Oracle® Fusion Cloud EPM

使用 Calculation Manager 進行設計

F28881-19

Oracle Fusion Cloud EPM 使用 Calculation Manager 進行設計,

F28881-19

版權所有 © 2008, 2025, Oracle 和 (或) 其關係公司。

主要作者：EPM Information Development Team

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software, software documentation, data (as defined in the Federal Acquisition Regulation), or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle®, Java, MySQL, and NetSuite are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

目錄

說明文件協助工具

說明文件意見

1 建立與執行 EPM Center of Excellence

2 Calculation Manager 的概觀

關於 Calculation Manager 2-1
使用 Calculation Manager 的服務 2-1
啟動 Calculation Manager 2-2
檢視 Calculation Manager 中的物件 2-2
自訂檢視中的欄 2-2
篩選物件 2-3
使用查詢篩選物件 2-3
查看規則、元件或範本的圖形圖片 2-3

3 設計商業規則

關於商業規則 3-2
設計商業規則的最佳作法 3-2
建立商業規則 3-7
建立 Groovy 商業規則 3-10
- 關於 Groovy 商業規則 3-10
- 為 ASO 立方體建立 Groovy 商業規則 3-11
- 為 BSO 立方體建立 Groovy 商業規則 3-12
- 為 Groovy 商業規則或範本編輯指令碼 3-13
- Groovy 規則適用的 Java API 參照 3-13
- Groovy 商業規則範例 3-14
- Groovy 商業規則教學課程影片 3-14
- Groovy 商業規則教學課程 3-15

開啟商業規則 3-16
編輯商業規則 3-17
編輯商業規則 3-17
在指令碼模式中編輯商業規則 3-18
在指令碼模式中進行編輯時的可用選項 3-19
儲存商業規則 3-20
儲存商業規則 3-20
使用不同的名稱儲存商業規則 3-20
執行商業規則 3-21
複製商業規則 3-21
在商業規則中搜尋 3-21
搜尋商業規則指令碼中的文字字串 3-21
搜尋並取代圖形商業規則中的文字 3-22
列印商業規則 3-22
刪除商業規則 3-22
定義商業規則元件中的常用維度 3-23
分析並偵錯商業規則 3-23
分析商業規則 3-23
隱藏追蹤維度的成員 3-25
正在擷取統計資訊 3-25
分析商業規則的指令碼 3-25
比較兩個商業規則的指令碼 3-25
比較變更的商業規則和儲存的商業規則 3-26
偵錯商業規則 3-27
停用商業規則中的元件 3-28
重新整理商業規則或商業規則集 3-28
重新整理中繼資料變更 3-29
顯示商業規則或規則集的使用情況 3-29
最佳化商業規則 3-29
商業規則最佳化的概觀 3-29
最佳化商業規則的步驟 3-29
在規劃應用程式活動報表中識別執行緩慢的商業規則 3-30
在 Calculation Manager 日誌訊息中識別執行緩慢的商業規則 3-30
使用日誌訊息以最佳化商業規則 3-31
範例商業規則 3-32

4 設計商業規則集

關於商業規則集 4-1
建立商業規則集 4-2

開啟商業規則集 4-2
開啟商業規則集中的商業規則 4-3
將商業規則新增至商業規則集 4-3
從商業規則集中移除商業規則 4-3
將商業規則集複製到另一個應用程式 4-3
儲存商業規則集 4-4
刪除商業規則集 4-4

5 使用系統範本

關於系統範本 5-1
顯示系統範本 5-2
使用系統範本 5-2
使用清除資料範本 5-2
使用複製資料範本 5-3
使用金額-單位-利率範本 5-4
使用配置 - 層級間範本 5-5
使用配置範本 5-7
使用聚總範本 5-8
使用 SET 命令範本 5-9
使用幣別轉換範本 5-10
幣別轉換範本範例 5-12
顯示範本流程 5-15
將系統範本另存為自訂範本 5-16
將系統範本從商業規則中移除 5-16

6 使用自訂範本

關於自訂範本 6-1
建立自訂範本 6-2
建立圖形自訂範本 6-2
建立使用 UpperPOV 設計時期提示之圖形範本的範例 6-3
建立指令碼自訂範本 6-4
為 Planning BSO 立方體建立 Groovy 範本 6-5
為 Planning ASO 立方體建立 Groovy 範本 6-5
針對自訂範本建立設計時期提示 6-6
設計階段提示的類型 6-6
屬性 DTP 6-6
布林值 DTP 6-7
跨維度 DTP 6-7

條件 DTP 6-8
DateAsNumber DTP 6-9
維度 DTP 6-10
維度 DTP 6-11
整數 DTP 6-12
成員 DTP 6-12
成員 DTP 6-14
成員範圍 DTP 6-15
數值 DTP 6-16
密碼 DTP 6-17
百分比 DTP 6-17
限制清單 DTP 6-18
分隔符號 DTP 6-19
智慧型列示 DTP 6-19
StringAsNumber DTP 6-20
字串 DTP 6-21
UDA DTP 6-21
為設計時期提示定義相依性 6-22
為設計時期提示定義限制 6-23
為設計時期提示建立步驟 6-23
尋找和取代設計時期提示中的文字 6-24
開啟自訂範本 6-25
重新整理自訂範本 6-25
顯示自訂範本的使用情況 6-25
複製和貼上自訂範本 6-25
刪除自訂範本 6-25
尋找和取代圖形自訂範本中的文字 6-26

7 使用元件以設計商業規則和範本

關於元件 7-1
公式元件 7-2
關於公式元件 7-2
建立公式元件 7-2
設計公式元件 7-3
使用條件建置器建立條件陳述式 7-6
輸入公式陳述式的備註 7-8
開啟公式元件 7-8
編輯公式元件 7-8
刪除公式元件 7-9

複製和貼上公式元件 7-9
指令碼元件 7-9
建立指令碼元件 7-10
設計指令碼元件 7-10
開啟指令碼元件 7-12
編輯指令碼元件 7-13
刪除指令碼元件 7-13
複製和貼上指令碼元件 7-13
條件元件 7-14
關於條件元件 7-14
建立條件元件 7-14
開啟條件元件 7-15
編輯條件元件 7-15
刪除條件元件 7-16
複製和貼上條件元件 7-16
成員區塊元件 7-16
關於成員區塊元件 7-17
建立成員區塊元件 7-17
開啟成員區塊元件 7-17
編輯成員區塊元件 7-17
刪除成員區塊元件 7-18
複製和貼上成員區塊元件 7-18
成員範圍元件 7-18
關於成員範圍元件 7-19
建立成員範圍元件 7-19
開啟成員範圍元件 7-20
編輯成員範圍元件 7-20
刪除成員範圍元件 7-21
複製和貼上成員範圍元件 7-21
固定迴圈元件 7-21
關於固定迴圈元件 7-21
建立固定迴圈元件 7-22
開啟固定迴圈元件 7-22
編輯固定迴圈元件 7-22
刪除固定迴圈元件 7-23
複製和貼上固定迴圈元件 7-23
中繼資料迴圈元件 7-23
關於中繼資料迴圈元件 7-24
建立中繼資料迴圈元件 7-24
開啟中繼資料迴圈元件 7-25

刪除中繼資料迴圈元件 7-25
複製和貼上中繼資料迴圈元件 7-25
DTP 指派元件 7-26
關於 DTP 指派元件 7-26
建立 DTP 指派元件 7-26
開啟 DTP 指派元件 7-27
編輯 DTP 指派元件 7-27
刪除 DTP 指派元件 7-27
複製和貼上 DTP 指派元件 7-27
在 DTP 指派元件中使用設計時期提示函式 7-28
關於設計時期提示函式 7-29
@AvailDimCount 7-29
@Compare 7-31
@Compliment 7-31
@Concat 7-32
@DenseMember 7-33
@Dependency 7-33
@DimAttribute 7-35
@DimMember 7-35
@DimName 7-36
@DimType 7-37
@DimUDA 7-37
@EndsWith 7-38
@Evaluate 7-38
@FindFirst 7-39
@FindLast 7-40
@GetData 7-41
@Integer 7-41
@Intersect 7-42
@IsAncest 7-42
@IsChild 7-43
@IsDataMissing 7-44
@IsSandBoxed 7-44
@IsVariable 7-45
@Length 7-45
@Matches 7-46
@Member 7-46
@MemberGeneration 7-47
@MemberLevel 7-48
@MsgFormat 7-49
@Notin 7-50

@OpenDimCount 7-51
@Plandim 7-52
@PlanDimMember 7-53
@Quote 7-54
@RemoveQuote 7-54
@ReplaceAll 7-55
@ReplaceFirst 7-56
@SmartListFromIndex 7-56
@SmartListFrom 值 7-57
@SparseMember 7-59
@StartsWith 7-59
@SubString 7-60
@ToLowerCase 7-61
@ToMDX 7-61
@ToUpperCase 7-63
@Trim 7-63
@Union 7-64
@ValueDimCount 7-65
共用指令碼和公式元件 7-66
關於共用指令碼和公式元件 7-66
將公式元件和指令碼元件從共用更改為不共用 7-66
將公式元件和指令碼元件從不共用變更為共用 7-67
複製元件 7-67
複製和貼上商業規則元件的子項 7-67
複製並貼上對商業規則公式或指令碼元件的參照 7-68
正在儲存元件 7-69
儲存元件 7-69
使用不同的名稱儲存公式元件和指令碼元件 7-69
重新整理公式元件和指令碼元件 7-69
顯示公式元件和指令碼元件的使用情況 7-70
在流程圖中使用元件 7-70
關於在流程圖中使用元件 7-70
在流程圖中收合和展開元件 7-70
從流程圖中移除元件 7-71
在流程圖中複製和貼上元件 7-71
在流程圖中複製和貼上對元件的參照 7-71
在流程圖中複製和貼上元件群組 7-72

8 使用聚總儲存元件設計商業規則

關於使用聚總儲存元件設計商業規則 8-1

使用檢視點元件 8-2
建立檢視點元件 8-2
編輯檢視點元件 8-6
使用配置元件 8-6
建立配置元件 8-6
編輯配置元件 8-10
開啟檢視點或配置元件 8-10
刪除檢視點或配置元件 8-10
複製和貼上檢視點或配置元件 8-11
儲存檢視點或配置元件 8-11
使用聚總儲存公式元件 8-11
建立聚總儲存公式元件 8-12
開啟聚總儲存公式元件 8-13
編輯聚總儲存公式元件 8-14
刪除聚總儲存公式元件 8-14
複製和貼上聚總儲存公式元件 8-15
將聚總儲存公式元件複製至另一個應用程式或資料庫 8-15
顯示聚總儲存公式元件的使用狀況 8-16

9 使用成員選擇、變數、函式、智慧型列示，以及 Planning 公式表示式以設計元件

關於成員選擇、變數、函式與智慧型列示 9-1
將成員和函式新增至元件 9-2
關於新增成員和函式到元件 9-2
將一或多個維度中的成員或函式新增到元件中 9-2
成員 9-3
函式 9-4
搜尋 9-6
從元件中移除成員和函式 9-6
搜尋成員 9-7
在「成員選取器」中搜尋「成員」 9-7
依名稱、別名或特性在成員選取器中搜尋成員 9-7
使用變數 9-8
關於變數 9-8
建立變數 9-9
輸入數值變數的變數值 9-10
輸入字串變數的變數值 9-11
輸入陣列變數的值 9-11
輸入成員範圍變數的值 9-12

輸入跨維度變數的變數值 9-13
輸入維度變數的變數值 9-13
輸入成員或成員變數的變數值 9-13
輸入百分比變數的變數值 9-14
輸入整數變數的值 9-15
針對「數字形式的字串」變數輸入變數值 9-15
針對「數字形式的日期」變數輸入變數值 9-15
輸入執行時期提示變數 9-16
選取變數 9-16
編輯變數 9-17
刪除變數 9-17
重新整理變數 9-18
複製變數 9-18
在變數設計工具尋找和取代文字 9-19
顯示變數的使用方法 9-20
使用函式 9-21
關於函式 9-21
將函式插入元件 9-22
Calculation Manager 所支援的 Essbase 函式 9-23
使用自訂函式 9-24
關於自訂函式 9-25
使用帶有年度參數的自訂函式 9-25
@CalcMgrExcel 帶有日期參數的自訂函式 9-25
位元函式 9-25
@CalcMgrBitAnd 9-26
@CalcMgrBitOR 9-26
@CalcMgrBitExOR 9-26
@CalcMgrBitExBoolOR 9-26
@CalcMgrBitCompliment 9-27
@CalcMgrBitShiftLeft 9-27
@CalcMgrBitShiftRight 9-27
@CalcMgrBitUnsignedShiftRight 9-27
計數器函式 9-27
@CalcMgrCounterAddNumber 9-28
@CalcMgrCounterAddText 9-28
@CalcMgrCounterClear 9-28
@CalcMgrCounterClearAll 9-29
@CalcMgrCounterClearKey 9-29
@CalcMgrCounterDecrement 9-29
@CalcMgrCounterDecrementKey 9-29

@CalcMgrCounterGetKeyNumber 9-29
@CalcMgrCounterGetKeyText 9-30
@CalcMgrCounterGetNumber 9-30
@CalcMgrCounterGetText 9-30
@CalcMgrCounterIncrement 9-30
@CalcMgrCounterIncrementKey 9-30
@CalcMgrCounterUpdate 9-31
@CalcMgrCounterUpdateNumber 9-31
@CalcMgrCounterUpdateNumberText 9-31
@CalcMgrCounterUpdateText 9-31
日期/時間函式 9-31
@CalcMgrAddDate 9-33
@CalcMgrAddDatePart 9-33
@CalcMgrAddDays 9-34
@CalcMgrAddMonths 9-34
@CalcMgrAddWeeks 9-34
@CalcMgrAddYears 9-34
@CalcMgrDateDiff 9-34
@CalcMgrDateToExcel 9-35
@CalcMgrDatesToExcel 9-35
@CalcMgrDateTimeToExcel 9-35
@CalcMgrDateTimesToExcel 9-36
@CalcMgrDateToString 9-36
@CalcMgrDaysBetween 9-36
@CalcMgrDaysDiff 9-36
@CalcMgrDiffDate 9-37
@CalcMgrExcelADD 9-37
@CalcMgrExcelDATE 9-37
@CalcMgrExcelDATEDIF 9-38
@CalcMgrExcelDAYOFYEAR 9-38
@CalcMgrExcelDAYS360 9-38
@CalcMgrExcelDAYSINMONTH 9-39
@CalcMgrExcelEOMONTH 9-39
@CalcMgrExcelHOUR 9-40
@CalcMgrExcelMINUTE 9-40
@CalcMgrExcelMONTH 9-40
@CalcMgrExcelNETWORKDAYS 9-40
@CalcMgrExcelSECOND 9-41
@CalcMgrExcelToDate 9-41
@CalcMgrExcelToDateTime 9-42
@CalcMgrExcelWEEKNUM 9-42
@CalcMgrExcelWEEKDAY 9-42

@CalcMgrExcelWORKDAY 9-43
@CalcMgrExcelYEAR 9-43
@CalcMgrExcelYEARFRAC 9-43
@CalcMgrGetCurrentDate 9-44
@CalcMgrGetCurrentDateTZ 9-44
@CalcMgrGetCurrentDateTime 9-44
@CalcMgrGetCurrentDateTimeTZ 9-45
@CalcMgrGetCustomDate 9-45
@CalcMgrGetCustomDateTime 9-45
@CalcMgrGetDatePart 9-45
@CalcMgrGetDateTimePart 9-46
@CalcMgrGetDay 9-46
@CalcMgrGetDayOfYear 9-46
@CalcMgrGetFormattedDate 9-46
@CalcMgrGetMaxDaysInMonth 9-46
@CalcMgrGetMonth 9-47
@CalcMgrGetStringFormattedDateTime 9-47
@CalcMgrGetWeekOfMonth 9-47
@CalcMgrGetWeekOfYear 9-47
@CalcMgrGetYear 9-48
@CalcMgrIsLeapYear 9-48
@CalcMgrMonthsBetween 9-48
@CalcMgrMonthsDiff 9-48
@CalcMgrRollDate 9-49
@CalcMgrRollDay 9-49
@CalcMgrRollMonth 9-49
@CalcMgrRollYear 9-49
@CalcMgrWeeksBetween 9-50
@CalcMgrWeeksDiff 9-50
@CalcMgrYearsBetween 9-50
@CalcMgrYearsDiff 9-51
財務函式 9-51
@CalcMgrExcelACCRINT 9-52
@CalcMgrExcelACCRINTM 9-53
@CalcMgrExcelAMORDEGRC 9-53
@CalcMgrExcelAMORLINC 9-53
@CalcMgrExcelCOUPDAYBS 9-53
@CalcMgrExcelCOUPDAYS 9-54
@CalcMgrExcelCOUPDAYSNC 9-54
@CalcMgrExcelCOUPNCD 9-55
@CalcMgrExcelCOUPNUM 9-55
@CalcMgrExcelCOUPPCD 9-55

@CalcMgrExcelCUMIPMT 9-56
@CalcMgrExcelCUMPRINC 9-56
@CalcMgrExcelDB 9-56
@CalcMgrExcelDDB 9-56
@CalcMgrExcelDISC 9-57
@CalcMgrExcelDOLLARDE 9-57
@CalcMgrExcelDOLLARFR 9-57
@CalcMgrExcelDURATION 9-57
@CalcMgrExcelEFFECT 9-58
@CalcMgrExcelFV 9-58
@CalcMgrExcelFVSCHEDULE 9-58
@CalcMgrExcelMDURATION 9-59
@CalcMgrExcelINTRATE 9-59
@CalcMgrExcelPMT 9-59
@CalcMgrExcelIRR 9-60
@CalcMgrExcelISPMT 9-60
@CalcMgrExcelMIRR 9-60
@CalcMgrExcelNPER 9-60
@CalcMgrExcelNPV 9-61
@CalcMgrExcelPPMT 9-61
@CalcMgrExcelPRICE 9-61
@CalcMgrExcelPRICEDISC 9-61
@CalcMgrExcelPRICEMAT 9-62
@CalcMgrExcelPV 9-62
@CalcMgrExcelRATE 9-62
@CalcMgrExcelRECEIVED 9-63
@CalcMgrExcelSLN 9-63
@CalcMgrExcelSYD 9-63
@CalcMgrExcelTBILLEQ 9-63
@CalcMgrExcelTBILLPRICE 9-64
@CalcMgrExcelTBILLYIELD 9-64
@CalcMgrExcelXIRR 9-65
@CalcMgrExcelXNPV 9-65
@CalcMgrExcelYIELD 9-65
@CalcMgrExcelYIELDDISC 9-66
@CalcMgrExcelYIELDMAT 9-66
記錄函式 9-66
@CalcMgrLogMessageTrace 9-67
@CalcMgrIsValidMember 9-67
@CalcMgrIsValidSLMember 9-67
@CalcMgrSLMember 9-67
數學函式 9-68

@CalcMgrExcelCEILING 9-68
@CalcMgrExcelCOMBIN 9-68
@CalcMgrExcelEVEN 9-69
@CalcMgrExcelFACT 9-69
@CalcMgrExcelFLOOR 9-69
@CalcMgrExcelGCD 9-69
@CalcMgrExcelLCM 9-69
@CalcMgrExcelMROUND 9-70
@CalcMgrExcelMULTINOMIAL 9-70
@CalcMgrExcelODD 9-70
@CalcMgrExcelPOWER 9-70
@CalcMgrExcelPRODUCT 9-70
@CalcMgrExcelROUNDDOWN 9-71
@CalcMgrExcelROUNDUP 9-71
@CalcMgrExcelSQRT 9-71
@CalcMgrExcelSQRTPI 9-71
@CalcMgrExcelSUMPRODUCT 9-71
@CalcMgrExcelSUMSQ 9-72
統計函式 9-72
@CalcMgrExcelAVEDEV 9-72
CalcMgrExcelBINOMDIST 9-73
@CalcMgrExcelDEVSQ 9-73
@CalcMgrExcelLARGE 9-73
@CalcMgrExcelMEDIAN 9-73
@CalcMgrExcelNORMSDIST 9-74
@CalcMgrExcelNORMSINV 9-74
@CalcMgrExcelPERCENTILE 9-74
@CalcMgrExcelPERCENTRANK 9-74
@CalcMgrExcelRANK 9-75
@CalcMgrExcelSMALL 9-75
@CalcMgrExcelSTDEV 9-75
@CalcMgrExcelVAR 9-75
@CalcMgrExcelVARP 9-76
@CalcMgrIsFinite 9-76
字串函式 9-76
@CalcMgrCompare 9-77
@CalcMgrConcat 9-78
@CalcMgrDecimalFormat 9-78
@CalcMgrDoubleFromString 9-78
@CalcMgrDoubleToString 9-78
@CalcMgrDQuote 9-78
@CalcMgrEndsWith 9-79

@CalcMgrFindFirst 9-79
@CalcMgrFindLast 9-79
@CalcMgrFormatDouble 9-79
@CalcMgrGetListCount 9-79
@CalcMgrGetListItem 9-80
@CalcMgrIndexOf 9-80
@CalcMgrIntegerToString 9-80
@CalcMgrLastIndexOf 9-80
@CalcMgrLowercase 9-81
@CalcMgrMatches 9-81
@CalcMgrMessageFormat 9-81
@CalcMgrPadText 9-81
@CalcMgrUppercase 9-82
@CalcMgrRemoveQuotes 9-82
@CalcMgrRemoveDQuotes 9-82
@CalcMgrRemoveSQuotes 9-82
@CalcMgrReplaceAll 9-82
@CalcMgrReplaceFirst 9-83
@CalcMgrSortAndReturn 9-83
@CalcMgrSortList 9-83
@CalcMgrSortValues 9-83
@CalcMgrSplit 9-84
@CalcMgrSQuote 9-84
@CalcMgrStartsWith 9-84
@CalcMgrStringsToString 9-84
@CalcMgrSubstring 9-84
@CalcMgrTextLength 9-85
@CalcMgrTrim 9-85
使用智慧型列示 9-85
關於智慧型列示 9-85
插入智慧型列示 9-85
使用 Planning 公式表示式 9-86
智慧型列示 9-86
維度 9-87
規劃使用者變數 9-87
期間 9-88
Period(periodName) 9-88
NumberofPeriodsinYear 與 NumberofYears 9-89
案例 9-89
交互參照 9-93
CrossRef(accountName) 9-93

CrossRef(accountName, prefix) 9-94
CrossRef(accountName, prefix, true) 9-94
人力立方體年累計 9-95
CYTD(memberName) 9-95
CYTD(memberName, calTpIndexName, fiscalTPIndexName) 9-95
獲取字串 ID 9-96
搭配 Essbase 中的「混合彙總」使用 9-96
混合彙總中的動態計算 9-96
「混合彙總」不支援計算命令 9-97
「混合彙總」不支援函式 9-98

10 驗證與部署

從系統檢視驗證商業規則、商業規則集以及公式元件和指令碼元件 10-1
從規則設計工具驗證商業規則 10-2
部署商業規則和商業規則集 10-2
關於部署商業規則和商業規則集 10-2
將商業規則和商業規則集設為可部署和不可部署 10-3
從部署檢視部署商業規則和商業規則集 10-3
從規則或規則集設計工具部署商業規則或商業規則集 10-4
部署商業規則 (含捷徑) 10-4
指定哪個已部署的商業規則會顯示在 Planning 中 10-5

11 啟動商業規則

關於啟動商業規則 11-1
從規則設計工具啟動 Planning 商業規則和檢視日誌 11-1

12 匯出和匯入商業規則、商業規則集、範本、公式和指令碼元件

關於匯出和匯入 12-1
匯出商業規則、商業規則集、範本、公式元件和指令碼元件 12-1
匯出應用程式 12-2
將日誌訊息匯出至檔案 12-2
匯入規則、規則集、範本、公式和指令碼 12-3

13 為規劃應用程式管理 Essbase 伺服器、應用程式和資料庫

使用資料庫特性 13-2
檢視和編輯資料庫特性 13-2
一般資料庫特性 13-2

維度特性 13-4
統計特性 13-4
聚總儲存應用程式的統計 13-4
區塊儲存應用程式的統計 13-5
交易特性 13-6
修改特性 13-7
移除資料庫物件的鎖定 13-7
啟動和停止應用程式 13-8
啟動和停止資料庫 13-8
重新建構資料庫 13-9
驗證大綱 13-9
從資料庫清除資料 13-10
將資料從聚總儲存應用程式中清除 13-11
將資料區塊從區塊儲存應用程式中清除 13-11
使用區塊儲存應用程式的位置別名 13-12
關於位置別名 13-12
顯示位置別名的清單 13-12
匯出位置別名 13-12
在聚總儲存資料庫上使用查詢追蹤 13-13
壓縮聚總儲存資料庫大綱 13-13
匯入及匯出層級 0 資料 13-14
從 ASO 立方體匯入層級 0 資料 13-14
從 ASO 立方體匯出層級 0 資料 13-15
從 BSO 立方體匯入層級 0 資料 13-15
從 BSO 立方體匯出層級 0 資料 13-15
合併累加式資料塊 13-16
聚總資料 13-16
執行聚總程序 13-17
合併累加式資料塊與移除零值的儲存格 13-17
啟用查詢追蹤 13-18
執行建立查詢的動作 13-18
使用查詢追蹤執行聚總 13-18
管理要求 13-19
新增規劃鑽研定義 13-20

說明文件協助工具

如需有關 Oracle 對於協助工具的承諾資訊，請瀏覽 Oracle Accessibility Program 網站，網址為 http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc。

取得 Oracle 支援

已經購買客戶支援的 Oracle 客戶可從 My Oracle Support 取得網路支援。如需資訊，請瀏覽 http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info；如您有聽力障礙，請瀏覽 http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs。

說明文件意見

若您對此說明文件有任何意見，請按一下任何「Oracle 說明中心」主題中頁面底部的「意見」按鈕。您也可以寄送電子郵件給 epmdoc_ww@oracle.com。

1

建立與執行 EPM Center of Excellence

EPM 的最佳作法是建立 CoE (Center of Excellence)。

EPM CoE 是確保應用與最佳作法的共同努力成果，可推動與績效管理和使用技術解決方案的相關商業程序轉型。

組織可透過雲端應用，改善業務靈活性與提升創新解決方案。EPM CoE 可監督您的雲端先導計畫，能夠協助保護與維護您的投資，並促進有效率的使用。

EPM CoE 團隊：

- 確保雲端應用，協助組織充分利用您的 Oracle Fusion Cloud EPM 投資
- 作為最佳作法的指導委員會
- 領導 EPM 相關變革管理初步計畫並推動轉型

所有客戶都能從 EPM CoE 獲益，包含已實作 EPM 的客戶在內。

如何開始使用？

按一下以取得適用於您自己的 EPM CoE 最佳作法、指導和策略：EPM Center of Excellence 簡介。

進一步瞭解

- 觀看 Cloud Customer Connect 網路研討會：建立與執行 Cloud EPM 的 Center of Excellence (CoE)
- 觀看影片：概觀：EPM Center of Excellence 和 建立 Center of Excellence。
- *建立與執行 EPM Center of Excellence* 中的查看 EPM CoE 的業務優勢和價值主張。

2

Calculation Manager 的概觀

另請參閱：

- 關於 Calculation Manager
  請使用 Calculation Manager 來建立、驗證、部署和啟動可解決商業問題的計算。
- 使用 Calculation Manager 的服務
  多項 Enterprise Performance Management Cloud 服務使用 Calculation Manager。
- 啟動 Calculation Manager
  啟動 Calculation Manager 以建立規則、規則集、元件，以及範本。
- 檢視 Calculation Manager 中的物件
  檢視可讓您查看不同環境定義中的 Calculation Manager 物件。
- 自訂檢視中的欄
  您可以自訂每個檢視中顯示的欄，以及欄的顯示順序。
- 篩選物件
  在「系統檢視」或「篩選檢視」中，篩選 Calculation Manager 中的物件。
- 使用查詢以篩選物件
  在篩選條件檢視中篩選物件之後，即可使用查詢進一步精簡要顯示的物件。
- 查看規則、元件或範本的圖形圖片
  規則、元件和範本會在「規則設計工具」與「範本設計工具」的流程圖中，以圖形方式顯示。

關於 Calculation Manager

請使用 Calculation Manager 來建立、驗證、部署和啟動可解決商業問題的計算。

您可以在計算中建立下列類型的物件：

- **規則** - 包含元件、範本和其他規則的物件
- **規則集** - 包含可同時或依序計算之兩個或多個商業規則的物件
- **元件** - 包含公式、指令碼、條件式、成員和資料範圍、固定迴圈，和設計時期提示的物件。(元件是不可部署的。)
- **範本** - 您可以在商業規則中用於執行計算或計算集合的物件。

使用 Calculation Manager 的服務

多項 Enterprise Performance Management Cloud 服務使用 Calculation Manager。

- Planning
- Planning 模組
- Enterprise Profitability and Cost Management
- Financial Consolidation and Close

啟動 Calculation Manager

啟動 Calculation Manager 以建立規則、規則集、元件，以及範本。

若要啟動 Calculation Manager，請執行下列動作：

1. 按一下 Planning 首頁左上角的 ☰ 。
2. 在**建立及管理**中，按一下**規則**。

檢視 Calculation Manager 中的物件

檢視可讓您查看不同環境定義中的 Calculation Manager 物件。

當您在檢視中，您可以使用**檢視**功能表中的選項，以自訂化顯示的欄，以及顯示順序。

啟動 Calculation Manager 時，會自動顯示**系統檢視**。若要切換檢視，請從**選取檢視**旁的下拉清單選取檢視。

Calculation Manager 包含下列檢視：

- **系統檢視** — 當您啟動 Calculation Manager 時，會顯示預設檢視。其列出您具有存取權的所有應用程式和物件。

  您的存取權限由在應用程式管理指派給您的角色決定。存取權限根據應用程式指派。
- **自訂檢視** — 建立資料夾並將物件新增至資料夾，以建立對您有意義的檢視。

  若要在**自訂檢視**中建立資料夾：

  1. 請在右側窗格中，在應用程式按一下右鍵，然後選取**新增**，然後選取**資料夾**。
  2. 在**新資料夾**中，輸入資料夾名稱，然後按一下**確定**。
  3. 藉由將物件從**現有物件**窗格拖曳至資料夾，以新增物件至資料夾中。
- **部署檢視** — 依應用程式類型與應用程式列示可部署的規則與規則集，及其部署與驗證狀態。您可以選取可部署的規則與規則集，然後將物件部署至應用程式。

  在應用程式中部署一或多個規則和規則集稱為部分部署。在應用程式中部署所有規則和規則集稱為完整部署。
- **篩選檢視** — 篩選顯示的物件。

  用於在**篩選**對話方塊中定義篩選選項，然後使用查詢，以進一步精簡篩選。

自訂檢視中的欄

您可以自訂每個檢視中顯示的欄，以及欄的顯示順序。

若要自訂檢視中的欄，請執行下列其中一項動作：

- 依序選取**檢視**、**欄**，然後選取要顯示的欄。藉由依序選取**檢視**、**重新排序欄**，然後選取欄順序，以重新排序欄。
- 依序選取**檢視**、**欄**，然後選取**管理欄**。在**管理欄**對話方塊中，選取要顯示的欄，以及顯示順序。

> **備註：**
>
> Oracle Financials Cloud 中有一個新的欄「解除鎖定」功能。預設並不會顯示此欄。解除鎖定功能可讓管理員解除鎖定被其他管理員鎖定的物件。

篩選物件

在「系統檢視」或「篩選檢視」中，篩選 Calculation Manager 中的物件。

您可以根據其應用程式類型、應用程式、計算類型、計畫類型、資料庫、物件類型 (即商業規則、商業規則集、公式和指令碼元件及範本)，以及其部署或驗證狀態，來篩選物件。

若要篩選物件，請執行下列動作：

1. 在**系統檢視**或**篩選檢視**中，按一下 。
2. 在**篩選**對話方塊中，輸入要求的資訊，然後按一下**確定**。

使用查詢篩選物件

在篩選條件檢視中篩選物件之後，即可使用查詢進一步精簡要顯示的物件。

若要使用查詢篩選物件，請執行下列動作：

1. 在**篩選條件檢視**中，按一下 ，然後在**篩選**對話方塊中輸入資訊。
2. 按一下 ，在欄上方顯示文字方塊。
3. 在欄上方的文字方塊中，輸入要進一步精簡篩選條件的查詢。

   您無法使用萬用字元或部分文字字串。若要搜尋計畫類型，請輸入第一個字的開頭字元。例如，如果規劃類型的名稱為 "Plan1"，而您在查詢中輸入 "la"，將只會顯示以 "La" 開頭的物件。在此範例中，若要獲得想要的結果，您將輸入 "Pl" 以顯示以 "Pl" 開頭的所有規劃類型。

> **備註：**
>
> 您也可以在「變數設計工具」中使用查詢來篩選物件。

查看規則、元件或範本的圖形圖片

規則、元件和範本會在「規則設計工具」與「範本設計工具」的流程圖中，以圖形方式顯示。

開啟規則或範本時，您可以在流程圖中選定元件 (例如公式、指令碼、條件、範圍和迴圈)，以查看詳細資料。也可以放大或縮小流程圖的大小，以檢視或隱藏元件的詳細資料。

在流程圖中選取一個元件後，其特性、使用狀況和其他資訊都會顯示在流程圖下方的頁籤中。當您在元件之間移動時，流程圖下方的頁籤也會改變。

例如，如果您開啟包含公式元件和指令碼元件的公式，並選取流程圖中的公式元件，公式的特性 (名稱、描述、應用程式和應用程式類型) 會顯示在流程圖下方的頁籤中。如果您在流程圖中選取指令碼元件，則該指令碼元件的文字、特性和使用狀況，都會顯示在流程圖下方的頁籤中。

3

設計商業規則

另請參閱：

- 關於商業規則
  Calculation Manager 可讓您建立、驗證、部署和管理複雜的多維商業規則。
- 設計商業規則的最佳作法
- 建立商業規則
  商業規則是 Calculation Manager 物件，該物件是由分組至元件的計算組成。
- 建立 Groovy 商業規則
  Oracle 支援建立以 Groovy 指令碼語言所撰寫的商業規則。
- 開啟商業規則
  開啟 Calculation Manager 時，您可以從預設顯示的「系統檢視」開啟商業規則。
- 編輯商業規則
- 儲存商業規則
  將商業規則儲存至為其建立的應用程式和應用程式類型。
- 執行商業規則
  執行之前，您必須先開啟商業規則。
- 複製商業規則
  您可以將商業規則複製到另一個應用程式和計畫類型。
- 在商業規則中搜尋
  搜尋商業規則指令碼中的文字字串。搜尋並取代圖形商業規則中的文字。
- 列印商業規則
  您可以列印商業規則的特性、流程圖及元件詳細資料。
- 刪除商業規則
  您只能刪除並未被其他規則或規則集使用的商業規則。
- 定義商業規則元件中的常用維度
  藉由開啟業務規則，並選取每個維度常用的成員、變數和函式，來定義常用維度。
- 分析並偵錯商業規則
  藉由執行規則並擷取統計資訊以分析商業規則。藉由執行規則及檢查其指令碼以偵錯商業規則。
- 重新整理商業規則或商業規則集
  在「系統檢視」、「自訂檢視」和「部署檢視」中，您可以重新整理任何層級的應用程式清單。
- 重新整理中繼資料變更
- 顯示商業規則或規則集的使用情況
  顯示正在使用某商業規則或商業規則集的規則、範本和規則集。
- 最佳化商業規則
  利用可用工具並有效管理您的規則。

關於商業規則

Calculation Manager 可讓您建立、驗證、部署和管理複雜的多維商業規則。

建立商業規則一般是為了：

- 在實體之間配置成本
- 模型收入
- 模型費用
- 製作資產負債表
- 計算現金流量
- 計算外匯折算調整
- 計算集團股權和少數股權
- 計算遞延稅

在建立商業規則之前，您應熟悉資料庫大綱和使用的應用程式。此資訊將可助您更有效率地建立商業規則。您還必須了解資料的下列資訊：

- 資料儲存和聚總方式
- 資料載入資料庫的層級
- 計算順序
- 進行計算的關鍵假設

您可以使用諸如公式、指令碼、迴圈、資料和成員範圍、範本和變數 (包括執行時期提示變數) 等元件建立商業規則。(請參閱使用元件以設計商業規則和範本。)

當您建立商業規則時，您可以保留您正使用開啟的元件、範本以及變數。Calculation Manager 會在標籤式介面中顯示這些物件，以便您可以在建立規則時，輕鬆地在這些標籤之間移動。您可以開啟多達十個頁籤。不過，對於獲得最佳效能，您不應該同時開啟多達十個頁籤。

若要在規劃中啟動商業規則，管理員將啟動權限提供給規則。

> **備註：**
>
> 您也可以建立包含兩個以上可同時或依序啟動之相關規則 (或規則集) 的商業規則集。請參閱設計商業規則集。

設計商業規則的最佳作法

編寫不當的規則會對應用程式的各方面產生重大影響。以下是設計商業規則時的一些最佳作法。遵循最佳作法建議可以帶來顯著的效能優勢。

請遵循以下商業規則的重要設計準則。

- 規則的前 10 個最佳作法
- 使用計算新增商業邏輯

- 建置聚總
- 設定檢視點
- 建置詳細計算
- Calculation Manager 診斷
- 範例規則問題與解決方案

規則的前 10 個最佳作法

設計商業規則時，請遵循以下最佳作法建議。這些準則可以帶來顯著的效能優勢，因為編寫不當的規則會對應用程式的各方面產生重大影響。

1. 對於 SET 命令，請遵循以下準則：
   - 在規則頂端，請勿使用 `SET CREATEBLOCKONEQ ON` 和 `SET CREATENONMISSINGBLK ON`。
   - 請勿在一般使用者規則中使用像這樣的管理類型命令，因為這會需要重新建立結構。

     ```
     SET CLEARBLOCK
                       EMPTY
     ```

   - 避免使用或測試使用 `SET CALCTASKDIMS` 的規則。(Oracle Essbase 通常會自動執行此作業。)
   - 資料複製規則應包含以下內容，以避免複製 Fix 陳述式中的空白區塊。

     ```
     SET COPYMISSINGBLOCK
                       OFF
     ```

2. 應使用資料複製或稀疏成員指派來完成建立區塊。`@createblockoneq` 和 `@createblock` 函數應作為有限 Fix 陳述式內的最後手段。
3. 避免在 Fix 陳述式中缺少維度參照 (例如，在資料複製規則中)。這會浪費處理時間、增加資源競爭，並且會針對缺少維度的所有層級建立不必要的區塊。
4. 移除商業規則中與表單關聯的平行計算。Calc Parallel 或 Fix Parallel 應只與管理/批次規則搭配使用。
5. 請勿建立非必要的零，因為這會導致區塊和資料暴增。請仔細複查商業邏輯並新增必要的 If 條件來檢查零。將零轉換為 `#missing`。
6. 排除對相同區塊的多次傳送；而是設定適當的外部 fix，以及根據需要的移入和移出。合併 If 陳述式，而不是在相同交集上使用與重複使用 If。
7. 避免在等式左邊使用跨維參照。這會導致效能影響。
8. 依指令碼中建立最多區塊到建立最少區塊的順序來聚總維度，例如，Agg (1st Most Blocks Dimension, 2nd Most Blocks Dimension, 3rd Most Blocks Dimension)。Agg 的速度比 Calc Dim 更快，是偏好使用的聚總機制。在一般使用者規則中一直使用 `@ancestors` 聚總至維度頂端可能會導致區塊競爭。
9. 使用執行時期提示，而不是建立多個具有相同基礎邏輯的規則。規則越多，需要的維護越多。
10. 使用範本來分解與重複使用商業邏輯。但是，範本不應該是具有 Fix、EndFix 的功能齊全規則。結合各種範本的規則應該有一個適當的外部 Fix，並視需要移入與移出較小的片段。

使用計算新增商業邏輯

若要將商業邏輯納入到應用程式中，您可以使用 Calculation Manager 建置計算。這可讓您建立、驗證、部署及管理可解決商業問題的複雜計算。

建立商業規則和規則集一般是為了：

- 執行收入模型化
- 執行支出模型化
- 計算 KPI
- 執行配置

Calculation Manager 包括這些物件：

- 規則：包含元件和範本
- 元件：協助您建立規則
- 規則集：包含可同時或循序計算的規則
- 範本：包括執行計算的系統範本和可由管理員設計的自訂範本
- 元件：協助您建置規則

若要深入瞭解如何建立計算，請參閱使用 Oracle Planning and Budgeting Cloud Service 的 Calculation Manager 來設計中的準則。

建置聚總

聚總會將應用程式向上彙整至維度中的摘要層級成員，例如 Entity 或是任何其他稀疏維度。

Calculation Manager 包括範本來協助您建立聚總。以下是如何使用範本的一些建議。

設定檢視點

設定檢視點時，只會針對所選的成員執行規則。為維度使用執行時期提示，可讓使用者在啟動規則時為這些維度指定成員值。如此一來，使用者就可以針對不同年度、案例和版本啟動規則數次，而不必在 Calculation Manager 中修改規則。

一般設定值：

- 完整密集聚總：如果在密集維度中的父項值未設定為動態計算，請完成此區段。通常此頁籤會保留空白。
- 完整稀疏聚總：請選取需要聚總的稀疏維度。選取維度的順序不相關。
- 局部維度聚總，密集：如果在密集維度中的父項值未設定為動態計算，請完成此區段。通常此頁籤會保留空白。
- 將資料聚總至當地幣別：否
- 聚總資料庫中缺少的值：是
- 最佳化「稀疏」維度上的計算：關閉
- 選取計算器快取的值：預設值
- 您是否要啟用此精靈的除錯模式？：「除錯精靈開啟」或「除錯精靈關閉」。如果您想要查看產生的指令碼，以顯示在此範本中某些設計階段提示的選擇，請選取「偵錯精靈」。

最佳作法：

- 為成員運用執行階段提示，例如 Entity、Scenario 與 Version。這可讓您的規則變成動態並且根據使者輸入執行。
- 通常不需要將 Account 與 Period 等密集維度聚總。如果是這樣，您可以將父成員設定為動態計算。然而，如果您在密集維度上有成員公式，而且未將它們設為動態計算，則會需要 Calc Dim 規則。

建置詳細計算

您可以使用 Calculation Manager 來建立、驗證、部署和管理可解決商業問題的計算。

Calculation Manager 中可計算的物件類型有三種：

- 規則集：包含可同時或循序計算的規則
- 規則：包含元件和範本
- 元件：包含公式元件、指令碼元件、條件元件、範圍元件，以及固定迴圈元件

最佳作法：

- 建置規則的第一步，是確保您瞭解商業邏輯以及此規則會套用至哪些實體或部門。例如，瞭解在規則中所包含的科目。
- 請務必瞭解來源與目標科目。
- 在您完全瞭解計算的動因後，請使用適當的物件元件或是範本以建置規則。元件和範本會協助成員選擇以協助部署規則。
- 為成員運用執行階段提示 (例如 Entity、Scenario 與 Version)，可讓您的規則變成動態並根據使用者輸入執行。

Calculation Manager 診斷

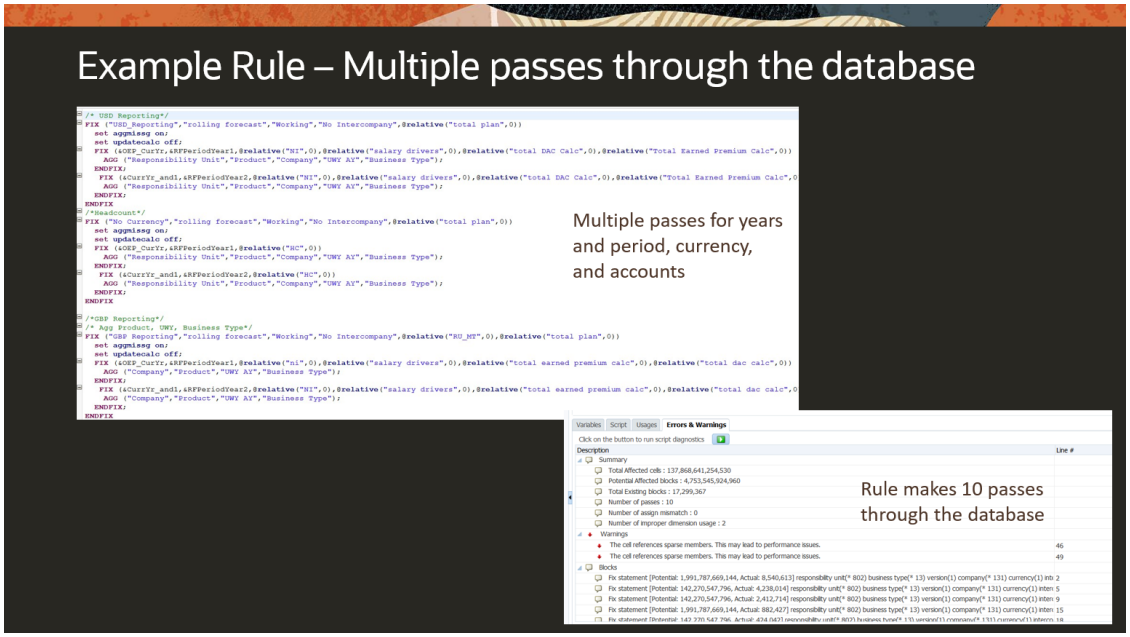
在部署規則之前執行錯誤和警告。這可提供有用的資訊，包括：

- 通過資料庫的數目
- 任何必要的警告
- 關於區塊數目以及是否缺少維度的資訊
- 任何需要最佳化的規則
- 如果等式右邊有任何元件包含零，則衍生的成員將會是 0。則聚總之後就會有很多 0。
- 若要解決此問題，規則應該包含一個 if 陳述式，說明 `if (<q>Earned Premium</q><>0)`。如此，零在應用程式中就不會那麼重要。

範例規則問題與解決方案

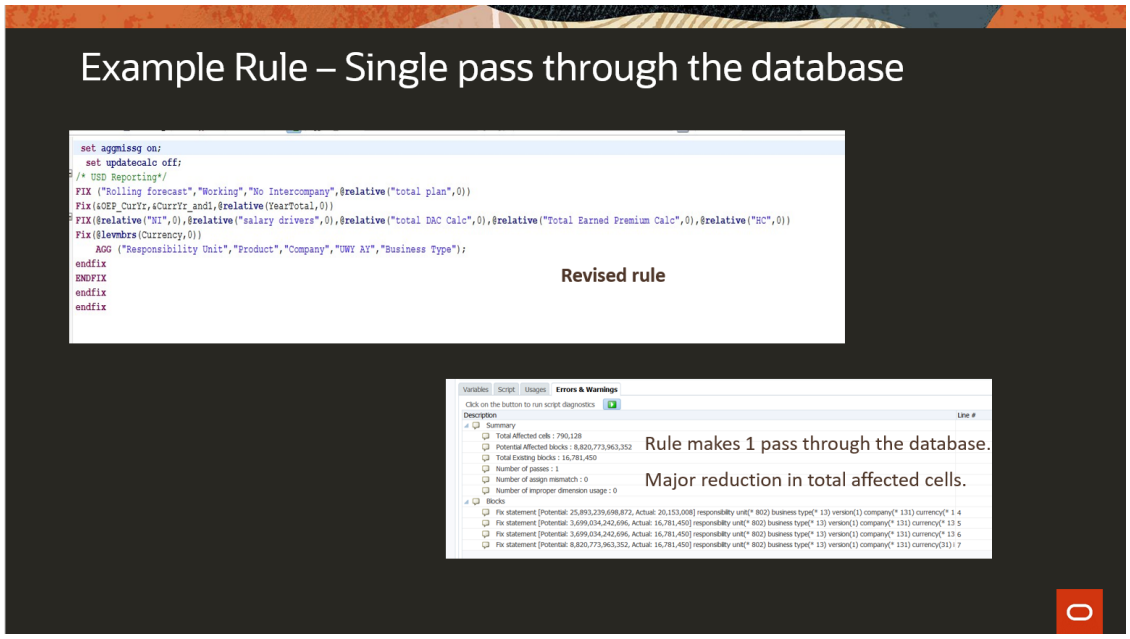
範例 1：會通過資料庫多次的規則

以下規則會針對年份和期間、幣別和帳戶通過 10 次資料庫。



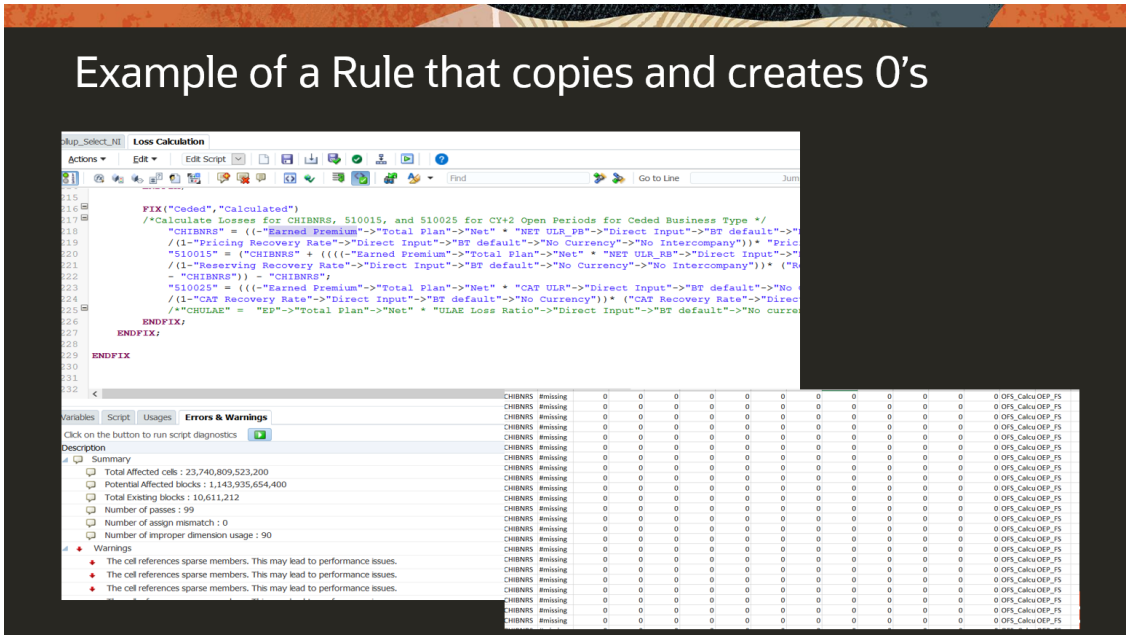
範例 2：只會通過資料庫一次的規則

以下規則只會通過資料庫一次。這會大幅減少受影響的儲存格總數。



範例 3：會複製與建立零的規則

在此規則中，如果等式右邊有任何元件包含 0，則衍生的成員將會是 0。聚總之後就會有很多 0。若要解決此問題，規則應該包含一個 if 陳述式，說明 `if ("Earned Premium"<>0)`。如此，零在應用程式中就不會那麼重要。



範例 4：從 BSO 立方體移除零

以下公式會得到原始值，或若為 0，則會變更為 #missing。

```
"DenseMbr" = "DenseMbr" * "DenseMbr"/"DenseMbr";
```

```
"SparseMbr" = "SparseMbr" * "SparseMbr"/"SparseMbr";
```

若要解決此問題，請建立一個利用此公式的規則。請使用包含最少成員的稀疏維度 (例如年度、案例或版本)，因為需要對每個成員進行計算。

```
Fix(idescendants(Entity),@levmbrs(Accounts,0),@levmbrs(Period,0),@idescendant
s(Custom dimensions),
@levmbrs(Years,0),@levmbrs(Versions))
Actual=Actual*Actual/Actual;
   Fix("Actual")
      CLEARBLOCK EMPTY:
   Endfix
Endfix
```

之後請記得進行重新建立結構。否則，將不會刪除空白區塊。

建立商業規則

商業規則是 Calculation Manager 物件，該物件是由分組至元件的計算組成。

一個規則可包含一或多個元件、範本或規則。

您可以為您有存取權的應用程式建立商業規則。您建立規則的能力，由指派給您的角色決定。(請參閱 *管理 Oracle Enterprise Performance Management Cloud 的使用者佈建*)。

規則以圖形方式顯示在流程圖中，您可以在其中拖放元件以設計規則。

若要建立商業規則，請執行下列動作：

1. 請執行下列其中一項動作：

- 在**系統檢視**、**自訂檢視**、**部署檢視**或**篩選檢視**中，按一下 ，然後在**新增物件**對話方塊中輸入資訊。務必選取**規則**做為**物件類型**。
- 在**系統檢視**中，在**規則**按一下右鍵，然後選取**新增**，然後在**新增規則**對話方塊中輸入資訊。

> **備註：**
>
> 規則名稱中不允許下列字元：'\r'、'\n'、'\t'、'\f'、'\b'、'<'、'>'、'('、')'、'"'、'\\'、'{'、'}'、'['、']'、'*'、'?'

2. 在「規則設計工具」中，在**新增物件**與**現有物件**下選取物件，然後將其拖曳至**開始**和**結束**之間的流程圖。

   請注意下列事項：

   - 當您將現有公式元件或指令碼元件拖曳到流程圖中時，依照預設，公式或指令碼會成為共用物件。如果並不想共用，可以清除公式元件或指令碼元件**特性**上的**共用**核取方塊。請參閱共用指令碼和公式元件。
   - 若要在其圖形格式中檢視指令碼元件，請在流程圖中的指令碼元件按一下右鍵，然後選取**轉換至圖形**。

     只有在指令碼為有效時，指令碼元件才能轉換為圖形格式。若要還原圖形格式的轉換，請用滑鼠右鍵按一下元件，然後選取**還原**。
   - 您可以建立獨立於規則外的物件 (如公式和指令碼)，之後再將它們新增至規則。
   - 若要以非圖形格式 (指令碼格式) 使用商業規則，請按一下**設計工具**旁的下拉清單，然後選取**編輯指令碼** (請參閱在指令碼模式中編輯商業規則。)

3. 輸入規則的**特性**。

   特性會隨著您在規則中新增元件，以及在流程圖的元件之間移動而變化。若要為特定元件輸入特性，請在流程圖中選取該元件。

   當您在流程圖中選取**開始**或**結束**時，會顯示下列特性。

   - **一般**：名稱、描述與備註
   - **位置**：應用程式與計畫類型
   - **選項**：專用於目前應用程式的選項

表格 3-1　選項

特性	描述
建立動態成員	當您在已啟用執行階段提示之成員類型的變數中指定預設動態父項時，會建立成員。 如果您在商業規則中建立動態成員，然後選取預設動態父項，便會在應用程式啟動規則*之前*自動建立該父項的子成員。 **注意：**如果您選取建立動態成員，但商業規則啟動失敗，新建立的成員會被刪除。 **注意：**Financial Consolidation and Close 應用程式無法使用此特性。

表格 3-1 (續) 選項

特性	描述
刪除動態成員	當您在已啟用執行階段提示之成員類型的變數中指定預設動態父項時，會刪除成員。 如果刪除商業規則中的動態成員，然後選取預設動態父項，則該父項的子成員會在應用程式啟動規則*之後*自動刪除。 **注意：Financial Consolidation and Close** 應用程式無法使用此特性。
啟用通知	啟用商業規則，以在啟動包含或不包含錯誤的規則時，傳送電子郵件通知至登入的使用者。

4. 在「規則設計工具」中的下列頁籤中，輸入或複查資訊：
   - **全域範圍**：定義商業規則元件中的常用維度。

     請參閱定義商業規則元件中的常用維度。
   - **變數**：複查並定義關於商業規則中使用之變數的資訊。

     > **備註：**
     > – **變數**頁籤只會在商業規則包含執行時期提示值時顯示。
     > – 若為非 Groovy 規則，*聚總儲存選項* (ASO) 類型的 Planning 立方體只支援 *member* 或 *members* 類型的變數。
     > – 當您從 Calculation Manager 啟動規則時，*不*支援下列選項：
     > * **已隱藏**
     > * **安全性**
     > * Planning 中定義的任何有效交集

   - **指令碼**：檢視針對規則產生的指令碼。

     您無法在此頁籤上進行變更。若要對指令碼進行變更，請在**設計工具**旁的下拉清單中選取**編輯指令碼**。
   - **使用情況**：檢視使用此規則的規則和規則集。

     您無法在此頁籤上編輯資訊。預設情況下，規則在建立時不會被任何規則或規則集使用。
   - **錯誤與警告**：按一下 ，以針對商業規則執行指令碼診斷。當您進行此項作業時，Calculation Manager 會分析商業規則指令碼並顯示下列任一項：
     – 驗證錯誤 (若規則驗證失敗的話)。

       若要修復驗證錯誤，請在錯誤上按一下右鍵，然後選取**在指令碼中顯示**或**在設計工具中顯示**。當您選取**在設計工具中顯示**時，Calculation Manager 會顯示帶有錯誤的元件，您可以進行必要的變更、儲存規則，並重新執行指令碼診斷。您必須在「設計工具」檢視中，才能夠編輯元件。
     – 摘要、警告和區塊 (若規則驗證成功的話)。

       在警告或區塊項目上按一下，然後選取**在指令碼中顯示**或**在設計工具中顯示**。您只能在「設計工具」檢視中編輯元件。

* **摘要**：計算中的資料儲存格數目、資料的通過次數以及不正確使用的維度數目等統計資料。
* **警告**：是否對儲存格參照指定所有稀疏維度、儲存格是否參照稀疏成員以及指派是否參照不同資料區塊中的稀疏維度成員等資訊。
* **區塊**：例如「對於每個 FIX 陳述式，可能受到 FIX 陳述式影響的潛在和預估區塊數為何？」這樣的資訊

5. 按一下 以儲存規則。

建立 Groovy 商業規則

Oracle 支援建立以 Groovy 指令碼語言所撰寫的商業規則。

另請參閱：

- 關於 Groovy 商業規則
  Groovy 商業規則可讓您設計複雜的規則，以解決一般商業規則無法解決的使用案例；例如，如果資料值高於預先定義的臨界值，則可設計防止使用者在表單上儲存資料的規則。
- 為 ASO 立方體建立 Groovy 商業規則
  您可以為 ASO 立方體建立 Groovy 商業規則。
- 為 BSO 立方體建立 Groovy 商業規則
  您可以為 BSO 立方體建立 Groovy 商業規則。
- 為 Groovy 商業規則或範本編輯指令碼
  您可以編輯 Groovy 規則或範本的指令碼。
- Groovy 規則適用的 Java API 參照
  如果是企業版應用程式，有 Java API 參照供您建立 Groovy 規則時使用。
- Groovy 商業規則範例
  我們提供可用的範例 Groovy 指令碼。
- Groovy 商業規則教學課程影片
  請觀賞下列教學課程影片，以瞭解實作及使用 Groovy 商業規則的詳細資料和最佳作法。
- Groovy 商業規則教學課程
  請完成下列教學課程，親自操作如何實作 Groovy 商業規則的範例。

關於 Groovy 商業規則

Groovy 商業規則可讓您設計複雜的規則，以解決一般商業規則無法解決的使用案例；例如，如果資料值高於預先定義的臨界值，則可設計防止使用者在表單上儲存資料的規則。

備註：

Groovy 是 Cloud EPM 平台提供的一種進階可自訂規則架構，隨附於 Enterprise Performance Management Enterprise Cloud Service 以及 Enterprise PBCS 和 PBCS Plus One。您可以在以下應用程式建立和編輯 Groovy 規則：

- Planning (包括以下應用程式類型：自訂、模組、任意格式、銷售規劃、策略性人力規劃 及現金預測)
- Enterprise Profitability and Cost Management
- Financial Consolidation and Close
- FreeForm
- 報稅

您在 Calculation Manager 中建立 Groovy 規則，然後在應用程式中任何您可以執行計算指令碼規則的位置執行這些規則；例如，在「規則」頁面上、在表單的相關內容中、在工作排程器中、在儀表板中、在任務清單中等等。

在規則集中也支援 Groovy 規則。您可以在規則集中將計算指令碼規則與 Groovy 規則組合在一起。

複合表單中不支援 Groovy 規則。

您可以從 Groovy 規則同步執行類型規則、規則集和範本的工作。

您可以撰寫 Groovy 指令碼，直接在 Oracle Enterprise Performance Management Cloud 中執行 select EPM Automate 命令，無須在用戶端機器上安裝 EPM Automate 用戶端。如需瞭解哪些 EPM Automate 命令可讓您藉由 Groovy 來執行，以及取得指令碼範例，請參閱*使用 EPM Automate* 中的在不安裝 EPM Automate 的情況下執行命令和支援的命令。

Oracle 支援兩種類型的 Groovy 規則：

- 這些規則可在執行時期根據執行時期提示以外的環境定義來動態產生計算指令碼，並在之後傳回針對 Oracle Essbase 執行的計算指令碼。

  例如，您可以僅針對專案的持續時間 (開始和結束日期) 建立計算專案費用的規則。

  另一個範例是趨勢型計算，這會將計算限制為表單上可用的科目。您可以將此計算用於「收入」、「費用」、「資產負債表」以及「現金流」中的不同表單。這可達成最佳化和重複使用。
- 純 Groovy 規則，例如若輸入的資料違反公司政策，可以執行資料驗證並取消作業。

影片

您的目標	觀看本影片
瞭解如何在 Oracle Fusion Cloud Enterprise Performance Management 中建立 Groovy 規則的訓練選項。	瞭解 Oracle Cloud EPM 中的 Groovy

為 ASO 立方體建立 Groovy 商業規則

您可以為 ASO 立方體建立 Groovy 商業規則。

1. 在**系統檢視**中，在**規則**按一下右鍵，然後選取**新增**，然後在**新增規則**對話方塊中輸入資訊。
2. 按一下**設計工具**旁的箭頭以變更至指令碼模式，然後選取**編輯指令碼**。

   如果您在*圖形*規則上選取**編輯指令碼**，便會開啟指令碼設計程式，且規則為空白。關閉後再重新開啟規則，規則則會以圖形方式開啟。如果您在「編輯指令碼」中儲存規則，規則將為空。
3. 輸入 Groovy 指令碼，然後儲存、驗證以及部署規則至您的應用程式。

> **備註：**
>
> ASO 立方體的 Groovy 商業規則會為所有變數顯示執行時間提示 (RTP)。

> **備註：**
>
> 在設計 Groovy 規則時，在 Groovy 指令碼頂端新增此參照標頭：
>
> `/*RTPS: */`
>
> 如果參照執行時期提示 (RPT)，請將執行時期提示新增至參照標頭中，如下所示：
>
> `/*RTPS: {rtp1}, {rtp2} */`
>
> 例如，如果您有名為 `rtpYear` 和 `rtpScenario` 的兩個變數，則指令碼頂端的參照標頭將會是
>
> `/* RTPS:{rtpYear}, {rtpScenario} */`
>
> 如需詳細資訊，請參閱類別 RtpValue。

為 BSO 立方體建立 Groovy 商業規則

您可以為 BSO 立方體建立 Groovy 商業規則。

1. 在**系統檢視**中，在**規則**按一下右鍵，然後選取**新增**，然後在**新增規則**對話方塊中輸入資訊。
2. 按一下**設計工具**旁的箭頭以變更至指令碼模式，然後選取**編輯指令碼**。
3. 在指令碼編輯器的右上角按一下**指令碼類型**旁的箭頭，然後選取 **Groovy 指令碼**。
4. 輸入 Groovy 指令碼，然後儲存、驗證以及部署規則至您的應用程式。

> **備註：**
>
> 在設計 Groovy 規則時，在 Groovy 指令碼頂端新增此參照標頭：
>
> `/*RTPS: */`
>
> 如果參照執行時期提示 (RPT)，請將執行時期提示新增至參照標頭中，如下所示：
>
> `/*RTPS: {rtp1}, {rtp2} */`
>
> 例如，如果您有名為 `rtpYear` 和 `rtpScenario` 的兩個變數，則指令碼頂端的參照標頭將會是
>
> `/* RTPS:{rtpYear}, {rtpScenario} */`
>
> 如需詳細資訊，請參閱類別 RtpValue。

為 Groovy 商業規則或範本編輯指令碼

您可以編輯 Groovy 規則或範本的指令碼。

若要為 Groovy *規則* 編輯指令碼，請執行下列動作：

1. 在**系統檢視**中，展開應用程式下的**規則**節點。
2. 在**規則**下，開啟所要的規則。
3. 在**規則編輯工具**中，在左邊算來的第三個下拉清單中，確認已選取**編輯指令碼**。
4. 編輯所要的指令碼，然後按一下**儲存**。

> **備註：**
>
> 請參閱在指令碼模式中編輯商業規則以取得詳細資訊。

若要為 Groovy *範本* 編輯指令碼，請執行下列動作：

1. 在**系統檢視**中，展開應用程式下的**範本**節點。
2. 在**範本**下，開啟所要的範本。
3. 在**範本編輯工具**中，在左邊算來的第三個下拉清單中，確認已選取**編輯指令碼**。
4. 編輯所要的指令碼，然後按一下**儲存**。

> **備註：**
>
> 請參閱為 Planning BSO 立方體建立 Groovy 範本，以取得如何建立 Groovy 範本的詳細資訊。

Groovy 規則適用的 Java API 參照

如果是企業版應用程式，有 Java API 參照供您建立 Groovy 規則時使用。

Java API 參照包含範例來示範 EPM Groovy 物件模型的語法和功能。

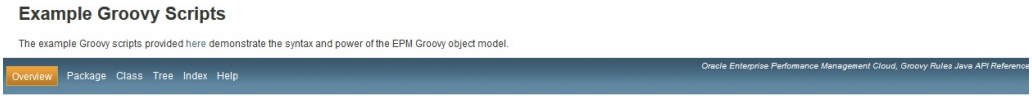
若要檢視 Java API 參照，請參閱雲端說明中心上的 Java API Reference for Oracle Enterprise Performance Management Cloud Groovy Rules。您也可以從 Oracle Enterprise Planning and Budgeting Cloud 學院存取此參照。若要存取學院，請登入，然後按一下**學院**。

Groovy 商業規則範例

我們提供可用的範例 Groovy 指令碼。

如要參閱範例 Groovy 指令碼，請執行下列動作：

1. 請參閱 Java API Reference for Oracle Enterprise Performance Management Cloud Groovy Rules，https://docs.oracle.com/cloud/latest/epm-common/GROOV/。
2. 請執行下列其中一項動作：
   - 在主要頁面的**範例 Groovy 指令碼**下，按一下「這裡」這個詞，以檢視範例指令碼：

   - 在左邊窗格的**所有類別**下，按一下類別以參閱該類別的範例。

     例如，若要查看「策略模型化」範例，請在左窗格中按一下 StrategicModel 類別。

Groovy 商業規則教學課程影片

請觀賞下列教學課程影片，以瞭解實作及使用 Groovy 商業規則的詳細資料和最佳作法。

您的目標	請觀賞這個影片
查看在 Oracle Fusion Cloud Enterprise Performance Management 中建立 Groovy 規則的訓練選項：	瞭解 Oracle Cloud EPM 中的 Groovy
使用 Groovy 規則，計算在「資料管理」中以累加的方式載入的資料。	使用 Groovy 規則，計算在「資料管理」中以累加的方式載入的資料
使用 Groovy 範本來改善使用者動作的可用性和計算效能。	使用 Groovy 範本來自訂動作以改善效能

Groovy 商業規則教學課程

請完成下列教學課程，親自操作如何實作 Groovy 商業規則的範例。

實作 Groovy：基本任務

您的目標	學習方式
瞭解 Groovy 指令碼語言，以及如何在 Oracle Fusion Cloud Enterprise Performance Management 中為商業程序建立 Groovy 指令碼。	Groovy 商業規則簡介
實作 Groovy 指令碼以使用資料方格及資料方格迭代器。 · 使用條件式邏輯來設定資料方格的背景顏色 · 實作效能基準測試 · 設定方格中的預設資料值	使用 Groovy 中的資料方格及迭代器
建立 Groovy 指令碼來處理商業程序中繼資料，本例中為將成員從一個父項移至另一個父項。指令碼將包含 RTP，以便提示使用者輸入資料。 您也會瞭解如何建立搭配功能表項目來呼叫指令碼的滑鼠右鍵點擊動作，以及如何建立動作功能表與資料表單之間的關聯。	利用 Groovy 來移動維度成員
使用 Groovy 規則和「智慧型發送」，將資料從來源位置發送到目標位置。	使用 Groovy 和智慧型發送來移動修改過的資料

實作 Groovy：進階任務

您的目標	學習方式
實作 Groovy 指令碼，以便根據儲存在動因立方體中的允許範圍來驗證資料輸入。	利用 Groovy 來驗證資料輸入規則
建立 Groovy 指令碼來處理商業程序中繼資料，本例中為新增維度成員。指令碼將包含 RTP，以便提示使用者輸入資料。 您也會瞭解如何建立搭配功能表項目來呼叫指令碼的滑鼠右鍵點擊動作，以及如何建立動作功能表與資料表單之間的關聯。	利用 Groovy 來新增維度成員
在商業程序中產生重點計算指令碼，用於只計算已經過編輯的資料，而不是整個資料輸入表單。	使用 Groovy 來計算修改過的資料

您的目標	學習方式
瞭解策略模型化概念，並瞭解如何使用 Groovy 指令碼整合策略模型與商業程序應用程式之間的資料。	使用 Groovy 來整合 Planning 與策略模型之間的資料
利用互動式的「設計時期提示」(DTP) 來建立 Groovy 範本。	建立 Groovy 範本
瞭解如何在商業程序中使用 Groovy 商業規則，從 Cloud EPM 匯出維度中繼資料，然後將中繼資料匯入商業程序。	使用 Groovy 規則，將來自 Enterprise Data Management Cloud 的維度中繼資料與 Planning 整合

實作 Groovy：REST API

您的目標	學習方式
呼叫資料管理 REST API 來執行資料載入規則，將使用者實體的最新產品數量載入商業程序。	使用 Groovy 來呼叫內部 REST API
從商業程序的 Groovy 指令碼呼叫外部 Rest API。 您也會瞭解如何建立搭配功能表項目來呼叫指令碼的滑鼠右鍵點擊動作，以及如何建立動作功能表與資料表單之間的關聯。	使用 Groovy 來呼叫外部 REST API

開啟商業規則

開啟 Calculation Manager 時，您可以從預設顯示的「系統檢視」開啟商業規則。

您也可以從「自訂檢視」、「篩選檢視」或「部署檢視」，開啟規則。

若要開啟商業規則，請執行下列其中一項動作：

- 用滑鼠右鍵按一下該規則，然後選取**開啟**。
- 請按兩下該規則。

備註：

在 Oracle Financials Cloud 中，規則若被另一位管理員鎖定，您將會看到以下訊息：「objectname 以唯讀模式開啟。此物件目前正由 FINUSER2 編輯」。如果發生此狀況，請完成以下步驟：

1. 按一下**確定**。
2. 對該規則或規則集按一下滑鼠右鍵，然後選取**解除鎖定**。便會顯示下列訊息：

   您要解除鎖定選取的項目嗎？目前編輯中使用者所做的任何變更都將遺失。
3. 按一下**確定**以開啟規則，或按一下**取消**以關閉規則而不開啟。

   如果規則被解除鎖定，正在編輯該規則的管理員並不會收到訊息。如果規則是由原始管理員編輯並儲存，之後解除鎖定該規則的管理員做了變更並儲存，則會保留最後儲存的內容，由原始管理員所做的編輯會被覆寫。

編輯商業規則

Related Topics

- 編輯商業規則
  您可以透過新增、刪除或變更元件 (包括公式、指令碼、條件、範圍和迴圈) 以編輯商業規則的結構，
- 在指令碼模式中編輯商業規則
  您可以編輯商業規則的指令碼。
- 在指令碼模式中進行編輯時的可用選項

編輯商業規則

您可以透過新增、刪除或變更元件 (包括公式、指令碼、條件、範圍和迴圈) 以編輯商業規則的結構，

您也可以編輯商業規則之元件的特性，以及商業規則本身的特性。

您可以編輯下列商業規則特性：

- 名稱與標題
- 描述與註解
- 維度、成員和變數

若要編輯商業規則，請執行下列動作：

1. 開啟規則。
2. 在「規則設計工具」中，新增元件，然後複製並刪除規則流程圖的現有元件。
   - 若要新增元件，請從**新物件**或**現有物件**拖曳物件，並將其拖曳至流程圖中。

     當您將現有公式元件或指令碼元件新增至流程圖中時，依照預設，公式或指令碼會成為共用物件。如果您不想共用，請在公式或指令碼元件的**特性**中，清除**共用**核取方塊。請參閱共用指令碼和公式元件。
   - 若要複製元件，請用滑鼠右鍵按一下元件，然後選取**複製**，然後將其貼到流程圖。

- 若要刪除元件，請用滑鼠右鍵按一下元件，然後選取**移除**。

3. 在**特性**中，編輯規則的特性。

   特性會隨著您在規則中新增元件，以及在流程圖的元件之間移動而變化。若要為特定元件輸入特性，請在流程圖中選取該元件。請參閱使用元件以設計商業規則和範本。)

4. 按一下 ，以儲存變更。

> **提示：**
>
> 在商業規則中編輯元件時，您可能希望增加或減少元件圖示的大小，以及流程圖中顯示的詳細資料量。若要這麼做，請使用 ，將流程圖放大和縮小。
>
> 如果流程圖的顯示小尺寸，就不會顯示元件標題，但您可以將滑鼠指標放在圖示上，以讀取其標題。無論流程圖中的元件大小如何，您都可以選取一個元件以檢視其特性。

在指令碼模式中編輯商業規則

您可以編輯商業規則的指令碼。

依預設，您可以在圖形模式中，使用「規則設計工具」建立商業規則，以設計規則的圖形流程圖。建立和儲存商業規則後，您可以在圖形模式或指令碼模式中進行編輯。如果選擇在指令碼模式中進行編輯，您可以稍後再回到圖形模式進行編輯。

若要在指令碼模式中編輯商業規則，請執行下列動作：

1. 開啟規則。
2. 在「規則設計工具」中，在**設計工具**旁的下拉清單中，選取**編輯指令碼**。

   若要切換回圖形模式，請在**編輯指令碼**旁的下拉清單中，選取**設計工具**。

   當您從圖形模式編輯切換成指令碼模式編輯時，如果商業規則包含共用元件 (例如公式和指令碼元件) 或範本，則此商業規則的指令碼在「指令碼編輯器」中只會包含對共用元件和範本的參照。共用元件和範本所產生的完整指令碼會顯示在下方窗格的**指令碼**頁籤上。

   請注意下列事項：

   - 在 ASO 規則中選取**編輯指令碼**時，指令碼會自動轉換成 Groovy。如果您之後儲存規則，指令碼會儲存成 Groovy，而且無法轉換回圖形。若要保留圖形格式，請選取**另存新檔**，然後以不同名稱將規則儲存成 Groovy 指令碼。
   - 在 BSO 規則中選取「編輯指令碼」時，圖形規則會轉換成以計算指令碼語法為基礎的指令碼模式。若要將指令碼轉換成 Groovy，請在「指令碼類型」選取 **Groovy 指令碼**。
   - 在指令碼模式中編輯時，請勿使用下列關鍵字：
     - `/*STARTCOMPONENT*/`
     - `/*STARTCOMPONENT:SCRIPT*/`
     - `/*STARTCOMPONENT:FORMULA*/`
     - `/*ENDCOMPONENT*/`
     - `/*STARTRULE*/`

