



## **Outlook best practices (mail)**

Project name: Outlook (mail) Best practices - Cloud Studio

Package version: 1.1.2

Version	Date	Description
1	December 21, 2020	Document created
1.1	January 12, 2021	Update automation "Search mails" Add automation "Move mails"
1.2	February 9, 2021	Add automation "Download image from mail"
1.3	March 08, 2021	Change template Add information about SDK dependencies
1.4	May 17, 2021	Add automation "Reply to a mail" Add note about catching errors in automation Change automation "Search mails" to get the mail address of the sender and not just the name
1.5	August 12, 2021	Changing the version of the package

# TABLE OF CONTENTS

INTRODUCTION .....	4
IMPORTANT RECOMMANDATION .....	5
General .....	5
Reuse the sample as a new project .....	5
DESCRIPTION .....	7
Settings .....	7
<i>Environment variables</i> .....	7
<i>Dependent packages</i> .....	7
Captures .....	7
Datatypes .....	7
<i>MailContent</i> .....	7
Automations.....	7
<i>Send Single mail</i> .....	7
<i>Send multiple mails</i> .....	8
<i>Search mails</i> .....	8
<i>Move mails</i> .....	9
<i>Download image from mail</i> .....	9
<i>Reply to a mail</i> .....	10
VERSION .....	11
SAP Build Process Automation .....	11
Target application .....	11
PREREQUISITES.....	12
Global setup .....	12
Specific steps to follow before launching the agent .....	12
EXPECTED OUTPUT .....	13

## INTRODUCTION

This document describes the SAP Build Process Automation sample **Outlook (mail) Best practices - Cloud Studio** and provides the following information:

- Description (functional and technical)
- Version used to generate this sample

It also contains information on prerequisites, such as the steps to follow before launching the agent.

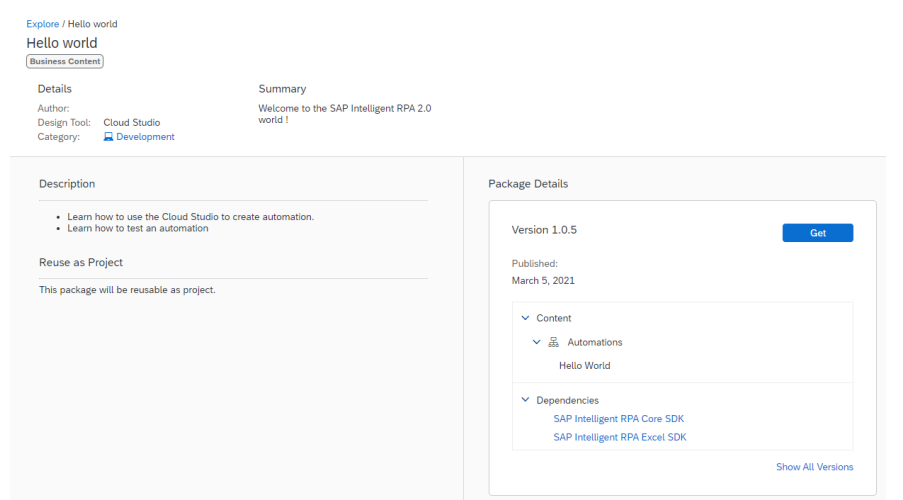
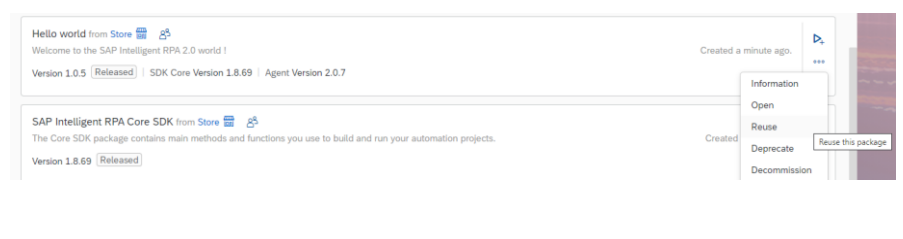
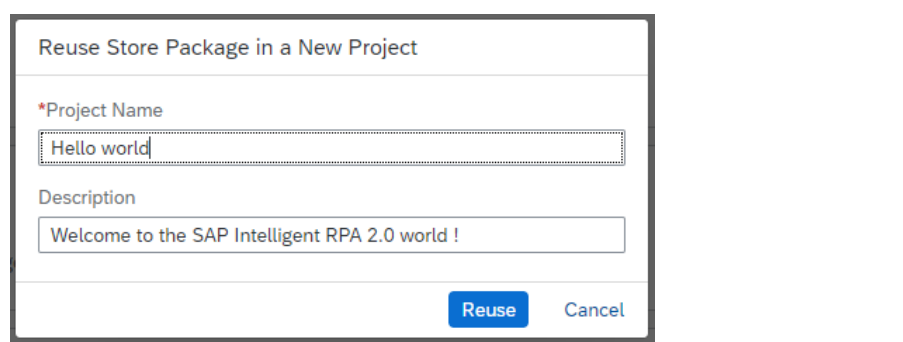
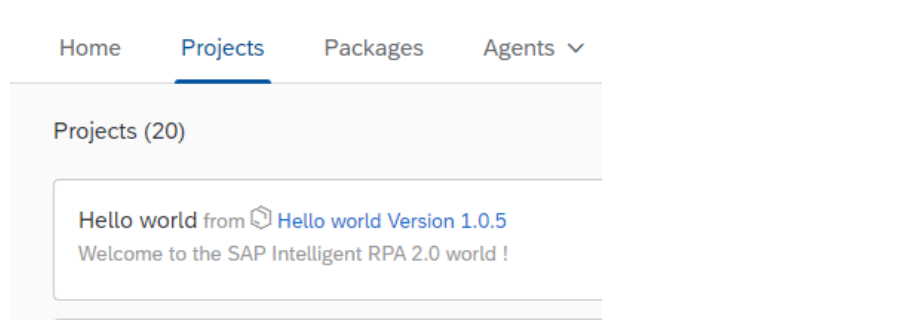
## IMPORTANT RECOMMENDATION

