Real-world Business Intelligence with Microsoft Dynamics GP

Become an expert at preparing reports using Dynamics GP quickly and efficiently

Belinda Allen  Mark Polino
In this package, you will find:

- The authors biography
- A preview chapter from the book, Chapter 8 'Business Intelligence for Inventory Control'
- A synopsis of the book’s content
- More information on Real-world Business Intelligence with Microsoft Dynamics GP

About the Authors

**Belinda Allen** is a Microsoft Most Valuable Professional (MVP), certified professional for Dynamics GP (MCP), and Microsoft Certified Trainer (MCT). She is also one of the cofounders of Smith & Allen Consulting, Inc. (www.saci.com), a firm with over 22 years' experience specializing in business intelligence, analytics, and ERP software.

Belinda began implementing ERP systems so long ago that Windows was not an operating system but an application. At that time, larger businesses used mainframes with monitors that projected a green type on black backgrounds, and smaller businesses did their accounting by hand. Having seen the evolution that has taken place over the years from sheets of paper to integrated analytics, Belinda still gets excited every time she helps a business improve.

Belinda is also well known for her blog, www.BelindaTheGPcsi.com. On her blog, she shares really useful information about the product, quickly and succinctly. She has earned the nickname GP CSI because she excels at reviewing GP problems and figuring out what went wrong… and why. With followers from all over the globe, she is able to share knowledge and achieve her mission: "to improve the lives and business success of my followers."

When not delving into GP problems and spearheading business success for clients, Belinda enjoys sailing, crochet/knitting, sewing/quilting, reading, woodturning, and playing the Ukulele (often participating in Ukulele jams in NYC).
Mark Polino is a Microsoft MVP for Dynamics GP, certified public accountant, and Microsoft MVP. He also holds additional accounting certifications. Mark is the author of the premier Dynamics GP-related blog, www.DynamicAccounting.net, and the creator and presenter of the successful series *Getting More Out of Microsoft Dynamics GP: 50 Tips in 50 Minutes*. He has worked with Dynamics GP and its predecessor, Great Plains, for more than 15 years. Mark is currently a senior consultant with Integrated Business Group, implementing Dynamics GP. He is the author or coauthor of four other Dynamics GP books by Packt Publishing. He is also the author of a novel.
Real-world Business Intelligence with Microsoft Dynamics GP

Do you keep hearing the term "Business Intelligence" (BI) at conferences, on websites and social media, and from Microsoft? Do you think it's something you need or want, but you don't know where to start? If your answer to any of these questions is "yes" or you want to improve your current BI methodology, then this book is for you. We are going to share with you some real-world solutions for GP users, build solutions, and then edit them to fit your specific needs.

**Question**

Baseball, Jason Bourne, and James Bond—what do these three things have to do with Microsoft Dynamics GP or any accounting or enterprise resource planning (ERP) system?

**Answer**

Intelligence! I know you are probably saying, "What?" So let me explain.

**Baseball**

Since the formation of the National League (United States professional baseball) in 1876, statistics for all major league players and teams have been kept. These statistics have been used to determine how well a player is performing and how well a team is performing. In 1977, author Bill James began writing baseball statistics books, developing these statistics into a methodology he called sabermetrics. He defined sabermetrics as "the search for objective knowledge about baseball."

For more information on books by Bill James published by ACTA Sports (www.actasports.com), see his website at www.baseballinfosolutions.com.

In 2002, Oakland A's Billy Beane and Paul DePodesta took the A's to the playoffs (despite the team having the third lowest salary in the league, and therefore, no star players) using sabermetrics. Until that point, baseball stats were used only to analyze a player's performance and not to determine whom to sign as players. This seems like a "duh!" moment now, but in 2002, it was radical. The success of the Oakland As has inspired many teams to hire full-time sabermetrics analysts. This fascinating story is captured in a wonderfully written book by Michael Lewis titled Moneyball, The Art of Winning an Unfair Game. The book was turned into a full-length feature film in 2011. If you haven't seen it, watch it. If you have seen it, watch it again with the perspective of someone who is working to achieve BI.
So, obtaining accurate and timely baseball statistics makes way for the ability to hire players who can perform in the areas in which the team needs improved performance. This information allows more wins and possibly the ultimate baseball goal: winning the World Series.

**Jason Bourne**

James Bourne is a fictional character created by author Robert Ludlum who works (or worked) for the Central Intelligence Agency (CIA). First, I would like to address the intelligence of Mr. Bourne, and then I'll address the intelligence of the CIA.

In the first *Bourne Trilogy* book, *The Bourne Identity*, Mr. Bourne is fighting for his life, while fighting amnesia at the same time. Although he cannot remember who he is, he does recall how to assess situations. For some, this may be a natural trait, but for most, including Mr. Bourne, it was a learned trait. This quick assessment of the situation (for example, who may be watching me? Who may be looking for me? Who may have a gun? Who may be dangerous? and so on) is a powerful example of using *timely* intelligence. For Mr. Bourne, "real-time data analysis" is critical because he has no past information on which to rely.

