DC TAP

Dublin Core Tabular
Application Profile

https://github.com/dcmi/dctap

DCMI Application Profiles WG

- ➢Karen Coyle (Chair)
- ➢Tom Baker, DCMI
- ➢Phil Barker
- ➢John Huck, University of Alberta
- ➢Ben Reisenberg, University of Washington
- ➢Nishad Thalhath

… and others. Thank you!

A profile

A profile is the definition of a metadata practice that constrains structures, properties and values. A profile is a reuse of vocabulary terms that have been previously defined.

Profile purpose

- Document a community practice
  - For humans
  - For machines
- Input to data creation forms
- Translate to data validation methods
- Manage ingest from data stores

TAP functions

- Define metadata structure
- Define constraints
  - cardinality
  - data types
  - value constraints
- Provide readability
  - labels
  - notes
- Note: designed with RDF metadata in mind

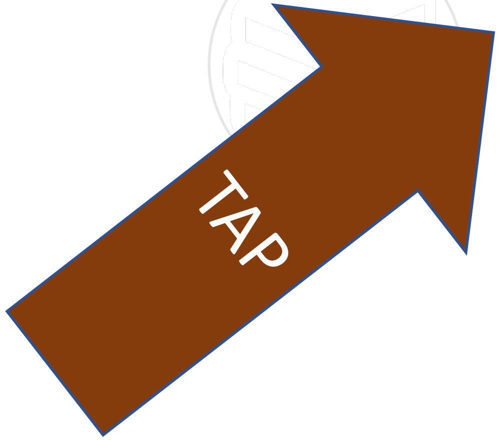
The "meta" problem

1. Vocabulary and structure for a profile

2. A Profile that describes one's metadata and constraints

3. Instance data that conforms to the profile

The "meta" problem

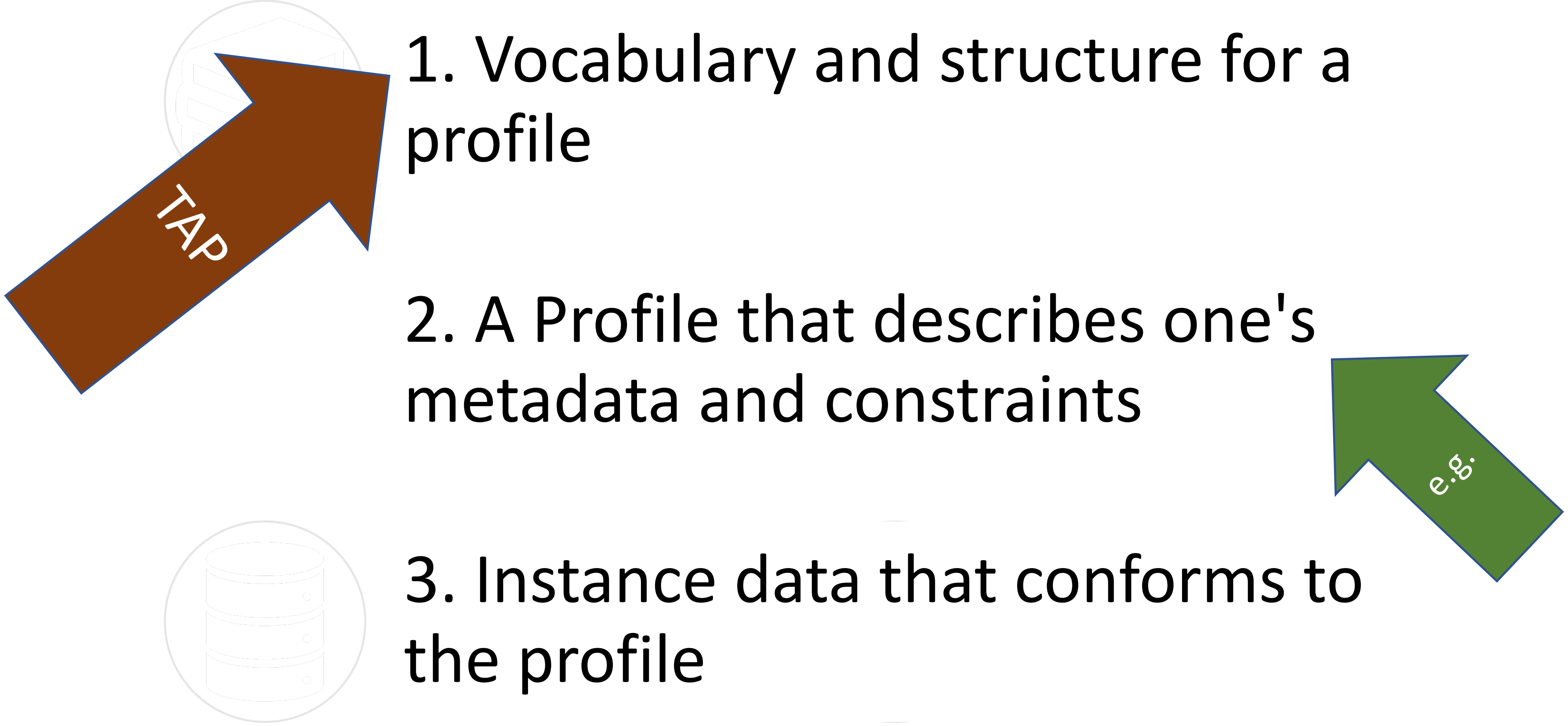


1. Vocabulary and structure for a profile

2. A Profile that describes one's metadata and constraints

3. Instance data that conforms to the profile

The "meta" problem



1. Vocabulary and structure for a profile

