

Overview of Solver's BI360: A Powerful, Excel-based BI Solution

Reporting - Planning - Dashboards - Data Warehouse

A Solver White Paper

July 2012 solverusa.com Copyright © 2012



Table of Contents

Table of Contents	1
Introduction	3
Architecture Overview	3
User Interface Overview	5
Reporting Components	6
Excel Ribbon Excel Task Pane Composer for Ad-Hoc Reporting Practical Usage Examples	7 9 0
BI360 – Planning Module1	4
User Interface Overview1	5
Planning Components1	5
Data Settings	.7 .7
Typical Implementation Process2	0
BI360 – Dashboard Module2	2
User Interface Overview2	3
System Requirements2	4
BI360 – Data Warehouse2	7
Data Warehouse Interface (Admin Console) Overview2	7
Data Warehouse Manager Components2	7
Modules2Dimensions2Currency2Data Explorer3	8 9

Processes Practical Usage Examples	
Typical Implementation Process	. 33
Module – Dimension Mapping	.34



BI360 White Paper | Page 2

Introduction

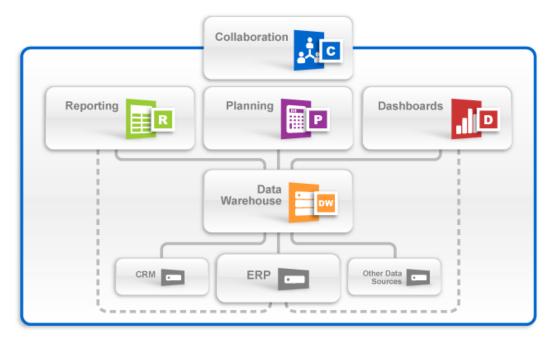
BI360 is the first full Business Intelligence (BI) suite powered by Microsoft Excel (Budgeting and Reporting), Microsoft Silverlight (Dashboards) and Microsoft SQL Server (Data Warehouse). Built for quick, low risk implementations and to be completely managed by business users, BI360 is designed to completely satisfy the business performance management needs of divisions as well as corporate headquarters.

The features that differentiate BI360 the most from other BI applications are:

- 1. Third generation Excel add-ins for simple or complex reporting and budgeting processes.
- 2. Pre-Configured and extensible "smart" data warehouse.

Architecture Overview

Below you can see a simplified architecture of all BI360 modules. The *Reporting* module can both report live on any of the Microsoft Dynamics ERP systems as well as off the BI360 *Data Warehouse Manager*. The *Planning* module reads data to and from the *Data Warehouse Manager*, while the *Dashboard* module reads the KPIs from the *Data Warehouse Manager*. The *Data Warehouse Manager* allows BI360 to work on top of an organization's data source(s) by loading data in and exposing it to BI360's three end-user tools.





As an Excel-based report writer that can connect directly to your Dynamics ERP or CRM system, and to the BI360 *Data Warehouse Manager*, BI360 *Reporting* (OneStop Reporting 'OSR') can be used for financial reporting as well as various types of operational reporting.

With all, or a majority of the reports built with BI360 *Reporting*, organizations typically find that they now get 80% of the "BI" and reporting done with 20% of the effort it previously required. In other words, some of the key reasons for implementing OSR are:

- 1. Excel interface that most business users are comfortable with.
- 2. Easy to build reports with drag and drop wizards and pre-built formulas.
- 3. Reporting on GL as well as sub-ledgers and Microsoft CRM.
- 4. Reporting on *Data Warehouse Manager* (if implemented).
- 5. Single report writer for many purposes.
- 6. Consolidations across companies (requires BI360 *Data Warehouse Manager* if multiple different ERP systems are in use).
- 7. Currency conversion (requires BI360 *Data Warehouse Manager* for advanced currency conversion).
- 8. Special needs such as allocations and reconciliations.
- 9. Eliminate any manual Excel exports/reports as well as proprietary report writers.
- 10. Ad-hoc reporting (OSR Composer module) without the need for formulas, OLAP cubes, and pivot tables.

As seen in the architecture diagram earlier in this document, BI360 *Reporting* is Excel-based and it is one of four BI360 modules. The other three modules are:

- 1. BI360 Data Warehouse Manager Optional, this is for support of reporting across multiple ERP systems, etc.).
- 2. BI360 Planning module Optional.

3. BI360 *Dashboard* module – Optional, this provides a dashboard interface to data stored to the *Data Warehouse Manager*.

In the architecture diagram earlier in this document, you can see how BI360 *Reporting* interacts with the other BI360 modules. In essence, it serves as a report writer and an ad-hoc query tool both for live reporting on Microsoft's ERP and CRM systems, as well as for the BI360 *Data Warehouse Manager*, in which case data can come from any source system.

Here are some scenarios for when to use BI360 *Reporting* with the BI360 *Data Warehouse Manager*:

- Reporting across multiple source systems (like multiple ERP systems).
- Advanced currency conversion.
- Performance (want faster reporting without slowing down the source system).
- Budget reports (either for BI360's *Planning* module or for other budget system).
- Designing budget templates and other business input forms (with BI360's *Planning* module).

User Interface Overview

BI360 *Reporting* is a modern Excel add-in that has been designed with the latest Microsoft technologies and follows Microsoft best practices for interface design and usability. The Interface was designed to maintain a similar look and feel to the Microsoft Office products, including the use of a ribbon and an Outlook-style menu bar as the main navigation components.

BI360 *Reporting* also has a back-end tool called the Meta Data Designer. It is the tool used by power users to create integrations between the Excel front-end and the data source (such as the ERP system). The meta data produced by the Meta Data Designer provides BI360 *Reporting* with major advantages over standard query tools and report writers that connects directly to a data source without any form of meta data. To begin with, it shields report designer users from any type of SQL queries and it translates often difficult SQL field and table names into plain English business terms. Maybe even more important, the Meta Data Designer allows developers and power users to pre-define important logic such as time/period logic, which then can eliminate the use of technical period formulas when users design reports.

Note: For Microsoft Dynamics systems, the BI360 Data Warehouse Manager and certain other systems, the meta data is already developed and out-of-the box, while for other systems it takes configuration to be set up. The Meta Data Designer can also be made available to partners/customers so that they, on their own, can expand the use of BI360 Reporting to also provide live reporting on other in-house databases.

Reporting Components

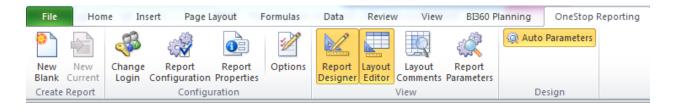
BI360 *Reporting* is an Excel add-in just like BI360's *Planning* module. As you see in the image below, both BI360 *Reporting* (OneStop Reporting) and BI360 *Planning* are located right next to each other on the Excel ribbon. In other words, *Reporting* and *Planning* are available side-by-side without the need to leave Excel.

BI360 *Reporting* consists of two menus: The Excel ribbon and the Excel task pane. The Excel ribbon is usually only used by administrators for setup and maintenance of BI360 *Reporting* reports.

