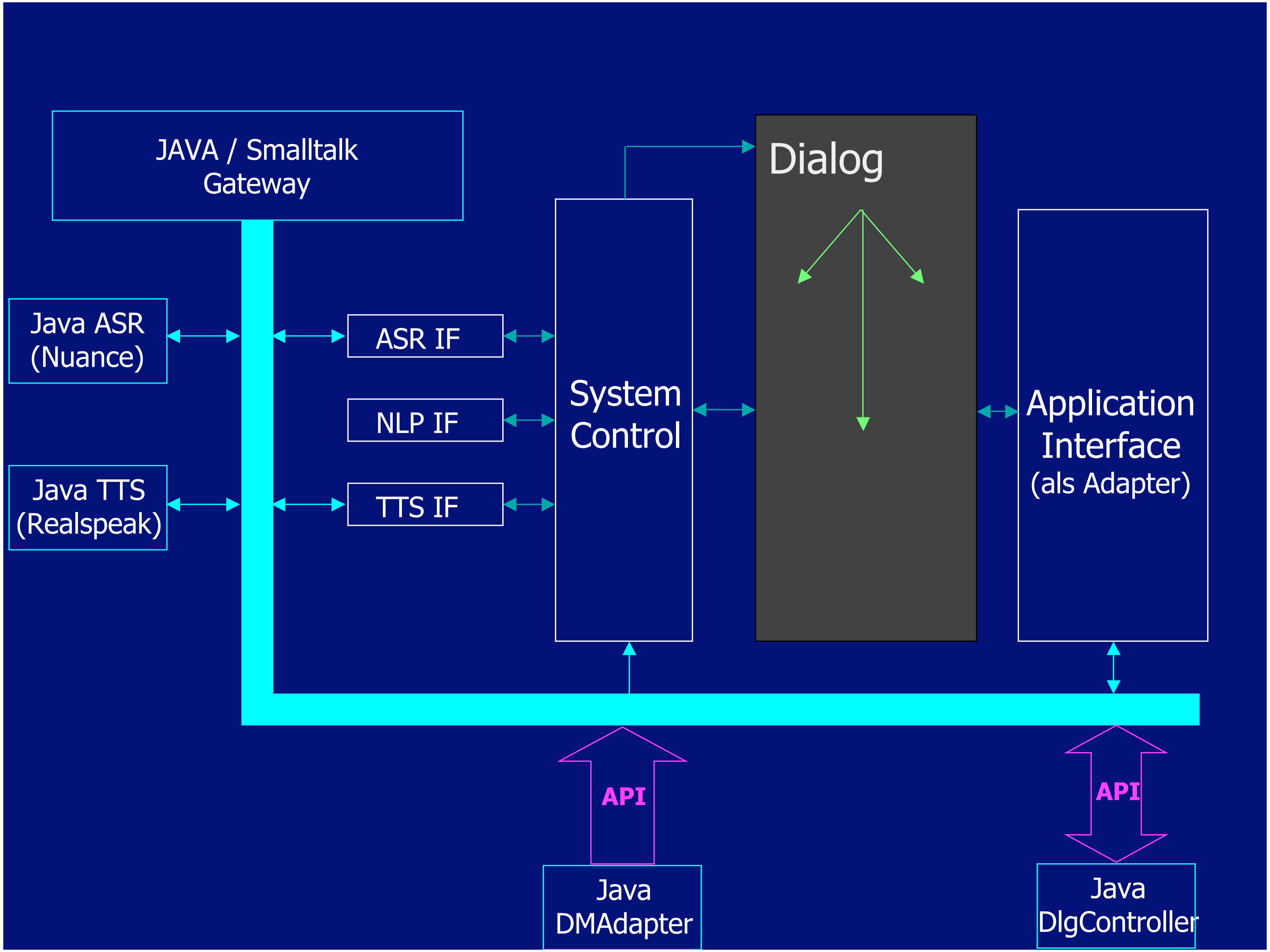
loadDialog(script)
startDialog / abortDialog
pauseDialog / continueDialog
processEvent(evName)

Key Features:

