Research Higher Degree Logical Model

Prepared by:  
Thursdays **Group 4**

Sam Deane dean0109

Andrew Zschorn zsch0003

Version 0.1-DRAFT

4/5/2014

Created as part of the requirements for Advanced Database GE 2014

Contents

[1. Derive relations 1](#_Toc387914333)

[1.1. Conceptual E-R diagram 1](#_Toc387914334)

[1.2. Strong Entity types 2](#_Toc387914335)

[1.3. Weak Entity types 2](#_Toc387914336)

[1.4. One-to-many binary relationships 3](#_Toc387914337)

[1.5. One-to-one binary relationships 3](#_Toc387914338)

[1.5.1. Mandatory on both sides 3](#_Toc387914339)

[1.5.2. Mandatory on one side 4](#_Toc387914340)

[1.5.3. Optional on both sides 4](#_Toc387914341)

[1.6. One-to-one recursive relationships 4](#_Toc387914342)

[1.7. Superclass/subclass relationships 4](#_Toc387914343)

[1.8. Many-to-many binary relationship types 4](#_Toc387914344)

[1.8.1. University Staff Member *works in* Research Area 4](#_Toc387914345)

[1.8.2. University Staff Member *oversees* Research Area 5](#_Toc387914346)

[1.8.3. Application *in* Research Area 5](#_Toc387914347)

[1.8.4. University Staff Member *flags* Application 6](#_Toc387914348)

[1.9. Complex relationship types 6](#_Toc387914349)

[1.10. Multi-valued attributes 6](#_Toc387914350)

[1.11. Document relations and foreign key attributes 6](#_Toc387914351)

[2. Normalisation 7](#_Toc387914352)

[2.1. Lookup relations 9](#_Toc387914353)

[2.1.1. Lookup relationships 10](#_Toc387914354)

[2.2. Normalised Relations 11](#_Toc387914355)

[2.3. Logical E-R Diagram 13](#_Toc387914356)

[3. User transaction validation 14](#_Toc387914357)

[3.1. Transaction pathways 15](#_Toc387914358)

[4. Check integrity constraints 16](#_Toc387914359)

[4.1. Required data 16](#_Toc387914360)

[4.2. Attribute domain constraints 16](#_Toc387914361)

[4.3. Multiplicity 16](#_Toc387914362)

[4.4. Entity integrity 17](#_Toc387914363)

[4.5. Referential integrity 17](#_Toc387914364)

[4.5.1. Null attributes (~insert rules) 17](#_Toc387914365)

[4.5.2. Updates & Deletes 17](#_Toc387914366)

[4.6. General constraints. 17](#_Toc387914367)

[4.7. Document all integrity constraints 17](#_Toc387914368)

[5. Review logical data model 18](#_Toc387914369)

[6. Check for future growth 18](#_Toc387914370)

[7. Develop Test Plan 18](#_Toc387914371)

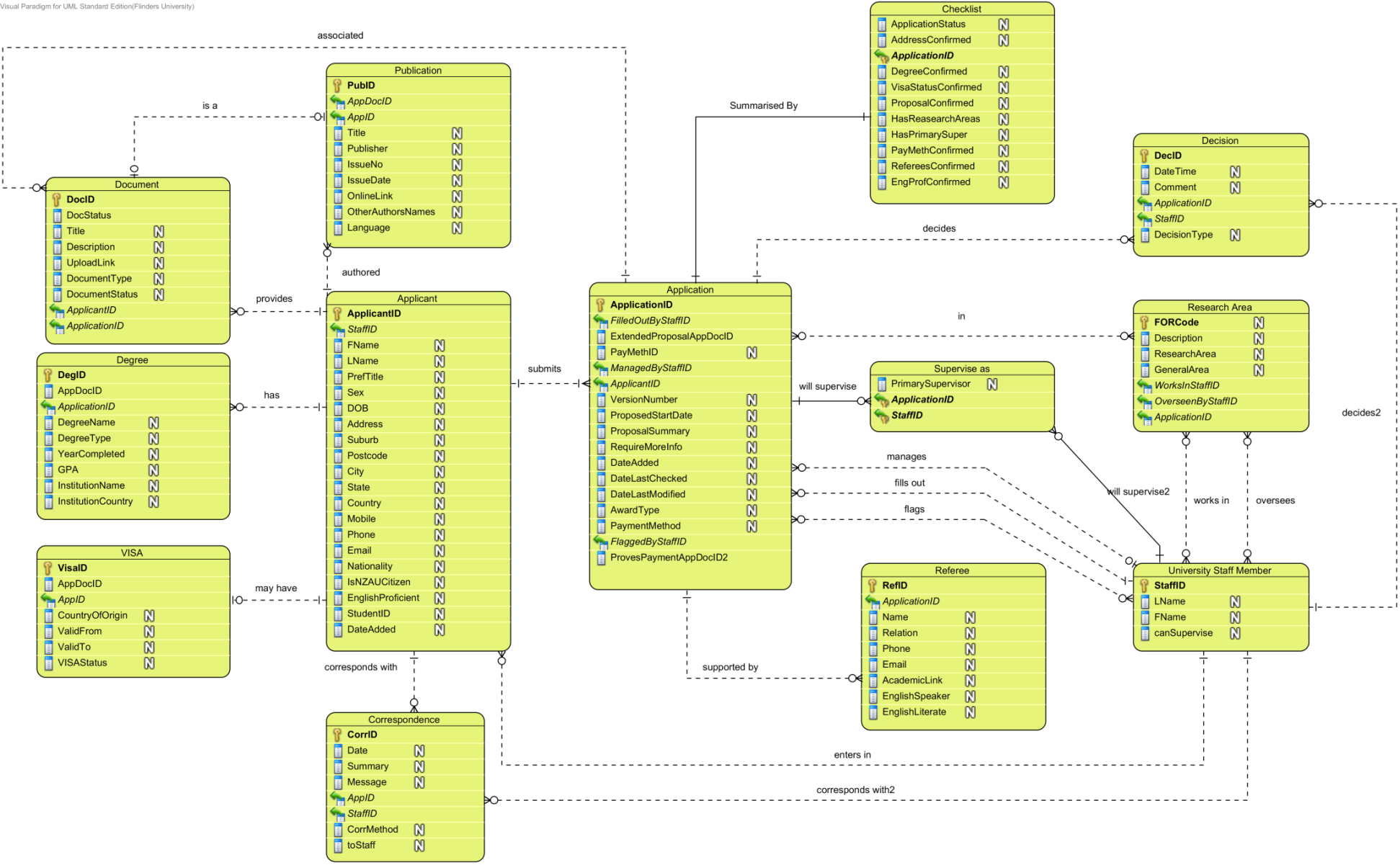
[8. Data dictionary 18](#_Toc387914372)

