

---

# **netsgiro documentation**

*Release 1.0.0*

**Otovo AS**

**Jul 28, 2017**



<b>1</b>	<b>License</b>	<b>3</b>
<b>2</b>	<b>Project resources</b>	<b>5</b>
2.1	Quickstart . . . . .	5
2.2	Objects API . . . . .	12
2.3	Records API . . . . .	16
2.4	Constants . . . . .	22
2.5	Changelog . . . . .	24
	<b>Python Module Index</b>	<b>27</b>



netsgiro is a Python library for working with [Nets](#) AvtaleGiro and OCR Giro files.

AvtaleGiro is a direct debit solution that is in widespread use in Norway, with more than 15 000 companies offering it to their customers. OCR Giro is used by Nets and Norwegian banks to update payees on recent deposits to their bank accounts. In combination, AvtaleGiro and OCR Giro allows for a high level of automation of invoicing and payment processing.

The netsgiro library supports:

- Parsing AvtaleGiro agreements
- Creating AvtaleGiro payment requests
- Creating AvtaleGiro cancellations
- Parsing OCR Giro transactions

netsgiro is available from PyPI. To install it, run:

```
pip install netsgiro
```

For details and code examples, see [the netsgiro documentation](#).

For further details, please refer to the official [AvtaleGiro](#) and [OCR Giro](#) documentation from Nets.



# CHAPTER 1

---

## License

---

Copyright 2017 [Otovo AS](#). Licensed under the Apache License, Version 2.0. See the `LICENSE` file for the full license text.







```

Agreement (service_code=<ServiceCode.AVTALEGIRO: 21>, number=5, registration_type=
↪<AvtaleGiroRegistrationType.NEW_OR_UPDATED_AGREEMENT: 1>, kid='001044400507783',
↪notify=True),
Agreement (service_code=<ServiceCode.AVTALEGIRO: 21>, number=6, registration_type=
↪<AvtaleGiroRegistrationType.NEW_OR_UPDATED_AGREEMENT: 1>, kid='001045000507792',
↪notify=False),
Agreement (service_code=<ServiceCode.AVTALEGIRO: 21>, number=7, registration_type=
↪<AvtaleGiroRegistrationType.NEW_OR_UPDATED_AGREEMENT: 1>, kid='001057800507922',
↪notify=False),
Agreement (service_code=<ServiceCode.AVTALEGIRO: 21>, number=8, registration_type=
↪<AvtaleGiroRegistrationType.NEW_OR_UPDATED_AGREEMENT: 1>, kid='001060300509570',
↪notify=True),
Agreement (service_code=<ServiceCode.AVTALEGIRO: 21>, number=9, registration_type=
↪<AvtaleGiroRegistrationType.NEW_OR_UPDATED_AGREEMENT: 1>, kid='001087600508176',
↪notify=True),
Agreement (service_code=<ServiceCode.AVTALEGIRO: 21>, number=10, registration_type=
↪<AvtaleGiroRegistrationType.NEW_OR_UPDATED_AGREEMENT: 1>, kid='001105600508416',
↪notify=True),
Agreement (service_code=<ServiceCode.AVTALEGIRO: 21>, number=11, registration_type=
↪<AvtaleGiroRegistrationType.NEW_OR_UPDATED_AGREEMENT: 1>, kid='001123000508621',
↪notify=True),
Agreement (service_code=<ServiceCode.AVTALEGIRO: 21>, number=12, registration_type=
↪<AvtaleGiroRegistrationType.NEW_OR_UPDATED_AGREEMENT: 1>, kid='001124000508637',
↪notify=True),
Agreement (service_code=<ServiceCode.AVTALEGIRO: 21>, number=13, registration_type=
↪<AvtaleGiroRegistrationType.NEW_OR_UPDATED_AGREEMENT: 1>, kid='001138900509107',
↪notify=False),
Agreement (service_code=<ServiceCode.AVTALEGIRO: 21>, number=14, registration_type=
↪<AvtaleGiroRegistrationType.NEW_OR_UPDATED_AGREEMENT: 1>, kid='001143700509281',
↪notify=True),
Agreement (service_code=<ServiceCode.AVTALEGIRO: 21>, number=15, registration_type=
↪<AvtaleGiroRegistrationType.NEW_OR_UPDATED_AGREEMENT: 1>, kid='001146800509317',
↪notify=True),
Agreement (service_code=<ServiceCode.AVTALEGIRO: 21>, number=16, registration_type=
↪<AvtaleGiroRegistrationType.NEW_OR_UPDATED_AGREEMENT: 1>, kid='001186100509492',
↪notify=False)]

```

As you can see, all updates here are for new or updated agreements. If a payer deletes their AvtaleGiro agreement the `registration_type` field will be `DELETED_AGREEMENT`.

## Building payment requests

Once you have established AvtaleGiro agreements with some payers, you can start creating payment requests based on your invoices.

You start by creating a *Transmission* with Nets as the recipient:

```

>>> from datetime import date
>>> from decimal import Decimal
>>> import netsgiro
>>> transmission = netsgiro.Transmission(
...     number='1703231',
...     data_transmitter='01234567',
...     data_recipient=netsgiro.NETS_ID)

```

Then, add an AvtaleGiro transaction assignment to the transmission using `add_assignment()`:

```

>>> assignment = transmission.add_assignment(
...     service_code=netsgiro.ServiceCode.AVTALEGIRO,
...     assignment_type=netsgiro.AssignmentType.TRANSACTIONS,

```







```

Transmission(number='0170031', data_transmitter='00008080', data_recipient=
↳'00010200', date=datetime.date(1992, 1, 20))
>>> transmission.number
'0170031'

```

Each transmission can contain any number of assignments:

```

>>> len(transmission.assignments)
1
>>> assignment = transmission.assignments[0]
>>> assignment
Assignment(service_code=<ServiceCode.OCR_GIRO: 9>, type=<AssignmentType.
↳TRANSACTIONS: 0>, number='0000002', account='99991042764', agreement_id=
↳'001008566', date=datetime.date(1992, 1, 20))
>>> assignment.number
'0000002'
>>> assignment.get_earliest_transaction_date()
datetime.date(1992, 1, 20)
>>> assignment.get_latest_transaction_date()
datetime.date(1992, 1, 20)
>>> assignment.get_total_amount()
Decimal('51449')