Excel Ribbon

The Excel ribbon has buttons that provide access to the following functions:

- Start the design of new reports (either from scratch or based on an existing Excel report).
- Manage your login (server connection).
- Manage configuration of the current report (as it relates to the data source).
- Manage report properties and setting options.
- Open/close Task pane and Design windows.
- Manage report parameters (prompts).



Excel Task Pane

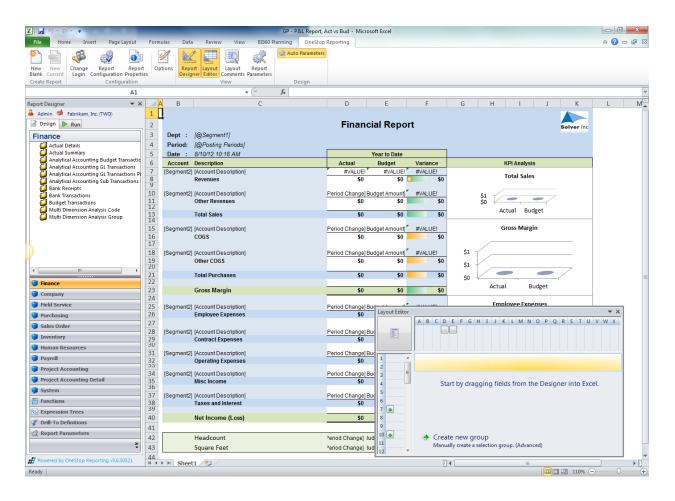
BI360's *Reporting* task pane features the popular Microsoft Outlook-style menu bar. It comes in two main flavors:

- 1. OneStop Reporting Designer: Gives access to both design and run reports (typically for power users).
- 2. OneStop Reporting Player: Gives access to run reports (typically for end users).

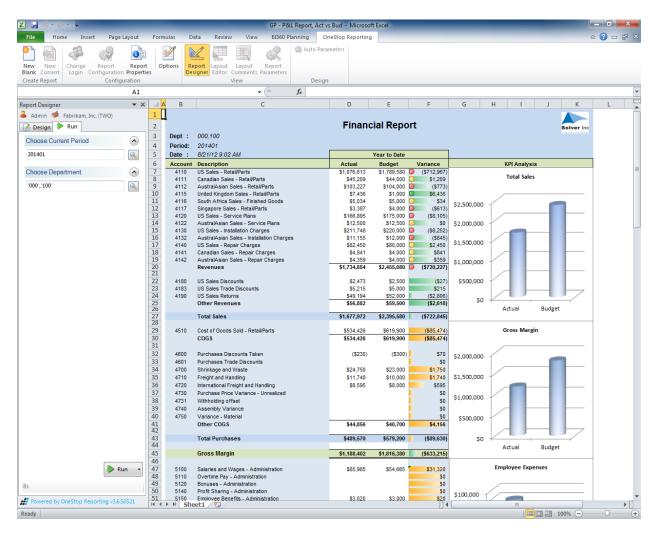
The OSR Designer menu is shown on the left side of the image below and it offers the following main features:

- Access to fields in different modules in the source database (like GL, Sales, Accounts Payable, etc.).
- KPIs (KPIs can be pre-defined and dragged into any report).
- Functions (pre-defined period functions such as: Current Month, YTD, Rolling 12 Month and many more).
- Expression Trees (pre-defined trees that can be dragged into any report. E.g. Profit & Loss account trees).
- Report Parameters (access to the parameters that will display for the users as filters when they run a report).

When designing a report, the user selects the fields, trees, periods, etc. from the OSR Designer menu and drags and drops these items into the appropriate location in the Excel sheet. Most reports can be designed without entering a single formula, with the exception of for example a standard Excel formula that deducts one Excel cell from another to calculate the variance between a column with actual figures and a column with budget figures.



The OSR Player menu is available as a second tab on the OSR task pane for users with the OSR Designer license and it is also available as a standalone for true end-users that only need it to run reports. Either way, it looks and works exactly the same. Essentially, it displays the report parameters as prompts on the left side of the screen (see image below) and the user can click on the lookup buttons to choose from dimension list that will appear (e.g. to choose a period or one or more departments). The report parameters (prompts) are created by the person designing the report, and they make a report truly dynamic so it can be run for any month, any department, any vendor, any project, and so on. The resulting data will be filtered before it is pulled from the source database, thereby making reports faster and eliminating the need to push all data to Excel to then filter it.



Once a report has been run (see image above), the user can right-click in any cell to drill down and analyze the underlying detailed transactions. The drill-down function is always available without the need to pre-define any drill down functions when the report is designed.

Composer for Ad-Hoc Reporting

BI360 *Reporting* also offers an optional module called Composer (see image below). The Composer is designed to solve ad-hoc (on the fly, instant) reporting needs as compared to the BI360 *Reporting* Excel add-in which is focused on creating and running formatted reports. Typical ad-hoc users are accountants needing to see e.g. how much has been paid to a certain vendor so far this year, or a sales manager that wants to see what they have actually invoiced to a customer and if they have been paid yet.

The OSR Composer has a very simple interface where everything is accomplished with drag and drop from the familiar Outlook-style menu on

Copyright Solver, Inc., 2012

BI360 White Paper | Page 9

the left. This menu is the exact same menu that BI360 *Reporting* utilizes in Excel (see prior paragraphs about the OSR Designer.

The composer is also connected to the OSR Designer and a query designed in the Composer can, with a few clicks, be turned into a fully formatted Excel report template in the OSR Designer.

