

2014 協同產品設計實習報告

機械設計工程系二甲

April 23, 2014

Contents

前言	4
Pandoc 手冊	4
協同產品設計實習專案 (2ag1)	5
組員：	5
w8 考試	5
w9 考試	6
w12	8
網際鼓式煞車設計 (2ag2)	8
程式設計架構	8
結果與討論	9
W12 任務 (2ag3)	9
摘要報告	9
網際鼓式煞車設計 (2ag4)	11
網際四連桿機構設計 (2ag5)	12
程式設計架構	12

結果與討論	12
網際鼓式煞車設計 (2ag6)	12
程式設計架構	12
結果與討論	13
網際鼓式煞車設計 (2ag7)	13
程式設計架構	13
結果與討論	13
網際四連桿機構設計 (2ag8)	13
程式設計架構	13
結果與討論	13
網際鼓式煞車設計 (2ag9)	13
程式設計架構	14
結果與討論	14
分組工作日誌	14
分組評分	14
注意事項	23
各組 push 資料確認事項	24
同步雲端網站	24
2014scda 分組日誌	24
各週課後註記	27

Creo 討論區	27
Collaborative Engine for Distributed Mechanical Design	108
Abstract	108
cd2ag10 報告 (2ag10)	118
第十組組員	118
第八週考試摘要 (2ag10)	119
第八週報告 (2ag10)	121
第九週考試摘要 (2ag10)	122
第九週報告 (2ag10)	123
第十二週報告 (2ag10)	126
第十二週評分	128
網際鼓式煞車設計 (2ag11)	128
程式設計架構	128
結果與討論	128
網際鼓式煞車設計 (2ag12)	128
程式設計架構	128
結果與討論	128
協同產品設計實習專案 (2ag13)	129
組員：	129
程式設計架構	129

結果與討論	130
網際鼓式煞車設計 (2ag15)	130
程式設計架構	130
結果與討論	130
網際鼓式煞車設計 (2ag16)	130
程式設計架構	130
結果與討論	130
第八週摘要報告	130
第九週摘要報告	131
網際鼓式煞車設計 (2ag18)	131
程式設計架構	132
結果與討論	132
網際鼓式煞車設計 (2ag21)	132
程式設計架構	132
結果與討論	132
網際 OpenJSCAD 程式設計 (coursemdetw)	132
設計程式架構	132
結果與討論	132

前言

協同產品設計實習課程目標¹

¹這是註解的用法.

Here is an inline note.²

學習協同產品設計流程與環境的基本原理與架構.

學習如何在網際專案管理系統的協助下, 進行協同產品設計.

學習如何在協同設計流程中, 進行有效率的工程設計表達與產品資料管理.

延續程式語言與電腦輔助設計實習課程, 學習如何建構協同產品設計環境所需的工具.

Pandoc 手冊

<http://johnmacfarlane.net/pandoc/README.html>

footnotes

tables

flexible ordered lists

definition lists

fenced code blocks

superscript

subscript

strikeout

title blocks

automatic tables of contents

embedded LaTeX math

citations

markdown inside HTML block elements

協同產品設計實習專案 (2ag1)

組員：

40123101

40123102

40123132

OpenShift 網站: <http://2014cdag1-cadp13ag6.rhcloud.com/>

²Inlines notes are easier to write, since you don't have to pick an identifier and move down to type the note.

w8 考試

1. 請寫一個執行時可以列出 9×9 乘法表的網際 Python 程式, 然後 Push 到個人 bitbucket 空間, 而且同步指到 OpenShift 個人帳號上執行.

- 程式碼:

```
for x in range(0,10):  
    for y in range(0,10):  
        print(x, '*', y, '=', x*y)
```

- 解題過程:

打完程式後, 之後我把它上傳到 bitbucket, git add . → git commit -m "99" → git push, 之後為了要把 99 乘法表顯示在 openshift CMSimply 上, 進去 openshift CMSimply 的資料夾→ wsgi 資料夾→ application 檔案, 在裡面新增一個 class 的格式, 之後連線 FileZilla, 把 application 檔案覆蓋到 app-root/runtime/repo/wsgi 的 application 檔案下, 之後進去 openshift CMSimply 的網站, 就顯示 99 乘法表了。

- 解題心得:

雖然要打一個 99 乘法表的程式很快, 但要如何顯示在遠端的網頁上, 思考了很久, 在課堂後請教了 TA, 最後應用 cherrypy 的方式, 在 application 檔案, 新增一個 class 的格式, 就完成了。

2. 請將上述執行過程錄為 flv 後, 上傳到個人的 Vemeo 空間中, 並將網址回報到各組網站 (dokuwiki 與 CMSimply) 與報告中, 並且將相關心得與報告連結登錄到 wiki.mde.tw 第八週的分組頁面中.

- Bitbucket 連結: <https://bitbucket.org/40123102/40123102cmsimply/src>
- Vemeo 空間: <https://vimeo.com/user26935042/videos>
- copy 空間: <https://copy.com/XohJLdUBRJdk>
- dokuwiki 網站: <https://40123102cdg1dokuwiki-cadp13ag6.rhcloud.com/doku.php?id=start>
- CMSimply 網址: <http://40123102cdg1-cadp13ag6.rhcloud.com/w8test/>

3. 請在各組的雲端 dokuwiki 中, 根據下列 40 個帳號與密碼, 新增對應的使用者帳號與密碼後, 將雲端網址登錄在 wiki.mde.tw 各組第八週頁面中, 並說明操作過程與心得後, 將心得整理成 pdf 後繳交到 course@mde.tw.

- 解題過程:

程式做不出來, 只好一個一個建立。

- 解題心得:

想很久, 還是無法想出程式, 只好用最勤勞的方法, 一個一個建立, 唉!!!

w9 考試

1. 請寫一個執行時可以列出以十為底對數表的網際 Python 程式, 然後 Push 到個人 bitbucket 空間, 而且同步指到 OpenShift 個人帳號上執行.

- 程式碼:

```
import math
count = 0
text='%04d'
for i in range(100,200+10):
    print(text%round(math.log(i/100,10)*10000,0),end='-')
    count += 1
    if(count == 10):
        print()
        count = 0
```

- 解題過程:

打完程式後, 有顯示出來, 但無法顯示表單在 CMSimply 網址上, 只能一個一個慢慢打。

- 解題心得:

想了很久, 還是無法, 只會打程式, 只好把顯示出來的數字一個一個加上, 慢慢打。

2. 請在個人的 OpenShift 平台上建立一個能夠列印出與九九乘法表結果完全相同的網際程式, 接著在乘法表上端加上兩個輸入表單, 讓使用者輸入兩個整數, 按下送出鍵後, 程式會列出以此兩個整數為基底的乘法表, 例如: 若兩個欄位都輸入: 9, 則列出九九乘法表, 若輸入 9, 20, 則列出 9×20 的乘法表.

- 解題過程:

打完程式後, 之後我把它上傳到 bitbucket, git add . → git commit -m "w9_2" → git push, 之後為了要把 99 乘法表顯示在 openshift CMSimply 上, 進去 openshift CMSimply 的資料夾→ wsgi 資料夾→ application 檔案, 在裡面新增一個 class 的格式, 之後連線 FileZilla, 把 application 檔案覆蓋到 app-root/runtime/repo/wsgi 的 application 檔案下, 之後進去 openshift CMSimply 的網站, 輸入 9, 20, 則顯示 9×20 的乘法表.

- 解題心得:

這題跟上週小考差不多, 只是還要再新增兩個輸入表單, 所以還要在回想一下, 但大致上 ok, 所以完成了。

3. 請在各組的雲端 dokuwiki 中, 新增帳號與密碼都是由 abc001 ~ abc399 字串所組成的 399 名用戶登入對應資料, 並將製作過程與驗證流程拍成 flv 後上傳到個人的 Vimeo 資料區, 並將連結放在個人第九週頁面.

- 程式碼:

```
import hashlib
# convert user_password into sha1 encoded string
def gen_password(user_password):
    return hashlib.sha1(user_password.encode("utf-8")).hexdigest()
text='abc%03d'
for i in range(1,399+1):
    print(text%(i)+' ':''+gen_password(text%(i))+' ':''+text%(i)+'@gmail.com:'+''+text%(i)+'')
```

- 操作過程:

先寫一個程式使他能產生帳號與密碼都是由 abc001 ~ abc399 字串所組成的 399 名用戶登入對應資料，之後連線到 openshift dokuwiki 的 FileZilla，把產生出來的程式碼貼到 app-root/data/conf/users.auth.php 下，之後開啟 dokuwiki 網站即可。

- 心得:

一開始這個程式需要思考一下，如何使帳號與密碼都是由 abc001 ~ abc399 字串所組成的 399 名用戶，打出來後還要去讓所產生的程式碼符合一開始的格式，如同 users.auth.php，思考很久最後終於成功了。

w12

- 第十二週任務:

1. 請各組將第八週與第九週考試的摘要報告放入 Github 協同專案中的分組報告區，並將內容放入各組控管的同步 OpenShift 網站中。(佔期末成績 5 分)
2. 請各組設法利用 CherryPy 與 Pro/Web.Link 技術, 在 Github 協同專案中建立一個能夠透過連結或表單控制 Cube 零件, a, b, 或 c 零件尺寸的網際協同程式, 讓使用者可以藉以利用近端的 Creo 嵌入式瀏覽器控制 Cube 的尺寸後列出該零件的體積大小。(佔期末成績 5 分)

- 心得:

1. 在 URL 直接更改參數: 在對應的程式內，把迴圈改成可在 URL 後面輸入變數，EX:http://127.0.0.1:8080/cdag1/cube1?w=20&h=20&l=20
2. 在更改 FOR 迴圈時切記 {}! 在此 DEF 最下面還有一個括弧要刪除，否則會跑出 CATCH

- 組員自評:

40123101 : 85 分

40123102 : 95 分

40123132 : 95 分

網際鼓式煞車設計 (2ag2)

有關鼓式煞車

程式設計架構

鼓式煞車

結果與討論

這裡是結果與討論

W12 任務 (2ag3)

請各組將第九週考試的摘要報告放入 Github 協同專案中的分組報告區

摘要報告

第一題：

請寫一個執行時可以列出以十為底對數表的網際 Python 程式，然後 Push 到個人 bitbucket 空間，而且同步指到 OpenShift 個人帳號上執行。

```
import math
count = 0
text='%04d'
for i in range(100,200+10):
    print(text%round(math.log(i/100,10)*10000,0),end='-')
    count += 1
    if(count == 10):
        print()
        count = 0
```

解題心得：

可利用程式算出對數的值，就不需要一個一個按計算機。

第二題：

請在個人的 OpenShift 平台上建立一個能夠列印出與九九乘法表結果完全相同的網際程式，接著在乘法表上端加上兩個輸入表單，讓使用者輸入兩個整數，按下送出鍵後，程式會列出以此兩個整數為基底的乘法表，例如：若兩個欄位都輸入: 9，則列出九九乘法表，若輸入 9，20，則列出 9×20 的乘法表。

```
class Example(object):
    _cp_config = {
        # if there is no utf-8 encoding, no Chinese input available
        'tools.encode.encoding': 'utf-8',
        'tools.sessions.on' : True,
        'tools.sessions.storage_type' : 'file',
        'tools.sessions.locking' : 'explicit',
        'tools.sessions.storage_path' : data_dir+'/tmp',
        # session timeout is 60 minutes
        'tools.sessions.timeout' : 60
    }

    @cherry.py.expose
    def index(self):
        output = ''

        form = '''
        <form action='action'>
        num1:<INPUT type='text' name='num1'>
        num2:<INPUT type='text' name='num2'>
        <input type=submit>
        <input type=reset>
        </form>
        '''
        output += form
        return output

    @cherry.py.expose
    def action(self, num1=9, num2=9):
        num1 = int(num1)
        num2 = int(num2)
        output = ''
        for i in range(num1):
            for j in range(num2):
                output += str(i) + '*' + str(j) + '=' + str(i*j) + '<br />'
        return output
```

解題心得：

利用網站輸入值並算出，並需要有兩個輸入表單。

第三題：

請在各組的雲端 dokuwiki 中，新增帳號與密碼都是由 abc001 ~ abc399 字串所組成的 399 名用戶登入對應資料，並將製作過程與驗證流程拍成 flv 後上傳到個人的 Vimeo 資料區，並將連結放在個人第九週頁面。

```
import hashlib
#convert user_password into sha1 encoded string
def gen_password(user_password):
    return hashlib.sha1(user_password.encode("utf-8")).hexdigest()
text='abc%03d'
for i in range(1,399+1):
    print(text%(i)+':'+gen_password(text%(i))+':'+text%(i)+'@gmail.com:'+text%(i)+'@gmail.com')
```

解題心得：

可以利用程式創造多人帳密，一起管理網站，減少一個一個創建帳密，又會有被盜用的風險。

```
import math count = 0 text='%04d' for i in range(100,200+10): print(text%round(math.log(i/100,10)*1000)) count += 1 if(count == 10): print() count = 0
```

```
class Example(object): cpconfig = { # if there is no utf-8 encoding, no Chinese input available 'tools.encode.encoding': 'utf-8', 'tools.sessions.on' : True, 'tools.sessions.storage_type' : 'file', 'tools.sessions.locking' : 'explicit', 'tools.sessions.storage_path' : data_dir+'/tmp', # session timeout is 60 minutes 'tools.sessions.timeout' : 60 }
```

```
@cherry.py.expose def index(self): output = "
```

```
form = '''
<form action='action'>
num1:<INPUT type='text' name='num1'>
num2:<INPUT type='text' name='num2'>
<input type=submit>
<input type=reset>
</form>
'''
output += form
return output
```

```
@cherry.py.expose def action(self, num1=9, num2=9): num1 = int(num1) num2 = int(num2) output = '' for i in range(num1): for j in range(num2): output += str(i) + '+' + str(j) + '=' + str(i*j) + ' ' return output
```

```
import hashlib #convert user_password into sha1 encoded string def gen_password(user_password): return hashlib.sha1(user_password.encode("utf-8")).hexdigest() text='abc%03d' for i in range(1,399+1): print(text%(i)+':'+gen_password(text%(i))+':'+text%(i)+'@gmail.com:'+text%(i)+'@gmail.com')
```

網際鼓式煞車設計 (2ag4)

github 連結 40123107: <https://github.com/40123107> 40123120: <https://github.com/40123120>
cdag4: <https://github.com/2014cdag4/2014cdag4>

cmsimply 連結 40123107: <https://github.com/40123107> 40123120: <https://github.com/40123120>
40123150: <http://cdg4-40123150.rhcloud.com/> ===

第九週 1.(第一題 30%) 請寫一個執行時可以列出以十為底對數表的網際 Python 程式, 然後 Push 到個人 bitbucket 空間, 而且同步指到 OpenShift 個人帳號上執行. http://cdg4-40123150.rhcloud.com/get_page?heading=%E7%AC%AC%E4%B8%80%E9%A1
2.(第二題 40%) 請在個人的 OpenShift 平台上建立一個能夠列印出與九九乘法表結果完全相同的網際程式, 接著在乘法表上端加上兩個輸入表單, 讓使用者輸入兩個整數, 按下送出鍵後, 程式會列出以此兩個整數為基底的乘法表, 例如: 若兩個欄位都輸入: 9, 則列出九九乘法表, 若輸入 9, 20, 則列出 9×20 的乘法表. <http://cdg4-40123150.rhcloud.com/example/> <http://vimeo.com/92577008>
3.(第三題 30%) 請在各組的雲端 dokuwiki 中, 新增帳號與密碼都是由 abc001 ~ abc399 字串所組成的 399 名用戶登入對應資料, 並將製作過程與驗證流程拍成 flv 後上傳到個人的 Vimeo 資料區, 並將連結放在個人第九週頁面. <https://php-40123150.rhcloud.com/doku.php?id=start> <http://vimeo.com/92577072> ===
w12 請連結 <https://github.com/2014cdag4/2014cdag4> ===

這裡是結果與討論

網際四連桿機構設計 (2ag5)

有關連桿設計

程式設計架構

連桿計算

結果與討論

這裡是結果與討論

by 2014cdag5

網際鼓式煞車設計 (2ag6)

有關鼓式煞車

程式設計架構

鼓式煞車

結果與討論

這裡是結果與討論

網際鼓式煞車設計 (2ag7)

有關鼓式煞車

程式設計架構

鼓式煞車

結果與討論

這裡是結果與討論

網際四連桿機構設計 (2ag8)

有關連桿設計

程式設計架構

連桿計算

結果與討論

這裡是結果與討論

by 2014cdag5

網際鼓式煞車設計 (2ag9)

有關鼓式煞車

程式設計架構

鼓式煞車

結果與討論

這裡是結果與討論

這是第九組

分組工作日誌

分組評分

```
#coding: utf-8
'''
```

```
http://www.beyondmech.com/pro-e/cad-topic-27.html
```

```
" 本程式的目的在輔助設計者選擇齒輪的尺寸大小，";
```

```
" 由於相嚙合的兩齒輪其徑節 (Diametral Pitch) 相同";
```

```
"，齒的大小也相同。因徑節為每單位直徑的齒數，因此徑節愈大，則其齒的尺寸愈小";
```

```
"；反之，徑節愈小，則齒的尺寸則愈大。";
```

```
" 一般在設計齒輪對時，為避免使用過大的齒及過寬的齒面厚度，因此必須要就齒輪大小與強度與負載加
```

```
" 一般而言是希望齒輪面的寬度 (Face Width) 能大於 3 倍周節 (Circular Pitch)，以避免選用太大的
```

```
" 並且希望齒輪面的寬度 (Face Width) 能小於 5 倍周節，以便齒面傳遞負載時能有較為均勻的分佈，因
```

```
" 設  $d$  為齒輪的節圓直徑 (Pitch Diameter)，單位為英吋";
```

```
" $N$  為齒數";
```

```
" $P$  為徑節，即單位英吋的齒數";
```

```
" 因此  $d=N/P$ ";
```

```
" 設  $V$  為節線速度 (Pitch Line Velocity)，單位為英呎/分鐘";
```

```
" 因此  $V=(PI) * d * n/12$ ";
```

```
" 其中  $n$  為齒輪轉速，單位為 rpm";
```

```
" 設傳輸負載大小為  $W$ ，單位為 pounds";
```

```
" 因此  $W=33000H/V$ ";
```

```
" 其中  $H$  為傳輸功率，單位為 horse power";
```

```
" 若設  $K$  為速度因子 (Velocity Factor)";
```

```
" 因此  $K=1200/(1200+V)$ ";
```

```

" 最後可求出齒輪的齒面寬度 (Face Width)  $F$  , 單位為英吋";
" 即  $F=WP/KYS$ ";
" 其中  $S$  為齒面的材料彎曲應力強度";
" 設計要求: 控制所選齒的尺寸大小, 在滿足強度與傳輸負載的要求下, 讓齒面厚度介於 3 倍周節與 5
" 設計者可以選擇的參數:";
" 安全係數 (建議值為 3 以上)";
" 齒輪減速比";
" 馬達傳輸功率, 單位為 horse power";
" 馬達轉速, 單位為 rpm";
" 齒制 (Gear System)";
" 齒輪材料與強度";
'''

# 這個程式要計算正齒輪的齒面寬, 資料庫連結希望使用 pybean 與 SQLite
# 導入 pybean 模組與所要使用的 Store 及 SQLiteWriter 方法
from pybean import Store, SQLiteWriter
import math

def interpolation(小齒輪齒數, 齒形):
    global SQLite 連結
    # 使用內插法求值
    # 找出比目標齒數大的其中的最小的, 就是最鄰近的大值
    lewis_factor = SQLite 連結.find_one("lewis", "gearno > ?", [小齒輪齒數])
    if(齒形 == 1):
        larger_formfactor = lewis_factor.type1
    elif(齒形 == 2):
        larger_formfactor = lewis_factor.type2
    elif(齒形 == 3):
        larger_formfactor = lewis_factor.type3
    else:
        larger_formfactor = lewis_factor.type4
    larger_toothnumber = lewis_factor.gearno

    # 找出比目標齒數小的其中的最大的, 就是最鄰近的小值
    lewis_factor = SQLite 連結.find_one("lewis", "gearno < ? order by gearno DESC", [小齒輪齒數])
    if(齒形 == 1):
        smaller_formfactor = lewis_factor.type1
    elif(齒形 == 2):
        smaller_formfactor = lewis_factor.type2
    elif(齒形 == 3):
        smaller_formfactor = lewis_factor.type3
    else:
        smaller_formfactor = lewis_factor.type4
    smaller_toothnumber = lewis_factor.gearno
    calculated_factor = larger_formfactor + (小齒輪齒數 - larger_toothnumber) * (larger_f
    # 只傳回小數點後五位數
    return round(calculated_factor, 5)

```

```

# 取得設計參數
馬力 = 100
轉速 = 1120
減速比 = 4
齒形 = 4
安全係數 = 3
#unsno_treatment
材料 = "G10350_CD"
小齒輪齒數 = 18

# 改寫為齒面寬的設計函式
def gear_width(馬力, 轉速, 減速比, 齒形, 安全係數, 材料, 小齒輪齒數):
    # 根據所選用的齒形決定壓力角
    if(齒形 == 1 or 齒形 == 2):
        壓力角 = 20
    else:
        壓力角 = 25

    # 根據壓力角決定最小齒數
    if(壓力角 == 20):
        最小齒數 = 18
    else:
        最小齒數 = 12

    # 直接設最小齒數
    if 小齒輪齒數 <= 最小齒數:
        小齒輪齒數 = 最小齒數
    # 大於 400 的齒數則視為齒條 (Rack)
    if 小齒輪齒數 >= 400:
        小齒輪齒數 = 400

    # 根據所選用的材料查詢強度值
    # 由 material 之序號查 steel 表以得材料之降伏強度 S 單位為 kpsi 因此查得的值要成乘上 10
    # 利用 Store 建立資料庫檔案對應物件, 並且設定 frozen=True 表示不要開放動態資料表的建立
    SQLite 連結 = Store(SQLiteWriter("lewis.db", frozen=True))
    # 指定 steel 資料表
    steel = SQLite 連結.new("steel")
    # 資料查詢
    # 將 unsno 與 treatment 從材料字串中隔開
    unsno, treatment = 材料.split("_")
    #print(unsno, treatment)

    material = SQLite 連結.find_one("steel","unsno=? and treatment=?", [unsno, treatment])
    # 列出 steel 資料表中的資料筆數
    #print(SQLite 連結.count("steel"))

```



```

print (material.yield_str)
strengthstress = material.yield_str*1000
# 由小齒輪的齒數與齒形類別，查詢 lewis form factor
# 先查驗是否有直接對應值
on_table = SQLite 連結.count("lewis","gearno=?", [小齒輪齒數])
if on_table == 1:
    # 直接進入設計運算
    print(" 直接運算")
    print(on_table)
    lewis_factor = SQLite 連結.find_one("lewis","gearno=?", [小齒輪齒數])
    #print(lewis_factor.type1)
    # 根據齒形查出 formfactor 值
    if(齒形 == 1):
        formfactor = lewis_factor.type1
    elif(齒形 == 2):
        formfactor = lewis_factor.type2
    elif(齒形 == 3):
        formfactor = lewis_factor.type3
    else:
        formfactor = lewis_factor.type4
else:
    # 沒有直接對應值，必須進行查表內插運算後，再執行設計運算
    print(" 必須內插")
    #print(interpolation(小齒輪齒數，齒形))
    formfactor = interpolation(小齒輪齒數，齒形)

# 開始進行設計運算

ngear = 小齒輪齒數 * 減速比

# 重要的最佳化設計---儘量用整數的 diametralpitch
# 先嘗試用整數算若 diametralpitch 找到 100 仍無所獲則改用 0.25 作為增量再不行則宣告 fa
counter = 0
i = 0.1
facewidth = 0
circularpitch = 0
while (facewidth <= 3 * circularpitch or facewidth >= 5 * circularpitch):
    diametralpitch = i
    #circularpitch = 3.14159/diametralpitch
    circularpitch = math.pi/diametralpitch
    pitchdiameter = 小齒輪齒數/diametralpitch
    #pitchlinevelocity = 3.14159*pitchdiameter* 轉速/12
    pitchlinevelocity = math.pi * pitchdiameter * 轉速/12
    transmittedload = 33000 * 馬力/pitchlinevelocity
    velocityfactor = 1200/(1200 + pitchlinevelocity)
    # formfactor is Lewis form factor

```

```

# formfactor need to get from table 13-3 and determined ty teeth number and type
# formfactor = 0.293
# 90 is the value get from table corresponding to material type
facewidth = transmittedload * diametralpitch * 安全係數/velocityfactor/formfactor
if(counter>5000):
    print(" 超過 5000 次的設計運算，仍無法找到答案!")
    print(" 可能所選用的傳遞功率過大，或無足夠強度的材料可以使用!")
    # 離開 while 迴圈
    break
i += 0.1
counter += 1
facewidth = round(facewidth, 4)
if(counter<5000):
    print(" 進行"+str(counter)+" 次重複運算後，得到合用的 facewidth 值為:"+str(facewidth))

