Timesheets

Integrating Timing in XML

Warner ten Kate
Patrick Deunhouwer
Ramon Clout

Philips
Research

PHILIPS

Overview

- Motivation
- Requirements to timing model
- Approaches
- Timesheets
- Experiment / demonstration

Goal

Goal:

Provide timing syntax and semantics that can be integrated with any XML document.

Cf. XSL provides layout formatting for any document.

Requirements on Timing Model

- Req. 1: Independent of semantics of XML document (generally applicable)
  - Ability to independently structure timing
- Req. 2: Independent timing control of various components/properties (layout, style, content)
  - Ability to control classes of elements
  - Ability to build templates

Consequence

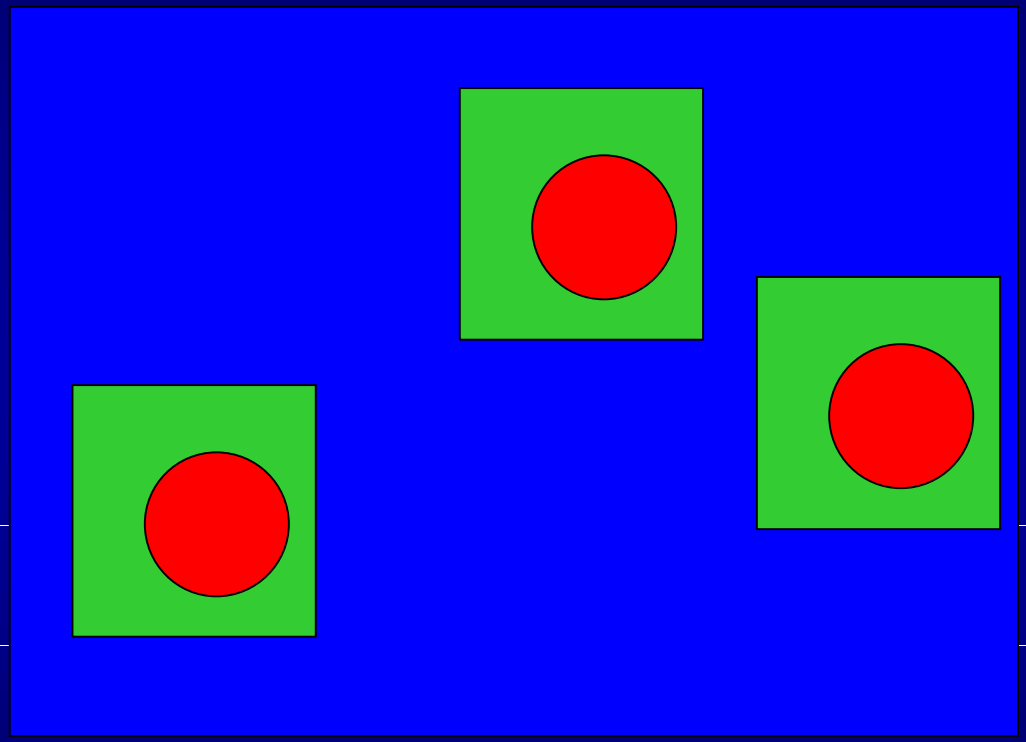
- Timing in logically separate section
- Allow timing structure to be incongruent with structure of other component:
  - Containment
  - Order
  - Attributes

Structure

(timing; layout; content)

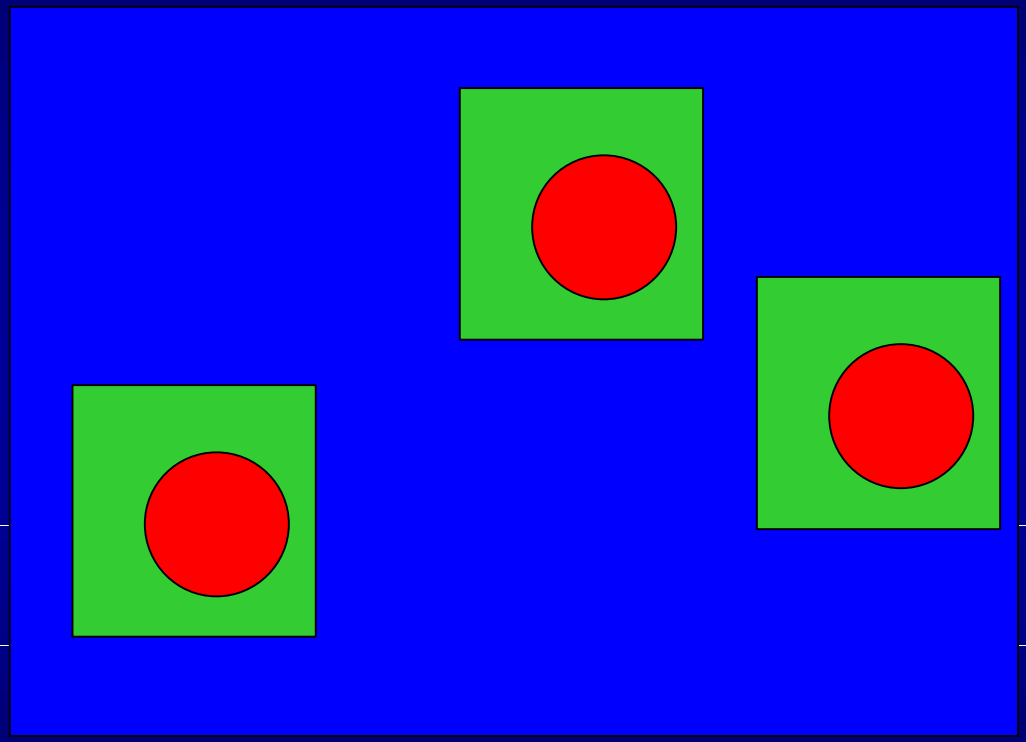
- Containment
  - during; within; (sub)list
- Order (implied by containment)
  - sequential; side-a-side; list items
- Attributes
  - refine within scope of containment

