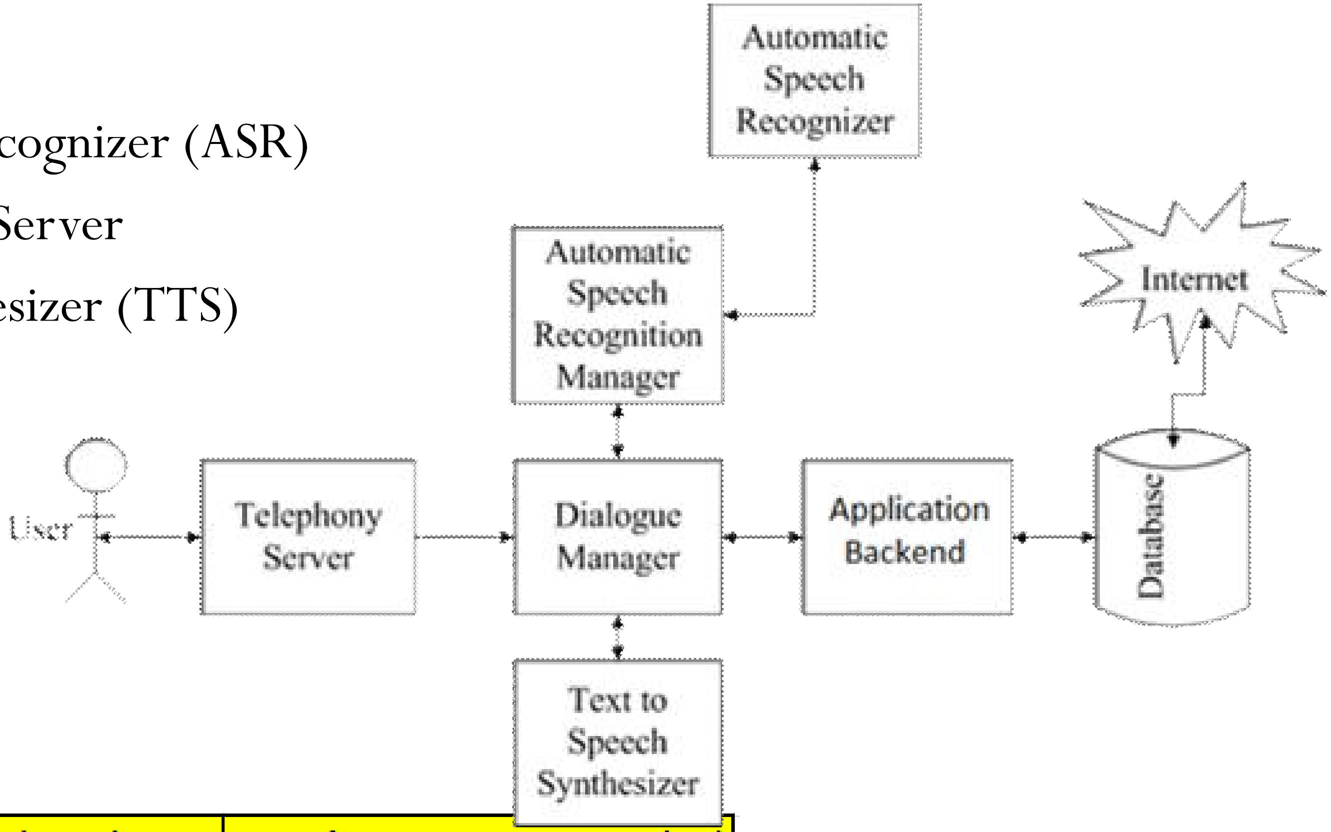
Software Infrastructure for Spoken Dialogue System

Presenter: Aneef Izhar Ul Haq

4th March, 2014	Center for Language Engineering (CLE)

Components of a Spoken Dialogue System

- Audio Telephony Server
- Dialogue Manager
- Automatic Speech Recognizer (ASR)
- Application Backend Server
- Text to Speech Synthesizer (TTS)


Automatic Speech Recognizer
Automatic Speech Recognition Manager
Internet
User
Telephony Server
Dialogue Manager
Application Backend
Database
Text to Speech Synthesizer

Components of a Spoken Dialogue System

- Audio Telephony Server
  - Used to input speech from the user/caller via a telephone line.
  - Also used to playback the synthesized speech to the user.
  - **Linksys Gateway device** is used to route incoming calls on telephone line to the Audio Server.
  - **TrixBox** is a software that is used to communicate between Gateway device and the server.
  - **Asterisk** is the underlying platform of the audio server that is used as a communication application.
- Dialogue Manager
  - The dialogue manager performs the responsibilities of the control of the dialogue.
  - Responsible for taking an appropriate action in case of an ambiguity.
  - Responsible for handling error-events.

Components of a Spoken Dialogue System

- Automatic Speech Recognizer
  - Responsible for decoding the input speech from user into text.
- Application Backend server
  - Provides database for Location and Weather based services.
- Text to Speech Synthesizer
  - Responsible for synthesizing the text form of the dialogue / final output into speech form.

The need of an Infrastructure

- An Infrastructure is required:
  - To manage proper call flow.
  - To provide logging of events.
  - For session management.
  - For handling of multiple calls / sessions.

