Table of contents

1 Introduction 1-1
   Definition of the Baan GUI Style Guide 1-1
   Contents of the Baan GUI Style Guide 1-2
   GUI support 1-3

2 Design principles 2-1
   Definition of usability 2-2
   User-centered design 2-3
   Usability assessment 2-5
   Information processing 2-6
   Understanding users 2-7

3 Basic concepts 3-1
   User-interface objects 3-1
   The object-action principle 3-7

4 Window types 4-1
   General window properties 4-3
   Browse list 4-5
   Combined window 4-8
   Desktop manager 4-13
   Details window 4-15
   Dialog box 4-18
   Menu browser 4-31
   Message boxes 4-34
   Overview window 4-44
   Parameters 4-48
   Printing and Processing 4-51
   Progress indicator 4-57
   Wizards 4-59

5 Windows components 5-1
   Form page 5-2
   Grid 5-4
   Menus 5-6
   Scroll bar 5-17
   Status bar 5-19
   Title bar 5-20
About this document

This document describes the standards and guidelines for the Baan Software User Interface. This style guide is intended for those who are designing and developing Baan software.

Purpose of the Baan GUI Style Guide

The purpose of this document is to inform the application developer on the following topics:

- Design principles and basic concepts
- Standard Baan interface
- Software text standards

These standards and guidelines are important for developing easy-to-use applications which are consistent within and across other Windows applications. The style guide also provides you with a tool that you can use to develop Baan software in an efficient and effective manner.

Guidelines online

New guidelines are also available online. You can find these new GUI guidelines on the intranet Web site of GUI-Consultancy. The guidelines will be updated whenever new UI-decisions are there.

References

This document is based on the following publications:


Common User Access, IBM, 1989

BAAN V User Interface, U7034B US

Software Text Standards, S5020A US

Baan Writing Standards, S5005A US
# What’s new

## Chapter 3: Basic concepts

<table>
<thead>
<tr>
<th>Topic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>User-interface objects</td>
<td>In the design of the Baan user interface, objects play an important role. This guideline introduces user-interface objects, and describes how they should be used.</td>
</tr>
<tr>
<td>Object-action principle</td>
<td>The Baan user interface should be based on the object-action model. The guidelines describe how the object-action principle should be implemented.</td>
</tr>
</tbody>
</table>

## Chapter 4: Window types

<table>
<thead>
<tr>
<th>Topic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Browse list</td>
<td>The browse list displays a list of objects or properties of objects. There are some new guidelines regarding the buttons on a browse list.</td>
</tr>
<tr>
<td>Combined window</td>
<td>The combined window is the former overview window type 3, which is used to present two type of views on the user-interface object: combined view and group by view.</td>
</tr>
<tr>
<td>Generic Browser Framework (GBF)</td>
<td>The GBF is a window type for graphically depicting hierarchical structures. The GBF provides a container view on a particular user interface object: it shows the data of the object as a collection of other objects. The GBF uses a tree structure to show the objects.</td>
</tr>
<tr>
<td>Menu browser</td>
<td>The menu browser should only contain user-interface objects. All the actions and views must be removed from the tree structure and must be linked to its related object.</td>
</tr>
<tr>
<td>Message box</td>
<td>Messages are used to display information about a particular situation or condition. They occur during data entry, during the execution of processes, and when opening objects or BAAN. Messages are presented in a secondary window called a message box. This section gives detailed guidelines on using message boxes.</td>
</tr>
<tr>
<td>Wizard</td>
<td>A wizard is a special way of providing assistance to the user. A wizard automates a task for the user with the use of several dialog boxes. Wizards are meant for complex tasks or tasks that are not done on a regular basis. These guidelines explain when and how you should design a wizard.</td>
</tr>
</tbody>
</table>
### Chapter 5: Window components

<table>
<thead>
<tr>
<th>Topic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Menus</td>
<td>A menu is a list of commands presented to the user. The user can make a menu appear from a menu bar, by means of the secondary mouse button or with a menu button. This section provides detailed guidelines on the labeling, sequencing, and grouping of menu items in the Specific menu.</td>
</tr>
<tr>
<td>Status bar</td>
<td>A status bar is the part of a window that shows the current state of what is being viewed in the window or message on a successfully completed action. This section gives detailed guidelines on using the status bar.</td>
</tr>
<tr>
<td>Title bar</td>
<td>The guidelines on window titles have been changed according to the object-action principle.</td>
</tr>
</tbody>
</table>

### Chapter 6: Window controls

<table>
<thead>
<tr>
<th>Topic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Browse control</td>
<td>The browse control is the former browse area.</td>
</tr>
<tr>
<td>Column heading</td>
<td>The guidelines on the column heading have been elaborated. For instance: the labels of the column headings must be top-left aligned.</td>
</tr>
<tr>
<td>Command button</td>
<td>A command button is a control that starts an action when pressed. It contains a label that describes the action. The command button is used for actions in secondary windows, where there is no menu bar. New guidelines on the type and position of the buttons have been added.</td>
</tr>
<tr>
<td>Details button</td>
<td>A details button is a small command button that is positioned next to an input field. By activating this button the details of the input field are opened in a modal secondary window.</td>
</tr>
<tr>
<td>Menu button</td>
<td>The menu button is a new control within Baan. The menu button is a command button that displays a drop-down menu when it is pressed. The user can choose one of the items in the menu.</td>
</tr>
<tr>
<td>Selection range</td>
<td>A selection range is used when a selection of data is required. These ranges consist of From-To fields. Guidelines have been changed because of the possibilities of the dynamic form editor.</td>
</tr>
<tr>
<td>Text box</td>
<td>Guidelines on text boxes have been elaborated. For instance on the field alignment.</td>
</tr>
<tr>
<td>Toolbar buttons</td>
<td>The toolbar buttons will only be designed by GUI-Consultancy. New buttons can be requested by filling in the request form on the intranet site of GUI-consultancy.</td>
</tr>
</tbody>
</table>
### Unit fields

A Unit field is a Static Text field, which presents read-only information about a related field. Guidelines have been changed because of the possibilities of the dynamic form editor.

### Chapter 7: Interaction and navigation

<table>
<thead>
<tr>
<th>Topic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Browse filter</td>
<td>Browse filters prevent the user from selecting an object in a browse list that is not valid. Browse filters also reduce the number of objects presented in the browse list, which helps you find an object. You should use browse filters on each input field with a check and a browse list.</td>
</tr>
<tr>
<td>Navigation flow</td>
<td>Because of the restructuring of the menu browser, (the removal of all the actions and views), the navigation flow has been changed as well.</td>
</tr>
<tr>
<td>Pointer</td>
<td>A pointer must be displayed whenever a pointing device (mouse) is used. A pointer is used to point to choices and objects that a user wants to select or otherwise interact with. The shape of the pointer changes to provide feedback about a particular location, operation, or state. Guidelines can be found in the Input basics section.</td>
</tr>
<tr>
<td>Shortcuts</td>
<td>The range of available shortcuts for the menu items in the Specific menu have been extended.</td>
</tr>
</tbody>
</table>
1 Introduction

The design of the user interface influences the way the user experiences the Baan software. This style guide promotes good interface design, as well as visual and functional consistency within Baan and across Window-based applications.

Definition of the Baan GUI Style Guide

The Baan GUI Style Guide consists of a set of statements that define the look and feel of the Baan User Interface.

This reference is based on and extends the following guidelines:

- **Microsoft**

- **IBM**
  Common User Access [1989].

These documents have been extended because they do not specify precise directions and rules for the development of the Baan user interface.

When to use

- To develop Baan software that is easy to use and consistent.
- To develop Baan software that is consistent with Microsoft applications.

Guidelines

- Avoid changing an existing behavior for common elements. A user builds up expectations about the workings of an interface. Inconsistencies not only confuse the user; they also add unnecessary complexity.

- Exceptions to the guidelines must be approved by a GUI consultant.
Contents of the Baan GUI Style Guide

This style guide is intended for those who are designing and developing Baan software.

The guide covers the following areas:

- **Fundamentals**
  Chapters 2 and 3 include fundamental design philosophy, assumptions about human behavior, and concepts embodied in the Baan software.

- **Baan interface**
  Chapters 4, 5, 6, 7, and 8 include guidelines about the various windows used in the Baan environment, about the window components, and about the use of controls and navigation and interaction. The last chapter lists the software text standards that you must follow.

- **Appendix**
  The appendix includes checklists of the various window types and includes Software Text Standards.

Each paragraph consists of the following parts:

- **Definition**
  Provides a description about the subject.

- **When to use**
  Defines when a particular subject is used.

- **Guidelines**
  Defines how a particular subject should be applied.

- **Components**
  Defines which components are included in a particular subject (if relevant).

- **Related topics**
  Defines a list of related topics.
GUI support

If you have questions or remarks about the user interface, please contact us. Your contribution can improve the usability of the Baan software.

Pass on your questions or remarks about the Baan user interface to the GUI consultancy group.

Who are we?

- The GUI consultancy group is located in Barneveld (NL) and in Hyderabad (India).
- The Dutch group consists of the following persons: Anneke Aldershof, Marieke Hoorweg, Angelique Matton, Evelyn van der Stap, Johan Versendaal, Harro Vons.
- The Indian team consists of the following persons: Kaladhar Bapu, Amit Chowdhury, Eggone Phani Kumar, Sreedhar Ranganathan.

What is our goal?

- Our goal is to develop a BAAN user interface that is easy to use, easy to learn, and consistent within and across Microsoft applications.

How can you contact us?

- Describe your problem or remark and mail it to our mailbox: GUI-Consultancy@baan.nl.
2 Design principles

A user interface is more than just what shows up on the screen. The user interface is that part of the application that determines how well the user and the system communicate with each other. This means that the user interface is crucial for the dialog between user and system. Through the user interface, the user must be able to perform his tasks in an effective, efficient, and intuitive manner.

Components

The user interface consists of three levels along which the communication between user and system takes place:

1 The conceptual level
   On this level the functionality of the system is defined.

2 The navigation level
   On this level the interaction-style and the structure of the dialog are defined.

3 The presentation level
   This level describes the input and output elements that are visible on the screen.
Related topics

Definition of usability
In this section the term usability is defined and it describes the general framework for usability.

User centered design
This section describes the seven design principles, which focus on the user: user in control, directness, consistency, forgiveness, feedback, aesthetics, and simplicity.

Usability assessment
This is an important part in the design process. It provides information about the way users do their work and it defines the usability requirements a system has to meet. This part also describes two kinds of usability techniques: usability testing and walkthrough.

Information processing
During the interaction with the system, multiple cognitive processes take place inside the user. These processes are described in this section.

Understanding users
This part describes some general user characteristics. This knowledge can be useful in developing a better understanding about what makes an application more or less usable.

Definition of usability
The general framework for usability embraces the four principal components of any work situation:

- User
- Task
- System
- Environment

Good design for usability depends on achieving successful harmony in the dynamic interplay of these four components. Therefore, usability can be defined in terms of the interaction between the user and the system in order to perform the task within the environment [Shackel, 1990].
Usability requirements

A usable application is an application, that is dedicated to the users and their tasks. An application is usable when it meets the following requirements:

- **Effectiveness**
  The user has to be able to perform the tasks for which the application is designed.

- **Efficiency**
  The user has to be able to perform the tasks with minimal effort.

- **Intuitively**
  The user has to have an immediate understanding of how the application works.

User-centered design

A well-designed user interface is based on a development process that is centered on the user and his activities. Here seven design principles are given which focus on the user. These principles are derived from MS-Windows. A full description can be found in The Window Interface Guidelines for Software Design [Microsoft, 1995].
User in control
The user should always have the feeling of being in control of the software, rather than feeling controlled by the software.

Directness
Users should be able to directly manipulate software representations of information (drag-and-drop). Users should see how the actions that are taken affect the objects on the screen. Visibility of information and choices reduce the user's mental workload. Recognition is easier than recall. Familiar metaphors provide direct and intuitive interface to user tasks and these support user recognition.

Consistency
Consistency within software applications makes the interface familiar and predictable. This enables the user to develop confidence in the application. Users do not need to spend time trying to remember the differences in interaction. Besides consistency within an application, there should also be consistency among applications. In this case Baan has to be consistent with Microsoft applications.

Forgiveness
The users like to explore the application by trial and error. A good UI anticipates this attitude and provides a limited set of options. However, avoid situations that are likely to result errors. If errors or mistakes occur, make them easy for the user to recover. This means make actions reversible and recoverable (Undo).

Feedback
Always provide feedback for a user’s action. Effective feedback is timely, and is presented as close to the point of a user’s interaction as possible. It is equally important that the type of feedback is appropriate to a particular task.

Aesthetics
Visual attributes provide valuable impressions and communicate important cues to the interaction behavior of particular objects. However, keep in mind that users have a limited attention span and that every visual element that appears on the screen demands attention of the user.
Simplicity
An interface should be simple, easy to learn, and easy to use. The interface must also provide access to all functionality provided by the application. Maximizing functionality and maintaining simplicity are two opposite objectives. This means that a balance must be found.

Usability assessment
Usability testing is a key part in the design process, but testing design prototypes is only one aspect. Another important part is usability assessment.

Usability assessment
Usability assessment starts in the early stages of product development. It is used to gather information about how users do their work, which serves as input into the design process. Subsequently, usability assessment provides valuable input for analyzing initial design concepts and it can be used to test specific product tasks. Consider the user's entire experience as part of a product's usability.

Usability techniques
Two kind of techniques are described:

1  Usability tests
   Usability testing starts by defining the target audience, test goals (requirements) and scenarios. Four or five participants are enough to identify 80 percent of most design problems.

   When conducting the usability test, provide an environment comparable to the target setting. Make participants feel comfortable; they are not tested, the product is tested.

   During the test, do not interrupt the participants, unless they get stuck. Ask participants to think aloud as they work, so that assumptions and inferences become visible. Record the test results by using a video camera. The recorded data is used to review and evaluate the results. The provided information can be used to redesign the product to ensure usability.

2  Walk-through
   With this technique the user is taken through a set of sample scenarios and is asked about his impressions along the way. By means of the intention of the user and the feedback of the system, it is possible to detect which actions pose problems to the user.
**Information processing**

During interaction with the system several processes are going on in the head of the user. The following processes are needed to carry out the action:

- The user has to define his goal and to form an intention: what do I want to achieve?
- The user has to specify the action: what kind of action needs to be performed to achieve the goal?
- The user has to carry out the action.

The system provides the effect of the action (output) on the screen. To evaluate the output:

- The user has to perceive the state of the system: what do I see?
- The user has to make sense of it; interpret the state of the system: what does it mean?
- The user has to evaluate the outcome in relation to his intentions: have I achieved my goal?

*Figure 3, Processes during interaction between user and system*
Understanding users

This part describes the following subjects concerning users in relation to designing user interfaces: general user characteristics and types of users. Additional information can be found in The Window Interface Guidelines for Software Design [Microsoft, 1995].

General user characteristics

Knowledge about user characteristics can be useful to develop a better understanding about what makes an application more or less usable. Some general user characteristics are described below.

Limitation of short-term memory (STM)

A user can only remember five to nine (7 +/- 2) related items at one time. This means that a maximum of nine related items should be placed in one logical group. The implication is that information has to be divided into meaningful groups.

Limited attention span

A user can only focus on one logical group at one time. For this reason related items should be placed near each other. This means that meaningful groups of menu items have to be designed, which are distinctive of each other. The same counts for the grouping of items in meaningful group boxes.

Limitations of long term memory (LTM)

Information is better and more easily acquired if it makes sense, if it can be integrated into some conceptual framework. Moreover, retrieval from LTM is apt to be slow and to contain errors. This implies that task-relevant information has to be visible on the screen and it has to be meaningful for the user.

Recognition is better than recall

Users can recognize objects better than having to recall them from memory. Making everything relevant to a task visible on the screen relieves the load on the user’s memory. This means that the user can pay attention to the task to be performed. Make things visible. (A picture is worth a thousand words.)
Types of users

The user interface should be built to suit the needs of the users. However, the user does not exist. Groups of users vary in their knowledge of computers and their task knowledge. It is important to know what types of people will be using the interface. Because the usability of an application depends on the type of user, an application has to be effective, efficient, and intuitive, but these criteria are not equally important for each type of user.

Novice users

These users need an application that is effective and, more importantly intuitive.

Advanced users

These users need an application that is effective and efficient.

The challenge in designing for advanced users is providing efficiency without introducing complexity for less-experienced users. (Shortcut methods are often useful for supporting these users).
3 Basic concepts

In the design of BaanERP’s user interface, objects play an important role. This chapter introduces user-interface objects, and describes how they should be used.

User-interface objects

BaanERP has an object-oriented user interface. This means that users handle objects (as opposed to applications or sessions). The objective of object-orientation is to make the user interface intuitive: users manipulate objects in their daily life all the time.

By using objects in the user interface that remind the user of objects known from the real world, the user will more easily know what to use the objects for.

Definitions

<table>
<thead>
<tr>
<th>User interface object</th>
</tr>
</thead>
<tbody>
<tr>
<td>A user-interface object is a visual component that the user recognizes from reality and which is used as an independent unit.</td>
</tr>
</tbody>
</table>

There are numerous types of user-interface objects. Examples of such objects in MS Windows are the Recycle Bin, a folder, a document or a printer. Within BaanERP a Sales Order or a Project are examples of user-interface objects.

Figure 4, Some user-interface objects from Microsoft Windows.
Views

There are several ways to represent one single object in the user interface. For instance, a folder can be shown as an icon or in a window. In Figure 5 two such representations of the same folder are shown.

![Figure 5](image)

*Figure 5, Two ways of showing the same folder (called Documents) in Microsoft Windows: as an icon or in a window.*

When you show a user-interface object in a window, you can choose between different ways of presenting the information, called different views.

<table>
<thead>
<tr>
<th>View</th>
</tr>
</thead>
<tbody>
<tr>
<td>A way of presenting the information of one single user-interface object in one single window, without changing the contents of the object.</td>
</tr>
</tbody>
</table>
In BaanERP, you can distinguish between three basic types of views on a user-interface object: a data view, a container view, and a combined view. Table 1 explains these three views.

<table>
<thead>
<tr>
<th>View</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data view</td>
<td>Shows (part of) the data of the user interface object with fields and labels.</td>
<td><img src="image" alt="A data view on a Machine." /></td>
</tr>
<tr>
<td>Container view</td>
<td>Shows (part of) the data of the user interface object as a list or a tree.</td>
<td><img src="image" alt="A container view on a list of Business Partners." /></td>
</tr>
<tr>
<td>Combined view</td>
<td>Shows (part of) the data of the user interface object as a combination of labels, fields and a list.</td>
<td><img src="image" alt="A combined view on an ‘Order’." /></td>
</tr>
</tbody>
</table>

Table 1, *The three standard views on a user-interface object in BaanERP.*
Window types

For each of the basic views on a user-interface object a standard type of window is available in BaanERP.

Table 2 describes which window type you can use for each particular view.

<table>
<thead>
<tr>
<th>View</th>
<th>Window type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data view</td>
<td>Details window</td>
</tr>
<tr>
<td>Container view</td>
<td>Overview window ¹</td>
</tr>
<tr>
<td></td>
<td>Menu browser</td>
</tr>
<tr>
<td></td>
<td>Desktop manager</td>
</tr>
<tr>
<td>Combined view</td>
<td>Combined window ²</td>
</tr>
</tbody>
</table>

Table 2, The different window types for each view in BaanERP.

The examples of Table 1 show a data view in a details window, a container view in an overview window, and a combined view in a combined window. Next to these windows, BaanERP features other windows that show a container view, such as the menu browser. This window shows its contents with a tree.

Figure 6, The menu browser shows a container view on a user-interface object with a tree structure.

¹ The overview window is the former overview window without a group area (form type 2).

² The combined window is the former overview window with a group area (form type 3)
Refer to Chapter 4, ‘Window Types’ for more information on when and how to use each type of window.

**Example**

Figure 7 shows an example of three different user-interface objects in BaanERP. The first user-interface object is a list of Projects. It is opened in a window labeled Projects. This window just shows a list of objects, so it gives a container view on the object.

Each Project in the list is also a user-interface object. One of these Projects is opened in another window labeled Project - Business Partners. This window shows the selected Project in a combined view. It contains some data of the Project and a list of its Business Partners.

The Business Partners in the Project are all user-interface objects as well. One of them is opened in a window titled Business Partner. Because it shows the information of the objects just by means of labels and fields, this window shows a data view on the Business Partner.

*Figure 7, Three different user-interface objects: a list of Projects, a Project, and a Business Partner.*
**When to use**

Make a user-interface object for everything you want to show more information about than just an icon or a single line in a list. That means that you need a user-interface object whenever you want to show information by means of a details window, an overview window, or a combined window.

Use each type of window as follows:

- **Details window**: when you need to present detailed information about one single user-interface object.
- **Overview window**: when you need to present information about a user-interface object in the form of a list or a tree. For instance when the user-interface object contains other objects.
- **Combined window**: when you need to present both detailed information about a user-interface object and a list in one window.

**Guidelines**

- You can use user-interface objects that can be easily recognized from the users daily life and work. For instance, if the user normally works with a list of doubtful customers, make it an object in the user interface as well.
- Give a user-interface object the same name as the corresponding object in the user’s real life. For instance, if the user calls the list of doubtful customers the Black List, then use this name for the corresponding user-interface object as well.
- Only make user-interface objects for things that exist in the user’s real world. For example, a line on an order is not an object in the real world. Therefore, it should not be an object in the user interface.
- Group related information on a user-interface object in different views. For instance, one user may only be interested in sold-to Business Partners of a Project, and another user may only need to see the Project’s buy-from Business Partners. In this case you will need to provide two different views on the Project, one for each type of Business Partner.
- Provide only user-interface objects that are relevant for the user’s task.
- Only provide those actions and views on a user-interface object that are relevant for the user’s task.
The object-action principle

The object-action principle is the key principle of an object-oriented user interface: it describes how users work with user-interface objects. According to this principle, a user selects an object, and then selects an action to perform on this object.

The object-action principle is deeply embedded in Microsoft Windows. In a folder, a user can select any object and then choose one out of several actions. Usually, several different ways exist to select an action. Figure 8 shows two different ways to access actions in a folder: from a menu in the window or from a menu that pops up when the user clicks on the object with the right mouse button.

In BaanERP, the object-action principle is implemented in a similar way. In a combined window or an overview window, the user can select an object in the grid or in the tree and choose an action on it from the menu bar in the window.

When to use

Use the object-action principle for any action on any user-interface object. Some examples:

- Print and processing sessions perform actions on an object. The user should be allowed to first make a selection of one or more objects in an overview window or a combined window and then activate a print or processing session.

- Many actions in the menu bar of a window perform actions on an object. In an overview window or a combined window an action can apply to the whole list as such or to one or more objects selected in the list.
The menu browser shows a list of user-interface objects in a tree structure. The user can select an object and perform an action on it. Currently, the only action possible is to open an object by double-clicking on it.

A details window shows just one object. All actions in that window apply to that object.

**Guidelines**

**Menu browser**

- Only place user-interface objects in the menu browser. The user can open the object in a separate window and select another action from the menu of that window.
- Do not place actions or views in the tree of the menu browser. Print and processing sessions are not allowed in the menu browser.

**Overview window and combined window**

- Double clicking on an object in the grid opens it in a details window.
- Open a container view or a combined view on an object in the grid with an action in the Specific menu (and possibly a shortcut key or button on the toolbar).
- Switch between different views on an object by means of actions in the Specific menu of the object’s menu bar. For instance, if you have a list of Business Partners that shows all sold-to BPs, you can switch to the view that shows all buy-from BPs with an action in the Specific menu.
- Label actions that change the view on a user-interface object to a different grouping: “Group by” `<grouping attribute>`. Example: “Group by sold-to Business Partners”.
- Label actions that change the presentation of the list in a user-interface object (for instance by filtering or sorting it): “View `<description>`”. Examples: “View All”, “View in Buffer”, “View not Arrived”, “View Started”, and “View Completed”.
- Place an action on one or more objects from the list in the Specific menu. For instance, in a list of Work Centers, the action Update Resources only applies to the selected Work Center(s) in the list.
Only enable actions that apply to objects in the list when the user has selected at least one object. The action Update Resources for example, is only available after the user has selected the Work Center whose resources need to be updated.

Figure 9, An action on an object should be disabled when the object is not selected.

Place actions that open a view on a selected object from the list in the Specific menu. Label this action <View>, such as “Locations” for a Warehouse.

An exception to this guideline is opening an object in a details window. This action is called Open and is placed in the File menu of the overview window or combined window.

Label other actions on selected objects according to the following table:

<table>
<thead>
<tr>
<th>Action</th>
<th>Label format of menu-item</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action on entire object</td>
<td>&lt;action&gt; (do not mention the object)</td>
</tr>
<tr>
<td>Action on selected object(s) in the list</td>
<td>Overview window: &lt;action&gt; (do not mention the grid items) Combined window: &lt;action&gt; &lt;grid items&gt;</td>
</tr>
</tbody>
</table>

For more guidelines on the labeling, ordering, and grouping of actions in the Specific menu, refer to Chapter 5, ‘Window components’, Section ‘Menus’.

