MainBoss 4.2
Configuration

Copyright © 2018, Thinkage Ltd.

Revision 27

Table of Contents

Introduction
Welcome to MainBoss

This guide provides a detailed explanation of the process of setting up MainBoss before putting the software into active service.

☐ For information on installing the software, see the <u>Installation and Administration Guide</u>.

☐ For a quick overview of MainBoss set-up, see the <u>Configuration Quick-Start</u>.

☐ For information about day-to-day use, see the <u>Day-to-Day Operations Guide</u>.

Important: If you have not licensed a particular module, you will not see information related to that module. For example, if you have not licensed the **MainBoss Requests** module, you will not see various types of information related to requests.

Documentation conventions

This guide uses the following documentation conventions:

☐ Data-entry fields are shown in bold face, enclosed in double quotes:
  Fill in the "**Date**" field...

☐ Buttons are shown in bold face and underlined:
  Click the **Edit** button.

☐ Keyboard keys are shown in bold face enclosed in **<angle brackets>**:
  Press the **<Tab>** key.

Using the Mouse: Whenever this guide tells you to click on a button, a menu item, or some other object on the screen, click with the left button of the mouse. When the guide says to *right-click* an object, click with the right button of the mouse.

The Golden Rules of MainBoss Set-Up

Rule 1: Plan in advance for the things you need, and don't put in anything else.

Here's an example: every work order can be assigned a priority. However, MainBoss doesn't have a predefined set of priority classifications—*you* decide what priorities you want to use. For example, you might use a simple system like

```
High priority
Average
Low priority
```

or time-oriented priorities like

```
Immediately
By the end of the shift
Within 24 hours
Within a week
Whenever
```

Of course, your organization may already have an established set of priorities, in which case, you just use those.

It's up to you to choose what priority system will work best with your operations. However, if you aren't ready to decide, **don't use priorities at all**. Work orders don't *need* priorities...and it's better to use a system you have confidence in than something you make up without much thought. Changing your mind and reorganizing after the fact can be a lot of work.

If you don't use priorities to begin with, you can start using them later. Once you're comfortable with MainBoss, you can think about what priority system would be useful and relevant. You still don't have to assign priorities to every work order—just the ones that have higher or lower urgency than normal.

The same principle applies in everyday operations: don't use things you don't need. For example, MainBoss lets you record *access information* on a work order. This might indicate that you need to obtain someone's permission before taking equipment offline or that repairs are only allowed at certain times of day. However, if there are no special restrictions on a particular job, don't fill in the "**Access Code**" field. Leave the field blank in normal situations, so that when you *do* fill it in, workers are more likely to notice that this job has special circumstances.

Rule 2: Choose useful categories and codes

MainBoss offers the option of defining categories and information codes for various types of data. For example, you can assign categories and/or codes to your work orders, your requests, your suppliers, and so on.

Your organization may already have categories and codes for this kind of information. If not, you might create work-order categories like

```
Plumbing
Electrical
Mechanical
Inspection
Clean-up
```

Using such categories can be helpful in getting a clear picture of your operations. Do we spend more on plumbing than electrical jobs? How much time do we actually put in on clean-up? How many inspections has Joe done in the last month?

As for codes, MainBoss lets you define closing codes to be assigned to work orders after the job is finished. Many organizations use these codes to specify the original source of the problem, such as

```
Operator error
Accidental breakage
Poor lubrication
Normal wear
Vandalism
```

By labeling each work order with an appropriate code, you can later get detailed reports on how common each type of problem is and the cost associated with each problem class.

> One of the most important functions of MainBoss is to provide answers to questions—not just your own questions, but questions asked by customers, upper management, and other people outside the maintenance department. When setting up MainBoss, it's important to think about the questions you'll need to answer and to record information from which the answers can be extracted—you can't change what you can't measure.

MainBoss can answer questions easily if you attach categories and codes to your records.

☐ You might choose to figure out categories and codes *before* you enter any other data. This provides a structured way of thinking about your operations: you come up with an organization plan before you put in detailed information about jobs, equipment, work materials, etc.

☐ You might decide on categories and codes *after* you've used MainBoss for a while. You can then draw on your experience with the software to decide which categories and codes will be useful.

Either way can work. If you're already familiar with computerized maintenance management, you may have an existing set of categories and codes you like to use. If not, you may prefer to wait until you're more comfortable with MainBoss.

Whether you start with categories and codes or add them later, you have to think about what will actually be useful to you. If, for example, your company's budgeting process breaks down expenses in a certain way, then you'd better use the same approach in MainBoss. Otherwise, you'll end up with headaches trying to deal with differences.

Even if you don't have to worry about external factors, you still should think through what categories and codes you want to track. Otherwise, just omit categories and codes from work orders until you have time to decide what would be useful. Choosing a set of categories and codes that *don't* work for you (or don't give you the whole picture) will be more trouble than it's worth.

Rule 3: Make a phase-in plan

If you've never used computerized maintenance management before, we recommend that you phase in MainBoss a little at a time. Trying to computerize your entire operation in one fell swoop will be a daunting job. Furthermore, you'll be making decisions before you have a chance to get some experience with the software.

As an example of phase-in, you might begin with just tracking work requests (problem reports). Once you're comfortable with that, you might move on to work orders and the scheduling of planned maintenance. After that, you add inventory tracking...and so on.

The point is to *have* a plan: know what you want to get working first and what steps to take to reach your goals. What information do you have to collect and record? What decisions do you have to make? Who has to be trained to use the software?

Don't just start entering data and hope for the best.

☐ Think ahead

☐ Take manageable steps

☐ Only start with what you know you need; add extras later

Rule 4: Be Consistent

Before you begin entering *any* information, make sure you have a coding policy that will be followed by everyone who uses MainBoss. Stick to the policy with absolute

consistency. This is particularly important in large organizations—you have to make sure every MainBoss user is taking the same approach.

For example, consider the names by which you identify your equipment. You might decide on names like this:

```
Air conditioner 1
Air conditioner 2
Air conditioner 3
```

If someone then uses the name

```
AC 4
```

it will throw off the consistency of your records. In a list of equipment sorted in alphabetical order, the inconsistent one won't be sorted with the other air conditioners. It will be harder to find the unit you want when you're visually searching the list; the unit may also slip between the cracks when you're printing off reports about your equipment.

The same applies to all other types of information in your database. **Names must be consistent: similar things should have similar names.** In particular, you should choose the format of names so that similar things appear together when sorted in alphabetical order.

Decide on standard name formats *before* you start entering data. This decision affects how much benefit MainBoss can provide. A haphazard naming scheme reduces the software's ability to organize and analyze data in effective ways.

Rule 5: Be Safe

SQL Server has facilities for backing up and restoring data. **We strongly recommend that you make regular backups of your MainBoss database and store copies in a safe place.** Crashes happen, and you can't recover what you haven't saved.

We also recommend that you *mirror* your disks, so that you don't put all your eggs in one basket. For more information, see

Mirroring Your Disks

Basic Principles of MainBoss

What You See

Here's a typical picture of MainBoss's main window:
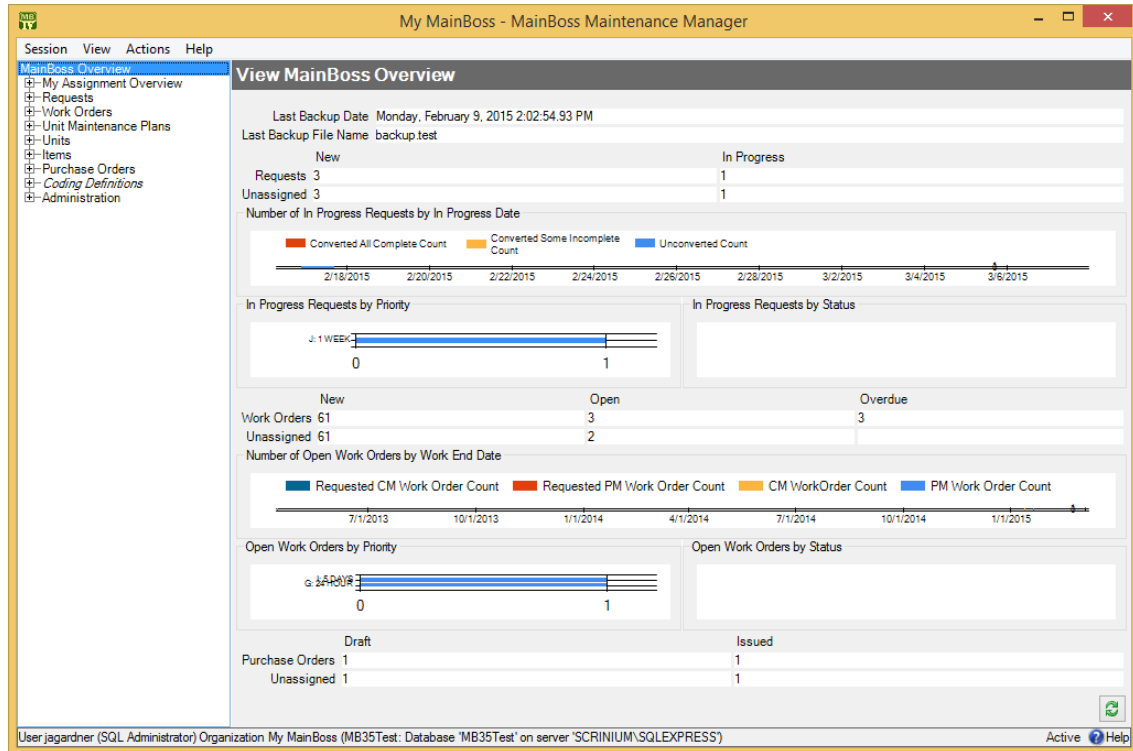
On the left is the *control panel*. This provides access to all standard actions and information. At the bottom right, the ↻ (**Refresh**) button updates the information displayed (in case other users have recently made changes to your organization's data).

> **General Principle:** In MainBoss, you use the control panel (on the left) to go to a table, then you use the table's buttons (on the right) to do work.

This manual uses the notation **Entry** | **Subentry** to refer to items in the control panel. For example, **Work Orders** | **Reports** refers to **Reports** under the **Work Orders** entry in the control panel.

Whenever you select an entry in the control panel, the rest of the window changes to show information about what you selected. For example, if you click **Work Orders** in the panel, the right-hand part of the window changes to show information on work orders.

We recommend that you maximize the MainBoss window so that MainBoss has the most possible screen-space to display information.

Keyboard Shortcuts: MainBoss allows standard Windows keyboard shortcuts such as **<Ctrl+C>** for copy and **<Ctrl+V>** for paste.

You can also use **<Tab>** to move to the next field in a window. If you're filling out a form, you can just tab from one field to another rather than using the mouse. **<Shift+Tab>** moves backward to the previous field.

In any table viewer, the shortcut sequence **<Ctrl+F>** opens the search window. The search will be applied to whatever list of records you clicked most recently. For example, if you most recently clicked a list of unit records, **<Ctrl+F>** will let you search through units. However, if you were looking at a unit record and your most recent click was on the unit's list of service contracts, then **<Ctrl+F>** would search through service contracts. (**<Ctrl+F>** will not work if your most recent click was on the MainBoss control panel—it only works if your most recent click was in some sort of table viewer.)

If the search window is not open, the **<F3>** function key has the effect of **Find Next**, searching for whatever you searched for most recently. This means that pressing **<F3>** repeatedly will perform the same search over and over.

MainBoss buttons often allow keyboard shortcuts. To see the shortcuts in a window, press the **<Alt>** key. One letter of each button will be underlined, as in **Save**. Pressing **<Alt>** plus the underlined letter is equivalent to clicking the button, so that **<Alt+S>** is equivalent to clicking the **Save** button.

View Options: The **View** menu provides options that control how information is displayed:

☐ **Zoom in on View**: If you click this menu item, MainBoss will hide the control panel in this window. To bring back the control panel, click **Zoom in on View** again.

☐ **Open in New Window**: If you select this option, MainBoss opens a new window containing the same information as the current one. This is useful if you want to compare information between tables: you can have two windows open, one showing one table and one showing the other.

☐ **Change Active Filter**: This item lets you change the definition of "active" requests, work orders and purchase orders. For example, you can specify that work orders should only be displayed in table viewers if they're less than three months old. Changes that you make through this menu item affect you but not other users.

☐ **Show compact browsers**: If this is checkmarked, table viewers will not show details from the currently selected record. This makes it possible for a table viewer to list many more records. If the entry is not checkmarked, table viewers will have

an area showing the contents of the currently selected record; this reduces the number of records that the window can show at any one time.

☐ **Form Customization**: Offers a submenu of actions related to customized table viewers, editors, and reports. The submenu contains:

> ☐ **Enable**: If this is checkmarked, you can customize table viewers, editors and report windows, as described in "Form Customization" later in this document. If you remove the checkmark, you will not be able to make such customizations.

> ☐ **Customize Main Menu**: Lets you choose which entries appear in the control panel.

> ☐ **Publish**: This saves all your current form customizations as a single set of "preferred" customizations. You can make other customizations after publishing, but they will be forgotten when you quit MainBoss. Only published customizations are remembered from one session to the next.

> ☐ **Restore from published**: Gets rid of any customizations you have made since you last clicked **Publish**. You will return to the preferred customizations you have already saved.

> ☐ **Clear all customization**: Removes any and all form customizations you have created, restoring everything to MainBoss's default forms. This does *not* delete your published customizations; therefore, you can use **Restore from published** to return to your published customizations.

> ☐ **Export**: This saves your current customizations in a file. (MainBoss will open a window so you can specify a file name and location.) Saving customizations in a file lets you pass your customizations along to someone else. It also keeps a record of your current customizations if you intend to publish a different set of customizations.

> ☐ **Replace from file**: Replaces all your current customizations with customizations from a file. The specified file should contain customizations previously saved with **Export**.

> ☐ **Add from file**: Adds customizations from a file to your current set of customizations. The customizations will be added to any customizations that already exist; however, if a customization in the file conflicts with one of your current customizations, the customization in the file will replace your current customization.

☐ **Select View**: Contains submenus that reproduce all the entries of the control panel. This is useful for people who would rather work with menus (especially menu shortcut keys) rather than using the mouse to click on the control panel.

Coding Definitions: The **Coding Definitions** part of the control panel deals with information that doesn't change very often. For example, your planned maintenance task descriptions usually don't change much: you set them up when you first start using MainBoss, then mostly leave alone.

> Most of the work for configuring MainBoss is done in **Coding Definitions**.

Identification Codes and Descriptions

Every record has an identification code. This code is just a label used to identify the record; it can contain any text character.

Codes are restricted in length. Different records have different length restrictions, sometimes as short as 10 characters, sometimes as long as 30.

Codes have different names in different types of records—for example, work orders are identified by a "**Number**" while inventory items are identified by a "**Code**"—but the purpose of the identifier is always the same: to provide a label that distinguishes one record from another.

Most records also have a "**Description**" field. This is often used to expand on the ID code. For example, in a unit record for a water boiler, the "**Code**" might be `Boiler NorthEast` while the "**Description**" might specify what type of boiler it is. MainBoss limits the length of descriptions too, but descriptions are always allowed to be much longer than codes.

> **Remember the Golden Rule:** Codes and descriptions should be consistent. Similar things should have similar names. The best codes are short forms of the descriptions.

Categories and Codes

You can define your own categories for various types of information. For example, you can divide up the places you maintain into categories—you might choose to create categories like Washrooms, Office Space, Stores, and so on.

Categories are optional. You don't have to categorize your data before you start using MainBoss. However, categories are useful for organizing your data, especially when creating reports. For example, you might ask MainBoss to create a report on work orders, organized according to the type of space where the work was done. This lets you compare the amount you spend maintaining washrooms, versus the amount you spend maintaining office space and the amount you spend on stores.

Breaking down your costs in this way can give you a clearer picture of where you're spending your money and where you might get the best bang for your buck by making improvements. For example, if you discover that office spaces require a lot more attention than stores, you may be able to redistribute your resources to take that into

consideration. A small improvement in something you do a lot can have a significant effect over time.

Well-organized information also helps you when dealing with upper management (e.g. in budget talks). If you can show your boss hard-and-fast proof of how much you're spending and exactly where the money goes, you can justify your budget requests in dollars and cents.

Defining Categories: MainBoss lets you define categories for work orders, personnel, suppliers, and much more. It's up to you to decide what categories you think are useful. Remember that categories are used to classify and organize your data so you get the perspective you need. Think about what *you* would find helpful and define your categories accordingly.

Maintenance Organizations

When you first begin using MainBoss, you may be asked to specify your maintenance organization:
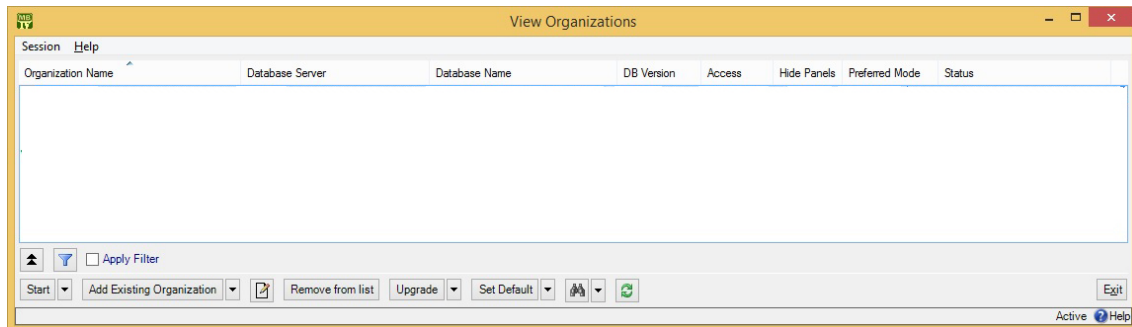
If you have no existing maintenance organizations (i.e. if you just installed MainBoss), see the Installation and Administration Guide for information on creating a maintenance organization. Otherwise, click **Add Existing Organization**. This opens a window where you can specify what organization you want to use:
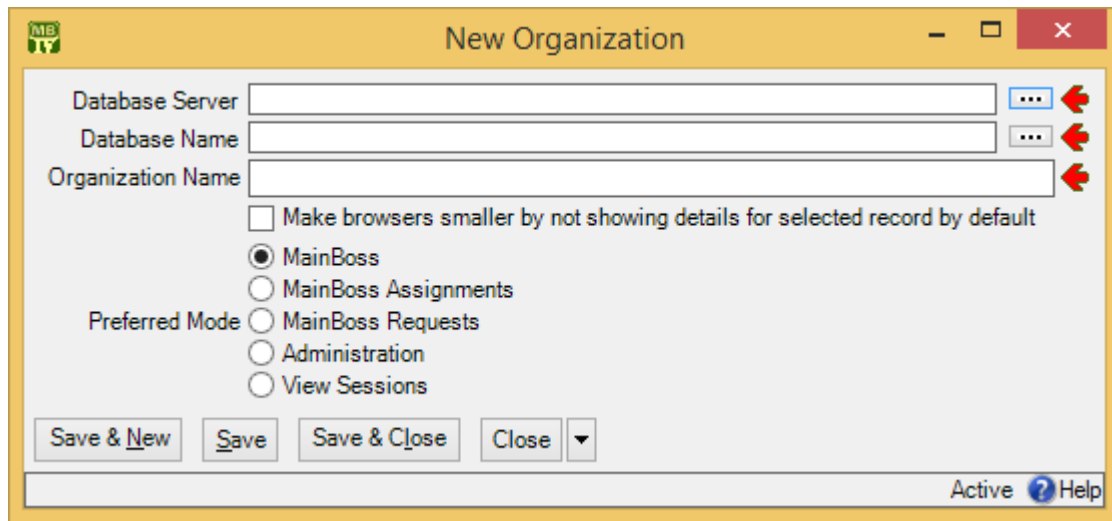
During the MainBoss installation process, your MainBoss administrator will have created an appropriate database on a server computer. Ask your administrator for the data you need to fill in the above window. Click **OK** when finished.

When you add an organization to your list, the new addition is automatically made your default organization. Whenever you start MainBoss in future, the program will automatically open this maintenance organization. Therefore, most users will never have to specify a maintenance organization in future.

Changing to a Different Maintenance Organization

Some companies may create multiple maintenance organizations. For example, a property management company with multiple clients may create a separate database for each client. As a result, workers in such a company may need to switch from one maintenance organization to another while using MainBoss.

To do this, open the **Session** menu and click **Change Maintenance Organization**. This opens a window that lists all the maintenance organizations known to you. Choose an organization from this list, then click **Start**. MainBoss will open the corresponding database and you can begin working with the associated data.

Each MainBoss database must have its own set of licenses—you cannot use the same set of licenses on different databases. For more information on licenses, see your MainBoss licensing agreement or contact Thinkage.

Quitting MainBoss

To quit MainBoss, simply close all the MainBoss windows that you have open.

You can close a main MainBoss window and still have other windows open. MainBoss continues to run on your computer until you close the last window.

Table Viewers

A table is a list of entries, each of which contains information about something. For example, the **Vendors** table is a list of your suppliers.

Each entry in a table is called a record. For example, each entry in the **Vendors** table gives information about one of your suppliers.

A table viewer lists the records in a particular table. For example, the **Vendors** table viewer lists the entries in your **Vendors** table.

Table viewers usually have two sections:

> **View**: A list of normal records—this can be anything from work orders to information about spare parts.

> **Defaults**: Default values for creating new records. For example, the default values in the **Vendors** viewer will be used whenever you create a new vendor record.
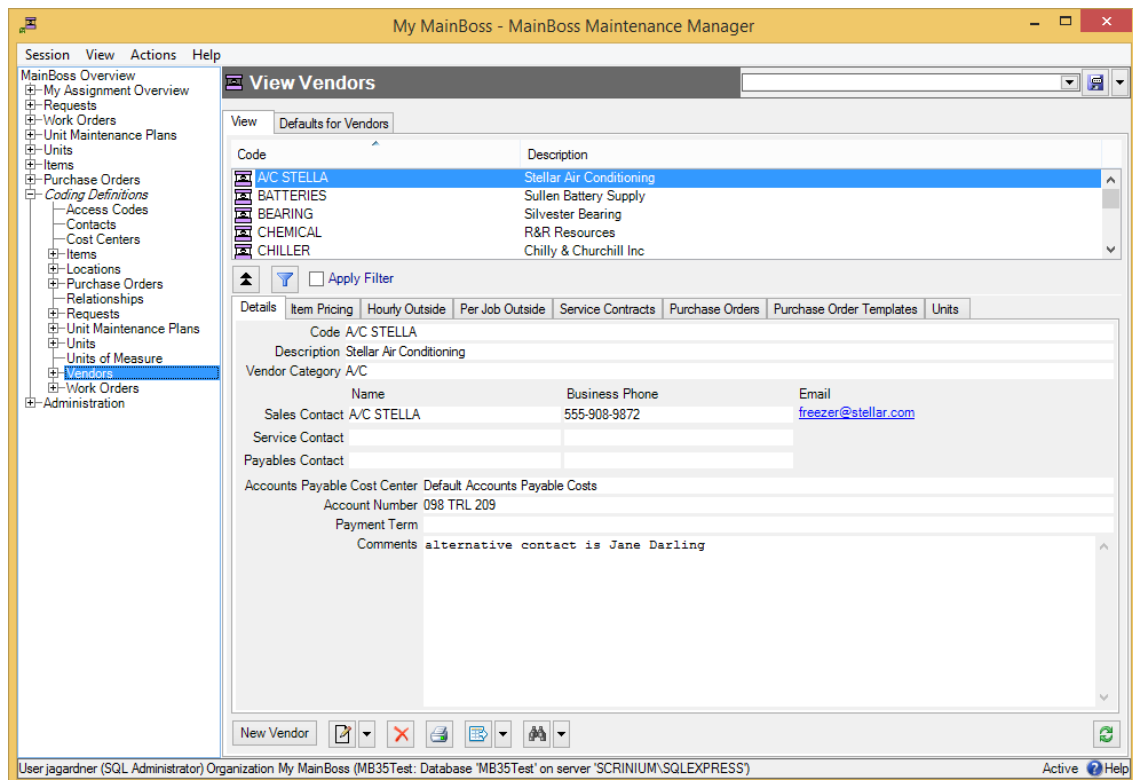
Special Selection Actions

If you right-click on the records listed in a table viewer, MainBoss opens a menu of actions that can help you select entries in the table. For example, in most tables, the menu will include a **Select all** action; clicking this action in the menu selects all the entries in the table.

As another example, some right-click menus have an action named **Select all editable**. This selects all the entries that can be edited from this particular table viewer. For example, the **Units** table displays both units and the locations that contain the units. Records for the units can be edited from this table; locations cannot be. Therefore, **Select all editable** will select all the unit records in the table but will leave locations unselected.

Tables with entries that can be expanded or contracted have a menu item named **Expand new entries**. This lets you determine whether the entries are fully expanded or fully contracted when you start up MainBoss.

Hiding the Details for a Selected Record

By default, when you select a record in a table, the table viewer displays information from that record. Typically, this is shown in a *details panel* beneath the list of records in the table.

If your computer has a small monitor screen, you may decide that the details panel takes up too much room on the screen. If so, you can tell MainBoss not to display the details panel; you do this by clicking the following button:

When you click this button, the details panel disappears. The list of records can then use the space that was previously occupied by the panel.

If you want to see the details panel again, click the same button.

Active and All in Table Viewers

In the bottom right hand corner of most table viewers and editor windows, you'll see either `Active` or `All`:

☐ If the word is `Active`, lists of records displayed in the window only include *active* records. This means that the window only shows records that haven't been deleted. For requests, work orders, and purchase orders, it also means that it only shows records that are "recent". By default, a record is "recent" if it's less than 500 days old; your MainBoss administrator may change what counts as "recent."
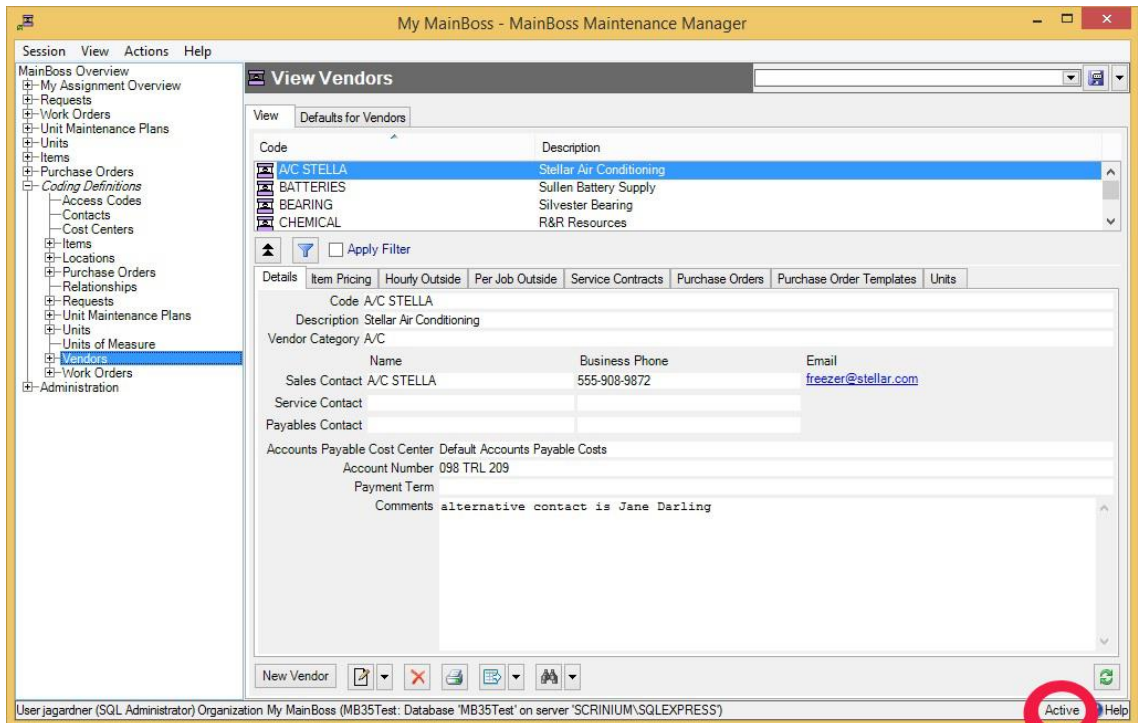
☐ If the word is `All`, the viewer is showing *all* records, no matter how old, including those that have been deleted. Deleted records are marked with a red X on the left. (Note that if you have many old records, MainBoss may take a considerable time to obtain them all from the database.)
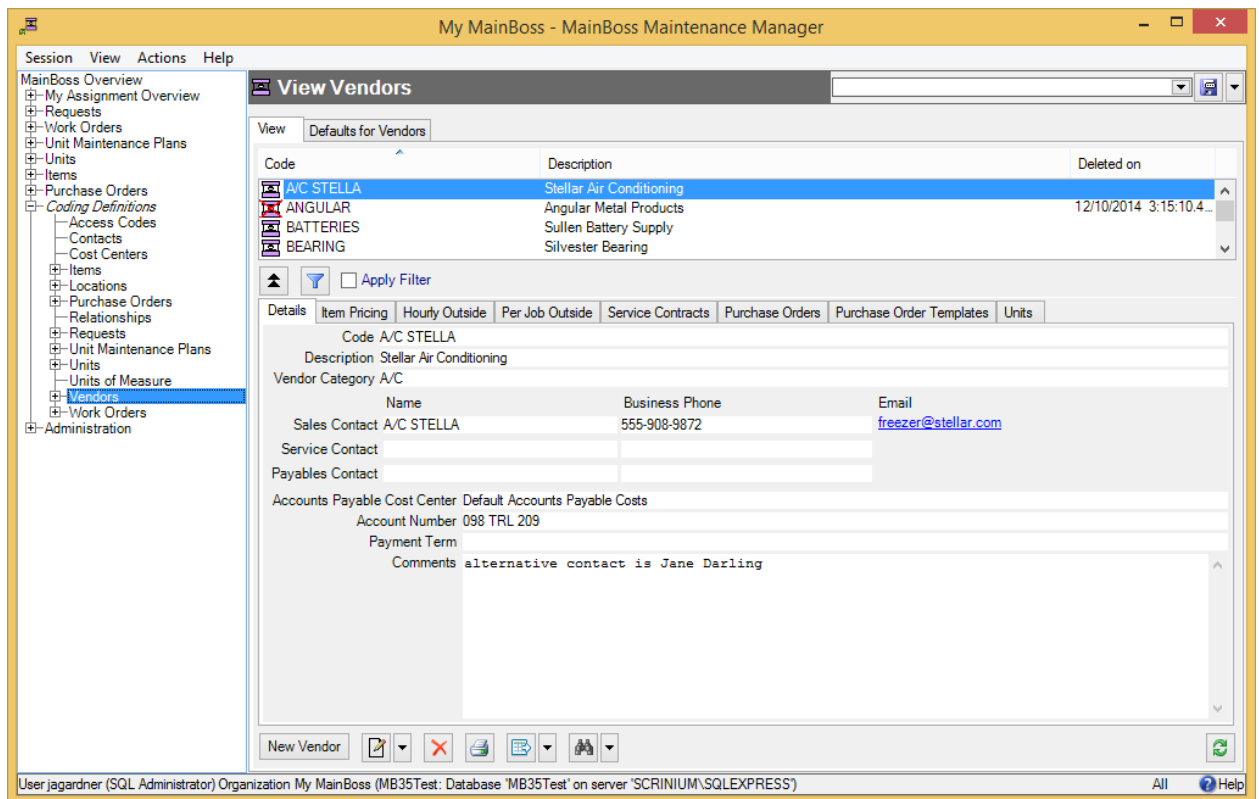
In most tables, MainBoss never completely erases deleted records. This makes it possible to maintain a complete audit trail of everything you do.

To switch from `Active` to `All` or vice versa, *double-click* the word. You'll see old and deleted records appear or disappear.

The **Restore** button lets you bring back a record that was previously deleted. MainBoss opens a window where you can examine the record to be restored and make changes if necessary. When you save the record and close the window (typically by using **Save & Close**), the record will be restored to the active list.

Records That Refer to Deleted Information: When you delete a record, it's possible that other records still refer to it. For example, suppose a worker leaves your organization so you delete that person from your **Employees** list. That worker's name will still appear in old records (e.g. work orders that the person worked on).

MainBoss indicates that data comes from a deleted record by ~~striking out~~ the data. For example, if you delete Joe Smith's employee record, all work orders that refer to that person will show the name as ~~Joe Smith~~. This shows that the associated employee record has been deleted (and will reduce your confusion when you can't find Joe Smith's name in your list of `Active` employees).

The Active Filter

The *active filter* determines which work orders, requests, and purchase orders count as `Active`. In other words, it determines what a table will display when the word `Active` appears in the table's bottom right-hand corner.

> Work orders, requests and purchase orders are always considered active until they are closed or voided. For example, if a work order is open, its age doesn't matter—it will be active until you close it.

To set your active filter, go to the **View** menu and click **Change Active Filter**. This opens a window that contains the following:

> **Show only records less than this many days old**: You can set this field's value to a number. If you do, work orders, requests, and purchase orders will only be considered `Active` if their most recent history record is less than the given number of days old. For example, if you set the value to 400, the only active work orders, requests and purchase orders will be ones whose histories have changed in the past 400 days. Typically, this means you won't see any work orders, requests and purchase orders that were closed more than 400 days ago.

> **Show only records since**: You can set this field's value to a date. If you do, work orders, requests and purchase orders will only be considered `Active` if their most recent history record was created after the specified date. For example, if you set the value to `January 1, 2009`, the only active work orders, requests and purchase orders will be ones which have had some activity since that date.

The options you set through the **View** menu only affect your own MainBoss sessions. For example, if you only want to see very recent records, you might set "**Show only records less than this many days old**" to 30. MainBoss would only show you work orders, requests and purchase orders less than 30 days old. You can change this number at any time.

> You can use **Administration** | **Company Information** to set active filter defaults for everyone who uses the MainBoss database.

Table Filters

A *table filter* tells MainBoss to display a subset of all the records that are in a particular table. In other words, it "filters out" records that you don't want to see.

For example, suppose you want to see which jobs are scheduled to begin today. You could set up a filter on the **Work Orders** table saying that you only want to see work orders whose "**Work Start Date**" is today. MainBoss would then filter out any work orders that started on some other day, so that the table viewer only displays work orders starting today.

As another example, suppose you only want to see information on units in your HVAC system. You could set up a filter on the **Units** table saying that you only want to see work orders whose "**System**" is HVAC. MainBoss would then filter out non-HVAC unit records, so that the table viewer only displays HVAC units.

You can specify a filter in any table viewer. To do so, click the filter set-up button beneath the table's list of records:

This opens a window similar to the one used to search for individual records. Use this window to specify which records you want to see. For example, suppose you're in the **Items** window and you specify
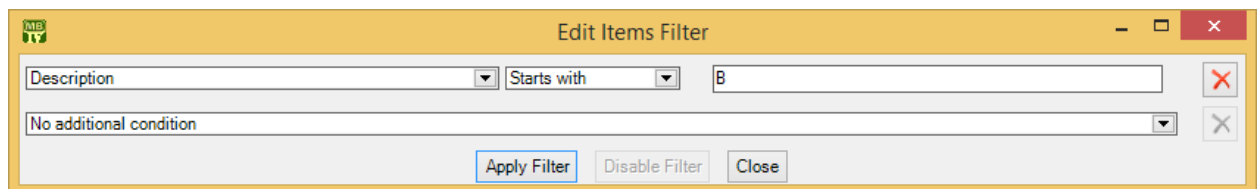
This indicates that you only want MainBoss to display item records whose "**Description**" field starts with the letter B. Once you enter this information, click the **Apply Filter** button. You'll see that the table viewer changes so that it only shows records that match what you've specified. If you **Close** the filter window, the table remains filtered. To return to the table's usual state, click the ✖ button associated with the filter.

When you specify one condition, MainBoss gives you a new line where you can specify an additional condition. Each time you specify a condition, MainBoss gives you a new line for more.

