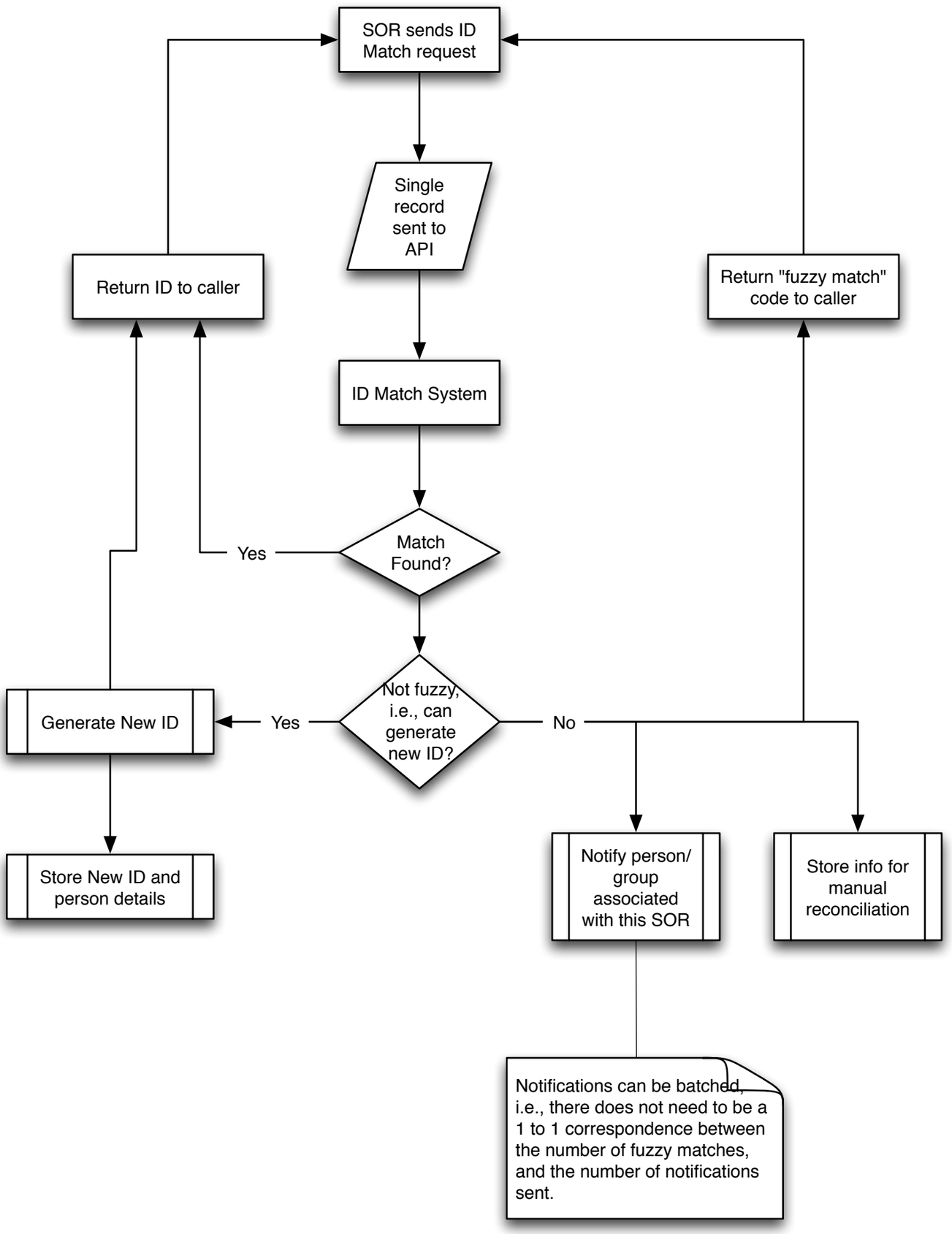
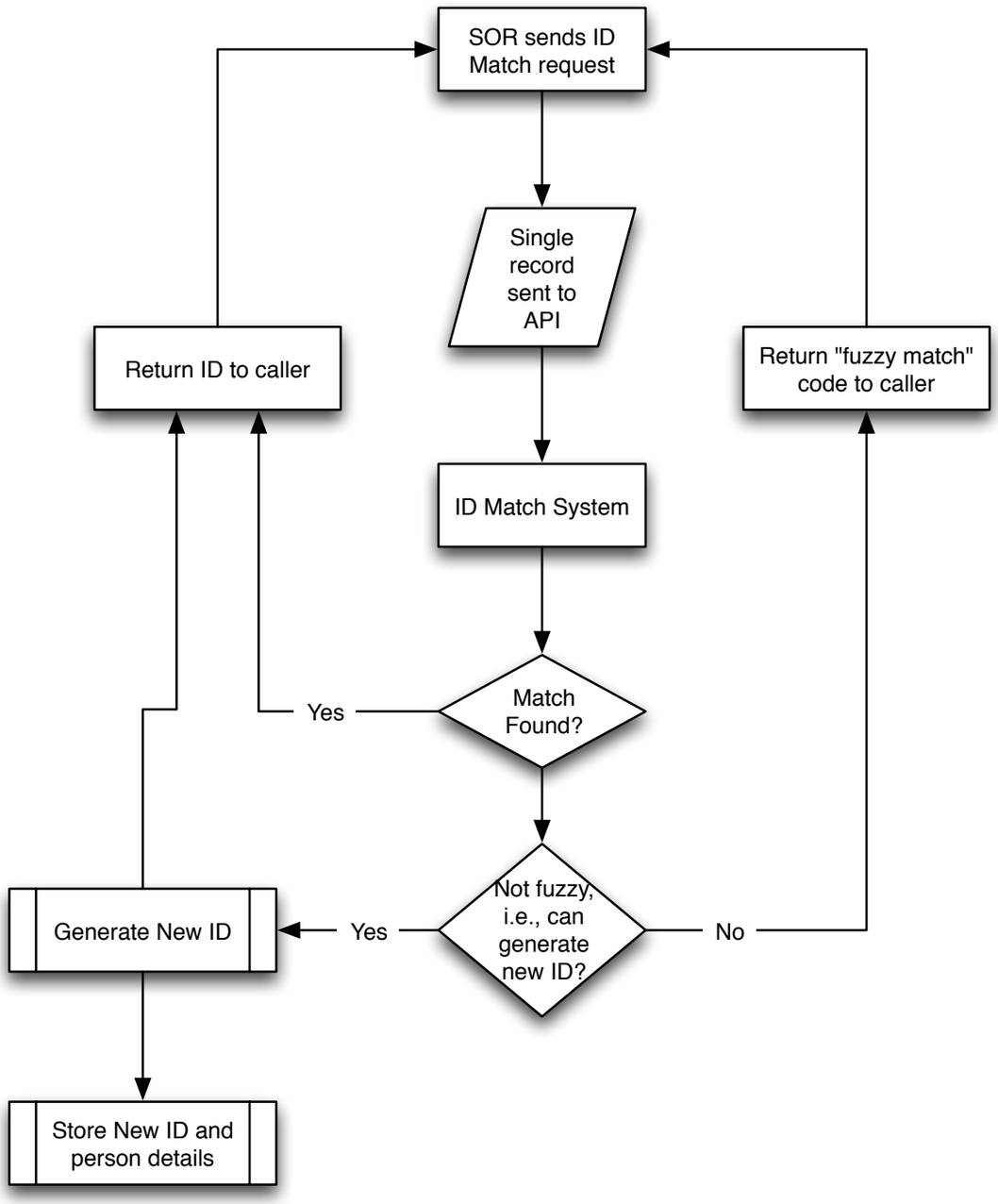
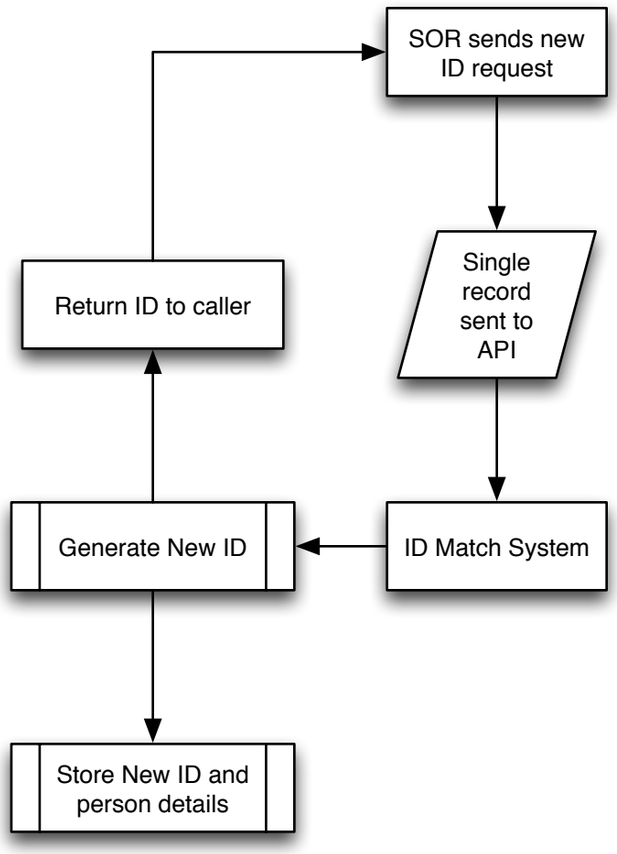
Use Case: SOR "Batch"
Description: SOR is not capable of dealing with fuzzy matches in real-time. "Batch" is in quotes because it is still a process where a single record is sent to the ID Match system and the caller waits for a response. That response may have an ID, or it may say "fuzzy match, you're gonna have to do reconciliation on this one."
