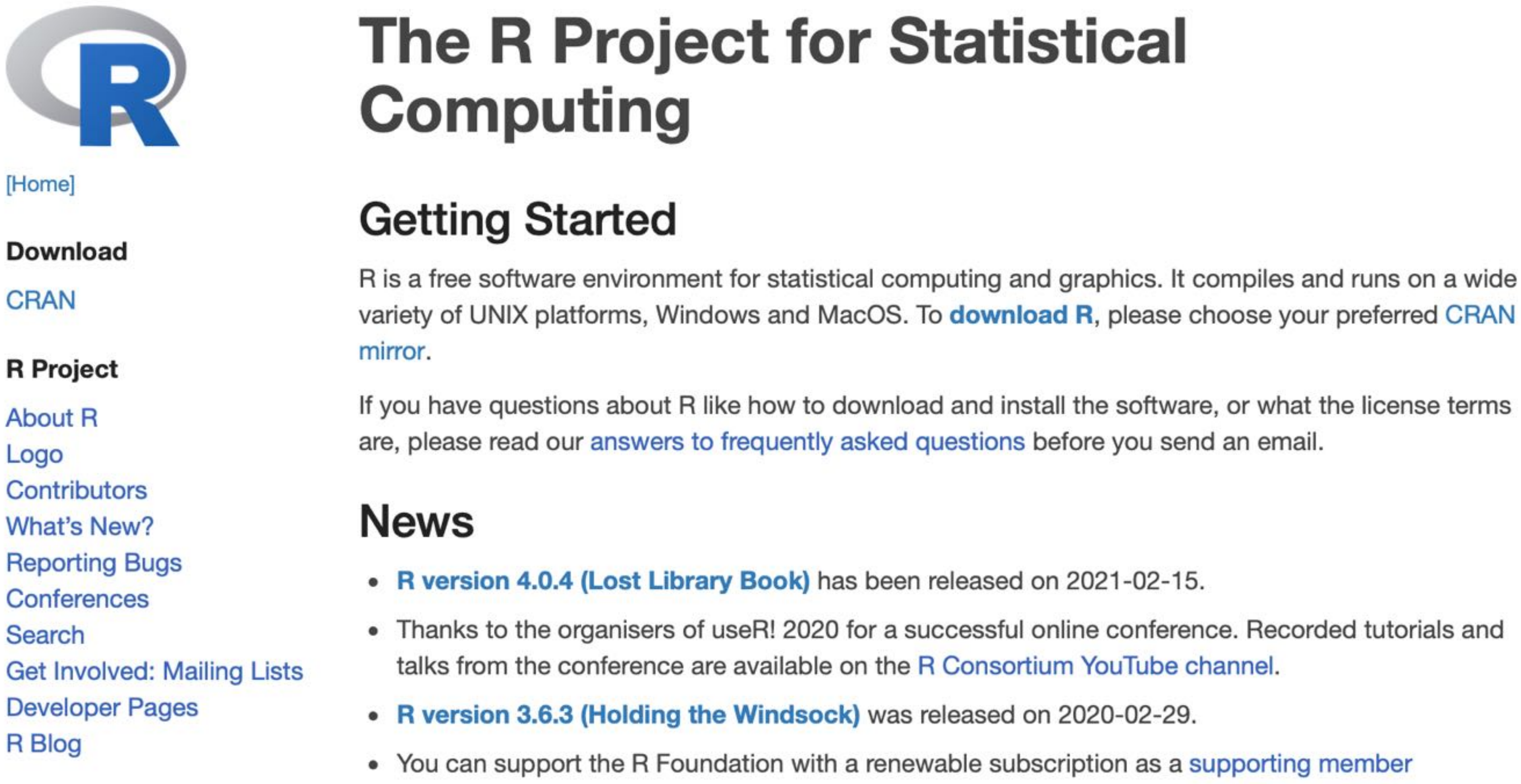
Analyzing and Visualizing Double-Entry Bookkeeping: A Digital History Methodology

Jesse Sadler
Virginia Tech

@vivalosburros
jessesadler.com
github.com/jessesadler
jessesadler.github.io/debkeepr

Slides: jessesadler.com/slides/rsa2021.pdf



[Home]

Download

CRAN

R Project

- About R
- Logo
- Contributors
- What's New?
- Reporting Bugs
- Conferences
- Search
- Get Involved: Mailing Lists
- Developer Pages
- R Blog

The R Project for Statistical Computing

Getting Started

R is a free software environment for statistical computing and graphics. It compiles and runs on a wide variety of UNIX platforms, Windows and MacOS. To **download R**, please choose your preferred CRAN mirror.

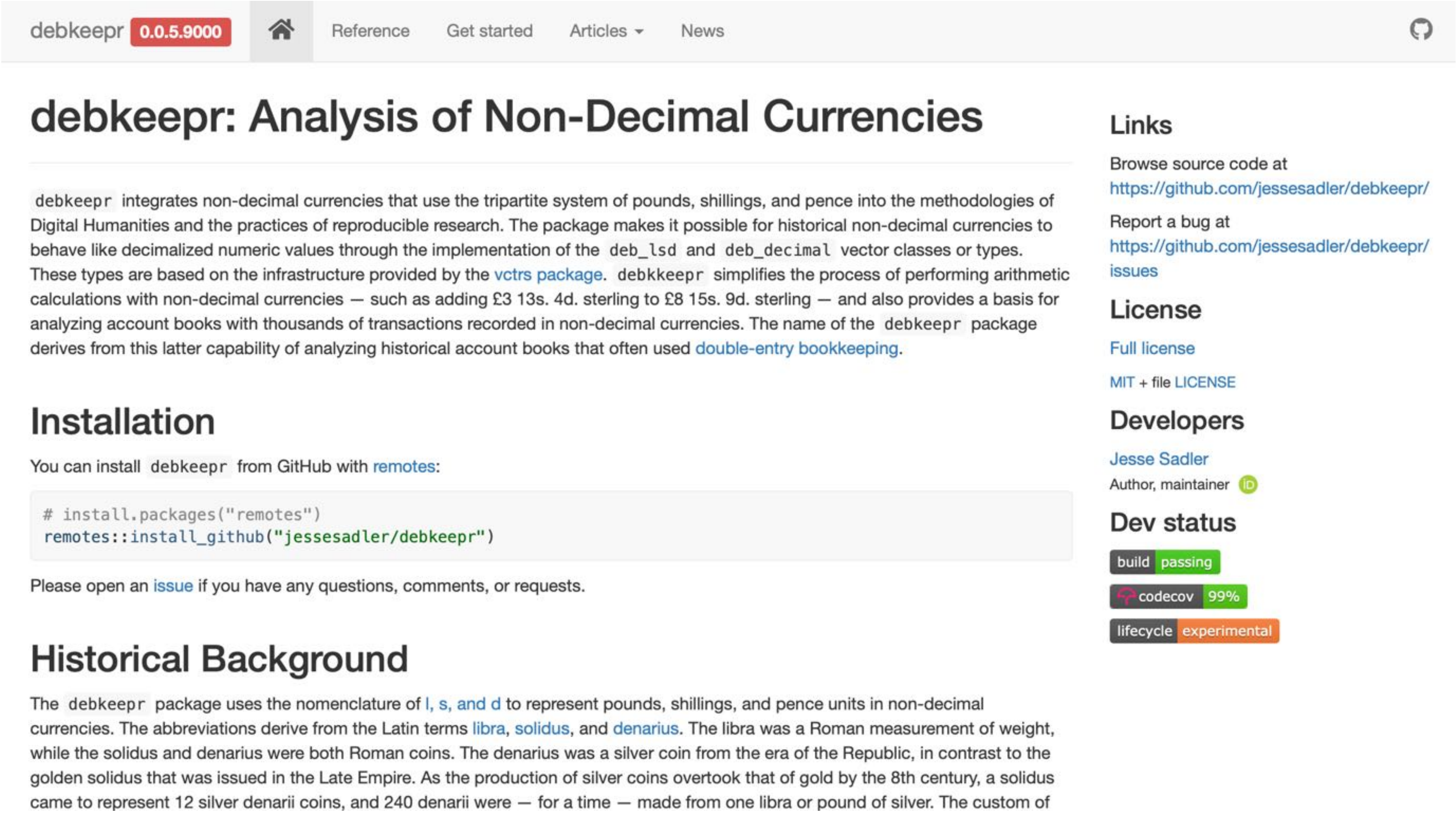
If you have questions about R like how to download and install the software, or what the license terms are, please read our answers to frequently asked questions before you send an email.

News

- **R version 4.0.4 (Lost Library Book)** has been released on 2021-02-15.
- Thanks to the organisers of useR! 2020 for a successful online conference. Recorded tutorials and talks from the conference are available on the R Consortium YouTube channel.
- **R version 3.6.3 (Holding the Windsock)** was released on 2020-02-29.
- You can support the R Foundation with a renewable subscription as a supporting member