3 The items in the grid are identified to indicate that the action is applicable to the item(s) in the grid and not to the entire object.
General

- Make the title of the window reflect the type of user-interface object and the view it provides on the object. Refer to Chapter 5, ‘Window Components’, Section ‘Title Bar’ for guidelines on window titles.

- In a dynamic form, changing views on a user-interface object should not open a new window. The view should change within the same window.

Related topics

- Command buttons
- Menus
- Selection ranges
- Window types
A window is an area on the display screen used to present objects, action options, and messages.

**Guidelines**

- Make sure the start position of the window enables the whole window to be visible on an 800x600-resolution screen.
- Avoid dragging the window to make it visible.

**Table of windows**

The following table lists the various window types and their properties, such as primary/secondary, modal/modeless, and display/editable.

<table>
<thead>
<tr>
<th>Window types</th>
<th>Primary/Secondary</th>
<th>Display/Editable</th>
<th>Modeless/Modal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Browse list</td>
<td>Secondary</td>
<td>Display</td>
<td>Modal</td>
</tr>
<tr>
<td>Combined window</td>
<td>Primary</td>
<td>Display/edit</td>
<td>Modeless</td>
</tr>
<tr>
<td>Desktop manager</td>
<td>Primary</td>
<td>Display/edit</td>
<td>Modeless</td>
</tr>
<tr>
<td>Details window</td>
<td>Primary</td>
<td>Display/edit</td>
<td>Modeless</td>
</tr>
<tr>
<td>Dialog box</td>
<td>Secondary</td>
<td>Display/edit</td>
<td>Modeless/modal</td>
</tr>
<tr>
<td>Generic Browse Framework (GBF)</td>
<td>Primary</td>
<td>Display</td>
<td>Modeless</td>
</tr>
<tr>
<td>Menu browser</td>
<td>Primary</td>
<td>Display</td>
<td>Modeless</td>
</tr>
<tr>
<td>Message box</td>
<td>Secondary</td>
<td>Display</td>
<td>Modal</td>
</tr>
<tr>
<td>Overview window</td>
<td>Primary</td>
<td>Display/edit</td>
<td>Modeless</td>
</tr>
<tr>
<td>Parameters</td>
<td>Secondary</td>
<td>Edit</td>
<td>Modeless</td>
</tr>
<tr>
<td>Printing &amp; processing</td>
<td>Secondary</td>
<td>Edit</td>
<td>Modal</td>
</tr>
<tr>
<td>Progress indicator</td>
<td>Secondary</td>
<td>Display</td>
<td>Modal</td>
</tr>
<tr>
<td>Wizard</td>
<td>Secondary</td>
<td>Display/edit</td>
<td>Modal</td>
</tr>
</tbody>
</table>
## Table of components

An overview of the components included in each window is given below. A full description of these components is provided in Chapter 5.

<table>
<thead>
<tr>
<th>Window type</th>
<th>TiB</th>
<th>M</th>
<th>ToB</th>
<th>StB</th>
<th>ScB</th>
<th>G</th>
<th>MFP</th>
<th>CB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Browse list</td>
<td>☑</td>
<td></td>
<td>☑</td>
<td></td>
<td>☑</td>
<td></td>
<td></td>
<td>☑</td>
</tr>
<tr>
<td>Combined window</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td></td>
<td></td>
<td>☑</td>
</tr>
<tr>
<td>Desktop manager</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td></td>
<td>☑</td>
<td></td>
<td></td>
<td>☑</td>
</tr>
<tr>
<td>Details window</td>
<td>☑</td>
<td></td>
<td>☑</td>
<td></td>
<td>☑</td>
<td></td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td>Dialog box</td>
<td>☑</td>
<td></td>
<td>☑</td>
<td></td>
<td>☑</td>
<td></td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td>GBF</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td></td>
<td></td>
<td>☑</td>
</tr>
<tr>
<td>Menu browser</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td></td>
<td>☑</td>
<td></td>
<td></td>
<td>☑</td>
</tr>
<tr>
<td>Message box</td>
<td>☑</td>
<td></td>
<td>☑</td>
<td></td>
<td>☑</td>
<td></td>
<td></td>
<td>☑</td>
</tr>
<tr>
<td>Overview window</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td></td>
<td>☑</td>
<td></td>
<td></td>
<td>☑</td>
</tr>
<tr>
<td>Parameters</td>
<td>☑</td>
<td></td>
<td>☑</td>
<td></td>
<td>☑</td>
<td></td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td>Printing &amp; Processing</td>
<td>☑</td>
<td></td>
<td>☑</td>
<td></td>
<td>☑</td>
<td></td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td>Progress indicator</td>
<td>☑</td>
<td></td>
<td>☑</td>
<td></td>
<td>☑</td>
<td></td>
<td></td>
<td>☑</td>
</tr>
<tr>
<td>Wizard</td>
<td>☑</td>
<td></td>
<td>☑</td>
<td></td>
<td>☑</td>
<td></td>
<td></td>
<td>☑</td>
</tr>
</tbody>
</table>

The next paragraphs provide detailed information about each window type: definition, when to use, guidelines, components, and related topics.

---

4 TiB = Title bar; M = Menu bar; ToB = Toolbar; StB = Status bar; ScB = Scroll bar; G = Grid; MFP = Multiple form pages; CB = Command button.

5 A status bar will be included in each window type (probably Vivaldi).

6 Generic Browser Framework (GBF)
General window properties
Windows can be divided according to their properties: primary, secondary, modeless, and modal. A window can be either primary or secondary: primary window is always a modeless window, and secondary can be modeless or modal. These four properties are described in this paragraph.

Definition of primary and secondary
- A primary window is a window in which the main interaction between the user and an object takes place.
- A secondary window is a window that is used to supplement the interaction in the corresponding primary window.

When to use
Primary window
- To display a view of an object that is independent of all other windows with respect to the closing, opening, minimizing, and restoring actions.
- To display all views of objects.

Secondary window
- To display an action window that allows a user to specify criteria/settings.
- To view an object, only if the secondary window is used to support interaction with its associated primary window.
- To display a message.

Components
- Components included in a primary window:
  - Title bar, menus, toolbar, scroll bar, status bar, form page, and grid
- Components included in a secondary window:
  - Title bar, multiple form pages, and command buttons
Definition of modeless and modal

- A modeless secondary window allows the user to interact with either the secondary window or the primary window, just as the user can switch between primary windows.
- A modal secondary window requires the user to complete interaction within the secondary window and close it before continuing with any further interaction outside the window.

When to use

- Modeless secondary windows are used for parameter windows.
- Modal secondary windows are used for printing & processing, dialog box, message box, browse list, and progress indicator.

Guidelines

- Modeless: Actions such as selecting another record, deleting a record, or inserting a record are still possible in the primary window while the secondary window remains opened. The user does not have to close the secondary window.
- Modal: The modal window must be closed before the user can continue in another window.
- Use modal secondary windows sparingly because they restrict the user’s choice.
- A primary window is always modeless.
- Closing the primary window means that all secondary windows will also be closed.
Browse list

A browse list is a modal, secondary window that is started when the user browses on an input field in the details window or overview window to retrieve a value for that input field.

The browse list displays a list of objects or properties of objects. Only those objects or properties of objects are displayed that are valid within the context of the related input field (browse filter).

Guidelines

Size

- The Browse list must fit on an 800 x 600-resolution screen.
- No more than 13 visible rows are allowed.
- This window is not resizable for the user.

Window title

- Title format for browse list: <Grid Items>
- The <Grid Items> must always be plural in a browse list, because it lists multiple object/items. For example: Employees

- Refer to Chapter 5, ‘Window Components’, section ‘Title Bar’ for more guidelines on window titles.
Command buttons

- Predefined command buttons:
  - **OK**: to export the selected value and to close the browse list.
  - **Cancel**: to close the browse list without exporting the value.
  - **Sort By**: to change the current key of the main table.
  - **Find**: to search a specific value.
  - **Filter**: to select values that match a user-defined query.
  - **Open Read-Only**: to open a modal secondary display window that is used to view the details of a selected object in the Browse list.
  - **New**: to open a modal secondary editable window in which a new object can be inserted.
  - **Help**: to display Help.

- Avoid adding any other actions to the browse list. If you need to do so, only add those actions that make it easier for the user to find the objects for which they are looking. These actions should be placed in a menu button labeled Advanced. Place this button between the New and Help buttons.

- Refer to Chapter 6, ‘Window Controls’, section ‘Command buttons’ and section ‘Menu button’ for more guidelines.

Column headings

- All labels must be top-left aligned (independent of the alignment of the fields inside the columns).

<table>
<thead>
<tr>
<th>Position</th>
<th>Rem</th>
<th>Effcl. Date</th>
<th>Expiry Date</th>
<th>Net Quantity</th>
<th>Inv. Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Y</td>
<td>225 05-05-98</td>
<td></td>
<td>40.0000</td>
<td>cm</td>
</tr>
</tbody>
</table>

*Figure 11, Column headings*

- Refer to Chapter 6, ‘Window Controls’, section ‘Column headings’ for more specific guidelines on column headings.
Shortcut keys

- **CTRL+B**: to browse on an input field in the details window.
- **Enter**: key to activate the OK button.
- **Esc**: key to activate the Cancel button.
- **New**: CTRL+N
- **Find**: CTRL+F
- **Help**: F1

Status bar

- Refer to Chapter 5, ‘Window Components’, section ‘Status bar’ for more guidelines on the status bar.

Components

- Components included in the browse list:
  - Title bar, command buttons, grid, scroll bar, and status bar
- For guidelines on these components, refer to Chapter 5, ‘Window components’.
- Components not included in the browse list:
  - Menu bar, toolbar, and multiple form pages

Related topics

- Browse Control
- Browse Filter
- Column headings
- Command buttons
- Details window
- Menu Button
- Navigation flow
Combined window

A combined window is a modeless primary window that can either be a display or editable. A combined window is used to present two different views: combined view and group by.

When to use

- Use a combined window when you need to present detailed information about a user-interface object and a list in one window (combined view). In other words, a combined window presents one user-interface object whose details are shown in the group area, and its related objects/items are listed in the grid.

For example, Purchase Order Advice is a user-interface object. The details of the object are presented in the group area of the combined window and its Lines are listed in the grid.

---

7 The combined window is the former multi occurrence with a group area (form type 3)
• Use a combined window when you need to group a list of objects/items (group view). The grouping is defined in the group area and the objects/items belonging to the active group are listed in the grid.

For example, List of Purchase Contracts is a user-interface object. The Purchase Contracts in the list can be grouped by Purchase Office. The active group is presented in the group area and the Contracts belonging to this group are listed in the grid.

![Figure 13, Combined window which presents a 'Group By' view](image)

**Guidelines**

**Synchronization**

• A combined window (type 3) must be synchronized with a details window of type 3.

• Refer to Chapter 7, ‘Navigation’, section ‘Synchronization’ for more information on synchronization.
Size

- An overview window must fit on an 800 x 600-resolution screen.
- Avoid horizontal scrolling.
- The window is resizable for the user.

View

- The combined window presents:
  - a combined view on the user-interface object.
  - a grouping on a list of objects
- In a dynamic form, changing views on a user-interface object should not open a new window. The view should change within the same window.
- Refer to Chapter 3, 'Basic concepts', section 'User-interface objects' for more information on views.

Window title

- Make the title of the window reflect the type of user-interface object and the view it provides on the object.
- Title format for combined window: <Object type> - <View>
  - The <Object type> must be singular in case the combined window presents a single user-interface object, but it must be plural in case the combined window is used for grouping.
  - The <View> must reflect the active view on the user-interface object.

Use the window title format <Object type> - "Quick Entry" in case the combined window is used to perform quick entry of particular objects in an editable grid.

Refer to Chapter 5, ‘Window Components’, section ‘Title Bar’ for more guidelines on window titles.
Menu

- File - Open or double-clicking on an object in the grid, opens a details window in which more information of the object is shown.
- Open a container view or a combined view on an object in the grid by means of an action in the Specific menu (and possibly a shortcut key or toolbar button).
- Place an action on the user-interface object itself or one or more objects/items listed in the grid in the Specific menu. For instance the action ‘Validate’ on the user-interface object Bill of Material.
- For more guidelines on labeling, ordering and grouping of actions in the Specific menu, refer to Chapter 5, ‘Window components’, Section ‘Menus’.

Group fields

- The Group fields must present the most important fields of the user-interface object in case the combined window is used to present one object. For example, the details of a Purchase Order are presented in the group area.

![Figure 14, Group fields used for object details](image1)

- The Group fields must present the group criteria in case the combined window is used to group a list of object/items by a particular criteria. For example, the list of contracts is grouped by Purchase Office.

![Figure 15, Group fields used for the criteria](image2)

- The Group fields must be positioned in the group area above the grid.
Do not use a group box around the Group fields.

**Column headings**

- All labels must be top-left aligned (independent of the alignment of the fields inside the columns).

![Table Example]

*Figure 16, Column headings*

- Refer to Chapter 6, ‘Window Controls’, Section ‘Column headings’ for more specific guidelines on column headings.

**Grid**

- Display only columns, that are relevant for the user's tasks. Put remaining fields in details window.

- For more guidelines on the grid and on field alignment, refer to Chapter 5, ‘Window components’, Section ‘Grid’.

**Form pages**

- Has no multiple form pages.

**Status bar**

- Refer to Chapter 5, ‘Window Components’, section ‘Status bar’ for more guidelines on the status bar.

**Components**

- Components included in the overview window:
  - Title bar, menus, toolbar, scroll bar, status bar, and grid

- For guidelines on these components, refer to Chapter 5, ‘Window components’.

- Components not included in the overview window:
  - Multiple form pages, and command buttons
Desktop manager

The desktop manager is a primary window that displays the user-interface objects in a windowed environment. In fact it provides customized representation of the objects in the tree structure of the menu browser. It is used to maintain groups of related objects.

The advantage is that the user does not have to browse through the multiple levels of the menu browser. The desktop manager provides direct access to the objects.

Figure 17, Desktop manager
When to use

- To group windows and objects to perform a task.
- To have quick access to your most used user-interface objects.
- To modify your workspace to suit your own needs.

Guidelines

View

- The desktop manager presents a container view on the user-interface object.
- Refer to Chapter 3, Basic concepts, section User-interface objects for more information on views.

Predefined menu bar

<table>
<thead>
<tr>
<th>Menu used in Desktop manager</th>
<th>Options</th>
<th>Clipboard</th>
</tr>
</thead>
<tbody>
<tr>
<td>File</td>
<td>Edit</td>
<td></td>
</tr>
<tr>
<td>New…</td>
<td>New Group…</td>
<td>Auto Arrange</td>
</tr>
<tr>
<td>Open…</td>
<td>New Item…</td>
<td>Minimize on Use</td>
</tr>
<tr>
<td>Save… CTRL+S</td>
<td>Properties…</td>
<td>Change Company…</td>
</tr>
<tr>
<td>Save As…</td>
<td>Delete</td>
<td>Defaults…</td>
</tr>
<tr>
<td>Delete</td>
<td>Copy</td>
<td></td>
</tr>
<tr>
<td>Properties</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Run Program</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Exit</td>
<td>ALT+F4</td>
<td></td>
</tr>
</tbody>
</table>

Application | Window | Help |
---|---|---|
Menu Browser | Cascade | Contents |
| | Tile | Using Desktop |
| | Arrange Icons | About BAAN |
| | Select Group… | |
Components not included in desktop manager:
- Multiple form pages, command buttons, grid, and toolbar

Related topics
- Menu browser
- Primary window

Details window
A details window (formerly single occurrence session) is a primary modeless window in which one user-interface object is shown. It provides more specific details of the selected object in the overview window or combined window. A details window can be either a display or an editable window.

When to use
- You can use a details window when you need to present detailed information about one single user-interface object (data view).

![Image of Details window]

Figure 18, Details window
Guidelines

Synchronization
- A details window of type 1 must be synchronized with an overview window of type 2.
- A details window of type 3 must be synchronized with a combined window of type 3. In this case the Group fields of the combined window are used in the details window and the value of these fields cannot be changed.
- Refer to Chapter 7, ‘Navigation’, section ‘Synchronization’ for more information on synchronization.

Size
- A details window must fit on an 800 x 600-resolution screen.
- No scrollable area.

View
- The details window presents a data view on the user-interface object.
- Refer to Chapter 3, 'Basic concepts', section 'User-interface objects' for more information on views.

Window title
- Make the title of the window reflect the type of user-interface object and the view it provides on the object.
- Title format for details window: <Object type> - <View>
  - The <Object type> must always be singular in a details window, because it always presents one single object. For example: Purchase Requisition.
  - No <View> has to be added in case the details window only presents one single data view on the object. Otherwise the default view has be defined as 'Details'. For example, Family - Details.
- Refer to Chapter 5, ‘Window Components’, Section ‘Title Bar’ for more guidelines on window titles.
Form pages

- Details windows can contain multiple form pages.
- Divide the fields into logical groups over the form pages.
- Include meaningful form-page labels. Make clear what kind of information can be found on the various form pages.

| General | Buying | Shipping | Invoicing | Paying | Changes |

- Refer to Chapter 5, ‘Window Components’, section ‘Form page’ for more guidelines on Form pages.

Key fields

- The Key fields must identify the active user-interface object.
- Position Key fields on the same place on each form page.
- Key fields on the first form page are editable and display on the following form pages.
- Separate Key fields and first fields/group boxes with a blank line (if enough space).
- Do not place Key fields inside a group box.

```
<table>
<thead>
<tr>
<th>BOM</th>
<th>Routing</th>
<th>Item Methods</th>
<th>Repetitive</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item</td>
<td></td>
<td></td>
<td>030650</td>
</tr>
<tr>
<td></td>
<td>Test JAB</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Search Key</td>
<td>TEST JAB</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bill of Material</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BOM Quantity</td>
<td></td>
<td>1</td>
<td>pcs</td>
</tr>
</tbody>
</table>
```

Figure 19, Key fields
Components

- Components included in the details window:
  - Title bar, and multiple form pages
- For guidelines on these components, refer to Chapter 5, ‘Window components’.
- Components not included in the details window:
  - Scroll bar, grid, menu bar, toolbar, status bar, and command buttons

Related topics

- Basic concepts
- Command buttons
- Form pages
- Labels
- Overview window
- Synchronization
- Title bar
- Window layout

Dialog box

A dialog box is a modal secondary window, which can be editable or display. In exceptional cases a dialog box can be modeless. A dialog box provides an exchange of information or dialog between the user and the application.

When to use

- To obtain additional information from the user needed to carry out a particular command or task.
- A dialog box is started after you active a command followed by ellipsis (…).

Guidelines

Window title

- Make the title of the window reflect the type of user-interface object and the action.
- Title format for dialog boxes: <Action> <Object type>.
  For example: Find Item
Command buttons

- When to use which Command buttons:

<table>
<thead>
<tr>
<th>Command button</th>
<th>When to use</th>
</tr>
</thead>
<tbody>
<tr>
<td>OK, Cancel, and sometimes Help</td>
<td>Used in an editable dialog box.</td>
</tr>
<tr>
<td>OK</td>
<td>Used in display dialog box.</td>
</tr>
<tr>
<td>OK</td>
<td>To apply the values in the dialog box and close the window</td>
</tr>
<tr>
<td>Cancel</td>
<td>to ignore the changes made by the user and to close the window. Canceling the operation the user chose.</td>
</tr>
</tbody>
</table>

- Refer to Chapter 6, ‘Window Controls’, section ‘Command buttons’ for more guidelines on command buttons.

Types of dialog boxes in Baan

Find dialog box

- Is a modal window.
- Command buttons: OK, Cancel.
- The find dialog box contains index information to make the connection between sorting and fields more clear to the user. When the user chooses another sorting sequence, the appropriate fields will appear on the dialog box.
- The option set with sorting possibilities is limited to six options. When there are more sortings defined, a drop-down list box must be used.
- If an exact match is found, the object must be highlighted.
- If no exact match is found then a message must be displayed in the status bar of the window from which the Find is activated.

![Figure 20, Searching on the first index in the Find dialog box](image)
Figure 21, Searching on the third index in the Find dialog box

Option dialog box
- Is a modal window.
- Command buttons: OK, Cancel, Save, Undo, and Help.

Filter dialog box
- Is a modeless window.

Figure 22, Filter dialog box

Make job dialog box
- Is a modal window.
- Command buttons: OK, Cancel, Save Defaults Get Defaults, New Job, and Help.
New job dialog box
- Is a modal window.
- Command buttons: Save, OK, Cancel, Undo, Find, Filter, New, and Help.

Components
- Components included in the dialog boxes:
  - Title bar, command buttons, and multiple form pages
- For guidelines on these components, refer to Chapter 5, ‘Window components’.
- Components not included in the dialog boxes:
  - Menu bar, toolbar, scroll bar, and grid

Related topics
- Browse area
- Command buttons
- Title bar
Generic Browser Framework

The Generic Browser Framework (GBF) is a window type for graphically depicting hierarchical structures. Figure 23 shows an example of a GBF window.

Like the overview window, the GBF provides a container view on a particular user interface object: it shows the data of the object as a collection of other objects. The main difference between the overview window and the GBF is that the former uses a grid to show all objects as a list, where the GBF uses a tree structure to show the objects.

The GBF depicts hierarchical structures as a tree of objects. To the user, there are two types of object: folders ( ), which contain other objects, and leaf objects ( ) that are at the end of a branch. Usually leaf objects represent user interface objects.

![Generic Browser Framework](image)

*Figure 23, An example of a hierarchical structure in a Generic Browser Framework.*

When to use

For what situations the GBF is more suitable than an overview window or a combined window, depends on the usability requirements for the window. These usability requirements should follow from the user’s task description and/or the market requirements.

Refer to Table 3 for the usability requirements relevant for deciding what window type to use, and the corresponding characteristics of each window type.
Figure 24 illustrates this table by showing the same user interface object in a combined window and in a GBF window.

In general, you should use the GBF when:

- You are dealing with hierarchical structures, such as parent-child relationships.

Do not use the GBF when:

- The session does not show data with hierarchical relationships.
- The session will be heavily used for entering data. Use an overview or combined window with an editable grid instead.
- Apart from a list, the session also needs to show other important information about the object that contains the list or tree.
  For instance, if the window shows a Bill of Material for a certain Item, particular information about this Item (other than the BOM itself) might be very important to the user. With the GBF it is not possible to show this information in the same window as the tree. If this information is important for a particular task, you will need to provide a combined window (possibly next to the GBF).

Note that you might decide to offer the user two windows for the same user interface object. One with a grid (for instance for easy data entry) and one with a tree structure (for instance for easy navigation).
When to use an overview/combined window or the GBF

<table>
<thead>
<tr>
<th>Usability Requirement</th>
<th>Overview window or combined window</th>
<th>Generic Browser Framework</th>
</tr>
</thead>
</table>
| Essential information on the objects in the list or tree | ![Table](image) | - **Serialized Item**: 04 (T-18654)  
  - All information on an object is shown as an icon and one single label.  
  - There is no description of the information on the objects. This makes it impossible to show much information.  
  - It is not possible to edit the information of an object directly in the tree. |
| Illustrative information on the objects in the list or tree | - It is not possible to use icons for objects.  
  - It is not possible to use colors for objects. | - It is possible to use different icons for different (types of) objects.  
  - It is possible to use different colors for different (types of) objects. |
| Other information than the list or tree (e.g., group fields). | - By using a combined window, it is possible to add labels and fields to the window. | - It is not possible to add any fields and labels to the window. |
| Depicting Hierarchies | - It is not possible to depict hierarchical structures. | - Shows hierarchical structures.  
  - Allows the user to navigate through the hierarchy. |

Table 3, When to use an overview or a combined window, or the GBF.
Guidelines

Compared to the combined and overview window, the GBF allows far more modifications to the user interface. To ensure consistency within all applications, please observe the following guidelines:

Title bar

- Make the title of the GBF of the format: <Object Type> or <Object Type> - <View Name>:
  - Use <Object Type> if the GBF is the only window in which you can open the object.
  - Use <Object Type> - <View Name> when apart from the GBF you can also open the object in an overview, combined or details window (such as for example the Physical Breakdown in Figure 24).

If there is only one GBF (one Tree View) for one particular object type, use “Tree” for the view name. For example:

- Do not show the session code or company number in the title bar.
Menu bar

- Use the standard menu structure of the GBF as much as possible, but remove the Group menu.

<table>
<thead>
<tr>
<th>File</th>
<th>Edit</th>
<th>Find</th>
<th>Sort</th>
<th>Specific</th>
<th>Help</th>
</tr>
</thead>
</table>

- Only show actions in the menu structure that will actually be used in the window. Remove all actions that will never be used.

- Disable an action when it is not applicable in a particular situation, for instance because the user has to select an object first. Also refer to the Baan GUI Style Guide (Verdi), Chapter 5, “Specific menu: disabling/enabling”.

<table>
<thead>
<tr>
<th>View Only Effective</th>
<th>Ctrl+F4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level Up</td>
<td>Ctrl+F5</td>
</tr>
</tbody>
</table>

- Add new menu items in the appropriate menu. Refer to Table 4 for the meanings of the different menus.

