Designing a Block
SAP AG
Neurottstraße 16
69190 Walldorf
Germany
T +49/18 05/34 34 24
F +49/18 05/34 34 20
www.sap.com

© Copyright 2005 SAP AG. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Microsoft, Windows, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.

IBM, DB2, DB2 Universal Database, OS/2, Parallel Sysplex, MVS/ESA, AIX, S/390, AS/400, OS/390, OS/400, iSeries, pSeries, xSeries, zSeries, z/OS, AFP, Intelligent Miner, WebSphere, Netfinity, Tivoli, and Informix are trademarks or registered trademarks of IBM Corporation in the United States and/or other countries.

Oracle is a registered trademark of Oracle Corporation.

UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group.

Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems, Inc.

HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

Java is a registered trademark of Sun Microsystems, Inc.

JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

MaxDB is a trademark of MySQL AB, Sweden.

SAP, R/3, mySAP, mySAP.com, xApps, xApp, SAP NetWeaver, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.

These materials are subject to change without notice. These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

Disclaimer
Some components of this product are based on Java™. Any code change in these components may cause unpredictable and severe malfunctions and is therefore expressively prohibited, as is any decompilation of these components.

Any Java™ Source Code delivered with this product is only to be used by SAP’s Support Services and may not be modified or altered in any way.
Typographic Conventions

<table>
<thead>
<tr>
<th>Type Style</th>
<th>Represents</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>Example Text</em></td>
<td>Words or characters quoted from the screen. These include field names, screen titles, pushbuttons labels, menu names, menu paths, and menu options. Cross-references to other documentation.</td>
</tr>
<tr>
<td><strong>EXAMPLE TEXT</strong></td>
<td>Technical names of system objects. These include report names, program names, transaction codes, table names, and key concepts of a programming language when they are surrounded by body text, for example, SELECT and INCLUDE.</td>
</tr>
<tr>
<td><em>Example text</em></td>
<td>Emphasized words or phrases in body text, graphic titles, and table titles.</td>
</tr>
<tr>
<td><strong>Example text</strong></td>
<td>Output on the screen. This includes file and directory names and their paths, messages, names of variables and parameters, source text, and names of installation, upgrade and database tools.</td>
</tr>
<tr>
<td><em>Example text</em></td>
<td>Exact user entry. These are words or characters that you enter in the system exactly as they appear in the documentation.</td>
</tr>
<tr>
<td><code>&lt;Example text&gt;</code></td>
<td>Variable user entry. Angle brackets indicate that you replace these words and characters with appropriate entries to make entries in the system.</td>
</tr>
<tr>
<td><strong>EXAMPLE TEXT</strong></td>
<td>Keys on the keyboard, for example, F2 or ENTER.</td>
</tr>
</tbody>
</table>

Icons

<table>
<thead>
<tr>
<th>Icon</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>!</td>
<td>Caution</td>
</tr>
<tr>
<td>🔄</td>
<td>Example</td>
</tr>
<tr>
<td>💡</td>
<td>Note</td>
</tr>
<tr>
<td>🕵️‍♂️</td>
<td>Recommendation</td>
</tr>
<tr>
<td>🛠️</td>
<td>Syntax</td>
</tr>
</tbody>
</table>
Scenario
This tutorial describes how to create different block types in the CAF process layer (Guided Procedures).

Introduction
Blocks are the structural units that build a process in Guided Procedures. They are re-usable and may group different items, such as actions and nested blocks. The block type defines how the items in the block are executed. The following block types exist:

- **Sequential block** – the items in the block are executed sequentially in the order that you defined.
- **Precondition loop block** – the items in the block are executed in a loop and the loop criterion is checked before execution of the first item.
- **Postcondition loop block** – the items in the block are executed in a loop, but the loop criterion is checked after the last item has been executed. This guarantees that the items in the block are executed at least once.
- **Alternative block** – the user chooses one of the available alternatives at runtime, and his or her decision defines which items are executed.
- **Parallel block** – the items in the block flow are executed simultaneously.
- **Parallel dynamic block** – the items in the block flow are executed simultaneously and multiple processors can be assigned for items execution.

