

Sage Payments Solutions

# Sage Exchange Virtual Desktop

XML Message Formats and Field Definitions

## Contents

Revisions .....	<b>Error! Bookmark not defined.</b>
XMLSchema .....	6
XML Reserved Characters .....	12
Message Formats.....	13
Introduction .....	13
Sale.....	14
XML Sale Request with User Interface Sample.....	14
XML Sale Request without User Interface Sample .....	15
XML Sale Request and Vault Storage with User Interface Sample .....	16
Authorization .....	17
XML Authorization Request with User Interface Sample .....	17
XML Authorization Request without User Interface Sample.....	18
Capture .....	19
XML Capture Request with User Interface .....	19
XML Capture Request without User Interface.....	19
Force .....	20
XML Force Request with User Interface Sample .....	20
XML Force Request without User Interface Sample .....	21
Level 2 .....	22
XML Level 2 Sale Request with User Interface Sample .....	22
Level3Type Element.....	23
LineItems Element .....	24
Level3LineItemType Element.....	25
Void.....	26
XML Void Request without User Interface .....	26
Credit .....	27
XML Credit Request with User Interface .....	27
XML Credit Request without User Interface .....	27
Batch Inquiry.....	28
XML Batch Inquiry Request.....	28
Batch Close .....	29
XML Batch Close Request with Net and Count Verification .....	29

XML Batch Close Request without Net and Count Verification.....	29
Vault Operation .....	30
XML Vault Operation Request for Creating a Storage Record.....	30
XML Vault Operation Request for Updating a Storage Record .....	30
XML Vault Operation Request for Deleting a Storage Record.....	31
XML Vault Operation Request for Viewing a Storage Record .....	31
XML Vault Operation Request for Updating a Storage Record hiding the Account Number in the UI ..	32
XML Vault Operation Request for Updating a Storage Record masking the Account Number in the UI	32
Vault Account.....	33
XML Vault Account Request.....	33
Account Query .....	34
XML Account Query Request .....	34
Transaction Status Query .....	34
XML Transaction Status Query .....	34
Vault Status Query .....	35
XML Vault Status Query .....	35
Multi-Payment Processing.....	35
XML Multi-Payment Request with User Interface .....	35
Split Payment Processing.....	37
XML Split Payment Request with User Interface.....	37
User Interface Management.....	39
XML Authorization Request with User Interface Management .....	39
XML Vault Update Operation with User Interface Management.....	41
Request Field Definitions .....	42
Request_V1 Type Element.....	42
Application Type Element.....	42
MerchantType Element .....	43
Payments Element .....	43
Batch Element.....	43
PaymentType Element.....	44
TransactionBaseType Element.....	45
Customer Element: PersonType Element .....	46
ShippingRecipient Element: PersonType Element .....	46

Level2Type Element .....	47
Level3Type Element .....	47
LineItems Element .....	48
Level3LineItemType Element.....	48
VaultStorageType Element .....	49
RecurringType Element.....	50
BatchType Element .....	51
PostbackType Element.....	51
CompanyType Element .....	52
NameType Element .....	52
AddressType Element .....	53
TransactionStatusQueryType Element .....	53
RecurringStatusQueryType Element.....	55
VaultOperationType Element .....	55
VaultAccountType Element.....	55
AccountQueryType Element .....	56
UIType Element.....	56
Response Field Definitions.....	56
PaymentResponseType Element .....	56
ResponseType Element.....	57
VaultResponseType Element .....	57
RecurringResponseType Element .....	58
TransactionResponseType Element.....	58
BatchResponseType Element.....	61
TransactionSettlementStatusType Element .....	61
TransactionStatusQueryResponseType Element .....	62
RecurringStatusQueryResponseType Element .....	62
VaultAccountResponseType Element .....	63
AccountQueryResponseType Element.....	63
Field Validation .....	64
Alpha Numeric Fields* .....	64
Numeric Fields* .....	64
Gateway Error Codes .....	65



## XMLSchema

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema elementFormDefault="qualified" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="Request v1" nillable="true" type="Request v1" />
  <xs:complexType name="Request_v1">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="Application" type="ApplicationType" />
      <xs:element minOccurs="0" maxOccurs="1" name="Payments" type="ArrayOfPaymentType" />
      <xs:element minOccurs="0" maxOccurs="1" name="Batch" type="BatchType" />
      <xs:element minOccurs="0" maxOccurs="1" name="TransactionStatusQueries"
type="ArrayOfTransactionStatusQueryType" />
      <xs:element minOccurs="0" maxOccurs="1" name="RecurringStatusQueries"
type="ArrayOfRecurringStatusQueryType" />
      <xs:element minOccurs="0" maxOccurs="1" name="VaultStatusQuery" type="VaultStatusQueryType" />
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="ApplicationType">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="ApplicationID" type="xs:string" />
      <xs:element minOccurs="0" maxOccurs="1" name="LanguageID" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="ArrayOfPaymentType">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="PaymentType" nillable="true"
type="PaymentType" />
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="PaymentType">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="Merchant" type="MerchantType" />
      <xs:element minOccurs="0" maxOccurs="1" name="TransactionBase" type="TransactionBaseType" />
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="MerchantType">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="MerchantID" type="xs:string" />
      <xs:element minOccurs="0" maxOccurs="1" name="MerchantKey" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="TransactionBaseType">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="TransactionID" type="xs:string" />
      <xs:element minOccurs="0" maxOccurs="1" name="TransactionType" type="xs:string" />
      <xs:element minOccurs="0" maxOccurs="1" name="Referencel" type="xs:string" />
      <xs:element minOccurs="1" maxOccurs="1" name="Amount" type="xs:double" />
      <xs:element minOccurs="0" maxOccurs="1" name="AuthCode" type="xs:string" />
      <xs:element minOccurs="0" maxOccurs="1" name="VANReference" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="PersonType">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="Name" type="NameType" />
      <xs:element minOccurs="0" maxOccurs="1" name="Address" type="AddressType" />
      <xs:element minOccurs="0" maxOccurs="1" name="Company" type="CompanyType" />
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="NameType">
    <xs:sequence>
```

```

        <xs:element minOccurs="0" maxOccurs="1" name="FirstName" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="MI" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="LastName" type="xs:string" />
    </xs:sequence>
</xs:complexType>
<xs:complexType name="AddressType">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="AddressLine1" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="AddressLine2" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="City" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="State" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="ZipCode" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="Country" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="EmailAddress" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="Telephone" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="Fax" type="xs:string" />
    </xs:sequence>
</xs:complexType>
<xs:complexType name="CompanyType">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Name" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="Address" type="AddressType" />
    </xs:sequence>
</xs:complexType>
<xs:complexType name="Level2Type">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="CustomerNumber" type="xs:string" />
        <xs:element minOccurs="1" maxOccurs="1" name="TaxAmount" type="xs:double" />
    </xs:sequence>
</xs:complexType>
<xs:complexType name="Level3Type">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Level2" type="Level2Type" />
        <xs:element minOccurs="1" maxOccurs="1" name="ShippingAmount" type="xs:double" />
        <xs:element minOccurs="0" maxOccurs="1" name="DestinationZipCode" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="DestinationCountryCode" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="VATNumber" type="xs:string" />
        <xs:element minOccurs="1" maxOccurs="1" name="DiscountAmount" type="xs:double" />
        <xs:element minOccurs="1" maxOccurs="1" name="DutyAmount" type="xs:double" />
        <xs:element minOccurs="1" maxOccurs="1" name="NationalTaxAmount" type="xs:double" />
        <xs:element minOccurs="0" maxOccurs="1" name="VATInvoiceNumber" type="xs:string" />
        <xs:element minOccurs="1" maxOccurs="1" name="VATTaxAmount" type="xs:double" />
        <xs:element minOccurs="1" maxOccurs="1" name="VATTaxRate" type="xs:double" />
        <xs:element minOccurs="0" maxOccurs="1" name="LineItems" type="ArrayOfLevel3LineItemType" />
    </xs:sequence>
</xs:complexType>
<xs:complexType name="ArrayOfLevel3LineItemType">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="unbounded" name="Level3LineItemType" nillable="true"
type="Level3LineItemType" />
    </xs:sequence>
</xs:complexType>
<xs:complexType name="Level3LineItemType">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="CommodityCode" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="Description" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="ProductCode" type="xs:string" />
        <xs:element minOccurs="1" maxOccurs="1" name="Quantity" type="xs:int" />
        <xs:element minOccurs="0" maxOccurs="1" name="UnitOfMeasure" type="xs:string" />
        <xs:element minOccurs="1" maxOccurs="1" name="UnitCost" type="xs:double" />
        <xs:element minOccurs="1" maxOccurs="1" name="TaxAmount" type="xs:double" />
        <xs:element minOccurs="1" maxOccurs="1" name="TaxRate" type="xs:double" />
        <xs:element minOccurs="1" maxOccurs="1" name="DiscountAmount" type="xs:double" />
        <xs:element minOccurs="0" maxOccurs="1" name="AlternateTaxIdentifier" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="TaxTypeApplied" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="DiscountIndicator" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="NetGrossIndicator" type="xs:string" />
        <xs:element minOccurs="1" maxOccurs="1" name="ExtendedItemAmount" type="xs:double" />
        <xs:element minOccurs="0" maxOccurs="1" name="DebitCreditIndicator" type="xs:string" />
    </xs:sequence>
</xs:complexType>
<xs:complexType name="RecurringType">
    <xs:sequence>

```

```

        <xs:element minOccurs="1" maxOccurs="1" name="Schedule" type="ScheduleType" />
        <xs:element minOccurs="1" maxOccurs="1" name="Interval" type="xs:int" />
        <xs:element minOccurs="1" maxOccurs="1" name="DayOfMonth" type="xs:int" />
        <xs:element minOccurs="0" maxOccurs="1" name="StartDate" type="xs:string" />
        <xs:element minOccurs="1" maxOccurs="1" name="Amount" type="xs:double" />
        <xs:element minOccurs="1" maxOccurs="1" name="TimesToProcess" type="xs:int" />
        <xs:element minOccurs="1" maxOccurs="1" name="NonBusinessDay" type="NonBusinessDayType" />
    </xs:sequence>
</xs:complexType>
<xs:simpleType name="ScheduleType">
    <xs:restriction base="xs:string">
        <xs:enumeration value="MONTHLY" />
        <xs:enumeration value="DAILY" />
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="NonBusinessDayType">
    <xs:restriction base="xs:string">
        <xs:enumeration value="THATDAY" />
        <xs:enumeration value="BEFORE" />
        <xs:enumeration value="AFTER" />
    </xs:restriction>
</xs:simpleType>
<xs:complexType name="VaultStorageType">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="GUID" type="xs:string" />
        <xs:element minOccurs="1" maxOccurs="1" name="Service" type="VaultServiceType" />
    </xs:sequence>
</xs:complexType>
<xs:simpleType name="VaultServiceType">
    <xs:restriction base="xs:string">
        <xs:enumeration value="CREATE" />
        <xs:enumeration value="UPDATE" />
        <xs:enumeration value="RETRIEVE" />
        <xs:enumeration value="DELETE" />
    </xs:restriction>
</xs:simpleType>
<xs:complexType name="PostbackType">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="HttpsUrl" type="xs:string" />
    </xs:sequence>
</xs:complexType>
<xs:complexType name="BatchType">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Merchant" type="MerchantType" />
        <xs:element minOccurs="1" maxOccurs="1" name="Net" type="xs:double" />
        <xs:element minOccurs="1" maxOccurs="1" name="Count" type="xs:int" />
        <xs:element minOccurs="1" maxOccurs="1" name="BatchPayment" type="BatchPaymentType" />
    </xs:sequence>
</xs:complexType>
<xs:simpleType name="BatchPaymentType">
    <xs:restriction base="xs:string">
        <xs:enumeration value="CREDITCARD" />
        <xs:enumeration value="PURCHASECARD" />
    </xs:restriction>
</xs:simpleType>
<xs:complexType name="ArrayOfTransactionStatusQueryType">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="unbounded" name="TransactionStatusQueryType"
nillable="true" type="TransactionStatusQueryType" />
    </xs:sequence>
</xs:complexType>
<xs:complexType name="TransactionStatusQueryType">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Merchant" type="MerchantType" />
        <xs:element minOccurs="0" maxOccurs="1" name="TransactionID" type="xs:string" />
    </xs:sequence>
</xs:complexType>
<xs:complexType name="ArrayOfRecurringStatusQueryType">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="unbounded" name="RecurringStatusQueryType"
nillable="true" type="RecurringStatusQueryType" />
    </xs:sequence>
</xs:complexType>
<xs:complexType name="RecurringStatusQueryType">

```



