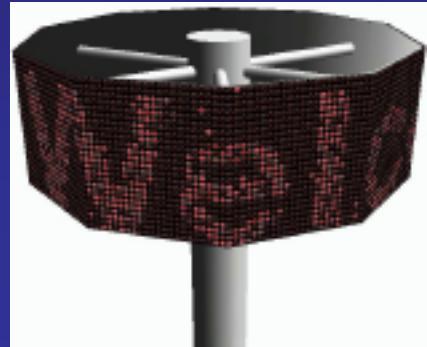
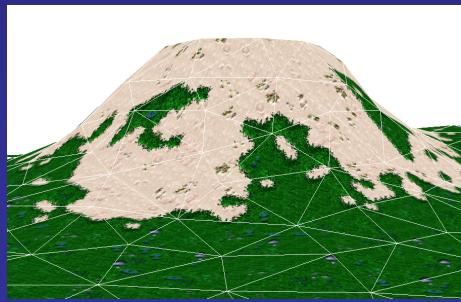
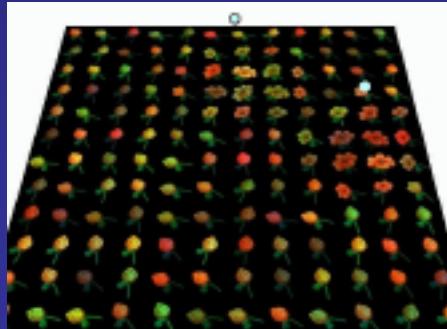


Pattern Based Procedural Textures

Sylvain Lefebvre

Fabrice Neyret

iMAGIS - GRAVIR / IMAG - INRIA



<http://www-imagis.imag.fr/Membres/Sylvain.Lefebvre/pattern>

Overview

- Motivations
- Previous Work
- Contributions
- Our Framework
 - Case study
 - Results
- Conclusion

Motivations

- Texturing large areas
 - Landscapes in simulators
 - Video games



- Requirements

Low memory cost and high resolution

Avoiding periodicity

User control

Previous work

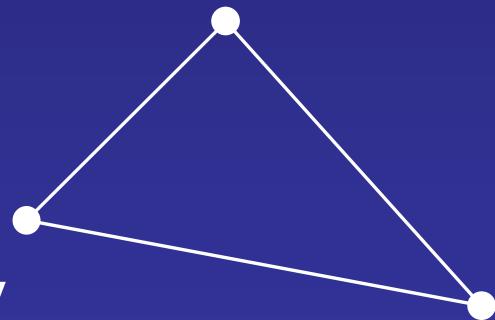
- Large explicit texture
- Extensions
 - Clipmaps [Tanner et al. 98]
 - Texture Compression
 - Empty space compression [Krauss Ertl 02]
- Drawbacks
 - Memory cost
 - Lossy Compression

Previous work

- Procedural textures
 - [Perlin 85] [Worley 96] [Ebert 94]
- Drawbacks
 - Calculation cost
 - Control by the artist not trivial
 - Not all materials

Previous work

- Pattern based texturing
 - Tiling
 - Aperiodic tiling [Stam 97]
 - Triangular patterns [Neyret Cani 99]
 - Virtual atlases [Soler 02]
 - Sparse convolutions [Lewis 89, Erbert 94]

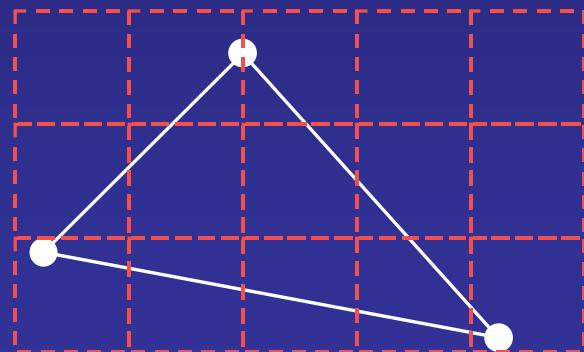


- Drawbacks
 - Mesh dependency
 - Local variations not easy

Previous work

- Pattern based texturing
 - Tiling
 - Aperiodic tiling [Stam 97]
 - Triangular patterns [Neyret Cani 99]
 - Virtual atlases [Soler 02]
 - Sparse convolutions [Lewis 89, Erbert 94]

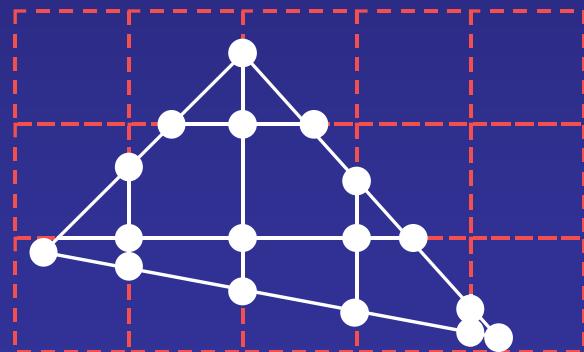
- Drawbacks
 - Mesh dependency
 - Local variations



Previous work

- Pattern based texturing
 - Tiling
 - Aperiodic tiling [Stam 97]
 - Triangular patterns [Neyret Cani 99]
 - Virtual atlases [Soler 02]
 - Sparse convolutions [Lewis 89, Erbert 94]

- Drawbacks
 - Mesh dependency
 - Local variations

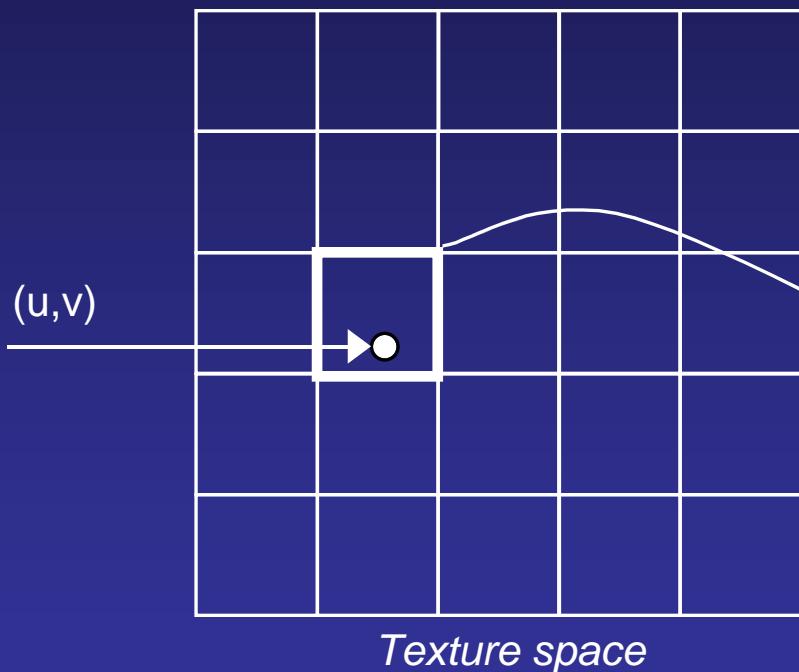


Contributions

- Framework for creating large textures by combining patterns
 - No constraint on mesh (texture space)
 - Low memory cost
- Runs on today graphics hardware
- As generic as possible

