Structure of CSV clearing/trading files

**History of changes**

|  |  |
| --- | --- |
| 18.12.2002 | The following fields were added into ‘f07.dbf’: ‘poses\_rubl’, ‘limit’, ‘kof’, ‘risk\_wr’, ‘coffout’, ‘base\_fut’, ‘is\_spread’, ‘name’, ‘date2’. The following fields were removed: ‘sel\_depo’, ‘buy\_depo’. Notes have been added. |
| The following fields were added into ‘o07.dbf’: ‘poses\_rubl’, ‘fut\_contr’, ‘strike put’, ‘evrop’. |
| Description of the ‘go\_netto’ field was edited. Notes have been edited. |
| Description of the ‘go\_netto’ field was edited in ‘fposclXXYY.dbf’. Notes were edited. |
| Description of the ‘go’ field was edited in ‘oposXXYY.dbf’. |
| Description of the ‘go’ field was edited in ‘oposclXXYY.dbf’. |
| Notes were edited in ‘monXXYY.dbf’. |
| The following fields were removed from ‘f04\_XXYY.dbf’ and ‘f08\_XX00.dbf’: ‘sub\_sell’, ‘sub\_buy’, ‘comiss’, ‘nkd’, ‘name’, ‘summa’. Type and purpose of the ‘type’ was changed. The following fields were added: ‘fee\_buy’, ‘fee\_sell’, ‘date2’. Notes were added. |
| The following fields were removed from ‘o04\_XXYY.dbf’ and ‘o08\_XX00.dbf’: ‘sub\_sell’, ‘sub\_buy’, ‘comiss’, ‘nkd’, ‘type’, ‘name’, ‘summa’. Type and purpose of the ‘type’ field was changed. The following fields were added: ‘fee\_buy’, ‘fee\_sell’, ‘date2’. Notes were added. |
| 30.12.2002 | The ‘execution2’ field was added into ‘f07.dbf’. |
| The ‘date2’ and ‘execution2’ fields were added into ‘o07.dbf’. Notes were added. |
| The ‘date2’ field was added into ‘fordlogXXYY.dbf’. Notes were added. |
| The ‘date2’ field was added into ‘oordlogXXYY.dbf’. |
| 20.02.2003 | The ‘deposit’ field was added into ‘f07.dbf’. |
| The ‘depo\_uncov’, ‘depo\_сov’ fields were added into ‘o07.dbf’. |
| 28.04.2003 | The ‘user\_to’ field was added into ‘oordlogXXYY.dbf’. |
| 24.09.2003 | The ‘pos\_exec’ field was added into ‘oposXXYY.dbf’ and ‘oposclXXYY.dbf’. |
| 09.12.2003 | The ‘pos\_endcir’ field was added into ‘oposXXYY.dbf’ and ‘oposclXXYY.dbf’. |
| 10.02.2004 | The ‘comm\_buy’ and ‘comm\_sell’ fields were added into ‘f04\_XXYY.dbf’, ‘o04\_XXYY.dbf’, ‘f08\_XX00.dbf’ and ‘o08\_XX00.dbf’. |
| 26.02.2004 | The ‘payXX00.dbf’ file was added. |
| 16.08.2004 | The ‘comment’ field was added into ‘oordlogXXYY.dbf’. |
| 05.03.2005 | The ‘du\_buy’, ‘du\_sell’ fields and p.15 of notes were added into ‘f04\_XXYY.dbf’.  |
| The ‘du\_buy’, ‘du\_sell’ fields and p.13 of notes were added into ‘o04\_XXYY.dbf’.  |
| The ‘du’ field was added into ‘fposXXYY.dbf’. |
| The ‘du’ field was added into ‘fposclXXYYZZZ.dbf’. |
| The ‘du’ field was added into ‘oposXXYY.dbf’. |
| The ‘du’ field was added into ‘oposclXXYYZZZ.dbf’. |
| The ‘du’ field was added into ‘monXXYY.dbf’. |
| The ‘du’ field was added into ‘monXXYYZZZ.dbf’. |
| The ‘du\_buy’ and ‘du\_sell’ fields were added into ‘f08\_XX00.dbf’. |
| The ‘du\_buy’ and ‘du\_sell’ fields were added into ‘o08\_XX00.dbf’. |
| The ‘du’ field was added into ‘fordlogXXYY.dbf’. |
| The ‘du’ field was added into ‘oordlogXXYY.dbf’. |
| The ‘du’ field was added into ‘payXX00.dbf’. |
| 14.06.2005 | The ‘sbor\_exec’ field was added into ‘fposXXYY.dbf’. |
| The ‘sbor\_exec’ field was added into ‘fposclXXYY.dbf’. |
| The ‘sbor\_exec’ field was added into ‘oposXXYY.dbf’. |
| The ‘sbor\_exec’ field was added into ‘oposclXXYY.dbf’. |
| 16.06.2005 | The ‘name’, ‘close’\_’time’, ‘volat’, ‘theorprice’ fields were added into ‘o07.dbf’. |
| ‘fut\_deal.dbf’ and ‘opt\_deal.dbf’ were added. |
| 24.08.2005 | The ‘fee\_ns\_b’, ‘fee\_ns\_s’ fields and p.16, 17 of notes were added into ‘f04\_XXYY.dbf’. |
| The ‘fee\_ns\_b’, ‘fee\_ns\_s’ fields and p.14, 15 of notes were added into ‘o04\_XXYY.dbf’. |
| The‘sbor\_nosys’ field was added into ‘fposXXYY.dbf’. p.5,6 of notes were edited. |
| The ‘sbor\_nosys’ field was added into ‘fposclXXYYZZZ.dbf’. |
| The ‘sbor\_nosys’ field was added into ‘oposXXYY.dbf’, p.5 and 6 of notes were edited. |
| The ‘sbor\_nosys’ field was added into ‘oposclXXYYZZZ.dbf’. |
| The ‘fee\_ns\_b’ and ‘fee\_ns\_s’ fields were added into ‘f08\_XXYY.dbf’. |
| The ‘fee\_ns\_b’ and ‘fee\_ns\_s’ fields were added into ‘o08\_XXYY.dbf’. |
| 29.08.2005 | p.3, 6 of notes were edited in f04\_XXYY.dbf. |
| p.3 of notes was edited in o04\_XXYY.dbf. |
| 15.09.2005 | The ‘pem\_buy’ and ‘prem\_sell’ fields were added into ‘o04\_XXYY.dbf’. p.6 of notes was edited. PP.16-19 of notes were added. |
| The‘pem\_buy’ and ‘prem\_sell’ fields were added into ‘o08\_XXYY.dbf’. |
| 24.01.2006 | The ‘comment’ field’s type was changed from ‘char(10)’ to ‘char(50)’ in ‘payXX00.dbf’. |
| 31.03.2006 | The ‘vol\_rubl’ and ‘poses\_rubl’ fields’ types were changed from ‘numeric(16, 5)’ to ‘numeric(17, 2)’ in ‘f07.dbf’. |
| The ‘vol\_rubl’ and ‘poses\_rubl’ fields’ types were changed from ‘numeric(16, 5)’ to ‘numeric(17, 2)’ in ‘o07.dbf’. |
| 29.05.2006 | The ‘user\_to’ field’s purpose was changed in ‘fordlogXXYY.dbf’. In this field one should type in the RTS code of the company, to which the order was directed, instead of a trading system user, to whom the order was directed. The ‘price\_rur’ and ‘ext\_id’ fields were added. The ‘price’ field’s type was changed from ‘numeric(20, 5)’ to ‘numeric(16, 5)’. |
| The ‘user\_to’ field’s purpose was changed in ‘oordlogXXYY.dbf’. In this field one should type in the RTS code of the company, to which the order was directed, instead of a trading system user, to whom the order was directed. The ‘price\_rur’ and ‘ext\_id’ fields were added. The ‘price’ field’s type was changed from ‘numeric(20, 5)’ to ‘numeric(16, 5)’. |
| The ‘is\_percent’, ‘perc\_rate’, ‘sett\_rur’ fields were added into ‘f07.dbf’. |
| p.3, 4, 6 of notes were edited in f04\_XXYY.dbf. The ‘price\_rur’, ‘ext\_id\_b’, ‘ext\_id\_s’ fields were added. The ‘price’ field’s type was changed from ‘numeric(20, 5)’ to ‘numeric(16, 5)’. |
| The ‘price\_rur’ and ‘ext\_id\_b, ext\_id\_s’ fields were added into ‘f08\_XXYY.dbf’. The ‘price’ field’s type was echanged from ‘numeric(20, 5)’ to ‘numeric(16, 5)’. |
| The ‘type’ field was added into ‘fut\_deal.dbf’ and ‘opt\_deal.dbf’. OTC trades are reported now in ‘fut\_deal.dbf’ and ‘opt\_deal.dbf’ too. P.4 of notes was added. |
| P.4 of notes was edited in ‘o04\_XXYY.dbf’. The ‘price\_rur’, ‘ext\_id\_b’ and ‘ext\_id\_s’ fields were added. the ‘price’ field’s type was changed from ‘numeric(20, 5)’ to ‘numeric(16, 5)’. |
| The ‘price\_rur’, ‘ext\_id\_b’ and ‘ext\_id\_s’ fields were added into ‘o08\_XXYY.dbf’. The ‘price’ field’s type was changed from ‘numeric(20, 5)’ to ‘numeric(16, 5)’. |
| 15.06.2006 | ‘deliveryXX00.dbf’ was added. |
| The ‘payer’, ‘inn’, ‘bik’ and ‘purpose’ fields were added into ‘payXX00.dbf’. |
| 30.10.2006 | The ‘lot\_volume’ and ‘tick\_pr\_go’ fields were added into ‘f07.dbf’. |
| The ‘tick\_pr\_go’ field was added into ‘o07.dbf’. |
| 09.03.2007 | ‘delinfoXX00.dbf’ was added. |
| ‘clientsXX00.dbf’ was added. |
| 31.07.2007 | p.5 of notes was edited in ‘f04\_XXYY.dbf’. |
| p.5 of notes was edited in ‘o04\_XXYY.dbf’. |
| The ‘limi\_l1’ field was added into ‘f07.dbf’. |
| The‘gowide’, ‘freewide’ and ‘margincall’ fields were added into ‘monXXYY.dbf’. |
| The‘gowide’, ‘freewide’ and ‘margincall’ fields were added into ‘monclXXYY.dbf’. |
| ‘daymonXXYY.dbf’ was added. |
| 01.08.2007 | The ‘pr\_setll’ and ‘pr\_settl\_r’ fields were added into ‘f07.dbf’. |
| The ‘pr\_volat’ and ‘pr\_theorpr’ fields were added into ‘o07.dbf’. |
| 03.09.2007 | The ‘exp\_vm’ field was added into ‘daymon.dbf’. The following fields’ descriptions were edited: ‘fut\_sbor’, ‘opt\_sbor’, ‘amountwide’. |
| 31.10.2007 | The ‘type\_exec’ field was added into ‘f07.dbf’. |
| 16.05.2008 | The ‘date\_exp’, ‘n\_order1’ and ‘date\_clr’ fields were added into ‘fordlogXXYY.dbf’ |
| The ‘date\_exp’, ‘n\_order1’ and ‘date\_clr’ fields were added into ‘oordlogXXYY.dbf’ |
| The ‘date\_clr’ field was added into ‘f04\_XXYY.dbf’  |
| The ‘date\_clr’ field was added into ‘o04\_XXYY.dbf’  |
| The ‘date\_clr’ field was added into ‘f08\_XXYY.dbf’  |
| The ‘date\_clr’ field was added into ‘o08\_XXYY.dbf’. |
| The ‘pos\_iskl’, ‘pos\_neisp’, ‘neisp’ and ‘step’ fields were added into ‘deliveryXX00.dbf’. p.5 of notes was added into ‘deliveryXX00.dbf’. |
| Description of ‘delivery\_step1XX00.dbf’ was added. |
| 10.12.2008 | Descriptions of ‘tranfeeXXYY.dbf’, ‘tranoXXYY.dbf’, ‘tranfXXYY.dbf’were added. |
| 10.02.2009 | The ‘repo\_id’ field was added into ‘f04\_XXYY.dbf’. The following fields’ descriptions were edited: ‘type’, ‘user\_sell’, ‘user\_buy’. |
| The ‘section’, ‘spot’, ‘base’ and ‘type\_sbor’ fields were added into ‘f07.dbf’ |
| The ‘repo\_id’ field was added into ‘f08\_XX00.dbf’. The following fields’ descriptions were edited: ‘type’, ‘user\_sell’, ‘user\_buy’. |
| The ‘spot’ field was added into ‘clientsXX00.dbf’. |
| The‘accum\_go’ and ‘fee\_trans’ fields were added into ‘fposXXYY.dbf’. |
| The ‘accum\_go’ and ‘fee\_trans’ fields were added into ‘fposclXXYYZZZ.dbf’. |
| The ‘var-marg\_s’, ‘var\_marg\_b’ fields and notes to them were added into ‘o08\_XX00.dbf’ . Notes to the ‘prem\_buy’ and ‘prem\_sell’ fields were edited. |
| Notes to the ‘var\_marg’ and ‘prem’ fields were added into ‘monXXYY.dbf’. |
| Notes to the ‘var\_marg’ and ‘prem’ fields were added into ‘monclXXYY.dbf’. |
| The‘fut\_type’ field was added into ‘o07.dbf’. |
| The ‘var\_marg\_p’ and ‘var\_marg\_d’ fields were added into ‘oposXXYY.dbf’. |
| The ‘var\_marg\_b’ and ‘var\_marg\_s’ fields were added into ‘o04.dbf’. |
| 10.04.2009 | The ‘ns\_volume’, ‘ns\_trades’, ‘ns\_fee’, ‘ns\_volrubl’ and ‘l\_tradeday’ fields were added into ‘f07.dbf’. |
| The ‘dealtr’ field was removed from ‘tranfXXYY.dbf’. |
| The ‘dealtr’ field was removed from ‘tranoXXYY.dbf’. |
| The‘dealtr’ field was replaced by the ‘fee’ field in ‘tranfeeXXYY.dbf’. |
| The ‘gts\_dvp’ field was removed from ‘delinfoXX00.dbf’. The ‘code\_rps’ field was renamed to ‘codedealer’ and the ‘check’ field was added. |
| 16.04.2009 | A new type of trade was added to the ‘type’ field in ‘f04\_XXYY.dbf’. |
| A new type of trade was added to the ‘type’ field in ‘f08\_XX00.dbf’. |
| 20.04.2009 | ‘dayf07.dbf’ and ‘dayo07.dbf’ were added. |
| 30.07.2009 | The ‘basegobuy’ field was added to ‘o07.dbf’. |
| 28.08.2009 | The following changes were implemented in ‘f04\_XXYY.dbf’, ‘o04\_XXYY.dbf’, ‘f08\_XX00.dbf’ and ‘o08\_XX00.dbf’ * Descriptions of the following fields were edited: ‘fee\_buy’, ‘fee\_sell’, ‘fee\_ns\_b, ‘fee\_ns\_s’.
* The new fields were added: ‘fee\_ex\_b’, ‘vat\_ex\_b’, ‘fee\_cc\_b’, ‘vat\_cc\_b’, ‘fee\_ex\_s’, ‘vat\_ex\_s’, ‘fee\_cc\_s’, ‘vat\_cc\_s’.
 |
| The following changes were implemented in ‘fposXXYY.dbf’, ‘fposclXXYYZZZ.dbf’, ‘oposXXYY.dbf’, ‘oposclXXYYZZZ.dbf’:* Descriptions of the following fields were edited: ‘sbor’, ‘sbor\_exec’, ‘sbor\_nosys’
* The new fields were added: ‘sbor\_ex’, ‘vat\_ex’, ‘sbor\_сс’, ‘vat\_cc’
 |
| The following changes were implemented in ‘monXXYY.dbf’, ‘monclXXYYZZZ.dbf’:* Descriptions of the following fields were changed: ‘fut\_sbor’, ‘opt\_sbor’
* The new fields were added: ‘sbor\_ex’, ‘vat\_ex’, ‘sbor\_сс’, ‘vat\_cc’
 |
| Descriptions of the ‘fut\_sbor’ and ‘opt\_sbor’ fields were edited in ‘daymonXXYY.dbf’. |
| 30.09.2009 | The following changes were implemented in ‘tranfeeXXYY’: * Descriptions of the ‘fee’ and ‘sbortr’ fields were edited, types of the fields were changed
* the ‘vat\_sbortr’ field was added
 |
| 12.03.2010 | The following changes were implemented in ‘f04\_XXYY.dbf’ in the ‘type’ field: * some new types of trades were added
* the field size was changed
 |
| The following changes were implemented in ‘f08\_XX00.dbf’ for the ‘type’ field: * some new types of trades were added
* the field size was changed
 |
| The field size was changed for the ‘type’ field in ‘o04\_XXYY.dbf’. |
| The field size was changed for the ‘type’ field in ‘o08\_XX00.dbf’. |
| 16.04.2010 | ‘repodeal\_XXYY.dbf’ and ‘repoordlog\_XXYY.dbf’ were added. |
| 19.04.2010 | The following changes were made to the fields in ‘dayf07.dbf’:* ‘vol\_rubl’ – the field size was changed from 16,2 to 17,2
* ‘fee’ – the field size was changed from 16,2 to 16,5
* ‘poses\_rubl’ – the field size was changed from 16,2 to 17,2
* ‘kof’ – the field size was changed from 9,6 to 10,6
* ‘risk\_wr’ – the field size was changed from 16,2 to 16,5
* ‘coffout’ – the field size was changed from 16,5 to 7,5
* ‘is\_spread’ – the field size was changed from 10 to 1
* ‘deposit’ – the field size was changed from 16,2 to 16,5
* ‘is\_percent’ – the field size was changed from 10 to 1
* ‘perc\_rate’ – the field size was changed from 16,2 to 7,2
* ‘step\_price’ – the field was removed
* ‘tick\_pr\_go’ – the field was added
* ‘limit\_l1’ – the field was added
* ‘type\_exec’ – the field size was changed from 10 to 1, a new trade type was added
 |
| The following changes were made to the fields in ‘f07.dbf’:* pr\_settl\_r - the field size was changed from 16,6 to 16,5
* type\_exec – a new trade type was added
 |
| The ‘basegobuy’ field was added into ‘dayo07.dbf’. |
| The ‘fee\_exec’ and ‘fine\_exec’ fields were added into ‘fposclXXYYZZZ.dbf’. |
| ‘f04clXXYYZZZ.dbf’, ‘o04clXXYYZZZ.dbf’ and ‘payclXXYYZZZ.dbf’ were added. |
| The ‘f08\_XX00.dbf’and ‘o08\_XX00.dbf’ files are no longer available. |
| The ‘fut\_deal.dbf’ and ‘opt\_ deal.dbf’ files are no longer distributed within EDF system. |
| 07.06.2010 | The new type (‘11’) was added into the ‘type’ field in ‘f04\_XXYY.dbf’ and ‘f04cl\_XXYYZZZ.dbf’. |
| 11.06.2010 | The following fields were added into ‘f04\_XXYY.dbf’and ‘f04cl\_XXYYZZZ.dbf’:* ‘id\_mult’
* ‘signs’
 |
| The following field was added into ‘o04\_XXYY.dbf’ and ‘o04cl\_XXYYZZZ.dbf’:* ‘signs’
 |
| 26.08.2010 | The following files were added: ‘multilegf04\_XXYY.dbf’, ‘multilegf04clXXYYZZZ.dbf’, ‘multilegordlog\_XXYY.dbf’, ‘multileg\_deal.dbf’, ‘multileg\_dict.dbf’. |
| The following changes were made to‘f07.dbf’:* ‘contract’ – the field description was edited
* ‘multileg’ – the new field was added
 |
| Additional information (the link to **ftp://ftp.rts.ru/pub/FORTS/JointDeliveryFORTS/Информация для бэк-офисов Участникам.doc**) added into the ‘signs’ field in ‘f04\_XXYY.dbf’, ‘f04cl\_XXYYZZZ.dbf’, ‘o04\_XXYY.dbf’ and ‘o04cl\_XXYYZZZ.dbf’. |
| 27.08.2010 | The following changes were made to ‘dayf07.dbf’:* ‘contract’ – the field description was edited
* ‘multileg – the field was added
 |
| 27.12.2010 | The ‘price\_rur2’ field was added into ‘multilegf04\_XXYY.dbf’ and ‘multilegf04cl\_XXYYZZZ.dbf’. |
| The ‘money’ field was added into ‘clientsXX00.dbf’. |
| The ‘depo\_acc’ field size was changed from char (15) to char (20) in delinfoXX00.dbf.  |
| The ‘rate’ field size was changed from numeric(16,2) to numeric(16,5) in ‘multilegf04\_XXYY.dbf’, ‘multilegf04cl\_XXYYZZZ.dbf’, ‘multilegordlog\_XXYY.dbf’, ‘multileg\_deal.dbf’, ‘repodeal\_XXYY.dbf’ and ‘repoordlog\_XXYY.dbf’. |
| The new types (‘12’ and ‘13’) were added into the ‘type’ field in ‘f04\_XXYY.dbf’, ‘f04cl\_XXYYZZZ.dbf’. |
| 01.03.2011 | The ‘repodeal\_XXYY.dbf’ and ‘repoordlog\_XXYY.dbf’ files are no longer available. |
| 23.05.2011 | The following fields were added into ‘clientsXX00.dbf’:* ‘du’
* ‘segr’
* ‘isrepo’
 |
| 27.05.2011 | The following fields descriptions were edited in ‘multilegordlog\_XXYY.dbf’:* ‘price’
* ‘price\_rur’
 |
| 01.07.2011 | The ‘fordlogXXYY.dbf’ and ‘oordlogXXYY.dbf’ files are no longer available. |
| 19.12.2011 | ‘RTS Standard’ was changed to ‘Standard’. |
| 02.02.2012 | ‘moncbXXYY.dbf’ was added. |
| 21.05.2012 | The following changes were made to ‘delinfoXX00.dbf’:* The ‘depo\_acc’ field size was changed from char(20) to char(32)
* The ‘codedealer’ field was renamed to ‘id’. The field description was renewed.
* The ‘shortname’ field was added.
 |
| 08.10.2012 | The ‘rate’ field size was changed from numeric (16,2) to numeric (16,5) in ‘moncbXXYY.dbf’. |
| 03.12.2012 | Information about RTS Money instrument was deleted due to the market liquidation. |
| 21.12.2012 | The following fields sizes were changed in ‘f04\_XXYY.dbf’, ‘f04cl\_XXYYZZZ.dbf’, ‘o04\_XXYY.dbf’, ‘o04cl\_XXYYZZZ.dbf’, ‘multilegf04\_XXYY.dbf’ and ‘multilegf04clXXYYZZZ.dbf’:* ‘no\_buy’ – from numeric(10) to numeric(15)
* ‘no\_sell’ – from numeric(10) to numeric(15)
* ‘signs’ – from numeric(10) to numeric(11)
 |
| The following fields sizes were changed in ‘multilegordlog\_XXYY.dbf’:* ‘numb\_order’ – from numeric(10) to numeric(15)
* ‘n\_order1’ – from numeric(10) to numeric(15)
 |
| 08.01.2013 | Fields containing VAT values are not used due to the changes to the Tax Code of Russia. |
| 18.02.2013 | The following changes have been made to ‘delinfoXX00.dbf’:* The ‘id’ field size was changed from char(12) to char(25)
* The ‘id’ field description was edited.
 |
| 27.05.2013 | monXXYY.dbf, monclXXYYZZZ.dbf, and daymonXXYY.dbf have been changed to include the following field:* ext\_rez
 |
|  | f04\_XXYY.dbf and f04cl\_XXYYZZZ.dbf have been changed to include types “14” and “15” to the “type” field |
|  | fposXXYY.dbf and fposclXXYYZZZ.dbf have been changed to include:* the “pos\_failed” field
 |
| 01.10.2013 | The dayf07, dayo07 and daymonXXYY files have been transferred to the csv format (from dbf) |
| 21.10.2013 | The tranfXXYY.dbf and tranoXXYY.dbf files have been withdrawn  |
|  | tranfeeXXYY.dbf has been changed as follows:* descriptions of the futopt, mm, fee, sbortr and vat\_sbortr files have been changed
* the new “trantype” filed has been added
* the csv format has been introduced
 |
|  | The following files with ineffective transactions have been added:* tranfeeupdXXYY.csv
* tranfeeupddetailsXXYY.csv
* tranfeeshareXXYY.csv
 |
|  | The following files with erroneous transactions have been added:* usersXXYY.csv
* tranerrXXYY.csv
* tranerrfeeXXYY.csv
 |
| 11.11.2013 | Csv format has been introduced for the following files:clientsXX00delinfoXX00delivery\_step1XX00deliveryXX00f04\_XXYYf04clXXYYZZZf07fposXXYYfposclXXYYZZZfut\_dealmonXXYYmonclXXYYZZZmoncbXXYYmultileg\_dealmultileg\_dictmultilegf04\_XX00multilegf04clXXYYZZZmultilegordlog\_XX00o04\_XXYYo04clXXYYZZZo07oposXXYYoposclXXYYZZZopt\_dealpayXX00payclXXYYZZZvolat |
| 18.11.2013 | The daymonXXYY.csv file has been changed as follows:* the rub\_beg, rub\_pay, rubl2, com\_pl\_beg, com\_pl\_pay, and com\_pll2 fields have been added,
* descriptions of the type, amount\_beg, pay, amountl2, and freel2 fileds have been changed
 |
|  | monXXYY.csv and monclXXYY.csv have been changed as follows:* the rub\_beg, rub\_pay, rub\_end, com\_pl\_beg, com\_pl\_pay, and com\_pl\_end fields have been added
* descriptions of the type, amount\_beg, pay, amount\_end, and free fields have been changed
 |
|  | The following filed has been added to moncbXXYY.csv:* com\_ensure
 |
|  | Description of the following filed has been changed in payXX00.csv and payclXXYYZZZ.csv:* type
 |
| 16.12.2013 | fut\_deal.csv has been changed as follows:* size of the type filed has been changed to numeric(2) (from numberic(1))
* description of the type filed has been changed
 |
| 25.04.2014 | The paycbXX00.csv file has been added |
| 09.06.2014 | The toeqXXYY.csv file has been added |
| 28.07.2014 | The **riskposXXYY.csv and tofxXXYY.csv files have been added** |
|  | The following fields have been added to clientsXX00.**csv**:* rk
* rk\_type
 |
|  | The new type “16” has been added to type field of f04\_XXYY. **Csv** |
|  | monXXYY.csv and **daymonXXYY**.csv have been changed as follows:* descriptions of the kod and account fields have been changed
 |
| 22.08.2014 | The id\_code field in clientsXX00.**csv has been changed as follows:*** the size has been changed to char(50) (from char(30))
 |
| 03.10.2014 | The new type “16” has been added to the field “type” of fut\_deal.**csv.**Description of the field “id” has been changed in delinfoXX00.csv |
| 05.12.2014 | The field “broker\_ref” has been added to toeqXXYY.**csv** |
|  | Descriptions of the fields have been changed due to abandonement of the Standard Market  |
|  | persinvestXX00.csv has been added  |
| 06.04.2015 | f04\_XXYY.csv, f04clXXYYZZZ.csv, o04\_XXYY.csv, o04clXXYYZZZ.csv: |
|  | * Description of the field “signs” has been changed
 |
|  | clientsXX00.**csv**:* “cross\_trade” and “account\_forts” have been added
* “spot” and “money” have been added
 |
| 03.06.2015 | delivery\_ofz\_ctd\_XX00.csv has been added |
| 01.07.2015 | **mmfutXXYY.csv, mmopt\_strikesXXXX.csv and mmopt\_averageXXXX.csv have been added** |
| 06.08.2015 | delinfoXX00.csv has been cancelled |
| 31.08.2015 | f04\_XXYY.csv, f04clXXYYZZZ.csv, o04\_XXYY.csv, o04clXXYYZZZ.csv:* The field “counterparty” has been added
 |
| 04.07.2016 | clientsXX00.**csv**:* The field “margin\_type” has been added
* The size of “rk” has been changed to char(12) (from char(5))
 |
|  | monXXYY.csv, daymonXXYY.csv, monclXXYYZZZ.csv:* The size of “kod” has been changed from char(7) to char(12)
 |
| 16.08.2016 | riskparamsXXYY.csv has been added  |
| 04.10.2016 | usersfcXXYY.csv and tranfсfeeXXYY.csv have been added |
| 01.12.2016 | Description of filed “is\_percent” has been updated in f07.csv and dayf07.csv  |
| New type “2” has been added to filed trantype of tranfeeXXYY.csv |
| Phrase “erroneous transactions” has been changed to “erroneous transactions other than Flood Control errors” in the following files:* usersXXYY.csv
* tranerrXXYY.csv
* tranerrfeeXXYY.csv
 |
| 26.12.2016 | Changes in riskparamsXXYY.csv:* Fields “type”, “del\_ord”, “prohibit\_coeff\_bf”, “state\_bf”, “type\_bf”, and “del\_ord\_bf” have been added.
 |
| File dayriskparamsXXYY.csv has been added |
| Descriptions of fields “payer”, “inn” and “bik” have been changed in files payXX00.csv and payclXXYYZZZ.csv |
| 27.02.2017 | File payXXYY.csv has been added |
| 01.06.2017 | File “mmLP\_XXYY.csv**”** has been added |
| 04.09.2017 | Changes in clientsXX00.csv:* Added field: “sb”
 |

