



eps e-payment standard
STANDARD IMPLEMENTATION
GUIDELINE
Version 2.6.1

Date: 06. August 2018
Author: Joachim Geisler / STUZZA
Mag. Alexander Schilling / STUZZA
Martin Pass / STUZZA
Matthias Kautz / ARZ

Content

DOCUMENT HISTORY	5
PREAMBLE	6
1 INTRODUCTION	7
1.1 EPS E-PAYMENT STANDARD	7
1.2 DEFINITIONS	7
1.3 REFERENCES	8
2 THE EPS ONLINE-ÜBERWEISUNG LOGO	9
3 ORGANIZATIONAL PREREQUISITS	10
3.1 ESERVICE SCHEME OPERATOR (SO)	10
3.2 MERCHANT-CONTRACT	10
3.3 SELECTION OF EPS DEBTOR BANKS	10
4 TECHNICAL REQUIREMENTS	11
4.1 XML SPECIFICATION	11
4.2 EPS XML NAMESPACE	11
4.3 CHARACTERSET OF EPS XML SCHEMA	11
4.4 TRANSFER PROTOCOL	12
4.5 TIME OUT	12
4.6 URL SPECIFICATION	12
4.7 URL ENCODING	12
4.8 SECURITY	13
4.9 MANDATORY DATA ELEMENTS	13
4.10 ERROR HANDLING	13
5 STRUCTURE EPS STANDARD	17
6 SPECIFICATION EPS XML SCHEMA & MESSAGES	18
6.1 ECBS_EPI SCHEMA	18
6.2 EPSPAYMENT SCHEMA	18
6.2.1 <i>PaymentInitiatorDetails: Payment Request</i>	18
6.2.1.1 epi:EpiDetails	19
6.2.1.1.1 epi:IdentificationDetails	20
6.2.1.1.2 epi:PartyDetails	23
6.2.1.1.3 epi:PaymentInstructionDetails	25
6.2.1.2 atrul:AustrianRulesDetails	28
6.2.2 <i>PaymentConfirmationDetails - eps payment confirmation</i>	29
6.2.2.1 epi:RemittanceIdentifier	31
6.2.2.2 epi:UnstructuredRemittanceIdentifier	31
6.2.2.3 eps:PaymentInitiatorDetails	31
6.2.2.4 eps:PayConApprovingUnitDetails	31
6.2.2.5 eps:PayConApprovalTime	32
6.2.2.6 eps:PaymentReferenceIdentifier	32

6.2.2.7	eps:StatusCode	32
6.2.2.8	dsig:Signature	34
6.2.3	<i>ShopConfirmationDetails - merchant confirmation</i>	34
6.3	AUSTRIANRULES SCHEMA.....	35
6.3.1	<i>atrul:Realization</i>	35
6.3.2	<i>atrul:PaymentDescription</i>	36
6.3.3	<i>atrul:TradeCategory</i>	36
6.3.4	<i>atrul:DigSig</i>	36
6.3.5	<i>atrul:ExpirationTime</i>	37
6.4	MESSAGE: EPS PAYMENT INITIATION	38
6.5	MESSAGE: BANK REPOSE.....	46
6.6	MESSAGE: VITALIY-CHECK	48
6.7	MESSAGE: EPS PAYMENT CONFIRMATION	49
6.8	MESSAGE: MERCHANT CONFIRMATION TO PAYMENT CONFIRMATION MESSAGE	51
7	EPS WORKFLOW	63
7.1	EPS WORKFLOW WITHOUT CENTRAL SO DEBTOR BANK SELECTION	63
7.1.1	<i>Step I-1: debtor bank selection</i>	64
7.1.2	<i>Step I-2: eps payment initiation merchant-SO</i>	65
7.1.3	<i>Step I-3: eps payment initiation SO-debtor bank</i>	65
7.1.4	<i>Step I-4: Bank response debtor bank-SO</i>	65
7.1.5	<i>Step I-5: bank response SO-merchant</i>	66
7.1.6	<i>Step I-6: Redirect Online-Banking</i>	67
7.1.7	<i>Step III-1: payment& vitality check</i>	67
7.1.8	<i>Step III-2: Vitality Check SO-merchant</i>	68
7.1.9	<i>Step III-3: confirmation Vitality Check merchant-SO</i>	68
7.1.10	<i>Step III-4: Vitaliy Check SO-debtor bank</i>	69
7.1.11	<i>Step III-6: eps confirmation debtor bank-SO</i>	69
7.1.12	<i>Step III-7: eps confirmation SO-merchant</i>	69
7.1.13	<i>Step III-8: confirmation of eps confirmation merchant-SO</i>	70
7.1.13.1	eps confirmation is validated by the merchant	71
7.1.13.2	eps confirmation can not be validated by the merchant.....	71
7.1.14	<i>Step III-9: confirmation eps confirmation SO-debtor bank</i>	71
7.1.15	<i>Step III-10: Buyer redirect to merchant</i>	71
7.1.15.1	correct workflow	72
7.1.15.2	Incorrect workflow	72
7.1.16	<i>Overview possible redirect options</i>	73
7.2	EPS WORKFLOW WITH CENTRAL DEBTOR BANK SELECTION	74
7.2.1	<i>Step II-1: selection of payment method</i>	76
7.2.2	<i>Step II-2: eps payment initiation merchant-SO</i>	76
7.2.3	<i>Step II-3: Bank response SO-merchant</i>	76
7.2.4	<i>Step II-4: Buyer Redirect to SO debtor bank selection service</i>	77
7.2.5	<i>Step II-5: eps payment initiation SO-debtor bank</i>	77
7.2.6	<i>Step II-6: bank response debtor bank-SO</i>	77
7.2.7	<i>Step II-7: buyer rediret to the debtor bank</i>	77

8 TABLE OF FIGURES..... 95

DOCUMENT HISTORY

Version 2.6.1, August 2018	
Matthias Kautz, ARZ	
Change	note
Implementation Guideline	
<ul style="list-style-type: none"> • some linguistical changes 	
<ul style="list-style-type: none"> • exchange of some URL 	due to changes to the STUZZA web-sites – eservice.stuzza.at
<ul style="list-style-type: none"> • chapter 1.3 • chapter 3.3 • chapter 7.1.1 	added description of the bank-selection-widget
UPDATE 26.5.2014	
Chapter 6.4: Update epsp:AuthenticationDetails - merchant identification - MD5 Fingerprint documentation	Update MD5 Fingerprint specification
Implementation Guideline	
<ul style="list-style-type: none"> • New XML Schema and Namespace for IGL V25 • Update ePI Schema: new dataelement "UnstructuredRemittanceIdentifier" (max140 Text) • Update eps messages due to new XML Schema • Update debtor bank selection service • time indication (expiration time) for minimum and maximum period to process an eps transaction: new dataelement in Austrian Rules Schema • New Statuscode UNKNOWN (SO ⇔ merchant) • Update and information on Error Handling • debtor cancellation before login to online-banking: always to send eps confirmation message to merchant with status NOK • PaymentReferenceIdentifier (unique payment reference assigned by the bank): change to mandatory dataelement in eps confirmation message 	

PREAMBLE

The eps Standard Implementation Guideline describes the content and technical structure of the eps e-payment standard.

Since eps Version 2.4 the entire eps workflow has to be routed over the Scheme Operator (SO).

The eps e-payment standard is an open and public standard without licence fees. Relevant documents and specifications, e.g. implementation guideline, XML Schemas, are available under <https://eservice.stuzza.at/en>.

To use eps as a merchant an agreement has to be closed with an eps-bank or an acquirer.

The eps-payment standard is based on [ePI](#) (electronic Payment Initiator) which was published by ECBS (European Committee for Banking Standards, which now is the European Payments Council [EPC](#)) in September 2002.

By choosing ePI as eps-payment-standard the Austrian banks implemented a standard which is fully SEPA compliant and uses IBAN and BIC mandatory. Therefore eps fulfills the EU-criteria according to EU regulation 2560/2001/EC.

1 INTRODUCTION

1.1 eps e-payment standard

The eps e-payment standard has been developed by STUZZA together with Austrian banks, the Ministry of Finance (BMF) and the CIO (Chief Information Office, <http://www.cio.gv.at>) of the Austrian Government.

The eps e-payment standard is an open XML standard for electronic online payments, but not a product!

The eps e-payment standard allows the buyer/debtor to make online payments directly in the online-banking of the debtor (buyer) bank.

The big advantage of the eps standard is the buyers confidence in and highest security standards of the online-banking application offered by the debtor bank.

In the entire eps workflow no private or payment related data and information of the buyer/debtor is requested, provided to or stored by any third party, nor is it forwarded in any eps XML messages to the merchant.

The buyer as account holder always authenticates himself directly at the online-banking of the debtor bank with known security/authentication mechanisms, e.g. ID/PIN or by digital signature.

The standard can be implemented by merchants as well as E-Government/public administration and is supported by eps banks.

1.2 Definitions

- merchant: web shop, E-Government/public administration
- buyer/customer: customer who buys/orders goods or services in the web shop and/or E-Government application
- eps bank: bank or payment institution offering an online payment system based on the eps e-payment standard
- eps acquirer: contracting partner for eps merchant contract
- creditor bank: bank or payment institution (account holder of merchant) signing the eps merchant contract with the merchant
- debtor bank: bank or payment institution (account holder of buyer/debtor) offering the eps standard in its online-banking application to its customers (the buyer)
- eService Scheme Operator (SO): central routing service in eps workflow
- eps message: XML message as documented in eps implementation guideline
- BEI: SWIFT Business Entity Identifier, may be used to identify the beneficiary
- Beneficiary: merchant (e.g. web shop, E-Government). The beneficiary is the ultimate party to be credited as a result of a funds transfer. The beneficiary gener-

ates an eps XML Payment Request which is then forwarded via the SO to the debtor bank.

- Beneficiary's Financial Institution (BFI): Beneficiary's bank which holds the account of the beneficiary
- BIC: ISO Business Identifier Code (ISO 9362), online information at <http://www.swift.com>, BIC Directory Online
- Guaranteed Payment: eps payment with guaranteed payment option: an eps debtor bank undertakes the guarantee for the credit transfer with status "OK" in the eps payment confirmation message; this XML eps payment confirmation message is always created by the debtor bank.
- IBAN (International Bank Account Number, ISO 13616): Composition: ISO-Country code, 2 digit check-digit, bank code and account number. The length of the Austrian IBAN is 20 digits (possible IBAN length is up to 34 digits).
- Ordering Customers Financial Institution (OFI): The bank of the ordering customer holding the account of the buyer (ordering customer)/debtor bank.
- Online-Banking: term for electronic payment applications, e.g. internet banking
- Ordering Customer: the buyer
- eService legal framework: agreement for eServices provided by the SO
- XMLDSig: W3C Standard (<http://www.w3c.org>), specifies and describes the rules of the digital signature in XML Syntax.

1.3 References

Document	Version, Date
ECBS ePI Standard Document	EBS602 Version 1.1, September 2003
ECBS ePI SIG Document	SIG605 Version 1.1, November 2003
Signature profile (Signaturprofil) E-Government	Version 1.0, 19.03.2004
eps bank-selection-widget	Version 1.0, 27.08.2015
e-Services technical parameters	Version 1.5_englisch

2 eps-ÜBERWEISUNG LOGO

The eps e-payment standard is the basis for the online payment product **eps-Überweisung** for secure internet payments.



The logo can be downloaded at <http://eservice.stuzza.at/en>.

3 ORGANIZATIONAL PREREQUISITS

3.1 eService Scheme Operator (SO)

All eps XML messages have to be routed via the central Scheme Operator (SO).

The SO can be addressed with the central Routing URL "**https://routing.eps.or.at**".

In case a creditor bank is using a special Routing URL, it's under the creditor bank's responsibility to inform the merchant accordingly.

3.2 Legal contract

If an eps bank wants to offer eServices it must enter into an eServices legal framework agreement with STUZZA GmbH as SO.

In order to provide eServices "eps Online-Überweisung" an eps bank must approve the product sheet eps Online-Überweisung in the legal framework.

3.3 eps bank registration at SO

After signing the legal contract the eps bank will be registered at the Scheme Operator, e.g. testing or productive stage.

The Scheme Operator registers an Admin User, authentication details will be forwarded by the SO.

3.4 merchant-contract

In order to offer the eps-Überweisung a merchant has to conclude and sign the merchant legal contract.

3.5 Selection of eps debtor banks

The merchant has to ensure that the buyer can select his debtor bank (the account holding bank of the buyer) in order to start the eps payment method eps-Überweisung.

The following technical options are available and supported:

- bank selection directly in the merchants webshop (see chapter [7.1](#))
- central debtor bank selection service offered by the SO (see chapter [7.2](#))
- usage of the bank-selection-widget (see chapter 7.1 and document eps bank-selection-widget_E_v1.0_20160308.pdf)

4 TECHNICAL REQUIREMENTS

4.1 XML Specification

The eps XML structure is specified as XML Schema in accordance with W3C Standards.

All eps XML messages must be UTF-8 encoded.

Data coding must use content type:text/xml.

4.2 eps XML Namespace

Each eps XML Schema has a unique XML namespace.

Example of XML Header eps message with new namespace eps XML Schema V26:

```
<?xml version="1.0" encoding="UTF-8"?>
<epsp:EpsProtocolDetails SessionLanguage="DE"
xsi:schemaLocation="http://www.stuzza.at/namespaces/eps/protocol/2014/10 EPSPProtocol-V26.xsd"
xmlns:atrul="http://www.stuzza.at/namespaces/eps/austrianrules/2014/10"
xmlns:epi="http://www.stuzza.at/namespaces/eps/epi/2013/02"
xmlns:eps="http://www.stuzza.at/namespaces/eps/payment/2014/10"
xmlns:epsp="http://www.stuzza.at/namespaces/eps/protocol/2014/10"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
```

The latest eps XML Schema are available at

<http://www.stuzza.at/namespaces/eps/protocol/2014/10/> and have to be supported by the merchant!

4.3 Characterset of eps XML Schema

The eps XML Schema has some restrictions to supported characters (note: within the eps XML Schema the used and supported character set is documented in the XML pattern of the data elements).

Special care for remittance information <epi:RemittanceIdentifier> as of eps XML Schema Version 2.4!

Supported restricted character set ⇔

a b c d e f g h i j k l m n o p q r s t u v w x y z
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
0 1 2 3 4 5 6 7 8 9
/ - ? : () . , ' +
Space

For e.g. name/address the following extended characters - as supported in Austria - can be used:

Supported extended character set ⇨

a b c d e f g h i j k l m n o p q r s t u v w x y z
 A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
 0 1 2 3 4 5 6 7 8 9
 Ä Ö Ü ä ö ü ß

Character	-	€	\$	§	%	!	=	#	~	;	+	/	?	:	()	.	,	'
XML Entity																			

Character		&	>	<	"		*	{	}	[]	@	\	_	°	^
XML Entity	 	&	>	<	"											

4.4 Transfer protocol

The entire Message exchange must be via HTTP or HTTPS; under no circumstances via FTP transfer.

4.5 Time Out

For security reasons specific and individual time out sessions are requested by the SO.

Time out sessions (connection/socket time-out) can be found at <http://eservice.stuzza.at/en>.

4.6 URL Specification

URL information has to be completely transferred to the SO, e.g. <http://www.anyurl.com:8080/shop/kasse>.

Please note that the URL length is limited to 512 characters!

The vendor-Login-URL for each merchant to SO communication is always the central SO-routing URL (see chapter 3.1).

Further information (e.g. URL for SO test environment) can be found in an individual eps bank supplement.

4.7 URL Encoding

URLs sent in the XML message have to be URL-encoded by the merchant; transfer via XML must be XML-encoded.

Example:

Redirect-URL: <http://www.haendler.at/transactionok>

Parameter 1: param1=value1

Parameter 2: param2=temp&xxx

Parameter 3: param3=Händler

⇒ URL-encoded URL:

<http://www.haendler.at/transactionok?param1=value1¶m2=temp%26xxx¶m3=H%E4ndler>

⇒ XML-encoded URL-encoded URL:

<http://www.haendler.at/transactionok?param1=value1¶m2=temp%26xxx¶m3=H%E4ndler>

4.8 Security

Communication to the SO (merchant-SO) and further SO to debtor bank is exclusively over HTTPS. We recommend securing the communication between the buyer and the merchant with a SSL certificate.

Communication from the SO to the merchant (vitality check and payment confirmation messages) depends on the merchant, which is evident from the ConfirmationUrl conveyed by the merchant (e.g. <https://www.abc.aa> for SSL coding).

Depending on the security status the SO either sends a reduced payment confirmation message (no SSL encoding) or with merchant SSL coding the original payment request together with the payment confirmation message (see chapter [6.2.2](#)).

4.9 Mandatory data elements

All defined mandatory data elements described in the standard implementation guideline must be present with content in the XML eps message (even if they are not used within the Austrian payments-process).

4.10 Error Handling

Please note: Error Code and Error Information (ErrorMsg) within an eps message!

To be used in an eps message, e.g. epsp:BankResponseDetails, and forwarded to the merchant, see chapter [7.1.4](#)

Existing error codes and error details:

ErrorCode	ErrorMsg
000	No error
002	Confirmation URL incorrect/wrong
003	Currency incorrect/wrong
004	Identification incorrect/wrong
007	Error in XML-Stream
008	Unknown error

009	internal error
010	IBAN invalid
011	BIC invalid
012	Expiration Time or execution time invalid
013	No cross-border transactions allowed
014	Connection timeout to bank or service at online-banking
015	Bank selection invalid
020	Transaction-ID does not exist
021	Transaction not completed

ErrorMessages from the SO are indicated with the prefix „SO:“!

Example:

```
<epsp:ErrorCode>008</epsp:ErrorCode><epsp:ErrorMsg>SO: Unbekannter Fehler:integration error (no info to merchant):integrationWinName</epsp:ErrorMsg>
```

ErrorCode	ErrorMsg
002	SO: confirmationUrl is incorrect/wrong ⇒ in the confirmation URL a local networkaddress or a relative server address is used
004	SO: identification is wrong/incorrect ⇒ merchant deleted/disabled/not found, signature invalid or checksum incorrect/invalid
007	SO: error in XML-Stream ⇒ Error while parsing the XML or invalid content (content type, date specification code, etc.)
008	SO: unknown error ⇒ http timeout, invalid http status sode, BIC or bankgroupID not found, invalid execution date, payment option at merchnt not set, incorrect call of central bank selection, wrong implementation of window management, etc.
009	SO: internal error ⇒ general system error
010	SO: IBAN invalid ⇒ payment initiaiton with unregistered merchant IBAN
011	SO: BIC invalid ⇒ payment initiaiton with unregistered BIC
012	SO: Timeout transaction ⇒ expiration time or execution time expired or invalid
014	SO: Connection timeout ⇒ debtor bank/online-banking not reachable due to maintenance work
015	SO: Bank selection invalid ⇒ the pre-selection of the bank by the merchant is not allowed
020	SO: Transaktions- ID does not exist ⇒ An unknown transaction ID was transmitted in the TransactionDetailsRequest or ConfirmationStatusRequest.
021	SO: Transaction not completed ⇒ At the time of the ConfirmationStatusReques, the transaction was not yet completed.

additional error handling at buyers redirect:

the debtor bank complements the submitted epsp: TransactionNokUrl with a respective ERROR code, e.g. status code ERROR1 in epsp: TransactionNokUrl as a parameter, e.g. <https://testbeguenstiger.at?epserrorcode=ERROR1>

More information in chapter [7.1.15.2](#).

SO specific error handling/information

Mapping eps Vitality-Check

Vitality-Check from Debtor bank	Vitality-Check to merchant
XML corrupt	No forward to merchant! Info to debtor bank: HTTP-Status 412

Mapping eps confirmation

Payment Initiation	confirmation-Url merchant	confirmation from debtor bank	confirmation to merchant
SIG (atrul) not set	https	Signatur valid	digitally signed confirmation msg by bank
SIG (atrul) not set	http	Signatur valid	Unsigned/no signature
SIG (atrul) is set	https	Signatur valid	digitally signed confirmation msg by bank
SIG (atrul) is set	http	Signatur valid	Digitally signed by SO
*	https	Signatur invalid	Incorrect digitally signed eps confirmation message by debtor bank
*	http	Signatur invalid	none, HTTP -Status 412 to debtor bank
*	*	No signature	none, HTTP -Status 412 to debtor bank
*	*	ApprovingUnitIdentifier not allocated	none, HTTP -Status 412 to debtor bank
*	*	XML corrupt	none, HTTP -Status 412 to debtor bank

Mapping merchant response (Vitality-Check and eps confirmation)

Response from merchant	Response to debtor bank
XML corrupt	none, HTTP -Status 412 to debtor bank
Timeout	none, HTTP -Status 412 to debtor bank

missing BIC or other bank identification eps confirmation message

message from debtor bank	Message an Händler
XML incorrect: wrong or missing BIC or other bank identificationof the debtor bank in Dataelement "ApprovingBankIdentifier"	No forward to merchant! Info to debtor bank: HTTP-Status 412

5 STRUCTURE eps XML STANDARD

The eps e-payment Standard consists of several XML data containers/schemas which all have different XML Namespaces.

EPSProtocol-V26.xsd: main XML schema, responsible for the structure of the eps XML messages;

Namespace: <http://www.stuzza.at/namespaces/eps/payment/2014/10>

Prefix: epsp

It contains the following XML schemes:

1. **ECBS_ePI_V12.xsd**, ECBS ePI Standard: specifies the ePI payment request
Namespace: <http://www.stuzza.at/namespaces/eps/epi/2013/02>
Prefix: epi
2. **EPSPayment-V26.xsd:** specifies eps messages payment request (PaymentInitiatorDetails), payment confirmation (PaymentConfirmationDetails) and the shop confirmation message (ShopConfirmationDetails) once a payment confirmation is received. Inherited in EPSProtocol-V26.xsd data container
Namespace: <http://www.stuzza.at/namespaces/eps/payment/2014/10>
Prefix: eps
3. **AustrianRules-V26.xsd:** inherited in EPSPayment-V26.xsd data container
Namespace: <http://www.stuzza.at/namespaces/eps/austrianrules/2014/10>
Prefix: atrul
4. **W3C-XMLDSig.xsd:** W3C XML schema for digital signature, inherited in EPSPayment-V25.xsd and EPSProtocol-V25.xsd data container
Namespace: <http://www.w3.org/2000/09/xmlsig#>
Prefix: dsig

The individual eps messages are based on the valid EPSProtocol scheme.

Syntax

The XML eps e-payment standard is based on W3C XML schema rules.

Maintenance

Maintenance and publication of the eps e-payment standard is under control and responsibility of STUZZA.

6 SPECIFICATION eps XML SCHEMA & MESSAGES

The following describes the data content of each eps XML Schema as well as the content/structure of individual eps messages.

6.1 ECBS_ePI Schema

The ECBS ePI Standard is integrated and described in the EPSPayment data container.

Due to SEPA guidelines the eps standard Version 2.5 has some restrictions to supported characters (note: within the eps XML Schema the used and supported character set is documented in the XML pattern of the data elements, mainly data elements in the ePI XML Schema) ⇒ see also chapter 4.3.

To ensure the correct routing of the **remittance identifier - structured** (max 35 characters) or **unstructured** (max 140 characters) - the ECBS_ePI standard (version 1.2) has been revised accordingly (see also chapter 6.2.1.1.3).

6.2 EPSPayment Schema

6.2.1 *PaymentInitiatorDetails: Payment Request*

The ePI (electronic Payment initiator) XML standard is the integral part of the eps e-payment standard.

Data elements within the ePI, which are currently not used by the Austrian payments or which might have different format specifications (e.g. length specification) are explicitly described.

The payment request message includes all relevant payment information to initiate a payment.

All defined mandatory fields and appropriate data formats must be present within the XML stream, even if they are not used in the initiation of a credit transfer.

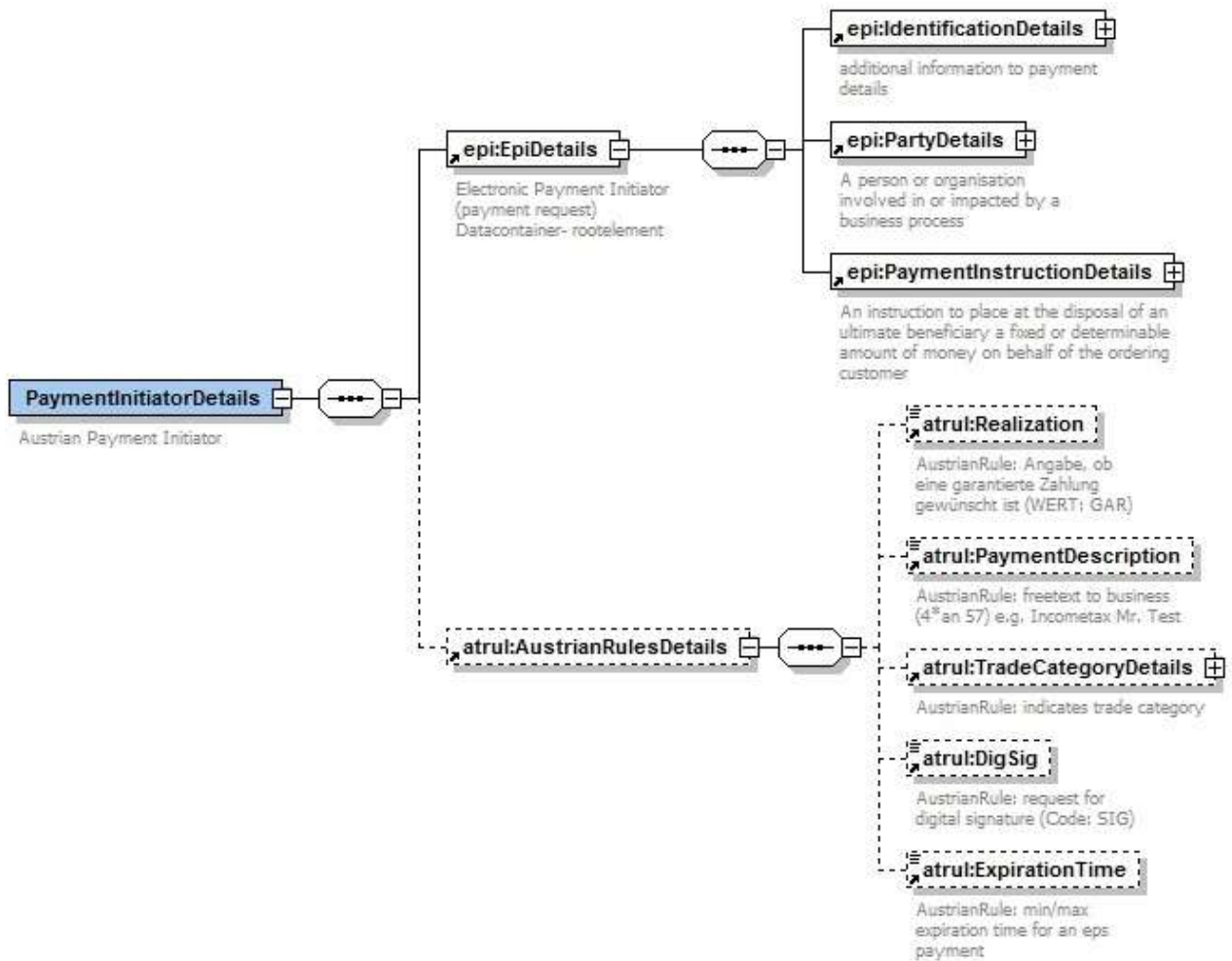


figure 6-1: PaymentInitiatorDetails

6.2.1.1 epi:EpiDetails

The ePI XML Schema is integrated into the EPSPayment Schema.