Prerequisites:

<table>
<thead>
<tr>
<th>Documents</th>
<th>Before you start with this tutorial, see:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overview Guided Procedures Design Time (PPT)</td>
<td></td>
</tr>
<tr>
<td>Design a Process from Scratch</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Authorizations</th>
<th>You must hold the following Guided Procedures roles:</th>
</tr>
</thead>
<tbody>
<tr>
<td>GP Business Expert</td>
<td></td>
</tr>
<tr>
<td>GP Expert User</td>
<td></td>
</tr>
<tr>
<td>GP User</td>
<td></td>
</tr>
</tbody>
</table>

Applicable Releases
This tutorial is compatible with the following release” Beginning with SAP NetWeaver 2004s SPS06”.

Disclaimer
Any software coding and/or code lines / strings (“Code”) included in this documentation are only examples and are not intended to be used in a productive system environment. The Code is only intended better explain and visualize the syntax and phrasing rules of certain coding. SAP does not warrant the correctness and completeness of the Code given herein, and SAP shall not be liable for errors or damages caused by the usage of the Code, except if such damages were caused by SAP intentionally or grossly negligent.
The Step-By-Step Solution

The goal is to create a process with a set of blocks, actions and callable objects that illustrate the usage of blocks. You will see how to create different types of blocks depending on your particular needs.

Create a New Folder

1. Launch the portal using the following URL:
   http://<host>:<port>/irj/portal
   Replace <host> with the host name of your application server and <port> with the server’s port for HTTP communications.


3. Choose “Create Folder”

4. Enter the name of the folder and a meaningful description. Choose Create.
Create a Conditional Loop Block

The only difference between Precondition Loop Block and Post-condition Loop Block is that the post-condition loop block is executed at least once, and then checked for the loop condition. The Precondition Loop Block may not be executed at all, since the loop condition is checked before execution.

Since there is only a minor difference between the two blocks, we will only discuss the creation of a Postcondition Loop Block.

The below example presents a variation of the time-off process, in which the initiator of the process can file more than one time-off request.

You need to create:

- A process to hold all blocks, actions and callable objects.
- A post-conditional block within the process that holds all actions and callable objects of the time-off process, as well as the loop criterion.

You can use the Time-Off Process block, which is already designed.

- A decision action and callable object of type Decision Dialog that serves as a loop criterion.

There are two basic ways in which you can start designing the process: top-down and bottom-up. Here we use the bottom-up approach, that is, we first create callable objects and actions that are later integrated into blocks and processes.

Create a Callable Object of Type Decision Dialog

1. In the Gallery, choose Create Callable Object.

Enter the following data for the callable object:

- **Type:** Decision Dialog
- **Name:** Decision Dialog (yes/no)
- **Description:** Decision dialog (yes/no)
- **Original Language:** English
- **Location:** Design a block (use Choose … to select a folder).

2. Choose Next.
3. Mark the option *Decision for Loops* and choose *Next*.

4. Skip the *Define Input* screen by choosing *Next*.

5. In the *Set Configuration* step enter the following data:
   - **Heading:** Do you want to create another time off request?
   - **Message:** Click on "yes" if you like to create one more time off request. Click on "no" to leave.
   - **Continue Button Label:** Yes
   - **Break Button Label:** No

   At runtime this callable object displays a screen where you can answer with *Yes* or *No*.

6. Choose *Next* and then *Finish and Open*. 
Test the Callable Object

7. In the object design time, click the Test tab page and choose Execute.

8. Check the results.

9. Choose (Activate) in the contextual panel and confirm that you want to activate the callable object.

You have successfully created the callable object. Choose Gallery.

Create an Action for the Callable Object of Type Decision Dialog

10. From the contextual panel, choose Create Action and confirm your decision.

The action design time opens.
11. Enter the following basic data for the action:
   • Name: Decision (yes/no)
   • Description: Decision (yes/no)
   • Original Language: English
   • Location: Design a Block

12. From the Item dropdown box select Callable Object for Execution and choose (Insert).

13. Navigate in the Gallery to the callable object Decision Dialog (yes/no) and choose Select.
   The callable object appears in the list.
   In the contextual panel, choose (Activate) and navigate back by choosing Gallery.

