# Why should we create VCard entities linked to Authorships?

...And not just use the "author name as listed" property?

...And not link to foaf:Person records unless we have good information that the name string on the publication in fact represents the person we think it does

## VIVO 1.6 includes a change from the recommendation to create foaf:Person records
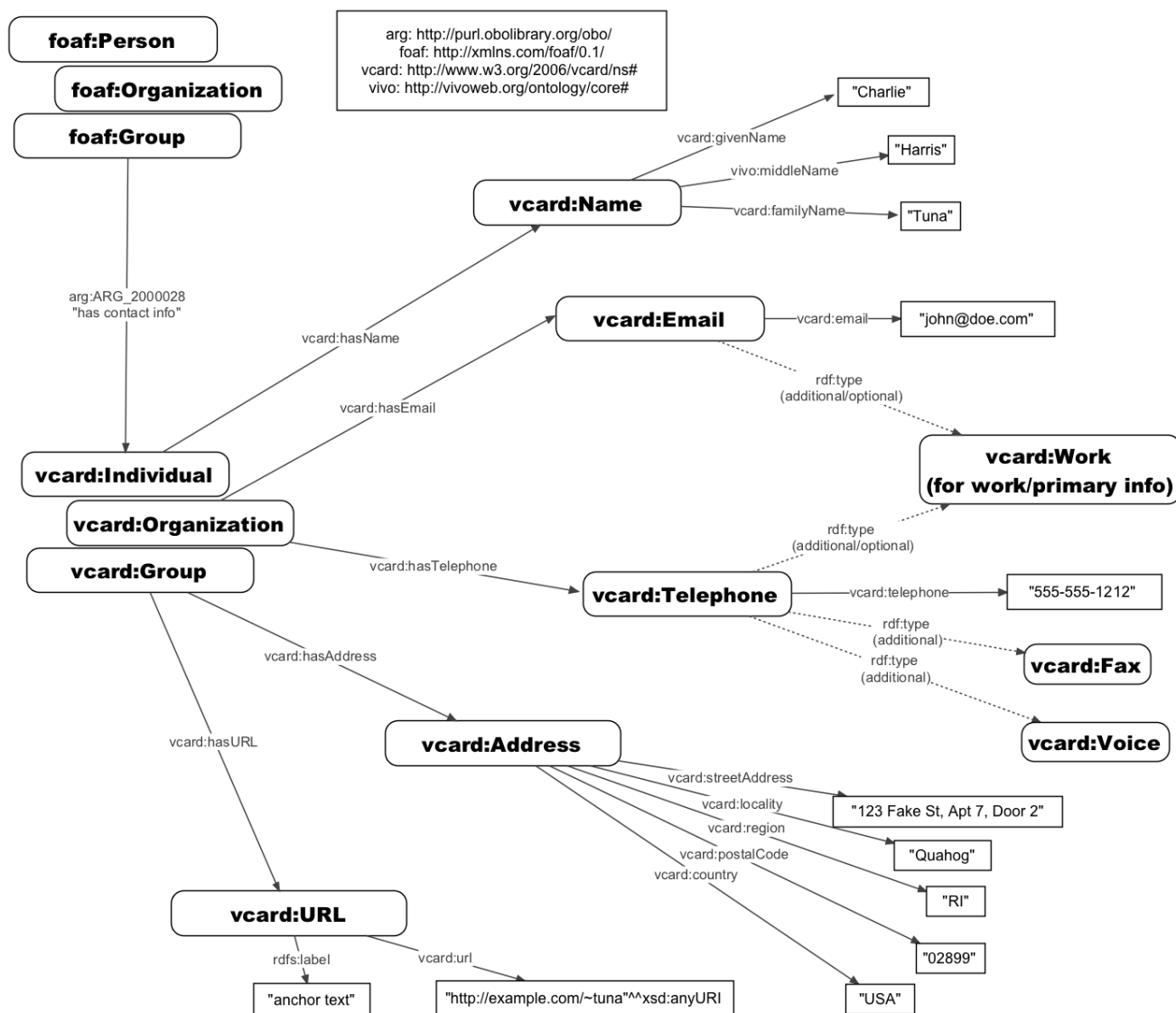
With version of the VIVO ontology prior to 1.6, the VIVO ontology team recommended linking vivo:Authorship records to foaf:Person entities, creating a new foaf:Person if no match had been found from existing persons in a given VIVO. This approach avoids creating a proliferation of data properties on the vivo:Authorship itself (e.g., firstNameAsListed, lastNameAsListed, emailAsListed, etc), but has been controversial because it does result in the creation of a large number of unknown foaf:Persons in a VIVO for a university of any significant size – often tens of thousands of them.

