Australian Government
Geoscience Australia

Automatically Calculating the Adherence to License Requirements

Nicholas J. Car
Geoscience Australia
nicholas.car@ga.gov.au

Matthew P. Stenson
CSIRO
matthew.stenson@csiro.au

Australian Government
Geoscience Australia

Automatically Calculating the Adherence to License Requirements

Nicholas J. Car
Geoscience Australia
nicholas.car@ga.gov.au

Matthew P. Stenson
CSIRO
matthew.stenson@csiro.au

Do you agree to something undefined and untestable before viewing this presentation?

[X] I agree

Continue

Motivation

- Reduce the management effort in long-tail data repositories
- Reduce risk of license condition breaches
- Handle licence entailment better
    - Currently handled poorly, all or nothing

About the authors

- Both apply computer science & engineering to data management
- < 2015 Both part of a government research institute: CSIRO
- 2015 Both built a data sharing repo described here: BA Repo
- 2016 Nicholas moves to a data custodian govt. department: GA

Nicholas Car

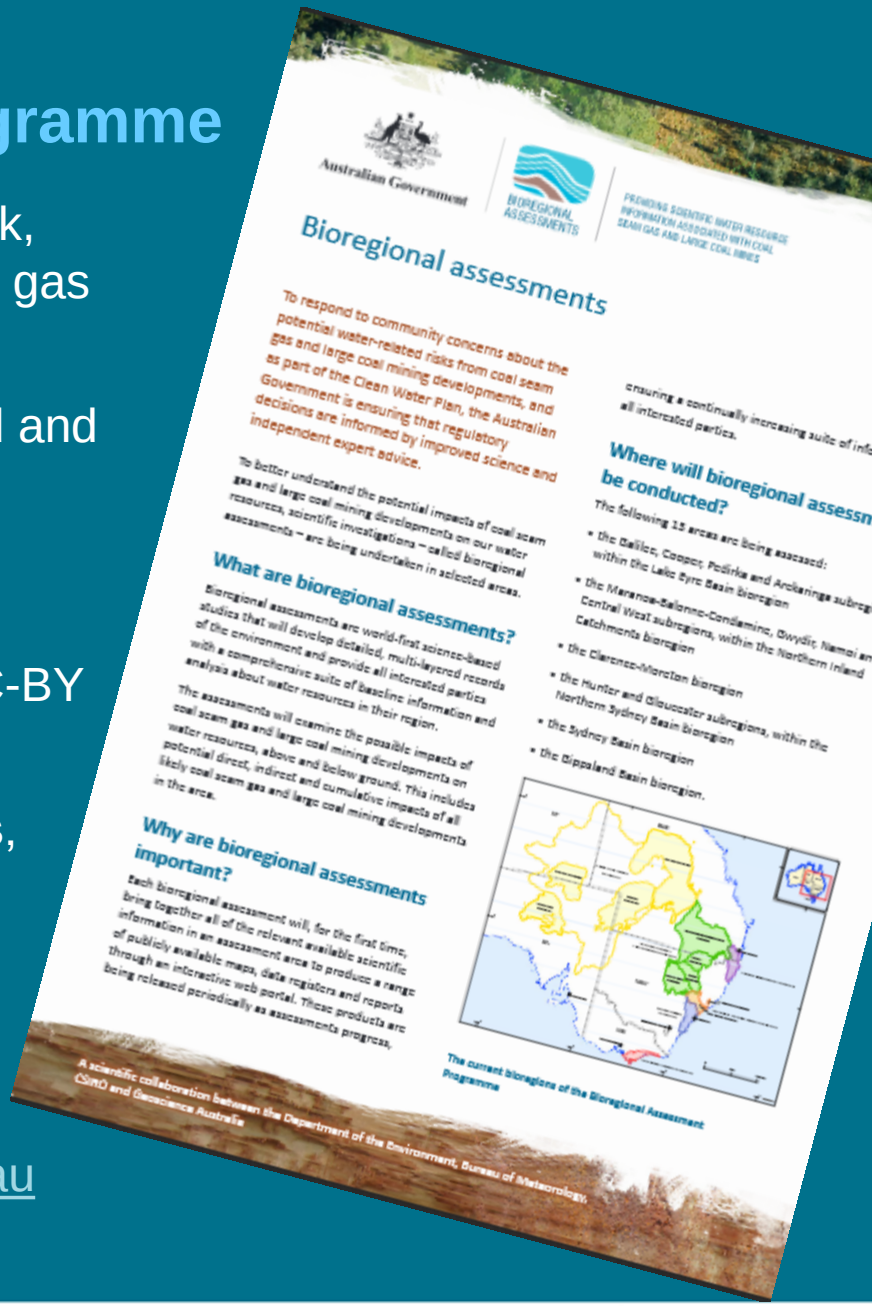
Matthew Stenson

Objectives

- As much as possible, automatically handle datasets according to their license conditions
- Prompt for user actions for un-handlable conditions

Bioregional Assessments Programme

- Multidisciplinary program of scientific work, assessing impacts of coal and coal seam gas mining on water resources.
- Several thousand datasets both collected and generated by the BAP, stored long-term, enabling reuse over time.
- ‘Derived’ datasets – generated by BAP – licenced for open access and reuse – CC-BY 3.0.
- ‘Source’ datasets from mining companies, community groups and government agencies have a range of licences and restrictions.



