

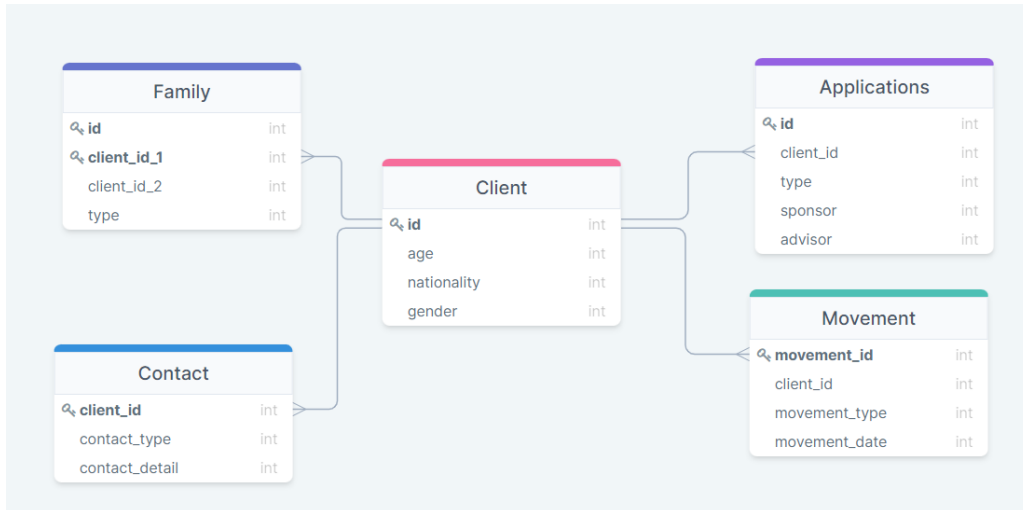
The magic of network graphs:

How R Shiny enables exploration of complex networks

Jane Li & Chris Samson
Risk Analytics and Data Science | Immigration New Zealand
R Exchange, May 2025

A paradigm shift in how to think about data

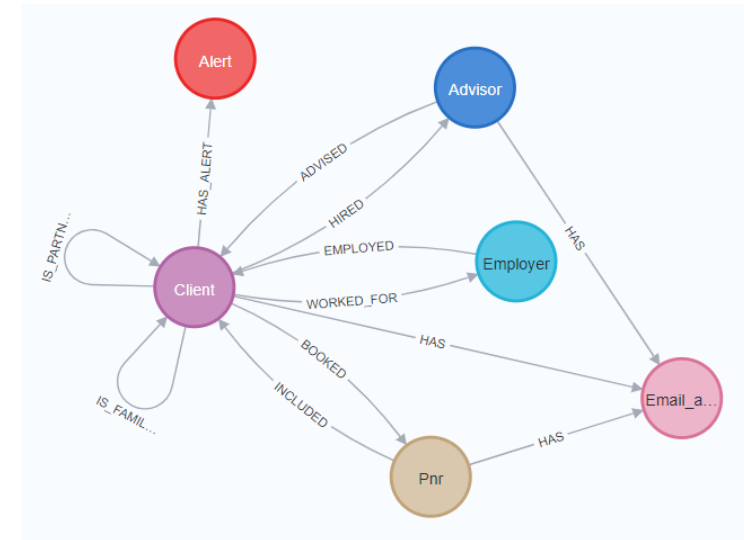
Static approach



Discrete data, as represented in a relational database

- Tables and keys

Connected approach

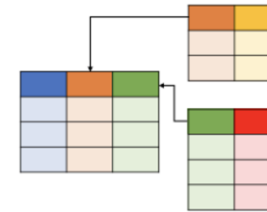


Data represented as network in a “graph”-structure

- Nodes and Links (entities and relations)

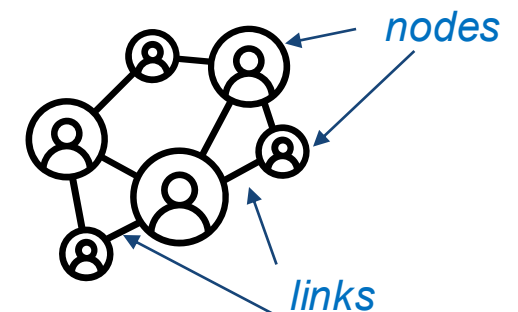
Why Network Analytics and Graphs?

- **Relational Databases:** *don't handle relationships well!*
 - ✓ Not well adapted to interconnected semi-structure
 - ✓ Relationship queries **require painful SQL & joins**
 - ✓ *Practically limited to three degrees of separation*



A **graph** is a collection of **vertices (nodes)** representing entities and **edges** representing the **links (relationships)** among them.

