CS243
ThingTalk & ThingPedia

1. Motivation
2. Overview of ThingSystem
3. ThingTalk Language
4. ThingTalk Implementation

https://thingengine.stanford.edu/
“ACQUIRES SMART THERMOSTAT MAKER NEST FOR $3.2 BILLION”

Forbes, Jan 13, 2014

Who will be the Google of the internet of things?

Financial Times, May 1, 2014
Where is MY Data?

• Locked up in different silos:
  • Nest, webcam, car
  • Fitness band, scale, ECG reader, blood pressure
  • Facebook, Google, LinkedIn, email, messenger
• Can I access all my data easily?
• Can I share with arbitrary friends, in arbitrary ways?
• With control over the ownership of data?

Scenario: You are at SxSW

• So many people, yet so little interaction
• Find others sharing same jobs
• Share our LinkedIn profile and find each other!

Easy to use: Dynamic group
Privacy: No 3rd party owning data
DIY: No LinkedIn dependency
Easy to code: 10 lines of code
2. Overview of ThingSystem

Ann’s ThingSystem

Ann’s ThingSystem
Can be run on the phone, at home, in the cloud

Thing Manager

Thing Engine

Sabrina: Virtual Assistant

Omlet Chat
Open Messaging
**ThingPedia: Open-Source Crowd-Sourced Repository of IoT Apps**

IoT App = ThingTalk Apps + Device Interfaces (run on ThingEngines)

**Device Interfaces**
- Web resources
- IoTs

**ThinkTalk Apps**
- @facebook.profile ( , , , pictureurl, , , )
- @twitter.setprofile (pictureurl);

**Authentication**
- user-password, OAuth2

**Triggers**
- twitter.source(text, hashtags, …)

**Actions**
- twitter.setprofile( , , , pictureurl, …)

---

**Demo (Get Warrior News)**

**Device Trigger Channel**

```java
nba_team(watchedTeamAlias : String,
otherTeamAlias : String,
watchedIsHome : Boolean,
awayName : String,
homeName : String,
gameStatus : String,
scheduledTime : Date,
awayPoints : Number,
homePoints : Number)
```

**ThingTalk App**

```java
@sportradar.nba_team (Team, _, _, awayTeam, homeTeam, "closed", _, awayScore, homeScore)
=> @notify("Final score for " + awayTeam + " @ " + homeTeam + ": " + awayScore + " - " + homeScore);
```

---

M. Lam
Stanford University
Future Scenario: A Medical Coach
Continuous Personalized Medical attention

Dr. Yeung

Bring your weight down to 150 pounds.
I see that there is a new scale in the house,
could I configure that for you?
Keep up your diet! You're just 2 pounds over your target

Walk 5000 steps a day.
You are only averaging 1000 steps this week,
It is sunny out, why not go out for a walk?

If the blood pressure > 140,
notify me.
It's time to measure your blood pressure.
Your blood pressure is fine — no need to worry.

Social Interactions

Omlet Group Feed

Ann's Thing System
Ann's LinkedIn Acct

Bob's Thing System
Bob's LinkedIn Acct

Carol's Thing System
Carol's LinkedIn Acct
LinkedInApp

```javascript
LinkedInApp(F) {  
  var Industry[]:  
    (String, String);  
  var NewColleague:  
    (String, String);  
  out Colleagues:  
    Map(String, String);
}
!Colleagues(_)  
  => Colleagues($emptyMap());
@linkedin.profile (name, _, field, _, _, _)  
  => Industry[self](name, field);
Industry[self](_, field), Industry[m](name, field), m in F, m != self  
  => NewColleague(name, field);
NewColleague(name, field)  
  => Colleagues($insert(Colleagues, name, field));
NewColleague(name, field)  
  => @$notify("I found that " + name + " is your colleague");
```

Benefits

**USERS**
- Interoperability
- Full access to all our personal digital assets
- Group share & compute
- Privacy: choice of ThingSystem
- Easy-to-use: Natural language interface

**MAKERS**
- Interoperability
- Open competition as ThingSystem host (vs. monopoly)
- Universal: open-source, crowd-sourced
- Complete stack: Assistant - discovery
- Extensible: devices, ThingTalk apps
- Easy-to-code: a few lines
### Related Work

<table>
<thead>
<tr>
<th></th>
<th>Distributed</th>
<th>Proprietary / Centralized</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Open System</strong></td>
<td>Google</td>
<td>Apple</td>
</tr>
<tr>
<td><strong>Virtual Assistant</strong></td>
<td>Sabrina</td>
<td>Google Now</td>
</tr>
<tr>
<td><strong>Language</strong></td>
<td>ThingTalk</td>
<td></td>
</tr>
<tr>
<td><strong>Data</strong></td>
<td>ThingSystem</td>
<td>Google</td>
</tr>
<tr>
<td><strong>Messaging</strong></td>
<td>Omlet</td>
<td>OS</td>
</tr>
<tr>
<td><strong>Discovery</strong></td>
<td>AllSeen, OIC, ... (coming)</td>
<td>Brillo</td>
</tr>
<tr>
<td><strong>App Database</strong></td>
<td>ThingPedia</td>
<td></td>
</tr>
</tbody>
</table>

**Omlet: An Open Messaging Platform for Distributed Apps**

- Omlet Chat Apps API
- Disney
- Baidu cloud
- Cloud API
- WeChat + Distributed Facebook
- A distributed semantic file system

Data owned by end users; Deleted from Omlet server after 2 weeks
**IFTTT (If-this-then-that)**

- Repository of device channels & IFTTT recipes
- Centralized service: they get your credentials
- Web form interfaces: no formal language
- Limitation:
  - Single trigger & action, no computation
  - No social interactions

