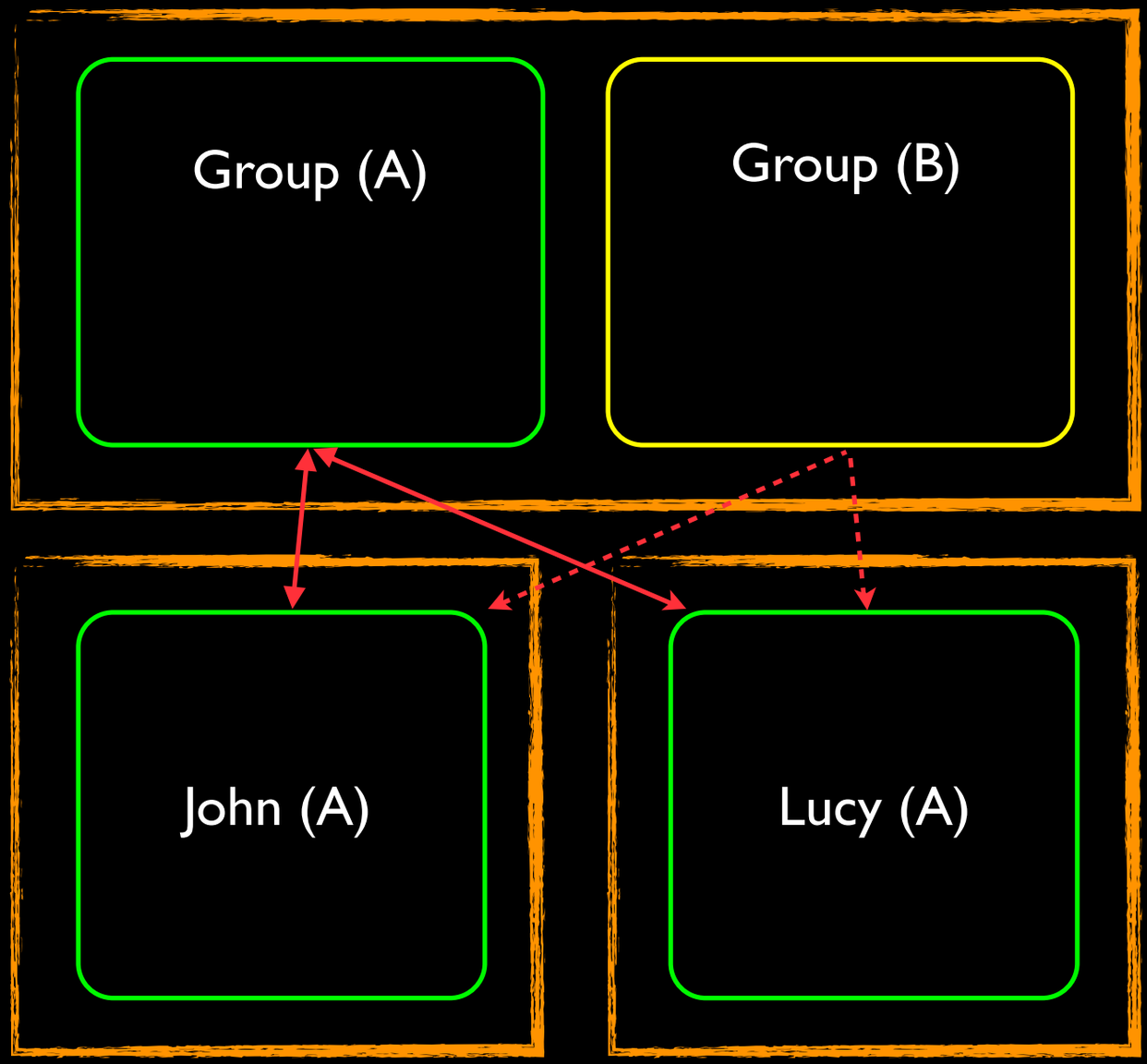
Person and Person Group Branch

- Persistent Root
- Current Branch
- Additional Branch
- Public Reference
- Private Reference
- Public Composite Ref
- Private Composite Ref



