



---

# GTM Level 1 Proposal

April 6, 2008

*Feedback from the committee  
given in text boxes like this  
one.*



# GTM Level 1 Proposal

---

- **This is a first GTM level 1 proposal**
  - intended as a strawman to kick-start discussion
  - will be properly formalized once feedback indicates that the basic form of the proposal is accepted by the committee
- **Feedback wanted!**
  - is this headed in the right direction?
  - what is good?
  - what is bad?
  - what is missing?
  - what is too much?
  - what is not clear?
  - ...
- **Let's not go too deep in the details yet**
  - things will change
  - we don't know how this will be specified formally yet



# Mapping to TMCL

---

- **GTM level 1 will have a defined mapping to TMCL**
  - this mapping is not fully defined in this proposal yet
- **Note: TMCL schemas are expressed as topic maps**
- **Note: GTM will *not* define an interchange format for the graphical representation**
  - it defines the shapes of diagrams and their mapping to TMCL *only*
  - interchange must be done with TMCL (that is, Topic Maps)



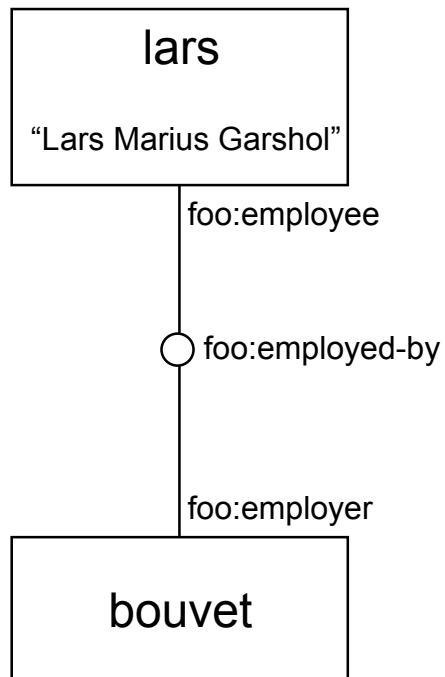
## Issue: name change?

---

- **Should we replace terms level 0 and level 1 with**
  - level 0 -> CTM-G
  - level 1 -> TMCL-G

# Topic (not topic type) in level 0

---



# Topic types

---

foo:person

- **Topic types are always boxes**
- **A QName (or id) giving the subject (or item) identifier must be present**
- **Prefixes are declared with floating text in CTM syntax**

%prefix foo <http://psi.example.org/>

## Issue: multiple subject identifiers

---

- **What do we do if the topic type has multiple subject identifiers?**
  - show them in the box?
  - show them as an annotation?
  - don't show them at all?
  - show as a separate box?

*If we add an extra division to the topic box where instance-level assertions can be made, the extra identifiers can be put there.*

foo:person

# Properties

---

foo:person	
tm:name	1..1
foo:given-name	1..1
foo:family-name	1..1
foo:email : string	1..1
foo:biography : uri	0..*
@ oasis:language	

- **A division for names may or may not be present**
- **A division for occurrences may or may not be present**
  - must always be the second division
  - first can be empty
- **Cardinality is omissible**
- **Datatypes are omissible**
- **@ oasis:language means foo:biography can be scoped with topics of this type**



## Issue: line after heading

---

- Should there be a line between the heading and the names part?

Yes.

foo:person	
tm:name	1..1
foo:given-name	1..1
foo:family-name	1..1
foo:email : string	1..1
foo:biography : uri	0..*
@ oasis:language	



## Issue: TMCL mismatch

---

- The type of an occurrence is only specified once, for all topic types, in TMCL
- GTM specifies it separately every time
- This is a mismatch
- But, is it a problem?

*We don't consider this a problem. GTM specifies that the datatype for a single occurrence type must be the same everywhere. Tools would typically update it everywhere when the user changes it in one place.*

## Issue: datatype separator

---

- In topic boxes GTM uses “:” between the typing topic and the datatype
- CTM uses “^^” to indicate datatype
- Should GTM be aligned?