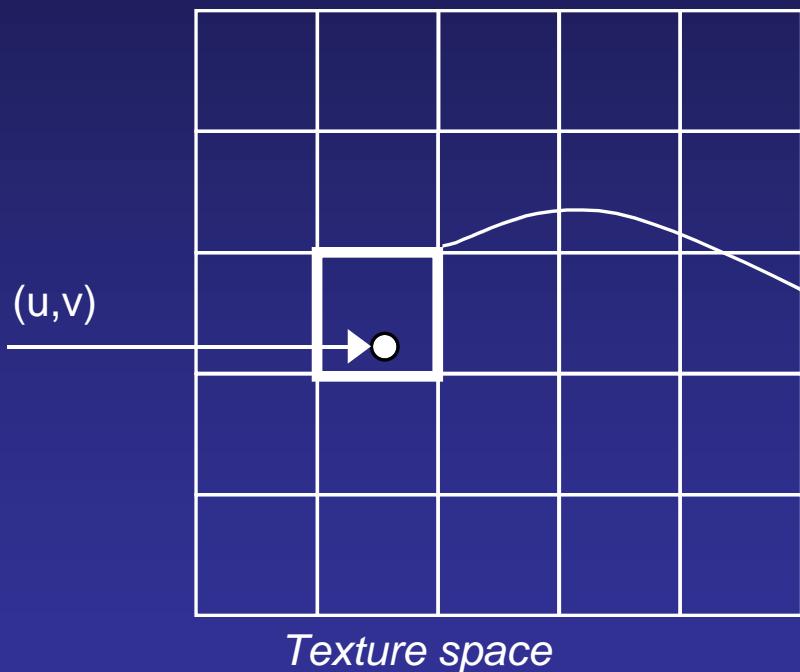
Method overview

Texture coordinates pattern



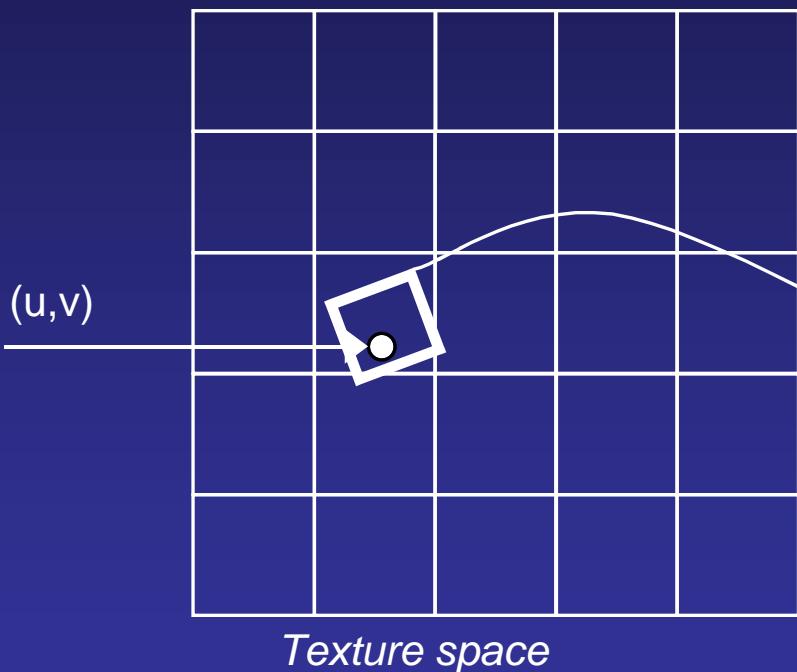
Method overview

Texture coordinates pattern color



Method overview

Texture coordinates



pattern



color

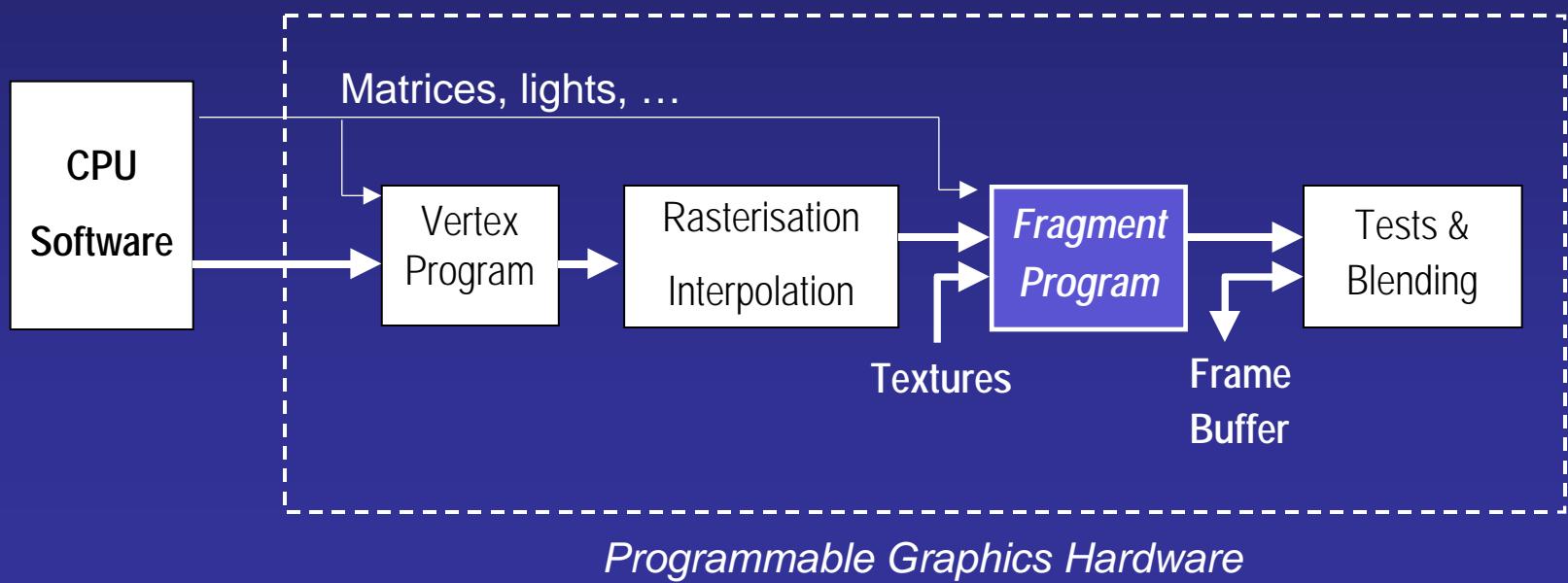
color →

Method overview

Texture coordinates

color

Corresponds to hardware *fragment program*



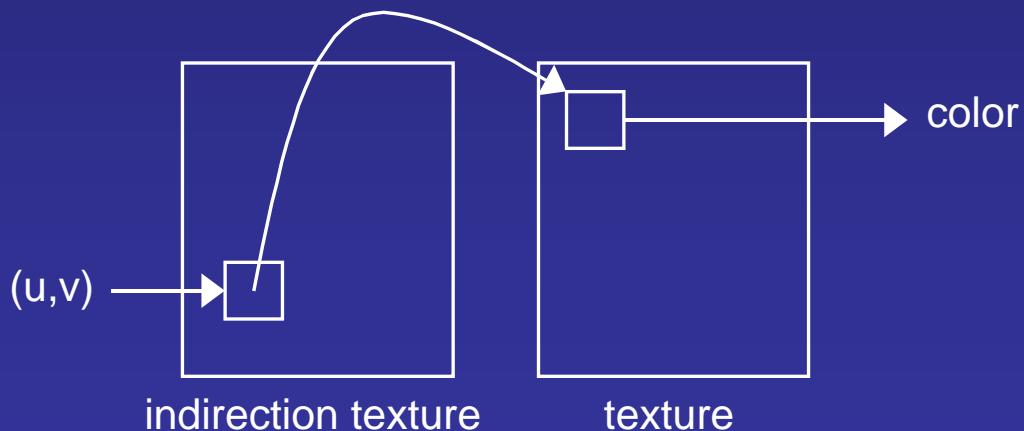
Method overview

Texture coordinates

color