debkeepr: Analysis of Non-Decimal Currencies in R

jessesadler.github.io/debkeepr/



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

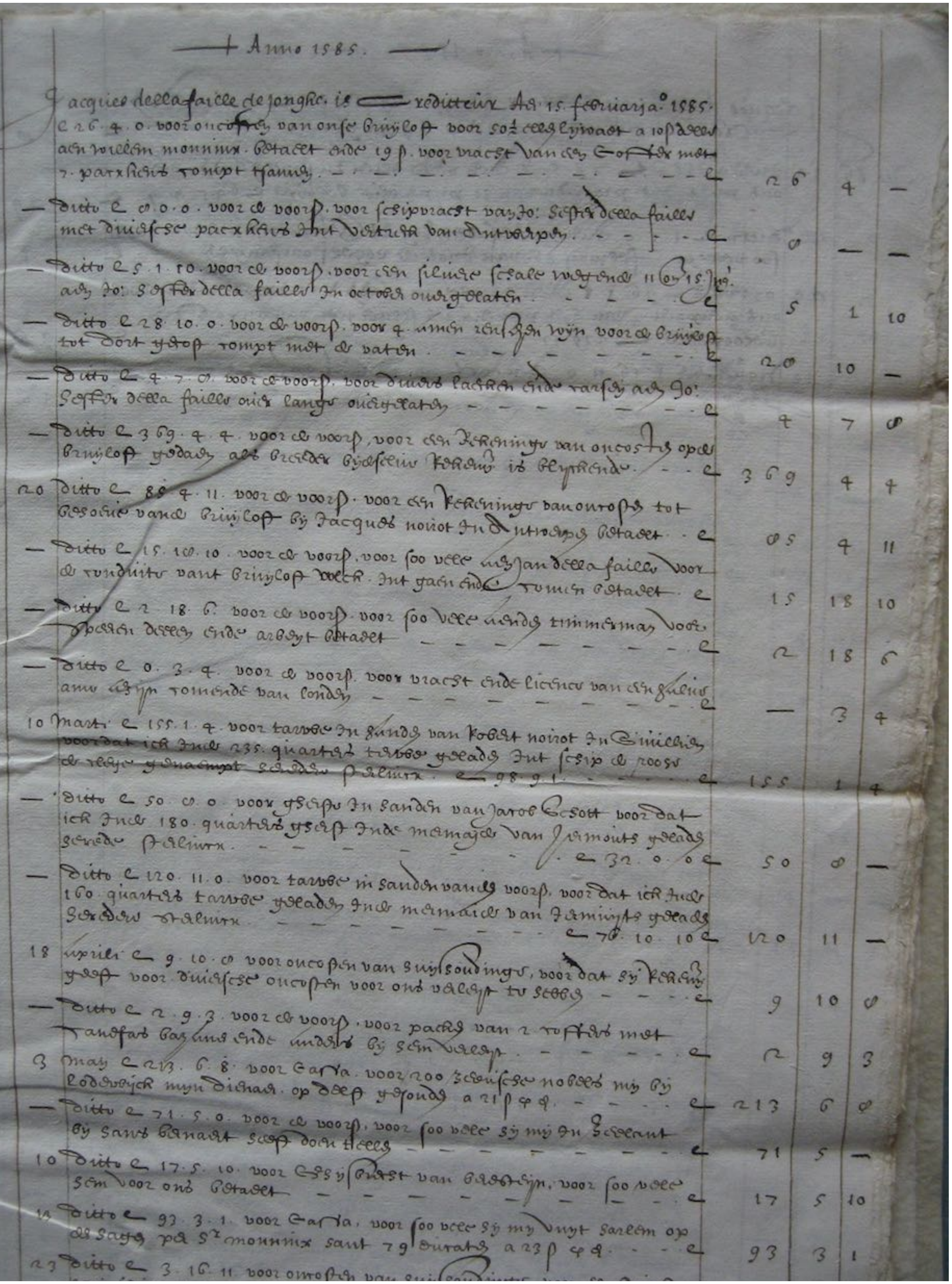
MIT + file LICENSE

Developers

Jesse Sadler
Author, maintainer

Dev status

Accounting



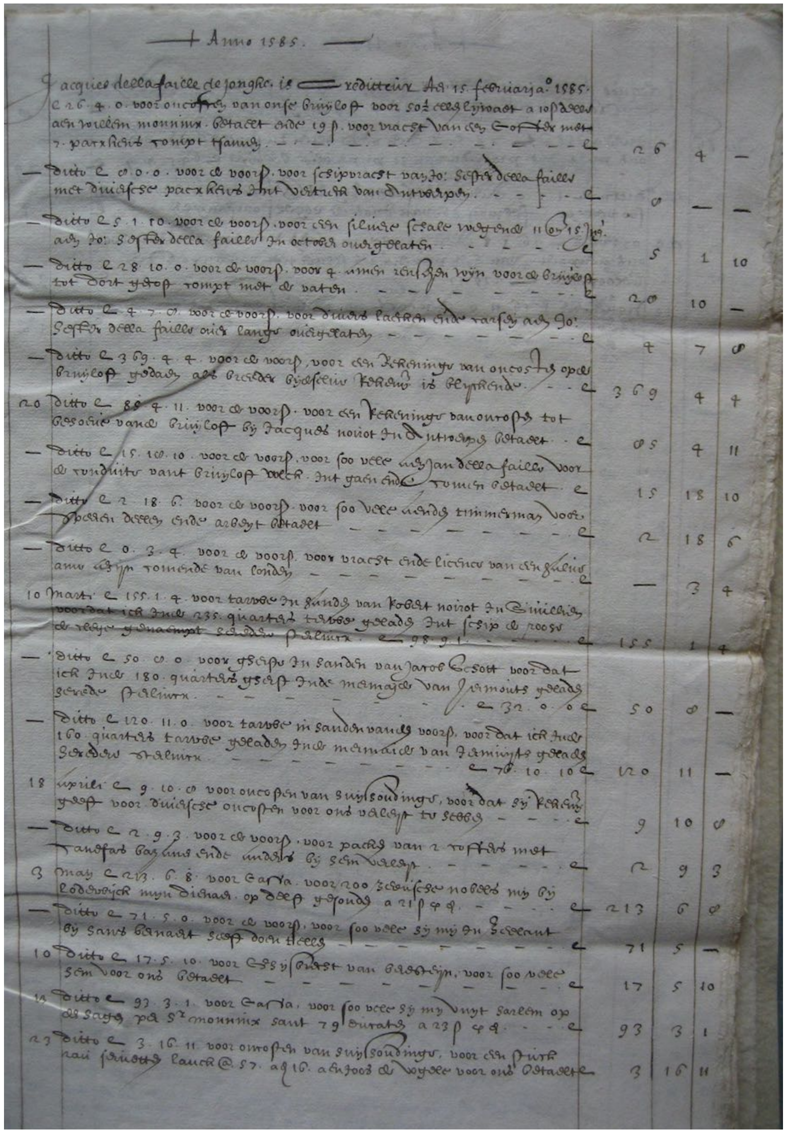
Anno 1585.

Jacques della faille de jonghe. is ——— crediteur Ao. 15. februarija ao 1585.

	£ 26. 4. 0. voor oncosten van onse bruijloft voor 50½ ells lywaet a 10 ß d'ell aen willem monnix betaelt ende 19 ß. voor vracht van een coffer met 7. packkens compt tsamen. — — — — — — — £	26	4	—
—	Ditto £ 0. 0. 0. voor de voors. voor schipvracht van Jo: Schoter della faille met diversche packkens uit Vertrecht van Antwerpen. — — £	0	—	—
—	Ditto £ 5. 1. 10. voor de voors. voor een silvere schale wegende 11 on 15 ing. aen Jo: Schoter della faille in october overgelaten — — — £	5	1	10
—	Ditto £ 28. 10. 0. voor de voors. voor 4. aimen rensschen wyn voor de bruijloft tot Dort gecocht compt met de vaten — — — — £	20	10	—
—	Ditto £ 4. 7. 0. voor de voors. voor diverse lacken ende carsaij aen Jo: Schoter della faille over langen overgelaten — — — — £	4	7	0
—	Ditto £ 369. 4. 4. voor de voors. voor een rekeninge van oncosten op de bruijloft gedaen als breeder bij de selve rekeninge is blyckende. — — £	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 veel 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 veel ick den timmerman voor sparren delen ende arbeyt betaelt — — — — £	2	18	6
—	Ditto £ 0. 3. 4. voor de voors. voor vracht ende licentie van een ghelein ammo cassijn comende van Londen — — — — £	—	3	4
10 Marti	£ 155. 1. 4. voor tarwe in handen van Robert noirot in Emmichem ~~voordat ick inde~~ 235. quartier tarwe geladen int schip de roose ~~de reyse genaempt schuldich~~ betalen. £ 98. 9. 1. — — £	155	1	4
—	Ditto £ 50. 0. 0 voor gerste in handen van Jacob Scott voor dat ick inde 180. quartiers gerst inde mamagie van Jermouts geladen gesonden betalen. — — — — £ 32. 0. 0 £	50	0	—
—	Ditto £ 120. 11. 0. voor tarwe in handen vande voors. voor dat ick inde 160 quartiers tarwe geladen inde mamaide van Jaminijts geladen schuldich betalen — — — — — £ 78. 10. 10 £	120	11	—
18 aprili	£ 9. 10. 0 voor oncosten van buijsoudinge, voor dat sij rekeninge gaf voor diversche oncosten voor ons valuit te stellen — — £	9	10	0
—	Ditto £ 2. 9. 3. voor de voors. voor packen van 2 coffers met tandefaes bassins ende ander bij hem geleyt. — — — £	2	9	3
3 May	£ 213. 6. 8. voor Carta. voor 200 zeeusche nobels mij bij Lodewijck myn dienaer op deese gesonden a 21 ß 4 d. — — £	213	6	8
—	Ditto £ 71. 5. 0. voor de voors. voor soo veel sij mij in zeelant bij Sants Banaert schip doen heels — — — — £	71	5	—
10	Ditto £ 17. 5. 10. voor verschenst van Baerdijn, voor soo veel hem voor ons betaelt — — — — — £	17	5	10
13	Ditto £ 93. 3. 1. voor Carta, voor soo veel sij mij uijt Haerlem op de Sage par Sr monnix sant 79 ducaten a 23 ß 4 d. — — £	93	3	1
23	Ditto £ 3. 16. 11 voor oncosten van buijsoudinge			

Accounting and digital history

- Quantitative analysis
- Visualization
- Reproducibility



Anno 1585.

Jacques della faille de jonghe is [illegible] crediteur Ad 15 februarij aº 1585.
£ 26. 4. 0. voor oncosten van onse bruijloft voor 50½ ells [illegible] a 10 s [illegible]
aen willem monnix betaelt ende 19 s. voor vracht van de [illegible] met
7. parckeels rompt [illegible] £ 26 4 —

— Ditto £ 0. 0. 0. voor de voors. voor schipvracht van Jo: Schetter della faille
met duijtsche packeels int [illegible] van Antwerpen £ 0 — —

— Ditto £ 5. 1. 10. voor de voors. voor een silvere schale wegende 11 on 15 [illegible]
aen Jo: Schetter della faille in october overgelaten £ 5 1 10

— Ditto £ 20. 10. 0. voor de voors. voor 4 amen rinsschen wijn voor de bruijloft
tot Dort gecocht rompt met de vaten £ 20 10 —

— Ditto £ 4. 7. 0. voor de voors. voor diverse lacken ende [illegible] aen Jo:
Schetter della faille over langs overgelaten £ 4 7 0

— Ditto £ 369. 4. 4. voor de voors. voor een rekeninge van oncosten op de
bruijloft gedaen als breeder bij selve 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 vele mijn Jan della faille voor
de conduite vant bruijloft werck int gaen ende comen betaelt £ 15 18 10

— Ditto £ 2. 18. 6. voor de voors. voor soo vele [illegible] timmerman voor
[illegible] ende arbeyt betaelt £ 2 18 6

— Ditto £ 0. 3. 4. voor de voors. voor vracht ende licentie van [illegible]
[illegible] comende van Londen £ — 3 4

10 Marti £ 155. 1. 4. voor tarwe in Sanden van Robert noirot in [illegible]
voordat ick inde 235. quartiers tarwe geladen int schip de roose
de [illegible] [illegible] [illegible] £ 98. 9. 1. £ 155 1 4

— Ditto £ 50. 0. 0. voor gerste in Sanden van Jacob Scott voor dat
ick inde 180. quartiers gerst inde [illegible] van Jermouts geladen
[illegible] [illegible] £ 32. 0. 0. £ 50 0 —

— Ditto £ 120. 11. 0. voor tarwe in Sanden vande voors. voor dat ick inde
160 quartiers tarwe geladen inde [illegible] van [illegible] geladen
[illegible] [illegible] £ 76. 10. 10. £ 120 11 —

18 April £ 9. 10. 0. voor oncosten van [illegible], voor dat sij [illegible]
[illegible] voor duijtsche oncosten voor ons [illegible] te [illegible] £ 9 10 0

— Ditto £ 2. 9. 3. voor de voors. voor packe van 2 coffers met
[illegible] [illegible] ende [illegible] bij [illegible] £ 2 9 3

3 May £ 213. 6. 8. voor Carta. voor 200 [illegible] nobels mij bij
Lodewijck [illegible] op [illegible] [illegible] a 21 s 4 d £ 213 6 0

— Ditto £ 71. 5. 0. voor de voors. voor soo vele sij mij in Zeelant
bij Sant [illegible] [illegible] [illegible] £ 71 5 —

10 Ditto £ 17. 5. 10. voor [illegible] van [illegible], voor soo vele
[illegible] voor ons betaelt £ 17 5 10

13 Ditto £ 93. 3. 1. voor Carta, voor soo vele sij mij uijt Haerlem op
[illegible] [illegible] St monnix [illegible] 79 [illegible] a 23 s 4 d £ 93 3 1

27 Ditto £ 3. 16. 11. voor oncosten van [illegible], voor een stuck
[illegible] [illegible] lanck @ 57. a 16. [illegible] de [illegible] voor ons betaelt £ 3 16 11

Non-decimal currency nomenclature
lsd

Latin units	libra	solidus	denarius
Abbreviation	£	s.	d.
English	Pound	shilling	penny (pence)
Bases	∞	20	12

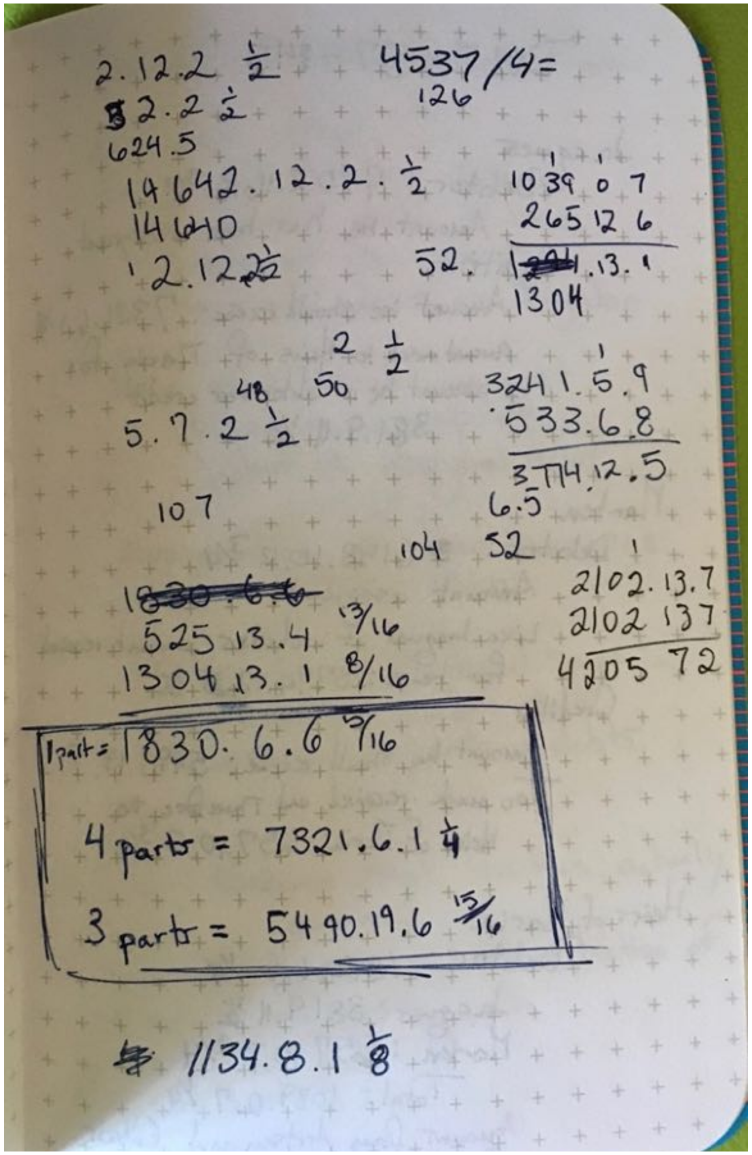
Names for non-decimal currencies

- Pound sterling: pound_{∞}, shilling_{20}, penny_{12}
- Pound Flemish: pond_{∞}, schelling_{20}, groot_{12}
- Holland guilders: guilder_{∞}, stuiver_{20}, penning_{16}
- French crowns: crown_{∞}, sous_{60}, denier_{12}
- Polish florins: florin_{∞}, gros_{30}, denar_{18}