- http://www.bioregionalassessments.gov.au

Bioregional Assessments Programme

Original dataset licensing methodology

- Manually collect licence information for each Source dataset
  - Store the info in the catalog of datasets
- Handle license conditions based on classes
- Present a stacked attribution statement for each of ancestors:

  © CSIRO 2011
  © Commonwealth of Australia (Bureau of Meteorology) 2012
  © Commonwealth of Australia (Bioregional Assessments) 2015

Resulted in 60+ distinct licences

Bioregional Assessments Programme

Original dataset licensing methodology

- Manually collect licence information for each Source dataset
  - Store the info in the catalog of datasets
- Handle license conditions based on classes
- Present a stacked attribution statement for each ancestors:

  © CSIRO 2011
  © Commonwealth of Australia (Bureau of Meteorolog
  © Commonwealth of Australia (Bioregional Assessme

Most restrictive wins

Usually all or nothing

Bioregional Assessments Programme

Original dataset licensing methodology

- Manually collect licence information for each Source dataset
  - Store the info in the catalog of datasets
- Handle license conditions based on classes
- Present a stacked attribution statement for each dataset, based on its ancestors:

  © CSIRO 2011
  © Commonwealth of Australia (Bureau of Meteorolog
  © Commonwealth of Australia (Bioregional Assessm

Was recorded manually

Requires a knowledge of dataset provenance

Could be automatic, could be useful in other ways

Hypothesis

By implementing a faceted license handling system, we can more automatically deliver datasets according to their particular conditions

Methods - Provenance

Provenance:

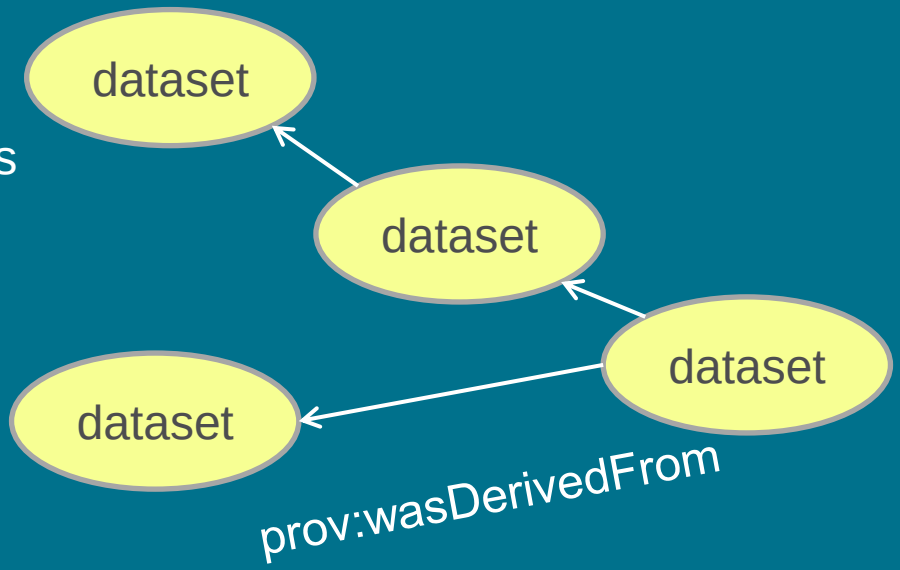
"informatics uses the term 'provenance' to mean the lineage of data, as per data provenance, with research in the last decade extending the conceptual model of causality and relation to include processes that act on data and agents that are responsible for those processes"

Wikipedia, https://en.wikipedia.org/wiki/Provenance#Computer_science

Provenance graph of dataset is the point of truth regarding dataset relations and therefore dependent licence relations

I'm using the PROV Data Model[1]

