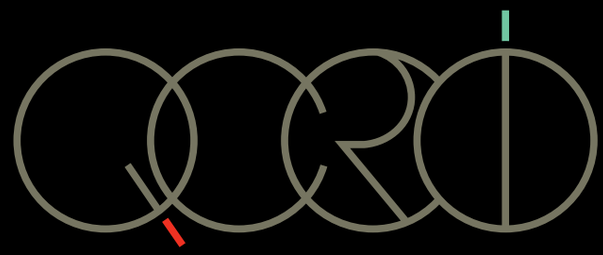
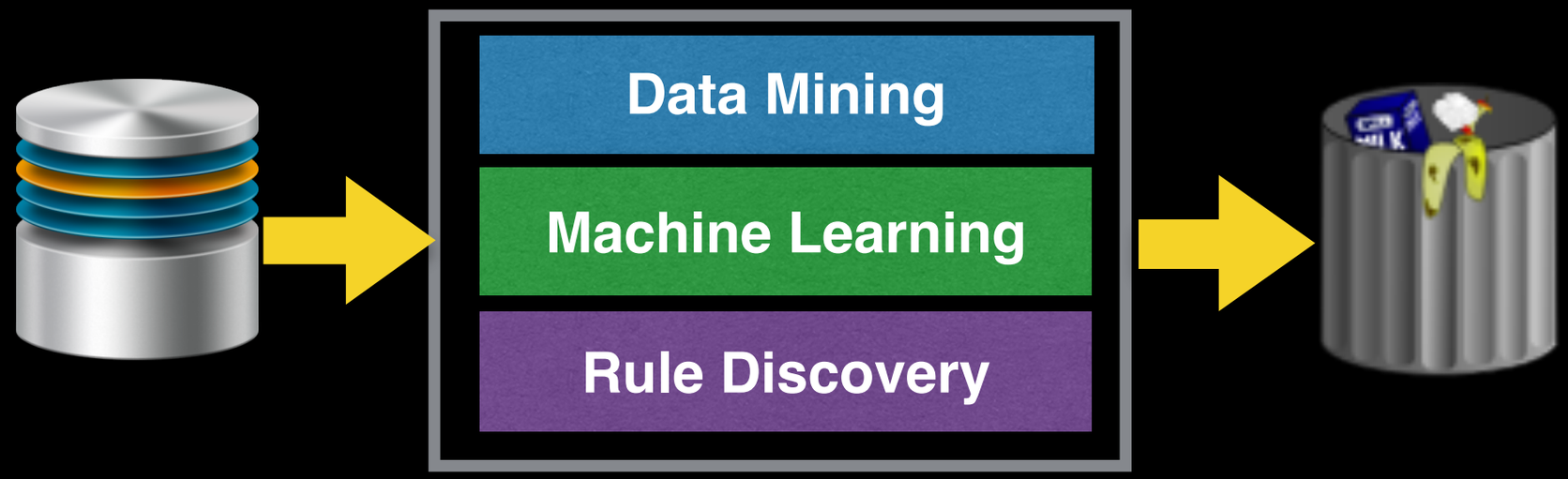


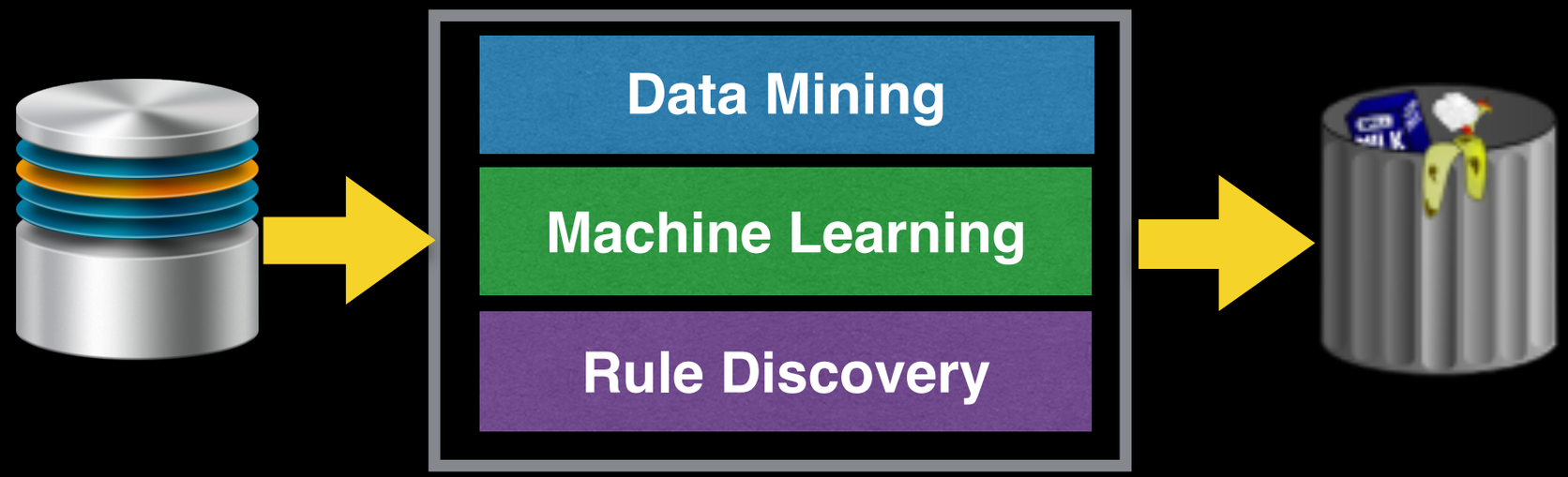
Big RDF Data Cleaning

Nan Tang

Roadblocks to Get Value from Data?

