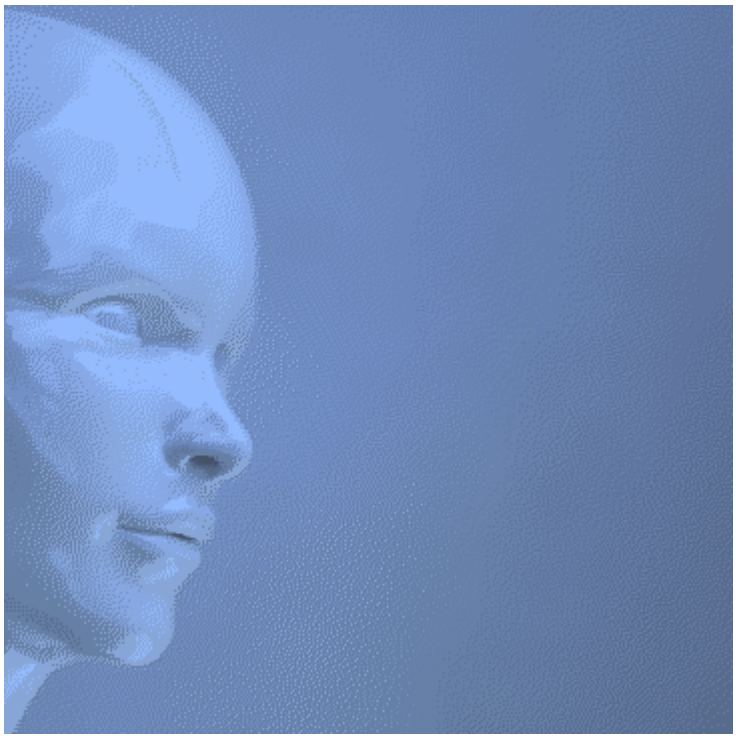
# Scientific challenge:
# Beat my natural language reasoner



# Problem description

We only know very little of the logic of language. Actually, for centuries, algebra is limited to support reasoning using verb “is/are” in the present tense form, like in:

> Given: “John **is** a father.”
> Given: “Every father **is** a man.”
•
• Logical conclusion:
< “John **is** a man.”.

But humans are also capable of possessive reasoning – using possessive verb “has” – like in:

> Given: “Paul is a son of John.”
•
• Logical conclusion:
< “John **has** a son, called Paul.”

And humans are able to reason in the past tense, like in:

> Given: “James **was** the father of Peter.”
•
• Logical conclusions:
< “Peter has no father anymore.”
< “Peter **had** a father, called James.”

So, why doesn't algebra support past tense reasoning – nor possessive reasoning – in a natural way? Why should any predicate beyond present tense verb “is/are” be described in an artificial way? Why is algebra still not equipped for linguistics, after those centuries of scientific research?

And even though algebra describes the Exclusive OR (XOR) function in a natural way, automated reasoners still don't implement its linguistic equivalent: conjunction “or”. So, automated reasoners are unable to generate the following question:

> Given: “Every person is a man **or** a woman.”
> Given: “Addison is a person.”
•
• Logical question:
< “Is Addison a man **or** a woman?”

So, even 60 years after the start of this field, knowledge technology still has a fundamental problem:

Words like definite article “the” (see Block 5), conjunction “or” (see Block 4), possessive verb “has/have” (see Block 1 and Block 2) and past tense verbs “was/were” and “had” (see Block 3) have a naturally intelligent function in language. However, their naturally intelligent function is not described in any scientific paper. Apparently, scientists don't understand their naturally intelligent function in language.

# Generally accepted workaround

The generally accepted workaround in the field of Artificial Intelligence (AI) and knowledge technology, to enter knowledge containing verb “have”, is to program it directly into a reasoner, like: has_son(john,paul). However, this is **not** a generic solution (=science), but a specific solution to a specific problem (=engineering), because you need to program each and every noun directly into the reasoner (has_daughter, has_father, has_mother, and so on), and for each and every new language. As a consequence, there is no technique available to convert a sentence like “Paul is a son of John” to “John **has** a son, called Paul” in a generic way (=through an algorithm), by which noun “son” and proper nouns “John” and “Paul” don't have to be programmed into the reasoner. It is just the first example of this challenge (see Block 1).