Because of movies and characters such as Jason Bourne, we often think of the CIA as an organization that does nothing but spying. The CIA has three roles: collecting information (intelligence), analyzing the information they collect, and then performing (covert) operations based on this information if the President of the United States directs them to do so. Note that information, or intelligence, is a major part of what the CIA does and is.


**James Bond**

Ian Fleming's character, James Bond, has infiltrated the lives of almost every movie goer and TV movie watcher at some point or another. We all know that Mr. Bond is a member of Britain's MI6 (Military Intelligence, Section 6). MI6 (the Secret Intelligence Service or SIS) is responsible for the United Kingdom's national security and economic well-being.
Now, I am not bringing this up simply because Mr. Bond is in an intelligence agency; I am bringing it up because it is intelligence that keeps Mr. Bond alive (along with his skills in weaponry, piloting, swimming, driving, charming women, and hand-to-hand combat; well, you get the point). Think about it: in the first part of each story, Mr. Bond meets Q to obtain "spy gadgets" that will assist him with his mission. At some point in the story, Q's gadgets always end up getting Mr. Bond out of trouble and/or saving his life, regardless of whether it is a belt jet pack, a Lotus Esprit that has submarine capabilities, or even a digital watch that doubles as a phone. Q always knew exactly what each situation would require by understanding the fundamentals surrounding Mr. Bond's mission.

Ian Fleming worked for Britain's Naval Intelligence Division during WWII, overseeing many operations, including Operation Golden Eye. He used the knowledge obtained during his service as background for his James Bond novels. So, Fleming used his intelligence to set the groundwork.

Basically, obtaining accurate and timely information helps Q provide James Bond with just the right gadget for the operation. It also allows MI6 to maintain the national security and economic welfare of the United Kingdom.


**Baseball, Jason Bourne, and James Bond versus. ERP BI**

So now that we've spent a bit of time discussing baseball, Jason Bourne and James Bond, let me explain why these are great examples of business intelligence for your accounting and/or ERP system.

**Past performance**

Baseball sabermetrics uses a player's or team's performance based on everything they have done. This is similar to our financial statements; for example, how did we do last month compared to the budget, the previous month, or even the year before?

**Present performance**

Jason Bourne was an expert at reviewing his surroundings and protecting himself and his assets. Paying attention to what is happening in real time is critical to business success as well. This means entering information into your system every day so that data can be reviewed and captured. What good does it do for you to find out that you overspent 45 days ago?
Future performance

Q always had James Bond prepared by analyzing what should be necessary for an operation. This is similar to how businesses and organizations budget and forecast. Being prepared for the future has a direct relation to studying what is occurring and what has occurred. If we do not forecast or budget, how do we know whether we are improving as per criteria such as the economy and industry trends?

What This Book Covers

Chapter 1, *What Is BI and What Are BI Tools for Microsoft Dynamics GP?*, explains how BI should be viewed to help users determine what they need to see. In this chapter, we tackle the challenge of deciding what should be viewed by thinking about "problems/goals" and not "reports." I also briefly describe the various BI tools available with Microsoft Dynamics GP.

Chapter 2, *Business Intelligence for the General Ledger*, is all about using General Ledger data as our source of information. This chapter covers examples and guidance on building a balance sheet dashboard (changes in cash, AR, AP, inventory, and profit/loss) and the ratio of AP to AR.

Chapter 3, *Business Intelligence for Bank Reconciliation*, is more about managing cash flow than monitoring a bank balance. This chapter contains examples and guidance on:

- Building bank balances with last reconciliation dates
- Where the money goes

Chapter 4, *Business Intelligence for Payables Management*, explains how proper use of the payables management module combined with good BI can improve cash flow. This chapter covers examples and guidance on building unapplied credit/paid payments, invoices on hold and vendor holds, and shareable aged trial balance.

Chapter 5, *Business Intelligence for Receivables Management*, helps you know what is owed to you. Making a sale is great, but collecting on it is even more important. This chapter will contains examples and guidance on building a shareable aged trial balance and customers' sales in the current year compared to the last year.

Chapter 6, *Business Intelligence for Sales Order Processing*, covers the heart and soul of all businesses and organizations—revenue (or income). This chapter covers examples and guidance on building the following: a dashboard of sales by customer, state, item class, and salesperson; a dashboard of open orders by customer, state, item class, and salesperson; and items sold by the manufacturer.
Chapter 7, *More Business Intelligence on Sales Order Processing*, demonstrates order management, making fulfillment easier and improving customer service. This chapter contains examples and guidance on building the following: SOP documents on process holds, items shipped within the last 30 days with tracking numbers, and a returned items report.

