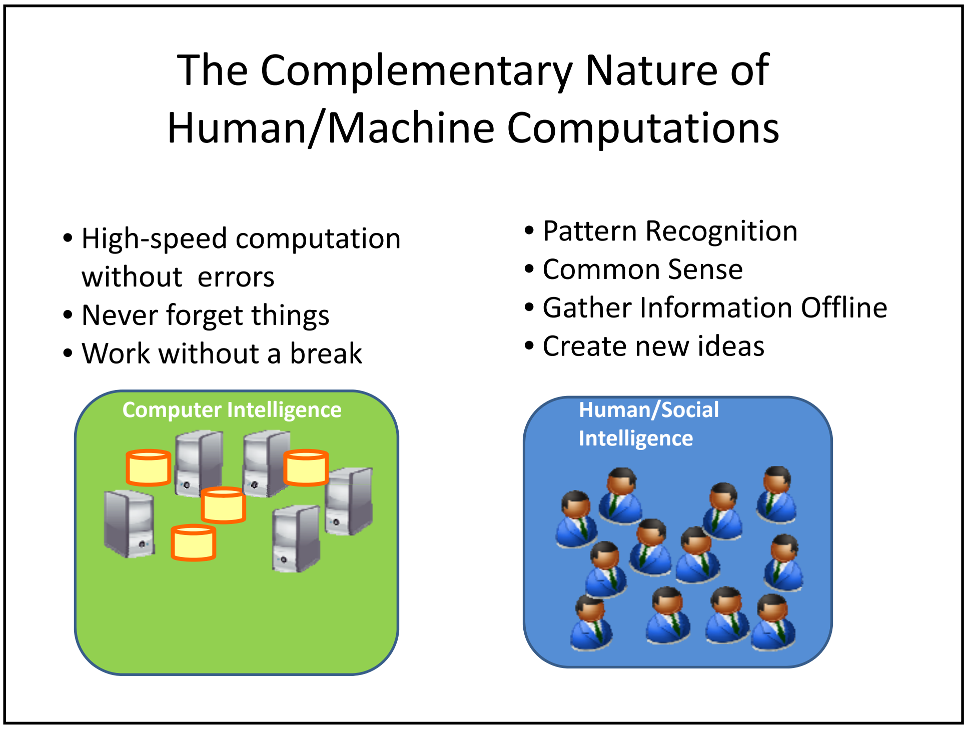
The Power of Integrated Abstraction of Data-centric Human/Machine Computations

Atsuyuki Morishima, Norihide Shinagawa
Shoji Mochizuki
University of Tsukuba

VLDS2011 held with VLDB2011, Seattle, Sep. 2011

Outline

1. Background
2. CyLog
3. Prototype Development
4. Related Work and Discussions



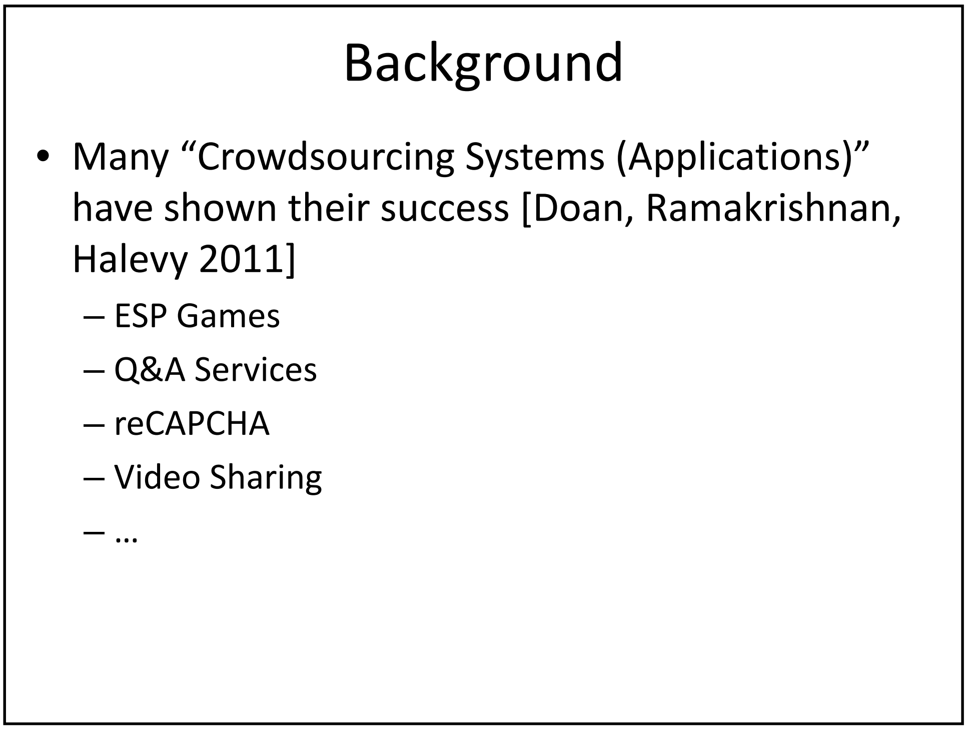
The Complementary Nature of Human/Machine Computations

