



**Service Sales Extract
Predefined Integration Package (PIP) v1.1
Product Guide**

Document Version 1.0

Service Sales Extract PIP	Product Version: 1.1
Product Guide	Date: 08/09/2010

Service Sales Extract Predefined Integration Package (PIP) v1.1 Product Guide

Product Version 1.1
Document Version 1.0
August 2010

**ADP, INC. • DEALER SERVICES GROUP / 1950 HASSELL ROAD / HOFFMAN ESTATES, IL 60169-6308
DEALERSUITE.COM**

© 2008-2010 ADP, Inc. – Dealer Services Group. The ADP logo is a registered trademark of ADP of North America, Inc. ADP disclaims proprietary interest in the marks and names of others.

Service Sales Extract PIP	Product Version: 1.1
Product Guide	Date: 08/09/2010

Table of Contents

1. Introduction	4
1.1 Overview	4
1.2 Definitions, Terms and Abbreviations	4
1.3 ADP Third-Party Integration Program	4
1.4 Product Perspective	5
2. Service Sales Extract Request	6
3. Field selection and Query definition	6
3.1 Extract Criteria	6
3.2 Service Sales Extract Data Elements	6
3.2.1 Service Sales (Header)	7
3.2.2 Service Sales Detail	8
3.2.3 Technician Hours	9
3.2.4 Parts	10
3.2.5 Miscellaneous, Lube and Sublet	10
3.2.6 STORY	11
3.3 Examples	13
3.3.1 Request Example 1	13
3.3.2 Request Example 2	14
3.3.3 Response Example	17

Service Sales Extract PIP	Product Version: 1.1
Product Guide	Date: 08/09/2010

1. Introduction

1.1 Overview

The Service Sales Extract PIP (Predefined Integration Package) is an integration product offering that enables third-party vendors to integrate directly and securely with the Service Application within the ADP DMS. The Service Sales Extract PIP allows dealership client users to extract Repair Order data from the DMS.

1.2 Definitions, Terms and Abbreviations

Term	Definition
3PA	ADP Third Party Access
DMS	Dealer Management System
PIP	Predefined Integration Package

1.3 ADP Third-Party Integration Program

The ADP Predefined Integration Package (PIP) product is part of ADP's Third-Party Integration Program, which is a web services-based framework that enables third-party software providers to leverage pre-defined integration calls to quickly and seamlessly interface their Internet-, Windows® - or UNIX-based applications and systems to ADP's w.e.b.Suite and Drive DMS applications. This framework:

- Manages and controls a dealer's third-party vendors' access to the DMS using Pre-Defined Integration Points (PIP). These PIPs allow for both read and write transfer of data between the third-party application and the DMS in a controlled and defined manner
- Delivers data on a real-time basis in the format designated by the data integration product
- Monitors and troubleshoots all integration processes
- Provides the third party with an online tool that gives them information on all dealers enrolled in their program and outstanding issues or communications between ADP and the Dealer
- Provides the dealer, upon request, information on who is receiving data, the data elements currently being delivered, and how often their DMS is accessed

ADP currently has many third-party application providers participating in and enjoying the benefits of ADP-approved bi-directional integration, which results in the following benefits for dealership customers:

- Superior input of data which provides easier system management and an additional level of information protection
- Elimination of double data entry
- Support for authorized vendors by ADP Dealer Services
- Reduction of problems dealing with Third-Party "hostile" interfaces

Service Sales Extract PIP	Product Version: 1.1
Product Guide	Date: 08/09/2010

1.4 Product Perspective

Figure 1-1 illustrates the logical elements envisioned for a PIP, and their relationships to each other. The intent of the diagram is not to illustrate discrete development components, but rather high-level logical components and relationships. It is important to note that, while the diagram illustrates components that reside with the Vendor Partner and ADP realms, this document will not focus on all three. Instead, this document will focus on messages between the Partner Web Processor and the ADP PIP Processor. The remaining components and functionality illustrated are provided for reference purposes.

The logical components pictured in **Figure 1-1** are:

Vendor's Application – The ADP client or vendor partner develops and maintains an application used by their clientele. The vendor stores a mapping of individual dealerships to a 3PA dealer id, which is used to identify a specific rooftop dealership and properly route the request to the appropriate data store. The application transmits HTTP Post requests to the 3PA PIP Processor and receives responses from the 3PA PIP Processor.

3PA PIP– This component acts as an intermediary between the Partner applications and the ADP DMS applications. It processes Partner requests, locates ADP dealers, and translates the requests to the DMS applications. It receives responses from the DMS applications, which are processed and forwarded to the Partner, and handles PIP processing exceptions.

