Case 1

P31
a: 1
b, c, d

P21
a: 1
b, c

P11
a:1
b

P12
a:2
c

P13
a:3
d

P31
a: 1?
b, d

P11
a:1
b

P12
a:2
c

P13
a:3
d

Case 2

P31
a: 2
b, c, d

P21
a: 2
b, c

P11
a:1
b

P12
a:2
c

P13
a:3
d

P31
a: 1/3?
b, d

P11
a:1
b

P12
a:2
c

P13
a:3
d

Case 3

P31
a: 3
b, c, d

P21
a: 1
b, c

P11
a:1
b

P12
a:2
c

P13
a:3
d

P31
a: 3?
b, d

P11
a:1
b

P12
a:2
c

P13
a:3
d

Case 4

P31
a: 3
b, c, d

P21
a: 2
b, c

P11
a:1
b

P12
a:2
c

P13
a:3
d

P31
a: 1/3?
b, d

P11
a:1
b
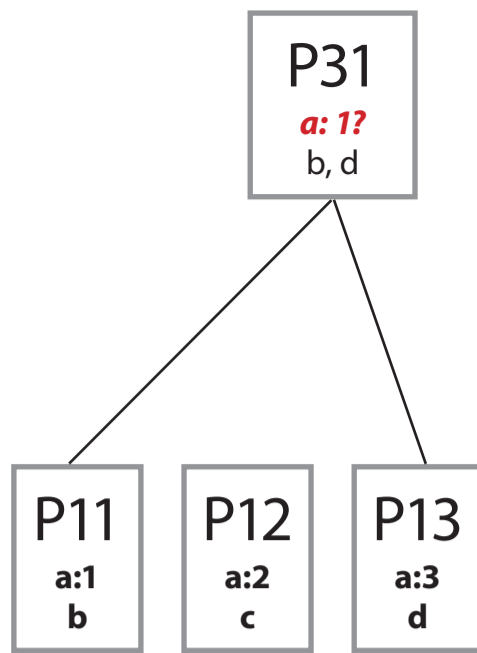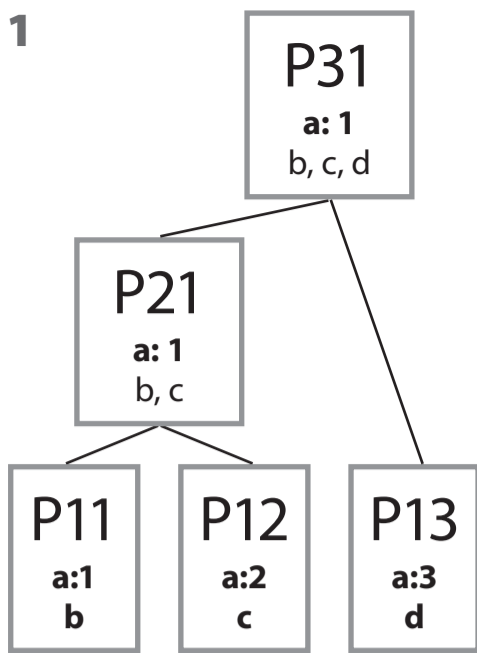
P12
a:2
c

P13
a:3
d

before

after

Undoing a Conclusion Permutations

This goes through every possibility in a simple example of removing a conclusion Record.

Each Person shows the current PFACT-values inside. Note that only the bold PFACTs are physically recorded in the Person (via pointer or copying). The other values are inherited from below.
The only clash in this example occurs with PFACT "a".

P21 is combining P11 and P12.
On combining these two, I resolved the clash about "a", one way or the other.

Now I want to undo the combination of P11 and 12. I've decided
- that these aren't the same human being,
- and that P11 belongs in this tree, and P12 doesn't.
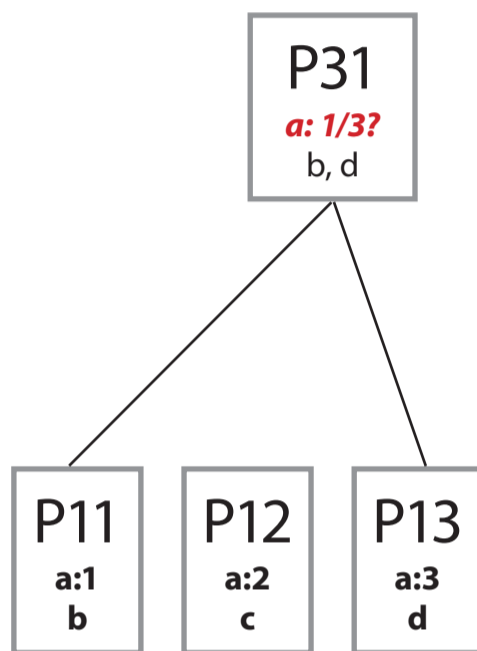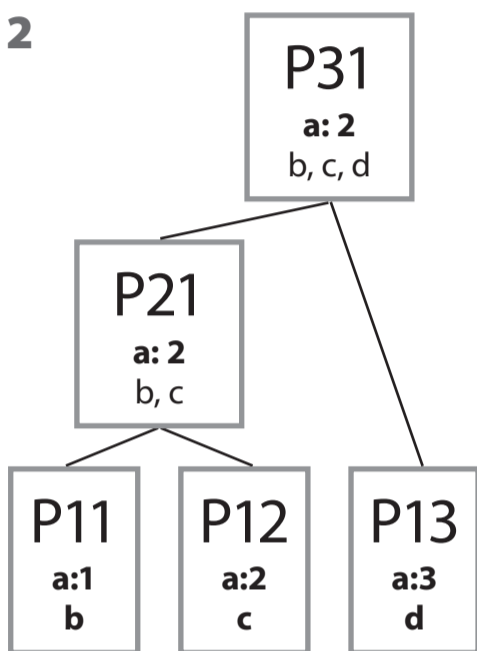*So I want to remove P21 and attach P11 to P31.*