Below, a contribution I received from a student, in an attempt to solve this problem. With his permission, his Excel implementation for the English language:

> = IF(ISERROR(SEARCH("has a";A1));MID(A1;SEARCH("of";A1)+3;999) & " has a" & IF(ISERROR(SEARCH("is an";A1));" ";"n ") & MID(SUBSTITUTE(A1;"is an";"is a");SEARCH("is a"; SUBSTITUTE(A1;"is an";"is a"))+5;SEARCH("of"; SUBSTITUTE(A1;"is an";"is a"))-SEARCH("is";SUBSTITUTE(A1;"is an";"is a"))-6) & " called " & LEFT(A1;SEARCH("is";SUBSTITUTE(A1;"is an";"is a"))-1);MID(SUBSTITUTE(A1;"has an";"has a");SEARCH("called";SUBSTITUTE(A1;"has an";"has a"))+7;999) & " is a" & IF(ISERROR(SEARCH("has an";A1));" ";"n ") & MID(SUBSTITUTE(A1;"has an";"has a");SEARCH("has a"; SUBSTITUTE(A1;"has an";"has a"))+6;SEARCH("called"; SUBSTITUTE(A1;"has an";"has a"))-SEARCH("has";SUBSTITUTE(A1;"has an";"has a"))-7) & " of " & LEFT(A1;SEARCH("has";SUBSTITUTE(A1;"has an";"has a"))-1))

This solution doesn't check for word types, as explained in paragraph 2.3.4. The function of word types in reasoning of my fundamental document. Besides that, this logic needs to be copied for each language, while a generic solution has only one logical implementation. Moreover, this implementation can't be expanded to process for example multiple specifications words, like in: “Paul is a son of John and Anna” or “John has 2 sons, called Paul and Joe”. So, this implementation is not flexible, and therefore not generic, and not scientific.

The field of AI and knowledge technology has no foundation in nature. This field is therefore limited to deliver specific solutions to specific problems (=engineering), like Excel implementation mentioned above. However, this challenge is about uplifting this field of engineering towards a science, by developing generic solutions, based on a foundation in nature, like I am developing.

My fundamental approach shows that verb “has/have” is complementary to verb “is/are”. So, verb “has/have” can also be used in predicate logic. In order to utilize the natural meaning (naturally intelligent function) of non-keywords (structure words), I have defined natural intelligence first. Then I have identified a few (Natural Laws of) Intelligence in embedded in Grammar. And by implementing these laws of nature as a set of structuring algorithms is my system able to structure the knowledge of the system autonomously.

# The rules of this challenge

• There are 5 blocks to beat the most basic techniques of my system. Your implementation should deliver the results of at least one block listed below;
• Your implementation should not have any prior knowledge. Instead, the system should derive its knowledge directly from the input sentences listed below. Preferable: The nouns and proper nouns of the listed examples are unknown upfront;
• Your implementation should be implemented as generic as can be, in such a way that all examples of this challenge can be integrated into one single system. The reasoning screenshots of my natural language reasoner illustrate how multiple reasoning constructions reinforce each other. And the Screen shots of this challenge – included at the end of this document – show the execution by my software of the examples listed below;
• Your implementation should be published as open source software, so that its functionality is transparent. My software is also open source;
• Your implementation should be accepted by a scientific committee (conference or journal);
• In case your results are slightly different, you need to explain why you have chosen differently;
• It is an on-going challenge, until my system has been beaten on all blocks;
• I am the jury. (If you don't like me being the jury, feel free to have your solution scientifically accepted without my approval.)