1. File f04\_XXYY.csv. Information on trades in futures of the brokerage firm (BF) and its clients (XXYY is the brokerage firm code). The file is in CSV format.

Structure of the file **f04\_XXYY.csv**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| id\_deal | numeric(10) | Trade ID in the trading system  |
| isin | char(25) | Unique code of the instrument (futures contract) |
| price | numeric(16,5) | Futures trade price (in points) |
| vol | numeric(10) | Number of instruments in the trade (units) |
| kod\_sell | char(7) | Seller code |
| kod\_buy | char(7) | Buyer code |
| date | char(10) | Trade date in YYYY/MM/DD format |
| time | char(8) | Trade time |
| profit\_usd | numeric(20,4) | not used |
| type | numeric(2) | Trade type:0 – anonymous;1 – options expiration;2 – negotiated trade/position transfer between section with the same identity details with the bit mask attribute of 0х8 шт the gateway and a unique non-zero id in the field “signs”;3 – off book trade executed during the clearing session, not used;4 – the first leg of repo, not used;5 – the second leg of repo, not used;7 – selling/buying across asset manager’s sections (instead of repo trades), not used;8 – settlement of the deliverable futures via the Standard instrument, not used;9 – the first leg of repo trades T+N, not used;10 – the second leg of repo trades T+N, not used;11 – close of the position after settlement of the cash-settled/deliverable futures (recorded as the counter-directed trade to close the position in the register);12 – not used;13 – not used;14 – the first trade for futures contracts of the calendar spread;15 – the second trade for futures contracts of the calendar spread;16 – trade in a risk management instrument. |
| var\_marg\_b | numeric(16,2) | Buyer’s variation margin (RUB) |
| var\_marg\_s | numeric(16,2) | Seller’s variation margin (RUB) |
| user\_sell | char(20) | Seller’s login in the trading system. The special code OPERATOR\_REPO (CC’s login) for repo trades and buying/selling across the asset manager’s sections |
| user\_buy | char(20) | Buyer’s login in the trading system. The special code OPERATOR\_REPO (CC’s login) for repo trades and buying/selling across the asset manager’s sections |
| no\_buy | numeric(15) | No. of the buyer’s order in the trading system  |
| no\_sell | numeric(15) | No. of the seller’s order in the trading system  |
| fee\_buy | numeric(16,2) | Exchange and clearing fees of the buyer (RUB) |
| fee\_sell | numeric(16,2) | Exchange and clearing fees of the seller (RUB) |
| date2 | Date | Trade date |
| comm\_buy | char(20) | Notes from the buyer  |
| comm\_sell | char(20) | Notes from the seller |
| du\_buy | numeric(1) | Attribute of the seller: 1 - the seller is the trust manager, 0 - the seller is not the trust manager) |
| du\_sell | numeric(1) | Attribute of the buyer: 1 - the buyer is the trust manager, 0 - the buyer is not the trust manager) |
| fee\_ns\_b | numeric(16, 2) | Exchange and clearing fees for negotiated trades by the buyer (RUB) |
| fee\_ns\_s | numeric(16, 2) | Exchange and clearing fees for negotiated trades by the seller (RUB) |
| price\_rur | numeric(16, 5) | Futures price (in RUB) |
| ext\_id\_b | numeric(11) | Buyer external ID |
| ext\_id\_s | numeric(11) | Seller external ID |
| date\_clr | date | Clearing date |
| repo\_id | numeric(11) | ID of the related repo leg, not used |
| fee\_ex\_b | numeric(16,2) | Exchange fee for the buyer (RUB) |
| vat\_ex\_b | numeric(16,2) | VAT in the Exchange fee (RUB) – not used |
| fee\_cc\_b | numeric(16,2) | Fee charged by the buyer’s clearing house (RUB) |
| vat\_cc\_b | numeric(16,2) | VAT in the buyer’s clearing fee (RUB)– not used |
| fee\_ex\_s | numeric(16,2) | Seller’s exchange fee (RUB) |
| vat\_ex\_s | numeric(16,2) | VAT in the seller’s Exchange fee (RUB) – not used |
| fee\_cc\_s | numeric(16,2) | Fee charged by the seller’s clearing house (RUB) |
| vat\_cc\_s | numeric(16,2) | VAT in the seller’s clearing fee (RUB) – not used |
| id\_mult | numeric(10) | Attribute for a technical trade constituting a part of the multi-leg trade. This field contains an ID of the multi-leg trade. This field is similar to field id\_deal\_multileg in Plaza-2 gateway in the trade table (see file “Информации дл бэк-офисов Участни” at ftp://ftp.moex.com/pub/FORTS/). This field is empty for regular trades. |
| signs | numeric(11) | Bitmask of the trade attributes. It is the full analogue of the bit mask of the trade attributes in the Plaza-2 gateway’s trade table (please visit ftp://ftp.moex.com/pub/FORTS/Plaza2/docs/, file «p2gate\_ru.pdf ) |
| counterparty | char(7) | Code of the counterparty to the trade (code of the clearing member or brokerage firm of the clearing member) executed on the basis of the non-anonymous order |

Notes:

1. “id\_deal” corresponds with the trade ID assigned at the gateway.
2. “f04\_XXYY.isin” = “f07.contract”.
3. Specific formulae are applied to convert the price into roubles.
4. If “f07.is\_percent” = 0,



1. If “f07.is\_percent” = 1,



where n – the difference in days between “f07.execution” and “f07.date”.

1. This file contains all trades of the brokerage firm’s clients ‘XXYY’. If “kod\_sell” <> ‘’, “kod\_sell” is the BF’s client ‘XXYY’, or not BF’s client otherwise. If “kod\_buy” <> ‘’, “kod\_buy” is the BF’s client ‘XXYY’, or not BF’s client otherwise. If “kod\_sell” = ‘’, “var\_marg\_s” = 0, “user\_sell” = ‘’, “no\_sell” = 0, “fee\_sell” = 0, “comm.\_sell” = ‘’, “ext\_id\_s” = 0. If “kod\_buy” = ‘’, “var\_marg\_b” = 0, “user\_buy” = ‘’, “no\_buy” = 0, “fee\_buy” = 0, “comm.\_buy” = ‘’, and “ext\_id\_b” = 0.
2. If “type” = 3, then “id\_deal” = 0; “user\_buy” = ’’; “user\_sell” = ’’; “no\_buy” = 0; “no\_sell” = 0. If “type” = 1, then “id\_deal” = 0, if the settlement trade was made at clearing, or “id\_deal” <> 0, if the settlement trade was made at trading; “user\_buy” = ’’, “user\_sell” = ’’, “no\_buy” = 0, and “no\_sell” = 0.
3. If “kod\_buy” <> ‘’ and “f07.is\_percent” = 0, “var\_marg\_b” = vol \* round((f07.settl – price) \* f07.tick\_price / f07.tick, 2). If kod\_sell <> ‘’ and f07.is\_percent = 0, var\_marg\_s = vol \* round((price - f07.settl) \* f07.tick\_price / f07.tick, 2). If kod\_buy <> ‘’ and f07.is\_percent = 1, var\_marg\_b = vol \* round((1000000 / ((1 + price / 36500) ^ n) – 1000000 / ((1 + f07.settl / 36500) ^ n)), 2). If “kod\_sell” <> ‘’ and f07.is\_percent = 1, var\_marg\_s = vol \* round((1000000 / ((1 + f07.settl / 36500) ^ n) – 1000000 / ((1 + price / 36500) ^ n)), 2), where n – the difference in days between “f07.execution” and “f07.date”.
4. “no\_buy” and “no\_sell” are IDs of the orders resulted in the trade. Such IDs correspond with IDs assigned during the gateway procedures.
5. The field “date2” is of similar import to the field date. The fields differ only in formats.
6. For all BF’s trades in the instrument isin: ∑(fee\_buy + fee\_sell) equals “fposXXYY.sbor”, where “fposXXYY.date” = “date2” and “fposXXYY.isin” = “isin” and “fposXXYY.kod” = ‘XXYY’ + ‘000’ and “fposXXYY.account” = ‘BF’.
7. For all trades in the instrument isin of the client ‘XXYYYYY’: ∑(fee\_buy + fee\_sell) equals “fposXXYY.sbor”, where “fposXXYY.date” = “date2” and “fposXXYY.isin” = “isin” and “fposXXYY.kod” = ‘XXYYYYY’ and “fposXXYY.account” = ‘CL’.
8. For all BF’s trades in all instruments: ∑(fee\_buy + fee\_sell) equals ‘monXXYY.fut\_sbor”, where “monXXYY.date” = “date2” and “monXXYY.kod” = ‘XXYY’ + ‘000’ and “monXXYY.account” = ‘BF’ and “monXXYY.type” = ‘MN’.
9. For all trades of the client ‘XXYYYYY’ in all instruments: ∑(fee\_buy + fee\_sell) equals “monXXYY.fut\_sbor”, where “monXXYY.date” = “date2” and “monXXYY.kod” = ‘XXYYYYY’ and “monXXYY.account” = ‘CL’ and “monXXYY.type” = ‘MN’.
10. For all BF’s trades in the instrument isin: ∑(var\_marg\_b + var\_marg\_s) equals “fposXXYY.var\_marg\_d”, where “fposXXYY.date” = “date2” and “fposXXYY.isin” = “isin” and “fposXXYY.kod” = ‘XXYY’ + ‘000’ and “fposXXYY.account” = ‘BF’.
11. For all trades of the client ‘XXYYYYY’ in the instrument isin: ∑(var\_marg\_b + var\_marg\_s) = “fposXXYY.var\_marg\_d”, where “fposXXYY.date” = “date2” and “fposXXYY.isin” = “isin” and “fposXXYY.kod” = ‘XXYYYYY’ and “fposXXYY.account” = ‘CL’.
12. If the counterparty to the trade acts on behalf of the trust manager, the fields “du\_buy” (or “du\_sell”) = 1 regardless of whether the counterparty is the client of the BF ‘XXYY’. For example, client XX00001 bought five contracts. The counterparty to the trade acts on behalf of the asset manager. In this case “kod\_buy” = ‘XX00001’, “kod\_sell” = ‘’, “du\_sell” = 1.
13. For all BF’s trades in the instrument isin: ∑(fee\_ns\_b + fee\_ns\_s) = “fposXXYY.sbor\_nosys”, where “fposXXYY.date” = “date2” and “fposXXYY.isin” = “isin” and “fposXXYY.kod” = ‘XXYY’ + ‘000’ and “fposXXYY.account” = ‘BF’.
14. For all trades of client ‘XXYYYYY’ in instrument isin: ∑(fee\_ns\_b + fee\_ns\_s) = “fposXXYY.sbor\_nosys”, where “fposXXYY.date” = “date2” and “fposXXYY.isin” = “isin” and “fposXXYY.kod” = ‘XXYYYYY’ and “fposXXYY.account” = ‘CL’.
15. File f04clXXYYZZZ.csv. Information on client trades in futures (XXYYZZZ is the section code). The file is in CSV format.

