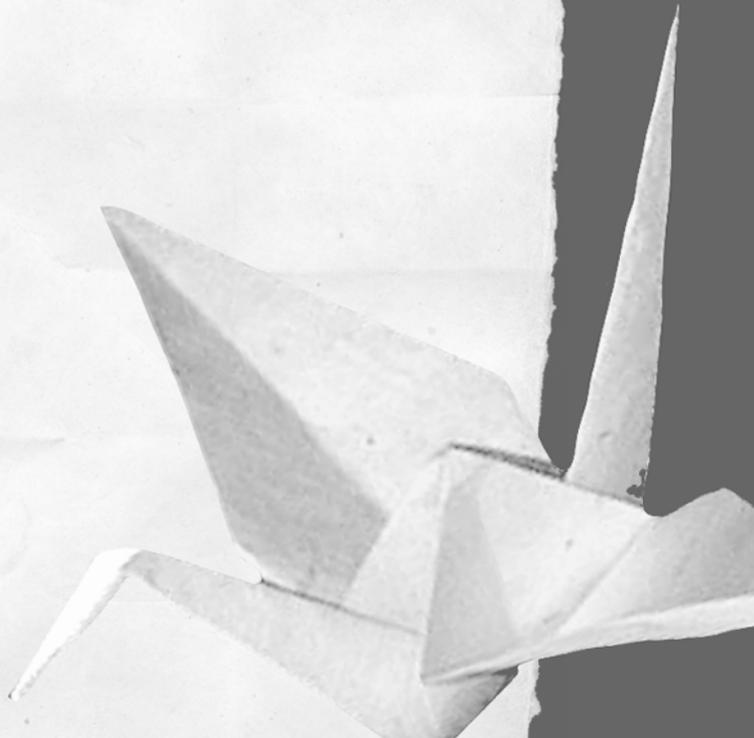


Configuration Tools

FEATURING:

- ▶ **INTRODUCTION TO SAP**
- ▶ **A NEW APPROACH TO SYSTEM CUSTOMIZATION**
- ▶ **THE IMPLEMENTATION GUIDE (IMG)**
- ▶ **THE SERVICE MARKET PLACE**

COPYRIGHTED MATERIAL



Before you undertake your first SAP configuration project, it is important to understand the concepts behind table-driven customization as well as some of the tools, tips, and tricks that can be used. The purpose of this chapter is to provide the foundation for successfully carrying out SAP Finance and Controlling (FI/CO) configuration. We'll give you an overview of SAP, talk about the new approach to system configuration, discuss how to use the implementation guide, and introduce you to the Service Market Place.

Although this chapter is a must-read for people new to configuration, configuration “old-timers” can also pick up a trick or two from reading it.



NOTE The terms *customization* and *configuration* are used interchangeably throughout this book.

Introduction to SAP

SAP stands for Systems, Applications, and Products in Data Processing. Founded by five German engineers in 1972, SAP is the world's leading provider of business software, offering applications and services to companies of all sizes across more than 25 industries. SAP offers an integrated system, which means that all SAP modules are designed to share information and automatically create transactions based on various business processes.

SAP Products

SAP has slowly evolved in terms of its product offerings. You will still come across SAP consultants who refer to the SAP system as SAP R/3, but as SAP's product offerings have broadened, the reference to R/3 has been dropped. R/3 initially referred to SAP's only product, the Enterprise Resource Planning (ERP) system, but today SAP offers a host of products of which SAP ERP Central Component is the heart (often referred to as SAP ECC). The SAP ERP Central Component is where the original ERP (R/3) functionality is housed, and it is where all the data processing/business process transacting takes place. This book is focused on the Financial (FI) and Controlling (CO) modules found in SAP ECC version 6.0.

The goal of this book is not to teach you how to implement one specific solution but to teach you how to configure the SAP system. Attempting to cover every possible

configuration scenario you might encounter would be an impossible task, but after reading the book, you will be able to apply what you have learned and configure your system based on your business requirements.

SAP has now introduced many areas of functionality from its data warehouse—often referred to as its *business warehouse* (BW) or, now more correctly, *business intelligence* (BI)—which includes a host of reporting tools and functionality, not limited to business objects.

SAP also offers the following software suites:

- ▶ Supplier Relationship Management (SRM)
- ▶ Strategic Enterprise Management (SEM)
- ▶ Catalog Content Management
- ▶ Compliance Management for SOA
- ▶ Supply Chain Management (SCM)
- ▶ Product Lifecycle Management
- ▶ Customer Relationship Management (CRM)

SAP Terms

Now that you understand how the different SAP products break down, you'll need to become familiar with some common terms that explain different parts of the SAP system; you will see the following terms used throughout the book:

ABAP (ABAP/4) ABAP/4 stands for Advanced Business Application Programming/4th Generation Language. SAP is coded in ABAP. ABAP is also used for extensions and extra programs that are written for SAP. ABAP is similar to other fourth-generation languages and is a first cousin of COBOL, without the JCL.

Basis Generally, SAP projects, and the folks who work on them, are lumped into two groups—technical and functional. The technical system includes ABAP, database administration, transport management, security, authorizations, and so on. Basis is a subset of the technical group and consists of the folks who take care of all technical components of the system except for ABAP. The Basis group, in more common terms, consists of your project database administrators (DBAs) plus more.

Variant A variant is a specific setting that is saved when a program is executed. Some data input screens allow you to save and execute variants. Variants can also be created in the program maintenance screen of the program. Using variants is a good way to save time because they allow you to execute a routine transaction without having to enter all of the parameters needed by the program every time.

Menu pathSAP, like most client/server applications, utilizes menus to allow a user to navigate through the system. When we refer to or list menu paths in the book, we are starting from the root menu and progressing down through each menu hierarchy to reach the needed transaction. When we refer to only the menu path, we are talking about the Implementation Guide (IMG) menu path. SAP application menu paths are explicitly noted.

Transaction code A transaction code (tcode) is generally a four-character code (later versions of SAP have introduced longer tcodes) that is entered in the command field on the toolbar. Transaction codes are not case sensitive. SAP provides two ways of executing a transaction, via a menu path and a transaction code.



NOTE Although it may be tempting for you to use tcodes to get to a specific screen/transaction, it is useful to use the menu paths. The benefit is that you can get a better grasp of how SAP has logically laid out the options in configuration and the functionality in the functional menu paths. SAP has developed a large array of options to cater to almost any imaginable business scenario, and using the menu paths will allow you to see these options, which results in a broader set of options.

It is important to note that, unless you are at the main SAP menu or the main menu of a submodule such as G/L, it is necessary to include /N or /O before the transaction code in order to execute a transaction in a different module. For example, if you are currently in the Cost Center accounting module in the screen used to create cost centers and you want to enter a G/L document (transaction code FB01), you must enter /NFB01 or /OFB01 to execute the transaction. /N takes you back to the root menu and then executes the transaction code. /O opens up a new session and then executes the transaction code. Remember, you can have only six open sessions of SAP at once.



