

BaanERP 5.0c

Component model

A publication of:

Baan Development B.V.
P.O.Box 143
3770 AC Barneveld
The Netherlands

Printed in the Netherlands

© Baan Development B.V. 1999.
All rights reserved.

The information in this document is subject to change without notice. No part of this document may be reproduced, stored or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of Baan Development B.V.

Baan Development B.V. assumes no liability for any damages incurred, directly or indirectly, from any errors, omissions or discrepancies between the software and the information contained in this document.

Document Information

Code: U7330A US
Group: User Documentation
Edition: A
Date: November, 1999

Table of contents

1	Introduction	1-1
2	The component model	2-1
	Components	2-1
	Component applications	2-5
	Sub applications	2-8
3	Component conversion information	3-1
4	Component conversion	4-1
	To convert the applications where used	4-1
	To rename the component and component release where used	4-4
5	Export/import component models	5-1
	To export component data	5-1
	To import component data	5-3
	To import a component model in XML format	5-4
6	To create a report of application use	6-1
7	Glossary	7-1
8	Appendix - Language codes	8-1

Component model
ii

About this document

This document explains the component model that is used in DEM models.

For more information on how to perform a conversion see *U7136B US*.

1 Introduction

The component model is introduced in BaanERP 5.0c references exist in business process activities and support applications to individual components of the component model. A component is, for example, BaanERP or a third party product. With the component model you can also convert the DEM models from one component release to another. To be able to convert a DEM model from one component release to another, conversion information between these component releases must be maintained.

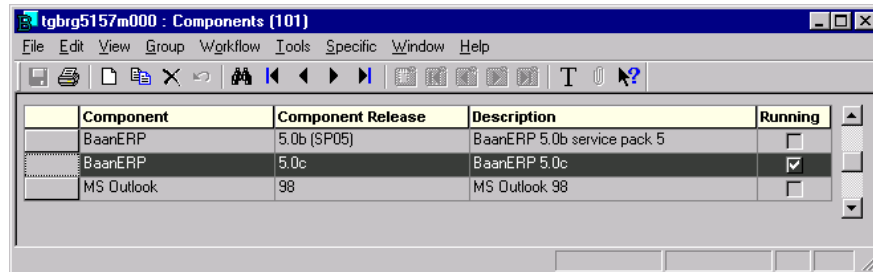
This document contains the following information:

Chapter 1	Introduction
Chapter 2	How to set up a component
Chapter 3	How to maintain the conversion information between components
Chapter 4	How to perform an actual conversion of a DEM model
Chapter 5	How to export or import a component model between systems or versions
Chapter 6	How to print a report of where applications are used in DEM models

Component model
1-2

2 The component model

A component is a layer that describes some general aspects of the applications it contains. For example, how the applications must be represented, how they can be started, and what the applications support (only applicable for Baan components).



The screenshot shows a window titled 'tgbg5157m000 : Components (101)'. The window has a menu bar with 'File', 'Edit', 'View', 'Group', 'Workflow', 'Tools', 'Specific', 'Window', and 'Help'. Below the menu bar is a toolbar with various icons. The main area contains a table with the following data:

Component	Component Release	Description	Running
BaanERP	5.0b (SP05)	BaanERP 5.0b service pack 5	<input type="checkbox"/>
BaanERP	5.0c	BaanERP 5.0c	<input checked="" type="checkbox"/>
MS Outlook	98	MS Outlook 98	<input type="checkbox"/>

Figure 2-1, Component releases

Components

For a component you need to enter the following:

On the **Details** tab (see figure 2-2)

- **Component**
Name of the component (for example, BaanERP).
- **Component Release**
Name of the release (for example, 5.0c).
- **Description**
Short description of the component release.