Architectures of Spoken Dialogue System

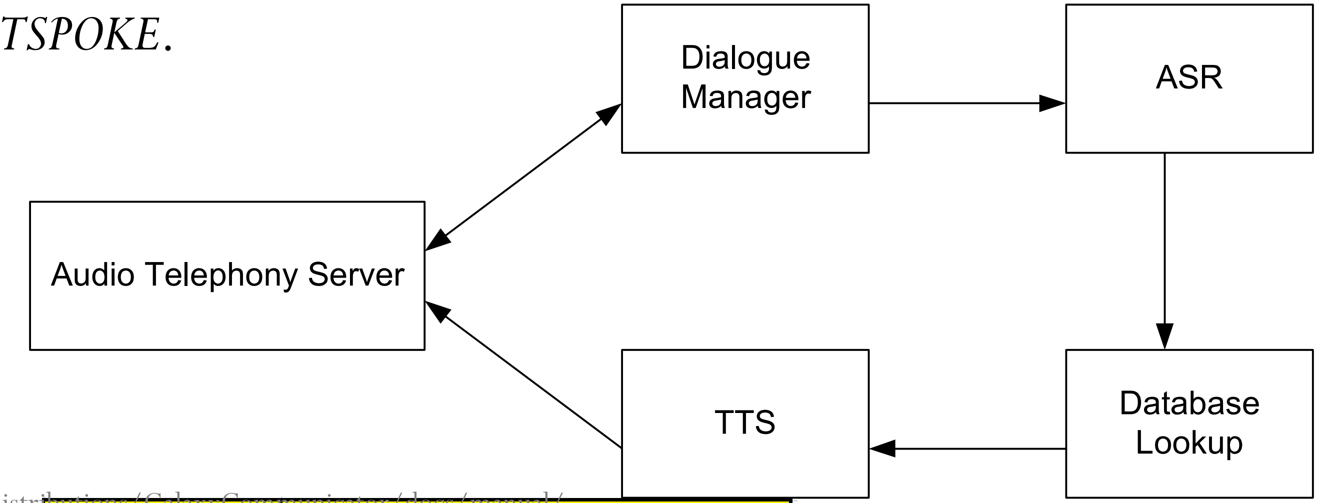
Architectures of Spoken Dialogue Systems can be broadly categorized as:

1. Sequential Architecture
2. Centralized Architecture

Architectures of Spoken Dialogue System

- Sequential Architecture
  - Each individual module communicates directly with the other module forming a pipeline.

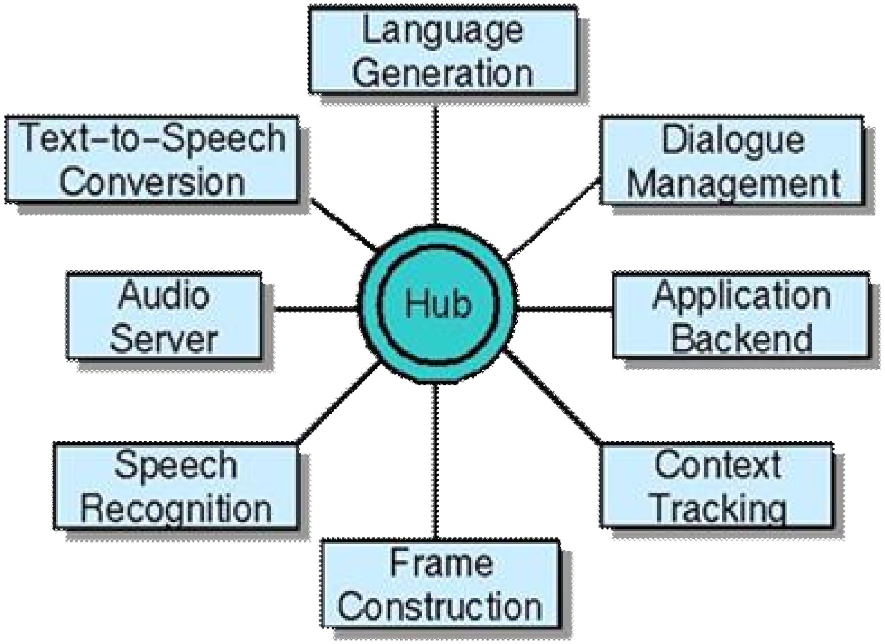
- Systems built using this architecture include *SUNDIAL*, *ITSPOKE*.



http://communicator.sourceforge.net/sites/MITRE/distributions/GalaxyCommunicator/docs/manual/

Architectures of Spoken Dialogue System

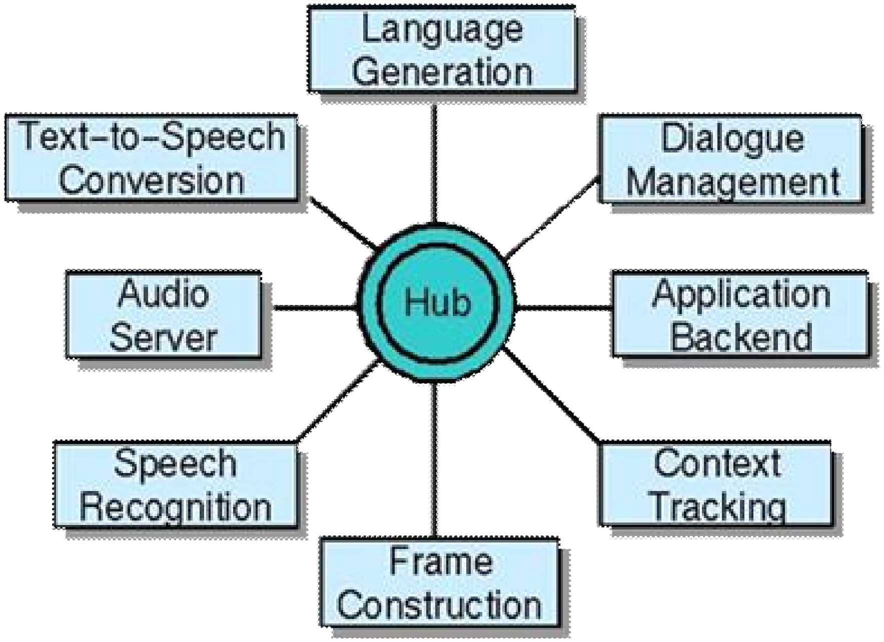
- Centralized Architecture
  - A central module or central communication manager is present which connects all the modules together.
  - All modules interact with each other through this communication manager.
  - Most widely used architectural framework is the *GALAXY* Communicator.
  - *CMUnicator*, *Jupiter*, *Mercury*, *Olympus*, are all based on *GALAXY* Communicator.



