

# ULRR

## CensusIRL: Historical census data preparation with MDD support

|               |   |
|---------------|---|
| Item Type     | Meetings and Proceedings  |
| Authors       | Doherty, Adam J.;Murphy, Rachel A;Schieweck, Alexander;Clancy, Stuart;Breathnach, Ciara;Margarita, Tiziana                    |
| Citation      | 2022 IEEE International Conference on Big Data (Big Data), Osaka, Japan, 2022, pp. 2507-2514                                  |
| Publisher     | IEEE Computer Society   |
| Download date | 2026-04-22 08:16:03   |
| Item License  | <a href="https://creativecommons.org/licenses/by-nc-sa/4.0/">https://creativecommons.org/licenses/by-nc-sa/4.0/</a>           |
| Link to Item  | <a href="https://doi.org/10.34961/researchrepository-ul.25370464">https://doi.org/10.34961/researchrepository-ul.25370464</a> |

# CensusIRL: Historical census data preparation with MDD support

Adam J. Doherty<sup>\*†‡</sup>, Rachel A. Murphy<sup>\*</sup>, Alexander Schieweck<sup>\*†</sup>, Stuart Clancy<sup>\*</sup>,  
Ciara Breathnach<sup>\*‡</sup>, and Tiziana Margaria<sup>\*†‡</sup>

<sup>\*</sup>University of Limerick, Ireland - {name.surname@ul.ie}

<sup>†</sup>Lero, the Science Foundation Ireland Research Centre for Software, Limerick, Ireland

<sup>‡</sup>HRI, Health Research Institute, University of Limerick, Ireland

**Abstract**—Census returns are a critical source of information for governments globally. They underpin a wide spectrum of public planning including health, housing, work and education. Historically, census forms have captured names, places, dates, age, occupation, family structure, and religion. In more recent times, sexual orientation and ethnicity, queries that can be intrusive to vulnerable communities, have been added to the criteria, and for such reasons data security is of paramount importance. Most governments restrict access to individual census returns, presenting the data in aggregate report format. The Irish government is particularly strict, enforcing a statutory closure period of 100 years. An exception was made for the Irish 1911 census which were digitised and released for free online consultation in 2009 [1]. They are an excellent source for genealogists and historians alike but exist as separate digital siloes. This project uses an eXtreme Model-Driven Development (XMDD) environment to create linkages between both datasets. It will discuss the development process of the CensusIrl application and the process used in developing the matching algorithm used. We will discuss the census records and the data cleansing process used in creating the initial proof of concept application. We detail the different approaches to the development life-cycle of the application and describe the different utilises used in the sanitation of data points in the records and the match-making process.

**Index Terms**—Census Data, Model-Driven Development, PBL, Digital Humanities, Optimization, Low-code Application Development

## I. INTRODUCTION

The first attempt of a full census of Ireland was undertaken in 1821, religious parish-based enumerators collated the information on paper returns. Censuses were taken on a decennial basis thereafter until 1911, and were collated primarily by the police forces. There was no 1921 census due to the War of Independence and Civil War, so the next census was conducted by the Irish Free State in 1926. Unfortunately for historians, the only complete manuscript census returns that exist for the period 1821-1911 are the 1901 and 1911 censuses. This is because the remaining returns were either partially or completely destroyed. The 1861 and 1871 censuses were destroyed by government order, the 1881 and 1891 censuses were pulped during the First World War and, other than a few fragments, the remaining nineteenth century censuses were completely destroyed during the 1922 Public Record Office fire during the Irish Civil War in 1922.

The fully digitised 1911 census was made available to the public in 2009 [2], with the 1901 census following in 2010. These scanned records are now publicly available on National Archives of Ireland’s Census of Ireland website [3]. The records can be searched in multiple ways rendering them a key source for historians and genealogists. The availability of both the 1901 and 1911 censuses allows for record linking techniques, in this case ‘census matching’. This is conducted manually, and is time-consuming and complicated work, particularly in Ireland where the traditional naming-pattern means that a number of people with the same name can live in close proximity or in the same household [4]. While the search functionality within each census is good, there is not as yet any means of searching between censuses, or an automated way of tracing individuals across both records and verifying their identity.

This paper discusses the early stages of the development of an application that will analyse and link data between censuses. Using cutting edge XMDD technologies we will create a “digital thread” for corresponding census entries, presenting historians and genealogists with certain or very likely matches. This application will be scalable, so that when the long-awaited 1926 census of Ireland is made available to the public, it can be incorporated with little to no development, using our low-code/no-code approach.

The remainder of this paper discusses the first steps towards the development of this application. In [section II](#) we provide a literature review relating to historical data linkage. [section III](#) introduces the CensusIRL web-based application. In [section IV](#) we will detail the development process and the different approaches taken in the initial stages of the application. Then, [section V](#) describes the current outcomes and discusses our experience while analysing and using the national archive census records. Lastly, [section VI](#) summarises our findings to date and indicates some future plans.