- **Model-Time Details** group box
Here you can enter the information that is needed to represent the component at model time.
 - **Component Icon**
An icon that visually represents the component in the Dynamic Menu Browser and Process Viewer.
 - **Component Type**
Because the component release description can differ from the official Baan release description, you can select the description of the official Baan release. It is necessary information because of standard conversions between Baan releases.
 - **Singleton**
Select this check box if the component release consist of only one application.
 - **Text**
Any information you want to provide on the component release.
- **Run-Time Details** group box
Here you can enter the information needed to start the applications of the component.
 - **Client/Server**
Specify whether the component release runs on the server or on the client.
 - **Interface**
Select the type of interface supported by the component release.
 - **Executable**
With a command-line interface specify the executable for the component release at run time.
 - **ProgId**
With a COM interface specify the program identification of the component release as it is known by COM. You can find the program identification in the COM interface specification of the component release.

– Running Baan Component

One Baan component release must be set as the running Baan component, (see the right column in Figure 2-1). Applications defined in the running Baan component are started directly and therefore need no interface defined in the **Run Time Details** group box of the Components (tgbrg5157m000) details session. A component release can be set to the running Baan component by using the DEM Parameters (tgbrg0135s000) session. You can easily start the DEM Parameters session through the **Specific** menu of the Components (tgbrg5157m000) session.

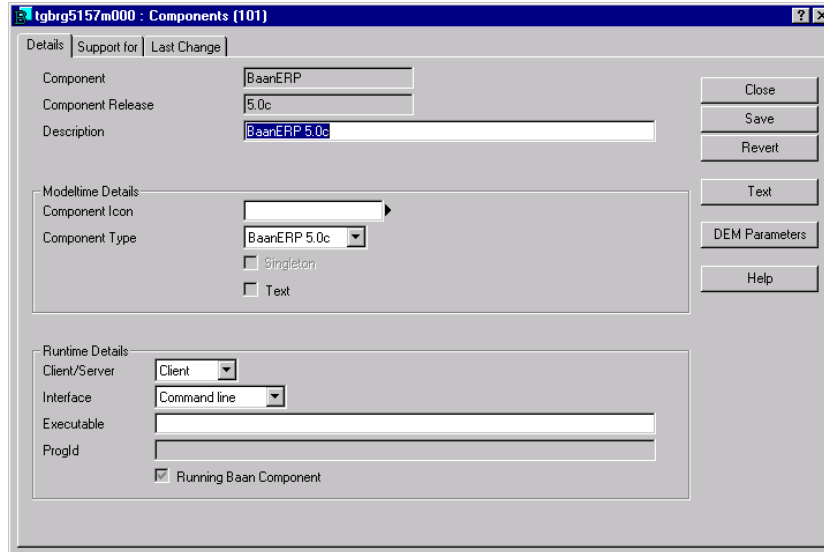


Figure 2-2, Component Release (Details tab)

Support for tab (only applicable for Baan component releases, see figure 2-3)

- **Authorizations**

The authorization types supported by the component release. When modeling an activity you can only select the authorization types selected for the component release.

- **Standard Application Options**

The default options available in the applications defined for the component release. When modelling an activity you can only select the standard application options selected for the component release.

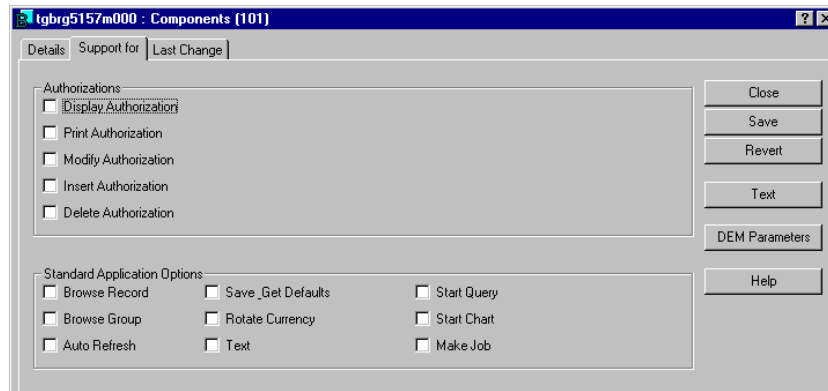


Figure 2-3, Component Release (Support For tab)

Component applications

For a component release, applications must be defined through the Applications by Component (tgbg5155m000) session. You can start this session through the **Specific** menu of the Components (tgbg5157m000) session.

