



# Utilizing Python in RayStation for Proton Therapy Efficiency and Standardization

Samantha Hedrick, PhD, DABR



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

## Table of Contents



- Python Basics
- Python and RayStation
- Python and Proton Therapy
- Tips

# The Zen of Python

The Zen of Python, by Tim Peters:

Simple is better than complex.  
Complex is better than complicated.

Readability counts.

There should be one-- and preferably only one --obvious way to do it.

If the implementation is hard to explain, it's a bad idea. If the implementation is easy to explain, it may be a good idea.

## Other Example



```
37 #include <iostream>
38 #include <stdio.h>
39 using namespace std;
40
41 int main()
42 {
43     int iTotal = 0, iCntr, iRes = 0;
44     cin >> "Enter the value of iTotal.\n" >> iTotal;
45
46     scanf("%d", &iTotal);
47
48     for(iCntr = 1; iCntr <= iTotal; ++iCntr) //for loop terminates if iCntr > n
49     {
50         iRes += iCntr;
51     }
52     cout << "iTotal" << iRes;
53     getchar ();
54     return 0;
55 }
56 }
```

4

Hard to read, not simple

## Python Example



```
x = [1, 2, 3, 4, 5]
y = [6, 7, 8, 9, 10]

nx = len(x)
ny = len(y)

z=[[0 for i in range(nx)] for j in range(ny)]

for i in range (0, nx):
    for j in range (0, ny):
        z[i][j] = x[i] * y[j]

for i in range (ny):
    for j in range (nx):
        print('{0:4d}'.format(z[i][j]),end=' ')
    print()
```

5

Easier to read, simple

**Training**

The image shows two side-by-side screenshots. The left screenshot is a RaySearch training schedule. It lists various courses with their durations. Two courses are highlighted with red boxes: 'Basic RayStation Scripting - 3 days' and 'RayStation Scripting - 2 days'. The right screenshot is a Google search result for 'python how to for loop'. It shows a list of search results, including 'Python For Loops' with a list of 8 exercises, 'Python For Loops - W2Schools', 'Python For Loops - W3Schools', 'Loops - Learn Python - Free Interactive Python Tutorial', and 'ForLoop - Python Wiki'. A small number '6' is visible at the bottom center of the screenshot area.

I have no formal training, I learned through the Manual and google.  
I trained in Fortran, which is like learning Latin, it's a dead language

## Python and RayStation



- Recording
- Console and State Tree

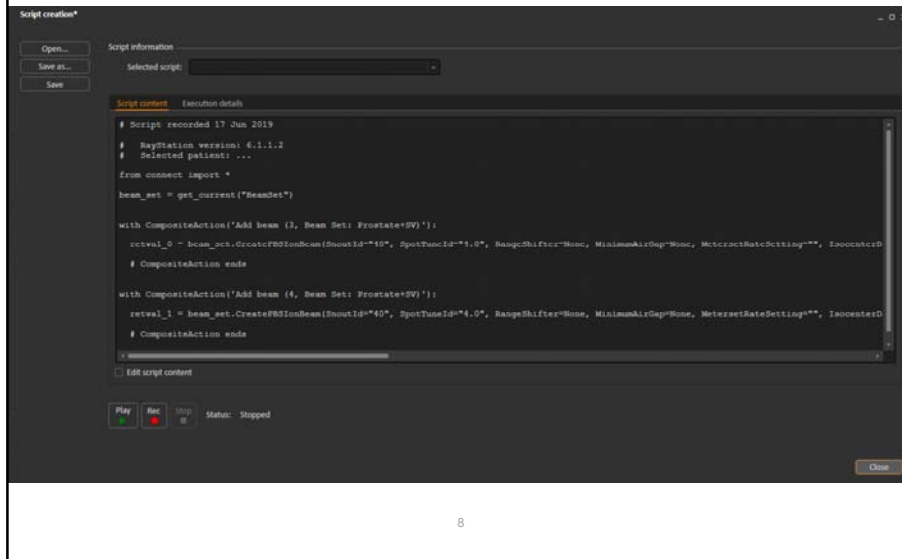
7

Python is expanding, it's already used in RayStation and it's starting to be adopted in other planning systems, like Eclipse

Python and RayStation allow the script to write to the database, not just read.