2. A Profile that describes one's metadata and constraints

3. Instance data that conforms to the profile

DC Tabular Application Profile
12 elements

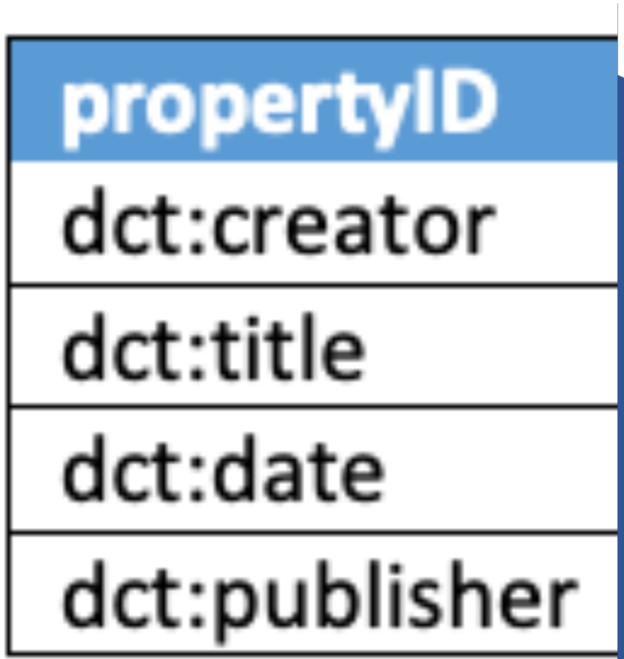
- shapeID
    - shapeLabel
- propertyID
    - propertyLabel
    - note
    - mandatory
    - repeatable
    - valueNodeType
    - valueDatatype
    - valueConstraint
    - valueConstraintType

Tabular format

shapeID	shapeLabel	propertyID	propertyLabel	mandatory	repeatable	valueNodeType	valueDataType	valueShape
book	Book	dct:creator	Author	y	y			person
		dct:title	Title	y	n	LITERAL	xsd:string	
		dct:date	Year of publication	n	n	LITERAL	xsd:year	
person	Author	foaf:name	Name	y	n	LITERAL	xsd:string	
		foaf:mbox	Email	n	n	IRI		
		dct:date	Birth year	n	n	LITERAL	xsd:year	

Tabular format

shapeID	shapeLabel	propertyID	propertyLabel	mandatory	repeatable	valueNodeType	valueDataType	valueShape
book	Book	dct:creator	Author	y	y			person
		dct:title	Title	y	n	LITERAL	xsd:string	
		dct:date	Year of publication	n	n	LITERAL	xsd:year	
person	Author	foaf:name	Name	y	n	LITERAL	xsd:string	
		foaf:mbox	Email	n	n	IRI		
		dct:date	Birth year	n	n	LITERAL	xsd:year	



propertyID
dct:creator
dct:title
dct:date
dct:publisher

Only propertyID is required

Add data type

propertyID	valueDataType
dct:creator	xsd:string
dct:title	xsd:string
dct:date	xsd:year
dct:publisher	xsd:string

