



COMMERZBANK

Basic technical information

What is XML?

XML stands for Extensible Markup Language, a text-encoding system that allows the storage of hierarchically structured data in a text file that is both human- and machine-readable.

Elements, i.e. data fields or groups of data fields, are defined using tags. Square brackets at the beginning and at the end of an element represent opening and closing tags. A closing tag is identical to the opening tag, except that the tag name is preceded by a forward slash. The content of an element is in between the tags.

`<Tag>Content</Tag>`

A group of elements may look like this:

```
<Person>
<Name>John Smith</Name>
<Address>
<Address line>Main St. 1</Address line>
<Address line>12345 Bigtown</Address line>
</Address>
<Date of birth>01/09/1986</Date of birth>
</Person>
```

The graphical representation of layers can be optimised to improve readability.

```
<Person>
  <Name>John Smith</Name>
  <Address>
    <Address line>Main St. 1</Address line>
    <Address line>12345 Bigtown</Address line>
  </Address>
  <Date of birth>01/09/1986</Date of birth>
</Person>
```

Elements may also contain attributes; for example, providing the definition of the date format for a date of birth.

```
<Person>
  <Name>John Smith</Name>
  <Address>
    <Address line>Main St. 1</Address line>
    <Address line>12345 Bigtown</Address line>
  </Address>
  <Date of birth format=dd/mm/yyyy>01/09/1986</Date of birth>
</Person>
```

Unstructured data, such as name and address lines, can be structured to allow for better validation of the data.

```
<Person>
  <Name>
    <First name>John</First name>
    <Last name>Smith</Last name>
  </Name>
  <Address>
    <Street name>Main St.</Street name>
    <House no.>1</House no.>
    <Postcode>12345</Postcode>
    <City>Bigtown</City>
  </Address>
  <Date of birth format=dd/mm/yyyy>01/09/1986</Date of birth>
</Person>
```

As you can see, whilst the data is identical, it has been structured, enabling validation.

The actual data files with an “.xml” extension are usually accompanied by a schema file that defines the structure of the data file and may also determine permissible field content, for example the fact that the “postcode” field has to comprise exactly five numeric characters. The structure specifies in which sequence elements appear, whether they are mandatory and how often they are allowed to appear. The schema file is saved with an “.xsd” extension.

XML and ISO 20022

All formats within ISO 20022 have the same basic structure. The Group Header provides information on the source of the file and the date of creation; it is followed by one, or several, collectors which in turn can contain one, or several, individual transactions. The collector in payment files contains, e.g., information on the payer and payer’s account, whilst the individual transactions include details on amount, payee, and remittance information for specific payments. In account statements, the collector contains information on the settlement account and statement number; the individual transactions represent the individual settlements.

English is used as a universal language for the element names, mostly excluding vowels.

Message reference	MessageIdentification	<MsgId>
Date (and time) of creation	CreationDateTime	<CreDtTm>

The ISO 20022 format definition is notably broad and takes potential idiosyncrasies around the world into account. The UTF-8 encoding standard is very extensive and can encode even Cyrillic or Asian characters.

Restrictions have been established for SEPA payments and cross-border transactions to simplify handling. Character encodings are limited to the standard alphabet, numbers, and a few special characters. Within elements, many special characters that are relevant only for certain countries, are excluded.

The German Banking Industry Committee is responsible for publishing the formats valid in Germany. Format descriptions are available for download with the relevant products.

ISO 20022: Message types and versions

Message type definition under ISO 20022 is subject to a fixed structure. We have outlined it below, using the example of a SEPA Direct Debit in the pain.008.001.02 format:

pain.008.001.02

		Version number (in this case: 02 = 2009)
		Sub-format (001 = ISO 20022 standard)
		Message (008 = direct debit)
		Business area (pain = Payments Initiation)

Common messages between customer and bank

Payment transactions	
pain.001	Credit Transfer Initiation
pain.008	Direct Debit Initiation
pain.002	Payment Status Report/log
Account information	
camt.053	Customer Statement (end of day)
camt.052	Customer Account Report (intraday)

Versions

The version numbers from one year can vary between messages. SEPA is currently based on the versions from 2009. The version number for credit transfers is 03, the version number for direct debits is 02. Whilst all formats are scheduled to be updated to their 2019 versions by 2025, the deviating numeration will remain in place. Target formats will be version number 09 for credit transfers and 08 for direct debits.

SEPA transfer	Foreign bank transfer	Account statement	Payment Status Report
pain.001.001.09	pain.001.001.09	camt.053.001.08	pain.002.001.10

XML files can be opened with a text editor. The format version used is visible at first glance.

```
<?xml version="1.0" encoding="UTF-8"?>
<Document xmlns="urn:iso:std:iso:20022:tech:xsd:pain.001.001.03"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:iso:std:iso:20022:tech:xsd:pain.001.001.03 pain.001.001.03.xsd">
  <CstmrCdtTrfInitn>
    ...
```

Payment file structure

All ISO 20022 XML files have a similar structure:

GroupHeader (only once per physical file)

Contains information such as message ID, creation date, and submitting individual/organisation.

Collector level *PaymentInformation* (at least once per file)

Contains information about the payer, payer's account, execution date, and payment method.

Transaction level *TransactionInformation* (at least once per collector)

Contains information about the payee, payee's account, payment amount, and remittance information.

GroupHeader <MsgId>

e.g. creation date, originator, checksum

PaymentInformation <PmtInfId>

e.g. payer, execution date, payment method, checksum

Collector

TransactionInformation <InstrId>

e.g. payee, payment amount, remittance information

Transaction

PaymentInformation <PmtInfId>

e.g. payer, execution date, payment method, checksum

Collector

TransactionInformation <InstrId>

e.g. payee, payment amount, remittance information

Transaction