

Functional Lenses

Plan

- View Update Problem
- Explain Lenses
- Example
- Composing Lenses
- How I'm using them in Ruby

View Update Problem

Source

View

```
<person>  
  <fname>Archibald</fname>  
  <lname>Buttle</lname>  
  <other>...</other>  
</person>
```

Buttle, Archibald

Updated Source

Updated View

```
<person>  
  <fname>Archibald</fname>  
  <lname>Tuttle</lname>  
  <other>...</other>  
</person>
```

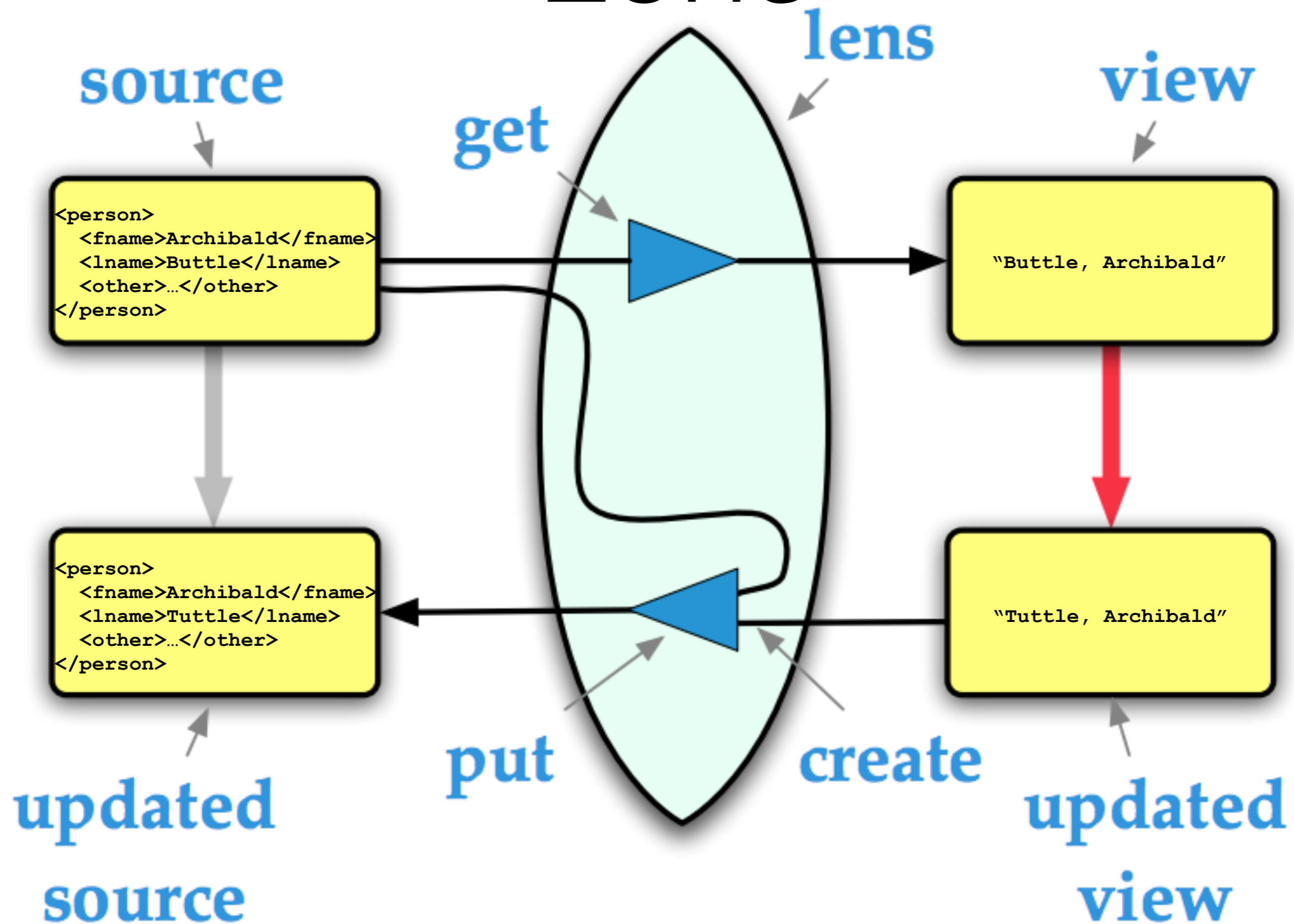
Tuttle, Archibald



What is a lens?