http://communicator.sourceforge.net/sites/MITRE/distributions/GalaxyCommunicator/docs/manual/

Galaxy Communicator

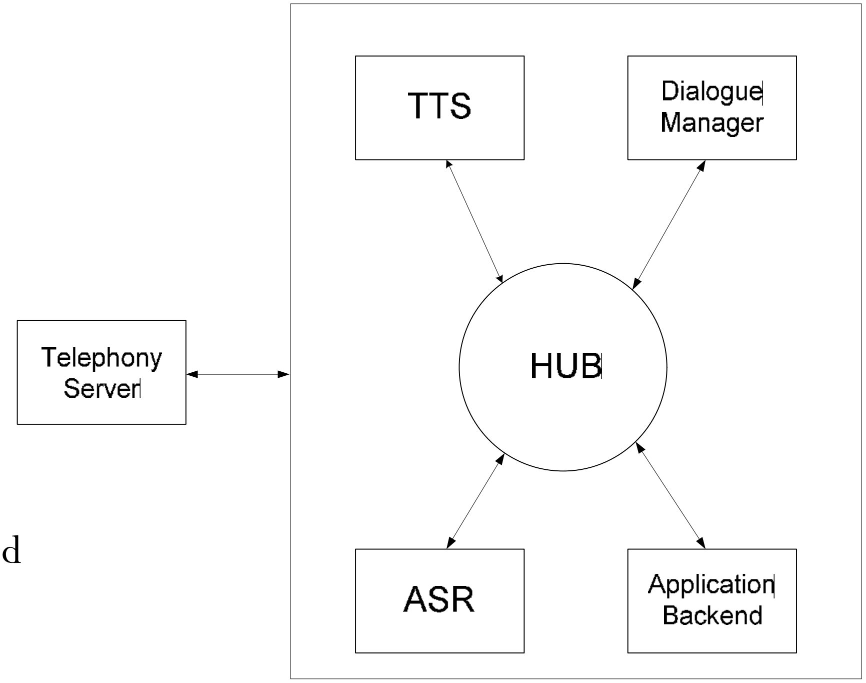
- Open-source architecture for developing new spoken dialogue systems.
- Centralized Architecture.
- Hub and Spoke Infrastructure
- Message based system.



or.sourceforge.net/download/GalaxyCommunicator.html

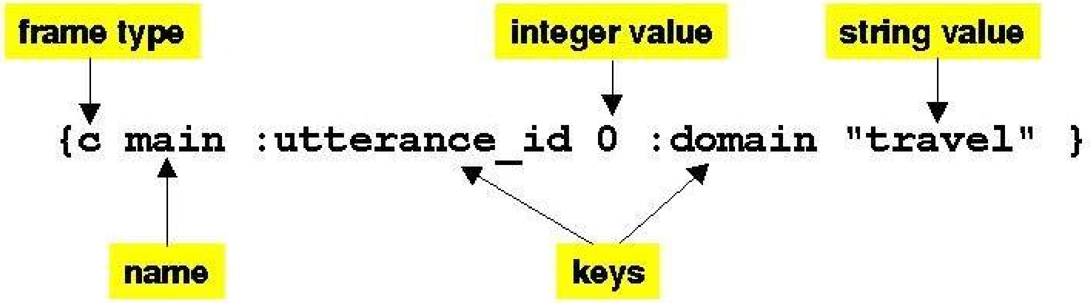
Hub

- Programmed using a high-level scripting language.
  - Script includes
    - List of servers
    - Details about host machine
    - IPs and ports used for communication
    - Set of functions supported by each server
- Hub Programs
  - Sequence of rules that dictate:
    - the functions to be invoked
    - the conditions under which the functions are invoked
    - the servers on which they are invoked
    - the inputs and outputs



Hub

- Communication is in the form of frames
  - A frame consists of
    - Names of servers and/or functions
    - Set of pair of keys
    - Associated values for keys



```
{c main :utterance_id 0 :domain "travel" }
```

Communication Startup

- First the servers are started on their respective ports
- Hub loads the routing rules and Hub programs
- Hub communicates with the servers
- User commences a session using a telephone
- Communication between Telephony server and Galaxy Communicator takes place using Socket connections

Sample Dialogue for Prototype System

For the Location based Spoken Dialogue system, consider a sample dialogue:

System:

’’مرکز تحقیقات و لسانیات میں خوش آمدید! پہلی بیپ کے بعد اپنی موجودہ جگہ کا نام بتایے جبکہ دوسری

بیپ کے بعد اپنی مطلوبہ جگہ کا نام بتایے‘‘

User:

ماڈل ٹاوَن

گوالمنڈی

System:

’’ماڈل ٹاوَن تا گوالمنڈی کا راستہ یہ ہے۔۔ شہید چوک سے دائیں مڑ جائیں۔۔۔۔۔۔‘‘

Sample Dialogue for Prototype System

System:

