

Data Bricolage

Mixed methods to verify, summarize,
clean, and enhance data in and out
of the ILS

Kristina Spurgin

E-Resources Cataloger - UNC-Chapel Hill

kspurgin@email.unc.edu

Data Bricolage

Mixed methods to verify, summarize,
clean, and enhance data in and out
of the ILS

Kristina Spurgin

E-Resources Cataloger - UNC-Chapel Hill

kspurgin@email.unc.edu

<http://dx.doi.org/10.3920/978-90-8686-723-3>



Wageningen Academic
Publishers

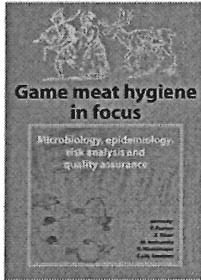
Life Science is our growing business

search

Login | 0 items in shopping cart

[Home](#)
[Bookshop](#)
[Animal Sciences](#)
[Food Sciences](#)
[Social Sciences](#)
[Plant Sciences](#)
[Environmental Sciences](#)
[Dutch language publications](#)
[Book series](#)
[Journals](#)
[Open Access](#)
[OASES](#)
[EAAP](#)
[For authors](#)
[Distributors and representatives](#)
[News](#)
[Contact](#)

[Bookshop > Game meat hygiene in focus \(e-book\)](#)
Game meat hygiene in focus (e-book)
Microbiology, epidemiology, risk analysis and quality assurance
edited by: P. Paulsen, A. Bauer, M. Vodnansky, R. Winkelmayr and F.J.M. Smuklers
2011, 352 pages, E-book (pdf-file)
ISBN: 978-90-8686-723-3
DOI: 10.3920/978-90-8686-723-3
Price (€): 78.00 (excluding VAT)
Add to cart Also available as [paper version](#)
Game meat is consumed world-wide. In most regions, it contributes only a small part to the overall meat and food supply, but for reasons of animal welfare and sustainability it is sometimes considered an alternative to meat from farmed animals. Despite differences in game species, ante mortem conditions (free-range or fenced; wild or semi-domesticated), hunting or harvesting procedures and further handling of the carcass, there are common requirements as regards meat safety and quality. Whereas meat hygiene and safety have been an issue in game meat for export/import for a long time, primary production, domestic supply and direct supply to the consumer have recently been addressed by legislation and these sectors still present unresolved questions and challenges.
This book combines 24 contributions presenting the view of experts in game meat hygiene and quality. They address four main topics: i.e. 'hygiene and microbiology', 'epidemiology', 'risk assessment and management' and 'muscle biology and meat quality'. In addition to contributions on this topic by authors from eight European countries, a South African perspective is provided, thus representing the standpoint of a major game meat exporter.
This volume is the first in a series on safety and quality assurance along the game meat chain, following a 'from forest to fork' approach and is targeted to scientists in academia and industry, graduate students as well as to governmental officials in veterinary public health and food safety.
[Download table of contents of the book 'Game meat hygiene in focus \(e-book\)'. \(PDF file\)](#)
[Meat hygiene related books can also be found in the Food safety assurance Series](#)
Add to cart Also available as [paper version](#)



**Game meat hygiene
in focus**
Microbiology, epidemiology,
risk analysis and
quality assurance



[How to order?](#)
[General terms and conditions](#)
[Privacy policy](#)

<http://dx.doi.org/10.1007/978-3-642-23777-5>

The screenshot displays the SpringerLink website interface. At the top, the browser address bar shows the URL <http://dx.doi.org/10.1007/978-3-642-23777-5>. The SpringerLink logo is prominently displayed. Below the logo, there is a search bar with the text "SEARCH FOR" and a "GO" button. To the right of the search bar, there are links for "AUTHOR OR EDITOR", "PUBLICATION", "VOLUME", "ISSUE", and "PAGE". A "Carolina Population Center" link is also visible. The main content area features the book title "ADVANCES IN INTELLIGENT AND SOFT COMPUTING" and "Volume 164, 2011, DOI: 10.1007/978-3-642-23777-5". Below the title, the authors "David Jin and Sally Lin" are listed. A "Link Out to this Book at University of North Carolina-Chapel Hill" section includes buttons for "Find at UNC" and "Find at UNC". The bottom of the page shows a table of contents with various chapters and their page numbers.