Create a Postcondition Block

14. In the Gallery choose Create Block from the contextual panel.

15. In the dropdown box for the block type, select Postcondition Loop Block and choose Create.
   The block design time opens.

16. Enter the following basic data:
   • Name: Postcondition Block
   • Description: Postcondition block
   • Original Language: English
   • Location: Design a Block

17. Choose (Insert).

18. Make sure that Loop Decision Action is selected in the Item dropdown box. Navigate in the Gallery to the
19. In the Item dropdown box select Loop Body Block and choose (Insert).


21. Check if the block has a decision action and body block, choose Roles.

22. For convenience when testing, consolidate all roles to one – for example, Employee. To do that, select all roles, and enter Employee in the Consolidate To field. Choose Go.

23. Choose Save.

   You do not need to consolidate the parameters. Consolidation is already applied within the Time-Off Process block.

24. Choose (Activate) in the contextual panel and navigate back by choosing Gallery.

Create a Process Template

25. From the contextual panel, choose Create Process.

   The process design time opens.

26. Enter the following basic data for the process template:
   • Name: Postcondition Block
   • Description: Postcondition block
27. The process flow for the Time-off Process consists of one block.

28. You have already created this block. To insert the block into the process flow, choose (Insert).

29. Navigate in the Gallery and select the Postcondition Block loop block. Choose Select.

30. On the Roles tab page, change the role type to Initiator.

Note: In the running instance of this template the Employee role is automatically filled by the role starting the process.

31. On the Built-in Roles tab page, change all roles to Initiator.

Note: In the running instance of this template, the built-in process roles (Administrator, Overseer and Owner) are automatically filled by the role starting the process.

32. Choose (Activate).

The process template has been successfully created. Now you can test it.

Execute the Process
33. In the portal, navigate to Guided Procedures → Runtime → Start a New Process. Select the appropriate folder (in this example: Design a Block). Select the process containing the conditional block (in this example: Postcondition Block). Choose Next.

34. Choose Next and Initiate.

35. Complete the steps of the Time-Off Process to reach the postconditional step Decision (yes/no).

36. To repeat the process, choose Yes. To end the process, choose No.

37. The steps that you completed are displayed in the Activities contextual panel.

Create an Alternative Block Type

In the following example an employee files a time-off request, but first decides on the number of people that are required to approve the request (the decision depends, for example, on the specific company position of the employee). This functionality is enclosed in a block of type alternatives.

For the purpose, you need to create:

- A process
- A main sequential block within the process to hold all other blocks, actions and callable objects
- An alternatives block within the main block that holds a decision action and a callable object of type Decision Dialog that is used for Complex Decision with Result States
  This callable object has the following alternatives:
  - Approval by one person
  - Approval by two persons
- Two separate sequential blocks that hold the actions for each of the alternatives listed above. For this purpose you use the actions and objects already designed in the time-off process

Here we use the top-down approach, that is, we first create processes and blocks that are later filled with actions and callable objects.

**Create a Process and a Main Sequential Block**

1. Create the Design a Block Alternatives folder to store the objects that you create in this example.

2. From the contextual panel in the gallery, choose Create Process. Select a language for the process and confirm your decision.
   The process design time opens.

3. Enter the following basic data for the process template:
   - Name: Alternatives Block Process
   - Description: Alternatives block process
   - Original Language: English
   - Folder: Design a Block Alternatives
4. To create a main sequential block, select **Block** in the **Item** dropdown box and choose ☐ (Create New).

5. In the block type dropdown box, select **Sequential Block** and choose **Select**.
   The block design time opens.

6. Enter the following basic data:
   - Name: **Main process block**
   - Description: **Main process block**
   - Original Language: **English**
   - Location: **Design a Block Alternatives**

**Create a Block of Type Alternatives**

7. The first step in the process is the alternatives decision block. Create a block by choosing ☐ (Create New).
8. In the block type dropdown box, select *Alternatives Block*.