```

gear_width(馬力，轉速，減速比，齒形，安全係數，材料，小齒輪齒數)

```

#coding: utf-8
'''
" 本程式的目的在輔助設計者選擇齒輪的尺寸大小，";
" 由於相嚙合的兩齒輪其徑節 (Diametral Pitch) 相同";
"，齒的大小也相同。因徑節為每單位直徑的齒數，因此徑節愈大，則其齒的尺寸愈小";
"; 反之，徑節愈小，則齒的尺寸則愈大。";
" 一般在設計齒輪對時，為避免使用過大的齒及過寬的齒面厚度，因此必須要就齒輪大小與強度與負載加以限制";
" 一般而言是希望齒輪面的寬度 (Face Width) 能大於 3 倍周節 (Circular Pitch)，以避免選用太大的齒輪";
" 並且希望齒輪面的寬度 (Face Width) 能小於 5 倍周節，以便齒面傳遞負載時能有較為均勻的分佈，且能減少齒面磨損";
" 設 d 為齒輪的節圓直徑 (Pitch Diameter)，單位為英吋";
"N 為齒數";
"P 為徑節，即單位英吋的齒數";
" 因此 d=N/P";
" 設 V 為節線速度 (Pitch Line Velocity)，單位為英呎/分鐘";
" 因此 V=(PI) * d * n/12";
" 其中 n 為齒輪轉速，單位為 rpm";
" 設傳輸負載大小為 W，單位為 pounds";
" 因此 W=33000H/V";
" 其中 H 為傳輸功率，單位為 horse power";
" 若設 K 為速度因子 (Velocity Factor)";
" 因此 K=1200/(1200+V)";
" 最後可求出齒輪的齒面寬度 (Face Width) F，單位為英吋";
" 即 F=W/KYS";
" 其中 S 為齒面的材料彎曲應力強度";
" 設計要求：控制所選齒的尺寸大小，在滿足強度與傳輸負載的要求下，讓齒面厚度介於 3 倍周節與 5 倍周節";
" 設計者可以選擇的參數:";
" 安全係數 (建議值為 3 以上)";
" 齒輪減速比";
" 馬達傳輸功率，單位為 horse power";
'''

```

```

" 馬達轉速，單位為 rpm";
" 齒制 (Gear System)";
" 齒輪材料與強度";
'''
# 這個程式要計算正齒輪的齒面寬，資料庫連結希望使用 pybean 與 SQLite
# 導入 pybean 模組與所要使用的 Store 及 SQLiteWriter 方法
from pybean import Store, SQLiteWriter
import math

SQLite 連結 = Store(SQLiteWriter("lewis.db", frozen=True))
overall_counter = 0
查表次數 = 0
材料計數 = 0

# 執行 formfactor 內插運算的函式
def interpolation(小齒輪齒數, 齒形):
    global SQLite 連結
    # 使用內插法求值
    # 找出比目標齒數大的其中的最小的，就是最鄰近的大值
    lewis_factor = SQLite 連結.find_one("lewis", "gearno > ?", [小齒輪齒數])
    if(齒形 == 1):
        larger_formfactor = lewis_factor.type1
    elif(齒形 == 2):
        larger_formfactor = lewis_factor.type2
    elif(齒形 == 3):
        larger_formfactor = lewis_factor.type3
    else:
        larger_formfactor = lewis_factor.type4
    larger_toothnumber = lewis_factor.gearno

    # 找出比目標齒數小的其中的最大的，就是最鄰近的小值
    lewis_factor = SQLite 連結.find_one("lewis", "gearno < ? order by gearno DESC", [小齒輪齒數])
    if(齒形 == 1):
        smaller_formfactor = lewis_factor.type1
    elif(齒形 == 2):
        smaller_formfactor = lewis_factor.type2
    elif(齒形 == 3):
        smaller_formfactor = lewis_factor.type3
    else:
        smaller_formfactor = lewis_factor.type4
    smaller_toothnumber = lewis_factor.gearno
    calculated_factor = larger_formfactor + (小齒輪齒數 - larger_toothnumber) * (larger_f
    # 只傳回小數點後五位數
    print(" 小齒輪齒數:" + str(小齒輪齒數) + ", 齒形" + str(齒形) + ", formfactor:" + str(round(calculated_factor, 5)))
    return round(calculated_factor, 5)

```

```

# 取得設計參數
'''
馬力 = 100
轉速 = 1120
減速比 = 4
齒形 = 4
安全係數 = 3
#unsno_treatment
材料 = "G10350_CD"
小齒輪齒數 = 18
'''

# 改寫為齒面寬的設計函式
def gear_width(馬力, 轉速, 減速比, 齒形, 安全係數, 材料, 小齒輪齒數):
    global SQLite 連結
    global overall_counter
    global 查表次數
    global 材料計數
    # 根據所選用的齒形決定壓力角
    if(齒形 == 1 or 齒形 == 2):
        壓力角 = 20
    else:
        壓力角 = 25

    # 根據壓力角決定最小齒數
    if(壓力角 == 20):
        最小齒數 = 18
    else:
        最小齒數 = 12

    # 直接設最小齒數
    if 小齒輪齒數 <= 最小齒數:
        小齒輪齒數 = 最小齒數
    # 大於 400 的齒數則視為齒條 (Rack)
    if 小齒輪齒數 >= 400:
        小齒輪齒數 = 400

    # 根據所選用的材料查詢強度值
    # 由 material 之序號查 steel 表以得材料之降伏強度 S 單位為 kpsi 因此查得的值要成乘上 10
    # 利用 Store 建立資料庫檔案對應物件, 並且設定 frozen=True 表示不要開放動態資料表的建立
    #SQLite 連結 = Store(SQLiteWriter("lewis.db", frozen=True))
    # 指定 steel 資料表
    steel = SQLite 連結.new("steel")
    # 資料查詢
    # 將 unsno 與 treatment 從材料字串中隔開
    unsno, treatment = 材料.split("_", 1)

```

```

#print(unsno, treatment)
treatment = treatment.replace("_", " ")
#print(treatment)
material = SQLite 連結.find_one("steel","unsno=? and treatment=?", [unsno, treatment])
# 列出 steel 資料表中的資料筆數
#print(SQLite 連結.count("steel"))
#print (material.yield_str)
if material.yield_str > 100:
    材料計數 += 1
strengthstress = material.yield_str*1000
# 由小齒輪的齒數與齒形類別，查詢 lewis form factor
# 先查驗是否有直接對應值
on_table = SQLite 連結.count("lewis","gearno=?", [小齒輪齒數])
if on_table == 1:
    # 直接進入設計運算
    #print(" 直接運算")
    #print(on_table)
    lewis_factor = SQLite 連結.find_one("lewis","gearno=?", [小齒輪齒數])
    #print(lewis_factor.type1)
    # 根據齒形查出 formfactor 值
    if(齒形 == 1):
        formfactor = lewis_factor.type1
    elif(齒形 == 2):
        formfactor = lewis_factor.type2
    elif(齒形 == 3):
        formfactor = lewis_factor.type3
    else:
        formfactor = lewis_factor.type4
else:
    # 沒有直接對應值，必須進行查表內插運算後，再執行設計運算
    #print(" 必須內插")
    #print(interpolation(小齒輪齒數, 齒形))
    查表次數 += 1
    print(" 第"+str(查表次數)+" 次查表")
    formfactor = interpolation(小齒輪齒數, 齒形)

# 開始進行設計運算

ngear = 小齒輪齒數 * 減速比

# 重要的最佳化設計---儘量用整數的 diametralpitch
# 先嘗試用整數算若 diametralpitch 找到 100 仍無所獲則改用 0.25 作為增量再不行則宣告 fa
counter = 0
i = 0.1
facewidth = 0
circularpitch = 0

```

```

while (facewidth <= 3 * circularpitch or facewidth >= 5 * circularpitch):
    diametralpitch = i
    #circularpitch = 3.14159/diametralpitch
    circularpitch = math.pi/diametralpitch
    pitchdiameter = 小齒輪齒數/diametralpitch
    #pitchlinevelocity = 3.14159*pitchdiameter* 轉速/12
    pitchlinevelocity = math.pi * pitchdiameter * 轉速/12
    transmittedload = 33000 * 馬力/pitchlinevelocity
    velocityfactor = 1200/(1200 + pitchlinevelocity)
    # formfactor is Lewis form factor
    # formfactor need to get from table 13-3 and determined ty teeth number and type
    # formfactor = 0.293
    # 90 is the value get from table corresponding to material type
    facewidth = transmittedload * diametralpitch * 安全係數/velocityfactor/formfactor
    if(counter>5000):
        print(" 超過 5000 次的設計運算，仍無法找到答案!")
        print(" 可能所選用的傳遞功率過大，或無足夠強度的材料可以使用!")
        # 離開 while 迴圈
        break
    i += 0.1
    counter += 1
facewidth = round(facewidth, 4)
if(counter<5000):
    print(" 進行"+str(counter)+" 次重複運算後，得到合用的 facewidth 值為:"+str(facewidth))
overall_counter += counter

# 執行正齒輪齒面寬的設計運算
#gear_width(馬力，轉速，減速比，齒形，安全係數，材料，小齒輪齒數)

# 執行輸入檔案的解讀
輸入檔案 = open('design_input.txt', encoding="UTF-8") # 開檔案的內建模式為 read
# 先將數字檔案中各行資料打包成為 list
# 先讀第一行的標題資料行
各行資料 = 輸入檔案.readline()
design_count = 0
輸入 = []
while True:
    # readline() 讀取單行
    # readlines() 讀取多行，並放入串列資料格式中
    各行資料 = 輸入檔案.readline()
    design_count += 1
    print("design#:"+str(design_count))
    #print(各行資料,end="")
    # 以下兩行判斷式在確定檔案讀到最後一行後就會跳出 while 迴圈，不會無限執行
    if len(各行資料) == 0: # 若該行的字數為 0，表示已經到底
        break

```

```

# 去掉各行最後面的跳行符號
各行資料 = 各行資料.rstrip()
#print(各行資料,end="")
# 依照資料的區隔符號 "\t" 將各行資料拆開，並且存為 list，到這裡各行資料為 list
各行資料 = 各行資料.split("\t")
'''

# 取得設計參數
馬力 = 100
轉速 = 1120
減速比 = 4
齒形 = 4
安全係數 = 3
#unsno_treatment
材料 = "G10350_CD"
小齒輪齒數 = 18
'''

馬力 = int(各行資料 [0])
轉速 = int(各行資料 [1])
減速比 = float(各行資料 [2])
齒形 = int(各行資料 [3])
安全係數 = float(各行資料 [4])
材料 = 各行資料 [5]
小齒輪齒數 = int(各行資料 [6])
gear_width(馬力, 轉速, 減速比, 齒形, 安全係數, 材料, 小齒輪齒數)
print(" 總運算次數:"+str(overall_counter))
# 可以將各行資料印出檢查
#print(各行資料)
# 將各行資料數列再疊成 數字 list
# 輸入.append(各行資料)
#print(輸入)
# 取得各行輸入值後，再呼叫 gear_width 執行齒面寬的設計運算
輸入檔案.close()
print(" 查表次數:"+str(查表次數))
print(" 材料計數:"+str(材料計數))

```

Welcome to the 2014cdb wiki!

注意事項

- Python 版次必須使用 3.3 版
- Leo Editor 必須使用 4.11 版
- Creo 必須使用 64 位元教育版本
- IE 建議使用 11 版

- 建立 pdf 與 html 文件則需要 pandoc 與 portableLatex 可攜應用程式 (位於 apps 目錄下)
- Git 則使用可攜版本 (位於 apps 目錄下)

各組 push 資料確認事項

- git config --list 中的 user.name 是否為各組代號 (代號錯誤將無法取分)
- application 網站是否可以正確執行
- .leo 檔案是否可以被 Leo Editor 正確開啟
- 專案是否可以正確產生協同報告 pdf 與 html 檔案
- 與 .leo 專案對應的外部目錄與檔案是否正確設定

同步雲端網站

<http://2014cda-mdenfu.rhcloud.com/>

2014scda 分組日誌

[2014scda 分組日誌]

2014cdag1 分組網站: <http://2014cdag1-cadp13ag6.rhcloud.com/>

2014cdag1 分組 Github: <https://github.com/2014cdag1>

2014cdag2
分組網
站:https://2014cdag2-40123103.rhcloud.com/get_page?heading=
2014cdag2
分組
Github:<https://github.com/2014cdag2>

2014cdag3 分組網站 : <http://2014cdag3-cadp13ag10.rhcloud.com/>

2014cdag3 分組 Github : <https://github.com/2014cdag3/cdg3>

2014cdag4 分組網站:<http://cdg4-40123150.rhcloud.com/>

2014cdag4 分組 Github:<https://github.com/2014cdag4>

2014cdag5
分組網
站：
[http://2014cdag5-
yimin40123157.rhcloud.com/](http://2014cdag5-yimin40123157.rhcloud.com/)
2014cdag5
分組
Github：
<https://github.com/2014cdag5>

2014cdag6 分組網站：<http://2014cdag6-40123109cd2014.rhcloud.com/>
2014cdag6 分組 Github：<https://github.com/2014cdag6/2014cdag6>

2014cdag7
分組網
站:[http://python-
49823207.rhcloud.com/](http://python-49823207.rhcloud.com/)
2014cdag7
分組
Github:<https://github.com/2014cdag7/2014cdag7>

2014cdag8 分組網站:<http://2014cdag8-40123141.rhcloud.com/>
2014cdag8 分組 Github:<https://github.com/2014cdag8>

2014cdag9 分組網站：<http://2014cadg9-40123114.rhcloud.com/>
2014cdag9 分組 Github：<https://github.com/2014cdag9>

2014cdag10
分組網
站：
[http://cdag10-
40123153.rhcloud.com/](http://cdag10-40123153.rhcloud.com/)
2014cdag10
分組
Github：
<https://github.com/2014cdag10/2014cdag10>

2014cdag12 分組網站：<http://cadpa1-40123123.rhcloud.com/>

2014cdag12 分組 Github：<https://github.com/2014cdag12>

2014cdag13

分組網

站：

<http://python-40123124.rhcloud.com/>

2014cdag13

分組

Github：

<https://github.com/2014cdag13>

2014cdag14 分組網站：<https://github-cadpag5.rhcloud.com/>

2014cdag14 分組 Github：<https://github.com/2014cdag14>

2014cdag15

分組網

站：

<http://2014cdag15-40123146.rhcloud.com/>

2014cdag15

分組

Github：

<https://github.com/2014cdag15>

2014cdag16 分組網站：<http://github-cda2014g16.rhcloud.com/>

2014cdag16 分組 Github：<https://github.com/2014cdag16/2014cdag16>

2014cdag17

分組網

站：

<http://2014cd17-ccwei.rhcloud.com/>

2014cdag17

分組

Github：

<https://github.com/2014cdag17/CDAG17>

2014cdag18 分組網站：<http://2014cdag18-40123126.rhcloud.com/>

2014cdag18 分組 Github：<https://github.com/2014cdag18>

2014cdag21

分組網

站：

<https://cmsimply-40123156.rhcloud.com/>

2014cdag21

分組

Github：

<https://github.com/2014cdag21/c21>

各週課後註記

自 W11 使用 `pandoc_auto` 產生協同報告文件, 不再使用手動增加分組報告, 只要在 `doc/分組代號/` 中加入 `doc.txt`, 並且使用正確的 `pandoc markdown` 格式就可以從 `pandoc_auto` 按鈕產生對應的 `html` 與 `pdf` 檔案.

自 W12 起各組必須在分組 `doc/分組代號/doc.txt` 中加入第八週與第九週考試內容與心得.

Creo 討論區

<http://www.mcadcentral.com/creo-user-forum-topics/>

```
var drwmodels = drawing.ListModels(); newWin.document.writeln (“<br>” +  
“Drawing:” + drawing.FileName + ” ( ” + drwmodels.Count + ” drw models  
)”);
```

```
var modelitemtype = pfcCreate(“pfcModelItemType”);
```

```
/-Retrieve the symbol definition from the system-/
```

```
var symDef = drawing.RetrieveSymbolDefinition (“sym_name”, void null, void  
null, true);
```

```
for (var i = 0; i < drwmodels.Count; i++) {
```

```
drwmodel = drwmodels.Item(i); var showndims = drawing.ListShownDimensions(drwmodel,  
modelitemtype.ITEM_DIMENSION); newWin.document.writeln (“<br>” +  
” Model ” + (i+1) + “:” + drwmodel.FileName + ” ( ” + showndims.Count  
+ ” shown dims )”);
```

```
for (var j = 0; j < showndims.Count; j++) {
```

```

var pnt3d = void null; var showndim = showndims.Item(j); var symbol = showndim.Symbol; var value = showndim.DimValue; pnt3d = showndim.Location;

newWin.document.writeln ("<br>" + "    Dim " + (j+1) + ":" + symbol + " = " + value + " ( " + pnt3d.Item(0) + " , " + pnt3d.Item(1) + " , " + pnt3d.Item(2) + " )");

var position = void null; var instrs = pfcCreate ("pfcDetailSymbolInstInstructions").Create (symDef); position = pfcCreate ("pfcFreeAttachment").Create (pnt3d); var allAttachments = pfcCreate ("pfcDetailLeaders").Create ();

position.AttachmentPoint = pnt3d; allAttachments.ItemAttachment = position; instrs.InstAttachment = allAttachments;

/-Set the VariantTexts (for var text)-/

var vtp = pfcCreate ("pfcDetailVariantText").Create ("vartext", j+1);
var vtp2 = pfcCreate ("pfcDetailVariantTexts"); vtp2.append(vtp);
instrs.TextValues = vtp2;

/-Create and display the symbol-/

var symInst = drawing.CreateDetailItem (instrs); symInst.Show(); } }

pfcWindow.Window.ExportRasterImage to save graphics

ProWindowPanZoomMatrixSet

```

ComponentConstraintType—Using the TYPE options, you can specify the placement constraint types. They are as follows:

ASM_CONSTRAINT_MATE—Use this option to make two surfaces touch one another, that is coincident and facing each other.

ASM_CONSTRAINT_MATE_OFF—Use this option to make two planar surfaces parallel and facing each other.

ASM_CONSTRAINT_ALIGN—Use this option to make two planes coplanar, two axes coaxial and two points coincident. You can also align revolved surfaces or edges. ○ ASM_CONSTRAINT_ALIGN_OFF—Use this option to align two planar surfaces at an offset.

ASM_CONSTRAINT_INSERT—Use this option to insert a “male” revolved surface into a “female” revolved surface, making their respective axes coaxial.

ASM_CONSTRAINT_ORIENT—Use this option to make two planar surfaces to be parallel in the same direction.

ASM_CONSTRAINT_CSYS—Use this option to place a component in an assembly by aligning the coordinate system of the component with the coordinate system of the assembly.

ASM_CONSTRAINT_TANGENT—Use this option to control the contact of two surfaces at their tangents.

ASM_CONSTRAINT_PNT_ON_SRF—Use this option to control the contact of a surface with a point.

ASM_CONSTRAINT_EDGE_ON_SRF—Use this option to control the contact of a surface with a straight edge.

ASM_CONSTRAINT_DEF_PLACEMENT—Use this option to align the default coordinate system of the component to the default coordinate system of the assembly.

ASM_CONSTRAINT_SUBSTITUTE—Use this option in simplified representations when a component has been substituted with some other model

ASM_CONSTRAINT_PNT_ON_LINE—Use this option to control the contact of a line with a point.

ASM_CONSTRAINT_FIX—Use this option to force the component to remain in its current packaged position.

ASM_CONSTRAINT_AUTO—Use this option in the user interface to allow an automatic choice of constraint type based upon the references.

```
getBodyDiagonalInfo: function() { try { var cModel = this.getCurrentModel();  
var mitems = cModel.ListItems(pfcCreate("pfcModelItemType").ITEM_COORD_SYS);  
var cor = mitems.Item(0); var trans3 = cor.CoordSys;
```

```
var modtyp = pfcCreate("pfcModelItemTypes");  
var item1 = pfcCreate("pfcModelItemType").ITEM_AXIS;  
var item2 = pfcCreate("pfcModelItemType").ITEM_POINT;  
var item3 = pfcCreate("pfcModelItemType").ITEM_COORD_SYS;
```

```
modtyp.Insert(0,item1);  
modtyp.Insert(1,item2);  
modtyp.Insert(2,item3);  
var outline = cModel.EvalOutline(trans3,modtyp);  
var point2 = outline.Item(0);  
var x = point2.Item(0);  
var y = point2.Item(1);  
var z = point2.Item(2);  
var point3 = outline.Item(1);  
var p = point3.Item(0);  
var q = point3.Item(1);  
var r = point3.Item(2);  
var sum = Math.pow(p - x, 2) + Math.pow(q - y, 2) + Math.pow(r - z, 2);  
var diagonal = Math.sqrt(sum);  
return diagonal;
```

```
} catch (e) { throw e; } }
```

建立元件的約束物件

`pfcComponentConstraint.Create()`

`ComponentConstraintType` Using the TYPE options, you can specify the placement constraint types. They are as follows:

兩平面進行面接約束 (`ASM_CONSTRAINT_MATE`)

○ `ASM_CONSTRAINT_MATE` Use this option to make two surfaces touch one another, that is coincident and facing each other.

兩平面進行面接約束, 且相隔一段距離 (`ASM_CONSTRAINT_MATE_OFF`)

○ `ASM_CONSTRAINT_MATE_OFF` Use this option to make two planar surfaces parallel and facing each other.

兩平面或兩軸對齊 (`ASM_CONSTRAINT_ALIGN`)

○ `ASM_CONSTRAINT_ALIGN` Use this option to make two planes coplanar, two axes coaxial and two points coincident. You can also align revolved surfaces or edges.

兩平面或兩軸對齊, 且維持一特定距離 (`ASM_CONSTRAINT_ALIGN_OFF`)

○ `ASM_CONSTRAINT_ALIGN_OFF` Use this option to align two planar surfaces at an offset.

插入約束 (`ASM_CONSTRAINT_INSERT`)

○ `ASM_CONSTRAINT_INSERT` Use this option to insert a “male” revolved surface into a “female” revolved surface, making their respective axes coaxial.

同方位約束 (`ASM_CONSTRAINT_ORIENT`)

○ `ASM_CONSTRAINT_ORIENT` Use this option to make two planar surfaces to be parallel in the same direction.

座標系統定位約束 (`ASM_CONSTRAINT_CSYS`)

○ `ASM_CONSTRAINT_CSYS` Use this option to place a component in an assembly by aligning the coordinate system of the component with the coordinate system of the assembly.

垂直約束 (`ASM_CONSTRAINT_TANGENT`)

○ `ASM_CONSTRAINT_TANGENT` Use this option to control the contact of two surfaces at their tangents.

控制與某曲面上一點接觸約束 (`ASM_CONSTRAINT_PNT_ON_SRF`)

○ `ASM_CONSTRAINT_PNT_ON_SRF` Use this option to control the contact of a surface with a point.

控制與某曲面上一線接觸約束 (`ASM_CONSTRAINT_EDGE_ON_SRF`)

○ `ASM_CONSTRAINT_EDGE_ON_SRF` Use this option to control the contact of a surface with a straight edge.

與組立中的特定座標系統對齊 (ASM_CONSTRAINT_DEF_PLACEMENT)

○ ASM_CONSTRAINT_DEF_PLACEMENT Use this option to align the default coordinate system of the component to the default coordinate system of the assembly.

取代約束 (ASM_CONSTRAINT_SUBSTITUTE)

○ ASM_CONSTRAINT_SUBSTITUTE Use this option in simplified representations when a component has been substituted with some other model

線與點進行接觸約束 (ASM_CONSTRAINT_PNT_ON_LINE)

○ ASM_CONSTRAINT_PNT_ON_LINE Use this option to control the contact of a line with a point.

固定約束 (ASM_CONSTRAINT_FIX)

○ ASM_CONSTRAINT_FIX Use this option to force the component to remain in its current packaged position.

○ ASM_CONSTRAINT_AUTO Use this option in the user interface to allow an automatic choice of constraint type based upon the references. AssemblyDatumSide Orientation of the assembly. This can have the following values:

進行正面方向約束

○ Yellow The primary side of the datum plane which is the default direction of the arrow.

進行反面方向約束

○ Red The secondary side of the datum plane which is the direction opposite to that of the arrow.

ComponentReference A reference on the placed component.

ComponentDatumSide Orientation of the assembly component. This can have the following values:

正面約束

○ Yellow The primary side of the datum plane which is the default direction of the arrow.

