

CausalML

A XML-based file format for storing Why-Because-Graphs

Version 1.0.1

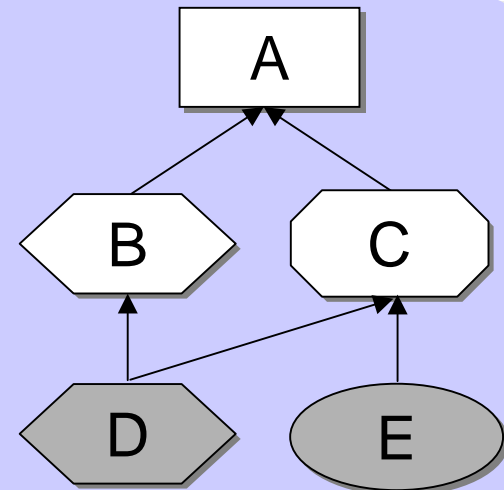
Dipl.-Ing. Oliver Lemke, 02/2004

- Introduction
- Requirements
- Basic concepts
- Realisation
- Example

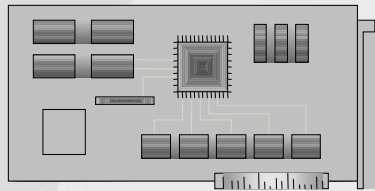
One result of a WBA is a set of relationships between causal factors. Basically, this is abstract data:

$B \Rightarrow A$
$C \Rightarrow A$
$D \Rightarrow B$
$D \Rightarrow C$
$E \Rightarrow C$

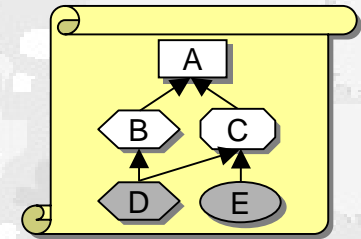
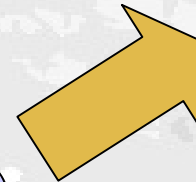
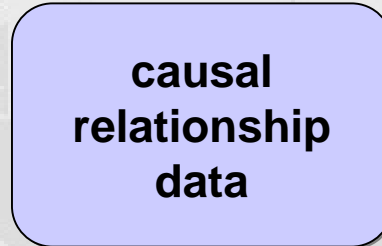
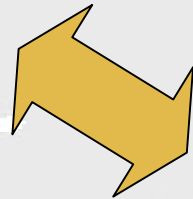
These relationships are usually rendered as a Why-Because-Graph, which is an intuitive way of displaying the result of a WBA.



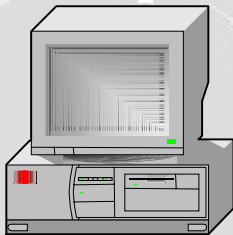
Possible usage of causal relationship data



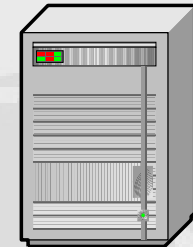
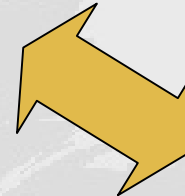
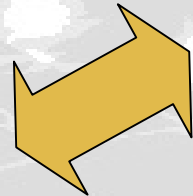
process



display



edit



store

Due to the central role of causal relationship data, a universal file format for storing this information is needed.

- Introduction
- **Requirements**
- Basic concepts
- Realisation
- Example

General requirements of a file format

Standardized and well documented

- File format should be based on an open standard (no licensing)
- File format should be well documented

Flexible and widely supported

- It should be easy to adapt the file format to the specific needs
- Reading and writing of data in the file format should be supported by a wide range of operating systems, programming languages and development toolkits

Nice-to-have-features

- File format should be human readable, at least to a certain degree

Specific requirements for CausalML

Freely definable kinds of nodes

- The WBA originally distinguishes between states, events, processes and non-events.
- Some users may want to define their own types of nodes like e.g. countermeasures (“Siemens-style”)
- Therefore it should be user definable, which kinds of nodes are available in a document.

Consistency checking

- The schema should enable parsers to perform consistency checking like e.g. checks for double or missing IDs, invalid links etc.

Adaptive complexity

- Only absolutely necessary items should be mandatory; additional features should be optional

- Introduction
- Requirements
- **Basic concepts**
- Realisation
- Example

XML as basis for a file format

Standardized and well documented

- XML is a mark-up language standardized by the W3C
- XML is well documented; documentation is available to the public

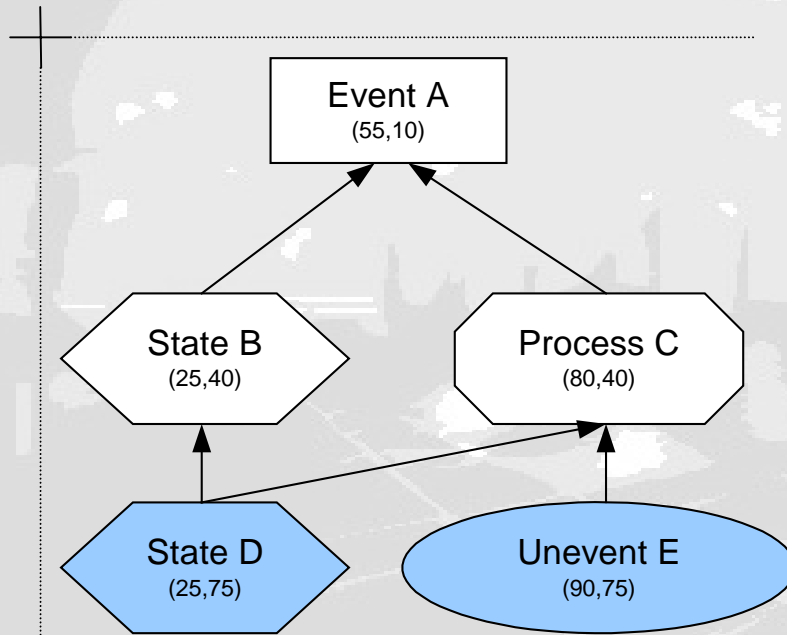
Flexible and widely supported

- The XML standard only defines *how* data is stored, but not *what* information is contained in a file – everyone can create his own file format based on XML
- XML is supported by a large number of toolkits available for nearly all programming languages and system platforms