- High-speed computation without errors
- Never forget things
- Work without a break

Computer Intelligence

- Pattern Recognition
- Common Sense
- Gather Information Offline
- Create new ideas

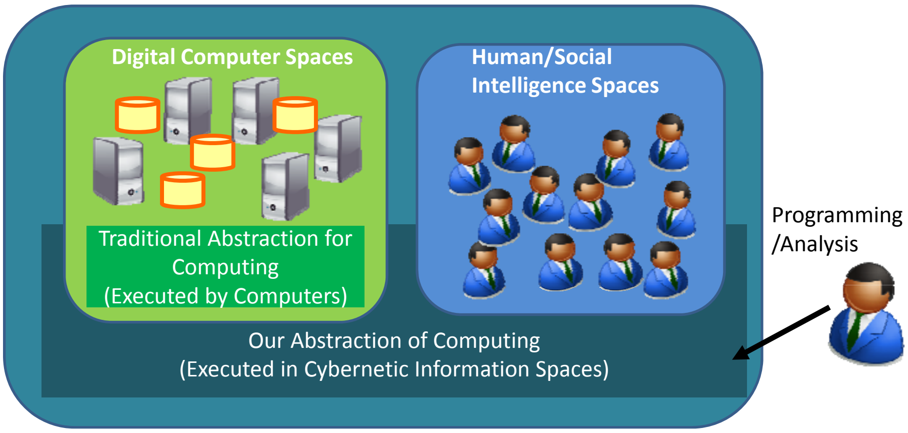
Human/Social Intelligence



Background

- Many “Crowdsourcing Systems (Applications)” have shown their success [Doan, Ramakrishnan, Halevy 2011]
  - ESP Games
  - Q&A Services
  - reCAPCHA
  - Video Sharing
  - …

Our Challenge: Develop a Systematic Framework to Quickly Build Programs for the Integration of Human/ Machine Computations



Digital Computer Spaces

Human/Social Intelligence Spaces

Traditional Abstraction for Computing (Executed by Computers)

Programming /Analysis

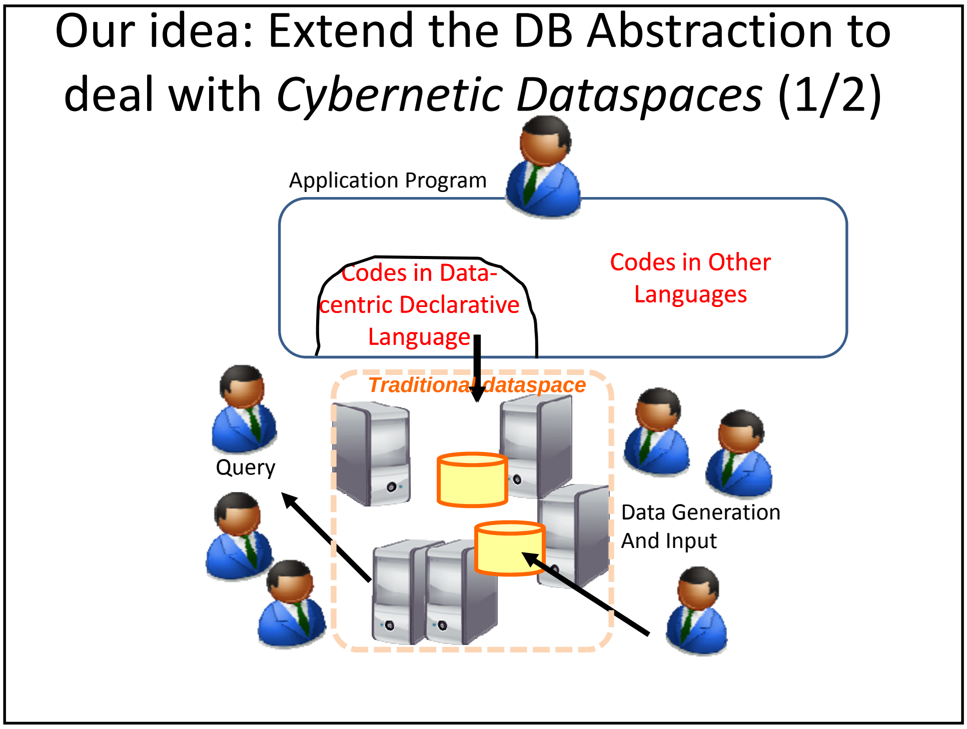
Our Abstraction of Computing (Executed in Cybernetic Information Spaces)

A Natural (and Important) Question

What is a good *abstraction* to describe (and program) such applications of human/machine computation?