- /*ENDRULE*/
- /*STARTTEMPLATE*/
- /*ENDTEMPLATE*/

3. 視需要編輯指令碼。

   請參閱在指令碼模式中進行編輯時的可用選項。

4. 按一下 。

複查下列額外資訊：

- 在指令碼模式中進行編輯時的可用選項
- 複查指令碼

在指令碼模式中進行編輯時的可用選項

Table 3-2　在指令碼模式中進行編輯時的可用選項

圖示	描述	另請參閱
	顯示或隱藏指令碼行號。預設會顯示行號。	
	插入函式及其參數 會為函式提供自動建議。當您鍵入函式的前幾個字元時，按下 CTRL+空白鍵以顯示建議。選取其中一個建議，並按 Enter 以完成函式。	使用函式
	插入跨維度成員	將成員和函式新增至元件
	插入從維度中選取的成員範圍	將成員和函式新增至元件
	插入變數	使用變數
	插入智慧型列示	使用智慧型列示
	編輯範本	使用系統範本
	註解 取消註解 顯示/隱藏註解	
	格式代碼 當您選取「格式化代碼」，指令碼行會分為一組並縮排，並移除空白行。	
	驗證語法	
	換行指令碼，讓捲出到頁面之外的長指令碼行以多行的方式顯示在同一頁上。	
	代碼完成開啟/代碼完成關閉 啟用或隱藏完成代碼的建議 例如，若要起始代碼完成，請在輸入 FIX 之後輸入 SHIFT + 空格鍵，來讓 Calculation Manager 為您輸入 () ENDFIX。	

Table 3-2 (Cont.) 在指令碼模式中進行編輯時的可用選項

圖示	描述	另請參閱
	尋找並取代指令碼中的文字字串	搜尋商業規則指令碼中的文字字串
	搜尋文字字串時，選取是否要**符合大小寫**或**符合完整文字**	搜尋商業規則指令碼中的文字字串

儲存商業規則

將商業規則儲存至為其建立的應用程式和應用程式類型。

另請參閱：

- 儲存商業規則
- 使用不同的名稱儲存商業規則

儲存商業規則

儲存商業規則時，會將商業規則儲存到您在建立規則時使用的應用程式和應用程式類型。在儲存之後，即可部署、驗證和啟動商業規則。您可以在 Calculation Manager 中部署和驗證規則，以及從規劃啟動規則。

若要在建立或編輯商業規則之後進行儲存，請按一下 。

> **備註：**
>
> 若要在儲存商業規則後於**系統檢視**中查看此商業規則，您可能需要重新整理應用程式清單。如果要執行這項作業，請用滑鼠右鍵按一下**系統檢視**中的任何節點或物件，然後選取**重新整理**。

使用不同的名稱儲存商業規則

您可以使用「另存新檔」，以不同的名稱儲存商業規則。也可以使用「另存新檔」，將規則從一個規則集複製到屬於同一規則集類型的另一規則集。「另存新檔」能夠以不同的名稱為原始商業規則建立副本，以區別於原始的商業規則。

若要使用不同的名稱儲存商業規則，請執行下列動作：

1. 按兩下該規則。
2. 在規則設計工具中，選取**動作**，然後選取**另存新檔**。
3. 在**另存新檔**中輸入規則的新名稱，然後選取**應用程式**。
4. 選取**計畫類型**。
5. 按一下**確定**。

   新規則已新增到**系統檢視**中的應用程式清單中。

執行商業規則

執行之前，您必須先開啟商業規則。

若要執行商業規則，請執行下列動作：

1. 按兩下規則，或用滑鼠右鍵按一下規則，然後選取**開啟**。
2. 當規則開啟時，請按一下 (啟動按鈕)。

複製商業規則

您可以將商業規則複製到另一個應用程式和計畫類型。

在將商業規則複製到另一個應用程式時，您可以使用其他名稱來建立新的商業規則。在商業規則中使用的任何變數也會複製到新的應用程式。

若要將商業規則複製到另一個應用程式和計畫類型，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下規則，然後選取**複製到**。
2. 在**複製到**中，輸入商業規則的新名稱或接受預設名稱，然後選取應用程式、應用程式類型和計畫類型。

   您無法將商業規則複製到多個應用程式和計畫類型。
3. 如果您將含共用元件的商業規則複製到其他應用程式，而且您要將共用元件複製到該應用程式，請選取**複製共用元件**。

   這會在複製的應用程式中建立共用元件，且複製的規則會使用來自其應用程式和計畫類型的共用元件。如果您不選取**複製共用元件**，則會在規則中複製共用元件，但複製的規則會使用來自原始規則之應用程式和計畫類型的共用元件。

   當您將含變數的商業規則複製到另一個應用程式時，會在複製目標的應用程式中建立變數。如果複製目標的應用程式中已有變數存在，則會在複製時將變數建立為規則層級變數。
4. 如果您複製共用元件，請選取**取代現有物件**來覆寫有相同名稱的物件。

在商業規則中搜尋

搜尋商業規則指令碼中的文字字串。搜尋並取代圖形商業規則中的文字。

另請參閱：

- 搜尋商業規則指令碼中的文字字串
- 搜尋並取代圖形商業規則中的文字

搜尋商業規則指令碼中的文字字串

若要搜尋商業規則指令碼中的文字字串：

1. 開啟包含指令碼的商業規則。
2. 在「規則設計工具」中，選取**指令碼**頁籤。
3. 輸入要搜尋的文字，然後按一下 。

搜尋並取代圖形商業規則中的文字

當您搜尋字串時，Calculation Manager 會從在流程圖中選取元件後的第一個元件開始搜尋，直到流程圖的結束，然後再從流程圖的開始，直到到達流程圖中選取的元件為止。

在找到文字字串的符合項目後，如果您要再搜尋另一個字串，則必須重新開始搜尋，以找到下一個符合項目。

若要搜尋和取代圖形商業規則中的文字，請執行下列動作：

1. 開啟商業規則。
2. 在規則設計工具中，用滑鼠右鍵按一下商業規則中的**開始**或是**結束**圖示，或者用滑鼠右鍵按一下商業規則流程圖中的元件，然後執行下列其中一項作業
   - 選取**尋找**以尋找文字字串的例項。
   - 選取**全部取代**，以尋找和取代文字字串的例項。

列印商業規則

您可以列印商業規則的特性、流程圖及元件詳細資料。

例如，如果要列印的商業規則中包含用於配置支出的公式元件，列印輸出會顯示公式語法、組成公式的函式和變數、規則流程圖中的步驟摘要 (非圖形格式)，以及規則的特性。

如果您僅列印指令碼中的商業規則：

- PDF 檔案中的「規則詳細資料」區段包含商業規則指令碼。
- PDF 檔案的「流程圖」區段僅包含「開始」和「結束」節點。
- PDF 檔案中的「流程摘要」區段是空白。

> **備註：**
>
> 您無法列印商業規則集或元件。如果您要列印的商業規則中使用元件，您可以列印那些元件。

若要列印商業規則，請執行下列動作：

1. 在**系統檢視**中，在商業規則按一下右鍵，然後選取**列印**。
2. 在**列印預覽**對話方塊中輸入資訊，然後按一下**產生 PDF**。

   商業規則的 PDF 檔案用 Adobe Acrobat 開啟。

刪除商業規則

您只能刪除並未被其他規則或規則集使用的商業規則。

如果其他規則或規則集正在使用該規則，則在刪除前必須先將此商業規則從正在使用它的規則和規則集中移除，或為正在使用它的規則和規則集建立該規則的副本。

若要刪除商業規則，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下規則，然後選取**顯示使用情況**，以確保其未由其他規則或規則集使用。
2. 如果規則未由其他規則或規則集使用，請用滑鼠右鍵按一下規則，然後選取**刪除**。
3. 按一下**確定**，確認刪除規則。

定義商業規則元件中的常用維度

藉由開啟業務規則，並選取每個維度常用的成員、變數和函式，來定義常用維度。

若要定義商業規則元件中的常用維度，請執行下列動作：

1. 開啟商業規則。
2. 在「規則設計工具」中，按一下流程圖中的**開始**或**結束**。
3. 在**全域範圍**頁籤中，選取維度，然後按一下 ，然後選取每個維度常用的成員、變數及函式。

   您為維度選取的值即為規則啟動時用來計算的值。

   如果您選取變數，請選取**動態連結變數**，將該變數動態連結到規則，如此只要在對變數進行變更時，變更便會在規則中更新。

> **備註：**
>
> 選取**排除方格值**以在商業規則中建立 "Exclude/EndExclude" 指令碼，而非 "Fix/EndFix" 指令碼。

分析並偵錯商業規則

藉由執行規則並擷取統計資訊以分析商業規則。藉由執行規則及檢查其指令碼以偵錯商業規則。

另請參閱：

- 分析商業規則
- 分析商業規則的指令碼
- 比較商業規則的指令碼
- 比較變更的商業規則和儲存的商業規則
- 偵錯商業規則
- 停用商業規則中的元件

分析商業規則

當您分析商業規則時，會執行商業規則並擷取統計資訊，例如元件執行多久、執行多少次，和元件執行前後的成員交集值。分析花費的時間取決於您的系統記憶體大小而定。

> **備註：**
>
> 「分析」功能不適用於「成員範圍」、「成員區塊」、「固定迴圈」或「條件」元件。

若要分析商業規則，請執行下列動作：

1. 開啟商業規則。
2. 在「規則設計工具」中，選取**動作**，然後選取**分析**。
3. 在**輸入分析準則**對話方塊中：
   - 輸入要在分析中顯示的數字項目，或選取**全選**以顯示項目。

     在此處輸入的數字是用於計算與顯示的成員交集數。例如，假設您在應用程式中有下列維度：

     - Year = 12 個成員
     - Measures = 50, 525 個成員
     - Product = 450 個成員
     - Market = 2000 個成員
     - Scenario = 4 個成員

     在此範例中，要計算的成員交集數為：2, 182, 680, 000 (12 x 50,525 x 450 x 2,000 x 4 或 Year x Measures x Product x Market x Scenario)
   - 針對每個維度，選取要分析的成員。

     以下選項可用於選取成員：

     - 按一下 ，以存取「成員選取器」對話方塊。
     - 選取維度，然後按一下 。
     - 鍵入成員。

     您只能在每個維度選取一個成員。
4. 按一下**確定**以開始分析。
5. 分析完成時，按一下商業規則流程圖中的元件，然後選取**分析資訊**頁籤。

> **提示：**
>
> 若要匯出分析資訊，請在**分析資訊**頁籤的方格中按一下右鍵，然後選取**匯出**。

> **備註：**
>
> 分析可能會因為 Oracle Essbase 最佳化而無法顯示正確的開始與結束值。如果分析無法顯示至少一個交集的開始和結束值，則可能是「計數」和經過時間不正確。

隱藏追蹤維度的成員

若要隱藏待追蹤維度的成員，請在**分析資訊**頁籤中，選取**隱藏待追蹤的維度**。如果您未選取此選項，將顯示針對維度處理的所有成員。

在**分析資訊**頁籤中顯示的前後值，表示已變更追蹤成員值的成員。前後值是您追蹤的成員的值。例如，如果您正在追蹤產品 200-30，則會看到產品 200 的行。產品 200 的行所顯示的值為 200-30 的值。

正在擷取統計資訊

若要在分析商業規則時擷取統計資訊，請記住下列要點：

- Oracle Essbase 的內部呼叫必須由成員區塊圍繞。
- 如果您只在「固定」陳述式中讓一個維度保留空白，且您不使用該維度的追蹤成員，則會針對成員區塊使用該維度或其中一個子代。
- 如果您只在「固定」陳述式中讓一個維度保留空白，且您指定該維度中為動態計算成員的追蹤成員，則成員區塊會是該維度或非動態計算或僅標籤成員的其中一個子代。
- **分析資訊**頁籤會顯示追蹤成員的值，以及成員區塊中使用的成員。

分析商業規則的指令碼

您可以執行圖形商業規則或指令碼商業規則，並分析其指令碼。

若要分析商業規則的指令碼，請執行下列動作：

1. 在**系統檢視**中，在商業規則上按一下右鍵，然後選取**開啟**。
2. 透過顯示的規則，選取**動作**，然後選取**分析指令碼**。

   此可執行商業規則。
3. 如果商業規則包含執行時間提示變數，請在**輸入 RTP 值**對話方塊中，輸入執行時間提示值，然後指定是否要將值套用至商業規則，然後按一下**確定**。
4. 在**指令碼分析**頁籤中，分析指令碼。

   如果指令碼中的行有分析資訊，會顯示在行旁邊。選取行，以在畫面右方的**分析資訊**區域中複查其分析資訊。如果沒有行的分析資訊，則會顯示「未提供分析資訊」。擁有最長執行時間的行，會標記為紅色。

   特性區域會顯示有關商業規則的資訊，例如其名稱、應用程式、計畫類型，以及分析指令碼所需的時間長度 (以毫秒計)。
5. 當您完成複查指令碼資訊時，請按一下 ，以關閉**指令碼分析**頁籤。

比較兩個商業規則的指令碼

您可能會想檢視兩個商業規則的指令碼，以查看兩者之間的不同、您所做的變更，或是查看圖形商業規則的指令碼。

> **備註：**
>
> 您一次只能比較兩個商業規則。此外，您無法比較商業規則的版本。例如，您無法比較商業規則的先前版本和您儲存的版本。

如要比較兩個商業規則的指令碼：

1. 在**系統檢視**中，選取兩個商業規則。
2. 用滑鼠右鍵按一下選取的商業規則，然後選取**比較指令碼**。

   在**比較指令碼**視窗中：

   - 如果任一指令碼有新增的內容，該行會以淡灰色背景顯示。
   - 如果任一指令碼有刪除的內容，該行會以暗灰色背景顯示。
   - 如果任一指令碼有變更，該行會以黃色背景顯示。
   - 第一欄顯示指令碼的行號。
   - 第二欄顯示三個字元的其中一個：
     - * 指出顯示於左欄指令碼的行，不同於顯示於右欄指令碼的行。
     - + 指出與顯示於左欄的指令碼相較，顯示於右欄的指令碼新增了一行。
     - – 指出與顯示於左欄的指令碼相較，顯示於右欄的指令碼刪除了一行
   - 第三和第四欄顯示兩個指令碼。
   - 底部視窗分成兩個窗格，您在上方窗格所選指令碼的完整行會顯示在其中。第一個窗格顯示左欄的指令碼，第二個窗格顯示右欄的指令碼。
3. 當您完成指令碼的比較之後，請按一下**確定**。

比較變更的商業規則和儲存的商業規則

您可檢視商業規則儲存版本和商業規則變更版本的指令碼，以查看這兩個版本之間有何不同、查看您所做的變更，或以並排顯示方式，查看儲存的和變更的圖形商業規則指令碼。

若要比較儲存的和變更的商業規則，請執行下列動作：

1. 開啟圖形或計算指令碼商業規則。
2. 選取**編輯**，然後選取**與儲存的版本比較**。

   在**比較指令碼**視窗中：

   - 如果商業規則或計算指令碼有新增的內容，該行會以淡灰色背景顯示。
   - 如果商業規則或計算指令碼有刪除的內容，該行會以深灰色背景顯示。
   - 如果商業規則或計算指令碼有變更，該行會以黃色背景顯示。
   - 第一欄顯示商業規則或計算指令碼的行號。
   - 第二欄顯示三個字元的其中一個：
     - * 指出顯示於左欄商業規則或計算指令碼的行，不同於顯示於右欄商業規則或計算指令碼的行。
     - + 指出與顯示於左欄的商業規則或計算指令碼相較，顯示於右欄的商業規則或計算指令碼新增了一行。

  - – 指出與顯示於左欄的商業規則或計算指令碼相較，顯示於右欄的商業規則或計算指令碼刪除了一行。
- 第三欄和第四欄顯示儲存的和變更的商業規則或計算指令碼。
- 底部視窗分成兩個窗格，您在上方窗格所選商業規則或計算指令碼的完整行會顯示在其中。第一個窗格於左欄顯示商業規則或計算指令碼的儲存版本，第二個窗格於右欄顯示商業規則或計算指令碼的修改版本。選取指令碼中的一列，使其顯示於下方窗格。儲存的版本位於上方，修改的版本位於下方。

3. 按一下**確定**。

偵錯商業規則

偵錯商業規則包括執行商業規則，並逐行檢驗其指令碼，以查看指令碼執行的方式。

您可以針對下列應用程式偵錯商業規則：

- Oracle Essbase 區塊儲存應用程式 (如果您具備寫入權限)
- 規劃應用程式，如果您具有啟動權限

> **備註：**
>
> 您無法在偵錯時編輯商業規則。

若要偵錯商業規則，請執行下列動作：

1. 開啟商業規則。
2. 在「規則設計工具」中，選取**動作**，然後選取**偵錯**。

   商業規則指令碼會顯示於指令碼偵錯程式。指令碼中的每個陳述式會顯示在個別行中。
3. 偵錯商業規則指令碼中的陳述式。

   當您偵錯陳述式時，您可以：

   - 在想要停止指令碼執行的地方插入和移除中斷點，以檢驗陳述式中成員的交集值。當執行在暫停點停止時，會顯示陳述式中所有成員的交集值。

     若要新增中斷點，請用滑鼠右鍵按一下中斷點右側，然後選取**新增中斷點**。您可新增暫停點至指派和條件式陳述式。您只能在顯示灰階中斷點的指令碼的那些行，新增中斷點。若要移除中斷點，請在其上按一下滑鼠右鍵，然後選取**移除中斷點**。
   - 新增條件至中斷點，讓陳述式的執行只在條件符合時停止。只有在包含中斷點之陳述式中使用的成員可用於條件中。

     若要新增條件至中斷點，請在中斷點上按一下滑鼠右鍵，然後選取**新增條件**。在**新增條件**對話方塊中，按一下 ，然後在**條件建置器**中輸入條件。

     若要編輯條件，請在其上按一下滑鼠右鍵，然後選取**編輯條件**。
   - 偵錯含中斷點的陳述式。

     若要偵錯含中斷點的陳述式，請用滑鼠右鍵按一下陳述式，然後選取**開始偵錯**。您正在偵錯的陳述式會醒目提示。陳述式成員、中斷點，以及執行前後的成員交集值會顯示在下列頁籤中：

     - **成員** - 顯示位於偵錯中斷點的目前成員交集。若要查看下一個成員交集，請按一下**繼續除錯**。

– **中斷點** - 顯示來自含中斷點的指令碼的表示式。**中斷點的值**頁籤會顯示表示式成員，並包含其在偵錯執行前後的值。

> **提示：**
>
> 若要繼續偵錯，請按一下 。您必須持續按 ，直到已偵錯所有成員交集為止。對所有成員交集進行偵錯後，您會收到「指令碼偵錯完成」的訊息。

停用商業規則中的元件

您可以停用商業規則中的元件。例如，您可能想將元件從驗證中排除，以分析哪一個元件會造成驗證錯誤。

若要停用商業規則中的元件，請執行下列動作：

1. 選取商業規則流程圖中的元件。
2. 在**特性**中，選取**停用**。

停用元件之後，將商業規則儲存至其所屬的地方，元件圖示會在商業規則流程圖中停用。您移動至已停用元件或者在已停用元件中建立任何元件，都會繼承已停用元件的「已停用」狀態，而且不會被視為商業規則的一部分計算。

> **備註：**
>
> 您無法停用「條件」元件。如果停用已共用的元件，則只會針對其已停用的商業規則停用。

重新整理商業規則或商業規則集

在「系統檢視」、「自訂檢視」和「部署檢視」中，您可以重新整理任何層級的應用程式清單。

您可以重新整理規劃應用程式類型、規劃應用程式、計畫類型、多個規則集或規則、或一個規則集或規則。

根據預設，當您重新整理應用程式清單中的任何節點時，所屬的所有規則，規則集、元件及範本也會被重新整理。但是，重新整理應用程式中的規則集或規則，並不會重新整理應用程式清單中的更高層級，或是屬於其他應用程式的規則集或規則。

例如，如果您重新整理規劃應用程式和計畫類型中的規則，將會重新整理該應用程式和計畫類型中的所有規則，但不會重新整理其他計畫類型或其他規劃應用程式中的規則。

若要重新整理商業規則或規則集，請用滑鼠右鍵按一下該規則或規則集，然後選取**重新整理**

> **備註：**
>
> 若要從 Planning 更新中繼資料變更，請在系統檢視中的應用程式節點按一下滑鼠右鍵，然後選取**重新整理**。您也可以用滑鼠右鍵按一下 Planning 應用程式類型、應用程式或是含有您要重新整理之商業規則的計畫類型，然後選取**重新整理**。

重新整理中繼資料變更

若要從 Planning 重新整理中繼資料變更 (立方體、維度、智慧型列示、UDA)：

1. 在系統檢視中，在應用程式節點或立方體節點按一下滑鼠右鍵。
2. 選取**重新整理**。

顯示商業規則或規則集的使用情況

顯示正在使用某商業規則或商業規則集的規則、範本和規則集。

當您想要刪除該規則或規則集，需要知道哪些物件正在使用它時，檢視規則或規則集的使用情況非常有用。

若要顯示商業規則或規則集的使用情況，請執行下列動作：

1. 用滑鼠右鍵按一下該規則或規則集，然後選取**顯示使用方法**。
2. 複查**使用方法**對話方塊中的資訊，然後按一下**確定**。

最佳化商業規則

利用可用工具並有效管理您的規則。

另請參閱：

- 商業規則最佳化的概觀
- 最佳化商業規則的步驟
- 在 Planning 應用程式活動報表中識別執行緩慢的商業規則
- 在 Calculation Manager 日誌訊息中識別執行緩慢的商業規則
- 使用日誌訊息以最佳化商業規則
- 範例商業規則

商業規則最佳化的概觀

您可以使用規劃活動報表來診斷商業規則的效能問題，查明並解決任何問題。活動報表可識別哪一個商業規則的執行時間最長。然後，您可以在 Calculation Manager 中開啟規則，複查日誌，並最佳化規則中的步驟以提高效能。

最佳化商業規則的步驟

若要最佳化商業規則，請執行下列動作：

1. 從 Planning 應用程式活動報表中，找出執行時間最長的商業規則 (請參閱*管理 Planning* 中的檢視活動報表)。
2. 將應用程式從生產服務例項複製到測試服務例項。
3. 對於每個執行緩慢的商業規則，請完成下列步驟：
   - 從 Calculation Manager 執行商業規則 (請參閱執行商業規則)。

- 使用 Calculation Manager 的日誌訊息，以識別規則中所花時間最長的步驟。
- 最佳化已識別的步驟，並再次執行商業規則，以確保執行時間更短，但仍能計算出正確的結果。

4. 最佳化商業規則後，變更將會移轉回生產服務例項。

在規劃應用程式活動報表中識別執行緩慢的商業規則

請注意報表中的商業規則/計算指令碼名稱。

Top 5 Worst Performing Calc Scripts Commands over 1 Min

Duration (Min:Sec)	Begin Time	End Time	Context	Calc Script Command
01:20	02:46:18	02:47:38	Calc Script: YTD_Calc Blocks Read: 3,937 Blocks Updated: 3,339 Blocks Created: 3,190	FIX ("BaseData", "Plan", "FY17", "FY16") .... See More
01:04	02:57:26	02:58:30	Calc Script: YTD_Calc Blocks Read: 3,321 Blocks Updated: 2,731 Blocks Created: 2,590	FIX ("BaseData", "Plan", "FY17", "FY16") .... See More

Top 10 Worst Performing Calc Scripts

Duration (Min:Sec)	Begin Time	End Time	Application	Calc Script
01:20	02:46:18	02:47:38	Vision	YTD_Calc
01:04	02:57:26	02:58:30	Vision	YTD_Calc

Top 10 Worst Performing Business Rules over 30 Seconds

Duration (Min:Sec)	Begin Time	End Time	User	Business Rule	Run Time Prompts
01:20	02:46:18	02:47:38	epm_default_cloud_admin	YTD_Calc	
01:04	02:57:26	02:58:30	epm_default_cloud_admin	YTD_Calc	

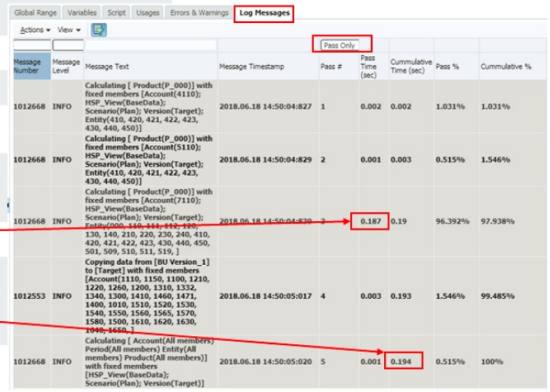
備註：

請參閱 *管理 Planning* 的〈檢視活動報表〉，以瞭解規劃活動報表的詳細資訊。

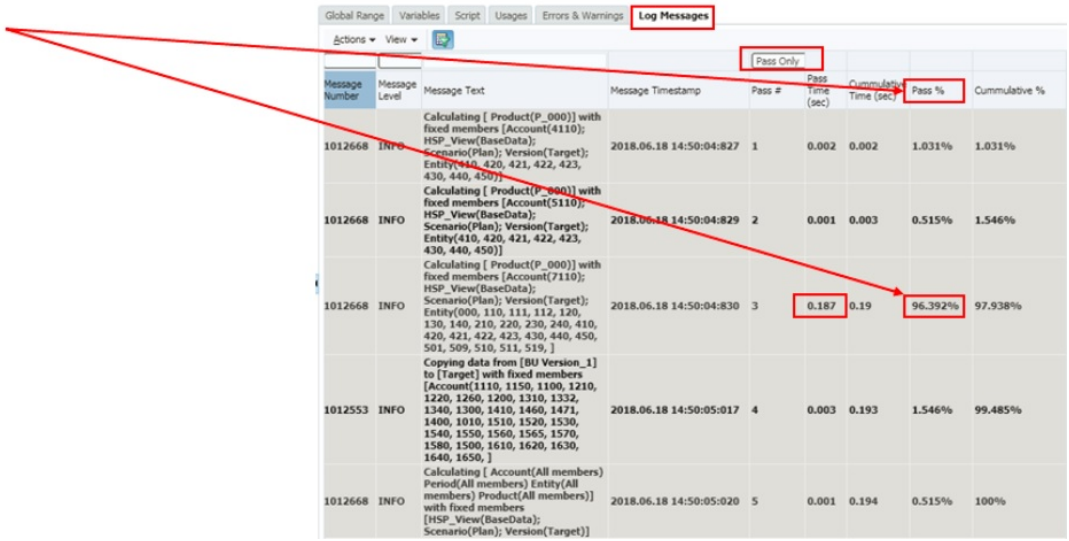
在 Calculation Manager 日誌訊息中識別執行緩慢的商業規則

- 執行商業規則後，會顯示新的「日誌訊息」頁籤：

Message Number	Message ID of the log entry
Message Level	Level of the message (e.g., INFO)
Message Text	Complete text of the log entry
Message Timestamp	Timestamp when the log entry was generated
Pass #	Number of the current pass. The highest number in this column is the total number of passes in the business rule.
Pass Time (sec)	Execution time, in seconds, of the current pass only
Cumulative Time (sec)	Time in, in seconds, since the start of the rule ~ (i.e., previous Cumulative Time + current Pass Time) The last entry in this column is the total business rule execution time
Pass %	Pass time as a percentage of the total business rule elapsed time
Cumulative %	Percentage of the total time of all passes in the business rule to that line When all passes are complete, this should show 100%.



- 使用此資訊，您可以查看哪個執行 (或多個執行) 佔計算時間總計的最高百分比。



使用日誌訊息以最佳化商業規則

- 「日誌訊息」頁籤可幫助商業規則實施者確定總商業規則執行時間在計算執行過程中的分佈情況。
- 新執行的開始以粗線顯示在日誌訊息中。
- 檢視**執行次數**和**執行百分比**欄，以 確定哪些執行耗用了商業規則執行時間總計的大部分。
- 佔總時間百分比最大的區段是最佳化的候選項目。
- 變更完成後，重新執行商業規則並比較日誌訊息。

> **備註：**
>
> 日誌訊息限制為 20,000 則。超過 20,000 則之後的其他訊息一律略過。

範例商業規則

- 在此範例中，我們已將五個年累計成員新增至 *Account* 維度："4110_UTD"、"4120_YTD"、"4130_YTD"、"4140_YTD"、"4150_YTD"。然後建立商業規則：

Pass 1

```
SET UPDATECALC OFF;
FIX ("BaseData","Plan","FY17", "FY16")
"BU Version_1"(
IF(@ismbr("Jan"))
"4110_YTD" = "4110"; "4120_YTD" = "4120"; "4130_YTD" = "4130"; "4140_YTD" = "4140"; "4150_YTD" = "4150";
Else
"4110_YTD"="4110" + @prior("4110_YTD"); "4120_YTD"="4120" + @prior("4120_YTD");
"4130_YTD"="4130" + @prior("4130_YTD"); "4140_YTD"="4140" + @prior("4140_YTD");
"4150_YTD"="4150" + @prior("4150_YTD");
Endif)
Agg("Entity","Product");
ENDFIX
```

Pass 2

商業規則計算兩個 YTD 科目，然後透過 *Product* 和 *Entity* 維度聚總值。

- 查看已選取之*僅執行*的日誌訊息，我們可以看到大部分規則時間是在第一次執行時獲得的。

Variables | Script | Usages | Errors & Warnings | **Log Messages**

Actions ▾ View ▾

Message Number	Message Level	Message Text	Message Timestamp	Pass # (Pass Only)	Pass Time (sec)	Cummulative Time (sec)	Pass %	Cummulative %
1012668	INFO	Calculating [Version(BU Version_1)] with fixed members [HSP_View(BaseData); Year(FY16, FY17); Scenario(Plan)]	2018.07.04 09:46:19:293	1	79.235	79.235	99.995%	99.995%
1012670	INFO	Aggregating [Entity(All members) Product(All members)] with fixed members [HSP_View(BaseData); Year(FY16, FY17);	2018.07.04 09:47:38:528	2	0.004	79.239	0.005%	100%

- 取消選取*僅執行*，可讓我們查看日誌的 pass1 區段中的區塊讀取/寫入次數。

Message Text	Message Timestamp	Pass #
Calculator Information Message: Total Block Created: [3.1900e+03] Blocks Sparse Calculations: [3.2000e+03] Writes and [3.2100e+03] Reads Dense Calculations: [0.0000e+00] Writes and [0.0000e+00] Reads Sparse Calculations: [0.0000e+00] Cells Dense Calculations: [0.0000e+00] Cells	2018.06.29 10:47:34:...	1

- 檢閱語法，我們可以看到在 *Entity* 和 *Product* 上的計算並沒有 Fix；因此，所有規則都在 *Entity* 與 *Product* 的所有層級上執行。不需要計算 pass1 的上層，因為無論如何這些都會被第二次執行的 AGG 陳述式覆寫。

```
SET UPDATECALC OFF;
FIX ("BaseData","Plan","FY17", "FY16")
Pass 1
"BU Version_1"(
IF(@ismbr("Jan"))
"4110_YTD" = "4110"; "4120_YTD" = "4120"; "4130_YTD" = "4130"; "4140_YTD" = "4140"; "4150_YTD" = "4150";
Else
"4110_YTD"="4110" + @prior("4110_YTD"); "4120_YTD"="4120" + @prior("4120_YTD");
"4130_YTD"="4130" + @prior("4130_YTD"); "4140_YTD"="4140" + @prior("4140_YTD");
"4150_YTD"="4150" + @prior("4150_YTD");
Endif)
Pass 2
Agg("Entity","Product");
ENDFIX
```

- 複查規則，並增加額外的 Fix 陳述式，以將 *Entity* 與 *Product* 的 lev0 新增至 Pass1。

```
SET UPDATECALC OFF;
FIX ("BaseData","Plan","FY17")
Pass 1
Fix(@LEVMBRS("Entity",0), @LEVMBRS("Product",0))
"BU Version_1"(
IF(@ismbr("Jan"))
"4110_YTD" = "4110";
"4120_YTD" = "4120";
Else
"4110_YTD"="4110" + @prior("4110_YTD");
"4120_YTD"="4120" + @prior("4120_YTD");
Endif)
ENDFIX
Pass 2
Agg("Entity","Product");
ENDFIX
```

- 然後再次執行規則。
- 修改之後，Pass1 仍需要最長的時間，但因為不再需要計算上層區塊，現在時間加快 20%。整體計算時間也提高 20%。

Variables | Script | Usages | Errors & Warnings | **Log Messages**

Actions ▾ View ▾

Message Number	Message Level	Message Text	Message Timestamp	Pass #	Pass Time (sec)	Cummulative Time (sec)	Pass %	Cummulative %
1012668	INFO	Calculating [Version(BU Version_1)] with fixed members [HSP_View(BaseData); Year(FY16, FY17); Scenario(Plan); Entity(000, 110, 111, 112, 120, 130, 140, 210, 220, 230, 240, 410, 420, 421, 422, 423, 430, 440, 450, 501, 509, 510, 511, 519, 520, 530, 535,]	2018.07.04 09:57:27:086	1	63.332	63.332	99.998%	99.998%
1012670	INFO	Aggregating [Entity(All members) Product(All members)] with fixed members [HSP_View(BaseData); Year(FY16, FY17); Scenario(Plan)]	2018.07.04 09:58:30:418	2	0.001	63.333	0.002%	100%

- 取消選取*僅執行*，我們可以看到日誌的 Pass1 區段中的區塊、讀取、寫入次數在變更後下降。

BEFORE rule change

Message Text	Message Timestamp	Pass #
Calculator Information Message: Total Block Created: [3.1900e+03] Blocks Sparse Calculations: [3.2000e+03] Writes and [3.2100e+03] Reads Dense Calculations: [0.0000e+00] Writes and [0.0000e+00] Reads Sparse Calculations: [0.0000e+00] Cells Dense Calculations: [0.0000e+00] Cells	2018.06.29 10:47:34:...	1

AFTER rule change

Message Text	Message Timestamp	Pass #
Calculator Information Message: Total Block Created: [2.5900e+03] Blocks Sparse Calculations: [2.5920e+03] Writes and [2.5940e+03] Reads Dense Calculations: [0.0000e+00] Writes and [0.0000e+00] Reads Sparse Calculations: [0.0000e+00] Cells Dense Calculations: [0.0000e+00] Cells	2018.06.29 10:38:22:...	1

- 正如我們在活動報表中所見，這樣可以減少對整個業務規則 (兩次執行) 的讀取與寫入次數。

BEFORE rule change

Top 5 Worst Performing Calc Scripts Commands over 1 Min

Duration (Min:Sec)	Begin Time	End Time	Context	Calc Script Command
01:20	02:46:18	02:47:38	Calc Script: YTD_Calc Blocks Read: 3,937 Blocks Updated: 3,339 Blocks Created: 3,190	FIX ("BaseData", "Plan", "FY17", "FY16") ... See More
01:04	02:57:26	02:58:30	Calc Script: YTD_Calc Blocks Read: 3,321 Blocks Updated: 2,731 Blocks Created: 2,590	FIX ("BaseData", "Plan", "FY17", "FY16") ... See More

AFTER rule change

4

設計商業規則集

另請參閱：

- 關於商業規則集
  您可以組合能夠同時或依序啟動的商業規則 (或商業規則集)，以建立商業規則集。
- 建立商業規則集
  您也可以從「系統」、「篩選」、「自訂」和「部署」檢視，以及在「規則集」和「範本設計工具」中建立商業規則集。
- 開啟商業規則集
  您可以從「系統」、「篩選條件」、「部署」和「自訂檢視」中開啟商業規則集。
- 開啟商業規則集中的商業規則
  您可以從商業規則集開啟商業規則。
- 將商業規則新增至商業規則集
  當您將商業規則新增至商業規則集，則可依次或同時啟動規則集中的規則。
- 從商業規則集中移除商業規則
  在從商業規則集中移除商業規則時，規則並未真正刪除。規則存在於資料庫中，與規則集彼此獨立。
- 將商業規則集複製到另一個應用程式
  當您將商業規則集複製到其他應用程式時，請確保將其複製到相同的應用程式類型。
- 儲存商業規則集
  儲存商業規則集時，會將商業規則集儲存到您在建立規則集時使用的應用程式及應用程式類型。
- 刪除商業規則集
  只有在其他商業規則集未使用該商業規則集時，您才能將它刪除。

關於商業規則集

您可以組合能夠同時或依序啟動的商業規則 (或商業規則集)，以建立商業規則集。

您建立規則集的能力，由指派給您的角色決定。請參閱*使用 Oracle Planning and Budgeting Cloud*。

您建立和儲存規則集之後，您要在 Calculation Manager 中進行驗證和部署。然後，您可以從應用程式啟動之。

> **提示：**
>
> 當您建立商業規則集時，您可能希望保留您正使用開啟的商業規則、元件、範本以及變數。Calculation Manager 會在標籤式介面中顯示這些物件，以便您可以在建立商業規則集時，輕鬆地在這些標籤之間移動。您可以在 Calculation Manager 內擁有開啟多達十個標籤，但是 Oracle 建議您不要同時開啟超過十個以上的物件，以達到最佳效能。

建立商業規則集

您也可以從「系統」、「篩選」、「自訂」和「部署」檢視，以及在「規則集」和「範本設計工具」中建立商業規則集。

若要建立商業規則集，請執行下列動作：

1. 在**規則集**按一下右鍵，然後選取**新增**。
2. 在**新增規則集**對話方塊中，輸入名稱，並選取應用程式類型與應用程式，然後按一下**確定**。

   如果您從「系統檢視」新增規則集，會自動填入應用程式類型和應用程式。
3. 從**現有物件**中，將現有規則和規則集拖曳至流程圖。

   規劃規則集可以包含建立在以及部署至不同應用程式中的規則和規則集。
4. 執行下列任一項任務：
   - 若要變更規則集中的規則順序，請按一下向上或向下箭頭，或在規則上按一下右鍵，然後選取**上移**或**下移**。
   - 若要從規則集移除規則，請用滑鼠右鍵按一下規則，然後選取**移除**。
   - 若要開啟商業規則以進行編輯，請用滑鼠右鍵按一下規則，然後選取**開啟**。
5. 在**特性**中，輸入規則集的特性。

   在「規則集設計工具」中，如果選取正在建立之規則集中的規則，則**特性**上會顯示其特性，而非規則集的特性。

   如果想要規則集中的規則和規則集同時啟動，請選取**啟用平行執行**。預設情況下，屬於同一規則集的規則和規則集會依序啟動 - 規則集中的每個規則或規則集都必須正確無誤地執行，下一個規則或規則集才會啟動。

   如果規則集包含巢狀規則集，且巢狀規則集的**啟用平行執行**設定與父項規則集的設定不同，則會套用巢狀規則集的設定。例如，如果 ruleset1 (旗標標示為*平行*處理) 中包含 rule1、rule2 和 ruleset2 (標示為*依序*處理)，則即使 ruleset1 標示為平行處理，ruleset2 中的規則和規則集仍然為依序處理。
6. 在**使用情況**頁籤上，複查哪些規則集正在使用此規則集 (如果有)。

   根據預設，規則集在建立時，不會被任何其他規則集使用，因此，當您一開始建立商業規則集時，此頁籤為空白。

   如果您在規則集中選取規則，可以查看商業規則集的名稱。
7. 在**變數**頁籤上，複查並輸入關於規則集中所使用之變數的資訊。

   選取**合併變數**，以合併此規則集之規則中所使用*相同變數*的所有例項，因此，啟用規則時，只會顯示每個變數的第一個例項。如果未合併變數，則會顯示各變數的所有例項。

   如果選取**合併變數**，使用者為執行時期提示輸入的第一個值，會用於驗證和啟動期間往後每一次出現的執行時期提示。
8. 按一下 。

開啟商業規則集

您可以從「系統」、「篩選條件」、「部署」和「自訂檢視」中開啟商業規則集。

若要開啟商業規則集，請執行下列其中一項動作：

- 用滑鼠右鍵按一下該規則集，然後選取**開啟**。
- 請按兩下該規則集。

開啟商業規則集中的商業規則

您可以從商業規則集開啟商業規則。

- 在**系統檢視**中，展開**規則集**，展開想要的規則集，然後按兩下想要的規則。
- 在開啟要編輯的商業規則集後，在規則集中按兩下想要的規則。

將商業規則新增至商業規則集

當您將商業規則新增至商業規則集，則可依次或同時啟動規則集中的規則。

若要將商業規則新增至商業規則集，請執行下列動作：

1. 開啟規則集。
2. 從**現有物件**中，將現有規則拖曳至規則集。
3. 按一下 。

從商業規則集中移除商業規則

在從商業規則集中移除商業規則時，規則並未真正刪除。規則存在於資料庫中，與規則集彼此獨立。

若要從商業規則集中移除商業規則，請執行下列動作：

1. 開啟該規則集。
2. 用滑鼠右鍵按一下想要的規則，然後選取**移除**。

將商業規則集複製到另一個應用程式

當您將商業規則集複製到其他應用程式時，請確保將其複製到相同的應用程式類型。

若要將商業規則集複製到另一個應用程式，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下商業規則集，然後選取**複製到**。
2. 在**複製到**中，輸入商業規則集的名稱，或接受預設名稱，然後選取應用程式類型和應用程式。

   您無法將商業規則集複製到多個應用程式類型和應用程式。
3. 選取是否複製任何共用元件，以及是否取代具有相同名稱的現有元件。

   您要複製到新應用程式的商業規則集中必須要有共用元件，才能啟用這些選項。
4. 按一下**確定**。

   新商業規則集新增到您選取的應用程式和應用程式類型。若要在「系統檢視」中顯示該商業規則集，可能必須重新整理應用程式清單。若要重新整理應用程式清單，請用滑鼠右鍵按一下應用程式，然後選取**重新整理**。請參閱重新整理商業規則或商業規則集。

儲存商業規則集

儲存商業規則集時，會將商業規則集儲存到您在建立規則集時使用的應用程式及應用程式類型。

在儲存之後，即可部署、驗證及啟動規則集

您可以在 Calculation Manager 中部署和驗證商業規則集，以及從規劃啟動規則集。

若要在建立或編輯商業規則集之後進行儲存，請按一下 。

> **備註：**
>
> 若要在儲存後於**系統檢視**中查看此商業規則集，您可能需要重新整理應用程式清單。如果要執行這項作業，請用滑鼠右鍵按一下「系統檢視」中的任何節點或物件，然後選取**重新整理**。

刪除商業規則集

只有在其他商業規則集未使用該商業規則集時，您才能將它刪除。

如果其他商業規則集正在使用規則集，您必須先將該商業規則集從正在使用它的商業規則集中移除，或是為正在使用它的商業規則集建立該商業規則集的副本，然後才能進行刪除。

若要刪除商業規則集，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下規則集，然後選取**顯示使用情況**，以確保其未由其他規則集使用。
2. 如果規則集未由其他規則集使用，請用滑鼠右鍵按一下規則集，然後選取**刪除**。
3. 按一下**確定**以確認刪除規則集。

5

使用系統範本

另請參閱：

- 關於系統範本
  系統範本是執行計算的預先定義範本。
- 顯示系統範本
  此處顯示的系統範本取決於它們是*圖形*範本或*規則*範本而定。
- 使用系統範本
  若要使用系統範本，請選取範本，並將其拖曳至流程圖或指令碼。
- 顯示範本流程
  當您複查系統範本所產生的指令碼時，這有時對於查看範本流程很有用。
- 將系統範本另存為自訂範本
  您可能想自訂系統範本的內容。
- 將系統範本從商業規則中移除
  您可以將系統範本從商業規則中移除。

關於系統範本

系統範本是執行計算的預先定義範本。

系統範本執行的計算包含清除資料；複製資料；計算金額、單位和費率；分布資料；配置值；聚總資料；輸入指令碼命令以最佳化效能；並轉換科目值以報告幣別。

您可以包含系統範本：

- 圖形或指令碼商業規則
- 圖形或指令碼自訂範本

如同商業規則的元件或自訂範本，系統範本包含您輸入參數的一系列步驟。這些參數與範本邏輯搭配使用，將在商業規則或範本內產生計算指令碼。這可讓範本更容易使用，並減少潛在的計算指令碼語法錯誤。

在 Calculation Manager 中，所有系統範本都是作為精靈來提供。在精靈中使用系統範本能更輕鬆地使用範本，並減少潛在計算指令碼語法錯誤。

系統範本中的提示是依據您在精靈所作之選擇來進行篩選的。例如，「聚總」系統範本可顯示選取密集維度至聚總這一步驟，但如果沒有可用於聚總的密集維度，則不會針對完整和部分密集維度顯示該步驟。

> **備註：**
>
> 您可以在規劃區塊儲存應用程式中使用系統範本。

顯示系統範本

此處顯示的系統範本取決於它們是*圖形*範本或*規則*範本而定。

當您建立或是開啟*圖形* 規則或範本，會以下列方式顯示系統範本：

- **新物件**—會在**新物件**中將系統範本顯示成個別物件。
- **現有物件**—會在**現有物件**的**系統範本**標頭下顯示系統範本。

當您建立或是開啟*指令碼* 規則或範本時，只會在**現有物件**中顯示系統範本。

若要存取系統範本的精靈，請執行下列動作：

- 對於*圖形* 規則或範本，請選取想要的範本並將它拖曳至**開始**與**結束**之間的流程圖。
- 對於*指令碼* 規則或是範本，請選取範本並將它拖曳至指令碼。

使用系統範本

若要使用系統範本，請選取範本，並將其拖曳至流程圖或指令碼。

另請參閱：

- 使用清除資料範本
  使用「清除資料」範本清除立方體中的成員資料。
- 使用複製資料範本
  使用「複製資料」範本，可將資料從資料庫中某個位置複製到另一個位置。
- 使用金額-單位-費率範本
  使用「金額-單位-利率」範本，以計算您針對金額、單位和利率所輸入的三個成員之一。
- 使用配置-層級至層級範本
  使用「配置 - 層級間」範本，在資料庫大綱中從某個層級配置到其他層級。
- 使用配置範本
  以平均的方式或是根據指定的動因，使用「配置」範本將值從來源配置到目的地。
- 使用聚總範本
  使用「聚總」範本可聚總您所指定的成員資料值。
- 使用 SET 命令範本
  您可使用「SET 命令」範本，輸入將計算指令碼效能最佳化的指令碼命令。
- 使用幣別轉換範本
  「幣別轉換」範本使用系統類型科目或您選取的科目，將科目值轉換為報表幣別。

使用清除資料範本

使用「清除資料」範本清除立方體中的成員資料。

若要清除資料，必須指定您要清除其值的成員。您可以為一個成員或一個區塊的成員清除資料。

例如，您可以先使用「清除資料」範本來清除預測資料，再將資料從 "actual" 複製到 "forecast" 以進行變更。

若要使用「清除資料」範本，請執行下列動作：

1. 建立或開啟商業規則或是範本。

2. 請執行下列其中一項動作：
   - 對於*圖形*規則或範本，在**新物件**之下選取**清除資料**，然後將它放入**開始**與**結束**之間的流程圖中。
   - 對於*指令碼*規則或範本，請展開**系統範本**，選取**清除資料**，然後將它放入指令碼中想要的區域。
3. 在**清除資料精靈**中輸入要求的資訊。

表格 5-1　清除資料精靈步驟

步驟	解釋
資料選擇	為每個維度選取一或多個成員，以定義要清除的資料。 下列選項為可用選項： · 按一下**使用預先定義的選擇**旁邊的 以選取變數，從而將所有維度填入值。 · 按一下**成員選取器**旁邊的 ，以選取每個維度的成員與函式。 · 選取維度，然後按一下 ，以選取變數、成員或函式。 在**選取「清除區塊」選項**中，選取選項以定義如何清除資料： · **全部** — 清除所有區塊的資料 · **上層** — 僅清除整合的資料區塊。 · **動態** — 清除含有衍生自「動態計算」和「儲存」成員組合之值的區塊 · **非輸入** — 清除含有衍生值的資料區塊。此功能僅適用於完全由計算作業建立的區塊，而不是有任何值載入到其中的區塊。 · **不適用** — 清除資料區塊並將成員設定成 #Missing
設定值	如果您具有將 `HSP_Rates` 設為 "sparse" 的多重幣別應用程式或是單一幣別應用程式，**設定值**會顯示在**資料選擇**中選取的成員。 按一下下拉清單，然後選取在其中選取了單一成員的維度，最好為密集維度。 **備註：**因為指令碼使用您從維度中選取的成員，如果您不能選取在其中選取了單一成員的維度，您將無法使用此範本。

使用複製資料範本

使用「複製資料」範本，可將資料從資料庫中某個位置複製到另一個位置。

例如，您可能想用「複製資料」範本，以複製 2014 年科目值和實體值的方式來建立 2015 年預算。在這種情況下，您會將 2014 年的實際資料複製到 2015 年的預測資料。您也可以使用「複製資料」範本將 "worst case " 預算案例中的預算資料複製到 "best case" 案例，以進行變更。

若要使用「複製資料」範本，請執行下列動作：

1. 建立或開啟商業規則或是範本。
2. 請執行下列其中一項動作：
   - 對於*圖形*規則或範本，在**新物件**之下選取**複製資料**，然後將它放入**開始**與**結束**之間的流程圖中。
   - 對於*指令碼*規則或範本，請展開**系統範本**，選取**複製資料**，然後將它放入指令碼中想要的區域。
3. 在**複製資料精靈**中輸入要求的資訊。

表格 5-2　複製資料精靈步驟

步驟	解釋
資訊	描述範本的功能。
檢視點	為每個維度選取一或多個成員，以定義要複製的資料。將複製時會發生變化的維度保留空白。您將在下列步驟中指定這些成員 下列選項為可用選項： · 按一下**使用預先定義的選擇**旁邊的 以選取變數，從而將所有維度填入值。 · 按一下**成員選取器**旁邊的 ，以選取每個維度的成員與函式。 · 選取維度，然後按一下 ，以選取變數、成員或函式。
複製來源	選取複製資料來源的成員。 **備註：**至少為每個維度選取一個成員。如果您沒有為每個維度選取一個成員，將會顯示警告訊息。
複製到	選取要複製資料的成員。
選項	**選擇性：**輸入要套用至資料儲存格目的範圍的增加或減少百分比。 例如，若要增加目標資料 10%，請輸入 **10**，若要減少目標資料 10%，請輸入 **-10**。 若要輸入變數，按一下 。例如，您可以定義執行時期的提示變數，然後輸入在執行時期的此增加值或減少值。

使用金額-單位-利率範本

使用「金額-單位-利率」範本，以計算您針對金額、單位和利率所輸入的三個成員之一。

例如，如果您想要計算產品收入，而且您知道數量和價格，則可以選取**計算金額**，並為金額輸入產品收入，為單位輸入數量以及為費率輸入價格。這會將數量與價格相乘以計算產品收入。

若要使用「金額-單位-利率」範本，請執行下列動作：

1. 建立或開啟商業規則或是範本。
2. 請執行下列其中一項動作：
   - 對於*圖形*規則或範本，在**新物件**之下選取**金額-單位-利率**，然後將它放入**開始**與**結束**之間的流程圖中。
   - 對於*指令碼*規則或範本，請展開**系統範本**，選取**金額-單位-利率**，然後將它放入指令碼中想要的區域。
3. 在**金額-單位-利率**中輸入要求的資訊。

表格 5-3　金額-單位-利率精靈步驟

步驟	解釋
資訊	描述範本的功能。
選取計算方法	從下拉清單中選取下列其中一個選項，以定義要使用的計算方法： · **計算金額**—金額 = 單位 * 費率 · **計算單位**—單位 = 金額 / 費率 · **計算費率**—費率 = 金額 / 單位 · **計算全部**—如果具有 `#Missing` 的成員可能改變或不明，則選取此選項。此計算方法會搜尋 `#Missing` 值，並套用適當的計算方法以解決這些不明值。

表格 5-3 (續) 金額-單位-利率精靈步驟

步驟	解釋
檢視點	定義在配置程序中不會發生變化的成員。 以下是選取成員的可用選項。 • 按一下**使用預先定義的選擇**旁邊的 以選取變數，從而將所有維度填入值。 • 按一下**成員選取器**旁邊的 ，以選取每個維度的成員。 • 選取維度，然後按一下 ，以選取變數、成員或函式。
計算金額/ 計算單位 / 計算費率 / 計算全部	此步驟視您為計算方法所選取的項目而異。 • 如果您選取**計算金額**，請輸入成員或是按一下 ，選取「金額」的成員，按「下一步」以輸入「單位」的成員，然後按「下一步」以輸入「費率」的成員。在「單位」與「費率」步驟中，您可以輸入與「檢視點」不同的成員。 • 如果您選取**計算單位**，請輸入成員或是按一下 ，選取「單位」的成員，按「下一步」以輸入「金額」的成員，然後按「下一步」以輸入「費率」的成員。在「單位」與「費率」步驟中，您可以輸入與「檢視點」不同的成員。 • 如果您選取**計算費率**，請輸入成員或是按一下 ，選取「費率」的成員，按「下一步」以輸入「金額」的成員，然後按「下一步」以輸入「單位」的成員。在「金額」與「單位」步驟中，您可以輸入與「檢視點」不同的成員。 • 如果您選取**計算全部**，請輸入「金額」、「單位」和「費率」的成員。

使用配置 - 層級間範本

使用「配置 - 層級間」範本，在資料庫大綱中從某個層級配置到其他層級。

例如，您可以使用「配置 - 層級間」範本將產品從某個層級配置到另一個層級。假設您有下列產品大綱：

```
T_TP
    P_000
    T_TP1
   P_100
           P_110
   etc
     T_TP2
          P_200
          P_210
   etc
     T_TP3
   P_291
           P_292
```

在此範例中，您將會在「配置 - 層級間」範本中為開始使用層級/層代 3，為結束使用層級/層代 4，將它們從父成員 (T_TP1、T_TP2、T_TP3) 配置到其子項 (層級 0) 成員。

若要使用「配置 - 層級間」範本，請執行下列動作：

1. 建立或開啟商業規則或是範本。
2. 請執行下列其中一項動作：

- 對於*圖形*規則或範本，在**新物件**之下選取**配置 - 層級間**，然後將它放入**開始**與**結束**之間的流程圖中。
- 對於*指令碼*規則或範本，請展開**系統範本**，選取**配置 - 層級間**，然後將它放入指令碼中想要的區域。

3. 在**配置 - 層級間**中輸入要求的資訊。

表格 5-4　配置 - 層級間精靈步驟

步驟	解釋
檢視點	為每個在配置期間不要改變的所列維度，選取一或多個成員。 下列選項為可用選項： · 按一下**使用預先定義的選擇**旁邊的 以選取變數，從而將所有維度填入值。 · 按一下**成員選取器**旁邊的 ，以選取每個維度的成員與函式。 · 選取維度，然後按一下 ，以選取變數、成員或函式。
來源	為每個列出的維度，選取您要配置資料的成員。選取成員後，請輸入要配置的來源百分比。 **備註：**請將您想用於層級間配置的維度保留為空。(您可在下一個步驟中選取該維度)
配置範圍	透過輸入要求的資訊，以定義配置範圍。 規則會將所選成員下方的所有資料從指定層級配置作為開始層級，而位於指定層級上方的所有資料配置成為結束層級。
目標	· 如果在**來源**步驟中已定義維度且尚未用於配置，請為每個維度定義目標成員。 您在**來源**中選取的成員預設會在此處輸入。使用下列其中一個選項定義目標成員： – 按一下**使用預先定義的選擇**旁邊的 以選取變數，從而將所有維度填入值。 – 按一下**成員選取器**旁邊的 ，以選取每個維度的成員與函式。 – 選取維度，然後按一下 ，以選取變數、成員或函式。 基於最佳化，選取僅有一個成員的維度。 · 選取要從中選取單一成員的維度。
偏移	如果您不想要產生偏移計算，請定義偏移或是將這個步驟中的表格保留空白。 如果您未定義偏移，請執行下列動作： · 定義偏移的維度範圍 若要將配置值的*總數*寫入偏移成員，請選取要定義偏移的成員。 · 在配置維度上定義偏移成員 輸入成員，或是按一下 以選取成員。
動因	定義用來計算百分比的動因 (基礎)，以套用至配置範圍的每個成員。 · 僅為與檢視點不同的維度選取成員。使用下列其中一個選項以選取成員： – 按一下**使用預先定義的選擇**旁邊的 以選取變數，從而將所有維度填入值。每個維度僅選取一個成員。 – 按一下**成員選取器**旁邊的 ，以選取每個維度的成員與函式。 – 選取維度，然後按一下 ，以選取變數、成員或函式。 · 至是否在配置維度上更新動因聚總。
其他選項	回答精靈中的問題。

使用配置範本

以平均的方式或是根據指定的動因，使用「配置」範本將值從來源配置到目的地。

例如，您可以使用「配置」範本，透過以 Gross Sales 做為基礎，將管理支出配置到 Product 維度的層級 0 成員。

若要使用「配置」範本，請執行下列動作：

1. 建立或開啟商業規則或是範本。
2. 請執行下列其中一項動作：
   - 對於*圖形* 規則或範本，在**新物件**之下選取**配置**，然後將它放入**開始**與**結束**之間的流程圖中。
   - 對於*指令碼* 規則或範本，請展開**系統範本**，選取**配置**，然後將它放入指令碼中想要的區域。
3. 在**配置精靈**中輸入要求的資訊。

表格 5-5　配置精靈步驟

步驟	解釋
資訊	描述範本的功能。
檢視點	輸入在配置中不會發生變化的成員。您必須為各個維度至少選取一個成員。 以下是選取成員的可用選項： · 按一下**使用預先定義的選擇**旁邊的 以選取變數，從而將所有維度填入值。 · 按一下**成員選取器**旁邊的 ，以選取每個維度的成員。 · 選取維度，然後按一下 ，以選取變數、成員或函式。
來源	為每個維度選取要配置資料的成員，然後輸入要配置的來源百分比。 例如，若要配置來源值的 25%，請輸入 25。
配置範圍	選取要配置資料的維度，並為此維度選取父成員。 會將此資料配置給資料庫大綱中指定父成員下方的層級 0 成員。(層級 0 成員是大綱中的最低成員，在其下方沒有成員。)
目標 - 標的	為來源中定義的每個維度指定目標成員。
目標 - 偏移	**選擇性。** · 為每個維度輸入成員，以指定偏移來產生偏移計算。 偏移值可作為來源的相同或相對值計算。 例如，如果您在要配置的來源中有美元 $1,000，即可使用偏移，如此便不需要將數字加倍。當您執行規則時，會將美元 $1000 配置給層級 0，而且偏移將會在指定的成員中建立 -1000。 · 指定是否要在配置程序開始前先清除偏移資料。 · 指定偏移值是否應為已配置資料的反向符號。
排除	**選擇性。**選取您希望從配置範圍排除的成員。 這些成員將不會收到任何配置的資料，且其動因值已從動因百分比基礎排除。
動因	定義「使用動因來配置」或「平均配置」。 為與檢視點不同的每個維度選取一個成員。將會自檢視點和規則中定義的上方成員範圍繼承您保持空白的維度。
傳播方式	如果您在**動因**步驟中已選取「平均配置」，請選取選項以傳播資料。

表格 5-5 (續) 配置精靈步驟

步驟	解釋
其他選項	回答精靈中的問題。 如果您選取進位方法，視所選進位方法而定，會有一些其他步驟。可以使用下列進位方法： · **定義進位成員**—如果您選取此選項，下一個步驟是**在成員進位**，在此輸入要用於配置的小數位數，然後選取要放配置進位差異的成員。 · **使用最大值**—如果您選取此選項，下一個步驟是**進位最大**，在此您輸入配置要使用的小數位數。 · **使用最小值**—如果您選取此選項，下一個步驟是**最小位數進位**，在此您輸入要用於配置的小數位數。 · **無進位**—如果您選取此選項，則無其他步驟。

教學課程

您的目標	完成此教學課程
瞭解如何使用 Calculation Manager 中的「配置系統」範本來配置「規劃與預算」成本。這是一種無需指令碼的成本配置方法。	使用 Calculation Manager 中的「配置」範本進行成本的配置

使用聚總範本

使用「聚總」範本可聚總您所指定的成員資料值。

使用聚總範本，您能執行下列事項：

- 在檢視點中輸入成員以限制聚總
- 指定是否聚總遺漏值
- 指定是否要將值聚總為當地幣別 (不支援混合聚總)

聚總值時，支援下列計算命令：

- SET UPTOLOCAL ON：將資料聚總為當地幣別 (此命令不支援混合聚總)。
- SET CACHE HIGH：設定計算器快取的值 (此命令不支援混合聚總)。
- SET AGGMISSG ON：聚總資料庫中缺少的值
- SET FRMLBOTTOMUP ON：最佳化「稀疏」維度上的計算

例如，您可以使用「聚總」範本透過父成員聚總密集或是稀疏維度，然後選取聚總層級，例如子代、子項或是祖代 (全部)。

若要使用「聚總」範本，請執行下列動作：

1. 建立或開啟商業規則或是範本。
2. 請執行下列其中一項動作：
   - 對於*圖形* 規則或範本，在**新物件**之下選取**聚總**，然後將它放入**開始**與**結束**之間的流程圖中。
   - 對於*指令碼* 規則或範本，請展開**系統範本**，選取**聚總**，然後將它放入指令碼中想要的區域。
3. 在**聚總精靈**中輸入要求的資訊。

表格 5-6　聚總精靈步驟

步驟	解釋
資訊	描述範本的功能。
檢視點	為您要限制聚總的維度選取成員。 以下是選取成員的可用選項。 · 按一下**使用預先定義的選擇**旁邊的 以選取變數，從而將所有維度填入值。 · 按一下**成員選取器**旁邊的 ，以選取每個維度的成員。 · 選取維度，然後按一下 ，以選取變數、成員或函式。
完整密集聚總	最多選取兩個密集維度以完全聚總，並指定密集維度是否已儲存非層級 0 成員。請注意下列事項： · 如果密集維度已儲存非層級 0 成員，則會在指令碼中建立 Calc Dim (密集維度)。 · 如果密集維度未儲存非層級 0 成員，則不會為密集維度產生指令碼。 · 如果您要針對密集維度 (含函式的父成員：子代、子項或祖代) 執行局部聚總，則請勿在此步驟中選取此維度。
完整稀疏聚總	最多選取三個稀疏維度以完全聚總，並指定稀疏維度是否具有需要計算的成員公式。如果稀疏維度具有成員公式，則會計算稀疏維度。如果稀疏維度沒有成員公式，則會聚總稀疏維度。
局部維度聚總密集	最多選取兩個密集維度進行局部聚總。為每個密集維度選取成員和聚總層級。請注意下列事項： · 如果您在「完整密集聚總」步驟中有未使用的密集維度，會顯示「局部維度聚總 - 密集」步驟。如果您在「完整密集聚總」頁籤中使用兩個密集維度，而且在應用程式中已沒有密集維度，則不會顯示「局部維度聚總 - 密集」步驟。 · 如果您要針對稀疏維度 (含函式的父成員：子代、子項或祖代) 執行局部聚總，則請勿在此步驟中選取此維度。
局部維度聚總 - 稀疏	最多選取三個稀疏維度進行局部聚總。為每個稀疏維度選取成員和聚總層級。請注意下列事項： **備註：** · 局部聚總可用的稀疏維度會以灰色顯示。 · 如果您在「完整稀疏聚總」步驟中有未使用的稀疏維度，則會顯示「局部維度聚總 - 稀疏」步驟。如果您在「完整稀疏聚總」頁籤中使用三個稀疏維度，且在應用程式中沒有其他的稀疏維度，則不會顯示**局部維度聚總 - 稀疏**步驟。 · 如果您要針對密集維度 (含函式的父成員：子代、子項或祖代) 執行局部聚總，則請勿在此步驟中選取此維度。
設定值	指定此步驟的設定值。

使用 SET 命令範本

您可使用「SET 命令」範本，輸入將計算指令碼效能最佳化的指令碼命令。

您可以包括資料量、資料處理、記憶體使用，以及執行緒和記錄指令碼命令。

若要使用「SET 命令」範本，請執行下列動作：

1. 建立或開啟商業規則或是範本。
2. 請執行下列其中一項動作：
   - 對於*圖形*規則或範本，在**新物件**之下選取 **SET 命令**，然後將它放入**開始**與**結束**之間的流程圖中。

- 對於*指令碼*規則或範本，請展開**系統範本**，選取 **SET 命令**，然後將它放入指令碼中想要的區域。

3. 在 **Set 命令精靈**中輸入要求的資訊。

   此精靈包括下列步驟：

   - **資料磁碟區**
   - **資料處理**
   - **記憶體使用量**
   - **執行緒**
   - **日誌**

   每個步驟都有數個問題，可以透過選取問題旁邊的下拉清單中的選項來回答，您可以在此範本中選取一或多個選項。

> **備註：**
>
> 您不需要回答「SET 命令」精靈中的每個問題。您可以只回答適用於自己情況的問題。

> **備註：**
>
> 當應用程式位於 Essbase Hybrid 中時，精靈中將僅顯示**資料磁碟區**、**資料處理**和**執行緒**頁籤。

使用幣別轉換範本

「幣別轉換」範本使用系統類型科目或您選取的科目，將科目值轉換為報表幣別。

您要在使用選取的多種幣別建立的規劃應用程式中使用「幣別轉換」範本。

如需使用「幣別轉換」範本的範例，請參閱幣別轉換範本範例。

> **備註：**
>
> 如果規劃應用程式是使用多種幣別而建立，「幣別轉換」範本不會顯示在「系統範本」清單中。

若要使用「幣別轉換」範本，請執行下列動作：

1. 建立或開啟商業規則或是範本。
2. 請執行下列其中一項動作：
   - 對於*圖形*規則或範本，在**新物件**之下選取**幣別轉換**，然後將它放入**開始**與**結束**之間的流程圖中。
   - 對於*指令碼*規則或範本，請展開**系統範本**，選取**幣別轉換**，然後將它放入指令碼中想要的區域。

3. 在**幣別轉換精靈**中輸入要求的資訊。

表格 5-7 幣別轉換精靈步驟

步驟	解釋
資訊 - 幣別	描述範本的功能以及使用此範本所需的資訊。
匯率選項	回答精靈中關於幣別的問題： · **您的幣別位於哪個維度？** 選取對應至幣別維度的維度。 · **請選取報表幣別。** 選取您要轉換的報表幣別。 · **請選取包含幣別成員的父成員。** 選取在應用程式中使用的所有幣別其父項的父成員，例如，"Input Currencies"。 · **此報表幣別使用哪一種匯率 (幣別)？** 選取與您在上方所選報表幣別對應的幣別。例如，如果您已為報表幣別選取 *EUR 報表*，您將會選取 *EUR* (其中 *EUR* 是用以將資料轉換至 *EUR 報表* 的幣別)。 · **是否要使用幣別轉換的科目類型？** 選取**是**使用幣別轉換的科目類型。如果您選取**是**，費用與收入科目將會使用平均費率，而負債與權益科目將會使用期末匯率來轉換。 如果您選取**否**，會在精靈中稍後提示，以指定哪個科目要用於平均匯率與期末匯率 · **您是否要計算「平均」匯率？** 如果您選取**不計算平均**，不會顯示與平均相關的步驟，而且不會產生根據平均匯率計算幣別的指令碼。 · **您是否要計算「期末」匯率？** 選取是否要計算期末匯率。 如果您選取**不計算期末**，不會顯示與期末相關的步驟，而且不會產生根據期末匯率計算幣別的指令碼。
POV	選取成員以用於轉換為報表幣別。 以下選項是選取成員的可用選項。 · 按一下**使用預先定義的選擇**旁邊的 以選取變數，從而將所有維度填入值。 · 按一下**成員選取器**旁的 ，以選取每個維度的成員。 · 選取維度，然後按一下 ，以選取變數、成員或函式。
科目平均	輸入要使用平均匯率以轉換至報表幣別的科目。 **備註：**這個步驟只會在您於**匯率選項**步驟的**您是否要使用幣別轉換的科目類型？**中選取**否**時才會顯示。
科目期末	輸入要使用期末匯率以轉換至報表幣別的科目。 **備註：**這個步驟只會在您於**匯率選項**步驟的**您是否要使用幣別轉換的科目類型？**中選取**否**時才會顯示。
FX 平均	選取包含平均匯率的成員。 **備註：**您只需要選取與檢視點中成員不同的成員。
FX 結束	選取含有期末匯率的成員。 **備註：**您只需要選取與檢視點中成員不同的成員。

幣別轉換範本範例

1. 在未使用多種幣別來建立的 Planning 應用程式中，除了現有維度之外，請建立名為 "Currencies: 的自訂維度並新增下列成員：
   - 輸入幣別
     - USD
     - CAD
     - EUR
     - GBP
     - INR
   - 報表幣別
     - EUR 報表
     - US 報表
     - Can 報表
2. 在 "Account" 維度中，為 FX_Rates 新增下列成員：
   - FX_Average
   - FX_Ending
3. 在 "Entity" 維度，新增實體 "Company Assumptions"。
4. 使用 Oracle Smart View for Office 或規劃表單，在下列儲存格輸入匯率：
   - **科目** - FX_Average 和 FX_Ending
   - **期間** - 輸入每個月的匯率
   - **年度** - FY12
   - **案例 -** Current
   - **版本** - BU Version_1
   - **實體** - Company Assumptions
   - **產品** - 000
   - **幣別** - 輸入每個幣別的匯率
5. 在「變數」中：
   a. 使用名為 "Reporting_Currency" 的成員變數，為 Currencies 維度建立應用程式層級執行時期提示，然後輸入預設值「EUR 報表」
   b. 使用名為 「輸入幣別」的成員變數，為 Currencies 維度建立應用程式層級執行時期提示，然後輸入預設值「EUR」。
   c. 在此 Planning 應用程式中建立商業規則，然後將幣別轉換系統範本拖放至商業規則流程圖中。

      「幣別轉換精靈」會顯示出來。
6. 在「幣別轉換精靈」中，執行下列步驟：
   a. **步驟 1 – 匯率選項**

- 您的幣別位於哪個維度？按一下下拉清單，然後選取您在以上建立的 Currencies 維度。
- 選取報表幣別：依序按一下 ，選取**變數**，然後選取 Reporting_Currency 變數。

  在此範例中，我們對報表幣別使用變數，以便可以在規劃中針對每個報表幣別啟動規則。
- 選取包含您的幣別成員的父成員：按一下，然後選取「輸入幣別」父成員。

  您需要為幣別成員選取父項的原因是，如果未來新增幣別，則在 Planning 中只要重新部署並再啟動一次此商業規則即可。範本的指令碼會將新幣別新增至指令碼，並正確計算報表幣別。
- 此報表幣別使用哪一種匯率 (幣別)？按一下，然後選取「輸入幣別」變數。

  在此範例中，我們對報表幣別使用變數，以便可以在規劃中針對每個報表幣別啟動規則。
- 是否要使用幣別轉換的科目類型？使用下拉清單，然後選取**是**。

  如果您選取「否」，稍後的步驟中將會提示您指定哪些科目要使用「平均/期末」。
- 您是否要計算「平均」匯率？

  使用下拉清單，然後選取**計算平均**。
- 您是否要計算「期末」匯率？

  使用下拉清單，然後選取**計算期末**。

b. **步驟 2 – POV**

針對每個維度輸入下列資訊：

- **Period** — 將期間保留為空白，以便它可寫入至所有期間
- **年度** — "FY12"
- **案例** — "Current"
- **版本** — “BU Version_1"
- **實體** — @Relative("South",0)
- **產品** — @Relative("Tennis",0),@Relative("Golf",0)

c. **步驟 3 — FX_Average**

- 選取包含平均匯率的成員：
  - **科目** — "FX_Average"
  - **期間**
  - **年度**
  - **案例**
  - **版本**
  - **實體** — "Company Assumptions"
  - **產品** — "000"
- 將 Period 維度保留為空白，以便它對每個期間使用該匯率。
- 將 Year，Scenario 和 Version 保留為空白，以便它們對這些維度使用 POV 輸入的成員。

d. **步驟 4 – FX_Ending**

- 選取包含平均匯率的成員：
  - **科目** — "FX_Average"
  - **期間**
  - **年度**
  - **案例**
  - **版本**
  - **實體** — "Company Assumptions"
  - **產品** — "000"
- 將 Period 維度保留為空白，以便它對每個期間使用該匯率。
- 將 Year，Scenario 和 Version 保留為空白，以便它們對這些維度使用 POV 輸入的成員。

7. 儲存，驗證商業規則並部署至 Planning。

   此範例中的應用程式包含下列資料：

表格 5-8　範例資料第 1 部分

N/A	N/A	N/A	FY12	目前	BU Version_1
N/A	N/A	田納西	田納西	佛羅里達	佛羅里達
N/A	N/A	1 月	1 月	1 月	1 月
N/A	N/A	G400	G100	G400	G100
銷售總額	USD	10750	13450	9500	9699
銷售總額	CAD	10450	10000	14450	13000
銷售總額	EUR	40000	41450	42450	65000
銷售總額	GBP	13250	16750	172000	16300
銷售總額	INR	750000	750000	750000	750000
銷售總額	US 報表	#Missing	#Missing	#Missing	#Missing
銷售總額	CAN 報表	#Missing	#Missing	#Missing	#Missing
銷售總額	EUR 報表	#Missing	#Missing	#Missing	#Missing
薪資	USD	5000	5000	5000	5000
薪資	CAD	4500	4500	4500	4500
薪資	EUR	5500	5500	5500	5705
薪資	GBP	1200	1200	1200	1200
薪資	INR	100000	100,000	100,000	100,000
薪資	US 報表	#Missing	#Missing	#Missing	#Missing
薪資	CAN 報表	#Missing	#Missing	#Missing	#Missing
薪資	EUR 報表	#Missing	#Missing	#Missing	#Missing

請注意，此表格中的任何報表幣別都沒有值。

8. 在規劃中啟動規則。對於「報表幣別」變數，請輸入 **EUR 報表**，對於輸入幣別，請輸入 **EUR**。按一下**啟動**。

9. 對於規劃中的每個科目，請確定 **Eur 報表**現在是以歐元顯示科目總計：

表格 5-9　範例資料第 2 部分

N/A	N/A	N/A	FY12	目前	BU Version_1
N/A	N/A	田納西	田納西	佛羅里達	佛羅里達
N/A	N/A	1 月	1 月	1 月	1 月
N/A	N/A	G400	G100	G400	G100
銷售總額	USD	10750	13450	9500	9699
銷售總額	CAD	10450	10000	14450	13000
銷售總額	EUR	40000	41450	42450	65000
銷售總額	GBP	13250	16750	172000	16300
銷售總額	INR	750000	750000	750000	750000
銷售總額	US 報表	#Missing	#Missing	#Missing	#Missing
銷售總額	CAN 報表	#Missing	#Missing	#Missing	#Missing
銷售總額	EUR 報表	52161391	52168683	52170600	52190796.69
薪資	USD	5000	5000	5000	5000
薪資	CAD	4500	4500	4500	4500
薪資	EUR	5500	5500	5500	5705
薪資	GBP	1200	1200	1200	1200
薪資	INR	100000	100000	100000	100000
薪資	US 報表	#Missing	#Missing	#Missing	#Missing
薪資	CAN 報表	#Missing	#Missing	#Missing	#Missing
薪資	EUR 報表	6963318	6963318	6963318	6965471.887

