



Louis Berger



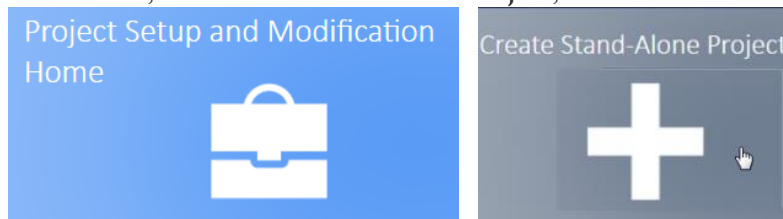
# New Project Setup From New Project Wizard

Summer, 2017

[louisberger.com](http://louisberger.com)

## Overview

This document describes the New Project Wizard for a Stand-Alone project. When the user selects **Project Setup and Modification**, then **Create Stand Alone Project**,



the New Project Wizard opens.

## New Project Wizard

Basic Information

PROJECT # 2042054

OPPORTUNITY \*Please select a value to load opportunity information Pull down arrow to choose an opportunity

PROJECT SHORT NAME\* ?  Long & short names fill in from opportunity

PROJECT LONG NAME ?

PROJECT TYPE\* \*Please select a value ▾

OWNING ORG\* ?

PROJECT MANAGER (PM)\* None Available ▾

CONTRACT CURRENCY\* ? \*USD (\$) - U. S. Dollar ▾

Project Number is automatically generated by incrementing the last new project number.

### Opportunity Drpdown

Opportunities populate the opportunity dropdown using

- PDS\_Project\_Direct\_GetBergerWinOpportunities\_With\_Practice for Company 20
- PDS\_Project\_Direct\_GetBergerWinOpportunities\_Without\_Practice for all other companies

PDS\_Project\_Direct\_GetBergerWinOpportunities\_Without\_Practice.sql follows on the next 3 pages.

Green font between /\* \*/ delimiters are comments

All SQL code is entered here as screenshots to preserve the SQL syntax colors

## PDS\_Project\_Direct\_GetBergerWinOpportunities\_Without\_Practice.sql

```

1 ALTER PROCEDURE [dbo].[PDS_Project_Direct_GetBergerWinOpportunities_Without_Practice] @SQLDBIUserId int
2     , @ParentPDSProjectId int, @PDSProjectClassificationId int, @Company varchar(10)
3 AS
4 BEGIN
5 /*If this is a TO we are trying to create, get the IQC parent number
6     to be used when filtering available opportunities.*/
7 DECLARE @IQCProjectNumber varchar(30);
8 IF @ParentPDSProjectId IS NOT NULL AND @PDSProjectClassificationId = 102 --If task order
9 BEGIN
10 SELECT @IQCProjectNumber = Project_Number --get parent (ongoing, contractual) project number
11 FROM PDS_Project
12 WHERE PDS_Project_Id = [dbo].[PDS_GetTopLevelProjectForProject](@ParentPDSProjectId)
13         AND PDS_Project_Classification_Id = 100;--IQC
14 END
15
16 DECLARE @SQL nvarchar(max);
17 /*Else- get available, applicable opportunities*/
18 SET @SQL = N'
19 /*all part of N*/
20 SELECT O.OpportunityId, O.Opportunity,
21 /*Opportunity Display=Opportunity "number"-Opportunity Name, to a max of 100 characters
22     Examples show various formats for Opportunity number:
23         004252-ARB Base Support OPP-10212-SaudiArabia Consultancy
24         CPx_00756-Update GAP Anal at NSA Annapolis and.... */
25     LEFT(O.Opportunity + ' ' - '' + O.Name, 100) OpportunityDisplay,
26 /*Opportunity Name fom Opportunity table, CostPoint organization from PDS_Project (BPOM)
27     ,Contract Cost Type from PDS_Project_BergerWin_Contract_Cost_Type - see line 43 below */
28     O.Name, BPOM.CPx100_ORG_Id, CCT.CPx100_PROJ_TYPE_Id,
29     CA_E.CPx100_EMPL_Id AS Contract_Administrator_Id,
30     PM_E.CPx100_EMPL_Id AS Project_Manager_Id
31 /*ConnectionString communicates with Bergerwin, the database that holds the Opportunity table*/
32 FROM {ConnectionString}.Opportunity O
33 /*OCF is a BergerWin View - CustomConnectionString is to Bergerwin View*/
34 --match opportunityId
35 INNER JOIN {CustomConnectionString}.CPx_OpportunityCustomTabField OCF ON OCF.OpportunityID = O.OpportunityID
36 --match Lead Division
37 INNER JOIN PDS_Project_BP_Org_Mapping BPOM ON BPOM.GovWin_Lead_Division_Code = OCF.CustLeadDivision
38 --CP_Org = BP.CP_Org;Company is input variable
39 INNER JOIN CPx100_ORG ORG ON ORG.CPx100_ORG_Id = BPOM.CPx100_ORG_Id AND ORG.COMPANY_ID = @SQL_Company
40     AND ORG.Eligible_For_Direct_Project = 1 --org can create Direct Project
41 /*CT: 01=Basic Ordering Agreement (BOA) 02-Direct Contract 03-Framewk
42     04-Master Contract 05-Task Order 06-Undetermined */
43 INNER JOIN PDS_Project_BergerWin_Contract_Type CT ON CT.BergerWin_Contract_Type_Code = OCF.CustContractType
44 /*Opportunity Stage significance:
45     Only Opportunities at Stage 08 and 09 are eligible to be used as new projects.
46     08-Awarded (Contract Not Signed) - Stage 08 are considered at risk for project assignment
47     09-Awarded (Contract Signed)-awaiting assignment to a PM, 10-Awarded (Active Project) */