### General

To use this sample, you need to have a basic knowledge and understanding of SAP Build Process Automation tool. At the very least you need to know how to build an automation, add and modify activities and generate a package.

### Reuse the sample as a new project

*Note: screenshot might display a different name than the one of this sample.*

<p>From the Cloud Factory, open the Store tab and select the sample you want to retrieve.</p> <p>Click on the <b>Get</b> button.</p>	 <p>Explore / Hello world Hello world (Business Content)</p> <p>Details Author: Cloud Studio Design Tool: Cloud Studio Category: Development</p> <p>Summary Welcome to the SAP Intelligent RPA 2.0 world !</p> <p>Description</p> <ul style="list-style-type: none"> <li>Learn how to use the Cloud Studio to create automation.</li> <li>Learn how to test an automation</li> </ul> <p>Reuse as Project This package will be reusable as project.</p> <p>Package Details</p> <p>Version 1.0.5 <b>Get</b></p> <p>Published: March 5, 2021</p> <p>Content</p> <ul style="list-style-type: none"> <li>Automations             <ul style="list-style-type: none"> <li>Hello World</li> </ul> </li> </ul> <p>Dependencies</p> <ul style="list-style-type: none"> <li>SAP Intelligent RPA Core SDK</li> <li>SAP Intelligent RPA Excel SDK</li> </ul> <p>Show All Versions</p>
<p>Once the package is retrieved, open the Packages tab of the Cloud Factory.</p> <p>Click on the Options button of the package you just retrieved and select the option <b>Reuse</b>.</p>	 <p>Hello world from Store Welcome to the SAP Intelligent RPA 2.0 world ! Version 1.0.5 (Released)   SDK Core Version 1.8.69   Agent Version 2.0.7 Created a minute ago.</p> <p>SAP Intelligent RPA Core SDK from Store The Core SDK package contains main methods and functions you use to build and run your automation projects. Version 1.8.69 (Released) Created</p> <p>Information Open Reuse Deprecate Decommission</p> <p>Reuse this package</p>
<p>Set a name for the project to be created.</p>	 <p>Reuse Store Package in a New Project</p> <p>*Project Name Hello world</p> <p>Description Welcome to the SAP Intelligent RPA 2.0 world !</p> <p>Reuse Cancel</p>
<p>Open the project that has just been created.</p>	 <p>Home <b>Projects</b> Packages Agents</p> <p>Projects (20)</p> <p>Hello world from Hello world Version 1.0.5 Welcome to the SAP Intelligent RPA 2.0 world !</p>
<p>If needed, update the content of this project, and generate a new package from it.</p>	

You need to execute this procedure to be able to open the project and see all its content (the captured applications, the declared items, the automations, etc.).

