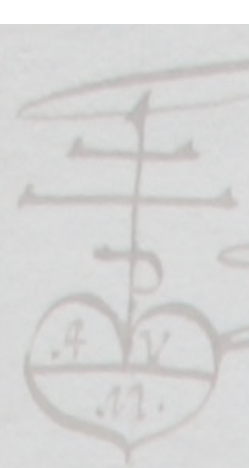


Visualizing Double-Entry Bookkeeping: A Digital History Methodology

Jesse Sadler
Virginia Tech

@vivalosburros
jessesadler.com
github.com/jessesadler

Slides: jessesadler.com/slides/chico2021.pdf


DANIEL. VANDER .
MEVLEN .
A°1583 .
HESTER.DELLA .
FAILLE .
A°1583.

Solving niche historical problems with Digital Humanities: Non-decimal currencies

- Finding my footing in Digital Humanities
- debkeepr: Analysis of Non-Decimal Currencies in R
- Example: The estate of Jan della Faille de Oude (1515–1582)

Finding my footing in Digital Humanities

Find a project

Power of coding

Just pick a project

How to draw an owl

1.

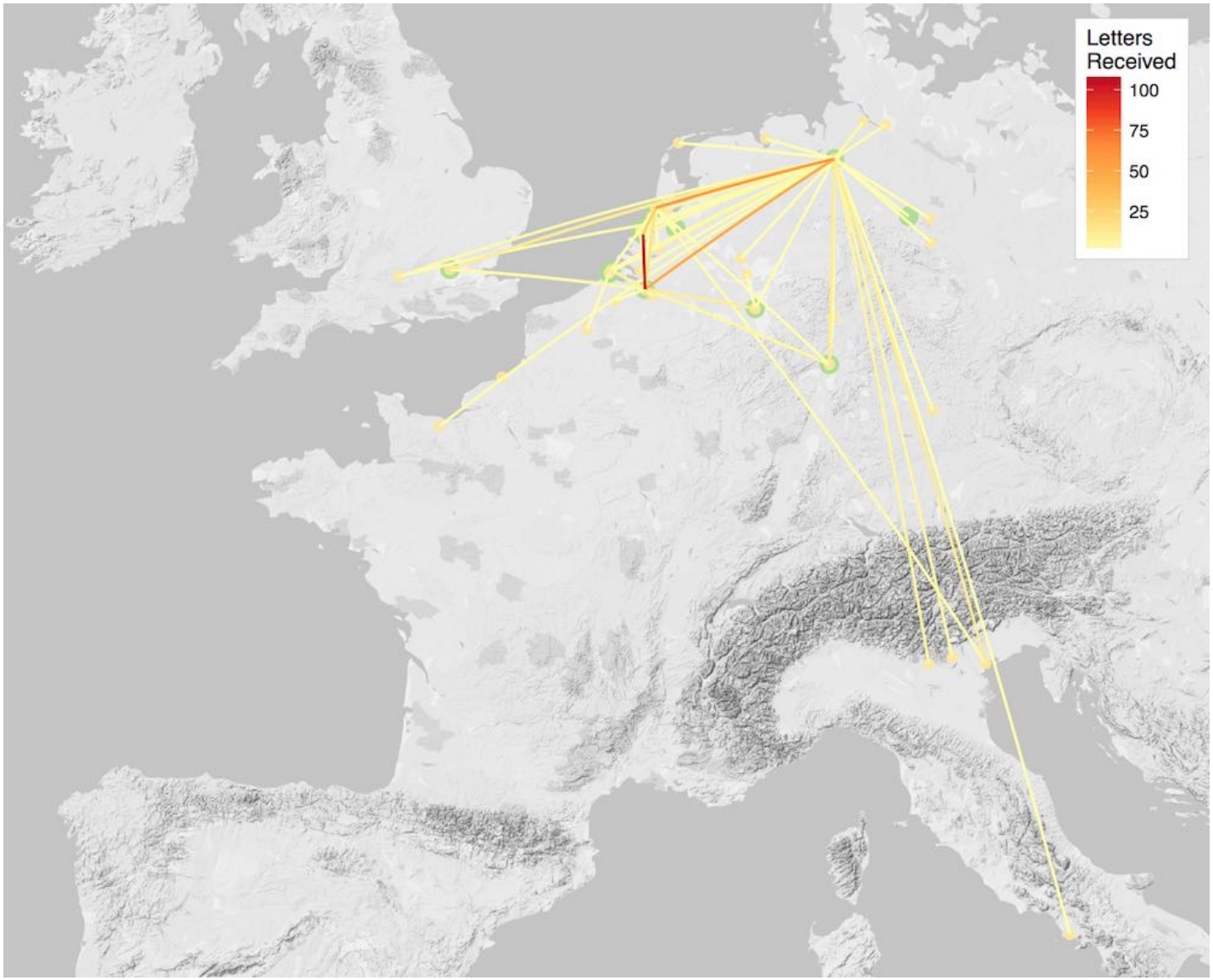
2.

1. Draw some circles

2. Draw the rest of the fucking owl

6,000 letters sent to Daniel van der Meulen between 1578 and 1600

Letters received by Daniel van der Meulen, 1578–1591



jessesadler.com/project/dvdm-correspondence/

Wickham and Grolemund, R for Data Science

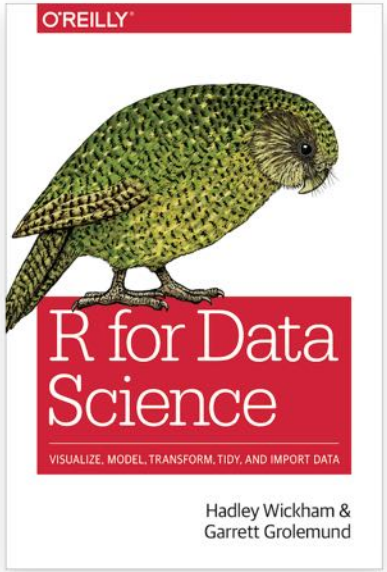
R for Data Science

Search

Table of contents

Welcome

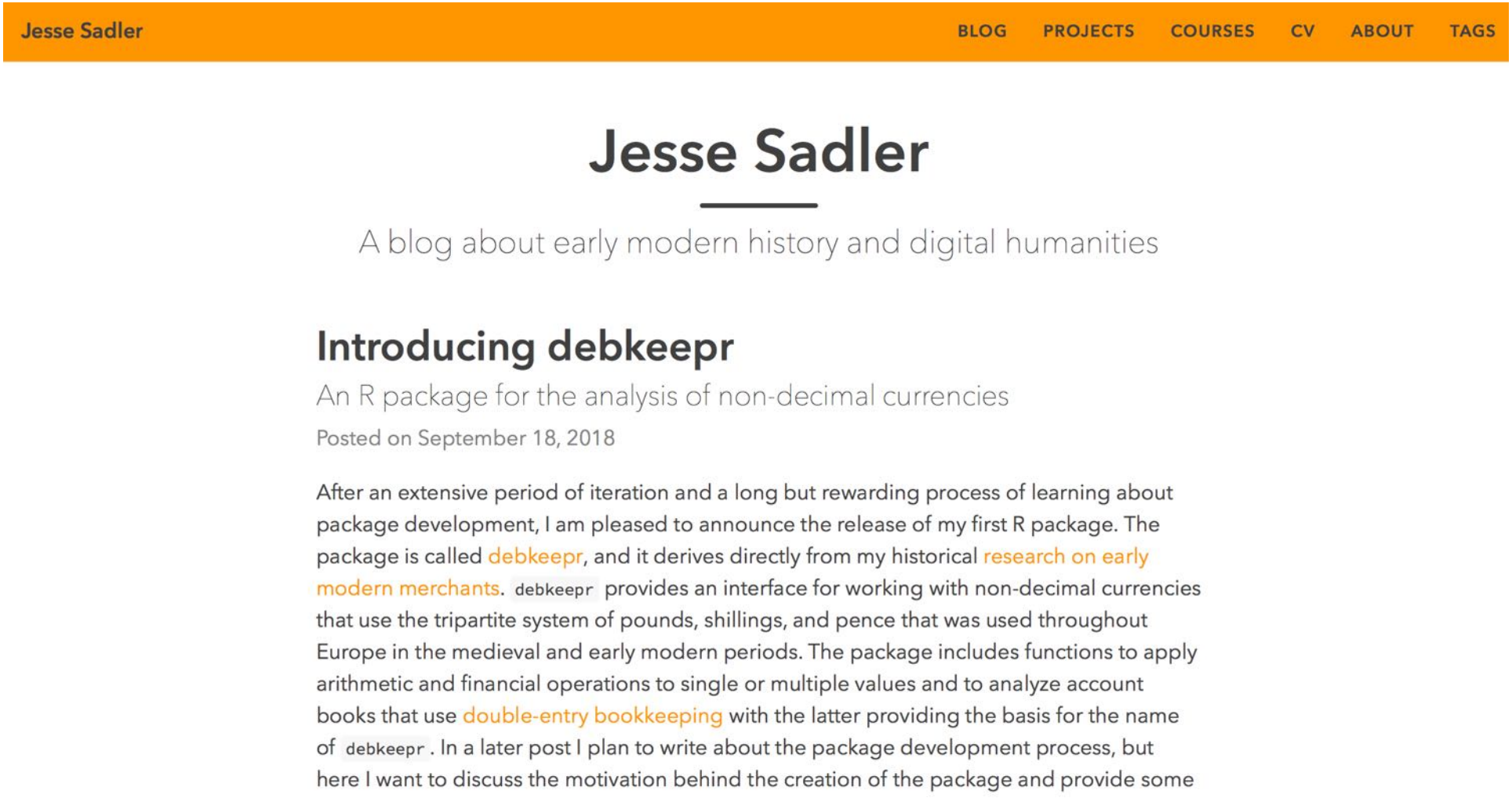
This is the website for **"R for Data Science"**. This book will teach you how to do data science with R: You'll learn how to get your data into R, get it into the most useful structure, transform it, visualise it and model it. In this book, you will find a practicum of skills for data science. Just as a chemist learns how to clean test tubes and stock a lab, you'll learn how to clean data and draw plots—and many other things besides. These are the skills that allow data science to happen, and here you will find the best practices for doing each of these things with R. You'll learn how to use the grammar of graphics, literate programming, and reproducible research to save time. You'll also learn how to manage cognitive resources to facilitate discoveries when wrangling, visualising, and exploring data.



This website is (and will always be) **free to use**, and is licensed under the Creative Commons Attribution-NonCommercial-NoDerivs 3.0 License. If you'd like a **physical copy** of the book, you can order it from amazon; it was published by O'Reilly in January 2017. If you'd like to **give back** please make a donation to Kākāpō Recovery: the kākāpō (which appears on the cover of R4DS) is a critically endangered native NZ

https://
r4ds.had.co.nz

jessesadler.com



My approach to Digital Humanities

- Develop transferable skills
- Build small-scale projects
- Use open-source tools that facilitate reproducible research
- Use skills to solve niche problems

