

# KinOath, Kinship Software

## Beta Stage of Development

The kinship archiving software under development by Peter Withers at the Language Archive, MPI, Nijmegen

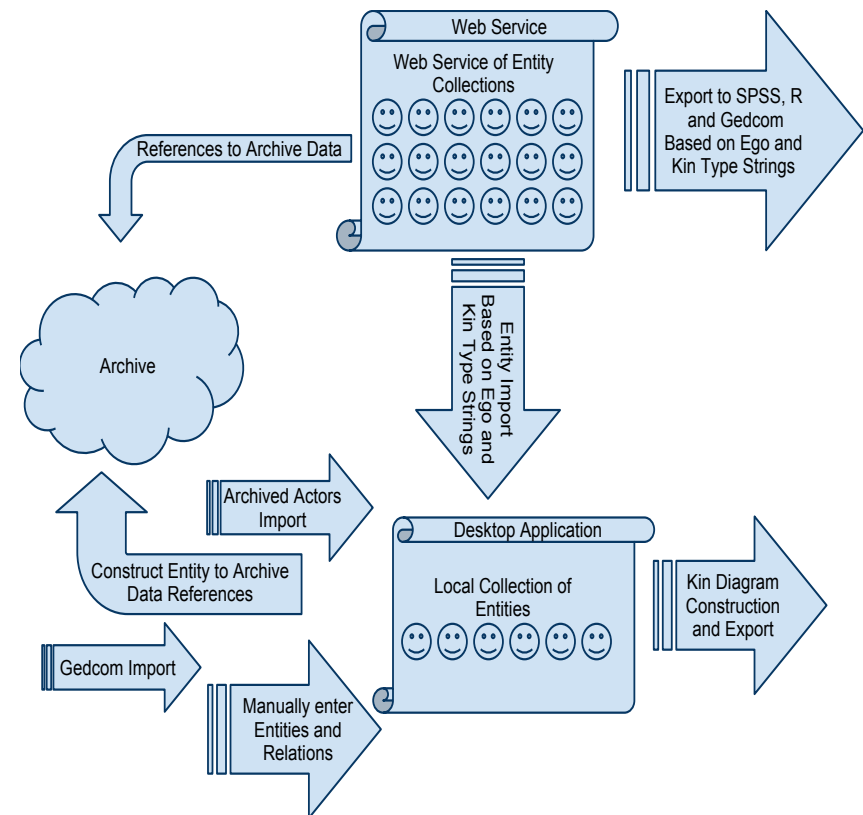
# Introduction

- KinOath is a kinship application under development by Peter Withers at the Language Archive of the Max Planck Institute for Psycholinguistics Nijmegen.
- Its primary goal is to connect kinship data with archived data, such as audio, video or written resources while also being closely integrated with the archive software such as Arbil.
- Beyond this goal it is designed to be flexible and culturally nonspecific, such that culturally different social structures can equally be represented.

# Application Overview

There are two main parts to the application:

- Web Application
- Desktop Application
- Each of these can link to archived or external data



# Current State of Development

- The desktop application is in beta stage
- The web application is still under development
- This talk will show examples of how the current beta of the desktop application can be used and also discuss features that are planned or in development

# Beta Stage of Development Means:

- Usability testing has begun and the interface is being refined
- Most features are present, but not all features fully functional
- Formal software testing has not begun so there will be bugs
- Public beta release will be very soon

# Core Aspects

- Graphical representation of the data is an important part of the application and the diagrams produced are intended to be very flexible and of publishable quality.
- Kin type strings are used throughout the application for constructing and searching data sets.
- The representation of kin terms is also integrated into the application allowing comparative diagrams of kin terminology.

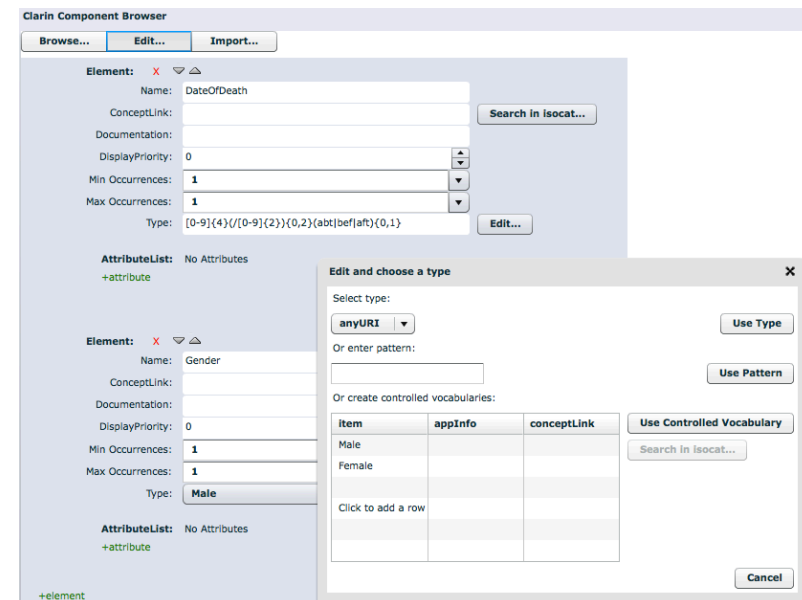
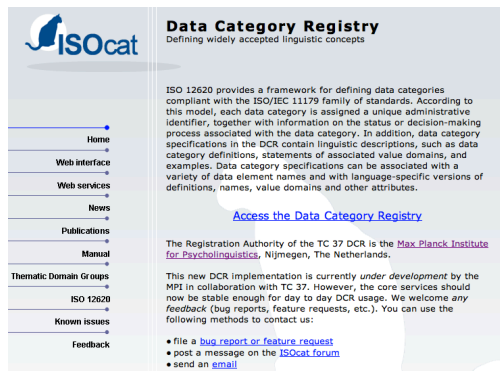
# Publishable Diagrams

- All the diagrams produced are in a vector format of publishing quality
- All diagrams can be exported into PDF format
- The working files are vector graphics files that can be viewed in a web browser



# Flexible Data Structure

- The Clarin Component Registry can be used
- <http://catalog.clarin.eu/ds/ComponentRegistry>
- <http://www.isocat.org/>



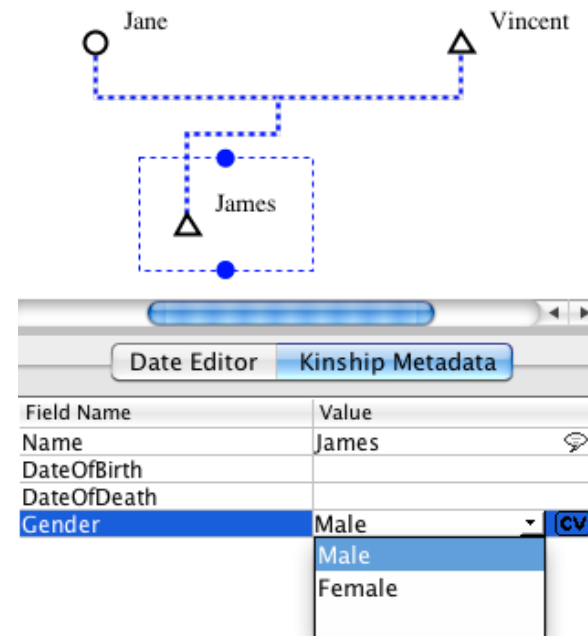


# Multiple Data Types

- The data structure designed in the component registry can be used on the kin diagrams
- Multiple data structure types are possible on the same diagram