Chapter 8, *Business Intelligence for Inventory Control*, is designed to help you manage what you have in your inventory in an attempt to keep your inventory stock at a minimum, while maintaining what you need. This chapter covers examples and guidance on building a rolling 12-month turnover report, and items on which you most frequently adjust quantities and transfers.

Chapter 9, *More Business Intelligence for Inventory Control*, is all about items that are returned. This chapter contains examples and guidance on building the top 10 most invoiced and returned items, items with quantities that are not available for sale, and items that have not been sold within the last year.

Chapter 10, *Business Intelligence for Purchase Order Processing*, includes both distribution purchase orders and non-inventory/distribution purchase orders. This chapter covers examples and guidance on building purchase orders that await approval and open purchase orders older than 45 days.

Chapter 11, *Miscellaneous Business Intelligence*, covers additional content, including information based on employees, projects, and fixed assets. This chapter contains examples and guidance on building commission details and an employee list with personal details by location.
This chapter is all about creating reports to make managing inventories easier. We've seen that companies with incorrect inventory on the books face a significant loss or gain in the accounting numbers when the inventory valuation is corrected. The impact is not only on taxes but on the appearance to lending institutions as well. They might believe that perhaps, your business is not being as well managed as it could be. Fortunately, you can avoid this type of situation by knowing when your inventory changes and why.

The name of this module is inventory control. It's meant to help you take control of your inventory. It is not called Inventory "Set It and Forget It". As inventory managers, control is critical, as both the Sales Order Processing module and the Purchase Order Processing module can affect your inventory with increases and decreases. If those users make an error, it affects your quantities and valuation.

There is no magic pill for managing inventory. It takes consistent and regular monitoring. When done right, it provides a wealth of information that can help you make decisions that result in more profitability.

We will build the following reports in this chapter:

- Rolling 12-month turnover report
- Items on which you most frequently adjust quantities
Rolling 12-month turnover report
We all understand that too much inventory can tie up cash. To help improve cash flow while keeping enough inventory to fulfil orders, many companies use a turnover report to see how much cash they have invested in their inventory. This report can also review items in stock to help answer many questions, including some of the following: Are these items selling? Are they selling regularly throughout the year or more seasonally? Should we try to discount them to clear them out of the inventory and generate cash flow? Are the items with expiration dates selling fast enough to prevent write-offs and returns?

Brief background of the company
This report was built for a distribution company that sells food items. Although all the food items they sell are prepackaged with a long shelf life, they want to keep the product on their shelves for the shortest period of time they can so that their customers can keep them for a longer period of time.

Pain or goal defined
Although GP has a nice turnover report right out-of-the-box, it only calculates YTD, so for the first couple of months of the year, the report is not as wide-ranging as they require. They would also like the warehouse manager to run this report, and currently he does not have access to GP.

BI and BI tool chosen
We will be using the PivotTable feature of Excel.

Step-by-step guide to build the BI content
There are two parts to these steps. The first part is technical, containing steps that involve work in SQL Server. The second part is the actual building of the report. You may choose to obtain assistance with the technical part, especially if you do not have access to SQL Server.

Technical matter (in case you're not a techie!)
Create the following view in SQL Server:
For directions on creating a view, follow the technical matters step for the first report in Chapter 2, *Business Intelligence for the General Ledger*.

Create View view_BI_Items_Qty_Tr

```
CREATE VIEW view_BI_Items_Qty_Tr
AS
SELECT dbo.IV00101.ITMCLSCD AS Item_Class,
       RTRIM(dbo.IV00101.ITEMNMBR) AS Item_Number,
       RTRIM(dbo.IV00101.ITEMDESC) AS Item_Desc,
       dbo.IV00102.QTYONHND AS Qty_On_Hand,
       dbo.IV00102.QTYINUSE + dbo.IV00102.QTYINSVC + dbo.IV00102.QTYRTRND +
       dbo.IV00102.QTYDMGED AS Qty_Not_Available,
       0 AS Qty_Purchased,
       0 AS Qty_Sold,
       0 AS Qty_Adjusted,
       0 AS Qty_Assembled,
       dbo.IV00102.LOCNCODE AS Site_ID,
       GETDATE() AS Trx_Date,
       'NA' AS Doc_Number,
       0 AS Line_Item_Seq
FROM dbo.IV00101
INNER JOIN dbo.IV00102 ON dbo.IV00101.ITEMNMBR = dbo.IV00102.ITEMNMBR
WHERE (dbo.IV00102.LOCNCODE <> ' ')
UNION
SELECT dbo.IV00101.ITMCLSCD AS Item_Class,
       RTRIM(dbo.IV00101.ITEMNMBR) AS Item_Number,
       RTRIM(dbo.IV00101.ITEMDESC) AS Item_Desc,
       0 AS Qty_On_Hand,
       0 AS Qty_Not_Available,
       CASE IV30300.DOCTYPE
           WHEN '4' THEN (IV30300.TRXQTY * IV30300.QTYBSUOM)
           ELSE 0
       END AS Qty_Purchased,
       CASE IV30300.DOCTYPE
           WHEN '5' THEN (IV30300.TRXQTY * IV30300.QTYBSUOM)
           WHEN '6' THEN (IV30300.TRXQTY * IV30300.QTYBSUOM)
           ELSE 0
       END AS Qty_Sold,
```
CASE IV30300.DOCTYPE
  WHEN '1' THEN (IV30300.TRXQTY * IV30300.QTYBSUOM)
  WHEN '2' THEN (IV30300.TRXQTY * IV30300.QTYBSUOM)
  ELSE 0
END AS Qty_Adjusted,
CASE IV30300.DOCTYPE
  WHEN '7' THEN (IV30300.TRXQTY * IV30300.QTYBSUOM)
  ELSE 0
END AS Qty_Assembled,
dbo.IV30300.TRXLOCTN AS Site_ID,
dbo.IV30300.DOCDATE AS Trx_Date,
dbo.IV30300.DOCNUMBR AS Doc_Number,
dbo.IV30300.LNSEQNBR AS Line_Item_Seq
FROM dbo.IV00101
INNER JOIN dbo.IV30300 ON dbo.IV00101.ITEMNMBR = dbo.IV30300.ITEMNMBR
WHERE (dbo.IV30300.DOCDATE > DATEADD(yy, - 1, GETDATE()))
GO
Grant select on view_BI_Items_Qty_Trx to DYNGRP

