WORKSHOP: HOW TO CREATE A DATABASE WITH HEURIST

Instructor: Helen Katsiadakis, Historian, Researcher Emerita of the Academy of Athens

East Room of the Main Building of the Academy of Athens, 28 Panepistimiou Ave., Athens, Greece

Wednesday, 9 June 2023, 09:00-15:00 EET

ΑΚΑΔΗΜΙΑ ΑΘΗΝΩΝ

ΨΗΦΙΑΚΟ ΤΟΠΙΟ
στις Ανθρωπιστικές Επιστήμες

ΕΛΙΔΕΚ.
Ελληνικό Ίδρυμα Έρευνας & Καινοτομίας

Workshop diagramme

ΑΚΑΔΗΜΙΑ ΑΘΗΝΩΝ

ΨΗΦΙΑΚΟ ΤΟΠΙΟ
στις Ανθρωπιστικές Επιστήμες

ΕΛ.ΙΔ.Ε.Κ.
Ελληνικό Ίδρυμα Έρευνας & Καινοτομίας

Why databases and not spreadsheets?

- Spreadsheets are NOT databases

- A database requires a relational structure of the data

- A database consists of:
  - Entities or record types
  - Attributes (text fields, numbers etc.)
  - Relationships
  - Management level

A bad spreadsheet example:

Evil spreadsheets

4 entities: Production, Person, Organisation, Place
2 relations: Production - Person (Role),
Production - Organisation (Role)

A bad-case, but not atypical, spreadsheet

Rows are notionally theatrical productions, but include header rows and general notes about a time period. Some productions spread over two or three rows.

- Missing values
- Date range for missing values implied by position
- Inconsistent formatting of text and date values
- Notes appended to values
- Multiple values in one field (People in Cast, but also subsections for Crew, Musicians, Chorus etc.)
- Uncontrolled values eg. company and location (with plenty of typos)

ΑΚΑΔΗΜΙΑ ΑΘΗΝΩΝ

ΨΗΦiΑΚΟ ΤΟΠiΟ
στις Ανθρωπιστικές Επιστήμες

ΕΛ.Ι.Δ.Ε.Κ.
Ελληνικό Ίδρυμα Έρευνας & Καινοτομίας

Why a database?

Advantages:

- Central management of all data

- Consistency of information

- Use of search filters

- Export of subset to webpage, map etc.

- Publication in webpage in real time

- Export of subset to specialized analysis and visualization tools

Not suitable for:

- Purely textual data

- 3D data

- Statistical analysis

Advantages of Heurist

- Allows to quickly design, populate, explore and publish data

- Does not require programming skills (unlike other DBMS systems)

- Is available at no charge

- Is compliant with FAIR principles

- Manages any type of data (text, numbers, geospatial data, various types of file formats, relationships etc.)

- Relationships can be associated with a specific semantic that is searchable

- Allows design modifications and experimenting as part of the research process

- Manages permissions

- Enables working collaboratively.

A selection of Heurist use cases

- [Judaism and Rome](#)

- [Map of the Absentees](#)

- [Beyond 1914](#)

- [Digital Harlem](#)

- [Libraries, Reading Communities & Cultural Formation in the Eighteenth-Century Atlantic](#)

- https://anavathmis.eu/

Help from within the database is accessible in three ways
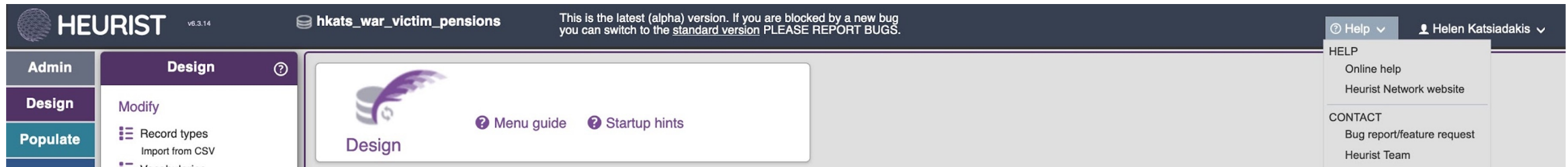
- Online help: Link to the systematic help database: https://heuristref.net/heurist/?db=Heurist_Help_System&website