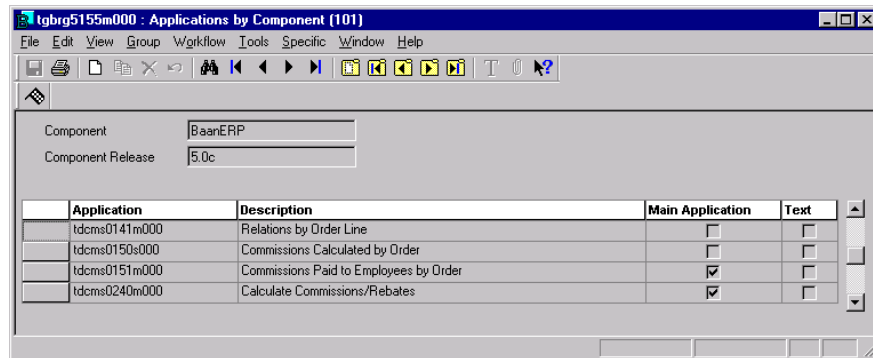


Figure 2-4, Applications by component release

To import the Baan sessions of the running Baan component:

- 1 Choose the **Import applications from Running Baan Component** command on the **Specific** menu.
- 2 The Import Applications from Running Baan component (tgbg5255m000) session is started, where you can specify a range of applications (Baan sessions) to import and some options.
 - Select the **Overwrite Existing Values** check box to overwrite applications that already exist in the component release. The arguments and text of an application are never overwritten.
 - Select the **Remove Applications that No Longer Exist** check box to remove the applications from the component release (that is marked as the running Baan component) that no longer exist in the installed Baan release.
 - Select the **Test Run** check box to simulate what would happen if you run the import. To visualize the simulation, select the **Print** check boxes.
 - Select the **Print** check boxes to print the changes to the component release that is marked as the running Baan component.

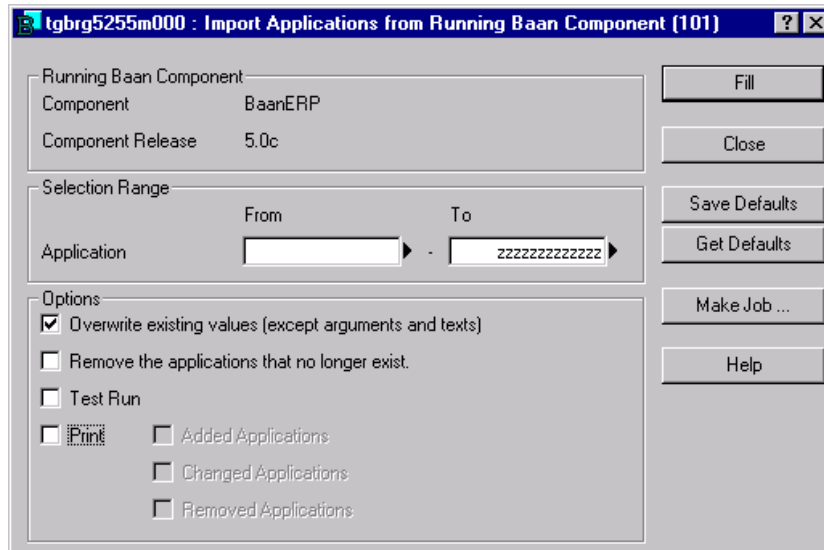


Figure 2-5, Import Applications from Running Baan Component

If you insert a new application manually by clicking the **New** button, the following information on the application must be given:

- **Application**
 - The component is the running Baan component: use the Baan session code.
 - Component uses the command line interface: use the ID of the executable application.
 - The component uses the COM interface: use the ID of the application as it is known by the COM interface.
- **Description**
A short description of the application
- **Details** group box
 - **Argument**
The argument that is passed on to the component release at run time. To reference the arguments specified for an activity, you can use \$ n stands for the subsequential activity's argument. In other words, \$1 is the first argument of the activity, \$ 2 is the second, and so on.
 - **Main Application**
Indicates whether the application is a main application that can be started directly. Only main applications can be linked to an activity or support application.
 - **Text**
Additional information on the application

The screenshot shows a dialog box titled "tgbg5155m000 : Applications by Component (101)". It contains the following fields and controls:

- Component:** BaanERP
- Component Release:** 5.0c
- Application:** tdcms0100s000
- Description:** Commissions/Rebates Parameters
- Details:**
 - Argument:** (empty text field)
 - Main Application
 - Text
- Last Change:**
 - Updated with Session: tgbg5255m000 Import Applications from Running Baan Component
 - Date: 25-10-1999
 - Time: 9:56:22
 - User: jvddobbe

Buttons on the right side include Close, Save, Revert, Text, Start Application, and Help.

Figure 2-6, Application information

Sub applications

Applications can have one or more sub applications. You can define the sub applications through the **Sub Applications by Application** command on the **Specific** menu of the Applications by Component (tgbrg5155m000) session.

NOTE: The sub applications can be nested. For an application to have sub applications, it must be a main application.

To set authorizations for applications that can be started via a main application (for example the applications that you can start from the **Specific** menu) you must model sub applications.

3 Component conversion information

This chapter describes how to maintain the conversion information that is needed in Chapter 4 to convert applications that are used in business processes and support applications from one component release to another.

This information is not used by the Rename Component & Release Codes (tgbg5220m000) session, which is used only to rename a component release of applications used in business processes and support applications.

The conversion information between component releases is maintained in the Conversion of Applications (tgbg5150m000) session.

For the standard conversion between Baan releases, the conversion information is maintained and supplied.

For conversions other than standard conversions, you need to insert your own conversion data.

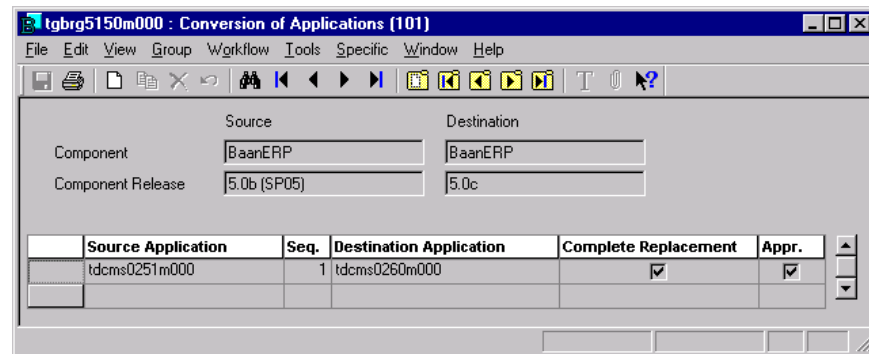


Figure 3-1, Conversion of Applications

A conversion can be defined between a source component release and a destination component release. The source and destination component release have respectively source and destination applications defined.

If, in a business process or a support application, a source application is linked, this source application (included in the source component release) can be converted to a destination application (included in the destination component release).

When source applications are converted to destination applications, the destination applications determine how the source applications are replaced.