Opens up a huge world of automation

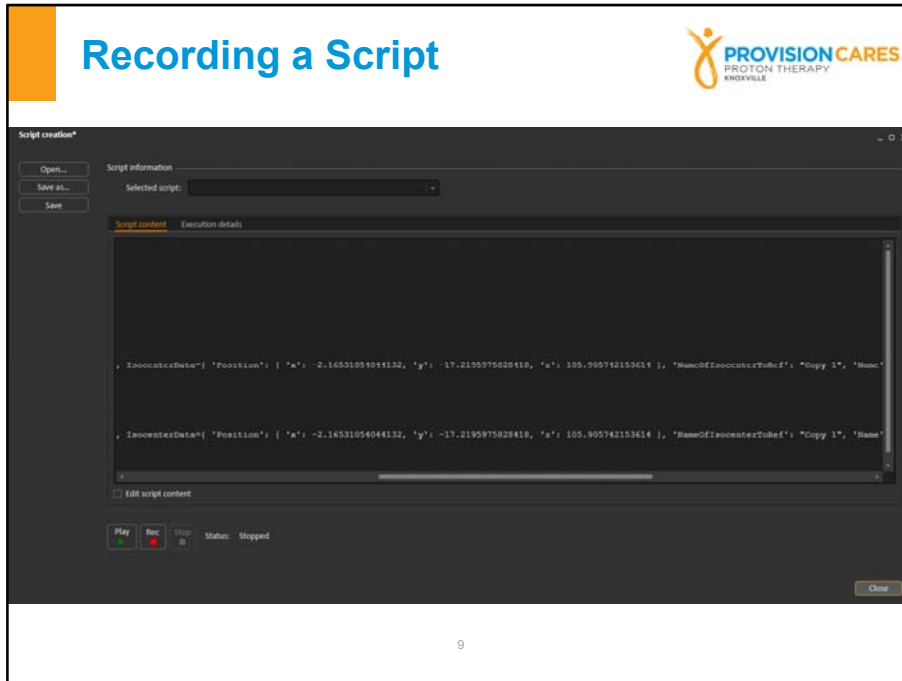
## Recording a Script



Recording a script allows you to visualize the lines of code necessary to accomplish the goal you just did through button clicks  
Very useful for getting started