Person Group

Persons

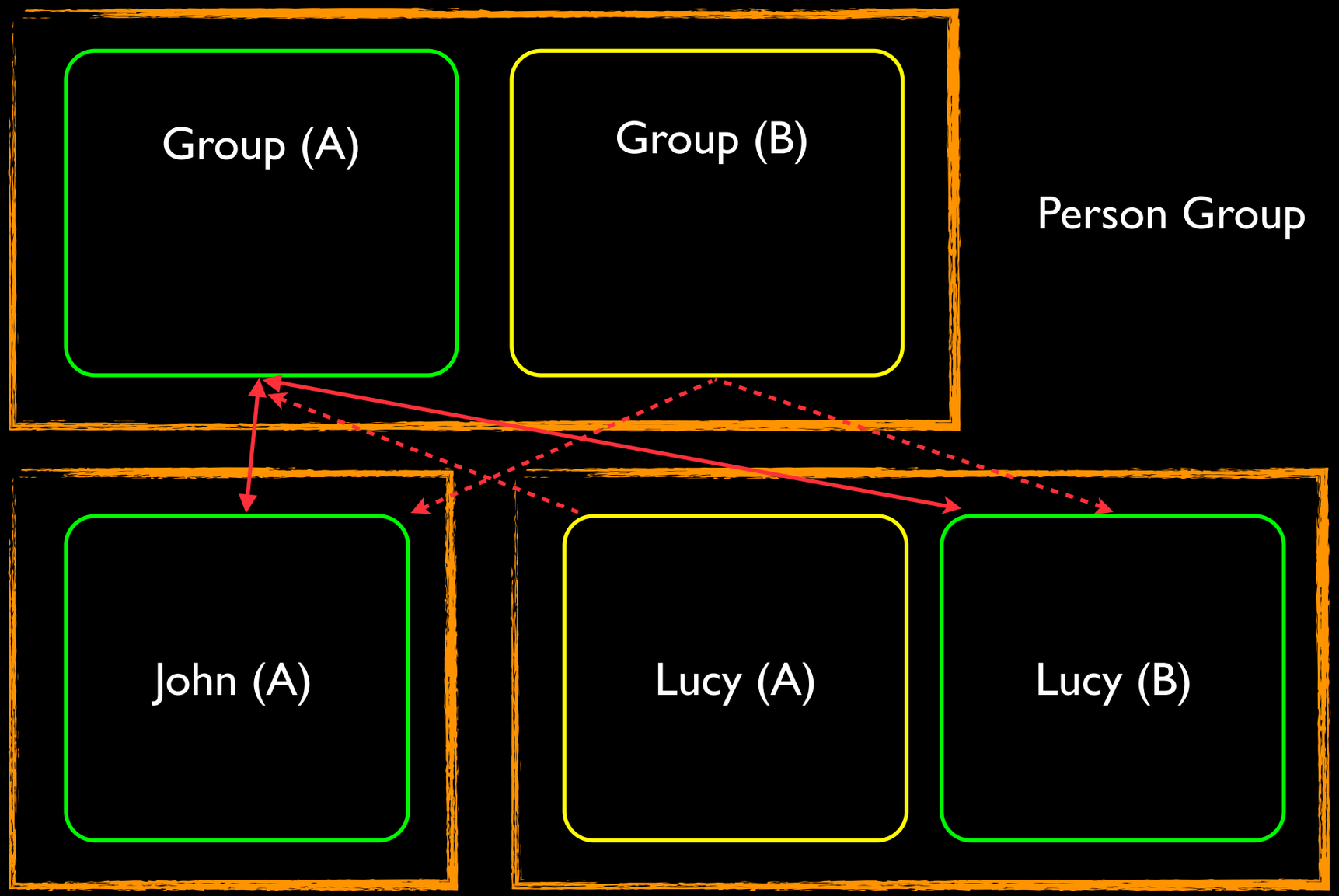
For cross persistent roots relationships accross Objects (as opposed to Documents), we restrict references to COPath on current branch

Lucy(current).parent: { Group(current) }
John(current).parent: { Group(current) }
Group(A).children: { Lucy(current), John(current) }
Group(B).children: { Lucy(current), John(current) } // constraint violation

Note: I think it's better to relax constraints on non-current branches rather obtaining Lucy.parent: { Group(A), Group(B) } which seems meaningless (e.g. for the user experience in an AddressBook application).

Person Branch and Person Group Branch

Persistent Root
Current Branch
Additional Branch
Public Reference
Private Reference
Public Composite Ref
Private Composite Ref



For cross persistent roots relationships accross Objects (as opposed to Documents), we restrict references to COPath on current branch

Persons

Lucy(A).parent: { Group(current) } // constraint violation
Lucy(B).parent: { Group(current) }
John(current).parent: { Group(current) }
Group(A).children: { Lucy(current), John(current) }
Group(B).children: { Lucy(current), John(current) } // constraint violation

Note: I think it's better to relax constraints on non-current branches rather obtaining Group(A or B).children: { Lucy(current), Lucy(B), John(current) } which seems meaningless (e.g. for the user experience in an AddressBook application).