*No, we don't care that this is different, and ^^ is just too ugly.*

foo:person	
tm:name	1..1
foo:given-name	1..1
foo:family-name	1..1
foo:email : string	1..1
foo:biography ^^ uri	0..*
@ oasis:language	

# Abstract topic types

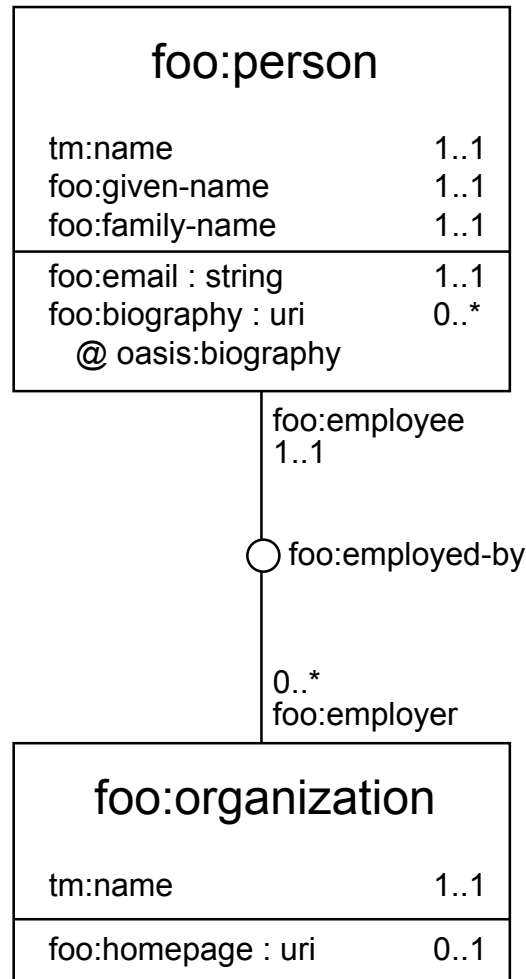
---

<i>foo:person</i>	
tm:name	1..1
foo:given-name	1..1
foo:family-name	1..1
foo:email : string	1..1
foo:biography : uri	0..*
@ oasis:biography	

- **As in UML**
  - write the topic type identifier in italics

*What do you do about this on a whiteboard? Underline would work on a whiteboard, but is ugly on the screen. An alternative is to wrap the identifier in angle brackets. It seems like UML also uses angle brackets.*

# Binary associations



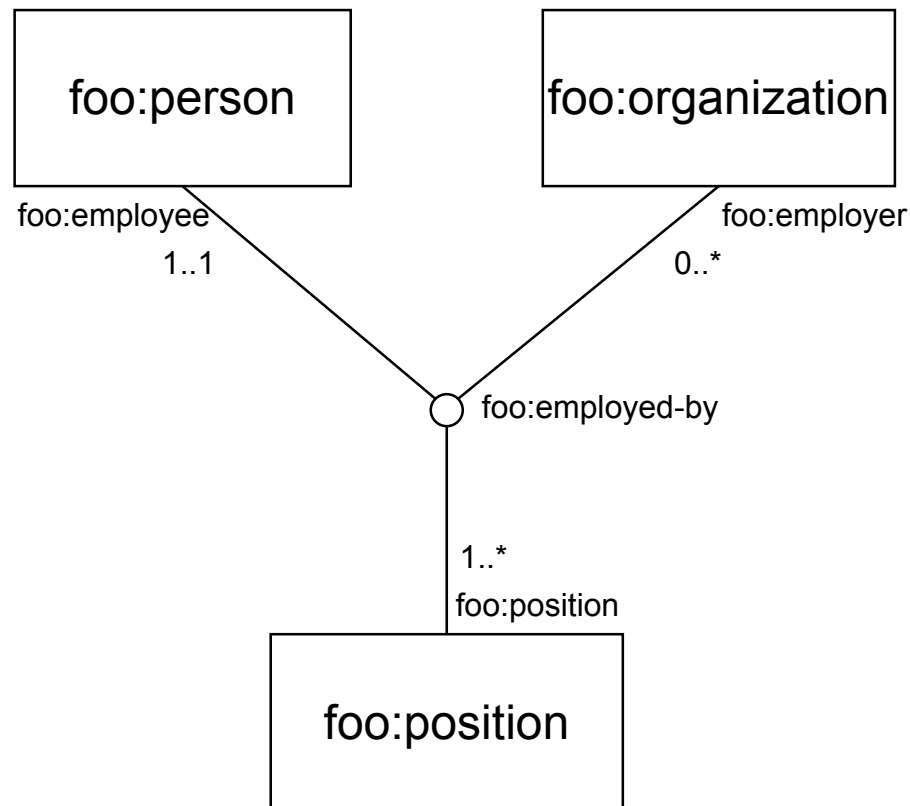
- **Binary associations are lines**
  - association type given in middle
  - role types given near player
  - cardinality from player side given near player