- Heurist Network website: https://heuristnetwork.org/

- Clicking on one of the Heurist functionality menus on the left and the question mark provides access to a menu guide and startup hints for the specific functionality.

https://heuristnetwork.org

Heurist: Designing without programming
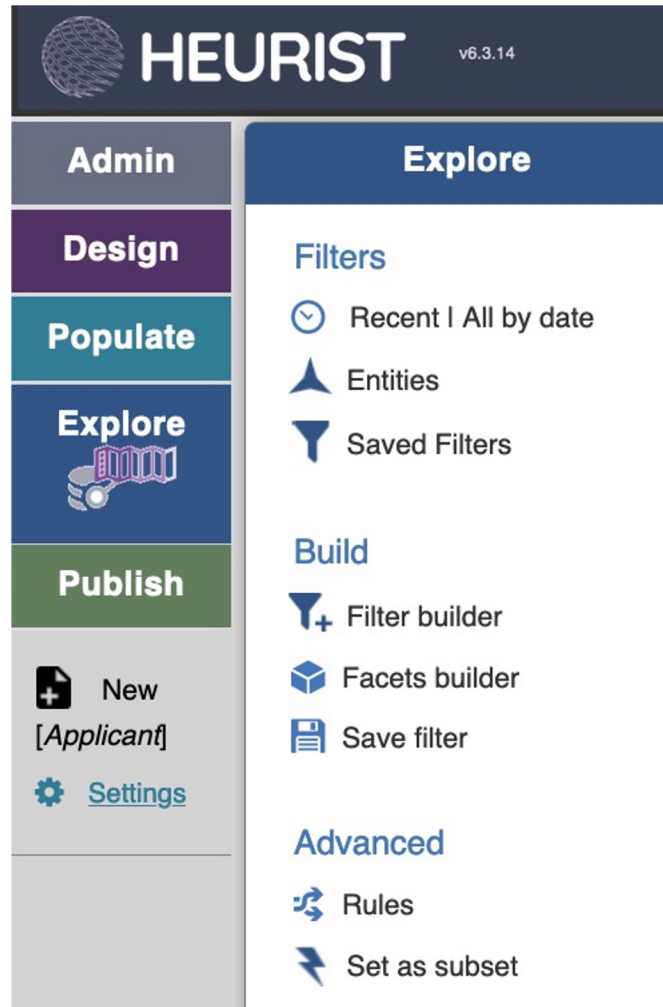
Heurist functionality menus:

- Admin: Administration (Creation of database, user management and integrity verification)

- Design: Design and modification of database structure, entities, fields, vocabularies