## II. RELATED WORK

Data linking is not a new technique; it is an approach that historians and genealogists have always taken, but in a manual way to understand the life course of individuals. Such information can provide more understanding about household structures, health, etc etc. In an Irish context Guinnane (1992) took this approach to understand age at leaving home, while Fitzpatrick (1983), and Gibbon and Curtin

(1978, 1983) used census matching to understand household structures and prevalence of the stem family [5]–[8]. Since the advent of computing, historical demographers, economic and social historians have sought ways of automating data linking and record matching. Historical demographers led the way with Fleury and Henry compiling a manual for reconstituting families. The Cambridge Group for the History of Population and Social Structure (CAMPOP) [9] was one of the earliest adopters of computing in historical data studies. In 1964 it embarked on a large-scale family reconstitution study using volunteers to link baptisms, marriages and burials. Information was entered into an early computer system after being punched onto paper tape. Subsequently it was entered into a relational database, and more recently an Intermediate Data Structure (IDS) was applied to migrate the project to more modern methods and enable further research on fertility [10]. IDS is an internationally agreed standard that organises individual level data in a uniform way in preparation for further analysis in statistical software [11].

These efforts represent a brief example of valiant attempts to standardise formats over the past few decades using relational database techniques and the semantic web. *BurgerLinker* is a data format developed by CLARIAH (Common Lab Research Infrastructure for the Arts and Humanities) [12]

#### A. *Linked Data*

**Artificial Intelligence:** More recent developments include the use of machine learning to create linked data between censuses. Feigenbaum (2016) uses machine learning to create a linked sample across the 1915 Iowa State Census and 1940 Federal census [13]. Bailey et al (2020) considered the performance of record-linking algorithms, noting high error rates and highlighting the potential consequences of linking errors [14]. However Abramitzky et al (2021) argue that automated methods perform well [15].

*BurgerLinker* [12], which is used on Dutch civil registration data, is one of the most advanced platforms in historical population studies but is not applicable to the condition of Irish data which has not been codified to international standards. For example, in the *BurgerLinker* implementation tool, the developer is able to input constraints into the matching algorithm because the condition of the underpinning data is in a standardised smooth data format. Operating from a different data base, we argue that Low code environments in XMDD offer new departure. In the case of the *CensusIRL* application, developer constraint inputs are not required as the application only uses the data to infer information on the person. Another key note is that this is a software package that can be used by developers while the *CensusIRL* web based application that would be publicly available for use without a need for the user to be able to develop an application around the software package. An example of this but in a different context is the transcription of the General Register Office records [16] where they have developed a web based application for that enables the transcription of the death and burial records into a digital format. We were very successful in the development

of HDA, the *Historian Dime* Application, and its use in the context of a series of transcriptions that gave students of several cohorts of the Master in History of Family at the University of Limerick the opportunity to do field work during transcription projects [17], [18]. In fact, we organized an entire session at the recent 4th Conference of the European Society Of Historical Demography in March 2022 dedicated to this interdisciplinary work. The application we purpose will use the same rapid modelling framework as this work but aim of this work was to digitize records that have yet to be digitized using object character recognition as an aid to those partaking in the transcriptions. One of the key differences between these related works and the *CensusIRL* application will be the uses a single set of records instead of accessing multiple different dataset to use in our application. As the census records in Ireland have continued to be captured, digitised and made publicly available after 100 years since the year the census was taken, the *CensusIRL* application is dependant on a continuous growing dataset. Another aspect is that instead of using multiple sources of information to attempt to trace a person through history, our application uses the single source of data and infers identifiable factors about the person in the period since the initial census was taken. Throughout our research into the topic, we are yet to discover another application that applies the same process. During the development of the *CensusIRL* application, we have followed a number of the same principals depicted in previous Computational Archival Science journals to uphold the integrity and the long term preservation [19] of the digital assets and through the use of the unique graphical development approach, maintained understandability of processing being done to create the *CensusIRL* application.

### III. THE *CENSUSIRL* APPLICATION

The *CensusIRL* application has been developed using XMDD technology, specifically a low code development environment. We have chosen *DIME* [20], a low code development environment which enables the rapid prototyping and development of web based applications. *DIME* applications are generated from a number of distinct model types, and the resulting code is deployable in a variety of runtime environments. This specific variant of Model Driven Development is called *eXtreme Model Driven Development* [21], [22], and it follows a Domain Specific Languages approach both at the level of the modelling languages (LDE, [23]), and at the level of the application domains. Prior experience in bioinformatics [24], [25], telecommunications [26], [27], smart manufacturing [28], [29] and geoinformation systems [30], [31] have validated the approach, that we now apply to the domain of analytics of historical data sources. *DIME* provides a model-level lingua franca for the orchestration of domain specific components, and it overcomes the diversity of the underlying runtime technologies by using *Docker* [32], a widely used virtualisation platform that abstracts away these differences and supports deployment on most environments. Application Development in *DIME* does not follow the prevailing text-based coding approach. Instead, development takes place in a graphical

environment, where the application is assembled by composing preexisting reusable building blocks in a drag and drop fashion. This development lends itself to the co-design of applications with application domain experts, in our case the team of historians in the Death and Burial Data: Ireland 1864-1922 (DBDIrl) project.

In DIME, three main Domain Specific Languages (DSLs) express the Data model, the Process models, and the Graphical User Interface (GUI) models. Each of these model types is responsible for its specific aspect of the development of a web based application.