9. Choose *Select*.

The block design time opens.

10. Enter the following basic data:
    - Name: *Choose Approval Type*
    - Description: *Choose Approval Type*
    - Original Language: *English*
    - Location: *Design a Block Alternatives*

**Create a Decision Action and Callable Object of Type Decision Dialog**

11. Select *Decision Action* in the *Item* dropdown box and choose ![Create New](Create New).
12. Enter the following basic data:
   - Name: Approval decision action
   - Description: Approval decision action
   - Original Language: English
   - Location: Design a Block Alternatives

13. Select Callable Object for Execution in the Item dropdown box and choose Create New.

14. Create a callable object of type Decision Dialog.

15. Enter the following basic data:
   - Name: Decision Dialog with 1/2 Approvals
   - Description: Decision Dialog with 1/2 Approvals
   - Original Language: English
   - Location: Design a Block Alternatives

      Choose Next.

16. Select Complex Decision with Result States and choose Next.

17. Skip the Define Input screen by choosing Next.
18. In step *Set Configuration*, define a Decision group.
   1. Choose *Edit* to change the name of the group: Choose Approval Type.
   2. Choose *Save*.

19. Define options for the decision group:
   1. Choose *Edit* to edit the *New Decision Option*.
      
      Option ID: *oneapprover*
      
      Name: *Approval by one person*  
      
      2. Choose *Save*.
      
      3. Create one more option by choosing *Add Option*.
      
      4. Use the following data:
      
      Option ID: *twoapprover*
      
      Name: *Approval by two persons*  
      
      5. Choose *Create*. 
20. Choose Next and then Finish.

Open the Test tab page of the callable object to check if the object works correctly. Execute the object with each of the options selected, and check the result state reached.

21. Activate the object by choosing (Activate).

Create a Sequential Block for Alternative One Approval

22. In the Choose Approval Type block design time, select Block from the Item dropdown box and choose (Create New).

Set the block type to Sequential.
23. Enter the following basic data:
   - Name: One approval block
   - Description: One approval block
   - Original Language: English
   - Location: Design a Block Alternatives

24. Select Action from the Item dropdown box and choose (Insert). Browse to select the actions Create Request, 1stApproval and Book Request from the time-off process.

25. Define target actions for the result states of 1st Approval:
   - For Approve: Book Request
   - For Return to Initiator: Create Request
   - For Reject: Terminal

Create a Sequential Block for Alternative Two Approvals

26. In the Choose Approval Type block design time, select Block from the Item dropdown box and choose (Create New).
    Set the block type to Sequential.

27. Enter the following basic data:
   - Name: Two approvals block
   - Description: Two approvals block
   - Original Language: English
   - Location: Design a Block Alternatives

28. Select Action from the Item dropdown box and choose (Insert). Browse to select the actions Create Request, 1st Approval, 2nd
Approval and Book Request from time-off process.

29. Define target actions for the result states of 1st Approval:
   • For Approve: 2nd Approval
   • For Return to Initiator: Create Request
   • For Reject: Terminal

30. Define target actions for the result states of 2nd Approval:
   • For Approve: Book Request
   • For Return to Initiator: Create Request
   • For Reject: Terminal

Define Targets for the Result States of Decision Dialog with 1/2 Approvals

31. In the Decision Dialog with 1/2 Approvals object design time, expand the Result State node. Define the target for each of the states:

<table>
<thead>
<tr>
<th>Custom ID: oneapprover</th>
<th>One approval block</th>
</tr>
</thead>
<tbody>
<tr>
<td>Custom ID: twoapprover</td>
<td>Two approvals block</td>
</tr>
</tbody>
</table>

32. You should now have the complete structure of the process. Choose (Save All).

Consolidate Roles and Parameters
33. Open the Roles tab page and consolidate the roles. For the purpose of testing, consolidate all roles to one, for example Processor.

34. Select Initiator for both the Processor role and the built-in roles.

35. Consolidate the parameters in each block and also in the process as a whole.

For more information on consolidating parameters see the “Appendix”.