Difficulties with non-decimal currencies

- One value is represented by three separate units
- The solidus and denarius units have non-decimal bases
- The bases could differ by currency

Arithmetic



2.12.2 ½ 4537/4=
126
~~5~~2.2 ½
624.5
14642.12.2.½
14640
2.12.2½

1039 0 7
265 12 6
52. 1~~[illegible]~~.13.1
1304

2 ½
48 50
5.7.2 ½

3241.5.9
533.6.8
3714.12.5
6.5

107
104 52

~~1830.6.6~~
525 13.4 13/16
1304 13.1 8/16

2102.13.7
2102 137
4205 72

1 part = 1830. 6.6 5/16

4 parts = 7321.6.1 ¼

3 parts = 5490.19.6 15/16

~~½~~ 1134.8.1 ⅛

£	s.	d.
28	15	8
32	8	11
54	18	7
18	12	9

Compound unit arithmetic

	£	s.	d.
	28	15	8
	32	8	11
	54	18	7
	18	12	9
Unit total	132	53	35

Compound unit arithmetic

		£	s.	d.
		28	15	8
		32	8	11
		54	18	7
		18	12	9
	Unit total	132	53	35
Normalization	Carried forward	2	2	-
	Divide by base	-	55 / 20	35 / 12
	Remainder	-	15	11
	Answer	**£134**	**15s.**	**11d.**

Decimalization

£	s.	d.	Decimal
28	15	8	28.78333
32	8	11	32.44583
54	18	7	54.92917
18	12	9	18.63750

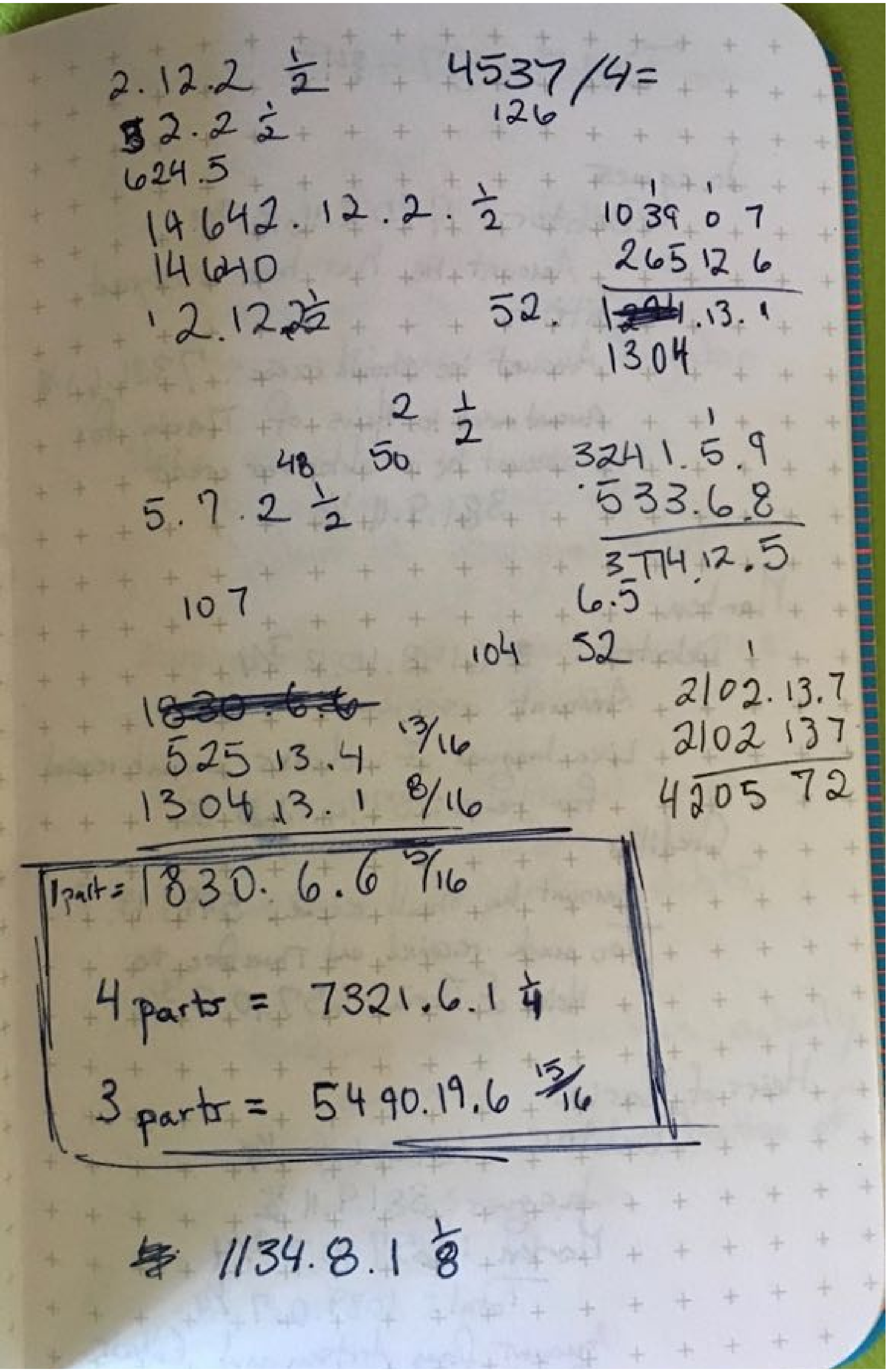
Decimalization

	£	s.	d.	Decimal
	28	15	8	28.78333
	32	8	11	32.44583
	54	18	7	54.92917
	18	12	9	18.63750
Total				134.7958

Decimalization

	£	s.	d.	Decimal
	28	15	8	28.78333
	32	8	11	32.44583
	54	18	7	54.92917
	18	12	9	18.63750
Total				134.7958
Non-decimal	?	?	?	