[9. Conceptual E-R Diagram 18](#_Toc387914373)

[10. Logical E-R Diagram 18](#_Toc387914374)

# Derive relations

## Conceptual E-R diagram



## Strong Entity types

|  |
| --- |
| **Applicant** (ApplicantID, FName, LName, PrefTitle, Sex, DOB, StreetAddress, Suburb, Postcode, City, State, Country, Mobile, Phone, Email, Nationality, isNZAUCitizen, EnglishProficient, StudentID, DateAdded)  **Primary key** ApplicantID  **Alternate key** (FName, LName, DOB, StreetAddress, Suburb, Postcode, City, State, Country) |
| **Application** (ApplicationID, ProposedStartDate, ProposalSummary, RequireMoreInfo, DateAdded, DateLastChecked, DateLastModified, AwardType, flindersCampus, fullTime, PaymentMethod)  **Primary key** ApplicationID |
| **Correspondence** (CorrID, Date, Summary, Message, CorrMeth, toStaff)  **Primary key** CorrID |
| **Decision** (DecID, DateTime, Comment, DecisionType)  **Primary key** DecID |
| **Degree** (DegID, DegreeName, DegreeType, YearCompleted, GPA, InstitutionName, InstituitonCountry)  **Primary key** DegID |
| **Document** (DocID, UploadLink, DocStatus, DocumentType, Title, Description)  **Primary key** UploadLink |
| **Publication** (PubID, ApplicantID, Title, Publisher, IssueNo, IssueDate, OnlineLink, OtherAuthorsNames, Language)  **Primary key** PubID |
| **Referee** (RefID, Name, Relation, Phone, Email, AcademicLink, EnglishSpeaker, EnglishLiterate)  **Primary key** RefID |
| **Research Area** (FORCode, Description, ResearchArea, GeneralArea)  **Primary key** FORCode |
| **University Staff Member** (StaffID, FName, LName, canSupervise)  **Primary key** StaffID |
| **Visa** (VisaID, VISAStatus, CountryOfOrigin, ValidFrom, ValidTo)  **Primary key** VisaID |

## Weak Entity types

|  |
| --- |
| **Checklist** (ApplicationStatus, AddressConfirmed, DegreeConfirmed, VisaStatusConfirmed, ProposalConfirmed, HasResearchArea, HasPrimarySuper, PayMethConfirmed, RefrereesConfirmed, EngProfConfirmed)  **Primary key** None (at this stage) |
| **Supervise as** (PrimarySupervisor)\*  **Primary key** None (at this stage) |

\* This entity was created during the conceptual phase to remove a relationship attribute

## One-to-many binary relationships

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Relationship name** | **Identifying** | **Parent relation** | **Parent multiplicity** | **Foreign Keys** | **Child relation** | **Child multiplicity** |
| associated | No | Application | 0..1 | ApplicationID | Document | 0..\* |
| authored | No | Applicant | 1..1 | ApplicantID | Publication | 0..\* |
| corresponds with | No | Application | 1..1 | ApplicationID | Correspondence | 0..\* |
| corresponds with2 | No | University Staff Member | 1..1 | StaffID | Correspondence | 0..\* |
| decides | No | Application | 1..1 | ApplicationID | Decision | 0..\* |
| decides2 | No | University Staff Member | 1..1 | StaffID | Decision | 0..\* |
| has | Yes | Applicant | 1..1 | ApplicantID | Degree | 0..\* |
| last to modify |  | University Staff Member | 1..1 | StaffID → LastToModifyStaffID | Applicant | 0..\* |
| last to update | No | University Staff Member | 1..1 | StaffID →LastToUpdateStaffID | Application | 0..\* |
| manages | No | University Staff Member | 0..1 | StaffID → ManagedByStaffID | Application | 0..\* |
| provides | No | Applicant | 1..1 | ApplicantID | Document | 0..\* |
| submits | Yes | Applicant | 1..1 | ApplicantID | Application | 0..\* |
| supported by | No | Application | 1..1 | ApplicationID | Referee | 1 |
| will supervise | Yes | Application | 1..1 | ApplicationID | Supervise as | 0..\* |
| will supervise2 | Yes | University Staff Member | 1..1 | StaffID | Supervise as | 0..\* |

## One-to-one binary relationships

### Mandatory on both sides

For the **Application *summarised by* Checklist** relationship, we bring all the attributes of **Checklist** into **Application**. The primary key remains ApplicationID; **Checklist** was a weak entity and never had a primary key. The **Checklist** relation is dropped.