Status: MANDATORY

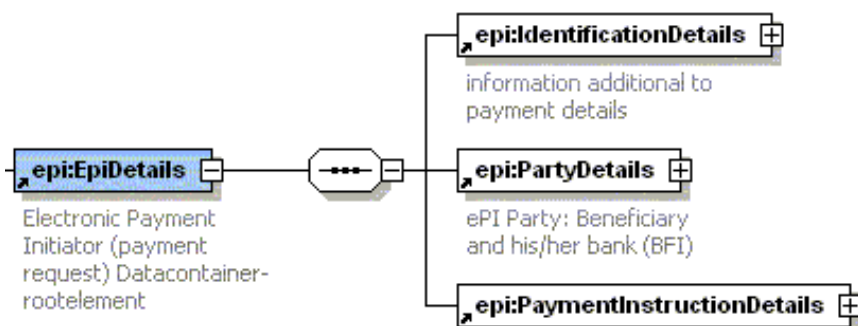


figure 6-2: epi:EpiDetails

6.2.1.1.1 epi:IdentificationDetails

Explanation: additional information to payment details; Currently not relevant in Austria

These data elements are currently not supported in online-banking applications!

Status: MANDATORY

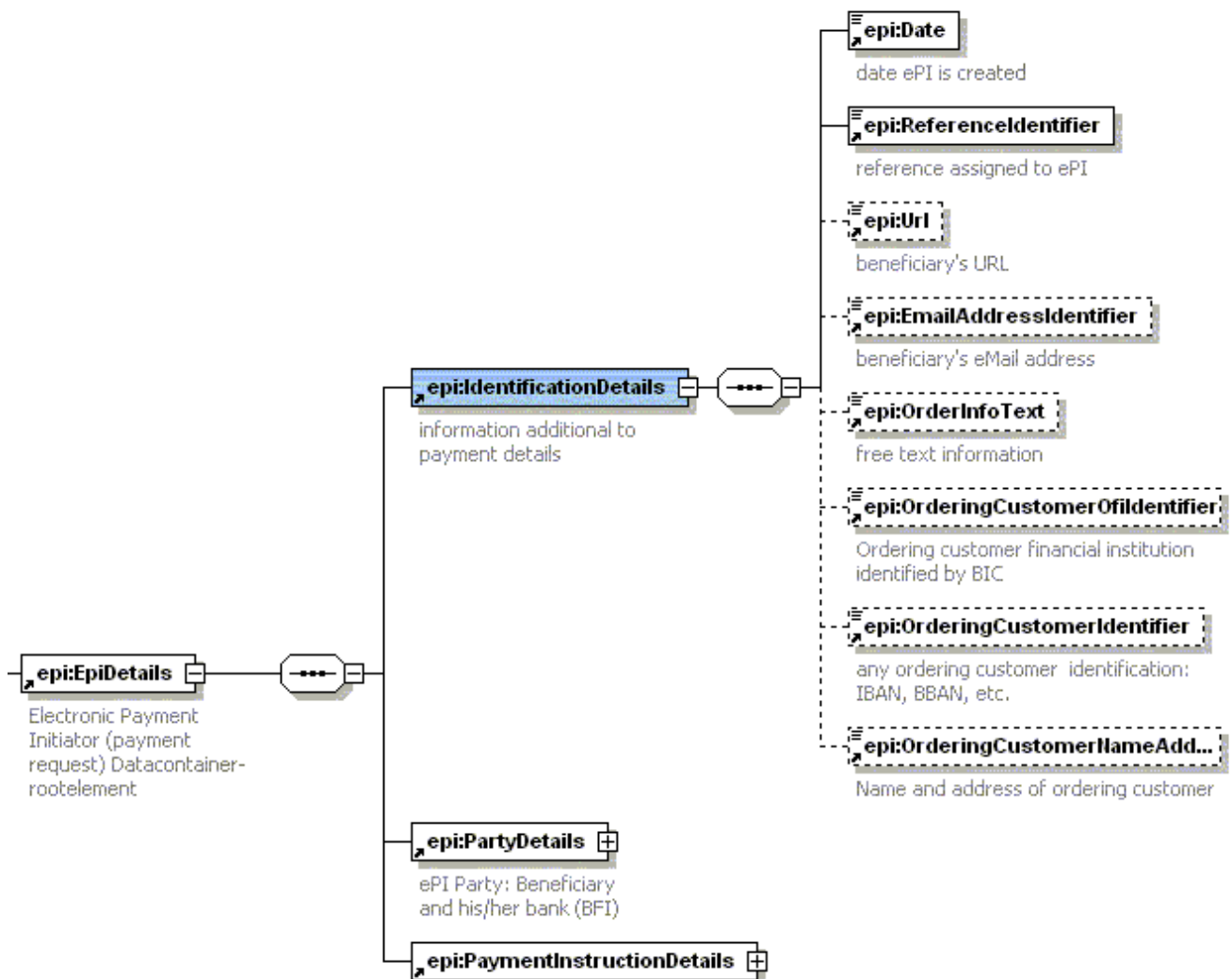


figure 6-3: epi:IdentificationDetails

- **epi:Date**

Explanation: The date on which the payment request has been created

Status: MANDATORY

Data type: xsd:date, e.g. 2013-05-02

NOTE:

Due to the ePI XML specification this data element has to be present in the XML message but it will not be processed by and forwarded in the Austrian payments system.

- **epi:ReferenceIdentifier**

Explanation: a reference assigned by the beneficiary, e.g. for reconciliation

Status: MANDATORY

Data type: an..35

Supported charset: extended charset (see chapter 4.3)

NOTE

Due to the ePI XML specification this data element has to be present in the XML message but it will not be processed by and forwarded in the Austrian payments system.

- **epi:Url**

Explanation: beneficiary's information to a URL, e.g. the website of the merchant

Status: OPTIONAL

Data type: x..512

- **epi:EmailAddressIdentifier**

Explanation: e-mail address of the e.g. beneficiary or a third party

Status: OPTIONAL

Data type: x..512

- **epi:OrderInfoText**

Explanation: unstructured free text information between the beneficiary/merchant and the buyer which will not be part of the payment instruction

Status: OPTIONAL

Data type: 5*an..70

Supported charset: extended charset (see chapter 4.3)

The following data elements could be used e.g. for routing information within additional business scenarios (e.g. EBPP).

epi:OrderingCustomerOfiIdentifier

Explanation: identification of the customer's (buyer's) financial institution by a BIC

Status: OPTIONAL

Data type: an 11

epi:OrderingCustomerIdentifier

Explanation: identification of the customer's (buyer's) account identification by e.g. IBAN, BBAN, etc.

Status: OPTIONAL

Data type: an..34

epi:OrderingCustomerNameAddressText

Explanation: Identification of ordering customer (name and/or address of the buyer) in non-coded form

Status: OPTIONAL

Data type: 4*an..35

Supported character set: extended character set (see chapter 4.3)

6.2.1.1.2 epi:PartyDetails

Explanation: payment related information of the beneficiary/merchant

Status: MANDATORY

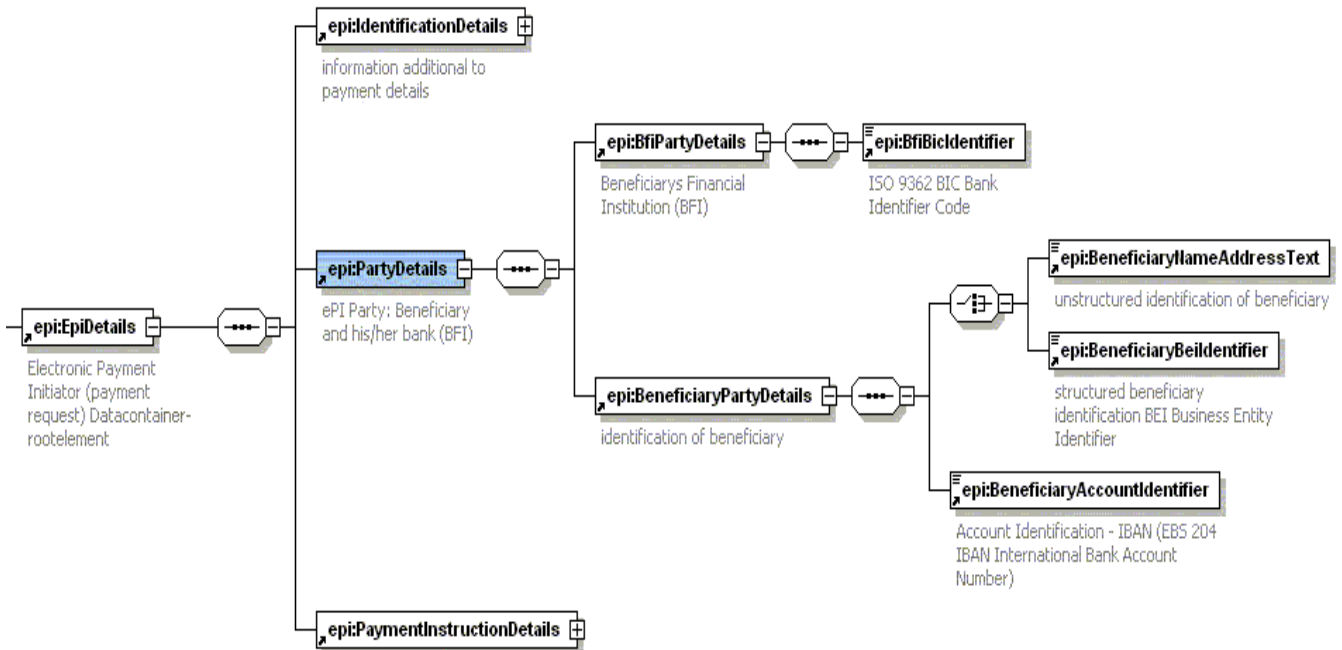


figure 6-4: epi:PartyDetails

epi:BfiPartyDetails

Explanation: Beneficiary's/merchant's financial institution

Status: MANDATORY

Note

1. the beneficiary's financial institution must be identified by its BIC.
2. the beneficiary is identified by either name/name and address or BEI and
3. identification of beneficiary's account by IBAN

- **epi:BfiBicIdentifier**

Explanation: ISO 9362 Business Identifier Code (BIC)

Status: MANDATORY

Data type: an 11

This information can not be changed by the buyer!

- **epi:BeneficiaryPartyDetails**

Explanation: beneficiary's identification either with his name and/or address or with a BEI (Business Entity Identifier, SWIFT code) code together with his account identification.

Status: MANDATORY

- **epi:BeneficiaryNameAddressText**

Explanation: identification of the beneficiary in non-coded form (name and/or address).

Status: CONDITIONAL

Data type: 4*..an35

This information can not be changed by the buyer!

ATTENTION: the Austrian online-banking only supports 2*an..35 (= max. 70) characters!

Supported charset: extended charset (see chapter 4.3)

- **epi:BeneficiaryBeiIdentifier**

Explanation: identification of the beneficiary with the Business Entity Identifier (BEI) registered to the beneficiary

Status: CONDITIONAL

Data type: an 11

This information can not be changed by the buyer!

Data element is currently not supported in online-banking applications!

- **epi: BeneficiaryAccountIdentifier**

Explanation: The unique and unambiguous identification of the beneficiary's account. **This must be an IBAN** (International Bank Account Number, ISO 13616), e.g. AT611904300234573201 (compare with Austrian account number 00234573201)

Status: MANDATORY

Data type: an..34

This information can not be changed by the buyer!

Note: the IBAN is the required and mandatory account identification of the beneficiary

6.2.1.1.3 epi:PaymentInstructionDetails

Explanation: specifies relevant payment information which has to be forwarded to the beneficiary throughout the whole payment chain for reconciliation

Status: MANDATORY

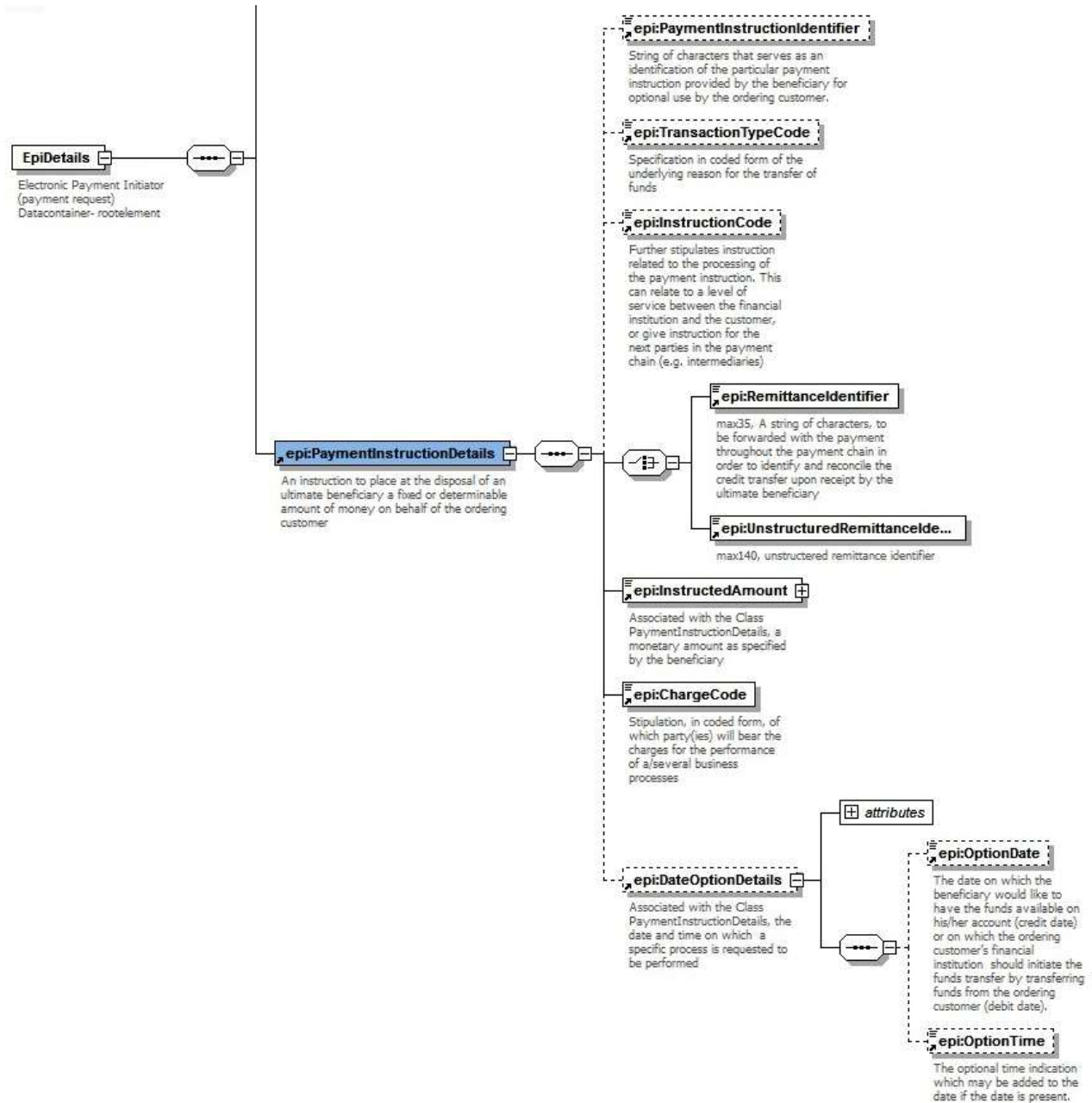


figure 6-5: epi:PaymentInstructionDetails

- **epi:PaymentInstructionIdentifier**

Explanation: string of characters that serves as an identification of the particular payment instruction

Status: OPTIONAL

Data type: an..35

This context (i.e. additional payment information) can be displayed to the buyer in the online-banking application of the debtor bank.

Please note that not every eps debtor bank supports this functionality.

Supported charset: restricted charset (see chapter 4.3)

- **epi:TransactionTypeCode**

Explanation: specification in coded form of the underlying reason for the transfer of funds

Status: OPTIONAL

Data type: an 3

This information can not be changed by the buyer!

Data element is not supported in online-banking applications!

- **epi:InstructionCode**

Explanation: further stipulates instruction related to the processing of the payment instruction. This can relate to a level of service between the financial institution and the customer, or give instruction for the next parties in the payment chain (e.g. intermediaries)

Status: OPTIONAL

Data type: an..35

This information can not be changed by the buyer!

Data element is not supported in online-banking applications!

- **epi:RemittanceIdentifier**

Explanation: structured remittance identifier "Zahlungsreferenz", a unique string of characters (a reference), to be forwarded unaltered/unchanged with the payment throughout the entire payment chain in order to identify and reconcile the credit transfer upon receipt by the ultimate beneficiary

Status: MANDATORY

Data type: an..35

To be presented in the online-banking to the buyer/debtor!

The remittance information can not be changed by the buyer!

Supported charset: restricted charset (see chapter 4.3)

Note

The structured remittance information will be processed in a SEPA Credit Transfer (SCT) as structured remittance information (max35Text).

- **epi:UnstructuredRemittanceIdentifier**

Explanation: unstructured remittance identifier "Verwendungszweck"; a unique string of characters (reference) to be forwarded unaltered throughout the entire payment chain in order to identify and reconcile the credit transfer upon receipt by the ultimate beneficiary

Status: CONDITIONAL (MANDATORY)

Data type: an..140

To be presented in the online-banking to the buyer/debtor!

The unstructured remittance information can not be changed by the buyer!

Supported charset: restricted charset (see chapter 4.3)

Note

The unstructured remittance information will be processed in a SEPA Credit Transfer (SCT) as unstructured remittance information (max140Text).

- **epi:InstructedAmount**

Status: MANDATORY

Data type: decimal

Note: the amount format has to be e.g. 150.55 (no semicolon!)

Attribute: AmountCurrencyIdentifier

Explanation: ISO 4217 currency code

Status: MANDATORY

Data type: a 3

This information can not be changed by the buyer!

- **epi:ChargeCode**

Explanation: stipulation, in coded form, of which party(ies) will bear the charges for the performance of a/several business processes

In domestic online payments the beneficiary must use the code "SHA" (abbreviation for "shared": transaction charges on the ordering customer's financial institution side are to be borne by the ordering customer, transaction charges on the beneficiary's financial institution side are to be borne by the beneficiary customer)

Status: MANDATORY

Data type: a 3

This information can not be changed by the buyer!

Data element is not supported in online banking applications!

- **epi:DateOptionDetails**

Status: OPTIONAL

Attribute: DateSpecificationCode

Explanation: the date and time on which a specific process is requested to be performed

Status: MANDATORY

Data type: a 3

Code: DBD (debit date) - must be used by the beneficiary in eps payment request!

CRD (credit date) - Data element is currently not supported in online-banking applications!

- **epi:OptionDate**

Explanation: the debit date set by the beneficiary/merchant

Status: OPTIONAL

Data type: xsd:date (z. B. 2003-03-14)

NOTE: the debit date option (e.g. to set a future debit date) is not supported by every debtor bank in the online-banking application!

- **epi:OptionTime**

Explanation: the optional time indication which may be added to the date if the date is present

Status: CONDITIONAL

Data type: xsd:time (e.g. 12:00:00)

Data element is not supported in onlinebanking applications!

6.2.1.2 *atrul:AustrianRulesDetails*

Part of the AustrianRules datacontainer, see specification in chapter [6.3](#).

6.2.2 PaymentConfirmationDetails - eps payment confirmation

Content of the eps payment confirmation message

- the eps debtor bank always creates a full payment confirmation including the original payment request
- the eps debtor bank always has to digitally sign the eps confirmation message when forwarding to the SO

content of eps payment confirmation message from SO to merchant

- a merchant is not using a SSL certificate to secure communication: the SO forwards a reduced payment confirmation message to the merchant with the following data elements:
 - epi:RemittanceIdentifier **OR** epi:UnstructuredRemittanceIdentifier (unique reference for reconciliation)
 - eps:PayConApprovingUnitDetails: issuing unit
 - eps:PayConApprovalTime: confirming time when payment information is created
 - eps:PaymentReferenceIdentifier: unique payment reference assigned by the bank
 - eps:StatusCode
- a merchant is using SSL security (noted by the beneficiary in his ConfirmationURL, e.g. <https://www.testshop...>):
the SO forwards a full payment confirmation including the original payment request to the merchant.

Usage Rules:

A merchant requests digitally signed eps payment confirmation message:
set the code "DigSig" in the data element „SIG" within the AustrianRules data

Payment Initiation	Confirmation-Url merchant	Confirmation from debtor bank	Confirmation to merchant
SIG (atrul) not set	https	Signatur valid	Digitally signed eps confirmation message by debtor bank, full eps payment confirmation message
SIG (atrul) not set	http	Signatur valid	Not signed, reduced eps payment confirmation message
SIG (atrul) is set	https	Signatur valid	Digitally signed eps confirmation message by debtor bank, full eps payment confirmation message
SIG (atrul) is set	http	Signatur valid	Digitally signed by SO, reduced eps payment confirmation message
*	https	Signatur invalid	Invalid signed by debtor bank
*	http	Signatur invalid	none, http-Status 412 to debtor bank
*	*	unsigned	none, http-Status 412 to debtor bank
*	*	ApprovingUnitIdentifier unknown	none, http-Status 412 to debtor bank
*	*	XML corrupt	none, http-Status 412 to debtor bank

Digitally signed eps payment confirmation message

In the payment confirmation only the PaymentConfirmationDetails will be signed.

Certificate quality: X.509 v3 certificate, e.g. A-Trust¹ server certificate „a.sign corporate signature" (at minimum the certificate „a.sign corporate light" shall be used).

¹ <http://www.a-trust.at>

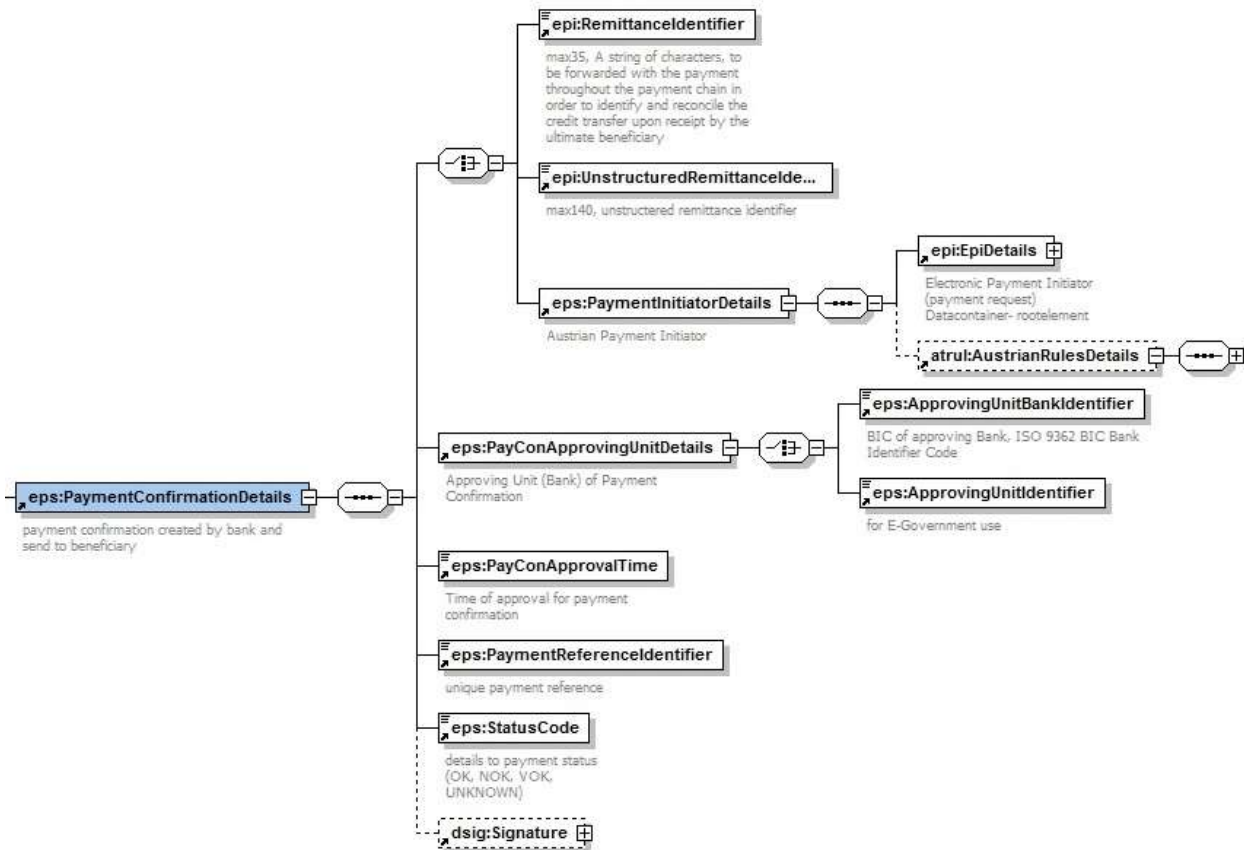


figure 6-6: PaymentConfirmationDetails

6.2.2.1 *epi:RemittanceIdentifier*

Status: CONDITIONAL (Mandatory)

Details see chapter [6.2.1.1.3](#)

6.2.2.2 *epi:UnstructuredRemittanceIdentifier*

Status: CONDITIONAL (Mandatory)

Details see chapter [6.2.1.1.3](#)

6.2.2.3 *eps:PaymentInitiatorDetails*

Status: CONDITIONAL

Details see chapter [6.2.1](#)

6.2.2.4 *eps:PayConApprovingUnitDetails*

Explanation: issuing unit of payment confirmation, e.g. bank or E-Government

Status: MANDATORY

For additional information on issuing the eps payment confirmation, see chapter 7.1.7 and 7.1.11!