- **Graphs** are *optimised for data problems which involve many-to-many relationships*:
 - ✓ Navigate **deep hierarchies**
 - ✓ Find **hidden connections** between distant items
 - ✓ Discover **inter-relationships** between items
 - ✓ Identifying key players and **detecting communities**



ETL process and Graph logic

Data collection

- Using SQL connection to INZ server
- Package: DBI

```
my_server = DBI::dbConnect(odbc::odbc(),  
                           driver = "SQL Server",  
                           server = "foo.wd.govt.nz",  
                           encoding = "US-ASCII")  
  
Clients <- dbGetQuery(my_server,  
  "SELECT TOP(10)  
    [Client ID]  
    ,[Name]  
    ,[Nationality]  
    ,[Visa]  
  FROM [Immigration].[Client] "
```

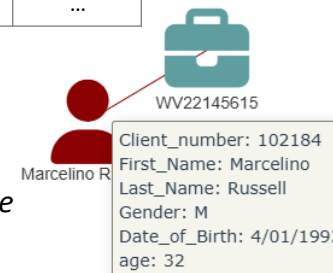
Data processing

- Data cleaning and deduplication
- Dataframes for nodes and links
- Properties for different types of node and link

"Client" Table

ID	Firs Name	Last Name	Gender	DoB
102184	Marcelino	Russell	M	4/01/1993
...	...	...	...	...

Node type
"Client"



Graph creation

- Nodes and links saved as *parquet* files to use in Databricks
- Graph from nodes and links *igraph::graph_from_data_frame*

Main R packages:

- *igraph*
- *visNetwork*
- *shiny*

Network Analytics App

Search by node id

Search by connected types with condition

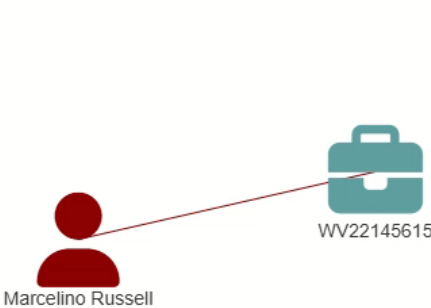
Search by node types and property

Search shortest path between nodes

Select by id

Select by group

Client 102184, 1 hop(s)



Expand/Contract Node

Remove Node

Graph stats

Download Graph (HTML)

Download nodes & links (Excel)