Arithmetic by hand



2.12.2 ½ 4537/4=

126

~~5~~2.2 ½

624.5

14642.12.2.½ 1039 0 7

14640 265 12 6

1 2.12.2½ 52. ~~1291~~.13.1

1304

2 ½

48 50 3241.5.9

5.7.2 ½ 533.6.8

3774.12.5

107 6.5

104 52 2102.13.7

~~1830.6.6~~ 2102 137

525 13.4 13/16 4205 72

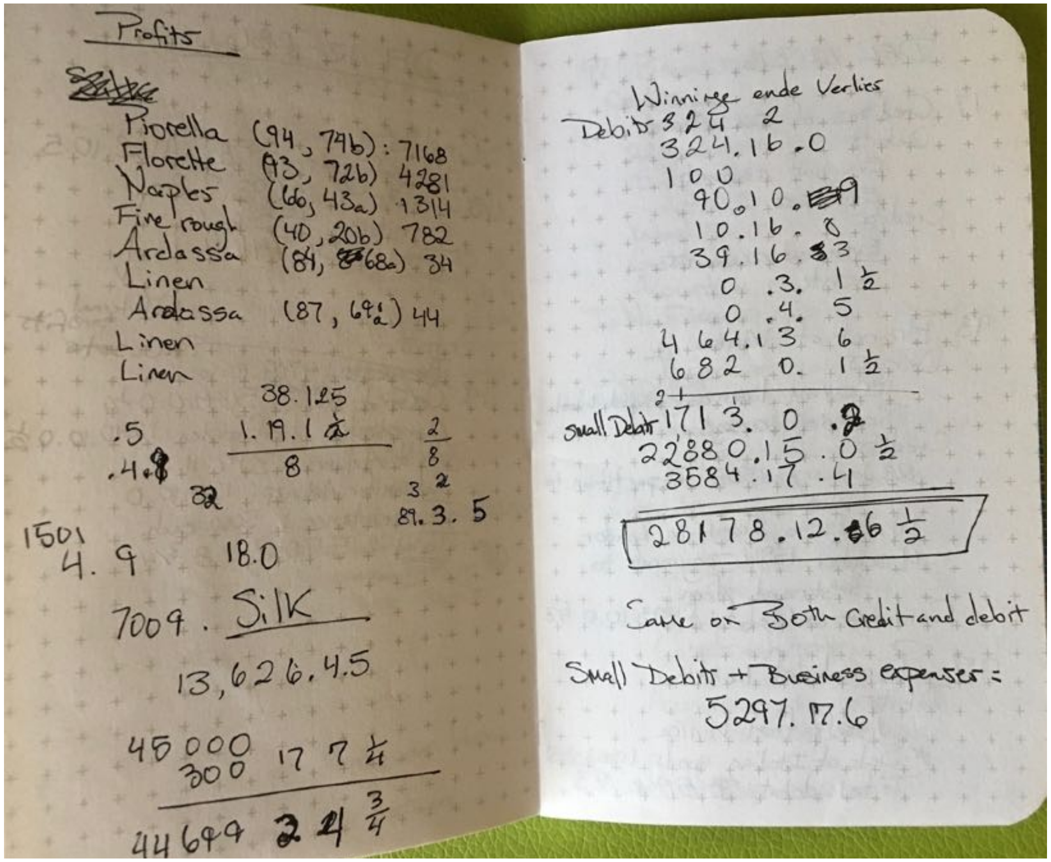
1304 13.1 8/16

1 part = 1830.6.6 5/16

4 parts = 7321.6.1 ¼

3 parts = 5490.19.6 15/16

~~5~~ 1134.8.1 ⅛



Profits

Fiorella (94, 74b): 7168

Florette (93, 72b) 4281

Naples (66, 43a) 1314

Fine rough (40, 20b) 782

Ardassa (84, ~~8~~68a) 34

Linen

Ardassa (87, 69½) 44

Linen

Linen

38.125

.5 1.19.1 ½ 2/8

.4.8 8

32 3 2

89.3.5

1501

4. 9 18.0

7009. Silk

13,626.4.5

45 000 17 7 ¼

300

44699 2 4 ¾

Winninge ende Verlies

Debits 324 2

324.16.0

100

90.10.~~8~~9

10.16.8

39.16.~~8~~3

0.3. 1 ½

0.4. 5

464.13 6

682 0. 1 ½

Small Debts 1713. 0 .2

22880.15.0 ½

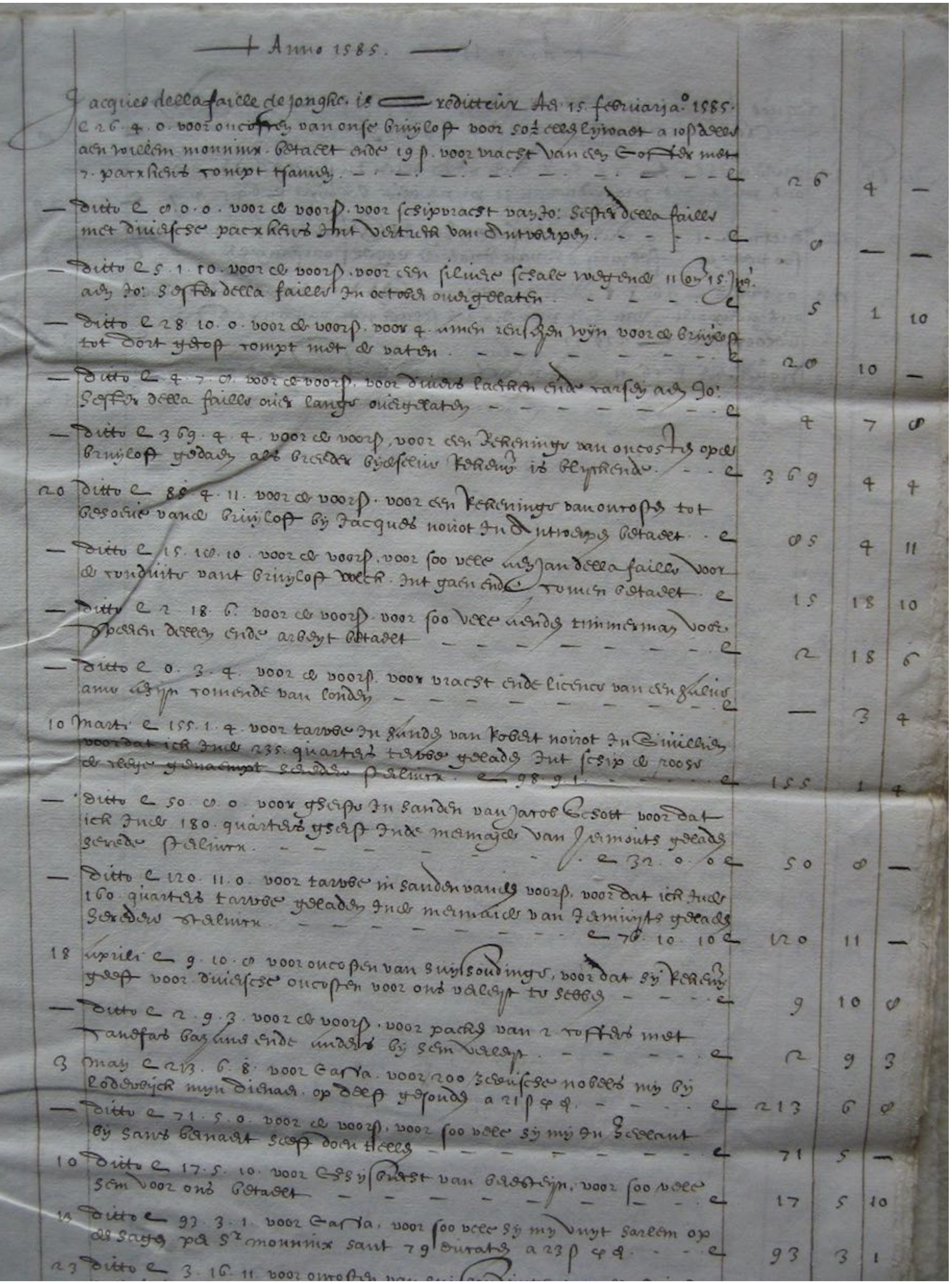
3584.17.4

28178.12.~~8~~6 ½

Same on Both Credit and debit

Small Debits + Business expenses =

5297.17.6



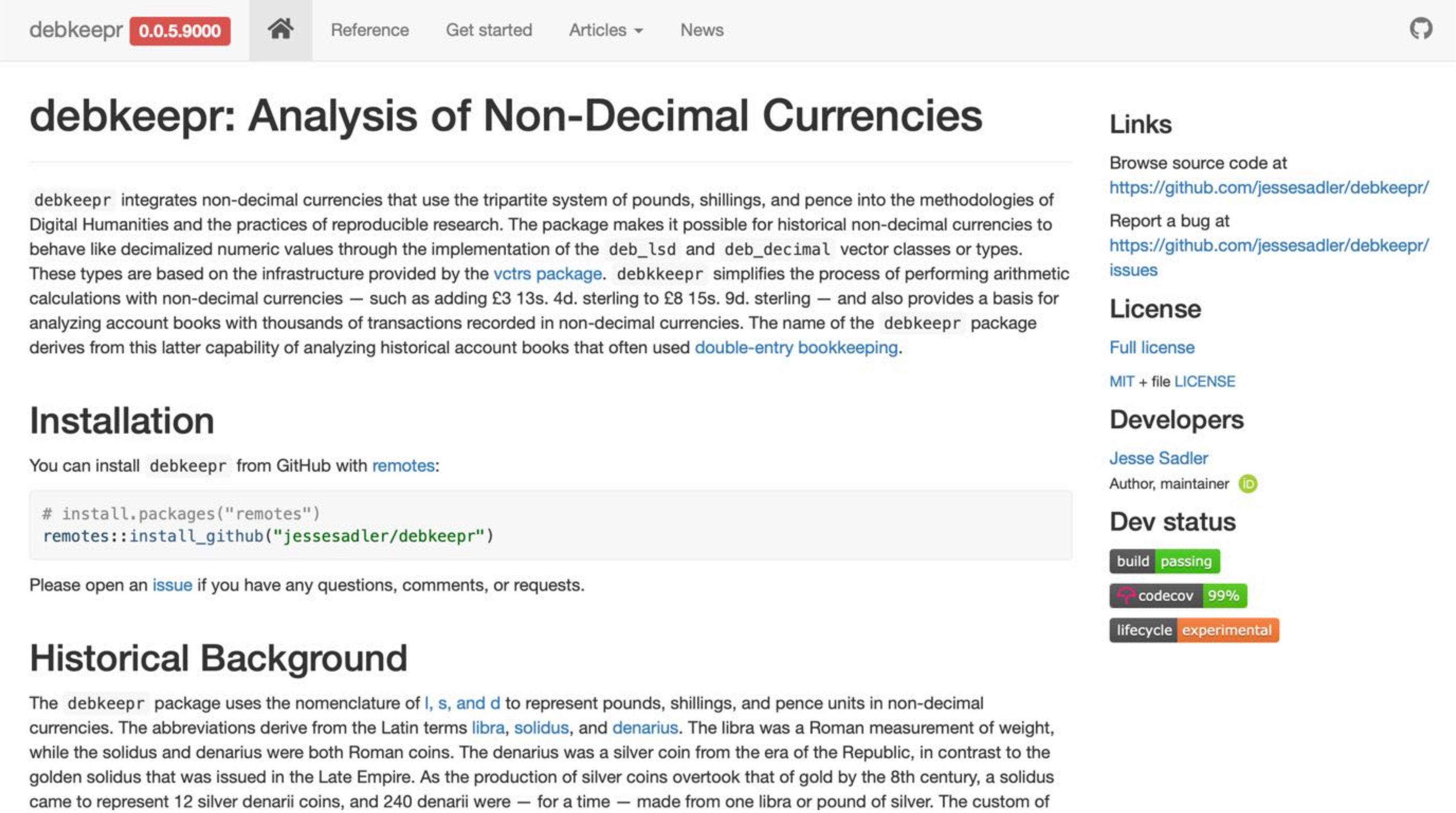
Anno 1585.

	Jacques della faille de jonghe. is crediteur Adi 15 februarij aº 1585. £ 26. 4. 0. voor oncosten van onse Bruijloft voor 50½ ell. lywaet a 10 s. d'eel aen Willem Monnincx betaelt ende 19 s. voor vracht van een coffer met 7 packetten compt t'samen	26	4	—
—	ditto £ 0. 0. 0. voor de voors. voor schipvracht van Jo: Schetter della failles met diversche packetten uut Vertrecht van Antwerpen	0	—	—
—	ditto £ 5. 1. 10. voor de voors. voor een silvere schale weghende 11 on 15 ing. aen Jo: Schetter della failles in october overghelaten	5	1	10
—	ditto £ 28. 10. 0. voor de voors. voor 4 aimen rinschen wyn voor de Bruijloft tot Dort ghecocht compt met de vaten	20	10	—
—	ditto £ 4. 7. 0. voor de voors. voor diverse lacken ende varschen aen Jo: Schetter della failles over langhe overghelaten	4	7	0
—	ditto £ 369. 4. 4. voor de voors., voor een rekeninghe van oncosten op de Bruijloft ghedaen als breeder bij deselve rekeninghe is blijckende	369	4	4
20	ditto £ 85. 4. 11. voor de voors. voor een rekeninghe van oncosten tot behoeve vande Bruijloft bij Jacques Noirot in Antwerpen betaelt	05	4	11
—	ditto £ 15. 18. 10. voor de voors., voor soo veel 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 veel aende timmermans voor [illegible] deelen ende arbeyt betaelt	2	18	6
—	ditto £ 0. 3. 4. voor de voors. voor vracht ende licentie van eenighe [illegible] comende van Londen	—	3	4
10 Marti	£ 155. 1. 4. voor tarwe in handen van Robert Noirot in Enckhuijsen voordat ick inde 235 quarters tarwe gheladen int schip de Roose ~~de reyse ghenaempt schipper~~ [illegible] £ 98. 9. 1	155	1	4
—	ditto £ 50. 0. 0 voor gherste in handen van Jacob Scott voor dat ick inde 180 quarters gherst inde [illegible] van Jermouts gheladen schipper [illegible] £ 32. 0. 0	50	0	—
—	ditto £ 120. 11. 0. voor tarwe in handen vanden voors. voor dat ick inde 160 quarters tarwe gheladen inde [illegible] van Jaminghs gheladen schipper [illegible] £ 76. 10. 10	120	11	—
18 April	£ 9. 10. 0 voor oncosten van huijshoudinghe, voor dat sij rekeninghe gheeft voor diversche oncosten voor ons [illegible] te [illegible]	9	10	0
—	ditto £ 2. 9. 3. voor de voors., voor packen van 2 coffers met [illegible] ende anders bij sijn [illegible]	2	9	3
3 May	£ 213. 6. 8. voor Carta. voor 200 Engelsche nobels mij bij Lodowijck van Dichaer op deese ghesonden a 21 s 4 d.	213	6	0
—	ditto £ 71. 5. 0. voor de voors., voor soo veel sij mij in Zeelant bij Sander Lenaert [illegible]	71	5	—
10	ditto £ 17. 5. 10. voor [illegible] van [illegible], voor soo veel sijn voor ons betaelt	17	5	10
14	ditto £ 93. 3. 1. voor Carta, voor soo veel sij mij uut Haerlem op de Saye [illegible] Sr Monnincx sant 79 ducaten a 23 s 4 d.	93	3	1
23	ditto £ 3. 16. 11 voor oncosten van [illegible]			

How to enter the values into a spreadsheet?

debkeepr: Analysis of Non-Decimal Currencies in R

jessesadler.github.io/debkeepr/



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

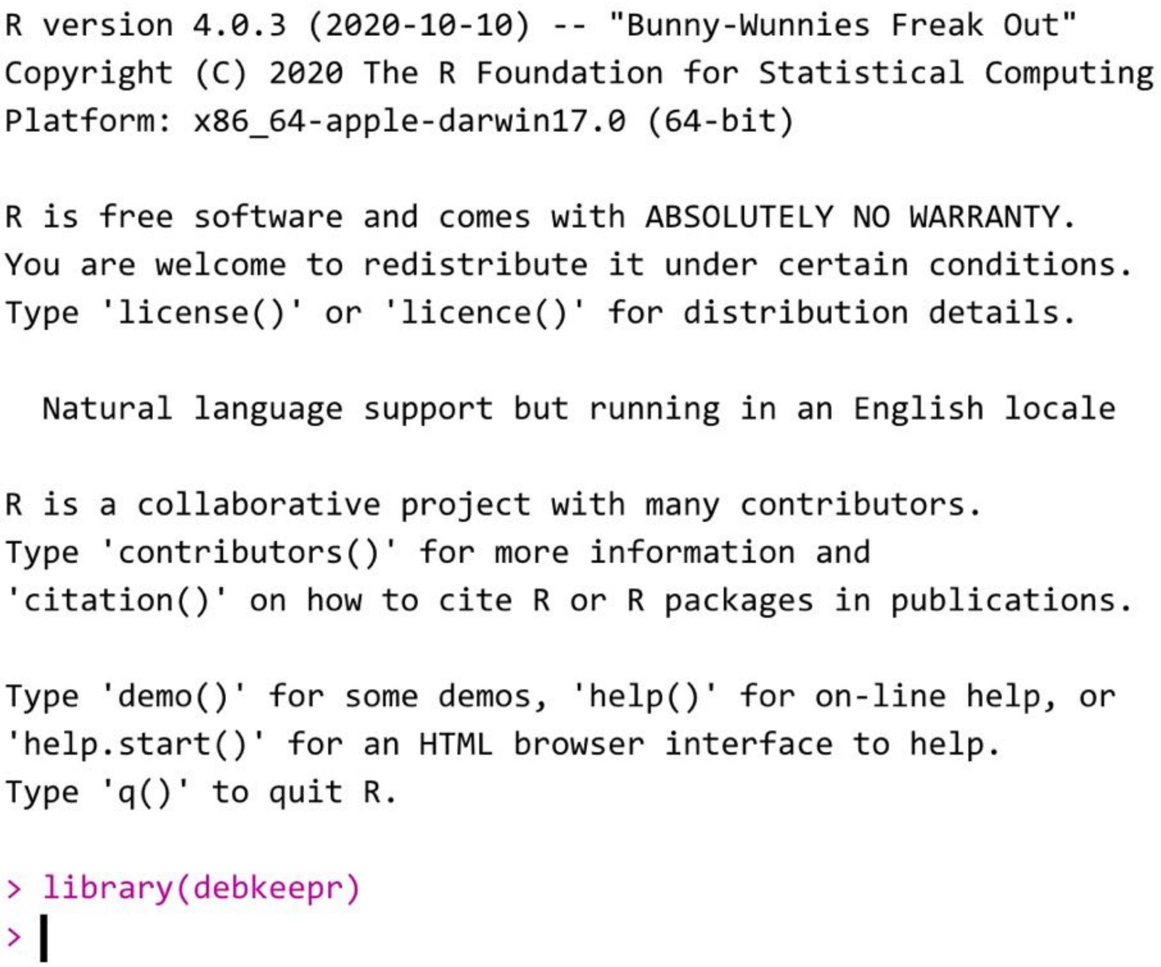
MIT + file LICENSE

Developers

Jesse Sadler
Author, maintainer

Dev status

Normalization



```
R version 4.0.3 (2020-10-10) -- "Bunny-Wunnies Freak Out"
Copyright (C) 2020 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin17.0 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> library(debkeepr)
> 
```

Compound unit

	£	s.	d.
	28	15	8
	32	8	11
	54	18	7
	18	12	9
Unit total	132	53	35
Carried	2	2	-
Divide by	-	55 / 20	35 / 12
Remainder	-	15	11
Answer	**£134**	**15s.**	**11d.**