Strategies have been developed to limit the effect of this on the public-facing application, by designating an Institutional Internal Class in the VIVO application and then classifying individual people by one of two additional class assertions:

- For foaf:Persons that should be found through browsing or searching – designate all persons known to belong to the primary VIVO organization as internal and then restricting certain menu pages to include only these designated internal individuals
- For foaf:Persons from other institutions or having no known affiliation, creating a local UnknownPerson class and excluding all individuals in that class from the search index

With the transition from the VIVO 1.5 ontology to the VIVO-ISF for version 1.6, another option is possible – leveraging the W3C vCard ontology to enter whatever information is included in a publication citation about an author, but not creating a corresponding foaf:Person. This involves creating a vcard:Individual record in place of the foaf:Person, and then populating the vCard structure with only as much information has been provided through the citation.

The diagram below indicates the vCard ontology structure included in the VIVO-ISF:

**Namespace legend (box in diagram):**
arg: http://purl.obolibrary.org/obo/
foaf: http://xmlns.com/foaf/0.1/
vcard: http://www.w3.org/2006/vcard/ns#
vivo: http://vivoweb.org/ontology/core#

**Diagram nodes and relationships:**

foaf:Person
foaf:Organization
foaf:Group

vcard:Name — vcard:givenName → "Charlie"
vcard:Name — vivo:middleName → "Harris"
vcard:Name — vcard:familyName → "Tuna"

arg:ARG_2000028 "has contact info"

vcard:Individual
vcard:Organization
vcard:Group

vcard:hasName → vcard:Name
vcard:hasEmail → vcard:Email
vcard:Email — vcard:email → "john@doe.com"
vcard:Email — rdf:type (additional/optional) → vcard:Work (for work/primary info)

vcard:hasTelephone → vcard:Telephone
vcard:Telephone — vcard:telephone → "555-555-1212"
vcard:Telephone — rdf:type (additional/optional) → vcard:Work
vcard:Telephone — rdf:type (additional) → vcard:Fax
vcard:Telephone — rdf:type (additional) → vcard:Voice

vcard:hasAddress → vcard:Address
vcard:Address — vcard:streetAddress → "123 Fake St, Apt 7, Door 2"
vcard:Address — vcard:locality → "Quahog"
vcard:Address — vcard:region → "RI"
vcard:Address — vcard:postalCode → "02899"
vcard:Address — vcard:country → "USA"

vcard:hasURL → vcard:URL
vcard:URL — rdfs:label → "anchor text"
vcard:URL — vcard:url → "http://example.com/~tuna"^^xsd:anyURI

The vCard object has a large number of fields to represent contact information – more than we need, no doubt – but can become a way to represent each unique version of an author name plus any associated affiliation or contact information.

Note the a vCard entity is nested – the vcard:Individual itself will for most authors have a vcard:hasName object property relationship to a vcard:Name entity.  The vCard ontology can also represent organizations as authors by populating instead a vcard:Organization.

# Reasons FOR creating vCard records

1. Having vCard individuals will require minor modifications for list views and queries to look for the vCard by preference (since it represents the name form used in authoring any given publication, and will provide the closest match to a published citation) and then optionally a foaf:Person.
   a. The database migration from 1.5 to 1.6 includes creating vCards for all existing people
   b. Similarly, having vCard individuals will also require refactoring the visualization code
2. Having VCard individuals means that we can add any useful information about the author available from the publication itself or a citation in a database or from a CV – even if we only consider them a phantom name until more is known – using established VCard properties including first (given) name, last (family) name, middle (additional) name, email address, phone number, organizational affiliation, and optionally the date of the publication as a revision date to the VCard.
   - this will prove very helpful for managing name matching information as a unit – everything from one publication in a consistent VCard object – and in carrying that information forward from one run of an ingest process to another.
   - when additional information about an author *in association with the same publication* is found, the VCard object can be corrected or supplemented interactively
   - when new information indicates that two VCards from two publications represent the same person, the VCards can each be linked to the same foaf:Person

3. Having URIs for these vCards (and for any known foaf:Persons connected to the VCards) will make it easier to reference other URIs for them in the future
   a. Having vCard individuals makes the information more addressable for display as a coherent unit, and more visible data is more likely to be corrected
4. Creating vCards instead of foaf:Persons will not clutter up your VIVO with tens of thousands of unknown individuals that to the application are indistinguishable from people you do know, either because they belong to your institution or have an external identifier or URI – unless managed carefully through use of the Institutional Internal Class they will clutter menu pages, and unless asserted as Unknown Persons they will clutter search results and dilute the effectiveness of VIVO.

## Reasons AGAINST creating v:VCard records

1. More work.
   - yes, perhaps, but with the typical ingest process the work of setting everything up is 95% of the job, and once set up a slightly more complex way to structure the data is unlikely to continue to cause any additional work – except in that you can see where your data needs cleaning.
2. More data.
   - Only marginally so, if at all – properties that are not populated take up no space; an individual has a URI property and a type property in addition to the string, but if the same string is duplicated over and over on multiple authorships the net storage might be greater.
3. Others? please comment