

**THE CENTRAL BANK OF THE RUSSIAN FEDERATION
(BANK OF RUSSIA)**

**Unified Formats
of Electronic Banking Messages**

**STRUCTURE AND RULES FOR FILLING IN
THE HEADERS OF A SERVICE ENVELOPE**

Version 2018.32019.1.0

Moscow

2018

Abstract

This document contains a description of the structure and rules for filling in the headers of a service envelope used for exchange of electronic banking messages (payment and reporting messages) with customers of the Bank of Russia and between business units of the Bank of Russia using, for transfer of payment messages and proprietary information, the Transport System of Electronic Settlements and the Transport Media for Delivery of Messages, respectively.

This document has been developed for the unification of interfaces of the transport level.

This document may be supplemented and modified in connection with changes in the regulations of the Bank of Russia.

Table of Contents

NO TABLE OF CONTENTS ENTRIES FOUND.

1. Compatibility Requirements

In this document, the following verbs are used to define the level of the requirements:

MUST is used to indicate that it is necessary to meet this requirement in any circumstances.

SHOULD is used to indicate that it is necessary to meet this requirement if there are no serious reasons preventing it.

MAY is used to indicate that this requirement is optional and may be implemented or not, depending on the need.

Implementation is deemed to be incompatible if at least one of the NECESSARY requirements of this document has been violated. Implementation satisfying all NECESSARY and RECOMMENDED REQUIREMENTS is called fully compatible, and implementation satisfying all NECESSARY but not all RECOMMENDED requirements is called conditionally compatible.

2. General Information

The transport message contains components of information executed pursuant to the specifications of a particular protocol. For transmission of an electronic message containing application information by a transport message, such electronic message shall be placed in a service envelope providing an interface between the application level and the transport level. The service envelope consists of a block of headers and a body. The block of headers transmits technological information. The body of a service information transmits an EM containing application information.

3. The structure and syntax of a service envelope and headers

3.1 Namespaces

For this document version, the following namespaces are used:

'<http://www.w3.org/2003/05/soap-envelope>' (prefix env)

"urn:cbr-ru:msg:props:v1.3" (prefix props)

Note: A namespace prefix does not have any meaning and is used only for tying the names of components and attributes to the designation of a namespace.

3.2 Service envelope

The service envelope is generated pursuant to the recommendation [SOAP 12]. The description of the element composition of a service envelope is provided in the table below (see table 1). To specify a range of potential values of the power of a set of elements, the multiplicity of an element is used (see table 2).

Namespace

"<http://www.w3.org/2003/05/soap-envelope>" (the prefix env)

Table 1. Service envelope (env:Envelope)

Element description	Element type	Multiplicity
1 Envelope header (env:Header)		[0..1]
1.1 Header contents (any)	Header components	[0..n]
2 Envelope body (env:Body)		[1]
2.1 Envelope body contents (any)	Element containing EM	[0..n]

Table 2. Designation of the multiplicity of elements

Multiplicity	Value
[0..1]	Optional element, the maximum number of instances is 1
[0..n]	Optional element, the maximum number of instances is not limited
[1]	Compulsory element, only one instance
[1..n]	Compulsory element, the maximum number of instances is not limited

The service envelope can contain various information blocks in the header **env:Header**. Additional blocks of the header can be added at the transport level. Thus, the envelope header contains a set of transport and service headers.

The service envelope MAY contain an EM in the component **env:Body**. In UFEBM, in the service envelope body, **not more than one** EM is transmitted.

If an EM is executed in an EDS (AC) envelope, the component **env:Body** MUST contain an EDS (AC) envelope.