- **eps:ApprovingUnitBankIdentifier**

Explanation: BIC (ISO 9362, Business Identifier Code) of issuing bank

Status: CONDITIONAL

Data type: an 11

- **eps:ApprovingUnitIdentifier**

Explanation: identification issuing unit/organisation if not recognized/identified by a BIC

Status: CONDITIONAL

Data type: an..255

6.2.2.5 *eps:PayConApprovalTime*

Explanation: time the payment confirmation is created (most likely not the same date when the beneficiary is credited)

Status: MANDATORY

Data type: xs:dateTime

6.2.2.6 *eps:PaymentReferenceIdentifier*

Explanation: unique payment reference assigned by the debtor bank;
this could be the unique payment reference („Ersterfasserreferenz“) of the Austrian payments system; otherwise an eps bank should create an eps unique payment reference analogue the Austrian reference for “Ersterfasserreferenz”

Status: MANDATORY

Data type: an..28

6.2.2.7 *eps:StatusCode*

Explanation: status of the eps payment process (OK, VOK, NOK, UNKNOWN)

Status: MANDATORY

NOTE: Buyer Bank sends only OK or NOK status to the SO in the eps payment confirmation message!

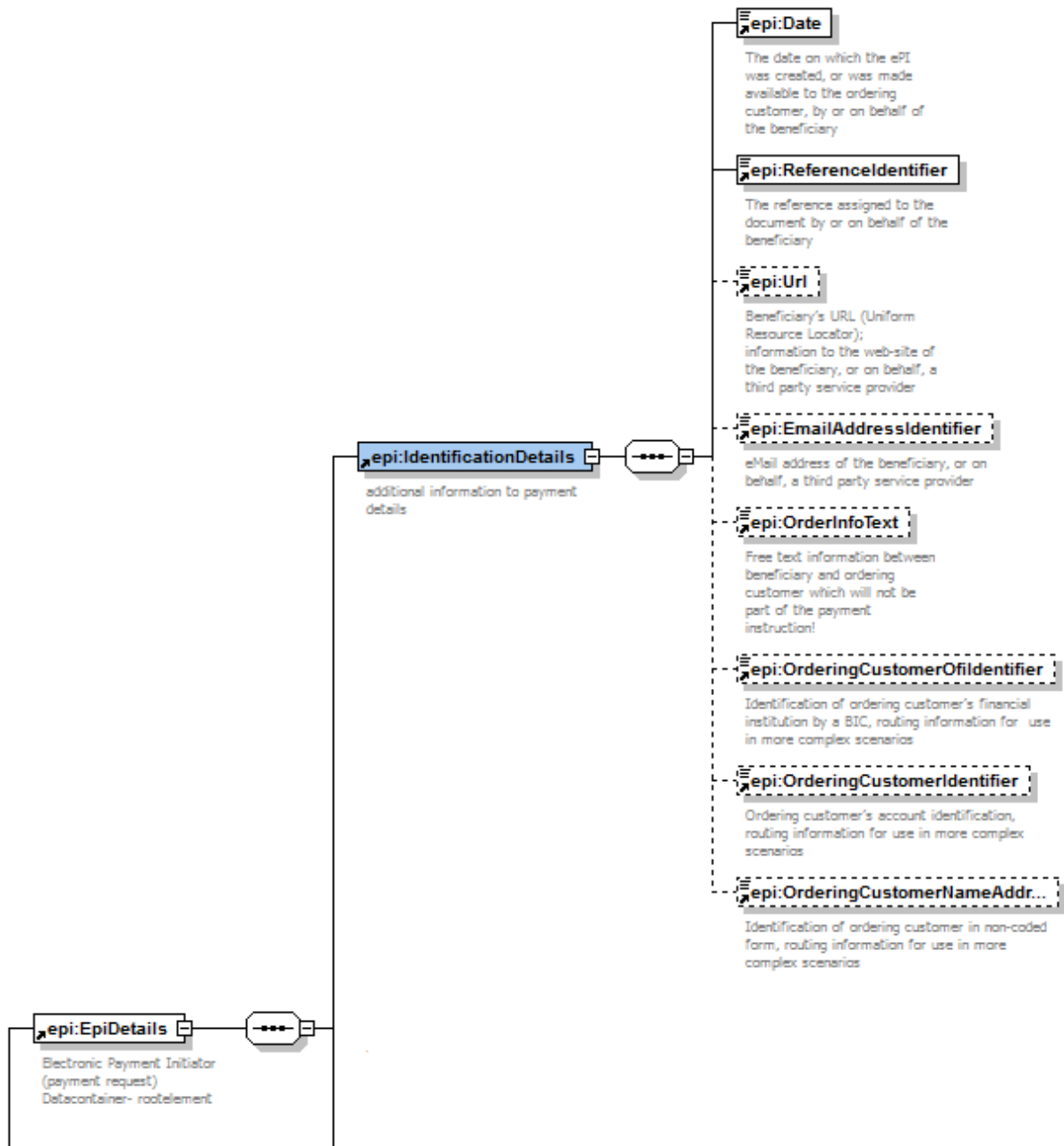


Figure 6-7: epi:EpiDetails

6.2.2.8 epi:OrderingCustomerOfIdentifier

Explanation: BIC of the buyer bank, which is passed on to the merchant for Refund purposes.

Status: OPTIONAL

Daty type: an..11

6.2.2.9 *epi:OrderingCustomerIdentifier*

Explanation: IBAN of the buyer bank, which is passed on to the merchant for Refund purposes.

Status: OPTIONAL

Daty type: an..34

6.2.2.10 *epi:OrderingCustomerNameAdressText*

Explanation: Name and Adresse of the buyer, which is passed on to the merchant for Refund purposes. .

Status: OPTIONAL

Daty type: an..140

Usage status code UNKNOWN:

Within the eps payment initiation message (epsp: Transfer Initiator Details) to the debtor bank the SO sends his own URL for TransactionOkUrl or TransactionNokUrl so that the SO is called in the redirect from the online-banking to the shop.

If the SO has not received an eps confirmation message from the debtor bank in time, the SO will send an eps confirmation message with the status code UNKNOWN to the shop. Finally the buyer will be redirected to the shop.

If the debtor bank forwards an eps payment confirmation message after the redirect has happened, this confirmation message will be rejected as usual with HTTP status 412.

6.2.2.11 *dsig:Signature*

Explanation: external XML schema for digital signature (W3C Standard , XMLD-Sig Schema)

Status: OPTIONAL

6.2.3 ShopConfirmationDetails - merchant confirmation

Merchant has to confirm the receipt of the eps payment confirmation - Content and structure please see chapter [6.4](#).

6.3 AustrianRules Schema

This XML data container specifies additional elements which are not part of the ePI Schema to offer supplementary services. This XML schema is part of the EPSPayment schema.

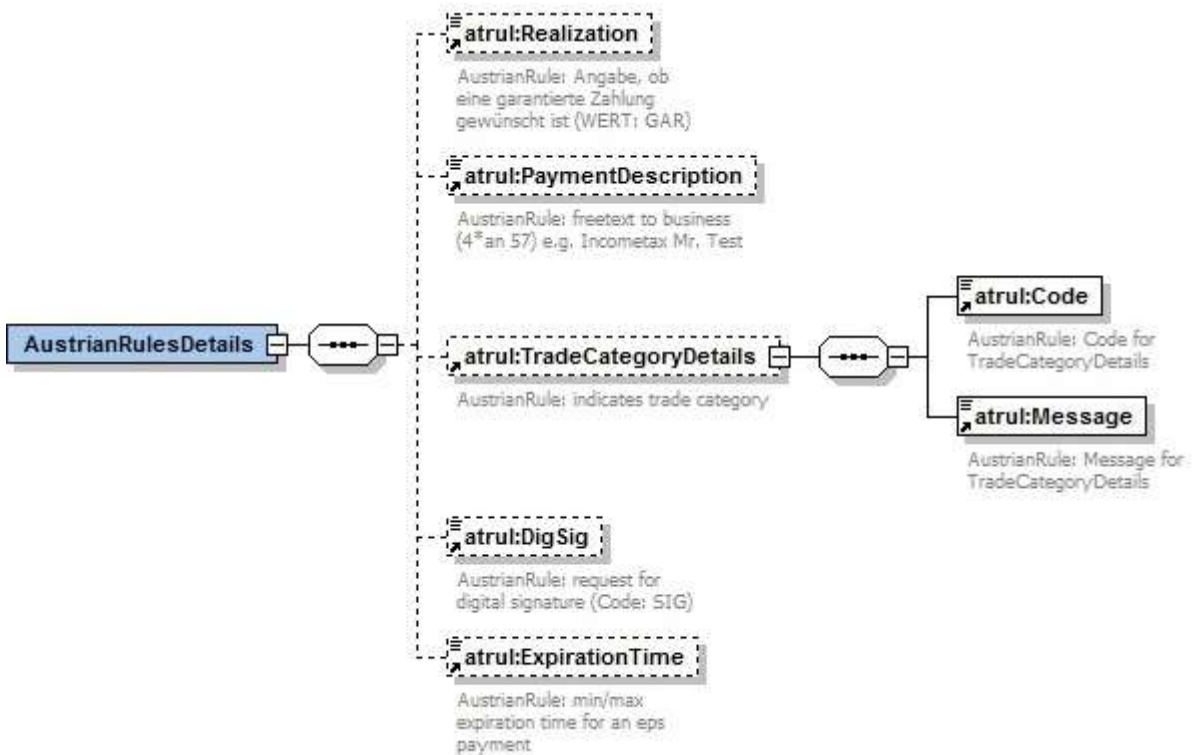


figure 6-8: atrul:AustrianRulesDetails

6.3.1 atrul:Realization

For future use!

At present this data element is not supported in online-banking applications and therefore can not be used by the merchant!

At the moment guaranteed eps payments are specified and handled by the underlying eps merchant contract.

Explanation:	information by the beneficiary if a guaranteed payment is requested
Status:	OPTIONAL
Data type:	an..3
Code:	GAR

6.3.2 atrul:PaymentDescription

At present this data element is not supported in online-banking applications and therefore can not be used by the merchant!

Explanation: further information for the underlying business
Status: OPTIONAL
Data type: 4*an..57

6.3.3 atrul:TradeCategory

For future use!

At present this data element is not supported in online-banking applications and therefore can not be used by the merchant!

Explanation: specification of business
Status: OPTIONAL
Code not specified yet!
Data type: an 3

This information can not be changed by the buyer!

Message

Explanation: information to TradeCategory
Status: OPTIONAL
Data type: an..255

This information can not be changed by the buyer!

6.3.4 atrul:DigSig

Explanation: merchant information whether he requests a digitally signed payment confirmation
Status: OPTIONAL
Data type: an..3
Code: SIG

- merchant requests digitally signed eps payment confirmation message: merchant has to set the code "DigSig" in the data element „SIG" within the AustrianRules data, the payment confirmation then will be signed, see chapter 6.2.2
- no request by merchant: it is up to the bank's system whether the payment confirmation is signed or not, see chapter 6.2.2

6.3.5 atrul:ExpirationTime

Explanation: min-/max execution time for an eps transaction/payment

Status: OPTIONAL

Data type: xsd:datetime

Explanation:

- merchant can set a time stamp in the eps initiation message (min. 5, max. 60 minutes)
- no time stamp set by merchant: SO adds default 60 minutes
- always as absolut time specification (including the timezone)
- debtor bank has to check the given/submitted time stamp in the online-banking when the payment is authorized by the debtor (not with the eps initiation!)
- after expiration of the execution time the debtor bank is not allowed to accept an eps initiation to carry out the payment - thus the status code "OK" may not be sent
- after expiration of the execution time the debtor bank has to inform the buyer in the online-banking application accordingly; an eps confirmation message with status code "NOK" is created and forwarded to the SO

NOTE: Buyer Bank sends only OK or NOK status to the SO in the eps payment confirmation message!

Usage status code UNKNOWN:

Within the eps payment initiation message (epsp: Transfer Initiator Details) to the debtor bank the SO sends his own URL for TransactionOkUrl or TransactionNokUrl so that the SO is called in the redirect from the online-banking to the shop.

If the SO has not received an eps confirmation message from the debtor bank in time, the SO will send an eps confirmation message with the status code UNKNOWN to the shop. Finally the buyer will be redirected to the shop.

If the debtor bank forwards an eps payment confirmation message after the redirect has happened, this confirmation message will be rejected as usual with HTTP status 412.

6.4 Message: eps Payment Initiation

eps XML message based on XML Schema EPSProtocol-V25.xsd:

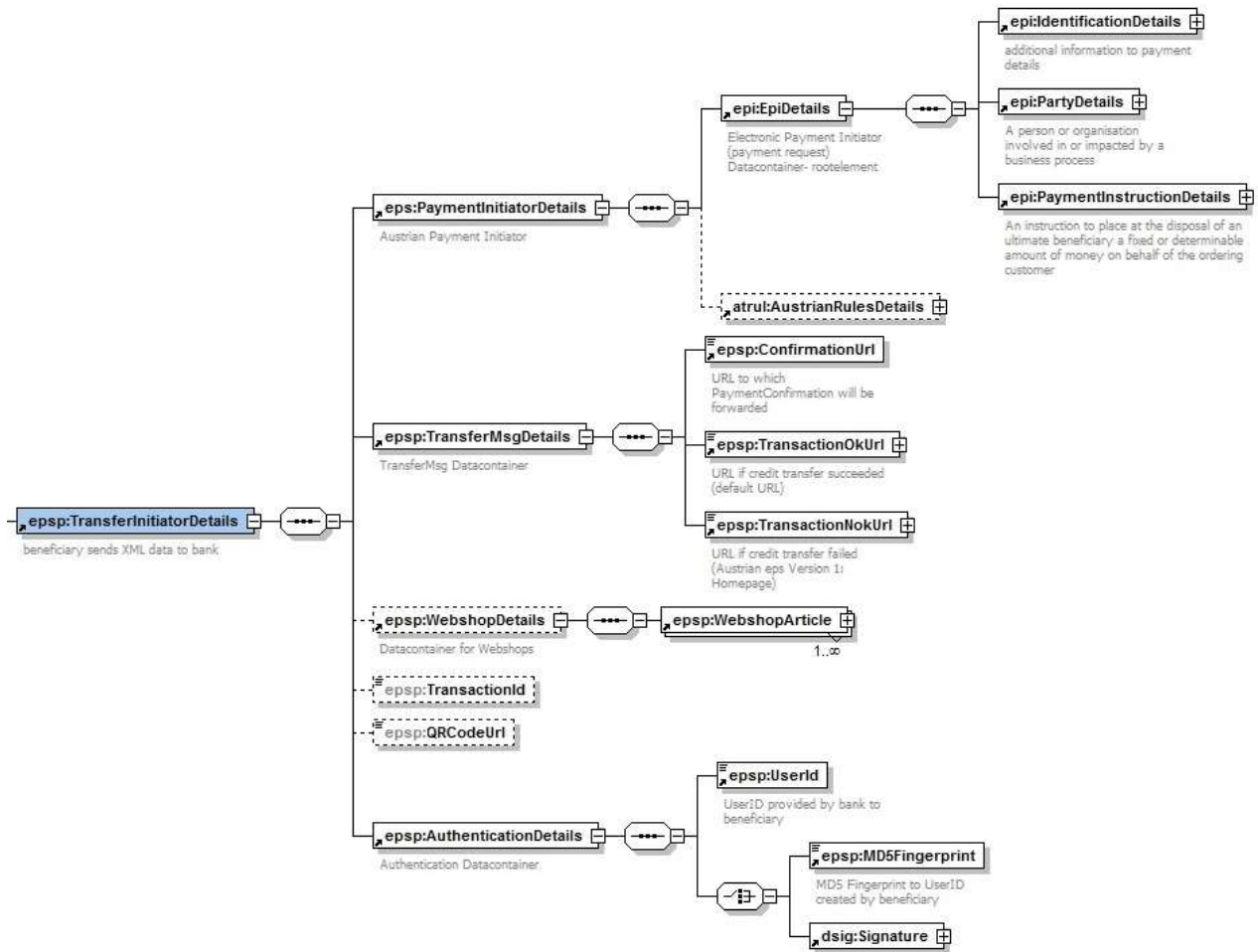


figure 6-9: eps:TransferInitiatorDetails

XML message example without signature (merchant -> Scheme Operator)

```
<?xml version="1.0" encoding="UTF-8"?>
<eps:EpsProtocolDetails SessionLanguage="DE"
xsi:schemaLocation="http://www.stuzza.at/namespaces/eps/protocol/2014/10 EPSProtocol-V26.xsd"
xmlns:atrul="http://www.stuzza.at/namespaces/eps/austrianrules/2014/10"
xmlns:epi="http://www.stuzza.at/namespaces/eps/epi/2013/02"
xmlns:eps="http://www.stuzza.at/namespaces/eps/payment/2014/10"
xmlns:eps="http://www.stuzza.at/namespaces/eps/protocol/2014/10"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <eps:TransferInitiatorDetails>
    <eps:PaymentInitiatorDetails>
      <epi:EpiDetails>
        <epi:IdentificationDetails>
          <epi:Date>2013-02-28</epi:Date>
          <epi:ReferenceIdentifier>1234567890ABCDEFG</epi:ReferenceIdentifier>
        </epi:IdentificationDetails>
        <epi:PartyDetails>
          <epi:BfiPartyDetails>
            <epi:BfiBicIdentifier>GAWIATW1XXX</epi:BfiBicIdentifier>
          </epi:BfiPartyDetails>
          <epi:BeneficiaryPartyDetails>

```

```

        <epi:BeneficiaryNameAddressText>Max Mustermann</epi:BeneficiaryNameAddressText>
<epi:BeneficiaryAccountIdentifier>AT611904300234573201</epi:BeneficiaryAccountIdentifier>
    </epi:BeneficiaryPartyDetails>
</epi:PartyDetails>
    <epi:PaymentInstructionDetails>
        <epi:RemittanceIdentifier>AT1234567890XYZ</epi:RemittanceIdentifier>
        <epi:InstructedAmount AmountCurrencyIdentifier="EUR">150.00</epi:InstructedAmount>
        <epi:ChargeCode>SHA</epi:ChargeCode>
        <epi:DateOptionDetails DateSpecificationCode="CRD">
            <epi:OptionDate>2013-02-28</epi:OptionDate>
            <epi:OptionTime>11:00:00-05:00</epi:OptionTime>
        </epi:DateOptionDetails>
    </epi:PaymentInstructionDetails>
</epi:EpiDetails>
    <atrul:AustrianRulesDetails>
        <atrul:DigSig>SIG</atrul:DigSig>
        <atrul:ExpirationTime>2013-02-28T09:30:47Z</atrul:ExpirationTime>
    </atrul:AustrianRulesDetails>
</eps:PaymentInitiatorDetails>
<epsp:TransferMsgDetails>
<epsp:ConfirmationUrl>http://10.18.70.8:7001/vendorconfirmation</epsp:ConfirmationUrl>
    <epsp:TransactionOkUrl
TargetWindow="Mustershop">http://10.18.70.8:7001/transactionok?danke.asp</epsp:TransactionOkUrl>
    <epsp:TransactionNokUrl
TargetWindow="Mustershop">http://10.18.70.8:7001/transactionnok?fehler.asp</epsp:TransactionNokUrl>
</epsp:TransferMsgDetails>
<epsp:WebshopDetails>
    <epsp:WebshopArticle ArticleName="Toaster" ArticleCount="1" ArticlePrice="150.00"/>
</epsp:WebshopDetails>
<epsp:AuthenticationDetails>
    <epsp:UserId>AKLJS231534</epsp:UserId>
    <epsp:MD5Fingerprint>b5b5559cf873094a689c743e56bd0b4d</epsp:MD5Fingerprint>
</epsp:AuthenticationDetails>
</epsp:TransferInitiatorDetails>
</epsp:EpsProtocolDetails>

```

XML message example with signature (SchemeOperator -> Bank)

```

<?xml version="1.0" encoding="UTF-8"?>
<epsp:EpsProtocolDetails SessionLanguage="DE"
xsi:schemaLocation="http://www.stuzza.at/namespaces/eps/protocol/2014/10 EPSPProtocol-V26.xsd"
xmlns:atrul="http://www.stuzza.at/namespaces/eps/austrianrules/2014/10"
xmlns:epi="http://www.stuzza.at/namespaces/eps/epi/2013/02"
xmlns:eps="http://www.stuzza.at/namespaces/eps/payment/2014/10"
xmlns:epsp="http://www.stuzza.at/namespaces/eps/protocol/2014/10"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <epsp:TransferInitiatorDetails>
        <eps:PaymentInitiatorDetails>
            <epi:EpiDetails>
                <epi:IdentificationDetails>
                    <epi:Date>2014-10-28</epi:Date>
                    <epi:ReferenceIdentifier>1234567890ABCDEFG</epi:ReferenceIdentifier>
                </epi:IdentificationDetails>
                <epi:PartyDetails>
                    <epi:BfiPartyDetails>
                        <epi:BfiBicIdentifier>GAWIATW1XXX</epi:BfiBicIdentifier>
                    </epi:BfiPartyDetails>
                    <epi:BeneficiaryPartyDetails>
                        <epi:BeneficiaryNameAddressText>Max Mustermann</epi:BeneficiaryNameAddressText>
                    </epi:BeneficiaryPartyDetails>
                </epi:PartyDetails>
            </epi:PaymentInitiatorDetails>
            <epi:RemittanceIdentifier>AT1234567890XYZ</epi:RemittanceIdentifier>
            <epi:InstructedAmount AmountCurrencyIdentifier="EUR">150.00</epi:InstructedAmount>
            <epi:ChargeCode>SHA</epi:ChargeCode>
            <epi:DateOptionDetails DateSpecificationCode="CRD">

```

```

        <epi:OptionDate>2014-10-28</epi:OptionDate>
        <epi:OptionTime>11:00:00-05:00</epi:OptionTime>
    </epi:DateOptionDetails>
</epi:PaymentInstructionDetails>
</epi:EpiDetails>
<atrul:AustrianRulesDetails>
    <atrul:DigSig>SIG</atrul:DigSig>
    <atrul:ExpirationTime>2014-10-28T09:30:47Z</atrul:ExpirationTime>
    <atrul:StatusMsgEnabled>>true</atrul:StatusMsgEnabled>
</atrul:AustrianRulesDetails>
</eps:PaymentInitiatorDetails>
<eps:TransferMsgDetails>
<eps:ConfirmationUrl>http://10.18.70.8:7001/vendorconfirmation</eps:ConfirmationUrl>
    <eps:TransactionOkUrl
TargetWindow="Mustershop">http://10.18.70.8:7001/transactionok?danke.asp</eps:TransactionOkUrl>
    <eps:TransactionNokUrl
TargetWindow="Mustershop">http://10.18.70.8:7001/transactionnok?fehler.asp</eps:TransactionNokUrl>
</eps:TransferMsgDetails>
<eps:WebshopDetails>
    <eps:WebshopArticle ArticleCount="1" ArticleName="Toaster" ArticlePrice="150.00"/>
</eps:WebshopDetails>
<eps:TransactionId>epsHXOSINN8T</eps:TransactionId>
<eps:QRCodeUrl>epspayment://eps.or.at/?transactionid=123HOXA123 </eps:QRCodeUrl>
<eps:AuthenticationDetails>
    <eps:UserId>epsSchemeOperator</eps:UserId>
    <dsig:Signature Id="hotVault" xmlns:dsig="http://www.w3.org/2000/09/xmldsig#">
        <dsig:SignedInfo>
            <dsig:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
            <dsig:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1" />
            <dsig:Reference Id="reference-data-0" URI="">
                <dsig:Transforms>
                    <dsig:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature" />
                </dsig:Transforms>
                <dsig:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
                <dsig:DigestValue>hr37g9hhaOxCrutyMAX8XIK1atk=</dsig:DigestValue>
            </dsig:Reference>
        </dsig:SignedInfo>
        <dsig:SignatureValue>OOMR5YBr4ZiQpAhxfbU02Qi7t1F4prUlb+Y2DafoKwEZXuSpALFiQbOzCMXRHisa
77hbg2Hc6R68n1x/PrlOd+nGMCRIGlzI7UyWqBZLhRnYqy+hKB/fmaoPI5NafTDP
jnCdN3mL3Rbc7+2gqFjhk1XuJJeIYN5dwwrtMefAuQ=</dsig:SignatureValue>
    <dsig:KeyInfo>
        <dsig:X509Data>
            <dsig:X509Certificate>MIIEpjCCA46gAwIBAgIADAMrDMA0GCSqGSIb3DQEBBQUAMIGfMQswCQYDVQQGEwJB
VDFIMEYGA1UEChM/QS1UcnVzdCBHZXMuIGYyUFlNpY2hlcmlhXARzC3lzdGVtZSBp
bSBibGVrdHlulERhdGVudmVya2VociBHbWJIMS1wIAVDVQQLExhLXNpZ24tY29y
cG9yYXRILWxpZ2h0LTAxMS1wIAVDVQQLExhLXNpZ24tY29ycG9yYXRILWxpZ2h0
LTAxMB4XDTA0MTAwNjA5MDY1MVVoXDTA3MTAwNjA5MDY1MVoweTELMAkGA1UEBHMW
YXQxJDAiBgNVBAoTG1NwYXRyXNzZW4gRGF0ZW5kaWVuc3QgR21iSDEQMA4GA1UE
CxMHU3BhcmRhdEBlbGkGA1UEAxMSU3BhcmRhdC1lcHMtU2lnLTAxMRUwEwYDVQQF
Eww2MTk0OTM5ODUxNzcwZ8WdQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAKkbmACF
B8spL+bLkg+E5h14i/D4vALmrrBwSc98fVLB87rZ/TVK+cuXlh2ug3P5cOESQwCo
buVfLGN0V+WrrH/JLgZn7KRc14j/VM0vITURkiEkDnQgjUF5tDUCkG/ft5uJqjH9
IK8+DtKJgsSd0OYyB9Ewzi/8t33M9oOAI7tjAgMBAAGjggGSMIIBjAJBgNVHRME
AjAAMBEGA1UdDgQKBAHP62yLyG0dIDBYBgNVHSAEUTBPME0GBYoABEBBwEwQjBA
BggrBgEFBQcCARY0aHR0cDovL3d3dy5hLXRYdXN0LmF0L2RvY3MvY3AvYS1zaWdu
LWNvcnBvcmlF0ZS1saWdodDATBgNVHSMEDDAKGAhOnn/UL8kfHzB/BggrBgEFBQcB
AQRzMHEwRgYIKwYBBQUHMAKGOMh0dHA6Ly93d3cuYS10cnVzdC5hdC9jZjZJ0cy9h
LXNpZ24tY29ycG9yYXRILWxpZ2h0LTAxYS5jcjQwJwYIKwYBBQUHMAGGG2h0dHA6
Ly9vY3NwLmEtDjH1c3QuYXQvbnZncDAOBgNVHQ8BAf8EBAMCBLAwbGyDVR0fBGcw
ZTBjoGGgX4ZdbGRhcDovL2xkYXAuYS10cnVzdC5hdC9vdT1hLXNpZ24tY29ycG9y
YXRILWxpZ2h0LTAxLGA89QS1UcnVzdCxpUFUP2NlcnRpZmljYXRlcmV2b2NhdGlv
bmxc3Q/MA0GCSqGSIb3DQEBBQUAA4IBAQBsr+16GlfUpl0YYae39pVLdz4holom
nbx3k9/KwjkReE2dJJeRDk/46BWUf19V/xPHOJ4GjaAU0WJpO7ITCsQeiVdUJbB5
pgHYFyHjgOyKz9DwCtcpWdS3luspSjwrYhDd/Hk6+FxstDaKPN/O3Dj/7FcBChR
hldvrCXymk2ah4ezi+B2hQ+n2pWttXkPvDXCUqEjOqAnTc1FBk34CBISUphul0W5
G/NUtlc/HzjkfSFDZvSfRmCZmQRq4IIWYhSua7RuP93iAn8zrJ71PGzIAHowkk
Hhchb9ZpjI93sIX1Qa0hH+4AQ6ImvHaBwioG0so8Gd/Vu2PQI1LBU9No</dsig:X509Certificate>
        </dsig:X509Data>
    </dsig:KeyInfo>

```



```

</dsig:Signature>
</eps:AuthenticationDetails>
</eps:TransferInitiatorDetails>
</eps:EpsProtocolDetails>

```

eps:PaymentInitiatorDetails

Explanation: for detailed information to eps:PaymentInitiatorDetails please see chapter [6.2.1](#).

