README
Consistent Updating of Databases with Marked Nulls.
Jacques Chabin, Mirian Halfeld Ferrari, Dominique Laurent

Section 1) How to install the prototype

++ You need java on your computer in order to run our program

Then you just need to follow the steps below:

1) Download UpdateChase.jar

2) On a terminal, execute

java -jar UpdateChase.jar

Section 2) How to use the interface
---

4 Tabs are proposed on the top of the screen

TAB 1) Chase:

- The screen is composed of 3 parts.
- In the first part you can insert the atoms in the database instance
- In the second part you can insert the constraints
- The third part shows the result obtained by running the chase algorithm
- (button on the top)

Syntaxe rules
- Variable should start with an X
- Null values should start with _N
- Atoms in the constraint body should be separated by a comma
- The body and the head of a constraint are separated by ->
- Predicates marked with - are hose to be deleted when performing backward processing

---
EXAMPLE

1) Source database (add each atom - bottom of the screen, part 1)

Attends(Ann, VLDB18)
Attends(Bob, _N1)
Conf(_N1)
Conf(VLDB18)
Researcher(Ann)
Researcher(Bob)

2) Constraints (add each constraint - bottom of the screen, part 2)

Conf(X1) -> Loc(X1, X2)

Loc(X1, X2) -> VisaReg(X2, X3)

Attends-(X1, X2), Conf(X2) -> Registered(X1, X2)

The database instance obtained after running the chase, appear on part 3 of the screen. New added atoms are marked.

For the example above, the result will be:

Attends('Ann', 'VLDB18')
Attends('Bob', '_N1#0#0')
Loc('_N1#0#0', '_X2New3_1*#1')
Loc('VLDB18', '_X2New3_2#1')
Conf('_N1#0#0')
Conf('VLDB18')
Researcher('Ann')
Researcher('Bob')
VisaReg('_X2New3_2#1', '_X3New5_2*#2')
VisaReg('_X2New3_1*#1', '_X3New5_1*#2')
Registered('Ann', 'VLDB18')
Registered('Bob', '_N1#0#0')

To continue working on this new database instance, you should ACCEPT the chase result (see button on the bottom of the screen, part 3)

TAB 2) Core:

The screen has 2 parts. The database obtained after the chase computation appears on the left (if you have accepted it).

After running the core algorithm (button on the top)
the new database instance is shown.

EXAMPLE: For the example above, the result will be (nothing is changed here):

Attends('Ann', 'VLDB18')
Attends('Bob', '_N1#0#0')
Loc('_N1#0#0', '_X2New3_1*#1')
Loc('VLDB18', '_X2New3_2#1')
Conf('_N1#0#0')
Conf('VLDB18')
Researcher('Ann')
Researcher('Bob')
VisaReg('_X2New3_2#1', '_X3New5_2*#2')
VisaReg('_X2New3_1*#1', '_X3New5_1*#2')
Registered('Ann', 'VLDB18')
Registered('Bob', '_N1#0#0')

TAB 3) Insertion:

- Insert atoms on the top right part of the screen.
- Run insertion algorithm
- Accept the new database. Two options are given:

    - accept without performing the core
    - accept performing the core

EXAMPLE
In our example, insert the following atoms:

Insertion 1:
a) insert Loc('VLDB2018', 'Rio')

b) Run Insert Algorithm: the algorithms proposes to insert the following atoms:

Loc('VLDB2018', 'Rio')
VisaReg('Rio', '_X3New6_3#1')

c) Then we have 2 options

c.1) Accept: Atoms are just added (we should come back to TAB Core to run core)

c.2) Accept and run core: insertion and core algorithm are performed

Resulting database:

Attends('Ann', 'VLDB18')

Attends('Bob', '_N1#0')
Loc('VLDB18', '_X2New2_2#1')
Loc('VLDB2018', 'Rio')
Conf('_N1#0')
Conf('VLDB18')
Researcher('Ann')
Researcher('Bob')
VisaReg('_X2New2_2#1', '_X3New4_2*#2')
VisaReg('Rio', '_X3New6_3#1')
Registered('Bob', '_N1#0')
Registered('Ann', 'VLDB18')

Insertion 2:
a) Insert
Attends(Bob, PODS2019)
Conf(PODS2019)
Loc(PODS2019,Rio)

b) Run insertion algorithm. The following side effects are computed

Attends('Bob', 'PODS2019')
Loc('PODS2019', 'Rio')
Conf('PODS2019')
Registered('Bob', 'PODS2019')

c) Accept + core gives

Attends('Ann', 'VLDB18')
Attends('Bob', 'PODS2019')
Loc('VLDB18', '_X2New2_2#1')
Loc('VLDB2018', 'Rio')
Loc('PODS2019', 'Rio')
Conf('PODS2019')
Conf('VLDB18')
Researcher('Ann')
Researcher('Bob')
VisaReg('_X2New2_2#1', '_X3New4_2*#2')
VisaReg('Rio', '_X3New6_3#1')
Registered('Bob', 'PODS2019')
Registered('Ann', 'VLDB18')

TAB 4) Deletion:

- Insert atoms to be deleted on the top right part of the screen.
- Run deletion algorithm
- Accept the new database.

EXAMPLE

a) In our example, add Loc(PODS2019, Rio) in the set to be deleted
(right top part of the screen)

b) Run the deletion algorithm, the result appears in the bottom of
the screen

Attends('Ann', 'VLDB18')
Attends('Bob', 'PODS2019')
Loc('VLDB18', '_X2New2_2#1')
Loc('PODS2019', '_X2New11_1#1')
Loc('VLDB2018', 'Rio')
Conf('PODS2019')
Conf('VLDB18')
Researcher('Ann')
Researcher('Bob')
VisaReg('_X2New2_2#1', '_X3New4_2*#2')
VisaReg('Rio', '_X3New6_3#1')
Registered('Bob', 'PODS2019')
Registered('Ann', 'VLDB18')

c) If we accept, this result corresponds to the new database instance

---
Section 3) Files with examples

The files contain examples from our paper.
In the interface, chick on FILE (top left option) to choose the initial database or the set of constraints or the set of atoms to insert of delete.

For each example we may have the following files:

BddInit.dlp --> initial set of facts (a database not necessarily consistent)
CompleteBdd.dlp --> complete initial set of facts after chase and core (the consistent database built from a given non consistent one, by completion according to constraints)
Constraints.dlp --> set of constraints
InsertFacts.dlp --> set of facts we want to insert in the database
SuppFacts.dlp --> set of facts we want to delete from the database

---
Section 4) Files with reproducible tests

Here we show how to perform the tests presented in Section 8 of the paper.

All tests are performed on the same database instance, that is, after each test we should upload the initial database CompleteBdd.dlp

STEPS:
1) Charge the database instance CompleteBdd.dlp
2) Choose the set of constraints corresponding to the test (test1, test2, ... test7)
3) Click on Insertion;
4) Choose the set of atoms to be inserted (file: InsertFacts)
5) Click on "Run insert algorithm"
6) Accept and apply core
7) Click on Deletion;
8) Choose the set of atoms to be deleted (according to the test: in directories test1,..., test7).The name of the file is SuppFacts.dlp
9) Click on "Run delete algorithm"
10) Accept