Proxima 2.0

WYSIWYG generic editing for the Web

Martijn M. Schrage

Oblomov Systems

Universiteit Utrecht

This talk

Oblomov Systems

- Overview Proxima 1.0
- Overview Proxima 2.0
- Demo
- Speed issues
- Conclusion / Future

Proxima

Oblomov Systems

- Generic presentation-oriented (WYSIWYG) editor
- Modeless mix of
  - Structural editing: e.g. change section to subsection
  - Free-text editing: e.g. delete [1+2, 5] → [15]
- Graphical presentation with derived information
- Applications:
  - Source editor
  - Word processing
  - Form editors (e.g. declaration form)
  - Sudoku editor
- ~15.000 lines of Haskell

Proxima Screenshots

Oblomov Systems

Dazzle documentation editor

File

3.1 Echo-endoscopie loco-

De variabele modelleert voor een pat
betrekking tot de aanwezigheid van n

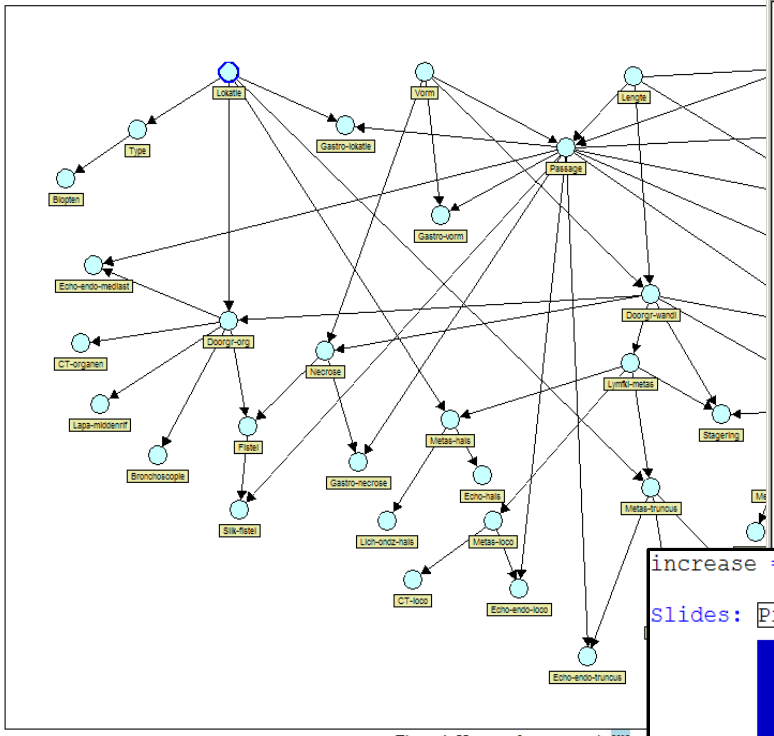
Kanstabel voor "Echo-endo-loco" m

Echo-endo-loco	Metas-loco	P
ja	moeite	ja
ja	moeite	ne
ja	gepureerd	ja
ja	gepureerd	ne
ja	vloeibaar	ja
ja	vloeibaar	ne
ja	geen	ja
ja	geen	ne
nee	moeite	ja
nee	moeite	ne
nee	gepureerd	ja
nee	gepureerd	ne
nee	vloeibaar	ja
nee	vloeibaar	ne
nee	geen	ja
nee	geen	ne
niet-gedetermineerd	moeite	ja
niet-gedetermineerd	moeite	ne
niet-gedetermineerd	gepureerd	ja
niet-gedetermineerd	gepureerd	nee
niet-gedetermineerd	vloeibaar	ja
niet-gedetermineerd	vloeibaar	nee
niet-gedetermineerd	geen	ja
niet-gedetermineerd	geen	nee

Dazzle documentation editor

File

Een beslisnetwerk voor oesofagus carcinomen



Figuur 1: Het oesofagus netwerk.