## Recording a Script

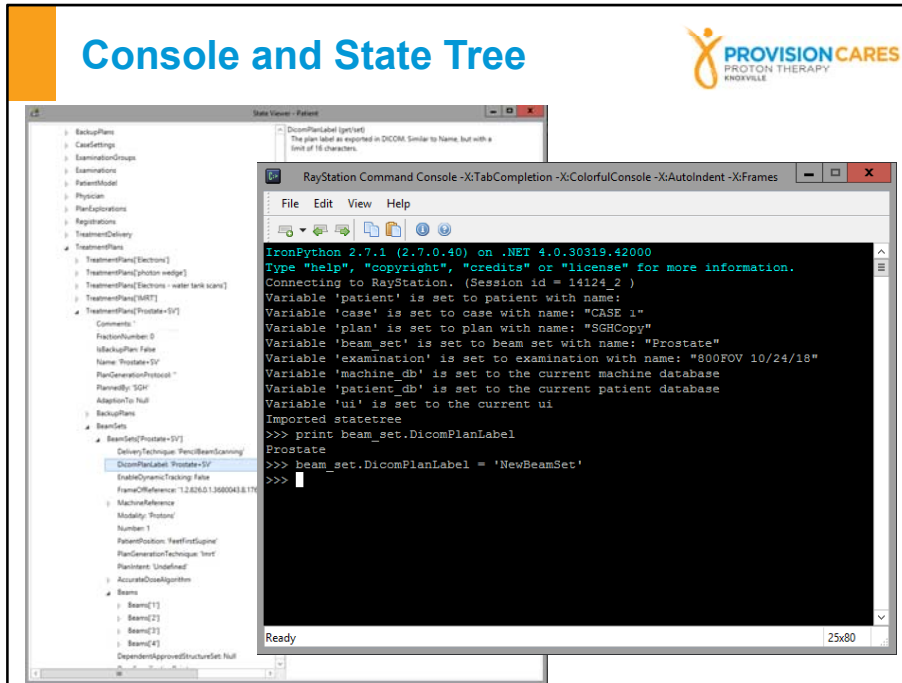


Recording has limitations

It hardcodes things like plan names, beam names, isocenter locations, etc

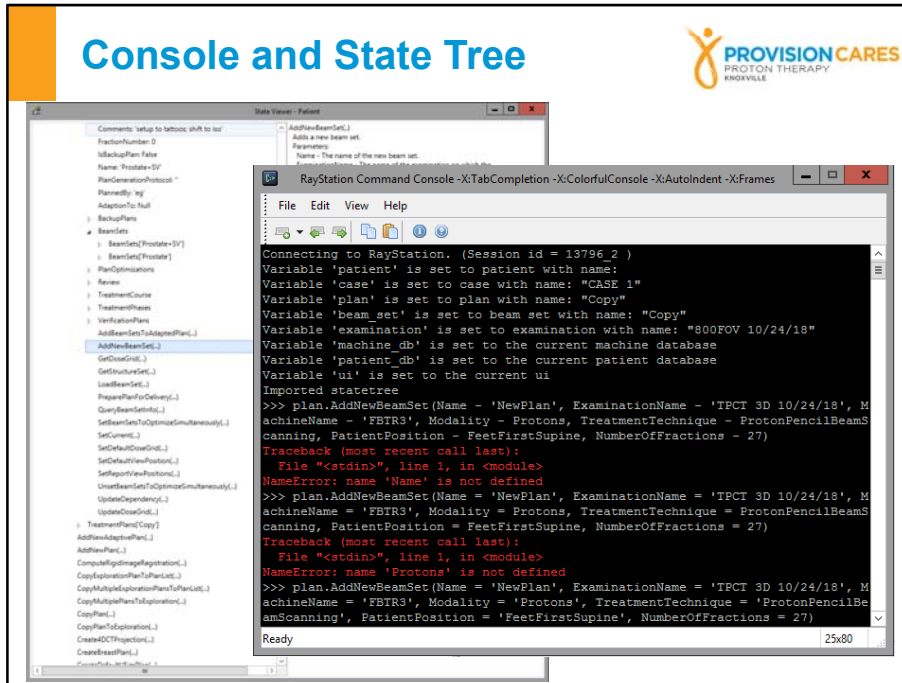
The scripter has to figure out how to make these parameters more generic and apply to a broader environment

## Console and State Tree



State Tree is the “behind the scenes” of RayStation’s use of Python  
Uses a hierarchy, from Case to Patient to Plan to Beamset to Beam  
Can access functions well beyond recording scripts

## Console and State Tree



Console is used for trial and error

In my example, I thought you used a "-" to fulfill the necessary information but it required an "="

