Multiplan existing table changes

Sunday, January 24, 2016   7:00 AM

Multiplan DB Changes

Tuesday, January 12, 2016   9:50 AM

Tellus multiplan feature

UPDATE: Updated for Instances.

Multiplan Phase 1:
Each Cron job has one test plan.
One instance corresponds to one (TellusTestPlanID and JustifierConfigName) combination.

Multiplan Phase 2:
Each Cron job has multiple test plans.
One instance corresponds to multiple (TellusTestPlanID and JustifierConfigName) combinations.

In both phases, a row in CalendarJobs table corresponds to one Cron job.

Database Changes

It reuses the following tables, but with a different interpretation:
- SchedulerTestGroups has job start/end datetimes
- CalendarEvent has the job configurations (business fields)
- CalendarParent has the recurrence configuration (RecurType, RecurEndDate, etc.)

It moves pre-multiplan business fields from CalendarParent to CalendarEvent for series (recurrence) job configuration.

It adds CalendarInstances table.

Note: The Multiplan columns EndDateTime and StartDate are no longer needed in SchedulerTestGroups table, as groups are no longer used for date-time display in calendar. The Start and Finish date-times are used for Calendar date-time display.

calendar/php/Multiplan.php:
Business fields are defined Multiplan::$jobFieldDefs.
Recurrence fields are defined in Multiplan::$recurrenceFieldDefs

Deprecated (data will be migrated to Multiplan):
CalendarEventTestCaseAssociations
CalendarParentTestCaseAssociations
CalendarEventChunks

calendar/php/CustomDef.php is deprecated.

Table join relationships

Notation:
```
Table
  +(ForeignKey)-->ForeignTable(ForeignKey)
```

To get groups with test plan+cases:

```
SchedulerTestGroups
  + (InstanceID)-->CalendarInstances(InstanceID)
                +(EventID)-->CalendarEvent(EventID)
                +(ParentID)-->CalendarParent(ParentID)
```

```
    +(GroupID)-->CalendarGroupPlanAccociations
               +(PlanConfigID)-->CalendarEventPlanConfig(PlanConfigID)
                              +(PlanConfigID)-->CalendarEventPlanConfigCaseAssociations(PlanConfigID)
```

To get series with test plan+cases:

```
CalendarEvent(EventID)
    +(ParentID)-->CalendarParent(ParentID)
    +(EventID)-->CalendarEventPlanConfig(EventID)
                    +(PlanConfigID)-->CalendarEventPlanConfigCaseAssociations(PlanConfigID)
```

Relationship Details

- SchedulerTestGroups.InstanceID links to CalendarInstances(InstanceID)
- CalendarInstances.EventID links to CalendarEvent(EventID) for the business fields (WindowsVersion, etc.)
- CalendarInstances.SeriesID links CalendarEvent(EventID) for the *series* (recurrencing) business fields
- An "exceptional" instances of recurrence has one row in CalendarEvent for the exceptional data, and another for the series.
- The row in CalendarEvent for the series has flag IsSeries = 1
- CalendarEvent.ParentID links to CalendarParent(ParentID) for the recurrence definition (RecurType, RecurEndDate, etc.)
- "Non-exceptional" recurrence has same EventID as SeriesID (SchedulerTestGroups.EventID == SchedulerTestGroups.SeriesID); both refer to *series* job configuration
- "Exceptional" recurrence instance: SchedulerTestGroups.EventID and SeriesID are not NULL; SchedulerTestGroups.EventID != SchedulerTestGroups.SeriesID
- One time job: SchedulerTestGroups.EventID is not NULL, SeriesID is NULL

Note: StartDate and StartTime columns of CalendarEvent are no longer used.

Example SQL

Multiplan get events SQL examples
Multiplan SWP-GREEN SQL examples

Multiplan get events SQL examples

Tuesday, January 12, 2016   10:58 AM

Example PHP Code

`CalendarModel::getFullEvent($instanceID, $dbh)` gets full event with test plan and test case hierarchy for one single event instance.