36. Activate the process by choosing (Activate).

37. In the portal, navigate to Guided Procedures → Run Time → Start a New Process → Design a Block Alternatives and choose Next.
38. Initiate and test the process for each of the possible alternatives.

**Create a Parallel Block**

For this example, you need to create a parallel block of two comment actions that have to be executed in parallel before the request is approved. This enables the approver to make a decision based on the opinion of two other persons.

You can re-use part of the time-off process. You have to construct the process in the following way:

- Parallel block process
  - Main process block
    - Create request
    - Parallel comment block
      - 1<sup>st</sup> comment
      - 2<sup>nd</sup> comment
    - View comments
    - Approval
    - Book request

To construct the process template, we use the top-down design method.

**Create a Process and a Main Sequential Block**

1. Create a new folder, for example *Parallel Block Process*.

2. From the contextual panel, choose *Create Process* and enter the required data:
   - Name: *Parallel Block Process*
   - Description: *Parallel Block Process*
   - Original Language: *English*
   - Location: *Parallel Block Process*
3. To create a main sequential process block, choose \( \text{(Create New)} \). It contains all other blocks and actions in the process. Enter the following data:
   - Name: \text{Main Process Block}
   - Description: \text{Main Process Block}
   - Original Language: \text{English}
   - Location: \text{Parallel Block Process}

4. Save your interim result.

**Insert Create Request from Time-Off Process**

5. Mark the created “Main Process Block” of step 3.

6. Select \text{Action} from the \text{Item} dropdown list and choose \( \text{(Insert)} \). Browse to select action \text{Create Request} from the \text{Time-Off Process} folder.

7. Save the interim result.

**Create a Parallel Comment Block**

8. In the \text{Main process block} design time, select \text{Block} from the \text{Item} dropdown list and choose \( \text{(Create New)} \) to create a block that holds the parallel comment actions. Set the block type to \text{Parallel Block} and choose \text{Select}.

9. Select the block and enter the following data:
   - Name: \text{Parallel Comment Block}
   - Description: \text{Parallel Comment Block}
10. Save the interim result.

Create Actions 1st Comment and 2nd Comment

11. Mark the “Parallel Comment Block” (steps 8 to 9), select Action in the Item dropdown box, and choose (Create New). For the basic data enter:
   - Name: 1st Comment
   - Description: 1st Comment
   - Original Language: English
   - Location: Parallel Block Process

12. Mark the action “1st Comment” and select Callable Object for Execution from the Item dropdown box. Choose (Create New).

13. In the Type list, expand the Data Forms category and choose Input. Enter the required data:
   - Name: Comment form 1
   - Description: Comment form 1
   - Original Language: English
   - Location: Parallel Block Process
   Choose “Next” to continue.

14. Define the input parameters. To create parameters, refer to the table:

<table>
<thead>
<tr>
<th>Context Parameter</th>
<th>Technical Name</th>
<th>Object Type</th>
<th>A Child of</th>
</tr>
</thead>
<tbody>
<tr>
<td>Employee Data</td>
<td>employeeData</td>
<td>Structure</td>
<td></td>
</tr>
<tr>
<td>First Name</td>
<td>firstname</td>
<td>String</td>
<td>Employee Data</td>
</tr>
<tr>
<td>Last Name</td>
<td>lastname</td>
<td>String</td>
<td>Employee Data</td>
</tr>
<tr>
<td>Time-off data</td>
<td>timeOffData</td>
<td>Structure</td>
<td></td>
</tr>
</tbody>
</table>
Choose **Insert New...** to enter the input parameters mentioned above.

To enter a new input parameter that is supposed to be “a child of” of an input parameter of type “structure”, mark the last parameter mentioned (e.g. “Employee Data”) and choose **Insert Child...**.

After entering the required input parameters, choose “Next”.

15. On the **Define Output** screen, define the output parameters listed in the table:

<table>
<thead>
<tr>
<th>Context Parameter</th>
<th>Technical Name</th>
<th>Object Type</th>
<th>A Child of</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comment 1</td>
<td>comment1</td>
<td>Structure</td>
<td>Comment 1</td>
</tr>
<tr>
<td>First Name</td>
<td>firstname</td>
<td>String</td>
<td>Comment 1</td>
</tr>
</tbody>
</table>
Choose Insert New.... to enter the output parameters mentioned above.