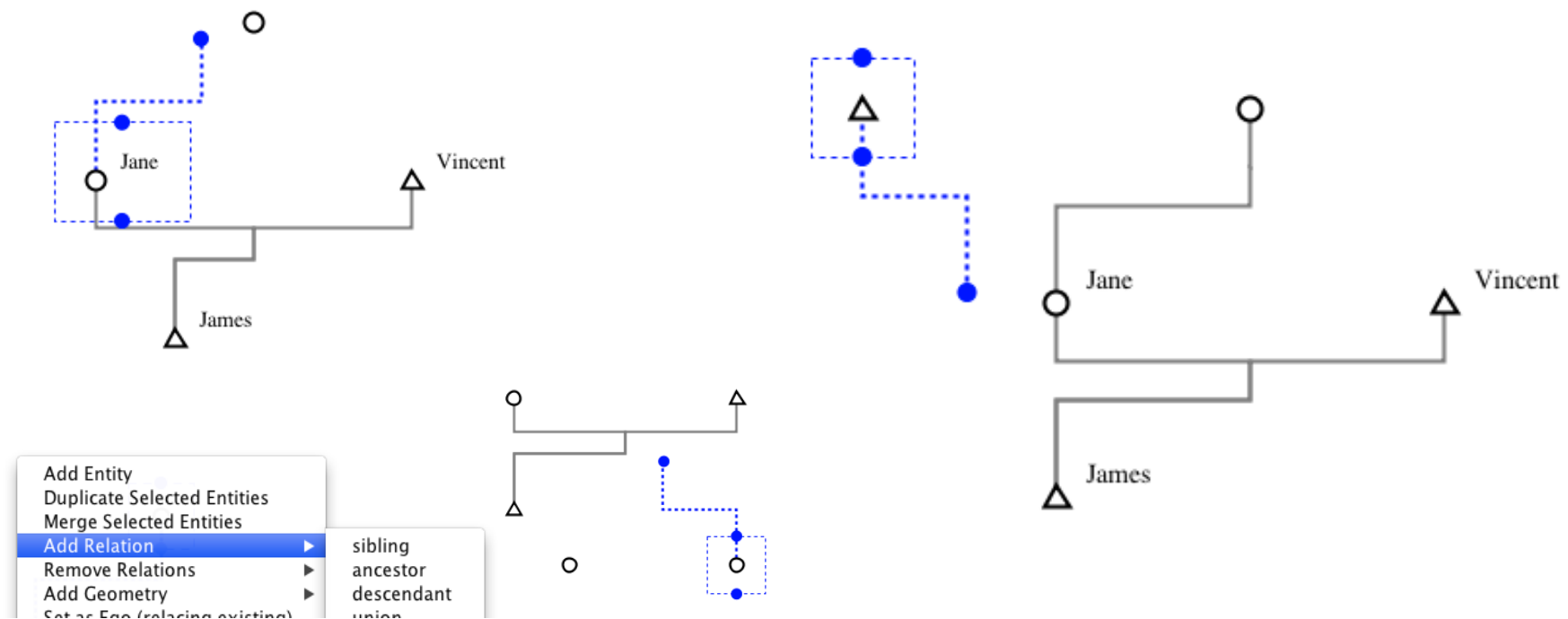
The screenshot shows the 'Clarin Component Browser' window. The 'Element' tab is active, displaying configuration for 'DateOfDeath'. Fields include Name, ConceptLink, Documentation, DisplayPriority (0), Min Occurrences (1), Max Occurrences (1), and Type ([0-9]{4}/[0-9]{2}){0,2}(abt|bef|aft){0,1}. An 'AttributeList' section shows 'No Attributes'. Below, the 'Elements' section shows 'Gender' with a 'Type' of 'Male'. An 'Edit and choose a type' dialog is open, showing 'anyURI' selected and a table for controlled vocabularies.

Item	appInfo	conceptLink
Male		
Female		
Click to add a row		



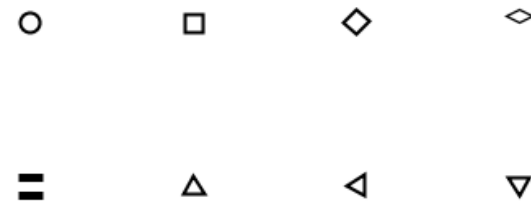
# Adding Relations

Relations can be constructed either via the context menu or by dragging the blue dots with the mouse



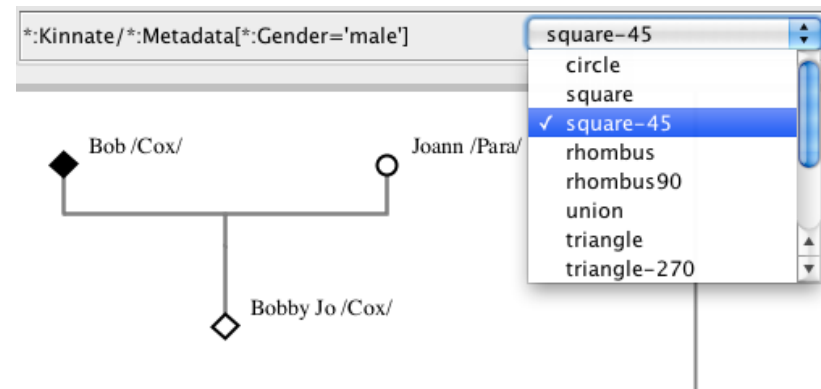
# Creating Custom Symbols

- Custom symbols can be inserted into a kin diagram and used like any existing symbol
- Currently there is no simple way to add them via the user interface
- In the interim time this can be achieved by using an external editor such as Inkscape and adding the new symbol manually.