There are four different types of replacement:

- **Complete replacement**
The functionality of the destination application includes the functionality of the source application. Complete replacement can only be applied in case of multiple replacement or single replacement.
In this session you can indicate that the functionality of a source application and destination application is the same by selecting the **Complete Replacement** check box.
- **Multiple replacement**
There are multiple destination applications for each source application. The source application is replaced by the destination application with the lowest sequence number.
- **Single replacement**
Exactly one destination application for each source application exists. The source application is replaced by the destination application. However, differences in functionality between the source and destination application can exist. This is the case unless the **Complete Replacement** check box is selected, which indicates that the applications contain the same functionality.
- **No replacement**
No destination application is defined for the source application. As a result no conversion takes place.

For a conversion you need to define the following information:

- **Component** (Source/Destination)
The source and destination component
- **Component Release** (Source/Destination)
The source and destination component release
- **Description** (Source/Destination)
The source and destination component release description
- **Source**
 - **Application**
The application that needs to be converted
- **Destination**
 - **Sequence Number**
In case of more than one possible destination application give the one that best fits the lowest sequence number

- **Application**
The application that replaces the source application
- **Authorization**
Here you can specify a new authorization that differs from the old authorization connected to the source application. For example, in BaanERP 5.0b the authorization of an application is only **Display**, the new authorization in BaanERP 5.0c must be **Full**.
- **Complete Replacement**
Select this check box if the destination application replaces the functionality of the source application, or covers the functionality of the source application.
- **Approved Conversion**
Select this check box to approve the conversion. During the actual conversion you can choose whether or not to convert based on the **Approved Conversion** check box.
- **Text**
For each conversion a text can be entered by clicking the **Text** button. The **Text** check box is selected if a text is present. If the conversion is not a complete replacement, the text can be used to describe the differences between the source and destination application.

Source		Destination	
Component	BaanERP	Component	BaanERP
Component Release	5.0b (SP05)	Component Release	5.0c
Description	BaanERP 5.0b service	Description	BaanERP 5.0c

Source	
Source Application	Idcms0251m000
Print and/or Delete Commissions/Rebates	

Destination	
Sequence Number	1
Destination Application	Idcms0260m000
Print and/or Delete Commission/Rebate History	
Authorization	[Dropdown]

Complete Replacement
 Text

Approved conversion

Creating user	jvddobbe	Date of creation	25-10-1999
Modifying user		Date of modification	
Approving user	jvddobbe	Date of approval	25-10-1999

Figure 3-2, Conversion of application details

Component model

3-4

4 Component conversion

In business processes and support applications, links are created to applications of component releases. With a conversion you can:

- Convert the applications defined in the source component release to a related application in the destination component release
- Rename the component and component release of the applications

To convert the applications where used

For a conversion you need to specify the source component release. This is the component release that is used in the DEM models that you want to convert to a destination component release. For both the source and destination component release you can specify a range of applications that you want to convert or to which you want to convert. The conversion takes place only if a conversion is defined for the source application and the destination application.

- This check box is always selected if the **Execute Test Run** option is active. A report is created with all conversions that took place.

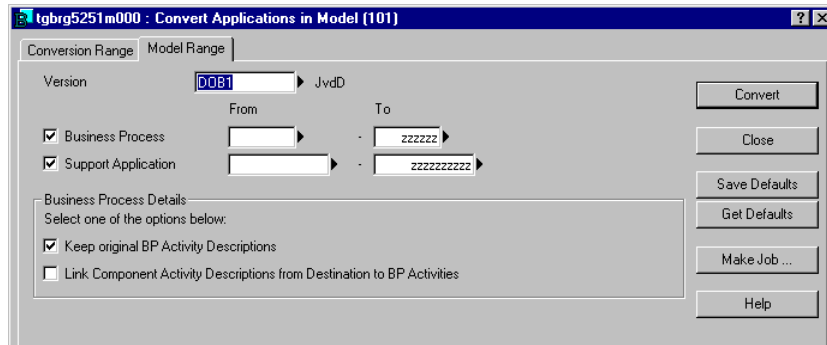


Figure 4-2, Conversion of applications 2

- 5 Enter the version where the business processes and support applications are located that you want to convert.
- 6 Specify a range of business processes or support applications that you want to convert. You need to select the related check box to activate the selection range.
- 7 You can choose to keep the descriptions as they are. Or you can link the activity description to the application description of the destination component.