**Data model** The data model is a representation of the data structures used in the application, as well as their relations. It is similar to one or more table schemata in a relational database, or a class diagram in object oriented programming. These data structures are modelled in terms of elementary and complex data types, and used throughout the application to represent and share information. Data types within the data model can have primitive attributes, like numbers or text, as well as uni- and bi-directional entity relationships between the custom data types that have been defined for the specific application (see in Fig. 2 the complex census record data type).

**Process models** Process models are used to define and encapsulate the business logic. They are graphical definitions of the control and data flow within an application. A process model is composed of a set of Service-Independent Building Blocks (SIBs), which encapsulate lower level actions into a graphical representation. DIME itself has a rich collection of built-in common SIBs. Additionally, a process model can become itself a (hierarchical) process SIB, so that it can be described once and then reused in other model throughout the application (see Fig. 3 for an example of hierarchical blueprint process model).

**GUI models** The GUI models in DIME describe the user interface view of an application. DIME GUI development follows a "What you see is what you get" approach to defining a user interface, based on elementary and complex components and containers. In fact, complex user interfaces are designed by encapsulating reusable components into groups (like a table or a list), so the hierarchical approach applies to the GUI models as well, allowing for rapid prototyping. (see Fig. 1 for an example of the live web application developed through the MDD modelling environment)

**Native SIBs** Custom SIBs are developed in Java and then "published" via a textual DSL as collections of ready-made SIBs, to be included inside process models. This integration of functionalities developed in native code (here Java, but it could be Python, R or any other language) makes the inclusion of external services and libraries extremely easy during the development lifecycle, as native external SIBs. The important aim of this integration mechanism for external functionalities and components is to create easily reusable services that can be shared

through the project, the application domain, or even in completely different application domains. Next to using the DIME common SIBs, the native SIBs developed for the CensusIRL application deliver various different functions, such as the ingestion of the Census records from a CSV format to DIME modelled objects, and the optimised filtering mechanism used in the process that finds corresponding records in the different censuses.

In the current iteration of the CensusIRL application, we have implemented a minimal user interface that can be seen in Fig. 1. This simple GUI aims to allow views into the Census records that aid the users in their understanding of the matching process. The small set of views currently available in the application will be extended once the initial proof of concept application has been validated.

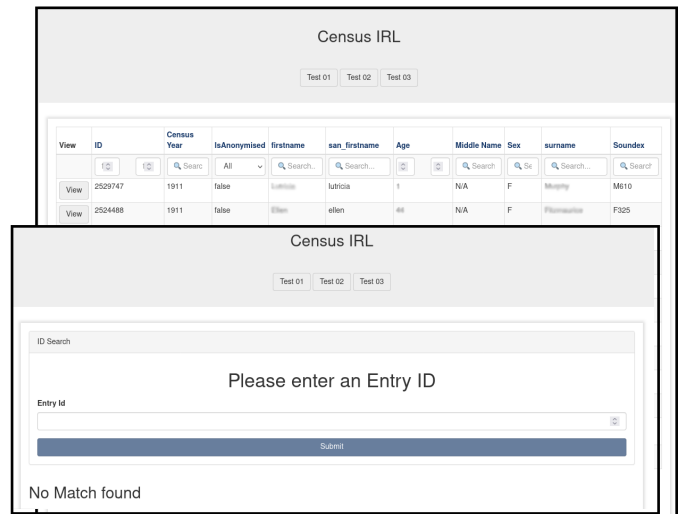


Fig. 1. Views of the live user interface of the CensusIRL web application: query by Entry ID and response

#### IV. THE DEVELOPMENT OF CENSUSIRL

By choosing an eXtreme Model-Driven Development (XMDD) approach and DIME as the low-code development platform, we ensure that the CensusIRL application is designed, prototyped and implemented in a quick fashion with minimum cost to the project. Approaching application development in this fashion enables us to collaborate and co-design it with experts from other domains. This rapid and continuous feedback loop while developing is very helpful to the computer scientists, enabling us to accurately develop an application in an agile fashion.

##### A. Census Record Modelling

The data set provided to us by The National Archives of Ireland consists of the entirety of the records taken from the 1901 and 1911 censuses. These records consist of forms in which data has been entered manually. As part of the census digitisation project they were subsequently scanned and stored as images. The records were then transcribed in digital format,

these transcriptions being provided to us in CSV format. The initial step of the application development concerned creating a data model that accurately describes both census record formats in our use case, for the 1901 and 1911 census, while still being able to differentiate the records. Census design evolved over time, so there is variation in data gathered in each census. For example, the 1901 census includes marital status, but the 1911 census required more information on this data point. We sought to build a generalised model that covered data included in both censuses, while being aware that it will need to be extendable to accommodate future censuses as they are published. The records report the information taken at the time of the census for all households in Ireland. To aid in the rapid design and prototyping of the application, we extracted a subset of the datasets specifically from the county of Limerick. In the early twentieth century, Limerick county experienced a relatively low ratio of outward migration, making it highly suitable for our use case: trying to match individuals across both census instances. The 1901 dataset consists of 20 data points per record, covering the following categories:

- Name and Surname,
- Relation to Head of Family,
- Religion,
- Education,
- Age,
- Sex,
- Rank Profession or Occupation,
- Marriage,
- Where Born,
- Irish Language,
- and any specified illnesses.