10. 在 Planning 中，您可以針對每個報表幣別，視需要多次啟動商業規則。您可以每次變更變數輸入，並針對變數以及應用程式中可用的每個報表幣別，輸入所需的「報表幣別」及其對應的匯率名稱。

    在此範例中，您可以執行兩次以上的規則，一次針對使用 USD 的美國報表，然後針對使用 CAD 的加拿大報表。

顯示範本流程

當您複查系統範本所產生的指令碼時，這有時對於查看範本流程很有用。

範本流程會顯示：

- 範本所產生的指令碼流程
- 使用您在範本步驟中所選取的項目來取代設計時期提示的文字

若要顯示範本的流程，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下含有此系統範本的商業規則或自訂範本，然後選取**開啟**。
2. 如果您尚未這樣做，請在範本中填入所有的資訊。
3. 在規則設計工具流程圖中，用滑鼠右鍵按一下該系統範本，然後選取**顯示範本流程**。

   範本流程會在「範本設計工具」中以唯讀模式顯示。您可以按一下流程圖中的每個元件，以顯示與該元件相關聯的指令碼以及在相關 DTP 的步驟中所輸入的值。

   如果在流程圖中的元件是以灰色顯示，則表示該元件不是根據您在範本中輸入資料時所選取項目而產生之指令碼的一部分。

   當具有條件元件以及同時適用於條件之 True 和 False 的指令碼時，這對於查看範本流程特別有用。用以產生指令碼的範本所使用的路徑會以粗體顯示，而另一個路徑則以灰色顯示。

將系統範本另存為自訂範本

您可能想自訂系統範本的內容。

雖然您不能編輯系統範本，但是您可以使用系統範本上的「另存新檔」，以建立可以編輯的自訂範本。原始系統範本並未變更。

若要將系統範本另存為自訂已定義的範本，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下含有此系統範本的商業規則，然後選取**開啟**。
2. 在**新物件**或**現有物件**之下，或是在規則設計工具流程圖或指令碼中，用滑鼠右鍵按一下系統範本，然後選取**開啟**。
3. 在**範本設計工具**中開啟系統範本時，將會以唯讀方式開啟。按一下**確定**，選取**動作**，然後選取**另存新檔**。
4. 在**另存新檔**對話方塊中，輸入範本的新名稱，然後選取應用程式類型、應用程式以及立方體，然後按一下**確定**。

   新範本會顯示在應用程式的**範本**節點以及所選的立方體中。您可以開啟並進行自訂。

> **備註：**
>
> 您可能必須重新整理**系統檢視**中的應用程式清單，才能查看新範本。用滑鼠右鍵按一下**範本**節點，然後選取**重新整理**。

將系統範本從商業規則中移除

您可以將系統範本從商業規則中移除。

若要移除系統範本，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下想要從中移除系統範本的商業規則，然後選取**開啟**。
2. 對於圖形規則，在規則設計工具的流程圖中，用滑鼠右鍵按一下系統範本，然後選取**移除**。
3. 對於指令碼規則，醒目提示含有系統範本的整行，按一下滑鼠右鍵，然後選取**刪除**。
4. 按一下 。

6

使用自訂範本

另請參閱：

- 關於自訂範本
  自訂範本是由管理員設計，用於商業規則和其他範本中。
- 建立自訂範本
  您可以建立*圖形*自訂範本與*指令碼*自訂範本。
- 針對自訂範本建立設計時期提示
  您可以為圖形自訂範本輸入設計時期提示，這樣您在使用該範本設計商業規則時，就會看到輸入正確資訊的提示。
- 開啟自訂範本
  您可以從系統檢視、部署檢視和自訂檢視中開啟自訂範本。
- 重新整理自訂範本
  建立自訂範本之後，您可能必須重新整理系統檢視中的應用程式清單，才能查看範本節點中的新範本。
- 顯示自訂範本的使用情況
  您可以顯示使用自訂範本的商業規則清單。
- 複製和貼上自訂範本
  您可以複製自訂範本，並貼到其他商業規則中。
- 刪除自訂範本
  您可以從系統檢視、自訂檢視及部署檢視刪除自訂範本。
- 尋找和取代圖形自訂範本中的文字
  您可搜尋自訂範本中的文字字串。您也可以取代文字字串的所有例項。

關於自訂範本

自訂範本是由管理員設計，用於商業規則和其他範本中。

您可以從下列地方存取自訂範本：

- 任何檢視內計畫類型的**範本**節點
- **現有物件**下的規則設計工具

有兩種類型的自訂範本：

- *圖形*自訂範本可包含公式、指令碼、條件式、成員區塊、成員範圍、固定迴圈、中繼資料迴圈，以及 DTP 指派元件。
- *指令碼*自訂範本可包含指令碼，但不包含元件。

圖形和指令碼自訂範本也可包含現有規則、公式和指令碼 (共用元件)、其他自訂範本，以及系統範本。

您可以在規則中包含圖形和指令碼範本，或其他自訂範本。

圖形和指令碼範本之間的差異：

- 圖形範本有 DTP 指派元件、中繼資料迴圈元件，以及在元件中使用 DTP 條件式的能力。
- 指令碼範本可部署至 Planning，並可從中啟動。

建立自訂範本

您可以建立*圖形*自訂範本與*指令碼*自訂範本。

另請參閱：

- 建立圖形自訂範本
- 建立指令碼自訂範本
- 為 Planning BSO 立方體建立 Groovy 範本
- 為 Planning ASO 立方體建立 Groovy 範本

建立圖形自訂範本

若要建立圖形自訂範本，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下**範本**，然後選取**新增**。
2. 在**新自訂範本**中，輸入必要資訊，然後按一下**確定**。

   新自訂範本對話方塊會自動填入您要在「系統檢視」中使用的規劃應用程式類型、應用程式，以及計畫類型。
3. 在**範本設計工具**中，將新物件和現有物件拖曳至範本的流程圖。

   若要新增物件，請將物件從**新物件**或**現有物件**中，拖曳至**開始**和**結束**之間的流程圖。

表格 6-1　要插入圖形自訂範本中的新物件

物件	另請參閱
公式	公式元件
指令碼	指令碼元件
條件	條件元件
成員區塊	成員區塊元件
成員範圍	成員範圍元件
固定的迴圈	固定迴圈元件
中繼資料迴圈	中繼資料迴圈元件
DTP 指派	DTP 指派元件

表格 6-2　要插入圖形自訂範本中的現有物件

物件	另請參閱
規則	設計商業規則
指令碼	共用指令碼和公式元件
公式	共用指令碼和公式元件
範本	使用自訂範本

4. 在**特性**中，輸入範本的特性。

當您將元件新增至範本，並在流程圖中的元件之間移動時，特性會改變。若要為特定元件輸入特性，請在流程圖中選取該元件。

5. 使用**設計時期提示**頁籤，以輸入下列類型的設計時期提示：
   - 設計時期提示可在您使用範本時，提示您輸入資訊。
   - 不可提示的設計時期提示可用於 DTP 指派元件、啟用條件，以及驗證條件。

   請參閱針對自訂範本建立設計時期提示。

   當您開啟自訂範本時，依預設會包含下列系統設計時期提示，您可以視需要將其加入範本：
   - **應用程式類型** — 用於確定應用程式是傳統 (一般) 或 Oracle Project Financial Planning (PFP) 應用程式。
   - **應用程式** — 用於確定應用程式是單一、多重或簡易幣別。
   - **UpperPOV** — 用於確定是否有全域範圍中的成員，或是否有範本插入成員範圍。請參閱使用 UpperPOV 設計時期提示建立圖形範本的範例。
6. 按一下 ，以儲存範本。

> **提示：**
>
> 當您編輯範本中的元件時，您可能希望增加或減少元件圖示的大小，以及流程圖中顯示的詳細資料量。若要這麼做，請使用 ，以在流程圖中放大或縮小。

建立使用 UpperPOV 設計時期提示之圖形範本的範例

下列步驟顯示如何建立使用 UpperPOV 設計時期提示的圖形範本。此範例中的步驟使用範例規劃應用程式。

1. 在**系統檢視**中，用滑鼠右鍵按一下**範本**，然後選取**新增**。
2. 在**新自訂範本**中，輸入要求的資訊，然後按一下**確定**。
3. 在**設計時期提示**頁籤中，按一下 ，然後選取**在結尾插入列**。
4. 在**建立提示**對話方塊中，執行下列動作：
   a. 在**特性**頁籤中，輸入下列資訊：
      - **名稱** — 輸入 "MR1"。
      - **類型** - 選取「成員範圍」
      - 選取**給予提示？**
      - **DTP 文字** - 輸入成員範圍
   b. 在**相依性**頁籤中，針對 **UpperPOV** 按下拉清單，然後選取**排除**。
   c. 按一下**確定**。
5. 在**範本設計工具**流程圖中，在*成員範圍*元件中拖曳。
6. 在**成員範圍**頁籤中，按一下 。

7. 在 **DTP 選取器**對話方塊中，選取步驟 4 中建立的「成員範圍」設計時期提示，然後按一下**確定**，接著確認已選取**動態連結變數**。
8. 在範本中建立步驟，然後將名稱為 "MR1" 的新 DTP 加入步驟。
9. 儲存範本。
10. 在規則中的*成員範圍*中拖曳，然後針對 "Entity" 與 "Product" 輸入成員。
11. 將新圖形範本拖曳至規則中的成員範圍。

請注意已顯示*成員範圍*類型 DTP 的步驟，並注意未顯示 "Entity" 與 "Product" 維度。這是因為我們已將*成員範圍*類型的 DTO 排除 UpperPOV，而且 UpperPOV (拖曳至規則的成員範圍) 正使用維度 "Entity" 與 "Product"。

如果您將*成員範圍*類型之 DTP 的相依性變更為*內含*，在規則中，只會顯示維度 "Entity"，而在範本步驟中，只會顯示維度 "Product"。

您也可以在 DTP 條件之圖形範本中使用 UpperPOV 系統 DTP，以確定 UpperPOV 是否擁有成員 (不為空白) 或沒有成員 (為空白)，並確定是否有已拖曳範本 (可用) 或無拖曳範本 (無可用) 的成員範圍元件。

建立指令碼自訂範本

當您使用指令碼建立自訂範本，您可以：

- 將指令碼範本部署至規劃
- 啟動規劃中的範本，其中的步驟將顯示可讓您在執行時期輸入資料
- 使用執行時期函式和設計時期提示

若要使用指令碼建立自訂範本，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下**範本**，然後選取**新增**。
2. 在**新建自訂已定義的範本**對話方塊中，輸入要求的資訊，然後按一下**確定**。

   系統會自動在對話方塊中填入您在「系統檢視」中使用的 Planning 應用程式類型、應用程式及立方體。
3. 開啟範本時，按一下**設計工具**旁的下拉清單，然後選取**編輯指令碼**。

   這可將範本從圖形轉換成指令碼。
4. 在**指令碼類型**下拉清單中，選取**計算指令碼**。

   如果您沒有看到**指令碼類型**下拉清單，請按一下「範本設計工具」右側的**>>**。
5. 在**特性**中，輸入範本的特性。
6. 在**範本設計工具**中輸入指令碼。
7. 使用**設計時期提示**頁籤，以輸入設計時期提示，該題是可在您啟動規劃中的範本時，提示您輸入資訊。請參閱針對自訂範本建立設計時期提示
8. 按一下 ，以儲存範本。
9. 將範本部署至規劃。
10. 開啟規劃，然後啟動範本。

為 Planning BSO 立方體建立 Groovy 範本

如何為 Planning BSO 立方體建立 Groovy 範本：

1. 在**系統檢視**中，用滑鼠右鍵按一下**範本**，然後選取**新增**。
2. 在**新建自訂已定義的範本**對話方塊中，輸入要求的資訊，然後按一下**確定**。

   系統會自動在對話方塊中填入您在「系統檢視」中使用的 Planning 應用程式類型、應用程式及立方體。
3. 開啟範本時，按一下**設計工具**旁的下拉清單，然後選取**編輯指令碼**。
4. 在**指令碼類型**下拉清單中，選取 **Groovy 指令碼**。

   如果您沒有看到**指令碼類型**下拉清單，請按一下「範本設計工具」右側的**>>**。
5. 在**特性**中，輸入範本的特性。
6. 在**範本設計工具**中輸入指令碼。
7. 使用**設計時期提示**頁籤，以輸入設計時期提示，該題是可在您啟動規劃中的範本時，提示您輸入資訊。

   請參閱針對自訂範本建立設計時期提示。
8. 按一下 ，以儲存範本。
9. 將範本部署至規劃。
10. 開啟規劃，然後啟動範本。

> **備註：**
>
> 請參閱 指派 Groovy 範本的存取權限，以取得如何指派 Planning 中 Groovy 範本之存取權限的相關資訊。

為 Planning ASO 立方體建立 Groovy 範本

如何為 Planning ASO 立方體建立 Groovy 範本：

1. 在**系統檢視**中，用滑鼠右鍵按一下**範本**，然後選取**新增**。
2. 在**新建自訂已定義的範本**對話方塊中，輸入要求的資訊，然後按一下**確定**。

   系統會自動在對話方塊中填入您在「系統檢視」中使用的 Planning 應用程式類型、應用程式及立方體。
3. 在**特性**中，輸入範本的特性。
4. 在**範本設計工具**中輸入指令碼。
5. 使用**設計時期提示**頁籤，以輸入設計時期提示，該題是可在您啟動規劃中的範本時，提示您輸入資訊。

   請參閱針對自訂範本建立設計時期提示。
6. 按一下 ，以儲存範本。

7. 將範本部署至規劃。
8. 開啟規劃，然後啟動範本。

針對自訂範本建立設計時期提示

您可以為圖形自訂範本輸入設計時期提示，這樣您在使用該範本設計商業規則時，就會看到輸入正確資訊的提示。

相關主題

- 設計階段提示的類型
- 為設計時期提示建立步驟
- 定義設計階段提示的相依性
- 為設計時期提示定義限制
- 搜尋與取代設計時期提示中的文字

設計階段提示的類型

屬性 DTP

從該自訂範本所屬的應用程式定義屬性。

例如，假設您建立設計時期提示以輸入產品尺寸。在這種情況下，您可以選取「尺寸」維度。您也可以視需要輸入預設值；例如，「大型」。如果您選取**給予提示？**，則在使用範本時，系統會提示使用者輸入屬性成員 (在此範例中為維度大小)。

若要建立屬性設計時期提示，請執行下列動作：

1. 在**範本設計工具**中，開啟現有範本或建立新範本。
2. 在**設計時期提示**頁籤中，按一下 。
3. 如果您已針對範本定義設計時期提示，請選取輸入新設計時期提示的位置。
4. 在**建立提示**對話方塊中，在**特性**頁籤上，執行下列動作：
   - 輸入**名稱**，並選取**屬性**做為**類型**。
   - 選取需要的選項：
     - **給予提示？** — 選取此選項以允許 DTP 用於步驟中。在規則中使用範本或在規劃中啟動指令碼範本時，會在提示使用者輸入的範本步驟中顯示 DTP。
     - **強制？** — 選取此選項，使其強制為設計時期提示輸入資訊。
     - **唯讀** — 選取此選項讓使用者只能讀取提示。
     - **已隱藏** — 選擇此選項可隱藏使用者的設計時期提示。在 Planning 中，在表單設計中的**商業規則**頁籤中，將具有範本或指令碼範本的規則新增至表單。在**商業規則特性**面板中，選取**隱藏提示**，然後選取**使用表單上的成員**。
   - 在 **DTP 文字**中，輸入當系統提示使用者輸入時，您希望使用者看見的文字。如果您選取**給予提示？**，則需要 **DTP 文字**。
   - **選擇性。**在**註解**中，輸入關於 DTP 的註解。
   - 在**維度**中，選取屬性維度。

「維度」下拉清單顯示來自目前應用程式的屬性。如果您未選取維度，當系統提示您輸入時，將看見所有屬性維度。

- **選擇性。**在**預設值**中，指派 DTP 的預設值。

  若要指派預設值，請按一下 ，然後選取**成員**，然後在**成員選取器**中選取成員。

  維度下拉清單中選取的屬性維度隨即顯示。展開此維度以顯示可用成員。

布林值 DTP

「True」或「False」類型 DTP。

若要建立布林值設計時期提示，請執行下列動作：

1. 在**範本設計工具**中，開啟現有範本或建立新範本。
2. 在**設計時期提示**頁籤中，按一下 。
3. 如果您已針對範本定義設計時期提示，請選取輸入新設計時期提示的位置。
4. 在**建立提示**對話方塊中，在**特性**頁籤上，執行下列動作：
   - 輸入**名稱**，並選取**布林值**做為**類型**。
   - 選取需要的選項：
     - **給予提示？** — 選取此選項以允許 DTP 用於步驟中。在規則中使用範本或在規劃中啟動指令碼範本時，會在提示使用者輸入的範本步驟中顯示 DTP。
     - **強制？** — 選取此選項，使其強制為設計時期提示輸入資訊。
     - **唯讀** — 選取此選項讓使用者只能讀取提示。
     - **已隱藏** — 選擇此選項可隱藏使用者的設計時期提示。在 Planning 中，在表單設計中的**商業規則**頁籤中，將具有範本或指令碼範本的規則新增至表單。在**商業規則特性**面板中，選取**隱藏提示**，然後選取**使用表單上的成員**。
   - 在 **DTP 文字**中，輸入當系統提示使用者輸入時，您希望使用者看見的文字。如果您選取**給予提示？**，則需要 **DTP 文字**。
   - **選擇性。**在**註解**中，輸入關於 DTP 的註解。
   - **選擇性。**在**預設值**中，選取 **True** 或 **False**。

跨維度 DTP

從範本所屬的應用程式，定義成員的跨維度語法 (->) 範圍。

跨維度 DTP 允許使用者從每個維度選取一個成員。例如，跨維度設計時期提示可以用於公式來源。

1. 在**範本設計工具**中，開啟現有範本或建立新範本。
2. 在**設計時期提示**頁籤中，按一下 。
3. 如果您已針對範本定義設計時期提示，請選取輸入新設計時期提示的位置。
4. 在**建立提示**對話方塊中，在**特性**頁籤上，執行下列動作：
   - 輸入**名稱**，並選取**跨維度**做為**類型**。
   - 選取需要的選項：

  - **給予提示？** — 選取此選項以允許 DTP 用於步驟中。在規則中使用範本或在規劃中啟動指令碼範本時，會在提示使用者輸入的範本步驟中顯示 DTP。
  - **強制？** — 選取此選項，使其強制為設計時期提示輸入資訊。
  - **唯讀** — 選取此選項讓使用者只能讀取提示。
  - **已隱藏** — 選擇此選項可隱藏使用者的設計時期提示。在 Planning 中，在表單設計中的**商業規則**頁籤中，將具有範本或指令碼範本的規則新增至表單。在**商業規則特性**面板中，選取**隱藏提示**，然後選取**使用表單上的成員**。
- 在 **DTP 文字**中，輸入當系統提示使用者輸入時，您希望使用者看見的文字。如果您選取**給予提示？**，則需要 **DTP 文字**。
- **選擇性。**在**註解**中，輸入關於 DTP 的註解。
- **選擇性。**在**預設值**中，指派 DTP 的預設值。

  若要指派預設值，請執行下列其中一項動作：

  - 按一下 ，然後在每個維度旁邊按一下 ，然後選取 **DTP**、**成員**或**函式**。
  - 按一下 ，以選取 DTP 作為預設值。

5. **選擇性：**在**相依性**頁籤中，選取設計時期提示，然後選取相依性選項：
   - **排除** — 讓設計時期提示 (例如，DTP2) 排除另一個設計時期提示 (例如 DTP1)。當系統提示您 DTP2 時，只會顯示未用於 DTP1 的維度。
   - **內含** — 讓設計時期提示 (例如，DTP4) 內含另一個設計時期提示 (例如 DTP3)。當系統提示您 DTP4 時，只會顯示用於 DTP3 的維度。
   - **無** — 設計時期提示未定義任何相依性。
6. **選擇性：**在**預設限制**頁籤中，執行下列動作：
   - 在**密度**中，選取選項：
     - **顯示稀疏** — 僅針對 DTP 顯示稀疏維度
     - **顯示密集** — 僅針對 DTP 顯示密集維度
     - **兩者** — 針對 DTP 顯示密集與稀疏維度
   - 在**維度類型**中，選取要限制 DPT 的維度類型，以僅顯示選取的維度類型。

     例如，如果您只希望使用者輸入 *Account*、*Entity* 與 *Year* 的成員，請在**維度類型**中選取這些維度。當系統提示使用者輸入時，顯示的唯一維度是類型 *Account*、*Entity* 以及 *Year*。

條件 DTP

定義條件。

若要建立條件設計時期提示，請執行下列動作：

1. 在**範本設計工具**中，開啟現有範本或建立新範本。
2. 在**設計時期提示**頁籤中，按一下 。
3. 如果您已針對範本定義設計時期提示，請選取輸入新設計時期提示的位置。
4. 在**建立提示**對話方塊中，在**特性**頁籤上，執行下列動作：
   - 輸入**名稱**，並選取**條件**作為**類型**。

- 選取需要的選項：
  - **給予提示？** — 選取此選項以允許 DTP 用於步驟中。在規則中使用範本或在規劃中啟動指令碼範本時，會在提示使用者輸入的範本步驟中顯示 DTP。
  - **強制？** — 選取此選項，使其強制為設計時期提示輸入資訊。
  - **唯讀** — 選取此選項讓使用者只能讀取提示。
  - **已隱藏** — 選擇此選項可隱藏使用者的設計時期提示。在 Planning 中，在表單設計中的**商業規則**頁籤中，將具有範本或指令碼範本的規則新增至表單。在**商業規則特性**面板中，選取**隱藏提示**，然後選取**使用表單上的成員**。
- 在 **DTP 文字**中，輸入當系統提示使用者輸入時，您希望使用者看見的文字。

  如果您選取**給予提示？**，則需要 **DTP 文字**。
- **選擇性。**在**註解**中，輸入關於 DTP 的註解。

DateAsNumber DTP

將日期定義為數字。

若要建立 DateAsNumber 設計時期提示，請執行下列動作：

1. 在**範本設計工具**中，開啟現有範本或建立新範本。
2. 在**設計時期提示**頁籤中，按一下 \ 。
3. 如果您已針對範本定義設計時期提示，請選取輸入新設計時期提示的位置。
4. 在**建立提示**對話方塊中，在**特性**頁籤上，執行下列動作：
   - 輸入**名稱**，並選取 **DateAsNumber** 做為**類型**。
   - 選取需要的選項：
     - **給予提示？** — 選取此選項以允許 DTP 用於步驟中。在規則中使用範本或在規劃中啟動指令碼範本時，會在提示使用者輸入的範本步驟中顯示 DTP。
     - **強制？** — 選取此選項，使其強制為設計時期提示輸入資訊。
     - **唯讀** — 選取此選項讓使用者只能讀取提示。
     - **已隱藏** — 選擇此選項可隱藏使用者的設計時期提示。在 Planning 中，在表單設計中的**商業規則**頁籤中，將具有範本或指令碼範本的規則新增至表單。在**商業規則特性**面板中，選取**隱藏提示**，然後選取**使用表單上的成員**。
   - 在 **DTP 文字**中，輸入當系統提示使用者輸入時，您希望使用者看見的文字。

     如果您選取**給予提示？**，則需要 **DTP 文字**。
   - **選擇性。**在**註解**中，輸入關於 DTP 的註解。
   - 選取**允許 #Missing** 以允許 DTP 中的 #Missing 值，而非日期。
   - 在**限制**中，按一下 ，然後定義您可以在設計時期提示中輸入的最小和最大數字。
   - **選擇性。**在**預設值**中，按一下 ，然後選取 **DTP** 值。

維度 DTP

從該範本所屬的應用程式定義維度。

若要建立維度設計時期提示，請執行下列動作：

1. 在**範本設計工具**中，開啟現有範本或建立新範本。
2. 在**設計時期提示**頁籤中，按一下 。
3. 如果您已針對範本定義設計時期提示，請選取輸入新設計時期提示的位置。
4. 在**建立提示**對話方塊中，在**特性**頁籤上，執行下列動作：
   - 輸入**名稱**，並選取**維度**作為**類型**。
   - 選取需要的選項：
     - **給予提示？** — 選取此選項以新增 DTP 至步驟中。當範本用於規則中時，或者當規劃中啟動指令碼範本時，此 DTP 顯示在提示使用者進行輸入的範本步驟中。
     - **強制？** — 選取此選項，使其強制為設計時期提示輸入資訊。
     - **唯讀** — 選取此選項讓使用者只能讀取提示。
     - **已隱藏** — 選擇此選項可隱藏使用者的設計時期提示。在 Planning 中，在表單設計中的**商業規則**頁籤中，將具有範本或指令碼範本的規則新增至表單。在**商業規則特性**面板中，選取**隱藏提示**，然後選取**使用表單上的成員**。
   - 在 **DTP 文字**中，輸入當系統提示使用者輸入時，您希望使用者看見的文字。

     如果您選取**給予提示？**，則需要 **DTP 文字**。
   - **選擇性。**在**註解**中，輸入關於 DTP 的註解。
   - **選擇性。**在**預設值**中，按一下下拉清單，然後從範本所屬的應用程式選取維度。
5. **選擇性。**在**相依性**頁籤中，選取設計時期提示，然後選取相依性選項：
   - **排除** — 讓設計時期提示 (例如，DTP2) 排除另一個設計時期提示 (例如 DTP1)。當系統提示您 DTP2 時，只會顯示未用於 DTP1 的維度。
   - **內含** — 讓設計時期提示 (例如，DTP4) 內含另一個設計時期提示 (例如 DTP3)。當系統提示您 DTP4 時，只會顯示用於 DTP3 的維度。
   - **無** — 設計時期提示未定義任何相依性。

   維度 DTP 可在「成員範圍」、「維度」或「跨維度」DTP 上擁有相依性。
6. **選擇性。在定義限制**頁籤中，執行下列動作：
   - 在**密度**中，選取選項：
     - **顯示稀疏** — 僅針對 DTP 顯示稀疏維度
     - **顯示密集** — 僅針對 DTP 顯示密集維度
     - **兩者** — 針對 DTP 顯示密集與稀疏維度
   - 在**維度類型**中，選取要限制 DPT 的維度類型，以僅顯示選取的維度類型。

     例如，如果您只希望使用者輸入 *Account*、*Entity* 與 *Year* 的成員，請在**維度類型**中選取這些維度。當系統提示使用者輸入時，顯示的唯一維度是類型 *Account*、*Entity* 以及 *Year*。

維度 DTP

從該範本所屬的應用程式定義維度。

若要建立維度設計時期提示，請執行下列動作：

1. 在**範本設計工具**中，開啟現有範本或建立新範本。
2. 在**設計時期提示**頁籤中，按一下 。
3. 如果您已針對範本定義設計時期提示，請選取輸入新設計時期提示的位置。
4. 在**建立提示**對話方塊中，在**特性**頁籤上，執行下列動作：
   - 輸入**名稱**，並選取**維度**作為**類型**。
   - 選取需要的選項：
     - **給予提示？** — 選取此選項以允許 DTP 用於步驟中。在規則中使用範本或在規劃中啟動指令碼範本時，會在提示使用者輸入的範本步驟中顯示 DTP。
     - **強制？** — 選取此選項，使其強制為設計時期提示輸入資訊。
     - **唯讀** — 選取此選項讓使用者只能讀取提示。
     - **已隱藏** — 選擇此選項可隱藏使用者的設計時期提示。在 Planning 中，在表單設計中的**商業規則**頁籤中，將具有範本或指令碼範本的規則新增至表單。在**商業規則特性**面板中，選取**隱藏提示**，然後選取**使用表單上的成員**。
   - 在 **DTP 文字**中，輸入當系統提示使用者輸入時，您希望使用者看見的文字。如果您選取**給予提示？**，則需要 **DTP 文字**。
   - **選擇性。**在**註解**中，輸入關於 DTP 的註解。
   - **選擇性。**在**預設值**中，指派 DTP 的預設值。

     若要指派預設值，請按一下 ，然後選取 **DTP** 或**維度**，然後選取 DTP 或維度。
5. **選擇性。**在**相依性**頁籤中，選取設計時期提示，然後選取相依性選項：
   - **排除** — 讓設計時期提示 (例如，DTP2) 排除另一個設計時期提示 (例如 DTP1)。當系統提示您 DTP2 時，只會顯示未用於 DTP1 的維度。
   - **內含** — 讓設計時期提示 (例如，DTP4) 內含另一個設計時期提示 (例如 DTP3)。當系統提示您 DTP4 時，只會顯示用於 DTP3 的維度。
   - **無** — 設計時期提示未定義任何相依性。

   維度 DTP 可在「成員範圍」、「維度」或「跨維度」DTP 上擁有相依性。
6. **選擇性。**在**定義限制**頁籤中，執行下列動作：
   - 在**密度**中，選取選項：
     - **顯示稀疏** — 僅針對 DTP 顯示稀疏維度
     - **顯示密集** — 僅針對 DTP 顯示密集維度
     - **兩者** — 針對 DTP 顯示密集與稀疏維度
   - 在**維度類型**中，選取要限制 DPT 的維度類型，以僅顯示選取的維度類型。

     例如，如果您只希望使用者輸入 *Account*、*Entity* 與 *Year* 的成員，請在**維度類型**中選取這些維度。當系統提示使用者輸入時，顯示的唯一維度是類型 *Account*、*Entity* 以及 *Year*。

整數 DTP

定義整數。

若要建立整數設計時期提示，請執行下列動作：

1. 在**範本設計工具**中，開啟現有範本或建立新範本。
2. 在**設計時期提示**頁籤中，按一下 。
3. 如果您已針對範本定義設計時期提示，請選取輸入新設計時期提示的位置。
4. 在**建立提示**對話方塊中，在**特性**頁籤上，執行下列動作：
   - 輸入**名稱**，並選取**數值**做為**類型**。
   - 選取需要的選項：
     - **給予提示？** — 選取此選項以允許 DTP 用於步驟中。在規則中使用範本或在規劃中啟動指令碼範本時，會在提示使用者輸入的範本步驟中顯示 DTP。
     - **強制？** — 選取此選項，使其強制為設計時期提示輸入資訊。
     - **唯讀** — 選取此選項讓使用者只能讀取提示。
     - **已隱藏** — 選擇此選項可隱藏使用者的設計時期提示。在 Planning 中，在表單設計中的**商業規則**頁籤中，將具有範本或指令碼範本的規則新增至表單。在**商業規則特性**面板中，選取**隱藏提示**，然後選取**使用表單上的成員**。
   - 在 **DTP 文字**中，輸入當系統提示使用者輸入時，您希望使用者看見的文字。如果您選取**給予提示？**，則需要 **DTP 文字**。
   - **選擇性。**在**註解**中，輸入關於 DTP 的註解。
   - 選取**允許 #Missing** 以允許 DTP 中的 #Missing 值，而非整數。
   - 在**限制**中，按一下 ，然後定義您可以在設計時期提示中輸入的最小和最大數字。
   - **選擇性。**在**預設值**中，按一下 ，然後選取 **DTP** 值。

成員 DTP

從應用程式中的維度定義成員。

若要建立 Member 設計時期提示，請執行下列動作：

1. 在**範本設計工具**中，開啟現有範本或建立新範本。
2. 在**設計時期提示**頁籤中，按一下 。
3. 如果您已針對範本定義設計時期提示，請選取輸入新設計時期提示的位置。
4. 在**建立提示**對話方塊中，在**特性**頁籤上，執行下列動作：
   - 輸入**名稱**，並選取**成員**做為**類型**。
   - 選取需要的選項：
     - **給予提示？** — 選取此選項以允許 DTP 用於步驟中。在規則中使用範本或在規劃中啟動指令碼範本時，會在提示使用者輸入的範本步驟中顯示 DTP。
     - **強制？** — 選取此選項，使其強制為設計時期提示輸入資訊。

  - **唯讀** — 選取此選項讓使用者只能讀取提示。
  - **已隱藏** — 選擇此選項可隱藏使用者的設計時期提示。在 Planning 中，在表單設計中的**商業規則**頁籤中，將具有範本或指令碼範本的規則新增至表單。在**商業規則特性**面板中，選取**隱藏提示**，然後選取**使用表單上的成員**。
- 在 **DTP 文字**中，輸入當系統提示使用者輸入時，您希望使用者看見的文字。

  如果您選取**給予提示？**，則需要 **DTP 文字**。
- **選擇性。**在**註解**中，輸入關於 DTP 的註解。
- 在**維度**中，選取維度。

  「維度」下拉清單顯示來自目前應用程式的維度。

  在*圖形*範本中，選取維度是選擇性的。如果您選取維度，則無法使用**定義限制**頁籤。

  在*指令碼*範本中，您必須選取維度。
- **選擇性。**在**預設值**中，指派 DTP 的預設值。

  若要指派預設值，請按一下 ，然後選取**成員**，然後在**成員選取器**中選取成員。**維度**下拉清單中選取的屬性維度隨即顯示。展開此維度以顯示可用成員。
- **僅限指令碼範本。**在**限制**中，輸入傳回一組使用者可選取之成員的 DTP、成員或函式。

  例如，若要在提示時限制 DTP，以僅顯示父項「收入」下的層級 0 科目，請選取「科目」作為**維度**；然後在**限制**中，輸入 "ILvL0Descendants(Revenues)"。

  請注意： 針對「限制」使用函式時，建議使用 Planning 函式，而不要使用 Oracle Essbase 函式。在某些情況下，Essbase 函式不會傳回預期的成員，例如當評估的 Essbase 函式包括動態成員時。針對成員設計時期提示中的「限制」，請使用 "ILvl0Descendants("Mbr Name") "，而不使用 Essbase 函式 "@Relative("Mbr Name", 0)"。

5. **選擇性。**在**相依性**頁籤中，選取設計時期提示，然後選取相依性選項：
   - **排除** — 讓設計時期提示 (例如，DTP2) 排除另一個設計時期提示 (例如 DTP1)。當系統提示您 DTP2 時，只會顯示未用於 DTP1 的維度。
   - **內含** — 讓設計時期提示 (例如，DTP4) 內含另一個設計時期提示 (例如 DTP3)。當系統提示您 DTP4 時，只會顯示用於 DTP3 的維度。
   - **無** — 設計時期提示未定義任何相依性。
6. **選擇性。定義限制**頁籤僅適用於*圖形*範本，而且該頁籤只會在您未選取維度時才會顯示。如果您在圖形範本中，而且未選取維度，請執行下列動作：
   - 在**密度**中，選取選項：
     - **顯示稀疏** — 僅針對 DTP 顯示稀疏維度
     - **顯示密集** — 僅針對 DTP 顯示密集維度
     - **兩者** — 針對 DTP 顯示密集與稀疏維度
   - 在**維度類型**中，選取要限制 DPT 的維度類型，以僅顯示選取的維度類型。

     例如，如果您只希望使用者輸入 *Account*、*Entity* 與 *Year* 的成員，請在**維度類型**中選取這些維度。當系統提示使用者輸入時，顯示的唯一維度是類型 *Account*、*Entity* 以及 *Year*。

成員 DTP

從應用程式中選取的維度定義多個成員。

若要建立成員設計時期提示，請執行下列動作：

1. 在**範本設計工具**中，開啟現有範本或建立新範本。
2. 在**設計時期提示**頁籤中，按一下 。
3. 如果您已針對範本定義設計時期提示，請選取輸入新設計時期提示的位置。
4. 在**建立提示**對話方塊中，在**特性**頁籤上，執行下列動作：
   - 輸入**名稱**，並選取**成員**做為**類型**。
   - 選取需要的選項：
     - **給予提示？** — 選取此選項以允許 DTP 用於步驟中。在規則中使用範本或在規劃中啟動指令碼範本時，會在提示使用者輸入的範本步驟中顯示 DTP。
     - **強制？** — 選取此選項，使其強制為設計時期提示輸入資訊。
     - **唯讀** — 選取此選項讓使用者只能讀取提示。
     - **已隱藏** — 選擇此選項可隱藏使用者的設計時期提示。在 Planning 中，在表單設計中的**商業規則**頁籤中，將具有範本或指令碼範本的規則新增至表單。在**商業規則特性**面板中，選取**隱藏提示**，然後選取**使用表單上的成員**。
   - 在 **DTP 文字**中，輸入當系統提示使用者輸入時，您希望使用者看見的文字。

     如果您選取**給予提示？**，則需要 **DTP 文字**。
   - **選擇性。**在**註解**中，輸入關於 DTP 的註解。
   - 在**維度**中，選取維度。

     「維度」下拉清單顯示來自目前應用程式的維度。

     在*圖形*範本中，選取維度是選擇性的。如果您選取維度，則無法使用**定義限制**頁籤。

     在*指令碼*範本中，您必須選取維度。
   - **選擇性。**在**預設值**中，指派 DTP 的預設值。

     若要指派預設值，請按一下 ，然後選取**成員**，然後在**成員選取器**中選取成員。**維度**下拉清單中選取的屬性維度隨即顯示。展開此維度以顯示可用成員。
   - **僅限指令碼範本。**在**限制**中，輸入傳回一組使用者可選取之成員的 DTP、成員或函式。

     例如，若要在提示時限制 DTP，以僅顯示父項「收入」下的層級 0 科目，請選取「科目」作為**維度**；然後在**限制**中，輸入 "ILvl0Descendants(Revenues)"。

     請注意： 針對「限制」使用函式時，建議使用 Planning 函式，而不要使用 Oracle Essbase 函式。在某些情況下，Essbase 函式不會傳回預期的成員，例如當評估的 Essbase 函式包括動態成員時。針對成員設計時期提示中的「限制」，請使用 "ILvl0Descendants("Mbr Name") "，而不要使用 Essbase 函式 "@Relative("Mbr Name", 0)"。
5. **選擇性。**在**相依性**頁籤中，選取設計時期提示，然後選取相依性選項：
   - **排除** — 讓設計時期提示 (例如，DTP2) 排除另一個設計時期提示 (例如 DTP1)。當系統提示您 DTP2 時，只會顯示未用於 DTP1 的維度。

- **內含** — 讓設計時期提示 (例如，DTP4) 內含另一個設計時期提示 (例如 DTP3)。當系統提示您 DTP4 時，只會顯示用於 DTP3 的維度。
- **無** — 設計時期提示未定義任何相依性。

6. **選擇性。定義限制**頁籤僅適用於*圖形*範本，而且該頁籤只會在您未選取維度時才會顯示。如果您在圖形範本中，而且未選取維度，請執行下列動作：
   - 在**密度**中，選取選項：
     - **顯示稀疏** — 僅針對 DTP 顯示稀疏維度
     - **顯示密集** — 僅針對 DTP 顯示密集維度
     - **兩者** — 針對 DTP 顯示密集與稀疏維度
   - 在**維度類型**中，選取要限制 DPT 的維度類型，以僅顯示選取的維度類型。

     例如，如果您只希望使用者輸入 *Account*、*Entity* 與 *Year* 的成員，請在**維度類型**中選取這些維度。當系統提示使用者輸入時，顯示的唯一維度是類型 *Account*、*Entity* 以及 *Year*。

成員範圍 DTP

從應用程式中選取的維度定義成員範圍。

成員範圍 DTP 可具有來自每個維度的一或多個成員。例如，您可以使用「成員範圍」類型 DTP 以建立您要求使用者為規則輸入成員的檢視點步驟。

若要建立成員範圍設計時期提示，請執行下列動作：

1. 在**範本設計工具**中，開啟現有範本或建立新範本。
2. 在**設計時期提示**頁籤中，按一下 。
3. 如果您已針對範本定義設計時期提示，請選取輸入新設計時期提示的位置。
4. 在**建立提示**對話方塊中，在**特性**頁籤上，執行下列動作：
   - 輸入**名稱**，並選取**成員範圍**做為**類型**。
   - 選取需要的選項：
     - **給予提示？** — 選取此選項以允許 DTP 用於步驟中。在規則中使用範本或在規劃中啟動指令碼範本時，會在提示使用者輸入的範本步驟中顯示 DTP。
     - **強制？** — 選取此選項，使其強制為設計時期提示輸入資訊。
     - **唯讀** — 選取此選項讓使用者只能讀取提示。
     - **已隱藏** — 選擇此選項可隱藏使用者的設計時期提示。在 Planning 中，在表單設計中的**商業規則**頁籤中，將具有範本或指令碼範本的規則新增至表單。在**商業規則特性**面板中，選取**隱藏提示**，然後選取**使用表單上的成員**。
   - 在 **DTP 文字**中，輸入當系統提示使用者輸入時，您希望使用者看見的文字。

     如果您選取**給予提示？**，則需要 **DTP 文字**。
   - **選擇性。**在**註解**中，輸入關於 DTP 的註解。
   - **選擇性。**在**預設值**中，指派 DTP 的預設值。

     若要指派預設值，請執行下列其中一項動作：
     - 按一下 ，然後在每個維度旁邊按一下 ，然後選取 **DTP**、**成員**或**函式**。

  - 按一下 ，以使用 DTP 作為預設值。

5. **選擇性。**在**相依性**頁籤中，選取設計時期提示，然後選取相依性選項：
   - **排除** — 讓設計時期提示 (例如，DTP2) 排除另一個設計時期提示 (例如 DTP1)。當系統提示您 DTP2 時，只會顯示未用於 DTP1 的維度。
   - **內含** — 讓設計時期提示 (例如，DTP4) 內含另一個設計時期提示 (例如 DTP3)。當系統提示您 DTP4 時，只會顯示用於 DTP3 的維度。
   - **無** — 設計時期提示未定義任何相依性。

   成員範圍 DTP 可以取決於以下類型的 DTP：成員範圍、跨維度、單一維度，以及多個維度。
6. **選擇性。**在**定義限制**頁籤中，執行下列動作：
   - 在**密度**中，選取選項：
     - **顯示稀疏** — 僅針對 DTP 顯示稀疏維度
     - **顯示密集** — 僅針對 DTP 顯示密集維度
     - **兩者** — 針對 DTP 顯示密集與稀疏維度
   - 在**維度類型**中，選取要限制 DPT 的維度類型，以僅顯示選取的維度類型。

     例如，如果您只希望使用者輸入 *Account*、*Entity* 與 *Year* 的成員，請在**維度類型**中選取這些維度。當系統提示使用者輸入時，顯示的唯一維度是類型 *Account*、*Entity* 以及 *Year*。

數值 DTP

定義數字。例如，您可以在公式中使用數值 DTP。

若要建立數值設計時期提示，請執行下列動作：

1. 在**範本設計工具**中，開啟現有範本或建立新範本。
2. 在**設計時期提示**頁籤中，按一下 。
3. 如果您已針對範本定義設計時期提示，請選取輸入新設計時期提示的位置。
4. 在**建立提示**對話方塊中，在**特性**頁籤上，執行下列動作：
   - 輸入**名稱**，並選取**數值**做為**類型**。
   - 選取需要的選項：
     - **給予提示？** — 選取此選項以允許 DTP 用於步驟中。在規則中使用範本或在規劃中啟動指令碼範本時，會在提示使用者輸入的範本步驟中顯示 DTP。
     - **強制？** — 選取此選項，使其強制為設計時期提示輸入資訊。
     - **唯讀** — 選取此選項讓使用者只能讀取提示。
     - **已隱藏** — 選擇此選項可隱藏使用者的設計時期提示。在 Planning 中，在表單設計中的**商業規則**頁籤中，將具有範本或指令碼範本的規則新增至表單。在**商業規則特性**面板中，選取**隱藏提示**，然後選取**使用表單上的成員**。
   - 在 **DTP 文字**中，輸入當系統提示使用者輸入時，您希望使用者看見的文字。

     如果您選取**給予提示？**，則需要 **DTP 文字**。
   - **選擇性。**在**註解**中，輸入關於 DTP 的註解。

- 選取**允許 #Missing** 以允許 DTP 中的 #Missing 值，而非數字。
- 在**限制**中，按一下 ，然後定義您可以在設計時期提示中輸入的最小和最大數字。
- **選擇性。**在**預設值**中，按一下 ，然後選取 **DTP** 值。

密碼 DTP

定義密碼以用於設計階段提示。

若要建立密碼設計時期提示，請執行下列動作：

1. 在**範本設計工具**中，開啟現有範本或建立新範本。
2. 在**設計時期提示**頁籤中，按一下 。
3. 如果您已針對範本定義設計時期提示，請選取輸入新設計時期提示的位置。
4. 在**建立提示**對話方塊中，在**特性**頁籤上，執行下列動作：
   - 輸入**名稱**，並選取**密碼**做為**類型**。
   - 選取需要的選項：
     - **給予提示？** — 選取此選項以允許 DTP 用於步驟中。在規則中使用範本或在規劃中啟動指令碼範本時，會在提示使用者輸入的範本步驟中顯示 DTP。
     - **強制？** — 選取此選項，使其強制為設計時期提示輸入資訊。
     - **唯讀** — 選取此選項讓使用者只能讀取提示。
     - **已隱藏** — 選擇此選項可隱藏使用者的設計時期提示。在 Planning 中，在表單設計中的**商業規則**頁籤中，將具有範本或指令碼範本的規則新增至表單。在**商業規則特性**面板中，選取**隱藏提示**，然後選取**使用表單上的成員**。
   - 在 **DTP 文字**中，輸入當系統提示使用者輸入時，您希望使用者看見的文字。

     如果您選取**給予提示？**，則需要 **DTP 文字**。
   - **選擇性。**在**註解**中，輸入關於 DTP 的註解。
   - **選擇性。**在**預設值**中，輸入密碼值。

百分比 DTP

定義百分比。

若要建立百分比設計時期提示，請執行下列動作：

1. 在**範本設計工具**中，開啟現有範本或建立新範本。
2. 在**設計時期提示**頁籤中，按一下 。
3. 如果您已針對範本定義設計時期提示，請選取輸入新設計時期提示的位置。
4. 在**建立提示**對話方塊中，在**特性**頁籤上，執行下列動作：
   - 輸入**名稱**，並選取**百分比**做為**類型**。
   - 選取需要的選項：
     - **給予提示？** — 選取此選項以允許 DTP 用於步驟中。在規則中使用範本或在規劃中啟動指令碼範本時，會在提示使用者輸入的範本步驟中顯示 DTP。

  - **強制？** — 選取此選項，使其強制為設計時期提示輸入資訊。
  - **唯讀** — 選取此選項讓使用者只能讀取提示。
  - **已隱藏** — 選擇此選項可隱藏使用者的設計時期提示。在 Planning 中，在表單設計中的**商業規則**頁籤中，將具有範本或指令碼範本的規則新增至表單。在**商業規則特性**面板中，選取**隱藏提示**，然後選取**使用表單上的成員**。
- 在 **DTP 文字**中，輸入當系統提示使用者輸入時，您希望使用者看見的文字。

  如果您選取**給予提示？**，則需要 **DTP 文字**。
- **選擇性。**在**註解**中，輸入關於 DTP 的註解。
- 選取**允許 #Missing** 以允許 DTP 中的 #Missing 值，而非百分比。
- 在**限制**中，按一下 ，然後定義您可以在設計時期提示中輸入的最小和最大數字。
- **選擇性。**在**預設值**中，按一下 ，然後選取 **DTP** 值。

限制清單 DTP

定義限制清單。

例如，在「配置」系統範本中，您可以選取進位方法，然後使用「限制清單」設計時期提示以選擇要用於範本的進位類型。

在範本的指令碼中，您可以為每種進位建立指令碼，並且只有在使用者選取進位選項時才使用該指令碼。

若要建立「限制清單」設計時期提示，請執行下列動作：

1. 在**範本設計工具**中，開啟現有範本或建立新範本。
2. 在**設計時期提示**頁籤中，按一下 。
3. 如果您已針對範本定義設計時期提示，請選取輸入新設計時期提示的位置。
4. 在**建立提示**對話方塊中，在**特性**頁籤上，執行下列動作：
   - 輸入**名稱**，並選取**限制清單**做為**類型**。
   - 選取需要的選項：
     - **給予提示？** — 選取此選項以允許 DTP 用於步驟中。在規則中使用範本或在規劃中啟動指令碼範本時，會在提示使用者輸入的範本步驟中顯示 DTP。
     - **強制？** — 選取此選項，使其強制為設計時期提示輸入資訊。
     - **唯讀** — 選取此選項讓使用者只能讀取提示。
     - **已隱藏** — 選擇此選項可隱藏使用者的設計時期提示。在 Planning 中，在表單設計中的**商業規則**頁籤中，將具有範本或指令碼範本的規則新增至表單。在**商業規則特性**面板中，選取**隱藏提示**，然後選取**使用表單上的成員**。
   - 在 **DTP 文字**中，輸入當系統提示使用者輸入時，您希望使用者看見的文字。

     如果您選取**給予提示？**，則需要 **DTP 文字**。
   - **選擇性。**在**註解**中，輸入關於 DTP 的註解。
5. 選取**限制清單**頁籤，然後輸入**規則建置器值**與**取代值**
6. **選擇性。**返回**特性**頁籤，並輸入要在設計時期提式中顯示的預設值。

若要定義預設值，請按一下**預設值**下拉清單，然後選取您在**限制清單**頁籤中輸入的**規則建置器值**。

分隔符號 DTP

分隔符號是範本步驟中的間隔或空格字元。

若要建立分隔符號設計時期提示，請執行下列動作：

1. 在**範本設計工具**中，開啟現有範本或建立新範本。
2. 在**設計時期提示**頁籤中，按一下 。
3. 如果您已針對範本定義設計時期提示，請選取輸入新設計時期提示的位置。
4. 在**建立提示**對話方塊中，在**特性**頁籤上，執行下列動作：
   - 輸入**名稱**，並選取**分隔符號**做為**類型**。
   - 選取需要的選項：
     - **給予提示？** — 選取此選項以允許 DTP 用於步驟中。在規則中使用範本或在規劃中啟動指令碼範本時，會在提示使用者輸入的範本步驟中顯示 DTP。
     - **強制？** — 選取此選項，使其強制為設計時期提示輸入資訊。
     - **唯讀** — 選取此選項讓使用者只能讀取提示。
     - **已隱藏** — 選擇此選項可隱藏使用者的設計時期提示。在 Planning 中，在表單設計中的**商業規則**頁籤中，將具有範本或指令碼範本的規則新增至表單。在**商業規則特性**面板中，選取**隱藏提示**，然後選取**使用表單上的成員**。
   - 在 **DTP 文字**中，輸入當系統提示使用者輸入時，您希望使用者看見的文字。

     如果您選取**給予提示？**，則需要 **DTP 文字**。
   - **選擇性。**在**註解**中，輸入關於 DTP 的註解。

智慧型列示 DTP

定義智慧型列示以用於設計階段提示。

若要建立「智慧型列示」設計時期提示，請執行下列動作：

1. 在**範本設計工具**中，開啟現有範本或建立新範本。
2. 在**設計時期提示**頁籤中，按一下 。
3. 如果您已針對範本定義設計時期提示，請選取輸入新設計時期提示的位置。
4. 在**建立提示**對話方塊中，在**特性**頁籤上，執行下列動作：
   - 輸入**名稱**，並選取**智慧型列示**做為**類型**。
   - 選取需要的選項：
     - **給予提示？** — 選取此選項以允許 DTP 用於步驟中。在規則中使用範本或在規劃中啟動指令碼範本時，會在提示使用者輸入的範本步驟中顯示 DTP。
     - **強制？** — 選取此選項，使其強制為設計時期提示輸入資訊。
     - **唯讀** — 選取此選項讓使用者只能讀取提示。

  - **已隱藏** — 選擇此選項可隱藏使用者的設計時期提示。在 Planning 中，在表單設計中的**商業規則**頁籤中，將具有範本或指令碼範本的規則新增至表單。在**商業規則特性**面板中，選取**隱藏提示**，然後選取**使用表單上的成員**。
- 在 **DTP 文字**中，輸入當系統提示使用者輸入時，您希望使用者看見的文字。

  如果您選取**給予提示？**，則需要 **DTP 文字**。
- **選擇性。**在**註解**中，輸入關於 DTP 的註解。

5. 在**智慧型列示**中，按下拉清單，然後選取「智慧型列示」。

   顯示的智慧型列示將從規劃填入。如果規劃中沒有智慧型列示，則下拉選單為空白。
6. **選擇性。**在**預設值**中，按下拉清單，然後選取預設值。

StringAsNumber DTP

將字串定義為數字。

若要建立 StringAsNumber 設計時期提示，請執行下列動作：

1. 在**範本設計工具**中，開啟現有範本或建立新範本。
2. 在**設計時期提示**頁籤中，按一下 。
3. 如果您已針對範本定義設計時期提示，請選取輸入新設計時期提示的位置。
4. 在**建立提示**對話方塊中，在**特性**頁籤上，執行下列動作：
   - 輸入**名稱**，並選取 **StringAsNumber** 做為**類型**。
   - 選取需要的選項：
     - **給予提示？** — 選取此選項以允許 DTP 用於步驟中。在規則中使用範本或在規劃中啟動指令碼範本時，會在提示使用者輸入的範本步驟中顯示 DTP。
     - **強制？** — 選取此選項，使其強制為設計時期提示輸入資訊。
     - **唯讀** — 選取此選項讓使用者只能讀取提示。
     - **已隱藏** — 選擇此選項可隱藏使用者的設計時期提示。在 Planning 中，在表單設計中的**商業規則**頁籤中，將具有範本或指令碼範本的規則新增至表單。在**商業規則特性**面板中，選取**隱藏提示**，然後選取**使用表單上的成員**。
   - 在 **DTP 文字**中，輸入當系統提示使用者輸入時，您希望使用者看見的文字。

     如果您選取**給予提示？**，則需要 **DTP 文字**。
   - **選擇性。**在**註解**中，輸入關於 DTP 的註解。
   - 選取**允許 #Missing** 以允許 DTP 中的 #Missing 值，而非字串。
   - **選擇性。**選取**使用系統預設值**，然後在**預設值**中，按下拉清單以選取要用於提示中的預設值。預設值最初填入使用者名稱。

> **備註：**
>
> 如果您選取**使用系統預設值**，並從預設值中選取「使用者名稱」，則您必須選取**給予提示**，然後在 **DTP 文字**中輸入文字。將此設計時期提示新增至步驟中。在規則中，當系統提示您輸入此設計時期提示時，請選取 StringAsNumber 變數，此變數以**使用系統預設值**和使用者名稱作為預設值。

字串 DTP

定義文字字串。

若要建立字串設計時期提示，請執行下列動作：

1. 在**範本設計工具**中，開啟現有範本或建立新範本。
2. 在**設計時期提示**頁籤中，按一下 。
3. 如果您已針對範本定義設計時期提示，請選取輸入新設計時期提示的位置。
4. 在**建立提示**對話方塊中，在**特性**頁籤上，執行下列動作：
   - 輸入**名稱**，並選取**字串**做為**類型**。
   - 選取需要的選項：
     - **給予提示？** — 選取此選項以允許 DTP 用於步驟中。在規則中使用範本或在規劃中啟動指令碼範本時，會在提示使用者輸入的範本步驟中顯示 DTP。
     - **強制？** — 選取此選項，使其強制為設計時期提示輸入資訊。
     - **唯讀** — 選取此選項讓使用者只能讀取提示。
     - **已隱藏** — 選擇此選項可隱藏使用者的設計時期提示。在 Planning 中，在表單設計中的**商業規則**頁籤中，將具有範本或指令碼範本的規則新增至表單。在**商業規則特性**面板中，選取**隱藏提示**，然後選取**使用表單上的成員**。
   - 在 **DTP 文字**中，輸入當系統提示使用者輸入時，您希望使用者看見的文字。

     如果您選取**給予提示？**，則需要 **DTP 文字**。
   - **選擇性。**在**註解**中，輸入關於 DTP 的註解。
   - **選擇性。**在**預設值**中，按一下 ，然後選取 **DTP**、**成員**或**函式**，然後輸入 DTP、成員或函式作為預設值。

UDA DTP

定義使用者定義的屬性，以用於設計階段提示。

例如，在規劃中，「科目」類型建立為 UDA 並且用在「幣別轉換」範本中。*費用*與*收入*科目已使用*平均費率*轉換，而*資產*、*負債*和*權益*科目通常會使用*期末費率*轉換。

若要建立 UDA 設計時期提示，請執行下列動作：

1. 在**範本設計工具**中，開啟現有範本或建立新範本。
2. 在**設計時期提示**頁籤中，按一下 。
3. 如果您已針對範本定義設計時期提示，請選取輸入新設計時期提示的位置。
4. 在**建立提示**對話方塊中，在**特性**頁籤上，執行下列動作：
   - 輸入**名稱**，並選取 **UDA** 做為**類型**。
   - 選取需要的選項：
     - **給予提示？** — 選取此選項以允許 DTP 用於步驟中。在規則中使用範本或在規劃中啟動指令碼範本時，會在提示使用者輸入的範本步驟中顯示 DTP。
     - **強制？** — 選取此選項，使其強制為設計時期提示輸入資訊。

  - **唯讀** — 選取此選項讓使用者只能讀取提示。
  - **已隱藏** — 選擇此選項可隱藏使用者的設計時期提示。在規劃中，在表單設計中的**商業規則**頁籤中，將具有範本或指令碼範本的規則新增至表單。在**商業規則特性**面板中，選取**隱藏提示**，然後選取**使用表單上的成員**。
- 在 **DTP 文字**中，輸入當系統提示使用者輸入時，您希望使用者看見的文字。

  如果您選取**給予提示？**，則需要 **DTP 文字**。
- **選擇性。**在**註解**中，輸入關於 DTP 的註解。
- 從**維度**下拉清單中，選取維度。

  顯示的維度是規劃應用程式中的維度。
- **選擇性。**在**預設值**中，按下拉清單，然後根據您選取的維度選取預設值。

5. 在**相依性**頁籤中，定義相依性。請參閱為設計時期提示定義相依性。
6. 在**定義限制**頁籤中，選取**密度**和**維度類型**。請參閱為設計時期提示定義限制

為設計時期提示定義相依性

您可以為*跨維度*、*多個維度*、*單一維度*、*單一成員*、*多個成員*和*成員範圍*設計時期提示，定義內含與排除相依性。如果您指定提示 (對於範例 DTP1) 作為另一個提示 (對於範例 DTP2) 的內含項，向使用者提示 DTP1 時，只會顯示來自 DTP2 的維度。如果您指定提示 (對於範例 DTP3) 作為另一個提示 (對於範例 DTP4) 的排除項，向使用者提示 DTP3 時，只會顯示 DTP4 未使用的維度。

> **備註：**
>
> 設計時期提示在設計時期提示方格中只能為設計時期提示之前的內含項或排除項，因此，設計時期提示的順序是非常重要的。

若要定義相依性，請執行下列動作：

1. 在**系統檢視**中，建立或開啟自訂範本。
2. 在**設計時期提示**頁籤上，按一下 ，然後選取要插入列的位置。
3. 在**建立提示**對話方塊的**特性**頁籤上，於**類型**欄位中選取下列其中一個選項：
   - **跨維度**
   - **維度**
   - **維度**
   - **成員**
   - **成員**
   - **成員範圍**
   - **UDA**

   當您選取這其中一個選項時，便會顯示**相依性**頁籤。
4. 在**相依性**頁籤中，選取設計時期提示，然後選取相依性選項：

- **排除** — 讓設計時期提示 (例如，DTP2) 排除另一個設計時期提示 (例如 DTP1)。當系統提示您 DTP2 時，只會顯示未用於 DTP1 的維度。
- **內含** — 讓設計時期提示 (例如，DTP4) 內含另一個設計時期提示 (例如 DTP3)。當系統提示您 DTP4 時，只會顯示用於 DTP3 的維度。
- **無** — 設計時期提示未定義任何相依性。

5. 按一下**確定**，然後按一下 (**儲存**按鈕)。

為設計時期提示定義限制

為自訂範本建立*跨維度*、*單一維度*、*多個維度*、*單一成員*、*多個成員*或*成員範圍*設計時期提示時，必須指定您是要同時對密集維度和稀疏維度顯示提示，還是僅對密集維度或稀疏維度顯示提示。

若要定義限制，請執行下列動作：

1. 在**系統檢視**中，建立或開啟自訂範本。
2. 在**設計時期提示**頁籤上，按一下 ，然後選取要插入列的位置。
3. 在**建立提示**對話方塊的**特性**頁籤上，於**類型**欄位中選取下列其中一個選項：
   - **跨維度**
   - **單一維度**
   - **多個維度**
   - **單一成員**
   - **多個成員**
   - **成員範圍**
   - **UDA**

   當您選取這其中一個選項時，便會顯示**定義限制**頁籤。
4. 在**定義限制**頁籤上：
   - 在**密度**中，選取選項：
     - **顯示稀疏** — 僅針對 DTP 顯示稀疏維度
     - **顯示密集** — 僅針對 DTP 顯示密集維度
     - **兩者** — 針對 DTP 顯示密集與稀疏維度
   - 在**維度類型**中，選取要限制設計時期提示的維度類型，以僅顯示選取的維度類型。

     例如，如果您只希望使用者輸入 *Account*、*Entity* 與 *Year* 的成員，請在**維度類型**中選取這些維度。當系統提示使用者輸入時，顯示的唯一維度是類型 *Account*、*Entity* 以及 *Year*。
5. 按一下**確定**，然後按一下 (**儲存**按鈕)。

為設計時期提示建立步驟

使用「範本精靈設計工具」建立範本的步驟。精靈可讓您決定要在每個步驟顯示的設計時期提示，以及是否顯示或隱藏步驟 (啟用條件)，或是否要顯示錯誤或警告訊息 (驗證條件)。

若要建立設計時期提示的步驟，請執行下列動作：

1. 在**設計時期提示**頁籤中，按一下 。
2. 在**範本精靈設計工具**中，按一下 ，以建立精靈中的步驟。
3. 在**新增步驟**對話方塊中，輸入關於步驟的資訊，然後按一下**確定**。

   您新增的步驟會顯示在**範本精靈設計工具**中的**步驟**下拉清單。
4. 移動設計時期提示，以顯示從**可用 DTP** 到**選取的 DTP** 的步驟。
5. 在**啟用條件**頁籤中，輸入有關要顯示或隱藏步驟的資訊：
   - 在**條件前綴**中，從下拉清單選取前綴。
   - 在 **DTP** 中，按一下 ，以選取 DTP 或函式。
   - 在**運算子**中，從下拉清單選取運算子。
   - 在**值**中，輸入或選取條件的值。

   重複這些步驟，到定義了條件中的全部陳述式為止。若要新增列，按一下前一列中的加號圖示 (+)。

   條件中的第一列是 IF 陳述式，每個附加的列都是一個 AND 陳述式。例如，

   每一列預設為 AND 陳述式，但是，如果您按一下 AND，則可以變更為 OR。
6. 在**驗證條件**頁籤中，輸入關於當使用者輸入不需要 (錯誤) 的資料或可能不需要 (警告) 的資料時，是否要顯示錯誤或警告訊息：
   - 在**驗證條件**中，按一下 ，以定義驗證條件。
   - 在**訊息層級**中，選取**錯誤**或**警告**。

     錯誤訊息可防止下一步驟。警告訊息可讓您在警告訊息中按一下**確定**之後，進行下一步驟。
   - 在**驗證訊息**中，輸入將在步驟中向使用者顯示的訊息。

   按一下加號圖示 (+)，以將其他錯誤或警告新增至步驟。
7. 按一下**確定**，然後按一下 。

尋找和取代設計時期提示中的文字

您可以搜尋和取代自訂範本之設計時期提示中的文字字串。

> **注意：**
>
> 您取代設計時期提示的順序和位置非常重要。如果您要同時在「設計時期提示」頁籤與「範本設計工具」頁籤中尋找和取代設計時期提示名稱，必須先在「設計時期提示」頁籤中進行搜尋並儲存範本，然後在「範本設計工具」頁籤中進行尋找和取代。如果您不要依此順序執行搜尋，將不會在「範本設計工具」頁籤中取代設計時期提示名稱。

若要在設計時期提示中搜尋和取代文字，請執行下列動作：

1. 在**系統檢視**中，開啟自訂範本。

2. 在**設計時期提示**頁籤中，按一下 ，輸入要尋找和取代的資訊，然後按一下**取代**或**全部取代**。

開啟自訂範本

您可以從系統檢視、部署檢視和自訂檢視中開啟自訂範本。

您也可以在「規則設計工具」中，從商業規則的流程圖中開啟自訂範本。

若要開啟自訂範本，按兩下此範本，或是用滑鼠右鍵按一下範本，然後選取**開啟**。

重新整理自訂範本

建立自訂範本之後，您可能必須重新整理系統檢視中的應用程式清單，才能查看範本節點中的新範本。

當您重新整理自訂範本所屬的應用程式類型、應用程式、計算或計畫類型時，依照預設會重新整理**範本**節點。然而，重新整理**範本**節點，並不會重新整理應用程式清單中的較高層級 (亦即，計算或計畫類型、應用程式或應用程式類型)。

若要重新整理自訂範本的清單，用滑鼠右鍵按一下**範本**節點，然後選取**重新整理**。

顯示自訂範本的使用情況

您可以顯示使用自訂範本的商業規則清單。

當您想要刪除自訂範本，並且需要知道哪些物件 (如果有) 正在使用該範本時，檢視自訂範本的使用情況將會非常有用。您必須先將範本從所有使用它的物件中移除，然後才可以刪除該範本。

若要顯示自訂範本的使用情況，在**系統檢視**中，展開**範本**節點，用滑鼠右鍵按一下範本，然後選取**顯示使用方法**。

複製和貼上自訂範本

您可以複製自訂範本，並貼到其他商業規則中。

作為複製目的地的規則必須屬於相同的計畫類型。例如，您可以將自訂範本，從 Planning 應用程式中 Plan1 計畫類型的規則，複製到 Planning 應用程式中 Plan1 計畫類型的另一個規則，而不能複製到「資本資產」計畫類型的規則中。

若要複製並貼上自訂範本，請執行下列其中一項動作：

- 展開**範本**節點，然後用滑鼠右鍵按一下範本，選取**複製到**，在**複製到**對話方塊中輸入必要資訊，然後按一下**確定**。
- 在商業規則的流程圖中，用滑鼠右鍵按一下要複製的範本，然後選取**複製**。開啟您要貼上範本的商業規則，用滑鼠右鍵按一下商業規則的流程圖，然後選取**貼上**。

刪除自訂範本

您可以從系統檢視、自訂檢視及部署檢視刪除自訂範本。

刪除自訂範本之前，請確定沒有任何商業規則或任何自訂資料夾正在使用它。若要顯示範本使用情況，請用滑鼠右鍵按一下範本，然後選取**顯示使用情況**。

若要刪除自訂範本，請展開**範本**節點，用滑鼠右鍵按一下範本，然後選取**刪除**。

> **備註：**
>
> 如果指令碼範本已部署至 Planning，當您選取**刪除**時，會顯示下列訊息：
>
> 部分選取的物件已部署至規劃。要從規劃伺服器刪除選取的項目嗎？您是否希望從規劃伺服器刪除它們？
>
> 按一下方塊會同時從規劃和 Calculation Manager 刪除指令碼範本。

尋找和取代圖形自訂範本中的文字

您可搜尋自訂範本中的文字字串。您也可以取代文字字串的所有例項。

當您搜尋字串時，Calculation Manager 會從在範本的流程圖中選取的元件之後的第一個元件開始搜尋，並繼續至流程圖的尾端，然後從流程圖的開頭開始，直到到達流程圖中選取的元件為止。在找到文字字串的符合項目後，如果您要再搜尋另一個字串，則必須重新開始搜尋，以找到下一個符合項目。

若要搜尋文字或是搜尋和取代自訂範本中的文字，請執行下列動作：

1. 在**系統檢視**中，開啟自訂範本。
2. 請執行下列其中一項動作：
   - 在**範本設計工具**頁籤上，選取**編輯**，然後選取**尋找**。在**尋找**對話方塊中，輸入要尋找的文字，然後按一下**尋找**。
   - 在**範本設計工具**頁籤上，選取**編輯**，然後選取**全部取代**。在**取代**對話方塊中，輸入要尋找和取代的資訊，然後選取**全部取代**

7

使用元件以設計商業規則和範本

另請參閱：

- 關於元件
  商業規則與範本可包含多種元件類型。
- 公式元件
  公式元件由公式計算陳述式組成。
- 指令碼元件
  指令碼元件可以在商業規則或範本中使用。
- 條件元件
  條件元件由條件陳述式組成，即 True 或 False。
- 成員區塊元件
  成員區塊元件定義在指令碼中括住一或多個陳述式的成員。
- 成員範圍元件
  成員範圍元件是由 Planning 維度中的各種成員組成的迴圈類型。
- 固定迴圈元件
  固定迴圈元件是以固定次數在中繼資料成員清單循環的物件。
- 中繼資料迴圈元件
  中繼資料迴圈元件可讓您使用函式 (中繼資料) 或開始與結束值 (固定)，指派值給多個成員。
- DTP 指派元件
  使用 DTP 指派元件以將 DTP、成員、函式，或輸入文字指派至自訂範本中的設計時期提示。
- 共用指令碼和公式元件
  共用公式或指令碼元件存在於公式層級和指令碼層級。
- 複製元件
  複製並貼上商業規則項目的子項，或複製參照，並將其貼到商業規則公式或指令碼元件。
- 儲存元件
  在「元件設計工具」中設計公式元件和指令碼元件後，您需要儲存這些元件。
- 重新整理公式元件和指令碼元件
  建立公式元件或指令碼元件後，您必須重新整理「系統檢視」中的應用程式清單，才能在「公式」或「指令碼」節點中看到該項目。
- 顯示公式元件和指令碼元件的使用情況
  您可以顯示指令碼或公式元件的使用情況。
- 在流程圖中使用元件
  您可以在流程圖中的元件上執行動作。

關於元件

商業規則與範本可包含多種元件類型。

- **公式元件** — 可以使用成員、函式和條件陳述式編寫或設計的計算陳述式

- **指令碼元件** — 僅限計算指令碼陳述式。
- **條件元件** — 包含 True 或 False 的條件陳述式 (也就是 If...Then 陳述式)。
- **成員區塊元件** — 包含您指定的某個成員。
- **成員範圍元件 (或中繼資料迴圈)** — 包含中繼資料成員的清單 (如科目清單)。
- **固定迴圈元件** — 包含中繼資料迴圈，例如通過科目等的成員清單迴圈。
- **中繼資料迴圈元件** — (僅限自訂範本使用者) 包含您可套用至父項維度之子項的參數。
- **DTP 指派元件** — (僅限自訂範本使用者) 包含您為其定義的設計時期提示和條件。

公式和指令碼元件的建立，可以單獨進行，與使用它們的規則和範本彼此獨立。由於它們是獨立物件，因此可以在「系統檢視」中開啟、儲存、編輯、刪除和匯出。

其他元件類型不同於公式和指令碼元件，必須在規則和範本中建立。您無法在所屬的規則和範本之外，單獨將它們開啟、儲存、刪除或匯出。

> **備註：**
>
> 當您建立元件時，您可能希望保留您正使用開啟的商業規則、元件、範本以及變數。Calculation Manager 會在頁籤式介面中顯示這些物件，以便您可以在建立元件時，輕鬆地在這些頁籤之間移動。您可以在 Calculation Manager 內擁有開啟多達十個標籤，但是 Oracle 建議您不要同時開啟超過十個以上的物件，以達到最佳效能。

公式元件

公式元件由公式計算陳述式組成。

另請參閱：

- 關於公式元件
- 建立公式元件
- 設計公式元件
- 開啟公式元件
- 編輯公式元件
- 刪除公式元件
- 複製和貼上公式元件

關於公式元件

公式元件由公式計算陳述式組成。若要為公式建立計算陳述式，必須輸入或選取成員、函式和條件。在建立公式時，該公式的每個計算陳述式都會列在「元件設計工具」方格的列中。

建立公式元件

您可以從「系統檢視」建立公式元件。公式元件可以在商業規則或範本中使用。

若要建立公式元件，請執行下列動作：

1. 在**系統檢視**中，按一下**新增物件**圖示。
2. 輸入**應用程式類型**。
3. 選取**應用程式**。應用程式必須是有效的 Planning 應用程式。
4. 選取**計畫類型**。

> **備註：**
>
> 在「系統檢視」下，如果您用滑鼠右鍵按一下「公式」並選取「新建」以建立新公式，「新增公式」對話方塊會填入應用程式類型名稱、應用程式名稱，以及您正在使用的計畫類型名稱。

5. 選取**公式**作為**物件類型**。
6. 輸入公式名稱，然後按一下**確定**。

   此時公式會顯示在「元件設計工具」中。若要設計公式元件，請參閱設計公式元件。

設計公式元件

您可以從「系統檢視」或任何其他檢視建立公式元件。您也可以在設計規則或範本時，在規則或範本設計工具中建立公式元件。公式元件可以在商業規則或範本中使用。

若要建立公式元件，請執行下列動作：

1. 在**系統檢視**中，在**公式**上按一下右鍵，然後選取**新增**。

   新增公式對話方塊中會填入規劃應用程式類型、規劃應用程式，以及計畫類型。

> **備註：**
>
> 您也可以透過將新公式元件拖放到商業規則或範本流程圖，在商業規則或範本中建立公式元件。

2. 輸入名稱、應用程式類型、應用程式，以及公式元件所屬的計畫類型。
3. 按一下**確定**。
4. **選擇性：**在**元件設計工具**上，您可以執行下列其中一項任務：
   - 按一下**新增方格**，然後選取**插入方格 (之前)**、**插入方格 (之後)**，或**在末端插入方格**，以新增其他方格。依預設，「公式」頁籤會顯示一個方格。
   - 按一下**刪除方格**以刪除選取的方格。
   - 按一下**尋找**圖示，以在您作業中的公式方格中尋找文字。請執行下列任務：
   - 按一下**尋找和取代**圖示，以尋找並取代指令碼中的文字。
5. 在**公式**上，輸入公式的標題。
6. **僅限為範本建立公式元件的使用者：**如果您想使用公式元件中的設計時期提示，請在**公式**頁籤上，選取**使用設計提示**。然後在條件方格中，透過執行下列任務，定義設計時期提示的條件：

   a. 在 **DTP** 中，選取設計時期提示。

b. 在**運算子**中，選取運算子： = = 或 <>。

c. 在**值**中，從下拉清單中選取值。

d. 重複這些步驟，根據需要建立任意數量的陳述式。

> **提示：**
>
> 按一下加號 (+) 和減號 (-) 圖示，在方格中新增和刪除列。按一下 And 旁的欄位，將 And 變更為 Or；從下拉清單中選取 Or。

7. 根據預設，公式元件計算的處理會從您在方格中輸入的第一個成員開始。如果希望從其他成員開始處理公式元件計算，請在**成員區塊**中輸入該成員或函式的名稱，或按一下**省略符號**圖示，從「成員選取器」中選取一個成員。請參閱關於向元件新增成員和函式。

8. 在**備註**中，為要建立的條件陳述式和公式陳述式輸入備註。

9. **選擇性**：若要為公式元件建立條件陳述式 (即 IF 陳述式)，請按一下**新增條件**。請參閱使用條件建置器建立條件陳述式。

10. **選擇性**：在顯示的 IF 列中，輸入條件陳述式的文字，或按一下該列右欄的**新增/編輯**條件圖示，以存取「條件建置器」。「條件建置器」可讓您以圖形方式設計條件陳述式。