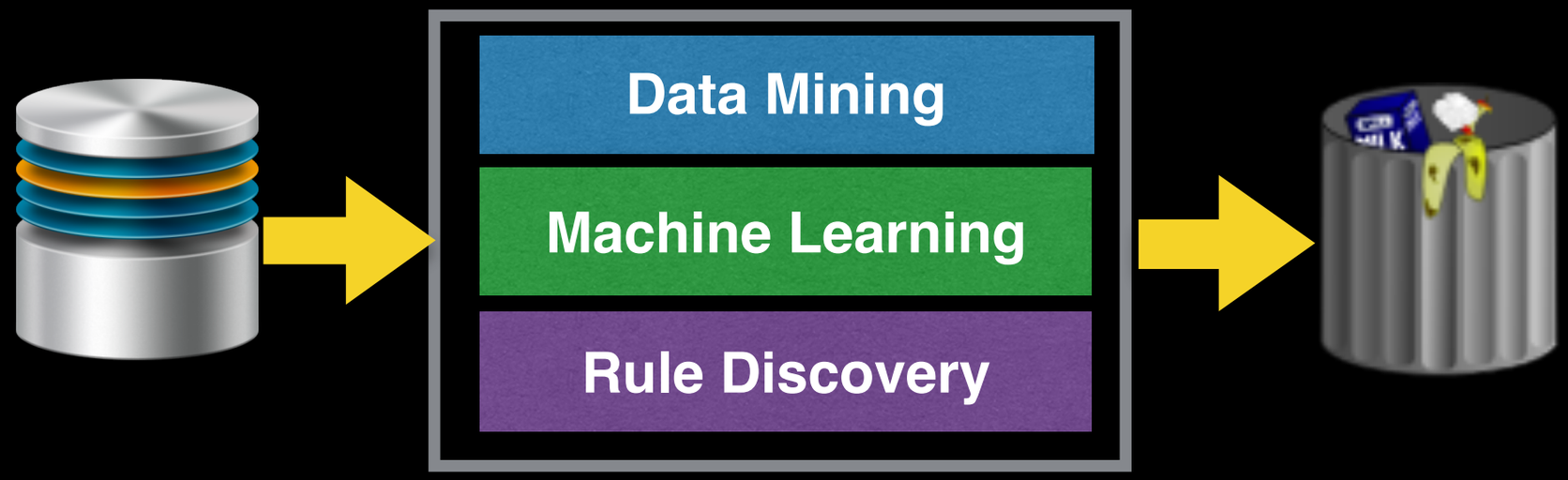


Roadblocks to Get Value from Data?



Roadblocks to Get Value from Data?

Data Quality and Consistency



Roadblocks to Get Value from Data?

Data Quality and Consistency

$3 Trillion Problem: Three Best P
Today's Dirty Data Pandemic

Maybe your software is healthy, but is your data terminally ill?

BY **HOLLIS TIBBETTS** ARTICLE RATING: ☆☆☆☆☆

SEPTEMBER 10, 2011 12:00 PM EDT READS: 17,513

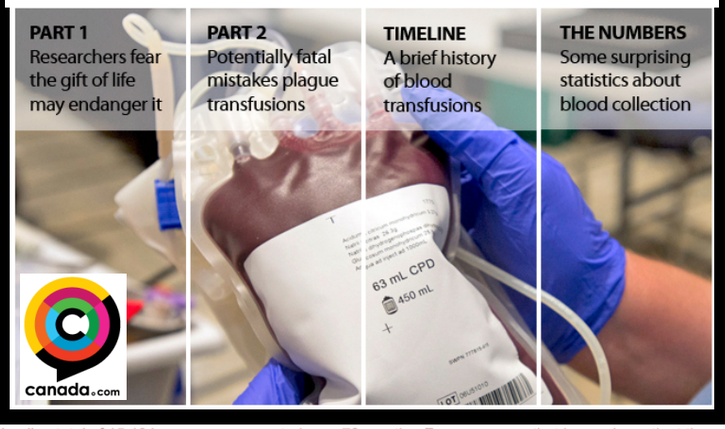
RELATED PRINT EMAIL FEEDBACK ADD THIS BLOG THIS

In survey after survey, about half of IT executives consistently agree that data quality and data consistency is one of the biggest roadblocks to them getting full value from their data.

This has been consistently true all since the Chinese invented the abacus. I suspect it will be true long after quantum computing has solved every other problem that humanity faces.

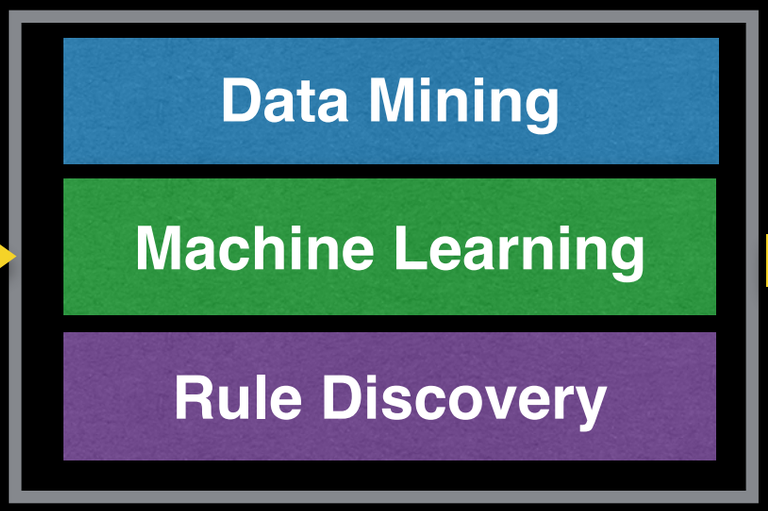
According to Gartner, "by 2017, 33 percent of Fortune 100 organizations will experience an information crisis, due to their inability to effectively value, govern and trust their enterprise information." These large organizations need to manage extensive amounts of data across numerous business units, often leading to unavoidable data quality issues. How do you get key stakeholders to really understand the impacts data quality has on the business?

New Canadian research raises concerns over number, types of transfusion errors



In all, a total of 15,134 errors were reported over 72 months. For every error that harmed a patient there were 657 errors that were detected and intercepted before the blood could reach the patient. "Wrong blood in tube" — blood drawn from the wrong patient for matching — occurred once in every 10,250 samples collected.

typos
normalization
incomplete
inconsistent
...



Roadblocks to Get Value from Data?

Data Quality and Consistency

$3 Trillion Problem: Three Best P
Today's Dirty Data Pandemic

Maybe your software is healthy, but is your data terminally ill?

BY **HOLLIS TIBBETTS** ARTICLE RATING: ☆☆☆☆☆

SEPTEMBER 10, 2011 12:00 PM EDT READS: 17,513

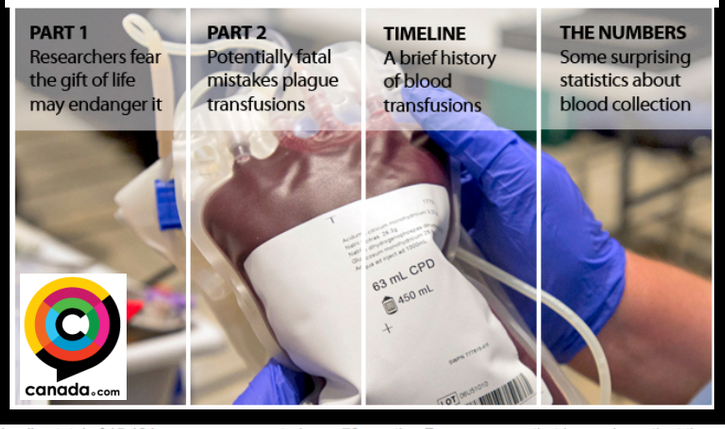
RELATED PRINT EMAIL FEEDBACK ADD THIS BLOG THIS

In survey after survey, about half of IT executives consistently agree that data quality and data consistency is one of the biggest roadblocks to them getting full value from their data.

This has been consistently true all since the Chinese invented the abacus. I suspect it will be true long after quantum computing has solved every other problem that humanity faces.