Corresponds to hardware *fragment program*

Relies on *indirection textures* (like [Krauss Ertl 02])



Our Framework

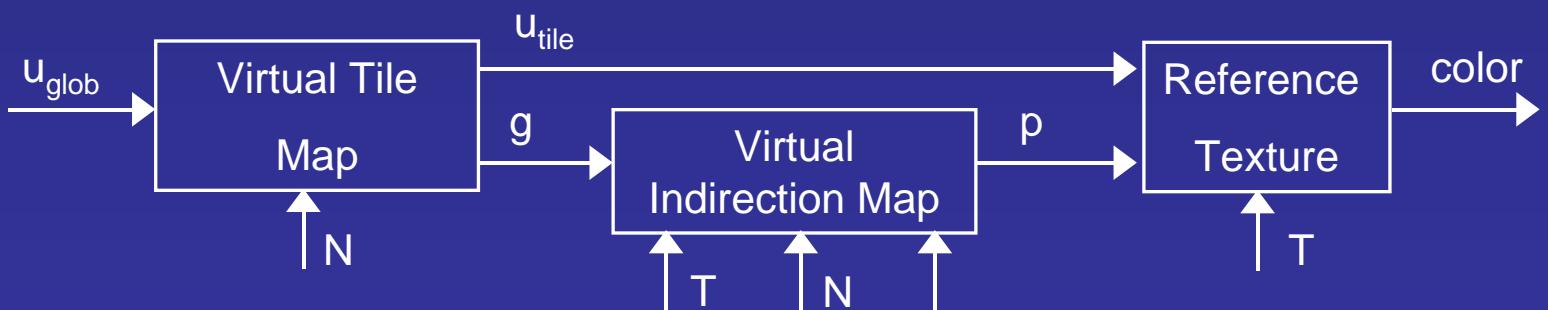
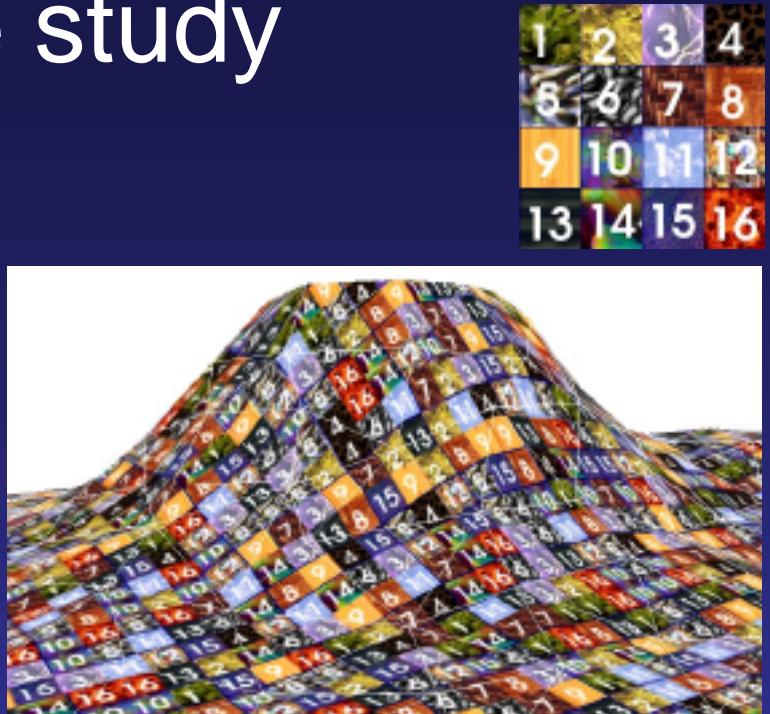
- Set of basic blocks
- 1 block = 1 functionality
- Textures by combining blocks

Overview

- Motivations
- Previous Work
- Contributions
- Our Framework
 - Case study
 - Results
- Conclusion

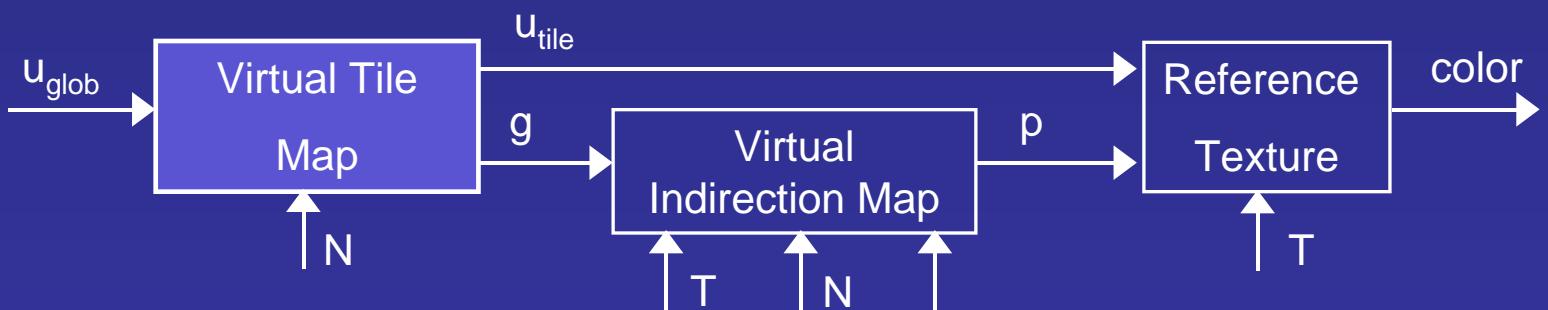
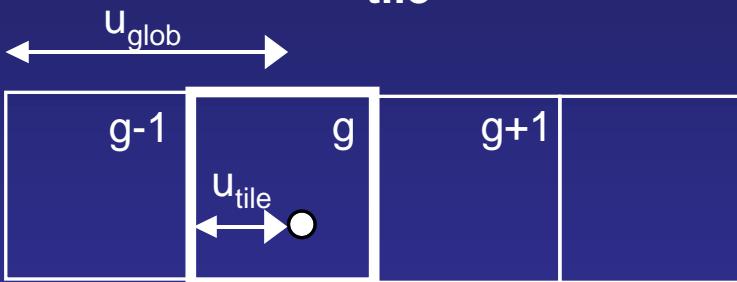
Case study