"Hello and Welcome to Center for Language Engineering. Please record your current location after the first beep tone and your destination location after the second beep tone"

User:

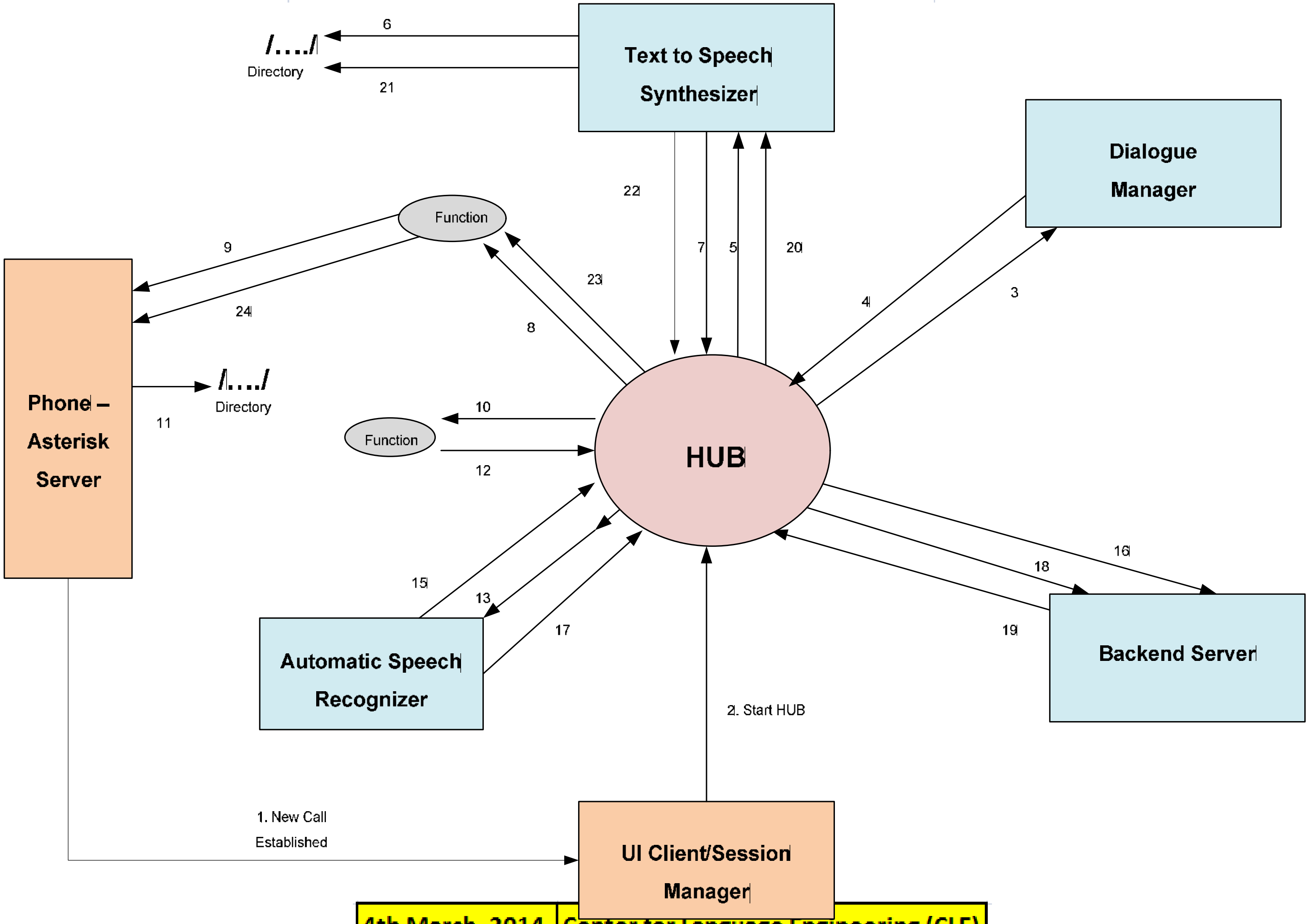
Model Town

Gawal Mandi

System:

"From Model Town Lahore to Gawal Mandi Lahore . . Distance is 9 km long. Turn right from Shaheed Chowk . . . . . . "