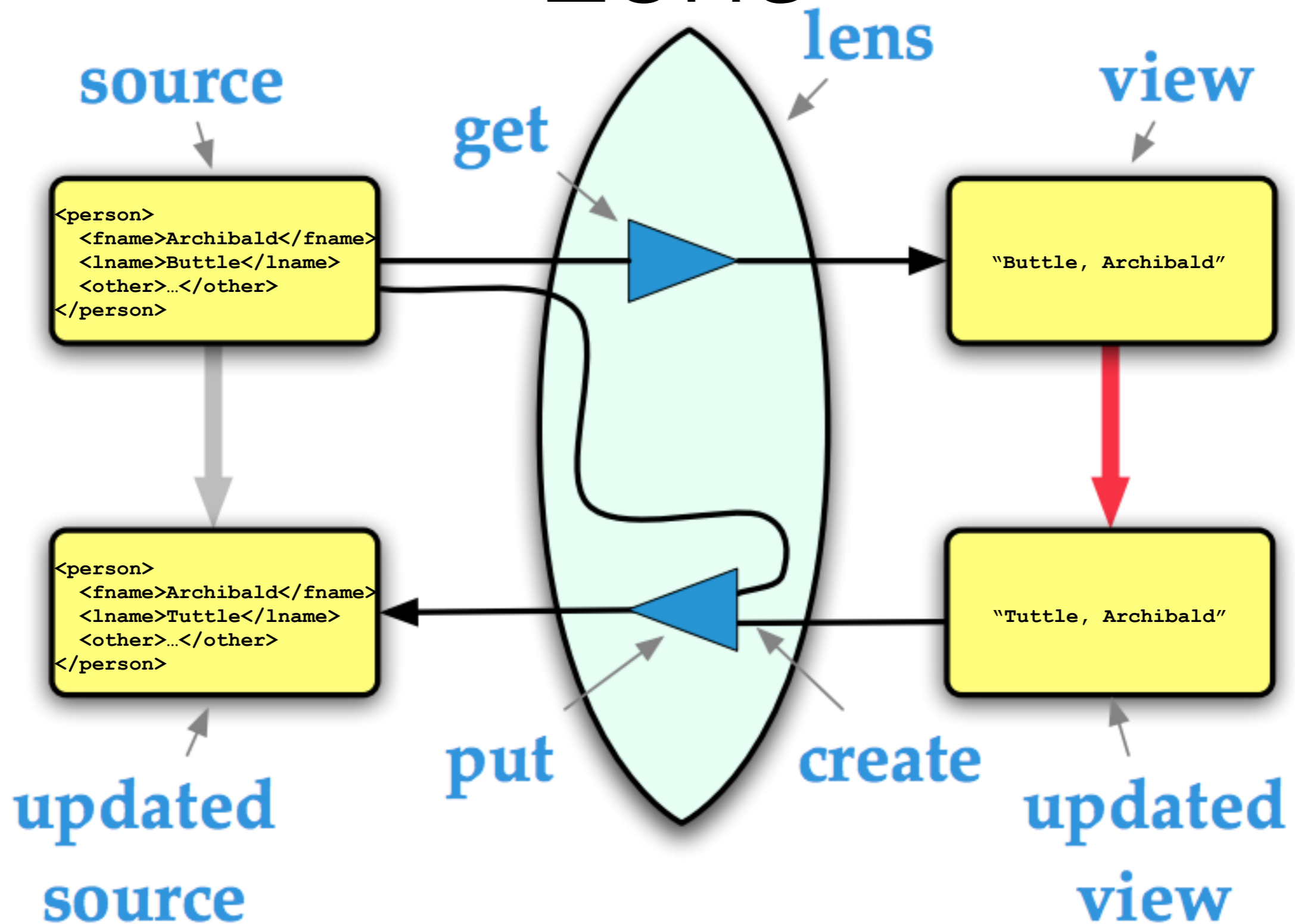
A group of functions that map between a complex value and one of its parts. These “getter” and “setter” functions also allow mutating.