. Additionally to these categories, we were provided with the full addresses for each entry. The 1911 dataset consists of the same data as the 1901 census, with additional data points to further detail the person's marital status, de facto describing the family circumstances. These data points define

- information on the length of a person's marriage,
- the total number of children born alive, and
- the number of children still living.

Fig. 2 shows that these fields become attributes in the corresponding DIME data model.

We encountered a number of difficulties while working with the data set.

Taking into account the time when these records were initially gathered and the context of the time, the literacy of the person recording the data on behalf of the households literacy plays a large role in the correctness of the data, as does the lack of standardisation of data that was entered into the forms. As an example, the 1901 Co. Limerick subset presents approximately 251 different representations of the Roman Catholic religion. Data concerning age is another issue; at this time birthdays were not celebrated as they are now, so many people did not know their exact date of birth, providing just an estimate of their current age.

In addition, the introduction of the old age pension in Ireland in 1908 [33], led to a spike of individuals that declared themselves above that age threshold. Another difficulty relates to some entries having been anonymised due to the person's situation at the time. These records include soldiers and policemen, as well as individuals located in institutions such as convents, workhouses, asylums and prisons. The full names of such individuals are only represented by initials, but these records still contain information other information including age, religion, profession and birthplace. Such anonymisation makes the process of verifying a match in the subsequent census much more difficult. A related consideration is that a person could be anonymised in one census return, but not in the previous or subsequent one. For these reasons, we have chosen to exclude the anonymised records from the sample used in our initial proof of concept.

Fig. 2 shows the initial generalised model we developed to represent entries in both census records. This model consists of any data point provided on a person in either the 1901 or 1911 records, with additional attributes, like the census year, that distinguish the different census records. By utilising these additional data points we distinguish the different attributes available in each record to express the information gathered at the time of the census. This initial data needed to be extended during the development of the proof of concept to aid in the data cleansing and matching process.

### B. Matching process

The process of the initial approach to records matching records is reported in Fig. 3. The entire data set of 1901 and 1911 census records is ingested. The process cleanses each data point, then extrapolates from this cleansed data model a search model whose goal is to depict what details about an entry we can infer knowing certain variables. For the search and matchmaking we have two main pieces of information to work from:

- 1) the current census record for the person,
- 2) the time between the current census entry and the next census where we aim to find this person.

This approach revolved around the concept that we might infer information about a given record that could be used to find this person in the following census, and further to the nth census. Using a search model is a standard practice in many enterprise grade application development frameworks [34]: it aids in encapsulating the search process of records in relational databases without having the full context of the record one is searching against. As we are attempting to find the matching entry in the following censuses, this approach of attempting to infer the possible "next state" of the person in question was the most fitting for our initial approach. After prototyping this approach, we found that we needed a much more stringent cleansing process in order to retrieve good results from the search process. This discovery led to the use of multiple utilities when attempting to generalise the information provided in the census records.

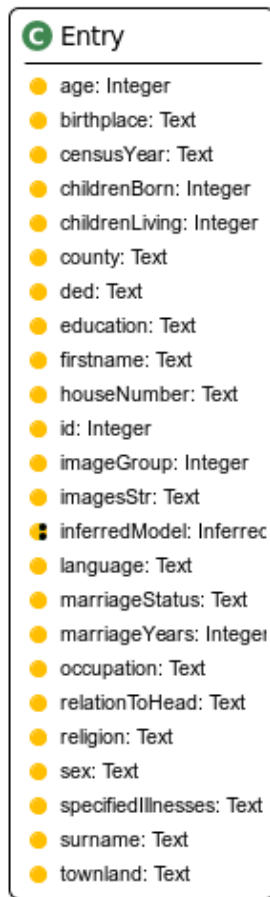


Fig. 2. The DIME data model of the census record entry: a single class represents both the 1901 and 1911 record schemata

### C. Data Generalisation

Matching names, specifically first names and last names, incurred difficulties.

a) *First Name field*: One of the main issues we encountered was the use of different variations of the same first name. An example of this is the name Margaret, which presents many variants, like Maggie and Peggy, as well as abbreviations such as Mgt and Margt. When using the direct exact matching approach initially planned, such variants can lead to a mismatch causing people to be discounted from the search process. This observation led to the use of an online resource that provides alternative spellings and variants for Irish first names [35]. This utility was implemented in the ingestion process, to return a generalised first name for an entry.

b) *Last Name field*: Similar inconsistencies occur with the last name data point. Inconsistent spellings of a surname can arise due to low literacy levels of a family; where an individual was not literate an enumerator might make a guess at spelling the surname; and finally, sometimes surnames were added to or simplified e.g. O’Sullivan, O Sullivan and Sullivan might all refer to the same surname. These surname variants resulted in inaccuracies in the matching algorithm. To address