```

Each assignment can contain any number of transactions, in this case OCR Giro payment transactions:

```

>>> pprint(assignment.transactions)
[Transaction(service_code=<ServiceCode.OCR_GIRO: 9>, type=<TransactionType.
↳PURCHASE_WITH_TEXT: 21>, number=1, date=datetime.date(1992, 1, 20),
↳amount=Decimal('1020'), kid='0000531', reference='099038562', text='Foo bar baz',
↳ centre_id='13', day_code=20, partial_settlement_number=1, partial_settlement_
↳serial_number='01464', sign='0', form_number='9636827194', bank_date=datetime.
↳date(1992, 1, 16), debit_account='99990512341', _filler='0000000'),
Transaction(service_code=<ServiceCode.OCR_GIRO: 9>, type=<TransactionType.FROM_
↳GIRO_DEBITED_ACCOUNT: 10>, number=2, date=datetime.date(1992, 1, 20),
↳amount=Decimal('1020'), kid='0036633', reference='097596016', text=None, centre_
↳id='13', day_code=20, partial_settlement_number=1, partial_settlement_serial_
↳number='01464', sign='0', form_number='9797596016', bank_date=datetime.date(1992,
↳ 1, 16), debit_account='99991005524', _filler='1883206'),
Transaction(service_code=<ServiceCode.OCR_GIRO: 9>, type=<TransactionType.FROM_
↳GIRO_DEBITED_ACCOUNT: 10>, number=3, date=datetime.date(1992, 1, 20),
↳amount=Decimal('560'), kid='0048763', reference='097581675', text=None, centre_
↳id='13', day_code=20, partial_settlement_number=1, partial_settlement_serial_
↳number='01464', sign='0', form_number='9797581675', bank_date=datetime.date(1992,
↳ 1, 17), debit_account='99991042764', _filler='1883206'),
Transaction(service_code=<ServiceCode.OCR_GIRO: 9>, type=<TransactionType.FROM_
↳GIRO_DEBITED_ACCOUNT: 10>, number=4, date=datetime.date(1992, 1, 20),
↳amount=Decimal('1020'), kid='0063851', reference='097585760', text=None, centre_
↳id='13', day_code=20, partial_settlement_number=1, partial_settlement_serial_
↳number='01464', sign='0', form_number='9797585760', bank_date=datetime.date(1992,
↳ 1, 17), debit_account='99991005524', _filler='1883206'),
Transaction(service_code=<ServiceCode.OCR_GIRO: 9>, type=<TransactionType.FROM_
↳GIRO_DEBITED_ACCOUNT: 10>, number=5, date=datetime.date(1992, 1, 20),
↳amount=Decimal('1020'), kid='0120243', reference='097591579', text=None, centre_
↳id='13', day_code=20, partial_settlement_number=1, partial_settlement_serial_
↳number='01464', sign='0', form_number='9797591579', bank_date=datetime.date(1992,
↳ 1, 17), debit_account='99990567898', _filler='1883206'),
Transaction(service_code=<ServiceCode.OCR_GIRO: 9>, type=<TransactionType.FROM_
↳GIRO_DEBITED_ACCOUNT: 10>, number=6, date=datetime.date(1992, 1, 20),
↳amount=Decimal('560'), kid='0165867', reference='097585109', text=None, centre_
↳id='13', day_code=20, partial_settlement_number=1, partial_settlement_serial_
↳number='01464', sign='0', form_number='9797585109', bank_date=datetime.date(1992,
↳ 1, 16), debit_account='99991042764', _filler='1883206'),
Transaction(service_code=<ServiceCode.OCR_GIRO: 9>, type=<TransactionType.FROM_
↳GIRO_DEBITED_ACCOUNT: 10>, number=7, date=datetime.date(1992, 1, 20),
↳amount=Decimal('1020'), kid='0178357', reference='097584855', text=None, centre_
↳id='13', day_code=20, partial_settlement_number=1, partial_settlement_serial_
↳number='01464', sign='0', form_number='9797584855', bank_date=datetime.date(1992,
↳ 1, 16), debit_account='99991005524', _filler='1883206'),

```

```

Transaction(service_code=<ServiceCode.OCR_GIRO: 9>, type=<TransactionType.FROM_
↳STANDING_ORDERS: 11>, number=8, date=datetime.date(1992, 1, 20), amount=Decimal(
↳'1500'), kid='02212291038306', reference='600155380', text=None, centre_id='13',
↳day_code=20, partial_settlement_number=1, partial_settlement_serial_number='01570
↳', sign='0', form_number='9600000000', bank_date=datetime.date(1992, 1, 20),
↳debit_account='99999545528', _filler='0000000'),
Transaction(service_code=<ServiceCode.OCR_GIRO: 9>, type=<TransactionType.FROM_
↳DIRECT_REMITTANCE: 12>, number=9, date=datetime.date(1992, 1, 20),
↳amount=Decimal('1200'), kid='02311291038304', reference='001020169', text=None,
↳centre_id='13', day_code=20, partial_settlement_number=1, partial_settlement_
↳serial_number='01570', sign='0', form_number='0000000000', bank_date=None, debit_
↳account='99991015406', _filler='0000000'),
Transaction(service_code=<ServiceCode.OCR_GIRO: 9>, type=<TransactionType.FROM_
↳BUSINESS_TERMINAL_GIRO: 13>, number=10, date=datetime.date(1992, 1, 20),
↳amount=Decimal('550'), kid='02310291038308', reference='220663800', text=None,
↳centre_id='13', day_code=20, partial_settlement_number=1, partial_settlement_
↳serial_number='01570', sign='0', form_number='0000000000', bank_date=None, debit_
↳account='99991017530', _filler='0000000'),
...
Transaction(service_code=<ServiceCode.OCR_GIRO: 9>, type=<TransactionType.FROM_
↳GIRO_DEBITED_ACCOUNT: 10>, number=20, date=datetime.date(1992, 1, 20),
↳amount=Decimal('540'), kid='02311291133188', reference='091308861', text=None,
↳centre_id='13', day_code=20, partial_settlement_number=9, partial_settlement_
↳serial_number='04514', sign='0', form_number='9636778117', bank_date=datetime.
↳date(1992, 1, 17), debit_account='99991011125', _filler='0000000')]

```

## Objects API

The objects API is the higher level API. It parses the “OCR” file into an object structure that for most use cases is more pleasant to work with than the lower level *records API*.

See *the quickstart guide* for an introduction to how to use the API.

`netsgiro.parse(data: str) → netsgiro.objects.Transmission`  
 Parse an OCR file into a Transmission object.

## Transmission class

**class** `netsgiro.Transmission`(*number*, *data\_transmitter*, *data\_recipient*, *date=None*, *assignments=NOTHING*)

Transmission is the top-level object.

An OCR file contains a single transmission. The transmission can contain multiple *Assignment* objects of various types.

**add\_assignment**(*\**, *service\_code*: `netsgiro.enums.ServiceCode`, *assignment\_type*: `netsgiro.enums.AssignmentType`, *agreement\_id*: `typing.Union[str, NoneType]` = `None`, *number*: `str`, *account*: `str`, *date*: `typing.Union[datetime.date, NoneType]` = `None`) → `netsgiro.objects.Assignment`

Add an assignment to the transmission.

**classmethod** **from\_records**(*records*: `typing.List[netsgiro.records.Record]`) → `netsgiro.objects.Transmission`

Build a Transmission object from a list of record objects.

**get\_num\_records**() → `int`

Get number of records in the transmission.

Includes the transmission’s start and end record.