Software Infrastructure for Spoken Dialogue System



Text to Speech Synthesizer

Dialogue Manager

HUB

Phone – Asterisk Server

Automatic Speech Recognizer

Backend Server

UI Client/Session Manager

/…./ Directory

Function

1. New Call Established

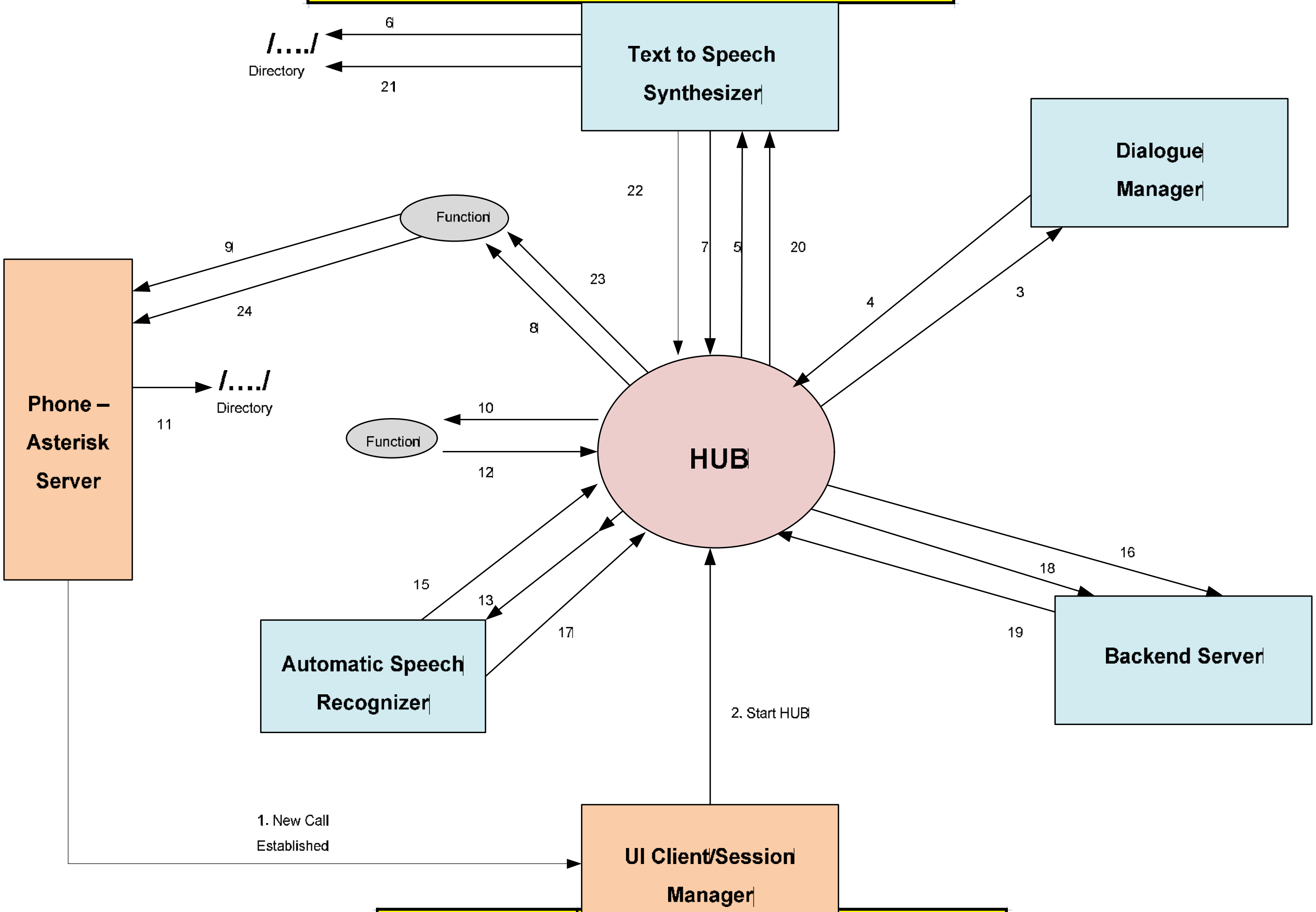
2. Start HUB

Sample Call Flow

0. Wait for new calls from user.
1. User calls using a telephone/softphone.
2. New session for Galaxy Hub is created.
   - Telephony server (Asterisk) session ID is mapped with the Galaxy Session ID
   - Hub Program is initiated
3. Hub invokes the Dialogue Manager's *greeting* function
4. Dialogue Manager returns the frame with the *greeting* string

"Hello andWelcome to Center for Language Engineering. Please record your current location after the first beep tone and your destination location after the second beep tone"

