Gramáticas de forma para transformação urbana

Seminário | 21 Abril 2014
10h00 | Gramáticas de Forma e Gramáticas de Transformação
Sara Eloy

10h20 | A Ferramenta shell para Gramáticas de Forma GSG
Joaquim Reis

10h40 | A Interface para o Sistema GSG
Paulo Canilho

11h00 | Sistemas de Informação Geográfica e City Engine
Rui Ricardo e Rúben Reis

11h20 | Desenvolvimento de Apps para arquitetura
Pedro Faria Lopes

11h40 | Debate

Organização: Miguel Sales Dias, Sara Eloy, Joaquim Reis
Autoria da imagem de capa: José Ferrão
Edição: Ana Moural
Gramáticas de Forma e Gramáticas de Transformação

Sara Eloy
Shape grammar uses rules that applied step-by-step to existing shapes will generate a language of designs or describe an existing one.

\[ A \rightarrow B \]

Vocabulary (shapes)

Spatial relations

Shape rule

New Design = \([\text{Design} \cdot f(A)] + f(B)\)

Spatial transformations

Boolean operations
Procedure for defining the entasis of a column
(Palladio)

The columns in each order ought to be form’d in such a manner, that the diameter of the upper part of the column may be greater than at the bottom, with a kind of a swelling in the middle.

As to the manner of making the swelling in the middle, we have no more to learn from Vitruvius but his base promenade, which is the reason that most writers differ from one another upon that subject.

The method I use in making the profile of the swellings is this: I divide the first part of the column into three parts, and leave the lower part perpendicular; to the side of the extremity of which I apply the edge of a thin rule, of the same length, or a little longer than the column, and bend that part which reaches from the third part upwards, until it touches the point of the diminution of the upper part of the column under the collar. I then make as the curve directs, which gives the column a kind of swelling in the middle, and makes it project very gracefully.

And although I never could imagine a more expeditious and successful method than this, I am nevertheless confirmed in my opinion, since Signor Piero Catenio was so well pleased when I told him of it, that he gave it a place in his Treatise of Architecture, with which he has not a little illustrated this profile.

A. B. the third part of the column, which is left directly perpendicular.
B C. the two thirds that are drawn. And,
C. the point of diminution under the collar.

Shape rules

(initial stage)  (final stage)

RULE

Derivation

1  2  3  4

EMERGENCE
seeing without memory
Corpus of solutions

SHAPE GRAMMARS CREATES / DEFINES A LANGUAGE OF DESIGN

Parametric rules

\[ X \rightarrow X + b \{ X \} \]
\[ (x_c, y_c) \rightarrow (x_c, y_c) + (x_c, y_c) + (x_c, y_c) \]
\[ (x_c, y_c) \rightarrow (x_c, y_c) + (x_c+b, y_c) + (x_c+b, y_c+b) \]

\[ X \rightarrow b \{ X \} + n \cdot b \{ X \} \]
\[ (x_c, y_c) \rightarrow (x_c, y_c) + (x_c, y_c) + (x_c, y_c) + (x_c, y_c) \]
\[ (x_c, y_c) \rightarrow (x_c, y_c) + (x_c+b, y_c) + (x_c+b, y_c+b) \]
Analytical grammars
make it possible to understand existing languages

Original grammars
enable new design languages to be created

Analytical grammars
make it possible to understand existing languages

Original grammars
enable new design languages to be created

ice ray, chinese lattice design, George Stiny
palladian villas grammar, George Stiny and William Mitchell
frank lloyd wright’s prairie houses grammar, Koning and Eisenberg
chinese architecture grammar, Andrew Li
gramática da malagueira, José Pinto Duarte
digital thonet, Mário Barros
ice rays, chinese lattice designs _ George Stiny

differentiation of an ice-ray design

palladian villas grammar _ George Stiny and William Mitchell

real or fake?
malagueira grammar  _ José Pinto Duarte

digital thonet  _ Mário Barros
Analytical grammars
make it possible to understand existing languages

Original grammars
enable new design languages to be created

- color grammar, Terry Knight
- city maker, José Beirão
- "rabo-de-bacalhau" transformation grammar, Sara Eloy

color grammar _ Terry Knight

spatial relation

additive rule with 256 different patterns
Analytical grammars
make it possible to understand existing languages

Original grammars
enable new design languages to be created

Grammar interpreters

Gedit  Tapia 1996
Analytical grammars
make it possible to understand existing languages

