Creating Web Presentations

with W3C Slidy

Gerald Senarclens de Grancy <gerald@senarclens.eu>

Grazer Linuxtage, April 2014, Graz

Outline

- Web Presentations - Programs, Purpose and Powers
- Tiny Tour of What Can Be Done
- How to Get Started
- Summary

Programs

- Opera Show
- Google Docs
- Prezi
- W3C Slidy
- S3
- ...
- List of web-based slideshows

Purpose

Create multimedia presentations

- Available to a wide audience
- No "non-standard" software required
- Supported a very wide range of hardware platforms

Powers

- You can do anything you could do on a website
  - *HTML 5*
  - *CSS 3*
  - *JavaScript*
  - *...*
- Allows for real version control
- Full control and flexibility of everything the web has to offer
- At the cost of requiring necessary skills

Limitations

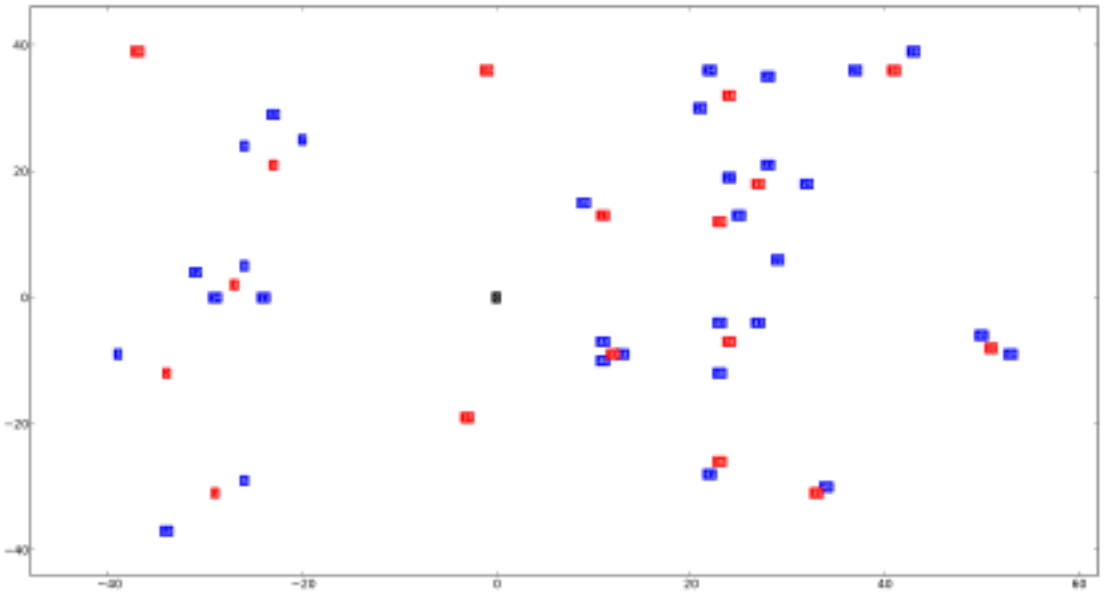
- IMHO a presentation tool does not replace a speaker
  - *For that case, use a regular website or create a video tutorial*

