

# BLOOMIMPORT

Stata package to import Bloomberg exported data

---

*Ultimo aggiornamento: 3 giugno 2024*

Nicola Tommasi  
C.I.D.E.  
[nicola.tommasi@univr.it](mailto:nicola.tommasi@univr.it)

# 1. Il comando bloomimport

bloomimport è un nuovo comando Stata che permette

- di importare in Stata i dati esportati tramite una particolare procedura che usa le API Bloomberg di Excel
- di trasformarli in serie storica
- di convertire in numeriche le variabili create

I dati devono essere stati esportati tramite la procedura "Creare spreadsheet" -> "Tabella dati storici" e devono rispettare la disposizione-layout mostrata in figura 1 e d'ora in avanti definita "formato wide"

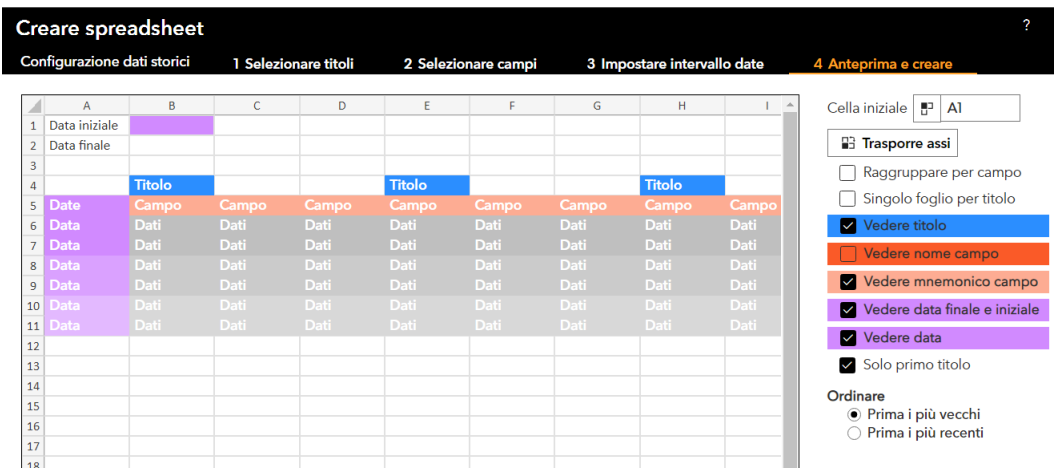


Figura 1 – Layout wide dei dati

oppure rispettare la disposizione-layout mostrata in figura 2, e d'ora in avanti definita "formato long", che si ottiene cliccando sul 'Transpose Axes', in alto a sinistra<sup>1</sup>.