Original grammars
enable new design languages to be created
+ color grammar, Terry Knight
+ city maker, José Beirão
+ "rabo-de-bacalhau" transformation grammar, Sara Eloy

rehabilitation methodology

knowledge data base
existing dwelling description
functional programme for the dwelling
pack of ICAT functions
adapted dwelling description
ICAT elements

family

ICAT

functional housing requirements

domestic groups

rehabilitation methodology
(specific) transformation grammar

- Assignment rules
- Permuting rules
- Rules to divide rooms or eliminate/reduce wall openings (adding walls)
- Rules to connect or enlarge rooms (eliminating walls)
- Rules for changing the stage in the derivation
- Rules for preparing the floor plan
- Rules for integrating ICAT elements

Sample transformation
#1

-1 preparing the floor plan

0 1 2 3 4 5 6 7 8

Rules -1.1
Creation of a compound representation

#2

-1 preparing the floor plan

0 1 2 3 4 5 6 7 8

Rules -1.2
Creation of a compound representation
Rules 0.1
Assignment of isolated kitchen for strategy 2

Rules 1.1
Assignment of hall
#7
Rules 2.1b
Assignment of double bedroom

#8
Rules 2.3b
Assignment of single bedroom
Rules 2.6
Assignment of main private bathroom

#9

-1
0
1 defining private area
2
3
4
5
6
7
8

Levels

spaces

sample transformation

Rules 2.7a
Attribution of new bathroom placement weight (type a and b)

#10

-1
0
1 defining private area
2
3
4
5
6
7
8

Levels

spaces

sample transformation
#11

Rules 2.8b
Assignment of second private bathroom

#12

Rules 7.4b
Changing a room’s dimension (enlarging or reducing) by moving a wall
Rules 2.16a
Erasing the new bathroom placement weight (type a and b)

Rules 2.5
Permuting bedroom assignment due to area criteria
Rules 3.1a
Assignment of living room

Rules 3.2b
Assignment of dining room

#15
-1 0 1 2 3 defining social area 4 5 6 7 8

#16
-1 0 1 2 3 defining social area 4 5 6 7 8
#17

Rules 3.4
Assignment of isolated home office

#18

Rules 3.11
Assignment of guest bathroom
Rules 7.6d
Changing a door position (change door to a perpendicular wall)

Rules 4.1
Assignment of private corridors
Rules 4.2
Assignment of corridors

Rules 7.3b
Remove part of a room area to assign to circulation area
Rules 5.1
Assignment of isolated laundry

Rules 7.6
Change door position (change door to a perpendicular wall)
**#25**

Rules 7.3a
Remove part of a room to assign to circulation area

1. 1
2. 0
3. 1
4. 2
5. 3
6. 4
7. adapt shape
8. 5

---

**#26**

Rules 4.2
Assignment of social corridor

1. 1
2. 0
3. 1
4. defining circulation
5. 2
6. 3
7. 4
8. 5

---
Rules 7.1i
Widening the connection between two rooms (by partially eliminating walls on both sides of a door opening)

Sample transformation
Rules 8.1a
Allocating of water detectors

Rules 8.1b
Allocating of smoke detectors

Integration of ICAT
### Sample Transformation

#### #30

- **Rules 8.1c**
  - Allocating of temperature detectors

#### #31

- **Rules 8.1d**
  - Allocating of gas detectors

8 Integration of ICAT
#32

Rules 8.1d
Allocating of movement detectors (for the hall and corridors with door to the exterior)

1
2
3
4
5
6
7
8 Integration of ICAT

#33

Rules 8.10a1
Allocating of interfaces: control panel

1
2
3
4
5
6
7
8 Integration of ICAT
Shape Grammar | Advantages

- Optimization
- Systemization
- Generative power
- Diversity
- Capture design language
- Structured formulation that allows computer implementation
- Reduction of design errors
A Ferramenta shell para Gramáticas de Forma GSG

Joaquim Reis
a ferramenta *shell* para gramáticas de forma GSG

**Joaquim Reis**

Dept. Ciências e Tecnologias de Informação
ISCTE-IUL/ADETTI-IUL
Lisboa, Portugal

Joaquim.Reis@iscte.pt

GFTU - Gramáticas de Forma para Transformação Urbana, 21 de Abril de 2014, ADETTI / ISTAR-IUL, ISCTE-IUL

---

**plan**

- shape grammars & examples
- shape emergence & maximal shapes
- computation with shapes

- the GSG project
  - computational architecture
  - shape operations & examples & testing
  - shape representation & main classes
  - shape ioe & interface

- work in progress & future work
shape grammars
(Stiny & Gips 1972) (Stiny 1980)

- Shape Grammars (SG) describe recursive **computations with shapes**
- SGs are **visual**
- Can be used to represent **styles**
- Based on production **rules**

\[
\langle \text{shape-to-match} \rangle \implies \langle \text{substitution-shape} \rangle
\]

elements of a SG:
- **vocabulary** of shapes, e.g., points, lines, 2D, 3D shapes (may include dimensions, colors)
- **rules** of the grammar
- **initial shape**