- Populate: Adding, importing new data

- Explore: Searching, analyzing and visualizing results

- Publish: Creating websites, exporting data

Model visualization
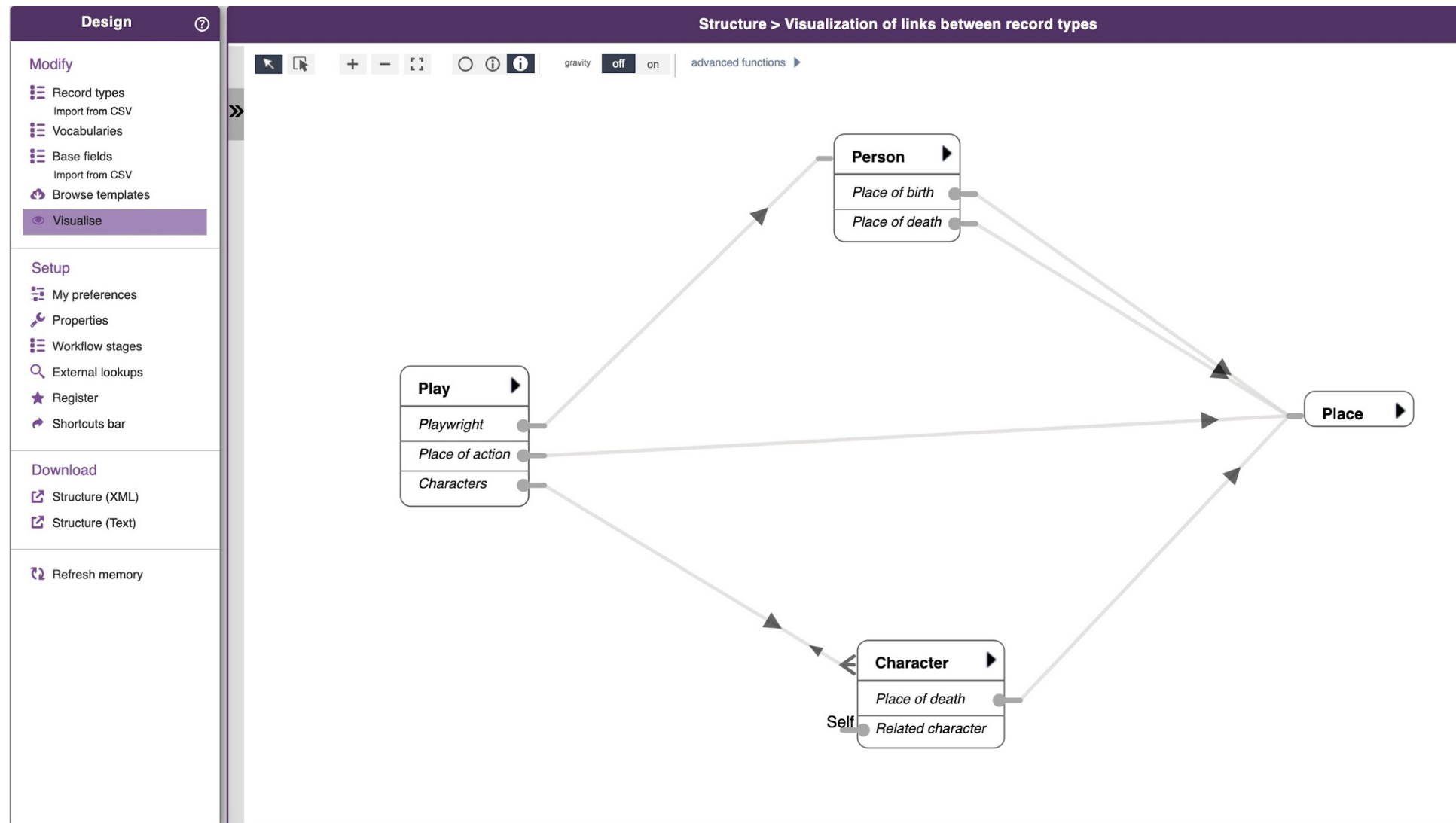
Record type or Entity:
the basic element for organizing information

Record types/ Entities

Vocabularies

Record types/ Entities: Types of fields

Attributes/ Fields:

Text

       Text (single line)

       Extended text (multiple lines)

Dropdown list (vocabularies, thesauri)

Simple fields

       Numbers

       Dates

Complex fields

       Geospatial (points, lines, polygons, circles, rectangles)

       fuzzy data (temporal, geographical boundaries)

       Files/ multimedia (image, video, urls)

Relationships

       Record pointer

       Record relationships

Composite title (for mask)

Designing record type fields

Relationships

Record pointers:

A field of one entity consists of the base field of another entity (foreign key)

E.g.: **Entity Person: Person name**

   Place of birth<<<<<<<<<< Entity Place: Name of place

Record relationships:

This relationship has the characteristics of an entity with its own attributes (relationship type, relationship duration etc.)

E.g.: two persons are related by relationship type 'marriage', relationship duration from date x to date z). This is not a simple relationship. It has its own attributes. It is therefore considered as a distinct entity.

Record Relationship: a distinct entity

Vlachaki Maria | byMarriage.IsMarriedTo - > Vlachakis Nikolaos

🗄️ **Record relationship**: id 627 ✏️

The Relationship type is a vocabulary term which is to be found in
Vocabularies
             RELATIONSHIPS

Primary information

Source record	Vlachaki Maria
Relationship type	byMarriage.IsMarriedTo
Target record	Vlachakis Nikolaos

A special relationship

Child record: a subclass of record pointers

When a record, e.g., of the entity Place, is connected to more records of other entities, this relationship is described as a record pointer.

When a record, e.g., of the entity Works, is connected only to one record of another entity, e.g., the entity Composer, this relationship is described as a child record, a subclass of the record pointer relationship.