**get\_num\_transactions** () → int  
Get number of transactions in the transmission.

**get\_total\_amount** () → decimal.Decimal  
Get the total amount from all transactions in the transmission.

**to\_ocr** () → str  
Convert the transmission to an OCR string.

**to\_records** () → typing.Iterable[netsgiro.records.Record]  
Convert the transmission to a list of records.

**assignments** = Attribute(name='assignments', default=Factory(factory=<class 'list'>, takes\_self=False), validator=<function str\_of\_length.<locals>.validator>, rep=...) → List of assignments.

**data\_recipient** = Attribute(name='data\_recipient', default=NOTHING, validator=<function str\_of\_length.<locals>.validator>, rep=...) → Data recipient's Nets ID. String of 8 digits.

**data\_transmitter** = Attribute(name='data\_transmitter', default=NOTHING, validator=<function str\_of\_length.<locals>.validator>, rep=...) → Data transmitter's Nets ID. String of 8 digits.

**date** = Attribute(name='date', default=None, validator=<optional validator for <instance\_of validator for type <class 'datetime.date'>>>, rep=...) → For OCR Giro files from Nets, this is Nets' processing date.  
For AvtaleGiro payment request, the earliest due date in the transmission is automatically used.

**number** = Attribute(name='number', default=NOTHING, validator=<function str\_of\_length.<locals>.validator>, rep=...) → Data transmitters unique enumeration of the transmission. String of 7 digits.

## Assignment class

**class** netsgiro.**Assignment** (*service\_code, type, number, account, agreement\_id=None, date=None, transactions=NOTHING*)

An Assignment groups multiple transactions within a transmission.

Use `netsgiro.Transmission.add_assignment()` to create assignments.

**add\_payment\_cancellation** (\*, *kid: str, due\_date: datetime.date, amount: decimal.Decimal, reference: typing.Union[str, NoneType] = None, payer\_name: typing.Union[str, NoneType] = None, bank\_notification: typing.Union[bool, str] = False*) → netsgiro.objects.Transaction

Add an AvtaleGiro cancellation to the assignment.

The assignment must have service code `AVTALEGIRO` and assignment type `AVTALEGIRO_CANCELLATIONS`.

Otherwise, the cancellation must be identical to the payment request it is cancelling.

**add\_payment\_request** (\*, *kid: str, due\_date: datetime.date, amount: decimal.Decimal, reference: typing.Union[str, NoneType] = None, payer\_name: typing.Union[str, NoneType] = None, bank\_notification: typing.Union[bool, str] = False*) → netsgiro.objects.Transaction

Add an AvtaleGiro payment request to the assignment.

The assignment must have service code `AVTALEGIRO` and assignment type `TRANSACTIONS`.

**classmethod** **from\_records** (*records: typing.List[netsgiro.records.Record]*) → netsgiro.objects.Assignment

Build an Assignment object from a list of record objects.

**get\_earliest\_transaction\_date** () → typing.Union[datetime.date, NoneType]  
Get earliest date from the assignment's transactions.

**get\_latest\_transaction\_date** () → typing.Union[datetime.date, NoneType]  
Get latest date from the assignment's transactions.

**get\_num\_records** () → int  
Get number of records in the assignment.  
Includes the assignment's start and end record.

**get\_num\_transactions** () → int  
Get number of transactions in the assignment.

**get\_total\_amount** () → decimal.Decimal  
Get the total amount from all transactions in the assignment.

**to\_records** () → typing.Iterable[netsgiro.records.Record]  
Convert the assignment to a list of records.

**account** = Attribute(name='account', default=NOTHING, validator=<function str\_of\_length.<locals>.validator>, repr=True, cmp=True, hash=None, init=True, c  
The payee's bank account. String of 11 digits.

**agreement\_id** = Attribute(name='agreement\_id', default=None, validator=<optional validator for <function str\_of\_length.<locals>.validator>, repr=True, cmp=True, hash=None, init=True, c  
Used for OCR Giro.  
The payee's agreement ID with Nets. String of 9 digits.

**date** = Attribute(name='date', default=None, validator=<optional validator for <instance\_of validator for type <class datetime.datetime>>, repr=True, cmp=True, hash=None, init=True, c  
Used for OCR Giro.  
The date the assignment was generated by Nets.

**number** = Attribute(name='number', default=NOTHING, validator=<function str\_of\_length.<locals>.validator>, repr=True, cmp=True, hash=None, init=True, c  
The assignment number. String of 7 digits.

**service\_code** = Attribute(name='service\_code', default=NOTHING, validator=None, repr=True, cmp=True, hash=None, init=True, c  
The service code. One of *ServiceCode*.

**transactions** = Attribute(name='transactions', default=Factory(factory=<class 'list'>, takes\_self=False), validator=<optional validator for <instance\_of validator for type <class netsgiro.records.Transaction>>, repr=True, cmp=True, hash=None, init=True, c  
List of transaction objects, like *Agreement*, *PaymentRequest*, *Transaction*.

**type** = Attribute(name='type', default=NOTHING, validator=None, repr=True, cmp=True, hash=None, init=True, c  
The transaction type. One of *TransactionType*.

## Transaction classes

**class** netsgiro.**Agreement** (*service\_code*, *number*, *registration\_type*, *kid*, *notify*)  
Agreement contains an AvtaleGiro agreement update.  
Agreements are only found in assignments of the *AVTALEGIRO\_AGREEMENTS* type, which are only created by Nets.

**classmethod** **from\_records** (*records*: typing.List[netsgiro.records.Record]) → netsgiro.objects.Agreement  
Build an Agreement object from a list of record objects.

**to\_records** () → typing.Iterable[netsgiro.records.Record]  
Convert the agreement to a list of records.

**TRANSACTION\_TYPE** = 94

**kid** = Attribute(name='kid', default=NOTHING, validator=<optional validator for <instance\_of validator for type <class int>>, repr=True, cmp=True, hash=None, init=True, c  
KID number to identify the customer and invoice.

**notify** = Attribute(name='notify', default=NOTHING, validator=<instance\_of validator for type <class 'bool'>>, repr=True, cmp=True, hash=None, init=True, c  
Whether the payer wants notification about payment requests.

**number** = Attribute(name='number', default=NOTHING, validator=<instance\_of validator for type <class 'int'>>, repr=True, cmp=True, hash=None, init=True, c  
Transaction number. Unique and ordered within an assignment.

**registration\_type** = Attribute(name='registration\_type', default=NOTHING, validator=None, repr=True, cmp=True, hash=None, init=True, c  
Type of agreement registration update. One of *AvtaleGiroRegistrationType*.

**service\_code** = Attribute(name='service\_code', default=NOTHING, validator=None, repr=True, cmp=True, hash=NOTHING)  
The service code. One of *ServiceCode*.

**class** netsgiro.**PaymentRequest**(*service\_code, type, number, date, amount, kid, reference, text, payer\_name*)

PaymentRequest contains an AvtaleGiro payment request or cancellation.

To create a transaction, you will normally use the helper methods on *Assignment*: *add\_payment\_request()* and *add\_payment\_cancellation()*.

**classmethod** **from\_records**(*records: typing.List[netsgiro.records.Record]*) → netsgiro.objects.Transaction  
Build a Transaction object from a list of record objects.