**Non-technical matter**

Let's build a trend report!

Guess what the first step is? If you guessed: "Know what you want", you are a winner! We’ve already completed this step, and have included a screenshot here:
Let's establish a connection to the GP data:

1. Open a Blank workbook in Excel. From the menu bar, choose DATA. In the Get External Data portion of the ribbon, choose From Other Sources. Then choose From SQL Server as the source, like this:

2. Enter the Server name and your network or AD login credentials. Click on Next:
Just a reminder: if you are building this for someone who does not have access to GP, refer to the first report in Chapter 2, Business Intelligence for the General Ledger, to see how to set up security.

3. Select your database from the drop-down list; select view_BI_Items_Qty_Trx. Click on Finish:

4. On the Import Data window, select PivotTable Report. Make sure that you are putting the data in the Existing worksheet option in the =A$1 cell. Then click on OK. The PivotTable canvas will appear.
Let’s build the foundation of our PivotTable report:

1. In the **PivotTable Fields** list on the right, drag the **Item_Number** and **Item_Desc** fields into the **ROWS** area, as follows:

![PivotTable Fields](image1.png)

2. Click anywhere on the PivotTable, and new menu options will appear, called **PIVOTTABLE TOOLS**. On the **Design** tab under **PIVOTTABLE TOOLS**, choose **Report Layout** in the **Layout** section of the ribbon. Select **Show in Tabular Form**:

![Report Layout](image2.png)
3. Again, click anywhere on the PivotTable, and new menu options will appear, called **PIVOTTABLE TOOLS**. From the **Design** tab under **PIVOTTABLE TOOLS**, choose **Subtotals** in the **Layout** section of the ribbon. Select **Do Not Show Subtotals**:

Let's add the custom calculations:

1. From the **ANALYZE** tab under **PIVOTTABLE TOOLS**, choose **Fields, Items & Sets** in the **Calculations** section of the ribbon. Select **Calculated Field**:

2. In the **Insert Calculated Field** window, in the **Name** field, enter **Current Inventory**. Populate the **Formula** field by selecting **Fields** and clicking on **Insert Field**. From the view we created earlier, we'll need the following: **Qty_On_Hand** and **Qty_Not_Available** (this is everything in use, in service, returned, and damaged in GP). We'll put these fields together, creating the following: **Formula: = Qty_On_Hand + Qty_Not_Available**. Click on **OK**.
3. You’ll notice that the new field we created is displayed as a value:
4. On the **Analyze** tab under **PIVOTTABLE TOOLS**, choose **Fields, Items & Sets** in the **Calculations** section of the ribbon. Select **Calculated Field**.

5. In the **Insert Calculated Field** window, in the **Name** field, enter **Rolling YR Sold**. Populate the **Formula** field by selecting fields from **Fields** and clicking on **Insert Field**. From the view we created earlier, we'll need **Qty_Sold**. This is everything sold in SOP less returns in GP. (By default, items sold are shown as a negative, while items returned are shown as a positive). We'll need to make the sold items positive by creating the following: **Formula**: = Qty_Sold * -1. Click on **OK**:

![Insert Calculated Field Window](image)

6. From the **Analyze** tab under **PIVOTTABLE TOOLS**, choose **Fields, Items & Sets** in the **Calculations** section of the ribbon. Select **Calculated Field**.