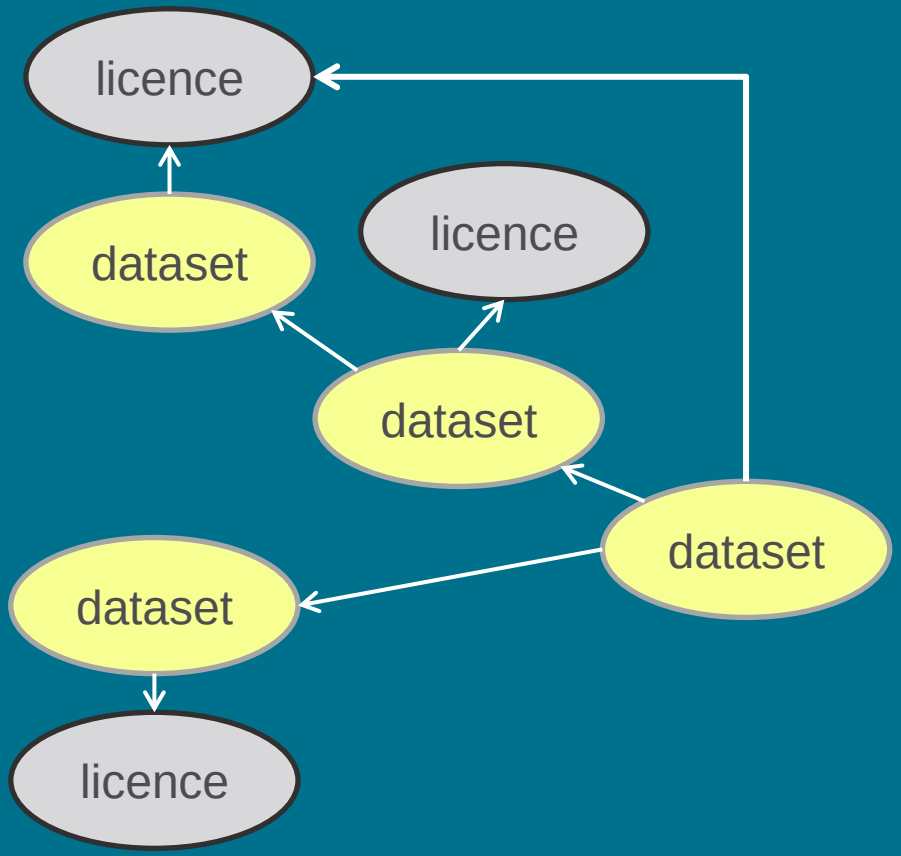
[1] https://www.w3.org/TR/prov-dm/



Methods - Provenance

Associate licenses, as distinct objects, with datasets

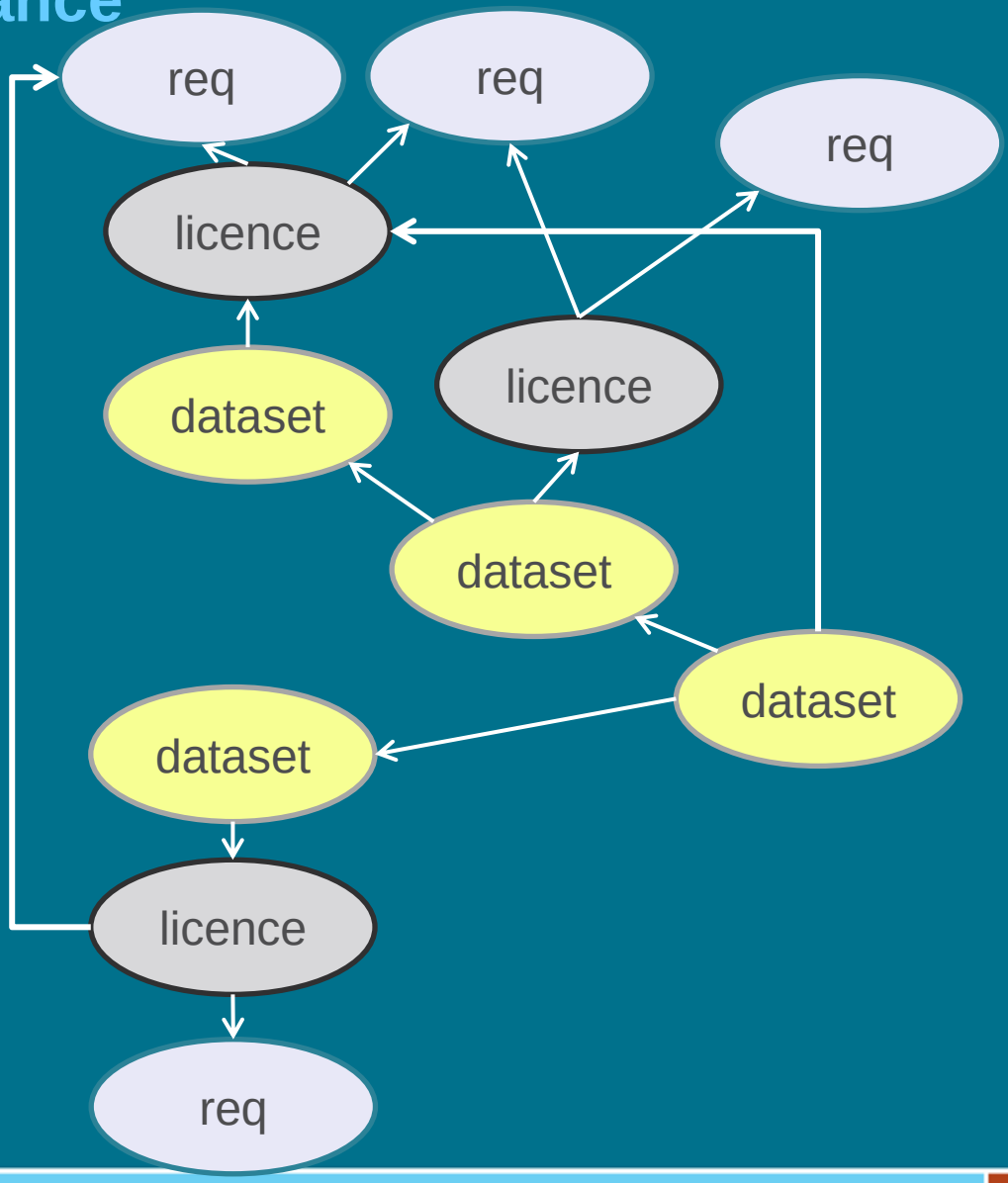
Reuse!



Methods - Provenance

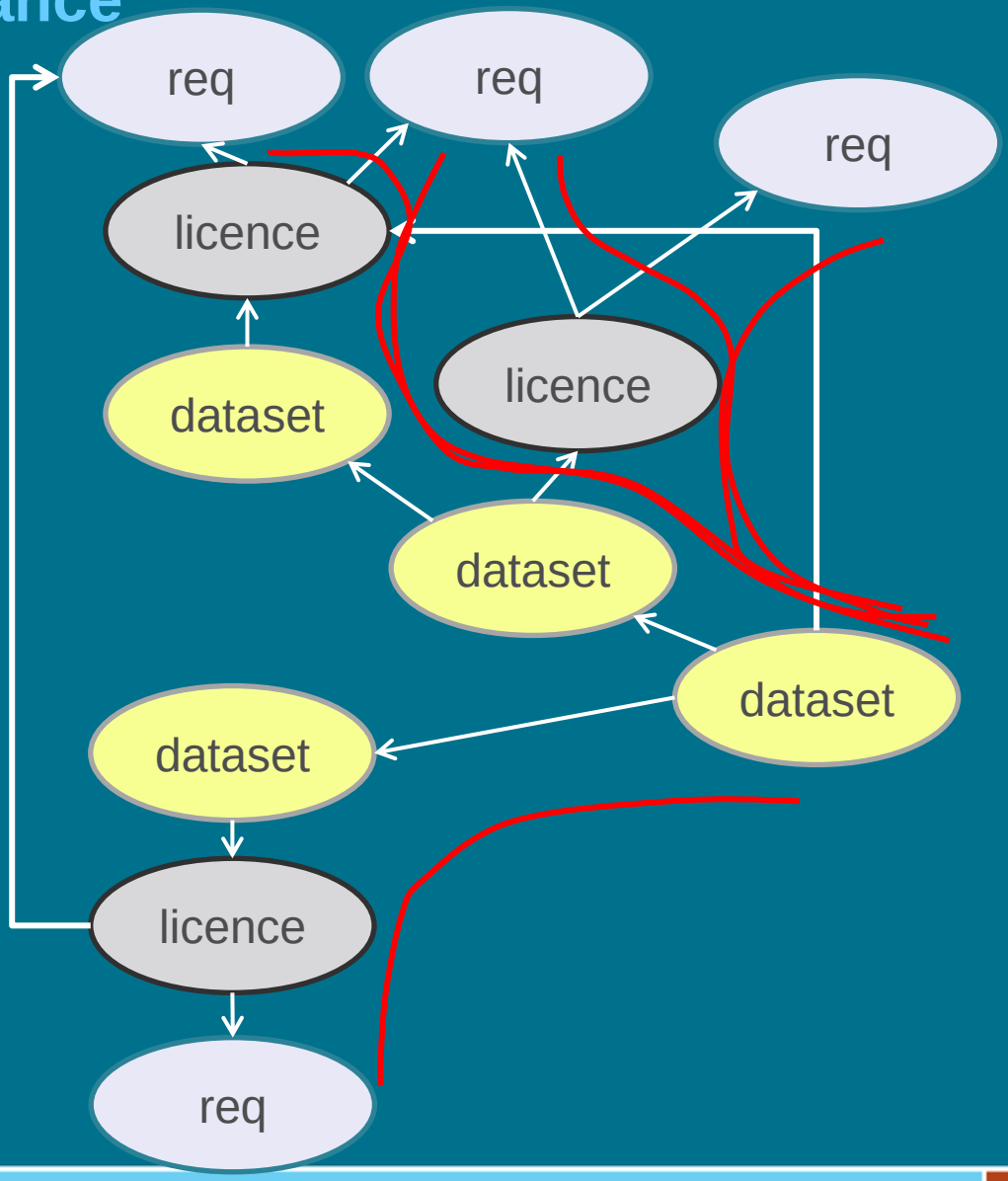
Associate license *requirements as* distinct objects with licenses