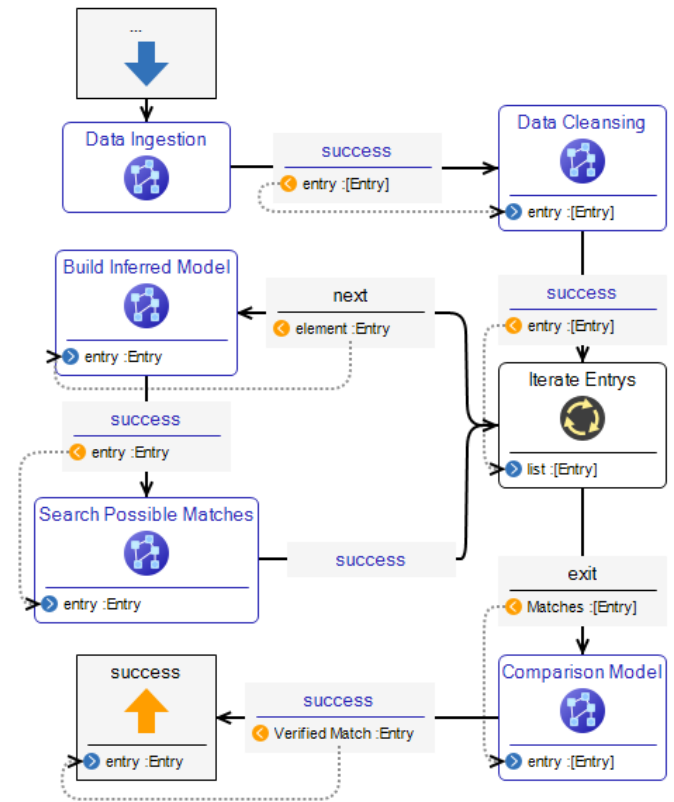


Fig. 3. A high level overview of the search process in DIME

this we used Soundex [36] to generalise the data, and found this quite effective. The most common alterations seen from our testing involve the vowels in the last name. Soundex takes in a (sur)name and returns a four character code representing that name based on the consonants in the name. This provided us with good generalised dataset of last names to search against.

### D. Inferred Model

With the growing need for sanitised information to be stored at each entry, we needed to uphold the integrity of the data provided. This led to an evolution of the data model: we created another complex object to aid the search model when inferring information about the current entry. Fig. 4 shows the inferred model built from the current entry. This model has a bidirectional association with the entry model. This means that at any point the inferred model can find and dynamically evaluate the information based on the current Entry model. Fig. 4 contains several types of attributes, with an icon that is different from that of the primitive types. Some of these new attributes are not the usual statically allocated attributes: they are full process models that evaluate information based on the inferred model or the base Entry model. For example, the value of the attribute labelled "canBeMarried" is computed by a process that was created to address the issue of women's surnames changing upon marriage. In the absence of civil records to check if a person has married, we need to include this possibility in our matching process: if we are unable to