Note: It is now using JustifierTestPlanCache and JustifierTestCaseCache tables to get JustifierTestPlanName and JustifierTestCaseName.

Multiplan Phase 1:
    Each Cron job has one test plan.
    One instance corresponds to one (TellusTestPlanID and JustifierConfigName) combination.

SQL example for getting events with test plans and test cases hierarchy

```sql
-- Multiplan groups with series, test plans, cases
SELECT
    g.GroupID
    , g.CronID
    , g.StartDateTime AS GroupStart
    , g.Duration AS GroupDuration
    , i.InstanceID
    ,CONVERT(varchar, i.Start, 126) AS Start
    ,CONVERT(varchar, i.Finish, 126) AS Finish
    ,i.EventID
    ,i.SeriesID
    ,e.ParentID, e.IsMultiplan, e.IsSeries, e.IsException, e.Title, e.AllDay, e.Priority,
e.Duration, e.LastModified, e.ExecHost, e.IsDev, e.RunNow, e.ScheduleType,
e.EmailRecipient, e.BuildID, e.BuildVersion, e.BitVersion, e.WindowsVersion,
e.OfficeVersion, e.LanguageName, e.VMSUserID, e.Stage, e.PrebuildName, e.HTMLReports,
e.HTMLReportsOS, e.CaccessCode, e.TickerListRequired, e.TickerList, e.BrowserType,
e.BrowserVersion, e.TestServer, e.ControlServer, e.JustifierSystemID, e.JustifierBuildID,
e.JustifierLibraryID, e.JustifierProductID, e.TestEnvironment, e.SeleniumGridHub,
e.Platform, e.PlatformName, e.PlatformVersion, e.DeviceName, e.SGBrowserName,
e.SGBrowserVersion, e.WebStage, e.TestCaseStage, e.TestCasePrebuildName, e.FixedTime,
e.IsRegressionJob, e.IsLateRegressionJob, e.BaseBuild, e.TestBuild, e.Gherkin
    ,r.RecurStartDate, r.StartTime, r.RecurType, r.RecurEvery, r.Day0, r.Day1, r.Day2,
r.Day3, r.Day4, r.Day5, r.Day6, r.Day7, r.Day8, r.Day9, r.Day10, r.Day11, r.Day12,
r.Day13, r.RecurMonthlyType, r.RecurMonthlyNth, r.RecurMonthlyDOW, r.RecurYearlyMonth,
r.RecurEndType, r.RecurEndDate, r.RecurEndNum, r.GenStart, r.GenLeft, r.IsGenComplete,
r.ScheduledStart

    ,gpa.PlanConfigID
    ,pc.TellusTestPlanID
    ,pc.JustifierTestPlanID
    ,pc.JustifierConfigName
    ,pcca.TellusTestCaseID
    ,jtca.JustifierTCID AS JustifierTestCaseID
    ,c.CategoryName AS TellusTestPlanName
    ,t.TestName AS TellusTestCaseName
    ,jtpc.JustifierTestPlanName
    ,jtcc.JustifierTestCaseName
    ,m.ProjectType
    ,m.SubProjectType
    ,m.InstallerNeeded
FROM SchedulerTestGroups g
JOIN CalendarInstances i
```

```sql
        ON g.InstanceID = i.InstanceID
JOIN CalendarEvent e
    ON i.EventID = e.EventID
LEFT JOIN CalendarEvent s
    ON i.SeriesID = s.EventID
LEFT JOIN CalendarParent r
    ON s.ParentID = r.ParentID
LEFT JOIN CalendarGroupPlanAssociations gpa
    ON g.GroupID = gpa.GroupID
LEFT JOIN CalendarEventPlanConfig pc
    ON gpa.PlanConfigID = pc.PlanConfigID
LEFT JOIN CalendarEventCaseAssociations pcca
    ON pc.PlanConfigID = pcca.PlanConfigID
LEFT JOIN JustifierTestCaseAssociations jtca
    ON pcca.TellusTestCaseID = jtca.TellusTCID
LEFT JOIN SchedulerTestCategories c
    ON pc.TellusTestPlanID = c.CategoryID
LEFT JOIN TcSelMapping m
    ON c.CategoryID = m.CategoryID
LEFT JOIN SchedulerTests t
    ON pcca.TellusTestCaseID = t.TestID
LEFT JOIN JustifierTestPlanCache jtpc
    ON c.JustifierID = jtpc.JustifierTestPlanID
LEFT JOIN JustifierTestCaseCache jtcc
    ON jtca.JustifierTCID = jtcc.JustifierTestCaseID
WHERE e.IsMultiplan = 1
    AND g.IsDeleted = 0
     AND i.IsDeleted = 0
    AND e.IsDeleted = 0
    AND (s.IsDeleted = 0 OR s.IsDeleted IS NULL)
    AND (r.IsDeleted = 0 OR r.IsDeleted IS NULL)
    AND (gpa.IsDeleted = 0 OR gpa.IsDeleted IS NULL)
    AND (pc.IsDeleted = 0 OR pc.IsDeleted IS NULL)
```