If you specify multiple conditions, MainBoss only shows records that satisfy *all* the specified conditions. For example, suppose you're filtering the **Requests** table. One condition might specify that you only want to see requests on a particular unit. A second condition might specify that you only want to see requests with a particular priority code. MainBoss will then display all requests on that unit with the given priority code.

The filter set-up window contains the following:

Fields, filter conditions, values: Select the records to be displayed. You specify your selections in the same way that you specifying search conditions when searching for a particular record.

Apply Filter: If you click this button, MainBoss immediately filters the table using the given criteria. Note that the table does not get filtered until you click this button.

Disable Filter: If you click this button, MainBoss stops filtering the table—you see all the records you normally would.

Close: Closes the window where you specify filters. If you have not clicked **Apply Filter**, MainBoss will *not* filter the table. Furthermore, MainBoss will not record any filter conditions you've specified—the next time you open the filter window, you'll start from scratch again.

On every table viewer, there is a checkbox labeled **Apply Filter**. If this box is blank, the table is unfiltered—you see everything you'd usually see. If you checkmark the box, the table will be filtered using the filter that is currently defined. By blanking or checkmarking the box, you can switch back and forth between filtered and unfiltered displays.

Some filters let you specify patterns rather than specific character strings. For example,

```
       Description    Matches Pattern   .*furnace.*
```

corresponds to any `Description` which contains the string "furnace" anywhere. (The `.*` pattern is a particularly useful one which stands for any string of zero or more characters.) Patterns are often called *regular expressions*, and MainBoss supports any regular expression as defined in the Microsoft .NET environment. For more information, see

`http://www.codeproject.com/KB/dotnet/regextutorial.aspx`

> Request, work order and purchase order tables are also affected by the *active filter*. For more, see *The Active Filter* on page 16.

Customized Table Settings

Table viewers can be *customized*. The customization area is above the table, consisting of a drop-down list and a drop-down button:

A customized setting is name that stands for a particular set of table display options, especially filters (see *Table Filters* on page 16). For example, suppose your organization manages a lot of properties, but you yourself only work in Building A or Building B. You might create three customized settings for the **Units** table, named A, B, and A&B. If you choose the A setting, the **Units** table will only show you the units in Building A; similarly for the B and A&B settings. In this way, you only see the units that matter to you, not any units in other buildings.

> **Note:** Settings that can be saved are indicated with blue lettering rather than black. For example, "**Apply Filter**" will appear in blue in table viewers, indicating that filter settings can be saved as a customized setting for the associated table.

To create a customized setting, you first set a filter that chooses only the records you want to see. For example, if you only want to see units in Building A, you could set a filter like
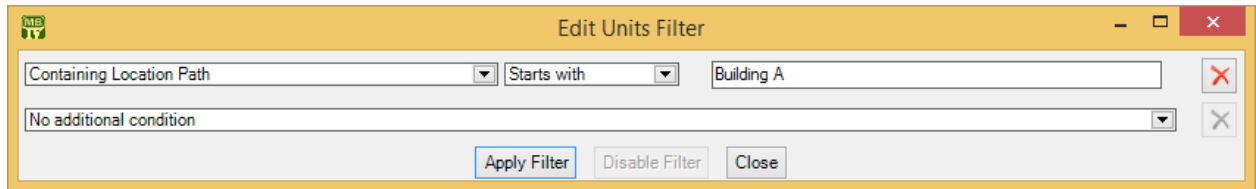

You then click **Apply Filter** and then **Close**. You'll see that the table only shows the units whose containing location is Building A.

To give this table setting a name, click the drop-down arrow associated with the drop-down button in the customized setting area. In the resulting menu, click **Save As**. A window will appear letting you specify a name for the filter you've just set up:
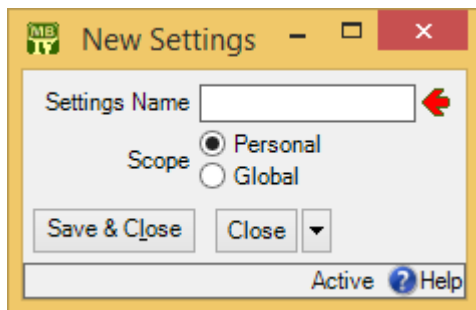
In "**Settings Name**", enter a name for this setting (e.g. A to stand for Building A).

The scope of your settings may be **Personal** or **Global**. If you choose **Global**, then all people using your MainBoss database will see the name of this customization in the drop-down list of possible customizations. (Only people with the Administration security role may create **Global** customizations.) If you choose **Personal**, this name will appear in your own list of possible customizations, but not in anyone else's.

Note: You may wish to create a number of standard global customizations that can be used by anyone in your organization. This is particularly helpful if some of your users aren't knowledgeable enough about MainBoss to create customizations of their own.

Click **Save & Close** to save your setting and close the window. Once you've done this, you'll see the name of the customization appear in the accompanying drop-down list. Anytime you come to this table, you can quickly apply the setting by selecting the name from the list. Once you've chosen a customization, MainBoss uses it every time you come back to this window until either you quit MainBoss or you choose a different customization or filter.

The drop-down button for customizations contains the following choices:

Save: Saves the current filter settings under an existing name. This is how you change the definition of an existing customization.

Save as: Saves the current filter settings under a new name.

Delete: Deletes the name that's currently selected in the customization drop-down list.

Set as Default: Specifies that the selected customization should be treated as the default for this table. In this case, you don't have to select the customization—whenever you start MainBoss, the customization is automatically applied. For example, if the default customization for your **Units** table only shows units in Building A, then the table is automatically filtered that way whenever you start MainBoss. You will never see units that are in other locations, unless you change the filter or choose a different customization.

Form Customization

MainBoss lets users with the `Administration` security role control what information is displayed in table viewers, editor windows, and report windows. This is done by means of *form customization*. Using form customization, you can reduce the amount of information being displayed; this may be useful if you don't care about some of the usual information, or if you're working on a small display screen. (Note that some information cannot be removed from a window, e.g. compulsory fields.)

It is possible to disable form customization. To do so, go to the **View** menu and click **Form Customization**. In the resulting menu, remove the checkmark on **Enable**.

If you are attempting to customize a window and it doesn't work, make sure that form customization is enabled (i.e. make sure that **Enable** has a checkmark beside it).

To customize a particular table viewer, editor window or report window, click the drop-down arrow beside the customization button:

(This button appears in the top right corner of table viewers and report windows, but the bottom right corner of editor windows.) After clicking the drop-down arrow, click **Customize Form** in the resulting menu.

MainBoss will display a window where you can use checkboxes to determine which information will appear in the associated form. Each table viewer, editor window and report window has its own customization form. For example, here's the form for customizing the **Work Orders** table viewer:
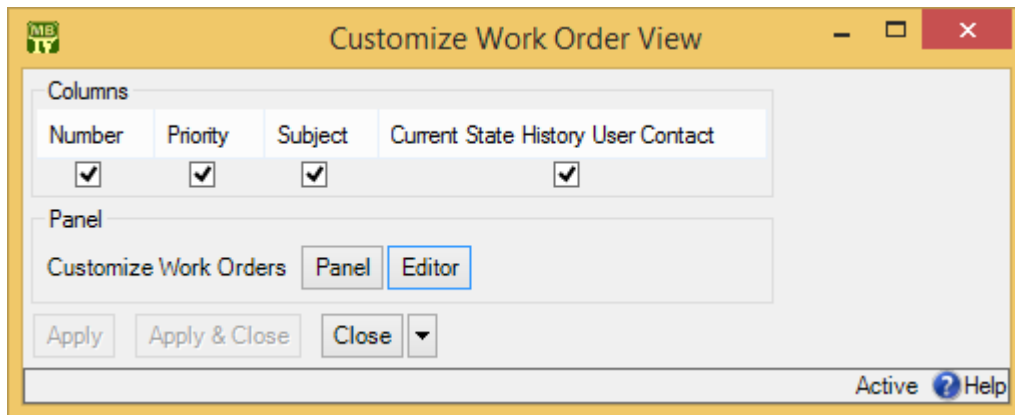
There is a checkbox for each column that usually appears at the top of the table viewer. If you blank out a checkbox, the corresponding column will no longer appear in the table viewer.

The **Panel** button in the customization form lets you customize the panel that appears at the bottom of the table viewer (under the list of work orders). The **Editor** button lets you customize the standard work order editor. If you click either button, MainBoss will open another window where you can use checkboxes to specify which information will and won't appear in the corresponding location.

The customization form also has a button labeled **Customize Work Order Report Form**. This lets you customize the standard report for printing work orders (**Work Orders** | **Print**). It is included here so that the same window lets you customize the work orders table viewer, the work orders editor, and the work order report.

After you have specified any changes you want to make to a form, click **Apply** to save those changes. Click **Apply & Close** to save the changes and close the customization window, or click **Close** just to close the customization window.

Note: If you customize a form, it will no longer match the help files associated with the form. For example, if your customization leaves out certain fields, the help files will still describe those fields. This may confuse people who wonder why the help file doesn't match what they see on the screen.

Publishing Your Customizations: By default, any customizations you have made will be discarded when you quit MainBoss. If you want to keep your customizations from one MainBoss session to the next, you have to *publish* them.

To publish your customizations, go to the **View** menu, click **Form Customization**, and then click **Publish**. This saves *all* your current customizations. From this point onward, those customizations will be used every time MainBoss starts. If you make a new customization, then want to go back to your old customizations, you can click **Restore from published** in the same menu.

You can use **Clear all customization** from the same menu to get rid of all the customizations you have made. However, you'll go back to your published

customizations the next time you start MainBoss, unless you **Clear all customization** and then **Publish** (so that the "cleared" customizations are your "official" ones).

Exporting Customizations: You can save your current customizations in a file. This makes it easy to restore the customizations later if you wish, or to switch between different sets of customizations.

To save your current customizations, go to the **View** menu, click **Form Customization**, and then click **Export**. MainBoss will ask you to specify the name of a file to hold your customizations.

At a later date, you can use **Replace from file** in the same menu to get rid of any current customizations and instead use the ones you exported to the file. You can use **Add from file** in the same menu to add the saved customizations to any other customizations that are currently in effect. With **Add from file**, if a saved customization conflicts with one of your current customizations, the saved customization replaces your current one.

Selecting Multiple Table Entries

A table viewer lets you select multiple records from the table. To do this, you use the standard Windows conventions:

☐ If you hold down the **<Shift>** key, click on one record, then click on another, you will select both records and all the records in between.

☐ If you hold down the **<Ctrl>** key and click on a record, you select that record. If you hold down **<Ctrl>** and click a different record, you select that record and the first record stays selected. In this way, you can use **<Ctrl+Click>** to select any number of separate records.

After you select multiple records, you can perform the following operations:

✕: Deletes the selected records. A window will open to ask if you really want to delete those records.

🗎 : Opens the first selected record in an editor window. This window offers you all the features available in a normal editor window for this type of record. In addition, it has a **Next** button; if you click **Next**, MainBoss closes the current window and opens one where you can edit the next record that you selected from the table viewer. By repeatedly clicking **Next**, you can move through the records that you've selected, editing each one as you wish.

If you use the editor window to change the contents of a record, the **Next** button disappears until you click **Save** (to save your changes) or **Cancel** (to cancel the changes). You will not be able to go to the next record until you save or cancel your changes.

Next is a drop-down button with the following entries:

Previous: Closes the current window and opens an editor window to edit the previous record from the list of selected records. By using **Next** and **Previous**, you can go forward and backward through the list of selected records.

Close: Closes the current window and goes back to the table viewer. This says that you are finished using **Next** and **Previous** to edit the selected records.

Cancel: Works like the usual **Cancel** button, canceling any changes you have made.

: Works exactly like except that you can only look at the contents of the selected records, not make changes. You can use **Next** and **Previous** buttons to go forward and backward through the list of selected records.

Restore: Is only enabled if the records you have selected are all currently deleted. If so, MainBoss opens an editor window that works exactly like . If you **Save** the current record, it will be restored. You can use **Next** and **Previous** to move forward and backward through the list of selected records.

Right-Clicking: When looking at a multi-selection list, you can right-click on the list to get a menu of useful actions that can be performed. For example, the menu gives you the option of checkmarking all the entries in the list or blanking them out. With long lists, we recommend that you click the "..." button to get a selection window, then right-click on the list in the window to select the entries you want.

Editors

An editor is a window where you can enter information. Most editor windows have a **Details** section where you can enter basic information; many editor windows have more sections for additional information.

☐ Mandatory fields are marked with a red arrow until they're filled in. You can't save the record until you specify values for those fields. (In the picture below, "**Subject**" and "**Unit**" are mandatory.)

☐ Read-only fields can't be changed. They can be recognized because they are *not* surrounded by a black border line. In the picture above, several fields are read-only. For example, the fields in the "**Current**" line and "**Business Phone**" are all read-only.

☐ Some fields can only be edited at certain times. For example, some fields on a purchase order can't be changed once the purchase order is issued—issuing the purchase order implies that the purchase order is ready to be sent to the supplier, at which point, some information is considered "unchangeable". (Actually, some information *can* be changed, but you have to withdraw the purchase order before you're allowed to make the change.)

MainBoss also prevents you from changing information that has already generated accounting records—for the sake of security, accounting records can't be changed or deleted. For example, when you record the use of materials in connection with a work order, MainBoss generates accounting records indicating the cost of those materials. You can't directly edit this information once the accounting records have been written; if necessary, you must create a *correction*, which will generate a *new* accounting record to correct any mistakes in an old one.

☐ As shown above, editor windows have several buttons containing the word **Save**:

Save: Just saves information you've entered.

Save & Close: Saves the information and closes the window.

Save & New: Saves the information and starts a new record. This lets you create one record after another; for example, if you're recording contact information for people, you can **Save & New** after each person, which saves what you've just entered and starts a record for someone else. (Fields in the new record are initialized with default values as set in the **Defaults** section of the appropriate table viewer.)

Save & Print: Saves the information and opens a window that will let you print the record you've just created.

General Tip: When you're filling out a record, if you don't know what value a field should have, leave it blank. Filling in dubious values will cause confusion; finding values that need to be changed can be difficult. If MainBoss really needs a field to be filled in, it uses the red arrow to mark the field as mandatory.

Drop-Down Lists (Pickers)

Editor windows often specify information using *drop-down lists*, also known as *pickers*. For example, the "**Requestor**" field in a work request editor is a drop-down list:

As shown above, a drop-down list ends with three symbols:

☐ The left-most arrow displays a list of available values for the field. In the case of "**Requestor**" this will be a list of the people whom you've authorized to submit work requests.

☐ The "..." opens a window showing the complete table. For example, if you click the "..." for "**Requestor**", you get the **Requestors** table. Click an entry in the table, then click **Select** to fill in the original field.
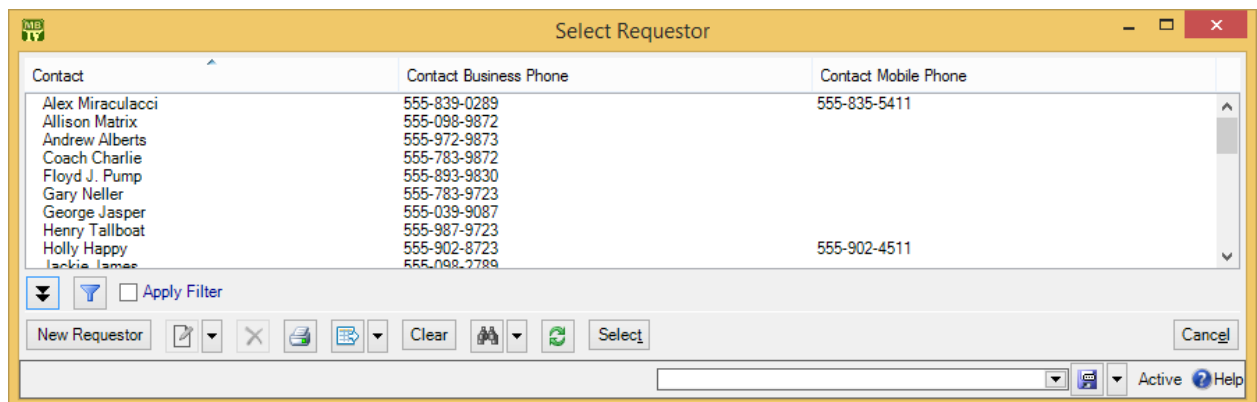
☐ The right-most arrow of the drop-down list opens a menu that gives you a number of options. For example, if you click this arrow for the "**Requestor**" field, you'll see the following:

Browse lets you look at the existing **Requestors** table, just like the "..." button. **New Requestor** lets you create a new requestor record; if you do this, the requestor will be filled into the original "**Requestor**" field once you save and close the new record. This makes it easy to create new records as needed if you don't already have an appropriate record. **Edit** lets you edit the selected requestor record, while **View** lets you look at the record (but not change it). **Export**, **Import**, and **Save XML Schema** are related to importing and exporting data; for more information, see the MainBoss Reference Manual. **Clear** deletes the current contents of the field (if any).

Errors

MainBoss puts a marker beside data fields that contain invalid data. The error marker is a red circle containing a white exclamation point. This is shown below, where the "**Effective Date**" field should contain a date, but instead contains meaningless text:

If you try to save a record when it contains an invalid field, MainBoss will display an error message and won't save the record.

To find out why a data value is invalid, move your cursor to the error marker. MainBoss will display a tooltip with a message explaining why the current value can't be accepted. This is shown below:
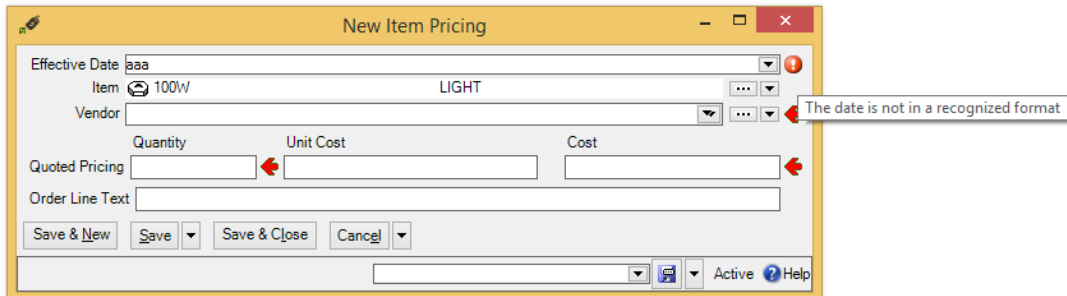
Concurrency Errors: In the rare case that two people try to edit the same record at the same time, MainBoss will report a *concurrency error*.

For example, suppose you start editing a **Units** table record and before you're done, someone else tries to edit the same record. The first person to save the record will succeed. The second person to save the record will receive an error message indicating that the record has changed while you were making your changes:
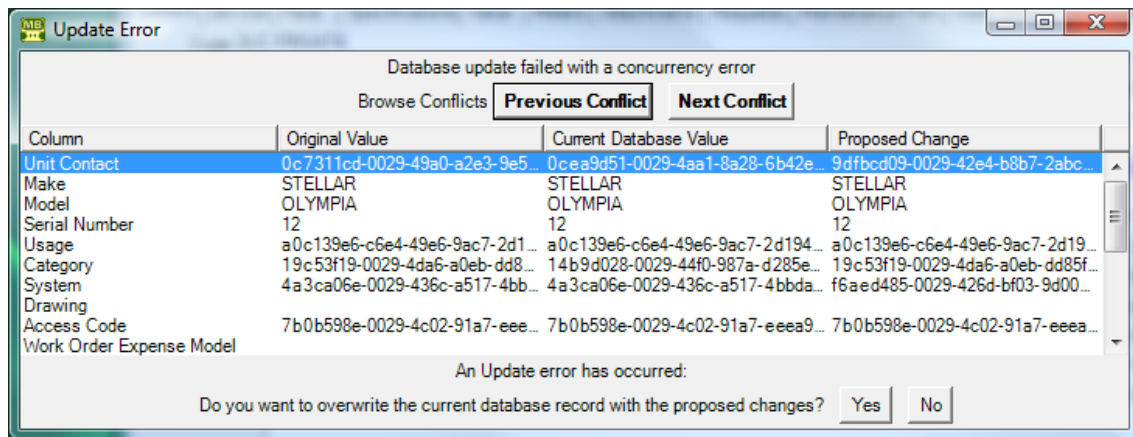
MainBoss is asking you to resolve conflicts between the changes that were just made. The error message lists the fields in the record. For each field, there are three columns:

☐ The first column shows the record's values at the time you began editing

☐ The second column shows what the values are now (the ones that the other person saved)

☐ The third column shows the new values you've specified yourself

Where the values are different, you can see fields where your values differed from the other person's. The **Next Conflict** and **Previous Conflict** buttons jump to fields where the values conflict.

In many records, some field values will look like nonsense strings of characters. These strings are actually links to other tables. For example, the "**Category**" field refers to an entry in the **Unit Categories** list. The field's value is a meaningless string that serves as a pointer. You can see if two pointers have the same value, but you can't tell anything about the record they point to.

When conflicts occur, it's best to click the **No** button on every conflict; this basically says, "Ignore what I just did.") Then just go back and start editing the record again. This time you'll see the values that the other person put in. You can then decide which of these you want to change.

Reports

Reports can display and analyze your maintenance information. Many sections in the control panel have subsections that offer reports; for example, **Work Orders** | **Reports** offers you a number of reports dealing with work orders.

When you choose the control panel entry for a report, you get a window that lets you specify options for the report:
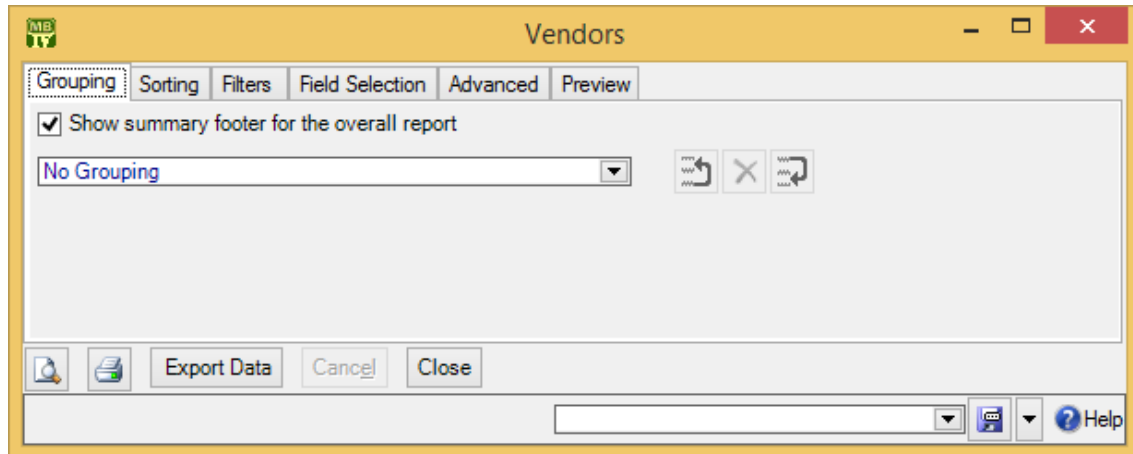
Report Sections

The window associated with a report organizes options into sections.

Grouping and **Sorting**: Let you control how the report is organized. For example, suppose you're printing information on work orders. The **Grouping** options will divide the report into sections. For example, you might group your work orders by the location where the work will take place. Inside each group, you can

specify subgroups if you wish.

The **Sorting** options tell how to sort records each each group and subgroup. For example, if you are grouping work orders by location, you might sort each group section by the date the work orders are supposed to begin. You can specify additional sorting options as well. For example, if you are sorting group sections by starting date, within each such date, you might choose to sort by expected duration; in this way, shorter jobs would be listed before longer ones.

For both **Grouping** and **Sorting**, you can specify either ascending or descending order. For example, if you sort records by date, you can go from earliest to latest or vice versa.

Each grouping/sorting option is followed by three buttons:

These let you change the position of the option within the list of options or delete the option entirely.

Grouping options break a report into sections. Each section has headings to indicate the start of the section, and may have footers as well. If a section has more than five entries, the total number of entries will be shown in parentheses after the group heading. *Sorting* options sort information within each section; they do not result in headings.

If a particular group has no elements, the group will not appear in the report. For example, suppose you're printing information about inventory items, grouped by item category. If a particular category has no items in it, that category will be omitted from the report.

Filters: A *report filter* lets you select which records you do and don't want in the report. For example, you might want to restrict a report about units to units in a particular category or range of unit IDs. Filter sections let you specify such restrictions.

Field Selection: Lets you specify what information should be included in the report. The section lists all the pieces of information that *may* be included in the report. Each possibility has a checkbox beside it; if you checkmark the checkbox, the information will appear in the report. Otherwise, it won't.

The first time you go to a print window during a MainBoss session, a default set of checkboxes will be checkmarked. If you change any of these, MainBoss will remember your changes throughout the rest of the session. However, when you quit MainBoss, everything gets set back to the defaults.

For some reports, the **Field Selection** window may have special checkboxes. For example, some reports have a **Suppress Costs** checkbox; if you checkmark this checkbox, any money amounts that would otherwise be printed will be omitted from the report.

Some print windows will have a number of checkboxes at the bottom of the page under the heading **Show Child Records**. These checkboxes let you choose extra optional information that may be included in the report.

The columns in a report are made wider or narrower, depending on how many pieces of information you've asked for. Asking for many different data fields will shrink the width of each column and may make the report difficult to read. Printing a report in landscape format rather than portrait may improve readability. Another option is to export the report to Microsoft Excel; if you do that, you can avoid the problems that arise from trying to cram a lot of information onto a limited piece of paper. One other solution is to use the "**Report width in pages**" option discussed below.

Advanced: Lets you specify a title for the report when it is printed, the fonts to be used, and additional information.

In some reports, **Advanced** options let you obtain *summaries*. A summary typically gives totals and sub-totals of monetary amounts for the groups and sub-groups of the report. In some cases, the summary may also give you averages (e.g. the average lifespan of a particular kind of equipment). If you do not specify **Grouping** section options, summaries may not give you much information.

Report width in pages: By default, reports are one page wide (where the page width is dictated by your current printer settings). If a report contains many fields, you can set the value of this field to a number higher than 1; for example, if you set the field to 2, each line in the report will be two pages wide. This allows more space for the information you want to see.

When a report is more than one page wide, the effects vary depending on whether you look at the report on your monitor screen or whether you actually print it out. In a screen preview, you can simply use the usual scroll bars to look at lines that are too wide to fit on the screen. In a printed report, each line of the report will be the given number of pages wide. For example, if the report width is 2 pages, each line will be two printed pages wide. You could then tape pages together to get a more readable report.

In general, "**Report width in pages**" should be set to 1, unless you actually need wider lines. If you specify a large page-width when you don't need it, the report will look worse than if you use the default width. In addition, a large page-width is likely inappropriate when you're printing single work orders for workers or purchase orders for vendors—the results will probably be harder to read than the default format.

Preview: Displays a preview of the report—a version of what you'd see if you actually printed the report on paper. Preparing such a preview may take some time, since MainBoss may have to process a lot of data; therefore, you might have to wait while MainBoss creates the preview.

The **Preview** section has buttons that let you specify which printer you want to use, the size of page margins, and other printing details. Another button lets you export data from the report (e.g. to Microsoft Excel). Finally, the **Preview** section has a button that will actually print the report.

Depending on the contents of your database, certain combinations of options in **Grouping**, **Sorting**, **Filters**, **Field Selection** and **Advanced** may result in a report with little or no useful information. You are encouraged to experiment with the options to find combinations that are particularly helpful to your work.

Filtering on Locations and Units: Various reports let you filter by locations and/or units. In general, if you use a filter to select one or more locations, the report does *not* cover sublocations (unless you select the sublocations too). Similarly, if you use a filter to select one or more units, the report does *not* cover sub-units (unless you select the sub-units too).

However, when you filter on locations, the filter window offers you the option **Location and all sublocations**. If you select this option, MainBoss lets you specify a single location; the associated report will cover that location and all sublocations within the location. Similarly, when you filter on units, the filter window offers you the option **Unit and all subunits**. If you select this option, MainBoss lets you specify a single unit; the associated report will cover that unit and all subunits within the unit.

Report Buttons

Every window for printing a report contains a number of buttons at the bottom:

    🖨: Immediately prints the report, using whatever sorting and filtering options you have specified. You can use this button if you don't want to preview it first. You can cancel printing with the **Cancel** button.

Export Data: Exports the report's data in XML format. The result is primarily intended to serve as input for user-written programs which process the data in some way. This contrasts with the **Preview** export facilities (discussed below), which are most useful for embedding in Microsoft Word documents and other "unprocessed" contexts.

Clear Select for Printing: If you click this button, MainBoss removes the checkmark from all **Select for Printing** boxes on all work orders. Typically, you do this after printing work orders that have been marked **Select for Printing**.

Cancel: Cancels the preparation and printing of a report. This button is enabled only when MainBoss is preparing a report, i.e. gathering the data and formatting it. Once the report is ready to print, MainBoss hands off the result to the Microsoft Windows printing facilities; after the report has been handed off to Windows, MainBoss has no more control over the print-out and the **Cancel** button won't work. (If you want to cancel the print-out after that, you must use the standard

Windows printer control facilities.)

Depending on the configuration of Windows and the Microsoft Report Viewer, canceling a report may result in a diagnostic message. Typically, you can just ignore this message.

Close: Closes the window. (This button is only present if you opened a separate window to print the report.)

↻: Goes to the **Preview** section of the window and creates a preview of the report. Preparing such a preview may take some time.
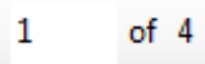
The Preview Window

The **Preview** section of a print window contains a control bar like this:

| ◄◄ ◄ 1 of 46 ► ►► | ← ⊗ | 🖨 📄 📖 💾▾ | 100% ▾ | Find | Next |

This bar contains the following (from left to right):

> If you click an entry in the left-hand panel of the preview, the preview changes to show you that entry.

◄◄   First Page button: Goes to the start of the document.

◄   Previous Page button: Goes to the previous page of the document.

1 of 4   Gives the number of the page that is currently being displayed and the total number of pages in the report. You can go to a specific page by typing the appropriate page number into the box. (The number of the last page is not finalized until the entire report has been generated. This means that the number may not appear when a preview begins, and the number may also change as report generation continues.)

►   Next Page button: Goes to the next page of the document.

►►   Last Page button: Goes to the last page of the document.

←   Parent Report button: Only applies when a report contains sub-reports, and this feature is not currently used.

⊗   Stop button: Stops MainBoss processing the report. You might click this if MainBoss is preparing a lengthy report and you decide you don't want to see the report after all.

Print button: This is the button that actually prints the report.

Print Layout button: Switches the format of the preview display.

By default, the left side of the display contains an index of the information in the report. Clicking an entry in the index will move to the corresponding information. The mouse wheel can move forward or backward through the report but it doesn't necessarily move backward or forward by a whole number of pages.

If you switch to the alternate format, you see a scaled-down version of the page as it will actually print. The mouse wheel will move from one page to the next.

Page Setup button: Opens a window where you can specify paper size, margin size, and other printing information.

Page Setup information specified within MainBoss always overrides any default options associated with your printer, even if you specify printer options at the time you print the report. For example, suppose you specify a page setup of Landscape orientation, but when you go to print the report, you specify Portrait. The Landscape overrides Portrait because the "page setup" in MainBoss overrides anything you specify directly with the printer.

Page Setup options specified for one MainBoss report do not affect options for different MainBoss reports. For example, if you set options when printing work orders, those options will not affect the Page Setup for printing any other MainBoss report. The options will also be forgotten when you quit MainBoss.

Export button: Exports information from the report into a file. If you click the down-arrow associated with this button, you'll see a list of available export formats. The formats currently supported include Microsoft Excel, Microsoft Word, and Acrobat PDF.

Once you specify what type of output you want, MainBoss will open a window asking you to specify a file where the exported data should be stored. Loosely speaking, MainBoss exports the data in a format similar to what's displayed in the preview. For example, if you export data in Excel format, the result can be read into Excel (or embedded in a Microsoft Word document) to produce something that looks very much like the original report. If, on the other hand, you want to export data for processing by some user-written program, it's probably better to use the **Export Data** button (discussed previously).

> **Note:** When you export data in PDF format, MainBoss uses the margin sizes and paper size as specified in the report's current **Page Setup** settings.

100% ▾   Scale: Controls the scale of information displayed in the preview window. For example, if you set the scale to 200%, you'll see the information at twice its normal size. If you set the scale to 50%, you'll see it at half its normal size.

Find | Next   Searching: To search for information in the report, type the text you want to find in the box, then click "Find". For example, if you're looking at a full Units report and want information on a particular unit, type the unit name in the box and click "Find". Clicking the associated "Next" button goes to the next occurrence of the same text.

Paper Size and Margins

When you start MainBoss, it sets the margin sizes for any report to default values, and sets the paper size to the default size associated with your default printer. You can change both the margins and the paper size by clicking the **Page Setup** button in the **Preview** section of any window that prints reports.

> **Note:** Any change in margin size or paper size only affects the current printing window. Different reports will still have the default margin size and paper size. Also, if you close a report window then open it again, the margin size and paper size for that report will be set back to the defaults.

When you export data in PDF format, MainBoss uses the margin sizes and paper size associated with your current **Page Setup**. This means that you might use **Page Setup** to specify a paper size and/or margin sizes that your printer can't actually handle, if you intend to export the report in PDF format.

Formatting in Columns

Many reports have a **Show Report in Columns** option in the **Advanced** section. This organizes a report so that there is a single "line" for each record in the report, with the fields of each line broken up into separate columns. This format can be useful if you only want to see a few pieces of information from each record (as dictated by the **Show** checkboxes). However, it becomes unreadable if you try to cram too many pieces of information on a single line.

Note that if you print out the report in *landscape* format rather than *portrait*, you can put more on a line without losing readability. To change the format, use the "Page Setup" button near the top of the **Preview** section.

The column format is particularly useful if you intend to export the report's data to Microsoft Excel. In this case, it doesn't matter if the columns don't fit well on a piece of paper—in Excel, you can expand the columns so that they're readable on your computer screen.

Charts

A chart is special type of report that shows information in a graphical way. A chart report consists of one or more bar charts running either vertically or horizontally.

Some charts use different colors to distinguish different types of data on the same chart. For example, preventive maintenance work orders may be shown in a different color than corrective work orders.

Often the **Grouping** section of a chart report window will determine how chart information is organized. For example, the **Costs by Grouping** chart shows work order costs organized by whatever option(s) you specify in the **Grouping** section. If you choose **Work Category**, costs are broken down by work category; if you choose **Unit Location**, costs are broken down by location; and so on.

Some data and option selections may lead to charts that are difficult to read. In such cases, we recommend using the **Filters** section to reduce the amount of data you're trying to show in a single chart. Breaking your data into multiple charts may make the final results easier to understand.

Interrelated Records

MainBoss records are often linked to each other. For example, a record describing a piece of equipment might have a link to a service contract record, indicating that the equipment is covered by the service contract. This link is roughly analogous to a link on a web page: it's a pointer that MainBoss can follow in order to find related information.

If record A links to record B, record B often links back to record A. For example, just as an equipment record can link to a service contract, each service contract record has links to all the units covered by the contract. This makes it possible for MainBoss to go from equipment to service contracts or vice versa.

Now suppose you're creating a new unit record and you go to the **Service** section. If you haven't yet saved the unit record, you'll see that the **New Unit Service Contract** button in the **Service Contracts** section is disabled. Why? Because the unit record doesn't exist until you save it. If you tried to create a new service contract record before the unit record was saved, there'd be nothing to link the service contract back to.

For this reason, the **New Unit Service Contract** button for service contracts stays disabled until you save the unit record. After that, the **New Unit Service Contract** button is enabled and you can use it to create a new service contract.

Note that if you hold the cursor on a disabled **New** button, a tooltip will explain the situation.

If this seems confusing, just remember this rule: **If you're in the middle of creating a new record, MainBoss may not let you create a connected record until you save the first record.**

If you find that a **New** button is deactivated, saving the current record will usually activate the button.

Accounting Facilities

MainBoss lets you keep track of various types of costs: the amount you pay for equipment, materials, outside contractors, etc., and the amount you charge for your maintenance department's work.