Software Infrastructure for Spoken Dialogue System



Text to Speech Synthesizer

Dialogue Manager

Phone – Asterisk Server

HUB

Automatic Speech Recognizer

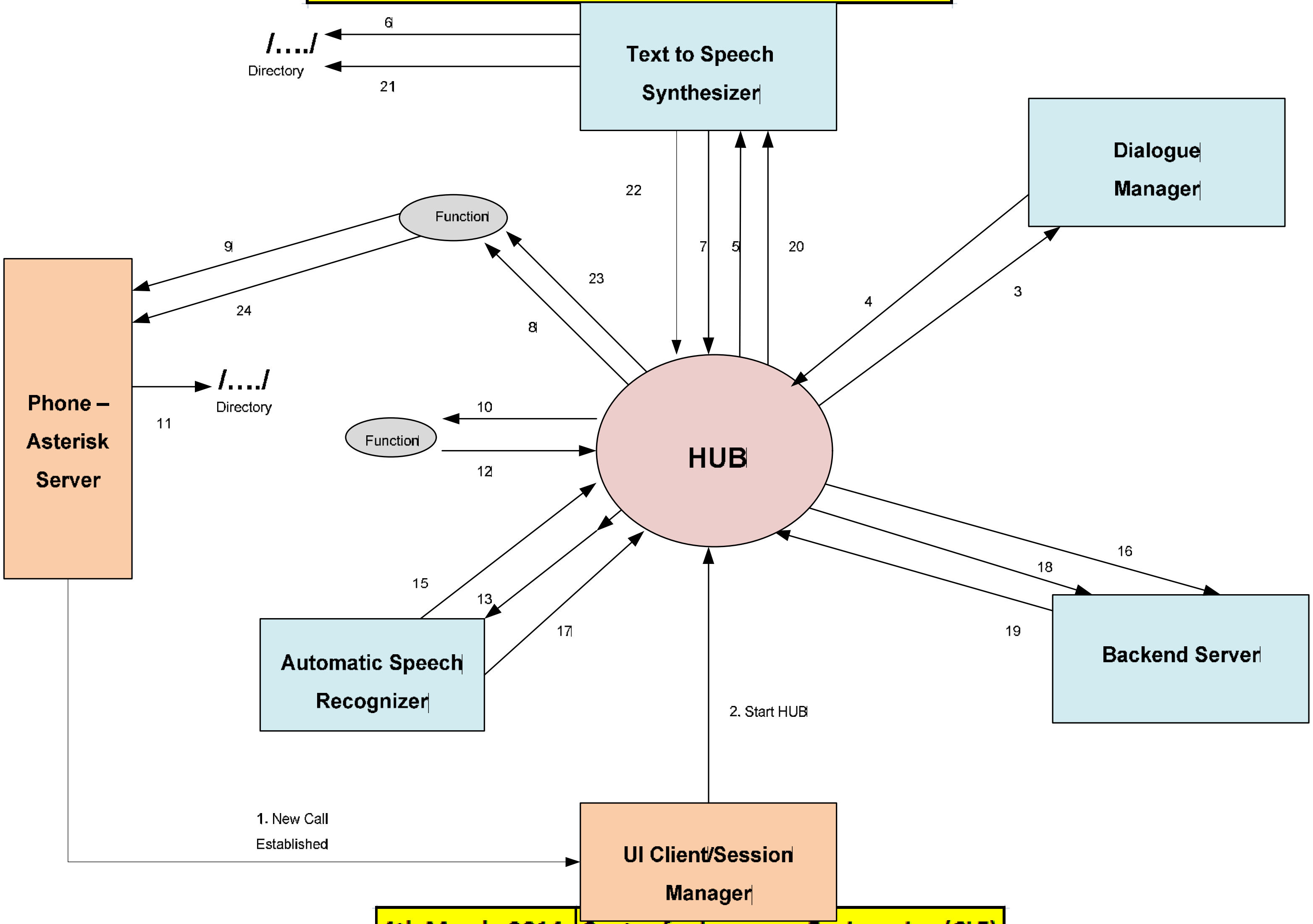
Backend Server

UI Client/Session Manager

Sample Call Flow

5. Hub forwards the *greeting* frame to TTS for speech synthesis
6. TTS Synthesizes the speech and stores it on a local directory
7. TTS returns the path of synthesized speech file to Hub using a frame
8. Hub invokes a function that sends the synthesized speech file to the Telephony server over a socket connection

Software Infrastructure for Spoken Dialogue System



4th March, 2014 | Center for Language Engineering (CLE)

Sample Call Flow

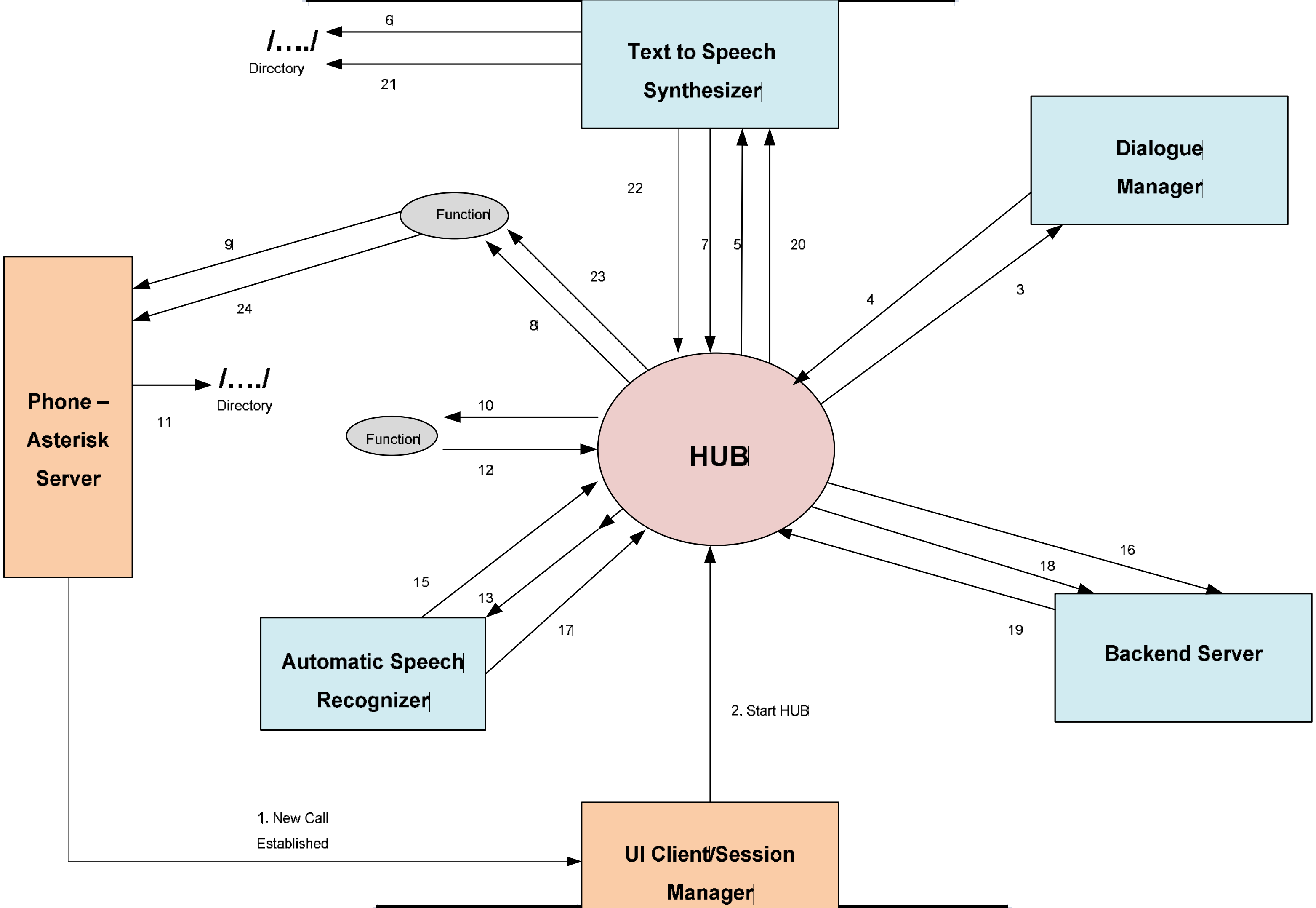
10. Hub initiates a lookup function to search for the source and destination location speech files from the user.
11. User records the current and destination locations on successive beeps:

Model Town

Gawal Mandi

After recording, these files would be sent to Galaxy Communicator over a socket connection

Software Infrastructure for Spoken Dialogue System

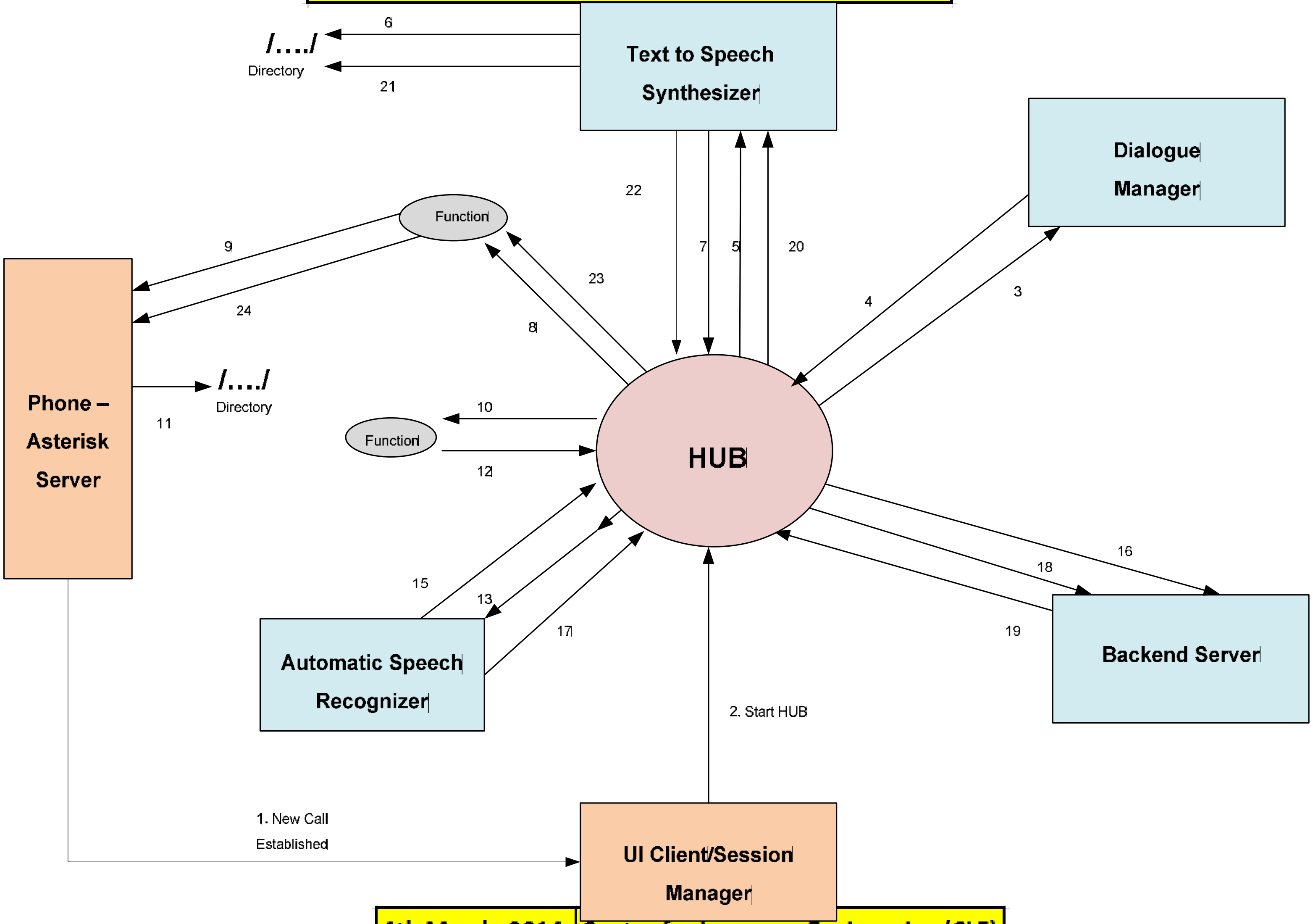


4th March, 2014 | Center for Language Engineering (CLE)

Sample Call Flow

12. The location of received files are sent to the Hub
13. Hub forwards the received frame to ASR for recognition
14. Decoding process starts in the ASR.
15. Decoded source location *"Model Town"*is sent to Hub in a frame
16. Hub forwards the received frame to Application Backend server
17. Decoded destination location *"Gawal Mandi"*is sent to Hub
18. Hub forwards the received frame to Application backend server

Software Infrastructure for Spoken Dialogue System



4th March, 2014 | Center for Language Engineering (CLE)