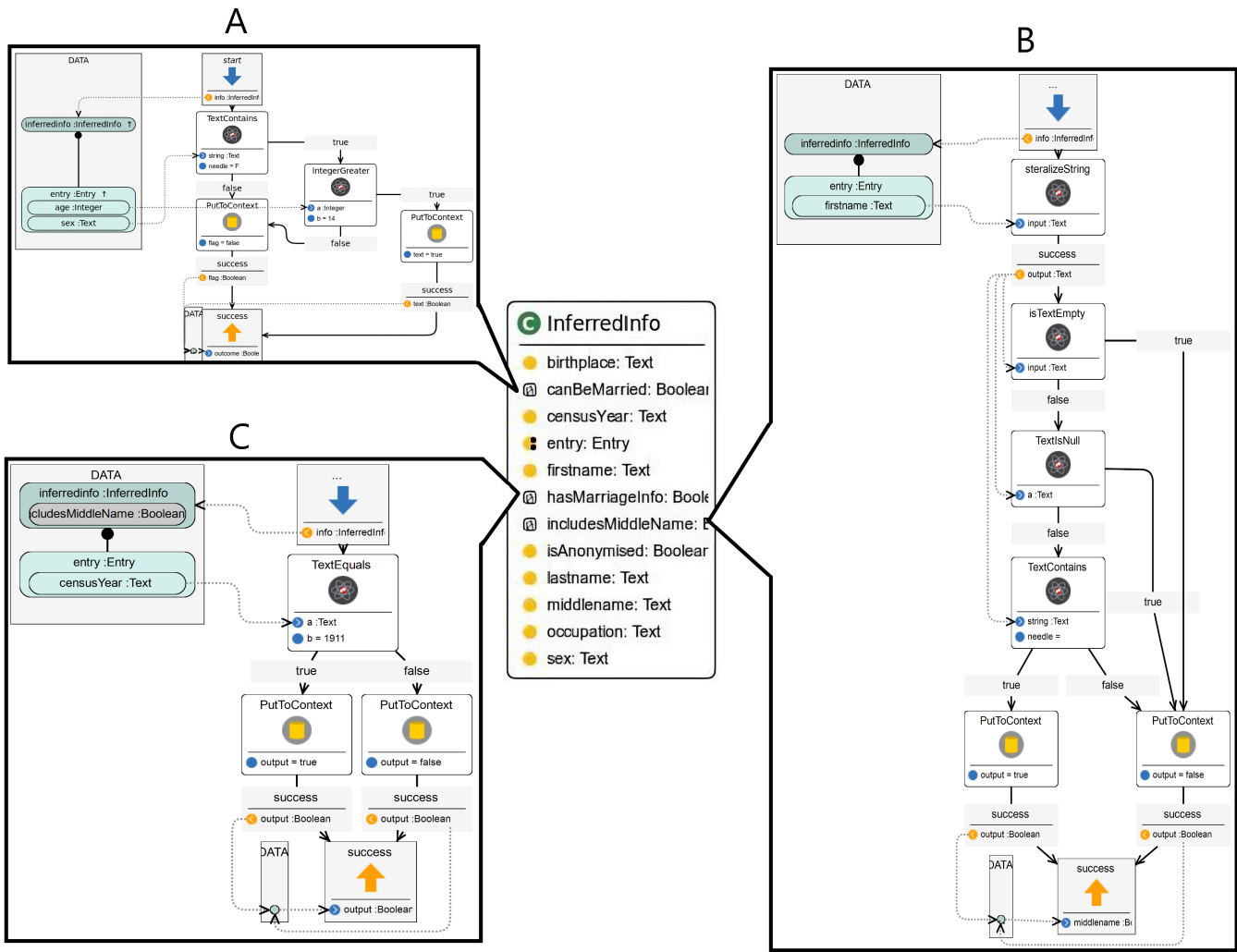


Fig. 4. The inferred information model in the DIME Development environment

find a person through the initial search process, on the second iteration we can check if this individual may have married, resulting in a surname (and therefore Soundex code change). Fig. 5 shows the process model used to infer whether a person could have been married since the last census. As the legal marriage age at the time was 14, if a female is above this age and cannot be located in the initial search, the search process will exclude the Soundex filter from the matching process.

#### E. Expanding the Search Model

In the initial stages of the search model, the process would run every filter, and if nothing was returned, then we would assume that the person was not inside our subset of the census records. This assumption proved to be incorrect. Instead, the more likely reason for mismatch was due to slight inaccuracies in information given in the census records. To account for this, if the current process returns no results, instead of terminating its search, it will first check if the person's last name could have changed in the interim of the Census, and if so, the process will search again without taking into account the soundex attribute.

If there is no possibility of a last name change, we then widen the lower and upper age bounds and attempt the search again. This process is currently set to widen the bounds for given search statically, but in future iterations of the CensusIRL application we would aim to do this dynamically.

#### F. Next Steps for the CensusIRL Application

The next steps for the proof of concept application will be to implement a weighted verification model to shorten the list of possible matches down to a single possible match. Implementing this verification step will require some time and fine tuning to assure that the chosen weights are appropriate for the data point. The weight fine tuning may be initially manual, and subsequently handled by some form of machine learning model. This would require us to manually create a labelled data set of matched census records, to be used to dynamically apply the different weights to each data point. Further classification is required for the data points available in the Census data set. As mentioned in section IV above, there is a number of hurdles to pass in regards to the data classification. Further research

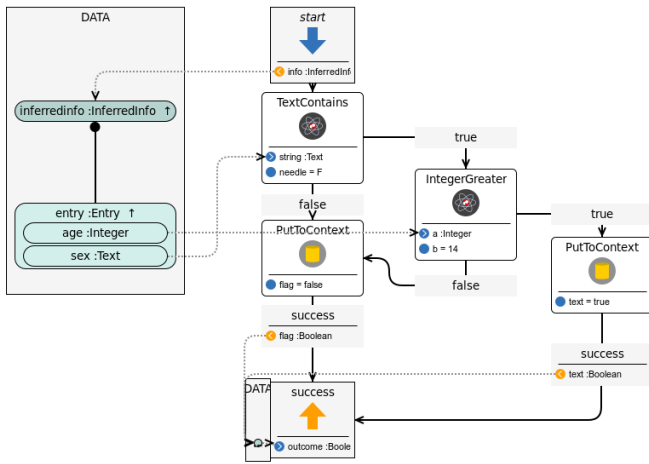


Fig. 5. A view of the canBeMarried process seen in Fig. 4

into different language models similar to BERT [37] seen in previous workshops will be needed to assert its applicability to the data points in the Census data.

## V. KEY OUTCOMES AND LEARNINGS

The key outcome is the proof-of-concept application, CensusIRL. Even though in its infancy, it is able to search a census record and provide a list of possible matches in the corresponding census records. A further verification system is required to further refine the list of possible matches and provide a more accurate match for each record. Outlined below are some key learnings from the development process of the CensusIRL application.

*a) Data Cleansing process:* Initially we assumed that transcribed census data could be used directly in the matching process, but we found this assumption to be incorrect. The level of literacy at the time means inconsistencies in the original records. This, combined with lack of input validation, meant there were a number of issues in working with uncleaned data. We found that most personal data required some level of cleansing/generalisation to enable these data points to be used in the matching process. An example of this is the Occupation data point. Enumerators were provided with a list of occupation classifications, and we have used these to classify the occupation data point. While it was common for a person to have the same occupation for their entire working career, particularly in a rural environment, we need to consider likely promotion stages (e.g. from private to lieutenant), movement to allied jobs, or alternative descriptions of the same occupation (e.g. labourer, general labourer). Furthermore, women and children are often described in terms of the occupation of the head of house (e.g. farmer’s wife, baker’s daughter). Further research is required to determine the most appropriate way of classifying this data point before it can be added to the search model.