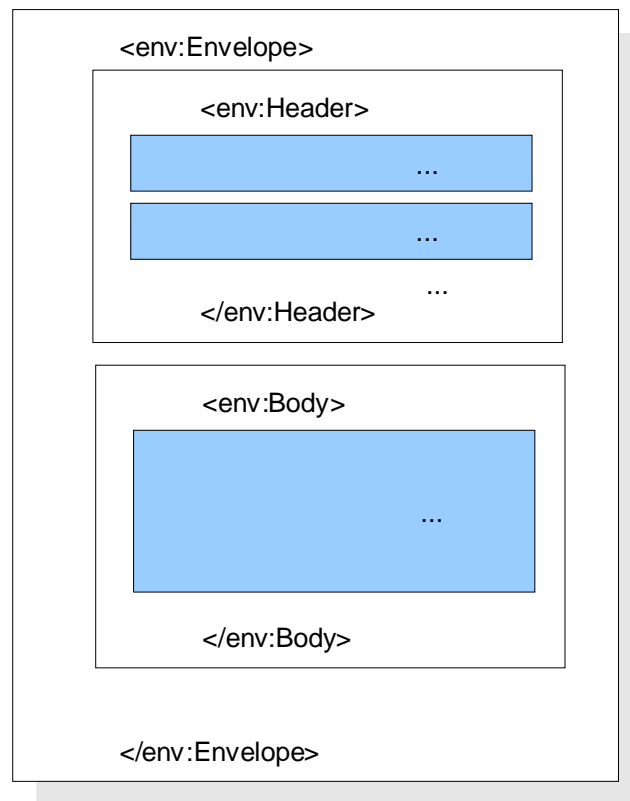


Fig. 1. Service envelope structure pursuant to the recommendation [SOAP12]

3.3 Principal information block of the header (MessageInfo)

The information block of a service envelope (**props:MessageInfo**) contains the principal information on an electronic message that is transmitted between the application software and *the transport adapter*, software functioning at the level of the transport layer (see table 3). The information block of a service envelope (**props:MessageInfo**) is compulsory.

3.3.1 Element composition

Namespace

"urn:cbr-ru:msg:props:v1.3" (the prefix props)

Table 3. Elements of the information block MessageInfo (props:MessageInfo)

Element description	Element type	Multiplicity
1 Recipient's address. The element is generated and filled in by the author of a service envelope transmitting a message for delivery. If multiplicity >1, an individual tag is used for each addressee. (<i>props:To</i>)	Identifier. Address information. Textual.	[1..1000]
2 Sender's address. The element is generated and filled in by the author of a service envelope transmitting a message for delivery (<i>props:From</i>)	Identifier. Address information. Textual.	[1]
3 Application identifier. The element is generated and filled in by the author of a service envelope transmitting a message for delivery (<i>props:AppMessageID</i>)	Identifier. Identifier of an electronic message. Textual, up to 100 characters.	[0..1]

Element description	Element type	Multiplicity
4 Identifier of a service envelope. The element is generated and filled in by the author of a service envelope transmitting a message for delivery (<i>props:MessageID</i>)	Identifier. Identifier of a service envelope. Textual, up to 100 characters.	[1]
5 Reference identifier of a service envelope. Generated for response messages and contains the value of the component props:MessageID of the initial message. The element is generated and filled in by the author of a service envelope transmitting a message for delivery (<i>props:CorrelationMessageID</i>)	Identifier. Identifier of a service envelope. Textual, up to 100 characters.	[0..1]
6 Type of electronic message. The element is generated and filled in by the author of a service envelope transmitting a message for delivery (<i>props:MessageType</i>)	Code. Type of electronic message. (1 = messages of the payment system, 2 = for exchange of information data (incl. banking and statistic reporting), 3 = for technological acknowledgement messages (within the framework of transport systems), 5 = other service messages.	[1]
7 Priority of the message. The element is generated and filled in by the author of a service envelope transmitting a message for delivery (<i>props:Priority</i>)	Code. Transport priority of an electronic message. Numeric, 1-digit.	[1]
8 Name of the file. The element is generated and filled in by the author of a service envelope transmitting a message for delivery (<i>props:LegacyTransportFileName</i>)	Identifier. File name. Textual.	[0..1]
9 Date and time of creation of the transport message. The element is generated and filled in by the author of a service envelope transmitting a message for delivery (<i>props:CreateTime</i>)	DateTime. [GOST ISO 8601-2001]. Format YYYY-MM-DDThh:mm:ssZ. GMT.	[0..1]
10 Date and time of sending of a transport message. The element is generated and filled in by the transport adapter of the sender (<i>props:SendTime</i>)	DateTime. [GOST ISO 8601-2001]. Format YYYY-MM-DDThh:mm:ssZ. GMT.	[0..1]
11 Date and time of receipt of a message in the payment system of a BoR BU or a participant. The element is filled in when a message from a participant is received in the payment system of a BoR BU or from the payment system of a BoR BU in the payment system of a participant (<i>props:ReceiveTime</i>)	DateTime. [GOST ISO 8601-2001]. Format YYYY-MM-DDThh:mm:ssZ. GMT.	[0..1]
12 Date and time a message was read by the receiving application. The element is filled in by the receiving application when reading the message (<i>props:AcceptTime</i>)	DateTime. [GOST ISO 8601-2001]. Format YYYY-MM-DDThh:mm:ssZ. GMT.	[0..1]