Structure of the file **f04clXXYYZZZ.csv**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| id\_deal | numeric(10) | Trade ID in the trading system  |
| isin | char(25) | Unique code of the instrument (futures contract) |
| price | numeric(16,5) | Futures price (in points) |
| vol | numeric(10) | Number of instruments in the trade (units) |
| kod\_sell | char(7) | Seller code |
| kod\_buy | char(7) | Buyer code |
| date | char(10) | Trade date in YYYY/MM/DD format |
| time | char(8) | Trade time |
| profit\_usd | numeric(20,4) | not used |
| type | numeric(2) | Trade type:0 – anonymous trade,1 – options expiration ,2 – negotiated trade/position transfer between section with the same identity details with the bit mask attribute of 0х8 шт the gateway and a unique non-zero id in the field “signs”, 3 – off book trade executed during the clearing session, not used,4 – repo first leg, not used,5 – repo second leg, not used,7 – selling/buying between sub-accounts in trust (instead of repo trades), not used,8 – settlement of the deliverable futures via the Standard instrument, not used,9 – the first leg of repo trades T+N, not used,10 – the second leg of repo trades T+N, not used,11 – close of the position after settlement of the cash-settled (recorded as the counter-directed trade to close the position in the register),12 – not used,13 – not used,14 – the first trade for futures contracts of the calendar spread,15 – the second trade for futures contracts of the calendar spread,16 – trade in a risk instrument. |
| var\_marg\_b | numeric(16,2) | Buyer’s variation margin (RUB) |
| var\_marg\_s | numeric(16,2) | Seller’s variation margin (RUB) |
| user\_sell | char(20) | Seller’s login in the trading system. The special code OPERATOR\_REPO (CC’s login) for repo trades and buying/selling across the asset manager’s sections |
| user\_buy | char(20) | Buyer’s login in the trading system. The special code OPERATOR\_REPO (CC’s login) for repo trades and buying/selling across the asset manager’s sections |
| no\_buy | numeric(15) | No. of the buyer’s order in the trading system  |
| no\_sell | numeric(15) | No. of the seller’s order in the trading system  |
| fee\_buy | numeric(16,2) | Exchange and clearing fees of the buyer (RUB) |
| fee\_sell | numeric(16,2) | Exchange and clearing fees of the seller (RUB) |
| date2 | Date | Trade date |
| comm\_buy | char(20) | Comments from the buyer  |
| comm\_sell | char(20) | Comments from the seller |
| du\_buy | numeric(1) | Attribute of the seller: 1 - the seller is the trust manager, 0 - the seller is not the trust manager) |
| du\_sell | numeric(1) | Attribute of the buyer: 1 - the buyer is the trust manager, 0 - the buyer is not the trust manager) |
| fee\_ns\_b | numeric(16, 2) | Exchange and clearing fees for negotiated trades by the buyer (RUB) |
| fee\_ns\_s | numeric(16, 2) | Exchange and clearing fees for negotiated trades by the seller (RUB) |
| price\_rur | numeric(16, 5) | Price of futures (RUB) |
| ext\_id\_b | numeric(11) | Buyer external ID |
| ext\_id\_s | numeric(11) | Seller external ID |
| date\_clr | date | Clearing date |
| repo\_id | numeric(11) | ID of the related repo leg, not used |
| fee\_ex\_b | numeric(16,2) | Exchange fee of the buyer (RUB) |
| vat\_ex\_b | numeric(16,2) | VAT in the Exchange fee (RUB) – not used |
| fee\_cc\_b | numeric(16,2) | Fee charged by the buyer’s clearing house (RUB) |
| vat\_cc\_b | numeric(16,2) | VAT in the buyer’s clearing fee (RUB)– not used |
| fee\_ex\_s | numeric(16,2) | Seller’s exchange fee (RUB) |
| vat\_ex\_s | numeric(16,2) | VAT in the seller’s Exchange fee (RUB) – not used |
| fee\_cc\_s | numeric(16,2) | Fee charged by the seller’s clearing house (RUB) |
| vat\_cc\_s | numeric(16,2) | VAT in the seller’s clearing fee (RUB) – not used |
| id\_mult | numeric(10) | Attribute of technical trades being a part of multi-leg trades. The field shows ID of the related trade. It is similar to the field id\_deal\_multileg in the Plaza-2 gateway’s trade table. Not shown for buy/sell trades. |
| signs | numeric(11) | Bit mask of the trade attributes. It is the full analogue of the bit mask of the trade attributes in the Plaza-2 gateway’s trade table (please visit ftp://ftp.moex.com/pub/FORTS/Plaza2/docs/, p2gate\_ru.pdf)  |
| counterparty | char(7) | Code of the counterparty to the trade (code of the clearing member or brokerage firm of the clearing member) executed on the basis of the non-anonymous order |

Notes:

1. The file is produced if the reporting service was ordered for the clearing registers section.

1. File o04\_XXYY.csv. Inforamtion on options trades of the brokerage form (BF) and its clients (XXYY is the brokerage firm code). The file is in CSV format.

Structure of the file **o04\_XXYY.csv**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| id\_deal | numeric(10) | Trade ID in the trading system  |
| isin | char(25) | Unique code of the instrument (options contract) |
| price | numeric(16,5) | Trade price (points) |
| vol | numeric(10) | Number of instruments in the trade (units) |
| kod\_sell | char(7) | Seller code |
| kod\_buy | char(7) | Buyer code |
| date | char(10) | Trade date in YYYY/MM/DD format |
| time | char(8) | Trade time |
| profit\_usd | numeric(20,4) | Not used |
| type | numeric(2) | Trade type:0 – anonymous trade,1 – options expiration, 2 – negotiated trade/position transfer between section with the same identity details with the bit mask attribute of 0х8 шт the gateway and a unique non-zero id in the field “signs”,3 – negotiated trade executed during the clearing session, not used,4 – options not exercised at expiry. |
| user\_buy | char(20) | Buyer’s login in the trading system |
| user\_sell | char(20) | Seller’s login in the trading system |
| no\_buy | numeric(15) | No. of the buyer’s order in the trading system  |
| no\_sell | numeric(15) | No. of the seller’s order in the trading system  |
| fee\_buy | numeric(16,2) | Sum of the Exchange fee and the buyer’s clearing center fee (RUB) |
| fee\_sell | numeric(16,2) | Sum of the Exchange fee and the seller’s clearing center fee (RUB) |
| date2 | date | Trade date |
| comm\_buy | char(20) | Comments from the buyer  |
| comm\_sell | char(20) | Comments from the seller |
| du\_buy | numeric(1) | Attribute of the seller: 1 - the seller is the trust manager, 0 - the seller is not the trust manager) |
| du\_sell | numeric(1) | Attribute of the buyer: 1 - the buyer is the trust manager, 0 - the buyer is not the trust manager) |
| fee\_ns\_b | numeric(16, 2) | Exchange and clearing fees for negotiated trades by the buyer (RUB) |
| fee\_ns\_s | numeric(16, 2) | Exchange and clearing fees for negotiated trades by the seller (RUB) |
| prem\_buy | numeric(16, 2) | Buyer premium (RUB)  |
| prem\_sell | numeric(16, 2) | Seller premium (RUB)  |
| price\_rur | numeric(16, 5) | Trade price in RUB |
| ext\_id\_b | numeric(11) | Buyer external ID |
| ext\_id\_s | numeric(11) | Seller external ID |
| date\_clr | date | Clearing date |
| var\_marg\_b | numeric(16,5) | Buyer variation margin (RUB) |
| var\_marg\_s | numeric(16,5) | Seller variation margin (RUB) |
| fee\_ex\_b | numeric(16,2) | Exchange fee of the buyer (RUB) |
| vat\_ex\_b | numeric(16,2) | VAT in the Exchange fee (RUB) – not used |
| fee\_cc\_b | numeric(16,2) | Fee charged by the buyer’s clearing house (RUB) |
| vat\_cc\_b | numeric(16,2) | VAT in the buyer’s clearing fee (RUB)– not used |
| fee\_ex\_s | numeric(16,2) | Seller’s exchange fee (RUB) |
| vat\_ex\_s | numeric(16,2) | VAT in the seller’s Exchange fee (RUB) – not used |
| fee\_cc\_s | numeric(16,2) | Fee charged by the seller’s clearing house (RUB) |
| vat\_cc\_s | numeric(16,2) | VAT in the seller’s clearing fee (RUB) – not used |
| signs | numeric(11) | The bitmask of trade’s attributes. It is the full analogue of the bit mask of the trade attributes in the Plaza-2 gateway’s trade table (please visit ftp://ftp.moex.com/pub/FORTS/Plaza2/docs/, p2gate\_ru.pdf). |

Notes:

1. “id\_deal” corresponds with the trade ID assigned during the gateway procedures.
2. “o04\_XXYY.isin” = “o07.contract”.
3. In general, price is measured in points. In specific cases, it is expressed in RUB. Specific formula is applied to transfer price into roubles: price(in RUB) = round(price \* o07.tick\_price / o07.tick, 2).
4. This file contains all trades of BF’s clients ‘XXYY’. If “kod\_sell” <> ‘’, “kod\_sell” is BF’s client ‘XXYY’, and not client otherwise. If “kod\_buy” <> ‘’, “kod\_buy” is BF’s client ‘XXYY’, and not client otherwise. If “kod\_sell” = ‘’, “no\_sell” = 0, “fee\_sell” = 0, “comm.\_sell” = ‘’, “ext\_id\_s” = 0. If “kod\_buy” = ‘’, “no\_buy” = 0, “fee\_buy” = 0, “comm.\_buy” = ‘’, “ext\_id\_b” = 0.
5. If “type” = 3, “id\_deal” = 0; “user\_buy” = ’’; “user\_sell” = ’’; “no\_buy” = 0; “no\_sell” = 0. If “type” = 1, “id\_deal” = 0, if the settlement trade was made at clearing, or “id\_deal” <> 0, if the settlement trade was made at trading; “user\_buy” = ’’; “user\_sell” = ’’; “no\_buy” = 0; “no\_sell” = 0; “price” = 0.
6. For options other than futures-style the buyer premium for the trade is “prem\_buy” = -vol \* round(price \* o07.tick\_price / o07.tick, 2), the seller premium for the trade “prem\_sell” = vol \* round(price \* o07.tick\_price / o07.tick, 2). For futures-style options the field value is nul.
7. “no\_buy” and “no\_sell” are IDs of the orders resulted in the trade. Such IDs correspond with IDs assigned during the gateway procedures.
8. The field “date2” is of similar import to the field “date”. The fields differ only in formats.
9. For all BF’s trades in the instrument “isin”: ∑(“fee\_buy” + “fee\_sell”) = “oposXXYY.sbor”, where “oposXXYY.date” = “date2” and “oposXXYY.isin” = “isin” and “oposXXYY.kod” = ‘XXYY’ + ‘000’ and “oposXXYY.account” = ‘BF’.
10. For all trades of client ‘XXYYYYY’ in instrument “isin”: ∑(“fee\_buy” + “fee\_sell”) = “oposXXYY.sbor”, where “oposXXYY.date” = “date2” and “oposXXYY.isin” = “isin” and “oposXXYY.kod” = ‘XXYYYYY’ and “oposXXYY.account” = ‘CL’.
11. For all BF’s trades in all instruments: ∑(“fee\_buy” + “fee\_sell”) = “monXXYY.opt\_sbor”, where “monXXYY.date” = “date2” and “monXXYY.kod” = ‘XXYY’ + ‘000’ and “monXXYY.account” = ‘BF’ and “monXXYY.type” = ‘MN’.
12. For all trades of client ‘XXYYYYY’ in all instruments: ∑(“fee\_buy” + “fee\_sell”) = “monXXYY.opt\_sbor”, where “monXXYY.date” = “date2” and “monXXYY.kod” = ‘XXYYYYY’ and “monXXYY.account” = ‘CL’ and “monXXYY.type” = ‘MN’.
13. If the counterparty to the trade acts on behalf of the trust manager, the fields “du\_buy” (or “du\_sell”) = 1 regardless of whether the counterparty is the client of the BF ‘XXYY’. For example, client XX00001 bought five contracts. The counterparty to the trade acts on behalf of the asset manager. In this case “kod\_buy” = ‘XX00001’, “kod\_sell” = ‘’, and “du\_sell” = 1.
14. For all BF’s trades in the instrument “isin”: ∑(“fee\_ns\_b” + “fee\_ns\_s”) = “oposXXYY.sbor\_nosys”, where “oposXXYY.date” = “date2” and “oposXXYY.isin” = “isin” and “oposXXYY.kod” = ‘XXYY’ + ‘000’ and “oposXXYY.account” = ‘BF’.
15. For all trades of client ‘XXYYYYY’ in instrument “isin”: ∑(“fee\_ns\_b” + “fee\_ns\_s”) = “oposXXYY.sbor\_nosys”, where “oposXXYY.date” = “date2” and “oposXXYY.isin” = “isin” and “oposXXYY.kod” = ‘XXYYYYY’ and “oposXXYY.account” = ‘CL’.
16. For options other than futures-style and all BF’s trades in the instrument “isin”: ∑(“prem\_buy” + “prem\_sell”) = “oposXXYY.pem”, where “oposXXYY.date” = “date2” and “oposXXYY.isin” = “isin” and “oposXXYY.kod” = ‘XXYY’ + ‘000’ and “oposXXYY.account” = ‘BF’.
17. For options other than futures-style and all trades of client ‘XXYYYYY’ in instrument “isin”: ∑(“prem\_buy” + “prem\_sell”) = “oposXXYY.prem”, where “oposXXYY.date” = “date2” and “oposXXYY.isin” = “isin” and “oposXXYY.kod” = ‘XXYYYYY’ and “oposXXYY.account” = ‘CL’.
18. For options other than futures-style and all BF’s trades in all instruments: ∑(“prem\_buy” + “prem\_sell”) = “monXXYY.prem”, where “monXXYY.date” = “date2” and “monXXYY.kod” = ‘XXYY’ + ‘000’ and “monXXYY.account” = ‘BF’ and “monXXYY.type” = ‘MN’.
19. For options other than futures-style and all trades of client ‘XXYYYYY’ in all instruments: ∑(“prem\_buy” + “prem\_sell”) = “monXXYY.prem”, where “monXXYY.date” = “date2” and “monXXYY.kod” = ‘XXYYYYY’ and “monXXYY.account” = ‘CL’ and “monXXYY.type” = ‘MN’.
20. The fields “var\_marg\_b” and “var\_marg\_s” are nul for options other than futures-style.
21. File o04clXXYYZZZ.csv. Information on client trades in options (XXYYZZZ is the section code). The file is in CSV format.

Structure of the file **o04clXXYYZZZ.csv**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| id\_deal | numeric(10) | Trade ID in the trading system  |
| isin | char(25) | Unique code of the instrument (options contract) |
| price | numeric(16,5) | Trade price (points) |
| vol | numeric(10) | Number of instruments in the trade (units) |
| kod\_sell | char(7) | Seller code |
| kod\_buy | char(7) | Buyer code |
| date | char(10) | Trade date in YYYY/MM/DD format |
| time | char(8) | Trade time |
| profit\_usd | numeric(20,4) | not used |
| type | numeric(2) | Trade type:0 – anonymous trade,1 – options expiration,2 – negotiated trade/position transfer between section with the same identity details with the bit mask attribute of 0х8 шт the gateway and a unique non-zero id in the field “signs”,3 – negotiated trade executed during the clearing session, not used, 4 – resulted from the unsettled options expiry. |
| user\_buy | char(20) | Buyer’s login in the trading system |
| user\_sell | char(20) | Seller’s login in the trading system |
| no\_buy | numeric(15) | No. of the buyer’s order in the trading system  |
| no\_sell | numeric(15) | No. of the seller’s order in the trading system  |
| fee\_buy | numeric(16,2) | Sum of the Exchange fee and the buyer’s clearing center fee (RUB) |
| fee\_sell | numeric(16,2) | Sum of the Exchange fee and the seller’s clearing center fee (RUB) |
| date2 | date | Trade date |
| comm\_buy | char(20) | Comments from the buyer  |
| comm\_sell | char(20) | Comments from the seller |
| du\_buy | numeric(1) | Attribute of the seller: 1 - the seller is the trust manager, 0 - the seller is not the trust manager |
| du\_sell | numeric(1) | Attribute of the buyer: 1 - the buyer is the trust manager, 0 - the buyer is not the trust manager |
| fee\_ns\_b | numeric(16, 2) | Exchange and clearing fees for negotiated trades by the buyer (RUB) |
| fee\_ns\_s | numeric(16, 2) | Exchange and clearing fees for negotiated trades by the seller (RUB) |
| prem\_buy | numeric(16, 2) | Buyer premium (RUB)  |
| prem\_sell | numeric(16, 2) | Seller premium (RUB)  |
| price\_rur | numeric(16, 5) | Trade price in RUB |
| ext\_id\_b | numeric(11) | Buyer external ID |
| ext\_id\_s | numeric(11) | Seller external ID |
| date\_clr | date | Clearing date |
| var\_marg\_b | numeric(16,5) | Buyer variation margin (RUB) |
| var\_marg\_s | numeric(16,5) | Seller variation margin (RUB) |
| fee\_ex\_b | numeric(16,2) | Exchange fee of the buyer (RUB) |
| vat\_ex\_b | numeric(16,2) | VAT in the Exchange fee (RUB) – not used |
| fee\_cc\_b | numeric(16,2) | Fee charged by the buyer’s clearing house (RUB) |
| vat\_cc\_b | numeric(16,2) | VAT in the buyer’s clearing fee (RUB)– not used |
| fee\_ex\_s | numeric(16,2) | Seller’s exchange fee (RUB) |
| vat\_ex\_s | numeric(16,2) | VAT in the seller’s Exchange fee (RUB) – not used |
| fee\_cc\_s | numeric(16,2) | Fee charged by the seller’s clearing house (RUB) |
| vat\_cc\_s | numeric(16,2) | VAT in the seller’s clearing fee (RUB) – not used |
| signs | numeric(11) | The bitmask of trade’s attributes. It is the full analogue of the bit mask of the trade attributes in the Plaza-2 gateway’s trade table (please visit ftp://ftp.moex.com/pub/FORTS/Plaza2/docs/, p2gate\_ru.pdf). |
| counterparty | char(7) | Code of the counterparty to the trade (code of the clearing member or brokerage firm of the clearing member) executed on the basis of the non-anonymous order |

Notes:

1. The file is produced if the reporting service was ordered for the clearing registers section.

1. File f07.csv. Trading results for futures. The file is in CSV format.

Structure of the file **f07.csv**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | char(10) | Trading day in the format YYYY/MM/DD |
| contract | char(25) | Unique code of the instrument (futures contract, multi-leg instrument or risk management instrument) |
| execution | char(10) | The instrument last trading day (settlement date) in the format YYYY/MM/DD |
| volume | numeric(10) | Turnover in contracts |
| vol\_rubl | numeric(17,2) | Turnover in RUB |
| low | numeric(16,5) | Minimum trade price (points) |
| high | numeric(16,5) | Maximum trade price (points) |
| open | numeric(16,5) | Opening price (points) |
| close | numeric(16,5) | Closing price (points) |
| settl | numeric(16,5) | Settlement price (points) |
| trades | numeric(10) | Number of trades |
| interest | numeric(10) | Number of positions |
| fee | numeric(16,5) | Sum of the exchange fee and clearing fee for one futures contract (in RUB) |
| tick\_price | numeric(16,5) | Price tick value (RUB) |
| tick | numeric(16,5) | Price tick (points) |
| avrg | numeric(16,5) | Average weighted price (points) |
| poses\_rubl | numeric(17,2) | Open interest (RUB) |
| limit | numeric(16,5) | Price fluctuation threshold for the nearby futures (points) |
| kof | numeric(10,6) | Coefficient showing the relation between the futures settlement date and IM base rate  |
| risk\_wr | numeric(16,5) | Additional risk associated with client selling positions in the options on a given futures (RUB) |
| coffout | numeric(7,5) | The limit coefficient for options on a given futures |
| base\_fut | char(25) | Underlying asset code |
| is\_spread | numeric(1) | An attribute showing if the futures contract is included in the intermonth spread. “1” means the futures is in the spread, “0” means it is not. |
| name | char(25) | Instrument short code (futures contract) |
| date2 | Date | Trading day |
| execution2 | date | Instrument last trading day (settlement date) |
| deposit | numeric(16,5) | Base IM for one position (RUB) |
| is\_percent | numeric(1) | Futures contract attribute:0 – not valued in per cent,1 – valued in per cent,2 – electric power contract,3 – Eurobond contract,4 – RUONIA rate contract. |
| perc\_rate | numeric(7,2) | Variation margin interest rate for interest futures  |
| settl\_rur | numeric(16,5) | Settlement price in RUB |
| lot\_volume | numeric(10) | Number of underlying asset units in one contract (lot) |
| tick\_pr\_go | numeric(16,5) | Price tick value in RUB for calculation of IM (RUB) |
| limit\_l1 | numeric(16,5) | Limit L1. The wide limit (points) |
| pr\_setll | numeric(16,5) | Settlement price of the intraday clearing session (points) |
| pr\_settl\_r | numeric(16,5) | Settlement price of the intraday clearing session (RUB) |
| type\_exec | numeric(1) | Settlement type: 0 – cash-settled; 1 - deliverable  |
| section | char(50) | Market section on which the instrument is traded |
| spot | char(50) | Attribute of spot instruments; not used |
| base | char(50) | Underlying contract of the spot instrument; not used |
| type\_sbor | char(50) | Calculation method of the exchange fee indicated in the fields «fee» and «nc\_fee»: RUR – in RUB per contract; PERCENT – per cent of the trade price |
| ns\_volume  | numeric(10) | Volume of negotiated trades in contracts |
| ns\_trades  | numeric(10) | Number of negotiated trades |
| ns\_fee | numeric(16,5) | Negotiated trade fee |
| ns\_volrubl | numeric(16,5) | Volume of negotiated trades in RUB |
| l\_tradeday | date | Instrument last trading day |
| multileg | numeric(1) | Trade attribute: 0 – regular trade, 1- multi-leg trade |

Notes:

1. The key fields are “date” (or “date2”) and “contract”.
2. The base futures contract (“base\_fut”) includes all futures on such base contract.
3. The field “date2” is similar in sense to the field “date”. The fields differ only in the format.
4. The field “execution2” is similar in sense to the field “execution”. The fields differ only in the format.
5. The field “settl\_rur” = ‘0‘ for multi-leg trades.
6. Calendar spread trades are not included in low, high, avrg.
7. File o07.csv. Trading results for options. The file is in CSV format.