**to\_records**() → typing.Iterable[netsgiro.records.Record]  
Convert the transaction to a list of records.

**amount** = Attribute(name='amount', default=NOTHING, validator=None, repr=True, cmp=True, hash=None, init=True)  
Transaction amount in NOK with two decimals.

**amount\_in\_cents**  
Transaction amount in NOK cents.

**date** = Attribute(name='date', default=NOTHING, validator=<instance\_of validator for type <class 'datetime.date'>>)  
The due date.

**kid** = Attribute(name='kid', default=NOTHING, validator=<optional validator for <instance\_of validator for type <class 'int'>>>)  
KID number to identify the customer and invoice.

**number** = Attribute(name='number', default=NOTHING, validator=<instance\_of validator for type <class 'int'>>)  
Transaction number. Unique and ordered within an assignment.

**payer\_name** = Attribute(name='payer\_name', default=NOTHING, validator=<optional validator for <instance\_of validator for type <class 'str'>>>)  
The value is only used to help the payee cross-reference reports from Nets with their own records. It is not visible to the payer.

**reference** = Attribute(name='reference', default=NOTHING, validator=<optional validator for <instance\_of validator for type <class 'str'>>>)  
This is a specification line that will, if set, be displayed on the payers account statement. Alphanumeric, max 25 chars.

**service\_code** = Attribute(name='service\_code', default=NOTHING, validator=None, repr=True, cmp=True, hash=NOTHING)  
The service code. One of *ServiceCode*.

**text** = Attribute(name='text', default=NOTHING, validator=<optional validator for <instance\_of validator for type <class 'str'>>>)  
This is up to 42 lines of 80 chars each of free text used by the bank to notify the payer about the payment request. It is not used if the payee is responsible for notifying the payer.

**type** = Attribute(name='type', default=NOTHING, validator=None, repr=True, cmp=True, hash=None, init=True, cmp=NOTHING)  
The transaction type. One of *TransactionType*.

**class** netsgiro.**Transaction**(*service\_code, type, number, date, amount, kid, reference, text, centre\_id, day\_code, partial\_settlement\_number, partial\_settlement\_serial\_number, sign, form\_number, bank\_date, debit\_account, filler*)

Transaction contains an OCR Giro transaction.

Transactions are found in assignments with the service code *OCR\_GIRO* type, which are only created by Nets.

**classmethod** **from\_records**(*records: typing.List[netsgiro.records.Record]*) → netsgiro.objects.Transaction  
Build a Transaction object from a list of record objects.

**to\_records**() → typing.Iterable[netsgiro.records.Record]  
Convert the transaction to a list of records.

**amount** = Attribute(name='amount', default=NOTHING, validator=None, repr=True, cmp=True, hash=None, init=True)  
Transaction amount in NOK with two decimals.







`to_ocr ()` → str  
Get record as OCR string.

**RECORD\_TYPE = 10**

**data\_recipient** = Attribute(name='data\_recipient', default=NOTHING, validator=<function str\_of\_length.<local

**data\_transmitter** = Attribute(name='data\_transmitter', default=NOTHING, validator=<function str\_of\_length.

**service\_code** = Attribute(name='service\_code', default=NOTHING, validator=None, repr=True, cmp=True, hash

**transmission\_number** = Attribute(name='transmission\_number', default=NOTHING, validator=<function str\_o

**class** netsgiro.records.**TransmissionEnd** (*service\_code, num\_transactions, num\_records, to-*  
*tal\_amount, nets\_date*)

TransmissionEnd is the first record in every OCR file.

**from\_string** (*line: str*) → netsgiro.records.Record  
Parse OCR string into a record object.

`to_ocr ()` → str  
Get record as OCR string.

**RECORD\_TYPE = 89**

**nets\_date** = Attribute(name='nets\_date', default=NOTHING, validator=None, repr=True, cmp=True, hash=None,

**num\_records** = Attribute(name='num\_records', default=NOTHING, validator=None, repr=True, cmp=True, hash=

**num\_transactions** = Attribute(name='num\_transactions', default=NOTHING, validator=None, repr=True, cmp=

**service\_code** = Attribute(name='service\_code', default=NOTHING, validator=None, repr=True, cmp=True, hash

**total\_amount** = Attribute(name='total\_amount', default=NOTHING, validator=None, repr=True, cmp=True, hash

**class** netsgiro.records.**AssignmentStart** (*service\_code, assignment\_type, assign-*  
*ment\_number, assignment\_account, agree-*  
*ment\_id=None*)

AssignmentStart is the first record of an assignment.

Each assignment can contain any number of transactions.

**from\_string** (*line: str*) → netsgiro.records.Record  
Parse OCR string into a record object.

`to_ocr ()` → str  
Get record as OCR string.

**RECORD\_TYPE = 20**

**agreement\_id** = Attribute(name='agreement\_id', default=None, validator=<optional validator for <function str\_of

**assignment\_account** = Attribute(name='assignment\_account', default=NOTHING, validator=<function str\_of\_l

**assignment\_number** = Attribute(name='assignment\_number', default=NOTHING, validator=<function str\_of\_ler

**assignment\_type** = Attribute(name='assignment\_type', default=NOTHING, validator=None, repr=True, cmp=Tr

**service\_code** = Attribute(name='service\_code', default=NOTHING, validator=None, repr=True, cmp=True, hash

**class** netsgiro.records.**AssignmentEnd** (*service\_code, assignment\_type, num\_transactions,*  
*num\_records, total\_amount=None,*  
*nets\_date\_1=None, nets\_date\_2=None,*  
*nets\_date\_3=None*)

AssignmentEnd is the last record of an assignment.

**from\_string** (*line: str*) → netsgiro.records.Record  
Parse OCR string into a record object.

`to_ocr ()` → str  
Get record as OCR string.

**RECORD\_TYPE = 88**

**assignment\_type** = Attribute(name='assignment\_type', default=NOTHING, validator=None, repr=True, cmp=True)

**nets\_date**

Nets' processing date.

Only used for OCR Giro.

**nets\_date\_1** = Attribute(name='nets\_date\_1', default=None, validator=None, repr=True, cmp=True, hash=None, i

**nets\_date\_2** = Attribute(name='nets\_date\_2', default=None, validator=None, repr=True, cmp=True, hash=None, i

**nets\_date\_3** = Attribute(name='nets\_date\_3', default=None, validator=None, repr=True, cmp=True, hash=None, i

**nets\_date\_earliest**

Earliest date from the contained transactions.

**nets\_date\_latest**

Latest date from the contained transactions.

**num\_records** = Attribute(name='num\_records', default=NOTHING, validator=None, repr=True, cmp=True, hash=

**num\_transactions** = Attribute(name='num\_transactions', default=NOTHING, validator=None, repr=True, cmp=

**service\_code** = Attribute(name='service\_code', default=NOTHING, validator=None, repr=True, cmp=True, hash=

**total\_amount** = Attribute(name='total\_amount', default=None, validator=None, repr=True, cmp=True, hash=Non

```
class netsgiro.records.TransactionAmountItem1 (service_code, transaction_type, trans-
action_number, nets_date, amount,
kid, centre_id=None, day_code=None,
partial_settlement_number=None, par-
tial_settlement_serial_number=None,
sign=None)
```

TransactionAmountItem1 is the first record of a transaction.