To enter a new input parameter that is supposed to be "a child of" of an input parameter of type "structure", select the last parameter mentioned (e.g. "Comment 1") and choose Insert Child....

💡 Set the Value Required indicator for all parameters.

16. Choose Next.
17. Choose Finish.

18. The callable object design time opens. You can test and activate the callable object “Comment form 1” before you create the second action.

19. To test the callable object “Comment form 1”, select it and choose the Test tab.
20. In the process flow design time, select the Parallel Comment Block and create action 2nd Comment by analogy with 1st Comment.

21. In action 2nd Comment, create a callable object for execution named Comment form 2 following the instructions for callable object Comment form 1.

   All input and output parameters are the same except for structure Comment 2.

   You need two different callable objects with the identical structure. If you use the same object twice, the structure parameters cannot be consolidated correctly.

Create Action View Comments

22. In the Main process block, create a new action by choosing (Create New).

23. Enter the required data:
   - Name: View Comments
   - Description: View Comments
   - Original Language: English
   - Location: Parallel Block Process

24. Select Callable Object for Execution from the Item dropdown list and choose (Create New).
25. In the *Type* list, expand the *Data Forms* category and choose *Data input form*. Enter the required data:

- **Name**: View Comments
- **Description**: View Comments
- **Original Language**: English
- **Location**: Parallel Block Process

Choose *Next*.

26. Define the input parameters listed in the table:

<table>
<thead>
<tr>
<th>Context Parameter</th>
<th>Technical Name</th>
<th>Object Type</th>
<th>A Child of</th>
</tr>
</thead>
<tbody>
<tr>
<td>Employee Data</td>
<td>employeeData</td>
<td>Structure</td>
<td></td>
</tr>
<tr>
<td>First Name</td>
<td>firstname</td>
<td>String</td>
<td>Employee Data</td>
</tr>
<tr>
<td>Last Name</td>
<td>lastname</td>
<td>String</td>
<td>Employee Data</td>
</tr>
<tr>
<td>Time-off data</td>
<td>timeOffData</td>
<td>Structure</td>
<td></td>
</tr>
<tr>
<td>Type of absence</td>
<td>absenceType</td>
<td>String</td>
<td>Time-off data</td>
</tr>
<tr>
<td>Paid</td>
<td>paidflag</td>
<td>String</td>
<td>Time-off data</td>
</tr>
<tr>
<td>Start Date</td>
<td>startDate</td>
<td>Date</td>
<td>Time-off data</td>
</tr>
<tr>
<td>End Date</td>
<td>endDate</td>
<td>Date</td>
<td>Time-off data</td>
</tr>
<tr>
<td>Comment 1</td>
<td>comment1</td>
<td>Structure</td>
<td></td>
</tr>
<tr>
<td>First Name</td>
<td>firstname</td>
<td>String</td>
<td>Comment 1</td>
</tr>
<tr>
<td>Last Name</td>
<td>lastname</td>
<td>String</td>
<td>Comment 1</td>
</tr>
<tr>
<td>Text</td>
<td>text</td>
<td>String</td>
<td>Comment 1</td>
</tr>
<tr>
<td>Comment 2</td>
<td>comment2</td>
<td>Structure</td>
<td></td>
</tr>
<tr>
<td>First Name</td>
<td>firstname</td>
<td>String</td>
<td>Comment 2</td>
</tr>
<tr>
<td>Last Name</td>
<td>lastname</td>
<td>String</td>
<td>Comment 2</td>
</tr>
<tr>
<td>Text</td>
<td>text</td>
<td>String</td>
<td>Comment 2</td>
</tr>
</tbody>
</table>
Insert 1st Approval and Book Request from Time-Off Process

27. In the Main process block, insert actions 1st Approval and Book Request from the Time-Off Process template.

Consolidate Roles and Parameters

28. In the Parallel Block Process design time, open the Roles tab page and consolidate the roles. For testing purposes, you can consolidate all roles to one, for example Processor.

