# The Markup Way to multimodal toolkits

Stéphane Sire Stéphane Chatty IntuiLab http://www.intuilab.com

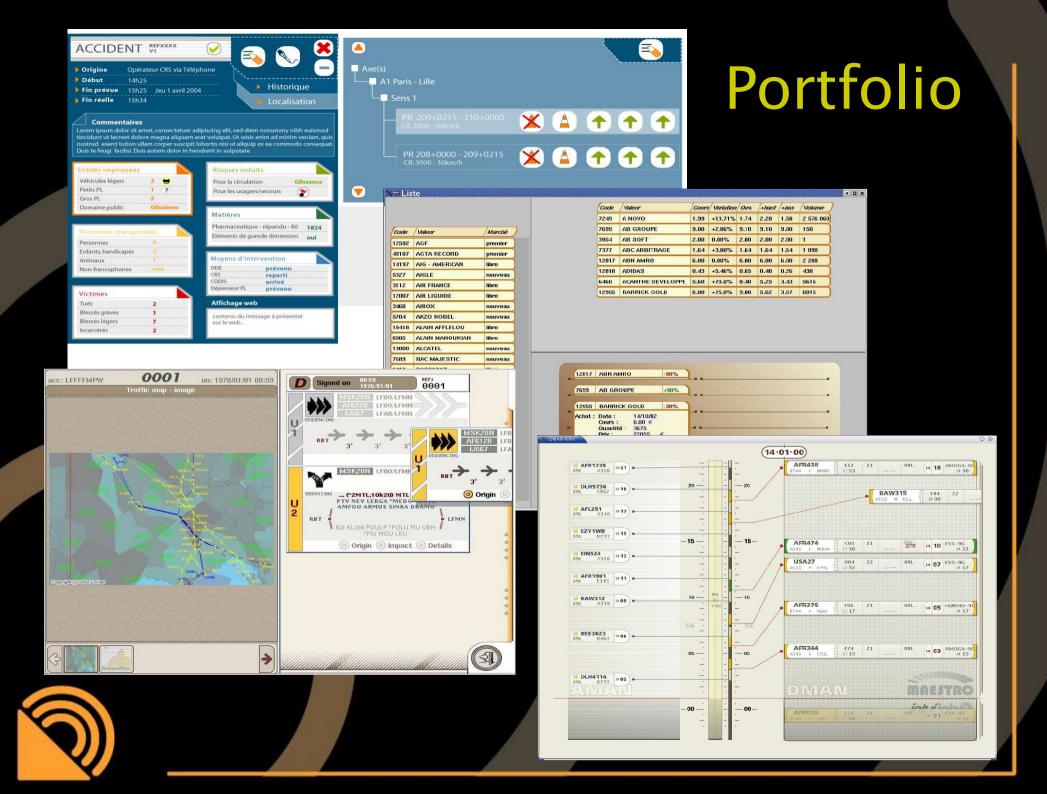


#### IntuiLab?

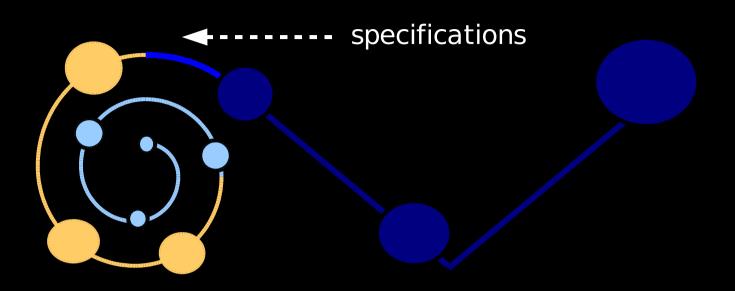
- Foundation:
  - Damien Figarol / Laurent Karsenty / Stéphane Chatty
- Intuitive User Interaction Style