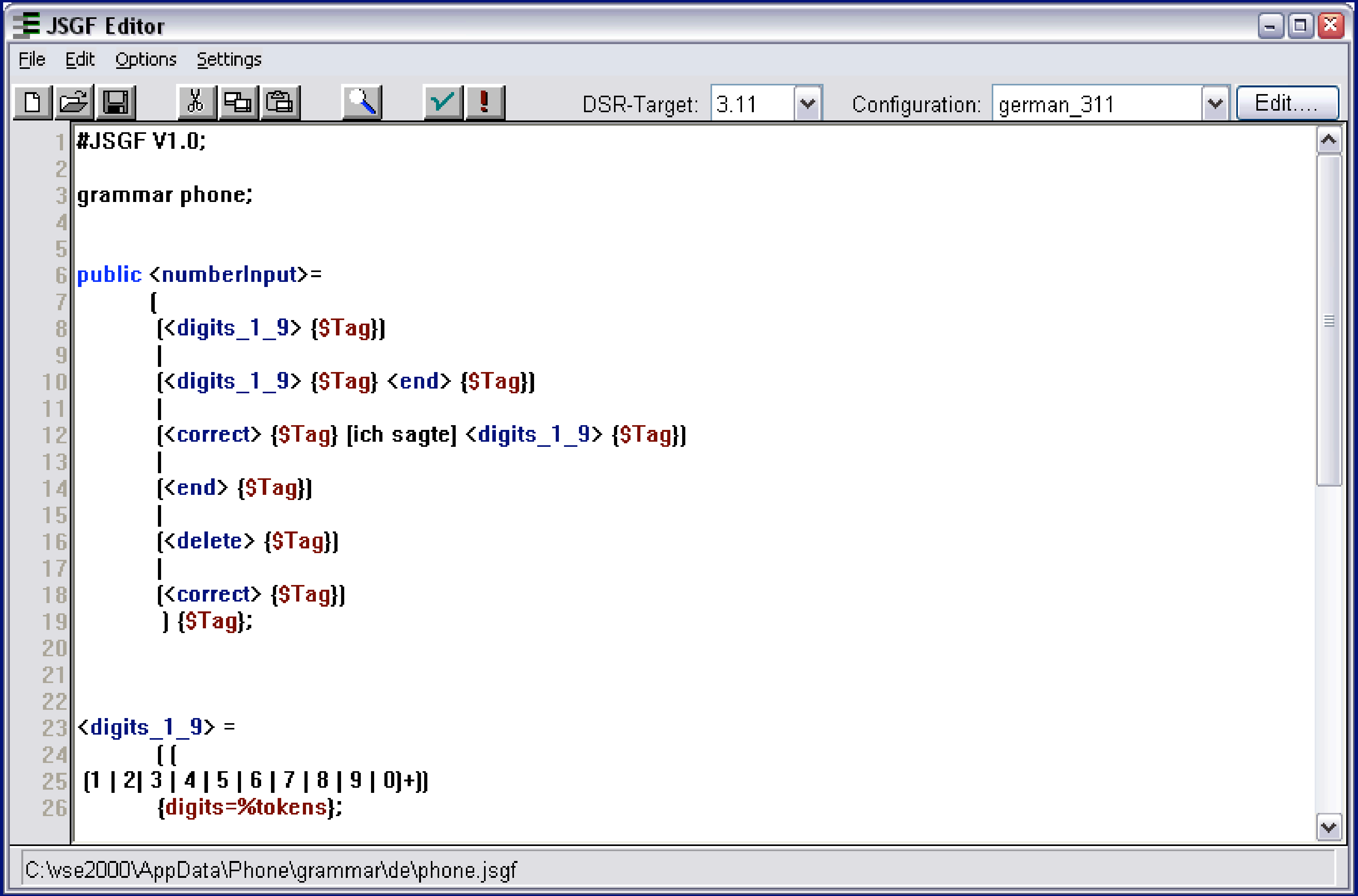
- Generic Framework for the integration of speech technology components (ASR, TTS)
- Existing Interfaces to many commercial components (e.g. Nuance, Temic, SVox, Loquendo)
- Easy to learn and apply scripting language for Dialogue Management (SDML)
- Designed for reducing complexity and maximum usability (at several levels)

Reducing Complexity:




JAVA / Smalltalk Gateway
Dialog
Java ASR (Nuance)
ASR IF
NLP IF
System Control
Application Interface (als Adapter)
Java TTS (Realspeak)
TTS IF
API
API
Java DMAdapter
Java DlgController

Tool Example – Grammar Editor:



```
#JSGF V1.0;

grammar phone;


public <numberInput>=
        (
         (<digits_1_9> {$Tag})
         |
         (<digits_1_9> {$Tag} <end> {$Tag})
         |
         (<correct> {$Tag} [ich sagte] <digits_1_9> {$Tag})
         |
         (<end> {$Tag})
         |
         (<delete> {$Tag})
         |
         (<correct> {$Tag})
          ) {$Tag};



<digits_1_9> =
         ( (
 (1 | 2| 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0)+))
         {digits=%tokens};
```

C:\wse2000\AppData\Phone\grammar\de\phone.jsgf