```

<xs:sequence>
  <xs:element minOccurs="0" maxOccurs="1" name="Merchant" type="MerchantType" />
  <xs:element minOccurs="0" maxOccurs="1" name="RecurringID" type="xs:string" />
  <xs:element minOccurs="0" maxOccurs="1" name="StartDate" type="xs:string" />
  <xs:element minOccurs="0" maxOccurs="1" name="EndDate" type="xs:string" />
</xs:sequence>
</xs:complexType>
<xs:complexType name="VaultStatusQueryType">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Merchant" type="MerchantType" />
    <xs:element minOccurs="0" maxOccurs="1" name="VaultID" type="xs:string" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="VaultOperationType">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="VaultID" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="Merchant" type="MerchantType" />
    <xs:element minOccurs="0" maxOccurs="1" name="VaultStorage" type="VaultStorageType" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="VaultAccountType">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Company" type="CompanyType" />
    <xs:element minOccurs="0" maxOccurs="1" name="Contact" type="PersonType" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="UIType">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Display" type="UIDisplay" />
    <xs:element minOccurs="0" maxOccurs="1" name="Style" type="UIStyle" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="UIDisplay">
  <xs:sequence>
    <xs:element minOccurs="1" maxOccurs="1" name="Header" type="xs:boolean" />
    <xs:element minOccurs="1" maxOccurs="1" name="SupportLink" type="xs:boolean" />
    <xs:element minOccurs="1" maxOccurs="1" name="CheckPayment" type="xs:boolean" />
    <xs:element minOccurs="1" maxOccurs="1" name="CardPayment" type="xs:boolean" />
    <xs:element minOccurs="1" maxOccurs="1" name="SELogo" type="xs:boolean" />
    <xs:element minOccurs="1" maxOccurs="1" name="VisaLogo" type="xs:boolean" />
    <xs:element minOccurs="1" maxOccurs="1" name="AmexLogo" type="xs:boolean" />
    <xs:element minOccurs="1" maxOccurs="1" name="DiscoverLogo" type="xs:boolean" />
    <xs:element minOccurs="1" maxOccurs="1" name="MasterCardLogo" type="xs:boolean" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="UIStyle">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="MainFontColor" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="MainBackColor" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="HeaderBackColor" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="TotalsBoxBackColor" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="DividerBackColor" type="xs:string" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="AccountQueryType">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Merchant" type="MerchantType" />
  </xs:sequence>
</xs:complexType>
<xs:element name="Response v1" nillable="true" type="Response v1" />
<xs:complexType name="Response_v1">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="PaymentResponses"
type="ArrayOfPaymentResponseType" />
    <xs:element minOccurs="0" maxOccurs="1" name="BatchResponse" type="BatchResponseType" />
    <xs:element minOccurs="0" maxOccurs="1" name="TransactionStatusQueryResponses"
type="ArrayOfTransactionStatusQueryResponseType" />
    <xs:element minOccurs="0" maxOccurs="1" name="RecurringStatusQueryResponses"
type="ArrayOfRecurringStatusQueryResponseType" />
    <xs:element minOccurs="0" maxOccurs="1" name="VaultStatusQueryResponse"
type="VaultStatusQueryResponseType" />
    <xs:element minOccurs="0" maxOccurs="1" name="VaultResponse" type="VaultResponseType" />
  </xs:sequence>
</xs:complexType>

```

```

        <xs:element minOccurs="0" maxOccurs="1" name="VaultAccountResponse"
type="VaultAccountResponseType" />
        <xs:element minOccurs="0" maxOccurs="1" name="AccountQueryResponse"
type="AccountQueryResponseType" />
    </xs:sequence>
</xs:complexType>
<xs:complexType name="ArrayOfPaymentResponseType">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="unbounded" name="PaymentResponseType" nillable="true"
type="PaymentResponseType" />
    </xs:sequence>
</xs:complexType>
<xs:complexType name="PaymentResponseType">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Response" type="ResponseType" />
        <xs:element minOccurs="0" maxOccurs="1" name="VaultResponse" type="VaultResponseType" />
        <xs:element minOccurs="0" maxOccurs="1" name="RecurringResponse"
type="RecurringResponseType" />
        <xs:element minOccurs="0" maxOccurs="1" name="TransactionResponse"
type="TransactionResponseType" />
        <xs:element minOccurs="0" maxOccurs="1" name="Customer" type="PersonType" />
    </xs:sequence>
</xs:complexType>
<xs:complexType name="ResponseType">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="ResponseIndicator" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="ResponseCode" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="ResponseMessage" type="xs:string" />
    </xs:sequence>
</xs:complexType>
<xs:complexType name="VaultResponseType">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Response" type="ResponseType" />
        <xs:element minOccurs="0" maxOccurs="1" name="GUID" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="ExpirationDate" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="Last4" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="PaymentDescription" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="PaymentTypeID" type="xs:string" />
    </xs:sequence>
</xs:complexType>
<xs:complexType name="RecurringResponseType">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="RecurringID" type="xs:string" />
    </xs:sequence>
</xs:complexType>
<xs:complexType name="TransactionResponseType">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="AuthCode" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="AVSResult" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="CVVResult" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="VANReference" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="TransactionID" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="Last4" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="PaymentDescription" type="xs:string" />
        <xs:element minOccurs="1" maxOccurs="1" name="Amount" type="xs:double" />
        <xs:element minOccurs="0" maxOccurs="1" name="PaymentTypeID" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="Reference1" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="TransactionDate" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="AuxiliaryData" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="EntryMode" type="xs:string" />
        <xs:element minOccurs="1" maxOccurs="1" name="TaxAmount" type="xs:double" />
        <xs:element minOccurs="1" maxOccurs="1" name="ShippingAmount" type="xs:double" />
    </xs:sequence>
</xs:complexType>
<xs:complexType name="BatchResponseType">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Response" type="ResponseType" />
        <xs:element minOccurs="0" maxOccurs="1" name="BatchNumber" type="xs:string" />
        <xs:element minOccurs="0" maxOccurs="1" name="BatchReference" type="xs:string" />
        <xs:element minOccurs="1" maxOccurs="1" name="Net" type="xs:double" />
        <xs:element minOccurs="1" maxOccurs="1" name="Count" type="xs:int" />
        <xs:element minOccurs="1" maxOccurs="1" name="BatchPayment" type="BatchPaymentType" />
    </xs:sequence>
</xs:complexType>

```

```

<xs:complexType name="ArrayOfTransactionStatusQueryResponseType">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="TransactionStatusQueryResponseType"
nillable="true" type="TransactionStatusQueryResponseType" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="TransactionStatusQueryResponseType">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Response" type="ResponseType" />
    <xs:element minOccurs="0" maxOccurs="1" name="VaultResponse" type="VaultResponseType" />
    <xs:element minOccurs="0" maxOccurs="1" name="RecurringResponse"
type="RecurringResponseType" />
    <xs:element minOccurs="0" maxOccurs="1" name="TransactionResponse"
type="TransactionResponseType" />
    <xs:element minOccurs="0" maxOccurs="1" name="TransactionSettlementStatus"
type="TransactionSettlementStatusType" />
    <xs:element minOccurs="0" maxOccurs="1" name="Customer" type="PersonType" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="TransactionSettlementStatusType">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="TransactionType" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="SettlementType" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="SettlementDate" type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="BatchReference" type="xs:string" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="ArrayOfRecurringStatusQueryResponseType">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="RecurringStatusQueryResponseType"
nillable="true" type="RecurringStatusQueryResponseType" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="RecurringStatusQueryResponseType">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Responses" type="ArrayOfResponseType" />
    <xs:element minOccurs="0" maxOccurs="1" name="TransactionResponses"
type="ArrayOfTransactionResponseType" />
    <xs:element minOccurs="0" maxOccurs="1" name="TransactionSettlementStatuses"
type="ArrayOfTransactionSettlementStatusType" />
    <xs:element minOccurs="0" maxOccurs="1" name="Customers" type="ArrayOfPersonType" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="ArrayOfResponseType">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="ResponseType" nillable="true"
type="ResponseType" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="ArrayOfTransactionResponseType">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="TransactionResponseType"
nillable="true" type="TransactionResponseType" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="ArrayOfTransactionSettlementStatusType">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="TransactionSettlementStatusType"
nillable="true" type="TransactionSettlementStatusType" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="ArrayOfPersonType">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="PersonType" nillable="true"
type="PersonType" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="VaultStatusQueryResponseType">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Response" type="ResponseType" />
    <xs:element minOccurs="0" maxOccurs="1" name="VaultResponse" type="VaultResponseType" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="VaultAccountResponseType">

```

```

<xs:sequence>
  <xs:element minOccurs="0" maxOccurs="1" name="Response" type="ResponseType" />
  <xs:element minOccurs="0" maxOccurs="1" name="VaultAccount" type="VaultAccountType" />
  <xs:element minOccurs="0" maxOccurs="1" name="Merchant" type="MerchantType" />
</xs:sequence>
</xs:complexType>
<xs:complexType name="AccountQueryResponseType">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Response" type="ResponseType" />
    <xs:element minOccurs="0" maxOccurs="1" name="Merchant" type="MerchantType" />
    <xs:element minOccurs="0" maxOccurs="1" name="Services" type="ArrayOfString" />
    <xs:element minOccurs="1" maxOccurs="1" name="Active" type="xs:boolean" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="ArrayOfString">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="string" nillable="true"
type="xs:string" />
  </xs:sequence>
</xs:complexType>
</xs:schema>

```

## XML Reserved Characters

In XML, some characters are reserved for internal use and you must replace them by entity references when they are used in data submitted to the Sage Exchange. The following table shows the characters that must be replaced by their entity references.

Character	Entity Reference
>	&gt;
<	&lt;
&	&amp;
%	&#37;

## Message Formats

### Introduction

The following illustrations provide examples of how the request message components are structured. All requests are framed within a Request\_v1 root and must include an Application element. Optional XML elements can be omitted from the request message when not used. Some elements are arrays; in cases where more than one element is present in the array processing will occur in FIFO. XML reserved characters must be observed and encoded appropriately.

## Sale

Sale is a transaction type in which the transaction is authorized and captured in one step. Sale is used only to process purchases of goods or services that do not require physical shipment or "soft" goods that are delivered electronically.

### XML Sale Request with User Interface Sample

```
<?xml version="1.0" encoding="utf-16"?>
<Request v1 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Application>
    <ApplicationID>DEMO</ApplicationID>
    <LanguageID>EN</LanguageID>
  </Application>
  <Payments>
    <PaymentType>
      <Merchant>
        <MerchantID>999999999997</MerchantID>
        <MerchantKey>K3QD6YWyhFD</MerchantKey>
      </Merchant>
      <TransactionBase>
        <TransactionID>5ea9747c-12a4-46af-970f-f8a92f6d4f65</TransactionID>
        <TransactionType>11</TransactionType>
        <Referencel>INV# 886478943</Referencel>
        <Amount>1892.59</Amount>
      </TransactionBase>
      <Customer>
        <Name>
          <FirstName>Jane</FirstName>
          <MI> </MI>
          <LastName>Doe</LastName>
        </Name>
        <Address>
          <AddressLine1>67890 Road</AddressLine1>
          <AddressLine2></AddressLine2>
          <City>South Padre Island</City>
          <State>Texas</State>
          <ZipCode>78597</ZipCode>
          <Country>USA</Country>
          <EmailAddress>jane.doe@sagepayments.com</EmailAddress>
          <Telephone></Telephone>
          <Fax></Fax>
        </Address>
      </Customer>
    </PaymentType>
  </Payments>
</Request_v1>
```

## XML Sale Request without User Interface Sample

```
<?xml version="1.0" encoding="utf-16"?>
<Request v1 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Application>
    <ApplicationID>DEMO</ApplicationID>
    <LanguageID>EN</LanguageID>
  </Application>
  <Payments>
    <PaymentType>
      <Merchant>
        <MerchantID>999999999997</MerchantID>
        <MerchantKey>K3QD6YWyhFD</MerchantKey>
      </Merchant>
      <TransactionBase>
        <TransactionID>e856a127-8527-431c-807f-6efacd8bdf83</TransactionID>
        <TransactionType>01</TransactionType>
        <Reference1>INV# 451777674</Reference1>
        <Amount>2152.92</Amount>
      </TransactionBase>
      <Customer>
        <Name>
          <FirstName>John</FirstName>
          <MI> </MI>
          <LastName>Doe</LastName>
        </Name>
        <Address>
          <AddressLine1>12345 Street</AddressLine1>
          <AddressLine2></AddressLine2>
          <City>South Padre Island</City>
          <State>Texas</State>
          <ZipCode>78597</ZipCode>
          <Country>USA</Country>
          <EmailAddress>john.doe@sagepayments.com</EmailAddress>
          <Telephone></Telephone>
          <Fax></Fax>
        </Address>
      </Customer>
      <VaultStorage>
        <GUID>dd83d7559a274fb2b66e774a4febced7</GUID>
        <Service>RETRIEVE</Service>
      </VaultStorage>
    </PaymentType>
  </Payments>
</Request_v1>
```

## XML Sale Request and Vault Storage with User Interface Sample