That returns hierarchy: EventID > InstanceID > GroupID > TellusTestPlanID+JustifierConfigName > TellusTestCaseID.

Multiplan SWP-GREEN SQL examples

Tuesday, January 12, 2016   10:47 AM

Example SWP-GREEN get recurrences with test plans and test cases (hierarchical data)

The `CalendarModel::getRecurrenceToDo method` will get IsRegressionJob recurrences:

```
require_once __DIR__ . "/../calendar/php/Multiplan.php";
use \Tellus\Util\DbUtil;
$dbh = DbUtil::connect();
$recurrencesToDo = Multiplan::getRecurrenceToDo($dbh);
$dbh = null;
```

It returns test plan and test cases hierarchy:

```
EventID: {value},
Other job fields...,
TestPlans: [{
    TellusTestPlanID: {value},
    TellusTestPlanName: {value},
    JustifierTestPlanID: {value},
    JustifierTestPlanName: {value},
    JustifierConfigName: {value},
    ProjectType: {value}
    SubProjectType: {value}
    InstallerNeeded: 0|1
    TestCases: [{
        TellusTestCaseID: {value},
        TellusTestCaseName: {value},
        JustiferTestCaseID: {value},
        JustiferTestCaseName: {value},
    }]
}]
```

It executes the following SQL:

```
-- Get recurrence to do
SELECT e.EventID
    ,e.IsMultiplan, e.IsException, e.IsSeries, e.ParentID, e.Title, e.AllDay, e.Priority,
e.Duration, e.LastModified, e.ExecHost, e.IsDev, e.RunNow, e.ScheduleType, e.EmailRecipient,
e.BuildID, e.BuildVersion, e.BitVersion, e.WindowsVersion, e.OfficeVersion, e.LanguageName,
e.VMSUserID, e.Stage, e.PrebuildName, e.HTMLReports, e.HTMLReportsOS, e.CaccessCode,
e.TickerListRequired, e.TickerList, e.BrowserType, e.BrowserVersion, e.TestServer,
e.ControlServer, e.JustifierSystemID, e.JustifierBuildID, e.JustifierLibraryID,
e.JustifierProductID, e.TestEnvironment, e.SeleniumGridHub, e.Platform, e.PlatformName,
e.PlatformVersion, e.DeviceName, e.SGBrowserName, e.SGBrowserVersion, e.WebStage,
e.TestCaseStage, e.TestCasePrebuildName, e.FixedTime, e.IsRegressionJob, e.IsLateRegressionJob,
