MAPPING RULES: REPLACE & SUBSTITUTE (i)

- REPLACE removes *all the tags* in the reading that matches the target and adds the inserted tag(s) if all the contextual tests are satisfied.
- SUBSTITUTE it removes the tags from the search tags and adds the new tags in its place in a reading that matches the target if all the contextual tests are satisfied.

E.g.

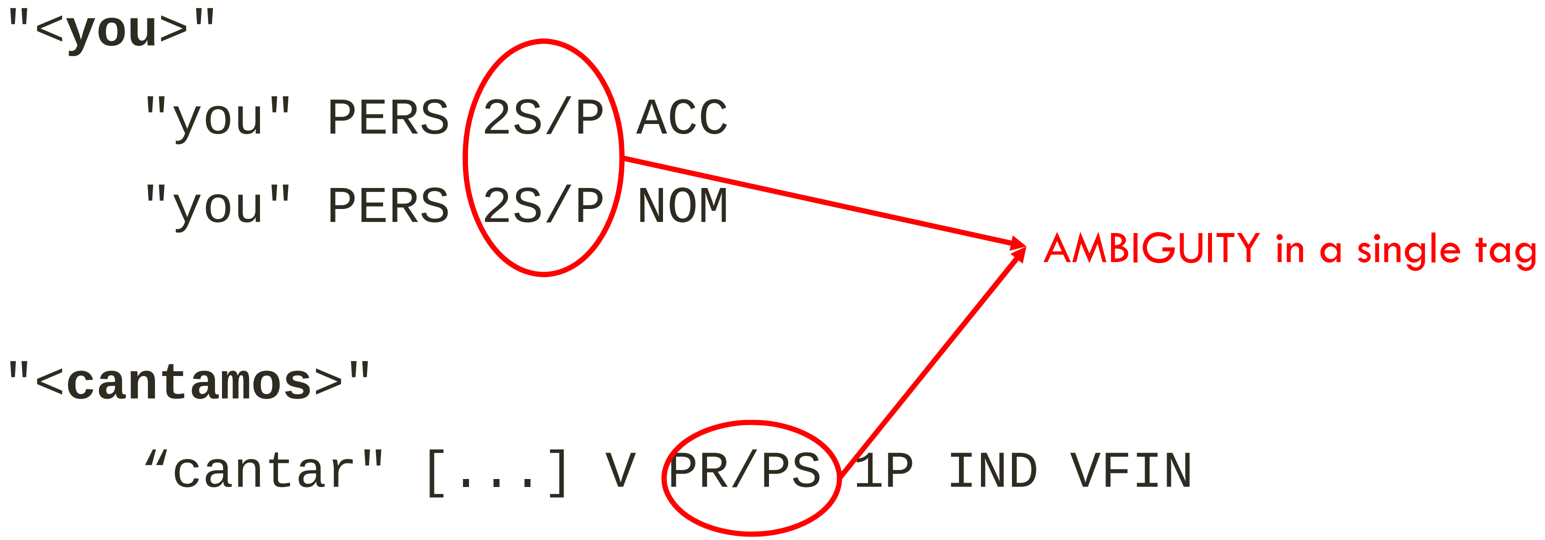
Operation	inserted tag	optional	targetset	optional	test to be satisfied	
REPLACE	(PERS 2P ACC)	TARGET	AMBIGUOUS_YOU	IF	(1 PLURAL_NOUN)	;

This rule replaces the whole reading by PERS 2S ACC if the reading belongs to an ambiguous *you* and it has a plural noun next to the right e.g. *you guys*. Be careful! It replaces the whole reading; the rest of the tags are erased.

Operation	(search tag)	(new tag)	optional	targetset	optional	(test to be satisfied)	
SUBSTITUTE	(S/P)	(2P)	TARGET	AMBIGUOUS_YOU	IF	(1 PLURAL_NOUN)	;

This rule replaces the tags 2S/P by 2S if the reading belongs to an ambiguous *you* and it has a plural noun next to the right e.g. *you guys*.

SUBSTITUTE for disambiguation (i)



```
"<you>"
    "you" PERS 2S/P ACC
    "you" PERS 2S/P NOM

"<cantamos>"
    “cantar" [...] V PR/PS 1P IND VFIN
```

AMBIGUITY in a single tag

SUBSTITUTE for disambiguation (ii)

You guys must hurry up.

```
"<you>"
        "you" <*> PERS 2S/P ACC
        "you" <*> PERS 2S/P NOM
"<guys>"
        "guy" V PR 3S
        "guy" <H> <cc-cord> N P NOM
"<must>"
        "must" V PR
        "must" <amount> N S NOM
"<hurry>"
        "hurry" V PR -3S
        "hurry" V INF
        "hurry" V IMP
        "hurry" <act> <f-psych> <sit> N S NOM
"<up>"
        "up" <adir> ADV
        "up" PRP
"<$.>"
        "." PU <<< @PU
```