1 Inleiding

De doelstelling van de ontwikkeling van een beslisnetwerk voor oesofagus carcinomen is het verkrijg

Helium editor

File

```
Slides: XML view

        <slides>
          <slide><title>slide_1</title>
            <itemList style="Bullet">
              <stringItem>item_1</stringItem>
              <heliumItem>
                \x -> increaze x
              </heliumItem>
              <stringItem>item_2</stringItem>
              <listItem>
                <itemList style="Number">
                  <stringItem>nested_item_1</stringItem>
                  <listItem>
                    <itemList style="Bullet">
                      <stringItem>nested_nested_item</stringIte
                      <stringItem>nested_nested_item</stringIte
                      <stringItem>nested_nested_item</stringIte
                    </itemList>
                  </listItem>
                  <stringItem>nested_item_2</stringItem>
                  <stringItem>nested_item_3</stringItem>
                </itemList>
              </listItem>
            </itemList>
```

```
increase = \x -> x+1;

Slides: Presentation view
```

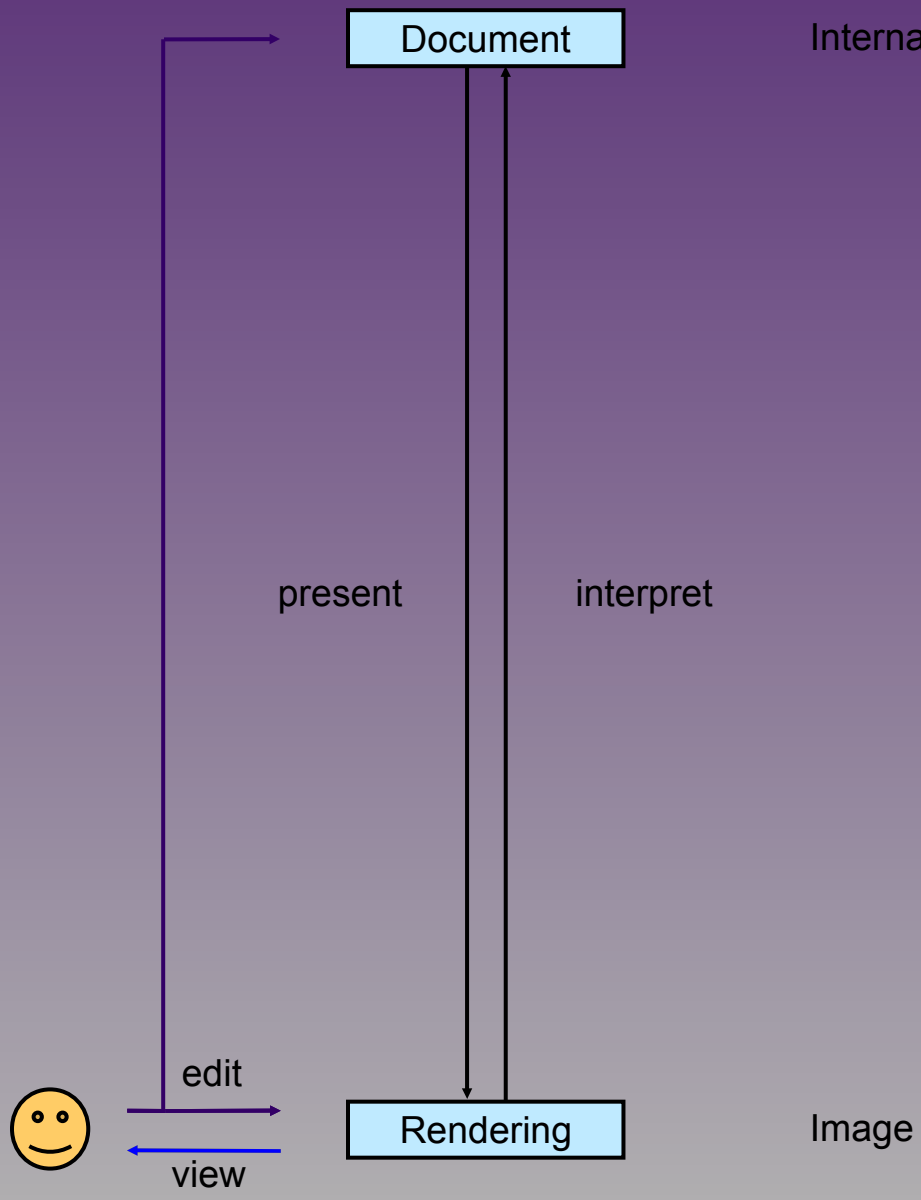
slide_1

- item_1
- \x -> increaze x
- item_2
  1) nested_item_1
     - nested_nested_item
     - nested_nested_item
     - nested_nested_item
  3) nested_item_2
  4) nested_item_3

```
(-1,-1): Undefined variable "increaze"
  Hint: Did you mean "increase" ?
```