Structure of the file **o07.csv**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | char(10) | Trading day in the format YYYY/MM/DD |
| contract | char(25) | Unique code of the instrument (options contract) |
| execution | char(10) | Instrument last trading day (expiry date) in the format YYYY/MM/DD |
| volume | numeric(10) | Turnover in contracts |
| vol\_rubl | numeric(16,2) | Turnover in RUB |
| low | numeric(16,5) | Minimum trade price (points) |
| high | numeric(16,5) | Maximum trade price (points) |
| open | numeric(16,5) | Opening price (points) |
| close | numeric(16,5) | Closing price (points) |
| avrg | numeric(16,5) | Average weighted price (points) |
| trades | numeric(10) | Number of trades (points) |
| interest | numeric(10) | Number of positions |
| fee | numeric(16,5) | Sum of the exchange and clearing fees (RUB) |
| tick\_price | numeric(16,5) | Price tick value (RUB)  |
| tick | numeric(16,5) | Price tick (points) |
| poses\_rubl | numeric(17,2) | Open interest (RUB) |
| depo\_uncov | numeric(16,5) | IM base rate for one unsecured position (RUB) |
| depo\_cov | numeric(16,5) | IM base rate for one synthetic position (RUB) |
| fut\_contr | char(25) | Code of the futures contract |
| strike | numeric(16,5) | Option strike price (points) |
| put | char(1) | Option type: PUT(‘P’) or CALL(‘C’) |
| evrop | char(1) | Option style: european (‘E’) or american (‘A’) |
| date2 | date | Trading day |
| execution2 | date | Instrument last trading day (expiry date) |
| name | char(25) | Instrument attribute (for options contract) |
| close\_time | char(8) | Last trade time |
| volat | numeric(16, 5) | Volatility |
| theorprice | numeric(16, 5) | Theoritical price (points). |
| tick\_pr\_go | numeric(16, 5) | Price tick value in RUB for calculation of IM (RUB) |
| pr\_volat | numeric(16, 5) | Volatility at the intraday clearing |
| pr\_theorpr | numeric(16, 5) | Theoritical price of the intraday clearing session (points) |
| fut\_type | char(1) | An attribute indicating if the options contract is futures-style: : «0» – option is not futures-style; «1» – option is futures-style;  |
| basegobuy | numeric(16,2) | IM base rate for buyers of futures-style options |

Notes:

1. The key fields are “date” (or “date2”) and “contract”.
2. The field “date2” is similar in sense to the field “date”. The fields differ only in the format.
3. The field “execution2” is similar in sense to the field “execution”. The fields differ only in the format.
4. File dayf07.csv. Trading results for futures after the interim clearing session. The file is in CSV format.

Structure of the file **dayf07.csv**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | char(10) | Trading day in the format YYYY/MM/DD |
| contract | char(25) | Unique code of the instrument (a futures contract, multi-leg or risk management instrument) |
| execution | char(10) | The instrument last trading day (settlement date) in the format YYYY/MM/DD |
| volume | numeric(10) | Turnover in contracts |
| vol\_rubl | numeric(17,2) | Turnover in RUB |
| low | numeric(16,5) | Minimum trade price (points) |
| high | numeric(16,5) | Maximum trade price (points) |
| open | numeric(16,5) | Opening price (points) |
| close | numeric(16,5) | Closing price (points) |
| settl | numeric(16,5) | Settlement price (points) |
| trades | numeric(10) | Number of trades |
| interest | numeric(10) | Number of positions |
| fee | numeric(16,5) | Sum of the exchange and clearing fees for one futures contract (in RUB) |
| tick\_price | numeric(16,5) | Price tick value (RUB) |
| tick | numeric(16,5) | Price tick (points) |
| avrg | numeric(16,5) | Average weighted price (points) |
| poses\_rubl | numeric(17,2) | Open interest (RUB) |
| limit | numeric(16,5) | Price fluctuation threshold for the nearby futures (points) |
| kof | numeric(10,6) | Coefficient showing the relation between the futures settlement date and IM base rate  |
| risk\_wr | numeric(16,5) | Additional risk associated with client selling positions in the options on a given futures (RUB) |
| coffout | numeric(7,5) | The limit coefficient for options on a given futures |
| base\_fut | char(25) | Underlying asset code |
| is\_spread | numeric(1) | An attribute showing if the futures is included in the intermonth spread. 1 means the futures is in the spread, 0 means it is not. |
| name | char(25) | Instrument short code (futures contract) |
| date2 | Date | Trading day |
| execution2 | date | Instrument last trading day (settlement date) |
| deposit | numeric(16,5) | Base IM for one position (RUB) |
| is\_percent | numeric(1) | Futures contract attribute:0 – not valued in per cent,1 – valued in per cent,2 – electric power contract,3 – Eurobond contract,4 – RUONIA rate contract. |
| perc\_rate | numeric(7,2) | Interest rate to ccalculate variation margin for interest futures. |
| settl\_rur | numeric(16, 5) | Settlement price in RUB |
| lot\_volume | numeric(10) | Number of underlying asset units in one contract (lot) |
| tick\_pr\_go | numeric(16,5) | Price tick value in RUB for calculation of IM (RUB) |
| limit\_l1 | numeric(16,5) | Limit L1. The wide limit (points); not used |
| pr\_setll | numeric(16, 5) | Settlement price of the intraday clearing session (points) |
| pr\_settl\_r | numeric(16, 5) | Settlement price of the intraday clearing session (RUB) |
| type\_exec | numeric(1) | Settlement type: 0 – cash-settled; 1 - deliverable |
| section | char(50) | Market section on which the instrument is traded |
| spot | char(50) | Attribute of spot instruments; not used |
| base | char(50) | Underlying contract of the spot instrument; not used |
| type\_sbor | char(50) | Calculation method of the exchange fee indicated in the fields «fee» and «nc\_fee»: RUR – in RUB per contract; PERCENT – per cent of the trade price |
| ns\_volume  | numeric(10) | Volume of negotiated trades in contracts |
| ns\_trades  | numeric(10) | Number of negotiated trades |
| ns\_fee | numeric(16,5) | Negotiated trade fee |
| ns\_volrubl | numeric(16,5) | Volume of negotiated trades in RUB |
| l\_tradeday | date | Instrument last trading day |
| multileg | numeric(1) | Trade attribute:: 0 – regular trade, 1- multi-leg trade |

Notes:

* + - 1. The key fields are “date” (or “date2”) and “contract”.
			2. The base futures contract includes all futures on such base contract.
			3. The field “date2” is similar in sense to the field “date”. The fields differ only in the format.
			4. The field “execution2” is similar in sense to the field “execution”. The fields differ only in the format.
			5. The field “settl\_rur” = ‘0‘ for multi-leg trades.
			6. Calendar spread trades are not included in low, high, avrg.
1. File dayo07.csv. Trading results for options after the interim clearing session. The file is in CSV format.

Structure of the file **dayo07.csv**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | char(10) | Trading day in the format YYYY/MM/DD |
| contract | char(25) | Unique code of the instrument (options contract) |
| execution | char(10) | Instrument last trading day (expiry date) in the format YYYY/MM/DD |
| volume | numeric(10) | Turnover in contracts |
| vol\_rubl | numeric(17,2) | Turnover in RUB |
| low | numeric(16,5) | Minimum trade price (points) |
| high | numeric(16,5) | Maximum trade price (points) |
| open | numeric(16,5) | Opening price (points) |
| close | numeric(16,5) | Closing price (points) |
| avrg | numeric(16,5) | Average weighted price (points) |
| trades | numeric(10) | Number of trades (points) |
| interest | numeric(10) | Number of positions |
| fee | numeric(16,5) | Sum of the exchange and clearing fees (RUB) |
| tick\_price | numeric(16,5) | Price tick value (RUB)  |
| tick | numeric(16,5) | Price tick (points) |
| poses\_rubl | numeric(17,2) | Open interest (RUB) |
| depo\_uncov | numeric(16,5) | IM base rate for one unsecured position (RUB) |
| depo\_cov | numeric(16,5) | IM base rate for one synthetic position (RUB) |
| fut\_contr | char(25) | Code of the futures contract |
| strike | numeric(16,5) | Option strike price (points) |
| put | char(1) | Option type: PUT(‘P’) or CALL(‘C’) |
| evrop | char(1) | Option style: european (‘E’) or american (‘A’) |
| date2 | date | Trading day |
| execution2 | date | Instrument last trading day (expiry date) |
| name | char(25) | Instrument attribute (option contract) |
| close\_time | char(8) | Last trade time |
| volat | numeric(16, 5) | Volatility. |
| theorprice | numeric(16, 5) | Theoritical price (points) |
| tick\_pr\_go | numeric(16, 5) | Price tick value in RUB for calculation of IM (RUB) |
| pr\_volat | numeric(16, 5) | Volatility at the intraday clearing |
| pr\_theorpr | numeric(16, 5) | Theoritical price of the intraday clearing session (points) |
| fut\_type | char(1) | An attribute indicating if the options contract is futures-style: «0» – the option is not futures-style; «1» – the option is futures-style |
| basegobuy | numeric(16,2) | IM base rate for buyers of futures-style options |

Notes:

* + - 1. The key fields are “date” (or “date2”) and “contract”.
			2. The field “date2” is similar in sense to the field “date”. The fields differ only in the format.
			3. The field “execution2” is similar in sense to the field “execution”. The fields differ only in the format.
1. File fposXXYY.csv. Futures positions balances of the clearing member (CM)/brokerage firm (BF) and their clients. The file is in CSV format.

Structure of the file **fposXXYY.csv**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | char(10) | Clearing date |
| kod | char(7) | CM’s/BF’s/client’s code |
| account | char(2) | Flag of CM/BF/client (‘RF’ - CM; ‘BF’ - BF; ‘CL’ – client) |
| isin | char(25) | Unique code of the instrument (futures contract)  |
| pos\_beg | numeric(11) | Number of positions as at the beginning of the day |
| pos\_end | numeric(11) | Number of positions as at the end of the day |
| var\_marg\_p | numeric(16,2) | Variation margin for open interest at the beginning of the day(RUB) |
| var\_marg\_d | numeric(16,2) | Variation margin for trades (RUB) |
| sbor | numeric(16,2) | Sum of the exchange and clearing fees (RUB) |
| go\_netto | numeric(16,2) | Not used |
| go\_brutto | numeric(16,2) | Gross IM (RUB) |
| pos\_exec | numeric(11) | Number of settled positions (a positive/negative value means settlement of selling/buying positions) |
| du | numeric(1) | An attribute of trust accounts (1 - trust account, 0 - not trust account) |
| sbor\_exec | numeric(16,2) | Settlement fee of the Clearing Center (RUB) |
| sbor\_nosys | numeric(16,2) | Sum of the exchange and clearing fees for negotiated trades (RUB). |
| fee\_exec | numeric(16,2) | Penalties for late settlement of futures contracts |
| fine\_exec | numeric(16,2) | Default fine for futures contracts  |
| accum\_go | numeric(16,2) | Collateral deposit for the spot instrument accumulated over the position lifetime; not used |
| fee\_trans | numeric(16,2) | Penalty for positions rollovers through repo trades or buying/selling trades for trust accounts |
| sbor\_ex | numeric(16,2) | Exchange fee (RUB) |
| vat\_ex | numeric(16,2) | VAT in the Exchange fee (RUB) – Not used |
| sbor\_сс | numeric(16,2) | Clearing fee (RUB) |
| vat\_cc | numeric(16,2) | VAT in the Clearing fee (RUB) – Not used |
| pos\_failed | numeric(11) | Number of defaulting positions  |

Notes:

1. The key fields are “date”, “kod”, “account”, and “isin”.
2. Total variation margin is calculated by (“var\_marg\_p” + “var\_marg\_d”).
3. The fields “pos\_beg” and “pos\_end” show net postion of the clearing member and its clients including brokerage firms and their clients.
4. The fields “pos\_beg” and “pos\_end” show net position of the brokerage firm and its clients.
5. The fields “var\_marg\_p”, “var\_marg\_d”, “sbor”, “sbor\_exec”, and “sbor\_nosy” are the sum of relevant fields for the CM and its clients including BFs and their clients.
6. The fields “var\_marg\_p”, “var\_marg\_d”, “sbor”, “sbor\_exec”, and “sbor\_nosys” are the sum of relevant fields for the BF and its clients.
7. The field “go\_brutto” is the sum of the same fields for all CM’s BFs.
8. The field “go\_brutto” for a BF is the gross IM of all BF’s clients.
9. If “pos\_beg” <> 0 on the settlement day, “pos\_end” = 0 and “pos\_exec” = - “pos\_beg”.
10. The following rule is applied to spot instruments: “pos\_exec” + “value of negotiated trades executed during the T0 after-hours trading session” = - “pos\_beg”.
11. The following items are considered in the field “pos\_failed”:

OFZ futures contract. Number of failed positions (determined by absence of trades and incorrect orders in the order register) are considered on the first and second settlement days. A default penalty is charged.

1. File fposclXXYYZZZ.csv. Client positions balances in futures (with XXYYZZZ designating the section code). The file is in CSV format.

Structure of the file **fposclXXYYZZZ.csv**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | char(10) | Clearing date |
| kod | char(7) | Client ID |
| account | char(2) | Client attribute (‘CL’ – client) |
| isin | char(25) | Unique code of the instrument (futures contract) |
| pos\_beg | numeric(11) | Number of positions as at the beginning of the day |
| pos\_end | numeric(11) | Number of positions as at the end of the day |
| var\_marg\_p | numeric(16,2) | Variation margin for open interest at the beginning of the day(RUB) |
| var\_marg\_d | numeric(16,2) | Variation margin for trades (RUB) |
| sbor | numeric(16,2) | Sum of the exchange and clearing fees (RUB) |
| go\_netto | numeric(16,2) | Not used |
| go\_brutto | numeric(16,2) | Gross IM (RUB) |
| pos\_exec | numeric(11) | Number of settled positions (a positive/negative value means settlement of selling/buying positions) |
| du | numeric(1) | An attribute of trust accounts (1 - trust account, 0 - not trust account) |
| sbor\_exec | numeric(16,2) | Settlement fee of the Clearing Center (RUB). |
| sbor\_nosys | numeric(16,2) | Sum of the exchange and clearing fees for negotiated trades (RUB) |
| fee\_exec | numeric(16,2) | Penalties for late settlement of futures contracts |
| fine\_exec | numeric(16,2) | Default fine for futures contracts  |
| accum\_go | numeric(16,2) | Collateral deposit for the spot instrument accumulated over the position lifetime; not used |
| fee\_trans | numeric(16,2) | Penalty for positions rollovers through repo trades or buying/selling trades for trust accounts |
| sbor\_ex | numeric(16,2) | Exchange fee (RUB) |
| vat\_ex | numeric(16,2) | VAT in the Exchange fee (RUB) – Not used |
| sbor\_сс | numeric(16,2) | Clearing fee (RUB) |
| vat\_cc | numeric(16,2) | VAT in the Clearing fee (RUB) – Not used |
| pos\_failed | numeric(11) | Number of defaulting positions |

Notes:

1. The key fields are “date”, “kod”, “account”, and “isin”.
2. Total variation margin is calculated by (“var\_marg\_p” + “var\_marg\_d”).
3. If “pos\_beg” <> 0 on the settlement day, “pos\_end” = 0, and “pos\_exec” = - “pos\_beg”.
4. The file is produced if the reporting service was ordered for the clearing registers section.
5. The following values are indicated for the field “pos\_failed”:

OFZ futures contract. Number of failed positions (determined by absence of trades and incorrect orders in the order register) are considered on the first and second settlement days. A default penalty is charged.

1. File oposXXYY.csv. Options positions balances of the clearing member (CM)/brokerage firm (BF) and their clients. The file is in CSV format.

Structure of the file **oposXXYY.csv**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | Char(10) | Clearing date |
| kod | char(7) | CM’s/BF’s/client’s code |
| account | char(2) | Flag of CM/BF/client (‘RF’ - CM; ‘BF’ - BF; ‘CL’ – client) |
| isin | char(25) | Options contract |
| pos\_beg | numeric(11) | Number of positions as at the beginning of the day |
| pos\_end | numeric(11) | Number of positions as at the end of the day |
| prem | numeric(16,2) | Trade premium (RUB) |
| sbor | numeric(16,2) | Sum of the exchange and clearing fees (RUB) |
| go | numeric(16,2) | Gross IM (RUB) |
| pos\_exec | numeric(11) | Number of expired positions (a positive/negative value indicates expiry of selling/buying positions ) |
| pos\_endcir | numeric(11) | Number of positions deleted following the expiry of the option contract (a positive/negative value indicates expiry of put/call option ) |
| du | numeric(1) | An attribute of trust accounts (1 - trust account, 0 - not trust account) |
| sbor\_exec | numeric(16,2) | Settlement fee of the Clearing Center (RUB). |
| sbor\_nosys | numeric(16,2) | Sum of the exchange and clearing fees for negotiated trades (RUB). |
| var\_marg\_p | numeric(16,2) | Variation margin per open interest as at the beginning of the day (RUB) |
| var\_marg\_d | numeric(16,2) | Variation margin per trades (RUB) |
| sbor\_ex | numeric(16,2) | Exchange fee (RUB) |
| vat\_ex | numeric(16,2) | VAT in the Exchange fee (RUB) – Not used |
| sbor\_сс | numeric(16,2) | Clearing fee (RUB) |
| vat\_cc | numeric(16,2) | VAT in the Clearing fee (RUB) – Not used |

Notes:

1. The key fields are “date”, “kod”, “account”, and “isin”.
2. The fields “pos\_beg” and “pos\_end” show net postion of the clearing member and its clients including brokerage firms and their clients.
3. The fields “pos\_beg” and “pos\_end” show net position of the brokerage firm and its clients.
4. The fields “prem”, “sbor”, “pos\_exec”, “pos\_endcir”, “sbor\_exec”, and “sbor\_nosys” are the sum of relevant fields for the CM and its clients including BFs and their clients.
5. Fields “prem”, “sbor”, “pos\_exec”, “pos\_endcir”, “sbor\_exec”, and “sbor\_nosys” are the sum of relevant fields for the BF and its clients.
6. The field “prem” =0 for futures-style options.
7. The fields “var\_marg\_p” and “var\_marg\_d” are nul for options other than futures-style.
8. File oposclXXYYZZZ.csv. Information on clients’ options positions (with XXYYZZZ designating the section code). The file is in CSV format.

Structure of the file **oposclXXYY.csv**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | Char(10) | Clearing date |
| kod | char(7) | Client ID |
| account | char(2) | Client attribute ('CL' - client) |
| isin | char(25) | Options contract |
| pos\_beg | numeric(11) | Number of positions as at the beginning of the day |
| pos\_end | numeric(11) | Number of positions as at the end of the day |
| prem | numeric(16,2) | Trade premium (RUB) |
| sbor | numeric(16,2) | Sum of the exchange and clearing fees (RUB) |
| go | numeric(16,2) | Gross IM (RUB) |
| pos\_exec | numeric(11) | Number of expired positions (a positive/negative value indicates expiry of selling/buying positions ) |
| pos\_endcir | numeric(11) | Number of positions deleted following the expiry of the option contract (a positive/negative value indicates expiry of put/call option ) |
| du | numeric(1) | An attribute of trust accounts (1 - trust account, 0 - not trust account) |
| sbor\_exec | numeric(16,2) | Settlement fee of the Clearing Center (RUB) |
| sbor\_nosys | numeric(16,2) | Sum of the exchange and clearing fees for negotiated trades (RUB). |
| var\_marg\_p | numeric(16,2) | Variation margin per open interest as at the beginning of the day (RUB) |
| var\_marg\_d | numeric(16,2) | Variation margin per trades (RUB) |
| sbor\_ex | numeric(16,2) | Exchange fee (RUB) |
| vat\_ex | numeric(16,2) | VAT in the Exchange fee (RUB) – Not used |
| sbor\_сс | numeric(16,2) | Clearing fee (RUB) |
| vat\_cc | numeric(16,2) | VAT in the Clearing fee (RUB) – Not used |

Notes:

1. The key fields are “date”, “kod”, “account”, and “isin”.
2. The field “prem”=0 for futures-style options.
3. The fields “var\_marg\_p” and “var\_marg\_d” are nul for options other than futures-style.
4. The file is produced if the reporting service was ordered for the clearing registers section.
5. File monXXYY.csv. Information on cash position of the clearing member (CM)/brokerage firm (BF) and their clients. The file is in CSV format.