Examples: Student bulk load, or new UC Path
Need to consider: Not clear if we should always just return the fuzzy match, or if the request could indicate that a fuzzy match can not be handled, so just return a code (and match performed ID) instead of the actual fuzzy matches.
Also, does the ID Match system keep track of whether or not it performed this particular match (by providing a "match performed ID")? This might be needed for when a "force new ID" request, or "add to existing id" request is sent. If so, the response will have to include the "match performed ID" which the SOR would then have to use when doing a "force new ID" or "add to existing ID" request.



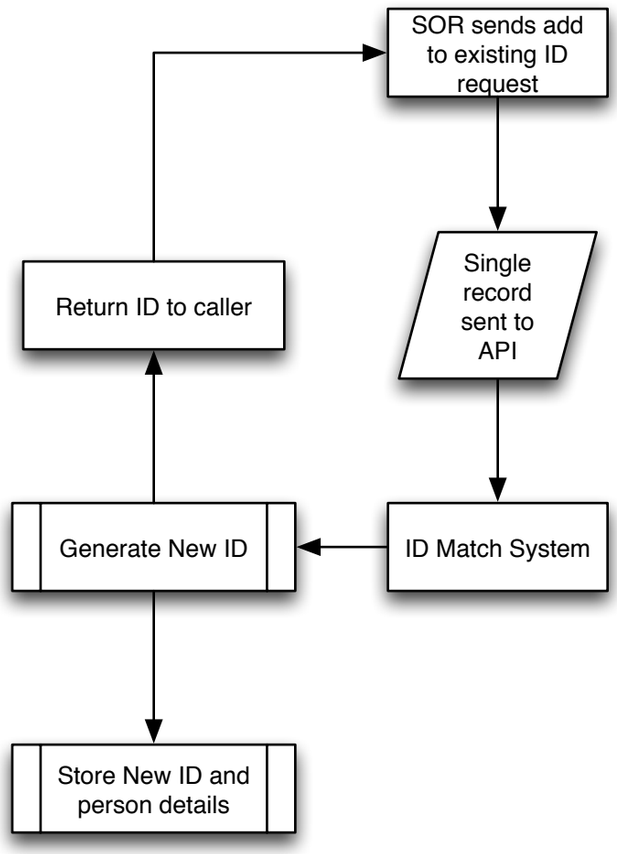
Use Case: SOR Real-Time
Description: SOR is capable of dealing with fuzzy matches in real-time, so there is no need for the ID Match system to save the information for manual reconciliation.
Examples: e*Value, PPS
Need to consider:



Use Case: Force New ID
Description: SOR has already checked to see if there is a match, got a fuzzy match result, and has determined that there is not a match, so it is asking the ID Match system to generate a new ID based on the passed in information.
Examples: Any SOR sending a "force new ID" request, or the manual reconciliation process sending a "force new ID" request.
Need to consider: It is not clear if the ID Match system would need to ensure that a match request had already been performed, i.e., it would refuse to create a new ID if there was not a cookie (i.e., some identifier that the ID match system could verify that it had already performed a check) provided back to the ID match system indicating that a fuzzy match had already been done. This means that the ID Match system would have to keep a history of the matches it performed, but could purge the history after a force was done, or after a certain period of time. Like, maybe a fuzzy match is good for *n* amount of time.



Use Case: Add to Existing
Description: SOR has already checked to see if there is a match, got a fuzzy match result, and has determined that there is a match, and wants the data it is sending to be associated with the provided ID.
Examples: Any SOR sending an add to existing ID request, or the manual reconciliation process sending an add to existing ID request.
Need to consider: It is not clear if the ID Match system would need to ensure that a match request had already been performed, i.e., it would refuse to associate the data with an existing ID if there was not a cookie (i.e., some identifier that the ID match system could verify that it had already performed a check) provided back to the ID match system indicating that a fuzzy match had already been done. This might be a good feature to have (but could be configured to not enforce it). This means that the ID Match system would have to keep a history of the matches it performed, but could purge the history after a force was done, or after a certain period of time. Like, maybe a fuzzy match is good for *n* amount of time.



Use Case: SOR Query
Description: SOR just wants to see if there is a person in the system that matches, i.e., no new ID is generated if a match does not exist.
Examples: Any SOR checking to see if there is a person that matches the given info.
Need to consider: If we are enforcing that a match request needs to be done before a "force new" request or an "add to existing" request, then perhaps this kind of query does not qualify as a match request for those two use cases.