According to Gartner, "by 2017, 33 percent of Fortune 100 organizations will experience an information crisis, due to their inability to effectively value, govern and trust their enterprise information." These large organizations need to manage extensive amounts of data across numerous business units, often leading to unavoidable data quality issues. How do you get key stakeholders to really understand the impacts data quality has on the business?

New Canadian research raises concerns over number, types of transfusion errors



In all, a total of 15,134 errors were reported over 72 months. For every error that harmed a patient there were 657 errors that were detected and intercepted before the blood could reach the patient. "Wrong blood in tube" — blood drawn from the wrong patient for matching — occurred once in every 10,250 samples collected.

typos
normalization
incomplete
inconsistent
…

aligning instances/ontology

value/cardinality constraint
(OWL/RDFS)

Roadblocks to Get Value from Data?

Data Quality and Consistency

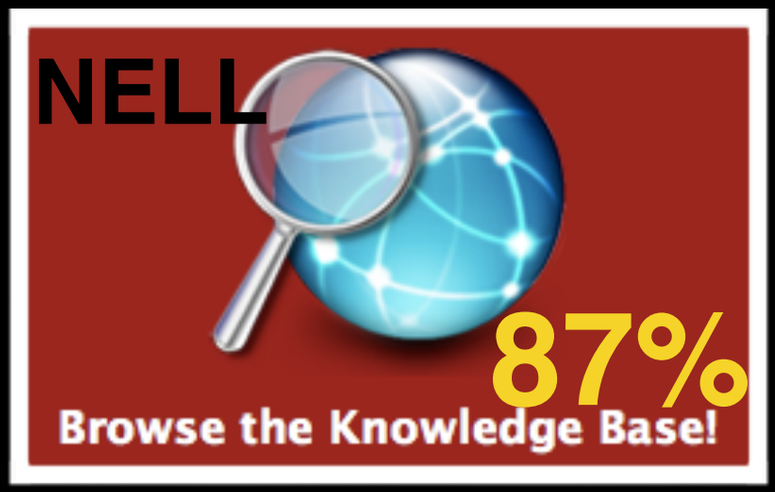
typos
normalization
incomplete
inconsistent
...

aligning instances/ontology

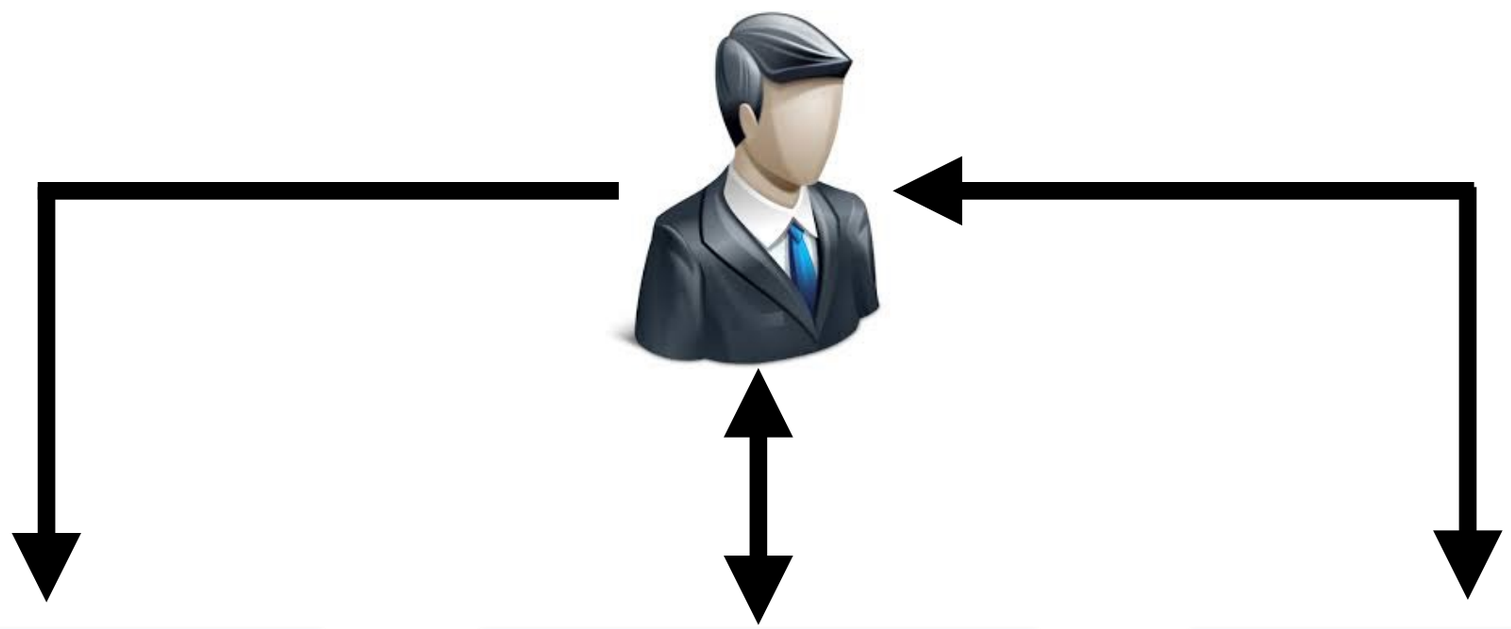
value/cardinality constraint
(OWL/RDFS)

Roadblocks to Get Value from Data?

Data Quality and Consistency



Data Cleaning



Data

Data Quality Rules

1. You SHALL!
2. You WILL!
3. You MUST!

Error Detection

Data Repair

Knowledge Matching

Knowledge Matching

name	institute	position	location
Nan Tang	QCRI	Scientist	Doha
Nan Tang	QCRI	Senior Scientist	Doha
Nan Tang	CWI	Postdoc	Netherlands
Stratos Idreos	CWI	PhD.	Amsterdam

Knowledge Matching

name	institute	position	location
Nan Tang	QCRI	Scientist	Doha
Nan Tang	QCRI	Senior Scientist	Doha
Nan Tang	CWI	Postdoc	Netherlands
Stratos Idreos	CWI	PhD.	Amsterdam

Knowledge Matching

name	institute	position	location
Nan Tang	QCRI	Scientist	Doha
Nan Tang	QCRI	Senior Scientist	Doha
Nan Tang	CWI	Postdoc	Netherlands
Stratos Idreos	CWI	PhD.	Amsterdam

Knowledge Matching

Biography

Dr. Nan Tang is a scientist at Qatar Computing Research Institute (QCRI) at Qatar Foundation, Doha, Qatar. Prior to joining QCRI in Dec. 2011, He was a Research Fellow at LFCS (Laboratory for Foundations of Computer Science) at the University of Edinburgh (2010--2011). He was a scientific staff member with the CWI (Dutch National Research Center for Mathematics and Computer Science), Amsterdam (2008--2010). He got his PhD. degree from The Chinese University of Hong Kong (2007). He holds a visiting position at University of Waterloo, Canada (03/2007-08/2007).