> **備註：**
>
> 雖然可以從向下箭頭中選取 IF、ELSE IF 和 ELSE，但在預設情況下，第一項陳述必須是 IF 陳述式。

11. **選擇性**：使用「條件建置器」設計 IF 條件陳述式。請參閱使用條件建置器建立條件陳述式。

12. 在**公式**列中，按一下等號*左邊*的列以建立公式陳述式。按一下**動作**圖示，並選取：

> **備註：**
>
> 如果在步驟 9 至步驟 11 建立了條件陳述式 (即 IF 陳述式)，則在此步驟中建立的公式陳述式是條件的 THEN 陳述式。

- 變數 (請參閱使用變數。)
- 成員 (請參閱將成員和函式新增至元件。)
- 函式 (請參閱使用函式。)
- 智慧型列示 (請參閱使用智慧型列示。)

13. 在**公式**列中，按一下等號*右邊*的列以完成公式陳述式。按一下**動作**圖示，並選取：
    - 變數 (請參閱使用變數。)
    - 成員 (請參閱將成員和函式新增至元件。)
    - 函式 (請參閱使用函式。)
    - 智慧型列示 (請參閱使用智慧型列示。)

14. 按一下**註解**圖示，輸入關於計算陳述式列的備註。按一下**確定**。
15. **選擇性**：如果要建立其他 IF 陳述式、ELSE IF 或 ELSE 陳述式，請完成下列步驟：
    a. 按一下**新增條件**。
    b. 按一下**向下箭頭**，並選取下列選項之一：
        - **IF**：選取此選項以建立 IF 條件陳述式。如 IF 陳述式為 TRUE，則執行此動作；如果 IF 陳述式為 FALSE，則執行其他動作。
        - **ELSE IF**：選取此選項以建立 ELSE IF 條件陳述式。如果存在關聯的 IF 陳述式，且該 IF 陳述式為 FALSE，則執行此陳述式中的動作。
        - **ELSE**：選取此選項以建立 ELSE 條件陳述式。如果存在關聯的 ELSE IF 陳述式，且該 ELSE IF 陳述式為 FALSE，則執行此陳述式中的動作。

        > **備註：**
        > 如果從 IF 或 ELSE IF 區塊中移除條件陳述式，而下一個區塊中包含了 ELSE IF 陳述式，則下一個區塊會被變更為 IF 區塊。但是，如果下一個區塊中包含 ELSE 陳述式，則該區塊中的條件陳述式會被移除。

    c. 重複步驟 9 至步驟 14，以針對公式元件設計其他陳述式。公式方格可以由一或多個包含公式陳述式集合的公式區塊組成。您也可以為區塊輸入備註和條件 (視情況為選擇性)。

    > **提示：**
    > 如有必要，可按一下 **+** 圖示，以新增更多公式列。

16. 在**特性**上，完成下列步驟：
    a. **僅限為商業規則建立公式元件的使用者：**當您從**新物件**中將公式元件拖放到商業規則中時，並不會選取**共用**。若要將此公式設成共用，請選取公式的**特性**頁籤，選取**共用**，然後指定公式的名稱。

    當您從**系統檢視**建立公式元件時，依照預設，**共用**為已選取，而且無法變更。相反地，如果您要在商業規則中建立公式複本，請將共用公式拖放到商業規則中，然後在**特性**頁籤上清除**共用**核取方塊。這樣做將會在商業規則中建立共用公式複本；即使共用公式變更，共用公式的複本也不會變更。當您儲存商業規則時，公式將不再具有名稱。

    b. **選擇性：**輸入 50 個字元以內的新名稱以編輯公式名稱。(預設名稱為「新增公式」對話方塊中的名稱。)
    c. **選擇性：**為公式輸入 255 個字元以內的描述。
    d. **選擇性**：為公式輸入標題。標題顯示在**設計工具**與**範本設計工具**流程圖內公式的下方。

    > **備註：**
    > 如果元件沒有標題而且未共用該元件，元件的第一部分會顯示在流程圖中。例如，如果您有一個未共用而且沒有標題的公式 "Salaries" = 40，則該公式元件在流程圖中會顯示為 "Salaries" = 40。如果元件為共用但沒有標題，流程圖中會顯示元件的名稱。

e. 為公式輸入備註。例如，您可能希望告訴使用者公式的用途。

17. **僅限為商業規則建立公式元件的使用者：**在**使用方法**上，您可以檢視使用此公式元件的規則。

> **備註：**
>
> 「使用狀況」頁籤上的所有資訊都不可編輯。

18. 按一下 。

使用條件建置器建立條件陳述式

「條件建置器」可讓您為公式和條件元件建立條件陳述式。條件陳述式又稱為 If...Then 陳述式。如果條件陳述式的第一部分 (即 If 部分) 為 True，則該陳述式的第二部分 (即 Then 部分) 也為 True。如果條件陳述式的第一部分為 False，則第二部分可能為 True，也可能為 False。

若要建立條件陳述式，請執行下列動作：

1. 在公式上按一下右鍵，然後選取**開啟**。
2. 在**元件設計工具**中，按一下**新增條件**圖示。

   具有各種欄位和下拉清單以供您建立條件的兩個列會新增到「元件設計工具」。
3. 在第一列中，選取您是否要以 IF、ELSE 或 ELSE IF 開始條件中的第一個陳述式，並輸入條件陳述式的文字。或者，按一下列右側的**新增/編輯條件**圖示，以存取「條件建置器」。「條件建置器」可讓您以圖形方式設計條件陳述式。

   雖然您可以從向下箭頭中選取 IF、ELSE IF 和 ELSE，但根據預設，第一個陳述式必須是 IF 陳述式。
4. 在「條件建置器」中，選取**中繼資料條件**或**資料條件**。
5. 執行下列其中一項任務：
   - 輸入**函式**中的值 (針對中繼資料條件) 或**公式** (針對資料條件)。
   - 按一下**函式**或**公式**列，以顯示**動作**圖示。按一下**動作**圖示，然後選取選項以建立公式或函式：
     - 變數 (請參閱使用變數。)
     - 成員 (請參閱將成員和函式新增至元件。)
     - 函式 (請參閱使用函式。)
6. 輸入下列運算元之一：
   - = (等於)
   - < (大於)
   - > (小於)
   - <> (不等於)
   - >= (小於等於)
   - <= (大於等於)
7. 執行下列其中一項任務：

- 在**值**中輸入值。
- 按一下**值**列，以顯示**動作**圖示。按一下**動作**並選取選項，為公式或函式建立值：
  - 變數 (請參閱使用變數。)
  - 成員 (請參閱將成員和函式新增至元件。)
  - 函式 (請參閱使用函式。)

8. 為條件輸入備註。
9. 按一下 **+** 圖示，為「條件」方格新增條件。(您也可以使用 **+** 圖示為選定的列建立副本，並將該副本新增到「條件」方格中。) 請使用 **-** 圖示，以中繼資料或資料條件取代「條件」方格中選定的列。

> **提示：**
>
> 使用向上和向下箭頭圖示，將方格中的條件陳述式重新排序。使用「群組」和「取消群組」圖示，對條件陳述式各個部分進行分組 (在其中新增括弧) 或者取消分組 (在其中移除括弧)。藉由按一下 And 旁的欄位，並使用下拉選單選取選項，將 And 變更為 Or、And Not 或 Or。

10. 按一下**確定**，以結束**條件建置器**對話方塊。

    此時條件已經插入 IF 陳述式中。
11. **選擇性：**針對您要建立的每個條件陳述式，重複步驟 2 至步驟 10。
12. **選擇性：**在其他各條件陳述式中，按兩下第一列，以顯示下拉清單，讓您從中選取運算元作為各條件陳述式的開頭：
    - IF：*只有*第一項條件陳述式可以用 IF 開頭。(本選項只會在您建立的第一項條件陳述式中顯示為預設選項。)
    - NOT IF：*只有*第一項條件陳述式可以用 NOT IF 開頭。
    - AND：若要建立至少兩個以上的複合條件陳述式時，可以使用 AND 作為除了第一項條件之外任何條件陳述式的開頭。
    - OR：若要建立至少兩個以上的複合條件陳述式時，可以使用 OR 作為除了第一項條件之外任何條件陳述式的開頭。
    - AND NOT：若要在公式元件中包括接在運算元之後的條件陳述式時，可以使用 AND NOT 作為除了第一項條件之外任何條件陳述式的開頭。
    - OR NOT：若要在公式元件中排除接在運算元之後的條件陳述式時，可以使用 OR NOT 作為除了第一項條件之外任何條件陳述式的開頭。

> **提示：**
>
> 使用「群組」和「取消群組」圖示，在條件陳述式中新增或移除括弧。若要將多項條件陳述式劃分群組或取消群組，請使用「Ctrl + 按一下」或「Shift + 按一下」選取要劃分群組或取消群組的條件陳述式。

13. 按一下**確定**。

    條件陳述式會被插入「條件」列中。

輸入公式陳述式的備註

若要為公式陳述式輸入備註，請執行下列動作：

1. 透過展開的**公式**，執行下列其中一項任務：
   - 如果想要新增註解的公式位於商業規則中，請展開**規則**，用滑鼠右鍵按一下包含公式元件的規則，然後選取**開啟**。
   - 如果想要新增備註的公式位於範本中，請展開**範本**節點，並在包含公式元件的規則上按一下滑鼠右鍵，然後選取**開啟**。
   - 如果想要開啟公式本身，請展開**公式**節點，在該公式上按一下滑鼠右鍵，然後選取**開啟**。
2. 執行下列其中一項任務：
   - 如果您要為商業規則中之公式元件新增備註，請在**規則設計工具**流程圖中選取公式元件和**公式**頁籤。在**備註**文字方塊中輸入備註。
   - 如果要為範本中的公式新增備註，請在**範本設計工具**流程圖中，選取公式元件和**特性**頁籤。然後在**備註**文字方塊中輸入備註。
   - 如果要為公式元件本身新增備註，請在**元件設計工具**中的**備註**文字方塊中輸入備註。
3. 按一下 。

開啟公式元件

您可以從「系統檢視」開啟公式元件，或是從規則設計工具或範本設計工具的流程圖中開啟公式元件。

若要開啟公式元件，在**系統檢視**中，展開**公式**，用滑鼠右鍵按一下公式，然後選取**開啟**。

> **備註：**
>
> 如果商業規則中使用了該公式元件，您也可以在「規則設計工具」的規則流程圖中，用滑鼠右鍵按一下該公式元件並選取「開啟」予以開啟，或連按兩下該公式元件予以開啟。

編輯公式元件

您可以編輯包含公式元件以及公式元件的備註、標題、名稱及描述的陳述式。

若要編輯公式元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下公式，然後選取**開啟**。
2. 在**元件設計工具**中，編輯公式特性，然後按一下 。

   請查看設計公式元件和使用條件建置器建立條件陳述式。

刪除公式元件

只有公式元件未在任何規則或範本中使用時，您才能將其刪除。您可顯示其使用狀況，以檢視是否有任何規則或範本正在使用某個公式元件。(請參閱顯示公式元件和指令碼元件的使用情況。) 如果規則或範本中原本使用了某公式元件，而現在已經不再需要該元件，則可以將公式元件從該規則或範本中移除，然後再刪除公式元件。您也可以刪除該規則或範本，這樣會同時刪除其中包含的公式元件。

若要刪除公式元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下公式，然後選取**顯示使用情況**。

   如果有任何規則或範本正在使用該公式元件，您必須將該公式元件從中移除。

2. 確保沒有規則或範本使用該公式元件。

   請參閱從流程圖中移除元件。

3. 再次在公式上按一下右鍵，然後選取**刪除**。

複製和貼上公式元件

您可以從規則或範本中複製公式元件，然後貼上到同一個或另一個規則或範本中。您也可以複製公式元件內之條件方格的內容，然後貼到相同或不同的公式元件中。您無法將公式元件複製並貼到其他公式元件或其他元件類型中。

若要複製和貼上公式元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下規則或範本，然後選取**開啟**。
2. 在「規則設計工具」或「範本設計工具」流程圖中，用滑鼠右鍵按一下公式元件，然後選取**複製**。

   如果您要複製的元件是共用元件，您可以用滑鼠右鍵按一下公式元件，然後選取「複製參照」將參照複製到共用元件，而不是複製元件本身。(請參閱複製並貼上對商業規則公式或指令碼元件的參照。)

   如果您要複製的元件包含其他元件 (即子項)，您可以使用 Ctrl + 按一下來選取全部，或者用滑鼠右鍵按一下每個元件，然後選取「複製群組」，以複製群組中的所有元件。(請參閱在流程圖中複製和貼上元件群組。)

3. 執行下列其中一項任務：
   - 如果要將公式元件貼到*相同*商業規則或範本流程圖中，請用滑鼠右鍵按一下流程圖中要貼上該公式元件的位置，然後選取**貼上**。)
   - 如果要將公式元件貼到*另一個*商業規則或範本流程圖中，請開啟要貼上該元件的商業規則或範本，用滑鼠右鍵按一下流程圖中要貼上該公式元件的位置，然後選取**貼上**。
4. 按一下 。

指令碼元件

指令碼元件可以在商業規則或範本中使用。

另請參閱：

- 建立指令碼元件
- 設計指令碼元件

- 開啟指令碼元件
- 編輯指令碼元件
- 刪除指令碼元件
- 複製和貼上指令碼元件

建立指令碼元件

您可以從「系統檢視」建立指令碼元件。指令碼元件可以在商業規則或範本中使用。

若要建立指令碼元件，請執行下列動作：

1. 輸入指令碼的名稱與應用程式類型。
2. 選取**應用程式名稱**。應用程式名稱必須是有效的 Planning 應用程式。
3. 選取**計畫類型**，然後按一下**確定**。

> **備註：**
>
> 在「系統檢視」下，如果您用滑鼠右鍵按一下「指令碼」然後選取「新增」，「新增指令碼」對話方塊會填入應用程式類型、應用程式，以及您正在使用的計畫類型。

設計指令碼元件

指令碼元件可以在商業規則或範本中使用。您可以從「系統檢視」或任何其他檢視建立指令碼元件。也可以在設計商業規則或範本時，從規則或範本設計工具中建立指令碼元件。

> **備註：**
>
> 您也可以從商業規則或範本中設定指令碼元件。

若要設計指令碼元件，請執行下列動作：

1. 在**系統檢視**中，在**指令碼**上按一下右鍵，然後選取**新增**。
2. 輸入指令碼的名稱、應用程式類型、應用程式及計畫類型。
3. 按一下**確定**。
4. 使用圖示設計指令碼元件。

> **備註：**
>
> 與規則、範本和其他元件不同，當您開啟指令碼時，無法以圖形格式 (在流程圖內) 檢視該指令碼。您只能以指令碼格式檢視該指令碼。

您可使用指令碼執行下列其中一項任務：

- 按一下**隱藏/顯示行號**圖示，以隱藏 (或顯示) 指令碼行號。預設會顯示行號。
- 按一下**插入函式和其參數**圖示，將函式插入指令碼。請參閱使用函式。
- 按一下**插入在維度選取的成員**圖示，將成員插入指令碼。請參閱將成員和函式新增至元件。
- 按一下**插入變數**圖示，將變數插入指令碼。請參閱使用變數。
- 按一下**插入智慧型列示**，在指令碼中插入「智慧型列示」。請參閱使用智慧型列示。
- 按一下**備註**與**取消備註**圖示，以從指令碼新增和移除備註行。
- 按一下**隱藏備註**圖示隱藏指令碼中系統產生的備註。
- 按一下**驗證語法**圖示，以檢查指令碼語法是否錯誤。

> **備註：**
>
> 如果有語法錯誤，包含錯誤的代碼會變成紅色文字，而不會顯示錯誤訊息。如果沒有語法錯誤，則會顯示「找不到語法錯誤」訊息。

- 按一下**換行**圖示，讓捲出到頁面之外的長指令碼行以多行的方式顯示在同一頁上。
- 按一下**設定代碼完成關閉**圖示，以隱藏完成代碼的建議。(此圖示功能為切換。若要將代碼完成開啟，請再按一下圖示。依預設，代碼完成會設為「開啟」。)
- 按一下**取代**圖示以*尋找並取代*指令碼中的文字字串。請參閱搜尋商業規則指令碼中的文字字串。
- 按一下**尋找**圖示以*尋找*指令碼中的文字字串。請參閱搜尋商業規則指令碼中的文字字串。
- 在**尋找**中輸入搜尋文字然後按一下**上一個**或**下一個**，以尋找上一個 (在指令碼中向上搜尋) 或下一個 (在指令碼中向下搜尋) 文字例項。
- 在**移至行**中輸入行號，然後按一下箭頭，在指令碼中醒目提示該行號。
- 按一下行左邊加號或減號，在指令碼中展開或收合一行。例如，根據預設，FIX 陳述式會和所顯示的所有陳述式行一起顯示，但您可以按一下其左側的減號來隱藏 FIX 陳述式行。
- 選取您要複製的指令碼行，然後按一下**複製到剪貼簿**圖示。
- 選取您要剪下的指令碼行，然後按一下**剪下到剪貼簿**圖示。
- 將游標放在您要貼上之指令碼的位置，然後按一下**從剪貼簿貼上**圖示。
- 會為函式提供自動建議。當您鍵入函式的前幾個字元時，按下 **CTRL+空白鍵**以顯示建議。選取其中一個建議，然後按一下或按 **Enter** 來完成函式。

5. 在**特性**上，完成下列步驟：

   a. 當您建立指令碼元件時，**共用**為未選取。若要將指令碼元件設成共用，請依序選取**特性**頁籤、**共用**，然後輸入名稱。

      當您從「系統檢視」建立指令碼元件時，依照預設，**共用**為已選取，而且無法變更。相反地，如果您要取得商業規則中的指令碼複本，請將共用指令碼拖曳至商業規則中，然後在**特性**頁籤上，清除**共用**核取方塊。這樣做會在商業規則中建立共用指令碼的複本；如果共用指令碼沒有變更，則共用指令碼的複本也不會變更。當您儲存商業規則時，指令碼將不再具有名稱。

b. **選擇性**：透過輸入不超過 50 個字元的新名稱來編輯指令碼的名稱。(該名稱預設來自「新增指令碼」對話方塊。)

> **備註：**
>
> 您只能夠命名共用指令碼的名稱。如果指令碼不是共用，您無法在「名稱」欄位中輸入。

c. 為指令碼輸入 255 個字元以內的描述。

d. 輸入指令碼的標題。標題顯示在**規則設計工具**與**範本設計工具**流程圖中的指令碼下方。

e. 為指令碼輸入備註。例如，您可能要告訴使用者指令碼的用途。

6. 在**指令碼**上，輸入指令碼元件的標題。

7. **僅限為範本建立指令碼元件的使用者：**如果想在指令碼元件中使用設計時期提示，請選取**使用設計提示**。然後在條件方格中，透過執行下列任務，定義設計時期提示的條件：

   a. 在 **DTP** 中，選取設計時期提示。

   b. 在**運算子**中，選取運算子： = = 或 <>。

   c. 在**值**中，使用下拉清單選取值。您無法在此欄位中輸入。

   d. 重複這些步驟，根據需要建立任意數量的陳述式。

> **提示：**
>
> 按一下加號 (+) 和減號 (-) 圖示，在方格中新增和刪除列。

8. **僅限建立獨立指令碼元件 (也就是不在商業規則或範本內的指令碼元件) 或商業規則的指令碼元件的使用者：**在**使用狀況**上，您可以檢視使用指令碼元件的規則。

> **備註：**
>
> 您無法在此頁籤上編輯任何資訊。

9. 按一下 。

開啟指令碼元件

您可以從「系統檢視」中開啟指令碼元件，或是從規則設計工具或範本設計工具的流程圖中開啟指令碼元件。

若要開啟指令碼元件，請執行下列動作：

1. 在**系統檢視**中，展開**指令碼**。

2. 用滑鼠右鍵按一下指令碼，然後選取**開啟**。

> **備註：**
>
> 如果共用指令碼元件要用於商業規則中，您也可以透過用滑鼠右鍵按一下指令碼元件然後選取「開啟」，或者按兩下指令碼元件，在「規則設計工具」中從規則的流程圖加以開啟。

編輯指令碼元件

您可以編輯您用來撰寫指令碼元件的函式、變數和成員以及指令碼元件的備註、標題、名稱和描述。

若要編輯指令碼元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下指令碼，然後選取**開啟**。

   > **備註：**
   >
   > 與規則、範本和其他元件不同，當您開啟指令碼時，無法以圖形格式 (在流程圖內) 檢視該指令碼。您只能以指令碼格式檢視該指令碼。

2. 編輯所需的指令碼特性，然後按一下 。

   請參閱設計指令碼元件。

刪除指令碼元件

只有指令碼元件未在任何規則或範本中使用時，您才能將其刪除。您可顯示其使用狀況，以檢視是否有任何規則或範本正在使用某個指令碼元件。(請參閱顯示公式元件和指令碼元件的使用情況。) 如果規則或範本中原本使用了某指令碼元件，而現在已經不再需要該元件，則可以將指令碼元件從該規則或範本中移除，然後再刪除指令碼元件。您也可以刪除該規則或範本，這樣會同時刪除其中包含的指令碼元件。

若要刪除指令碼元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下指令碼，然後選取**顯示使用情況**。
2. 確保沒有規則或範本使用該指令碼元件

   如果有任何規則或範本正在使用該指令碼元件，您必須將該指令碼元件從中移除。請參閱從流程圖中移除元件。
3. 再次在指令碼上按一下右鍵，然後選取**刪除**。

複製和貼上指令碼元件

您可以從規則或範本中複製指令碼元件，然後將其貼上到同一或另一個規則或範本中。您也可以複製*指令碼元件中的指令碼*，並貼至同一個或不同的指令碼元件中。您無法將指令碼元件複製並貼上到另一個指令碼元件或元件類型中。

若要複製和貼上指令碼元件，請執行下列動作：

1. 在**系統檢視**中，選取包含指令碼元件的規則或範本。

2. 在**規則設計工具**或**範本設計工具**流程圖中，用滑鼠右鍵按一下指令碼元件，然後選取**複製**。

> **備註：**
>
> 如果您要複製的元件是共用元件，您可以用滑鼠右鍵按一下指令碼元件，然後選取「複製參照」以複製共用元件的參照，而不是複製元件本身。(請參閱複製並貼上對商業規則公式或指令碼元件的參照。)

3. 執行下列其中一項任務：
   - 如果要將指令碼元件貼到*相同*商業規則或範本中，請用滑鼠右鍵按一下流程圖中要貼上該元件的位置，然後選取**貼上**。
   - 如果要將指令碼元件貼到*另一個*商業規則或範本中，請開啟要貼上指令碼元件的商業規則或範本，用滑鼠右鍵按一下流程圖中您要貼上該元件的位置，然後選取**貼上**。
4. 按一下 。

條件元件

條件元件由條件陳述式組成，即 True 或 False。

另請參閱：

- 關於條件元件
- 建立條件元件
- 開啟條件元件
- 編輯條件元件
- 刪除條件元件
- 複製和貼上條件元件

關於條件元件

條件元件由條件陳述式組成 (即 IF...THEN 陳述式)，即 True 或 False。如果條件為 True，則系統將執行您指定的動作；如果條件為 False，則系統將執行您指定的其他動作。條件可以是中繼資料條件或資料條件。條件元件不能共用。

建立條件元件

您可以從商業規則或範本中建立條件元件。與指令碼和公式元件不同，條件元件無法作為獨立物件建立。條件元件會連結到為其建立的商業規則或範本。條件元件無法共用。

若要建立條件元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下規則或範本，然後選取**開啟**。
2. 在流程圖中確定要建立條件的位置後，從**新物件調色盤**，將**條件**元件拖放到流程圖中的該位置。

   條件物件在流程圖中顯示為一個帶問號的菱形。
3. **選擇性：**在**條件**上，輸入標題以識別條件元件。標題顯示在使用該條件的任何規則或範本的流程圖中的元件上方。

4. **僅限為範本建立條件元件的使用者：**如果想在條件元件中使用設計時期提示，請選取**使用設計提示**。然後在條件方格中，透過執行下列任務定義設計時期提示的條件：
   a. 在 **DTP** 中，選取設計時期提示。
   b. 在**運算子**中，選取運算子： = = 或 <>。
   c. 在**值**中，使用下拉清單選取值。您無法在此欄位中輸入。
   d. 重複這些步驟，根據需要建立任意數量的陳述式。

> **備註：**
>
> 如果使用設計時期提示來定義條件，您不能在「條件」方塊中輸入條件，也不能使用「條件建置器」為該方塊建置條件。

5. 如果您要建立成員區塊，請按一下**省略符號 (...)** 圖示，然後選取成員或函式。依預設，條件元件計算的處理會從您在方格中輸入的第一個成員開始。如果希望從另一個成員開始處理，請在**成員區塊**輸入成員或函式名稱，或按一下**省略符號**圖示，從「成員選取器」中選取一個成員。請參閱關於向元件新增成員和函式
6. 執行下列其中一項任務：
   - 在「條件」文字方塊中輸入條件陳述式。
   - 按一下**新增條件**，以使用「條件建置器」建立條件陳述式。請參閱使用條件建置器建立條件陳述式。
7. 為條件元件輸入註解，然後按一下 。

開啟條件元件

您可以從條件元件所屬的商業規則或範本的流程圖中將條件元件開啟。與公式和指令碼元件不同，DTP 指派元件不能從「系統檢視」中開啟。

若要開啟條件元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下規則或是範本，然後選取**開啟**。
2. 規則或範本開啟後，在流程圖中選取條件元件以檢視條件特性。

編輯條件元件

您可以編輯函式、變數和成員，以用來建立條件元件及條件元件的備註與標題。

若要編輯條件元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下規則或範本，然後選取「開啟」。
2. 在**規則設計工具**或**範本設計工具**中，選取條件以編輯**條件**中的特性。您可以編輯以下任意一個固定迴圈特性。(請參閱建立條件元件。)
   - 標題
   - 條件陳述式
   - 註解
   - **僅限範本使用者：**設計時期提示

3. 按一下 。

刪除條件元件

您可透過將條件元件從其所屬的商業規則或範本中移除來將其刪除。由於條件元件只能在一個商業規則或範本中使用，因此只需從商業規則或範本中將其移除即可刪除它。

若要刪除條件元件，請執行下列動作：

1. 在**系統設計工具**中，用滑鼠右鍵按一下規則或範本，然後選取**開啟**。
2. 在**規則設計工具**或**範本設計工具**中，選取您要在流程圖中刪除的條件元件。
3. 用滑鼠右鍵按一下該條件，然後選取**移除**。

   條件元件無法共用，因此，當您從商業規則或範本移除條件元件時，將會同時從資料庫刪除。

4. 確認刪除該元件，然後按一下 。

複製和貼上條件元件

您可以從規則或範本中複製條件元件，然後將其貼上到同一或另一個規則或範本中。您無法將條件元件複製並貼上到另一個條件元件或元件類型中。

若要從規則設計工具或範本設計工具中複製並貼上條件元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下規則或範本，然後選取**開啟**。
2. 在「規則設計工具」或「範本設計工具」流程圖中，用滑鼠右鍵按一下您要複製的條件元件，然後選取**複製**。
3. 執行下列其中一項任務：
   - 如果要將條件元件貼到*相同*商業規則或範本中，請用滑鼠右鍵按一下流程圖中要貼上該條件元件的位置，然後選取**貼上**。
   - 如果要將條件元件貼到*另一個*商業規則或範本中，請開啟要貼上該元件的商業規則或範本，用滑鼠右鍵按一下流程圖中您要貼上條件元件的位置，然後選取**貼上**。
4. 按一下 。

成員區塊元件

成員區塊元件定義在指令碼中括住一或多個陳述式的成員。

另請參閱：

- 關於成員區塊元件
- 建立成員區塊元件
- 開啟成員區塊元件
- 編輯成員區塊元件
- 刪除成員區塊元件
- 複製和貼上成員區塊元件

關於成員區塊元件

成員區塊元件定義在指令碼中括住一或多個陳述式的成員。例如，If 條件需要由一個成員區塊括住。成員區塊可用於商業規則和自訂範本。

如果您在公式的目標中使用 Oracle Essbase 替代變數或交叉維度成員，則需要有成員區塊。如果公式之目標的連續陳述式中有反覆的成員，也可以使用成員區塊。

成員區塊元件不會以資料庫物件的身分存在，資料庫物件獨立於建立這些資料庫物件的商業規則或範本之外；成員區塊元件只有作為其所屬範本或商業規則的一部分時才存在。成員區塊無法在規則和範本間共用。

建立成員區塊元件

您可以在設計規則或範本時，從「規則」或「範本設計工具」中建立成員區塊元件。您無法從「系統檢視」建立成員區塊元件。

若要建立成員區塊元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下規則或範本，然後選取**開啟**。
2. 在流程圖中確定您想要建立成員區塊的位置後，請從**新物件**中，將成員區塊物件拖放到流程圖中。

   成員區塊物件在流程圖中顯示為四個區塊，並以一條線連起來。
3. **選擇性：**如果您要為範本建立成員區塊元件，您也可以為其建立設計時期提示。請參閱針對自訂範本建立設計時期提示。
4. 在**成員區塊**頁籤的**成員區塊**旁，按一下**動作**圖示，請執行下列其中一項動作：
   - 選取**變數**以搜尋或建立變數，然後按一下**確定**。請參閱使用變數。
   - 選取**成員**，以搜尋成員，然後按一下**確定**。請參閱關於向元件新增成員和函式。
5. 在**錯誤與警告**頁籤上，按一下按鈕以執行指令碼診斷。
6. 在**特性**中，選取**停用**，以將成員區塊從驗證和計算中排除。然後輸入選擇性的描述和備註。
7. 按一下 。

開啟成員區塊元件

您可以從使用某個成員區塊元件的商業規則或範本的流程圖中開啟該成員區塊元件。成員區塊元件不能從「系統檢視」中開啟。

若要開啟成員區塊元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下規則或是範本，然後選取**開啟**。
2. 規則或範本開啟後，請在流程圖中選取成員區塊元件以檢視成員區塊特性。

編輯成員區塊元件

您可以變更成員區塊中的成員和成員區塊元件的設計時期提示 (僅限於自訂範本使用者)。

若要編輯成員區塊元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下規則或範本，然後選取**開啟**。

2. 在**規則設計工具**或**範本設計工具**中，選取流程圖中的成員區塊，以在**成員區塊**中編輯其特性。

   您可以變更您對成員區塊選取的成員或變數、成員區塊的描述與備註，以及是否停用成員區塊，使其不會包含在商業規則或範本的驗證和計算中。

3. 按一下 。

刪除成員區塊元件

您可以透過將成員區塊元件從其所屬的商業規則或範本中移除來刪除它。成員區塊元件只能在一個商業規則或範本中使用；成員區塊元件無法在其他商業規則或範本間共用。

若要刪除成員區塊元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下規則或範本，然後選取**開啟**。
2. 在**規則設計工具**或**範本設計工具**中，選取要從流程圖中刪除的成員區塊元件。
3. 用滑鼠右鍵按一下成員區塊，然後選取**移除**。

   成員區塊僅存在您對其建立的商業規則或範本中，因此，當您移除成員區塊時，將同時會從資料庫移除。

複製和貼上成員區塊元件

您可以從規則或範本中複製成員區塊元件，然後將其貼到同一個或另一個規則或範本中。您無法將成員區塊元件複製並貼到另一個成員區塊元件或元件類型中。

若要從規則設計工具或範本設計工具中複製成員區塊元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下規則或範本，然後選取**開啟**。
2. 在**規則設計工具**或**範本設計工具**流程圖中，用滑鼠右鍵按一下您要複製的成員區塊元件，然後選取**複製**。
3. 執行下列其中一項任務：
   - 若要將成員區塊元件貼到*同一個*商業規則或範本中，請用滑鼠右鍵按一下流程圖中要貼上該成員區塊元件的位置，然後選取**貼上**。
   - 若要將成員區塊元件貼到*另一個*的商業規則或範本中，請開啟要貼上該元件的商業規則或範本，用滑鼠右鍵按一下流程圖中您要貼上成員區塊元件的位置，然後選取**貼上**。
4. 按一下 。

成員範圍元件

成員範圍元件是由 Planning 維度中的各種成員組成的迴圈類型。

另請參閱：

- 關於成員範圍元件
- 建立成員範圍元件
- 開啟成員範圍元件
- 編輯成員範圍元件
- 刪除成員範圍元件

- 複製和貼上成員範圍元件

關於成員範圍元件

成員範圍元件是由 Planning 維度中的各種成員組成的迴圈類型。成員範圍元件不能共用，因此每次向商業規則或範本新增成員範圍元件時，都要建立新的成員範圍元件。

建立成員範圍元件

您可以在設計規則或範本時，從規則或範本設計工具中建立成員範圍元件。這與能獨立存在於使用它們的商業規則和範本的指令碼元件和公式元件不同，您無法從「系統檢視」中建立成員範圍元件。成員範圍元件連結到它們所屬的商業規則和範本，成員範圍元件無法共用。

若要建立成員範圍元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下規則或範本，然後選取**開啟**。
2. 從**新物件**中，拖曳成員範圍物件，並將其放置在流程圖中。

   「成員範圍」物件顯示為用一條線連起來的兩個圓。
3. **僅限自訂範本使用者：**如果您要為範本建立成員範圍元件，請在**設計時期提示**頁籤上，為其建立設計時期提示。請參閱針對自訂範本建立設計時期提示。
4. 在**成員範圍**頁籤中，請執行下列其中一項動作：
   - 選取**變數選取器**，以使用變數來定義成員範圍。然後輸入或選取一個變數。請參閱使用變數。

     > **備註：**
     >
     > 如果您選取此選項並變更成員，將會收到下列訊息：「編輯方格將會移除動態變數連結。確定要繼續嗎？」如果您要移除變數連結，請按一下「是」。按一下「是」可移除變數連結並保留內含變數原有成員的成員範圍。如果您不想失去變數連結，請按一下「否」。
     >
     > 如果您選取此選項，請選取**動態連結變數**，如此一來，不論何時對變數進行變更，規則或範本中的變數都會隨著這些變更同時更新。

   - 請執行下列步驟：
     a. 在要為其定義成員範圍的各個維度所對應的**值**欄中輸入成員，或按一下各列以顯示**動作**圖示。

     > **提示：**
     >
     > 為每個維度選取成員時，您可以按一下「成員選取器」圖示為方格中的所有維度選取成員，而無需一次一列。在「成員選取器」中按一下「確定」後，方格中將顯示您為每個維度選取的成員。

     b. 按一下**動作**圖示，然後選取下列選項之一來定義成員範圍：
        - 變數 (請參閱使用變數。)
        - 成員 (請參閱將成員和函式新增至元件。)
        - 函式 (請參閱使用函式。)

c. 按一下**註解**圖示輸入對成員範圍的備註。

d. 對要為其定義成員範圍的各個維度重複上述步驟。

e. 選取**排除方格值**，將您選取的成員排除在計算之外。

f. 按一下**重設方格**以清除您在方格中所輸入的任何成員、變數及函式。

g. 選取**啟用平行處理**以同時處理成員範圍命令，而不是循序處理。(預設為使用循序處理，但平行處理可以將計算最佳化。) 然後，在文字方塊中輸入平行處理可用的執行緒數量，或者按一下**變數選取器**，如果您是在商業規則中建立成員範圍，請選取數值或整數類型變數，如果您是在建立成員範圍範本，則請選取數值設計時期提示。

   根據預設，平行處理的執行緒數量為二。對於 32 位元平台，請選取 1 到 4 之間的整數。對於 64 位元平台，請選取 1 到 8 之間的整數。

5. **僅限商業規則使用者：**在**錯誤與警告**頁籤上，按一下按鈕以執行指令碼診斷。

6. **選擇性：**在**特性**中，選取**停用**以從驗證排除成員範圍 (以及其中任何元件)。當商業規則或範本中的元件導致商業規則或範本無法驗證時，您可能要使用此項目來進行疑難排解。

   您也可以輸入成員範圍的描述、備註及標題。標題顯示在該元件所屬的規則或範本的流程圖中的元件下方。

7. 按一下 。

開啟成員範圍元件

您可以從使用某個成員範圍元件的商業規則或範本的流程圖中開啟該成員範圍元件。與公式和指令碼元件不同，DTP 指派元件不能從「系統檢視」中開啟。

若要開啟成員範圍元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下規則或是範本，然後選取**開啟**。

2. 規則或範本開啟後，在流程圖中選取成員範圍元件以檢視成員範圍特性。

編輯成員範圍元件

您可以編輯用於定義成員範圍元件的維度和成員或變數，還可以編輯是否在日誌檔中記錄處理成員範圍元件所用的時間 (僅適用於 Oracle Hyperion Financial Management 應用程式) 以及成員範圍元件的註解和標題。

若要編輯成員範圍元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下規則或範本，然後選取**開啟**。

2. 在**規則設計工具**或**範本設計工具**中，選取成員範圍，以在**成員範圍**中編輯其特性。您可以編輯以下任意一個成員範圍特性。(請參閱建立成員範圍元件。)

   - 標題、描述和備註
   - 為定義成員範圍而選取的變數
   - 成員範圍中包含的維度
   - 定義了各個維度的成員範圍的成員

3. 按一下 。

刪除成員範圍元件

您可以透過將成員範圍元件從其所屬的商業規則或範本中移除來刪除它。成員範圍元件只能在一個商業規則或範本中使用，您只需從商業規則或範本中將其移除即可刪除它。

若要刪除成員範圍元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下規則或範本，然後選取**開啟**。
2. 在**規則設計工具**或**範本設計工具**中，選取要從流程圖中刪除的成員範圍元件。
3. 用滑鼠右鍵按一下成員範圍，然後選取**移除**。

   移除成員範圍會將其從資料庫刪除。

複製和貼上成員範圍元件

您可以從規則或範本中複製成員範圍元件，然後將其貼上到同一或另一個規則或範本中。您無法將成員範圍元件複製並貼上到另一個成員範圍元件或元件類型中。

若要從規則設計工具或範本設計工具中複製成員範圍元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下規則或範本，然後選取**開啟**。
2. 在**規則設計工具**或**範本設計工具**中，用滑鼠右鍵按一下要從流程圖中複製的成員範圍元件，然後選取**複製**。
3. 執行下列其中一項任務：
   - 若要將成員範圍元件貼到*相同*商業規則或範本，請用滑鼠右鍵按一下要從流程圖中貼上元件的位置，然後選取**貼上**。
   - 若要將成員範圍元件貼到*另一個*商業規則或範本中，請開啟要貼上該元件的商業規則或範本，用滑鼠右鍵按一下流程圖中您要貼上該元件的位置，然後選取**貼上**。
4. 按一下 。

固定迴圈元件

固定迴圈元件是以固定次數在中繼資料成員清單循環的物件。

另請參閱：

- 關於固定迴圈元件
- 建立固定迴圈元件
- 開啟固定迴圈元件
- 編輯固定迴圈元件
- 刪除固定迴圈元件
- 複製和貼上固定迴圈元件

關於固定迴圈元件

固定迴圈元件是以固定次數在中繼資料成員清單循環的物件。例如，您可以建立在科目清單中循環 10 次的固定迴圈。

建立固定迴圈元件

您可以透過將固定迴圈元件的圖示拖放到「規則設計工具」或「範本設計工具」流程圖，在商業規則或範本中建立固定迴圈元件。固定迴圈元件僅存在於為其建立該元件的商業規則或範本中。無法在商業規則或範本之間共用固定迴圈元件。

若要建立固定迴圈元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下規則或範本，然後選取**開啟**。
2. 在商業規則或範本流程圖確定好要建立固定迴圈的位置後，從**新物件**中，將**固定迴圈**物件拖放到流程圖中。

   固定迴圈顯示為以一條線連起來的兩個圓。
3. **僅限自訂範本使用者：**如果您要為範本建立固定迴圈元件，請在**設計時期提示**頁籤上，為其建立設計時期提示。請參閱針對自訂範本建立設計時期提示。
4. 在**固定迴圈**頁籤上，執行下列任務：
   a. 在**值**中，輸入希望在中繼資料或資料中循環的次數。或者，按一下**動作**圖示，然後選取**變數** (如果您使用商業規則)、或 **DTP** (如果您使用自訂範本)、或**函式** (如果您使用商業規則或自訂範本) 以選取變數、設計時期提示或函式，來定義迴圈。請參閱使用變數與使用函式。
   b. 在**中斷變數**中，輸入變數，或者按一下**變數**圖示 (如果您使用商業規則) 或 **DTP** 圖示 (如果您使用自訂範本) 以分別選擇數值變數或設計時期提示，以結束固定迴圈。中斷變數的值必須為 1。
   c. 在**錯誤與警告**頁籤上，按一下按鈕以執行指令碼診斷。
5. **選擇性：**在**特性**頁籤上，執行下列其中一項任務：
   - 如果您要從商業規則或範本驗證和計算排除固定迴圈元件，請選取**停用**。
   - 輸入標題以識別固定迴圈元件。標題顯示在該元件所屬的規則或範本的流程圖中的元件下方。
   - 輸入固定迴圈元件的描述和備註。
6. 按一下 。

開啟固定迴圈元件

您可以從固定迴圈元件所屬的商業規則或範本的流程圖中開啟該固定迴圈元件。與公式和指令碼元件不同，DTP 指派元件不能從「系統檢視」中開啟。

若要開啟固定迴圈元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下規則或是範本，然後選取**開啟**。
2. 規則或範本開啟後，在流程圖中選取固定迴圈元件以檢視固定迴圈特性。

編輯固定迴圈元件

您可以編輯指派給固定迴圈元件的值以及為該元件選取的變數。如果您正在建立商業規則的固定迴圈，您還可以編輯標題和中斷變數。中斷變數指定了結束固定迴圈的時間。

若要編輯固定迴圈元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下包含固定迴圈元件的規則或範本，然後選取**開啟**。
2. 按一下 。

刪除固定迴圈元件

您可以透過將固定迴圈元件從其所屬的商業規則或範本中移除來刪除它。固定迴圈元件只能在一個商業規則或範本中使用，您只需從商業規則或範本中將其移除即可刪除它。

若要刪除固定迴圈元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下包含要刪除之固定迴圈元件的規則或範本，然後選取**開啟**。
2. 在**規則設計工具**或**範本設計工具**中，選取要從流程圖中刪除的固定迴圈元件。
3. 用滑鼠右鍵按一下固定迴圈元件，然後選取**移除**。
4. 確認刪除該元件，然後按一下 。

複製和貼上固定迴圈元件

您可以從規則或範本中複製固定迴圈元件，然後將其貼上到同一或另一個規則或範本中。您無法將固定迴圈元件複製並貼上到另一個固定迴圈元件或另一個元件類型中。

若要從「規則設計工具」或「範本設計工具」中複製並貼上固定迴圈元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下包含要複製的固定迴圈元件的規則或範本，然後選取**開啟**。
2. 在**規則設計工具**或**範本設計工具**中，用滑鼠右鍵按一下要在流程圖中複製的固定迴圈元件，然後選取**複製**。
3. 執行下列其中一項任務：
   - 若要將固定迴圈元件貼到*相同*商業規則或範本，請用滑鼠右鍵按一下要從流程圖中貼上元件的位置，然後選取**貼上**。
   - 如果要將固定迴圈元件貼到*另一個*商業規則中，請開啟要貼上該元件的商業規則，用滑鼠右鍵按一下流程圖中您要貼上固定迴圈元件的位置，然後選取**貼上**。
4. 按一下 。

中繼資料迴圈元件

中繼資料迴圈元件可讓您使用函式 (中繼資料) 或開始與結束值 (固定)，指派值給多個成員。

另請參閱：

- 關於中繼資料迴圈元件
- 建立中繼資料迴圈元件
- 開啟中繼資料迴圈元件
- 刪除中繼資料迴圈元件
- 複製和貼上中繼資料迴圈元件

關於中繼資料迴圈元件

中繼資料迴圈元件可讓您使用函式 (中繼資料) 或開始與結束值 (固定)，指派值給多個成員。

例如，您可能想將 `#missing` 指派至父項「毛利」下的所有「層級 0」科目。在此範例中，您可選取中繼資料迴圈中的下列項目：

- 維度 - "Account"
- 函式 - `@RELATIVE("GP",0)`
- 變數 - "member" 類型設計時期提示

建立中繼資料迴圈元件

您可透過拖曳圖示並放在自訂範本的流程圖，以建立中繼資料迴圈元件。中繼資料迴圈元件只存在於您建立此元件所在的範本中。您無法在範本之間共用中繼資料迴圈元件。

若要建立中繼資料迴圈元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下自訂範本，然後選取**開啟**。
2. 在**範本設計工具**的**新物件**中，選取**中繼資料迴圈**物件，然後將其拖曳到流程圖中。
3. 在**中繼資料迴圈**頁籤中，選取下列其中一個迴圈類型：
   - **中繼資料**

     如果您選取**中繼資料**，請輸入下列資訊：
     - **索引 DTP** - 選取做為中繼資料迴圈的索引的現有設計時期提示。您選取的設計時期提示值在中繼資料迴圈開始前已重設為零，且每個迴圈的增量為一。
     - **維度** - 選取包含用於中繼資料迴圈之父項的維度。
     - **函式** - 選取函式，以指定如何將變數套用至維度成員。
     - **變數** - 建立設計時期提示*成員*，以用於變數。此設計時期提示用在公式中，而且 Calculation Manager 會使用指定函式建立的每個成員來取代設計時期提示。
   - **固定**

     如果您選取**固定**，請輸入下列資訊：
     - **索引 DTP** - 選取做為中繼資料迴圈的索引的現有步驟設計時期提示。您選取的設計時期提示值在中繼資料迴圈開始前已重設為零，且每個迴圈的增量為一。
     - **開始索引** - 迴圈開始的值。按一下 ，以選取要做為開始索引的設計時期提示。
     - **結束索引** - 迴圈結束的值。按一下 ，以選取要做為結束索引的設計時期提示。
4. 為中繼資料迴圈輸入特性：
   - 選取**停用**，以排除來自範本的中繼資料迴圈元件，以供驗證和計算目的。

     您可能想在商業規則無驗證，且您必須找到驗證錯誤來源時停用元件。當您停用元件時，會在流程圖中顯示為灰色。
   - 為中繼資料迴圈輸入**標題**。

     此標題顯示在流程圖中的中繼資料迴圈元件圖示的下方。
   - 輸入**描述**與**備註**。

5. 按一下 。

開啟中繼資料迴圈元件

您從所屬自訂範本的流程圖開啟中繼資料迴圈元件。與公式和指令碼元件不同，您無法直接從**系統檢視**中開啟中繼資料迴圈元件。

若要開啟中繼資料迴圈元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下自訂範本，然後選取**開啟**。
2. 在**範本設計工具**中，在流程圖中選取中繼資料迴圈元件以查看其特性。

刪除中繼資料迴圈元件

您可刪除在所屬自訂範本內的中繼資料迴圈元件。

若要刪除中繼資料迴圈元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下包含中繼資料迴圈的自訂範本，然後選取**開啟**。
2. 在**範本設計工具**中，在流程圖中，用滑鼠右鍵按一下中繼資料迴圈元件，然後選取**移除**。

> **備註：**
>
> 這些步驟只會從範本刪除中繼資料迴圈。您也應該刪除您建立用來與中繼資料迴圈搭配使用的設計時期提示。若要刪除設計時期提示，請從**設計時期提示**頁籤中，用滑鼠右鍵按一下包含設計時期提示的列，然後選取**刪除列**，或選取包含設計時期提示的列，然後按一下 。

複製和貼上中繼資料迴圈元件

您可以從自訂範本中複製中繼資料迴圈元件，然後將其貼上到相同或另一個自訂範本中。

若要複製和貼上中繼資料迴圈元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下包含中繼資料迴圈元件的自訂範本，然後選取**開啟**。
2. 在**範本設計工具**中，用滑鼠右鍵按一下流程圖中要複製的中繼資料迴圈元件，然後選取**複製**。

   如果您將公式或指令碼元件拖曳到中繼資料迴圈內，請選取**複製群組**，以複製中繼資料迴圈內的中繼資料迴圈和元件
3. 執行下列其中一項任務：
   - 若要將中繼資料迴圈元件貼到*相同*範本，請用滑鼠右鍵按一下流程圖中的所需位置，然後選取**貼上**。
   - 若要將中繼資料迴圈元件貼到*另一個*範本，請開啟所需的範本，然後用滑鼠右鍵按一下流程圖中所需的位置，然後選取**貼上**。
4. 按一下 。

> **備註：**
>
> 如果設計時期提示用於中繼資料迴圈，並且要加以複製時，請在**設計時期提示**頁籤中，用滑鼠右鍵按一下包含要複製的設計時期提示的列，然後選取**複製**。
>
> 若要複製所有設計階段提示，請選取**複製方格**，然後開啟您要貼上設計階段提示的範本，然後在**設計階段提示**頁籤中，按一下方格，然後按一下右鍵，並選取**貼上**。

DTP 指派元件

使用 DTP 指派元件以將 DTP、成員、函式，或輸入文字指派至自訂範本中的設計時期提示。

另請參閱：

- 關於 DTP 指派元件
- 建立 DTP 指派元件
- 開啟 DTP 指派元件
- 編輯 DTP 指派元件
- 刪除 DTP 指派元件
- 複製和貼上 DTP 指派元件
- 在 DTP 指派元件中使用設計時期提示函式

關於 DTP 指派元件

您可以使用 DTP 指派元件指派 DTP、成員、函式，或輸入文字至自訂範本中的設計時期提示。DTP 指派元件可降低範本流程圖的複雜性，並使範本邏輯更易開發和維護。

建立 DTP 指派元件

您可以在範本中建立 DTP 指派元件，方法是將此元件的圖示拖放到範本設計工具的流程圖中。

建立 DTP 指派元件時，請注意下列事項：

- DTP 指派元件只存在於您建立此元件所在的範本中，
- 除了未使用 DTP 條件的成員範圍、成員區塊或條件元件內以外，您可以在流程圖的任何地方放置 DTP 元件。

若要建立 DTP 指派元件，請執行下列動作：

1. 在**系統檢視**中，開啟自訂範本。
2. 在**範本設計工具**的**新物件**中，選取 **DTP 指派**物件，然後將其拖曳到流程圖中。
3. **選擇性：**輸入元件的設計時期提示。請參閱針對自訂範本建立設計時期提示。
4. 在**公式**頁籤中，針對範本的設計時期提示定義條件。
   - 在等號左側的下拉清單中，選取要定義值的設計時期提示。
     如果下拉清單中沒有任何設計時期提示，則不會為您所處理的範本建立任何內容。
   - 在等號右邊的文字方塊中，輸入值，或按一下 ，以選取變數、成員或函式。

- 按一下 +，以針對其他設計時期提示指派值。

5. 針對 DTP 指派元件輸入特性：
   - 選取**停用**，以排除來自範本的中繼資料迴圈元件，以供驗證和計算目的。

     您可能想在商業規則無驗證，且您必須找到驗證錯誤來源時停用元件。當您停用元件時，會在流程圖中顯示為灰色。
   - 針對 DTP 指派元件輸入**標題**。

     此標題顯示在流程圖中的 DTP 指派元件圖示的下方
   - 輸入**描述**與**備註**。
6. 按一下 。

開啟 DTP 指派元件

您可以從 DTP 指派元件所屬的範本的流程圖中開啟該 DTP 指派元件。與公式和指令碼元件不同，您無法直接從**系統檢視**開啟 DTP 指派元件。

若要開啟 DTP 指派元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下自訂範本，然後選取**開啟**。
2. 在**範本設計工具**中，在流程圖中選取 DTP 指派元件以查看其特性。

編輯 DTP 指派元件

您可以編輯 DTP 指派元件的描述、標題和註解。您也可以在使用該 DTP 指派元件的範本中編輯您指派給設計時期提示的值。

若要編輯 DTP 指派元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下包含待編輯之 DTP 指派元件的範本，然後選取**開啟**。
2. 在**範本設計工具**中，選取流程圖中的 DTP 指派元件。
3. 進行所需的變更，然後按一下 。

刪除 DTP 指派元件

您可從其所屬的範本刪除 DTP 指派元件。

若要刪除 DTP 指派元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下包含 DTP 指派元件的自訂範本，然後選取**開啟**。
2. 在**範本設計工具**中，在流程圖中，用滑鼠右鍵按一下 DTP 指派元件，然後選取**移除**。

複製和貼上 DTP 指派元件

您可以從範本中複製 DTP 指派元件，然後將其貼上到同一個或另一個範本中。

若要複製和貼上 DTP 指派元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下包含 DTP 指派元件的自訂範本，然後選取**開啟**。

2. 在**範本設計工具**中，用滑鼠右鍵按一下流程圖中要複製的 DTP 指派元件，然後選取**複製**。
3. 執行下列其中一項任務：
   - 若要將 DTP 指派元件貼到*相同*範本，請用滑鼠右鍵按一下要從流程圖中貼上元件的位置，然後選取**貼上**。
   - 若要將 DTP 指派元件貼到*另一個*範本，請開啟所需的範本，然後用滑鼠右鍵按一下流程圖中所需的位置，然後選取**貼上**。

     如果您將 DTP 指派貼到新範本，您也必須建立 DTP 指派元件所使用的設計時期提示，或將其複製到新範本。
4. 按一下 。

在 DTP 指派元件中使用設計時期提示函式

另請參閱：

- 關於設計時期提示函式
- @AvailDimCount
- @Compare
- @Compliment
- @Concat
- @DenseMember
- @Dependency
- @DimAttribute
- @DimMember
- @DimName
- @DimType
- @DimUDA
- @EndsWith
- @Evaluate
- @FindFirst
- @FindLast
- @GetData
- @Integer
- @Intersect
- @IsAncest
- @IsChild
- @IsDataMissing
- @IsSandBoxed
- @IsVariable
- @Length

- @Matches
- @Member
- @MemberGeneration
- @MemberLevel
- @MsgFormat
- @Notin
- @OpenDimCount
- @Plandim
- @PlanDimMember
- @Quote
- @RemoveQuote
- @ReplaceAll
- @ReplaceFirst
- @SmartListFromIndex
- @SmartListFromValue
- @SparseMember
- @StartsWith
- @SubString
- @ToLowerCase
- @ToMDX
- @ToUpperCase
- @Trim
- @Union
- @ValueDimCount

關於設計時期提示函式

您可以針對自訂範本，使用 DTP 指派元件中的設計時期提示函式。這些函式可讓您執行的任務包括成員和成員比較、在兩個設計時期提示之間取得共通或非共通的成員清單、新增或移除字串中的引號、轉換字串中字元的大小寫，以及其他任務。

@AvailDimCount

`@AVAILDIMCOUNT` 傳回可用維度數目。

DTP 類型：

`@AVAILDIMCOUNT` 可指派至*數值*類型的 DTP。

語法：

`@AVAILDIMCOUNT(`*`DTP`*`,Dense/Sparse)`

參數：

參數	描述
DTP	傳回成員範圍或跨維度的設計時期提示或函式。
Dense/Sparse	**選擇性。**如果保留空白，會傳回所有可用維度的計數。如果已輸入 "Dense" 或 "Sparse"，只會計入指定類型的可用維度。

範例 1：

```
[DTP1]=@AVAILDIMCOUNT([MR1])
```

其中：

- [DTP1] 是*數值* 類型的 DTP。
- [MR1] 是*成員範圍* 類型的 DTP，具有下列輸入：
  - Account：
  - Period：Jan、Feb、Mar
  - HSP_View：
  - Year：
  - Scenario：
  - Version：作業中
  - Entity：
  - Product：P_000

在此範例中，[DTP1] 傳回 "8"，因為總共有八個維度。

範例 2：

```
[DTP2]=@AVAILDIMCOUNT([MR2],"Dense")
```

其中：

- [DTP2] 是*數值* 類型的 DTP。
- [MR2] 是*成員範圍* 類型的 DTP，具有下列輸入：
  - Account：密集
  - Period：密集
  - HSP_View：稀疏
  - Year：稀疏
  - Scenario：稀疏
  - Version：稀疏
  - Entity：稀疏
  - Product：稀疏

在此範例中，DTP2 傳回 "2"，因為只有 "Account" 與 "Period" 是密集的。

@Compare

`@COMPARE` 將在比較成員與另一個成員、維度與另一個維度、密碼與另一個密碼，或字串與限制清單時，傳回 "true" 或 "false"。(對於限制清單，`@COMPARE` 將使用規則建置器值進行比較，而非取代值。)

DTP 類型：

`@COMPARE` 可指派至*布林值* 類型的 DTP。

語法：

`@COMPARE(String,compareString,IgnoreCase)`

參數：

參數	描述
String	設計時期提示或文字
compareString	設計時期提示或文字
IgnoreCase	**選擇性**。如果是 "true"，則可忽略參數 1 和 2 的大小寫。如果是 "false"，則參數 1 和 2 必須符合大小寫。如果 *IgnoreCase* 保留空白，其預設為 "true"。

範例：

`[DTP1]=@COMPARE([MBR1],[MBR2],true)`

其中：

- `[DPT1]` 是*布林值*類型之不可提示的 DTP。
- `[MBR1]`=Jan
- `[MBR2]`=Jan

在此範例中，`[DTP1]` 傳回 "true"。

@Compliment

`@COMPLIMENT` 會傳回不在 DTP2 而在 DTP1 中的成員，以及不在 DTP2 而在 DTP1 中的成員。換句話說，`@COMPLIMENT` 會傳回 DTP1 和 DTP2 之間不通用的成員。

DTP 類型：

`@COMPLIMENT` 可指派至*成員範圍*或*跨維度* 類型的 DTP。

語法：

`@COMPLIMENT(Argument1,Argument2)`

參數：

參數	描述
Argument1	傳回*成員範圍*或*跨維度*的設計時期提示或函式
Argument2	傳回*成員範圍*或*跨維度*的設計時期提示或函式

範例 1：

```
[DTP_MR]=@COMPLIMENT([MR1],[MR2])
```

其中：

- `[DTP_MR]` 是*成員範圍*類型的 DTP，而非確認提示。
- `[MR1]`="Salaries,"FY14","Local","Tennessee","USA"
- `[MR2]`="Jan", "Feb", "Mar","FY15","Actual", "Budget","Working","Florida","California","000", "G100"

在此範例中，`DTP_MR` 傳回 "Salaries"、"Jan"、"Feb"、"Mar"、"Actual"、"Budget"、"Working"、"Local"、"000"、"G100"、"USA"

範例 2：

```
[DTP_CD]=@COMPLIMENT([CD1],[CD2])
```

其中：

- `[DTP_CD]` 是*跨維度*類型的 DTP，而非確認提示。
- `[CD1]`="Salaries"->"Jan"->"Actual"->"Local"
- `[CD2]`="50100"->"Feb"->"Budget"->"Working"->"California"->"000"

在此範例中，`[DTP_CD]` 傳回 "Working"->"Local"->"California"->"000"

@Concat

`@CONCAT` 會將第二個參數連接至此第一個參數的尾端。

DTP 類型：

`@CONCAT` 可指派至*成員*、*維度*、*字串*、*限制清單*、*密碼*或*屬性*類型的 DTP。

語法：

`@CONCAT(`*`Text, Concat String, Seperator`*`)`

參數：

參數	描述
Text	設計時期提示或文字
Concat String	設計時期提示或文字 **注意：***concatString* 將新增至*字串*結尾
Seperator	**選擇性：**如果已使用分隔符號，則只有在*文字*有值的時候，才會在*文字*和*連接字串*之間新增分隔符號。

範例：

```
[DTP_CTmbr]=@CONCAT([MBR1],[MBR2])
```

其中：

- `[DTP_CTmbr]` 是*成員範圍*類型的 DTP，而非確認提示。

- [MBR1]="California"
- [MBR2]="Washington"

在此範例中，[DTP_CTmbr] 傳回 "CaliforniaWashington"。

@DenseMember

@DENSEMEMBER 會傳回跨維度中的第一個密集維度成員。

DTP 類型：

@DENSEMEMBER 可被指派至*成員*、*多個成員*、*維度*、*多個維度*、*成員範圍*、*跨維度*或*字串* 類型的 DTP。

語法：

@DENSEMEMBER(*Members*)

參數：

參數	描述
Members	傳回*跨維度*或*成員範圍*的設計時期提示、成員或函式

範例：

[DenseMbr]=@DENSEMEMBER([CD1])

其中：

- [DenseMbr] 是非確認提示之*跨維度* 類型的 DTP。
- [CD1] 是可提示之*跨維度* 類型的 DTP。
- [CD1]="Salaries"->"Jan"->"Working"->"Tennessee"->"G401"

在此範例中，[Densembr] 會傳回 "Salaries"，因為 "Salaries" 是跨維度清單中的第一個密集成員。

@Dependency

「內含」針對具備從相同維度指定之成員的 Input 2，傳回來自 Input 1 的成員。「排除」針對在相同維度中沒有特定成員的 Input 2，傳回來自 Input 1 的成員。

目標 DTP 類型可與下列 @DEPENCENCY 的 DTP 搭配使用：

目標 DTP 類型	@DEPENDENCY 中所使用的參數
成員	成員範圍、跨維度、單一維度，或多個維度。如果成員用於 @DEPENDENCY，其必須用於函式 @DIMMEMBER (成員) 中。
成員	單一維度或多個維度
維度	成員範圍、跨維度、單一維度，或多個維度
維度	成員範圍或跨維度
成員範圍	成員範圍、跨維度、單一維度，或多個維度
跨維度	成員範圍、跨維度、單一維度，或多個維度

語法：

`@DEPENENCY (Input1, Dependence, Input2)`

參數：

參數	描述
Input1	設計時期提示或函式
Dependence	「內含」或「排除」
Input2	設計時期提示或函式

下列範例 1 與範例 2 假設下列輸入：

- `[cd_mr_i1]` 是非確認提示之*跨維度*類型的 DTP。
- `[CD1]` 是具有下列輸入之可提示的*跨維度*類型的 DTP。
  - Account：「單位」
  - Period：「一月」
  - HSP_View：
  - Year：「FY15」
  - Scenario：「實際」
  - Version：「作業中」
  - Entity：「111」
  - Product：
- `[POV]` 是具有下列輸入之可提示的*成員範圍* DTP。
  - Account：
  - Period：「二月」
  - HSP_View：
  - Year：「FY16」
  - Scenario：「實際」
  - Version：「作業中」
  - Entity：
  - Product：「P_000」

範例 1：

```
[cd_mr_i1]=@DEPENDENCY([CD1],"Inclusive",[POV])
```

在此範例中，`[cd_mr_i1]` 會傳回 "Jan->FY15->Actual->Working"，因為 "Period"、"Year"、"Scenario" 和 "Version" 具有 "CD1" 和 "POV" 的輸入值。

範例 2：

```
[cd_mr_e1]=@DEPENDENCY([CD1],"Exclusive",[POV]
```

在此範例中，`[cd_mr_e1]` 會傳回 "Units->"111"，因為 "Account" 與 "Entity" 是 "Input 1" 中唯一的維度，其擁有指定的成員，而且該成員未在 "Input 2" 中指定。

@DimAttribute

如果指定屬性與維度相關聯，`@DIMATTRIBUTE` 會傳回屬性名稱。

DTP 類型：

`@DIMATTRIBUTE` 可指派至*屬性*、*成員*、*維度*或*字串* 類型的 DTP。

語法：

`@DIMATTRIBUTE(`*`Dimension`*`,`*`Attribute`*`)`

參數：

參數	描述
Dimension	設計時期提示或維度下拉清單可讓您從應用程式選取維度
Attribute	傳回屬性維度名稱或屬性成員名稱的設計時期提示、屬性 (維度或成員)，或函式

範例：

`[DIMA_ATTRB]=@DIMATTRIBUTE([DIM],[ATTRB])`

其中：

- `[DIMA_ATTRB]` 是非確認提示之*屬性* 類型的 DTP。
- `[DIM]` 是可提示之*維度* 類型的 DTP。
- `[ATTRB]` 是可提示之*屬性* 類型的 DTP。

假設下列值是輸入：

- `[DIM]`: "Entity"
- `[ATTRB]`: "Small"

在此範例中，因為 "Entity" 具有名稱為 "Size" 的屬性維度，而且 "Small" 是 "Size" 下的成員，`[DIMA_ATTRB]` 會傳回 "Small"。

@DimMember

`@DIMMEMBER` 會在對指定維度有效時傳回成員。

DTP 類型：

`@DIMMEMBER` 可指派至*單一成員*、*多個成員*、*跨維度*、*成員範圍*、*單一維度*或*多個維度* 類型的 DTP。

語法：

`@DIMMEMBER(`*`Dimension`*`,`*`Member`*`)`

參數：

參數	描述
Dimension	設計時期提示或維度
Member	顯示要選取之應用程式中維度清單的設計時期提示、成員或函式下拉清單

範例：

```
[DIM_mbr1]=@DIMMEMBER([DIM],[Member])
```

其中：

- `[DIM_mbr1]` 是非確認提示之*成員* 類型的 DTP。
- `[DIM]` 是確認提示之*維度* 類型的 DTP。
- `[Member]` 是確認提示之*成員* 類型的 DTP。
- `[Dim]`="Entity"
- `[Member]`="Washington"

在此範例中，`[DIM_mbr1]` 會傳回「華盛頓」，因為華盛頓是 Entity 維度的成員。

@DimName

如果維度名稱對資料庫有效，`@DIMNAME` 會傳回該名稱。

DTP 類型：

`@DIMNAME` 可指派至*維度* 類型的 DTP。

語法：

`@DIMNAME(`*Dimension*`)`

參數：

參數	描述
Dimension	設計時期提示、維度或輸入文字。 按一下維度，以顯示可用維度的清單。

範例：

```
[DIMN_Product]=@DIMNAME(["Product"])
```

其中：

- `[DIMN_Product]` 是非確認提示之*維度* 類型的 DTP。
- `"Product"` 已輸入函式中。

如果 "Product" 是此規劃類型的有效維度名稱，則 `[DIMN_Product]` 會傳回 "Product"。

如果 "Product" 對此規劃類型不是有效的維度，則 `[DIMN_Product]` 會傳回 "empty"。

@DimType

`@DIMTYPE` 會在符合指定類型時傳回維度。

DTP 類型：

`@DIMTYPE` 可指派至*單一維度*、*多個維度*或*字串* 類型的 DTP。

語法：

`@DIMTYPE(Dimension Type)`

參數：

參數	描述
Dimension Type	用於應用程式中的維度類型

範例：

```
[Dims]=@DIMTYPE(Account)
```

其中：

- `[Dims]` 是非確認提示之*維度* 類型的 DTP。
- `[Account]` 已從下拉清單選取。

在此範例中，`[Dims]` 會傳回 "Account"。如果 Account 維度已在應用程式中命名為 Measures，則會傳回 "Measures"。

@DimUDA

如果指定的 UDA 對此維度有效，則 `@DIMUDA` 會傳回 UDA 名稱。

DTP 類型：

`@DIMUDA` 可指派至 *UDA*、*成員*、*維度*或*字串* 類型的 DTP。

語法：

`@UDA(Dimension,UDA)`

參數：

參數	描述
Dimension	設計時期提示或維度 按一下維度，以顯示可用維度的清單。
UDA	如果您在 *Dimension* 參數中選取 DTP 或 UDA，然後從下拉清單選取 UDA。

範例：

```
[dimu_mbr2]=@DIMUDA([DIM],[UDA2] [dimu_mbr]=@DIMUDA([DIM],[UDA]
```

其中：

- [dimu_mbr2] 是非確認提示之*成員* 類型的 DTP。
- [dimu_mbr] 是非確認提示之*成員* 類型的 DTP。
- [dim] 是確認提示之*維度* 類型的 DTP
- [UDA2] 是確認提示之 *UDA* 類型的 DTP。
- [UDA] 是確認提示之 *UDA* 類型的 DTP。

假設下列值是輸入：

- [dim]: "Account"
- [UDA2]: "HSP_LEAPYEAR"
- [UDA]: "Revenue"

在此範例中，[dimu_mbr2] 不會傳回任何內容，因為 "HSP_LEAPYEAR" 不是 "Account" 上的 UDA，而且 [dimu_mbr] 會傳回 "Revenue"。

@EndsWith

@ENDSWITH 測試此字串是否以指定的後綴結束。

DTP 類型：

@ENDSWITH 可指派至*布林值* 類型的 DTP。

語法：

@ENDSWITH(*String*,*Suffix*)

參數：

參數	描述
String	設計時期提示或文字
Suffix	設計時期提示或文字

範例：

[DTP_end]=@ENDSWITH([Mbr1],[String1])

其中：

- [Mbr1] 是確認提示之*成員* 類型的 DTP。
- String1 是確認提示之*字串* 類型的 DTP。

假設下列值為輸入內容：

- [Mbr1]: "Joe Smith"
- [String1]: "Smith"

在此範例中，[DTP_end] 會傳回 "true"。

@Evaluate

@EVALUATE 傳回表示式的結果。

DTP 類型：

`@Evaluate` 可指派至*數值*或*整數* 類型的 DTP。

語法：

`@EVALUATE(Value1,Operator,Value2)`

參數：

參數	描述
Value1	傳回數值的設計時期提示或函式
Operator	+、-、/ 或 *
Value2	傳回數值的設計時期提示或函式

範例：

```
[Eval_num_add]=@EVALUATE([num1], "+",[num2])
```

其中：

- `[Eval_numadd]` 是非確認提示之*數值* 類型的 DTP。
- `[num1]` 是*數值*類型的可提示 DTP。
- 運算子是 +。
- `[num2]` 是*數值*類型的可提示 DTP。

假設下列值為輸入內容。

- `[num1]`: "100"
- `[num2]`: "10"

在此範例中，`[Eval_numadd]` 會傳回 "110.0"。

@FindFirst

`@FINDFIRST` 尋找符合給定一般表示式之字串的第一個子字串。

DTP 類型：

`@FINDFIRST` 可指派至*字串*、*密碼*、*單一成員*、*多個成員*、*單一維度*或*多個維度* 類型的 DTP。

語法：

`@FINDFIRST(text,regExpr,IgnoreCase)`

參數：

參數	描述
text	設計時期提示、成員、函式或輸入文字
regExpr	請參閱有關 "java.util.regex.Pattern" 的 Java 文件。
IgnoreCase	**選擇性。**True 或 False。如果保留空白，預設值為 True。

範例：

```
[FINDF_PW_T]=@FINDFIRST([PW],[FindF_String_PW],"true")
```

其中：

- `[FINDF_PW_T]` 是非確認提示之*密碼* 類型的 DTP。
- `[PW]` 是*密碼*類型的可提示 DTP。
- `[FindF_String_PW]` 是*字串*類型的可提示 DTP。

假設下列值為輸入內容：

- `[PW]`: " test20pw15test"
- `[FindF_String_PW]`: "\d\d " (其為 JAVA 表示式，可在數字後面傳回任何數字)
- `Ignorecase`: "true"

在此範例中，`[FINDF_PW_T]` 會傳回 "20"。

@FindLast

`@FINDLAST` 尋找符合給定一般表示式之字串的最後一個子字串。

DTP 類型：

`@FINDLAST` 可指派至*字串*、*密碼*、*單一成員*、*多個成員*、*單一維度*或*多個維度* 類型的 DTP。

語法：

`@FINDLAST(text,regExpr,IgnoreCase)`

參數：

參數	描述
text	設計時期提示、成員、函式或輸入文字。
regExpr	請參閱有關 "java.util.regex.Pattern" 的 Java 文件。
IgnoreCase	**選擇性。**True 或 False。如果保留空白，預設值為 True。

範例：

```
[FINDL_PW_T]= @FINDLAST([PW],[FindL_String_PW],"true")
```

其中：

- `[FINDL_PW_T]` 是非確認提示之*密碼* 類型的 DTP。
- `[PW]` 是*密碼*類型的可提示 DTP。
- `[FindL_String_PW]` 是*密碼*類型的可提示 DTP。

假設下列值為輸入內容：

- `[PW]`: " test20pw15test"
- `[FindL_String_PW]`: "\d\d " (其為 JAVA 表示式，可在數字後面傳回任何數字)
- `Ignorecase`: "true"

在此範例中，`[FINDL_PW_T]` 傳回 "15"。

@GetData

`@GETDATA` 傳回資料塊的值。

DTP 類型：

`@GETDATA` 可指派至*數值*類型的 DTP。

語法：

`@GETDATA(Members)`

參數：

參數	描述
Members	設計時期提示、成員、或傳回跨維度的函式

範例：

```
@GETDATA([CD]) < 10
```

用於範本中步驟的函式，以啟用條件。

如果針對可提示設計時期提示 CD 輸入的成員具有小於 10 的值，則將會顯示步驟。如果針對可提示設計時期提示 CD 的成員值大於 10，則不會顯示步驟。

@Integer

`@INTEGER` 傳回整數。

DTP 類型：

`@INTEGER` 可指派至*整數*類型的 DTP。

語法：

`@INTEGER(value)`

參數：

參數：	描述
value	傳回值的設計時期提示或函式

範例：

```
[INT_NUM]=@INTEGER([num1])
```

其中：

- `[INT_NUM]` 是非確認提示之*整數*類型的 DTP。
- `[num1]` 是確認提示之*數值*類型的 DTP。

假設 `[num1]` 的輸入是 "15.7"

在此範例中，[INT_NUM] 傳回 15。

@Intersect

@INTERSECT 將傳回來自共同維度的成員。

DTP 類型：

@INTERSECT 可指派至*成員範圍*或*跨維度* 類型的 DTP。

語法：

@INTERSECT(*Argument1*,*Argument2*)

參數：

參數	描述
Argument1	設計時期提示、函式或文字
Argument2	設計時期提示、函式或文字

範例：

[IntersetMR]=@INTERSECT([MR1],[MR2])

其中：

- [MR1] 是確認提示之*成員* 類型的 DTP。
- [MR2] 是確認提示之*成員* 類型的 DTP。

假設 MR1 的輸入是：

- Account: "Salaries"
- Version: "Target"
- Entity: "Tennessee", "Florida"

而 MR2 的輸入是：

- Account: "50590", "50350"
- Years: "FY10"
- Scenario: "Budget"
- Version: "First Pass"

在此範例中，[IntersectMR] 傳回 "50590"、"50350"、"Salaries"、"First Pass"、"Target"。由於 "Entity" 的成員僅輸入於 MR1，而 "Scenario" 的成員僅輸入於 MR2，這些成員將不會包含在 [IntersectMR] 中。

@IsAncest

如果指定的祖代成員是子成員的祖代，@ISANCEST 會傳回 *true*。

DTP 類型：

@ISANCEST 可指派給*布林值* 類型的 DTP。

語法：

`@ISANCEST(`*祖代成員*`/`*子成員*`)`

參數：

參數	描述
祖代成員	*成員* 類型的設計時期提示
子成員	*成員* 類型的設計時期提示

範例 1：

```
[True_False]=@ISANCEST([Anc_Mbr],[mbr])
```

其中：

- `[True_False]` 為非確認提示的*布林值* 類型的 DTP。
- `[Prt_Mbr]` 是*成員類型*類型的可提示 DTP。
- `[Mbr]` 是*成員類型*類型的可提示 DTP。