ADP Application– This component represents an ADP application and its underlying data store containing a dealership's data. It processes data requests, queries the data store for information, handles error generation and response transmission.

The logical components perform several key functions that are critical to the request/response cycle. The functionality depicted in **Figure 1-1** is described below. These functions are not meant to describe the entire range of functionality for the PIP system. Rather, they represent the steps absolutely necessary to process each request.

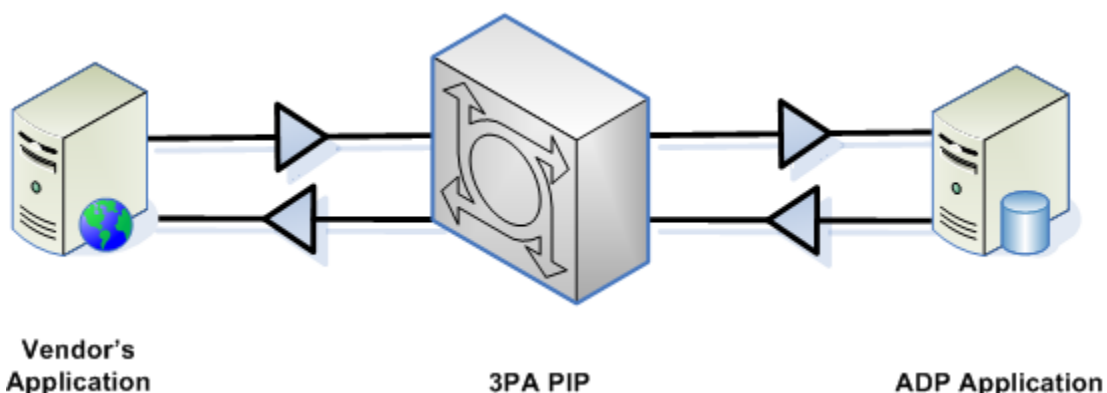


Figure 1-1: ADP PIP Product Perspective

Service Sales Extract PIP	Product Version: 1.1
Product Guide	Date: 08/09/2010

2. Service Sales Extract Request

Each request made of the Extract PIP web service must include the following data components. An example HTTP Post can be seen in the following section.

Field Name	Type	Length	Required	Field Description
User Name	Text	50	Y	Requesting party's user name.
Password	Text	50	Y	Requesting party's password.
DealerID	Text	25	Y	ADP 3PA's identifier for the dealer.
QueryID	Text	25	Y	ADP 3PA's identifier for the saved query.
DeltaDate	Date	N/A	N	Required if query type is Delta. MM/DD/YYYY
Parameters	Text	25	N	Depending on the predefined query, specific parameters may also be passed to further define the extract criteria. All parameters must have a prefix: qparam.

3. Field selection and Query definition

During the development and setup phase of integration, a vendor specifies the fields that they want to include in the extract and the criteria that should be used to extract the data. The query statements are developed for the vendor by an ADP Integration Analyst so that the vendor does not have to learn a new query language or worry about the names of internal data elements within the DMS. Each query is named and provided to the vendor so that they may call the query by name in the API. The queries are tested with the vendor to ensure that the extract is pulling the correct fields using the correct criteria. The data extract output is provided to the vendor application in XML format.

3.1 Extract Criteria

The following Extract types are available for the Service Sales Extract PIP.

- Bulk query- This query type pulls all available records from the Service file on the DMS.
- Delta query- This query pulls all records changed after a specified date. This query must always be accompanied by a parameter of the DeltaDate.
- Adhoc fixed selection query- This query type is used to further limit the data provided back in the extract. An example of an adhoc Service Sales selection statement would be to return all records with "closed date" not null.
- Adhoc variable selection query- Variable parameters can be included in these queries. An example of a variable adhoc Service Sales selection statement would be to return all records with "op code" equal to "qparamX". Each time the query is sent, a value for "qparamX" would need to be provided in the request.

3.2 Service Sales Extract Data Elements

Each repair order is comprised of 5 distinct elements: Service Sales, Service Sales Detail, Technician Hours, Parts and Miscellaneous. Each element can be selected based on the criteria of open or closed invoices. To select fields for open invoices, for example, the ServiceSalesOpen element would be called. In contrast ServiceSalesClosed would be called for closed invoices. Each member of the Open/Closed pair contain the same fields. The Technician Story is another element which can be added to the RO.

The following field lists are available as separate extracts from the DMS.