```
<?xml version="1.0" encoding="utf-16"?>
<Request v1 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Application>
    <ApplicationID>DEMO</ApplicationID>
    <LanguageID>EN</LanguageID>
  </Application>
  <Payments>
    <PaymentType>
      <Merchant>
        <MerchantID>999999999997</MerchantID>
        <MerchantKey>K3QD6YWyhFD</MerchantKey>
      </Merchant>
      <TransactionBase>
        <TransactionID>e856a127-8527-431c-807f-6efacd8bdf83</TransactionID>
        <TransactionType>11</TransactionType>
        <Reference1>INV# 451777674</Reference1>
        <Amount>2152.92</Amount>
      </TransactionBase>
      <Customer>
        <Name>
          <FirstName>John</FirstName>
          <MI> </MI>
          <LastName>Doe</LastName>
        </Name>
        <Address>
          <AddressLine1>12345 Street</AddressLine1>
          <AddressLine2></AddressLine2>
          <City>South Padre Island</City>
          <State>Texas</State>
          <ZipCode>78597</ZipCode>
          <Country>USA</Country>
          <EmailAddress>john.doe@sagepayments.com</EmailAddress>
          <Telephone></Telephone>
          <Fax></Fax>
        </Address>
      </Customer>
      <VaultStorage>
        <Service>CREATE</Service>
      </VaultStorage>
    </PaymentType>
  </Payments>
</Request_v1>
```



## Authorization

Authorization is a transaction type in which an account is verified to be valid and has not reached its limit. The total amount of the transaction is reserved against the account balance. Authorizations are used if goods are to be physically shipped or in other cases for which the merchant must first verify whether the order can be fulfilled. An approved Authorization is followed by a Capture, which prepares it for settlement.

### XML Authorization Request with User Interface Sample

```
<?xml version="1.0" encoding="utf-16"?>
<Request_v1 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Application>
    <ApplicationID>DEMO</ApplicationID>
    <LanguageID>EN</LanguageID>
  </Application>
  <Payments>
    <PaymentType>
      <Merchant>
        <MerchantID>999999999997</MerchantID>
        <MerchantKey>K3QD6YWyhfD</MerchantKey>
      </Merchant>
      <TransactionBase>
        <TransactionID>5ea9747c-12a4-46af-970f-f8a92f6d4f65</TransactionID>
        <TransactionType>12</TransactionType>
        <Reference1>INV# 886478943</Reference1>
        <Amount>1892.59</Amount>
      </TransactionBase>
      <Customer>
        <Name>
          <FirstName>Jane</FirstName>
          <MI> </MI>
          <LastName>Doe</LastName>
        </Name>
        <Address>
          <AddressLine1>67890 Road</AddressLine1>
          <AddressLine2></AddressLine2>
          <City>South Padre Island</City>
          <State>Texas</State>
          <ZipCode>78597</ZipCode>
          <Country>USA</Country>
          <EmailAddress>jane.doe@sagepayments.com</EmailAddress>
          <Telephone></Telephone>
          <Fax></Fax>
        </Address>
      </Customer>
    </PaymentType>
  </Payments>
</Request_v1>
```

## XML Authorization Request without User Interface Sample

```
<?xml version="1.0" encoding="utf-16"?>
<Request v1 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Application>
    <ApplicationID>DEMO</ApplicationID>
    <LanguageID>EN</LanguageID>
  </Application>
  <Payments>
    <PaymentType>
      <Merchant>
        <MerchantID>999999999997</MerchantID>
        <MerchantKey>K3QD6YWyhFD</MerchantKey>
      </Merchant>
      <TransactionBase>
        <TransactionID>e856a127-8527-431c-807f-6efacd8bdf83</TransactionID>
        <TransactionType>02</TransactionType>
        <Reference1>INV# 451777674</Reference1>
        <Amount>2152.92</Amount>
      </TransactionBase>
      <Customer>
        <Name>
          <FirstName>John</FirstName>
          <MI> </MI>
          <LastName>Doe</LastName>
        </Name>
        <Address>
          <AddressLine1>12345 Street</AddressLine1>
          <AddressLine2></AddressLine2>
          <City>South Padre Island</City>
          <State>Texas</State>
          <ZipCode>78597</ZipCode>
          <Country>USA</Country>
          <EmailAddress>john.doe@sagepayments.com</EmailAddress>
          <Telephone></Telephone>
          <Fax></Fax>
        </Address>
      </Customer>
      <VaultStorage>
        <GUID>dd83d7559a274fb2b66e774a4febced7</GUID>
        <Service>RETRIEVE</Service>
      </VaultStorage>
    </PaymentType>
  </Payments>
</Request_v1>
```

## Capture

Capture is a transaction type that puts an Authorization transaction into a Captured state for settlement. In the case of partial shipments, the Capture amount may be less than the Authorization amount. Captures can be initiated only after the purchased goods have been shipped.

### XML Capture Request with User Interface

```
<?xml version="1.0" encoding="utf-16"?>
<Request v1 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Application>
    <ApplicationID>DEMO</ApplicationID>
    <LanguageID>EN</LanguageID>
  </Application>
  <Payments>
    <PaymentType>
      <Merchant>
        <MerchantID>999999999997</MerchantID>
        <MerchantKey>K3QD6YWYhFD</MerchantKey>
      </Merchant>
      <TransactionBase>
        <TransactionID>0405aa29-9be2-4c46-b8b0-b103e25a39b6</TransactionID>
        <TransactionType>13</TransactionType>
        <Amount>4577.52</Amount>
        <VANReference>CBK9A0j650</VANReference>
      </TransactionBase>
    </PaymentType>
  </Payments>
</Request_v1>
```

### XML Capture Request without User Interface

```
<?xml version="1.0" encoding="utf-16"?>
<Request v1 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Application>
    <ApplicationID>DEMO</ApplicationID>
    <LanguageID>EN</LanguageID>
  </Application>
  <Payments>
    <PaymentType>
      <Merchant>
        <MerchantID>999999999997</MerchantID>
        <MerchantKey>K3QD6YWYhFD</MerchantKey>
      </Merchant>
      <TransactionBase>
        <TransactionID>0405aa29-9be2-4c46-b8b0-b103e25a39b6</TransactionID>
        <TransactionType>03</TransactionType>
        <Amount>4577.52</Amount>
        <VANReference>CBK9A0j650</VANReference>
      </TransactionBase>
    </PaymentType>
  </Payments>
</Request_v1>
```

## Force

Force is a transaction type used to force a transaction into settlement in cases where a Sale or Authorization transaction cannot be processed and the Authorization Code is obtained from an outside source.

### XML Force Request with User Interface Sample

```
<?xml version="1.0" encoding="utf-16"?>
<Request v1 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Application>
    <ApplicationID>DEMO</ApplicationID>
    <LanguageID>EN</LanguageID>
  </Application>
  <Payments>
    <PaymentType>
      <Merchant>
        <MerchantID>9999999999997</MerchantID>
        <MerchantKey>K3QD6YWyhfD</MerchantKey>
      </Merchant>
      <TransactionBase>
        <TransactionID>5ea9747c-12a4-46af-970f-f8a92f6d4f65</TransactionID>
        <TransactionType>15</TransactionType>
        <Reference1>INV# 886478943</Reference1>
        <Amount>1892.59</Amount>
        <AuthCode>123456</AuthCode>
      </TransactionBase>
      <Customer>
        <Name>
          <FirstName>Jane</FirstName>
          <MI> </MI>
          <LastName>Doe</LastName>
        </Name>
        <Address>
          <AddressLine1>67890 Road</AddressLine1>
          <AddressLine2></AddressLine2>
          <City>South Padre Island</City>
          <State>Texas</State>
          <ZipCode>78597</ZipCode>
          <Country>USA</Country>
          <EmailAddress>jane.doe@sagepayments.com</EmailAddress>
          <Telephone></Telephone>
          <Fax></Fax>
        </Address>
      </Customer>
    </PaymentType>
  </Payments>
</Request_v1>
```

## XML Force Request without User Interface Sample

```
<?xml version="1.0" encoding="utf-16"?>
<Request v1 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Application>
    <ApplicationID>DEMO</ApplicationID>
    <LanguageID>EN</LanguageID>
  </Application>
  <Payments>
    <PaymentType>
      <Merchant>
        <MerchantID>999999999997</MerchantID>
        <MerchantKey>K3QD6YWyhFD</MerchantKey>
      </Merchant>
      <TransactionBase>
        <TransactionID>e856a127-8527-431c-807f-6efacd8bdf83</TransactionID>
        <TransactionType>05</TransactionType>
        <Reference1>INV# 451777674</Reference1>
        <Amount>2152.92</Amount>
        <AuthCode>123456</AuthCode>
      </TransactionBase>
      <Customer>
        <Name>
          <FirstName>John</FirstName>
          <MI> </MI>
          <LastName>Doe</LastName>
        </Name>
        <Address>
          <AddressLine1>12345 Street</AddressLine1>
          <AddressLine2></AddressLine2>
          <City>South Padre Island</City>
          <State>Texas</State>
          <ZipCode>78597</ZipCode>
          <Country>USA</Country>
          <EmailAddress>john.doe@sagepayments.com</EmailAddress>
          <Telephone></Telephone>
          <Fax></Fax>
        </Address>
      </Customer>
      <VaultStorage>
        <GUID>dd83d7559a274fb2b66e774a4febced7</GUID>
        <Service>RETRIEVE</Service>
      </VaultStorage>
    </PaymentType>
  </Payments>
</Request_v1>
```

## Level 2

Level2 data is additional transaction data required for Level II commercial card qualification for credit card transactions.

### XML Level 2 Sale Request with User Interface Sample

```
<?xml version="1.0" encoding="utf-16"?>
<Request v1 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Application>
    <ApplicationID>DEMO</ApplicationID>
    <LanguageID>EN</LanguageID>
  </Application>
  <Payments>
    <PaymentType>
      <Merchant>
        <MerchantID>999999999997</MerchantID>
        <MerchantKey>K3QD6YWyhFD</MerchantKey>
      </Merchant>
      <TransactionBase>
        <TransactionID>5ea9747c-12a4-46af-970f-f8a92f6d4f65</TransactionID>
        <TransactionType>11</TransactionType>
        <Reference1>INV# 886478943</Reference1>
        <Amount>1892.59</Amount>
      </TransactionBase>
      <Customer>
        <Name>
          <FirstName>Jane</FirstName>
          <MI> </MI>
          <LastName>Doe</LastName>
        </Name>
        <Address>
          <AddressLine1>67890 Road</AddressLine1>
          <AddressLine2></AddressLine2>
          <City>South Padre Island</City>
          <State>Texas</State>
          <ZipCode>78597</ZipCode>
          <Country>USA</Country>
          <EmailAddress>jane.doe@sagepayments.com</EmailAddress>
          <Telephone></Telephone>
          <Fax></Fax>
        </Address>
      </Customer>
      <Level2>
        <CustomerNumber>123456789012</CustomerNumber>
        <TaxAmount>92.59</TaxAmount>
      </Level2>
    </PaymentType>
  </Payments>
</Request_v1>
```

## Level3Type Element

Optional