假設如下：

- `[Anc_Mbr]` 的輸入是 "Q1"。
- `[Mbr1]` 的輸入是 "Apr"。

在此範例中，`[True_False]` 會傳回 *false*。

@IsChild

如果指定的子成員是指定父成員的子項，`@ISCHILD` 會傳回 *true*。

DTP 類型：

`@ISCHILD` 可指派給*布林值* 類型的 DTP。

語法：

`@ISCHILD(`*父成員*`/`*子成員*`)`

參數：

參數	描述
父成員	*成員* 類型的設計時期提示
子成員	*成員* 類型的設計時期提示

範例 1：

```
[True_False]=@ISCHILD([Prt_Mbr],[mbr])
```

其中：

- `[True_False]` 為非確認提示的*布林值* 類型的 DTP。
- `[Prt_Mbr]` 是*成員類型*類型的可提示 DTP。

- `[Mbr]` 是*成員類型*類型的可提示 DTP。

假設如下：

- `[Prt_Mbr]` 的輸入是 "Q1"。
- `[Mbr]` 的輸入是 "Jan"。

在此範例中，`[True_False]` 會傳回 *true*。

@IsDataMissing

如果缺少資料塊的值，`@ISDATAMISSING` 會傳回 True。

DTP 類型：

`@ISDATAMISSING` 可指派至*布林值* 類型的 DTP。

語法：

`@ISDATAMISSING(`*`Members`*`)`

參數：

參數	描述
members	傳回跨維度的設計時期提示、成員或函式

範例：

```
@ISDATAMISSING([CD])=false
```

用於範本中步驟的函式，以啟用條件。

如果針對可提示設計時期提示 CD 輸入的成員帶有值，則會顯示步驟，如果成員的值為 `#Missing`，則不會顯示步驟

@IsSandBoxed

`@ISSANDBOXED` 判斷目前應用程式是否處理沙箱中。

DTP 類型：

`@ISSANDBOXED` 可指派至*布林值* 類型的 DTP。

語法：

`@ISSANDBOXED(`*`CubeName`*`)`

參數：

參數	描述
CubeName	設計時期提示或文字

範例：

```
[Sand1]=@ISSANDBOXED([cube1])
```

其中：

- `[Sand1]` 是非確認提示之*布林值* 類型的 DTP。
- `[cube1]` 是*字串*類型的可提示 DTP。

假設 `[cube1]:Plan1`。在此範例中，如果已針對沙箱啟用 "Plan1"，則 `[Sand1]` 會傳回 *true*，如果未針對沙箱啟用立方體 "Plan1"，則會傳回 *false*。

@IsVariable

`@ISVARIABLE` 決定引數是否為變數。

DTP 類型：

`@ISVARIABLE` 可指派至*布林值* 類型的 DTP。

語法：

`@ISVARIABLE(`*`Argument`*`)`

參數：

參數	描述
Argument	DTP 類型：Member、Members、String、Numeric、Percent、Integer、StringAsNumber、DateAsNumber

範例：

```
[isVar_mbr]=@ISVARIABLE([Mbr1])
```

其中：

- `[isVar_mbr]` 是非確認提示的*布林值* 類型的 DTP。
- `[Mbr1]` 是*成員類型*類型的可提示 DTP。

假設 `[Mbr1]` 的輸入是 "{Version}"。

在此範例中，`[isVar_mbr]` 會傳回 "true"。

@Length

`@LENGTH` 會傳回文字字串的長度。

DTP 類型：

可以將 `@LENGTH` 指派給*數值* 或是*整數* 類型的 DTP。

語法：

`@LENGTH(`*`Text`*`)`

參數：

參數	描述
Text	*字串* 類型的設計時期提示

範例：

```
[Len1]=@LENGTH([String1])
```

其中：

- `[Len1]` 是非勾選提示的數值設計時期提示
- `[String1]` 是可提示的設計時期提示類型字串

假設下列值為輸入：

`[String1]`: 今天是秋天的第一天

在本範例中，`Len1=30`

@Matches

如果字串的第一個子字串符合指定的一般表示式，`@MATCHES` 會傳回 "true"。

DTP 類型：

可以將 `@MATCHES` 指派給*字串*、*密碼*、*成員*、*多個成員*、*維度* 或*多個維度* 類型的 DTP。

語法：

`@MATCHES(text,regExpr,IgnoreCase)`

參數：

參數	描述
text	設計時期提示
regExpr	請參閱有關 "java.util.regex.Pattern" 的 Java 文件
IgnoreCase	**選擇性。**True 或 False。如果保留空白，預設值為 True。

範例：

```
[Matches_Mbr]=@MATCHES([Mbr],[Match_String_Mbr],"true")
```

其中：

- `[Mbr]` 是可提示的 DTP *成員類型*。
- `[Match_String_Mbr]` 是可提示的 DTP *字串*。
- `IgnoreCase` 為 "true"。

假設下列值為輸入：

- `[Mbr]`: "P_100"
- `[Match_String_Mbr]` "\p{Alnum}"，是會傳回英數字元的 Java 一般表示式。

在此範例中，`[Matches_Mbr]` 會傳回 "true"。

@Member

`@MEMBER` 會在成員有效時傳回成員。

DTP 類型：

可以將 `@MEMBER` 指派給*成員* 類型的 DTP。

語法：

`@MEMBER(Member)`

參數：

參數	描述
Member	設計時期提示、成員或其他設計時期提示函式

範例：

`[Mbr_mr1]=@UNION([MR1],@MEMBER([Member]))`

其中：

- `[Mbr_mr1]` 是非勾選提示之*成員範圍* 類型的 DTP。
- 只要輸入的成員是一個有效的成員，`[@UNION]` 會建立 DTP、MR1 及 DTP 成員的範圍。
- `[MR1]` 是勾選提示之*成員範圍* 類型的 DTP。
- `[Member]` 是勾選提示之*成員* 類型的 DTP。

假設下列值為輸入：

- `[MR1]`: Account "Salaries", Scenario "Actual", Entity "Tennessee"
- `[Member]`：「000」，是來自 Product 維度的成員

在此範例中，`[Mbr_mr1]` 會傳回 "Salaries","Actual","Tennessee","000"

@MemberGeneration

`@MEMBERGENERATION` 會傳回成員的層代編號。

DTP 類型：

可以將 `@MEMBERGENERATION` 指派給*數值* 或是*整數* 類型的 DTP。

語法：

`@MEMBERGENERATON(Member)`

參數：

參數	描述
Member	設計時期提示、成員或函式 (設計時期提示或函式應該傳回單一成員)

範例：

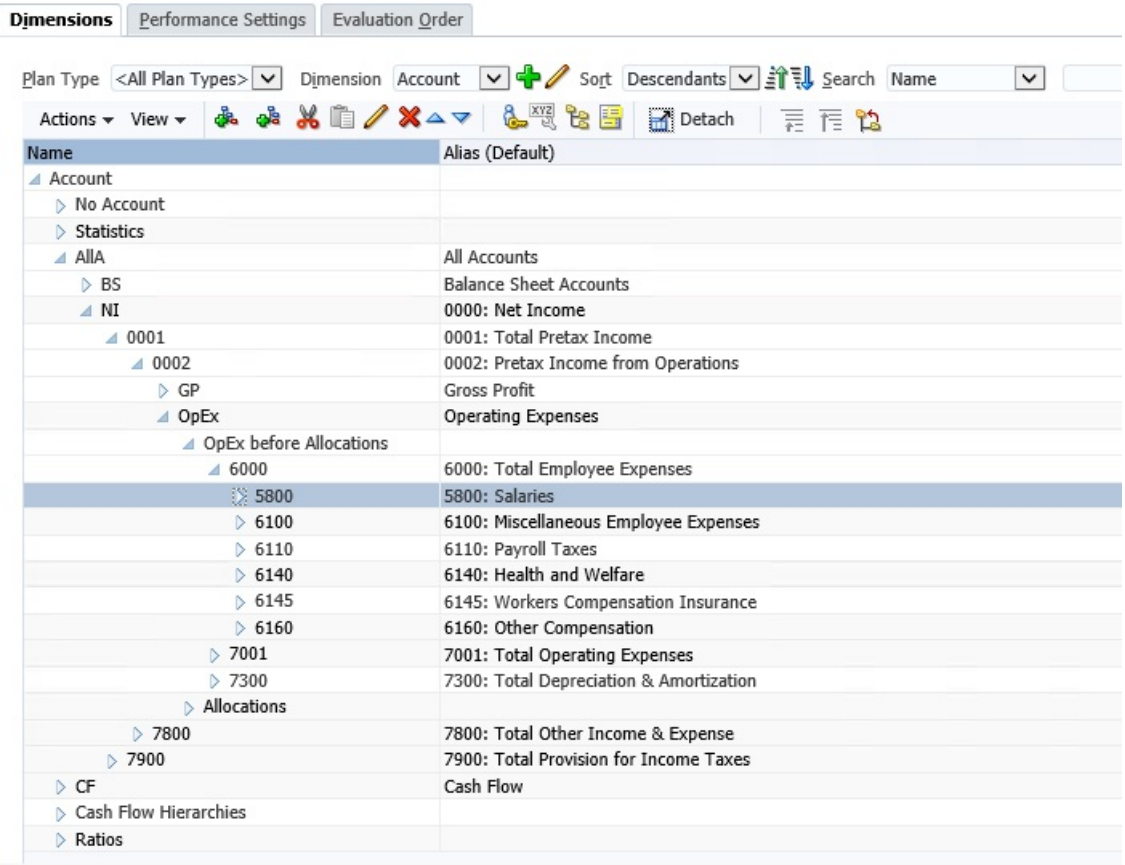
`[MBRGEN1]=@MEMBERGENERATION([member])`

其中：

- [MBRGEN1] 是非勾選提示之*數值* 類型的 DTP。
- [member] 是勾選提示之*成員* 類型的 DTP。

假設 [member]的輸入是 "5800"。

使用下列計畫大綱，[MBRGEN1] 會傳回 "9"。



@MemberLevel

@MEMBERLEVEL 會傳回成員的層級編號。

DTP 類型：

可以將 @MEMBERLEVEL 指派給*數值* 或是*整數* 類型的 DTP。

語法：

@MEMBERLEVEL(*Member*)

參數：

參數	描述
Member	設計時期提示、成員或函式 (設計時期提示或函式應該傳回單一成員)

範例：

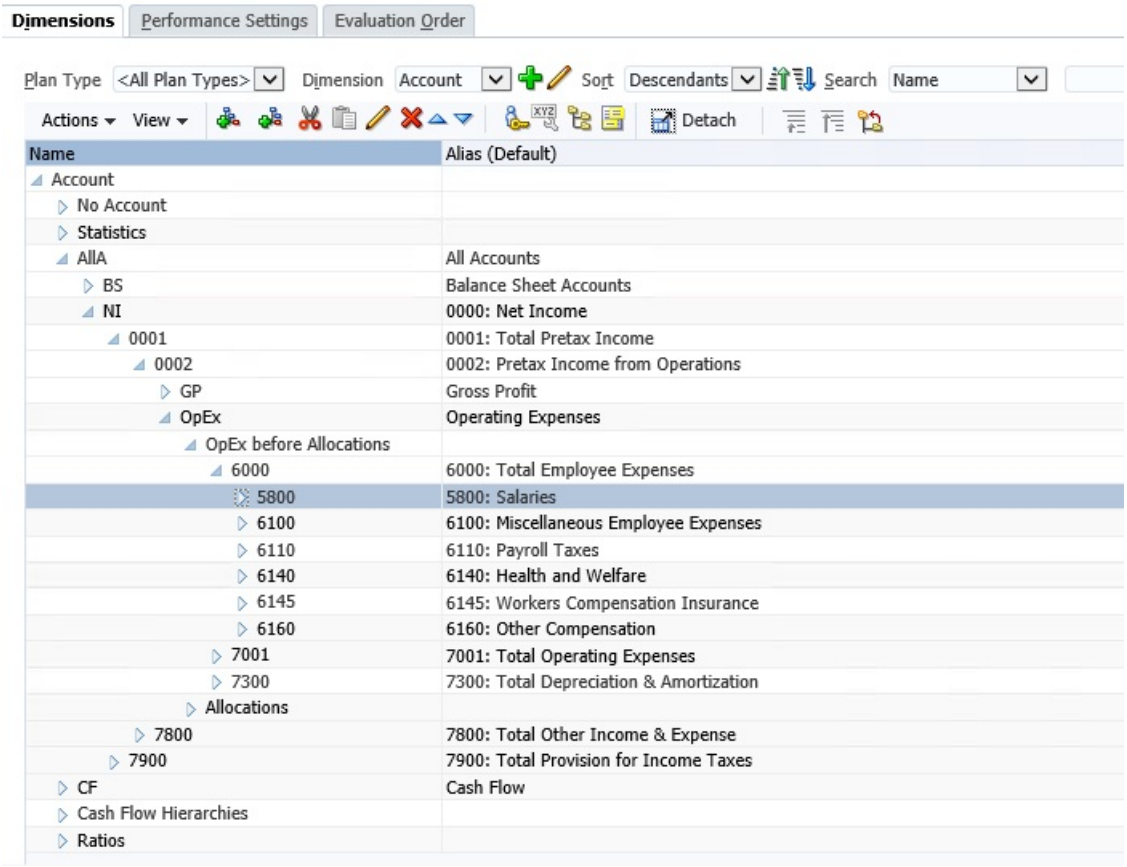
`[MBRLEV_MBR]=@MEMBERLEVEL([member])`

其中：

- `[MBRLEV_MBR]` 是非勾選提示之*數值* 類型的 DTP。
- `[member]` 是勾選提示之*成員* 類型的 DTP。

假設 `[member]`的輸入是 "5800"。

使用下列規劃大綱，`[MBRLEV_MBR]` 會傳回 0。



@MsgFormat

`@MSGFORMAT` 會採用一組物件，設定這些物件格式，然後將格式化的字串插入適當位置的模式中。(請參閱有關 "java.text.MessageFormat" 的 JAVA 文件。)

DTP 類型：

`@MSGFORMAT` 必須是*字串*。

語法：

`@MSGFORMAT(text,param1,param2,param3,param4)`

參數：

參數	描述
text	設計時期提示*字串*
param1	設計時期提示
param2	設計時期提示
param3	設計時期提示
param4	設計時期提示

範例：

```
[MSFG4]=@MSGFORMAT([String4],[FirsName],[LastName],[Month],[Year])
```

其中：

- `[MSFG4]` 是非勾選提示之*字串* 的 DTP。
- `[String4]` 是可提示的 DTP *字串*。
- `[FirstName]` 是可提示的 DTP *字串*。
- `[LastName]` 是可提示的 DTP *字串*。
- `[Month]` 是可提示的 DTP *成員*，包含為維度選取的 "Period"。
- `[Year]` 是可提示的 DTP *成員*，包含為維度選取的 "Year"

假設下列值為輸入：

- `[String4]`: "{0} {1} completed this task in {2} {3}"
- `[FirstName]`: "Joe"
- `[LastName]`: "Smith "
- `[Month]`: "Feb"
- `[Year]`: "FY15"

在此範例中，`[MSFG4]` 會傳回 "Joe Smith completed this task in Feb "FY15"。

@Notin

`@NOTIN` 會傳回屬於某個表示式，但不屬於另一個表示式的成員。換句話說，`@NOTIN` 會傳回非兩個表示式共同的成員。

DTP 類型：

可以將 `@NOTIN` 指派給*成員範圍* 或是*跨維度* 類型的 DTP。

語法：

```
@NOTIN(Argument1,Argument2)
```

參數：

參數	描述
Argument1	設計時期提示、函式或是成員
Argument2	設計時期提示、函式或是成員

範例：

```
[NOTIN1]=@NOTIN([MR1],[MR2])
```

其中：

- `[NOTIN1]` 是非勾選提示之*成員範圍*類型的 DTP。
- `[MR1]` 是勾選提示之*成員範圍*類型的 DTP。
- `[MR2]` 是勾選提示之*成員範圍*類型的 DTP。

假設 `MR1` 的輸入是：

- Period: "Jan", "Feb", "Mar"
- Scenario: "Actual"
- Currency: "Local"
- Product: "000"

假設 `MR2` 的輸入是：

- Period: "Feb"
- Version: "Working"
- Currency: "Local"
- Product: "000"

在此範例中，`[@NOTIN1]` 會傳回 "Actual"，因為這是唯一不屬於 `MR2`，但屬於 `MR1` 的成員。

@OpenDimCount

`@OPENDIMCOUNT` 傳回未指定成員的維度數量。

DTP 類型：

可以將 `@OPENDIMCOUNT` 指派給*數值*類型的 DTP。

語法：

```
@OPENDIMCOUNT(DTP,Dense/Sparse)
```

參數：

參數	描述
DTP	傳回成員範圍或跨維度的設計時期提示或函式。
`Dense/Sparse`	**選擇性。**如果保留空白，會傳回未指定成員的所有可用維度計數。如果輸入 "Dense" 或 "Sparse"，只會計算沒有成員的特定類型維度。

範例 1：

```
[DTP1]=@OPENDIMCOUNT([MR1])
```

其中：

- `[DTP1]` 是*數值* 類型的 DTP。
- `[MR1]` 是具有下列輸入之*成員範圍* 類型的 DTP：
  - Account：
  - Period：Jan, Feb, Mar
  - HSP_View：
  - Year：
  - Scenario：
  - Version：Working
  - Entity：Sparse
  - Product：P-100

在此範例中，`[DTP1]` 會傳回 "5"，因為總共有八個維度，但是只有三個有成員輸入。

範例 2：

```
[DTP2]=@OPENDIMCOUNT([MR1],"Dense")
```

其中：

- `[DTP2]` 是*數值* 類型的 DTP。
- `[MR1]` 是具有下列輸入之*成員範圍* 類型的 DTP：
  - Account：Dense
  - Period：Dense
  - HSP_View：Sparse
  - Year：Sparse
  - Scenario：Sparse
  - Version：Sparse
  - Entity：Sparse
  - Product：Sparse

在此範例中，`[DTP2]` 會傳回 "1"，因為 "Account" 與 "Period" 為密集，而 "Period" 具有成員輸入。

@Plandim

`@PLANDIM` 會在應用程式的資料庫中有維度名稱時將其傳回。

DTP 類型：

可以將 `@PLANDIM` 指派給*成員* 或*維度* 類型的 DTP。`@PLANDIM` 僅在第二個參數使用一個值時才會運作。

語法：

`@PLANDMIN(Database Name,Dimension)`

參數：

參數	描述
Database Name	設計時期提示或文字
Dimension	設計時期提示、維度或文字

範例：

`[PDIM_mbr]=@PLANDIM([PlTy],[Dim])`

其中：

- `[PDIM_mbr]` 是非勾選提示之*成員* 類型的 DTP。
- `[PlTy]` 是勾選提示之*字串* 類型的 DTP。
- `[Dim]` 是勾選提示之*維度* 類型的 DTP。

假設下列值為輸入：

- `[PlTy]`: "Plan1"
- `Dim`: "Product"

在此範例中，`[PDIM_mbr]` 會傳回 "Product"，因為 Product 是 "Plan1" 中有效的維度。

@PlanDimMember

`@PLANDIMMEMBER` 傳回成員，如果該成員在指定計畫類型對指定維度為有效的話。

DTP 類型：

可以將 `@PLANDIMMEMBER` 指派給*成員*類型的 DTP。

語法：

`@PLANDIMMEMBER(`*計畫類型*、*維度*、*成員*`)`

參數：

參數	描述
計畫類型	成員計畫類型的設計時期提示
維度	傳回維度名稱的設計時期提示
成員	傳回成員名稱的設計時期提示

範例 1：

`[DIM_mbr1]=@ PLANDIMMEMBER("Plan1",[DIM],[Member])`

其中：

- `[DIM_mbr1]` 是非確認提示之*成員* 類型的 DTP。

- [DIM] 是確認提示之*維度* 類型的 DTP。
- [Member] 是確認提示之*成員* 類型的 DTP。
- [Dim]="Entity"
- [Member]="Washington"

@Quote

@QUOTE 會在字串前後加上雙引號。

DTP 類型：

可以將 @QUOTE 指派給*成員*、*字串* 或是*密碼* 類型的 DTP。

語法：

@QUOTE(*String*)

參數：

參數	描述
String	設計時期提示或文字

範例：

[Quote_St]=@QUOTE([String1])

其中：

- [Quote_St] 是勾選提示之*字串* 類型的 DTP。
- [String1] 是勾選提示之*字串* 類型的 DTP。

假設 [String1] 的輸入是 "Yellow"。

在此範例中，[Quote_St] 會傳回 "Yellow"。

@RemoveQuote

@REMOVEQUOTE 將移除字串的雙引號。

DTP 類型：

可以將 @REMOVEQUOTE 指派給*成員*、*字串* 或是*密碼* 類型的 DTP。

語法：

@REMOVEQUOTE(*String*)

參數：

參數	描述
String	設計時期提示或文字

範例：

```
[RemoveQuote_St]=@REMOVEQUOTE([String1])
```

其中：

- `[RemoveQuote_St]` 是*字串*類型的 DTP，而且是非勾選提示。
- `[String1]` 是勾選提示之*字串*類型的 DTP。

假設 `[String1]` 的輸入是 "Yellow"。

在此範例中，`[RemoveQuote_St]` 會傳回 "Yellow"。

@ReplaceAll

`@REPLACEALL` 將以取代字串來取代字串中含有此表示式的部分。

DTP 類型：

可以將 `@REPLACEALL` 指派給*字串*、*密碼*、*成員*、*多個成員*、*維度*或*多個維度*類型的 DTP。

語法：

`@REPLACEALL(String,Regular Expression,replaceString)`

參數：

參數	描述
String	設計時期提示或文字
Regular Expression	設計時期提示或文字
replaceString	設計時期提示或文字

範例：

```
[ReplaceAll_Str]=@REPLACEALL([String1],[Rep_String],[new_String])
```

其中：

- `[ReplaceAll_Str]` 是非勾選提示之*字串*類型的 DTP。
- `[String1]` 是勾選提示之*字串*類型的 DTP。
- `[Rep_String]` 是勾選提示之*字串*類型的 DTP。
- `[new_String]` 是勾選提示之*字串*類型的 DTP。

假設參數如下：

- 字串："This is a test of a testing string"
- 一般表示式："Test"
- 取代字串 "XYZ"

在此範例中，`[ReplaceAll_Str]` 會傳回："This is a XYZ of a XYZing string for one XYZ and two XYZ"。它將以 "test with XYZ" 取代。

@ReplaceFirst

`@REPLACEFIRST` 會以取代字串取代第一個出現的一般表示式。

DTP 類型：

可以將 `@REPLACEFIRST` 指派給*字串*、*密碼*、*成員*、*多個成員*、*維度* 或*多個維度* 類型的 DTP。

語法：

`@REPLACEFIRST(String,Regular Expression,replaceString)`

參數：

參數	描述
String	設計時期提示或文字
Regular Expression	設計時期提示或文字
replaceString	設計時期提示或文字

範例：

`[ReplaceFirst_Str]=@REPLACEFIRST([String1],[Rep_String],[new_String])`

其中：

- `[ReplaceFirst_Str]` 是非勾選提示之*字串* 類型的 DTP。
- `[String1]` 是勾選提示之*字串* 類型的 DTP。
- `[Rep_String]` 是勾選提示之*字串* 類型的 DTP。
- `[new_String]` 是勾選提示之*字串* 類型的 DTP。

假設參數如下：

- 字串："This is a testing of the new test of a test today"
- 一般表示式："Test"
- 取代字串："XYZ"

在此範例中，`[ReplaceFirst_Str]` 傳回："This is a XYZing of the new test of a test today"。它只會取代測試中*第一個*出現的項目，不會取代所有出現的項目。

@SmartListFromIndex

`@SMARTLISTFROMINDEX` 會根據 SmartList 中的成員順序傳回智慧型列示成員。

DTP 類型：

可以將 `@SMARTLISTFROMINDEX` 指派給*字串*、*密碼*、*成員* 或是*多個成員* 類型的 DTP。

語法：

`@SMARTLISTFROMINDEX(SmartList,Index)`

參數：

參數	描述
SmartList	設計時期提示或維度維度會顯示應用程式中的智慧型列示維度。
Index	設計時期提示或文字

範例：

```
[sl_ind_Str]=@SmartListFromIndex([String1],[index_ST])
```

其中：

- `[sl_ind_Str]` 是非勾選提示之*字串*類型的 DTP。
- `[String1]` 是勾選提示之*字串*類型的 DTP。
- `[index_ST]` 是勾選提示之*數值*類型的 DTP。

假設參數如下：

- 智慧型列示："CreditRatings"
- 索引："2"

在此範例中，智慧型列示 "CreditRatings" 具有下列資料：

名稱	標籤
AA	ID_ENUMNETLABEL_AA
AAA	ID_ENUMNETLABEL_AAA
AAMinus	ID_ENUMNETLABEL_AA22
A	ID_ENUMNETLABEL_A
AMinus	ID_ENUMNETLABEL_AA22
AAPlus	ID_ENUMNETLABEL_AA11
APlus	ID_ENUMNETLABEL_A11
B	ID_ENUMNETLABEL_B
BPlus	ID_ENUMNETLABEL_B11
BMinus	ID_ENUMNETLABEL_B22
BB	ID_ENUMNETLABEL_BB
BBPlus	ID_ENUMNETLABEL_BB11
BBMinus	ID_ENUMNETLABEL_BB22
BBB	ID_ENUMNETLABEL_BBB
AAAMinus	ID_ENUMNETLABEL_AAA-
BBBPlus	ID_ENUMNETLABEL_BBB+
BBBMinus	ID_ENUMNETLABEL_BBB-
CCCPlus	ID_ENUMNETLABEL_CCC+
CCC	ID_ENUMNETLABEL_CCC

`[sl_ind_Str]` 會傳回 "AAMinus"，因為其為第二個索引位置處的智慧型列示成員名稱。

@SmartListFrom 值

`@SMARTLISTFROMVALUE` 會根據 SmartList 中的成員 ID 傳回智慧型列示成員。

DTP 類型：

如果僅輸入一個成員，可以將 `@SMARTLISTFROMVALUE` 指派給*字串*、*密碼*、*成員* 或是*多個成員* 類型的 DTP。

語法：

`@SMARTLISTFROMVALUE(`*`SmartList`*`,`*`Value`*`)`

參數：

參數	描述
Smartlist	設計時期提示或維度此維度會顯示應用程式中的智慧型列示維度。
Value	設計時期提示或維度

範例：

```
[sl_val_Str]=@SmartListFromValue([String1],[value_ST])
```

其中：

- `[sl_val_Str]` 是非勾選提示之*字串* 類型的 DTP。
- `[String1]` 是勾選提示之*字串* 類型的 DTP。
- `[value_ST]` 是勾選提示之*數值* 類型的 DTP。

假設參數如下：

- 智慧型列示："CreditRatings"
- 值："5"

在此範例中，智慧型列示 "CreditRatings" 具有下列資料：

名稱	標籤
AA	ID_ENUMNETLABEL_AA
AAA	ID_ENUMNETLABEL_AAA
AAMinus	ID_ENUMNETLABEL_AA22
A	ID_ENUMNETLABEL_A
AMinus	ID_ENUMNETLABEL_A22
AAPlus	ID_ENUMNETLABEL_AA11
APlus	ID_ENUMNETLABEL_A11
B	ID_ENUMNETLABEL_B
BPlus	ID_ENUMNETLABEL_B11
BMinus	ID_ENUMNETLABEL_B22
BB	ID_ENUMNETLABEL_BB
BBPlus	ID_ENUMNETLABEL_BB11
BBMinus	ID_ENUMNETLABEL_BB22
BBB	ID_ENUMNETLABEL_BBB
AAAMinus	ID_ENUMNETLABEL_AAA-
BBBPlus	ID_ENUMNETLABEL_BBB+
BBBMinus	ID_ENUMNETLABEL_BBB-
CCCPlus	ID_ENUMNETLABEL_CCC+

名稱	標籤
CCC	ID_ENUMNETLABEL_CCC

`[sl_val_Str]` 傳回 "AMinus"，因為這是 ID 為 5 的智慧型列示成員名稱。

@SparseMember

`@SPARSEMEMBER` 會傳回跨維度中的第一個稀疏維度成員。

DTP 類型：

可以將 `@SPARSEMEMBER` 指派給*成員*、*成員*、*維度*、*多個維度*、*成員範圍*、*跨維度* 或是 *字串* 類型的 DTP。

語法：

`@SPARSEMEMBER(Members)`

參數：

參數	描述
Members	傳回成員清單的設計時期提示、成員或函式

範例：

`[SparseMbr]=@SPARSEMEMBER([CD1])`

其中：

- `[SparseMbr]` 是非勾選提示之*跨維度* 類型的 DTP。
- `[CD1]` 是可提示的*跨維度* 類型的 DTP。
- `[CD1]` 是 "Gross Sales"->"Jan"->"FY14"->"Working"->"California"

在此範例中，`[Sparsembr]` 會傳回 "FY14"，因為 FY14 是跨維度清單中的第一個稀疏成員。

@StartsWith

`@STARTSWITH` 測試此字串是否以指定的前綴開始。

DTP 類型：

可以將 `@STARTSWITH` 指派給*布林值* 類型的 DTP。

語法：

`@STARTSWITH(String,Prefix)`

參數：

參數	描述
String	設計時期提示或文字
Prefix	設計時期提示或文字

範例：

```
[DTP_Start]=@STARTSWITH([Mbr1],[String1])
```

其中：

- `[Mbr1]` 是勾選提示之*成員* 類型的 DTP。
- `[String1]` 是勾選提示之*字串* 類型的 DTP。

假設下列值為輸入：

- `[Mbr1]`: "Joe Smith"
- `[String1]`: "Smith"

在此範例中，`[DTP_Start]` 會傳回 "false"。

@SubString

`@SUBSTRING` 將傳回以索引開頭的字元開始，並延伸至索引結尾前字元的子字串。索引開頭以零開始，且傳回的文字不包括索引結尾。如果索引結尾大於字串中的字元數目，將不會傳回任何字元。

DTP 類型：

可以將 `@SUBSTRING` 指派給*成員*、*多個成員*、*維度*、*多個維度*、*字串* 或*密碼* 類型的 DTP。

語法：

```
@SUBSTRING(String,Beginning Index,Ending Index)
```

參數：

參數	描述
String	設計時期提示或文字
Beginning Index	設計時期提示*數值* 或*文字*
Ending Index	*數值* 或*文字* 的設計時期提示

範例：

```
[SUBSTRING_Str]=@SUBSTRING([String1],[start_ind],[end_ind])
```

其中：

- `[SUBSTRING_Str]` 是非勾選提示之*字串* 類型的 DTP。
- `[String1]` 是勾選提示之*字串* 類型的 DTP。
- `[start_ind]` 是勾選提示之*數值* 類型的 DTP。
- `[end_ind]` 是勾選提示之*數值* 類型的 DTP。

假設參數如下：

- 字串："This is a test of converting characters to strings."
- 開始索引："2"
- 結束索引："16"

在這個範例中，`[SUBSTRING_Str]` 會傳回 "his is a test o"，因為字串中的第 2 個字元是 "this" 中的 "h"，且第 15 個字元是 "of" 這個字開頭的 "o"。

@ToLowerCase

`@TOLOWERCASE` 將使字串中的字元變為小寫。

DTP 類型：

可以將 `@TOLOWERCASE` 指派給*成員*、*多個成員*、*維度*、*多個維度*、*字串* 或*密碼* 類型的 DTP。

語法：

```
@TOLOWERCASE(String)
```

參數：

參數	描述
String	設計時期提示或文字

範例：

```
[TOLOWERCASE_Str]=@TOLOWERCASE([String1])
```

其中：

- `[TOLOWERCASE_Str]` 是非勾選提示之*字串* 類型的 DTP。
- `[String1]` 是勾選提示之*字串* 類型的 DTP。

假設參數如下：

字串："Testing for Today"

在此範例中，`[TOLOWERCASE_Str]` 會傳回 "Testing for Today"

@ToMDX

`@TOMDX` 會傳回 MDX 表示式。

DTP 類型：

可以將 `@TOMDX` 指派給*字串* 類型的 DTP。

語法：

```
@TOMDX(Members, Filter Shared, Non Empty, Generate Crossjoin)
```

參數：

參數	描述
Members	設計時期提示、成員或函式 (設計時期提示或函式會傳回一個成員或多個成員)
Filter Shared	**選擇性。**True 或 False。MDX 語法新增 True，以篩選共用成員。預設值是 False。
Non Empty	**選擇性。**True 或 False。`NON EMPTY` 在 MDX 語法前面新增 True。預設值是 False。

參數	描述
Generate Crossjoin	**選擇性**。True 或 False。True 會傳回含交叉聯結的 MDX 語法。

範例：

```
[MDX_function1]=@TOMDX(@UNION([MR],[MR2]),"true","true","true")
```

```
[MDX_function2]=@TOMDX(@UNION([MR],[MR2]),"false","false","false")
```

其中：

- `[MDX_function]` 是非可提示的 DTP *字串*。
- `[Mbr]` 是可提示的 DTP *成員類型*。
- `[Mbr]` 是可提示的 DTP *成員範圍*。

假設 `[MR1]` 具有下列輸入：

- Account：@Relative("Gross Margin, %, 0")
- Period：「四月」、「五月」
- HSP_View：「BaseData」
- Year：「FY15」
- Scenario：「計畫」、「實際」
- Version：「作業中」
- Entity：@Relative("100", 0)
- Product：「P_110」、「P_150」、「P-100」

假設 `[MR2]` 具有下列輸入：

- Account：
- Period：「一月、二月、三月」
- HSP_View：
- Year：
- Scenario：
- Version：「最終」
- Entity：
- Product：

在此範例中：

`[MDX_function1]` 會傳回：

```
NON EMPTY
(Crossjoin(Crossjoin(Crossjoin(Crossjoin(Crossjoin(Crossjoin(Crossjoin(FILTER
({RELATIVE([Gross Margin %],  0}, NOT
[Account].CurrentMember.SHARED_FLAG),FILTER ({[Apr],[May],Jan,Feb,Mar}, NOT
[Period].CurrentMember.SHARED_FLAG)),FILTER ({[BaseData]}, NOT
[HSP_View].CurrentMember.SHARED_FLAG)),FILTER ({[FY15]}, NOT
```

```
[Year].CurrentMember.SHARED_FLAG)),FILTER ({[Plan],[Actual]}, NOT
[Scenario].CurrentMember.SHARED_FLAG)),FILTER ({[Working],[Final]}, NOT
[Version].CurrentMember.SHARED_FLAG)),FILTER ({[110]}, NOT
[Entity].CurrentMember.SHARED_FLAG)),FILTER ({[P_110],[P_150],[P_000]}, NOT
[Product].CurrentMember.SHARED_FLAG)))
```

`[MDX_function2]` 會傳回：

```
[@Relative("Gross Margin %", 0)],[Apr","May",Jan,Feb,Mar],[BaseData],[FY15],
[Plan","Actual],[Working","Final],[110],[P_110","P_150","P_000]
```

@ToUpperCase

`@TOUPPERCASE` 將使字串中的字元變為大寫。

DTP 類型：

可以將 `@TOUPPERCASE` 指派給*成員*、*多個成員*、*維度*、*多個維度*、*字串* 或*密碼* 類型的 DTP。

語法：

```
@TOUPPERCASE(String)
```

參數：

參數	描述
String	設計時期提示或文字

範例：

```
[TOUPPERCASE_Str]=@TOUPPERCASE([String1])
```

其中：

- `[TOUPPERCASE_Str]` 是非勾選提示之*字串* 類型的 DTP。
- `[String1]` 是勾選提示之*字串* 類型的 DTP。

假設參數如下：

字串："Testing to convert to upper case."

在此範例中，`[TOUPPERCASE_Str]` 會傳回 "TESTING TO CONVERT TO UPPER CASE."

@Trim

`@TRIM` 將移除前置空格或後置空格。

DTP 類型：

可以將 `@TRIM` 指派給*成員*、*多個成員*、*字串* 或是*密碼* 類型的 DTP。

語法：

```
@TRIM(String)
```

參數：

參數	描述
String	設計時期提示或文字

範例：

```
[TRIM_ST]=@TRIM([String1])
```

其中：

- `[TRIM_ST]` 是非勾選提示之*字串*類型的 DTP。
- `[String1]` 是勾選提示之*字串*類型的 DTP。

假設參數如下：

字串："This is a test of leading and trailing spaces"

此參數在文字的前後有空格。將顯示為：

```
String1 =       this is a test of leading and trailing spaces
```

@Union

`@UNION` 會傳回用於多個設計時期提示的成員組合。

DTP 類型：

可以將 `@NOTIN` 指派給*成員範圍*或是*跨維度*類型的 DTP。

語法：

`@UNION(`*`Argument1`*`,`*`Argument2`*`)`

參數：

參數	描述
Argument1	設計時期提示、函式或是成員
Argument2	設計時期提示、函式或是成員

範例：

```
[UnionMR]=@UNION([MR1],[MR2])
```

其中：

- `[UnionMR]` 是非勾選提示之*成員範圍*類型的 DTP。
- `[MR1]` 是勾選提示之*成員範圍*類型的 DTP。
- `[MR2]` 是勾選提示之*成員範圍*類型的 DTP。

假設 `MR1` 與 `MR2` 的輸入是：

- `[MR1]`: "50350",@Relative("Q1", 0),"California","Tennessee"

- [MR2]: "Apr","Washington", "Tennessee"

在此範例中，UnionMR 會傳回：

"50350",@Relative ("Q1", 0),"Apr","California","Tennessee","Washington"

@ValueDimCount

@VALUEDIMCOUNT 會傳回指定成員的維度數目。

DTP 類型：

可以將 @VALUEDIMCOUNT 指派給*數值* 類型的 DTP。

語法：

(@VALUEDIMCOUNT(*DTP*,Dense/Sparse)

參數：

參數	描述
DTP	傳回成員範圍或跨維度的設計時期提示或函式。
Dense/Sparse	**選擇性。**如果保留空白，會傳回已指定成員的所有維度計數。如果輸入 "Dense" 或 "Sparse"，只會計算已指定成員的特定類型維度。

範例 1：

[DTP1]=@VALUEDIMCOUNT([MR1])

其中：

- [DTP1] 是*數值* 類型的 DTP，而且不是勾選提示。
- [MR1] 是具有下列輸入之*成員範圍* 類型的 DTP：
  - Account：
  - Period：Jan、Feb、Mar
  - HSP_View：
  - Year：
  - Scenario：
  - Version：Working
  - Entity：Sparse
  - Product：P-100

在此範例中，[DTP1] 會傳回 "3"，因為已經有三個已指定成員的維度。

範例 2：

[DTP2]=@VALUEDIMCOUNT([MR1],"Dense")

其中：

- [DTP2] 是*數值* 類型的 DTP，而且不是勾選提示。
- [MR1] 是具有下列輸入之*成員範圍* 類型的 DTP：

- Account：Dense
- Period：Dense
- HSP_View：Sparse
- Year：Sparse
- Scenario：Sparse
- Version：Sparse
- Entity：Sparse
- Product：Sparse

在此範例中，`[DTP2]` 會傳回 "1"，因為 "Account" 與 "Period" 為密集，而 "Period" 已指定成員。

共用指令碼和公式元件

共用公式或指令碼元件存在於公式層級和指令碼層級。

另請參閱：

- 關於共用指令碼和公式元件
  您可以在屬於相同應用程式類型的 Oracle Hyperion Financial Management、Planning 及 Oracle Essbase 區塊儲存商業規則和範本之間共用公式和指令碼元件。
- 將公式元件和指令碼元件從共用更改為不共用
- 將公式元件和指令碼元件從不共用變更為共用

關於共用指令碼和公式元件

您可以在屬於相同應用程式類型的 Oracle Hyperion Financial Management、Planning 及 Oracle Essbase 區塊儲存商業規則和範本之間共用公式和指令碼元件。

您可以跨 Planning 商業規則與範本共用公式和指令碼元件。

> **備註：**
>
> 您不能共用成員範圍、條件、固定迴圈、成員區塊或「DTP 指派」元件。

共用公式或指令碼元件存在於公式層級和指令碼層級。共用公式和指令碼元件會被使用，因此，當您變更公式或指令碼時，變更反映在使用共用元件的所有規則和範本中。當共用元件時，系統將建立指向原始元件的交叉參照。透過建立指向原始元件的交叉參照而非建立其副本，能減少在資料庫中使用的空間，並且可能會縮短處理時間。

將公式元件和指令碼元件從共用更改為不共用

在將共用公式元件或指令碼元件變更為不共用之前，必須先確定該元件沒有用於多個商業規則或範本。您可以使用「顯示使用狀況」功能來檢視哪些商業規則和範本使用了該公式元件或指令碼元件。請參閱顯示公式元件和指令碼元件的使用情況。然後您可以透過從規則和範本中清除共用元件的「共用」核取方塊，為使用該元件的每個商業規則和範本建立該共用元件的副本。

若要將公式元件或指令碼元件從共用變更為不共用，請執行下列動作：

1. 在**系統檢視**中，在規則或範本上按一下右鍵，然後選取**開啟**。
2. 在**規則設計工具**或**範本設計工具**中，執行下列其中一項任務：
   - 如果不希望共用新增到流程圖中的公式元件或指令碼元件，請在將公式元件或令碼元件拖放到流程圖後，清除**特性**中的**共用**。
   - 如果不希望共用已存在於流程圖中的公式元件或指令碼元件，請在流程圖中選取該元件，然後清除**特性**中的**共用**。

   這將在規則或範本中建立該元件的副本。
3. 按一下 。

將公式元件和指令碼元件從不共用變更為共用

若要將公式或指令碼元件從不共用變更為共用，您必須確定*在規劃應用程式類型內*，沒有另一個具有相同名稱的共用或不共用公式或指令碼元件。共用物件在多個應用程式之間必須具有唯一的名稱，所以建立共用物件時不能使用已使用過的名稱。

若要共用公式元件或指令碼元件，請確定在將現有元件拖放到某個規則或範本的流程圖時，在該元件的「特性」頁籤上已選取「共用」核取方塊。(預設情況下會選取「共用」核取方塊。)

若要將公式元件或指令碼元件從不共用變更為共用，請執行下列動作：

1. 在**系統檢視**中，在規則或範本上按一下右鍵，然後選取**開啟**。
2. 開啟**規則設計工具**或**範本設計工具**時，執行下列其中一項任務：
   - 若要共用公式元件或指令碼元件，請執行下列動作：
     - 在**新物件**中，將公式或指令碼元件拖放到流程圖內的所要位置。
     - 在元件的**特性**頁籤中，選取**共用**，然後指定元件名稱。
   - 若要共用已存在於流程圖中的公式元件或指令碼元件，請執行下列動作：
     a. 選取流程圖中的元件。
     b. 在元件的**特性**頁籤中，選取**共用**，然後指定元件名稱。所有共用元件都必須要有名稱。
3. 選取**儲存**。

複製元件

複製並貼上商業規則項目的子項，或複製參照，並將其貼到商業規則公式或指令碼元件。

另請參閱：

- 複製和貼上商業規則元件的子項
- 複製並貼上對商業規則公式或指令碼元件的參照

複製和貼上商業規則元件的子項

當您使用含有元件的商業規則時，您可能想要複製元件的子項 (即群組在流程圖中某個元件下方的元件)。您可以複製元件的子項，然後貼到同一個商業規則或者不同的商業規則中。

若要複製和貼上商業規則元件的子項，請執行下列動作：

1. 開啟商業規則。
2. 在規則設計工具流程圖中，用滑鼠右鍵按一下您要複製之子項的成員範圍、固定迴圈，或條件元件，然後選取**複製群組**。
3. 執行下列其中一項任務：
   - 若要將元件子項貼到*相同*商業規則中，請在流程圖中用滑鼠右鍵按一下位於元件子項預期顯示位置左側的元件，然後選取**貼上**。(元件的子項顯示在您所選元件的右側。)
   - 如果您要將元件的子項貼到*另一個*商業規則，請開啟您要將元件的子項貼到其中的商業規則，然後在流程圖中用滑鼠右鍵按一下位於元件子項預期顯示位置左側的元件，然後選取**貼上**。
4. 按一下 。

> **備註：**
>
> 複製和貼上元件的子項時，也會一起複製所有的共用元件。

複製並貼上對商業規則公式或指令碼元件的參照

複製並貼上對商業規則共用指令碼或共用公式元件的參照時，是將捷徑複製並貼到元件，元件本身並不會複製。在您複製參照的原始商業規則中，元件只能有一個副本。參照的功能是指向包含該共用元件的應用程式和計畫類型。如果您在未共用的元件上選取複製參照，則貼上的元件為元件副本，並且無任何捷徑至原始元件。在此例項中，複製參照的表現與複製相同。

您可以複製參照並貼到相同商業規則或不同商業規則內的共用公式或共用指令碼元件中。

若要複製參照並將其貼至共用公式或共用指令碼元件，請執行下列動作：

1. 用滑鼠右鍵按一下包含共用公式或共用指令碼元件的規則，然後選取**開啟**。
2. 在「規則設計工具」流程圖中，用滑鼠右鍵按一下共用公式或共用指令碼元件，然後選取**複製參照**。
3. 執行下列其中一項任務：
   - 如果您要將元件的參照貼到*相同*商業規則中，請在流程圖中用滑鼠右鍵按一下位於元件參照的預期顯示位置左側的元件。(元件的參照顯示在選取元件的右側。) 然後選取**貼上**。
   - 如果您要將元件的參照貼到*另一個*商業規則，請開啟您要將元件的參照貼到其中的商業規則，然後在流程圖中用滑鼠右鍵按一下位於元件參照的預期顯示位置左側的元件。然後按一下**貼上**。

   > **備註：**
   >
   > 您可以複製參照並貼到屬於相同應用程式或不同應用程式的商業規則中，只要應用程式屬於相同的應用程式類型。

4. 按一下 。

正在儲存元件

在「元件設計工具」中設計公式元件和指令碼元件後，您需要儲存這些元件。

另請參閱：

- 儲存元件
- 使用不同的名稱儲存公式元件和指令碼元件

儲存元件

在「元件設計工具」中設計公式元件和指令碼元件後，您需要儲存這些元件。在「規則設計工具」或「範本設計工具」中儲存條件元件、範圍元件和固定迴圈元件所屬的商業規則或範本時，您同時也儲存了這些元件。

如果您使用元件，可以將它儲存為自訂範本。如果正在使用公式元件或指令碼元件，可以使用不同的名稱儲存該元件。

若要儲存元件，當您在「元件設計工具」、「範本設計工具」或「規則設計工具」中完成元件設計後，按一下 。

使用不同的名稱儲存公式元件和指令碼元件

您可以使用「另存新檔」以不同的名稱儲存指令碼元件和公式元件。「另存新檔」會建立公式元件或指令碼元件的副本。如果元件是共用元件，而您不想要共用該元件，那麼您需要建立該元件的副本。請參閱共用指令碼和公式元件。

若要使用不同的名稱儲存指令碼元件或公式元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下公式或是指令碼，然後選取**開啟**。
2. 在元件設計工具中，選取**檔案**，然後選取**另存新檔**。
3. 在**另存新檔**中，輸入公式或指令碼的新名稱，然後選取應用程式。然後選取**計畫類型**。
4. 按一下**確定**。

   儲存後，您可能必須在「系統檢視」中重新整理應用程式清單，才能查看公式或指令碼元件。請參閱重新整理公式元件和指令碼元件。

重新整理公式元件和指令碼元件

建立公式元件或指令碼元件後，您必須重新整理「系統檢視」中的應用程式清單，才能在「公式」或「指令碼」節點中看到該項目。

將應用程式類型、應用程式或者計算或計畫類型重新整理為所屬的公式元件或指令碼元件時，預設會重新整理公式元件及指令碼元件。然而，重新整理公式元件或指令碼元件並不會重新整理應用程式清單中較高的層級 (亦即計算或計畫類型、應用程式或應用程式類型)。

若要重新整理公式或指令碼元件，在**系統檢視**中，用滑鼠右鍵按一下**指令碼**或是**公式**，然後選取**重新整理**。

> **備註：**
>
> 您也可以重新整理資料庫大綱中的較高層級，以重新整理其中的物件。例如，若要為應用程式重新整理指令碼和公式，請用滑鼠右鍵按一下應用程式名稱，然後選取**重新整理**。

顯示公式元件和指令碼元件的使用情況

您可以顯示指令碼或公式元件的使用情況。

若要顯示指令碼元件或公式元件的使用情況，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下指令碼或是公式，然後選取**顯示使用方法**。
2. 複查**使用方法**視窗中的資訊，然後按一下**確定**。

在流程圖中使用元件

您可以在流程圖中的元件上執行動作。

另請參閱：

- 關於在流程圖中使用元件
- 在流程圖中收合和展開元件
- 從流程圖中移除元件
- 在流程圖中複製和貼上元件
- 在流程圖中複製和貼上對元件的參照
- 在流程圖中複製和貼上元件群組

關於在流程圖中使用元件

您可以在流程圖中對元件執行動作，包括展開和收合元件以顯示或隱藏詳細資料、移除元件、將元件另存為範本、複製並貼上元件，以及複製並貼上元件群組和元件參照。您還可以還原許多對「範本設計工具」和「規則設計工具」流程圖中使用之元件所進行的變更。

在流程圖中收合和展開元件

如果商業規則或範本中具有許多複雜元件，那麼您可能想要在流程圖中收合或展開其中的一些元件。透過在流程圖中收合和展開元件，您可以將需要使用的元件的顯示空間達到最大，同時將不需要使用的元件的顯示空間達到最小。

若要在流程圖中收合元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下規則或範本，然後選取**開啟**。
2. 在**規則設計工具**或**範本設計工具**流程圖中，在元件上執行下列其中一個動作：
   - 若要展開元件，請用滑鼠右鍵按一下該元件，然後選取**展開**。
   - 若要收合元件，請用滑鼠右鍵按一下該元件，然後選取**收合**。

3. 按一下 。

從流程圖中移除元件

從商業規則或範本的流程圖中移除條件、成員範圍、資料範圍或固定迴圈元件之後將會刪除該元件。這些元件不能共用，元件只會存在於一個商業規則或範本中。

從流程圖中移除公式元件或指令碼元件可能會刪除該元件，也可能不會。如果公式元件或指令碼元件為*未共用*，則從流程圖中移除元件後將刪除該元件。如果公式元件或指令碼元件為*共用*，則只會從移除了元件的商業規則或範本中刪除該元件。

若要從流程圖中移除元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下商業規則或是範本，然後選取**開啟**。
2. 在**規則設計工具**或是**範本設計工具**的流程圖中，用滑鼠右鍵按一下要移除的元件，然後選取**移除**。
3. 按一下 。

在流程圖中複製和貼上元件

當在商業規則或範本的流程圖中複製元件時，您可以將其貼上到位於同一商業規則或範本的流程圖中的不同位置，也可以將其貼上到屬於同一應用程式類型的不同商業規則或範本的流程圖中。

若要複製和貼上元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下規則或範本，然後選取**開啟**。
2. 在**規則設計工具**或**範本設計工具**的流程圖中，用滑鼠右鍵按一下要複製的元件，然後選取**複製**。

   > **提示：**
   >
   > 您還可以使用「編輯」，「複製」。

3. 執行下列其中一項任務：
   - 如果要將元件貼到*相同*商業規則或範本中，請用滑鼠右鍵按一下流程圖中要貼上該元件的位置，然後選取**貼上**。
   - 若要將元件貼到*另一個*商業規則或範本，請開啟商業規則或範本，用滑鼠右鍵按一下流程圖中要貼上該元件群組的位置，然後選取**貼上**。
4. 按一下 。

在流程圖中複製和貼上對元件的參照

與複製元件本身不同，複製對元件的參照僅複製指向該元件的指標。在複製對元件的參照後，該元件本身僅存在於建立它的原始位置。

當您將參照複製到元件時，您可以將它貼到相同的商業規則或範本中，或是貼到屬於相同應用程式類型的不同商業規則或範本中。

若要在流程圖中複製和貼上對元件的參照，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下規則或範本，然後選取**開啟**。
2. 在**規則設計工具**或**範本設計工具**中，用滑鼠右鍵按一下要從流程圖中複製其參照的元件，然後選取**複製參照**。

> **提示：**
>
> 您還可以使用「編輯」，「複製參照」。

> **備註：**
>
> 複製參照僅適用於共用元件 (即公式和指令碼) 或用於另一個規則或範本之流程中的規則和範本。

3. 執行下列其中一項任務：
   - 如果要將元件參照貼到*相同*商業規則或範本中，請用滑鼠右鍵按一下流程圖中要貼上該參照的位置，然後選取**貼上**。
   - 若要將元件參照貼到*另一個*商業規則或範本，請開啟商業規則或範本，用滑鼠右鍵按一下流程圖中要貼上該參照的位置，然後選取**貼上**。
4. 按一下 。

在流程圖中複製和貼上元件群組

如果某個元件包含其他元件 (也就是如果在元件中含有元件)，則可以複製元件群組並將其貼到同一流程圖中的其他位置，或者將其貼到其他流程圖中。

若要在流程圖中複製和貼上元件群組，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下規則或範本，然後選取**開啟**。
2. 在**規則設計工具**或**範本設計工具**中，用滑鼠右鍵按一下要從流程圖中複製的元件群組，然後選取**複製群組**。

> **提示：**
>
> 您還可以使用「編輯」，「複製群組」。

3. 執行下列其中一項任務：
   - 如果要將元件群組貼到*相同*的商業規則或範本，請用滑鼠右鍵按一下流程圖中要貼上該群組的位置，然後選取**貼上**。
   - 如果要將元件群組貼到*另一個*商業規則或範本中，請開啟該商業規則或範本，用滑鼠右鍵按一下流程圖中要貼上該群組的位置，然後選取**貼上**。
4. 按一下 。

8

使用聚總儲存元件設計商業規則

另請參閱：

- 關於使用聚總儲存元件設計商業規則
  使用聚總儲存元件設計商業規則。
- 使用檢視點元件
  您可以建立並編輯檢視點元件。
- 使用配置元件
  您可以建立並編輯配置元件。
- 開啟檢視點或配置元件
  您可從檢視點或配置元件所屬的商業規則流程圖中開啟該元件，但無法從「系統檢視」中開啟。
- 刪除檢視點或配置元件
  若要刪除檢視點或配置元件，請從該元件所屬的商業規則中將其移除。
- 複製和貼上檢視點或配置元件
  您可以從商業規則複製檢視點或配置元件，然後可以將元件貼到相同或不同的商業規則。
- 儲存檢視點或配置元件
  當您在「規則設計工具」中儲存檢視點或配置元件所屬的商業規則時，將會同時儲存該檢視點和配置元件。
- 使用聚總儲存公式元件
  建立、開啟、編輯、刪除、複製、貼上及顯示聚總儲存公式元件的使用情況。

關於使用聚總儲存元件設計商業規則

使用聚總儲存元件設計商業規則。

規劃聚總儲存應用程式中的商業規則由規劃區塊儲存應用程式中商業規則以外的各種元件組成。

您使用這三個元件在 Planning 聚總儲存應用程式中設計商業規則

- 「檢視點」元件包含中繼資料成員清單 (例如科目清單)。

> **備註：**
>
> 您可以將某個「檢視點」在另一個「檢視點」元件中相互套疊構成巢狀架構。

- 配置元件包含許多計算，這些計算能將資料庫大綱中某一層級的成員資料分配給大綱中的其他成員。
- 公式元件包含您使用成員、函式及變數所設計出的計算陳述式。

當您建立元件時，您可能希望保留您正使用開啟的商業規則、元件、範本以及變數。Calculation Manager 會在頁籤式介面中顯示這些物件，以便您可以在建立元件時，輕鬆地在這些頁籤之間移動。您可以在 Calculation Manager 中開啟多達十個頁籤，但是 Oracle 建議您不要同時開啟超過十個以上的物件，以達到最佳效能。

使用檢視點元件

您可以建立並編輯檢視點元件。

另請參閱：

- 建立檢視點元件
- 編輯檢視點元件

建立檢視點元件

您可以從商業規則建立「檢視點」元件。

若要建立「檢視點」元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下規則，然後選取**開啟**。

   「規則設計工具」中會顯示商業設計。

2. 在商業規則流程圖中確定要建立「檢視點」元件的位置後，將**檢視點**元件從**新物件調色盤**中拖放到流程圖中的該位置。

   「檢視點」物件會顯示為內含箭頭的兩個圓。

3. 在**檢視點**上，輸入標題以識別「檢視點」元件。此標題會顯示於任何使用它之規則的流程圖中的元件上方。

4. **選擇性：**請執行下列其中一項任務，以定義「檢視點」的全域範圍：

> **備註：**
>
> 如果已針對您建立「檢視點」元件之商業規則定義全域範圍，「檢視點」頁籤預設會顯示商業規則的成員選擇。若要檢視是否已針對該商業規則定義全域範圍，請選取流程圖中的「開始」或「結束」頁籤。接著按一下「全域範圍」頁籤以查看是否已定義任何成員、函式或變數。

- 按一下**變數選取器**可選取或建立用於定義檢視點的變數。若選取變數，您可選取**動態連結變數**，確保在對變數進行變更時能以動態方式同時更新變數。
- 按一下**成員選取器**可選取用於定義檢視點的成員。
- 在**值**欄中按一下維度的列，可輸入用於定義檢視點的成員名稱。
- 按一下維度的列，接著按一下**動作**圖示，然後選取下列其中一個選項以輸入成員：
  - 成員 (請參閱將成員和函式新增至元件)
  - 變數 (請參閱使用變數)

    若為非 Groovy 規則，*聚總儲存選項* (ASO) 類型的 Planning 立方體只支援 *member* 或 *members* 類型的變數。
  - 函式 (請參閱使用函式)

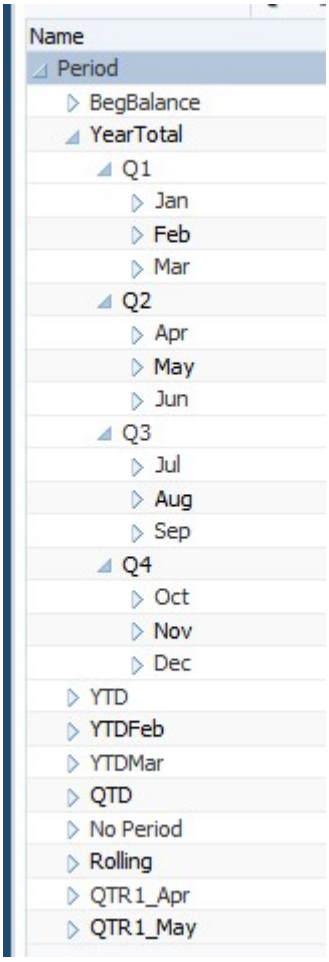
    您輸入的函式應該只會傳回層級 0 成員，且應在函式名稱前加上一個 @ 符號。對於任何具有 `List` 參數的函式，輸入必須是單一成員，或是傳回成員或成員清單的函式。

您可輸入下列函式：

* @ANCESTOR(*Member Name*,*Index*|*Layer*) 會在您給予輸入成員時，傳回位於指定索引或層的祖代。(請只在規則的 POV 中使用 @ANCESTOR。)
* @ATTRIBUTE(*Attribute Member Name*) 會產生與指定的屬性成員相關聯之所有基礎成員的清單。
* @COUSIN(*Member Name*,*Cousin Member Name*) 會傳回與另一個祖代的成員位於相同位置的子成員。(請只在規則的 POV 中使用 @COUSIN。)
* @DISTINCT(*List1*) 會刪除某個集合中重複的 Tuple。(請只在規則的 POV 中使用 @DISTINCT。)
* @EXCEPT(*List1*,*List2*) 會傳回包含兩個集合之間差異的子集，而您可以選擇保留重複項目。(請只在規則的 POV 中使用 @EXCEPT。)
* @FilterDynamic(*Dimension Name*,*Member Name*) 會移除成員清單中的所有動態成員。(請只在規則的 POV 中使用 @FilterDynamic。)
* @FilterShared(*Dimension Name*,*Member Name*) 會移除成員清單中的所有共用成員。(請只在規則的 POV 中使用 @FilterShared。)
* @FilterSharedAndDynamic(*Dimension Name*,*Member Name*) 會移除成員清單中的所有動態成員和共用成員。(請只在規則的 POV 中使用 @ANCESTOR。)
* @FIRSTCHILD(*Member Name*) 會傳回輸入成員的第一個子項。(請只在規則的 POV 中使用 @FIRSTCHILD。)
* @FIRSTSIBLING(*Member Name*) 會傳回輸入成員之父項的第一個子項。請只在全域 POV 中使用 @FIRSTSIBLING。
* @GEN(*Member Name*) 會傳回由指定成員的輸入層代號碼所指定的成員。(請只在規則的 POV 中使用 @GEN。)
* @GENMBRS(*Member Name*,*Generation*) 會傳回由輸入層代號碼所指定的層代。(請只在規則的 POV 中使用 @GENMBRS。)
* @ILSIBLINGS 傳回指定的成員及其左側同層級。(請只在規則的 POV 中使用 @ILSIBLINGS。)
* @INTERSECT(*List1*,*List2*) 會兩個輸入集之間的交集，而您可以選擇保留重複項目。(請只在規則的 POV 中使用 @INTERSECT。)
* @IRSIBLINGS 傳回指定的成員及其右側同層級。(請只在規則的 POV 中使用 @IRSIBLINGS。)
* @LAGGEN(*Member Name*,*Index*) 會使用資料庫大綱中成員的順序，傳回落後指定成員 n 個步驟，且屬於相同層代的成員。請只在全域 POV 中使用 @LAGGEN。
* @LAGLEV(*Member Name*,*Index*) 會使用資料庫大綱中成員的順序，傳回落後指定成員 n 個步驟，且屬於相同層級的成員。請只在全域 POV 中使用 @LAGLEV。
* @LASTCHILD(*Member Name*) 會傳回輸入成員的最後一個子項。(請只在規則的 POV 中使用 @LASTCHILD。)
* @LASTSIBLING(*Member Name*) 會傳回輸入成員之父項的最後一個子項。請只在全域 POV 中使用 @LASTSIBLING。
* @LAYERGEN(*Member Name*,*Index*) 會傳回指定成員的層代式層。

`@LAYERGEN` 只能用於其參數正在尋找層的函式 (例如 `@ANCESTOR(member,layer,index)`) 中。您無法單獨使用 `@LAYERGEN`。請只在規則的 POV 中使用 `@LAYERGEN`。

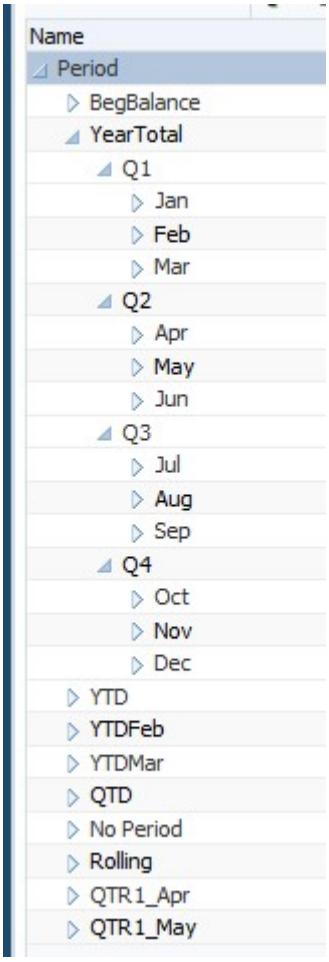
在下列 Period 維度中，`@Level0Descendants(@ANCESTOR(Jul,@LAYERGEN(Period,3)))` 會傳回「Jul, Aug, Sep」



* `@LAYERLEVEL(Member Name,Index)` 會傳回指定成員的層級式層。

  `@LAYERLEVEL` 只能用於其參數正在尋找層的函式 (例如 `@ANCESTOR(member,layer,index)`) 中。您無法單獨使用 `@LAYERLEVEL`。請只在規則的 POV 中使用 `@LAYERLEVEL`。

  在下列 Period 維度中，`@ANCESTOR(Oct,@LAYERLEVEL(Period,0))` 會傳回成員「Oct, Nov, Dec」



* `@LEADGEN(Member Name,Index)` 會使用資料庫大綱中成員的順序，傳回領先指定成員 *n* 個步驟，且屬於相同層代的成員。請只在全域 POV 中使用 `@LEADGEN`。
* `@LEADLEV(Member Name,Index)` 會使用資料庫大綱中成員的順序，傳回領先指定成員 n 個步驟，且屬於相同層級的成員。請只在全域 POV 中使用 `@LEADLEV`。
* `@Level0Descendant(Member Name)` 會擴充以包含指定成員的所有層級 0 子代。
* `@LIST(Argument)` 會傳回成員清單。(請只在規則的 POV 中使用 `@LIST`。)
* `@LSIBLINGS` 傳回指定成員的左側同層級。(請只在規則的 POV 中使用 `@LSIBLINGS`。)
* `@NEXTLEVMBR(Member Name)` 會使用資料庫大綱中成員的順序，傳回相同層級中的下一個成員。請只在全域 POV 中使用 `@NEXTLEVMBR`。
* `@NEXTMBR(Member Name)` 會使用資料庫大綱中成員的順序，傳回相同層代或層級中的下一個成員。(請只在規則的 POV 中使用 `@NEXTMBR`。)
* `@PARENT(Member Name)` 會傳回成員的父項。(請只在規則的 POV 中使用 `@PARENT`。)

* `@PREVLEVMBR(Member Name)` 會使用資料庫大綱中成員的順序，傳回相同層級中的上一個成員。請只在全域 POV 中使用 `@PREVLEVMBR`。
* `@PREVMBR(Member Name)` 會使用資料庫大綱中成員的順序，傳回相同層代或層級中的上一個成員。(請只在規則的 POV 中使用 `@PREVMBR`。)
* `@RSIBLINGS` 傳回指定成員的右側同層級。(請只在規則的 POV 中使用 `@RSIBLINGS`。)
* `@Siblings(Member Name)` 會擴充以包含指定成員的所有同層級。
* `@UDA(Dimension Name, User-Defined Attribute String)` 會根據您在 Oracle Essbase 伺服器上定義為使用者定義屬性 (UDA) 的一般屬性來選取成員。

5. 若要針對您為維度選取的成員輸入備註，請按一下**備註**。
6. 按一下**重設方格**可清除對方格所做的任何項目變更。
7. 按一下 。

編輯檢視點元件

您可編輯成員、變數，以及包含「檢視點」元件全域範圍的函式。您也可以編輯在流程圖中顯示上述元件的圖示標題，以及為「檢視點」中每個維度選取之值所輸入的備註。

若要編輯「檢視點」元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下包含檢視點的商業規則，然後選取**開啟**。
2. 在「規則設計工具」中，選取要在流程圖中編輯的「檢視點」元件，以便顯示其特性。您可編輯以下「檢視點」元件的任何特性。(請參閱建立檢視點元件。)
   - 顯示在商業規則流程圖中「檢視點」元件上的標題
   - 用於定義「檢視點」的成員、變數及函式
   - 當對變數進行變更時，是否以動態方式同時更新在「檢視點」元件中使用的所有變數
   - 用於定義「檢視點」全域範圍的維度及成員，是否已為其輸入備註
   - 當「檢視點」元件所屬的商業規則處於已驗證或已啟動狀態時，在該元件中的成員值是否已計算
3. 按一下 。

使用配置元件

您可以建立並編輯配置元件。

另請參閱：

- 建立配置元件
- 編輯配置元件

建立配置元件

配置元件可讓您將來自成員的資料分散至該成員的層級 0 子代。您可以在商業規則中建立配置元件，該元件只存在於該商業規則內，且無法在商業規則間共用。

若要建立配置元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下規則，然後選取**開啟**。

   「規則設計工具」中會顯示商業設計。

2. 在商業規則流程圖中確定要建立配置元件的位置後，從**新物件調色盤**將**配置**元件拖放到流程圖中的該位置。

   > **備註：**
   >
   > 如果在另一個檢視點元件內放置檢視點，則第二個檢視點會繼承第一個 (即上層) 檢視點的成員、變數及函式。

3. 在「分配」精靈的**檢視點**中，針對配置期間不想變更的每個維度，執行下列其中一項任務：
   - 從**使用預先定義選取項目**中選取預先定義的選取項目，以便為列出的維度填入值。
   - 按一下**成員選取器**圖示，為列出的每個維度選取成員和變數。

     > **備註：**
     >
     > 在「成員選取器」中，在精靈的目前步驟中列出的維度可供使用者從「維度」中選取。這樣您便可為精靈的目前步驟中列出的任何維度選取成員和函式。
     >
     > 請確定您選取的所有成員均為有效的層級 0 成員。

   - 選取清單中的維度，然後按一下**動作**以選取成員或變數。請參閱使用成員選擇、變數、函式、智慧型列示，以及 Planning 公式表示式以設計元件。

4. 在「配置精靈」的**來源**中，針對每個列出的維度，透過執行下列其中一項任務，選取要配置其資料的成員。

   > **備註：**
   >
   > 您必須為列出的每個維度選取一個成員。
   >
   > 來源成員可以是非層級 0 的成員。

   - 從**使用預先定義選取項目**中選取預先定義的選取項目，以便為列出的維度填入值。

     > **備註：**
     >
     > 如果預設的選取項沒有為列出的每個維度皆輸入一個值，您必須為所有空白的維度輸入值。

   - 按一下**成員選取器**圖示，以便為列出的每個維度選取成員。
   - 選取清單中的維度，然後按一下**動作**以選取成員或變數。請參閱使用成員選擇、變數、函式、智慧型列示，以及 Planning 公式表示式以設計元件。

> **備註：**
>
> 您無法在「配置」元件的這個步驟中使用函式。

- 在**選擇性**中，如果您要配置特定值，請輸入要配置的量，而非以上選擇。

5. 如果要配置的來源量為零，請從下拉清單中選取下列其中一個選項。
   - 選取下一個集區記錄。
   - 停止處理配置。
6. 在**配置範圍**中，執行下列其中一項動作：
   - 輸入要用於配置之維度的父成員。

     若要輸入父成員，請執行下列其中一項任務：
     - 從**使用預先定義選取項目**中選取預先定義的選取項目，以便為列出的維度填入值。
     - 按一下**成員選取器**圖示，選取維度的父成員以作為配置資料的目標。
     - 輸入父成員，或取清單中的維度，然後按一下**動作**圖示以選取 (主要維度的) 父成員以配置資料。請參閱使用成員選擇、變數、函式、智慧型列示，以及 Planning 公式表示式以設計元件。

       資料被配置給資料庫大綱中父成員之下的層級 0 的成員 (即大綱中層級最低的成員，其下沒有任何成員)。
   - 選取是否要在配置程序之前，清除區域。

     如果選取**是**清除區域，您要在配置執行之前，在**清除區域**欄位中輸入要清除的成員 (請參閱步驟 8)。
7. 在**目標**中，為其餘的維度選取層級 0 的成員以配置資料。執行下列其中一項任務：
   - 從**使用預先定義選取項目**中選取預先定義的選取項目，以便為列出的維度填入值。
   - 按一下**成員選取器**圖示，為列出的每個維度選取成員。
   - 選取清單中的維度，然後按一下**動作**圖示以選取成員或變數。請參閱使用成員選擇、變數、函式、智慧型列示，以及 Planning 公式表示式以設計元件。
8. 在**清除區域**中，輸入在配置程序之前要透過「邏輯清除」清除的層級 0 成員。執行下列其中一項任務：
   - 從**使用預先定義選取項目**中選取預先定義的選取項目，以便為列出的維度填入值。
   - 按一下**成員選取器**圖示，為列出的每個維度選取成員。
   - 選取清單中的維度，然後按一下**動作**圖示以選取成員或變數。請參閱使用成員選擇、變數、函式、智慧型列示，以及 Planning 公式表示式以設計元件。

> **備註：**
>
> 如果沒有看到**清除區域**，表示您沒有在**配置範圍**中選取**是**在配置之前清除資料。

9. 在**偏移**中，執行下列其中一項任務：
   - 從**使用預先定義選取項目**中選取預先定義的選取項目，以便為列出的維度填入值。
   - 按一下**成員選取器**圖示，為列出的每個維度選取成員。

- 選取清單中的維度，然後按一下**動作**圖示以選取成員或變數。請參閱使用成員選擇、變數、函式、智慧型列示，以及 Planning 公式表示式以設計元件。

> **備註：**
>
> 您必須為偏移指定成員；不能保留空白。

10. **選擇性**：在**排除**中，選取要從配置中排除的任何成員。執行下列其中一項任務：
    - 從**使用預先定義選取項目**中選取預先定義的選取項目，以便為列出的維度填入值。
    - 按一下**成員選取器**圖示，為列出的每個維度選取成員。
    - 選取清單中的維度，然後按一下**動作**圖示以選取成員或變數。請參閱使用成員選擇、變數、函式、智慧型列示，以及 Planning 公式表示式以設計元件。
11. 在**基礎**中，執行下列任務：
    a. 選取配置方法，以指定配置資料的方式。
       - 選取**平均配置**，在配置範圍中平均配置資料值。然後在**平均方法的基礎選項**中，指定如果基礎為負數、零、有缺少的值或所有成員均被排除時要完成的動作。
       - 選取**使用動因配置**，以計算要套用至配置範圍中每個成員的百分比。然後在**基礎選項**中，指定如果基礎為負數或等於零時要完成的動作。
    b. 您未指定的任何維度成員是從您之前定義的 POV 所繼承的，但您可以執行下列其中一項任務來置換這些 POV 選擇：
       - 從**使用預先定義選取項目**中選取預先定義的選取項目，以便為列出的維度填入值。
       - 按一下**成員選取器**圖示，以便為列出的每個維度選取成員。
       - 選取清單中的維度，然後按一下**動作**圖示以選取成員或變數。請參閱使用成員選擇、變數、函式、智慧型列示，以及 Planning 公式表示式以設計元件。
    c. 按**下一步**。
12. 在**進位**中，完成下列步驟：
    a. 輸入要用於此配置的小數位數，或按一下**動作**圖示，選取代表此值的成員或變數。
    b. 選取放置進位差異的位置。
       - 選取**定義位置**以指定要在其中設定放置進位差異的一或多個成員。
       - 選取**使用最大值**，將資料值進位為其最大值
       - 選取**使用最小值**，將資料值進位為其最小值。
       - 選取**捨棄進位錯誤**，依原狀使用配置的資料值。
13. 如果在上一步選取**定義位置**，請在**進位成員**中執行下列其中一項任務：
    - 從**使用預先定義選取項目**中選取預先定義的選取項目，以便為列出的維度填入值。
    - 按一下**成員選取器**圖示，以便為列出的每個維度選取成員。
    - 選取清單中的維度，然後按一下**動作**圖示以選取成員或變數。請參閱使用成員選擇、變數、函式、智慧型列示，以及 Planning 公式表示式以設計元件。