## DESCRIPTION

This package contains captures, datatype and automations that are described below. See chapter Version for more details about the version of the Desktop Agent and the SDK dependencies.

### Settings

This section describes the settings of the project such as environment variables or dependent packages that are used in the automation.

#### Environment variables

Name	Description	Type
ContentMailFilePath	Location of the file with the content of the mail to be sent	String
DefaultExcelSheetName	Default name of Excel spreadsheet. Ex: Sheet1	String
DownloadFolderPath	Location of the folder where attachments should be downloaded	String
MailInputParameter	Parameters used to create a mail	Email Item Parameters
SearchCriteria	Criteria used to search mails	Email Search Criteria

#### Dependent packages

N/A

### Captures

This section describes the captures which were made to pilot the application in this sample. It will also describe the different methods which were used to capture the pages and declare the items.

N/A

### Datatypes

This section describes the datatype used in this sample. It describes the structure of the datatype and where it is used in the automations.

#### MailContent

Name of attribute	Type	Description
Recipients	String	Addresses of the recipient of the mail to be sent. If multiple recipients, use semicolon ; to separate them
UserName	String	Name of the recipient(s)
AttachedFile	String	Path of the file to be sent as attachment

### Automations

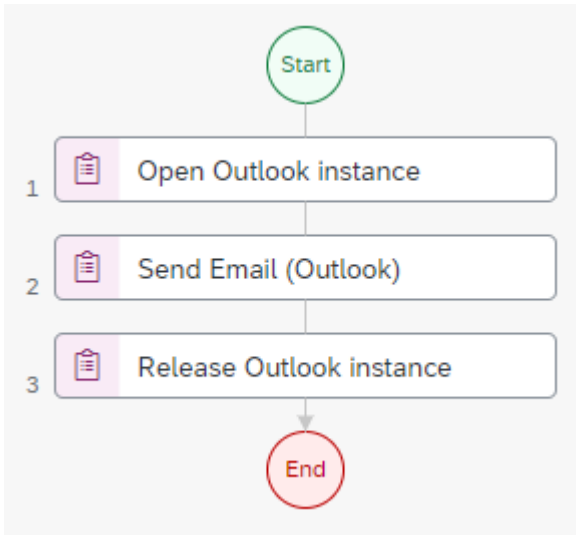
Note: all the automations contain Try/Catch blocks. For readability reasons they were not included in the screenshots of the automations in this document.

#### Send Single mail

**Type:** Attended

**Input:** None

**Output:** None



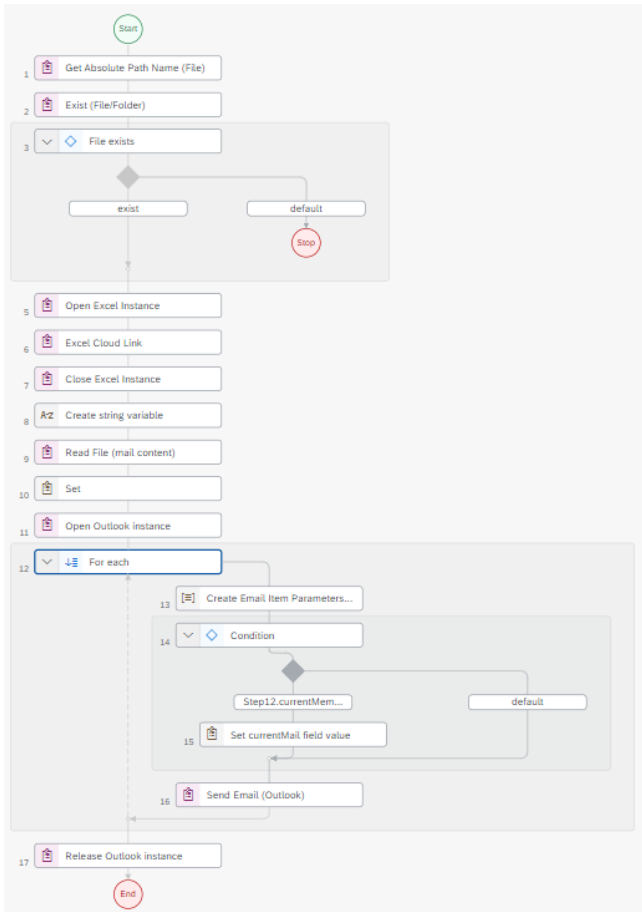
The agent will use the environment variable **MailInputParameter** to create and send a new mail.

