Demographic / contact data update

Scenario Background

An employee has a change in demographic and contact information from an institutional source system.

Narrative

1. The institutional source system invokes the Person Registration and Update service either via a REST API (synchronous method) or by placing a Person Update message in the Person Update queue (asynchronous method).
2. The Person Registration and Update Service invokes the Person Match service to determine if the change to demographic information results in a match with another person in the Entity Registry.
    a. If not, the update proceeds without an attempt to link.
    b. If the demographic update triggers a match, a message is placed in the Person Verification queue, and an institutionally defined workflow is invoked to resolve the link. Once the institutionally defined workflow completes and a link has been performed (if needed), the update will continue.
3. The Person Registration and Update Service stores the updated demographic and contact data in the Master Person Store.
4. The Person Registration and Update Service places a message in the Person Update queue to indicate to downstream systems that this person's data has been updated.

Pseudocode

```
function Registry.handleUpdate(incomingAttributes) {
    var matches = PersonMatchService.findMatchesByAttributes(incomingAttributes);
    if(matches.size() == 1) {
      currentPerson = matches.getNext();
      currentPerson.replaceAttributes(incomingAttributes);
      return currentPerson;

    } else if(matches.size() > 1) {
      var matchResolutionTicket = MatchResolutionQueue.enqueue(new WorkItem(matches, incomingAttributes))
      return matchResolutionTicket;

    } else if(matches.isEmpty()) {
      /* TODO: what do we want to do here? Create a new person, or throw an error? */
      throw NoMatchingPerson_Exception;
    }

  }

}
```