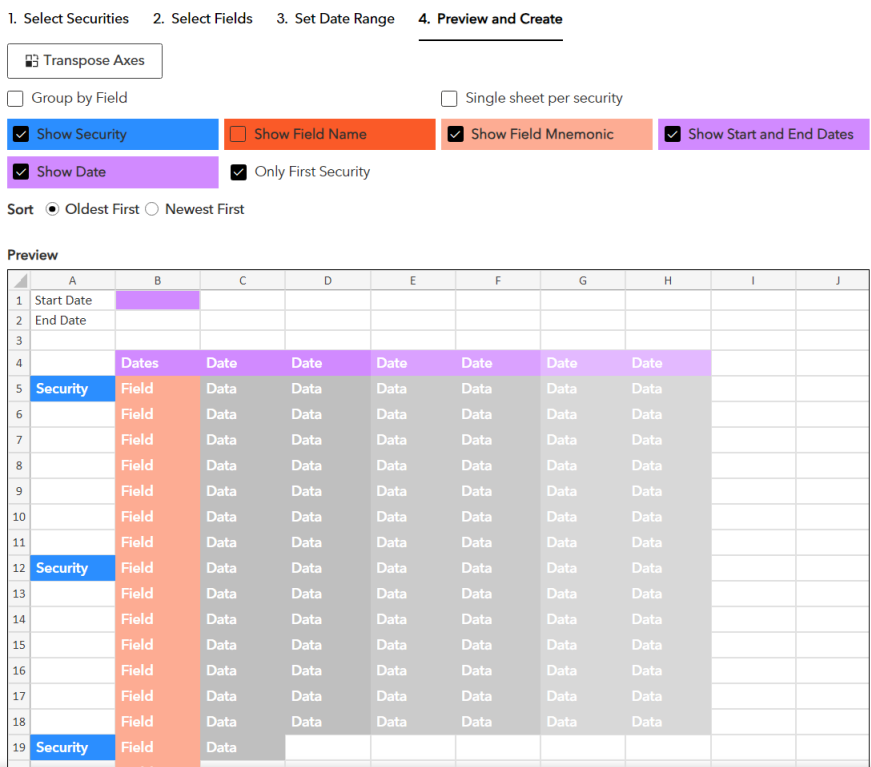


Figura 2 – Layout long dei dati

1. La figura figura 1 fa riferimento alla precedente versione di esportazione di Bloomberg, la figura 2 a quella più recente.

Notare che per entrambi i layout non deve essere selezionato "Vedere nome campo" (Show Field Name). Se i dati esportati presentano questa riga bisogna selezionarla, eliminarla e risalvare il file. Per il formato wide, a partire dalla cella di partenza (A4) ci deve essere la riga con i tickers e subito dopo la riga con "Dates" e i campi esportati come si vede in figura 3

	A	B	C	D	E	F
1	Data iniziale	CY 2007				
2	Data finale	CY 2020				
3						
4		FR0000120404 ISIN				
5	Dates	ENTERPRISE_VALUE	CF_FREE_CASH_FLOW_FIRM	FREE_CASH_FLOW_EQUITY	CF_FREE_CASH_FLOW	WACC
6	31/12/2007	11997.6209	-180.5602	1198	-249	11.2288
7	31/12/2008	9100.4967	-453.3492	955	-524	9.857
8	31/12/2009	10400.7761	-453.3492	17	-325	9.6987
9	31/12/2010	8525.2385	-453.3492	-1490	44	11.8169
10	30/12/2011	4868.7196	140.0526	-250	83	10.1933
11	31/12/2012	6613.1855	-818.6639	8	-849	8.9421

Figura 3 – Esempio di dati esportati in formato wide e conformi al comando

Per il formato long, a partire dalla cella di partenza (A4) ci deve essere la riga con il ticker nella prima colonna e il nome del campo nella seconda colonna come si vede in figura 4. A seguire le colonne intestate con il riferimento temporale dei dati.

	A	B	C	D	E	F
1	Start Date	CY 2005				
2	End Date	CY 2023				
3						
4		Dates	30/12/2005	29/12/2006	31/12/2007	31/12/2008
5	FBMS US Equity	BS_NON_PERFORM_ASSET	0.5937	2.3461	4.188	4.9694
6		TANGIBLE_COMMON_EQUITY	18.4776	30.9064	34.9724	35.3291
7		BS_RSRV_LOAN_LOSS	2.3668	3.7929	4.2212	4.7849
8		EPS_GROWTH	51.4019	66.6667	-5.1852	-51.5625
9		GROWTH_IN_TOT_LOAN	22.1255	44.1342	30.7444	-12.9675
10		GROWTH_IN_TOT_DPST	54.2751	45.3701	9.7937	-2.0948
11		EFF_RATIO	68.4435	67.0179	68.686	77.1366
12		RETURN_ON_ASSET	0.7535	0.931	0.8366	0.3809
13		RETURN_COM_EQY	10.8424	13.0404	11.1375	5.0759
14		NET_INC_GROWTH	53.6274	73.6342	15.3139	-51.6342
15		NON_PERF_ASSET_TO_TOT_A	0.2017	0.5616	0.8443	1.0466
16		NON_PERFORM_ASSET_TO_TOT_A	0.3014	0.8263	1.1282	1.5381
17		TOT_LOAN_TO_TOT_DPST	81.4181	80.7258	96.1298	85.4543