Status: MANDATORY

eps:TransferMsgDetails

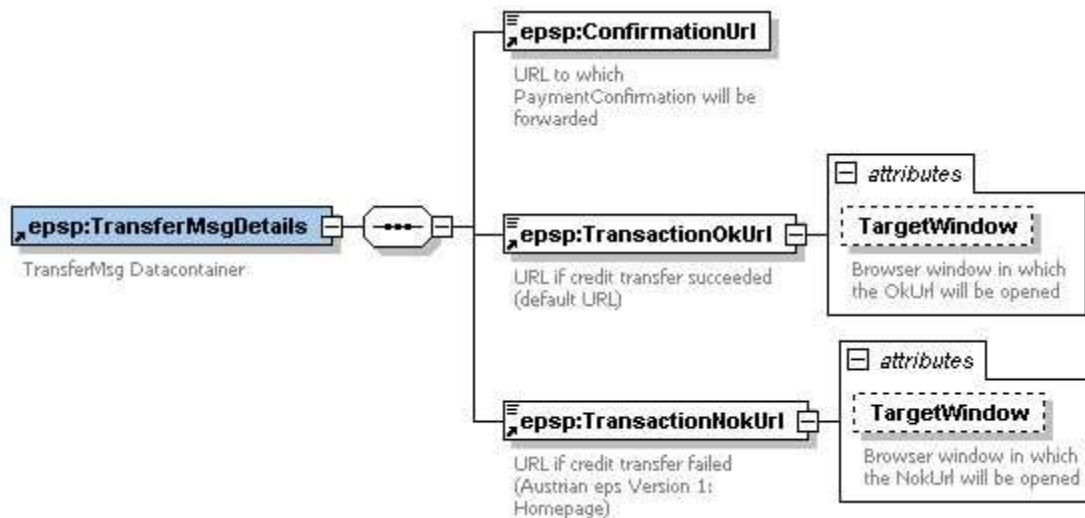


figure 6-10: eps:TransferMsgDetails

Explanation: relevant URLs provided by the merchant

Status: MANDATORY

Data element	Explanation	Status	Data Type
eps:ConfirmationUrl	<p>URL length limited to max 512!</p> <p>URL provided by the merchant to which the eps vitality check and payment confirmation message is sent by the SO.</p> <p>After receipt of the eps payment confirmation message the merchant must confirm immediately</p> <ul style="list-style-type: none"> to the SO the receipt of the payment confirmation as described in chapter 7.1.13 we recommend the merchant to set the buyer's status within the shop application to "paid" and not to wait for the buyer's return 	M	xs:anyURI

Data element	Explanation	Status	Data Type
	to the merchant's shop in order to avoid the loss of the buyer due to unexpected failures (e.g. buyer closes browser window).		
epsp:TransactionOkUrl	URL length limited to max 512! URL provided by the merchant to redirect the buyer to the merchant's shop. Please note: the payment transaction should NOT depend on the URL call, it should be completed with the receipt of the eps payment confirmation message!	M	xs:anyURI
epsp:TransactionNokUrl	URL length limited to max 512! If a transaction could not be executed, the buyer (after notification by the system) will be redirected to this URL (e.g. merchant's homepage).	M	xs:anyURI

epsp:TransactionOkUrl, epsp:TransactionNokUrl

Attribut: TargetWindow

Explanation: target browser window to which an OkUrl or NoKUrl will be redirected.

Status: OPTIONAL

epsp:WebshopDetails

Explanation: the merchant can provide additional information of the underlying business to the buyer; this information can optionally be displayed in the payment process to the merchant (depending on the banks online banking application)

Status: OPTIONAL

Data element	Explanation	Status	Data Type
epsp:WebshopArticle	Detailed information which can be displayed to the buyer. Important notice <u>Mandatory</u> if epsp:WebShopDetails is present in the eps XML message!	C (1..∞)	
Attribute			
ArticleName	Article description without Euro-character (€)	M	an..255
ArticleCount	Number of items	M	an..5

Data element	Explanation	Status	Data Type
ArticlePrice	Price per article/unit	M	n..15

epsp:TransactionID

Explanation: For the clear identification of an eps transaction in the further process.

The TransactionID is generated by the E-Service Scheme Operator and forwarded to the bank as well as returned to the merchant in BankResponse. Using the TransactionID, transaction data can be queried via an XML interface.

Status: OPTIONAL

Data type: xsd:pattern value="[a-zA-Z0-9\-\._~]{1,36}", z.B. epsHXOSINN8T

epsp:QRCodeUrl

Explanation: Information on creating a QR code on the merchant webshop or on the bank login page.

The QRCodeUrl is generated by the E-Service Scheme Operator and forwarded to the bank as a URL (example: epspayment://eps.or.at/?transactionid=123HOXA123) and also returned to the merchant (BankResponse).

Based on the URL incl. Transaction-ID, different scenarios such as "Browser2APP via QR-Code", "POS2APP via QR Code" and "APP2APP " can be realised.

The encoding of the URL as a QR code or presentation as a hyperlink to open another mobile app is up to the merchant or the bank and depends on the particular application.

Status: OPTIONAL

A Data type:xsd:anyURI, **URL length limited to max 512!**

epsp:AuthenticationDetails

Explanation: a payment request (eps payment initiation message) must have a valid merchant identification, either a UserID + MD5Fingerprint OR a UserID + certificate (dsig:Signature)!

Status: MANDATORY

NOTE: The SO always acts as a merchant towards the buyers bank, the authorization takes place exclusively via electronic signature (per bank group can be configured whether in the UserID the original dealer ID or 'epsSchemeOperator' is sent)!

Merchant identification

2 alternatives:

- identification with UserID (merchantID) and MD5Fingerprint
- identification with UserID (merchantID) and electronic signature and certificate

Requested attributes of the MD5Fingerprint:

- Secret (Merchant PIN)
- epi:IdentificationDetails:
 - epi:Date
 - epi:ReferenceIdentifier
- epi:PartyDetails
 - epi:BeneficiaryPartyDetails
 - epi:BeneficiaryAccountIdentifier
- epi:PaymentInstructionDetails:
 - epi:RemittanceIdentifier
 - epi:InstructedAmount
 - AmountCurrencyIdentifier
- AuthenticationDetails:
 - epsp:UserId

The content of above requested attributes (without separators!) will be strung together as basis for the MD5Fingerprint calculation.

Please note:

The data for the MD5Fingerprint calculation has to be UTF-8 encoded, both upper and lower case is possible.

An electronic signature offers the following security features:

- **data integrity**: the assurance that the content has not been altered or destroyed in an unauthorised manner.
- **authentication**
- **confidentiality**

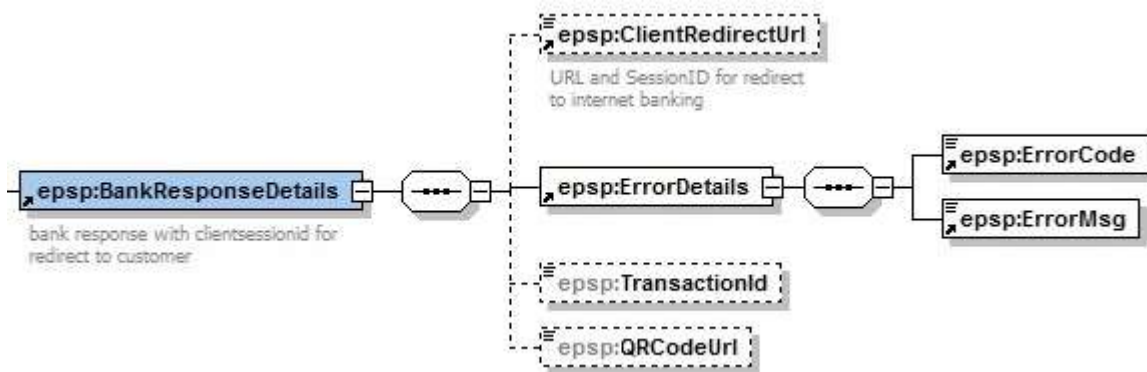
Explanations to the signature profile for E-Government ("Signaturprofil für E-Government") can be downloaded at <http://eservice.stuzza.at/en>.

Note: this document is only available in German!

Data element	Explanation	Status	Data Type
epsp:UserID	Identification of a merchant by his UserID, which is provided by the bank	C	an..25
epsp:MD5Fingerprint	MD5Fingerprint calculated by the merchant and forwarded as hash to the bank	C	an..255
dsig:Signature	W3C Standard for electronic signature (XMLDSig Schema), external XML Schema for merchant identification by using a certificate	C	

6.5 Message: bank reponse

XML message based on XML Schema EPSProtocol-V25.xsd:



3

figure 6-71: epsp:BankResponseDetails

XML message example

```
<?xml version="1.0" encoding="UTF-8"?>
<eps:EpsProtocolDetails SessionLanguage="DE"
xsi:schemaLocation="http://www.stuzza.at/namespaces/eps/protocol/2014/10 EPSProtocol-V26.xsd"
xmlns:atrul="http://www.stuzza.at/namespaces/eps/austrianrules/2014/10"
xmlns:epi="http://www.stuzza.at/namespaces/eps/epi/2013/02"
xmlns:eps="http://www.stuzza.at/namespaces/eps/payment/2014/10"
xmlns:eps="http://www.stuzza.at/namespaces/eps/protocol/2014/10"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <eps:BankResponseDetails>
    <eps:ClientRedirectUrl>http://epsbank.at/asdk3935jdlf043</eps:ClientRedirectUrl>
    <eps:ErrorDetails>
      <eps:ErrorCode>000</eps:ErrorCode>
      <eps:ErrorMsg>Keine Fehler</eps:ErrorMsg>
    </eps:ErrorDetails>
    <eps:TransactionId>epsHXOSINN8T</eps:TransactionId>
    <eps:QRCodeUrl>epspayment://eps.or.at/?transactionid=123HOXA123 </eps:QRCodeUrl>
  </eps:BankResponseDetails>
</eps:EpsProtocolDetails>
```

eps:BankResponseDetails

Status: MANDATORY

Data element	Explanation	Status	Data Type
eps:ClientRedirectUrl	Complete Redirect-URL provided by the debtor bank to redirect the buyer to his online-banking application if no error has occurred (ErrorCode „000“)	O	xs:anyURI
eps:ErrorDetails		M	
eps:ErrorCode	Error status in coded form	M	an 3

Data element	Explanation	Status	Data Type
epsp:ErrorMsg	error respectively status information in text form Note: supplementary bank service – optional error information together with the ErrorCode may be provided by the bank.	M	an..255
epsp:TransactionId	clear identification of an eps transaction in the further process	O	an 12
epsp:QRCodeUrl	URL length limited to max 512! Information on creating a QR code or hyperlinks at the merchant webshop or on the bank login page	O	xs:anyURI

Available error codes:

ErrorCode	ErrorMsg
000	No error
003	Currency incorrect/wrong
004	Identification incorrect/wrong
007	Error in XML-Stream
008	Unknown error
009	Internal error
010	IBAN invalid
011	BIC invalid
012	Expiration Time or execution time invalid
013	No cross-border transactions allowed (only to be used by SO!, not by debtor bank!)
014	Connection timeout to bank or service at online-banking
015	Bank selection invalid

In the case of an error the transaction is ended.

6.6 Message: Vitaliy-Check

XML message based on XML Schema EPSProtocol-V25.xsd:

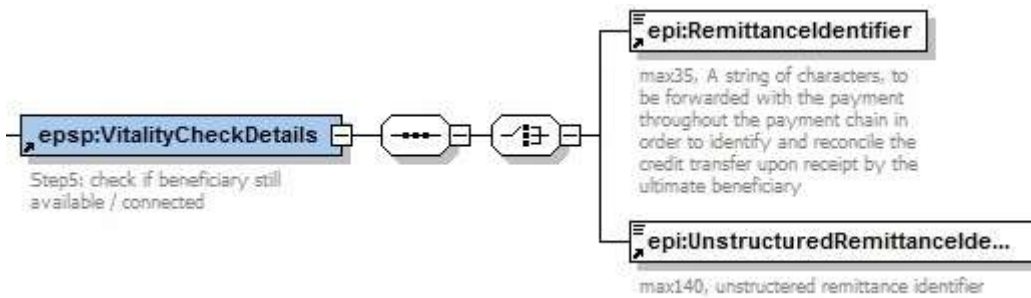


figure 6-12: epsp:VitaliyCheckDetails

XML message example

```
<?xml version="1.0" encoding="UTF-8"?>
<eps:EpsProtocolDetails SessionLanguage="DE"
xsi:schemaLocation="http://www.stuzza.at/namespaces/eps/protocol/2014/10 EPSProtocol-V26.xsd"
xmlns:atrul="http://www.stuzza.at/namespaces/eps/austrianrules/2014/10"
xmlns:epi="http://www.stuzza.at/namespaces/eps/epi/2013/02"
xmlns:eps="http://www.stuzza.at/namespaces/eps/payment/2014/10"
xmlns:eps="http://www.stuzza.at/namespaces/eps/protocol/2014/10"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <eps:VitaliyCheckDetails>
    <epi:RemittanceIdentifier>AT1234567890XYZ</epi:RemittanceIdentifier>
  </eps:VitaliyCheckDetails>
</eps:EpsProtocolDetails>
```

epsp:VitaliyCheckDetails

Status: MANDATORY

PLEASE NOTE!

Either

< epi:RemittanceIdentifier> or
< epi:UnstructuredRemittanceIdentifier>

Field has to be forwarded in this message!

Data element	Explanation	Status	Data Type
epi:RemittanceIdentifier	Unique string of characters (the payment reference/remittance identifier), created by the beneficiary (merchant), taken from the original payment request.	C (M)	an..35
epi:UnstructuredRemittanceIdentifier	Unique string of characters (the payment reference/remittance identifier), created by the beneficiary (merchant), taken from the original payment request.	C (M)	an..140

6.7 Message: eps payment confirmation

The debtor bank has to create a digitally signed eps confirmation message, including the original eps initiation request/details, and to forward it to the SO (see also chapter 6.2.2).

Explanations to the signature profile for E-Government ("Signaturprofil für E-Government") can be downloaded at <http://www.eps.or.at>.

Note: this document is only available in German!

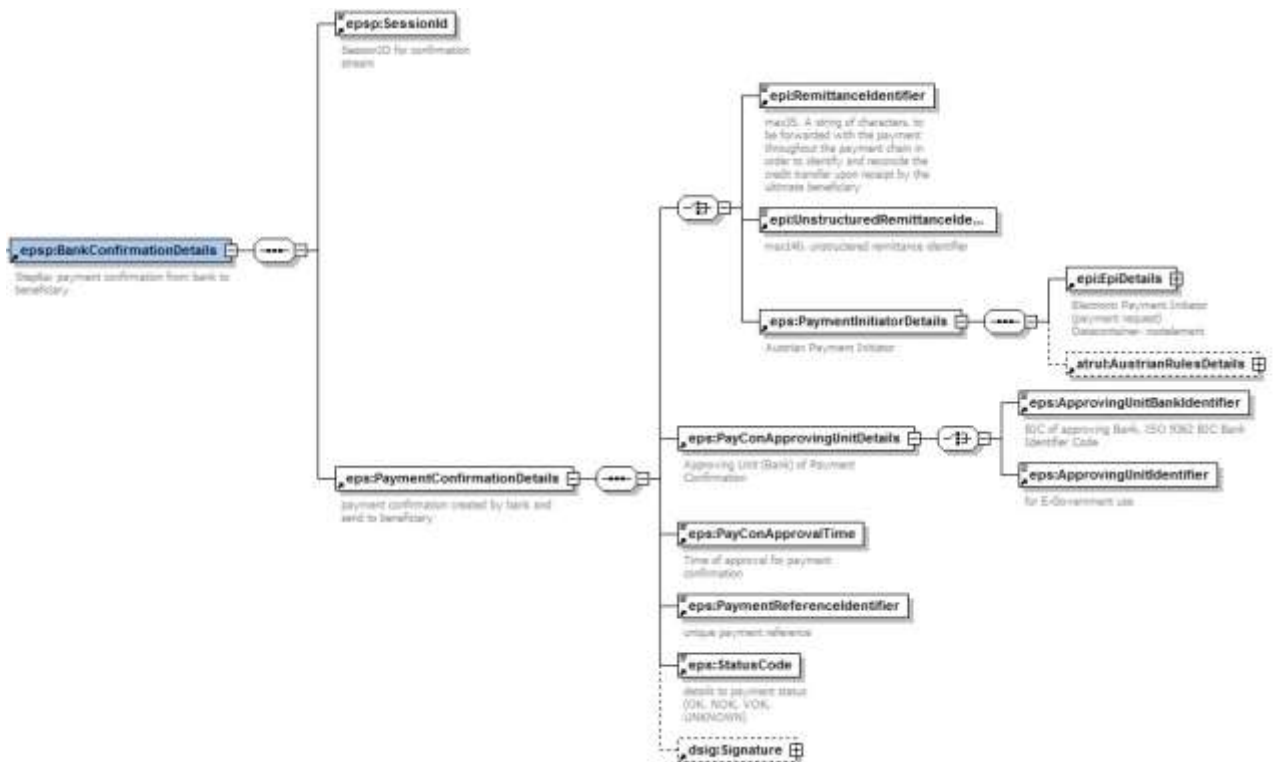


figure 6-13: epsp:BankConfirmationDetails

XML message example (with example electronic signature and epi: RemittanceIdentifier)

```
<?xml version="1.0" encoding="UTF-8"?>
<eps: EpsProtocolDetails SessionLanguage="DE"
  xsi:schemaLocation="http://www.stuzza.at/namespaces/eps/protocol/2014/10 EPSProtocol-V26.xsd"
  xmlns:atrul="http://www.stuzza.at/namespaces/eps/austrianrules/2014/10"
  xmlns:epi="http://www.stuzza.at/namespaces/eps/epi/2013/02"
  xmlns:eps="http://www.stuzza.at/namespaces/eps/payment/2014/10"
  xmlns:epsproto="http://www.stuzza.at/namespaces/eps/protocol/2014/10"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <eps:BankConfirmationDetails>
    <eps:SessionId>String</eps:SessionId>
    <eps:PaymentConfirmationDetails>
      <eps:PaymentInitiatorDetails>
        <epi:EpiDetails>
          <epi:IdentificationDetails>
            <epi:Date>2017-09-28Z</epi:Date>

```

```

<epi:ReferenceIdentifier>9417d3d41f1b4666bfceb3ee21569e08</epi:ReferenceIdentifier>
<epi:OrderInfoText>53600000012960769</epi:OrderInfoText>
<epi:OrderingCustomerOfIdentifier>HAABAT2K</epi:OrderingCustomerOfIdentifier>
<epi:OrderingCustomerIdentifier>AT7122000000123456789</epi:OrderingCustomerIdentifier>
<epi:OrderingCustomerNameAddressText>Max Kauf</epi:OrderingCustomerNameAddressText>
</epi:IdentificationDetails>
<epi:PartyDetails>
  <epi:BfiPartyDetails>
    <epi:BfiBicIdentifier>GAWIATW1XXX</epi:BfiBicIdentifier>
  </epi:BfiPartyDetails>
  <epi:BeneficiaryPartyDetails>
    <epi:BeneficiaryNameAddressText>Max Mustermann</epi:BeneficiaryNameAddressText>
  </epi:BeneficiaryPartyDetails>
  <epi:BeneficiaryAccountIdentifier>AT611904300234573201</epi:BeneficiaryAccountIdentifier>
  </epi:BeneficiaryPartyDetails>
  </epi:PartyDetails>
  <epi:PaymentInstructionDetails>
    <epi:RemittanceIdentifier>AT1234567890XYZ</epi:RemittanceIdentifier>
    <epi:InstructedAmount AmountCurrencyIdentifier="EUR">150.00</epi:InstructedAmount>
    <epi:ChargeCode>SHA</epi:ChargeCode>
    <epi:DateOptionDetails DateSpecificationCode="CRD">
      <epi:OptionDate>2013-02-28</epi:OptionDate>
      <epi:OptionTime>11:00:00-05:00</epi:OptionTime>
    </epi:DateOptionDetails>
  </epi:PaymentInstructionDetails>
</epi:EpiDetails>
<atrul:AustrianRulesDetails>
  <atrul:DigSig>SIG</atrul:DigSig>
  <atrul:ExpirationTime>2017-09-28T09:30:47Z</atrul:ExpirationTime>
</atrul:AustrianRulesDetails>
</eps:PaymentInitiatorDetails>
<eps:PayConApprovingUnitDetails>
<eps:ApprovingUnitBankIdentifier>AAAAAAAAAAAA</eps:ApprovingUnitBankIdentifier>
</eps:PayConApprovingUnitDetails>
<eps:PayConApprovalTime>2017-09-28T14:30:47-05:00</eps:PayConApprovalTime>
<eps:PaymentReferenceIdentifier>AT1234567890XYZ</eps:PaymentReferenceIdentifier>
<eps:StatusCode>OK</eps:StatusCode>
<dsig:Signature Id="hotVault" xmlns:dsig="http://www.w3.org/2000/09/xmldsig#">
  <dsig:SignedInfo>
    <dsig:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
    <dsig:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1" />
    <dsig:Reference Id="reference-data-0" URI="" />
    <dsig:Transforms>
      <dsig:Transform Algorithm="http://www.w3.org/2002/06/xmldsig-filter2">
        <xf2:XPath Filter="intersect"
xmlns:eps="http://www.stuzza.at/namespaces/eps/payment/20031001" xmlns:xf2="http://www.w3.org/2002/06/xmldsig-
filter2">here()/ancestor::eps:PaymentConfirmationDetails[1]</xf2:XPath>
        </dsig:Transform>
      <dsig:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature" />
      <dsig:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
    </dsig:Transforms>
    <dsig:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
    <dsig:DigestValue>CIF6Qt/xrwTsICP4o5kJGmK+K6Q=</dsig:DigestValue>
  </dsig:Reference>
</dsig:SignedInfo>
  <dsig:SignatureValue>EYZGtd+QUhhe5U0zH/3Q0OU76umCm5FrE6mpiIHAdHGqpbApT3d22gHYXfdhlecO
LxrgXbV5ejjH751EcZ6KxXb2cGuDiszkuYuAHy9MTKTL4GEUQo+97JJ86PQFpOs
kG01oErkgOGjx5efB5oDrSPJ2zk+PhsQw8Eqo8seFx8=</dsig:SignatureValue>
  <dsig:KeyInfo>
    <dsig:X509Data>
      <dsig:X509Certificate>MIIEpjCCA46gAwIBAgI DAMrDMA0GCSqGSIb3DQEBBQUAMIGfMQswCQYDVQQGEwJB
VDFIMEYGA1UEChMQS1UcnVzdCBHZXMuIGYulFNpY2hlcmlhLRzc3lzdGVtZSBp
bSBibGVrdHlulERhdGVudmVya2VociBHbWJIMSlwIAYDVQQLExlhLXNpZ24tY29y
cG9yYXRILWxpZ2h0LTAxMSlwiAYDVQQDExlhLXNpZ24tY29ycG9yYXRILWxpZ2h0
LTAxMB4XDTA0MTAwNjA5MDY1MVVoXDTA3MTAwNjA5MDY1MVoweTELMAKGA1UEBhMC
YXQxJDAiBgNVBAoTG1NwYXJrYXNzZW4gRGF0ZW5kaWVuc3QgR21iSDEQMA4GA1UE
CjxMHU3BhcmRhdEhMBkGA1UEAxMSU3BhcmRhdC1lchMTU2InLTAxMRUwEwYDVQQF
Eww2MTk0OTM5ODUxNzcwZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAKkmbmACF
B8spL+bLKg+E5h14i/D4vALmrrBWSc98fVLB87rZ/VK+cuXlh2ug3P5cOESQwCo
buVfLgn0v+WrrH/JLgZn7KRc14j/VM0vITURkiEkDnQjUf5tDUCKG/ft5uJqjH9

```

```

IK8+DtKJgsSd0OYyB9Ewzi/8t33M9oOAI7tjAgMBAAGjggGSMIIBjAJBgNVHRME
AjAAMBEGA1UdDgQKBAhP62yLyG0dIDBYBgNVHSAEUTBPME0GBYooABEBBwEwQjBA
BggrBgEFBQCcARY0aHR0cDovL3d3dy5hLXRydXN0LmF0L2RvY3MvY3AvYS1zaWdu
LWNvcnBvcnF0ZS1saWdodDATBgNVHSMEDDAKGAhOnn/UL8kfHzB/BggrBgEFBQCcB
AQRzMEwRgYIKwYBBQUHMAKGOmh0dHA6Ly93d3cuYS10cnVzdC5hdC9jZj0cy9h
LXNpZ24tY29ycG9yYXRILWxpZ2h0LTAxYS5jcnQwJwYIKwYBBQUHMAGGG2h0dHA6
Ly9vY3NwLmEtdHJ1c3QuYXQvb2NzcDAOBgNVHQ8BAf8EBAMCBLAwbgYDVR0fBGcw
ZTBjoGGgX4ZdbGRhcDovL2xkYXAuYS10cnVzdC5hdC9vdT1hLXNpZ24tY29ycG9y
YXRILWxpZ2h0LTAxLW89QS1UcnVzdCxiPUFUP2NlcnRpZmlyYXRlcmV2b2NhdGlv
bmxc3Q/MAOGCSqGSIb3DQEBAQA4IBAQBsr+16GIfUploYYae39pVLdz4holom
nbx3k9/KwjKReE2djJeRDk/46BWUf19V/xPHOJ4GjaAU0WJpO7ITCsQeiVdUJbB5
pgHYFyHjgOyKz9DwCtcpWzdS3luspSJwrYhDd/Hk6+FxstDaKPN/O3Dj/7FcBChR
hldvrCXymk2ah4ezl+B2hQ+n2pWtXkPvDXCUqEjOqAnTc1FBk34CBISUphul0W5
G/NUtmIc/HzjKfSFDZvSfRmCZmQRq4IIWYhSua7RuP93iAn8zrJ71PGzIAHowkk
Hhchb9ZpjI93slX1Qa0hH+4AQ6ImvHaBwioG0so8Gd/Vu2PQI1LBU9No</dsig:X509Certificate>
</dsig:X509Data>
</dsig:KeyInfo>
</dsig:Signature>
</eps:PaymentConfirmationDetails>
</eps:BankConfirmationDetails>
</eps:EpsProtocolDetails>

```

epsp:SessionID

Explanation: session ID generated by the bank
Status: MANDATORY
Data type: x..512

eps:PaymentConfirmationDetails

Status: MANDATORY

For information on structure and content, please see chapter [6.2.2](#).