```
<Level3>
  <Level2></Level2>
  <ShippingAmount></ShippingAmount>
  <DestinationZipCode></DestinationZipCode>
  <DestinationCountryCode></DestinationCountryCode>
  <VATNumber></VATNumber>
  <DiscountAmount></DiscountAmount>
  <DutyAmount></DutyAmount>
  <NationalTaxAmount></NationalTaxAmount>
  <VATInvoiceNumber></VATInvoiceNumber>
  <VATTaxAmount></VATTaxAmount>
  <VATTaxRate></VATTaxRate>
  <LineItems></LineItems>
</Level3>
```

The Level3 element contains the optional CustomerNumber, TaxAmount, DestinationZipCode, NationalTax, VATNumber, DiscountAmount, DutyAmount, VATInvoiceNumber, VATTaxAmount, VATTaxRate, DestinationCountryCode, and LineItems elements used to qualify a Purchase Card payment for Level III.

Element Name	Data Type	Length	Required	Comments
<b>Level2</b>	Level2Type		Yes	Contains the Level2 elements CustomerNumber and TaxAmount
<b>ShippingAmount</b>	decimal	9	Yes	Shipping amount charged to the transaction
<b>DestinationZipCode</b>	string	9	Yes	Postal zip code where the goods/services are shipped
<b>VATNumber</b>	string	13	Yes	Customer's Value Added Tax Number
<b>DiscountAmount</b>	decimal	9	Yes	Discount amount applied to the transaction
<b>DutyAmount</b>	decimal	9	Yes	Duty amount
<b>VATInvoiceNumber</b>	string	15	Yes	Value Added Tax invoice number
<b>VATTaxAmount</b>	decimal	9	Yes	Value Added Tax amount
<b>VATTaxRate</b>	decimal	9	Yes	Value Added Tax rate
<b>DestinationCountryCode</b>	integer	3	Yes	ISO Country Code where the goods/services are shipped
<b>LineItems</b>	Level3LineItemType		No	Contains zero or more Level3LineItemType elements
<b>NationalTaxAmount</b>	string	9	Yes	National Tax amount applied to the transaction

## LineItems Element

*Optional*

```
<LineItems>  
  < Level3LineItemType></ Level3LineItemType>  
</LineItems>
```

The LineItems element can contain zero or more Level3LineItemType elements. When more than one Level III line item is required to be processed by the calling application multiple Level3LineItemType elements are used.



## Level3LineItemType Element

Optional

```
<Level3LineItemType>
  <CommodityCode></CommodityCode>
  <Description></Description>
  <ProductCode></ProductCode>
  <Quantity></Quantity>
  <UnitOfMeasure></UnitOfMeasure>
  <UnitCost></UnitCost>
  <TaxAmount></TaxAmount>
  <TaxRate></TaxRate>
  <DiscountAmount></DiscountAmount>
  <AlternateTaxIdentifier></AlternateTaxIdentifier>
  <TaxTypeApplied></TaxTypeApplied>
  <DiscountIndicator></DiscountIndicator>
  <NetGrossIndicator></NetGrossIndicator>
  <ExtendedItemAmount></ExtendedItemAmount>
  <DebitCreditIndicator></DebitCreditIndicator>
</Level3LineItemType>
```

The Level3LineItemType element contains the CommodityCode, Description, ProductCode, Quantity, UnitOfMeasure, UnitCost, TaxAmount, TaxRate, DiscountAmount, AlternateTaxIdentifier, TaxTypeApplied, DiscountIndicator, NetGrossIndicator, ExtendedItemAmount, DebitCreditIndicator, and TotalAmount elements.

Element Name	Data Type	Length	Required	Comments
<b>CommodityCode</b>	string	12	Yes	Commodity code that applies to the item
<b>Description</b>	string	35	Yes	A brief description of the item
<b>ProductCode</b>	string	12	Yes	Product code that applies to the item
<b>Quantity</b>	integer	12	Yes	Quantity of item(s) purchased
<b>UnitOfMeasure</b>	string	12	Yes	Units of measure of the item(s) purchased
<b>UnitCost</b>	decimal	9	Yes	Cost of the item purchased
<b>TaxAmount</b>	decimal	9	Yes	The tax amount for the item
<b>TaxRate</b>	decimal	5	Yes	The tax rate for the item
<b>DiscountAmount</b>	decimal	9	Yes	Discount amount applied to item
<b>AlternateTaxIdentifier</b>		15	Yes	
<b>TaxTypeApplied</b>		4	Yes	
<b>DiscountIndicator</b>	string	1	Yes	
<b>NetGrossIndicator</b>	string	1	Yes	
<b>ExtendedItemAmount</b>	decimal	9	Yes	The total amount of the individual item. = ( ItemCost X Quantity ) – ( DiscountAmount x Quantity )
<b>DebitCreditIndicator</b>	string	1	Yes	

## Void

Void is a transaction type that cancels a transaction that has not yet been settled.

### XML Void Request without User Interface

```
<?xml version="1.0" encoding="utf-16"?>
<Request v1 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Application>
    <ApplicationID>DEMO</ApplicationID>
    <LanguageID>EN</LanguageID>
  </Application>
  <Payments>
    <PaymentType>
      <Merchant>
        <MerchantID>999999999997</MerchantID>
        <MerchantKey>K3QD6YWYhfD</MerchantKey>
      </Merchant>
      <TransactionBase>
        <TransactionID>782ad8d0-0dd2-4763-8d64-cc9fddfad441</TransactionID>
        <TransactionType>04</TransactionType>
        <VANReference>ABL9LKQaI0</VANReference>
      </TransactionBase>
    </PaymentType>
  </Payments>
</Request_v1>
```

## Credit

Credit is a transaction type that transfers funds to the account, rather than from the account. This transaction type is typically used to refund money for a transaction that was previously settled.

### XML Credit Request with User Interface

```
<?xml version="1.0" encoding="utf-16"?>
<Request v1 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Application>
    <ApplicationID>DEMO</ApplicationID>
    <LanguageID>EN</LanguageID>
  </Application>
  <Payments>
    <PaymentType>
      <Merchant>
        <MerchantID>999999999997</MerchantID>
        <MerchantKey>K3QD6YWyhfD</MerchantKey>
      </Merchant>
      <TransactionBase>
        <TransactionID>0405aa29-9be2-4c46-b8b0-b103e25a39b6</TransactionID>
        <TransactionType>16</TransactionType>
        <Amount>4577.52</Amount>
        <VANReference>CBK9A0j650</VANReference>
      </TransactionBase>
    </PaymentType>
  </Payments>
</Request_v1>
```

### XML Credit Request without User Interface

```
<?xml version="1.0" encoding="utf-16"?>
<Request v1 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Application>
    <ApplicationID>DEMO</ApplicationID>
    <LanguageID>EN</LanguageID>
  </Application>
  <Payments>
    <PaymentType>
      <Merchant>
        <MerchantID>999999999997</MerchantID>
        <MerchantKey>K3QD6YWyhfD</MerchantKey>
      </Merchant>
      <TransactionBase>
        <TransactionID>0405aa29-9be2-4c46-b8b0-b103e25a39b6</TransactionID>
        <TransactionType>06</TransactionType>
        <Amount>4577.52</Amount>
        <VANReference>CBK9A0j650</VANReference>
      </TransactionBase>
    </PaymentType>
  </Payments>
</Request_v1>
```

## Batch Inquiry

Batch Inquiry is used to obtain the transaction net and transaction count of the current open batch awaiting settlement.

### XML Batch Inquiry Request

```
<?xml version="1.0" encoding="utf-16"?>
<Request v1 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Application>
    <ApplicationID>DEMO</ApplicationID>
    <LanguageID>EN</LanguageID>
  </Application>
  <Batch>
    <Merchant>
      <MerchantID>999999999997</MerchantID>
      <MerchantKey>K3QD6YWyhFD</MerchantKey>
    </Merchant>
    <Net>-1</Net>
    <Count>-1</Count>
    <BatchPayment>CREDITCARD</BatchPayment>
  </Batch>
</Request_v1>
```

## Batch Close

Batch Close is used to settle transactions in the current open batch awaiting settlement. Sales, Captures, Forces, and Credit transactions qualify for settlement.

### XML Batch Close Request with Net and Count Verification

```
<?xml version="1.0" encoding="utf-16"?>
<Request v1 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Application>
    <ApplicationID>DEMO</ApplicationID>
    <LanguageID>EN</LanguageID>
  </Application>
  <Batch>
    <Merchant>
      <MerchantID>999999999997</MerchantID>
      <MerchantKey>K3QD6YWyhFD</MerchantKey>
    </Merchant>
    <Net>2561.23</Net>
    <Count>5</Count>
    <BatchPayment>CREDITCARD</BatchPayment>
  </Batch>
</Request_v1>
```

### XML Batch Close Request without Net and Count Verification

```
<?xml version="1.0" encoding="utf-16"?>
<Request v1 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Application>
    <ApplicationID>DEMO</ApplicationID>
    <LanguageID>EN</LanguageID>
  </Application>
  <Batch>
    <Merchant>
      <MerchantID>999999999997</MerchantID>
      <MerchantKey>K3QD6YWyhFD</MerchantKey>
    </Merchant>
    <Net>0</Net>
    <Count>0</Count>
    <BatchPayment>CREDITCARD</BatchPayment>
  </Batch>
</Request_v1>
```

## Vault Operation

Vault Operation is used to capture sensitive card holder data and insert or update the storage in the Sage Payment Solutions Vault. No payment is processed.

### XML Vault Operation Request for Creating a Storage Record

```
<?xml version="1.0" encoding="utf-16"?>
<Request_v1 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Application>
    <ApplicationID>DEMO</ApplicationID>
    <LanguageID>EN</LanguageID>
  </Application>
  <VaultOperation>
    <Merchant>
      <MerchantID>999999999997</MerchantID>
      <MerchantKey>K3QD6YyhFD</MerchantKey>
    </Merchant>
    <VaultStorage>
      <Service>CREATE</Service>
    </VaultStorage>
    <VaultID>2341234-12431243-2341235</VaultID>
  </VaultOperation>
</Request_v1>
```

### XML Vault Operation Request for Updating a Storage Record

```
<?xml version="1.0" encoding="utf-16"?>
<Request_v1 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Application>
    <ApplicationID>DEMO</ApplicationID>
    <LanguageID>EN</LanguageID>
  </Application>
  <VaultOperation>
    <Merchant>
      <MerchantID>999999999997</MerchantID>
      <MerchantKey>K3QD6YyhFD</MerchantKey>
    </Merchant>
    <VaultStorage>
      <Service>UPDATE</Service>
      <GUID>sfdas-ee3u38d-dagdi3-efad83</GUID>
    </VaultStorage>
    <VaultID>2341234-12431243-2341235</VaultID>
  </VaultOperation>
</Request_v1>
```

## XML Vault Operation Request for Deleting a Storage Record

```
<?xml version="1.0" encoding="utf-16"?>
<Request_v1 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Application>
    <ApplicationID>DEMO</ApplicationID>
    <LanguageID>EN</LanguageID>
  </Application>
  <VaultOperation>
    <Merchant>
      <MerchantID>999999999997</MerchantID>
      <MerchantKey>K3QD6YWyhFD</MerchantKey>
    </Merchant>
    <VaultStorage>
      <Service>DELETE</Service>
      <GUID>sfdas-ee3u38d-dagdi3-efad83</GUID>
    </VaultStorage>
    <VaultID>2341234-12431243-2341235</VaultID>
  </VaultOperation>
</Request_v1>
```

## XML Vault Operation Request for Viewing a Storage Record

```
<?xml version="1.0" encoding="utf-16"?>
<Request_v1 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Application>
    <ApplicationID>DEMO</ApplicationID>
    <LanguageID>EN</LanguageID>
  </Application>
  <VaultOperation>
    <Merchant>
      <MerchantID>999999999997</MerchantID>
      <MerchantKey>K3QD6YWyhFD</MerchantKey>
    </Merchant>
    <VaultStorage>
      <Service>RETRIEVE</Service>
      <GUID>sfdas-ee3u38d-dagdi3-efad83</GUID>
    </VaultStorage>
    <VaultID>2341234-12431243-2341235</VaultID>
  </VaultOperation>
</Request_v1>
```

## XML Vault Operation Request for Updating a Storage Record hiding the Account Number in the UI

```
<?xml version="1.0" encoding="utf-16"?>
<Request_v1 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Application>
    <ApplicationID>DEMO</ApplicationID>
    <LanguageID>EN</LanguageID>
  </Application>
  <VaultOperation>
    <Merchant>
      <MerchantID>999999999997</MerchantID>
      <MerchantKey>K3QD6YWyhFD</MerchantKey>
    </Merchant>
    <VaultStorage>
      <Service>UPDATE</Service>
      <GUID>sfdas-ee3u38d-dagdi3-efad83</GUID>
    </VaultStorage>
    <VaultID>2341234-12431243-2341235</VaultID>
  </VaultOperation>
  <UI>
    <VaultOperation>
      <AccountNumber>
        <Enabled>true</Enabled>
        <Visible>false</Visible>
      </AccountNumber>
    </VaultOperation>
  </UI>
</Request_v1>
```

## XML Vault Operation Request for Updating a Storage Record masking the Account Number in the UI