*b) Data set magnitude and runtime:* The initial subset of the dataset provided for this application consisted of 74,155 entries derived from 1901 and 1911 census records. DIME

uses a Postgres database to store an application’s data in the different types of data structures described in the data model. This is not a large number of rows for a Postgres database to hold. The issue we faced revolved around the processing of this dataset. DIME uses JavaEE to implement the business logic described in the process models. In the initial phase of the application design, we attempted to execute a set of processes on the dataset to build the inferred information model. As mentioned above in subsection IV-C, we use a utility that contains the list of generalised names. At application start, we ingested this utility into the data model so that we could access this resource at any time during processing. The issue occurred when we attempted to use this resource thousands of times in the iterative search model: there were thousands of calls to the Postgres database, which incurred a lengthy delay and the process thread timed out causing the deployment of the application to fail. To resolve this issue we implemented the name utility as a statically allocated resource. This enabled us to use this name utility in the data cleansing process and the search model within incurring additional calls to the database.

## VI. CONCLUSIONS

This paper has demonstrated our iterative approach and the development lifecycle of the current version of the CensusIRL web-based application. Preliminary results of the matching algorithm based on a subset of the Census data show a promising outcome for the CensusIRL application and its next iterations. This current version has an evolved data model that dynamically evaluates complex attributes thanks to built-in processes, but it reaches the limits of the JavaEE execution environment because the computations are not pooled. One of the main objectives of the next iteration of the application will be to address this issue through the use of multi-threaded or pool processing to resolve this computations bottleneck. As an overarching goal for the application, we aim to be able to ingest the nth Census into the application without the requirement for alterations to the data model. Further collaboration with domain experts will be needed to model a generalised representation of a census entry that would satisfy the requirement for the evolution of Irish Census data capture. As we can see from the 1901 and 1911 Census dataset described in this work, in one iteration of the Census there is an additional three data points added to the data capture. The next steps of the iterative evolution and refinement need to make the app more responsive, and prevent more errors in the estimate of potential matches. We aim to achieve this by implementing a weighted comparison model to assert the likelihood of an accurate match.

## ACKNOWLEDGMENT

“Death and Burial Data: Ireland 1864-1922” is a project funded by an Irish Research Council Laureate Award IRC/CLA/2017/32.

This publication has emanated from research supported in part by a grant from Science Foundation Ireland under Grant number 13/RC/2094\_P2, and a cooperation within the Health Research Institute (HRI) at the University of Limerick.

For the purpose of Open Access, the authors have applied a CC BY public copyright licence to any Author Accepted Manuscript version arising from this submission.