*The association does not need to be a line under the circle. It's really two lines from the boxes to the circle.*

*Do the lines have to be lines? Can they be curves? Can they be a sequence of lines? What to do when a box is in the way?*

*The positioning of the roles & cardinalities is the opposite of what UML has. This is an issue.*

# N-ary associations



- **N-ary associations use a circle to represent the association type**
  - behaviour is otherwise as for binaries
  - Position really means “position title” here, as in “CTO”. So “CTO” can have any number of associations where a person has that position in an organization

# Unary associations

---

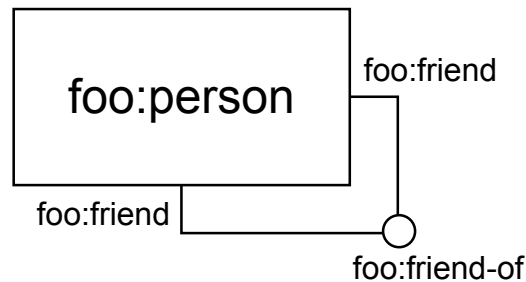


- **Unary associations follow the same pattern**
- **There must be restrictions on the possible cardinalities here**

*There is an interaction with scope here, which would make multiple associations possible after all. This needs more work in TMCL, so we leave this for now.*

# Symmetric associations

---



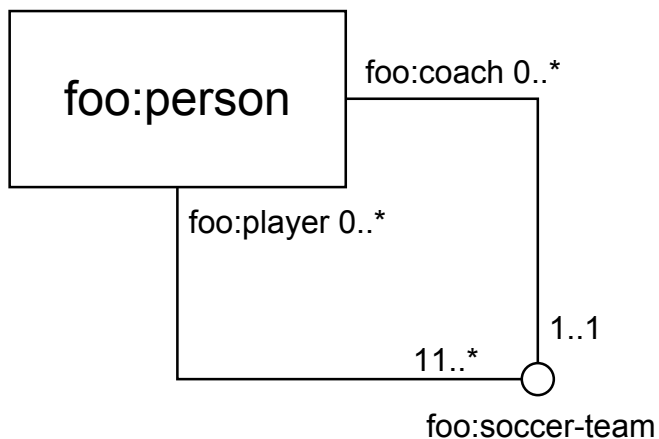
- **Binary associations where the same role type appears on both sides**
- **Issue: cardinality on both sides must be consistent**

*Solve the same way as previous consistency issue: the standard requires the cardinalities to be consistent. How this is implemented is another issue.*

*Could we have two boxes for foo:person here? Is that legal? What about two different topic types? Needs to be considered.*