After removing P21,
- the PFACT "c", exclusive to P12, does not appear in higher levels any more. Good.
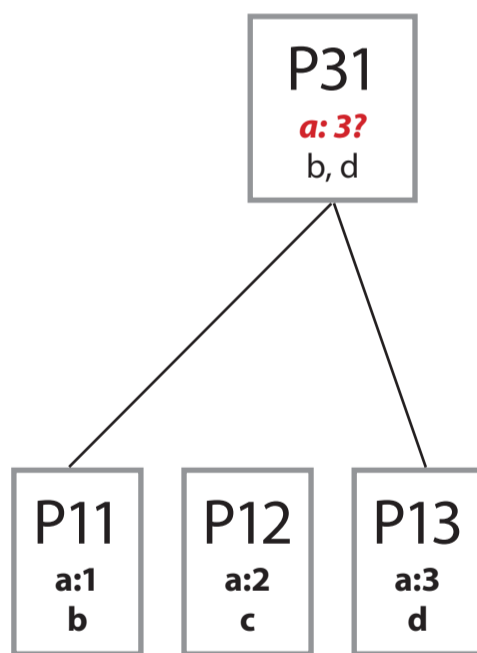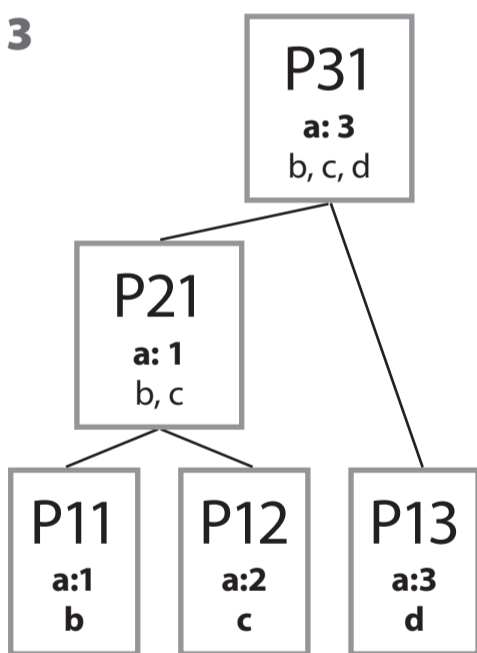*- the clashing PFACT "a" needs new attention.*

Depending on what was the preferred value, there might be a good automatic guess at the new preferred value. But I would not want my software to do anything more than offer a choice with the most likely value already selected.

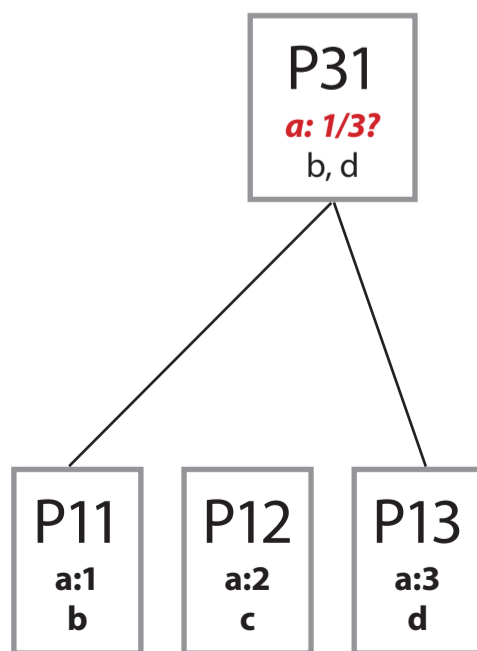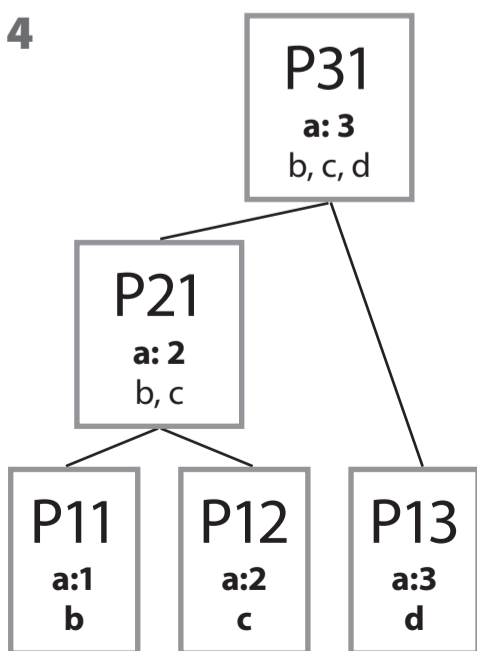*PS: In reality, P21 would not have to be deleted, it could just be unlinked from P31.*