反面約束 ○ Red The secondary side of the datum plane which is the direction opposite to that of the arrow.

Offset The mate or align offset value from the reference.

In Pro/ENGINEER, a component's constraints are defined to restrict its movement with respect to other components. Under certain conditions, a component's definition dialog box shows an option to "Allow assumptions" on the component.

When you select this option, Pro/ENGINEER adds more constraints to completely constrain the component's motion with respect to its connected, neighboring components.

Because these extra constraints are hidden, the exporter cannot accurately determine the DoF restrictions between a component and its connected, neighboring components.

To Allow Constraint Orientation Assumptions

The Allow Assumptions check box, located in the Placement Status area of the Component Placement dialog box, allows you to switch system constraint orientation assumptions on and off. The Allow Assumptions check box is available whenever assumptions are made or could be made; when a component is fully constrained, the check box disappears. The setting of Allow Assumptions is component specific, and the setting is saved with the component.

Allow assumptions - will allow Pro/E to make assumptions as to what you think something to be fully constrained, like a bushing in a cylinder after you use the insert and align will be listed as fully constrained even though a rotational degree of freedom still exist.

Typically, three placement constraints are required to fully constrain a component. Fewer constraints may be applied if the system is able to position the component by making assumptions regarding orientation.

In reality, there is still a rotational degree of freedom open for this component, but that may not be critical in our design, so we can allow the system to fix this rotational degree of freedom for us. If we were to uncheck the "Allow Assumptions" box, we would have to define additional placement constraints to fix this rotational degree of freedom.

```
#coding: utf-8
import sqlite3
from pkg_resources import parse_version

__version__ = "0.2.1"
__author__ = "Mickael Desfrenes"
__email__ = "desfrenes@gmail.com"
```

Yen 2013.04.08, 將 Python2 的 .next() 改為 next(), 以便在 Python 3 中使用

```
class SQLiteWriter(object):

    """
    In frozen mode (the default), the writer will not alter db schema.
    Just add frozen=False to enable column creation (or just add False
    as second parameter):
```



```

query_writer = SQLiteWriter(":memory:", False)
"""
def __init__(self, db_path=":memory:", frozen=True):
    self.db = sqlite3.connect(db_path)
    self.db.isolation_level = None
    self.db.row_factory = sqlite3.Row
    self.frozen = frozen
    self.cursor = self.db.cursor()
    self.cursor.execute("PRAGMA foreign_keys=ON;")
    self.cursor.execute('PRAGMA encoding = "UTF-8";')
    self.cursor.execute('BEGIN;')
def __del__(self):
    self.db.close()

def replace(self, bean):
    keys = []
    values = []
    write_operation = "replace"
    if "id" not in bean.__dict__:
        write_operation = "insert"
        keys.append("id")
        values.append(None)
    self.__create_table(bean.__class__.__name__)
    columns = self.__get_columns(bean.__class__.__name__)
    for key in bean.__dict__:
        keys.append(key)
        if key not in columns:
            self.__create_column(bean.__class__.__name__, key,
                                type(bean.__dict__[key]))
        values.append(bean.__dict__[key])
    sql = write_operation + " into " + bean.__class__.__name__ + "("
    sql += ",".join(keys) + ") values ("
    sql += ",".join(["?" for i in keys]) + ")"
    self.cursor.execute(sql, values)
    if write_operation == "insert":
        bean.id = self.cursor.lastrowid
    return bean.id

def __create_column(self, table, column, sqltype):
    if self.frozen:
        return
    if sqltype in [float, int, bool]:
        sqltype = "NUMERIC"
    else:
        sqltype = "TEXT"
    sql = "alter table " + table + " add " + column + " " + sqltype

```

```

        self.cursor.execute(sql)

def __get_columns(self, table):
    columns = []
    if self.frozen:
        return columns
    self.cursor.execute("PRAGMA table_info(" + table + ")")
    for row in self.cursor:
        columns.append(row["name"])
    return columns

def __create_table(self, table):
    if self.frozen:
        return
    sql = "create table if not exists " + table + "(id INTEGER PRIMARY KEY AUTOINCREMENT)"
    self.cursor.execute(sql)

def get_rows(self, table_name, sql = "1", replace = None):
    if replace is None : replace = []
    self.__create_table(table_name)
    sql = "SELECT * FROM " + table_name + " WHERE " + sql
    try:
        self.cursor.execute(sql, replace)
        for row in self.cursor:
            yield row
    except sqlite3.OperationalError:
        return

def get_count(self, table_name, sql="1", replace = None):
    if replace is None : replace = []
    self.__create_table(table_name)
    sql = "SELECT count(*) AS cnt FROM " + table_name + " WHERE " + sql
    try:
        self.cursor.execute(sql, replace)
    except sqlite3.OperationalError:
        return 0
    for row in self.cursor:
        return row["cnt"]

def delete(self, bean):
    self.__create_table(bean.__class__.__name__)
    sql = "delete from " + bean.__class__.__name__ + " where id=?"
    self.cursor.execute(sql, [bean.id])

def link(self, bean_a, bean_b):
    self.replace(bean_a)

```

```

self.replace(bean_b)
table_a = bean_a.__class__.__name__
table_b = bean_b.__class__.__name__
assoc_table = self.__create_assoc_table(table_a, table_b)
sql = "replace into " + assoc_table + "(" + table_a + "_id," + table_b
sql += "_id) values(?,?)"
self.cursor.execute(sql,
    [bean_a.id, bean_b.id])

def unlink(self, bean_a, bean_b):
    table_a = bean_a.__class__.__name__
    table_b = bean_b.__class__.__name__
    assoc_table = self.__create_assoc_table(table_a, table_b)
    sql = "delete from " + assoc_table + " where " + table_a
    sql += "_id=? and " + table_b + "_id=?"
    self.cursor.execute(sql,
        [bean_a.id, bean_b.id])

def get_linked_rows(self, bean, table_name):
    bean_table = bean.__class__.__name__
    assoc_table = self.__create_assoc_table(bean_table, table_name)
    sql = "select t.* from " + table_name + " t inner join " + assoc_table
    sql += " a on a." + table_name + "_id = t.id where a."
    sql += bean_table + "_id=?"
    self.cursor.execute(sql, [bean.id])
    for row in self.cursor:
        yield row

def __create_assoc_table(self, table_a, table_b):
    assoc_table = "_".join(sorted([table_a, table_b]))
    if not self.frozen:
        sql = "create table if not exists " + assoc_table + "("
        sql += table_a + "_id NOT NULL REFERENCES " + table_a + "(id) ON DELETE cascade"
        sql += table_b + "_id NOT NULL REFERENCES " + table_b + "(id) ON DELETE cascade"
        sql += " PRIMARY KEY (" + table_a + "_id," + table_b + "_id));"
        self.cursor.execute(sql)
        # no real support for foreign keys until sqlite3 v3.6.19
        # so here's the hack
        if cmp(parse_version(sqlite3.sqlite_version), parse_version("3.6.19")) < 0:
            sql = "create trigger if not exists fk_" + table_a + "_" + assoc_table
            sql += " before delete on " + table_a
            sql += " for each row begin delete from " + assoc_table + " where " + table_a
            self.cursor.execute(sql)
            sql = "create trigger if not exists fk_" + table_b + "_" + assoc_table
            sql += " before delete on " + table_b
            sql += " for each row begin delete from " + assoc_table + " where " + table_b

```

```

        self.cursor.execute(sql)
    return assoc_table

def delete_all(self, table_name, sql = "1", replace = None):
    if replace is None : replace = []
    self.__create_table(table_name)
    sql = "DELETE FROM " + table_name + " WHERE " + sql
    try:
        self.cursor.execute(sql, replace)
    return True
    except sqlite3.OperationalError:
        return False

def commit(self):
    self.db.commit()

class Store(object):
    """
    A SQL writer should be passed to the constructor:

    beans_save = Store(SQLiteWriter(":memory"), frozen=False)
    """
    def __init__(self, SQLWriter):
        self.writer = SQLWriter

    def new(self, table_name):
        new_object = type(table_name,(object,),{})()
        return new_object

    def save(self, bean):
        self.writer.replace(bean)

    def load(self, table_name, id):
        for row in self.writer.get_rows(table_name, "id=?", [id]):
            return self.row_to_object(table_name, row)

    def count(self, table_name, sql = "1", replace=None):
        return self.writer.get_count(table_name, sql, replace if replace is not None else None)

    def find(self, table_name, sql = "1", replace=None):
        for row in self.writer.get_rows(table_name, sql, replace if replace is not None else None):
            yield self.row_to_object(table_name, row)

    def find_one(self, table_name, sql = "1", replace=None):

```

```

try:
    return next(self.find(table_name, sql, replace))
except StopIteration:
    return None

def delete(self, bean):
    self.writer.delete(bean)

def link(self, bean_a, bean_b):
    self.writer.link(bean_a, bean_b)

def unlink(self, bean_a, bean_b):
    self.writer.unlink(bean_a, bean_b)

def get_linked(self, bean, table_name):
    for row in self.writer.get_linked_rows(bean, table_name):
        yield self.row_to_object(table_name, row)

def delete_all(self, table_name, sql = "1", replace=None):
    return self.writer.delete_all(table_name, sql, replace if replace is not None else None)

def row_to_object(self, table_name, row):
    new_object = type(table_name, (object,), {}]()
    for key in row.keys():
        new_object.__dict__[key] = row[key]
    return new_object

def commit(self):
    self.writer.commit()

```

<http://inversionconsulting.blogspot.in/2008/06/proe-vb-api-not-just-for-visual-basic.html>

'Code for Auto Assembly Program Imports pfcls Imports System Imports System.IO

```

Public Class Form1
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
        Dim conn As IpfcAsyncConnection = Nothing
        Dim session As IpfcBaseSession
        Dim pat, pat1 As String
        Dim model As IpfcModel
        Dim modelDesc As IpfcModelDescriptor
        Dim modelDesc1 As IpfcModelDescriptor
        Dim i, k As Integer
        Dim win As IpfcWindow
        Dim loc As String
        Dim modelnames As String() = Nothing
        Dim asmModels As Cstringseq
        Dim drawingPath As String
        Dim drawingName, drawingName1 As String
        Dim drawingName2() As String
        Dim errMsg As String = ""

```

'For Assembly constraints Dim components As IpfcFeatures Dim component As

```

IpfcComponentFeat Dim compConstraints As IpfcComponentConstraints Dim
compConstraint As IpfcComponentConstraint Dim Constraint1 As IpfcCompo-
nentConstraint Dim Constraints1 As ipfcComponentConstraints Dim assembly
As IpfcAssembly Dim assemblyDatums(2) As String Dim componentDatums(2)
As String Dim asmReference As IpfcSelection Dim compReference As IpfcSe-
lection Dim constraintType As String Dim componentModel As IpfcSolid Dim
asmcomp As IpfcComponentFeat

If Not System.Diagnostics.Process.GetProcessesByName("nmsd").Length > 0
Then errMsg = "Name service is not running on the system." End If

If System.Environment.GetEnvironmentVariable("PRO_COMM_MSG_EXE")
= "" Then If errMsg = "" Then errMsg = "Environment variable
'PRO_COMM_MSG_EXE' has not been set." Else

errMsg = errMsg + Chr(13).ToString + "Environment variable 'PRO_COMM_MSG_EXE'
has not been set."

End If End If

If System.Environment.GetEnvironmentVariable("PRO_DIRECTORY") Is
Nothing Then

If errMsg = "" Then errMsg = "Environment variable 'PRO_DIRECTORY'
has not been set." Else

errMsg = errMsg + Chr(13).ToString + "Environment variable 'PRO_DIRECTORY'
has not been set."

End If

End If 'If Services are not running Exit Application

If Not errMsg = "" Then errMsg = errMsg + Chr(13).ToString + "These
may lead to errors in running the application." MsgBox(errMsg, Msg-
BoxStyle.Critical)

Else 'Connect to Pro/Engineer

If conn Is Nothing OrElse Not conn.IsRunning Then conn = (New CCpfcAsync-
Connection).Connect(Nothing, Nothing, Nothing, Nothing) session =
conn.Session

If session Is Nothing Then Throw New Exception("Session does not exist") Else
session.EraseUndisplayedModels()

'Directory for Pro/E Models

loc = "d:".ToLower asmModels = session.ListFiles("*.asm", EpfcFileListOpt.EpfcFILE_LIST_LATEST,
loc)

If asmModels.Count <> 0 Then

k = 0

```

```

For k = 0 To asmModels.Count - 1 drawingPath = asmModels.Item(k)
drawingName = drawingPath.Substring(loc.Length) drawingName1 =
drawingName.Substring(0, drawingName.Length - 4) drawingName2 =
drawingName1.Split("-") pat1 = drawingName.Substring(0, 1).ToUpper
& "SAMPLE" & drawingName2(1).ToUpper & ".prt" drawingName1 =
drawingName1.Substring(drawingName1.Length - 7, 7).ToUpper

```

```

modelDesc1 = (New CCpfcModelDescriptor).Create(0, drawingName, Nothing)
session.RetrieveModel(modelDesc1) session.OpenFile(modelDesc1).Activate()
model = session.CurrentModel assembly = CType(model, IpfcAssembly)
modelDesc = (New CCpfcModelDescriptor).Create(1, pat1, Nothing) componentModel = session.RetrieveModel(modelDesc) pat1 = pat1.Substring(0, pat1.Length - 4) components = assembly.ListFeaturesByType(False, EpfcFeatureType.EpfcFEATTYPE_COMPONENT)

```

'Search through the models in the current assembly

```

For i = 0 To components.Count - 1 component = components.Item(i) modelDesc = component.ModelDescr

```

```

If modelDesc.InstanceName = pat1 Then compConstraints = component.GetConstraints() Constraints1 = New CpfcComponentConstraints
asmcomp = assembly.AssembleComponent(componentModel, Nothing)

```

```

For j = 0 To compConstraints.Count - 1 compConstraint = compConstraints.Item(j) constraintType = constraintTypeToString(compConstraint.Type)
asmReference = compConstraint.AssemblyReference compReference = compConstraint.ComponentReference

```

```

Select Case (constraintType)

```

```

Case "Mate" Constraint1 = (New CCpfcComponentConstraint).Create__(EpfcComponentConstraintType.Mate)
Constraint1.AssemblyReference = asmReference Constraint1.ComponentReference = compReference Constraints1.Insert(Constraints1.Count, Constraint1)

```

```

Case "Align" Constraint1 = (New CCpfcComponentConstraint).Create__(EpfcComponentConstraintType.Align)
Constraint1.AssemblyReference = asmReference Constraint1.ComponentReference = compReference Constraints1.Insert(Constraints1.Count, Constraint1)

```

```

Case "Align Offset" Constraint1 = (New CCpfcComponentConstraint).Create__(EpfcComponentConstraintType.AlignOffset)
Constraint1.AssemblyReference = asmReference Constraint1.ComponentReference = compReference Constraint1.Offset = 10 Constraints1.Insert(Constraints1.Count, Constraint1)

```

```

End Select

```

```

Next

```

```

asmcomp.SetConstraints(Constraints1, Nothing) assembly.Regenerate(Nothing)
session.GetModelWindow(assembly).Repaint() model.Save()

```

```

End If

```

```

Next
session.CurrentWindow.Activate() win = session.CurrentWindow Next
conn.Disconnect(2)
End If
End If
End If
End If
End Sub

Private Function constraintTypeToString(ByVal type As Integer) As String
Select Case (type)
Case EpfcComponentConstraintType.EpfcASM_CONSTRAINT_MATE
Return ("Mate")
Case EpfcComponentConstraintType.EpfcASM_CONSTRAINT_MATE_OFF
Return ("Mate Offset")
Case EpfcComponentConstraintType.EpfcASM_CONSTRAINT_ALIGN Re-
turn ("Align")
Case EpfcComponentConstraintType.EpfcASM_CONSTRAINT_ALIGN_OFF
Return ("Align Offset")
Case EpfcComponentConstraintType.EpfcASM_CONSTRAINT_INSERT
Return ("Insert")
Case EpfcComponentConstraintType.EpfcASM_CONSTRAINT_ORIENT
Return ("Orient")
Case EpfcComponentConstraintType.EpfcASM_CONSTRAINT_CSYS
Return ("Csys")
Case EpfcComponentConstraintType.EpfcASM_CONSTRAINT_TANGENT
Return ("Tangent")
Case EpfcComponentConstraintType.EpfcASM_CONSTRAINT_PNT_ON_SRF
Return ("Point on Surf")
Case EpfcComponentConstraintType.EpfcASM_CONSTRAINT_EDGE_ON_SRF
Return ("Edge on Surf")
Case EpfcComponentConstraintType.EpfcASM_CONSTRAINT_DEF_PLACEMENT
Return ("Default")
Case EpfcComponentConstraintType.EpfcASM_CONSTRAINT_SUBSTITUTE
Return ("Substitute")
Case EpfcComponentConstraintType.EpfcASM_CONSTRAINT_PNT_ON_LINE
Return ("Point on Line")

```



```

Case EpfcComponentConstraintType.EpfcASM_CONSTRAINT_FIX Return
("Fix")

Case      EpfcComponentConstraintType.EpfcASM_CONSTRAINT_AUTO
Return ("Auto")

End Select

Return ("Unrecognized Type") End Function End Class

```

```

/*
    HISTORY

    14-NOV-02   J-03-38   $$1   JCN       Adapted from J-Link examples.
    07-MAR-03   K-01-03   $$2   JCN       UNIX support

*/

function replaceBoltsInAssembly ()
{
    var oldInstance = "PHILLIPS7_8";
    var newInstance = "SLOT7_8";

    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");

    /*-----*\
        Get the current assembly
    /*-----*/
    var session = pfcGetProESession ();
    var assembly = session.CurrentModel;

    if (assembly.Type != pfcCreate ("pfcModelType").MDL_ASSEMBLY)
        throw new Error (0, "Current model is not an assembly");

    /*-----*\
        Get the new instance model for use in replacement
    /*-----*/
    var bolt = session.GetModel ("BOLT", pfcCreate ("pfcModelType").MDL_PART);

    var row = bolt.GetRow (newInstance);

    var newBolt = row.CreateInstance();

    var replaceOps = pfcCreate ("pfcFeatureOperations");

    /*-----*\

```

```

    Visit the assembly components
/*-----*/
var components = assembly.ListFeaturesByType (false,
    pfcCreate ("pfcFeatureType").FEATTYPE_COMPONENT);

for (ii = 0; ii < components.Count; ii++)
{
    var component = components.Item(ii);

    var desc = component.ModelDescr;

    if (desc.InstanceName == oldInstance)
    {
        var replace = component.CreateReplaceOp (newBolt);

        replaceOps.Append (replace);
    }
}

/*-----*/
    Carry out the replacements
/*-----*/
assembly.ExecuteFeatureOps (replaceOps, null);

return;
}

/*=====*/
This function displays each constraint of the component visually on
the screen, and includes a text explanation for each constraint.
/*=====*/

/*=====*/
FUNCTION: highlightConstraints
PURPOSE: Highlights and labels a component's constraints
/*=====*/
function highlightConstraints ()
{

    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");
/*-----*/
    Get the constraints for the component.
/*-----*/
    var session = pfcGetProESession ();

```

```

session.CurrentWindow.SetBrowserSize (0.0);

var options = pfcCreate ("pfcSelectionOptions").Create ("membfeat");
options.MaxNumSels = 1;
var selections = session.Select (options, void null);
if (selections == void null || selections.Count == 0)
    return;

selections.Item(0).UnHighlight();

var feature = selections.Item (0).SelItem;

if (feature.FeatType != pfcCreate ("pfcFeatureType").FEATTYPE_COMPONENT)
    return;

var asmcomp = feature;

var constrs = asmcomp.GetConstraints ();

if (constrs == void null || constrs.Count == 0)
    return;

for (var i = 0; i < constrs.Count; i++)
{
    /*-----*\
    Highlight the assembly reference geometry
    \*-----*/
    var c = constrs.Item (i);

    var asmRef = c.AssemblyReference;

    if (asmRef != void null)
        asmRef.Highlight (pfcCreate ("pfcStdColor").COLOR_ERROR);

    /*-----*\
    Highlight the component reference geometry
    \*-----*/
    var compRef = c.ComponentReference;

    if (compRef != void null)
        compRef.Highlight (pfcCreate ("pfcStdColor").COLOR_WARNING);

    /*-----*\
    Prepare and display the message text.
    \*-----*/

```

```

        var offset = c.Offset;
        var offsetString = "";
        if (offset != void null)
offsetString = ", offset of "+offset;

        var cType = c.Type;
        var cTypeString = constraintTypeToString (cType);

        alert ("Showing constraint " + (i+1) + " of " + constrs.Count + "\n" +
            cTypeString + offsetString + ".");

/*-----*\
Clean up the UI for the next constraint
\*-----*/
        if (asmRef != void null)
        {
            asmRef.UnHighlight ();
        }

        if (compRef != void null)
        {
            compRef.UnHighlight ();
        }
    }
}

/*=====*\
FUNCTION: constraintTypeToString
PURPOSE: Utility: convert the constraint type to a string for printing
\*=====*/
function constraintTypeToString (type /* pfcComponentConstraintType */)
{
    var constrTypeClass = pfcCreate ("pfcComponentConstraintType");
    switch (type)
    {
        case constrTypeClass.ASM_CONSTRAINT_MATE:
            return "(Mate)";
        case constrTypeClass.ASM_CONSTRAINT_MATE_OFF:
            return "(Mate Offset)";
        case constrTypeClass.ASM_CONSTRAINT_ALIGN:
            return "(Align)";
        case constrTypeClass.ASM_CONSTRAINT_ALIGN_OFF:
            return "(Align Offset)";
        case constrTypeClass.ASM_CONSTRAINT_INSERT:
            return "(Insert)";
        case constrTypeClass.ASM_CONSTRAINT_ORIENT:

```

```

        return "(Orient)");
    case constrTypeClass.ASM_CONSTRAINT_CSYS:
        return "(Csys)";
    case constrTypeClass.ASM_CONSTRAINT_TANGENT:
        return "(Tangent)";
    case constrTypeClass.ASM_CONSTRAINT_PNT_ON_SRF:
        return "(Point on Surf)";
    case constrTypeClass.ASM_CONSTRAINT_EDGE_ON_SRF:
        return "(Edge on Surf)";
    case constrTypeClass.ASM_CONSTRAINT_DEF_PLACEMENT:
        return "(Default)";
    case constrTypeClass.ASM_CONSTRAINT_SUBSTITUTE:
        return "(Substitute)";
    case constrTypeClass.ASM_CONSTRAINT_PNT_ON_LINE:
        return "(Point on Line)";
    case constrTypeClass.ASM_CONSTRAINT_FIX:
        return "(Fix)";
    case constrTypeClass.ASM_CONSTRAINT_AUTO:
        return "(Auto)";
    default:
        return "(Unrecognized Type)";
    }
}

/*
The following example demonstrates how to assemble a component into an
assembly, and how to constrain the component by aligning datum planes.
If the complete set of datum planes is not found, the function will show
the component constraint dialog to the user to allow them to adjust the
placement as they wish.
*/

/*=====*\
FUNCTION: UserAssembleByDatums
PURPOSE: Assemble a component by aligning named datums.
\*=====*/
function assembleByDatums (componentFilename /* string as ??????.??? */)
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");
    var interactFlag = false;
    var identityMatrix = pfcCreate ("pfcMatrix3D");
    for (var x = 0; x < 4; x++)
        for (var y = 0; y < 4; y++)
        {
            if (x == y)

```

```

        identityMatrix.Set (x, y, 1.0);
    else
        identityMatrix.Set (x, y, 0.0);
    }
    var transf = pfcCreate ("pfcTransform3D").Create (identityMatrix);
/*-----*\
    Get the current assembly
\*-----*/
    var session = pfcGetProESession ();
    var model = session.CurrentModel;
    if (model == void null || model.Type != pfcCreate ("pfcModelType").MDL_ASSEMBLY)
        throw new Error (0, "Current model is not an assembly.");

    var assembly = model;

    var descr =
        pfcCreate ("pfcModelDescriptor").CreateFromFileName (componentFilename);
    var componentModel = session.GetModelFromDescr (descr);

    if (componentModel == void null)
    {
        componentModel = session.RetrieveModel (descr);
    }

/*-----*\
    Set up the arrays of datum names
\*-----*/
    var asmDatums = new Array ("ASM_D_FRONT", "ASM_D_TOP", "ASM_D_RIGHT");
    var compDatums = new Array ("COMP_D_FRONT",
                                "COMP_D_TOP",
                                "COMP_D_RIGHT");

/*-----*\
    Package the component initially
\*-----*/
    var asmcomp = assembly.AssembleComponent (componentModel,
                                                transf);

/*-----*\
    Prepare the constraints array
\*-----*/
    var constrs = pfcCreate ("pfcComponentConstraints");

    for (var i = 0; i < 3; i++)
    {
/*-----*\