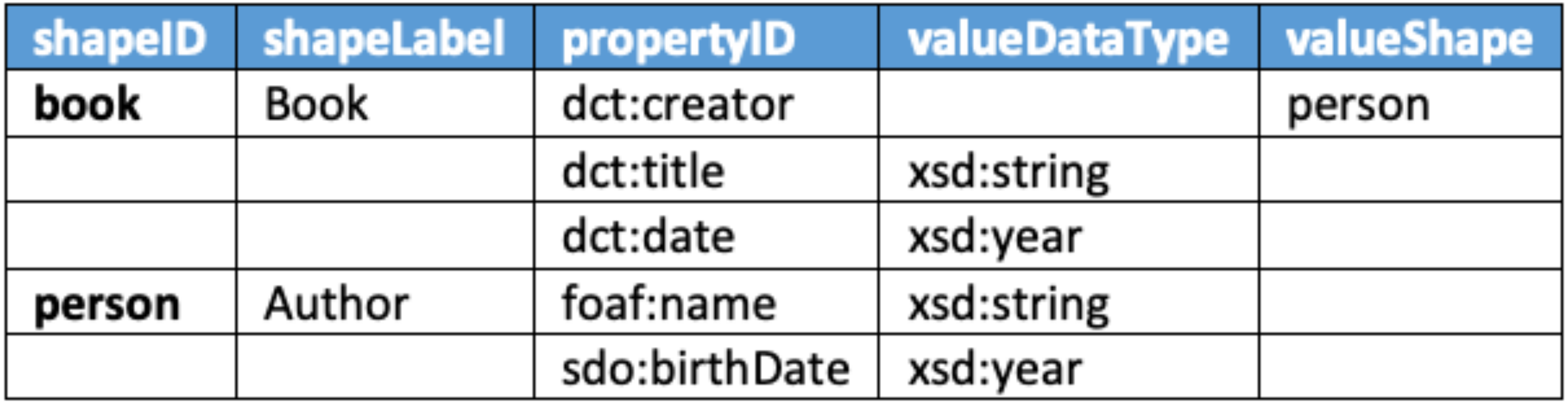
Include labels and notes

propertyID	propertyLabel	note
dct:creator	Author	name in natural order
dct:title	Title	take title from the cover
dct:date	Date of publication	year only

Cardinality

propertyID	valueDataType	mandatory	repeatable
dct:creator	xsd:string	y	y
dct:title	xsd:string	y	n
dct:date	xsd:year	y	n
dct:publisher	xsd:string	n	n

Creating <u>shapes</u> for "things" described in the metadata



shapeID	shapeLabel	propertyID	valueDataType	valueShape
book	Book	dct:creator		person
		dct:title	xsd:string	
		dct:date	xsd:year	
person	Author	foaf:name	xsd:string	
		sdo:birthDate	xsd:year	

Shape

shapeID	shapeLabel	propertyID	propertyLabel	mandatory	repeatable	valueNodeType	valueDataType	valueShape
book	Book	dct:creator	Author	y	y			person
		dct:title	Title	y	n	LITERAL	xsd:string	
		dct:date	Year of publication	n	n	LITERAL	xsd:year	
person	Author	foaf:name	Name	y	n	LITERAL	xsd:string	
		foaf:mbox	Email	n	n	IRI		
		dct:date	Birth year	n	n	LITERAL	xsd:year	