6.8 Message: merchant confirmation to payment confirmation message

6.8.1 Payment confirmation – validated by merchant

Explanation the merchant has to confirm the receipt of a **correct payment confirmation** by returning a confirmation message (epsp:BankConfirmationDetails) to the SO.

XML message structure according to XML Schema EPSProtocol-V26.xsd:



figure 6-14: eps:ShopResponseDetails – validated

XML message example

```
<?xml version="1.0" encoding="UTF-8"?>
<eps:EpsProtocolDetails SessionLanguage="DE"
xsi:schemaLocation="http://www.stuzza.at/namespaces/eps/protocol/2014/10 EPSProtocol-V26.xsd"
xmlns:atrul="http://www.stuzza.at/namespaces/eps/austrianrules/2014/10"
xmlns:epi="http://www.stuzza.at/namespaces/eps/epi/2013/02"
xmlns:eps="http://www.stuzza.at/namespaces/eps/payment/2014/10"
xmlns:eps="http://www.stuzza.at/namespaces/eps/protocol/2014/10"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <eps:ShopResponseDetails>
    <eps:SessionId>13212452dea</eps:SessionId>
    <eps:ShopConfirmationDetails>
      <eps:StatusCode>OK</eps:StatusCode>
      <eps:PaymentReferenceIdentifier>120000302122320812201106461</eps:PaymentReferenceIdentifier>
    </eps:ShopConfirmationDetails>
  </eps:ShopResponseDetails>
</eps:EpsProtocolDetails>
```

eps:SessionID

Explanation: session ID generated by the bank

Status: MANDATORY

Data type: x..512

eps:StatusCode

Explanation: status of the eps payment process

Status: MANDATORY

Data type: an..10

eps:PaymentReferenceIdentifier

Explanation: unique payment reference assigned by the bank

Status: MANDATORY

Data type: an..28

6.8.2 Payment confirmation – NO validation by merchant

Explanation: the merchant answers with an error message after the receipt of an **incorrect payment confirmation** (e.g. invalid XML structure, invalid electronic signature) to the SO.

XML message structure according to XML Schema EPSProtocol-V26.xsd:

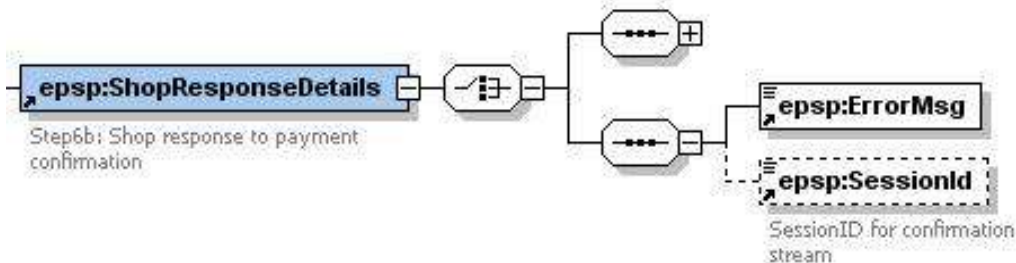


figure 6-15: epsp:ShopResponseDetails - non validated

XML message example

```
<?xml version="1.0" encoding="UTF-8"?>
<epsp:EpsProtocolDetails SessionLanguage="DE"
xsi:schemaLocation="http://www.stuzza.at/namespaces/eps/protocol/2014/10 EPSProtocol-V26.xsd"
xmlns:atrul="http://www.stuzza.at/namespaces/eps/austrianrules/2014/10"
xmlns:epi="http://www.stuzza.at/namespaces/eps/epi/2013/02"
xmlns:eps="http://www.stuzza.at/namespaces/eps/payment/2014/10"
xmlns:epsp="http://www.stuzza.at/namespaces/eps/protocol/2014/10"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <epsp:ShopResponseDetails>
    <epsp:ErrorMsg>XML Message nicht korrekt und validerbar</epsp:ErrorMsg>
    <epsp:SessionId>13212452dea</epsp:SessionId>
  </epsp:ShopResponseDetails>
</epsp:EpsProtocolDetails>
```

epsp:ErrorMsg

Explanation: error information by the merchant

Status: MANDATORY

Data type: an..255

epsp:SessionID

Explanation: session ID generated by the bank

Status: OPTIONAL

Data type: x..512

In the data element "ErrorMsg" the merchant can specify to the SO the error that occurred, e.g. "XML message can not be validated".

6.9 Message: eps TransactionDetailsRequest

Depending on the scenario (see Chapter 7), the buyers bank can request an eps-payment request (eps4mobile transaction data) from the SO via the epsp: TransactionDetailsRequest.

Note: Transaction details can be requested multiple times from different banks. Only when SO receives confirmation for the requested transaction, no more TransactionDetails will be delivered and the request will be terminated with the code 020.

XML message structure according to XML Schema EPSProtocol-V26.xsd:

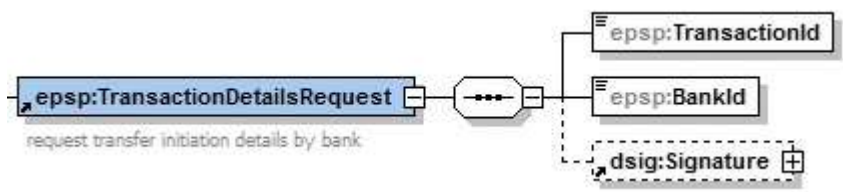


Figure 6-16: epsp: TransactionDetailsRequest

XML message example

```
<?xml version="1.0" encoding="UTF-8"?>
<epsp:EpsProtocolDetails SessionLanguage="DE"
xsi:schemaLocation="http://www.stuzza.at/namespaces/eps/protocol/2014/10 EPSPProtocol-V26.xsd"
xmlns:atrule="http://www.stuzza.at/namespaces/eps/austrianrules/2014/10"
xmlns:epi="http://www.stuzza.at/namespaces/eps/epi/2013/02"
xmlns:eps="http://www.stuzza.at/namespaces/eps/payment/2014/10"
xmlns:epsp="http://www.stuzza.at/namespaces/eps/protocol/2014/10"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <epsp:TransactionDetailsRequest>
    <epsp:TransactionId>epsHXOSINN8T</epsp:TransactionId>
    <epsp:BankId>HYPTAT22XXX</epsp:BankId>
    <dsig:Signature Id="hotVault" xmlns:dsig="http://www.w3.org/2000/09/xmldsig#">
      <dsig:SignedInfo>
        <dsig:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
        <dsig:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1" />
        <dsig:Reference Id="reference-data-0" URI="">
          <dsig:Transforms>
            <dsig:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature" />
          </dsig:Transforms>
          <dsig:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
          <dsig:DigestValue>hr37g9hhaOxCrutyMAX8XIK1atk=</dsig:DigestValue>
        </dsig:Reference>
      </dsig:SignedInfo>
      <dsig:SignatureValue>OOMR5YBr4ZiQpAxfbU02Qi7t1F4prUlb+Y2DafoKwEZXuSpALFiQbOzCMXRHis
77hb2Hc6R68n1x/PriOdnGMCRiGIZl7UywQbZLhRnYqy+hKB/fmaoPI5NafTDp
jnCdN3mL3Rbc7+2gqFjhk1XujJeIYN5dwwrtMefAuQ=</dsig:SignatureValue>
      <dsig:KeyInfo>
        <dsig:X509Data>
          <dsig:X509Certificate>MIIEpjCCA46gAwIBAgIDAMrDMA0GCSqGSIB3DQEBBQUAMIGfMQswCQYDVQQGEwJB
VDFIMEYGA1UEChM/QS1UcnVzdCBHZXMuIGYUuIFNpY2hlcmlhaXRzc3lzdGVtZSBp
bSBibGVrdHIuIERhdGVudmVya2VociBhbWJIMSIAIDVQVQLExlhLXNpZ24tY29y
cG9yYXRILWxpZ2h0LTAxMSIwIAIDVQVQDExlhLXNpZ24tY29yYXpZ2h0
LTAxMBA4XD0MTAwNjA5MDY1MVoXDTA3MTAwNjA5MDY1MVoweTELMAkGA1UEBhMC
YXQxJDAiBgNVBAoTG1NwYXRyXNzZW4gRGF0ZW5kaVWuc3QgR21iSDEQMA4GA1UE
CxMhU3BhcmRhdDEBMBkGA1UEAxMSU3BhcmRhdC1lchMmTU2lnLTAxMRUwEwYDVQ==</dsig:X509Certificate>
        </dsig:X509Data>
      </dsig:KeyInfo>
    </dsig:Signature>
  </epsp:TransactionDetailsRequest>
</epsp:EpsProtocolDetails>
</?xml>
```



```
Eww2MTk0OTM5ODUxNzcwZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAKkbmACF
B8spL+bLKg+E5h14i/D4vALmrrBWSc98fVLB87rZ/TVK+cuXlh2ug3P5cOESQwCo
buVfLGn0V+WrrH/JLgZn7KRc14j/VM0vITURkiEkDnQgjUF5tDUCKG/ft5uJqjH9
IK8+DtKJgsSd0OYyB9Ewzi/8t33M9oOAI7tjAgMBAAGjggGSMIIbjAJBgNVHRME
AjaAMBEGA1UdDgQKBAhP62yLyG0dIDBYBgNVHSAEUTBPME0GByoABEBBwEwQjBA
BggrBgEFBQcCARY0aHR0cDovL3d3dy5hLXRydXN0LmF0L2RvY3MvY3AvYS1zaWdu
LWNvcnBvcnF0ZS1saWdodDATBgNVHSMEDDAKgAhOnn/UL8kfHzB/BggrBgEFBQcB
AQRzMHEwRgYIKwYBBQUHMAKG0mh0dHA6Ly93d3cuYS10cnVzdC5hdC9jZjZ0cy9h
LXNpZ24tY29ycG9yYXRILWxpZ2h0LTAxYS5jcnQwJwYIKwYBBQUHMAGGG2h0dHA6
Ly9vY3NwLmEtdHJ1c3QuYXQvb2NzcDAOBgNVHQ8BAf8EBAMCBLAwbgYDVR0fBGcw
ZTBjoGGgX4ZdbGRhcDovL2xkYXAuYS10cnVzdC5hdC9vdT1hLXNpZ24tY29ycG9y
YXRILWxpZ2h0LTAxLW89QS1UcnVzdCxiPUFUP2NlcnRpZmljYXRlcmV2b2NhdGlv
bmxc3Q/MAOGCSqGSIb3DQEBBQUAA4IBAQBsr+16GIfUploYYae39pVLdz4holom
nbx3k9/KwjKReE2djJeRDk/46BWUf19V/xPHOJ4GjaAU0WJpO7ITCsQeiVdUJbB5
pgHYFyHjgOyKz9DwCtcpWzdS3luspSjwrYhDd/Hk6+FxstDaKPN/O3Dj/7FcBChR
hldvrCXYmk2ah4ezi+B2hQ+n2pWttXkPvDXCUqEjOqAnTc1FBk34CBISUphul0W5
G/NUtmlc/HrzjksFDZvSfRmCZmQRq4llWYhSua7RuP93iAn8zrJ71PGziAHowkk
Hhchb9Zpj93slX1Qa0hH+4AQ6lmvHaBwioG0so8Gd/Vu2PQl1LBU9No</dsig:X509Certificate>
</dsig:X509Data>
</dsig:KeyInfo>
</dsig:Signature>
</epsp:TransactionDetailsRequest>
</epsp:EpsProtocolDetails>
```

epsp:TransactionID

Explanation: for the clear identification of an eps transaction in the further process.

The TransactionID is generated by the eService Scheme Operator and forwarded to the bank as well as returned to the merchant. The transaction ID can be used to query transaction data via an XML query interface.

Status: MANDATORY

Data type: an 36
z.B. epsHXOSINN8T

epsp:BankID

Explanation: BIC of the bank or BIC of the main institution of a bank group

Status: MANDATORY

Data type: an 11

6.10 Message: eps TransactionDetailsResponse

Der SO answers the epsp:TransactionDetailsRequest with an epsp:TransactionDetailsResponse Message.

XML message structure according to XML Schema EPSProtocol-V26.xsd:

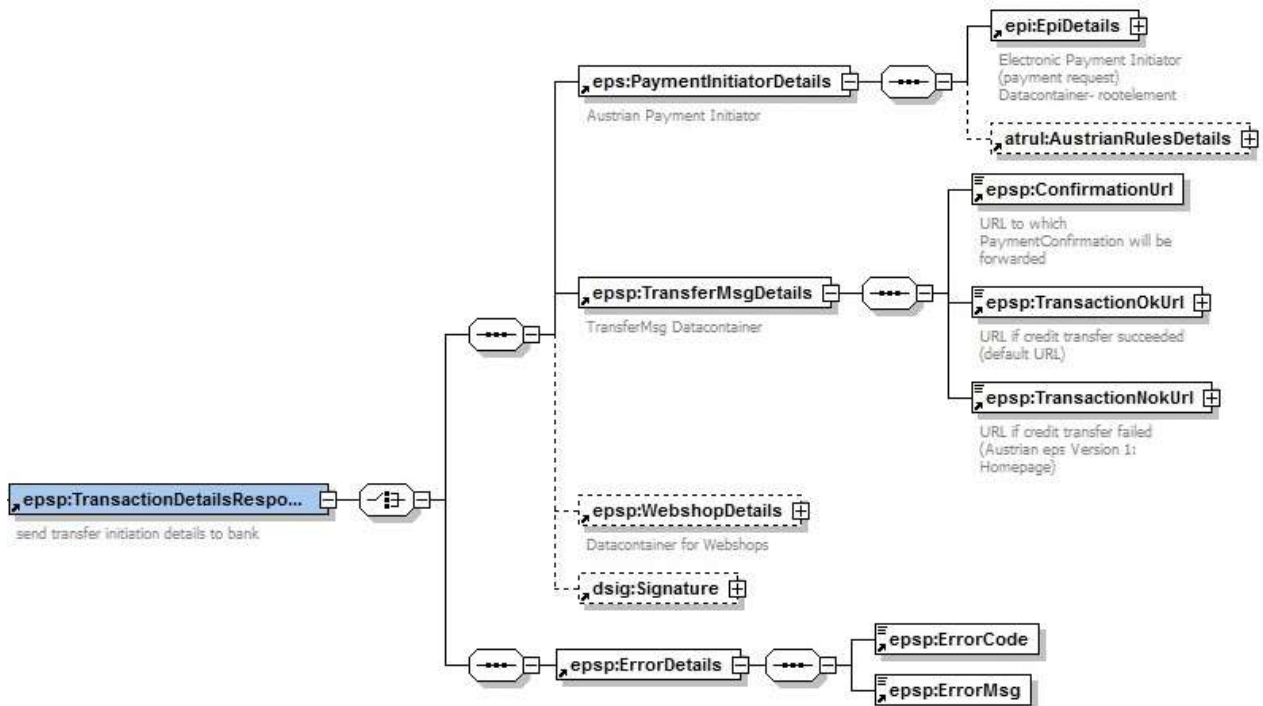


Figure 6-17: epsp: TransactionDetailsResponse

XML message example

```
<?xml version="1.0" encoding="UTF-8"?>
<eps:EpsProtocolDetails SessionLanguage="DE"
xsi:schemaLocation="http://www.stuzza.at/namespaces/eps/protocol/2014/10 EPSProtocol-V26.xsd"
xmlns:atrul="http://www.stuzza.at/namespaces/eps/austrianrules/2014/10"
xmlns:epi="http://www.stuzza.at/namespaces/eps/epi/2013/02"
xmlns:eps="http://www.stuzza.at/namespaces/eps/payment/2014/10"
xmlns:epsp="http://www.stuzza.at/namespaces/eps/protocol/2014/10"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <eps:TransactionDetailsResponse>
    <eps:PaymentInitiatorDetails>
      <epi:EpiDetails>
        <epi:IdentificationDetails>
          <epi:Date>2014-10-28</epi:Date>
          <epi:ReferenceIdentifier>1234567890ABCDEF</epi:ReferenceIdentifier>
        </epi:IdentificationDetails>
        <epi:PartyDetails>
          <epi:BfiPartyDetails>
            <epi:BfiBicIdentifier>GAWIATW1XXX</epi:BfiBicIdentifier>
          </epi:BfiPartyDetails>
          <epi:BeneficiaryPartyDetails>
            <epi:BeneficiaryNameAddressText>Max Mustermann</epi:BeneficiaryNameAddressText>
          </epi:BeneficiaryPartyDetails>
        </epi:PartyDetails>
      </epi:EpiDetails>
    </eps:PaymentInitiatorDetails>
    <epsp:TransferMsgDetails>
      <epsp:ConfirmationUri>
      <epsp:TransactionOkUri>
      <epsp:TransactionNokUri>
    </epsp:TransferMsgDetails>
    <epsp:ErrorDetails>
      <epsp:ErrorCode>
      <epsp:ErrorMsg>
    </epsp:ErrorDetails>
  </eps:TransactionDetailsResponse>
</eps:EpsProtocolDetails>
```



```

<epi:PaymentInstructionDetails>
  <epi:RemittanceIdentifier>AT1234567890XYZ</epi:RemittanceIdentifier>
  <epi:InstructedAmount AmountCurrencyIdentifier="EUR">150.00</epi:InstructedAmount>
  <epi:ChargeCode>SHA</epi:ChargeCode>
  <epi:DateOptionDetails DateSpecificationCode="CRD">
    <epi:OptionDate>2014-10-28</epi:OptionDate>
    <epi:OptionTime>11:00:00-05:00</epi:OptionTime>
  </epi:DateOptionDetails>
</epi:PaymentInstructionDetails>
</epi:EpiDetails>
<atrul:AustrianRulesDetails>
  <atrul:DigSig>SIG</atrul:DigSig>
  <atrul:ExpirationTime>2014-10-28T09:30:47Z</atrul:ExpirationTime>
  <atrul:StatusMsgEnabled>true</atrul:StatusMsgEnabled>
</atrul:AustrianRulesDetails>
</eps:PaymentInitiatorDetails>
<epsp:TransferMsgDetails>
  <epsp:ConfirmationUrl>http://10.18.70.8:7001/vendorconfirmation</epsp:ConfirmationUrl>
  <epsp:TransactionOkUrl
TargetWindow="Mustershop">http://10.18.70.8:7001/transactionok?danke.asp</epsp:TransactionOkUrl>
  <epsp:TransactionNokUrl
TargetWindow="Mustershop">http://10.18.70.8:7001/transactionnok?fehler.asp</epsp:TransactionNokUrl>
</epsp:TransferMsgDetails>
<epsp:WebshopDetails>
  <epsp:WebshopArticle ArticleCount="1" ArticleName="Toaster" ArticlePrice="150.00"/>
</epsp:WebshopDetails>
<dsig:Signature Id="hotVault" xmlns:dsig="http://www.w3.org/2000/09/xmldsig#">
  <dsig:SignedInfo>
    <dsig:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
    <dsig:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1" />
    <dsig:Reference Id="reference-data-0" URI="">
      <dsig:Transforms>
        <dsig:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature" />
      </dsig:Transforms>
    </dsig:Reference>
    <dsig:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
    <dsig:DigestValue>hr37g9hhaOxCrutyMAX8XIK1atk=</dsig:DigestValue>
  </dsig:SignedInfo>
  <dsig:SignatureValue>OOMR5YBr4ZiQpAhxfbU02Qi7t1F4prUlb+Y2DafoKwEZXuSpALFiQbOzCMXRHisa
77hbg2Hc6R68n1x/PrIOd+nGMCRIgIz7UyWqBZLhRnYqy+hKB/fmaoPI5NafTDP
jnCdN3mL3Rbc7+2gqFjhlk1XuJJelYN5dwwrtMefAuQ=</dsig:SignatureValue>
  <dsig:KeyInfo>
    <dsig:X509Data>
      <dsig:X509Certificate>MIIEpjCCA46gAwIBAgIDAMrDMA0GCSqGSIb3DQEBBQUAMIGfMQswCQYDVQQGEwJB
VDFIMEYGA1UEChM/QS1UcnVzdCBHZXMuIGYulFNpY2hlcmlhlaXRzc3lzdGVtZSBp
bSBibGVrdHlUeRhdGVudmVya2VociBhbWJIMSJwIAYDVQQLExlhLXNpZ24tY29y
cG9yYXRILWxpZ2h0LTAxMSIwIAYDVQQDExlhLXNpZ24tY29ycG9yYXRILWxpZ2h0
LTAxMB4XDTA0MTAwNjA5MDY1MVoXDTA3MTAwNjA5MDY1MVoweTELMAkGA1UEBhMC
YXQxJDAiBgNVBAoTG1NwYXJrYXNzZW4gRGF0ZW5kaWVuc3QgR21iSDEQMA4GA1UE
CxMHU3BhcmRhdDEbMBkGA1UEAxMSU3BhcmRhdC1lcHMtU2lnLTAxMRUwEwYDVQQF
Ewww2MTk0OTM5ODUxNzZwZG8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAKkbmACF
B8spL+bLKg+E5h14i/D4vALmrrBWSc98fVLB87rZ/TVK+cuXlh2ug3P5cOESQwCo
buVfLGN0V+WrrH/JLgZn7KRc14j/VM0vITURkiEkDnQgjUF5tDUCKG/ft5uJqjH9
IK8+DtKJgsSd0OYyB9Ewzi/8t33M9oOAI7tjAgMBAAGjggGSMIIIBjAIBgNVHRME
AjaAMBEGA1UdDgQKBAHP62yLyG0dIDBYBgNVHSAEUTBPME0GBYoABEBBwEwQjBA
BggrBgEFBQcCARY0aHR0cDovL3d3dy5hLXRydXN0LmF0L2RvY3MvY3AvYS1zaWdu
LWNvcnBvcnF0ZS1saWdodDAtBgNVHSMEDDAKGAhOnn/UL8kfHzB/BggrBgEFBQcB
AQRzMHEwRgYIKwYBBQUHMAKGOMh0dHA6Ly93d3cuYS10cnVzdC5hdC9jZjJ0cy9h
LXNpZ24tY29ycG9yYXRILWxpZ2h0LTAxYS5jcnQwJwYIKwYBBQUHMAGGG2h0dHA6
Ly9vY3NwLmEtdHJ1c3QuYXQvb2NzcDAOBgNVHQ8BAf8EBAMCBLAwbGyDVR0fBGcw
ZTBjoGGgX4ZdbGRhcDovL2xkYXUyS10cnVzdC5hdC9vdT1hLXNpZ24tY29ycG9y
YXRILWxpZ2h0LTAxLGA89QS1UcnVzdCxpUFUP2NlcnRpZmljYXRlcmV2b2NhdGlv
bmxc3Q/MA0GCSqGSIb3DQEBBQUAA4IBAQBsr+16GIfUploYYae39pVLdz4holom
nbx3k9/KwjReE2djJeRDK/46BWUf19V/xPHOJ4GjaAU0WJpO7ITCsQeiVdUJbB5
pgHYFyHjgOyKz9DwCtcpWzdS3luspSjwrYhDd/Hk6+FxstDaKPN/O3Dj/7FcBChR
hldrCYmk2ah4ezi+B2hQ+n2pWtXkPvDXCQeJQAnTc1FBk34CBISUphul0W5
G/NUtlmc/HzjkfSFDZvSfRmCZmQRq4IIWYhSua7RuP93iAn8zrJ71PGzIAHowkk
Hhchb9ZpjI93slX1Qa0hH+4AQ6ImvHaBwioG0so8Gd/Vu2PQ11LBU9No</dsig:X509Certificate>
    </dsig:X509Data>
  </dsig:KeyInfo>
</dsig:Signature>
</epi:PaymentInstructionDetails>

```

```

    </dsig:KeyInfo>
  </dsig:Signature>
</eps:TransactionDetailsResponse>
</eps:EpsProtocolDetails>

```

6.11 Message: eps StatusMsg

The SO sends the merchant a status to the eps transaction in the eps: StatusMsg message after the bank has requested the eps payment data from the SO.

In order to receive this message from the SO, merchant must set an appropriate ID in the eps payment initiation:

Specification in the AustrianRulesDetails

```

<atrul:AustrianRulesDetails>
  <atrul:PaymentDescription>Test 12 2014</atrul:PaymentDescription>
  <atrul:DigSig>SIG</atrul:DigSig>
  <atrul:ExpirationTime>2014-12-17T16:00:00</atrul:ExpirationTime>
  <atrul:StatusMsgEnabled>true</atrul:StatusMsgEnabled>
</atrul:AustrianRulesDetails>

```

The data element <StatusMsgEnabled> can have the values *true* (message will be sent) or *false* (message will not be sent) and is generally optional. If the field is not set by the merchant, it will be interpreted as false.

The merchant can use the information to show the buyer the status of payment in the POS.