In the second attempt, I was missing quotation marks are necessary information

## Python and RayStation



**Standardization**



**Automation**



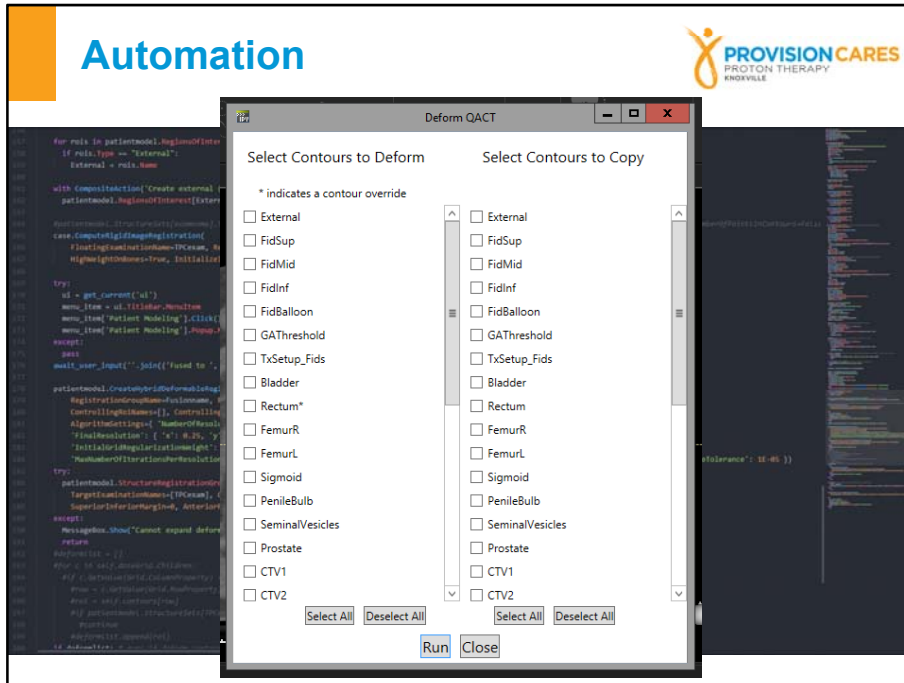
**Efficiency**

12

Scripting can be used for automation, standardization and efficiency  
The only thing it can't be used for is MD approval (mostly kidding)

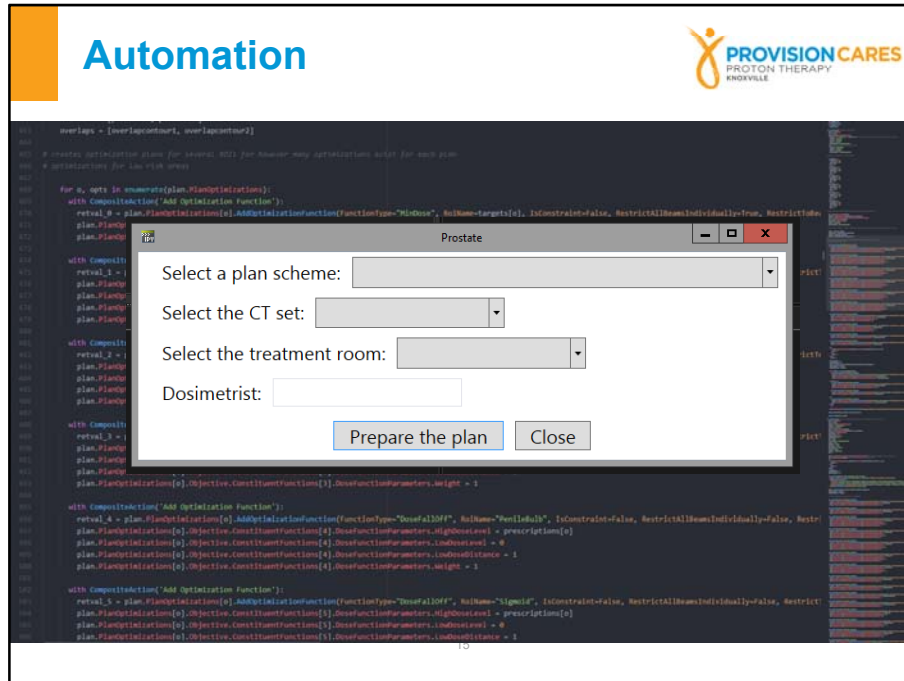


# Automation



## QACT import:

- Proton therapy requires periodic CTs during treatment to evaluate setup consistency, weight loss, tumor changes, etc
- These QACTs are tedious to evaluate, with a lot of button clicks
- Script will rename CT, apply density table, fuse with TPCT, deform OAR contours, copy target contours, calculate dose
- User has to check the contours and evaluate the dose on the QACT