The record is used both for AvtaleGiro and for OCR Giro.

**from\_string** (line: str) → netsgiro.records.Record

Parse OCR string into a record object.

**to\_ocr** () → str

Get record as OCR string.

**RECORD\_TYPE** = 30

**amount** = Attribute(name='amount', default=NOTHING, validator=None, repr=True, cmp=True, hash=None, init=T

**centre\_id** = Attribute(name='centre\_id', default=None, validator=<optional validator for <function str\_of\_length.>

**day\_code** = Attribute(name='day\_code', default=None, validator=None, repr=True, cmp=True, hash=None, init=Tr

**kid** = Attribute(name='kid', default=NOTHING, validator=None, repr=True, cmp=True, hash=None, init=True, con

**nets\_date** = Attribute(name='nets\_date', default=NOTHING, validator=None, repr=True, cmp=True, hash=None,

**partial\_settlement\_number** = Attribute(name='partial\_settlement\_number', default=None, validator=None, r

**partial\_settlement\_serial\_number** = Attribute(name='partial\_settlement\_serial\_number', default=None, v

**service\_code** = Attribute(name='service\_code', default=NOTHING, validator=None, repr=True, cmp=True, hash=

**sign** = Attribute(name='sign', default=None, validator=<optional validator for <function str\_of\_length.<locals>.vali

**transaction\_number** = Attribute(name='transaction\_number', default=NOTHING, validator=None, repr=True,

**transaction\_type** = Attribute(name='transaction\_type', default=NOTHING, validator=None, repr=True, cmp=T

```
class netsgiro.records.TransactionAmountItem2 (service_code, transaction_type,
                                              transaction_number, reference,
                                              form_number=None, bank_date=None,
                                              debit_account=None, filler=None,
                                              payer_name=None)
```

TransactionAmountItem2 is the second record of a transaction.

The record is used both for AvtaleGiro and for OCR Giro.

```
from_string (line: str) → netsgiro.records.Record
    Parse OCR string into a record object.
```

```
to_ocr () → str
    Get record as OCR string.
```

**RECORD\_TYPE = 31**

**bank\_date** = Attribute(name='bank\_date', default=None, validator=None, repr=True, cmp=True, hash=None, init=True)

**debit\_account** = Attribute(name='debit\_account', default=None, validator=<optional validator for <function str\_of\_

**form\_number** = Attribute(name='form\_number', default=None, validator=<optional validator for <function str\_of\_

**payer\_name** = Attribute(name='payer\_name', default=None, validator=None, repr=True, cmp=True, hash=None, init=True)

**reference** = Attribute(name='reference', default=NOTHING, validator=None, repr=True, cmp=True, hash=None, init=True)

**service\_code** = Attribute(name='service\_code', default=NOTHING, validator=None, repr=True, cmp=True, hash=None, init=True)

**transaction\_number** = Attribute(name='transaction\_number', default=NOTHING, validator=None, repr=True, cmp=True, hash=None, init=True)

**transaction\_type** = Attribute(name='transaction\_type', default=NOTHING, validator=None, repr=True, cmp=True, hash=None, init=True)

```
class netsgiro.records.TransactionAmountItem3 (service_code, transaction_type, transac-
                                              tion_number, text)
```

TransactionAmountItem3 is the third record of a transaction.

The record is only used for some OCR Giro transaction types.

```
from_string (line: str) → netsgiro.records.Record
    Parse OCR string into a record object.
```

```
to_ocr () → str
    Get record as OCR string.
```

**RECORD\_TYPE = 32**

**service\_code** = Attribute(name='service\_code', default=NOTHING, validator=None, repr=True, cmp=True, hash=None, init=True)

**text** = Attribute(name='text', default=NOTHING, validator=None, repr=True, cmp=True, hash=None, init=True, copy=True)

**transaction\_number** = Attribute(name='transaction\_number', default=NOTHING, validator=None, repr=True, cmp=True, hash=None, init=True)

**transaction\_type** = Attribute(name='transaction\_type', default=NOTHING, validator=None, repr=True, cmp=True, hash=None, init=True)

```
class netsgiro.records.TransactionSpecification (service_code, transaction_type,
                                                transaction_number, line_number,
                                                column_number, text)
```

TransactionSpecification is used for AvtaleGiro transactions.

The record is only used when bank notification is used to notify the payer.

Each record contains half of an 80 char long line of text and can be repeated up to 84 times for a single transaction for a total of 42 lines of specification text.

```
from_string (line: str) → netsgiro.records.Record
    Parse OCR string into a record object.
```

```
classmethod from_text (*, service_code, transaction_type, transaction_number, text) → typ-
    ing.Iterable[typing.TransactionSpecification]
    Create a sequence of specification records from a text string.
```

`to_ocr ()` → str  
Get record as OCR string.

**classmethod** `to_text` (*records: typing.Sequence[typing.TransactionSpecification]*) → str  
Get a text string from a sequence of specification records.

**RECORD\_TYPE = 49**

**column\_number** = Attribute(name='column\_number', default=NOTHING, validator=None, repr=True, cmp=True,

**line\_number** = Attribute(name='line\_number', default=NOTHING, validator=None, repr=True, cmp=True, hash=

**service\_code** = Attribute(name='service\_code', default=NOTHING, validator=None, repr=True, cmp=True, hash=

**text** = Attribute(name='text', default=NOTHING, validator=<instance\_of validator for type <class 'str'>>, repr=True,

**transaction\_number** = Attribute(name='transaction\_number', default=NOTHING, validator=None, repr=True,

**transaction\_type** = Attribute(name='transaction\_type', default=NOTHING, validator=None, repr=True, cmp=T

**class** `netsgiro.records.AvtaleGiroAgreement` (*service\_code, transaction\_type, transaction\_number, registration\_type, kid, notify*)  
AvtaleGiroAgreement is used by Nets to notify about agreement changes.

This includes new or deleted agreements, as well as updates to the payer's notification preferences.

**from\_string** (*line: str*) → `netsgiro.records.Record`  
Parse OCR string into a record object.

`to_ocr ()` → str  
Get record as OCR string.