---

data:image/png;base64,iVBORw0KGgoAAAANSUhEUgAAAAYAAADqCAYAAAD0474zAAAAA3NCSVQICAjb4U/gAAAgAElEQVR42mPbSuQzYAAAAABJRU5ErkJggg==
example

2

define

example

3
example

\[ \triangle \rightarrow \triangle \]

4

example

(with shape emergence ?)

\[ \triangle \rightarrow \triangle \]

1
example
(with shape emergence ?)

```
    △ △
```

2

example
(with shape emergence ?)

```
      △ △

      △ △

      △ △
```

3
example
(with shape emergence ?)

example
(with shape emergence ?)
example
with shape emergence

1

example
with shape emergence

2
example
with shape emergence

example
with shape emergence
example
with shape emergence

example
with shape emergence
example
with shape emergence

example
with shape emergence
example
with shape emergence

1

example
with shape emergence

1

2

3
example

with shape emergence

4

example

with shape emergence

5
shape emergence
(a closer look)

=>

shape emergence
(a closer look)

=>

=>
shape emergence
(a closer look)

\[
\begin{array}{c}
\square \\
\Rightarrow \\
\square \\
\Rightarrow \\
\square \\
\Rightarrow \\
\square \\
\Rightarrow \\
\square
\end{array}
\]

shape emergence
(a closer look)

\[
\begin{array}{c}
\square \\
\Rightarrow \\
\square \\
\Rightarrow \\
\square \\
\Rightarrow \\
\square \\
\Rightarrow \\
\square
\end{array}
\]
shape emergence
(a closer look)

recognized as (embedded) lines

shape emergence
(a closer look)
shape emergence
(a closer look)

maximal shape
(Stiny & Gips 1972) (Stiny 1980)

- maximal shape representation - a finite set of shape elements (points, lines, planes, solids, ...) that are maximal in combination with each other (i.e., with no overlap).
  - each particular shape has a unique maximal representation
  - maximal representation accommodates for shape emergence

shape elements are maximal and each represents an infinite number of smaller possible elements contained in them
  - a maximal line represents an infinite set of points and lines
  - a maximal plane represents an infinite set of points, lines and planes
  - a maximal solid represents an infinite set of points, lines, planes and solids
  - ...

- algebras \(U_j\) for maximal shapes define shape operations: \(+\) (shape union), \(-\) (shape difference), \(\cdot\) (shape intersection) and \(\leq\) (sub-shape)
mechanics of
shape computation
with rules

rule: \[ A \implies B \]

applicable to \( C \) if: \[ T(A) \leq C \]

after application \( C' \) is: \[ (C - T(A)) + T(B) \]

C - a design, or composition
T - a geometric transformation (rotation, translation, scale, mirror)

GSG
generic Shape Grammars
(Reis 2011, 2011)

- a tool for building computational systems to support work in design, architecture, art

based on shape grammars

- intelligent rule-based systems
- interfaces
- computational geometry

for past work see papers (Reis 2006-2013)
non-maximal shape algebras $U_{12}$

- non-maximal shape operations
  - $+$ (union, sum)
  - $-$ (difference, subtraction)
  - $\ast$ (intersection, multiplication)
  - $\leq$ (sub-shape, inclusion)

- shapes are sets of lines
- no potential for shape emergence
- examples with shapes with co-linear lines

non-maximal shape operations

example with (colinear) non-maximal lines, union
non-maximal shape operations
example with (colinear) non-maximal lines, union
non-maximal shape operations
example with (collinear) non-maximal lines, union
non-maximal shape operations
example with (colinear) non-maximal lines, union
non-maximal shape operations
example with (colinear) non-maximal lines, union
non-maximal shape operations
example with (colinear) non-maximal lines, union

non-

maximal

shape

operations

eexample with (colinear) non-maximal lines, union

+
non-maximal shape operations
example with (colinear) non-maximal lines, union

non-maximal shape operations
example with (colinear) non-maximal lines, difference
non-maximal shape operations
example with (colinear) non-maximal lines, difference

non-maximal shape operations
example with (colinear) non-maximal lines, difference
non-maximal shape operations
example with (colinear) non-maximal lines, difference
non-maximal shape operations
example with (colinear) non-maximal lines, difference
non-maximal shape operations
example with (colinear) non-maximal lines, difference
non-maximal shape operations
example with (colinear) non-maximal lines, difference
non-maximal shape operations
example with (colinear) non-maximal lines, difference

non-maximal shape operations
example with (colinear) non-maximal lines, intersection
non-maximal shape operations

example with (colinear) non-maximal lines, intersection
non-maximal shape operations
example with (colinear) non-maximal lines, intersection

non-maximal shape operations
example with (colinear) non-maximal lines, intersection
non-maximal shape operations
example with (colinear) non-maximal lines, intersection
non-maximal shape operations
example with (colinear) non-maximal lines, intersection

non-maximal shape operations
example with (colinear) non-maximal lines, intersection
non-maximal shape operations
example with (colinear) non-maximal lines, intersection