> **備註：**
>
> 您在此步驟中選取的成員必須是配置範圍的一部分。

14. 按一下**完成**。

編輯配置元件

若要編輯配置元件，請開啟該元件所屬的商業規則。當「規則設計工具」中顯示該商業規則時，只要在商業規則流程圖中選取配置元件的特性，便可檢視該元件的特性。

若要編輯配置元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下包含配置元件的商業規則，然後選取**開啟**。
2. 在「規則設計工具」中，選取要在流程圖中編輯的配置元件，以便顯示其特性。您可編輯以下任一個配置元件特性。(請參閱建立配置元件。)
   - 您要配置其資料的成員
   - 您要為其配置資料的層級 0 成員
   - 您要配置的資料及其數量
   - 是否將已配置的資料總量寫入偏移成員
   - 您要平均配置資料，或使用動因按不同的數量配置資料
   - 是否要四捨五入已配置的資料；如果要，請選取四捨五入的方式
3. 按一下 。

開啟檢視點或配置元件

您可從檢視點或配置元件所屬的商業規則流程圖中開啟該元件，但無法從「系統檢視」中開啟。

若要開啟檢視點或配置元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下含有此元件的規則，然後選取**開啟**。

   「規則設計工具」中會顯示商業設計。
2. 當規則開啟時，在該規則的流程圖中按一下檢視點元件或連按兩下配置元件，即可開啟該元件。

刪除檢視點或配置元件

若要刪除檢視點或配置元件，請從該元件所屬的商業規則中將其移除。

由於檢視點或配置元件僅能用於一個商業規則中，只要從這些元件所屬的商業規則中將其移除，便可刪除這些元件。

若要刪除檢視點或配置元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下包含檢視點的規則，然後選取**開啟**。
2. 在規則設計工具中，選取要從流程圖中刪除的檢視點或配置元件。
3. 用滑鼠右鍵按一下檢視點或配置元件，然後選取**移除**。

如果配置元件是在檢視點元件內，則移除檢視點元件會同時移除配置元件。

4. 按一下 。

複製和貼上檢視點或配置元件

您可以從商業規則複製檢視點或配置元件，然後可以將元件貼到相同或不同的商業規則。

若要在商業規則流程圖中複製並貼上檢視點或配置元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下包含檢視點或配置元件的規則，然後選取**開啟**。

   「規則設計工具」中會顯示商業設計。

2. 在**規則設計工具**中，用滑鼠右鍵按一下要從商業規則流程圖中複製的檢視點或配置元件，然後選取**複製**，只複製元件，或選取**複製群組**，以複製元件和其中包含的元件。

3. 執行下列其中一項任務：

   - 如果要將元件貼到*相同*商業規則中，請用滑鼠右鍵按一下流程圖中要貼上該元件的位置，然後選取**貼上**。
   - 若要將元件貼到*另一個*商業規則，請開啟商業規則，用滑鼠右鍵按一下流程圖中要貼上該元件群組的位置，然後選取**貼上**。

4. 按一下 。

儲存檢視點或配置元件

當您在「規則設計工具」中儲存檢視點或配置元件所屬的商業規則時，將會同時儲存該檢視點和配置元件。

不同於公式元件，檢視點和配置元件無法在本身建立依據的商業規則之外獨立存在。

若要儲存檢視點或配置元件，請在完成設計之後，按一下 。

使用聚總儲存公式元件

建立、開啟、編輯、刪除、複製、貼上及顯示聚總儲存公式元件的使用情況。

另請參閱：

- 建立聚總儲存公式元件
- 開啟聚總儲存公式元件
- 編輯聚總儲存公式元件
- 刪除聚總儲存公式元件
- 複製和貼上聚總儲存公式元件
- 將聚總儲存公式元件複製至另一個應用程式或資料庫
- 顯示聚總儲存公式元件的使用狀況

建立聚總儲存公式元件

聚總儲存公式元件由公式計算陳述式組成。若要為公式建立計算陳述式，必須輸入或選取成員和變數。在建立公式時，該公式的每個計算陳述式都會列在「元件設計工具」方格的列中。

您可以從「系統檢視」或「規則設計工具」建立公式元件。公式元件會以獨立物件的形式存在於資料庫中，因此可在商業規則之間共用。

若要為聚總儲存應用程式建立公式元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下**公式**，然後選取**新增**。
2. 輸入公式名稱和應用程式類型。
3. 選取**應用程式**。

   應用程式名稱必須是有效的 Planning 應用程式。
4. 選取**資料庫**。

   > **備註：**
   >
   > 如果您用滑鼠右鍵按一下「公式」並選取「新建」以建立新公式，則「新增公式」對話方塊中會自動填入您在「系統檢視」中使用的應用程式類型、應用程式與資料庫。

5. 按一下**確定**。
6. 在「元件設計工具」的**特性**中，完成下列步驟：
   a. **選擇性**：依預設，公式在建立時為共用；您無法選取或清除「共用」核取方塊。

      > **提示：**
      >
      > 若要建立*不*共用的公式，可先開啟商業規則，然後將新公式元件拖放到商業規則的流程圖中。但「共用」核取方塊並未選取。如果決定將公式共用，請選取「共用」核取方塊。

   b. **選擇性：**輸入 50 個字元以內的新名稱以編輯公式名稱。(預設名稱為「新增公式」對話方塊中的名稱。)
   c. **選擇性：**為公式輸入 255 個字元以內的描述。
   d. **選擇性**：為公式輸入標題。標題在**規則設計工具**流程圖中會顯示於公式的下方。

      > **備註：**
      >
      > 如果不輸入標題，流程圖中會顯示該元件的名稱。

   e. **選擇性**：為公式輸入備註。例如，您可能想要告訴使用者應如何使用公式元件。
7. 在**公式**上，輸入公式的標題。

8. **選擇性：**如果要從**偏移成員**將公式元件中所有公式的*總計*數量寫入偏移成員或跨維度成員，請按一下**省略符號**圖示以選取成員。

> **備註：**
>
> 在您建立的公式下方，可手動定義偏移成員。在公式元件中定義的偏移，是以所有已計算數量的總和計算之。

9. 若要建立公式陳述式，按一下等號*左側*的第一個公式陳述式列。然後，再輸入成員或跨維度成員選擇，或按一下**動作**以選取：
   - 變數 (請參閱使用變數。)
   - 成員 (請參閱將成員和函式新增至元件。)

> **備註：**
>
> 若要輸入成員的跨維度選取項目，請輸入每個成員名稱，並以向右箭頭分隔。例如，mem1->mem2->mem3。

10. 若要完成公式陳述式，請按一下等號*右側*的列。然後，再輸入成員或跨維度成員選擇，或按一下**動作**以選取：
    - 變數 (請參閱使用變數。)
    - 成員 (請參閱將成員和函式新增至元件。)
11. **選擇性**：針對每個公式陳述式列按一下**備註**圖示，輸入關於公式陳述式的備註。按一下**確定**。
12. 在**使用情況**中，您可以檢視使用公式元件的規則。

> **備註：**
>
> 此頁籤上的所有資訊都無法編輯。

13. 按一下 。

開啟聚總儲存公式元件

您可以從「系統檢視」或者從使用公式元件之商業規則的「規則設計工具」流程圖開啟聚總儲存公式元件。

若要開啟聚總儲存公式元件，在**系統檢視**中，展開**公式**，然後執行下列其中一項任務：

- 用滑鼠右鍵按一下要開啟的公式，然後選取**開啟**。
- 按兩下要開啟的公式。

  公式元件會在「元件設計工具」中開啟。

> **備註：**
>
> 若要開啟商業規則內的公式元件，請用滑鼠右鍵按一下公式元件並選取「開啟」，或連按兩下該元件，便可從規則流程圖開啟該公式元件。

編輯聚總儲存公式元件

您可以編輯包括聚總儲存公式元件以及公式元件的備註、標題、名稱及描述的公式陳述式。

若要編輯聚總儲存公式元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下公式元件，然後選取**開啟**。
2. 在「元件設計工具」中，您可編輯以下公式元件的任何特性。請參閱建立聚總儲存公式元件。
   - 標題
   - 公式陳述式
   - 名稱
   - 描述
   - 註解
3. 按一下 。

刪除聚總儲存公式元件

只有在沒有任何商業規則使用聚總儲存公式元件的情況下，您才能夠刪除聚總儲存公式元件。若要查看是否有任何商業規則正在使用公式元件，您可以顯示公式元件的使用狀況。請參閱顯示聚總儲存公式元件的使用狀況。

如果某個商業規則正在使用該公式元件，且您已不再需要在該規則中使用該公式元件，請從規則中將該公式元件移除，然後將其刪除。如果某個商業規則正在使用該公式元件，且您不再需要此商業規則，則您可刪除該商業規則。

如果沒有商業規則使用該公式元件，您可以刪除此元件。

若要刪除聚總儲存公式元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下公式，然後選取**顯示使用情況**。
2. 確保沒有商業規則使用該公式元件。

   請參閱顯示聚總儲存公式元件的使用狀況。
3. 從正在使用該公式元件的任何商業規則中將此公式元件移除。

   請參閱從流程圖中移除元件。
4. 再次在公式上按一下右鍵，然後選取**刪除**。
5. 確認刪除該公式。

複製和貼上聚總儲存公式元件

您可以從規則複製聚總儲存公式元件，然後貼到相同或不同的規則中。也可以複製公式元件中方格的內容，然後將內容貼到同一個或另一個公式元件中。您無法將公式元件複製並貼到其他公式元件或其他元件類型中。

若要複製和貼上聚總儲存公式元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下包含公式元件的規則，然後選取**開啟**。
2. 在「規則設計工具」流程圖中，用滑鼠右鍵按一下要複製的公式元件，然後選取**複製**。

> **備註：**
>
> 如果您要複製的元件是共用元件，則可以依次選取「編輯」、「複製參照」以複製對共用元件的參照，而不必複製元件本身。(請參閱複製並貼上對商業規則公式或指令碼元件的參照。)

3. 執行下列其中一項任務：
   - 若要將公式元件貼至*相同*的商業規則流程圖，請用滑鼠右鍵按一下流程圖的位置，然後選取**貼上**。
   - 若要將公式元件貼至*另一個*的商業規則流程圖，請開啟商業規則，用滑鼠右鍵按一下商業規則中要貼上該元件的位置，然後選取**貼上**。
4. 按一下 。

將聚總儲存公式元件複製至另一個應用程式或資料庫

您可以從某個應用程式複製聚總儲存公式元件到另一個應用程式和資料庫，或者從某個資料庫複製到相同應用程式的另一個資料庫。

若要複製聚總儲存公式元件，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下要複製的公式元件，然後選取**複製到**。
2. 在**另存新檔**中，執行下列其中一項任務：
   - 若要將公式元件複製到其他應用程式，請在**應用程式**中輸入名稱。
   - 若要將公式元件複製到其他應用程式或資料庫，請在**應用程式**和**資料庫**中輸入名稱。
   - 若要將公式元件複製到相同應用程式中的其他資料庫中，請在**資料庫**中輸入名稱。
3. 按一下**確定**。

   公式元件會顯示在您要將其複製到應用程式和資料庫的「公式」節點中。

> **備註：**
>
> 您可能需要重新整理您要將公式元件複製到的應用程式或資料庫節點。請用滑鼠右鍵按一下應用程式或資料庫節點，然後選取「重新整理」。

顯示聚總儲存公式元件的使用狀況

您可以透過從系統檢視顯示公式元件的使用狀況，來查看哪些商業規則正在使用聚總儲存公式元件以及與商業規則相關的其他資訊。

若要顯示聚總儲存公式元件的使用狀況，請執行下列動作：

1. 在**系統檢視**中，用滑鼠右鍵按一下要查看其使用情況的公式，然後選取**顯示使用方法**。
2. 您可檢視以下的公式元件資訊：
   - 正在使用該公式元件之商業規則的名稱
   - 正在使用該公式元件之商業規則的應用程式名稱
   - 正在使用該公式元件之商業規則的資料庫名稱
   - 公式元件的擁有者
   - 正在使用該公式元件之商業規則是否已部署
   - 正在使用該公式元件之商業規則是否已驗證
   - 正在使用公式元件之商業規則的描述

> **備註：**
>
> 此外，您也可以從「使用狀況」頁籤上的「元件設計工具」中檢視公式元件的使用狀況。

9

使用成員選擇、變數、函式、智慧型列示，以及 Planning 公式表示式以設計元件

另請參閱：

- 關於成員選擇、變數、函式與智慧型列示
  與使用元件設計商業規則一樣，您可以使用成員、變數和函式設計元件。
- 將成員和函式新增至元件
  將成員新增到公式元件、指令碼元件、條件元件，以及成員和資料範圍元件。
- 搜尋成員
- 使用變數
  設計商業規則和範本時，在元件中使用變數。
- 使用函式
- 使用自訂函式
  使用自訂函式執行任務，例如複製和匯出資料、移除及新增文字字串的單引號或雙引號、比較兩個文字字串，以及將日期轉換為其他格式。
- 使用智慧型列示
- 使用 Planning 公式表示式
  在 Calculation Manager 圖形化或指令碼規則中使用規劃公式表示式。
- 搭配 Essbase 中的「混合彙總」使用
  區塊儲存資料庫的混合彙總意味著在可能的情況下，區塊儲存資料計算執行的效率類似於區塊儲存資料庫的計算執行效率。

關於成員選擇、變數、函式與智慧型列示

與使用元件設計商業規則一樣，您可以使用成員、變數和函式設計元件。

您在公式、指令碼、條件、成員與資料範圍，以及成員區塊元件中使用成員選擇，以選取成員及可傳回成員清單的函式。請參閱將成員和函式新增至元件。

您可以使用變數為公式、迴圈和條件元件建置公式。變數的類型有兩種：執行變數 (在啟動商業規則時執行計算) 和取代變數 (替代更複雜的公式或函式)。(請參閱使用變數。)

您可以建立以下類型的變數：

- 全域變數，用於屬於某個應用程式類型的所有應用程式
- 應用程式變數，僅用於一個應用程式

  計畫類型或資料庫變數，用於一個計畫類型或資料庫中
- 規則變數，僅用於一個商業規則

函式是可以在迴圈、條件和公式元件中使用的預先定義公式。(請參閱使用函式來取得 Planning 應用程式可用函式類型的描述。)您可以使用函式執行以下類型的計算：

- 將日期字串轉換為數字

- 計算範圍內成員的平均值
- 計算資產在某個期間內的折舊
- 在時間維度中計算成員期間累計的值

您要使用「智慧型列示」來選取預先定義的選項，而不是在規劃的資料表單儲存格中輸入選項。

將成員和函式新增至元件

將成員新增到公式元件、指令碼元件、條件元件，以及成員和資料範圍元件。

另請參閱：

- 關於向元件新增成員和函式
- 將一或多個維度中的成員或函式新增到元件中
- 從元件中移除成員和函式

關於新增成員和函式到元件

您可以將成員新增到公式元件、指令碼元件、條件元件以及成員和資料範圍元件。也可以將傳回成員清單的函式新增到公式、指令碼和條件元件。可以從元件所屬的應用程式的維度中選取成員和函式。

根據使用的元件，可以從一個維度中選取一個或多個成員或函式，也可以從多個維度中選取一個或多個成員或函式。

- 下面是您可以從中為多個維度選取單一成員的環境：
  - 在公式元件的公式方格中
  - 在必要參數為單一成員的函式中
- 當您定義其類型為成員的變數值時，您可以為多個維度選取多個成員。
- 下面是您可以從中為單一維度選取多個成員的環境：
  - 在商業規則的全域範圍中
  - 在設計時期提示類型為資料交集的範本中
  - 在定義其類型為成員的變數值時
  - 在定義其類型為成員的變數限制時
  - 在迴圈元件中
  - 在參數為成員的所有函式中
- 下面是您可以從中為單一維度選取單一成員的環境：
  - 在定義其類型為成員的變數值時
  - 在參數為成員的所有函式中

將一或多個維度中的成員或函式新增到元件中

使用**成員選取器**將一或多個維度的成員或函式新增到元件中。

若要存取**成員選取器**並新增成員或函式，請執行下列動作：

1. 在**系統檢視**中，在範本或商業規則上按一下右鍵，然後選取**開啟**。

2. 當商業規則或範本開啟時，在其流程圖中選取元件。

> **備註：**
>
> 您無法將成員新增到固定迴圈元件中。

3. 按一下 (**成員選取器**圖示)。

成員選取器可讓您選取維度中的成員。請使用 [+] 和 [-] 展開和收合維度中的成員。

成員選取器包含下列頁籤：

- 成員
- 函式 (如果已為維度定義任何函式)
- 搜尋

所選維度中的所有成員與函式都在**成員**與**函式**頁籤上。使用**搜尋**頁籤來搜尋成員或成員描述。

您選取的成員與函式會列於右邊的**選項**下方。當您位於可讓您選取多個成員的元件中時，可以使用 **Shift + 按一下**和 **Ctrl + 按一下**來選取連續或非連續的多個成員。

成員

在**成員**頁籤上，選取一或多個成員，然後按一下 (選取)，將其移至**選項**清單。您還可以使用此表中的選項進一步定義選取內容。

Table 9-1　成員選取器按鈕

按鈕	描述
(新增特殊選項)	**(僅限 Planning 與 Oracle Essbase 區塊儲存應用程式使用者)** 選取下列其中一項，以新增與您在頁籤上選取之成員或函式相關的其他成員或函式： · 成員 · 子項 · iChildren · 子代 · iDescendents · 同層級 · iSiblings · 父項 · iParent · 祖代 · iAncestors · 相對 · 層級 0 (基礎) · 包含 **備註：**Planning 在其「新增特殊選項」選擇中沒有「層級 0 (基礎)」或「包含」。
(選取)	選取該按鈕可將成員或函式移到「選取內容」清單。
(取消選取)	選取該按鈕可從「選取內容」清單中移除成員或函式。

Table 9-1 (Cont.) 成員選取器按鈕

按鈕	描述
(取消全選)	選取該按鈕可從「選取內容」清單中移除所有成員和函式。

函式

在**函式**頁籤上，選取一或多個函式，然後根據下表輸入所需的函式值：

Table 9-2 函式與值

函式	要輸入的值	描述
@ALLANCESTORS	成員名稱	輸入成員名稱或按一下「成員」選取一個成員。
@ANCEST	· 維度名稱 · 層代層級編號 · 成員名稱	1. 輸入在「維度」中選取的維度名稱。 2. 輸入一個整數值，該值用於定義傳回祖代值的層代或層級號碼。正整數定義層代號碼。0 或負整數定義層級號碼。 3. 輸入任意成員名稱或成員組合。
@ANCESTORS	· 成員名稱 · 層代層級編號 · 層代層級名稱	1. 輸入成員名稱或成員組合。 2. 輸入一個整數值，該值用於定義要包含的成員的最大絕對層代或層級號碼。正整數定義層代號碼。0 或負整數定義層級號碼。 3. 輸入要在選取內容中包含的最高的層級名稱或層代名稱。
@ATTRIBUTE	屬性成員名稱	對於輸入的維度，輸入希望在選取內容中包括的屬性成員名稱或成員組合。
@CHILDREN	成員名稱	輸入成員名稱或成員組合，或輸入傳回成員或成員組合的函式。
@CURRMBR	維度名稱	輸入維度名稱。
@DESCENDANTS	· 成員名稱 · 層代層級編號 · 層代層級名稱	1. 輸入成員名稱或成員組合。 2. 輸入一個整數值，該值用於定義要包含的成員的最大絕對層代或層級號碼。正整數定義層代號碼。0 或負整數定義層級號碼。 3. 輸入要在選取內容中包含的最高的層級名稱或層代名稱。
@GENMBRS	· 維度名稱 · genName	1. 輸入維度名稱。 2. 輸入 dimName 中的層代名稱。正整數定義層代號碼。
@IALLANCESTORS	成員名稱	輸入成員名稱或成員組合，或輸入傳回成員或成員組合的函式。
@IANCESTORS	· 成員名稱 · 層代層級編號	1. 輸入成員名稱或成員組合，或輸入傳回成員或成員組合的函式。 2. 輸入一個整數值，該值用於定義要在選取內容中包含的成員的最大絕對層代或層級號碼。正整數定義層代號碼。0 或負整數定義層級號碼。
@ICHILDREN	成員名稱	輸入成員名稱或成員組合，或輸入傳回成員或成員組合的函式。
@IDESCENDANTS	· 成員名稱 · 層代層級編號	1. 輸入成員名稱或成員組合，或輸入傳回成員或成員組合的函式。 2. 輸入一個整數值，該值用於定義要在選取內容中包含的成員的最大絕對層代或層級號碼。正整數定義層代號碼。0 或負整數定義層級號碼。

Table 9-2 (Cont.) 函式與值

函式	要輸入的值	描述
@ILSIBLINGS	成員名稱	輸入成員名稱或成員組合，或輸入傳回成員或成員組合的函式。
@IRDESCENDANTS	· 成員名稱 · 層代層級編號	1. 輸入成員名稱或成員組合，或輸入傳回成員或成員組合的函式。 2. 輸入一個整數值，該值用於定義要在選取內容中包含的成員的最大絕對層代或層級號碼。正整數定義層代號碼。0 或負整數定義層級號碼。
@IRSIBLINGS	成員名稱	輸入成員名稱或成員組合，或輸入傳回成員或成員組合的函式。
@ISIBLINGS	成員名稱	輸入成員名稱或成員組合，或輸入傳回成員或成員組合的函式。
@LEVMBRS	· 維度名稱 · 層級名稱	1. 輸入維度名稱。 2. 輸入用於定義層級號碼的層級名稱或整數值。整數值必須為 0 或正整數。
@LIST	引數	輸入將作為一個引數收集和處理的引數清單，以便這些引數由父函式進行處理。引數可以是成員名稱、成員組合、成員集函式、範圍函式和數值表示式。
@LSIBLINGS	成員名稱	輸入成員名稱或成員組合，或輸入傳回成員或成員組合的函式。
@MATCH	· 成員、層代 · genName · 需要比對的模式	1. 輸入選取內容所基於的成員的預設名稱或使用者定義的名稱。系統將搜尋指定成員及其子代的成員名稱和別名。 2. 輸入選取內容所基於的層代的預設名稱或使用者定義的名稱。系統將搜尋層代中的所有成員名稱及成員別名。 3. 輸入要搜尋的字元模式，包括萬用字元 (* 或 ?)。? 用於替代任何字元。您可在圖樣中任何一處使用 ?。而 * 能代替任意數目的字元。只能在模式的結尾使用 *。要在字元模式中包含空格，請用雙引號 ("") 將該模式括住。
@MEMBER	字串	輸入字串 (用雙引號括住)，或輸入傳回字串的函式。
@MERGE	· 清單 1 · 清單 2	1. 輸入要合併的第一個成員清單。 2. 輸入要合併的第二個成員清單。
@PARENT	· 維度名稱 · 成員名稱	1. 輸入維度名稱。 2. 輸入成員名稱或成員組合，或者輸入傳回成員或成員組合函式，以便與傳回的父項結合。
@RANGE	· 成員名稱 · 範圍清單	1. 輸入成員名稱或成員組合，或者輸入傳回成員或成員組合函式，以便與傳回的父項結合。 2. 輸入成員名稱、逗號分隔的成員名稱清單、成員集函式或範圍函式。如果未指定範圍清單，則系統將使用標記為 Time 的維度中的 0 級成員。
@RDESCENDANTS	· 成員名稱 · 層代層級編號	1. 輸入成員名稱或成員組合，或者輸入傳回成員或成員組合函式，以便與傳回的父項結合。 2. 輸入一個整數值，該值用於定義要選取的成員的最小絕對層代或層級號碼。正整數定義層代號碼。0 或負整數定義層級號碼。

Table 9-2 (Cont.) 函式與值

函式	要輸入的值	描述
@RELATIVE	· 成員名稱 · 層代層級編號	1. 輸入成員名稱或成員組合，或者輸入傳回成員或成員組合函式，以便與傳回的父項結合。 2. 輸入一個整數值，該值用於定義要選取的成員的最小絕對層代或層級號碼。正整數定義層代號碼。0 或負整數定義層級號碼。
@REMOVE	· 清單 1 · 清單 2	1. 輸入要合併的第一個成員清單。 2. 輸入要合併的第二個成員清單。
@RSIBLINGS	成員名稱	輸入成員名稱或成員組合，或輸入傳回成員或成員組合的函式。
@SHARE	範圍清單	輸入逗號分隔的成員清單、傳回成員的函式或成員範圍。範圍清單中的所有成員都必須來自同一維度。
@SIBLINGS	成員名稱	輸入成員名稱或成員組合，或輸入傳回成員或成員組合的函式。
@UDA	· 維度名稱 · 使用者定義屬性	1. 輸入與使用者定義的屬性關聯的維度的名稱。 2. 輸入使用者定義的屬性出現在資料庫大綱中時的名稱。
@WITHATTR	· 維度名稱 · 運算子 · 值	1. 輸入屬性維度的名稱。 2. 輸入運算子規範 (用引號 ("") 括住)。 3. 輸入一個值，該值和運算子組合，定義了必須符合的條件。該值可以是屬性成員規範、常量或日期格式的函式 (即 @TODATE)。
@XRANGE	· 成員名稱 1 · 成員名稱 2	1. 輸入成員名稱、成員組合或傳回單一成員的函式。 2. 輸入成員名稱、成員組合或傳回單一成員的函式。如果 mbrName1 是跨維度成員 (例如 Actual->Jan)，則 mbrName2 也必須為跨維度成員，並且維度順序必須與 mbrName1 中使用的順序相符合。

搜尋

在**搜尋**頁籤上，執行下列任務以搜尋一或多個成員：

1. 從**維度**中選取要在其中搜尋成員的維度。
2. 在**尋找**下方，選取要搜尋的成員類型、成員名稱或其描述。
3. 輸入要搜尋的成員名稱或其描述，以搜尋或顯示維度中的所有成員，接受預設萬用字元 ()。
4. 選取**搜尋**可搜尋在欄位中輸入的成員。(請參閱在「成員選取器」中搜尋「成員」 。)
5. 選取**進階搜尋**可存取進階搜尋選項。(請參閱依名稱、別名或特性在成員選取器中搜尋成員。)
6. 選取一或多個成員，按一下**向右鍵**，將其移至**選擇**清單。

從元件中移除成員和函式

您可以從公式、指令碼、條件以及成員和資料範圍元件中移除*成員*。您可以從公式、指令碼及條件元件中移除*函式*。

從元件中移除成員和函式時，不會將它們從資料庫中刪除。若要從共用元件中移除成員和函式，則必須首先將此元件設為不共用。

若要從元件中移除成員或函式，請執行下列動作：

1. 在**系統檢視**中，在範本或商業規則上按一下右鍵，然後選取**開啟**。
2. 開啟商業規則或範本時，在其流程圖中選取含有要移除的成員或函式的元件。
3. 按一下 (**成員選取器**圖示)。
4. 在**成員選取器**的**選項**中，選取一個維度以移除該維度的所有成員，或選取一個成員以只從維度中移除該成員。
5. 使用向左鍵，將成員或函式從**選項**移到**成員**或是**函式**。
6. 按一下**確定**，然後按一下 (儲存)。

搜尋成員

Related Topics

- 在「成員選取器」中搜尋「成員」
- 依名稱、別名或特性在成員選取器中搜尋成員

在「成員選取器」中搜尋「成員」

您可以在「成員選取器」中搜尋成員。

若要存取「成員選取器」並搜尋成員，請執行下列動作：

1. 在**系統檢視**中，在範本或商業規則上按一下右鍵，然後選取**開啟**。
2. 當範本或商業規則開啟時，在其流程圖中選取元件。
3. 按一下 (**成員選取器**按鈕)。
4. 在**成員選取器**中，維度名稱、其別名 (如果有的話) 及計數 (維度中的成員數) 會顯示在**成員**頁籤上。大綱預設會處於收合狀態。
5. 從**搜尋**頁籤中，選取是要依**名稱**還是**別名**進行搜尋，輸入要搜尋的名稱或別名，然後按一下**尋找**按鈕。
6. **選擇性：**按一下 (**進階搜尋**) 以依成員名稱、別名或其特性之一來搜尋成員。請參閱依名稱、別名或特性在成員選取器中搜尋成員。
7. 選取一或多個成員，然後按一下 (**向右鍵**)，將其移至**選項**清單，並按一下**確定**。

依名稱、別名或特性在成員選取器中搜尋成員

您可以使用「成員選取器」中的「進階搜尋」，按成員名稱、別名或其特性之一搜尋成員。

若要按名稱、別名或特性搜尋成員，請執行下列動作：

1. 從**成員選取器**中，選取**搜尋**頁籤，然後按一下**進階搜尋**。
2. 在**尋找成員**中，從**搜尋依據**選取其中一個選項：
   - **名稱**，依其名稱搜尋成員。然後前往步驟 4。

- **別名**，依其別名搜尋成員。然後前往步驟 4。
- **特性**，依其中一個特性搜尋成員。然後前往步驟 3。

3. 如果您已選取**特性**，請輸入或選取**特性名稱**。
4. 輸入名稱、別名或特性的值。
5. 按一下**確定**。

   如果找到別名、名稱或特性，將會顯示在**結果**中。不會顯示大綱，僅會顯示找到的成員。

> **備註：**
>
> 當您依別名搜尋成員時，具有符合搜尋條件之別名的所有成員 (包括具有其他語言表示之別名的成員) 會顯示在「結果」中。不會顯示大綱，僅會顯示找到的成員。但是，「成員選取器」中只會顯示正在使用中的語言的成員別名。

6. **選擇性：**如果有多個成員符合您的搜尋條件，請使用向上箭號和向下箭號或捲軸 (如果有的話) 來上移和下移，以尋找符合搜尋條件的所有成員。

使用變數

設計商業規則和範本時，在元件中使用變數。

另請參閱：

- 關於變數
- 建立變數
- 輸入執行時期提示變數
- 選取變數
- 編輯變數
- 刪除變數
- 重新整理變數
- 複製變數
- 在變數設計工具尋找和取代文字
- 顯示變數的使用方法

關於變數

變數均採用您為其定義的值。您可以在設計商業規則和範本時在元件中使用變數。

您可以下列方式建立變數：

- 按一下 以啟動「變數設計工具」。
- 在「變數選取器」對話方塊可用的任何位置，從規則、指令碼、公式或範本中建立變數。例如：
  - 開啟規則，然後在「成員範圍」元件中拖曳。

- 按一下維度旁的 ，然後選取**變數**。
- 在**選取變數**對話方塊中，按一下**建立**以建立變數。

當您建立變數時，將會在您選擇的層級建立變數：全域、應用程式、規劃類型，或商業規則。如果在每個層級建立了相同的具名變數，則會在規則中使用最低層級的變數。例如，如果您建立名為 vMonth 的全域變數與名為 Month 的計畫類型變數，規則將使用計畫類型中的變數。

變數有兩種類型：

- **執行** — 啟動商業規則時，執行為變數定義的計算。您可以在指令碼元件或固定迴圈元件中使用執行變數。
- **取代**—設計或啟動商業規則時，變數將由計算替代。您可以在任何元件中使用取代變數。

  您可以建立數個類型的執行和取代變數。根據應用程式類型，以及是要建立執行還是取代變數，您可以建立的變數會有所不同。

> **備註：**
>
> 若為非 Groovy 規則，*聚總儲存選項* (ASO) 類型的 Planning 立方體只支援 *member* 或 *members* 類型的變數。

您可以建立變數，用於在使用者啟動商業規則時，提示使用者輸入資訊。這些執行時期提示變數可以提示使用者資訊為成員、文字、日期還是數字。提示可以告知使用者應當使用的資料類型。

例如：

- 選取月份。
- 輸入預計每個季度的客戶拜訪次數。
- 指定您預計下個月收入變化的百分比。

根據您要為哪種應用程式類型建立變數，一個變數最多可以有四種相關聯的資料庫物件。變數可同時存在多個物件中，且在每個物件中可以有相同的名稱。

建立變數

若要建立變數，請執行下列動作：

1. 在**系統檢視**、**自訂檢視**、**篩選檢視**或**部署檢視**中，按一下 。
2. 在**變數導覽器**中，展開應用程式類型。

   例如，展開**規劃**。
3. 選擇要件利變數的層級。
   - **全域** — 在**<全域>**按一下右鍵，然後選取**新增**，以建立可用於相同應用程式類型之任何應用程式中的變數。
   - **應用程式** — 用滑鼠右鍵按一下某個應用程式，然後選取**新增**，建立只能在該應用程式中使用的變數。
   - **計畫或資料庫** — 用滑鼠右鍵按一下計畫類型或資料庫，然後選取**新增**，建立僅可在該計畫類型或資料庫中使用的變數。
   - **商業規則** — 用滑鼠右鍵按一下某個商業規則，然後選取**新增**，建立只能在該規則中使用的變數。

4. 選取要建立的變數類型：
   - **取代** — 設計或啟動商業規則時，變數將由計算替代。您可以在任何元件中使用取代變數。

     若要建立取代變數，請在**取代**頁籤中，依序選取**動作**、**新增**，然後輸入下列資訊：
     - **名稱** — 變數名稱
     - **描述** — 變數的描述
     - **群組** — 如果要將此變數包括在某個群組中，請輸入群組名稱。在儲存並重新整理變數後，該群組名稱將顯示在**群組**欄下。
     - **類型** — 按一下下拉清單，並選取類型。
     - **RTP** — 如果這是執行時期提示變數：
       * 選取 **RTP**，並在每次使用變數時輸入要顯示的文字。
       * 視需要輸入**預設值**。
       * 選取**使用最後輸入的值**，在下次提示時顯示為提示輸入的最後一個值作為預設值。
   - **執行** — 啟動商業規則時，執行為變數定義的計算。您可以在指令碼元件或固定迴圈中使用執行變數。

     若要建立執行變數，請在**執行**頁籤中，依序選取**動作**、**新增**，然後輸入下列資訊：
     - **名稱** — 變數名稱
     - **群組** — 如果要將此變數包括在某個群組中，請輸入群組名稱。在儲存並重新整理變數後，該群組名稱將顯示在**群組**欄下。
     - **值** — 變數值
       * 對於數值變數，請參閱輸入數值變數的變數值
       * 對於字串變數，請參閱輸入字串變數的變數值
       * 對於成員範圍變數，請參閱輸入成員或成員變數的變數值
5. 按一下

輸入數值變數的變數值

數值變數可以是 Planning 取代變數。

若要輸入數值變數的值，請執行下列動作：

1. 從**類型**中，選取**數值**。
2. 若要使用「智慧型列示」，請在應用程式、計畫類型或規則層級建立變數。(您無法在全域層級使用「智慧型列示」。) 接著依序按一下「智慧型列示」方塊、下拉清單，然後選取「智慧型列示」。

   請參閱*管理 Oracle Planning and Budgeting Cloud Service 的規劃*，以取得這一版的相關資訊。例如，可以設定一個整數智慧型列示用於表示值為 1-5 的報告週期：每年 (1)、每季度 (2)、每月 (3)、每天 (4) 和每小時 (5)。當使用者選取「每月」時，系統就會將數字 3 儲存在資料庫中。這樣使用者便無需記住這些數字。

   您還可以設定文字字串或日期作為「智慧型列示」的值。

> **備註：**
>
> 您必須選取支援使用「智慧型列示」的規劃應用程式。

3. 若要使用數值執行時期提示，請將**智慧型列示**方塊保留空白，然後前往下一步。
4. **選擇性：**在**限制**中，選取變數限制。
5. **選擇性：**輸入變數的預設值。
6. 預設選取 **RTP**。如果不希望為此變數建立執行時期提示，請清除 **RTP**。

> **備註：**
>
> 如果不選取 RTP，則必須為該變數輸入一個預設值。

7. 如果您已選取 **RTP**，請輸入要向使用者顯示為預設值的執行時期提示文字。
8. 指定是否允許遺漏資料值。
9. 按一下 。

若要為 Oracle Hyperion Financial Management 取代或執行數值變數輸入值，請執行下列動作：

輸入字串變數的變數值

字串變數可以是 Planning 取代變數。字串變數必須是英數字元，並且不得超過 255 個字元。它可以包含空值，但不能在值中包含前置的 & (和) 字元。

若要輸入取代字串變數的值，請執行下列動作：

1. 從**類型**中，選取**字串**。
2. 在**值**表中，輸入變數的值。
3. 預設選取 **RTP**。如果不希望為此變數建立執行時期提示，請清除 **RTP**。

> **備註：**
>
> 如果不選取 RTP，則必須為該變數輸入一個預設值。

4. 如果您已選取 **RTP**，請輸入您要為使用者顯示的執行時期提示文字。
5. 按一下 。

若要為 Oracle Hyperion Financial Management 取代或執行字串變數輸入值，請執行下列動作：

輸入陣列變數的值

陣列變數可以是 Planning 執行變數。陣列中包含可以是多維的值清單。

通常，陣列用於儲存屬於成員公式的變數。陣列變數的大小由對應維度中的成員數確定。例如，如果 Scenario 維度有四個成員，下列命令將建立一個名為 Discount、具有四個項目的陣列。您可以一次使用多個陣列。

```
ARRAY Discount[Scenario];
```

若要輸入陣列變數的值，請執行下列動作：

1. 選取**陣列**核取方塊，將此執行變數設定為「陣列」。選取**陣列**之後，您必須從下拉清單選取維度，否則它將預設為計畫類型中的一個維度。
2. 在**值**欄位中，輸入可在計畫層級建立變數的值。
3. **選擇性**：輸入變數的群組。
4. 按一下 。

輸入成員範圍變數的值

成員範圍變數可以是 Planning 取代變數。成員範圍變數應該包含一個範圍的成員。

若要輸入成員範圍變數的值，請執行下列動作：

1. 從**類型**中，選取**成員範圍**。
2. 在**變數方格**中，執行下列步驟：
   a. 對於表格中您要選取成員範圍限制的每個維度，按一下限制欄位並輸入限制。(顯示的維度是屬於要為其建立變數的應用程式的維度。)

      在全域層級，如果您選取「維度類型」選項，則只有標準維度類型才會顯示出來。如果您選取「維度名稱」選項，您可以輸入任何維度名稱。

      > **備註：**
      > 在「限制」欄位中輸入文字前，必須先選取 RTP。

      > **備註：**
      > 針對「限制」使用函式時，建議使用 Planning 函式，而不要使用 Oracle Essbase 函式。在某些情況下，Essbase 函式不會傳回預期的成員，例如當評估的 Essbase 函式包括動態成員時。針對成員範圍變數中的「限制」，請使用 "ILvl0Descendants("Mbr Name") "，而不要使用 Essbase 函式 "@Relative("Mbr Name", 0)"。

   b. 輸入或使用「成員選取器」選取成員範圍的預設值，或者，如果變數為執行時期提示，您可以將預設值保留為空白。您可以為列出的每個維度選取多個成員和函式。
   c. 如果希望在啟動變數時提示使用者相關資訊，請為列出的每個維度選取 **RTP**。

      > **備註：**
      > 如果不選取 RTP，則必須為該變數輸入一個值。

d. 對於您選取 **RTP** 的每個維度，輸入執行時期提示文字。

e. 在方格上方的 RTP 文字方塊中，輸入每次針對該維度啟動變數時，要向使用者顯示的執行時期提示文字。

3. 按一下 。

輸入跨維度變數的變數值

跨維度變數是 Planning 取代變數。其中包含多個維度的成員，可讓您跨多個維度啟動商業規則。

若要輸入跨維度變數的值，請執行下列動作：

1. 在**類型**中，選取**跨維度**。
2. **選擇性：**對於您想要選取變數限制的表格中的每一個維度，在限制欄位中按一下，然後輸入限制。(顯示的維度是屬於要為其建立變數的應用程式的維度。)

> **備註：**
>
> - 在「限制」欄位中輸入文字前，必須先選取 RTP。
> - 針對「限制」使用函式時，建議使用 Planning 函式，而不要使用 Oracle Essbase 函式。在某些情況下，Essbase 函式不會傳回預期的成員，例如當評估的 Essbase 函式包括動態成員時。針對跨維度變數中的「限制」，請使用 "ILvl0Descendants("Mbr Name") "，而不要使用 Essbase 函式 "@Relative("Mbr Name", 0)"。
> - 您可以使用函式，但該函式必須從維度傳回單一成員，才會被視為有效的選取項目。

3. 輸入或使用「成員選取器」為變數選取值。您可以選取成員或函式。
4. 輸入執行時期提示文字。
5. 按一下 。

輸入維度變數的變數值

維度變數是 Planning 取代變數。此變數包含您選取的維度。

若要輸入維度變數的值，請執行下列動作：

1. 從**類型**中，選取**維度**。
2. 在**變數方格**中，選取維度。顯示的維度是屬於要為其建立變數的應用程式的維度。
3. 如果選取了 **RTP**，請輸入要向使用者顯示的執行時期提示文字。
4. 按一下 。

輸入成員或成員變數的變數值

成員變數是 Planning 取代變數。這些變數包含了所選維度的一個成員或多個成員。

若要輸入一個成員或多個成員變數的值，請執行下列動作：

1. 從**類型**中，選取**單一成員**或**多個成員**。
2. 在**變數方格**中，選取維度。顯示的維度是屬於您對其建立變數之應用程式的維度。
3. 輸入或使用「成員選取器」選取變數的限制。您可以只從在步驟 2 中選取的維度選取成員。也可以選取函式。請參閱使用函式。

> **備註：**
>
> 針對「限制」使用函式時，建議使用 Planning 函式，而不要使用 Oracle Essbase 函式。在某些情況下，Essbase 函式不會傳回預期的成員，例如當評估的 Essbase 函式包括動態成員時。針對成員或成員變數中的「限制」，請使用 "ILvl0Descendants("Mbr Name") "，而不要使用 Essbase 函式 "@Relative("Mbr Name", 0)"。

4. 輸入或使用「成員選取器」選取變數的預設值。您可以對 *member* 變數選取一個成員或函式，以及對 *members* 變數選取多個成員和函式。
5. 預設選取 **RTP**。如果不希望建立執行時期提示變數，請清除 **RTP**。

> **備註：**
>
> 如果不選取 RTP，則必須為該變數輸入一個值。

6. 如果選取了 **RTP**，請輸入要向使用者顯示的執行時期提示文字。
7. 選取**檔案**，然後選取**儲存**。

輸入百分比變數的變數值

百分比變數是 Planning 取代變數。百分比變數同時也是規劃執行變數。此變數包含您指定的百分比。

若要輸入百分比變數的值，請執行下列動作：

1. 從**類型**中，選取**百分比**。
2. 在**變數方格**中，按一下**限制**，以定義變數的下限與上限值。
3. 為變數輸入一個數值。
4. 依預設，會選取 **RTP**。如果不希望建立執行時期提示，請清除 **RTP**。

> **備註：**
>
> 如果不選取 RTP，則必須為該變數輸入一個值。

5. 如果選取了 **RTP**，請輸入要向使用者顯示的執行時期提示文字。
6. 選取是否允許遺漏資料值。

7. 選取**檔案，儲存**。

輸入整數變數的值

整數變數是一個 Planning 取代變數。

若要輸入整數變數的值，請執行下列動作：

1. 從**類型**中，選取**整數**。
2. **選擇性：**在**值**檔案，按一下**限制**來定義變數的整數值上下限。
3. **選擇性：**按一下預設值，然後對變數輸入一個整數。
4. 預設選取 **RTP**。如果不希望為此變數建立執行時期提示，請清除 **RTP**。

   如果不選取 RTP，則必須為該變數輸入一個預設值。
5. 如果選取了 **RTP**，請輸入要向使用者顯示的執行時期提示文字。
6. 選取是否允許 #Missing 值。
7. 選取**檔案，儲存**。

針對「數字形式的字串」變數輸入變數值

「數字形式的字串」變數可以是 Planning 取代或執行變數。

若要輸入「數字形式的字串」變數的值，請執行下列動作：

1. 從**類型**中，選取**數字形式的字串**。
2. **選擇性：**在**值**表格中，按一下**限制**以定義變數的下限與上限值。下限與上限值應該以 YYYYMMDD 格式的數字輸入。
3. **選擇性：**對變數輸入數值。
4. 預設選取 **RTP**。**RTP** 是 StringAsNumber 變數的必要項目，因此，您不可清除 **RTP** 核取方塊。
5. 選取是否允許 #Missing 值。
6. 選取**使用最後輸入的值**以允許使用者使用最後所輸入的值。
7. 按一下 。

針對「數字形式的日期」變數輸入變數值

「數字形式的日期」變數可以是 Planning 取代或執行變數。

若要輸入「數字形式的日期」變數的值，請執行下列動作：

1. 從**類型**中，選取**數字形式的日期**。
2. **選擇性：**在**值**表格中，按一下**限制**以定義變數的下限與上限值。下限與上限值應該以 `YYYYMMDD` 格式的數字輸入。
3. **選擇性：**對變數輸入數值。對於 dateasnumber 變數，請以 `YYYYMMDD` 的數值格式輸入日期。
4. 預設選取 **RTP**。如果不希望建立執行時期提示，請清除 **RTP**。

如果不選取 RTP，則必須為該變數輸入一個值。

5. 如果選取了 **RTP**，請輸入要向使用者顯示的執行時期提示文字。
6. 選取是否允許 #Missing 值。
7. 選取**使用最後輸入的值**以允許使用者使用最後所輸入的值。
8. 按一下 。

輸入執行時期提示變數

> **備註：**
>
> - 您無法將 RUNTIMESUBVARS 區段新增至 Calculation Manager 中 Oracle Essbase 商業規則的指令碼。只有在您部署 Essbase 應用程式時，在 Calculation Manager 中建立的規則層級執行時期提示變數才會轉換為 RUNTIMESUBVARS。
> - 如果您從 Administration Services、MaxL 或可啟動計算指令碼的任何元件啟動具有執行時期提示的商業規則，該商業規則的執行時期提示必須具有預設值。
> - 當進行驗證、偵錯、部署、分析以及啟動 Calculation Manager 中的規則時，您可以輸入或編輯執行時期提示變數的值。您也可以在驗證或部署商業規則集時輸入或編輯執行時期提示變數。如果執行時期提示包含成員限制，驗證會僅檢查有效的成員名稱 (不會驗證成員是否在限制範圍內)。執行時期提示限制會驗證*數值*和*整數*類型變數。
> - 如需設計遵循成員核准安全性的執行時期提示相關資訊，請參閱 *管理 Planning* 中的關於核准安全性中的執行時期提示。

若要輸入執行時期提示變數值，請執行下列動作：

1. 當您驗證、偵錯、部署、分析、或者啟動商業規則、或者驗證或部署業務規則集時，如果未發生任何錯誤，將會顯示**輸入 RTP 值**對話方塊。

   當您驗證，偵錯，部署或分析商業規則，或者驗證或部署業務規則集時，只有在有一或多個使用中的執行時期提示變數商業規則 (或商業規則集) 遺漏值時，才會顯示「輸入 RTP 值」對話方塊。如果所有執行時期提示變數都有值，則不會顯示「輸入 RTP 值」對話方塊。

   當您啟動商業規則時，無論執行時期提示變數是否有值，每次都會顯示「輸入 RTP 值」對話方塊。如果有值，根據預設，那些值會顯示在「輸入 RTP 值」對話方塊中。
2. 對於每個列出的執行時期提示，請輸入或選取一個值。
3. **選擇性：**如果您使用商業規則，請選取**將值套用至規則**核取方塊，以便您提供的值會在變數值中的動態更新，以及顯示在規則設計工具內**變數**頁籤的**值**欄中。

   如果您從「系統檢視」驗證商業規則，將不提供此核取方塊。
4. 按一下**確定**。
5. 如果發生驗證錯誤，請修正錯誤，然後重複您要輸入執行時期提示變數值的作業。

選取變數

您可以從各個位置選取變數。當您在「元件設計工具」內建立元件、在「範本設計工具」內建立設計時期提示，以及 Calculation Manager 中的其他位置時，您可以選取變數。

若要選取變數，請執行下列動作：

1. 執行下列其中一項任務：
   - 用滑鼠右鍵按一下包含要新增變數的元件的範本，然後選取**開啟**。
   - 用滑鼠右鍵按一下包含要新增變數的元件的商業規則，然後選取**開啟**。
2. 開啟商業規則或範本時，在其流程圖中選取要向其中插入變數的元件。
3. 在流程圖下方的頁籤中，執行下列其中一項任務：
   - 對於成員範圍元件，在維度欄位中按一下，選取**動作**圖示，然後選取**變數**，以選取成員範圍變數。或者是按一下**變數選取器**，以選取成員範圍變數。
   - 對於固定迴圈元件的資料範圍，請在**變數**欄位中，選取**變數**圖示。
   - 對於公式元件，按一下**動作**圖示，然後選取**變數**。
   - 對於指令碼元件，按一下**插入變數**圖示。
   - 對於條件元件，啟動**條件建置器**，按一下**動作**圖示，然後選取**變數**。
4. 在**選取變數**中，執行下列其中一項任務：
   - 若要建立新變數，請按一下**建立**，以存取**變數設計工具**。請參閱建立變數。
   - 若要選取現有變數，請從**類別**選取包含您要使用的變數。您可以選取：
     - 全域：變數是在全域層級中建立，並且可用於此應用程式類型下的所有應用程式。
     - 應用程式：變數是在應用程式層級中建立，並且只會對該應用程式顯示。
     - 計畫類型或資料庫：變數是在計畫類型或資料庫層級中建立，並且僅在計畫類型或資料庫中顯示。
     - 規則：變數是在規則層級中建立，並且僅在其建立的規則中顯示。

> **備註：**
>
> 對於成員範圍、成員區塊、公式、固定迴圈，以及條件元件，顯示的變數限於該元件使用的預期變數類型，因此，依預設不會顯示所選取範圍的所有可用變數。若要查看所選取範圍中的所有可用變數，請選取「顯示所有變數」核取方塊。

5. 在**取代**或**執行**中，選取一或多個變數，以插入元件。
6. 按一下**確定**，以及 。

編輯變數

您可以從「變數設計工具」中編輯變數的任何特性。當您對變數進行變更時，如果該變數已用於規則中，您應該開啟規則、儲存、驗證，然後重新部署。請參閱驗證與部署。

刪除變數

您可以從「變數設計工具」中刪除一個或多個變數，前提是沒有任何元件或成員公式使用它們。如果在元件中使用了某個變數，則必須先從元件中移除該變數，然後才能刪除該變數。

若要刪除變數，請執行下列動作：

1. 在**系統檢視**或**篩選檢視**中，按一下**變數設計工具**圖示。

2. 在「變數導覽器」中，展開應用程式類型與應用程式。
3. 執行下列其中一項任務：
   - 如果變數為全域變數，請選取 **<全域>**。
   - 如果變數為應用程式變數，請選取與該變數關聯的應用程式。
   - 如果變數為計畫類型或資料庫變數，請選取與變數關聯的計畫類型或資料庫。
   - 如果變數為商業規則變數，請選取與變數關聯的商業規則。

   與應用程式類型、應用程式、計算類型、計畫類型或資料庫以及商業規則關聯的任何變數都將顯示在**取代**或**執行**中。
4. 在**取代**或**執行**上，用滑鼠右鍵按一下您要刪除的變數，然後選取**刪除**。
5. 在**刪除確認**中，選取**是**，以確認刪除變數。

重新整理變數

您可以在「變數導航器」中重新整理變數清單，以檢視新增、刪除或變更變數後的最新清單。

若要在「變數導航器」中重新整理變數清單，請執行下列動作：

1. 在**系統檢視**或**篩選檢視**中，按一下**變數設計工具**圖示。
2. 在「變數導航器」中，建立、編輯或刪除變數。
3. 在**取代**或**執行**頁籤上方，按一下**重新整理**圖示。

複製變數

您可以使用複製和貼上，將變數複製至相同或不同的變數範圍 (亦即全域層級、應用程式層級、整合、計畫或資料庫層級、或規則層級)。如果您正在複製的變數在您複製到其中的位置有同名的變數，您可以對正在複製的變數提供新名稱、略過複製變數或覆寫變數的內容。

若要複製並貼上變數，請執行下列動作：

1. 在**系統檢視**或**篩選檢視**中，按一下**變數設計工具**圖示。
2. 在「變數導航器」中，展開 Planning，然後選取**全域**或者應用程式、計畫類型，或者與您要複製之變數關聯的商業規則。
3. 用滑鼠右鍵按一下您要複製的變數，然後選取**複製**。
4. 用滑鼠右鍵按一下您要貼上複製變數的位置或變數，然後選取**貼上**。(例如，如果您正在複製規劃全域變數，可以將其複製成另一個規劃全域變數或計畫類型變數)
   - 如果您嘗試複製變數並貼至具有同名變數的位置，將會顯示**解決衝突**對話方塊。執行下列其中一項任務：
     - 提供變數的新名稱。(同一位置不能有兩個同名的變數。)
     - 指定以略過複製變數。(複製變數的內容不會貼至新位置。)
     - 指定以覆寫變數。(複製變數的內容將貼至新位置並覆寫複製至變數的內容。)
   - 如果您嘗試複製變數並貼上未包含同名變數的位置，變數將貼至新位置。

在變數設計工具尋找和取代文字

您可以在「變數設計工具」中搜尋和取代文字。您可以搜尋任何範圍的變數：(全域、應用程式、計畫或資料庫，或是商業規則)。

您可以在「變數設計工具」的「尋找」功能中輸入變數名稱來搜尋變數。您也可以搜尋變數中的文字字串。(例如，您可以搜尋變數中所使用的預設值。)

根據預設，Calculation Manager 使用變數的預設值、任何為其定義的限制和任何提示文字進行搜尋。如果您在搜尋中包括變數的基本特性，Calculation Manager 會使用變數名稱、群組及描述進行搜尋。

您可以取代變數或文字字串的所有例項，或是取代選取的例項。當您取代文字，而「限制」欄位包含類型智慧型列示或數字的變數時，這些內容並不會受到排除。取代這些欄位可能會導致變數定義的狀態不正確 (例如，變數可能會有無效的智慧型列示名稱，或是不在指定限制之內的預設值。)

若要在「變數設計工具」尋找文字，請執行以下動作：

1. 從任何檢視中，按一下**變數設計工具**圖示。
2. 在**變數導覽器**中，用滑鼠右鍵按一下應用程式類型「全域」(僅限規劃和 Oracle Hyperion Financial Management 使用者)、計畫類型或資料庫，或者您要進行搜尋中商業規則，然後選取**尋找**。
3. 從**任何文字**中，選取：
   - **開始於**，僅顯示名稱以指定字元開頭的變數
   - **結束於**，僅顯示名稱以指定字元結尾的變數
   - **包含**，僅顯示名稱包含指定字元的變數
   - **相符**，僅顯示名稱符合指定字元的變數
4. 在**搜尋**欄位中輸入要搜尋的變數文字。
5. 選取下列其中一或多個選項：
   - 如果您搜尋的文字不需要符合**搜尋**欄位中輸入文字的大小寫，請選取**忽略大小寫**。
   - 如果您要在搜尋中包含變數名稱、描述和群組，請選取**包含基本特性**。

     當您啟動「變數設計工具」時，依預設將忽略大小寫並選取「包含基本特性」。若您清除這些核取方塊，這些核取方塊將維持取消，直到您關閉並重新開啟「變數設計工具」為止。
   - 如果您想要搜尋所選取節點以下層級中的變數，請選取**包含子項範圍中的變數**。若您搜尋位於*應用程式類型*層級中的變數 (例如 Oracle Essbase 或規劃)，此核取方塊依預設選取，且無法進行變更。在此選項被選取的情況下，將顯示應用程式、計畫類型、整合類型或資料庫，以及商業規則的變數。除了變數名稱、描述、預設值、群組和擁有者欄，「應用程式」欄、「計畫類型」欄和「規則」欄也會顯示在變數清單中。

     若您搜尋位於*應用程式*層級中的變數，此核取方塊可以進行選取或清除。當您選取此選項時，將會顯示應用程式、其計畫類型、整合類型或資料庫，以及商業規則的變數。除了變數名稱、描述、預設值、群組和擁有者欄，也會顯示「計畫類型」欄和「規則」欄。

     在規劃中搜尋全域變數時將不提供此選項。當您搜尋全域變數時，將僅顯示全域變數。
6. 按一下**確定**。

   若找到您所搜尋的文字，找到的變數將顯示在「取代」和「執行」頁籤中。

找到文字後，您可以取代一個或多個文字例項。

若要在「變數設計工具」取代文字，請執行以下其中一項任務：

- 若要取代文字字串選取的例項，請執行以下動作：
  1. 選取您要取代文字字串的變數。
  2. 依序選取**動作**、**取代已選取項目**。
  3. 在**取代已選取項目**對話方塊的**取代為**中，輸入想要取代文字字串的文字。
  4. 按一下**取代已選取項目**。
- 若要取代文字字串的所有例項，請執行以下動作：
  1. 依序選取**動作**、**全部取代**。
  2. 在**全部取代**對話方塊的**取代為**中，輸入想要取代文字字串的文字。
  3. 按一下**全部取代**。

> **備註：**
>
> 您在搜尋文字字串時所選取的任何選項，將在「取代已選取項目」和「全部取代」對話方塊的「尋找」區域中預設為選取，且無法進行變更。例如，若您在搜尋文字字串時清除了「忽略大小寫」核取方塊，「取代已選取項目」/「全部取代」對話方塊中的「忽略大小寫」核取方塊將取消，且無法進行變更。

顯示變數的使用方法

您可以顯示使用變數的商業規則。顯示變數的使用方法時，將顯示以下資訊：

- 使用變數之商業規則的名稱
- 使用變數之商業規則的應用程式名稱
- 使用變數之商業規則的計畫類型
- 使用變數之商業規則的擁有者
- 使用變數的商業規則是否已部署
- 使用變數的商業規則是否已驗證
- 使用變數之商業規則的描述

若要顯示變數的使用方法，請執行下列動作：

1. 從「系統」或「篩選」檢視，按一下**變數設計工具**圖示。
2. 在「變數導航器」中，選取包含您要檢視其使用方法之變數的資料庫物件。針對該物件定義的變數會顯示在「變數設計工具」中的**取代**與**執行**頁籤。
3. 用滑鼠右鍵按一下要檢視其使用方法的變數，然後選取**顯示使用方法**。
4. 複查這些資訊後，按一下**確定**。

使用函式

另請參閱：

- 關於函式
  函式可定義傳回資料值或成員的成員公式。
- 將函式插入元件
  您可選取的函式類型，將視您正在使用的應用程式及元件類型而有所不同。
- Calculation Manager 所支援的 Essbase 函式
  以下是「區塊儲存」應用程式中 Calculation Manager 所支援的 Oracle Essbase 函式。

關於函式

函式可定義傳回資料值或成員的成員公式。

例如，可以使用函式 (數學或邏輯運算子) 傳回所指定成員的同層級、父項或子項清單，傳回大於或小於指定值的資料值清單，或從指定成員中指派資料值。選取函式時，將提示您輸入正確的參數。

如果您使用 Planning 應用程式，您可以在公式、指令碼、條件和成員範圍元件中使用函式。

以下是您可以在 Planning 和 Oracle Essbase 區塊儲存元件中使用的函式類型清單。(如需完整的函式清單與描述，請參閱 Oracle Essbase Technical Reference (僅英文版)。)

- 布林值
- 關聯
- 計算運算子
- 控制流程
- 資料宣告
- 函數型
- 數學型
- 成員集合
- 範圍 (財務)
- 配置
- 預測
- 統計型
- 日期和時間
- 雜項
- 自訂

> **Note:**
>
> 可以在「成員選取器」和「函式選取器」中使用函式。
>
> 您只能在 Essbase 聚總儲存應用程式和元件 (包含 Oracle General Ledger 使用的那些應用程式和元件) 中使用「成員集」函式。(如需完整的成員集合函式清單與描述，請參閱 Oracle Essbase Technical Reference (僅英文版)。)

將函式插入元件

您可選取的函式類型，將視您正在使用的應用程式及元件類型而有所不同。

如果您使用 Planning 應用程式，您可以將函式插入到公式、指令碼、條件和成員範圍元件中。

若要將函式插入公式、指令碼、條件或成員範圍元件中，請執行下列動作：

1. 執行下列其中一項任務：
   - 開啟包含要向其插入函式的元件的商業規則。然後，選取在商業規則的流程圖中要為其插入函式的元件。
   - 開啟包含要向其插入函式的元件的範本。然後，選取在商業規則的流程圖中要為其插入函式的元件。
   - 開啟您要向其插入函式的公式元件。
   - 開啟您要向其插入函式的指令碼元件。
2. 執行下列其中一項任務：
   - 若要將函式插入公式元件，請在**公式**頁籤上，按一下**公式**列，選取**動作**圖示，然後選取**函式**。
   - 若要將函式插入指令碼元件，請在**指令碼**頁籤上，按一下**插入函式及其參數**圖示。
   - 若要將函式插入成員範圍元件，請在**成員範圍**頁籤上，按一下維度的**值**欄，選取**動作**圖示，然後選取**函式**。
   - 若要將函式插入條件元件，請在**條件**頁籤上，按一下**條件建置器**圖示。在「條件建置器」中，從**公式**、**函式**或**值**中，選取**動作**圖示，然後選取**函式**。
3. 在**函式選取器**中，執行下列其中一項任務：

   > **備註：**
   >
   > 根據您開啟「函式選取器」所在的環境，可用函式類型可能會比使用函式中描述的函式類型更有限。

   - 如果您可以從**類別**選取函式類型，請選取一個，或選取**所有函式**，以顯示所有函式類型的函式。該類別的函式或所有函式均將在「類別」選取下方的清單中顯示。
   - 如果您無法在**類別**中的函式類型之間選取，請繼續 4。
4. 從函式清單中選取一個函式。
5. 為該函式輸入參數。
6. 按一下**確定**。

Calculation Manager 所支援的 Essbase 函式

以下是「區塊儲存」應用程式中 Calculation Manager 所支援的 Oracle Essbase 函式。

表格 9-3 Calculation Manager 所支援的 Essbase 函式

@ABS	@ISANCEST	@MOVSUMX
@ACCUM	@ISATTRIBUTE	@NAME
@ALLANCESTORS	@ISCHILD	@NEXT
@ALIAS	@ISDESC	@NEXTS
@ALLOCATE	@ISGEN	@NEXTSIBLING
@ANCEST	@ISIANCEST	@NOTEQUAL
@ANCESTORS	@ISIBLINGS	@NPV
@ANCESTVAL	@ISICHILD	@PARENT
@ATTRIBUTE	@ISIDESC	@PARENTVAL
@ATTRIBUTEBVAL	@ISIPARENT	@POWER
@ATTRIBUTESVAL	@ISISIBLING	@PREVSIBLING
@ATTRIBUTEVAL	@ISLEV	@PRIOR
@AVG	@ISMBR	@PRIORS
@AVGRANGE	@ISMBRUDA	@PTD
@BETWEEN	@ISMBRWITHATTR	@RANGE
@CALCMODE	@ISPARENT	@RANGEFIRSTVAL
@CHILDREN	@ISRANGENONEMPTY	@RANGELASTVAL
@COMPOUND	@ISSAMEGEN	@RANK
@COMPOUNDGROWTH	@ISSAMELEV	@RDESCENDANTS
@CONCATENATE	@ISSIBLING	@RELATIVE
@CORRELATION	@ISUDA	@RELXRANGE
@COUNT	@LANCESTORS	@REMAINDER
@CREATEBLOCK	@LDESCENDANTS	@REMOVE
@CURGEN	@LEV	@RETURN
@CURLEV	@LEVMBRS	@ROUND
@CURRMBR	@LIKE	@RSIBLINGS
@CURRMBRRANGE	@LIST	@SANCESTVAL
@DATEDIFF	@LN	@SHARE
@DATEPART	@LOG	@SHIFT
@DATEROLL	@LOG10	@SHIFTMINUS
@DECLINE	@LSIBLINGS	@SHIFTPLUS
@DESCENDANTS	@MATCH	@SHIFTSIBLING
@DISCOUNT	@MAX	@SIBLINGS
@ENUMVALUE	@MAXRANGE	@SLN
@EQUAL	@MAXS	@SPARENTVAL
@EXP	@MAXSRANGE	@SPLINE
@EXPAND	@MBRCOMPARE	@STDEV
@FACTORIAL	@MBRPARENT	@STDEVP

表格 9-3 (續) Calculation Manager 所支援的 Essbase 函式

@FORMATDATE	@MDALLOCATE	@STDEVRANGE
@GEN	@MDANCESTVAL	@SUBSTRING
@GENMBRS	@MDPARENTVAL	@SUM
@GROWTH	@MDSHIFT	@SUMRANGE
@IALLANCESTORS	@MEDIAN	@SYD
@IANCESTORS	@MEMBER	@TODATE
@ICHILDREN	@MEMBERAT	@TODATEEX
@IDESCENDANTS	@MERGE	@TODAY
@IALLANCESTORS	@MIN	@TREND
@ILDESCENDANTS	@MINRANGE	@TRUNCATE
@ILSIBLINGS	@MINS	@UDA
@INT	@MINSRANGE	@VAR
@INTEREST	@MOD	@VARPER
@INTERSECT	@MODE	@VARIANCE
@IRDESCENDANTS	@MOVAVG	@VARIANCEP
@IRR	@MOVMAX	@WITHATTR
@IRREX	@MOVMED	@XRANGE
@IRSIBLINGS	@MOVMIN	@XREF
@ISACCTYPE	@MOVSUM	@XWRITE

使用自訂函式

使用自訂函式執行任務，例如複製和匯出資料、移除及新增文字字串的單引號或雙引號、比較兩個文字字串，以及將日期轉換為其他格式。

另請參閱：

- 關於自訂函式
- 使用帶有年度參數的自訂函式
- @CalcMgrExcel 帶有日期參數的自訂函式
- 位元函式
- 計數器函式
- 日期/時間函式
- 財務函式
- 記錄函式
- 數學函式
- 統計函式
- 字串函式

關於自訂函式

您在 Calculation Manager 條件、指令碼和公式元件中從函式選取器存取自訂函式。

自訂函式可讓您執行的任務包括，複製和匯出資料、在文字字串中移除和新增單引號和雙引號、比較兩個字串、將日期轉換為 `YYYYMMDD` 格式，以及其他許多任務。

您可以在規劃商業規則元件中使用自訂函式。

使用帶有年度參數的自訂函式

在某些自訂函式中，參數可具有包含 *year* 選項的下拉清單。如果您在名為 *year* 的應用程式中具有維度或成員，使用自訂函式與 *year* 的選項將無法驗證。這對於同時也是維度名稱或成員名稱的任何參數選擇 (不只是 *year*)，都可能會造成問題。

若要解決這個問題，在從參數的下拉式清單中選取 *year* 後，在規則中新增 `@name( )` 以括住 year，使其顯示如下：`@name(year)`。

@CalcMgrExcel 帶有日期參數的自訂函式

用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。您可以使用下列函式，將日期從 `YYYYMMDD` 格式 (序列格式) 轉換為 Excel 格式：

- `@CalcMgrExcelDATE` 傳回特殊日期的序號。

  以下範例將 `20181214` (`YYYYMMDD` 格式) 轉換為 Excel 日期

  `@CalcMgrExcelDATE(20181214)`

- `@CalcMgrDateToExcel` 將 `YYYYMMDD` 格式的單一日期轉換為 Excel 日期
- `@CalcMgrDatesToExcel` 將 `YYYYMMDD` 格式的多個日期轉換為 Excel 日期

  在下列範例中：

  `@CalcMgrDatesToExcel(@LIST("Jan"->"Date_123", "Feb"->Date_123)`

  `"Jan"->"Date_123"` 會顯示為 `01/31/19`，而 `"Feb"->Date_123` 會顯示為 `02/31/19`

位元函式

另請參閱：

- @CalcMgrBitAnd
- @CalcMgrBitOR
- @CalcMgrBitExOR
- @CalcMgrBitExBoolOR
- @CalcMgrBitCompliment
- @CalcMgrBitShiftLeft
- @CalcMgrBitShiftRight
- @CalcMgrBitUnsignedShiftRight

@CalcMgrBitAnd

目的：

執行位元 AND 運算，比較第一個運算子的每個位元與第二個運算子的對應位元。如果兩個位元為 1，對應的結果位元會設為 1；否則，對應的結果位元會設為 0。

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.BitwiseFunctions.AND(double,double)

CDF Spec: @CalcMgrBitAnd(number1,number2)
```

@CalcMgrBitOR

目的：

執行位元 OR 運算，比較第一個運算子的每個位元與第二個運算子的對應位元。如果任一位元為 1，對應的結果位元會設為 1；否則，對應的結果位元會設為 0。

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.BitwiseFunctions.OR(double,double)

CDF Spec: @CalcMgrBitOR(number1,number2
```

@CalcMgrBitExOR

目的：

執行互斥位元 OR 運算，比較第一個運算子的每個位元與第二個運算子的對應位元。如果任一位元為 1，對應的結果位元會設為 1；否則，對應的結果位元會設為 0。

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.BitwiseFunctions.EXCLUSIVEOR(double,double)

CDF Spec: @CalcMgrBitExOR(number1,number2)
```

@CalcMgrBitExBoolOR

目的：

執行互斥位元 OR 運算，比較第一個運算子的每個位元與第二個運算子的對應位元。如果任一位元為 1，對應的結果位元會設為 1；否則，對應的結果位元會設為 0。

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.BitwiseFunctions.EXCLUSIVEOR(double,double)

CDF Spec: @CalcMgrBitExOR(number1,number2)
```

@CalcMgrBitCompliment

目的：

執行一元位元補數，其會回復每個位元。

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.BitwiseFunctions.UNARYCOMPLIMENT(double)
```

```
CDF Spec: @CalcMgrBitCompliment(number1)
```

@CalcMgrBitShiftLeft

目的：

執行已簽署的左移位。

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.BitwiseFunctions.SIGNEDSHIFTLEFT(double,dou
ble)
```

```
CDF Spec: @CalcMgrBitShiftLeft(number1,number2)
```

@CalcMgrBitShiftRight

目的：

執行已簽署的右移位。

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.BitwiseFunctions.SIGNEDSHIFTRIGHT(double,do
uble)
```

```
CDF Spec: @CalcMgrBitShiftRight(number1,number2)
```

@CalcMgrBitUnsignedShiftRight

目的：

執行未簽署的右移位。

語法：

```
com.hyperion.calcmgr.common.excel.cdf.BitwiseFunctions.UNSIGNEDSHIFTRIGHT(double,
double)
```

```
@CalcMgrBitUnsignedShiftRight(number1,number2)
```

計數器函式

另請參閱：

- @CalcMgrCounterAddNumber

- @CalcMgrCounterAddText
- @CalcMgrCounterClear
- @CalcMgrCounterClearAll
- @CalcMgrCounterClearKey
- @CalcMgrCounterDecrement
- @CalcMgrCounterDecrementKey
- @CalcMgrCounterGetKeyNumber
- @CalcMgrCounterGetKeyText
- @CalcMgrCounterGetNumber
- @CalcMgrCounterGetText
- @CalcMgrCounterIncrement
- @CalcMgrCounterIncrementKey
- @CalcMgrCounterUpdate
- @CalcMgrCounterUpdateNumber
- @CalcMgrCounterUpdateNumberText
- @CalcMgrCounterUpdateText

@CalcMgrCounterAddNumber

目的：

將數字新增至計數器並傳回機碼

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.CounterFunctions.addNumber(double)
```

```
CDF Spec: @CalcMgrCounterAddNumber(number)
```

@CalcMgrCounterAddText

目的：

將文字字串新增至計數器並傳回機碼

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.CounterFunctions.addText(String)
```

```
CDF Spec: @CalcMgrCounterAddText(text)
```

@CalcMgrCounterClear

目的：

清除由機碼指定的計數器

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.CounterFunctions.clear(double)
```

```
CDF Spec: @CalcMgrCounterClear(key)
```

@CalcMgrCounterClearAll

目的：

將所有機碼和值從計數器移除

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.CounterFunctions.clearAll()
```

```
CDF Spec: @CalcMgrCounterClearAll()
```

@CalcMgrCounterClearKey

目的：

將值從與機碼相關聯的計數器移除

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.CounterFunctions.clearKey(String)
```

```
CDF Spec: @CalcMgrCounterClearKey(key)
```

@CalcMgrCounterDecrement

目的：

根據機碼遞減計數器中的值。如果找不到機碼，則會為機碼設定零值

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.CounterFunctions.decrement(double)
```

```
CDF Spec: @CalcMgrCounterDecrement(key)
```

@CalcMgrCounterDecrementKey

目的：

根據機碼遞減計數器中的值。如果找不到機碼，則會為機碼設定零值

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.CounterFunctions.decrementKey(String)
```

```
CDF Spec: @CalcMgrCounterDecrementKey(key)
```

@CalcMgrCounterGetKeyNumber

目的：

根據機碼傳回在計數器中找到的文字。如果找不到機碼，則會傳回缺少的值。

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CounterFunctions.getKeyNumber(String,double)
```

```
CDF Spec: @CalcMgrCounterGetKeyNumber(key, missing_value)
```

@CalcMgrCounterGetKeyText

目的：

根據機碼傳回在計數器中找到的文字。如果找不到機碼，則會傳回缺少的值。

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CounterFunctions.getKeyText(String,String)

CDF Spec: @CalcMgrCounterGetKeyText(key,missing_value)
```

@CalcMgrCounterGetNumber

目的：

從機碼指定的計數器傳回數字。如果找不到機碼，或值不是數字，則會傳回缺少的值。

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CounterFunctions.getNumber(double,double)

CDF Spec: @CalcMgrCounterGetNumber(key,missingValue)
```

@CalcMgrCounterGetText

目的：

根據機碼傳回在計數器中找到的文字。如果找不到機碼，則會傳回缺少的值。

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CounterFunctions.getText(double,String)

CDF Spec: @CalcMgrCounterGetText(key,missing_value)
```

@CalcMgrCounterIncrement

目的：

在機碼指定的計數器中遞增值。

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.CounterFunctions.increment(double)

CDF Spec: @CalcMgrCounterIncrement(key)
```

@CalcMgrCounterIncrementKey

目的：

根據機碼遞增計數器中的值。如果找不到機碼，則會為機碼設定零值

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.CounterFunctions.incrementKey(String)
```

```
CDF Spec: @CalcMgrCounterIncrementKey(key)
```

@CalcMgrCounterUpdate

目的：

在帶有指定機碼的計數器中設定數字

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CounterFunctions.update(double,double)

CDF Spec: @CalcMgrCounterUpdate(key, number)
```

@CalcMgrCounterUpdateNumber

目的：

在帶有指定機碼的計數器中更新數字

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CounterFunctions.updateNumber(String,double)

CDF Spec: @CalcMgrCounterUpdateNumber(key, number)
```

@CalcMgrCounterUpdateNumberText

目的：

在帶有指定機碼的計數器中更新數字

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CounterFunctions.updateNumberText(double,String)

CDF Spec: @CalcMgrCounterUpdateNumberText(key,number)
```

@CalcMgrCounterUpdateText

目的：

在帶有指定機碼的計數器中更新文字

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CounterFunctions.updateText(String,String)

CDF Spec: @CalcMgrCounterUpdateText(key, text)
```