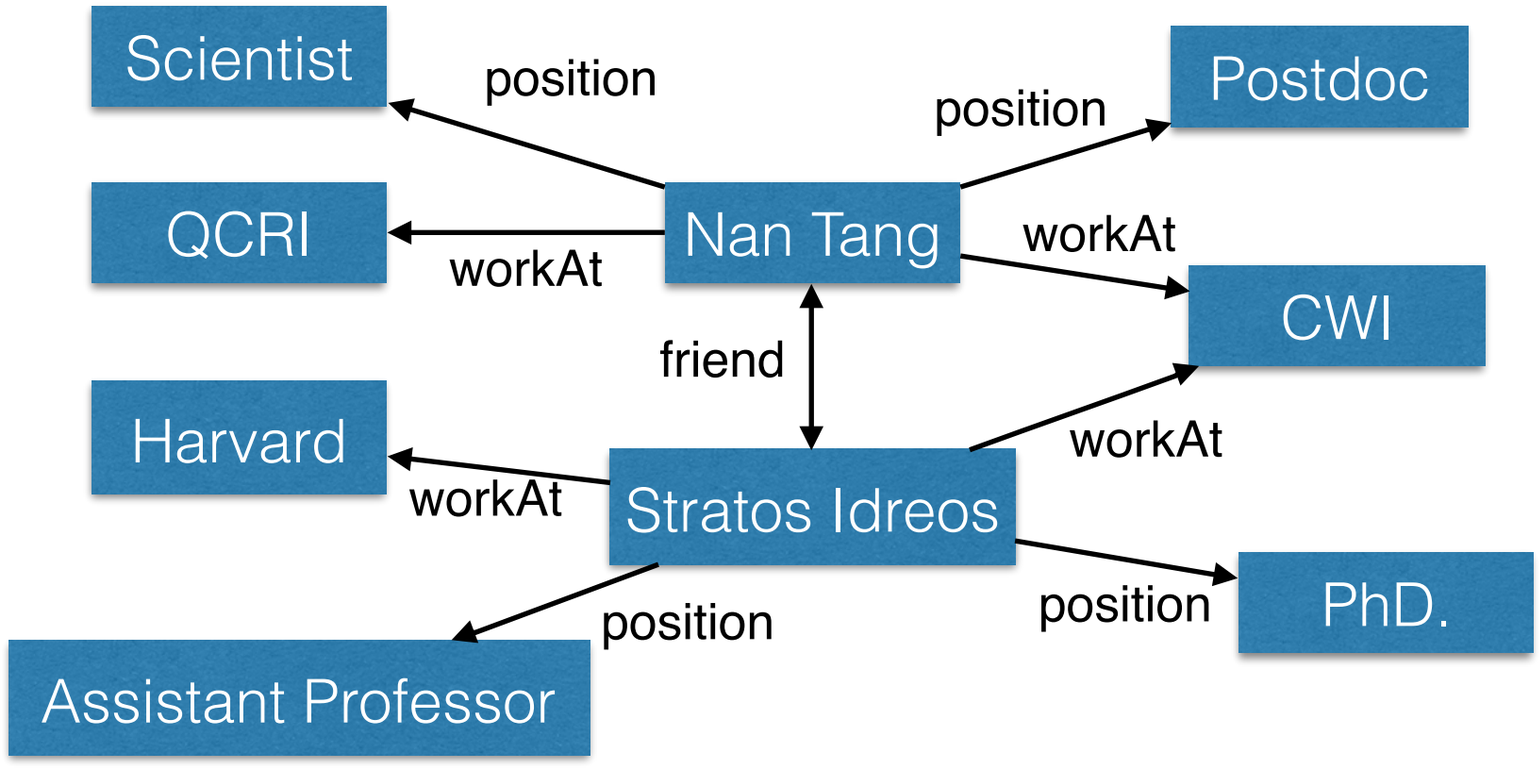
name	institute	position	location
Nan Tang	QCRI	Scientist	Doha
Nan Tang	QCRI	Senior Scientist	Doha
Nan Tang	CWI	Postdoc	Netherlands
Stratos Idreos	CWI	PhD.	Amsterdam

Knowledge Matching

Biography

Dr. Nan Tang is a scientist at Qatar Computing Research Institute (QCRI) at Qatar Foundation, Doha, Qatar. Prior to joining QCRI in Dec. 2011, He was a Research Fellow at LFCS (Laboratory for Foundations of Computer Science) at the University of Edinburgh (2010--2011). He was a scientific staff member with the CWI (Dutch National Research Center for Mathematics and Computer Science), Amsterdam (2008--2010). He got his PhD. degree from The Chinese University of Hong Kong (2007). He holds a visiting position at University of Waterloo, Canada (03/2007-08/2007).

name	institute	position	location
Nan Tang	QCRI	Scientist	Doha
Nan Tang	QCRI	Senior Scientist	Doha
Nan Tang	CWI	Postdoc	Netherlands
Stratos Idreos	CWI	PhD.	Amsterdam



KATARA

knowledge bases

crowdsourcing

BIGDANSING

big data cleansing

RDF data repair

KATARA in Action

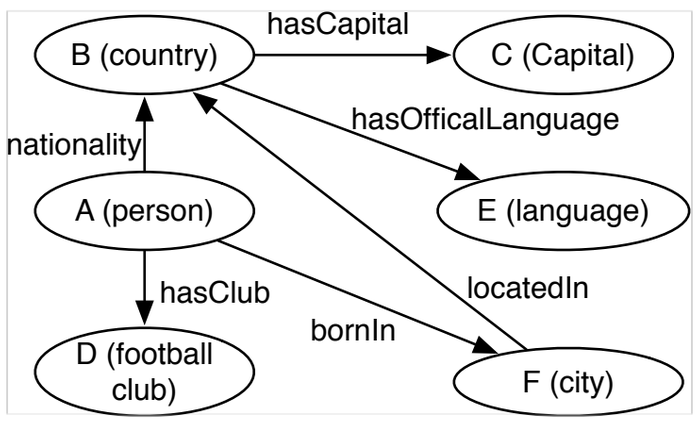
	A	B	C	D	E	F	G
t_1	Rossi	Italy	Rome	Verona	Italian	Proto	1.78
t_2	Klate	S. Africa	Pretoria	Pirates	Afrikaans	P. Eliz.	1.69
t_3	Pirlo	Italy	Madrid	Juve	Italian	Flero	1.77

yago
select knowledge

KATARA in Action

	A	B	C	D	E	F	G
t_1	Rossi	Italy	Rome	Verona	Italian	Proto	1.78
t_2	Klate	S. Africa	Pretoria	Pirates	Afrikaans	P. Eliz.	1.69
t_3	Pirlo	Italy	Madrid	Juve	Italian	Flero	1.77