Reuse!



Methods - Provenance

Calculate requirements automatically



Faceted License model

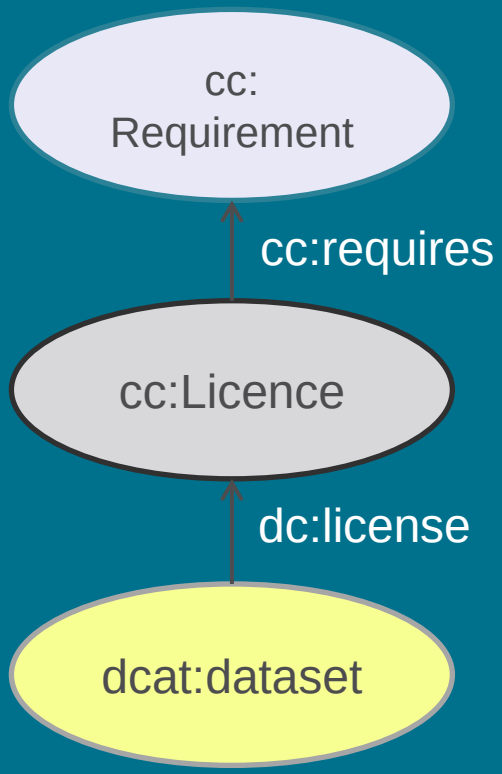
Requirements

The Creative Commons licence information model[1] has a “Requirement” class that a licence can point to

Requirement class is “an action that may or may not be requested of you”

This is a sub-license unit

[1] http://creativecommons.org/ns

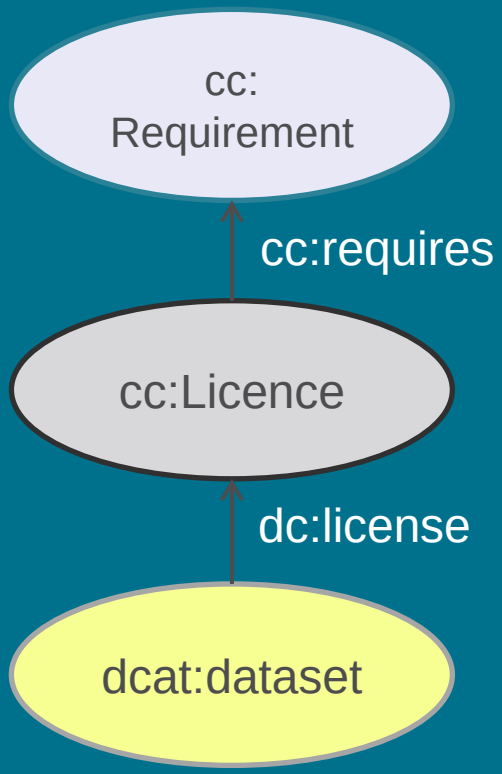


Faceted License model

Requirements

BA defined a series of Requirement instances[1]