Champagne fairs

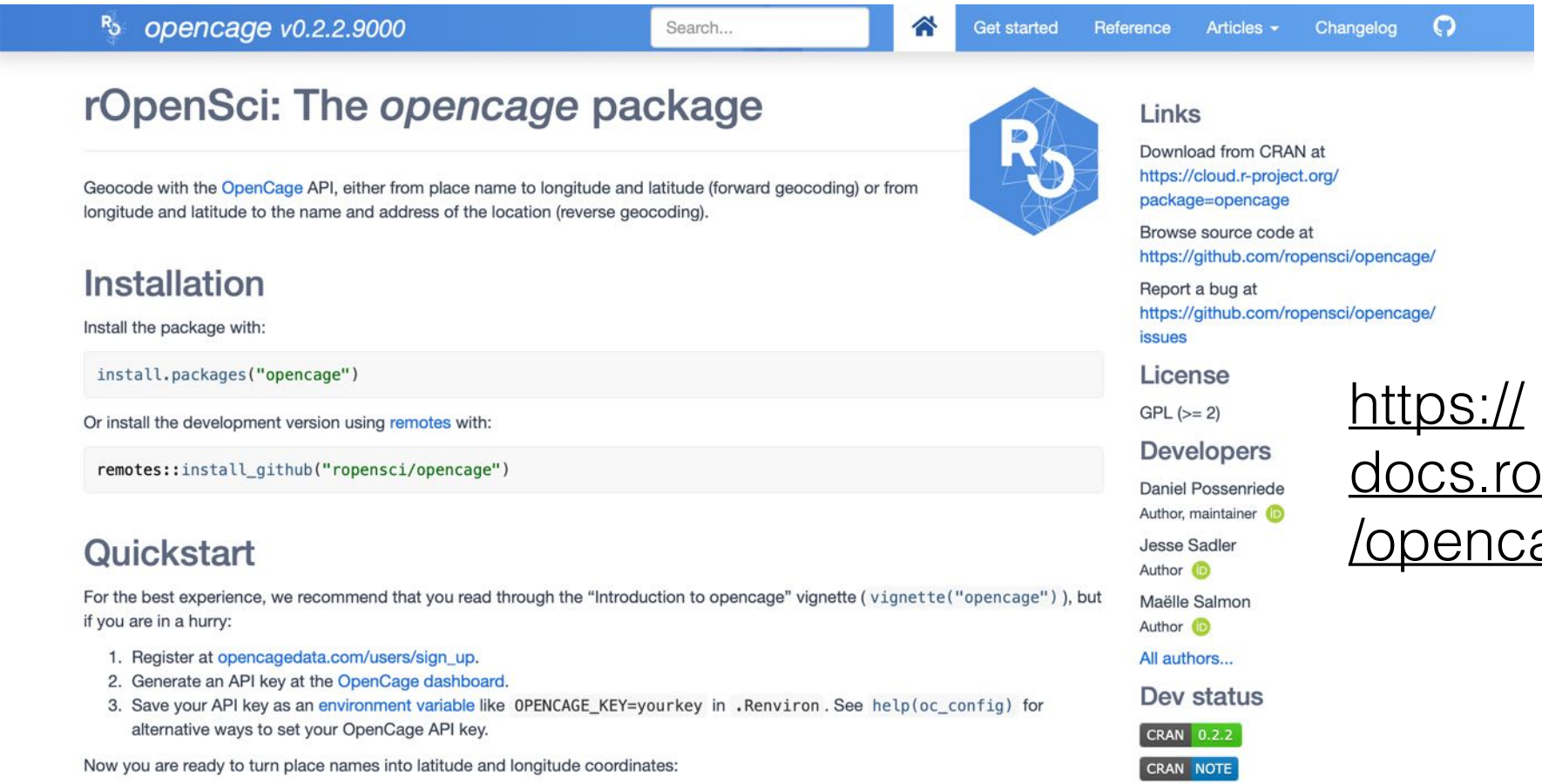
Network of Florentine banks

Renaissance Italy

Contributing to data science

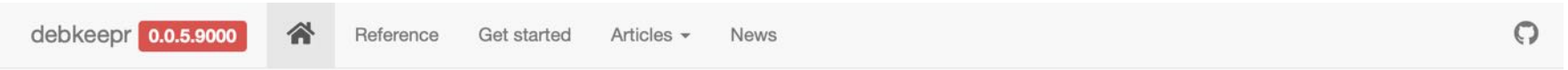
- Approaching learning digital skills from a humanistic perspective
- Blogging: Learning in public
- Becoming a part of the open-source community

opencage: Open-source geocoding



https://docs.ropensci.org/opencage/

debkeepr: Analysis of Non-Decimal Currencies in R



debkeepr: Analysis of Non-Decimal Currencies

`debkeepr` integrates non-decimal currencies that use the tripartite system of pounds, shillings, and pence into the methodologies of Digital Humanities and the practices of reproducible research. The package makes it possible for historical non-decimal currencies to behave like decimalized numeric values through the implementation of the `deb_lsd` and `deb_decimal` vector classes or types. These types are based on the infrastructure provided by the vctrs package. `debkkeepr` simplifies the process of performing arithmetic calculations with non-decimal currencies — such as adding £3 13s. 4d. sterling to £8 15s. 9d. sterling — and also provides a basis for analyzing account books with thousands of transactions recorded in non-decimal currencies. The name of the `debkeepr` package derives from this latter capability of analyzing historical account books that often used double-entry bookkeeping.

Installation

You can install `debkeepr` from GitHub with remotes:

```
# install.packages("remotes")
remotes::install_github("jessesadler/debkeepr")
```

Please open an issue if you have any questions, comments, or requests.

Historical Background

The `debkeepr` package uses the nomenclature of l, s, and d to represent pounds, shillings, and pence units in non-decimal currencies. The abbreviations derive from the Latin terms libra, solidus, and denarius. The libra was a Roman measurement of weight, while the solidus and denarius were both Roman coins. The denarius was a silver coin from the era of the Republic, in contrast to the golden solidus that was issued in the Late Empire. As the production of silver coins overtook that of gold by the 8th century, a solidus came to represent 12 silver denarii coins, and 240 denarii were — for a time — made from one libra or pound of silver. The custom of

Links

Browse source code at https://github.com/jessesadler/debkeepr/

Report a bug at https://github.com/jessesadler/debkeepr/issues

License

Full license

MIT + file LICENSE

Developers

Jesse Sadler
Author, maintainer

Dev status

https://jessesadler.github.io/debkeepr/

GitHub: https://github.com/jessesadler/debkeepr/

— + Anno 1585. —

	£	s	d
Jacques della faille de jonghe. is — crediteur Adi 15. februarij aº 1585. £ 26. 4. 0. voor oncosten van onse Bruijloft voor 50½ ell [?] a 10s aen willem monnincx. betaelt ende 19 s. voor vracht van den coffer met 7. packetten compt tsamen — — — £	26	4	—
— ditto £ 0. 0. 0. voor de voors. voor schipvracht van Jo: Soster della failles met diversche packetten Int vertreck van Antwerpen. — — £	0	—	—
— ditto £ 5. 1. 10. voor de voors. voor een silveren schale wegende 11 oz 15 Ing. aen Jo: Soster della faille In october overgelaten. — — £	5	1	10
— ditto £ 28. 10. 0. voor de voors. voor 4. amen renschen wijn voor de Bruijloft tot Dort gecost compt met de vaten. — — — £	28	10	—
— ditto £ 4. 7. 0. voor de voors. voor diverse lacken ende carsaij aen Jo: Soster della faille over lange overgelaten — — — £	4	7	0
— ditto £ 369. 4. 4. voor de voors. voor een rekeninge van oncosten opde Bruijloft gedaen als breeder bij deselve rekeninge is blijckende. — — £	369	4	4
20 ditto £ 85. 4. 11. voor de voors. voor een rekeninge van oncosten tot behoeve vande Bruijloft bij Jacques noirot In Antwerpen betaelt. £	85	4	11
— ditto £ 15. 18. 10. voor de voors. voor soo veele mij Jan della faille voor de conduite vant Bruijloft volck Int gaen ende comen betaelt £	15	18	10
— ditto £ 2. 18. 6. voor de voors. voor soo veele aende timmerman voor sparren deelen ende arbeyt betaelt — — — £	2	18	6
— ditto £ 0. 3. 4. voor de voors. voor vracht ende licentie van een gheleijs [?] aen [illegible] comende van londen — — — £	—	3	4
10 Marti £ 155. 1. 4. voor tarwe In handen van Robert noirot In Gimichem [?] voordat ick Inde 235. quarters tarwe geladen Int schip de roose de reijse gemaect [illegible]			

Non-decimal currency nomenclature lsd

libra	solidus	denarius
£	s.	d.
Pound	shilling	penny (pence)

Names for non-decimal currencies

- Pound sterling: 1 pound = 20 shillings; 1 shilling = 12 pence
- Pound Flemish: 1 pound = 20 schellingen; 1 schelling = 12 groten
- Holland guilders: 1 guilder = 20 stuivers; 1 stuiver = 16 penningen
- French crowns: 1 crown = 60 sous; 1 sous = 12 deniers
- Polish florins: 1 florin = 30 gros; 1 gros = 18 denar
- Portuguese real: 1 milréis = 1,000 réis

