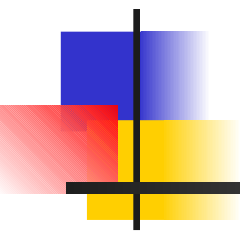


# An interactive natural-language genealogy quiz engine

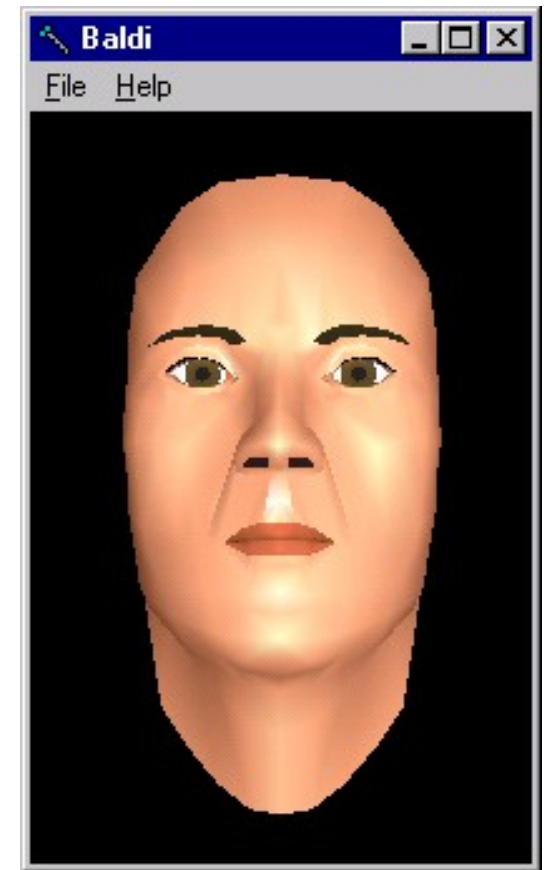


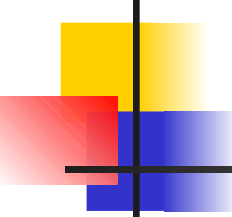
---

Aric Bills, Rebecca Rees, Merrill  
Hutchison, Clint Tustison, Nick Stetich,  
Mike Manookin, Hans Nelson and  
Deryle Lonsdale  
(BYU Soar Research Group)

# Background

- **GEDSpeak: interface to GEDCOM file contents**
  - Speech recognition, TTS
  - Animated agent oracle
  - Question answering
  - Several query types
  - Word spotting, partial structure





# Another perspective (interaction)

---

- **System focus**

**vs.**

- User initiates question
- User queries system
- System gives response
- One discourse turn per participant
- Not very personable, natural

- **User-directed**

- System initiates questions
- System queries user
- User provides answers
- Several turns per participant
- Potential for engaging (or alienating) the user



# Another perspective (data)

---

- **Low-level view**

**vs.**

- Factoids are paramount
- Details are only loosely connected
- Personal dimension, connection with “real world” are tenuous
- Access method is browsing, WIMPy

- **Holistic view**

- Generalizations, tendencies are key
- Observations emerge from across records
- Relates factoids to real people, world
- Access method is linguistic, SILKy