TIP As stated earlier, unless you are at the main SAP menu, or a submodule main menu, it is necessary to include /N or /O before a transaction code in order to execute a transaction in a different module.

Parameter ID A parameter ID is a special identifier given to some fields in SAP. It can be stored in your user profile with its default values. For example, the parameter ID for company code is BUK. A user who is responsible only for entering documents in company code 1000 would set up the BUK parameter ID with a default of 1000 in their user profile. By specifying this parameter ID, the user will never have to enter the company code in a transaction; the company code will automatically default to 1000. Parameter IDs are stored in the Technical Information field box. An explanation of how to display the Technical Information box is included in “Finding the Table to Configure” later in this chapter.

Batch input session A batch input session stores values to be entered during a normal system transaction. Some transactions automatically create batch input sessions because of the heavy processing required. To complete the transaction, you must select the batch input session and then run the batch input session manager. Most data transfer programs are executed via batch input sessions. A good way to think of a batch input session is to think of it as a macro. A macro uses standard functioning to input data that is stored to automate a repeated task. You can use transaction code SM35 to run and manage batch input sessions.

Jobs A job is similar to a batch input session in that it executes a standard SAP transaction in the background, usually at night. Jobs are set up and scheduled for processor-intensive transactions and reports. If you do not correctly specify the print parameters on a print request, your print request will be stored as a job. This means that when you start a print transaction from within SAP and you do not check the Print Immediately box the print request is stored in the print spool as a job and has to be manually released through the job manager to print. Your company’s Basis group usually manages jobs.

User menus You can create your own user menu with your most commonly used transactions. Then you can assign this personalized menu to your user ID in your user preferences. If you are developing a system to be used by a client site, user menus can also be set up for a group of users with limited access to the system. This includes users who might not use the system often enough to remember the menu paths they need to use to execute a transaction.

Distributed systems (ALE) Some SAP installations have more than one productive instance of SAP running at any one time. SAP provides a tool called Application Link Enabling (ALE) to allow two different SAP systems to share data with each other.



TIP SAP is an integrated system, which means configuration choices and decisions made by people configuring other models in the system can have an impact on FI/CO. Any business transaction that has a financial impact will have an integration point with FI/CO; therefore, the FI/CO team is often central to all design reviews.

A New Approach to System Customization

For many years, organizations struggled with extremely long project timelines in order to develop information systems that met their specific requirements. Most IT projects used structured development methodologies that were very unforgiving in terms of missed or changing business requirements. The development of custom code was a tedious process requiring armies of programmers as well as significant end-user involvement.

The project timeline was also extended because often business owners didn't know what they wanted until they saw it, which led to what is commonly referred to in the IT industry as *analysis paralysis* in projects. Upon project completion, large IT staffs needed to be retained to maintain the custom programming and to update the programs with requirements that may have changed during the long development cycle. Numerous companies also had departmentalized systems, which oftentimes did not share information. These numerous departmental systems became "information silos" within the organization. Separate systems per function and/or department can lead to inconsistent results.

These disparate and numerous systems also created the need for many distinct interfaces between systems that were not designed to talk with each other. Despite the interfaces, the systems would never be integrated. Worst of all, accounting systems were updated with financial data by means of batch programs. Batch programs are run on a fixed schedule, generally daily, weekly, or monthly, which means that the data is never current.

To fulfill the new requirements of information systems, a new breed of software systems, now called *Enterprise Resource Planning (ERP)* systems, was created. ERP systems provide a single source of data with designed integration between different functional modules (for example, Accounting, Sales and Distribution, Materials Management, Production Planning, and so on) to take full advantage of an enterprise's stored information. A common set of source code was needed for these packages so that changes in technology could be rapidly introduced via upgrades to the

programs. To facilitate these requirements, a new way of customizing systems was needed. This new way of customizing systems is known as *table-driven customization*, or *configuration*. By configuring the system via tables rather than changing source code, you now have a very clear logical approach to managing your system.

Table-driven customization allows for rapid changes in business requirements with a common set of source code or programs. The common programs are coded to focus on settings in specific tables to make the programs react in various ways to fit different business needs. This is what makes ERP systems, and SAP in particular, so flexible—there are more than 10,000 tables in the SAP database structure! Because table settings instead of old-fashioned hard-coded program logic are what drive program functionality, new and changed business requirements can be rapidly implemented and tested in the system. Table-driven configuration (customization) is at the heart of what the functional SAP consultant delivers.

So that you can benefit from the power of SAP, a careful analysis of your company's current business processes is in order. SAP has industry-specific best business processes to take full advantage of the most efficient business and technological processes.

When using SAP ECC, you can use two menu paths: the user menu and the Implementation Guide (IMG). When you log on to SAP, the first screen and menu path you are presented with is the user menu. The user menu is where end users will be spending all of their time. This is where you would use SAP's functionality, such as creating a vendor invoice or journal entry.

The IMG menu is specifically for configuring the system so that an organization can tailor it to address the needs unique to its business.



TIP You can find the configuration menu by using the following menu path or transaction code: Tools > Customizing > IMG > SPRO – Execute Project. The command window in the upper-left side of the SAP screen will allow you to simply enter the transaction code—SPRO—to get you directly to the configuration menu.

Many companies use the implementation of SAP as an opportunity to reengineer their entire business and develop the most efficient processes available. SAP has invested a great deal of time and money into delivering the best business practices for almost all sectors of the economy. It has industry solutions for the following:

- ▶ Banking
- ▶ Defense and security

- ▶ Healthcare
- ▶ Higher education and research
- ▶ Insurance
- ▶ Public sector
- ▶ Manufacturing
 - ▶ Aerospace and defense
 - ▶ Automotive
 - ▶ Consumer products
- ▶ Service
 - ▶ Media
 - ▶ Professional services

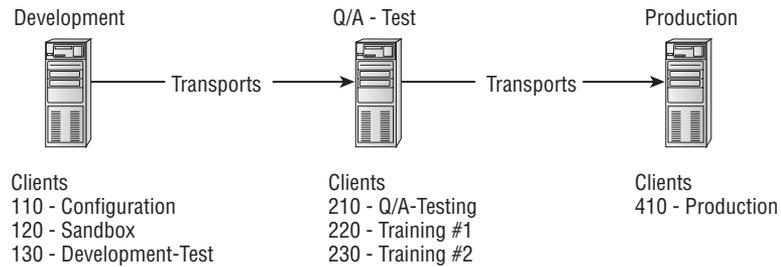
For small companies that have neither the time nor the resources to undertake a full business process reengineering project, SAP has a very well-researched and well-designed implementation methodology that will help with business process optimization efforts. This, along with the Industry Solutions (ISs), will allow for you to utilize the software to ensure a more effective and efficient company.

For smaller companies, there are preconfigured clients with many templates and industry-standard reports.