**RECORD\_TYPE = 70**

**kid** = Attribute(name='kid', default=NOTHING, validator=None, repr=True, cmp=True, hash=None, init=True, con

**notify** = Attribute(name='notify', default=NOTHING, validator=None, repr=True, cmp=True, hash=None, init=True,

**registration\_type** = Attribute(name='registration\_type', default=NOTHING, validator=None, repr=True, cmp=

**service\_code** = Attribute(name='service\_code', default=NOTHING, validator=None, repr=True, cmp=True, hash=

**transaction\_number** = Attribute(name='transaction\_number', default=NOTHING, validator=None, repr=True,

**transaction\_type** = Attribute(name='transaction\_type', default=NOTHING, validator=None, repr=True, cmp=T

## Constants

`netsgiro.NETS_ID`  
Nets' data transmitter/recipient ID

```
>>> import netsgiro
>>> netsgiro.NETS_ID
'00008080'
```

**class** `netsgiro.ServiceCode`  
Service codes tell which Nets service the record applies to.

**NONE = 0**  
Used for the transmission start and end record.

**OCR\_GIRO = 9**  
Used for all OCR Giro records.

**AVTALEGIRO = 21**  
Used for all AvtaleGiro records.

**class** `netsgiro.RecordType`  
Record types tell what type of record this is.

**TRANSMISSION\_START = 10**

See netsgiro.TransmissionStart.

**ASSIGNMENT\_START = 20**

See netsgiro.AssignmentStart.

**TRANSACTION\_AMOUNT\_ITEM\_1 = 30**

See netsgiro.TransactionAmountItem1.

**TRANSACTION\_AMOUNT\_ITEM\_2 = 31**

See netsgiro.TransactionAmountItem2.

**TRANSACTION\_AMOUNT\_ITEM\_3 = 32**

See netsgiro.TransactionAmountItem3.

**TRANSACTION\_SPECIFICATION = 49**

See netsgiro.TransactionSpecification.

**TRANSACTION\_AGREEMENTS = 70**

See netsgiro.TransactionAgreements.

**ASSIGNMENT\_END = 88**

See netsgiro.AssignmentEnd.

**TRANSMISSION\_END = 89**

See netsgiro.TransmissionEnd.

**class netsgiro.AssignmentType**

Assignment types tell what type of assignment this is.

**TRANSACTIONS = 0**

Used both for AvtaleGiro payment requests and OCR Giro transactions.

**AVTALEGIRO\_AGREEMENTS = 24**

Used for AvtaleGiro agreement updates.

**AVTALEGIRO\_CANCELLATIONS = 36**

Used for AvtaleGiro cancellations.

**class netsgiro.TransactionType**

Assignment types tell what type of transaction this is.

**FROM\_GIRO\_DEBITED\_ACCOUNT = 10**

Used for OCR Giro.

**FROM\_STANDING\_ORDERS = 11**

Used for OCR Giro.

**FROM\_DIRECT\_REMITTANCE = 12**

Used for OCR Giro.

**FROM\_BUSINESS\_TERMINAL\_GIRO = 13**

Used for OCR Giro.

**FROM\_COUNTER\_GIRO = 14**

Used for OCR Giro.

**FROM\_AVTALEGIRO = 15**

Used for OCR Giro.

**FROM\_TELEGIRO = 16**

Used for OCR Giro.

**FROM\_CASH\_GIRO = 17**

Used for OCR Giro.

**REVERSING\_WITH\_KID = 18**

Used for OCR Giro.

**PURCHASE\_WITH\_KID = 19**

Used for OCR Giro.

**REVERSING\_WITH\_TEXT = 20**

Used for OCR Giro.

**PURCHASE\_WITH\_TEXT = 21**

Used for OCR Giro.

---

**Note:** The value 21 is used for both `PURCHASE_WITH_TEXT` and `AVTALEGIRO_WITH_BANK_NOTIFICATION`. The enum representation will be `<PURCHASE_WITH_TEXT: 21>` in either case.

---

**AVTALEGIRO\_WITH\_PAYEE\_NOTIFICATION = 2**

Used for AvtaleGiro when you want to notify the payer yourself.

**AVTALEGIRO\_WITH\_BANK\_NOTIFICATION = 21**

Used for AvtaleGiro when you want the bank to notify the payer.

**AVTALEGIRO\_CANCELLATION = 93**

Used for transactions that are part of an AvtaleGiro cancellation assignment.

**AVTALEGIRO\_AGREEMENT = 94**

Used by Nets for updates to AvtaleGiro agreements.

**class** `netsgiro.AvtaleGiroRegistrationType`

AvtaleGiro registration types tell what kind of update this is.

**ACTIVE\_AGREEMENT = 0**

Used when the AvtaleGiro agreement assignment contains all currently active agreements.

**NEW\_OR\_UPDATED\_AGREEMENT = 1**

Used when the AvtaleGiro agreement assignment contains only changes, and the current agreement is new or updated.

**DELETED\_AGREEMENT = 2**

Used when the AvtaleGiro agreement assignment contains only changes, and the current agreement has been deleted.

## Changelog

### v1.0.0 (2017-05-20)

No changes from v1.0.0a3, which has been used in production for a few weeks without any issues.

### v1.0.0a3 (2017-05-03)

- Rename `netsgiro.TransactionType.AVTALEGIRO_AGREEMENTS` (plural) to `netsgiro.TransactionType.AVTALEGIRO_AGREEMENT` (singular).
- When writing record to OCR, cut `netsgiro.records.TransactionAmountItem2.payer_name` to 10 first chars, as that is all the field has room for.

### v1.0.0a2 (2017-04-26)

Major improvements and changes.

- The objects API now supports parsing all known file variants with `netsgiro.parse()` and can recreate the parsed OCR data perfectly with `netsgiro.Transmission.to_ocr()`.

- The objects API now does all the bookkeeping necessary for building payment requests. With this improvement, the code necessary to produce a payment request is cut to from around 100 to 25 lines of code.
- New *quickstart guide* shows how to parse files and build payment requests.
- All public methods and fields of both the *objects API* and *records API* are now documented.
- The low-level *records API* has been moved to the `netsgiro.records` module.

### **v1.0.0a1 (2017-04-17)**

Initial alpha release. No promises about backwards compatibility.



**n**

`netsgiro`, 12

`netsgiro.records`, 16



## A

account (netsgiro.Assignment attribute), 14  
 ACTIVE\_AGREEMENT (netsgiro.AvtaleGiroRegistrationType attribute), 24  
 add\_assignment() (netsgiro.Transmission method), 12  
 add\_payment\_cancellation() (netsgiro.Assignment method), 13  
 add\_payment\_request() (netsgiro.Assignment method), 13  
 Agreement (class in netsgiro), 14  
 agreement\_id (netsgiro.Assignment attribute), 14  
 agreement\_id (netsgiro.records.AssignmentStart attribute), 19  
 amount (netsgiro.PaymentRequest attribute), 15  
 amount (netsgiro.records.TransactionAmountItem1 attribute), 20  
 amount (netsgiro.Transaction attribute), 15  
 amount\_in\_cents (netsgiro.PaymentRequest attribute), 15  
 amount\_in\_cents (netsgiro.Transaction attribute), 15  
 Assignment (class in netsgiro), 13  
 assignment\_account (netsgiro.records.AssignmentStart attribute), 19  
 ASSIGNMENT\_END (netsgiro.RecordType attribute), 23  
 assignment\_number (netsgiro.records.AssignmentStart attribute), 19  
 ASSIGNMENT\_START (netsgiro.RecordType attribute), 23  
 assignment\_type (netsgiro.records.AssignmentEnd attribute), 19  
 assignment\_type (netsgiro.records.AssignmentStart attribute), 19  
 AssignmentEnd (class in netsgiro.records), 19  
 assignments (netsgiro.Transmission attribute), 13  
 AssignmentStart (class in netsgiro.records), 19  
 AssignmentType (class in netsgiro), 23  
 AVTALEGIRO (netsgiro.ServiceCode attribute), 22  
 AVTALEGIRO\_AGREEMENT (netsgiro.TransactionType attribute), 24  
 AVTALEGIRO\_AGREEMENTS (netsgiro.AssignmentType attribute), 23