# Your rewards

• A small gesture from me: € 200 for each scientifically accepted block;
• You will be the first one to have these results implemented in a scientifically accepted way, because I didn't use any technique known in this field.

You can contact me via the contact page of my website, or via LinkedIn.

# Block 1: Conversions

*The algebra of language* listed below, applies structure words: “is”, “has”, “called”, “of”, “every” and “part of”.

---

“{proper noun 1} is {indefinite article + singular noun} of {proper noun 2}”

equals to

“{proper noun 2} has {indefinite article + singular noun} called {proper noun 1}”

---

“Every {singular noun 1} has {indefinite article + singular noun 2}”

equals to

“{indefinite article + singular noun 2} is part of every {singular noun 1}”

---

**Implementation:**

> Given: “Paul is a son of John.”
•
• Generated conclusion:
< “John has a son, called Paul.”
>
> Given: “Anna has a daughter, called Laura.”
•
• Generated conclusion:
< “Laura is a daughter of Anna.”
>
> Given: “Every car has an engine.”
•
• Generated conclusion:
< “An engine is part of every car.”
>
> Given: “A sail is part of every sailboat.”
•
• Generated conclusion:
< “Every sailboat has a sail.”

# Block 2: Grouping of knowledge

*The algebra of language* listed below, applies structure words: “is”, “has”, “of”, “and” and “called”.

---

“{proper noun 1} is {indefinite article + singular noun} of {proper noun 2}”

equals to

“{proper noun 2} has {indefinite article + singular noun} called {proper noun 1}”

---

“{proper noun 1} has {indefinite article + singular noun 1} called {proper noun 2}”

and

“{proper noun 1} has {indefinite article + singular noun 1} called {proper noun 3}”

equals to

“{proper noun 1} has {number: 2} {plural form of singular noun 1} called {proper noun 2} and {proper noun 3}”

---

**Implementation:**

> Given: “John is a parent of Paul.”
•
• Generated conclusion:
< “Paul has a parent, called John.”
>
> Given: “Anna is a parent of Paul.”
•
• Generated conclusion:
< “Paul has 2 parent [plural of 'parent' is unknown], called John and Anna.”
>
> Given: “Paul has 2 parents, called John and Anna.”
•
• Detected that the generated conclusion is confirmed:
< “Paul has 2 parent [plural of 'parent' is unknown], called John and Anna.”
•
• Detected: You have Given plural noun “parents”, which was unknown to me.

# Block 3: Past tense reasoning

*The algebra of language* listed below, applies structure words: “the”, “was”, “has”, “had”, “of”, “called” and “no […] anymore”.

---

“{proper noun 1} was {**definite** article + singular noun} of {proper noun 2}”

from which can be concluded

“{proper noun 2} has no {singular noun} anymore”

---

“{proper noun 1} was {**definite** or indefinite article + singular noun} of {proper noun 2}”

from which can be concluded

“{proper noun 2} had {indefinite article + singular noun} called {proper noun 1}”.

---

**Implementation:**

> Given: “James was the father of Peter.”

•

• Generated conclusions:

< “Peter has no father anymore.”

< “Peter had a father, called James.”

# Block 4: Detection of a conflict and generation of a question

*The algebra of language* listed below, applies structure words: “is”, “or”, “and”, “every” and “not”.

---

“Every {singular noun 1} is {indefinite article + singular noun 2} **or** {indefinite article + singular noun 3}”

conflicts with

“{proper noun 1} is {indefinite article + singular noun 2} **and** {indefinite article + singular noun 3}

---

“Every {singular noun 1} is {indefinite article + singular noun 2} or {indefinite article + singular noun 3}”

and

“{proper noun 1} is {indefinite article + singular noun 1}”

from which can be concluded

“{proper noun 1} is {indefinite article + singular noun 2} or {indefinite article + singular noun 3}”

---

“{proper noun 1} is {indefinite article + singular noun 2} or {indefinite article + singular noun 3}”

equals to

“Is {proper noun 1} {indefinite article + singular noun 2} or {indefinite article + singular noun 3}?”