|  |
| --- |
| **Application** (ApplicationID, ApplicationStatus, AddressConfirmed, DegreeConfirmed, VisaStatusConfirmed, ProposalConfirmed, HasResearchAreas, HasPrimarySuper, PayMethConfirmed, EngProfConfirmed, RefereesConfirmed, LastToUpdateStaffID, ProposedStartDate, ProposalSummary, RequireMoreInfo, DateAdded, DateLastChecked, DateLastModified, AwardType, PaymentMethod, flindersCampus, fullTime)  **Primary key** ApplicationID |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Relationship name** | **Identifying** | **Parent relation** | **Parent multiplicity** | **Foreign Keys** | **Child relation** | **Child multiplicity** |
| Summarised by | Yes | Application | 1 | ApplicationID | Checklist | 1 |

### Mandatory on one side

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Relationship name** | **Identifying** | **Parent relation** | **Parent multiplicity** | **Foreign Keys** | **Child relation** | **Child multiplicity** |
| may have | No | Applicant | 1 | ApplicantID | VISA | 0..1 |

### Optional on both sides

None in this case.

## One-to-one recursive relationships

None in this case.

## Superclass/subclass relationships

None in this case.

## Many-to-many binary relationship types

### University Staff Member *works in* Research Area

Introduce the following relation:

|  |
| --- |
| **University Staff Member\_Research Area** (StaffID, FORCode)  **Primary key** StaffID, FORCode  **Foreign key** StaffID **references** University Staff Member(StaffID)  **Foreign key** FORCode **references** Research Area(FORCode) |

Introduce the following relationships:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Relationship name** | **Identifying** | **Parent relation** | **Parent multiplicity** | **Foreign Keys** | **Child relation** | **Child multiplicity** |
| works in | Yes | University Staff Member | 1 | StaffID | University Staff Member\_Research Area | 0..\* |
| works in2 |  | Research Area | 1 | FORCode | University Staff Member\_Research Area | 0..\* |

### University Staff Member *oversees* Research Area

Introduce the following relation:

|  |
| --- |
| **University Staff Member\_Research Area2** (StaffID, FORCode)  **Primary key** StaffID, FORCode  **Foreign key** StaffID **references** University Staff Member(StaffID)  **Foreign key** FORCode **references** Research Area(FORCode) |

Introduce the following relationships:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Relationship name** | **Identifying** | **Parent relation** | **Parent multiplicity** | **Foreign Keys** | **Child relation** | **Child multiplicity** |
| Oversees | Yes | University Staff Member | 1 | StaffID | University Staff Member\_Research Area2 | 0..\* |
| Oversees2 | Yes | Research Area | 1 | FORCode | University Staff Member\_Research Area2 | 0..\* |

### Application *in* Research Area

Introduce the following relation:

|  |
| --- |
| **Application\_Research Area** (ApplicationID, FORCode)  **Primary key** ApplicationID, FORCode  **Foreign key** ApplicationID **references** Application(ApplicationID)  **Foreign key** FORCode **references** Research Area(FORCode) |

Introduce the following relationships:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Relationship name** | **Identifying** | **Parent relation** | **Parent multiplicity** | **Foreign Keys** | **Child relation** | **Child multiplicity** |
| in |  | Application | 1 | ApplicationID | Application\_Research Area | 1..\* |
| in2 |  | Research Area | 1 | FORCode | Application\_Research Area | 0..\* |

### University Staff Member *flags* Application

Introduce the following relation:

|  |
| --- |
| **University Staff Member\_Application** (StaffID, ApplicationID)  **Primary key** StaffID, ApplicationID  **Foreign key** StaffID **references** University Staff Member(StaffID)  **Foreign key** ApplicationID **references** Application(ApplicationID) |

Introduce the following relationships:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Relationship name** | **Identifying** | **Parent relation** | **Parent multiplicity** | **Foreign Keys** | **Child relation** | **Child multiplicity** |
| flags | Yes | University Staff Member | 1 | StaffID | University Staff Member\_Application | 0..\* |
| flags2 | Yes | Application | 1 | ApplicationID | University Staff Member\_Application | 0..\* |

## Complex relationship types

None in this case

## Multi-valued attributes

None in this case

## Document relations and foreign key attributes

Since the relationships ‘will supervise’ and ‘will supervise 2’ have been defined in section 1.3, foreign keys are introduced into the weak entity ‘supervise as’ such that

|  |
| --- |
| **Supervise as** (PrimarySupervisor, StaffID, ApplicationID)  **Primary key** StaffID, ApplicationID  **Foreign key** StaffID **references** University Staff Member(StaffID)  **Foreign key** ApplicationID **references** Application(ApplicationID) |

To save space, a full list of intermediate relationships are not listed here, instead please see section 2.2 for a full list of normalised relations.