SAP System Environment

It is very important for everyone on the implementation project team to understand the SAP system environment used on the project. A system environment is referred to by some as an *instance*. Others will sometimes refer to a client as an instance. In this book, both *environment* and *instance* are used interchangeably to refer to different systems, such as development, quality assurance/testing environment, and production. In some cases, an SAP term may have, or may seem to have, more than one meaning, depending on which part of the system you are working in. One such term is *client*. As defined in the enterprise structure, it means the organization for which SAP is being configured (for example, the XYZ Corporation, or the example corporation used throughout this book, Extreme Sports). When defined in Basis terms (the SAP technical system), *client* means the different installations of SAP used for a specified purpose. Really, these are the same things, but it is difficult to

understand the client concept in this light when you are just starting out in SAP. In the standard project setting, there will be three environments: the development environment, the quality assurance/testing environment (QA), and the production environment. Within each environment there are different clients that are used for specified purposes.



The development environment is where the majority of implementation work takes place. It should have a minimum of three clients: sandbox, configuration, and development testing. The sandbox client is used to test configuration ideas and theories at any time. It is also where all system design work should take place. Once you are comfortable with your configuration solution in the sandbox client, you can re-create your solution in the configuration client. The configuration client is also called the *transport client*. This is where all final configuration that needs to be moved through the testing cycle, and finally into production, takes place. The configuration client has automatic transport recording turned on (covered in the next section, “Transports”). Ideally, the configuration client should also be your “golden” client; that is, no transactions or testing should take place in this client. Once a transport has been created, it should be moved to the development-testing client. Once the configuration is in the development-testing client, the transport should be thoroughly unit-tested. Usually, only unit testing is conducted in the development system; some projects may conduct integration (string) testing in this client as well. Once the transport has been successfully tested, it is ready to move into the QA environment. Normally, all transports for particular projects or rollout phases are moved into QA at one time.

The QA environment is where all final testing is conducted prior to moving transports to the production environment. Normally, this is where integration (end-to-end business process) testing and user acceptance testing (UAT) is conducted. There is a minimum of one QA client that is used to conduct testing. There may be additional clients you can use in the QA environment to test different transactions for

training, data conversion, and user sandboxes. Once the entire project solution has been tested successfully in QA, it is ready to move to production.

The production environment is where all day-to-day business activities occur. This is the client that all end users use to perform their daily job functions. There is usually only one production client per SAP installation. It is very important to move into production only transports that have passed all testing cycles. Inadequately tested or understood changes to the system can lead to production system issues. These production issues generally occur if you, as the configurator, do not fully understand the integration points between the module effecting the change and (generally) FI/CO. Production issues can be as catastrophic as the company's inability to ship goods or post cash.

Transports

Transports are the vehicles by which your configuration settings are moved from client to client and environment to environment. Normally, your configuration client should be the only client that creates transports. Transports in the configuration client are created anytime you make a change to a configuration table or program. This is known as *automatic recording* of transports. The setting to allow for automatic recording of transports is made at the client level in table T000. Although you can make customizing settings in the sandbox, client transports are not automatically created. The sandbox and configuration clients are the only clients in which changes to configuration tables or programs should be allowed.

Without ensuring this level of discipline within the system, several issues with the integrity of the production system can occur. The following are some examples:

- ▶ Redundant versions of a program
- ▶ Incorrect values for a table
- ▶ Incorrect settings for a configuration table

These inconsistencies can result in disruptions to the business and lengthy, often complex recollections between the systems to determine what needs to be backed out. This does not include the corrections to the business results that may be required.

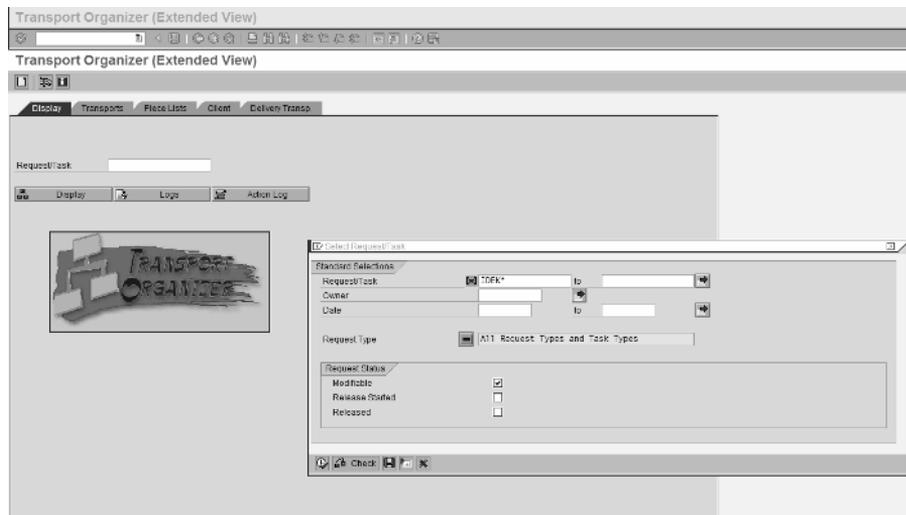


NOTE Your Basis group should be responsible for maintaining and moving transports from client to client. In some small implementation projects, individual consultants are responsible for moving their transports from the configuration client to the development-testing client.



NOTE Most transports record only the changes to the table. However, some transports created by a small number of tables copy the entire table, not just the changes. It is very important that these whole table transports are watched and managed carefully so that only the latest changes are reflected in the target clients. This is especially important once a project is “live” and in maintenance mode. When dealing with whole table transports, you always run the risk of moving into other environments’ configuration that shouldn’t be moved. You can also very easily “leapfrog” transports moved by other developers and overwrite new configuration with old configuration. To avoid the leapfrogging of values in a table, the configuration team should be organized in a way that one person owns a specific area of functionality or business process. This structure should include the review of transports to ensure that common tables are being carefully monitored.

You can use the change request query screen to create a transport automatically in a configuration client. A transport number is assigned automatically, but you are free to add the description yourself.

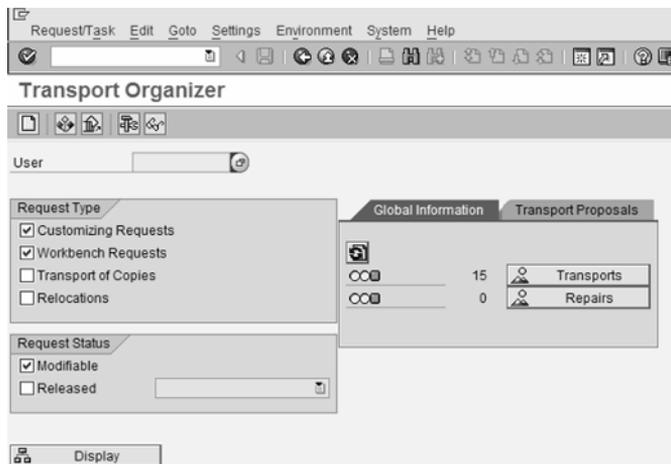


There are two types of transports: *client dependent* and *client independent*. When an environment such as the development environment is created with multiple clients, most objects are copied one for one to be used for each individual client. For example, table T030 contains the settings for automatic account assignment. In the development environment, the sandbox, configuration, and development-testing clients each have its own copy of table T030. Any change to table T030 results in a client-dependent transport—all the T030 tables in the various places reflect the change only after it is transported.