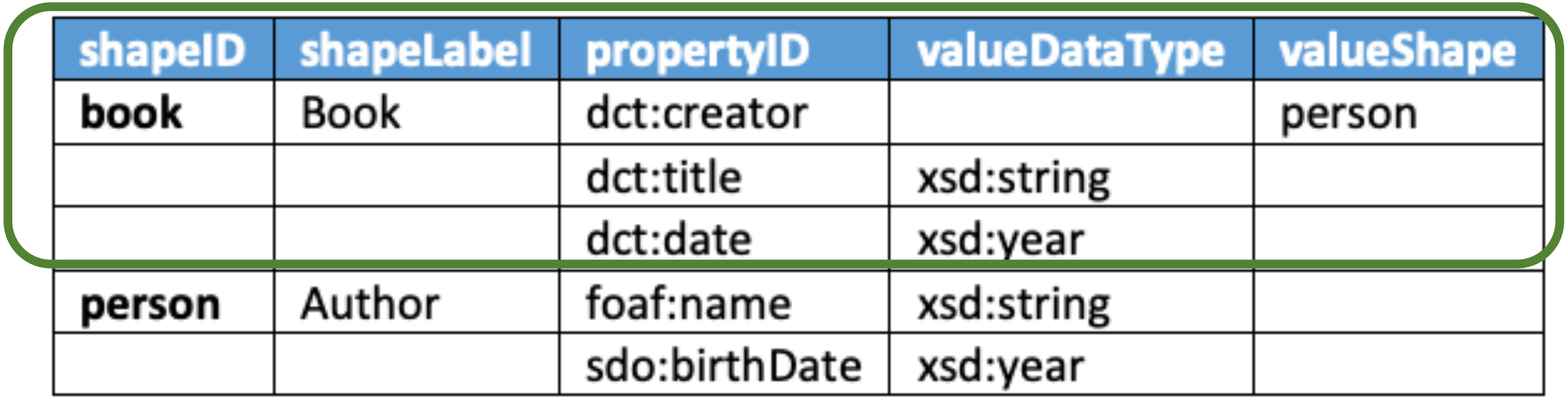
A group of statement constraints that share a subject node and are identified with the same shapeID.

Statement constraint

shapeID	shapeLabel	propertyID	propertyLabel	mandatory	repeatable	valueNodeType	valueDataType	valueShape
book	Book	dct:creator	Author	y	y			person
		dct:title	Title	y	n	LITERAL	xsd:string	
		dct:date	Year of publication	n	n	LITERAL	xsd:year	
person	Author	foaf:name	Name	y	n	LITERAL	xsd:string	
		foaf:mbox	Email	n	n	IRI		
		dct:date	Birth year	n	n	LITERAL	xsd:year	

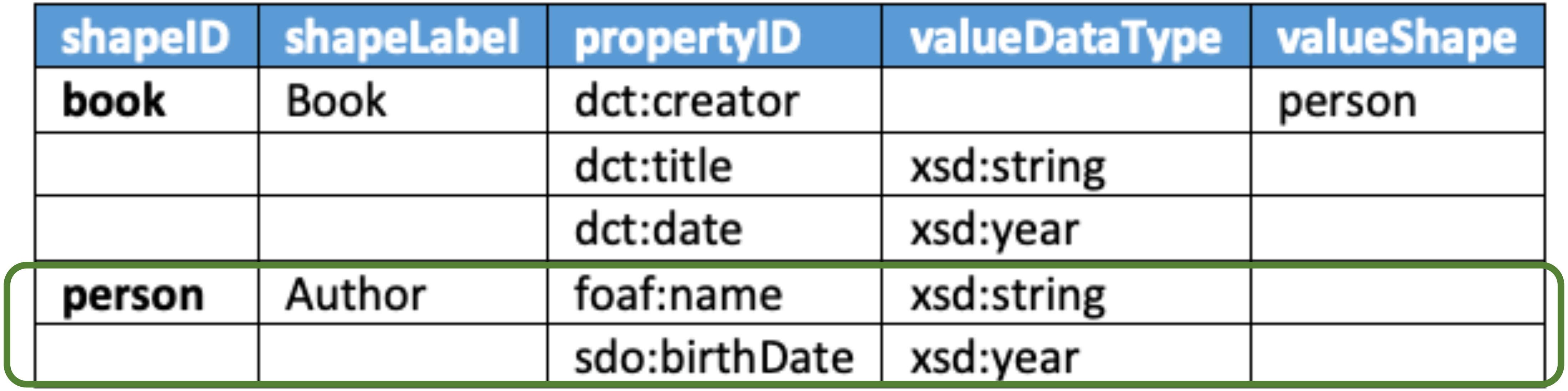
A statement constraint consists of a property and any rules that constrain the property and its value.