E.g.: Composer: Mozart

Child records: Piano Concerto No. 22 in E♭ major, K. 482 …

Symphony No. 40 in G minor, K. 550 …

Don Giovanni, K. 527 …

Other entity properties

Each entity has at least one mandatory field (field name in red). E.g., the Name is the mandatory field for the entity Person.

One field can be single or repeatable, i.e., have multiple entries. For instance, a Writer may have written multiple works.

Field types

Mandatory field → **Family name**

Repeatable field → Alternate name(s) / title(s) ⊞

Vocabulary →

Record pointer →

Date →

Person | Modify structure | ☐ Show help | ☑ Optional fields | ⬇ Template | 🐜 Bug report

Primary information

Field	
Family name	
Given name(s)	
Alternate name(s) / title(s) ⊞	
Gender	▼
Person type ⊞	▼
Birth date	🗓 ⌄
Place of birth ▶	▶ select : Place
Death date	🗓 ⌄
Place of death ▶	▶ select : Place
Cause of death	▼

Save then | Dupe | New | Save | Save + Close | Close | Drop Cha

Creating a new field

SOS: Do not forget the title mask when designing an entity

The title mask is a composite title constructed dynamically from field values. It is a string into which field values are inserted to create the title and is used as the extended title displayed in search results and other lists.

This is an example of a title mask used for one entity:

- [Surname], [Given names]-[Place of origin]-[Date of incident]-[Place of incident]

For the person Andreas Angelakis the search result is

Angelakis Andreas Perivolia, Chania 15 Mar 1918 Samos.

Creating the title mask

Record Type Title Mask Edit

The title mask builds a composite title from fields in the record (see manual). Field values identified by [ ] are inserted into the mask, e.g. *[Title], pp. [Start_Page]-[End_Page]* might generate: *"Alice in Wonderland, pp. 37-39"*

The element names in square brackets should match field names for this record type.
The constructed value is used as the extended title displayed in search results and other lists.
To insert a literal square-bracket, use two consecutive square-brackets ([[or]]).

Add optional text if a value is not available by adding {\Text for existing value \Text for missing value} after a field.
e.g. *[Starting_date] {\Starting date of: \No starting date}* will either generate:
"Starting date of: 04-11-1974" or *"No starting date"* if there isn't Start_date is empty.
Additional add a character cap to field values by adding the desire amount within the optional text like so,
{50 \Text for existing value \Text for missing value} will cause the value, if it exists, to show 50 characters max.

Select fields to insert
(add to record structure first)

☐ Select All Visible checkboxes
▷ *metadata*
◢ fields
  ☐ Surname
  ☐ Given names
  ≡ Gender
  ≡ Family status
  ≡ Education level
  ☐ Occupation
  ▷ *Related Person(s)*
  ☐ Year of military class
  ≡ Enlistment
  ☐ Served since
  ≡ Rank
  ☐ Regiment
  ▷ *Place of origin*
  ≡ Evidence
  ☐ Date of incident
  ☐ Illness
  ≡ Type of incident
  ▷ *Place of incident*
  ☐ Mappable location (geospatial)
  ≡ War period
  ▷ *War*
  ☐ Short summary

Build Mask: [Insert fields] *Select fields in tree and click button to insert in mask at cursor position*

For legibility we strongly advise splitting field specifications onto separate lines (which will be combined in the result)

[Surname] [Given names] [Place of origin..Primary place name] [Date of incident] [Place of incident..Primary place name]

Test mask

select from records of this type (if any exist) ... ▾

[Test]

[SAVE MASK] [Cancel]

Two more fields with distinct properties: Time

Two more fields with distinct properties: Place

You will often return to design mode to modify the structure

Modifying entity structure

Explore menu: Using filters

A. Existing filters

- All (By date)

- Entities

- Saved filters

B. Building filters

- Simple filter builder

- Facets builder

Explore menu: Results display

- Records display

- List

- Map and timeline display

- Report building

- Export

- Network

- Crosstabs

Heurist: Search I

Heurist: Search II

Where is my data?

A. Heurist is a web application installed on two servers (in Australia and France), accessible

- through most browsers (Chrome, Firefox, Safari)
- from many computers and by many users.