- ESP Games
- Q&A Services
- reCAPCHA
- Video Sharing
- …

A possibility: Since they are data-centric, *database languages* can be a starting point to develop such an abstraction



Our idea: Extend the DB Abstraction to deal with *Cybernetic Dataspaces* (1/2)

Application Program

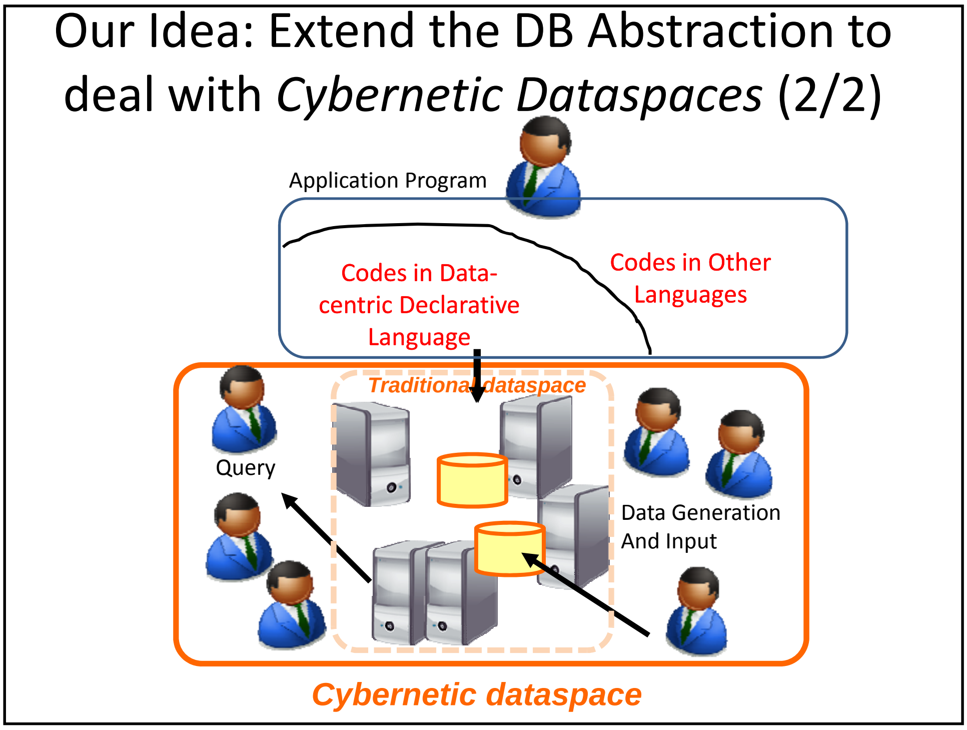
Codes in Data-centric Declarative Language

Codes in Other Languages

Traditional dataspace

Query

Data Generation And Input



Our Idea: Extend the DB Abstraction to deal with *Cybernetic Dataspaces* (2/2)

Application Program

Codes in Data-centric Declarative Language

Codes in Other Languages

Traditional dataspace

Query

Data Generation And Input

Cybernetic dataspace

Integrated Abstraction of Data-centric Human/Machine Computations: An Example of CyLog Rule

metadata(x, y) :- img(x), keyword(x, y), inDict(y)

Evaluated by data

Evaluated by humans

Evaluated by data

Many Ongoing Projects

- We saw exciting ongoing projects in publications in 2011
  - Qurk [MIT]
  - sCOOP/hQuery [Stanford & Santa-Cruz]
  - CrowdDB [UC Berkeley, ETH Zurich]

  …
- They try to achieve database functions in the presence of human data-sources

How is CyLog Different?

- Introduces the concept of *rational data source*, as a new type of Web data source
- *Open Predicates/Attributes* to model the interaction with people
- *Data games* for obtaining appropriate values
- Our first international presentation was in 2010!*

*Atsuyuki Morishima. A Database Abstraction for Social Applications, KJDB2010, May 2010.

Outline

✓ 1. Background

[✓ 2.] CyLog

3. Prototype Development

4. Related Work and Discussions

Point 1: Datalog-like Declarative Language

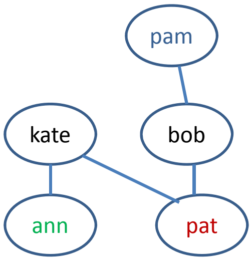
metadata(x, y) :- img(x), keyword(x, y), inDict(y)

Evaluated by machine

Evaluated by humans

Evaluated by machine

Point 2: Open Predicates (1/3) - CWA



Parent(pam, bob)

Parent(bob, pat)

Parent(kate, pat)

Parent(kate, ann)