XML message structure according to XML Schema EPSProtocol-V26.xsd:

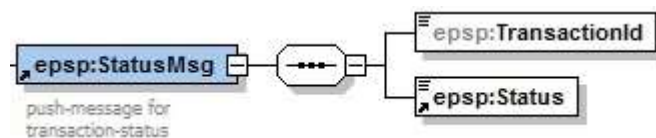


Figure 6-18: eps: StatusMsg

XML message example

```

<?xml version="1.0" encoding="UTF-8"?>
<eps:EpsProtocolDetails SessionLanguage="DE"
xsi:schemaLocation="http://www.stuzza.at/namespaces/eps/protocol/2014/10 EPSProtocol-V26.xsd"
xmlns:atrul="http://www.stuzza.at/namespaces/eps/austrianrules/2014/10"
xmlns:epi="http://www.stuzza.at/namespaces/eps/epi/2013/02"
xmlns:eps="http://www.stuzza.at/namespaces/eps/payment/2014/10"
xmlns:eps="http://www.stuzza.at/namespaces/eps/payment/2014/10"
xmlns:eps="http://www.stuzza.at/namespaces/eps/protocol/2014/10"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <eps:StatusMsg>
    <eps:TransactionId>epsHXOSINN8T</eps:TransactionId>
    <eps:Status>PAYMENT_IN_PROCESS</eps:Status>
  </eps:StatusMsg>
</eps:EpsProtocolDetails>

```

epsp:TansactionID

Explanation: for the clear identification of an eps transaction in the further process.

The TransactionID is generated by the eService Scheme Operator and forwarded to the bank as well as returned to the merchant. The transaction ID can be used to query transaction data via an XML query interface.

Status: MANDATORY

Aufbau: `xsd:pattern value="[a-zA-Z0-9\-\._~]{1,36}"`, z.B. epsHXOSINN8T

epsp:Status

Explanation: Information on the status of the eps transaction

Fixed value: PAYMENT_IN_PROCESS

Status: MANDATORY

Data type: string

6.12 Message: eps ConfirmationStatusRequest

If the submission of the eps payment confirmation (epsp:BankConfirmationDetails) to the merchant has failed, the merchant can later use epsp: ConfirmationStatusRequest to query the status of the transaction.

XML message structure according to XML Schema EPSProtocol-V26.xsd:



Figure 6-19: `epsp:ConfirmationStatusRequest`

The MD5 fingerprint consists of the following attributes:

- Secret (Merchant Pin)
- `epsp:TransactionId`
- `epsp:UserId`

XML message example

```
<?xml version="1.0" encoding="UTF-8"?>
<eps:EpsProtocolDetails SessionLanguage="DE"
xsi:schemaLocation="http://www.stuzza.at/namespaces/eps/protocol/2014/10 EPSProtocol-V26.xsd"
xmlns:atrul="http://www.stuzza.at/namespaces/eps/austrianrules/2014/10"
xmlns:epi="http://www.stuzza.at/namespaces/eps/epi/2013/02"
xmlns:eps="http://www.stuzza.at/namespaces/eps/payment/2014/10"
xmlns:eps="http://www.stuzza.at/namespaces/eps/protocol/2014/10"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <eps:ConfirmationStatusRequest>
    <eps:TransactionId>epsHXOSINN8T</eps:TransactionId>
    <eps:AuthenticationDetails>
      <eps:UserId>HAENDLERID_1234567</eps:UserId>
      <eps:MD5Fingerprint>513600648eadfb78e623f24eb7172589</eps:MD5Fingerprint >
    </eps:AuthenticationDetails>
  </eps:ConfirmationStatusRequest>
</eps:EpsProtocolDetails>
```

6.13 Message: eps ConfirmationStatusResponse

XML message structure according to XML Schema EPSProtocol-V26.xsd:

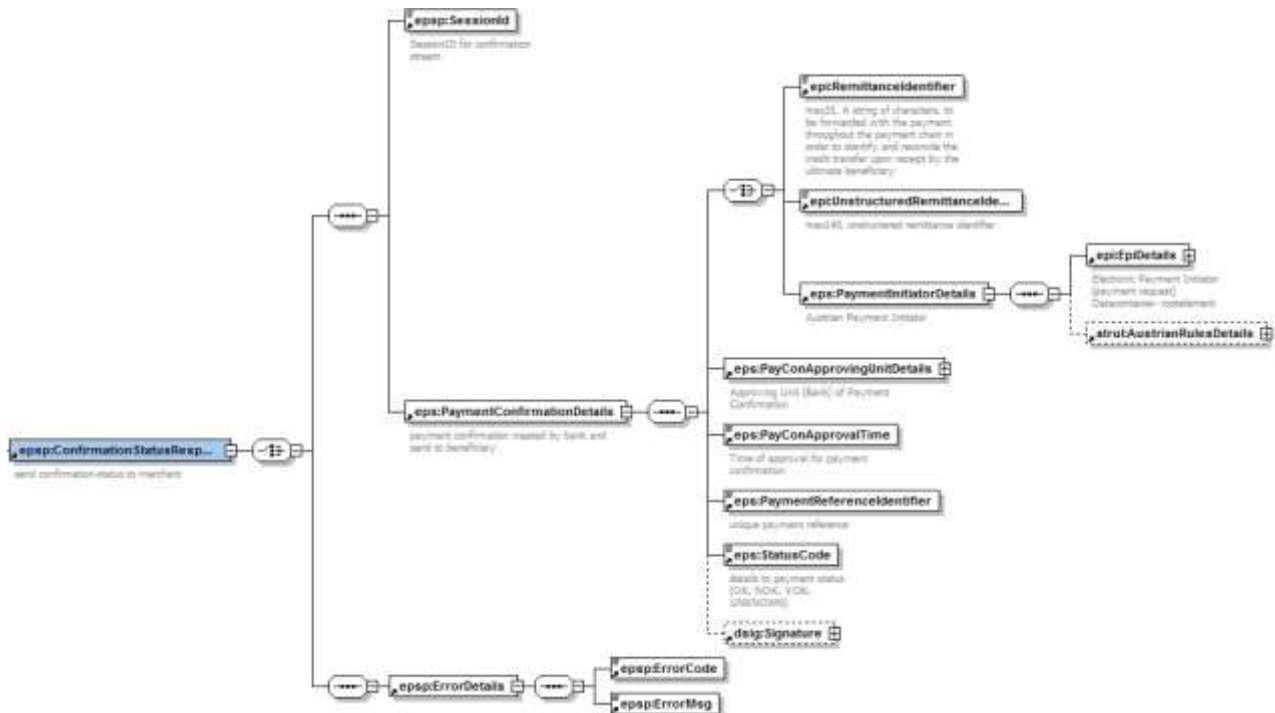


Figure 6-20: eps:ConfirmationStatusResponse

XML message example

```
<?xml version="1.0" encoding="UTF-8"?>
<eps:EpsProtocolDetails SessionLanguage="DE"
xsi:schemaLocation="http://www.stuzza.at/namespaces/eps/protocol/2014/10 EPSProtocol-V26.xsd"
xmlns:atrul="http://www.stuzza.at/namespaces/eps/austrianrules/2014/10"
xmlns:epi="http://www.stuzza.at/namespaces/eps/epi/2013/02"
xmlns:eps="http://www.stuzza.at/namespaces/eps/payment/2014/10"
xmlns:eps="http://www.stuzza.at/namespaces/eps/protocol/2014/10"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <eps:ConfirmationStatusResponse>
    <eps:SessionId>String</eps:SessionId>
    <eps:PaymentConfirmationDetails>
```

```

<eps:PaymentInitiatorDetails>
  <epi:EpiDetails>
    <epi:IdentificationDetails>
      <epi:Date>2014-10-28</epi:Date>
      <epi:ReferenceIdentifier>1234567890ABCDEFG</epi:ReferenceIdentifier>
    </epi:IdentificationDetails>
    <epi:PartyDetails>
      <epi:BfiPartyDetails>
        <epi:BfiBicIdentifier>GAWIATW1XXX</epi:BfiBicIdentifier>
      </epi:BfiPartyDetails>
      <epi:BeneficiaryPartyDetails>
        <epi:BeneficiaryNameAddressText>Max Mustermann</epi:BeneficiaryNameAddressText>
      </epi:BeneficiaryPartyDetails>
    </epi:PartyDetails>
    <epi:PaymentInstructionDetails>
      <epi:RemittanceIdentifier>AT1234567890XYZ</epi:RemittanceIdentifier>
      <epi:InstructedAmount AmountCurrencyIdentifier="EUR">150.00</epi:InstructedAmount>
      <epi:ChargeCode>SHA</epi:ChargeCode>
      <epi:DateOptionDetails DateSpecificationCode="CRD">
        <epi:OptionDate>2014-10-28</epi:OptionDate>
        <epi:OptionTime>11:00:00-05:00</epi:OptionTime>
      </epi:DateOptionDetails>
    </epi:PaymentInstructionDetails>
  </epi:EpiDetails>
  <atrul:AustrianRulesDetails>
    <atrul:DigSig>SIG</atrul:DigSig>
    <atrul:ExpirationTime>2014-10-28T09:30:47Z</atrul:ExpirationTime>
    <atrul:StatusMsgEnabled>true</atrul:StatusMsgEnabled>
  </atrul:AustrianRulesDetails>
</eps:PaymentInitiatorDetails>
<eps:PayConApprovingUnitDetails>
<eps:ApprovingUnitBankIdentifier>HYPTAT22XXX</eps:ApprovingUnitBankIdentifier>
</eps:PayConApprovingUnitDetails>
<eps:PayConApprovalTime>2010-10-28T09:25:47-05:00</eps:PayConApprovalTime>
<eps:PaymentReferenceIdentifier>AT1234567890XYZ</eps:PaymentReferenceIdentifier>
<eps:StatusCode>OK</eps:StatusCode>
<dsig:Signature Id="hotVault" xmlns:dsig="http://www.w3.org/2000/09/xmldsig#">
  <dsig:SignedInfo>
    <dsig:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
    <dsig:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1" />
    <dsig:Reference Id="reference-data-0" URI="">
      <dsig:Transforms>
        <dsig:Transform Algorithm="http://www.w3.org/2002/06/xmldsig-filter2">
          <xf2:XPath Filter="intersect"
xmlns:eps="http://www.stuzza.at/namespaces/eps/payment/20031001" xmlns:xf2="http://www.w3.org/2002/06/xmldsig-filter2">here()/ancestor::eps:PaymentConfirmationDetails[1]</xf2:XPath>
          </dsig:Transform>
          <dsig:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature" />
          <dsig:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
        </dsig:Transforms>
        <dsig:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
        <dsig:DigestValue>CIF6Qt/xrwTslCP4o5kJGmK+K6Q=</dsig:DigestValue>
      </dsig:Reference>
    </dsig:SignedInfo>
    <dsig:SignatureValue>EYZGtd+QUhhe5U0zH/3Q0OU76umCm5FrE6mpilHAdHGqpbApT3d22gHYXfdhlecO
LxrgXbV5ejjH751EcZ6KxXb2cGuDiszkuYuAHy9MTKTL4GEUQo+97JJ86PQFpfOs
kG01oErkgOGjx5efB5oDrSPJ2zk+PhsQw8Eqo8seFx8=</dsig:SignatureValue>
  <dsig:KeyInfo>
    <dsig:X509Data>
      <dsig:X509Certificate>MIIEpjCCA46gAwIBAgIDAMrDMA0GCSqGSIb3DQEBBQUAMIGfMQswCQYDVQQGEwJB
VDFIMEYGA1UEChM/QS1UcnVzdCBHZXMuIGYUfNpY2hlcmlhXARzc3lzdGVtZSBp
bSBibGVrdHlulERhdGVudmVya2VociBHBwJlMSlwlAYDVQQLExlhLXNpZ24tY29y
cG9yYXRILWxpZ2h0LTAxMSlwlAYDVQQDExlhLXNpZ24tY29ycG9yYXRILWxpZ2h0
LTAxMB4XDTA0MTAwNjA5MDY1MVoXDTA3MTAwNjA5MDY1MVoweTELMAkGA1UEBhMC
YXQxJDAiBgNVBAoTG1NwYXRyYXNzZW4gRGF0ZW5kaWVuc3QgR21iSDEQMA4GA1UE
CzMHU3BhcmRhdDEbMBkGA1UEAxMSU3BhcmRhdC1lcHMtU2lnLTAxMRUwEwYDVQQF
Eww2MTk0OTM5ODUxNzcwZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAKkbnACF
B8spL+bLkg+E5h14i/D4vALmrrBWS98fVLB87rZ/TVK+cuXlh2ug3P5cOESQwCo

```

buVfLGn0V+WrrH/JLgZn7KRc14j/VM0vITURkiEkDnQgjUF5tDUckG/ft5uJqjH9
IK8+DtKJgsSd0OYyB9Ewzi/8t33M9oOAI7tjAgMBAAGiggGSMIIBjjAJBgNVHRME
AjaAMBEGA1UdDgQKBAhP62yLyG0dIDBYBgNVHSAEUTBPMEOGBYoABEBBwEwQjBA
BggrBgEFBQcCARY0aHR0cDovL3d3dy5hLXRydXN0LmF0L2RvY3MvY3AvYS1zaWdu
LWNvcnBvcnF0ZS1saWdodDATBgNVHSMEDDAKGAhOnn/UL8kfHzB/BggrBgEFBQcB
AQRzMHEwRgYIKwYBBQUHMAKGOmh0dHA6Ly93d3cuYS10cnVzdC5hdC9jZXJ0cy9h
LXNpZ24tY29ycG9yYXRILWxpZ2h0LTAxYS5jcnQwJwYIKwYBBQUHMAGGG2h0dHA6
Ly9vY3NwLmEtdHJ1c3QuYXQvb2NzcDAOBgNVHQ8BAf8EBAMCBLAwbGVDVR0fBGcw
ZTBjoGGgX4ZdbGRhcDovL2xkYXAuYS10cnVzdC5hdC9vdT1hLXNpZ24tY29ycG9y
YXRILWxpZ2h0LTAxLW89QS1UcnVzdCxiPUFUP2NlcnRpZmljYXRlcmV2b2NhdGlv
bmxc3Q/MAOGCSqGSIb3DQEBBQUAA4IBAQBsr+16GIfUploYYae39pVLdz4holom
nbx3k9/KwjkReE2djJeRDk/46BWUfl9V/xPHOJ4GjaAU0WJpO7ITCsQeiVdUJbB5
pgHYFyHjgOyKz9DwCtcpWzdS3luspSJwrYhDd/Hk6+FxstDaKPN/O3Dj/7FcBChR
hldvrCXymk2ah4ezl+B2hQ+n2pWttXkPvDXCUqEjOqAnTc1FBk34CBISUphul0W5
G/NUtmIc/HzjkfSFDZvSfRmCZmQRq4IIWYhSua7RuP93iAn8zrJ71PGzIAHowkk
Hhchb9Zpjl93slX1Qa0hH+4AQ6ImvHaBwioG0so8Gd/Vu2PQI1LBU9No</dsig:X509Certificate>
</dsig:X509Data>
</dsig:KeyInfo>
</dsig:Signature>
</eps:PaymentConfirmationDetails>
</epsp:ConfirmationStatusResponse>
</epsp:EpsProtocolDetails>

7 EPS WORKFLOW

7.1 eps workflow without central SO debtor bank selection

The following steps describe the eps workflow without using the dynamic bank selection service offered by the SO.

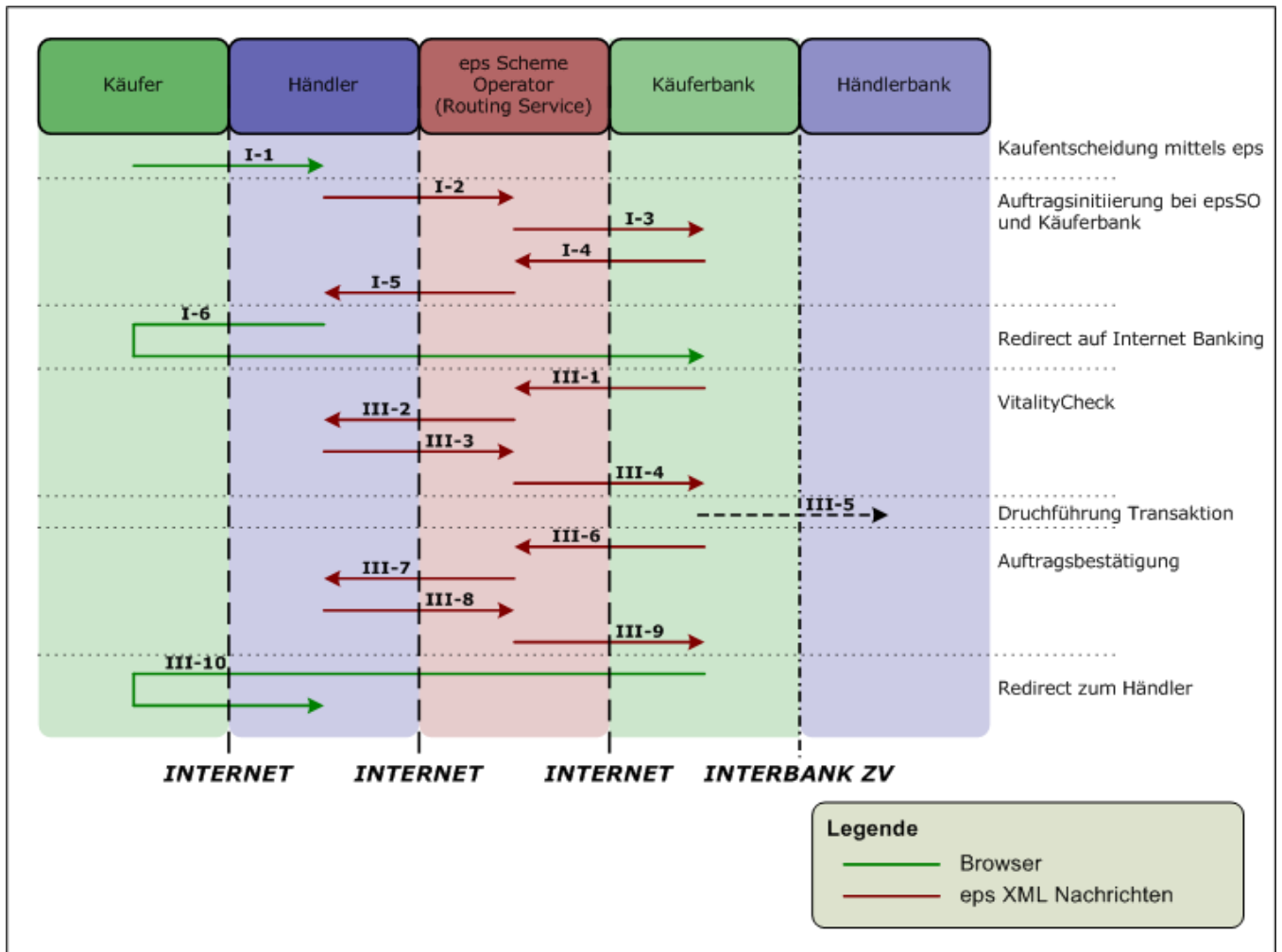


figure 7-1: eps workflow

Bank selection directly in the merchants webshop

The merchant has to implement the eps standard and workflow following the eps Implementation Guideline (for more information see also <http://eservice.stuzza.at/en>).

The merchant has to create the eps payment initiation message and to forward this eps XML message directly to the SO by using the central SO URL:

```
<EPS SO DOMAIN>/<appPath>/<initUri>/<BANKGRUPPENPARAMETER>?<ALLGEMEINE PARAMETER>
```

Example: https://routing.eps.or.at/appl/epsSO/transinit/eps/v2_6/bgrp1-uid (or https://routing.eps.or.at/appl/epsSO/transinit/eps/v2_6/bgrp1-uid)

By using the XML Scheme epsSOBankListProtocol.xsd, send to https://routing.eps.or.at/appl/epsSO/data/haendler/v2_6, the merchant will receive an up-to-date list of all registered eps debtor banks (including the BANKGRUPPEN-UID).

The merchant must provide ALL available eps debtor banks in the bank selection within the webshop!

Example "Testbank":