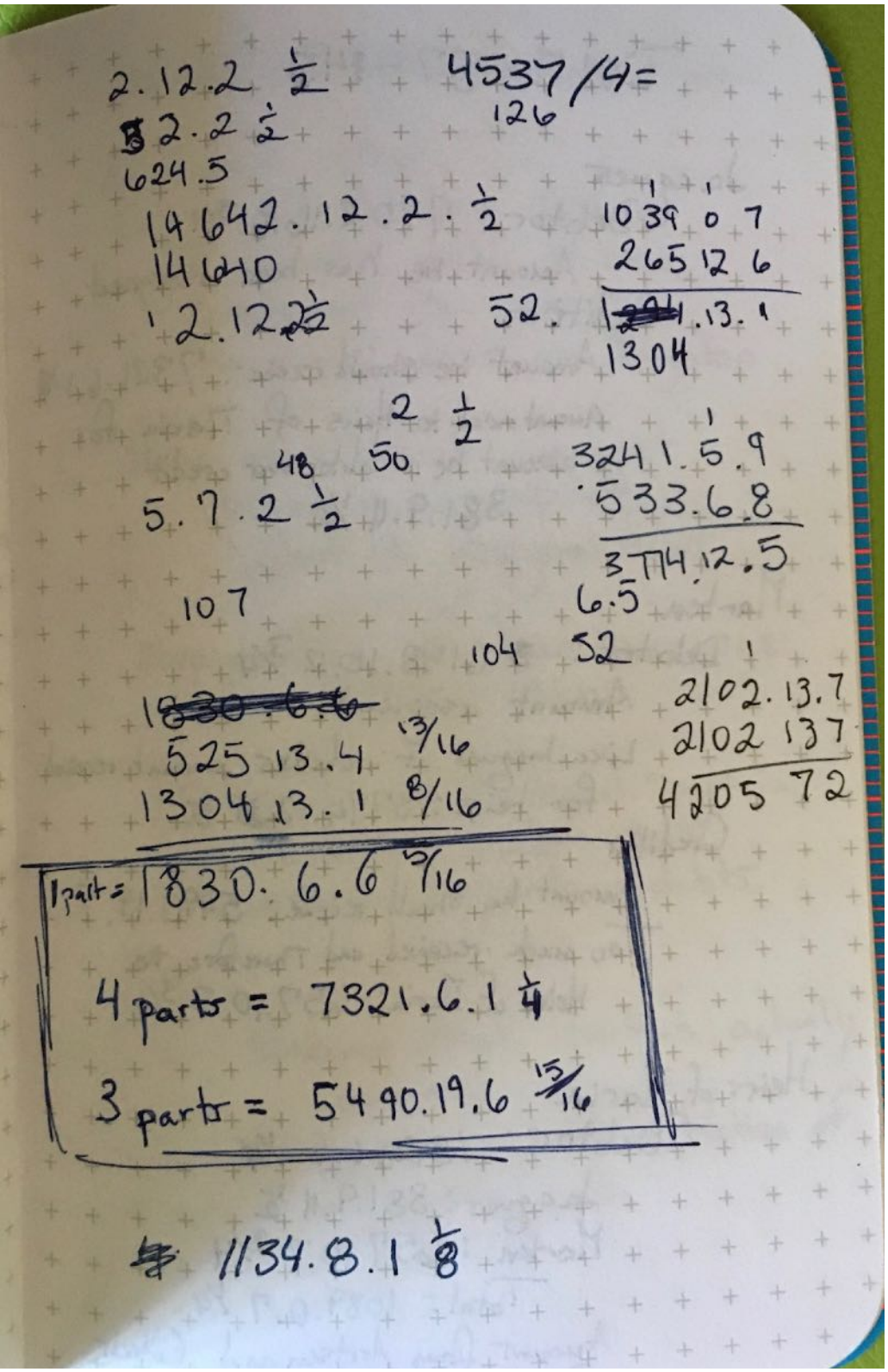
Problem space

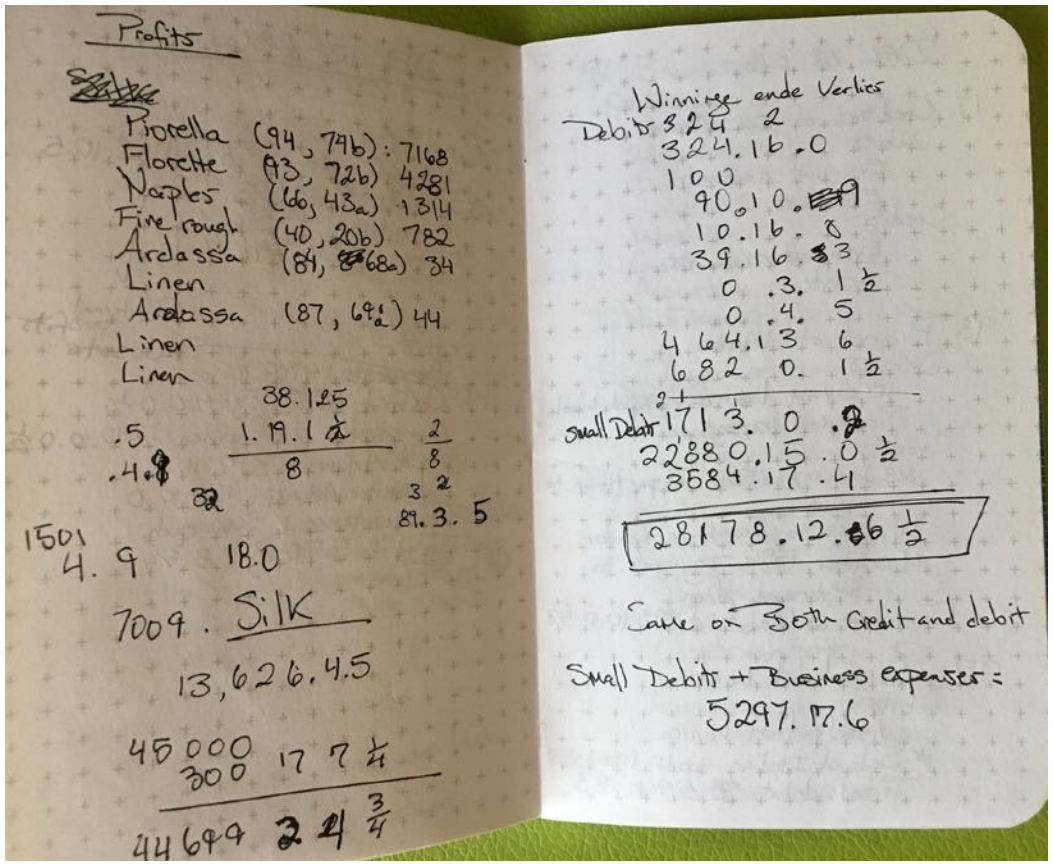
Compound unit arithmetic

	£	s.	d.
	28	15	8
	32	8	11
	54	18	7
	18	12	9
Answer	**£134**	**15s.**	**11d.**
Unit total	132	53	35
Divide by base	-	53 / 20	35 / 12
Carried forward	2	2	-
Remainder	-	13	11

- Three separate units make up one value
- The units have non-decimal bases
- Need to use compound-unit arithmetic to normalize values
- The non-decimal bases differed by currency

Arithmetic by hand





Design principles for a solution

- A class that maintains the tripartite structure of non-decimal currencies
- Decimalized class as fall back
- Track the bases of shillings and pence units
- Vectors with different bases cannot be combined
- Choose and track unit represented by decimalized class
- Vectors with different units can be combined but need coercion path

Tripartite structure

```
deb_lsd(l = c(17, 32, 18),
        s = c(16, 7, 12),
        d = c(6, 9, 3))
```

```
#> <deb_lsd[3]>
#> [1] 17:16s:6d 32:7s:9d
#> [3] 18:12s:3d
#> # Bases: 20s 12d
```

Decimalized

```
deb_decimal(x = c(17.8250,
                  32.3875,
                  18.6125))
```

```
#> <deb_decimal[3]>
#> [1] 17.8250 32.3875
#> [3] 18.6125
#> # Unit: pounds
#> # Bases: 20s 12d
```

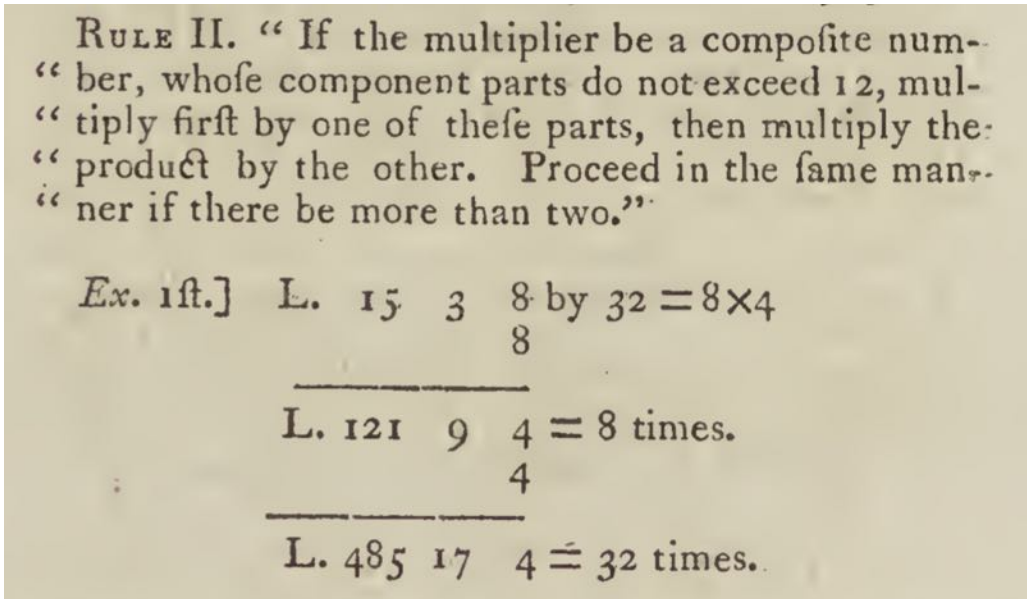
Normalization

Compound unit arithmetic

	£	s.	d.
	28	15	8
	32	8	11
	54	18	7
	18	12	9
Answer	**£134**	**15s.**	**11d.**
Unit total	132	53	35
Divide by base	-	53 / 20	35 / 12
Carried forward	2	2	-
Remainder	-	13	11

```
deb_normalize(c(132, 53, 35))

#> <deb_lsd[1]>
#> [1] 134:15s:11d
#> # Bases: 20s 12d
```

Multiplication



RULE II. "If the multiplier be a compoſite number, whoſe component parts do not exceed 12, multiply firſt by one of theſe parts, then multiply the product by the other. Proceed in the ſame manner if there be more than two."

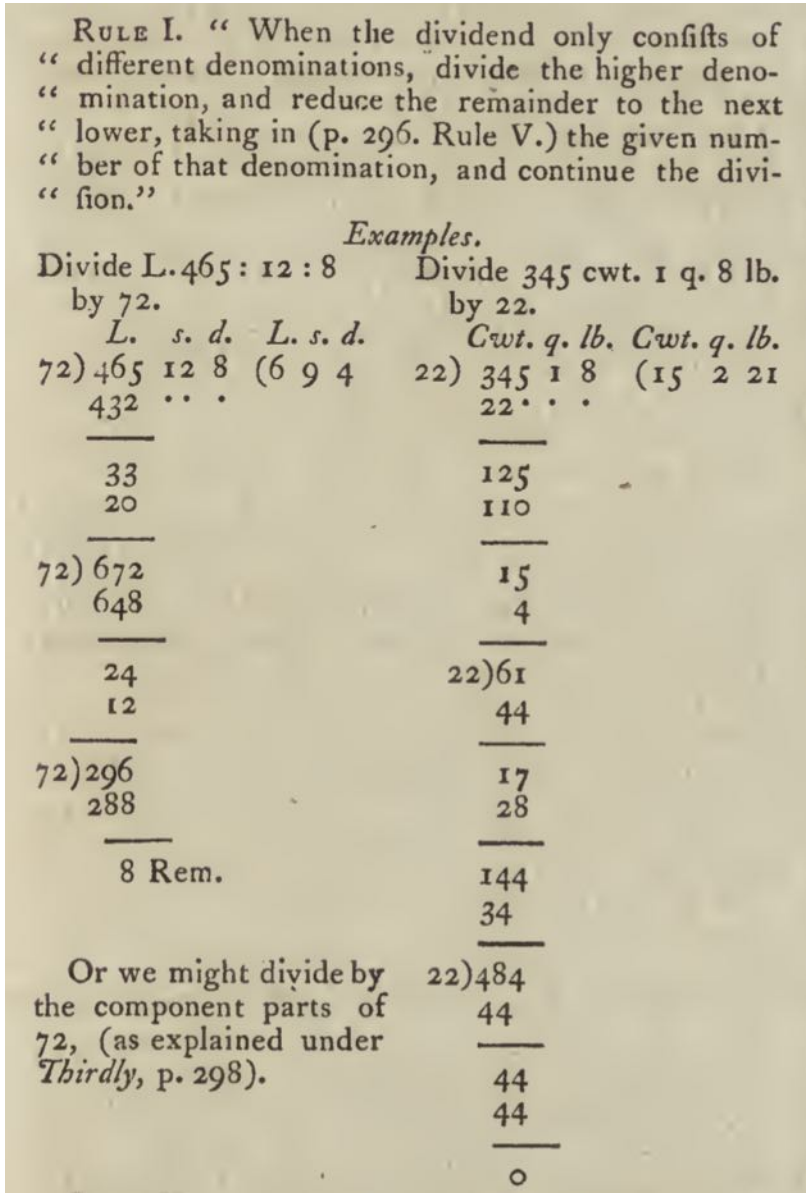
Ex. 1ſt.] L. 15 3 8 by 32 = 8×4
8

L. 121 9 4 = 8 times.
4

L. 485 17 4 = 32 times.

```
# Multiply £15 3s. 8d.
  sterling by 32

deb_lsd(15, 3, 8) * 32

#> <deb_lsd[1]>
#> [1] 485:17s:4d
#> # Bases: 20s 12d
```

Division



RULE I. "When the dividend only consists of "different denominations, divide the higher denomination, and reduce the remainder to the next "lower, taking in (p. 296. Rule V.) the given number of that denomination, and continue the division."

Examples.

Divide L.465 : 12 : 8 by 72.

L.	*s.*	*d.*	*L.*	*s.*	*d.*
72) 465	12	8	(6	9	4
432	· ·	·			
33					
20					
72) 672					
648					
24					
12					
72) 296					
288					
8 Rem.					

Or we might divide by the component parts of 72, (as explained under *Thirdly*, p. 298).

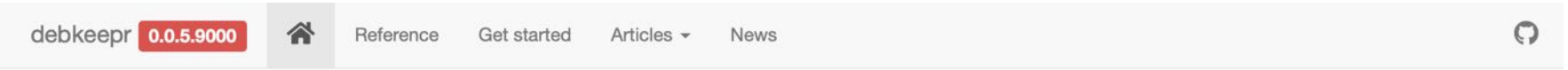
Divide 345 cwt. 1 q. 8 lb. by 22.

Cwt.	*q.*	*lb.*	*Cwt.*	*q.*	*lb.*
22) 345	1	8	(15	2	21
22	· ·	·			
125					
110					
15					
4					
22)61					
44					
17					
28					
144					
34					
22)484					
44					
44					
44					
0					

```
# Divide 345cwt. 1q. 8lbs.
  by 22

x <- deb_lsd(345, 1, 8,
             bases = c(4, 28))

x / 22

#> <deb_lsd[1]>
#> [1] 15:2s:22d
#> # Bases: 4s 28d
```

```
# Arithmetic

deb_lsd(15, 15, 9) + deb_lsd(6, 13, 4)

#> <deb_lsd[1]>
#> [1] 22:9s:1d
#> # Bases: 20s 12d

# Comparison

deb_lsd(15, 15, 9) < deb_lsd(10, 128, 432)

#> TRUE
```

debkeepr: Analysis of Non-Decimal Currencies in R



debkeepr: Analysis of Non-Decimal Currencies

`debkeepr` integrates non-decimal currencies that use the tripartite system of pounds, shillings, and pence into the methodologies of Digital Humanities and the practices of reproducible research. The package makes it possible for historical non-decimal currencies to behave like decimalized numeric values through the implementation of the `deb_lsd` and `deb_decimal` vector classes or types. These types are based on the infrastructure provided by the vctrs package. `debkkeepr` simplifies the process of performing arithmetic calculations with non-decimal currencies — such as adding £3 13s. 4d. sterling to £8 15s. 9d. sterling — and also provides a basis for analyzing account books with thousands of transactions recorded in non-decimal currencies. The name of the `debkeepr` package derives from this latter capability of analyzing historical account books that often used double-entry bookkeeping.

Installation

You can install `debkeepr` from GitHub with remotes:

```
# install.packages("remotes")
remotes::install_github("jessesadler/debkeepr")
```

Please open an issue if you have any questions, comments, or requests.