To rename the component and component release where used

Use the Rename Cmp. & Release Codes in Activities and Supp. App's (tgbrg5220m000) session to rename the component and component release of an application, where it is used in business processes and support applications, to another component release. If the application does not exist in the destination component release, a report is created of where this occurred.

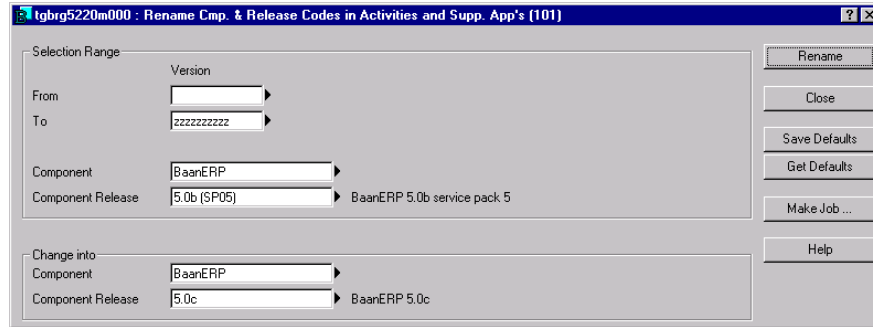


Figure 4-3, Conversion of component release where used

Select the **Exclude Rich Text Format (rtf)** check box if you do not want the texts, that are in rtf format, to be exported.

In the **Export** group box, specify the selection range of components and their releases.

In the **Component Export Details** group box, specify the range of applications to export from the selection range of components and their releases.

Select the **Sub Applications** check box to export the sub applications linked to the applications (only for the selection range of applications).

You can choose to export the application conversion information defined between component releases. To do this select the **Application Conversion Data** check box and go to the **Application Conversion Data** tab.

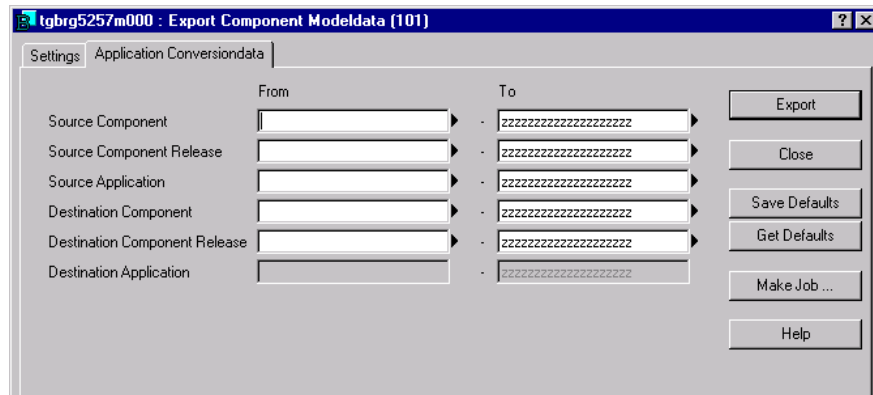


Figure 5-2, Export component model information 2

On the **Application Conversion Data** tab you can make a selection on the conversion data.

To import component data

Use the Import Component Model Data (tgbg5256m000) session to import an export file created with the Export Component Model Data (tgbg5257m000) session.