Service Sales Extract PIP	Product Version: 1.1
Product Guide	Date: 08/09/2010

3.2.1 Service Sales (Header)

Field Name	Type	Length	Description
HostItemID	Key	40	Host Item ID.
AccountingAccount	Text	17	Associated Accounting Account
ActualHours	Numeric (2)	8	Sum of actual hours on each labor operation.
Address	Text	35	Address
ApptDate	Date	8	Date of appointment if RO was rolled.
ApptFlag	Text	1	Flag indicating if RO was rolled from an appointment.
ApptTime	Time	8	Time of appointment if RO was rolled.
CashierNo	Text	17	User ID used when logging onto the DMS.
CityStateZip	Text	40	City, state abbreviation, and ZIP code
CloseDate	Date	8	Closed date of repair order.
ComebackFlag	Text	1	Comeback repair as determined by the SA.
ContactEmailAddress	Text	50	E-mail given during writeup in which to contact the customer
ContactPhoneNumber	Numeric (0)	10	Phone given during writeup in which to contact the customer
CustNo	Text	17	Customer number
EstCompletionDate	Date	8	Estimated completion date (Used by dispatching)
EstCompletionTime	Time	8	Estimated completion time (Used by dispatching)
LaborSale	Numeric (2)	10	Labor sale amount
LastServiceDate	Date	8	Last Service Date of Vehicle.
Make	Text	10	Make abbreviation for the vehicle
Mileage	Numeric (0)	7	Mileage-in entered in RO.
MileageLastVisit	Numeric (0)	7	Mileage entered from last visit.
MiscSale	Numeric (2)	10	Sum of all sale in MLS (miscellaneous, lube and sublet) for the repair order.
Model	Text	10	Model abbreviation for the vehicle
Name1	Text	45	Customer Name
Name2	Text	45	Customer name2
OpenDate	Date	8	Opened date of repair order.
OpenedTime	Time	8	Opened time of repair order.
OriginalPromiseDate	Date	8	Original Promised Date.
OriginalPromiseTime	Time	8	Original Promised Time.
OriginalWaiterFlag	Text	1	Original Customer wait for repairs flag.
PartsSale	Numeric (2)	10	Amount of parts sold
PostedDate	Date	8	Posted date of repair order.
PriorityFlag	Text	1	Flag indicating if Priority Value has been overridden.
PriorityValue	Text	5	Priority value used to determine when to dispatch the RO.
PromisedDate	Date	8	Promised date entered in RO
PromisedTime	Time	8	Promised time entered in RO
Rental	Text	1	Flag indicating if customer is using a rental vehicle.
RONumber	Text	12	RO Number.

Service Sales Extract PIP	Product Version: 1.1
Product Guide	Date: 08/09/2010

Field Name	Type	Length	Description
ROStatusCode	Text	5	Status of the RO.
ROStatusCodeDesc	Text	20	Status Code description for the repair order.
Service	Text	17	Service Logon that RO was created on.
ServiceAdvisor	Text	17	Service Advisor Number
SoldByDealer	Text	1	Flag indicating if the vehicle was bought from this dealer.
SourceFile	Text	1	Source file code
SpecialCust	Text	1	Flag indicating if customer needs special treatment.
Tag	Text	8	Tag number assigned to the vehicle while it is being worked on
VehID	Text	17	Vehicle ID on the RO.
VIN	Text	17	Vehicle identification number or serial number
VoidDate	Date	8	Voided date of repair order.
WaiterFlag	Text	1	Customer wait for repairs flag.
Year	Numeric (0)	4	Model year of the vehicle
LaborSaleCustomerPay	Numeric (2)	10	Customer pay labor sale amount
LaborSaleInternal	Numeric (2)	10	Internal labor sale amount
LaborSaleWarranty	Numeric (2)	10	Warranty labor sale amount
MiscSaleCustomerPay	Numeric (2)	10	Customer pay miscellaneous sale amount
MiscSaleInternal	Numeric (2)	10	Internal miscellaneous sale amount
MiscSaleWarranty	Numeric (2)	10	Warranty miscellaneous sale amount
PartsCostCustomerPay	Numeric (2)	10	Customer pay cost of parts sold for labor operations
PartsCostInternal	Numeric (2)	10	Internal cost of parts sold for labor operations
PartsCostWarranty	Numeric (2)	10	Warranty cost of parts sold for labor operations
ROComment1	Text	48	Repair Order Invoicing comments.
ROComment2	Text	48	Repair Order Invoicing comments.
ROComment3	Text	48	Repair Order Invoicing comments.
ROComment4	Text	48	Repair Order Invoicing comments.
ROComment5	Text	48	Repair Order Invoicing comments.
ROComment6	Text	48	Repair Order Invoicing comments.
ROComment7	Text	48	Repair Order Invoicing comments.
ROComment8	Text	48	Repair Order Invoicing comments.
ROComment9	Text	48	Repair Order Invoicing comments.
ROMiscSaleCP	Numeric (2)	10	Customer pay miscellaneous sale amount
ROMiscSaleIP	Numeric (2)	10	Internal miscellaneous sale amount
ROMiscSaleWP	Numeric (2)	10	Warranty miscellaneous sale amount