Historical Background

The `debkeepr` package uses the nomenclature of l, s, and d to represent pounds, shillings, and pence units in non-decimal currencies. The abbreviations derive from the Latin terms libra, solidus, and denarius. The libra was a Roman measurement of weight, while the solidus and denarius were both Roman coins. The denarius was a silver coin from the era of the Republic, in contrast to the golden solidus that was issued in the Late Empire. As the production of silver coins overtook that of gold by the 8th century, a solidus came to represent 12 silver denarii coins, and 240 denarii were — for a time — made from one libra or pound of silver. The custom of

Links

Browse source code at https://github.com/jessesadler/debkeepr/

Report a bug at https://github.com/jessesadler/debkeepr/issues

License

Full license

MIT + file LICENSE

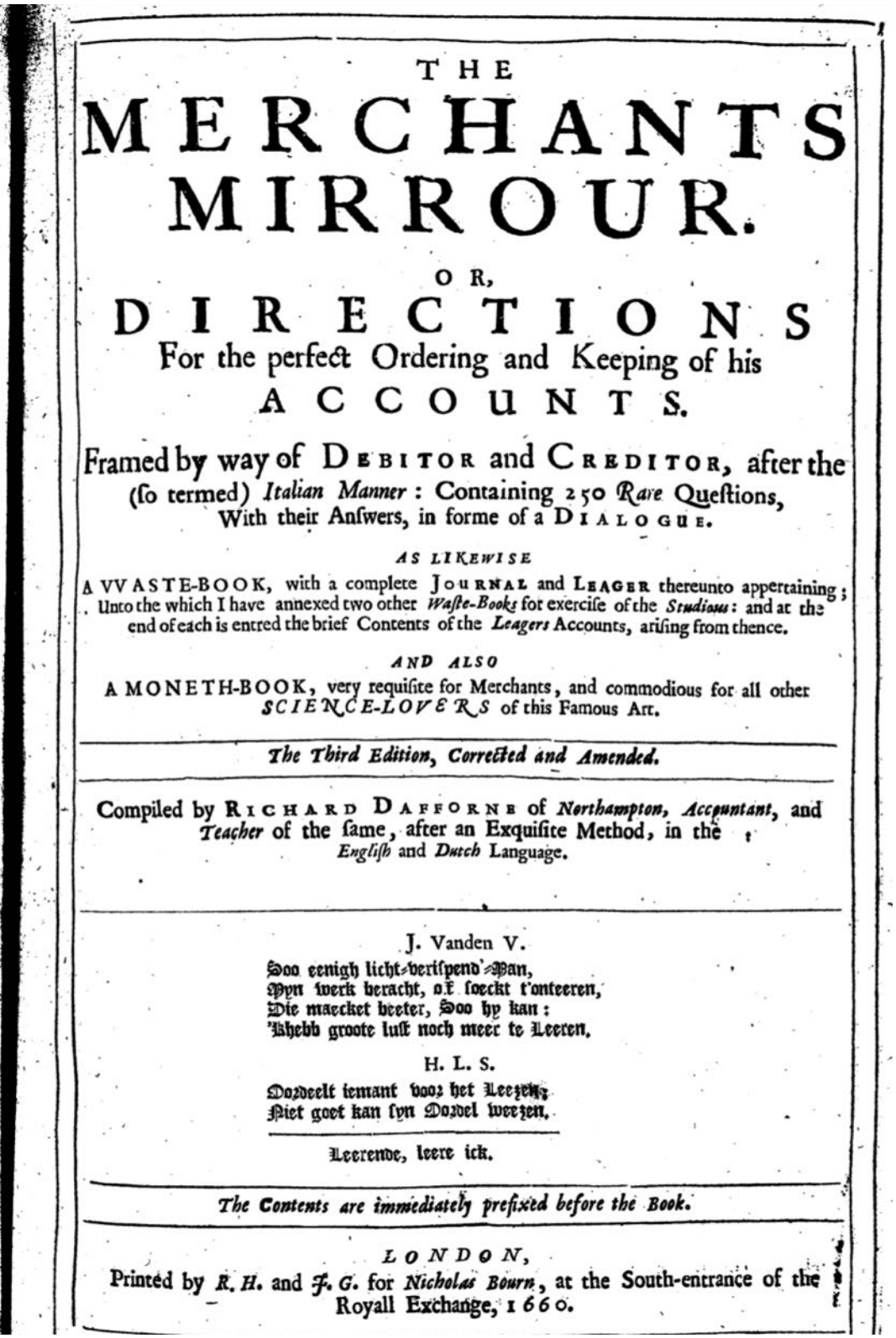
Developers

Jesse Sadler
Author, maintainer

Dev status

https://jessesadler.github.io/debkeepr/

GitHub: https://github.com/jessesadler/debkeepr/



THE
MERCHANTS MIRROUR.
OR,
DIRECTIONS
For the perfect Ordering and Keeping of his
ACCOUNTS.

Framed by way of DEBITOR and CREDITOR, after the (ſo termed) *Italian Manner*: Containing 250 *Rare* Queſtions, With their Anſwers, in forme of a DIALOGUE.

AS LIKEWISE

A VVASTE-BOOK, with a complete JOURNAL and LEAGER thereunto appertaining; Unto the which I have annexed two other *Waſte-Books* for exerciſe of the *Studious*: and at the end of each is entred the brief Contents of the *Leagers* Accounts, ariſing from thence.

AND ALSO

A MONETH-BOOK, very requiſite for Merchants, and commodious for all other *SCIENCE-LOVERS* of this Famous Art.

The Third Edition, Corrected and Amended.

Compiled by RICHARD DAFFORNE of *Northampton*, *Accountant*, and *Teacher* of the ſame, after an Exquiſite Method, in the *Engliſh* and *Dutch* Language.

J. Vanden V.

Soo eenigh licht-verſpend' Man,
Syn werk beracht, oft ſoeckt t'onteeren,
Die maecket beeter, Soo hy kan:
'Khebb groote luſt noch meer te Leeren.

H. L. S.

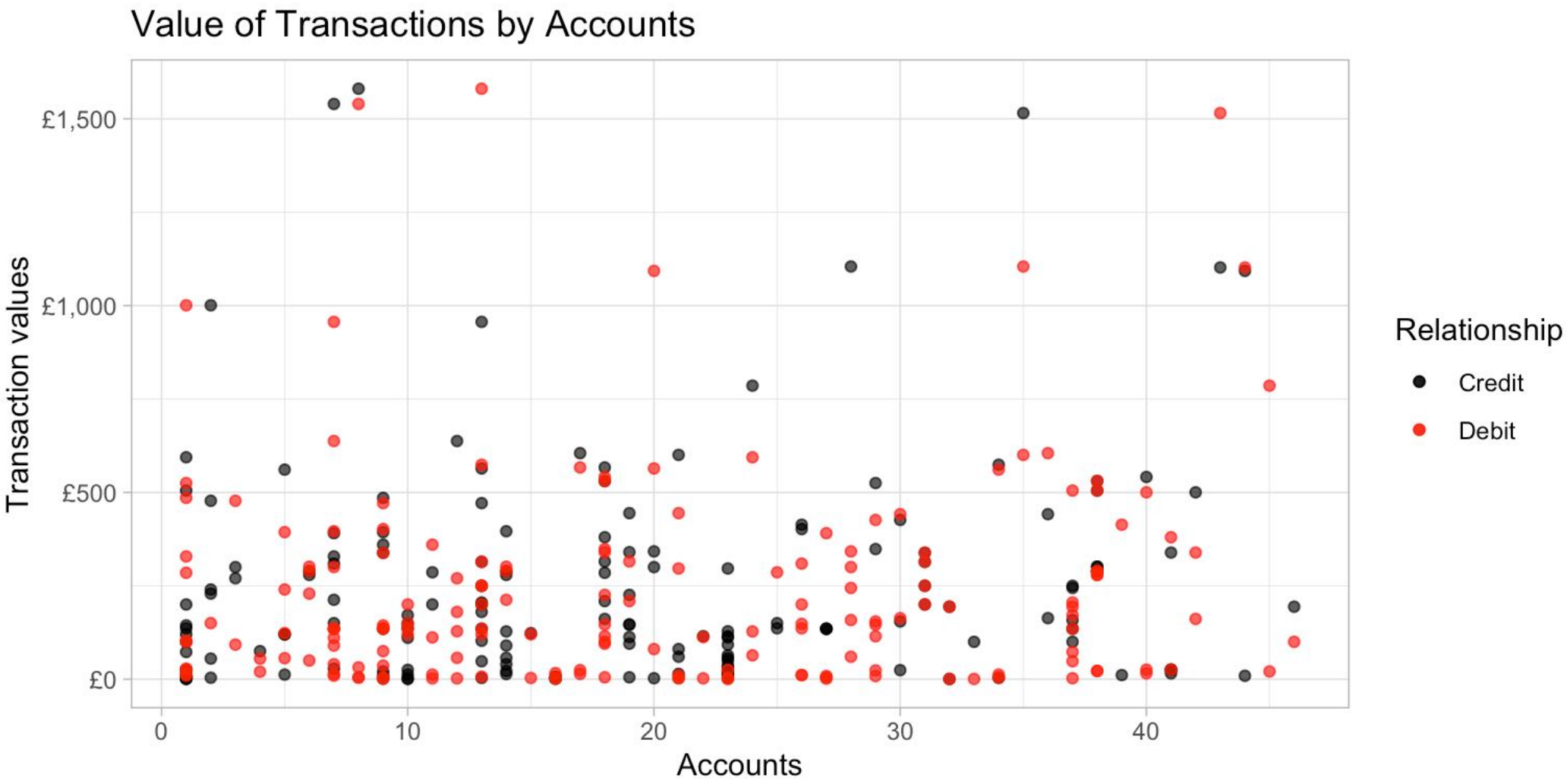
Oordeelt iemant voor het Leezen;
Niet goet kan ſyn Oordel weezen.

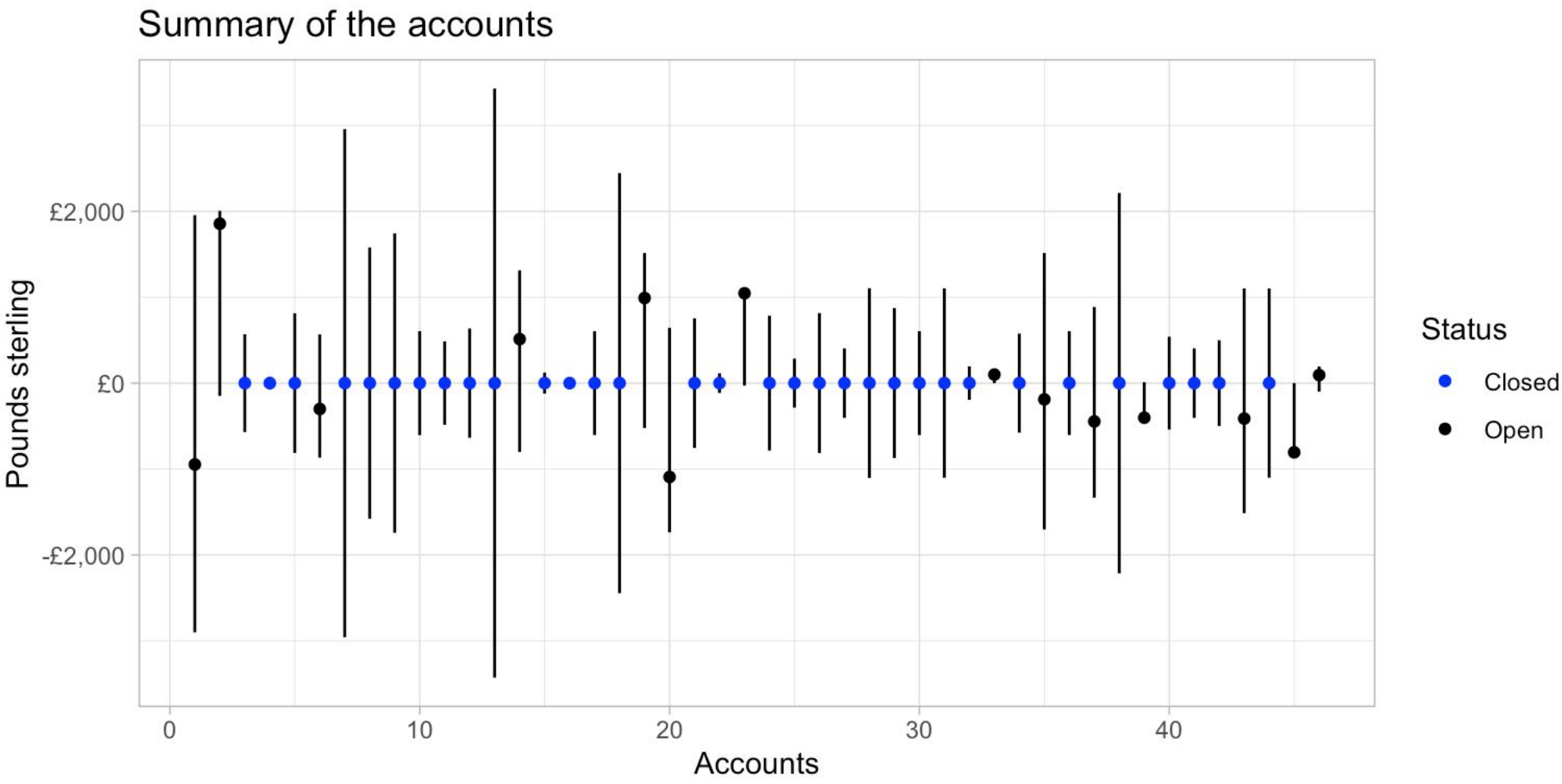
Leerende, leere ick.