7. In the **Insert Calculated Field** window, in the **Name** field, enter **Rolling YR Purchases**. Populate the **Formula** field by selecting fields from the **Fields** area and clicking on **Insert Field**. The formula we want will create a column for the total amount added to the inventory, usually purchased. From the view we created earlier, we'll need the following: **Qty_Purchased** (this is everything purchased in Purchase Order Processing in GP), **Qty_Adjusted** (this is the net amount of all inventory transactions, both adjustments and variances in GP), and **Qty_Assembled** (this is the amount of items built using the bill of materials module in GP). We'll account for purchases by creating the following: **Formula**: = Qty_Purchased + Qty_Adjusted + Qty_Assembled. Click on **OK**.
8. On the ANALYZE tab under PIVOTTABLE TOOLS, choose Fields, Items & Sets in the Calculations section of the ribbon. Select Calculated Field.

9. In the Insert Calculated Field window, in the Name field, enter Beginning Inventory. Populate the Formula field by selecting fields from the Fields area and clicking on Insert Field. The formula we want will create a column for the inventory balance at the beginning of the rolling 12-month period (or exactly 1 year ago). We'll use the previous calculations in our formula to determine the beginning Inventory. Here is the formula: = 'Current Inventory' + 'Rolling YR Sold' - 'Rolling YR Purchases'. Click on OK:
10. In the Pivot Table Fields list on the right, re-sort the values by dragging and dropping them in the following order: Sum of Beginning Inventory, Sum of Rolling YR Purchases, Sum of Rolling YR Sold, and Sum of Current Inventory:

![Pivot Table Fields](image)

11. On the Analyze tab under PIVOTTABLE TOOLS, choose Fields, Items & Sets in the Calculations section of the ribbon. Select Calculated Field.

12. In the Insert Calculated Field window, in the Name field, enter Average Inventory YR. Populate the Formula field by selecting fields from the Fields area and clicking on Insert Field. The formula we want will create a column that averages the beginning and ending inventory balances. Here is the formula: \(= \frac{\text{('Current Inventory') + 'Beginning Inventory'}}{2}\). Click on OK:
Inventory systems will often create a turnover report based on the beginning and ending inventory of each item for each month being reported, and then create an average. With our approach, we do not have the ability to determine the beginning and inventory amounts for each period without complex programming. Therefore, we adopted the commonly used calculation of creating an average based on the beginning and ending inventory balance for the entire period. Using either calculation consistently will provide you with excellent results.

13. On the **ANALYZE** tab under **PIVOTTABLE TOOLS**, choose **Fields, Items & Sets** in the **Calculations** section of the ribbon. Select **Calculated Field**.

14. In the **Insert Calculated Field** window, in the **Name** field, enter **Turns**. Populate the **Formula** field by selecting fields from the **Fields** area and clicking on **Insert Field**. The formula we want will create a column that takes the amount sold and divides it by the average inventory. This will allow us to see what percent of the average inventory turned over for the reporting period. Here is the formula: 

\[ \frac{(\text{'Rolling YR Sold'})}{(\text{IF('Average Inventory YR','=',0,1,'Average Inventory YR'))}} \]

Click on **OK**:

In this formula, we are first determining whether the average inventory for the rolling 12-month period is 0. If it is 0, then we divide by 1 to prevent a calculation error.
Business Intelligence for Inventory Control

Let's format and finish!

1. In the PivotTable Fields list, in the VALUES area, click on the drop-down list for the Sum of Beginning_Balance field. Select Value Field Settings:

2. In the Value Field Settings window that opens, make sure that Sum is selected as the Summarize value field by option. Click on Number Format. The Format Cells window will open. Change Category to Number. Select how you want your values to be displayed. Then click on OK for each window to close it:
3. Repeat the formatting for all the fields in the value section of the PivotTable Fields list. We set all our columns to show the 1000 separator, and set the decimals to 0 for all columns except Turns; we left the decimals for this field at 2.

4. On the ANALYZE tab under PIVOTTABLE TOOLS, choose Insert Slicer in the Filter section of the ribbon:

5. In the Insert Slicer window, select the checkboxes for Item_Class and Site_ID. Click on OK:

You can pick whatever you want for the field (or fields) on which you want to filter this report. These are our preferences.
6. Highlight each slicer (this step must be done one slicer at a time), and the Excel ribbon will change to **SLICER TOOLS | OPTIONS**. Select Slicer Settings.

7. In the Slicer Settings window, mark **Hide items with no data**. This will keep the slicers smaller when actively using this report.

8. Another addition you may want to make is a **Timeline** filter on the report. The report will calculate for whatever time period you select, as long as the time period you select is within the previous 12-month period.

   This view extracts only data within the last 12 months and any transaction with a future date.