URL Checker Functions

Automatic Report [Fri 02-25-11] 84848 Errors

1. Results 2. Previous 3. Summary

56994: Moved Permanently(301)
 22: Forbidden(403)
 3: Service Unavailable(503)
 428: Malformed URL(2)
 49: No URL(2.3)

URL

#	Record #	Go	Error	Go	URL
1	036258299	http://www.aps...	504		
2	047694434	http://www.m...	503		
3	047615035	http://www.c...	503		
4	053489904	http://www.c...	503		
5	064476339	http://www...	500		
6	063522918	http://www...	500		
7	063532797	http://www...	500		
8	063532748	http://www...	500		
9	063529657	http://www...	500		
10	063529698	http://www...	500		
11	055239125	http://www...	500		
12	052103872	http://www...	500		
13	052103999	http://www...	500		
14	051977436	http://www...	500		
15	051977060	http://www...	500		
16	051977055	http://www...	500		
17	049905150	http://www...	500		
18	048516296	http://www...	500		
19	047697746	http://www...	500		
20	047655299	http://www...	500		
21	043686380	http://www...	500		
22	037963983	http://www...	500		
23	037963992	http://www...	500		
24	037963931	http://www...	500		
25	037963272	http://www...	500		
26	037962814	http://www...	500		
27	037962407	http://www...	500		
28	037962536	http://www...	500		
29	037962364	http://www...	500		
30	037962350	http://www...	500		
31	037962176	http://www...	500		
32	037961986	http://www...	500		
33	037961937	http://www...	500		
34	037961901	http://www...	500		
35	037961895	http://www...	500		

URL CHECKER

ACCESS CHECKER

0 / 84848

Access checker:

Script use: input

- Data sources:
 - Extract from MARC file pre-load using MARCedit
 - Export from Millennium Create Lists post-load
- URL must be final column – One URL per row
- Any number of columns can be included before the URL

1	bnum	title	date	url				
2	Spr978364	Advances	2013	http://dx.doi.org/10.1007/978-3-642-37798-3				
3	Spr978364	Advances	2013	http://dx.doi.org/10.1007/978-3-642-37807-2				
4	Spr978364	Health Inf	2013	http://dx.doi.org/10.1007/978-3-642-7899-7				
5	Spr978146	Global Ov	2013	http://dx.doi.org/10.1007/978-1-4614-6265-1				
6	Spr978146	Reviews c	2013	http://dx.doi.org/10.1007/978-1-4614-6470-9				
7	Spr978146	Phosphori	2013	http://dx.doi.org/10.1007/978-1-4614-6803-5				
8	Spr978159	Transport	2013	http://dx.doi.org/10.5822/978-1-59726-242-2				

Access checker:

Script use: running the script

In Windows Powershell:

```
jruby -S .\access_checker.rb .\data\URLS_to_check.csv .\data\output.csv
```

Access checker:

Script use: running the script

In Windows Powershell:

```
=====
What platform/package are you access checking?
Type one of the following:
  asp : Alexander Street Press links
  ebr  : Ebrary links
  ss   : SerialsSolutions links
  srmo : Sage Research Methods Online links
  spr  : SpringerLink links
  upso : University Press <inc. Oxford> Scholarship Online links
=====
Package? spr
```

Access checker:

Script use: running the script

In Windows Powershell:

```
Preparing to check access...
1 of 8, access = full
2 of 8, access = full
3 of 8, access = DOI error
4 of 8, access = full
5 of 8, access = full
6 of 8, access = full
7 of 8, access = full
8 of 8, access = full
PS C:\ruby_scripts\Ebook-Access-Checker>
```

Access checker:

Script use: output

	A	B	C	D	E
1	bnum	title	date	url	access
2	Spr97836423	Advances	2013	http://dx.	full
3	Spr97836423	Advances	2013	http://dx.	full
4	Spr97836423	Health Inf	2013	http://dx.	DOI error
5	Spr97814614	Global Ov	2013	http://dx.	full
6	Spr97814614	Reviews c	2013	http://dx.	full
7	Spr97814614	Phosphor	2013	http://dx.	full
8	Spr97815972	Transport	2013	http://dx.	full
9	Spr97815972	Creating C	2013	http://dx.	full

Access checker:

Other info

- Looks at the “landing page” for each URL – does not download or harvest any full text content
- Written in JRuby
- Open source – [Code available from GitHub](#)
- Instructions for use also at GitHub – I tried to write them for people not familiar with using scripts

A script to summarize PAID data from order records

DEALING WITH PAYMENT DATA

Payment data processor:

The problem addressed

- Millennium will export payment data from Create Lists of order records
- BUT the format of the exported data makes it virtually unusable.
 - 9 payment field columns, repeated
- One row in the output below had data all the way to column ST!