About dataset

Why graphs

ETL process

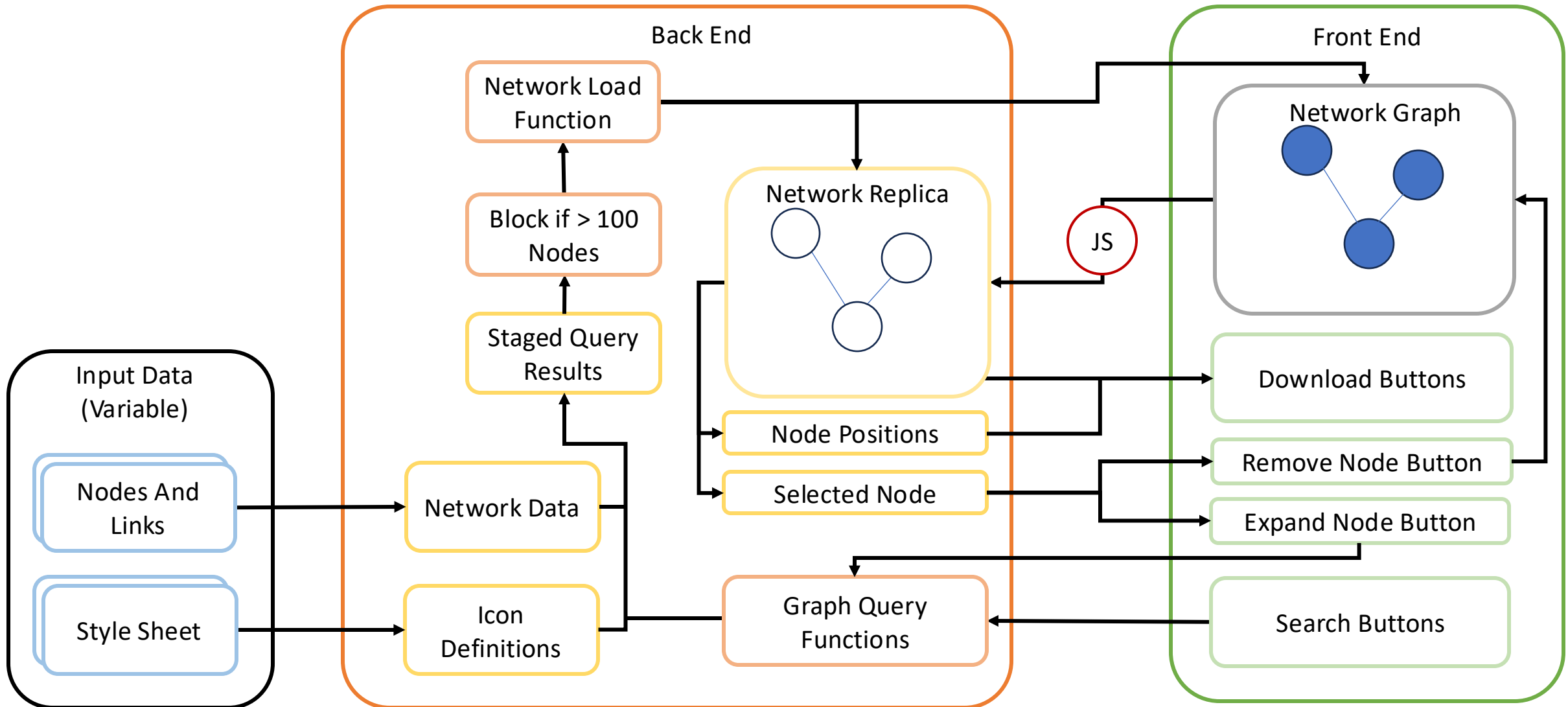
Demo

App schematic

Deployment

Lessons learned

IN-CONFIDENCE



Why graphs

ETL process

Demo

App schematic

Deployment

Lessons learned

IN-CONFIDENCE

Deployment: App

Option	Pros	Cons
Local R 	- Easy to distribute - Zero infrastructure required	- A little technical to launch - Version control not possible
 databricks	- Good version control - Notebook environment for development	- Users need admin rights - Need to launch compute to activate
 snowflake	- Easy set up with Snowpark Container Services - Good access privilege control - Good version control	- Requires a containerized version of app - No native notebook environment (Posit app available)
Third party managed cloud environment (e.g.,  posit)	- Often easy to use - Environment for development & version control	- An additional service to fit into instructed and security setup - Vary in quality and features
Self-managed cloud app (e.g., Azure App Services )	- Fully modular and controllable	- Requires skillset to set up and maintain - Typically requires creation of a containerized version of app

Why graphs

ETL process

Demo

App schematic

Deployment

Lessons learned

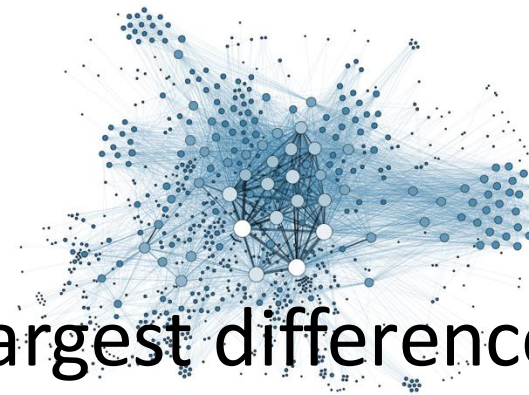
IN-CONFIDENCE

Deployment: Data

Option	Pros	Cons
File with App 	- Easy to distribute - Zero infrastructure required	- Only works with small scale data - Difficult to update
	- Common database - Can easily be integrated with existing cloud and on-premises architecture	- Inefficient to query, particularly when traversing multiple edges - Cannot be effectively used for graph specific algorithms (i.e. shortest paths) - Does not scale well and cannot be effectively partitioned
Graph DB  	- Gold standard for storing graph data - Efficient to query - [Most] support graph algorithms	- Almost always stand-alone tools that need to be procured and integrated into infrastructure - No standard query language

Some lessons learned

- ❖ Engage with stakeholders early.
- ❖ Things grow fast.
- ❖ Sometimes small things make the largest difference.
- ❖ Deployment is a large effort.
- ❖ Building an app is only the beginning.





**NEW ZEALAND
IMMIGRATION**

THANK YOU!



**MINISTRY OF BUSINESS,
INNOVATION & EMPLOYMENT**
HĪKINA WHAKATUTUKI

Te Kāwanatanga o Aotearoa
New Zealand Government