### Send multiple mails

**Type:** Attended

**Input:** excelFilePath is the path of the Excel file used to send mails to multiple recipients

**Output:** None



Using Excel file used as input, the agent will create a collection of **MailContent** objects (one for each row in the Excel file).

It will use the ContentMailFilePath variable to read the file containing the body of the mails to be sent.

Then for each MailContent object, it will create an **Email Item Parameter** (where its content comes from the variables detailed above). If there is an attached file to be sent, it automatically adds it.

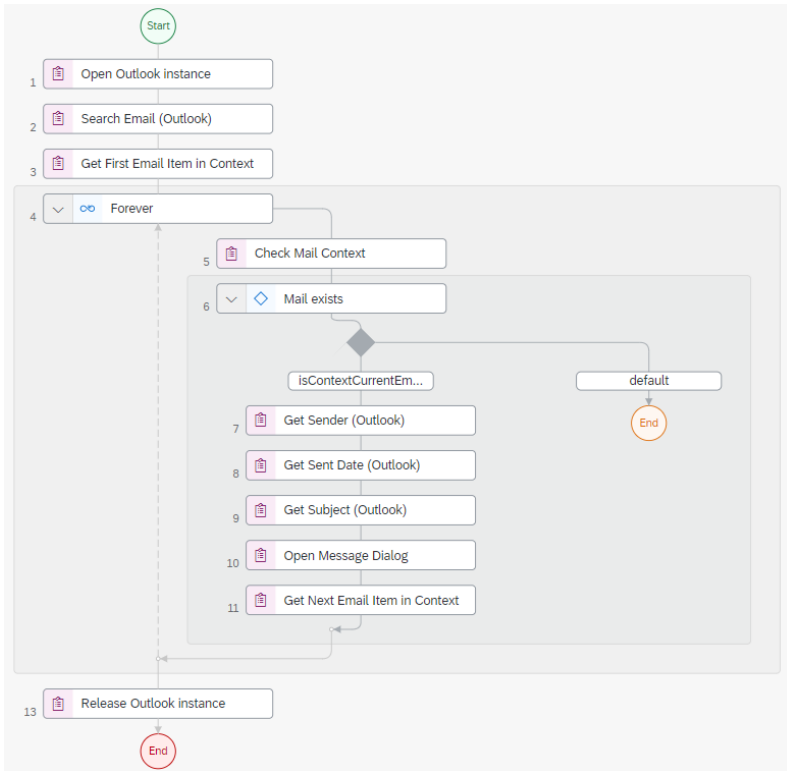
### Search mails

**Type:** Attended

**Input:** None

**Output:** None





Using the environment variable **SearchCriteria**, the agent will look for some given mails in the mailbox.

It will then iterate over the collection to display the following information in a popup:

- Subject
- Sender
- Sent date

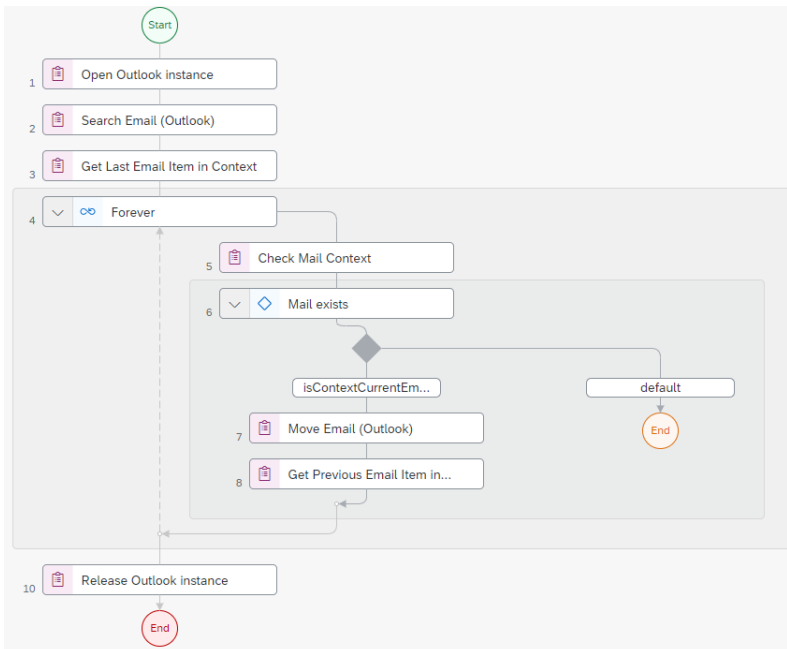
Do not use **Move Email** and **Delete Email** activities in this automation, as it would change the Context.