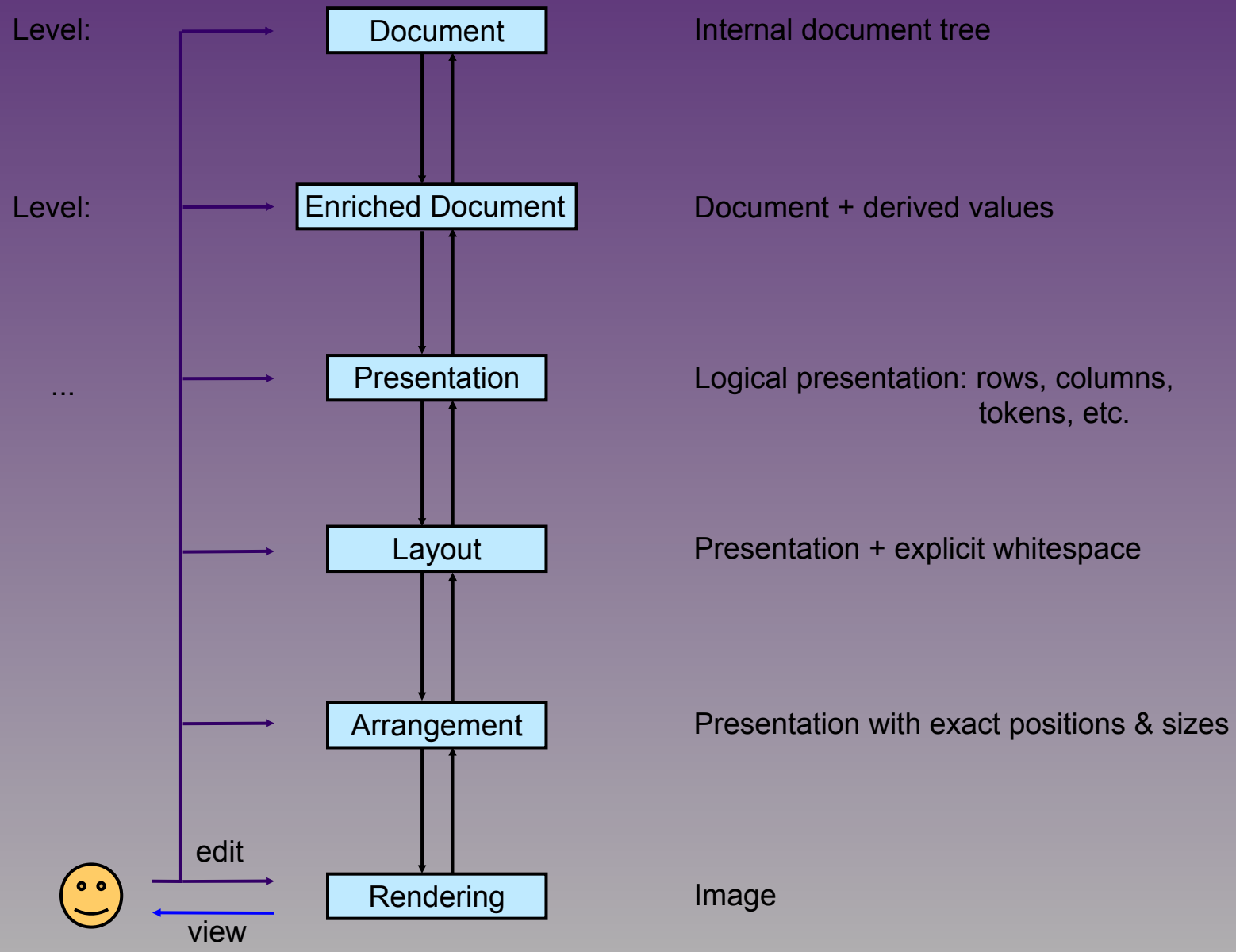
Architecture

Oblomov Systems



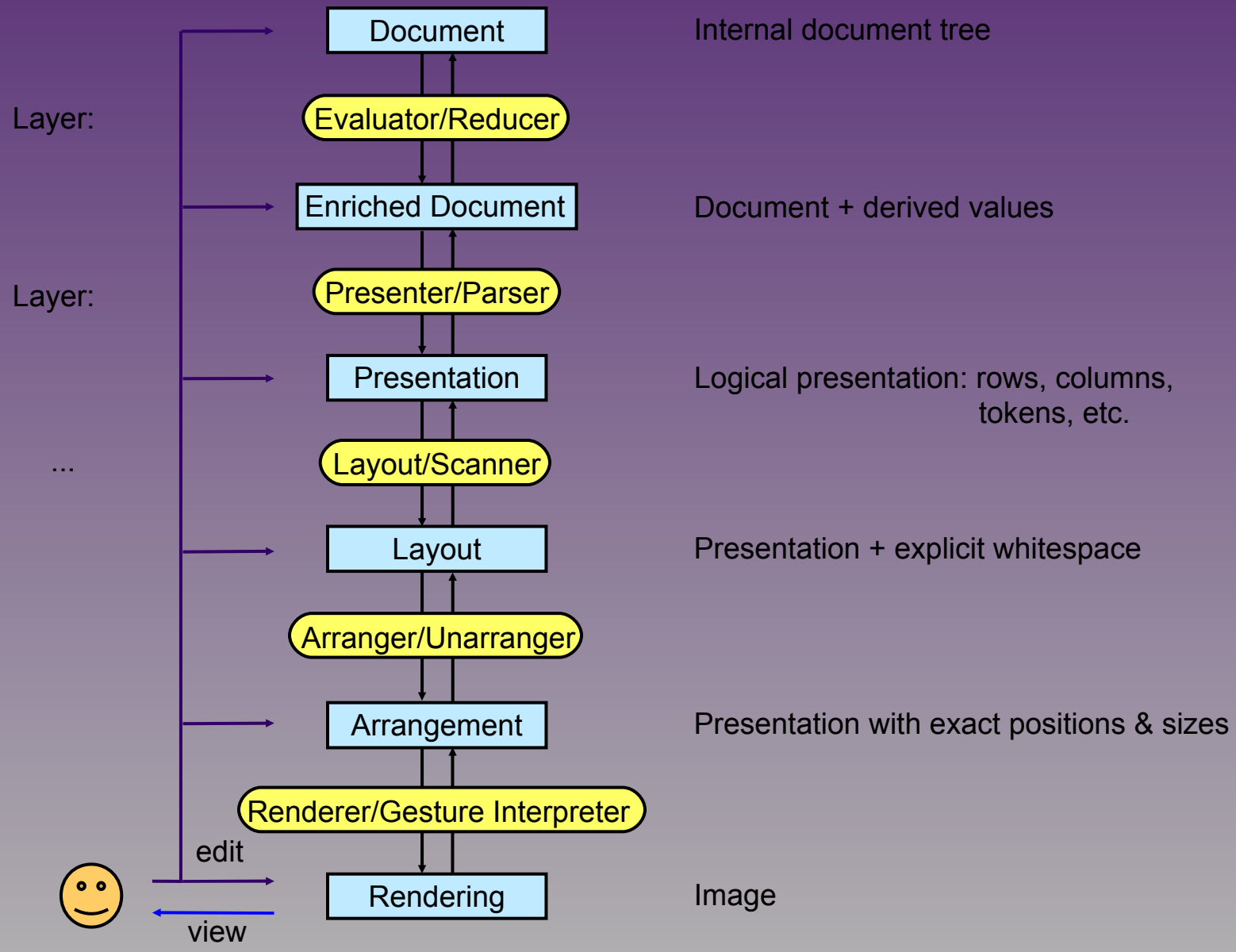
Architecture

Oblomov Systems



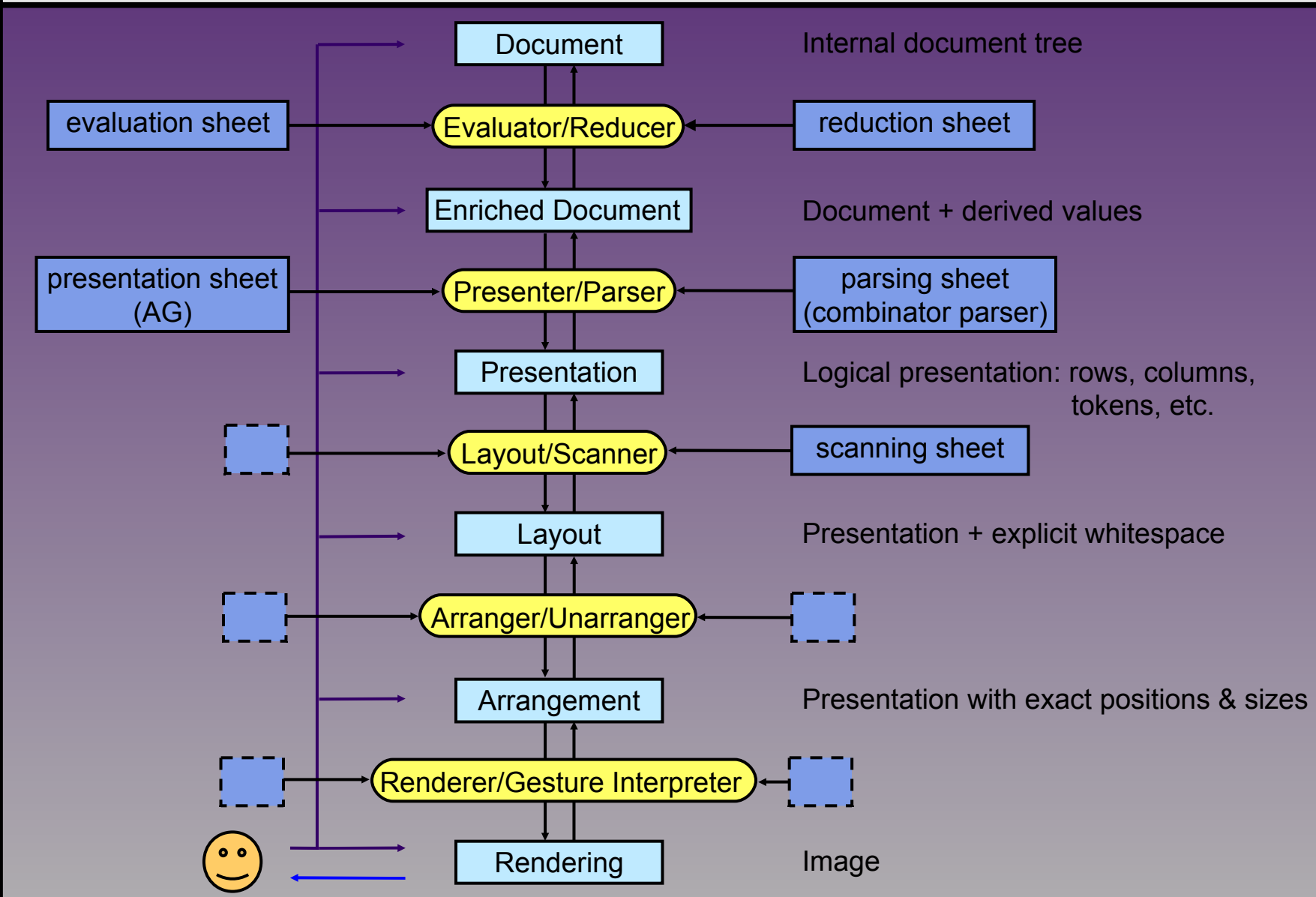
Architecture

Oblomov Systems



Architecture

Oblomov Systems