Sample Call Flow

19. Application backend returns the path from "*Model Town*" to "*Gawal Mandi*", and forwards it to Hub

"From MODEL TOWN,Lahore to GAWAL MANDI,Lahore

Total Distance is 14.1 km, Head south,

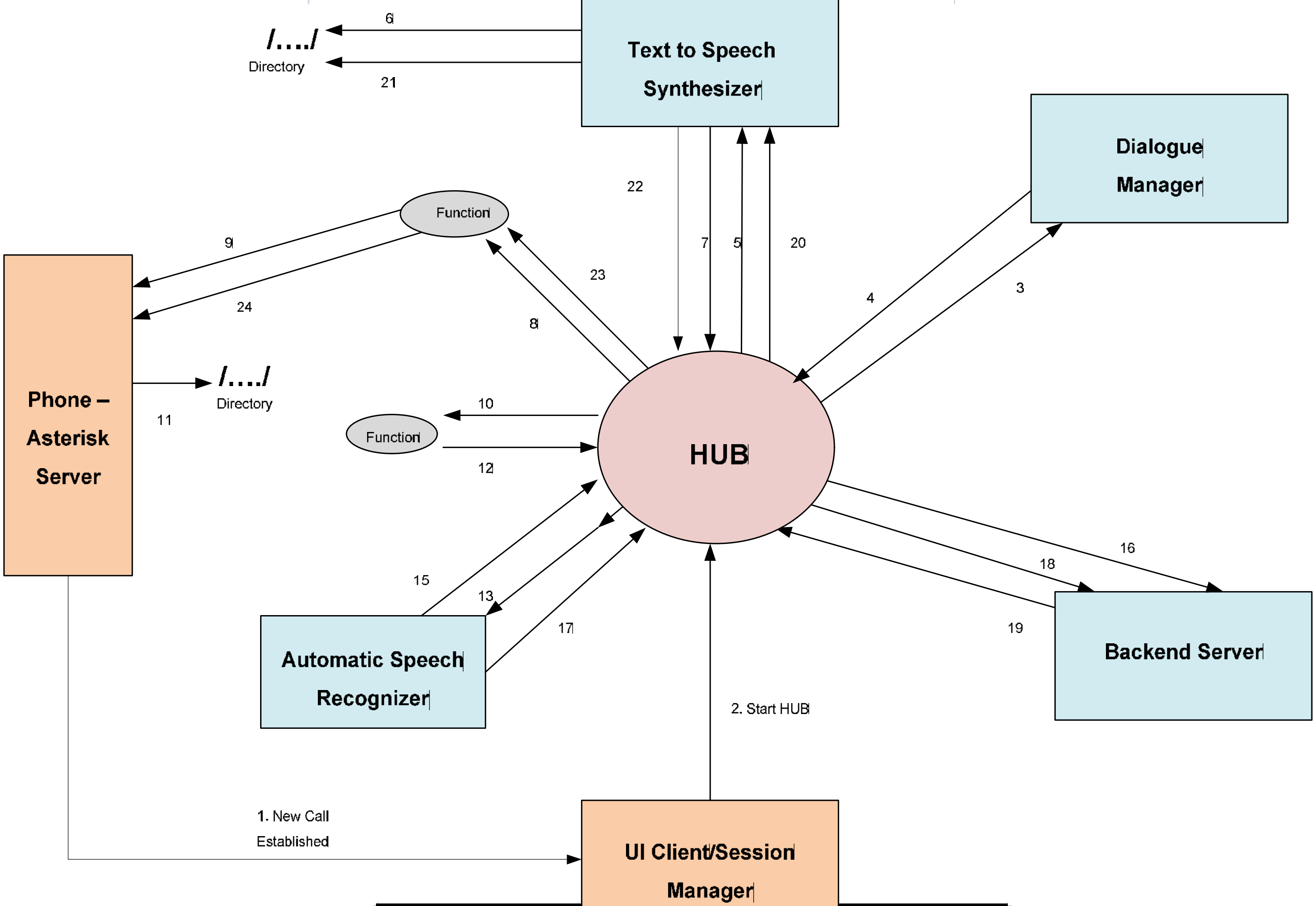
After 0.2 km, Take the 1st left toward Ferozepur Rd,

After 0.9 km, Take the 3rd right toward Ferozepur Rd,

After 0.3 km, Turn left onto Ferozepur Rd,

After 2.6 km, Continue straight onto Kalma Chowk Flyover……(continued).."

Software Infrastructure for Spoken Dialogue System



Text to Speech Synthesizer

Dialogue Manager

Phone – Asterisk Server

HUB

Automatic Speech Recognizer

Backend Server

UI Client/Session Manager

1. New Call Established

2. Start HUB

4th March, 2014 | Center for Language Engineering (CLE)

Sample Call Flow

20. Hub forwards the *greeting* frame to TTS for speech synthesis

21. TTS Synthesizes the speech and stores it on a local directory

22. TTS returns the path of synthesized speech file to Hub

23. Hub invokes a function that sends the synthesized speech file to the Telephony server over a socket connection

24. Synthesized speech file is sent over the socket connection

25. Speech file is played-back to the user

26. Call ends

Prototype Demo

System:

"Hello and Welcome to Center for Language Engineering. Please record your current location after the first beep tone and your destination location after the second beep tone"

User:

Model Town

Gawal Mandi

System:

"From Model Town Lahore to Gawal Mandi Lahore . . Total Distance is 14.1 km, Head south,

After 0.2 km, Take the 1st left toward Ferozepur Rd .... "

Challenges

- Multiple and Concurrent calls handling.
- Integrating the Ravenclaw Dialogue Manager in Galaxy Communicator
- Building a telephony server that could handle an E1 line/multiple trunks.
- System stability testing.

Questions?

Thank you for your patience!