🧕 🖺 📁 🔕 🕻 🥌 <u>R</u> ecent Files 🔹 🌡					
manya	< Posting Periods	201401:201403 🔍 🎗			
🥥 Advanced Budgeting					
Advanced Budget Detail Advanced Budgeting Summary					
Advanced budgeting summary Bank Receipts					
Bank Transactions					
Budget Transactions					
Actual Summary					
Actual Details Multi Dimension Analysis Code					
 Multi Dimension Analysis Code Multi Dimension Analysis Group Analyticial Accounting GL Transactions 	Segment1 🗠				
Analyticial Accounting Budget Tranactions	Account String	Account Description	Period Balance	Period Change	
🟳 This Company	■ 000				
	- 100				
	100-5150-00	Employee Benefits - Administration	5,859.00		
	100-5170-00	Payroll Taxes - Administration	2,051.86	2,051.86	
	100-6100-00	Training - Administration	1,500.00		
	100-6110-00	Company Car - Administration	297.50	297.50	
	100-6130-00	Supplies/Hardware - Administration	249.50	249.50	
	100-6140-00	Supplies/Software - Administation	229.00	229.00	
	100-6150-00	Supplies-Allocated - Administration	243.75	243.75	
	100-6160-00	Dues & Subscriptions - Administration	50.00	50.00	
	100-6170-00 100-6180-00	Repairs & Maintenance - Administration Rent Expense - Administration	259.31 2.787.19	259.31 2.787.19	
	100-6190-00	Utilities Expense - Administration	315.79	315.79	
	100-6500-00	Postage/Freight - Administration	77.91	77.91	
	100-6510-00	Telephone - Administration	225353	2.253.53	
	100-6520-00	Travel - Administration	2.049.28	2,049.28	
Finance	100-6530-00	Meals/Entertainment - Administration	250.00	250.00	
Company	100-9010-00	Square Footage-Administration	50,000.00	50,000.00	
Field Service			68,473.62	68,473.62	
Purchasing	- 200				
Sales Order	200-5170-00	Payroll Taxes - Accounting	8,773.40	8,773.40	
	200-6100-00	Training - Accounting	500.00	500.00	
Inventory	200-6130-00	Supplies/Hardware - Accounting	489.95	489.95	
Fixed Assets	200-6140-00	Supplies/Software - Accounting	537.00	537.00	
Human Resources	200-6150-00	Supplies-Allocated - Accounting	609.38	609.38	
	200-6170-00	Repairs & Maintenance - Accounting	864.38	864.38	
Payroll	200-6180-00	Rent Expense - Accounting	2,787.19		
Manufacturing	200-6190-00	Utilities Expense - Accounting	315.79	315.79	
Project Accounting	200-6500-00	Postage/Freight - Accounting	129.29	129.29	
Project Accounting Detail	200-6510-00	Telephone - Accounting	2,253.53	2,253.53	
Functions	200-9010-00	Square Footage-Accounting	50,000.00	50,000.00	
			67,259.91	67,259.91	
KPIs	- 300				
Reports	300-5130-00	Commissions - Sales	50,840.61	50,840.61	
	300-6100-00	Training - Sales Supplies/Software - Sales	1,520.00 792.00	1,520.00 792.00	
	300-6140-00				

Practical Usage Examples

In most cases, BI360 *Reporting* will be used as a reporting and ad-hoc query tool, either stand alone or as part of the BI360 suite. Either way, there are two main deployment options:

- 1. Give the reporting tools to a few central power users.
- 2. Provide the tools to end users so that on their own they can get information and drill-down whenever needed.

Below is a table listing various deployment options:

Deployment	Advantage	Disadvantage
Power users run reports centrally and distribute/e-mail to users	 No license required for end users 	 End users cannot get reports exactly when they need them End users may want to see reports for other filter criteria End users cannot drill down
Power users run reports and post to a shared network drive or SharePoint	 No license required for end users 	 End users cannot get reports exactly when they need them End users may want to see reports for other filter criteria End users cannot drill down
Power users create ad- hoc queries and export result to Excel and e- mail/print/save to server	 No license required for end users 	 End users cannot answer instant questions when they need it and without requesting information from power user
End-users run their own reports	 End users get reports exactly when they need them End users can run reports with different filter criteria End users can drill down 	License required for end-user
End-users create ad-hoc queries	• End users can answer instant questions when they need it and without requesting information from power users	 License required for end-user

Here are examples of the numerous potential uses of BI360's Reporting module:

Purpose	Usage Examples
Financial reporting	 Financial statements in various formats, like Profit & Loss, Balance Sheet and Cash Flow reports.
Operational reporting	 Sales reports Vendor reports Project reports HR/Payroll reports
Dashboard reports	 Create Excel-based dashboards that combine metrics/KPIs with charts and indicators (Excel 2007 and later supports traffic lights, arrows, etc.). If data is coming from multiple sources, deploy BI360's <i>Data Warehouse Manager</i> module, import data there and deploy <i>Dashboard</i> reports on top of the <i>Data Warehouse Manager</i>.
Scorecard reports	 Create Excel-based scorecards with KPIs (Excel 2007 and later supports traffic lights, arrows, etc.). Management's scorecard comments can be stored with BI360's Planning module. If data is coming from multiple sources, deploy BI360's <i>Data Warehouse Manager</i>, import data there and deploy dashboard reports on top of the <i>Data Warehouse</i> <i>Manager</i>.
KPI reports	 Create Excel-based reports that generate and display KPIs. (Excel 2007/2010 supports traffic lights, arrows, etc.).
Allocation reports	 Create Excel-based reports that calculate allocations. If you want to store the allocations and bring them back to the ERP system, use <i>BI360's Planning</i> module. You can also use <i>BI360 Planning</i> to collect and store allocation drivers.
Commentary input	 Create Excel-based reports (e.g. a P&L) and set up BI360's budget module to collect and save text comments. Create Excel-based reports to show all comments made to certain accounts, for

	certain variances, etc.
Budget template design	 Create Excel-based input templates for use with BI360's budget module.
Budget reports	 Create Excel-based reports to consolidate/view budget.
Workflow reports	 Create Excel-based reports for instant views of all Approved/Pending/Rejected budgets.

Typical Implementation Process

A typical implementation process of BI360 *Reporting* could look like this (in this example assume that BI360 *Reporting* is being used for typical reporting processes):

- 1. Train power users.
- 2. Identify reporting needs.
- 3. Preparation of the BI360 *Data Warehouse Manager* Skip this step if OSR will only be used for live reporting
 - a. Populate the *Data Warehouse Manager* with data and dimensions from your source systems. Examples of dimensions are: Account, Department, Project, Product, Employee, Asset, etc.
 - b. Enter any other dimensions or data directly in the *Data Warehouse Manager* if they do not already reside in another database from which they can be imported.
- 4. Design Excel report templates with BI360 *Reporting*.
- 5. Set up security (if different users require different access rights).
- 6. Train end users (*will only take minutes as they just need to learn how to run reports and drill down*).

Planning

BI360 – Planning Module

As an Excel-based data entry tool with a flexible SQL Server database and a powerful companion report writer (BI360 *Reporting*), *Planning* can be used to automate budgeting and forecasting as well as other business processes that require data entry and a database. In other words, think of BI360 *Planning* as an automated way to save any data from Excel into a structured SQL database. For most people, budgets and forecasts are the first processes that come to mind, but in reality, most organizations have numerous manual data collection processes that could be automated with Planning/BI360.

As seen in the architecture diagram earlier in this document, *Planning* is Excel-based and it is one of four BI360 modules. The other three modules are:

- 1. BI360 *Data Warehouse Manager* Required, this is where *Planning* stores all data from Excel.
- 2. BI360 Reporting module Optional.
 - a. Used if you want to create dynamics budget templates.
 - b. Used as a report writer to report on all the data that *Planning* store to the *Data Warehouse Manager* database.
- 3. BI360 Dashboard module Optional, this provides a dashboard interface with the data stored to the *Data Warehouse Manager*.

In the architecture diagram shown earlier in this document, you see how BI360 *Planning* interacts with the other BI360 modules. As you will learn later in this document, administrators first configure *Planning* and the *Data Warehouse Manager* database for the Excel model where the data entry and write-back will take place. The Excel template(s) is then provided to the end-users for data entry. After the users have entered and saved their data, managers can start reporting on the entered data. Finally, if desirable, the data can be transferred from the *Data Warehouse Manager* database and back to other source systems such as a budget table in the general ledger.