table pattern

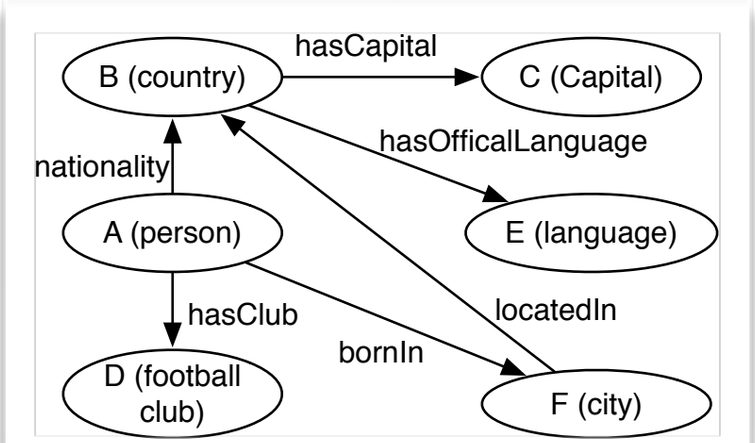


KATARA in Action

	A	B	C	D	E	F	G
t_1	Rossi	Italy	Rome	Verona	Italian	Proto	1.78
t_2	Klate	S. Africa	Pretoria	Pirates	Afrikaans	P. Eliz.	1.69
t_3	Pirlo	Italy	Madrid	Juve	Italian	Flero	1.77

yago
select knowledge

table pattern



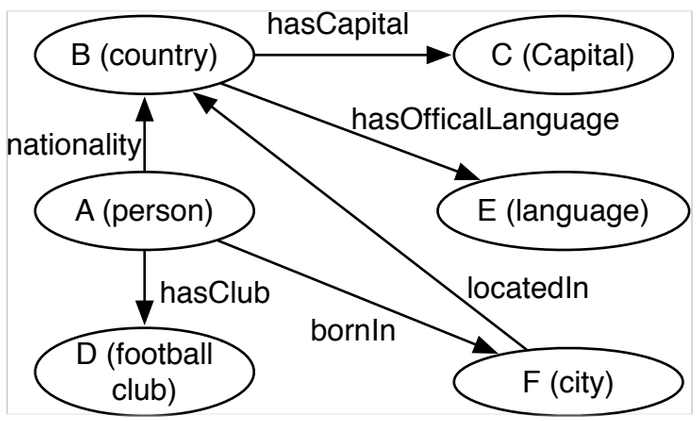
Q_1 : *What is the most accurate type of the highlighted column?*
(A, B, C, D, E, F, ...)
(Rossi, Italy, Rome, Verona, Italian, Proto, ...)
(Pirlo, Italy, Madrid, Juve, Italian, Flero,, ...)
◯ country ◯ economy
◯ state ◯ none of the above

Q_2 : *What is the most accurate relationship for*
highlighted columns (A, B, C, D, E, F, ...)
(Rossi, Italy, Rome, Verona, Italian, Proto, ...)
(Pirlo, Italy, Madrid, Juve, Italian, Flero, ...)
◯ B hasCapital C ◯ C locatedIn B ◯ none of the above

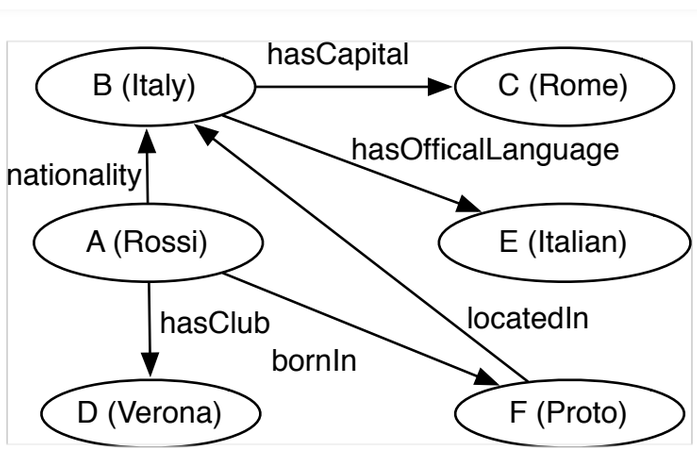
KATARA in Action

	A	B	C	D	E	F	G
t_1	Rossi	Italy	Rome	Verona	Italian	Proto	1.78
t_2	Klate	S. Africa	Pretoria	Pirates	Afrikaans	P. Eliz.	1.69
t_3	Pirlo	Italy	Madrid	Juve	Italian	Flero	1.77

table pattern



t1

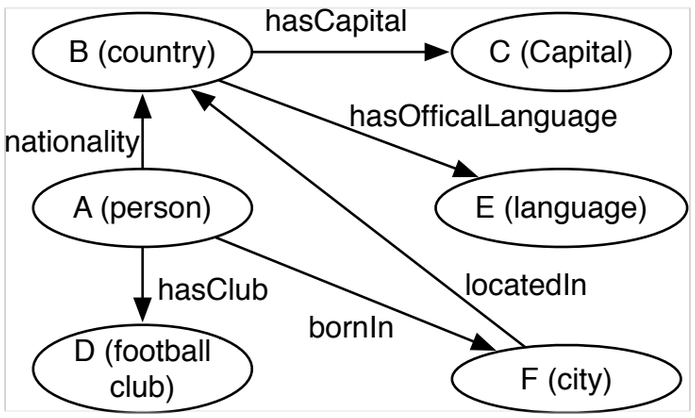


KATARA in Action

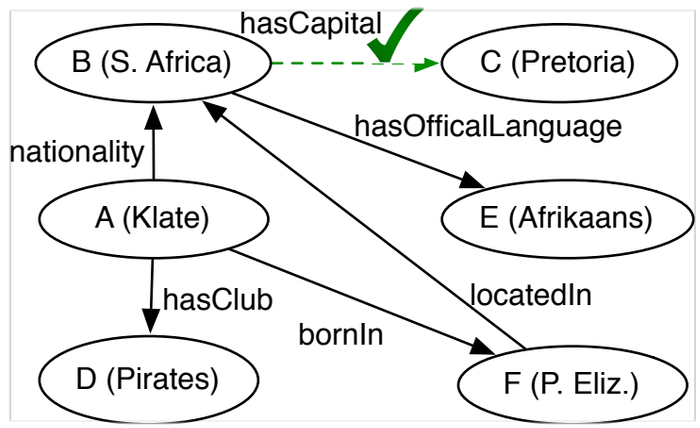
	A	B	C	D	E	F	G
t_1	Rossi	Italy	Rome	Verona	Italian	Proto	1.78
t_2	Klate	S. Africa	Pretoria	Pirates	Afrikaans	P. Eliz.	1.69
t_3	Pirlo	Italy	Madrid	Juve	Italian	Flero	1.77