The Contents are immediately prefixed before the Book.

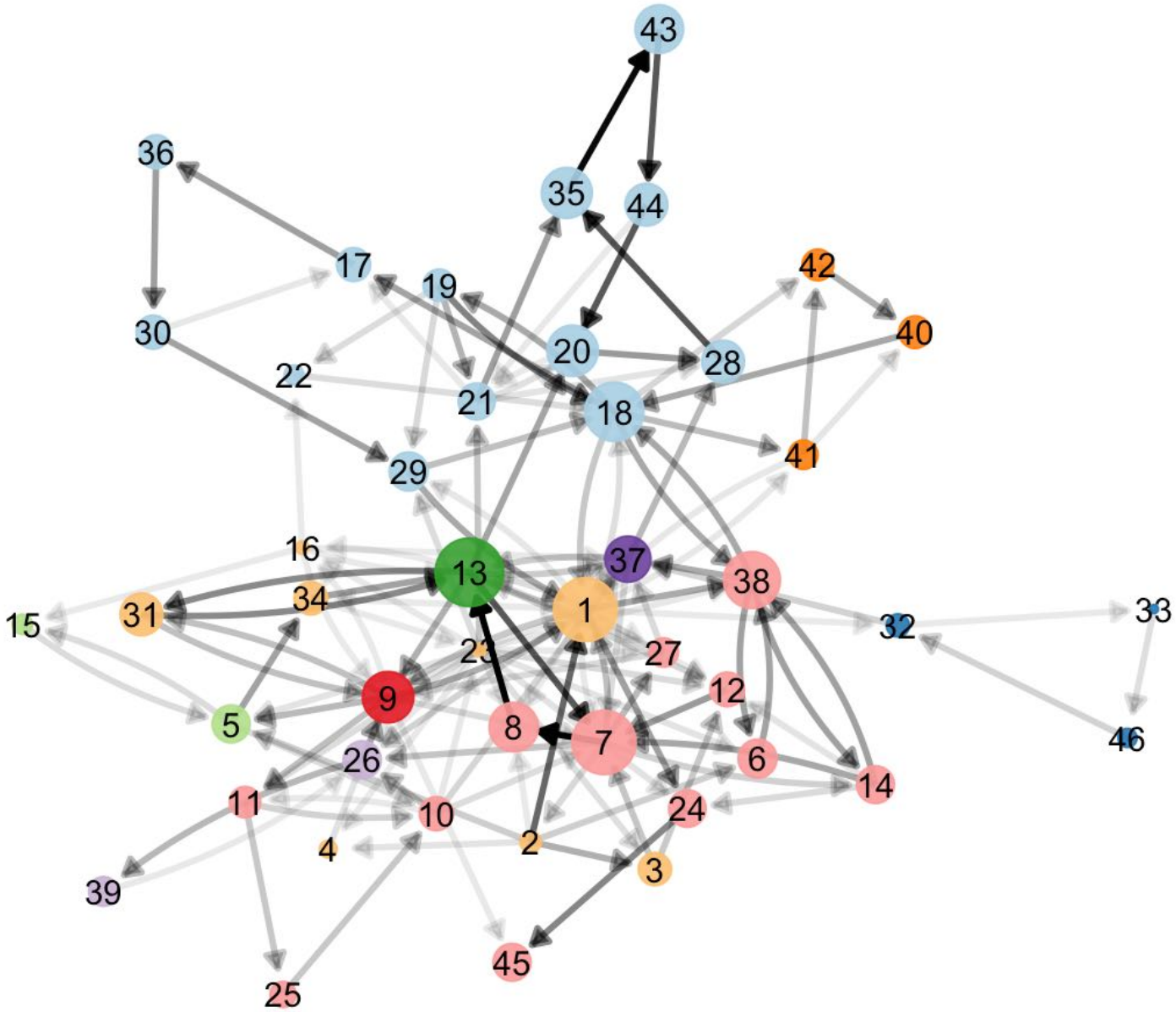
LONDON,
Printed by *R. H.* and *F. G.* for *Nicholas Bourn*, at the South-entrance of the Royall Exchange, 1660.

Richard Dafforne,
The Merchant's Mirrour
(1660)


Value of Transactions by Accounts
Transaction values
£1,500
£1,000
£500
£0
0
10
20
30
40
Accounts
Relationship
Credit
Debit


Summary of the accounts
£2,000
£0
-£2,000
Pounds sterling
0
10
20
30
40
Accounts
Status
Closed
Open

Network of Dafforne's Journal



Accumulated Value

£1,000

£2,000

£3,000

Accumulated Transactions

£400

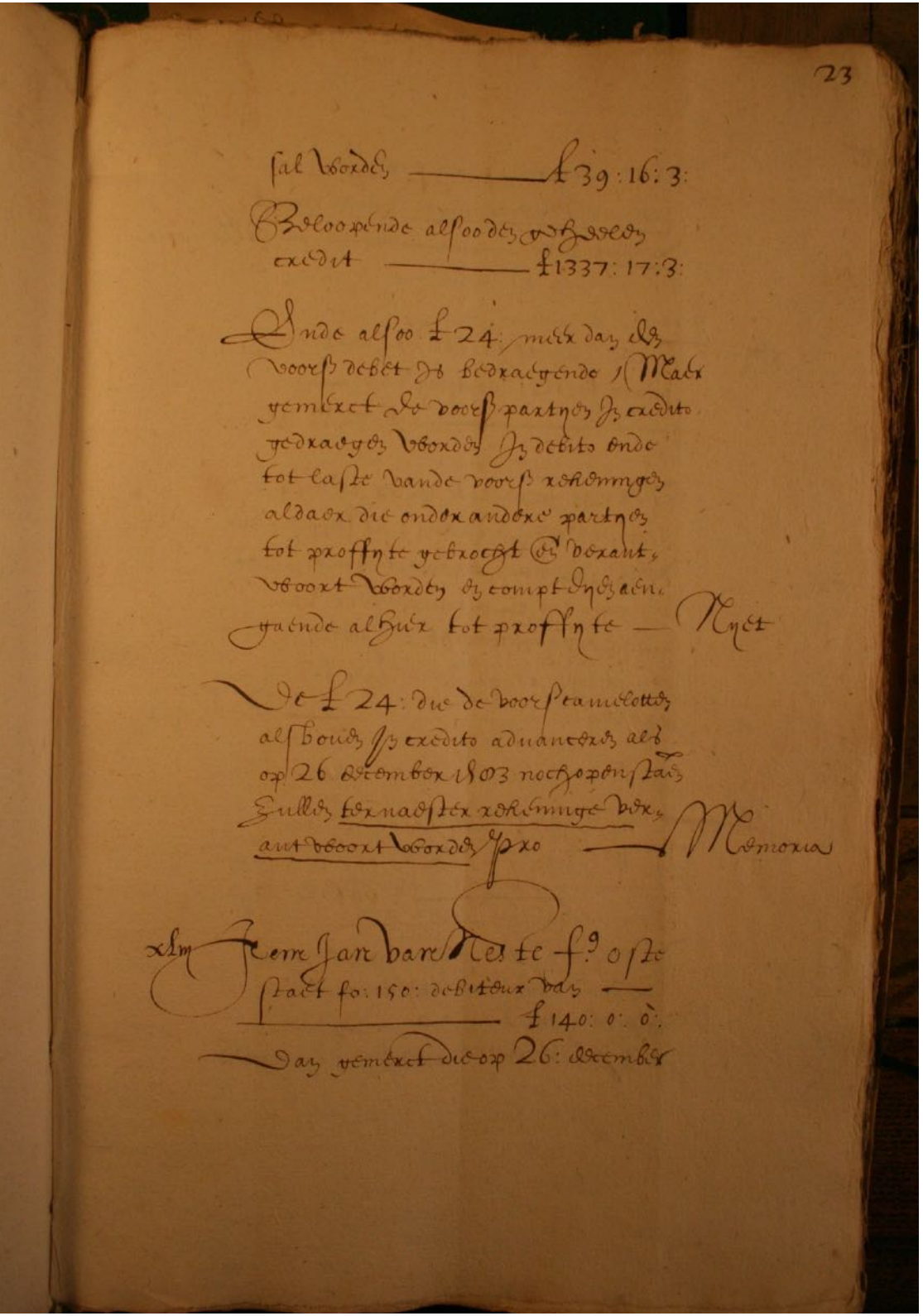
£800

£1,200

Investor

- Andrew Hitchcock
- Arthur Mumperson
- Diego del Varino
- Ego
- George Pinchback
- Jacob Symonson
- James Wilkinson
- Jean du Boys
- Linden, Does, Reinst
- Randoll Rice

Example: The estate of Jan della Faille de Oude (1515–1582)



Accounts of the estate of Jan della Faille de Oude

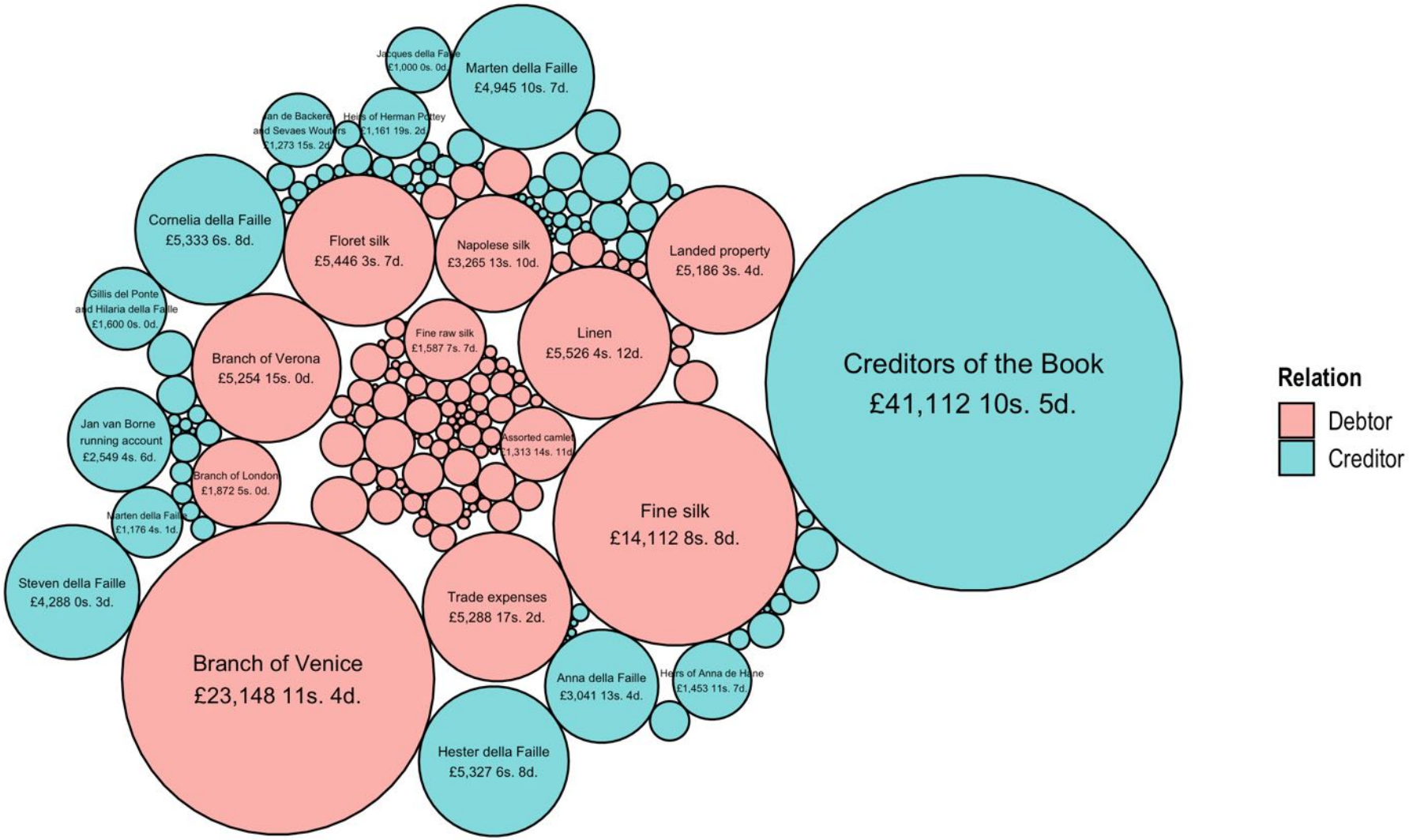
- Date: 8 November 1582 to 31 December 1594

- 8 November 1582 to 26 December 1583
- 26 December 1583 to 31 December 1594

- **Jan** (c. 1542–1618): Executor of estate but not successor to his father.
- **Anna** (c. 1543–1622): Married Robert van Eeckeren, a close associate of her father.
- **Marten** (c. 1544–1620): Chosen by his father as his primary successor.
- **Carlo** (c. 1546–1617): Constant source of trouble for his siblings.
- **Jacques** (c. 1549–1615): Executor of estate. Moved to Haarlem in 1584.
- **Steven** (c. 1550–1621): Married twice only to have his marriages annulled through the will of his father.
- **Maria** (1555–1578): Married and had three children before her early death.
- **Hester** (c. 1558–1643): Married the merchant Daniel van der Meulen.
- **Cornelia** (c. 1563–1582): Died shortly after her father.