User Interface Overview

BI360 *Planning* is a modern Excel add-in that has been designed with the latest Microsoft technologies and follows Microsoft best practices for interface design and usability. The interface was designed to maintain similar look and feel to the Office products, including the use of a ribbon and Outlook-style menu bar as the main navigation components.

Planning Components

BI360 *Planning* is an Excel add-in just like BI360 *Reporting*. As you see in the image below, both *Planning* and *Reporting* are located right next to each other on the Excel ribbon. In other words, reporting and budgeting are available side-by-side without the need to leave Excel.

BI360 *Planning* consists of two menus: the Excel ribbon and the Excel task pane. The latter is only used by administrators for setup and maintenance of *Planning*.

Excel Ribbon

- View, Check-in & Check-out Assignments.
- Save data.
- Update model settings from the server (the BI360 *Data Warehouse Manager*).
- Connection information (to the BI360 *Data Warehouse Manager* database).
- Access to administrative functions (task pane).
- Direct link to resources (Including BI360 documentation, videos and samples).

File	Home	Insert Pag	e Layout	Formulas	Data	Review	v View	BI360 Planning	OneStop Reporting
-this		<u>s</u> 2	🚛 Offline	t -		W	Po	🚯 🕕 About	
Enter Data	Save Data	Assignments *		Adm Pan			Publish to PowerPoint	Resource Center	
Da	ta	Т	asks	Configu	iration	Office	Publisher	Resources	

Excel Task Pane

BI360 *Planning* task pane features the popular Microsoft Outlook-style menu bar. It has three main menu items:

Data Settings

This is where the administrator performs the initial configuration by providing *Planning* with vital information about where data and dimensions are located in the underlying spreadsheet. Depending on how well organized and structured your Excel model is, BI360 Planning can be auickly configured to handle unlimited data input sections in one or multiple worksheets in a workbook. BI360 *Planning* can also automatically adapt to spreadsheets that "grow" (e.g. a new row or column that gets inserted in the Excel model) or "shrink" without the administrator having to update the settings. If there is a significant structural change to the company's Excel model, then the administrator can make the necessary changes to the *Planning* Data Settings that the end-users will see next time they open the Excel template (if the same Excel template is accessed centrally by all users) or if the Excel model has been distributed to endusers, they simply click the "Update Settings" button on the *Planning* Excel ribbon and it will retrieve the latest settings from the BI360 Data *Warehouse Manager* database. Part of the data setting setup is to choose where in the BI360 Data Warehouse Manager database the data from Excel should be stored based on the type of data. For example, data (like budgets) related to Accounts is stored to the Finance (General Ledger) Module, detailed revenue data is stored in the Revenue Module, detailed employee/position data is stored to the Payroll Module, and so on.

Admin Panel 🔹 🗙	12		B C D F G	H I J K L N O P C R S T U
Data Settings		4	Company : [@Entity] Department : [@Department]	ACT LY
4 🛅 Forecast	1	6 7	Forecast Ver: [@Scenario] Currency : USD	Forecast
ForecastGL		8	currency : 05D	
		9 10	ACT FCST Account Account Description [FYPeriod] [FYPeriod]	Data Settings
		12 13 14	Revenues #VALUEI (Amount) [Description] #VALUEI (Amount) 0 Total Revenues 0 0 0 Expenses 0 0 0	Data Settings Create settings for the data in your Excel workbook that should be written to the database
	+	18 19	Total Expenses 0 0 Net Profit (Loss)	Worksheet: Forecast Copy Settings from 🔹
		21 22 23 24	Headcount Total Revenue (Act Last Year) ↑ ↑ ↑ Total Revenue (Act + Fcst) ↑ ↑ ↑ Profit Margin #DIV/0! #DIV/0!	Grid Title: ForecastGL Grid Rules: Ignore T columns, Ignore T r Module Data Grid Range Add Optional Dimensions:
		26 27 28	1	GL • Forecast!\$F\$13:\$L\$14 _ • Ok
		29 30 31		Dimension Dimension Location Cell Reference Account Row Forecast(\$B\$13
		32 33 34	0	Currency Global V Forecast/\$C\$7
		35 36	100.0%	Entity Global 👻 ForecastI\$C\$4
		37 38	80.0%	Scenario Global - Forecasti\$C\$6 _ @
		39 40 41	Log. (Profit Margin)	Period Column - Forecast(\$F\$10:\$L\$10 _ 🔒
		42 43 44	0.0% [FYPeriod] [FYPeriod]	Add Storage Field: UDF1, UDF2, Amount
		45 46 47		Field Column Reference
		48 49		UDF2 Forecast(\$J\$13 _ 🗶
Utilities Workflow		50 51 52		Amount Forecast/\$F\$13:\$G\$13 _ 💥 👻
Interface Settings Data Settings		53 54 55		Ok Cancel
»,		56	Forecast / 😒 /	

Copyright Solver, Inc., 2012

BI360 White Paper | Page 16

Interface Settings

The *Planning* Interface Settings (see screenshot below) are directly related to the *Planning* Data Settings. While the Data Settings are focused on what sections of Excel should be saved where in the *Data Warehouse* and with which dimensions, the Interface Settings are focused on what you want the end-users to see if they use the *Planning* data entry screen (shown in the next paragraph).

Settings For InfoManager	
Data Grid Selection :	ForecastGL
Chart Properties	*
Data Grid Properties	*
🥅 Hide this component in InfoManager	
Worksheet and Cell range containing the column labels you want to display:	Forecast!\$P\$10:\$Q\$10
Worksheet and Cell range containing the row labels to display (for first row in gr	id): Forecast!\$B\$13:\$C\$13
Worksheet and Cell range containing the data to display (for first row in grid):	Forecast!\$P\$13:\$Q\$13
Display Label :	Actual
	Display Decimals 0 束
Input Grid Properties	
Hide this component in InfoManager	
Worksheet and Cell range containing the column labels you want to display:	Forecast!\$F\$10:\$G\$10
Worksheet and Cell range containing the row labels to display (for first row in gr	id): Forecast!\$B\$13:\$C\$13
Worksheet and Cell range containing the data to display (for first row in grid):	Forecast!\$F\$13:\$G\$13
Display Label :	Forecast
	Display Decimals 0
Line Item Details Proper	ties 🛛 🕅
Totals Component Proper	ties 🛛
Slider Properties	*
Slider Range Percentage 50 💭 Minimum Sli	der Move Percentage 5
Formatting Options Use 1,000 Separator (,)	Ok Cancel

Data Entry Window

The *Planning* data entry window (see image below) is an optional way of entering data in Excel. In other words, users can type information directly into Excel, or they can open the BI360 *Planning* data entry screen and enter it there. When the data has been entered into Excel, it is automatically saved into the underlying row in Excel and then can be saved to the *Data Warehouse* database. The *Planning* data entry screen is a powerful tool for:

- Automatic spreading of numbers across many cells in Excel. Many spreading rules are available, including rules that can use prior year actuals or seasonal trends as a base.
- Input of text comments at the row or line item level.