3.2.2 Service Sales Detail

Field Name	Type	Length	Description
HostItemID	Key	40	Host Item ID.
BookerNo	Text	17	The booker number for the LineCode.
CampaignCode	Text	20	Campaign code for the labor line.

Service Sales Extract PIP	Product Version: 1.1
Product Guide	Date: 08/09/2010

Field Name	Type	Length	Description
Causes	Text	55	Cause entered in Warranty window.
ComeBack	Text	1	Come back Y/N
ComplaintCode	Text	20	Complaint code number
DispatchCode	Text	6	Dispatch Code
LaborSale	Numeric (2)	10	Labor sale amount for the op code
LaborType	Text	5	Labor type
LineCode	Text	3	Line Code.
LopSeqNo	Text	3	Labor operation sequence number
MiscSale	Numeric (2)	10	Sale in MLS (miscellaneous, lube and sublet) for the line.
OpCode	Text	20	Op-code for the labor operation.
OpCodeDescription	Text	70	Original labor operation
PartsSale	Numeric (2)	10	Amount of parts sold on the op code
ErrorLevel	Text	1	ErrorLevelFlag
ErrorMessage	Text	80	Error Message
TechNo	Text	17	Technician number
TechNo2	Text	17	Technician number
TechNo3	Text	17	Technician number
TechNo4	Text	17	Technician number
TechNo5	Text	17	Technician number
TechNos	Text	17	Technician numbers

3.2.3 Technician Hours

Field Name	Type	Length	Description
HostItemID	Key	40	Host Item ID.
ErrorLevel	Text	1	ErrorLevelFlag
ErrorMessage	Text	80	Error Message
PickItemID	Text	40	ItemID from LABOR-OPS or HISTORY record.
MCDLaborPercentage	Numeric (0)	3	The labor split percentage.
RONumber	Text	12	RO Number.
VehID	Text	17	Vehicle ID on the RO.
LineCode	Text	3	Line Code.
LopSeqNo	Numeric (0)	3	Labor operation sequence number
ActualHours	Numeric (2)	8	Technician actual hours.
TimeCardHours	Numeric (2)	8	Labor operation sequence number
SoldHours	Numeric (2)	8	Technician Sold Hours.
LaborCost	Numeric (2)	10	Technician Labor Cost.
LaborSale	Numeric (2)	10	Technician Labor Sale.
OtherHours	Numeric (2)	8	Technician Other Hours.
LaborType	Text	5	Labor Type (customer, internal or warranty pay)
TechNo	Text	17	Technician number booked to the labor operation.

Service Sales Extract PIP	Product Version: 1.1
Product Guide	Date: 08/09/2010

3.2.4 Parts

Field Name	Type	Length	Description
HostItemID	Key	40	Host Item ID.
ErrorLevel	Text	1	ErrorLevelFlag
ErrorMessage	Text	80	Error Message
List	Numeric (2)	10	Escalated list price of the part.
Description	Text	25	Description of the part.
Employee	Text	17	Counterperson's employee number
PartsCompLineCode	Text	10	Parts comp code.
Source	Text	4	Source number of the part.
SpecStatus	Text	2	Special Status of the part.
CoreSale	Numeric (2)	10	Unit core sale charge on the part.
Class	Text	12	Class of the part number.
UnitServiceCharge	Numeric (2)	10	Service Charge entered in PRO for a returned part.
OutsideSalesman	Text	17	Outside salesman assigned to this customer.
PartNumber	Text	30	Part number.
PartsSaleCompany	Text	3	Parts sale company.
PartsSaleJournal	Text	3	Parts sale journal.
PartsCoreCost	Numeric (2)	10	Parts core cost.
PartsCoreAccount	Text	10	Parts core account.
PartsExtendedCost	Numeric (2)	10	Calculated total parts cost amount.
PartsExtendedSale	Numeric (2)	10	Calculated total parts sale amount.
MCDPartsPercentage	Numeric (0)	3	Multiple charge distribution percentage
PartsSaleAccount	Text	10	Parts Sale Account.
PickItemID	Text	40	ItemID from PARTS or MCD-PARTS records
LaborType	Text	7	Labor Type (customer, internal or warranty pay)
LopSeqNo	Text	3	Labor operation sequence number
ReferNo	Text	12	RONumber.
VehID	Text	17	Vehicle ID on the RO.
QuantityOrdered	Numeric (0)	6	Quantity Ordered in PRO.
Bin1	Text	8	Bin 1 of the part.
QuantitySold	Numeric (0)	6	Quantity Sold in PRO.
Cost	Numeric (2)	10	Unit cost price of the part.
SalePrice	Numeric (2)	10	Unit sale price of the part.
Comp	Numeric (2)	10	Unit comp price of the part.
MiscCost	Numeric (2)	10	Miscellaneous cost (Parts Fee).
MiscSale	Numeric (2)	10	Miscellaneous sale (Parts Fee).

