Thursday 25 January: convenience steps [updated] and what's Next?

Convenience steps

```
convSteps(X0,XN):-
       denest(X0,X1),        no changes
       justRoots(X1,X2),     changed to keep cardinalities from being removed
       timeSequence(X2,X3),  reformat; remove present tenses and appearance of the *
       nfTree(X3,XN).        no changes

| ?- parseOne('John has been eating the three ripe peaches.', P),
convSteps(P,P1),pretty(P+P1).
```

Base Tree

```
[.,
  arg(claim,
      *(time(tense(present), aspect(simple), aux(+), def(+), finite(tensed))),
      [has,
       arg(A,
           *(time(tense(past),
                  aspect(perfect),
                  aux(+),
                  def(B),
                  finite(participle))),
           [[be]>en,
            arg(C,
                *(time(tense(present),
                       aspect(prog),
                       aux(-),
                       def(D),
                       finite(participle))),
                [[eat>ing,
                  arg(dobj,
                      *(the),
                      [[[peach>plural, modifier(amod, ripe>)],
                        modifier(numAsMod, three)],
                       modifier(identity, the)])],
                 arg(subject, *(name), [John:NP])])])])]
```

After convSteps

```
  + [.,
     arg(claim,
         *([time(tense(past),
                 aspect(perfect),
                 aux(+),
                 def(B),
                 finite(participle))]),
         [eat,
          arg(dobj,
              *(the),
              [peach>plural, modifier(amod, ripe), modifier(numAsMod, three)]),
          arg(subject, *(name), [John:NP])])]
```

What's Next?

(qlf): quasi logical form. we've discussed the questions below, now let's talk about handling 'generics'.

Q: what should it do?

Q: the jobs of ***opaque*** and ***qq***?

Q: cardinality? scoping?

Q: What kind of quantifier does a time sequence introduce?

Generics

```
| ?- parseOne('all cats are fools.',X),convSteps(X,XP),pretty(X+XP).
```

Base tree:

```
[.,
  arg(claim,
      *(time(tense(present), aspect(simple), aux(-), def(A), finite(tensed))),
      [[[be],
        arg(predication(xbar(v(-), n(+))), *(generic), fool>plural)],
       arg(subject, *(universal), [cat>plural, modifier(identity, all)])])]
```

After convSteps:

```
[.,
     arg(claim,
         *([time(tense(present),
                 aspect(simple),
                 aux(-),
                 def(A),
                 finite(tensed))]),
         [be,
          arg(predication(xbar(v(-), n(+))), *(generic), fool>plural),
          arg(subject, *(universal), [cat>plural])])]
```

```
| ?- parseOne('John have been eating peaches.',X),convSteps(X,XP),pretty(X+XP).
```

Base tree:

```
[.,
  arg(claim,
      *(time(tense(present), aspect(simple), aux(+), def(+), finite(tensed))),
      [has,
       arg(A,
           *(time(tense(past),
                  aspect(perfect),
                  aux(+),
                  def(B),
                  finite(participle))),
           [[be]>en,
            arg(C,
                *(time(tense(present),
                       aspect(prog),
                       aux(-),
                       def(D),
                       finite(participle))),
                [[eat>ing, arg(dobj, *(generic), peach>plural)],
                 arg(subject, *(name), [John:NP])])])])]
```

After convSteps:

```
[.,
     arg(claim,
         *([time(tense(past),
                 aspect(perfect),
                 aux(+),
                 def(B),
                 finite(participle))]),
         [eat,
          arg(dobj, *(generic), peach>plural),
          arg(subject, *(name), [John:NP])])]
```

Q: Do generics introduce quantifiers? what kind of quantifiers?
Q: In qlf what scope are they usually given?

2.2 Definite Plurals

```
Definite plurals can often be non-anaphoric and behave like universally
quantified noun phrases (90). However, as with (generic) bare plurals, the
force of the quantification can also be less than universal (91). Whether this
lessening of quantificational force is due to the noun phrase or to the
predicate of which the NP is an argument is unclear (92, 93).
```

fracas-090 answer: yes

P1	The chairman read out the items on the agenda.
Q	Did the chairman read out every item on the agenda?
H	The chairman read out every item on the agenda.

```
Non-anaphoric, universal plural definite
```

fracas-091 answer: unknown

P1	The people who were at the meeting voted for a new chairman.
Q	Did everyone at the meeting vote for a new chairman?
H	Everyone at the meeting voted for a new chairman.
Why	*Some people may have abstained from the vote*

fracas-092 answer: yes

P1	All the people who were at the meeting voted for a new chairman.
Q	Did everyone at the meeting vote for a new chairman?
H	Everyone at the meeting voted for a new chairman.

fracas-093 answer: yes

P1	The people who were at the meeting all voted for a new chairman.
Q	Did everyone at the meeting vote for a new chairman?
H	Everyone at the meeting voted for a new chairman.