e.BaseBuild, e.TestBuild, e.Gherkin, e.SITEnvironment
    ,r.ScheduledStart, r.RecurStartDate, r.RecurStartTime, r.RecurType, r.RecurEvery, r.Day0,
r.Day1, r.Day2, r.Day3, r.Day4, r.Day5, r.Day6, r.Day7, r.Day8, r.Day9, r.Day10, r.Day11,
r.Day12, r.Day13, r.RecurMonthlyType, r.RecurMonthlyNth, r.RecurMonthlyDOW, r.RecurYearlyMonth,
r.RecurEndType, r.RecurEndDate, r.RecurEndNum, r.GenStart, r.GenLeft, r.IsGenComplete
    ,epc.PlanConfigID
    ,epc.JustifierTestPlanID
    ,epc.JustifierConfigName
    ,epc.TellusTestPlanID
    ,epcca.TellusTestCaseID
    ,jtca.JustifierTCID AS JustifierTestCaseID
    ,c.CategoryName AS TellusTestPlanName
    ,t.TestName, t.TestPath, t.BasePath, t.ProdTestPath, t.ProdBasePath
    ,jtpc.JustifierTestPlanName
    ,jtcc.JustifierTestCaseName
    ,m.ProjectType
    ,m.SubProjectType
    ,m.InstallerNeeded
```

```
FROM CalendarEvent e
JOIN CalendarParent r
    ON r.ParentID = e.ParentID
JOIN CalendarEventPlanConfig epc
    ON epc.EventID = e.EventID
JOIN CalendarEventCaseAssociations epcca
    ON epcca.PlanConfigID = epc.PlanConfigID
JOIN JustifierTestCaseAssociations jtca
    ON jtca.TellusTCID = epcca.TellusTestCaseID
JOIN SchedulerTestCategories c
    ON c.CategoryID = epc.TellusTestPlanID
JOIN TcSelMapping m
    ON m.CategoryID = epc.TellusTestPlanID
JOIN SchedulerTests t
    ON t.TestID = epcca.TellusTestCaseID
LEFT JOIN JustifierTestPlanCache jtpc
    ON jtpc.JustifierTestPlanID = c.JustifierID
LEFT JOIN JustifierTestCaseCache jtcc
    ON jtcc.JustifierTestCaseID = jtca.JustifierTCID
WHERE e.IsMultiplan = 1
    AND e.IsSeries = 1
        AND r.RecurType = 'B'
    AND e.IsDeleted = 0
    AND r.IsDeleted = 0
    AND (epc.IsDeleted = 0 OR epc.IsDeleted IS NULL)
ORDER BY e.EventID, epc.TellusTestPlanID, epc.JustifierConfigName, epcca.TellusTestCaseID
```

The records are in the order, (EventID, TellusTestPlanID, JustifierConfigName, TellusTestCaseID. So one would need to detect that the row is a new TellusTestPlanID, for example.