9. On the **ANALYZE** tab under **PIVOTTABLE TOOLS**, choose **Insert Timeline** in the **Filter** section of the ribbon:

10. In the **Insert Timeline** window, select the checkbox for **Trx_Date**. Click on **OK**. Note that this window will give you options for date fields only, and there is only one date field in this view:

11. Add a logo and any text you like. To refresh the data from GP, click on **Refresh All** under the **DATA** tab.
12. Now your report can be pulled using date filters, item class filters, and/or the inventory site location ID:

Let's have a little fun! Before we wrap up this report, let's have fun with it by making a few changes that might make it easier to read:

1. Let's leave everything as is, except that we'll eliminate all items that have 0 for the number of Turns.

   Click on the drop-down list for Item_Desc and select Value Filters and then Does Not Equal:
2. When the Value Filter (Item_Desc) window opens, change Show items for which to Sum of Turns, and set does not equal to 0. Click on OK to close the window:

3. Here's the result:

```
<table>
<thead>
<tr>
<th>Item_Desc</th>
<th>Sum of Beginning Inventory</th>
<th>Sum of Ending Inventory</th>
<th>Sum of Current Year</th>
<th>% Increase</th>
<th>Sum of 3-Year Average</th>
<th>% Increase</th>
<th>Sum of 3-Year Average</th>
<th>% Increase</th>
</tr>
</thead>
<tbody>
<tr>
<td>T-shirt - Red</td>
<td>12.00</td>
<td>10.00</td>
<td>11.00</td>
<td>-10.00</td>
<td>15.00</td>
<td>-10.00</td>
<td>15.00</td>
<td>-10.00</td>
</tr>
<tr>
<td>Polo Shirt - Blue</td>
<td>5.00</td>
<td>4.00</td>
<td>4.00</td>
<td>25.00</td>
<td>5.00</td>
<td>25.00</td>
<td>5.00</td>
<td>25.00</td>
</tr>
<tr>
<td>Sweater - Black</td>
<td>7.00</td>
<td>6.00</td>
<td>6.00</td>
<td>16.67</td>
<td>7.00</td>
<td>16.67</td>
<td>7.00</td>
<td>16.67</td>
</tr>
<tr>
<td>Shorts - White</td>
<td>9.00</td>
<td>8.00</td>
<td>8.00</td>
<td>11.11</td>
<td>9.00</td>
<td>11.11</td>
<td>9.00</td>
<td>11.11</td>
</tr>
<tr>
<td>Shorts - Blue</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>0.00</td>
<td>1.00</td>
<td>0.00</td>
<td>1.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Skirt</td>
<td>2.00</td>
<td>2.00</td>
<td>2.00</td>
<td>0.00</td>
<td>2.00</td>
<td>0.00</td>
<td>2.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Hat</td>
<td>3.00</td>
<td>3.00</td>
<td>3.00</td>
<td>0.00</td>
<td>3.00</td>
<td>0.00</td>
<td>3.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Scarf</td>
<td>4.00</td>
<td>4.00</td>
<td>4.00</td>
<td>0.00</td>
<td>4.00</td>
<td>0.00</td>
<td>4.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Gloves</td>
<td>5.00</td>
<td>5.00</td>
<td>5.00</td>
<td>0.00</td>
<td>5.00</td>
<td>0.00</td>
<td>5.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Boots</td>
<td>6.00</td>
<td>6.00</td>
<td>6.00</td>
<td>0.00</td>
<td>6.00</td>
<td>0.00</td>
<td>6.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Flip Flops</td>
<td>7.00</td>
<td>7.00</td>
<td>7.00</td>
<td>0.00</td>
<td>7.00</td>
<td>0.00</td>
<td>7.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Watch</td>
<td>8.00</td>
<td>8.00</td>
<td>8.00</td>
<td>0.00</td>
<td>8.00</td>
<td>0.00</td>
<td>8.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Phone Case</td>
<td>9.00</td>
<td>9.00</td>
<td>9.00</td>
<td>0.00</td>
<td>9.00</td>
<td>0.00</td>
<td>9.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Wallet</td>
<td>10.00</td>
<td>10.00</td>
<td>10.00</td>
<td>0.00</td>
<td>10.00</td>
<td>0.00</td>
<td>10.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>
```

Let's add a chart:

1. On the ANALYZE tab under PIVOTTABLE TOOLS, choose PivotChart in the Tools section of the ribbon.
2. In the Insert Chart window, select Line and click on OK.
3. Click on the chart and remove Item_Desc from AXIS in the PivotChart Fields list. Note that this will also remove it from the PivotTable.
4. As you did in the last step, click on the drop-down list for Item_Number. Select Value Filters and then Does Not Equal.
5. When the Value Filter (Item_Number) window opens, change Show items for which to Sum of Turns, and set does not equal to 0. Click on OK to close the window.
Isn't it easy to see that the **ACCS-CRD-25BK** item has the highest turnover rate?

Use your imagination!