- Aperiodic tiling
 - $N \times N$ virtual cells
 - $T \times T$ patterns
 - 3 blocks



Aperiodic tiling

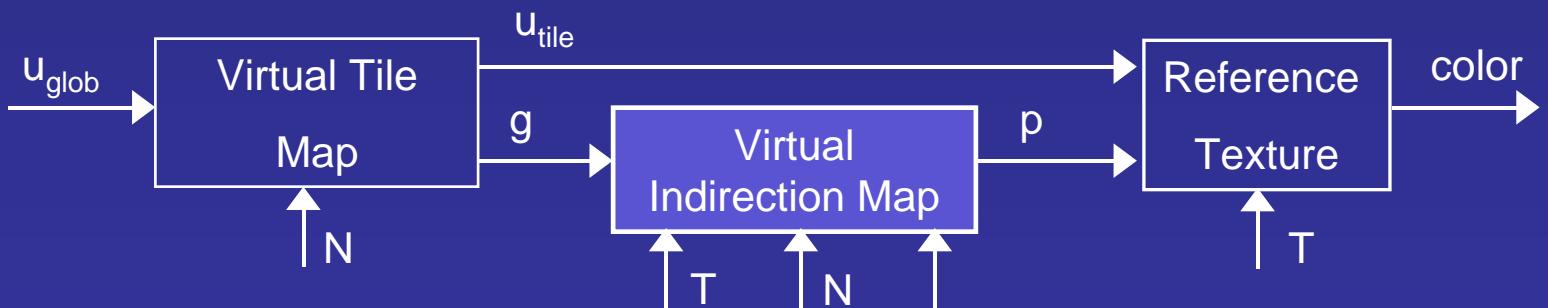
- Virtual Tile Map
 - cell index g
 - relative coordinates \mathbf{u}_{tile}



Aperiodic tiling

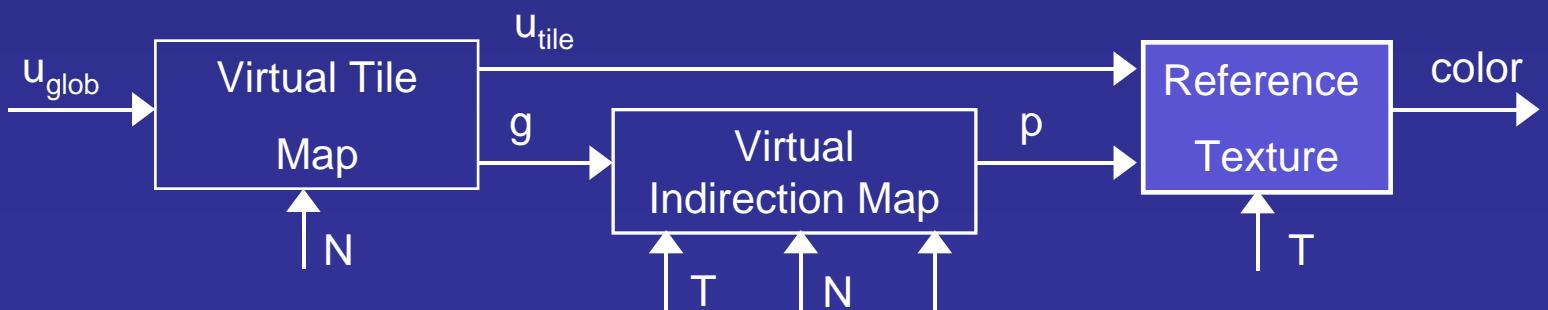
- Virtual Indirection Map
 - random index p from g (aperiodic)
 - uses permutation table σ

$$p = \sigma\left(\frac{g}{T^2} + \sigma\left(\frac{g}{T} + \sigma(g)\right)\right)$$