[1] http://data.bioregionalassessments.gov.au/id/requirement/



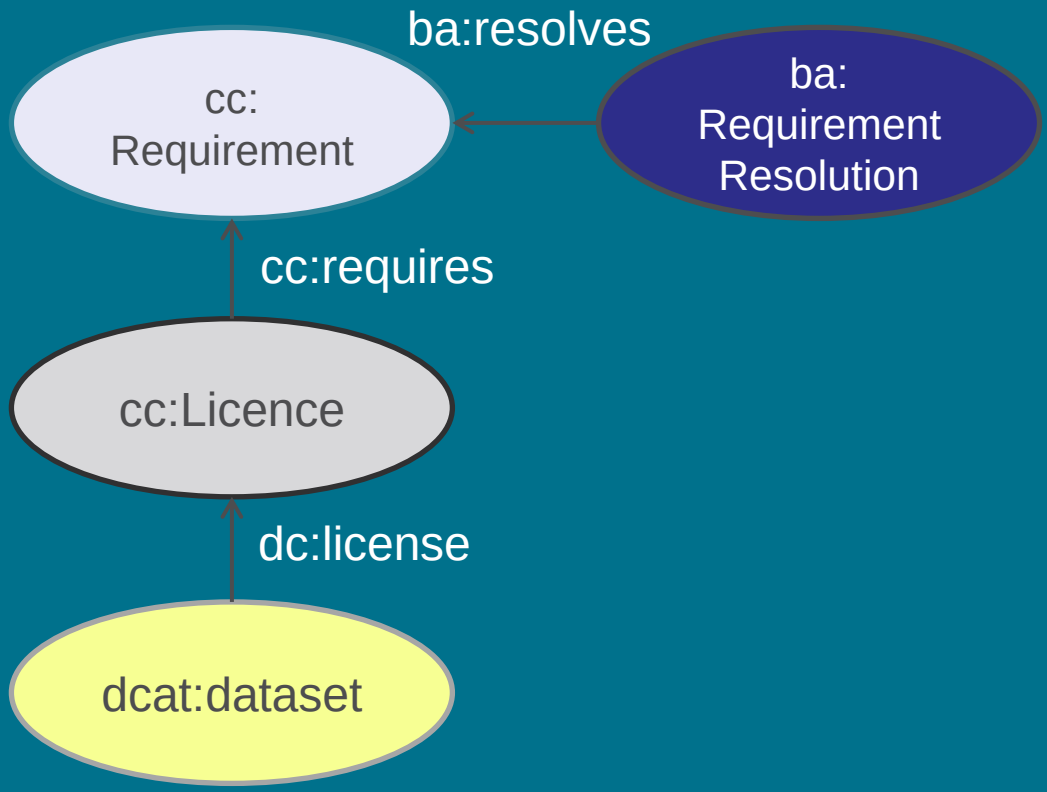
Faceted License model

Requirements Resolution

BA defined a series of Requirement Resolutions

These resolve particular Requirements

RRs are an *Entity*, like a dataset



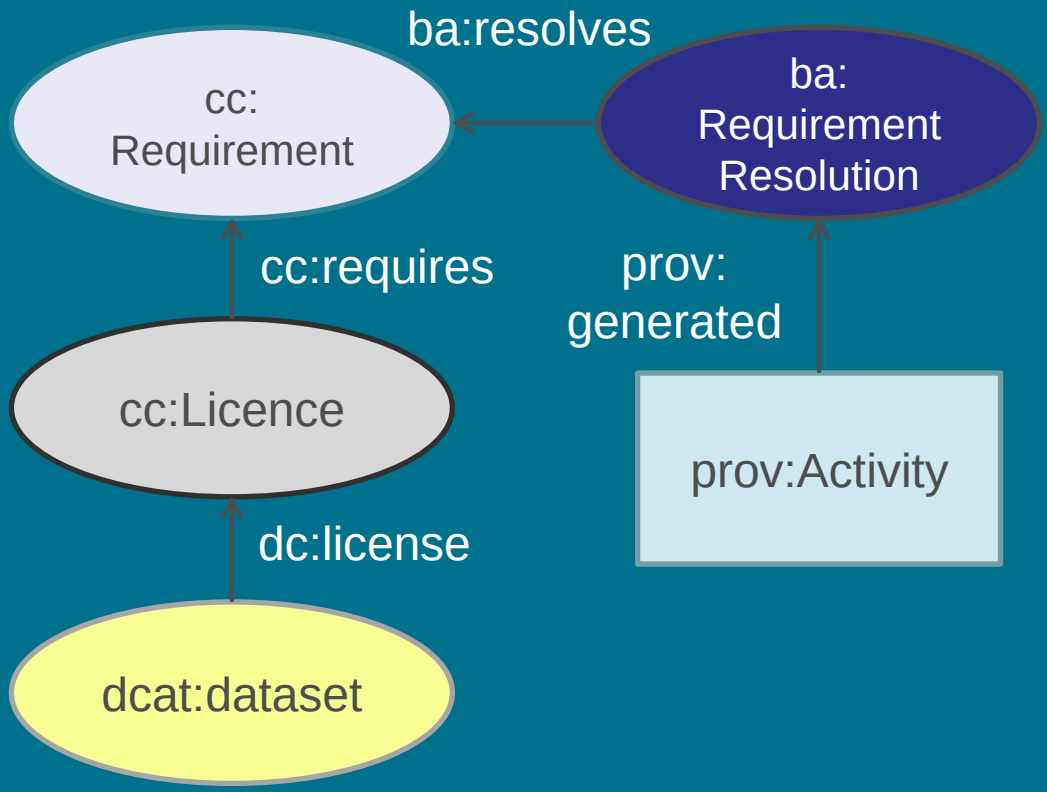
Faceted License model

Generating Requirements Resolutions

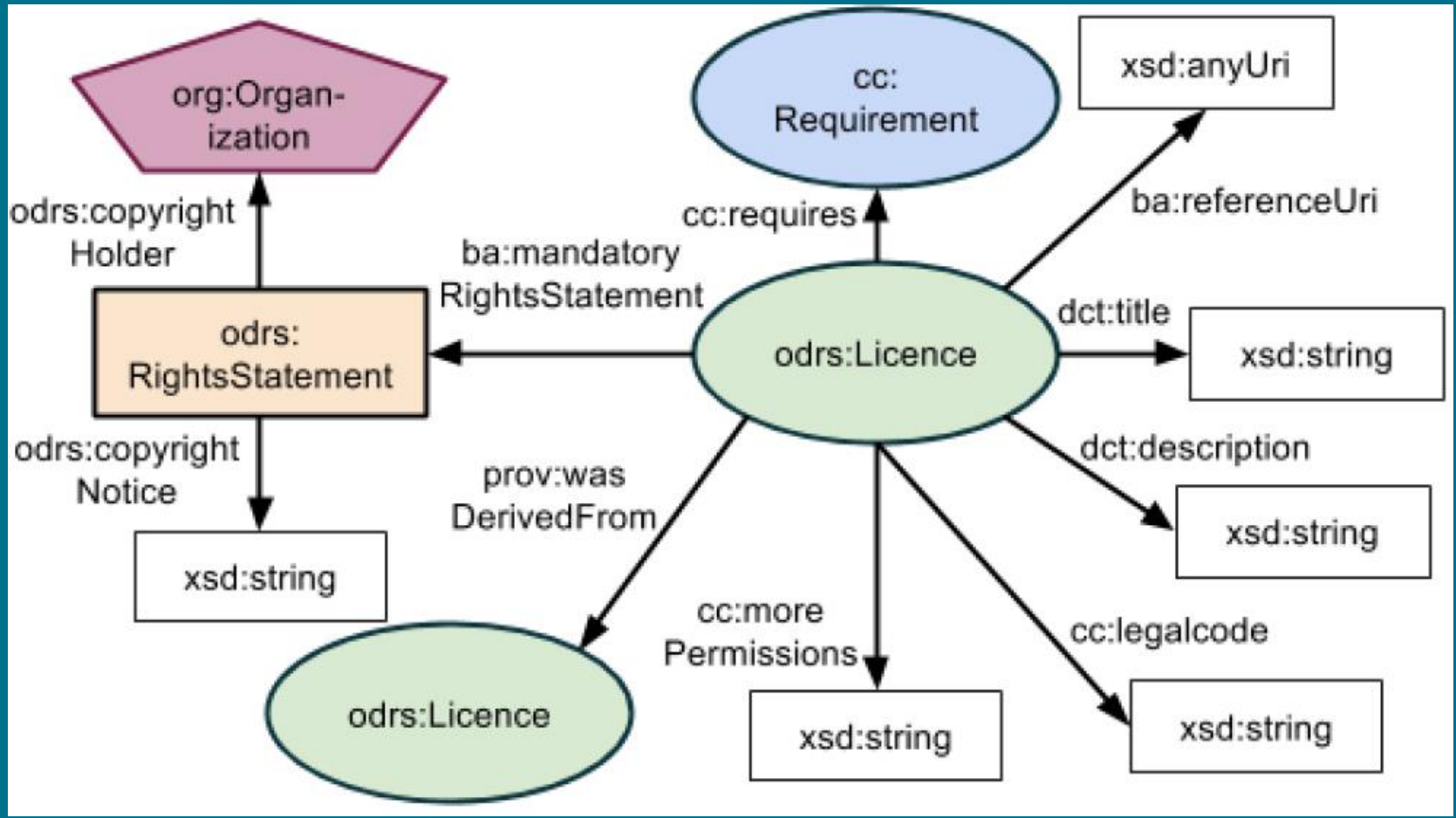
In PROV[1] Entities are generated by Activities