```

```

    Find the assembly datum
/*-----*/
    var asmItem =
        assembly.GetItemByName (pfcCreate ("pfcModelItemType").ITEM_SURFACE,
                                asmDatums [i]);

    if (asmItem == void null)
    {
        interactFlag = true;
        continue;
    }

/*-----*/
    Find the component datum
/*-----*/
    var compItem =
        componentModel.GetItemByName (pfcCreate ("pfcModelItemType").ITEM_SURFACE,
                                      compDatums [i]);

    if (compItem == void null)
    {
        interactFlag = true;
        continue;
    }

/*-----*/
    For the assembly reference, initialize a component path.
    This is necessary even if the reference geometry is in the assembly.
/*-----*/
    var ids = pfcCreate ("intseq");

    var path = pfcCreate ("MpfcAssembly").CreateComponentPath (assembly,
                                                                ids);

/*-----*/
    Allocate the references
/*-----*/
    var MpfcSelect = pfcCreate ("MpfcSelect");
    var asmSel = MpfcSelect.CreateModelItemSelection (asmItem, path);
    var compSel = MpfcSelect.CreateModelItemSelection (compItem, void null);

/*-----*/
    Allocate and fill the constraint.
/*-----*/
    var constr = pfcCreate ("pfcComponentConstraint").Create (
        pfcCreate ("pfcComponentConstraintType").ASM_CONSTRAINT_

```

```

        constr.AssemblyReference = asmSel;
        constr.ComponentReference = compSel;

        constr.Attributes = pfcCreate ("pfcConstraintAttributes").Create (false, false);

        constrs.Append (constr);
    }

    /*-----*\
    Set the assembly component constraints and regenerate the assembly.
    \*-----*/
    asmcomp.SetConstraints (constrs, void null);

    assembly.Regenerate (void null);

    session.GetModelWindow (assembly).Repaint();

    /*-----*\
    If any of the expect datums was not found, prompt the user to constrain
    the new component.
    \*-----*/
    if (interactFlag)
    {
        alert ("Unable to locate all required datum references.  New component is packaged
        asmcomp.RedefineThroughUI();
    }
}

/*
    HISTORY

    14-NOV-02   J-03-38   $$1   JCN   Adapted from J-Link examples.
    07-MAR-03   K-01-03   $$2   JCN   UNIX support

*/

/*
    The following example code shows a utility function that sets angular
    tolerances to a specified range.  For each angular dimension in the current
    model the function gets the dimension value and adds or subtracts the range
    to it to get the upper and lower limits.  The function then initializes a
    pfcDimTolLimits tolerance object and assigns it to the dimension.  The
    function displays each shown dimension.
*/

```



```

function setAngularToleranceToLimits (range /* number */)
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");
    /*-----*\
        Get the current solid model
    \*-----*/
    var session = pfcGetProESession();
    var model = session.CurrentModel;

    if (model == void null || (model.Type != pfcCreate ("pfcModelType").MDL_PART &&
        model.Type != pfcCreate ("pfcModelType").MDL_ASSEMBLY))
        throw new Error (0, "Current model is not a part or assembly.");

    /*-----*\
        Ensure that dimensions will be shown with tolerances
    \*-----*/
    session.SetConfigOption ("tol_display", "yes");

    /*-----*\
        List all model dimensions
    \*-----*/
    var dimensions = model.ListItems (pfcCreate ("pfcModelItemType").ITEM_DIMENSION);

    for (var i = 0; i < dimensions.Count; i++)
    {
        var dimension = dimensions.Item (i);

        /*-----*\
            Check for angular dimensions
        \*-----*/
        var dType = dimension.DimType; // from class pfcBaseDimension

        if (dType == pfcCreate ("pfcDimensionType").DIM_ANGULAR)
        {
            /*-----*\
                Assign the limits tolerance
            \*-----*/
            var dvalue = dimension.DimValue; //from class pfcBaseDimension

            var upper = dvalue + range/2.0;

```

```

    var lower = dvalue - range/2.0;

    limits = pfcCreate ("pfcDimTolLimits").Create(upper, lower);

    dimension.Tolerance = limits; // from class pfcDimension

/*-----*\
    Display the modified dimension
/*-----*/

    var showInstrs =
        pfcCreate ("pfcComponentDimensionShowInstructions").Create (void null);

    dimension.Show (showInstrs);

}
}
}

/*

    HISTORY:

    14-NOV-02    J-03-38    $$1    JCN    Adapted from J-Link examples.
    07-MAR-03    K-01-03    $$2    JCN    UNIX support
    02-Jun-03    K-01-08    $$3    JCN    Fix comparison for mouse button
*/

/*
    This example demonstrates the use of mouse tracking methods to draw graphics
    on the screen. The static method DrawRubberbandLine prompts the user to
    pick a screen point. The example uses 'complement mode' to cause the line
    to display and erase as the user moves the mouse around the window.

    NOTE: This example uses the method transformPosition to convert the
    coordinates into the 3D coordinate system of a solid model, if one is
    displayed.
*/
function drawRubberbandLine ()
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");
/*-----*\
    Select the first end of the rubber band line. Expect the user to pick with
    the left mouse button.
/*-----*/

```

```

var session = pfcGetProESession();

session.CurrentWindow.SetBrowserSize (0.0);

mouse = session.UIGetNextMousePick (pfcCreate ("pfcMouseButton").MOUSE_BTN_LEFT);

/*-----*\
   Transform screen point -> model location, if necessary
/*-----*/
var firstPos = transformPosition (session, mouse.Position);

/*-----*\
   Set graphics mode to complement, so that graphics erase after use.
/*-----*/
var currentMode = session.CurrentGraphicsMode;
session.CurrentGraphicsMode = pfcCreate ("pfcGraphicsMode").DRAW_GRAPHICS_COMPLEMENT;

/*-----*\
   Get current mouse position.
/*-----*/
var mouse = session.UIGetCurrentMouseStatus (false);
while (mouse.SelectedButton == pfcCreate ("pfcMouseButton").MouseButton_nil)
{
    session.SetPenPosition (firstPos);
    var secondPos = transformPosition (session, mouse.Position);

/*-----*\
   Draw rubberband line
/*-----*/
    session.DrawLine (secondPos);

    mouse = session.UIGetCurrentMouseStatus (false);

/*-----*\
   Erase previously drawn line
/*-----*/
    session.SetPenPosition (firstPos);
    session.DrawLine (secondPos);
}

session.CurrentGraphicsMode = currentMode;

return;
}

```

```

/* This method transforms the 2D screen coordinates into
   3D model coordinates - if necessary. */
function transformPosition (s /* pfcSession */, inPnt /* pfcPoint3D */)
{
    var mdl = s.CurrentModel;

    /*-----*\
    Skip transform if not in 3D model
    \*-----*/
    if (mdl == void null)
        return inPnt;

    var type = mdl.Type;
    var modelTypeClass = pfcCreate ("pfcModelType");
    var isSolid = ((type == modelTypeClass.MDL_PART) ||
        (type == modelTypeClass.MDL_ASSEMBLY) ||
        (type == modelTypeClass.MDL_MFG));
    if (!isSolid)
        return inPnt;

    /*-----*\
    Get current view's orientation and invert it
    \*-----*/
    var currView = mdl.GetCurrentView();
    var invOrient = currView.Transform;
    invOrient.Invert();

    /*-----*\
    Get the point in the model csys
    \*-----*/
    var outPnt = invOrient.TransformPoint (inPnt);

    return outPnt;
}

/*
    HISTORY

    23-JUL-07   L-01-35   $$1   SNV       Created and submitted

*/

/*=====*\
FUNCTION : placeDetailSymbol()
PURPOSE  : This function creates a free instance of a symbol
           definition with drawing unit heights, variable text and

```

```

        groups. A symbol is placed with no leaders at a specified
        location.
    \*=====*/
function placeDetailSymbol(groupName , variableText, symHeight)
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");

    /*-----*\
        Get the current drawing
    \*-----*/
    var session = pfcGetProESession ();
    var drawing = session.CurrentModel;

    if (drawing.Type != pfcCreate ("pfcModelType").MDL_DRAWING)
        throw new Error (0, "Current model is not a drawing");

    /*-----*\
        Retrieve the symbol definition from the system
    \*-----*/
    var symDef = drawing.RetrieveSymbolDefinition ("detail_symbol_example",
        "./", void null, void null);

    /*-----*\
        Select the locations for the symbol
    \*-----*/
    var browserSize = session.CurrentWindow.GetBrowserSize();
    session.CurrentWindow.SetBrowserSize (0.0);

    var stop = false;
    var point
    while (!stop)
    {
        var mouse =
            session.UIGetNextMousePick (pfcCreate ("pfcMouseButton").MouseButton_nil);

        if (mouse.SelectedButton == pfcCreate ("pfcMouseButton").MOUSE_BTN_LEFT)
        {
            point = mouse.Position;
        }
        else
            stop = true;
    }

    session.CurrentWindow.SetBrowserSize (browserSize);

```

```

/*-----*\
    Allocate the symbol instance instructions
/*-----*/
var instrs =
    pfcCreate ("pfcDetailSymbolInstInstructions").Create (symDef);

/*-----*\
    Set the symbol height in drawing units
/*-----*/
if (symHeight > 0)
{
    instrs.ScaledHeight = symHeight;
}

/*-----*\
    Set text to the variable text in the symbol. This will be displayed
    instead of the text defined when creating the symbol
/*-----*/
if (variableText != void null)
{
    var varText = pfcCreate ("pfcDetailVariantText").Create("VAR_TEXT" , variableText)
    var varTexts = pfcCreate("pfcDetailVariantTexts");
    varTexts.Append(varText);

    instrs.TextValues = varTexts;
}

/*-----*\
    Display the groups in symbol depending on group name
/*-----*/
if (groupName == "ALL")
    instrs.SetGroups(pfcCreate("pfcDetailSymbolGroupOption").DETAIL_SYMBOL_GROUP_ALL , m
else if (groupName == "NONE")
    instrs.SetGroups(pfcCreate("pfcDetailSymbolGroupOption").DETAIL_SYMBOL_GROUP_NONE , m
else
{
    var allGroups = instrs.SymbolDef.ListSubgroups();
    group = getGroup(allGroups , groupName );
    if (group != void null)
    {
        groups = pfcCreate("pfcDetailSymbolGroups");
        groups.Append(group);
        instrs.SetGroups (pfcCreate("pfcDetailSymbolGroupOption").DETAIL_SYMBOL_GROUP_CU
    }
}

```

```

/*-----*\
    Set the attachment structure
/*-----*/
var position = pfcCreate ("pfcFreeAttachment").Create (point);
var allAttachments = pfcCreate ("pfcDetailLeaders").Create ();
allAttachments.ItemAttachment = position;

instrs.InstAttachment = allAttachments;

/*-----*\
    Create and display the symbol
/*-----*/
var symInst = drawing.CreateDetailItem (instrs);
symInst.Show();
}

/*=====*\
FUNCTION : getGroup()
PURPOSE  : Return the specific group depending on the group name.
\=====*/
function getGroup(groups, groupName)
{
    var group;
    var groupInstrs;

    if (groups.Count <=0 )
    {
        return null;
    }

/*-----*\
    Loop through all the groups in the symbol and return the group with
    the selected name
/*-----*/
    for(var i=0;i<groups.Count;i++)
    {
        group = groups.Item(i);
        groupInstrs = group.GetInstructions();

        if (groupInstrs.Name == groupName)
            return group;
    }
    return null;
}

```

```

/*
    HISTORY

    14-NOV-02   J-03-38   $$1   JCN   Adapted from J-Link examples.
    07-MAR-03   K-01-03   $$2   JCN   UNIX support
    08-JAN-08   L-01-42   $$3   SNV   Fix for SPR 1430116
*/

/*=====*\
FUNCTION: createDrawingFromTemplate
PURPOSE:  Create a new drawing using a predefined template.
\*=====*/
function createDrawingFromTemplate (newDrawingName /* string */)
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");

    var predefinedTemplate = "c_drawing";

    if (newDrawingName == "")
    {
        throw new Error ("Please supply a drawing name.  Aborting...");
    }

    /*-----*\
    Use the current model to create the drawing.
    \*-----*/
    var session = pfcGetProESession ();
    var solid = session.CurrentModel;
    modelTypeClass = pfcCreate ("pfcModelType");

    if (solid == void null || (solid.Type != modelTypeClass.MDL_PART &&
        solid.Type != modelTypeClass.MDL_ASSEMBLY))
    {
        throw new Error ("Current model is not usable for new drawing.  Aborting...");
    }

    var options = pfcCreate ("pfcDrawingCreateOptions");
    options.Append (pfcCreate ("pfcDrawingCreateOption").DRAWINGCREATE_DISPLAY_DRAWING);

    /*-----*\
    Create the required drawing.
    \*-----*/
    var drw = session.CreateDrawingFromTemplate (newDrawingName,
        predefinedTemplate,
        solid.Descr, options);

```



```
}
```

```
/*=====*\
FUNCTION : listSheets()
PURPOSE  : Command to list drawing sheet info in an information window
\*=====*/
function listSheets()
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");

/*-----*\
    Open a browser window to contain the information to be displayed
\*-----*/

    var newWin = window.open ('', "_LS", "scrollbars");
    if (pfcIsWindows())
    {
        newWin.resizeTo (300, screen.height/2.0);
        newWin.moveTo (screen.width-300, 0);
    }
    newWin.document.writeln ("<html><head></head><body>");

/*-----*\
    Get the current drawing
\*-----*/
    var session = pfcGetProESession ();
    var drawing = session.CurrentModel;

    if (drawing.Type != pfcCreate ("pfcModelType").MDL_DRAWING)
        throw new Error (0, "Current model is not a drawing");

/*-----*\
    Get the number of sheets
\*-----*/
    var sheets = drawing.NumberOfSheets;

    for (i = 1; i <= sheets; i++)
    {
/*-----*\
        Get the drawing sheet size etc.
\*-----*/
        var info = drawing.GetSheetData (i);

        var format = drawing.GetSheetFormat (i);
    }
}
```

```

/*-----*\
    Print the information to the window
\*-----*/

    var unit = "unknown";
    var lengthUnitClass = pfcCreate ("pfcLengthUnitType");

    switch (info.Units.GetType())
    {
    case lengthUnitClass.LENGTHUNIT_INCH:
        unit = "inches";
        break;
    case lengthUnitClass.LENGTHUNIT_FOOT:
        unit = "feet";
        break;
    case lengthUnitClass.LENGTHUNIT_MM:
        unit = "mm";
        break;
    case lengthUnitClass.LENGTHUNIT_CM:
        unit = "cm";
        break;
    case lengthUnitClass.LENGTHUNIT_M:
        unit = "m";
        break;
    case lengthUnitClass.LENGTHUNIT_MCM:
        unit = "mcm";
        break;
    }

    newWin.document.writeln ("<h2>Sheet "+ i + "</h2>");
    newWin.document.writeln ("<table>");
    newWin.document.writeln (" <tr><td> Width </td><td> "+
        info.Width + " </td></tr> ");
    newWin.document.writeln (" <tr><td> Height </td><td> "+
        info.Height + " </td></tr> ");
    newWin.document.writeln (" <tr><td> Units </td><td> "+
        unit + " </td></tr> ");

    var formatName;
    if (format == void null)
formatName = "none";
    else
formatName = format.FullName;
    newWin.document.writeln (" <tr><td> Format </td><td> "+

```

```

        formatName + " </td></tr> ");
    newWin.document.writeln ("</table>");
    newWin.document.writeln ("<br>");

}
newWin.document.writeln ("</body></html>");
}

/*=====*\
FUNCTION : listViews()
PURPOSE : Command to list view info in an information window
\*=====*/
function listViews()
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");

/*-----*\
    Open a browser window to contain the information to be displayed
\*-----*/

    var newWin = window.open ('', "_LV", "scrollbars");
    if (pfcIsWindows())
    {
        newWin.resizeTo (300, screen.height/2.0);
        newWin.moveTo (screen.width-300, 0);
    }
    newWin.document.writeln ("<html><head></head><body>");

/*-----*\
    Get the current drawing
\*-----*/
    var session = pfcGetProESession ();
    var drawing = session.CurrentModel;

    if (drawing.Type != pfcCreate ("pfcModelType").MDL_DRAWING)
        throw new Error (0, "Current model is not a drawing");

/*-----*\
    Collect the views
\*-----*/

    var views = drawing.List2DViews ();

    for(i=0; i<views.Count; i++)

```

```

{

    var view = views.Item (i);

    /*-----*\
    Get the name & sheet number for this view
    \*-----*/
    var viewName = view.Name;
    var sheetNo = view.GetSheetNumber ();

    /*-----*\
    Get the name of the solid that the view contains
    \*-----*/
    var solid = view.GetModel ();
    var descr = solid.Descr;

    /*-----*\
    Get the outline, scale, and display state
    \*-----*/
    var outline = view.Outline;
    var scale = view.Scale;
    var display = view.Display;

    /*-----*\
    Write the information to the browser window file
    \*-----*/
    displayStyleClass = pfcCreate ("pfcDisplayStyle");
    var dispStyle;
    switch(display.Style)
    {
    case displayStyleClass.DISPSTYLE_DEFAULT:
        dispStyle = "default";
        break;
    case displayStyleClass.DISPSTYLE_WIREFRAME:
        dispStyle = "wireframe";
        break;
    case displayStyleClass.DISPSTYLE_HIDDEN_LINE:
        dispStyle = "hidden line";
        break;
    case displayStyleClass.DISPSTYLE_NO_HIDDEN:
        dispStyle = "no hidden";
        break;
    case displayStyleClass.DISPSTYLE_SHADED:
        dispStyle = "shaded";
        break;
    }
}

```

```

newWin.document.writeln ("<h2>View "+ viewName + "</h2>");
newWin.document.writeln ("<table>");
newWin.document.writeln (" <tr><td> Sheet </td><td> "+
    sheetNo + " </td></tr> ");
newWin.document.writeln (" <tr><td> Model </td><td> "+
    descr.GetFullName() + " </td></tr> ");
newWin.document.writeln (" <tr><td> Outline </td><td> ");
newWin.document.writeln ("<table><tr><td> <i>Lower left:</i> </td><td>");
newWin.document.writeln (outline.Item (0).Item (0) + ", " +
    outline.Item (0).Item(1) + ", " +
    outline.Item (0).Item(2));
newWin.document.writeln ("</td></tr><tr><td> <i>Upper right:</i></td><td>");
newWin.document.writeln (outline.Item (1).Item (0) + ", " +
    outline.Item (1).Item(1) + ", " +
    outline.Item (1).Item(2));
newWin.document.writeln ("</td></tr></table></td>");
newWin.document.writeln (" <tr><td> Scale </td><td> "+ scale +
    " </td></tr> ");
newWin.document.writeln (" <tr><td> Display style </td><td> "+
    dispStyle + " </td></tr>");
newWin.document.writeln ("</table>");
newWin.document.writeln ("<br>");
}
newWin.document.writeln ("</body></html>");
}

/*=====*\
FUNCTION: drawingSolidReplace()
PURPOSE: Replaces all instance solid models in a drawing with their
         generic. Similar to the Pro/ENGINEER behavior,
         the function will not replace models if the target generic
         model is already present in the drawing.
\*=====*/
function replaceModels()
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");
    /*-----*\
    Get the current drawing
    \*-----*/
    var session = pfcGetProESession ();
    var drawing = session.CurrentModel;

    if (drawing.Type != pfcCreate ("pfcModelType").MDL_DRAWING)
        throw new Error (0, "Current model is not a drawing");

```

```

/*-----*\
Visit the drawing models.
\*-----*/
var solids = drawing.ListModels ();

/*-----*\
Loop on all of the drawing models.
\*-----*/
for (i = 0; i < solids.Count; i++)
{
    var solid = solids.Item (i);
}
/*-----*\
If the generic is not an instance, continue (Parent property
from class pfcFamilyMember)
\*-----*/
var generic = solid.Parent;

if (generic == void null)
    continue;

/*-----*\
Replace all instances with their (top-level) generic.
\*-----*/
try
{
    drawing.ReplaceModel (solid, generic, true);
}
catch (err)
{
    if (pfcGetExceptionType (err) == "pfcXToolkitFound")
        ; // Target generic is already in drawing; do nothing
    else
        throw err;
}
}

/*=====*\
FUNCTION : util_solidFind()
PURPOSE  : Utility to select a solid using the file browser and retrieve
           it if it is not already in session.
\*=====*/
function util_solidFind(filename /* string "???.???" format */)
{
/*-----*\

```

```

    Find it, if its in session and return it
    /*-----*/
    var session = pfcGetProESession ();
    var mdlDescr =
        pfcCreate ("pfcModelDescriptor").CreateFromFileName (filename);
    var mdl = session.GetModelFromDescr (mdlDescr);

    if (mdl != void null)
        return mdl;

    /*-----*/
    Try to retrieve the solid
    /*-----*/
    mdl = session.RetrieveModel (mdlDescr);

    if (mdl != void null)
        return mdl;

    throw new Error (0,
        "Model "+filename+" cannot be found or retrieved.");

    return void null;
}

/*=====*/
FUNCTION : createSheetAndViews()
PURPOSE  : Create a new drawing sheet with a general, and two
           projected, views of a selected solid
/*=====*/
function createSheetAndViews(solidName /* string as ????.??? */)
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");
    /*-----*/
    Get the current drawing, create a new sheet
    /*-----*/
    var session = pfcGetProESession ();
    var drawing = session.CurrentModel;

    if (drawing.Type != pfcCreate ("pfcModelType").MDL_DRAWING)
        throw new Error (0, "Current model is not a drawing");

    var sheetNo = drawing.AddSheet ();
    drawing.CurrentSheetNumber = sheetNo;

```

```

/*-----*\
    Find the solid model, if its in session
/*-----*/
var mdlDescr =
    pfcCreate ("pfcModelDescriptor").CreateFromFileName (solidName);
var solidMdl = session.GetModelFromDescr (mdlDescr);

if (solidMdl == void null)
{
/*-----*\
    If its not found, try to retrieve the solid model
/*-----*/
    solidMdl = session.RetrieveModel (mdlDescr);

    if (solidMdl == void null)
        throw new Error (0,
            "Model "+solidName+" cannot be found or retrieved.");
}

/*-----*\
    Try to add it to the drawing
/*-----*/
try
{
    drawing.AddModel (solidMdl);
}
catch (err)
{
    if (pfcGetExceptionType (err) == "pfcXToolkitInUse")
        ; // model is already in this drawing, nothing to do
    else
        throw err;
}

/*-----*\
    Create a general view from the Z axis direction at a predefined location
/*-----*/
var matrix = pfcCreate ("pfcMatrix3D");
for (i = 0; i < 4; i++)
    for (j = 0; j < 4; j++)
    {
        if (i == j)
            matrix.Set (i, j, 1.0);
        else
            matrix.Set (i, j, 0.0);
    }

```



```

var transf = pfcCreate ("pfcTransform3D").Create (matrix);

var pos = pfcCreate ("pfcPoint3D");
pos.Set (0, 200.0);
pos.Set (1, 600.0);
pos.Set (2, 0.0);

var instrs =
    pfcCreate ("pfcGeneralViewCreateInstructions").Create (solidMdl,
                                                            sheetNo, pos, transf);

var genView = drawing.CreateView (instrs);

/*-----*\
   Get the position and size of the new view
  \*-----*/
var outline = genView.Outline;

/*-----*\
   Create a projected view to the right of the general view
  \*-----*/
pos.Set (0, outline.Item (1).Item (0) + (outline.Item(1).Item(0) -
                                         outline.Item (0).Item (0)));
pos.Set (1, (outline.Item (0).Item(1) + outline.Item (1).Item(1))/2);

instrs =
    pfcCreate ("pfcProjectionViewCreateInstructions").Create (genView,
                                                            pos);

drawing.CreateView (instrs);

/*-----*\
   Create a projected view below the general view
  \*-----*/
pos.Set (0, (outline.Item (0).Item(0) + outline.Item (1).Item(0))/2);
pos.Set (1, outline.Item (0).Item (1) - (outline.Item(1).Item(1) -
                                         outline.Item (0).Item (1)));

instrs =
    pfcCreate ("pfcProjectionViewCreateInstructions").Create (genView,
                                                            pos);

drawing.CreateView (instrs);
}