Lens

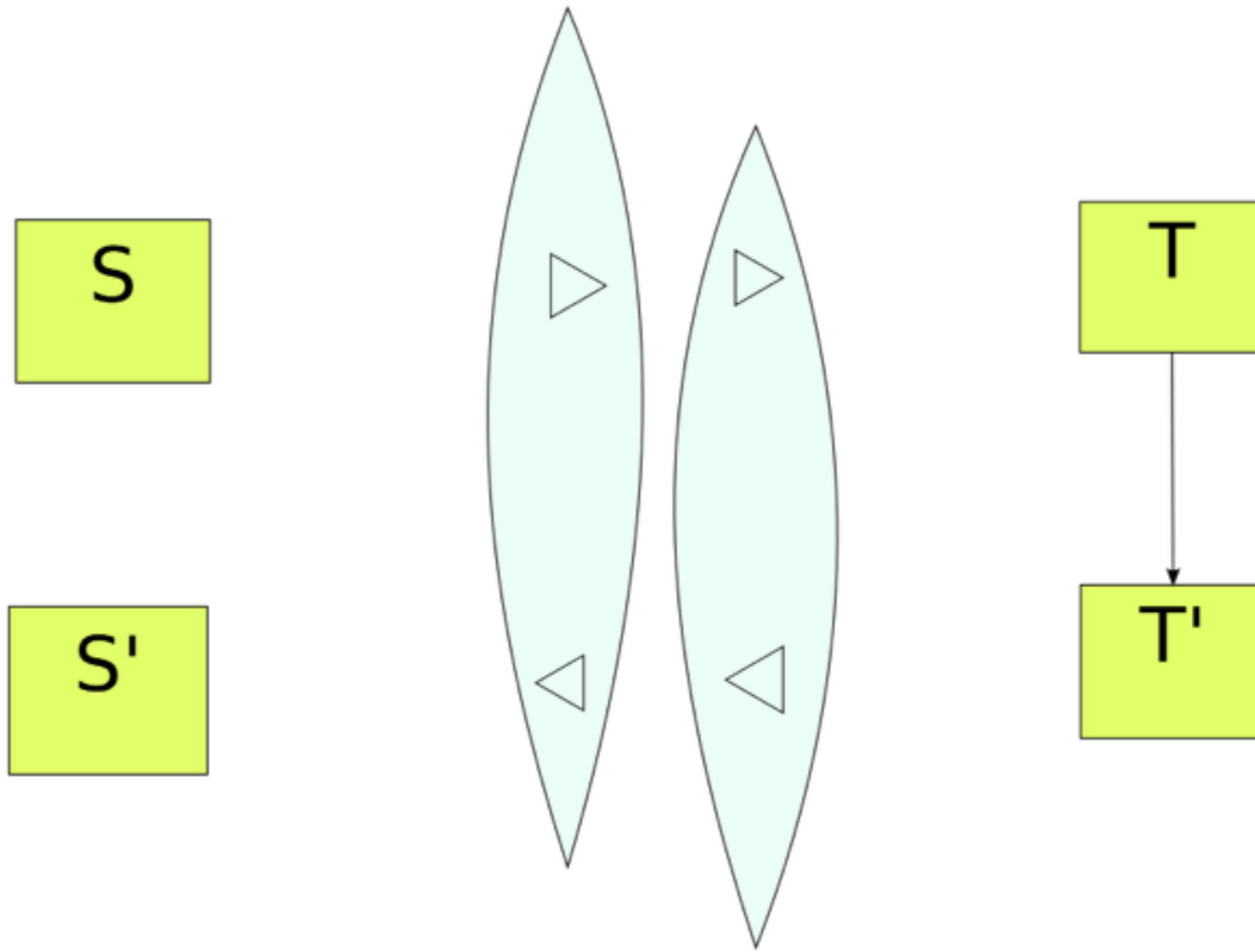


<code>

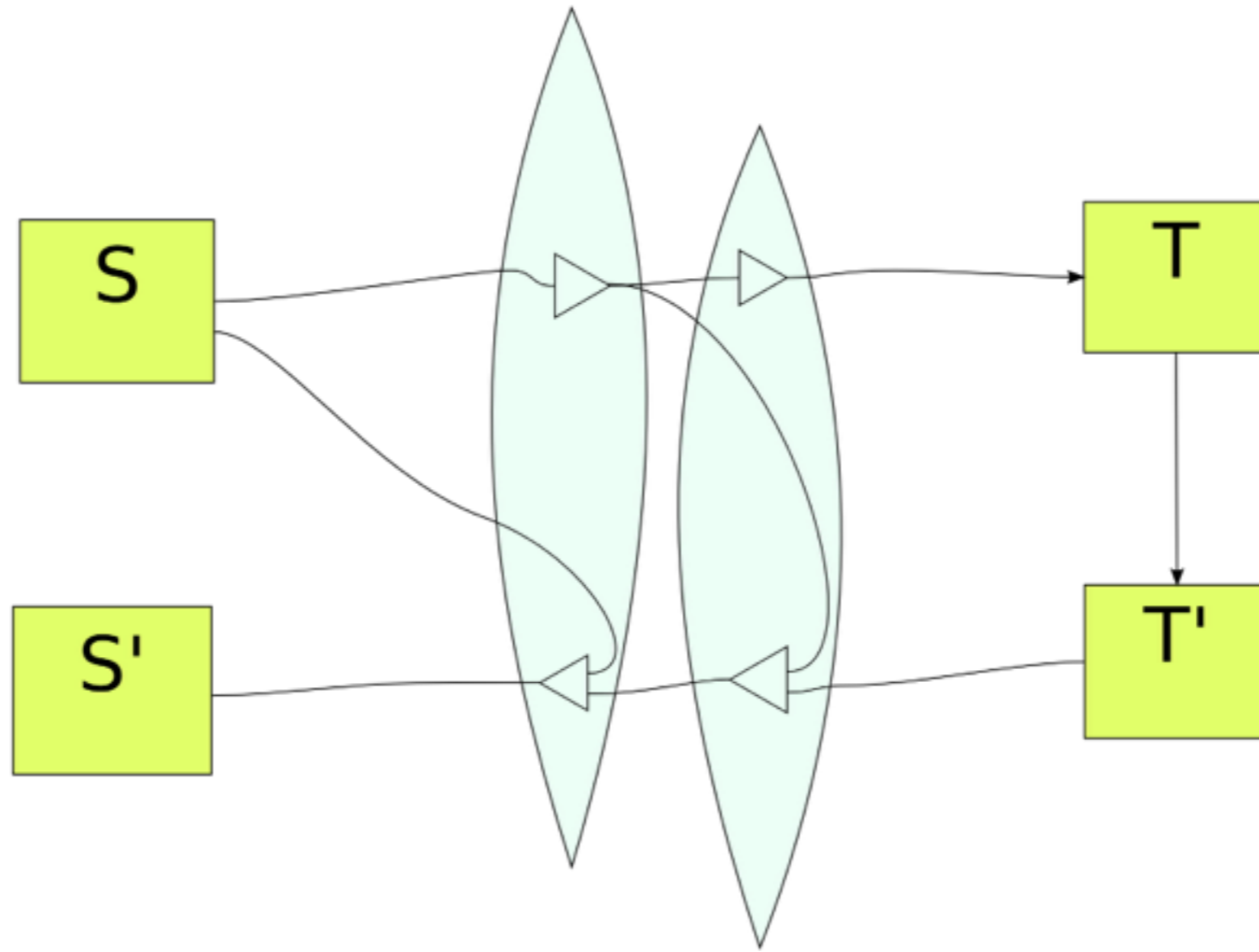
Lens



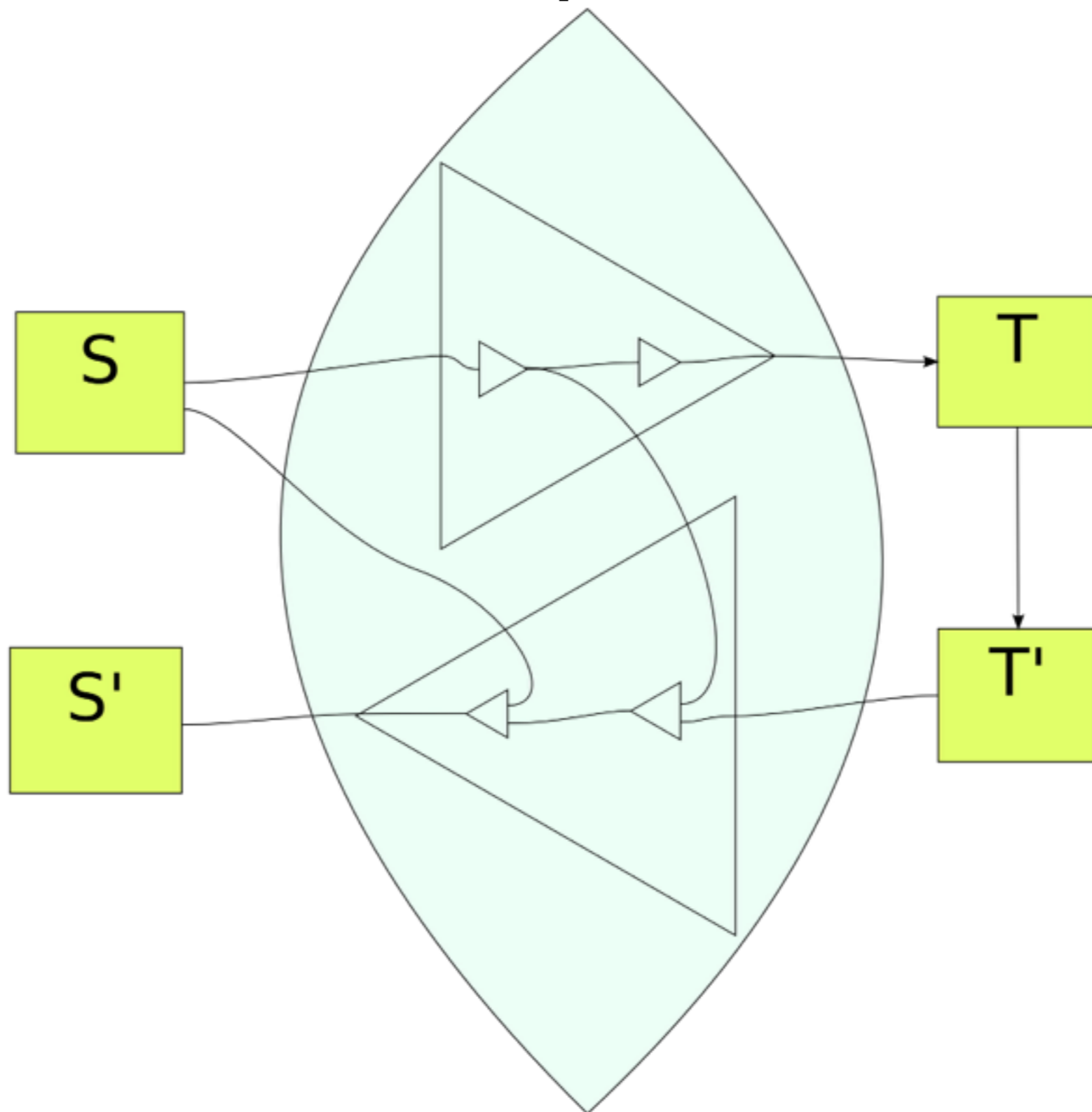
Compose



Compose



Compose



Well Behaved

GetPut: $(\text{lens-put } s (\text{lens-get } s)) = s$

PutGet: $(\text{lens-get } (\text{lens-put } s v)) = v$

CreateGet: $(\text{lens-get } (\text{lens-create } v)) = v$



- RDF graph database
- many nodes