# Selecting Custom Symbols

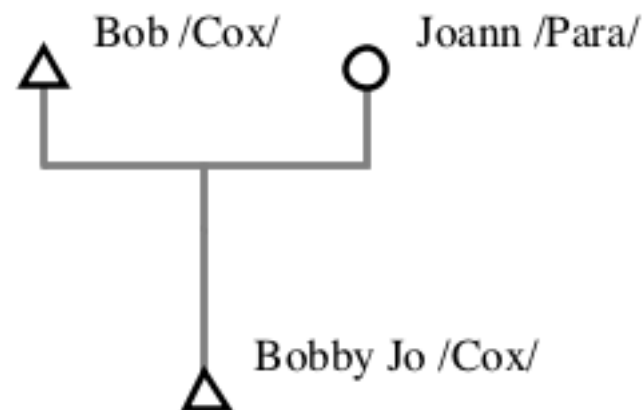
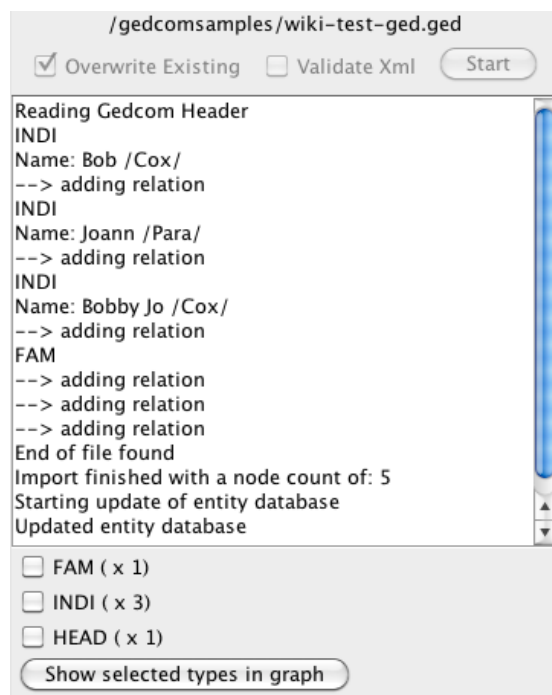
- The symbols are placed on the diagram based on the data for that individual
- Any number of symbols can be use on a diagram
- For instance `"*:Kinnate/*:Entity[*:Gender='male']"` can be associated with the symbol "rhombus"
- Note that the table data is extremely flexible and could equally be `"*:Kinnate/*:Entity[*:caste='Y']"`
- Additional overlay symbols will be available in the future



Field Name	Value
Name	
DateOfBirth	
DateOfDeath	
Gender	Male

# Gedcom Import

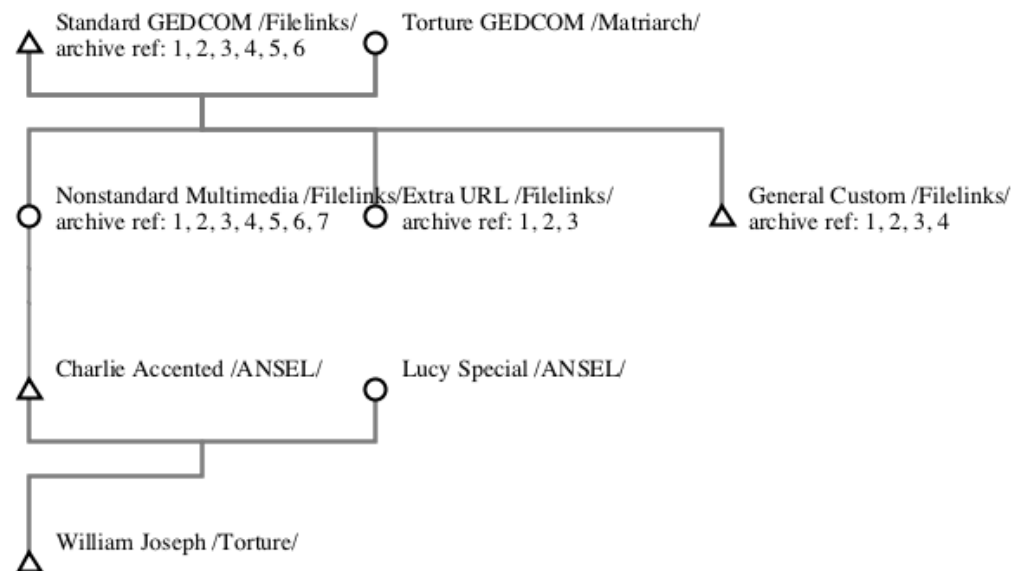
```
sample.ged
0 HEAD
1 SOUR Reunion
2 VERS V8.0
2 CORP Leister Productions
1 DEST Reunion
1 DATE 11 FEB 2006
1 FILE test
1 GEDC
2 VERS 5.5
1 CHAR MACINTOSH
0 @I1@ INDI
1 NAME Bob /Cox/
1 SEX M
1 FAMS @F1@
1 CHAN
2 DATE 11 FEB 2006
0 @I2@ INDI
1 NAME Joann /Para/
1 SEX F
1 FAMS @F1@
1 CHAN
2 DATE 11 FEB 2006
0 @I3@ INDI
1 NAME Bobby Jo /Cox/
1 SEX M
1 FAMC @F1@
1 CHAN
2 DATE 11 FEB 2006
0 @F1@ FAM
1 HUSB @I1@
1 WIFE @I2@
1 MARR
1 CHIL @I3@
0 TRLR
```



- All Gedcom fields are imported
- This simple example is from the Wikipedia Gedcom page
- <http://en.wikipedia.org/wiki/Gedcom>
- For testing the “GEDCOM 5.5 Torture Test Files” are used
- <http://www.geditcom.com/gedcom.html>

# Resource Files

- Like manually created resource links, imported GEDCOM resources files are available both on the diagram and via the diagram tree



# Kin Type Queries

- Kin type queries consist of kin types followed by search parameters

E[Margaret of\_Sweden]

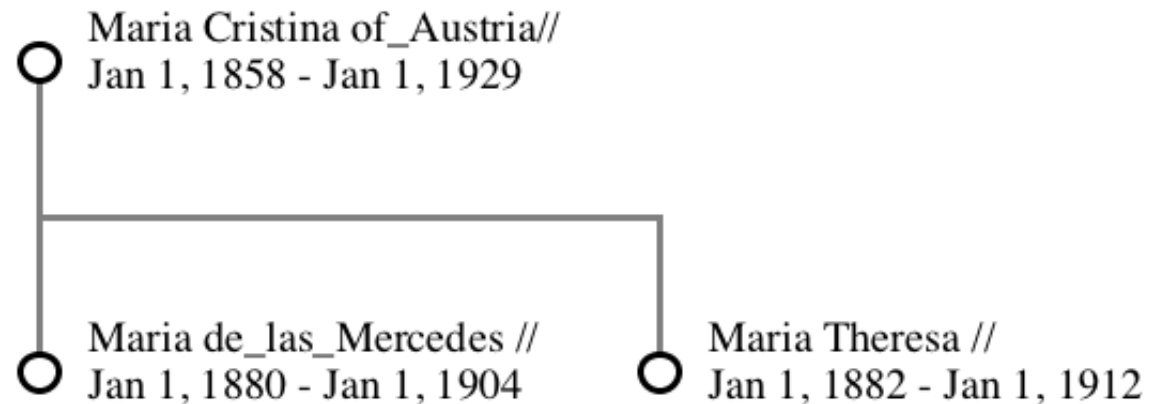
● Margaret of\_Sweden //  
Jan 15, 1882 - May 1, 1920