Containment



```
<rectangle … >
    <square … >
        <circle … >
    </square … >
    <square … >
        <circle … >
    </square … >
    <square … >
        <circle … >
    </square … >
</rectangle … >
```

Containment



```
<seq>
    <rectangle … />
    <par>
      <square … />
      <square … />
      <square … />
    </par>
    <par>
      <circle … />
      <circle … />
      <circle … />
    </par>
</seq>
```

Incongruency

```
<rectangle … >
   <square … >
      <circle … >
   </square … >
   <square … >
      <circle … >
   </square … >
   <square … >
      <circle … >
   </square … >
</rectangle … >
```

```
<seq>
   <rectangle … />
   <par>
     <square … />
     <square … />
     <square … />
   </par>
   <par>
     <circle … />
     <circle … />
     <circle … />
   </par>
</seq>
```

Order

```
<winner>
  <gold> photon </gold>
  <silver> sound </silver>
  <bronze> smell </bronze>
</winner>
```

- Layout: list fastest first
- Time: list first prize last

Circumvent incongruency

- Containment
  - flatten tree
- Order
  - no order semantic
  - absolute attributing

- Counteract the purpose of XML:
  - structure information

PHILIPS

Approaches

- Inline
- Stylesheet
- Timesheet

Inline

- Apply modularization and profiling
  - Example: HTML + SMIL
  - Attribute oriented (use begin, end attributes)
  - No order semantics
- Pro: timing appears with the object (intuitive authoring)
- Con: less suited for more complex timing

Stylesheet

- (CSS) stylesheet contains timing properties
  - basically same as Inline
  - class alignment, template usage
- Pro: template usage
- Con:
  - timing property & timing structure separated
  - timing control over style is hard

Timesheet

Timesheet's operation is that of

- temporally selecting entities
  - i.e. *not* instantiating
- applying host language semantics to selected sub-document