- each node has many attributes
- each attribute has one or more values
- some values are links to other nodes
- values could be in other formats (string of xml)
- xml could contain links to other documents

Simple case

- get the node at a given url
- get the :title attribute
- use that value for the title attribute of our object
- when the object is saved, update the :title attribute on the node

Complex case

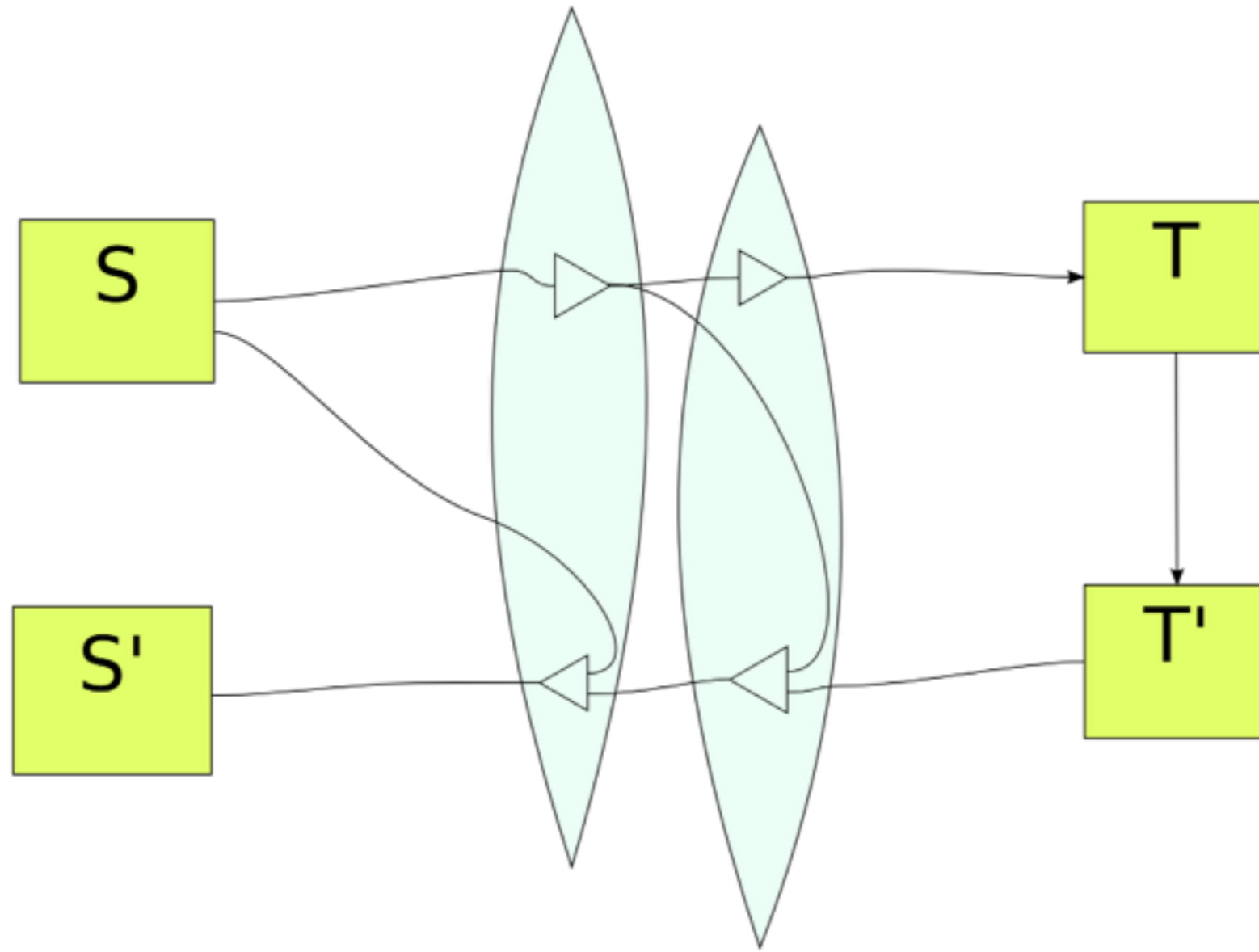
- get the node at a given url
- get the first value in the :relationships attribute
- parse the xml in that attribute
- get the url at “relationships relationship[type=secondary]”
- load the node at that url
- get the first :paragraph attribute on that node
- use that value as the first_paragraph attribute on our object
- when we save our object, save the appropriate model’s attribute