This suits our needs

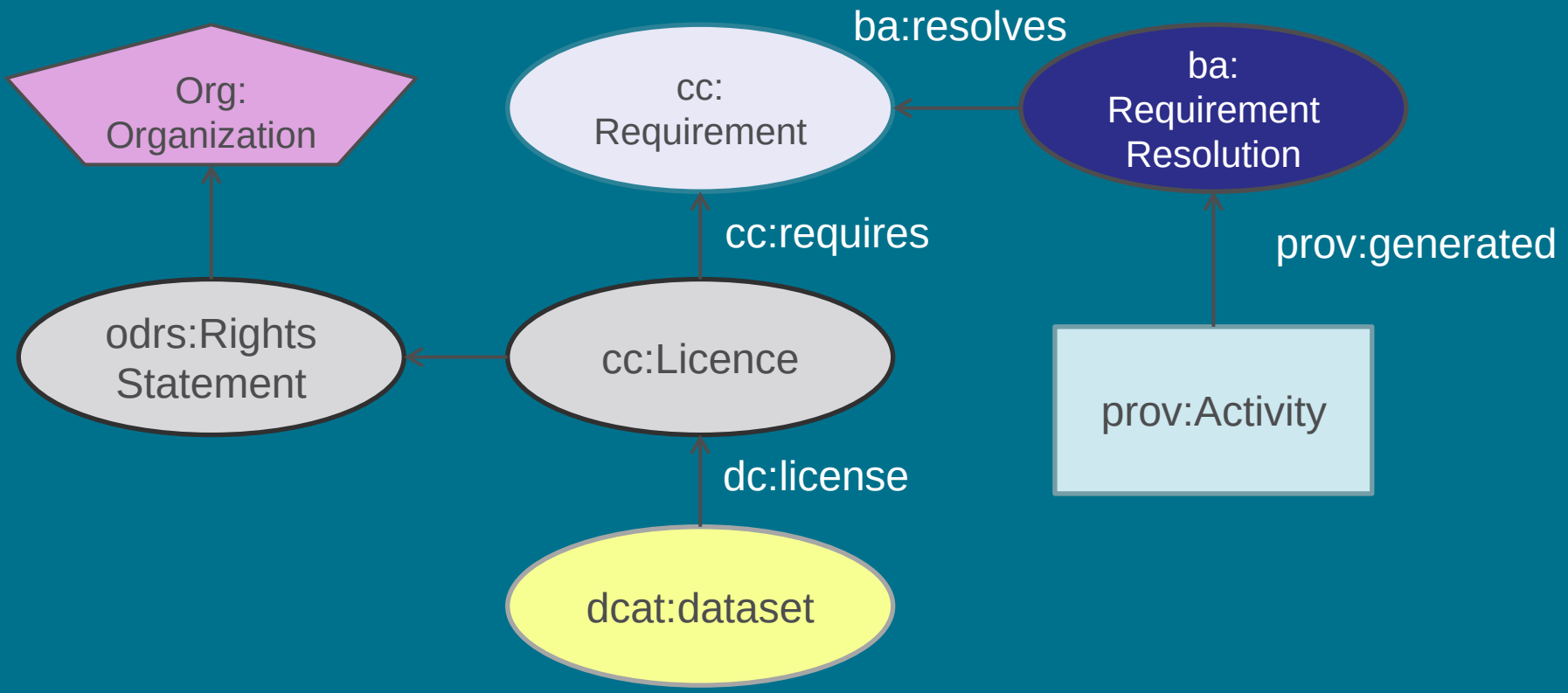
[1] https://www.w3.org/TR/prov-dm/



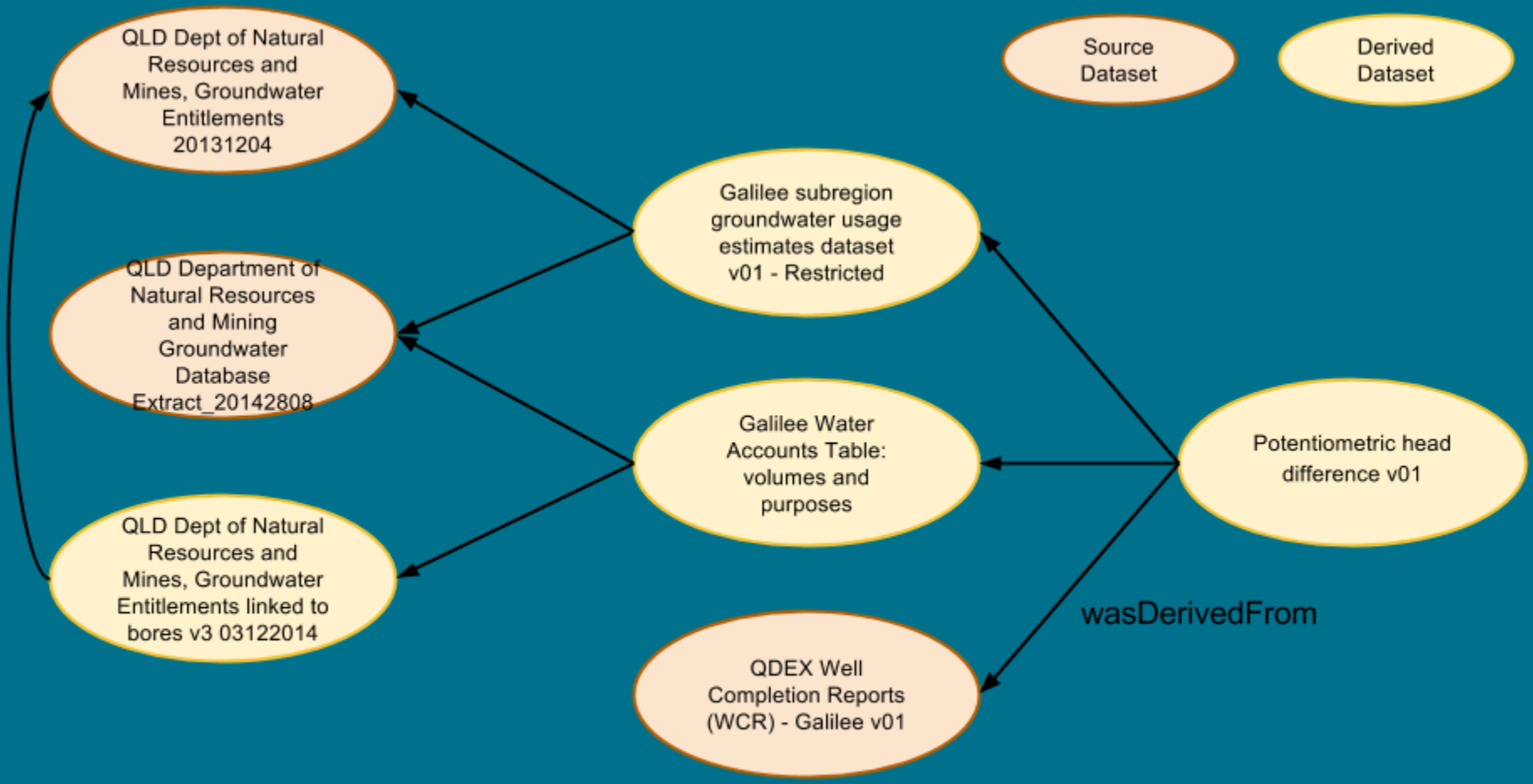
Faceted license model



Faceted license model

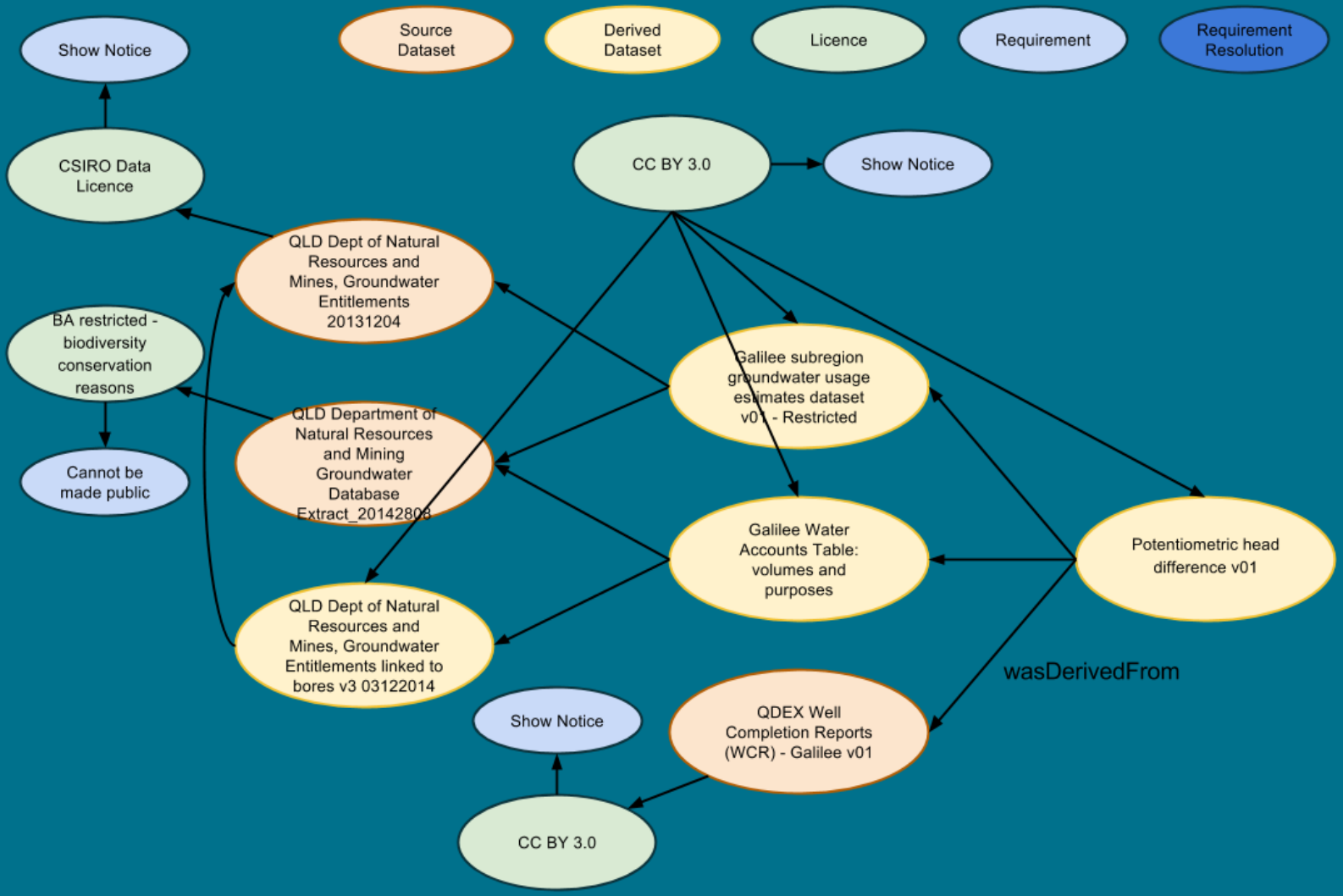


A real BA example



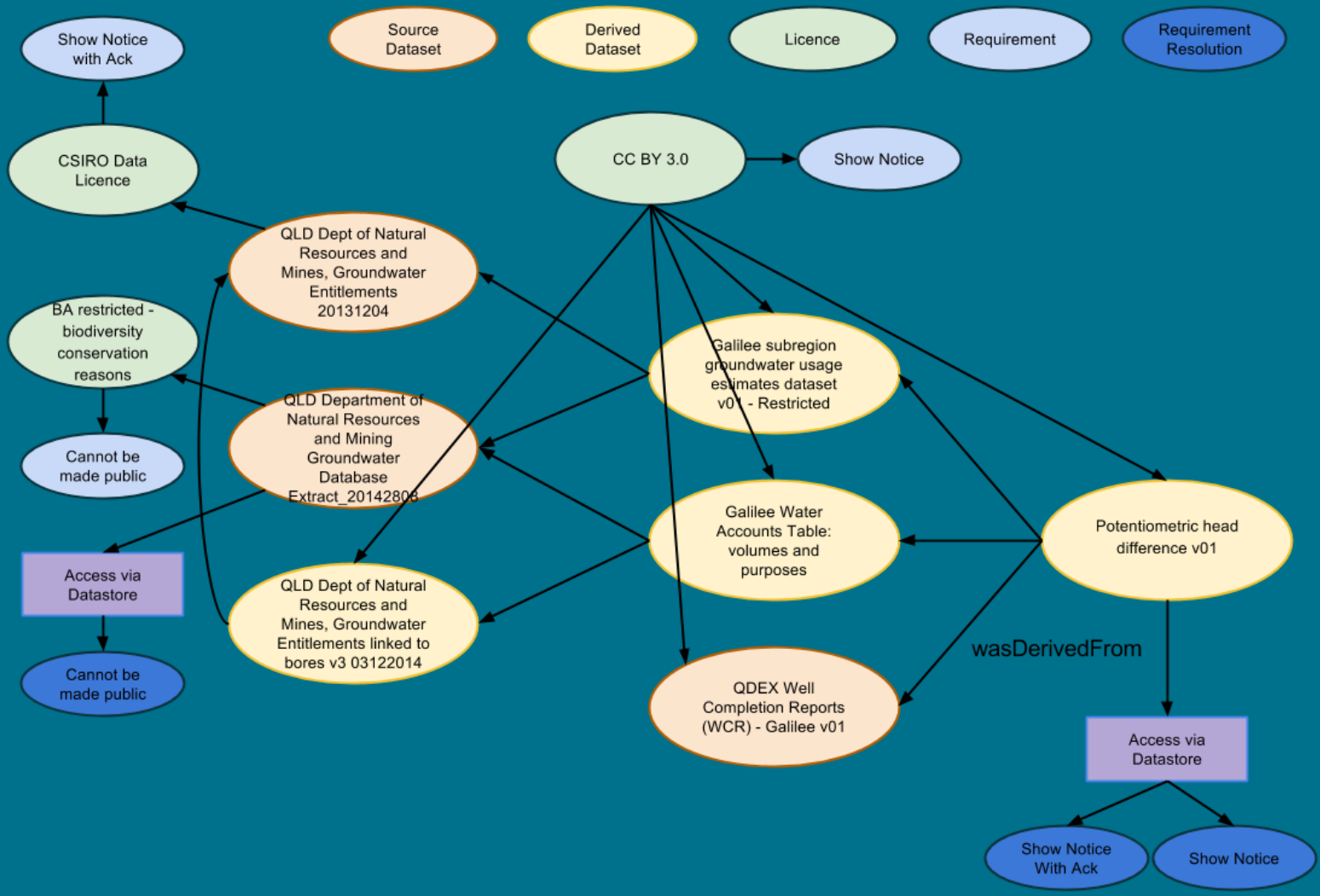
From: http://data.bioregionalassessments.gov.au/function/metadataexporter/6c212a1a-658c-41de-92a7-6054349a848b?_view=provenance

A real BA example



From: http://data.bioregionalassessments.gov.au/function/metadataexporter/6c212a1a-658c-41de-92a7-6054349a848b?_view=provenance

A real BA example



From: http://data.bioregionalassessments.gov.au/function/metadataexporter/6c212a1a-658c-41de-92a7-6054349a848b?_view=provenance

BA Findings

- Handling Requirements individually and citations separately reduces licences from 100s (later 60+) to about 10
- There are very few distinct Requirements
  - Perhaps 10
- Most Requirements can be resolved by data store actions
  - Show notices
  - Limit access
- Some Requirements can be resolved only by one-time deliberate action
  - Removing/lumping/de-identifying/averaging
- Resolution status of all Requirements for all datasets can be calculated easily using a provenance graph + metadata

Ongoing Work

- Data Centre/Agency-based and ongoing
  - GA is applying this approach “forever”
- Implementing registers for multiple uses
  - Licenses
  - Requirements
  - Organizations
  - Requirement Resolutions
  - Activities that generate RRs
  - Systems that perform the Activities

Future Work

- Further normalise Requirements to cater for identical actions, different data (e.g. attribution)
- Make a project-independent licencing data model
  - Ontology
  - Extension to PROV & CC
- Publish real Requirement & RequirementResolution objects
  - As demonstrators
  - For others to use directly (e.g. “show notice”)
- Have this reviewed by lawyers

- Modelling other forms of Agreement: see “Agreeing about Agreements” in the session “Getting the incentives right”