Element description	Element type	Multiplicity
13 Flag for requesting for technological acknowledgements. The element is generated and filled in by the author of a service envelope transmitting a message for delivery (<i>props:AckRequest</i>)	Indicator. Flag condition.	[0..1]
14 Sign of receipt of a message on removable media. The element is filled in with the value '1' if the EM was received from an EM author on removable media. The element is not to be filled in if EM was sent through communication channels (<i>props:ReceivedOnExternalMedium</i>)	Code. Sign of receipt of a message on removable media.	[0..1]
15 Sign of repeated sending. (<i>props:PDI</i>)	Indicator. Flag condition.	[0..1]

3.3.2 Rules for filling in

3.3.2.1 Addresses of the recipient and the sender

Addresses of the recipient and the sender **MUST** be generated by an author of a service envelope in accordance with the logical addresses of exchange participants. If a message is sent to several participants, the address of each recipient is executed as a separate component **props:To**.

Address format:

"<prefix>:<value>", where

<prefix> is the string prefix of an address identifying the notation (presentation form) of a subsequent address;

<value> is an address value.

For exchange of EMs, a unique identifier of an author (UIMA) should be used in the following format: "uic:<UIMA><NA>", where:

<UIMA> is a 10-digit unique identifier of an author for the payment system; for information systems, UIMA will be determined additionally.

<NA> is the 2-digit numeric identifier defining the sequential number of the AWS of an author of an electronic message within one UIMA (from 0 to 99). To be assigned by a settlement participant proceeding from the requirement to ensure the uniqueness of the identifier for all AWSs of this participant. If an author of an EM has only one AWS, <NA> is equal to '00'. Example of address generation:

```
uic:040167000032
```

Processing of MD payment information after centralisation of processing is performed in the uniform SMDIP, whose UIMA is calculated based on the MD's RCBIC. UIMA MDIP for CFO = 4583001999 is performed in the centralised component of the payment system of the Bank of Russia.

Participants of electronic exchange and external ASs operating directly with the SMDIP centralised component of the payment system of the Bank of Russia indicate UIMA MDIP for CFO = 4583001999 in the element **props:To** of the transport envelope.

3.3.2.2 Identifier of a service envelope and the reference identifier of a service envelope

The identifier of a service envelope **MUST** be generated by the application of the EM sender in a special format.

The identifier of a service envelope shall meet the requirement of spatial-temporal uniqueness.

The identifier of a service envelope **MUST** be executed in the following format:

"<prefix>:<value>", where:

<prefix> is the prefix of the identifier of the service envelope;

<value> is the identifier value.

If the identifier is a 128-byte Globally Unique Identifier (GUID), a prefix value shall be 'guid', and the identifier value shall be in hexBinary format

Note: hexBinary format represents each binary octet (8 bit) as a character sequence of two hexadecimal digits (0–9, A, B, C, D, E, F).

If the identifier is a message file name, the prefix value shall be 'file', and the identifier value shall be a file name. If the element **props:LegacyTransportFileName** is generated, the value of this element MUST coincide with the value of the identifier of the service envelope.