Note: MainBoss is not a full-fledged accounting package. However, it does keep track of the transactions needed by such a package. This information can be exported for use with an accounting package.

MainBoss's accounting facilities are intended to simplify day-to-day operations, so that workers using MainBoss don't have to know which ledger account is used for which expenses. In order to make this possible, someone familiar with your organization's accounting practices must set up a number of definitions that workers can then use.

The Accounting Module: If you have licensed the MainBoss **Accounting** module, you will have access to various special facilities for tracking accounting information:

☐ *Cost Centers*: The MainBoss **Accounting** module lets you define cost centers. Most organizations will set up a single cost center for each general ledger account used by the maintenance department. However, it is possible to set up multiple cost centers all corresponding to a single general ledger account. This makes it possible to have detailed cost breakdowns for your department's purposes, even if such detailed breakdowns aren't required by your organization as a whole.

For example, suppose you only use a single ledger account for material expenses. You can create a single cost center named `Materials` for that ledger account. However, you might decide to create multiple cost centers like `Electrical Materials`, `Plumbing Materials`, `Mechanical Materials`, and so on, all corresponding to the same single ledger account. This lets MainBoss give you a more detailed breakdown of your materials expenses, even though your actual accounting system doesn't subdivide expenses in this way.

If you have not licensed the MainBoss **Accounting** module, MainBoss hides all references to cost centers. Therefore, you won't define any cost centers and you won't see them used anywhere in the software.

☐ *Expense Models:* An expense model is a type of MainBoss record that describes how you're going to "cost out" a particular work order. For example, on jobs that you do for Client X, you might be required to categorize expenses in one way; on jobs for Client Y, you may have to use a different system of classification. As another example, organizations that receive government grants may have to write up expenses in different ways on grant projects than on other types of work.

When you create a work order, you choose the appropriate expense model for the job. You should have a separate expense model for all the different expense-reporting formats that you may have to use.

☐ *Expense Categories:* Each expense model has a set of associated expense categories. These are the categories that must be used to classify expenses in this particular situation. You might have categories like

```
Spare parts
Work materials
Labor from outside contractors
```

When you prepare a work order, you look at each expense and ask, "What *type* of expense is this?" You choose the expense category from the list of available categories associated with the expense model.

Each expense category within an expense model has an associated cost center. This indicates the general ledger account that should be used for that expense category within that expense model. Linking an expense category to a cost center is called an *expense mapping*.

As an example, suppose that a property management company creates two separate general ledger accounts for each tenant: one for labor expenses and one for material expenses. This might lead you to set up the following cost centers:

```
     Cost Centers         Ledger Account
TenantA Labor                 00001
TenantA Materials             00002
TenantB Labor                 00003
TenantB Materials             00004
TenantC Labor                 00005
TenantC Materials             00006
```

For each tenant, you create an expense model: `TenantA Model`, `TenantB Model`, `TenantC Model`. The `TenantA Model` would contain the following expense categories:

```
   Category            Associated Cost Center
    Labor                  TenantA Labor
    Materials              TenantB Materials
```

`TenantB Model` and `TenantC Model` would be similar.

Whenever you create a work order, you have to assign it an expense model. Suppose you're doing a job for Tenant A. You look at your list of expense models and see that there is one especially associated with Tenant A; you therefore assign that expense model to the work order.

Next you have to assign an expense category to each expense on the work order. When you look at the list of categories associated with `TenantA Model`, you see there are only two: `Labor` and `Materials`. Therefore, you just have to assign one of those two categories to each expense on the work order. This also associates a cost center with each expense, and therefore a general ledger account. Note, however, that when you fill out the work order, you only have to make two types of decisions:

☐ Which model do we use (what general approach will we use to "cost out" this job)?

☐ For each expense on the work order, which available expense category does it belong to?

Default Accounts: The MainBoss accounting facilities let you separate out expenses in a sophisticated way (e.g. if you keep separate accounts for each tenant). If your needs are more simple, MainBoss has default expense models and expense categories that you can use, with or without modification.

If you don't need accounting at all, just use the default expense model for all work orders. Choose the default expense categories for every item on the work order.

> Whenever you create a work order, you *must* specify an expense model. If you haven't set up your own expense models, use the default model.
>
> Similarly, whenever you add an item to a work order, you *must* specify an expense category. If you haven't set up your own categories, use the default ones.

Displaying Accounting Records: To display the accounting records that MainBoss creates, you use **Administration** | **Accounting**.

Security Roles

In order to use MainBoss, users need to be assigned one or more security roles. The available roles are listed in **Administration** | **Security Role**. The comments provided in each role record indicate what permissions that role provides.

Security roles affect what a user sees: if a user doesn't have permission to see a particular type of information, that information will be blanked out or missing entirely from MainBoss windows. Security roles also affect what a user can do; for example, if a user has permission to view work orders (`WorkOrderView`) but not to close them, the **Close Work Order** button will be disabled for that user.

Important: When information is blank because you don't have appropriate permissions, move the cursor to the blank; MainBoss will tell you what security roles will give you permission to see that information. Similarly, if a button is disabled, move the cursor to the button; MainBoss will tell you what security roles will allow you to use that button. Note, however, that in some cases, information will disappear entirely (e.g. sections of records just won't be displayed); in this case, you may not even know that something is missing.

To assign security roles to a user, you use the **Security Roles** section of a user record (**Administration** | **Users**). In order to decide what roles an individual user should be granted, you must consider what that user is expected to do in his/her job. For example, consider someone working on a help-desk and receiving problem reports from clients outside the maintenance department. What does that person need to do?

☐ He or she will need to create work requests describing problems; that requires `Request` permission.

☐ Help-desk personnel also need a small amount of information on work orders (so that they can tell a client, "Your work has been scheduled for Thursday"). This suggests the `WorkOrderSummary` security role.

☐ The `UnitSummary` role might also be useful. It gives help-desk personnel enough information about units to make sure they understand which unit the client is talking about.

Other roles may also be appropriate, depending on what help-desk personnel are expected to do, and on your organization's policies.

Related Security Roles: Security roles tend to come in related groupings; for example, `WorkOrder`, `WorkOrderView`, `WorkOrderFulfillment` and `WorkOrderSummary` are all roles related to work orders.

In such a grouping, some roles automatically include others. For example:

☐ `WorkOrder` automatically includes all the other roles in the grouping.

☐ `WorkOrderView` automatically includes `WorkOrderSummary`.

For this reason, there's no point in giving someone both `WorkOrder` and `WorkOrderView` (for example), because `WorkOrder` automatically includes all the permissions of `WorkOrderView`.

In general, a `View` role automatically includes the related `Summary` role and a role with a simple name automatically includes roles of the same name with extra words (e.g. `PurchaseOrder` automatically includes `PurchaseOrderView`, `PurchaseOrderSummary`, and `PurchaseOrderReceive`).

Testing Security Roles

It can be difficult to figure out the effects of various security roles, especially when a user has several different roles combined. To test what a particular user can and can't do, you can use the **Evaluate Security As** button in **Administration** | **Users**. (You must have the `Administration` security role to use **Evaluate Security As**.)

If you select a user in the **Users** table, then click **Evaluate Security As**, your MainBoss session changes to have the same security permissions as the selected user. If certain types of data are invisible to that user, they'll become invisible to you too; if certain buttons are disabled, they'll be disabled for you too.

In this way, **Evaluate Security As** lets you see all the consequences of a user's security roles. Once you've examined what the user can and can't do, you can return to your own security roles by going to the **Session** menu and clicking **Reset to user YOURNAME security**.

Note: If you use **Evaluate Security As** to evaluate the security roles of someone who doesn't have `Administration` or `AdministrationView`, the first thing that will happen is that you lose permission to see **Administration** | **Users**. This may look as if you're being told that you don't have permission to use **Evaluate Security As**; however, it really means that the person you're evaluating doesn't have that permission.

Evaluate Security As gives you the same security roles as the selected user, but it does *not* change who you are. In particular, you keep your own set of assigned requests, work orders and purchase orders. It also doesn't give you SQL Server permissions if you don't already have them. This can lead to unusual error messages if you try to do certain operations. Since you're partly one user and partly another, we recommend that you avoid doing significant work while using **Evaluate Security As**; it can lead to confusion.

If you use **Evaluate Security As** to "become" another user, unusual results may occur if you already have other MainBoss windows open. For example, suppose you have a window open for editing work orders and then you use **Evaluate Security As** to become a user who has no access to work orders. If you try to use the other window to edit the work order, you may receive odd error messages—you are currently supposed to be a user with no access to work orders, so MainBoss is confused about how you can have a work order window open. While this probably won't cause any lasting difficulties, we recommend that you close all other windows before you start using **Evaluate Security As**.

Roles for Configuring MainBoss

If you are configuring MainBoss, you should at least have the `CodingDefinitions` security role. It will also be useful to have `ItemSummary`, `PurchaseOrderSummary`,

and `WorkOrderSummary`. If you are entering information about units, you will need the `Unit` role; otherwise, `UnitView` or `UnitSummary` may be sufficient.

Bar Code Support

MainBoss can add bar codes to various printed reports including work orders, requests, purchase orders, storeroom assignments, and unit records. This provides a level of support for the use of bar codes in your operations.

To print bar codes on a report, go to the **Advanced** section of the report. On the line labeled **Bar Code Symbology**, choose the type of bar coding you wish to use. (There are numerous systems for representing data as bar codes, so you have to tell MainBoss which one you use.) Once you have specified a format, MainBoss will include a suitable bar code for each record in the print-out.

The bar code represents different information depending on the type of records being printed:

☐ For work orders, the bar code represents the work order's number (e.g. `WO 00000001`)

☐ For requests, the bar code represents the request's number (e.g. `WR 00000001`)

☐ For purchase orders, the bar code represents the purchase order's number (e.g. `PO 00000001`)

☐ For storeroom and storeroom assignment records, the bar code represents the value of the record's "**External Tag**" field; this field appears in the **Details** section of the associated record

☐ For unit, the bar code represents the value of the record's "**Unit Tag**" field; this field appears in the **Details** section of the associated record

When you read a bar code into MainBoss, MainBoss will open an editor for the entity associated with that bar code. For example, suppose you want information about a unit. If you read the bar code on the unit, MainBoss will immediately open the appropriate unit record in an editor window; this lets you obtain all the usual information from the unit record.

> **Important:** Make sure that a window containing MainBoss's control panel is the active window at the time you read the bar code. If any other type of window is active (e.g. a window for editing a record or for printing a report), reading the bar code will not have an effect.

Default Bar Code Encoding: You can set your organization's default bar coding by going to **Administration** | **Company Information** and clicking **Edit**. This opens a window where you can specify which type of bar coding is most often used by your

organization. From this point on, that type of bar coding will always be set as the default in any report that can print out bar codes.

As with all default values, you can specify a different value in any specific situation. For example, if you don't want a particular report to include bar codes, you can set **Bar Code Symbology** to `None` when you print that particular report.

Drawbacks to Bar Codes: Many would-be bar code users think they can simply use the standard "universal" UPC bar codes that appear on most products (such as the codes that appear on grocery items in the supermarket). However, these usually aren't satisfactory.

For example, consider standard D cell batteries. To most maintenance departments, these are all the same. However, each brand of battery comes with a different UPC code; also different packages from the same manufacturer have different codes, so a two-battery package is different from a four-battery package. You might like a single "D cell" code for all such batteries, but you can't use the standard UPC code for that purpose.

A second problem is that manufacturers recycle UPC codes. Rubbermaid is well-known for this practice. When they discontinue a product, they re-use the old code for something new. They also use the same code for products with minor differences, e.g. similar products with different colors. This may or may not make a difference to your operations.

Similar problems apply if you use anyone else's bar codes, e.g. bar codes linked to a particular supplier's catalog numbers. If you use bar codes that come from anyone else, you may be caught by surprise changes or inconsistencies. For reasons like this, you will probably have to create your own bar codes for anything you want to identify. This can take a great deal of work, and must be planned carefully.

Note: MainBoss includes Barcode software developed by Neodynamic SRL (http://www.neodynamic.com/). All rights reserved.

Locations and Contacts

This chapter discusses the most important tables in MainBoss: **Locations** table and **Contacts**. These tables talk about the most fundamental aspects of your operation: the places and the people you work with.
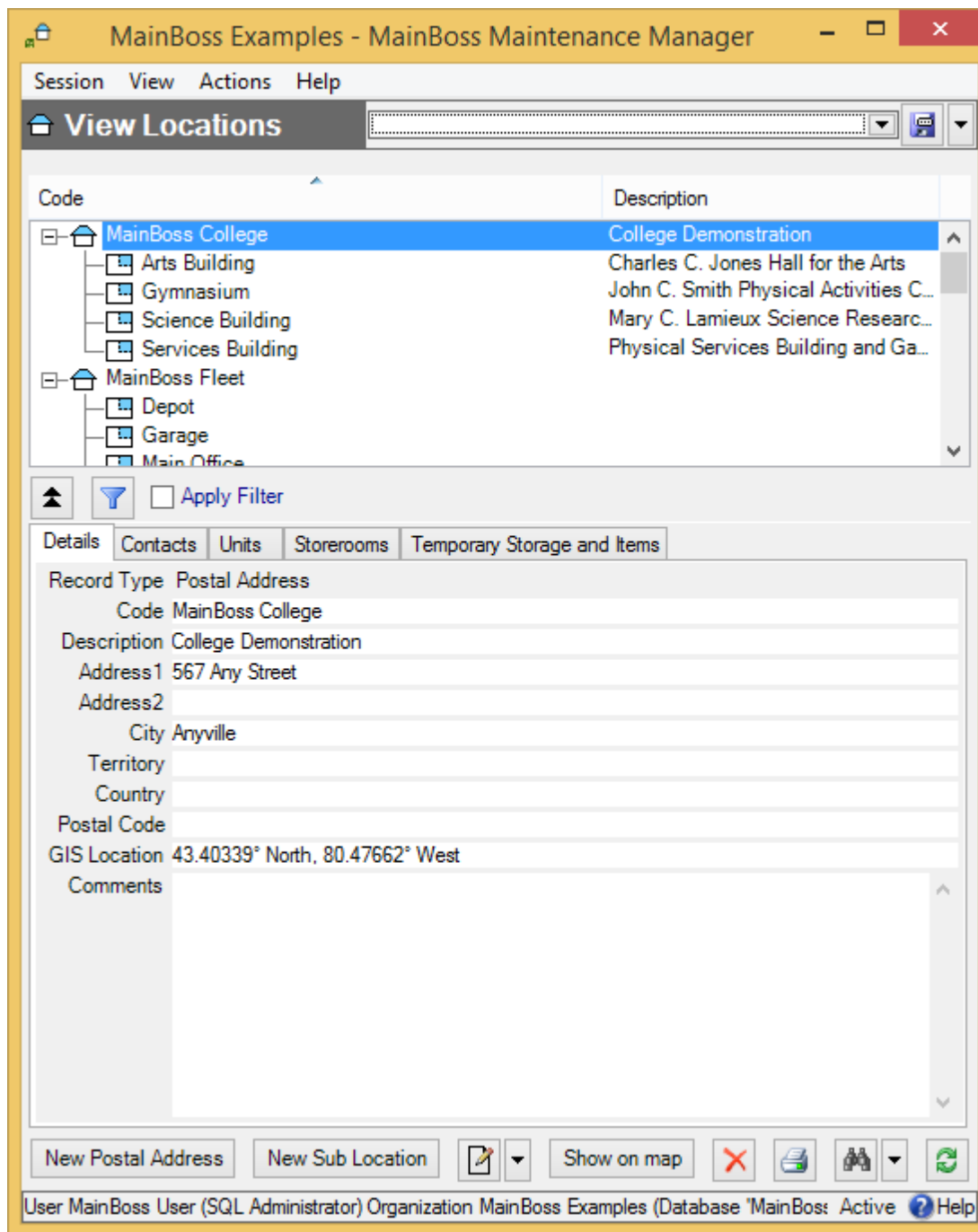
Locations

The **Locations** table lists locations that are of interest to your work. Examples include:

☐ The building sites where you perform maintenance operations

☐ Home addresses of your personnel

☐ Business addresses of your suppliers

☐ Any other addresses or places you want to record

In other words, the **Locations** table is like an address book, containing any geographical information you consider useful.

At a minimum, your **Locations** should list all the buildings covered by your maintenance operations. Almost everything you do with MainBoss requires an associated location. When someone submits a problem report, you have to specify where the problem is; when you create a work order, you have to tell workers where they should go; when you track spare parts, you have to tell where the parts are stored.

Here's a simple **Locations** table:

This table has two types of locations:

- *Postal addresses* specifying a single building (or a set of buildings that all have the same postal address, as in a college campus). When you create a postal address record, you specify all the usual address information (street address, city, state or province, etc.).

☐ *Sub Locations* which are locations inside other locations. When you create a sub location record, you specify the location that contains the sub location. (The containing location must already have a record in your **Locations** table.)

Sub locations can have sub locations of their own. In the preceding picture, `Room 100` is a sub location of `Floor 1` which is a sub location of `Our Building`. You can organize your locations and sub locations in any way you like.

Notice that MainBoss depicts locations using a *location map*. This is a picture similar to the ones that Windows Explorer uses to show folders, subfolders, and files. The map makes it easy to see which locations contain which sub locations.

Postal Addresses

Here's a typical example of a postal address record:

As shown, it contains a brief identifier code, a longer description of the location, and standard address information.

□ **To create a postal address record:**

1. In the control panel, click **Locations**. This opens the **Locations** table viewer.
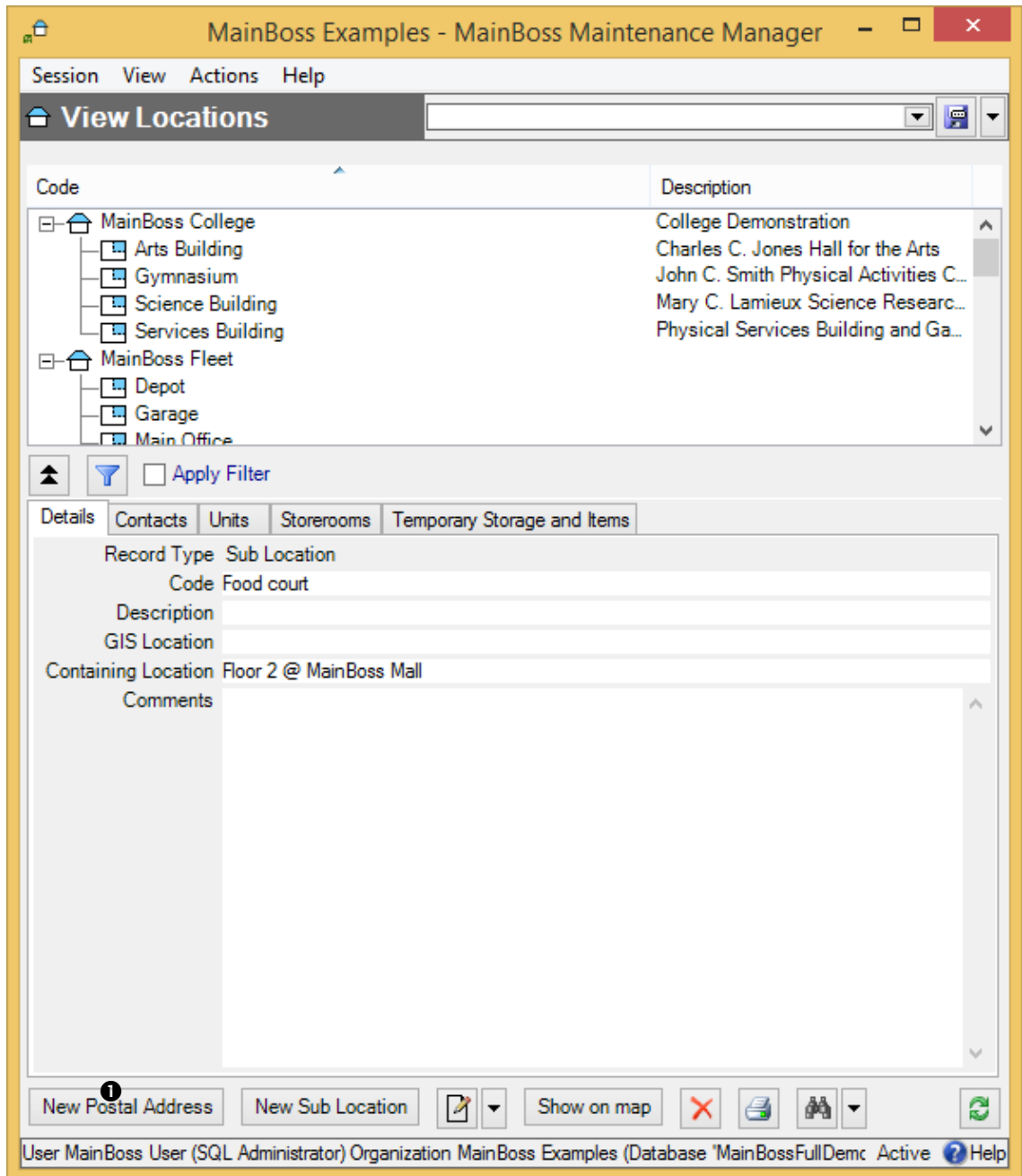
2. In the **View** section of the table viewer, click **New Postal Address** ❶. This opens a window where you can enter information about the location.

3. In "**Code**" ❷, enter a short identification code for this location.

4. In "**Description**" ❸, enter a longer description of the location.

5. In the other fields of the record, enter address information for the location.

6. Click **Save & Close** ❹ to save the information and close the window.

If you're typing in a lot of postal addresses at once, you can use **Save & New** to save the information you've just typed, then set up a new empty record so you can record another address.

A postal address record contains a "**Comments**" field. You can use this field to record any information that isn't covered by the other fields. For example, you might provide driving directions from your headquarters to the address if your personnel would find such information useful.

The **Contacts** area of the postal address record lists people you know at the given location. These are people from your **Contacts** list (discussed later in this chapter). When you first start setting up MainBoss, this area will be blank. It's filled in automatically when you begin recording contact information.

As a special feature, MainBoss makes it easy for you to use Google Maps to specify "**GIS Location**". First, use Google Maps to find the location's geographic position. Then, right-click on the location in the Google Maps display and click **Center map here** in the resulting menu. Next, click the button in the left-hand panel with the link icon on it (a section of chain). Finally, copy the given URL (beginning with http:) and
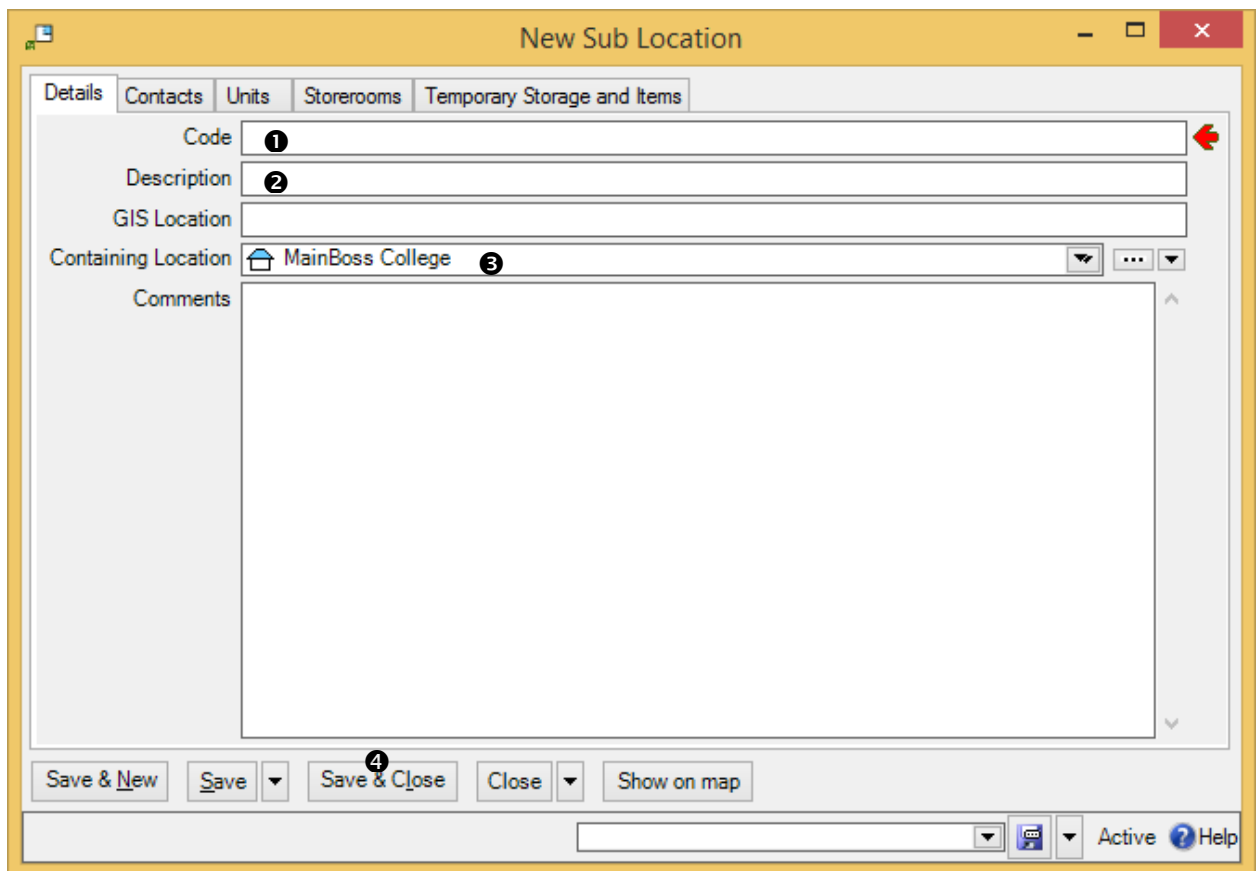
paste it into the "**GIS Location**" field in MainBoss. MainBoss will extract the longitude and latitude from the Google Maps URL.

Sub Locations

A sub location is contained in another location. Sub locations can contain their own sub locations.

☐ **To create a new sub location record:**

1. In the **View** section of the **Locations** table viewer, click **New Sub Location**. This opens a window where you can enter information about the sub location:

2. In "**Code**" ❶, enter a short identification code for this location.

3. In "**Description**" ❷, enter a longer description of the location.

4. In "**Containing Location**" ❸, specify the location that contains this sub location.

5. Click **Save & Close** ❹ to save the information and close the window.

Note that "**Containing Location**" is a drop-down list. You usually fill in the value by clicking the first arrow, which displays your current locations. However, you can also click the "..." (which displays your full **Locations** table) or by clicking the last arrow (which offers you a menu of options for filling in the field).

Few Locations, Many Units: As discussed later in this manual, a unit is a place or piece of equipment that you maintain. In general, we recommend that you only have a few locations and sub locations; the contents of those locations should all be units.

One reason for this is that work orders are written against units, not locations. For example, if you want to write up a work order for renovations on Room 101, that room has to be a unit, not a location. Similarly, if you want to write a work order for filling a pothole in the north parking lot, that parking lot has to be a unit, not a location. In general then, locations and sub locations are intended for large-scale entities; smaller scale entities should be units.
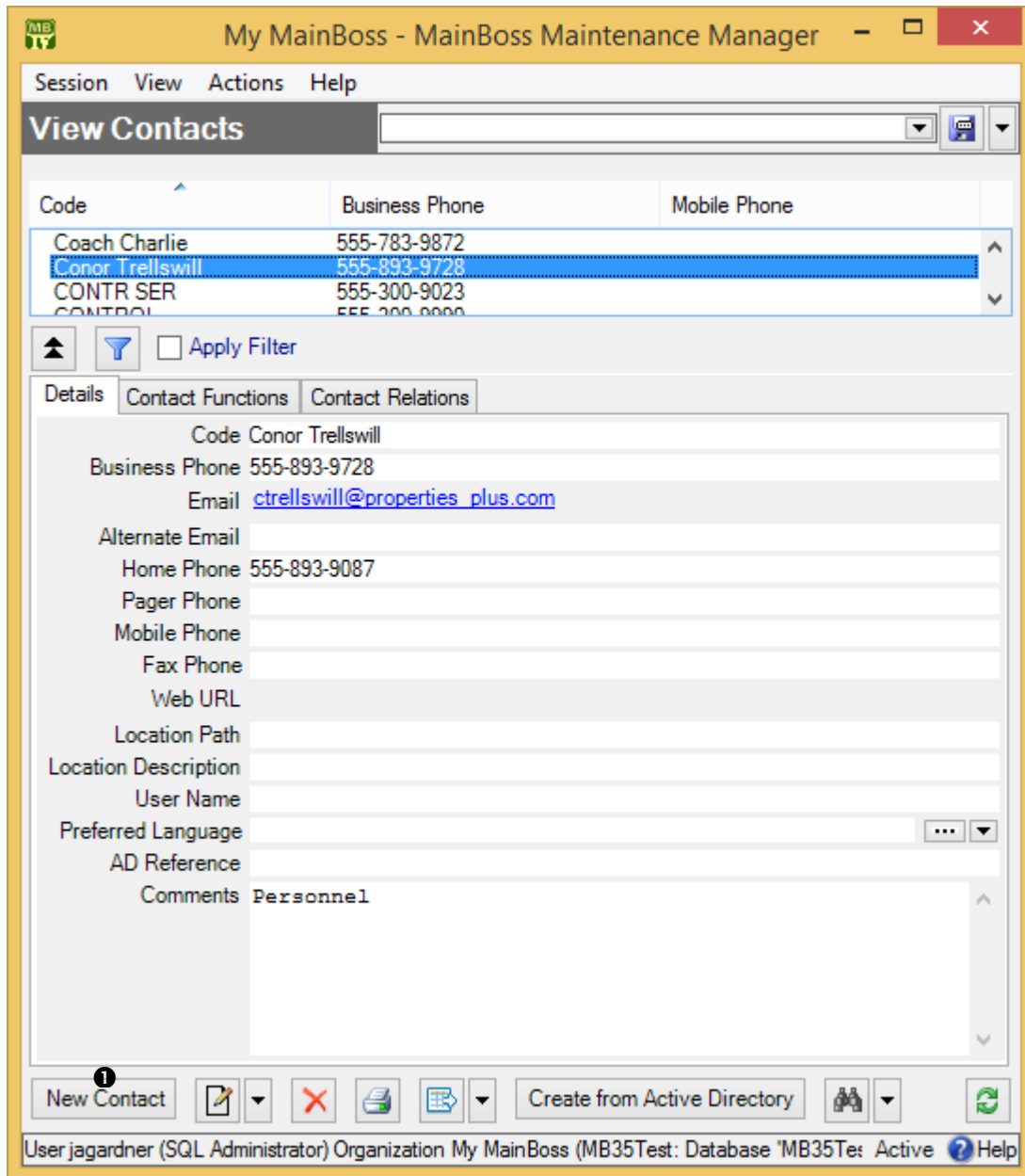
Contacts

The **Contacts** table contains contact information for every person who's relevant to your maintenance work. This includes maintenance workers, other personnel in the company, outside contractors, suppliers, clients, customers, and anyone else whom you may wish to contact.

When you record information about a contact, you may specify a location for that person. You do this by specifying a record in your **Locations** table. This is often a postal address: the contact person's mailing address. However, for people inside your company, you might specify a sub location within one of your buildings. For example, in a shopping mall, you might have a contact person associated with each store; the location associated with each such contact is the store itself.

> You can create an appropriate location record before you start creating a contact record. Alternatively, you can create a location record as needed in the middle of creating the contact record.

To create contact records:

1. In the control panel, go to **Contacts**. This opens the **Contacts** table viewer.

2. In the **View** section of the **Contacts** table viewer, click **New Contact** ❶. This opens a window where you can enter contact information:

3. In "**Code**" ❷, enter the person's name.

4. In "**Location**" ❸, specify the person's location by selecting an entry in the **Locations** table. (If you don't yet have a **Locations** record for this person, click the "..." button for "**Location**". This opens a window where you can create a new location record.)

5. Fill in any of the other fields for which you have appropriate information.

6. Click **Save & Close** ❹ to save your data and close the window.

For people who are actually users on your computer network, Windows maintains information in what is called *Active Directory*. MainBoss can copy information from Active Directory into a contact record by using **Initialize all from Active Directory** (for new contact records) and **Update all from Active Directory** (for contact records that were previously initialized from Active Directory). For more information, see the online help.

Vendors

MainBoss uses the term *vendors* to refer to anyone from whom you purchase goods or services. This includes your suppliers and any outside contractors whom you hire for special jobs.
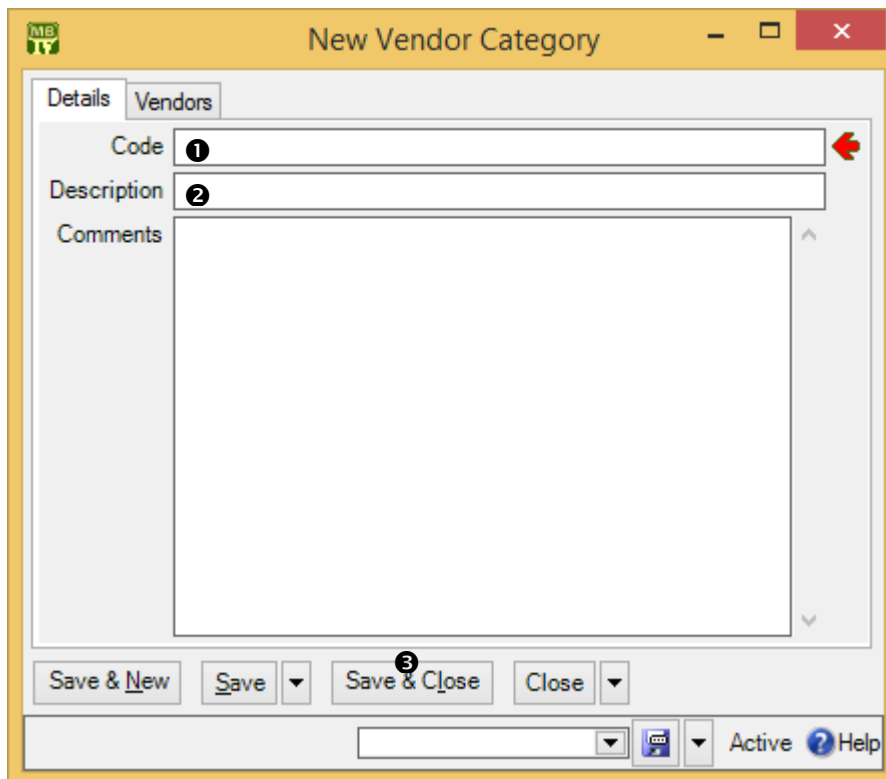
Vendor Categories

Before creating vendor records, you'll find it useful to define a few vendor categories. You can use these categories to classify your vendors and organize your reports. Here are some typical vendor categories:

```
Spare parts
Service contractor
Tools
Electrical
Plumbing
Major equipment
```

If you set up and use vendor categories, you can obtain reports that answer questions like, "Where do we spend the most on plumbing suppliers?" or "How much did we spend last year on service contractors?"

☐ **To define a vendor category:**

1. Go to **Coding Definitions** | **Vendors** | **Vendor Categories**.

2. Click **New Vendor Category**. MainBoss opens a window where you can define a new category:

3. In "**Code**" ❶ type a brief identification code for the category.

4. In "**Description**" ❷ type a longer description of the category.

5. Click **Save & Close** ❸ to save your data and close the window.

_____

Vendor Records

To record information about vendors, use **Coding Definitions** | **Vendors**. Clicking **New Vendor** opens a window where you can enter information about a vendor:

The "**Vendor Category**" field is where you specify a vendor category, if you've defined such categories.

As shown above, you can record several contact people for a single vendor. "**Sales Contact**" is the person you talk to about buying goods, "**Service Contact**" is the person you talk to about service, and "**Accounts Payable Contact**" is the person you talk to about billing. Each of these people should have an entry in your **Contacts** table.