*
non-maximal shape operations
example with (colinear) non-maximal lines, intersection

non-maximal shape operations
example with (colinear) non-maximal lines, sub-shape
non-maximal shape operations
example with (colinear) non-maximal lines, sub-shape
non-maximal shape operations
example with (colinear) non-maximal lines, sub-shape
non-maximal shape operations
example with (colinear) non-maximal lines, sub-shape
non-maximal shape operations
example with (colinear) non-maximal lines, sub-shape
non-maximal shape operations

example with (colinear) non-maximal lines, sub-shape
non-maximal shape operations
example with (colinear) non-maximal lines, sub-shape

maximal shape algebras $U_{12}$

- maximal shape operations
  + (union, sum)
  - (difference, subtraction)
  * (intersection, multiplication)
  ≤ (sub-shape, inclusion)

- shapes are sets of maximal lines
- high potential for shape emergence
- examples with shapes with co-linear lines
maximal shape operations
example with (colinear) maximal lines, union

maximal shape operations
example with (colinear) maximal lines, union
maximal shape operations
example with (colinear) maximal lines, union
maximal shape operations
example with (colinear) maximal lines, union

maximal shape operations
example with (colinear) maximal lines, union
maximal shape operations
example with (colinear) maximal lines, union

maximal shape operations
example with (colinear) maximal lines, union
maximal shape operations
example with (colinear) maximal lines, union

maximal shape operations
example with (colinear) maximal lines, union
maximal shape operations
example with (colinear) maximal lines, union

maximal shape operations
example with (colinear) maximal lines, union
maximal shape operations
example with (colinear) maximal lines, difference
maximal shape operations
example with (colinear) maximal lines, difference
maximal shape operations
example with (colinear) maximal lines, difference
maximal shape operations
example with (colinear) maximal lines, difference
maximal shape operations
example with (colinear) maximal lines, difference
maximal shape operations
example with (colinear) maximal lines, difference
maximal shape operations
example with (colinear) maximal lines, intersection

maximal shape operations
example with (colinear) maximal lines, intersection
maximal shape operations
example with (colinear) maximal lines, intersection
maximal shape operations
example with (colinear) maximal lines, intersection
maximal shape operations
example with (colinear) maximal lines, intersection

maximal shape operations
example with (colinear) maximal lines, intersection
maximal shape operations
example with (colinear) maximal lines, intersection
maximal shape operations
example with (colinear) maximal lines, intersection

maximal shape operations
example with (colinear) maximal lines, intersection
maximal shape operations
example with (colinear) maximal lines, sub-shape
maximal shape operations
example with (colinear) maximal lines, sub-shape
maximal shape operations
example with (colinear) maximal lines, sub-shape
maximal shape operations
example with (colinear) maximal lines, sub-shape
maximal shape operations
example with (colinear) maximal lines, sub-shape
maximal shape operations
example with (colinear) maximal lines, sub-shape

maximal shape operations
example with (colinear) maximal lines, sub-shape
(defun sh+ (s1 s2 &optional name (maximal-p nil max-given-p))
  "The + (union) shape function operation."
  : )

(defun sh- (s1 s2 &optional name (maximal-p nil max-given-p))
  "The - (difference) shape function operation."
  : )

(defun sh* (s1 s2 &optional name (maximal-p nil max-given-p))
  "The * (intersection) shape function operation."
  : )

(defun sh<= (s1 s2 &optional (maximal-p nil max-given-p))
  "The <= (sub-shape) shape function operation."
  : )
testing shape operations

a shape in a test file (with 8 horizontal lines at y = 0)

(line :x1 10 :y1 0 :x2 30 :y2 0)
(line :x1 40 :y1 0 :x2 60 :y2 0)
(line :x1 70 :y1 0 :x2 90 :y2 0)
(line :x1 110 :y1 0 :x2 130 :y2 0)
(line :x1 150 :y1 0 :x2 170 :y2 0)
(line :x1 180 :y1 0 :x2 190 :y2 0)
(line :x1 200 :y1 0 :x2 210 :y2 0)
(line :x1 220 :y1 0 :x2 240 :y2 0)

a test report file for ≤

:
:
test-shape-sub-slope0-case003 T
test-shape-sub-slope0-case002 T
test-shape-sub-slope0-case001 T
116 cases: 116 matches, 0 mismatches.