AVTALEGIRO\_CANCELLATION (netsgiro.TransactionType attribute), 24  
 AVTALEGIRO\_CANCELLATIONS (netsgiro.AssignmentType attribute), 23  
 AVTALEGIRO\_WITH\_BANK\_NOTIFICATION (netsgiro.TransactionType attribute), 24  
 AVTALEGIRO\_WITH\_PAYEE\_NOTIFICATION (netsgiro.TransactionType attribute), 24  
 AvtaleGiroAgreement (class in netsgiro.records), 22  
 AvtaleGiroRegistrationType (class in netsgiro), 24

## B

bank\_date (netsgiro.records.TransactionAmountItem2 attribute), 21  
 bank\_date (netsgiro.Transaction attribute), 16

## C

centre\_id (netsgiro.records.TransactionAmountItem1 attribute), 20  
 centre\_id (netsgiro.Transaction attribute), 16  
 column\_number (netsgiro.records.TransactionSpecification attribute), 22

## D

data\_recipient (netsgiro.records.TransmissionStart attribute), 19  
 data\_recipient (netsgiro.Transmission attribute), 13  
 data\_transmitter (netsgiro.records.TransmissionStart attribute), 19  
 data\_transmitter (netsgiro.Transmission attribute), 13  
 date (netsgiro.Assignment attribute), 14  
 date (netsgiro.PaymentRequest attribute), 15  
 date (netsgiro.Transaction attribute), 16  
 date (netsgiro.Transmission attribute), 13  
 day\_code (netsgiro.records.TransactionAmountItem1 attribute), 20  
 day\_code (netsgiro.Transaction attribute), 16  
 debit\_account (netsgiro.records.TransactionAmountItem2 attribute), 21  
 debit\_account (netsgiro.Transaction attribute), 16  
 DELETED\_AGREEMENT (netsgiro.AvtaleGiroRegistrationType attribute), 24

## F

form\_number (netsgiro.records.TransactionAmountItem2 attribute), 21

form\_number (netsgiro.Transaction attribute), 16

FROM\_AVTALEGIRO (netsgiro.TransactionType attribute), 23

FROM\_BUSINESS\_TERMINAL\_GIRO (netsgiro.TransactionType attribute), 23

FROM\_CASH\_GIRO (netsgiro.TransactionType attribute), 23

FROM\_COUNTER\_GIRO (netsgiro.TransactionType attribute), 23

FROM\_DIRECT\_REMITTANCE (netsgiro.TransactionType attribute), 23

FROM\_GIRO\_DEBITED\_ACCOUNT (netsgiro.TransactionType attribute), 23

from\_records() (netsgiro.Agreement class method), 14

from\_records() (netsgiro.Assignment class method), 13

from\_records() (netsgiro.PaymentRequest class method), 15

from\_records() (netsgiro.Transaction class method), 15

from\_records() (netsgiro.Transmission class method), 12

FROM\_STANDING\_ORDERS (netsgiro.TransactionType attribute), 23

from\_string() (netsgiro.records.AssignmentEnd method), 19

from\_string() (netsgiro.records.AssignmentStart method), 19

from\_string() (netsgiro.records.AvtaleGiroAgreement method), 22

from\_string() (netsgiro.records.TransactionAmountItem1 method), 20

from\_string() (netsgiro.records.TransactionAmountItem2 method), 21

from\_string() (netsgiro.records.TransactionAmountItem3 method), 21

from\_string() (netsgiro.records.TransactionSpecification method), 21

from\_string() (netsgiro.records.TransmissionEnd method), 19

from\_string() (netsgiro.records.TransmissionStart method), 18

FROM\_TELEGIRO (netsgiro.TransactionType attribute), 23

from\_text() (netsgiro.records.TransactionSpecification class method), 21

## G

get\_earliest\_transaction\_date() (netsgiro.Assignment method), 13

get\_latest\_transaction\_date() (netsgiro.Assignment method), 13

get\_num\_records() (netsgiro.Assignment method), 13

get\_num\_records() (netsgiro.Transmission method), 12

get\_num\_transactions() (netsgiro.Assignment method), 14

get\_num\_transactions() (netsgiro.Transmission method), 12

get\_total\_amount() (netsgiro.Assignment method), 14

get\_total\_amount() (netsgiro.Transmission method), 13

## K

kid (netsgiro.Agreement attribute), 14

kid (netsgiro.PaymentRequest attribute), 15

kid (netsgiro.records.AvtaleGiroAgreement attribute), 22

kid (netsgiro.records.TransactionAmountItem1 attribute), 20

kid (netsgiro.Transaction attribute), 16

## L

line\_number (netsgiro.records.TransactionSpecification attribute), 22

## N

nets\_date (netsgiro.records.AssignmentEnd attribute), 20

nets\_date (netsgiro.records.TransactionAmountItem1 attribute), 20

nets\_date (netsgiro.records.TransmissionEnd attribute), 19

nets\_date\_1 (netsgiro.records.AssignmentEnd attribute), 20

nets\_date\_2 (netsgiro.records.AssignmentEnd attribute), 20

nets\_date\_3 (netsgiro.records.AssignmentEnd attribute), 20

nets\_date\_earliest (netsgiro.records.AssignmentEnd attribute), 20

nets\_date\_latest (netsgiro.records.AssignmentEnd attribute), 20

netsgiro (module), 12

netsgiro.NETS\_ID (built-in variable), 22

netsgiro.records (module), 16

NEW\_OR\_UPDATED\_AGREEMENT (netsgiro.AvtaleGiroRegistrationType attribute), 24

NONE (netsgiro.ServiceCode attribute), 22

notify (netsgiro.Agreement attribute), 14

notify (netsgiro.records.AvtaleGiroAgreement attribute), 22

num\_records (netsgiro.records.AssignmentEnd attribute), 20

num\_records (netsgiro.records.TransmissionEnd attribute), 19

num\_transactions (netsgiro.records.AssignmentEnd attribute), 20

num\_transactions (netsgiro.records.TransmissionEnd attribute), 19

number (netsgiro.Agreement attribute), 14

number (netsgiro.Assignment attribute), 14

number (netsgiro.PaymentRequest attribute), 15

number (netsgiro.Transaction attribute), 16

number (netsgiro.Transmission attribute), 13

O

OCR\_GIRO (netsgiro.ServiceCode attribute), 22

P

parse() (in module netsgiro), 12

parse() (in module netsgiro.records), 18

partial\_settlement\_number (netsgiro.records.TransactionAmountItem1 attribute), 20