Structure of the file **monXXYY.csv**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | char(10) | Clearing date |
| kod | char(12) | CM/BF/client/settlement code |
| account | char(2) | Flag of CM/BF/client (‘RF’ - CM; ‘BF’ - BF; ‘CL’ – client; ‘RK’ – settlement code) |
| type | char(2) | Cash account type (‘MN’ – roubles and full collateral assets in foreign currency (in RUB); ‘PL’ – collateral assets in securities (no full prefunding) (in RUB)) |
| amount\_beg | numeric(16, 2) | For type =‘MN’: total assets in RUB and full collateral assets in foreign currency (in RUB) as at the beginning of the trading day/for type =‘PL’: securities collateral (no full prefunding) (in RUB) as at the beginning of the trading day  |
| var\_marg | numeric(16, 2) | Variation margin (RUB) |
| prem | numeric(16, 2) | Option premium (RUB) |
| pay | numeric(16, 2) | Daily сhanges in balances::for type ‘MN’: per assets in RUB and full collateral assets in foreign currency (in RUB); for type ‘PL’: per collateral assets in securities (no full prefunding) (in RUB) |
| fut\_sbor | numeric(16, 2) | Sum of the Exchange and clearing fees for trades in futures (RUB) |
| opt\_sbor | numeric(16, 2) | Sum of the exchange and clearing fees for trades in options (RUB) |
| go | numeric(16, 2) | Total IM for all instruments (RUB) |
| amount\_end | numeric(16, 2) | For type =‘MN’: total assets in RUB and full collateral assets in foreign currency as at the end of the trading day /for type =‘PL’: securities collateral (no full prefunding) as at the end of the trading day  |
| free | numeric(16, 2) | For type ‘MN’: available assets in RUB and full collateral assets in foreign currency (in RUB) as at the end of the trading day /for type ‘PL’: available securities collateral (no full prefunding) as at the end of the trading day  |
| du | numeric(1) | An attribute of trust accounts (1 - trust account, 0 - not trust account) |
| gowide | numeric(16, 2) | Total IM for all instruments, calculated by the wide limit (RUB). Not used |
| freewide | numeric(16, 2) | Available assets (RUB), including IM by the wide limit; not used |
| margincall | char(1) | Margin call attribute. It may take one of three values: ‘A’, ‘B’ or ‘ ’ (the attribute is not set). Not used. |
| sbor\_ex | numeric(16,2) | Exchange fee (RUB) |
| vat\_ex | numeric(16,2) | VAT in the Exchange fee (RUB) – Not used |
| sbor\_cc | numeric(16,2) | Clearing fee (RUB) |
| vat\_cc | numeric(16,2) | VAT in the Clearing fee (RUB) – Not used |
| rub\_beg | numeric(16,2) | Cash in RUB at the beginning of the trading day |
| rub\_pay | numeric(16,2) | Intraday change in cash in RUB  |
| rub\_end | numeric(16,2) | Cash in RUB at the end of the trading day |
| com\_pl\_beg | numeric(16,2) | Estimated value of foreign currency as at the beginning of the trading day (in RUB) |
| com\_pl\_pay | numeric(16,2) | Daily change in the estimated value of the foreign currency (in RUB) |
| com\_pl\_end | numeric(16,2) | Estimated value of foreign currency as at the end of the trading day (in RUB) |
| ext\_rez | numeric(20,2) | Amount in RUB to be blocked for positions in RUONIA contracts. This amount is reserved to cover changes in RUONIA (used to calculated variation margin) published by CBR  |

Notes:

1. The key fields are “date”, “kod”, “account”, and “typе”.
2. The fields kod=‘Расчетный код’ and account=‘RK’ are given in the report “monXX00” at the level of the clearing member.
3. The field “go” shows the sum of relevant fields of all CM’s BFs.
4. The field “go” shows the sum of relevant fields of BF’s clients.
5. The field “free” shows the value of (‘amount\_end’ – ‘go’ – ‘ext\_rez’).
6. The field “prem” shows the total premium for trades in non-futures-style options.
7. The field “var\_marg” shows total variation margin for futures and futures-style options and total amount of collateral deposits.
8. For type ‘PL’ fields “rub\_beg”, “rub\_pay”, “rub\_end”, “com\_pl\_beg”, “com\_pl\_pay”, and “com\_pl\_end” are not filled.
9. File monclXXYYZZZ.csv. Information on clients’ cash positions (with XXYYZZZ designating the section code). The file is in CSV format.

Structure of the file **monclXXYYZZZ.csv**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | char(10) | Clearing date |
| kod | char(12) | Client ID |
| account | char(2) | Client attribute ('CL' - client) |
| type | char(2) | For type =‘MN’: total assets in RUB and full collateral assets in foreign currency (in RUB) as at the beginning of the trading day /for type =‘PL’: securities collateral (no full prefunding) (in RUB) as at the beginning of the trading day  |
| amount\_beg | numeric(16, 2) | For type =‘MN’: total assets in RUB and full collateral assets in foreign currency as at the beginning of the trading day /for type =‘PL’: securities collateral (no full prefunding) as at the beginning of the trading day  |
| var\_marg | numeric(16, 2) | Variation margin (RUB) |
| prem | numeric(16, 2) | Option premium (RUB) |
| pay | numeric(16, 2) | Intraday сhanges in balances:for type ‘MN’: per assets in RUB and full collateral assets in foreign currency (in RUB); for type ‘PL’: per collateral assets in securities (no full prefunding) (in RUB) |
| fut\_sbor | numeric(16, 2) | Exchange fee, (RUB) |
| opt\_sbor | numeric(16, 2) | Sum of the exchange and clearing fees for trades in options (RUB) |
| go | numeric(16, 2) | Total IM for all instruments (RUB) |
| amount\_end | numeric(16, 2) | For type =‘MN’: total assets in RUB and full collateral assets in foreign currency (in RUB) as at the end of the trading day /for type =‘PL’: securities collateral (no full prefunding) (in RUB) as at the end of the trading day  |
| free | numeric(16, 2) | For type ‘MN’: available assets in RUB and full collateral assets in foreign currency (in RUB) as at the end of the trading day /for type ‘PL’: available securities collateral (no full prefunding) as at the end of the trading day  |
| du | numeric(1) | An attribute of trust accounts (1 - trust account, 0 - not trust account) |
| gowide | numeric(16, 2) | Total IM for all instruments, calculated by the wide limit (RUB). Not used |
| freewide | numeric(16, 2) | Available assets (RUB) including IM by the wide limit; not used |
| margincall | char(1) | Margin calls attribute. It may take one of three values: ‘A’, ‘B’ or ‘ ’ (the attribute is not set). Not used |
| sbor\_ex | numeric(16,2) | Exchange fee (RUB) |
| vat\_ex | numeric(16,2) | VAT in the Exchange fee/fee of the settlement organization (RUB) – Not used |
| sbor\_cc | numeric(16,2) | Clearing fee (RUB) |
| vat\_cc | numeric(16,2) | VAT in the Clearing fee (RUB) – Not used |
| rub\_beg | numeric(16,2) | Cash in RUB at the beginning of the trading day |
| rub\_pay | numeric(16,2) | Daily change in cash in RUB  |
| rub\_end | numeric(16,2) | Cash in RUB at the end of the trading day |
| com\_pl\_beg | numeric(16,2) | Estimated value of foreign currency as at the beginning of the trading day (in RUB) |
| com\_pl\_pay | numeric(16,2) | Daily change in the estimated value of the foreign currency (in RUB) |
| com\_pl\_end | numeric(16,2) | Estimated value of foreign currency as at the end of the trading day (in RUB) |
| ext\_rez | numeric(20,2) | Amount in RUB to be blocked for positions in RUONIA contracts. This amount is reserved to cover changes in RUONIA (used to calculated variation margin) published by CBR  |

Notes:

1. The key fields are “date”, “kod”, “account”, and “typе”.
2. The field “prem” shows total premium for trades in non-futures-style options.
3. The field “var\_marg” shows total variation margin for futures and futures-style options, as well as an amount of collateral deposits.
4. The file is produced if the reporting service was ordered for the clearing registers section.
5. For type ‘PL’ fields “rub\_beg”, “rub\_pay”, “rub\_end”, “com\_pl\_beg”,
com\_pl\_pay”, and “com\_pl\_end” are not filled.
6. File f08\_XX00.dbf. Information on trades in futures and spot instruments of the clearing member (CM) including trades of its all brokerage firms (XX is the clearing member code). The file is in FoxPro 2.x format.

Cancelled from 19 April 2010

 Structure of the file **f08\_XX00.dbf**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| id\_deal | numeric(10) | Trade ID in the trading system  |
| isin | char(25) | Unique code of the instrument (a futures contract, spot instrument) |
| price | numeric(16,5) | Trade price (points) |
| vol | numeric(10) | Number of instruments in the trade (units) |
| kod\_sell | char(7) | Seller code |
| kod\_buy | char(7) | Buyer code |
| date | char(10) | Trade date in YYYY/MM/DD format |
| time | char(8) | Trade time |
| profit\_usd | numeric(20,4) | Not used |
| type | numeric(2) | Trade type (0 – order book, 1 – options expiration , 2 – negotiated trade executed during the trading, 3 - off book trade executed during the clearing session, 4 – repo first leg, 5 – repo second leg, 7 – selling/buying between sub-accounts in trust (instead of repo trades),8 - trades to settle deliverable futures contracts, 9 – the first leg of repo trades T+N,10 - the second leg of repo trades T+N. |
| var\_marg\_b | numeric(16,2) | Buyer’s variation margin for the trade (RUB) |
| var\_marg\_s | numeric(16,2) | Seller’s variation margin for the trade (RUB) |
| user\_sell | char(20) | Seller’s login in the trading system. The special code OPERATOR\_REPO (CC’s login) for repo trades and buying/selling across the asset manager’s sections |
| user\_buy | char(20) | Buyer’s login in the trading system. The special code OPERATOR\_REPO (CC’s login) for repo trades and buying/selling across the asset manager’s sections |
| no\_buy | numeric(10) | No. of the buyer’s order in the trading system  |
| no\_sell | numeric(10) | No. of the seller’s order in the trading system  |
| fee\_buy | numeric(16,2) | Sum of the Exchange fee and the buyer’s clearing center fee (RUB) |
| fee\_sell | numeric(16,2) | Sum of the Exchange fee and the seller’s clearing center fee (RUB) |
| date2 | date | Trade date |
| comm\_buy | char(20) | Comments from the buyer  |
| comm\_sell | char(20) | Comments from the seller |
| du\_buy | numeric(1) | Attribute of the seller: 1 - the seller is the trust manager, 0 - the seller is not the trust manager) |
| du\_sell | numeric(1) | Attribute of the buyer: 1 - the buyer is the trust manager, 0 - the buyer is not the trust manager) |
| fee\_ns\_b | numeric(16, 2) | Sum of the exchange and clearing fees for buyer’s negotiated trades (RUB) |
| fee\_ns\_s | numeric(16, 2) | Sum of the exchange and clearing fees for seller’s negotiated trades (RUB) |
| price\_rur | numeric(16, 5) | Trade price in RUB |
| ext\_id\_b | numeric(11) | Buyer external ID |
| ext\_id\_s | numeric(11) | Seller external ID |
| date\_clr | date | Clearing date |
| repo\_id | numeric(11) | ID of the related repo leg  |
| fee\_ex\_b | numeric(16,2) | Exchange fee of the buyer (RUB) |
| vat\_ex\_b | numeric(16,2) | VAT in the Exchange fee (RUB) |
| fee\_cc\_b | numeric(16,2) | Fee charged by the buyer’s clearing house (RUB) |
| vat\_cc\_b | numeric(16,2) | VAT in the buyer’s clearing fee (RUB) |
| fee\_ex\_s | numeric(16,2) | Seller’s exchange fee (RUB) |
| vat\_ex\_s | numeric(16,2) | VAT in the seller’s Exchange fee (RUB) |
| fee\_cc\_s | numeric(16,2) | Fee charged by the seller’s clearing house (RUB) |
| vat\_cc\_s | numeric(16,2) | VAT in the seller’s clearing fee (RUB) |

1. File o08\_XX00.dbf. Information on options trades of the clearing member (CM) including trades of its all brokerage firms (XX is the clearing member code). The file is in FoxPro 2.x format.

Cancelled from 19 April 2010

Structure of the file **o08\_XX00.dbf**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| id\_deal | numeric(10) | Trade ID in the trading system  |
| isin | char(25) | Unique code of the instrument (options contract) |
| price | numeric(16,5) | Trade price (points) |
| vol | numeric(10) | Number of instruments in the trade (units) |
| kod\_sell | char(7) | Seller code |
| kod\_buy | char(7) | Buyer code |
| date | char(10) | Trade date in YYYY/MM/DD format |
| time | char(8) | Trade time |
| profit\_usd | numeric(20,4) | Not used |
| type | numeric(2) | Trade type (0 – order book, 1 – options expiration , 2 – off book trade executed during the trading, 3 - negotiated trade executed during the clearing session, 4 – resulted from the unsettled options expiry). |
| user\_buy | char(20) | Buyer’s login in the trading system |
| user\_sell | char(20) | Seller’s login in the trading system |
| no\_buy | numeric(10) | No. of the buyer’s order in the trading system  |
| no\_sell | numeric(10) | No. of the seller’s order in the trading system  |
| fee\_buy | numeric(16,2) | Sum of the Exchange fee and the buyer’s clearing center fee (RUB) |
| fee\_sell | numeric(16,2) | Sum of the Exchange fee and the seller’s clearing center fee (RUB) |
| date2 | date | Trade date |
| comm\_buy | char(20) | Comments from the buyer  |
| comm\_sell | char(20) | Comments from the seller |
| du\_buy | numeric(1) | Attribute of the seller: 1 - the seller is the trust manager, 0 - the seller is not the trust manager) |
| du\_sell | numeric(1) | Attribute of the buyer: 1 - the buyer is the trust manager, 0 - the buyer is not the trust manager) |
| fee\_ns\_b | numeric(16, 2) | Sum of the exchange and clearing fees for buyer’s negotiated trades (RUB) |
| fee\_ns\_s | numeric(16, 2) | Sum of the exchange and clearing fees for seller’s negotiated trades (RUB) |
| prem\_buy | numeric(16, 2) | Buyer premium (RUB). It is always <= 0 |
| prem\_sell | numeric(16, 2) | Seller premium (RUB). It is always >= 0 |
| price\_rur | numeric(16, 5) | Trade price in RUB |
| ext\_id\_b | numeric(11) | Buyer external ID |
| ext\_id\_s | numeric(11) | Seller external ID |
| date\_clr | date | Clearing date |
| var\_marg\_b | numeric(16,5) | Buyer variation margin(RUB)  |
| var\_marg\_s | numeric(16,5) | Seller variation margin (RUB)  |
| fee\_ex\_b | numeric(16,2) | Exchange fee of the buyer (RUB) |
| vat\_ex\_b | numeric(16,2) | VAT in the Exchange fee (RUB) |
| fee\_cc\_b | numeric(16,2) | Clearing fee for the buyer (RUB) |
| vat\_cc\_b | numeric(16,2) | VAT in the Clearing fee for the buyer (RUB) |
| fee\_ex\_s | numeric(16,2) | Seller’s exchange fee (RUB) |
| vat\_ex\_s | numeric(16,2) | VAT in the seller’s Exchange fee (RUB) |
| fee\_cc\_s | numeric(16,2) | Clearing fee for the seller (RUB) |
| vat\_cc\_s | numeric(16,2) | VAT in the сlearing fee for the seller (RUB) |

1. File fordlogXXYY.dbf. The log fixing entries, matchings and cancels of orders in futures and spot instruments of the clearing member (CM)/brokerage firm (BF) and their clients (with XXYY designating the following: AA00 – data on CM with code AA, AA01 – data on BF with code AA01). The file is in FoxPro 2.x format.

Cancelled from 01 July 2011

Structure of the file **fordlogXXYY.dbf**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| isin | char(25) | Unique code of the instrument (futures or instrument of the Standard market, RTS Money instrument) |
| name | char(25) | Instrument attribute (for futures contract or spot instrument) |
| price | numeric(16,5) | Order price (points) |
| vol | numeric(10) | Number of instruments in the order  |
| rest\_vol | numeric(10) | Number of instruments in the order after matching |
| kod | char(7) | Client ID |
| tip | numeric(1) | Order direction (1 - buy; 2 - sell)  |
| sost | numeric(1) | Operation with the order (1 - enter, 2 - matching, 0 - deletion) |
| date | char(10) | Operation date in the format YYYY/MM/DD |
| time | char(8) | Operation time in the format HH:MM:SS |
| numb\_order | numeric(10) | Order number |
| user | char(20) | Trading system user performing the operation  |
| comment | char(20) | User’s comments |
| gate | numeric(1) | An attribute indicating if the operation was made via the gateway (1 - gateway operation, 0 - non-gateway operation) |
| user\_to | char(20) | Adressee of the order in the trading system. If the field is not empty, the order is considered to be negotiated one  |
| date2 | date | Operation date in the format YYYY/MM/DD |
| du | numeric(1) | An attribute of trust accounts (1 - trust account, 0 - not trust account) |
| price\_rur | numeric(16,5) | Order price in RUB  |
| ext\_id | numeric(11) | External ID  |
| date\_exp | date | Order expiry date |
| n\_order1 | numeric(10) | Order number from entry to rollover to next trading sessions |
| date\_clr | date | Clearing date |
| status | numeric(10) | Not used |
| id | numeric(10) | Not used |

Notes:

1. The field “date2” is similar in sense to the field “date”. The fields differ only in the format.
2. File oordlogXXYY.dbf. The log fixing entries, matchings and cancels of options orders of the clearing member (CM)/brokerage firm (BF) and their clients (with XXYY designating the following: AA00 – data on CM with code AA, AA01 – data on BF with code AA01). The file is in FoxPro 2.x format.

Cancelled from 01 July 2011

Structure of the file **oordlogXXYY.dbf**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| isin | char(25) | Unique code of the instrument (options contract) |
| name | char(25) | Instrument attribute (for options contract) |
| price | numeric(16,5) | Order price (points) |
| vol | numeric(10) | Number of instruments in the order  |
| rest\_vol | numeric(10) | Number of instruments in the order after matching |
| kod | char(7) | Client ID |
| tip | numeric(1) | Order direction (1 - buy; 2 - sell)  |
| sost | numeric(1) | Operation with the order (1 - enter, 2 - matching, 0 - deletion) |
| date | char(10) | Operation date in the format YYYY/MM/DD |
| time | char(8) | Operation time in the format HH:MM:SS |
| numb\_order | numeric(10) | Order number |
| user | char(20) | Trading system user performing the operation  |
| gate | numeric(1) | An attribute indicating if the operation was made via the gateway (1 - gateway operation, 0 - non-gateway operation) |
| date2 | date | Operation date in the format YYYY/MM/DD |
| user\_to | char(20) | Adressee of the order in the trading system. If the field is not empty, the order is considered to be negoatiated one |
| comment | char(20) | User’s comments |
| du | numeric(1) | An attribute of trust accounts (1 - trust account, 0 - not trust account) |
| price\_rur | numeric(16,5) | Order price in RUB  |
| ext\_id | numeric(11) | External ID  |
| date\_exp | date | Order expiry date |
| n\_order1 | numeric(10) | Order number from entry to rollover to next trading sessions |
| date\_clr | date | Clearing date |
| status | numeric(10) | Not used |
| id | numeric(10) | Not used |

Notes:

1. The field “date2” is similar in sense to the field “date”. The fields differ only in the format.
2. File payXXYY.csv. Transactions of the clearing member (CM) and brokerage firm (XX is the clearing member code, XXYY is the brokerage firm code). The file is in CSV format.

Structure of the file **payXXYY.csv**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | char(10) | Clearing date |
| kod | char(7) | Code of the cash register section/securities collateral register  |
| account | char(2) | Flag of CM/BF/client (‘RF’ - CM; ‘BF’ - BF; ‘CL’ – client). ‘CL’ always |
| type | char(2) | Cash account type (‘MN’ – roubles and full collateral assets in foreign currency (in RUB); ‘PL’ – collateral assets in securities (no full prefunding) (in RUB)) |
| id\_pay | numeric(10) | Transaction ID |
| type\_pay | numeric(10) | Transaction type |
| pay | numeric(16, 2) | Amount of payment (RUB) |
| name | char(75) | Transaction (payment) name  |
| comment | char(50) | Number of the transfer order for cash deposits in RUB/foreign currency |
| du | numeric(1) | An attribute of trust accounts (1 - trust account, 0 - not trust account) |
| payer | char(200) | Not used  |
| inn | char(12) | Not used  |
| bik | char(9) | Not used  |
| purpose | char(255) | Payment purpose if cash in RUB/foreign currency has been deposited |

Notes:

1. The key field is “id\_pay”.
2. The field “type\_pay” definitely identifies the field “name”. The payment type is described in the field “name”.
3. The field “pay” may take positive and negative values. If it is negative/positive, assets are debitied/credited.
4. Sum of values of all fields pay with the same values of the fields “date”, “kod”, “account”, and “type” equals the value of the field “pay” in the file “monXXYY” with values of the fields “date”, “kod”, “account”, and “type” similar to those in the file “payXXYY”.
5. File payclXXYYZZZ.csv. Transactions in the client section (XXYYZZZ is the client section). The file is in CSV format.