3.2.5 Miscellaneous, Lube and Sublet

Field Name	Type	Length	Description
------------	------	--------	-------------

Service Sales Extract PIP	Product Version: 1.1
Product Guide	Date: 08/09/2010

Field Name	Type	Length	Description
HostItemID	Key	40	Host Item ID.
LaborType	Text	5	Labor Type (customer, internal or warranty pay)
OpCode	Text	20	Operation code
OpCodeDescription	Text	70	Description of operation code
FailureCode	Text	12	Failure Code entered in PFC's ASL window.
MiscCost	Numeric (2)	10	Cost entered in the PFC ASL window.
MiscSale	Numeric (2)	10	Sale Price entered in the PFC's ASL window.
SaleCompany	Text	3	Sale Company for the MLS charge.
DebitCompany	Text	3	Debit Company for the MLS charge.
SourceCompany	Text	3	Source Company for the MLS charge.
MLSType	Text	1	M=Miscellaneous, L=Lube, S=Sublet
SaleAccount	Text	10	Sale Account entered in the PFC's ASL window.
PickItemID	Text	40	ItemID from RO-MLS, MCD-LOPS, or HISTORY records
RONumber	Text	12	Repair order number
VehID	Text	17	Vehicle ID on the RO.
SaleControl	Text	17	Sale Control number
ErrorLevel	Text	1	ErrorLevelFlag
ErrorMessage	Text	80	Error Message
DebitAccount	Text	10	Debit Account entered in the PFC's ASL window.
DebitControl	Text	17	Debit control number
Journal	Text	3	Journal Number entered in the PFC's ASL window.
LineCode	Text	2	Line Code.
PONumber	Text	10	Sublet purchase order number.
POCreateDate	Date	10	Sublet purchase order create date.
FeeID	Text	12	Item ID of fee used to generate this fee.
POLineID	Text	12	PO line ID assigned to the PO line of the purchase order.

3.2.6 STORY

Field Name	Type	Length	Description
------------	------	--------	-------------

Service Sales Extract PIP	Product Version: 1.1
Product Guide	Date: 08/09/2010

Field Name	Type	Length	Description
HostItemID	Key	40	PICK file item-id
ErrorLevel	Text	1	ErrorLevel Flag
ErrorMessage	Text	80	Error Message
LastUpdatedByNumber	Text	17	Employee number who last updated this story
StoryText	Key	80	Text of the technician's story
UpdateDate	Date	8	Date the story was last updated

Service Sales Extract PIP	Product Version: 1.1
Product Guide	Date: 08/09/2010

3.3 Examples

The following examples are intended to show how the integration could be designed. The actual URLs would be different and will be provided with the vendor's authentication tokens.

3.3.1 Request Example 1

This is an example HTTP Post in Java.

```
public static void main(String[] args) {

    HttpClient cl = new HttpClient();
    PostMethod postMethod = new PostMethod("http://localhost/pip-service-extract/test");
    //PostMethod postMethod = new PostMethod("http://exampleurl.com/pip-service-extract/test");

    NameValuePair[] params = {
        new NameValuePair("dealerId", "3PADEALERID"),
        new NameValuePair("queryId", "QUERYID123"),
        new NameValuePair("qparamX", "4500"),
        new NameValuePair("deltaDate", "09/23/2009")
    };

    postMethod.setRequestBody(params);

    cl.getParams().setAuthenticationPreemptive(true);
    Credentials defaultCreds = new UsernamePasswordCredentials("USERNAME", "PASSWORD");
    cl.getState().setCredentials(AuthScope.ANY, defaultCreds);

    try {
        try {
            cl.executeMethod(postMethod);
        } catch (IOException e) {
            e.printStackTrace(); //To change body of catch statement use File | Settings | File Templates.
        }

        BufferedReader br;
```

