

### Atmire

Lieven Droogmans

## Storage of name variants



#### Storage of name variants

- Storage in DB:
  - Additional field in the relationship db table
- Storage of possible name variants and referencing a name variant
  - 1) Relationship contains reference to name variant
  - · 2) Person doesn't contain any name variant info
  - 3) Hybrid approach: combining the above options



## 1) Relationship contains reference to name variant

- The <u>person item</u> contains:
  - · name.variant.1: Edwards, Anna
  - · name.variant.2: Michaels, Anna
  - · name.variant.3: Michaels, A.
- The <u>relationship</u> contains:
  - relationship.leftwardLabel: name.variant.2
- Creating a publication with name variant "Edwards, A." requires the name variant to be stored in the person item

Note: name.variant.1 doesn't reflect schema.element.qualifier nor an index in schema.element.



## 2) Person doesn't contain any name variant info

- The person item:
  - doesn't contain the potential name variants.
- The <u>relationship</u> contains:
  - relationship.leftwardLabel: Michaels, Anna
- Creating a publication with name variant doesn't impact the person item.
- Retrieving a list of potential name variants for a person would use an inefficient database query.



### 3) Hybrid approach

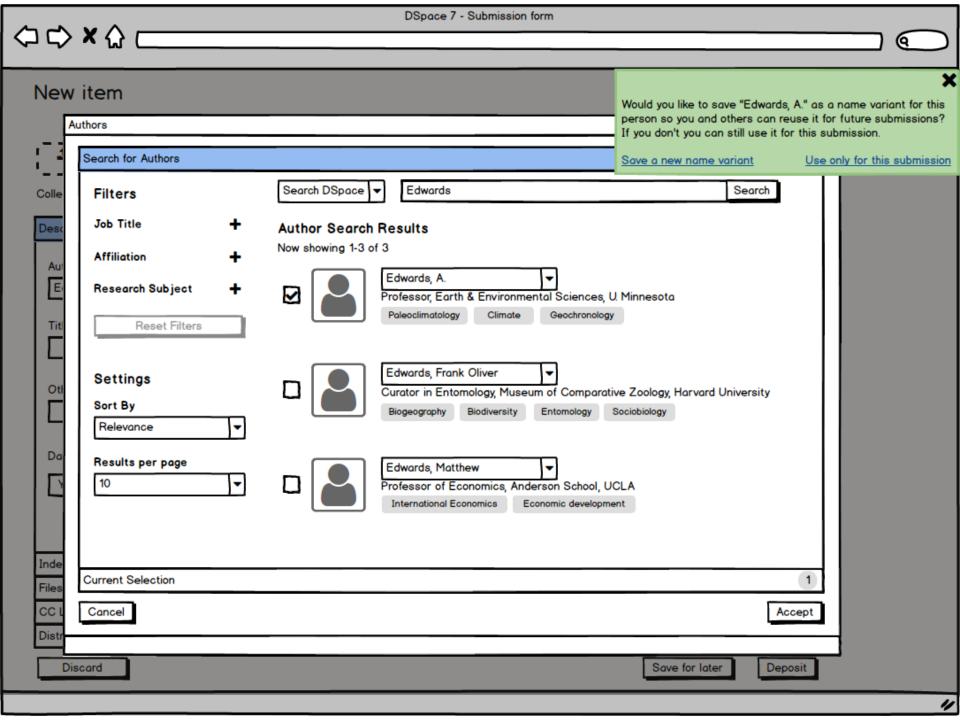
- The <u>person item</u> contains:
  - name.variant.1: Edwards, Anna
  - · name.variant.2: Michaels, Anna
  - name.variant.3: Michaels, A.
- The <u>relationship</u> contains:
  - relationship.leftwardLabel: Michaels, Anna
- Creating a publication with name variant "Edwards, A." doesn't impact the person item.
- Including "Edwards, A." as a suggested name variant for the person is possible but requires update to the person item.



### 3) Hybrid approach

- In order to ease updates to the name variants stored in the person item, an additional usability feature can be created.
- When a new name variant is chosen instead of using an existing one, the UI can verify whether the current permissions has permissions to edit the person item.
- The submitter can be offered a single-click feature to add the name variant to the person item.





### Configuration

### Configuration needs

- For which relationship and in which direction can a name variant be used.
- If applicable, where can the list of potential name variants be found.
- Which virtual metadata field should be populated with the name variant.
- When rendering the related item on the primary item's page, how should the name variant be used.



#### Areas of configuration

- A) Configuration in the submission forms
- B) Configuration in the spring virtualmetadata.xml
- C) Configuration on the publication item page



## A) Configuration in the submission forms

- The submission forms can be extended to detail:
  - For which relationship and in which direction can a name variant be used.
  - Where the list of potential name variants can be found.
- Extend the submission form config and REST response to include a nameVariantMetadata in the selectableRelationship section if a name variant can be chosen, with the metadata field containing the potential name variants.



## B) Configuration in the spring virtual-metadata.xml

- The spring virtual-metadata.xml file currently contains all the details about the virtual metadata.
- The bean can be extended to include an optional property to state it should be populated with the name variant

cproperty name="populateWithNameVariant" value="true"/>



# C) Configuration on the item page

- Create a custom template which will render identical to the standard item list rendering of the person type except for the name. This can be achieved using one of the methods below:
  - The name is no longer built from firstMetadataValue('person.familyName') + ', ' + firstMetadataValue('person.givenName') but rather from the value in the virtual metadata. If no name variant is present, this is the same value. Otherwise, this contains the name variant
  - The relationship's leftwardLabel (or rightwardLabel) is used to retrieve the name variant. If it's present, the name is no longer built from firstMetadataValue('person.familyName') + ', ' + firstMetadataValue('person.givenName') but rather from the given name variant



### **Current selection**