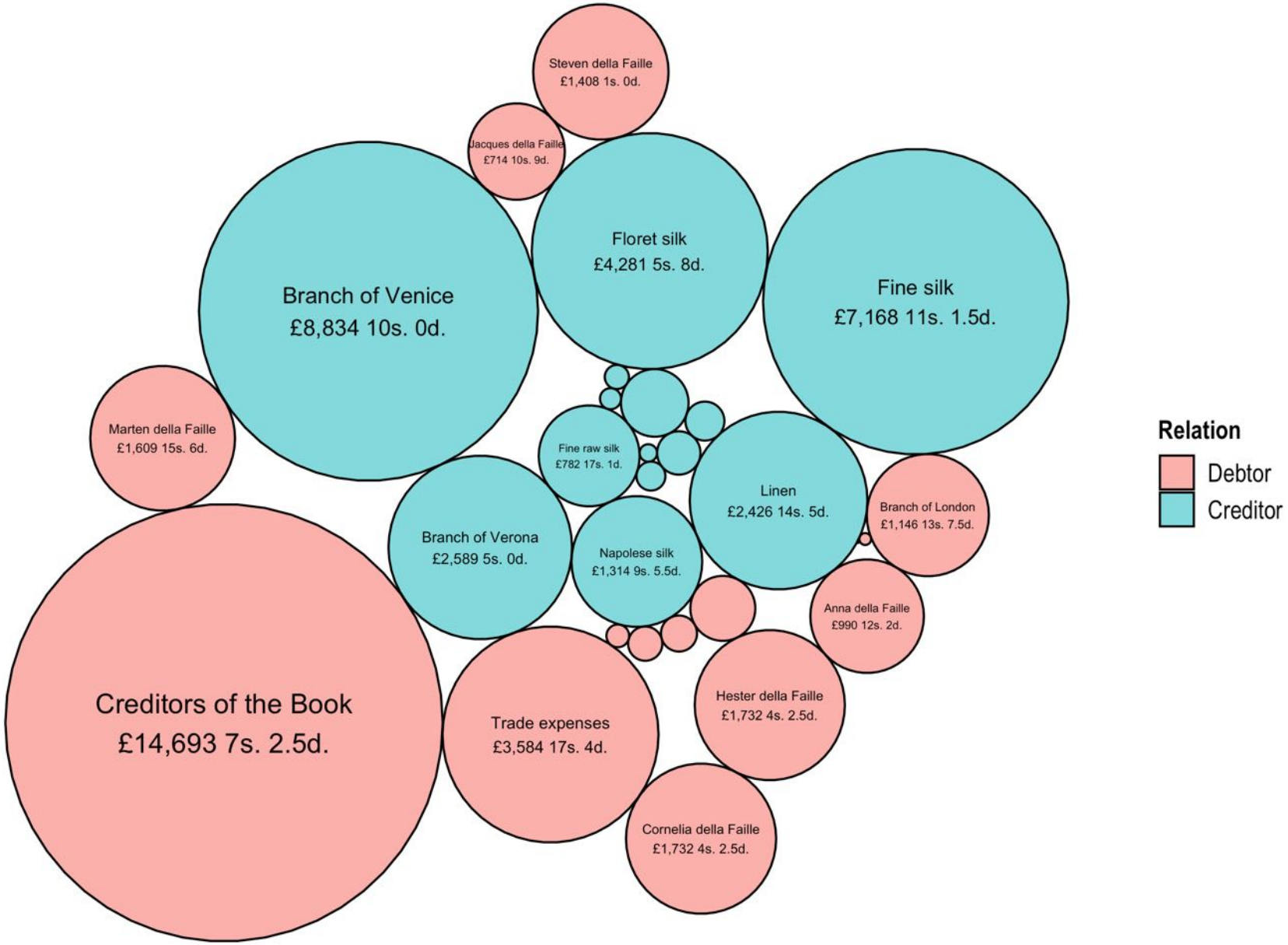
Value of accounts in the estate of Jan della Faille de Oude, 8 December 1582

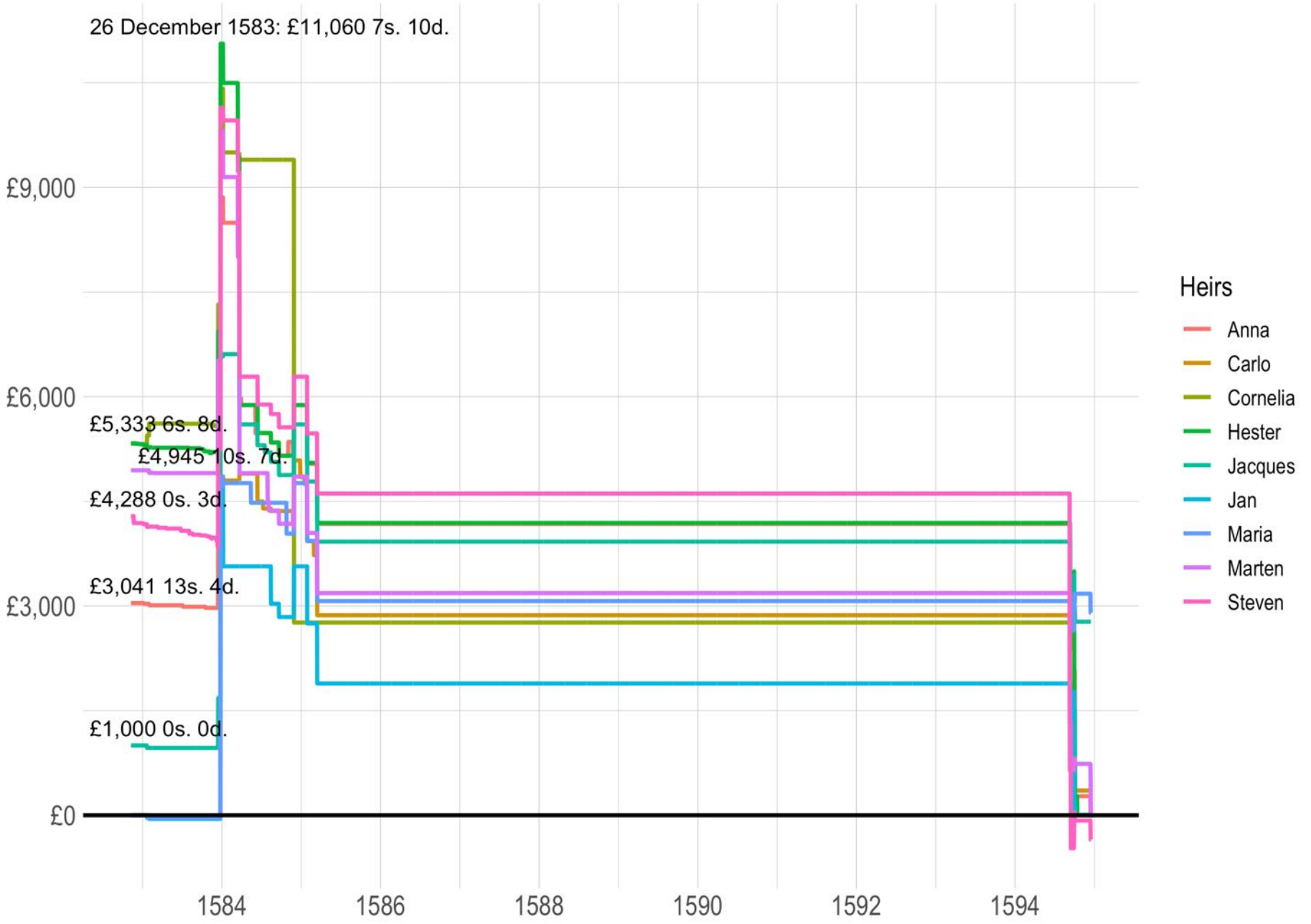
Opening value of the estate: £82,813 5s. 8d.



Profits and losses in the trade of Jan de Oude

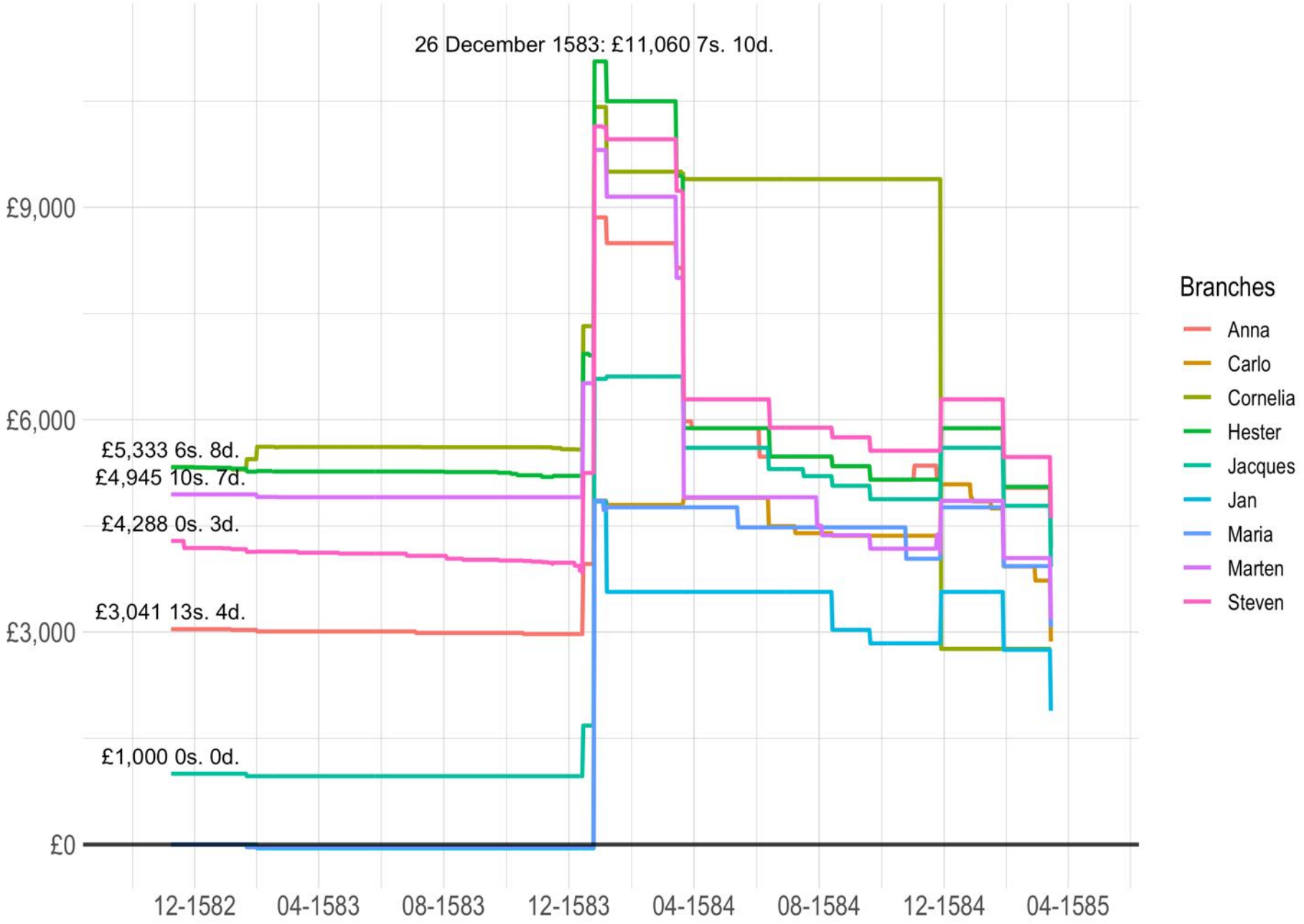
1 January 1579 to 26 December 1583




Inheritance due to the heirs of Jan de Oude
Estate of Jan della Faille de Oude, 1582–1594
26 December 1583: £11,060 7s. 10d.
£9,000
£6,000
£3,000
£0
£5,333 6s. 8d.
£4,945 10s. 7d.
£4,288 0s. 3d.
£3,041 13s. 4d.
£1,000 0s. 0d.
1584
1586
1588
1590
1592
1594
Heirs
Anna
Carlo
Cornelia
Hester
Jacques
Jan
Maria
Marten
Steven