Structure of the file **payclXXYYZZZ.csv**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | char(10) | Clearing date |
| kod | char(7) | Code of the cash register section/securities collateral register |
| account | char(2) | Constant attribute ‘CL’  |
| type | char(2) | Cash account type (‘MN’ – roubles and full collateral assets in foreign currency (in RUB); ‘PL’ – collateral assets in securities (no full prefunding) (in RUB)) |
| id\_pay | numeric(10) | Transaction ID |
| type\_pay | numeric(10) | Transaction type |
| pay | numeric(16, 2) | Amount of payment (RUB) |
| name | char(75) | Transaction (payment) name |
| comment | char(50) | Number of the transfer order for cash deposits in RUB/foreign currency |
| du | numeric(1) | An attribute of trust accounts (1 - trust account, 0 - not trust account) |
| payer | char(200) | Not used  |
| inn | char(12) | Not used  |
| bik | char(9) | Not used  |
| purpose | char(254) | Payment purpose if cash in RUB/foreign currency has been deposited |

Notes:

1. The file is produced if the reporting service was ordered for the clearing registers section.

1. File fut\_deal.csv. Information on all on-exchange trades in futures. The file is in CSV format.

Structure of the file **fut\_deal.csv**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | char(10) | Date |
| time | char(8) | Time in the format HH:MM:SS |
| isin | char(25) | Instrument |
| price | numeric(16, 5) | Price (points) |
| vol | numeric(10) | Number of contracts |
| id\_deal | numeric(10) | Trade ID in the trading system |
| type | numeric(2) | Trade type:0 – anonymous trade,1 – options expiration ,2 – non-anonymous trade/position tramfer between sections with the same identity details,3 – off book trade executed during the clearing session, not used,4 – repo first leg, not used,5 – repo second leg, not used, 7 – selling/buying between sub-accounts in trust (instead of repo trades), not used,8 – settlement of the deliverable futuresvia the Standard instrument, not used,9 – the first leg of repo trades T+N, not used,10 – the second leg of repo trades T+N, not used,11–close of the position after settlement of the cash-settled/delivarable futures (recorded as the counter-directed trade to close the position in the register),12 – not used,13 – not used,14 – the first trade for futures contracts of the calendar spread,15 – the second trade for futures contracts of the calendar spread,16 – trade in arisl instrument. |

Notes:

1. The key field is “id\_deal”.
2. The field “isin” corresponds in sense with the field “f07.contract”.
3. The field “id\_deal” corresponds in sense with the fields “f04\_XXYY.id\_deal”.
4. The field “type” corresponds in sense with the fields “f04\_XXYY.type”.
5. File opt\_deal.csv. Information on all on-exchange trades in options. The file is in CSV format.

Structure of the file **opt\_deal.csv**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | char(10) | Date |
| time | char(8) | Time in the format HH:MM:SS |
| isin | char(25) | Option  |
| price | numeric(16, 5) | Price (points). |
| vol | numeric(10) | Number of contracts |
| id\_deal | numeric(10) | Trade ID in the trading system |
| type | numeric(1) | Trade type:0 – anonymoius trade,2 – non-anonymous trade/position tramfer between sections with the same identity details. |

Notes:

1. The key field is “id\_deal”.
2. The field “isin” corresponds in sense with “o07.contract”.
3. The field “id\_deal” corresponds in sense with the fields “o04\_XXYY.id\_deal”.
4. The field “type” corresponds in sense with the fields “f04\_XXYY.type”.
5. File delinfoXX00.csv. Trading accounts of the clearing member (CM) including all its brokerage firms (XX is the clearing member code). The file is in CSV format.

Cancelled from 6 August 2015

Structure of the file **delinfoXX00.csv**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | Char(10) | Clearing date |
| kod | char(7) | CM’s/BF’s/client’s code |
| account | char(2) | Flag of CM/BF/client (‘RF’ - CM; ‘BF’ - BF; ‘CL’ – client). ‘CL’ always |
| code\_rdep | char(10) | Settlement depositary code |
| depo\_acc | char(32) | Number of the trading securities account |
| code\_rorg | char(10) | Settlement house code |
| money\_acc | char(20) | Number of the trading cash account |
| check | numeric(1) | The field is completed following the daily checking on whether the accounts stated in the fields «depo\_acc» and «money\_acc» are available for settlement.0 - settlement is not possible;1 - settlement is possible. It is completed if relevant data was provided before 17:00 MSK |
| id | char(25) | Trading member SE MICEX ID used for delivery of contracts on bonds. If delivery is performed by a client, short Client ID is shown separated by a space  |
| shortname | char(12) | Short name of the SE MICEX trading member which is used for delivery under contracts on bonds |

1. File clientsXX00.csv. Clearing register sections of the clearing member (CM) including all its brokerage firms (XX is the clearing member code). The file is in CSV format.

Structure of the file **clientsXX00.csv**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | char(10) | Clearing date |
| kod | char(7) | CM’s/BF’s/client clearing section code |
| account | char(2) | Flag of CM/BF/client (‘RF’ - CM; ‘BF’ - BF; ‘CL’ – client). ‘CL’ always |
| name | char(50) | Section name |
| id\_code | char(50) | Taxpayer ID (INN), or any code used instead |
| date\_open | date | Clearing section opening date  |
| reports | numeric(1) | Reports attribute («1» - the reports are produced, «0» – the reports are not produced) |
| date\_begin | date | Reporting start date for the section  |
| date\_end | date | Reporting end date for the section  |
| send\_kod | char(7) | Section code for debiting a fee for reports for the section |
| code\_adr | char(7) | Code for off order book trades |
| du | numeric(1) | The attribute showing whether or not the cash register section (securities collateral register) is intended for registering funds in asset management: 1 – the section is for registering funds in asset management. |
| segr | numeric(1) | Attribute «1» of the segregated Brokerage firm |
| isrepo | numeric(1) | An attribute for brokerage firms indicating a position rollover in the event of default in Standard instruments («0» - not active, «1» - active (it is set by default for all codes XXYY000 regardless of whether the broker is admitted to Standard or not), not used |
| rk | char(12) | Settlement code of the brokerage firm with the codes XX00000 and XXYY000 |
| rk\_type | char(1) | Settlement code type:'S' – proprietary; 'L' – client;'D' – asset management. |
| cross\_trade | numeric(1) | The attribute for cross agence trades:1 – cross agency trades are permitted  |
| account\_forts | char(30) | Number of the FORTS Account (type “Blocked for clearing on FORTS” of the trade section) |
| margin\_type | numeric(1) | Ongoing approach to calculate IM per every settlement code:2 – partial netting with regard to the brokerarge firm3 – partial netting with regard to the settlement code |
| sb | numeric(1) | Special Brokerage firm flag:1 – is a Special Brokerage firm0 – is not a Special Brokerage firm |

1. File daymonXXYY.csv. Information on a cash position of the clearing member (CM)/brokerage firm (BF) and their clients following the interim clearing session. The file is in CSV format.

Structure of the file **daymonXXYY.csv**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | char(10) | Clearing date |
| kod | char(7) | CM/BF/client/settlement code |
| account | char(2) | Flag of CM/BF/client (‘RF’ - CM; ‘BF’ - BF; ‘CL’ – client) |
| type | char(2) | Cash account type (‘MN’ – roubles and full collateral assets in foreign currency (in RUB); ‘PL’ – collateral assets in securities (no full prefunding) (in RUB)) |
| amount\_beg | numeric(16, 2) | For type =‘MN’: total assets in RUB and full collateral assets in foreign currency (in RUB) as at the beginning of the trading day /для type =‘PL’ – collateral in securities as at the beginning of the trading day (no full prefunding) (in RUB) |
| var\_marg | numeric(16, 2) | Variation margin by the end of the intraday clearing session (RUB) |
| prem | numeric(16, 2) | Option premium by the end of the intraday clearing session (RUB) |
| pay | numeric(16, 2) | Changes in balances by the end of the intraday clearing session:for type ‘MN’: per assets in RUB and full collateral assets in foreign currency (in RUB); for type ‘PL’: per collateral assets in securities (no full prefunding) (in RUB) |
| fut\_sbor | numeric(16, 2) | Sum of the exchange and clearing by the end of the interim clearing session (RUB). |
| opt\_sbor | numeric(16, 2) | Sum of the exchange and clearing fees for trades in options by the end of the intraday clearing session (RUB) It is calculated by market, on-market trades and option settlement trades |
| res\_vm | numeric(16, 2) | Excess of variation margin after the intraday clearing session (RUB). It is calculated after an option contract expiry date and used to calculated sufficiency of assets within the wide limit  |
| exp\_vm | numeric(16, 2) | Variation margin for options settlement trades. It is used to calculate a debt within the wide limit; not used |
| gol2 | numeric(16, 2) | Total IM posted for futures and options by the end of the intraday clearing session (RUB) |
| gowide | numeric(16, 2) | Total IM for futures and options which is calculated by the The wide limit (RUB); not used |
| amountl2 | numeric(16, 2) | For type ‘MN’: assets in RUB and full collateral in foreign currency (in RUB) by the end of the intraday clearing session /For type ‘PL’: collateral in securities (no full prefunding) by the end of the intraday clearing session (in RUB) |
| amountwide | numeric(16, 2) | A value calculated by the formula amount\_beg + exp\_vm + prem + pay - fut\_sbor - opt\_sbor by the end of the intraday clearing session not including VM besides that posted for options settlement; not used |
| freel2 | numeric(16, 2) | For type ‘MN’: available assets in RUB and full collateral in foreign currency (in RUB) by the end of the intraday clearing session /For type ‘PL’: available collateral in securities (no full prefunding) by the end of the intraday clearing session (in RUB). It is calculated as amountl2 - gol2 |
| freewide | numeric(16, 2) | Available assets by the end of the intraday clearing session (RUB). It is calculated as amountwide – gowide - res\_vm. Not used |
| margincall | char(1) | Margin call attribute. It may take one of three values: ‘A’, ‘B’ or ‘ ’ (the attribute is not set). Not used. |
| du | numeric(1) | An attribute of trust accounts (1 - trust account, 0 - not trust account) |
| rub\_beg | numeric(16,2) | Cash in RUB at the beginning of the trading day |
| rub\_pay | numeric(16,2) | Change in cash in RUB by the end of the intraday clearing session  |
| rubl2 | numeric(16,2) | Cash in RUB by the end of the interim clearing session  |
| com\_pl\_beg | numeric(16,2) | Estimated value of foreign currency as at the beginning of the trading day (in RUB) |
| com\_pl\_pay | numeric(16,2) | Change in the estimated value of the foreign currency by the end of the interim clearing session (in RUB) |
| com\_pll2 | numeric(16,2) | Estimated value of the foreign currency by the end of the intraday clearing session (in RUB) |
| ext\_rez | numeric(20,2) | Amount in RUB to be blocked for positions in RUONIA contracts. This amount is reserved to cover changes in RUONIA (used to calculated variation margin) published by CBR  |

Notes:

1. The key fields are “date”, “kod”, “account”, and “typе”.
2. The fields kod=‘Расчетный код’ and account=‘RK’ are given in the report “daymonXX00” at the level of clearing member.
3. For type ‘PL’ the fields “rub\_beg”, “rub\_pay”, “rubl2”, “com\_pl\_beg”, “com\_pl\_pay”, and “com\_pll2” are not filled.
4. File delivery\_step1XX00.csv. Information on first phase settlement of trades in deliverable futures and Standard instruments for the clearing member (CM) including its all brokerage firms (XX is the clearing member code). The file is in CSV format.

Cancelled due to the abandonment of the Standard Market.

Structure of the file **delivery\_step1XX00.csv**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | char(10) | Clearing date |
| kod | char(7) | CM’s/BF’s/client’s code |
| account | char(2) | Flag of CM/BF/client (‘RF’ - CM; ‘BF’ - BF; ‘CL’ – client). ‘CL’ always |
| isin | char(25) | Futures contract/Standard instrument |
| pos | numeric(11) | Number of positions to be settled as at the first delivery stage |
| pos\_iskl | numeric(11) | Number of futures positions cancelled due to the same INN (code) in the registries  |
| pos\_neisp | numeric(11) | Number of positions not settled during the delivery first phase |
| neisp | numeric(11) | An attribute indicating whether positions stated in the field pos\_neisp was settled («0» - settled, «1» - not settled) |
| settl\_pair | char(7) | Not used. |
| asset\_code | char(25) | Trading code of the underlying asset |
| issue\_code | char(25) | Depositary code of the underlying asset |
| oblig\_rur | numeric(18, 2) | Obligations in RUB  |
| oblig\_uni | numeric(18) | Obligations in securities (pieces) |
| fulfil\_rur | numeric(18, 2) | Fulfilled obligations in RUB  |
| fulfil\_uni | numeric(18) | Fulfilled obligations in securities (pieces) |
| step | numeric(11) | Sequence number of a delivery stage |

Notes:

1. The key fields are “date”, “kod”, “account”, “isin”, “settl\_pair”, “asset\_code”, and “issue\_code”.
2. If “pos” > 0, there are buying obligations. Otherwise, there are selling obligations.
3. In the event of default, the field “issue\_code” will be empty.
4. The field “pos” is replicated for rows with same “date”, “kod”, “account”, and “isin”.
5. The field “neisp” takes the following values:
	1. 0 – due settlement,
	2. 1 – default.
6. File deliveryXX00.csv. Information on second phase settlement of deliverable futures and Standard instruments when the settlement was rolled over, for the clearing member and its all brokerage forms (XX is the clearing member code). The file is in CSV format.

Cancelled due to the abandonment of the Standard Market.

Structure of the file **deliveryXX00.csv**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | char(10) | Clearing date |
| kod | char(7) | CM’s/BF’s/client’s code |
| account | char(2) | Flag of CM/BF/client (‘RF’ - CM; ‘BF’ - BF; ‘CL’ – client). ‘CL’ always |
| isin | char(25) | Futures contract/Standard instrument |
| pos | numeric(11) | Number of positions to be settled as at the beginning of the delivery second stage (besides those for which INN (code) coincided)  |
| pos\_iskl | numeric(11) | Zero value always |
| pos\_neisp | numeric(11) | Number of positions not settled during the delivery first phase |
| neisp | numeric(11) | An attribute indicating whether positions stated in the field pos\_neisp was settled («0» - settled, «1» - not settled) |
| settl\_pair | char(7) | Not used |
| asset\_code | char(25) | Trading code of the underlying asset |
| issue\_code | char(25) | Depositary code of the underlying asset |
| oblig\_rur | numeric(18, 2) | Obligations in RUB  |
| oblig\_uni | numeric(18) | Obligations in securities (pieces) |
| fulfil\_rur | numeric(18, 2) | Fulfilled obligations in RUB  |
| fulfil\_uni | numeric(18) | Fulfilled obligations in securities (pieces) |
| step | numeric(11) | Sequence number of a delivery stage |

Notes:

1. The key fields are “date”, “kod”, “account”, “isin”, “settl\_pair”, “asset\_code”, and “issue\_code”.
2. If “pos” > 0, there are buying obligations. Otherwise, there are selling obligations.
3. In the event of default, the field “issue\_code” will be empty.
4. The field “pos” is replicated for rows with same “date”, “kod”, “account”, and “isin”.
5. The field “neisp” takes the following values:
	1. 0 – due settlement,
	2. 1 – default.
6. File tranfeeXXYY.dbf. Information about transaction fees of CF/BF and their clients. The file is in CSV format.

The file’sstructure is as follows:

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Field description  |
| date | char(10) | Clearing date |
| kod | char(7) | CF/BF/client’s code |
| account | char(2) | Flag of CF/BF/client ('RF' - CF; 'BF' - BF; 'CL' - client) |
| futopt | numeric (1) | Value = 0, the field is inactive  |
| MM | numeric (1) | Value = 0, the field is inactive  |
| addtr | numeric(10) | Value = 0, the field is inactive  |
| fee | numeric(16,2) | Value = 0, the field is inactive  |
| sbortr | numeric(16,2) | Amount of the fee charged by the technical center for trantype transactions that is to be deducted on the next trading day from the clearing register subaccount per the flag 'CL' (RUB) |
| vat\_sbortr | numeric(16,2) | VAT amount in the transaction fee charged by the technical center (RUB) |
| trantype | Numeric(1) | Transaction type (“0”: ineffective transaction fee, “1”: fee for an erroneous transaction other than Flood Control error; “2”: fee for Flood Cobtrol errors) |

Notes:

1. Key fields: “date’, ‘kod’, ‘account’, ‘sbortr’ and ‘trantype’.
2. If the erroneous transaction fee is less than RUB1,000 the field “sbortr” is to be empty.
3. File tranfXXYY.dbf. Information on a number of orders and trades in futures and spot instruments for the clearing member (CM)/brokerage firm (BF) and their clients. The file is in FoxPro 2.x format.

Cancelled from 21 October 2013

Structure of the file **tranfXXYY.dbf**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | char(10) | Clearing date |
| kod | char(7) | CM’s/BF’s/client’s code |
| account | char(2) | Flag of CM/BF/client (‘RF’ - CM; ‘BF’ - BF; ‘CL’ – client) |
| isin | char(25) | Futures contract |
| MM | numeric(1) | An attribute indicating whether it is the market maker’s account for a given instrument («1» - MM, «0» - not MM)  |
| addtr | numeric(10) | Number of transactions made in a given instrument by using the client account/CM/BF |

Notes:

1. The key fields are “date”, “kod”, “account”, and “isin”.
2. File tranoXXYY.dbf. Information on a number of orders and trades in options for the clearing member (CM)/brokerage firm (BF) and their clients. The file is in FoxPro 2.x format.

Cancelled from 21 October 2013

Structure of the file **tranoXXYY.dbf**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | char(10) | Clearing date |
| kod | char(7) | CM’s/BF’s/client’s code |
| account | char(2) | Flag of CM/BF/client (‘RF’ - CM; ‘BF’ - BF; ‘CL’ – client) |
| isin | char(25) | Option  |
| MM | numeric(1) | An attribute indicating whether it is the market maker’s account for a given instrument («1» - MM, «0» - not MM)  |
| addtr | numeric(10) | Number of transactions made in a given instrument by using the client account/CM/BF |

Notes:

1. The key fields are “date”, “kod”, “account”, and “isin”.
2. File repodeal\_XXYY.dbf. Information on repos made on the exchange’s trading regime for trades with settlement on T+N, for the clearing member (CM)/brokerage firm (BF) and their clients. The file is in the FoxPro 2.x format.

Cancelled from 01 March 2011

Structure of the file **repodeal\_XXYY.dbf**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| Repodealid | numeric(10) | Repo ID  |
| Isin1 | char(25) | Code (ID) of the security with a specific Settlement date for the first leg of the repo |
| Isin2 | char(25) | Code (ID) of the security with a specific Settlement date for the second leg of the repo |
| price1 | numeric(16,5) | Price of the first leg of the repo (in points) |
| price2 | numeric(16,5) | Price of the second leg of the repo (in points) |
| Swap\_price | numeric(16,5) | Swap price (in points) |
| vol | numeric(10) | Number of securities lots in the repo (in pieces) |
| rate | numeric(16,5) | Repo rate (in per cent per annum) |
| days | numeric(4) | Repo term (in calendar days) |
| kod\_sell | char(7) | Code of the position register section indicated in the repo order entered by the seller in the repo  |
| kod\_buy | char(7) | Code of the position register section indicated in the repo order entered by the buyer in the repo  |
| date | char(10) | Registration date of the repo trade |
| time | char(8) | Registration time of the repo trade |
| type | numeric(1) | An attribute of the repo:0 – the repo was executed based on anonymous order,1 – the repo was executed based on non-anonymous order,2 – two related buying/selling trades executed based on anonymous order,3 – two related buying/selling trades executed based on non-anonymous order,4 – the repo T+N executed as a part of a position mandatory closing out,5 – two related buying/selling trades executed as a part of a position mandatory closing out. |
| var\_marg\_b | numeric(16,2) | Obligations of the buyer of the repo to pay variation margin for the repo (in RUB) |
| var\_marg\_s | numeric(16,2) | Obligations of the seller of the repo to pay variation margin for the repo (in RUB) |
| user\_sell | char(20) | Name of the seller of the repo |
| user\_buy | char(20) | Name of the buyer of the repo |
| no\_buy | numeric(10) | ID of the repo order entered by the buyer  |
| no\_sell | numeric(10) | ID of the repo order entered by the seller |
| fee\_buy | numeric(16,2) | Sum of the exchange and clearing fees to be paid by the buyer of the repo (in RUB) |
| fee\_sell | numeric(16,2) | Sum of the exchange and clearing fees to be paid by the seller of the repo (in RUB) |
| date2 | date | Registration date of the repo trade |
| comm\_buy | char(20) | Notes to the repo order entered by the buyer of the repo  |
| comm\_sell | char(20) | Notes to the repo order entered by the seller of the repo  |
| du\_buy | numeric(1) | An attribute indicating whether the repo was executed by the buyer using trust assets:1 – trust assets were used,0 - trust assets were not used |
| du\_sell | numeric(1) | An attribute indicating whether the repo was executed by the seller using trust assets:1 – trust assets were used,0 - trust assets were not used |
| fee\_ns\_b | numeric(16,2) | Sum of the exchange and clearing fees for the repo executed based on non-anonymous repo order to be paid by the buyer of the repo (in RUB) |
| fee\_ns\_s | numeric(16,2) | Sum of the exchange and clearing fees for the repo executed based on non-anonymous repo order to be paid by the seller of the repo (in RUB) |
| price1\_rur | numeric(16,5) | Price of one security in the first part of the repo (in RUB) |
| price2\_rur | numeric(16,5) | Price of one security in the second part of the repo (in RUB) |
| swap\_rur | numeric(16,5) | Swap price (in RUB) |
| ext\_id\_b | numeric(11) | Additional ID stated in the repo order entered by the buyer of the repo  |
| ext\_id\_s | numeric(11) | Additional ID stated in the repo order entered by the seller of the repo  |
| date\_clr | date | Clearing date |
| fee\_ex\_b | numeric(16,2) | Exchange fee to be paid by the buyer of the repo (in RUB) |
| vat\_ex\_b | numeric(16,2) | VAT in the exchange fee to be paid by the buyer of the repo (in RUB) |
| fee\_cc\_b | numeric(16,2) | VAT in the exchange fee to be paid by the seller of the repo (in RUB) |
| vat\_cc\_b | numeric(16,2) | VAT in the clearing fee to be paid by the buyer of the repo(in RUB) |
| fee\_ex\_s | numeric(16,2) | Exchange fee to be paid by the buyer of the repo (in RUB) |
| vat\_ex\_s | numeric(16,2) | VAT in the exchange fee to be paid by the seller of the repo (in RUB) |
| fee\_cc\_s | numeric(16,2) | Clearing fee to be paid by the seller of the repo (in RUB) |
| vat\_cc\_s | numeric(16,2) | VAT in the clearing fee to be paid by the seller of the repo (in RUB) |
| id\_trade1 | numeric(10) | An identifier of the trade table record changing the position on the delivery date of the first part of the repo (the first trade registered following fulfilment of two related orders)  |
| id\_trade2 | numeric(10) | An identifier of the trade table record changing the position on the delivery date of the second part of the repo (the second trade registered following fulfilment of two related orders)  |

1. File repoordlog\_XXYY.dbf. Information on repo orders for on-exchange trades with settlement on T+N for the clearing member (CM)/brokerage firm (BF) and their clients. The file is in FoxPro 2.x format.