All programs and a small amount of tables are shared among the clients within an environment. These are known as client-independent objects. For example, table (view) V_T021S is client independent, meaning that, when this table is changed in the configuration client, the setting automatically takes effect in all clients in that environment because there is only one V_T021S that is used by all clients in that environment. A change to a client-independent table should be made only in the configuration client; the Basis group controls this setting when it sets up the client. The option to allow client-independent changes is set at the client level in table T000. Sometimes, when testing new design and development in the sandbox, you are required to make a change in the configuration client. It is fine to design in the configuration client as long as you are making only client-independent changes.

Each consultant/developer is responsible for keeping track of their individual transports. Transaction code SE10 allows you to view and manage all transports you have created. You can also view transports created by other developers. This transaction allows viewing of only modifiable (unreleased), only released, or both released and unreleased transports. The default is set to modifiable (unreleased) transports. Figure 1.1 shows the initial Transport Organizer screen.

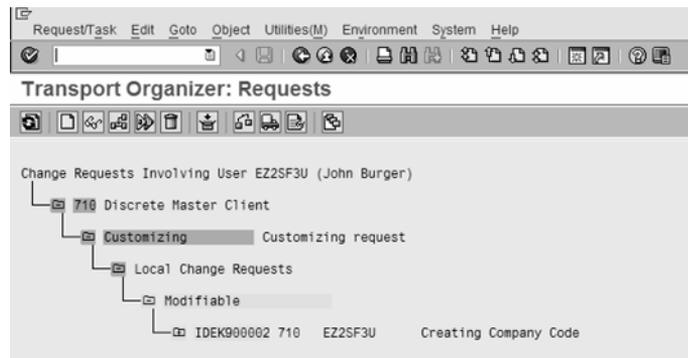
FIGURE 1.1 You can use transaction code SE10 to view and manage transports on the Transport Organizer screen.



Once you are ready to unit-test your configuration, the related transports must be released so that the changes can move from the configuration client to the development-testing client. It is normally the responsibility of the consultant/developer to release their own transports and let the Basis group know that they are ready to move via the procedures set forth in their project. This is an implementation

activity, and this transport movement can happen once the system is in production. Remember that it is only necessary to release and move client-dependent transports to the development-testing client. Client-independent transports are already reflected in all clients in the environment. Client-independent transports need to be released and moved only when sending changes from environment to environment. Figure 1.2 shows the listing of transports created by an individual consultant or developer. This screen is obtained by proceeding through the screen displayed by transaction code SE10 shown earlier in Figure 1.1.

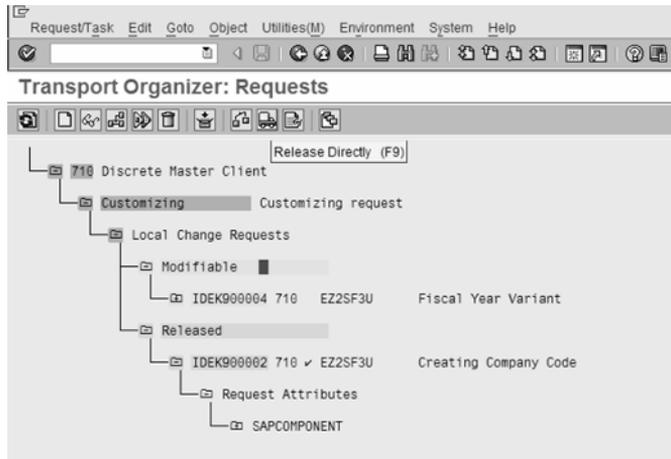
FIGURE 1.2 The transports “owned” by John Burger



Each transport in the list (as shown in Figure 1.2) can be expanded to show the tasks included in it. Each transport has at least one related task assigned to it. The tasks actually contain the table changes. The upper-level transport acts only as a container for these tasks. When you release transports for the Basis group, you must release the individual tasks related to the transport before releasing the upper-level transport. It is important to note that each task always has objects attached to it. It is not necessary to release objects to the Basis group, only the tasks themselves. This is because the tasks carry the objects with them. When you release a task, a snapshot is taken of the key that is contained in the task. The configuration that resides in the snapshot is what is written to the operating system to be transported. It is very important to understand this timing and how it affects your transports if other consultants are configuring the same key. For example, consultant X makes a change to object key 123 on May 1st; on May 30th, consultant Y makes a change to the same object key 123. When consultant X releases his task on June 1st, it contains the changes that consultant Y made on May 30th, not the changes that consultant X made on May 1st. As you can see, communication is a key success factor on any SAP project. Transports are released using the same screen that is generated by following

transaction code SE10, selecting the task, and clicking the Release Directly button, as seen Figure 1.3. When you release a task, it is released (copied) to its corresponding transport request. The transport request is then released for export. The transport request is what is actually moved between clients and environments.

FIGURE 1.3 Releasing a transport



The Implementation Guide (IMG)

The Implementation Guide (IMG) provides step-by-step details on the configuration settings that need to take place in each module of the SAP system. The IMG is grouped by functional modules and the business processes that occur in each module, as shown in Figure 1.4. It provides the front end to the customizing tables as well as explanations of the functionality affected by each table. In the design and development phase of a project, consultants and developers spend a majority of their time in the IMG.

The IMG can be displayed in three different views: the SAP Reference IMG, the Enterprise IMG, and the Project IMG. The SAP Reference IMG comes with your installation of SAP. It contains all components for all modules of SAP. The Enterprise IMG is generated from the SAP Reference IMG. The Enterprise IMG normally contains all modules and their related business processes for your specific instance of SAP and the countries being implemented. It is usually a safe bet that everything from the SAP Reference IMG has been copied to the Enterprise IMG.

