Slots in Translation

Get some proper explanation of this document.

Slots not appearing here are "compound slots", that are used to group other slots. E.g. `OpenProps` groups `Restrictions`, `CCLicenseCompliance`, and `OpenFlags`. As such, these slots do not have a value of their own, and thus we do not need to discuss them here.

Slots

Result Slots

Used for communicating results outside of the model.

DataRepoCompliance	Root slot, grouping all other slots together. Not used in model.
Open	Result slot
FAIR	Result slot
Citable	Result slot
Trustworthy	Result slot
Findable	Result slot
Accessible	Result slot
Interoperable	Result slot
Reusable	Result slot

Open

Slots affecting the `Open` slot value.

NOTE	All conditions pertaining to the `Open` slot are conditioned in that the `CCLicenceCompliance` is above `not`.

OpenFlags
- openFormat ⇒ Open in `partially..fully`
- platformSupportsDataWork ⇒ Open in `partially..fully`
- ccLicenseOK ⇒ Open in `partially..fully`
- restrictionsNotJustified ⇒ Result slot

Restrictions
- none ⇒ Open in `partially..fully`
- minimal ⇒ Open is `partially`
- significant ⇒ Open in `not..partially open`

CCLicenseCompliance
- nonCompliant ⇒ Open in `not..partially`
- none ⇒ Open in `not..partially`
- adequate ⇒ Open in `partially..fully`
- good ⇒ Open in `partially..fully`
- full ⇒ Open in `partially..fully`

Accessible

MetadataPersistence
- no ⇒ Accessible in `not..partially`
- by evidence ⇒ Accessible in `partially..fully`
- by stated policy ⇒ Accessible in `partially..fully`

AccessibleFlags
- humanAccessible ⇒ Accessible in `partially..fully`
- machineAccessible ⇒ Accessible in `partially..fully`
- persistentMetadata ⇒ Accessible in `partially..fully`
- licenseOK ⇒ Accessible in `partially..fully`
- stdApi ⇒ Accessible in `partially..fully`

Findable

FindableFlags
- internalSearchOK ⇒ Findable in `partially..fully`

PersistentIdentifier
- none ⇒ Findable in `not..partially`
- internalPID ⇒ Findable is `partially`
- externalPID ⇒ Findable in `partially..fully`

MetadataGrade

- minimal ⇒ Findable in `not..partially`
- limited ⇒ Findable is `partially`
- rich ⇒ Findable in `partially..fully`

IdInMetadata

- none ⇒ Findable in `not..partially`
- partial ⇒ Findable is `partially`
- all ⇒ Findable in `partially..fully`

Interoperable

InteroperableFlags

- formalMetadataVocabularyOK ⇒ Interoperable in `partially..fully`
- fairMetadataOK ⇒ Interoperable in `partially..fully`
- qualifiedMetadataReferencesOK ⇒ Interoperable in `partially..fully`
- studyLinkageOK ⇒ Interoperable in `partially..fully`

MetadataFAIRness

Finer-grained `fairMetadataOK`.

- minimal ⇒ Interoperable in `not..partially`
- allowed ⇒ Interoperable in `partially..fully`
- enforced ⇒ Interoperable in `partially..fully`

MetadataReferenceQuality

Finer-grained `qualifiedMetadataReferencesOK`.

- freeText ⇒ Interoperable in `not..partially`
- informal ⇒ Interoperable in `partially..fully`
- formal ⇒ Interoperable in `partially..fully`

StudyLinkage

Finer-grained `studyLinkageOK`

- none ⇒ Interoperable in `not..partially`
- freeText ⇒ Interoperable in `not..partially`
- textualMetadata ⇒ Interoperable in `partially..fully`
- machineReadableMetadata ⇒ Interoperable in `partially..fully`

Reusable

ReusableFlags
- documentationOK ⇒ Reusable in `partially..fully`
- dkNetMetadataOK ⇒ Reusable in `partially..fully`
- communityStandard ⇒ Reusable in `partially..fully`
- generalMetadata ⇒ Reusable in `partially..fully`
- metadataProvenanceOK ⇒ Reusable in `partially..fully`

DocumentationLevel   Finer-grained `documentationOK`
- lacking ⇒ Reusable in `not..partially`
- adequate ⇒ Reusable in `partially..fully`
- good ⇒ Reusable in `partially..fully`
- full ⇒ Reusable in `partially..fully`

ReuseLicense   Finer-grained `licenseOK`
- none ⇒ Reusable in `not..partially`
- repositoryLevel ⇒ Reusable in `partially..fully`
- datasetLevel ⇒ Reusable in `partially..fully`

MetadataProvenance   Finer-grained `metadataProvenanceOK`.
- unclear ⇒ Reusable in `not..partially`
- adequate ⇒ Reusable in `partially..fully`
- full ⇒ Reusable in `partially..fully`

DkNetMetadataLevel   Finer-grained `dkNetMetadataOK`
- none ⇒ Reusable in `not..partially`
- dataset ⇒ Reusable in `partially..fully`
- datasetAndSubject ⇒ Reusable in `partially..fully`

Citable

OrcidAssociation
- none ⇒ Citable in `not..partially`
- supported ⇒ Citable in `partially..fully`
- required ⇒ Citable in `partially..fully`

CitationMetadataLevel

- none ⇒ Citable in `not..partially`
- partial ⇒ Citable is `partially`
- full ⇒ Citable in `partially..fully`

MachineReadableLandingPage

- none ⇒ Citable in `not..partially`
- exists ⇒ Citable is `partially`
- supportsDataCitation ⇒ Citable in `partially..fully`

Trustworthiness

GovernanceTransparency

- opaque ⇒ Trustworthy in `significantConcerns..minorConcerns`
- partial ⇒ Trustworthy is `minorConcerns`
- full ⇒ Trustworthy in `minorConcerns..noConcerns`

StakeholderGovernance

- none ⇒ Trustworthy in `significantConcerns..minorConcerns`
- weak ⇒ Trustworthy is `minorConcerns`
- good ⇒ Trustworthy is `minorConcerns`
- full ⇒ Trustworthy in `minorConcerns..noConcerns`

SourceOpen

- no ⇒ Trustworthy in `significantConcerns..minorConcerns`
- partially ⇒ Trustworthy is `minorConcerns`
- yes ⇒ Trustworthy in `minorConcerns..noConcerns`