shape representation

interface

programmatic
shape & rule
input/output/edition
SG interpreter

visual
shape & rule
input/output/edition
SG interpreter

symbolic / api

rbs
inference engine
symbolic reasoning
with shapes & spatial relations

facts & rules
shapes & SG rules
spatial relations

geom
shapes & SG rules
shape operations (≥,≤,=)
maximal shape reduction
shape transformations
spatial relations

GFPTU 21 Abr-2014 Joaquim Reis ADETT/ISTAR-IUL, ISCTE-IUL
GSG main classes

a new beast: line intersections
GSG main classes

line intersections

cross  tee  corner  apparent-tee  apparent-corner

cross  tee  corner  apparent-tee  apparent-corner
GSG (obsolete) interface

work in progress

programmatic
shape & rule
input/output/edition
SG interpreter

visual
shape & rule
input/output/edition
SG interpreter

visual

symbolic / api

interface

rbs
inference engine
symbolic reasoning
with shapes & spatial relations

kb
facts & rules
shapes & SG rules
spatial relations

geom
shapes & SG rules
shape operations (...,)
maximal shape reduction
shape transformations
spatial relations
future work

programmatic shape & rule input/output/edition
SG interpreter

visual shape & rule input/output/edition
SG interpreter

interface

symbolic / api

rbs
inference engine
symbolic reasoning with shapes & spatial relations

kb
facts & rules shapes & SG rules
spatial relations

geom
shapes & SG rules
shape operations (\(\subseteq, \supseteq, \equiv\))
maximal shape reduction
shape transformations
spatial relations

---

Joaquim Reis - GSG & Related Shape Grammar Publications


Technical Reports

GSG

a shell tool for shape grammar support

Joaquim Reis

Dept. Ciências e Tecnologias de Informação
ISCTE-IUL/ADETTI-IUL
Lisboa, Portugal

Joaquim.Reis@iscte.pt

GFTU - Gramáticas de Forma para Transformação Urbana,
April 21th, 2014, ADETTI / ISTAR-IUL, ISCTE-IUL
A Interface para o Sistema GSG

Paulo Canilho
Professor Doutor Coordenador de Projecto - Joaquim Reis

Aluno Bolseiro - Paulo Canilho
Introdução

• O que é o GSG?
  • Descrição

• Interface Gráfica
  • Desactualizada
  • Problemas
  • Proposta

GSG

• Descrição
Geometria Computacional

Interface Gráfica (Desactualizada)
GSG - New Line

GSG - Ambiente
GSG - Edit-Line

GSG - Edit-LB
GSG - Tools

Problemas

- Janelas desagrupadas;
- Poucas funcionalidades de edição;
- Falta de apoio na criação de formas;
- e.t.c.

Difícil utilização pelo utilizador!
Proposta

- Eliminar o uso de janelas secundárias;
- Agrupar funcionalidades em painéis dimensionáveis;
- Desenvolver ferramentas de ajuda para a criação de formas;
- Implementar layouts e funcionalidades "User Friendly";
Janela de Apresentação

G.S.G
Generic Shape Grammar

New Project
Name
Save Path

Recent Projects
1. Example1
2. Project015
3. Project21

Manuals
1. Introduction
2. Shapes Grammar
3. Operations

About us
Features

Criação

Menus

Painel

Formas

Modos de funcionamento
Criação - Generic/Line

Criação - Generic/Labelled Point
File - Import/Export

File | Edit | Tools | History | Help
--- | --- | --- | --- | ---
New... | Open... | Grammar |
Save | Save as... | |
Import | Export | Into Project | Into New Project |
Quit

File | Edit | Tools | History | Help
--- | --- | --- | --- | ---
New... | Open... | Grammar |
Save | Save as... | |
Import | Export | All... | Selected only... |
Quit

File - Tools/Preferences

G.S.G
Generic Shape Grammar

Preferences | F2
--- | ---
Profile
Load Profile
Apply Changes
Save Profile
Reset Defaults

LINE

Intersection Angle
- Degrees
- None

Intersection Point Style
- Circle
- Crosshair

Intersection Point Radius

MISCELLANEOUS

Background Color
- Colour
- Black

Cursor Style
- Normal
- Crosshair

Maximal Entities

Point
- Straight Line Dash
- Colour
- Red
Bibliografia

- Reis, Joaquim, Agents with Style – Multi-Agent Visual Composition with Shape Grammars, 2006, Lisboa.


- Chase, Scott C., A model for user interaction in grammar-based design systems, Department of Architectural and Design Science, University of Sydney, Sydney, 2006, Australia.

Sistemas de Informação Geográfica e City Engine

Rui Ricardo, Rúben Reis
http://prezi.com/qt1adw5fi_ru/cityengine/

Ruben

GEOBIM

Rui Ricardo

ISCTE-IUL/CML
Criação de modelos baseados em regras procedimentais

City Engine?
Porque não?