partial\_settlement\_number (netsgiro.Transaction attribute), 16

partial\_settlement\_serial\_number (netsgiro.records.TransactionAmountItem1 attribute), 20

partial\_settlement\_serial\_number (netsgiro.Transaction attribute), 16

payer\_name (netsgiro.PaymentRequest attribute), 15

payer\_name (netsgiro.records.TransactionAmountItem2 attribute), 21

PaymentRequest (class in netsgiro), 15

PURCHASE\_WITH\_KID (netsgiro.TransactionType attribute), 23

PURCHASE\_WITH\_TEXT (netsgiro.TransactionType attribute), 24

R

RECORD\_TYPE (netsgiro.records.AssignmentEnd attribute), 19

RECORD\_TYPE (netsgiro.records.AssignmentStart attribute), 19

RECORD\_TYPE (netsgiro.records.AvtaleGiroAgreement attribute), 22

RECORD\_TYPE (netsgiro.records.TransactionAmountItem1 attribute), 20

RECORD\_TYPE (netsgiro.records.TransactionAmountItem2 attribute), 21

RECORD\_TYPE (netsgiro.records.TransactionAmountItem3 attribute), 21

RECORD\_TYPE (netsgiro.records.TransactionSpecification attribute), 22

RECORD\_TYPE (netsgiro.records.TransmissionEnd attribute), 19

RECORD\_TYPE (netsgiro.records.TransmissionStart attribute), 19

RecordType (class in netsgiro), 22

reference (netsgiro.PaymentRequest attribute), 15

reference (netsgiro.records.TransactionAmountItem2 attribute), 21

reference (netsgiro.Transaction attribute), 16

registration\_type (netsgiro.Agreement attribute), 14

registration\_type (netsgiro.records.AvtaleGiroAgreement attribute), 22

REVERSING\_WITH\_KID (netsgiro.TransactionType attribute), 23

REVERSING\_WITH\_TEXT (netsgiro.TransactionType attribute), 24

S

service\_code (netsgiro.Agreement attribute), 14

service\_code (netsgiro.Assignment attribute), 14

service\_code (netsgiro.PaymentRequest attribute), 15

service\_code (netsgiro.records.AssignmentEnd attribute), 20

service\_code (netsgiro.records.AssignmentStart attribute), 19

service\_code (netsgiro.records.AvtaleGiroAgreement attribute), 22

service\_code (netsgiro.records.TransactionAmountItem1 attribute), 20

service\_code (netsgiro.records.TransactionAmountItem2 attribute), 21

service\_code (netsgiro.records.TransactionAmountItem3 attribute), 21

service\_code (netsgiro.records.TransactionSpecification attribute), 22

service\_code (netsgiro.records.TransmissionEnd attribute), 19

service\_code (netsgiro.records.TransmissionStart attribute), 19

service\_code (netsgiro.Transaction attribute), 16

ServiceCode (class in netsgiro), 22

sign (netsgiro.records.TransactionAmountItem1 attribute), 20

sign (netsgiro.Transaction attribute), 16

T

text (netsgiro.PaymentRequest attribute), 15

text (netsgiro.records.TransactionAmountItem3 attribute), 21

text (netsgiro.records.TransactionSpecification attribute), 22

text (netsgiro.Transaction attribute), 16

to\_ocr() (netsgiro.records.AssignmentEnd method), 19

to\_ocr() (netsgiro.records.AssignmentStart method), 19

to\_ocr() (netsgiro.records.AvtaleGiroAgreement method), 22

to\_ocr() (netsgiro.records.TransactionAmountItem1 method), 20

to\_ocr() (netsgiro.records.TransactionAmountItem2 method), 21

to\_ocr() (netsgiro.records.TransactionAmountItem3 method), 21

to\_ocr() (netsgiro.records.TransactionSpecification method), 21

to\_ocr() (netsgiro.records.TransmissionEnd method), 19

to\_ocr() (netsgiro.records.TransmissionStart method), 18

to\_ocr() (netsgiro.Transmission method), 13

to\_records() (netsgiro.Agreement method), 14

to\_records() (netsgiro.Assignment method), 14  
 to\_records() (netsgiro.PaymentRequest method), 15  
 to\_records() (netsgiro.Transaction method), 15  
 to\_records() (netsgiro.Transmission method), 13  
 to\_text() (netsgiro.records.TransactionSpecification class method), 22  
 total\_amount (netsgiro.records.AssignmentEnd attribute), 20  
 total\_amount (netsgiro.records.TransmissionEnd attribute), 19  
 Transaction (class in netsgiro), 15  
 TRANSACTION\_AGREEMENTS (netsgiro.RecordType attribute), 23  
 TRANSACTION\_AMOUNT\_ITEM\_1 (netsgiro.RecordType attribute), 23  
 TRANSACTION\_AMOUNT\_ITEM\_2 (netsgiro.RecordType attribute), 23  
 TRANSACTION\_AMOUNT\_ITEM\_3 (netsgiro.RecordType attribute), 23  
 transaction\_number (netsgiro.records.AvtaleGiroAgreement attribute), 22  
 transaction\_number (netsgiro.records.TransactionAmountItem1 attribute), 20  
 transaction\_number (netsgiro.records.TransactionAmountItem2 attribute), 21  
 transaction\_number (netsgiro.records.TransactionAmountItem3 attribute), 21  
 transaction\_number (netsgiro.records.TransactionSpecification attribute), 22  
 TRANSACTION\_SPECIFICATION (netsgiro.RecordType attribute), 23  
 TRANSACTION\_TYPE (netsgiro.Agreement attribute), 14  
 transaction\_type (netsgiro.records.AvtaleGiroAgreement attribute), 22  
 transaction\_type (netsgiro.records.TransactionAmountItem1 attribute), 20  
 transaction\_type (netsgiro.records.TransactionAmountItem2 attribute), 21  
 transaction\_type (netsgiro.records.TransactionAmountItem3 attribute), 21  
 transaction\_type (netsgiro.records.TransactionSpecification attribute), 22  
 TransactionAmountItem1 (class in netsgiro.records), 20  
 TransactionAmountItem2 (class in netsgiro.records), 20  
 TransactionAmountItem3 (class in netsgiro.records), 21  
 transactions (netsgiro.Assignment attribute), 14  
 TRANSACTIONS (netsgiro.AssignmentType attribute), 23  
 TransactionSpecification (class in netsgiro.records), 21  
 TransactionType (class in netsgiro), 23  
 Transmission (class in netsgiro), 12  
 TRANSMISSION\_END (netsgiro.RecordType attribute), 23  
 transmission\_number (netsgiro.records.TransmissionStart attribute), 19  
 TRANSMISSION\_START (netsgiro.RecordType attribute), 22  
 TransmissionEnd (class in netsgiro.records), 19  
 TransmissionStart (class in netsgiro.records), 18  
 type (netsgiro.Assignment attribute), 14  
 type (netsgiro.PaymentRequest attribute), 15  
 type (netsgiro.Transaction attribute), 16