# Normalisation

|  |  |  |
| --- | --- | --- |
| **Relation** | **Functional Dependencies** | **Remarks** |
| Applicant | ApplicantID → FName, LName, PrefTitle, Sex, DOB, StreetAddress, Suburb, Postcode, City, State, Country, Mobile, Phone, Email, Nationality, isNZAUCitizen, EnglishProficient, StudentID, DateAdded, LastToModifyStaffID  FName, LName, DOB, StreetAddress, Suburb, Postcode, City, State, Country → ApplicantID, PrefTitle, Mobile, Phone, Email, Nationality, isNZAUCitizen, EnglishProficient, StudentID, DateAdded, LastToModifyStaffID  FName, LName, DOB, Email → ApplicantID, PrefTitle, StreetAddress, Suburb, Postcode, City, State, Country Mobile, Phone, Nationality, isNZAUCitizen, EnglishProficient, StudentID, DateAdded, LastToModifyStaffID | No repeating groups.  No partial dependencies.  No transitive dependencies.  Primary key fully determines non-primary-key attributes  ⇒ in 3NF |
| Application | ApplicationID → ApplicantID, AddressConfirmed, DegreeConfirmed, VisaStatusConfirmed, ProposalConfirmed, HasResearchAreas, HasPrimarySuper, PayMethConfirmed, EngProfConfirmed, RefereesConfirmed, LastToUpdateStaffID, ProposalDocID, AwardType, ManagedByStaffID, ProposedStartDate, ProposedSummary, RequireMoreInfo, DateAdded, DateLastChecked, DateLastModified, PayDocID, PaymentMethod, ApplicationStatus, flindersCampus, fullTime | No repeating groups.  No partial dependencies.  No transitive dependencies.  Primary key fully determines non-primary-key attributes  ⇒ in 3NF |
| Correspondence | CorrID → Date, Summary, Message, ApplicationID, StaffID, CorrMethID, toStaff | No repeating groups.  No partial dependencies.  No transitive dependencies.  Primary key fully determines non-primary-key attributes  ⇒ in 3NF |
| Decision | DecID → Date, Comment, ApplicationID, StaffID | No repeating groups.  No partial dependencies.  No transitive dependencies.  Primary key fully determines non-primary-key attributes  ⇒ in 3NF |
| Degree | DegID →ApplicantID, Name, Type, YearCompleted, GPA, InstitutionName, InstituitonCountry  ApplicantID, Name → DegID, Type, YearCompleted, GPA, InstitutionName, InstituitonCountry | No repeating groups.  No partial dependencies.  No transitive dependencies.  Primary key fully determines non-primary-key attributes  ⇒ in 3NF |
| Document | DocID → UploadLink, DocStatus, DocumentType, Title, Description, ApplicantID, ApplicationID  UploadLink → DocID, DocStatus, DocumentType, Title, Description, ApplicantID, ApplicationID | No repeating groups.  No partial dependencies.  No transitive dependencies.  Primary key fully determines non-primary-key attributes  ⇒ in 3NF |
| Publication | PubID → ApplicantID, DocID, Title, Publisher, IssueNo, IssueDate, OnlineLink, OtherAuthorsNames, Language | Attributes Publication, IssueNo, IssueDate, OnlineLink and OtherAuthorsNames are optional, therefore should not be considered as candidate keys.  No repeating groups.  No partial dependencies.  No transitive dependencies.  Primary key fully determines non-primary-key attributes  ⇒ in 3NF |
| Referee | RefID → ApplicationID, Name, Relation, Phone, Email, Profession, AcademicLink, EnglishSpeaker, EnglishLiterate | No repeating groups.  No partial dependencies.  No transitive dependencies.  Primary key fully determines non-primary-key attributes  ⇒ in 3NF |
| ResearchArea | FORCode → Description, ResearchArea, GeneralArea | No repeating groups.  No partial dependencies.  No transitive dependencies.  Primary key fully determines non-primary-key attributes  ⇒ in 3NF |
| University Staff Member | StaffID → LName, FName, canSupervise | No repeating groups.  No partial dependencies.  No transitive dependencies.  Primary key fully determines non-primary-key attributes  ⇒ in 3NF |
| Visa | VisaID → ApplicantID, VisaStatus, OriginCountry, ValidFrom, ValidTo | No repeating groups.  No partial dependencies.  No transitive dependencies.  Primary key fully determines non-primary-key attributes  ⇒ in 3NF |

## Lookup relations[[1]](#footnote-1)

In this schema there are several attributes with domains that allow only a small set of possible values. These attributes are Applicant(Nationality), Applicant(Country), Application(ApplicationStatus), Application(PaymentMethod), Document(DocumentStatus), Document(DocumentType), Visa(VisaStatus), Decision(DecisionType), Correspondence(CorrespondencMethod), and Award(AwardType).

In order to reduce the chance of update anomalies from keying-in incorrect values, we choose to create a lookup relation for each of these. In all cases, these relations are in one-to-many relationships with the relations containing their corresponding attributes. So, in each case, the lookup relation is designated the parent relation, the other relation is designated the child, and the primary key of the lookup table is posted to the child relation in place of the attribute.

This approach also has the benefit that users can: alter the associated description/types easily; or extend these domains to cover new cases as they arise.

|  |
| --- |
| **ApplicationStatus** (Status, Description)  **Primary key** Status |
| **Decision Type** (Type)  **Primary key** Type |
| **CorrespondenceMethod** (method)  **Primary key** Method |
| **AwardType** (Type, Description, Method)  **Primary key** Type |
| **Payment Method** (Method)  **Primary key** Method |
| **DocumentStatus** (Status, Description)  **Primary key** Status |
| **DocumentType** (Type, Description)  **Primary key** Type |
| **Visa Status** (Status, Description)  **Primary key** Status |
| **Country** (CountryISOCode, Name)  **Primary key** CountryISOCode |