```

## UndentPDS\_Project\_Direct\_GetBergerWinOpportunities\_Without\_Practice.sql

```

48 INNER JOIN PDS_Project_BergerWin_Opportunity_Status OS ON OS.BergerWin_Status_Code = 0.Stage
49 /*----CCT codes ----- 137,140,141 are CPx100_Proj_TypeIds-----*/
50 /*01=Cost Plus/Award Fee-141 02=Cost Plus/Fixed Fee-141 03=Firm Fixed Price (FFP)-140
51 04=Lump Sum -140 05=Performance Based Contract-140 06=TM-137 */
52 LEFT JOIN PDS_Project_BergerWin_Contract_Cost_Type CCT ON
53 CCT.BergerWin_Contract_Cost_Type_Code = OCF.CustContractCostType
54 /*-----Stakeholders-----*/
55 LEFT JOIN CPx100_EMPL_CA_E ON CA_E.EMPL_ID = OCF.ID_custContractAdministrator
56 AND CA_E.COMPANY_ID = @SQL_Company AND CA_E.S_EMPL_STATUS_CD = 'ACT'
57 LEFT JOIN CPx100_EMPL_PM_E ON PM_E.EMPL_ID = O.ProjMgr
58 AND PM_E.COMPANY_ID = @SQL_Company AND PM_E.S_EMPL_STATUS_CD = 'ACT'
59 WHERE OCF.CustContractType IS NOT NULL AND OS.Eligible_For_Direct_Project = 1
60 AND
61 ( /*100=IQC -parent to Task Orders; 101=Stand-Alone; 102=Task order */
62 (@SQL_PDSProjectClassificationId IN (100, 101) AND OCF.CustSelectIQC IS NULL) OR
63 (@SQL_PDSProjectClassificationId = 102 AND OCF.CustSelectIQC IS NOT NULL
64 AND @SQL_IQCProjectNumber IS NOT NULL AND OCF.CustSelectIQC = @SQL_IQCProjectNumber)
65 )
66 AND NOT EXISTS
67 (
68 SELECT 1 FROM PDS_Project P --not already a project
69 WHERE P.BergerWin_Opportunity_Id = 0.OpportunityId
70 AND P.PDS_Project_Classification_Id IN (100, 101, 102)
71 )
72 AND
73 (
74 (
75 @SQL_PDSProjectClassificationId = 100 AND --IQC
76 EXISTS
77 (
78 SELECT 1
79 FROM AB_User_Scope US
80 /* UST: 103-Contract Admin ..105-Project Mgr..109-Div Mgr...118-Super Contract Admin*/
81 INNER JOIN AB_User_Scope_Type UST ON UST.AB_User_Scope_Type_Id = US.AB_User_Scope_Type_Id
82 AND UST.Name IN
83 ('Org Assigned - Contract Administrator', 'Org Assigned - Super Contract Administrator')
84 /*user=input user variable */
85 WHERE US.AB_User_Id = @SQL_SQLDBIUserId AND US.AB_Member_Id = ORG.AB_Group_Id
86 )
87 )
88 OR
89 (
90 @SQL_PDSProjectClassificationId IN (101, 102) -- 101=Stand-Alone; 102=Task order
91 AND EXISTS
92 (
93 SELECT 1
94 FROM AB_User_Scope US