```

```

/*=====*\
FUNCTION : lineEntityCreate()
PURPOSE  : Utility to create a line entity between specified points
/*=====*/
function lineEntityCreate()
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");

    var color = pfcCreate ("pfcStdColor").COLOR_QUILT;
    var session = pfcGetProESession ();

/*-----*\
    Get the current drawing & its background view
/*-----*/
    var drawing = session.CurrentModel;

    if (drawing.Type != pfcCreate ("pfcModelType").MDL_DRAWING)
        throw new Error (0, "Current model is not a drawing");

    var currSheet = drawing.CurrentSheetNumber;
    var view = drawing.GetSheetBackgroundView (currSheet);

/*-----*\
    Select the endpoints of the line
/*-----*/
    session.CurrentWindow.SetBrowserSize (0.0);

    var left = pfcCreate ("pfcMouseButton").MOUSE_BTN_LEFT;
    var mouse1 = session.UIGetNextMousePick (left);
    var start = mouse1.Position;
    var mouse2 = session.UIGetNextMousePick (left);
    var end = mouse2.Position;

/*-----*\
    Allocate and initialize a curve descriptor
/*-----*/
    var geom = pfcCreate ("pfcLineDescriptor").Create (start, end);

/*-----*\
    Allocate data for the draft entity
/*-----*/
    var instrs = pfcCreate ("pfcDetailEntityInstructions").Create (geom,
                                                                    view);

```

```

/*-----*|
  Set the color to the specified Pro/ENGINEER predefined color
/*-----*/
var rgb = session.GetRGBFromStdColor (color);
instrs.Color = rgb;

/*-----*|
  Create the entity
/*-----*/
drawing.CreateDetailItem (instrs);

/*-----*|
  Display the entity
/*-----*/
session.CurrentWindow.Repaint();
}

/*=====|
FUNCTION : createSurfNote()
PURPOSE  : Utility to create a note that documents the surface name or id.
The note text will be placed at the upper right corner of the selected view.
\=====*/
function createSurfNote()
{
  if (!pfcIsWindows())
    netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");
/*-----*|
  Get the current drawing & its background view
/*-----*/
var session = pfcGetProESession ();
var drawing = session.CurrentModel;

if (drawing.Type != pfcCreate ("pfcModelType").MDL_DRAWING)
  throw new Error (0, "Current model is not a drawing");

/*-----*|
  Interactively select a surface in a drawing view
/*-----*/
var browserSize = session.CurrentWindow.GetBrowserSize();
session.CurrentWindow.SetBrowserSize(0.0);

var options = pfcCreate ("pfcSelectionOptions").Create ("surface");
options.MaxNumSels = 1;

var sels = session.Select (options, void null);

```

```

var selSurf = sels.Item (0);
var item = selSurf.SelItem;

var name = item.GetName();
if (name == void null)
    name = new String ("Surface ID "+item.Id);

session.CurrentWindow.SetBrowserSize(browserSize);

/*-----*\
    Allocate a text item, and set its properties
\*-----*/
var text = pfcCreate ("pfcDetailText").Create (name);

/*-----*\
    Allocate a new text line, and add the text item to it
\*-----*/
var texts = pfcCreate ("pfcDetailTexts");
texts.Append (text);

var textLine = pfcCreate ("pfcDetailTextLine").Create (texts);

var textLines = pfcCreate ("pfcDetailTextLines");
textLines.Append (textLine);

/*-----*\
    Set the location of the note text
\*-----*/
var dwgView = selSurf.SelView2D;
var outline = dwgView.Outline;
var textPos = outline.Item (1);

// Force the note to be slightly beyond the view outline boundary
textPos.Set (0, textPos.Item (0) + 0.25 * (textPos.Item (0) -
        outline.Item (0).Item(0)));
textPos.Set (1, textPos.Item (1) + 0.25 * (textPos.Item (1) -
        outline.Item (0).Item(1)));

var position = pfcCreate ("pfcFreeAttachment").Create (textPos);
position.View = dwgView;

/*-----*\
    Set the attachment for the note leader
\*-----*/
var leaderToSurf = pfcCreate ("pfcParametricAttachment").Create (selSurf);

```

```

/*-----*\
    Set the attachment structure
/*-----*/
var allAttachments = pfcCreate ("pfcDetailLeaders").Create ();
allAttachments.ItemAttachment = position;
allAttachments.Leaders = pfcCreate ("pfcAttachments");
allAttachments.Leaders.Append (leaderToSurf);

/*-----*\
    Allocate a note description, and set its properties
/*-----*/
var instrs = pfcCreate ("pfcDetailNoteInstructions").Create (textLines);

instrs.Leader = allAttachments;

/*-----*\
    Create the note
/*-----*/
var note = drawing.CreateDetailItem (instrs);

/*-----*\
    Display the note
/*-----*/
note.Show ();
}

/*=====*\
FUNCTION : placeSymInst()
PURPOSE  : Place a CG symbol with no leaders at a specified location
/*=====*/
function placeSymInst()
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");
/*-----*\
    Get the current drawing
/*-----*/
var session = pfcGetProESession ();
var drawing = session.CurrentModel;

if (drawing.Type != pfcCreate ("pfcModelType").MDL_DRAWING)
    throw new Error (0, "Current model is not a drawing");

/*-----*\
    Retrieve the symbol definition from the system
/*-----*/

```

```

var symDef = drawing.RetrieveSymbolDefinition ("CG",
        void null, void null, void null);

/*-----*\
  Select the locations for the symbol
/*-----*/
var browserSize = session.CurrentWindow.GetBrowserSize();
session.CurrentWindow.SetBrowserSize (0.0);

var stop = false;
var points = pfcCreate ("pfcPoint3Ds");
while (!stop)
{
    var mouse =
    session.UIGetNextMousePick (pfcCreate ("pfcMouseButton").MouseButton_nil);

    if (mouse.SelectedButton ==
        pfcCreate ("pfcMouseButton").MOUSE_BTN_LEFT)
    {
        points.Append (mouse.Position);
    }
    else
    stop = true;
}

session.CurrentWindow.SetBrowserSize (browserSize);

/*-----*\
  Allocate the symbol instance instructions
/*-----*/
var instrs =
    pfcCreate ("pfcDetailSymbolInstInstructions").Create (symDef);
var position = pfcCreate ("pfcFreeAttachment").Create (points.Item (0));
var allAttachments = pfcCreate ("pfcDetailLeaders").Create ();
for (i = 0; i < points.Count; i++)
{

/*-----*\
  Set the location of the note text
/*-----*/
    position.AttachmentPoint = points.Item (i);

/*-----*\
  Set the attachment structure
/*-----*/
    allAttachments.ItemAttachment = position;

```

```

        instrs.InstAttachment = allAttachments;

/*-----*\
    Create and display the symbol
\*-----*/
    var symInst = drawing.CreateDetailItem (instrs);
    symInst.Show();
}

/*=====*\
FUNCTION : createGroup()
PURPOSE  : Command to create a new group with selected items
\*=====*/
function createGroup (groupName /* string */)
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");
/*-----*\
    Select notes, draft entities, symbol instances
\*-----*/
    var session = pfcGetProESession ();
    var selOptions =
        pfcCreate ("pfcSelectionOptions").Create ("any_note,draft_ent,dtl_symbol");
    var selections = session.Select (selOptions, void null);

    if (selections == void null || selections.Count == 0)
        return;

/*-----*\
    Allocate and fill a sequence with the detail item handles
\*-----*/
    var items = pfcCreate ("pfcDetailItems");

    for (i = 0; i < selections.Count; i ++)
    {
        items.Append (selections.Item (i).SelItem);
    }

/*-----*\
    Get the drawing which will own the group
\*-----*/
    var drawing = items.Item (0).DBParent;

```

```

/*-----*\
    Allocate group data and set the group items
/*-----*/
    var instrs =
        pfcCreate ("pfcDetailGroupInstructions").Create (groupName, items);

/*-----*\
    Create the group
/*-----*/
    drawing.CreateDetailItem (instrs);
}

/*=====*\
FUNCTION : createTableOfPoints()
PURPOSE  : Command to create a table of points
\*=====*/
function createTableOfPoints()
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");

    var widths = new Array ();
    widths [0] = 8.0;
    widths [1] = 10.0;
    widths [2] = 10.0;
    widths [3] = 10.0;

/*-----*\
    Select a coordinate system. This defines the model (the top one
    in that view), and the reference for the datum point positions.
/*-----*/
    var session = pfcGetProESession ();

    session.CurrentWindow.SetBrowserSize (0.0);

    var selOptions = pfcCreate ("pfcSelectionOptions").Create("csys");
    selOptions.MaxNumSels = 1;
    var selections = session.Select (selOptions, void null);

    if (selections == void null || selections.Count == 0)
        return;

/*-----*\
    Extract the csys handle, and assembly path, and view handle.
/*-----*/
    var selItem = selections.Item (0).SelItem;

```



```

var selPath = selections.Item (0).Path;
var selView = selections.Item (0).SelView2D;

if (selView == void null)
    throw new Error (0, "Must select coordinate system from a drawing view.");

var drawing = selView.DBParent;

/*-----*\
   Extract the csys location (property CoordSys from class pfcCoordSystem)
\*-----*/
var csysTransf = selItem.CoordSys;
csysTransf.Invert ();

/*-----*\
   Extract the cys name
\*-----*/
var csysName = selItem.GetName();

/*-----*\
   Get the root solid, and the transform from the root to the
   component owning the csys
\*-----*/

var asmTransf = void null;
var rootSolid = selItem.DBParent;
if (selPath != void null)
{
    rootSolid = selPath.Root;
    asmTransf = selPath.GetTransform(false);
}

/*-----*\
   Get a list of datum points in the model
\*-----*/

var points = rootSolid.ListItems (
    pfcCreate ("pfcModelItemType").ITEM_POINT);

if (points == void null || points.Count == 0)
    return;

/*-----*\
   Set the table position
\*-----*/
var location = pfcCreate ("pfcPoint3D");

```

```

location.Set (0, 200.0);
location.Set (1, 600.0);
location.Set (2, 0.0);

/*-----*\
  Setup the table creation instructions
/*-----*/
var instrs =
  pfcCreate ("pfcTableCreateInstructions").Create (location);

instrs.SizeType =
  pfcCreate ("pfcTableSizeType").TABLESIZE_BY_NUM_CHARS;

var columnInfo = pfcCreate ("pfcColumnCreateOptions");

for (i = 0; i < widths.length; i++)
{
  var column = pfcCreate ("pfcColumnCreateOption").Create (
    pfcCreate ("pfcColumnJustification").COL_JUSTIFY_LEFT,
    widths [i]);
  columnInfo.Append (column);
}
instrs.ColumnData = columnInfo;

var rowInfo = pfcCreate ("realseq");

for (i = 0; i < points.Count + 2; i++)
{
  rowInfo.Append (1.0);
}
instrs.RowHeights = rowInfo;

/*-----*\
  Create the table
/*-----*/
var dwgTable = drawing.CreateTable (instrs);

/*-----*\
  Merge the top row cells to form the header
/*-----*/
var topLeft = pfcCreate ("pfcTableCell").Create (1, 1);
var bottomRight = pfcCreate ("pfcTableCell").Create (1, 4);
dwgTable.MergeRegion (topLeft, bottomRight, void null);

/*-----*\
  Write header text specifying model and csys

```

```

/*-----*/
writeTextInCell (dwgTable, 1, 1,
    "Datum points for "+rootSolid.FileName + " w.r.t. csys "+csysName);

/*-----*/
Add subheadings to columns
/*-----*/
writeTextInCell (dwgTable, 2, 1, "Point");
writeTextInCell (dwgTable, 2, 2, "X");
writeTextInCell (dwgTable, 2, 3, "Y");
writeTextInCell (dwgTable, 2, 4, "Z");

/*-----*/
For each datum point...
/*-----*/
for(p=0; p<points.Count; p++)
{
    var point = points.Item (p);

/*-----*/
Add the point name to column 1
/*-----*/
    writeTextInCell (dwgTable, p+3, 1, point.GetName());

/*-----*/
Transform the location w.r.t to the csys
/*-----*/
    var trfPoint = point.Point;
    if (asmTransf != void null)
trfPoint = asmTransf.TransformPoint (point.Point);
    trfPoint = csysTransf.TransformPoint (trfPoint);

/*-----*/
Add the XYZ to column 2,3,4
/*-----*/
    writeTextInCell (dwgTable, p+3, 2, trfPoint.Item (0));
    writeTextInCell (dwgTable, p+3, 3, trfPoint.Item (1));
    writeTextInCell (dwgTable, p+3, 4, trfPoint.Item (2));
}

/*-----*/
Display the table
/*-----*/
dwgTable.Display ();
}

/*=====*/
FUNCTION : writeTextInCell()

```

```

PURPOSE : Utility to add one text line to a table cell
\=====*/
function writeTextInCell(table /* pfcTable */, row /* integer */,
                        col /* integer */, text /* string */)
{
    var cell = pfcCreate ("pfcTableCell").Create (row, col);
    var lines = pfcCreate ("stringseq");
    lines.Append (text);
    table.SetText (cell, lines);
}

/*=====*\
FUNCTION: createPointDims()
PURPOSE : Command to create dimensions to each of the models' datum points
\=====*/
function createPointDims()
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");
    var hBaseline = void null;
    var vBaseline = void null;

    /*-----*\
    Select a coordinate system. This defines the model (the top one
    in that view), and the common attachments for the dimensions
    \-----*/
    var session = pfcGetProESession ();

    session.CurrentWindow.SetBrowserSize (0.0);

    var selOptions = pfcCreate ("pfcSelectionOptions").Create("csys");
    selOptions.MaxNumSels = 1;
    var selections = session.Select (selOptions, void null);

    if (selections == void null || selections.Count == 0)
        return;

    /*-----*\
    Extract the csys handle, and assembly path, and view handle.
    \-----*/
    var csysSel = selections.Item (0);
    var selItem = csysSel.SelItem;
    var selPath = csysSel.Path;
    var selView = csysSel.SelView2D;
    var selPos = csysSel.Point;

```

```

    if (selView == void null)
        throw new Error (0, "Must select coordinate system from a drawing view.");

    var drawing = selView.DBParent;

    /*-----*\
    Get the root solid, and the transform from the root to the
    component owning the csys
    \*-----*/

    var asmTransf = void null;
    var rootSolid = selItem.DBParent;
    if (selPath != null)
    {
        rootSolid = selPath.Root;
        asmTransf = selPath.GetTransform(true);
    }

    /*-----*\
    Get a list of datum points in the model
    \*-----*/

    var points =
        rootSolid.ListItems (pfcCreate ("pfcModelItemType").ITEM_POINT);

    if (points == void null || points.Count == 0)
        return;

    /*-----*\
    Calculate where the csys is located on the drawing
    \*-----*/

    var csysPos = selPos;
    if (asmTransf != void null)
    {
        csysPos = asmTransf.TransformPoint (selPos);
    }
    var viewTransf = selView.GetTransform();
    csysPos = viewTransf.TransformPoint (csysPos);

    var csys3DPos = pfcCreate ("pfcVector2D");

    csys3DPos.Set (0, csysPos.Item (0));
    csys3DPos.Set (1, csysPos.Item (1));

    /*-----*\

```

```

    Get the view outline
/*-----*/
    var outline = selView.Outline;

/*-----*/
    Allocate the attachment arrays
/*-----*/
    var senses = pfcCreate ("pfcDimensionSenses");
    var attachments = pfcCreate ("pfcSelections");

/*-----*/
    For each datum point...
/*-----*/
    for(var p=0; p <points.Count; p++)
    {

/*-----*/
        Calculate the position of the point on the drawing
/*-----*/
        var point = points.Item (p);
        var pntPos = point.Point;

        pntPos = viewTransf.TransformPoint (pntPos);

/*-----*/
        Set up the "sense" information
/*-----*/
        var sense1 = pfcCreate ("pfcPointDimensionSense").Create (
            pfcCreate ("pfcDimensionPointType").DIMPOINT_CENTER);
        senses.Set (0, sense1);
        var sense2 = pfcCreate ("pfcPointDimensionSense").Create (
            pfcCreate ("pfcDimensionPointType").DIMPOINT_CENTER);
        senses.Set (1, sense2);

/*-----*/
        Set the attachment information
/*-----*/
        var pntSel =
pfcCreate ("MpfcSelect").CreateModelItemSelection (point,
                                                    void null);
        pntSel.SelView2D = selView;
        attachments.Set (0, pntSel);
        attachments.Set (1, csysSel);

/*-----*/
        Calculate the dim position to be just to the left of the

```

```

        drawing view, midway between the point and csys
    /*-----*/
        var dimPos = pfcCreate ("pfcVector2D");
        dimPos.Set (0, outline.Item (0).Item (0) - 20.0);
        dimPos.Set (1, (csysPos.Item (1) + pntPos.Item (1))/2.0);

    /*-----*|
    Create and display the dimension
    /*-----*/
        var createInstrs =
        pfcCreate ("pfcDrawingDimCreateInstructions").Create (attachments,
                                                             senses,
                                                             dimPos,
                                                             pfcCreate ("pfcOrientationHint").ORIENTHINT_VERTICAL);
        var dim = drawing.CreateDrawingDimension (createInstrs);

        var showInstrs =
        pfcCreate ("pfcDrawingDimensionShowInstructions").Create (selView,
                                                                void null);

        dim.Show (showInstrs);

    /*-----*|
    If this is the first vertical dim, create an ordinate base
    line from it, else just convert it to ordinate
    /*-----*/
        if(p==0)
        {
            vBaseline = dim.ConvertToBaseline (csys3DPos);
        }

        else
        dim.ConvertToOrdinate (vBaseline);

    /*-----*|
    Set this dimension to be horizontal
    /*-----*/
        createInstrs.OrientationHint =
        pfcCreate ("pfcOrientationHint").ORIENTHINT_HORIZONTAL;

    /*-----*|
    Calculate the dim position to be just to the bottom of the
    drawing view, midway between the point and csys
    /*-----*/
        dimPos.Set (0, (csysPos.Item (0) + pntPos.Item (0))/2.0);
        dimPos.Set (1, outline.Item (1).Item (1) - 20.0);

        createInstrs.TextLocation = dimPos;

```

```

/*-----*\
    Create and display the dimension
/*-----*/

    dim = drawing.CreateDrawingDimension (createInstrs);
    dim.Show (showInstrs);

/*-----*\
    If this is the first horizontal dim, create an ordinate base line
    from it, else just convert it to ordinate
/*-----*/

    if(p==0)
    {
        hBaseline = dim.ConvertToBaseline (csys3DPos);
    }

    else
    dim.ConvertToOrdinate (hBaseline);

}

/*
    HISTORY

14-NOV-02   J-03-38   $$1   JCN   Adapted from J-Link examples.
07-MAR-03   K-01-03   $$2   JCN   UNIX support
*/

/*=====*\
FUNCTION: addHoleDiameterColumns
PURPOSE:  Add all hole diameters to the family table of a model.
\*=====*/
function addHoleDiameterColumns ()
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");

/*-----*\
    Use the current solid model.
/*-----*/

    var session = pfcGetProESession ();
    var solid = session.CurrentModel;
    modelTypeClass = pfcCreate ("pfcModelType");

    if (solid == void null || (solid.Type != modelTypeClass.MDL_PART &&

```



```

        solid.Type != modelTypeClass.MDL_ASSEMBLY))
    {
        throw new Error (0, "Current model is not a part or assembly.");
    }

    /*-----*\
    List all holes in the solid model
    \*-----*/
    var holeFeatures = solid.ListFeaturesByType (true,
        pfcCreate ("pfcFeatureType").FEATTYPE_HOLE);
    for (var ii = 0; ii < holeFeatures.Count; ii++)
    {
        var holeFeat = holeFeatures.Item(ii);

    /*-----*\
    List all dimensions in the feature
    \*-----*/
        dimensions =
            holeFeat.ListSubItems(pfcCreate ("pfcModelItemType").ITEM_DIMENSION);

        for (var jj = 0; jj < dimensions.Count; jj++)
        {
            var dim = dimensions.Item(jj);

    /*-----*\
    Determine if the dimension is a diameter dimension
    \*-----*/
            if (dim.DimType == pfcCreate ("pfcDimensionType").DIM_DIAMETER)
            {
    /*-----*\
    Create the family table column (from pfcFamilyMember class)
    \*-----*/
                var dimColumn = solid.CreateDimensionColumn(dim);
    /*-----*\
    Add the column to the family table. Second argument could be
    a sequence of pfcParamValues to use for each family table instance.
    \*-----*/
                solid.AddColumn(dimColumn, void null);
            }
        }
    }

    /*
    HISTORY

```

```

14-NOV-02   J-03-38   $$1   JCN       Adapted from J-Link examples.
07-MAR-03   K-01-03   $$2   JCN       UNIX support
09-APR-09   L-03-29   $$3   SRV       Remvng Enum:pfcIntfCATIA
02-Oct-09   L-05-10   $$4   tshmeleva  Removed pfcIntfPDGS
*/

/*
   This function will return a Feature object when provided with a solid
   a coordinate system name and an import feature's filename. The method
   find the coordinate system in the model, sets the Import Feature Attributes,
   and creates the Import Feature. Then the Feature is returned.
*/
function createImportFeatureFromDataFile (fileType /* string */,
                                          fileName /* string */,
                                          csysName /* string */)
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");

    var fileClass = void null;

    if (fileType == "Neutral")
        fileClass = pfcCreate ("pfcIntfNeutralFile");
    else if (fileType == "IGES")
        fileClass = pfcCreate ("pfcIntfIges");
    else if (fileType == "SET")
        fileClass = pfcCreate ("pfcIntfSet");
    else if (fileType == "STEP")
        fileClass = pfcCreate ("pfcIntfStep");
    else if (fileType == "VDA")
        fileClass = pfcCreate ("pfcIntfVDA");
    else
        throw new Error (0, "Unrecognized file type");

    /*-----*\
       Get the current part
    /*-----*/
    var session = pfcGetProESession ();
    var solid = session.CurrentModel;

    if (solid.Type != pfcCreate ("pfcModelType").MDL_PART)
        throw new Error (0, "Current model is not an assembly");

    /*-----*\
       Find the indicated coordinate system
    /*-----*/

```

```

var cSystem = solid.GetItemByName (pfcCreate ("pfcModelItemType").ITEM_COORD_SYS,
                                   csysName);
if (cSystem == void null)
    throw new Error (0, "Couldn't find named coordinate system.");

/*-----*\
   Prepare the import feature instructions classes and create the feature
\*-----*/
var dataSource = fileClass.Create(fileName);
var featAttr = pfcCreate ("pfcImportFeatAttr").Create();
featAttr.JoinSurfs = true;
featAttr.MakeSolid = true;
featAttr.Operation = pfcCreate("pfcOperationType").ADD_OPERATION;

var importFeature = solid.CreateImportFeat(dataSource, cSystem, featAttr);

return importFeature;
}

/*
   HISTORY

14-NOV-02   J-03-38   $$1   JCN   Adapted from J-Link examples.
07-MAR-03   K-01-03   $$2   JCN   UNIX support
*/

/*
   This method allows a user to evaluate the assembly for a presence of any
   interferences. Upon finding one, this method will highlight the interfering
   surfaces, compute and highlight the interference volume.
*/
function showInterferences()
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");

/*-----*\
   Get the current assembly
\*-----*/
var session = pfcGetProESession ();
var assembly = session.CurrentModel;

if (assembly.Type != pfcCreate ("pfcModelType").MDL_ASSEMBLY)
    throw new Error (0, "Current model is not an assembly");

```

```

/*-----*\
    Calculate the assembly interference
/*-----*/

var gblEval =
    pfcCreate ("MpfcInterference").CreateGlobalEvaluator(assembly);

var gblInters = gblEval.ComputeGlobalInterference(true);

if (gblInters != void null)
{
    var size = gblInters.Count;

/*-----*\
    For each interference object display the interfering surfaces
    and compute the interference volume
/*-----*/

    session.CurrentWindow.SetBrowserSize (0.0);
    session.CurrentWindow.Repaint();
    alert ("Interferences detected, highlighting each instance.");
    for (var i = 0; i < size; i++)
    {
        var gblInter = gblInters.Item (i);

        var selectPair = gblInter.SelParts;
        var sel1 = selectPair.Sel1;
        var sel2 = selectPair.Sel2;
        sel1.Highlight(pfcCreate ("pfcStdColor").COLOR_HIGHLIGHT);
        sel2.Highlight(pfcCreate ("pfcStdColor").COLOR_HIGHLIGHT);

        var vol = gblInter.Volume;
        var totalVolume = vol.ComputeVolume();
        vol.Highlight(pfcCreate ("pfcStdColor").COLOR_PREHIGHLIGHT);
        alert ("Interference " + (i + 1) + " = " + totalVolume);

        sel1.UnHighlight();
        sel2.UnHighlight();
    }
}

/*
    HISTORY

14-NOV-02    J-03-38    $$1    JCN    Adapted from J-Link examples.
07-MAR-03    K-01-02    $$3    JCN    UNIX support
*/