Structure of the classes

Tripartite structure

`deb_lsd`

```
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`

```
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
```

Structure of the classes

Tripartite structure

`deb_lsd`

```
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
```

Class and length

Values

Unit attribute

Bases attribute

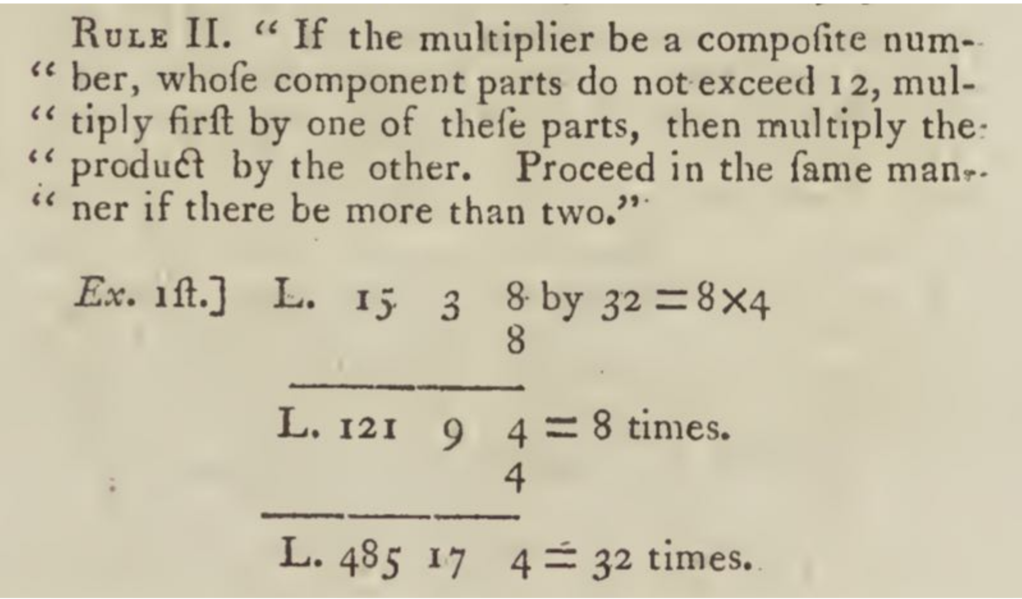
Decimalized

`deb_decimal`

```
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
```

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

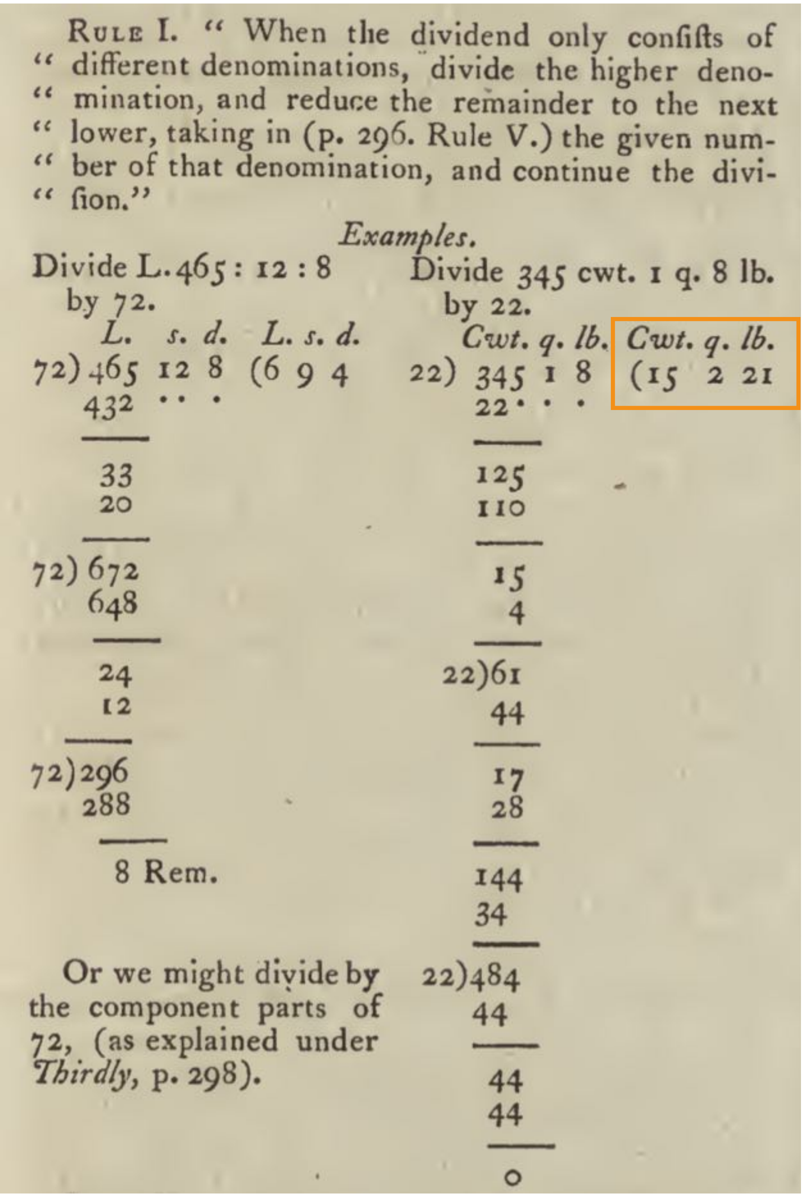
L. 121 9 4 = 8 times.

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 confiſts 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.
```

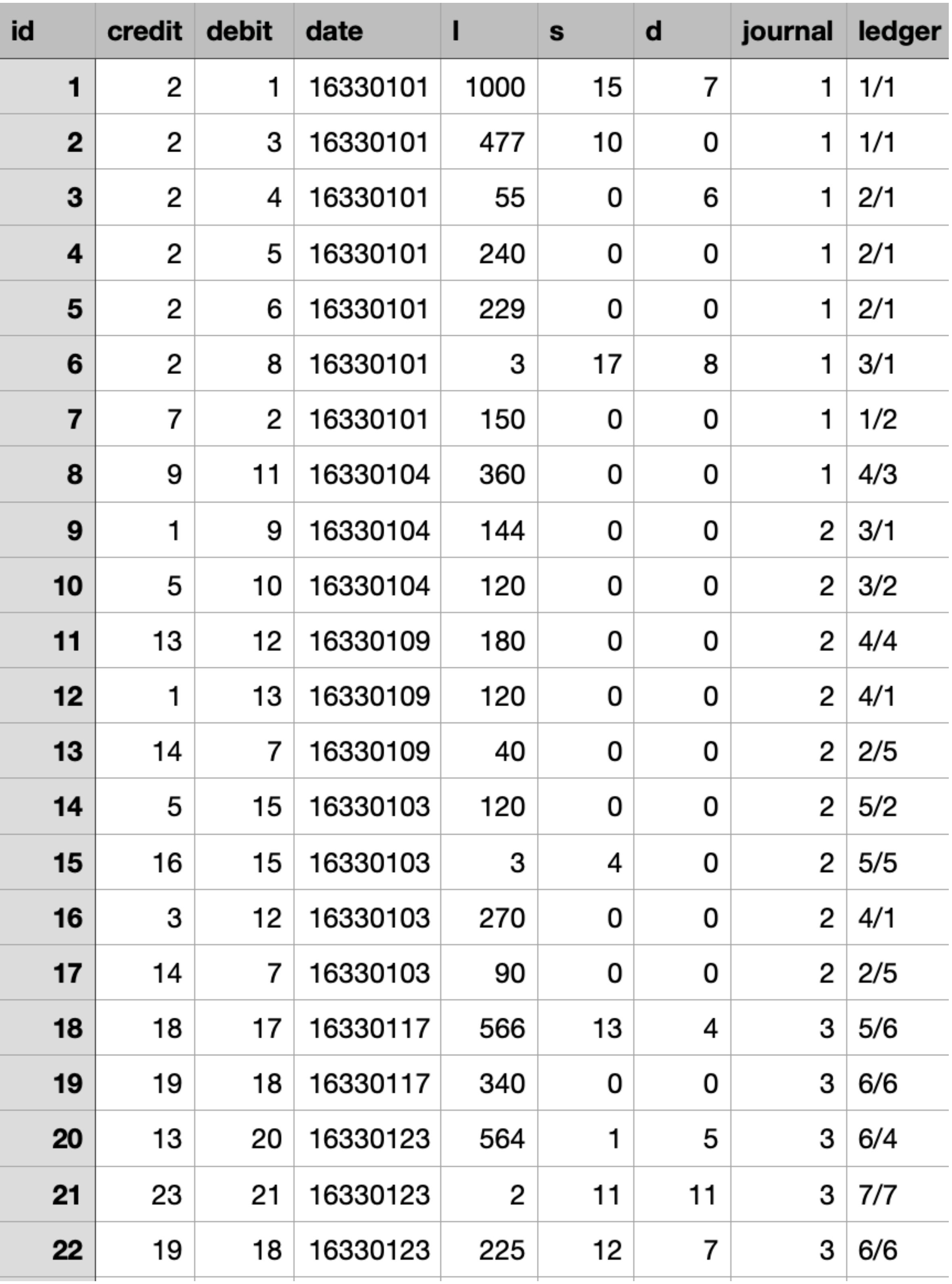
Or we might divide by the component parts of 72, (as explained under *Thirdly*, p. 298).

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
```

```r
# See the debkeepr website for more examples

x <- deb_lsd(l = c(28, 32, 54, 18),
             s = c(15, 8, 18, 12),
             d = c(8, 11, 7, 9))
sum(x)

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

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

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

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

#> <deb_lsd[1]>
#> [1] 9:2s:5d
#> # Bases: 20s 12d
```



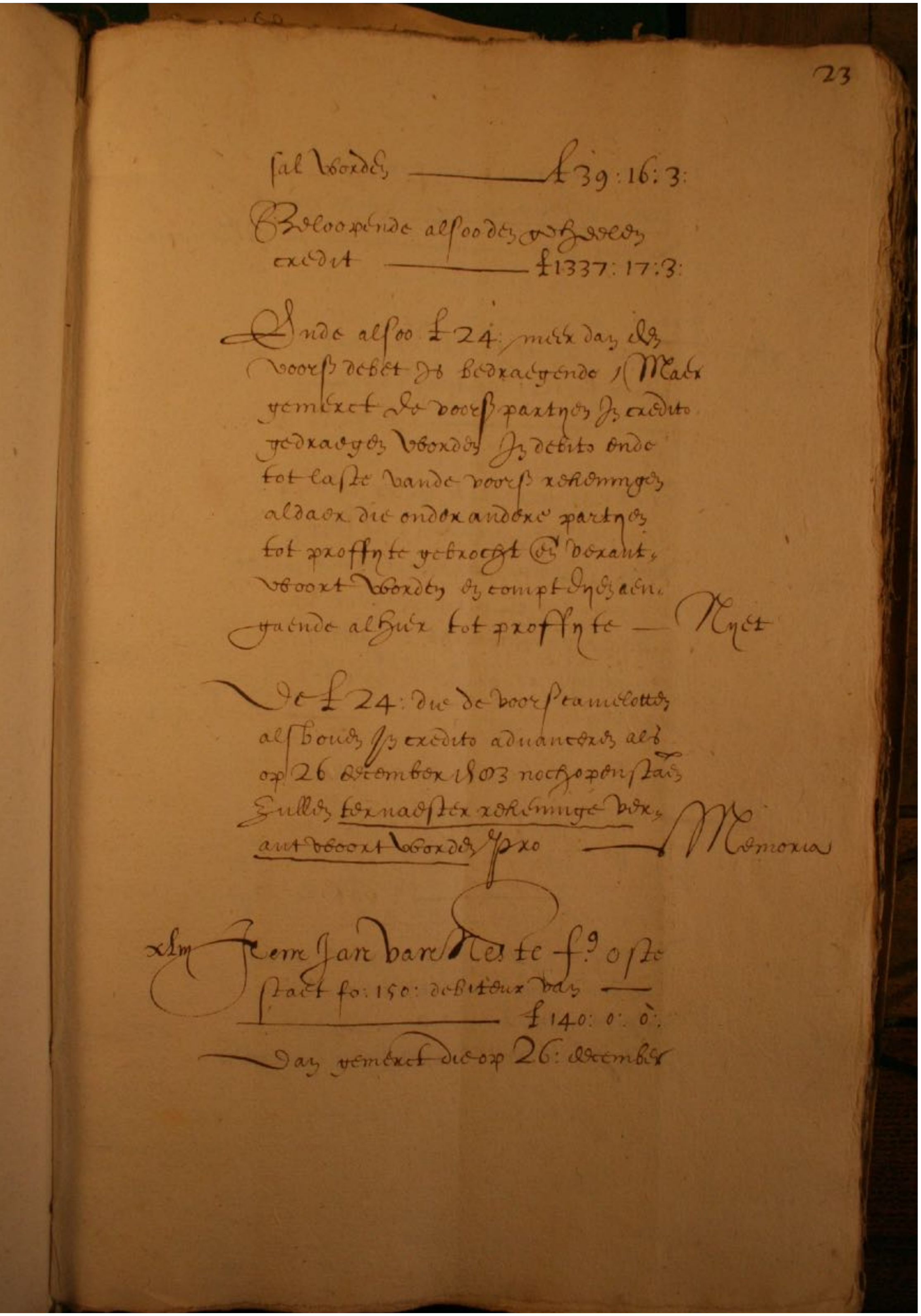
id	credit	debit	date	l	s	d	journal	ledger
1	2	1	16330101	1000	15	7	1	1/1
2	2	3	16330101	477	10	0	1	1/1
3	2	4	16330101	55	0	6	1	2/1
4	2	5	16330101	240	0	0	1	2/1
5	2	6	16330101	229	0	0	1	2/1
6	2	8	16330101	3	17	8	1	3/1
7	7	2	16330101	150	0	0	1	1/2
8	9	11	16330104	360	0	0	1	4/3
9	1	9	16330104	144	0	0	2	3/1
10	5	10	16330104	120	0	0	2	3/2
11	13	12	16330109	180	0	0	2	4/4
12	1	13	16330109	120	0	0	2	4/1
13	14	7	16330109	40	0	0	2	2/5
14	5	15	16330103	120	0	0	2	5/2
15	16	15	16330103	3	4	0	2	5/5
16	3	12	16330103	270	0	0	2	4/1
17	14	7	16330103	90	0	0	2	2/5
18	18	17	16330117	566	13	4	3	5/6
19	19	18	16330117	340	0	0	3	6/6
20	13	20	16330123	564	1	5	3	6/4
21	23	21	16330123	2	11	11	3	7/7
22	19	18	16330123	225	12	7	3	6/6

Entering data into a spreadsheet