<code>

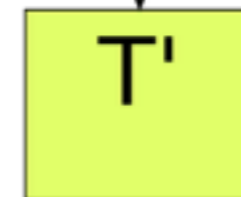
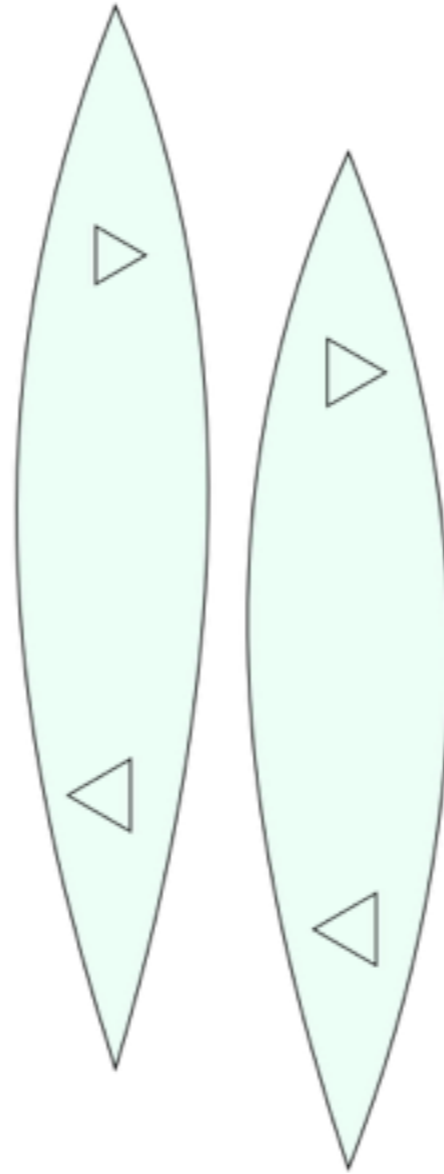
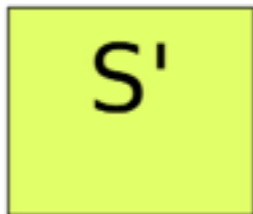
https://github.com/curationexperts/fedora_lens/blob/13b3cf5ddd374e81cebc68568edab2b090b7d24e/lib/fedora_lens/lenses.rb#L16

https://github.com/curationexperts/fedora_lens/blob/13b3cf5ddd374e81cebc68568edab2b090b7d24e/lib/test_class.rb#L37

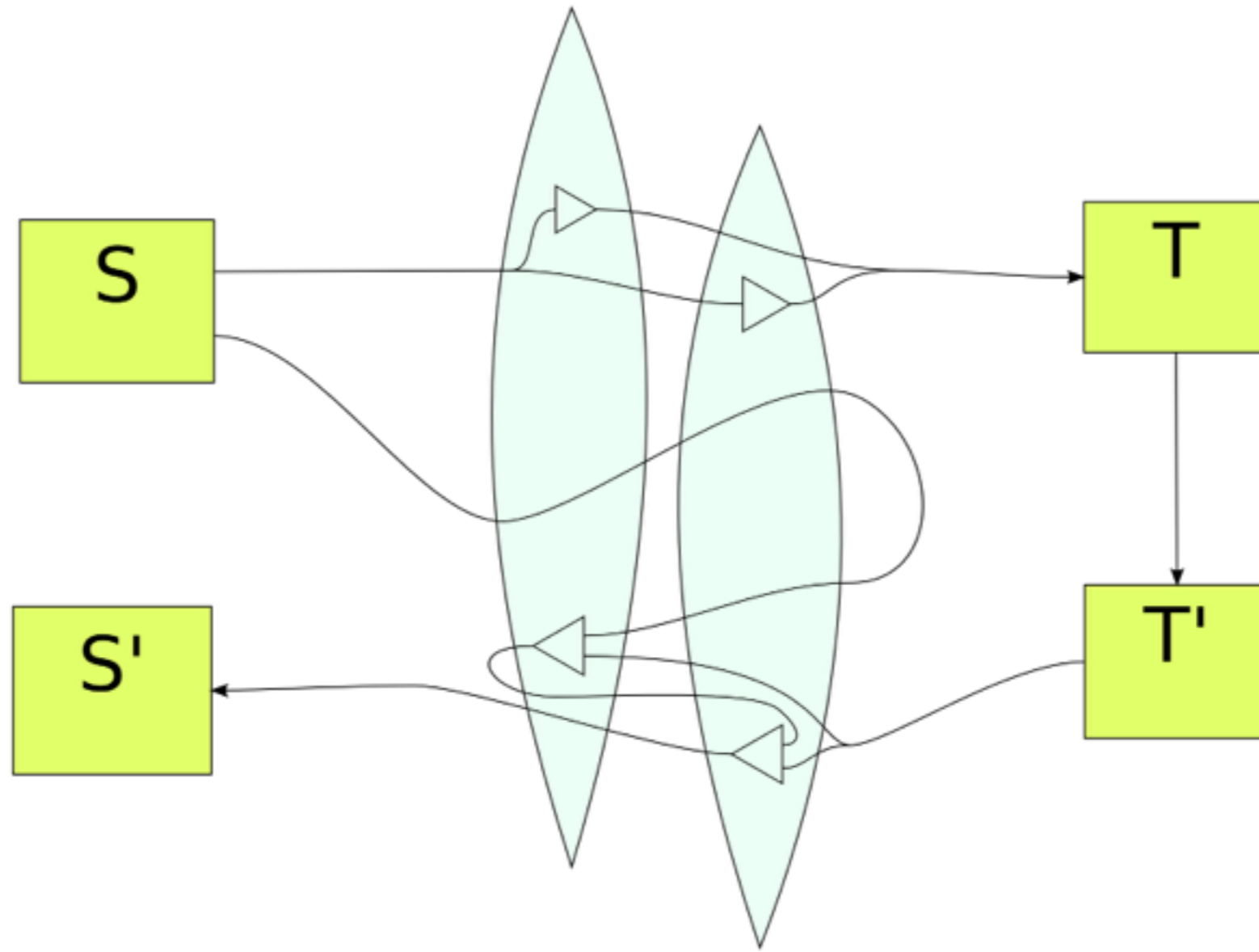
A problem...