Ancestor(X,Y) <- Parent(X,Y),

Ancestor(X,Z) <- Parent(X, Y), Ancestor(Y, Z)

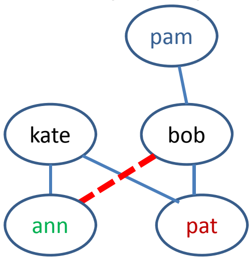
?- Ancestor(pam, pat)

yes

?- Ancestor(pam, ann)

No

Point 2: Open Predicates (2/3)



Parent(pam, bob)

Parent(bob, pat)

Parent(kate, pat)

Parent(kate, ann)

Ancestor(X,Y) <- Parent(X,Y),

Ancestor(X,Z) <- Parent(X, Y), Ancestor(Y, Z)

Parent(X,W)/open <- Parent(X,Y), Parent(Z, Y), Parent(Z, W)

Yes!

?- Ancestor(pam, pat)

yes

?- Ancestor(pam, ann)

Yes!

Point 2: Open Predicates (3/3) - Details

- Can have open attributes

  keyword(x,y)/open<- img(x)

- Possible to actively ask people

  keyword(x,y)/open{group}:active

- Can be an open “fact”

  img(x)/open

- Open for a specified set of humans

  keyword(x,y)/open{group}

Point 3: Data Games (1/2)

Challenge: Obtaining appropriate values in the presence of human data sources.

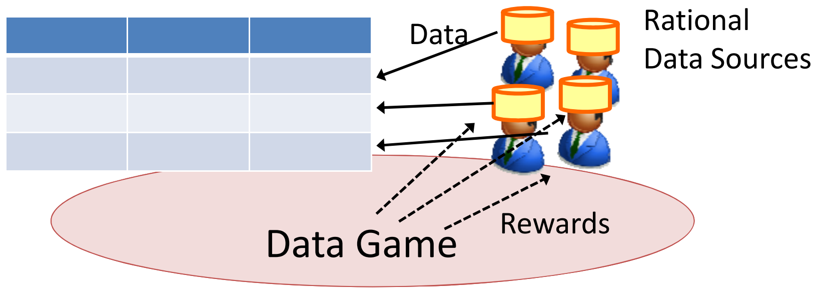
Approaches :

- Majority Voting
- Probabilistic Approach*
- Approach Using Item-Response Theory*
- Data Games

*Mentioned in [Parameswaran et al. 2011]

Point 3: Data Games (2/2)

- A concept to connect data flows with reward systems
- Models each human as a *rational data source* who behaves rationally according to the rewards given in the games.



- This framework gives a possibility to use the game theory as an analysis tool.
- We provide some built-in data games to define the reward and aggregation to produce values.

Games

A game can be described with players, their options, and payoffs

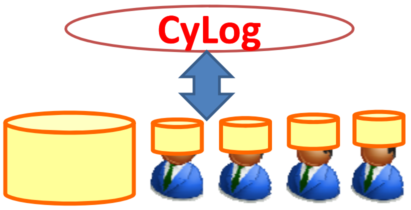
Ex1) payoff matrix of a simple ESP Game

Player A＼B	Term A	Term B
Term A	(1, 1) Solution	(0,0)
Term B	(0,0)	Solution (1,1)

Ex2) payoff matrix of a Q&A Service Game

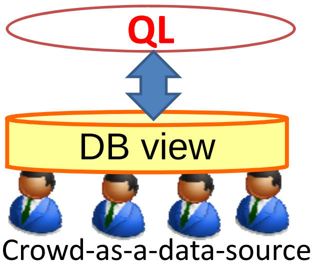
Player A＼B	Best Answer	Worst Answer
Best Answer	(15, 15) Solution	(30, 0)
Second Best Answer	(0,30)	(0,30)