---

“Is {proper noun 1} {indefinite article + singular noun 1} or {indefinite article + singular noun 2}?”

and

“{proper noun 1} is not {indefinite article + singular noun 1}”

from which can be concluded

“{proper noun 1} is {indefinite article + singular noun 2}”

---

“Is {proper noun 1} {indefinite article + singular noun 1} or {indefinite article + singular noun 2}?”

and

“{proper noun 1} is not {indefinite article + singular noun 2}”

from which can be concluded

“{proper noun 1} is {indefinite article + singular noun 1}”

---

**Implementation:**

> Given: “Every person is a man or a woman.”
> Given: “Addison is a man and a woman.”
!
! Detected conflict: This sentence is not accepted, because it is in conflict with itself.
!
> Given: “Addison is a person.”
•
• Generated question:
< “Is Addison a man or a woman?”
>
> Given: “Addison is not a woman.”
•
• Detected that the generated question has been answered:
< “Is Addison a man or a woman?”
•
• Generated conclusion:
< “Addison is a man.”

# Block 5: Archiving of knowledge

*The algebra of language* listed below, applies structure words: “the”, “is”, “has”, “of”, “called”, “previous” and “new”.

---

“{proper noun 1} is {**definite** article + singular noun} of {proper noun 2}”

from which can be concluded

“{proper noun 2} has {indefinite article + singular noun} called {proper noun 1}”

---

“{proper noun 1} is {**definite** article + singular noun} of {proper noun 2}”

followed by

“{proper noun 3} is {**definite** article + singular noun} of {proper noun 2}”

from which can be concluded

“{proper noun 2} has {indefinite article} new {singular noun} called {proper noun 3}”

and

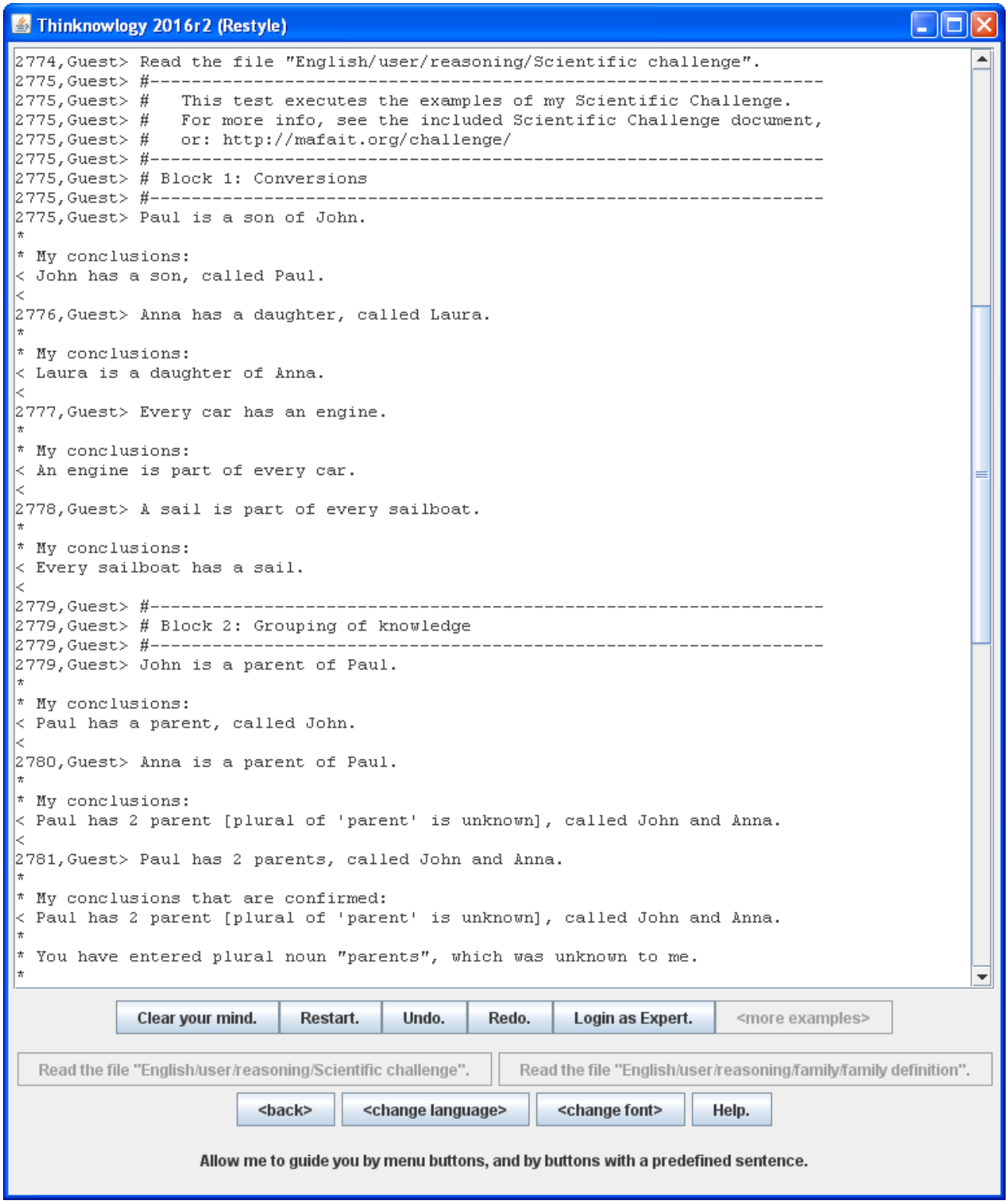
“{proper noun 2} has {indefinite article} previous {singular noun} called {proper noun 1}”