---

8 The GBF uses the actions in the Group menu for navigation through the tree, which is different from the way the Group menu is used in the combined window. To prevent confusion, do not use the Group menu in the GBF (navigation in the tree is already possible with mouse clicks or with the keyboard).
### The different menus of the GBF

<table>
<thead>
<tr>
<th>Menu title</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>File</td>
<td>Contains all actions on the entire tree (or the object that contains the tree).</td>
</tr>
<tr>
<td>Edit</td>
<td>Contains all actions on one or more selected objects in the tree. Note that “Undo”, “Cut”, “Copy”, and “Paste” are predefined by Microsoft Windows. If they are used within the GBF, they should have the same functionality. For more information, refer to The Window Interface Guidelines for Software Design, Microsoft Press, 1995.</td>
</tr>
<tr>
<td>Find</td>
<td>Contains all actions required to search for information in the tree.</td>
</tr>
<tr>
<td>Sort</td>
<td>Contains all actions related to sorting the objects in the tree.</td>
</tr>
<tr>
<td>Group</td>
<td>Contains some actions for navigation through the tree. The user can also do this by clicking on the tree, or using the keyboard. Do not use this menu in the GBF. The term ‘Group’ is already used for different functionality in the combined windows. Using the same terminology for two different types of functionality will confuse the user.</td>
</tr>
<tr>
<td>Specific</td>
<td>Contains all actions that are not part of the standard menus. Refer to the Baan GUI Style Guide (Verdi), Chapter 5, “Menus” for more guidelines on the Specific menu.</td>
</tr>
<tr>
<td>Help</td>
<td>Contains all actions that give access to the online documentation for the session.</td>
</tr>
</tbody>
</table>

Table 4. The different menus of the GBF.

- Always assign an action in the menu structure as the default action. This is the action that will be executed when the user double clicks on a leaf object in the tree structure. Preferably double clicking on a leaf object should open the leaf object in a details window or a combined window.

- Always add access keys to every menu title and menu item. Refer to the Baan GUI Style Guide (Verdi), Chapter 7, “Access Keys” for guidelines on assigning access keys.

- When adding non-standard actions to the menu structure, also consider giving it a shortcut key (for example, Ctrl+N is the shortcut key for New). Shortcut keys should only be assigned to frequently used actions. For more information on assigning shortcut keys, refer to the Baan GUI Style Guide (Verdi), Chapter 5, “Menus”.

---

Baan GUI Style Guide

4-27
Window types

Toolbar

- Use the standard toolbar of the GBF. To avoid confusion for the user, remove the buttons for First, Previous, Next, Last, New Group, First Group, Previous Group, Next Group, Last Group. The resulting standard toolbar will look like this:

![Standard Toolbar](image)

- If required, you can add buttons for other frequently used actions to the standard toolbar of the GBF (the GBF does not have a secondary toolbar). For icons for these buttons, please contact GUI Consultancy. They will help you get an icon with the correct meaning that matches Baan’s House Style.

- For guidelines on using the toolbar, refer to the Baan GUI Style Guide (Verdi), Chapter 5, “Toolbar”.

Tree

- If you make the labels of the tree static, make sure the meaning of the labels is clear. Use short labels. For instance, instead of:

  ![Tree Example 1](image)

  you can make the name of the folder “Cost Object – Budgeted Costs”, so that the labels of the objects can become:

  ![Tree Example 2](image)

  - If you fill the labels of the tree with fields from records from a table, use as few fields as possible. Only use those fields required to uniquely identify the object.

  - Avoid using more than 4 levels in one tree:

    ![Tree Example 3](image)
Window types

- Basically, this means that level 4 may only contain leaf objects, and no folders.
  Making the tree more than 4 levels deep will make it harder for the user to find his an object in the tree. If you need more than 4 levels, consider opening a new window for a particular part of the tree.

Colors

- Use colors to emphasize particular information in the tree. If you use a color to tell the user something, always make sure that the user will also get the message when he does not see the color. For example:
  
  - Configuration: gr (1) [Expired]
  - Item: 30200 [Future]
  - Here giving an object a particular color emphasizes its status: red for expired items, blue for future items. If the user is not able to see the coloring, he will still be able to tell the status of an object because the status is also mentioned in the object’s label.
  - Users may not be able to see a particular color when they use a black-and-white screen, change the color scheme of their Windows environment or are colorblind.
  - Always allow the user to make his own color-settings. Never hard-code colors. The meaning of a color may differ depending on the user’s culture and line of business. Apart from that, users may not be able to see a particular color.
  - In dialogs where the user can select a color, use MS Windows’ standard color selection dialog to pick a particular color.
  - When setting the background color and foreground color of the GBF, make sure to link these to the user’s system color scheme. If the user changes his color scheme in MS Windows, the colors of the GBF should change too.

Icons

You can use icons to distinguish between different types of objects in the tree.
By default the GBF provides icons for folders (opened: , closed: ) and leaf objects ( ).

- If you want to use other icons than those provided by the GBF, please contact GUI Consultancy. They will help you get an icon with the correct meaning that matches Baan’s House Style.
Status bar

- In the two left-most fields of the status bar, show the session code and the company number as in:
  
  tccon4100s900   Company: 570
  
- Minimize the amount of information in the status bar. Do not add obvious or irrelevant information (for example the current date).

- If required, you can use the status bar to show extra information about the object that is selected in the tree. Precede this information with a label that explains what the information shows. For example:

  Description: FRAME

Components

- Components available in the GBF are:
  - Title bar, menus, toolbar, scroll bar, and status bar.

- For more guidelines on these components, refer to Chapter 5, 'Window components'.

- Components not available in the GBF are:
  - Form pages, grid, and command buttons.

Related topics

- Access keys
- Basic concepts
- Combined window
- Menus
- Overview window
- Shortcut keys
- Toolbar buttons
- Title bar
Menu browser

The menu browser is a primary, display window, which represents the user-interface objects in a hierarchical way to the user, such as a tree control (menu structure).

![Menu browser](image)

**Figure 25, Menu browser**

**When to use**

- To view all the user-interface objects by packages.
- To activate an object which results in an overview window, a combined window or parameter window.
Guidelines

The menu browser consists of four levels: packages, module, business object, and user-interface object. These levels are presented by folders.

- **The menu browser is based on the object-action model.**
  The menu browser must only contain user-interface objects. All the actions and views must be removed from the menu browser and these must be linked to the related object. This also includes the printing and processing actions. Reason: In accordance with the object-action principle (Chapter 3).

- **Objects are strictly arranged by package.**
  For example, Logistic Tables (module tc mcs) are only in the menu structure of package tc. Manufacturing parameters are only in the menu structure of package ti. Reason: A clear view on decoupled packages.

- **Parameters by package in one menu, for all modules of the package.**
  The Parameter menu is the last menu at the first level of each package. No parameters are available by menu of module. Reasons: Easy-to-find parameters, fewer vague menus entitled miscellaneous, fewer sessions in other menus, and easier control of authorizations.

- **Avoid vague descriptions such as miscellaneous or general.**
  The folders and objects should have clear and meaningful descriptions.

- **Do not make the tree structures too deep.**
  It saves time in searching and clicking. Place no more than nine objects in one folder.

View

- The menu browser presents a container view on the user-interface object.

- Refer to Chapter 3, 'Basic concepts', section 'User-interface objects' for more information on views.
Predefined menu bar

Menu used in menu browser

<table>
<thead>
<tr>
<th>File</th>
<th>Find</th>
<th>Options</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>Run Program…</td>
<td>By Description…</td>
<td>Send to Desktop</td>
<td>Contents</td>
</tr>
<tr>
<td>Exit ALT+F4</td>
<td>By Code</td>
<td>Change Company…</td>
<td>Find Help On</td>
</tr>
<tr>
<td></td>
<td>First</td>
<td>Defaults…</td>
<td>Using Sessions</td>
</tr>
<tr>
<td></td>
<td>Previous</td>
<td></td>
<td>About Baan</td>
</tr>
<tr>
<td></td>
<td>Next</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Last</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Components

- Components included in the menu browser:
  - Title bar, menus, scroll bar, and status bar
- For guidelines on these components, refer to Chapter 5, ‘Window components’.
- Components not included in the menu browser:
  - Command buttons, toolbar, multiple form pages, and grid

Related topics

- Combined window
- Object-action
- Overview window
- Parameters window
- Primary window
- Printing and processing window
Message boxes

Messages are used to display information about a particular situation or condition. They occur during data entry, during the execution of processes, and also when opening objects or BaanERP. Messages are presented in a secondary window called a message box.

![Message box](image)

Figure 27, Message box

Modes

Messages can be presented in two ways:

- Interactive: In this mode a modal message box appears as soon as the situation arises. The user must explicitly close the message box before he can proceed with the task. This ensures that the user will notice the message, but it also interrupts the user. It is also rather easy to identify the field(s) that caused the situation.

- Non interrupting: In this mode the message is written to a separate, modeless window. In this case, the user is not distracted by messages during data entry. Since the user may not immediately notice the message, it can be difficult to identify the field(s) that caused the message.

A non interrupting mode does not mean an absence of interactive messages. In case the user must acknowledge the message immediately, the message is still presented in a modal dialog box (for example in case the user closes a window on which data has been changed, the save these changes message must be answered immediately).

Page-mode is a special case of the non interrupting mode. In page-mode the content of several controls is checked at once, instead of directly after the situation arises. The content of the controls is checked after the user passes a so-called sync-point (examples of sync-points: moving to another tab, starting a browse session, resizing the window, and using the scrollbar). This means several messages may appear at once in page-mode.
Currently the message-mode is set for each user in the user profile, so no adjustments in the application software have to be made dependent on the active message-mode. This means that the user always gets either interactive or noninterrupting messages.

**When to use**

You can use messages to:

- Inform the user about unexpected or undesirable situations and to help the user take care of the situations if possible/applicable.
- Indicate that a process has been completed successfully, but there is additional information about the status of the completion that the user must see.
Types of messages:

<table>
<thead>
<tr>
<th>Type</th>
<th>When to Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>Information message</td>
<td>To provide information about the results of a command, which the user must. Offers no choices to the user; the user acknowledge the messages by clicking the OK button. NB. Information messages can also be presented in the status bar in case the information is not essential to the task of the user.</td>
</tr>
<tr>
<td>Figure 29, Information message box</td>
<td></td>
</tr>
<tr>
<td>Warning message</td>
<td>To alert the user that an undesirable situation may occur (such as an action with destructive or irreversible consequences) or that a situation has occurred. The user must respond in order to continue or cancel the process. The message can be in the form of a question.</td>
</tr>
<tr>
<td>Figure 30, Warning message box</td>
<td></td>
</tr>
<tr>
<td>Critical message</td>
<td>To inform the user of a serious problem that requires intervention or correction before work can continue. Avoid this type as much as possible.</td>
</tr>
<tr>
<td>Figure 31, Critical message box</td>
<td></td>
</tr>
</tbody>
</table>
Interaction

- Present only one message box at a time. Combine the presented information of the different messages into one message.

Baan Guidelines

Message text

- Use terms the user understands. Avoid technical jargon or system-oriented information (such as; DAL, table, database and enum), unless the user is already familiar with those terms.

- Use the active voice if the user must act. Use the passive voice if the system/BaanERP has done anything. For example: "Enter a reason", instead of "A reason should be entered".
- Example of passive voice: "The order has been successfully processed".

- Use the imperative. Avoid using personal pronouns. For example: "Enter a reason for acceptance" instead of "You must enter a reason for acceptance".

- Avoid the past tense.

- Use punctuation marks when necessary. Always end a sentence with a full stop or question mark.

- Do not blame the user. For example: "BaanERP cannot find the specified business partner" instead of "You selected a non-existing business partner".

- Be brief, but complete. The maximum message text length is 132 characters.

- Provide an additional Help text if further explanation is necessary. The help text can be opened with a Help button on the message window.
- Limit a message to two or three lines. Use line breaks and blank lines to improve the readability of long messages. Use line breaks after a whole sentence. (A line break can be inserted by including ‘/n’ in the message text.)

- If the message informs the user of an error that the user can solve, the message should contain three parts. Not all parts are always applicable. It is often possible to combine the first two parts in the actual message:
  1. Problem/error description.
  2. The cause of the problem/error.
  3. The solution.

  For example:
  1. “Baan ERP cannot save the order …”
  2. “… because you have selected a business partner with the Prospect status.”
  3. “Select a Business Partner with the Active status, or change the status of the selected business partner.”

- Display a warning message if BaanERP can continue with the data provided but that there are certain restrictions (for example the user must supply some missing information later, or some options will not be available). Such messages consists of two parts:
  1. A statement of the situation.
  2. The resulting condition of restrictions.

  Example:
  1. You are defining reference designators for a phantom.
  2. They will not be shown on the material list of shop floor control.

- Be specific about the problem, the cause and the solution. For example:
  “Enter a date.”
  “This date format is not allowed.”
  “The date value is not allowed.”
  Do not use:
  “Invalid date” (too general)

- Be specific about the location/fields. The location can be:
  - The field the user is in. For example:
    “Enter an inventory unit.”
  - A field of the browse list that is linked to the current field. For example:
    “Destination warehouse must be of type Normal.”
  - One or more fields on a window that the user tries to save. For example:
    “If Price Calculation is selected, you must enter either the Item or the Item Group, or both.”
- One or more fields on a line of a record the user tries to save.
  Example: "The ledger accounts for the commission/rebate lines are not present for order line 10."

- Do not rely on default/generic messages provided by the standard program.
- Doubts? Ask a Knowledge Developer or GUI consultant to help you.

**Standard/common errors**

- Standard and common messages for standard problems.

<table>
<thead>
<tr>
<th>Standard and common messages</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Cause</strong></td>
</tr>
<tr>
<td>The user does not fill in a mandatory input field.</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>The user enters a value in a field that is not in a valid format.</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>The user enters a value in a field that is not allowed.</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>The user enters a value in a field that does not exist in the database.</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>The user enters a value in a field that exists in the database, but is not allowed.</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>The user selects ranges in a printing window or processing window that do not contain information.</td>
</tr>
</tbody>
</table>
Command buttons

Baan Tools provides two types of message boxes:

- Messages
- Questions

A Message message box can be used to present information messages, warning messages, and critical messages.

A Question message box can be used to present warnings and critical messages.

<table>
<thead>
<tr>
<th></th>
<th>Information</th>
<th>Warning</th>
<th>Critical</th>
</tr>
</thead>
<tbody>
<tr>
<td>Message messages</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Question messages</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

If a “Message” message box is used, only an OK button and Help button can be presented.

If a Question message box is used, several standard or even customized button combinations can be used.

- Use a Question message window to offer the user the possibility to solve the problem (by stating an appropriate message along with a matching button combination).
- Include command buttons that match the question or choices presented in the message.

<table>
<thead>
<tr>
<th>Default button combination</th>
<th>When to use examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>OK (– Help)</td>
<td>In case the user has no options</td>
</tr>
<tr>
<td>OK – Cancel (– Help)</td>
<td>In case the user wants to abort the action on behalf of the expected outcome</td>
</tr>
<tr>
<td>Abort – Retry – Ignore (– Help)</td>
<td>-</td>
</tr>
<tr>
<td>Retry – Cancel (– Help)</td>
<td>In case BaanERP could not start up due to network problems</td>
</tr>
<tr>
<td>Yes – No (– Help)</td>
<td>-</td>
</tr>
<tr>
<td>Yes – No – Cancel (– Help)</td>
<td>In case of a question whether changes need to be saved before closing</td>
</tr>
</tbody>
</table>

- The Default button must be the least destructive option.
- Do not change the order of the buttons of the default button combinations.
- Do not use the following button (combinations):
  - Close
  - OK – Close
  - Close – Cancel
  - OK – Close – Cancel.

**General guidelines**

- Use the Microsoft Windows messaging system if possible.

**Size and position**

- Position a message box in the middle and on top of the parent window.
- Adjust the size of the message box, according to the size (width and height) of the message text.

**Title bar text**

- Predefined title format used in BaanERP:
  `<Message number> - <Title of parent window>
  For example: ttstpq0109c - Items

- Use the title bar to identify the context in which the message is displayed; usually the name of the object and the action that caused the error.
  Preferred title format: `<Object type> - <Action or situation>`

- Do not use descriptive text for message box title text, such as warning, caution, or error.
- Refer to Chapter 5, ‘Window Components’, Section ‘Title Bar’ for more guidelines on window titles.
Graphical symbols

- Include a graphical symbol that indicates what type of message is being presented:

<table>
<thead>
<tr>
<th>Message type</th>
<th>Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>Information message</td>
<td>![i]</td>
</tr>
<tr>
<td>Warning message</td>
<td>![⚠️]</td>
</tr>
<tr>
<td>Critical message</td>
<td>![❌]</td>
</tr>
</tbody>
</table>

- Display the graphical symbol to the left of the message text.
- Do not use the Question mark icon.

Text/font

- Use the standard MS Windows font for message boxes.
- Left align the message text.

Buttons

- Center the buttons at the bottom of the message box.
- Include a Help button in case a Help text is available. The Help button must be positioned to the right of the other buttons.

Audio feedback

- Play the standard MS Windows sounds (for the different message types), when a message is presented.

Components

- Components included in the message boxes:
  - Title bar, and command buttons
- For guidelines on these components, refer to Chapter 5, ‘Window components’.
- Components not included in the message boxes:
  - Menu bar, toolbar, status bar, scroll bar, grid, and multiple formpages
Related topics

- Feedback
- Command buttons
- Status bar
- Software text standards
Overview window

An overview window is a modeless primary window in which multiple objects are shown to the user. In such a window the primary viewing and editing of objects takes place. The overview windows can be either display or editable.

When to use

- Use an overview window when you need to present information about a user-interface object in the form of a list or a tree (container view).
- For example: A list of RFQs is a user-interface object. This object is presented in an overview window, to show information of multiple RFQs in the form of list.

![Figure 32, The user-interface object 'List of RFQs', which is presented in an overview window.](image)

9 The overview window is the multi occurrence window without a group area (form type 2)
Guidelines

Synchronization

- An overview window (type 2) must only be synchronized with a details window of type 1.
- Refer to Chapter 7, ‘Navigation’, section ‘Synchronization’ for more information on synchronization.

Size

- An overview window must fit on an 800 x 600 resolution screen.
- Avoid horizontal scrolling.
- The window is resizable for the user.

Window title

- Make the title of the window reflect the type of user-interface object.
- Title format for overview window: <Object type>
  - The <Object type> must always be plural in an overview window, because it lists multiple objects/items. For example: Purchase Requisitions.
- Refer to Chapter 5, ‘Window Components’, section ‘Title Bar’ for more guidelines on window titles.

View

- The overview window presents a container view on the user-interface object.
- In a dynamic form, changing views on a user-interface object should not open a new window. The view should change within the same window.
- Refer to Chapter 3, 'Basic concepts', section 'User-interface objects' for more information on views.

Menu

- File-Open or double clicking on an object in the grid opens a details window in which more information of the object is shown.
- Open a container view or a combined view on an object in the grid by means of an action in the Specific menu (and possibly a shortcut key or button on the toolbar).

- Place an action on one or more objects from the list in the Specific menu. For instance, in a list of requisitions the action Convert must be placed in the Specific menu of the overview window Purchase requisitions.

- For more guidelines on labeling, ordering, and grouping of actions in the Specific menu, refer to Chapter 5, ‘Window components’, section ‘Menus’.

![Purchase Requisitions](image)

Figure 33, Specific menu in overview window

**Column headings**

- All labels must be top-left aligned (independent of the alignment of the fields inside the columns).

- Refer to Chapter 6, ‘Window Controls’, section ‘Column headings’ for more specific guidelines on column headings.

![Column headings](image)

Figure 34, Column headings
Grid
- Display only columns, which are relevant for the user’s tasks. Put remaining object fields in details window.
- For more guidelines on the grid and on field alignment, refer to Chapter 5, ‘Window components’, section ‘Grid’.

Form pages
- Has no multiple form pages.

Status bar
- Refer to Chapter 5, ‘Window Components’, section ‘Status bar’ for more guidelines on the status bar.

Components
- Components included in the overview window:
  - Title bar, menus, toolbar, scroll bar, status bar, and grid
- For guidelines on these components, refer to Chapter 5, ‘Window components’.
- Components not included in the overview window:
  - Multiple form pages, and command buttons

Related topics
- Access keys
- Basic concepts
- Column headings
- Details window
- Grid
- Generic browser framework (GBF)
- Menus
- Shortcut keys
- Status bar
- Synchronization
- Title bar
Parameters

The parameters settings are stored in a display overview window (type 2). The setting of parameters is done in a secondary, modal window (type 1).

Each Baan module has only one set of parameters.

The setting of parameters is not according to the object-action principle, because the parameters windows are directly started from the menu browser on package level (action-object), instead of first selecting the modules.

These types of windows are not synchronized with any other window.

When to use

- To view the parameter history.

- To define the parameters for each module.

Figure 35, History parameters
Guidelines

Size
- The parameter window must fit on an 800 x 600-resolution screen.

Window title
- Title format for parameter window: <Object type> "Parameters"
  - The <Object type> is often the module name. For example: Assembly Control.

- Refer to Chapter 5, ‘Window Components’, section ‘Title Bar’ for more guidelines on window titles.

Position in menu browser
- Directly started from the menu browser.
- Only one set of parameters for each module.
- Place the parameters of all modules of one package in one menu.
- This Parameter menu is the last one of the first level of each package.
**Command buttons**

- Predefined command buttons: Close, Save, Revert, Help
- Refer to Chapter 6, ‘Window Controls’, Section ‘Command buttons’ for more guidelines on command buttons.

**Components**

- Components included in the parameters window:
  - Title bar, command buttons, and multiple form pages
- For guidelines on these components, refer to Chapter 5, ‘Window components’.
- Components not included in the parameters window:
  - Menu bar, toolbar, scroll bar, status bar and grid

**Related topics**

- Command buttons
- Form pages
- Menu browser
- Overview window
- Title bar
- Window layout
Printing and Processing

The window, which is used for the printing or processing of objects, is a secondary window (type 4). According to the object-action principle this window must be started from another window and therefore it must be modal. This window may not be started from the menu browser.

When to use

- Printing window is used to enter data for printing objects. For example, Print RFQ Reminders which is activated from the File - Print menu of the RFQs overview window.

![Figure 38, Printing window](image)

- The processing window is used to enter data for processing objects. For example, Detect Loops in Bills of Material, which is activated from the Specific menu of the Bill of Material Combined window.

![Figure 39, Processing window](image)
Guidelines

Synchronization

- This window is not synchronized with other windows.

Size

- A Printing and Processing window must fit on an 800 x 600-resolution screen.
- This window is not resizable for the user.

Title bar

- Make the title of the window reflect the type of user-interface object and the action performed on the object.
- Title format for printing and processing window: <Action> <Object type>. For example,
  - Print Purchase Orders
  - Approve Purchase Orders
- Refer to Chapter 5, ‘Window Components’, section ‘Title Bar’ for more guidelines on window titles.

Command buttons

- Predefined command buttons:
  - Action such as Print, Delete and so on
  - Cancel
  - Save Defaults
  - Get Defaults
  - Make Job...
  - Help
- Refer to Chapter 6, ‘Window Controls’, section ‘Command buttons’ for more guidelines on command buttons.
Form pages

- The Printing and Processing window can contain multiple form pages.
- The first form page should contain the selection ranges for the data to be processed or printed. The form-page label is Selection.
- Second form page in printing windows often contains the Settings and Options group box. The form-page label is Options.

![Figure 40, Common form-page labels](image)

Refer to Chapter 5, ‘Window Components’, section ‘Form pages’ for more guidelines on form pages.

Sequence and kind of group boxes

- For more guidelines on group boxes refer to Chapter 6, ‘Window Controls’, Section ‘Group boxes’.

1 Print What

- Print What is represented by a drop-down list box.
- Print What may appear in two different types:
  - Print What without a group box which is used in normal printing windows:
    ![Figure 41, Print What](image)
  - Print What as group box label, which is a predefined group box used in the Print Parameter dialog box:
    ![Figure 42, 'Print What' group box used in Print Parameter dialog box](image)
2 Selection Range

- Group box with label Selection Range.
- Contains From-To columns and from-to rows.

![Selection Range Group Box](image1.png)

*Figure 43, Group box Selection Range*

- For more guidelines on selection ranges refer to Chapter 6, ‘Window Controls’, Section ‘Selection Ranges’.

3 Settings

- Group box with label Settings.
- Contains a label followed by an input field in which properties can be defined.

![Settings Group Box](image2.png)

*Figure 44, Group box Settings*

- Currency is always positioned in group box Settings.

![Settings with Currency](image3.png)

*Figure 45, Currency in group box Settings*

4 Attachments

- Group box with label Attachments.
Window types

- The options are check boxes.
- A check box is preceded by a label (such as text) and followed a description (such as the kind of text). For example, Text…[Check box] History.
- Attachment group box can be placed on other form.

![Figure 46, Group box Attachments](image)

5 Reports
- Group box with the label Reports.
- The options are check boxes, followed by a label.
- The Reports group box can be placed on another form.