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<epsSOBankListProtocol xmlns="http://www.eps.or.at/epsSO/epsSOBankListProtocol/201008">
<bank>
  <bic>TESTBANKXXX</bic>
  <bezeichnung>Testbank</bezeichnung>
  <land>AT</land>
  <epsUrl>https://routing.eps.or.at/appl/epsSO/transinit/eps/v2\_5/23ea3d14-278c-4e81-a021-d7b77492b611</epsUrl>
  <zahlungsweiseNat terminueberweisung="false">EPG</zahlungsweiseNat>
</bank>
</epsSOBankListProtocol>
```

By sending the eps payment initiation message to https://routing.eps.or.at/appl/epsSO/transinit/eps/v2_6/23ea3d14-278c-4e81-a021-d7b77492b611 (**as an example!!**), the "Testbank" will be addressed via the SO.

Alternative selection process:

The merchant can also address the debtor bank directly within the eps payment initiation message by forwarding the BIC of the debtor bank in the <epi:OrderingCustomerOfiIdentifier>.

The eps XML message then needs to be forwarded to the SO (https://routing.eps.or.at/appl/epsSO/transinit/eps/v2_6).

7.1.1 Step I-1: debtor bank selection

Having agreed on terms of purchase the buyer selects "eps-Überweisung" as online payment method.

The buyer then selects his debtor bank at the web-shop. Here either the bank-selection of the merchant or the bank-selection-widget (see the chapter 7.3 usage of the Bank selection widget) may be used.

7.1.2 Step I-2: eps payment initiation merchant-SO

The merchant creates the eps payment initiation message (step 2a „epsp:TransferInitiatorDetails“ according to the eps XML Schema) and sends the eps XML message to the central SO URL.

The SO checks and validates the incoming eps payment initiation message.

The following validation will be carried out by the eps SO:

- Content-Type
- XML syntactic validity of the eps message
- merchant authorisation
- check whether the credit account in the central merchant database corresponds to the account in the eps payment initiation
- check whether the merchant is allowed (as specified in the merchant database) to use the option for future date in the eps payment initiation
- check of valid expiration time (max. 60 minutes)

In case of a negative validation a XML response message will be sent to the merchant with the particular error code/error message.

7.1.3 Step I-3: eps payment initiation SO-debtor bank

In case of a positive validation an “epsp:TransferInitiatorDetails” message will be sent from the SO to the debtor bank.

In the eps process the SO always acts as merchant between the SO and the debtor bank; merchant authorization is exclusively via electronic signature together with SO UserID "epsSchemeOperator".

The SO replaces the merchant ID (with the SO UserID "epsSchemeOperator") as well as the confirmationURL, TransactionOkUrl and TransactionNokUrl with SO URLs.

Additionally the SO sets in the data element "atrul:Realization" the identifier "EPN" or "EPG". In the data element "atrul:PaymentDescription" the SO forwards the merchant name or alias (if available).

The outgoing eps message is digitally signed by the SO.

In any case the SO always expects a digitally signed eps confirmation message by the debtor bank.

If the "epsp:TransferInitiatorDetails" message is not answered within the timeout by the debtor bank (all messages arriving after a timeout will no longer be processed by the SO, the connection then will be closed when the timeout occurs), an "epsp:BankResponseDetails" message (see 7.1.4) with a corresponding error (**008 – unknown error**) will be returned to the merchant.

In the case of an error the transaction is ended.

Incoming and outgoing messages are stored at the SO and are available through the SO Admin Tool.

7.1.4 Step I-4: Bank response debtor bank-SO

The debtor bank checks the authorisation of the SO (UserID: „epsSchemeOperator“ AND the digital signature), processes and responds in any case to the eps payment initiation message with an “epsp:BankResponseDetails” message.

If no errors occur (ErrorCode “000”) the debtor bank sends the “epsp:BankResponseDetails” message to the SO, including the „epsp:ClientRedirectUrl” as redirect URL to the online-banking application of the debtor bank.

If an error occurs the debtor bank has to send the “epsp:BankResponseDetails” message to the SO including the appropriate error code. In the case of an error the transaction is ended.

Available error codes:

ErrorCode	ErrorMsg
000	No error
003	Currency incorrect/wrong
004	Identification incorrect/wrong
007	Error in XML-Stream
008	Unknown error
009	Internal error
010	IBAN invalid
011	BIC invalid
012	Expiration Time or execution time invalid
013	<i>No cross-border transactions allowed (<u>only to be used by SO!, not by debtor bank!</u>)</i>

In case of an error, transaction is terminated.

Incoming and outgoing messages are stored at the SO and are available through the SO Admin Tool.

7.1.5 Step I-5: bank response SO-merchant

The “epsp:BankResponseDetails” message, created by the debtor bank, is forwarded from the SO to the merchant (workflow step 2b).

In case the debtor bank is not reachable (time out or service) a bank response message together with Error Code 014 will be forwarded to the merchant.

The incoming and outgoing message is journaled by the SO in the original and is available in the SO Admin Tool.

7.1.6 Step I-6: Redirect Online-Banking

In case no error has occurred the merchant will redirect (HTTP-REDIRECT) the buyer to the online-banking application of the debtor bank with the URL (epsp:ClientRedirectUrl) received in the epsp:BankResponseDetails message.

Please note: the buyer still is in a HTTP-REQUESTS status (start of payment process) and awaits a response by the merchant.

7.1.7 Step III-1: payment & vitality check

The buyer identifies himself at the online-banking of the debtor bank by known authentication methods.

After login to the online-banking application the buyer can select his account. The data of the eps payment initiation will be displayed to the buyer within the online-banking; he authorizes the credit transfer with his authorisation method.

Errorhandling AFTER successful login to online-banking (also see 7.1.11)

If an error occurs or the buyer cancels the eps payment, the debtor bank has to create and forward an eps confirmation message with status code "**NOK**" according to step III-6, without sending prior the "epsp:VitalityCheckDetails" message (target URL = URL in data element epsp:ConfirmationUrl)!

In the eps confirmation message the debtor bank has to forward the following mandatory identifier:

- BIC of the debtor bank in data element **<epsp:ApprovingUnitBankIdentifier>**
OR
- any other ID in the data element **< eps:ApprovingUnitIdentifier>**, see also 6.2.2.

Note: without this mandatory information, no eps confirmation message can be processed by the SO and thus forwarded to the merchant! The SO will answer with an error (HTTP status 412) to the debtor bank.

Errorhandling BEFORE successful login to online-banking

If the buyer cancels the eps payment before successful login to the online-banking, the debtor bank has to create and forward an eps confirmation message with status code "**NOK**" according to step III-6, without sending prior the "epsp:VitalityCheckDetails" message (target URL = epsp:ConfirmationUrl)!

In the eps confirmation message the debtor bank has to forward the following mandatory identifier:

- BIC of the debtor bank in data element **<epsp:ApprovingUnitBankIdentifier>**
OR

- any other ID in the data element < **eps:ApprovingUnitIdentifier**>, see also 6.2.2.

Note: without this mandatory information, no eps confirmation message can be processed by the SO and thus forwarded to the merchant! The SO will answer with an error (HTTP status 412) to the debtor bank.

Successful payment authorisation

After authorization of the payment by the debtor/buyer and before processing of the credit transfer the debtor bank has to create the "epsp: **VitalityCheckDetails**" message (to check whether the merchant application is technically reachable for the transfer of the eps payment confirmation message; target URL = URL in data element "epsp: ConfirmationUrl").

The VitalityCheck message has to be forwarded by the debtor bank to the SO.

The incoming message is stored at the SO, available through the SO Admin Tool.

Errorhandling:

debtor bank forwards corrupt or invalid VitalityCheck message to the SO ⇒ no transfer to merchant possible, SO will answer with an error (HTTP status 412) to the debtor bank.

7.1.8 Step III-2: Vitality Check SO-merchant

Having received a valid "epsp:VitalityCheckDetails" message from the debtor bank the SO forwards the VitalityCheck message to the merchant.

7.1.9 Step III-3: confirmation Vitality Check merchant-SO

The SO expects the return of its vitality check message (same message as documented in Step III-2) by the merchant.

Errorhandling

If the "epsp:VitalityCheckDetails" message is not answered by the merchant within or before the timeout, no response will be returned to the Vitality Check message of the debtor bank; the connection is closed.

The SO informs the debtor bank accordingly (HTTP Status 412).

The debtor bank informs the buyer in the online-banking with an error information and redirects with the „epsp:TransactionNokUrl" the buyer back to the merchant (webshop).

No credit transfer will be executed; the TAN is marked as used.

Timeout information is available at <https://eservice.stuzza.at>.

7.1.10 Step III-4: Vitaliy Check SO-debtor bank

The SO forwards to the debtor bank the confirmation of the merchant to the Vitality Check.

The outgoing messages is stored at the SO, available through the SO Admin Tool.

In reaction to the positive confirmation of the Vitality Checks the debtor bank settles the credit transfer transaction (⇒ **Step III-5**).

7.1.11 Step III-6: eps confirmation debtor bank-SO

The debtor bank creates the „`eps:BankConfirmationDetails`” message (when the credit transfer is settled with status code "OK") and forwards the message to the SO.

The “`eps:BankConfirmationDetails`” message has to be be digitally signed by the debtor bank!

With status code "OK" the debtor bank has to forward in the eps data element <eps:ApprovingUnitBankIdentifier> the BIC of the (approving) debtor bank!

Errorhandling:

without the mandatory BIC ⇒ no eps confirmation message can be processed by the SO and thus forwarded to the merchant! ⇒ the SO will answer with an error (HTTP status 412) to the debtor bank.

Errorhandling AFTER successful login to online-banking

If an error occurs or the buyer cancels the eps payment, the debtor bank has to create and forward an eps confirmation message with status code "**NOK**" according to step III-6, without sending prior the “`eps:VitaliyCheckDetails`” message (target URL = URL in data element `eps:ConfirmationUrl`)!

In the eps confirmation message the debtor bank has to forward the following mandatory identifier:

- BIC of the debtor bank in data element **<eps:ApprovingUnitBankIdentifier> OR**
- any other ID in the data element **< eps:ApprovingUnitIdentifier>**, see also 6.2.2.

Note: without this mandatory information, no eps confirmation message can be processed by the SO and thus forwarded to the merchant! The SO will answer with an error (HTTP status 412) to the debtor bank.

The incoming message is stored at the SO, available through the SO Admin Tool.

7.1.12 Step III-7: eps confirmation SO-merchant

The SO receives and processes the incoming digitally signed eps payment confirmation message of the debtor bank and forwards this message unchanged (without any additional signature or processing of the message) to the merchant. Thus the debtor bank signature can be checked by the merchant (see also [6.2.2](#)).

Please note that the guaranteed eps payment option is handled by the eps merchant contract.

Overview eps status codes	payment will be executed	buyer termination or payment not executed
eps merchant contract guaranteed (without option date in future)	OK	NOK
eps merchant contract not guaranteed (with option date in future)	VOK	NOK

The response to the "epsp:ConfirmationUrl" happens with a HTTP-POST. The SO has to use the ConfirmationUrl as provided by the merchant, no further query-parameter may be attached to the ConfirmationUrl from SO!

e.g. a merchant ConfirmationUrl is

<http://dealer.com/paymentModule?mode=confirmation>

⇒ SO will not attach any further query-parameter e.g.

<http://dealer.com/paymentModule?mode=confirmation&addParam=4711>.

7.1.13 Step III-8: confirmation of eps confirmation merchant-SO

The merchant has to confirm the receipt of the eps payment confirmation message (epsp:BankConfirmationDetails message) by returning a confirmation message to the SO.

Errorhandling:

Mapping of merchant response (Vitality-Check and confirmation)

Response from merchant	Response to debtor bank
XML corrupt	No message; HTTP-Status 412 to debtor bank
Timeout	message incl. ErrorMessage

If the "epsp:BankConfirmationDetails" message is not answered by the merchant before or within the timeout (see chapter [4.5](#)) an "epsp:ShopResponseDetails" message with a corresponding error text is forwarded to the debtor bank.

Within the timeout/handling time the debtor bank has up to 3 sending attempts to forward the eps payment confirmation message to the SO.

After the 3rd attempt the buyer will be informed by the debtor bank in the online-banking application that the credit transfer has been executed, but the merchant could not be informed about the eps payment confirmation status.

Further steps are to be agreed between the byuer and the merchant.

7.1.13.1 eps confirmation is validated by the merchant

The merchant has to confirm the receipt of a **correct eps confirmation message** by returning a confirmation message (epsp:ShopResponseDetails) to the SO.

In this confirmation message the merchant has to return those data elements inclusive their content/value which are already part of the eps payment confirmation message (as received from the SO):

epsp:ShopResponseDetails

- epsp:SessionId
- eps:ShopConfirmationDetails + eps:StatusCode + eps:PaymentReferenceIdentifier

7.1.13.2 eps confirmation can not be validated by the merchant

The merchant answers with an error message after the receipt of an **incorrect eps confirmation message** (e.g. invalid XML structure, invalid electronic signature) to the SO.

epsp:ShopResponseDetails

- epsp:ErrorMsg (mandatory) + epsp:SessionId (optional)

In the data element "ErrorMsg" the merchant can specify to the SO the error that occurred, e.g. "XML message can not be validated".

7.1.14 Step III-9: confirmation eps confirmation SO-debtor bank

The SO receives and processes the incoming response to the eps payment confirmation message of the merchant and passes it to the debtor bank.

No manipulation of the data is carried out.

Incoming and outgoing messages are stored at the SO and are available through the SO Admin Tool.

7.1.15 Step III-10: Buyer redirect to merchant

7.1.15.1 correct workflow

After a positive Vitality Check and the merchant's final confirmation of the receipt of the eps payment confirmation message ([7.1.14](#)) the debtor bank redirects the buyer to a URL specified by the merchant (URL for **epsp:TransactionOkUrl**).

7.1.15.2 Incorrect workflow

In the following scenarios the debtor bank redirects the buyer to a URL specified by the merchant (URL for **epsp:TransactionNokUrl**):

- payment could not be executed
- buyer cancelled/aborted the payment process
- a technical error occurred (e.g. merchant does not respond to the vitality check prior to sending the eps payment confirmation message)

Supplementary error codes

The following error codes will be added by the debtor bank and forwarded together with the `epsp:TransactionNokUrl`, e.g. `ERROR1` as parameter in the `epsp:TransactionNokUrl`.

For example

<https://testbeguenstigter.at?epserrorcode=ERROR1>

or

<https://testbeguenstigter.at?parameter1=1&epserrorcode=ERROR1>.

- | | |
|---------------|--|
| ERROR1 | error at forwarding the payment confirmation to the merchant (merchant not accessible) |
| ERROR2 | XML incorrect respectively incorrect electronic signature |
| ERROR3 | buyer cancellation (within the online-banking) |

Based on the forwarded error messages the merchant can handle and redirect the buyer, e.g. redirect to the shop basket.

7.1.16 Overview possible redirect options

Status Code eps payment confirmation message	Merchant confirmation upon receipt of eps payment confirmation message	URL redirect to merchant
OK	1:1 (no error), HTTP-Status Code= 200	Transaction Ok Url
OK	Merchant not reachable (Confirmation-Response with HTTP-Status Code >= 400, independent of HTTP-Message-Body)	Transaction No- k Url?epserrorcode=ERROR1
OK	HTTP-StatusCode of Response <> 200 or XML incorrect or ShopResponseDetails by merchant with ErrorMessage, e.g. XML incorrect, broken signature	Transaction No- k Url?epserrorcode=ERROR2
NOK	merchant returns the received payment confirmation unchanged via SO to the debtor bank and confirms correct interpretation of payment confirmation	Transaction Nok Url Buyer cancellation: Transaction No- k Url?epserrorcode=ERROR3
NOK	Merchant not reachable (Vitality Check or Confirmation Response with HTTP-Status-Code >=400, independent of HTTP-Message-Body)	Transaction No- k Url?epserrorcode=ERROR1 Buyer cancellation: Transaction No- k Url?epserrorcode=ERROR3
NOK	HTTP-Status-Code of Response <> 200 or XML incorrect or ShopResponseDetails by merchant with ErrorMessage, e.g. XML incorrect, broken signature	Transaction No- k Url?epserrorcode=ERROR2 Buyer cancellation: Transaction No- k Url?epserrorcode=ERROR3

7.2 eps workflow with central debtor bank selection

In the following steps "II-1 to II-7" the eps workflow using the dynamic central debtor bank selection service offered by the Scheme Operator is described.

The steps „III-1 bis III-10" are already described in section 7.1 following.

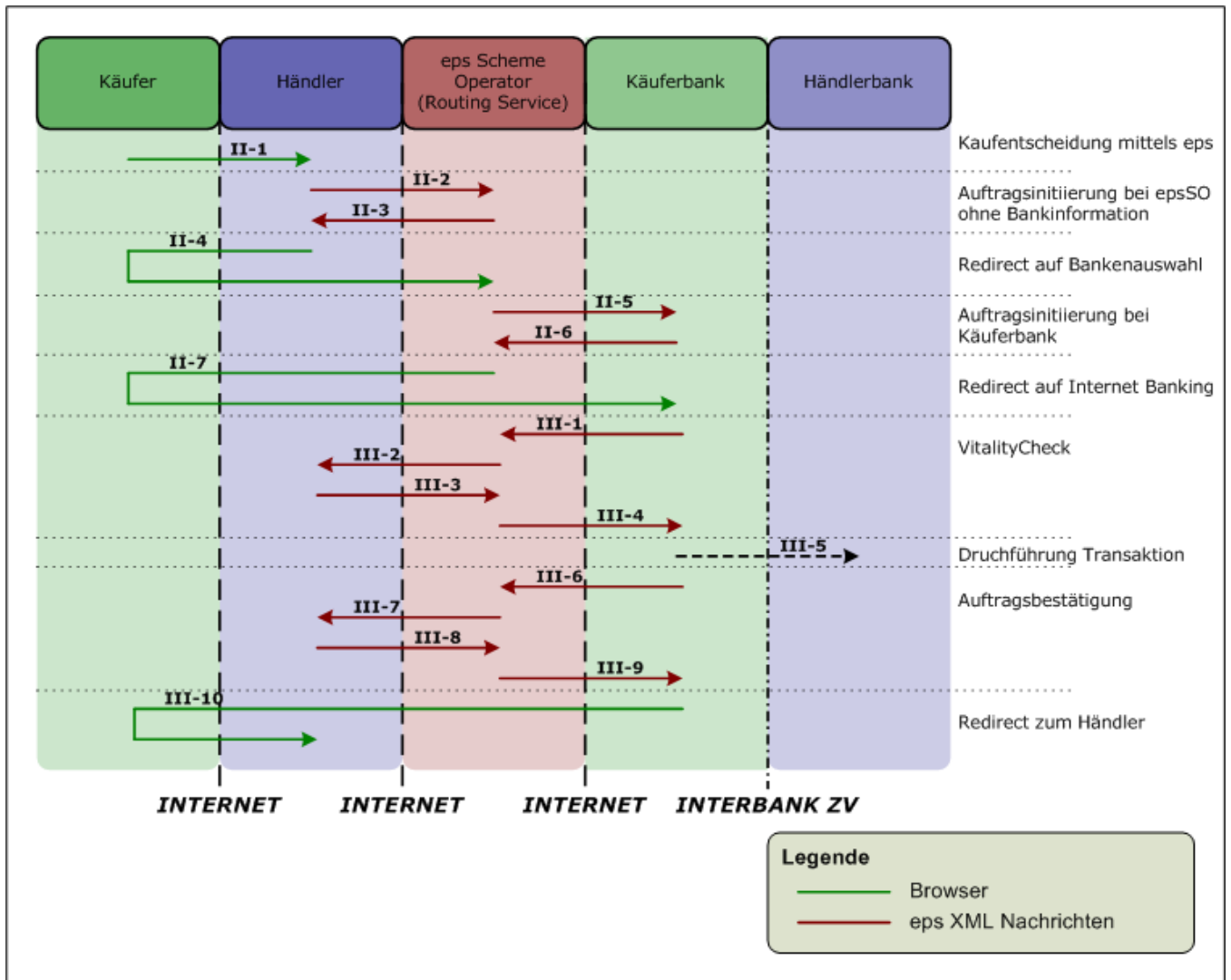


figure 7-2: eps workflow description with central debtor bank selection

Central debtor bank selection at SO

The SO offers a dynamic central debtor bank selection process which provides all active eps debtor banks as listed and registered in the SO bank database.

This service can be implemented by the merchant without having to maintain the eps debtor banks (the bank list) by himself since this is the responsibility of the SO only.

When using this service the merchant forwards the eps payment initiation message to the central SO RoutingURL https://routing.eps.or.at/appl/epsSO/transinit/eps/v2_5 without any routing information (= e.g. BIC) of the debtor bank.

After successful check and validation of the incoming eps payment initiation message the SO responds with an "epsp:BankResponseDetails" message to the merchant. In this message the „ClientRedirectURL“ is pointing to the dynamic central debtor bank selection process at the SO.

The merchant has to redirect the buyer to the URL provided in the „ClientRedirectURL“, e.g.

`https://routing.eps.or.at/appl/epsSO/transinit/bankauswahl_prepare.html?lang=de&caISO=%2BaDRiYhLjZXKuB19*CkCTIMQBN6sYSHmjNPQkIglglcYeFS98ZCVrvzVdGw5tF1Fzi0JrGhL*WWFcSHu6PWY2FCY2BTH0umA-`

The buyer can finally select the debtor bank (note: at the SO selection service!), after successful selection the buyer will be redirected by the SO to the debtor bank.

The central debtor bank selection service/webpage has a neutral layout (note: shown in german):




ONLINE ÜBERWEISUNG
Bitte wählen Sie Ihre Bank aus.

Bankname/BIC/BLZ:

Vorgang abbrechen Weiter zum Online-Banking

figure 7-3: Layout central debtor bank selection

It can not be called in a window with a dedicated name, nor can it be integrated in a Frame.

The central debtor bank selection service/webpage can run without active JavaScript.

Additional functionalities:

- Flexible size to the window size

- filtering of list during text input
- navigation with arrow keys and confirmation with Enter-key or double-click in the selection list
- Preselection of the last selected banking group with repeated use of bank selection (due to cookie - with a click or Enter to get on to online-banking)
- Prevention of double-click events on the buttons (JavaScript - prevent multiple initiation of the same transaction at the bank)

The redirect to the online-banking of the debtor bank is handled with redirect within the window.

In the circumstance that the central debtor bank selection service (the SO) has to redirect the buyer to the TransactionNOK-URL, the buyer will be informed immediately in the window of the central debtor bank selection service with an ERROR message.

By clicking on the "zurück zum Händler {X}"-Button (back to merchant button), a POST will be initiated back to the targetWindow specified by the merchant (if not available, a new window will be opened).

7.2.1 Step II-1: selection of payment method

Having agreed on terms of purchase the buyer selects "eps Online-Überweisung" as online payment method.

7.2.2 Step II-2: eps payment initiation merchant-SO

The merchant creates the eps payment initiation message (step 2a „epsp:TransferInitiatorDetails“ according to the eps XML Schema) and sends the eps XML message to the central SO URL, without any routing information (= e.g. BIC) of the debtor bank.

7.2.3 Step II-3: Bank response SO-merchant

The SO checks and validates the incoming eps payment initiation message.

The following validation will be carried out by the eps SO:

- Content-Type
- XML syntactic validity of the eps message
- merchant authorisation
- check whether the credit account in the central merchant database corresponds to the account in the eps payment initiation
- check whether the merchant is allowed (as specified in the merchant database) to use the option for future date in the eps payment initiation
- check of valid expiration time (max. 60 minutes)

In case of a positive validation an "eps:BankResponseDetails" message (step 2b according to eps XML Schema) will be forwarded from the SO to the merchant, including the „eps:ClientRedirectUrl" as redirect URL to the central debtor bank selection service of the SO.

In case of a negative validation a XML response message will be sent to the merchant with the particular error code/error message.

7.2.4 Step II-4: Buyer Redirect to SO debtor bank selection service

The merchant receives the "eps:BankResponseDetails" message from the SO and if everything is OK redirects the buyer to the central debtor bank selection (as specified URL specified in the eps:BankResponseDetails' message)

After redirect the buyer then selects the desired debtor bank in the central debtor bank selection.

7.2.5 Step II-5: eps payment initiation SO-debtor bank

In the eps process the SO always acts as merchant between the SO and the debtor bank; merchant authorization is exclusively via electronic signature together with SO UserID "epsSchemeOperator".

The SO replaces the merchant ID with the SO UserID "epsSchemeOperator" as well as the confirmationURL, additionally the SO sets in the data element "atrul:Realization" the identifier "EPN" or "EPG". In the data element "atrul:PaymentDescription" the SO forwards the merchant name or alias (if available).

The SO forwards the eps payment initiation message („eps:TransferInitiatorDetails" according to the eps XML Schema V25) to the selected debtor bank.

The outgoing message will be digitally signed by the SO.

If the "eps:TransferInitiatorDetails" message is not answered by the debtor bank within the timeout or in case of a technical error, the buyer is not presented with the debtor bank selection, but redirected to the Transaction NOK URL together with error code ERROR3.

ERROR 3 indicates that the debtor bank is currently not available or that an error occurred at the debtor bank in the course of processing the eps payment initiation.

The outgoing message is stored at the SO and are available through the SO Admin Tool.

7.2.6 Step II-6: bank response debtor bank-SO

The debtor bank checks the authorisation of the SO, processes the initiation message and responds to the eps payment initiation message with an "eps:BankResponseDetails" message.

If no errors occur (ErrorCode "000") the debtor bank sends the "epsp:BankResponseDetails" message to the SO, including the „epsp:ClientRedirectUrl" as redirect URL to the online-banking application of the debtor bank.

The incoming message is stored at the SO, available through the SO Admin Tool.

7.2.7 Step II-7: buyer redirect to the debtor bank

The SO redirects the buyer to the selected online-banking of the debtor bank.

In case of an error occurred at the debtor bank in the course of processing the eps payment initiation the buyer will be redirected to the Transaction NOK URL together with error code ERROR3.

7.3 Usage of the bank-selection-widget

The bank-selection-widget supports eps-merchants in implementing a bank-selection for customers direct in their webshop as an alternative to the central bank selection site. The bank-selection-widget allows customers to easily select their bank by searching for the banks name oder the BIC. The customers search-strings are compared to list of eps- respectively giro-pay-banks and possible hits are shown in a drop-down-list. After that the bank-name and the BIC which is necessary for the initiation of an eps-transaction are forwarded to the address mentioned in the form.

7.3.1 Search

Prerequisite for the usage of the widhet is activated JavaScript on the customers side. The best results are achieved by using the latest Versions of the respective internet-browsers (Internet Explorer, Chrome, Firefox, Safari, Opera, iOS, Android).

Some notes for the search:

- No difference is made between upper and lower case.
- Single search-strings separated by blanks are logical linked with AND. That means that every string needs to be part of the banks name or BIC.
- If the number of banks that match the search-string exceeds the limit to be shown in the drop-down-list the text „-- weitere Ergebnisse verfügbar --" will be shown at the bottom of the list. In that case the search-string needs to be more specific.
- Wildcards or free variable parameters can ´t be used.

7.3.2 Implementation

Basis for the implementation is a HTML-page with an input-field for the bank. The input-field will be enhanced by an autocomplete-function using JavaScript. The implementation is quite simple and may be applied to nearly every form.

7.3.2.1 HTML

The following example shows a complete HTML-page with a form, an input-field and a submit-button:

```
<html>
<head>
  <title>epsSO Bankauswahl Widget Example</title>
  <link type="text/css"
href="https://routing.eps.or.at/appl/epsSO/data/widget/v1/bankauswahl.css"      rel="stylesheet"
media="all" />
  <script type="text/javascript"
src="https://routing.eps.or.at/appl/epsSO/data/widget/v1/jquery-1.11.3.min.js">
  </script>
  <script type="text/javascript"
src="https://routing.eps.or.at/appl/epsSO/data/widget/v1/bankauswahl.js">
  </script>
  <script>
    $(document).ready(function() {
      $('#bankname-id').bankauswahlAutocomplete({kind: 'EG'});
    });
  </script>
</head>
<body>
  <h2>epsSO Bankauswahl Widget Example</h2>
  <form action="http://httpbin.org/post" method="post">
    <input id="bankname-id" name="bankname" type="text" autocomplete="off" />
    <input type="submit" value="Submit" />
  </form>
</body>
</html>
```

For the input-field the id-attribute needs to be stated. The name can be chosen freely and is `id="bankname-id"` in the above mentioned example. The usage of the attribute `autocomplete="off"` is recommended, to deactivate the auto-complete-function of the browser for the bank-input-field.

By using the defined id-attribute a hidden input-field is generated dynamically. Its id and name is equivalent to the freely chosen ID of the visible input field, followed by „-bic“. This name is at the same time the identifier which is used to transfer the BIC within the form.

Therefore the enhanced form within the above mentioned example looks like that:

```
<form action="http://httpbin.org/post" method="post">
  <input id="bankname-id" name="bankname" type="text" autocomplete="off" />
  <input id="bankname-id-bic" name="bankname-id-bic" type="hidden" />
  <input type="submit" value="Submit" />
</form>
```

7.3.2.2 JavaScript

To ensure the functionality of the widget the following JavaScript-file needs to be imported:

```
<script type="text/javascript"
src="https://routing.eps.or.at/appl/epsSO/data/widget/v1/bankauswahl.js"></script>
```

Additionally jQuery Version 1.10.x or 1.11.x is required. If jQuery is not available on the merchants site an import can be made using:

```
<script type="text/javascript"
src="https://routing.eps.or.at/appl/epsSO/data/widget/v1/jquery-1.11.3.min.js"></script>
```

The activation of the intelligent search-bar happens with the following JavaScript-code:

```
<script>
    $(document).ready(function() {
        $('#bankname-id').bankauswahlAutocomplete({kind: 'EG'});
    });
</script>
```

The attribute `#bankname-id` identifies the input-field from the above mentioned HTML-example and needs to be adapted for the individual implementation if applicable.

By using the parameter `kind` the available banks can be controlled. The following characters can be used in arbitrary combination.

- E ... all Austrian banks supporting eps
- G ... all German banks supporting giro pay

Unknown or not defined characters will be ignored.

Further optional parameter for the call of the function `bankauswahlAutocomplete`

delay	Defines the delay between the typed characters and the evaluation of the search-string in milliseconds. Default: 100
maxResults	Defines the maximum number of banks shown in the drop-down-list. Default: 30, Minimum: 5, Maximum: 100

The following example demonstrates the configuration of the function `bankauswahlAutocomplete` with a delay of 200ms and a maximum of 80 results.