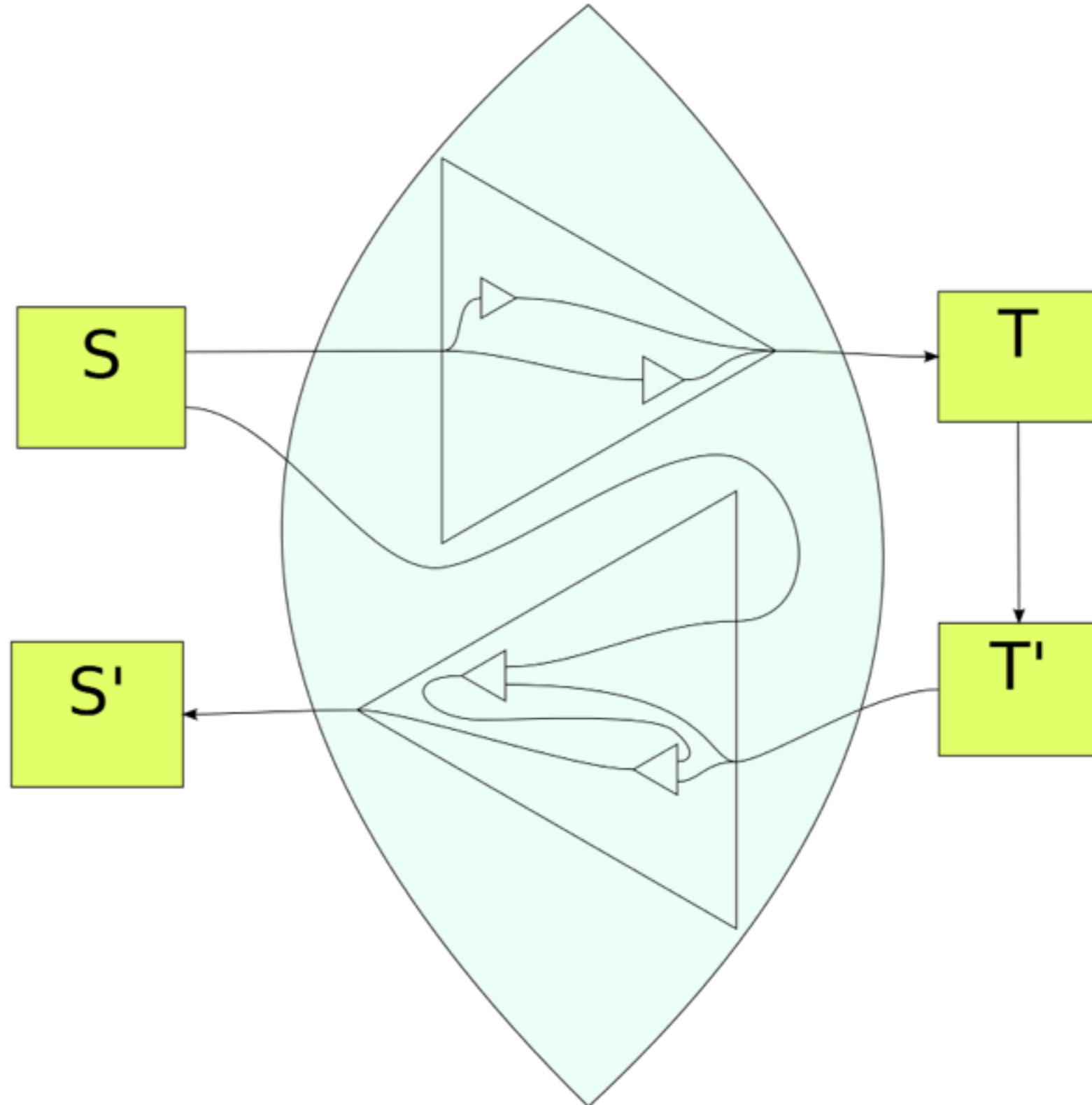
Merge?