![Figure 47, Group box Reports](image)

6 Options
- Group box with the label Options.
- The options are check boxes.
- Options depend on the kind of selection made.
- Options are usually placed on second form page with the label Options.

![Figure 48, Group box Options](image)
7 Sorted By
- Sorted By is not placed in a group box.
- The Sorted By option is represented using a drop-down list box.
- A drop-down list box is preceded by a Sorted-By label.

![Sorted By](image)

*Figure 49, Sorted By*

Exception to the sequence of group boxes
- If there is a functional relation between group boxes.
  For example, the Options group box may be placed before the Settings group box, if the Options determine which settings are available.

Components
- Components included in the Printing and Processing window:
  - Title bar, multiple form pages, and command buttons
- For guidelines on these components, refer to Chapter 5, ‘Window components’.
- Components not included in the Printing and Processing window:
  - Menu bar, toolbar, scroll bar, status bar, and grid

Related topics
- Check boxes
- Drop-down list box
- Form pages
- Group boxes
- Selection ranges
- Window layout
Progress indicator

A progress indicator is displayed in a modal secondary window that informs the user about the status of a process.

The progress indicator is a control that is used to show the percentage of completion of a lengthy operation. It consists of a rectangular bar that fills from left to right. The indicator should show the expected duration of the process.

![Figure 50, Progress Indicator](image)

When to use

- If a process lasts more than 5 seconds, display a progress indicator to provide visual feedback to the user about the duration of the process.

Guidelines

Title bar

- Make the title of the window reflect the type of user-interface object and the action performed on the object.
- Title format for Progress Indicator: `<Action> "Progress" - <Object Type>`. For example:
  - Print Progress - Bills of Material

Refer to Chapter 5, ‘Window Components’, Section ‘Title Bar’ for more guidelines on window titles.
Text

- Provide static text or information to help communicate the purpose of the progress indicator.
- Place the text above the progress indicator bar.

Command buttons

- Provide a command button that allows the user to end the process and close the window, such as Stop. However, Stop must be avoided in processing windows. Also, clear feedback must be provided to the user about what has already been processed.
- Provide a command button that allows the user to undo a process and close the window, such as Cancel. However, use Cancel only in cases when a good reliable roll back is provided.
- Place the command button under the progress indicator bar.
- If a process takes less than 5 seconds, than provide another kind of feedback to the user. For example, a message displayed in the status bar.
- Refer to Chapter 6, ‘Window Controls’, section ‘Command buttons’ for more guidelines on command buttons.

Components

- Components included in the progress indicator:
  - Title bar, and command buttons
- For guidelines on these components, refer to Chapter 5, ‘Window components’.
- Components not included in the progress indicator:
  - Menu bar, toolbar, status bar, scroll bar, grid, and multiple form pages

Related topics

- Command buttons
- Feedback
- Title bar
- Software text standards
Wizards

A wizard is a special way of providing assistance to the user. A wizard automates a task for the user with the use of several dialog boxes. Wizards are meant for complex tasks or tasks that are not done on a regular basis.

**When to use**

- A wizard must not be the only way to perform a task. Experienced users may find a wizard to be inefficient or not provide them with sufficient access to all functionality.
- Normal sessions for the same task should be available for the user next to the wizard.
Use a wizard for situations in which the user has to perform a complex task. You can recognize a complex task by (among others):

- The amount of sessions required to complete the task. If a task requires more than two sessions, it tends to become complex.
- Sessions with mandatory input fields, other than the key fields.
- Sessions with input fields that are related to other fields. For example, the input fields Date, Range, and Compare in Figure 52 are all related to the field Parameters, because the value in this field determines which of the other fields is enabled. When the related fields are not on the same form page, the session easily becomes complex.

Examples of complex tasks are:

- Setting parameters.
- Using batch processes.
- Using import/export functionality.
- Installing a new software package.

Guidelines

Design steps

Design a wizard following these steps:

1. Identify the task in cooperation with a GUI Consultant.
2. Break down the task into steps and put them in a flow. This flow is the framework for the wizard. See, for example Figure 53.
3. Make questions for every step in the wizard. Consult a GUI Consultant on this point.
4. Choose the controls with which the user can answer the questions.
5 Make paper designs of the forms. Include descriptions of the behavior of the wizard (disabling/enabling of controls, mandatory fields, and so on).

6 Have the wizard (paper version) reviewed by a GUI Consultant.

7 Start programming the wizard.

Figure 53, An example of the framework of a wizard. Each step corresponds to one page.

**Wizard title**

- The title bar must briefly describe the purpose of the wizard.
- Predefined Wizard title format: `<action> <object type> "Wizard"`. For example, New Project Wizard or Install BaanERP Wizard.

**Wizard forms**

- The first page of the wizard must describe the objective of the wizard.
- On the last page, instruct the user that the wizard has enough information to complete the task. Include the Finish button on this page.
- Between the first and the last page, the user should get at least one, but not more than eight other pages.
- Do not use more than two questions on a single page.
- Make the wizard about 600 pixels wide and about 350 pixels high.

**Command buttons**
- On the bottom of each page, place the buttons Cancel, Back, Next, and Finish, as shown in Figure 54.
- Only enable the Finish button on the last page of the wizard.

![Image of buttons](image)

*Figure 54, The buttons on a page of a wizard.*

- Refer to Chapter 6, ‘Window Controls’, section ‘Command buttons’ for more guidelines on command buttons.

**Navigation**
- Do not let pages advance automatically; the user has to be in control.
- It is not allowed to open any other modeless BaanERP sessions from within the wizard.
- It is allowed to have the user open a browse list from an input field.

**Starting a wizard**
- Do not start a wizard automatically. The user should decide whether to use the wizard or the standard sessions.
- In an Overview window, use a menu item in the Specific menu (and possibly a toolbar button) to start a wizard.
- In a Details window, use a command button to start a wizard.
- Make it clear to the user that there is a choice between the wizard and the standard session. For example, if a menu has an action Return Tools, place the action Return Tools Wizard after it in the same menu.

**Language**
- Be sparse with text. Do not use any more text than necessary.
- Write the text in the wizard with someone who is not the developer, for instance a documentation engineer or a GUI Consultant.
- Use conversational language instead of instructional language.
- Refer to the user as you.
- Be consistent in the use of terms within the wizard.
- Use short terms, write clear language and short sentences
- Do not use passive sentences.
- Do not use future tense.
- Do not use negations.
- Do not use technical or programming terminology. Use the user’s language instead.

**Design**
- A wizard must be easy to understand. The user should immediately understand the purpose of the wizard. Make a wizard as self-evident as possible; the user will not read all instructions in the wizard carefully.
- Use the left part of each page (about 1/3 of the form) to provide the user with information that helps answer the questions on the page.
- Use the right part of each page (about 2/3 of the form) for the questions and the controls.

**Components**
- Avoid related fields in a wizard (as in Figure 52). If you do use them, place the related fields on the same page.

**Window type**
- Use a session of type Wizard.
- Make the session a main session.
- The session code for a wizard has the format: ppmmmxxw000.

**Form**
- The form should be of type 4.
- All the fields on the wizard must be form fields. The domain of the form fields should be the same as the domain of the corresponding table fields (in order to ensure domain checks).
Use the following standard options for the wizard:
- First Form: Exit Session, Edit Records
- Last Form: Exit Session, Save Data and Exit, Edit Records
- Other Forms: Exit Session, Edit Records

Script
- All the checks, for example property checks, check for duplicate records, and reference checks should be done explicitly.

Related topics
- Command buttons
- Software text standards
- For more detailed information on wizards, refer to *The Development Guide ActiveApps (D0330A US).*
This section describes a number of interactive elements, which are components of windows. The following components are included in alphabetical order:

**Form page**

```
<table>
<thead>
<tr>
<th>General</th>
<th>Buying</th>
<th>Shipping</th>
<th>Invoicing</th>
<th>Paying</th>
<th>Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Buy-from BP</td>
<td>1 PARTNER</td>
<td>Test Business Partner</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ship-from BP</td>
<td>1 PARTNER</td>
<td>Test Business Partner</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Purchase Office</td>
<td>002</td>
<td>arun's P.O.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

*Figure 55, Multiple form pages*

**Grid**

```
<table>
<thead>
<tr>
<th>RFQ</th>
<th>RFQ Date</th>
<th>Return Date</th>
<th>Buyer</th>
</tr>
</thead>
<tbody>
<tr>
<td>VIN000052</td>
<td>17-12-97</td>
<td>27-12-97</td>
<td></td>
</tr>
<tr>
<td>VIN000053</td>
<td>17-12-97</td>
<td>27-12-97</td>
<td></td>
</tr>
<tr>
<td>VIN000054</td>
<td>17-12-97</td>
<td>27-12-97</td>
<td></td>
</tr>
<tr>
<td>VIN000055</td>
<td>17-12-97</td>
<td>27-12-97</td>
<td></td>
</tr>
</tbody>
</table>
```

*Figure 56, Grid with display appearance*

**Menus**

```
File  Edit  View  Group  Workflow  Tools  Specific  Window  Help
```

*Figure 57, Menu bar*

**Scroll bar**

```
< | >
```

*Figure 58, Horizontal scrollbar*
Form page

Form pages are the components of windows on which the controls are placed. Form pages determine how the information is presented to the user in the windows (the look of the BAAN user interface).

<table>
<thead>
<tr>
<th>General</th>
<th>Buying</th>
<th>Shipping</th>
<th>Invoicing</th>
<th>Paying</th>
<th>Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Buy-from BP</td>
<td>TPARTNER</td>
<td>Test Business Partner</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ship-from BP</td>
<td>TPARTNER</td>
<td>Test Business Partner</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Purchase Office</td>
<td>002</td>
<td>arun's P.O.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

When to use

- Used in secondary windows and details window to define the controls.
- Used in overview window, combined window, and browse list to define the group fields and column headings.
Guidelines

Alignment

- Position the first label at row 4 and column 4 on the form page.
- Position the left group box border at column 2 on the form page.

![Selection Range](image)

Figure 63, Alignment of controls on the form page

Multiple form pages

- Only allowed in secondary windows.
- Used if the number of items exceed one form page.
- All the form pages in one window have the same size and box attributes.
- Divide the items in functional groups over the multiple form pages.
- Provide a label for each form page that is meaningful to the user.
- Place the key fields at the same position across the multiple form pages.
- Group boxes placed on the multiple form pages have equal width.

Navigation keys

- Previous form page: CTRL+SHIFT+TAB
- Next form page: CTRL+TAB

---

10 Not applicable for development with the Dynamic Form Editor.
Windows components

- Make sure the cursor is placed in the first enabled control of a From page when the user switches to it by pressing one of these navigation keys.

Related topics
- Group boxes
- Labels
- Tab control
- Window layout

Grid

The grid is the table of objects/items in an overview window, combined window, or browse list. In the rows, the fields of one object are displayed. The columns contain the mutual fields of multiple objects. The grid can be editable or display.

- In an editable grid the user can enter values of an object/item directly in the grid itself.

<table>
<thead>
<tr>
<th>Level</th>
<th>Seq.</th>
<th>Effective Date</th>
<th>Expiration Date</th>
<th>Metric Def</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>30</td>
<td>09/12/97</td>
<td>00:00:00</td>
<td>S-2</td>
</tr>
<tr>
<td>None</td>
<td>30</td>
<td>12/12/97</td>
<td>00:00:00</td>
<td>S-1</td>
</tr>
<tr>
<td>None</td>
<td>30</td>
<td>18/12/97</td>
<td>00:00:00</td>
<td>S-1</td>
</tr>
<tr>
<td>None</td>
<td>30</td>
<td>03/01/08</td>
<td>00:00:00</td>
<td>YA5</td>
</tr>
<tr>
<td>None</td>
<td>30</td>
<td>27/11/98</td>
<td>00:00:00</td>
<td></td>
</tr>
</tbody>
</table>

*Figure 64, Grid with editable appearance*

- In a display grid the user can only manipulate values of an item/object by double clicking on a row which opens its related details window.

<table>
<thead>
<tr>
<th>RFQ</th>
<th>RFQ Date</th>
<th>Return Date</th>
<th>Buyer</th>
</tr>
</thead>
<tbody>
<tr>
<td>VIN000052</td>
<td>17-12-97</td>
<td>27-12-97</td>
<td></td>
</tr>
<tr>
<td>VIN000053</td>
<td>17-12-97</td>
<td>27-12-97</td>
<td></td>
</tr>
<tr>
<td>VIN000054</td>
<td>17-12-97</td>
<td>27-12-97</td>
<td></td>
</tr>
<tr>
<td>VIN000055</td>
<td>17-12-97</td>
<td>27-12-97</td>
<td></td>
</tr>
</tbody>
</table>

*Figure 65, Grid with display appearance*

When to use

- Used in overview windows
- Used in combined windows
- Used in browse lists
Guidelines

- The grid is positioned under the group fields (if any).
- The columns of the grid are resizable by the user.
- The column headings, which identify the contents of each column are positioned above the columns.
- The maximum number of objects/items visible to the user is thirteen (rows).
- Provide feedback about which column or row is selected by the user.
- Provide a horizontal scroll bar if the grid is not fully visible in the window.
- The fields in the grid are left aligned. The only exception to this is numeric fields in a column, which are right aligned.

### Field alignment

<table>
<thead>
<tr>
<th>Field type</th>
<th>Alignment</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alphabetical</td>
<td>Left</td>
<td>Hello</td>
</tr>
<tr>
<td>Alphanumeric</td>
<td>Left</td>
<td>AA123</td>
</tr>
<tr>
<td>Date fields</td>
<td>Left</td>
<td>30-10-98</td>
</tr>
<tr>
<td>Numeric</td>
<td>Right</td>
<td>1234567</td>
</tr>
</tbody>
</table>

Related guidelines

- Refer to Chapter 6, 'Window Controls', section 'Column heading' for more guidelines on the alignment and positioning of the column heading labels.

Related topics

- Browse list
- Column headings
- Feedback
- Overview window
- Group fields
Menus

A menu is a list of commands presented to the user. The user can make a menu appear from a menu bar, by means of the secondary mouse button or with a menu button.

- A menu bar is the area between the title bar and the toolbar that provides access to the different drop-down menus in the window. Each item on the menu bar is a single word, called a menu title.

![Menu bar of overview window](image)

- The menu that appears when the user clicks on the menu bar is called a drop-down menu.

![Drop-down menu](image)

- A submenu is a menu that appears from a higher-level menu. Submenus are used to reduce the length of menus and contain a set of menu items that are related to the higher-level menu.

![Submenu](image)

- A menu item is a command that appears in a menu.
When to use

- Use a menu to present actions or commands that the user needs while performing tasks.
- In primary windows, menus can be accessed by means of a menu bar or with the secondary mouse button.
- In secondary windows, menus can be accessed by means of a menu button or the secondary mouse button.
- Only use submenus to reduce the length of a menu.

Guidelines

General

- Place related commands (menu items) together in a menu.
- In the overview and combined window, place all actions other than the standard actions in the Specific menu.
- Provide unique access keys for each command in a menu.
- Provide unique shortcut keys for each frequently used command in a drop-down menu.
- Use separators to distinguish groups of related commands.
- Allowed number of commands:

<table>
<thead>
<tr>
<th>Type of menu</th>
<th>Minimum</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Menu bar</td>
<td></td>
<td>10</td>
</tr>
<tr>
<td>Drop-down menu</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>Cascading</td>
<td>2</td>
<td>5</td>
</tr>
</tbody>
</table>

Menu bar

- The number of commands must not exceed ten choices.
- Use single words for menu titles.
- Capitalize the first letter of the menu title.
- Provide unique access keys for all menu bar choices.
- Do not use ellipsis or arrows after menu titles.
Drop-down menu
■ The number of commands must not exceed twelve choices.
■ Place related commands (menu items) together in one drop-down menu.
■ Keep the relative order of identical commands in the menus the same among different windows.

Submenu
■ Only use submenus to reduce the length of a menu. Collect functionally related components and put these in a submenu:
  – If more than one menu item has the same verb/noun as starting word, a submenu can be created.
  – Pay attention to the verb/noun in the primary menu because it must represent all items in the submenu.
■ The maximum number of commands in a submenu is five.
■ Only one level of submenus is allowed.
■ An arrow pointed to the right after the menu item indicates a submenu.
■ Separators and shortcut keys are not possible in submenus.
■ Do not repeat the routing label in the submenu label.
□ Possible submenus in BaanERP: Print, Sort By, and Text, as well as in the Specific menu.
■ Use submenu only when absolutely necessary as they add complexity to the interface.
Specific menu: grouping and sequencing

- Place all specific commands (menu items) in the Specific menu of the overview window and combined window.
- Label a command, through which a hierarchical structure (for example a GBF-tree) is depicted: "View Tree".
- The menu items must be grouped, ordered, and labeled according to the following standards:

<table>
<thead>
<tr>
<th>Order</th>
<th>Action Description</th>
<th>Label Format of Menu-Item</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>View on the selected grid-item</td>
<td><code>&lt;View&gt;</code></td>
</tr>
<tr>
<td>2</td>
<td>Range action on item(s) in grid</td>
<td>Overview window: <code>&lt;action&gt;</code> &quot;...&quot; (do not mention the grid items) Combined window: <code>&lt;action&gt; &lt;grid items&gt;</code> &quot;...&quot;</td>
</tr>
<tr>
<td></td>
<td>Action on entire object</td>
<td><code>&lt;action&gt;</code> (do not mention the object)</td>
</tr>
<tr>
<td>3</td>
<td>View on entire object</td>
<td>&quot;Group By&quot; <code>&lt;grouping attribute&gt;</code> &quot;View&quot; <code>&lt;view&gt;</code></td>
</tr>
</tbody>
</table>

---

11 The items in the grid be identified to indicate that the action is applicable to the item(s) in the grid and not to the entire object.

12 Mind ellipsis if an additional dialog is presented to complete the action.
### Example for overview window: Request for Quotations

<table>
<thead>
<tr>
<th>Order</th>
<th>Action</th>
<th>Label of menu item</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>View on the selected grid-item</td>
<td>Suppliers</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The suppliers of the selected RFQ will be displayed.</td>
</tr>
<tr>
<td>2</td>
<td>Range action on item(s) in grid</td>
<td>Evaluate...</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The selected RFQ's will be evaluated.</td>
</tr>
<tr>
<td></td>
<td>Action on entire object</td>
<td>Action on the entire list of objects hardly occurs in BaanERP.</td>
</tr>
<tr>
<td>3</td>
<td>View on entire object</td>
<td>Group by Supplier</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The RFQ's will be grouped by supplier.</td>
</tr>
</tbody>
</table>

### Example for combined window: Bills of Material

<table>
<thead>
<tr>
<th>Order</th>
<th>Action</th>
<th>Label of menu-item</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>View on the selected grid-item</td>
<td>Reference Designators</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The reference designator for the selected BOM-line will be displayed.</td>
</tr>
<tr>
<td>2</td>
<td>Range action on item(s) in grid</td>
<td>Replace Item...</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The selected BOM-line will be replaced.</td>
</tr>
<tr>
<td></td>
<td>Action on entire object</td>
<td>Validate...</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The active BOM will be validated.</td>
</tr>
<tr>
<td>3</td>
<td>View on entire object</td>
<td>View End Items</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The End items of the active BOM will be displayed.</td>
</tr>
</tbody>
</table>

- Since 'Views on the selected grid-item' and 'Actions' are the most frequently used these two groups should be placed on top, followed by 'Views on the entire object'. This is the general guideline to group menu items.

13 "Group by Supplier" will open the combined window "Request for Quotations - by Supplier". "Group By" is considered as a special view on the entire list "Request for Questions", although only one Supplier is shown at the time.
- Use separators between the groupings. Make sure that the grouping breaks up the menu visually so that it becomes easier to locate menu items.
- Within groups the sequence of menu-items should be based on:
  - Frequency of use
  - Task / Process flow
- Common sense should be used when grouping and sequencing the menu items. The above mentioned groupings might be overruled in case frequency or task/process gives reason for another grouping or sequence. However, this should be approved by a GUI Consultant.

![Purchase Requisitions](image)

*Figure 69. Example of the grouping, sequencing, and labeling of the menu items in the Specific menu of an overview window*

**Specific menu: disabling/enabling**

- Disable an menu item in the Specific menu in case:
  - The action applies to one or more items in the grid, but no item is selected in the grid.
  - The action is not applicable for the selected item in the grid. For example because of the item’s status.
For performance reasons, only check the status within one Business Object or Module. Always enable the action in case a status check takes place within another Business Object or Module.

- Enable an menu item in the Specific menu in case the action starts a batch session.

**Menu labels**

- Define unique item names for each menu.
- Avoid lengthy names. Preferably use no more than three words (do not consider conjunctions such as and, to, and so on, as separate words).
- Do not use acronyms, abbreviations, and shortened forms, when you name the menu item. For example for Service Center, do not put Serv. Center.
- Only acronyms that are prevalent in the industry such as EDI (Electronic Data Interchange), BOM (Bill of Material) and so on, are allowed.
- For English-language versions, capitalize the first letter of every word, except for articles, conjunctions, and prepositions that occur other than at the beginning or end of a phrase.
  - For example: New, Save Defaults, Switch to Details, Group By.

**Print menu**

- Do not repeat the word Print in the cascading Print menu.

  ![Print Menu Example](image)

- Only mention the items to be printed. Format of menu label: `<grid items>`

  ![Print Menu Label Example](image)

**Shortcut keys**

- Define shortcut keys for frequently used and repetitive commands. Do not provide shortcut keys to all the items in the Specific Menus, especially if the menu is large.
- Provide the same shortcut key to an action throughout its scope.
Available shortcut keys are:
- CTRL+F1 through CTRL+F10
- CTRL+SHIFT+A through CTRL+SHIFT+Z

Do not use CTRL+%, CTRL+#, CTRL+I and CTRL+SHIFT+I.

Use the SHIFT key if a command, which is carried out by using a shortcut key, must be reversed.
- For example: Ctrl+F1 is for Filter Calls, then Shift+Ctrl+F1 is for Undo filtering.

Reversing is not possible with shortcut keys that have alphabetic characters.

Refer to chapter 7 'Interaction and navigation', section 'Shortcut Keys' for more information.

Access keys

- Provide an access key for each menu item.
- Refer to chapter 7 'Interaction and navigation', section 'Access keys' for more information.

Ellipsis

- Add an ellipsis (…) to the label to indicate that the user must give further information to complete the action.
  - For example: Save As… has ellipsis because the user must provide a name before the action can be performed.

Do not use an ellipsis with a command that carries out action immediately (as opposed to displaying a dialog box that requests more information).
- For example: Save does not require an ellipsis since the save operation is performed immediately on choosing the Save command.

Option button

- Use an option button before a command in a drop-down menu to indicate which option is applied to an object.
- Only for mutually exclusive menu items.
- Often used for View commands, such as Sort By.

Separator

- Use a separator to visually separate menu items into distinct groups in the menus.

- In the Specific menu, use separators to group:
  - Action on entire object
  - View on entire object
  - Range action on item(s) in grid
  - View on the selected grid item

- Use a line as separator.
Disabling

- Disable a menu item that cannot be selected in the current context.

```
File  Edit  View  Group
New   Ctrl+N
Open  Ctrl+O
Open Read-Only
Save  Ctrl+S
```

- Do not disable commands that lead to drop-down menus or submenus.
- Always display the menu even if all the commands in it are not available in a certain context.

