

“Extendable,  
Semantics-Driven  
Parsing  
of Pidgin Language”

Robin Stewart, Max Goldman, (all y'all)

# Intuition:

- **“karger 3pm tuesday G592”**
- This makes sense to us because we know “karger” refers to a person, “3pm tuesday” refers to a time, and “G592” refers to a place.
- And the most obvious way to interpret a person, time and place is as an event.
- Order doesn't matter!  
“3pm tuesday karger G592”

# Need: extensibility

- **“desk chair \$350 circa 1846”**
- Meeting with a desk chair in 32-350 in 1846?
- No, “desk chair” is a household object, “\$350” is a price, and “1846” is a date.
- It probably means you spent \$350 on an antique desk chair that was made in 1846.

# Need: personal information

- What/who is “karger”?
- “karger” != “desk chair”
- “G592”
- Personally/locally relevant info.

# Type-driven parsing

- **“karger 3pm tuesday G592”**
- initialize with bag of string tokens:  
“karger”, “3pm”, “tuesday”, “karger 3pm”,  
“3pm tuesday”, “tuesday G592”, ...
- Higher-level recognizers simply add new tokens to the chart (along with new type information and weight)
- When no new tokens can be added, choose the full parse (if any) with highest weight

# Type recognizers

- Two kinds:
  - RegExp-based: date/times, email addresses
  - Ontology-based (aka “semantic web”): search the triple store for objects of a particular class, e.g. “Person”, and then perform substring matching, e.g. “dav kar”. Weight is based on length of match.

# person.n3

```
<#Person> a owl:Class;  
  rdfs:comment "";  
  nl:surface "person";  
  nl:surface "someone";  
  rdfs:label "Person" .
```

```
<#fullname> a owl:Property;  
  rdfs:label "Full Name";  
  nl:prep "named";  
  nl:prep "name";  
  rdfs:range dt:string .
```

# person.n3

```
<#Person> a owl:Class;  
  rdfs:comment "";  
  nl:surface "person";  
  nl:surface "someone";  
  rdfs:label "Person" .
```

```
<#fullname> a owl:Property;  
  rdfs:label "Full Name";  
  nl:prep "named";  
  nl:prep "name";  
  rdfs:range dt:string .
```



# person.n3

```
<#Person> a owl:Class;  
  rdfs:comment "";  
  nl:surface "person";  
  nl:surface "someone";  
  rdfs:label "Person" .
```

```
<#fullname> a owl:Property;  
  rdfs:label "Full Name";  
  nl:prep "named";  
  nl:prep "name";  
  rdfs:range dt:string .
```

# people.n3

.  
.

**<#eMax>**

```
    rdf:type          person:Person ;  
    person:fullname  "Max VanKleek" ;
```

.

**<#ProfKarger>**

```
    rdf:type          person:Person ;  
    person:fullname  "David Karger" ;
```

.

.  
.

# meeting.n3 (excerpts)

```
<#Meeting> a owl:Class;  
  rdfs:label "Meeting";  
  .
```

```
<#attendee> a owl:Property;  
  rdfs:domain :Meeting;  
  rdfs:range person:Person .
```

```
<#location> a owl:Property;  
  rdfs:range place:Place .
```

```
<#dtstart> a owl:Property;  
  rdfs:range dt:dateTime .
```

# “Prepositioners”

- **“meeting with karger at 3pm on tuesday in G592”**
- Some people like to be verbose
- Can reduce ambiguity
- We introduced a way to specify natural language realization in an ontology:
  - “nl:surface” for classes (“meeting”)
  - “nl:prep” for properties (“with”)

# meeting.n3 (expanded)

```
<#Meeting> a owl:Class;  
  nl:surface "meeting";  
  nl:surface "meet";  
  .  
  
<#attendee> a owl:Property;  
  nl:prep "with";  
  rdfs:range person:Person .  
  
<#location> a owl:Property;  
  nl:prep "at";  
  nl:prep "in";  
  rdfs:range place:Place .  
  
<#dtstart> a owl:Property;  
  nl:prep "at";  
  nl:prep "starting at";  
  nl:prep "from";  
  rdfs:range dt:dateTime .
```

# Current system output:

> meeting with karger at 3pm on tuesday in G592

# Current system output:

> meeting with karger at 3pm on tuesday in G592

Success

0.09375 CREATE A NEW <http://people.csail.mit.edu/stewart/meeting.n3#Meeting>

Dictionary

[http://people.csail.mit.edu/stewart/meeting.n3#dtstart:](http://people.csail.mit.edu/stewart/meeting.n3#dtstart)  
(2007, 5, 22, 15, 0, 0, 1, 142, -1)

[http://www.w3.org/1999/02/22-rdf-syntax-ns#type:](http://www.w3.org/1999/02/22-rdf-syntax-ns#type)

<http://people.csail.mit.edu/stewart/meeting.n3#Meeting>

[http://people.csail.mit.edu/stewart/meeting.n3#location:](http://people.csail.mit.edu/stewart/meeting.n3#location)

file:///6863repository/entities/places.n3#32-G592

[http://people.csail.mit.edu/stewart/meeting.n3#attendee:](http://people.csail.mit.edu/stewart/meeting.n3#attendee)

file:///6863repository/entities/people.n3#ProfKarger

# Other successful parses:

meeting with karger at 4pm in g531  
tomorrow 4pm in g531 meeting with karger  
karger 4pm tomorrow g531  
g531 4pm karger



# Other successful parses:

meeting with karger at 4pm in g531  
tomorrow 4pm in g531 meeting with karger  
karger 4pm tomorrow g531  
g531 4pm karger

meet robin at au bon pain noon thursday  
Meet someone named Max somewhere called Kiva at 3:30

# Other successful parses:

meeting with karger at 4pm in g531  
tomorrow 4pm in g531 meeting with karger  
karger 4pm tomorrow g531  
g531 4pm karger

meet robin at au bon pain noon thursday  
Meet someone named Max somewhere called Kiva at 3:30

3pm karger in pain  
dav kar kiv 333  
today arg pain

# Other successful parses:

meeting with karger at 4pm in g531  
tomorrow 4pm in g531 meeting with karger  
karger 4pm tomorrow g531  
g531 4pm karger

meet robin at au bon pain noon thursday  
Meet someone named Max somewhere called Kiva at 3:30

3pm karger in pain  
dav kar kiv 333  
today arg pain

karger@mit.edu david karger  
david karger email karger@mit.edu

# Future Work!

- Dealing with Ambiguity: refine weighting system (and ability to specify in ontology?)
- More efficient parsing???
- More efficient type recognizers?
- Ability to UPDATE (when some fields are recognized & some fields are “create”s?)
- Discourse model - “gaps” filled in by previous tokens:

# Future Work!

- Dealing with Ambiguity: refine weighting system (and ability to specify in ontology?)
- More efficient parsing???
- More efficient semantic type recognition?
- Ability to UPDATE (when some fields are recognized & some fields are “create”s?)
- Discourse model - “gaps” filled in by previous tokens:
  - > meetings tomorrow
  - > 3pm karger
  - > 4pm max

(end)

# people.n3

.

.

<#eMax>

```
    rdf:type          person:Person ;  
    person:fullname  "Max VanKleek" ;
```

.

<#ProfKarger>

```
    rdf:type          person:Person ;  
    person:fullname  "David Karger" ;
```

.

.

.