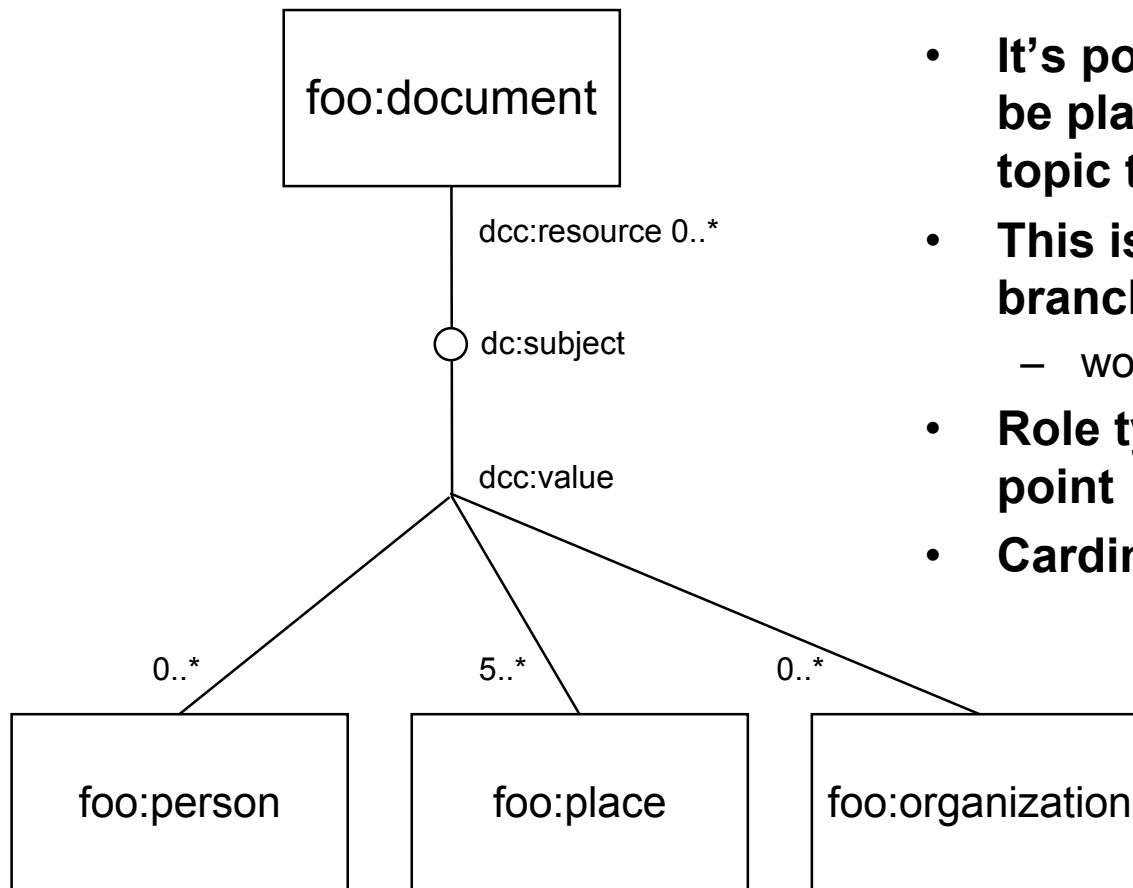
# Repeatable roles



- For n-ary associations the cardinalities of roles in instance associations can be given
- For binary associations they are fixed at 1..1 (except if repeated, as in symmetric)
- Disclaimer: this is *not* an example of good modelling

*What happens if foo:soccer-team appears elsewhere with other roles/cardinalities/whatever?*

# Roles with many player types



- It's possible for a role type to be played by more than one topic type
- This is represented by branching the line
  - works for both n-ary and binary
- Role type given at branching point
- Cardinalities given at player

*We might want a shape of some kind at the branching point. It couldn't be a circle, but perhaps a diamond.*



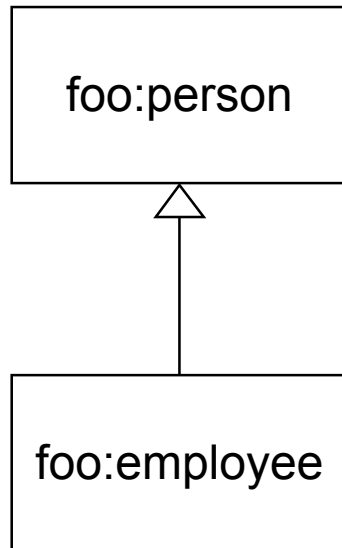
## **Issue: roles played by many types**

- **TMCL does not support roles played by more than one type**
- **Is this acceptable?**