```
Workflow  Tools
Cancel   Suspend
Reject   Next
Current
```

- Remove a menu item from the menu when the user is not authorized to perform the action.
- The disabling and enabling of form commands will be taken care of by the tools. This can be specified in the Command Availability field while maintaining the form command itself.

Predefined menus in overview and combined window

<table>
<thead>
<tr>
<th>File</th>
<th>Edit</th>
<th>View</th>
</tr>
</thead>
<tbody>
<tr>
<td>New</td>
<td>Revert</td>
<td>Sort By</td>
</tr>
<tr>
<td>Open</td>
<td>Ctrl+D</td>
<td>Ctrl+F Filter</td>
</tr>
<tr>
<td>Open Read-Only</td>
<td>Duplicate</td>
<td>Refresh F5</td>
</tr>
<tr>
<td>Save</td>
<td>Delete</td>
<td>Ctrl+Delete</td>
</tr>
<tr>
<td>Print</td>
<td>Ctrl+P</td>
<td>Delete Range</td>
</tr>
<tr>
<td>Send to Desktop</td>
<td>Text</td>
<td>Ctrl+T</td>
</tr>
<tr>
<td>Exit</td>
<td>Alt+F4</td>
<td>Attachments</td>
</tr>
</tbody>
</table>
### Windows components

<table>
<thead>
<tr>
<th>Group</th>
<th>Workflow</th>
<th>Tools</th>
</tr>
</thead>
<tbody>
<tr>
<td>New</td>
<td>Ctrl+G</td>
<td>Cancel</td>
</tr>
<tr>
<td>Find</td>
<td>Ctrl+Page Up</td>
<td>Suspend</td>
</tr>
<tr>
<td>Duplicate</td>
<td>Ctrl+Page Up</td>
<td>Rejected</td>
</tr>
<tr>
<td>First</td>
<td>Ctrl+Page Up</td>
<td>Next</td>
</tr>
<tr>
<td>Previous</td>
<td>Ctrl+Page Up</td>
<td>Current</td>
</tr>
<tr>
<td>Next</td>
<td>Ctrl+Page Up</td>
<td>Options…</td>
</tr>
<tr>
<td>Last</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Specific</th>
<th>Window</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action on entire object</td>
<td>Save Defaults</td>
<td>Help Topics</td>
</tr>
<tr>
<td>View on entire object</td>
<td>Get Defaults</td>
<td>Current Session F1</td>
</tr>
<tr>
<td>Action on grid item(s)</td>
<td>Freeze Columns</td>
<td>Processes</td>
</tr>
<tr>
<td>View on grid item</td>
<td>Unfreeze Columns</td>
<td>Reference</td>
</tr>
<tr>
<td>Switch to Details</td>
<td>Alt+F7</td>
<td>Glossary</td>
</tr>
<tr>
<td>Validate</td>
<td></td>
<td>Using Help</td>
</tr>
</tbody>
</table>

#### Related topics

- Access keys
- Combined window
- Generic browser framework (GBF)
- Menu button
- Overview window
- Shortcut keys
Scroll bar

A scroll bar is a window component, associated with a scrollable area, that indicates to a user that more information is available or can be added in a particular direction and can be scrolled into view. Scroll bars can be horizontal and vertical.

The components of a scroll bar are:
- Scroll arrows
- Scroll box
- Scroll bar shaft

When to use

- Provide a scroll bar if the information is not fully visible in a window in a particular direction.
- Provide a scroll bar if the information in a window can be extended to a size that is larger than the window.
- Used in overview window, combined window, and browse list.
- Not used in details window and secondary windows.
Guidelines

- Provide a scroll bar along the dimension in which more information is available.
- Overview window, combined window, and browse list usually provide a vertical scrollbar for scrolling through the table of objects.
- Avoid horizontal scrolling in overview window, combined window, and browse list.

Scrolling actions

The BAAN scroll bar offers the following facilities:

- Scroll one record up/down by clicking the up/down scroll buttons.
- Scroll one screen up/down by clicking in the shaft above or below the scroll box.
- Scroll to beginning/end of table by dragging the scroll box to the top or bottom of the scroll bar.

Navigation keys

- Use navigation keys to support scrolling with the keyboard.
- Page Up: Scrolling one screen up.
- Page Down: Scrolling one screen down.
- CTRL+Home: Scrolling to the beginning of the table.
- CTRL+End: Scrolling to the end of the table.

Related topics

- Browse list
- Combined window
- Desktop manager
- Menu browser
- Overview window
Status bar

A status bar is a part of a window that shows the current state of what is being viewed in the window or message on a successfully completed action. The status bar consists of two parts: the message area and the status field.

- The message area is the left part of the status bar where information appears about the object or choice that the cursor is on. Information about the normal completion of a process can also appear in the information area.

  ![Figure 72, Status bar]

- The status fields are placed on the right side of the status bar. They show the state of an object, or the state of a particular view of an object.

  ![Figure 73, Message area]

When to use

- Used in primary windows.
- To provide information about the current state of an object or the status of a process.

Guidelines

- Place the status bar at the bottom of the window.
- Contains read-only information or noninteractive information.

Message area

- Show information (whenever available) and messages that do not need to be confirmed.

Status fields

- Display the status of the process such as modify or add.
Title bar

The title bar is the area at the top of each window that contains the BAAN symbol, a session code\(^\text{14}\), a window title, a company code and the Close, Minimize, Maximize, Restore buttons.

![Title bar of combined window](image)

*Figure 75, Title bar of combined window*

When to Use

- To identify what is displayed in the window.
- To display the session code and the company code.
- To serve as a control point for moving the window.
- Is used in primary windows and secondary windows.

Guidelines

- Active and inactive windows have different title bar colors to identify which window has the input focus.

![Title bar of active window](image)

*Figure 76, Title bar of active window*

![Title bar of inactive window](image)

*Figure 77, Title bar of inactive window*

\(^{14}\) The session code and company code will be removed from the title bar and positioned in the status fields of the status bar (probably Vivaldi). For this reason all the figures in this guide do not depict a session code or company code in the title bar.
Title bar text

- The title must be meaningful to the user.
- The title must identify the information presented in the window.
- The title is left-aligned.
- The title format depends on the type of window:

<table>
<thead>
<tr>
<th>Window type</th>
<th>Predefined window title format</th>
</tr>
</thead>
<tbody>
<tr>
<td>Browse list</td>
<td><code>&lt;Grid items&gt;</code></td>
</tr>
<tr>
<td></td>
<td><em>For example: Languages</em></td>
</tr>
<tr>
<td>Combined window</td>
<td><code>&lt;Object type&gt; - &lt;View&gt;</code></td>
</tr>
<tr>
<td></td>
<td><em>For example: Purchase Contract - Lines</em></td>
</tr>
<tr>
<td></td>
<td><em>Contracts - by Purchase Office (grouping)</em></td>
</tr>
<tr>
<td>Details window</td>
<td><code>&lt;Object type&gt; - &lt;View&gt;</code></td>
</tr>
<tr>
<td></td>
<td><em>For example: Family - Details</em></td>
</tr>
<tr>
<td>Dialog box</td>
<td><code>&lt;Action&gt; &lt;Object type&gt;</code></td>
</tr>
<tr>
<td></td>
<td><em>For example: Find Items</em></td>
</tr>
<tr>
<td>Generic Browser (GBF)</td>
<td><code>&lt;Object Type&gt; - &lt;View&gt;</code></td>
</tr>
<tr>
<td></td>
<td><em>For example: Physical Breakdown - Tree</em></td>
</tr>
<tr>
<td>Message box</td>
<td><code>&lt;Message number&gt; - &lt;Title of parent window&gt;</code></td>
</tr>
<tr>
<td></td>
<td><em>For example: ttstpq0109c - Items</em></td>
</tr>
<tr>
<td>Overview window</td>
<td><code>&lt;Object type&gt;</code></td>
</tr>
<tr>
<td></td>
<td><em>For example: Projects</em></td>
</tr>
<tr>
<td>Parameter window</td>
<td><code>&lt;Object type&gt; Parameters</code></td>
</tr>
<tr>
<td></td>
<td><em>For example: Assembly Control Parameters</em></td>
</tr>
<tr>
<td>Printing window</td>
<td><code>&lt;Action&gt; &lt;Grid Items&gt;</code></td>
</tr>
<tr>
<td></td>
<td><em>For example: Print Purchase Orders</em></td>
</tr>
<tr>
<td>Processing window</td>
<td><code>&lt;Action&gt; &lt;Grid Items&gt;</code></td>
</tr>
<tr>
<td></td>
<td><em>For example: Approve Purchase Orders</em></td>
</tr>
<tr>
<td>Progress Indicator</td>
<td><code>&lt;Action&gt; Progress - &lt;Object type&gt;</code></td>
</tr>
<tr>
<td></td>
<td><em>For example: Print Progress - Bills of Material</em></td>
</tr>
<tr>
<td>Wizard</td>
<td><code>&lt;Action&gt; &lt;Object type&gt; Wizard</code></td>
</tr>
<tr>
<td></td>
<td><em>For example: Install BaanERP Wizard</em></td>
</tr>
</tbody>
</table>

- Use plural for the object type if it refers to multiple objects.
Figure 78, Plural because the window presents a list of multiple objects

- Use singular for the object type in case it refers to a single object.

Figure 79, Singular because the window presents a view of a single object

- For exceptions and for more detailed information about titles, see Chapter 4, the section on specific window types.

Session code

- The basic format is ppmmmsfnnp000.

- Codes: pp (=package code), mmm (= module code), s (=submodule code), f (=function code), nn (sequence number), p (process type).
  For example: tdpur2501m000.

Figure 80, Title bar with session code and company code
Buttons

- Act as shortcuts to specific window commands.
- Placed on the right side of the title bar.
- Primary windows include the: Close, Minimize, Maximize, Restore buttons.
- Secondary windows include the: Close, Help buttons.

<table>
<thead>
<tr>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>☑️ Close</td>
<td>Closes the window</td>
</tr>
<tr>
<td>⏪ Minimize</td>
<td>Minimizes the window</td>
</tr>
<tr>
<td>⬤ Maximize</td>
<td>Maximizes the window</td>
</tr>
<tr>
<td>◯ Restore</td>
<td>Restores the window</td>
</tr>
</tbody>
</table>

*Figure 81, Title bar buttons and operations*

Related topics

- Browse list
- Combined window
- Details window
- Dialog box
- Generic browser framework (GBF)
- Message box
- Overview window
- Parameter window
- Printing and processing window
- Progress indicator
- Wizard
**Toolbar**

A toolbar is a panel that contains a set of buttons (controls), which provides quick access to the most frequently used menu items. The buttons can be accessed with the mouse. When the mouse pointer points at a toolbar button, a ToolTip is displayed.

A ToolTip is a small pop-up window that displays the name of the control when the control has no text label.

![Figure 82, Toolbar buttons and Tool Tips](image)

**When to use**

- To provide quick access to menu items.
- Used in primary windows.
- Is not used in a secondary window.

**Guidelines**

- The function of the toolbar button must be consistent with its related menu item.
- Divide the buttons into functional groups in the toolbar.
- Provide a ToolTip for each button, which describes the function of the button.

**Disabling**

- Disable a button if it is not appropriate or applicable in a particular context.
- Disable the button if its related menu item on the menu bar is disabled.
- If the button is disabled, its ToolTip must still be visible.

![Figure 83, Disabled buttons](image)
Primary toolbar

- The primary toolbar used in BAAN is predefined.
- The way the buttons are grouped is standard in BAAN.

![Primary toolbar screenshot](image)

Figure 84, Toolbar of overview window

Secondary toolbar

- Place the buttons that provide quick access to frequently used menu items of the Specific menu in a second toolbar.
- Icons on toolbar buttons will be designed by GUI consultants only (GUI-Consultancy@baan.nl). Icons can be requested with the use of an icon request form, which can be found on the intranet website of GUI Consultancy.
Refer to Chapter 6, section 'Toolbar buttons' for specific guidelines on the toolbar button in the second toolbar.

Related topics
- Combined window
- Generic browser framework (GBF)
- Menus
- Overview window
- Toolbar buttons
6 Window controls

A control is a visual user interface component that allows a user to interact with the system. Some controls display and allow editing of particular values. Other controls start an associated command. Controls are usually identified by text; for example headings, labels, and titles.

When to use

- Provide a control in a window to allow a user to interact with the system.

Guidelines

- Identify each control or input field with a label.
- Provide feedback to indicate when the control has the input focus and when it is activated.

Type of controls

The following controls are described in the next sections of this chapter in alphabetical order:

Browse control

![Figure 85, Browse control](image)

Check boxes

![Figure 86, Check box](image)

Command buttons

![Figure 87, Command button](image)
Details button

*Figure 88, Details button*

Column heading

*Figure 89, Column heading*

Drop-down list box

*Figure 90, Drop-down list box*

Group box

*Figure 91, Group box*

Menu button

*Figure 92, Menu button*
Selection ranges

From [ ] To [ ]

Figure 93, Selection range

Static text fields

Requisition Date [ ]

Figure 94, Label: Requisition Date

002 Expedition (Amsterdam)

Figure 95, Description: Expedition

Price [ ] 150.0000 HFL

Figure 96, Unit field: HFL

Text box

Search Key [ ]

Figure 97, Text box with editable appearance

Tab control

BOM | Routing | Item Methods | Repetitive |

Figure 98, Tab control

Toolbar buttons

Figure 99, Toolbar buttons
Window controls

Window layout\(^{15}\)

This section includes guidelines for the placement of controls in a window.

Dividing information

- If the information is not fully visible in a primary window, provide a scroll bar.
- If the information is too much to fit on a secondary window, provide multiple form pages.
- If there are more than seven unequal or nine equal items in the window, use group boxes. The items must be divided into functional groups.
- If there are less than seven items, blank lines can be used to make functional groups of items.

Blank lines

- Use a blank line to visually separate controls into distinct groups in windows.
- A blank line is used such as a group box without borders.
- Insert a blank line between key fields and first record field/group box.

<table>
<thead>
<tr>
<th>Activities</th>
<th>Planning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field Change Order</td>
<td>010000003</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>General</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
</tr>
<tr>
<td>Order Date</td>
</tr>
</tbody>
</table>

- Insert a blank line between group fields and the column headings.

\(^{15}\) This section is not applicable when developing windows with the Dynamic Form Editor.
Do not insert a blank line in the form editor between group boxes, that are placed under each other.

Avoid to many blank lines on one form page.

Do not place more then two blank lines under each other.

**Alignment**

- Multiple columns of items are not allowed on a secondary window unless items placed on one line are related to each other, or in the case of check boxes.
- Place the command buttons that affect the entire secondary window in a column at the right of the window.
- The first label on the form pages is positioned on column 4 and row 4.
- The left border of the group box is positioned on column 2.
- Key fields must be placed on the same position across the multiple form pages. On the first form page the key fields are editable. On the following form pages the key fields are display. Key fields are never placed inside a group box.

**Tab order**

- The standard input sequence is from top to bottom and from left to right.
- If the input of a field depends on the value of another field, this is considered an input hierarchy. Placing dependent fields below determining fields indicates the hierarchy. The dependent fields can only be accessed for input if these are relevant (disable/enable).
Browse control

The browse control is a small arrow that allows the user to browse on an input field. If the cursor is above this area, it changes into a magnifier. After you click with the left mouse button a browse list is started, which displays only the possible field inputs that can be selected.

<table>
<thead>
<tr>
<th>Browse Control</th>
<th>A control that allows the user to browse on an input field.</th>
</tr>
</thead>
</table>

**Figure 100, Browse control**

**Figure 101, Browse control with cursor**

<table>
<thead>
<tr>
<th>Matrix</th>
<th>Def</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BALAJI</td>
<td></td>
<td>balaji’s matrix</td>
</tr>
<tr>
<td>KIT01</td>
<td></td>
<td>kIt01</td>
</tr>
<tr>
<td>S-2</td>
<td></td>
<td>Sales Office</td>
</tr>
<tr>
<td>ALA1</td>
<td></td>
<td>Macy’s new matrix</td>
</tr>
</tbody>
</table>

**Figure 102, Browse control in grid**

**When to use**

- Used to define a value for an input field.
- Used for the input fields of secondary windows.
- Used for the input fields of an editable overview window or combined window.

**Guidelines**

- Browse control is positioned on the right of the input field.
- Browsing leads to a browse list.
- Use a browse filter for the input field if the user can browse from it and the value of the field is checked automatically.
Navigation

- The shortcut key for browsing through a field is Ctrl+B

Related topics

- Browse list
- Combined window
- Details windows
- Dialog box
- Input fields
- Overview window
- Window layout

Check boxes

A check box is a control used to support options that are either on or off. Check boxes are used for independent or nonexclusive choices.

Check boxes can be used in a group to provide a multiple-choice field. A check box is displayed as a square box with an accompanying label. When the choice is set, a check mark appears in the box.

<table>
<thead>
<tr>
<th>Check box</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control that displays a setting, either checked (set) or unchecked (not set).</td>
</tr>
</tbody>
</table>

![Include_Fully_Paid](image)

*Figure 103, Check box*

When to use

- Use a check box to display an individual settings choice that can be set on or off.
- Use a group of check boxes for setting choices that are not mutually exclusive and can each be set to on or off.
- Use a check box only if a user will clearly understand the meaning of the choice when it is selected or not selected. If this is not the case, use another control to set a single selection, such as a drop-down list box.
- Do not use check boxes that behave such as option buttons.
Guidelines

- Display a check box to the left of its associated label.
- Use a single line of text for the label.
- The label positively describes the option chosen if the check box is checked (no negation, certainly no double negation).

An example of bad labels for check boxes, because of the negation:

![Don't add this signature to replies or forwarded messages](image1)

*Figure 104, Bad example of labels for check boxes*

An example of good labels for check boxes: it is clear what will happen when the option is checked:

![Options]

- Print Approved Orders
- Print Errors

*Figure 105, Good example of labels for check boxes*

- Do not insert blank characters between the check box and its associated label in the form editor.
- Always place a group of related check boxes inside a group box. The check boxes can be positioned in rows or columns, or both. (Note that the tab order of the check boxes must be columnwise).
- A group box with a group of check boxes usually carries the Options group box label.
- Disable a check box and its label if the choice is currently unavailable.

![Select by Level of Inventory on Hand](image2)

*Figure 106, Disabled check box*
Alignment

- If only check boxes are placed inside a group box, then the check boxes must be left-aligned with the groupbox label.

![Figure 107, Check box alignment](image)

- If a check box is positioned in a group box together with other controls, the check box must be left-aligned with other controls in the group box.

![Figure 108, Check box alignment](image)

- If a check box and other controls are both placed outside a group box, then the check box must be aligned with those other controls.

![Figure 109, Check box alignment](image)

- If a check box is placed outside a group box and other controls are placed inside a group box, the check box must be left aligned with the group box label.
Related fields

- The setting of a check box can be used to affect other controls.
- The state of the check box determines the availability of the controls. If the check box is not checked (clear), than all the related controls inside the group box are disabled.
Number of check boxes

- The maximum number of check boxes on the form is twelve.
- All check boxes are always visible.

Related topics

- Group boxes
- Static text fields
- Window layout

Column headings

A column heading control, also known as header control, displays a heading above columns of text or numbers. The control is divided into two or more parts to provide headings for multiple columns.

Column heading

A column heading control consist of multiple header parts that are positioned above the columns. The header part label must identify the content of the column.
When to use

- Used to identify the contents of the columns in the grid.
- Used in overview windows, combined window, and browse list.

Guidelines

- Provide a header part for each column in the grid.
- The recommended width of the label is fifteen positions. The minimum width is three positions.
- All header part labels must be top-left aligned, independent of the alignment of the fields inside the column.
  - Example of bad alignment of the header part labels:

<table>
<thead>
<tr>
<th>Order Origin</th>
<th>Order</th>
<th>Set</th>
<th>Order Group</th>
<th>Order Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sales (manus)</td>
<td>234567028</td>
<td>1</td>
<td>WWW</td>
<td></td>
<td>Warehouse Order Type</td>
</tr>
</tbody>
</table>

  *Figure 114, Wrong alignment of the header part labels*

  - Example of good alignment of the header part labels:

<table>
<thead>
<tr>
<th>Purchase Contract</th>
<th>Description</th>
<th>Buy-from BP</th>
<th>Name</th>
<th>Contract Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>010000004</td>
<td>KPU's Contract</td>
<td>SIVA</td>
<td>Sivakumar</td>
<td>Normal Contract</td>
</tr>
</tbody>
</table>

  *Figure 115, Correct alignment of the header part labels*

- The maximum label height is three lines. However, try to avoid three-line labels.
- Avoid abbreviations.
- All the labels in the same row must have the same number of lines reserved.
- The label width is equal to or larger than the width of the fields inside the column.
- Ellipsis after the label are not allowed.

*Figure 116, Wrong header part label*
Related topics

- Grid
- Static text fields
- Window layout

Command buttons

A command button is a control that starts an action when pressed. It contains a label that describes the action.

![Command button example](Image)

When to use

- Used for actions in secondary windows where there is no menu bar.

Guidelines

Button types

- A generic button applies to the whole window (all forms).
- A group specific button applies to a group of related fields.
- A field specific button only applies to a particular field.

Position

- Generic buttons are positioned on a:
  - Details window, printing and processing window, and a dialog box in a column on the right side starting at the top.

![Generic position of buttons](Figure 117)
Message box and the progress indicator in a centred row on the bottom of the dialog box.

![Generic buttons in message box and progress indicator](image)

- The group specific button must be positioned on the bottom right of the group box containing the related fields.

![Position of group specific buttons](image)

- The field-specific button must be positioned immediately after the field. You do not need to group the field and the command button in a group box.

![Position of field-specific button](image)

- If command buttons apply to multiple form pages, place them in the same position on each form page in one window.

**Labels**

- The label must describe the action the button performs. Use an imperative, for example: Delete, Replace, Copy.

- Make the label fit on the command button.

- Do not use abbreviations.

- Avoid using the label Continue. Describe the action the button performs instead (such as Calculate or Delete).
### Predefined command button labels

<table>
<thead>
<tr>
<th>Label</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>OK</td>
<td>Saves the changes and closes the window</td>
</tr>
<tr>
<td>Cancel</td>
<td>Discards any changes and closes the window</td>
</tr>
<tr>
<td>Close</td>
<td>Prompts the user to save any uncommitted changes, or to cancel the action. If confirmed, the window is closed.</td>
</tr>
<tr>
<td>Save</td>
<td>Saves all the changes without closing the window.</td>
</tr>
<tr>
<td>Revert</td>
<td>Reverses any changes since the last save.</td>
</tr>
<tr>
<td>Stop</td>
<td>Halts a process.</td>
</tr>
</tbody>
</table>

### Ellipsis

- Add ellipsis (…) to the label to indicate that the user must give further information to complete the action. For example: Save As… has an ellipsis because the user must provide a name before the action can be performed.

![Save As...]

- Properties has no ellipsis because the action is to show the object properties which is completed immediately after the command button is activated.

![Properties]

### Disabling

- Disable command buttons when they are not available/appropriate at a certain moment.

![Revert]

- Only use buttons if they are relevant for the task.
- DFE: Hide a command button if the user is not authorized for the action.

### Default button

- Make the command button the default button of the most desired / likely action.
- Avoid making a command button the default button if its action is destructive.
- Provide a visible cue to indicate which command button in a window performs the default action.

![Button Example]

- Place the default button first in the column/row.
- Pressing the ENTER key activates the default button.
- Pressing the ESC key activates the Cancel or Close button.
- Define a default button whenever possible.

**Sequence**

- The first button is a positive action such as the OK button or the default button.
- The second button is a negative action such as the Cancel button.
- The Help button is always placed last in the column/row.

![Button Example]

**Combinations**

- Do not use Close or Continue or OK simultaneously.
- Do not use Close and Cancel simultaneously.

**Number of buttons**

- Minimum number of command buttons in a window is one.
- Maximum number of command buttons in a window is seven.
- All items are always visible.
### Predefined command buttons

<table>
<thead>
<tr>
<th>Window type</th>
<th>Command buttons</th>
</tr>
</thead>
<tbody>
<tr>
<td>Details window</td>
<td>Close, Save, Revert, New (if applicable), Help</td>
</tr>
<tr>
<td>Printing window</td>
<td>Print, Close, Save Defaults, Get Defaults, Make Job, Help</td>
</tr>
<tr>
<td>Processing window</td>
<td>&lt;Action&gt;, Close, Save Defaults, Get Defaults, Make Job, Help</td>
</tr>
<tr>
<td>Parameter window</td>
<td>OK, Cancel, Save, Revert, Print, Help</td>
</tr>
<tr>
<td>Browse list</td>
<td>OK, Cancel, Sort By, Find, Filter, Open Read-Only, New, Print, Help</td>
</tr>
<tr>
<td>Make job dialog box</td>
<td>OK, Cancel, Save Defaults Get Defaults, New Job, Help</td>
</tr>
<tr>
<td>New job dialog box</td>
<td>Save, OK, Cancel, Find, Filter, New, Help</td>
</tr>
</tbody>
</table>

### Related topics

- Browse list
- Details button
- Details window
- Dialog boxes
- Labels
- Message boxes
- Parameter windows
- Printing and processing windows
- Software text standards

### Details button

A details button is a small command button that is positioned next to an input field. By activating this button the details of the input field are opened in a modal secondary window.

#### When to use

- Use a details button to activate a modal secondary window in which the details of the related input field can be edited or displayed.
Only use a details button in details windows.

![Invoice-to](Figure 121, Details buttons)

Guidelines

- Position a details button on the right of its related input field, immediately after the browse widget (if present).

![Address](Guidelines)

- Disable the details button if the related field is empty. Only enable the details button after the field has been filled in.

Related topics

- Details window
- Command button
- Text box

Drop-down list boxes

A drop-down list box is a control with which a single item from a list of items can be selected. The list is displayed when the user opens the list by clicking with the mouse or by using the keyboard. A drop-down list box does not allow a user to type information into it.
When to use