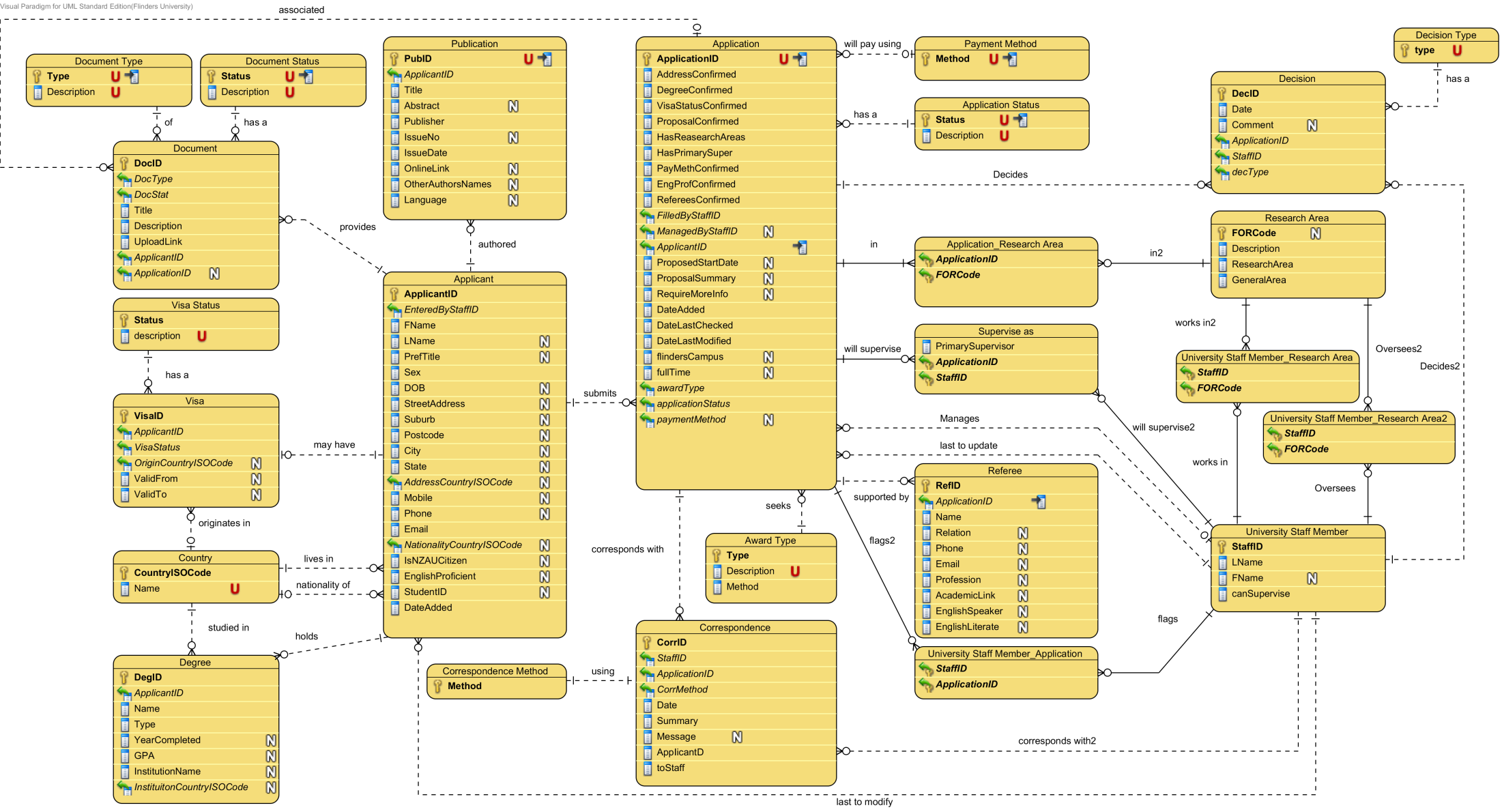
### Lookup relationships[[2]](#footnote-2)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Relationship name** | **Identifying** | **Parent relation** | **Parent multiplicity** | **Foreign Keys** | **Child relation** | **Child multiplicity** |
| lives in | No | Country | 1..1 | CountryISOCode→ AddressCountryISOCode | Applicant | 0..\* |
| nationality of | No | Country | 0..1 | CountryISOCode→ NationalityCountryISOCode | Applicant | 0..\* |
| has a | No | Visa Status | 1..1 | VisaStatus | Visa | 0..\* |
| originates in | No | Country | 1..1 | CountryISOCode → OriginCountryISOCode | Visa | 0..\* |
| has a | No | Document Status | 1..1 | Status → DocStat | Document | 0..\* |
| of | No | Document Type | 1..1 | Type → DocType | Document | 0..\* |
| will pay using | No | Payment Method | 0..1 | Method →paymentMethod | Application | 0..\* |
| has a | No | Application Status | 1..1 | Status → applicationStatus | Application | 0..\* |
| has a | No | Decision Type | 1..1 | type → decType | Decision | 0..\* |