FIGURE 1.4 The Implementation Guide (IMG) main screen

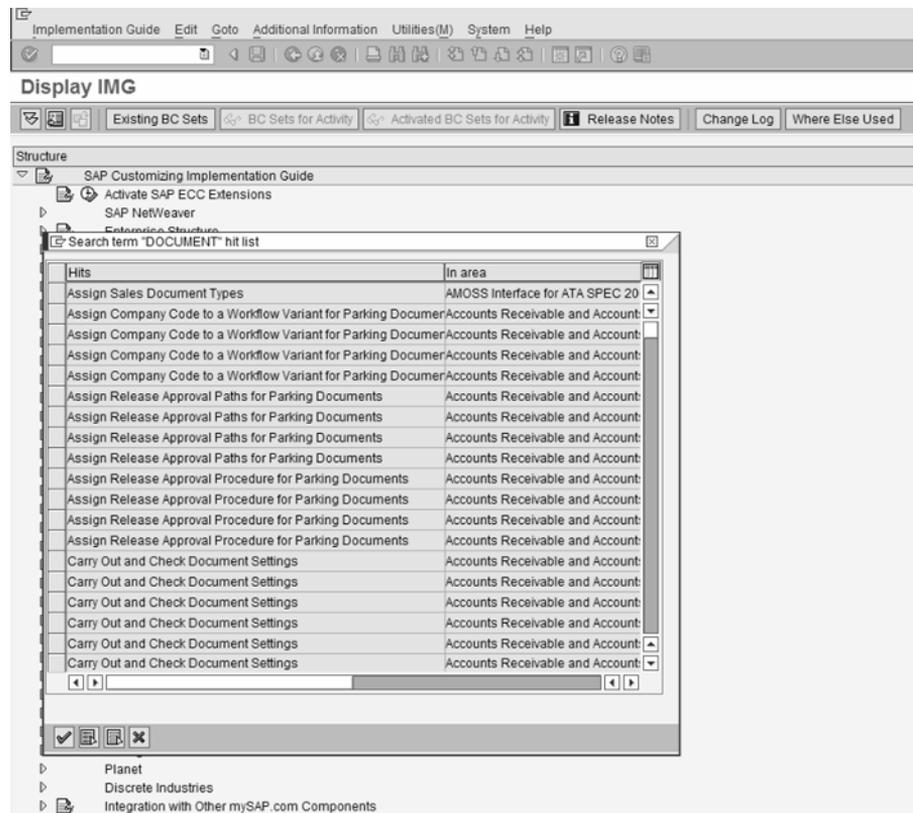
The Project IMG is created by the project team manager or by module team leaders. It contains only those modules and business processes that the creator of the Project IMG (generally a single person or the project manager) deems necessary. It is very important to carefully select what is needed and to not forget anything when generating the Project IMG. The Project IMG can also serve as a valuable project management and documentation tool. The status of project tasks can be viewed and exported to Microsoft Project for detailed project tracking. Using the IMG, you can store configuration documentation with related steps and tables.



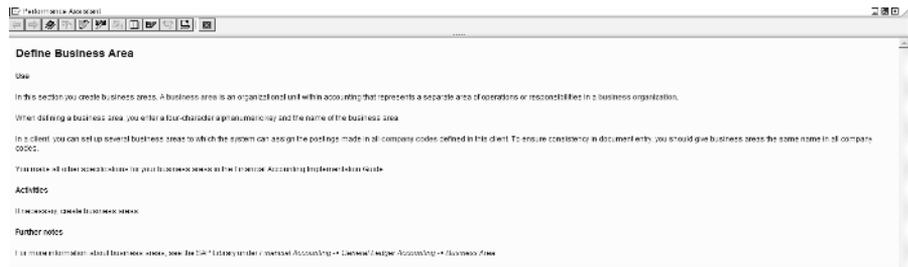
WARNING You and your customer will be happier and more productive in the long run if you keep detailed documentation on the configuration settings that are made in the system. It is not necessary to use the IMG as the documentation tool, but it is necessary to document your entire configuration. The only thing worse than trying to figure out someone else's configuration settings is going back and trying to figure out your own!

If your project doesn't use the Project IMG for documentation and status tracking, it is a good idea to use the SAP Reference IMG or the Enterprise IMG to find your configuration steps. You can expect with some certainty that all steps, processes, and tables will be included in the final product. It is not easy to discover what SAP functionality is needed if you do not have access to all of it. SAP has provided an easy view of virtually all configuration tasks, if you include all of the tasks in the IMG. If you forget to include those tasks in the IMG, you'll find that it is difficult to try to configure the needed functionality. When in doubt, use the SAP Reference IMG. The IMG can display both optional and mandatory activities, as shown in Figure 1.5. By no means must every task in the IMG be completed; the number of tasks and how specific tasks are customized depend on the functionality needed by the business processes being used. It is useful to use the search functionality included in the IMG to find where specific settings are made. To use the search function, click the binoculars icon in the top toolbar of the IMG (shown in Figure 1.5).

FIGURE 1.5 Using the search functionality in IMG



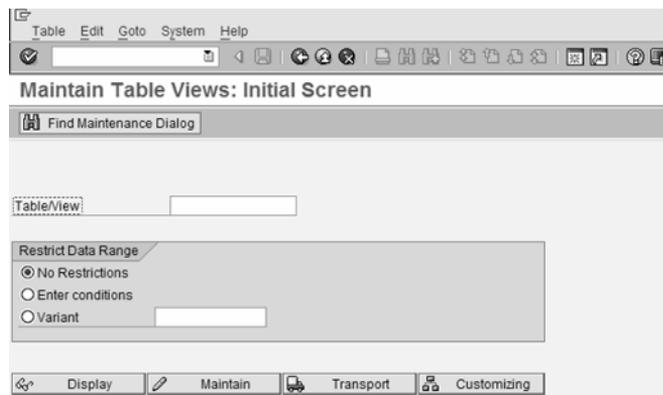
An explanation of the specific tasks can be viewed by clicking the note folder to the left of the configuration step, as shown below. The level of detail provided in the documentation can be very useful in determining which steps, or tasks, are suitable for your project.



Other Methods of Table Maintenance and Customizing

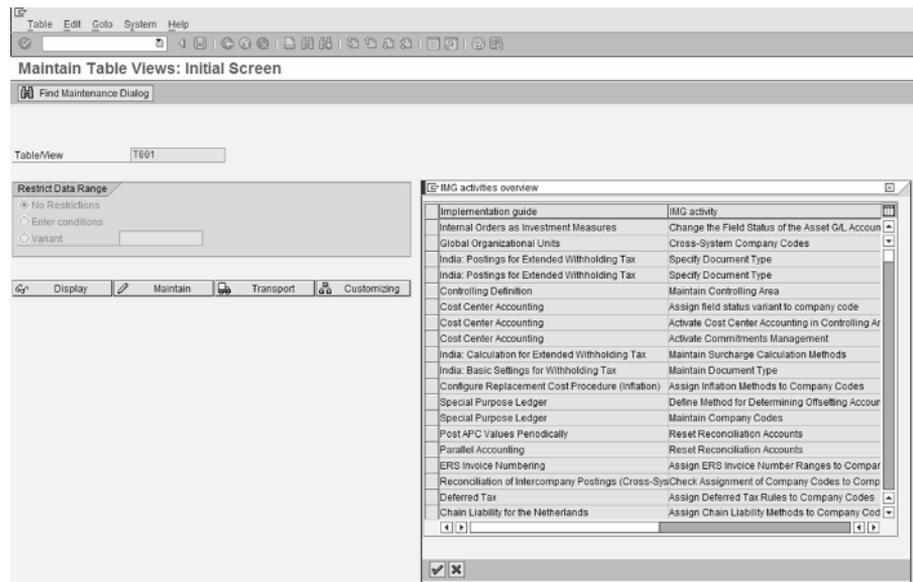
It is sometimes difficult to find the table you need to customize in the IMG. SAP provides two transaction codes that can be used when you know the name of the table you need to customize and you're not sure where it is in the IMG. Transaction code SM30 is used to access the screen for maintaining tables or table views, shown in Figure 1.6. Although it is more common to refer to customizing objects as tables, most customizing objects are actually *views* of tables. To customize an object, you must create a maintenance interface for it. SAP very rarely creates maintenance interfaces on tables themselves. Most of the time, maintenance interfaces are created for table views by SAP for R/3 as delivered. Do not try to create maintenance interfaces for SAP-delivered tables! Transaction code SM30 is used to maintain custom-created (user-defined) tables.