- Use a drop-down list box when a choice must be made from a list of items that are mutually exclusive.
- Use a drop-down list box to display a number of choices that vary in number or content.

Guidelines

- A drop-down list box is preceded by a label that must clarify the meaning of the list.

- Display an initial value from the list in the input field that contains read-only text.
- If a particular choice is not available, omit the choice from the list.
- Display the choices or objects in a drop-down list box in an order that is meaningful to a user. If you display names, this can be an alphabetical order. However a list of dates must be displayed in chronological order.
- If there is no natural or logical ordering for the content, use ascending or alphabetical ordering.
- Display at least four and no more than ten choices or objects at a time in a drop-down list box.
- Provide a vertical scroll bar in case there are more than ten choices or objects in the drop-down list.
- Place minimal four choices or objects and maximum twenty choices or objects in the drop-down list box.

Shortcut keys
- The keyboard combination for opening the drop-down list box is ALT + DOWN ARROW.
- To close the drop-down list box without changing the original value, click outside the box or press the ESC key.

Related topics
- Static text fields
- Shortcut keys
- Scroll bar
- Software text standards
Group boxes

A group box is a rectangular box drawn around a group of other controls to organize them. The group box label is used to identify the controls that are inside the group box.

<table>
<thead>
<tr>
<th>Group box</th>
</tr>
</thead>
<tbody>
<tr>
<td>A rectangular box drawn around a group of controls to indicate that the fields are related.</td>
</tr>
</tbody>
</table>

![Figure 125, Group box](image)

When to use

- Use a group box to indicate that controls are related.
- Only use a group box when the related controls must be visually distinct from other groups of controls in a window.
- Group boxes are only used in secondary windows.

Guidelines

- Place maximum of seven unequal or nine equal controls inside one group box.
- Key fields and group fields are never placed inside a group box.
- Never use a group box around command buttons.

Label

- Provide a label for each group box.
- The label identifies a set of related controls. It describes a mutual feature of the controls inside the group box.
- The label is not identical to the label of a control inside the group box.
If the key fields and only one group box are placed on a form, the group box label and the form page label are identical.

- The label is left-aligned.
- The label is positioned on column four in the form editor.

![Figure 126, Group box label](image)

**Alignment**

- The left border of the group box is positioned on column two.
- Label fields inside a group box are aligned with the group box label in column four.
- Insert a blank line between the group box and the key fields (if there is enough space).

![Figure 127, Blank line between key field and group box](image)

**Size**

- A group box is not larger (higher) than the number of items inside the group box, which means there is no blank line between the first item or after the last item and the group-box border.
- The width of group box depends on the longest item on the form page. This can be an item inside or outside the group box.
- The number of blank characters needed in the form editor between a field and the right group-box border depends on the kind of controls:
− If the control is a label or input field, at least one blank character is needed between the label and the right group-box border.

− If the control is a browse control, at least two blank characters are needed between the browse area and the right group-box border.

− If there are two browse controls, at least three blank characters are needed between the last browse area and the right group-box border.

− If the control is a drop-down list box, at least four blank characters are needed between the drop-down list and the right group-box border.

**Multiple Group Boxes**

- If group boxes are placed under each other on one form page, they must be of equal width.
- If group boxes are placed next to each other, they must together have the same width as the width of the group box on the form page. These group boxes must also have the same height.

![Example of group boxes](image)

- If group boxes are placed on different form pages in one window, these group boxes must be of equal width. The widest group box determines the width of the group boxes on the other form pages.
- Do not insert a blank line in the form editor between group boxes that are placed under each other.
- Insert one blank character in the form editor between group boxes that are placed next to each other.

**Predefined group boxes**

- Used in printing and processing windows.
  For more guidelines on group boxes used in printing and processing window refer to Chapter 4, “Window Types”, section “Printing and Processing windows”.
- Used for selection ranges.
  For more guidelines on group boxes used for selection ranges, refer to Chapter 6, “Window Controls”, section “Selection ranges”.

**Dynamic form editor**

- When you develop sessions with the Dynamic Form Editor, the alignment and the size for one or multiple group boxes are taken care of automatically.

**Related topics**

- Form page
- Static text
- Printing and processing windows
- Selection ranges
- Window layout
Menu button

The menu button is a command button that displays a drop-down menu when it is pressed. The user can choose one of the items in the menu.

![Menu button example]

Figure 128, A menu button is a command button that shows a drop-down menu when pressed by the user.

When to use

Try to avoid the need for a menu button. Actions on a menu button are concealed from the user and harder to access than a command button.

There is one situation in BaanERP in which the menu button can be used:

- In a browse list. Here only actions that support the user in choosing an object from the list are allowed, other than the default buttons on the browse list.

Guidelines

Do nots

- Do not use any of the actions that are on predefined command buttons (for example, OK, Cancel, Print, and so on) in the menu of a menu button. Refer to Chapter 6, Section ‘Predefined command buttons’ for more information on this topic.
- Do not use a menu button for frequently used actions or commands.
- Do not use submenus in the drop-down menu of the menu button.

Labels

- Use a short label for the menu button itself.
- Use a label that describes the contents of the menu. For example: Text and Advanced. Do not use labels such as:
  - Text-related commands (too long)
  - Other (does not describe the contents of the menu)
- Do not use an ellipsis (…) in the label of a menu button.
Add ellipsis to the label of a menu item if the user must give additional information to complete the command. The Save As... command, for example, has ellipsis because the user must give information in a separate dialog box before the action can be completed.

Do not use shortcut keys in the drop-down menu of the button.

**Number of buttons**
- Try to use no more than three menu buttons on one form.
- Use different menu buttons for different kinds of commands. For example, do not place all actions related to business partners and attachments in one menu button. Instead, use one menu button labeled Business Partner and another labeled Attachments.

**Number of items**
- The minimum number of items inside a menu button is two.
- The maximum number of items inside a menu button is twelve.
- All items inside an opened menu button are always visible.

**Browse list**
- Only use a menu button labeled Advanced in the browse list.
- Only place actions in the browse list that help the user find objects in the browse list. Any of these actions (other than the standard commands) are only allowed in the Advanced-menu button.

**Related guidelines**
- For the position of a menu button, the guidelines for command buttons apply. Refer to Chapter 6, “Window Controls”, section “Command buttons”.
- For more guidelines on designing menus, refer to Chapter 5, ‘Window Components’, section ”Menus”.

**Related topics**
- Menus
- Command buttons
- Details window
- Browse list
Selection ranges

A selection range is used if a selection of data is required. These ranges consist of From-To fields. Two kind of selection ranges exist in BaanERP:

- **Columns**: From-To is placed next to each other on the same row, as can be seen in Figure 131.

  ![Figure 129, From - To Columns](image)

- **Rows**: From Item-To Item is placed under each other, as can be seen in Figure 132.

  ![Figure 130, From - To Rows](image)

When to use

- To define a range of data.
- Used in printing and processing windows.
- Use the From-To columns as much as possible. Use From-To rows when the input fields are too wide to fit next to each other, such as the item fields. In other cases, where you want to put the selection range in rows contact a GUI consultant.

Guidelines

- The From-To fields are always placed inside a group box with the Selection Range group box label.
**Default values**

- The default values of the selection ranges must be filled according to the following guidelines:
  - If only one object is selected in the related overview window, the From-To fields must be filled with the selected Item/Object by default.
  - If no objects are selected in the related overview window, the From fields must be filled with the minimum value and the To fields must be filled with the maximum value.
  - If more than one object is selected in the related overview window, the From fields must be filled with the minimum value and the To fields must be filled with the maximum value.

- Fill the From fields with the minimum value by using the `fmin` function.
- Fill the To fields with the maximum value by using the `fmax` function.
- `fmin` and `fmax` fill a field with respectively the minimum or the maximum value for the actual language. This is the case for strings, numbers, and dates.

**From-To columns**

- Always use From-To columns except for wide fields, such as the item fields.
- Position the From-To labels on the first empty line in the group box.
- Place a label before From-To input fields which describes what kind of selection is made, as shown in Figure 131. This label is left-aligned and positioned under the group box label. The distance between the label and the From input fields is one character.
- The From-To labels start with the capitals F and T.
- The From-To labels are left-aligned.
- Left align the From-To input fields with the From-To labels.
- Separate the From input fields from the To input fields with a hyphen [-].

Figure 131, From - To columns
The distance between the From input fields and the To input fields is determined by the longest From input field:

- If the From field is longer than 4 characters, then use:
  2 characters, hyphen [-], 2 characters.

\[
\begin{array}{c|c}
\text{From} & \text{To} \\
\hline
\text{FFFFF} & \text{FFFFF} \\
\end{array}
\]

- If the From field is smaller than 4 characters, then use
  3 characters, hyphen [-], 2 characters.

\[
\begin{array}{c|c}
\text{From} & \text{To} \\
\hline
\text{FFF} & \text{FFF} \\
\end{array}
\]

- If the From field is a drop-down-box, then use:
  5 characters, hyphen [-], 2 characters.

**From-To rows**

- Use From-To rows only for item fields\(^{16}\). In case of an exception contact a GUI Consultant ([GUI-Consultancy@baan.nl](mailto:GUI-Consultancy@baan.nl)).
- Place the From and To fields directly under each other.
- The input fields are (left) aligned with each other.
- The From Item and To Item labels precede the item fields.
- The From Item and To Item labels start with the capitals F and T.
- The From Item and To Item labels are left-aligned with the other labels.

\[
\begin{array}{|c|c|}
\hline
\text{Selection Range} & \\
\text{From Item} & \\
\text{To Item} & \\
\hline
\end{array}
\]

*Figure 132, From - To Rows*

**Combination of columns and rows**

- Columns and rows are both allowed in one group box.

---

\(^{16}\) This guideline has to be followed when designing new sessions with the old form editor and when working with the dynamic form editor.
- From-To columns are placed together, and the From Item – To Item row are placed together.

- The From-To columns are always positioned as first on the form, followed by the From Item- To Item row. Exception: In case the item must be selected first, you must position the From Item – To Item on row first.

- The From-To columns and From Item- To Item row are separated by a blank line (if enough space).

- The From-To columns must not be intersected (not even by From Item- To Item rows).

![Figure 133, From - To combination](image)

**Alignment**

- Example of bad alignment:

![Figure 134, Bad example because of wrong alignment](image)
Example of good alignment:

![Selection Range](image)

**Figure 135, Good example**

### Related topics

- Group boxes
- Static Text
- Printing and processing

### Static text fields

A static text field is used to present read-only information. Unlike read-only text box controls, the text is not selectable.

<table>
<thead>
<tr>
<th>Static Text Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>A static text field is used to present read-only information that can not be selected.</td>
</tr>
</tbody>
</table>

Static text fields used in Baan are:

1. **Labels**

   ![Requisition Date label](image)

   **Figure 136, Requisition Date label**

2. **Descriptions**

   ![Tata's Warehouse for Pur.Items](image)

   **Figure 137, Description: Tata’s Warehouse for Pur.Items**

3. **Unit fields**

   ![Price](image)

   **Figure 138, HFL is the unit field for a currency (HFL).**
Label

A label is a static text field that identifies a control, such as a text box or a check box, as can be seen in Figure 139.

![Figure 139, Example of labels](image)

When to use

- Provide a label for each control unless the controls appear in columns and the column heading serves as the label.
- Labels must identify the meaning of the controls to the user.

Guidelines

- Labels must be meaningful, short, and informative.
- Labels do not contain the word Data.
- Do not place a colon (:) after the label.
- Avoid abbreviations in labels, because these are often incomprehensible to the user.
- Usually labels are left aligned. Exceptions are From-To labels.
- The first label on the form is positioned on column four and row four.
- The length of the label must be at least 1.5 times its own number of characters in English, because of translation. Recommended label length is twenty positions.
- Define label fields with the maximum number of available positions, with at least 1.5 times the number of the English positions.
- The maximum height for a label is three lines.
- Use the MS Sans Serif, 8pt font.
Description

A description provides additional information about the value of the text box.

When to use

- Use a description to clarify the value of a text box. For example, in case the value is a code.

Guidelines

- Descriptions are positioned on the right of the associated text box.

![Figure 140, Description behind the text box]

- If there is not enough space, the description is positioned under the text box.

Unit fields

A unit field describes the unit (for instance pieces or kg, but also currencies) of the input field to which it belongs.

![Figure 141, An example of a unit field for a currency (HFL)]

When to use

- Use a unit field for each input field that contains:
  - Currency,
  - Length,
  - Weight,
  - Duration,
  - Quantity,
  - Area,
  - Volume,
  - or a combination of these (such as USD per kg).
- In a grid, place the unit field in the header.

Guidelines

- The unit field be positioned after the related input field.
- There must be no brackets around a unit field.
Unit fields must be left-aligned.

There must be only one unit field positioned after the related input field.
- Exceptions to this guideline are combinations of units, such as a price per piece, or weight per volume. In both cases both unit fields must be positioned after the related input field with a slash in between, as shown in Figure 140.

![Figure 142, A combination of two unit fields](image)

**Related topics**

- Check boxes
- Drop-down list box
- Form page
- Group boxes
- Selection range
- Tab control
- Software text standards
Text box

A text box, see Figure 143, is a rectangular control in which the user can type in text. A text box can have three appearances:

1. Editable: the user can type in text
2. Read-only: the user cannot type in text, but can select it.
3. Disabled: the user cannot type in the text and cannot select the text.

<table>
<thead>
<tr>
<th>Text box</th>
</tr>
</thead>
<tbody>
<tr>
<td>A text box is a rectangular control that allows the user to enter or edit text.</td>
</tr>
</tbody>
</table>

Editable appearance

Text boxes with editable appearance display a value that can be edited and selected by the user.

![Figure 143, Text Box with editable appearance](image)

Read-only appearance

Text boxes with a read-only appearance display a value that cannot be edited, but can be selected.

![Figure 144, Text box with read-only appearance](image)

![Figure 145, Text box with read-only appearance and selected value](image)

Disabled appearance

Text boxes with disabled appearance display a value that cannot be edited nor selected, because the value is not available in the current context.

![Figure 146, Text box with disabled appearance](image)
When to use

- Provide an editable text box to allow a user to type values that cannot be supplied in a list of choices.
- Provide a text box containing read-only text if a value is displayed that a user can select, but cannot change due to authorization or the status of an object.
- In a combined window the text boxes are placed above the grid, as can be seen in Figure 147.

Guidelines

- Text boxes are left-aligned. The only exception to this are numeric fields in a column, which are right-aligned. Below, a table provides an overview of the alignment of text boxes:

<table>
<thead>
<tr>
<th>Field type</th>
<th>Alignment</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alphabetical</td>
<td>Left</td>
<td>Hello</td>
</tr>
<tr>
<td>Alphanumeric</td>
<td>Left</td>
<td>AA123</td>
</tr>
<tr>
<td>Date fields</td>
<td>Left</td>
<td>30-10-98</td>
</tr>
<tr>
<td>Numeric</td>
<td>Right</td>
<td>1234567</td>
</tr>
</tbody>
</table>

- Provide a label to identify the text box. The label must be positioned to the left of the text box.
- Provide a description to clarify the meaning of the value of the text box. The description must be positioned to the right of the text box. If there is not enough space, position the description under the text box.
- Unit fields must be positioned to the right of the text box.
- Make the relation between text boxes clear by positioning them close to each other, or by grouping them in a group box.

- Provide a browse control in case you can define a value for a text box with use of a browse list.

  ![Browse control](Figure 149, Browse control)

- Provide a details button to view or edit the details of a text box when applicable.

  ![Details button](Figure 150, Details button)

- Avoid needless blank lines between text boxes.

- From - To text boxes must be positioned according to specific standards. Refer to Chapter 6, "Window Controls", section "Selection Ranges" for more specific guidelines on From - To text boxes.

**Related topics**

- Browse control
- Details button
- Form pages
- Static text fields
- Selection ranges
Tab control

A tab control is analogous to a folder's tag in a file cabinet.

<table>
<thead>
<tr>
<th>Tab control</th>
<th>A control is used to define multiple logical pages or sections of information within the same window</th>
</tr>
</thead>
<tbody>
<tr>
<td>BOM</td>
<td>Routing</td>
</tr>
</tbody>
</table>

When to use

- Tabs are used to define multiple logical form pages of information in the same window.
- Tabs are used in secondary windows.

Guidelines

- A tab control displays only one row of tabs.
- Avoid multiple rows or scrolling a single row of tabs. Reason: they make it harder to read and access a particular tab.

<table>
<thead>
<tr>
<th>Track Changes</th>
<th>User Information</th>
<th>Compatibility</th>
<th>File Locations</th>
</tr>
</thead>
<tbody>
<tr>
<td>View</td>
<td>General</td>
<td>Edit</td>
<td>Print</td>
</tr>
</tbody>
</table>

*Figure 151, Example of a wrong tab control because of the multiple rows.*

Label

- Provide a label for each tab in case of multiple form pages in one window.
- Labels must clarify what kind of information is displayed in each form page.
- If only one group box and one key field are placed on the form, form page label and the group-box label are identical.
Navigation keys

- If a tab has the input focus, the left arrow or right arrow keys move between tabs.
- Switching between tabs:
  - CTRL+TAB: next form pages
  - CTRL+SHIFT+TAB: previous form page

Related topics

- Details windows
- Form pages
- Static text
- Window layout

Toolbar buttons

Toolbar buttons provide quick access to the most frequently used menu items. In Baan two kinds of toolbar buttons are used: standard buttons and specific buttons.

![Toolbar buttons](image)

*Figure 152, Toolbar buttons*

When to use

- Standard buttons are used in the first toolbar of primary windows. The first toolbar is predefined for overview windows. These buttons are used in the whole BAAN application.
- Specific buttons are used in the second toolbar of primary windows. These buttons provide quick access to items given on the Specific menu (formerly Application). These buttons can be specific for each package.

Mandatory

- The toolbar buttons must be consistent in look and feel throughout the whole Baan product. Therefore the icons of the toolbar buttons are designed by GUI consultants (GUI-Consultancy@baan.nl).
- Icons for toolbar buttons can be requested with the use of an icon request form. This form can be found on the intranet website of the GUI consultancy.
Guidelines

Consistent use of the buttons allows users to transfer learning and skills from package to package. If you use a common button for a different function you can confuse the user. Therefore the following requirements must be met:

General requirements

- Provide toolbar buttons for frequently used items.
- Buttons must be unambiguous and clearly represent the functions to which they refer.
- A group of related buttons must be identifiable as a set of buttons by, for example, style, color, size, metaphor, and mutual relationships.
- If a button in BAAN reuses the metaphor or characteristics of a common button of another application (such as the common toolbar buttons of Microsoft) then it must have the same meaning as the original button.
- Repeat the use of buttons when applicable.
- Do not use a button for two or more functions in Baan. Only one button can be used for each function throughout Baan.
- Always place the buttons in the same order and in the same position in the toolbar.

Detailed requirements

- Do not use letters or words on buttons. These make graphics difficult to apply in other cultures.
- Buttons must be functional on all backgrounds (light, dark, gray, and colored).
- Buttons must be designed assuming a light source from the upper left.
- Only use the amount of detail that is really necessary for user recognition and recall.

Technical requirements

- Buttons must be 15 pixels high and 16 pixels wide.
- Buttons must be in GIF format.
- The standard 16-color palette must be used for buttons.
- The bottom left pixel of an icon must be in the background color, because this pixel is used to determine the background color of an icon. (That color will be made transparent by the standard program).

**BAAN requirements**

- Mostly use the common colors of Baan Software: black, dark gray, white, and blue.
- Avoid using other colors, unless there is a good reason. For example, light yellow can be used to be consistent with the standard folder color of MS Windows.
- Keep in mind that colors have subjective values. For example, red is interpreted as danger/bad/false and green is interpreted as good.
- Use the color blue to emphasize the essential part of a button, if useful.
- Buttons must fit in the Baan style: business-such as, simple, distinct, and two-dimensional.
- Colors have a subjective meaning in different cultures.

**BAAN button archive**

- All buttons used in Baan must be present in the Baan button archive.
- These buttons must be re-used for the same function within BAAN. Do not use these buttons for other functions.
- If you have doubts about whether a button should or should not be re-used contact a GUI consultant.
- To keep the archive updated, all buttons must be sent to the GUI Consultancy.

*Figure 153, Standard toolbar buttons*
Window controls

Figure 154. Specific toolbar buttons

Related topics
- Combined window
- Details window
- Overview window
- Toolbar

Baan GUI Style Guide
6-42
Interaction and navigation

Interaction is how users interact with the user-interface components.

The placement of controls, keyboard and mouse support, and the effective use of title bars, shortcut keys, and access keys can help the user complete frequent tasks more easily and quickly.

Guidelines

- Make sure navigation appears from top left to bottom right within a group box, dialog box, or window. Reason: this kind of navigation is consistent with the way people read.
- Make sure menus and menu items appear in the same sequence across windows. Standardize the placement of the menu items within the Specific drop-down menu. Reason: consistency across windows.
- Provide parallel mouse and keyboard support. Reason: some users prefer the keyboard, and sometimes a mouse is not available.
- Do not use mouse and keystroke combinations. Reason: awkward for users and sometimes a mouse is not available.
- Allow users to switch between available input devices. Input devices are complimentary, not mutually exclusive.

This section describes how the interaction within Baan takes place. The following topics are included:

- Access keys
- Browse Filter
- Input basics
- Navigation flow
- Shortcut keys
- Synchronization
Access keys

An access key is a single, easy-to-remember alphanumeric key that moves the cursor to a choice and selects the choice.

For example: ALT+F+S activates the action File, Save.

When to use

- Use an alphanumeric key in combination with the ALT key to navigate a control and activate it.
- Provide the predefined access key for each predefined choice.

Guidelines

- Make sure the access key matches one of the characters in the text label of a command.
- Identify each access key with an underline.
- Do not capitalize access keys if the letter would not normally be capitalized.
- Be aware that access keys are not case sensitive.
- Make access keys unique:
  - Within each drop-down menu, cascaded menu, and pop-up menu.
  - For all choices, field prompts, controls that allow typing, command buttons, and menu bar choices within a window.

Use the following guidelines to assign unique access keys:

- The same letter that was assigned to a similar or equal function or feature in a peer window.
- The first letter of the first word in the label. For example:
  - Save
- The first letter of the next word in the label (if any). For example:
  - Save As...
- A distinctive consonant in the label.
  - Preview
- A distinctive vowel in the label.
  
  **Setup…**

- The next letter of the first word in the label.

- The next letter of the next word in the label (if any).
## Predefined access keys for menu items used in BAAN:

<table>
<thead>
<tr>
<th>Access keys</th>
<th>Group</th>
<th>Specific</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>File</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>New</td>
<td>New</td>
<td></td>
</tr>
<tr>
<td>Open</td>
<td>Find</td>
<td></td>
</tr>
<tr>
<td>Open Read-Only</td>
<td>Duplicate</td>
<td></td>
</tr>
<tr>
<td>Save</td>
<td>First</td>
<td></td>
</tr>
<tr>
<td>Print</td>
<td>Previous</td>
<td></td>
</tr>
<tr>
<td>Send to Desktop</td>
<td>Next</td>
<td></td>
</tr>
<tr>
<td>Exit</td>
<td>Last</td>
<td></td>
</tr>
<tr>
<td><strong>Edit</strong></td>
<td></td>
<td>Help</td>
</tr>
<tr>
<td>Revert</td>
<td>Cancel</td>
<td></td>
</tr>
<tr>
<td>Find</td>
<td>Suspend</td>
<td></td>
</tr>
<tr>
<td>Duplicate</td>
<td>Reject</td>
<td></td>
</tr>
<tr>
<td>Delete</td>
<td>Next</td>
<td></td>
</tr>
<tr>
<td>Delete Range</td>
<td>Current</td>
<td></td>
</tr>
<tr>
<td>Text</td>
<td></td>