日期/時間函式

另請參閱：

- @CalcMgrAddDate
- @CalcMgrAddDatePart

- @CalcMgrAddDays
- @CalcMgrAddMonths
- @CalcMgrAddWeeks
- @CalcMgrAddYears
- @CalcMgrDateDiff
- @CalcMgrDateToExcel
- @CalcMgrDatesToExcel
- @CalcMgrDateTimeToExcel
- @CalcMgrDateTimesToExcel
- @CalcMgrDateToString
- @CalcMgrDaysBetween
- @CalcMgrDaysDiff
- @CalcMgrDiffDate
- @CalcMgrExcelADD
- @CalcMgrExcelDATE
- @CalcMgrExcelDATEDIF
- @CalcMgrExcelDAYOFYEAR
- @CalcMgrExcelDAYS360
- @CalcMgrExcelDAYSINMONTH
- @CalcMgrExcelEOMONTH
- @CalcMgrExcelHOUR
- @CalcMgrExcelMINUTE
- @CalcMgrExcelMONTH
- @CalcMgrExcelNETWORKDAYS
- @CalcMgrExcelSECOND
- @CalcMgrExcelToDate
- @CalcMgrExcelToDateTime
- @CalcMgrExcelWEEKNUM
- @CalcMgrExcelWEEKDAY
- @CalcMgrExcelWORKDAY
- @CalcMgrExcelYEAR
- @CalcMgrExcelYEARFRAC
- @CalcMgrGetCurrentDate
- @CalcMgrGetCurrentDateTZ
- @CalcMgrGetCurrentDateTime
- @CalcMgrGetCurrentDateTimeTZ

- @CalcMgrGetCustomDate
- @CalcMgrGetCustomDateTime
- @CalcMgrGetDatePart
- @CalcMgrGetDateTimePart
- @CalcMgrGetDay
- @CalcMgrGetDayOfYear
- @CalcMgrGetFormattedDate
- @CalcMgrGetMaxDaysInMonth
- @CalcMgrGetMonth
- @CalcMgrGetStringFormattedDateTime
- @CalcMgrGetWeekOfMonth
- @CalcMgrGetWeekOfYear
- @CalcMgrGetYear
- @CalcMgrIsLeapYear
- @CalcMgrMonthsBetween
- @CalcMgrMonthsDiff
- @CalcMgrRollDate
- @CalcMgrRollDay
- @CalcMgrRollMonth
- @CalcMgrRollYear
- @CalcMgrWeeksBetween
- @CalcMgrWeeksDiff
- @CalcMgrYearsBetween
- @CalcMgrYearsDiff

@CalcMgrAddDate

目的：

將指定的年數、月數及天數新增至格式為 `YYYYMMDD` 的日期

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CalendarFunctions.addDate(int,int,int,int)

CDF Spec: @CalcMgrAddDate(date, years, months, days)
```

@CalcMgrAddDatePart

目的：

將指定的年數/月數/天數/週數新增至格式為 `YYYYMMDD` 的日期。「date_part」可以是下列其中之一：「day」、「month」、「week」、「year」

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CalendarFunctions.addDatePart(int,String,int)
```

```
CDF Spec: @CalcMgrAddDatePart(date,date_part, amountToAdd)
```

@CalcMgrAddDays

目的：

將指定的天數新增至格式為 YYYYMMDD 的日期

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.CalendarFunctions.addDays(int,int)
```

```
CDF Spec: @CalcMgrAddDays(date, daysToAdd)
```

@CalcMgrAddMonths

目的：

將指定的月數新增至格式為 YYYYMMDD 的日期

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.CalendarFunctions.addMonths(int,int)
```

```
CDF Spec: @CalcMgrAddMonths(date, monthsToAdd)
```

@CalcMgrAddWeeks

目的：

將指定的週數新增至格式為 YYYYMMDD 的日期

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.CalendarFunctions.addWeeks(int,int)
```

```
CDF Spec: @CalcMgrAddWeeks(date, weeksToAdd)
```

@CalcMgrAddYears

目的：

將指定的年數新增至格式為 YYYYMMDD 的日期

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.CalendarFunctions.addYears(int,int)
```

```
CDF Spec: @CalcMgrAddYears(date, yearsToAdd)
```

@CalcMgrDateDiff

目的：

依照標準格勒哥里曆，就指定的日期部分傳回格式為 YYYYMMDD 的兩個輸入日期之間的差異 (數字)

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CalendarFunctions.dateDiff(int,int,String)

CDF Spec: @CalcMgrDateDiff(fromDate,toDate,datePart)
```

> **備註：**
>
> @CalcMgrDateDiff 僅會傳回正數。如果您希望傳回負數 (如果適用)，請使用 @CalcMgrDiffDate。

@CalcMgrDateToExcel

目的：

將 YYYYMMDD 格式的單一日期轉換成 Excel 日期

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelDateTimeFunctions.DATE_TOEXCEL(double)
CDF Spec: @CalcMgrDateToExcel(date)
```

@CalcMgrDatesToExcel

目的：

將 YYYYMMDD 格式的多個日期轉換成 Excel 日期

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelDateTimeFunctions.DATES_TOEXCEL(doubl
e[])

CDF Spec: @CalcMgrDatesToExcel(dates)
```

@CalcMgrDateTimeToExcel

目的：

將 YYYYMMDDHHMMSS 格式的單一日期轉換成 Excel 日期

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelDateTimeFunctions.DATETIME_TOEXCEL(dou
ble)
CDF Spec: @CalcMgrDateTimeToExcel(date)
```

@CalcMgrDateTimesToExcel

目的：

將 YYYYMMDDHHMMSS 格式的多個日期轉換成 Excel 日期

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelDateTimeFunctions.DATETIMES_TOEXCEL(do
uble[])

CDF Spec: CalcMgrDateTimesToExcel(dates)
```

@CalcMgrDateToString

目的：

利用提供的格式，以字串形式傳回 YYYYMMDD 格式的日期。如需瞭解格式，請參閱 Java 文件中的 SimpleDateFormat

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CalendarFunctions.dateToString(int,String)

CDF Spec: @CalcMgrDateToString(date, format)
```

@CalcMgrDaysBetween

目的：

傳回介於格式為 YYYYMMDD 的兩個日期之間的天數

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CalendarFunctions.daysBetween(int,int)
CDF Spec: @CalcMgrDaysBetween(fromDate,toDate)
```

> **備註：**
>
> @CalcMgrDaysBetween 僅會傳回正數。如果您希望傳回負數 (如果適用)，請使用 @CalcMgrDaysDiff。

@CalcMgrDaysDiff

目的：

傳回介於格式為 YYYYMMDD 的兩個日期之間的天數

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.CalendarFunctions.daysDiff(int,int)
CDF Spec: @CalcMgrDaysDiff(fromDate,toDate)
```

> **備註：**
>
> 當使用 `@CalcMgrDaysDiff` 時，如果函式中的第一個日期晚於第二個日期，則會傳回負數。如果函式中的第一個日期早於第二個日期，則會傳回正數。如果您僅要傳回正數，請使用 @CalcMgrDaysBetween。

@CalcMgrDiffDate

目的：

依照標準格勒哥里曆，就指定的日期部分傳回格式為 `YYYYMMDD` 的兩個輸入日期之間的差異 (數字)

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CalendarFunctions.diffDate(int,int,String)

CDF Spec: @CalcMgrDiffDate(fromDate,toDate,datePart)
```

> **備註：**
>
> 當使用 `@CalcMgrDiffDate` 時，如果函式中的第一個日期晚於第二個日期，則會傳回負數。如果函式中的第一個日期早於第二個日期，則會傳回正數。如果您希望僅傳回正數，請使用 @CalcMgrDateDiff。

@CalcMgrExcelADD

目的：

新增最新的量

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelDateTimeFunctions.ADD(double,double,St
ring)

CDF Spec: @CalcMgrExcelADD(date, amount, what)
```

@CalcMgrExcelDATE

目的：

傳回特殊日期的序號

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelDateTimeFunctions.DATE(double,double,d
ouble)

CDF Spec: @CalcMgrExcelDATE(year,month,day)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式

@CalcMgrExcelDATEDIF

目的：

計算兩個日期之間的天數、月數，或年數。

在您需要計算帳齡的公式中很有用

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelDateTimeFunctions.DATEDIF(double,doubl
e,String)

CDF Spec: @CalcMgrExcelDATEDIF(start_date, end_date,unit)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelDAYOFYEAR

目的：

將序號轉換成一年中的日期

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelDateTimeFunctions.DAYOFYEAR(double)

CDF Spec: @CalcMgrExcelDAYOFYEAR(Date)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelDAYS360

目的：

根據 360 天的年度計算兩個日期之間的天數

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelDateTimeFunctions.DAYS360(double,doubl
e,boolean)
```

```
CDF Spec: @CalcMgrExcelDAYS360(start_date, end_date, method)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelDAYSINMONTH

目的：

將序號轉換成月份中的天數

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelDateTimeFunctions.DAYSINMONTH(double)
```

```
CDF Spec: @CalcMgrExcelDAYSINMONTH(date)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelEOMONTH

目的：

傳回特定月數之前或之後月份最後一天的序號

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelDateTimeFunctions.EOMONTH(double,doubl
e)
```

```
CDF Spec: @CalcMgrExcelEOMONTH(dateValue, adjustmentMonths)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelHOUR

目的：

將序號轉換成一天中的時數

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelDateTimeFunctions.HOUR(double)

CDF Spec: @CalcMgrExcelHOUR(date)
```

@CalcMgrExcelMINUTE

目的：

將序號轉換成分鐘數

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelDateTimeFunctions.MINUTE(double)

CDF Spec: @CalcMgrExcelMINUTE(date)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelMONTH

目的：

將序號轉換成月份

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelDateTimeFunctions.MONTH(double)

CDF Spec: @CalcMgrExcelMONTH(Date)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelNETWORKDAYS

目的：

傳回兩個日期間的工作天數

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelDateTimeFunctions.NETWORKDAYS(double,d
ouble,double[])

CDF Spec: @CalcMgrExcelNETWORKDAYS(startDate, endDate, holidays)
```

> **備註：**
>
> 如果要以規劃日期格式傳遞日期，則必須使用 @CalcMgrDateToExcel，將規劃日期轉換成 Excel 日期。

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelSECOND

目的：

將序號轉換成秒

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelDateTimeFunctions.SECOND(double)

CDF Spec: @CalcMgrExcelSECOND(date)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelToDate

目的：

將 Excel 日期轉換成 `YYYYMMDD` 格式。

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelDateTimeFunctions.EXCEL_TODATE(double)
CDF Spec: @CalcMgrExcelToDate(excel_date)
```

@CalcMgrExcelToDateTime

目的：

將 Excel 日期轉換成 `YYYYMMDDHHMMSS` 格式。

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelDateTimeFunctions.EXCEL_TODATETIME(dou
ble)
CDF Spec: @CalcMgrExcelToDateTime(excel_date)
```

@CalcMgrExcelWEEKNUM

目的：

傳回特殊日期的週數。例如，包含「1 月 1 日」的週是當年的第一週，而且編號為「第 1 週」。

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelDateTimeFunctions.WEEKNUM(double,doubl
e)

CDF Spec: @CalcMgrExcelWEEKNUM(date, method)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelWEEKDAY

目的：

傳回對應日期之一週中的日期依預設會以整數提供日期，範圍從 1 (星期日) 到 7 (星期六)。

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelDateTimeFunctions.WEEKDAY(double,doubl
e)

CDF Spec: @CalcMgrExcelWEEKDAY(serial_number, return_type)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelWORKDAY

目的：

傳回特定工作天數之前或之後的日期序號

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelDateTimeFunctions.WORKDAY(double,doubl
e,double[])
```

```
CDF Spec: @CalcMgrExcelWORKDAY(startDate, days, holidays)
```

> **備註：**
>
> 如果要以規劃日期格式傳遞日期，則必須使用 @CalcMgrDateToExcel，將規劃日期轉換成 Excel 日期。

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelYEAR

目的：

將序號轉換成年

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelDateTimeFunctions.YEAR(double)
```

```
CDF Spec: @CalcMgrExcelYEAR(date)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelYEARFRAC

目的：

傳回代表 `start_date` 與 `end_date` 之間天數的年分。

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelDateTimeFunctions.YEARFRAC(double,double,double)
```

```
CDF Spec: @CalcMgrExcelYEARFRAC(startDate, endDate, basis)
```

> **備註：**
>
> 如果要以規劃日期格式傳遞日期，則必須使用 @CalcMgrDateToExcel，將規劃日期轉換成 Excel 日期。

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrGetCurrentDate

目的：

以 `YYYYMMDD` 格式傳回目前日期 (例如：20140101)

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.CalendarFunctions.getCurrentDate(int)
```

```
CDF Spec: @CalcMgrGetCurrentDate()
```

@CalcMgrGetCurrentDateTZ

目的：

傳回以 `YYYYMMDD` 格式提供的時區目前日期 (例如：20140101)

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CalendarFunctions.getCurrentDate(String)
```

```
CDF Spec: @CalcMgrGetCurrentDateTZ(timeZone)
```

@CalcMgrGetCurrentDateTime

目的：

以 `YYYYMMDDHHMMSS` 格式傳回目前的日期與時間。例如：20140101143001 (年_月_日_時_分_秒)

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CalendarFunctions.getCurrentDateTime()
```

```
CDF Spec: @CalcMgrGetCurrentDateTime()
```

@CalcMgrGetCurrentDateTimeTZ

目的：

傳回以 YYYYMMDDHHMMSS 格式提供的時區目前日期與時間例如：20140101143001 (年_月_日_時_分_秒)

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CalendarFunctions.getCurrentDateTime(String)

CDF Spec: @CalcMgrGetCurrentDateTimeTZ(timeZone)
```

@CalcMgrGetCustomDate

目的：

以 YYYYMMDD 格式傳回自訂日期。

例如，20140101

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CalendarFunctions.getDate(double,double,double)

CDF Spec: @CalcMgrGetCustomDate(year, month, day)
```

@CalcMgrGetCustomDateTime

目的：

傳回 YYYYMMDDHHMMSS 格式的自訂日期與時間。例如：20140101143001 (年_月_日_時_分_秒)

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CalendarFunctions.getDateTime(double,double,doubl
e,double,double,double):

CDF Spec: @CalcMgrGetCustomDateTime(year, month, day, hour, min, sec)
```

@CalcMgrGetDatePart

目的：

以數字形式傳回某個日期 (格式為 YYYYMMDD) 的 Year/Month/DayOfMonth/WeekOfYear/WeekOfMonth/DayOfYear。「date_part_ex」可以是：「dayofmonth」、「dayofyear」、「month」、「weekofmonth」、「weekofyear」或「year」

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CalendarFunctions.datePart(Double,String)

CDF Spec: @CalcMgrGetDatePart(date, date_part_ex)
```

@CalcMgrGetDateTimePart

目的：

以數字形式從日期傳回 `Year/Month/DayOfMonth/WeekOfYear/WeekOfMonth/DayOfYear/Hour/Minute/Seconds` 。

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CalendarFunctions.dateTimePart(double,String)
```

```
CDF Spec: @CalcMgrGetDateTimePart(date, date_part_ex)
```

@CalcMgrGetDay

目的：

傳回格式為 `YYYYMMDD` 的日期是哪一天

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.CalendarFunctions.getDay(int)
```

```
CDF Spec: @CalcMgrGetDay(date)
```

@CalcMgrGetDayOfYear

目的：

傳回格式為 `YYYYMMDD` 的日期是一年中的哪一天 (1-366)。

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.CalendarFunctions.getDayOfYear(int)
```

```
CDF Spec: @CalcmgrGetDayOfYear(date)
```

@CalcMgrGetFormattedDate

目的：

將日期轉換成 `YYYYMMDD` 格式。例如，`@CalcMgrGetFormattedDate(12302014, "mmddyyyy")` 將傳回 20141230

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CalendarFunctions.getFormattedDate(int,String)
```

```
CDF Spec: @CalcMgrGetFormattedDate(date, format)
```

@CalcMgrGetMaxDaysInMonth

目的：

傳回格式為 `YYYYMMDD` 之日期所在月份的最大天數

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CalendarFunctions.getActualMaximumDays(int)

CDF Spec: @CalcMgrGetMaxDaysInMonth(date)
```

@CalcMgrGetMonth

目的：

傳回格式為 YYYYMMDD 的日期所在月份

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.CalendarFunctions.getMonth(int)

CDF Spec: @CalcMgrGetMonth(date)
```

@CalcMgrGetStringFormattedDateTime

目的：

將依照格式定義的日期轉換成 YYYYMMddHHmmss 格式的日期。

例如：@CalcMgrGetFormattedDate(12302014, "MMddyyyyHHmmss") 傳回 201412301430。

如需詳細資訊，請參閱 Java 文件中的 "SimpleDateFormat"。

格式的可能值：mmddyyyyHHmmss、ddmmyyyyHHmmss、yyyyddmmHHmmss

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CalendarFunctions.getStringFormattedDateTime(Stri
ng,String)
CDF Spec: @CalcMgrGetStringFormattedDateTime(date, format)
```

@CalcMgrGetWeekOfMonth

傳回格式為 YYYYMMDD 的日期是該月的第幾週

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.CalendarFunctions.getWeekOfMonth(int)

CDF Spec: @CalcMgrGetWeekOfMonth(date)
```

@CalcMgrGetWeekOfYear

目的：

傳回格式為 YYYYMMDD 的日期是一年中的哪一週

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.CalendarFunctions.getWeekOfYear(int)

CDF Spec: @CalcMgrGetWeekOfYear(date)
```

@CalcMgrGetYear

目的：

傳回某個日期 (格式為 YYYYMMDD) 的年度。「date_part_ex」必須是「dayofmonth」、「dayofyear」、「month」、「weekofmonth」、「weekofyear」或「year」

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.CalendarFunctions.getYear(int)

CDF Spec: @CalcMgrGetYear(date)
```

@CalcMgrIsLeapYear

目的：

判斷指定的日期是否為閏年。日期必須是 YYYYMMDD 或 YYYY 格式 (例如：20140101 或 2014)

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.CalendarFunctions.isLeapYear(int)
CDF Spec: @CalcMgrIsLeapYear(date)
```

@CalcMgrMonthsBetween

目的：

傳回介於格式為 YYYYMMDD 的兩個日期之間的月數

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CalendarFunctions.monthsBetween(int,int)
CDF Spec: @CalcMgrMonthsBetween(fromDate,toDate)
```

> **備註：**
>
> @CalcMgrMonthsBetween 僅會傳回正數。如果您希望傳回負數 (如果適用)，請使用 @CalcMgrMonthsDiff。

@CalcMgrMonthsDiff

目的：

傳回介於格式為 YYYYMMDD 的兩個日期之間的月數

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.CalendarFunctions.monthsDiff(int,int)
CDF Spec: @CalcMgrMonthsDiff(fromDate,toDate)
```

> **備註：**
>
> 當使用 `@CalcMonthsDiff` 時，如果函式中的第一個日期晚於第二個日期，則會傳回負數。如果函式中的第一個日期早於第二個日期，則會傳回正數。如果您希望僅傳回正數，請使用 @CalcMgrMonthsBetween。

@CalcMgrRollDate

目的：

加上或減去 (上或下) 給定日期欄位上的一單位時間，而不變更較大欄位。

例如，`@CalcMgrRollDate(19960131,"month",@_true)` 會產生日期 `19960229`。`@CalcMgrRollDate(19960131,"day",@_true)` 會產生日期 `19960101`。

date_part 的可能值為：日、月、週和年。

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CalendarFunctions.rollDate(int,String,boolean)

CDF Spec: @CalcMgrRollDate(date,date_part,up)
```

@CalcMgrRollDay

目的：

累加或累降天數至 `YYYYMMDD` 格式的日期

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CalendarFunctions.rollDay(int,boolean)

CDF Spec: @CalcMgrRollDay(date,up)
```

@CalcMgrRollMonth

目的：

累加或累降月數至 `YYYYMMDD` 格式的日期

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CalendarFunctions.rollMonth(int,boolean)

CDF Spec: @CalcMgrRollMonth(date,up)
```

@CalcMgrRollYear

目的：

累加或累降年數至 `YYYYMMDD` 格式的日期

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CalendarFunctions.rollYear(int,boolean)

CDF Spec: @CalcMgrRollYear(date,up)
```

@CalcMgrWeeksBetween

目的：

傳回介於格式為 YYYYMMDD 的兩個日期之間的週數

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CalendarFunctions.weeksBetween(int,int)
CDF Spec: @CalcMgrWeeksBetween(fromDate,toDate)
```

> **備註：**
>
> @CalcMgrWeeksBetween 僅會傳回正數。如果您希望傳回負數 (如果適用)，請使用 @CalcMgrWeeksDiff。

@CalcMgrWeeksDiff

目的：

傳回介於格式為 YYYYMMDD 的兩個日期之間的週數

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.CalendarFunctions.weeksDiff(int,int)
CDF Spec: @CalcMgrWeeksDiff(fromDate,toDate)
```

> **備註：**
>
> 當使用 @CalcMgrWeeksDiff 時，如果函式中的第一個日期晚於第二個日期，則會傳回負數。如果函式中的第一個日期早於第二個日期，則會傳回正數。如果您希望僅傳回正數，請使用 @CalcMgrWeeksBetween。

@CalcMgrYearsBetween

目的：

傳回介於格式為 YYYYMMDD 的兩個日期之間的年數

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.CalendarFunctions.yearsBetween(int,int)

CDF Spec: @CalcMgrYearsBetween(fromDate,toDate)
```

> **備註：**
>
> `@CalcMgrYearsBetween` 僅會傳回正數。如果您希望傳回負數 (如果適用)，請使用 @CalcMgrYearsDiff。

@CalcMgrYearsDiff

目的：

傳回介於格式為 `YYYYMMDD` 的兩個日期之間的年數

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.CalendarFunctions.yearsDiff(int,int)

CDF Spec: @CalcMgrYearsDiff(fromDate,toDate)
```

> **備註：**
>
> 當使用 `@CalcMgrYearsDiff` 時，如果函式中的第一個日期晚於第二個日期，則會傳回負數。如果函式中的第一個日期早於第二個日期，則會傳回正數。如果您希望僅傳回正數，請使用 @CalcMgrYearsBetween。

財務函式

另請參閱：

- @CalcMgrExcelACCRINT
- @CalcMgrExcelACCRINTM
- @CalcMgrExcelAMORDEGRC
- @CalcMgrExcelAMORLINC
- @CalcMgrExcelCOUPDAYBS
- @CalcMgrExcelCOUPDAYS
- @CalcMgrExcelCOUPDAYSNC
- @CalcMgrExcelCOUPNCD
- @CalcMgrExcelCOUPNUM
- @CalcMgrExcelCOUPPCD
- @CalcMgrExcelCUMIPMT
- @CalcMgrExcelCUMPRINC
- @CalcMgrExcelDB
- @CalcMgrExcelDDB
- @CalcMgrExcelDISC
- @CalcMgrExcelDOLLARDE

- @CalcMgrExcelDOLLARFR
- @CalcMgrExcelDURATION
- @CalcMgrExcelEFFECT
- @CalcMgrExcelFV
- @CalcMgrExcelFVSCHEDULE
- @CalcMgrExcelMDURATION
- @CalcMgrExcelINTRATE
- @CalcMgrExcelPMT
- @CalcMgrExcelIRR
- @CalcMgrExcelISPMT
- @CalcMgrExcelMIRR
- @CalcMgrExcelNPER
- @CalcMgrExcelNPV
- @CalcMgrExcelPPMT
- @CalcMgrExcelPRICE
- @CalcMgrExcelPRICEDISC
- @CalcMgrExcelPRICEMAT
- @CalcMgrExcelPV
- @CalcMgrExcelRATE
- @CalcMgrExcelRECEIVED
- @CalcMgrExcelSLN
- @CalcMgrExcelSYD
- @CalcMgrExcelTBILLEQ
- @CalcMgrExcelTBILLPRICE
- @CalcMgrExcelTBILLYIELD
- @CalcMgrExcelXIRR
- @CalcMgrExcelXNPV
- @CalcMgrExcelYIELD
- @CalcMgrExcelYIELDDISC
- @CalcMgrExcelYIELDMAT

@CalcMgrExcelACCRINT

目的：

傳回支付定期利息之證券的累計利息

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.ACCRINT(double,doub
le,double,double,double,double,double,boolean)
```

```
CDF Spec: @CalcMgrExcelACCRINT(issue, firstinterest, settlement, rate, par,
frequency, basis, method)
```

@CalcMgrExcelACCRINTM

目的：

傳回在到期時支付利息之證券的累計利息

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.ACCRINTM(issue,
settlement, rate, par, basis)
```

```
CDF Spec:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.ACCRINTM(double,dou
ble,double,double,double)
```

@CalcMgrExcelAMORDEGRC

目的：

使用折舊係數，針對每個會計期間傳回折舊。

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.AMORDEGRC(double,do
uble,double,double,double,double,double)
```

```
CDF Spec: @CalcMgrExcelAMORDEGRC(cost, purchased, firstPeriod, salvage, period,
rate, basis)
```

@CalcMgrExcelAMORLINC

目的：

針對每個會計期間傳回折舊

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.AMORLINC(double,dou
ble,double,double,double,double,double)
```

```
CDF Spec: @CalcMgrExcelAMORLINC(cost, date_purchased, first_period, salvage,
period, rate, basis)
```

@CalcMgrExcelCOUPDAYBS

目的：

傳回從付息期間開始到結算日的天數

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.COUPDAYBS(double,do
uble,double,double)
```

```
CDF Spec: @CalcMgrExcelCOUPDAYBS(settlement, maturity, frequency, basis)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelCOUPDAYS

目的：

傳回包含結算日之付息期間的天數

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.COUPDAYS(double,dou
ble,double,double)
```

```
CDF Spec: @CalcMgrExcelCOUPDAYS(settlement, maturity, frequency, basis)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelCOUPDAYSNC

目的：

傳回從結算日到下一個付息日期的天數

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.COUPDAYSNC(double,d
ouble,double,double)
```

```
CDF Spec: @CalcMgrExcelCOUPDAYSNC(settlement, maturity, frequency, basis)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelCOUPNCD

目的：

傳回代表結算日後之下一個付息日期的數字

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.COUPNCD(double,doub
le,double,double)

CDF Spec: @CalcMgrExcelCOUPNCD(settlement, maturity, frequency, basis)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelCOUPNUM

目的：

傳回結算日和到期日之間應付息票的數字，四捨五入至最接近的整數息票

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.COUPNUM(double,doub
le,double,double)

CDF Spec: @CalcMgrExcelCOUPNUM(settlement, maturity, frequency, basis)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelCOUPPCD

目的：

傳回代表結算日前之上一個付息日期的數字

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.COUPPCD(double,doub
le,double,double)

CDF Spec: @CalcMgrExcelCOUPPCD(settlement, maturity, frequency, basis)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelCUMIPMT

目的：

傳回 `start_period` 和 `end_period` 期間在貸款方面應付的累計利息

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.CUMIPMT(double,doub
le,double,double,double,double)
```

```
CDF Spec: @CalcMgrExcelCUMIPMT(rate, nper, pv, start_period, end_period, type)
```

@CalcMgrExcelCUMPRINC

目的：

傳回開始期間和結束期間之間，在貸款方面應付的累計本金

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.CUMPRINC(double,dou
ble,double,double,double,double)
```

```
CDF Spec: @CalcMgrExcelCUMPRINC(rate, per, nper, pv, fv, type)
```

@CalcMgrExcelDB

目的：

傳回使用固定下降結餘方法之特定期間的資產折舊

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.DB(double,double,do
uble,double,double)
```

```
CDF Spec: @CalcMgrExcelDB((cost, salvage, life, period, month)
```

@CalcMgrExcelDDB

目的：

傳回使用固定下降結餘方法或某些其他您指定的方法之特定期間的資產折舊

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.DDB(double,double,d
ouble,double,double)
```

```
CDF Spec: @CalcMgrExcelDDB(cost, salvage, life, period, factor)
```

@CalcMgrExcelDISC

目的：

傳回證券的貼現率

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.DISC(double,double,
double,double,double)
```

```
CDF Spec: @CalcMgrExcelDISC(settlement, maturity, pr, redemption, basis)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelDOLLARDE

目的：

轉換以整數部分和分數部分表示的美元價格，例如將 1.02 轉換成以小數表示的美元價格。分數美元數字有時候用於證券價格。

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.DOLLARDE(double,dou
ble)
```

```
CDF Spec: @CalcMgrExcelDOLLARDE(fractional_dollar,fraction)
```

@CalcMgrExcelDOLLARFR

目的：

將以小數表示的美元價格轉換成以分數表示的美元價格

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.DOLLARFR(double,dou
ble)
```

```
CDF Spec: @CalcMgrExcelDOLLARFR(decimal_dollar, fraction)
```

@CalcMgrExcelDURATION

目的：

傳回具有定期利息支付的證券年度持續時間。**備註：**在使用 `@CalcMgrExcelMDURATION` 函式時，Calculation Manager 和 Excel 兩者的計算可能會不一致。若要進行數字比對，請將小數變更為 7，並使用 Open Office。

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.DURATION(double,dou
ble,double,double,double,double)
```

```
CDF Spec: @CalcMgrExcelDURATION(settlement, maturity, coupon, yld, frequency,
basis)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelEFFECT

目的：

傳回有效的年度利息費率

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.EFFECT(double,doubl
e)
```

```
CDF Spec: @CalcMgrExcelEFFECT(nominal_rate, npery)
```

@CalcMgrExcelFV

目的：

傳回投資的未來值

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.FV(double,double,do
uble,double,double)
```

```
CDF Spec: @CalcMgrExcelFV(rate, nper, pmt, pv, type)
```

@CalcMgrExcelFVSCHEDULE

目的：

傳回套用一系列複利率後初始本金的未來值

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.FVSCHEDULE(double,d
ouble[])
```

```
CDF Spec: @CalcMgrExcelFVSCHEDULE(principal, schedule)
```

@CalcMgrExcelMDURATION

目的：

傳回具有保障票面價值 $100 之證券的 Macauley 已修改期間。**備註：**在使用 `@CalcMgrExcelDURATION` 函式時，Calculation Manager 和 Excel 兩者的計算可能會不一致。若要進行數字比對，請將小數變更為 7，並使用 Open Office。

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.MDURATION(double,do
uble,double,double,double,double)
```

```
CDF Spec: @CalcMgrExcelMDURATION(settlement,maturity,coupon,yld,frequency,basis)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelINTRATE

目的：

傳回完全投資證券的利息費率

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.INTRATE(double,doub
le,double,double,double)
```

```
CDF Spec: @CalcMgrExcelINTRATE(settlement, maturity, investment, redemption,
basis)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelPMT

目的：

傳回年金的定期支付

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.PMT(double,double,d
ouble,double,double)

CDF Spec: @CalcMgrExcelPMT(rate, nper, pv, fv, type)
```

@CalcMgrExcelIRR

目的：

傳回一系列現金流的內部報酬率

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.IRR(double[],double
)

CDF Spec: @CalcMgrExcelIRR(values, guess)
```

@CalcMgrExcelISPMT

目的：

計算特定投資期間的給付利息

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.ISPMT(double,double
,double,double)

CDF Spec: @CalcMgrExcelISPMT(rate, per, nper, pv)
```

@CalcMgrExcelMIRR

目的：

傳回以不同費率提供資金之正負現金流的內部報酬率

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.MIRR(double[],doubl
e,double)

CDF Spec: @CalcMgrExcelMIRR(values, finance_rate, reinvest_rate)
```

@CalcMgrExcelNPER

目的：

傳回投資的期間數量

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.NPER(double,double,
double,double,double)

CDF Spec: @CalcMgrExcelNPER(rate, pmt, pv, fv, type)
```

@CalcMgrExcelNPV

目的：

傳回以一系列定期現金流與貼現率為基礎的投資淨現值

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.NPV(double,double[]
)
```

```
CDF Spec: @CalcMgrExcelNPV(rate, values)
```

@CalcMgrExcelPPMT

目的：

根據定期、常數付款和常數利息費率，針對投資的給定期間傳回本金費用

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.PPMT(double,double,
double,double,double,double)
```

```
CDF Spec: @CalcMgrExcelPPMT(rate, per, nper, pv, fv, type)
```

@CalcMgrExcelPRICE

目的：

傳回支付定期利息之證券之每 $100 面值的價格

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.PRICE(double,double
,double,double,double,double,double)
```

```
CDF Spec: @CalcMgrExcelPRICE(settlement, maturity, rate, yld, redemption,
frequency, basis)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelPRICEDISC

目的：

傳回折扣證券之每 $100 面值的價格

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.PRICEDISC(double,do
uble,double,double,double)
```

```
CDF Spec: @CalcMgrExcelPRICEDISC(settlement, maturity, discount, redemption,
basis)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelPRICEMAT

目的：

傳回到期付息之證券每 $100 面值的價格

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.PRICEMAT(double,dou
ble,double,double,double,double)
```

```
CDF Spec: @CalcMgrExcelPRICEMAT((settlement, maturity, issue, rate, yld, basis)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelPV

目的：

傳回投資的目前值

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.PV(double,double,do
uble,double,double)
```

```
CDF Spec: @CalcMgrExcelPV(rate, nper, pmt, fv, type)
```

@CalcMgrExcelRATE

目的：

傳回每年度期間的利率

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.RATE(double,double,
double,double,double,double)
```

```
CDF Spec: @CalcMgrExcelRATE(nper, pmt, pv, fv, type, guess)
```

@CalcMgrExcelRECEIVED

目的：

傳回完全投資證券在到期時收到的金額

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.RECEIVED(double,dou
ble,double,double,double)
```

```
CDF Spec: @CalcMgrExcelRECEIVED(settlement, maturity, investment, discount,
basis)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelSLN

目的：

傳回單一期間資產的直線折舊

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.SLN(double,double,d
ouble)
```

```
CDF Spec: @CalcMgrExcelSLN(cost, salvage, life)
```

@CalcMgrExcelSYD

目的：

傳回特定期間資產的年數加總折舊

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.SYD(double,double,d
ouble,double)
```

```
CDF Spec: @CalcMgrExcelSYD(cost, salvage, life, per)
```

@CalcMgrExcelTBILLEQ

目的：

傳回國庫券的債券等值收益率

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.TBILLEQ(double,double,double)

CDF Spec: @CalcMgrExcelTBILLEQ(settlement, maturity, discount)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelTBILLPRICE

目的：

傳回國庫券之每 $100 面值的價格

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.TBILLPRICE(double,double,double)

CDF Spec: @CalcMgrExcelTBILLPRICE(settlement, maturity, discount)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelTBILLYIELD

目的：

傳回國庫券的收益率

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.TBILLYIELD(double,double,double)

CDF Spec: @CalcMgrExcelTBILLYIELD(settlement, maturity, pr)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelXIRR

目的：

傳回非特定期間排程現金流的內部報酬率

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.XIRR(double[],doubl
e[],double)
```

```
CDF Spec: @CalcMgrExcelXIRR(values, dates, guess)
```

@CalcMgrExcelXNPV

目的：

傳回非特定期間排程現金流的淨現值

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.XNPV(double,double[
],double[])
```

```
CDF Spec: @CalcMgrExcelXNPV(rate, values, dates)
```

@CalcMgrExcelYIELD

目的：

傳回支付定期利息之證券的收益率

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.YIELD(double,double
,double,double,double,double,double)
```

```
CDF Spec: @CalcMgrExcelYIELD(settlement, maturity, rate, pr, redemption,
frequency, basis)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelYIELDDISC

目的：

傳回貼現證券的年收益。例如，國庫債券。**備註：**在使用 `@CalcMgrExcelYIELDDISC` 函式時，Calculation Manager 和 Excel 兩者的計算可能會不一致。若要進行數字比對，請將小數變更為 7，並使用 Open Office。

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.YIELDDISC(double,do
uble,double,double,double)

CDF Spec: @CalcMgrExcelYIELDDISC(settlement, maturity, pr, redemption, basis)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

@CalcMgrExcelYIELDMAT

目的：

傳回證券的年收益，該證券在到期時支付利息。**備註：**在使用 `@CalcMgrExcelYIELDMAT` 函式時，Calculation Manager 和 Excel 兩者的計算可能會不一致。若要進行數字比對，請將小數變更為 7，並使用 Open Office。

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelFinancialFunctions.YIELDMAT(double,dou
ble,double,double,double,double)

CDF Spec: @CalcMgrExcelYIELDMAT(settlement, maturity, issue, rate, pr, basis)
```

> **備註：**
>
> 用於 `@CalcMgrExcel` 函式的日期參數必須使用 Excel 格式。請參閱@CalcMgrExcel 帶有日期參數的自訂函式。

記錄函式

另請參閱：

- @CalcMgrLogMessageTrace
- @CalcMgrIsValidMember
- @CalcMgrIsValidSLMember

- @CalcMgrSLMember

@CalcMgrLogMessageTrace

目的：

在 Calculation Manager 中啟動規則後，至「日誌訊息」頁籤中新增自訂訊息。

例如，`@CalcMgrLogMessageTrace(@NAME(@CURRMBR(Product)), @NAME(@CURRMBR(Period)) );` 會在「日誌訊息」頁籤中傳回自訂訊息的目前產品成員和期間。

語法：

```
CDM Spec: @CalcMgrLogMessageTrace(Member Names)
```

> **Note:**
>
> 規則在 Calculation Manager 中啟動時才能使用此函式。

@CalcMgrIsValidMember

目的：

成員名稱輸入若為有效成員，則傳回 "true"。

語法：

```
CDM Spec: @CalcMgrIsValidMember(Member Name)
```

@CalcMgrIsValidSLMember

目的：

與數值智慧型列示值關聯的成員若為有效成員，則傳回 "true"。

數值智慧型列示值的計算方式為將 `HSP_ID` 與 `Smartlist Value` 串接，然後查詢「智慧型列示」別名表格中的成員。

語法：

```
CDM Spec: @CalcMgrIsValidSLMember(Smartlist Value)
```

@CalcMgrSLMember

目的：

傳回與數值智慧型列示值關聯的成員。

數值智慧型列示值的計算方式為將 `HSP_ID` 與 `Smartlist Value` 串接，然後查詢「智慧型列示」別名表格中的成員。

語法：

```
CDM Spec: @CalcMgrSLMember(Smartlist Value)
```

數學函式

另請參閱：

- @CalcMgrExcelCEILING
- @CalcMgrExcelCOMBIN
- @CalcMgrExcelEVEN
- @CalcMgrExcelFACT
- @CalcMgrExcelFLOOR
- @CalcMgrExcelGCD
- @CalcMgrExcelLCM
- @CalcMgrExcelMROUND
- @CalcMgrExcelMULTINOMIAL
- @CalcMgrExcelODD
- @CalcMgrExcelPOWER
- @CalcMgrExcelPRODUCT
- @CalcMgrExcelROUNDDOWN
- @CalcMgrExcelROUNDUP
- @CalcMgrExcelSQRT
- @CalcMgrExcelSQRTPI
- @CalcMgrExcelSUMPRODUCT
- @CalcMgrExcelSUMSQ

@CalcMgrExcelCEILING

目的：

將數字無條件進位至 (背離零) 最接近的整數或是最接近之指定基數的倍數

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelMathFunctions.CEILING(double,double)
```

```
CDF Spec: @CalcMgrExcelCEILING(number, significance)
```

@CalcMgrExcelCOMBIN

目的：

為指定數目的物件傳回組合的數目

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelMathFunctions.COMBIN(double,double)
```

```
CDF Spec: @CalcMgrExcelCOMBIN(number, number_chosen)
```

@CalcMgrExcelEVEN

目的：

將數字無條件進位至最接近的偶數整數

語法：

```
Java Class: com.hyperion.calcmgr.common.excel.cdf.ExcelMathFunctions.EVEN(double)
```

```
CDF Spec: @CalcMgrExcelEVEN(number)
```

@CalcMgrExcelFACT

目的：

傳回數字的階乘

語法：

```
Java Class: com.hyperion.calcmgr.common.excel.cdf.ExcelMathFunctions.FACT(double)
```

```
CDF Spec: @CalcMgrExcelFACT(number)
```

@CalcMgrExcelFLOOR

目的：

將數字朝向零無條件捨去

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelMathFunctions.FLOOR(double,double)
```

```
CDF Spec: @CalcMgrExcelFLOOR(number, significance)
```

@CalcMgrExcelGCD

目的：

傳回最大公因數

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelMathFunctions.GCD(double[])
```

```
CDF Spec: @CalcMgrExcelGCD(numbers)
```

@CalcMgrExcelLCM

目的：

傳回最小公倍數

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelMathFunctions.LCM(double[])

CDF Spec: @CalcMgrExcelLCM(numbers)
```

@CalcMgrExcelMROUND

目的：

將數字四捨五入至指定的位數。

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelMathFunctions.MROUND(double,double)

CDF Spec: @CalcMgrExcelMROUND(number, num_digits)
```

@CalcMgrExcelMULTINOMIAL

目的：

傳回一組數字的多項式

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelMathFunctions.MULTINOMIAL(double[])

CDF Spec: @CalcMgrExcelMULTINOMIAL(numbers)
```

@CalcMgrExcelODD

目的：

將數字無條件進位至最接近的奇數整數

語法：

```
Java Class: com.hyperion.calcmgr.common.excel.cdf.ExcelMathFunctions.ODD(double)

CDF Spec: @CalcMgrExcelODD(number)
```

@CalcMgrExcelPOWER

目的：

傳回某個數字的次方結果

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelMathFunctions.POWER(double,double)

CDF Spec: @CalcMgrExcelPOWER(number, power)
```

@CalcMgrExcelPRODUCT

目的：

乘以其引數

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelMathFunctions.PRODUCT(double[])

CDF Spec: @CalcMgrExcelPRODUCT(numbers)
```

@CalcMgrExcelROUNDDOWN

目的：

將數字朝向零無條件捨去

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelMathFunctions.ROUNDDOWN(double,double)

CDF Spec: @CalcMgrExcelROUNDDOWN(number, num_digits)
```

@CalcMgrExcelROUNDUP

目的：

將數字背離零無條件進位

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelMathFunctions.ROUNDUP(double,double)

CDF Spec: @CalcMgrExcelROUNDUP(number, num_digits)
```

@CalcMgrExcelSQRT

目的：

傳回正平方根

語法：

```
Java Class: com.hyperion.calcmgr.common.excel.cdf.ExcelMathFunctions.SQRT(double)

CDF Spec: @CalcMgrExcelSQRT(number)
```

@CalcMgrExcelSQRTPI

目的：

傳回 (數字 * pi) 的平方根

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelMathFunctions.SQRTPI(double)

CDF Spec: @CalcMgrExcelSQRTPI(number)
```

@CalcMgrExcelSUMPRODUCT

目的：

傳回對應陣列元件的乘積總和

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelMathFunctions.SUMPRODUCT(double[],doub
le[])

CDF Spec: @CalcMgrExcelSUMPRODUCT(values1, values2)
```

@CalcMgrExcelSUMSQ

目的：

傳回引數的平方總和

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelMathFunctions.SUMSQ(double[])

CDF Spec: @CalcMgrExcelSUMSQ(numbers)
```

統計函式

另請參閱：

- @CalcMgrExcelAVEDEV
- CalcMgrExcelBINOMDIST
- @CalcMgrExcelDEVSQ
- @CalcMgrExcelLARGE
- @CalcMgrExcelMEDIAN
- @CalcMgrExcelNORMSDIST
- @CalcMgrExcelNORMSINV
- @CalcMgrExcelPERCENTILE
- @CalcMgrExcelPERCENTRANK
- @CalcMgrExcelRANK
- @CalcMgrExcelSMALL
- @CalcMgrExcelSTDEV
- @CalcMgrExcelVAR
- @CalcMgrExcelVARP
- @CalcMgrIsFinite

@CalcMgrExcelAVEDEV

目的：

從其均值傳回資料點的絕對離差平均值

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelStatisticalFunctions.AVEDEV(double[])
```

```
CDF Spec: @CalcMgrExcelAVEDEV(numbers)
```

CalcMgrExcelBINOMDIST

目的：

傳回個別項二項式分布機率

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelStatisticalFunctions.BINOMDIST(double,
double, double, boolean)
```

```
CDF Spec:@CalcMgrExcelBINOMDIST(successes, trials, probSuccess, cumulative)
```

參數：

`Cumulative`：決定函數形式的邏輯值

`@_true`：決定函數形式的邏輯值。若累計為 *true*，則 `BINOMDIST` 會傳回累計分佈函數，表示最多有 number_s 次成功的機率

`@_false`：傳回機率質量函數，表示有 number_s 次成功的機率

@CalcMgrExcelDEVSQ

目的：

傳回離差的平方總和

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelStatisticalFunctions.DEVSQ(double[])
```

```
CDF Spec: @CalcMgrExcelDEVSQ(numbers)
```

@CalcMgrExcelLARGE

目的：

傳回第 9 個最高數字

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelStatisticalFunctions.NTHLARGEST(doubl
e[],double)
```

```
CDF Spec: @CalcMgrExcelLARGE(values, rank)
```

@CalcMgrExcelMEDIAN

目的：

傳回指定數字的中位數

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelStatisticalFunctions.MEDIAN(double[])

CDF Spec: @CalcMgrExcelMEDIAN(values)
```

@CalcMgrExcelNORMSDIST

目的：

傳回常態分佈

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelStatisticalFunctions.NORMSDIST(double)

CDF Spec: @CalcMgrExcelNORMSDIST(value)
```

@CalcMgrExcelNORMSINV

目的：

傳回值 z，使標準常態隨機變數以機率 p 取得小於或等於 z 的值。標準常態隨機變數平均值為 0，標準差為 1。

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelStatisticalFunctions.NORMSINV(double)

CDF Spec: @CalcMgrExcelNORMSINV(probability)
```

@CalcMgrExcelPERCENTILE

用途：傳回範圍中值的第 k 個百分位數

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelStatisticalFunctions.PERCENTILE(doubl
e[],double)

CDF Spec: @CalcMgrExcelPERCENTILE(values,percentile)
```

@CalcMgrExcelPERCENTRANK

用途：以資料集的百分比方式傳回資料集中某個值的排名

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelStatisticalFunctions.PERCENTRANK(doubl
e[],double)

CDF Spec: @CalcMgrExcelPERCENTRANK(values, percentile)
```

@CalcMgrExcelRANK

用途：傳回數字清單中某個數字的排名

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelStatisticalFunctions.RANK(double,
double[], boolean)
```

```
CDF Spec: @CalcMgrExcelRANK(value, values, order)
```

參數：

`@_true`：以遞減順序排列值的排名

`@_false`：以遞增順序排列值的排名

@CalcMgrExcelSMALL

目的：

傳回第 9 個最小數字

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelStatisticalFunctions.NTHSMALLEST(doubl
e[],double)
```

```
CDF Spec: @CalcMgrExcelSMALL(values, rank)
```

@CalcMgrExcelSTDEV

目的：

依據範例估計標準差

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelStatisticalFunctions.STDEV(double[])
```

```
CDF Spec: @CalcMgrExcelSTDEV(values)
```

@CalcMgrExcelVAR

目的：

依據範例估計變異數

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelStatisticalFunctions.VAR(double[])
```

```
CDF Spec: @CalcMgrExcelVAR(values)
```

@CalcMgrExcelVARP

目的：

基於整個母體估計變異數

語法：

```
Java Class:
com.hyperion.calcmgr.common.excel.cdf.ExcelStatisticalFunctions.VARP(double[])

CDF Spec: @CalcMgrExcelVARP(values)
```

@CalcMgrIsFinite

目的：

評估指定的成員，判斷其值是否為有限值。如果所指定數字的量級為無限大 (NaN 或 Infinity)，則傳回 *false*；否則會傳回 *true*。

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.StringFunctions.isFinite(double)

CDF Spec: @CalcMgrIsFinite(doubleNumber)
```

範例：

在以下範例中，`@CalcMgrIsFinite` 會對 FIX 陳述式中的成員進行評估，以判斷其值是否為 NaN 或 Infinity。如果 "5800" 的值為 NaN 或 Infinity，則會將該值變更為 #Missing。

```
FIX ("BaseData", FY13, Plan, Working, "111", @Relative(P_100,0))
    "5800" (
        IF (NOT @CalcMgrIsFinite("5800"))
            "5800" = #Missing;
        ENDIF
    )
ENDFIX
```

字串函式

另請參閱：

- @CalcMgrCompare
- @CalcMgrConcat
- @CalcMgrDecimalFormat
- @CalcMgrDoubleFromString
- @CalcMgrDoubleToString
- @CalcMgrDQuote
- @CalcMgrEndsWith
- @CalcMgrFindFirst
- @CalcMgrFindLast

- @CalcMgrFormatDouble
- @CalcMgrGetListCount
- @CalcMgrGetListItem
- @CalcMgrIndexOf
- @CalcMgrIntegerToString
- @CalcMgrLastIndexOf
- @CalcMgrLowercase
- @CalcMgrMatches
- @CalcMgrMessageFormat
- @CalcMgrPadText
- @CalcMgrUppercase
- @CalcMgrRemoveQuotes
- @CalcMgrRemoveDQuotes
- @CalcMgrRemoveSQuotes
- @CalcMgrReplaceAll
- @CalcMgrReplaceFirst
- @CalcMgrSortAndReturn
- @CalcMgrSortList
- @CalcMgrSortValues
- @CalcMgrSplit
- @CalcMgrSQuote
- @CalcMgrStartsWith
- @CalcMgrStringsToString
- @CalcMgrSubstring
- @CalcMgrTextLength
- @CalcMgrTrim

@CalcMgrCompare

目的：

比較兩個字串

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.StringFunctions.compare(String,String,boolean)

CDF Spec: @CalcMgrCompare(text1, text2,ignoreCase)
```

@CalcMgrConcat

目的：

將指定的字串連接至此字串的尾端

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.StringFunctions.concat(String,String)
```

```
CDF Spec: @CalcMgrConcat(text1, text2)
```

@CalcMgrDecimalFormat

目的：

使用指定的格式字串傳回格式化字串。

對於格式規格，請參閱 Java 文件中的小數格式。

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.StringFunctions.decimalFormat(String,double)
```

```
CDF Spec: @CalcMgrDecimalFormat(formatString, value)
```

@CalcMgrDoubleFromString

目的：

從字串轉換倍精度數值

語法：

```
Java
Class:com.hyperion.calcmgr.common.cdf.StringFunctions.getDoubleFromString(String)
```

```
CDF Spec:@CalcMgrDoubleFromString(text)
```

@CalcMgrDoubleToString

目的：

將倍精度數值轉換為字串

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.MaxLFunctions.doubleToString(double)
```

```
CDF Spec: @CalcMgrDoubleToString(doubleNumber)
```

@CalcMgrDQuote

目的：

將未以雙引號括住的文字以雙引號括住

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.StringFunctions.dQuote(String)

CDF Spec: @CalcMgrDQuote(text)
```

@CalcMgrEndsWith

目的：

測試此字串是否以指定的後綴結束

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.StringFunctions.endsWith(String,String)

CDF Spec: @CalcMgrEndsWith(text, suffix)
```

@CalcMgrFindFirst

目的：

尋找此字串中符合指定一般表示式的第一個子字串

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.StringFunctions.findFirst(String,String,boolean)
CDF Spec: @CalcMgrFindFirst(text, regExpr, ignoreCase)
```

@CalcMgrFindLast

目的：

尋找此字串中符合指定一般表示式的最後一個子字串

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.StringFunctions.findLast(String,String,boolean)
CDF Spec: @CalcMgrFindLast(text, regExpr, ignoreCase)
```

@CalcMgrFormatDouble

目的：

使用指定的格式字串傳回格式化字串。

如需格式規格的資訊，請參閱 Java 文件中的列印格式。

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.StringFunctions.formatDouble(String,double)

CDF Spec: @CalcMgrFormatDouble(formatString, value)
```

@CalcMgrGetListCount

目的：

傳回清單中的項目數目

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.StringFunctions.getListCount(String[])
```

```
CDF Spec: @CalcMgrGetListCount(list)
```

@CalcMgrGetListItem

目的：

從清單中傳回索引項目

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.StringFunctions.getListItem(String[],int)
```

```
CDF Spec: @CalcMgrGetListCount(list,index)
```

@CalcMgrIndexOf

目的：

從指定索引開始，傳回此字串內第一次出現指定子字串的索引

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.StringFunctions.indexOf(String,String,int)
```

```
CDF Spec: @CalcMgrIndexOf(text,searchText,begIndex)
```

> **備註：**
>
> 如果您在 `@CalcMgrIndexOf` 或 `@CalcMgrLastIndexOf` (如下) 中為索引使用 -1，會搜尋整個字串。

@CalcMgrIntegerToString

目的：

將整數轉換為字串

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.MaxLFunctions.integerToString(int)
```

```
CDF Spec: @CalcMgrIntegerToString(integerNumber)
```

@CalcMgrLastIndexOf

目的：

向後搜尋並從指定索引開始，傳回此字串內最後一次出現指定子字串的索引

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.StringFunctions.lastIndexOf(String,String,int)

CDF Spec: @CalcMgrLastIndexOf(text,searchText,begIndex)
```

@CalcMgrLowercase

目的：

將文字轉換成小寫

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.StringFunctions.toLower(String)

CDF Spec: @CalcMgrLowercase(text)
```

@CalcMgrMatches

目的：

如果此字串中的第一個子字串符合指定的一般表示式，則傳回 True。

如需一般表示式的資訊，請參閱 Java 文件中的 "java.util.regex.Pattern"。

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.StringFunctions.matches(String,String,boolean)
CDF Spec: @CalcMgrMatches(text, regExpr, ignoreCase)
```

@CalcMgrMessageFormat

目的：

使用指定的樣式建立字串，然後用來格式化指定的引數。

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.StringFunctions.messageFormat(String,String[])
CDF Spec: @CalcMgrMessageFormat(text, parameters)
```

@CalcMgrPadText

目的：

在文字前後填入填補文字來補足長度。

例如，`@CalcMgrPadText("01",5,"0",@_true)` 會傳回 01000

`@CalcMgrPadText("01",5,"0",@_false)` returns 00001

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.StringFunctions.padText(String,int,String,boolean
)
```

```
CDF Spec: @CalcMgrPadText(text,length,padText,append)
```

@CalcMgrUppercase

目的：

將文字轉換成大寫

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.StringFunctions.toUpper(String)
```

```
CDF Spec: @CalcMgrUppercase(text)
```

@CalcMgrRemoveQuotes

目的：

移除文字字串前後的單引號或雙引號

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.StringFunctions.removeQuotes(String)
```

```
CDF Spec: @CalcMgrRemoveQuotes(text)
```

@CalcMgrRemoveDQuotes

目的：

移除文字字串前後的雙引號

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.StringFunctions.removeDQuotes(String)
```

```
CDF Spec: CDF Spec: @CalcMgrRemoveDQuotes(text)
```

@CalcMgrRemoveSQuotes

目的：

移除文字字串前後的單引號

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.StringFunctions.removeSQuotes(String)
```

```
CDF Spec: @CalcMgrRemoveSQuotes(text)
```

@CalcMgrReplaceAll

目的：

將此字串中與指定的一般表示式相符的每個子字串取代為指定的項目。

如需一般表示式的資訊，請參閱 Java 文件中的 `java.util.regex.Pattern`

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.StringFunctions.replaceAll(String,String,String)
```

```
CDF Spec: @CalcMgrReplaceAll(text, regExpr, replacement)
```

@CalcMgrReplaceFirst

目的：

將此字串中與指定一般表示式相符的第一個子字串取代為指定的項目。

如需一般表示式的資訊，請參閱 Java 文件中的 `java.util.regex.Pattern`

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.StringFunctions.replaceFirst(String,String,String
)
```

```
CDF Spec: @CalcMgrReplaceFirst(text, regExpr, replacement)
```

@CalcMgrSortAndReturn

目的：

根據值排序清單中的項目並傳回前 n 個元素

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.StringFunctions.sortAndReturn(String[],double[],i
nt,boolean)
```

```
CDF Spec: @CalcMgrSortAndReturn(list,values,topN,sortAscending)
```

@CalcMgrSortList

目的：

排序清單中的項目

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.StringFunctions.sortList(String[],boolean,boolean
)
```

```
CDF Spec: @CalcMgrSortList(list,caseSensitive,sortAscending)
```

@CalcMgrSortValues

目的：

根據值排序清單中的項目

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.StringFunctions.sortValues(double[],boolean)
```

```
CDF Spec: @CalcMgrSortValues(values,sortAscending)
```

@CalcMgrSplit

目的：

根據 regex 分割文字

如需一般表示式的資訊，請參閱 Java 文件中的 java.util.regex.Pattern

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.StringFunctions.split(String,String)
```

```
CDF Spec: @CalcMgrSplit(text, regex)
```

@CalcMgrSQuote

目的：

將未以單引號括住的文字以單引號括住

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.StringFunctions.sQuote(String)
```

```
CDF Spec: @CalcMgrSQuote(text)
```

@CalcMgrStartsWith

目的：

測試此字串是否以指定的前綴開頭

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.StringFunctions.startsWith(String,String)
```

```
CDF Spec: @CalcMgrStartsWith(text, prefix)
```

@CalcMgrStringsToString

目的：

將字串陣列轉換為使用分隔符號的字串

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.MaxLFunctions.stringsToString(String[],String)
```

```
CDF Spec: @CalcMgrStringsToString(strings, seperator)
```

@CalcMgrSubstring

目的：

傳回由此字串之子字串構成的新字串。

子字串從 startIndex 處開始並延伸至位於 index endIndex - 1 處的字元。因此子字串的長度為 endIndex-startIndex。

如果 endIndex 小於零，則 endIndex 將會是最後一個字元的索引。

語法：

```
Java Class:
com.hyperion.calcmgr.common.cdf.StringFunctions.substring(String,int,int)

CDF Spec: @CalcMgrSubstring(text, startIndex, endIndex)
```

@CalcMgrTextLength

目的：

傳回文字的長度

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.StringFunctions.length(String)

CDF Spec: @CalcMgrTextLength(text)
```

@CalcMgrTrim

目的：

移除文字字串前端和尾端的空格

語法：

```
Java Class: com.hyperion.calcmgr.common.cdf.StringFunctions.trim(String)

CDF Spec: @CalcMgrTrim(text)
```

使用智慧型列示

另請參閱：

- 關於智慧型列示
  「智慧型列示」是自訂下拉清單，使用者可從規劃應用程式的規劃資料表單儲存格來存取。
- 插入智慧型列示
  使用規劃商業規則中的「智慧型列示」，也可以使用公式元件或指令碼元件。

關於智慧型列示

「智慧型列示」是自訂下拉清單，使用者可從規劃應用程式的規劃資料表單儲存格來存取。

按一下資料表單儲存格以輸入資料，您可從下拉清單選取項目，而不必在儲存格中進行輸入。您無法在包含「智慧型列示」的儲存格中鍵入資料。

在 Calculation Manager 中，您可將「智慧型列示」插入變數或公式、指令碼、條件及成員範圍元件。

插入智慧型列示

使用規劃商業規則中的「智慧型列示」，也可以使用公式元件或指令碼元件。

在規劃管理員指定的特定資料儲存格內，您可在這些儲存格的規劃資料表單上取得「智慧型列示」。「智慧型列示」是一種自訂的下拉清單，使用者可選取清單中包含的選項。

若要插入「智慧型列示」，請執行下列動作：

1. 執行下列其中一項任務：
   - 開啟包含要向其插入「智慧型列示」之元件的商業規則。然後在商業規則的流程圖中，選取要向其插入「智慧型列示」的公式或指令碼元件。
   - 開啟您要向其插入「智慧型列示」的公式元件。
   - 開啟您要向其插入「智慧型列示」的指令碼元件。
2. 執行下列其中一項任務：
   - 若要將「智慧型列示」插入商業規則中，請在**指令碼**頁籤上，按一下**插入智慧型列示**圖示。
   - 若要將「智慧型列示」插入公式元件中，請在**公式**頁籤上，按一下**公式**列，選取**動作**圖示，然後選取**智慧型列示**。
   - 若要將「智慧型列示」插入指令碼元件中，請在**指令碼**頁籤上，按一下**插入智慧型列示**圖示。
3. 按一下**儲存**。

使用 Planning 公式表示式

在 Calculation Manager 圖形化或指令碼規則中使用規劃公式表示式。

您可以使用以下類型的公式表示式：

- 智慧型列示
- 維度
- 規劃使用者變數
- 期間
- 案例
- 交互參照
- 人力立方體年累計
- 獲取字串 ID

智慧型列示

您可以將「智慧型列示」作為變數包括在公式表示式中，例如公式表示式 `"Product Channel"=[[Channel.Retail]]`。

`"Product Channel"` 是「智慧型列示」類型的科目，`Channel` 是「智慧型列示」名稱，而 `Retail` 是「智慧型列示」項目。如果 Retail 的「智慧型列示 ID」為 2，`Channel.Retail` 在成員公式中會被取代為 2 (應用程式將「智慧型列示」視為數字)。如果 Retail 的「智慧型列示 ID」為 2，2 會被放入計算中，並儲存在資料庫中。

Calculation Manager 語法：

```
[[SLName.entryname]]
```

範例：

下列語法：

```
FIX (Mar, Actual, Working, FY15, P_000, "111")
   "Product Channel" =[[Channel.Retail]] ;
ENDFIX
```

會傳回下列指令碼：

```
FIX (Mar, Actual, Working, FY15, P_000, "111")
   "Product Channel" =2 ;
ENDFIX
```

維度

`Dimension(dimTag)` 傳回預先定義的維度名稱。

dimtag 為：

- DIM_NAME_PERIOD
- DIM_NAME_YEAR
- DIM_NAME_ACCOUNT
- DIM_NAME_ENTITY
- DIM_NAME_SCENARIO
- DIM_NAME_VERSION
- DIM_NAME_CURRENCY

Calculation Manager 語法：

```
[[Dimension("DIM_NAME_ENTITY")]]
```

範例：

```
CALC DIM([[Dimension("DIM_NAME_ENTITY")]]);
```

在此應用程式中，`Entity` 命名為 *Entity*，所以上述指令碼會傳回：

```
CALC DIM ("Entity");
```

如果 Entity 維度命名為 *Cost Center*，則會傳回：

```
CALC DIM ("Cost Center");
```

規劃使用者變數

規劃使用者變數會傳回使用者變數的成員。

Calculation Manager 語法：

```
[[PlanningFunctions.getUserVarValue("xyz")]]
```

範例：

```
FIX (Feb, Actual, Working, P_000, [[PlanningFunctions.getUserVarValue("Entity
View")]] )
   "5800" = 40;
ENDFIX
```

在此應用程式中，有一個名稱為 *Entity View* 的規劃使用者變數。對於此使用者，該變數會設為 112。所以，上述範例的指令碼會傳回：

```
FIX (Feb, Actual, Working, P_000, "112")
   "5800" = 40;
ENDFIX
```

期間

另請參閱：

- Period(periodName)
- NumberofPeriodsinYear 與 NumberofYears

Period(periodName)

`Period(periodName)` 傳回指定的期間。

期間名稱選項為：

- FIRST_QTR_PERIOD
- SECOND_QTR_PERIOD
- THIRD_QTR_PERIOD
- FOURTH_QTR_PERIOD
- FIRST_PERIOD
- LAST_PERIOD

Calculation Manager 語法：

```
[[Period("FIRST_QTR_PERIOD")]]
```

範例：

下列語法：

```
FIX ( Mar, Actual, Working, P_000, "6100", FY15 )
   "120" =[[Period("FIRST_QTR_PERIOD")]];
ENDFIX
```

會傳回下列指令碼：

```
FIX (Mar, Actual, Working, P_000, "6100", FY15)
   "120" = "Mar";
ENDFIX
```

NumberofPeriodsinYear 與 NumberofYears

`NumberofPeriodsInYear` 傳回年度中的期間數，而 `NumberofYears` 會在應用程式中傳回年數。

Calculation Manager 語法：

```
[[NumberOfPeriodsInYear]]
```

```
[[NumberOfYears]]
```

範例：

下列語法：

```
FIX (Mar, Actual, Working, P_000, "6100", FY15)
   "120"=[[NumberOfPeriodsInYear]];
   "120"=[[NumberOfYears]];
ENDFIX
```

會傳回下列指令碼：

```
FIX (Mar, Actual, Working, P_000, "6100", FY15)
   "120"=12;
   "120"=9;
ENDFIX
```

案例

以表示式來顯示「案例規劃範圍」的時間範圍和「模組規劃與預測準備」組態任務資訊，讓您能指定下列項目：

- **開始年度**：以字串格式傳回指定案例的開始年度。
- **結束年度**：以字串格式傳回指定案例的結束年度。
- **開始月份**：以字串格式傳回指定案例的開始月份。
- **結束月份**：以字串格式傳回指定案例的結束月份。
- **模組開始年度**：以字串格式傳回指定模組和案例的開始年度。
- **模組結束年度**：以字串格式傳回指定模組和案例的結束年度。
- **模組開始期間**：以字串格式傳回指定模組和案例的開始期間。
- **模組結束期間**：以字串格式傳回指定模組和案例的結束期間。
- **模組計畫開始年度**：如果是*目前會計年度*，會傳回整數值 "1"；如果是*下一個會計年度*，則會傳回 "0"。

備註：

「模組開始年度」、「模組結束年度」、「模組開始期間」、「模組結束期間」、及「模組計畫開始年度」只對 Planning 模組和 Strategic Workforce Planning 應用程式有效，且需要使用 "PlanningFunctions" 語法。

Calculation Manager 語法：

```
[[getStartYear("ScenarioName")]]
[[getEndYear("ScenarioName")]]
[[getStartMonth("ScenarioName")]]
[[getEndMonth("ScenarioName")]]
[[PlanningFunctions.getModuleStartYear("ModuleName","ScenarioName")]]
[[PlanningFunctions.getModuleEndYear("ModuleName","ScenarioName")]]
[[PlanningFunctions.getModuleStartPeriod("ModuleName","ScenarioName")]]
[[PlanningFunctions.getModuleEndPeriod("ModuleName","ScenarioName")]]
[[PlanningFunctions.isPlanStartYearSameAsCurrentFiscalYear("ModuleName")]]
```

案例名稱

可以對 `ScenarioName` 輸入「案例」成員，或者 Calculation Manager 執行階段提示成員類型變數。成員必須以雙引號括住。例如，`[[getStartYear("Actual")]]`。

`ScenarioName` 也可以是下列格式中的替代變數：

- `getSubVarValue("CubeName", "SubstitutionVariableName")` 會傳回處於由 `CubeName` 定義的立方體層級之指定替代變數的替代變數值
- `getSubVarValue("SubstitutionVariableName")` 會傳回處於所有立方體的應用程式層級之指定替代變數的替代變數值

您必須要用雙引號括住 `CubeName` 和 `SubstitutionVariableName`，請勿使用 & 或 {}。請參閱下方的範例 3 和範例 4。

模組名稱

`ModuleName` 必須是 Planning 和 Strategic Workforce Planning 中某個目前「模組」的預先定義名稱。這些使用 `ModuleName` 來作為參數的表示式，會使用 `ScenarioName` 的「規劃與預測準備」組態任務時間範圍資訊，而這可能會與「規劃與預設案例」之應用程式中的時間範圍不同。

您也可以在這些表示式中參照「規劃與預測」之外的其他「案例」，但表示式只會傳回您在維度編輯器中為「案例規劃範圍」設定的時間範圍，而這將會與應用程式中所有「模組」的時間範圍相同。如需「案例規劃範圍」的詳細資訊，請參閱關於案例。

「計畫開始年度」是「模組」之「計畫案例」專有的另一個「規劃與預設準備」組態任務。此表示式會傳回與其設定相關的值，像是會在設定為*目前會計年度*整數值 "1"，並在設定為*下一個會計年度*時傳回 "0"。此表示式只能在條件陳述式的環境定義中使用，因為它不會把維度成員當作值來傳回。請參閱下方的範例 5。

下列為 `ModuleName` 的有效值，而它們適用於特定的商業程序：

表格 9-4　有效的 ModuleName 值

商業程序	有效的 ModuleName 值
規劃模組	· "Capital" · "Financials" · "Projects" · "Workforce"
Strategic Workforce Planning	· "Strategic Workforce" · "Workforce"

> **備註：**
>
> ModuleName 值必須要用雙引號括住，且不區分大小寫。

範例 1

下列語法中，{rtpScenario} 是*成員* 類型且預設值為*實際* 的執行階段提示變數：

```
FIX({rtpScenario}, [[getStartYear({rtpScenario})]]:
[[getEndYear({rtpScenario})]],
   [[getStartMonth({rtpScenario})]]:[[getEndMonth({rtpScenario})]])
      FIX ( Working, P_000, "111")
        "5800" = 5500;
      ENDFIX
ENDFIX
```

會傳回下列指令碼：

```
FIX ("Actual", "FY10" : "FY18", "Jan" : "Dec")
    FIX (Working, P_000, "111")
      "5800" = 5500;
    ENDFIX
ENDFIX
```

範例 2

下列語法：

```
FIX({rtpScenario}, [[PlanningFunctions.getModuleStartYear("CAPITAL",
{rtpScenario})]]:
[[PlanningFunctions.getModuleEndYear("CAPITAL",{rtpScenario})]] , "Jan" :
"Dec")
  FIX(OEP_Working, P_000, "111")
    "5800" = 5500;
  ENDFIX
ENDFIX
```

會傳回下列指令碼，其中 Capital 模組「規劃與預測準備」組態任務分別將「規劃案例」的開始和結束年度設定為 FY18 及 FY22：

```
FIX("OEP_Plan", "FY18" : "FY22", "Jan" : "Dec")
  FIX(OEP_Working, P_000, "111")
    "5800" = 5500;
  ENDFIX
ENDFIX
```

範例 3

下列語法在立方體層級使用替代變數。在這個範例中，`Plan1` 是立方體名稱，而 `CurrentMonth` 是替代變數名稱。

```
FIX ("OEP_Plan", [[getStartMonth(getSubVarValue("Plan1", "CurrentMonth"))]],
"FY15",
"BU Version_1", "No Currency", "No Entity", "No Grades")
"Current" = 15;
ENDFIX
```

這個程序會產生下列指令碼：

```
FIX ("OEP_Plan", "Jan", "FY15", "BU Version_1", "No Currency", "No Entity",
"No
Grades")
"Current" = 15;
ENDFIX
```

範例 4

下列語法在應用程式層級使用替代變數。在這個範例中，`CurrentMonth` 是替代變數名稱。

```
FIX ("OEP_Plan", [[getStartMonth(getSubVarValue("CurrentMonth"))]], "FY15",
"BU
Version_1", "No Currency", "No Entity", "No Grades")
"Current" = 15;
ENDFIX
```

這個程序會產生下列指令碼：

```
FIX ("OEP_Plan", "Jan", "FY15", "BU Version_1", "No Currency", "No Entity",
"No
Grades")
"Current" = 15;
ENDFIX
```

範例 5

下列語法：

```
FIX("OEP_Plan","OEP_Working",FY20:FY24)
     "Bonus"
     (
```

```
     IF([[PlanningFunctions.isPlanStartYearSameAsCurrentFiscalYear
("Workspace")]] == 1)
          "Bonus" = "Salary" * 0.2;
     ELSEIF([[PlanningFunctions.isPlanStartYearSameAsCurrentFiscalYear
("Workforce")]] == 0)
          "Bonus" = "Salary" * 0.3;
     )
ENDFIX
```

會產生下列指令碼，其中 Workforce 模組「規劃與預測準備」組態任務已將「計畫開始年度」設定為 *下一個會計年度*，因此第一個條件測試 (IF) 失敗且沒有執行，而第二個條件測試 (ELSEIF) 通過並執行，因為 `isPlanStartYearSameAsCurrentFiscalYear` 在設定為 *下一個會計年度* 時會傳回整數 "0"：

```
FIX("OEP_Plan","OEP_Working", FY20:FY24)
     "Bonus"
     (
     IF(0 == 1)
          "Bonus" = "Salary" * 0.2;
     ELSEIF(0 == 0)
          "Bonus" = "Salary" * 0.3;
     )
ENDFIX
```

交互參照

另請參閱：

- CrossRef(accountName)
- CrossRef(accountName, prefix)
- CrossRef(accountName, prefix, true)

CrossRef(accountName)

`CrossRef(accountName)` 會藉由將預設前綴新增至每個維度名稱 (除了 Currency、Period 與 Year 以外)，接著是特定科目，以產生交互參照。

Calculation Manager 語法：

```
[[CrossRef(accountName)]]
```

範例：

假設應用程式具有下列維度：Account、Period、HSP_View、Year、Scenario、Version、Entity，以及 Product。在此範例中，下列語法：

```
FIX (Aug, Actual, Working, FY15, P_000, "112")
   "120" = [[CrossRef("5800")]];
ENDFIX
```

會傳回下列指令碼：

```
FIX (Aug, Actual, Working, FY15, P_000, "112")
   "120" = "BegBalance"->"No HSP_View"->"No Scenario"->"No Version"->"No
Entity"->"No Product"->"5800";
ENDFIX
```

CrossRef(accountName, prefix)

`CrossRef(accountName, prefix)` 會藉由將特定前綴新增至每個維度名稱 (除了 Currency、Period 與 Year 以外)，接著是特定科目，以產生交互參照。前綴必須以雙引號括住。

Calculation Manager 語法：

```
[[CrossRef(accountName, "prefix")]]
```

範例：

假設應用程式具有下列維度：Account、Period、HSP_View、Year、Scenario、Version、Entity，以及 Product。在此範例中，下列語法：

```
FIX (Aug, Actual, Working, FY15, P_000, "112")
   "110" = [[CrossRef("5800", "No")]];
ENDFIX
```

會傳回下列指令碼：

```
FIX (Aug, Actual, Working, FY15, P_000, "112")
   "110" = "BegBalance"->"No HSP_View"->"No Scenario"->"No Version"->"No
Entity"->"No Product"->"5800";
ENDFIX
```

CrossRef(accountName, prefix, true)

`CrossRef(accountName, prefix, true)` 會藉由將特定前綴新增至每個維度名稱，接著是特定科目，以產生交互參照。(此包括 Year，但不包括 Currency 與 Period。) 前綴必須以雙引號括住。

Calculation Manager 語法：

```
[[CrossRef(accountName, "prefix", true)]]
```

範例：

假設應用程式具有下列維度：Account、Period、HSP_View、Year、Scenario、Version、Entity，以及 Product。在此範例中，下列語法：

```
FIX (Aug, Actual, Working, FY15, P_000, "112")
   "111" = [[CrossRef("5800", "NoX", true)]];
ENDFIX
```

會傳回下列指令碼：

```
FIX (Aug, Actual, Working, FY15, P_000, "112")
   "111" = "BegBalance"->"NoXHSP_View"->"NoXYear"->"NoXScenario"-
>"NoXVersion"->"NoXEntity"->"NoXProduct"->"5800";
ENDFIX
```

人力立方體年累計

另請參閱：

- CYTD(memberName)
- CYTD(memberName, calTpIndexName, fiscalTPIndexName)

CYTD(memberName)

> **備註：**
>
> `CYTD(memberName)` 僅適用於人力立方體。

`CYTD(memberName)` 會產生成員的工作曆年累計公式

Calculation Manager 語法：

```
[[CYTD(memberName)]]
```

範例：

```
Fix (NOV, Actual, Working, FY15, P_000, "112")
   "5800" = [[CYTD("6100")]];
ENDFIX
```

CYTD(memberName, calTpIndexName, fiscalTPIndexName)

> **備註：**
>
> `CYTD(memberName, calTpIndexName, fiscalTPIndexName)` 僅適用於人力立方體。

`CYTD(memberName, calTpIndexName, fiscalTPIndexName)` 會產生成員的工作曆年累計公式，以及以工作曆年與會計年度為基礎的期間索引。重新命名成員時使用。預設成員名稱為 "Cal TP-Index" 與 "Fiscal TPIndex” 。

Calculation Manager 語法：