## Normalised Relations

|  |
| --- |
| **Applicant** (ApplicantID, FName, LName, PrefTitle, Sex, DOB, StreetAddress, Suburb, Postcode, City, State, AddressCountryISOCode, Mobile, Phone, Email, NationalityCountryISOCode, isNZAUCitizen, EnglishProficient, StudentID, DateAdded, LastToModifyStaffID)  **Primary key** ApplicantID  **Alternate key** (FName, LName, DOB, StreetAddress, Suburb, Postcode, City, State, Country)  **Alternate key** (FName, LName, DOB, Email)  **Foreign key** LastToModifyStaffID **references** University Staff Member(StaffID)  **Foreign key** AddressCountryISOCode **references** Country(CountryISOCode)  **Foreign key** NationalityCountryISOCode **references** Country(CountryISOCode) |
| **Application** (ApplicationID, applicationStatus, AddressConfirmed, DegreeConfirmed, VisaStatusConfirmed, ProposalConfirmed, HasResearchAreas, HasPrimarySuper, PayMethConfirmed, EngProfConfirmed, RefereesConfirmed, LastToUpdateStaffID, ProposedStartDate, ProposalSummary, RequireMoreInfo, DateAdded, DateLastChecked, DateLastModified, awardType, flindersCampus, fullTime, paymentMethod)  **Primary key** ApplicationID  **Foreign key** ApplicantID **references** Application(ApplicationID)  **Foreign key** LastToUpdateStaffID **references** University Staff Member(StaffID)  **Foreign key** ManagedByStaffID **references** University Staff Member(StaffID)  **Foreign key** awardType **references** AwardType(Type)  **Foreign key** applicationStatus **references** Application Status(Status)  **Foreign key** paymentMethod **references** Payment Method(Method) |
| **Application\_Research Area** (ApplicationID, FORCode)  **Primary key** ApplicationID, FORCode  **Foreign key** ApplicantID **references** Application(ApplicationID)  **Foreign key** FORCode **references** Research Area(FORCode) |
| **Application Status** (Status, Description)  **Primary key** Status |
| **AwardType** (Type, Description, Method)  **Primary key** Type |
| **Correspondence** (CorrID, Date, Summary, Message, ApplicantID, StaffID, CorrMeth, toStaff  **Primary key** CorrID  **Foreign key** ApplicantID **references** Applicant(ApplicantID)  **Foreign key** StaffID **references** UniversityStaffMember(StaffID)  **Foreign key** CorrMethod **references** Correspondence Method(Method) |
| **Correspondence Method** (Method)  **Primary key** Method |
| **Country** (CountryISOCode, Name)  **Primary key** CountryISOCode |
| **Decision** (DecID, Date, Comment, ApplicationID, StaffID, decType)  **Primary key** DecID  **Foreign key** ApplicationID **references** Application(ApplicationID)  **Foreign key** StaffID **references** UniversityStaffMember(StaffID)  **Foreign key** DecTypeID **references** Decision Type(type) |
| **Decision Type** (type)  **Primary key** type |
| **Degree** (DegID, ApplicantID, Name, Type, YearCompleted, GPA, InstitutionName, InstitutionCountryISOCode)  **Primary key** DegID  **Foreign key** ApplicantID **references** Applicant(ApplicantID)  **Foreign key** InstituitonCountryISOCode **references** Country(CountryISOCode) |
| **Document** (DocID, UploadLink, Title, Description, DocType, DocStat, ApplicantID, ApplicationID)  **Primary key** UploadLink  **Foreign key** DocType **references** DocumentType(Type)  **Foreign key** DocStat **references** DocumentStatus(Status)  **Foreign key** ApplicantID **references** Applicant(ApplicantID)  **Foreign key** ApplicationID **references** Application(ApplicationID) |
| **DocumentStatus** (Status, Description)  **Primary key** Status |
| **DocumentType** (Type, Description)  **Primary key** Type |
| **Payment Method** (Method)  **Primary key** Method |
| **Publication** (PubID, ApplicantID, Title, Publisher, IssueNo, IssueDate, OnlineLink, OtherAuthorsNames, Language)  **Primary key** PubID  **Foreign key** ApplicantID **references** Applicant(ApplicantID) |
| **Referee** (RefID, ApplicationID, Name, Relation, Phone, Email, AcademicLink, EnglishSpeaker, EnglishLiterate)  **Primary key** RefID  **Foreign key** ApplicationID **references** Application(ApplicationID) |
| **Research Area** (FORCode, Description, ResearchArea, GeneralArea)  **Primary key** FORCode |
| **Supervise as** (StaffID, ApplicationID, PrimarySupervisor)  **Primary key** StaffID, ApplicationID  **Foreign key** ApplicationID **references** Application(ApplicationID)  **Foreign key** StaffID **references** University Staff Member(StaffID) |
| **University Staff Member** (StaffID, FName, LName, canSupervise)  **Primary key** StaffID |
| **University Staff Member\_Application** (StaffID, ApplicationID)  **Primary key** StaffID, ApplicationID  **Foreign key** StaffID **references** University Staff Member(StaffID)  **Foreign key** ApplicationID **references** Application(ApplicationID) |
| **University Staff Member\_Research Area** (StaffID, FORCode)  **Primary key** StaffID, FORCode  **Foreign key** StaffID **references** University Staff Member (StaffID)  **Foreign key** FORCode **references** Research Area(FORCode) |
| **University Staff Member\_Research Area2** (StaffID, FORCode)  **Primary key** StaffID, FORCode  **Foreign key** StaffID **references** University Staff Member (StaffID)  **Foreign key** FORCode **references** Research Area(FORCode) |
| **Visa** (VisaID, ApplicantID, OriginCountryISOCode, VISAStatus, ValidFrom, ValidTo)  **Primary key** VisaID  **Foreign key** ApplicantID **references** Applicant(ApplicantID)  **Foreign key** OriginCountryISOCode **references** Country(CountryISOCode) |
| **Visa Status** (Status, Description)  **Primary key** Status |

## Logical E-R Diagram



# User transaction validation

University Staff Members

|  |  |
| --- | --- |
| # | pathway |
|  | Look up applicant + publications + degrees + visa Status + Associated documents by applicant name |
|  | Look up applicant’s applications by applicant name |
|  | Look up applicant’s applications by applicant email |
|  | Look up incomplete applications |
|  | Look up all correspondences relevant to an application |
|  | Create new applicant and associated application records |
|  | Look up which staff member updated an Application most recently |
|  | Check for any decision recorded about an application |
|  | Look up an existing application and attach a new standard type document to an application |
|  | Look up an existing application and attached a new exceptional type document to an application |
|  | Look up an existing application and list outstanding information (checklist). |
|  | Update the checklist to confirm that a mandatory information requirement has been met |
|  | Retrieve all on-going applications for which the user has made the most recent correspondence |
|  | Record making a decision about an application |
|  | Update the status of an application |