Figura 4 – Esempio di dati esportati in formato long e conformi al comando

## 2. Sintassi

Per installare il comando, dalla command bar di Stata digitate:

```
net install bloomimport, from(https://raw.githubusercontent.com/NicolaTommasi8/bloomimport/main/)
```

La sintassi del comando è la seguente:

```
bloomimport using filename, sheet("sheetname") cellrange([start]:[end])
[export(string)] nvar(#) lasttick(string) datastart(string) from(string) to(string)]
```

dove:

**filename** percorso e nome del file Excel dei dati esportati da Bloomberg

**sheet(sheetname)** foglio in cui sono presenti i dati

**cellrange([start][:end])** range di celle da importare dal foglio excel. Il range viene specificato usando la notazione di Excel, per esempio `cellrange(A1)` o `cellrange(A1:BC2000)`. Non serve specificare l'intero range, basta specificare la cella di partenza da cui leggere i dati; nell'esempio riportato in Figura 3 sarebbe `cellrange(A4)`.

**export(string)** specifica la disposizione con cui sono stati esportati i dati. La disposizione può essere `export(wide)` (opzione di default), oppure `export(long)`

**nvar(#)** specifica il numero di campi esportati per ciascun titolo (opzione obbligatoria se `export(wide)`)

**lasttick(string)** specifica la colonna dell'ultimo ticker dei dati da importare, da indicare secondo le regole di Excel (opzione obbligatoria se `export(wide)`).

**datastart(string)** specifica la colonna da cui partono i dati. Di default è la colonna successiva a quella specificata in `cellrange()`. Se non fosse quella va specificata qui.

**from(varlist)** lista dei campi che devono essere rinominati, tipicamente perché il nome è troppo lungo o perché il nome sarebbe incompatibile con le regole di Stata sui nomi delle variabili (è ancora da vedere bene se farlo e come farlo, per ora non usare questa opzione!)

**to(varlist)** lista dei nuovi nomi (nello stesso ordine) da assegnare ai campi specificati in `from(varlist)` (per ora non usare questa opzione, vedi `from(varlist)`)

Tenete presente che `bloomimport` cancella gli eventuali dati in memoria di Stata e li sostituisce con quelli caricati dal comando.

## 2.1. Esempi

Nel primo esempio si importano i dati esportati in formato wide dal foglio "Foglio1" dove ci sono 32 campi per ciascun ticker e l'ultimo di questi ticker è nella colonna DEP del foglio medesimo.

```
. bloomimport using "data/Vantaggio competitivo e WACC.xlsx", cellrange(A4) ///
> sheet("Foglio1") datastart(B) nvar(32) lasttick(DEP)

. summ
```

Variable	Obs	Mean	Std. dev.	Min	Max
-----+-----					
date	0				
ENTERPRISE~E	1,242	11074.45	22131.1	5.125	275673.4
CF_FREE_CA~M	1,150	464.8148	1048.991	-3928.55	8677.674
FREE_CASH_~Y	1,220	535.3761	1699.12	-5986	21368
CF_FREE_CA~2	1,252	322.9462	957.2912	-5003	8431
-----+-----					
WACC	1,250	7.00434	2.775853	-25.4714	21.747
WACC_COST_~Y	1,250	9.316001	3.114569	-25.6074	23.8439
WACC_WEIGH~Y	1,250	71.21741	22.55256	1.8608	100
WACC_COST_~T	1,250	1.568361	1.5902	-.9692	8.7885
WACC_WEIGH~T	1,250	28.67527	22.45683	0	98.1392
-----+-----					
SHORT_AND_~T	1,252	3320.436	7371.259	0	66354
CASH_AND_M~S	1,254	1399.036	3407.764	.083	36060
FCF~R_GROWTH	712	11.17473	23.38947	-72.5478	226.1581
FREE_CASH_~H	850	9.434474	28.7091	-54.3235	310.6351
FCF~L_GROWTH	1,088	88.41307	1820.618	-19456.96	32075.89
-----+-----					
HISTORICAL~P	93	227.6864	316.9819	14.3072	1669.341
EBIT	1,254	658.7461	1528.055	-16169	13266
IS_INC_TAX~P	1,254	150.5178	341.3357	-1689	2932