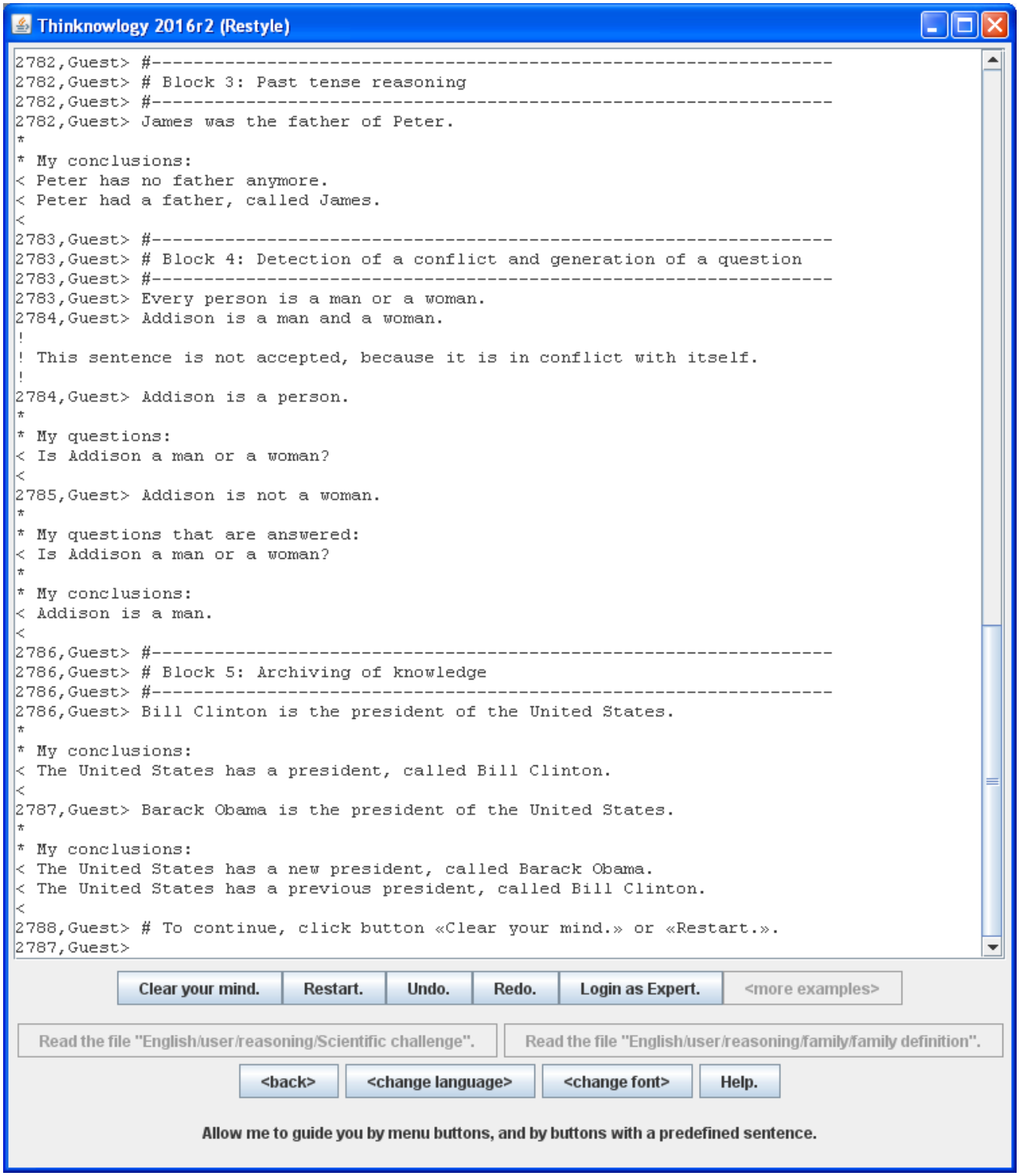
---

**Implementation:**

> Given: “Bill Clinton is the president of the United States.”
•
• Generated conclusion:
< “The United States has a president, called Bill Clinton.”
>
> Given: “Barack Obama is the president of the United States.”
•
• Generated conclusions:
< “The United States has a new president, called Barack Obama.”
< “The United States has a previous president, called Bill Clinton.”

# Screen shots of this challenge




Thinknowlogy 2016r2 (Restyle)
2782,Guest> #----------
2782,Guest> # Block 3: Past tense reasoning
2782,Guest> #----------
2782,Guest> James was the father of Peter.
*
* My conclusions:
< Peter has no father anymore.
< Peter had a father, called James.
<
2783,Guest> #----------
2783,Guest> # Block 4: Detection of a conflict and generation of a question
2783,Guest> #----------
2783,Guest> Every person is a man or a woman.
2784,Guest> Addison is a man and a woman.
!
! This sentence is not accepted, because it is in conflict with itself.
!
2784,Guest> Addison is a person.
*
* My questions:
< Is Addison a man or a woman?
<
2785,Guest> Addison is not a woman.
*
* My questions that are answered:
< Is Addison a man or a woman?
*
* My conclusions:
< Addison is a man.
<
2786,Guest> #----------
2786,Guest> # Block 5: Archiving of knowledge
2786,Guest> #----------
2786,Guest> Bill Clinton is the president of the United States.
*
* My conclusions:
< The United States has a president, called Bill Clinton.
<
2787,Guest> Barack Obama is the president of the United States.
*
* My conclusions:
< The United States has a new president, called Barack Obama.
< The United States has a previous president, called Bill Clinton.
<
2788,Guest> # To continue, click button «Clear your mind.» or «Restart.».
2787,Guest>
Clear your mind.
Restart.
Undo.
Redo.
Login as Expert.
<more examples>
Read the file "English/user/reasoning/Scientific challenge".
Read the file "English/user/reasoning/family/family definition".
<back>
<change language>
<change font>
Help.
Allow me to guide you by menu buttons, and by buttons with a predefined sentence.