## REFERENCES

- [1] P. Cullen, "Millions of files offer historic snapshot: 1901 census goes online," *Irish Times*, Jun 2010.
- [2] G. Carbery, "Detailed search of 1911 census goes online," *Irish Times*, Aug 2009.
- [3] "Census of ireland 1901/1911 and census fragments and substitutes,1821-51."
- [4] R. Breen, "Naming practices in western ireland," *Man*, pp. 701–713, 1982.
- [5] T. W. Guinnane, "Age at leaving home in rural ireland, 1901–1911," *The Journal of Economic History*, vol. 52, no. 3, pp. 651–674, 1992.
- [6] T. Walch, "Book review: The vanishing irish: Households, migration, and the rural economy in ireland, 1850–1914," 1999.
- [7] P. Gibbon and C. Curtin, "The stem family in ireland," *Comparative Studies in Society and History*, vol. 20, no. 3, pp. 429–453, 1978.
- [8] P. Gibbon and C. Curtin, "Irish farm families: facts and fantasies," *Comparative Studies in Society and History*, vol. 25, no. 2, pp. 375–380, 1983.
- [9] "The cambridge group for the history of population and social structure, cambridge."
- [10] G. Alter, G. Newton, J. Oepfen, *et al.*, "Re-introducing the cambridge group family reconstitutions," *Historical Life Course Studies*, vol. 9, pp. 24–48, 2020.
- [11] G. Alter, K. Mandemakers, *et al.*, "The intermediate data structure (ids) for longitudinal historical microdata, version 4," *Historical life course studies*, vol. 1, pp. 1–26, 2014.
- [12] J. Raad, "burgerlinker," Feb 2021.
- [13] J. J. Feigenbaum, "Automated census record linking: A machine learning approach," 2016.
- [14] M. J. Bailey, C. Cole, M. Henderson, and C. Massey, "How well do automated linking methods perform? lessons from us historical data," *Journal of Economic Literature*, vol. 58, no. 4, pp. 997–1044, 2020.
- [15] R. Abramitzky, L. Boustan, K. Eriksson, J. Feigenbaum, and S. Pérez, "Automated linking of historical data," *Journal of Economic Literature*, vol. 59, no. 3, pp. 865–918, 2021.
- [16] A. Schieweck, R. Murphy, R. Khan, C. Breathnach, and T. Margaria, "Evolution of the historian data entry application: Supporting transcriptions in the digital humanities through mdd," in *2022 IEEE 46th Annual Computers, Software, and Applications Conference (COMPSAC)*, pp. 177–186, IEEE, 2022.
- [17] C. Breathnach and T. Margaria, "Census: on paper, by governments, is still best," *Nature*, vol. 577, no. 7789, pp. 170–171, 2020.
- [18] M. H. Ter Beek, A. Fantechi, and L. Semini, *From Software Engineering to Formal Methods and Tools, and Back: Essays Dedicated to Stefania Gnani on the Occasion of Her 65th Birthday*, vol. 11865. Springer Nature, 2019.
- [19] B. Ambacher and M. Conrad, "Computational archival science is a two-way street," in *2021 IEEE International Conference on Big Data (Big Data)*, pp. 2192–2199, IEEE, 2021.
- [20] S. Boßelmann, J. Neubauer, S. Naujokat, and B. Steffen, "Model-driven design of secure high assurance systems: an introduction to the open platform from the user perspective," in *The 2016 International Conference on Security and Management (SAM 2016). Special Track "End-to-end Security and Cybersecurity: from the Hardware to Application*, pp. 145–151, 2016.
- [21] T. Margaria and B. Steffen, "Extreme model-driven development (xmdd) technologies as a hands-on approach to software development without coding," *Encyclopedia of Education and Information Technologies*, pp. 732–750, 2020.
- [22] T. Margaria and B. Steffen, "Agile it: thinking in user-centric models," in *International Symposium On Leveraging Applications of Formal Methods, Verification and Validation*, pp. 490–502, Springer, 2008.
- [23] B. Steffen, F. Gossen, S. Naujokat, and T. Margaria, "Language-driven engineering: from general-purpose to purpose-specific languages," in *Computing and Software Science*, pp. 311–344, Springer, 2019.
- [24] A.-L. Lamprecht, T. Margaria, B. Steffen, A. Sczyrba, S. Hartmeier, and R. Giegerich, "Genefisher-p: variations of genefisher as processes in bio-jeti," *BMC bioinformatics*, vol. 9, no. 4, pp. 1–15, 2008.
- [25] T. Margaria, C. Kubczak, M. Njoku, and B. Steffen, "Model-based design of distributed collaborative bioinformatics processes in the jabc," in *11th IEEE International Conference on Engineering of Complex Computer Systems (ICECCS'06)*, pp. 8–pp, IEEE, 2006.
- [26] B. Jonsson, T. Margaria, G. Naeser, J. Nyström, and B. Steffen, "Incremental requirement specification for evolving systems," *Nord. J. Comput.*, vol. 8, no. 1, pp. 65–87, 2001.
- [27] B. Steffen, T. Margaria, V. Braun, R. Nisius, *et al.*, "A constraint-oriented service creation environment," in *International Workshop on Tools and Algorithms for the Construction and Analysis of Systems*, pp. 418–421, Springer, 1996.
- [28] T. Margaria and A. Schieweck, "The digital thread in industry 4.0," in *IFM 2019, International Conference on Integrated Formal Methods, LNCS 11918*, pp. 3–24, LNCS Springer, 2019.
- [29] H. A. A. Chaudhary, I. Guevara, J. John, A. Singh, A. Ghosal, D. Pesch, and T. Margaria, "Model-driven engineering in digital thread platforms: a practical use case and future challenges," in *International Symposium on Leveraging Applications of Formal Methods*, pp. 195–207, Springer, 2022.
- [30] S. Al-areqi, A.-L. Lamprecht, T. Margaria, S. Kriewald, D. Reusser, and M. Wrobel, "Agile workflows for climate impact risk assessment based on the ci: grasp platform and the jabc modeling framework," 2014.
- [31] S. Al-areqi, A.-L. Lamprecht, and T. Margaria, "Constraints-driven automatic geospatial service composition: Workflows for the analysis of sea-level rise impacts," in *International conference on computational science and its applications*, pp. 134–150, Springer, 2016.
- [32] D. Merkel, "Docker: lightweight linux containers for consistent development and deployment," *Linux journal*, vol. 2014, no. 239, p. 2, 2014.
- [33] J. W. Budd and T. Guinnane, "Intentional age-misreporting, age-heaping, and the 1908 old age pensions act in ireland," *Population Studies*, vol. 45, no. 3, pp. 497–518, 1991.
- [34] M. Safronov and J. Winesett, *Web application development with Yii 2 and PHP*. Packt Publishing Ltd, 2014.
- [35] "Firstname variants from irish genealogy records – roots ireland."
- [36] D. Holmes and M. C. McCabe, "Improving precision and recall for soundex retrieval," in *Proceedings. International Conference on Information Technology: Coding and Computing*, pp. 22–26, IEEE, 2002.
- [37] A. Inbasekaran, R. K. Gnanasekaran, and R. Marciano, "Using transfer learning to contextually optimize optical character recognition (ocr) output and perform new feature extraction on a digitized cultural and historical dataset," in *2021 IEEE International Conference on Big Data (Big Data)*, pp. 2224–2230, IEEE, 2021.