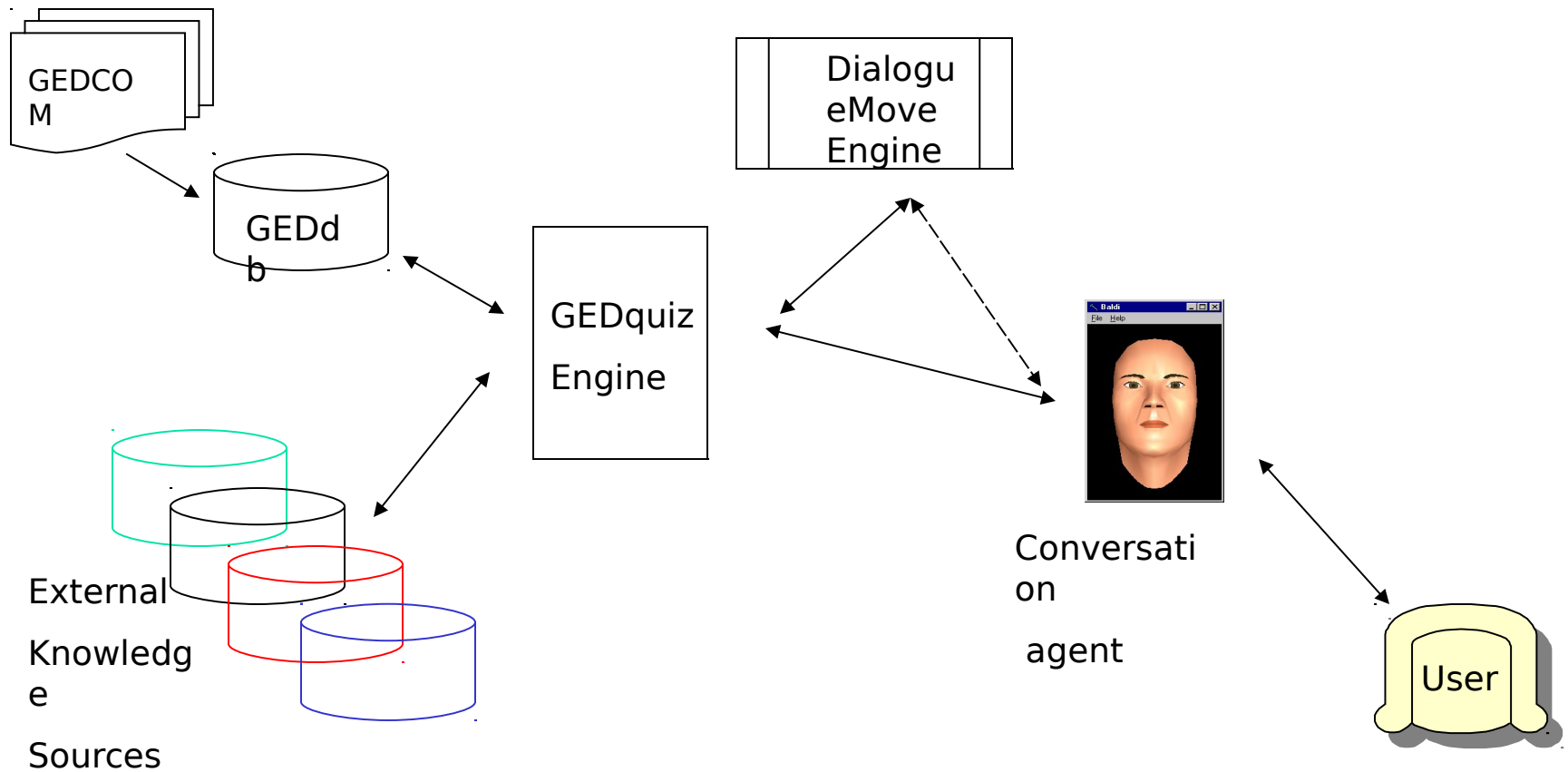


# Goal

---

- **Provide high-level insight into GEDCOM data**
- **Endow system with “omniscient” viewpoint (modulo closed-world assumption)**
- **Create learning/tutoring environment for acquiring/testing holistic genealogical knowledge**
- **Situate activities in natural goal-directed dialogues, conversations**
- **Leverage reasoning techniques, pragmatics**

# System architecture





# Sample questions

---

**S: Which of your ancestors had fifteen children?**

**U: Celinda Ann Heaton.**

**S: Who is a second cousin of yours?**

**a) Jared Scheuerman**

**b) Bill Scheuerman**

**c) Mary Barfuss**

**d) George Konschoo**

**U: b.**

**S: Did any of your Dutch ancestors die in the U.S?**

**U: No.**



# Database conversion

---

- **GEDCOM file input, translated via Perl to Prolog database**
  - Assertions stored in predicate-logic format
  - Contents visible to controlling engine, dialogue move engine
- **Serves as the basis for explicitly-encoded information**





# Control, knowledge processing

---

- **SICStus PROLOG engine**
  - Forward inferencing, theorem proving, goal-directed
- **Extensive knowledge base**
  - A few hundred representative meaning postulates
    - Hand-crafted for now; text mining is possible
- **Predicate matching to select questions**



# Sample predicate match

---

Which husband/wife combination was born on exactly the same day in exactly the same place?

```
husband_wife(HusbName,HBirthdate,WifeName,WBirthdate,X) :-  
  individual(Husb,name(HusbName),_,_,_,birthdate(HBirthdate),_,_,_,birthplace(X),_,_),  
    family(_,husband(Husband),_,_),  
    parse_date(HBirthdate,HDay,HMonth,HYear),  
  individual(Wife,name(WifeName),_,_,_,birthdate(WBirthdate),_,_,_,birthplace(X),_,_),  
    family(_,wife(Wife),_,_),  
    parse_date(WBirthdate,WDay,WMonth,WYear),  
    HYear == WYear,HMonth == WMonth,HDay == WDay.
```

**Husband\_Name = 'Garland /Bailey/'**  
**Husband\_Birth = '16 Apr 1912'**  
**Wife\_Name = 'Carolyn /Warren/'**  
**Wife\_Birth = '16 Apr 1912'**  
**Birthplace = 'Gracemont, Caddo,**  
**Oklahoma' ;**