- This example will show any entity that contains the string “Margaret of\_Sweden” on the diagram, in this case there is only one match

# Kin Type Query Syntax

- The kin types following a query will add any matching kin, for example the daughters of Maria

x[Maria Cristina of\_Austria]D





# Dragging to the Query Text

- The queries can be constructed by dragging values from the table below (when entities are selected) onto the query text

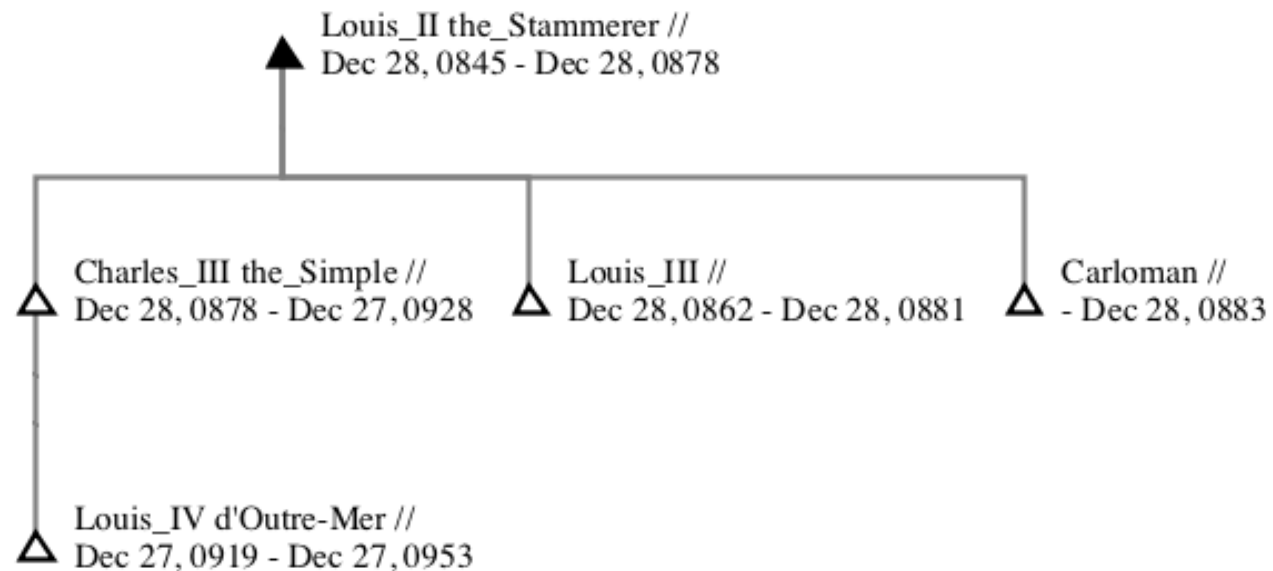
E[INDI.NAME=Louis\_II the\_Stammerer //]

Field Name	Value
NAME	Louis_II the_Stammerer //
TITL	King of France
SEX	M
BIRT.DATE	846
DEAT.DATE	879

# Kin Type String Queries

- Multiple queries can be used per kin type
- Queries can use = contains, == exact match, > greater than, < less than

E[DateOfBirth<0850][INDI.TITL=King of France]CC



# Kin Type Definitions

- The kin types used in the application can be customized
- Each kin type can use any string and any symbol
- These custom kin types are stored in the diagram file

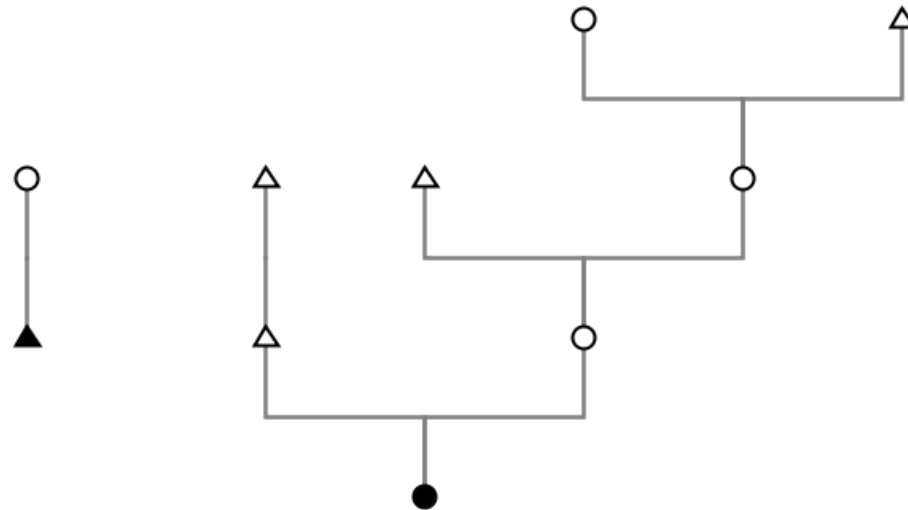
Kin Type String	Relation Type	Symbol Type	Display Name	
Ef	none	circle	Ego Female	<input type="checkbox"/>
Em	none	triangle	Ego Male	<input type="checkbox"/>
Fa	ancestor	triangle	Father	<input type="checkbox"/>
Mo	ancestor	circle	Mother	<input type="checkbox"/>
Br	sibling	triangle	Brother	<input type="checkbox"/>
Si	sibling	circle	Sister	<input type="checkbox"/>
So	descendant	triangle	Son	<input type="checkbox"/>
Da	descendant	circle	Daughter	<input type="checkbox"/>
Hu	union	triangle	Husband	<input type="checkbox"/>
Wi	union	circle	Wife	<input type="checkbox"/>
Pa	ancestor	square	Parent	<input type="checkbox"/>
Sb	sibling	square	Sibling	<input type="checkbox"/>
Sp	union	square	Spouse	<input type="checkbox"/>
Ch	descendant	square	Child	<input type="checkbox"/>
F	ancestor	triangle	Father	<input type="checkbox"/>
M	ancestor	circle	Mother	<input type="checkbox"/>
B	sibling	triangle	Brother	<input type="checkbox"/>
Z	sibling	circle	Sister	<input type="checkbox"/>
S	descendant	triangle	Son	<input type="checkbox"/>
D	descendant	circle	Daughter	<input type="checkbox"/>
H	union	triangle	Husband	<input type="checkbox"/>
W	union	circle	Wife	<input type="checkbox"/>
P	ancestor	square	Parent	<input type="checkbox"/>
G	sibling	square	Sibling	<input type="checkbox"/>
E	none	square	Ego	<input type="checkbox"/>
C	descendant	square	Child	<input type="checkbox"/>
m	none	triangle	Male	<input type="checkbox"/>
f	none	circle	Female	<input type="checkbox"/>
x	none	square	Undefined	<input type="checkbox"/>
*			Any Relation	<input type="checkbox"/>