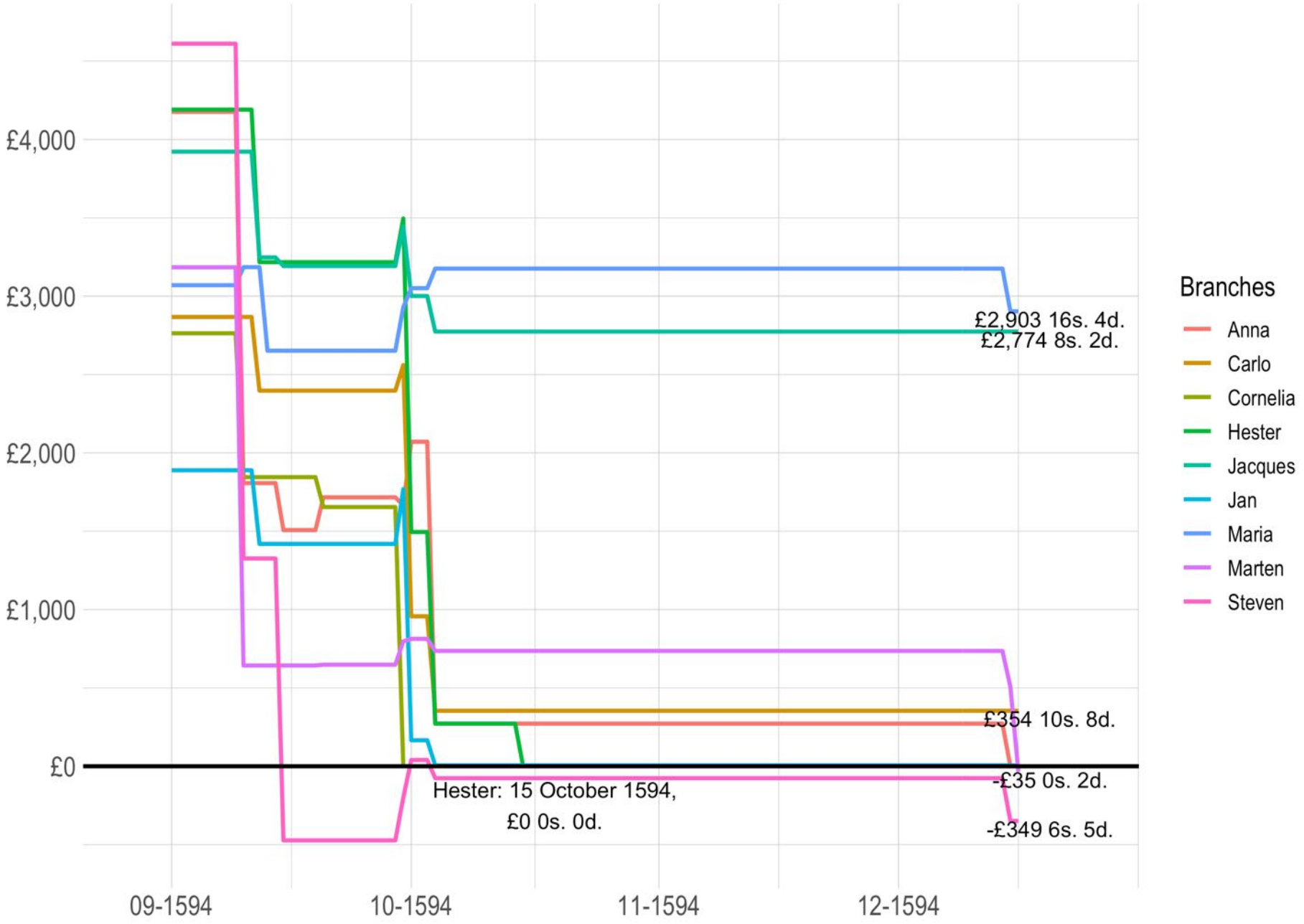
Inheritance due to the heirs of Jan de Oude