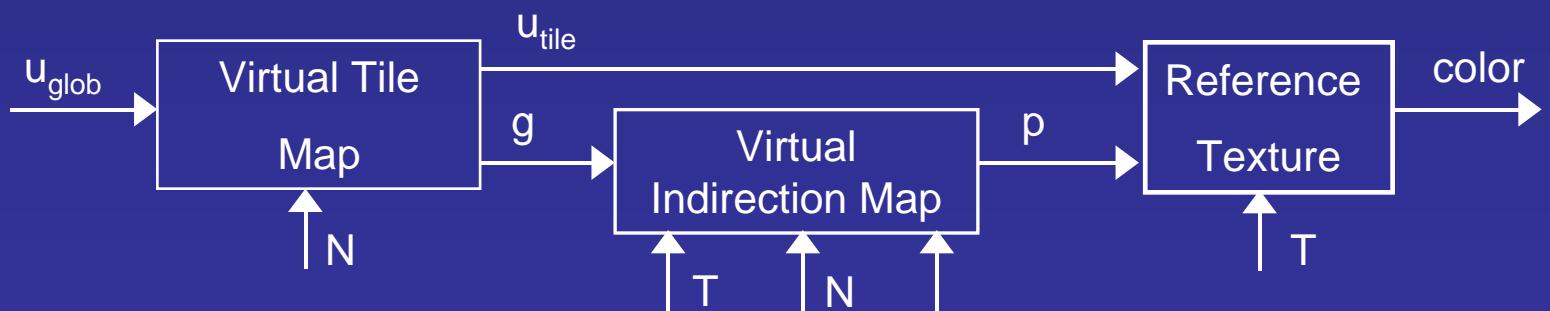


Aperiodic tiling

- Reference Texture

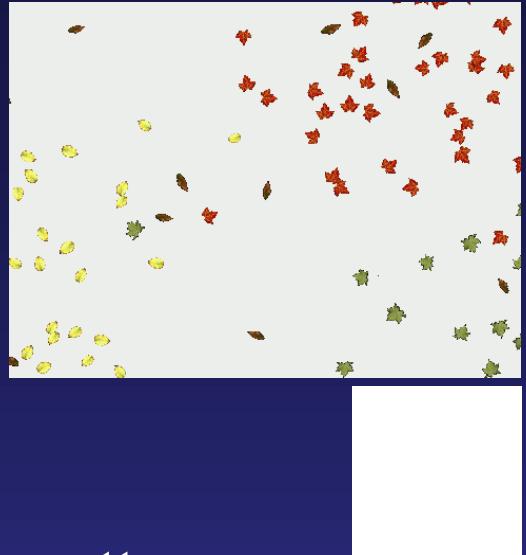


Aperiodic tiling



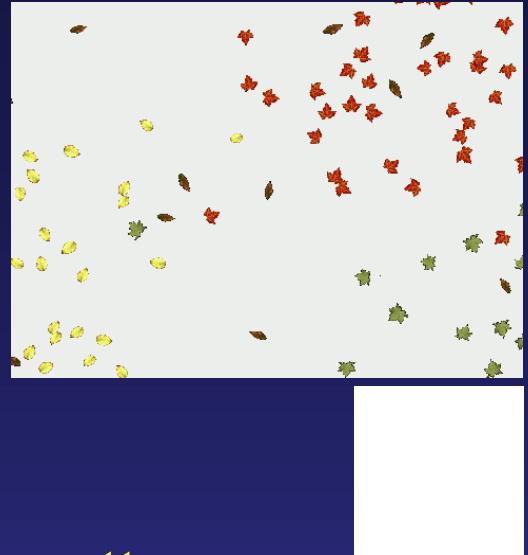
Blocks

- Pattern choice and positioning
- Transitions between neighboring patterns
- Animation



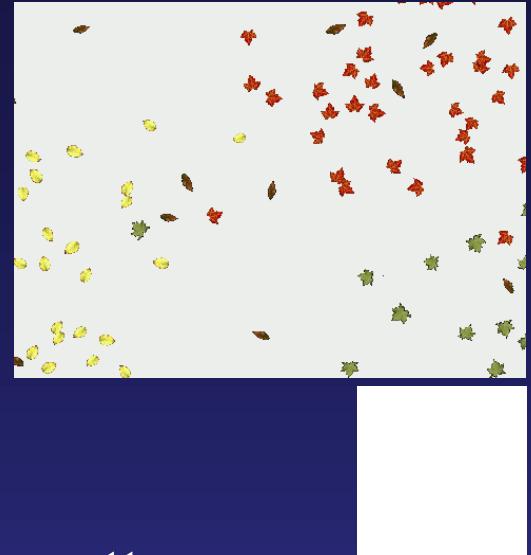
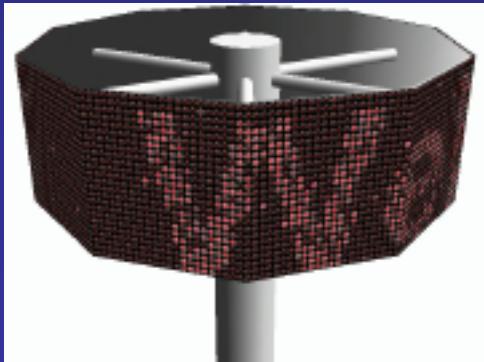
Blocks

- Pattern choice and positioning
- Transitions between neighboring patterns
- Animation



Blocks

- Pattern choice and positioning
- Transitions between neighboring patterns
- Animation

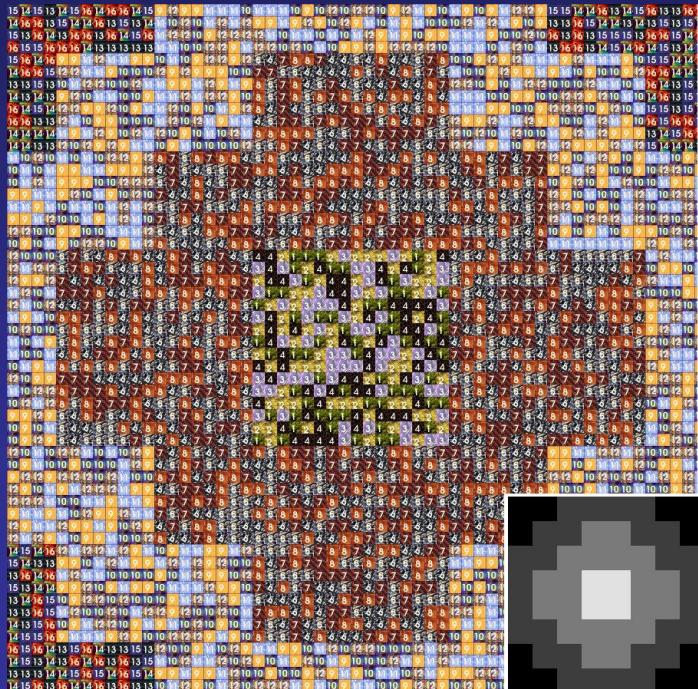


Overview

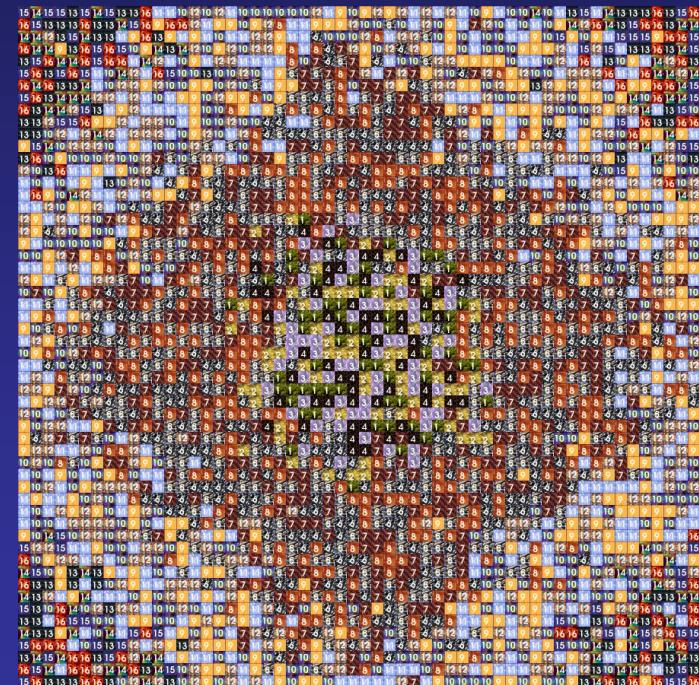
- Motivations
- Previous Work
- Contributions
- Our Framework
 - Case study
 - Results
- Conclusion