- Input of many rows of line item detail below a single row in Excel (e.g. the user can list many business trips that automatically will roll up to a single travel expense row in Excel).
- Automatic charting of the current row in Excel and comparison to e.g. another row with historical data.

1 🖬 🔊 - Ci - 🛕	-	2.51		SIM I	Forecast_wAtt	ributes_demo	.xlsx - Micro	soft Excel				-		- 2
File Home	nsert Page Layout	t Formulas	Data	Review	View S	olver InfoMan	ager De	veloper (OneStop Repo	rting Tea	n		🛥 🕜 🛥	đ
nter Save Assignm Data ~ Planning Tools	ents Current Curren Row + Sheet		Manageme Comments	nt Lupo	late Settings late Connectio n Admin Pane Adminstra	9	Resource Center	About						
		Approval			Administra	tion	1003	ources						
oManager 2.0 Data Grid: Actua													_	▼ ≈
Description	200902	200903	200904	200905	200906	200907	200908	200909	200910	200911	200912		Total	Â
40010 Product Rev	inue 1,179,300	1,191,200 1,	215,500	1,227,800	1,252,900	1,252,900	1,291,700	1,331,700	1,318,500	1,345,400	1,387,000		13,993,900	
nput Grid: Fored	ast													*
Input Grid	Percent 0		Ro	und Up 🔲	Spr	reading Total	Spreading	Method	Comments					
· ·							Even	-						
C Line Item Details			De	cimal	÷									
) (
Description	201002	201003	201004	201005	201006	201007	201008	201009	201010	201011	201012		Total	
40010 Product Rev	enue 100	0	0	0	0	0	0	0	0	0	0		100	
	Go to Forecas	tGL 🔻 1	3					Upda	te Excel	Can	cel	Clear		
A1	• (* <i>f</i> x													
2 🔺 A B	С		D	F	G	Н	1	J	K	L	M	N	0	
4 Company 5 Department : 6 Forecast Ver: 7 Currency	300 BUDGET									Fo	orecast			
8		,	АСТ	FCST	FCST	FCST	FCST	FCST	FCST	FCST	FCST	FCST	FCST	
10 Account	Account Description	20	1001	201002	201003	201004	201005	201006	201007	201008	201009	201010	201011	
12 13 40010 14 40020 15 40030 16 40040	Revenues Product Revenue Services Revenue Maintenance Revenu Other Revenue		,401,050 944,300 177,100 88,550	100										
17	Total Revenues	2	,611,000	100	0	0	0	()	0	0	0	0 0	
19	Expenses													
69	Total Expenses		625,350	0	0	0	0	()	0	0	0	0 0	
70	Net Profit (Loss)	1,9	985,650	100		-	-	-	-	-	-	-	-	
	Headcount Total Revenue (Act La		242,700 🤑 511,000 🤑	100 1	h - 1	h - 4	h -	∱ -	^ -		^ -	↑ -	1 -	<u>۱</u>
72 73 74 75	Total Revenue (Act + Profit Margin	- FCST) 2,1		100.0%	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	* #DIV/0!	#DIV/0!	#DIV/0!	
73 74	Total Revenue (Act + Profit Margin	0 70	6.0%	100.0%	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	* #DIV/0!	#DIV/0!	#DIV/0!	1
73 74 75	Profit Margin		6.0%	100.0%	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	•

In short, BI360 *Planning* can store data (numbers and text) from any Excel spreadsheet. The data will always be stored to the *Data Warehouse* database. From there, the stored data can be sent back to the ERP system, reported on or even processed into OLAP cubes for advanced analysis.

Practical Usage Examples

In most cases, *Planning* will be used as a budgeting and forecasting tool. Below is a list of the most typical variations of Excel budget models that BI360 *Planning* can handle. Note that the Excel "input forms" can either be your own "static" Excel files or dynamic Excel templates that you design with BI360 Reporting or with similar Excel-based tools:

Excel Configuration	Examples	Typical Distribution
Multiple Excel workbooks with a single budget sheet	Departmental Profit & Loss Departmental Expenses	Input done by central staff Spreadsheet\template distributed to end-users
Single Excel workbook with multiple budget sheets with the <u>same</u> layout	Different departments on different sheets Different products on different sheets	Input done by central staff
Multiple similar Excel workbooks with multiple different budget sheets in each workbook	Workbook with: Summary Profit & Loss or Expense sheet Revenue Detail Employee Detail Capital Expenses	Spreadsheet\template distributed to end-users

Here are examples of <u>other</u> uses of Planning as a data collection tool.

Note that the Excel "input forms" can either be your own "static" Excel files or dynamic Excel templates that you design with the BI360 report writer or with similar Excel-based tools:

Excel Configuration and Input Type	Usage Examples
Modeling	Create a product or service price model in Excel. Use <i>BI360 Planning</i> to store each scenario to the BI360 <i>Data Warehouse</i> and then use BI360 <i>Reporting</i> to output reports/charts.
Break-back analysis	Use one of Solver's special break-back templates (or design your own), to simulate items such as Net Profit and have BI360 auto-generate the related revenues and expenses based on for example historical trends.

Allocations	Create the allocations model in Excel (can use the BI360 report module to create automated, parameter-driven allocation calculations), use BI360 <i>Planning</i> to store the allocated numbers to the BI360 <i>Data</i> <i>Warehouse</i> . From there, use the allocations in BI360 reports or transfer the allocated numbers back to the ERP system.
 Electronic data entry of: Statistics Key Performance Indicators (KPIs) Other organizational information 	Create input forms with Excel for any text or numerical information and use <i>BI360</i> <i>Planning</i> to store it to the BI360 <i>Data</i> <i>Warehouse</i> . From there create reports with the BI360 <i>Reporting</i> , create OLAP cubes for analysis, or transfer the collected data to other systems.
Commentary input for: • Monthly actual/budget variance reports • Scorecard KPIs • Quarterly/Annual reports	Add comment columns to your Excel-based financial statements or scorecards (e.g. generated dynamically with BI360 <i>Reporting</i>), and use BI360 <i>Planning</i> to save the text to the BI360 <i>Data Warehouse</i> . Use BI360 Reporting to create reports that pulls the text comments back up from the database.

Typical Implementation Process

A typical implementation process of BI360 *Planning* would look like this (in this example assume that BI360 *Planning* is being used for budgeting):

- 1. Train power users.
- 2. Identify budgeting needs.
- 3. Preparation of the BI360 Data Warehouse.
- Populate the *Data Warehouse* with historical figures and dimensions from your General Ledger (and if needed also from systems like Payroll and Fixed Assets). Examples of dimensions are: Account, Department, Project, Product, Employee, Asset, etc.
- 5. Enter any other dimensions or historical data directly in the *Data Warehouse* if they do not already reside in another database from which they can be imported.
- 6. Preparation of Excel budget templates.
- 7. Design your budget templates (or use BI360's pre-defined templates) with BI360 *Reporting*. In most cases, you can also use other Excel add-ins to produce your Excel templates. They will now be dynamic

and parameter driven based on the dimensions and data you populated to the *Data Warehouse* in step 1 above.