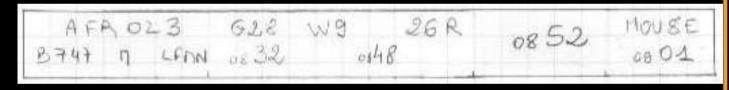




### Iterative Design Process



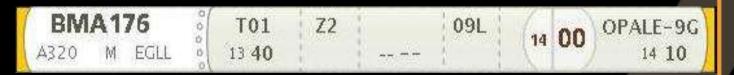
mockup



prototype



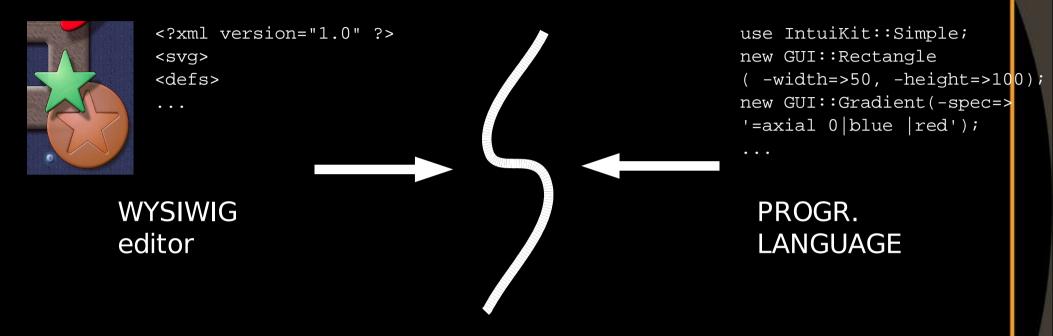
development and test





Example: three stages of a strip for Air trafic Control

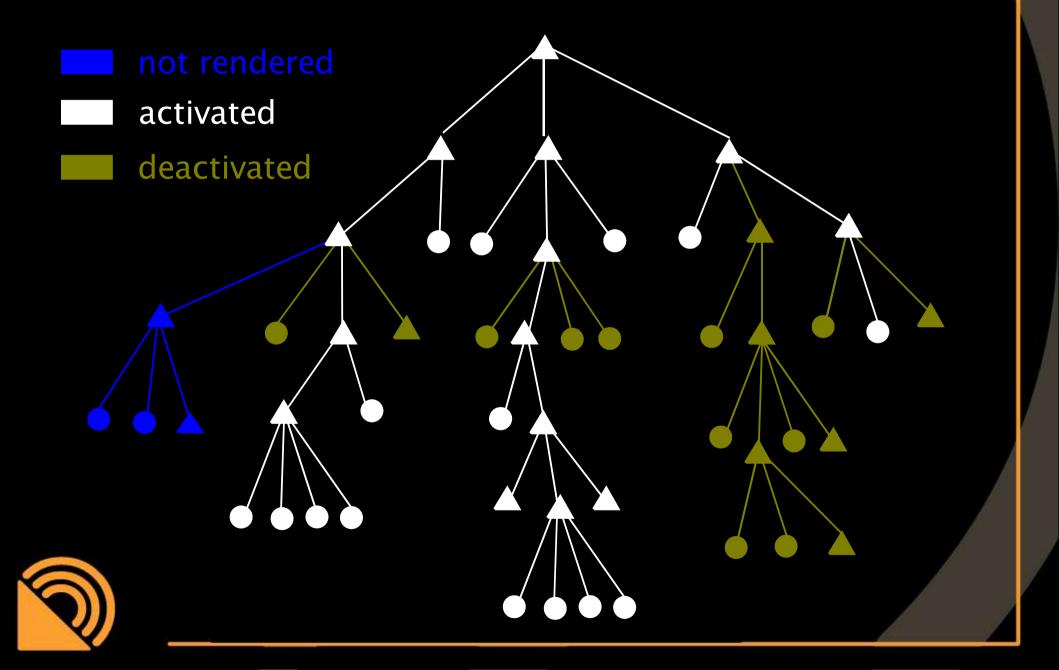
#### Model driven approach



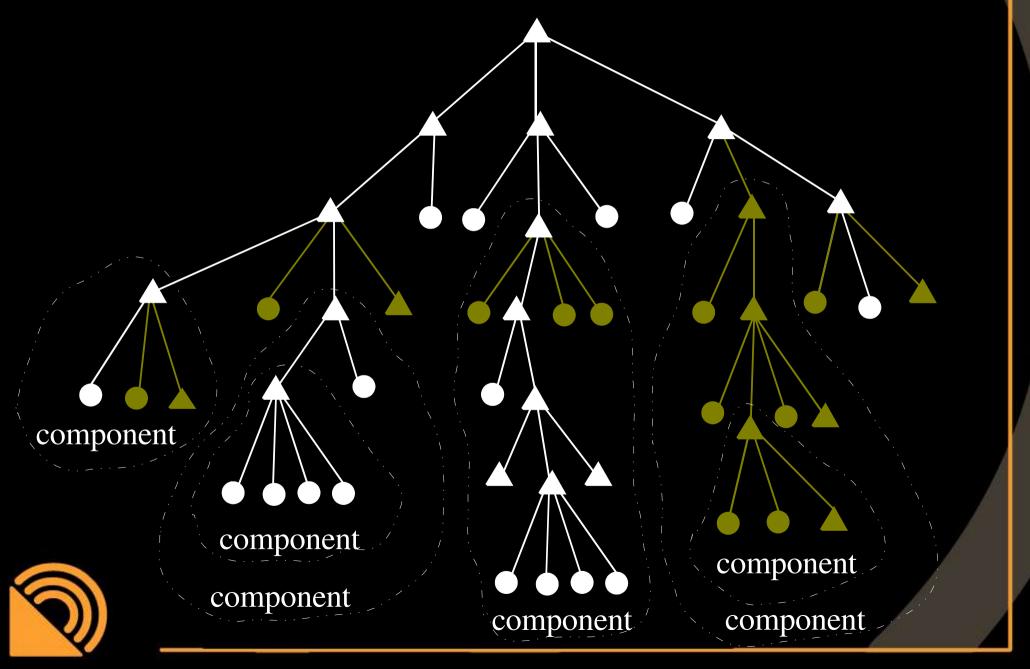
- Markup benefits
  - team work support
  - specialized editors (WYSIWIG)
  - thin and fat client deployment



#### IntuiKit Core Model



## Component Tree



#### Component services

- Properties
  - Customization
- Code execution
  - event handlers vs. methods
- Events
  - input vocabulary
  - output vocabulary
- Behaviour
  - State information and control structures

### Well covered with W3C Markup

- Properties
  - CSS