Procura a integração plena com o SIG através da integração com softwares como o Arcgis
Esri maior fabricante mundial de software SIG

Integração plena permite enquadrar este tipo de software de modelação procedimental em ambientes de trabalho empresariais de modo facilitado

- Como Input
  - Base 2D
  - Modelos Geoprocessamento
- Como Output
  - Sistemas de Vistas
  - Simulação
Mais que uma ferramenta de simulação, irá muito provavelmente num futuro próximo, integrar as ferramentas de análise espacial do SIG.
http://prezi.com/qt1adw5fi_ru/cityengine/

Ruben
CITYENGINE
UMA NOVA PERSPECTIVA PARA O PLANEAMENTO URBANO
BY RÚBEN REIS

O QUE FAZ?
Um software para a criação, planeamento e modelação de ambientes urbanos em 3D.
O QUE FAZ?
Gera rapidamente cidades em 3D a partir de dados 2D

Mantém, modifica e actualiza os dados

Mantém, modifica e actualiza os dados
Criação de cenários reais de forma a compará-los e analisá-los

http://video.arcgis.com/watch/2208/3d-urban-analysis

Modela ambientes urbanos em 3D para simulação e entretenimento

Partilha de planos urbanísticos via WEB
Partilha de planos urbanísticos via WEB

COMO FAZ?
case distanceToCenter < radiusApartmentArea : 20%:
    case distanceToCenter < radiusResidentialArea : 5%:
        else :
            attr buildingHeightFactor = 1
            attr nightMode = false // switch between night and day

Lot -->
    translate(rel,world,0,sidewalkHeight,0) LotAligned

LotAligned -->
    case buildingType == 4: YardGround Trees
    case buildingType == 3: ResidentialBlock
    case buildingType == 2: ApartmentBlock
    case buildingType == 1: CommercialBlock
    else :
        HighriseBlock
Compatibilidades

ArcGIS
AutoCAD
Maya
3D studio MAX
Revit
...
CITYENGINE
UMA NOVA PERSPECTIVA PARA O PLANEAMENTO URBANO

by RÚBEN REIS

O QUE FAZ?

Um software para a criação, planeamento e simulação de ambientes urbanos em 3D.
Desenvolvimento de Apps para arquitetura

Pedro Faria Lopes
Developing Apps for Architecture

Pedro Faria Lopes
ISCTE-IUL, ADETTI-IUL, ISTAR-IUL

Econtro "Gramáticas de Forma para Transformação Urbana"
2014-04-11

Index

• Three Apps
  • Defining the nuclear family cell, [(near)illiterate]
    – Emerg.cities4all project
  • Behavior observation by counting
    – Lisbon Pedestrian Network project
    – Close to cities and closer to people project
  • App | Automatic Housing Functional Programme
    – Prof. Sara Eloy’s PhD follow up project

• Lessons learned
• Conclusions
App 1

- Defining the nuclear family cell, [(near)illiterate]
  - Emerg.cities4all project

- Specification
  - P.F. Lopes, S. Eloy, R. Guerreiro

- Students, LEI, 3rd year, 2011-2012
  - Ana Marta Aparício, nº 33570
  - Fábio Tavares, nº 33575
  - Pedro Barros, nº 33605
  - Sara Guerreiro, nº 34391

Creating an interface

- Current paradigm
  - desktop paradigm
    - folders and files present in a virtual desktop
  - drag&drop, scroll, pop-up windows and menus, pull-down menus and drop-down list menus

- For people with low or no digital user expertise these are barriers: procedures to manipulate, select and input hidden information
**YSWYI**

- For users with no digital user expertise:
  - You See What You Input (YSWYI) approach
    - Features
      - No hidden information
      - The basic unit is the full screen
      - Input information visually present all the time
        - Minimal information per basic unit to avoid confusion!
      - Information input: successive screens step by step
      - YSWYI: simplicity and usability in the real world

---

**Input icons**

- The family unit is composed of parents, children, cousins, aunts, nephews, nieces, grandparents and others (friends, friends of friends, may be still relatives or not)
  - Extended family concept
- Possibility of having amputees in the household
  - Example: Mozambique, +20 people injured/killed by land mines each month (2011 report)
Input icons, study, S. Eloy
Input icons, study, S. Eloy

Now & to be done

• Current status of the study
  – Children are also affected by war consequences: child amputee icon?
    • necessary to define the household typology?
• Further steps
  – Icons tests with the target group
  – Creation of a mokup and tests
IPM, group 1

• Next slides: from student’s PPT (SPPT)
  • Not edited material, marked with SPPT
• UC: Interacção Pessoa-Máquina 2011-12
  – Students from LEI, 3rd Year
    • Ana Marta Aparício, nº 33570
    • Fábio Tavares, nº 33575
    • Pedro Barros, nº 33605
    • Sara Guerreiro, nº 34391