*We are not sure if TMCL supports this right now, but it will.*

# Subtyping

---



- **Effectively UML notation**
- **Maps to supertype/subtype association between topic types**



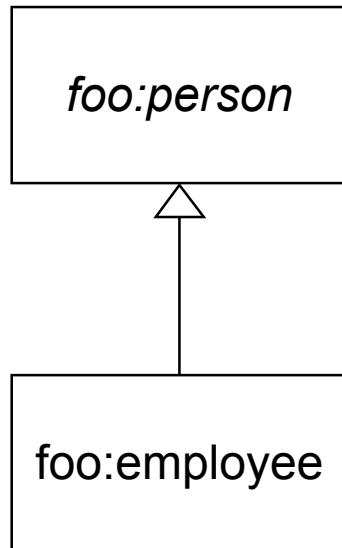
## Issue: fill the diamond?

---

- In some tools it's awkward to draw an unfilled diamond
- Possible resolutions:
  - use a better tool
  - require filled diamond
  - make fill optional

# Text notes

---



- Text notes documenting the diagrams are allowed
- They are given as simple rectangles containing the text note

We realize that employee is strictly speaking a role type, but...



## **Issue: text notes term**

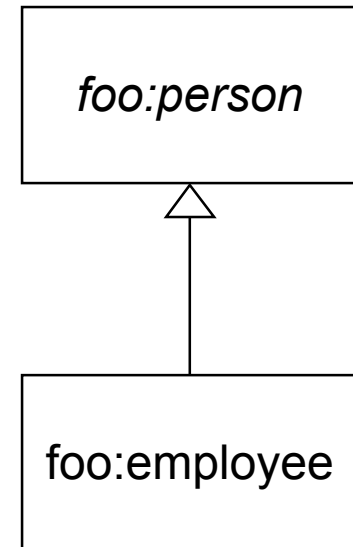
---

- **Some people claim “comment” is more intuitive?**
- **Should we change the term?**

## Issue: notes similar to topics

---

- **Are notes too visually similar to topics?**
- **Possible resolutions:**
  - leave as is
  - use turned corners instead
  - use dashed line
  - ...



We realize that employee is strictly speaking a role type, but...



# Identity constraints

---

foo:organization	
tm:name	1..1
foo:homepage : uri	0..1
identifier	1..*

- **A separate division for these**
  - divisions have a fixed order
  - names, occurrences, identities
- **Predefined names**
  - locator (subject locator)
  - identifier (subject identifier)
  - itemid (item identifier)
- **Datatypes fixed to “uri”**



## Issue: division identification

---

- **Divisions are currently identified by their order**
  - ie: names first, occurrences second, identifiers last
- **Some people did not like the defined order**
- **This also means divisions sometimes must be left empty**
- **Is it better to make the divisions visually distinctive?**
- **Possible resolutions:**
  - leave as is
  - use different line types (need two or three)
  - use small text symbols (like “@”, “^”, ...)
  - use some small indicator before the text
    - this might replace divisions completely



## Issue: What about names?

---

- **This proposal does not put names for typing topics in the diagram**
  - the rationale is that space savings are crucial for readability in large diagrams
- **So where are the names specified?**
- **In this proposal that is considered out of scope**
  - editing tools can allow the names to be edited manually
  - and/or they can generate default names from the PSIs
  - or they can ignore them entirely

*This is not sufficient. It needs to be possible to show diagrams to end-users with the actual names in them somehow. It would also be good to be able to give the different directional names for associations in the diagrams themselves. In fact, the issue of scoped names recurs for all typing topics.*



## Issue: scope support

---

- The support for scope needs more work
- Open questions:
  - interaction of cardinality with scope
  - multiple types of scoping topics
  - ...

*This is really a TMCL issue. TMCL support for scope is really limited at the moment. The real question is whether this is sufficient. Needs to be discussed in that context.*



## Issue: reification support

---

- Should there be any?
- What should it look like?

*This can be handled quite elegantly if we have a general concept of “linkage” (like the circled numbers), because this really is just a link between the reified statement type and the topic type used to represent reified statements.*



## Issue: assertion constraints

---

- Should query constraints be supported?
- Should regular expression constraints be supported?