<td>Tools</td>
</tr>
<tr>
<td>Attachments</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>View</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sort By</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Filter</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Refresh</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Refresh Rate</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Related topics

- Combined window
- Details window
- Keyboard
- Menus
- Overview window
- Shortcut keys
**Browse filter**

Browse filters prevent the user from selecting an object in a browse list that is not valid. In addition, a browse filter reduces the number of objects presented in the browse list, which makes an object easier to find.

A browse list with an active browse filter will only show those records that are accepted by the automatic DAL checks on the input field from which the user browses.

For example, an input field exists that browses to an address in which the user is only allowed to enter an address within a certain country. In this case, the browse list only shows those addresses within the allowed country.

**When to use**

- Use browse filters at all input fields that have a check and a browse list attached.

**Guidelines**

- Figure 155 shows the general principle of a browse filter.
- For more detailed information on implementing browse filters, please refer to the Active Apps Development Guide (D0330A US).

*Figure 155, A browse filter uses the checks in the DAL to filter out all those objects in the browse list that will not be accepted by the DAL.*
Related topics

- Browse list
- Browse control

Input basics

The user can interact with objects in the interface by using two types of input devices: a mouse and a keyboard.

Definition

- A keyboard is a device, that consists of systematically arranged keys and allows a user to type information, move the cursor, or select functions assigned to keys.
- A mouse is a commonly used pointing device, that contains one or more buttons and allows a user to interact with a product or the environment. A mouse button is mapped to one of the functions: selection, manipulation, and displaying a pop-up menu.

When to use

- Use these input items to provide access to all functions of an object.

Guidelines

Keyboard

- Use the CRTL, SHIFT, and ALT keys only to modify the function of other keys or key combinations on the keyboard.
- Use the ALT key only to provide access to access keys (mnemonics).
- Use the CTRL key to provide access to shortcut keys.

Mouse

- Make sure when a user clicks or double-clicks the left mouse button, the function assigned to a click of that button is performed.
- Do not assign a double-click as the only way to select a choice and perform an action. Provide a keyboard alternative, such as a menu choice or command button. Reason: the commands must be visible to the users.
Pointer

- Display a pointer whenever a pointing device (mouse) is used.
- Use a pointer to point to choices and objects that a user wants to select or with which the user wants to interact.
- Change the shape of the pointer to provide feedback about a particular location, operation, or state:
  - Display an arrow pointer, for pointing, selecting or moving.

  *Figure 156, Arrow pointer*

  - Display an hourglass pointer to indicate that the system is busy processing, so the user must wait.

  *Figure 157, Hourglass pointer*

  - Display an arrow plus an hourglass pointer to indicate that the system is busy processing in the background, but the user can still perform actions.

  *Figure 158, Arrow + hourglass pointer*

  - Display a magnifier, if the mouse pointer is positioned on a browse area in a Details window.

  *Figure 159, Magnifier*

Related topics

- Access keys
- Shortcut keys
Abstract:

The navigation flow within Baan is displayed in the following figure.

**Navigation flow**

The navigation flow within Baan is displayed in the following figure.

![Navigation flow diagram](image)

**Description of flow**

Selecting an object from the menu browser results in one of the following windows:

- Combined window
- Overview window
- Parameters window

A New or Open action in the overview window, combined window, or parameter overview results in a Details window in which more information of the selected object is presented.

Browsing on an input field results in a browse list in which all the possible input values are shown.

Activating an action that requires more information from the user opens a dialog box.
Activating a printing or processing action results in a printing window resp. processing window.

**Related topics**
- Browse Control
- Browse list
- Combined window
- Details window
- Menu browser
- Overview window
- Parameters window
- Printing & processing window

**Shortcut keys**
A shortcut key is a function key or combination of keys that a user can press to directly perform an action that is also available from a menu.

**When to use**
- Use shortcut keys in menus for features that are often used.
- Provide the predefined shortcut key assignment for each predefined choice in a menu.
- Provide a unique shortcut key assignment for each frequently used choice in a menu.

**Guidelines**
- Make sure shortcut keys are language independent.
- If a menu item has a shortcut key assignment, display the shortcut key to the right of the menu item.
- Make sure shortcut keys are left aligned.
- When displaying shortcut keys, use a plus sign (+) between the key names to indicate that a user must press two or more keys at the same time. No extra spacing is used: for example, CTRL+SHIFT+A. See also hyphenation.
- Perform the function assigned to a shortcut key only if it is assigned within the active window.
- Do not assign more than one shortcut key to the same function.
- Assign the same shortcut key for the same choice in all windows that provide that choice.
- Avoid using the ALT key as part of the shortcut assignment. ALT should be used for access keys.
- Remember that a menu item is disabled, its associated shortcut key should not be applicable.

### Predefined shortcut keys

#### Shortcuts used in Baan windows

The following shortcuts are related to the menu items in the overview window:

<table>
<thead>
<tr>
<th>Shortcut keys</th>
<th>File menu</th>
<th>Group menu</th>
</tr>
</thead>
<tbody>
<tr>
<td>New</td>
<td>CTRL+N</td>
<td>New</td>
</tr>
<tr>
<td>Open</td>
<td>CTRL+O</td>
<td>Previous</td>
</tr>
<tr>
<td>Save</td>
<td>CTRL+S</td>
<td>Next</td>
</tr>
<tr>
<td>Print</td>
<td>CTRL+P</td>
<td></td>
</tr>
<tr>
<td>Exit</td>
<td>ALT+F4</td>
<td>Calculator</td>
</tr>
<tr>
<td>Edit menu</td>
<td></td>
<td>Calendar</td>
</tr>
<tr>
<td>Find</td>
<td>CTRL+F</td>
<td>Rotate Currency</td>
</tr>
<tr>
<td>Duplicate</td>
<td>CTRL+D</td>
<td>Window menu</td>
</tr>
<tr>
<td>Delete</td>
<td>CTRL+DEL</td>
<td>Switch to Details</td>
</tr>
<tr>
<td>Text</td>
<td>CTRL+T</td>
<td>Help menu</td>
</tr>
<tr>
<td>View menu</td>
<td></td>
<td>Help Topics</td>
</tr>
<tr>
<td>Refresh</td>
<td>F5</td>
<td></td>
</tr>
</tbody>
</table>
The following shortcuts are not related to the menu items:

<table>
<thead>
<tr>
<th>Not menu related shortcut keys</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Cut</td>
<td>CTRL+X</td>
</tr>
<tr>
<td>Copy</td>
<td>CTRL+C</td>
</tr>
<tr>
<td>Paste</td>
<td>CTRL+V</td>
</tr>
<tr>
<td>Browse</td>
<td>CTRL+B</td>
</tr>
<tr>
<td>Next Control</td>
<td>TAB</td>
</tr>
<tr>
<td>Previous Control</td>
<td>SHIFT+TAB</td>
</tr>
<tr>
<td>Next Form Page</td>
<td>CTRL+TAB</td>
</tr>
<tr>
<td>Previous Form Page</td>
<td>CTRL+SHIFT+TAB</td>
</tr>
<tr>
<td>First Records</td>
<td>CTRL+HOME</td>
</tr>
<tr>
<td>Next Record</td>
<td>PAGE DOWN</td>
</tr>
<tr>
<td>Previous Records</td>
<td>PAGE UP</td>
</tr>
<tr>
<td>Last Records</td>
<td>CTRL+END</td>
</tr>
</tbody>
</table>

**Specific menu**

The Specific menu contains form-specific commands. Two types of form-specific commands can be defined: common commands and specific commands.

- Common commands are defined as 'used in multiple packages'.
- Specific commands are defined as 'used within one package'.

The shortcut keys for these commands must be allocated according to the following guidelines:

<table>
<thead>
<tr>
<th>Allocated shortcut keys</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequently used common commands: CTRL+F1 through CTRL+F10</td>
<td></td>
</tr>
<tr>
<td>Specific commands: CTRL+SHIFT+letter.</td>
<td></td>
</tr>
</tbody>
</table>
### Predefined shortcut keys for frequently used common commands

<table>
<thead>
<tr>
<th>Menu label</th>
<th>Shortcut keys</th>
</tr>
</thead>
<tbody>
<tr>
<td>Duplicate</td>
<td>CTRL+F1</td>
</tr>
<tr>
<td>Generate</td>
<td>CTRL+F2</td>
</tr>
<tr>
<td>Approve</td>
<td>CTRL+F4</td>
</tr>
<tr>
<td>Release to / Transfer to</td>
<td>CTRL+F6</td>
</tr>
<tr>
<td>Process</td>
<td>CTRL+F8</td>
</tr>
<tr>
<td>Close / Finalize</td>
<td>CTRL+F9</td>
</tr>
<tr>
<td>Move to History</td>
<td>CTRL+F10</td>
</tr>
<tr>
<td>Level Up</td>
<td>ALT+ RIGHT ARROW</td>
</tr>
<tr>
<td>Level Down</td>
<td>ALT+ LEFT ARROW</td>
</tr>
</tbody>
</table>

### Predefined shortcut keys for frequently used specific commands

<table>
<thead>
<tr>
<th>Menu label</th>
<th>Shortcut keys</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lines</td>
<td>CTRL+SHIFT+L</td>
</tr>
</tbody>
</table>

- Do not use CTRL+%, CTRL+#, CTRL+I, or CTRL+SHIFT+I.
- Use the same shortcut key for specific menu commands that share the same functionality across various windows.

### Related topics

- Access keys
- Combined window
- Details window
- Menus
- Overview window
Synchronization

Synchronization means that actions performed in one of the windows will directly effect other related windows.

When to use

- Use synchronization on one set of windows that operate on the same main table. Such a set consists of an overview window and a details window.

- Use synchronization to coordinate two or more sets of windows, a process called child-synchronization. Child synchronization is synchronization between windows that operate on different main tables.

Description

In the picture above, four windows are shown. Windows 1 and 3 are overview windows and windows 2 and 4 are details windows. Windows 1 and 2 operate on the same main table, and windows 3 and 4 operate on another table. So, window 1 and window 2 together form a set of windows, and window 3 and 4 form a set of windows. The windows of each set (1-2 and 3-4) are synchronized with each other.
Interaction and navigation

Procedure
- Select one object in window 1 according to the object-action model.
- Double-click or open the selected object to start window 2, which displays the details of the selected object.
- Select a menu item in the Specific menu to start window 3, which displays the children of the selected object in window 1.
- Select an object in window 3 and double-click or open it. Window 4 is started, which displays the details of the selected object.

Synchronization step-by-step

Situation 1
If another object is selected in window 1 and a default action is carried out, the following happens:
- In window 2, the data of the selected object is displayed. This is the normal synchronization.
- In window 3, the data is not refreshed.
- In window 4, the data is not refreshed.

Situation 2
If another object is selected in window 1 and a form specific command is carried out, the following happens:
- In window 2, the data is not refreshed by the detailed information of the selected object.
- In window 3, the children of the selected object are displayed which is child synchronization.
- In window 4, the data is not refreshed.

Situation 3
If the data has been changed in window 2 and another object is selected and a default action is carried out in window 1, the following will happen:
- In window 2, the following message is displayed: Do you want to save changes you made? The user can respond by clicking Yes or No.
- If the user clicks Yes, the changed data is saved and the data displayed in windows 2, 3 and, 4 will refresh as explained in situation 1.
- If the user clicks No, the changed data is not saved and the data displayed in windows 2, 3 and, 4 will refresh, as explained in Situation 1.

**Situation 4**

If the data has been changed in window 4 and another object is selected and the default action is carried out in window 1, the following happens:

- In window 4, the following message box is displayed: Do you want to save changes you made? The user can respond by clicking Yes or No.
- If the user selects Yes, the changed data is saved and the data displayed in windows 2, 3, and 4 will refresh, as explained in Situation 1.
- If the user selects No, the changed data is not saved and the data displayed in the windows 2, 3, and 4 will refresh, as explained in Situation 1.

**Guidelines**

- An overview window of type 2 is always synchronized with a details window of type 1.
- A combined window of type 3 is always synchronized with a details window of type 3.

<table>
<thead>
<tr>
<th>Synchronization</th>
<th>Overview window (type 2)</th>
<th>Combined window (type 3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Details window (type 1)</td>
<td>Synchronized</td>
<td></td>
</tr>
<tr>
<td>Details window (type 3)</td>
<td></td>
<td>Synchronized</td>
</tr>
</tbody>
</table>

- A printing & processing window (type 4) is never synchronized with other windows.

**Related topics**

- Details window
- Message boxes
- Object-action model
- Overview window
8 Software text standards

Introduction

This chapter provides the text standards for each Baan application software component. It replaces the text standards sections of the Software Coding Standards (S5003A US). The text standards must be applied when you develop software in the Baan Tools environment. This chapter describes:

- Text standards: general rules for spelling, capitalization, hyphenation, terminology, abbreviations, internationalization support, and translatability
- Text standards by software component: Specific naming rules and text usage for each type of software component.
- Labels: general rules for label usage and specific rules for each label type.
- Translatability restrictions: rules to ensure that the labels can be translated.

We have aimed at standardizing and simplifying the guidelines for the various software components as much as possible. Also, we have added textual guidelines for the new GUI elements that were introduced in BaanERP.

The first section describes the general guidelines for the user interface, terminology, as well as some general guidelines for internationalization and translation. The second section describes the specific guidelines for the individual software components such as modules, sessions, messages, and so on.

References

1 Baan Writing Standards [BWS], S5005A US, (http://baubel01/)
2 Baan Terminology Database (http://baandev-secure/btg)
3 Software Coding Standard [SCS], S5003A US (http://baandev-secure/bdm)
4 Microsoft Manual of Style (http://www.microsoft.com)
5 The APICS Dictionary (http://baubel01/TD/dictionaries.htm)
6 Microsoft Press Computer Dictionary (http://baubel01/TD/dictionaries.htm)
7 Standard Baan English reference card (V8006A US)
Text Standards

General text standards

The development language is American English. For differences between American and British English, see the BTG web site on the extranet. The general rules of Standard Baan English (SBE) terminology, spelling, and hyphenation are defined in the Baan Writing Standards (BWS). Unless overruled by specific text standards for a particular type of software component, the following general text standards apply:

- Software component descriptions must be brief and concise.
- All software component descriptions except messages and questions are written in headline-style capitalization (see examples in the section ‘Capitalization’).
- Descriptions do not end with a period (.)
- Refer to alphanumeric entities as Codes and numeric entities as Numbers.
- Do not include the words Code, Data, File, ID, Number if they do not add any information. Use: Item Group, Order. Avoid: Item Group Code, Order Number.
- Use no more than four nouns in noun clusters: Inventory-Receipt Transaction Costs.

Spelling

Follow the American English spelling as indicated in the Baan standard reference works. If you are looking for a logistic or IT term, make sure to check specialist dictionaries such as APICS or the Microsoft Computer Dictionary. If a term is not listed in these dictionaries, refer to the Merriam Webster’s dictionary.

Following these reference works, you can arrive at the correct way to write most terms used in Baan.

Examples:

- ABC analysis of by-products
- Decentralized inventory control
- Failsafe work methods
- Multicurrency budgeting
- Process-focused organization
- Transshipment cancellation
Capitalization

For more information about capitalization, consult the Capitalization section in the Microsoft Manual of Style.

**Headline-style of capitalization**

Most *descriptions* of interface components follow the headline-style capitalization. These components include:

- Windows, menus, toolbars, and tabs
- Fields, buttons, and arrows
- Dialog boxes (check boxes, text boxes, group boxes, and so on)
- Options in drop-down list boxes

Every noun, verb, adverb, adjective, and pronoun in the component’s description gets an initial capital. The name of the component type remains lowercase:

- Specific menu
- Purchase Orders by Production Order command
- Formal to Actual Business Processes (tgwms1520m000) session

Do not capitalize articles, conjunctions, and prepositions of fewer than five letters:

- Production Save and Exit command
- Journal Print with Finalization check box

Capitalize every first and last word of the description:

- Save As button

Do not capitalize the second part of hyphenated words if the second part is a preposition:

- Sold-to Business Partner field

  **NOTE**

- Sold-To group box (because *To* is the last part of the phrase here)
- Time-Phased Planning Data option (because *Phased* is not a preposition)

Use the regular sentence-style capitalization for running text, as used in messages and questions:

- Saving this data will overwrite the current item data file. Continue?
- Do you want to save the changes you made?
Hyphenation

Use the hyphenation rules as outlined in the Hyphenation section of the Microsoft Manual of Style. On a general level, the rules are as follows.

**Hyphenate two words that precede and modify a noun:**
- Buy-from business partner

**Hyphenate compound numerals and fractions**
- Eighty-four

**Do not hyphenate key combinations:**
- CTRL+A
- SHIFT+ENTER
- ALT+F

**Do not hyphenate Latin and Greek prefixes:**
- Cooperating companies
- Interrelated dependencies
- Multilevel E-BOMs
- Multicurrency accounting
- Nonblocking factors
- Parameter setting
- Semifinished item
- Subassembled item
- Transshipment data

Terminology

Use the following rules in order to make the software and the corresponding documentation more consistent:

- Terminology must be generic, that is, not specific to any one line of business.
- Use Baan Standard Terminology as defined in the R&D Multilingual Terminology Database (maintained by KD and BTG).
- Software terminology must be Microsoft-compliant as much as possible.
- Logistic terminology must be APICS-compliant.
- Terminology and phrasing must take the user’s rather than the developer’s perspective.
Terms must be unique within a package as well as across packages. Terms can be made unique by using the full term, for example, Service Area instead of Area. In this way Service Area can be distinguished from Inventory Area.

**Abbreviations**

Use the full term as much as possible. Apply the following rules when you must abbreviate words. For an extensive list of examples, see the Abbreviations table later in this chapter.

**Choose the shortest possible abbreviation that is still understandable:**

- Additional = Adn
- Holdback = Hk

**Do not use periods in abbreviations:**

- Business partner = BP
- Number = No. (always gets a period)

**Use the same abbreviation for the same term in all cases.**

**Abbreviate by omitting vowels instead of by truncation:**

- Standard = Std (not Stand)

**You must include vowels that are the first letter of the word and the letter y if it is the last letter of the word:**

- Company = Cpy (there are exceptions such as currency = Cur, Expiry = Exp)
- Invoice = Ivc

**Use the available space for essential information. In other word, abbreviate the common term and use the space for the specific term:**

- Serialized S-Itm Grp instead of Ser S-Item Group.

**Put a space between the abbreviations. If there is no room, you can write the abbreviations together. Use a capital for the first letter of each abbreviation:**

- SerS-ItmGrp
If the abbreviation is more familiar than the full term, use the abbreviation. For example:

- RPT order
- WIP warehouse
- P-BOM
- E-item
- ZIP code
- IRS-1099

Always include a label for the full term. The translators, localizers, technical writers, and others must be able to see the full term.

### Abbreviations of frequently used terms

<table>
<thead>
<tr>
<th>Term</th>
<th>Abbreviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Additional</td>
<td>Adn</td>
</tr>
<tr>
<td>Aggregated</td>
<td>Agg</td>
</tr>
<tr>
<td>Approved</td>
<td>App</td>
</tr>
<tr>
<td>Business partner</td>
<td>BP</td>
</tr>
<tr>
<td>Change order</td>
<td>ChO</td>
</tr>
<tr>
<td>Code</td>
<td>Cde</td>
</tr>
<tr>
<td>Company</td>
<td>Cpy</td>
</tr>
<tr>
<td>Completed</td>
<td>Cpl</td>
</tr>
<tr>
<td>Component</td>
<td>Cmp</td>
</tr>
<tr>
<td>Control</td>
<td>Ctrl</td>
</tr>
<tr>
<td>Cumulative</td>
<td>Cmv</td>
</tr>
<tr>
<td>Currency</td>
<td>Cur</td>
</tr>
<tr>
<td>Date</td>
<td>Dt</td>
</tr>
<tr>
<td>Effective</td>
<td>Eff</td>
</tr>
<tr>
<td>Engineering</td>
<td>Egr</td>
</tr>
<tr>
<td>Expiry</td>
<td>Exp</td>
</tr>
<tr>
<td>Financial</td>
<td>Fin</td>
</tr>
<tr>
<td>Fiscal</td>
<td>Fsc</td>
</tr>
<tr>
<td>Group</td>
<td>Grp</td>
</tr>
<tr>
<td>Holdback</td>
<td>Hk</td>
</tr>
<tr>
<td>Software text standards</td>
<td></td>
</tr>
<tr>
<td>-------------------------</td>
<td></td>
</tr>
<tr>
<td>Home currency</td>
<td>HC</td>
</tr>
<tr>
<td>Hour</td>
<td>Hr</td>
</tr>
<tr>
<td>Hourly</td>
<td>Hry</td>
</tr>
<tr>
<td>Inventory</td>
<td>Inv</td>
</tr>
<tr>
<td>Invoice</td>
<td>Ivc</td>
</tr>
<tr>
<td>Item</td>
<td>Itm</td>
</tr>
<tr>
<td>Multicurrency</td>
<td>Mltc</td>
</tr>
<tr>
<td>Number</td>
<td>No.</td>
</tr>
<tr>
<td>Operation</td>
<td>Opr</td>
</tr>
<tr>
<td>Payment</td>
<td>Pmt</td>
</tr>
<tr>
<td>Percentage</td>
<td>%</td>
</tr>
<tr>
<td>Period</td>
<td>Prd</td>
</tr>
<tr>
<td>Planning</td>
<td>Pln</td>
</tr>
<tr>
<td>Position</td>
<td>Pos</td>
</tr>
<tr>
<td>Price</td>
<td>Prc</td>
</tr>
<tr>
<td>Product</td>
<td>Prod</td>
</tr>
<tr>
<td>Production</td>
<td>Prdn</td>
</tr>
<tr>
<td>Purchase</td>
<td>Pur</td>
</tr>
<tr>
<td>Quantity</td>
<td>Qty</td>
</tr>
<tr>
<td>Rate</td>
<td>Rt</td>
</tr>
<tr>
<td>Rebate</td>
<td>Rbt</td>
</tr>
<tr>
<td>Receipt</td>
<td>Rct</td>
</tr>
<tr>
<td>Report</td>
<td>Rpt</td>
</tr>
<tr>
<td>Sales</td>
<td>Sls</td>
</tr>
<tr>
<td>Sequence</td>
<td>Seq</td>
</tr>
<tr>
<td>Serialized</td>
<td>Ser</td>
</tr>
<tr>
<td>Service</td>
<td>Srv</td>
</tr>
<tr>
<td>Service item</td>
<td>S-Itm</td>
</tr>
<tr>
<td>Settlement</td>
<td>Stl</td>
</tr>
<tr>
<td>Standard</td>
<td>Std</td>
</tr>
<tr>
<td>Subcontractor</td>
<td>Sctr</td>
</tr>
<tr>
<td>Transfer</td>
<td>Trn</td>
</tr>
<tr>
<td>Unit</td>
<td>Unt</td>
</tr>
<tr>
<td>Warehouse</td>
<td>Wh</td>
</tr>
<tr>
<td>Year</td>
<td>Yr</td>
</tr>
</tbody>
</table>
Currency and quantity formats

In flat text, refer to currencies by means of the international standard banking codes for currencies:

**EXAMPLE**
USD, DEM, GBP, NLG

Internationalization requirements

All terminology, functionality, icons, and examples must be internationally applicable. It must support all regular legal, technical, and functional requirements for the global marketplace. Make sure that the software:

- Supports international aspects such as character sets, code pages, sorting sequences, date formats, currency format, address formats.
- Is multibyte enabled (to support double-byte language character sets, such as Japanese, Chinese, and Korean).
- Is designed in a generic way.
- Does not include any national features, either in text or in icons.
- Recognizes the cultural aspects of colors, icons, and signs.

For more details, refer to the BTG web site, which also contains links to the Internationalization sections of [http://www.microsoft.com](http://www.microsoft.com).

General translation requirements

All Baan software products are translated into 16 languages. The following rules and guidelines facilitate the translation process:

- Separate technology (code) and text.
- Be concise: every word written costs $3 to translate.
- Use market-compliant text coding and storage formats.
- Design products with a modular structure (to allow time-phased translation).
- Make products, terminology, and examples generic rather than specific.
- Store all text elements in forms, reports, tables, and menus as labels or resources (no hard-coded strings or options).
- Recognize that most languages require 25-100% additional characters/space compared to English, e.g.: BOM is translated as Nomenclature (FR).
- Note that one English word may require different translations in different contexts.
Recognize that different languages may use a different word order, for example: Production Order (US) is translated as Ordre de Fabrication (FR).

For more details, refer to the BTG web site on the R&D extranet and to http://www.microsoft.com

See also Translatability Restrictions on Labels, at the end of this chapter.

Text standards by software component

Packages

The highest-level software component in the BaanERP product is the package. A package is a set of functionally related modules seen from the user’s point of view.

- The Baan core suite of packages is named BaanERP.
- Other Baan products are named BaanSCS, BaanFOS, BaanCOS, and so on.
- The individual packages are named Baan Common Data, Baan Order Management, and so on (without the addition “ERP”).

Modules

A package consists of various modules. A module is primarily a technical grouping of sessions and other software components.

- Module codes and descriptions must be logical, linguistically correct, and internationally acceptable. Avoid the use of vowels (a/e/i/o/u) in the codes. The use of vowels is more likely to create undesired words or letter combinations than consonants. BTG can check this for all 16 A and B languages.
- The three-letter module code is derived from the English module description.
- References to module codes in labels, messages, and texts are written in uppercase, without dots or spaces, for example: … the Routing (ROU) module…
- Module descriptions must be nouns to distinguish modules from sessions.

Example

Typical names of module descriptions are:

- Purchase Control
- General Ledger
- Text Management
Business objects

Business objects are sets of functionally related groups of sessions. Primarily, business objects serve to facilitate the implementation and documentation process.

- Business object descriptions must be nouns to avoid confusion with session descriptions and must be written in headline-style capitalization.
- Business object descriptions must be unique in a package. For example, instead of just Master Data, use: SMI Master Data (tdsmi00010).
- Business object descriptions have a maximum length of 60 characters.

Subfunctions

Below the level of business objects, subfunctions are functionally related groups of sessions or fields in a business object. They allow you to define generic texts for a set of sessions or fields.

- Subfunction descriptions must be unique within a package.
- Subfunction descriptions have a maximum length of 60 characters.

Examples of subfunction description:
- Sales Order Processing Procedure
- Sales Order Processing Example

Sessions

Technically, a session is the total of a main table, program script, session title, one or more forms, and one or more reports. Session codes and descriptions are important because they are the framework (table of contents) of the application.

- Use user-friendly and clear session descriptions. Avoid long titles such as: Print Inventory Data by Item Group/Item/Warehouse/Location/Order. Instead, use: Print Inventory by Item Group and Item.
- The different types of sessions have different naming conventions, which are described in the Basic Concepts chapter and the Window Components chapter.