Service Sales Extract PIP	Product Version: 1.1
Product Guide	Date: 08/09/2010

```

try {
    System.out.println(postMethod.getStatusCode());
    System.out.println(postMethod.getStatusText());

    br = new BufferedReader(new InputStreamReader(postMethod.getResponseBodyAsStream()),256);
    String line;
    char[] chars = new char[256];
    while (br.read(chars) != -1) {
        System.out.println(chars);
        chars = new char[256];
    }

} catch (IOException e) {
    e.printStackTrace(); //To change body of catch statement use File | Settings | File Templates.
}
} finally {
    postMethod.releaseConnection();
}

}

```

3.3.2 Request Example 2

This is an example HTTP Post in C# using .NET.

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Net;
using System.IO;

namespace ADP3PA
{
    public class VehicleExtract
    {
        public static void Test()
        {

```

Service Sales Extract PIP	Product Version: 1.1
Product Guide	Date: 08/09/2010

```

DownloadFile("http://staging.dmotorworks.com/pip-extract/vehicle","dealerId=3PATEST&queryId=QUERY1");
}

#region -- DownloadFile(url, localFileName) Method --
public static string DownloadFile(string url, string postData)
{
    string resultData = "";

    WebRequest request = null;
    WebResponse response = null;
    Stream stream = null;

    try
    {
        //encode post data
        ASCIIEncoding encoding = new ASCIIEncoding();
        byte[] data = encoding.GetBytes(postData);

        request = HttpWebRequest.Create(url);
        request.Method = "POST";
        request.ContentLength = data.Length;
        request.ContentType = "application/x-www-form-urlencoded";

        string username = "user";
        string password = "pass";
        CredentialCache mycache = new CredentialCache();
        mycache.Add(new Uri(url), "Basic", new NetworkCredential(username, password));

        request.Credentials = mycache;

        string usernamePassword = username + ":" + password;
        request.Headers.Add("Authorization", "Basic " + Convert.ToBase64String(new ASCIIEncoding().GetBytes(usernamePassword)));

        Stream newStream = request.GetRequestStream();
        // Send the post data.

```

Service Sales Extract PIP	Product Version: 1.1
Product Guide	Date: 08/09/2010

```

        newStream.Write(data, 0, data.Length);
        newStream.Close();

        response = request.GetResponse();
        stream = response.GetResponseStream();

        StreamReader sr = new StreamReader(stream);
        resultData = sr.ReadToEnd();

        stream.Close();
        stream.Dispose();
        response.Close();
    }
    catch (System.Net.WebException ex)
    {
        if (ex.Response != null)
        {
            Stream errorStream = ex.Response.GetResponseStream();

            StreamReader sr = new StreamReader(errorStream);
            string errordata = sr.ReadToEnd();
        }

        if (stream != null)
        {
            stream.Dispose();
        }

        throw;
    }
    catch
    {
        if (stream != null)
        {
            stream.Dispose();
        }
    }

```


Service Sales Extract PIP	Product Version: 1.1
Product Guide	Date: 08/09/2010

```

        throw;
    }

    return resultData;
}
#endregion
}
}

```

3.3.3 Response Example

3.3.3.1 Service Sales (Header) Response

```

<ServiceSales xmlns="http://www.dmotorworks.com/pip-extract-servicesales"><ServiceSalesOpen>
  <HostItemID>Y2299747*118157</HostItemID>
  <AccountingAccount>ABC-A</AccountingAccount>
  <ActualHours>0.00</ActualHours>
  <Address>4146 BRAMMER DR</Address>
  <ApptDate />
  <ApptFlag>1</ApptFlag>
  <ApptTime />
  <CashierNo />
  <CityStateZip>NEWPORT, DE 19884</CityStateZip>
  <CloseDate />
  <ComebackFlag>Y</ComebackFlag>
  <ContactEmailAddress />
  <ContactPhoneNumber />
  <CustNo>25988</CustNo>
  <EstCompletionDate>2010-01-06</EstCompletionDate>
  <EstCompletionTime>11:00:00</EstCompletionTime>
  <LaborSale>0.00</LaborSale>
  <LastServiceDate>2003-04-17</LastServiceDate>
  <Make>OLDS</Make>
  <Mileage>30714</Mileage>
  <MileageLastVisit>30924</MileageLastVisit>
  <MiscSale />
  <Model>BRAV</Model>

```

Service Sales Extract PIP	Product Version: 1.1
Product Guide	Date: 08/09/2010