yago
select knowledge

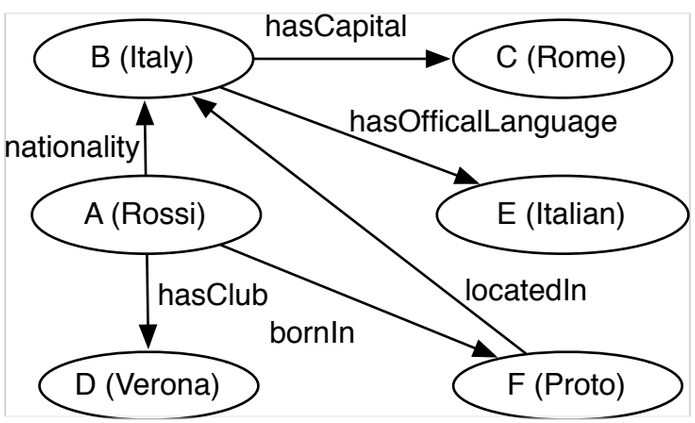
table pattern



t2



t1

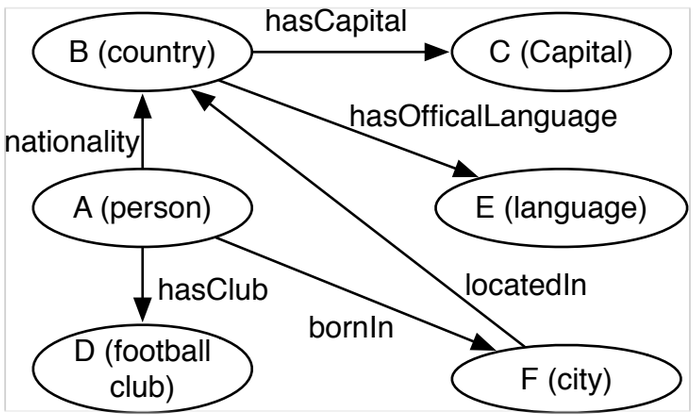


KATARA in Action

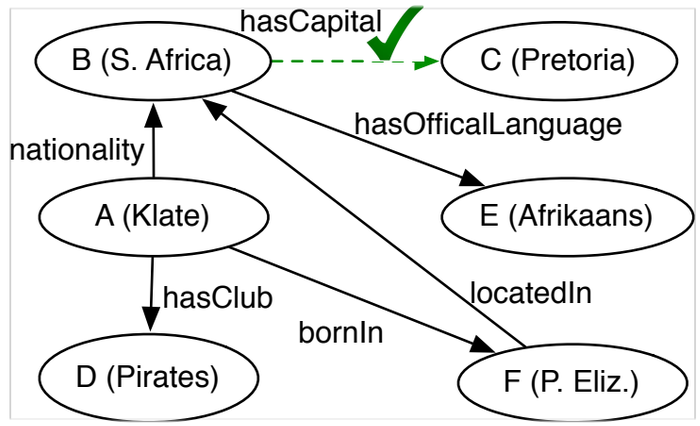
	A	B	C	D	E	F	G
t_1	Rossi	Italy	Rome	Verona	Italian	Proto	1.78
t_2	Klate	S. Africa	Pretoria	Pirates	Afrikaans	P. Eliz.	1.69
t_3	Pirlo	Italy	Madrid	Juve	Italian	Flero	1.77