Examples of session titles (the session descriptions):
- Addresses
- Business Partners
- Sales Contracts - History
If the session description indicates an action, use one of the standard session verbs:

<table>
<thead>
<tr>
<th>Standard verb</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Calculate</td>
<td>Arithmetic operations (calculations)</td>
</tr>
<tr>
<td>Clear</td>
<td>Clean-up data (do not use Purge)</td>
</tr>
<tr>
<td>Approve</td>
<td>Approving a record before processing</td>
</tr>
<tr>
<td>Duplicate</td>
<td>Duplication</td>
</tr>
<tr>
<td>New</td>
<td>Manual action to create a new record</td>
</tr>
<tr>
<td>Display</td>
<td>To show data on screen</td>
</tr>
<tr>
<td>Delete</td>
<td>Removing data</td>
</tr>
<tr>
<td>Enter</td>
<td>Occasional input of data</td>
</tr>
<tr>
<td>Generate</td>
<td>Automatic creation of (a range of) new records</td>
</tr>
<tr>
<td>Print</td>
<td>Output to a report on paper or to a file</td>
</tr>
<tr>
<td>Process</td>
<td>Treatment or conversion of data</td>
</tr>
<tr>
<td>Post</td>
<td>Financial recording of data</td>
</tr>
<tr>
<td>Select</td>
<td>Marking records (for further processing)</td>
</tr>
<tr>
<td>Transfer</td>
<td>Conveying data from one place (package) to another. For example, transfer data to central invoicing.</td>
</tr>
</tbody>
</table>

**Forms**

Each session has one form in the Dynamic Form Editor (DFE). The form’s description (the form title) is taken from the session description. All other text elements in the form must be stored as labels.

- All text elements in forms must be defined as labels.
- All form field descriptions must be defined as labels.
- Do not use abbreviations for form tab descriptions and for form command buttons. Preferably use one or two key words to define the function.
- For the forms that show a data view, the session title must be singular.
- Reserve the maximum length for form field descriptions on the form (the form label field length). This ensures that the label still fits on the form after translation.
Space needed for translated form field descriptions

Foreign language translations can be up to 50% longer than the equivalent English word, term or phrase. Label fields must therefore be as long as the available space in the form layout allows, even if the English label description is only short.

**Example**

Some typical form field descriptions are:

- BOM = Stückliste (German), Nomenclature (French)
- Where-Used = Teileverwendungsnachweis (German)
- Work Center Yr Utilization Histogram = Abteilungsjahreskapazitätsauslastungshistogram

Form pages (form tabs)

A form can consist of multiple form pages, each with their own form page label.

- If the tab or form page only contains the key fields and one group box, then the label can be the same as the group box label.

See also Labels, later in this chapter.

Form commands (form-specific options)

Each form can have:

- Standard commands
- Form commands that are specific for the form (form-specific options).

Form commands occur as menu commands and as command buttons.

See Menus and Command Buttons, later in this chapter.

Menus

The sessions in an application’s menu tree are grouped by means of menus. A second type of menu are the pulldown menus in the sessions.

Menus in the menu browser

- Menu titles must be unique within a package.
- The menu titles in the menu browser are the names of the BaanERP packages, modules, and business objects. The rules for the package names, module names, and business object names apply.
- Do not add the word Menu behind the title of each menu.
- All additional text elements are stored as labels.
The menu browser must not contain action dialogs. Only sessions containing objects are allowed in the menu browser. See the Window Types chapter for detailed guidelines.

**Drop-down menus and menu commands**

Drop-down menus can be accessed via the menu bar on top of the window.

- The menu titles in the menu bar must be single words; no multiword terms or abbreviations allowed. For example, File, Edit, View, Tools, Specific.
- The menu commands must not contain more than three words (not counting words such as “to” and “and”), except when they refer to session descriptions.
- Refer to the Windows components chapter for a detailed description of the guidelines for drop-down menus. These guidelines are about grouping, labeling, and positioning of menu items.

**Toolbar buttons**

- Icons for the toolbar buttons will be designed by the GUI Consultancy group.
- Do not use letters or words on toolbar buttons.
- Make sure icons are internationally recognized and acceptable.

**ToolTips**

- Contain the command name.

**Charts**

Charts are graphical displays of information.

- Chart titles follow the rules for session titles. See Sessions, earlier in this chapter.
- Chart options follow the rules for form commands. See Form commands, earlier in this chapter.

**Reports**

The layouts of printed documents are defined as reports.

- Report titles must be unique within a package.
- All text elements must be stored as labels.
Report descriptions do not include the verb Print.

**Example**

Typical report titles are:
- Purchase Orders by Supplier and Item
- Packing Slip
- List of Reserved Commissions/Rebates

**Tables**

Table descriptions are visible to the end user, therefore, table descriptions must be consistent.

**Table descriptions**

- Table descriptions must be unique within a package.
- Table descriptions must be nouns in plural. Example: Customers.
- Table descriptions are used in:
  - Session generator
  - Generated reference messages
  - Online Help (via %TB hypertext code)
  - User documentation

**Example**

Typical table descriptions are:
- Sales Parameters
- Prices by Price List and Item
- Sales

**Table fields**

- Do not use field codes (orno, cuno) as field descriptions.
- Do not include additions like [y/n] and [Unit] in labels. Take into account that table field labels are used in the online help, generated documentation, Easy SQL, and the session generator.

**Table index descriptions**

- Do not use field codes (orno, cuno) as field descriptions.
- Index parts are separated by commas (example: Warehouse, Item Group, Item)
Domains

Domains define the valid input characters for a particular field type. The only domain type visible to the end-user is "Enumerate" (Enum).

Enumerates and sets

- Enumerate and set descriptions are left aligned.
- The initial letter of enums and sets must, if possible, be unique, in order to allow the user to select an enum value with a single keystroke. An alternative is to start the description with a unique number or code, such as 1, 2, 3 or a, b, c.
- If enum descriptions cannot be shown in full in some situations, it can also be useful to start the description with a unique number or code. This is sometimes the case on reports.

**Example**

Typical numbered enum descriptions are:

- 1. By Order Quantity
- 2. By Order Date
- 3. By Supplier

Messages and Questions

See Message Boxes in the Window Types chapter for the detailed guidelines on writing messages and questions. Have your message text reviewed by a technical writer.

Apply the following rules to messages:

- Be short and business-like; avoid long sentences with complex verbal constructions.
- Do not be patronizing or overly polite (by using please in every message, for example).
- As a rule: the first part of the message explains what is wrong; the second part explains how to solve the situation.
- Do not capitalize all key words (except when referring to session descriptions, fields, parameters, enums, and so on).
- Place the description of software elements between single quotes when you refer to sessions, fields, parameters, and enums.
- Do not link one message to another. That can cause translation problems.
- Use language-specific date formats %Dxxx.
Messages informing the user about form commands must start with the option code followed by a description of that option in Microsoft style: 
CTRL+G = Global Receipt.

**EXAMPLE**

Typical messages are:

- Note! Sales below margin for position %d
- First set the 'Discount' parameter in the session 'Item Data' (tcibd0101m000)
- Date earlier than current date (not: Date less than / before current date)
- Line items not found: copying not possible
- Contract number not allowed; number too large
- Deliveries already entered; first confirm backorder before changing

**Labels**

This section contains:

- General rules for label text and label usage
- Specific rules for labels of different software components
- An explanation of the restrictions that translatability puts on labels

Labels are text elements that are stored in a central database and can be reused in different places. All text elements in forms, reports and menus must be entered as labels.

A label has a label code and a label description. You use the code to link the label to a form field. The label description defines the text that is displayed on the screen or printed on the report. See the Window Controls chapter for additional guidelines on labels.

Labels can and must be reused as much as possible. However, there are some restrictions as described in Translatability Restrictions, later in this chapter. You can only reuse a label if it refers to the same entity with the same meaning in the new context.

Using labels has many advantages:

- Language-independent forms, reports and menus.
- Reduced text volumes means lower localization, translation and maintenance cost.
- Consistent terminology helps the user to use the product correctly.
For the developer, this means:

- Do not use any hard-coded text in form, report, and menu layouts; use labels.
- No language-dependent forms, reports, and menus are allowed; all forms, reports, and menus are stored under language code 2 (the development language). Forms, menus and reports in other language codes must be linked to language 2.
- Labels must be reused in order to minimize the total amount of text to be stored and translated, but within the restrictions described in Translatability Restrictions, later in this chapter.
- To stop using a label, you must copy the label to the current VRC and select the Expired check box. Note that, if you delete a label, other variants of the same label code may still be taken from previous VRCs.
- To change a label, you must copy the existing label to the current VRC and select the Expired check box. Then create a new label with the same label code.

**General rules for labels**

The following rules apply to the labels of all types of text elements.

- Labels follow the capitalization rules of the corresponding text element
- Labels do not end with a period (.), exclamation mark (!) or question mark (?)
- The following characters are language-dependent and must therefore also be stored as labels: !, %, 1st, 2nd, 3rd, kg, USD, HFL
- The % sign serves as line break character for 2- and 3-line labels; to display the percent sign in a label, type “%%”
- Do not use spaces to align labels.
- Instead of abbreviating 4 or 6 words, try to summarize the same meaning in fewer words. Example: Maximum Number of Years to Retain History = History Years

Abbreviations must be informative and consistent. See Abbreviations, earlier in this chapter.
Field labels

Labels identify the meaning of a control. For input fields, the field label serves as a prompt that tells the user what to enter.

For field labels the following rules apply:

- Identify the meaning of the field or control.
- Are short and meaningful.
- Do not end with a period (.), exclamation mark (!) or question mark (?)
- Have a standard length of 20 positions in single-occurrence screens.
- Have a standard length of 2 positions.

Note: When you create a form you must reserve the maximum space for labels to allow for longer labels after translation. See also the section 'Forms,' earlier in this chapter.

Check box labels

In addition to the rules for field labels, the following rules apply specifically to check box labels:

- Stand to the right of the check box
- Describe the selected option
- Are maximally 1 line high

Command button labels

- Use the imperative form ("Find," "Modify")
- Must fit on the button
- Do not use abbreviations. Summarize by using one or two catchwords instead.
- Example, do not use: Maint.Ord Dt. Instead, use e.g.: Order Date

Form page (tab) labels

- Use meaningful titles to describe what type of information is found on the form
- If the tab or form page only contains the key fields and one group box, then the label can be the same as the group box label.
Group box labels

- Describe a common feature of the fields in the group box
- Must not be identical to any of the field labels in the group box
- Must be left-aligned
- If the form page only contains the key fields and one group box, the group box label is the same as the form page label.

Translatability restrictions on labels

The fact that labels must be translated puts certain restrictions on how they are defined.

Only reuse labels if they refer to the same term

Labels should be reused. However, only reuse labels that refer to the same term (entity or concept). Recognize the fact that the same word may refer to different terms and may therefore require different labels.

For example: both German and French have two different words for purchase order and sales orders. This implies that you cannot use one single label for the word Order, which may mean Purchase Order in one context, Sales Order in another, or even Sequence in a third instance.

For instance, ‘order’ can be translated as follows:

- Order = Sequence       GE = Reihenfolge
- Order = as in: Order now! GE = Bestellen Sie!
- Order = Purchase Order  GE = Bestellung
- Order = Sales Order     GE = Auftrag

Do not construct a term or phrase by linking two or more labels

Multiword terms or phrases (two or three words or even complete sentences) must always be defined as one single label. Don't construct such word groups by placing several one-word labels immediately after or above each other.

This type of construction is impossible to translate, because the foreign language:

- Can have a different word order
- Can use different suffixes depending on a word's grammatical function or gender
- Can use different terms in different contexts for what in English looks like the same word.
A single label can be used as follows:

- "Print " + " Text " cannot be translated into German as "Text drucken".
- Instead, define one label for "Print Text".

**All descriptions for a given label must be fully interchangeable**

In order to reuse an existing label in a field that is shorter than the label description, a shorter or abbreviated description must be created under the same label code. Each label code may have a number of descriptions, with different length and height, provided that all these label variants have the same meaning (are fully interchangeable).

For example, Total Discount and Discount do not have exactly the same meaning and therefore are not fully interchangeable.

Example:

<table>
<thead>
<tr>
<th>Label</th>
<th>Height</th>
<th>Length</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>tdsls041.opol</td>
<td>1</td>
<td>9</td>
<td>Item Cat.</td>
</tr>
<tr>
<td>tdsls041.opol</td>
<td>1</td>
<td>13</td>
<td>Item Category</td>
</tr>
<tr>
<td>tdsls041.opol</td>
<td>2</td>
<td>4</td>
<td>Item%Cat.</td>
</tr>
</tbody>
</table>

**Do not select a specific label by playing around with the label field length**

Do not define different descriptions under the same label code, and then make the shorter variant appear on a particular form by simply truncating the label field length. That does not guarantee that after translation or conversion to Dynamic Forms the same label will still appear.

WRONG:

<table>
<thead>
<tr>
<th>Label</th>
<th>Height</th>
<th>Length</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>tcgen.discount</td>
<td>1</td>
<td>8</td>
<td>Discount</td>
</tr>
<tr>
<td>tcgen.discount</td>
<td>1</td>
<td>14</td>
<td>Total Discount</td>
</tr>
</tbody>
</table>

CORRECT:

<table>
<thead>
<tr>
<th>Label</th>
<th>Height</th>
<th>Length</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>tcgen.discount</td>
<td></td>
<td>8</td>
<td>Discount</td>
</tr>
<tr>
<td>tcgen.discount:01</td>
<td>1</td>
<td>14</td>
<td>Total Discount</td>
</tr>
</tbody>
</table>
Use different label codes for labels with different meanings or features

The same rules apply to labels that contain the same text but differ in features like upper/lower case, brackets, spacing, and so on. These texts are not interchangeable. Therefore these variant descriptions must be stored as variant label codes by adding a sequence number to the label code (thus placing them together in the label file).

**Example**

Typical label variants are:

- `tds`s.`cost` Costs
- `tds`s.`cost.01` COSTS

**Example**

Consider the combination of two fields Year and Period in a specific form:

- Year / Period: 1998 -12

You must define one label for Year / Period instead of using a label for Year followed by a label for Period.
### Standard label codes

Where appropriate, use the following standard label codes with given length:

<table>
<thead>
<tr>
<th>Description</th>
<th>Where Used</th>
<th>Label Code</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>[kg]</td>
<td>form / report</td>
<td>tcgen.kg</td>
<td>4</td>
</tr>
<tr>
<td>[hour]</td>
<td>form / report</td>
<td>tcgen.hour</td>
<td>6</td>
</tr>
<tr>
<td>[%]</td>
<td>form / report</td>
<td>tcgen.percsign</td>
<td>3</td>
</tr>
<tr>
<td>Both</td>
<td>form / report</td>
<td>tcgen.both</td>
<td>4</td>
</tr>
<tr>
<td>Code</td>
<td>form / report</td>
<td>tcgen.code</td>
<td>6</td>
</tr>
<tr>
<td>Company</td>
<td>report header</td>
<td>tcgen.h.comp</td>
<td>10</td>
</tr>
<tr>
<td>Date</td>
<td>report header</td>
<td>tcgen.h.date</td>
<td>6</td>
</tr>
<tr>
<td>Date</td>
<td>menu / report header</td>
<td>tcgen.date</td>
<td>5</td>
</tr>
<tr>
<td>Description</td>
<td>form / report</td>
<td>tcgen.dsca</td>
<td>30</td>
</tr>
<tr>
<td>Inquiries</td>
<td>Menu</td>
<td>tcgen.inquir</td>
<td>11</td>
</tr>
<tr>
<td>Maintenance</td>
<td>Menu</td>
<td>tcgen.maintain</td>
<td>11</td>
</tr>
<tr>
<td>Miscell</td>
<td>Menu</td>
<td>tcgen.misc</td>
<td>11</td>
</tr>
<tr>
<td>Page</td>
<td>report header</td>
<td>tcgen.page</td>
<td>10</td>
</tr>
<tr>
<td>Procedure</td>
<td>Menu</td>
<td>tcgen.proced</td>
<td>11</td>
</tr>
<tr>
<td>Processing</td>
<td>Menu</td>
<td>tcgen.process</td>
<td>11</td>
</tr>
<tr>
<td>Reports</td>
<td>Menu</td>
<td>tcgen.reports</td>
<td>11</td>
</tr>
<tr>
<td>From</td>
<td>Form</td>
<td>tcgen.from</td>
<td>5</td>
</tr>
<tr>
<td>from</td>
<td>Form</td>
<td>tcgen.from.01</td>
<td>5</td>
</tr>
<tr>
<td>To</td>
<td>Form</td>
<td>tcgen.to</td>
<td>5</td>
</tr>
<tr>
<td>to</td>
<td>Form</td>
<td>tcgen.to.01</td>
<td>5</td>
</tr>
<tr>
<td>Time</td>
<td>form / report</td>
<td>tcgen.time</td>
<td>5</td>
</tr>
<tr>
<td>Text</td>
<td>form / report</td>
<td>tcgen.text</td>
<td>5</td>
</tr>
<tr>
<td>Period</td>
<td>form / report</td>
<td>tcgen.period</td>
<td>7</td>
</tr>
<tr>
<td>Options</td>
<td>form / report</td>
<td>tcgen.options</td>
<td>7</td>
</tr>
<tr>
<td>Settings</td>
<td>form / report</td>
<td>tcgen.settings</td>
<td>15</td>
</tr>
<tr>
<td>Selection</td>
<td>form / report</td>
<td>tcgen.selection</td>
<td>15</td>
</tr>
<tr>
<td>Amount</td>
<td>form / report</td>
<td>tcgen.amount</td>
<td>7</td>
</tr>
</tbody>
</table>

All applications can make use of these general tcgen labels.

Because Orgware and Tools can also be used as stand-alone products (without TC), the general labels for these packages must be encoded as "tggen" and "ttgen".
Appendix A. Checklist

Four checklists are provided in this appendix. These checklists summarize the guidelines covered in this style guide. You should use these checklists in your design of the following window types:

- Overview window
- Details window
- Printing & processing
- Parameters

Remember, the objective of the guidelines in this style guide is to benefit the users, not to enforce a rigid set of rules. Consistency within and across Baan software makes it easier for a user to transfer skills from one task to another.

When you want to diverge from or extend these guidelines, contact a GUI consultant.

**Procedure**

- Use the checklist that corresponds to the window you are designing.
- Mark the guideline when the guideline is applicable to your window design.
- Provide links to the corresponding part in this style guide. You can use these links to check whether all the guidelines are met.
- Remember that unchecked guidelines indicate which GUI aspects must be improved and/or redesigned in order to meet the guidelines.

**Checklist overview and combined window**

**General**

- The window follows the synchronization rules.

**Size**

- The window fits on an 800 x 600-resolution screen. Check the window size on a system running with this resolution.
Group fields (only combined window)
- The group fields are designed according to the guidelines.
- The labels are aligned according to the guidelines.

Grid & column heading
- Each column has a column heading.
- The label of the column heading is top-left aligned.
- The column heading is no more than three lines.
- The fields in the grid are left aligned. Numeric fields are right aligned.

Components
- No multiple form pages and no command buttons exist.
- The combined window consists of a standard title bar, menu bar, toolbar, grid, scroll bar, and status bar.

Specific menu
- The Specific menu only contains menu items that are specific for this window.
- The menu items are developed according to the guidelines and the software text standards.
- The menu items have the appropriate shortcut keys and access keys.

Second toolbar buttons
- The buttons are taken from the Baan toolbar button archive.
- New toolbar buttons must be designed and approved by GUI-Consultancy.
- If a Microsoft button is used, it must have the same function as the original button.
- Each button always has the same function within Baan.
Checklist details window

General
- The details window follows the synchronization rules.

Size
- The details window of type 1 is designed within the 80 columns of the form editor.
- The details window of type 3 is designed within the 80 columns of the form editor.
- The window fits on an 800 x 600 resolution screen. Check the window size on a system running with this resolution.

Form pages
- The form page is designed according to the standard window layout; this layout includes dividing information, alignment, and field order.
- The labels placed on the form page are designed according to the guidelines and the software text standards.
- The fields and the key fields are positioned on the form page(s) according to the guidelines.
- If the details window contains multiple form pages, these are designed according to the guidelines.
- If the details window contains multiple form pages, a tab control is provided, that includes meaningful labels and shortcut keys.

Components
- The details window has no menu bar, toolbar, status bar and scroll bar.
- The details window contains a title bar, and form page(s). These components are designed according to the guidelines.

Controls
- The command buttons on the details window follow the standard guidelines, which include the position, labels, disabling, default button, and the sequence and combination of command buttons.
- The group box on the details window has a meaningful group-box label, is correctly aligned, and has the right size.

- Multiple group boxes placed under each other and/or next to each other are aligned according to the guidelines.

- The check boxes are positioned and aligned according to the guidelines.

- The **From-To** selection ranges are designed according to the guidelines. These include specific rules for the **From-To** columns, the **From-To** rows and the combination of columns and rows.

- The browse area and the Details button are used and designed according to the guidelines.

**Shortcut keys**

- The following shortcut keys are used in the details:

<table>
<thead>
<tr>
<th>Command</th>
<th>Shortcut Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cut</td>
<td>CTRL+X</td>
</tr>
<tr>
<td>Copy</td>
<td>CTRL+C</td>
</tr>
<tr>
<td>Paste</td>
<td>CTRL+V</td>
</tr>
<tr>
<td>Browse</td>
<td>CTRL+B</td>
</tr>
<tr>
<td>Next Control</td>
<td>TAB</td>
</tr>
<tr>
<td>Previous Control</td>
<td>SHIFT+TAB</td>
</tr>
<tr>
<td>Next Form Page</td>
<td>CTRL+TAB</td>
</tr>
<tr>
<td>Previous Form Page</td>
<td>CTRL+SHIFT+TAB</td>
</tr>
</tbody>
</table>

Baan GUI Style Guide
4
Checklist printing & processing window

Size
- The printing and processing window of type 4 is designed within the 80 columns of the form editor.
- The window fits on an 800 x 600-resolution screen. Check the window size on a system running with this resolution.

Form pages
- The printing & processing window contains one or more form pages. These pages are designed according to the standard window layout. Multiple form pages include specific form page labels.
- The labels and fields placed on the form page are designed according to the guidelines. The labels also follow the software text standards.

Components
- The printing & processing window has no menus, toolbar, status bar, or scroll bar.
- This window contains a title bar, and form page(s). These components are designed according to the guidelines.

Controls
- The command buttons on the printing & processing window are predefined buttons, which are designed according to the guidelines.
- The group boxes on the printing & processing window have predefined labels, contents, and sequence. These group boxes are correctly aligned and have the right size.
- Multiple group boxes placed under each other and/or next to each other are aligned according to the guidelines.
- The check boxes are positioned and aligned according to the guidelines. The group of check boxes is positioned in the group box called Options.
- The From-To selection ranges are designed according to the guidelines. These include specific rules for the From-To columns, the From-To rows and the combination of columns and rows. The selection ranges are positioned in the first group box on the form page with the label Selection Range.
- The browse area and the Details button are used and designed according to the guidelines.

**Shortcut keys**
- The following shortcut keys are used in the printing & processing window:

<table>
<thead>
<tr>
<th>Command</th>
<th>Shortcut Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cut</td>
<td>CTRL+X</td>
</tr>
<tr>
<td>Copy</td>
<td>CTRL+C</td>
</tr>
<tr>
<td>Paste</td>
<td>CTRL+V</td>
</tr>
<tr>
<td>Browse</td>
<td>CTRL+B</td>
</tr>
<tr>
<td>Next Control</td>
<td>TAB</td>
</tr>
<tr>
<td>Previous Control</td>
<td>SHIFT+TAB</td>
</tr>
<tr>
<td>Next Form Page</td>
<td>CTRL+TAB</td>
</tr>
<tr>
<td>Previous Form Page</td>
<td>CTRL+SHIFT+TAB</td>
</tr>
</tbody>
</table>
Checklist parameters window

Size
- The parameter window of type 1 is designed within the 80 columns of the form editor.
- The window fits on an 800 x 600-resolution screen. Check the window size on a system running with this resolution.

Position
- The parameters windows are correctly positioned in the menu browser.

Form page
- The parameter window contains one or multiple form pages. These pages are designed according to the standard window layout.
- The labels and fields placed on the form page(s) are designed according to the guidelines. The labels follow the software text standards.

Components
- The parameters window has no menus, toolbar, status bar, or scroll bar.
- This window contains a title bar and form page(s). These components are designed according to the guidelines.

Controls
- The command buttons on the parameters window are predefined buttons, which are designed according to the guidelines.
- The group boxes on the parameters window are designed according to the guidelines. Multiple group boxes placed under each other and/or next to each other are aligned according to the guidelines.
- The check boxes are positioned and aligned according to the guidelines.
- The browse area is used and designed according to the guidelines.
Shortcut keys

- The following shortcut keys are used in the parameters window:

<table>
<thead>
<tr>
<th>Command</th>
<th>Shortcut keys</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cut</td>
<td>CTRL+X</td>
</tr>
<tr>
<td>Copy</td>
<td>CTRL+C</td>
</tr>
<tr>
<td>Paste</td>
<td>CTRL+V</td>
</tr>
<tr>
<td>Browse</td>
<td>CTRL+B</td>
</tr>
<tr>
<td>Previous form page</td>
<td>CTRL+SHIFT+TAB</td>
</tr>
<tr>
<td>Next form page</td>
<td>CTRL+TAB</td>
</tr>
</tbody>
</table>