### Move mails

**Type:** Attended

**Input:** targetFolder is the folder where mails should be moved to. targetStore is the other account that can be used in Outlook

**Output:** None



Using the environment variable **SearchCriteria**, the agent will look for some given mails in the mailbox.

All found mails will be moved to another location, using MoveFolder and MoveStore environment variable.

**Important:** do not leave these variables empty when you deploy your bot.

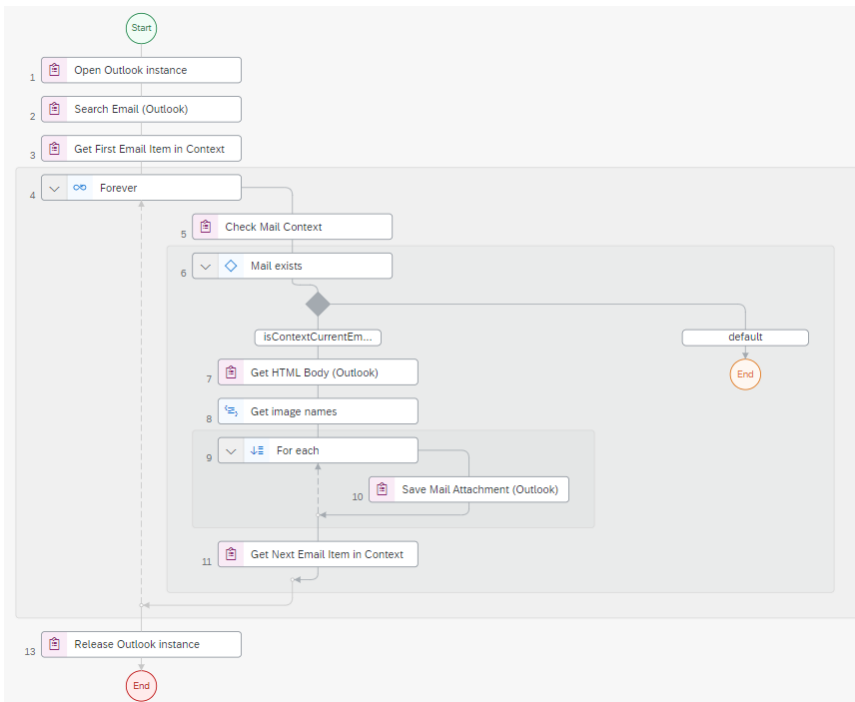
We always work with the last email in context as the email is removed from the context when we perform a **Move Email** action (the pattern to be used is the same when we want to delete multiple emails. [See documentation](#))

### Download image from mail

**Type:** Attended

**Input:** None

**Output:** List of names of images downloaded



Using the environment variable **SearchCriteria**, the agent will look for some given mails in the mailbox.

For each mail found, the agent automatically parses the content of the mail to extract the name of all images that were inserted.