- 8. Manually created Excel budget workbooks will also work. As a best practice, make them as structured as possible and make sure all dimensions are listed somewhere.
- 9. *Planning* Setup (as explained earlier in this section).
- 10. Train end users.

Dashboard

BI360 – Dashboard Module

Thanks to its capability to calculate and store metrics and KPIs, BI360 can support any dashboard product that can read from a Microsoft SQL Server database (or OLAP cubes if those are created on top of the BI360 *Data Warehouse*). The BI360 suite offers its own dashboard module as well. It is based on Microsoft Silverlight and can run as a stand-alone in your web browser or it can be deployed as part of a Microsoft SharePoint portal.

The main purpose of BI360's *Dashboard* module is to support an organization's performance management initiatives by offering a relatively low cost and easy to implement, professional dashboard environment.

Some of the key reasons for implementing BI360's *Dashboard* module are:

- 1. Get managers and users focused on key metrics that drive organizational performance and processes.
- 2. Very user-friendly Microsoft-Silverlight-based interface that usually does not require training for end-users.
- 3. Relatively low cost and easy to implement.
- 4. Business users can quickly learn how to build custom dashboards and scorecards using the Dashboard Designer

As seen in the architecture diagram earlier in this document, the *Dashboard* module is one of four BI360 modules. The other three modules are:

- 1. BI360 *Data Warehouse* Optional, this is for support of reporting across multiple ERP systems, etc.)
- 2. BI360 Planning module Optional
- 3. BI360 *Reporting* module Optional

In the architecture diagram earlier in this document, you saw how the BI360 *Dashboard* module interacts with the other BI360 modules. In

essence, it serves as a dashboard on top of the BI360 *Data Warehouse*, and the data can come from any source system. It can also connect to OLAP cubes or directly to transactional systems.

User Interface Overview

Below is a sample screenshot of a BI360 *Dashboard*. Using the Dashboard Designer, the power user can create virtually any kind of dashboard layout. The interface was designed to maintain a similar look and feel to the Office products, including the use of an Outlook-style menu bar as the main navigation component.





System Requirements

Depending on the components installed, the BI360 *Data Warehouse Manager* can be implemented on the same SQL server as the ERP. However, if warehouse reporting is intended to offload reporting directly from the ERP, it is recommended that a dedicated SQL server be used for all scenarios.

Performance metrics should be established once a baseline system is configured. There are many factors that contribute to the performance of the BI360 solution, including number of concurrent users, number of transactions, size and design of reports and input forms. For more details, please contact a certified BI360 partner or Solver.

Data Warehouse(Budget Database) SQL Server							
	1-50 Concurrent Users						
Operating System	Windows Server 2008 [Any edition], (32-bit or 64- bit ¹) Windows Server 2003 [Any edition], (32-bit or 64-	Same					

	bit ¹)	
Database	SQL Server 2012 [Any edition], (32-bit or 64-bit ¹) SQL Server 2008 [Any edition], (32-bit or 64-bit ¹) SQL Server 2005 [Any edition], (32-bit or 64-bit ¹)	Same
Processor	Two dual core /expandable	Single/dual quad-core - Expandable
Ram	4 GB or greater	6 GB or greater
Hard Disk	Based on data integrated into warehouse and/or transactional input within Budget model (contact Partner)	Same
Disk Configuration	RAID 10 is recommended, Raid 1 minimum recommended.	Same

Notes:

1. 64-bit recommended

BI360 – Client Components									
	Reporting ¹	Planning	DWM	Dashboard					
Operating System	 Win XP SP 3 Win Vista Win 7 (32-bit or 64-bit) Win Server 2003 (32-bit or 64-bit) Win Server 2008 (32-bit or 64-bit) 								
Ram	• 2 GB (4GB Recommended) ⁵								
Microsoft Office	 Excel 2003² Excel 2007 (32/64 bit)⁴ Excel 2010 (32/64 bit)⁴ 	<same <sup="">3</same>	N/A	N/A					
.NET	• 3,5 SP1 or newer		<u>.</u>						
Other software requirements	Visual Studio Tools for the Office system 3.0 Runtime Service Pack 1	<same< th=""><th>Not Required</th><th> Silverlight 3.0 or higher Explorer 7 or higher, Firefox 2 or higher, Safari 3 or higher </th></same<>	Not Required	 Silverlight 3.0 or higher Explorer 7 or higher, Firefox 2 or higher, Safari 3 or higher 					

Notes:

- 1. Reporting will install a small repository database on the same server as the ERP and/or Warehouse.
- 2. Reporting Designer requires Office Excel 2007 or later.
- 3. BI360 Planning Administrative features require Office Excel 2007 or later. The end user input will work on Excel 2003. The installed Office version will be detected during the BI360 Planning install.
- 4. Drill Anywhere runs on 32bit Win OS/Excel only. For more details on Drill Anywhere or Excel Dicer, please contact Solver.
- 5. Terminal Server deployments: Please contact your BI360 partner or Solver for assistance.

BI360 Dashboard Web Server						
Operating System	 MS Windows Server 2008 [Any edition], (32-bit or 64-bit). MS Windows Server 2003 [Any edition], (32-bit or 64-bit). 					
IIS • IIS 6 or 7						
Processor	• 1.5 GHz or faster, 2.0 GHz dual-core recommended					
Ram	• 2 GB or more, 4 GB recommended					
Hard Disk	 Installation directory: 30 MB Web application: 5 MB 					





BI360 – Data Warehouse

BI360 *Data Warehouse Manager* (DWM) is BI360's configurable data warehouse solution. The DWM database is fully integrated with BI360's Excel-based *Reporting*, *Planning* & *Dashboard* modules. This enables immediate access to advanced financial and operational reporting capabilities as well as fully customizable budgeting, forecasting and data entry features. The *Data Warehouse Manager* is delivered with a simple to use desktop interface where administrators can configure and manage the *Data Warehouse Manager* without the need for technical skills.

Data Warehouse Interface (Admin Console) Overview

The *Data Warehouse Manager* interface has been designed with the latest Microsoft technologies and it follows Microsoft best practices for interface design and usability. The interface was designed to maintain a similar look and feel to the Office products, including the use of a ribbon and Outlookstyle menu bar as the main navigation components.

Data Warehouse Manager Components

Modules

The BI360 *Data Warehouse Manager* database has been pre-designed with a number of popular financial modules as well as five generic (userdefined) modules for any data type. However, the design is not limited to financial modeling; non-financial modeling can be handled just as well. Pre-configured modules include:

- General Ledger
- Accounts Payable
- Accounts Receivable
- Capital (Asset Information)

- Payroll
- Projects
- Revenues
- Other X (Generic Modules 1 5)

To dramatically reduce implementation time, each module contains predefined fields specific to that module/area. The modules can be renamed using the Data Warehouse Manager's interface. For each module and without any need for customization, an unlimited number of User Defined Fields can be added using the Admin Console to further extend the capacity of the transaction tables.