F	ancestor	triangle	Father
M	ancestor	circle	Mother
B	sibling	triangle	Brother
Z	sibling	circle	Sister
S	descendant	triangle	Son
D	descendant	circle	Daughter
H	union	triangle	Husband
W	union	circle	Wife
P	ancestor	square	Parent
G	sibling	square	Sibling
E	none	square	Ego
C	descendant	square	Child

# Freeform Diagrams

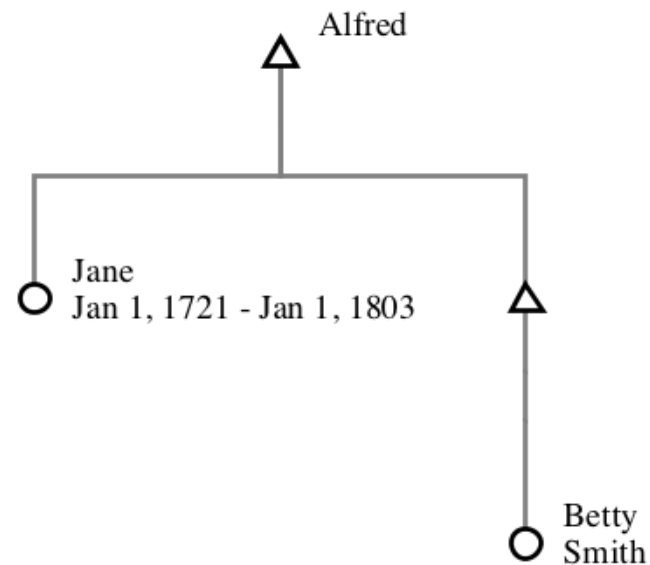
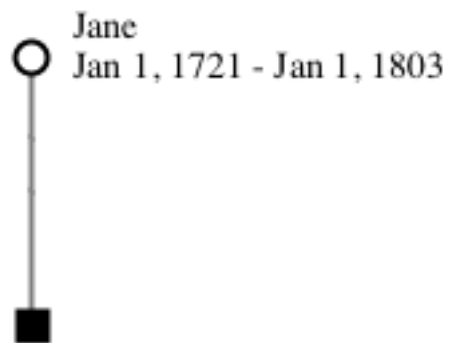
- Freeform diagrams are constructed simply by entering kin type strings

EfFF  
EfMF  
EfMMF  
EfMMM  
Em  
EmM



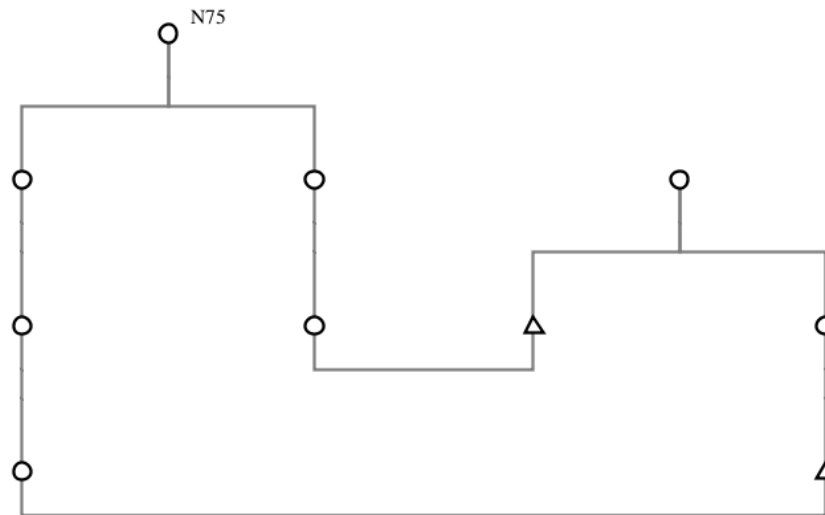
# Freeform Diagram Syntax

- <KinType>:<id>;<label>;<label...>;<DOB>-<DOD>:<KinType...>
- EM:Jane;1721-1803:
- f:#3;Jane;1721-1803:F:Alfred:SD:Betty;Smith:

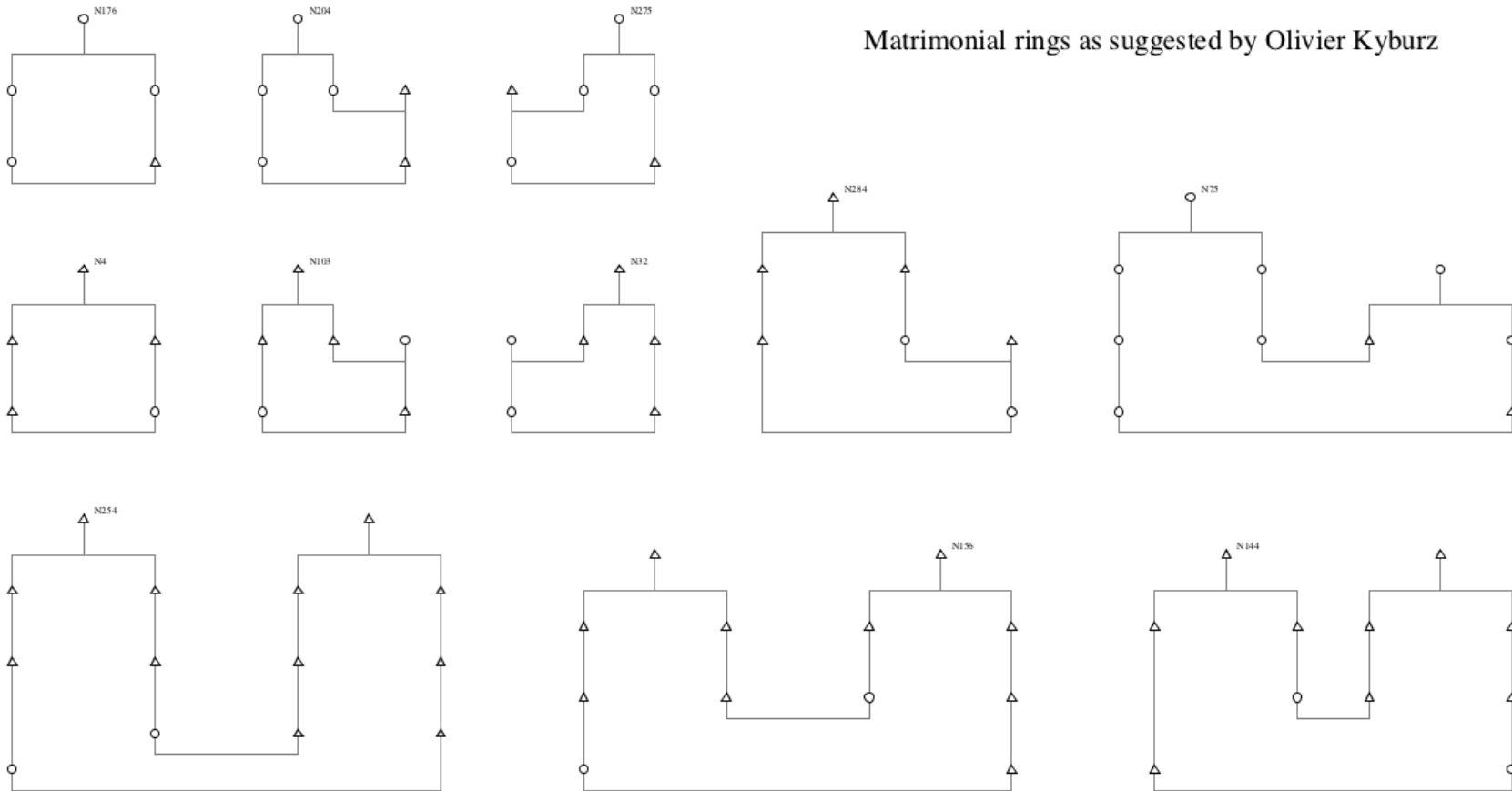


# Freeform Diagram Syntax

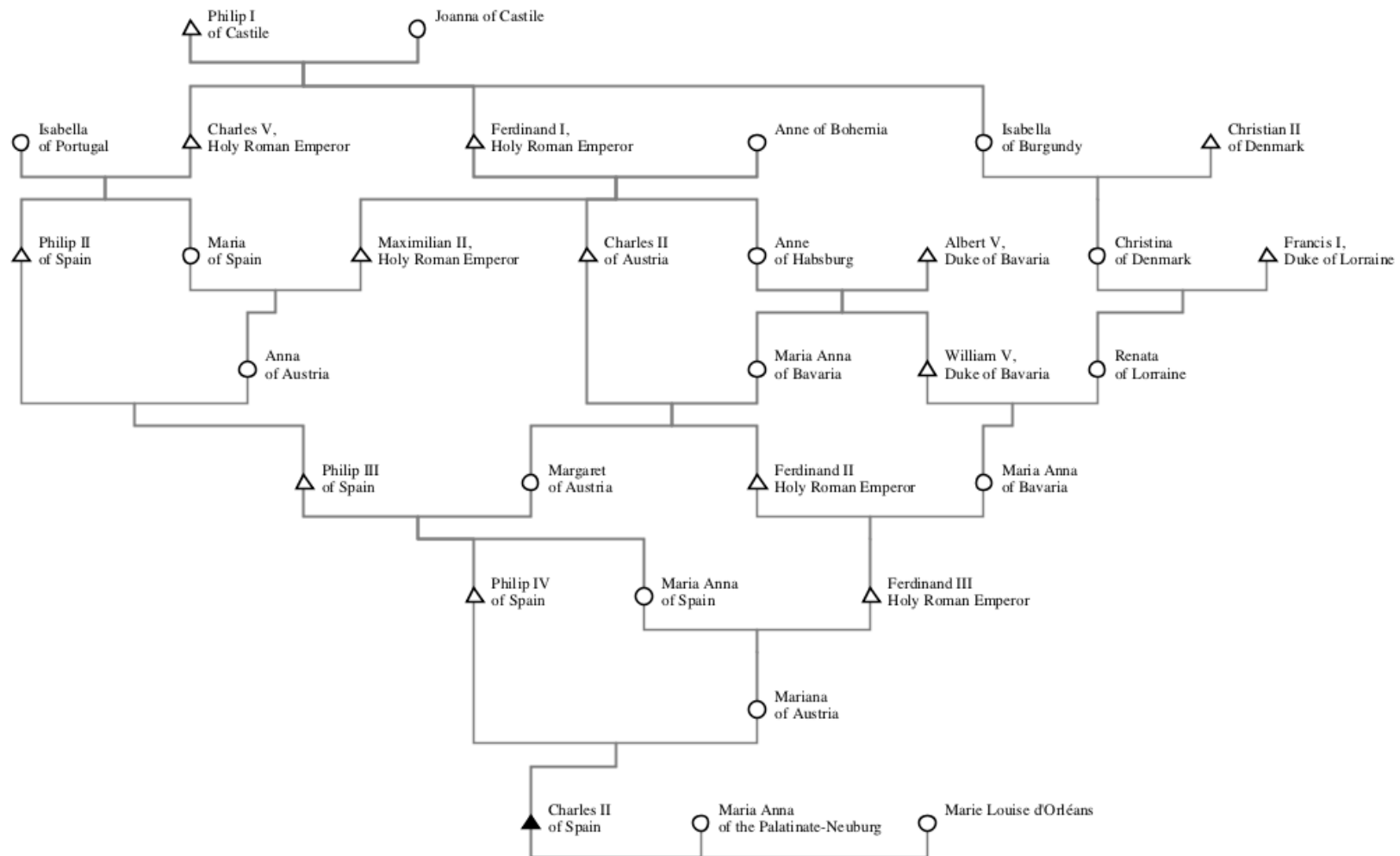
- By using the <id> preceding individuals can be referred to later in the string
- Here id #75 is used as a back reference to the first individual
- f:#75;N75:DDHMDSWMMM:#75:



# Example: Matrimonial Rings



# Example: Charles II of Spain



December 19, 2011

Peter.Withers@mpi.nl The Language  
Archive, Max Planck Institute for  
Psycholinguistics, Nijmegen



# Potential Interoperability

There are a number of opportunities for interoperability with other software

- Sharing data via the desktop application or via the web service
- Post or pre processing with R or SPSS
- A plugin framework to allow third party code gain access to the data and the graphical output features

# R and SPSS

- The queries as used in the diagram can also be used via the web application
- This can be used in R and SPSS as a data source
- This web service is not yet publicly available but it functions as follows:  
<webservice-url><kintype-query>  
where the kintype-query for example would be E[Bob]MMZ
- Usage in R would be as follows:  