```
# Create a data frame with a deb_lsd column

deb_gather_lsd()
```

```
# A tibble: 177 x 5
      id credit debit date                  lsd
   <dbl>  <dbl> <dbl> <date>     <lsd[20s:12d]>
 1     1      2     1 1633-01-01    1000:15s:7d
 2     2      2     3 1633-01-01     477:10s:0d
 3     3      2     4 1633-01-01       55:0s:6d
 4     4      2     5 1633-01-01      240:0s:0d
 5     5      2     6 1633-01-01      229:0s:0d
 6     6      2     8 1633-01-01       3:17s:8d
 7     7      7     2 1633-01-01      150:0s:0d
 8     8      9    11 1633-01-04      360:0s:0d
 9     9      1     9 1633-01-04      144:0s:0d
10    10      5    10 1633-01-04      120:0s:0d
# … with 167 more rows
```

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



23

sal worden —— £39:16:3:

Beloopende alsoo den ontheven
credit —— £1337:17:3:

Ende alsoo £24: meer dan den
voorss debet is bedraegende / Maer
gemeret de voorss partyen in credito
gedraegen worden in debito ende
tot laste vande voorss rekeningen
aldaer die onder andere partyen
tot proffyte gebrocht en verant,
voort worden en compt dyen aen-
gaende alhier tot proffyte — Niet

De £24: die de voorss camelotten
als bouen in credito aduanceren als
op 26 decembor 1583 noch openstaen
sullen ter naester rekeninge ver-
ant voort worden ipso —— Memoria

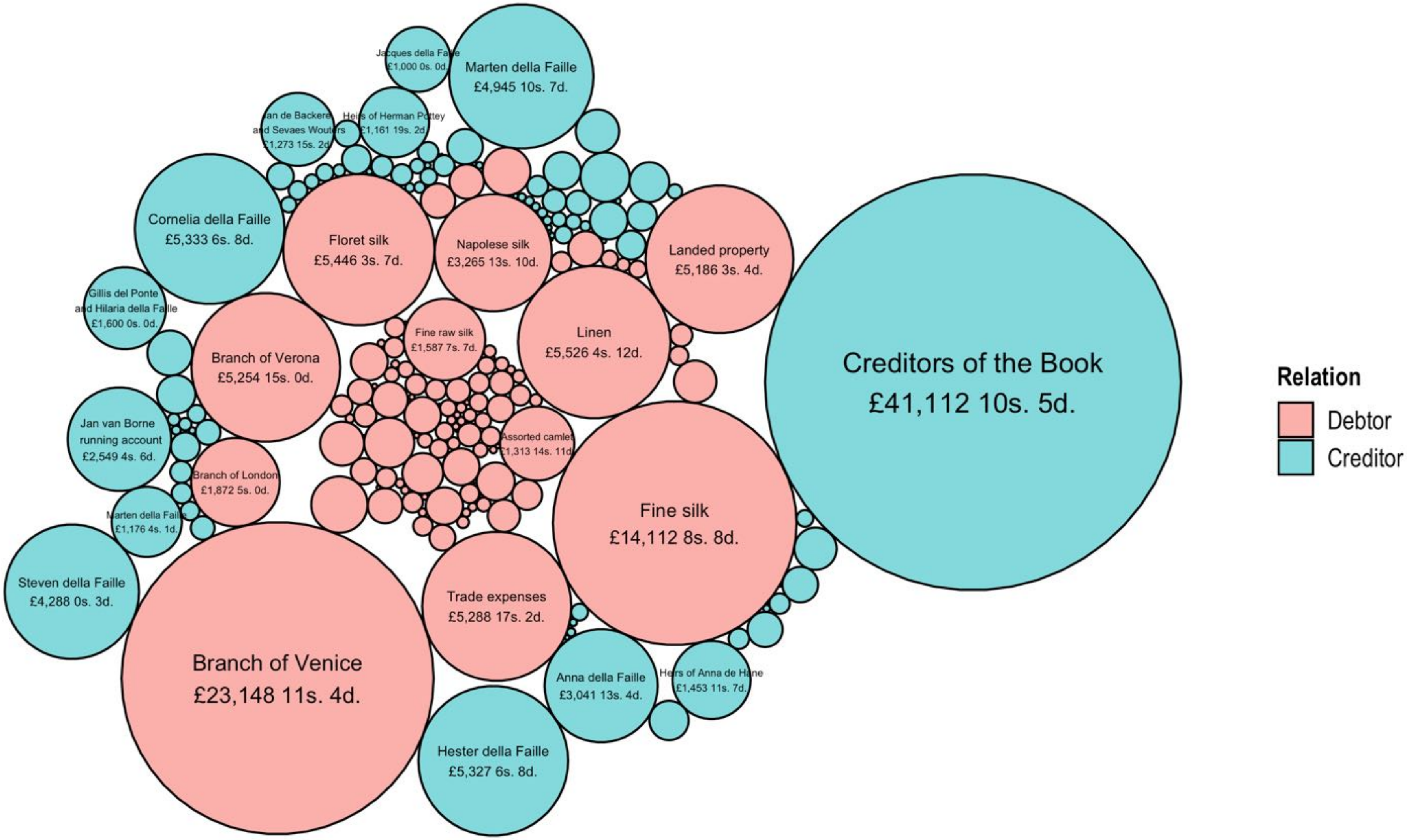
xlm Item Jan van Nieste f° oste
staet fo: 150: debiteur van ——
—— £140: 0: 0:
Dan gemeret die op 26: december

Accounts of the estate of Jan della Faille de Oude

- 8 November 1582 to 31 December 1594
- 2,155 transactions
- 480 accounts

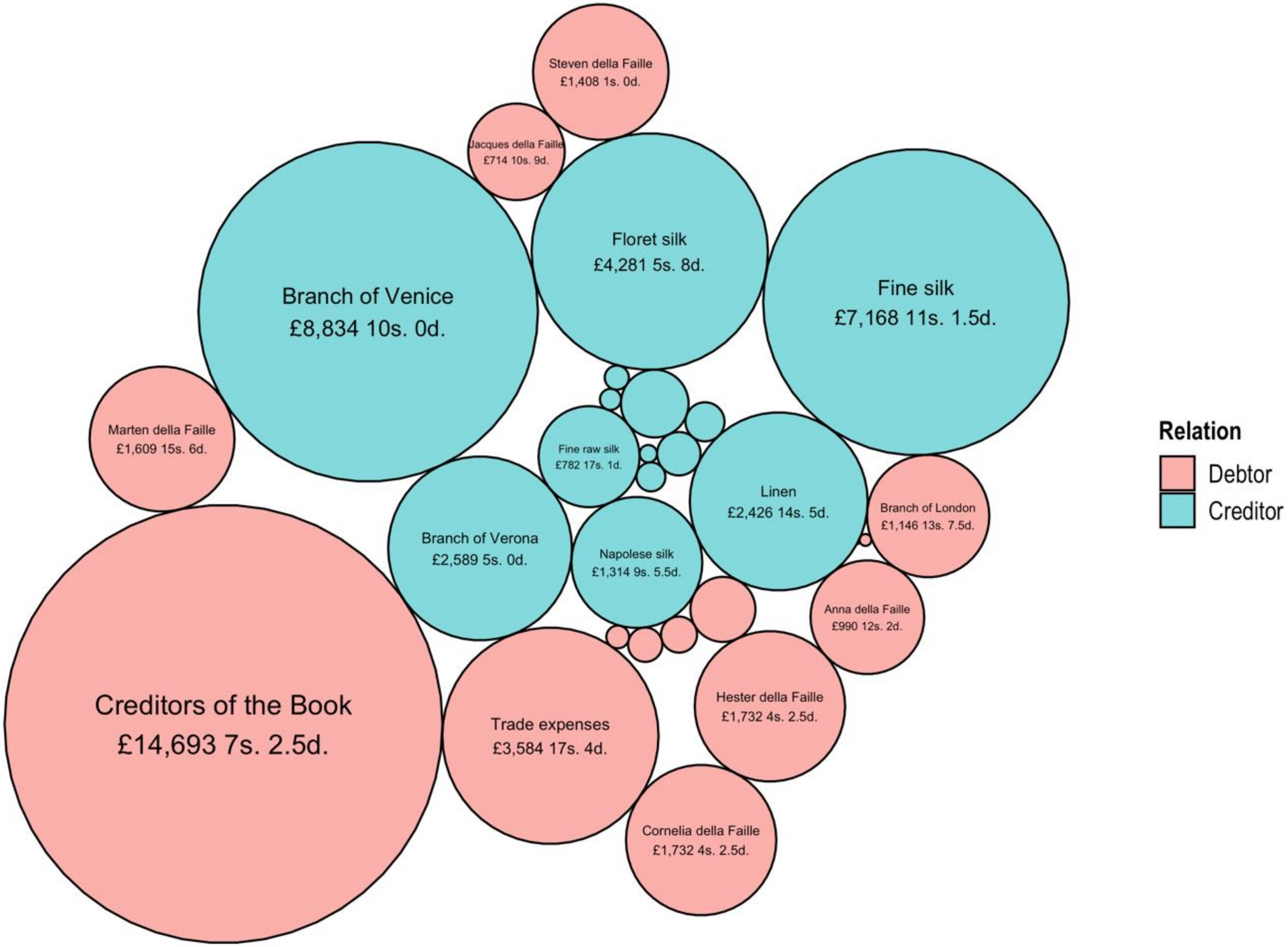
Value of accounts in the estate of Jan della Faille de Oude, 8 December 1582

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



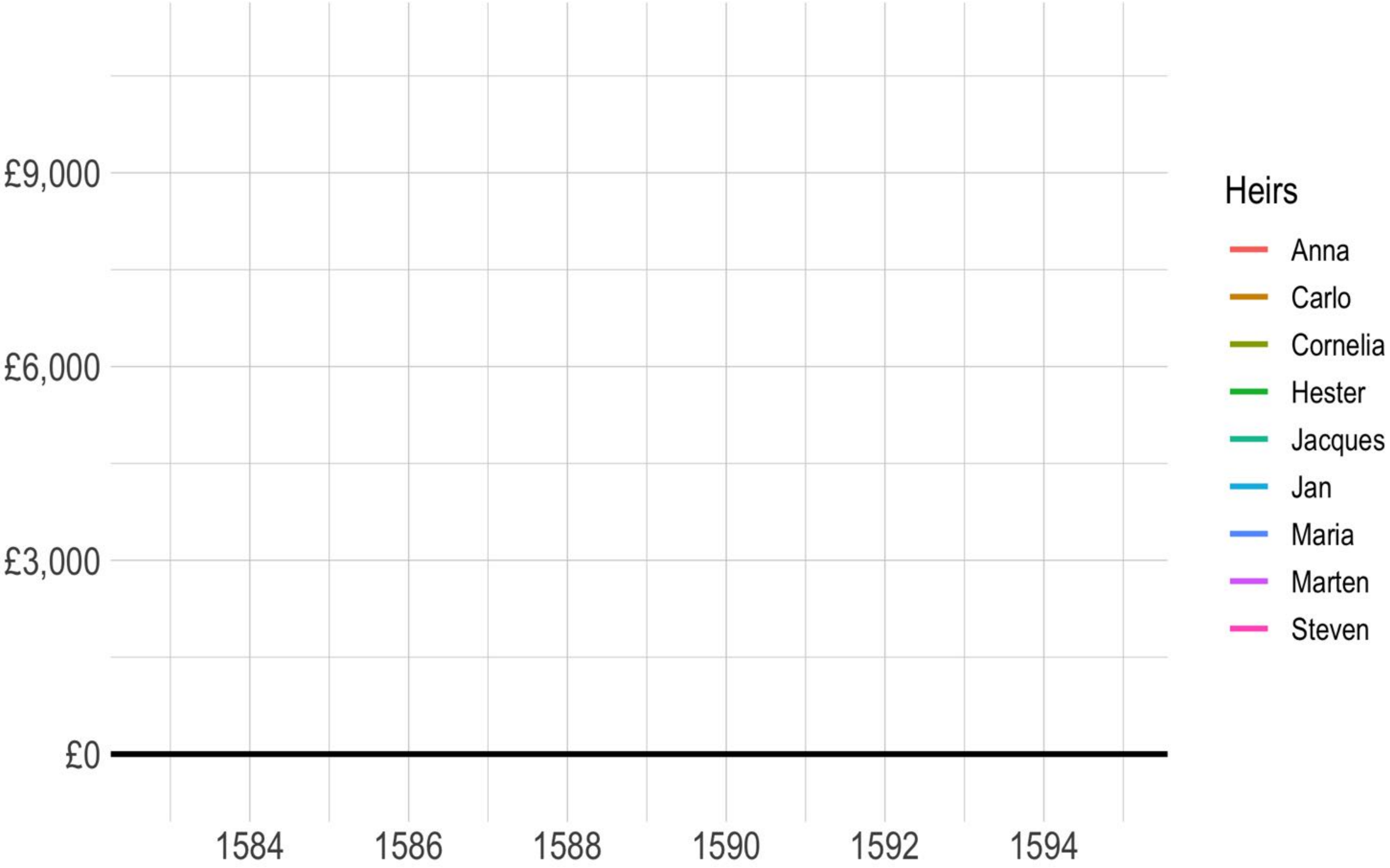
Profits and losses in the trade of Jan de Oude

1 January 1579 to 26 December 1583



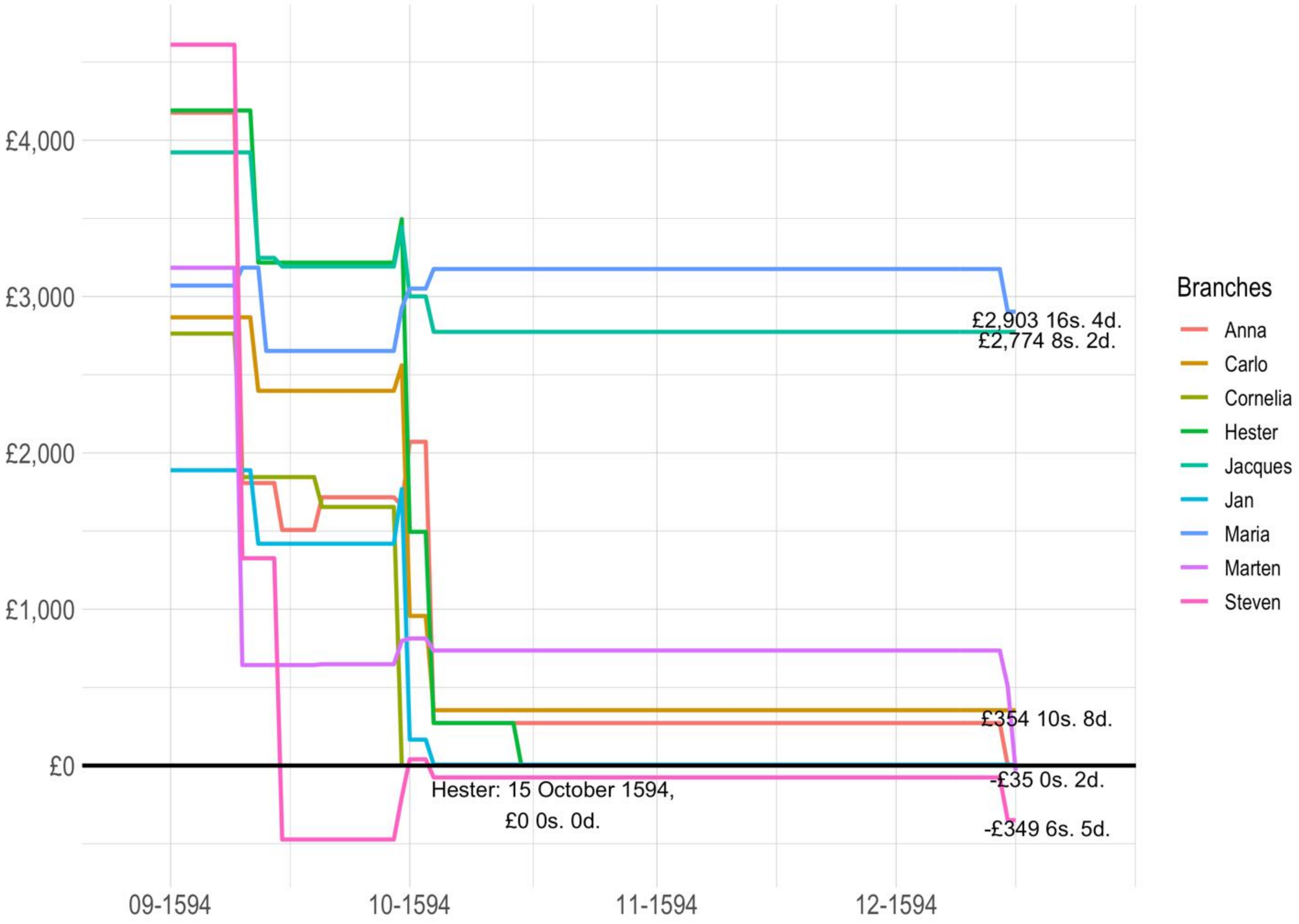
Inheritance due to the heirs of Jan de Oude

Estate of Jan della Faille de Oude, 1582–1594



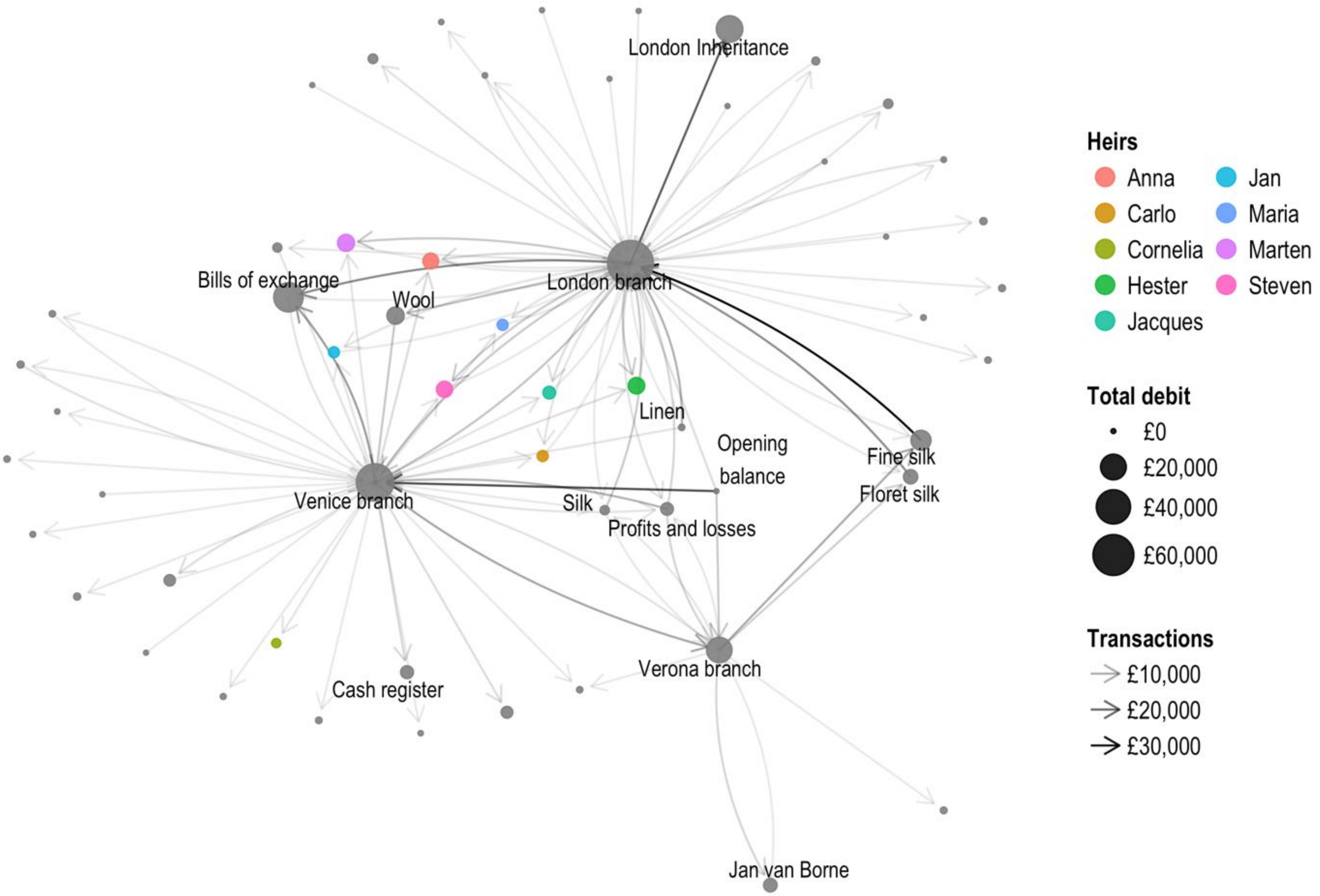
Inheritance due to the heirs of Jan de Oude

September 1594 to 16 December 1594



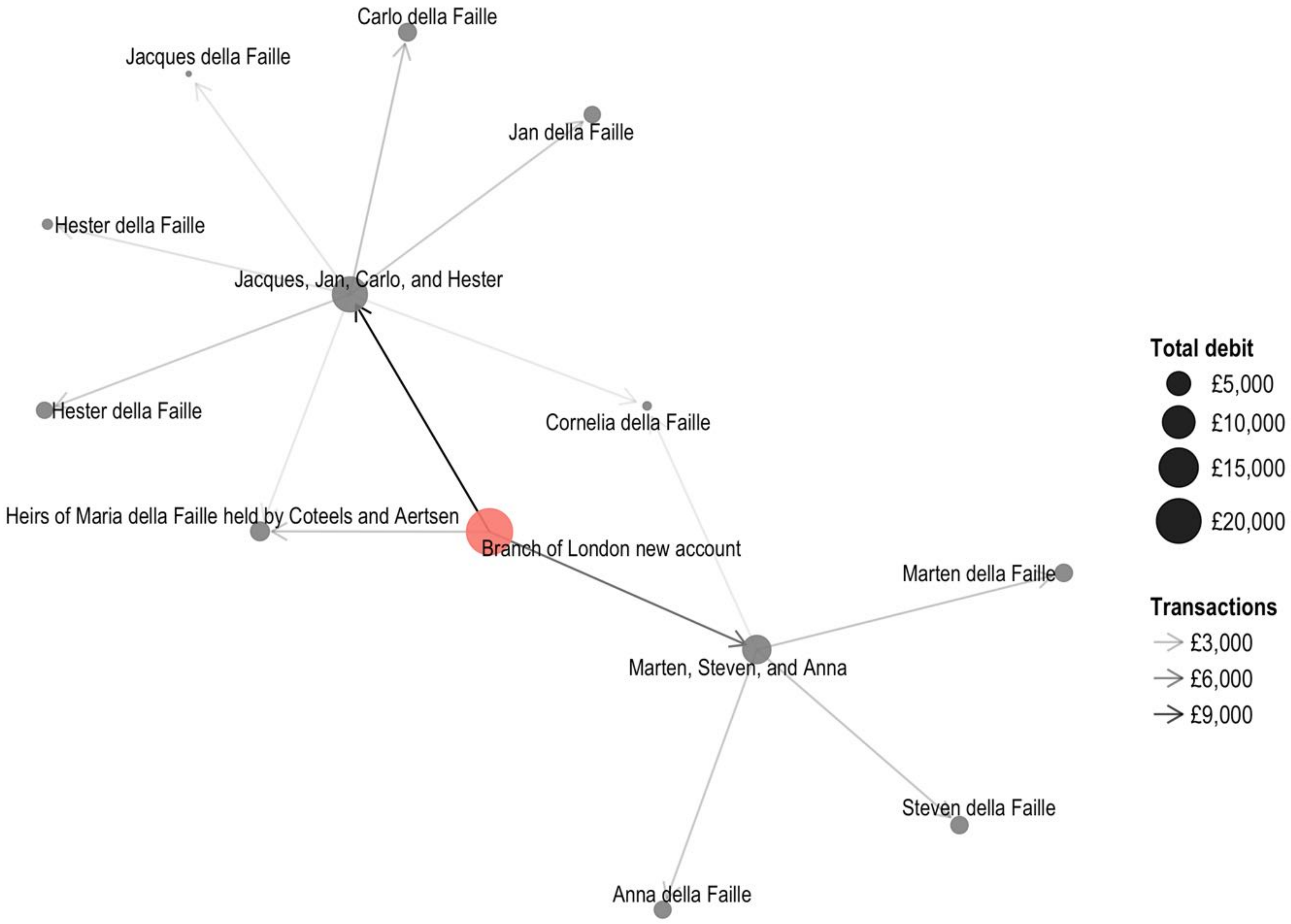