```

<Name1>WYSOCKI, THOMAS M</Name1>
<Name2>THOMAS M WYSOCKI</Name2>
<OpenDate>2009-12-22</OpenDate>
<OpenedTime>08:26:17</OpenedTime>
<OriginalPromiseDate>2003-04-22</OriginalPromiseDate>
<OriginalPromiseTime>17:00:00</OriginalPromiseTime>
<OriginalWaiterFlag>N</OriginalWaiterFlag>
<PartsSale>40.20</PartsSale>
<PostedDate />
<PriorityFlag>1</PriorityFlag>
<PriorityValue>9999</PriorityValue>
<PromisedDate>2003-05-23</PromisedDate>
<PromisedTime>17:00:00</PromisedTime>
<Rental>1</Rental>
<RONumber>118157</RONumber>
<ROStatusCode>I98</ROStatusCode>
<ROStatusCodeDesc>WORKING</ROStatusCodeDesc>
<Service>DMI3-S</Service>
<ServiceAdvisor>536</ServiceAdvisor>
<SoldByDealer>1</SoldByDealer>
<SourceFile>L</SourceFile>
<SpecialCust />
<Tag>T4669</Tag>
<VehID>Y2299747</VehID>
<VIN>1GHDT13WXY2299747</VIN>
<VoidDate />
<WaiterFlag>N</WaiterFlag>
<Year>2000</Year>
<ErrorLevel>0</ErrorLevel>
<ErrorMessage /></ServiceSalesOpen><ServiceSalesOpen><ErrorCode>0</ErrorCode><ErrorMessage/></ServiceSales>

```

3.3.3.2 Service Sales Detail Response

```

<ServiceSales xmlns="http://www.dmotorworks.com/pip-extract-servicesales"><ServiceSalesDetailsOpen>
  <HostItemID>Y2299747*118157*1*0</HostItemID>
  <BookerNo />
  <CampaignCode />

```

Service Sales Extract PIP	Product Version: 1.1
Product Guide	Date: 08/09/2010

```

<Causes />
<ComeBack>0</ComeBack>
<ComplaintCode>40</ComplaintCode>
<DispatchCode>GS140</DispatchCode>
<LaborSale>0.00</LaborSale>
<LaborType>CPR3</LaborType>
<LineCode>A</LineCode>
<LopSeqNo>1</LopSeqNo>
<MiscSale>0.00</MiscSale>
<OpCode>40</OpCode>
<OpCodeDescription>GM TRANSAXLE REPAIR</OpCodeDescription>
<PartsSale>40.20</PartsSale>
<ErrorLevel>0</ErrorLevel>
<ErrorMessage />
<TechNo>263</TechNo>
<TechNo2 />
<TechNo3 />
<TechNo4 />
<TechNo5 />
<TechNos>263</TechNos>
</ServiceSalesDetailsOpen><ServiceSalesDetailsOpen>
<HostItemID>Y2299747*118157*2*0</HostItemID>
<BookerNo />
<CampaignCode />
<Causes />
<ComeBack>0</ComeBack>
<ComplaintCode>40</ComplaintCode>
<DispatchCode>GS140</DispatchCode>
<LaborSale>0.00</LaborSale>
<LaborType>CPR3</LaborType>
<LineCode>B</LineCode>
<LopSeqNo>2</LopSeqNo>
<MiscSale>0.00</MiscSale>
<OpCode>40</OpCode>
<OpCodeDescription>GM TRANSAXLE REPAIR</OpCodeDescription>
<PartsSale>0.00</PartsSale>
<ErrorLevel>0</ErrorLevel>
<ErrorMessage />
<TechNo>263</TechNo>
<TechNo2 />
<TechNo3 />
<TechNo4 />
<TechNo5 />
<TechNos>263</TechNos>
</ServiceSalesDetailsOpen><ErrorCode>0</ErrorCode><ErrorMessage/></ServiceSales>

```

Service Sales Extract PIP	Product Version: 1.1
Product Guide	Date: 08/09/2010

3.3.3.3 Parts Lines Response