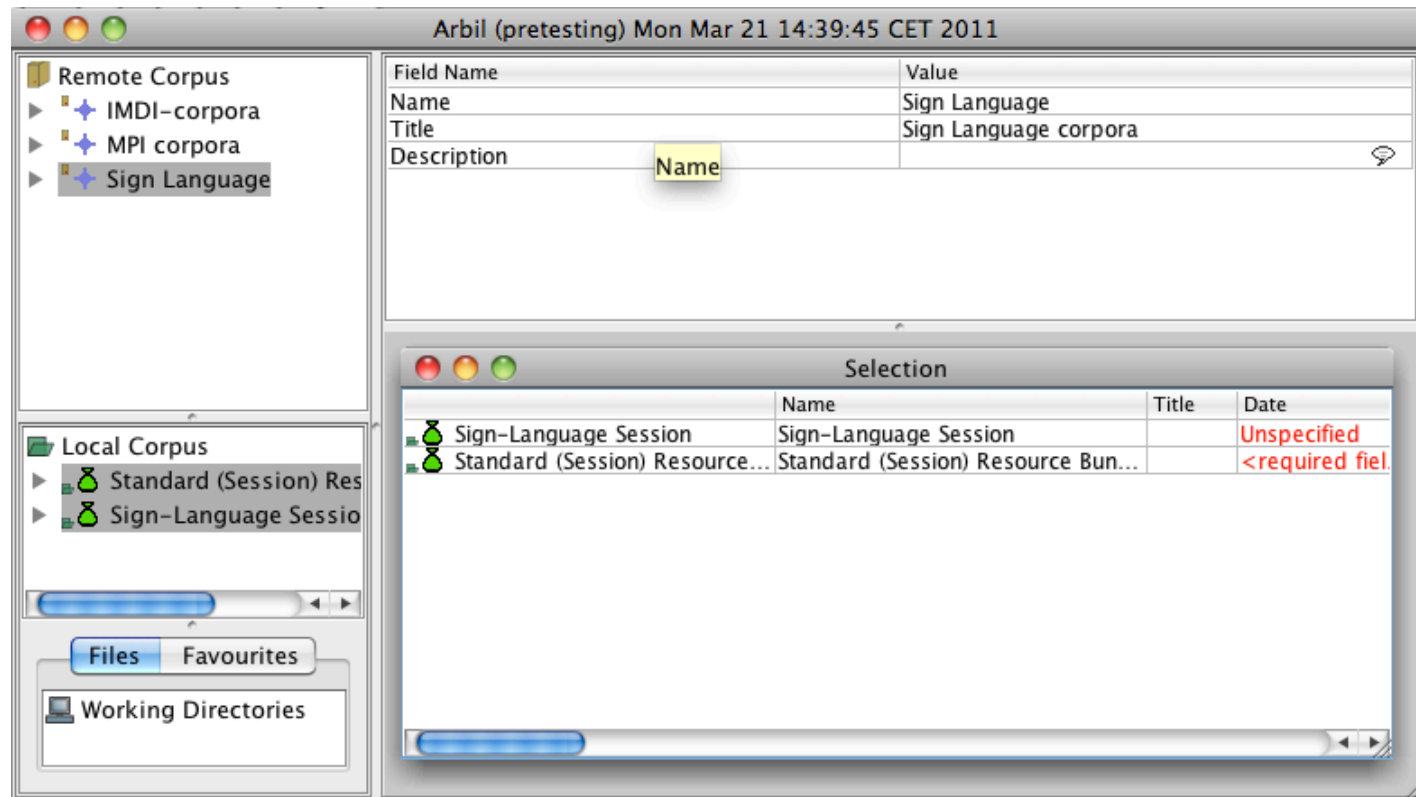
```
dataFrame <- read.table("http://mpi.nl/tls/kinship:8080/kinoath-rest/  
kinoath/getkin/csv?kts=E[Bob]MMZ",header=T)  
library(kinship)  
attach(dataFrame)  
pedigreeObj <- pedigree(id, dadid, momid, sex, affected)  
plot(pedigreeObj)
```

# Linking Archive Data

- Because this kinship application shares a lot of code with Arbil (explained in next slide), there is great flexibility in the metadata that can be consumed by it
- Many of the advantages of the Clarin metadata structures are available including the Data Category Register
- In order to link archive data many of the archive search tools found in Arbil are used
- Not all of these are ready for demonstration but they operate in a similar way as used in Arbil

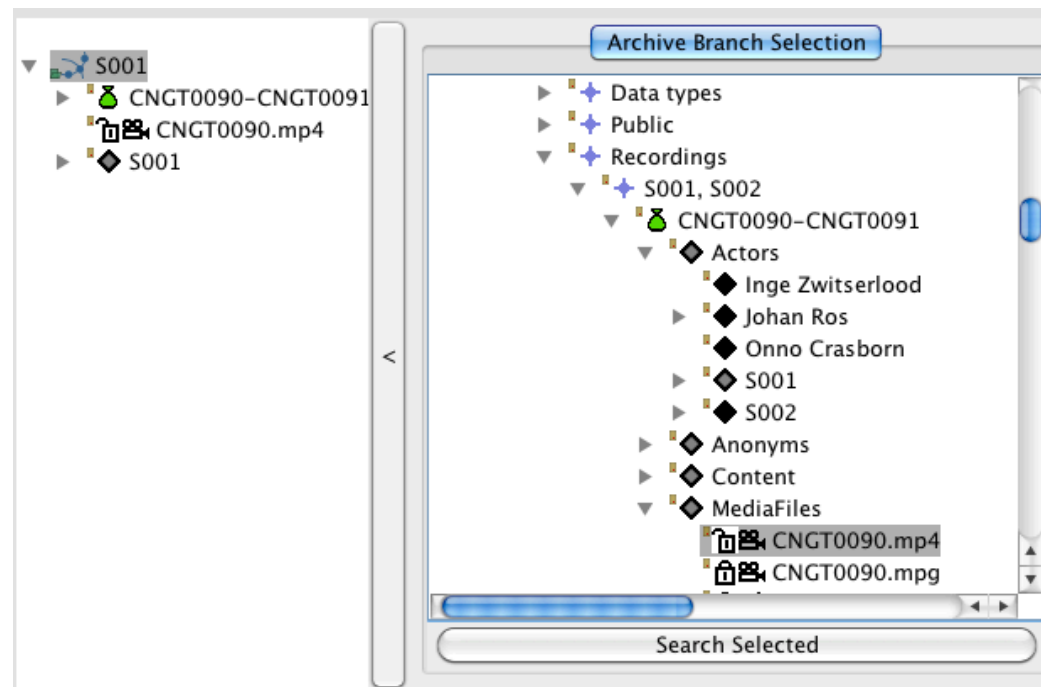
# Archive Intro

- Arbil demo...