Academic Staff

|  |  |
| --- | --- |
| # | pathway |
|  | Look up, add to, and delete from own current research areas |
|  | Search for all applications in certain research areas that have been added since a certain time |
|  | Flag interest in an application |

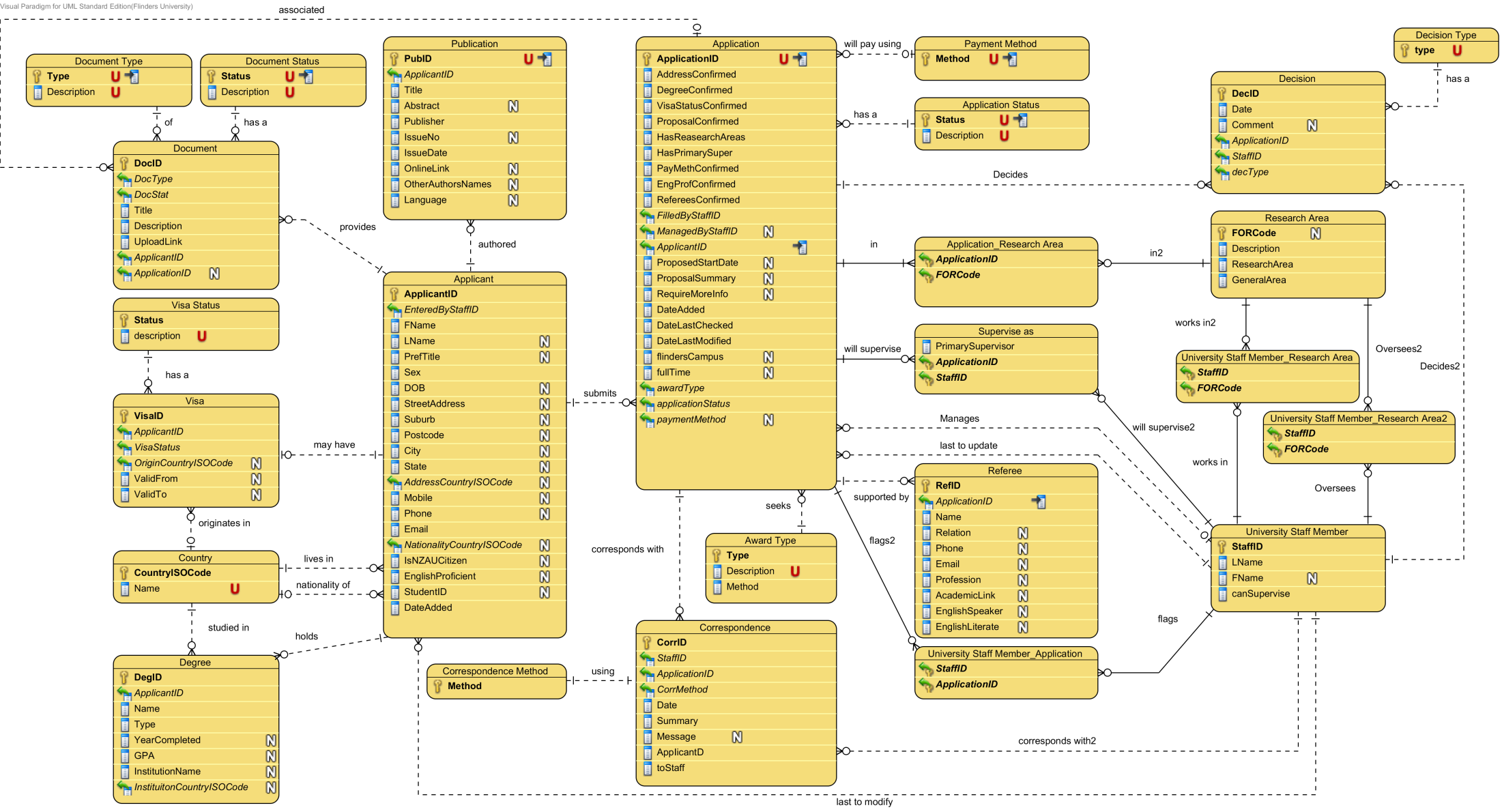
RHD Co-ordination Staff

|  |  |
| --- | --- |
| # | pathway |
|  | Retrieve all staff who have flagged an application, or have edited an application or applicant record most recently |
|  | Retrieve all ongoing applications |

Note that the following modelling changes were triggered when validating user transactions:

* (e) – we had Correspondence in relationship with Applicant, but that could not record which correspondence was in regard to which application. We changed this to have Correspondence in relationship Application.

## Transaction pathways



a1

a2

a3

a4

a5, i, j

b, c,

f, i

e, m

r

s

g, s

r

s

p

p

q

q

h, n

d, t, o

k, l, o

# Check integrity constraints

The following type of integrity constraints have been added to the logical model to protect the database from becoming incomplete, inaccurate, or inconsistent.