Then it downloads the files using the list of names which was generated in the **Get image names** activity (Custom script)

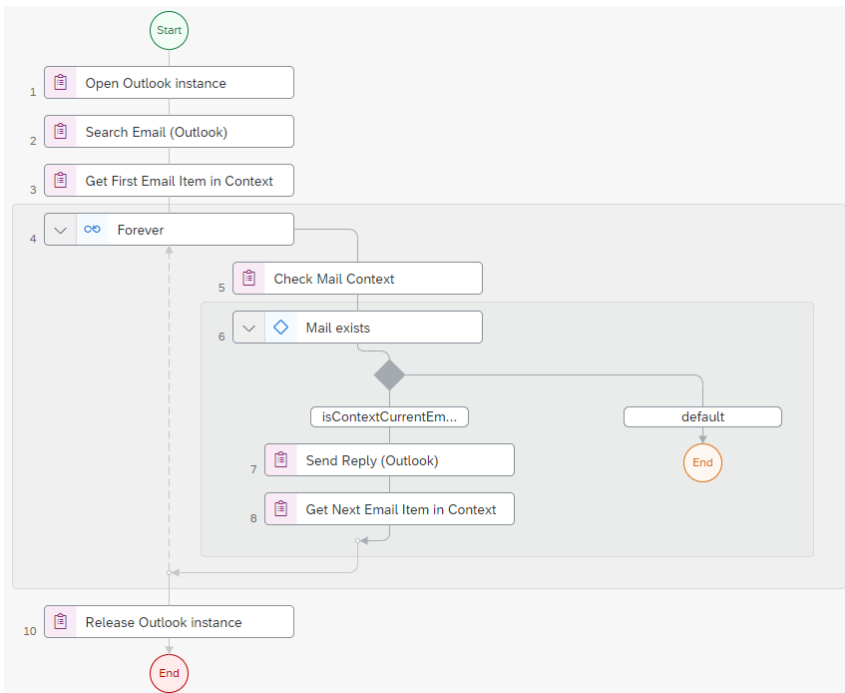
**Note:** If several mails have images that have the same name, you should take care to download them to different locations in order to not overwrite them.

### Reply to a mail

Type: Attended

Input: None

Output: None



Using the environment variable **SearchCriteria**, the agent will look for some given mails in the mailbox.

For each mail found, the agent automatically replies to senders, with a static text in the mail body.

## VERSION

The product versions used to generate this sample are detailed below. This sample is provided “as is”, with no warranty that it will work correctly with other versions. If some versions of your software are different (such as the tool version or the target application version), you may need to recapture the application and/or update the workflow activities.

### SAP Build Process Automation

This sample targets the Desktop Agent **2.0.7** or higher.

The following SDK dependencies were used to generate this sample:

irpa_core	1.14.54
irpa_excel	1.14.54
irpa_outlook	1.14.54
irpa_pdf	N/A
irpa_ui5	N/A
irpa_word	N/A

See [documentation](#) for more details about the compatibility between SDK version and Desktop Agent.

### Target application

N/A

## PREREQUISITES

### Global setup

SAP Build Process Automation must be installed in accordance with the installation guide available [here](#). An SAP Build Process Automation Factory must be available with a suitable environment (containing an agent). All information can be found in the “Getting Started” section accessible via the above link.

### Specific steps to follow before launching the agent

- Download the **samples.zip** archive from the sample and extract its content
- When you deploy your automation, set the environment variable **ContentMailFilePath** with the location of the text file **mailContent.txt**.
- Set a value to **DownloadFolderPath** variable

**EXPECTED OUTPUT**

N/A

[www.sap.com/contactsap](http://www.sap.com/contactsap)

© 2019 SAP SE or an SAP affiliate company. 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 SE or an SAP affiliate company.

The information contained herein may be changed without prior notice. Some software products marketed by SAP SE and its distributors contain proprietary software components of other software vendors. National product specifications may vary.

These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP or SAP affiliate company 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.

In particular, SAP SE or its affiliated companies have no obligation to pursue any course of business outlined in this document or any related presentation, or to develop or release any functionality mentioned therein. This document, or any related presentation, and SAP SE's or its affiliated companies' strategy and possible future developments, products, and/or platform directions and functionality are all subject to change and may be changed by SAP SE or its affiliated companies at any time for any reason without notice. The information in this document is not a commitment, promise, or legal obligation to deliver any material, code, or functionality. All forward-looking statements are subject to various risks and uncertainties that could cause actual results to differ materially from expectations. Readers are cautioned not to place undue reliance on these forward-looking statements, and they should not be relied upon in making purchasing decisions.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. All other product and service names mentioned are the trademarks of their respective companies. See [www.sap.com/copyright](http://www.sap.com/copyright) for additional trademark information and notices.

**THE BEST RUN**