```

<ServiceSales xmlns="http://www.dmotorworks.com/pip-extract-servicesales"><ServiceSalesPartsOpen>
  <HostItemID>XB114757*117349*1*0</HostItemID>
  <ErrorLevel>0</ErrorLevel>
  <ErrorMessage />
  <List>367.06</List>
  <Description>W-MODULE</Description>
  <Employee>660</Employee>
  <PartsCompLineCode />
  <Source>102</Source>
  <SpecStatus>NS</SpecStatus>
  <CoreSale />
  <Class>3107</Class>
  <UnitServiceCharge />
  <OutsideSalesman />
  <PartNumber>25320538</PartNumber>
  <PartsSaleCompany>55</PartsSaleCompany>
  <PartsSaleJournal>31</PartsSaleJournal>
  <PartsCoreCost />
  <PartsCoreAccount>46700</PartsCoreAccount>
  <PartsExtendedCost>0.00</PartsExtendedCost>
  <PartsExtendedSale>0.00</PartsExtendedSale>
  <MCDPartsPercentage>100</MCDPartsPercentage>
  <PartsSaleAccount>46700</PartsSaleAccount>
  <PickItemID>117349*1</PickItemID>
  <LaborType>CPE</LaborType>
  <LopSeqNo>1</LopSeqNo>
  <ReferNo>117349</ReferNo>
  <VehID>XB114757</VehID>
  <QuantityOrdered>0</QuantityOrdered>
  <Bin1 />
  <QuantitySold>0</QuantitySold>
  <Cost>191.64</Cost>
  <SalePrice>210.81</SalePrice>
  <Comp />
</ServiceSalesPartsOpen><ServiceSalesPartsOpen>
  <HostItemID>XB114757*117349*2*0</HostItemID>
  <ErrorLevel>0</ErrorLevel>
  <ErrorMessage />
  <List>550.56</List>
  <Description>W-MODULE</Description>
  <Employee>25</Employee>
  <PartsCompLineCode />
  <Source>102</Source>
  <SpecStatus>NS</SpecStatus>

```

Service Sales Extract PIP	Product Version: 1.1
Product Guide	Date: 08/09/2010

```

<CoreSale />
<Class>3107</Class>
<UnitServiceCharge />
<OutsideSalesman />
<PartNumber>25178736</PartNumber>
<PartsSaleCompany>55</PartsSaleCompany>
<PartsSaleJournal>31</PartsSaleJournal>
<PartsCoreCost />
<PartsCoreAccount>46700</PartsCoreAccount>
<PartsExtendedCost>311.93</PartsExtendedCost>
<PartsExtendedSale>343.13</PartsExtendedSale>
<MCDPartsPercentage>100</MCDPartsPercentage>
<PartsSaleAccount>46700</PartsSaleAccount>
<PickItemID>117349*2</PickItemID>
<LaborType>CPE</LaborType>
<LopSeqNo>1</LopSeqNo>
<ReferNo>117349</ReferNo>
<VehID>XB114757</VehID>
<QuantityOrdered>1</QuantityOrdered>
<Bin1 />
<QuantitySold>1</QuantitySold>
<Cost>311.93</Cost>
<SalePrice>343.13</SalePrice>
<Comp />
</ServiceSalesPartsOpen><ServiceSalesPartsOpen>
<HostItemID>XB114757*117349*3*0</HostItemID>
<ErrorLevel>0</ErrorLevel>
<ErrorMessage />
<List>3.99</List>
<Description>FILTER</Description>
<Employee>660</Employee>
<PartsCompLineCode />
<Source>108</Source>
<SpecStatus />
<CoreSale />
<Class>1836</Class>
<UnitServiceCharge />
<OutsideSalesman />
<PartNumber>25171377</PartNumber>
<PartsSaleCompany>55</PartsSaleCompany>
<PartsSaleJournal>31</PartsSaleJournal>
<PartsCoreCost />
<PartsCoreAccount>46700</PartsCoreAccount>
<PartsExtendedCost>3.18</PartsExtendedCost>
<PartsExtendedSale>3.50</PartsExtendedSale>

```

Service Sales Extract PIP	Product Version: 1.1
Product Guide	Date: 08/09/2010

```

<MCDPartsPercentage>100</MCDPartsPercentage>
<PartsSaleAccount>46700</PartsSaleAccount>
<PickItemID>117349*3</PickItemID>
<LaborType>CPE</LaborType>
<LopSeqNo>1</LopSeqNo>
<ReferNo>117349</ReferNo>
<VehID>XB114757</VehID>
<QuantityOrdered>1</QuantityOrdered>
<Bin1>G105DFM</Bin1>
<QuantitySold>1</QuantitySold>
<Cost>3.18</Cost>
<SalePrice>3.50</SalePrice>
<Comp />
</ServiceSalesPartsOpen> <ErrorCode>0</ErrorCode><ErrorMessage/></ServiceSales>

```