```

```

/*=====*\
FUNCTION: createParametersFromArguments
PURPOSE: Create/modify parameters in the model based on name-value pairs
         in the page URL
\*=====*/
function createParametersFromArguments ()
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");

    var propValue;
    var propsfile = "params.properties";
    var p;

    var args = getArgs ();

/*-----*\
    Use the current model as the parameter owner.
\*-----*/
    var session = pfcGetProESession ();
    var pOwner = session.CurrentModel;

    if (pOwner == void null)
        throw new Error (0, "No current model.");

/*-----*\
    Process each name/value pair as a Pro/E parameter.
\*-----*/
    for (var i = 0; i < args.length; i++)
    {
        var pName = args[i].Name;
        var pv = createParamValueFromString(args[i].Value);
        p = pOwner.GetParam(pName);

/*-----*\
        GetParam returns null if it can't find the param. Create it.
\*-----*/
        if (p == void null)
        {
            pOwner.CreateParam (pName, pv);
        }
        else
        {
            p.Value = pv;
        }
    }
}

```

```

    session.RunMacro ("~ Select `main_dlg_cur` `MenuBar1` `Utilities`;~ Close `main_dlg_cu
}

/*=====*\
FUNCTION: getArgs
PURPOSE: Parse arguments passed via the URL
\*=====*/
function getArgs ()
{
    var args = new Array ();

    var query = location.search.substring (1);

    var pairs = query.split ("%");
    for (var i = 0; i < pairs.length; i++)
    {
        var pos = pairs [i].indexOf ('=');
        if (pos == -1) continue;
        var argname = pairs[i].substring (0, pos);
        var value = pairs[i].substring (pos+1);
        var argPair = new Object ();
        argPair.Name = argname;
        argPair.Value = unescape (value);
        args.push (argPair);
    }

    return args;
}

/*=====*\
FUNCTION: createParamValueFromString
PURPOSE: Parses a string into a pfcParamValue object, checking for most
         restrictive possible type to use.
\*=====*/
function createParamValueFromString (s /* string */)
{
    if (s.indexOf (".") == -1)
    {
        var i = parseInt (s);
        if (!isNaN(i))
            return pfcCreate ("MpfcModelItem").CreateIntParamValue(i);
    }
    else
    {
        var d = parseFloat (s);

```

```

        if (!isNaN(d))
            return pfcCreate ("MpfcModelItem").CreateDoubleParamValue(d);
    }
    if (s.toUpperCase() == "Y" || s.toUpperCase ()== "TRUE")
        return pfcCreate ("MpfcModelItem").CreateBoolParamValue(true);

    if (s.toUpperCase() == "N" || s.toUpperCase ()== "FALSE")
        return pfcCreate ("MpfcModelItem").CreateBoolParamValue(false);

    return pfcCreate ("MpfcModelItem").CreateStringParamValue(s);
}

/*
    HISTORY

    14-NOV-02   J-03-38   $$1   JCN   Adapted from J-Link examples.
    07-MAR-03   K-01-03   $$2   JCN   UNIX support

*/

/*
    A sample that shows the use of the GetProEArguments
    method to access the Pro/ENGINEER command line arguments.
    The first argument is always the full path to the Pro/E executable.
    For this application the next two arguments can be either ("runtime" or
    "+development") or ("-Unix" or "-NT"). Based on these values 2 boolean
    variables are set and passed on to another function which makes use of
    this information.
*/

var runtime = true;
var unix = false;

function getArguments ()
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");

    var argseq = pfcCreate ("MpfcCOMGlobal").GetProEArguments();

    /*-----*\
        Making sure that there are three arguments.
    \*-----*/

```

```

    if (argseq.Count == 3)
    {
        /*-----*\
        First argument is Pro/E executable - skip it
        \*-----*/
        /*-----*\
        Set flags based on Pro/E input arguments
        \*-----*/
        var val=argseq.Item(1);

        setFlags (val);

        /*-----*\
        Set third flag based on Pro/E input argument
        \*-----*/
        val=argseq.Item(2);

        setFlags (val);
    }
    /*-----*\
    Pass the boolean values to another function
    \*-----*/
    //runApplication(runtime,unix);
}

function setFlags (val /* string */)
{
    if (val == "+runtime")
    {
        runtime=true;
    }
    else if (val == "+development")
    {
        runtime=false;
    }
    else if (val == "-Unix")
    {
        unix=true;
    }
    else if (val == "-NT")
    {
        unix=false;
    }
}

```



```

/*
    HISTORY

    23-JUL-07   L-01-35   $$1   SNV   Created and submitted

*/

/*=====*\
FUNCTION: createParamDimRelation_script_wrapper
PURPOSE:  Wrapper function for createParamDimRelation.
\*=====*/
function createParamDimRelation_script_wrapper()
{
    var i;
    var selections;
    var options ;

    var session = pfcGetProESession ();
    /*=====*\
    Get the current part model
    \*=====*/
    var solid = session.CurrentModel;
    modelTypeClass = pfcCreate ("pfcModelType");

    if (solid == void null || (solid.Type != modelTypeClass.MDL_PART))
    {
        throw new Error  (0, "Current model is not a part.");
    }

    /*=====*\
        Get selected components
    \*=====*/
    var browserSize = session.CurrentWindow.GetBrowserSize();
    session.CurrentWindow.SetBrowserSize (0.0);

    options = pfcCreate("pfcSelectionOptions").Create("feature");
    selections = session.Select (options, void null);

    session.CurrentWindow.SetBrowserSize (browserSize);
    if (selections == void null || selections.Count == 0)
    {
        throw new Error  (0, "Nothing selected");
    }
}

```

```

        var features = pfcCreate("pfcFeatures");
        for (i =0; i < selections.Count ; i++)
        {
            features.Append(selections.Item(i).SelItem);
        }

        createParamDimRelation(features);
    }

    /*=====*\
    FUNCTION: createParamDimRelation
    PURPOSE:  This function creates parameters for all dimensions in input
              features of a part model and adds relation between them.
    \*=====*/
    function createParamDimRelation (features)
    {
        if (!pfcIsWindows())
            netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");

        var i,j;
        var relations;
        var items;
        var dimName , paramName;
        var dimValue;
        var paramAdded;
        var param ;
        var paramValue;

        for (i =0; i < features.Count ; i++)
        {
            /*=====*\
            Get the selected feature
            \*=====*/
            var feature = features.Item(i);
            if (feature == void null)
            {
                continue;
            }

            /*=====*\
            Get the dimensions in the current feature
            \*=====*/
            items = feature.ListSubItems(pfcCreate ("pfcModelItemType").ITEM_DIMENSION);

```

```

if ((items == void null) || (items.Count == 0 ))
{
    continue;
}

relations = pfcCreate("stringseq");

/*=====*\
    Loop through all the dimensions and create relations
\*=====*/
for (j = 0; j < items.Count; j++)
{
    var item = items.Item(j);
    dimName = item.GetName();
    paramName = paramName = "PARAM_" + dimName;

    dimValue = item.DimValue;

    param = feature.GetParam(paramName);
    paramAdded = false;

    if (param == void null)
    {
        paramValue = pfcCreate ("MpfcModelItem").CreateDoubleParamValue(dimValue);
        feature.CreateParam (paramName, paramValue);
        paramAdded = true;
    }
    else
    {
        if (param.Value.discr == pfcCreate ("pfcParamValueType").PARAM_DOUBLE)
        {
            paramValue = pfcCreate ("MpfcModelItem").CreateDoubleParamValue(dimValue);
            param.Value = paramValue;
            paramAdded = true;
        }
    }

    if (paramAdded == true)
    {
        relations.Append(dimName + " = " + paramName);
    }
    param = void null;
}
feature.Relations = relations;

```

```

    }
}

```

```

/*
    HISTORY

    14-NOV-02   J-03-38   $$1   JCN       Adapted from J-Link examples.
    07-MAR-03   K-01-03   $$2   JCN       UNIX support
*/

/*
    This example code demonstrates how to invoke an interactive selection.
*/
function selectItems (options /* string[] */, max /* integer */)
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");

    /*-----*\
        Get the session. If no model in present abort the operation.
    \*-----*/
    var session = pfcGetProESession ();
    var model = session.CurrentModel;

    if (model == void null)
        throw new Error (0, "No current model.");

    /*-----*\
        Collect the options array into a comma delimited list
    \*-----*/
    var optString = "";
    for (var i = 0; i < options.length; i++)
    {
        optString += options [i];
        if (i != options.length -1)
            optString += ",";
    }

    /*-----*\
        Prompt for selection.
    \*-----*/
    selOptions = pfcCreate ("pfcSelectionOptions").Create (optString);

    if (max != "UNLIMITED")

```

```

    {
        selOptions.MaxNumSels = parseInt (max);
    }

    session.CurrentWindow.SetBrowserSize (0.0);

    var selections = void null;
    try {
        selections = session.Select (selOptions, void null);
    }
    catch (err) {
        /*-----*\
        Handle the situation where the user didn't make selections, but picked
        elsewhere instead.
        \*-----*/
        if (pfcGetExceptionType (err) == "pfcXToolkitUserAbort" ||
            pfcGetExceptionType (err) == "pfcXToolkitPickAbove")
            return (void null);
        else
            throw err;
    }
    if (selections.Count == 0)
        return (void null);

    /*-----*\
    Write selection info to the browser window
    \*-----*/
    var newWin = window.open ('', "_IS", "scrollbars");
    if (pfcIsWindows())
    {
        newWin.resizeTo (300, screen.height/2.0);
        newWin.moveTo (screen.width-300, 0);
    }
    newWin.document.writeln ("<head></head><body>");

    for (var i = 0; i < selections.Count; i ++)
    {
        var sel = selections.Item (i);
        newWin.document.writeln ("

## <td>Sel model: </td><td>"+ selModelName+"</td></tr>"); }


```

```

        var selItemInfo = "N/A";
        if (sel.SelItem != void null)
            selItemInfo = "Type: " + sel.SelItem.Type.ToString() +
                " id: " + sel.SelItem.Id;

        newWin.document.writeln ("<tr><td>Sel item: </td><td>" +
            selItemInfo + "</td></tr>");
        newWin.document.writeln ("</table>");
    }

    newWin.document.writeln ("<html><head></head><body>");

    return (selections);
}

/*
    This method highlights all the features in all levels of an assembly that have a give
*/
function createAndHighlightSelections (featureName /* string */)
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");

    /*-----*\
        Get the session. If no model in present abort the operation.
    /*-----*/
    var session = pfcGetProESession ();
    var assem = session.CurrentModel;

    if (assem == void null ||
        assem.Type != pfcCreate ("pfcModelType").MDL_ASSEMBLY)
        throw new Error (0, "Current model is not an assembly.");

    /*-----*\
        Start a recursive traversal of the assembly structure.
    /*-----*/
    intPath = pfcCreate ("intseq");

    highlightFeaturesRecursively (assem, intPath, featureName);
}

function highlightFeaturesRecursively (assem /* pfcAssembly */,
    intPath /* intseq */,
    featureName /* string */)

```

```

{
/*-----*\
    Obtain the model at the current assembly level.
/*-----*/
    var subcomponent;
    var cmpPath = void null;
    if (intPath.Count == 0)
        subcomponent = assem;
    else
    {
        cmpPath =
        pfcCreate ("MpfcAssembly").CreateComponentPath( assem, intPath );
        subcomponent = cmpPath.Leaf;
    }

/*-----*\
    Search for the desired feature.
/*-----*/
    var theFeat = subcomponent.GetFeatureByName (featureName);
    if (theFeat != void null)
    {
        var cmpSelection =
        pfcCreate ("MpfcSelect").CreateModelItemSelection ( theFeat, cmpPath );

        cmpSelection.Highlight(pfcCreate ("pfcStdColor").COLOR_HIGHLIGHT);
    }

/*-----*\
    Search for subcomponents, and traverse each of them.
/*-----*/
    var components = subcomponent.ListFeaturesByType(true,
        pfcCreate ("pfcFeatureType").FEATTYPE_COMPONENT);
    for (var i = 0; i < components.Count; i++)
    {
        var compFeat = components.Item (i);
        if (compFeat.Status == pfcCreate ("pfcFeatureStatus").FEAT_ACTIVE)
        {
            intPath.Append(components.Item (i).Id);
            highlightFeaturesRecursively (assem, intPath, featureName);
        }
    }

/*-----*\
    Clean up the assembly ids at this level before returning.
/*-----*/
    if (intPath.Count > 0)

```

```

    {
        intPath.Remove (intPath.Count - 1, intPath.Count);
    }
}

/*
    HISTORY

    02-Aug-10    L-05-28    $$1    pdeshmuk    Created.

*/

function deleteItemsInSimpRep ()
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");

    /*-----*\
        Get the current assembly
    \*-----*/
    var session = pfcGetProESession ();
    var assembly = session.CurrentModel;

    if (assembly.Type != pfcCreate ("pfcModelType").MDL_ASSEMBLY)
        throw new Error (0, "Current model is not an assembly");

    /*-----*\
        Get the current active simprep.
    \*-----*/
    var simp_rep = assembly.GetActiveSimpRep();

    /*-----*\
        Get the current number of items
    \*-----*/
    var simp_rep_instructions = simp_rep.GetInstructions();

    var number_items = simp_rep_instructions.Items.Count;
    document.getElementById("numItems").value = number_items;

    /*-----*\
        Deleting items
    \*-----*/
    var simp_rep_instructions_item = simp_rep_instructions.Items.Item(number_items-1);
    simp_rep_instructions_item.Action = null;
    simp_rep.SetInstructions(simp_rep_instructions);
}

```



```

        number_items = simp_rep.GetInstructions().Items.Count;
        document.getElementById("numItems").value = number_items;

    return;
}

function addItemInSimpRep ()
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");

    /*-----*|
       Get the current assembly
    /*-----*/
    var session = pfcGetProESession ();
    var assembly = session.CurrentModel;

    if (assembly.Type != pfcCreate ("pfcModelType").MDL_ASSEMBLY)
        throw new Error (0, "Current model is not an assembly");

    /*-----*|
       Get the current active simprep.
    /*-----*/
    var simp_rep = assembly.GetActiveSimpRep();

    /*-----*|
       Get the current number of items
    /*-----*/
    var simp_rep_instructions = simp_rep.GetInstructions();

    var number_items = simp_rep_instructions.Items.Count;
    document.getElementById("numItems").value = number_items;

    /*-----*|
       Add an item
    /*-----*/
    var item_path = pfcCreate ("intseq");

    /*-----*|

```

```

    Prompt for selection.
    /*-----*/
    selOptions = pfcCreate ("pfcSelectionOptions").Create ("feature");

    selOptions.MaxNumSels = parseInt (1);

    var selections = void null;
    try {
        selections = session.Select (selOptions, void null);
    }
    catch (err) {
        /*-----*/
        Handle the situation where the user didn't make selections, but picked
        elsewhere instead.
        /*-----*/
        if (pfcGetExceptionType (err) == "pfcXToolkitUserAbort" ||
            pfcGetExceptionType (err) == "pfcXToolkitPickAbove")
            return (void null);
        else
            throw err;
    }
    if (selections.Count == 0)
        return (void null);

    var selection = selections.Item(0);
    var componentpath = selection.Path;
    var intseqIds = componentpath.ComponentIds;
    item_path.Append(intseqIds.Item(0));
    simp_rep_comp_item_path = pfcCreate("pfcSimpRepCompItemPath").Create(item_path);

    simp_rep_item = pfcCreate("pfcSimpRepItem").Create(simp_rep_comp_item_path);

    simp_rep_action = pfcCreate("pfcSimpRepExclude").Create();

    simp_rep_item.Action = simp_rep_action;

    simp_rep_instructions.Items.Append(simp_rep_item);

    simp_rep.SetInstructions(simp_rep_instructions);

    simp_rep_instructions = simp_rep.GetInstructions();

    number_items = simp_rep_instructions.Items.Count;
    document.getElementById("numItems").value = number_items;

    return;

```

```

}

/*
    HISTORY

    14-NOV-02   J-03-38   $$1   JCN   Adapted from J-Link examples.
    07-MAR-03   K-01-03   $$2   JCN   UNIX support
*/

/* This method retrieves a MassProperty object from the provided solid
 * model. Then solid's mass, volume, and center of gravity point are printed
 */
function printMassProperties ()
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");

    /*-----*\
        Get the session. If no model in present abort the operation.
    /*-----*/
    var session = pfcGetProESession ();
    var solid = session.CurrentModel;

    if (solid == void null || (solid.Type != pfcCreate ("pfcModelType").MDL_PART &&
        solid.Type != pfcCreate ("pfcModelType").MDL_ASSEMBLY))
        throw new Error (0, "Current model is not a part or assembly.");

    /*-----*\
        Calculate the mass properties. Pass null to use the model
        coordinate system.
    /*-----*/
    properties = solid.GetMassProperty(void null);

    /*-----*\
        Display selected results.
    /*-----*/
    var newWin = window.open ('', "_MP", "scrollbars");
    if (pfcIsWindows())
    {
        newWin.resizeTo (300, screen.height/2.0);
        newWin.moveTo (screen.width-300, 0);
    }
    newWin.document.writeln ("<html><head></head><body>");

    newWin.document.writeln ("<p>The solid mass is: " + properties.Mass);

```

```

newWin.document.writeln ("<p>The solid volume is: " + properties.Volume);

COG = properties.GravityCenter;
newWin.document.writeln ("<hr><p>The Center Of Gravity is at ");
newWin.document.writeln ("<table>");
newWin.document.writeln ("<tr><td>X</td><td>" + COG.Item(0) + "</td></tr>");
newWin.document.writeln ("<tr><td>Y</td><td>" + COG.Item(1) + "</td></tr>");
newWin.document.writeln ("<tr><td>Z</td><td>" + COG.Item(2) + "</td></tr>");
newWin.document.writeln ("</table>");
newWin.document.writeln ("<html><head></head><body>");
}

/*
    HISTORY

    14-NOV-02    J-03-38    $$1    JCN    Adapted from J-Link examples.
    07-MAR-03    K-01-03    $$2    JCN    UNIX support
    */

/*=====*\
FUNCTION: createNodeUDFInPart
PURPOSE: Places copies of a node UDF at a particular coordinate system
         location in a part. The node UDF is a spherical cut centered at the
         coordinate system whose diameter is driven by the 'diam' argument to the
         method.
\*=====*/
function createNodeUDFInPart (csysName /* string */,
                             diam /* number */)
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");

    /*-----*\
    Use the current model to place the UDF.
    \*-----*/
    var session = pfcGetProESession ();
    var solid = session.CurrentModel;

    if (solid == void null || solid.Type != pfcCreate ("pfcModelType").MDL_PART)
        throw new Error (0, "Current model is not a part. Aborting...");

    /*-----*\
    The instructions for the UDF creation.
    \*-----*/
    var instrs =
        pfcCreate ("pfcUDFCustomCreateInstructions").Create ("node");

```

```

/*-----*\
Make non-variant dimensions blank so they cannot be changed.
\*-----*/
instrs.DimDisplayType =
    pfcCreate ("pfcUDFDimensionDisplayType").UDFDISPLAY_BLANK;

/*-----*\
Initialize the UDF reference and assign it to the instructions.
The string argument is the reference prompt for the particular
reference.
\*-----*/
csys =
    solid.GetItemByName (pfcCreate ("pfcModelItemType").ITEM_COORD_SYS,
        csysName);
if (csys == void null)
    throw new Error (0, "Requested coordinate system "+csysName+" not found.");
csysSel =
    pfcCreate ("MpfcSelect").CreateModelItemSelection (csys, void null);

var csysRef =
    pfcCreate ("pfcUDFReference").Create ("REF_CSYS", csysSel);

var refs = pfcCreate ("pfcUDFReferences");
refs.Append (csysRef);

instrs.References = refs;

/*-----*\
Initialize the variant dimension and assign it to the instructions.
The string argument is the dimension symbol for the variant
dimension.
\*-----*/
var varDiam =
    pfcCreate ("pfcUDFVariantDimension").Create ("d11", diam);

var vals = pfcCreate ("pfcUDFVariantValues");
vals.Append (varDiam);

instrs.VariantValues = vals;

/*-----*\
Create the new UDF placement.
\*-----*/
var group = solid.CreateUDFGroup (instrs);
return (group);

```

```

}

/*
    HISTORY

    14-NOV-02    J-03-38    $$1    JCN    Submitted.
    07-MAR-03    K-01-03    $$2    JCN    UNIX support
    01-MAY-07    L-01-31    $$3    JCN    New exception messaging.
*/

function isProEEmbeddedBrowser ()
{
    if (top.external && top.external.ptc)
        return true;
    else
        return false;
}

function pfcIsWindows ()
{
    if (navigator.appName.indexOf ("Microsoft") != -1)
        return true;
    else
        return false;
}

function pfcCreate (className)
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");

    if (pfcIsWindows())
        return new ActiveXObject ("pfc."+className);
    else
    {
        ret = Components.classes ["@ptc.com/pfc/" + className + ";1"].
            createInstance();

        return ret;
    }
}

function pfcGetProESession ()
{
    if (!isProEEmbeddedBrowser ())

```

```

    {
        throw new Error ("Not in embedded browser.  Aborting...");
    }

    // Security code
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");

    var glob = pfcCreate ("MpfcCOMGlobal");
    return glob.GetProESession();
}

function pfcGetScript ()
{
    if (!isProEEmbeddedBrowser ())
    {
        throw new Error ("Not in embedded browser.  Aborting...");
    }

    // Security code
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");

    var glob = pfcCreate ("MpfcCOMGlobal");
    return glob.GetScript();
}

function pfcGetExceptionDescription (err)
{
    if (pfcIsWindows())
        errString = err.description;
    else
        errString = err.message;

    return errString;
}

function pfcGetExceptionType (err)
{
    errString = pfcGetExceptionDescription (err);

    // This should remove the XPCOM prefix ("XPCR_C")
    if (errString.search ("XPCR_C") < 0)
    {
        errString = errString.replace ("Exceptions::", "");
    }
}

```

```

        semicolonIndex = errString.search (",");
        if (semicolonIndex > 0)
            errString = errString.substring (0, semicolonIndex);
        return (errString);
    }
    else
        return (errString.replace("XPCR_C", ""));
}

/*
    HISTORY

    14-NOV-02    J-03-38    $$1    JCN    Adapted from J-Link examples.
    07-MAR-03    K-01-03    $$2    JCN    UNIX support
    */

try
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");

    wpwl = pfcGetScript ();
    document.pwl = wpwl;
    wpwlc = wpwl.GetPWLConstants ();
    document.pwlc = wpwlc;
    wpwlf = wpwl.GetPWLFeatureConstants ();
    document.pwlf = wpwlf;
}
catch (err)
{
    alert ("Exception caught: "+pfcGetExceptionType (err));
}

function WlProEStart()
{
    if (document.pwl == void null)
    {
        alert("Connect failed.");
        return ;
    }
}

function WlProEConnect()

```



```

// Connect to a running Pro/ENGINEER session.
{
    WlProEStart();
}

function WlModelOpen()
// Open a Pro/ENGINEER model.
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");
    if (document.main.ModelName.value == "")
        return ;
    var ret = document.pwl.pwlMdlOpen(document.main.ModelName.value,
        document.main.ModelPath.value, true);
    if (!ret.Status)
    {
        if (ret.ErrorCode == -4 && document.main.ModelPath.value == "")
            return ;
        else
        {
            alert("pwlMdlOpen failed (" + ret.ErrorCode + ")");
            return ;
        }
    }
}

function WlModelRegenerate()
// Regenerate the Pro/ENGINEER model.
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");
    var ret = document.pwl.pwlMdlRegenerate(document.main.ModelNameExt.value);
    if (!ret.Status)
    {
        alert("pwlMdlRegenerate failed (" + ret.ErrorCode + ")");
        return ;
    }
}

function WlModelSave()
// Save a Pro/ENGINEER model.
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");
    var ret = document.pwl.pwlMdlSaveAs(document.main.ModelNameExt.value, void null, void null);
    if (!ret.Status)

```

```

    {
        alert("pwlMdlSaveAs failed (" + ret.ErrorCode + ")");
        return ;
    }
}

function WlModelSaveAs()
// Save a Pro/ENGINEER model under a new name.
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");
    var NewPath = document.main.NewPath.value;
    var NewName = document.main.NewName.value;
    if (NewPath == "")
    {
        NewPath = void null;
    }
    if (NewName == "")
    {
        NewName = void null;
    }
    var ret = document.pwl.pwlMdlSaveAs(document.main.ModelNameExt.value,
        NewPath, NewName);
    if (!ret.Status)
    {
        alert("pwlMdlSaveAs failed (" + ret.ErrorCode + ")");
        return ;
    }
}

function WlWindowRepaint()
// Repaint the active window.
{
    if (!pfcIsWindows())
        netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");
    var get_ret = document.pwl.pwlWindowActiveGet();
    if (!get_ret.Status)
    {
        alert("pwlWindowActiveGet failed (" + get_ret.ErrorCode + ")");
        return ;
    }
    /* You can also repaint the active window using -1 as the window
       identifier. */
    var ret = document.pwl.pwlWindowRepaint(parseInt(get_ret.WindowID));
    if (!ret.Status)
    {

```

```

        alert("pwlWindowRepaint failed (" + ret.ErrorCode + ")");
        return ;
    }
}

// Define the form with all the buttons to perform the above actions.
document.writeln("<form name='main'>");

document.writeln("<hr>");
document.writeln("<h4>Main Controls</h4>");
document.writeln("<p>");
document.writeln("<center>");
document.writeln("<input type='button' value='Start Pro/E' onclick='WlProEStart()'>");
document.writeln("<input type='button' value='Connect to Pro/E' onclick='WlProEConnect()'");
document.writeln("<p>");
document.writeln("Path: <input type='text' name='ModelPath' onchange='WlModelOpen()'>");
document.writeln("<spacer size=20>");
document.writeln("Model: <input type='text' name='ModelName' onchange='WlModelOpen()'>");
document.writeln("<spacer size=20>");
document.writeln("<input type='button' value='Open Model' onclick='WlModelOpen()'>");
document.writeln("<p>");
document.writeln("<table>");
document.writeln("<tr>");
document.writeln("<td><center>Model:</center></td>");
document.writeln("<td><center>New Path:</center></td>");
document.writeln("<td><center>New Name:</center></td></tr>");
document.writeln("<tr>");
document.writeln("<td><input type='text' name='ModelNameExt'></td>");
document.writeln("<td><input type='text' name='NewPath'></td>");
document.writeln("<td><input type='text' name='NewName'></td></tr>");
document.writeln("</table>");
document.writeln("<input type='button' value='Regenerate Model' onclick='WlModelRegenera");
document.writeln("<spacer size=10>");
document.writeln("<input type='button' value='Save Model' onclick='WlModelSave()'>");
document.writeln("<spacer size=10>");
document.writeln("<input type='button' value='Save Model As' onclick='WlModelSaveAs()'>");
document.writeln("<p>");
document.writeln("<input type='button' value='REPAINT SCREEN' onclick='WlWindowRepaint('");
document.writeln("</center>");
document.writeln("<hr>");

document.writeln("</form>");

```

Collaborative Engine for Distributed Mechanical Design

分散式機械設計協同引擎

Qianfu Ni and Wen Feng Lu

Abstract

摘要

Effective collaboration is essential for engineers at geographically dispersed locations to accomplish good design with less iteration.