Multiplan view - CalendarGroupsWithTestCases

Tuesday, February 16, 2016   10:59 AM

```
USE [tellus]
GO

/****** Object:  View [dbo].[MultiplanJobs]    Script Date: 3/17/2016 10:58:54 AM ******/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO


CREATE VIEW [dbo].[CalendarGroupsWithTestCases] AS
(
-- Multiplan groups with series, test plans, cases
SELECT
    g.GroupID
     , g.CronID
     , g.StartDateTime AS GroupStart
     , g.Duration AS GroupDuration
    , i.InstanceID
    ,CONVERT(varchar, i.Start, 126) AS Start
    ,CONVERT(varchar, i.Finish, 126) AS Finish
    ,i.EventID
    ,i.SeriesID
    ,e.ParentID, e.IsMultiplan, e.IsSeries, e.IsException, e.Title, e.AllDay, e.Priority,
e.Duration, e.LastModified, e.ExecHost, e.IsDev, e.RunNow, e.ScheduleType,
e.EmailRecipient, e.BuildID, e.BuildVersion, e.BitVersion, e.WindowsVersion,
e.OfficeVersion, e.LanguageName, e.VMSUserID, e.Stage, e.PrebuildName, e.HTMLReports,
e.HTMLReportsOS, e.CaccessCode, e.TickerListRequired, e.TickerList, e.BrowserType,
e.BrowserVersion, e.TestServer, e.ControlServer, e.JustifierSystemID, e.JustifierBuildID,
e.JustifierLibraryID, e.JustifierProductID, e.TestEnvironment, e.SeleniumGridHub,
e.Platform, e.PlatformName, e.PlatformVersion, e.DeviceName, e.SGBrowserName,
e.SGBrowserVersion, e.WebStage, e.TestCaseStage, e.TestCasePrebuildName, e.FixedTime,
e.IsRegressionJob, e.IsLateRegressionJob, e.BaseBuild, e.TestBuild, e.Gherkin
     ,r.RecurStartDate, r.StartTime, r.RecurType, r.RecurEvery, r.Day0, r.Day1, r.Day2,
r.Day3, r.Day4, r.Day5, r.Day6, r.Day7, r.Day8, r.Day9, r.Day10, r.Day11, r.Day12,
r.Day13, r.RecurMonthlyType, r.RecurMonthlyNth, r.RecurMonthlyDOW, r.RecurYearlyMonth,
r.RecurEndType, r.RecurEndDate, r.RecurEndNum, r.GenStart, r.GenLeft, r.IsGenComplete,
r.ScheduledStart

    ,gpa.PlanConfigID
    ,pc.TellusTestPlanID
    ,pc.JustifierTestPlanID
    ,pc.JustifierConfigName
    ,pcca.TellusTestCaseID
    ,jtca.JustifierTCID AS JustifierTestCaseID
    ,c.CategoryName AS TellusTestPlanName
    ,t.TestName AS TellusTestCaseName
    ,jtpc.JustifierTestPlanName
    ,jtcc.JustifierTestCaseName
    ,m.ProjectType
    ,m.SubProjectType
    ,m.InstallerNeeded
FROM SchedulerTestGroups g
JOIN CalendarInstances i
    ON g.InstanceID = i.InstanceID
JOIN CalendarEvent e
    ON i.EventID = e.EventID
```

```sql
LEFT JOIN CalendarEvent s
    ON i.SeriesID = s.EventID
LEFT JOIN CalendarParent r
    ON s.ParentID = r.ParentID
LEFT JOIN CalendarGroupPlanAssociations gpa
    ON g.GroupID = gpa.GroupID
LEFT JOIN CalendarEventPlanConfig pc
    ON gpa.PlanConfigID = pc.PlanConfigID
LEFT JOIN CalendarEventCaseAssociations pcca
    ON pc.PlanConfigID = pcca.PlanConfigID
LEFT JOIN JustifierTestCaseAssociations jtca
    ON pcca.TellusTestCaseID = jtca.TellusTCID
LEFT JOIN SchedulerTestCategories c
    ON pc.TellusTestPlanID = c.CategoryID
LEFT JOIN TcSelMapping m
    ON c.CategoryID = m.CategoryID
LEFT JOIN SchedulerTests t
    ON pcca.TellusTestCaseID = t.TestID
LEFT JOIN JustifierTestPlanCache jtpc
    ON c.JustifierID = jtpc.JustifierTestPlanID
LEFT JOIN JustifierTestCaseCache jtcc
    ON jtca.JustifierTCID = jtcc.JustifierTestCaseID
WHERE e.IsMultiplan = 1
    AND g.IsDeleted = 0
     AND i.IsDeleted = 0
    AND e.IsDeleted = 0
    AND (s.IsDeleted = 0 OR s.IsDeleted IS NULL)
    AND (r.IsDeleted = 0 OR r.IsDeleted IS NULL)
    AND (gpa.IsDeleted = 0 OR gpa.IsDeleted IS NULL)
    AND (pc.IsDeleted = 0 OR pc.IsDeleted IS NULL)

)


GO
```