```
<script>
  $(document).ready(function() {
    $('#bankname-id').bankauswahlAutocomplete({kind: 'EG', delay: 200, maxResults: 80});
  });
</script>
```

7.3.2.3 CSS

A default styling of the bank selection can be done via the following CSS import:

```
<link type="text/css"
href="https://routing.eps.or.at/appl/epsSO/data/widget/v1/bankauswahl.css" rel="stylesheet"
media="all" />
```

This styling may be altered individually or replaced completely.

7.3.3 Live Example

Following the link https://routing.eps.or.at/appl/epsSO-test/data/widget/v1/bankauswahl_sample.html you will find a live-demonstration of the bank-selection-widget using a neutral design and the default settings.

The used destination-adress of the form (<http://httpbin.org/post>) allows for a fast compilation of the data which is transferred and may be further used for the initiation of an eps-transaction.

8 EPS4MOBILE

To better support the mobile apps of the banks, normal eps workflow was extended with the eps4mobile workflow. The goal is to make the eps payment for the customer as easy and efficient as possible when using mobile apps. Central components for the mobile functionality are the fields TransactionID and QRCodeUrl, which are transmitted to the merchant in BankResponse. The TransactionID is used for the optional payment confirmation message request after the payment process. The QRCodeUrl can be used by the merchant to display a QR code for payment initiation.

eps4mobile can be used in the following scenarios:

1. Browser2App via QR-Code

After the redirect to the login page of the bank, the buyer can start the eps payment via QR code in addition to the well-known login in the online banking. The QR code can be imported by the buyer into the mobile banking app on the mobile device (by scanning function of the banking app) so the eps payment happens in the mobile banking app and not in online banking.

2. POS2App via QR-Code

The eps payment is not made in online banking, but by scanning the QR code in a mobile banking app.

The merchant initiates an eps payment, which is transmitted to the SO. SO provides the merchant with information that allows the merchant to create a QR code to be displayed to the buyer at a POS or local PC. The QR code can be imported by the buyer into the mobile banking app on the mobile device (by scanning feature of the mobile banking app), the eps payment is made in a mobile banking app..

3. App2App

The entire eps payment process takes place on a mobile device.

The eps payment is started and initiated from a merchant's app (e.g., ÖBB ticketing app).

The merchant initiates an eps payment, which is transmitted to the SO. SO sends the merchant information in the form of a URL with the specific scheme "epspayment" (for example, epspayment: //eps.or.at/? Transactionid = 123HOXA123).

The buyer will be redirected to the Banking app (App2App) on his mobile device by the Merchant app using the URL Scheme to make the eps payment. If the buyer has several banking apps installed on his mobile device, then he may have to manually select the banking app of his choice to conduct the eps payment on his device.

In all the scenarios mentioned, bank-specific data of the buyer are never queried or buffered by a third party, the buyer must always identify himself directly to the bank application and approve the eps online transfer by means

of known authorization procedures.

8.1 Technical specification - QR-Code

Based on the QRCodeUrl information generated by the SO, the merchant or the bank (on the login page of online banking) can generate a QR code for further processing in the purchaser's mobile banking app.

Example: The QR code contains the following link: `epspayment://eps.or.at/?transactionid=123HOXA123`

This contains the TransactionID as parameter (transactionid = 123HOXA123).

The QR code structure (size, color, etc.) is dependant on the merchant or the bank. We recommend the presentation in black, without border and without logo (see Figure 8 1: Example QR Code).



Figure 8-1: Example QR Code

QR-Code with border

Similar to the QR code for payment instructions, the bank or merchant can attach a border of the QR code with the words "eps4mobile".

In order to emphasize the function of the code and to make it safely identifiable, it has to be provided with a frame and labeled with the words "eps4mobile" (right from bottom to top).

The following dimensions rules should be applied:

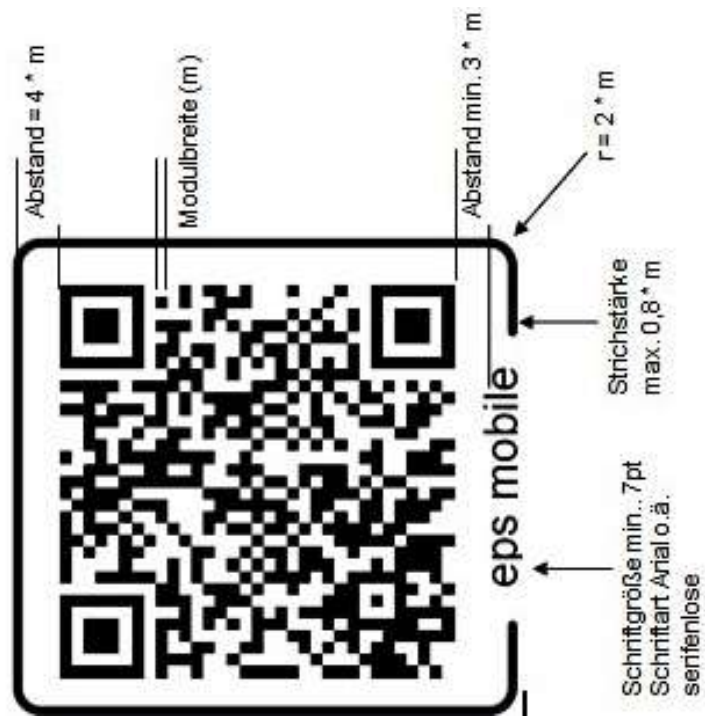


Figure 8-2: Example QR Code with border

8.2 eps4mobile Browser2App via QR-Code

The following workflow describes the eps4mobile Browser2App scenario via QR code. Only the steps that deviate from the previously described eps workflow are described.

Necessary requirement is that the merchant uses the central bank selection at SO.

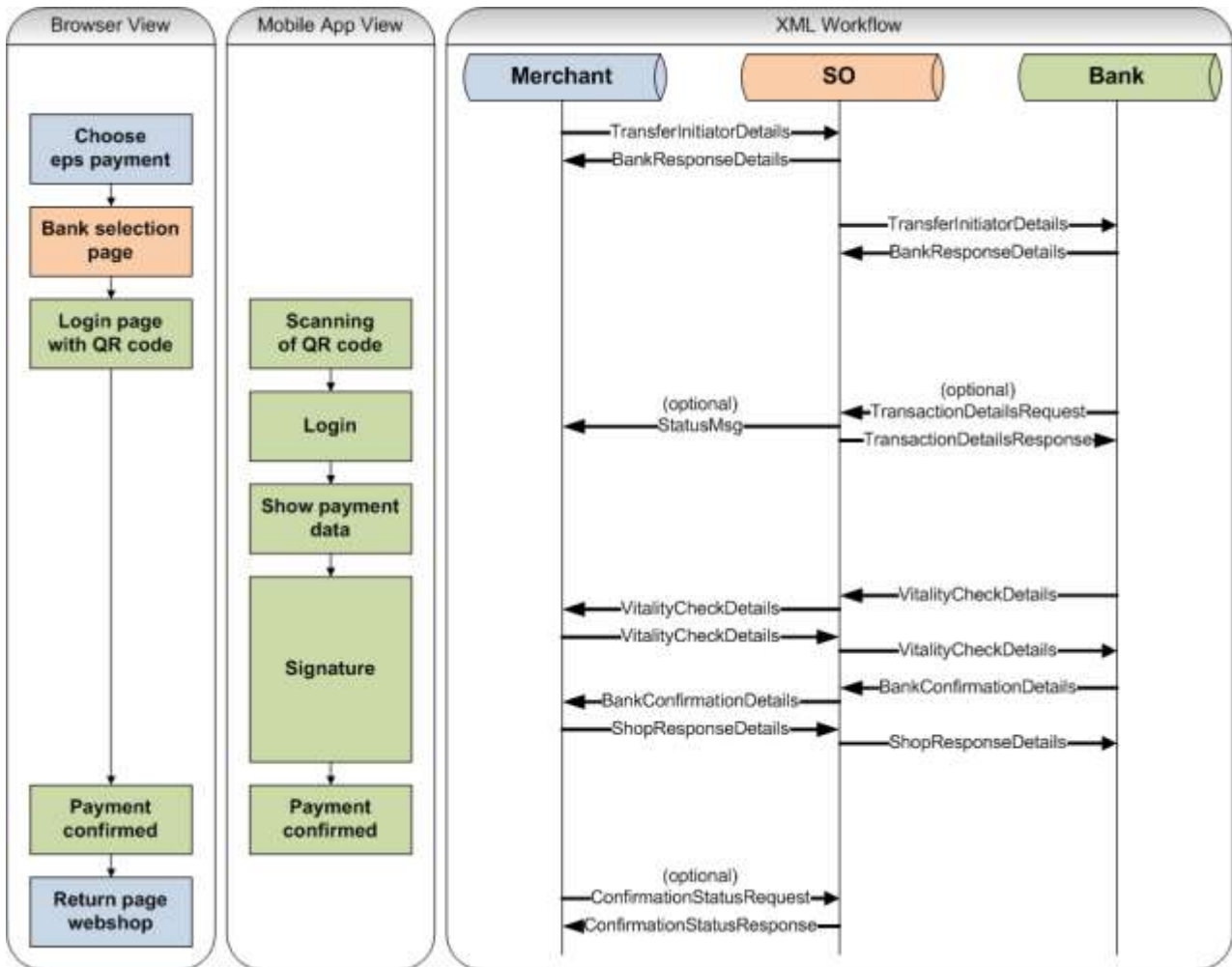


Figure 8-3: Browser2App workflow

SO - eps payment initiation to byer bank

The data elements "eps: TransactionID" and "eps: QRCodeUrl" generated by the SO are transmitted by the SO to the buyer bank.

Based on the eps: QRCodeUrl information generated by the SO, the buyer bank can generate a QR code for the buyer login into a mobile banking app.

Scanning QR-Code on online banking login site

On the login page of his bank, the buyer can choose between the known login for online banking (eg ID and PIN), or he can choose to transfer login procedure to a mobile device via QR code.

In the "Browser2App" scenario, the buyer scans the QR code with the mobile banking app.

⇒ Recommendation: after the successful login, the customer should also be informed about the status of the payment process on the desktop, e.g. Payment process has been started on a mobile device.

Bank – Init Request to SO

Optionally, the buyer bank may request the eps4mobile transaction data from the SO via an epsp: TransactionDetailsRequest.

SO – Answer to the Init Request from bank

The SO answers the epsp: TransactionDetailsRequest with an epsp: TransactionDetailsResponse message.

SO – PaymentInfo to merchant

The SO sends the status of the eps transaction to the merchant in the epsp: StatusMsg message after the bank has requested the eps transaction data from the SO.

In order for the merchant to receive this message from the SO, he must send the data element atrul: StatusMsgEnabled with the value true in the eps payment initiation.

Buyer - Transfer

The buyer bank must ensure that the eps payment transaction data is displayed to the buyer (for payment) after the successful QR code scan in the mobile banking app.

The payment is confirmed using the usual authorization methods in the buyer banking app.

Status information to the buyer

On the mobile device (for example mobile phone) the buyer will see the status of the eps transfer in the Banking App (OK / NOK); the eps payment is completed.

The status of the eps transfer is also displayed on the desktop ⇒ e.g. on the online banking page, e.g. Details on payment incl. Status (OK / NOK), Button "Back to Shop"

8.3 eps4mobile POS2App via QR-Code

The Figure 8-4 describes the eps4mobile POS2App via QR code workflow.

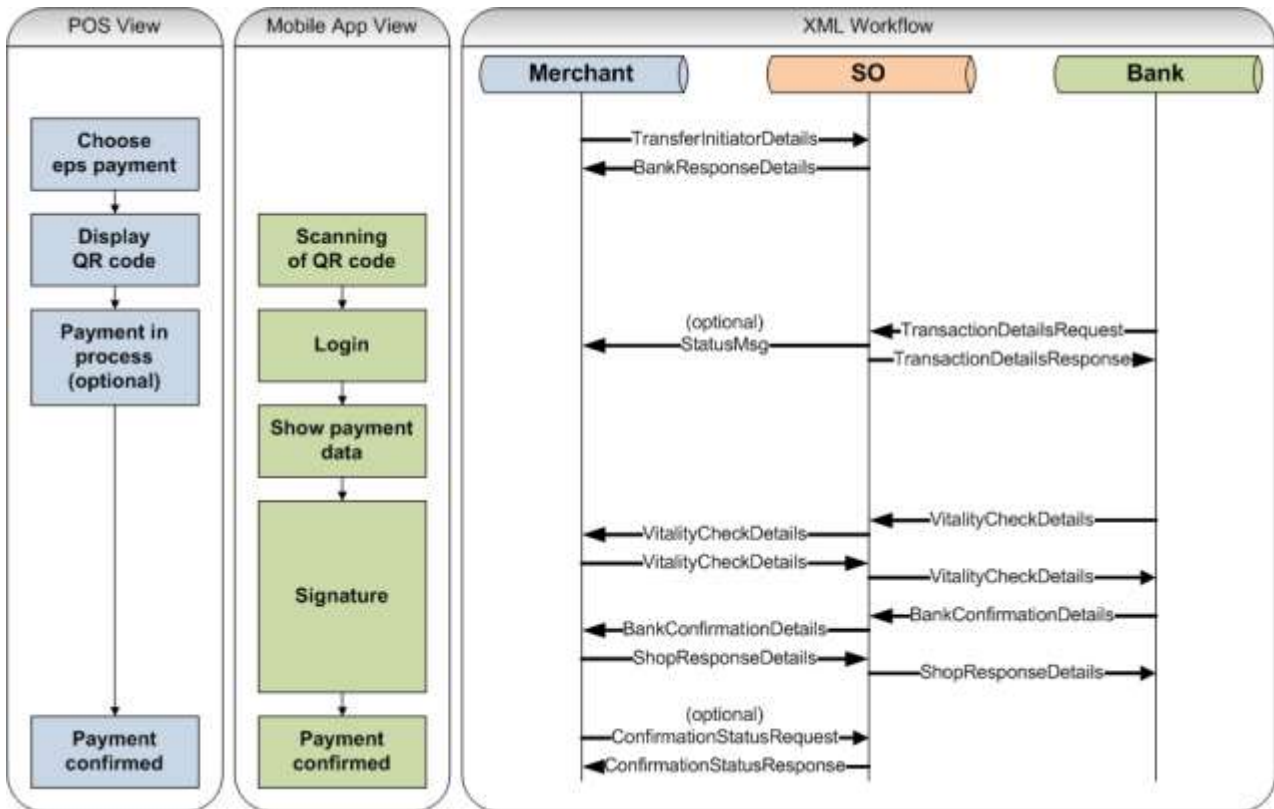


Figure 8-4: POS2App workflow

Merchant – QR code display

The merchant displays the QR code to the buyer.

Buyer - Scanning the QR code

In the "POS2App" scenario the buyer scans the displayed QR code with his online banking app.

The bank parses the TransactionID parameter from the QR code and uses the TransactionID to retrieve the Transaction Data.

OPTIONAL: 'Bank - Init Request to SO' and 'SO - Response to Init Request to Bank' can already be carried out at this point (after the bank has received the TransactionID and the payment data is displayed even before the buyers login in the online banking app actually happens).

Bank – Init Request to SO

The buyer bank has to request the eps4mobile transaction data from the SO via an epsp: TransactionDetailsRequest.

SO – Answer to the Init Request from bank

The SO answers the epsp: TransactionDetailsRequest with an epsp: TransactionDetailsResponse message.

SO – PaymentInfo to merchant

The SO sends the status of the eps transaction to the merchant in the epsp: StatusMsg message after the bank has requested the eps transaction data from the SO.

In order for the merchant to receive this message from the SO, he must send the data element atrul: StatusMsgEnabled with the value true in the eps payment initiation.

Buyer - Transfer

The buyer bank must ensure that the eps payment transaction data is displayed to the buyer (for payment) after the successful QR code scan in the mobile banking app.

The payment is confirmed using the usual authorization methods in the buyer banking app.

Status information to the buyer

On the mobile device (for example mobile phone) the buyer will see the status of the eps transfer in the Banking App (OK / NOK); the eps payment is completed.

Payment successful

Banking App: Based on the received confirmation message (step 18), the buyer must be notified of the payment status in the Banking App.

Merchant (POS): Based on the received confirmation message (step 18), the buyer must be notified by the merchant about the status of the payment transaction.

Payment not successful

Banking App: Based on the received confirmation message (step 18), the buyer must be notified of the payment status in the Banking App.

Merchant (POS): Based on the received confirmation message (step 18), the buyer must be notified by the merchant about the status of the payment transaction.

8.4 eps4mobile App2App

The Figure 8-5 describes the eps4mobile App2App workflow.

The whole process (buying in the merchant shop, payment in the banking app) is conducted on the same mobile device.

Any special authentication requirements to the bank are not shown here!

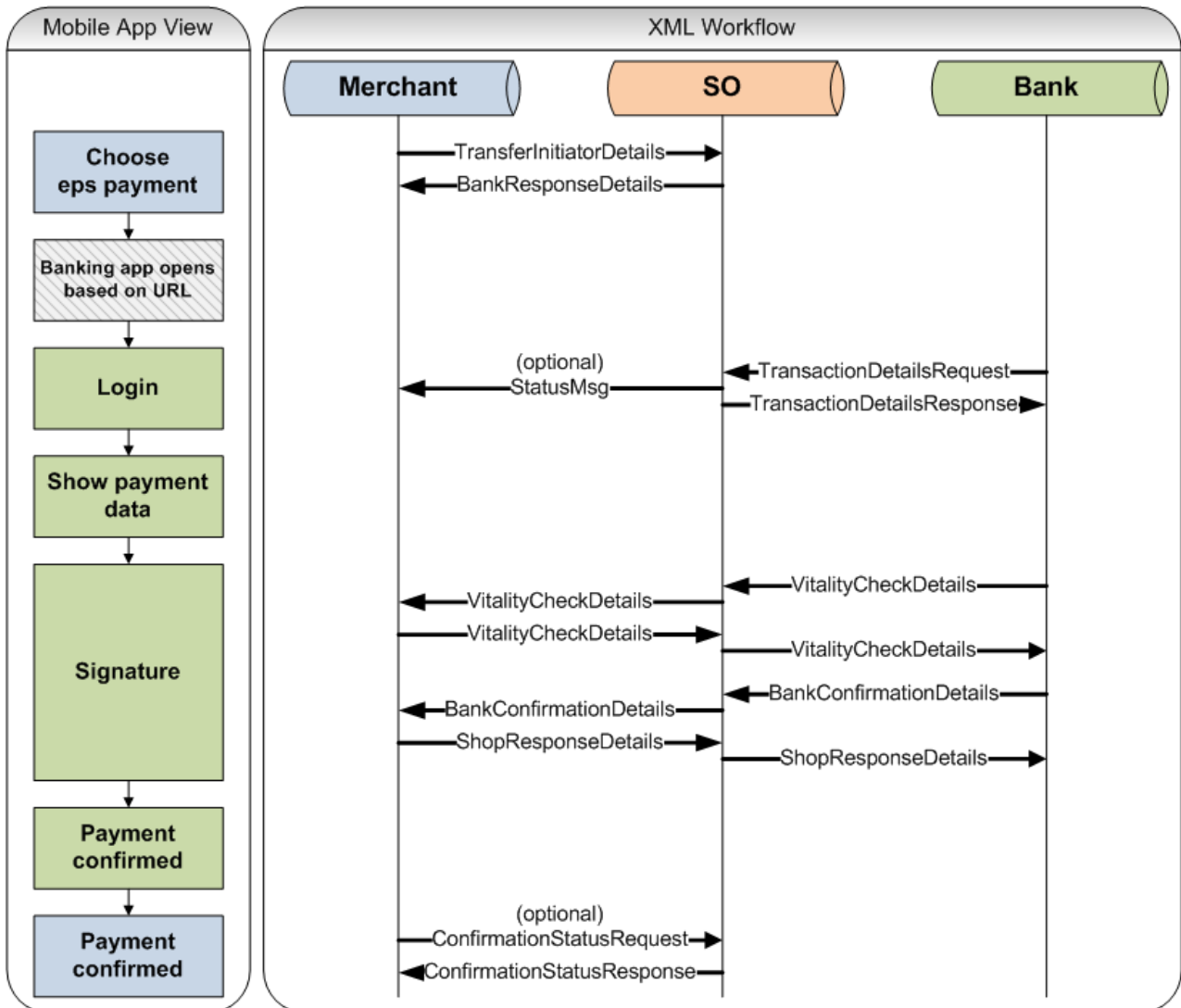


Figure 8-5: App2App workflow

Buyer - eps Online-Überweisung payment selection

The Buyer wishes to pay with "eps Online-Überweisung" and chooses the payment method within the merchant app.

Merchant - eps payment initiation to SO

The merchant creates the eps XML payment initiation message

(`eps: TransferInitiatorDetails`) according to the current eps XML standard scheme and transmits it to the URL of the SO.

The eps payment order message **does not contain any information about the buyer bank in this scenario!**

Forward App2App

The link transmitted in the element `QRCodeUrl` is called, and the buyer is forwarded from the merchant app to the banking app (App2App) in order to be able to process the eps payment directly on his mobile device.

Calling the mobile banking app is accomplished by the mobile operating system and will only work if the banking app on the mobile device is registered for the user-defined URI schema "**epspayment**".

If the buyer has several bank accounts and thus banking apps registered on "**epspayment**", the result is operating system dependent.

Current assumption (as of Q1-2015):

- Android and Windows Phone: The user receives a selection of the registered apps.
- iOS: The operating system selects (not traceable) one of the registered apps.

Buyer – Login banking app

The buyer authenticates himself to the bank.

Bank – Init Request to SO

The buyer bank has to request the eps4mobile transaction data from the SO via an `eps: TransactionDetailsRequest`.

SO – Answer to the Init Request from bank

The SO answers the `eps: TransactionDetailsRequest` with an `eps: TransactionDetailsResponse` message.

SO – PaymentInfo to merchant

The SO sends the status of the eps transaction to the merchant in the `eps: StatusMsg` message after the bank has requested the eps transaction data from the SO.

```

<?xml version="1.0" encoding="UTF-8"?>
<epsp:EpsProtocolDetails SessionLanguage="DE"
xmlns:atrul="http://www.stuzza.at/namespaces/eps/austrianrules/2014/10"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
xmlns:epi="http://www.stuzza.at/namespaces/eps/epi/2013/02"
xmlns:eps="http://www.stuzza.at/namespaces/eps/payment/2014/10"
xmlns:epsp="http://www.stuzza.at/namespaces/eps/protocol/2014/10"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.stuzza.at/namespaces/eps/protocol/2014/10 EPSProtocol-V26.xsd ">
  <epsp:StatusMsg>
    <epsp:TransactionId>eps1234567890</epsp:TransactionId>
    <epsp>Status>PAYMENT_IN_PROCESS</epsp>Status>
  </epsp:StatusMsg>
</epsp:EpsProtocolDetails>

```

In order for the merchant to receive this message from the SO, he must send the data element atrul: StatusMsgEnabled with the value true in the eps payment initiation.

Field specification in AustrianRulesDetails block

```

<atrul:AustrianRulesDetails>
  <atrul:PaymentDescription>Test 12 2014</atrul:PaymentDescription>
  <atrul:DigSig>SIG</atrul:DigSig>
  <atrul:ExpirationTime>2014-12-17T16:00:00</atrul:ExpirationTime>
  <atrul>StatusMsgEnabled>true</atrul>StatusMsgEnabled>
</atrul:AustrianRulesDetails>
</eps:PaymentInitiatorDetails>

```

The data element <StatusMsgEnabled> can have the values true (message will be sent) or false (message will not be sent) and is generally optional. If the field is not set by the merchant, it will be interpreted as false.

The merchant can use the information to show the payment status to the buyer.

Buyer - Transfer

The buyer bank must ensure that the eps payment transaction data is displayed to the buyer (for payment) after the successful QR code scan in the mobile banking app.

The payment is confirmed using the usual authorization methods in the buyer banking app.

9 ANNEX

9.1 MAPPINGTABLE eps – SCT pacs.008 (INTERBANK!)

The following overview demonstrates the mapping of eps data elements of the eps payment initiation with the Interbank SEPA Credit Transfer SCT pacs.008 (e.g. for further use in other applications). Please notice that not all elements are relevant for payments!

M...Mandatory, O...Optional, C...Conditional

Data elements	Description	eps	Data type	SCT pacs.008
epi:EpiDetails		M		
epi:IdentificationDetails		M		
epi:Date	The date on which the payment request has been created	M	xsd:date	
epi:ReferenceIdentifier	Reference of ePI message	M	an..35	
epi:Url	Info Beneficiary URL	O	x..512	
epi:EmailAddressIdentifier	Info Beneficiary Email	O	x..512	
epi:OrderInfoText	Additional order information	O	5*an..70	
epi:OrderingCustomerOfiIdentifier	For future scenarios, routing information (BIC of customer/buyer)	O	an 11	
epi:OrderingCustomerIdentifier	For future scenarios, routing information (e.g. IBAN of customer/buyer)	O	an..34	

Data elements		Description	eps	Data type	SCT pacs.008
epi:OrderingCustomerNameAddressText		For future scenarios, routing information (name/address of customer/buyer)	O	4*an..35	
epi:PartyDetails			M		
epi:BfiPartyDetails			M		
epi:BfiBicIdentifier		Bank Identifier	M	an 11	CdtrAgt/FinInstnId/BIC
epi:BeneficiaryPartyDetails			M		
epi:BeneficiaryNameAddressText		Unstructured name/address of the	C	4*an..35	Cdtr/Nm
epi:BeneficiaryBeiIdentifier		BEI (SWIFT code, the Business Entity Identifier)	C	an 11	
epi:BeneficiaryAccountIdentifier		IBAN: account identification of beneficiary/merchant	M	an..34	CdtrAcct/ID/IBAN
epi:PaymentInstructionDetails			M		
epi:PaymentInstructionIdentifier		string of characters that serves as an identification of the particular payment instruction provided by the beneficiary for optional use by the ordering customer	O	an..35	
epi:TransactionTypeCode		Specification in coded form of the underlying reason for the transfer of funds	O	an 3	
epi:InstructionCode		Further stipulates instruction related to the processing of the payment instruction	O	an..35	

Data elements		Description	eps	Data type	SCT pacs.008
epi:RemittanceIdentifier		Unique reference created by the beneficiary/merchant (structured)	C	an..35	RmtInf/Strd/CdtrRefInf/Ref
epi:UnstructuredRemittanceIdentifier		Unique reference created by the beneficiary/merchant (unstructured)	C	an..140	RmtInf/UStrd
epi:InstructedAmount		Amount	M	n..15	Amt/InstdAmt (with Ccy attribute)
	AmountCurrencyIdentifier	ISO 4217 currency code - attribute to InstructedAmount	M	a 3	Amt/InstdAmt@Ccy
epi:ChargeCode		Default: SHA (shared)	M	a 3	SCT: Standardwert „SLEV“
epi:DateOptionDetails			O		
	DateSpecificationCode	Attribute - DBD (debit date) - must be used by the beneficiary in eps payment request!	M	a 3	
	epi:OptionDate	the date on which the beneficiary/merchant would like to have the funds available	O	xsd:date	
	epi:OptionTime	Time indication to OptionDate	O	xsd:time	

10 TABLE OF FIGURES

FIGURE 6-1: PAYMENTINITIATORDETAILS	19
FIGURE 6-2: EPI:EPIDETAILS	19
FIGURE 6-3: EPI:IDENTIFICATIONDETAILS	20
FIGURE 6-4:EPI:PARTYDETAILS.....	23
FIGURE 6-5: EPI:PAYMENTINSTRUCTIONDETAILS	25
FIGURE 6-6: PAYMENTCONFIRMATIONDETAILS	31
FIGURE 6-7: EPI:EPIDETAILS	33
FIGURE 6-8: ATRUL:AUSTRIANRULESDetails.....	35
FIGURE 6-9: EPSP:TRANSFERINITIATORDETAILS	38
FIGURE 6-10: EPSP:TRANSFERMSGDETAILS	41
FIGURE 6-11: EPSP:BANKRESPONSEDETAILS	46
FIGURE 6-12: EPSP:VITALITYCHECKDETAILS	48
FIGURE 6-13: EPSP:BANKCONFIRMATIONDETAILS	49
FIGURE 6-14: EPSP:SHOPRESPONSEDETAILS – VALIDATED	52
FIGURE 6-15: EPSP:SHOPRESPONSEDETAILS - NON VALIDATED	53
FIGURE 6-16: EPSP: TRANSACTIONDETAILSREQUEST	54
FIGURE 6-17: EPSP: TRANSACTIONDETAILSRESPONSE.....	56
FIGURE 6-18: EPSP: STATUSMSG.....	58
FIGURE 6-19: EPSP:CONFIRMATIONSTATUSREQUEST	59
FIGURE 6-20: EPSP:CONFIRMATIONSTATUSRESPONSE.....	60
FIGURE 7-1: EPS WORKFLOW	63
FIGURE 7-2: EPS WORKFLOW DESCRIPTION WITH CENTRAL DEBTOR BANK SELECTION	74
FIGURE 7-3: LAYOUT CENTRAL DEBTOR BANK SELECTION	75
FIGURE 8-1: EXAMPLE QR CODE	83
FIGURE 8-3: BROWSER2APP WORKFLOW	85
FIGURE 8-4: POS2APP WORKFLOW.....	87
FIGURE 8-5: APP2APP WORKFLOW.....	89