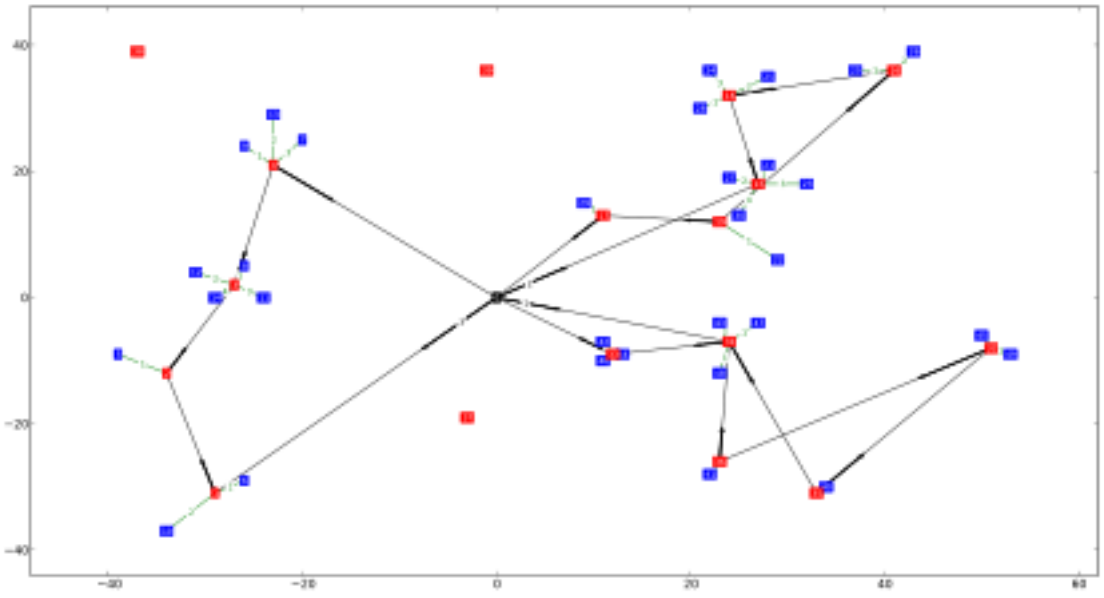
Various features

We can...

- Use bigger and smaller font and navigate the contents
- Show details only if required
  - *Use incremental display*
  - *Additional information*
- Provide print stylesheets for handouts
- Include (incremental images) and display videos
- Take advantage of HTML5 features like canvas
- Seamlessly link to external content
- Use the power of scripting languages like PHP or Python
- ...


I am a fork()


40
20
0
−20
−40
−60
−20
0
20
40
60


40
20
0
−20
−40
−20
0
20
40

Mathematical Formulas

- Use Latex syntax for simple mathematical typesetting

  Parameters

  N

  set of customers ($n \in N$)

  P_0

  set of parking sites including the depot ($p \in P_0$)

  M

  set of customers and parking sites ($m \in M$), $M = N \bigcup P_0$

  C

  set of clusters ($c \in C$), $|C|$ = $|N|$

  L

  set of service workers in any cluster ($l \in L$)

  Q

  sufficiently large number to exclude certain combinations of customers, parking sites and service workers

  wt_{pn}

  walking time from parking site p to customer n

  α_{pn}

  earliest starting time of a customer minus wt_{pn}

  β_{pn}

  latest starting time of a customer minus wt_{pn}

  st_{np}

  service time of customer n plus return time to parking site p

  d_n

  demand of customer n

cap

truck capacity

$\eta \in \{1, 0\}$

parameter controlling emphasis on minimizing the number of clusters versus the makespan

... as well as for more complex stuff

Decision Variables

$$x_{cp} = \begin{cases} 1, \text{if cluster } c \text{ uses parking site } p \\ 0, \text{otherwise} \end{cases}$$

$$y_{cpn} = \begin{cases} 1, \text{if customer } n \text{ is in cluster } c \text{ with parking site } p \\ 0, \text{otherwise} \end{cases}$$

$$v_{nn'}^{lc} = \begin{cases} 1, \text{if service worker } l \text{ walks from } n \text{ to } n' \text{ in cluster } c \\ 0, \text{otherwise} \end{cases}$$

δ_n

actual starting time at a customer n which is a real number with a lower bound of zero

$\overline{\xi_c}$

actual finishing time of cluster c (time when a truck leaves) which is a real number with a lower bound of zero

$\underline{\xi_c}$

actual starting time of cluster c (time when truck arrives) which is a real number with a lower bound of zero

$$\min z = \eta \sum_{c \in C} (\overline{\xi_c} - \underline{\xi_c}) + (1 - \eta) \sum_{c \in C} \sum_{p \in P0} x_{cp}$$

s.t.

$$\overline{\xi_c} - \underline{\xi_c} \geq 0$$

$$\sum_{p \in P_0} x_{cp} \leq 1, \forall c \in C$$

$$\sum_{n \in N} y_{cpn} \cdot d_n \leq \text{cap} \cdot x_{cp}, \forall p \in P_0, \forall c \in C$$

$$\sum_{c \in C} \sum_{p \in P0} y_{cpn} = 1, \forall n \in N$$

$$y_{cpn} \leq x_{cp}, \forall n \in N, p \in P_0, \forall c \in C$$

$$x_{cp} \leq \sum_{n \in N} y_{cpn}, \forall p \in P0, \forall c \in C$$

$$\sum_{l \in L} \sum_{\substack{n' \in M \\ n' \neq n}} v_{nn'}^{lc} \leq \sum_{p \in P_0} y_{cpn}, \forall c \in C, \forall n \in N$$

$$\sum_{n'' \in M} v_{n''n}^{lc} = \sum_{n' \in M} v_{nn'}^{lc}, \forall n \in N, \forall c \in C, \forall l \in L$$