Timesheet

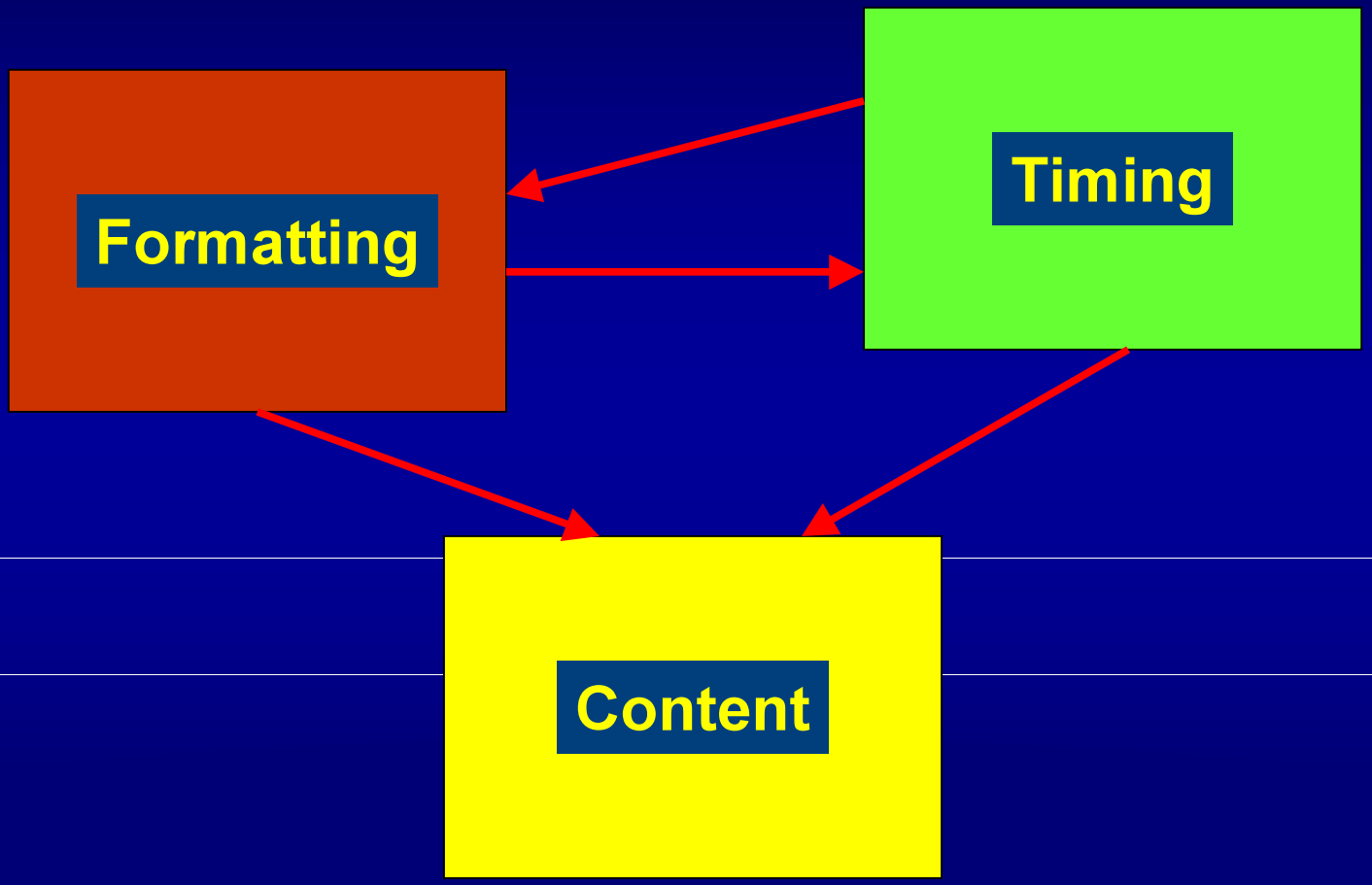
- Pro:
  - independent of XML structure and semantics
    - independent engine
  - temporal control of both content and style
  - template usage
- Con:
  - complex for simple timing operations

Two principles

- **Principle 1:**
  3 sections: content, format, time
  ⇒ Req 2: Independent timing control of various components/properties

- **Principle 2:**
  assign relations between entities
  (instead of: properties to entities)
  ⇒ Req 1: Independent of semantics of XML document

Principle 1: 3 sections



PHILIPS

Principle 2: assign relations

	"assign property"	*"assign relation"*

Timesheet	.seq {
	  timeContainer : seq;
	}
		<seq>
		  <item select="#item2"/>
		  <item select="#item1"/>
		</seq>

Document	<ul timeClass="seq"> - one - two 	 <li id="item1">one <li id="item2">two

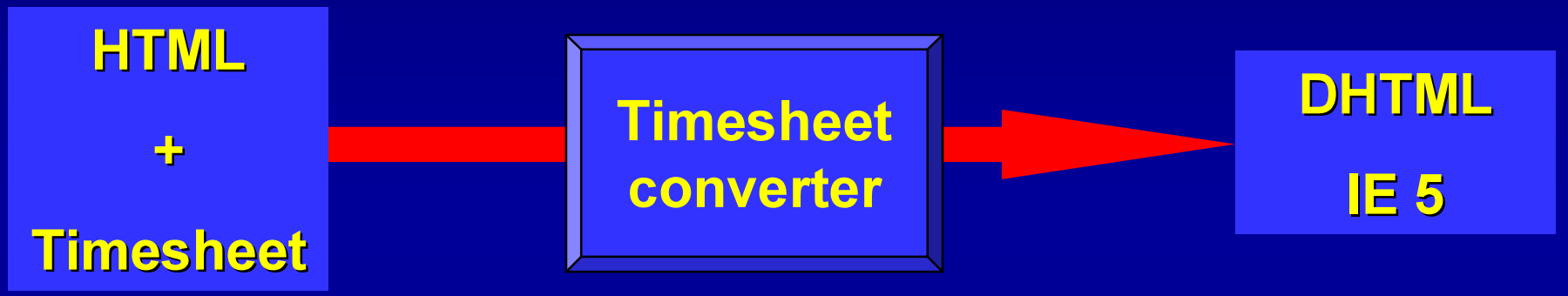
Implement Timesheet

- Generic and independent part:
  - timesheet (semantic) parser
  - scheduler
    - SMIL timing model
- Specific part:
  - XML-language semantics (e.g. HTML)

Experiment

- Convert HTML + timesheet into DHTML
- Perform the generic part
  - parse timesheet, build presentation schedule
- Leave HTML specific part to HTML browser
  - semantics of HTML handled by HTML browser
  - timing added by JavaScript

Demonstration



PHILIPS