**What this BI content achieved**

By using this report, you can easily see which items have a low turnover rate and, thereby, a greater investment in the inventory compared to those with a high turnover rate and, thereby, a lesser cash investment in the inventory. The preceding screenshot makes it very easy to see that the black 25-foot phone cord turns over more quickly than anything else during the quarter we are reviewing. Why is that so? Did we have this item on sale? During the quarter being reviewed, we did not sell any other phone cords. Then why? The two highest turnover items were both accessories. Why did accessories sell better than other items?

Just this bit of information helps us evaluate how our customers purchase what they purchase, and creates a large variety of questions that might just make your business more profitable, resulting in a bonus for you. Hey, it could happen!
Items on which you most frequently adjust quantities

In theory, inventory adjustments and variance transactions should never need to be done except for occasional damaged item. The reality is that most users of the inventory control module make adjustments on a regular basis. Even the stock count feature generates variances for discrepancies between the actual count and the count in GP. If everything were entered completely correctly, with no timing issues (for example, an item being shipped between the time the stock count report was created and the user counted the stock), GP would always be completely accurate and completely in balance with the General Ledger. Unfortunately, we live the real world, where stuff happens and people get sidetracked and make mistakes.

Microsoft Dynamics GP introduced Reason Codes in GP 2013. These reason codes enable GP users to explain why an adjustment or variance is made, even using different GL account numbers to record the entry.

Since we end up making adjustments and we can now track the reasons for these adjustments, we should look at entries and see whether there is any information we can obtain. We’ll be looking for anything that costs us more than it should and/or revenue opportunities.

Brief background of the company

This report is for a distribution company that sells big-ticket items. These are low-margin items (about 20 percent), so if they lose a single item, they will have to sell four more just to break even.

Pain or goal defined

The owner has noticed that there are a lot of inventory adjustments, both increases and decreases. It has been difficult to tell whether these result from items being shipped while counts are occurring, or some items are slipping out the back door. We use a two-step approach to get a better handle on the inventory. First, we identify the items that have the most adjustments. Then we perform more frequent inventory counts for those items.

BI and BI tool chosen

We will be using a basic PivotTable to access the data directly from GP. A timeline filter will be used to allow us to select a single period or a range of periods.
Step-by-step guide to build the BI content

There are two parts in these steps. The first part is technical, containing steps that involve work in SQL Server. The second part is the actual building of the report. You may choose to obtain assistance with the technical part, especially if you do not have access to SQL Server.

Technical matter (in case you're not a techie!)

Create the following view in SQL Server:

```
CREATE VIEW View_BI_Inv_Adj
AS
SELECT dbo.IV00101.ITEMNMBR AS Item_Number,
       RTRIM(dbo.IV00101.ITEMDESC) AS Item_Desc,
       CASE IV30300.DOCTYPE
           WHEN 1 THEN 'Adjustment'
           WHEN 2 THEN 'Variance'
           ELSE 'Other'
       END AS Trx_Type,
       dbo.IV30300.DOCDATE AS Trx_Date,
       dbo.IV30300.DOCNUMBR AS Trx_Number,
       dbo.IV30300.TRXQTY AS Quantity,
       dbo.IV30300.UOFM,
       dbo.IV30300.TRXLOCTN AS Site_ID,
       dbo.IV30300.Reason_Code,
       dbo.IV30300.DOCNUMBR + 'IV30300LNSEQNBR' AS Trx_Key
FROM dbo.IV30300
INNER JOIN dbo.IV00101 ON dbo.IV30300.ITEMNMBR = dbo.IV00101.ITEMNMBR
WHERE (dbo.IV30300.DOCTYPE < 3)
GO
GRANT SELECT ON View_BI_Inv_Adj to DYNGRP
```

Get help if you need it. Back-up before you do this and wait 30 minutes to swim after eating.

Non-technical matter

Let's build a report!

Know what you want, draw it, and create your requirements.
Would you fill gas in your car before you know how much gas is already in the tank?

Let's establish a connection to the GP data:

1. Open a Blank workbook in Excel. From the menu bar, choose DATA. In the Get External Data portion of the ribbon, choose From Other Sources. Then choose From SQL Server as the source.

2. Enter your Server name and network or AD login credentials. Click on Next.

Here's a reminder: the GP login credentials are the user name and password you use to log in to GP. If you are not a GP user, you'll need a security set up for yourself in SQL Server. See the technical part of the first report in Chapter 2, Business Intelligence for the General Ledger for more details.
3. Select your database from the drop-down list. Select *view_BI_Inv_Adj*. Then click on **Finish**:

![Data Connection Wizard](image)

4. On the **Import Data** window, select **PivotTable Report**. Make sure that you are putting the data in the **Existing worksheet** option in the =$A$1 cell. Then click on **OK**. The PivotTable canvas will appear.
Let's build the foundation of our PivotTable report:

1. In the **PivotTable Fields** list on the right, drag the following fields into the **ROWS** area: **Item_Number**, **Item_Description**, and **Reason_Code**:

   ![PivotTable Fields](image)

   We are putting **Reason_Code** in our table; you could put it in a slicer, as we'll do with the type (adjustment or variance), if you want to easily and frequently change the report to show by **Reason Code**.