yago
select knowledge

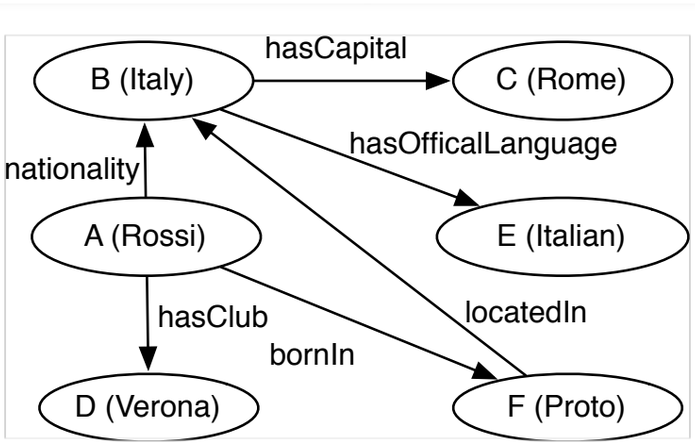
table pattern



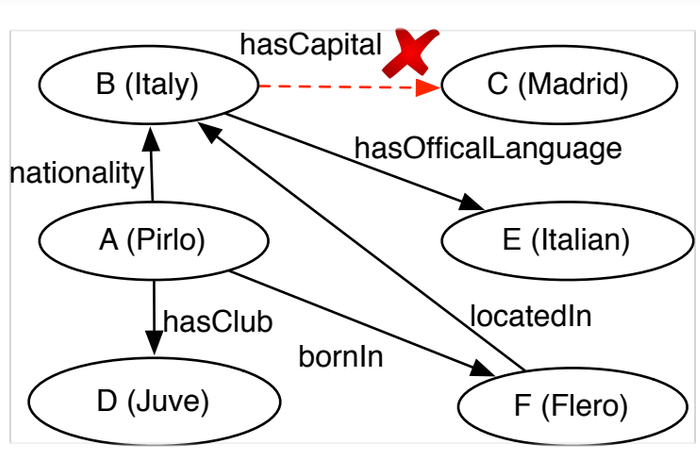
t2



t1



t3



BIGDANSING: Big Data Cleansing

ETLs
CFDs
MDs
Business rules

BIGDANSING

Rule Compiler

Violation Detection

Data Repairing

BIGDANSING: Big Data Cleansing

ETLs
CFDs
MDs
Business rules

Scope
Iterate
Block
Detect
GenFix

Scope
Iterate
Block
Detect
GenFix

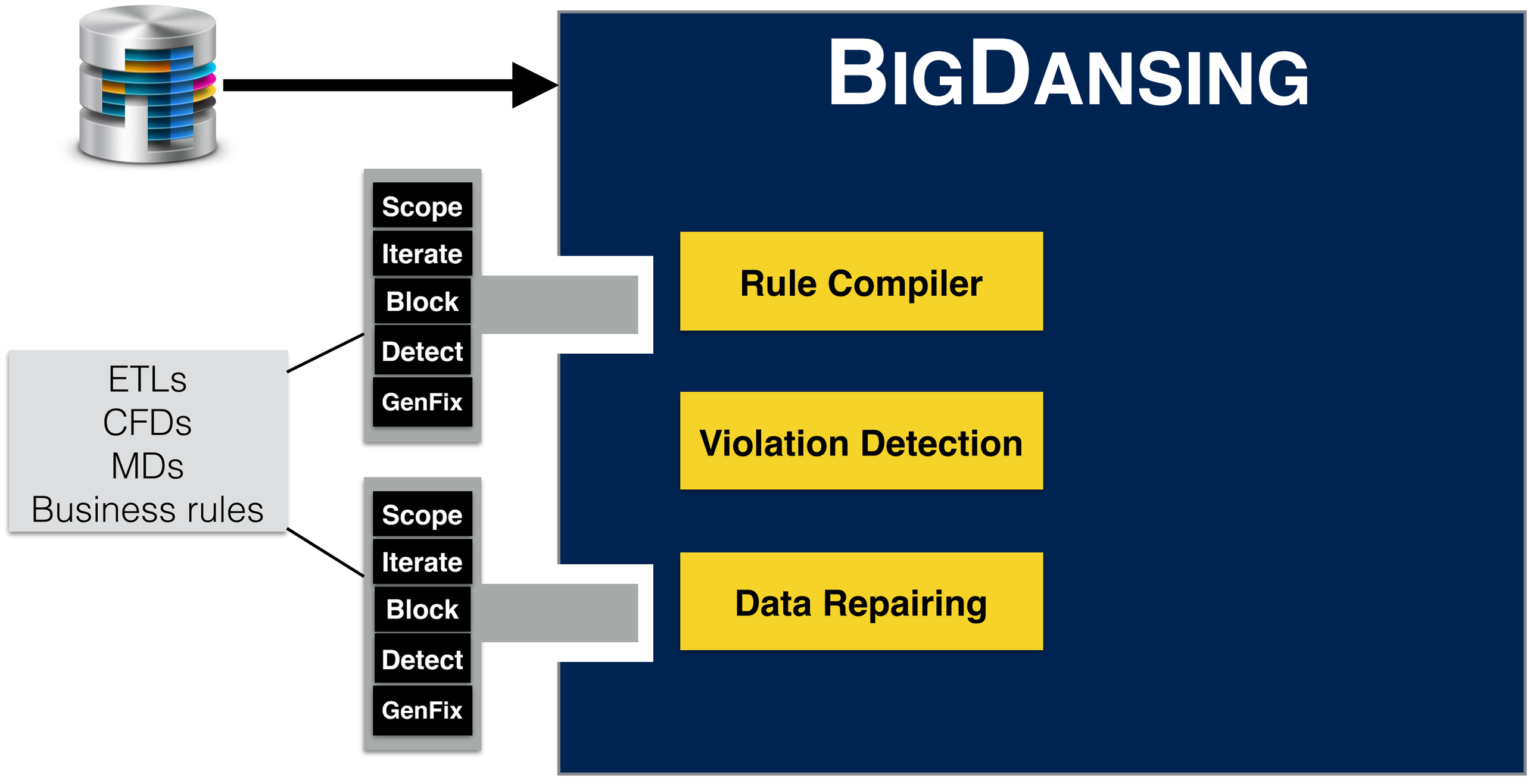
BIGDANSING

Rule Compiler

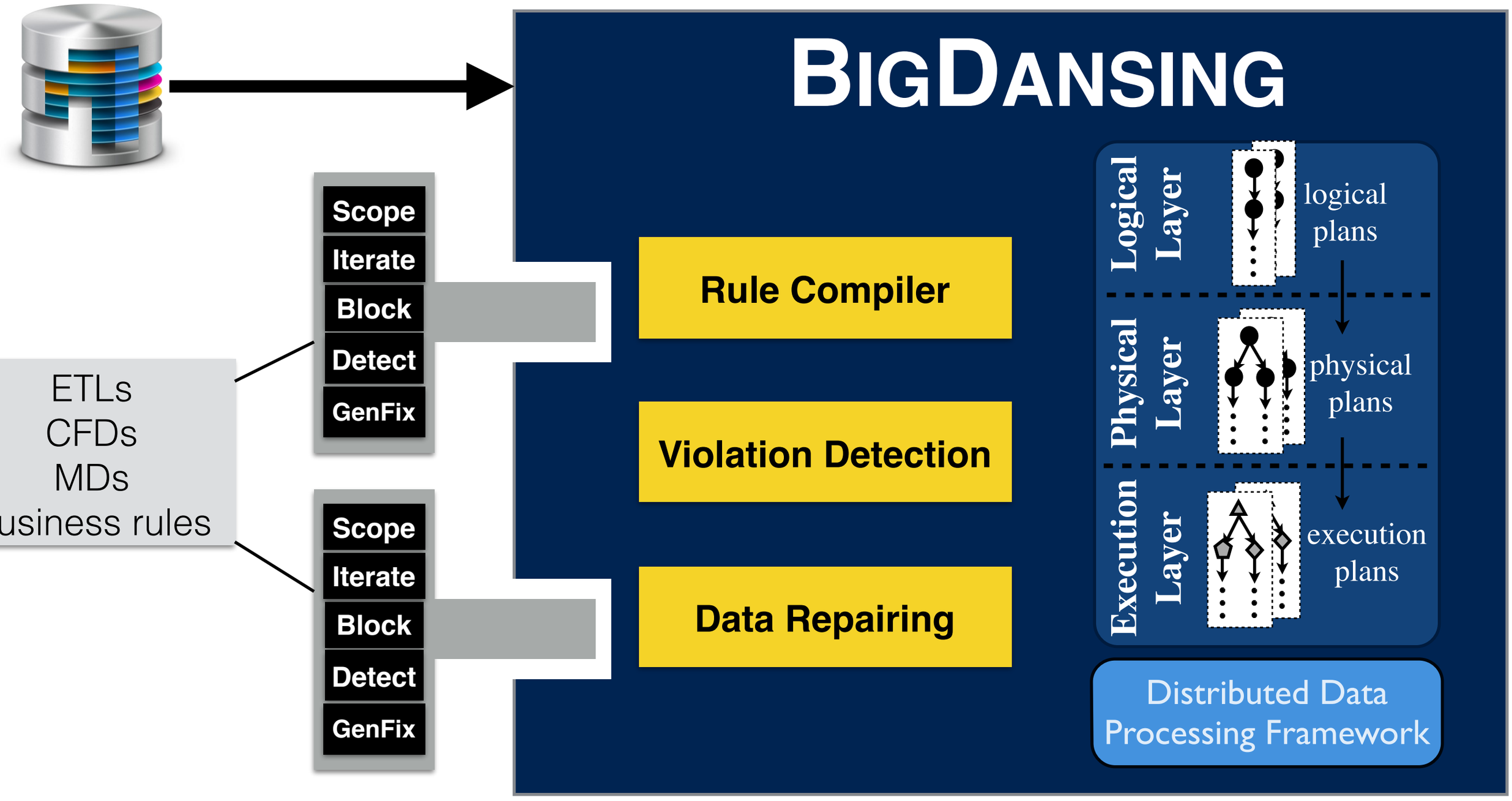
Violation Detection

Data Repairing

BIGDANSING: Big Data Cleansing



BIGDANSING: Big Data Cleansing



BigDansing Interfaces

Rule Editor

- Detect
- GenFix
- Scope
- Block
- Iterate

```
8     @Override
9     public Collection<Violation> detect(TuplePair tuplePair) {
10        List<Violation> result = new ArrayList<>();
11        Tuple left = tuplePair.getLeft();
12        Tuple right = tuplePair.getRight();
13
14        if (
15            Metrics.getEqual(
16                left.get("name"), right.get("name")) == 1.0 &&
17            Metrics.getLevenshtein(
18                left.get("address"), right.get("address")) > 0.8 &&
19            Metrics.getEqual(
20                left.get("gender"), right.get("gender")) == 1.0
21        ) {
22            Violation v = new Violation(getRuleName());
23            v.addTuple(left);
24            v.addTuple(right);
25            result.add(v);
26        }
27        return result;
28    }
29
30
```