- It is upgraded automatically and centrally.
- It offers increased security specifications (backup/recovery, malware protection).

B. It is possible to install Heurist on another, e.g., institutional or commercial

server, with all that this entails in terms of security, cost and maintenance.

C. A full backup can be downloaded to a local computer.

Just before moving to hands on

Support:

- https://heuristnetwork.org/
  - Tutorials
  - FAQ
  - Heurist Help System
- https://heuristnetwork.org/community/
  - Mailing list
  - User groups
- https://heurist.readthedocs.io/en/latest/index.html

HANDS ON: Exercise with pencil and paper - A

If you don't have a research subject, use the following example "Ibsen plays". Then Draw on paper these entities and the relationships between them.

Author: Henrik Ibsen

Birth date: 20 March 1828

Death date: 23 May 1906

Birth place: Skien

Death place: Oslo (Kristiania)

Play Name: *Bygmester Solness*

Composition date: 1890

1st performance date: 19 January 1893

1st performance place: München

Type of play: Drama

Characters:

Halvard Solness, master builder

Aline Solness, wife of Halvard

Doctor Herdal, physician

Knut Brovik, formerly an architect, now in Solness's employment

Ragnar Brovik, son of Knut Brovik, draftsman

Kaia Fosli, bookkeeper

Hilda Wangel, 23-year-old young girl

HANDS ON I: Exercise with pencil and paper - B

If you have a specific research subject, select 3-4 entities with a few fields each. At this stage do not make your base more complex.

• Make a list of entities and main fields (not exhaustive).

• Include fields that create relationships and note their relationship types (record pointers, child records, record relationships)
• Include the target entities that these fields link to
• If you have a vocabulary, list some vocabulary terms as examples.
• Define the field types (text, date etc.) and the properties of these fields (single/ multiple)

Draw on paper these entities and the relationships between them.

Registration: **https://heuristnetwork.org/**

https://heurist.huma-num.fr/HEURIST/heurist/startup/

Designed by researchers, for researchers, Heurist reduces complex relational structures to simple, logical choices and provides comprehensive tools to collect, manage, analyse, visualise, export, publish and archive information.

Heurist Network website

Set Up a New Database

Create your first database on this Heurist server (**int-heuristweb-prod.intersect.org.au**) by registering as a user.
As creator of a database you becomes the database owner and can manage the database and other database users.
For more information on Heurist see Heurist Network website

New Users

Please register in order to define the user who will become the database owner and administrator.

[Register]

Existing Users

If you are already a user of another database on this server, we suggest logging into that database and creating your new database via the Administration menu, as this will carry over your login information from the existing database.

Find your database

[   ] [GO]

You will be redirected to the Heurist database upon your selection

See all databases on server

DATABASE NAME
TEMPLATE

hdb_(name or initials)_(database name)

Using Heurist: Exercise 1

Design menu

1. Record types

Browse, select, and rearrange the groups.

Edit the record type **person** (edit fields):

> Given names: make it a repeatable field
>
> Person type (terms): Add a term to the vocabulary
>
> View the properties of the fileds(Temporal, pointer fields, record relatioships, vocabularies)
>
> Modify the position/ delete fields
>
> Edit mask: Add another field to the title mask.

Repeat the same exercise with the record type **place**

1. Add record type

> Give it a name and select a symbol
>
> Follow the instructions

Using Heurist: Exercise 2- Implementing your model

Begin with Design menu

Select one of the existing Record types and/ or create new ones according to your needs

Edit existing fields or add new ones.

Create the title mask for each record type.