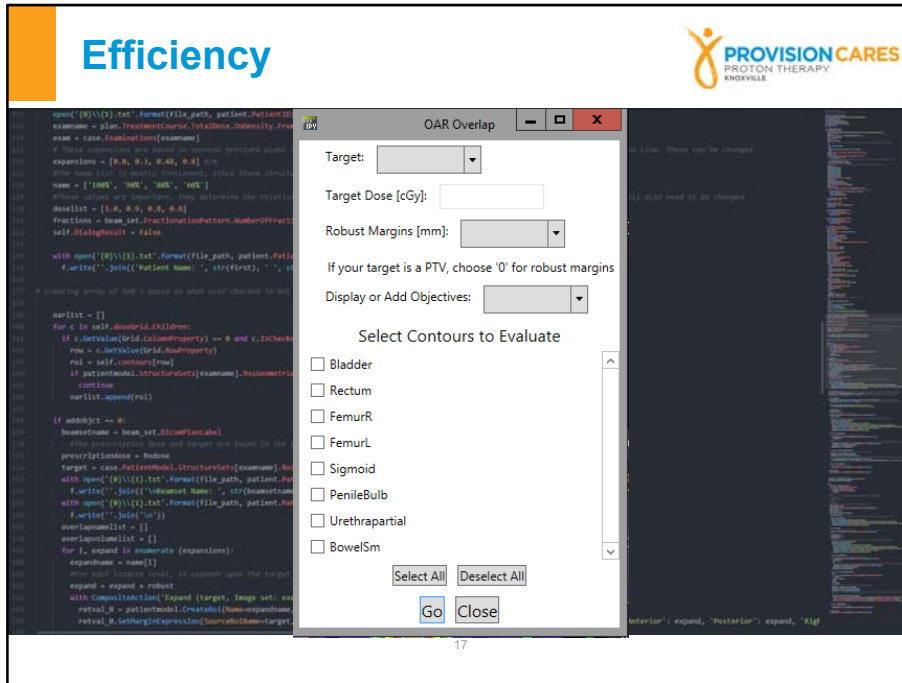
### Treatment planning:

- Some treatment sites, like head and neck, prostate, and breast, have standard beam angles and dose schemes
- Allows user to choose dose/fractionation, Treatment Planning CT, treatment room, and add their initials
- Creates new plan, creates isocenter, adds beams, adds objectives, adds clinical goals, runs optimization, pauses for review, runs perturbations, runs report





# Efficiency



## Predict DVH:

- Uses overlap of contours and targets to predict OAR sparing
- Allows the user to choose margins for robust planning or no margin for PTV planning
- Use “Display” to check your plan and see how well you met the predictions or
- Use “Add objectives” to add DVH objectives to your plan before you start planning