$$\sum_{c \in C} \sum_{l \in L} \sum_{\substack{n' \in M \\ n' \neq n}} v_{nn'}^{lc} = 1, \forall n \in N$$

$$\delta_n \geq \alpha_{pn} \cdot y_{cpn} - Q \cdot (1 - y_{cpn}), \forall n \in N, \forall p \in P0, \forall c \in C$$

$$\delta_n \leq \beta_{pn} \cdot y_{cpn} + Q \cdot (1 - y_{cpn}), \forall n \in N, \forall p \in P0, \forall c \in C$$

$$' + st_{pn'} - Q \cdot (1 - v_{n'n}^{lc}) - Q \cdot (1 - y_{cpn}), \forall n \in N, \forall p \in P0, \forall l \in L, \forall n' \in M, \text{`}$$

$$\underline{\xi_c} \leq \delta_n + Q \cdot (1 - y_{cpn}), \forall n \in N, p \in P_0, \forall c \in C$$

$$\overline{\xi_c} \geq \delta_n + (wt_{pn} + st_{pn}) \cdot y_{cpn} - Q \cdot (1 - y_{cpn}), \forall n \in N, p \in P_0, \forall c \in C$$

Selected Bibliography

VRPTW

Bräysy, Olli and Gendreau, Michel
Vehicle Routing Problem with Time Windows, Part I: Route Construction and Local Search Algorithms
Transportation Science vol. 39, iss. 1, pp 104-118, INFORMS (2005)

Goel, Asvin and Gruhn, Volker
Solving a dynamic real-life vehicle routing problem
Operations Research Proceedings 2005, pp 367–372, Springer (2006)

Pisinger, David and Ropke, Stefan
Large neighborhood search
Handbook of Metaheuristics, pp 399-419, Springer (2010)

Scheduling

Lenstra, J. K.; Shmoys, D. B. and Tardos, É.
Approximation algorithms for scheduling unrelated parallel machines
Mathematical Programming 46 (1990)

Pinedo, Michael L.
Scheduling – Theory, Algorithms, and Systems
Springer (2012)

Pinedo, Michael L.
Planning and Scheduling in Manufacturing and Services
Springer (2012)

External Content

Using prettify for source code

```
def name(arg1, arg2, ..., argN):
    """
    Textual explanation of what the function is supposed to do.
    """
    # calculate result...
    return result
```

Defining and using a function

```
#!/usr/bin/env python3

def interest(capital, rate, years=1):
    """
    Return the total interest of investing `capital` at `rate`.
    """
    value = capital * (1 + rate) ** years
    return value - capital

print(interest(100, 0.03))
result = interest(100, 0.03, 3)
profit = interest(1300, 0.025, 5)
```

Video tutorial

More External Content

- Python Anywhere

(x-3)*(x-5)*(x-7)+85

2

9

Plot

Resources

- Read Dave Raggett's presentation about slidy
- Have a look at this presentation's source
- Check out the video tutorial
- Configure your text editor
- Download a template
- Go wild :)

Summary

Summary

- Slidy allows you to do anything you could do on a website
- But requires at least basic HTML skills

Remains to be done

- Try it out! :)
- Improve it if needed

Acknowledgements

- The W3C, the good browser manufacturers and everybody else involved in making the web awesome
- Blender Foundation | www.blender.org for Big Buck BUNNY

The following list contains all external links in order of appearance in the presentation

- Gerald Senarclens de Grancy: http://www.senarclens.eu/~gerald/
- List of web-based slideshows: http://en.wikipedia.org/wiki/Web-based_slideshow#List_of_web-based_slide_shows
- Defining and using a function: http://pythontutor.com/iframe-embed.html#code=%23%21%2Fusr%2Fbin%2Fenv+python3%0A%0Adef+interest%28capital%2C+rate%2C+years%3D1%29%3A%0A++++%22%22%22%0A++++Return+the+total+interest+of+investing+%60capital%60+at+%60rate%60.%0A++++%22%22%22%0A++++value+%3D+capital+%2A+%281+%2B+rate%29+%2A%2A+years%0A++++return+value+-+capital%0A%0Aprint%28interest%28100%2C+0.03%29%29%0Aresult+%3D+interest%28100%2C+0.03%2C+3%29%0Aprofit+%3D+interest%281300%2C+0.025%2C+5%29%0A&cumulative=false&py=3&curInstr=0
- Video tutorial: http://www.youtube.com/watch?v=TFcUGSBTA78
- Python Anywhere: http://www.pythonanywhere.com/embedded3/
- presentation about slidy: http://www.w3.org/Talks/Tools/Slidy2/