Probability distribution control

- Areas map

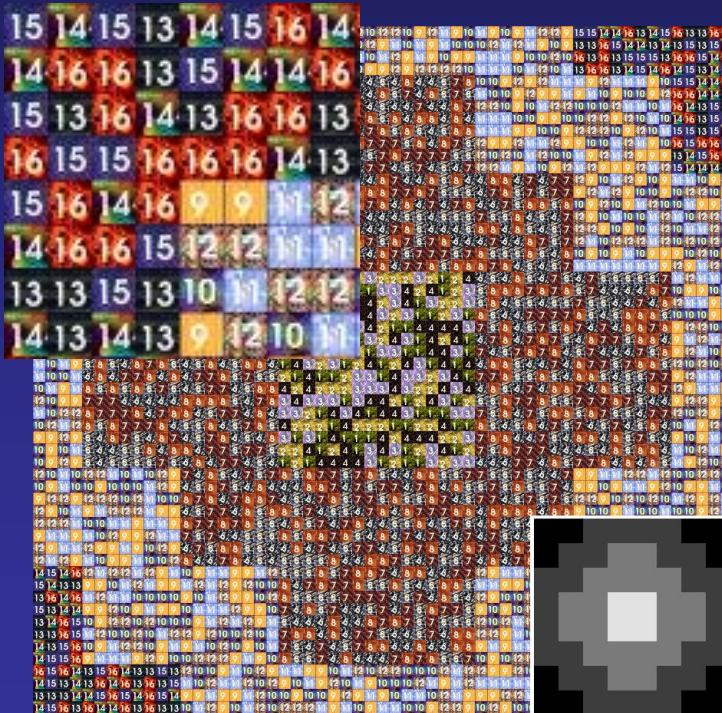


Interpolation of probabilities

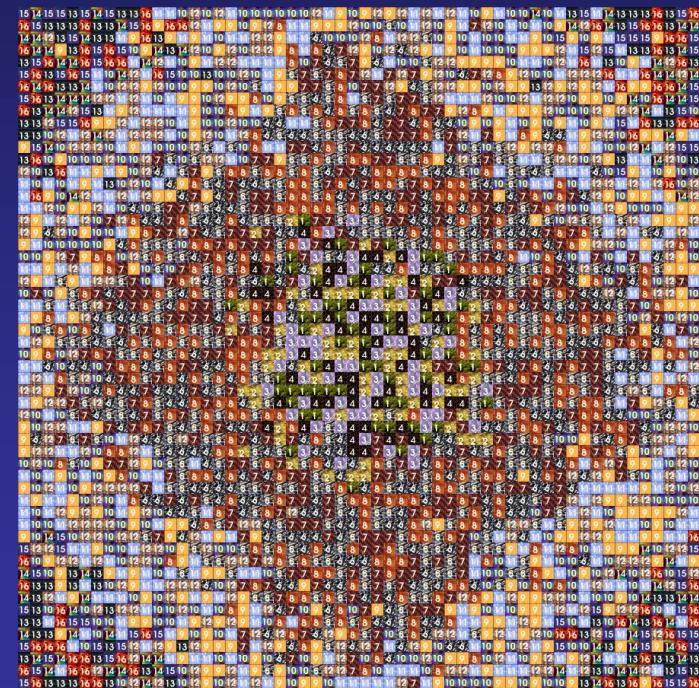


Probability distribution control

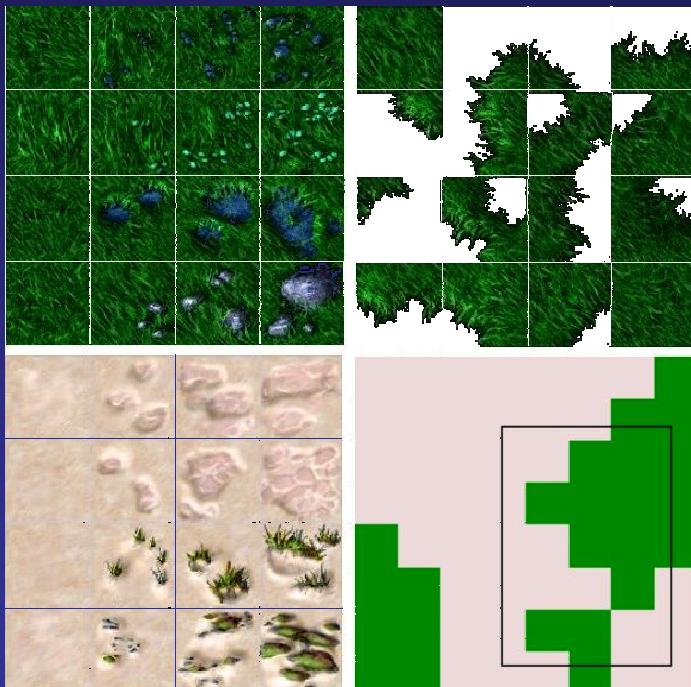
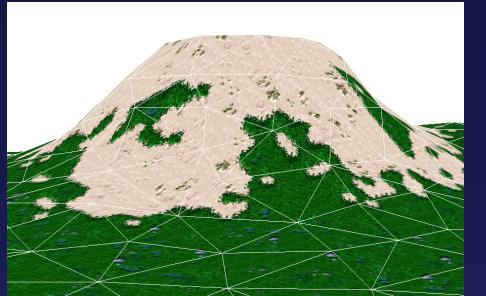
- Areas map



Interpolation of probabilities



Transitions

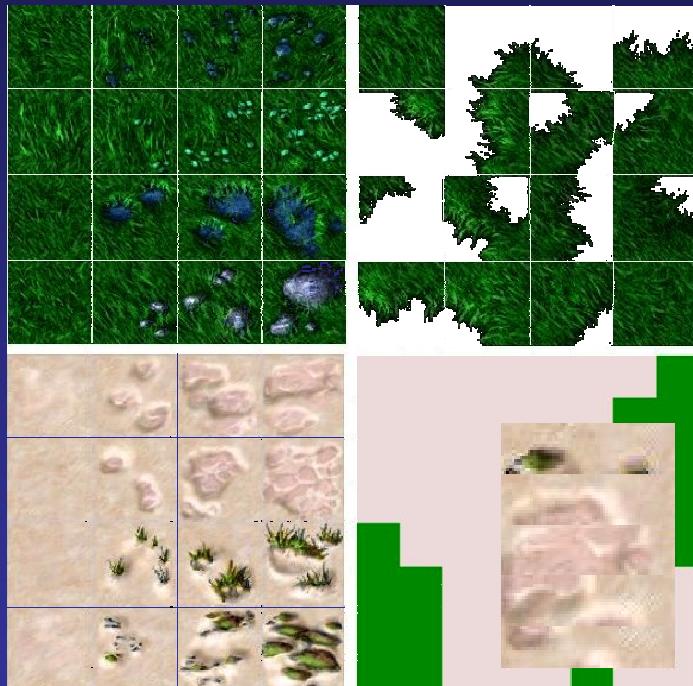
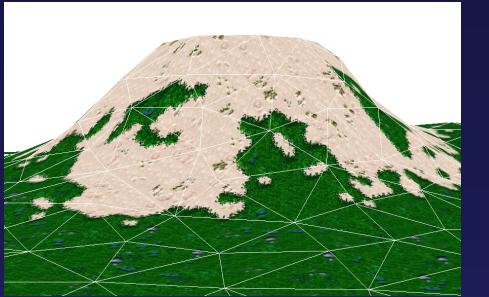