When you fill in one of the contact fields, the associated phone and email fields are automatically filled in with their entries in the **Contacts** table. For example, when you specify someone as your "**Sales Contact**", the associated "**Business Phone**" and

"**Email**" columns are filled in with that person's phone number and email address, as recorded in your **Contacts** table.

If you have licensed the **Accounting** module, you will see a line for the "**Accounts Payable Cost Center**". This is the cost center to be associated with payments to this vendor. (For more on cost centers and other accounting topics, see *Cost Centers* on page 56.)

"**Account Number**" lets you record your account number (if any) with this vendor.

"**Payment Terms**" lets you specify default payment terms for anything you buy from this vendor (e.g. `Net 30` or `Cash On Demand`). When you actually create a purchase order addressed to this vendor, you can choose different terms if appropriate.

All of the above fields are in the **Details** section of the vendor record. Other sections will be filled in automatically as you continue with other parts of MainBoss set-up. For example, the **Units** section will automatically get filled in as you record information about your units: specifically, which units you bought from which vendor.

Cost Centers

This chapter is only relevant if you have licensed the **Accounting** module. If you do not have a license key for this module, the control panel will not contain entries related to cost centers.
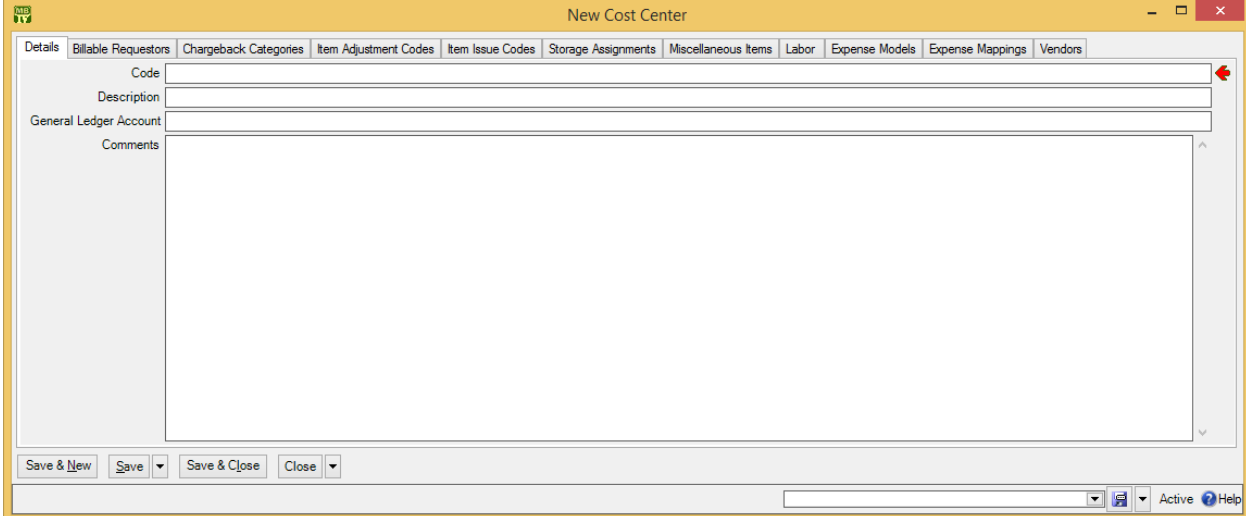
Cost centers correspond to general ledger account numbers or other basic identifiers that you use for accounting. For example, if your organization has a separate account for each department, you might set up a cost center for each department, for each supplier, for each customer to whom you supply maintenance services, and so on.

For an overview of how accounting works in MainBoss, see *Accounting Facilities* on page 37.

When you create a MainBoss database, a number of cost centers are automatically created in it. These *default* cost centers are used in situations where you haven't set up cost centers of your own. Although you can use these default cost centers, you probably want to set up cost centers of your own.

Setting Up Cost Centers

To set up your cost centers, go to **Coding Definitions** | **Cost Centers** and click **New Cost Center**. This opens a window where you can create a new cost center definition:

The only section you need to set up is the **Details** section. Fill in "**Code**" and "**Description**" with an identification code and a longer description, then fill in "**General Ledger Account**" with the account number. Repeat this for all the accounts that are relevant to your maintenance work.

MainBoss lets you associate several different cost centers with the same general ledger account. This makes it possible for you to break down your costs more than is allowed by your general ledger system. For an example, see *Accounting Facilities* on page 37.

Cost centers are used in various types of records, as will be seen in later chapters of this manual.

Changing Cost Centers: From time to time, some category of expense may be assigned to a new general ledger account. For example, the account number for labor costs may change from `1234` to `12345`. You may be tempted simply to change the account number associated with the cost center. This might mean changing the `Labor` cost center record so that its "**General Ledger Account**" field shows the new number.

However, you usually don't want to do this. MainBoss may have many existing records referring to the `Labor` cost center. If you change the associated ledger account number, it effectively changes the ledger account associated with all the old records. In other words, all the old records will state that their expenses were charged to the new `12345` when in fact, they were charged to the old `1234`.

If you want to change the account number associated with a cost center, you have to use a two-stage process.

1. Change the "**Code**" in your existing cost center record from `Labor` to something else (e.g. `Labor before Sept 1 2008`). Change the "**Description**" in a similar way. All existing records that use this cost center will continue to refer to this record. In other words, all your old records will continue to refer to the original account number: `1234`.

2. Create a new cost center record. This one can have the "**Code**" `Labor` and a corresponding description. Its "**General Ledger Account**" should be the new `12345`. When you use the `Labor` cost number in future records, they will all have the new ledger account `12345` rather than the old `1234`.

Note that you can change the ledger number in the existing record if you're just correcting a mistake. For example, suppose the correct number has always been `12345` but you originally typed `1234`. You can fix the situation simply by changing the number in the cost center. Then all records that refer to that cost center will automatically have the number changed too.

Cost Centers and Inventory

In MainBoss, every storeroom assignment has an associated cost center. This cost center is used whenever a quantity of the associated item is added or subtracted from the storeroom:

☐ When an item is received from a vendor and placed into a storeroom, two accounting records are created: one for accounts payable to the vendor, and one

increasing the value of the storeroom assignment by the same amount. (Note that the accounting records are written when an item is received, not when it is ordered.)

☐ When an item is issued to a work order, two accounting records are created: one that decreases the value of the storeroom assignment by the cost of the item, and one that charges the cost of the item to the work order (in accordance with the work order's expense model).

The same principle applies when you issue an item for some other purpose. In this case, the issue code specifies the cost center that gets charged. (See **Coding Definitions** | **Items** | **Item Issue Codes**.)

☐ When an item is transferred from one storeroom to another, two accounting records are created: one that decreases the value of the storeroom assignment in the first storeroom, and one that increases the value of the storeroom assignment in the second storeroom.

Units

A *unit* is anything on which you perform maintenance. Typically, units are either places or pieces of equipment. Here are some examples of units in different types of maintenance operations:

☐ In an office building, each office suite might be a separate unit. Common areas (e.g. elevators and washrooms) would also be units, as would equipment like HVAC systems and hot water heaters.

The same applies to other maintenance departments that deal with facilities (shopping malls, schools, health care facilities, government buildings, and so on). Individual rooms or suites would be units; other units would include the standard shared equipment (heaters, boilers, chillers, etc.) and any special equipment that the facility might have (e.g. lab equipment in a hospital).

☐ In a factory, each piece of manufacturing equipment might be considered a separate unit. Other units might include vehicles like fork-lift trucks or delivery vans, as well as the sort of units mentioned in connection with office buildings (washrooms, HVAC systems, hot water heaters, etc.).

How Units are Used

Units are used in work orders and requests to specify where work should be done. For example, a work order might call for workers to repair a piece of equipment or change a light bulb in a particular room. Units therefore indicate places or things you work on.

Units records contain information that may help maintenance management and workers on the job. This may include any of the following:

☐ Equipment specifications

☐ Meter readings

☐ Lists of spare parts that might be needed to service the unit

☐ Information about service contracts and warranties related to the unit

☐ Expected lifespan of the unit, as well as estimates of replacement costs

☐ References to files that contain further information about the unit (e.g. schematic diagrams or operating manuals)

☐ Lists of work requests and work orders on the unit

☐ Planned maintenance related to the unit

□ Any other information you may find useful—this could be something as simple as what type of light bulbs are used in a particular room, or detailed facts about a complicated machine

One of the most important uses of MainBoss is to serve as *institutional memory*—a repository of information that you want to make available to present and future personnel. By filling out as many fields as possible in a unit record (including any useful "**Comments**"), workers can communicate information to each other. As a simple example, suppose a worker takes five minutes to locate a fuse box because somebody has moved a bookshelf in front of it. If the worker adds a comment to the unit record ("The fuse box is behind the bookshelf"), it may save other workers from wasting the same amount of time.

Locations and Units

As noted previously, places can be units. This raises a question: which places should you put in the **Units** table and which should you put in the **Locations** table?

We recommend that location records should only be used for buildings and for large sections of buildings (e.g. for wings). Everything else should be a unit. In fact, if you have a large number of buildings in the same location, as on a university campus, you might use a single location record for the entire campus. Each building could be a unit of that single location, with all the rooms in the buildings as sub-units.

MainBoss associates work with units: when you write up a work request or work order, you specify the unit that needs attention. Therefore, when you declare that a place or thing is a unit, you're saying, "I want to track the work that we do on this, and keep it separate from work elsewhere."

@ Format: MainBoss sometimes uses the notation Unit @ Location to specify a particular unit at a particular location.

Unique Names: MainBoss lets you give the same name to different units in different locations. For example, you could have units called AIR CONDITIONER in many different locations. In practice, however, this can be confusing, especially if you move an air-conditioner from one room to another. By giving each air-conditioner a unique name, you make it easier to track each unit's maintenance history, even if it gets moved.

Sub-Units

Units can contain sub-units. For example, consider a shopping mall where every store is a separate unit. You could divide large stores into smaller sub-units. A multi-floor department store might have each floor considered a separate sub-unit, and the floors

might be divided into smaller subsections (so that each major area within the store is its own separate unit).

How Finely Do You Divide Your Units? Consider a washroom. Do you want to consider every plumbing fixture as a separate unit and to track their repairs individually? Or is it enough to consider the washroom as a unit and to lump together all repairs to all fixtures as repairs to the washroom overall?

It's a trade-off: how much detail do you want, and how much is too much? If your units are too big and general, you lose track of specifics: you might see that you're getting a lot of complaints about the same washroom, but not notice that the problem is always the same fixture. On the other hand, if your units are too small and specific, your record-keeping gets more complicated because you have a lot more to keep track of.

No single approach will work for all maintenance departments. Define your units to get a level of detail you can live with.

> If you aren't sure whether or not to subdivide a unit, *don't* subdivide it—it's easy to subdivide the unit later. On the other hand, it's hard to merge small sub-units into a larger whole—you'd have to edit each work order on each sub-unit to refer to the whole unit, which is a lot of work (and can require substantial accounting corrections).

Sub-Units and Spare Parts: Just as there's no perfect dividing line between units and locations, there's no perfect dividing line between sub-units and spare parts.

For example, consider a window air-conditioner. If the room that contains the air-conditioner is a unit, should you consider the air-conditioner a sub-unit or is it simply a spare part for the room? The answer depends on how you intend to maintain the air-conditioner. If you intend to repair the air-conditioner when it malfunctions, it should probably be considered a sub-unit. That way, you can use MainBoss to track all the maintenance you do on the unit. If you intend to replace the air-conditioner if anything goes wrong, then it's just a spare part: it's not something you maintain, it's something you throw out when it breaks.

A similar principle applies to other units. For example, if a complex piece of manufacturing equipment has several components, each of which may be maintained individually, these should probably be designated as sub-units. This is particularly important for components that might be swapped from machine to machine; you ought to track the repair histories of such components, because that information might be vital in identifying the source of a problem. On the other hand, if a component isn't subject to repairs—if you just replace it when it malfunctions—then it's usually a spare part, not a sub-unit.

Unit-Related Codes

Before setting up your unit records, there are a number of useful codes you should consider defining. You can create unit records without using these codes, but the codes are easy to set up and can provide you with useful information.

The purpose of all these codes is to help you create reports that break down your information in various ways. As an example, consider a maintenance department that maintains a commercial mall. Maintenance management might want to break down data in various ways:

☐ By type of work (how much is spent on repairs, how much on renovations, how much on preventive maintenance)

☐ By type of unit (how much is spent on plumbing equipment, how much on electrical, how much on mechanical)

☐ By type of location (how much is spent on stores, how much is spent on public spaces like washrooms, how much is spent on the parking lot)

☐ By tenant (how much is spent on each store)

☐ By integrated system (how much is spent on HVAC, how much is spent on security systems, how much is spent on fire prevention)

☐ By accounting classifications; this depends on how the parent organization sets up its books, but clearly the maintenance department needs a breakdown of its expenses in a format that matches the parent organization's accounting procedures

Other types of breakdowns may be desirable. For example, the parent organization's accounting system may not be very useful for the purposes of the maintenance department; you may like to set up an additional way of dividing expenses into accounts.

To make it possible to break down costs in various ways, MainBoss lets you define various codes and categories related to units. There are no hard-and-fast rules for how you use these. For example, every unit can have a *unit category* and a *usage code* (as well as other codes). You might use unit categories for one type of breakdown (plumbing, mechanical, electrical) and usage codes for a different kind (unmovable equipment, movable equipment, parts of the premises). You don't have to use all the available codes/categories, and you don't necessarily have to use them for every unit, but they can be very helpful in looking at your operations from various angles.

Unit Categories

Before you start recording information about your units, you should consider defining *unit categories*. As mentioned in the previous section, categories help organize your information, especially when printing out reports. If you choose your unit categories in

advance, you can use them when you're creating your unit records. The alternative is either to skip using categories, or to go back later on and edit all your unit records to add category information.

Different types of organizations will have different unit categories, but here are a few possibilities:

```
Office space
Elevators
Public areas
Washrooms
Outdoors
Air conditioners
Boilers
Manufacturing equipment
Safety equipment
Security Systems
Pumps
Vehicles
```

☐ **To define unit categories:**

1. Go to **Coding Definitions** | **Units** | **Unit Categories**.

2. Click **New Unit Category**. MainBoss opens a window where you can define a new category:
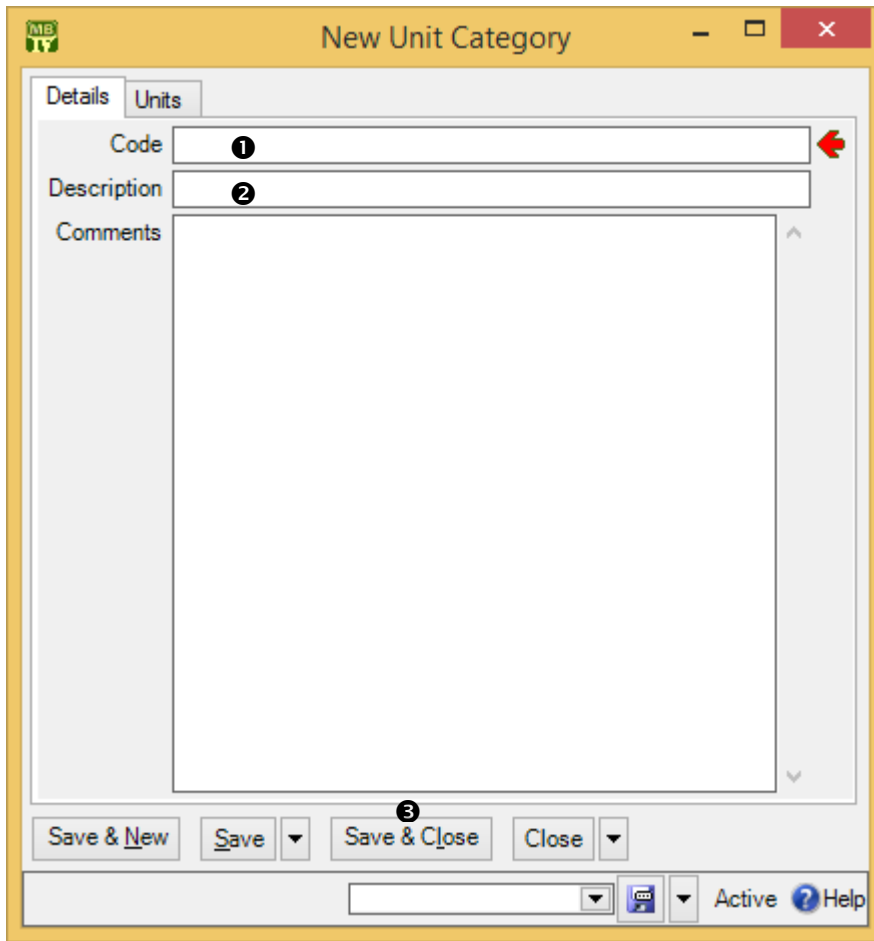
3. In "**Code**" ❶ type a brief identification code for the category.

4. In "**Description**" ❷ type a longer description of the category.

5. Click **Save & Close** ❸ to save your data and close the window.

Systems

MainBoss lets you define *systems* of related units. For example, the various pieces of equipment for heating and air-conditioning might all be considered to belong to a single HVAC system. Here are some other examples:

```
Electrical
Plumbing
Security
```

It's useful to specify that certain units make up an integrated system because you can then ask MainBoss for reports about the system: "Show me all the work orders for our HVAC system." "How much did we spend maintaining HVAC last year?"

A system is just a label that you can attach to unit records: "By the way, this is part of our HVAC system." Such a label makes it possible for MainBoss to give you overviews of the entire system, not just its individual components.

As is obvious, systems have a lot in common with unit categories. The usual difference is that categories are based on a unit's *type* while systems are based on a unit's *purpose*. There's no need to assign every unit to a system; just set up the system classifications that you'll find helpful.

Systems are most useful for organizing units that are spread across many locations. If you have a number of units that are all in the same place, it's better to classify them as sub-units of a larger unit. For example, you might consider an assembly line as a single unit; the machines that make up the line would all be sub-units of the larger unit.

To set up systems, use **Coding Definitions** | **Units** | **Systems**. The process is exactly like creating unit categories.

Other Unit-Related Categories and Codes

In addition to unit categories and systems, **Coding Definitions** | **Units** lets you define the following:

Asset Codes
These are similar to cost centers, in that they're typically related to your accounting system. You might use asset codes to break down cost centers into finer detail; for example, if you have very broad unit cost centers (e.g. just `Equipment` and `Premises`), you might use asset codes to subdivide these into smaller categories you might want to track. On the other hand, if you have very detailed unit cost centers, you might create more general asset codes so that you can obtain high-level overviews.

Ownerships
These indicate who "owns" each unit. This might mean literal ownership, if your maintenance department manages units owned by other parties. Alternatively, it might mean units which are temporarily associated with other parties (e.g. equipment currently in the possession of tenants).

Unit Usages
These are general purpose codes that can be set up for classifying units. (If you migrate from MainBoss Basic to MainBoss, the export-import process uses unit usage codes to distinguish between MainBoss Basic spaces and equipment. Otherwise, usage codes are available for any purpose you wish.)

Access Codes

Access codes can inform workers if there are any restrictions or requirements for accessing units. For example, you might have codes indicating that a unit can only be repaired on weekends or with written permission from someone in authority. Here's a sample list of access codes:
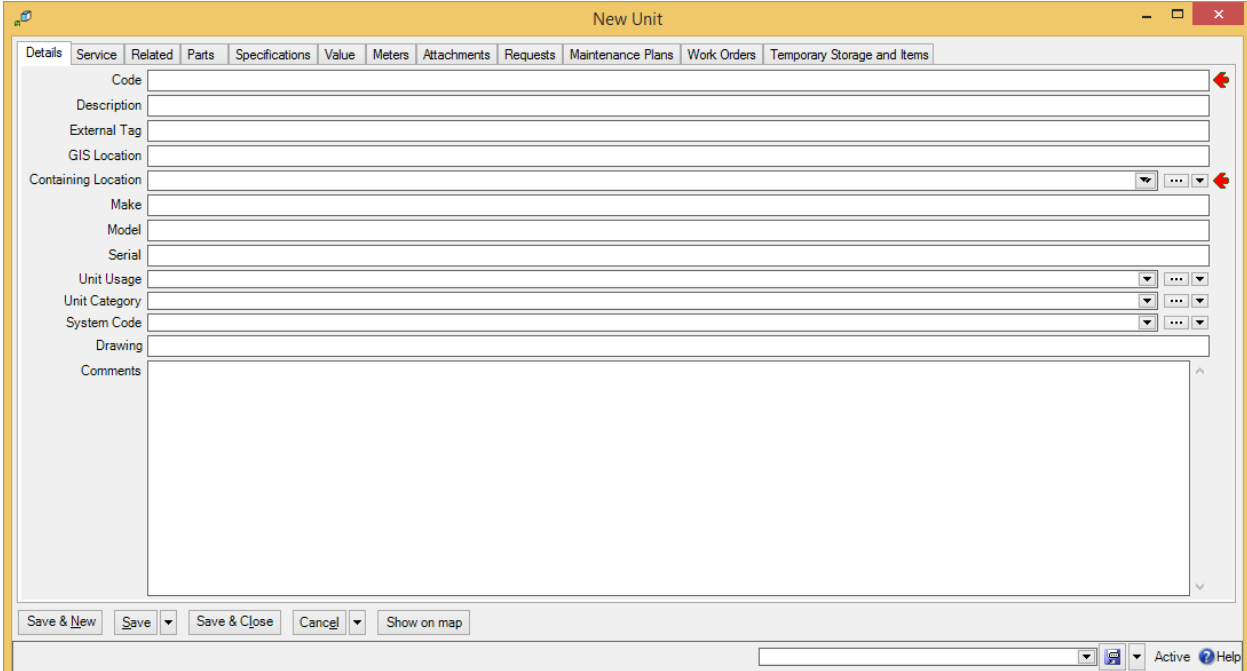
```
Anytime
Only 9-5
Only on night shift
Only when store is closed
Only when office is open
Only when manager is present
Get permission first
```

To set up access codes, use **Coding Definitions** | **Access Codes**. The process is exactly like creating unit categories.

Access codes may be used on work orders and work requests (indicating when the work should or shouldn't be done) and on unit records (to state overall when the unit may or may not be serviced).

Creating Unit Records

To create a new unit record, go to **Units** in the control panel. Click **New Unit** in the table viewer. This opens a window where you can record information about the unit.

This record has sections titled **Details**, **Service**, **Related**, and so on. Each section records a different type of information.

When you start setting up MainBoss, you'll only be able to fill in the **Details** section. This is because you have to set up other MainBoss tables before you can fill in other sections of the unit record. For example, to set up the **Parts** section of a unit record, you must enter information about your inventory of spare parts. Typically, you'll create your basic unit records long before you get around to recording inventory.

> You can give two different units the same **Code** name, provided they're in different locations. However, this is a bad idea. Not only will it be hard to distinguish one unit from another, but many types of units can be moved from one location to another, perhaps leaving two units with the same name in the same location.

Smart Coding: Unit reports typically show units in alphabetic order. Therefore it's useful to choose codes so that similar units will be sorted beside each other. One way to do this is to adopt a standard like

```
"What the unit is" "Size" "Brand"
```

For example, you might have a code like `AHU450Trane` for an air handling unit, 450 ton, from Trane. This coding ensures that all air handling units are sorted together (they start with `AHU`) and sizes are sorted together (all `AHU450` will be listed together). The brand is typically the least important piece of information, so it's left until last.

Starting Your Units Table

Entering complete information on all your units can be a huge undertaking. Fortunately, it's not necessary—you can start with a basic set-up and add more as time goes on. The more information you put in, the more results you can get out; however, even just the basics can provide you with noticeable benefits.

Here's one approach to phasing in unit information so that you don't have to do everything all at once:

1. Begin by making unit records for places. It should be straightforward to create a list of rooms, office suites, public areas, etc. that you maintain, and this will be enough to get you started on work orders, work requests and planned maintenance. Even if a job deals with repairing a piece of equipment, you can write up a reasonable work order by specifying the room where the equipment is located. This lets you get started right away, using MainBoss to manage work requests and work orders.

2. Create bare-bones unit records for equipment, starting with the equipment you service most. For example, when you have to repair a piece of equipment, create a record for that equipment. The record need not contain complete details about the equipment (although the more information you record, the more you can use later on). As soon as you create a unit record, you can begin tracking work on that unit.

3. Continue to create new unit records for equipment until your **Units** table lists *all* your equipment and all rooms, suites, etc. At the same time, add more details to your existing records. For example, begin to record the spare parts used by your most important pieces of equipment—if you make it easy for workers to determine what parts they may need *before* they go to the work site, you can save them wasted effort.

4. Record new information as it comes in. For example, if your company buys a new piece of equipment, you can record *full* details about that equipment right away. If you renew a service contract, make sure you record which units it covers and all relevant terms of the contract.

Note that a warranty is just a type of service contract with the manufacturer. Therefore, you can use the **Service Contracts** table to record warranties whenever you buy new equipment.

Stay on top of new information, gradually fill in the gaps on old units, and eventually your database will contain everything you need to do your work effectively. It may take you a while, but remember: the more you put in, the more you can get out.

Unit Details

The **Details** section holds basic information:

Code and **Description**: An identification code for the unit and a longer description.

External Tag: A short string of characters which can be turned into a bar code associated with this unit. This should be a unique identifier created by your organization. For more information, see *Bar Code Support* on page 42.

GIS Location: The geographic position of this location. Coordinates are specified in longitude and latitude.

As a special feature, MainBoss makes it easy for you to use Google Maps to specify "**GIS Location**". First, use Google Maps to find the location's geographic position. Then, right-click on the location in the Google Maps display and click **Center map here** in the resulting menu. Next, click the button in the left-hand panel with the link icon on it (a section of chain). Finally, copy the given URL (beginning with `http:`) and paste it into the "**GIS Location**" field in MainBoss. MainBoss will extract the longitude and latitude from the Google Maps URL.

Containing Location: Where to find the unit. The drop-down list will display a location map giving existing locations and units. If you choose a location, you're saying that you're describing a new unit in that location. If you choose a unit, you're saying that you're describing a new sub-unit of that unit.

Make, **Model**, **Serial Number**: These fields are mostly intended for pieces of equipment. They probably won't be relevant if the unit is a place.

> **Mobile Units and Serial Numbers:** If a unit can possibly be moved, it's useful to assign it a serial number, even if there isn't a serial number from the manufacturer. This makes it easier to keep track of which unit is which, especially when two units are moved into the same location.

Unit Usage: Essentially, an anything-you-want code that you can use to record more information about your units.

Unit Category: A category for the unit. For more about unit categories, see *Unit Categories* on page 62.

System Code: The system to which the unit belongs (if any).

Drawing: If you have hard-copy drawings of a piece of equipment, fill in this field with information that will help your workers find the drawing (e.g. an index number that indicates a drawer in a particular filing cabinet).

Comments: Any comments you wish to record about the unit. In particular, this is a great place for workers to record information that might be useful in future. Possibilities might include listing:

- Special tools needed for working on the unit
- Advice based on experience ("Always check for water damage—the unit is close to a door that stays open all day")
- Useful tips ("The fuse box is hidden behind the gray filing cabinet")
- Basic questions to ask someone reporting a problem on the unit, such as, "Is the machine plugged in? Does the electrical outlet actually have power? Have you tried turning it off and turning it on again?"

Unit Values

The **Value** section of a unit record records information about the unit's purchase value and replacement value. Typically, this section is only used for pieces of equipment; however, it might also be relevant to other types of units (e.g. roofs that will need replacing, parking lots that will need repaving, and so on). Fill in the fields with the following information:

Purchase Date: The date the unit was purchased.

Purchase Vendor: The supplier who sold you this unit. To fill in this field, you need to create an entry in the **Vendors** table; you can either do that now (in which case, see *Vendors* on page 129 for details) or wait until later, when you create the **Vendors** table.

Original Cost: The original cost of the unit.

Ownership: A code that lets you indicate who owns the unit. This might be the actual owner (e.g. a customer whose property you maintain). It might also be a

person or department in your own company who is responsible for the unit (e.g. the person you have to see for permission to repair the unit).

Asset Code: A code used to reflect accounting information for the unit.

Future Value: Fields in this area deal with expected replacement costs for the unit. This information will be used if you ask MainBoss to prepare a Replacement Forecast report.

Replacement Cost Last Date: The date on which you fill in the other fields. This means the day when you make the estimates, not the day when you expect the unit to be replaced. Recording this date makes it easy to tell how up-to-date the estimates are. If you see that it's been a long time since someone estimated replacement costs, it might be time to check that the estimates are still reasonable.

Replacement Cost: The estimated cost *if you had to replace the unit today*. In other words, this is the cost *without* taking into account future price inflation. You specify inflation rates when you actually prepare a Replacement Forecast report, not when you make your original replacement estimates.

Typical Life: The expected lifespan for this type of unit (in years). This means the typical *total* lifespan, not the remaining lifespan for this particular unit.

Scrap Date: The date when you expect to take this unit out of operation.

Scrap Value: The price you expect to sell the unit for after it's taken out of operation. Again, this is the price in today's currency, without accounting for future inflation.

Relationships

Relationships let you specify connections between units and other units, or between units and people. For example, suppose a particular water heater is located in Washroom A but provides hot water for Washrooms A, B, C and D. You could create a unit record for the heater, specifying that it's a sub-unit of Washroom A. You could then establish relationships between the heater's unit record and the unit records for all of the washrooms that the heater services. Thus if there's a problem with hot water in Washroom C, you can quickly check the washroom's unit record to find out where its water heater is. By setting up unit-to-unit relationships, you can show connections between units that may not be obvious, especially when the units are in different places.

As an example of a unit-to-person relationship, suppose that a commercial mall rents out store space to tenants. In the unit record for a particular store, you might set up relationships that connect the store to the contact information for the store's manager. You might also set up relationships to indicate several people you might contact in case there's some emergency involving the store. As another example, if you're maintaining a company's fleet of vehicles, the unit record for each car might have a unit-to-person relationship specifying the person who usually drives the car.

The first step in setting up a relationship is to create a *relationship* record. This puts into words what the relationship actually is. For example, suppose you're setting up a unit-to-person relationship between cars and drivers. In this case, the unit is the car and the contact person is the driver. Therefore, you might say that the unit `Is driven by` the person, and the person `Drives` the car.

To set up this record, go to **Coding Definitions** | **Relationships** and click **New Unit Contact Relationship**. In the resulting window, you can set up the relationship as shown below:

For a car driven by a person, you would specify `Drives` in "**B As Related To A Phrase**" and `Is driven by` in "**A As Related to B Phrase**". Once you have created the relationship record, you can have units and contacts make use of this relationship. The same applies for unit-to-unit relationships.

To specify a relationship between a unit and another unit or the unit and a person, you use the unit record's **Related** section.
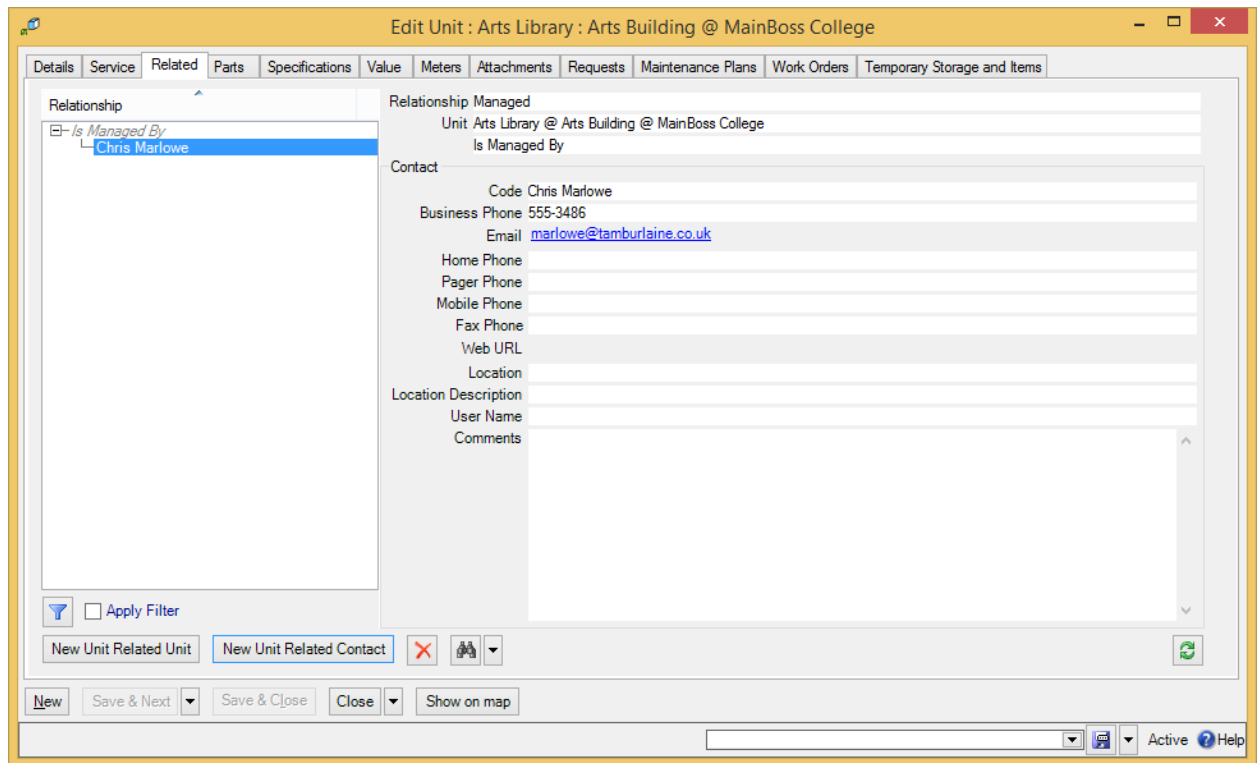
In the example depicted above, this unit record describes an area of a building that is managed by a particular person. This gives you someone to contact if you want to perform maintenance in the area. By creating relationships for a unit, you can provide workers with useful information about units and the people connected with them.

Other Sections of Unit Records

The other sections of unit record can contain an intimidating amount of data, especially if you've never used a CMMS package before. You may ask, "How much should I really care about?"

The answer depends on the nature of your operation. A small maintenance organization may only need to record a small amount of data; a large maintenance organization with obligations to third parties (e.g. a management company that maintains property for tenants) may be legally required to keep a great deal more information on file. Different organizations have different needs. Nevertheless, the points below should help you evaluate what information you do or don't want to record:

Service section: Records information about service contracts covering the unit. This includes the start and end dates of the contract, and what the contract covers (e.g. parts, labor or both).

You can also use this section to record warranty information—just write up the warranty as if it's a service contract from the manufacturer.

Parts: Records information about the spare parts used in maintaining the unit. By listing these parts in the unit record, you can inform workers about what they may need to take when they go out on a job. The list doesn't have to be exhaustive; for example, if a unit record describes a room, you might just list what type of light bulbs are used in the room. This type of simple information lets workers show up with the correct bulbs rather than wasting their time making two trips because they guessed wrong.

Another important benefit of recording spare parts is the ability to transfer parts in case of emergency. If a critical piece of equipment needs a part and you don't have that part in stock, you can check MainBoss records to see if the part is used in any other unit. If so, you can remove the part from a less important unit to use it in the more important one.

Specifications: Useful once you're comfortable with the basics of MainBoss, but not necessary in your initial set-up.

Specifications let you create a fill-in-the-blanks form about a particular type of unit. For example, suppose that a maintenance department maintains a fleet of delivery vans. You could create a form that contains blanks for Vehicle Identification Number, engine horsepower, tire size, gas tank capacity, maximum load size, and any other information you might find useful when performing vehicle maintenance. You then fill in this form for each of the vans you maintain. That way, when workers are assigned to work on a van, they can look up the van's specifications and make sure they have the right tools, parts, manuals, etc. to do the job.

As another example, a vacation resort might create a form for all the resort's guest cabins. The form might have blanks for number of beds, type of bulbs used in the cabin's lights, heating and air-conditioning systems, and so on. Again, the point is to record the sort of information that might be useful to workers before they start a job.

Deciding what information to record may take time; so does gathering that information for all of your units. Nevertheless, consider the value of your workers always knowing what kind of tools and materials they need *before* they head out to the job site. Specifications can save you from a lot of wasted time and effort.