Human-as-a-data-source



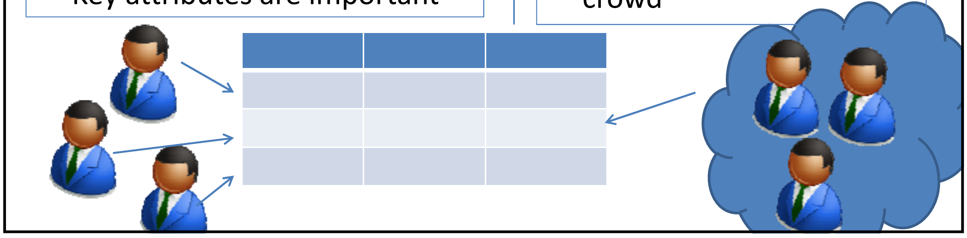
Human-as-a-data-source

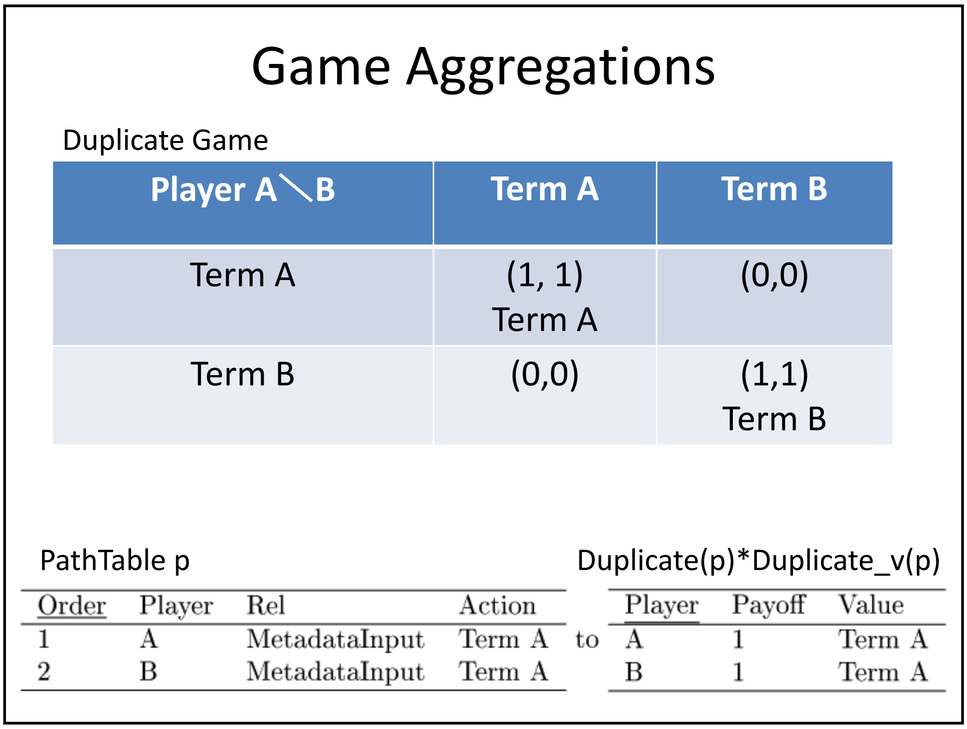
- Accept every input "as is"
- Implicit human-id attribute keywords(hid, img, keyword)
- Key attributes are important



Crowd-as-a-data-source

The DB view has the values computed by combining the inputs from the crowd





Game Aggregations

Duplicate Game

Player A＼B	Term A	Term B
Term A	(1, 1) Term A	(0,0)
Term B	(0,0)	(1,1) Term B

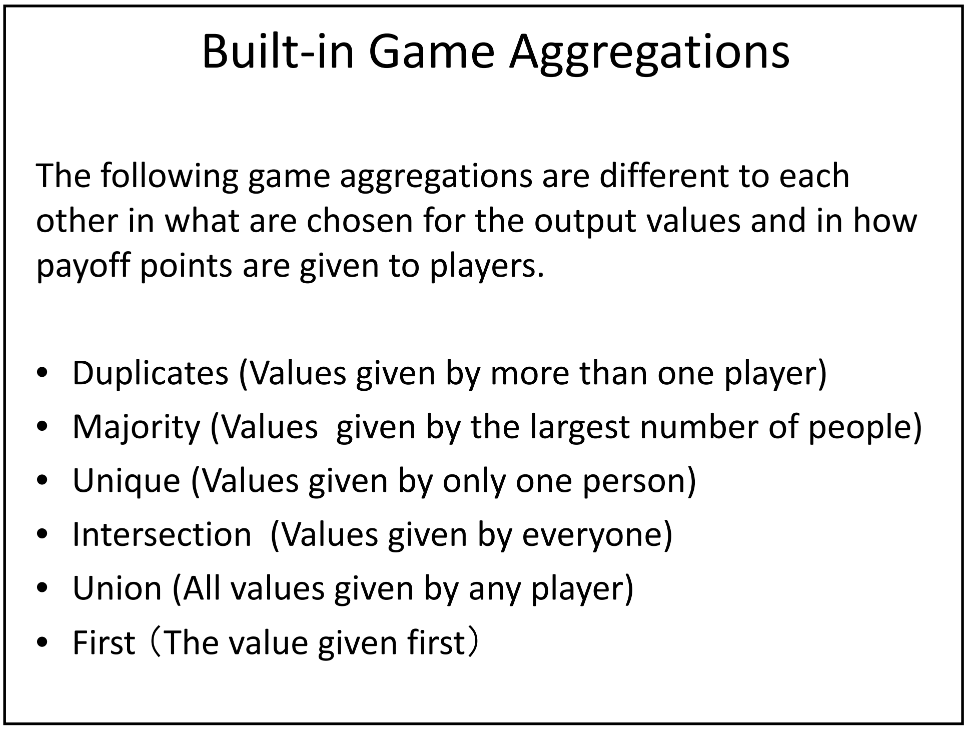
PathTable p

Order	Player	Rel	Action
1	A	MetadataInput	Term A
2	B	MetadataInput	Term A

to

Duplicate(p)*Duplicate_v(p)

Player	Payoff	Value
A	1	Term A
B	1	Term A



Built-in Game Aggregations

The following game aggregations are different to each other in what are chosen for the output values and in how payoff points are given to players.