Contexto do Projecto

SPPT
P.F. Lopes, J. Reis, F. Santos, S. Eloy, A. Paio, V. Rato,
Shaping emergent cities for all, SIGraDi 2011, XV Congreso de la
Sociedad Iberoamericana de Grafica Digital, 16-18 de Novembro 2011,

• Levantamento da Família Alargada
• You See What You Input (YSWYI)
  ➢ Brasil
  ➢ Angola
  ➢ Moçambique
Forma de desenvolvimento

- Não convencional
  - Loose specs
  - Ensaio / tentativa / erro
  - Testes iterativos
    - E interativos
  - Como testar sem o público alvo final?
  - O que são os ícones representativos para o público alvo final?

Ícones, A. Horta

1ª Versão

Versão Final
Estudo da Iconografia Regional

Testes

Número total:  45 testes + 4 testes preliminares

Categorias

Testes Preliminares: 2 alunos + 2 professores

Testes:

Professores: 7
Alunos: 22
Secretariado: 10
Idosos (> 50 anos): 6
Testes, Análise Qualitativa

SPPT
- Observadores: Notou-se que num dos testes a presença de 4 observadores se transformou num factor de stress significativo;
- Tipos de ícones: Algumas das pessoas, nos ícones da família, estranhavam os ícones “desaparecerem” de ecrã para ecrã – noção de conjunto;
- Uso da barra de navegação para inserir ícones: nos testes preliminares esta funcionalidade foi pedida e mostrou-se muito utilizada nos testes posteriores;
- Mesmo pessoas idosas computorfóbicas/tecnofóbicas interagiam com o dispositivo porque não é um computador.
  – Leitura de que o dispositivo é como um jogo;
- “Já estou a fazer asneira” – não é a pessoa a ser avaliada, mas o sistema.

Dificuldades, aprendizagem

SPPT
- Como testar sem o público alvo
  - Africanos rurais sem escolaridade ou modernidade;
  - Europeus: ou muito velhos ou muito novos.
  – Temas: possibilidade de usar cores, flores, brinquedos, compras, como temas de teste fora do público alvo;
- Que ícone para disabled person com público alvo que não conhece “cadeira de rodas”?
- Sair da zona de conforto: o que é não saber?
Algumas conclusões

- Não é fácil vestir a pele do outro
  - Especialmente quando se trata de “não saber”
  - O que nos parece evidente NÃO é evidente
- Dependendo do background do testador (uso de touch screen – Android, iPhone) mais ou menos imediatismo na interacção
  - Exemplo: aluno de MIA, apagar um elemento, tocar
    - Porquê? “Porque era a única possibilidade!”

Exemplo

SPPT
Vídeo, demo

App 2

- Behavior observation by counting
  - Lisbon Pedestrian Network project
  - Close to cities and closer to people project

- Specification
  - P.F. Lopes, S. Eloy

- Students, LEI, 3rd year, 2012-2013
  - Tiago Martins
  - Marco Menino
  - João Caldas
  - Flávio Freire
Context

- Settings
  - Define a virtual gate
  - Define the items to process
  - Count the items that traverse the virtual gate
  - Directly applicable in Space Syntax contexts
    - User configurable to add any items at will

The App

- Android App
- Developed in Human-Computer Interaction (HCI) Curricular Unit
- Targeted (initially) for Architects, usable by anyone
- Tested with Architects and Architecture students
- Participatory and iterative
  - Specification, design, testing, implementation, testing
Main functions

- **Single Count** and **Distributed Count**
- Automatic reports
- **Aggregate Reports**
- User configurations for later use

Single Count 1/2

- Enables the user to start counting in a very fast way after selecting **Local Name** and **Gate Name**, the **Time** interval and ...
Single Count 2/2

- ... the items to be counted.

Counting layout

- ... or a Custom Pick layout.

- Counting (increase and decrease) has an immediate non visual feedback: vibration, reassuring the user on the correct entry.
Reports

- Each count generates a **Report**.
- Multiple reports can be aggregated into an integrated report.
- Reports are in **CSV** text format (Comma Separated Values)
  - Spreadsheet ready

---

Preset

- A phone call can ruin a count: the **Airplane Mode** avoids calls.
- Broad day light affects readability: the **Application Brightness** control is available.
- A freshly fully charged battery is advised.
Testimonial

- Nádia Romão, MSc
  Architecture student:
  “The app is very useful, easy to use and much better than using a sheet of paper. The reports also eased the work, enabling me to create graphics with no need to introduce the counts one by one”.