Note: Specification forms can contain any text you want. You can, for example, use specifications to give instructions for standard maintenance operations on a unit. (This can also be done with attachments, as discussed below.)

Meters: Important if you intend to use meter readings as a basis for scheduling planned maintenance. If you regularly record meter readings and tell MainBoss you want to follow a particular maintenance routine (e.g. an oil change every 3000 miles), MainBoss will automatically schedule maintenance jobs in accordance with the meter readings.

Even if you don't use meters to schedule planned maintenance, you may find it useful to use MainBoss as a repository for meter readings. You then have the meter readings available for examination, perhaps to get a clearer idea of equipment usage over a period of time.

For more on setting up meters, see *Meters* on page 74.

Attachments: Useful if you have online files related to a particular unit (e.g. manuals or schematic diagrams). Basically, an attachment record just gives the name of the file; this makes it easy for workers to refer to the file if desired.

Requests and **Work Orders**: Lists any work requests/work orders associated with the given unit. You don't have to do anything in these sections during set-up. Once you start using MainBoss, any work requests and work orders you create for a unit will automatically show up in the appropriate unit record section.

Maintenance Plans: Lists any planned maintenance associated with the unit. You'll fill in this section when you set up planned maintenance (as discussed in *Planned Maintenance* on page 104).

Temporary Storage and Items: Lists any work orders that use this unit as a temporary storage location for work materials.

For further information on any of the above, see the online help.

Meters

Units may have associated meters. For example, vehicles have odometers to measure miles or kilometers. Other types of equipment might have meters to measure hours of operation, power consumption, or other kinds of usage information.

MainBoss can record meter readings. These may be useful for general information, but they can also be used to schedule planned maintenance. For example, a vehicle may need an oil change every 3000 miles, or a piece of equipment may need servicing every 100 hours of use. If you record meter readings, MainBoss can get an overall picture of equipment usage and can predict when planned maintenance will be required. For example, if you record a car's mileage every week, MainBoss can calculate the car's average mileage over a period of time. MainBoss can then estimate the next time the car will hit the 3000-mile mark and need an oil change. In this way, MainBoss can automatically schedule when the next oil change should take place.

In order to record meter readings, you create meter records. These state the unit where the meter is located and what the meter measures. The rest of this chapter provides details on how to set up meter records.

Units of Measure

The first step in setting up meter records is to define units of measurement. For example, you might define:

```
Feet
Miles
Pounds
Quarts
```

or

```
Meters
Kilometers
Kilograms
Liters
```

Quite possibly, you'll have to define multiple units for the same thing. For example, you might have some meters that measure weight in pounds and some that measure weight in kilograms. Therefore, your **Units of Measure** table will have to include both `Pounds` and `Kilograms`.

MainBoss often abbreviates "unit of measurement" as UOM.

Units of measurement are used in connection with meters and also in connection with materials. For example, your inventory records might note that a particular lubricant is measured by the quart or by the liter.

☐ **To record your units of measurement:**

1. Go to **Coding Definitions** | **Units of Measure**.

2. Click **New Unit of Measure**. MainBoss opens a window where you can record a new unit of measure:
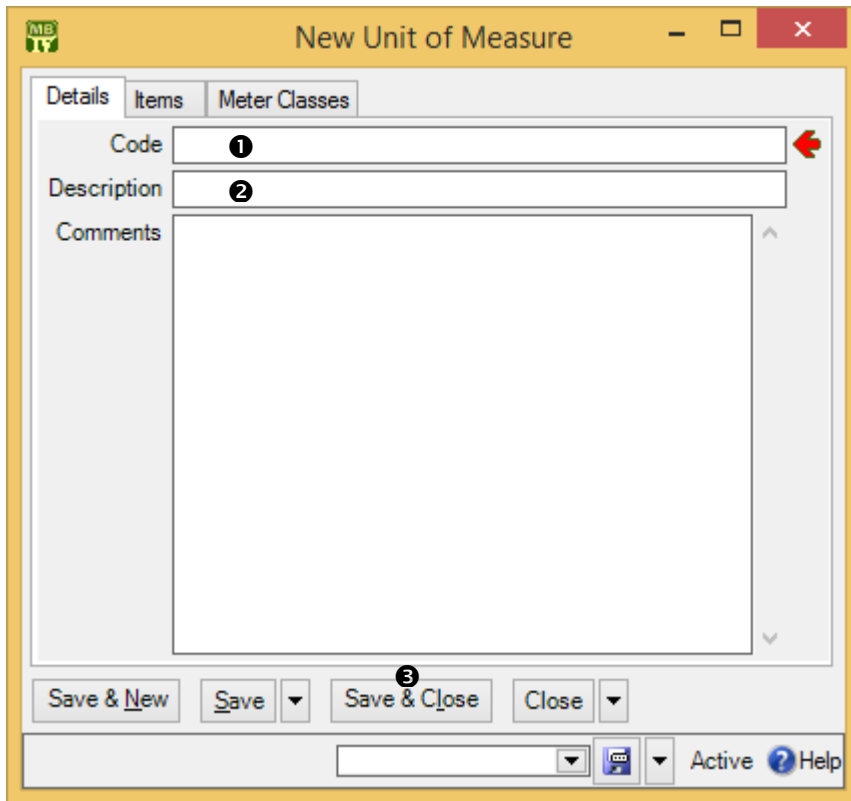
3. In "**Code**" ❶ type a brief identification code for the unit of measure.

4. In "**Description**" ❷ type a longer description of the UOM.

5. Click **Save & Close** ❸ to save your data and close the window.

Meter Classes

The next step in setting up meter records is to define *meter classes*. A meter class specifies a particular type of meter. For example, you might define classes like:

```
Odometer (miles)
Odometer (kilometers)
Kilowatt-hours
Hours of operation
```

Every meter class has an associated unit of measurement. For example, a type of meter that measures hours of operation would have "hours" as its UOM.

☐ **To create a meter class:**

1. Go to **Coding Definitions** | **Units** | **Meter Classes**.

2. Click **New Meter Class**. MainBoss opens a window where you can define a new meter class:

3. In "**Code**" ❶ type a brief identification code for the meter class.

4. In "**Description**" ❷ type a longer description of the meter class.

5. In "**UOM**" ❸, specify the units of measurement for this class.

6. Click **Save & Close** ❹ to save your data and close the window.

Creating Meter Records

Once you've defined your units of measurement and your meter classes, you can create meter records. To do so, you specify the class of the meter and the unit where the meter is located.

Each meter may also have an associated *offset*. An offset is a correction factor for situations where there's a difference between a meter's apparent reading and its true reading. For example, suppose that a meter only allows five digits, so it rolls over from 99999 to 00000. In this case, the offset will be 100000: for example, when the meter says 00023, the true reading should actually be 100023.

Setting the offset correctly is important if you're scheduling planned maintenance by meter reading. For example, if you're scheduling a vehicle's oil changes every 3000 miles and the odometer rolls over, you have to provide MainBoss with an appropriate offset; otherwise, MainBoss won't be able to correct for the difference between the apparent reading and the true reading, which means that MainBoss won't be able to schedule the required oil changes.

In practice, there are only a few times when you have to worry about a meter's offset:

☐ When a meter rolls over. In this case, the offset is what the meter would have read if it had enough digits.

☐ When you manually change the meter reading, either setting it backward or forward. In this case, the offset is the difference between the new reading and the old reading.

☐ When you install a new meter on a piece of equipment. In this case, the offset is what the old meter read just before you installed the new one. (If you installed the new meter because the old meter rolled over, set the new meter to what the old one would have read if it had enough digits.)

☐ **To create a meter record:**

1. Go to **Coding Definitions** | **Units** | **Meters**.

2. Click **New Meter**. MainBoss opens a window where you can define a new meter record:

3. In "**Meter Class**" ❶, specify the meter class to which this meter belongs. (MainBoss will fill in "**Meter Class UOM**" with the UOM associated with that meter class.)

4. In "**Unit**" ❷, specify the unit where the meter is located.

5. In "**Reading Offset**" ❸, enter an appropriate offset, if necessary.

6. Click **Save & Close** ❹ to save your data and close the window.

Important: MainBoss will not let you enter a meter reading with an "**Effective Date**" that is older than the most recent reading on the meter. You must enter meter readings in chronological order.

This is especially important to remember during configuration. If you want to enter old meter readings in order to create a meter history, you must enter the readings in the order they occurred. This means that you can't enter the most recent reading, then go back later to record older readings. If you want to enter an old reading but are prevented from doing so because there's a newer reading, you'll have to delete the new reading before entering the old one.

When you create a new meter, MainBoss automatically creates a meter reading record for that meter. By default, the effective date of the reading is the date on which you create the meter record; also by default, the value of the meter reading is zero. If this is not the case, go to the meter readings for the meter and delete the "zero" record. Then create a meter reading record that contains the actual value and date for the most recent reading on that meter.

Requests

This chapter is only relevant if you have licensed the **MainBoss Requests** module. If you do not have a license key for this module, the control panel will not contain entries related to requests.

The **MainBoss Requests** module lets you track problem reports using an easy-to-use format called a *request* (or *work request*).

Requests are designed to be as simple as possible; if necessary, they can be filled out by people with no maintenance experience. Of course, requests can also be filled out by maintenance personnel...but some companies allow non-maintenance people to submit requests (e.g. customers, clients, or people from other departments in the company). At other companies, requests are filled out by a maintenance department secretary who receives problem reports over the phone.

The **MainBoss Service** module can automatically process requests sent in by email. For more information, see *MainBoss Service* on page 86.

Requestors

A *requestor* is someone who submits a request. Since some organizations only allow requests from authorized requestors, MainBoss lets you create a list of people from whom you'll accept requests.

You create these lists through **Coding Definitions** | **Requests** | **Requestors**. You should create a **Contacts** record for the requestor first (giving the requestor's contact information) and then add the person to one or both requestor lists.

The **Contacts** record should contain the person's email address *in the format that is used when that person sends out email*. (In some organizations, a person might have multiple aliases that all go to the same mailbox. Mail sent from that box is always sent from a specific email address; that's the address which should go into the **Contacts** table.)

Remember that you can create a **Contacts** record directly from the window that creates a requestor. For example, here's the window for creating a requestor record:
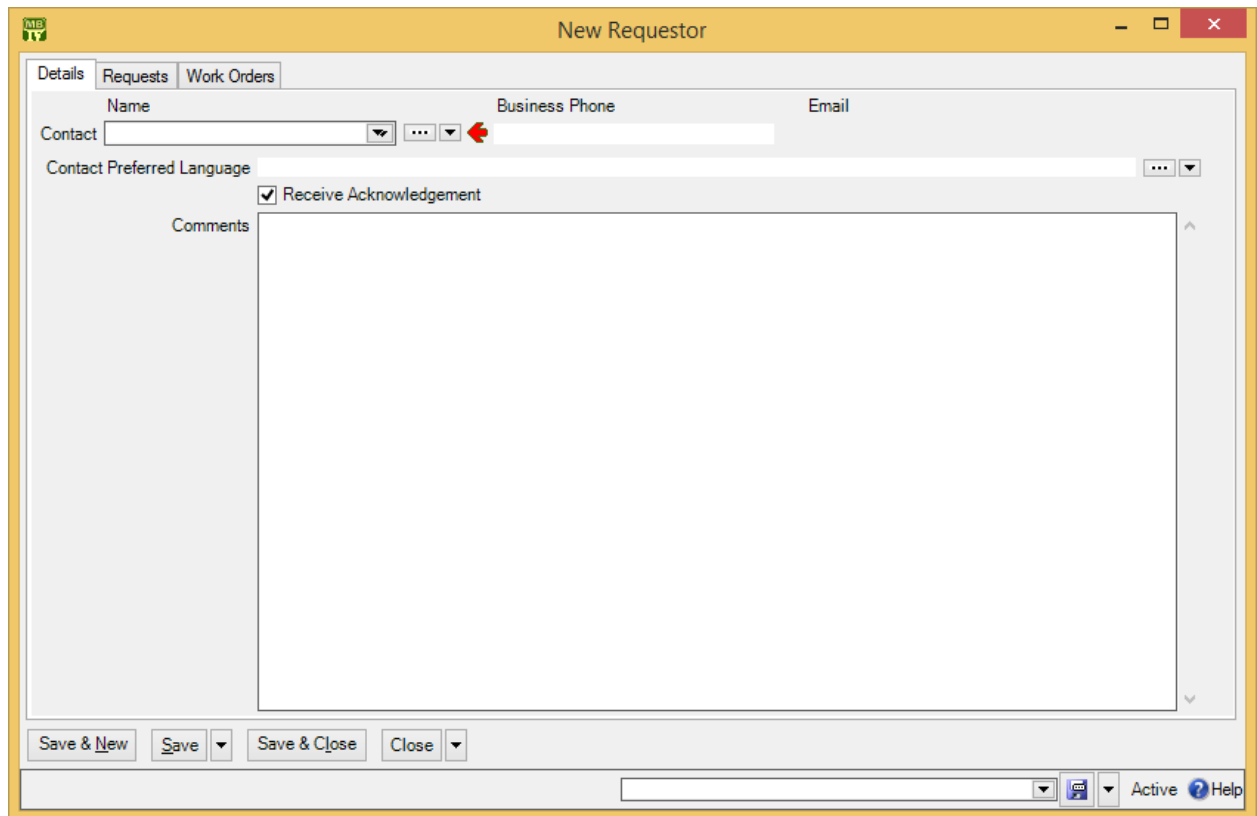
By clicking "..." or the right-hand arrow associated with "**Requestor Contact**", you can create a new **Contacts** record for the person in question. Once you specify a name in "**Requestor Contact**", MainBoss fills in "**Business Phone**" and "**Email**" with the chosen person's phone number and email address (as given in the **Contacts** table). MainBoss also fills in "**Comments**" with any comments in the person's **Contacts** record.

Receive Acknowledgements: This checkbox is used in connection with the **MainBoss Service** module. If the box is checkmarked (and if you have the **MainBoss Service** module), MainBoss will keep this requestor informed of the progress of any requests that the person submits. In particular, MainBoss sends email to inform the requestor when the request is marked as "in progress", when a work order is created from the request, and when the request is closed. This happens automatically, without maintenance personnel having to take any special action.

When Your Requestor Base is Huge: Your MainBoss license may limit the number of entries you may have in your **Requestors** table. In some cases, however, your maintenance department may accept requests from huge numbers of people; for example, a city's road-maintenance department could conceivably receive complaints about potholes from anyone in the city. In this case, your best solution might be to create a requestor record in the name of `Other`, or `General Citizen`, or something like that. When a request comes in from someone who is unlikely to submit many requests, you can record their request using this general requestor record, possibly using

- 81 -

the request's "**Comments**" field to record the requestor's actual name and contact information.

Requestor Sections: The requestor record contains the following sections:

> **Requests**: Shows requests submitted by this requestor. When you first create the requestor record, there won't be any such requests. Later on, however, this list can give you a good idea of how active this requestor has been.

> **Work Orders**: Shows any work orders associated with this requestor.

You don't fill in these sections during set-up. Instead, MainBoss automatically fills them in during day-to-day operations. For example, when you create a request for a particular requestor, the request will automatically appear in the **Requests** list of the requestor's record.

Request Priorities

A *request priority code* can indicate how urgent a particular request is. Here's a possible starting list

```
As soon as possible
By the end of the day
By the end of the week
Whenever
```

Each code has an associated numeric value so that MainBoss can sort them in order of urgency. Assign the lowest value to the most urgent code and successively higher values to less urgent codes.

Note: Priorities don't have to be given consecutive numbers. For example, you could assign 10 to the most urgent code, 20 to the next, 30 to the next, and so on. As long as the codes are in ascending order, MainBoss will sort them correctly. By leaving gaps between the numbers, you leave room for adding additional priorities if you decide they'd be useful...and by starting at 10, you make it possible to define even higher levels of urgency if it ever becomes necessary.

To create a request priority code, go to **Coding Definitions** | **Requests** | **Request Priorities** and click **New Request Priority**. This opens a window where you can define a priority code:

As usual, fill in "**Code**" with a brief code and "**Description**" with a fuller description. "**Rank**" indicates how the code compares to other priorities, with 1 being the most urgent. Whenever MainBoss lists priorities, it will list them from most urgent to least.

Work Order Priorities: MainBoss lets you create a second set of priority codes to be used on work orders. Work order priorities are separate from request priorities because work orders are created by maintenance personnel, while requests may be created by outsiders.

Your work order priority codes can be significantly different from your request priority codes. For example, you might have

```
Do immediately
Do within four hours
Do by end of shift
Do within 24 hours
Do this week
Do whenever
```

However, you'll probably find it easiest to have identical priority codes for both work orders and requests.

Many organizations find that the best policy is not to specify a priority at all, except in exceptional cases. If you usually don't fill in the priority field, a request marked Urgent has more impact; if every request has a priority, a request marked Urgent may not stand out as clearly.

Request Status Codes

Request status codes give you a way to indicate when a request needs someone's attention. For example, some maintenance departments have a policy of verifying certain types of problem reports before actually doing something about them. You might create a status `Waiting for confirmation` to mark a request which has been received from a client but which has not been verified by maintenance personnel. As another example, you might use requests to keep track of "wish-list" jobs: work you might do someday, but not in the immediate future. In this case, you might create a status `Long-Term` to mark requests that you want to remember but that you won't deal with anytime soon.

While it is possible to use status codes for any purpose, we recommend that you only use them for requests that are notable in some way. "Normal" requests should have their "**Status**" field blank.

To set up your request status codes, use **Coding Definitions** | **Requests** | **Request Statuses**. Clicking **New Request Status** lets you define a code.

Request Assignees

A request may be *assigned* to any number of people. Assigning a request to someone indicates that the request is of interest to that person. For example, you might assign a request to the requestor and to anyone who has an interest in seeing that the request receives timely attention (e.g. the help-desk person who received the request originally). Essentially, assigning a request to someone makes it easier for that person to check on the request's progress.

Before a request can be assigned to someone, that person must be listed in the **Request Assignee** table. The people listed in this table are authorized to be assigned requests.

To add someone to the **Request Assignee** table, the person should first be in the **Contacts** table. (For more on contacts, see *Contacts* on page 50.) Once you've created an appropriate contact record, go to **Coding Definitions** | **Requests** | **Request Assignees** and click **New Request Assignee**. In the "**Name**" field, specify the person's contact record.

The request assignee record also has a checkbox labeled **Receive Notifications**. If this is checkmarked, and if you have licensed the **MainBoss Service** module, MainBoss sends email to the assignee each time a history record is created for any request to which the person is assigned. In particular, the assignee will receive email when the request is marked "In Progress", when someone adds a comment to the request using **New Requestor Comment**, and when the request is closed. In this way, the assignee is automatically notified of significant changes in the request or its status.

The first notification sent to an assignee contains the original "**Description**" of the request. Subsequent notifications do not; they do, however, list all the comments associated with the request.

If an assignee can actually use MainBoss, that person can see his/her assigned requests in **My Assignment Overview** | **In Progress Requests**. This makes it easy for people to check up on any in-progress requests to which they've been assigned. For example, you might check your assigned requests first thing in the morning to see if there's anything new requiring your attention. You might also check to see if any requests have been sitting around for a while without making any progress.

Creating Requests

To begin creating a request, go to **Requests** in the MainBoss control panel. This displays a list of your current requests. Click **New Request** to begin creating a new request.

From this point on, it's just a matter of filling in the blanks: the requestor, the unit that needs attention, access code and priority (if appropriate), plus a "**Subject**" line summarizing the request and a "**Description**" providing as much detail as necessary. Once you've entered this information, you save it by clicking one of the **Save** buttons (e.g. **Save & Close** to save the request and close the window).

When you return to the list of the requests, you'll see that it now contains your new request. (If you don't see the new request, click **Refresh** to update the list.)

For full details about creating and using requests, see the <u>Day-to-Day Operations Guide</u>.

Request Defaults: If you turn on the **Requests are automatically set In Progress when linked to a work order, and are closed when work order closes** option, closing a work order automatically closes all linked requests, provided that the requests are only linked to this single work order. (If a request is linked to multiple work orders, you must close it manually—the request isn't closed automatically, even if you close all its linked work orders.) To turn on this option, go to **Requests** in the control panel, go to the **Defaults for Requests** section, and click **Edit Defaults**. Turning on this option also means that when you create a work order from a new request, the request is automatically put into the "in progress" state.

The **Edit Defaults** window also has a field labeled **Comment to requestor for requests that are automatically closed when their linked work order closes**. This field lets you enter a message that will automatically be emailed to requestors whenever one of their requests closes because a linked work order closed. For example, you might use this message to direct requestors to a web page that contains a user satisfaction survey. (Note that the message is only emailed out if you have licensed the **MainBoss Service** module.)

MainBoss Service

This chapter is only relevant if you have licensed the **MainBoss Service** module. If you do not have a license key for this module, the control panel will not contain entries related to MainBoss Service.

MainBoss Service is a piece of software which processes notifications sent to the assignees of requests, work orders, and purchase orders. It also processes requests submitted by authorized requestors via email.

In order to use the facilities of MainBoss Service, you must have a **MainBoss Service** license key. For information about installing MainBoss Service, see the Installation and Administration Guide.

Note: Earlier versions of MainBoss contained a facility named @Requests. MainBoss Service replaces @Requests; it does everything @Requests did but provides additional functionality.

Here's how MainBoss Service processes requests:

1. Users send email requests to a designated mail address.

2. On a regular basis (e.g. every half hour), MainBoss Service checks for incoming mail at the designated address. If there are new messages, each one is turned into a request and added to the **Requests** table. Once a request has been made in this way, MainBoss Service sends email to the requestor saying that the message has been received and processed.

3. Whenever the state of the request changes, MainBoss Service sends another acknowledgement message to the requestor. For example, when someone marks the request as **In Progress**, an appropriate email message is sent to the requestor.

MainBoss Service follows a similar system for sending out notifications: on a regular basis, MainBoss Service checks to see if any notifications should be sent out. If so, MainBoss Service does so.

You can configure how often MainBoss Service checks for incoming requests and how often it checks if notifications should be sent. These can have different schedules; for example, you might check on incoming requests every half hour and on outgoing notifications every ten minutes.

You control the operation of MainBoss Service using **Administration** | **MainBoss Service**. For information on all the features of configuring and running MainBoss Service, see the Installation and Administration Guide.

Customizing and Translating MainBoss Service Messages

MainBoss Service generates email messages to acknowledge the receipt of requests and for various other purposes. These messages are created by putting together various strings of text. For example, the subject line of a request acknowledgement is created from a string called `RequestorNotificationSubjectPrefix`, followed by the request's subject line.

MainBoss lets you customize these message strings. For example, you might add your organization's name to the subject line, or add a standardized comment that invites your clients to fill out a user satisfaction form. You can also create customized strings in other languages. For example, if you have a significant number of users who speak French, you can create a set of French strings; these will be used if the recipient's **Contact** record specifies a "**Preferred Language**" of French.

To specify a message string, go to **Administration** | **MainBoss Service** | **Configuration**, and click **Edit** to edit your configuration. Go to the **Messages** section of the record. This shows a list of the message strings you can edit. Each string has an identifier name that suggests how the string is used; for example, the `RequestDeniedPreamble` is used to begin a message to a requestor whose request has been denied (i.e. voided by someone with appropriate permissions).

To change an existing string or to supply a translation in some desired language, click on the string, then click **View**. After that, press **<F1>** and follow the online help to proceed.

Work Orders

MainBoss lets you prepare detailed work orders: the "who, what, where, when, how" instructions for performing a particular job.

Work orders can contain much more information than work requests. Here's a comparison:

☐ A request says something like: "There's a tap dripping in the first floor men's washroom."

☐ A work order says, "Chris, I'm assigning you to fix a dripping tap in the first floor men's washroom. The tap is to receive a Moen cartridge, obtained from Storeroom 103."

As this comparison suggests, requests are often submitted by people outside your maintenance department while work orders are prepared maintenance personnel (either a dispatcher or a manager).

> Work orders aren't just useful for telling workers what to do—they also preserve a history of what got done. After a job is finished, it's important to record the parts and labor used, as well as any other information that might be useful to remember.

Work Order Status Codes

Work order status codes give you a way to indicate when a work order needs someone's attention. For example, you might create a status `Waiting for approval` to mark a work order which has been written up but not approved by management. Other possible status codes might include:

```
Waiting for parts
Waiting for tenant sign-off
Waiting for government inspection
Temporarily suspended by customer request
```

> While it is possible to use status codes for any purpose, we recommend that you only use them for work orders that are notable in some way. "Normal" work orders should have their "**Status**" field blank.

Note that status codes can be a useful form of communication between users. For example, suppose that a worker has the `WorkOrderFulfillment` security role (meaning that the worker can record actual labor and material usage) but does not have the `WorkOrderClose` security role (which is needed to close a work order). The worker could give the work order a status like `Ready to Close`, indicating that all appropriate information has been recorded. A supervisor with `WorkOrderClose`

permission could regularly check all open work orders to see if any are `Ready to Close`; if so, the supervisor can review those work orders and close them as appropriate (removing the status code in the process).

> Remember that a status code is intended to bring a work order to someone's attention. You can therefore use status codes as signals for other users.

To set up your work order status codes, use **Coding Definitions** | **Work Orders** | **Work Order Statuses**. Clicking **New Work Order Status** lets you define a code.

Workers

Work orders are usually assigned to specific workers. In order to do this, you have to create a list of your employees using **Coding Definitions** | **Work Orders** | **Labor** | **Employees**. Clicking **New Employee** opens a window where you can record an employee's name:

In "**Contact**", you specify a record in the **Contacts** table giving the employee's contact information. (The **Hourly Inside** and **Per Job Inside** areas are discussed later in this chapter.)

> The **Employees** table is used for your own personnel. To record information about outside contractors, use the **Vendors** table (as discussed in *Vendors* on page 52).

Trades

You can categorize your employees by *trade* (e.g. job title or classification) using **Coding Definitions** | **Work Orders** | **Labor** | **Trades**. Here's a sample list:

```
Electrician
Electrician Apprentice
Cleaner
Engineer - Chief
Engineer – Class 1
```

Your trade categories apply to both your employees and to outside contractors.

Labor Costs

If you want to keep track of labor costs, you have to tell MainBoss the cost of each worker, inside and outside. There are two ways in which MainBoss can record the cost of work:

☐ *Hourly:* You specify the hourly rate for labor by a particular worker on a particular type of job.

☐ *Per Job:* You specify a flat rate to be charged for a particular type of job, no matter how long it takes or who performs the work.

For example, consider a car mechanic. For routine jobs like oil changes, a mechanic might charge a flat fee, regardless of how long the job takes. For other jobs, the mechanic might charge by the hour, based on the time actually spent.

> **Important:** Although MainBoss lets you keep track of your labor costs, it should not be considered a comprehensive payroll system. For example, it doesn't keep a running total of how much each person works, so it can't say, "Joe has already worked 40 hours this week; now he should be paid overtime." This is the sort of thing you have to keep track of on your own.

Chargebacks for Labor: Your organization may charge customers for labor on some types of jobs (e.g. if a customer requests special renovations). MainBoss completely separates these chargeback costs from actual employee costs. This makes it possible to charge customers a different amount than you actually paid to employees. Configuring chargeback information is discussed in *Chargebacks* on page 96.

Same Person, Different Charges: It's possible to have several records for the same worker. For example, if Chris Smith gets paid extra for overtime, you might have one Hourly record for Chris's normal rate and another for overtime. When assigning personnel to a job or recording labor expenses, you should make sure to choose the correct record.

As another example, you might have one Hourly record that reflects the worker's actual wage and a different one that uses the combined price of the worker's wages, benefits, overhead, etc. (sometimes called the *chargeout* rate). Some organizations may find it more useful to record labor expenses using the combined chargeout rate rather than wages alone.

As a final example, using multiple records for the same person is helpful when the worker is scheduled for a pay raise. For more information, see *Rate Changes* on page 94.

> The more complex your labor-pricing structure, the more hourly and per job records you'll need. If your cost structure is simple, your MainBoss records can be simple. If your cost structure is complicated, MainBoss's records have to deal with the complications.

When recording the cost of a specific job, you can always write in a different cost than the one in your records. Therefore, it's easy to handle one-time-only variations from standard labor charges.

Hourly Inside

You set up hourly charge records for your own employees by using **Coding Definitions** | **Work Orders** | **Labor** | **Hourly Inside** (where "inside" means that the cost applies to an employee, as opposed to an outside contractor). Clicking **New** opens a window that looks like this:

The record includes the following:

☐ A "**Code**" identifying the record (plus the usual longer "**Description**")

☐ "**Employee**" refers to a record in the **Employees** table—the employee whose costs you're recording

☐ The employee's "**Trade**"

☐ The "**Cost**" (hourly rate) to be charged for the worker

☐ The "**Cost Center**" to be associated with paying this worker

As noted earlier in this chapter, you can have multiple hourly inside records for the same worker. For example, you might have one record for a worker at normal pay and a second record for jobs that must be charged at time-and-a-half overtime.

Because you can have multiple cost records for the same worker, it's important to use unambiguous identification codes so you don't assign the wrong cost to a job. For example, you might use codes like

```
John Smith Normal
John Smith Overtime
```

to distinguish between John Smith's normal cost and his cost for overtime.
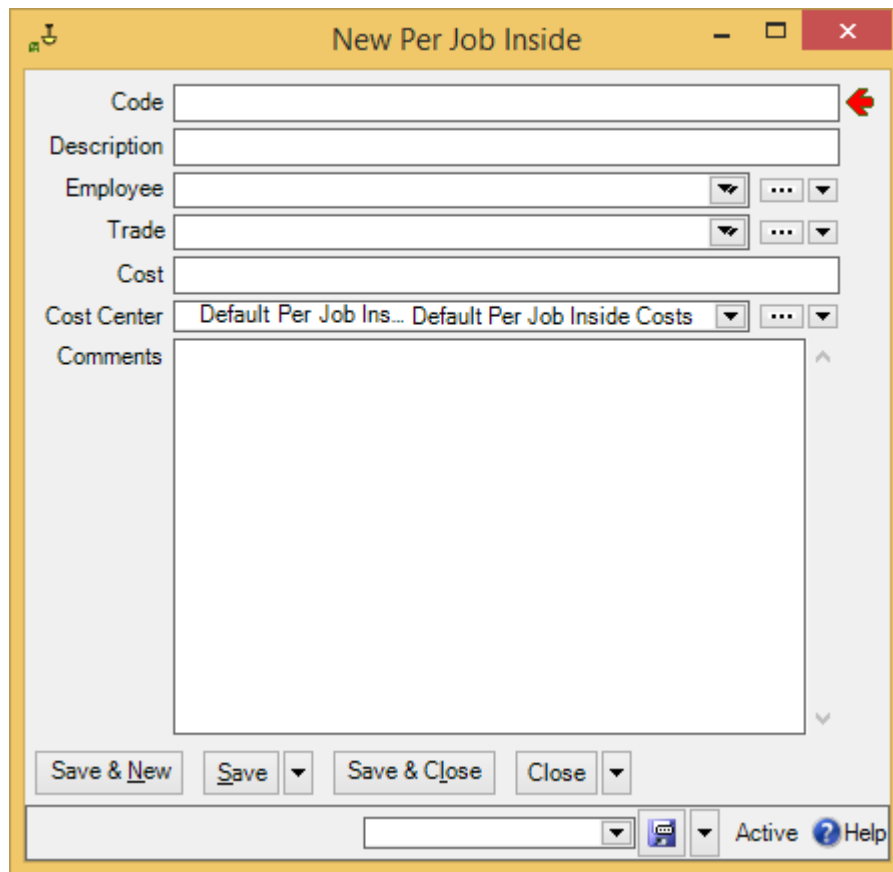
Generic Costs: In an hourly inside record, the "**Employee**" name isn't mandatory. This means you can create records that are associated with a trade or an employee class rather than a specific person.

For example, if you have several electricians who all cost the same hourly rate, you can create an hourly inside record with a "**Code**" like `Electrician` to stand for all your electricians. When you create a work order, you just say that it's being assigned to an Electrician rather than specifying someone in particular.

In many cases, using generic labor assignments is better than assigning a job to a specific person, since it establishes a standard cost for the job.

Per Job Inside

Per job pricing is simpler than hourly pricing, since it doesn't depend on the workers involved. You set up per job records by using **Coding Definitions** | **Work Orders** | **Labor** | **Per Job Inside**. Clicking **New Per Job Inside** opens a window that looks like this:

\
The record includes the following:

A "**Code**" identifying the record (plus the usual longer "**Description**")

 The cost of the job

 The "**Cost Center**" to be associated with paying for such work

Outside Contractors

You can assign work to outside contractors as well as to your own employees. For example, suppose you deal with plumbing problems by calling in an independent plumber. You should still write up work orders for plumbing jobs, because it's important to track *all* the maintenance work done on your premises.

Outside contractors are considered *vendors*—they sell you their services just as suppliers sell you materials. Therefore, you record information about outside contractors using the **Vendors** table. For more information, see *Vendors* on page 52.

You keep track of contractor costs with the same sort of records used with your own personnel.

 Coding Definitions | **Work Orders** | **Labor** | **Hourly Outside** lets you specify the hourly rates for outside contractors. As with your own employees, you have to set up multiple records for a contractor if the contractor charges you different hourly rates for different types of jobs.

 Coding Definitions | **Work Orders** | **Labor** | **Per Job Outside** lets you specify rates when outside contractors charge on a per job basis.

Hourly outside records are filled out much like hourly inside records, and per job outside records are similar to per job inside. As with your own inside labor records, you may need a lot of hourly outside and per job outside records if you have to account for complex pricing structures.

Rate Changes

If labor costs change for some reason (e.g. workers getting pay raises), you have to phase in the change to make sure that your database accurately reflects past, present, and future.

As an example, suppose that your Electricians are scheduled to get a raise from $30/hour to $31/hour as of September 1. You currently have an hourly inside record with the ID code `Electricians` that says their hourly rate is $30.

 If you merely change the `Electricians` record from $30 to $31, you immediately change all existing electrician demands to use the rate of $31/hour. This includes any demands on jobs that have already been finished but have not yet been actualized. Presumably, work on those jobs should still be charged at $30, not $31.

☐ In the period immediately before and after the rate change, you also have to make sure you assign the right cost to the work orders you create. For example, suppose you write up a number of work orders for electricians on August 31. If you schedule an electrician job for the same day, it should use the old rate; if you schedule a job for September 1 or later, it should use the new rate.

The way to handle this is with two records. Take the old `Electricians` record and change the code to something like `Electricians before Sept 1 2008`. Then create a new record with an hourly rate of $31 and a code like `Electricians after Sept 1 2008`. (Remember that you can have multiple price records for any worker.) Jobs scheduled before the rate change use the first record; jobs scheduled afterward use the second.

What happens if a job was scheduled for before the rate change but gets delayed and takes place after? You have to change the job assignment so that it refers to the correct rate for the electrician's work.

☐ After you've closed all the work orders done by electricians before September 1, there's no reason to keep the `Electricians before Sept 1 2008` record. You can therefore delete that record. Remember that deleting a record doesn't entirely erase it; the record is simply moved to a list of deleted records. MainBoss still retains the information that electricians were once paid $30/hour and all work orders referencing that record still refer to the correct price.

Once you delete `Electricians before Sept 1 2008`, you can rename `Electricians after Sept 1 2008` simply to `Electricians`. Why isn't this confused with the old $30 electrician rate? Because the old $30 record still exists—deleted records never totally disappear—and any work orders charged at $30 are still connected to the old record. It doesn't matter that the new record has the same code (`Electricians`) that the old record once had.

This process may sound complicated, but it's really no different than what you have to do when you want to switch the names of two files. If you want to change File1 to File2 and vice versa, you have to change File1 to some other name (e.g. X), then change File2 to File1 and change X to File2.

Note that it's useful to put the year into the names of these records. As wages change over time, you'll eventually have records like

```
Electricians before Sept 1 2008
Electricians before Sept 1 2009
Electricians before Sept 1 2010
```

If you don't put a year in the record names, you're in for a lot of confusion.