The reference identifier SHOULD be generated when creating a response message or an acknowledgement message. The reference identifier of a service envelope is generated by the transport software (when generating acknowledgements for an initial message) or by the software transmitting a message for delivery.

When generating a response message or an acknowledgement message for a message with the filled-in field props:MessageID, the identifier of the service envelope of the initial message MUST be specified in the reference identifier. When generating an acknowledgement for a message in an inherited format, as the reference identifier, the file name recorded in the initial message is used.

3.3.2.3 Application identifier

The application identifier MAY be generated only by the application of the EM sender in a special format.

This element allows the BoR BU to monitor the passage of the EM through the transport level from the application level. This element is optional.

The application identifier MUST have the following format:

"<prefix>:<value>", where:

<prefix> is the prefix of the message identifier. If the identifier is a Globally Unique Identifier (GUID), the prefix value shall be 'guid';

<value> is an identifier value in hexBinary format.

The requirements for the format of this identifier, the rules for its generation, and the indication of the necessity of its generation are:

- presented by the developers of the accounting and operational system operated by the payment system of the Bank of Russia;
- communicated to an exchange participant of the BoR RD to which this participant is connected.

3.3.2.4 Message type

This element MUST be filled in by the author of the service envelope. The element MUST contain one of the following values:

'1' = payment messages;

'2' = for exchange of information data (incl. statistical reporting);

'3' = for technological acknowledgement messages (within the framework of transport systems);

'5' = other service messages.

3.3.2.5 Message priority

The value of this element MUST be within the range: from 0 (the least) to 9 (the highest).

If additional requirements for the priority value are not determined, the value '5' shall be used by default.

3.3.2.6 File name for inherited transport

If inherited transport is targeted at the use of service information encoded in a file name that is being transmitted, then the application MUST contain a file name in the element **props:LegacyTransportFileName** of the service envelope that is being transmitted.

The transport adapter of inherited transport on the recipient's side, before placing a file in the transport layer, MUST assign it a name with a value of the element **props:LegacyTransportFileName** that ensures compatibility with the inherited transport.

The length of the file name shall not exceed 95 characters. This is necessary so that the file name is the value of the identifier of the service envelope.

3.3.2.7 Request for acknowledgements

The EM sender MAY request an acknowledgement of the message being sent. If the message is successfully processed, an acknowledgement of the positive result is generated.

Acknowledgements about error situations are generated and transmitted to the sender of the initial message irrespective of the value of this attribute.

The absence of the optional component **props:AckRequest** also means a waiver of acknowledgements.

3.3.2.8 Attribute of receipt of a message on removable media.

If an EM was received from an EM author on removable media, the application transmitting the EM for delivery or the transport software MUST place the value '1' in the element **props:ReceivedOnExternalMedium** of the service envelope of the message being sent.

3.4 Block of the header for specifying the sequence (SequenceInfo)

To secure the sequence of transmission of payment documents between customers of the Bank of Russia and the payment system of the Bank of Russia, the information block **props:SequenceInfo** enabling monitoring of the order of priority during a current business day (see table 4) MAY be added in the header of the service envelope.

3.4.1 Element composition

Namespace

"urn:cbr-ru:msg:props:v1.3" (the prefix props)

Table 4. Elements of the information block MessageInfo (props:SequenceInfo)

Element description	Element type	Multiplicity
1 Sequential number of a message during the business day. The element is generated and filled in by the author of a service envelope transmitting a message for delivery (<i>props:SequenceNumber</i>)	Number. Number of an electronic message. Integer, from 1 to 1,000,000,000.	[1]
2. Date of message generation. The element is generated and filled in by the author of a service envelope transmitting a message for delivery (<i>props:SequenceDate</i>)	Date [GOST ISO 8601-2001]. Format YYYY-MM-DD. GMT.	[1]
3 Unique identifier of the EM author. The element is generated and filled in by the author of a service envelope transmitting a message for delivery. The element is reserved for subsequent use; it is not used in this version (<i>props:SequenceUIC</i>)	Identifier. Unique identifier of the EM author; UIMA. [UIMA]. Numeric, 10-digit.	[0..1]

3.4.2 Rules for filling in

The values of these elements MUST be filled in by the application of a customer of the Bank of Russia.