---

**3. ThingTalk Language**
Abstraction of ThingTalk

User → ThingManager → ThingEngine → Twitter

1. enable app
2. get my "twitter"
3. get twitter interface
4. return twitter interface
5. Oauth
6. return authenticated twitter instance
7. interface with my Twitter
8. report event
9. notify user

TwitterHashtagFilter(HashTag : String) {
    @twitter.source(text, hashtags, _, from, _, false),
    $contains(hashtags, HashTag)
    ⇒ @$notify("Tweet from @" + from + ": " + text);
}

ThingTalk Basics

- Unification style (like Prolog, Datalog)
- External: Triggers (input) & Actions (output) (@ sign)
- Internal: keyword(value) (no @ sign)
  - LHS matches the value assigned to the keyword
  - RHS sets the keyword to the value (old value removed)
- Each rule can have at most one trigger & one action
- Rules with a trigger: evaluated whenever a trigger is fired
- Rules without triggers: evaluated whenever LHS keyword values change

@Bank1.getbalance(b1)
⇒ Balance1(b1)

@Bank2.getbalance(b2)
⇒ Balance2(b2)

Balance1(b1), Balance2(b2)
⇒ TotalBalance(b1 + b2);

TotalBalance(b), b < 500
⇒ @$notify("Low balance");
Naming Things

- By name (in ThingPedia): @twitter, @facebook
- By attributes in things in your ThingManager
  - @(type="scale"): all scales
  - @(type="scale", place="livingroom"): all scales with attribute “place” = “livingroom”
  - @(type="com.bodytrace.scale"): scales made by BodyTrace—a subset of @(type="scale")

Built-In Functions

<table>
<thead>
<tr>
<th>Function</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>@$notify(...)</td>
<td>send a notification through Sabrina; any value can be passed, not just strings</td>
</tr>
<tr>
<td>@$return(...)</td>
<td>like @$notify(), but also disable the app</td>
</tr>
<tr>
<td>@$logger(msg)</td>
<td>append a message to the logs</td>
</tr>
<tr>
<td>@$timer(1000)</td>
<td>time based triggers: parameters must be constants</td>
</tr>
<tr>
<td>@$at(&quot;8:00&quot;)</td>
<td></td>
</tr>
</tbody>
</table>
**ThingTalk Variables**

- **Variable scope**
  - `var v: type`
  - `out v: type` (available to users in “Show app”)
  - `extern v : type` (available to other users’ apps)

- Simple variable types: String, Int, Measure(unit)

- Compound variable types: tuples, Array, Map
  - e.g. `var record : Array (Measure(kg));`
  - `var trip: (date, date);`

- Built-in operators: `+`, `-`, `*`, `/`, `!`, `&&`, `||`, `<`, `>`, `<=`, `>=`, `=`, `!=`, `=~`
  - `!VarName(_): VarName == `null`

- Built-in functions: e.g. `$insert`, `$append`, `$contains`, `$lookup`, `$argmin`, `$concat`

```plaintext
var MyWeight : Measure(kg) ;

@(type="scale").source(_, w), !MyWeight(_)
=> MyWeight(w);

@(type="scale").source(_, w2), MyWeight(w1)
=> $notify("Your weight increased by " + $toString(w2 - w1) + " kgs");
```

**ThingTalk Shared Data Model**

- A social app has a feed [F] every member of the feed runs the SPMD program

- Social variables are declared with `[]`
  - `Locn[m](l): member m’s location`
  - `Locn[](l)`, `m ∈ F`
  - `=> @(type="Map").plot(m, l);`

```plaintext
GlympseApp[F] () {
  Locn[]: location;

  @gps(l, l)
  =>Locn[SELF](l);

  Locn [m](l), m ∈ F
  => @(type="Map").plot(m, l);
}
```

- Only owner can set the value
- All changed values are distributed to members
- No third party owns the data exchanged
LinkedInApp

LinkedInApp(F) { 
  var Industry[]: (String, String);
  var NewColleague: (String, String);
  out Colleagues: Map(String, String);

  !Colleagues(_)
    => Colleagues($emptyMap());
  @linkedin.profile (name, _, field, _, _, _)
    => Industry[self](name, field);
  Industry[self](_, field), Industry[m](name, field), m in F, m != self
    => NewColleague(name, field);
  NewColleague(name, field)
    => Colleagues($insert(Colleagues, name, field));
  NewColleague(name, field)
    => @$notify("I found that "+ name + " is your colleague");
}

Sabrina is Programmable too!

SabrinaTwitterFilter() {
  var HashTag : String;
  @sabrina.listen(text), $regex("text", "^notify on hashtag ([^ ]+)$", "i", tag)
    => HashTag(tag);
    @twitter.source(text, hashtags, _, from, _, _),
    HashTag(tag), $contains(hashtags, tag)
    => @$notify("Tweet from "+ from + ": " + text);
}
4. ThingTalk Implementation

**ThingEngine**

ThingTalk program $\rightarrow$ reactive-style JS programs

**Compiler**
- Non-deterministic evaluation order
- Compile each rule independently
- Program = combination of closures
  - Relies on the JS JIT
  - Fast enough

**Run-Time evaluation**
- Compiled apps register for events
- Dynamic I/O coalescing
**ThingEngine Implementation**

- Implemented in JS and nodejs
- Can be run on a home server, a phone, or as a cloud service

**Summary: Privacy & Distributed Computing**

- Big-data owned & controlled by “monopolies”
  -> Users own, share, compute data with friends of their choice
- Distributed computing on an open messaging platform
- Open Thing Platform
  - ThingSystem:
    - ThingManager (my accounts), ThingEngine (runs ThingTalk apps)
  - ThingTalk: Domain specific language
    - Connects cloud and IoT through public interfaces.
  - ThingPedia: Open-source, crowd-sourced repository