"Book" shape



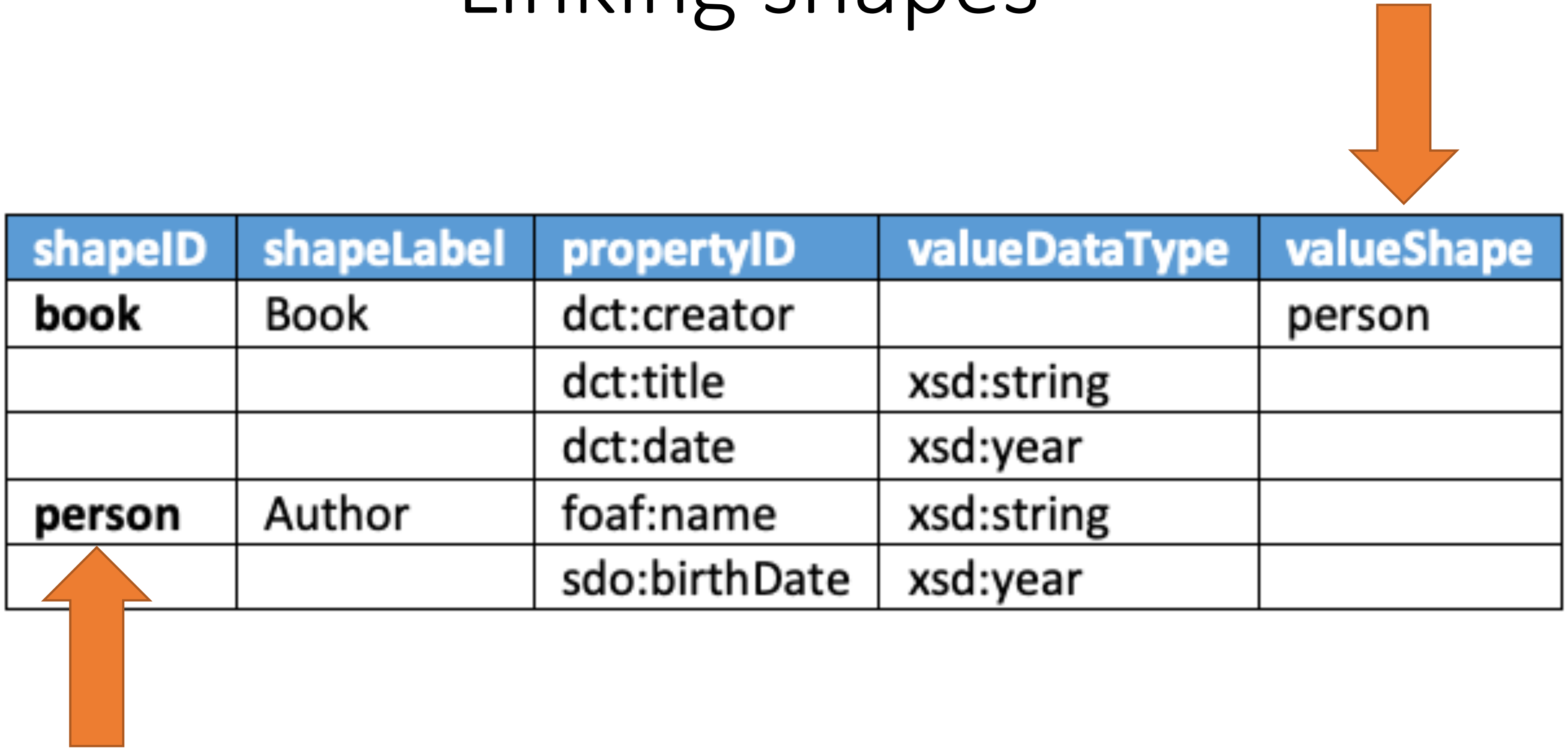
shapeID	shapeLabel	propertyID	valueDataType	valueShape
book	Book	dct:creator		person
		dct:title	xsd:string	
		dct:date	xsd:year	
person	Author	foaf:name	xsd:string	
		sdo:birthDate	xsd:year	