- Duplicates (Values given by more than one player)
- Majority (Values given by the largest number of people)
- Unique (Values given by only one person)
- Intersection (Values given by everyone)
- Union (All values given by any player)
- First（The value given first）

Discussions on Data Games

- The data game concept is widely applicable beyond the real “games,” since there are many applications in which connecting dataflow with feedback to people is the key.
- How to deal with payoff points depends on applications
- We believe that the data game is a general concept
  - The games can be used to obtain the “correct” values,
  - They can be used to obtain values chosen based on other criteria
  - The data games can handle wider situations beyond the AMT-style crowdsourcing setting.

Example: Little Known Hot Spots

- Show (possibly a part of) the list of restaurant
- Label each restaurant as
  - L1: Good
  - L2: Not good
  - L3: I have never been there
- Give more points to people who labeled as “Good” those restaurants that are good on average but labeled as “I have never been there” by many people

Example: The ESP game in CyLog

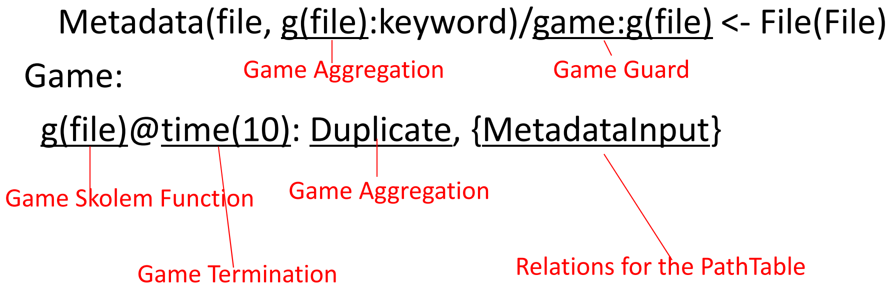
Data:

MetadataInput(file, keyword)/open <- File(file)

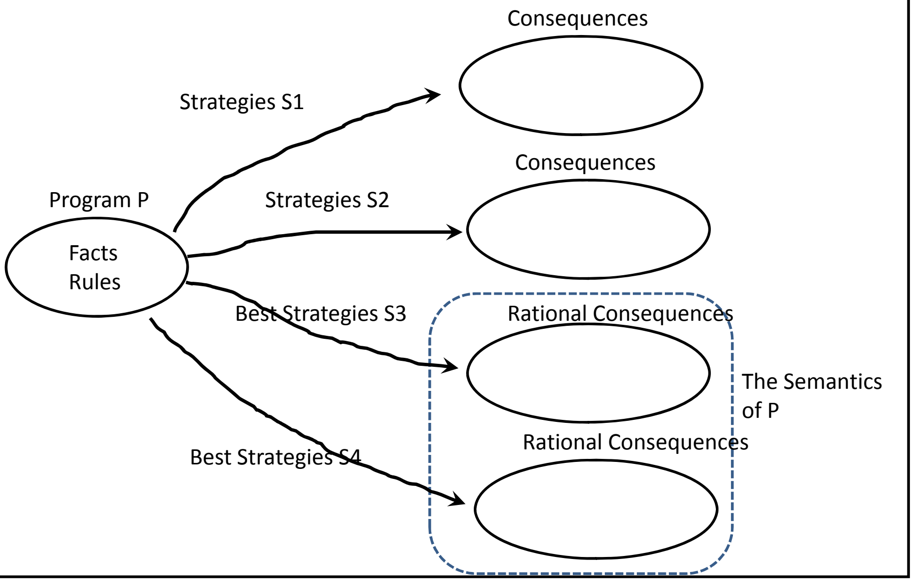
Metadata(file, g(file):keyword)/game:g(file) <- File(File)

Game:

g(file)@time(10): Duplicate, {MetadataInput}



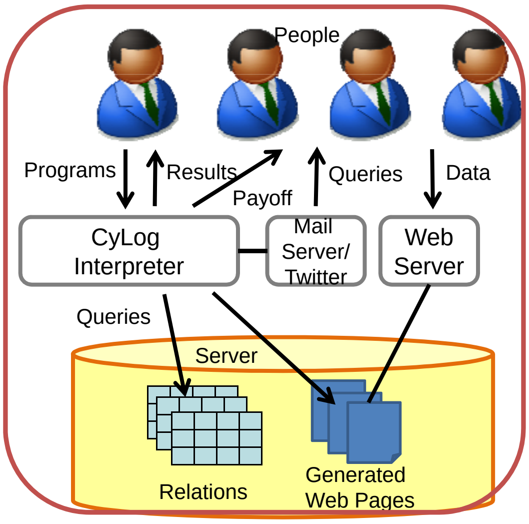
An Attempt to Define the Semantics of Cylog Programs