SUBSTITUTE for disambiguation (ii): rule

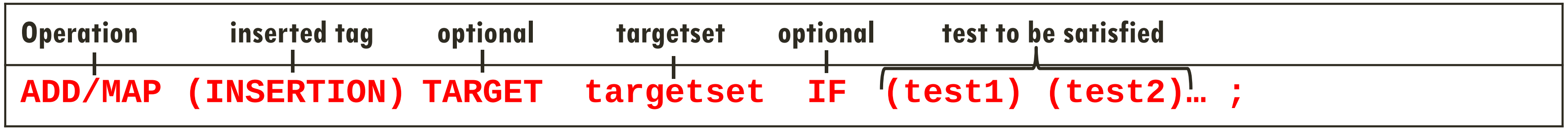
```
DELIMITERS = "<$.>" "<$!>" "<$?>" "<$\;>" "<$:>" "<$-->" "<$>" ;
MAPPING-PREFIX = @ ;

SETS
LIST PLURAL_NOUN = (N P NOM) ;
LIST AMBIGUOUS_YOU = ("you" PERS 2S/P) ;

MAPPINGS
SUBSTITUTE(2S/P)  (2P)  TARGET AMBIGUOUS_YOU  IF  (1 PLURAL_NOUN);
          |         |      |          |          |        |
      search tag  new tag optional  targetset  optional  condition
```

SUBSTITUTE for disambiguation: outcome

```
"<you>"
    "you" <*> PERS 2P ACC SUBSTITUTE:12
    "you" <*> PERS 2P NOM SUBSTITUTE:12
"<guys>"
    "guy" V PR 3S
    "guy" <H> <cc-cord> N P NOM
[...]
```

MAPPING: MAP & ADD

- MAP operation adds a tag (or sequence of tags) to all readings containing the target tag if all the tests are satisfied. A reading is not mapped twice.
- ADD operation is similar to MAP, but it continues mapping. Therefore, a reading can mapped with more than one tag.



```
ADD/MAP (INSERTION) TARGET  targetset  IF (test1) (test2)… ;
```

```
ADD (@AGREEMENT-MISTAKE)
    TARGET AUX
    IF (0 (PR -3S)) (-1 (PERS 3S NOM)) ;
```

This rule adds the tag @AGREEMENT-MISTAKE to some auxiliary verbs (i.e. to readings having a tag in set AUX) if they are in non-third-person-singular form (-3S) and the previous word is a singular personal pronoun.

He do not like paella.

```
"<he>"
       "he" <*> <masc> PERS MASC 3S NOM
"<do>"
       "do" <v.contact> V PR -3S
       "do" <v.contact> V INF
       "do" <v.contact> V IMP
"<not>"
       "not" ADV
"<like>"
       "like" ADJ POS
       "like" ADV
       "like" KS
       "like" <percep-f> N S NOM
       "like" <vtk+ADJ> V PR -3S
       "like" <vtk+ADJ> V INF
       "like" <vtk+ADJ> V IMP
       "like" PRP
"<paella>"
       "paella" <food-h> N S NOM
"<$.>“
       "." PU <<< @PU
```

MAPPING: rule

```
DELIMITERS = "<$.>" "<$!>" "<$?>" "<$\;>" "<$:>" "<$-->" "<$>" ;
MAPPING-PREFIX = @ ;

SETS
LIST HE/SHE/IT = (PERS 3S NOM) ; # "<he>" "<she>" or "<it>" wordforms
LIST AUX = "be" "have" "do" ;
LIST NOT3S = -3S 1S -1/3S; # to be => 1S (I am), -1/3S (we/you/they are)
SET AUX-NOT3S = AUX + NOT3S ; # concatenation of sets AUX and NOT3S

MAPPINGS
ADD (@AGREEMENT-MISTAKE)
     TARGET AUX-NOT3S
     IF (-1 HE/SHE/IT)
     ;
```

MAPPING: output

He do not like paella.

```
"<he>"
        "he" <*> <masc> PERS MASC 3S NOM
"<do>"
        "do" <v.contact> V PR -3S @AGREEMENT-MISTAKE ADD:11
        "do" <v.contact> V INF
        "do" <v.contact> V IMP
"<not>"
        "not" ADV
"<like>"
        "like" ADJ POS
        "like" ADV
        "like" KS
        "like" <percep-f> N S NOM
        "like" <vtk+ADJ> V PR -3S
        "like" <vtk+ADJ> V INF
        "like" <vtk+ADJ> V IMP
        "like" PRP
"<paella>"
        "paella" <food-h> N S NOM
"<$.>“
        "." PU <<< @PU
```

MAPPING

Check the same rule with the sentences:

He do not like paella. It have made a mistake. She are swimming. He were doing well.

And then with their correct versions:

He does not like paella. It has made a mistake. She is swimming. He was doing well.

Practical exercise 2