Work Order Assignees

Assigning a work order to someone is similar to assigning a request (as discussed in *Request Assignees* on page 84). Assigning a work order to someone indicates that the work order is of interest to that person. For example, you might assign a work order to the workers who'll do the job and to their supervisor. Assigning a work order to someone makes it easier for that person to check on the job's progress.

Before a work order can be assigned to someone, that person must be listed in the **Work Order Assignee** table. The people listed in this table are authorized to be assigned work orders.

To add someone to the **Work Order Assignee** table, the person should first be in the **Contacts** table. (For more on contacts, see *Contacts* on page 50.) Once you've created an appropriate contact record, go to **Coding Definitions** | **Work Orders** | **Work Order Assignees** and click **New Work Order Assignee**. In the "**Name**" field, specify the person's contact record, then click **Save & Close**.

If an assignee can actually use MainBoss, that person can see his/her assigned work orders in **My Assignment Overview** | **Open Work Orders**. This makes it easy for people to check up on any open work orders to which they've been assigned. For example, a worker might check assigned work orders first thing in the morning to see if he/she has been given any new jobs.

As with request assignees, you can use the **Receive Notifications** checkbox to indicate that work order assignees should be informed of changes to the work order history. The first such notification includes the original "**Work Description**". Subsequent notifications omit this information, but give comments from the work order history records.

Chargebacks

A *chargeback* record indicates when some or all of the costs for a particular job are charged to an outside party. For example, property management companies may charge tenants for certain types of work.

Billable Requestors

To set up for chargebacks, you must first create a list of *billable requestors* (**Coding Definitions** | **Work Orders** | **Billable Requestors**). This is simply a list of people who may be charged for work orders. You set up this list exactly like the **Requestors** list (see *Requestors* on page 80).

For each billable requestor, you have the option of specifying a cost center to be associated with bills sent to the requestor (i.e. accounts receivable from the requestor).

Chargeback Categories

It's useful to create a list of *chargeback categories* (**Coding Definitions** | **Work Orders** | **Chargeback Categories**). These codes let you divide chargebacks into categories. Here's a simple list:

```
Repairs
Renovations
Leasehold Improvements
Special Requests
```

With each chargeback category, you have the option of specifying a cost center to be associated with costs in that category.

> Once you have set up lists of chargeback codes and billable requestors, you can add chargebacks to work orders. For details about how to create chargebacks, see the Day-to-Day Operations Guide.

Closing Codes

When you finish a job, you *close* the work order to indicate that the job is done. At that point, you have the opportunity of specifying a *closing code*. Many companies use closing codes to indicate what necessitated the work in the first place; this lets you track the type of problems your operation deals with. Here's a possible starting list:

```
Normal wear and tear
Operator error
Breakage
Vandalism
Poor lubrication
Routine planned maintenance
```

("Routine planned maintenance" indicates that the work order dealt with planned maintenance, as opposed to repairing an actual problem.)

By attaching such codes to your work orders, you get an overall picture of the problems you face. For example, how much does vandalism actually cost you? Is it negligible, or is it substantial enough that your company should take some kind of action? If poor lubrication causes a significant number of problems, should you change your planned maintenance timing to lubricate more often?

The more you know about why problems happen, the better armed you are to prevent them in future. Closing codes are a simple way to record important knowledge.

Other Work Order Information

Work orders can use codes to provide information. For example, a *priority code* can indicate how urgent a particular work order is. Such codes are optional—you can create work orders without them—but they're also quite simple to set up. The codes include:

Priorities (**Coding Definitions** | **Work Orders** | **Work Order Priorities**): Indicate how urgent the work order is. Our preferred approach is to define priority codes in terms of time. Here's a possible starting list:

```
Immediately
Sometime today
In the next three days
Within a week
Within a month
Whenever
```

Each code has an associated numeric value so that MainBoss can sort them in order of urgency. You should assign a low value to the most urgent code and higher values to less urgent codes. However, the values don't have to be sequential. For example, you might use the values 10, 20, 30, etc. so that there's room between your existing codes, in case you decide to add more codes at some later time.

Note: As noted in *Request Priorities* on page 82, work orders and requests may have different priority codes. Request priorities are usually simple while work order priorities may allow for more detail.

Projects (**Coding Definitions** | **Work Orders** | **Projects**): A *project code* is a way to indicate when a number of work orders are all connected with each other. For example, suppose your department undertakes a large renovation and you want to track work orders associated with that renovation. You would create a project code for the renovation and assign that code to any work order related to the renovation. MainBoss reports can be sorted by project, so that you can see various types of information on a project-by-project basis.

Note that it's also useful to define project-specific expense models and expense categories to make it easy to separate one project's expenses from another.

Work categories (**Coding Definitions** | **Work Orders** | **Work Categories**): Work categories can be used to classify work orders any way you like, but most companies use them to state the kind of work done. Here's a starting list:

```
Plumbing
Electrical
Mechanical
Inspection
Lubrication
Adjustment
```

```
Clean-up
Vehicle maintenance
Air conditioning
Outdoor work
Work generated from inspection
```

> `Work generated from inspection` is a very useful category to track, since it lets you record when planned maintenance inspections discover problems *before* they lead to actual breakdowns.

In addition to codes and categories, the coding definitions associated with work orders include:

Miscellaneous Costs (**Coding Definitions** | **Work Orders** | **Miscellaneous Costs**): Used when a work order includes costs that aren't labor and materials, and that aren't covered by a purchase order—for example, when a worker has to pay for parking at a job site. Each miscellaneous cost has an associated cost center.

Some organizations won't define miscellaneous cost records during initial set-up. You'll simply create such records if and when they become necessary during day-to-day operations. (However, this requires that the person entering work orders has appropriate security roles to create miscellaneous cost records.)

Other organizations will define a small set of miscellaneous cost records in order to specify what types of costs can actually be charged as "miscellaneous". Essentially, this means writing down what kind of costs your organization will usually reimburse. Anything else will be considered unusual and may require special authorization.

Work Order Defaults

The **Defaults for Work Order** section of the **Work Orders** table viewer lets you control important aspects of the work order. In particular, you can specify:

☐ The format used for work order identification numbers. The default format is `WO 00000001`, `WO 00000002`, and so on. However, you can change this to suit the needs of your organization.

☐ The starting point for numbering work orders. For example, if your organization has already used work orders up to 10,000, you can tell MainBoss to start numbering at 10,001.

☐ Additional text that should appear on a printed work order. For example, you can add blank lines in case workers need to write in materials that were used but weren't on the original work order. You can also add lines for authorization signatures and other information that workers are expected to fill in.

To set any of these defaults, go to **Work Orders** in the control panel, click on the **Defaults for Work Order** section, then click **Edit Defaults**. MainBoss will open a window where you can change the current defaults. For information about the default window, see the online help (click the Help button at the bottom of the window, or press **<F1>**).

Creating Work Orders

Here are some ways you can create a work order:

- ☐ **From scratch:** Go to **Work Orders** in the control panel, then click **New Work Order**. This opens a window containing a blank work order. You can then fill in all the blanks.

- ☐ **From an existing request:** Go to **Requests**, select a request, then click **Create Work Order**. MainBoss opens a new work order, with some of the fields already filled in. For example, the "**Subject**" line of the work order will be the same as the original request.

- ☐ **From a planned maintenance job:** Go to **Unit Maintenance Plans**, select a unit maintenance plan, then click **Create unplanned Maintenance Work Order**. MainBoss opens a new work order, with most of the fields filled in from the selected plan. (For more on unit maintenance plans, see *Planned Maintenance* on page 104.)

From this point on, it's just a matter of filling in the blanks. For full details, see the [Day-to-Day Operations](#) guide and the online help.

Expense Categories

In order to analyze your expenses, it's useful to categorize every expense on a work order. If you have licensed the MainBoss **Accounting** module, you can create named expense categories. For example, you might have categories like

```
Inside labor
Outside labor
Spare parts
Electrical
Plumbing
Work materials
```

and so on. By labeling such expenses on every work order, you make it possible for MainBoss to produce analysis reports answering such questions as, "How much did we spend on outside labor this month?"

In order to categorize your expenses, you define expense categories; the preceding list gave some examples. You define expense categories using **Coding Definitions** | **Work**

Orders | **Expense Categories**. The process for defining expense categories is similar to the process for defining work categories: click **New Expense Category**, then enter a "**Code**" and "**Description**".

Expense categories may be restricted to certain uses. For example, you might specify that an expense category can be used for labor expenses but not for inventory items or for miscellaneous expenses. This prevents expense categories being used for inappropriate entries.

Expense Models

An *expense model* tells what expense categories are allowed on a particular work order. This means that you can select different sets of categories for different work orders.

Expense models are only available if you have licensed the MainBoss **Accounting** module.

For example, a property management company might have a different expense model for each client. The expense model for Client X might contain expense categories like

```
Inside labor, Client X
Outside labor, Client X
Spare parts, Client X
```

and so on. You'd define similar categories for each client. This makes it possible to separate each client's expenses; this would allow you to compare expenses between clients.

Many organizations will have no need for multiple expense models—you can simply use the same set of expense categories for every work order. However, organizations that serve multiple clients in the same locations may find it useful to be useful to separate expenses by client. This is done with multiple expense models containing different expense categories for different clients.

Using Expense Models: When you create a work order, you specify the expense model you want to use. For example, suppose you have separate models for each client; when you create a work order for Client X, you specify Client X's expense model. Then, when you categorize each expense on the work order, you will only see the expense categories that are available for this particular expense model.

In practice then, expense models are easy to use. Just specify the correct one for the work order that you're writing up. Once you do that, you categorize expenses on the work order; the only categories you see are the ones that are allowed in the model you've chosen.

Creating Expense Models: To create an expense model, we strongly recommend doing so in the following order:

1. Create the expense categories first (as described in the previous section of this guide).

2. Create the expense model record and save it.

3. Go to the expense model record's **Expense Mappings** section and create the expense mappings.

4. Save the expense model record again (if necessary), then set the default expense categories in the **Details** section of the expense model record.

Details of this process are given below.

To create an expense model, first create all the expense categories that should belong to the model. Then use **Coding Definitions** | **Work Orders** | **Expense Models**. Click **New Expense Model** and MainBoss opens a window where you can define a new expense model.

As usual, you fill in "**Code**" and "**Description**" with identification information. Do not fill in the other fields yet (e.g. "**Item default Mapping**", "**Hourly Inside default Mapping**", etc.).

Next **Save** the expense model record. Then go to the **Expense Mappings** section and click **New Expense Mapping**. This opens a window where you can specify an

additional expense category to belong to this expense model. You can add as many categories as you want to the model.

The window for adding an expense category to an expense model also lets you specify a cost center to be used for this expense category within this expense model. Items charged to this expense category while using this expense model will be charged to the associated cost center. For example, in the expense model for Client X, you might specify that inside labor expenses should be charged to cost center 1234.

In different expense models, the same expense category may be associated with different cost centers.

For more on the relationship between expense models, expense categories, and cost centers, see *Accounting Facilities* on page 37.

Once you have set up all the expense mappings in the expense model's **Expense Mappings** section, you can return to the **Details** section of the expense model record. The fields "**Item default Mapping**", "**Hourly Inside default Mapping**", and so on, are filled in with the expense categories that should be used as default for items, hourly inside expenses, and so on. Note that these are *defaults*; MainBoss uses them when you first create this type of expense on a work order. However, an expense model can include additional expense categories which can be used instead of the default.

For example, "**Item default Mapping**" might be the expense category for `Spare Parts`. However, the expense model may include additional categories for other types of items that might be used in a job.

Default Expense Models and Categories: MainBoss has built-in expense categories and a single built-in expense model containing all the built-in categories. If you don't need sophisticated expense tracking, you can simply use the defaults. This will assign one of the built-in categories to each expense you put on the work order. For many organizations, the defaults are all you'll ever need.

Planned Maintenance

This chapter is only relevant if you have licensed the **Planned Maintenance** module. If you do not have a license key for this module, the control panel will not contain entries related to planned maintenance.

Planned Maintenance lets you schedule regular planned maintenance jobs. The process works like this:

1. You specify the timing when work should be done. For example, you might specify "every three months" and/or "every 3000 miles".

2. You set up descriptions of the PM work you want to do. These descriptions are called *tasks*. For example, a task might provide step-by-step instructions or a checklist of equipment that the worker should inspect.

3. You associate specific units with tasks and timing. For example, a task might describe an oil change and a timing record might say "every three months". Associating a company vehicle with the oil change task and the "every three months" timing tells MainBoss that you want that vehicle to get an oil change every three months. The combination of a unit, a task, and a timing record is called a *unit maintenance plan*.

4. On a regular basis, you tell MainBoss to generate planned maintenance work orders. For example, every Friday you might generate PM work orders for the following week.

5. MainBoss checks its timing schedules to see which PM jobs will come due in the specified period. It then creates work orders for each of these jobs. These work orders are exactly like the work orders you create by hand, but they're filled in with information from the task description.

6. After MainBoss generates the PM work orders, you should review them in case any changes are necessary. For example, a unit maintenance plan may say that Worker X is always assigned to Job Y, so that's how MainBoss generates the work order. However, if Worker X is on vacation this week, you have to decide whether to assign the job to somebody else or else delay the job until Worker X gets back from holiday.

Timing by meter: For jobs whose timing is based on meter readings, MainBoss looks at all past meter readings for the unit and estimates when the unit is next likely to require maintenance. For example, suppose a car needs an oil change every 3000 miles, and suppose that you generate PM work orders every week. MainBoss looks at past mileage readings for the car and estimates whether the car is likely to hit the next 3000-mile mark in the coming week. If so, MainBoss creates a work order for the oil change.

In order for this process to work, you should record meter readings on a regular basis; otherwise, MainBoss doesn't have enough information to estimate future readings. (For information on meters, see *Meters* on page 74.)

Maintenance Timing

A maintenance timing record represents information like "every three months" or "every second Tuesday" or "every 5000 kilometers". You define maintenance timing records using **Coding Definitions** | **Unit Maintenance Plans** | **Maintenance Timings**.

Most maintenance timing records are simple. However, some maintenance organizations have complicated scheduling needs ("every ten days, but weekends don't count and never schedule the job on Mondays because Joe doesn't work that day"). MainBoss can accommodate such complexities, but setting up the corresponding timing record may be tricky.

Enabled Weekdays: In many organizations, some or all of the personnel only work on weekdays. Therefore, planned maintenance jobs shouldn't be scheduled on weekends. Other organizations, however, work every day of the week. In fact, there may be certain

jobs which should *only* be scheduled on weekends, to avoid interfering with weekday operations.

To accommodate such considerations, MainBoss lets you specify which days of the week are acceptable for a particular job. In the picture above, the record has checkmarks for Monday through Friday, but not for Saturday or Sunday. This says that any jobs using this timing record should only be performed on weekdays, not on weekends. Therefore, MainBoss will never schedule such jobs for Saturday or Sunday.

As another example, if you only checkmark Wednesday (with all other days blank), MainBoss will only schedule corresponding jobs on Wednesdays. This is something you might do if you want to schedule a job for the first Wednesday in every month. (However, to get the *first* Wednesday, you have to do more; this is explained later.)

Seasons: MainBoss lets you create seasonal timing records. For example, you might perform planned maintenance on air-conditioners in summer but not in winter; you might perform planned maintenance on heaters in winter but not in summer. Therefore, every timing record lets you set "**Season Start**" and "**Season End**" values if you wish. For example, if you set "**Season Start**" to May 1 and "**Season End**" to October 31, corresponding maintenance jobs will only be scheduled between those two dates.

The example using the fourth of May shows you the format you should use to enter dates in the "**Start**" and "**End**" fields. This format depends on your personal Windows settings and may be different for different people.

Deferrals: If you tell MainBoss not to schedule jobs for certain days of the week or for certain times of year, what should MainBoss do if a job comes due on a date that's not allowed?

For example, suppose you say that a job should be scheduled every 30 days. What happens if the next job should be scheduled for a Saturday, but Saturdays aren't permitted?

Normally, you want MainBoss to schedule the job for the next permissible day. For example, if the job should be scheduled on Saturday but you've ruled out weekends, you probably want MainBoss to schedule the job for the first acceptable day (Monday). In some cases, however, it may be more appropriate to skip the job completely. This is particularly true of seasonal jobs; if you want to skip certain jobs in summer, you probably don't want them all to be scheduled for the first day after summer ends.

Therefore, MainBoss gives you the option of saying, "Schedule as soon as possible after the date" or "Just skip the job and catch it the next time". This applies in three different situations:

> Overdue work orders: A work order is *overdue* when the job comes due before the date when you generate planned maintenance work orders. For example, you generate new work orders on June 4, but there was a job you should have done on June 1. Your options are:

Defer overdue work orders to the date maintenance is generated: In our example, this means scheduling the job for June 4, the day you generate the work orders.

Inhibit generation of overdue work orders: This means skipping the work entirely—you missed the June 1 date, so you'll do the job the next time it comes around on the schedule.

Seasonal exceptions: As noted previously, seasonal jobs are only performed during a certain season. For example, you may only service air-conditioners during the summer. If a job comes due outside the allowed season, you have these options:

Defer work orders triggered outside the season until the start of the season: In our air-conditioner example, this means scheduling the next inspection for the starting day of the summer months.

Inhibit generation of work orders outside the season: This means skipping the work entirely—if a job comes due outside the season, just ignore it. You only do jobs that come due within the season.

Weekday exceptions: If a job comes due on an unacceptable day of the week, you have the following options:

Defer work orders triggered on disabled weekdays until the next enabled weekday: For example, if a job can only be done on Saturdays and it comes due on a Thursday, you schedule the job for the Saturday as soon after the Thursday as possible.

Inhibit generation of work orders on disabled weekdays: This means skipping the work entirely—if a Saturday job comes due on a Thursday, you just don't do it. You keep skipping the job until it actually comes due on a Saturday.

Timing Periods: There are three ways to specify maintenance timing:

☐ A *daily period* specifies timing in terms of days, e.g. "every 90 days".

☐ A *monthly period* specifies timing in terms of months, e.g. "every 3 months".

☐ A *meter period* specifies timing in terms of meter readings, e.g. "every 3000 miles".

☐ **To specify daily period timing:**

1. In the control panel, go to **Coding Definitions** | **Unit Maintenance Plans** | **Maintenance Timings**.

2. Click **New Maintenance Timing**. This opens a window where you can create the schedule record.

3. In "**Code**" ❶, enter an identification code for this schedule.

4. In "**Description**" ❷, enter a longer description of the schedule.

5. Click **Save** ❸ to save this information.

6. Go to the **Periods** section:
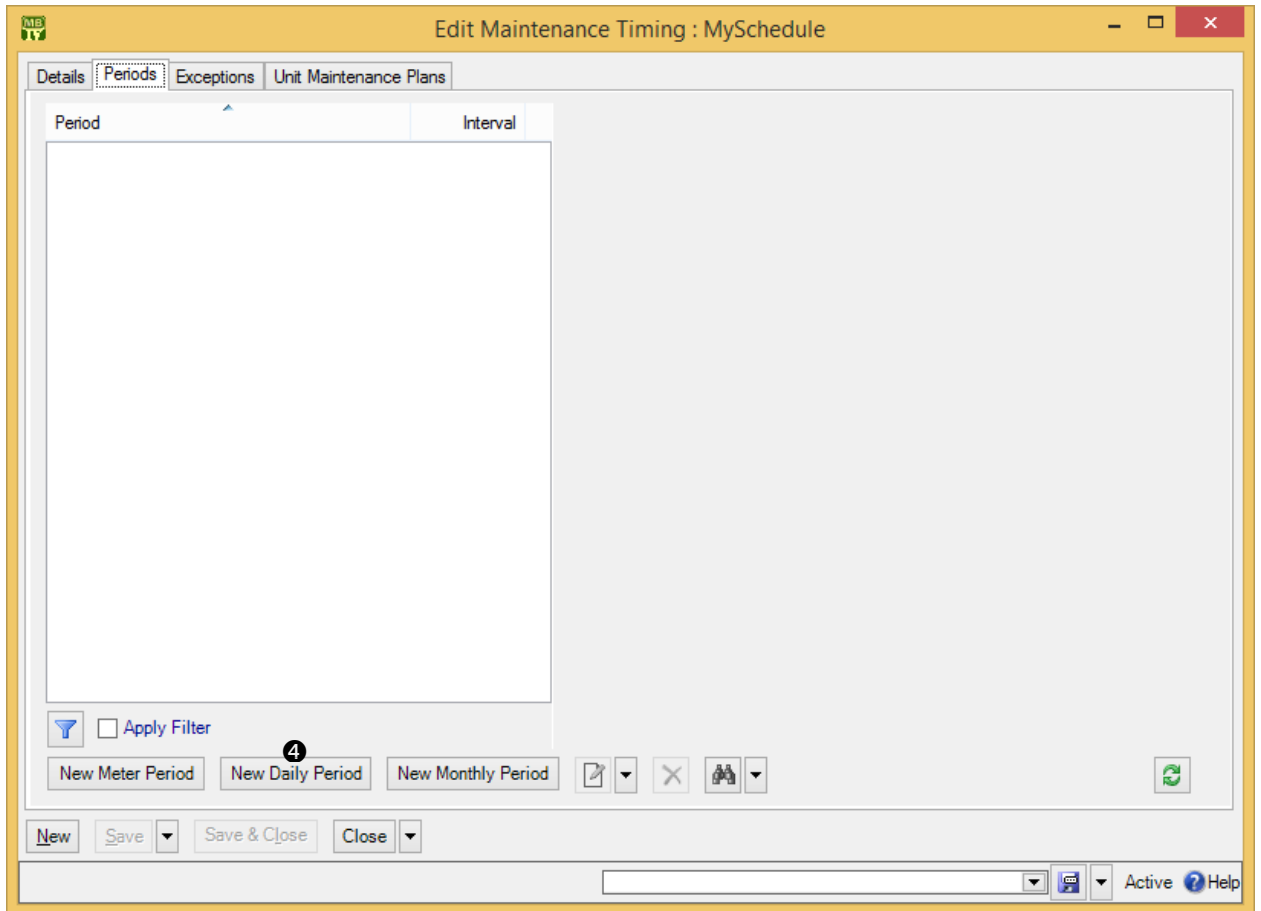
7. Click **New Daily Period** ❹. This opens a window where you can specify the number of days between jobs.
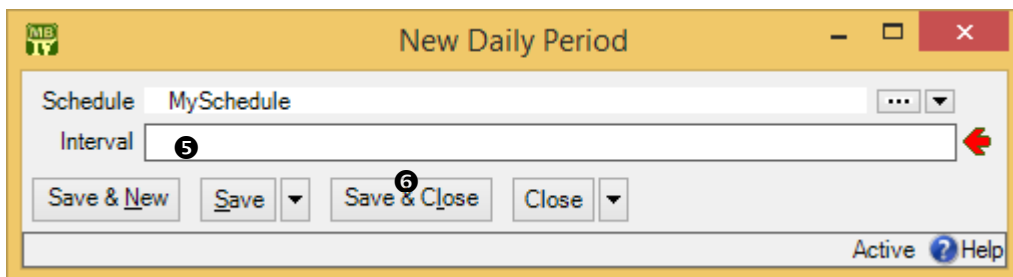
8. In "**Days**" ❺, enter the number of days between jobs (e.g. 30 if the schedule should be "every 30 days").

9. Click **Save & Close** ❻ to close the daily period window. You'll return to the schedule record. MainBoss will now show the daily period in the **Periods** list.

10. Click **Exceptions** ❼ to go to the **Exceptions** section.

11. Checkmark the days of the week when the work may be carried out. Leave the other days blank.

12. For seasonal schedules, fill in "**Season Start**" ❽ with the starting date of the season and "**Season End**" ❾ with the ending date. (The example at the end of this line shows what date format is expected. This can vary from user to user, depending on your Windows settings.)

13. Checkmark any inhibits that apply. (Inhibits were discussed earlier in this section.)

14. Click **Save & Close** ❿ to save your data and close the window.

The process of setting up monthly timing is similar. Using **New Monthly Period,** you specify the number of months between jobs.

Similarly, the process of setting up a meter period starts by clicking **New Meter Period** in the schedule record:
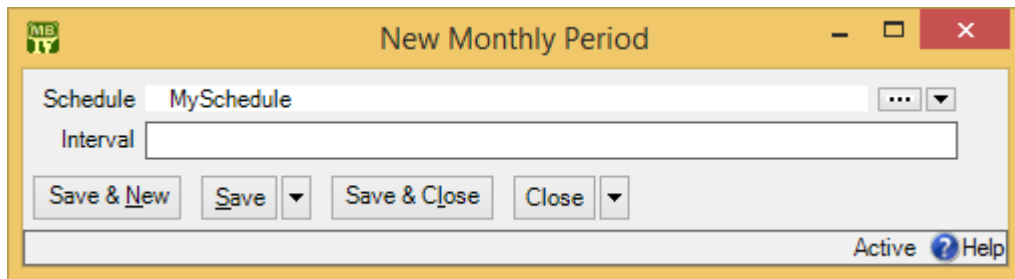
In "**Interval**", you enter the difference in meter readings between one job and the next. For example, if a job should be done every 3000 miles, set "**Interval**" to 3000. In "**Meter Class**", you specify the class of the meter that will be used. (MainBoss automatically fills in the "**UOM**" field to show the meter class's units of measurement.)

Combined Timing: A timing record can combine multiple types of periods: typically a date period and a meter period. For example, suppose an oil change should take place every three months or every 3000 miles, whichever comes first. To set up such timing, you first click **New Monthly Period** to set up an "every three month" schedule. Then, in the same timing record, you click **New Meter Period** to set up an "every 3000 miles" schedule. This means that the **Periods** section will contain two entries:
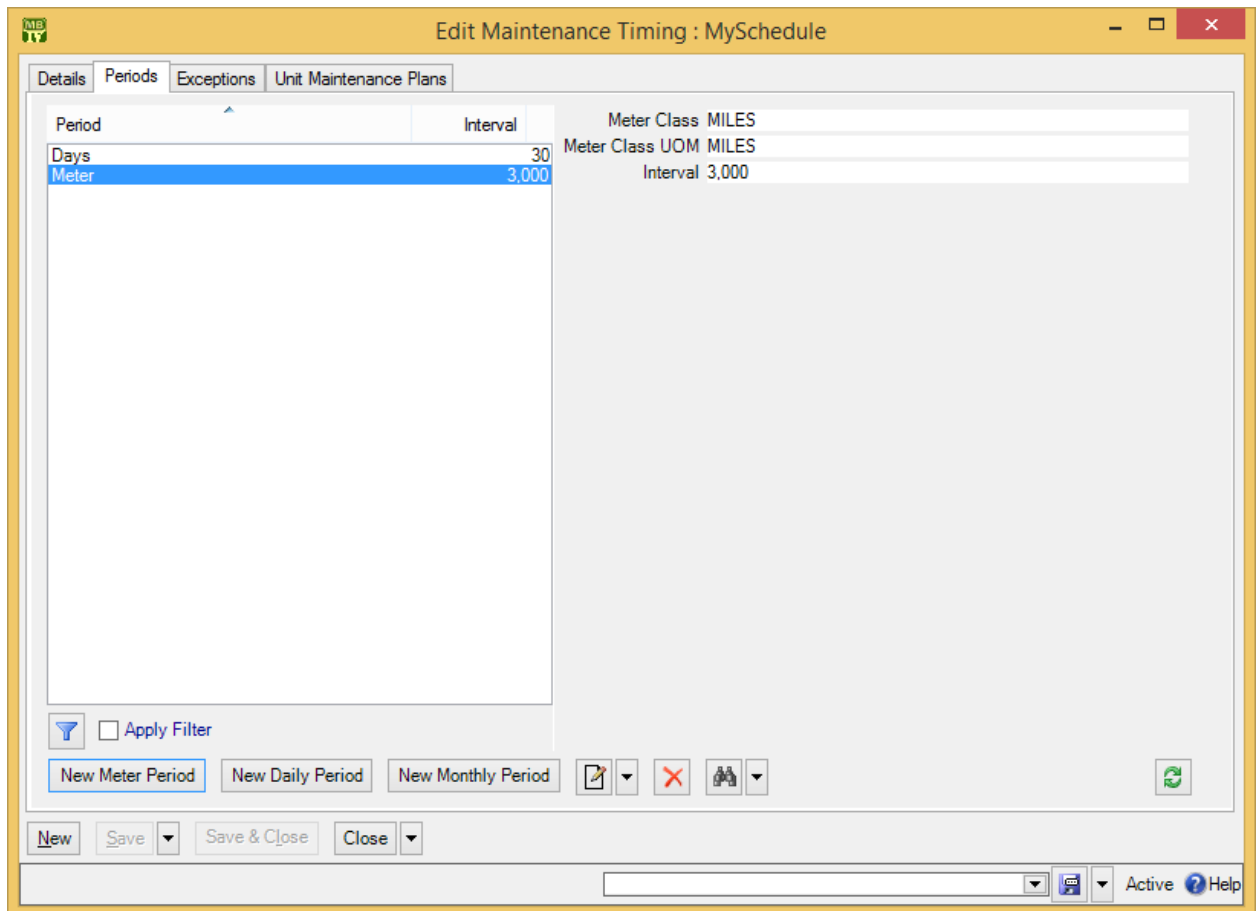
MainBoss will schedule the job whenever either of the possibilities is satisfied.

Tasks

A *task* record provides basic information on a maintenance job. Tasks often contain checklists or step-by-step instructions for what workers should do. Task descriptions might also contain useful information like any special tools or safety equipment that the worker will need. You can view task records through **Unit Maintenance Plans** | **Tasks** or **Coding Definitions** | **Unit Maintenance Plans** | **Tasks**.

In addition to a description of the work, task records may contain most of the information that can appear in a work order. For example, you can specify personnel assignments, demands for materials, access codes, temporary storage locations, and so on. When generating work orders, MainBoss copies all this information from the task record into the generated work orders.

Tasks can be used in two contexts:

☐ Tasks can be used to create work orders "on demand". For example, you might write up a task describing everything that should be done in a "muffler-change/brake-inspection" job. Once you have the task written up, you can use it for every such work order you ever need (changing details slightly if necessary).

In a busy maintenance department, you probably can't take the time to write up full instructions for every job you do, every time you do it. If you use tasks, however, you only have to make a full write-up once (when you create the task record). From that point on, you can quickly generate work orders from the task. Therefore, we recommend that you create a good complete task description for every standard job you do. You can then use the task to create good complete work orders whenever necessary.

For more on using tasks in this way, see *Standard (Boilerplate) Work Orders* on page 128.

☐ Tasks can also be used to create planned maintenance work orders. The rest of this chapter explains how this is done.

What a Task Record Should Contain: The contents of a task record are used to create subsequent work orders. Any essential information that is not in the task will have to be filled into the work order by hand.

For example, consider the question of who should be assigned to perform a particular job.

☐ If the same person is always assigned to do the job, you should record that assignment in the task record. MainBoss will then automatically assign the person to the job when work orders are generated. You don't have to fill in work order assignments yourself.

☐ If a job is frequently done by different people, you might specify an employee trade rather than a specific personnel assignment. Each time the task is used to generate a work order, you can review the generated work order and decide who'll do the work this time.

☐ If a job is usually done by the same person, but not always, you have to weigh your options. If you don't specify a personnel assignment in the task, then you'll *always* have to fill in the assignment later (once work orders have been generated). If you *do* specify a personnel assignment in the task, then sometimes you'll have to change the default assignment to someone else. This means running the risk that you'll forget to change the assignment, so the work order will be given to the wrong person.

Similar considerations apply to other fields in the task record.

> The basic principle of tasks is this: **Write up the job once, then use that write-up whenever you need it.** For this reason, we recommend that tasks provide details about a job, as completely as possible.

Task Specializations: A *task specialization* is a specialized version of another task. For example, suppose you have a standard `Oil Change` task which describes a standard oil change. However, the oil change for a particular vehicle might require a different type of oil, extra inspection operations, or other differences from the standard. In this case, you can create a task specialization based on the original `Oil Change` task. The specialization states ways in which the special job differs from the original. The following rules control what happens when a work order is generated from a task specialization:

☐ Information in the specialization is *added* to the original whenever possible. For example, the task description in a generated work order will consist of the description in the original task *plus* any description in the specialization. Similarly, the work order will contain all resources from the original *plus* all resources from the specialization.

☐ Information in the specialization *replaces* information from the original when the two can't coexist. For example, a work order can only have a single priority. Therefore, if the original task description has one priority and the specialization has a different one, the work order will take its priority setting from the specialization.

A task specialization can be used wherever a task can. For example, you can schedule a specialized task in the same way that you schedule a basic task. Similarly, you can create a specialization of a specialization.

In the **Tasks** viewer and other viewers, tasks and specializations are shown using a map similar to a location map. Specializations are shown under and indented from their base tasks.

Task Work Description

The most important part of a task record is its **Work Description** section. This is where you describe the planned maintenance work that should be done. This might include step-by-step instructions or a checklist. It might also specify tools, safety equipment, and anything else that might be of use to the person doing the work.

_____

☐ **To begin creating a new task:**

1. In the control panel, go to **Unit Maintenance Plans** | **Tasks**.

2. Click **New Task**. This opens a window where you can create the task record.

3. In "**Code**" ❶, enter an identification code for this task.

4. In "**Description**" ❷, enter a more complete description of the task.

5. In "**Subject**" ❸, enter a subject line for work orders based on this task.

6. In the **Work Description** ❹ section of the task record, enter complete information on the work to be done. You can use standard Windows editing techniques to fill in the section (e.g. copy and paste).

7. Click **Save** ❺ to save the record (but not close it).

Note: The "**Subject**" field is not mandatory in a task record, but it *is* required in work orders. Therefore, all your tasks should have "**Subject**" lines so that work orders can be created successfully. (If you use task specializations, either the main task or the specialization should have a "**Subject**" line. If both have "**Subject**" lines, MainBoss uses the "**Subject**" from the specialization.)

The **Details** section contains a number of other fields that may or may not be useful for a particular task:

Work Duration: The number of days that the task is expected to take.

This Task: For a task, this should specify the number of days that this task is expected to take. For a task specialization, this should specify the number of *additional* days this specialization is expected to take, over and above the task on which this specialization is based.

Basis Task: For a task, this is blank and read-only. For a task specialization, this is a read-only field giving the duration of the task on which this specialization is based.

Effective values: The total duration expected for this task, calculated by adding the values in "**This Task**" and "**Basis Task**".

Generate Lead Time: Is specified as a number of days. If you enter a number in this field, work orders using this task will be generated the given number of days earlier than they would be otherwise. For example, if the value is 7, work orders using this task will be generated a week earlier than normal. (The "**Work Start Date**" in the generated work order will tell when the job should actually start.)

By specifying a lead time, you create a delay between the time a work order is created and the time the job actually begins. This allows you time to make preparations. A lead time is particularly useful if you need to buy parts especially for a job. When MainBoss generates a work order, it immediately creates demands for all items required for the job; if some of these items aren't in stock, they'll show up on the **Restocking** report. In this way, you'll see ahead of time that you need to order the parts. If you set enough lead time, the parts will be in by the time you actually have to do the work.

This Task: For a task, this should specify the number of days of lead time this task needs. For a task specialization, this should specify the number

of *additional* days of lead time this specialization needs, over and above the task on which this specialization is based.

Basis Task: For a task, this is blank and read-only. For a task specialization, this is a read-only field giving the lead time of the task on which this specialization is based.

Effective Values: The total lead time expected for this task, calculated by adding the values in "**This Task**" and "**Basis Task**".

Work Category: The type of work involved in this task.

Access Code: An access code to be associated with work orders generated from this task. For more on access codes, see *Access Codes* on page 66.

Priority: A priority to be associated with work orders generated from this task.

Expense Model: The expense model to be associated with work orders generated from this task. For more on expense models, see *Accounting Facilities* on page 37.

Project: A project to be associated with this task (if any).

Downtime: A default downtime—how long the unit is expected to be down during the maintenance.

Close Code: A default closing code to be assigned to work orders generated from this task.

Closing Comment: Any comments that should be filled into the **Comments** section of generated work orders.

Comments: Comments about this task record. These comments will *not* appear on generated work orders.