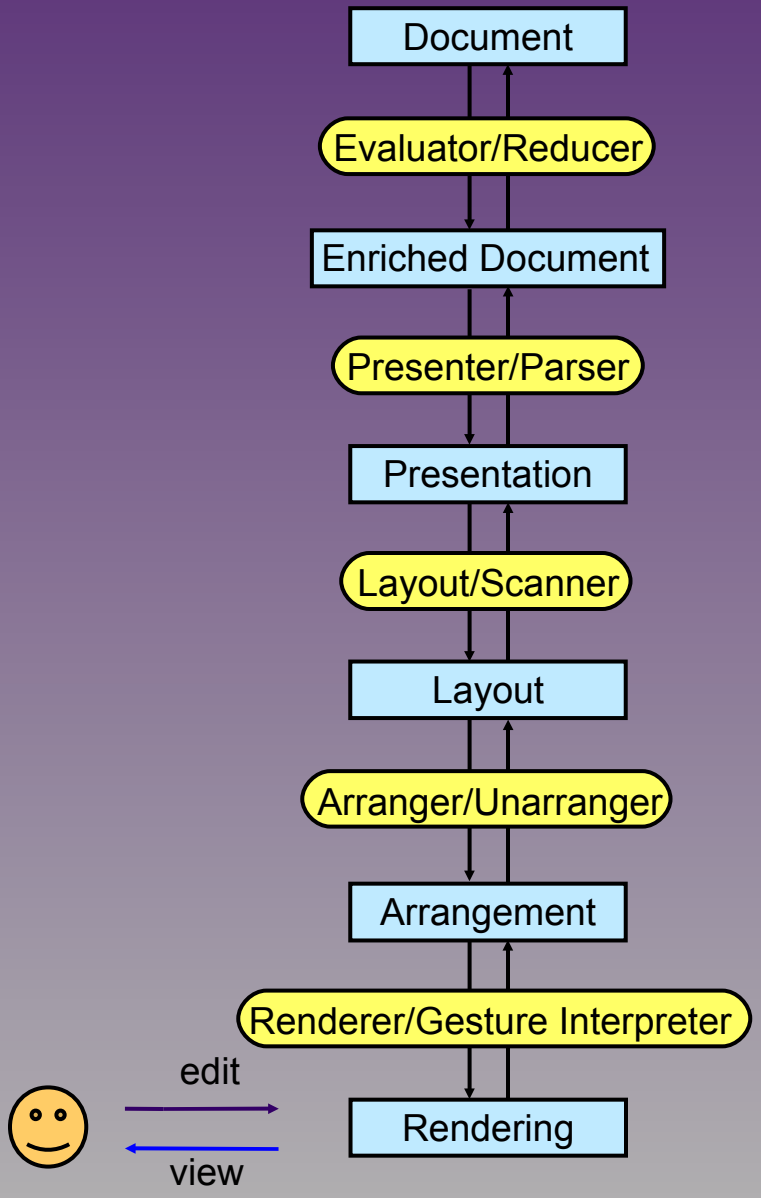
Proxima 2.0

Oblomov Systems

- Proxima 1.0
  - Has to be installed
  - 100% Haskell, but GUI dependency makes it hard to build
- Grant from NLnet foundation: open source; web-based
- Proxima 2.0
  - Server-based with (very) thin Ajax client
  - Every key press or mouse gesture is sent to server (à la VNC)
  - Powerful WYSIWYG web 2.0 editors
  - No installation required for editing users
  - Run on small client machines, e.g. iPhones
  - Integration with web-pages
  - Server is easy to compile