WarCraft3© Blizzard Entertainment



only one quad

Transitions

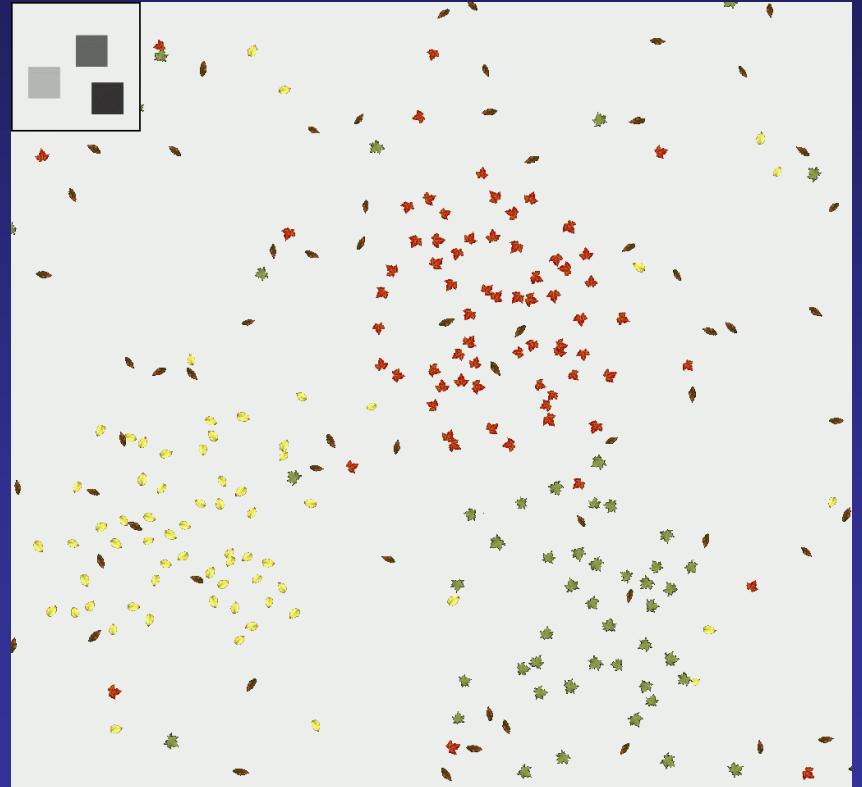
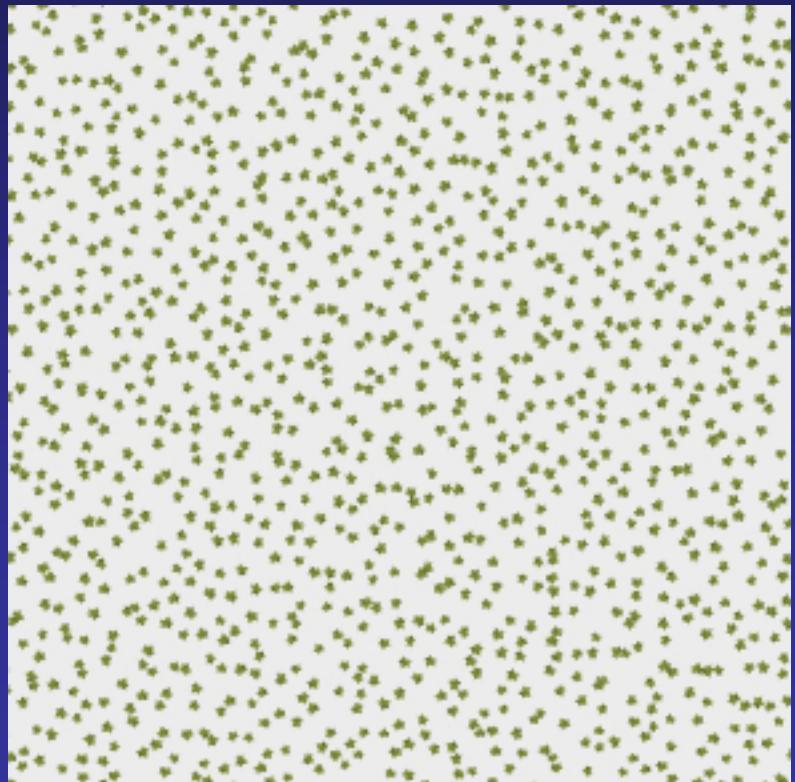


WarCraft3© Blizzard Entertainment

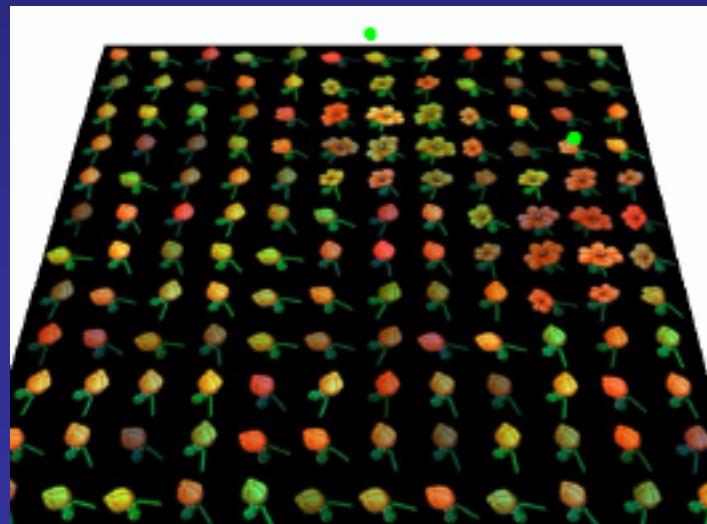
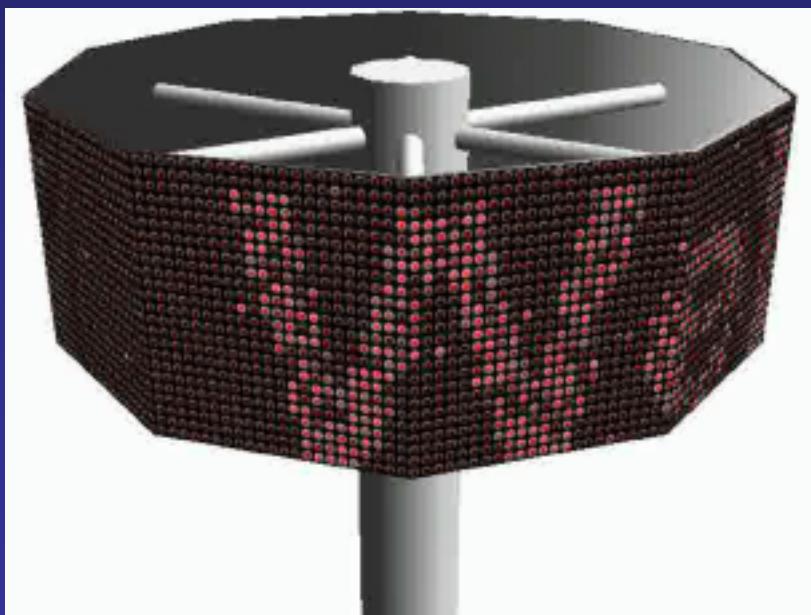
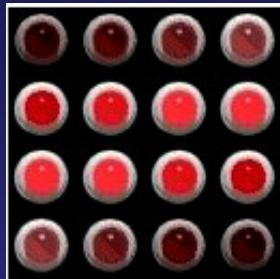