For more on access codes, see *Access Codes* on page 66. For more on projects, work categories and work order priorities, see *Other Work Order Information* on page 98. For more on expense models, see *Expense Models* on page 101.

Assigning Personnel to a Task

As noted in an earlier section, you can assign workers to a task. Whenever you generate work orders based on the task, the same personnel will be assigned to the work orders.

Before you can assign personnel to a task, you must create appropriate labor records: *hourly inside* for employees charging by the hour, *per job inside* for employees charged by the job, *hourly outside* for outside contractors charging by the hour, and *per job outside* for outside contractors charging by the job. For more on any of these, see *Labor Costs* on page 90.

Note: Hourly inside records can refer to a specific worker or they can refer to a general trade (e.g. "Electrician First Class"). If you don't want to assign a task to the same person each time, you can use an hourly inside record that just specifies a trade.

☐ **To assign hourly-paid employees to a task:**

1. In the **Resources** section of a task record, go to the **Inside** section of the window and click **New Task Demand Hourly Inside**. This opens a window where you can specify the work assignment.

2. In "**Hourly Inside**" ❶, specify an hourly inside record giving the name or trade of the worker whom you want to do the task. (MainBoss will fill in "**Hourly Rate**" with the worker's hourly rate, as given by the selected hourly inside record.)

3. In "**Demanded**" ❷, enter the length of time that you expect the job will take.

4. In "**Actual Cost Default**" ❸, choose the default for the way that actual costs will be associated with this demand. **Manual Entry** means that the person who records actual costs should actually type in the correct costs. **Current value calculation** means that the actual costs should be taken from the hourly inside record. **Demanded** means that actual costs should be taken from the amount specified in the demand.

5. Click **Save & Close** ❹ to save the information and close the window.

If you choose **Estimate total demanded cost when work order created**, MainBoss will automatically fill in cost estimates whenever it creates a work order. These estimates will be based on the hourly rate specified in the "**Hourly Inside**" record. If you choose **Leave total demanded cost blank when work order is created**, the generated work orders will not have cost estimates.

To assign an inside worker who gets paid by the job, follow the same sort of procedure but click the arrow beside **New Task Demand Hourly Inside** and click **New Task Demand Per Job Inside**.

A similar process can be used for outside contractors. Go to the task's **Resources** section, open the **Outside** section and click **New Task Demand Hourly Outside** or **New Task Demand Per Job Outside**. Each one opens a window where you can assign a contractor . For more details, see the online help.

Temporary Storage for Tasks

During the course of planned maintenance work, you may need temporary storage for tools, spare parts, and other materials used in the work. Here are some situations where this is useful:

☐ If a job takes several days, you may need someplace to store materials overnight. This is particularly important if different workers participate on different days. Workers on Monday need to store materials in a known location so that workers on Tuesday can find what they need.

☐ Large organizations sometimes have one worker in charge of delivering materials to a work site while other workers do the actual job. The delivery person needs to know where to put the materials and the workers need to know where the materials can be found.

For these reasons, MainBoss makes it possible to associate temporary storage locations with planned maintenance tasks. (You can also associate temporary storage locations with specific work orders—see the Day-to-Day Operations guide for details.) In order for MainBoss to use a location as temporary storage in a task, you must set up an appropriate record.
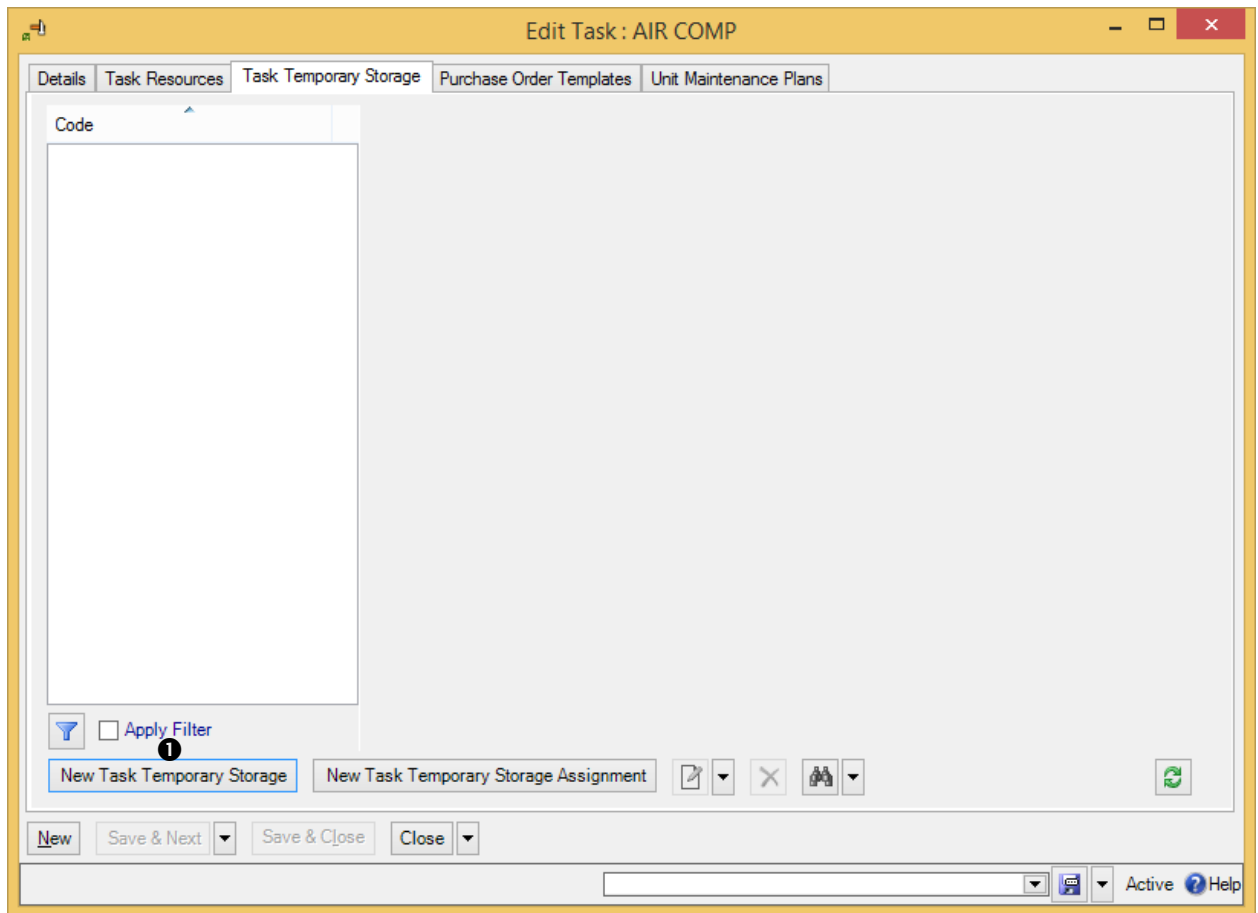
☐ **To designate that a location can be used for temporary storage in a task:**

1. In the task record, in the **Task Temporary Storage** section, click **New Task Temporary Storage** ❶. MainBoss opens a window where you can specify the storage location:
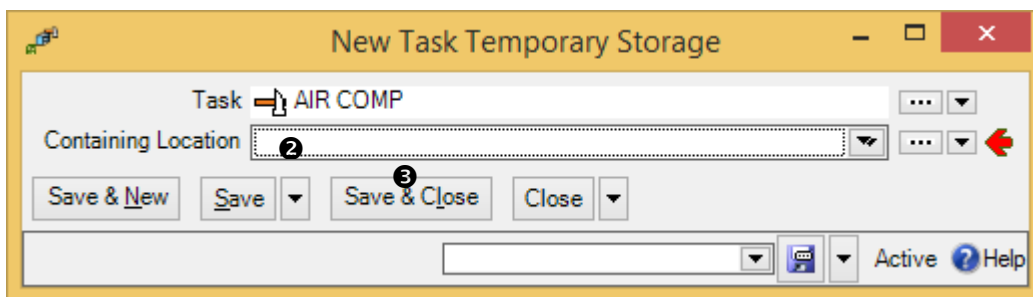
2. In "**Containing Location**" ❷, specify the location you want to use as the temporary storage location.

3. Click **Save & Close** ❸ to save the record. When you return to the task record, you will see the storage location listed.

Once you have specified a temporary storage location, you can assign items to be stored in that location.
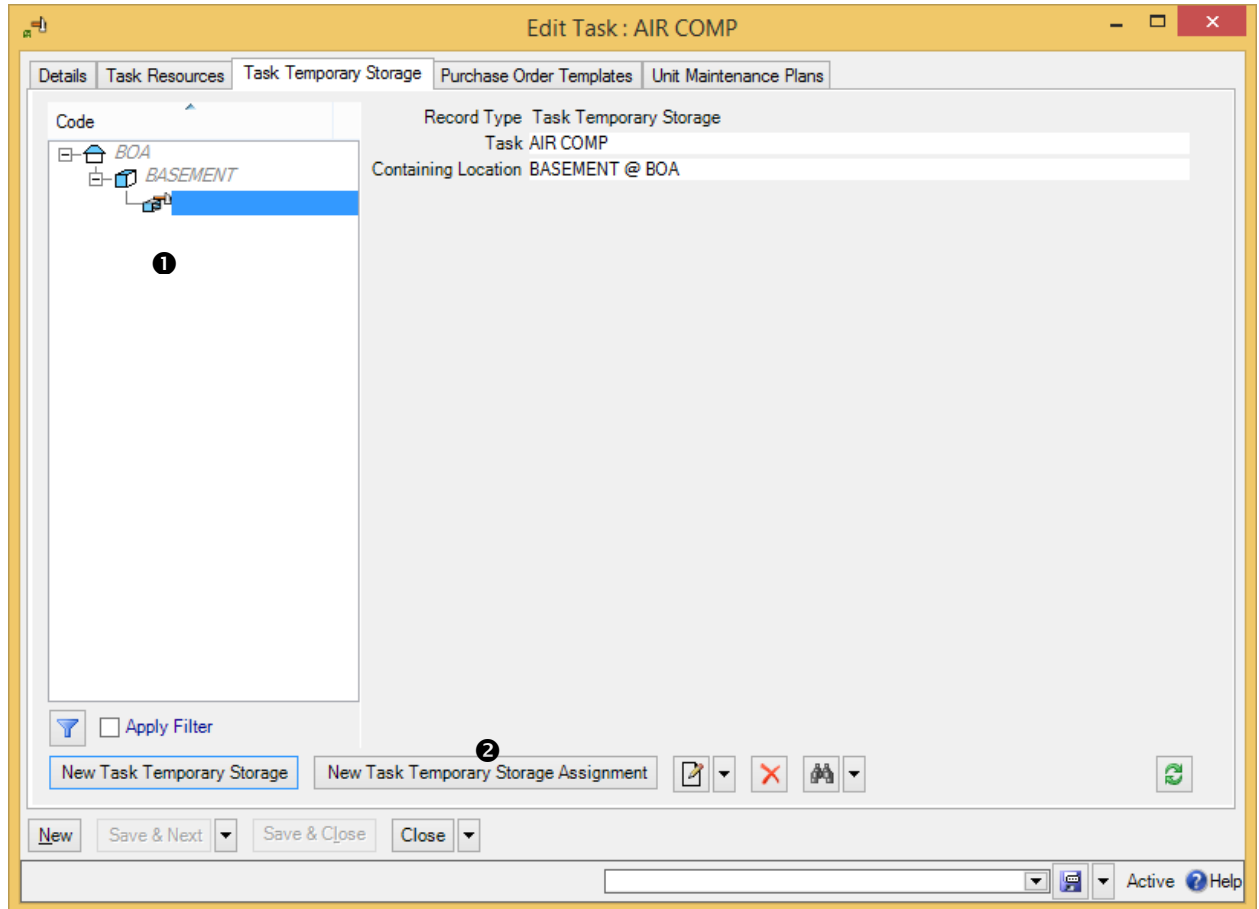
☐ **To indicate that items should be stored in a temporary location during a task:**

1. In the task record, in the **Task Temporary Storage** section, click the temporary storage location ❶ where you want to store the item.

2. Click **New Task Temporary Storage Assignment** ❷. MainBoss opens a window where you can assign an item to the temporary storage location:
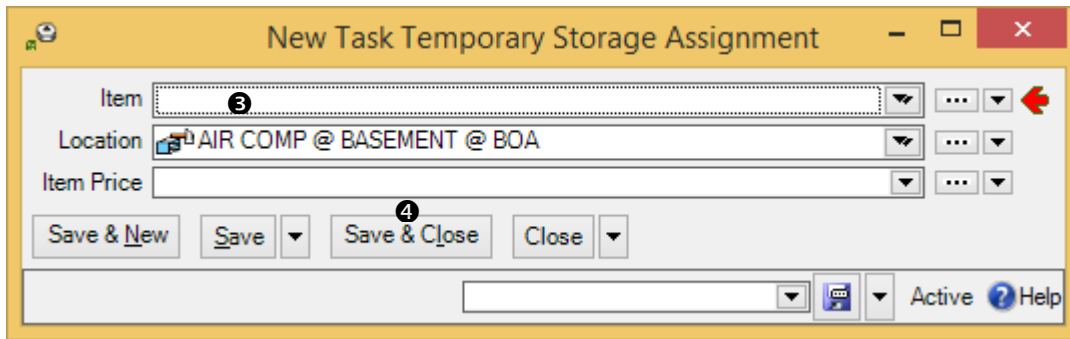
3. In "**Item**" ❸, specify an item you wish to store in the location.

4. Click **Save & Close** ❹ to save the record. When you return to the previous window, you will see the assignment in the task record's list.

(The "**Preferred Pricing**" field is optional and only needed if this item has a special price during this task.)

Other Task Record Sections

Here's a quick summary of other information in a task record:

Resources section: You can use this section to reserve spare parts and other materials for use on a job. However, you can't do that until you've set up information about your materials inventory. For more on inventory, see *Inventory Control* on page 129. (You can also use the **Resources** section to record miscellaneous expenses, e.g. the cost of renting special equipment that the job might need.)

Purchase Templates section: Just as a task provides the basis for creating a work order, a purchase template provides the basis for creating a purchase order. If a task has an associated purchase template, whenever MainBoss creates a work order from the task, MainBoss will also create a purchase order from the template.

For example, if a task will be done by an outside contractor, you might want to create a purchase order that will pay for the contractor's services. You can associate a purchase template with the task so that MainBoss will create an appropriate purchase order every time it creates a work order from the task.

For more on purchase templates, see *Purchase Order Templates* on page 146

Unit Maintenance Plans section: Lists any unit maintenance plans that use this task. (For more, see *Unit Maintenance Plans* on page 123.)

Unit Maintenance Plans

As noted previously, a task record describes a planned maintenance job. A timing record describes a schedule (such as "every three months"). In order to set up a planned maintenance schedule, you put the two together along with a unit where you want such work to be done.

The **Unit Maintenance Plans** list is the heart of your planned maintenance programs. It puts together what you want to do (the tasks), when you want to do it (the timing), and where the work should be done (the units).

□ **To set up a unit maintenance plan:**

1. Go to **Unit Maintenance Plans**.

2. Click **New Unit Maintenance Plan**. This opens the following window:

3. In "**Task**" ❶, specify a task record (e.g. one describing an oil change).

4. In "**Unit**" ❷, specify a unit to be maintained on the given schedule (e.g. a car whose oil should be changed every three months).

5. In "**Maintenance Timing**" ❸, specify a maintenance timing record (e.g. every three months).

6. Select one of the "**Timing Basis**" ❹ options (discussed below).

7. Click **Save & Close** ❺ to save your information and close the window.

"**Timing Basis**" options let you tell when new jobs should be scheduled after old ones. For example, if the schedule says every three months, the question is, "Three months after what?" Possibilities include:

Work Start: The timing for the next job is based on the actual start of the previous job (e.g. three months after the previous job's actual starting date).

Work End: The timing for the next job is based on the actual end of the previous job.

Scheduled Date: The timing for the next job is based on when the previous one was scheduled, whether or not the previous one actually took place on that date.

For example, suppose a job is scheduled for January 1 but actually takes place on January 4. Also suppose that the job is supposed to take place every three months.

☐ If you set "**Timing Basis**" to **Work Start**, the next job is scheduled for April 4: three months after the job really took place.

☐ If you set "**Timing Basis**" to **Scheduled Date**, the next job is scheduled for April 1: three months after the job was supposed to take place.

> Using **Scheduled Date** is often preferable to **Work Start** or **Work End**. If you use **Work Start** or **Work End**, jobs can migrate later and later—you might start with a balanced work schedule, but as jobs migrate, you can end up with big clusters of tasks all due at once.

Specifying the Scheduling Basis

Once you've created a unit maintenance plan, there's one last step before you can begin your planned maintenance program: you have to start the clock ticking.

For example, suppose a plan says a particular vehicle should receive an oil change every three months. You have to tell MainBoss when that vehicle got its most recent previous oil change. This gives MainBoss a starting point for scheduling future oil changes.

> **For each unit maintenance plan, you must set a scheduling basis. This is the date or meter value when the timing schedule should start. Without a scheduling basis, MainBoss can't start a maintenance schedule for the given unit.**

☐ **To start the scheduling basis for a unit:**

1. Go to **Unit Maintenance Plans**.

2. Click on the plan whose scheduling basis you want to set.

3. Click **Change Scheduling Basis**. This opens the following window:
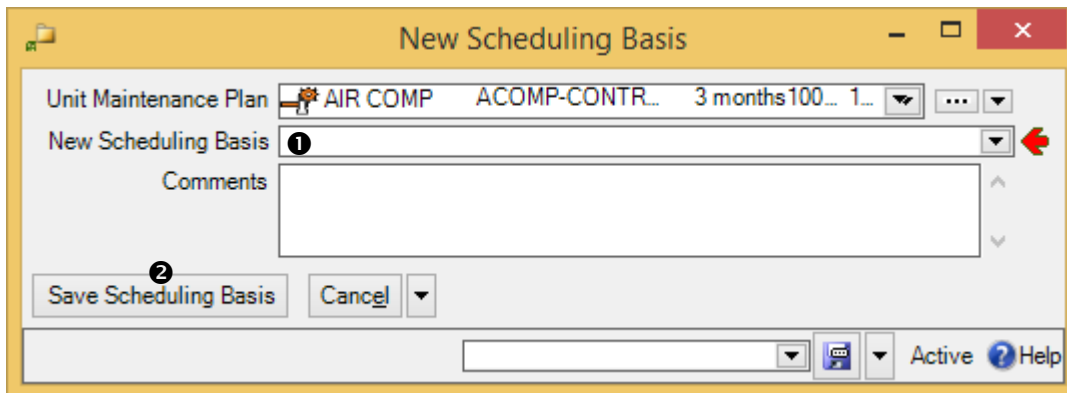
4. In "**Save Scheduling Basis**" ❶, specify a date or meter reading.

5. Click **Save & Close** ❷ to save your information and close the window.

> **Reminder:** You must set a scheduling basis for every unit maintenance plan.

Generating Planned Work Orders

At this point, you've finished almost all initial set-up for your planned maintenance program. The actual process of generating planned maintenance work orders is covered in the Day-to-Day Operations guide .

However, there is one last detail remaining in the set-up process: the *generation interval* for your planned maintenance work orders. This interval reflects how often you intend to generate such work orders.

How the Generation Interval Works: When you ask MainBoss to generate work orders, you ask for all work orders up to a particular date. For example, suppose that each Friday you ask for all work orders that should be done in the following work-week (from Monday to Sunday). This means that each time you generate work orders, you ask for all jobs up to and including the appropriate Sunday.

So suppose you've previously generated work orders up to Sunday, May 1. The next time you prepare to generate work orders, MainBoss automatically adds the generation interval onto the previous date. If your generation interval is 7, MainBoss would prepare to generate all work orders up to Sunday, May 8. Similarly, if your generation interval is 14, MainBoss would prepare to generate all work orders up to May 15.

Whatever your generation interval is, you can always ask MainBoss to prepare work orders up to a different date. The generation interval simply makes it easier to keep a regular schedule.

Most organizations generate PM work orders once a week; this is the default when you first install MainBoss. However, you may prefer a different approach, e.g. once every two weeks. In this case, you should change the generation interval value to reflect your preferences.

☐ **To change the generation interval:**

1. In the control panel, go to **Unit Maintenance Plans** | **Generate Planned Maintenance**.

2. Go to the **Defaults for Planned Maintenance Batch** section.

3. Click **Edit Defaults**. MainBoss opens a window where you can set PM generation defaults.
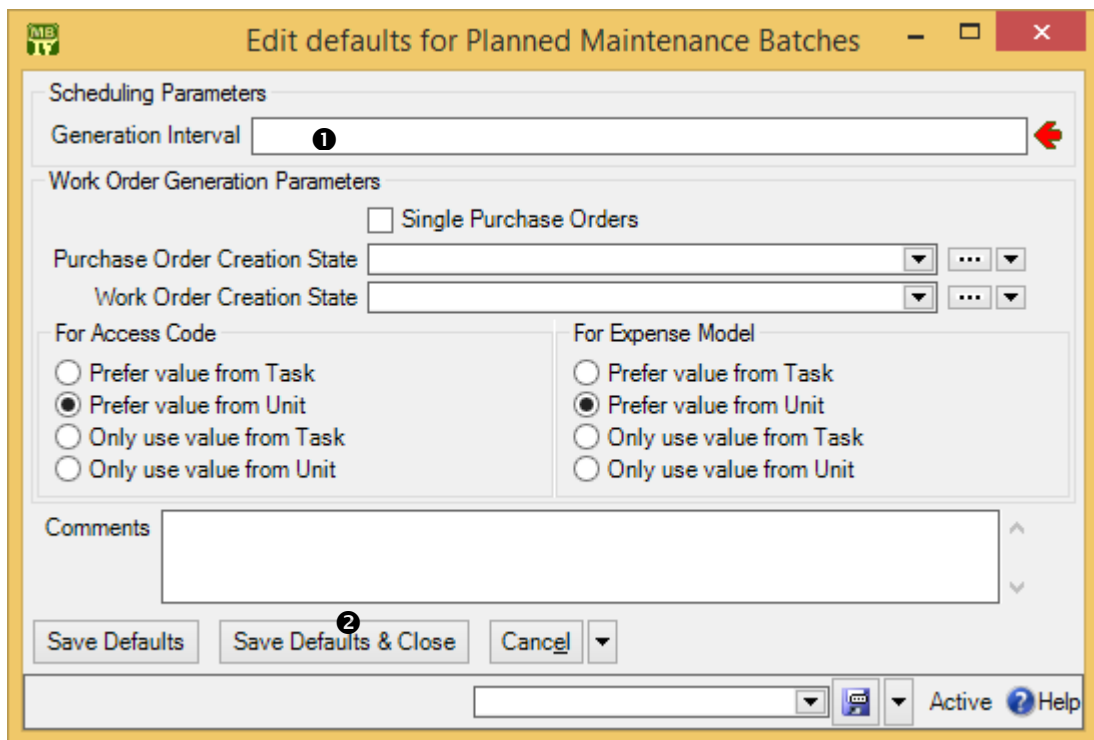
4. In "**Generation Interval**" ❶, enter the number of days you expect to pass between generating PM work orders. For example, if you plan to generate work orders every two weeks, enter 14.

5. Click **Save Defaults & Close** ❷ to save your information and close the window.

Standard (Boilerplate) Work Orders

As noted earlier in this chapter, tasks can be used to create work orders on demand. For example, a property management company might create one or more tasks describing what should be done after a tenant moves out or before a new tenant moves in. Similarly, a company with a number of vehicles to maintain might have standard procedures for what should be done when changing a muffler or brake pads.

Once you've written up the task, you can create a work order at any time using the task as a template: just go to **Unit Maintenance Plans** | **Tasks** and click **New Work Order From Task**. This opens a window where you can specify the unit on which you want to perform the task. For example, if you want to perform a muffler change on a car, find the "muffler change" task and click **New Work Order From Task**. In the resulting window, specify the car whose muffler you want to change. When you **Save & Close** the window, MainBoss will generate a muffler-change work order for that car.

You can also create work orders directly from a unit maintenance plan record. In this case, you go to **Unit Maintenance Plans**, click on the appropriate plan, and then click **Create unplanned Maintenance Work Order**. In the resulting window, click **Commit** to create the work order.

Because it's so convenient to generate work orders in this way, you might choose to create a number of unit maintenance plans specifically designed for this type of use. For example, consider the job of changing a muffler on a car. You could create a "change muffler" task and use this to create unit maintenance plan records: one for each car in your company fleet. Whenever you want to perform a muffler change on a particular car, find the "muffler change" unit maintenance plan for that car and use **Create unplanned Maintenance Work Order** to create an appropriate work order.

You might ask, "What kind of maintenance timing record would such a unit maintenance plan have?" A muffler change isn't usually done on a predictable schedule. For that reason, you could create a special maintenance timing record that stands for "no schedule". To create such a maintenance timing record, simply create one with nothing in the **Periods** section. Such a schedule will never come due. Instead, you just use the unit maintenance plan "manually" through **Create unplanned Maintenance Work Order**.

Inventory Control

MainBoss can keeps track of the materials you use in your work. This can include spare parts, lubricants and anything else whose use you want to record. In MainBoss, such materials are called *items* or *inventory items*.

The **MainBoss** module lets you track the inventory you use on work orders. This means you can create records describing inventory items, including price information and the quantities used on jobs.

The **Inventory** module adds the ability to keep track of where items are stored. In particular, you can describe which items each storeroom contains, as well as the maximum and minimum quantities of each item that you want to keep in each storeroom.

When setting up inventory records, start with the most important items in your inventory, such as expensive or vital spare parts. Tracking simple items like nails or 60-watt light bulbs is often more trouble than it's worth...but if your operations depend on certain materials, those are the ones you should track.

Note that MainBoss's records depend on physical inventory control. If anyone can go into a storeroom and take out materials without recording the fact in MainBoss, then you can't expect MainBoss's data to be accurate.

Item Categories

Coding Definitions | **Items** | **Item Categories** lets you define categories for organizing your inventory items. The procedure is the same as creating other categories (e.g. unit categories).

As with other types of categories, item categories make it easier to analyze your use of materials. For example, you might be able to answer questions like, "How much have we spent this year on light bulbs?" or "What's the total value of the spare parts we currently have in inventory?" (You answer these questions by specifying appropriate item category filters in the **Filters** section of reports.)

Issue and Adjustment Codes

Anytime you disburse inventory items, the action is called an *issue*. The most common issue action is using materials as part of a work order. However, you may issue items for other reasons. For example, a property management company may give or sell inventory materials to tenants. As another example, you may sell items that you no longer need (e.g. spare parts for equipment that has been taken out of service).

To deal with issue actions, you can define *issue codes* to classify your reasons for particular issues. Here are some typical ones:

```
Given to tenant
Transferred to other department
Sent back to vendor
Sold as surplus
```

You define issue codes using **Coding Definitions** | **Items** | **Item Issue Codes**. The process is similar to defining other types of codes (e.g. item categories).

Adjustment codes are similar to issue codes: they let you classify adjustments that you have to make to your inventory. Adjustment codes can be defined for any kind of change to your inventory, such as losses due to breakage or theft, and transfers from one storeroom to another.

One of the most important types of adjustment is Physical count: when you take physical inventory and discover that the actual quantity of an item is different from the expected quantity.

You define adjustment codes using **Coding Definitions** | **Items** | **Item Adjustment Codes**. The process is similar to defining issue codes.

> An "issue" operation always removes items from your inventory. An "adjustment" operation may remove items or add them.

Item Records

In order to use MainBoss's inventory control facilities, you have to create a list of the inventory items you wish to track. Many companies phase in this process: they begin by keeping track of a few spare parts that are crucial to their operations, then slowly expand to include more materials.

Items have an associated unit of measurement. For example, if you keep track of a particular lubricant, you might record that it is measured in quarts. Of course, many kinds of items are individual objects (e.g. light bulbs) and it doesn't make sense to measure them in any particular units. In this case, it's useful for your **Units of Measure** table to have a special unit that you can specify for such objects. You can give this special unit any name you like, but earlier versions of MainBoss used the name EACH as the unit for such items. (This comes from statements like, "Oil is a five dollars a quart and light bulbs are a dollar each.")

We recommend that the "**Description**" field of an inventory item contain whatever information is needed for an un-knowledgeable person to purchase such an item. For example, it might contain information on model number, size, and anything else necessary for people to buy the right thing, even if they don't know anything about what your maintenance department does.

On the other hand, there are times when you might want to *omit* information from the "**Description**" field. For example, consider car tires. A "**Description**" field might include the tire's width and diameter, but not the manufacturer—this indicates that you don't care whether the tire comes from Goodyear, Goodrich, Michelin, etc. as long as it's the right size. In other words, when you expect to get the same part from multiple manufacturers, don't put the manufacturer's name in the item's "**Description**" (or "**Code**").

☐ **To create an item record:**

1. In the control panel, go to **Items** or **Coding Definitions** | **Items**.

2. Click **New Item**. MainBoss opens a window where you can record item information.

3. In "**Code**" ❶, enter a brief code to identify the item.

4. In "**Description**" ❷, enter a more complete description of the item.

5. In "**Category**" ❸, specify an item category for the item (optional).

6. In "**UOM**" ❹, specify the units of measurement for the item.

7. Click **Save & Close** ❺ to save your information and close the window.

When you first set up an item record, you only fill in the **Item Properties** section, as just described. Other sections will be completed in later stages of set-up or in day-to-day operations.

Parts

Once you've created item records, you can record which items are used as spare parts in various units. This is useful information for workers going out to do work on a unit: they can pick up spare parts ahead of time so that when they arrive on the job, they already have the parts they might need.

The information is also useful if an important unit breaks down and you don't have the appropriate spare part in inventory. If you find out that other units use the same part, you may be able to take the part from a less important unit in order to repair the more important one.

The definition of *spare parts* can be as flexible as you want it to be. For example, if a unit is a room, the spare parts can be the type of light bulbs used in the room's lighting. This is useful information for workers—if someone reports a burnt-out light, the worker knows what kind of replacement bulb is needed.

Similarly, spare parts might include the type of lubricant used for a particular piece of equipment; again, this is useful information for workers to have before they go to handle a problem. In this example, it's also useful to record the quantity of lubricant that the unit might require, so that workers will take enough to do the job.

☐ **To record the spare parts associated with a unit:**

1. In the control panel, go to **Units**.

2. In the **View** section, select the unit whose spare parts you wish to record.

3. Click **Edit** to edit the unit's record. This opens a window where you can record unit information.
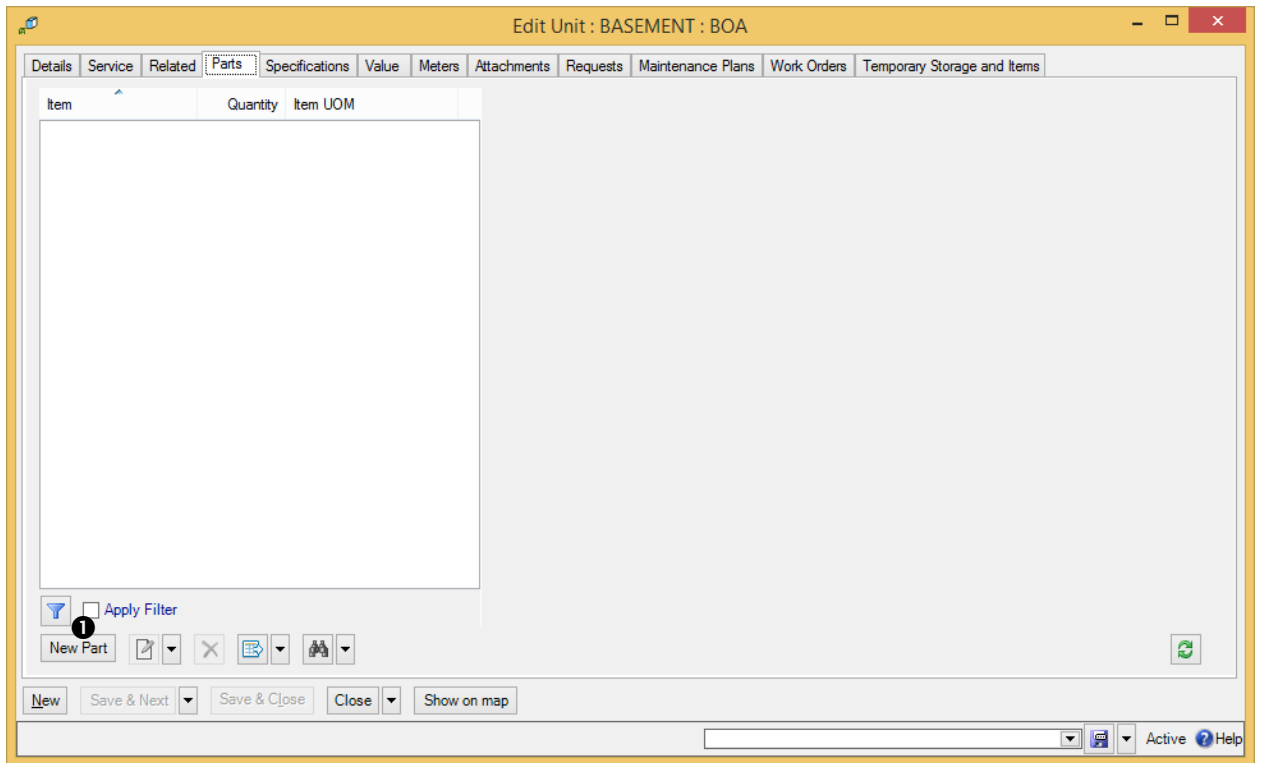
4. In the **Parts** section of the unit record, click **New Part** ❶. This opens a window where you can record a spare part used in connection with the unit.

5. In "**Item**" ❷, specify an item that serves as a spare part for this unit.

6. In "**Quantity**" ❸, enter the number of the quantity of the item that the unit requires.

7. Click **Save & Close** ❹ to save the information you've entered and close the window.

Note that the "**Comments**" field in a spare parts record is useful for situations where the part your maintenance department uses is not the same as the part named in the unit's repair manual. For example, consider headlight bulbs for a car. The car's manual may say, "Use bulb #A123." However, your department buys headlight bulbs from a different manufacturer, or possibly from multiple manufacturers. You can use the "**Comments**" field to say, "The manual calls for bulb #A123, but we use #B456."

In the same way, you can use the "**Comments**" field if the manufacturer has renumbered its parts since the repair manual was printed, or if there is any other type of discrepancy between the unit's manual and the parts that your workers will actually use.

Item Pricing Records

An *item pricing* record describes the price of a quantity of a particular item, as purchased from a specific vendor. Item pricing records can be created from the **Pricing** section of an item record or through the "**Preferred Pricing**" field of a storeroom assignment record.

When you first set up your inventory records, you may not know the original price of some materials, or even the original vendor. It may still be useful to estimate prices so you can get an approximate idea of how much your inventory is worth. As for the vendor, you can create a special entry `Unknown vendor` in your **Vendors** table, for use when you can't remember who sold you the items originally.

Once you get MainBoss set up, you'll begin to record real prices and price quotes during your day-to-day operations. Any estimates you start out with will gradually be superseded with more solid figures.