FIGURE 1.6 Transaction code SM30 is for maintaining tables or table views.



In those instances when you know the table name and not the customizing view name, the transaction code SM31 (Extended Table Maintenance) can be used to access the table maintenance screen. Simply enter the table name into the SM31 screen's Table/View field, and click the Customizing button. The first time you use this button, you will be asked for a project number, however this is not a required field and you can hit enter to continue. Once you do that you will be presented with the screen that is displayed in Figure 1.7.

FIGURE 1.7 A list of configuration steps that update table T001



This will give you a listing of the views of maintenance interfaces for that table. Then you can click the name of the appropriate view, and SAP will take you to the proper configuration screen. Note how many configuration steps can update T001 – company codes.

Finding the Table to Configure

Often it may be necessary to find a customizing setting when you're processing a business transaction in the system. The easiest way to accomplish this task is to select on the business transaction the field that contains the setting you want to customize.

In this example, say you want to configure a company code. You can use any transaction that references a company code; in this example, we're using the transaction used to display G/L line items—FBL3N. Figure 1.8 shows the appropriate screen.

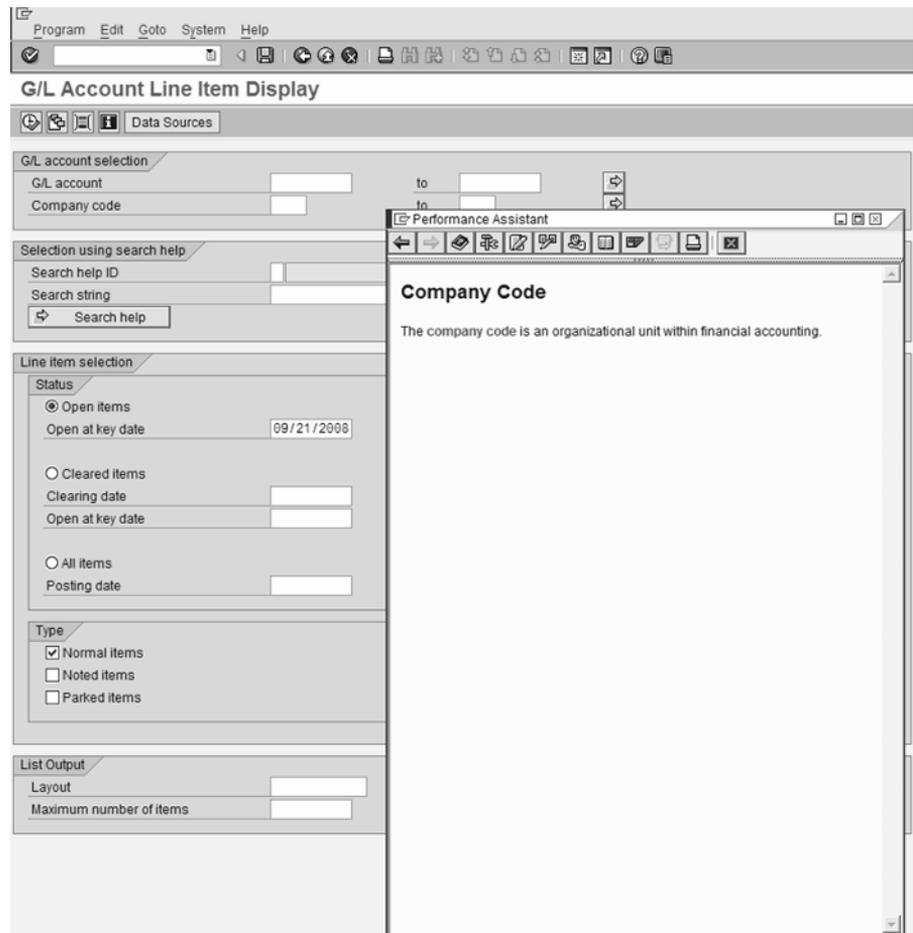
FIGURE 1.8 Displaying/changing line items—FBL3N

The screenshot shows the SAP FBL3N transaction screen. At the top, there is a menu bar with 'Program', 'Edit', 'Goto', 'System', and 'Help'. Below the menu bar is a toolbar with various icons. The main title is 'G/L Account Line Item Display'. The screen is divided into several sections:

- G/L account selection:** Contains two rows of input fields. The first row is 'G/L account' with a 'to' field and a double-headed arrow icon. The second row is 'Company code' with a 'to' field and a double-headed arrow icon.
- Selection using search help:** Contains two rows of input fields. The first row is 'Search help ID' and the second row is 'Search string'. Below these is a 'Search help' button with a double-headed arrow icon.
- Line item selection:** Contains three radio button options: 'Open items', 'Cleared items', and 'All items'. Each option has associated date fields and a double-headed arrow icon. The 'Open items' option has a date field with the value '09/21/2008'.
- Type:** Contains three checkboxes: 'Normal items' (checked), 'Noted items', and 'Parked items'.
- List Output:** Contains two rows of input fields: 'Layout' and 'Maximum number of items'.

Once the field is selected, press the F1 key or select the SAP help icon. This displays general information about the selected field, as shown in Figure 1.9. You can then select the technical information box (the icon with the hammer and spanner).

FIGURE 1.9 The help screen that SAP refers to as a Performance Assistant, which you can open by pressing F1 or clicking the help icon, can give you general information about selected fields.



The Technical Information box lists the field name and tables of the selected setting (table SKB1, field name BUKRS), as shown in Figure 1.10. Armed with this information, you are now ready to go to the configuration table and make the desired changes.

FIGURE 1.10 The table and field names can be discovered by clicking the technical information box.

The screenshot shows the SAP 'G/L Account Line Item Display' screen. A 'Technical Information' dialog box is open, displaying the following data:

Screen Data	
Report	RFITEMGL
Program Name	RFITEMGL
Screen number	1000

GUI Data	
Program Name	RSSYSTDB
Status	%_00

Field Data	
Table Name	SKB1
Field Name	BUKRS
Data Element	BUKRS
DE Supplement	0
Parameter ID	BUK

Field Description for Batch Input	
Screen Field	SD_BUKRS-LOW

The dialog box also includes a 'Navigate' button at the bottom left.

The Dictionary Display is an invaluable tool for viewing table structures as well as finding configuration tables and other pertinent system information. You can use the Dictionary Display to return to the customizing (configuration) table. Then use transaction code SE12 to display the table you found named in the help screen (the data Dictionary Display). After you click the Display button, SE12 lists all fields that are included in the table. Locate the field name found on the help screen, BUKRS (Figure 1.11).



TIP You can also double-click the field or table name on the Technical Information screen and it will take you into the Dictionary Display transaction for the particular table.