Final remarks

- Full battery is advised
- Calls do ruin counting
  - As with pencil/paper
- Sun glare, a problem
  - Choose the right spot
- Vibration feedback very useful: no looking down
- CSV text reports direct use
- 4 months, +5000 code lines
App 3

- App | Automatic Housing Functional Programme
  - Prof. Sara Eloy's PhD follow up project

- Specification
  - S. Eloy, P.F. Lopes

- Students, LEI, 3rd year, 2013-2014
  - David Paiva
  - Filipe Martins
  - João Paulino
  - Rúdi Luis

IPM, group 1

- Next slides: from student’s PPT (SPPT)
  - Not edited material, marked with SPPT
- Week 8 of classes (total of 9 weeks)
- UC: Interacção Pessoa-Máquina 2013-14
Evolução do programa

- Alteração da tabela, para melhor acesso aos dados;
- Finalização dos layouts do programa;
- Realização de pequenos testes.

---

Mini Testes

<table>
<thead>
<tr>
<th>Participante</th>
<th>Idade</th>
<th>Escolaridade</th>
<th>Font Size</th>
<th>Iniciar App</th>
<th>Sobre</th>
<th>Ajuda</th>
<th>Idioma</th>
<th>Smartphone?</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>36</td>
<td>12</td>
<td>+</td>
<td>+</td>
<td>-</td>
<td>+</td>
<td>+</td>
<td>N</td>
</tr>
<tr>
<td>2</td>
<td>47</td>
<td>9</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>N</td>
</tr>
<tr>
<td>3</td>
<td>49</td>
<td>4</td>
<td>+</td>
<td>+</td>
<td>-</td>
<td>+</td>
<td>+</td>
<td>N</td>
</tr>
<tr>
<td>4</td>
<td>33</td>
<td>12</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>5</td>
<td>28</td>
<td>licenciatura</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>S</td>
</tr>
<tr>
<td>6</td>
<td>20</td>
<td>12</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>N</td>
</tr>
<tr>
<td>7</td>
<td>21</td>
<td>12</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>N</td>
</tr>
<tr>
<td>8</td>
<td>22</td>
<td>12</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>S</td>
</tr>
<tr>
<td>9</td>
<td>22</td>
<td>12</td>
<td>+</td>
<td>+</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>N</td>
</tr>
<tr>
<td>10</td>
<td>21</td>
<td>12</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>S</td>
</tr>
<tr>
<td>11</td>
<td>20</td>
<td>12</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>S</td>
</tr>
<tr>
<td>12</td>
<td>32</td>
<td>licenciatura</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>S</td>
</tr>
</tbody>
</table>
Alteração dos layouts

Composição da Família

Filhos

Outros

Filhos

Outros

Vocês
Conjuge
Filho 1
Filho 2
Outro 1

Deslize para continuar

Relações na Família

Idade

00

00

00

00

00

Deslize para continuar
Alteração dos layouts

Restrições
Existe na sua família algum membro que tenha dificuldades motoras?

No    Yes

Sim    Não

Qualidade
Qual o nível de qualidade que deseja adquirir para si?

Minimum   Recommended

Mínimo   Recomendado

Deslize para continuar
IPM1314 Gr01 Week 7-8-9

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>O que fez?</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>David Paiva</td>
<td>Alteração da tabela; Produção do output do programa; Realização de alguns testes.</td>
<td>Continuação da produção do output.</td>
</tr>
<tr>
<td>Filipe Martins</td>
<td>Finalização dos layouts; Alteração da tabela; Realização de alguns testes.</td>
<td>Elaboração do guião de testes.</td>
</tr>
<tr>
<td>João Paulino</td>
<td>Alteração da tabela; Realização de alguns testes.</td>
<td>Elaboração do guião de testes.</td>
</tr>
<tr>
<td>Rudi Luís</td>
<td>Alteração da tabela; Produção do output do programa; Realização de alguns testes.</td>
<td>Continuação da produção do output.</td>
</tr>
</tbody>
</table>

Lessons learned

- LEI student mostly follow their heart
- Even when the discourse is/seems coherent …
  - Thoughts and sentences apparently user centered
- … the practice shows otherwise
  - Students tests practice is usually unintentionally biased
  - The corridor test
    - Informal tests with anyone passing in the corridor
    - Confirms biased previous students tests
- It’s a Maturity issue
- App 1, Defining the nuclear family cell, [(near)illiterate]
  - Sara Guerreiro, LEI student but Psychologist
  - It affected the project result in a huge positive way
Conclusions

• Development for the target user, changed to
• Development for and with the target user
  – Extensive, regular and iterative tests
• Major difficulties
  – 3rd year LEI students tend to be
    » Engineering centered, programming centered
    instead of
    » Target user centered
  – Good results depend heavily on close monitoring of
    » Specification
    » Weakly work done
    » Tests results with users