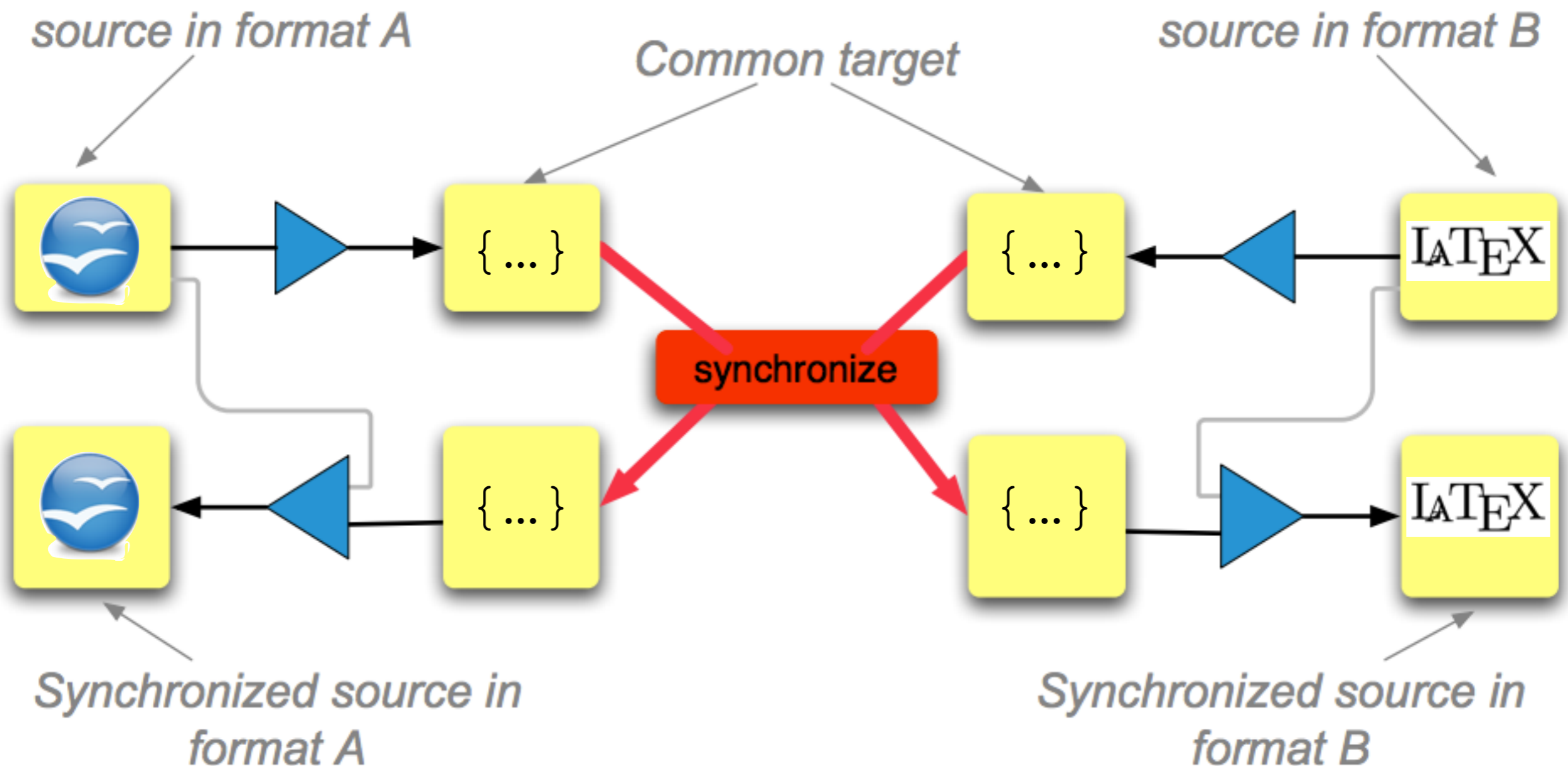
Merge?



Merge?



Synchronizing



Thanks!

@bmaddy

References:

Harmony language: <http://www.seas.upenn.edu/~harmony/manual.pdf>

Harmony slides: <http://www.cis.upenn.edu/~bcpierce/papers/lenses-etapsslides.pdf>

Clojure library: <https://github.com/cark/data.lenses>

cgrand on structured editing and sjacket: <http://vimeo.com/45695419>

Example by cgrand: <https://gist.github.com/cgrand/5683844>

View Update Problem: <http://www.ijcte.org/papers/278-D038.pdf>