only one quad

Random positioning

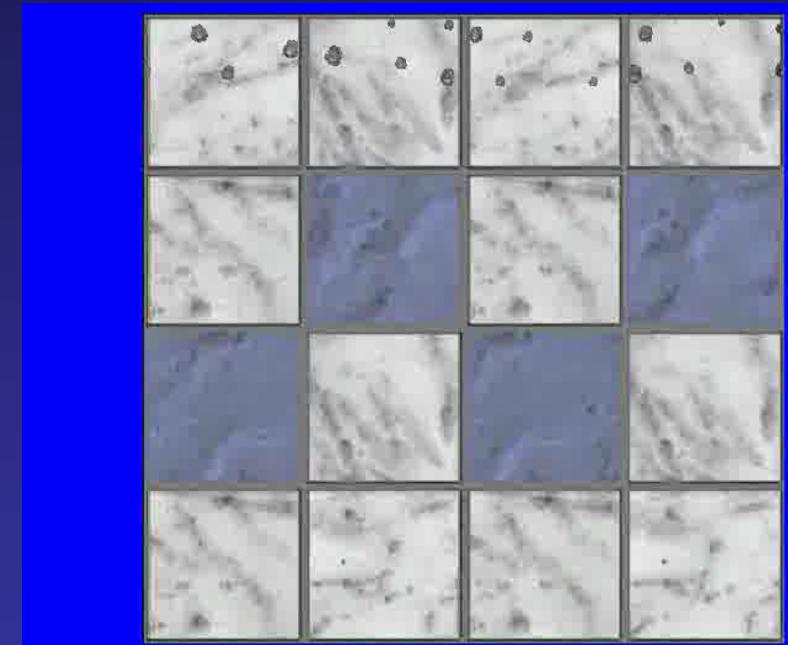
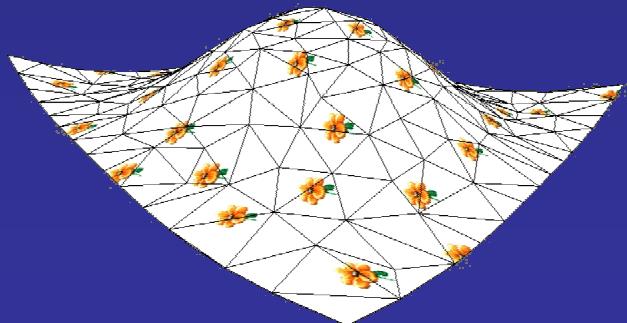
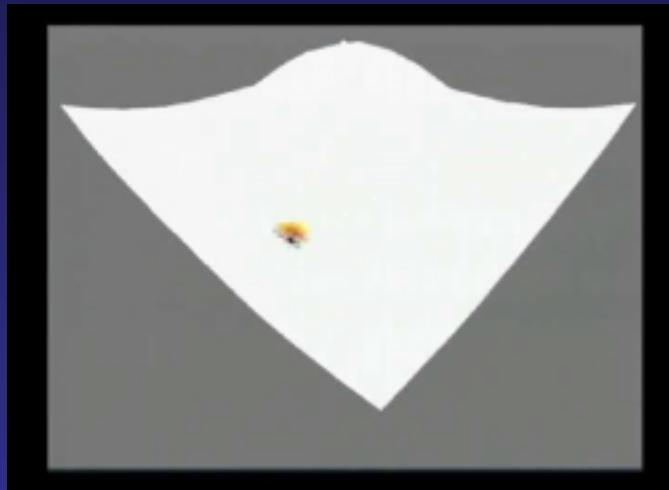


Animations



Animations

Using explicit positionning



Memory cost = pattern + 16x16 positionning map

Performances



- Measures on GeForce FX prototype (half speed)
- 32 bits precision
- texture covers full screen

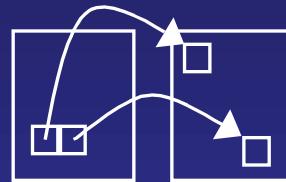
	Aperiodic tiling	Areas map	Dithered Areas Map	Dithered Areas Map & transitions
Code length	56 instr.	65 instr.	117 instr.	512 instr.
Tex. lookups	5	7	10	39
320x200	113 fps	73 fps	36 fps	5 fps
640x480	24.5 fps	15.5 fps	8.5 fps	1.1 fps

Filtering

- MIP – mapping



- Close view point
 - Linear interpolation
- Far view point
 - Indirection average color
- Issue in general with indirection textures



Conclusion

- Framework
- Runs on hardware
- Allows very large aperiodic textures
- Low memory cost

Filtering issue

Exploiting the framework

Thank you !

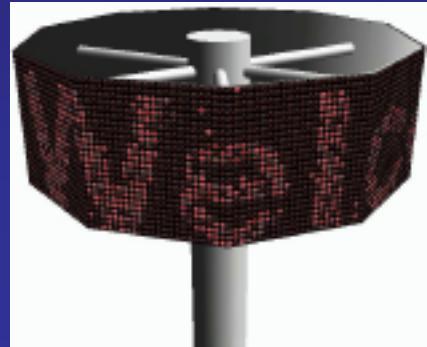
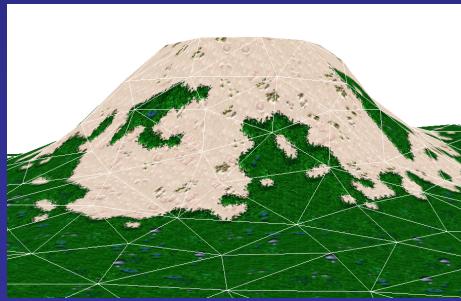
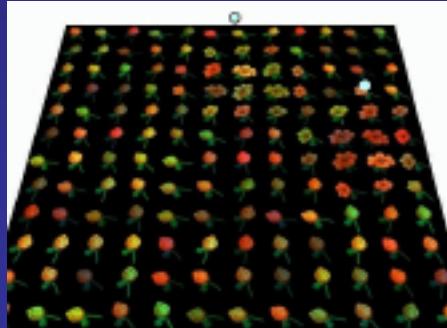
Questions ?

Pattern Based Procedural Textures

Sylvain Lefebvre

Fabrice Neyret

iMAGIS - GRAVIR / IMAG - INRIA



<http://www-imagis.imag.fr/Membres/Sylvain.Lefebvre/pattern>