The sequential number of the message **props:SequenceNumber** MUST be in the range from 1 to 1,000,000,000.

The element **props:SequenceUIC** is optional: it is not used at present but is planned for possible future extensions. The subscriber is identified by a unique logical address. This identification is necessary for monitoring the sequence of a message within the flow of messages of each customer.

3.5 Identification information block

The identification information block of the header of a service envelope (props:DocInfo) contains information identifying an application message transmitted in a service envelope. This information is intended for analysis in an application that does not have access to the content of the message that is being transmitted. The identification information block MAY add an application that generates a service envelope to the header. For messages of the payment system transmitted using the header described herein, this block MUST be filled in.

3.5.1 Element composition

Namespace

"urn:cbr-ru:msg:props:v1.3" (the prefix props)

Table 5. Elements of the identification information block DocInfo (props:DocInfo)

Element description	Element type	Multiplicity
1 Message format. The element is generated and filled in by the author of a service envelope transmitting a message for delivery (props:DocFormat)	Code. Designation of the format of the message to be transmitted. (1 = UFEBM, 2 = internal format of SSP AOS, 3 = files of BoR FD, 9 = other). Numeric, 1-digit.	[1]
2 Message type. The element is generated and filled in by the author of a service envelope transmitting a message for delivery (props:DocType)	Identifier. Message type (EDnnn, PacketEPD, PacketESID, MTnnn, etc.). Text, up to 100 characters	[0..1]
3 Digit-based group of elements of a UFEBM message. To be filled in if the element 'Message format' matches UFEBM (props:DocFormat = '1'). (props:EDRefID)		[0..1]
EM number during the business day (props:EDNo)	Number. Number of an electronic message. Integer, up to 9 places.	[1]
EM composition date (props:EDDate)	Date. [GOST ISO 8601-2001]. Format CCYY-MM-DD.	[1]
Unique identifier of the EM author – UIMA (props:EDAuthor)	Identifier. Unique identifier of the EM author; UIMA. [UIMA]. Numeric, 10-digit.	[1]
4 Identifier of the message to be transmitted. To be filled in for messages in the format SSP AOS (props:DocID)	Identifier. Unique identifier of the message in the format of the SSP AOS. Textual, up to 100 characters.	[0..1]

3.5.2 Rules for filling in

3.5.2.1 Message format

This element defines the format of a message contained in a service envelope.

This element MUST be filled in by the author of the message. The element MUST contain one of the following values:

'1' = the message matches the format defined by the UFEBM Album;

'2' = a message in the internal exchange format of the processing system of the SSP AOS;

'3' = the message contains a file of a BoR FD;

'9' = contains other data.

3.5.2.2 Message type

This element MAY be filled in by the author of the service envelope and contains a conventional designation of the message to be transmitted.

For messages in the UFEBM format, the element MUST contain the textual designation of a message type from the UFEBM Album: 'ED101', 'PacketEPD', etc.

For messages in the internal exchange format of the SSP AOS, the element MUST contain the textual designation of an EM type accepted in the applicable SSP: 'MT125', 'MT998', etc.

3.5.2.3 Digit-based element group

This group of elements contains information making it possible to unambiguously identify an EM in the UFEBM format and containing the elements:

'EM number during the business day (sequential number of the EM)', unique for each participant or BoR BU (VTs) during the day;

'EM composition date';

'Unique identifier of the EM author'.

This group of elements MUST be filled in by the message author if the message format matches UFEBM—that is, props:DocFormat = '1'.

3.5.2.4 Identifier of the message to be transmitted.

The element contains the unique identifier of a message to be transmitted in the service envelope. This element MUST be filled in by the author if the message is in the internal exchange format of the processing system of the SSP AOS (props:DocFormat = '2').

3.6 Acknowledgement

Acknowledgements are intended for confirmation by the transport subsystem of the phases of an electronic message passing from a participant to the BoR BU and the respective monitoring of the EM status by the participant. The acknowledgement message is sent to the actual address specified in the transport message.