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
RECORD #	Paid Date	Invoice Date	Invoice N°	Amount P	Voucher #	Copies	Sub From	Sub To	Note							
o1021814	12/10/1996	7/29/1996	96-117665	1881.5	78702	1	-	-	1997\	9/4/1997	7/29/1996	96-117665	-1219	87880	1	-
o1011642	6/19/1992	4/10/1992	mm248-41	2342.6	14482	1	-	-	\	6/6/1993	4/22/1993	MM248-41	2342.6	29793	1	-
o1022148	4/1/1997	3/25/1997	REC/00271	21200	82513	1	-	-	07/01/199	6/6/1997	4/16/1997	X133282	-5300	85255	1	-
o1010321	3/19/1992	1/17/1992	11792	832.1	10238	1	-	-	0392\	3/11/1993	1/22/1993	12293	858.6	25867	1	-
o1023386	1/20/1998	1/9/1998	10998	65224.36	91639	1	-	-	\							
o1025731	6/13/2000	5/25/2000	52500	35700	118203	1	-	-	6/1/00-5/	6/11/2001	6/9/2001	x331474	44275	123796	1	-

This seems to be another of those perennial questions on the IUG listserv, and there have been other good solutions presented.

This works well for us at UNC because the point-and-click Windows .exe is extremely simple to use.

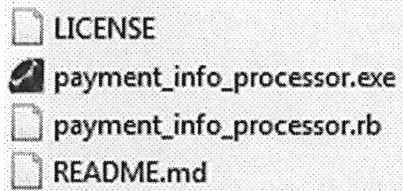
Payment data processor:

The solution

- Script outputs either:
 - One payment per line
 - Payments summarized by fiscal year

Payment data processor:

Script use: running the script



- You can run the Ruby (.rb) script from the command line
- BUT
- Everyone using this at UNC just double-clicks on the .exe

Payment data processor:

Script use: running the script

```
=====
WELCOME TO THE MILLENNIUM PAYMENT DATA PROCESSOR
version 1.2.0, 2013-04-16
written by Kristina Spurgin, ESM, kspurgin@email.unc.edu
=====

INPUT:
This program processes payment data exported from a List of
order records in Millennium.

Exported data must include:
- order - record number (first field exported)
- order - paid (last field exported)

Exported data may include:
- other fields, exported between record number and paid

OUTPUT:
There are two options for output. Both produce tab-delimited .txt
files you can open with Excel. Output options are:
- individual payments - outputs one line per payment made. Each
line contains order record number, fiscal year of payment,
(other fields exported), payment data fields
- payments summarized by fiscal year - outputs one line per order
record. Each line contains order record number, (other fields
exported), and one column for each fiscal year input by script
user. Each of these columns contains total payment amount for
that fiscal year.
Open the .txt file with Excel. Choose "65001 : Unicode (UTF-8)"
as file origin (encoding) to properly display diacritics.

Do you want to see instructions on how to export from Millennium
in order to use this application? <y/n>
```

Payment data processor:

Script use: running the script

```
=====
HOW TO EXPORT DATA FOR THIS SCRIPT FROM MILLENNIUM
=====
```

REQUIRED FIELDS:

- ORDER - RECORD NUMBER: required, must be first column
- ORDER - PAID: required, must be the last column

Note: you can export whatever fields you want in between.

SET FIELD DELIMITER:

- Click on "Field delimiter".
- In popup dialog, click "Control character (1-127)."
- Enter "42" (no quotes).
- Click ok.

SET TEXT QUALIFIER:

- Click on "Text qualifier".
- In popup dialog, select "None."
- Click ok.

EXPORT WITH REQUIRED NAME AND LOCATION SO SCRIPT CAN FIND DATA:

- The exported file MUST be named payment_data.txt
- Save the file in the "data" folder inside your "rubyscripts" folder.

Continue? <y/n>

Payment data processor:

Script use: running the script

```
What output would you like?
1. individual_payments
2. summary_of_payments_per_fiscal_year
? 1

=====
Done?
=====
Results are in your "rubyscripts\output" folder.
Look for a file called "payments.txt"
Press return/enter to exit.
```

Payment data processor:

Script use: output

RECORD #(ORDER)	FY	TITLE	Paid Date	Invoice Date	Invoice N	Amount P	Voucher N	Copies	Sub From	Sub To	Note
o10002637	1992	Africana bul	02/02/1993	11/20/1992	120/5884/	45.05	23418	1	- -	- -	T. 37\
o10002637	1994	Africana bul	05/08/1995	03/10/1995	95/K000/0	31.59	58990	1	- -	- -	T. 40\
o10002637	1995	Africana bul	07/07/1995	03/27/1995	95/K000/0	15.33	61081	1	- -	- -	T. 42\
o10002637	1995	Africana bul	05/13/1996	04/12/1996	PL/000445	13.74	71716	1	- -	- -	T. 43\
o10002637	1997	Africana bul	09/12/1997	08/18/1997	97/K000/1	13.98	88216	1	- -	- -	T. 44\
o10002637	1997	Africana bul	06/09/1998	05/18/1998	98/K000/1	12.28	96685	1	- -	- -	T. 45\
o10002637	1998	Africana bul	06/17/1999	05/17/1999	99/K000/1	26.71	107910	1	- -	- -	T. 46\
o10002637	1999	Africana bul	06/08/2000	05/15/2000	00?K000/0	28.41	117935	1	- -	- -	T. 47\
o10002637	2001	Africana bul	09/26/2001	08/31/2001	01/K000/1	39.11	132624	1	- -	- -	T. 48\
o10002637	2002	Africana bul	10/03/2002	07/22/2002	02/K000/0	31.75	144138	1	- -	- -	T. 49\
o10002637	2002	Africana bul	05/12/2003	02/25/2003	03/KS00/0	32.05	150890	1	- -	- -	T. 50\
o10002637	2003	Africana bul	05/25/2004	12/31/2003	04/KS00/0	25.26	161498	1	- -	- -	T. 51\
o10002637	2004	Africana bul	04/05/2005	02/24/2005	05/KS00/0	28.49	169976	1	- -	- -	T. 52\
o10002637	2005	Africana bul	05/17/2006	03/23/2006	FV-KES/06	28.64	181539	1	- -	- -	T. 53\
o10002637	2006	Africana bul	04/12/2007	03/22/2007	07/KS00/0	29	190223	1	- -	- -	T. 54\
o10002637	2008	Africana bul	07/29/2008	06/27/2008	08/KS00/0	45	201958	1	- -	- -	T. 55\
o10002637	2009	Africana bul	08/11/2009	07/28/2009	09/KS00/0	37.6	210812	1	- -	- -	T. 56\
o10002637	2010	Africana bul	08/19/2010	07/22/2010	10/KS00/0	37.6	220277	1	- -	- -	T. 57\
o10002637	2010	Africana bul	05/12/2011	04/28/2011	11/KS00/0	37.6	225926	1	- -	- -	T. 58\
o10003563	1995	Children's b	05/24/1996	05/15/1996	187937	118.3	72250	1	- -	- -	V. 21\
o10003563	1996	Children's b	02/10/1997	01/22/1997	301278	120.27	80648	1	- -	- -	V. 22\
o10003563	1997	Children's b	01/13/1998	12/31/1997	488433	126.18	91415	1	- -	- -	V. 23\

Payment data processor:

Script use: running the script

What output would you like?

1. individual_payments
 2. summary_of_payments_per_fiscal_year
- ? 2

Enter the years for which you want summary data.

If you want FY2009-10, FY2010-11, and FY2011-12, type:

2009,2010,2011

Important:

- the years are separated by commas
- there are NO spaces in between dates and commas
- the years are four digits
- entering 2009 gets you FY2009-10
- entering 2010 gets you FY2010-11, and so on.

You can request as many years as you like.

If there is no data for a year, its values will be zero.

Type years below and hit enter/return:

2010,2011,2012

Payment data processor:**Script use: output**

RECORD #	OR TITLE	FY2010-20	FY2011-20	FY2012-2013
o10002637	Africana b	75.2	0	0
o10003563	Children's	231.14	282	295
o10009553	Anthropol	104	152	0
o1001231x	Archivum	259.36	125.3	123.95
o10012692	CHUNG YA	0	0	0
o10016181	Bukkyō sh	181	96	105.5
o10018827	880-01 Jin	0	212.5	0
o10021826	New ramk	79.62	38.74	0
o1002265x	880-01 Ch	125.5	71.5	69.5
o10022673	880-01 Ch	96	102	113.5

Payment data processor:

Other info

- Written in Ruby
- Open source – [Code available from GitHub](#)
- Instructions for use also at GitHub – I tried to write them for people not familiar with using scripts

Questions?

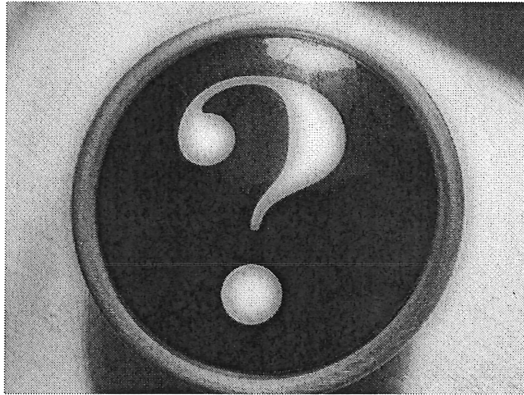


Photo by [theunquietlibrarian](#) on Flickr