**Husband\_Name = 'Charles Arthur**  
**/Goodpasture/'**  
**Husband\_Birth = '25 Dec 1894'**  
**Wife\_Name = 'Betty Lucille /Rittga/'**  
**Wife\_Birth = '25 Dec 1894'**  
**Birthplace = 'Gracemont, Caddo,**  
**Oklahoma'**



# Using “commonsense” knowledge

---

## COMMONSENSE KNOWLEDGE

**Which ancestors of yours were born in Oklahoma during the Great Depression?**

```
city(ok, 'Anadarko').
city(ok, 'Lookeba').
city(ok, 'Oklahoma City').
city(ok, 'Binger').
Loc = ok, Context = 'depression'.
city(ok, 'Stillwater').
```

situated\_event(Name, Loc, Context) :-

```
year_range(depression, 1929, 1939),
sub_atom(Name, _, _, birthdate(Birthdate), _, _, birthplace(Birthplace), _, _),
```

```
5) sub_atom(Birthplace, Before, X, After, ', '),
```

```
sub_atom(Birthplace, 0, Before, _, Y), city(Loc, Y),
```

```
parse_date(Birthdate Day Month Year) in year_range(Context Year)
```

## ANSWERS TO THE QUERY

Name = 'Audrey /Jack /Long' ;

Name = 'Willard Warren /Sullivan/' ;

Name = 'Alton /Chatham/' ;

Name = 'Jack L /Felton/' ;

Name = 'Ruby /Six/' ;

Name = 'Betty /Wanzor/' ;



# Dialogue structure

---

- **State-of-the-art discourse management engine**
  - Specify, manipulate dialogue/discourse turns
  - Manage model of total information state
    - Private beliefs, plans, discourse agenda
    - Shared knowledge: content, context, common ground
    - Accommodation of goals, partial and out-of-sequence info
  - More natural, powerful than simple finite-state techniques
  - V-commerce, call center management, conversation tracking, intelligent tutorial dialogues



# Verifying user answers

---

**\$U> Joe Clark.**

```
# assumeUsrMovesGrounded
> set#rec(shared^lu^speaker,usr)
> clear#rec(shared^lu^moves)
> forall_do(in(latest_moves,A),add#rec(shared^lu^moves,A,false))
# integrateUsrAnswer
> set_assoc#rec(shared^lu^moves,answer(joe_clark),true)
> pop#rec(shared^qud)
> add#rec(shared^com,name(joe_clark))
# removeAgendaFindout
> pop#rec(private^agenda)
# verifyUsrBelief
> add#rec(private^bel,xbelief(name(joe_clark),yes))
# refillAgendaFromPlan
> pop#rec(private^plan)
> push#rec(private^agenda,inform(name(joe_clark)))
# selectInform
> set(next_moves,set([inform_yn(name(joe_clark),yes)]))
```



# Accommodating the user

---

**% accomodate the qud with the topmost action on the agenda**

```
rule( accommodateQuestion,  
    [ val#rec( shared^lu^speaker, usr ),  
      in#rec( shared^lu^moves, answer(A) ),  
      not( lexicon :: yn_answer(A) ),  
      assoc#rec( shared^lu^moves, answer(A), false ),  
      fst#rec( private^agenda, findout(Q) ),  
      domain :: relevant_answer( Q, A ) ],  
    [ pop#rec( private^agenda ),  
      push#rec( shared^qud, Q ) ]  
    ).
```



# A mixed-initiative quiz

---

**S: Who was your paternal grandfather?**

**U: Peter Lonsdale.**

**S: Right.**

**S: Did you know he immigrated to  
Canada?**

**U: Yes, I knew that.**

**U: He was born in the U.S.**

**S: Correct.**

**S: Who was his paternal grandfather?**

**U: I don't know.**

**S: Ole Christensen of Oppland, Norway.**



# Question generation

---

- **Question formats**
  - YNQ
  - Multiple choice
  - Concept completion, open-ended, etc.
- **Linguistically nontrivial in English**
  - Anaphora, pronominalization, coreference, etc.
- **Context-sensitive template matching/filling (current)**
- **Context-free phrase-structure component (future)**





# User interaction

---

- **Currently via keyboard**
- **Are interfacing with speech engine**
  - Client/server architecture, sockets
  - Dynamically specified grammar (after Q-formulation) for improved recognition
- **Possibility for repair subdialogs**



# Summary

---

- **Advantages**
  - Framework for more natural interaction
- **Current functionality**
  - Works well as a prototype level
- **Development issues**
  - Getting speech toolkit to integrate seamlessly with dialogue engine



# Future work

---

- **Add to knowledge base**
- **User profiling**
- **Complete speech-based round-trip**
- **Return initiative to user again (expert natural-language discourse engine)**
- **Mixed-initiative discourse**
- **Port to Soar architecture**