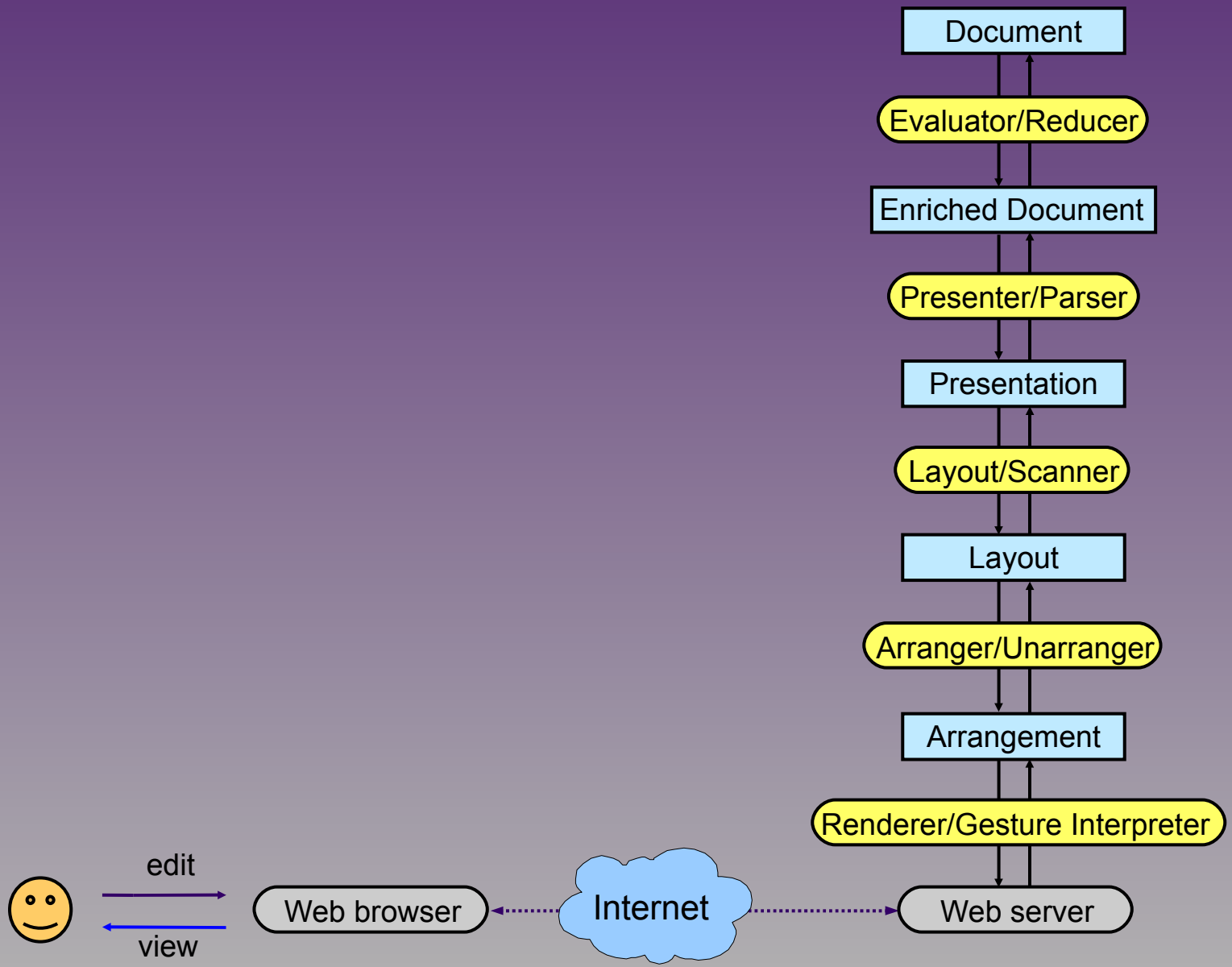
Architecture 1.0

Oblomov Systems



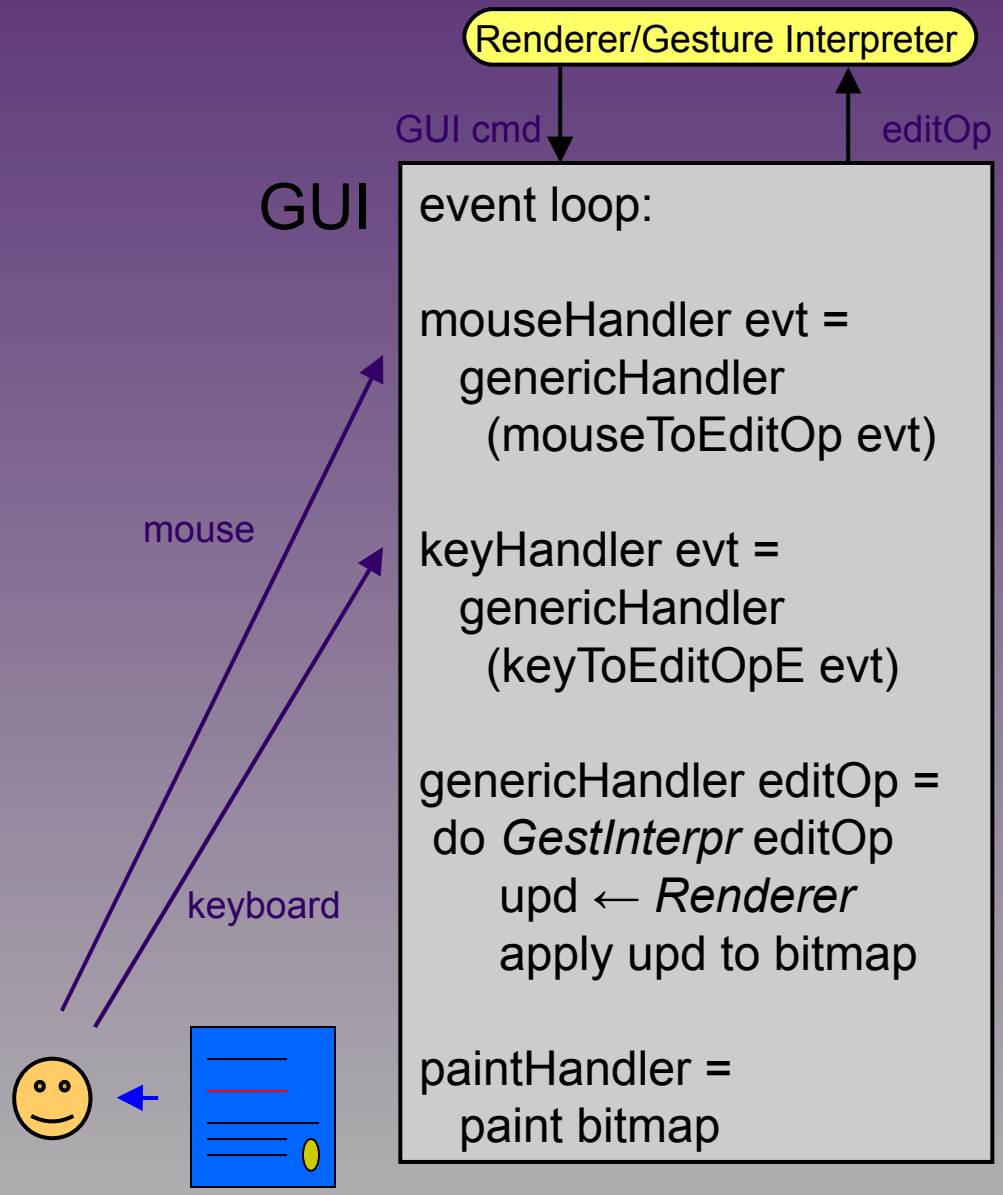
Architecture 2.0

Oblomov Systems



Proxima 1.0 (simplified)

Oblomov Systems



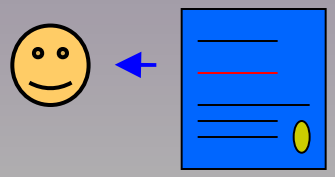
Proxima 2.0 (simplified)

Oblomov Systems

client (browser)

mouse

keyboard



```
<script>
  event loop:

  mouseHandler(evt) =
    genericHandler
      (mouseToEditOp(evt));

  keyHandler(evt) =
    genericHandler
      (keyToEditOp(evt));

  genericHandler(editOp) =
    send("GET "+editOp);

  on receiving HTML update,
  apply update to rendering tree
</script>
<rendering>
...
</rendering>
```

Renderer/Gesture Interpreter

HTMLupd

editOp

```
listen on HTTP port:

accept socket =
 do str ← getLine socket
    let editOp = parse str

    GestInterpr editOp
    htmlUpd ← Renderer
    putStr socket HTMLupd
```

server

Renderer

Oblomov Systems

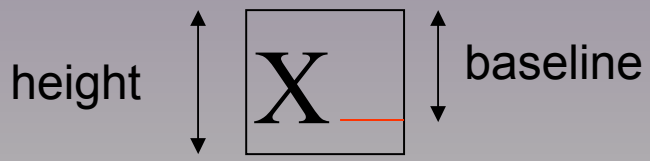
- Convert each arrangement to HTML div elements with exact sizes and positions
- Rows / columns:
  - Div element