# Efficiency



A190503 - Notepad

File Edit Format View Help

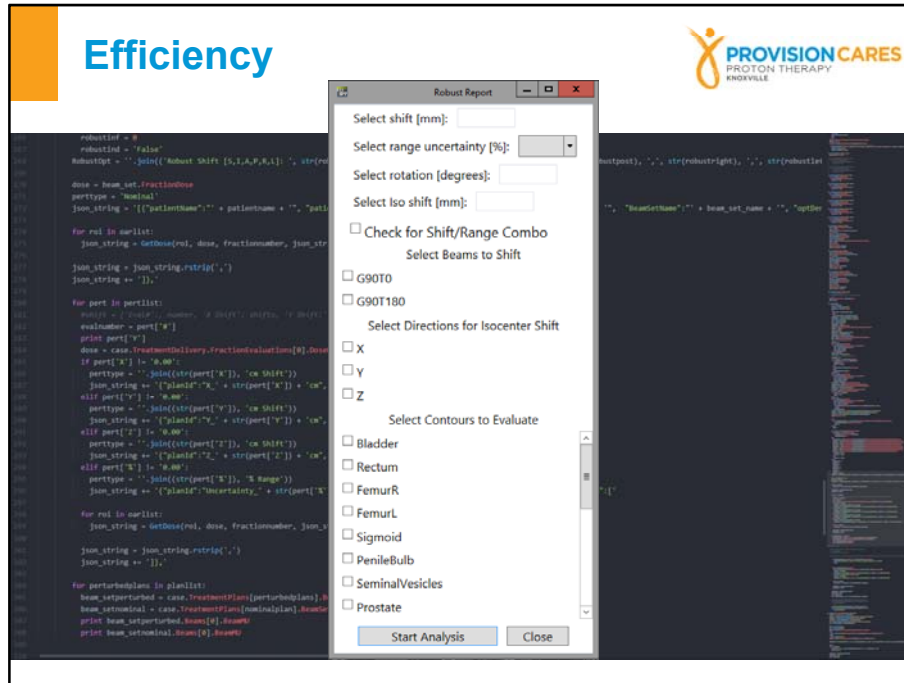
Patient Name: Patient Name

Beamsheet Name: PROSTATE+SV Target: CTV Robust: 0.3cm

Organ	Predicted Dose Volume	Actual Dose Volume
Bladder	V2400cGy = 4.9%	V2400cGy = 3.6%
Bladder	V2160cGy = 7.3%	V2160cGy = 6.5%
Bladder	V1920cGy = 9.0%	V1920cGy = 8.6%
Bladder	V1440cGy = 12.6%	V1440cGy = 12.1%
Rectum	V2400cGy = 2.5%	V2400cGy = 0.1%
Rectum	V2160cGy = 5.9%	V2160cGy = 2.5%
Rectum	V1920cGy = 8.5%	V1920cGy = 7.3%
Rectum	V1440cGy = 14.1%	V1440cGy = 14.0%
Femur_R	V2400cGy = 0.0%	V2400cGy = 0.0%
Femur_R	V2160cGy = 0.0%	V2160cGy = 0.0%
Femur_R	V1920cGy = 0.0%	V1920cGy = 0.0%
Femur_R	V1440cGy = 0.0%	V1440cGy = 0.0%
Femur_L	V2400cGy = 0.0%	V2400cGy = 0.0%
Femur_L	V2160cGy = 0.0%	V2160cGy = 0.0%
Femur_L	V1920cGy = 0.0%	V1920cGy = 0.0%
Femur_L	V1440cGy = 0.0%	V1440cGy = 0.0%
Sigmoid	V2400cGy = 0.0%	V2400cGy = 0.0%
Sigmoid	V2160cGy = 0.0%	V2160cGy = 0.0%
Sigmoid	V1920cGy = 0.0%	V1920cGy = 0.0%
Sigmoid	V1440cGy = 0.0%	V1440cGy = 0.0%
PenileBulb	V2400cGy = 0.0%	V2400cGy = 0.0%
PenileBulb	V2160cGy = 0.0%	V2160cGy = 0.0%
PenileBulb	V1920cGy = 19.6%	V1920cGy = 0.0%
PenileBulb	V1440cGy = 49.3%	V1440cGy = 3.1%

Predict DVH

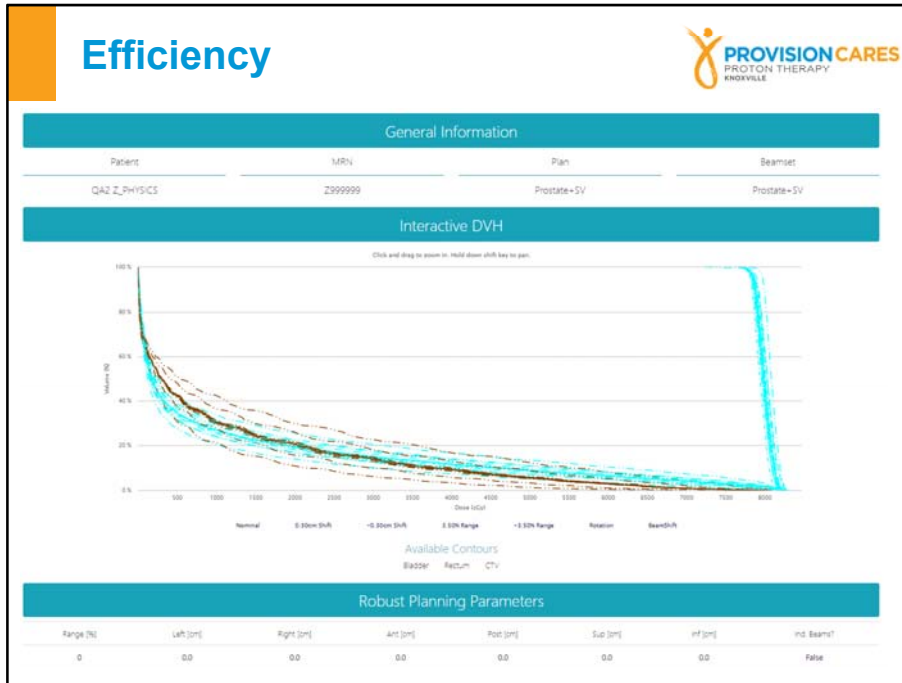
## Efficiency



### Robust report:

- Runs up to 18 perturbations, possibly more for additional beam isocenter shifts or combination perturbations
- Don't necessarily need to shift beams independently in all directions, choose fewest necessary
- Recommend outputting CTV targets and serial organs, e.g. Spinal Cord, Brainstem, etc
- Allows the dosimetrist to evaluate robustness before sending to physics, thus reducing failures and wasted time
- Allows the physicist to have an overall view of robustness before evaluating nominal dose