"Person" shape



shapeID	shapeLabel	propertyID	valueDataType	valueShape
book	Book	dct:creator		person
		dct:title	xsd:string	
		dct:date	xsd:year	
person	Author	foaf:name	xsd:string	
		sdo:birthDate	xsd:year	

Linking shapes



shapeID	shapeLabel	propertyID	valueDataType	valueShape
book	Book	dct:creator		person
		dct:title	xsd:string	
		dct:date	xsd:year	
person	Author	foaf:name	xsd:string	
		sdo:birthDate	xsd:year	

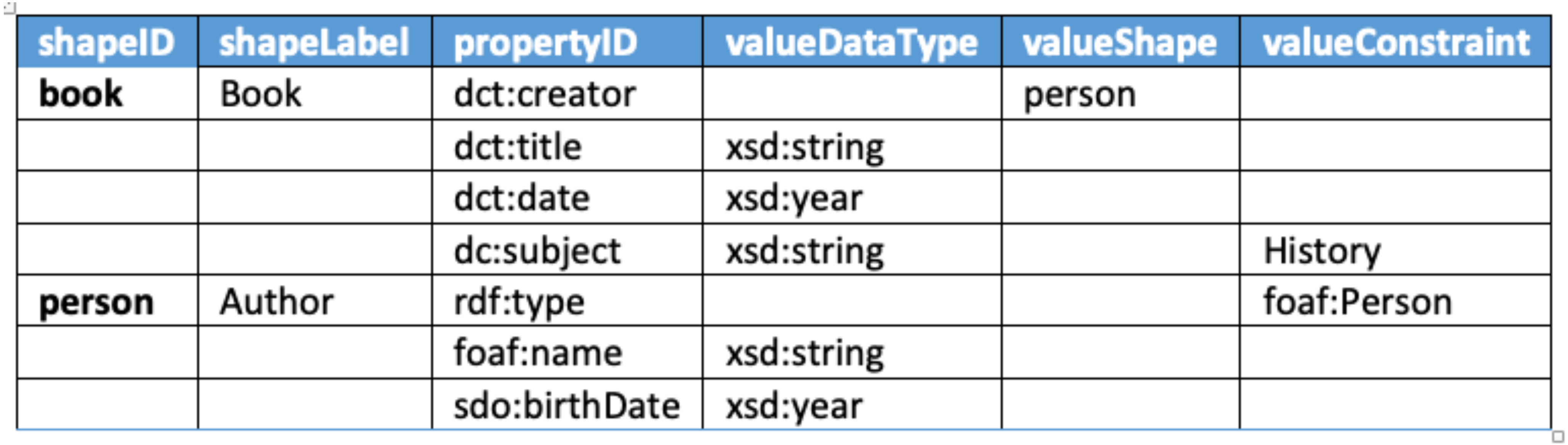
RDF nodes and literal property values

propertyID	valueNodeType	valueDataType
dct:creator	LITERAL	xsd:string
dct:title	LITERAL	xsd:string
dct:date	LITERAL	xsd:year
dct:subject	IRI	

RDF node type

Literal value type

Value constraints



shapeID	shapeLabel	propertyID	valueDataType	valueShape	valueConstraint
book	Book	dct:creator		person	
		dct:title	xsd:string		
		dct:date	xsd:year		
		dc:subject	xsd:string		History
person	Author	rdf:type			foaf:Person
		foaf:name	xsd:string		
		sdo:birthDate	xsd:year		

Only where value is a single string or IRI.

Unresolved: value constraint types

Value constraint types

- Beyond single constraint strings
  - URI stems (value from this vocabulary)
  - ranges (min/max on numbers)
  - pick lists (strings or URIs or URI stems)
  - etc.

valueConstraint	valueConstraintType
http://id.loc.gov/	uriStem
1990 - 2021	min - max
red | green | blue	list

Unresolved but needed functions (the "whatabouts")

Where to record namespaces and prefixes

- no place in table – can we standardize a separate file?
- CSV on the web style? https://www.w3.org/TR/tabular-data-primer/#metadata
- Could also include any administrative information, and alternate column names
- issue #3

How to handle multiple values in a table cell

- "IRI or BNODE"
- "red" "blue" "green"
- "title@en" "titre@fr"
- issue #4

Unresolved but needed functions

To quote or not to quote?