- Strings:
  - HTML text
- Graphical elements: (`CircleA`, `PolyA`, etc.)
  - SVG
- Images:
  - Image tag "`<img src='...'/>`"
  - Server needs to handle image requests

Font metrics

Oblomov Systems

- Server-side query not possible
  - Requires GUI library or parsing of font tables
  - Different browsers / platforms may render differently
- Solution: Query the client for metrics
- Hmm, Javascript has no `getFontMetrics()`..

- Solution:
  - Add lines with invisible characters to rendering
  - Copy each character 10 times; take line width / 10
  - For baseline, special trick with zero height element (—)



And..

Oblomov Systems

- Session handling for multiple viewers
- Document handling (up/download)
- Mouse handler for drag & drop and scrolling
- Popup handler

Proxima 2.0 demo

Oblomov Systems

- Bayesian network docs editor (800 lines)
- Helium editor (1200)
- Declaration form (160)
- Task list editor (140)
- Sudoku (210)

- Live demo!
- Editors run on virtual server in Utrecht

Speed: Bandwidth

Oblomov Systems

- Complex presentations take time to transmit
- Solution: Incrementality
- After edit operation, server computes diff on arrangement
- Client applies inserts, deletes and moves to html rendering tree
- Diff detects insertions/deletions in rows/columns

- Pathological cases always exist

Speed: Latency

Oblomov Systems

- Delay between gesture and server response
- Solution: Predictive rendering
- Drag & drop: don't send drag events
- Text input:
  - Insert character in HTML, move right siblings, move cursor
  - Subtle algorithm, two kinds of inserted chars: already sent to server vs. queued

Conclusion / Future

Oblomov Systems

- Conclusion
  - Editing model works!
  - Editors feel fast enough
- Open source project
- To do:
  - Establish community
  - Build more editors
  - More predictive rendering:
    - E.g. cursor moves
    - More intelligent, maybe with hints in presentation
  - Add support for IE (has no SVG)
  - Multi-user editing

WE WANT YOU!

Questions?

Oblomov Systems

martijn@oblomov.biz

WE WANT YOU!

http://www.oblomov.biz/proxima2.0.html