Dimensions

The BI360 *Data Warehouse Manager* database contains dimensions specific to each module but also contains ten "Free" dimensions which can be shared across modules. Using the *Data Warehouse Manager* interface, the dimensions can be mapped to each module and labeled as needed. Each dimension contains many attributes specific to the dimension. Each of the ten free dimensions also contains several User Defined Fields (UDFs) to customize the dimensions to match a customer's specific data model. Up to 40 User Defined Fields can be added to each dimension using the Admin Console.

Dimension Management

Dimensions can be managed within the *Data Warehouse Manager* interface. There are many ways to load the dimensions into the *Data Warehouse Manager*:

- 1. Direct import via the *Data Warehouse Manager* Interface import feature from a CSV file.
- 2. Automated SQL Server Integration Services (SSIS) integration import.
- 3. Manual entry: Manual entry is specifically useful for budgeting and forecasting. Since *Data Warehouse Manager* is a warehouse, Accounts and other dimensions that do not exist in the source system(s) can be created and maintained directly within the *Data Warehouse Manager* environment.

-			BI360 Data Wareho	use							×
Info Home											6
New Delete Import Export	Pri	nt Properti	ies								
Dimensions	<	Account [Dimension								
Accounts Payable	-	Code	Description	Alias	Account	tType	DebitCredit	AccountCategory	Capital Life	Active	
Accounts Receivable			= -	-	=	+	= -	= -		-	
🚞 Capital		10100	Cash		BSC	-				X	
🚞 Currency FX		11100	Accounts Receivable		BSC	-					
🦲 General Ledger		13000	Prepaid		BSC	*				V	
Category	=	13100	Other Current Assets		BSC	-				V	
Department		16100	Long-Term Account Receivables		BSC	-				V	
Source System		> 17100	Other Long-Term Receivables		BSC	-					
Entity		17300	Goodwill		BSC	-				V	
Scenario Period		18100	Building		BSC	•			30.00000	×	
Payroll		18200	Equipment		BSC	*			5.00000	V	
Directs	-	18300	Computer		BSC	*			3.00000		
A		18700	Accumulated Depreciation Building		BSC	*				V	
Dimensions		18800	Accumulated Depreciation Equipment		BSC	*				V	
		18900	Accumulated Depreciation Computer		BSC	•				V	
Processes		19100	Other Long-Term Assets		BSC	*				×	
Currency		21100	Accounts Payable		BSC	*				×	
		23100	Other Short-Term Debt		BSC	*					
Trees		24100	Long-Term Account Payables		BSC	*					
		25100	Other Long-Term Debt		BSC	*					
👼 Data Explorer		30000	Currency Translation Adjustment		BSC	*					
10		31000	Retained Earnings Product Revenue		BSC	•				X X	
Administration		40010	Services Revenue		PLC PLC	• •				× ×	
	+	40020	Services Revenue Maintenance Revenue		PLC	* *				× ×	
		40030			PLC	÷					
onnected to : { BIB60DW_CORPD	ЕМО	0_36 }									

Currency

The *Data Warehouse Manager* contains advanced currency features which work with the Business Rules to calculate foreign currency exchange. The tables accommodate daily rates, which can be loaded from a source system via SSIS integration or monthly rates can be directly entered into the *Data Warehouse Manager* interface. By combining several advanced backend features, the end user experience is very simple, yet advanced. The currency conversion process has several elements.

- 1. Currency Codes: Currency codes can be manually entered, directly imported or imported via SSIS integration.
- 2. Rate Types: Rate types can be manually entered, directly imported or imported via SSIS integration. Rate types such as Average, Closing, Historical, and Budget are typical.
- 3. Daily\Monthly Rates: If daily rates are loaded into the system, monthly rates can be automatically calculated. Otherwise, monthly rates can be manually loaded or imported.

Copyright Solver, Inc., 2012

4. Rate Configuration: The configuration ties together the Account type, such as Balance Sheet, the Scenario such as Actual and the Rate Type such as Average.

		BI360 Data \	Warehouse				- 0 X	
Info Currency							۵	
New Delete Import Export Pri		ulate Show/Hide M tes Effective Rates Tasks	ultiply/Divide Rates					
Currency < Monthly Exchange Rates								
Currency Codes	Rate Type AV	G - Average	*					
🛄 Rate Types		Jan 2009	Feb 2009	Mar 2009	Apr 2009	May 2009	Jun 2009	
Monthly Rates	CAD	0.815461143	0.803148341	0.790826414	0.816993464	0.86888522	0.887705282	
Daily Rates	EUR	1.323234408	1.278375134	1.303843862	1.319621585	1.36275429	1.401535803	
🛅 Rate Configuration	HKD	0.128921967	0.128971184	0.129026598	0.129027097	0.129014446	0.129019939	
	USD	1	1	1	1	1	1	
Dimensions Processes Currency Trees Data Explorer								
Mdministration	4 m							
Connected to : { BI360DW_CORPDEMO)_36 }							

Data Explorer

The Data Explorer is used to filter, group and display data for each *Data Warehouse Manager* module so that an administrator, at any time, can see and validate what is stored in the *Data Warehouse Manager* without the need to perform SQL queries or running reports. The data can be viewed within the *Data Warehouse Manager* interface or exported to Excel.

ll ▼ Info Home					BI36	0 Data Ware	house						
) 🗙 🛃 🗙 (roperties											
ata Explorer 🔇		Transacti	on Viewer	Piv	vot Vie	wer							
Accounts Payable Accounts Receivable Capital	Pe	riod 2	01201,2012	202,2012	03,201	204,201205,	201206,2012	207,201208,20	0120 🔍	📀 Refre	esh	Show	Amount Totals
Currency FX		Dep	artment 🔺										
Payroll Projects			sactionID		[escription	Scenario	Category	Entity	EntityCorr	Currency	Account	Department
📝 Revenues	1	љ <u>–</u>	•	= -			= -	= -	= -	= -	= -	<u>A</u> a 🔻	<u>A</u> a •
	Ð	000 (194	0 items)										
			7 items)										
			i5 items)										
)7 items)										
		400 (920											
Se Dimensions		500 (977											
Processes		1343		01/15/2			ACTUAL	MAIN	SAS		HKD	60010	500
-9 Hotesses		2166		01/30/2			ACTUAL	MAIN	SAS		HKD	60010	500
Currency		13440		01/15/2			ACTUAL	MAIN	SCA		CAD	60010	500
		2167		01/30/2			ACTUAL	MAIN	SCA		CAD	60010	500
Trees		1345		01/15/2			ACTUAL	MAIN	SEM		EUR	60010	500
🗟 Data Explorer		2168		01/30/2			ACTUAL	MAIN	SEM		EUR	60010	500
Data Explorer		1346		01/15/2			ACTUAL	MAIN	SUS		USD	60010	500
Administration		2169)	01/30/2			ACTUAL	MAIN	SUS		USD	60010	500
				02/01/2					SAS		HKD	60010	500
•		-						add thi	SC 0			BOILT	500

Processes

The Processes are SQL procedure-based rules which enable data transformation and movement within the *Data Warehouse Manager*. Because the rules are created as stored procedures within the *Data Warehouse Manager*, the rules can be accessed and/or scheduled using external tools, such as the SQL Server Agent. Rules can be loaded into the warehouse via an XML file. The XML file contains information about the parameters used to execute the rule as well as the SQL for the data manipulation.