```

## PDS\_Project\_Direct\_GetBergerWinOpportunities\_Without\_Practice.sql

```

95     INNER JOIN AB_User_Scope_Type UST ON UST.AB_User_Scope_Type_Id = US.AB_User_Scope_Type_Id
96     /* UST: 116-PM, 117-Super PM*/
97     AND UST.Name IN ('Org Assigned - Project Manager', 'Org Assigned - Super Project Manager')
98     /*Where user (identified by AB_User_Id) is PM or Super PM and
99     belongs to organization associated with AB_Group-Id
100     Examples: ORG_ID 20; ORG_NAME Louis Berger US Inc; AB_Group_Id -113
101     ORG_ID 20.C; ORG_NAME LB US Segment C; AB_Group_Id -154*/
102     WHERE US.AB_User_Id = @SQL_SQLDBIUserId AND US.AB_Member_Id = ORG.AB_Group_Id
103 )
104 )
105 )
106 AND EXISTS
107 (
108     SELECT 1
109     FROM BergerWin_Contract_Type_Project_Classifications CTPC
110     /* CT.PDS_Project_BergerWin_Contract_Type_Id 100-BOA 101-Direct Contract 102-Framework
111     103-Master Contract 104-Task Order 105-undetermined*/
112     WHERE CTPC.PDS_Project_BergerWin_Contract_Type_Id = CT.PDS_Project_BergerWin_Contract_Type_Id
113     /*CTPC table PDS_Project_Classification_Id (100, 101, or 102)
114     matches input variable PDSProjectClassificationId*/
115     AND CTPC.PDS_Project_Classification_Id = @SQL_PDSProjectClassificationId
116 )
117 /*Fill the Opportunity dropdown in the New Project Wizard*/
118 ORDER BY O.Opportunity, O.Name;';
119 /*N ends here */
120
121 /*ConnectionString accessing BergerWin from CPx */
122 DECLARE @ConnectionString varchar(300);
123 SET @ConnectionString = dbo.PDS_Project_GetBergerWinConnectionString();
124 SET @SQL = REPLACE(@SQL, '{ConnectionString}', @ConnectionString);
125
126 SET @ConnectionString = dbo.GetBergerWinCustomConnectionString();
127 SET @SQL = REPLACE(@SQL, '{CustomConnectionString}', @ConnectionString);
128 /*Execute N above - Pass User, Project Classification, IQCproject number, Company */
129 exec sp_executesql @SQL, N'@SQL_SQLDBIUserId int, @SQL_PDSProjectClassificationId int
130 ',@SQL_IQCProjectNumber varchar(30), @SQL_Company varchar(10)'
131 , @SQL_SQLDBIUserId = @SQLDBIUserId
132 , @SQL_PDSProjectClassificationId = @PDSProjectClassificationId
133 , @SQL_IQCProjectNumber = @IQCProjectNumber, @SQL_Company=@Company;
134 END
135

```

## Project Type

Populated by PDS\_CreateProjectWizard.ascx.cs

```
1  protected void ProjectTypeDropDownList_SelectedIndexChanged(object sender, EventArgs e)
2  { /* before selection, no checkbox shows or is checked */
3      this.RateCappedCheckbox.Checked = false;
4      this.UnitsCheckbox.Checked = false;
5      /* Select Project Type from drop down list.
6      Choices are CP (Cost Plus), TM (Time and Materials), LS (Lump Sum) */
7      var selectedProjectTypeId = int.Parse(ProjectTypeDropDownList.SelectedValue);
8
9      /* row for checkbox not visible */
10     this.RateCappedRow.Visible = false;
11     this.UnitsRow.Visible = false;
12     /*Call to GetProjectTypeOption with SelectedValue from ProjectTypeDropDownList as parameter*/
13     ProjectTypeOptions projectTypeOption = CPx_CP_Project_Type.GetProjectTypeOption(selectedProjectTypeId);
14     /*If SelectedProjectType=CP, show Rate Capped checkbox*/
15     if (projectTypeOption == ProjectTypeOptions.CPRateCappedOption)
16     {
17         this.RateCappedRow.Visible = true;
18     }
19     /*If SelectedProjectType=TM, show Units checkbox*/
20     else if (projectTypeOption == ProjectTypeOptions.TMUnitsOption)
21     {
22         this.UnitsRow.Visible = true;
23         /* Checkmark TM Units if company starts with 5:
24         50-Holdings Berger France, 51-Louis Berger SAS, 52-Louis Berger UK, 53-Louis Berger SRL (Romania),
25         54-Louis Berger International Design Center (Spain), 55-Louis Berger International SARL (France)*/
26
27         if(this.NewProjectCompany.StartsWith("5"))
28         {
29             this.UnitsCheckbox.Checked = true;
30         }
31     }
32     /*Update this view as specified by the above conditions*/
33     this.NewProjectUpdatePanel.Update();
34 }
35
```