☐ **To record pricing information for an item:**

1. In the control panel, go to **Items**.

2. In the **View** section of the **Items** table, select (click) the item whose price you want to record.

3. Click **Edit**. MainBoss opens the item record for the selected item.
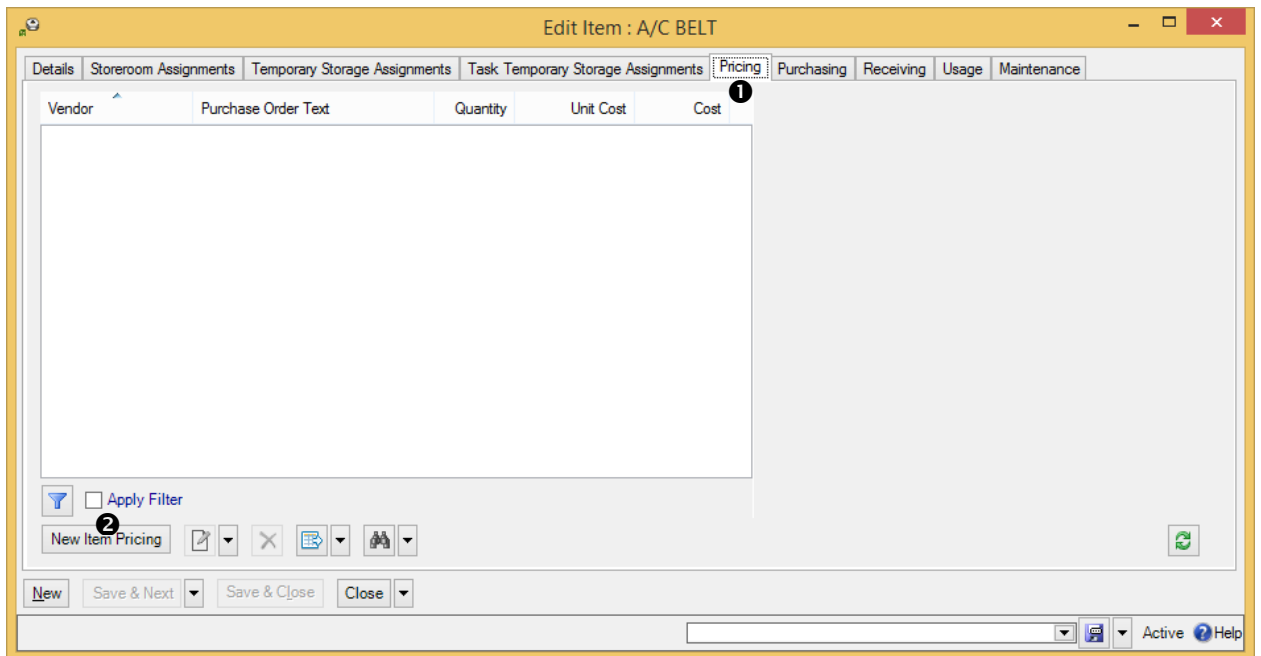
4. Go to the **Pricing** section ❶ and click **New Item Pricing** ❷. MainBoss opens a window where you can record pricing information.

5. In "**Vendor**" ❸, specify the supplier from whom you bought the item or who supplied the price quote.

6. In "**Quantity**" ❹, enter a number indicating the quantity of the item that you purchased from the vendor or that was quoted in the price quote.

7. In "**Cost**" ❺, enter the total cost for the specified quantity.

8. In "**Purchase Order Text**" ❻, enter the text that should appear on purchase orders if you order the specified item from "**Vendor**". Typically, this is the name that the vendor uses for the item, including any catalog number that the vendor might use.

9. Click **Save & Close** ❼ to save your information and close the window.

The above instructions suggest that you should fill in "**Cost**" with the total cost. Alternatively, you can fill in "**Unit Cost**" instead. In all MainBoss records, the following principle holds:
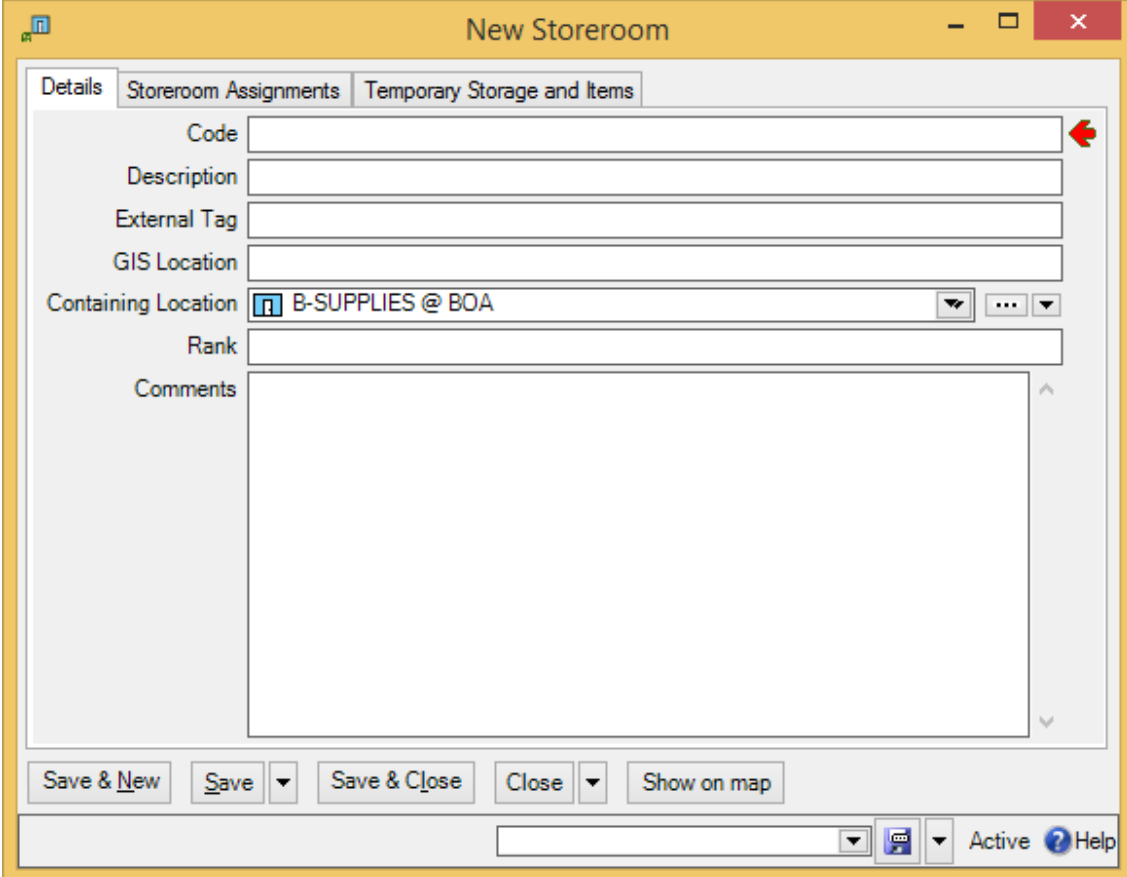
☐ If you fill in a total cost, MainBoss automatically fills in the associated unit cost if it can be calculated. (In an item pricing record, "**Unit Cost**" is "**Cost**" divided by "**Quantity**".)

☐ If you fill in a unit cost, MainBoss automatically fills in the associated total cost if it can be calculated. (In an item pricing record, "**Cost**" is "**Unit Cost**" multiplied by "**Quantity**".)

Storerooms

If you have the **Inventory** module, the next step in working with inventory is to record your *storerooms*: the locations where materials can be stored. By creating a list of storerooms, you make it possible to specify the location of materials: "Air filters are in Room 101."

If you do not have the **Inventory** module, you will not be able to record information about storerooms.

You specify storerooms using **Coding Definitions** | **Items** | **Storerooms**. Creating a **Storerooms** record indicates that an existing unit or location is a storeroom. For example, suppose Room 101 is a storeroom. Create a **Units** or **Locations** table entry for the room, then use **Coding Definitions** | **Items** | **Storerooms** to indicate that the room is a storeroom. Clicking **New Storeroom** opens a window where you can specify that a unit or location is a storeroom:

As shown above, storeroom records have a field called "**Rank**". This is used when you create a *physical count sheet*: a print-out used in taking physical inventory. (You can create such a sheet with **Items** | **Reports** | **Location and Status**.) Typically, such a sheet lists the items that are supposed to be in each storeroom. The sheet has blank space beside each entry where you can record how many such items are actually on hand.

The "**Rank**" value is a number that controls the order in which storerooms are listed on the count sheet: storerooms are listed in order from lowest order number to highest.

> **Note:** Since any unit can be a storeroom, you can declare a vehicle to be a storeroom. You might do this to deal with materials which are commonly stored in service vehicles owned by the maintenance department.

Storeroom Assignments

A *storeroom assignment* record deals with a particular type of item in a particular storeroom. For example, suppose you keep 1000-watt light bulbs in several different storerooms; you'd have a separate storeroom assignment record for each storeroom that contains such light bulbs.

Storeroom assignments can be created from an item record or from **Coding Definitions** | **Items** | **Storeroom Assignments**. If you click **New Storeroom Assignment**, MainBoss opens the following window:

"**Item**" specifies an item and "**Location**" specifies a storeroom where the item may be stored; this must be a room from the **Storerooms** table. Other fields in the record include:

External Tag: A short string of characters which can be turned into a bar code associated with this storeroom assignment. For more information, see *Bar Code Support* on page 42.

Item Price: Price information for the item. This may give the actual price paid for an item, or simply a price quote. It also gives the preferred vendor. For more information, see *Item Pricing Records* on page 135.

Minimum: The minimum amount of this item that you want to keep on hand at this location. For example, suppose that a particular storeroom should always contain at least five 1000-watt light bulbs. Then the "**Minimum**" value is 5.

Maximum: The maximum amount of this item that you want to keep on hand in this location.

Cost Center: The cost center associated with storing and managing this item in this storeroom.

Storeroom assignments describe where an item may *possibly* be stored. When creating a storeroom assignment record, you don't specify whether any quantity of the item is *currently* stored in that location. In order to record how much a storeroom actually contains, you have to create a *physical count* record, as explained in *Physical Count Records* on page 139.

Physical Count Records

The last step in setting up for using MainBoss inventory control is specifying *physical counts* for all your items. This means recording the quantity of each item in each storeroom.

There are two times when you record physical counts: when you first set up MainBoss's inventory records, and whenever you take inventory in the storeroom. In both cases, you use the procedure described below.

☐ **To record a physical count for an item**

1. In the control panel, go to **Coding Definitions** | **Items** | **Storeroom Assignments**.

2. In the **View** section of the table, select (click) the record for the storeroom and item whose count you want to set.

3. Click **Edit**. MainBoss opens the storage assignment record. Go to the **Item Activity** section:
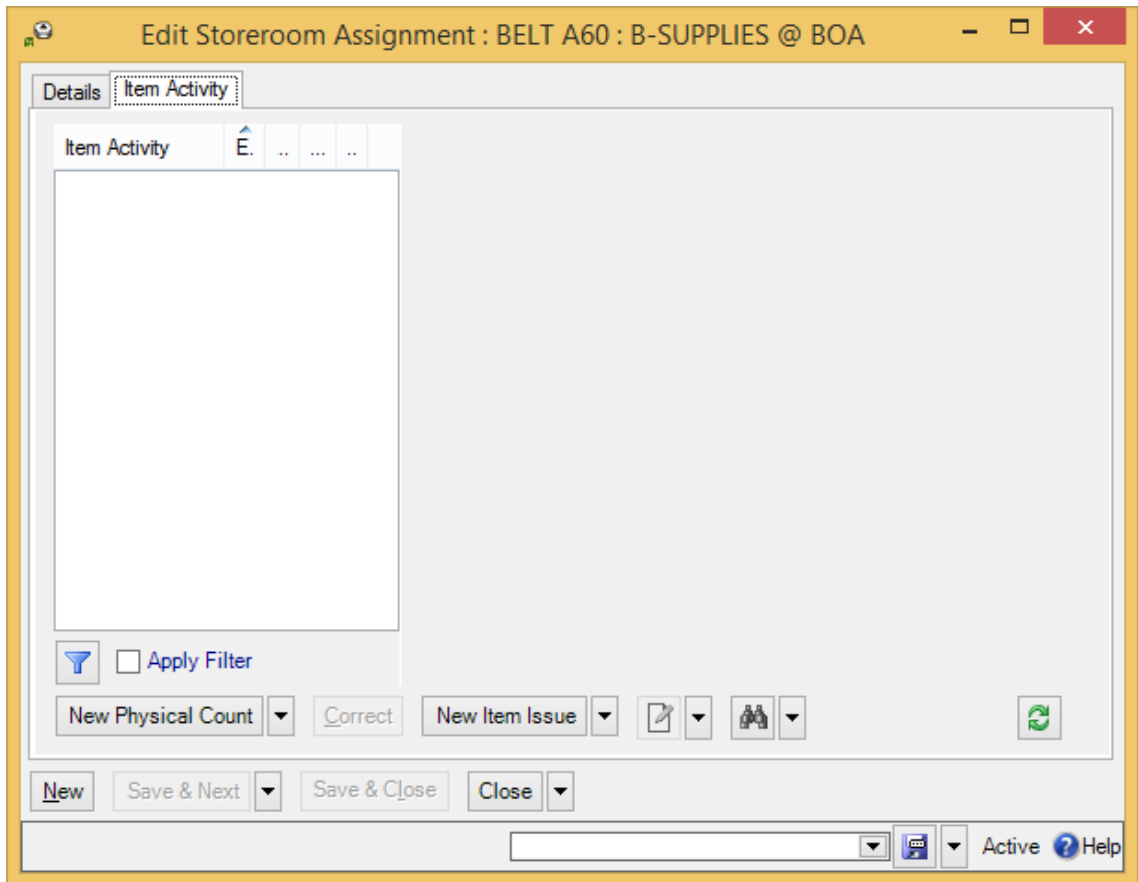
4. Click **New Physical Count** ❶. MainBoss opens a window where you can record the physical count for this item in this storeroom:

5. In "**Adjustment Code**" ❷, specify the code for taking a physical count (as discussed in *Issue and Adjustment Codes* on page 129).

6. In "**Quantity**" ❸, enter a number indicating the quantity of the item that you purchased from the vendor or that was quoted in the price quote.

7. If you already have pricing information for this item, MainBoss displays it ❹. If you want to use this pricing information as is, checkmark **Use calculated item price cost** ❺.

8. Otherwise, fill in either "**Unit Cost**" or "**Total Cost**" (total cost) ❻ in the "**Cost**" line. (If you don't know the cost, make your best estimate or simply set the cost to zero.)

9. Click **Save & Close** ❼ to save your information and close the window.

Critical Parts

Many organizations have a set of materials (especially spare parts) that are considered *critical*. Such parts are used to maintain units that are especially important to your operations. For example, if your company will be severely hurt by a breakdown in a production line, the parts used to maintain that line may be considered *critical parts*.

Obviously, you want to make sure that you always have an appropriate supply of critical parts on hand. This allows you to make repairs as soon as possible, rather than putting your operations on hold while you're waiting for parts to be delivered. You therefore need a way to indicate which items are critical, thereby distinguishing them from materials that are less important.

You could do this by creating an item category named `CRITICAL`, then assigning this category to all appropriate parts. If you do this, however, you can't specify the sort of information that the item category usually provides. For example, suppose that a particular type of belt is critical; all your other belts are put into a category called `BELTS`, but the critical belt must be put into the `CRITICAL` category. This means that reports on `BELTS` don't include the critical belt...which can be confusing.

A better approach might be to create a special critical storeroom inside each real storeroom. For example, suppose Room 101 is a storeroom which contains some critical parts. You might create a new storeroom called `CRITICAL: ROOM 101` which is located inside the main Room 101 (which means that the "**Location**" field in the critical storeroom record indicates that the critical storeroom is inside the main Room 101 storeroom).

Physically, the critical parts are simply stored in their usual place in Room 101. However, putting them in the special "sub-storeroom" makes them easy to identify in tables and in reports.

Purchasing/Receiving

This chapter is only relevant if you have licensed the **Purchasing** module. If you do not have a license key for this module, the control panel will not contain entries related to purchase orders.

The **Purchasing** module lets you create purchase orders and shipment receipts to track the flow of inventory materials. It also lets you track the labor costs of outside contractors. Whether or not your organization uses this module to produce "official" purchase orders, MainBoss's Purchasing/Receiving facilities are a good way to monitor your inventory purchases.

MainBoss as a Purchase Request System

Many organizations already have purchasing systems, often built into a general accounting system. For this reason, organizations may think that MainBoss's purchasing capabilities won't be useful.

However, the purchasing facilities in MainBoss can still be helpful when viewed as a purchase *request* system. You can use MainBoss to determine information such as:

☐ The items you should restock

☐ The best vendor for a particular item

☐ Expected receipt dates

☐ Actual receipt dates

☐ Where received items should be sent and stored

☐ Connections between purchase orders and work orders: this tells you when an item was ordered for a specific job, rather than just for general inventory

Using this information, you might use MainBoss as follows:

1. Use **Items** | **Item Restocking** to determine what inventory items you need to buy.

2. Use MainBoss item pricing records to determine the best vendors to buy from.

3. Draw up draft purchase orders for each vendor. Essentially, you're using MainBoss as a "scratchpad" where you can work out what to order from whom.

4. Once you've used MainBoss to clarify what you want to buy, you can use information from MainBoss's purchase orders to create purchase orders using your actual purchasing system.

5. When you submit a purchase order to your actual purchasing system, **Issue** the corresponding MainBoss purchase order (including any useful information from your purchasing system, e.g. ID numbers).

6. When items are received, you can use MainBoss receiving facilities to indicate where the items should be stored. MainBoss can also tell you if a received item is allocated to a specific job, or just to general inventory.

7. When all the items on a MainBoss purchase order have been received, you can **Close** the purchase order to indicate it's finished.

If you use MainBoss to record in this way, maintenance workers will have convenient access to relevant information (if they have the appropriate security roles). In addition, you can use MainBoss to analyze purchasing information and to include that information in other reports. You can obtain a more complete picture of all your maintenance operations by including the information from **Purchasing** as well as other MainBoss modules.

Payment Terms and Shipping Modes

Coding Definitions | **Purchase Orders** | **Payment Terms** lets you define the terms of payment you might use on purchase orders. Here's a short list of examples:

```
Cash at time of purchase
Charged to credit card
C.O.D.
Net 30
Added to account
```

If you specify payment terms for a purchase order, the terms will be printed on the P.O.

Coding Definitions | **Purchase Orders** | **Shipping Modes** lets you specify possible ways in which goods can be shipped from a vendor to your company. Here are some examples:

```
Overnight courier
Normal freight
Expedited freight
Personal pick-up
```

Again, if you specify a shipping mode on a purchase order, it will be printed on the P.O.

Payment term and shipping mode records are created in the same manner as most other simple records (e.g. unit categories as described in *Unit Categories* on page 62).

Purchase Order Status Codes

Purchase order status codes give you a way to indicate when a purchase order needs someone's attention. For example, you might create a status `Waiting for approval` to mark a purchase order which has been written up but not approved by management.

As another example, you might create a status `Items broken during shipment.` A receiving clerk could use this status to mark a purchase order with breakage problems. The status would alert other people in the maintenance department that someone should contact the vendor and deal with the situation.

While it is possible to use status codes for any purpose, we recommend that you only use them for purchase orders that are notable in some way. "Normal" purchase orders should have their "**Status**" field blank.

Note that status codes can be a useful form of communication between users. For example, suppose that a worker has the `PurchaseOrderReceive` security role (meaning that the worker can record the receipt of purchase order line items) but does not have the `PurchaseOrderClose` security role (which is needed to close a purchase order). The worker could give the purchase order a status like `Ready to Close`, indicating that all appropriate information has been recorded. A supervisor with `PurchaseOrderClose` permission could regularly check all open purchase orders to see if any are `Ready to Close`; if so, the supervisor can review those purchase orders and close them as appropriate (removing the status code in the process).

Remember that a status code is intended to bring a purchase order to someone's attention. You can therefore use status codes as signals for other users.

To set up your purchase order status codes, use **Coding Definitions** | **Purchase Orders** | **Purchase Order Statuses**. Clicking **New Purchase Order Status** lets you define a code.

Purchase Order Miscellaneous Records

Purchase orders may also include *miscellaneous* items. For example, if you pay a surcharge for fast delivery, you can specify this as a miscellaneous item added to the cost of the items that you bought. Sales tax might also be a miscellaneous cost. (Alternatively, you could include sales tax as part of each item's cost).

In practice, miscellaneous purchase order items are typically created "on the fly" rather than beforehand. For example, it's unlikely that you'll record all possible surcharges ahead of time; you won't create a record for a surcharge until you actually create a purchase order that uses the surcharge. On the other hand, the next time you incur the same surcharge, you'll have a record for it in the miscellaneous item table.

Purchase Order Templates

Preventive maintenance tasks may have associated costs for outside labor or for materials purchases. For example, you may hire a certified electrician to make an annual inspection of your electrical systems, or you may get a contractor to recharge your fire extinguishers. As another example, you may take company vehicles to an outside service center for regular maintenance like oil changes.

In such cases, you can add a *purchase order template* to the PM task description. This is not a real purchase order—it provides MainBoss with enough information to create real purchase orders when a work order is generated from the PM task.

For example, suppose your fire extinguishers need to be recharged every year and you hire an outside contractor to do the work. You can create a PM task description for recharging the fire extinguishers, and that task description can include a purchase order template to pay the company that does the recharging. When MainBoss generates a work order from the task description, it also generates a purchase order from the template. You can then send this purchase order to the recharging company as a request for service.

All purchase order templates have an identification code. This is *different* from the identifier that will appear on any real purchase orders generated from the template.

☐ **To create a purchase order template:**

1. In the control panel, go to **Coding Definitions** | **Unit Maintenance Plans** | **Purchase Order Templates**.

2. Click **New Purchase Order Template**. MainBoss opens a window where you can define the purchase order template.

3. In "**Code**" ❶, enter an identification code for this template.

4. In "**Description**" ❷, enter a longer description of the template.

5. In other fields of the template, enter any information that you want included in generated purchase orders, e.g. the vendor's name, payment terms, etc. (For information on filling out purchase orders, see the Day-to-Day Operations Guide.)

6. Click **Save** ❸ to save the current information on the purchase order.
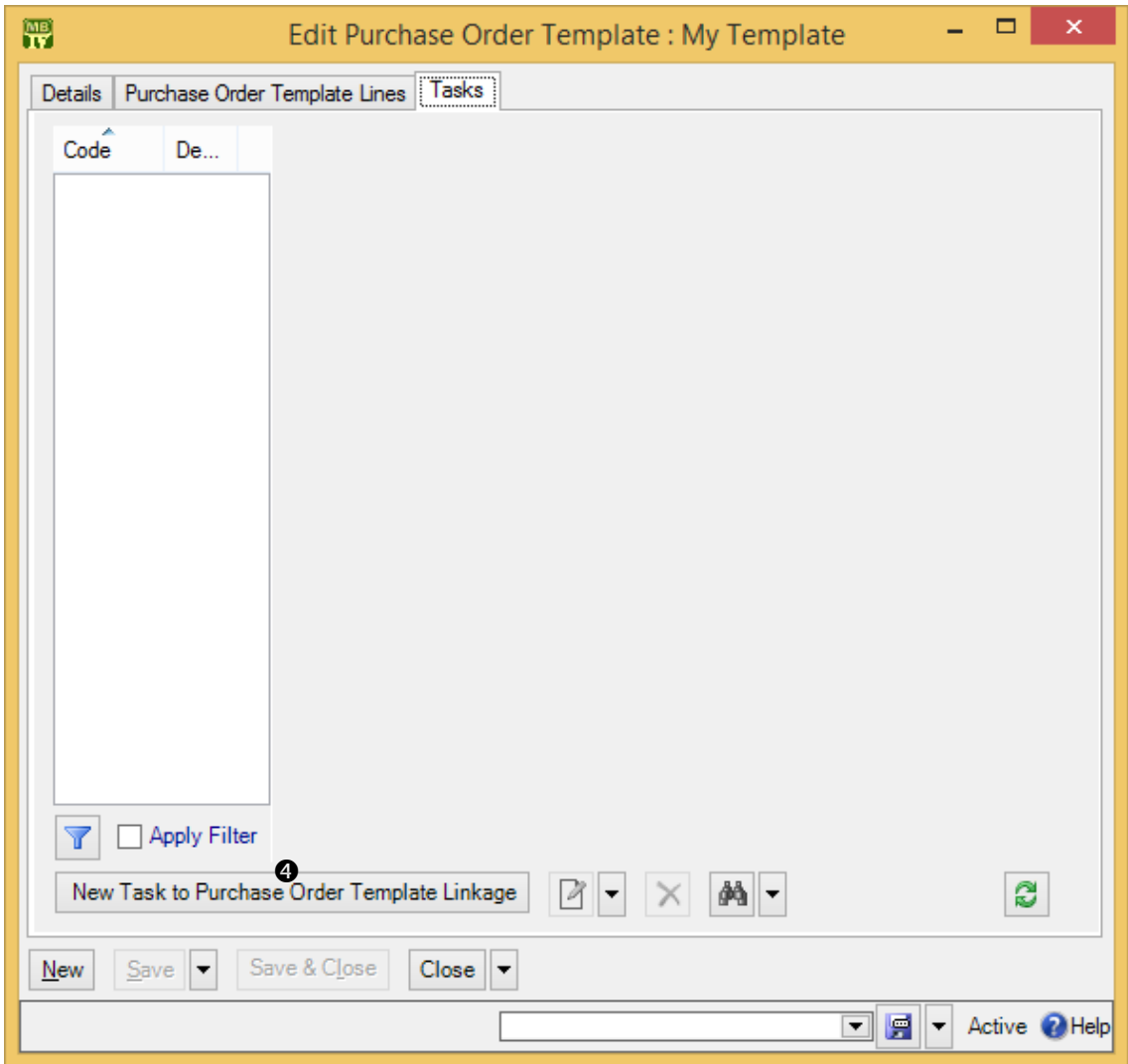
7. In the **Tasks** section of the purchase order template, click **New Task to Purchase Order Template Linkage** ❹. MainBoss opens a window where you can specify a task that should use this purchase order template.
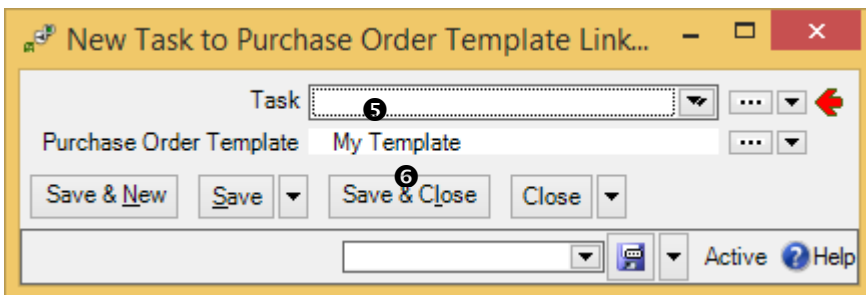
8. In "**Task**" ❺, specify an existing task that will use this template to create purchase orders.

9. Click **Save & Close** ❻ to save the information and close the window.

10. Click **Close** in the original purchase order template.

You can also link a purchase template and a task by starting with the task record. Just go to the task records **Purchase Templates** section and add a new template to the list. Either way, MainBoss will create an appropriate purchase order whenever it creates a work order from the task.

For more details about using purchase templates, see the [Reference Manual](#) or check the online help.

Purchase Order Assignees

Assigning a purchase order to someone is similar to assigning a request (as discussed in *Request Assignees* on page 84). Assigning a purchase order to someone indicates that the purchase order is of interest to that person. For example, you might assign a purchase order to the person who writes up the purchase order, that person's supervisor and the person who'll receive the incoming shipment. Assigning a purchase order to someone makes it easier for that person to check on the purchase order's progress.

Before a purchase order can be assigned to someone, that person must be listed in the **Purchase Order Assignee** table. The people listed in this table are authorized to be assigned purchase orders.

To add someone to the **Purchase Order Assignee** table, the person should first be in the **Contacts** table. (For more on contacts, see *Contacts* on page 50.) Once you've created an appropriate contact record, go to **Coding Definitions** | **Purchase Orders** | **Purchase Order Assignees** and click **New Purchase Order Assignee**. In the "**Name**" field, specify the person's contact record, then click **Save & Close**.

If an assignee can actually use MainBoss, that person can see his/her assigned purchase orders in **My Assignment Overview** | **Issued Purchase Orders**. This makes it easy for people to check up on any issued purchase orders to which they've been assigned. For example, you might check assigned purchase orders first thing in the morning to see if there are any new purchase orders you'd be interested in.

As with request assignees, you can use the **Receive Notifications** checkbox to indicate that purchase order assignees should be informed of changes to the purchase order history. The first such notification includes the original contents of the purchase order. Subsequent notifications omit this information, but give comments from the purchase order history records.

Purchase Order Defaults

The **Defaults for Purchase Order** section of the **Purchase Orders** table viewer lets you control important aspects of the purchase order. In particular, you can specify:

☐ The format used for purchase order identification numbers. The default format is `PO 00000001`, `PO 00000002`, and so on. However, you can change this to suit the needs of your organization.

☐ The starting point for numbering purchase orders. For example, if your organization has already used purchase orders up to 10,000, you can tell MainBoss to start numbering at 10,001.

☐ Additional text that should appear on a printed purchase order. For example, you can add extra text to the beginning and end of the purchase order (e.g. room for authorization signatures).

To set any of these defaults, go to **Purchase Orders** in the control panel, click on the **Defaults for Purchase Order** section, then click **Edit Defaults**. MainBoss will open a window where you can change the current defaults. For information about the default window, see the online help (press **<F1>**).

Administration

The **Administration** section of the control panel is mainly discussed in the Installation and Administration Guide and the online help. However, there is one entry in the **Administration** section that is relevant to configuration.

Company Information

The **Administration** | **Company Information** entry in the control panel lets you record information about your company.

☐ **To record company information:**

1. In the control panel, go to **Administration** | **Company Information**.

2. Click **Edit**. MainBoss opens a window where you can edit company information.

3. In "**Company Location**", specify the address of your organization. This should be a postal address record in your **Locations** table.

4. In "**Organization Name**", specify the name for your organization that you want to appear on various printed reports.

5. If you wish to record your company's logo, click **Select new Company Logo**. MainBoss opens a window where you can specify a file that contains your company logo in some suitable graphic format (e.g. JPG or BMP).

6. In "**Purchasing Contact**", specify the person within your company who should be contacted in connection with purchase orders. This person's contact information will appear on any purchase orders that MainBoss prints.

Don't forget to click **Save & Close** to save the information you've specified.

Division of Duties

In order to use MainBoss, you have to decide who does what: which maintenance personnel will be responsible for recording different types of information.

To help you make this decision, this chapter lists a number of *roles* that should be filled in order to use MainBoss productively. In most maintenance departments, one person will likely fill several roles; in a small maintenance department, a single person might do everything; in large departments, on the other hand, several people might all do the same role. Furthermore, there is no precise dividing line between roles. Even so, it is still useful to recognize the possible divisions of labor between personnel.

The roles are:

Implementation Committee: Decides how MainBoss will be used at your site and lays the ground rules (e.g. what naming conventions you'll use). The committee should establish a phase-in plan and get buy-in from everyone who has a stake in the outcome (management, workers, customers, etc.).

SQL Server Specialist: Ensures that SQL Server is installed and configured correctly for use by MainBoss. This person will also ensure the ongoing correctness of the configuration and ensure that regular backups of the MainBoss database are made.

Installer: Installs MainBoss software on all the computers where it will be needed and creates an initial database. This person must have Windows Administrator privileges in order to the install the software.

The Installer is the only person who needs Windows Administrator privileges. Other MainBoss users need no special Windows privileges.

Administrator: Records license keys for the database and makes sure that the Configuration Specialist (described next) is in the **Users** table. [Uses MainBoss's **Administration** features. Recommended security roles: `Administration`]

Configuration Specialist: Creates the initial MainBoss set-up. This person should be very familiar with all aspects of MainBoss, and with the policies of your maintenance department. Once the initial configuration is finished, this person's job is done (except for occasional tweaks to the system). [Only uses **Coding Definitions**. Recommended security roles: `CodingDefinitions` plus any others related to the data you will be entering. For example, if you are recording information on inventory items, you will also need `Item` and if you are recording information about accounting information, you will need `Accounting`.]

Technical Advisor for Configuration: Advises the Configuration Specialist on technical matters, particularly on planned maintenance tasks. This person should have extensive expertise in maintenance so that the task descriptions properly describe what's required for your existing units. [Advises Configuration Specialist on **Coding Definitions**, especially **Unit Maintenance Plans**. Recommended security roles: `Unit` and `UnitMaintenancePlans`, plus `ItemSummary`.]

Configuration requires knowledge of MainBoss and knowledge of maintenance. We've split these into two separate roles because they're two different types of expertise.

Equipment Specialist: Records information about units, or advises someone who actually types in the information. This person will be significantly involved in the configuration process, deciding what information should be recorded about each unit and gathering that information—specifications, spare part lists, warranty information, and so on. Someone must continue to fill this role after configuration, as new equipment is purchased and old equipment is retired. In particular, the equipment specialist should record planned maintenance procedures for new equipment and put those procedures into the schedule. [Uses **Units** and **Coding Definitions** | **Units**. Recommended security roles: `CodingDefinitions`, `Unit`, `UnitMaintenancePlans`, and `ItemSummary`.]

Inventory Specialist: Records information about inventory items, or advises someone who actually types in the information. [Uses **Items** and **Coding Definitions** | **Items**. Recommended security roles: `CodingDefinitionsView` and `Item`. If the person will enter item costs too, `Accounting` is needed.]

Help-desk: Receives problem reports and creates requests based on the information. Help-desk personnel don't need much expertise in maintenance or MainBoss use—just enough to ask relevant questions about the problem and to write an understandable description of what's gone wrong. [Only uses **Requests**. Recommended security roles: `Request`, `UnitSummary`, `WorkOrderSummary`, possibly `Contact`.]

Dispatcher: Creates work orders from requests. This may require obtaining more information about the problem, assigning personnel, reserving materials, and so on. The dispatcher generally needs a good knowledge of maintenance and of MainBoss. [Only uses **Work Orders**. Recommended security roles: `WorkOrder`, `AccountingWorkOrder`, `UnitSummary`, `ItemSummary`, `RequestView`, `RequestFulfillment`, `CodingDefinitionsView`, `UnitMaintenancePlansView`. If the Dispatcher will be linking work orders to purchase orders, `PurchaseOrderSummary` is also needed. If the Dispatcher will be recording the transfer of materials to some temporary location near the work site, `ItemFulfillment` is also needed. Finally, if the Dispatcher will be generating preventive work orders, `UnitMaintenancePlansFulfillment` is needed.]

Workers: Report back information from the job site; either they enter the information into MainBoss themselves, or they write out the information by hand for someone else to transcribe. Workers must know what information they're expected to record (e.g. time spent, materials used, closing codes, etc.). If they're asked to use MainBoss directly, they must have a basic familiarity with the software. [Only use **Work Orders**. Recommended security roles: `WorkOrderFulfillment`, `UnitSummary`, `ItemSummary`.]

Chargeback Administrator: Writes up chargebacks after a job is finished. This person must be familiar with contract agreements between your maintenance department and the chargeback customer—what the customer should and shouldn't be charged for, as well as the applicable rates. [Only uses **Work Orders** and **Coding Definitions** | **Work Orders** | **Billable Requestors**. Recommended security roles: `AccountingView`, `AccountingWorkOrder`, `WorkOrder`, `RequestView`, `CodingDefinitionsView`.]

Storeroom Personnel: Record the use of materials, take physical inventory, and ensure that supplies are restocked as required. [Only uses **Items**. Recommended security roles: `ItemView`, `ItemFulfillment`, `WorkOrderView`.]

Purchaser: Determines what goods and services need to be purchased and prepares appropriate purchase orders. [Uses **Items** and **Purchase Orders**. Recommended security roles: `PurchaseOrder`, `ItemSummary`, `WorkOrderSummary`.]

Receiver: Records the receipt of materials and arranges for them to be delivered to the correct storerooms. [Only uses **Purchase Orders** | **Receipts**. Recommended security roles: `PurchaseOrderReceive`, `ItemView`.]

Analyst: Prepares reports on various aspects of your maintenance operations. This person is usually a manager or a manager's assistant. [Uses **Reports** in various sections of the control panel. Recommended security roles: `View` versions of all relevant sections. For example, if the analyst is only concerned with work orders, then `WorkOrderView` is sufficient.]

Scrutineer: Maintains the consistency of the database and advises all other MainBoss users on use of the software. [May use any part of the software. Recommended security roles: `AccountingView`, `AdministrationView`, `CodingDefinitionsView`, `ItemView`, `PurchaseOrderView`, `UnitView`, `UnitMaintenancePlansView`, `WorkOrderView`.]

INDEX