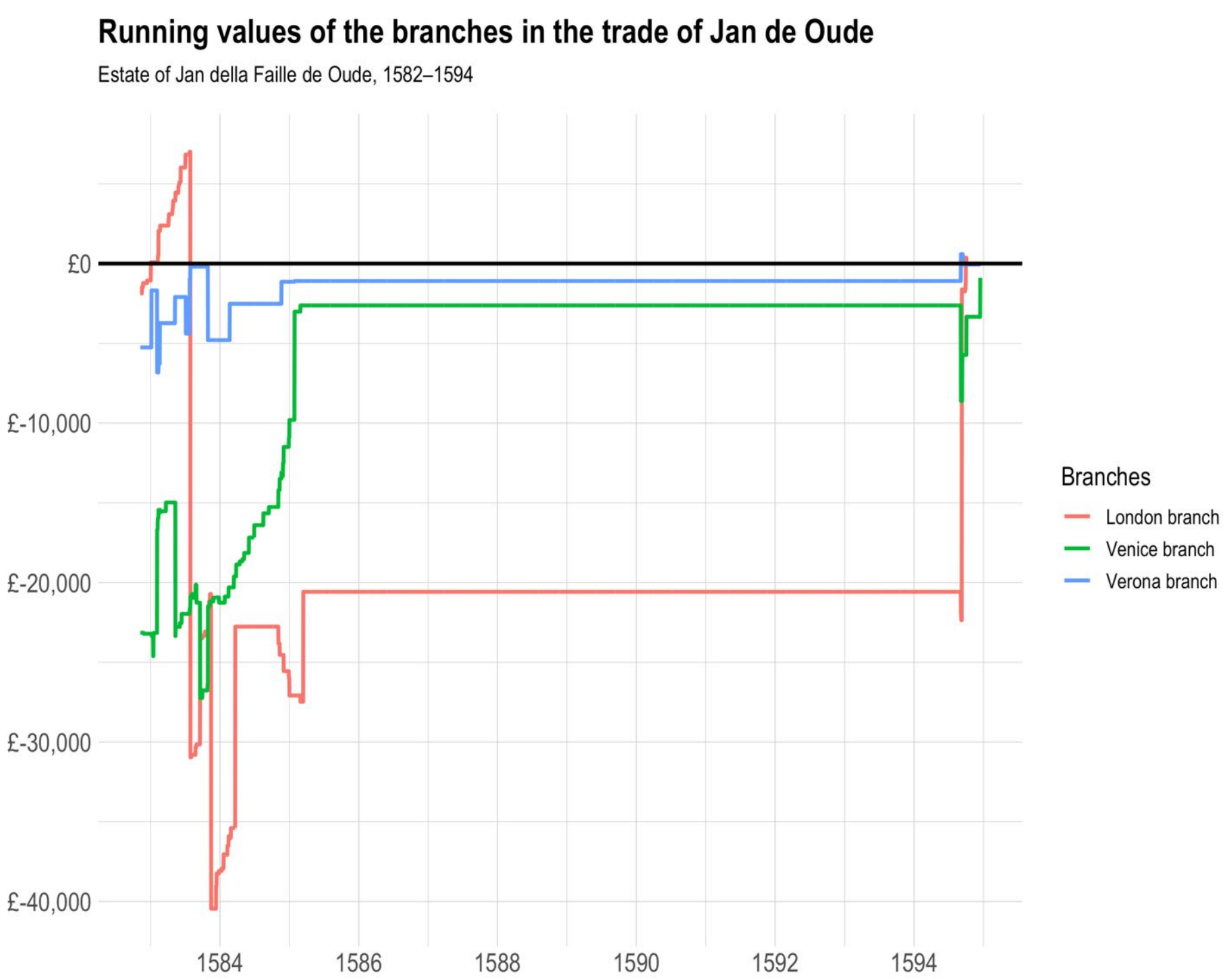
November 1582 to March 1585

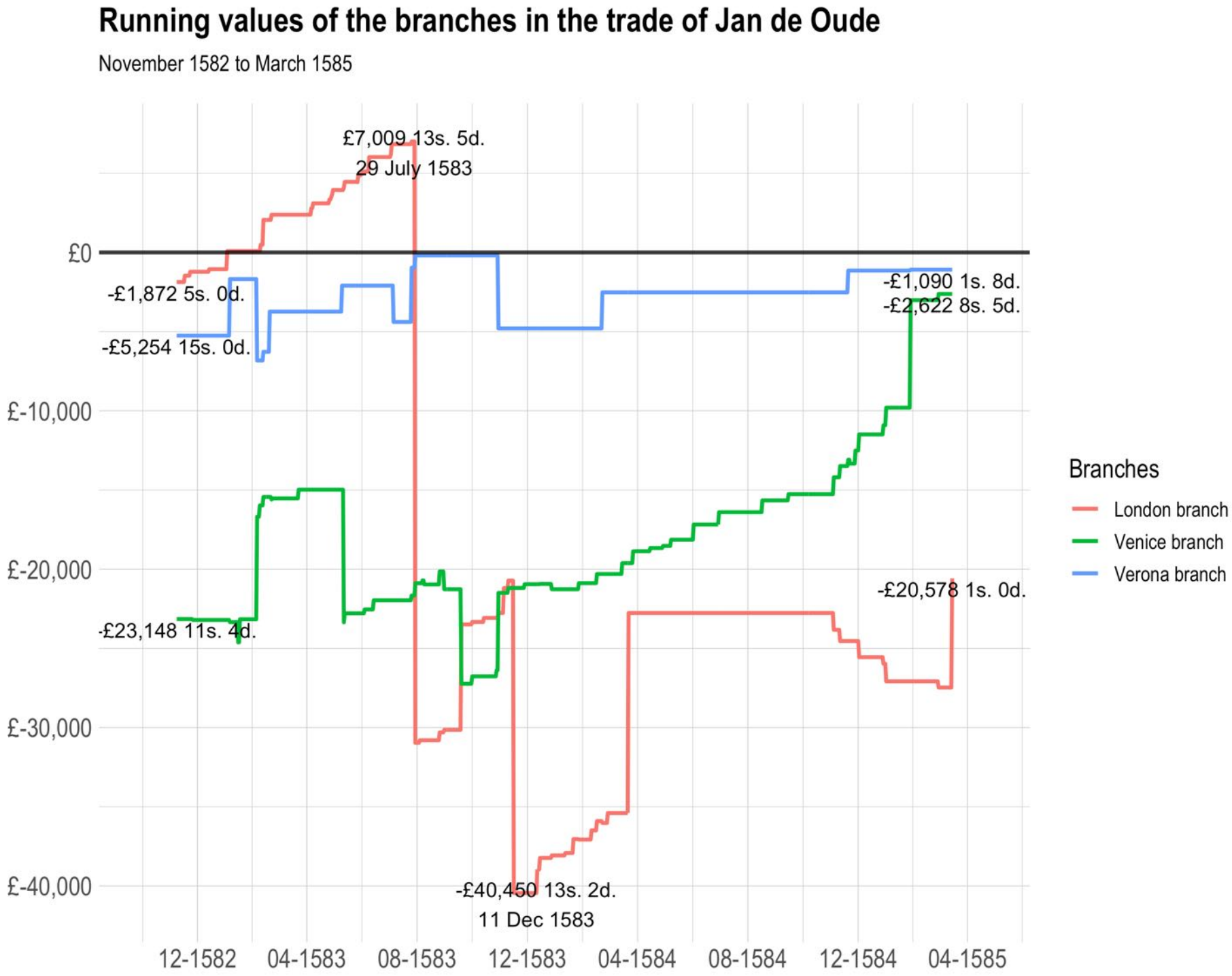


Inheritance due to the heirs of Jan de Oude

September 1594 to 16 December 1594

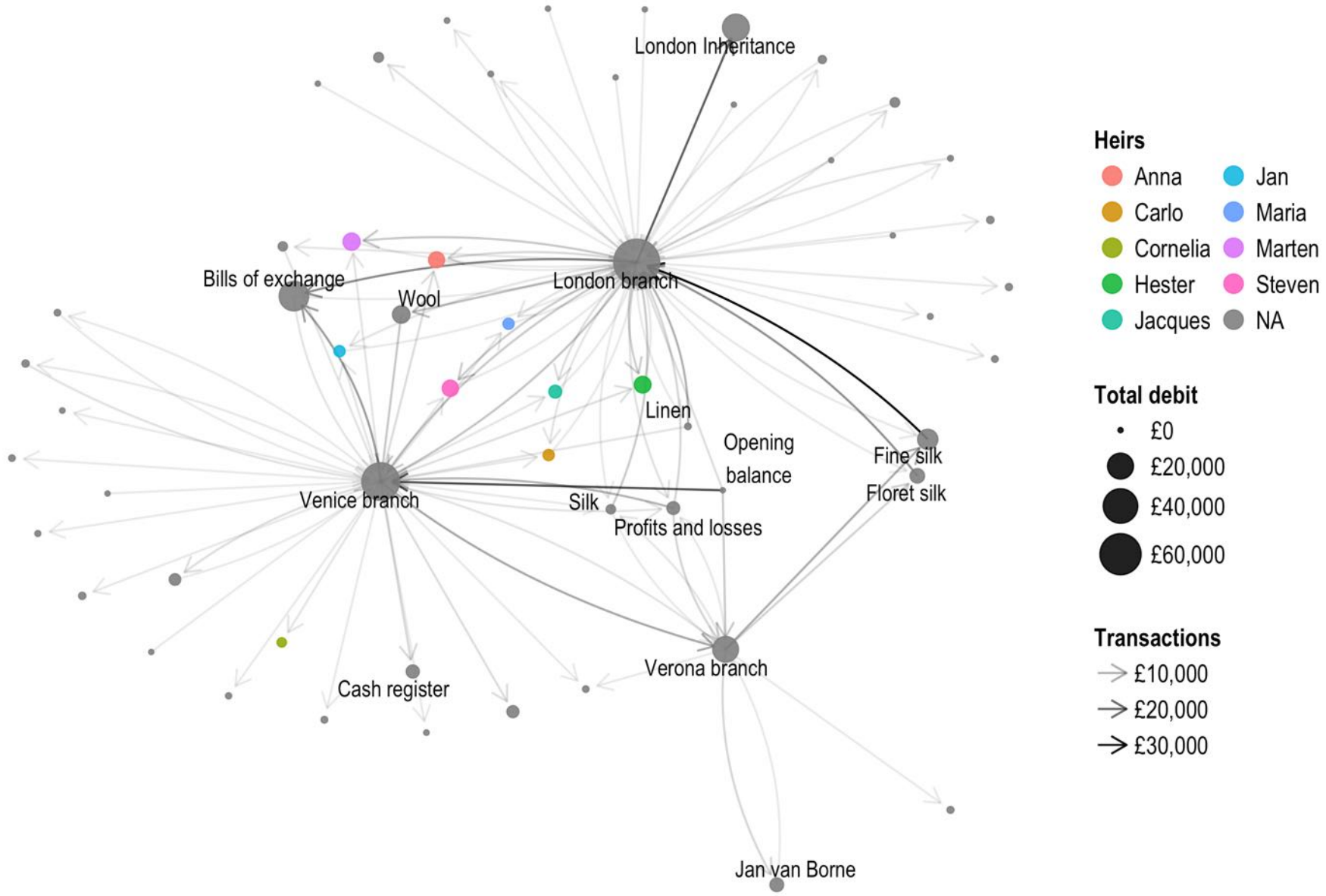



Running values of the branches in the trade of Jan de Oude
Estate of Jan della Faille de Oude, 1582–1594
£0
£-10,000
£-20,000
£-30,000
£-40,000
1584
1586
1588
1590
1592
1594
Branches
London branch
Venice branch
Verona branch


Running values of the branches in the trade of Jan de Oude
November 1582 to March 1585
£7,009 13s. 5d.
29 July 1583
£0
-£1,872 5s. 0d.
-£5,254 15s. 0d.
-£1,090 1s. 8d.
-£2,622 8s. 5d.
£-10,000
£-20,000
-£20,578 1s. 0d.
-£23,148 11s. 4d.
£-30,000
£-40,000
-£40,450 13s. 2d.
11 Dec 1583
12-1582
04-1583
08-1583
12-1583
04-1584
08-1584
12-1584
04-1585
Branches
London branch
Venice branch
Verona branch

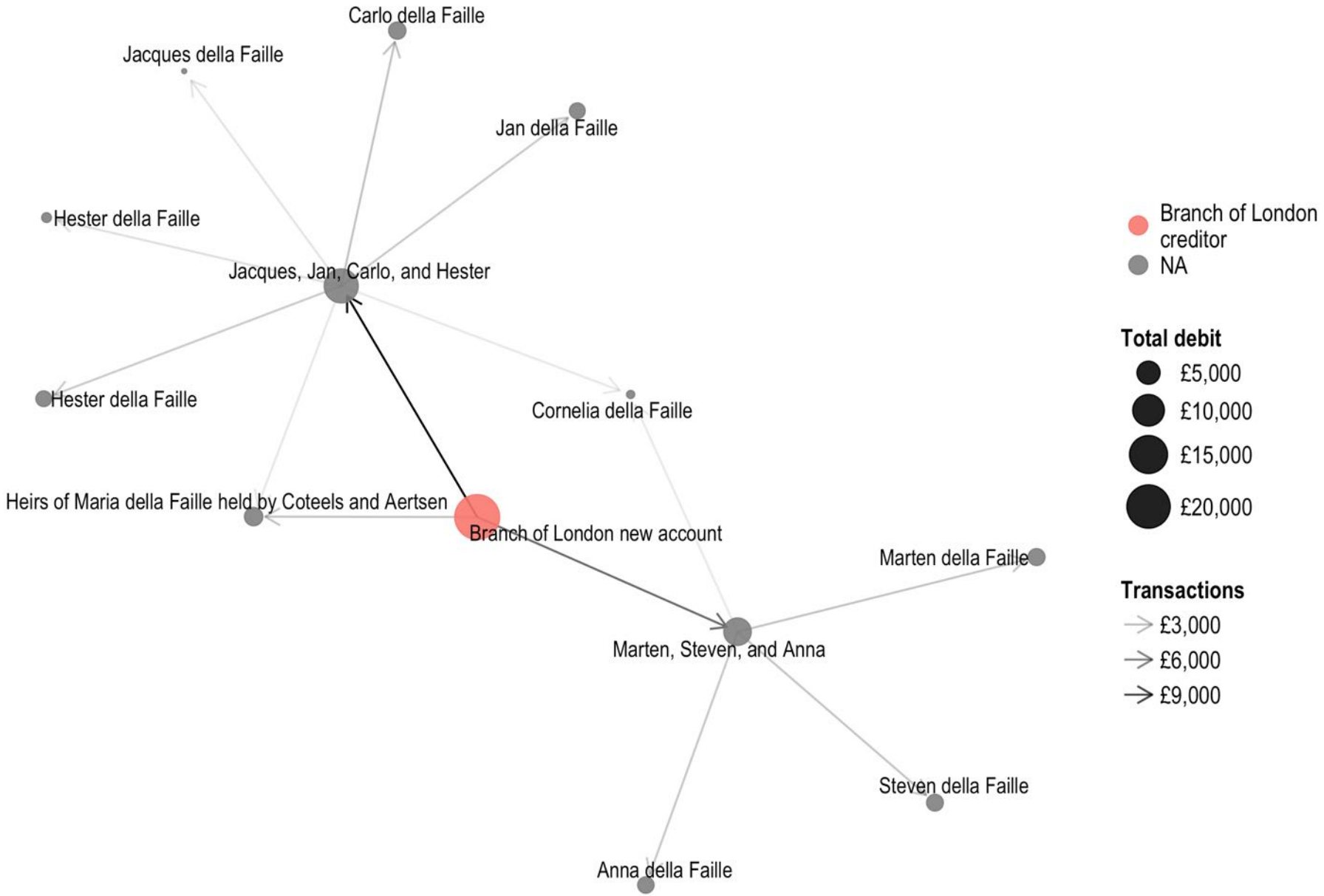
Subgraph of the branches in the trade of Jan de Oude

Estate of Jan della Faille de Oude, 1582–1594

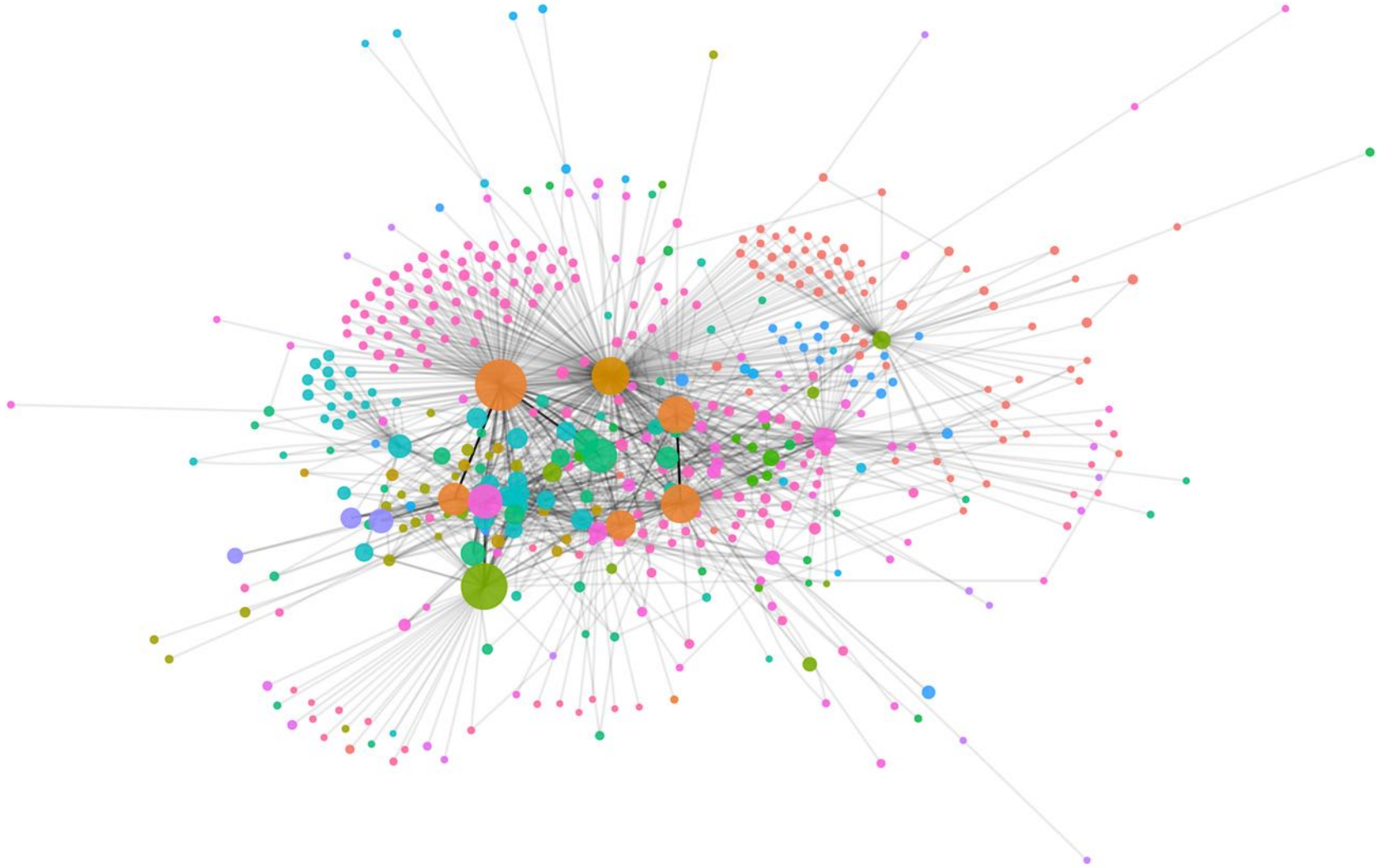


Subgraph of the inheritance from London

Estate of Jan della Faille de Oude, 1582–1594



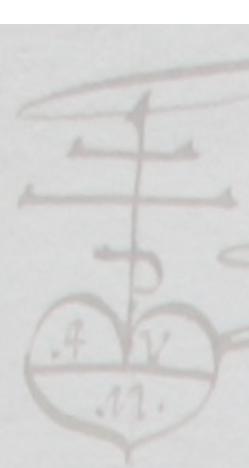
Estate of Jan della Faille de Oude, 1582–1594



Account types

Bequest · Branch · Cassa · Company · Erffgoed · Estate · Factor · Giovane · Goods · Heir

Inheritance · Kin · Law · Loan · London Inheritance · Miscellaneous · Political · Trade · Wissel · Written off



Thank you

Jesse Sadler

Twitter: @vivalosburros

website: jessesadler.com

GitHub: github.com/jessesadler

Slides: jessesadler.com/slides/chico2021.pdf