## Owning Org

The PDS\_GetAvailableOrgsForProjectSetup stored procedure populates the Owning Org dropdown.

New Project Wizard

Basic Information

PROJECT # 2042060

OPPORTUNITY

PROJECT SHORT NAME\* ?

PROJECT LONG NAME ?

PROJECT TYPE\*

OWNING ORG\* ?

PROJECT MANAGER (PM)\*

CONTRACT CURRENCY\* ?

```

1 ALTER PROCEDURE [dbo].[PDS_GetAvailableOrgsForProjectSetup]
2     (@AB_User_Id int, @Company varchar(20), @PDS_Project_Classification_Id int)
3 AS
4 BEGIN
5 DECLARE @AccountingDivisionOrgLevel int;
6 /*Get the Accounting Division Level number because
7   Accounting Division is the Owning Division for a project */
8 SELECT @AccountingDivisionOrgLevel = CS.Accounting_Division_Org_Level
9 FROM Company_Setting CS
10 WHERE CS.Company = @Company;
11
12 IF @PDS_Project_Classification_Id = 103 /*B&P*/
13 BEGIN
14
15     SELECT CPx100_ORG_Id AS LookupValue, ORG_ID + ' -- ' + ORG_NAME AS LookupText
16     FROM CPx100_ORG O
17     ORDER BY LookupText;
18
19 END
20 ELSE
21 BEGIN /* not 103 */
22
23 WITH GetInitialRolesForCompany AS
24 (
25     SELECT R.Name AS Role_Name
26     FROM Company_Setting CS
27     /*Proj status for company */
28     INNER JOIN Company_Setting_Project_Classification_Initial_Statuses CSPC ON CS.Company_Setting_Id = CSPC.Company_Setting_Id
29     /*Proj statuses such as Pending CA/CP/CM/At_Risk submittal/review/Approval */
30     INNER JOIN PDS_Project_Status_Type_Responsible_Role PSTR ON PSTR.PDS_Project_Status_Type_Id = CSPC.PDS_Project_Status_Type_Id
31     /*Role-AB_Group_Id (below)*/
32     /*ContractAdmin-100 ContractMgr-101 PM-102 ProjAccountant-103... DM-107... COO-111... Read_Only-372... */
33     INNER JOIN Role R ON R.Role_Id = PSTR.Role_Id
34     /*Company & Project Classification input */
35     /* proj classes: 100 IQC 101 stand alone 102 Task Order 103 B&P */
36     WHERE CS.Company = @Company AND CSPC.PDS_Project_Classification_Id = @PDS_Project_Classification_Id
37 )
38 SELECT DISTINCT CPx100_ORG_Id AS LookupValue, ORG_ID + ' -- ' + ORG_NAME AS LookupText
39 FROM AB_User_Scope US
40 /*scope_type= employee title */
41 INNER JOIN AB_User_Scope_Type UST ON UST.AB_User_Scope_Type_Id = US.AB_User_Scope_Type_Id
42 /*AB_Group_Id & AB_Member_Id explained with Role above; begins w - (dash)*//* Company eligible for Direct Project */
43 INNER JOIN CPx100_ORG O ON O.AB_Group_Id = US.AB_Member_Id AND O.COMPANY_ID = @Company AND O.Eligible_For_Direct_Project = 1
44 CROSS APPLY GetInitialRolesForCompany IR
45 /* User role determines what user sees on New Project Wizard */
46 WHERE AB_User_Id = @AB_User_Id AND UST.Name = 'Org Assigned - ' + IR.Role_Name
47
48 ORDER BY LookupText; /*order by ORG_ID + ' -- ' + ORG_NAME */
49 END
50 END

```