The Acknowledgement is a service envelope with an empty envelope body (**env:Body**) and a header (**env:Header**) consisting of an information block (**props:MessageInfo**) and an additional block (**props:AcknowledgementInfo**). The element composition of the additional block of an acknowledgement is presented in the table below (see table 5).

Acknowledgements are optional.

3.6.1 Element composition

Namespace

"urn:cbr-ru:msg:props:v1.3" (the prefix props)

Table 5. Elements of the transport acknowledgement block AcknowledgementInfo (props:AcknowledgementInfo)

Element description	Element type	Multiplicity
1 Acknowledgement type. The element is generated and filled in by the acknowledgement author (props:AcknowledgementType)	Code. Acknowledgement type. (1 = sending, 2 = receipt in the responsibility area of a BoR BU or a participant, 3 = reading by the receiving application).	[1]
2 result of the transport transaction (sending, receipt, acceptance, etc.). The element is generated and filled in by the acknowledgement author. (props:ResultCode)	Code. Transaction result. Numeric, 4-digit.	[1]

Element description	Element type	Multiplicity
3 Description of the transaction result. The element is generated and filled in by the acknowledgement author (<i>props:ResultText</i>)	Text. String, up to 3000 characters.	[0..1]

3.6.2 Rules for filling in

3.6.2.1 Special considerations of filling in the elements of the information block of the header

The address information **MUST** be filled in with actual values determining the direction of transmission from the author of the acknowledgement to the sender of the initial message.

The identifier of the service envelope (**props:MessageID**) **MUST** be filled in with a new, unique value. The reference identifier (**props:CorrelationMessageID**) **MUST** be filled in pursuant to Clause 3.3.2.2.

The message type (**props:MessageType**) **MUST** be equal to '3'.

The priority (**props:Priority**) of an acknowledgement message **MUST** be equal to the priority of the initial message.

Elements containing information on the date and time **MUST** be filled in with the actual values for the acknowledgement message.

The value of the element **props:CreateTime** of an acknowledgement message **MUST** coincide with the value of the element **props:ReceiveTime** of the initial message that is recorded when the initial message passes through the Transport Gateway of SVK.

3.6.2.2 Acknowledgement type

This element **MUST** be filled in by the author of an acknowledgement. The element **MUST** contain one of the following values:

'1' = sending (to be generated by the transport adapter as the result of placing a message in the transport layer);

'2' = receipt in the responsibility area of a BoR BU or a participant (for a BoR BU, to be generated in the Transport Gateway of SVK);

'3' = reading by the receiving application (to be generated by the receiving application as acknowledgment of acceptance of a message for processing).

3.6.2.3 Result of the transport transaction

The element **MUST** contain a transaction code. In the case of success—'0000'; otherwise, the code of the error arising when making a transaction. Transaction codes are defined in the documentation of systems generating acknowledgement messages.

3.6.2.4 Description of the transaction result

The element **MAY** contain the transaction result in a test form. This element is intended for analysis by a human and **MUST** contain a readable text in Russian.

4. Description of a service envelope through XML schemas

A description of files with XML schemas is given in the table below (see table 6).

Table 6. Description of files with XML schemas

Schema file name	Target space of schema names (targetNamespace)	Description of the schema
cbr_msg_props_v2017.2.0.xsd	"urn:cbr-ru:msg:props:v1.3"	Blocks of the header of a service envelope: Message characteristics
soap-envelope.xsd [*]	"http://www.w3.org/2003/05/soap-envelope"	Service envelope. To be created pursuant to the recommendation of SOAP12
xml.xsd [†]	"http://www.w3.org/XML/1998/namespace"	Definition of the attributes xml:base, xml:lang, xml:space. Used in SOAP12

^{*} The XML schema has been published on the Internet on the W3C website <http://www.w3.org/2003/05/soap-envelope/>

[†] The XML schema has been published on the Internet on the W3C website <http://www.w3.org/2001/03/xml.xsd>