Subgraph of the branches in the trade of Jan de Oude

Estate of Jan della Faille de Oude, 1582–1594

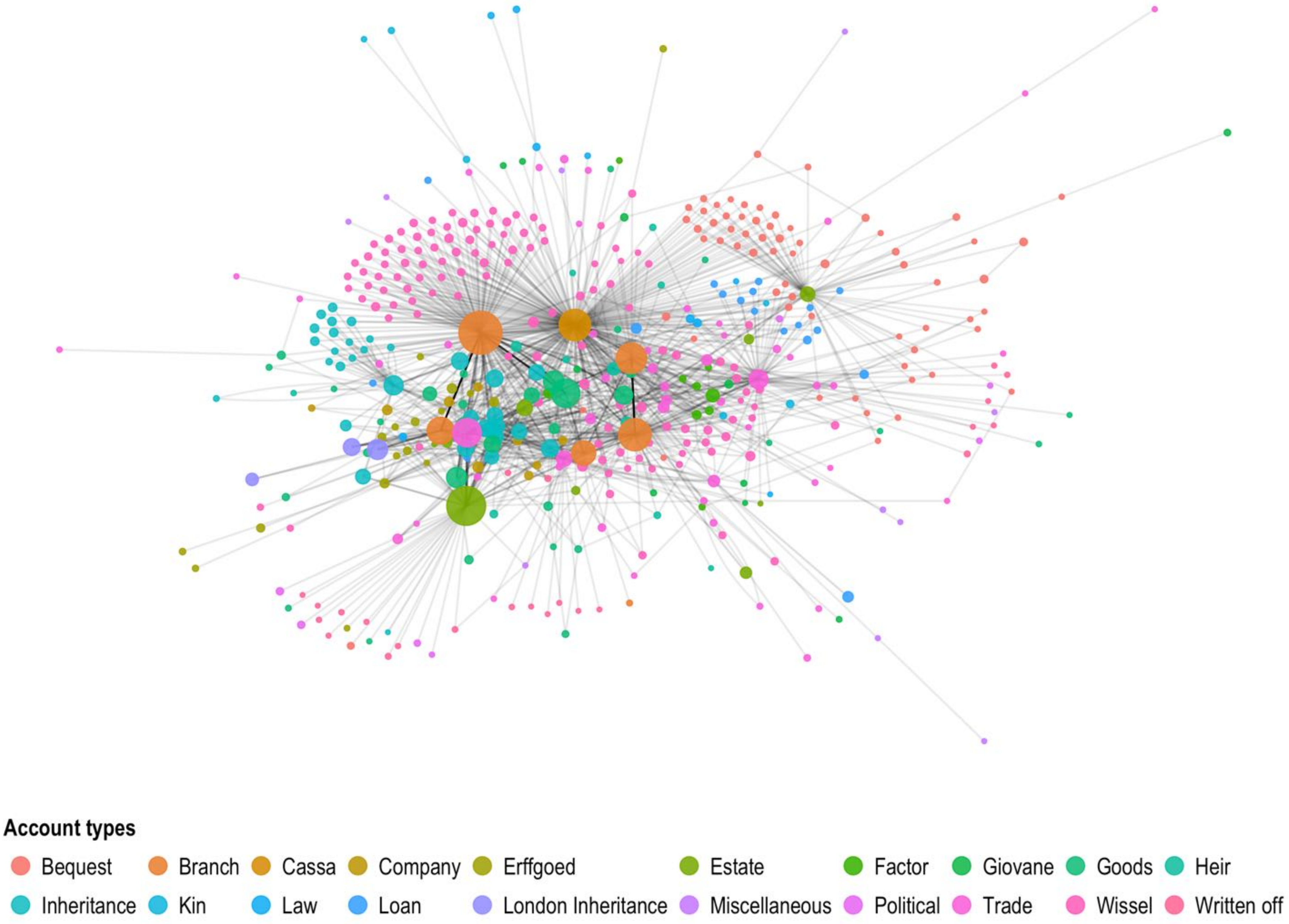


Subgraph of the inheritance from London

Estate of Jan della Faille de Oude, 1582–1594



Estate of Jan della Faille de Oude, 1582–1594

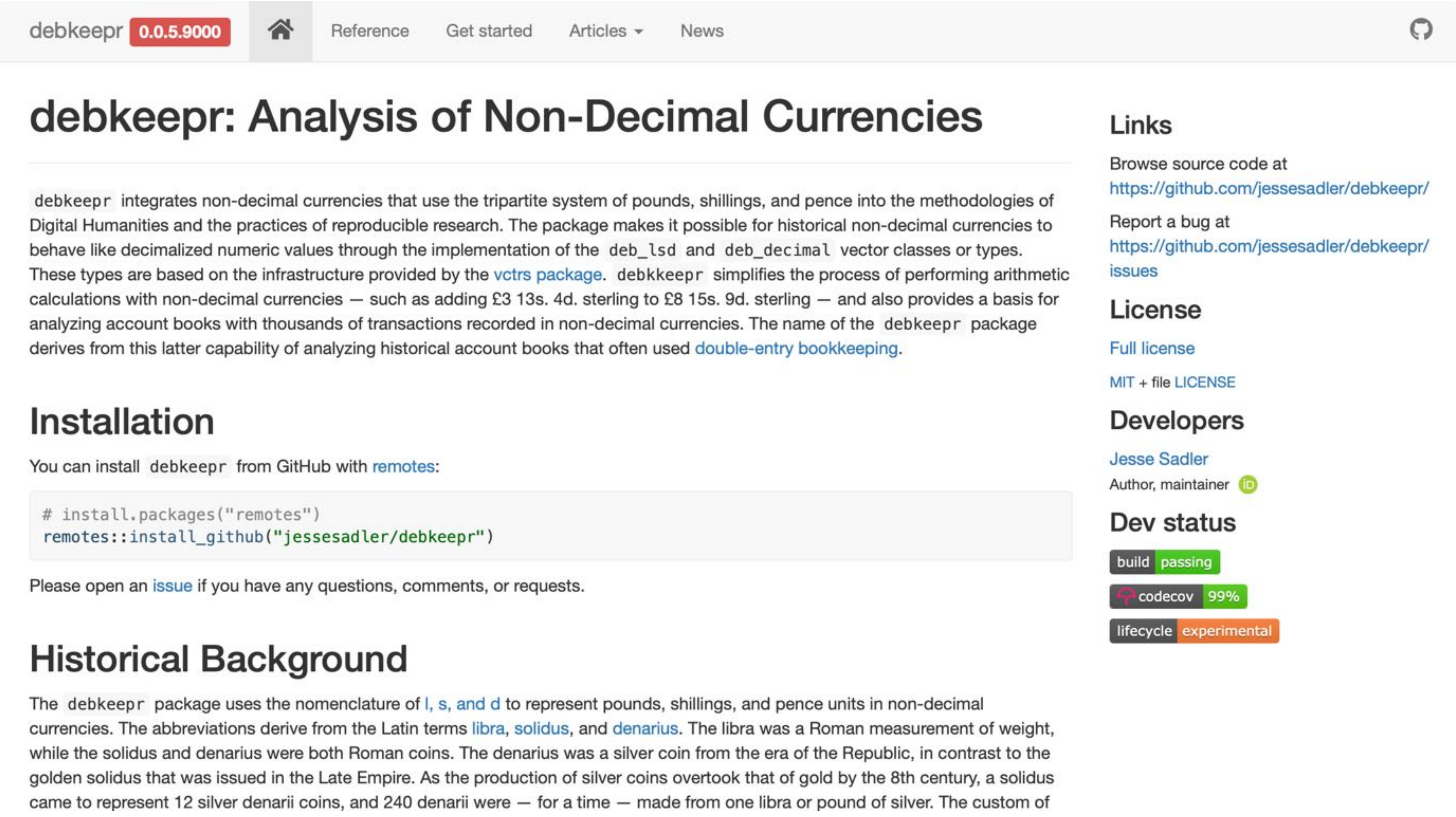


debkeepr: A digital history methodology for non-decimal currencies

- Simplify individual calculations
- Makes data entry of account books worth the effort
- Data analysis
- Visualization
- Reproducible research

debkeepr: Analysis of Non-Decimal Currencies in R

jessesadler.github.io/debkeepr/



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. `debkeepr` 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

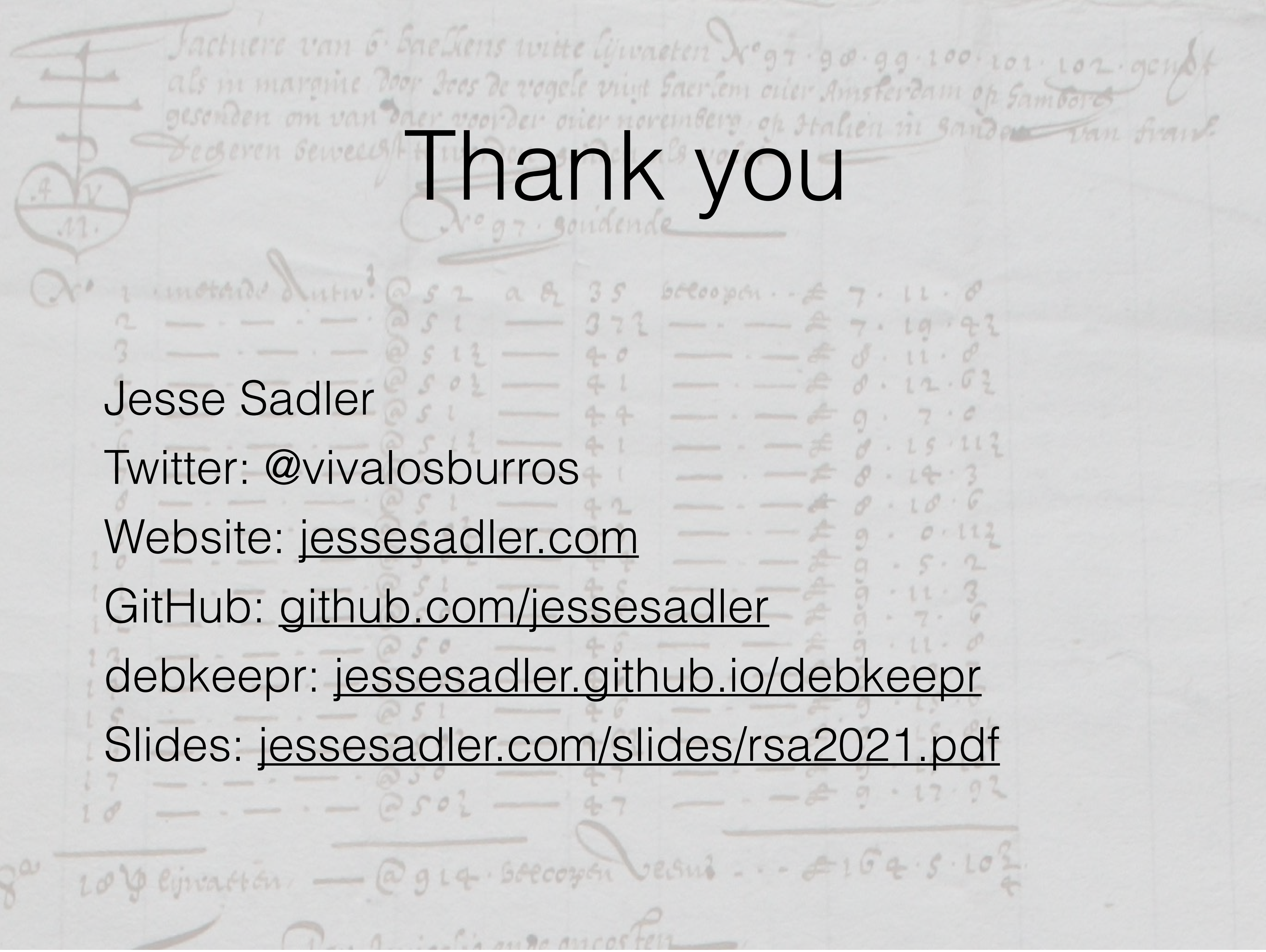
Jesse Sadler
Author, maintainer

Dev status

build passing

codecov 99%

lifecycle experimental



Thank you

Jesse Sadler

Twitter: @vivalosburros

Website: jessesadler.com

GitHub: github.com/jessesadler

debkeepr: jessesadler.github.io/debkeepr

Slides: jessesadler.com/slides/rsa2021.pdf