Cancelled from 01 March 2011

Structure of the file **repoordlog\_XXYY.dbf**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| numb\_order | numeric(10) | Repo order ID  |
| Isin1 | char(25) | Code (ID) of the security with a specific Settlement date for the first leg of repo |
| isin2 | char(25) | Code (ID) of the security with a specific Settlement date for the second leg of repo  |
| price1 | numeric(16,5) | Order price for the first part of the repo (in points) |
| price2 | numeric(16,5) | Order price for the second part of the repo (in points) |
| Swap\_price | numeric(16,5) | Swap price (in points) |
| vol | numeric(10) | Number of lots in the repo order (in security pieces)  |
| rate | numeric(16,5) | Repo rate (in per cent per annum) |
| days | numeric(4) | Repo term (in calendar days) |
| rest\_vol | numeric(10) | Number of security lots in the repo order residing the book after the repo trade was registered following the order partial filling |
| kod | char(7) | Code of the CM/BF/position register section |
| tip | numeric(1) | Order type: 1 – bid,2 – offer  |
| sost | numeric(1) | Operation with the repo order:1 – entry,2 – execution of the repo trade on the basis of entered repo order,0 – cancel |
| date | char(10) | Date of operation with the repo order |
| time | char(8) | Time of operation with the repo order |
| user | char(20) | Name of the trading member having performed the operation in the repo order |
| comment | char(20) | Notes to the repo order |
| gate | numeric(1) | An attribute showing whether the gateway was used:1 – Operation in the order for the repo trade was made via the gateway,0 – Operation in the order for the repo trade was not made via the gateway |
| hedge | numeric(1) | Hedge attribute:0 – active,1 – not active |
| user\_to | char(20) | Name of the trading member - the addressee of the repo trade (not empty for non-anonymous orders only) |
| type | numeric(1) | Repo trade attribute:0 – anonymous repo trade;1 – non-anonymous repo trade;2 – non-anonymous pair of related trades |
| date2 | date | Repo order operation date |
| du | numeric(1) | Trust assets attribute: 1 – trust assets were used,0 - trust assets were not used |
| price1\_rur | numeric(16,5) | Order price for the first part of the repo (in RUB) |
| price2\_rur | numeric(16,5) | Order price for the second part of the repo (in RUB) |
| Swap\_rur | numeric(16,5) | Swap price (in RUB) |
| ext\_id | numeric(11) | Additional ID indicated in the repo order |
| date\_exp | date | Repo order expiry date |
| n\_order1 | numeric(10) | Repo order ID on entry prior a rollover to the next trading sessions |
| date\_clr | date | Clearing date |

1. File multilegf04\_XXYY.csv. Information on on-exchange multi-leg trades executed by the clearing member (CM)/brokerage firm (BF) and their clients. The file is in CSV format.

Structure of the file **multilegf04\_XXYY.dbf**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| Id\_deal | numeric(10) | ID of the trade |
| isin | char(25) | Unique code of the instrument in the multi-leg trade |
| Price1 | numeric(16,5) | Price of the first part of the multi-leg trade (in points). |
| Price | numeric(16,5) | Trade price (in points).  |
| vol | numeric(10) | Number of contracts in the multi-leg trade (in pieces)  |
| rate | numeric(16,5) | Rate in per cent per annum. It is used for repo and a pair of trades  |
| days | numeric(4) | Term of the multi-leg trade (in calendar days). Applied to repos and pairs of trades |
| kod\_sell | char(7) | Position register section code stated in the seller’s order to execute the multi-leg trade  |
| kod\_rts\_s | char(7) | Seller code  |
| kod\_buy | char(7) | Position register section code stated in the buyer’s order to execute the multi-leg trade  |
| kod\_rts\_b | char(7) | Buyer code  |
| date | char(10) | Multi-leg trade registration date |
| time | char(8) | Multi-leg trade registration time |
| type | numeric(1) | An attribute for the repo trade:0 – the repo was executed based on anonymous order, not used,1 – the repo was executed based on non-anonymous order, not used,2 – two related buying/selling trades executed based on non-anonymous order,3 – two related buying/selling trades executed based on anonymous order |
| signs | numeric(11) | Bit mask of trade signs |
| var\_marg\_b | numeric(16,2) | Obligations of the buyer to pay variation margin for the multi-leg trade (in RUB), not used, |
| var\_marg\_s | numeric(16,2) | Obligations of the seller to pay variation margin for the multi-leg trade (in RUB), not used, |
| user\_sell | char(20) | Name of the buyer of the multi-leg trade |
| user\_buy | char(20) | Name of the seller of the multi-leg trade |
| no\_buy | numeric(15) | ID of the buyer’s order to execute the multi-leg trade  |
| no\_sell | numeric(15) | ID of the seller’s order to execute the multi-leg trade  |
| fee\_buy | numeric(16,2) | Sum of the exchange and clearing fees to be paid by the buyer (in RUB) |
| fee\_sell | numeric(16,2) | Sum of the exchange and clearing fees to be paid by the seller (in RUB) |
| date2 | Char(10) | Multi-leg trade registration date |
| comm\_buy | char(20) | Buyer’s notes to the multi-leg trade |
| comm\_sell | char(20) | Seller’s notes to the multi-leg trade |
| du\_buy | numeric(1) | An attribute showing that the multi-leg trade was executed by the buyer using trust assets:1 – trust assets were used,0 - trust assets were not used) |
| du\_sell | numeric(1) | An attribute showing that the multi-leg trade was executed by the seller using trust assets:1 - trust assets were used,0 - trust assets were not used |
| fee\_ns\_b | numeric(16,2) | Sum of the exchange and clearing fees for the multi-leg trade executed based on non-anonymous order to be paid by the buyer (in RUB) |
| fee\_ns\_s | numeric(16,2) | Sum of the exchange and clearing fees for the multi-leg trade executed based on non-anonymous order to be paid by the seller (in RUB) |
| price\_rur1 | numeric(16,5) | Price of the first part of the multi-leg trade (in RUB) |
| price\_rur | numeric(16,5) | Trade price in RUB.  |
| ext\_id\_b | numeric(11) | Additional ID indicated in the buyer’s order for the multi-leg trade  |
| ext\_id\_s | numeric(11) | Additional ID indicated in the seller’s order for the multi-leg trade  |
| date\_clr | Date | Clearing date |
| fee\_ex\_b | numeric(16,2) | Exchange fee to be paid by the buyer (in RUB) |
| vat\_ex\_b | numeric(16,2) | VAT in the exchange fee to be paid by the buyer (in RUB) – Not used |
| fee\_cc\_b | numeric(16,2) | Clearing fee to be paid by the buyer (in RUB) |
| vat\_cc\_b | numeric(16,2) | VAT in the clearing fee to be paid by the buyer (in RUB) – Not used |
| fee\_ex\_s | numeric(16,2) | Exchange fee to be paid by the seller (in RUB) |
| vat\_ex\_s | numeric(16,2) | VAT in the exchange fee to be paid by the seller (in RUB) – Not used |
| fee\_cc\_s | numeric(16,2) | Clearing fee to be paid by the seller (in RUB) |
| vat\_cc\_s | numeric(16,2) | VAT in the clearing fee to be paid by the seller (in RUB) – Not used |
| id\_trade | numeric(10) | An identifier of the trade table record changing the position on the delivery date of the first part of the multi-leg trade (the first trade registered following settlement of the multi-leg trade)  |
| price\_rur2 | numeric(16,2) | Amount of the second part of the repo (in RUB), not used. |

Notes:

1. The report is send following the main clearing session.
2. File multilegf04clXXYYZZZ.csv. Information on clients’ multi-leg trades (XXYYZZZ is the section code). The file is in CSV format.

Structure of the file **multilegf04clXXYY.csv**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| Id\_deal | numeric(10) | ID of the trade |
| isin | char(25) | Unique code of the instrument in the multi-leg trade |
| Price1 | numeric(16,5) | Price of the first part of the multi-leg trade (in points). |
| Price | numeric(16,5) | Trade price (in points). It is the swap price of the trade for repo and a pair of trades (in points). |
| vol | numeric(10) | Number of contracts of the multi-leg trade (in units)  |
| rate | numeric(16,5) | Rate in per cent per annum. Applied to repos and pairs of trades. Swap price is indicated for swap trades |
| days | numeric(4) | Term of the multi-leg trade (in calendar days). Applies to repos and pairs of trades |
| kod\_sell | char(7) | Position register section code stated in the seller’s order to execute the multi-leg trade  |
| kod\_rts\_s | char(7) | Seller code  |
| kod\_buy | char(7) | Position register section code stated in the buyer’s order to execute the multi-leg trade  |
| kod\_rts\_b | char(7) | Buyer RTS code  |
| date | char(10) | Multi-leg trade registration date |
| time | char(8) | Multi-leg trade registration time |
| type | numeric(1) | An attribute of the repo:0 – the repo was executed based on anonymous order,1 – the repo was executed based on non-anonymous order,2 – two related buying/selling trades executed based on non-anonymous order3 – two related buying/selling trades executed based on anonymous order |
| signs | numeric(11) | Bit mask of trade signs |
| var\_marg\_b | numeric(16,2) | Obligations of the buyer to pay variation margin for the multi-leg trade (in RUB), not used |
| var\_marg\_s | numeric(16,2) | Obligations of the seller to pay variation margin for the multi-leg trade (in RUB), not used |
| user\_sell | char(20) | Name of the buyer of the multi-leg trade |
| user\_buy | char(20) | Name of the seller of the multi-leg trade |
| no\_buy | numeric(15) | ID of the buyer’s order to execute the multi-leg trade  |
| no\_sell | numeric(15) | ID of the seller’s order to execute the multi-leg trade  |
| fee\_buy | numeric(16,2) | Sum of the exchange and clearing fees to be paid by the buyer (in RUB) |
| fee\_sell | numeric(16,2) | Sum of the exchange and clearing fees to be paid by the seller (in RUB) |
| date2 | char(10) | Multi-leg trade registration date |
| comm\_buy | char(20) | Buyer’s notes to the multi-leg trade |
| comm\_sell | char(20) | Seller’s notes to the multi-leg trade |
| du\_buy | numeric(1) | An attribute showing that the multi-leg trade was executed by the buyer using trust assets:1 - trust assets were used,0 - trust assets were not used |
| du\_sell | numeric(1) | An attribute showing that the multi-leg trade was executed by the seller using trust assets:1 - trust assets were used,0 - trust assets were not used |
| fee\_ns\_b | numeric(16,2) | Sum of the exchange and clearing fees for the multi-leg trade executed based on non-anonymous order to be paid by the buyer (in RUB) |
| fee\_ns\_s | numeric(16,2) | Sum of the exchange and clearing fees for the multi-leg trade executed based on non-anonymous order to be paid by the seller (in RUB) |
| price\_rur1 | numeric(16,5) | Price of the first part of the multi-leg trade (in RUB) |
| price\_rur | numeric(16,5) | Trade price in RUB.  |
| ext\_id\_b | numeric(11) | Additional ID indicated in the buyer’s order for the multi-leg trade  |
| ext\_id\_s | numeric(11) | Additional ID indicated in the seller’s order for the multi-leg trade  |
| date\_clr | Date | Clearing date |
| fee\_ex\_b | numeric(16,2) | Exchange fee to be paid by the buyer (in RUB) |
| vat\_ex\_b | numeric(16,2) | VAT in the exchange fee to be paid by the buyer (in RUB) – Not used |
| fee\_cc\_b | numeric(16,2) | Clearing fee to be paid by the buyer (in RUB) |
| vat\_cc\_b | numeric(16,2) | VAT in the clearing fee to be paid by the buyer (in RUB) – Not used |
| fee\_ex\_s | numeric(16,2) | Exchange fee to be paid by the seller (in RUB) |
| vat\_ex\_s | numeric(16,2) | VAT in the exchange fee to be paid by the seller (in RUB) – Not used |
| fee\_cc\_s | numeric(16,2) | Clearing fee to be paid by the seller (in RUB) |
| vat\_cc\_s | numeric(16,2) | VAT in the clearing fee to be paid by the seller (in RUB) – Not used |
| id\_trade | numeric(10) | An identifier of the trade table record changing the position on the delivery date of the first part of the multi-leg trade (the first trade registered following settlement of the multi-leg trade)  |
| price\_rur2 | numeric(16,2) | Amount of the second part of the repo (in RUB) |

Notes:

1. The file is produced if the reporting service was ordered for the clearing registers section.

1. File multilegordlog\_XXYY.csv. The operation log for orders (matching and cancellations) to execute multi-leg trades for the clearing member (CM)/brokerage firm (BF) and their clients. The file is in CSV format.

Structure of the file **multilegordlog\_XXYY.csv**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| numb\_order | numeric(15) | ID of the order for multi-leg trade |
| isin | char(25) | Unique code of the instrument in the multi-leg trade |
| price | numeric(16,5) | Price of the first part of the repo and multi-leg trade (in points) |
| vol | numeric(10) | Number of contracts in the order for the multi-leg trade (in pieces)  |
| rate | numeric(16,5) | Rate in per cent per annum sued for repos and multi-leg trades |
| days | numeric(4) | Term of the multi-leg trade (in calendar days). Used for repos and multi-leg trades  |
| rest\_vol | numeric(10) | Number of contracts in the order for the multi-leg trade which remained after the registration of the multi-leg trade if the order was filled in part  |
| kod | char(7) | Code of the CM/BF/position register section |
| tip | numeric(1) | Order type: 1 – bid,2 – offer  |
| sost | numeric(1) | Operation in the order for the multi-leg trade:1 – enter,2 – execution of the multi-leg trade based on the entered order,0 – deletion  |
| date | char(10) | Date of the operation with the order for the multi-leg trade |
| time | char(8) | Time of the operation  |
| user | char(20) | Name of the trading member having performed the operation in the order for the multi-leg trade |
| comment | char(20) | Notes to the order for multi-leg trade  |
| gate | numeric(1) | An attribute showing whether the gateway was used:1 – Operation in the order for the multi-leg trade was made via the gateway,0 – Operation in the order for the multi-leg trade was not made via the gateway |
| hedge | numeric(1) | Hedge attribute:0 – active,1 – not active |
| user\_to | char(20) | Name of the trading member to which the order for the multi-leg trade was addressed (for non-anonymous trades only) |
| type | numeric(1) | Repo attribute:0 – anonymous repo order; not used1 – non-anonymous repo order; not used2 – non-anonymous order for a multi-leg trade;3 – anonymous order for a multi-leg trade |
| date2 | date | Date of the operation with the order for the multi-leg trade |
| du | numeric(1) | An attribute showing whether the multi-leg trade is executed using trust assets:1 - trust assets were used,0 - trust assets were not used |
| price\_rur | numeric(16,5) | Price of the first part of the repo and multi-leg trade (in RUB) |
| ext\_id | numeric(11) | Additional ID indicated in the order for multi-leg trade |
| date\_exp | date | Expiry date of the order for multi-leg trade |
| n\_order1 | numeric(15) | ID of the order for multi-leg trade on entry, prior to rollover to next trading sessions  |
| date\_clr | date | Clearing date |

Notes:

1. The report is send following the main clearing session.
2. File multileg\_deal.csv. Information on all on-exchange multi-leg trades. The file is in CSV format.

Structure of the file **multileg\_deal.csv**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | Char(10) | Clearing date |
| time | char(8) | Time in the format HH:MM:SS |
| isin | char(25) | Unique code of the instrument in the multi-leg trade |
| price1 | numeric(16,5) | Price of the first part of the multi-leg trade (in points). |
| price | numeric(16, 5) | Trade price (in points).  |
| vol | numeric(10) | Number of contracts. |
| rate | numeric(16,5) | Rate in per cent per annum sued for repos and multi-leg trades |
| id\_deal | numeric(10) | Trade ID in the trading system. |
| type | numeric(1) | Trade type (0 – order book, 2 – negotiated) |

Notes:

1. The key field is “id\_deal”.
2. “id\_deal” corresponds in sense to the field “multileg04\_XXYY.id\_deal”.
3. File multileg\_dict.csv. Information on instruments of multi-leg trades. The file is in CSV format.

Structure of the file **multileg\_dict.csv**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | char(10) | Clearing date |
| isin | char(25) | Unique code of the instrument in the multi-leg trade |
| Num\_legs | numeric(10) | Number of parts in the multi-leg trade |
| isin\_leg | char(25) | Unique code of the instrument in the part of the multi-leg trade |
| vol | numeric(10) | Volume of the instrument in the multi-leg trade |

Notes:

1. The report contains data on instruments of multi-leg trades.
2. It contains data on volume of an instrument in each part of a multi-leg trade.
3. Volume of the instrument may be positive or negative. Positive volume of an instrument indicates a long position in a multi-leg trade. Positive/negative volume indicates long/short position in the multi-leg trade.
4. File moncbXXYY.csv. Information on foreign currency and securities posted as collateral by the clearing member (CM)/brokerage firm (BF) and their clients. The file is in CSV format.

Structure of the file **moncbXXYY.csv**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | char(10) | Clearing date DD/MM/YYYY |
| kod | char(7) | Cash register section code (register of securities collateral)  |
| account | char(2) | An attribute showing where foreign currency/securities are registered:'RF' – foreign currency/securities registered in all sections of the cash register/securities register of the clearing member;'BF' – foreign currency/securities registered in all sections of the cash register/collateral securities register of the clearing member;'CL' – foreign currency/securities registered in a section of the cash register/collateral securities register |
| type | char(60) | Code of the foreign currency/security  |
| amount\_beg | numeric(10) | Number of securities/amount of foreign currency as at the beginning of the reporting date |
| amt\_beg\_cb | numeric(16,2) | Value of foreign currency at the previous day’s CBR exchange rate at the beginning of the reporting date (in RUB). Null for securities  |
| pay | numeric(16,2) | Daily change in the number of securities or amount of foreign currency (calculated as follows: «amount\_end» - «amount\_beg»)  |
| pay\_rub | numeric(16,2) | Daily change in value of foreign currency (in RUB) at the exchange rate set by the Bank of Russia for the reporting date (calculated as follows: «amount\_end\_rub» - «amount\_beg\_rub»). Nul for securities |
| amount\_end | numeric(10) | Number of securities/amount of foreign currency as at the end of the reporting date  |
| amt\_end\_cb | numeric(16,2) | Value of foreign currency at the current day’s CBR exchange rate at the end of the reporting date (in RUB). Null for securities |
| rate\_cb | numeric(16,5) | CBR exchange rate for the reporting date. Nul for securities |
| rate | numeric(16,5) | Estimated value of a foreign currency and security unit (in RUB) |
| go | numeric(16,2) | Estimated value of the foreign currency and securities in RUB (calculated as follows «amount\_end» \* «rate») |
| com\_ensure | numeric(1) | Collateral type:0 – collateral in securities (no full prefunding);1 – full collateral in foreign currency |

Notes:

1. Estimated value of securities or foreign currency that is indicated on the fields “rate” and “go” is determined by the Clearing Center Methodology for Selecting and Valuating Foreign Currencies and Securities Accepted as Collateral.
2. The report is sent after the end of the main clearing session.
3. File tranfeeupdXXYY.csv: information about the ineffective transaction fees of a BF as a total for all clearing register subaccounts with one INN (or a code used instead of INN), where XXYY is a BF’s code. The file is in CSV format.