- CSV cells by definition are strings … are quotes needed?
- https://tools.ietf.org/html/rfc4180

i18n

- How can we internationalize the table (and its contents)?

Unresolved but needed functions

Boolean values

- Define values (true/false) or allow any set of values?
- issue #7

Open/closed

- Open = other properties or shapes are allowed
- Closed = only what is in the template is allowed
- Shape level? profile level? both?
- issue #8

Out of scope (tabular format limitations)

Interaction between properties or shapes (and, or, not)

- e.g. foaf:name OR (foaf:familyName AND foaf:givenName)
- e.g. mandatory Professor shape or TA shape per course
- some can be done with structure but that's limited

Tabular designs requiring multiple "sheets"

Some DC TAP resources

Github repository: https://github.com/dcmi/dctap/

Examples: https://github.com/dcmi/dctap/tree/main/examples

Please participate

Open and respond to issues on github
Join the mailing list
Offer data to be profiled
Offer profiles to be TAP'd

Questions? Comments?

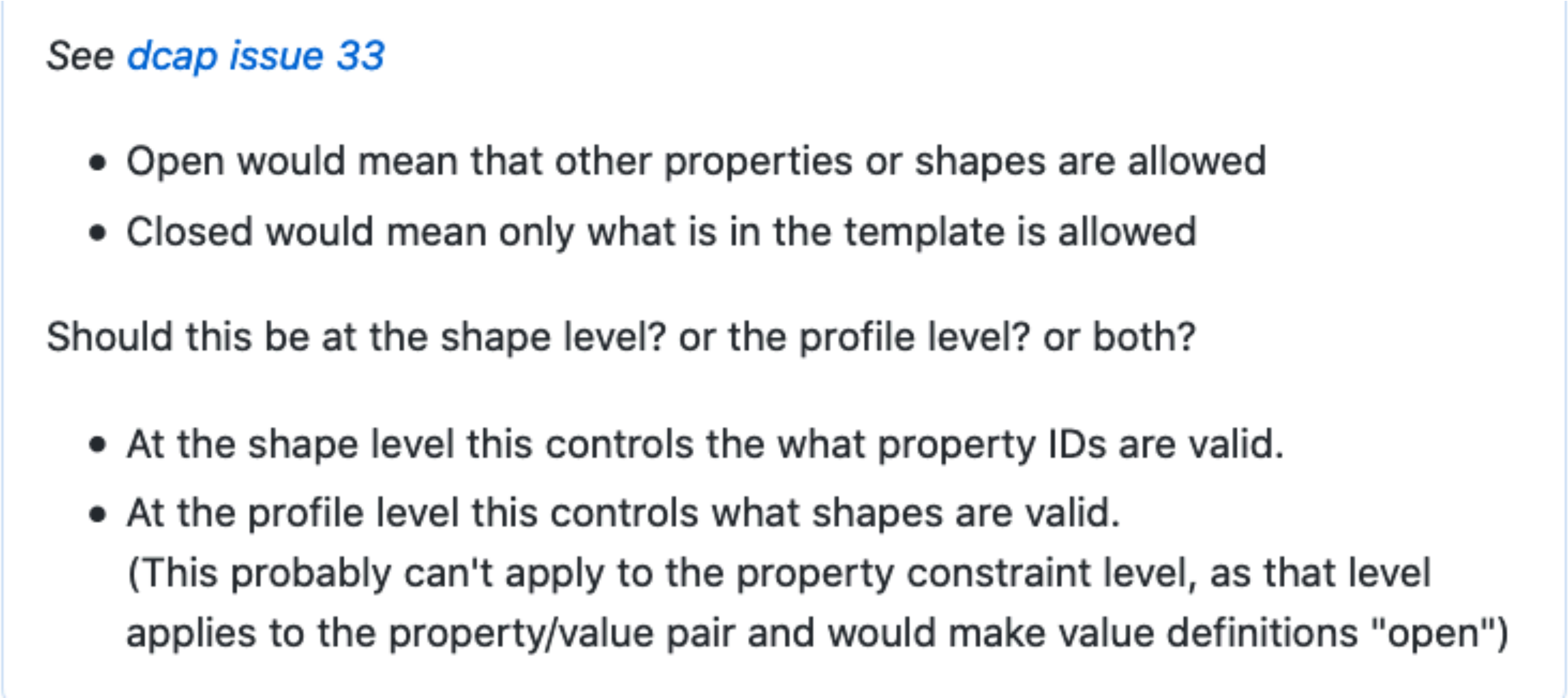
shapeID	shapeLabel	propertyID	propertyLabel	mandatory	repeatable	valueNodeType	valueDataType	valueShape
book	Book	dct:creator	Author	y	y			person
		dct:title	Title	y	n	LITERAL	xsd:string	
		dct:date	Year of publication	n	n	LITERAL	xsd:year	
person	Author	foaf:name	Name	y	n	LITERAL	xsd:string	
		foaf:mbox	Email	n	n	IRI		
		dct:date	Birth year	n	n	LITERAL	xsd:year	