FIGURE 1.11 The table SKB1 is displayed using transaction code SE12.

Dictionary: Display Table

Transp. Table: SKB1 Active
Short Description: G/L account master (company code)

Attributes Delivery and Maintenance Fields Entry help/check Currency/Quantity Fields

Search Help 1 / 40

Field	Data element	Data Ty	Foreign	Check table	Origin of the input help	Grch Help	D	Domain
MANDT	MANDT	CLNT	<input checked="" type="checkbox"/>	T008	Input help implemented with chH T008		<input type="checkbox"/>	MANDT
BUKRS	BUKRS	CHAR	<input checked="" type="checkbox"/>	T001	Input help implemented with chH T001		<input type="checkbox"/>	BUKRS
SAKOR	SAKOR	CIAR	<input checked="" type="checkbox"/>	SKA1	Explicit search help attachment SKA0		<input type="checkbox"/>	SAKOR
BRGRU	BRGRU	CHAR	<input type="checkbox"/>				<input type="checkbox"/>	BRGRU
BUSAB	BUSAB	CHAR	<input checked="" type="checkbox"/>	T001S	Input help implemented with chH T001S		<input type="checkbox"/>	BUSAB
DATLZ	DATLZ	DATS	<input type="checkbox"/>		Input help based on data type		<input type="checkbox"/>	DATLZ
ERDAT	ERDAT_RF	DATS	<input type="checkbox"/>		Input help based on data type		<input type="checkbox"/>	DATLZ
ERNAM	ERNAM1	CHAR	<input type="checkbox"/>				<input type="checkbox"/>	USNAM
FDGRV	FDGRV	CHAR	<input checked="" type="checkbox"/>	T035	Input help implemented with chH T035		<input type="checkbox"/>	FDGRV
FDLEV	FDLEV	CHAR	<input checked="" type="checkbox"/>	T036	Input help implemented with chH T036		<input type="checkbox"/>	FDLEV
FIFLS	FIFLS	NUMC	<input type="checkbox"/>				<input type="checkbox"/>	FIFLS
FSTAG	FSTAG	CIAR	<input checked="" type="checkbox"/>	T004F	Input help implemented with chH T004F		<input type="checkbox"/>	FSTAG
HRKTD	HRKTD	CHAR	<input checked="" type="checkbox"/>	T012	Input help implemented with chH T012		<input type="checkbox"/>	HRKTD
HKTID	HKTID	CHAR	<input checked="" type="checkbox"/>	T012K	Input help implemented with chH T012K		<input type="checkbox"/>	HKTID
KDFSL	KDFSL	CHAR	<input checked="" type="checkbox"/>	T038S	Input help implemented with chH T038S		<input type="checkbox"/>	KDFSL
NITKZ	NITKZ	CHAR	<input type="checkbox"/>		Input help with fixed values		<input checked="" type="checkbox"/>	NITKZ
SMWSK	SMWSK	CHAR	<input type="checkbox"/>				<input type="checkbox"/>	SMWSK
STEXT	STEXT_SKB1	CHAR	<input type="checkbox"/>				<input type="checkbox"/>	TEXT68
VZSKZ	VZSKZ	CHAR	<input checked="" type="checkbox"/>	T050	Input help implemented with chH T050		<input type="checkbox"/>	VZSKZ
WAERS	WAERS_SKB1	CUKY	<input checked="" type="checkbox"/>	ICURC	Input help implemented with chH		<input type="checkbox"/>	WAERS
WMETH	WMETH	CIAR	<input type="checkbox"/>		Input help with fixed values		<input type="checkbox"/>	WMETH
XGKON	XGKON	CHAR	<input type="checkbox"/>		Input help with fixed values		<input type="checkbox"/>	XFIELD
XINTB	XINTB	CHAR	<input type="checkbox"/>		Input help with fixed values		<input type="checkbox"/>	XFIELD
XKRES	XKRES	CHAR	<input type="checkbox"/>		Input help with fixed values		<input type="checkbox"/>	XFIELD
XL0EB	XL0EV	CHAR	<input type="checkbox"/>		Input help with fixed values		<input type="checkbox"/>	XFIELD
XKON	XKON	CHAR	<input type="checkbox"/>		Input help with fixed values		<input type="checkbox"/>	XFIELD

By viewing the structure of table SKB1, you can see that the field name BUKRS has a check table of T001. This is the table you need to configure in this scenario. You know you need to configure table T001 because this is where the primary key for company code (field name BUKRS) resides. In SAP, check tables always refer to the primary table that the field belongs in. If there is no check table next to the field name, you know that this instance of the field is the primary key and not included in the table via a foreign key relationship. In relational database terms, the field BUKRS in table T001 is the foreign key for table SKB1. Because it is the foreign key, you know that table T001 is the table in which BUKRS (a key field) is created and maintained. To find the configuration view you need to access, use transaction SM31 (discussed in the preceding section). Once you have entered T001 in the table name field and clicked the Find Maintenance Dialog button (shown earlier in Figure 1.6), a list of configuration views is returned.



TIP In relational databases, a *key field* is a unique identifier for a table. This field is used as a reference to the same data in other tables. For example, your employer uses your Social Security number as a unique identifier for information about you. Rather than listing all the relevant statistics about you in every table, a relational database connects the various kinds of information through a single key field that represents “you.” A *foreign field* is another sort of key field, but it’s the key field for a “foreign” table. For example, say your employer needs a table containing names and addresses. In this table, the key field might be the last name of the employee. The last name would certainly also be referenced in a table containing Social Security numbers, but it would be a foreign key there—used only to verify that two people with the same last name stay unique as entities in the various tables.

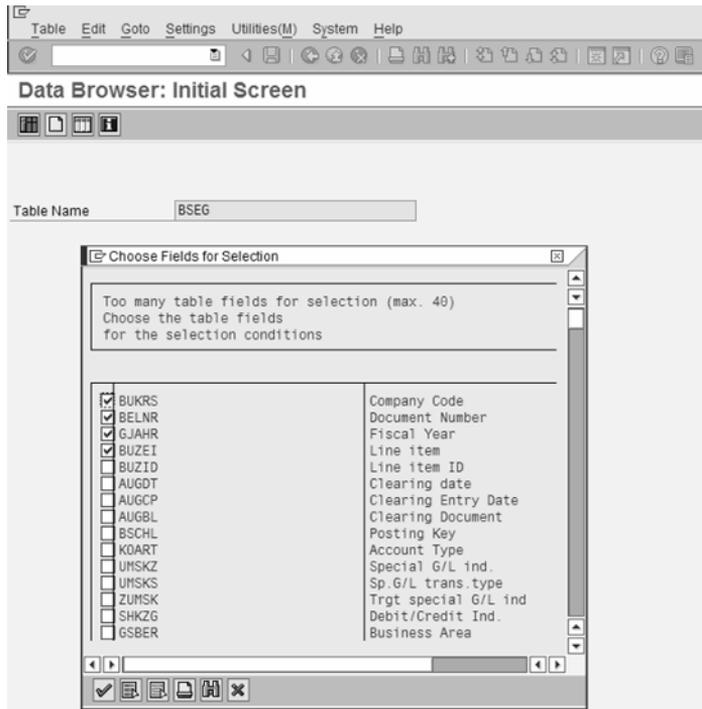
The Data Browser and Common Tables to Display

The Data Browser is a useful tool for displaying data table contents. Transaction SE12 is used to display the structure of a table, and the Data Browser, which is run via transaction code SE16, displays the contents of a table. Figure 1.12 shows the configuration screen for creating a company code. The same detail can be viewed through SE16 by using table T001. The difference between these two options will be how SAP presents the data in a configuration screen, often linking more than one table to the screen presented.