The file’s structure is as follows:

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Field description  |
| date | char(10) | Clearing date |
| bf | char(4) | BF’s code |
| inn | char(50) | INN (taxpayer ID) or a code used instead  |
| tranpoints | numeric(16,2) | Total points for all ineffective transactions included into the orders\_log |
| feepoints | numeric(16,2) | Amount of exchange fees multiplied by a grade assigned to relevant trades |
| sbortr | numeric(16,2) | The fee for ineffective transactions included into the orders\_log |

Notes:

1. Those pairs (bf, inn) are shown for which “feepoints” and “tranpoints” did not equal “0” concurrently.
2. Information on “tranpoints” is in the file Parameters of ineffective and erroneous transaction fees.
3. The report is sent after the end of the main clearing session.
4. File tranfeeupddetailsXXYY.csv: detailed information about the ineffective transaction fees of a BF as a total for all clearing register subaccounts with one INN (or a code used instead of INN), where XXYY is a BF’s code. The file is in CSV format.

The file’s structure is as follows:

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Field description  |
| date | char(10) | Clearing date |
| inn | char(50) | INN (taxpayer ID) or a code used instead |
| kod | char(7) | Code of the clearing register subaccount |
| mm | numeric(1) | Action flag:1 – made via the market maker’s subaccount,0 – otherwise. |
| futoptspot | numeric(1) | Action flag:0 – trade/transaction in a futures,1 – trade/transaction in an option. |
| lowliquid | numeric(1) | Action flag:1 – trade/transaction in a low liquid instrument,0 – otherwise. |
| tran\_count | numeric(1) | Number of transactions with three flags (mm, futoptspot, lowliquid) |
| fee\_sum | numeric(16,2) | Exchange fee for trades with three flags (mm, futoptspot, lowliquid) |

Notes:

1. Action shall mean a trading transaction or a trade; any action has three flags according to its underlying instrument.
2. Those combinations of kod, mm, futoptspot and lowliquid are shown for which “tran\_count” and “fee\_sum” did not equal “0” concurrently.
3. The report is sent after the end of the main clearing session.
4. File tranfeeshareXXYY.csv: information about the ineffective transaction fees that are to be deducted from clearing register subaccounts with one INN (or a code used instead) where XXYY is a BF’s code. The file is in CSV format.

The file’s structure is as follows:

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Field description  |
| date | char(10) | Clearing date |
| inn | char(50) | INN (taxpayer ID) or a code used instead |
| kod | char(7) | Code of a clearing register subaccount |
| trannumber | numeric(10) | Number of transactions made via a given subaccount and included into the orders\_log |
| fee\_share | numeric(16,2) | Share of the fee for transactions included into the orders\_log that is to be deducted from a given subaccount. |

Notes:

1. Those values of the field “kod” are shown for which “fee\_share” did not equal “0”.
2. The report is sent after the end of the main clearing session.
3. File usersXXYY.csv: information about the erroneous transaction (other than Flood Control errors) fee by client logins of CF, BF and their clients. The file is in CSV format.

The file’s structure is as follows:

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Field description  |
| date | char(10) | Clearing date |
| kod | char(7) | Code of a clearing register subaccount of the CF/Bf/client  |
| account | char(2) | Flag of CM/BF/client ('RF' - CM; ‘BF’ - BF; 'CL' – client) |
| login | char(50) | Login name |
| maxmsg | numeric(10) | Login throughput as at the end of the trading session |
| sbor\_err | numeric(16,2) | Erroneous transaction (other than Flood Control error) fee (RUB) |

Notes:

1. If the erroneous transaction fee is less that RUB 1,000, the field “sbor\_err” is to be empty.
2. The report is sent after the end of the main clearing session.
3. File tranerrXXYY.csv: information about the number of erroneous transactions (other than Flood Control errors) by client logins of CF, BF and their clients. The file is in CSV format.

The file’s structure is as follows:

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Field description  |
| date | char(10) | Clearing date |
| moment | datetime | Erroneous transaction (other than a Flood Control error) time (yyyy-mm-dd hh:mm:ss) |
| kod | char(7) | Code of a clearing register subaccount of CF/BF/client |
| account | char(2) | Flag of CM/BF/client ('RF' - CM; ‘BF’ - BF; 'CL' – client) |
| login | char(50) | Login name |
| tran\_type | char(20) | Transaction type |
| err\_code | numeric(10) | Error code |
| err\_cnt | numeric(10) | Number of transactions of a certain type with a certain error |

Notes:

1. Those combinations login/moment are shown for which the limit was reached.
2. Information on the fields “tran\_type” and “err\_code” is in the file Parameters of ineffective and erroneous transaction fees.
3. The report is sent after the end of the main clearing session.
4. File tranerrfeeXXYY.csv: information about the erroneous transaction (other than Flood Control error) fee by client logins of CF, BF and their clients. The file is in CSV format.

The file’s structure is as follows:

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Field description  |
| date | char(10) | Clearing date |
| moment | datetime | Erroneous transaction (other than Flood Control error) time (yyyy-mm-dd hh:mm:ss) |
| kod | char(7) | Code of a clearing register subaccount of CF/BF/client  |
| account | char(2) | Flag of CM/BF/client ('RF' - CM; 'BF' - BF; 'CL' – client) |
| login | char(50) | Login name |
| err\_count | numeric(10) | Number of erroneous transactions (other than Flood Control errors) |
| points | numeric(10) | Total points for erroneous transactions (other than Flood Control errors) |
| sbor\_err | numeric(16,2) | The fee for erroneous transactions (other than Flood Control errors) executed by the time “moment” |

Notes:

1. Those combinations login/moment are shown for which the limit was reached.
2. Information on the field “points” is in the file Parameters of ineffective and erroneous transaction fees.
3. The report is sent after the end of the main clearing session.
4. paycbXX00.csv: transactions in foreign currency and securities posted as collateral by the clearing member (XX – clearing member code). CSV format.

File structure:

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | char(10) | Clearing date |
| kod | char(7) | Section code of the cash register/securities collateral register |
| account | char(2) | Flag of CM/BF/client (‘RF’: CM; ‘BF’: BF; ‘CL’: client). ‘CL’ always |
| type | char(2) | Cash account type (‘MN’ – full collateral assets in foreign currency; ‘PL’ – collateral assets in securities (no full prefunding)) |
| id\_pay | numeric(10) | Transaction ID |
| type\_pay | numeric(10) | Transaction type |
| asset | char(60) | Code of foreign currency/security |
| pay | numeric(16,5) | Amount of foreign currency/securities (in units) to be deposited or withdrawn  |
| rub\_equ | numeric(16,2) | Valuation of foreign currency/security in RUB  |
| purpose | char(255) | Payment purpose. Filled in for foreign currency only |

Note:

1. The field “pay” may be with a positive or negative value. If it is negative/positive, the assets are to be withdrawn/deposited.
2. The report is sent following the end of the evening (main) clearing session.
3. ****toeqXXYY.csv: correspondence between the brokerage firm/ position register clearing section on the Derivatives Market and the clearing member’s trading and clearing account on the Equity & Bond Market (XXYY – brokerage firm code). CSV format.****

**File structure:**

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | char(10) | Clearing date |
| kod | char(7) | Code of the clearing section of the clearing member/brokerage firm/client on the Derivatives Market  |
| account | char(2) | Sign of the CM/BF/client (‘RF’: CM; ‘BF’: BF; ‘CL’: client) on the Derivatives Market  |
| kod\_tks | char(12) | Code of the Trading & Clearing Account on the Equity & Bond Market  |
| kod\_client | char(12) | Short code of the client of the trading member on the Equity & Bond Market  |
| fav\_tks\_own | char(12) | Trading & Clearing Account tied to Proprietary Brokerage Firms  |
| fav\_tks\_client | char(12) | Trading & Clearing Account tied to Client Brokerage Firms  |
| fav\_tks\_du | char(12) | Trading & Clearing Account tied to Brokerage Firms – asset managers |
| broker\_ref | char(20) | Note to the settlement trade |

Note:

* + - 1. The fields “fav\_tks\_own”, “fav \_tks\_client” and “fav \_tks\_du” are filled in only if account=’RF’.
			2. The report is sent following the end of the evening (main) clearing session.
1. ****riskposXXYY.csv: position snapshot of the clearing member/brokerage from and their clients regarding risk management instruments, futures and options contracts (XXYY – brokerage firm code). CSV format.****

File structure:

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | char(10) | Clearing date |
| kod | char(7) | Code of CM/BF/client  |
| account | char(2) | Sign of the CM/BF/client (‘RF’: CM; ‘BF’: BF; ‘CL’: client) |
| isin | char(25) | Instrument unique code |
| type | char(1) | Instrument type (‘R’: risk, ’F’: futures contract, ’O’: option contract)  |
| pos\_beg | numeric(11) | Number of positions at the beginning of the day |
| pos\_end | numeric(11) | Number of positions at the end of the day |
| go\_brutto | numeric(16,2) | Gross IM (RUB) |
| fee\_risk | numeric(16,2) | Clearing fee for risk instruments (RUB); not used |
| sbor | numeric(16,2) | Exchange and clearing fees (RUB) for instruments other than risk instruments |

Note:

1. The risk instrument is added to f07.csv and dayf07.csv.
2. fut\_deal.csv does not contain information on the risk instrument.
3. Key fields are “date”, “kod”, “account” and “isin”.
4. The fields “pos\_beg” and “pos\_end” return the net position of the clearing member and all its clients including brokerage firms and their clients.
5. The fields “pos\_beg” and “pos\_end” return the net position of the brokerage firm and its clients.
6. The field “go\_brutto” is the sum of the same fields for all CM’s BFs.
7. The field “go\_brutto” for the BF is the gross IM of all BF’s clients.
8. The report contains both valid risk management instructions (risk management instruments) and positions in futures and option contracts.
9. The report is sent following the end of the evening (main) clearing session.
10. ****tofxXXYY.csv: correspondence between the brokerage firm on the Derivatives Market and the clearing member’s trading and clearing account on the FX Market (XXYY –brokerage firm code). CSV format.****

File structure:

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | char(10) | Clearing date |
| kod\_bf | char(7) | Code of the clearing section of the Brokerage Firm on the Derivatives Market  |
| kod\_tks | char(12) | Code of the trading & clearing account on the FX Market |

Note:

1. The report is sent following the end of the evening (main) clearing session.

1. persinvestXX00****.csv: clearing register sections flagged as “Individual Investment Account” (XX – clearing member code). CSV format.****

File structure:

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | char(10) | Clearing date |
| kod | char(7) | Code of the section of clearing registers |
| account | char(2) | Section sign:always 'CL' |

Note:

1. The report is sent following the end of the evening (main) clearing session.

1. delivery\_ofz\_ctd\_XX00.csv: state registration numbers of Russian Federation government bonds (OFZ) chosen by default to settle contracts (XX – clearing member code). CSV format.

File structure:

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| isin | char(25) | Instrument unique code (futures contract on the government bond (OFZ) basket) |
| registration\_number | char(20) | State registration number of the OFZ issue |

Note:

1. The report is sent to sellers following the end of the evening (main) clearing session on the day immediately preceding the last trading day for the OFZ basket.

1. ****mmfutXXYY.csv: market maker performance with regard to futures contracts of the brokerage firm (XXYY – brokerage firm code). CSV format.****

File structure:

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| session\_date | datetime | Trading date |
| agreement\_number | varchar(50) | No. of the market maker agreement |
| group\_code | varchar(7) | Group code used to get aggregate performance of the market maker  |
| code | varchar(25) | Code of the position register section  |
| symbol | varchar(25) | Code of the underlying of the futures contract |
| instrument | varchar(25) | Full code of the futures contract |
| spread | numeric(16,5) | Quote spread obligation |
| average\_spread | numeric(16,5) | Actual quote spread |
| min\_size | int | Minimum order size obligation  |
| spread\_maintenance\_% | numeric(16,2) | Actual length of time during which quotes were maintained, % |
| spread\_maintenance\_seconds | int | Actual length of time during which quotes were maintained, sec |
| quant\_start | varchar(20) | Time the quantum begins |
| quant\_end | varchar(20) | Time the quantum ends |
| quant\_seconds | int | Duration of the quantum, sec |
| time\_first\_spread | varchar(25) | Time the first spread appears |
| trade\_number | int | Number of trades |
| contract\_number | int | Number of contracts |
| position\_number | int | Number of positions |
| min\_maintenance\_% | int | Minimum length of two-sided quotes maintenance obligation (by default, =0 for futures), % |
| partial\_maintenance \_% | int | Partial length of two-sided quotes maintenance obligation, % |
| full\_maintenance\_% | int | Maximum length of two-sided quotes maintenance obligation, % |

Note:

1. The report is sent following the end of the evening (main) clearing session.

1. ****mmopt\_strikesXXXX.csv: market maker performance with regard to option contracts (by strikes) of the brokerage firms (XXYY – brokerage firm code). CSV format****

File structure:

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| session\_date | datetime | Trading date |
| agreement\_number | varchar(50) | No. of the market maker agreement |
| group\_code | varchar(7) | Group code used to get aggregate performance of the market maker  |
| code | varchar(7) | Code of the position register section  |
| symbol | varchar(25) | Code of the underlying of the futures contract |
| underlying | varchar(25) | Futures contract code |
| expiration\_date | datetime | Option contract expiry date  |
| instrument | varchar(25) | Option contract short code (code details at <http://moex.com/s205>) |
| spread | numeric(16,5) | Quote spread obligation |
| average\_spread | numeric(16,5) | Actual quote spread |
| min\_size | int | Minimum order size obligation  |
| shift\_from\_cs | numeric(16,5) | Deviation from the сentral strike  |
| call\_put | varchar(3) | Option contract type |
| spread\_maintenance\_% | numeric(16,2) | Actual length of time during which quotes were maintained, % |
| spread\_maintenance\_seconds | int | Actual length of time during which quotes were maintained, sec |
| quant\_start | varchar(20) | Time the quantum begins |
| quant\_end | varchar(20) | Time the quantum ends |
| quant\_seconds | int | Duration of the quantum, sec |
| time\_first\_spread | varchar(25) | Time the first spread appeared |
| trade\_number | int | Number of trades |
| contract\_number | int | Number of contracts |
| position\_number | int | Number of positions |
| min\_maintenance\_% | int | Minimum length of two-sided quotes maintenance obligation (by strike), % |
| partial\_maintenance \_% | int | Partial length of two-sided quotes maintenance obligation, % |
| full\_maintenance\_% | int | Maximum length of two-sided quotes maintenance obligation, % |

Note:

1. The report is sent following the end of the evening (main) clearing session.

1. ****mmopt\_averageXXXX.csv: market maker performance with regard to option contracts (the average performance per cent by strikes) of the brokerage firms (XXYY – brokerage firm code).** CSV format**

File structure:

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| session\_date | datetime | Trading date |
| agreement\_number | varchar(50) | No. of the market maker agreement |
| code | varchar(7) | Code of the position register section  |
| symbol | varchar(25) | Code of the underlying of the futures contract |
| underlying | varchar(25) | Futures contract code |
| expiration\_date | datetime | Option contract expiry date  |
| average\_maintenance\_by\_strike\_% | numeric (6,2) | The average arithmetic percentage of two-sided quotes maintenance lengths across all strikes, % (subject to minimum strike) |
| lowest\_maintenance\_strike\_% | numeric (6,2) | Minimum percentage of two-sided quotes maintenance by strike out of all market maker’s strikes |
| quant\_start | varchar(20) | Time the quantum begins |
| quant\_end | varchar(20) | Time the quantum ends |
| min\_maintenance\_% | int | Minimum length of two-sided quotes maintenance obligation (by strike), % |
| partial\_maintenance \_% | int | Partial length of two-sided quotes maintenance obligation, % |
| full\_maintenance\_% | int | Maximum length of two-sided quotes maintenance obligation, % |

Note:

1. The report is sent following the end of the evening (main) clearing session.

1. ****mmLP\_XXYY.csv: the fullfilment of the market making obligations on trading volume in derivatives (XXYY – brokerage firm code).** CSV format.**

File structure:

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| session\_date | datetime | Date |
| agreement\_number | varchar(50) | No. of the market maker agreement |
| group\_code | varchar(7) | Group code used to get aggregate performance of the market maker |
| type | char(1) | Instrumetn type (‘C’ – calendar spread on the futures, ’F’ – futures contract, ’O’ – option)  |
| symbol | varchar(25) | Underlying asset code |
| volume\_contracts | int | Trading volume in derivative contracts made by the market maker during the trading day |
| volume\_fee | numeric(16,2) | Exchange fee charged to the market maker for the trading day |
| volume\_contracts\_level\_1 | numeric(16,2) | Threshold value No.1 in respect to trading volume in derivative contracts made by the market maker over the calendar month |
| volume\_contracts\_level\_2 | numeric(16,2) | Threshold value No. 2 in respect to trading volume in derivative contracts made by the market maker over the calendar month |
| volume\_fee\_level\_1 | numeric(16,2) | Threshold value No.1 in resect to the exchange fee charged to the market maker for the calendar month |
| volume\_fee\_level\_2 | numeric(16,2) | Threshold value No.2 in resect to the exchange fee charged to the market maker for the calendar month |

Note:

The file is sent upon the end of the main clearing session.

1. **riskparamsXXYY.csv/ dayriskparamsXXYY.csv: **parameters set to limit position opening/order entries by clearing registers sections, a brokerage firm (BF) (XXYY – brokerage firm code).** CSV format.**

File structure:

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | char(10) | Clearing date |
| kod | char(7) | Code of the brokerage firm |
| prohibit\_coeff | numeric(16,2) | Coefficient set by the Clearing Center on the basis of the instruction from the clearing member to limit position opening/order entry in respect to the position register section opened for the brokerage firm |
| state | int | Status of the limit on position opening/order entry in respect to each position register section: 0 – not active,1 – active. |
| type | int | Type of the limit on position opening/order entry in respect to each position register section: 0 – applicable to position opening,1 – applicable to order entry. |
| del\_ord | int | Option to delete active orders in resoect to each position register section: 0 – not delete,1 – delete. |
| prohibit\_coeff\_bf | numeric(16,2) | Coefficient set by the Clearing Center on the basis of the instruction from the clearing member to limit position opening/order entry in respect to the brokerage firm |
| state\_bf | int | Status of the limit on position opening/order entry in respect to each position register section incorporating the brokerage firm code: 0 – not active,1 – active. |
| type\_bf | int | Type of the limit on position opening/order entry in respect to each position register section incorporating the brokerage firm code: 0 – applicable to position opening,1 – applicable to order entry. |
| del\_ord\_bf | int | Option to delete active orders in resoect to each position register section incorporating the brokerage firm code: 0 – not delete,1 – delete. |

Notes:

1. Файл **riskparamsXXYY.csv** предоставляется по итогам основной клиринговой сессии.

2. Файл day**riskparamsXXYY.csv** предоставляется по итогам промежуточной клиринговой сессии.

1. usersfcXXYY.csv: number of erroneous transactions and Flood Control fees for them per client login of the clearing member/brokerage firm and their clients by the transaction time. CSV format.

File structure:

|  |  |  |
| --- | --- | --- |
| Field name | Field type | Description |
| date | char(10) | Clearing date |
| moment | datetime | Date and time of erroneous transaction included in Flood Control (yyyy-mm-dd hh:mm:ss) |
| kod | char(7) | Code of the clearing registers section of the CM/BF/client |
| account | char(2) | Sign of the CM/BF/client (‘RF’: CM; ‘BF’: BF; ‘CL’: client) |
| login | char(50) | Login |
| maxmsg | numeric(10) | Пропускная способность логина на конец торговой сессии |
| sbor\_err | numeric(16,2) | Flood Control fee (RUB) |

Notes:

1. Если за ошибочные транзакции Flood Control сбор меньше 1000,00 руб., то в «sbor\_err» такое значение не отражается.
2. The report is sent following the end of the previous trading session.
3. Файл tranfcfeeXXYY.csv – информация о количестве и сборах за ошибочные транзакции Flood Control по каждому клиентскому логину расчетной фирмы (РФ)/брокерской фирмы (БФ) и их клиентов на время совершения транзакции. Файл формата CSV.

Структура файла **tranfcfeeXXYY.csv**

|  |  |  |
| --- | --- | --- |
| Название поля | Тип поля | Описание поля |
| date | char(10) | Дата проведения клиринга |
| moment | datetime | Время совершения ошибочной транзакции Flood Control (гггг-мм-дд чч:мм:cc) |
| kod | char(7) | Код раздела клиринговых регистров РФ/БФ/клиента |
| account | char(2) | Признак РФ/БФ/клиента ('RF' - РФ; 'BF' - БФ; 'CL' - клиент) |
| login | char(50) | Наименование логина |
| fc\_count | numeric(10) | Количество ошибочных транзакций Flood Control |
| sbor\_fc | numeric(16,2) | Сбор за количество ошибочных транзакций Flood Control, совершенное на момент времени «moment»  |

Примечание:

1. Отчет предоставляется по итогам предыдущей торговой сессии.