```
<?xml version="1.0" encoding="utf-16"?>
<Request_v1 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Application>
    <ApplicationID>DEMO</ApplicationID>
    <LanguageID>EN</LanguageID>
  </Application>
  <VaultOperation>
    <Merchant>
      <MerchantID>999999999997</MerchantID>
      <MerchantKey>K3QD6YWyhFD</MerchantKey>
    </Merchant>
    <VaultStorage>
      <Service>UPDATE</Service>
      <GUID>sfdas-ee3u38d-dagdi3-efad83</GUID>
    </VaultStorage>
    <VaultID>2341234-12431243-2341235</VaultID>
  </VaultOperation>
  <UI>
    <VaultOperation>
      <AccountNumber>
        <Enabled>false</Enabled>
        <Visible>true</Visible>
      </AccountNumber>
    </VaultOperation>
  </UI>
</Request_v1>
```



## Vault Account

Vault Account is used to create a Vault service only merchant account in which no payment processing is possible but Vault storage is needed. The below sample shows the minimum required fields.

### XML Vault Account Request

```
<?xml version="1.0" encoding="utf-16"?>
<Request_v1 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Application>
    <ApplicationID>DEMO</ApplicationID>
    <LanguageID>EN</LanguageID>
  </Application>
  <VaultAccount>
    <Company>
      <Name>Test Account - Vault</Name>
      <Address>
        <AddressLine1>12345 Street</AddressLine1>
        <City>Brownsville</City>
        <State>Texas</State>
        <ZipCode>78520</ZipCode>
        <EmailAddress>none@sagepayments.com</EmailAddress>
        <Telephone>956-548-9400</Telephone>
        <Fax>956-548-9416</Fax>
      </Address>
    </Company>
    <Contact>
      <Name>
        <FirstName>John</FirstName>
        <LastName>Doe</LastName>
      </Name>
      <Address />
      <Company>
        <Address />
      </Company>
    </Contact>
  </VaultAccount>
</Request_v1>
```

## Account Query

Account Query is used to get the status and services available for a particular merchant account. This can be used to determine which features of the Exchange can be used with a merchant account.

### XML Account Query Request

```
<?xml version="1.0" encoding="utf-16"?>
<Request_v1 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Application>
    <ApplicationID>DEMO</ApplicationID>
    <LanguageID>EN</LanguageID>
  </Application>
  <AccountQuery>
    <Merchant>
      <MerchantID>9999999999997</MerchantID>
      <MerchantKey>K3QD6YWyhFD</MerchantKey>
    </Merchant>
  </AccountQuery>
</Request_v1>
```

## Transaction Status Query

Transaction Status Query is used to get the status of a transaction processed through the Sage Exchange by using the user defined Transaction ID provided during a previous Payment. In the event of a communication failure in which the response was not received this can be used to determine if the platform received and processed the transaction or if the transaction needs to be run again. This can be used to prevent duplicated transactions. If post transaction logic is required this can also provide additional information such as the current state of the transaction ( settled / expired / voided ).

### XML Transaction Status Query

```
<?xml version="1.0" encoding="utf-16"?>
<Request_v1 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Application>
    <ApplicationID>DEMO</ApplicationID>
    <LanguageID>EN</LanguageID>
  </Application>
  <TransactionStatusQueries>
    <TransactionStatusQueryType>
      <Merchant>
        <MerchantID>9999999999997</MerchantID>
        <MerchantKey>K3QD6YWyhFD</MerchantKey>
      </Merchant>
      <TransactionID>sdfasdf089273412903479a87sa</TransactionID>
    </TransactionStatusQueryType>
  </TransactionStatusQueries>
</Request_v1>
```

## Vault Status Query

Vault Status Query is used to get the status of a Vault operation processed through the Sage Exchange by using the user defined Vault ID provided during a previous Vault operation. In the event of a communication failure in which the response was not received this can be used to determine if the platform received and processed the operation or if the operation needs to be run again.

### XML Vault Status Query

```
<?xml version="1.0" encoding="utf-16"?>
<Request_v1 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Application>
    <ApplicationID>DEMO</ApplicationID>
    <LanguageID>EN</LanguageID>
  </Application>
  <VaultStatusQuery>
    <Merchant>
      <MerchantID>999999999997</MerchantID>
      <MerchantKey>K3QD6YWyhFD</MerchantKey>
    </Merchant>
    <VaultID>sdfasdf089273412903479a87sa</VaultID>
  </VaultStatusQuery>
</Request_v1>
```

## Multi-Payment Processing

The Sage Exchange application supports processing multiple payments in a single XML request. The request can combine both UI and Non-UI payments. If any payment contains a UI transaction type then the Multi-Payment UI will be shown.

### XML Multi-Payment Request with User Interface

```
<?xml version="1.0" encoding="utf-16"?>
<Request_v1 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Application>
    <ApplicationID>DEMO</ApplicationID>
    <LanguageID>EN</LanguageID>
  </Application>
  <Payments>
    <PaymentType>
      <Merchant>
        <MerchantID>999999999997</MerchantID>
        <MerchantKey>K3QD6YWyhFD</MerchantKey>
      </Merchant>
      <TransactionBase>
        <TransactionID>5ea9747c-12a4-46af-970f-f8a92f6d4f65</TransactionID>
        <TransactionType>11</TransactionType>
        <Reference1>INV# 886478943</Reference1>
        <Amount>1892.59</Amount>
      </TransactionBase>
      <Customer>
        <Name>
          <FirstName>Jane</FirstName>
          <MI> </MI>
          <LastName>Doe</LastName>
        </Name>
        <Address>
          <AddressLine1>67890 Road</AddressLine1>
          <AddressLine2></AddressLine2>
          <City>South Padre Island</City>
          <State>Texas</State>
          <ZipCode>78597</ZipCode>
          <Country>USA</Country>
          <EmailAddress>jane.doe@sagepayments.com</EmailAddress>
          <Telephone></Telephone>
        </Address>
      </Customer>
    </PaymentType>
  </Payments>
</Request_v1>
```

```
<Fax></Fax>
</Address>
</Customer>
</PaymentType>
<PaymentType>
  <Merchant>
    <MerchantID>999999999997</MerchantID>
    <MerchantKey>K3QD6YWyhfD</MerchantKey>
  </Merchant>
  <TransactionBase>
    <TransactionID>4fa9747c-13a2-46af-970f-f8a92f5d4f61</TransactionID>
    <TransactionType>11</TransactionType>
    <Reference1>INV# 7563456</Reference1>
    <Amount>50.50</Amount>
  </TransactionBase>
</Customer>
<Customer>
  <Name>
    <FirstName>John</FirstName>
    <MI> </MI>
    <LastName>Doe</LastName>
  </Name>
  <Address>
    <AddressLine1>4567 Street</AddressLine1>
    <AddressLine2></AddressLine2>
    <City>South Padre Island</City>
    <State>Texas</State>
    <ZipCode>78597</ZipCode>
    <Country>USA</Country>
    <EmailAddress>john.doe@sagepayments.com</EmailAddress>
    <Telephone></Telephone>
    <Fax></Fax>
  </Address>
</Customer>
</PaymentType>
</Payments>
</Request_v1>
```

## Split Payment Processing

The Sage Exchange application supports splitting a single payment across multiple merchant accounts. The total amount of the payment is the combine Amount of all the payments in the array. The first account in the payments array is considered the primary merchant account. The request can combine both UI and Non-UI payments. If any payment contains a UI transaction type then the Split Payment UI will be shown.

### XML Split Payment Request with User Interface

```
<?xml version="1.0" encoding="utf-16"?>
<Request_v1 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Application>
    <ApplicationID>DEMO</ApplicationID>
    <LanguageID>EN</LanguageID>
  </Application>
  <IsSplitPayment>true</IsSplitPayment>
  <Payments>
    <PaymentType>
      <Merchant>
        <MerchantID>999999999997</MerchantID>
        <MerchantKey>K3QD6YWyhfD</MerchantKey>
      </Merchant>
      <TransactionBase>
        <TransactionID>5ea9747c-12a4-46af-970f-f8a92f6d4f65</TransactionID>
        <TransactionType>12</TransactionType>
        <Reference1>INV# 886478943</Reference1>
        <Amount>1.00</Amount>
      </TransactionBase>
      <Customer>
        <Name>
          <FirstName>Jane</FirstName>
          <MI> </MI>
          <LastName>Doe</LastName>
        </Name>
        <Address>
          <AddressLine1>67890 Road</AddressLine1>
          <AddressLine2></AddressLine2>
          <City>South Padre Island</City>
          <State>Texas</State>
          <ZipCode>78597</ZipCode>
          <Country>USA</Country>
          <EmailAddress>jane.doe@sagepayments.com</EmailAddress>
          <Telephone></Telephone>
          <Fax></Fax>
        </Address>
      </Customer>
    </PaymentType>
    <PaymentType>
      <Merchant>
        <MerchantID>999999999990</MerchantID>
        <MerchantKey>D8H8M8F6K7A7</MerchantKey>
      </Merchant>
      <TransactionBase>
        <TransactionID>5ea9747c-12a4-46af-970f-f8a92f6d4f65</TransactionID>
        <TransactionType>12</TransactionType>
        <Reference1>INV# 886478943</Reference1>
        <Amount>1.00</Amount>
      </TransactionBase>
      <Customer>
        <Name>
          <FirstName>John</FirstName>
          <MI> </MI>
          <LastName>Doe</LastName>
        </Name>
        <Address>
          <AddressLine1>67890 Road</AddressLine1>
          <AddressLine2></AddressLine2>
          <City>South Padre Island</City>
          <State>Texas</State>
          <ZipCode>78597</ZipCode>
```

```
<Country>USA</Country>  
<EmailAddress>jane.doe@sagepayments.com</EmailAddress>  
<Telephone></Telephone>  
<Fax></Fax>  
</Address>  
</Customer>  
</PaymentType>  
</Payments>  
</Request_v1>
```

## User Interface Management

The SEVD supports control over UI elements such as background colors and key field elements.

### XML Authorization Request with User Interface Management

```
<?xml version="1.0" encoding="utf-16"?>
<Request v1 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Application>
    <ApplicationID>DEMO</ApplicationID>
    <LanguageID>EN</LanguageID>
  </Application>
  <Payments>
    <PaymentType>
      <Merchant>
        <MerchantID>999999999997</MerchantID>
        <MerchantKey>K3QD6YWyhFD</MerchantKey>
      </Merchant>
      <TransactionBase>
        <TransactionID>5ea9747c-12a4-46af-970f-f8a92f6d4f65</TransactionID>
        <TransactionType>12</TransactionType>
        <Referencel>INV# 886478943</Referencel>
        <Amount>1.00</Amount>
      </TransactionBase>
      <Customer>
        <Name>
          <FirstName>Jane</FirstName>
          <MI> </MI>
          <LastName>Doe</LastName>
        </Name>
        <Address>
          <AddressLine1>67890 Road</AddressLine1>
          <AddressLine2></AddressLine2>
          <City>South Padre Island</City>
          <State>Texas</State>
          <ZipCode>78597</ZipCode>
          <Country>USA</Country>
          <EmailAddress>jane.doe@sagepayments.com</EmailAddress>
          <Telephone></Telephone>
          <Fax></Fax>
        </Address>
      </Customer>
    </PaymentType>
  </Payments>
</UI>
<Display>
  <Header>true</Header>
  <SupportLink>false</SupportLink>
  <CheckPayment>false</CheckPayment>
  <CardPayment>true</CardPayment>
  <SELogo>true</SELogo>
</Display>
<Theme>
  <MainFontColor>#800000</MainFontColor>
  <MainBackColor>#FFF8DC</MainBackColor>
  <HeaderBackColor>#D2691E</HeaderBackColor>
  <TotalsBoxBackColor>#DEB887</TotalsBoxBackColor>
  <DividerBackColor>#CD853F</DividerBackColor>
</Theme>
<SinglePayment>
  <TransactionBase>
    <Referencel>
      <Enabled>false</Enabled>
      <Visible>false</Visible>
    </Referencel>
    <SubtotalAmount>
      <Enabled>false</Enabled>
      <Visible>true</Visible>
    </SubtotalAmount>
    <TaxAmount>
      <Enabled>true</Enabled>
      <Visible>true</Visible>
```

```
</TaxAmount>
<ShippingAmount>
  <Enabled>false</Enabled>
  <Visible>false</Visible>
</ShippingAmount>
</TransactionBase>
<Customer>
  <Name>
    <FirstName>
      <Enabled>false</Enabled>
      <Visible>true</Visible>
    </FirstName>
    <LastName>
      <Enabled>false</Enabled>
      <Visible>false</Visible>
    </LastName>
  </Name>
  <Address>
    <AddressLine1>
      <Enabled>false</Enabled>
      <Visible>false</Visible>
    </AddressLine1>
    <City>
      <Enabled>false</Enabled>
      <Visible>true</Visible>
    </City>
    <State>
      <Enabled>false</Enabled>
      <Visible>true</Visible>
    </State>
    <ZipCode>
      <Enabled>false</Enabled>
      <Visible>true</Visible>
    </ZipCode>
    <Country>
      <Enabled>false</Enabled>
      <Visible>true</Visible>
    </Country>
    <EmailAddress>
      <Enabled>false</Enabled>
      <Visible>true</Visible>
    </EmailAddress>
    <Telephone>
      <Enabled>false</Enabled>
      <Visible>true</Visible>
    </Telephone>
  </Address>
</Customer>
</SinglePayment>
</UI>
</Request_v1>
```



## XML Vault Update Operation with User Interface Management