Close | Save changes

BIGDANSING in Action: Repair RDF Data

BigDansing in Action: Repair RDF Data

There cannot exist two students in different universities with the same advisor:
(Paul, John) and (Paul, Sally)

S	P	O
Paul	student_in	Yale
John	student_in	UCLA
Sally	student_in	UCLA
William	professor_in	UCLA
Paul	advised_by	William
John	advised_by	William
Sally	advised_by	William

BigDansing in Action: Repair RDF Data

There cannot exist two students in different universities with the same advisor: (Paul, John) and (Paul, Sally)

S	P	O
Paul	student_in	Yale
John	student_in	UCLA
Sally	student_in	UCLA
William	professor_in	UCLA
Paul	advised_by	William
John	advised_by	William
Sally	advised_by	William

Scope (S, O, student_in, advised_by)

S	O
Paul	Yale
John	UCLA
Sally	UCLA
Paul	William
John	William
Sally	William

BIGDANSING in Action: Repair RDF Data

There cannot exist two students in different universities with the same advisor: (Paul, John) and (Paul, Sally)

S	P	O
Paul	student_in	Yale
John	student_in	UCLA
Sally	student_in	UCLA
William	professor_in	UCLA
Paul	advised_by	William
John	advised_by	William
Sally	advised_by	William

Scope (S, O, student_in, advised_by)

S	O
Paul	Yale
John	UCLA
Sally	UCLA
Paul	William
John	William
Sally	William

Block (subject)

S	O
Paul	Yale
Paul	William
Sally	UCLA
Sally	William
John	William
John	UCLA

BigDansing in Action: Repair RDF Data

There cannot exist two students in different universities with the same advisor: (Paul, John) and (Paul, Sally)

S	P	O
Paul	student_in	Yale
John	student_in	UCLA
Sally	student_in	UCLA
William	professor_in	UCLA
Paul	advised_by	William
John	advised_by	William
Sally	advised_by	William

Scope (S, O, student_in, advised_by)

S	O
Paul	Yale
John	UCLA
Sally	UCLA
Paul	William
John	William
Sally	William

Block (subject)

S	O
Paul	Yale
Paul	William
Sally	UCLA
Sally	William
John	William
John	UCLA

Iterate

S	O1	O2
Paul	William	Yale
John	William	UCLA
Sally	William	UCLA

BigDansing in Action: Repair RDF Data

There cannot exist two students in different universities with the same advisor: (Paul, John) and (Paul, Sally)

S	P	O
Paul	student_in	Yale
John	student_in	UCLA
Sally	student_in	UCLA
William	professor_in	UCLA
Paul	advised_by	William
John	advised_by	William
Sally	advised_by	William

Scope (S, O, student_in, advised_by)

S	O
Paul	Yale
John	UCLA
Sally	UCLA
Paul	William
John	William
Sally	William

Block (subject)

S	O
Paul	Yale
Paul	William
Sally	UCLA
Sally	William
John	William
John	UCLA

Iterate

S	O1	O2
Paul	William	Yale
John	William	UCLA
Sally	William	UCLA

Block (Object1)

O1	S	O2
William	Paul	Yale
William	John	UCLA
William	Sally	UCLA

BigDansing in Action: Repair RDF Data

There cannot exist two students in different universities with the same advisor: (Paul, John) and (Paul, Sally)

S	P	O
Paul	student_in	Yale
John	student_in	UCLA
Sally	student_in	UCLA
William	professor_in	UCLA
Paul	advised_by	William
John	advised_by	William
Sally	advised_by	William

Scope (S, O, student_in, advised_by)

S	O
Paul	Yale
John	UCLA
Sally	UCLA
Paul	William
John	William
Sally	William

Block (subject)

S	O
Paul	Yale
Paul	William
Sally	UCLA
Sally	William
John	William
John	UCLA

Iterate

S	O1	O2
Paul	William	Yale
John	William	UCLA
Sally	William	UCLA

Block (Object1)

O1	S	O2
William	Paul	Yale
William	John	UCLA
William	Sally	UCLA

Iterate

(Paul, *Yale*, John, *UCLA*)
(Paul, *Yale*, Sally, *UCLA*)
(John, *UCLA*, Sally, *UCLA*)

BIGDANSING in Action: Repair RDF Data

There cannot exist two students in different universities with the same advisor: (Paul, John) and (Paul, Sally)

S	P	O
Paul	student_in	Yale
John	student_in	UCLA
Sally	student_in	UCLA
William	professor_in	UCLA
Paul	advised_by	William
John	advised_by	William
Sally	advised_by	William

Scope (S, O, student_in, advised_by)

S	O
Paul	Yale
John	UCLA
Sally	UCLA
Paul	William
John	William
Sally	William

Block (subject)

S	O
Paul	Yale
Paul	William
Sally	UCLA
Sally	William
John	William
John	UCLA

Iterate

S	O1	O2
Paul	William	Yale
John	William	UCLA
Sally	William	UCLA

Block (Object1)

O1	S	O2
William	Paul	Yale
William	John	UCLA
William	Sally	UCLA

Iterate

(Paul, *Yale*, John, *UCLA*)
(Paul, *Yale*, Sally, *UCLA*)
(John, *UCLA*, Sally, *UCLA*)

Detect

(Paul, *Yale*, John, *UCLA*)
(Paul, Yale, Sally, *UCLA*)

BigDansing in Action: Repair RDF Data

There cannot exist two students in different universities with the same advisor:
(Paul, John) and (Paul, Sally)

S	P	O
Paul	student_in	Yale
John	student_in	UCLA
Sally	student_in	UCLA
William	professor_in	UCLA
Paul	advised_by	William
John	advised_by	William
Sally	advised_by	William

Scope (S, O, student_in, advised_by)

S	O
Paul	Yale
John	UCLA
Sally	UCLA
Paul	William
John	William
Sally	William

Block (subject)

S	O
Paul	Yale
Paul	William
Sally	UCLA
Sally	William
John	William
John	UCLA

Iterate

S	O1	O2
Paul	William	Yale
John	William	UCLA
Sally	William	UCLA

Block (Object1)

O1	S	O2
William	Paul	Yale
William	John	UCLA
William	Sally	UCLA

Iterate

(Paul, *Yale*, John, *UCLA*)
(Paul, *Yale*, Sally, *UCLA*)
(John, *UCLA*, Sally, *UCLA*)

Detect

(Paul, *Yale*, John, *UCLA*)
(Paul, Yale, Sally, *UCLA*)

GenFix

Paul.Yale = John.UCLA
Paul.Yale = Sally.UCLA

Open Problems

- Rule based RDF cleaning
- Constraint based RDF cleaning
- Using master knowledge to clean RDF data
- Interactive RDF data cleaning