CAPITAL_EX~D		1,252	-541.5659	1569.208	-16797	0
CF_CASH_FR~T		1,252	-729.68	2334.02	-34399	14286
-----						
IS_IMPAIRM~S		745	96.29379	481.9777	-103	9158
IS_IMPAIRM~L		834	101.086	469.6049	-11	4786
IS_IMPAIR~S		475	36.19235	100.9268	-50	1126
CF_CHNG_NO~P		1,238	-8.101133	397.3753	-2390	8759
RETURN_COM~Y		1,246	8.061317	28.50765	-527.0218	239.998
-----						
RETURN_ON~T		1,248	3.600192	15.08787	-136.1412	429.4941
RETURN_ON~P		1,221	5.655745	22.31824	-231.2528	139.3538
RETURN_ON~L		1,221	5.172286	25.98726	-734.1741	131.1622
OPER_MARGIN		1,254	43.76568	2343.848	-11153.5	81623.19
PE_RATIO		93	46.10994	106.7487	.1578	1025
-----						
PX_TO_BOOK~O		93	1.471926	1.489012	.2286	6.6377
EV_TO_T12M~A		1,226	53.0463	851.3633	.4938	21008.69
NET_DEBT_T~A		1,236	9.017561	169.2792	-38.8095	4180
ticker		0				

Nel secondo esempio il numero di campi è 13 e l'ultimo ticker è posizionato nella colonna GQO

```
. set maxvar 8000
```

```
. bloomimport using "data/us_banks.xlsx", cellrange(A4) sheet("US1") ///
> datastart(B) nvar(13) lasttick(GQO)
```

```
. summ
```

Variable		Obs	Mean	Std. dev.	Min	Max
-----						
date		0				
NET_INT_INC		26,861	94.85882	565.9889	-449.202	12644
IS_ACT_LOA~T		25,901	12.36292	151.3245	-576	16609
BS_TOT_LOAN		26,745	8090.337	46071.2	0	1033521
BS_LOAN_MTG		26,751	7966.754	45301.14	0	1022258
-----						
BS_TOTAL_D~S		3,903	457.9059	2468.646	-2.345	37205
BS_DEMAND~T		26,336	2634.726	16761.92	0	529051
INTEREST_B~O		26,605	6670.138	40938.49	0	996854
BS_CUSTOME~S		26,736	9263.505	57186.44	0	1482479
TOTAL_EQUITY		26,783	1384.682	8689.521	-.385	208079
-----						
BS_TIER1_C~O		26,077	13.74289	80.34161	0	12960
BS_NON_PER~T		26,140	104.7876	875.102	0	34573
BS_TOT_ASSET		26,751	12827.44	82475.08	0	1981349
BS_CASH_NE~M		26,738	283.0214	1641.788	0	63904
ticker		0				

Nel terzo esempio, si importano i dati in formato long

```
. bloomimport using "data/File270524bis.xlsx", sheet("FoglioA_download") cellrange(A4) ///
> export(long)
```

```
. summ
```