```
<?xml version="1.0" encoding="utf-16"?>
<Request v1 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Application>
    <ApplicationID>DEMO</ApplicationID>
    <LanguageID>EN</LanguageID>
  </Application>
  <VaultOperation>
    <Merchant>
      <MerchantID>9999999999997</MerchantID>
      <MerchantKey>K3QD6VWyhfD</MerchantKey>
    </Merchant>
    <VaultStorage>
      <Service>UPDATE</Service>
      <GUID>sfdas-ee3u38d-dagdi3-efad83</GUID>
    </VaultStorage>
    <VaultID>2341234-12431243-2341235</VaultID>
  </VaultOperation>
</UI>
  <Display>
    <Header>true</Header>
    <SupportLink>>false</SupportLink>
    <CheckPayment>>false</CheckPayment>
    <CardPayment>>true</CardPayment>
    <SELogo>>true</SELogo>
  </Display>
  <Theme>
    <MainFontColor>#800000</MainFontColor>
    <MainBackColor>#FFF8DC</MainBackColor>
    <HeaderBackColor>#D2691E</HeaderBackColor>
    <TotalsBoxBackColor>#DEB887</TotalsBoxBackColor>
    <DividerBackColor>#CD853F</DividerBackColor>
  </Theme>
  <VaultOperation>
    <AccountNumber>
      <Enabled>>false</Enabled>
      <Visible>>false</Visible>
    </AccountNumber>
  </VaultOperation>
</UI>
</Request_v1>
```

## Request Field Definitions

### Request\_V1 Type Element

*Required*

```
<Request_V1>
  <Application></Application>
  <Payments></Payments>
  <IsSplitPayment></IsSplitPayment>
</Request_V1>
```

The Request\_V1 element is the root. It contains the Application, Payments, and IsSplitPayment elements.

Element Name	Data Type	Length	Required	Comments
<b>Application</b>	ApplicationType		Yes	Identifies the calling application, version, and its certification as a valid integrated solution.
<b>Payments</b>	Array of PaymentType		No	An array of payments to be processed.
<b>IsSplitPayment</b>	boolean		No	Indicates if the array of payments is to be treated as split payments or multi payments. If omitted the default is false.

### Application Type Element

*Required*

```
<Application>
  <ApplicationID></ApplicationID>
  <LanguageID></LanguageID>
</Application>
```

The Application element contains the ApplicationID and LanguageID elements.

Element Name	Data Type	Length	Required	Comments
<b>ApplicationID</b>	string	50	Yes	Identifies the calling application, version, and its certification as a valid integrated solution. The value is obtained from Sage Payment Solutions through a registration/certification process.
<b>LanguageID</b>	string	10	No	Specifies the language to be used when displaying the user interface, the default is "en-US" for English United States. The values are derived from the lower case 2 letter language code from ISO 639-1 and the two letter upper case from ISO 3166.  Example: "fr-CA" = French Canadian

## MerchantType Element

*Required*

```
<Merchant>
  <MerchantID></MerchantID>
  <MerchantKey></MerchantKey>
</Merchant>
```

The Merchant element contains the required MerchantID and MerchantKey elements.

Element Name	Data Type	Length	Required	Comments
MerchantID	string	12	Yes	Identifies the merchant account on the VAN
MerchantKey	string	12	Yes	Identifies the merchant account on the VAN

## Payments Element

*Optional*

```
<Payments>
  <PaymentType></PaymentType>
</Payments>
```

The Payments element can contain one or more PaymentType elements. When more than one payment is required to be processed by the calling application multiple PaymentType elements are used. When only a single payment is required only a single PaymentType element is used. At least one PaymentType element is required in a payment request message.

## Batch Element

*Optional*

```
<Batch>
  <BatchType></BatchType>
</Batch>
```

The Batch element can contain one BatchType element.

## PaymentType Element

*Required*

```
<PaymentType>
  <Merchant></Merchant>
  <TransactionBase></TransactionBase>
  <Customer></Customer>
  <ShippingRecipient></ShippingRecipient>
  <Level2></Level2>
  <Level3></Level3>
  <VaultStorage></VaultStorage>
  <Recurring></Recurring>
  <Postback></Postback>
</PaymentType>
```

The PaymentType element refers to a single payment and must contain a Merchant and TransactionBase element and may contain the optional Customer, ShippingRecipient, Level2, Level3, VaultStorage, and Recurring elements.

Element Name	Data Type	Length	Required	Comments
<b>Merchant</b>	MerchantType		Yes	Contains the merchant account elements related to processing a payment.
<b>TransactionBase</b>	TransactionBaseType		Yes	Contains the transaction elements related to processing a payment.
<b>Customer</b>	PersonType		No*	Contains the elements related to the person making the payment.
<b>ShippingRecipient</b>	PersonType		No	Contains the elements related to the person receiving the goods/services related to the payment.
<b>Level2</b>	Level2Type		No	Contains the additional transaction elements related to qualifying for Level II. The Level2 and Level3 elements are mutually exclusive and should not be used together in a payment.
<b>Level3</b>	Level3Type		No	Contains the additional transaction elements related to qualifying for Level III. The Level2 and Level3 elements are mutually exclusive and should not be used together in a payment.
<b>VaultStorage</b>	VaultStorageType		No*	Contains the elements related to Storing/Retrieving/Updating data in the Vault service.
<b>Recurring</b>	RecurringType		No	Contains the elements needed to schedule a payment in the recurring system.

Element Name	Data Type	Length	Required	Comments
<b>Postback</b>	PostbackType		No	Contains the elements related to POSTing transaction response data to a publically available user defined URL.

*\*Element is required for a Sale, Authorization, or Force without a user interface*

## TransactionBaseType Element

*Required*

```

<TransactionBase>
  <TransactionID></TransactionID>
  <TransactionType></TransactionType>
  <Reference1></Reference1>
  <Amount></Amount>
  <AuthCode></AuthCode>
  <VANReferernce></VANReference>
</TransactionBase>

```

The TransactionBase element contains the TransactionID, TransactionType, Reference1, Amount, AuthCode, and VANReference elements.

Element Name	Data Type	Length	Required	Comments
<b>TransactionID</b>	string	32	Yes	Identifies the payment for status inquires. In the event the communication to the Payment Module is interrupted a payment response can be queried later using this element.
<b>TransactionType</b>	integer	2	Yes	No user interface: 01 = Sale 02 = Authorization 03 = Capture 04 = Void 05 = Force 06 = Credit 07 = Credit without Reference  User interface: 11 = Sale 12 = Authorization 13 = Capture 15 = Force 16 = Credit 17 = Credit without Reference
<b>Reference1</b>	string	50	No	User defined field, like Invoice Number, Purchase Order Number, etc...
<b>Amount</b>	decimal	9	No	Amount of the payment
<b>AuthCode</b>	string	6	No*	Authorization Code for Force transactions
<b>VANReference</b>	string	16	No**	VAN unique transaction identifier used with Credit, Capture, or Void transactions

*\*Element is required for a Force transaction*

*\*\*Element is required for a Credit, Capture, or Void transaction*

## Customer Element: PersonType Element

*Optional*

```
<Customer>
  <Name></Name>
  <Address></Address>
  <Company></Company>
</Customer>
```

The Customer element contains the Name, Address, and Company elements for the person making the payment.

Element Name	Data Type	Length	Required	Comments
<b>Name</b>	NameType		No	Contains elements related to the Customer's full name
<b>Address*</b>	AddressType		No	Contains elements related to the billing address, used during Address Verification Service for manually keyed transactions
<b>Company</b>	CompanyType		No	Contains elements related to the Customer's company

*\* Elements used for AVS when payment is manually keyed*

## ShippingRecipient Element: PersonType Element

*Optional*

```
<ShippingRecipient>
  <Name></Name>
  <Address></Address>
  <Company></Company>
</ShippingRecipient>
```

The ShippingRecipient element contains the Name, Address, and Company elements for the person receiving the goods/service.

Element Name	Data Type	Length	Required	Comments
<b>Name</b>	NameType		No	Contains elements related to the full name
<b>Address</b>	AddressType		No	Contains elements related to the address
<b>Company</b>	CompanyType		No	Contains elements related to the company

## Level2Type Element

*Optional*

```
<Level2>
  <CustomerNumber></CustomerNumber>
  <TaxAmount></TaxAmount>
</Level2>
```

The Level2 element contains the CustomerNumber and TaxAmount elements used to qualify a Purchase Card payment for Level II.

Element Name	Data Type	Length	Required	Comments
CustomerNumber	string	17	Yes	User defined reference number
TaxAmount	decimal	9	Yes	Tax amount applied to the transaction

## Level3Type Element

*Optional*

```
<Level3>
  <Level2></Level2>
  <ShippingAmount></ShippingAmount>
  <DestinationZipCode></DestinationZipCode>
  <DestinationCountryCode></DestinationCountryCode>
  <VATNumber></VATNumber>
  <DiscountAmount></DiscountAmount>
  <DutyAmount></DutyAmount>
  <NationalTaxAmount></NationalTaxAmount>
  <VATInvoiceNumber></VATInvoiceNumber>
  <VATTaxAmount></VATTaxAmount>
  <VATTaxRate></VATTaxRate>
  <LineItems></LineItems>
</Level3>
```

The Level3 element contains the optional CustomerNumber, TaxAmount, DestinationZipCode, NationalTax, VATNumber, DiscountAmount, DutyAmount, VATInvoiceNumber, VATTaxAmount, VATTaxRate, DestinationCountryCode, and LineItems elements used to qualify a Purchase Card payment for Level III. If both Level3 and Level2 elements are present Level3 will supersede Level2.

Element Name	Data Type	Length	Required	Comments
Level2	Level2Type		Yes	Contains the Level2 elements CustomerNumber and TaxAmount
ShippingAmount	decimal	9	Yes	Shipping amount charged to the transaction
DestinationZipCode	string	9	Yes	Postal zip code where the goods/services are shipped
VATNumber	string	13	Yes	Customer's Value Added Tax Number
DiscountAmount	decimal	9	Yes	Discount amount applied to the transaction
DutyAmount	decimal	9	Yes	Duty amount
VATInvoiceNumber	string	15	Yes	Value Added Tax invoice number
VATTaxAmount	decimal	9	Yes	Value Added Tax amount
VATTaxRate	decimal	9	Yes	Value Added Tax rate
DestinationCountryCode	integer	3	Yes	ISO Country Code where the goods/services are shipped

Element Name	Data Type	Length	Required	Comments
<b>LineItems</b>	Level3LineItemT		No	Contains zero or more
<b>NationalTaxAmount</b>	Decimal	9	Yes	National Tax amount applied to the transaction

## LineItems Element

*Optional*

```
<LineItems>
  <Level3LineItemType></Level3LineItemType>
</LineItems>
```

The LineItems element can contain zero or more Level3LineItemType elements. When more than one Level III line item is required to be processed by the calling application multiple Level3LineItemType elements are used.

## Level3LineItemType Element

*Optional*