- Graphics
  - SVG with transformations and animations
- Bindings
  - XML Events
- Speech Grammars

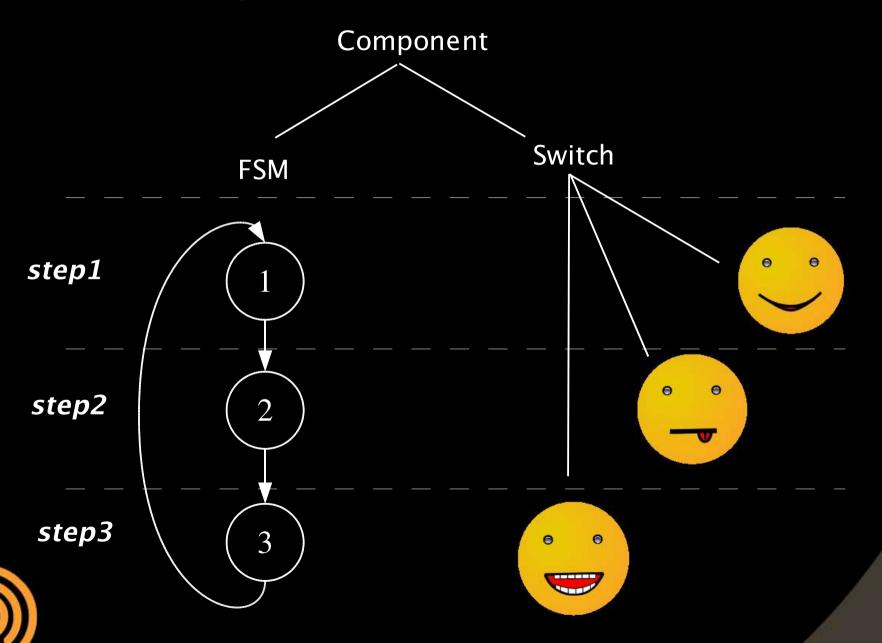


### Less covered with W3C Markup

- Control structures
  - distributed flow of control
  - FSM and Switch
- Core architecture
  - IntuiKit::Component and IntuiKit::Element
  - definition vs. instantiation
- Gestures
- Events
  - EMMA ?



# Sample control sructure



#### Core architecture

#### component definition

comp		XML	Progr. Language
component instantiation	XML	Markup Only	Code Behind
tantiation	Progr. Language	Markup Behind	Code Only

#### Discussion

- Do we need a component model?
  - XUL/XBL
  - SVG RCC for Xforms rendering
  - XAML: code behind
- Do we need more advanced control structures?
  - How to represent the distributed flow of control?