Variable	Obs	Mean	Std. dev.	Min	Max
<hr/>					
ticker	0				
year	1,691	2014	5.478846	2005	2023
ANN_NET_IN~N	1,545	3.650048	2.52467	.9355	92.0942
ARDR_LIQUI~0	38	118.4816	10.27879	106	165
ARDR_NET_S~0	14	3.43e+10	2.25e+10	124	4.80e+10
<hr/>					
BS_CUSTOME~S	1,557	35835.06	131601.6	219.2884	1482479
BS_DEMAND_~T	1,525	10774.64	38850.63	10.323	527748
BS_LEV_RAT~P	1,531	9.64031	1.861595	4	19.4
BS_LIQUIDI~0	38	118.4816	10.27879	106	165
BS_NON_PER~T	1,528	480.0016	4973.84	0	181872
<hr/>					
BS_NON_PER~S	1,522	416.6706	4513.457	0	166212
BS_RE_LOAN	1,554	14605.54	49766.79	66.217	495842
BS_RISK_WE~S	1,524	61941.18	957363	0	3.70e+07
BS_RSRV_LO~S	1,557	458.3737	1773.278	1.453	24516
BS_T~P_RATIO	1,544	12.80234	2.890617	6	32.57
<hr/>					
BS_T~Y_RATIO	895	11.65907	2.740535	-.1206	20.88
BS_TOT_ASSET	1,557	49138.65	186836.3	258.9534	1952911
BS_TOT_CAP~P	1,542	14.59836	2.864258	8	32.57
BS_TOT_LOAN	1,557	29945.87	102357.4	182.2295	993993
COM_EQY_TO~T	1,557	9.805642	2.3783	-1.6355	19.8588
<hr/>					
EFF_RATIO	1,556	60.47438	13.45661	11.1733	165.3167
EPS_GROWTH	1,530	16.11401	177.2297	-2575	3400
FDIC~250000K	18	12881.62	13000.88	1273.907	36408.25
GROWTH_IN_~T	1,540	10.90226	18.44787	-33.7049	362.9414
GROWTH_IN_~N	1,540	10.80239	18.11365	-63.1226	319.6432
<hr/>					
HISTORICAL~P	1,480	6492.774	24236.84	9.7506	296774.4
IS_ACT_LOA~T	1,538	162.7568	880.0535	-335	18168
NET_INC_GR~H	1,538	16.53337	220.1798	-2618.819	5497.222
NON_PERFOR~N	1,528	1.818147	15.20118	0	574.5461
NON_PERF_A~T	1,528	1.245741	11.03249	0	419.922
<hr/>					
NUM_OF_EMP~S	1,529	7255.333	28973.21	28	281000
RETURN_COM~Y	1,538	9.143434	14.5989	-264.7305	203.696
RETURN_ON_~T	1,540	.9632708	.784239	-6.0911	7.0985
RETURN_TOT~Y	1,538	9.511867	9.280677	-124.8476	74.5401
TANGIBLE_C~Y	1,523	3584.578	13595.56	-109.045	149373
<hr/>					
TEXAS_RATIO	1,492	16.06594	148.3282	0	5481.544
TOT_LOAN_T~T	1,557	87.59001	17.23081	15.3967	219.9147

Nel file "imp\_bloom.do", oltre al codice per replicare i due precedenti esempi, sono riportati ulteriori esempi relativi a dati presenti nel file testXbloomimport.xlsx.

### 3. Note Finali

bloomimport usa il comando aggiuntivo xframeappend di Roger Newson (Imperial College London, United Kingdom) e include un pezzo di codice scritto da William Matsuoka per convertire l'identificativo delle colonne di Excel, nel corrispondente numero di colonna. Trovate l'articolo e il codice qui: [Putexcel Part II: numofbase26\(\)](#).

Dati esportati con layout differente da quello previsto per il comando `bloomimport` dovrebbero essere importabili abbastanza facilmente con poche righe di codice. Se così non fosse, provate a sottoporvi il caso e vediamo cosa si può fare.

Se un campo viene, per errore, esportato due volte, per la seconda occorrenza viene aggiunto 2 alla fine del nome della variabile risultante in Stata. Per adesso non è contemplato il caso in cui lo stesso campo venga esportato tre o più volte.

Il comando funziona solo dalla versione 17 di Stata perché usa i frame per costruire il database finale. Ho in programma di renderlo compatibile anche con versioni precedenti ma per il momento la precedenza è al testing con diverse fonti di dati esportate da Bloomberg.

## Riferimenti bibliografici

- [1] Roger Newson, 2021. "XFRAMEAPPEND: Stata module to append data frames to the end of the current data frame", Statistical Software Components S458923, Boston College Department of Economics. <https://ideas.repec.org/c/boc/bocode/s458923.html>.
- [2] Nicola Tommasi, 2022. [Pagina del comando su GitHub](#).
- [3] William Matsuoka, 2015. [Putexcel Part II: numofbase26\(\)](#).