If you created your database according to the example, the model should look like this:
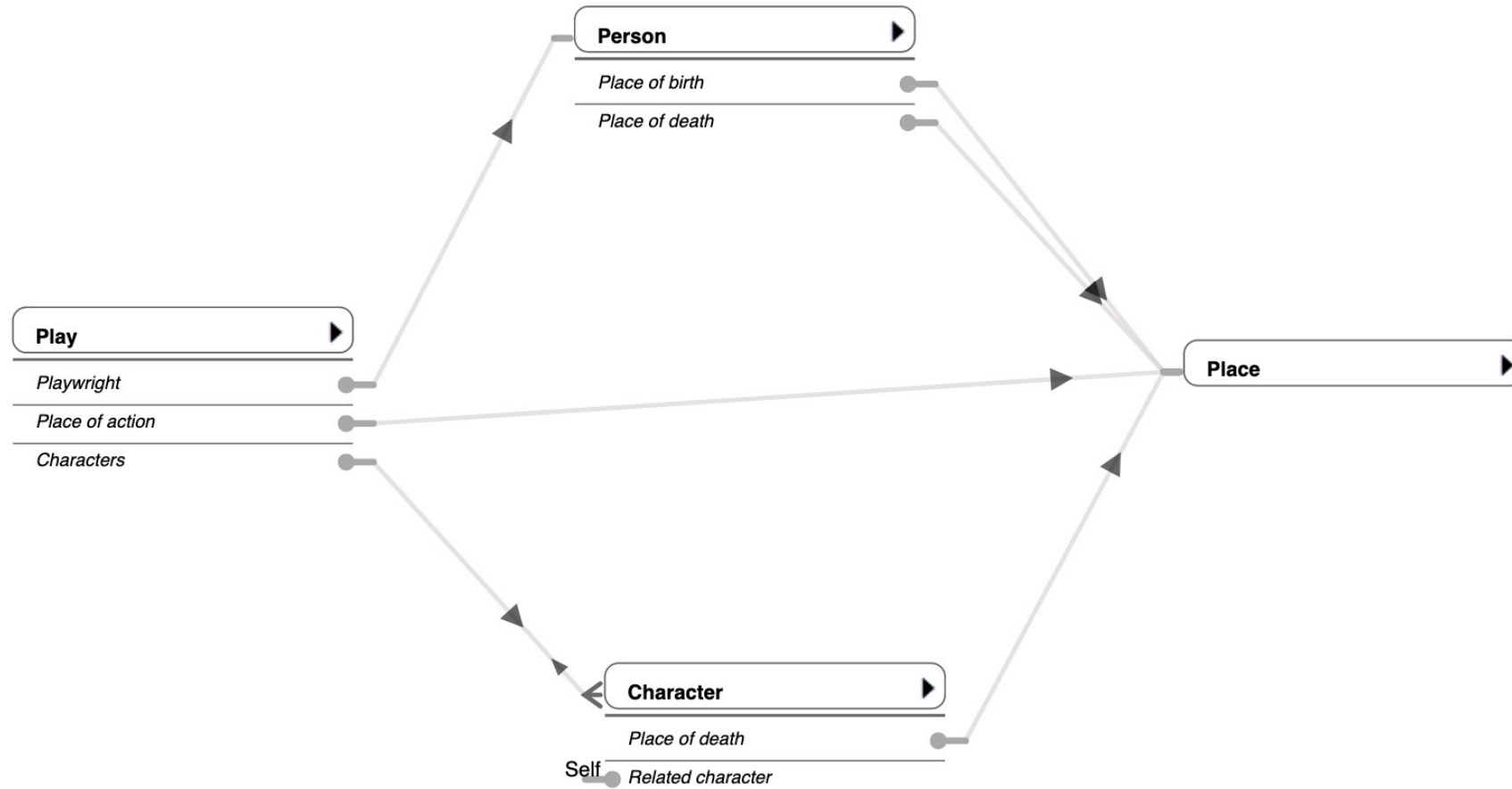
❋ ❋ ❋ ❋ ❋

ΑΚΑΔΗΜΙΑ   ΑΘΗΝΩΝ

ΨΗΦΙΑΚΟ ΤΟΠΙΟ
στις Ανθρωπιστικές Επιστήμες

ΕΛ.ΙΔ.Ε.Κ.
Ελληνικό Ίδρυμα Έρευνας & Καινοτομίας

Visualization

Using Heurist: Exercise 2- Implementing your model

Go to the **Populate menu**

And use the **New record** button

Select the record type for the new record(s) and add a few records. If necessary, select the **Modify structure** button to modify its structure. Pressing the **Save** button is mandatory. Saving is not automatic.

One of the record types you should chose should be **Place** in order to see how the **map digitizer** works.

❄❄❄❄

From the **Explore menu** perform a few simple searches to see how the results appear in the result windows.

Happy experimenting

Thank you

Helen Katsiadakis

hkatsiad@academyofathens.gr