29. Open the Built-In Roles tab page and assign Initiator to the built-in roles.
30. Consolidate parameters:

1. Map the parameters from Create Request, Book Request and 1st Approval together. You can simply map the corresponding structure parameters.

![Consolidation table]

For more information on consolidating parameters see the “Appendix”.

2. Consolidate the parameters you created in 1st Comment, 2nd Comment and View comments. If you followed the above instructions, the structures should be identical.

3. Consolidate each separate parameter in the structures you created to the corresponding parameter in the Time-Off Data and Employee Data structures of the Create Request action.

31. Save and activate the process.

**Execute the Process**

32. In GP runtime, choose Start a New Process.

33. In the Gallery, navigate to Parallel Block Process → Parallel Block Process template. Choose Next and Initiate.
34. Choose Create Request and proceed to the parallel block step.

35. Choose the 2nd Comment step. You see that the runtime jumps automatically to the 1st Comment step.

You can execute the 2nd Comment step even though the 1st Comment step is not completed. Furthermore, you can only reach the step View Comments after both comment steps have been executed.

36. Complete View Comments, 1st Approval and then Book Request if the request is approved.

The process is now finished.

Create a Parallel Dynamic Block

1. Create a new folder – for example, Design a Parallel Dynamic Block.
2. Choose *Create Process* from the contextual panel.

The process design time opens.

3. Enter the following basic data for the process template:
   - Name: **PDB Process**
   - Description: **PDB Process**
   - Original Language: **English**
   - Location: *Design a Parallel Dynamic Block*

4. Choose ☐ *(Create New)* to create a block that represents the phase of the process.

5. In the block type dropdown box, select *Sequential Block* and choose Select.

6. Select the created block and enter the following basic data:
   - Name: **PDB Process Block**
   - Description: **PDB Process Block**
   - Original Language: **English**
   - Location: *Design a Parallel Dynamic Block*

**Create a Data Input Form**

7. Select *Action* from the Item dropdown list and choose ☐ *(Create New)*.
8. Select the created action and enter the following basic data:
   - Name: Create Request
   - Description: Create Request
   - Original Language: English
   - Location: Design a Parallel Dynamic Block

9. Select Callable Object for Execution from the Item dropdown list and choose (Create New).

10. Enter the following data for the callable object:
    - Type: Expand node Data Forms, and select Data input form.
    - Name: Create Request
    - Description: Create Request
    - Original Language: English
    - Location: Design a Parallel Dynamic Block (click on Choose … to select a folder).

11. Choose Next to open the Define Output screen.
    a. Choose Insert New to enter an output parameter for the form. Add the following parameter:
       - Context parameter: Multiline Input Structure
       - Technical name: multilineInput
       - Object type: Structure
       Select List to define that the parameter may contain multiple values.
    b. Select the Multiline Input Structure parameter and choose Insert Child to create sub-entries for the structure. Add the following parameters:
       - Time-Off Data – a parameter of type Structure; its technical name is tabData. Choose Insert Child to create sub-entries for the Time-Off Data structure. Add the following parameters:
         o Employee ID – a parameter of type String; its technical name is employeeId
         o Type – a parameter of type String; its technical name is type
         o Start Date – a parameter of type Date; its technical name is startDate
         o End Date – a parameter of type Date; its technical name is endDate
• **Approvers** – a parameter of type *String*; its technical name is *approvers*

  Select *List* to define that the parameter may contain multiple values.

12. Choose *Next* and then *Finish*.

### Create a Parallel Dynamic Block

13. Select *PDB Process Block*, choose *Block* from the Item dropdown list, and choose ![Create New](Create New).

14. In the block type dropdown box, select *Parallel Dynamic Block* and choose *Select*.

15. Select the created block and enter the following basic data:

   - **Name**: *PDB Block*
   - **Description**: *PDB Block*
   - **Original Language**: *English*
   - **Location**: *Design a Parallel Dynamic Block*

     The block structure consists of a parallel dynamic block and a subordinate block that can be of any type – parallel or sequential for example. This subordinate block can be filled with any actions and/or blocks.