FIGURE 1.12 Creating a new company code configuration screen

Change View "Company Code": Details	
Company Code	3000
Company Name	IDES US INC
Additional data	
City	New York
Country	US
Currency	USD
Language	EN

Figure 1.13 shows the results of running transaction code SE16 and entering BSEG as the table name. You can choose from the listing of options for narrowing the results that you would like returned to you.

FIGURE 1.13 The options for narrowing the results of transaction code SE16

The fields that are automatically displayed for selection criteria aren't the only fields that can be used. By following the menu path **Settings > Fields for Selection**, you can choose additional fields to select. Remember that SAP will generally include the key fields as selection fields. If you are adding in selection fields and you decide to use this new field, you will be faced with some very noticeable increase in run times because tables are indexed by the key fields. You can check to see whether a field is an indexed field by using SE12. They are usually the top 2 to 10 fields and have a check mark under the column key. Be careful, because SAP defaults so that only 500 entries are returned. If you require more than 500 entries, you can change this option via the field at the bottom of the selection screen.

Please remember that the fields being displayed are also defaulted and can be modified by using the menu path **Settings > Format List > Choose Fields**. If there are a lot of entries within your selection parameters, it may be easier to analyze your results in Excel. You can easily achieve this by following the menu path **System > List > Save > Local File**.

You can save it to your hard drive, and then it is a very easy task of importing it into Excel and converting the file.

You can use the Data Browser to display data from all different types of tables, including summary and line item detail. This is very useful because reports using the Report Painter or Report Writer can be written only against summary-level tables. The Report Painter and Report Writer tools allow you to create customized reports based on users' needs. Most modules in the system can utilize Report Painter and Report Writer to create reports. Report Painter and Report Writer are especially useful for creating FI/CO custom reports. The table that is most commonly displayed by FI/CO team members is BSEG. This table contains all the segment (line item) detail behind every financial transaction in FI. You need to be very careful when displaying BSEG, or any segment-level table, because you can cripple performance on your system if you do not narrow your selection criteria enough. Running wide-ended table displays or queries is also a good way to get locked out of transactions by your friendly neighborhood Basis group, and a good way to feel the wrath of your project team.

The Service Market Place

The Service Market Place (<https://websmp201.sap-ag.de/notes>), formerly known as Online Support System (OSS), is a web-based tool provided to you by SAP to help solve application problems. All configurators and developers should have access to this valuable database and customer service center. You can log functionality or code issues with SAP on this website. Because of the vast install base that SAP has, there are normally several other customers who have encountered your issue and therefore SAP usually has already posted a solution. These solutions are known as SAP Notes.

SAP Notes usually supply additional code to fix program bugs. They may also be “consulting” notes explaining workarounds in the system for certain functionality. Whichever the case, the information provided in the Service Market Place is invaluable. If you can't find a needed piece of functionality in the system, always make sure you check Service Market Place to see if it already exists. To find SAP Notes, one would have to first log on to the Service Market Place. Please remember you will need an OSS logon ID—that is, a user account for the Service Market Place—which you can get by going to the SAP Support Portal. If you want to search directly for SAP Notes, there is a link just below the SAP Notes Search portal heading. This will

take you to your typical search parameters. Just fill in the criteria you're looking for (the release of SAP you are working with and the application area), and the applicable SAP Notes will be returned to your screen. If you find a note that applies to the problem at hand, download a copy to your hard drive for yourself, then let one of your Basis teammates know which note you would like applied. Unless you are very experienced, don't try to apply SAP Notes yourself. SAP Notes often deal with changing and adding to core SAP source code. Changing SAP programs is strongly advised against and can impact the support SAP provides to you. We will explore this in greater detail in the next section, "Modifications to SAP Source Code and User Exits."

In addition to SAP Notes (bug fixes, workarounds, and minor functionality enhancements), the Service Market Place has other purposes. As mentioned, you can log questions to SAP. Other uses include checking SAP training class schedules, reviewing new information and releases from SAP, and looking up information about Hot Packs.



NOTE In simple terms, *Hot Packs* are groups of SAP Notes that SAP recommends you should apply based on bugs in the system or on additional functionality provided by the notes. Instead of applying individual notes one by one, you can use Hot Packs to apply a group of SAP Notes.

Modifications to SAP Source Code and User Exits

The golden rule of packaged software and of SAP in particular is this:

Do *Not* Modify The Source Code!

Always try to live by the golden rule; a modification to SAP source code is a *bad* idea, to say the least. Once you modify the source code of a program, SAP generally will not support the program and related business processes and may not support your entire installation. Source code modifications also make for a nightmare when you're applying Hot Packs and are even more of a problem when you're trying to do upgrades. Some SAP clients have made modifications to source code and have paid the price: they are now trying to remove the modifications to get back to "core" code in order to regain support from SAP.

If standard SAP functionality just doesn't work for your business, there are other options. SAP has developed specific Industry Solutions (ISs). There are prefabricated ISs for certain industries, such as IS-Aerospace & Defense, IS-Oil, IS-Retail, and

many more. One of the existing ISs may work for your business. If not, work with SAP regarding your business needs; SAP may make—or allow you to make—SAP-supported modifications to your system.

SAP also provides what are called *user exits* in some standard programs. User exits allow developers to create their own code that is called by a standard SAP program. Once the custom code has finished, control returns to the standard SAP program for further processing. Contrary to programming modifications, home-brewed user exits are a good idea. SAP is providing increasing numbers of user exits in upgrades and new releases. To see if a program already has a user exit, while displaying the program source code, search for the string *customer exit*. If this string is found, the program already has a user exit. The transaction code CMOD also contains the customization projects for activating all user exits. You can search through these projects to see if the functionality you need is included.

Summary

In this chapter, you were introduced to the basic concepts you'll need to understand to get the most out of the rest of this book. You reviewed the basic configuration of a three-tiered database and learned about a table-driven approach to designing your own system. You also learned about transports and who has control over the various aspects of preparing and implementing a transport. We walked you through the IMG screens and a few tables that will be useful in configuration and customization. And you're now more familiar with the sorts of changes that you shouldn't make and how to find out about problems and solutions using the Service Market Place.

In the next chapter, we'll cover the Organizational Structure of the FI Enterprise system and begin to configure the system for our project company, Extreme Sports.