Outline

1. Background
2. CyLog
3. Prototype System
4. Related Work and Discussions

Prototype System



People

Programs

Results

Payoff

Queries

Data

CyLog Interpreter

Mail Server/ Twitter

Web Server

Queries

Server

Relations

Generated Web Pages

- The current working version of our prototype system provides a default function to generate HTML forms for open predicates
- External functions are allowed to implement complex algorithms and customized user interface
- Modules to work with AMT is under development

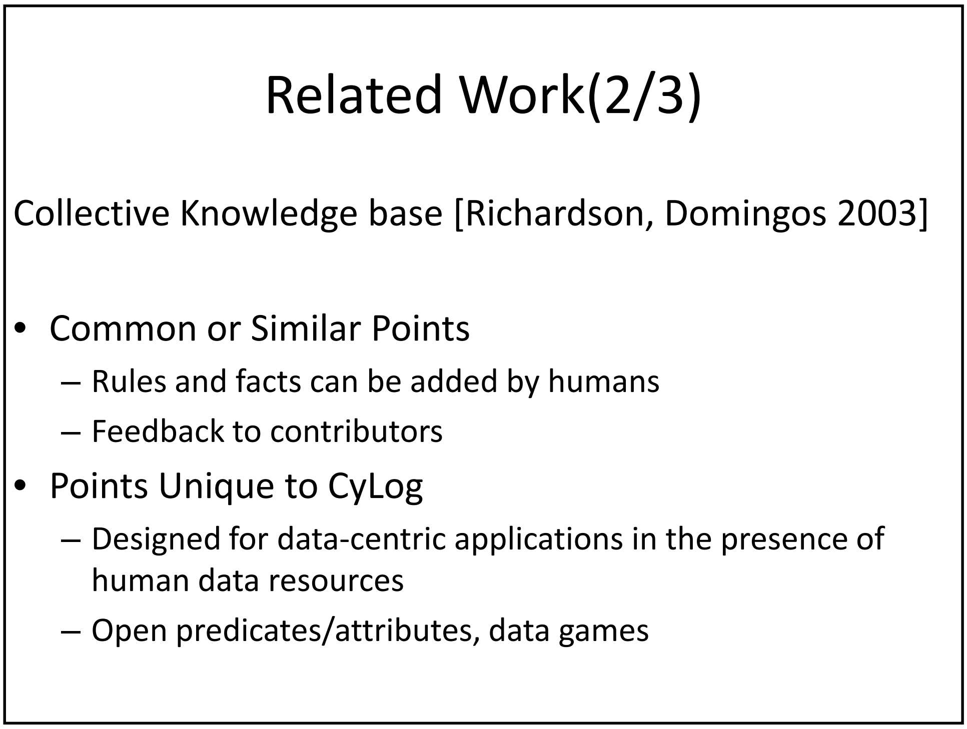
Outline

1. Background
2. CyLog
3. Prototype System
4. Related Work and Discussions

Related Work(1/3)

Recent Work: Qurk, sCOOP/hQuery, CrowdDB

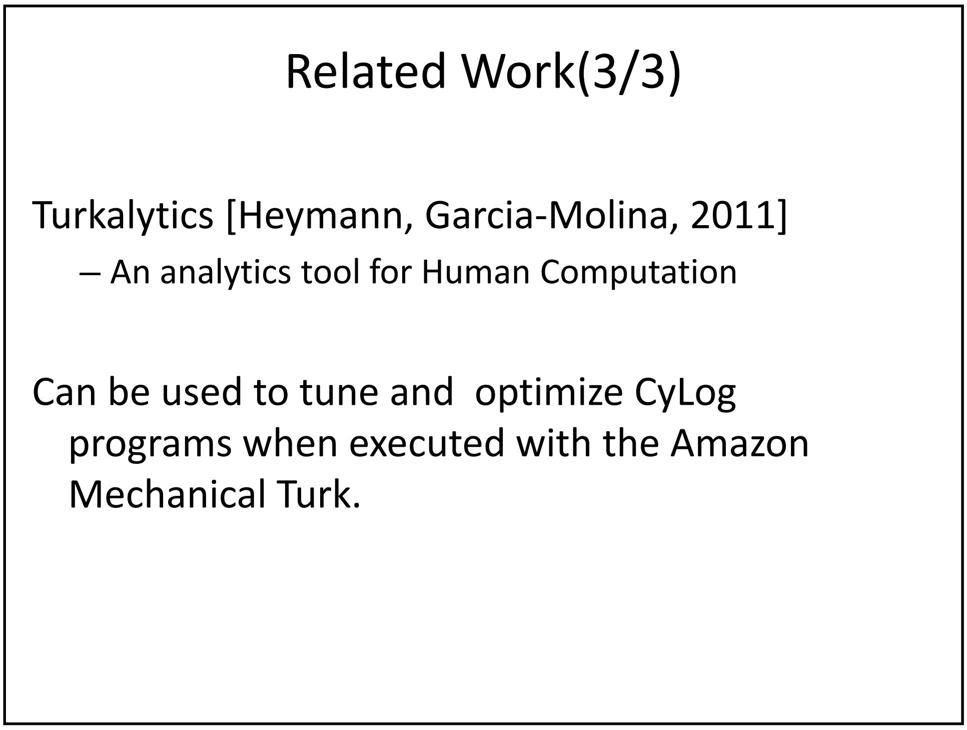
- Common or Similar Points
  - Declarative approach
  - Concepts similar to open predicates/attributes (hPred, CNULL,…)
- Points Unique to CyLog
  - Introduce rational data sources
  - Data games as a means to obtain appropriate values
  - Takes the human-as-a-data-sources approach to incorporate data games in the language.