```
[[CYTD(accountName, "Cal TP-Index", "Fiscal TPIndex")]]
```

範例：

```
Fix (Dec, Actual, Working, FY15, P_000, "112")
   "5800" = [[CYTD("6100", "Cal TP-Index", "Fiscal TPIndex")]];
ENDFIX
```

獲取字串 ID

在規劃中，當科目類型是*文字*時，您可以在 Calculation Manager 中撰寫公式來指派文字值。

Calculation Manager 語法：

```
[[PlanningFunctions.getIdForString("text")]]
```

範例：

在 Planning 中，您擁有名稱為 "acct1 text"，且類型為*文字*的科目。您希望將值從 FY16 Dec 複製到 FY17 Mar，並將文字科目變更為「未編列預算」。

```
FIX (Actual, Working, P_000, "210")
   DATACOPY FY16->Dec TO FY17->Mar;
   Mar("acct1 text"->FY17 = [[PlanningFunctions.getIdForString("Not
Budgeted")]];)
ENDFIX
```

搭配 Essbase 中的「混合彙總」使用

區塊儲存資料庫的混合彙總意味著在可能的情況下，區塊儲存資料計算執行的效率類似於區塊儲存資料庫的計算執行效率。

- 混合彙總中的動態計算
- 「混合彙總」不支援計算命令
- 「混合彙總」不支援函式

> **備註：**
>
> 如需有關 Oracle Essbase 中混合彙總的詳細資訊，請參閱 *Technical Reference for Oracle Analytics Cloud - Essbase* 指南。

混合彙總中的動態計算

Oracle Essbase 混合計算器評估動態成員公式的方式與傳統 Essbase 計算器不同。

- 傳統 Essbase 計算器會針對所有交集傳回動態成員公式的值。跨交集產生常數值的公式只會在指定的交集有區塊時，才會傳回值。
- Essbase 混合計算器已最佳化為可透過忽略常數來改善擷取的效能。這是因為混合計算器會處理擷取自稀疏維度的動態彙總，這是傳統 BSO 所無法做到的。

例如，在下列公式中：

Solve Order 0

```
IF(@ISMBR("Jan"))
1;
ELSEIF(@ISMBR("Feb"))
2;
ELSEIF(@ISMBR("Mar"))
3;
ELSEIF(@ISMBR("Apr"))
4;
ELSEIF(@ISMBR("May"))
5;
ELSEIF(@ISMBR("Jun"))
6;
ELSEIF(@ISMBR("Jul"))
7;
ELSEIF(@ISMBR("Aug"))
8;
ELSEIF(@ISMBR("Sep"))
9;
```

下列顯示沒有資料的交集：

	Forecast											
	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
Cal_TP_Index												
NOACCOUNT												

當您新增資料時，值會以下列方式顯示：

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
Cal_TP_Index	1	2	3	4	5	6	7	8	9	10	11	12
NOACCOUNT	555	555	66	77	88	99	99	99	99	99	99	99

Note:

只要有區塊，就會使用混合計算器評估動態公式。

「混合彙總」不支援計算命令

「混合彙總」不支援下列計算命令。如果系統在規則中遇到下列任何一個命令，驗證就會失敗，該規則也不會啟動。

- CALC ALL
- CCONV
- CLEARCCTRACK

- SET CACHE
- SET CCTRACKCALC
- SET CLEARUPDATESTATUS
- SET DATAIMPORTIGNORETIMESTAMP
- SET LOCKBLOCK
- SET MSG
- SET NOTICE
- SET REMOTECALC
- SET RUNTIMESUBVARS
- SET UPTOLOCAL

「混合彙總」不支援函式

「混合彙總」不支援下列函式。如果遇到這些函式，Oracle Essbase 預設為區塊儲存執行。

- @ALLOCATE
- @CREATEBLOCK
- @IRREX
- @MDALLOCATE
- @MDSHIFT
- @MOVSUMX
- @PTD
- @SANCESTVAL
- @STDEV
- @STDEVP
- @@STDEVRANGE
- @SYD
- @TREND
- @XWRITE

10

驗證與部署

另請參閱：

- 從系統檢視驗證商業規則、商業規則集以及公式元件和指令碼元件
  將商業規則、規則集、公式和指令碼元件部署至應用程式之前，可以先加以驗證，確定其語法是正確的。
- 從規則設計工具驗證商業規則
  當您建立或偵錯商業規則時，可以在規則設計工具中驗證規則。
- 部署商業規則和商業規則集
  部署商業規則和商業規則集。

從系統檢視驗證商業規則、商業規則集以及公式元件和指令碼元件

將商業規則、規則集、公式和指令碼元件部署至應用程式之前，可以先加以驗證，確定其語法是正確的。

驗證程序能確保：

- 所有維度成員對應用程式所含的維度有效。
- 所有函式都存在，都具有正確數量的參數，並對應用程式類型有效。
- 商業規則中的所有變數參照均有效。對於取代變數，首先用正確的字串對變數進行取代，然後進行驗證。對於執行變數，驗證程序可確保針對應用程式、應用程式類型內的應用程式、計畫類型，和/或商業規則定義變數。
- 指令碼產生期間無語法錯誤。

如果您正在驗證的商業規則的執行時期提示採用預設值，則驗證程序可確保執行時期提示中的所有成員對所選計畫類型和應用程式均有效，且無語法或語義錯誤。如果您要驗證的商業規則的執行時期提示*未*採用預設值，則不會執行驗證。

> **備註：**
>
> 如果部署之前未驗證規則和規則集，部署可能會成功，但是規則和規則集可能會無法啟動。

若要驗證商業規則、規則集或公式或指令碼元件，請執行下列動作：

1. 執行下列其中一項任務：
   - 若要驗證規則集，請展開**規則集**。

> **備註：**
>
> 對於 Planning 應用程式，每個位於與計畫類型和資料庫相同層級的應用程式都只有一個「規則集」節點。

- 若要驗證規則、公式、指令碼或範本，請展開計算類型、計畫類型或資料庫，以及**規則**、**公式**、**指令碼**或**範本**，取決於您要驗證的物件而定。

2. 執行下列其中一項任務：
   - 用滑鼠右鍵按一下要驗證的物件，然後選取**驗證**。
   - 選取要驗證的物件，然後選取**動作**，**驗證**。
3. 執行下列其中一項任務：
   a. 如果物件驗證成功，按一下**確定**。
   b. 如果出現錯誤，將會顯示這些錯誤。請修復錯誤，然後再次驗證該物件。

從規則設計工具驗證商業規則

當您建立或偵錯商業規則時，可以在規則設計工具中驗證規則。

若要從「規則設計工具」中驗證商業規則，請執行下列動作：

1. 在「系統檢視」中，依序展開應用程式類型、應用程式、計算類型、計畫類型或資料庫，以及**規則**。
2. 用滑鼠右鍵按一下想要驗證的規則，然後選取**開啟**。

部署商業規則和商業規則集

部署商業規則和商業規則集。

另請參閱：

- 關於部署商業規則和商業規則集
- 將商業規則和商業規則集設為可部署和不可部署
- 從部署檢視部署商業規則和商業規則集
- 從規則或規則集設計工具部署商業規則或商業規則集
- 部署商業規則 (含捷徑)
- 指定哪個已部署的商業規則會顯示在 Planning 中。

關於部署商業規則和商業規則集

您可以將商業規則和商業規則集部署到 Planning。您可以部署一個或多個商業規則或商業規則集 (即局部部署)，也可以部署應用程式中所有商業規則和規則集 (即完全部署)。

> **備註：**
>
> 若您的 Planning 商業規則包含存在於多個層級 (即存在於下列一個以上層級的變數：全域、應用程式、計畫類型或規則) 的變數，而您在最低層級刪除該變數，則您必須重新部署 Planning 應用程式，才能對規劃中使用該變數的所有規則進行刪除。若您僅執行部分重新部署，可能不會刪除變數，該變數仍會用於規劃中。

將商業規則和商業規則集部署到 Planning 之後，可以在資料表單內啟動它們，或者從「啟動」功能表獨立啟動。

有關在 Planning 中啟動商業規則和商業規則集的詳細資訊，請參閱*使用規劃*。

將商業規則和商業規則集設為可部署和不可部署

若要部署應用程式中的商業規則和商業規則集的子集，則必須將它們設定為可部署。若要將規則和規則集設為可部署，請在**部署檢視**中選取規則和規則集名稱旁的核取方塊。

> **備註：**
>
> 若只要部署一個商業規則或商業規則集，則無需在**部署檢視**中將它們設為可部署。不過，您可以藉由在規則或規則集上按一下滑鼠右鍵，並選取**部署**，以從**系統檢視**中進行部署。

若要在部署商業規則或規則集之後，將其從應用程式中移除，請在**部署檢視**中將其名稱旁的核取方塊清除。然後您可對其按一下滑鼠右鍵並選取**部署**，以執行應用程式的完整部署。

若要將商業規則和商業規則集設為可部署，請執行下列動作：

1. 在**部署檢視**中，展開應用程式類型以及含有要部署之規則或規則集的應用程式。
2. 展開**需要部署**，然後選取要部署之規則與規則集旁的核取方塊。

   在部署規則或規則集之前，您應該先驗證其語法是否正確。您可以使用「驗證」功能手動驗證規則和規則集。(請參閱從系統檢視驗證商業規則、商業規則集以及公式元件和指令碼元件。

從部署檢視部署商業規則和商業規則集

您可以從「部署檢視」部署商業規則和商業規則集。您也可以從「規則設計工具」(適用於商業規則) 或「規則集設計工具」(適用於商業規則集)，將一個商業規則或一個商業規則集部署至 Planning。請參閱從規則或規則集設計工具部署商業規則或商業規則集。

若要從「部署檢視」部署商業規則和商業規則集，請執行下列動作：

1. 在**系統檢視**中，選取**檢視**，然後選取**部署檢視**。

> **備註：**
>
> 另一個部署方式，是從「系統檢視」以滑鼠右鍵按一下商業規則和商業規則集，然後選取「部署」。

2. 在「部署檢視」中，展開應用程式類型。
3. 執行下列其中一項任務：
   - 若要部署應用程式內的*所有*規則和規則集，請選取您要部署的所有規則或規則集，用滑鼠右鍵按一下該應用程式，然後選取「部署」。
   - 若要部署商業規則和商業規則集的*子集* (又稱為部分部署)，請展開應用程式以及**需要部署**節點。然後執行下列步驟：
     a. 如果您要部署的規則集未選取，請選取這些規則集。
     b. 展開包含您要部署的規則的計畫類型。
     c. 如果您要部署的規則未選取，請選取這些規則。
     d. 用滑鼠右鍵按一下，然後選取**部署**。

     > **提示：**
     >
     > 若要部署多個規則或規則集，請使用「Ctrl + 按一下」和「Shift + 按一下」選取它們，然後用滑鼠右鍵按一下，並選取「部署」。

   如果部署成功，系統會顯示「部署成功」訊息。

從規則或規則集設計工具部署商業規則或商業規則集

設計完商業規則或規則集後，您可以在「規則設計工具」或「規則集設計工具」內部直接對其進行驗證和部署。

若要從「規則設計工具」或「規則集設計工具」部署商業規則或商業規則集，請執行下列動作：

1. 執行下列其中一項任務：
   - 若要部署商業規則，請展開包含該規則的計畫類型或資料庫，然後展開**規則**。
   - 若要部署商業規則集，請展開**規則集**。
2. 用滑鼠右鍵按一下您要部署的規則或規則集，然後選取**開啟**。
3. 從**規則設計工具**或**規則集設計工具**中，選取**動作**，然後選取**部署**。

   如果部署成功，系統會顯示「部署成功」訊息。

部署商業規則 (含捷徑)

如果商業規則帶有捷徑，則在將該商業規則部署到應用程式時，系統會將該規則的複本部署到已為其建立捷徑的各個應用程式中。

若要部署帶有捷徑的商業規則，請執行下列動作：

1. 在**系統檢視**中，選取**檢視**，然後選取**部署檢視**。
2. 展開應用程式類型、應用程式、**待部署**節點，然後計畫類型或資料庫。

3. 用滑鼠右鍵按一下您要部署的規則，然後選取**全部部署**。

指定哪個已部署的商業規則會顯示在 Planning 中

在 Calculation Manager 中部署商業規則後，就可以在 Planning 中的「商業規則」頁面檢視和執行。

您可以指定哪個商業規則要顯示在 Planning 中。若要這麼做，請執行下列動作：

1. 從**部署檢視**中，取消選取您不希望顯示在 Planning 中的規則。
2. 用滑鼠右鍵按一下應用程式節點，然後選取**部署**。

   如果規則先前顯示在 Planning 中，但已在 Calculation Manager 的「部署檢視」中取消勾選，則在部署應用程式後，規則就不會再顯示在 Planning 中。

11

啟動商業規則

另請參閱：

- 關於啟動商業規則
  您可以從「系統檢視」或 Calculation Manager 中的「規則設計工具」來啟動規劃商業規則。
- 從規則設計工具啟動 Planning 商業規則和檢視日誌
  您可以啟動規劃商業規則以及檢視所產生的日誌。

關於啟動商業規則

您可以從「系統檢視」或 Calculation Manager 中的「規則設計工具」來啟動規劃商業規則。

您也可以將 Planning 商業規則部署至 規劃，然後從規劃啟動。如需啟動 Planning 中 Planning 商業規則的相關資訊，請參閱*使用規劃*。

備註：

- 當進行驗證、偵錯、部署、分析以及啟動 Calculation Manager 中的規則時，您可以輸入或編輯執行時期提示變數的值。您也可以在驗證或部署商業規則集時輸入或編輯執行時期提示變數。如果執行時期提示包含成員限制，驗證會僅檢查有效的成員名稱 (不會驗證成員是否在限制範圍內)。執行時期提示限制會驗證*數值*和*整數*類型變數。
- 啟動 Calculation Manager 中的規則時未遵循有效的組合和安全性。
- 如需設計遵循成員核准安全性的執行時期提示相關資訊，請參閱 *管理 Planning* 中的關於執行時期提示及核准安全性。

從規則設計工具啟動 Planning 商業規則和檢視日誌

您可以啟動規劃商業規則以及檢視所產生的日誌。

當您開啟商業規則以便在「規則設計工具」中進行檢視或編輯時，您可以啟動該規則以及在「規則設計工具」的「日誌訊息」頁籤中檢視任何產生的日誌。您可以將日誌匯出至逗點分隔值 (.csv) 檔案。

若要從規則設計工具啟動規劃商業規則和檢視日誌，請執行下列動作：

1. 在**系統檢視**中，按兩下規則。
2. 在「規則設計工具」中，選取**動作**，然後選取**啟動**。

   規則執行之後，將會顯示確認訊息，指示啟動是否發生錯誤。按一下**確定**關閉確認訊息。
3. 選取**日誌訊息**頁籤。

   日誌訊息包含下列資訊：

- **訊息號碼** — 日誌檔中顯示的訊息 ID
- **訊息層級** — 訊息的嚴重性/層級
- **訊息文字** — 完整的訊息文字
- **訊息時間戳記** — 表示產生訊息的時間戳記
- **執行次數** — 目前執行的次數。最高/最後次數為規則中的執行次數。
- **執行時間** — 執行時間 (秒)。此為目前執行所花費的時間。
- **累計時間** — 總執行時間 (秒)：此為自規則執行開始後總共經過的時間。

  備註：由於資料庫同一時間產生許多日誌訊息，所以「執行時間」(秒) 僅將每次執行的時間顯示為該執行所花費的時間，而「累計時間」(秒) 顯示自規則開始後所花費的時間。所有其他列顯示為空白。
- **執行百分比** — 執行規則的總時間百分比。
- **累計百分比** — 所有規則執行的總時間百分比。當所有執行完成時，累計百分比應為 100%。

4. **選擇性。**篩選顯示的日誌訊息。

   您可以使用下列條件進行篩選：

   - **訊息號碼** — 請從**訊息號碼**欄上的下拉清單選取值。
   - **訊息層級** — 請從**訊息層級**欄上的下拉清單選取值。
   - **訊息文字** — 請在**訊息文字**欄上的文字方塊中輸入文字。
   - **僅執行** — 請從**執行次數**欄上的下拉清單中選取**僅執行**，只查看執行次數及每次執行花費的時間。若要回到顯示完整日誌資訊，請選取**執行次數**欄上的空白選項。

> **備註：**
>
> 將商業規則部署至 Planning 後，您也可以在 Planning 中啟動它們。如需相關資訊，請參閱*使用 Planning*。

> **備註：**
>
> 您可以將日誌訊息匯出至逗點分隔值 (.csv) 檔案。請參閱將日誌訊息匯出至檔案。

12

匯出和匯入商業規則、商業規則集、範本、公式和指令碼元件

另請參閱：

- 關於匯出和匯入
  匯出及匯入應用程式中的物件。
- 匯出商業規則、商業規則集、範本、公式元件和指令碼元件
  匯出應用程式、一個物件或多個物件時，會將它們匯出到一個 xml 檔案，該檔案可以匯入其他 Calculation Manager 應用程式。
- 匯出應用程式
  當您匯出應用程式時，應用程式內容會儲存為 xml 檔案。
- 將日誌訊息匯出至檔案
  當您從 Calculation Manager 內啟動 Planning 商業規則時，將會產生日誌訊息並顯示在「規則設計工具」內的「日誌訊息」頁籤中。
- 匯出規則、規則集、範本、公式與指令碼
  使用 Calculation Manager 將規則、規則集、範本、公式以及指令碼匯入您的應用程式。

關於匯出和匯入

匯出及匯入應用程式中的物件。

您可以匯出 Planning 應用程式中的所有物件；您也可以匯出應用程式中的個別商業規則、商業規則集、範本、公式和命令碼元件。

您也可以從 Oracle Hyperion Business Rules 匯出商業規則、序列、巨集及變數，並將其匯入至 Calculation Manager。序列已轉換為商業規則集，而巨集則轉換為 Calculation Manager 中的範本。

匯出應用程式和物件之後，您可以將它們匯入至其他規劃應用程式。例如，您可能想從生產電腦上的應用程式中將商業規則與商業規則集匯出，然後匯入至測試電腦上的另一應用程式中。

匯出商業規則、商業規則集、範本、公式元件和指令碼元件

匯出應用程式、一個物件或多個物件時，會將它們匯出到一個 xml 檔案，該檔案可以匯入其他 Calculation Manager 應用程式。

> **備註：**
>
> 您可以將物件從任何檢視中匯出，其中包括「系統檢視」、「自訂檢視」和「部署檢視」。您可以匯出一或多個物件。

若要匯出物件，請執行下列動作：

1. 執行下列其中一項任務：
   - 若要匯出規則集，請展開**規則集**。
   - 若要匯出規則、公式、指令碼或範本，請展開計畫類型，然後展開**規則**、**公式**、**指令碼**或**範本**
2. 執行下列其中一項任務：
   - 若只要匯出物件，請在物件上按一下右鍵，然後選取**匯出**。
   - 如果要匯出多個物件，請選取要匯出的物件，然後按一下滑鼠右鍵並選取**匯出**。請使用 **Shift + 按一下**及 **Ctrl + 按一下**，在不同的計算、計畫類型或資料庫、不同物件類型 (例如，商業規則及公式)，以及某一應用程式類型內的不同應用程式中，選取連續或非連續的物件。

     選取「匯出」後，系統會提示您開啟或儲存已產生的 .xml 檔案。
3. 在**檔案下載**中，執行下列其中一項任務：
   - 若要檢視產生之 .xml 檔案的內容，請選取**開啟**。
   - 若要儲存產生的 .xml 檔案，而不先檢視，請選取**儲存**，輸入檔案名稱 (或接受預設)，然後再按一下**儲存**。

匯出應用程式

當您匯出應用程式時，應用程式內容會儲存為 xml 檔案。

若要匯出應用程式，請執行下列動作：

1. 用滑鼠右鍵按一下應用程式，然後選取**匯出**。
2. 在**檔案下載**中，執行下列其中一項任務：
   - 若要檢視產生之 xml 檔案的內容，請選取**開啟**。
   - 如果要儲存已產生的 xml 檔案，而不是先檢視其內容，請選取**儲存**，為檔案輸入名稱 (或者接受預設值)，並再次按一下**儲存**。

將日誌訊息匯出至檔案

當您從 Calculation Manager 內啟動 Planning 商業規則時，將會產生日誌訊息並顯示在「規則設計工具」內的「日誌訊息」頁籤中。

您可以將這些日誌訊息匯出至逗點分隔值 (.csv) 檔案。請參閱從規則設計工具啟動 Planning 商業規則和檢視日誌。

若要匯出在 Calculation Manager 內啟動 Planning 商業規則時產生的日誌訊息，請執行下列動作：

1. 在**系統檢視**中，連按兩下您要啟動的規則。
2. 當規則在「規則設計工具」中開啟時，選取**動作**，然後選取**啟動**。

   規則執行之後，將會顯示確認訊息，指示啟動是否發生錯誤。
3. 按一下**確定**結束確認訊息。

   日誌訊息顯示在「日誌訊息」頁籤上。
4. 若要匯出啟動規則時所產生的日誌訊息，請依序選取**動作**、**匯出**。

匯出之後，您可以下載名為 `RuleLogMessages.csv` 的檔案，該檔案內含表格的所有日誌訊息。使用 Microsoft Excel 開啟然後使用逗號作為區隔符號儲存檔案。

匯入規則、規則集、範本、公式和指令碼

使用 Calculation Manager 將規則、規則集、範本、公式以及指令碼匯入您的應用程式。

匯入後，您可以將匯入結果儲存至本機檔案。

如果要匯入物件，則它們必須屬於下列檔案類型之一：

- `.xml`，一種包含物件的檔案，採用 xml 格式
- `.csc`，一種包含物件的檔案，採用記算指令碼格式
- `.zip`，zip 檔案只能包含 xml 檔案

若要匯入物件，請執行下列動作：

1. 在**系統檢視**中，選取**動作**，然後選取**匯入**，或按一下 。
2. 在**匯入**對話方塊中，在**檔案匯入詳細資料**中，按一下**瀏覽**，然後選取要匯入的檔案。
3. 在**位置詳細資料**中，輸入應用程式類型、應用程式和立方體。
   - 如果該檔案是一個 `.csc` 檔案，您必須輸入位置詳細資料。
   - 如果檔案是 `.xml` 檔案，如果位置詳細資料已存在於匯入檔案時，您不需要輸入位置詳細資料。

   位置詳細資料中輸入的資訊置換匯入檔案中指定的位置。如果匯入檔案中沒有指定任何位置，您必須在**位置詳細資料**中輸入資訊；否則，匯入會失敗。
4. 在**匯入選項**中，選取下列其中一個選項：
   - **覆寫現有物件** — 匯入的物件會取代應用程式和立方體中的物件。
   - **忽略現有物件** — 只要物件不存在，匯入的物件會新增至應用程式和立方體中的物件；否則，不會匯入物件，而且結果中的狀態會顯示「已忽略」。
   - **因重複而導致出錯** — 如果您匯入的物件和應用程式和立方體中已存在的物件重複，重複物件的名稱會寫入至日誌檔中，而且不會匯入任何物件，匯入程序則會停止。
5. 按一下**匯入**。
6. **選擇性：**按一下**另存新檔**，以將匯入的結果匯入本機檔案中。

13

為規劃應用程式管理 Essbase 伺服器、應用程式和資料庫

另請參閱：

- 使用資料庫特性
  檢視及編輯一般、維度、統計資料、交易，以及修改特性。
- 移除資料庫物件的鎖定
  視您的權限而定，您可以檢視和解除鎖定物件。
- 啟動和停止應用程式
  您可以啟動至少具有讀取權限的應用程式。Oracle Essbase 會將新啟動的應用程式載入 Essbase 伺服器上的記憶體。
- 啟動和停止資料庫
  當您啟動資料庫時，Oracle Essbase 會將資料庫載入 Essbase 伺服器上的記憶體。
- 重新建構資料庫
  重新建構資料庫 (例如，新增成員至密集維度) 時，可能會需要重新計算資料區塊。
- 驗證大綱
  您可以驗證 Oracle Essbase 大綱以查看是否有任何錯誤。
- 從資料庫清除資料
  清除聚總儲存應用程式中的資料，並將資料區塊從區塊儲存應用程式中清除。
- 使用區塊儲存應用程式的位置別名
- 在聚總儲存資料庫上使用查詢追蹤
  使用查詢資料以選取最適用的一組聚總檢視以具體化資料庫。
- 壓縮聚總儲存資料庫大綱
  壓縮檔案以移除已刪除成員的記錄，並減少大綱檔案大小。
- 匯入及匯出層級 0 資料
  從 ASO 和 BSO 立方體匯入及匯出層級 0 資料。
- 合併累加資料切片
  您可以將所有累加式資料塊合併至主要資料庫資料塊，或將所有累加式資料塊合併至單一資料塊，同時讓主要資料庫資料塊維持不變。
- 聚總資料
  針對包含資料以及授予您計算權限的聚總儲存資料庫計算聚總。
- 執行聚總處理程序
  執行聚總程序可改善擷取效能。
- 管理要求
  使用「階段作業」視窗中的資訊來管理作用中要求。
- 新增規劃鑽研定義
  在 Calculation Manager 中，您可以列示、新增、編輯和刪除規劃計畫類型的這些儲存格鑽研定義。

使用資料庫特性

檢視及編輯一般、維度、統計資料、交易，以及修改特性。

另請參閱：

- 檢視及編輯資料庫特性
- 一般資料庫特性
- 維度特性
- 統計特性
- 交易特定
- 修改特性

檢視和編輯資料庫特性

您可以檢視和編輯規劃區塊儲存和聚總儲存應用程式的資料庫特性。

若要檢視或編輯資料庫特性，請執行下列動作：

1. 在**系統檢視**中，按一下 (資料庫特性)。
2. 在**企業檢視**中，展開 Planning 應用程式與區塊儲存或是聚總儲存應用程式，然後選取資料庫。

> **備註：**
>
> ASO 應用程式名稱與 Planning 中的 ASO 計畫類型不同。展開 ASO 應用程式並查看 ASO 立方體名稱。若要找出相符的 ASO 立方體，請檢查 Planning 中的立方體名稱。
>
> 例如，Vision 應用程式有兩個 ASO 立方體，而且在 DB 特性中分別顯示為 "VisASO" 加上 VisASO 立方體、"BVision" 加上 Vis1ASO 立方體。立方體名稱與 Planning 中的 ASO 計畫類型相符。

3. 檢視或編輯資料庫特性頁籤上的資訊，然後按一下 。

> **備註：**
>
> 當您變更資料庫特性時，必須先停止應用程式後再重新啟動它。請參閱啟動和停止應用程式

一般資料庫特性

一般資料庫資訊，包括下列區域的特性：

- **一般** — 輸入資料庫的描述，並檢視資料庫類型、資料庫狀態 (無論是否載入)，和資料庫的最小存取層級。
- **計算** — *僅限區塊儲存應用程式*
  - **聚總缺少值** — 聚總資料庫計算期間的缺少值。

    根據預設，進行完整資料庫計算期間，Oracle Essbase 不會彙總遺漏 (#Missing) 值。當資料未載入至父層級時，聚總缺少值可能會增進計算效能。對於您擁有資料庫管理員權限的資料庫，您可選擇是否要聚總缺少值。

    如果您從未在父項層級中載入資料，聚總缺少值可能會改善計算效能。如果聚總缺少值，而且在父項層級中載入資料，即使結果為 #Missing 值，父項層級值仍會由資料庫整合的結果取代。
  - **在方程式中建立區塊** — 為特定的成員組合建立資料區塊。

    如果建立方程式區塊，當您將非常數值指派給沒有資料區塊的成員組合時，Essbase 會建立一個資料區塊。建立方程式上的區塊可產生非常大的資料庫。

    當您將常數指派給稀疏維度上的成員時，Essbase 會建立一個資料區塊。因此，當您指派常數給稀疏成員時 (例如，"West = 5")，請勿選取「在方程式中建立區塊」。

    指派任何非常數給稀疏成員時，如果您想要已建立區塊，則必須選取「在方程式中建立區塊」。例如，如果 Actuals (稀疏 Scenario 維度的成員) 沒有資料，則必須選取「在方程式中建立區塊」以執行以下配置：2002Forecast = Actuals * 1.05;
  - **二次計算** — 重新計算特定成員。

    如果您選取「二次計算」，預設計算之後，會重新計算標示為兩段式的成員。兩段式標籤對於標示為科目的維度成員以及任何維度的「動態計算」與「動態計算和儲存」成員有效。
- **資料擷取緩衝區**
  - **緩衝區大小** — 擷取緩衝區的大小。用於處理並最佳化來自試算表增益集和來自報表指令碼的擷取。
  - **排序緩衝區大小** — 擷取排序緩衝區的大小
- **儲存** — *僅限區塊儲存應用程式*
  - **目前的 I/O 存取模式** — 目前的存取模式
  - **擱置中的 I/O 存取模式** — 根據預設，已設定下列其中一個選項：
    * **緩衝的 I/O** — 使用檔案系統緩衝區快取。如果在資料庫完成建立時 essbase.cfg 檔案內的 Direction 設定未指定直接 I/O，則預設為緩衝式 I/O。
    * **直接 I/O** — 忽略檔案系統緩衝區快取，並執行非同步的重疊 I/O，提供較快速的回應時間及更大的潛能以最佳化快取大小。如果已選取直接 I/O，Essbase 會在每一次啟動資料庫時，嘗試使用直接 I/O。如果無法使用直接 I/O，Essbase 會使用緩衝 I/O。選取直接 I/O 以使用快取記憶體鎖定功能，或由作業系統提供的不等待 (非同步) I/O。
  - **資料壓縮** — 根據預設，已設定下列其中一個選項：
    * **點陣圖編碼** — 點陣圖是用來表示資料儲存格。只有點陣圖、區塊標頭，和其他控制資訊會儲存在磁碟中。點陣圖編碼是壓縮資料最有效的方式。Essbase 僅儲存非缺少值，並且不會壓縮重複值或零值。當資料庫將資料區塊帶入至資料快取中時，將會使用點陣圖重新建立缺少值，因此會完全展開區塊。

* **RLE (執行長度編碼)** — 壓縮連續的重複值 (包含零)，而且記錄會保留每一個重複值以及連續重複的次數。如果平均區塊密度未大於百分之三，或是如果資料庫包含許多連續的零值或零以外的任何連續重複值，RLE 會是比較好的作法。
* **ZLIB** — 建立以壓縮中資料為基礎的資料字典。通常，當資料非常密集時，ZLIB 壓縮提供最佳壓縮比率。然而，在某些情況下，其他壓縮方式可能會產生更好的結果。利用 ZLIB 壓縮，已儲存的儲存空間與缺少儲存格的數目或相等值的連續儲存格數目，有少許的關聯或毫無關聯。
* **不壓縮** — 不執行資料壓縮。

維度特性

資料庫的維度資訊，包括：

- 資料庫中的維度數：
- *(僅限區塊儲存資料庫)* 維度類型 (密集或稀疏)
- 維度成員
- 已儲存成員
- *(僅限聚總儲存資料庫)* 每一個維度中的層級數

> **備註：**
>
> 維度特性則為唯讀。

統計特性

> **備註：**
>
> 統計特性則為唯讀。

聚總儲存應用程式的統計

- **一般** — 一般統計資訊，包括：
  - **資料庫開始時間** — 根據資料庫伺服器時區的開始時間
  - **資料庫經過時間** — 格式為「小時:分鐘:秒」的經過時間
  - **連線數** — 連線的使用者數目
- **聚總儲存統計** — 聚總資料庫的儲存統計，包括：
  - 對於應用程式中的每個維度，用以儲存層級的層級數和位元數 (聚總儲存資料庫並未儲存所有的維度層級)。
  - **位元的最大索引鍵長度** — 所有維度使用的所有位元總計。例如，所有維度的索引鍵包含 20 個位元數，Year 維度使用了前四個位元數。
  - **最大金鑰長度 (位元組)** — 每個儲存格索引鍵使用的位元組數

- **輸入層級儲存格數目** — 在沒有公式的情況下且假設其均為層級零的儲存格，跨維度層級 0 交集的儲存格數目，使用者可以在其中輸入資料
- **累加式資料塊數目** — 可用累加方式計算 (僅限必要時) 而非立即計算的資料交集數目
- **累加輸入儲存格數目** — 可用累加方式計算 (僅限必要時) 而非立即計算的輸入儲存格數目
- **聚總檢視數目** — 含有聚總儲存格的檢視數目
- **累加儲存格數目** — 受到要求或擷取時必須計算的儲存格數目，因為此數目是由低層級值累加所得。聚總儲存格值將在每次要求時進行計算，或是預先計算並儲存在磁碟中。
- **累加聚總儲存格數目** — 只可在必要時進行更新的聚總儲存格數目
- **累加資料查詢成本 (總成本比率)** — 從關聯聚總檢視擷取值的平均時間
- **輸入層級資料大小 (KB)** — 所有零級儲存格的資料大小，以 KB 為單位
- **聚總資料大小 (KB)** — 所有聚總儲存格中聚總資料的大小，以 KB 為單位

- **執行時間** — 執行時間統計，包括：
  - **快取命中比率** — 相對於需要從磁碟擷取，在快取中找到資訊的成功率
  - **目前快取大小** — 動態產生的快取大小
  - **目前快取大小限制 (KB)** — 快取大小的限制，以 KB 為單位
  - **自上次啟動後的頁面讀取數目** — 自啟動應用程式後已讀取的索引頁面數目 (無論是自動啟動或由使用者啟動)
  - **自上次啟動後的頁面寫入數目** — 自啟動應用程式後已更新的索引頁面數目 (無論是自動啟動或由使用者啟動)
  - **頁面大小 (KB)** — 頁面的大小，以 KB 為單位
  - **配置的資料磁碟空間 (KB)** — 為資料儲存所配置的硬碟空間總數，以 KB 為單位
  - **資料使用磁碟空間 (KB)** — 資料儲存所使用的硬碟空間總數，以 KB 為單位
  - **配置的暫存磁碟空間 (KB)** — 為資料儲存所配置的暫存磁碟空間總數
  - **使用的暫存磁碟空間 (KB)** — 為資料儲存所使用的暫存磁碟空間總數

> **備註：**
>
> *磁碟空間* 為**預設**資料表空間所使用的空間，而 *暫存磁碟空間* 為 `Temp` 資料表空間所使用的空間。在這兩種情況中，可能不會使用一些檔案中的部分空間。

區塊儲存應用程式的統計

- **一般**——一般統計資訊：
  - **資料庫開始時間**—根據資料庫伺服器時區的開始時間
  - **資料庫經過時間**—格式為「小時:分鐘:秒」的經過時間
  - **連線數**—連線的使用者數目
- **區塊**—區塊儲存資料庫的資料區塊相關統計資料：
  - **現有區塊數目**—存在的區塊總數 (包含資料)

- **區塊大小**—每一個展開 (解壓縮) 的資料區塊大小，以位元組為單位 (儲存格數目 * 8；理想狀況是介於 8 到 10 KB 之間)。若要選擇區塊大小，您必須變更資料庫的密集-稀疏組態。
- **區塊的潛在數量**—區塊的數量上限 (由某一個稀疏維度的成員數乘以另一個稀疏維度的成員數所衍生)。例如，「基本範例」資料庫包含 19 個 Product 成員與 25 個 Market 成員 (共用成員或僅頁籤成員未計入)。因為 Product 與 Market 皆為儲存資料的稀疏維度，總共有 19 x 25 = 475 個潛在資料區塊。
- **現有的層級 0 區塊**—現有的層級 0 區塊 (區塊的稀疏維度成員沒有任何子項) 總數 (包含資料)。由於資料可在上層載入，層級 0 區塊與資料輸入所建立的區塊不需要相同。
- **現有的上層區塊**—現有之非層級 0 的區塊總數 (包含資料)。上層區塊包含所有上層稀疏成員的組合加上包含層級 0 稀疏成員的上層組合。
- **區塊密度 (%)**—根據現有資料區塊的範例，每一個資料區塊中資料點填入的平均百分比。密集-稀疏組態應該最大化區塊密度。最大化區塊密度，但是可能導致資料區塊的擴大。區塊大小與區塊擴大的考量可能會忽略嘗試最大化區塊密度。
- **現有最大區塊百分比**—比較現有區塊數與潛在區塊數的百分比。此百分比是資料庫的稀疏程度。很小的百分比並非少見；例如，小於 1%。
- **壓縮比率**—磁碟中所儲存區塊的壓縮效率程度。壓縮比率通常表示區塊密度。
- **平均叢集比**—資料 (.pag) 檔案的分割層級。最大值 1 表示無分散。如果您發現擷取、計算或資料載入效能降低，且叢集化比率值大幅小於 1，請考慮匯出並重新載入資料以強制資料檔案重新寫入。重新寫入檔案會重組檔案，造成接近 1 的叢集化比率。
- **平均資料分散商數** - 資料庫中的可用空間例如，平均分散資料商數值 3.174765 表示資料庫有 3% 的分散空間。更新和計算資料時，如果區塊不再適合其原始空間，則會出現空間，並且會在檔案末尾附加或置入足夠大的其他空間。數字越大，空間越多；因此，需要更多時間獲得特定記錄。平均資料分散商數有助於決定是否應該進行重組。

· **執行時間**

- **索引檔案數目**—索引檔案總數。
- **分頁檔數目**—分頁檔的總數。

索引 (.ind) 或頁面 (.pag) 檔案到達 2 GB 上限之前，會建立另一個。索引和分頁檔的數目可以顯示資料庫的大約大小，有助於疑難排解績效問題。例如，如果您有一個索引或分頁檔，則資料庫大小為大於或等於 2 GB。如果您有兩個索引或分頁檔，則資料庫大小為大於或等於 4 GB。

教學課程影片：

管理 BSO 資料庫特性

交易特性

> **備註：**
>
> 交易特性只適用於區塊儲存資料庫。

「交易」頁籤會顯示關於存取資料庫的資訊。

- **確定的存取**可以讓交易保留與交易相關之所有資料區塊的讀/寫鎖定，直到交易完成並確定為止。也可以預先設定下列並行選項：
  - **等待 (秒)** — 交易等待存取已鎖定資料區塊的秒數。預設為 20 秒，但也可以預先設定**不限制**或**不等候**等其他值。
  - **前影像存取** — 針對其他並行處理作業期間鎖定的資料區塊，使用者具有唯讀存取權。
- **未確定的存取**可以讓交易在逐塊 (預設設定) 的基礎上，保留讀/寫鎖定。也可以在下列區域預先設定同步化點：
  - **確認區塊** — 在 Oracle Essbase 執行確認之前所更新的資料區塊數目
  - **確認列** — Essbase 執行確認之前的資料載入期間所處理的資料檔案列數

修改特性

「修改」頁籤會顯示對資料庫執行之最近作業 (大綱更新、資料載入或計算) 的相關資訊：

- **作業** — 作業類型，例如資料載入或計算
- **使用者** — 執行作業的使用者名稱
- **開始時間** — 根據 Essbase Server 的作業開始時間，包含準備任務，例如鎖定資料 (如需作業期間，請參閱應用程式日誌中的經過時間項目)。
- **結束時間** — 根據 Essbase Server 的作業結束時間
- **備註** — 選擇性註解

移除資料庫物件的鎖定

視您的權限而定，您可以檢視和解除鎖定物件。

具備「管理員」權限的使用者可以解除鎖定任何物件。沒有「管理員」權限的使用者只能解除鎖定他們鎖定的物件。

此伺服器會使用資料庫物件 (例如計算指令碼、報表指令碼和商業規則檔) 的簽出功能，來確保一次只能一位使用者修改物件。

依預設，物件會在您開啟它們以進行修改時鎖定，然後在您關閉它們時解除鎖定。

當您對 Calculation Manager 中的物件執行動作時，物件可能會被鎖定。若要解除鎖定物件，您要在「企業檢視」中選取您要解除鎖定之物件的計畫類型。

若要從資料庫移除鎖定，請執行下列動作：

1. 在**系統檢視**中，按一下**資料庫特性**圖示。
2. 在**企業檢視**中，展開 Planning 應用程式類型以及包含您要移除鎖定的資料庫的應用程式。
3. 用滑鼠右鍵按一下資料庫，然後選取**移除鎖定**。

啟動和停止應用程式

您可以啟動至少具有讀取權限的應用程式。Oracle Essbase 會將新啟動的應用程式載入 Essbase 伺服器上的記憶體。

您可以指定資料庫在啟動其父項應用程式時啟動。在這種情況下，如果您在使用者連線至應用程式中的資料庫之前，先啟動應用程式，使用者可能會感受到較佳的初始化效能 (資料庫連線時)，因為應用程式與所有相關聯的資料庫皆在記憶體中。

當您停止應用程式時，Essbase 會從 Essbase 伺服器上的記憶體上傳應用程式與應用程式中的所有資料庫。因此，會增加可用的記憶體。若要確保應用程式中的資料庫不會被損毀，您必須適時停止應用程式。

如要啟動或停止應用程式，請執行下列動作：

1. 在**系統檢視**中，按一下**資料庫特性**圖示。
2. 在**企業檢視**中，展開 Planning 應用程式類型以及您要啟動或停止的應用程式
3. 在應用程式上按一下滑鼠右鍵，然後選取：
   - **啟動應用程式**
   - **停止應用程式**

> **備註：**
>
> 您需要在對資料庫設定值進行變更的任何時候停止，然後重新啟動應用程式。

啟動和停止資料庫

當您啟動資料庫時，Oracle Essbase 會將資料庫載入 Essbase 伺服器上的記憶體。

索引快取會自動配置，而且會在要求區塊時配置資料檔案和資料快取。如果您在使用者存取之前先啟動資料庫，使用者可能會感受到較佳的初始效能 (連線時)，因為資料庫位於記憶體中。

當您自未啟動的應用程式中啟動資料庫時，會載入應用程式和應用程式中的所有資料庫。您可以啟動應用程式的一個資料庫或所有資料庫。

當您停止資料庫時，Essbase 會從 Essbase 伺服器上的記憶體卸載資料庫，並將更新的資料提交至磁碟。因此，伺服器電腦上會增加可用的記憶體。

您可以停止應用程式的一個資料庫或所有資料庫。

> **備註：**
>
> 您可以啟動和停止區塊儲存應用程式資料庫。

如要啟動或停止資料庫，請執行下列動作：

1. 在**系統檢視**中，按一下**資料庫特性**圖示。
2. 在**企業檢視**中，展開 Planning 應用程式類型以及包含您要啟動或停止的資料庫的應用程式

3. 在資料庫上按一下滑鼠右鍵，然後選取：
   - **啟動資料庫**
   - **停止資料庫**

如要啟動或停止應用程式的所有資料庫，請執行下列動作：

1. 在**系統檢視**中，按一下**資料庫特性**圖示。
2. 在**企業檢視**中，展開 Planning 應用程式類型以及含有要啟動或停止之資料庫的應用程式
3. 在應用程式上按一下滑鼠右鍵，然後選取：
   - **啟動所有資料庫**
   - **停止所有資料庫**

重新建構資料庫

重新建構資料庫 (例如，新增成員至密集維度) 時，可能會需要重新計算資料區塊。

Oracle Essbase 會將所有資料區塊標示為已變更。當您計算重新建構的資料庫時，將會計算所有區塊。

> **備註：**
>
> 您可以重新建構區塊儲存應用程式資料庫。

在下列情況下，您應該重新建構資料庫：

- 新增、刪除或移動密集維度「動態計算和儲存」成員。
- 將密集維度「動態計算和儲存」成員變更為「動態計算」成員。
- 將密集維度「動態計算」成員變更為「動態計算和儲存」成員。
- 將密集維度中之非動態成員的儲存特性變更為「動態計算」。
- 將「動態計算」成員之密集維度的儲存特性變更為非動態值。
- 變更稀疏維度 Dynamic Calc 或 Dynamic Calc 及 Store 中之非動態成員的儲存特性。

若要重新建構資料庫，請執行下列動作：

1. 從**系統檢視**中，按一下**資料庫特性**圖示。
2. 在**企業檢視**中，展開 Planning 應用程式類型、應用程式、以及您要重新建構其資料庫的計畫類型。
3. 用滑鼠右鍵按一下計畫類型，然後選取**重新建構資料庫**。
4. 確認您是否要重新建構資料庫。

   Calculation Manager 會顯示訊息，通知您是否已經順利重新建構資料庫。

驗證大綱

您可以驗證 Oracle Essbase 大綱以查看是否有任何錯誤。

大綱驗證程序會考量大綱類型 (聚總儲存或區塊儲存)，並根據每種類型的規則來驗證大綱。大綱無錯誤之後，將驗證成員公式。

若要驗證大綱，請執行下列動作：

1. 從**系統檢視**中，按一下**資料庫特性**圖示。
2. 在**企業檢視**中，展開 Planning 應用程式類型、應用程式，以及您要驗證之大綱的計畫類型。
3. 在計畫類型按一下滑鼠右鍵，然後選取**驗證大綱**。

驗證大綱時，Essbase 會檢查下列項目：

- 所有成員與別名有效。成員名稱與別名不能與其他成員、別名、層代或層級的名稱相同。
- 只有一個維度標記為科目、時間、幣別類型或國家/地區。
- 共用成員有效。
- 層級 0 成員未標記為僅標籤。
- 僅標籤成員尚未指派公式。
- 幣別類別和幣別名稱對於幣別大綱有效。
- 疏鬆維度中的動態計算成員子項不超過 100 個。
- 如果父成員有一個子項，並且該子項是動態計算成員，則父成員必須也是動態計算。
- 如果父成員有一個子項，並且該子項是動態計算、二次計算成員，則父成員必須也是動態計算。
- 布林屬性維度的兩個成員名稱與為大綱定義的兩個布林屬性維度成員名稱相同。
- 日期屬性維度的層級 0 成員名稱必須符合日期格式名稱設定 (mm-dd-yyyy or dd-mm-yyyy)。如果維度沒有成員，則維度名稱必須符合設定，因為維度名稱是層級 0 成員。
- 數值屬性維度的層級 0 成員名稱為數值。如果維度沒有成員，則維度名稱必須為數值，因為維度名稱是層級 0 成員。
- 屬性維度位於大綱節尾，在所有標準維度後面。
- 標準維度的層級 0 動態計算成員具有公式。
- 成員公式有效。
- 在混合分析大綱中，只有維度的層級 0 成員可以啟用混合分析。

在大綱驗證期間，Essbase 也會對相應的數值屬性維度成員名稱執行以下轉換，並在大綱中顯示這些名稱：

- 它會將成員名稱中的減號從名稱的前面移到後面；例如，-1 變成 1-。
- 它會刪除成員名稱中前置或後置的零；例如，1.0 變成 1，00.1 變成 0.1。

從資料庫清除資料

清除聚總儲存應用程式中的資料，並將資料區塊從區塊儲存應用程式中清除。

另請參閱：

- 將資料從聚總儲存應用程式中清除
- 將資料區塊從區塊儲存應用程式中清除

將資料從聚總儲存應用程式中清除

若要將資料從聚總儲存應用程式中清除，請執行下列動作：

1. 在**系統檢視**中，按一下 。
2. 在**企業檢視**的**資料庫特性**頁籤中，展開聚總儲存應用程式。
3. 在 ASO 立方體上按一下滑鼠右鍵，然後選取**清除**，然後選取選項：
   - **所有資料** - 從資料庫清除所有資料。
   - **所有聚總** - 從資料庫清除所有聚總的資料值
   - **部分資料** - 將資料從聚總儲存資料庫中的特定區域中清除，並保留位於其他區域中的資料。

     當您選取清除部分資料時，請輸入 MDX 表示式，以定義要清除的區域，並選取**邏輯**或**實體**，以指定資料清除的類型

     – **邏輯** - 指定區域中的資料將以負補償值寫入新的資料切片中，並在清除的儲存格中產生零值。

     – **實體** - 將從聚總儲存資料庫中實際移除指定區域中的資料。

     若要使用「成員選取器」以建立 MDX 語法，請按一下 。在**成員選取器**對話方塊中，選取一或多個成員，或使用來自每個維度的函式，然後按一下**確定**以建立 MDX 表示式。

     例如：

     ```
     Crossjoin(Crossjoin(Crossjoin(Crossjoin(Crossjoin(Crossjoin(Crossjoin({[NI
     ].Levels(0).Members},{except(DESCENDANTS([Q1]),{[Q1]})}),{[FY12],[FY13],
     [FY14]}),{[Plan]}),{[Working]}),{[100].Levels(0).Members}),
     {[P_TP].Levels(0).Members}),{[FY06]}
     ```

     請注意，當您按一下**確定**以清除資料時，會驗證 MDX 表示式。如果您直接修改 MDX 表示式，將會在執行前驗證。

> **備註：**
>
> 您可以僅從您具有權限的資料庫清除資料。

將資料區塊從區塊儲存應用程式中清除

若要將資料區塊從區塊儲存應用程式中清除，請執行下列動作：

1. 在**系統檢視**中，按一下 。
2. 在**企業檢視**的**資料庫特性**頁籤中，展開區塊儲存應用程式。
3. 在 BSO 立方體上按一下滑鼠右鍵，然後選取**清除**，然後選取選項：
   - **所有資料** - 從資料庫清除所有資料。

- **上層區塊** - 可僅清除上層資料區塊。上層區塊的資料值會設成 #Missing。對於至少一個有一個稀疏成員是父成員的稀疏成員組合，會建立上層區塊。
- **非輸入區塊** - 可僅清除包含衍生自計算之值的資料區塊 (非輸入區塊)。當您清除非輸入區塊時，非輸入 (計算) 儲存格的資料值會設定成 #Missing。
- **動態區塊** - 可僅清除動態計算並儲存的資料區塊 (即「動態計算」成員和「儲存」成員)。
- **空白區塊** - 可僅清除空白資料的區塊 (即所有值為 #Missing 的區塊)。

> **備註：**
>
> 您可以僅從您具有權限的資料庫清除資料。

使用區塊儲存應用程式的位置別名

另請參閱：

- 關於位置別名
- 顯示位置別名的清單
- 匯出位置別名

關於位置別名

位置別名是資料來源的描述項。位置別名將資料庫的別名對映至該資料庫的實體位置。已在資料庫或應用程式層級設定位置別名，並指定別名、伺服器、應用程式、資料庫、使用者名稱，以及密碼。您可以顯示及匯出位置別名。

> **備註：**
>
> 位置別名*不*適用於聚總儲存資料庫。

顯示位置別名的清單

若要顯示區塊儲存應用程式的位置別名清單，請執行下列動作：

1. 在**系統檢視**中，按一下 (資料庫特性)。
2. 在**企業檢視**中，用滑鼠右鍵按一下區塊儲存應用程式，然後選取**位置別名**。
3. 按一下**確定**以確認您要啟動資料庫，然後在**資料庫啟動動作狀態**對話方塊中按一下**確定**。

匯出位置別名

若要匯出位置別名，請執行下列動作：

1. 在**系統檢視**中，按一下 (資料庫特性)。

2. 在**企業檢視**中，用滑鼠右鍵按一下區塊儲存應用程式，然後選取**位置別名**。
3. 按一下**確定**以確認您要啟動資料庫，然後在**資料庫啟動動作狀態**對話方塊中按一下**確定**。
4. 在**位置別名**對話方塊中，按一下 (匯出)，接著選取要匯出的檔案，然後按一下**確定**。

在聚總儲存資料庫上使用查詢追蹤

使用查詢資料以選取最適用的一組聚總檢視以具體化資料庫。

您可啟用查詢追蹤，以擷取關於對資料庫執行之每一個查詢的成本資料。查詢成本是從檢視擷取值所需要的估計平均擷取時間。對於第一次檢視 (依預設選取項目)，估計值為所有可能查詢的平均值。對於所使用查詢追蹤的檢視，估計值為所追蹤查詢的平均值。因此，檢視可能會依不同的情況，顯示不同的估計值。若要運算使用特殊檢視估計優勢的百分比，可將檢視的查詢成本值除以使用僅包含層級 0 值之檢視的查詢成本值。

一旦啟用，查詢追蹤會持續到以下其中一個情況發生為止：

- 為資料庫停用查詢追蹤，如本主題中所述。
- 應用程式關閉。如果應用程式已關閉，重新啟動應用程式時，查詢追蹤不會自動恢復。
- 為資料庫具體化其他聚總檢視。因為查詢追蹤資料會在具體化其他檢視時變成無效，具體化任何新檢視會重新設定查詢追蹤。

查詢追蹤僅儲存在記憶體中，包含來自 Oracle Hyperion Web Analysis、方格 API、報表指令碼、Java API 之類的查詢。

> **備註：**
>
> 查詢追蹤僅可用於聚總儲存資料庫。

若要啟用或停用查詢追蹤，請執行下列動作：

1. 在**系統檢視**中，按一下
2. 在**企業檢視**中，用滑鼠右鍵按一下計畫類型，選取**查詢追蹤**，然後選取下列其中一個選項：
   - **啟用**，以啟用查詢追蹤
   - **停用**，以停用查詢追蹤

     當您啟用查詢追蹤時，資料庫會記錄查詢資料；當您停用查詢追蹤時，資料庫會停止記錄查詢資訊，並將查詢資料從記憶體清除。
3. 按一下**確定**，以確認您想要啟用或停用查詢追蹤。

壓縮聚總儲存資料庫大綱

壓縮檔案以移除已刪除成員的記錄，並減少大綱檔案大小。

例如，由於在新增或刪除成員時，已變更聚總儲存大綱檔案 (`.otl` 檔案)，因此檔案大小可能會增加。壓縮大綱檔案之後，檔案會按照之前情況繼續增長。

壓縮大綱檔案會造成資料庫重新建立大綱。壓縮大綱不會造成資料庫清除資料。

從大綱刪除成員時，該成員在大綱檔案中的對應記錄會標記為「已刪除」，但是記錄仍會保留在大綱檔案中。壓縮大綱檔案不會移除已刪除成員的記錄。

> **備註：**
>
> 您僅可以壓縮聚總儲存資料庫大綱。壓縮大綱的處理程序僅發生在沒有其他使用者或處理程序頻繁地使用資料庫時

若要壓縮聚總儲存資料庫大綱，請執行下列動作：

1. 在**系統檢視**中，按一下
2. 在**企業檢視**中，用滑鼠右鍵按一下包含您要壓縮之大綱的計畫類型，然後選取**壓縮大綱**。
3. 按一下**確定**，以確認您想要壓縮資料庫大綱。

   壓縮大綱動作狀態對話方塊會顯示壓縮進度。當大綱已壓縮時，**壓縮大綱動作狀態**對話方塊會顯示成功訊息。
4. 在**壓縮大綱動作狀態**對話方塊中，按一下**顯示詳細資料**，以查看大綱壓縮的詳細資料，或按一下**確定**，以關閉對話方塊。

匯入及匯出層級 0 資料

從 ASO 和 BSO 立方體匯入及匯出層級 0 資料。

另請參閱：

- 從 ASO 立方體匯入層級 0 資料
- 從 ASO 立方體匯出層級 0 資料
- 從 BSO 立方體匯入層級 0 資料
- 從 BSO 立方體匯出層級 0 資料

從 ASO 立方體匯入層級 0 資料

若要從 ASO 立方體匯入層級 0 資料，請執行下列動作：

1. 在**系統檢視**中，按一下 。
2. 在**企業檢視**的**資料庫特性**頁籤中，展開聚總儲存應用程式。
3. 用滑鼠右鍵按一下 ASO 立方體，然後選取**匯入層級 0 資料**。
4. 在**匯入立方體的層級 0 資料**對話方塊中，輸入下列資訊，然後按一下**確定**。
   - **Zip 檔案名稱** - 匯入資料的 Zip 檔案名稱。
   - **複製聚總方法** - 定義如何為相同儲存格合併多個值。
     - **新增重複值** - 當緩衝區包含相同儲存格的多個值時新增值。
     - **假設值相等** - 驗證相同儲存格的多個值是否相同；如果是，則忽略重複值。如果相同儲存格的值不同，會停止資料載入並出現錯誤訊息。

  - **使用最後一個值** - 藉由使用最後載入至載入緩衝區之儲存格的值合併重複的儲存格。此選項適用於最多 10,000 個儲存格之比較小的資料載入。使用此選項時，即使沒有任何重複值，資料載入速度也會大幅降低。
- **選項** - 如果需要，您可以選擇以下選項：
  - **忽略缺少的值** - 忽略內送資料流的 #MISSING 值。
  - **忽略零值** - 忽略內送資料流的零值。

5. 在**層級 0 資料匯入狀態**對話方塊中，按一下**顯示詳細資料**，以顯示匯入的詳細資料，然後按一下**確定**。

   匯入資料後，您可以使用規劃表單以查看資料，或可以查看立方體統計以檢視值的變更方式。

> **備註：**
>
> 如果匯入資料包含的規劃文字值、智慧型列示值或幣別值來自匯入資料以外的來源，則資料可能毀損。

從 ASO 立方體匯出層級 0 資料

若要從 ASO 立方體匯出層級 0 資料，請執行下列動作：

1. 在**系統檢視**中，按一下 。
2. 在**企業檢視**的**資料庫特性**頁籤中，展開聚總儲存應用程式。
3. 用滑鼠右鍵按一下 ASO 立方體，然後選取**匯出層級 0 資料**。
4. 在**匯出立方體的層級 0 資料**對話方塊中，輸入匯出資料的 zip 檔案名稱，然後按一下**確定**。

若要查看規劃中包含匯出的層級零資料的 .zip 檔案：

1. 在「Planning 首頁」上，按一下**應用程式**，然後按一下**概觀**。
2. 在**應用程式**頁面中，按一下**動作**，然後按一下**收件匣/寄件匣檔案總管**。
3. 在**收件匣/寄件匣檔案總管**中，按一下 .zip 檔案旁的「動作」圖示，然後選取**下載檔案**。

從 BSO 立方體匯入層級 0 資料

若要從 BSO 立方體匯入層級 0 資料，請執行下列動作：

1. 在**系統檢視**中，按一下 。
2. 在**企業檢視**的**資料庫特性**頁籤中，展開區塊儲存應用程式。
3. 用滑鼠右鍵按一下 BSO 立方體，然後選取**匯入層級 0 資料**。
4. 在**匯入立方體的層級 0 資料**對話方塊中，輸入匯入資料的 zip 檔案名稱，然後按一下**確定**。

從 BSO 立方體匯出層級 0 資料

若要從 BSO 立方體匯出層級 0 資料，請執行下列動作：

1. 在**系統檢視**中，按一下 。
2. 在**企業檢視**的**資料庫特性**頁籤中，展開聚總儲存應用程式。
3. 用滑鼠右鍵按一下 BSO 立方體，然後選取**匯出層級 0 資料**。
4. 在**匯出立方體的層級 0 資料**對話方塊中，輸入匯出資料的 zip 檔案名稱，然後按一下**確定**。

若要查看規劃中包含匯出的層級零資料的 .zip 檔案：

1. 在「Planning 首頁」上，按一下**應用程式**，然後按一下**概觀**。
2. 在**應用程式**頁面中，按一下**動作**，然後按一下**收件匣/寄件匣檔案總管**。
3. 在**收件匣/寄件匣檔案總管**中，按一下 .zip 檔案旁的「動作」圖示，然後選取**下載檔案**。

合併累加式資料塊

您可以將所有累加式資料塊合併至主要資料庫資料塊，或將所有累加式資料塊合併至單一資料塊，同時讓主要資料庫資料塊維持不變。

若要合併資料塊，您必須具備與用於載入資料相同的權限 (「管理員」或「資料庫管理員」權限)。

> **備註：**
>
> 您只能針對聚總儲存資料庫合併累加式資料塊。

若要合併累加式資料塊，請執行下列動作：

1. 在**系統檢視**中，按一下
2. 在**企業檢視**中，用滑鼠右鍵按一下含有要合併資料之資料庫的計畫類型，選取**合併資料**，然後選取下列其中一個選項：
   - **全部**，將所有資料塊合併成一個
     - 保留零值的儲存格 (預設)
     - 移除零值的儲存格
   - **累加式**，將累加式資料塊合併成一個，並執行其中一項動作：
     - 保留零值的儲存格 (預設)
     - 移除零值的儲存格
3. 在**確認合併資料動作**對話方塊中，按一下**確定**以確認您要合併資料。

聚總資料

針對包含資料以及授予您計算權限的聚總儲存資料庫計算聚總。

若要執行聚總，請使用系統建議的檢視。檢視的選擇和聚總處理程序已合併成一個伺服器執行的無法設定作業。您可以根據使用者查詢樣式的檢視選擇，為產生的檔案選擇性地指定最大磁碟空間，並將向上彙整階層加入檢視選擇。

> **備註：**
>
> 您可以針對聚總儲存資料庫聚總資料。如需聚總資料時最佳作法的範例，請參閱執行聚總處理程序。

若要執行聚總，請執行下列動作：

1. 在**系統檢視**中，按一下
2. 在**企業檢視**中，用滑鼠右鍵按一下包含您要壓縮大綱之資料庫的計畫類型，然後選取**執行聚總**。
3. 在**執行聚總動作 - 使用建議的檢視**對話方塊中，選取下列其中一個選項：
   - **根據查詢資料？** - 根據收集的使用者查詢圖樣，聚總伺服器選取的檢視。此選項僅適用於開啟查詢追蹤時。
   - **包括累加選項？** - 在檢視選擇處理程序中包括次要階層 (帶有預設層級用途)。
   - **包含增長規模選項？** - 聚總伺服器選取的檢視，直到聚總資料庫的最大增長超過您指定的限制為止。輸入伺服器應停止聚總的規模 (以 MB 為單位) 上限。
4. 按一下**確定**。

   如果存在現有的聚總資料，會出現訊息詢問您是否要刪除現有聚總並重新執行聚總處理程序。如果存在現有資料，重新執行聚總處理程序之前會先刪除該資料。
5. 按一下**確定**以刪除現有聚總資料，並重新執行聚總。

執行聚總程序

執行聚總程序可改善擷取效能。

ASO 立方體未使用計算指令碼來聚總資料；相反地，ASO 嘗試動態計算上層成員。這可造成更快速的批次處理時間，但可能造成較長的擷取時間。您可以藉由針對某些作業 (例如使用表單或執行特設報表)，開啟「查詢追蹤」，以對應 ASO 立方體擷取查詢來改善此情況。這些查詢用於彙總程序，其告知 Oracle Essbase 以使用「查詢追蹤」選取的查詢樣式以建立「彙總」檢視。建立聚總檢視之後，您應該可看見在擷取效能方面的改善。

執行聚總程序之前，請執行下列動作：

- 合併增量資料塊並移除零值儲存格
- 啟用查詢追蹤
- 執行動作以建立查詢
- 使用查詢追蹤執行聚總

合併累加式資料塊與移除零值的儲存格

有些 Oracle Essbase 操作 (像是邏輯清除與載入資料) 可能會建立累加式資料塊與零值的儲存格。Essbase 不允許累加式資料塊的聚總。因此，您可能需要執行合併作業，以合併累加式資料塊。

若要執行合併作業和移除零值的儲存格，請執行下列動作：

1. 在**系統檢視**中，按一下 ，然後選取聚總儲存資料庫以載入其特性。

2. 在**企業檢視**中，在立方體上按一下滑鼠右鍵，然後依序選取**合併資料**、**全部**及**移除零值的儲存格**。
3. 按一下**確定**以確認合併資料動作。

   這會將所有的累加式資料塊合併到主要資料庫資料塊，並移除有零值的儲存格。(從某個區域中以邏輯方式清除資料，會導致零值的儲存格)。因此，資料庫大小將會大幅縮小。

   如果不需要合併，會顯示訊息指出「沒有累加式資料或是它已經合併。指定的合併並非必要。」(按一下**合併動作狀態**中的**顯示詳細資料**對話方塊，以查看完整訊息。)

啟用查詢追蹤

您可以針對 ASO 資料庫啟用查詢追蹤，以記錄有意義的一組查詢，然後使用記錄的查詢資料，以選取最適合的一組聚總檢視以體現該資料庫。

在規劃中完成的立方體重新整理作業會執行大綱重新建立作業。作為重新建立作業的一部分，Oracle Essbase 會移除追蹤的查詢。合併作業也會移除追蹤的查詢。

若要啟用查詢追蹤，請執行下列動作：

1. 在**企業檢視**中，展開 **Planning**，然後展開 ASO 應用程式，然後用滑鼠右鍵按一下 ASO 立方體，然後選取**設定查詢追蹤**。
2. 按一下**確定**，以確認查詢追蹤動作。
3. 在**資訊**對話方塊中按一下**確定**，會通知您資料庫上的查詢追蹤已啟用成功。

當您啟用「查詢追蹤」後，可重複上述步驟以停用之。如果您停用查詢追蹤，則**資訊**對話方塊會通知您已啟用「查詢追蹤」，並詢問您是否要停用查詢追蹤、停止應用程式，或執行「聚總」程序。

執行建立查詢的動作

僅儲存在記憶體中的「查詢追蹤」，可追蹤針對使用規劃中的 ASO 立方體開啟表單，以及執行特設報表等的作業進行的查詢。

在您啟用查詢追蹤後，可以繼續載入表單，執行商業規則，然後執行報表。Oracle Essbase 將會繼續追蹤查詢並儲存統計。在執行聚總時，您可以使用這些追蹤的統計。

使用查詢追蹤執行聚總

當您執行足夠的查詢量時，您可以使用查詢追蹤資訊執行聚總程序。

若要使用查詢追蹤資訊執行聚總程序，請執行下列動作：

1. 在**企業檢視**中，展開 **Planning**，然後展開 ASO 應用程式，然後用滑鼠右鍵按一下 ASO 立方體，然後選取**執行聚總**。
2. 在**執行聚總動作**對話方塊中，選取**根據查詢資料**，然後按一下**確定**。

   這項作業可能需要一些時間完成。

Calculation Manager 會檢查下列可能造成執行聚總程序失敗的條件：

- 如果查詢追蹤未啟用，或沒有查詢追蹤資料，則會顯示警告訊息。您將無法使用查詢追蹤選項，但您仍可處理聚總。若要使用查詢追蹤資訊，您必須啟用查詢追蹤或執行產生查詢追蹤資訊的作業。
- 當多個資料庫資料塊存在時，Oracle Essbase 不允許在 ASO 立方體上建立「彙總」檢視。會顯示警告訊息，您將無法繼續進行聚總，直到合併資料塊為止。

在此情況下，在**企業檢視**中，用滑鼠右鍵按一下立方體節點，並依序選取**合併資料**、**全部**，然後選取**移除包含零值的儲存格**。合併會清除所有追蹤的查詢資訊。因此，您必須執行產生查詢追蹤資訊的作業。

- 如果聚總已存在，則會出現警告訊息。
  在此情況下，按一下**確定**，然後在資訊對話方塊中，請執行下列其中一項動作：
  - 按一下**確定**，以拖曳現有聚總，並重新執行聚總處理程序。
  - 按一下**取消**，然後清除現有聚總，然後執行產生查詢追蹤資訊的作業。

  若要清除聚總，請執行下列動作：

  1. 用滑鼠右鍵按一下 ASO 立方體，然後依序選取**清除**、**所有聚總**。
  2. 在**確認資料庫聚總清除**對話方塊中，按一下**確定**。

管理要求

使用「階段作業」視窗中的資訊來管理作用中要求。

「階段作業」視窗列出作用中的使用者階段作業，並要求伺服器、應用程式或資料庫。使用者可擁有一個以上在任何給定時間開放的階段作業。例如，一個使用者可擁有在兩個資料庫上開放的階段作業。

如果您擁有「管理員」或「應用程式管理員」權限，您可以結束所有要求、結束使用者的所有要求，或結束特定要求。

若要結束要求，請執行下列動作：

1. 在**系統檢視**中，按一下
2. 在**企業檢視**中，用滑鼠右鍵按一下應用程式，然後選取**階段作業**。

   階段作業視窗會顯示作用中階段作業和要求的清單。如果您擁有「管理員」權限，則視窗會列出在伺服器上所有使用者的作用中使用者階段作業。如果您擁有「應用程式管理員」權限，則視窗會列出所有使用者 (包含您本身) 連接至您擁有「應用程式管理員」權限之任何應用程式的作用中階段作業。
3. 如果您想要讓一或多個使用者登出，請在**選項**下方，從**動作**下拉清單中，選取**登出**，然後執行下列其中一項任務：
   - 在**實體**中，選取**選取的使用者**，然後選取要登出的使用者。然後按一下**套用**將使用者登出。
   - 從**實體**中，選取**所有使用者**，然後執行下列其中一項任務：
     - 從**來源**中，選取**在選取的伺服器上**，在選取的伺服器上登出所有使用者。然後按一下**套用**。
     - 從**來源**中，選取**在選取的應用程式上**，在選取的應用程式上登出所有使用者。然後按一下**套用**。
     - 從**來源**中，選取**在選取的資料庫上**，在選取的資料庫上登出所有使用者。然後按一下**套用**。
   - 從**實體**中，選取**所有使用者例項**，然後執行下列其中一項任務：
     - 從**來源**中，選取**在選取的伺服器上**，然後選取您要登出的使用者例項。按一下**套用**將該伺服器上的所有使用者實例登出。
     - 從**來源**中，選取**在選取的應用程式上**，然後選取您要登出的使用者例項。按一下**套用**將該應用程式上的所有使用者實例登出。

  - 從**來源**中，選取**在選取的資料庫上**，然後選取您要登出的使用者例項。按一下**套用**將該資料庫上的所有使用者實例登出。

4. 若要結束一或多個要求，在**選項**下方，從**動作**下拉清單中，選取**清除**，然後執行下列其中一項任務：
   - 從**實體**中，選取**選取的要求**，然後選取想要從階段作業清單中結束的要求。然後按一下**套用**，以結束選取的要求。
   - 從**實體**中，選取**所有要求**，然後執行下列其中一項任務：
     - 從**來源**中，選取**在選取的伺服器上**，在選取的伺服器上結束所有使用者的所有要求。然後按一下**套用**。
     - 從**來源**中，選取**在選取的應用程式上**，在選取的應用程式上結束所有使用者的所有要求。然後按一下**套用**。
     - 從**來源**中，選取**在選取的資料庫上**，在選取的資料庫上結束所有使用者的所有要求。然後按一下**套用**。
   - 從**實體**中，選取**來自使用者的所有要求**，然後執行下列其中一項任務：
     - 從**來源**中，選取**在選取的伺服器上**，然後從階段作業清單中選取使用者的要求。按一下**套用**，以結束該使用者在伺服器上的所有要求。
     - 從**來源**中，選取**在選取的應用程式上**，然後從階段作業清單中選取使用者的要求。按一下**套用**，以結束該使用者在應用程式上的所有要求。
     - 從**來源**中，選取**在選取的資料庫上**，然後從階段作業清單中選取使用者的要求。按一下**套用**，以結束該使用者在資料庫上的所有要求。
5. 若要顯示及隱藏欄，請選取**檢視**，然後選取**欄**。執行下列任意操作：
   - 選取**全部顯示**以顯示所有欄。依預設，會顯示**連線來源**欄以外的所有欄。
   - 選取**管理欄**，在**隱藏欄**和**可見欄**清單之間移動欄。
6. 若要重新排序欄，請選取**檢視**，然後選取**重新排序欄**。在**重新排序欄**對話方塊中，使用上下方向鍵依您的喜好重新排序欄。
7. 若要依欄排序階段作業清單，請執行下列動作：
   - 若要以遞增順序排序欄，請按一下欄標頭，或按一下欄標頭旁邊的**向上**箭頭。

     例如，若要依字母順序排列**使用者**欄的順序，請按一下欄標頭。
   - 若要依遞減順序排序欄，請按 **Shift** 並按一下欄標頭，或按一下欄標頭旁邊的**向下**箭頭。

     例如，若要排序**登入時間**欄，使得最長登入時間出現在清單中第一個，請按 **Shift** 並按一下欄標頭。
8. 若要重新整理階段作業清單，請按一下**重新整理**。例如，如果您結束處理程序中顯示的階段作業，然後按一下**重新整理**，則階段作業清單會顯示該階段作業不再處理。
9. 若要將階段作業資料匯出至 Microsoft Excel 試算表，請按一下**匯出**按鈕，開啟或儲存檔案，然後按一下**確定**。

新增規劃鑽研定義

在 Calculation Manager 中，您可以列示、新增、編輯和刪除規劃計畫類型的這些儲存格鑽研定義。

如果您使用規劃計畫類型和其中包含的成員是從資料管理等的來源載入資料的規劃表單，您可以鑽研以檢視更多的儲存格資料來源詳細資料。

若要新增規劃鑽研定義，請執行下列動作：

1. 在**系統檢視**中，按一下 (資料庫特性)。
2. 在**企業檢視**中，用滑鼠右鍵按一下應用程式，然後選取**鑽研定義**。

   如果系統要求您確認是否要啟動資料庫，請按一下**確定**。
3. 按一下**確定**以確認您要啟動資料庫，然後在**資料庫啟動動作狀態**對話方塊中按一下**確定**。
4. 在**鑽研定義**對話方塊中，按一下 (新增)。
5. 在**建立鑽研定義**中，藉由輸入下列資訊，建立鑽研定義：
   - **URL 名稱** - 識別鑽研定義的名稱
   - **XML 內容** - 定義 URL 連結的 XML

     輸入不含伺服器和埠資訊的 URL。URL 必須包含來自 `TDATASEG` 表格的參數名稱和欄名稱並以符號 `$` 括住。例如，請輸入：`LEDGER_ID=$ATTR1$&GL_PERIOD=$ATTR2$`。在此範例中，`ATTR1` 中的值會傳遞做為 `LEDGER_ID` 參數的值，而 `ATTR2` 會傳遞做為 `GL_PERIOD` 參數的值。參數以 `&` 字元分隔。

     若要為鑽研 URL 格式指定用戶端與伺服器之間的要求-回應，請輸入下列一項：

     – `GET` - 將表單資料編碼至 URL

       例如，輸入：`GET@http://www.oracle.com/`。如果未輸入任何方法，則 `GET` 為假設的要求回應。

     – `POST` - 在訊息本文中顯示表單資料

       例如，輸入：`POST@http://www.oracle.com/`

     當您輸入 XML 內容時，可以按一下 ，以將檔案匯入「XML 內容」區域，而且您可以按一下 ，以將 XML 內容匯出至 Microsoft Excel。
   - **層級 0 的標誌** - 無論 URL 是否僅適用於區域的層級 0 子代。

     例如，如果已針對可鑽研區域 `DESCENDANTS("Market"),@CHILDREN(Qtr1)` 啟用「層級 0 標誌」，則 URL 適用於 "Qtr1" 之所有月份期間 "Market" 的所有狀態，並適用於剩餘維度之間所有層級 0 的成員。
   - **區域** - 資料庫的成員規則定義區域應該允許使用特定 URL 進行鑽研

     您使用一或多個維度之成員的成員規格來定義可鑽研地區。使用您用於定義安全性篩選條件之相同的 Oracle Essbase 成員集合計算語言來定義成員規格。例如，以下就是有效的成員規格，代表於 "Qtr1" 的月份期間，在東部地區中除了 "New York" 之外的所有州：`@REMOVE(@DESCENDANTS("Eastern Region"), "New York"), @CHILDREN(Qtr1)`。

     若要新增地區，請按一下**新增地區**。
6. 按一下**儲存**，然後按一下**確定**。