Practical applications of rules include:

- Currency conversion
- Data aggregation
- Entity eliminations
- Allocations
- Data movement
- Other transformations

Info Rules New Processes Rules Rules Name Description Category Module Creates Rules Name Description Category Module Creates Rules Name Description Category Module Creates Monthly_FX_Translati Monthly Currency Cc General Ledger 01/13/2 CopyVersions Copy scenarios from General General Ledger 02/14/2 Processes Image: CopyVersions Copy scenarios from General General Ledger 02/14/2 Processes Image: CopyVersions Copy scenarios from General General Ledger 02/14/2 Processes Image: CopyVersions Copy scenarios from General General Ledger 02/14/2 Processes Image: CopyVersions Copy scenarios from General General Ledger 02/14/2 Processes Image: Currency Fulle Explorer History Image: Currency General General Ledger States Processes Image: Currency Fulle Explorer History States							
Rules Name Description Category Module Create Aa CopyVersions Copy scenarios from General Ledger 02/14/2 CopyVersions Copy scenarios from General Ledger 02/14/2 It Explorer History It Explorer History It Explorer History It Explorer							
Image: Solution of the second seco							
Image: Angle State Monthly_FX_Translati Monthly_EX_Translati Monthly_EX_Translati Monthly_EX_Translati CopyVersions	ed On Created By						
CopyVersions Copy scenarios from General General Ledger 02/14/2 CopyVersions Copy scenarios from General General Ledger 02/14/2 CopyVersions Rule Explorer History History This rule cannot be edited. IF EXISTS (SELECT * FROM sys.objects WHERE object_id = OBJECT_ID(N'[dbo].[rules_Monthly_FX_Translation] Corrency IF EXISTS (SELECT * FROM sys.objects WHERE object_id = OBJECT_ID(N'[dbo].[rules_Monthly_FX_Translation] GO SET ANSI_NULLS ON SET ANSI_NULLS ON	т <u>А</u> а т						
Image: Dimensions Rule Explorer Processes This rule cannot be edited. IF EXISTS (SELECT * FROM sys.objects WHERE object_id = OBJECT_ID(N'[dbo].[rules_Monthly_FX_Translation] GO SET ANSI_NULLS ON	/2012 Generic Demo Mode						
Processes If EXISTS (SELECT * FROM sys.objects WHERE object_id = OBJECT_ID(N'[dbo].[rules_Monthly_FX_Translation] Currency IF EXISTS (SELECT * FROM sys.objects WHERE object_id = OBJECT_ID(N'[dbo].[rules_Monthly_FX_Translation] Trees SET ANSI_NULLS ON	/2012 BI360_CORP_DEMO_						
Processes IF EXISTS (SELECT * FROM sys.objects WHERE object_id = OBJECT_ID(N'[dbo].[rules_Monthly_FX_ DROP PROCEDURE [dbo].[rules_Monthly_FX_Translation] GO SET ANSI_NULLS ON							
Trees SET ANSI_NULLS ON CONTRACT ON							
SET QUOTED_IDENTIFIER ON GO	K_Translation]') AND type in (N'P',						
CREATE PROCEDURE [dbo].[rules_Monthly_FX_Translation] @vScenario nvarchar(50), @vEntitv nvarchar(50).	-						

Practical Usage Examples

The following are examples of various practical uses of BI360's *Data Warehouse*:

Data Warehouse	Usage Examples
Multi-source consolidation warehouse	 Load data from any ERP Load data from external system (ADP, FAS, Ceridian, etc.) Store multiple scenarios of budgets & forecasts via <i>Data Warehouse</i> Manager or any other Interface.
Instant Excel-based reporting, budgeting and modeling with BI360's <i>Reporting</i> and <i>Planning</i> modules	• Data Warehouse Manager pre-defined integration to Reporting and Planning enables immediate reporting and budgeting to/from all Data Warehouse Manager modules.

Dimension & Data Integration	 File Import/Export capabilities for all dimensions and data modules. SSIS Import/Export capabilities to and from any system accessible via SQL Server Integration Services.
Customize Modules, Dimensions & attribute names	 Labels can be modified to fit with the organizations requirements and to enhance the BI user experience.
Exchange Rate Administration	 Import Daily spot rates from ERP via SSIS Automatically calculate monthly Average and closing rates as well as any other user configured rates. Enter monthly rates directly into the database (when daily rates are not loaded)
Business Rules Engine	 Complex currency calculations for multi- national organizations
Data Viewer	 Filter, view and validate data loaded into each module.

Typical Implementation Process

A typical implementation process of the *Data Warehouse Manager* would encompass the following:

- 1. Installation of the *Data Warehouse Manager* pre-configured SQL Server database.
- 2. Installation of *Data Warehouse* Manager desktop application on the administrators' machines.
- 3. Determine source systems and data to be integrated with each system and map to *Data Warehouse Manager* modules and dimensions (see below).
- 4. Configure *Data Warehouse Manager*'s Period table and customize Module names, dimension names as requisite (or leave default). *Note: Custom labels will also be visible in Reporting, Planning & Dashboard modules.*
- 5. Configure currency (as requisite). Determine if daily rates will be loaded from external system or monthly Average/Closing, etc. rates will be loaded directly into the currency table. If it set up as an automated process, then configure the SSIS Integration to update daily spot rates in the *Data Warehouse Manager*.
- 6. Determine import method (automated with SSIS or manual imports):
- 7. SSIS: Configure SSIS packages [E.g. to Microsoft Dynamics] to previously determined module & dimension mappings. Or implement custom data integration to any data source.

- 8. Import: Manually import dimensions and data
- 9. Business Rules: If business rules, such as Currency Conversion are to be used, ensure step #5 is completed. No additional steps required, other than running rule.
- 10. Document integration decisions.
- 11. Quick training required (Walk through can be done in 1-2 hours).

Module – Dimension Mapping

Each module contains module specific dimensions as well as ten free dimensions, which can be shared across all modules. Each module also contains several other standardized dimensions, e.g. Entity, Currency, Scenario, Category, etc. Additionally, each module contains many attributes, both system defined and user defined, including three amount fields, four text comment fields as well as other module specific attribute information.

Module	Unique Dim	Unique Dim	Unique Dim
General Ledger	Account		
Account Payable	Vendor		
Accounts Receivable	Customer	Sales Person	
Capital	Asset		
Human Resources	Employee		
Project	Project	Customer	Item
Revenue	Customer	Product	Sales Person
Other	Other		