16. Choose ![Create New](Create New).
17. In the block type dropdown box, select *Sequential Block* and choose *Select*.

Select the created block and enter the basic data.

18. Select *Action* from the Item dropdown list and choose 📐 (Create New).

19. Select the created action and enter the following basic data:

- Name: *Approval*
- Description: *Approval*
- Original Language: *English*
- Location: *Design a Parallel Dynamic Block*

20. Select *Callable Object for Execution* from the Item dropdown list and choose 📐 (Create New).

21. Enter the following data for the callable object:

- Type: Expand node *Process Control* and select *Visual Approval*.
- Name: *Approval*
- Description: *Approval*
- Original Language: *English*
- Location: *Design a Parallel Dynamic Block* (click on *Choose ...* to select a folder).

22. Choose Next to open the *Define Input* screen.

23. Choose *Insert New* to enter an output parameter for the form. Add the following parameters:

- **Time-Off Data** – a parameter of type *Structure*; its technical name is *tabData*.

Choose *Insert Child* to create sub-entries for the *Time-Off Data* structure. Add the following parameters:

- **Employee ID** – a parameter of type *String*; its technical name is *employeeId*
- **Type** – a parameter of type *String*; its technical name is *type*
- 33 -

- **Start Date** – a parameter of type *Date*; its technical name is `startDate`

- **End Date** – a parameter of type *Date*; its technical name is `endDate`

- **Approvers** – a parameter of type *String*; its technical name is `approvers`
  
  Select *List* to define that the parameter may contain multiple values.

24. Choose *Next* to open the *Define Output* screen.

The output parameters for the callable object are pre-defined. You do not need to define additional ones.

25. Choose *Next* to open the *Set Configuration* screen.

Define the configuration parameters of the callable object as follows:

1. **Approve E-Mail** – Choose a template for the e-mails to be sent if the processor of the action chooses *Approve*.

2. **Recipients for Approval** – Define who will receive the approval e-mail; for example, you can choose to send the e-mail to the initiator of the process, to the processor of the action, to the administrator, owner, or overseer of the process, or to all process contributors.

3. **Reject E-Mail** – Choose a template for the e-mails to be sent if the processor of the action chooses *Reject*.

4. **Recipients for Rejection** – Define who will receive the rejection e-mail.

The system sends e-mails only if you select an e-mail template. If you leave the default option *No MIME Selected*, e-mails are not sent. For more information, see Setting Up Mail Templates.
26. Choose Next and then Finish.

27. Select the Parallel Dynamic Block subordinate block and open the Roles tab page.

28. Select the block role and enable the option Filled from Context Parameter.
   An additional dropdown list is displayed on the right.

29. Choose Approvers from the list.

Consolidate the parameters so that the output from the first action are used to fill in the input fields of the Parallel Dynamic Block and the block’s output parameters are used to fill in the input parameters of the display form.

30. To group the parameters, select the Parallel Dynamic Block and open the Parameters tab page.

31. Consolidate the parameters as shown.

32. Select the PDB Process and open the Roles tab page.
   Change the role type to Initiator.

33. On the Built-in Roles tab page, change all roles to Initiator.

34. Save your process by choosing 🔄.

35. Choose 🌟 (Activate).
   You can start the process by calling the Guided Procedure Runtime.
Appendix

In the examples described in the previous sections, you used some of the blocks, actions and callable objects designed for the Time-Off Process template. To ensure that parameters are correctly passed between these activities, you have to consolidate parameters before you activate and initiate the process. The following structure parameters are defined for all actions in the Time-Off Process:

- Time-Off Data
- Employee Data
- HR Consultant Data
- Approvers Data

Since these are identical in Create Request, 1st Approval, 2nd Approval and Book Request, you can consolidate them by simply choosing the corresponding structures from the parameter list and storing them under a collective name:

Context parameters that are present in one action only do not have to be passed as output or input parameters; therefore they do not need to be consolidated.

Structure parameters Processor of First Approval, Processor of Second Approval and Processor for HR Consultant in action Create Request need not be consolidated.

Follow the specific instructions provided for all other parameters in the examples.