*We could support this using something like the comments, with some kind of box that contains TMQL assertions.*

*Not sure about the regular expressions.*

## Issue: omitted information

---

- **GTM must indicate which information can be omitted**
- **How should omitted information be interpreted?**
  - should there be default cardinalities, for example?
  - is it possible to generate TMCL without specifying these?
  - is it better to just leave the issue of defaults to tools?
- **What about visual shorthands for omitted information?**
  - these would serve as indicators that something is present but not shown
  - is that useful? is it clutter? is it too much complexity?

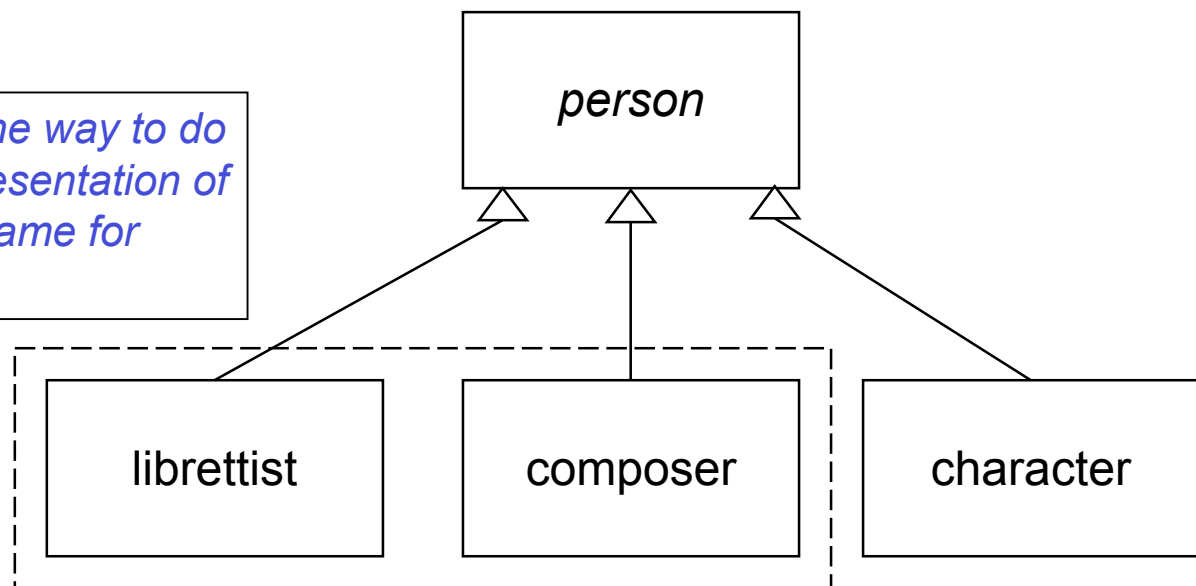
*This is partly a TMCL issue. In any case, it was deferred for later discussion.*

# Issue: overlapping types

---

- **It is possible for topic types to overlap**
  - for example: in the Italian Opera topic map the librettist and composer types overlap, in the sense that topics can be instances of both
- **In TMCL overlap must be explicitly stated to be allowed**
- **Should GTM support this?**
  - if so, how?

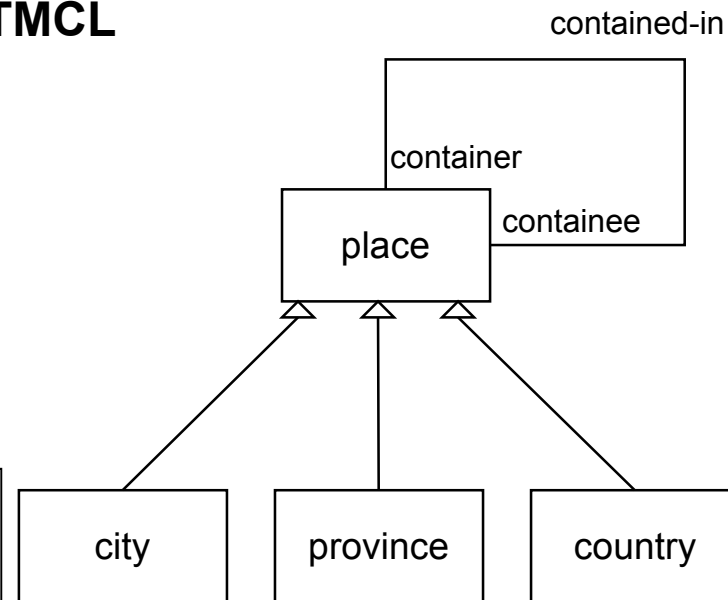
*Yes, this should be supported. One way to do it might be to use the level 0 representation of associations. (If so, why not the same for subtyping?)*