分散在世界各地的工程師若希望以較少的重複流程來完成好的設計, 需要仰賴有效率的協同.

Over the last several years, more and more efforts have been put into such research as many industries have distributed their product development to locations with knowledge force.

過去幾年, 有越來越多的業者將心力投注在利用各地的知識人力來進行分散式產品開發.

This paper presents a collaborative engine to facilitate collaborations among distributed mechanical designs.

本論文提出一種協同引擎 (設計), 可用來協助分散式機械設計有效進行協同.

Using component-based software technology, collaboration functionality is developed into a set of groupware that makes the collaborative engine applicable to develop new collaborative applications or integrate legacy applications into collaborative environments.

(本研究) 運用元件導向的軟體技術, 將協同功能置入組群組軟體中, 讓此一協同系統可以用來開發新的協同應用或將傳統應用程式與協同環境加以整合.

An XML-based information representation is developed to streamline the information transmission within the distributed environment.

(本研究) 採用以 XML 為基礎的資訊表示法, 以便讓分散式環境中的資訊傳遞得以流暢進行.

A case study is carried out to show how this engine facilitates designers to collaboratively create a 3D solid model of a same part in real time.

(本研究) 利用案例研究來說明此一 (協同) 引擎如何有效協助設計者以即時協同模式建立 3D 實體模型.

Keywords: collaboration, distributed design, collaborative engine

關鍵字: 協同, 分散式設計, 協同引擎

人偶零件: http://2014cda-mdenfu.rhcloud.com/cmsimply/download/?filepath=/var/lib/openshift/535631-root/data//downloads/lego_man.7z

```
@language python
```

```
import cherrypy
```

```
# 這是 MAN 類別的定義
```

```
'''
```

```
# 在 application 中導入子模組
```

```
import programs.cdag30.man as cdag30_man
```

```
# 加入 cdag30 模組下的 man.py 且以子模組 man 對應其 MAN() 類別
```

```
root.cdag30.man = cdag30_man.MAN()
```

```
# 完成設定後, 可以利用
```

```
/cdag30/man/assembly
```

```
# 呼叫 man.py 中 MAN 類別的 assembly 方法
```

```
'''
```

```
class MAN(object):
```

```
    # 各組利用 index 引導隨後的程式執行
```

```
    @cherrypy.expose
```

```
    def index(self, *args, **kwargs):
```

```
        outstring = '''
```

```
這是 2014CDA 協同專案下的 cdag30 模組下的 MAN 類別.<br /><br />
```

```
<!-- 這裡採用相對連結, 而非網址的絕對連結 (這一段為 html 註解) -->
```

```
<a href="assembly">執行 MAN 類別中的 assembly 方法</a><br /><br />
```

```
請確定下列零件於 V:/home/lego/man 目錄中, 且開啟空白 Creo 組立檔案.<br />
```

```
<a href="/static/lego_man.7z">lego_man.7z</a>(滑鼠右鍵存成 .7z 檔案)<br />
```

```
'''
```

```
        return outstring
```

```
    @cherrypy.expose
```

```
    def assembly(self, *args, **kwargs):
```

```
        outstring = '''
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<meta http-equiv="content-type" content="text/html; charset=utf-8">
```

```
<script type="text/javascript" src="/static/weblink/examples/jsript/pfcUtils.js"></script>
```

```
</head>
```

```
<body>
```

```
</script><script language="JavaScript">
```

```
/* 設計一個零件組立函示 */
```

```
// featID 為組立件第一個組立零件的編號
```

```

// inc 則為 part1 的組立順序編號，第一個入組立檔編號為 featID+0
// part2 為外加的零件名稱
function axis_plane_assembly(session, assembly, transf, featID, inc, part2, axis1, plane1)
var descr = pfcCreate("pfcModelDescriptor").CreateFromFileName ("v:/home/lego/man/"+part2);
var componentModel = session.GetModelFromDescr(descr);
var componentModel = session.RetrieveModel(descr);
if (componentModel != void null)
{
    var asmcomp = assembly.AssembleComponent (componentModel, transf);
}
var ids = pfcCreate("intseq");
ids.Append(featID+inc);
var subPath = pfcCreate("MpfcAssembly").CreateComponentPath(assembly, ids);
subassembly = subPath.Leaf;
var asmDatums = new Array(axis1, plane1);
var compDatums = new Array(axis2, plane2);
var relation = new Array (pfcCreate("pfcComponentConstraintType").ASM_CONSTRAINT_ALIGN, pfcCreate("pfcComponentConstraintType").ASM_CONSTRAINT_ALIGN);
var relationItem = new Array(pfcCreate("pfcModelItemType").ITEM_AXIS, pfcCreate("pfcModelItemType").ITEM_AXIS);
var constrs = pfcCreate("pfcComponentConstraints");
for (var i = 0; i < 2; i++)
{
    var asmItem = subassembly.GetItemByName (relationItem[i], asmDatums [i]);
    if (asmItem == void null)
    {
        interactFlag = true;
        continue;
    }
    var compItem = componentModel.GetItemByName (relationItem[i], compDatums [i]);
    if (compItem == void null)
    {
        interactFlag = true;
        continue;
    }
    var MpfcSelect = pfcCreate ("MpfcSelect");
    var asmSel = MpfcSelect.CreateModelItemSelection (asmItem, subPath);
    var compSel = MpfcSelect.CreateModelItemSelection (compItem, void null);
    var constr = pfcCreate("pfcComponentConstraint").Create (relation[i]);
    constr.AssemblyReference = asmSel;
    constr.ComponentReference = compSel;
    constr.Attributes = pfcCreate("pfcConstraintAttributes").Create (true, false);
    constrs.Append(constr);
}
asmcomp.SetConstraints(constrs, void null);
}
// 以上為 axis_plane_assembly() 函式
//

```

```

function three_plane_assembly(session, assembly, transf, featID, inc, part2, plane1, plane2, plane3, plane4, plane5, plane6)
var descr = pfcCreate("pfcModelDescriptor").CreateFromFileName ("v:/home/lego/man/"+part2+descr);
var componentModel = session.GetModelFromDescr(descr);
var componentModel = session.RetrieveModel(descr);
if (componentModel != void null)
{
    var asmcomp = assembly.AssembleComponent (componentModel, transf);
}
var ids = pfcCreate("intseq");
ids.Append(featID+inc);
var subPath = pfcCreate("MpfcAssembly").CreateComponentPath(assembly, ids);
subassembly = subPath.Leaf;
var constrs = pfcCreate("pfcComponentConstraints");
var asmDatums = new Array(plane1, plane2, plane3);
var compDatums = new Array(plane4, plane5, plane6);
var MpfcSelect = pfcCreate("MpfcSelect");
for (var i = 0; i < 3; i++)
{
    var asmItem = subassembly.GetItemByName(pfcCreate("pfcModelItemType").ITEM_SURFACE, i);

    if (asmItem == void null)
    {
        interactFlag = true;
        continue;
    }
    var compItem = componentModel.GetItemByName(pfcCreate("pfcModelItemType").ITEM_SURFACE, i);
    if (compItem == void null)
    {
        interactFlag = true;
        continue;
    }
    var asmSel = MpfcSelect.CreateModelItemSelection(asmItem, subPath);
    var compSel = MpfcSelect.CreateModelItemSelection(compItem, void null);
    var constr = pfcCreate("pfcComponentConstraint").Create(pfcCreate("pfcComponentConstraintAttributes").Create (false, false));
    constr.AssemblyReference = asmSel;
    constr.ComponentReference = compSel;
    constr.Attributes = pfcCreate("pfcConstraintAttributes").Create (false, false);
    constrs.Append(constr);
}
asmcomp.SetConstraints(constrs, void null);
}
// 以上為 three_plane_assembly() 函式
//
// 假如 Creo 所在的作業系統不是 Windows 環境
if (!pfcIsWindows())
// 則啟動對應的 UniversalXPConnect 執行權限 (等同 Windows 下的 ActiveX)

```

```

netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");
// pfcGetProESession() 是位於 pfcUtils.js 中的函式，確定此 JavaScript 是在嵌入式瀏覽器中執行
var session = pfcGetProESession();
// 設定 config option，不要使用元件組立流程中內建的假設約束條件
session.SetConfigOption("comp_placement_assumptions","no");
// 建立擺放零件的位置矩陣，Pro/Web.Link 中的變數無法直接建立，必須透過 pfcCreate() 建立
var identityMatrix = pfcCreate("pfcMatrix3D");
// 建立 identity 位置矩陣
for (var x = 0; x < 4; x++)
for (var y = 0; y < 4; y++)
{
    if (x == y)
        identityMatrix.Set(x, y, 1.0);
    else
        identityMatrix.Set(x, y, 0.0);
}
// 利用 identityMatrix 建立 transf 座標轉換矩陣
var transf = pfcCreate("pfcTransform3D").Create(identityMatrix);
// 取得目前的工作目錄
var currentDir = session.getCurrentDirectory();
// 以目前已開檔的空白組立檔案，作為 model
var model = session.CurrentModel;
// 查驗有無 model，或 model 類別是否為組立件，若不符合條件則丟出錯誤訊息
if (model == void null || model.Type != pfcCreate("pfcModelType").MDL_ASSEMBLY)
throw new Error (0, "Current model is not an assembly.");
// 將此模型設為組立物件
var assembly = model;

/**----- LEGO_BODY-----*/
// 設定零件的 descriptor 物件變數
var descr = pfcCreate("pfcModelDescriptor").CreateFromFileName("v:/home/lego/man/LEGO_BODY");
// 若零件在 session 則直接取用
var componentModel = session.GetModelFromDescr(descr);
// 若零件不在 session 則從工作目錄中載入 session
var componentModel = session.RetrieveModel(descr);
// 若零件已經在 session 中則放入組立檔中
if (componentModel != void null)
{
    // 注意這個 asmcomp 即為設定約束條件的本體
    // asmcomp 為特徵物件，直接將零件，以 transf 座標轉換矩陣方位放入組立檔案中
    var asmcomp = assembly.AssembleComponent(componentModel, transf);
}

// 建立約束條件變數
var constrs = pfcCreate("pfcComponentConstraints");
// 設定組立檔中的三個定位面，注意內定名稱與 Pro/E WF 中的 ASM_D_FRONT 不同，而是 ASM_FRONT。

```



```

// 組立檔案中的 Datum 名稱也可以利用 View->plane tag display 查詢名稱
// 建立組立參考面所組成的陣列
var asmDatums = new Array("ASM_FRONT", "ASM_TOP", "ASM_RIGHT");
// 設定零件檔中的三個定位面，名稱與 Pro/E WF 中相同
var compDatums = new Array("FRONT", "TOP", "RIGHT");
// 建立 ids 變數，intseq 為 sequence of integers 為資料類別，使用者可以經由整數索引擷取此資料
// intseq 等同 Python 的數列資料？
var ids = pfcCreate("intseq");
// 利用 assembly 物件模型，建立路徑變數
var path = pfcCreate("MpfcAssembly").CreateComponentPath(assembly, ids);
// 採用互動式設定相關的變數，MpfcSelect 為 Module level class 中的一種
var MpfcSelect = pfcCreate("MpfcSelect");
// 利用迴圈分別約束組立與零件檔中的三個定位平面
for (var i = 0; i < 3; i++)
{
    // 設定組立參考面，也就是 "ASM_FRONT", "ASM_TOP", "ASM_RIGHT" 等三個 datum planes
    var asmItem = assembly.GetItemByName (pfcCreate("pfcModelItemType").ITEM_SURFACE, asmDatums[i]);
    // 若無對應的組立參考面，則啟用互動式平面選擇表單 flag
    if (asmItem == void null)
    {
        interactFlag = true;
        continue;
    }
    // 設定零件參考面，也就是 "FRONT", "TOP", "RIGHT" 等三個 datum planes
    var compItem = componentModel.GetItemByName (pfcCreate ("pfcModelItemType").ITEM_SURFACE, compDatums[i]);
    // 若無對應的零件參考面，則啟用互動式平面選擇表單 flag
    if (compItem == void null)
    {
        interactFlag = true;
        continue;
    }
    // 因為 asmItem 為組立件中的定位特徵，必須透過 path 才能取得
    var asmSel = MpfcSelect.CreateModelItemSelection(asmItem, path);
    // 而 compItem 則為零件，沒有 path 路徑，因此第二變數為 null
    var compSel = MpfcSelect.CreateModelItemSelection(compItem, void null);
    // 利用 ASM_CONSTRAINT_ALIGN 對齊組立約束建立約束變數
    var constr = pfcCreate("pfcComponentConstraint").Create (pfcCreate ("pfcComponentConstraintType").ALIGN);
    // 設定約束條件的組立參考與元件參考選擇
    constr.AssemblyReference = asmSel;
    constr.ComponentReference = compSel;
    // 第一個變數為強制變數，第二個為忽略變數
    // 強制變數為 false，表示不強制約束，只有透過點與線對齊時需設為 true
    // 忽略變數為 false，約束條件在更新模型時是否忽略，設為 false 表示不忽略
    // 通常在組立 closed chain 機構時， 忽略變數必須設為 true，才能完成約束
    // 因為三個面絕對約束，因此輸入變數為 false, false
    constr.Attributes = pfcCreate("pfcConstraintAttributes").Create (false, false);
}

```

```

// 將互動選擇相關資料，附加在程式約束變數之後
constrs.Append(constr);
}

// 設定組立約束條件
asmcomp.SetConstraints (constrs, void null);
/**----- LEGO_ARM_RT 右手上臂-----**/
var descr = pfcCreate ("pfcModelDescriptor").CreateFromFileName ("v:/home/lego/man/LEGO_
var componentModel = session.GetModelFromDescr(descr);
var componentModel = session.RetrieveModel(descr);
if (componentModel != void null)
{
    // 注意這個 asmcomp 即為設定約束條件的本體
    // asmcomp 為特徵物件，直接將零件，以 transf 座標轉換放入組立檔案中
var asmcomp = assembly.AssembleComponent (componentModel, transf);
}
// 取得 assembly 項下的元件 id，因為只有一個零件，採用 index 0 取出其 featID
var components = assembly.ListFeaturesByType(true, pfcCreate ("pfcFeatureType").FEATTYPE
// 此一 featID 為組立件中的第一個零件編號，也就是樂高人偶的 body
var featID = components.Item(0).Id;

ids.Append(featID);
// 在 assembly 模型中建立子零件所對應的路徑
var subPath = pfcCreate("MpfcAssembly").CreateComponentPath(assembly, ids);
subassembly = subPath.Leaf;
// 以下針對 body 的 A_13 軸與 DTM1 基準面及右臂的 A_4 軸線與 DTM1 進行對齊與面接約束
var asmDatums = new Array("A_13", "DTM1");
var compDatums = new Array("A_4", "DTM1");
// 組立的關係變數為對齊與面接
var relation = new Array (pfcCreate ("pfcComponentConstraintType").ASM_CONSTRAINT_ALIGN,
// 組立元件則為軸與平面
var relationItem = new Array(pfcCreate("pfcModelItemType").ITEM_AXIS, pfcCreate("pfcModel
// 建立約束條件變數，軸採對齊而基準面則以面接進行約束
var constrs = pfcCreate ("pfcComponentConstraints");
for (var i = 0; i < 2; i++)
{
    // 設定組立參考面，asmItem 為 model item
var asmItem = subassembly.GetItemByName (relationItem[i], asmDatums [i]);
    // 若無對應的組立參考面，則啟用互動式平面選擇表單 flag
if (asmItem == void null)
{
    interactFlag = true;
    continue;
}

    // 設定零件參考面，compItem 為 model item
var compItem = componentModel.GetItemByName (relationItem[i], compDatums[i]);

```

```

        if (compItem == void null)
        {
            interactFlag = true;
            continue;
        }

        // 採用互動式設定相關的變數
        var MpfcSelect = pfcCreate ("MpfcSelect");
        var asmSel = MpfcSelect.CreateModelItemSelection (asmItem, subPath);
        var compSel = MpfcSelect.CreateModelItemSelection (compItem, void null);
        var constr = pfcCreate("pfcComponentConstraint").Create (relation[i]);
        constr.AssemblyReference = asmSel;
        constr.ComponentReference = compSel;
        // 因為透過軸線對齊，第一 force 變數需設為 true
        constr.Attributes = pfcCreate("pfcConstraintAttributes").Create (true, false);
        // 將互動選擇相關資料，附加在程式約束變數之後
        constrs.Append(constr);
    }
    // 設定組立約束條件，以 asmcomp 特徵進行約束條件設定
    // 請注意，第二個變數必須為 void null 表示零件對零件進行約束，若為 subPath，則零件會與原始零件組立
    asmcomp.SetConstraints (constrs, void null);
    /**----- LEGO_ARM_LT 左手上臂-----**/
    var descr = pfcCreate ("pfcModelDescriptor").CreateFromFileName ("v:/home/lego/man/LEGO_ARM_LT.prt");
    var componentModel = session.GetModelFromDescr(descr);
    var componentModel = session.RetrieveModel(descr);
    if (componentModel != void null)
    {
        // 注意這個 asmcomp 即為設定約束條件的本體
        // asmcomp 為特徵物件，直接將零件，以 transf 座標轉換放入組立檔案中
        var asmcomp = assembly.AssembleComponent(componentModel, transf);
    }
    // 取得 assembly 項下的元件 id，因為只有一個零件，採用 index 0 取出其 featID
    var components = assembly.ListFeaturesByType(true, pfcCreate ("pfcFeatureType").FEATTYPE_PART);
    var ids = pfcCreate ("intseq");
    // 因為左臂也是與 body 進行約束條件組立，因此取 body 的 featID
    // 至此右臂 id 應該是 featID+1，而左臂則是 featID+2
    ids.Append(featID);
    // 在 assembly 模型中建立子零件所對應的路徑
    var subPath = pfcCreate("MpfcAssembly").CreateComponentPath(assembly, ids);
    subassembly = subPath.Leaf;
    var asmDatums = new Array("A_9", "DTM2");
    var compDatums = new Array("A_4", "DTM1");
    var relation = new Array (pfcCreate ("pfcComponentConstraintType").ASM_CONSTRAINT_ALIGN,
    var relationItem = new Array(pfcCreate("pfcModelItemType").ITEM_AXIS, pfcCreate("pfcModelItemType").ITEM_AXIS);
    // 建立約束條件變數
    var constrs = pfcCreate ("pfcComponentConstraints");
    for (var i = 0; i < 2; i++)

```

```

{
    // 設定組立參考面, asmItem 為 model item
    var asmItem = subassembly.GetItemByName (relationItem[i], asmDatums [i]);
    // 若無對應的組立參考面, 則啟用互動式平面選擇表單 flag
    if (asmItem == void null)
    {
        interactFlag = true;
        continue;
    }

    // 設定零件參考面, compItem 為 model item
    var compItem = componentModel.GetItemByName (relationItem[i], compDatums [i]);
    if (compItem == void null)
    {
        interactFlag = true;
        continue;
    }

    // 採用互動式設定相關的變數
    var MpfcSelect = pfcCreate ("MpfcSelect");
    var asmSel = MpfcSelect.CreateModelItemSelection (asmItem, subPath);
    var compSel = MpfcSelect.CreateModelItemSelection (compItem, void null);
    var constr = pfcCreate("pfcComponentConstraint").Create (relation[i]);
    constr.AssemblyReference = asmSel;
    constr.ComponentReference = compSel;
    constr.Attributes = pfcCreate("pfcConstraintAttributes").Create (true, false);
    // 將互動選擇相關資料, 附加在程式約束變數之後
    constrs.Append(constr);
}
// 設定組立約束條件, 以 asmcomp 特徵進行約束條件設定
// 請注意, 第二個變數必須為 void null 表示零件對零件進行約束, 若為 subPath, 則零件會與原始零件
asmcomp.SetConstraints (constrs, void null);
/**----- LEGO_HAND 右手手腕-----**/
// 右手臂 LEGO_ARM_RT.prt 基準 A_2, DTM2
// 右手腕 LEGO_HAND.prt 基準 A_1, DTM3
var descr = pfcCreate ("pfcModelDescriptor").CreateFromFileName ("v:/home/lego/man/LEGO_1
var componentModel = session.GetModelFromDescr(descr);
var componentModel = session.RetrieveModel(descr);
if (componentModel != void null)
{
    // 注意這個 asmcomp 即為設定約束條件的本體
    // asmcomp 為特徵物件, 直接將零件, 以 transf 座標轉換放入組立檔案中
    var asmcomp = assembly.AssembleComponent (componentModel, transf);
}
// 取得 assembly 項下的元件 id, 因為只有一個零件, 採用 index 0 取出其 featID
var components = assembly.ListFeaturesByType(true, pfcCreate ("pfcFeatureType").FEATTYPEP
var ids = pfcCreate ("intseq");

```

```

// 組立件中 LEGO_BODY.prt 編號為 featID
// LEGO_ARM_RT.prt 則是組立件第二個置入的零件， 編號為 featID+1
ids.Append(featID+1);
// 在 assembly 模型中，根據子零件的編號，建立子零件所對應的路徑
var subPath = pfcCreate("MpfcAssembly").CreateComponentPath(assembly, ids);
subassembly = subPath.Leaf;
// 以下針對 LEGO_ARM_RT 的 A_2 軸與 DTM2 基準面及 HAND 的 A_1 軸線與 DTM3 進行對齊與面接
var asmDatums = new Array("A_2", "DTM2");
var compDatums = new Array("A_1", "DTM3");
// 組立的關係變數為對齊與面接
var relation = new Array (pfcCreate ("pfcComponentConstraintType").ASM_CONSTRAINT_ALIGN,
// 組立元件則為軸與平面
var relationItem = new Array(pfcCreate("pfcModelItemType").ITEM_AXIS, pfcCreate("pfcModelItemType").ITEM_PLANE);
// 建立約束條件變數，軸採對齊而基準面則以面接進行約束
var constrs = pfcCreate ("pfcComponentConstraints");
for (var i = 0; i < 2; i++)
{
    // 設定組立參考面，asmItem 為 model item
    var asmItem = subassembly.GetItemByName (relationItem[i], asmDatums [i]);
    // 若無對應的組立參考面，則啟用互動式平面選擇表單 flag
    if (asmItem == void null)
    {
        interactFlag = true;
        continue;
    }
    // 設定零件參考面，compItem 為 model item
    var compItem = componentModel.GetItemByName (relationItem[i], compDatums [i]);
    if (compItem == void null)
    {
        interactFlag = true;
        continue;
    }
    // 採用互動式設定相關的變數
    var MpfcSelect = pfcCreate("MpfcSelect");
    var asmSel = MpfcSelect.CreateModelItemSelection(asmItem, subPath);
    var compSel = MpfcSelect.CreateModelItemSelection (compItem, void null);
    var constr = pfcCreate("pfcComponentConstraint").Create (relation[i]);
    constr.AssemblyReference = asmSel;
    constr.ComponentReference = compSel;
    // 因為透過軸線對齊，第一 force 變數需設為 true
    constr.Attributes = pfcCreate("pfcConstraintAttributes").Create (true, false);
    // 將互動選擇相關資料，附加在程式約束變數之後
    constrs.Append(constr);
}
// 設定組立約束條件，以 asmcomp 特徵進行約束條件設定
// 請注意，第二個變數必須為 void null 表示零件對零件進行約束，若為 subPath，則零件會與原始零件

```

```

asmcomp.SetConstraints (constrs, void null);
// 利用函式呼叫組立左手 HAND
axis_plane_assembly(session, assembly, transf, featID, 2,
    "LEGO_HAND.prt", "A_2", "DTM2", "A_1", "DTM3");
// 利用函式呼叫組立人偶頭部 HEAD
// BODY id 為 featID+0, 以 A_2 及 DTM3 約束
// HEAD 則直接呼叫檔案名稱, 以 A_2, DTM2 約束
axis_plane_assembly(session, assembly, transf, featID, 0,
    "LEGO_HEAD.prt", "A_2", "DTM3", "A_2", "DTM2");
// Body 與 WAIST 採三個平面約束組立
// Body 組立面為 DTM4, DTM5, DTM6
// WAIST 組立面為 DTM1, DTM2, DTM3
three_plane_assembly(session, assembly, transf, featID, 0, "LEGO_WAIST.prt", "DTM4", "DTM5", "DTM6");
// 右腳
axis_plane_assembly(session, assembly, transf, featID, 6,
    "LEGO_LEG_RT.prt", "A_8", "DTM4", "A_10", "DTM1");
// 左腳
axis_plane_assembly(session, assembly, transf, featID, 6,
    "LEGO_LEG_LT.prt", "A_8", "DTM5", "A_10", "DTM1");
// 紅帽
axis_plane_assembly(session, assembly, transf, featID, 5,
    "LEGO_HAT.prt", "A_2", "TOP", "A_2", "FRONT");
</script>
</body>
</html>
'''

    return outstring

```

cd2ag10 報告 (2ag10)

以下為各週報告

第十組組員

小組網站:<http://goo.gl/aztMHi>

40123116-吳羽閔

40123118-吳聲麟

40123153-戴志軒

第八週考試摘要 (2ag10)

小組網站:<http://goo.gl/yi3qJ5>

2014S CD Week8

課程教材:

A collaborative writing approach to wikis

Collabrative engine for distributed mechanical design

Web-based collaborative engineering support system

第八週考試題目

下列題目完成後, 必須 (1) 將程式碼送到個人的 Bitbucket repository 下 (2) 程式可在近端與雲端部署執行 (3) 在 wiki.mde.tw 個人第八週心得中留下 Bitbucket 與雲端執行連結 (4) 整理出一份小考第一 (二, 或三) 題的 PDF 檔案, 寄到 course@mde.tw, 標題為: 學號 -小考第一 (二, 或三) 題, 內容必須包含程式碼, 解題過程, 解題心得, Bitbucket 連結, 雲端網址等資料.