```
<Level3LineItemType>
  <CommodityCode></CommodityCode>
  <Description></Description>
  <ProductCode></ProductCode>
  <Quantity></Quantity>
  <UnitOfMeasure></UnitOfMeasure>
  <UnitCost></UnitCost>
  <TaxAmount></TaxAmount>
  <TaxRate></TaxRate>
  <DiscountAmount></DiscountAmount>
  <AlternateTaxIdentifier></AlternateTaxIdentifier>
  <TaxTypeApplied></TaxTypeApplied>
  <DiscountIndicator></DiscountIndicator>
  <NetGrossIndicator></NetGrossIndicator>
  <ExtendedItemAmount></ExtendedItemAmount>
  <DebitCreditIndicator></DebitCreditIndicator>
</Level3LineItemType>
```

The Level3LineItemType element contains the CommodityCode, Description, ProductCode, Quantity, UnitOfMeasure, UnitCost, TaxAmount, TaxRate, DiscountAmount, AlternateTaxIdentifier, TaxTypeApplied, DiscountIndicator, NetGrossIndicator, ExtendedItemAmount, DebitCreditIndicator, and TotalAmount elements.

Element Name	Data Type	Length	Required	Comments
<b>CommodityCode</b>	string	12	Yes	Commodity code that applies to the item
<b>Description</b>	string	35	Yes	A brief description of the item
<b>ProductCode</b>	string	12	Yes	Product code that applies to the item
<b>Quantity</b>	integer	12	Yes	Quantity of item(s) purchased
<b>UnitOfMeasure</b>	string	12	Yes	Units of measure of the item(s) purchased
<b>UnitCost</b>	decimal	9	Yes	Cost of the item purchased
<b>TaxAmount</b>	decimal	9	Yes	The tax amount for the item
<b>TaxRate</b>	decimal	5	Yes	The tax rate for the item
<b>DiscountAmount</b>	decimal	9	Yes	Discount amount applied to item



Element Name	Data Type	Length	Required	Comments
AlternateTaxIdentifier		15	Yes	
TaxTypeApplied		4	Yes	
DiscountIndicator	string	1	Yes	
NetGrossIndicator	string	1	Yes	
ExtendedItemAmount	decimal	9	Yes	The total amount of the individual item. = ( ItemCost X Quantity ) – ( DiscountAmount x Quantity
DebitCreditIndicator	string	1	Yes	

## VaultStorageType Element

*Optional*

```
<VaultStorage>
  <Service></Service>
  <GUID></GUID>
</VaultStorage>
```

The Vault element contains the Service, and GUID elements.

Element Name	Data Type	Length	Required	Comments
Service	VaultServiceType		Yes	Used to indicate the Vault operation.  RETRIEVE = Pull data from Vault for processing  UPDATE = Update data in Vault with new data captured  CREATE = Insert data in the Vault with data captured
GUID	string	36	NO	The the Vault GUID referencing a previous payment captured in the Vault. Payment information will not need to be captured and instead it is retrieved from the Vault.

## RecurringType Element

*Optional*

```
<Recurring>
  <Schedule></Schedule>
  <Interval></Interval>
  <DayOfMonth></DayOfMonth>
  <StartDate></StartDate>
  <Amount></Amount>
  <TimesToProcess></TimesToProcess>
  <NonBusinessDay></NonBusinessDay>
</Recurring>
```

The Recurring element contains the ScheduleType, ScheduleInterval, DayOfMonth, StartDate, Amount, TimesToProcess, and NonBusinessDay elements.

Element Name	Data Type	Length	Required	Comments
<b>Schedule</b>	ScheduleType		Yes	The schedule type for the recurring transaction  DAILY = Schedule will be based on day  MONTHLY = Schedule will be based on month
<b>Interval</b>	integer	6	No	The interval between processing, Example: Every Other Month  Schedule = MONTHLY Interval = 2
<b>DayOfMonth</b>	integer	2	No	The day of the month to process the payment on when schedule is based monthly
<b>StartDate</b>	string	10	No	Format MM/DD/YYYY
<b>Amount</b>	decimal	9	No	The amount to charge, if different than the payment amount
<b>TimesToProcess</b>	integer	6	No	The number of payments to process -1 = Indefinite
<b>NonBusinessDay</b>	NonBusinessDayType	1	No	How to handle processing when the date is a non business day  THATDAY = Transaction is processed that day  BEFORE = Transaction is processed the day before  AFTER = Transaction is processed on the next business day

## BatchType Element

*Optional*

```
<BatchType>
  <Merchant></Merchant>
  <Net></Net>
  <Count></Count>
  <BatchPayment></BatchPayment>
</BatchType>
```

The BatchType element refers to a single payment and must contain a Merchant and BatchPayment element and may contain the optional Net and Count elements.

Element Name	Data Type	Length	Required	Comments
<b>Merchant</b>	MerchantType		Yes	Contains the elements related to the merchant account batch.
<b>Net</b>	decimal		No	The transaction total net amount of the batch being settled
<b>Count</b>	integer		No	The transaction count of the batch being settled
<b>BatchPayment</b>	BatchPaymentType		Yes	The payment type of batch being settled. CREDITCARD = Standard Credit Card transactions  PURCHASECARD = Level III qualified transactions

## PostbackType Element

*Optional*

```
<Postback>
  <HttpsUrl></HttpsUrl>
</Postback>
```

The PostbackType element refers to a URL in which transaction response data should be sent after processing and must contain a HttpsUrl element.

Element Name	Data Type	Length	Required	Comments
<b>HttpsUrl</b>	string	1024	Yes	The absolute URL to POST transaction response data back to. The communication requires SSL. This is to be used in integrations in which response handling is disconnected or in which response data should be delivered to a remote server.

## CompanyType Element

*Optional*

```
<Company>
  <Name></Name>
  <Address></Address>
</Company>
```

The CompanyType element refers to a Company and must contain a Name element and optional Address element.

Element Name	Data Type	Length	Required	Comments
<b>Name</b>	string	50	Yes	The company's DBA.
<b>Address</b>	AddressType		No	The company's mailing address.

## NameType Element

```
<Name>
  <FirstName></FirstName>
  <MI></MI>
  <LastName></LastName>
</Name>
```

The NameType element refers to a person's full name and must contain both a FirstName element and LastName element and a optional MI element.

Element Name	Data Type	Length	Required	Comments
<b>FirstName</b>	string	50	Yes	A person's first name
<b>MI</b>	string	1	No	The first letter of a person's middle
<b>LastName</b>	string	50	Yes	A person's last name

## AddressType Element

```
<Address>
  <AddressLine1></AddressLine1>
  <AddressLine2></AddressLine2>
  <City></City>
  <State></State>
  <ZipCode></ZipCode>
  <Country></Country>
  <EmailAddress></EmailAddress>
  <Telephone></Telephone>
  <Fax></Fax>
</Address>
```

The AddressType element refers to a person or a company address and must contain Address1, City, State, ZipCode, Country elements and the optional Address2, EmailAddress, Telephone, and Fax elements. This element is used during AVS verification when used in the Customer element.

Element Name	Data Type	Length	Required	Comments
AddressLine1	string	50	Yes	Street number
AddressLine2	string	50	No	Appartment or Suite Number
City	string	50	Yes	City
State	string	50	Yes	State
ZipCode	string	50	Yes	Postal Code
Country	string	50	Yes	Country
EmailAddress	string	256	No	Email Address
Telephone	string	50	No	Telepone number with area code
Fax	string	50	No	Fax number with area code

## TransactionStatusQueryType Element

```
<TransactionStatusQueryType>
  <Merchant></Merchant>
  <TransactionID></TransactionID>
</TransactionStatusQueryType>
```

The TransactionStatusQueryType element is used to query the status of a single transaction and must contain both Merchant and TransactionID elements.

Element Name	Data Type	Length	Required	Comments
Merchant	MerchantTyp		Yes	Merchant account elements
TransactionID	string	50	Yes	The unique user assigned transaction identifier used during a payment

## RecurringStatusQueryType Element

```
<RecurringStatusQueryType>
  <Merchant></Merchant>
  <RecurringID></RecurringID>
  <StartDate></StartDate>
  <EndDate></EndDate>
</RecurringStatusQueryType>
```

The RecurringStatusQueryType element is used to query the status of a recurring transaction and must contain Merchant, RecurringID, StartDate, and EndDate elements.

Element Name	Data Type	Length	Required	Comments
<b>Merchant</b>	MerchantType		Yes	Merchant account elements
<b>RecurringID</b>	string	50	Yes	The unique system assigned recurring identifier created during a recurring payment
<b>StartDate</b>	string	9	Yes	The start date for the query search
<b>EndDate</b>	string	9	Yes	The end date for the query search

## VaultOperationType Element

```
<VaultOperationType>
  <Merchant></Merchant>
  <VaultStorage></VaultStorage>
  <VaultID></VaultID>
</VaultOperationType>
```

The VaultOperationType element is used to insert or update a Vault storage record and must contain the Merchant and VaultStorage elements. This type of request will only capture and store data and will not process a payment.

Element Name	Data Type	Length	Required	Comments
<b>Merchant</b>	MerchantType		Yes	Merchant account elements
<b>VaultStorage</b>	VaultStorageType		Yes	The Vault service type and GUID elements.
<b>VaultID</b>	string	50	No	A unique identifier that can be used to query the results of a Vault operation

## VaultAccountType Element

```
<VaultAccountType>
  <Company></Company>
  <Contact></Contact>
</VaultAccountType>
```

The VaultAccountType element is used to create a Vault only Merchant Account and must contain the Company and Contact elements.

Element Name	Data Type	Length	Required	Comments
<b>Company</b>	CompanyType		Yes	Company that will be storing data in the Vault
<b>Contact</b>	PersonType		Yes	The contact for the Company storing data in the Vault

## AccountQueryType Element

```
<AccountQueryType>
  <Merchant></Merchant>
</AccountQueryType>
```

The AccountQueryType element is used to check the status and services available for a merchant account.

Element Name	Data Type	Length	Required	Comments
<b>Merchant</b>	MerchantType		Yes	Merchant account elements
<b>Contact</b>	PersonType		Yes	The contact for the Company storing data in the Vault

## UIType Element

```
<UIType>
  <SinglePayment></SinglePayment>
</UIType>
```

The UIType element is used to control the user experience.

Element Name	Data Type	Length	Required	Comments
<b>SinglePayment</b>	SinglePaymentUIType		No	Elements that provide additional control over the user experience during a Single Payment.

## Response Field Definitions

### PaymentResponseType Element

*Required*

```
<PaymentResponseType>
  <Response></Response>
  <VaultResponse></VaultResponse>
  <RecurringResponse></RecurringResponse>
  <TransactionResponse></TransactionResponse>
</PaymentResponseType>
```

The PaymentResponseType element contains the response elements related to a payment.

Element Name	Data Type	Length	Required	Comments
<b>Response</b>	ResponseType		Yes	Contains the response elements related to the Gateway request
<b>VaultResponse</b>	VaultResponseType		No	Contains the response elements related to the Vault operation
<b>RecurringResponse</b>	RecurringResponseType		No	Contains the response elements related to the transaction enlistment into the recurring system

Element Name	Data Type	Length	Required	Comments
<b>TransactionResponse</b>	TransactionResponse Type		Yes	Contains the response elements related to processing a transaction on the Gateway
<b>Customer</b>	PersonType		Yes	Contains the billing/customer elements used for the payments

## ResponseType Element

*Required*

```

<ResponseType>
  <ResponseIndicator></ResponseIndicator>
  <ResponseCode></ResponseCode>
  <ResponseMessage></ResponseMessage>
</ResponseType>

```

The ResponseType element contains the the Gateway response elements related to a Gateway request.

Element Name	Data Type	Length	Required	Comments
<b>ResponseIndicator</b>	string	1	Yes	The Gateway/Vault response indicator  A=Approved E=Declined X=Error I = Batch Inquiry ( only returned on batch inquiry requests )  <b>When returned during a Payment request this field is used to determine the status of a Payment ( Approved/Declined/Error)</b>
<b>ResponseCode</b>	string	6	Yes	The Gateway/Vault response code, when the indicator is a E or X this code and be used to determine the cause
<b>ResponseMessage</b>	string	32	Yes	The Gateway/Vault response text

## VaultResponseType Element

```

<VaultResponseType>
  <Response></Response>
  <GUID></GUID>
  <ExpirationDate></ExpirationDate>
  <Last4></Last4>
  <PaymentDescription></PaymentDescription>
  <PaymentTypeID></PaymentTypeID>
</VaultResponseType>

```

The VaultResponseType element contains the the response elements related to a Vault operation.

Element Name	Data Type	Length	Required	Comments
<b>Response</b>	ResponseType		Yes	Contains the response elements related to the Vault request
<b>GUID</b>	string	36	Yes	The Vault GUID used to reference the card data captured



Element Name	Data Type	Length	Required	Comments
ExpirationDate	string	4	Yes	The date the account data will expire, format MMY  <b>*This field is only applicable to responses with PaymentTypeID ( 3,4,5,6,7,D,O )</b>
Last4	string		Yes	A masked representation of the account data showing only the last four digits
PaymentDescription	string		Yes	The description of the account data.  <b>Credit Card payments will be the First 6 digits + masked digits + Last 4 digits ACH payments will be the routing number + space + masked account digits + Last 4 account digits</b>
PaymentTypeID	string	1	Yes	The payment type identifier  3=American Express 4=Visa 5=MasterCard 6=Discover 7=JCB D=Debit Card O=Other C=ACH

## RecurringResponseType Element