   GP 2013 had a new feature: adding a Reason or Reason Code to make an inventory adjustment or variance. If you have an inventory, this is worth the upgrade alone.
2. In the **Design** tab under **PIVOTTABLE TOOLS**, choose **Report Layout** in the **Layout** section of the ribbon. Select **Show in Tabular Form**:

![Image of Excel Design tab with Report Layout selected]

3. In the **Design** tab under **PIVOTTABLE TOOLS**, choose **Subtotals** in the **Layout** section of the ribbon. Select **Do Not Show Subtotals**:

![Image of Excel Design tab with Subtotals deselected]

4. In the **PivotTable Fields** list on the right, drag the **Quantity** and **Trx_Number** fields into the **VALUES** area:

![Image of Excel PivotTable Fields with fields dragged into VALUES area]
5. **Quantity** should appear in the PivotTable as a sum, while **Trx_Number** should appear as a count. If these are not correct, we can correct them when formatting the fields.

6. In the **PivotTable Fields** list, in the **VALUES** area, click on the drop-down list for the **Quantity** value. Select **Value Field Settings**:

![PivotTable Fields](image)

- **Quantity** is the actual number of items that were adjusted, while **Trx_Number** represents each occurrence of the adjustment. **Trx_Number** is a unique string field (a text field even if it contains only numbers). By counting this field, we can know how many adjustments were made, regardless of how many items were adjusted.

7. In the **Value Field Settings** window that opens, make sure that **Sum** is selected as the **Summarize value field by** option. Click on **Number Format**. The **Format Cells** window will open. Change **Category** to **Number**. Select how you want your values to be displayed, and then click on **OK** for each window to close it.
We prefer using the **Number** category with 0 decimals, but with a comma as a **1000** separator.

8. If `Trx_Number` is not set to **Count**, change **Value Field Settings** for that value as well.

9. On the **ANALYZE** tab under **PIVOTTABLE TOOLS**, choose **Insert Slicer** in the **Filter** section of the ribbon:
10. In the **Insert Slicer** window, select the checkbox for **Trx_Type**. Then click on **OK**:

![Insert Slicer window](image)

11. Drag the slicers to the right of your PivotTable. You can shrink a slicer by clicking on the handle and dragging it over empty space:

![PivotTable screenshot](image)

12. Another addition you may want to make is a **Timeline** filter on the report. The report will calculate for whatever time period you select. On the **ANALYZE** tab under **PIVOTTABLE TOOLS**, choose **Insert Timeline** in the **Filter** section of the ribbon:

![Timeline filter](image)
13. In the **Insert Timeline** window, select the checkbox for **Trx_Date**. Then click on **OK**. Note that this window will only give you options for date fields:

Now your report can be pulled using date filters and/or transaction type filters.

14. Another powerful option might be to add an inventory site (**Site_ID**) as a slicer:

15. Add a logo and any text you like. To refresh the data from GP, click on **Refresh All** under the **Data** tab.

An easy variation of this report has **Timeline** set in **YEARS**:
16. Add a slicer for **Reason_Code** and remove it from **ROWS**.
17. Remove **Trx_Number** from **VALUES**.
18. Add **Trx_Date** to **COLUMNS**.
19. Right-click on the first date in the first column and choose **Group**:

![Grouping window](image)

20. In the **Grouping** window, click on **Years**, so both **Years** and **Months** will be selected. Click on **OK**:
We are selecting **Years** to prevent columns from showing January for all years combined, and the same with the following months.

Our final report looks like this:

![Inventory Adjustments & Variances for a Given Period](image)

**What this BI content achieved**

By using this report, this company can easily see which items are getting adjustments made and which quantity is being changed. In the final report, we can see that there are three damaged items from 2016 and 2017 adjusted. Is this normal for these items? Did we return them to our suppliers for a credit or did we break them? So many questions, so little time!

**Summary**

Getting a handle on inventory alone can offer full and quick returns on your GP investment. The data is there, so use it to monitor your inventory and to make sure that your inventory is there as well.

In this chapter, we actually built reports that created many questions. How many of these questions can yield revenue opportunity, prevent unnecessary costs, and reduce shrinkage and/or theft?

Next, we'll continue the conversation on inventory control and business intelligence.
Where to buy this book

You can buy Real-world Business Intelligence with Microsoft Dynamics GP from the Packt Publishing website.

Alternatively, you can buy the book from Amazon, BN.com, Computer Manuals and most internet book retailers.

Click here for ordering and shipping details.