(第一題) 請寫一個執行時可以列出 9×9 乘法表的網際 Python 程式, 然後 Push 到個人 bitbucket 空間, 而且同步指到 OpenShift 個人帳號上執行.

(第二題) 請將上述執行過程錄為 flv 後, 上傳到個人的 Vimeo 空間中, 並將網址回報到各組網站 (dokuwiki 與 CMSimply) 與報告中, 並且將相關心得與報告連結登錄到 wiki.mde.tw 第八週的分組頁面中.

(第三題) (協同計分, 分組進行) 請在各組的雲端 dwiki 中, 根據下列 40 個帳號與密碼, 新增對應的使用者帳號與密碼後, 將雲端網址登錄在 wiki.mde.tw 各組第八週頁面中, 並說明操作過程與心得後, 將心得整理成 pdf 後繳交到 course@mde.tw.

40 個帳號與密碼

58B39 , 9J35UAVM

3624D , QANF34CW

7345B , 3PAFXKWZ

358DA , E6RJFKW4

3BC7B , HG2ASNYH

CA55C , XUZTHWQK

D2756 , UHK2W3D2

368B8 , A5QGYA6W

7948A , PF278WDQ

C65C4 , GT4KBCXU

A8964 , 3RPQSW2U

22422 , 7E57K7F3
9A5B4 , 4MVKRE5Z
B5A76 , 3DTAAHUF
57388 , NP39FGXR
8A833 , C7DNBHCQ
29AB8 , 6KMGK73Z
5ABD2 , PV5FH722
86293 , PJ69FBMS
9DCBC , U5HR6QR8
276DD , URE9FNWD
599AA , X2P6CTXF
9C449 , DKRN3V59
64236 , 86UWN3E9
43AAC , MNCJZCAX
73B93 , QX945VJJ
36283 , M3MQGUXD
7447C , TQZVDKPT
C73AB , MSP4GPPX
8284B , XMT8W9RD
62454 , SD4C7V89
44B3C , 636DBRJC
C75CB , M66RMMQ2
4A3CB , FF485EQ4
7D248 , FGJHQDAS
A7CC4 , R47AHA4Y
99BCC , RNQYZGQZ
6DA77 , HRCDP9D8
357AB , CKB4Q2EC
A9525 , JDVX75ST

第八週協同設計練習題目:

建立各組 OpenShift 上的 dokuwiki

<http://ethercalc.tw/> Openshift 上的 ethercalc

近端的 ethercalc

用 markdown 編輯電子書: <https://github.com/progit/progit> ,

<http://johnmacfarlane.net/pandoc/epub.html>

期中考試與分組報告必選題

請以三組共 9 個人的情況下 (座位為 3×3 配置), 分析 2014s_week3 的協同設計題目, 如何配置座位可以得到團隊的最大配分. 假如將組別擴大為五組共 15 人的情況下 (座位為 4×4 配置, 則團隊座位安排後的最大配合又是多少?

上課影片 <http://vimeo.com/user24079973/videos>

第八週報告 (2ag10)

小組網站:<http://goo.gl/YCNd7p>

```
import cherrypy
class HelloWorld(object):
    @cherrypy.expose
    def index(self, var1=9, var2=9):
        # initialize outstring
        outstring = ""
        # initialize count
        count = 0
        d = int(var1)
        e = int(var2)+1
        for i in range(1, d):
            for j in range(1, e):
                count += 1
                #print(count)
                if count%(int(var2)) == 0:
                    outstring += "<td>"+str(i) + "*" + str(j) + "=" + str(i*j) + "</td>"+
                else:
                    outstring += "<td>"+str(i) + "x" + str(j) + "=" + str(i*j) + "</td>"+
            return "<table border=2><tr><td>99 乘法表</td></tr><tr>"+outstring
#http://127.0.0.1:8080/index?var1=10&var2=20
#" "*4
```

```
cherrypy.quickstart(HelloWorld())
#application=cherrypy.Application(HelloWorld()) # 將符號打開上傳 openshift 即可使用, 並刪除
```

以上程式碼即可跑出 99 乘法表.

9x9openshift: <http://9x9-cadp13ag8.rhcloud.com/> (因 openshift 空間不足，故利用第九週程式，程式碼略有差別，主要為多新增兩個輸入表單，其餘相同)

同步 bitbucket: <https://bitbucket.org/40123153/input9x9> (同上)

————我是分隔線————

影片 (由於網路超慢，怕錄製影片過大所以只錄製最後結果)

<https://vimeo.com/92003966>

P.S 由於題目理解錯誤，所以上述影片為失效。

第九週考試摘要 (2ag10)

小組網站:<http://goo.gl/8nPx2Y>

2014S CD Week9

課程教材:

A collaborative writing approach to wikis

Collabrative engine for distributed mechanical design

Web-based collaborative engineering support system

期中考試題目

下列題目完成後，必須 (1) 將程式碼送到個人的 Bitbucket repository 下 (2) 程式可在近端與雲端部署執行 (3) 在 wiki.mde.tw 個人第九週心得中留下 Bitbucket 與雲端執行連結以及其它參考連結 (4) 整理出一份期中考第一 (二, 或三) 題的 PDF 檔案, 寄到 course@mde.tw, 標題為: cda_學號_姓名_期中考第一 (二, 或三) 題 (乙班將 cda 改為 cdb), 內容必須包含程式碼, 解題過程, 解題心得, Bitbucket 連結, 雲端網址等資料。

(第一題 30%) 請寫一個執行時可以列出以十為底對數表的網際 Python 程式, 然後 Push 到個人 bitbucket 空間, 而且同步指到 OpenShift 個人帳號上執行。

(第二題 40%) 請在個人的 OpenShift 平台上建立一個能夠列印出與九九乘法表結果完全相同的網際程式, 接著在乘法表上端加上兩個輸入表單, 讓使用者輸入兩個整數, 按下送出鍵後, 程式會列出以此兩個整數為基底的乘法表, 例如: 若兩個欄位都輸入: 9, 則列出九九乘法表, 若輸入 9, 20, 則列出 9×20 的乘法表。

(第三題 30%) 請在各組的雲端 dokuwiki 中, 新增帳號與密碼都是由 abc001 ~ abc399 字串所組成的 399 名用戶登入對應資料, 並將製作過程與驗證流程拍成 flv 後上傳到個人的 Vimeo 資料區, 並將連結放在個人第九週頁面。

第一題參考資料: 對數表與應用, 當 $x=10$, 對應到 0 行的值, 表示要對 1.00 取以十為底的對數, 所得到的值為 $\text{math.log}(1.00, 10)=0$, 而 $x=10$ 對應到 1 行的值, 表示要對 1.01 取以十為底的對數, 所得到的值為 $\text{math.log}(1.01, 10)=0.004321373782642578$ 然後再乘上 10000, 並且只取整數, 所以對應表

的值為 43, 也就是表中的 0043, 當 $x=20$, 而且對應到第 9 行的值, 則為 $\text{math.log}(2.09, 10)=0.32014628611105395$, 然後再乘上 10000, 只取整數, 所以對應表上的值為 3201.

第一題參考: Python 中 `str()` 可以將整數或浮點數轉為字串, `int()` 則可以將字串轉為整數, `round(1.0123, 2)` 表示只取小數點後兩位, `math.log(x, 10)` 表示對 x 取以 10 為底的 \log 值, html 表格請參考.

期中成績評量

請各組依據 2014s_week7 中的說明完成期中報告.(最後繳交期限為 2014.4.26 晚上 12:00)

成績評量時, 將依據各組在 wiki.mde.tw 中的頁面進行評量, 期中成績包括平時成績 (參考個人自評成績)、第八週考試與期中考試成績、期中報告成績等.

上課影片 <http://vimeo.com/user24079973/videos>

第九週報告 (2ag10)

小組網站:<http://goo.gl/Hy7Ktn>

第一題:

Log 以 10 為底之表單 openshift: <http://log-cadp13ag8.rhcloud.com/>

bitbucket: <https://bitbucket.org/40123153/log10-1.00-1.99>

因為迴圈關係, 多了一個 21...還在更新中。(解決)

P.S 已利用 `elif` 判斷式將最後的 21 給剔除!

當在最後 20 的地方下 `elif` 將

給剔除, 故不在往下做表格及列印出 21 數字。

```
import cherrypy
import os
import math
# 1. 導入所需模組

# 2. 設定近端與遠端目錄
# 確定程式檔案所在目錄, 在 Windows 有最後的反斜線
_curdir = os.path.join(os.getcwd(), os.path.dirname(__file__))
# 設定在雲端與近端的資料儲存目錄
if 'OPENSIFT_REPO_DIR' in os.environ.keys():
    # 表示程式在雲端執行
    download_root_dir = os.environ['OPENSIFT_DATA_DIR']
    data_dir = os.environ['OPENSIFT_DATA_DIR']
else:
```

```

# 表示程式在近端執行
download_root_dir = _curdir + "/local_data/"
data_dir = _curdir + "/local_data/"

# 3. 建立主物件
class HelloWorld(object):
    @cherry.py.expose
    def index2(self, input1=None, input2=None):
        return "Hello world!" + str(input1)
    @cherry.py.expose
    def inputform(self, input1=None, input2=None):
        return "input form" + str(input1)
    #index.exposed = True
    @cherry.py.expose
    def index(self):
        out=""
        c = 0
        k = 0
        for i in range(0,11):
            for j in range (0,10):
                k += 1
                c = 10 + i
                d = c/10+j/100
                e = (int(math.log(d,10)*10000))
                if (k%10 == 0 and k<=100):
                    out += "<td>" + str(e) + "</td></tr><td>" + str(c+1) + "</td><br />"
                elif k%100 == 0:
                    out += "<td>" + str(e) + "</td><br />"
                else:
                    out += "<td>" + str(e) + "</td>"
                #print("log=",int(math.log(d,10)*10000))
                #print("\n")
        return "<table border=1><tr><td>x</td><td>0</td><td>1</td><td>2</td><td>3</td><t

```

4. 安排啟動設定

配合程式檔案所在目錄設定靜態目錄或靜態檔案

```

application_conf = {'/static':{
    'tools.staticdir.on': True,
    'tools.staticdir.dir': _curdir+"/static"},
    '/downloads':{
    'tools.staticdir.on': True,
    'tools.staticdir.dir': data_dir+"/downloads"}
}

```

5. 在近端或遠端啟動程式

利用 HelloWorld() class 產生案例物件

```

root = HelloWorld()
# 假如在 os 環境變數中存在 'OPENSHIFT_REPO_DIR', 表示程式在 OpenShift 環境中執行
if 'OPENSHIFT_REPO_DIR' in os.environ.keys():
    # 雲端執行啟動
    application = cherrypy.Application(root, config = application_conf)
else:
    # 近端執行啟動
    '''
    cherrypy.server.socket_port = 8083
    cherrypy.server.socket_host = '127.0.0.1'
    '''
    cherrypy.quickstart(root, config = application_conf)

```

————我是分隔線————

第二題:

9×9 乘法表，並且有兩 input 可以改變 9×9 乘法表的範圍。例：input:9 and 20 則為 9×20 乘法表。

openshift: <http://9x9-cadp13ag8.rhcloud.com/> (備註:i 為前值，j 為後值)

bitbucket: <https://bitbucket.org/40123153/input9x9>

```

import cherrypy
class HelloWorld(object):
    @cherrypy.expose
    def index(self, var1=9, var2=9):
        # initialize outstring
        outstring = ""
        # initialize count
        count = 0
        d = int(var1)+1
        e = int(var2)+1
        for i in range(1, d):
            for j in range(1, e):
                count += 1
                if count%int(var2) == 0:
                    outstring += "<td>"+str(i) + "*" + str(j) + "=" + str(i*j) + "</td>"+
                else:
                    outstring += "<td>"+str(i) + "x" + str(j) + "=" + str(i*j) + "</td>"+
            return "<table border=3><tr><td>99 乘法表</td></tr><tr>"+outstring+'''<br/><form
            i 我是前值:<input type="text" name="var1"><br />
            j 我是後值:<input type="text" name="var2"><br />
            <input type="submit" value="send">
            </form>
            '''

```

```
#http://127.0.0.1:8080/index?var1=10&var2=20
```

```
#cherrypy.quickstart(HelloWorld())  
application=cherrypy.Application(HelloWorld())
```

————我是分隔線————

第三題:

新增帳號與密碼都是由 abc001 ~ abc399 字串所組成的 399。

影片解說: <https://vimeo.com/92574166>

首先，我們必須要先有 abc001~abc399 的帳號密碼值，故先用程式迴圈跑出。
(因為懶得寫輸出檔，或者是不熟悉所以不冒險去寫。)

將值複製至文件內將空白修掉，利用 EXCEL 把密碼在複製至 B 欄位。存成 CSV 檔

同樣利用文件將逗點修掉 (如果沒記錯空白處為一個 TAB，為了安全還是複製之前的。)

再利用程式，修改讀取的文件，將密碼處改為數列¹

跑出 user 之後利用 FZ 將檔案覆蓋，則可以在雲端上登錄。(為了確認完全傳入，將 abc001 改為管理權限，確認帳號輸入)

dokuwiki_openshift: <http://dokuwiki-cadp13ag8.rhcloud.com/doku.php>

bitbucket: https://bitbucket.org/40123153/dokuwiki_abc

第三題的程式碼實在不知道要打什麼….

第十二週報告 (2ag10)

小組網站:<http://goo.gl/28PW52>

第十二週任務

請各組將第八週與第九週考試的摘要報告放入 Github 協同專案中的分組報告區，並將內容放入各組控管的同步 OpenShift 網站中。(佔期末成績 5 分)

github 小組倉儲:<https://github.com/2014cdag10>

github 大分組倉儲:<https://github.com/coursemdetw/2014cda>

請各組設法利用 CherryPy 與 Pro/Web.Link 技術，在 Github 協同專案中建立一個能夠透過連結或表單控制 Cube 零件，a, b, 或 c 零件尺寸的網際協同程式，讓使用者可以藉以利用近端的 Creo 嵌入式瀏覽器控制 Cube 的尺寸後列出該零件的體積大小。(佔期末成績 5 分)

1. Creo 必須使用 64 位元版本 (配合 Windows 7 操作系統)
2. Creo web_enable_javascript 設為 on, regen_failure_handling 設為 resolve_mode
3. IE → 工具 → 網際網路選項 → 安全性 → 信任的網站 → 自訂等級允許信任網站執行 ActiveX
4. cda 專案必須將 <http://cdag10-40123153.rhcloud.com/> 設為信任網站
5. 開啟 Creo 2.0, 建立一個 cube, relations: $d0 = a$, $d1 = b$, $d2 = c$ (a, b, c 為 local parameters)
6. 然後在嵌入式 IE, 連接到 <http://cdag10-40123153.rhcloud.com/cdag10/cube10> (# 在 parameter 中將函式名稱改為 cube10)
7. 執行 <http://cdag10-40123153.rhcloud.com/cdag10/fourbar10> (# 在 parameter 中將函式名稱改為 fourbar10) 之前則需要先下載四連桿零件放在 V: 目錄中, 並開啟一個空的組立檔案, 執行時 Pro/Web.Link 程式會自動進行連桿組立.
8. Pro/Web.Link 應用: <http://www.ptc.com/company/news/inprint/taiwan/proe5.htm>
9. 參考資料: <http://inversionconsulting.blogspot.tw/search/label/WebLink>

首先必須將 creo 2.0 的 config.pro 檔做些設定, 必須要將 regen_failure_handling 設為 resolv_mode, 使他能重繪。

再來必須將 web_enable_javascript 設為 on, 這樣才能讓 javascript 可以動作。

很重要!! 這將會成為成功與失敗的原因...

並且將 IE 關於 ActiveX 都通通啟用, 將 <http://cdag10-40123153.rhcloud.com/> 設為例外網頁。

再來就是繪圖, 將零件的三個尺寸設關係式, $d0=a$, $d1=b$, $d2=c$ 。

所有前置動作都完成了, 再來就是利用 creo 裡面內建的 IE 跑囉!

將 <http://cdag10-40123153.rhcloud.com/cdag10/cube10> 貼至 IE 的 url 中, 就會開始跑囉!

(原始尺寸為 150x150x150 的方塊)

第一次:3375000

第二次:4096000

第三次:4913000

第四次:5832000

第五次:6859000

大致上就這樣吧...不過 c 變數忘記動到...不過只是照 a 與 b 同樣的型式...將 a 或 b 複製下來改成 c 就可以三軸同動了。

第十二週評分

40123116 : 80 分

40123118 : 80 分

40123153 : 80 分

網際鼓式煞車設計 (2ag11)

有關鼓式煞車

程式設計架構

鼓式煞車

結果與討論

這裡是結果與討論

網際鼓式煞車設計 (2ag12)

有關鼓式煞車

程式設計架構

鼓式煞車

結果與討論

這裡是結果與討論

組員：

40123124

[illegible]

考試摘要 <http://dokuwiki-40123155.rhcloud.com/doku.php>

- <http://dokuwiki-40123155.rhcloud.com/lib/exe/fetch.php?media=%E8%AA%B2%E7%A8%8B%E6%95%>

<http://dokuwiki-40123155.rhcloud.com/lib/exe/fetch.php?media=%E8%AA%B2%E7%A8%8B%E6%95%>

組員自評:

40123155 : 80 分

40123124 : 70 分

40123138 : 75 分

<<<<<<< HEAD 網際鼓式煞車設計 (2ag14) ===== 網際鼓式煞車設計 (2ag10) >>>>>>> 26a5b9324111e15820203a5b96463a4196def80e ==

有關鼓式煞車

鼓式煞車

結果與討論

這裡是結果與討論

網際鼓式煞車設計 (2ag15)

有關鼓式煞車

程式設計架構

鼓式煞車

結果與討論

這裡是結果與討論

網際鼓式煞車設計 (2ag16)

有關鼓式煞車

程式設計架構

鼓式煞車

結果與討論

這裡是結果與討論

第八週摘要報告

以協同方式寫 Wiki

本文目的在於讓學生利用開發的 wiki 與應用快速原型的協作式寫作的開發方法。而本文也由定性數據收集和分析方法的來評價。最後，對於協作方式來寫作的方式，其影響極其討論協議到組別與軟件開發方面的考慮和教學相關的要

求問題研究方法 wiki 的協作式寫作，基於設計的研究包括以下四個步驟：1. 研究的現狀與認識目的並審查與 wiki 的協作式寫作相關的問題。2. wiki 將用於促進以協同方式寫作來設計，參與和群體互動。3. 使用多種方法收集其經驗數據。4. 通過系統的評價分析，並通過各種方法收集資料。分析，設計，實施和評價是相互共存的。而缺點提出於每個週期提出，重新設計，重新實現，並重新評估。維基應用域教育在協同方式寫作方法而成的 Wiki 是適用於多種情況，期望得到的 wiki 應用在學習投資效益與協同方式來寫作。它十分靈活的，足以適應的專業條件。更具體地，它的應用領域包括不同級別的更高教育，從研究生到社會教育。除了協同寫作與學科相關，維基可以為一些應用程序的開發系統來使用，如產生教材，網路上互相評論，並收集數據於一種項目。

結論和未來研究方向 1. 這項結果不能限制於狹小的研究的範圍，即使調查結果反映了那些報導中的一致性研究文獻。2. 用來判斷以不同方式來呈現協同，尤其是同組評議，這擁有極高的教育價值與分析性思維，並參與交流。可以促進團體間的合作和時間，wiki 打開了協同寫作上新的視野，而小組互動絕不是容易的事情，解決了技術，教學和文化各種問題。未來的工作將更顯的協同的重要性，wiki 以此設計為基礎下更加精進。重要的是，它更進一步影響高等教育對於 wiki 的使用。最後，它也進一步實踐於系統評論的基礎

第九週摘要報告

題目一：請寫一個執行時可以列出以十為底對數表的網際 Python 程式，然後 Push 到個人 bitbucket 空間，而且同步指到 OpenShift 個人帳號上執行。

資料: <https://bitbucket.org/40123158/test>

題目三：請在個人的 OpenShift 平台上建立一個能夠列印出與九九乘法表結果完全相同的網際程式，接著在乘法表上端加上兩個輸入表單，讓使用者輸入兩個整數，按下送出鍵後，程式會列出以此兩個整數為基底的乘法表，例如：若兩個欄位都輸入：9，則列出九九乘法表，若輸入 9, 20，則列出 9×20 的乘法表。

資料：第 17 組 dokuwiki，新增 abc001 ~ abc399 用戶資料 <http://wikig17-weis.rhcloud.com/doku.php?id=start>

上傳影片 - 個人 Vimeo <https://vimeo.com/92577964>

Bitbucket 連結 <https://bitbucket.org/40123137/week9> Bitbucket 連結 <https://bitbucket.org/40123137/week9>

網際鼓式煞車設計 (2ag18)

有關鼓式煞車 g18

程式設計架構

鼓式煞車

結果與討論

這裡是結果與討論

網際鼓式煞車設計 (2ag21)

有關鼓式煞車

程式設計架構

鼓式煞車

結果與討論

這裡是結果與討論 00123

網際 OpenJSCAD 程式設計 (coursemdetw)

將 Spur 改為凸輪零件成型

設計程式架構

定義凸輪設計公式

結果與討論

有關凸輪程式設計的結果與討論