# Issue: codependent role player types

- **Given an association type contained-in that joins**
  - cities, provinces, and countries, where
  - cities must be in a province, and provinces must be in a country
  - it's not allowed to connect cities directly with countries
- **This constraint is expressible in TMCL**
- **Should it be expressible in GTM?**
- **If so, how?**



*Yes, this is wanted. Seems like it can be done by simply repeating the association type pattern for the valid type combinations. Need to verify whether this will actually generate the correct TMCL.*

# Composability of diagrams

---

- **It needs to be possible to indicate how multiple diagrams are composed into a whole**
  - for example, how they are broken up into pages, and what is on which page
- **Must also know how to merge multiple diagrams into a single schema**
  - this also means that repeated declarations for the same type have to be mergeable into a single set of constraints
- **We need a “package” concept, where “package” corresponds to TMCL schema**
  - which “package” a diagram belongs to can be indicated visually
  - however, this might be metadata about the diagram
  - do we want to support metadata in general?
  - should call “packages” schemas, like in TMCL



## **Issue: support for multiple schemas?**

---

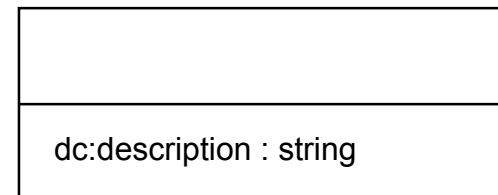
- **TMCL allows multiple schemas to be mixed in a single topic map**
- **Should GTM allow diagrams to indicate which schema they “belong to”?**
- **The current proposal stays well clear of this**
  - this is related to the package issue

## **Issue: documentation of non-topic types**

---

- **Should it be possible to make a GTM diagram that says**
  - dc:description is an occurrence type with datatype string?
- **That is, without assigning the occurrence type to any topic type...**
- **This would make it possible to create diagrams for ontology fragments**
  - on the other hand: is that useful?
- **Is the thing on the right the solution?**

*Use a star where the topic type identifier should be, to indicate wildcard semantics.*





## **Issue: Mixing levels 0 and 1**

---

- **Whether this should be allowed or not was discussed**
- **The need to be able to include example instances of the topic types and example statements persuaded the committee that it *should* be allowed**
- **Later, other uses for this were found**
  - for example: what if one wants to make statements about typing topics?
    - level 0 makes this easy
      - proposal of using filled circle for association *instances*
    - alternatively, there might be a separate division for statements about the typing topics (however, this will only work for topic types)



## Issue: Including CTM?

---

- **We may want to support including free-form CTM in diagrams as well**
  - how does this relate to including level 0 constructs?



## **Issue: Constraints on instances?**

---

- **Should this be supported?**
- **It's really a TMCL issue more than a GTM issue**



## **Issue: Role types as topic types?**

---

- **How can diagrams indicate that a role type is also a topic type?**
  - the initial proposal allows this, simply by including a topic type box for the role type
  - however, this means that when users see the role type they are not necessarily aware that it is also a topic type
  - it should be possible to connect the two somehow
  - different proposals were made:
    - a line of some sort, or
    - some kind of reference symbol (like a circle with numbers in it)
    - tools could support navigating this reference by “jumping”





## **Issue: scope on association types?**

- **How do we constrain that?**



## **Issue: subtyping of non-topic types**

- **Should it be possible to subtype anything else than just topic types?**
- **If so, how?**