Related Work(2/3)

Collective Knowledge base [Richardson, Domingos 2003]

- Common or Similar Points
  - Rules and facts can be added by humans
  - Feedback to contributors
- Points Unique to CyLog
  - Designed for data-centric applications in the presence of human data resources
  - Open predicates/attributes, data games



Related Work(3/3)

Turkalytics [Heymann, Garcia-Molina, 2011]

- An analytics tool for Human Computation

Can be used to tune and optimize CyLog programs when executed with the Amazon Mechanical Turk.

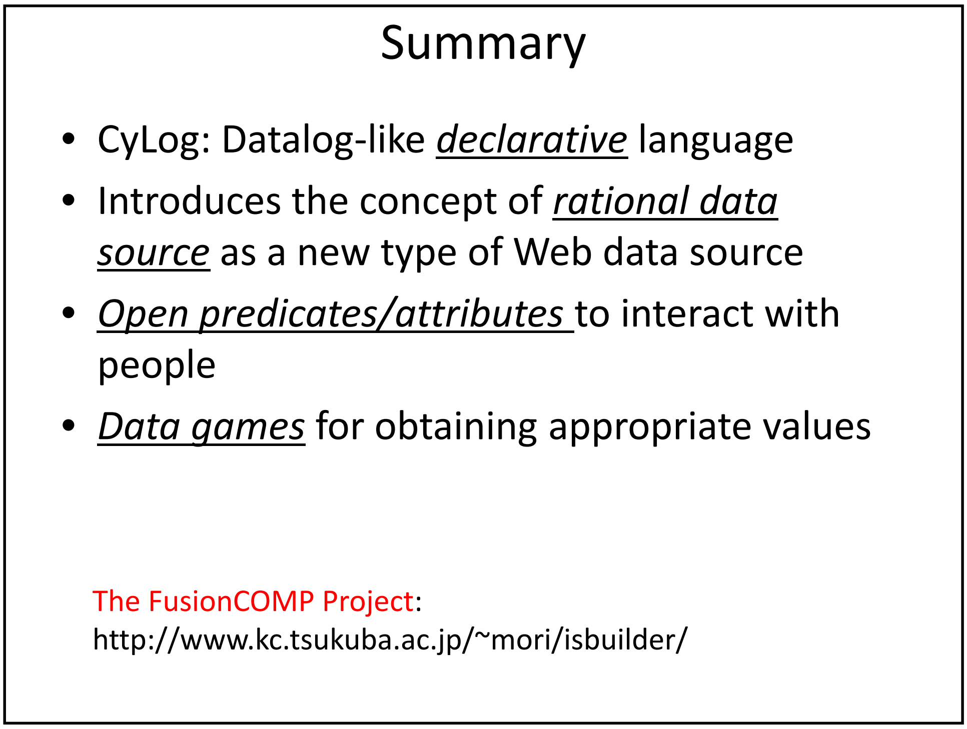
Open Problems

- Optimization issues
- Advanced mechanisms for player selection
- Development of various types of data-games
- Design theory
- Definitive rationality

Some of the above are addressed in the related work

The Current Status

- Updating and extending the syntax of CyLog
  - The basic idea is the same
  - Nest Structure for the concise description
  - Support of Status values for complex games
- Developing a software platform open to public



Summary

- CyLog: Datalog-like *declarative* language
- Introduces the concept of *rational data source* as a new type of Web data source
- *Open predicates/attributes* to interact with people
- *Data games* for obtaining appropriate values

The FusionCOMP Project:
http://www.kc.tsukuba.ac.jp/~mori/isbuilder/