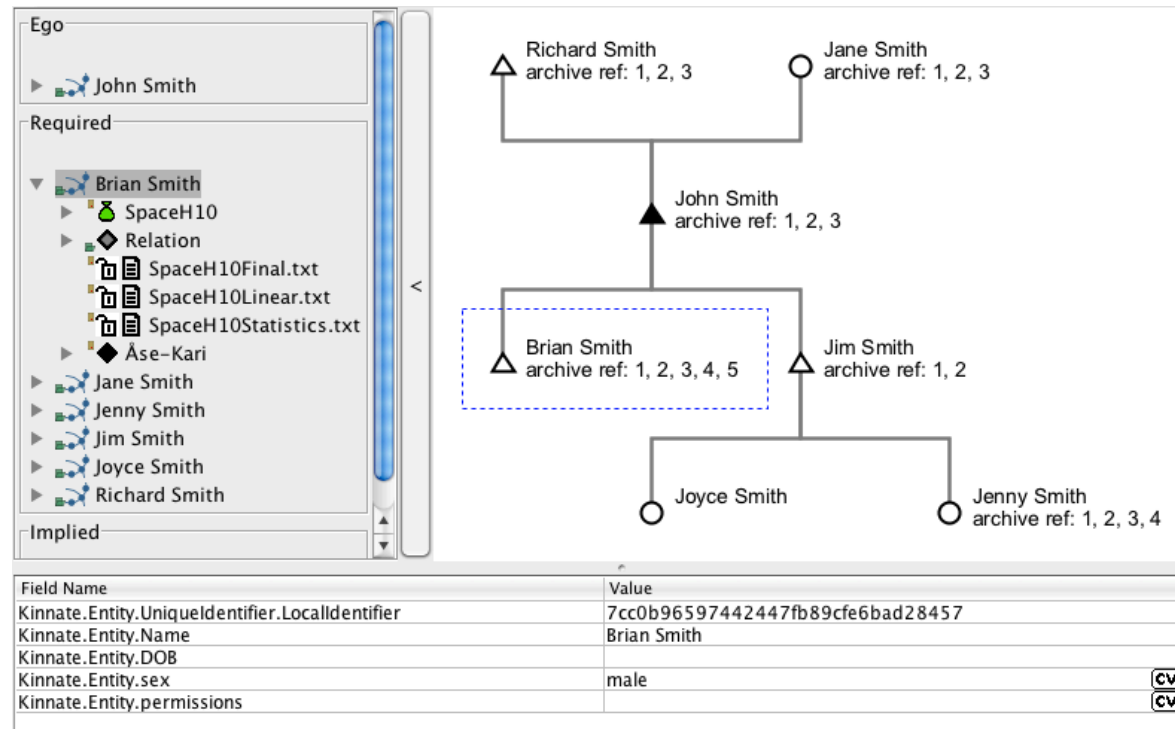
# Creating Archive Links

- By using the archive metadata to create kin entities the manual data entry is reduced



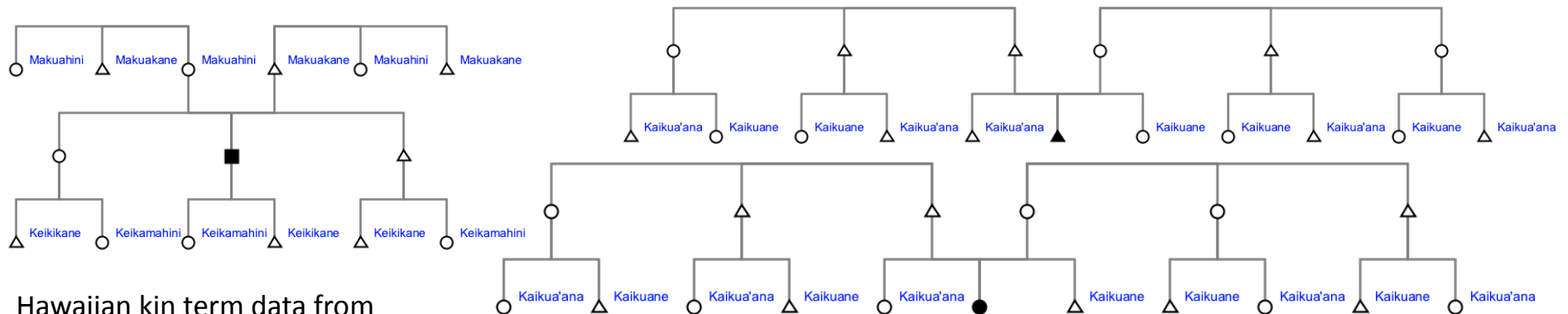
# Archive Links on the Diagram

- When added to a kin diagram the linked archive data are accessible from the kinship diagram



# Kin Terms

- Kin terms can be entered into a diagram
- A freeform diagram can be generated from these kin terms
- In the future these kin terms can be overlaid onto an existing diagram



Hawaiian kin term data from  
<http://umanitoba.ca/faculties/arts/anthropology/tutor/kinterms/hawaiian.html>

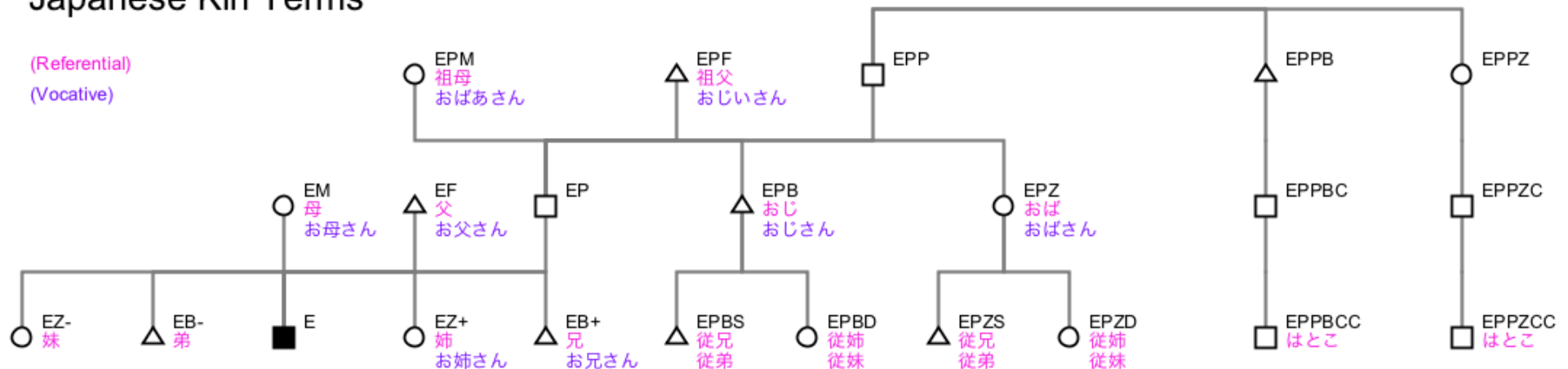
December 19, 2011

Peter.Withers@mpi.nl The Language  
Archive, Max Planck Institute for  
Psycholinguistics, Nijmegen

# Comparative Kin Terms

- Multiple groups of kin terms can be shown on one diagram
- In this case some of the Japanese vocative and referential kin terms are shown on the diagram

## Japanese Kin Terms



Data sourced from <http://ja.wikipedia.org/wiki/親族> and subsequent links

December 19, 2011

Peter.Withers@mpi.nl The Language  
Archive, Max Planck Institute for  
Psycholinguistics, Nijmegen



# Conclusion

- KinOath desktop is now in beta stage and already has many features with many yet to come:
- Create publishable-quality kinship diagrams;
- Query the internal kinship database constructing diagrams with links to archived data;
- Provides an efficient way to construct kinship diagrams including matrimonial rings;
- Create kin term diagrams with multiple kin term groups or language groups on the diagram;
- Provide a link from a published diagram back to the source data in the archive;
- Be financially accessible to everyone (free) .