# Efficiency



## Robust report:

- Interactive DVH allows user to turn contours on and off to only see 1 OAR at a time for evaluation
- Interactive DVH allows user to turn perturbations on and off
  - Range uncertainty is likely occurring every day, so I'm going to evaluate the range perturbations more seriously than shifts
  - Independent beam isocenter shifts indicate gradient within the plan, so I evaluate it more closely because it can predict sensitivity to weight loss/tumor change/setup changes

# Efficiency



## Max Dose Summary Statistics

Show 10 entries Search

Plan / Structure	Max (Max[Gy])	Min (Min[Gy])	Range (Max[Gy])	Avg (Avg[Gy])	Max (Max[%])	Min (Min[%])	Range (Max[%])	Avg (Avg[%])
-0.00m Shift	82.89	89.23	19.36	79.73	1.81	-11.85	13.48	-1.61
-1.00% Range	81.68	78.08	3.60	80.43	-0.52	-0.76	0.74	-0.63
0.00m Shift	82.40	79.48	2.92	81.87	3.04	-0.53	3.87	0.48
1.00% Range	82.80	79.03	3.87	81.87	0.52	0.38	0.16	0.48
BeamShift	83.05	74.03	9.02	81.11	2.21	-6.65	8.08	0.19
Bladder	82.74	81.54	1.20	82.24	0.84	-0.62	1.45	0.22
CTV	83.05	81.68	1.37	82.31	0.91	-0.76	1.87	0.25
Nominal	82.30	78.63	3.67	80.69	0.01	0.00	0.01	0.00
Rectum	81.07	89.23	11.94	79.23	3.24	-11.85	15.19	-0.48
Rotation	83.04	78.69	4.35	81.88	1.28	-0.03	1.31	0.48

Showing 1 to 10 of 10 entries Previous Next

## D95 Summary Statistics

Show 10 entries Search

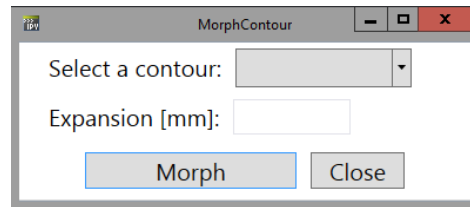
Plan / Structure	Max (D95[Gy])	Min (D95[Gy])	Range (D95[Gy])	Avg (D95[Gy])	Max (D95[%])	Min (D95[%])	Range (D95[%])	Avg (D95[%])
-0.00m Shift	78.89	78.23	0.66	78.41	0.16	-0.30	0.46	-0.07
-1.00% Range	78.18	78.18	0.00	78.18	-0.39	-0.39	0.00	-0.39
0.00m Shift	79.40	78.38	1.02	78.88	-0.08	-0.13	0.28	-0.11
1.00% Range	78.81	78.81	0.00	78.81	0.18	0.18	0.00	0.18
BeamShift	79.75	78.69	1.07	79.03	1.84	0.27	1.37	0.71
CTV	79.75	78.18	1.59	78.62	1.64	-0.39	2.09	0.20
Nominal	78.47	78.47	0.00	78.47	0.01	0.01	0.00	0.01
Rotation	78.73	78.47	0.26	78.59	0.34	0.01	0.33	0.18

Showing 1 to 8 of 8 entries Previous Next

Robust report:

- Tables summarize change in D95% for targets
- Tables summarize change in Dmax for OARs

## Efficiency



```
10 def Morph(self):
11     patient = self.patient
12     case = self.case
13     exptext = self.exp_text
14     if float(exptext) < 1:
15         expansion = float(exptext)
16     else:
17         expansion = float(exptext)/10
18     exam = self.current.Examination
19     ROI = self.selectContour.SelectedItem
20     self.DialogResult = False
21
22     with CompositionAction("Expand (ROI, Image set: exam)"):
23         case.PatientModel.RegionsOfInterest[ROI].CreateMarginGeometry(Examination=exam, SourceROIName=ROI, MarginSettings=[{"Type": "Expand", "Superior": expansion, "Inferior": expansion}])
24
25     with CompositionAction("Contract (ROI, Image set: exam)"):
26         case.PatientModel.RegionsOfInterest[ROI].CreateMarginGeometry(Examination=exam, SourceROIName=ROI, MarginSettings=[{"Type": "Contract", "Superior": expansion, "Inferior": expansion}])
```