```
<RecurringResponseType>
  <RecurringID></RecurringID>
</RecurringResponseType>
```

The RecurringResponseType element contains the the recurring identifier element to reference the recurring transaction.

Element Name	Data Type	Length	Required	Comments
RecurringID	string		Yes	The recurring record identifier for the recurring transaction enlised in the recurring system

## TransactionResponseType Element

```
<TransactionResponseType>
  <AuthCode></AuthCode>
  <AVSResult></AVSResult>
  <CVVResult></CVVResult>
  <VANReferene></VANReference>
  <TransactionID></TransactionID>
  <Last4></Last4>
  <PaymentDescription></PaymentDescription>
  <Amount></Amount>
  <PaymentTypeID></PaymentTypeID>
  <EntryMode></EntryMode>
  <NetworkID></NetworkID>
</TransactionResponseType>
```

The TransactionResponseType element contains the the response elements related to processing a transaction.

Element Name	Data Type	Length	Required	Comments
AuthCode	string	6	Yes	The authorization code for the approved payment
AVSResult	string	1	No	The AVS result for manually keyed transactions.  <b>This field is for information purposes and is not to be used to determine the status of a Payment.</b>  <b>*This field is only applicable to responses with PaymentTypeID ( 3,4,5,6,7,D,O )</b>
CVVResult	string	1	No	The CVV result for matching the verification value.  <b>This field is for information purposes and is not to be used to determine the status of a Payment.</b>  <b>*This field is only applicable to responses with PaymentTypeID ( 3,4,5,6,7,D,O )</b>
VANReference	string	10	Yes	The Gateway transaction reference to be used later in captures, voids, and credits.
TransactionID	string		Yes	The unique transaction identifier assigned by the calling application in the transaction request
Last4	string		Yes	A masked representation of the account data showing only the last four digits
PaymentDescription	string		Yes	The description of the account data.  <b>Credit Card payments will be the First 6 digits + masked digits + Last 4 digits</b>  <b>ACH payments will be the routing number + space + masked account digits + Last 4 account digits</b>
Amount	decimal		Yes	Amount of the payment
PaymentTypeID	string	1	Yes	The payment type identifier  3=American Express 4=Visa 5=MasterCard 6=Discover 7=JCB D=Debit Card O=Other C=ACH  <b>*Additional values could be added when future</b>
EntryMode	String	1	Yes	The method of entry:  K = Manually Keyed H = Track 1 D = Track 2 M = MICR Read

Element Name	Data Type	Length	Required	Comments
NetworkID	String	1	No	P = Paymentech T = TSYS E = Elavon F = First Data S = SPS ACH  *Additional values could be added when future networks become available to the system.

## BatchResponseType Element

```
<BatchResponseType>
  <Response></Response>
  <BatchNumber></BatchNumber>
  <BatchReference></BatchReference>
  <Net></Net>
  <Count></Count>
  <BatchPayment></BatchPayment>
</BatchResponseType>
```

The BatchResponseType element contains the the response elements related to processing a batch.

Element Name	Data Type	Length	Required	Comments
<b>Response</b>	ResponseType		Yes	Contains the response elements related to the Gateway request
<b>BatchNumber</b>	string	6	Yes	The batch sequence number
<b>BatchReference</b>	string	10	Yes	The unique Gateway batch identifier
<b>Net</b>	decimal		Yes	The net transaction amount of the batch, a negative amount is possible when processing credits/refunds
<b>Count</b>	integer		Yes	The total transaction count of the batch
<b>BatchPayment</b>	BatchPaymentType		Yes	CREDITCARD PURCHASECARD

## TransactionSettlementStatusType Element

```
<TransactionSettlementStatusType>
  <TransactionType></TransactionType>
  <SettlementType></SettlementType>
  <SettlementDate></SettlementDate>
  <BatchReference></BatchReference>
</TransactionSettlementStatusType>
```

The TransactionSettlementStatusType element contains the settlement status elements related to a transaction.

Element Name	Data Type	Length	Required	Comments
<b>TransactionType</b>	integer		Yes	The transaction type of the transaction processed
<b>SettlementType</b>	integer		Yes	The settlement type of the transacton processed  0 = Error / Declined 1= Batch 2 = Settled 3 = Expired
<b>SettlementDate</b>	string		No	The settlement date, if settled, for the transaction processed; MM/DD/YYYY HH:MM:SS
<b>BatchReference</b>	string	10	No	The batch reference, if settled, for the transaction processed

## TransactionStatusQueryResponseType Element

```
<TransactionStatusQueryResponseType>
  <Response></Response>
  <VaultResponse></VaultResponse>
  <RecurringResponse></RecurringResponse>
  <TransactionResponse></TransactionResponse>
  <TransactionSettlementStatus></TransactionSettlementStatus>
  <Customer></Customer>
</TransactionStatusQueryResponseType>
```

The TransactionStatusQueryResponseType element contains the response elements related to processing a transaction.

Element Name	Data Type	Length	Required	Comments
<b>Response</b>	ResponseType		Yes	Contains the response elements related to the Gateway request
<b>VaultResponse</b>	VaultResponseType		No	Contains the response elements related to the Vault operation
<b>RecurringResponse</b>	RecurringResponseType		No	Contains the response elements related to the transaction enlistment into the recurring system
<b>TransactionResponse</b>	TransactionResponseType		Yes	Contains the response elements related to processing a transaction on the Gateway
<b>TransactionSettlementStatus</b>	TransactionSettlementType		Yes	Contains the transaction settlement status elements
<b>Customer</b>	PersonType		Yes	Contains the billing/customer elements used for the payments

## RecurringStatusQueryResponseType Element

```
<RecurringStatusQueryResponseType>
  <Responses></Responses>
  <TransactionResponses></TransactionResponses>
  <TransactionSettlementStatuses></TransactionSettlementStatuses>
  <Customers></Customers>
</RecurringStatusQueryResponseType>
```

The RecurringStatusQueryResponseType element contains the response elements related to a recurring transaction processed by the Sage system.

Element Name	Data Type	Length	Required	Comments
<b>Responses</b>	ResponseType	Array	Yes	Contains the response elements related to the Gateway request
<b>TransactionResponses</b>	TransactionResponseType	Array	Yes	Contains the response elements related to the Vault operation
<b>TransactionSettlementStatuses</b>	TransactionSettlementType	Array	Yes	Contains the response elements related to the transaction enlistment into the recurring system
<b>Customers</b>	PersonType	Array	Yes	Contains the response elements related to processing a transaction on the Gateway

## VaultAccountResponseType Element

```
<VaultAccountResponseType>
  <Response></Response>
  <VaultAccount></VaultAccount>
  <Merchant></Merchant>
</VaultAccountResponseType>
```

The RecurringStatusQueryResponseType element contains the response elements related to a recurring transaction processed by the Sage system.

Element Name	Data Type	Length	Required	Comments
<b>Response</b>	ResponseType		Yes	Contains the response elements related to the account creation request
<b>VaultAccount</b>	VaultAccountType		Yes	Contains the account elements provided in the Request
<b>Merchant</b>	TransactionSettlementType	Array	Yes	Contains the newly created MerchantID and MerchantKey elements

## AccountQueryResponseType Element

```
<AccountQueryResponseType>
  <Response></Response>
  <Merchant></Merchant>
  <Services></Services>
  <Active></Active>
</AccountQueryResponseType>
```

The AccountQueryResponseType element contains the response elements related to a merchant account status and service list inquiry.

Element Name	Data Type	Length	Required	Comments
<b>Response</b>	ResponseType		Yes	Contains the response elements related to the account inquiry request
<b>Merchant</b>	MerchantType		Yes	Contains the account elements provided in the request
<b>Services</b>	string	Array	Yes	Contains the an array of service descriptions available to a merchant account.  VAULT = "Vault storage service" CREDITCARD = "Credit Card payment service"
<b>Active</b>	boolean		Yes	The status of the merchant account

## Field Validation

XML reserved characters must be observed and encoded appropriately.

## Alpha Numeric Fields\*

The following is a list of accepted characters for Alpha Numeric fields:

## White Space

a-z

A-Z

0-9

-. , # & ( ) / ! ' é è ê ë ò ó ô õ ö à á â ã ä å ì í î ï ù ú û ü ý ÿ

## Numeric Fields\*

The following is a list of accepted characters for Numeric fields:

0-9

1. —

\*The use of two or more dashes -- back to back is not permitted in any of the fields.

## Gateway Error Codes

Code	Message	Description
000000	INTERNAL_SERVER_ERROR	Server Error
900000	INVALID_T_ORDERNUM	Order number value is in an invalid format
900001	INVALID_C_NAME	Name value is in an invalid format or was left blank
900002	INVALID_C_ADDRESS	Address value is in an invalid format or was left blank
900003	INVALID_C_CITY	City value is in an invalid format or was left blank
900004	INVALID_C_STATE	State value is in an invalid format or was left blank
900005	INVALID_C_ZIP	Zip code value is in an invalid format or was left blank
900006	INVALID_C_COUNTRY	Country value is in an invalid format or was left blank
900007	INVALID_C_TELEPHONE	Telephone value is in an invalid format or was left blank
900008	INVALID_C_FAX	Fax value is in an invalid format or was left blank
900009	INVALID_C_EMAIL	Email value is in an invalid format or was left blank
900010	INVALID_C_SHIP_NAME	Shipping address name value is in an invalid format
900011	INVALID_C_SHIP_ADDRESS	Shipping Address value is in an invalid format
900012	INVALID_C_SHIP_CITY	Shipping city value is in an invalid format
900013	INVALID_C_SHIP_STATE	Shipping state value is in an invalid format
900014	INVALID_C_SHIP_ZIP	Shipping zip code value is in an invalid format
900015	INVALID_C_SHIP_COUNTRY	Shipping country value is in an invalid format
900016	INVALID_C_CARDNUMBER	Credit card number value is in an invalid format
900017	INVALID_C_EXP	Expiration date value is in an invalid format
900018	INVALID_C_CVV	CVV (card verification value) value is in an invalid format or was left blank (if set to required)
900019	INVALID_T_AMT	Grand Total must equal > \$0.00. Please check subtotal, shipping and tax values.
900020	INVALID_T_CODE	Transaction Code value is in an invalid format or was left blank
900021	INVALID_T_AUTH	Authorization code is in an invalid format or was left blank (required for Force transactions)
900022	INVALID_T_REFERENCE	Reference value is in an invalid format or was left blank (Required for Force or Void by Reference)
900023	INVALID_T_TRACKDATA	Track Data value is in an invalid format or was left blank (required for debit and retail transactions)
900024	INVALID_T_TRACKING_NUMBER	Tracking number value is in an invalid format
900025	INVALID_T_CUSTOMER_NUMBER	Customer number value is in an invalid format (used only for PCLIII transactions)
900026	INVALID_T_SHIPPING_COMPANY	Shipping company value is in an invalid format
900027	INVALID_T_RECURRING	Recurring value is in an invalid format (must be = 0 or 1)
900028	INVALID_T_RECURRING_TYPE	Recurring value is in an invalid format
900029	INVALID_T_RECURRING_INTERVAL	Recurring interval value is in an invalid format (must be numeric)
900030	INVALID_T_RECURRING_INDEFINITE	Recurring indefinite value is in an invalid format or was left blank
900031	INVALID_T_RECURRING_TIMES_TO_PROCESS	Recurring times to process value is in an invalid format (must be numeric)
900032	INVALID_T_RECURRING_NON_BUSINESS_DAY	Recurring non business days value is in an invalid format



Code	Message	Description
900033	INVALID_T_RECURRING_GROUP	Recurring Group was left blank or group not found
900034	INVALID_T_RECURRING_START_DATE	Recurring start date value is in an invalid format or was left blank
900035	INVALID_T_PIN	Pin number entered is incorrect (required for Pin- debit transactions)
901000		General data validation error, the message will contain additional information
910000	SERVICE NOT ALLOWED	The transaction you are trying to submit is not allowed.
910001	VISA NOT ALLOWED	Visa card type transactions are not allowed.
910002	MASTERCARD NOT ALLOWED	MasterCard card type transactions are not allowed.
910003	AMEX NOT ALLOWED	American Express card type transactions are not allowed.
910004	DISCOVER NOT ALLOWED	Discover card type transactions are not allowed.
910005	CARD TYPE NOT ALLOWED	Card type transactions are not allowed.
911911	SECURITY VIOLATION	M_id or M_key incorrect
920000	ITEM NOT FOUND	Item not found
920001	CREDIT VOL EXCEEDED	No corresponding sale found within last 6 months, credit couldn't be issued.
920002	AVS FAILURE	Address Verification Service failure.
920050	DEBIT VOID NOT ALLOWED	A debit transaction cannot be voided.