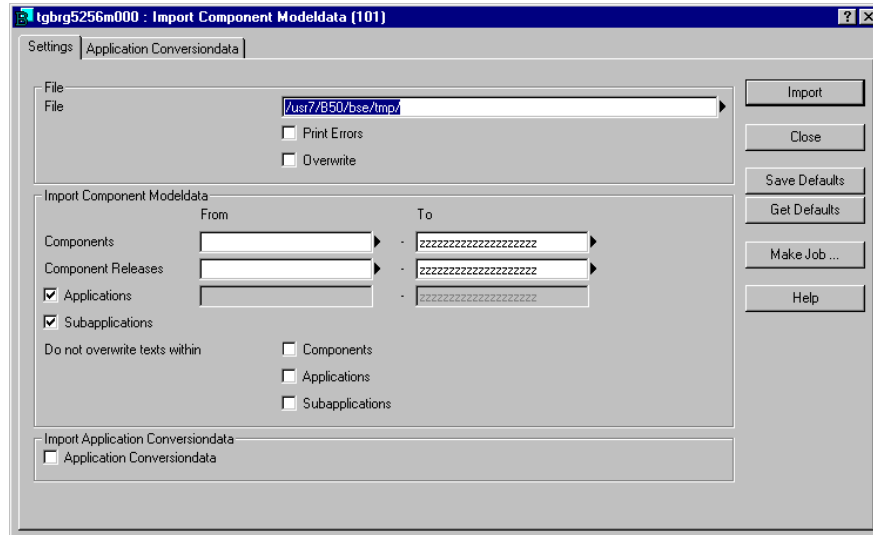


Figure 5-3, Import component model information 1

You can select a range of component releases to import of or overwrite existing component information. You can separately select whether or not to overwrite existing texts by selecting the **Do not Overwrite Texts Within** check boxes.

If you also want to import application conversion information, select the **Application Conversion Data** check box.

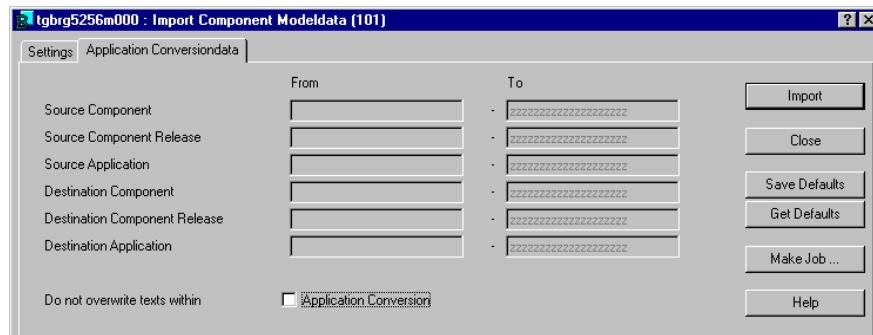


Figure 5-4, Import component model information 2

On the **Application Conversion Data** tab you can choose not to overwrite existing application conversion texts by selecting the **Application Conversion** check box.

To import a component model in XML format

With the following session you can import application information into a component release that you have defined.

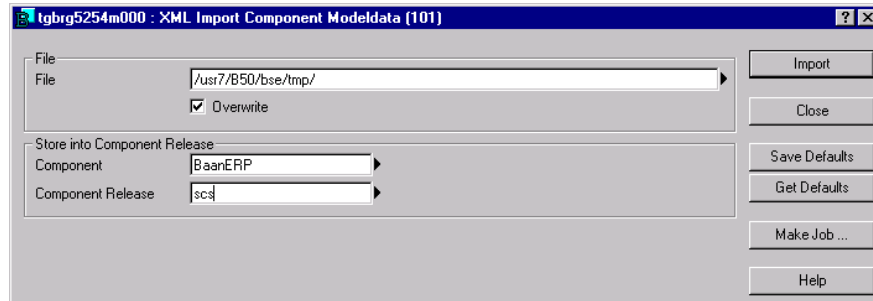


Figure 5-5, XML Import Component Model data

To be able to import the application information into the component release the application information must be structured in the following XML import scheme:

Example BaanERP

```

<applications>
  <application>
    <atr name="id">tdsls1500m000</atr>
    <atr name="main">true</atr>
    <atr name="text">This application is used for viewing Quotations</atr>
    <atr name="1">Prijsopgaven</atr>
    <atr name="2">Quoations</atr>
    <subapplications>
      <subapplication>
        <atr name="id">tdsls1100m000</atr>
      </subapplication>
      <subapplication>
        <atr name="id">tdsls1501m000</atr>
      </subapplication>
      <subapplication>
        <atr name="id">tdsls1550m000</atr>
      </subapplication>
      <subapplication>
        <atr name="id">tdsls1205s000</atr>
      </subapplication>
      <subapplication>
        <atr name="id">tdsls1210m000</atr>
      </subapplication>
    </subapplications>
  </application>
  <application>
    <atr name="id">tdsls1100m000</atr>
    <atr name="main">false</atr>
    <atr name="text">This application is used for updating Quotations</atr>
    <atr name="1">Prijsopgaven</atr>
    <atr name="2">Quotations</atr>
  </application>
  ...
</applications>

```

Example SCS planner

```

<applications>
  <application>
    <atr name="application.id">Demand Plan</atr>
    <atr name="description">View and edit demand orders and dependent demand.</atr>
  </application>
  <application>

```

```
<attr name="application.id">Order Promising</attr>
<attr name="description">Lets you view the quantity available to promise (ATP) over
time for any item or confirm that there is enough to cover a demand.</attr>
</application>
<application>
<attr name="application.id">Items</attr>
<attr name="description">Displays all item-related information including routing,
supply method, and inventory.</attr>
</application>
<application>
<attr name="application.id">Item Groups</attr>
<attr name="description">Allows the definition of grouping by Items.</attr>
</application>
<application>
<attr name="application.id">Site/Warehouse</attr>
<attr name="description">View and control site properties.</attr>
</application>
<application>
<attr name="application.id">Distribution Network</attr>
<attr name="description">Distribution Network</attr>
</application>
</applications>
```

Example Ms Outlook

```
<applications>
<application>
<atr name="id">New Message</atr>
<atr name="main">>true</atr>
<atr name="text">Application used for sending e-mail around the world</atr>
<atr name="2">New Message</atr>
<atr name="argument">/c IPM.note</atr>
</application>
</applications>
```

The sub applications can be nested as shown in the example. For an application to have sub applications, it must be a main application.

The application description can be given in multiple languages. The language codes must be consistent with the Baan standards.

See *Appendix – Language codes*.

7 Glossary

Approved conversion	Specifies whether a conversion is approved.
Complete replacement	Indicates (when set to Yes) whether the destination application includes the functionality of the source application.
Component	A set of applications and their related data.
Component release	The identification of the issue of a component.
Destination component, component release, and application	The combination of component, component release, and application to which to convert to.
Multiple replacement	Specifies that for each source application code, multiple destination session codes exists (1:N).
No replacement	Specifies that no destination application codes are defined for the source application code (1:0).
Sequence number	A number that is introduced to make distinctions among the several destination applications that can exist for one source application. The destination application with the lowest sequence number is the best candidate for replacing the source application.
Source component, component release, and application	The combination of component, component release, and application from which to convert.

Component model

7-2

8 Appendix - Language codes

The languages are divided into three categories, A, B, and C.

A Code	Language	B Code	Language	C Code	Language
2	US English	4	French	a	Arabic
1	Dutch	5	Spanish	b	Bulgarian
3	German	6	Italian	c	Catalan
j	Japanese	7	Danish	d	Polish
		8	Norwegian	e	Elladi (Greek)
		9	Swedish	E	Estonian
		P	Portuguese (European)	h	Hebrew
		p	Portuguese (Brazilian)	H	Hrvatski (Croat)
		f	Finnish	L	Latvian
		n	Chinese Traditional	m	Magyar (Hungarian)
		o	Chinese Simplified	q	Canadian French
		k	Korean	r	Romanian
				R	Russian
				S	Slovenian
				t	Czech
				T	Turkish
				v	Lithuanian

Component model
8-2