22

Not every script has to be huge or involved

Sometimes the simplest scripts can be big time-savers

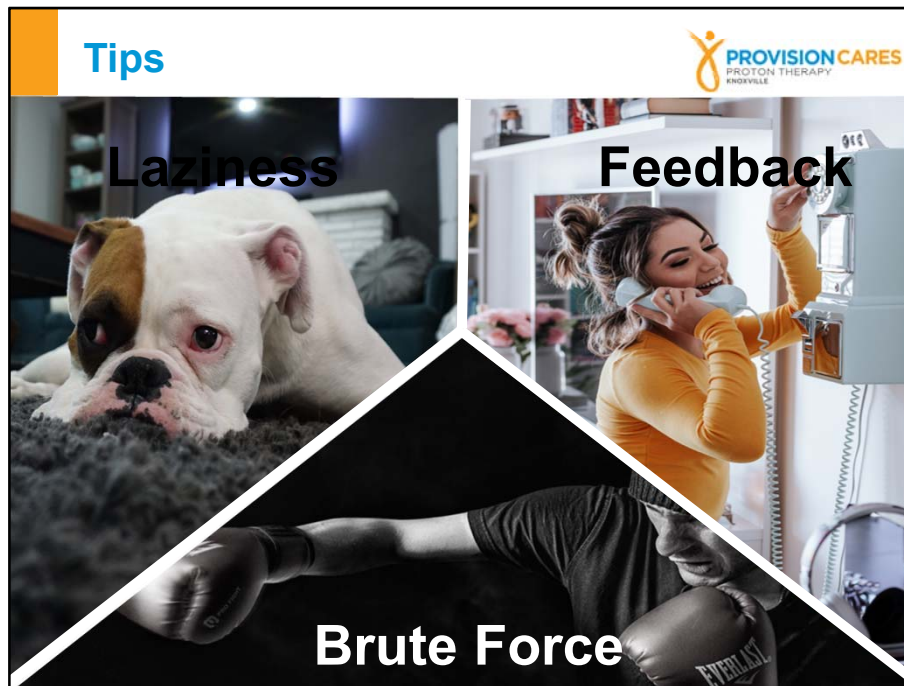
This script is barely more than 2 lines of actual work, but it is used nearly every day

# Efficiency



The image shows a Python script on the left and a QtOARs GUI window on the right. The script is a Python class that implements a target selection algorithm. It iterates through a list of regions of interest (ROIs) and selects those that are not already selected. The GUI window, titled 'QtOARs', has a 'Select a target:' dropdown menu, an 'Expansion [mm]:' input field, and a 'Select Contours to Evaluate' section with checkboxes for Bladder, Rectum, FemurR, FemurL, Sigmoid, PenileBulb, Urethrapartial, and BowelSm. At the bottom of the GUI are buttons for 'Select All', 'Deselect All', 'Go', and 'Close'.

This fairly simple script saves a lot of time and button clicks  
Simple to write, nearly fully recordable



**Laziness:**

- I find a lot of script ideas doing processes I don't normally do, e.g. creating a prostate plan
- It's a lot of clicks that I just don't want to do, so I automate it
- When writing up a plan, I don't want to remember all the required isodoses, contours, etc, so I automate it
- Dosimetrists aren't necessarily going to give you ideas to script them out of a job
- Laziness = improved efficiency

**Brute Force:**

- When I'm testing lines of code in the Console, I use the brute force method
- I plan out what I want the script to do, but not necessarily how I'm going to do
- It's not elegant, but it's effective

**Feedback:**

- I once watched a dosimetrist use the automate prostate plan and then systematically delete all the objectives the script just loaded
- Encourage users to embrace standardization, but understand why they're not using the standards
- Get constant feedback to find bugs and improve



## Lessons Learned



**Simple is better than complex**



**Power of automation**



**Embrace laziness**



**Get feedback**

25

Simple is better than complex  
Power of automation  
Embrace laziness  
Get feedback



 **PROVISIONCARES**  
PROTON THERAPY  
KNOXVILLE

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