## Required data

Required data has been specified through the use of not-null attributes identified in section 3 of the conceptual Documentation. These have been reviewed and included in the logical diagram. Since attributes in VP-UML are null-able by default, the following attributes were changed to not null, that is are required:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Applicant** | | **Degree** | | **ResearchArea** | |
| fName | 1.1 | name | 4.1 | FORCode | 9.1 |
| sex | 1.4 | type | 4.2 | Description | 9.2 |
| email | 1.15 | **Document** | | researchArea | 9.3 |
| dateAdded | 1.20 | Title | 5.1 | generalArea | 9.4 |
| **Application** | | uploadLink | 5.3 | **University Staff Member** | |
| dateAdded | 2.3 | documentType | 5.4 | staffID | 12.1 |
| dateLastChecked | 2.4 | documentStatus | 5.5 | lName | 12.3 |
| dateLastModified | 2.5 | **Publication** | | **Visa** | |
| awardType | 2.7 | title | 7.1 | VisaStatus | 13.3 |
| **Checklist** | | publisher | 7.3 | **Correspondence** | |
| applicationStatus | 3.1 | issueDate | 7.5 | date | 14.1 |
| addressConfirmed | 3.2 | **Referee** | | Summary/message | 14.2 |
| degreeConfirmed | 3.3 | name | 8.1 | corrMeth | 14.4 |
| visaConfirmed | 3.4 | relation | 8.2 | toStaff | 14.5 |
| proposalConfirmed | 3.5 | email | 8.4 | **Decision** | |
| engProfConfirmed | 3.6 | Profession | 8.5 | date | 15.1 |
| hasResearchAreas | 3.7 |  |  | decType | 15.2 |
| hasPrimarySuper | 3.8 |  |  | **SuperviseAs** | |
| payMethConfirmed | 3.9 |  |  | primarySupervisor | 16.1 |
| refereesConfirmed | 3.10 |  |  |  |  |

All other attributes not mentioned here are required to maintain integrity.

Note that some of these attributes are moved to separate entities in the next section, in these cases the foreign key will be not null. Furthermore any attributes added to one of these entities will also be not null and unique.

## Attribute domain constraints

The attribute domains identified in section 4 of the conceptual documentation, have been introduced in the form of lookup relations, as produced in sections 2.2 and 2.3, and have been added to the logical diagram.

## Multiplicity

The multiplicities identified in section 2 of the conceptual documentation and above in sections 1.3-17, have been added to the diagram. The additional multiplicities for the lookup tables identified in sections 2.2 and 2.3 have also been added to the logical diagram.

## Entity integrity

The primary keys identified in section 5 of the conceptual documentation, have been reviewed in section 1 and added to/updated in the diagram. The additional primary keys for the attribute domain entities identified in section 4.2 have also been identified in section 1 and added to the diagram.

## Referential integrity

The primary key of each parent relation has been added to the child relation in the form of a foreign key as specified by the Strong and weak entity table in sections 1.1, 1.2 and 1.7.

### Null attributes (~insert rules)

It should be noted however that the attribute domain relations Identified in section 4.1 that replace null-able attributes will add their primary key as a null-able foreign keys in the child relation. More specifically Award Type, Country, Payment Method and Study Load and Location can all be null in the applicant/applicant entities because they may not be known at the time of the initial input of the entry.

### Updates & Deletes

It is expected that there will be a significant amount of updates as an application (and applicant) progresses through the process. As such all entities have been set to cascade upon any update of the parent table. Even though it is expected that deletions will be rare (if at all) functionality has been included to maintain the integrity of the database. Deletions are performed such that

* All weak entities, that are children of applicant and or application only, CASCADE on delete excluding document which requires applicant to be deleted and
* All lookup tables and University Staff member relationships RESTRICT on all deletes.

This will enable all application/applicant specific material to be removed upon the deletion of an applicant and all application material removed upon a delete. Correspondingly the removal of statuses and other lookup tables (or staff tables) will not leave any of its children such as document without a status. This is particularly relevant for application status as the deletion of a status, will make it lose its point in the process.

RESTRICT has been chosen as it reflects the requirement that nothing should be deleted, since it is possible that mistakes and repeated data can be entered some deletion is enable in this way.

## General constraints.

Currently there are no high level constraints that will affect the validity of the logical model. If any are later realised that will be discovered during Review or the final checks of the physical diagram.

There are however pseudo constraints through the checklist (now merged into the application) whereby an application cannot have certain application statuses unless the corresponding checklist attributes are true.

## Document all integrity constraints

All integrity constraints have been applied to the logical model, which can be used to produce a data dictionary when required.

# Review logical data model

The Logical data model will be reviewed with Paul, the head of the Research Higher Degree Office to ensure that it meets all their requirements and is a true (or as close to possible) representation of the data requirements as specified by the Research Higher Degree Office and the staff members who will use the database.

# Check for future growth

It has been found that there are several possible areas that the logical database model may need to include in future. These include but are not limited to

* Expanding research areas, within the school of CSEM or the wider university,
* The possible inclusion of a housing & transport weak entity/attribute for the application/applicant,
* The inclusion of some form of disability/issues entity and or
* The addition of other statuses, types or other lookup tuples.

It has been found that these areas can be accounted for, in the form of adding new entities or attributes. However some possible future growths areas such as the reuse of referees for anther applications (by the same applicant) will break the model and force a redesign. Even though the redesign will be minor, creating a RefereeApplicaiton entity of referee and application IDs and replacing applicationID with applicantID (if maintaining connection with the applicant), it will mean the database and any programs will have to go offline while the modifications are made.

It has been assumed that such cases occur rarely and as such re-entering details will not be too inefficient. This will ensure that any detail changes that occur between applications are added as different entries.

# Develop Test Plan

# Data dictionary

# Conceptual E-R Diagram

# Logical E-R Diagram

1. applied methodology as described in ‘Chapter 18 Methodology – Monitoring and Tuning the Operational System Step 7.2 Duplicating non-key attributes in one-to-many (1:\*) relationships to reduce joins’ [↑](#footnote-ref-1)
2. applied methodology as described in ‘Chapter 18 Methodology – Monitoring and Tuning the Operational System Step 7.2 Duplicating non-key attributes in one-to-many (1:\*) relationships to reduce joins’ [↑](#footnote-ref-2)