https://github.com/dcmi/dctap

How should the TAP terms be formalized as a vocabulary? #6

- Presume IRIs for terms are needed
- Is there something other than RDF that would work?
- Does RDF help i18n, e.g. multiple language labels?

Open/closed - Issue #8



See dcap issue 33

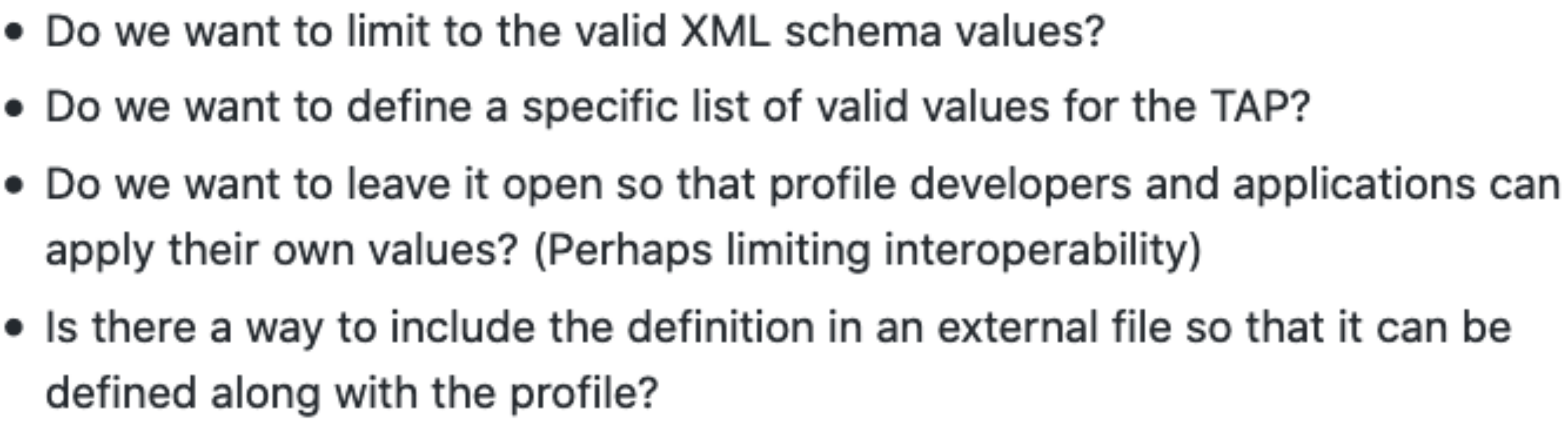
- Open would mean that other properties or shapes are allowed
- Closed would mean only what is in the template is allowed

Should this be at the shape level? or the profile level? or both?

- At the shape level this controls the what property IDs are valid.
- At the profile level this controls what shapes are valid.
  (This probably can't apply to the property constraint level, as that level applies to the property/value pair and would make value definitions "open")

Is there a default?

Boolean values – Issue #7



- Do we want to limit to the valid XML schema values?
- Do we want to define a specific list of valid values for the TAP?
- Do we want to leave it open so that profile developers and applications can apply their own values? (Perhaps limiting interoperability)
- Is there a way to include the definition in an external file so that it can be defined along with the profile?

Thanks, everyone!

https://github.com/dcmi/dctap/