Nice-to-have-features

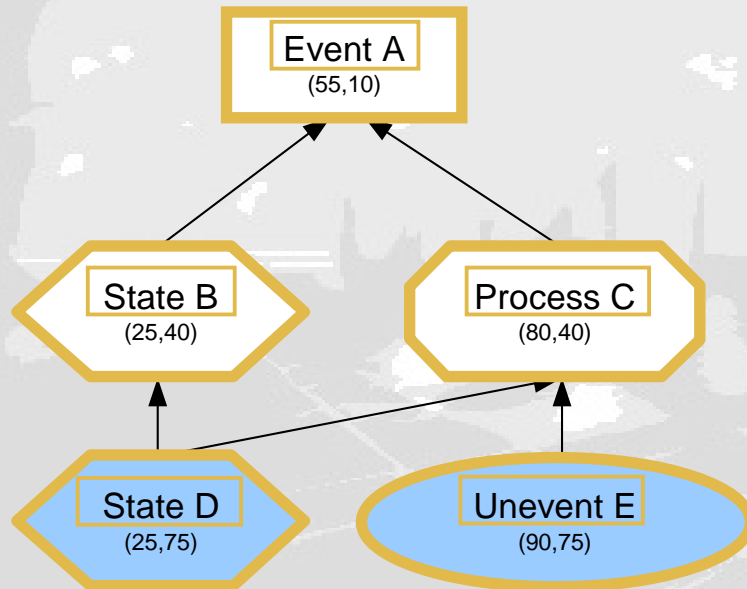
- All computer scientists fluently read XML... 😊

What information is contained in a WB-Graph?



What information is contained in a WB-Graph?

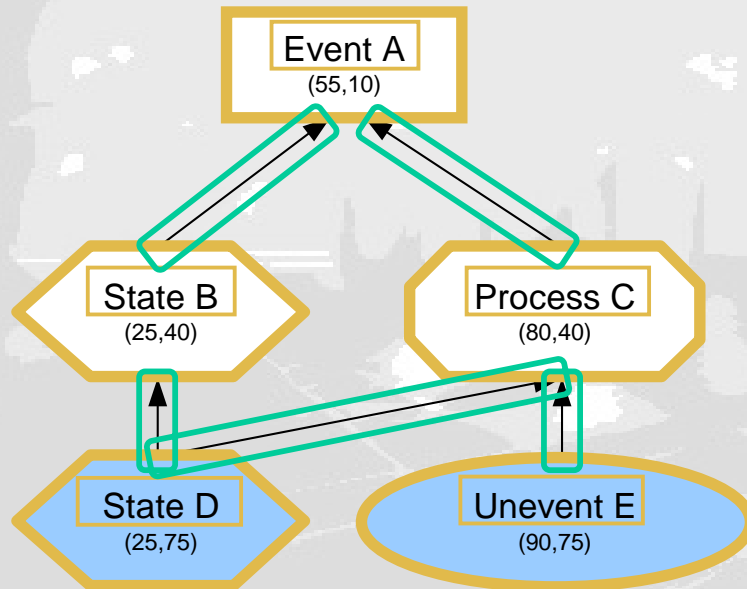
Information on the nodes



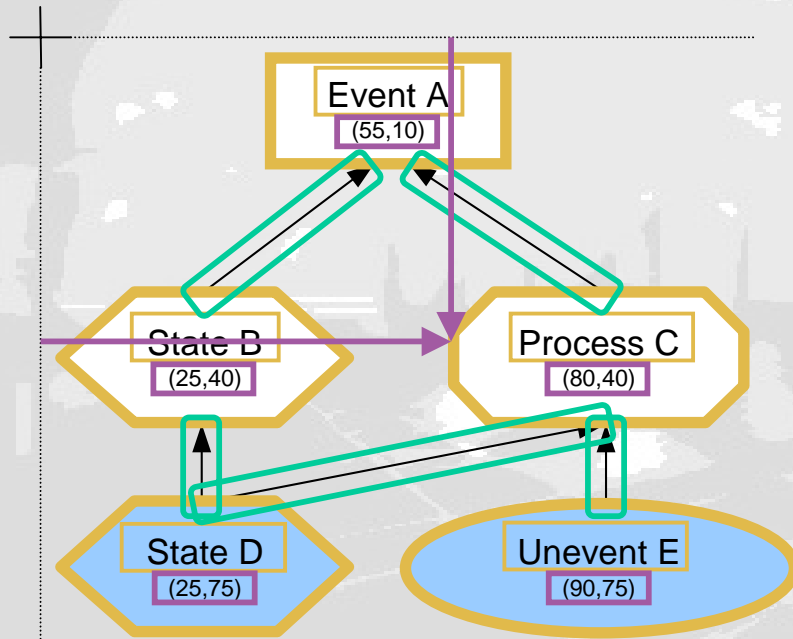
What information is contained in a WB-Graph?

Information on the nodes

Information on their relationships



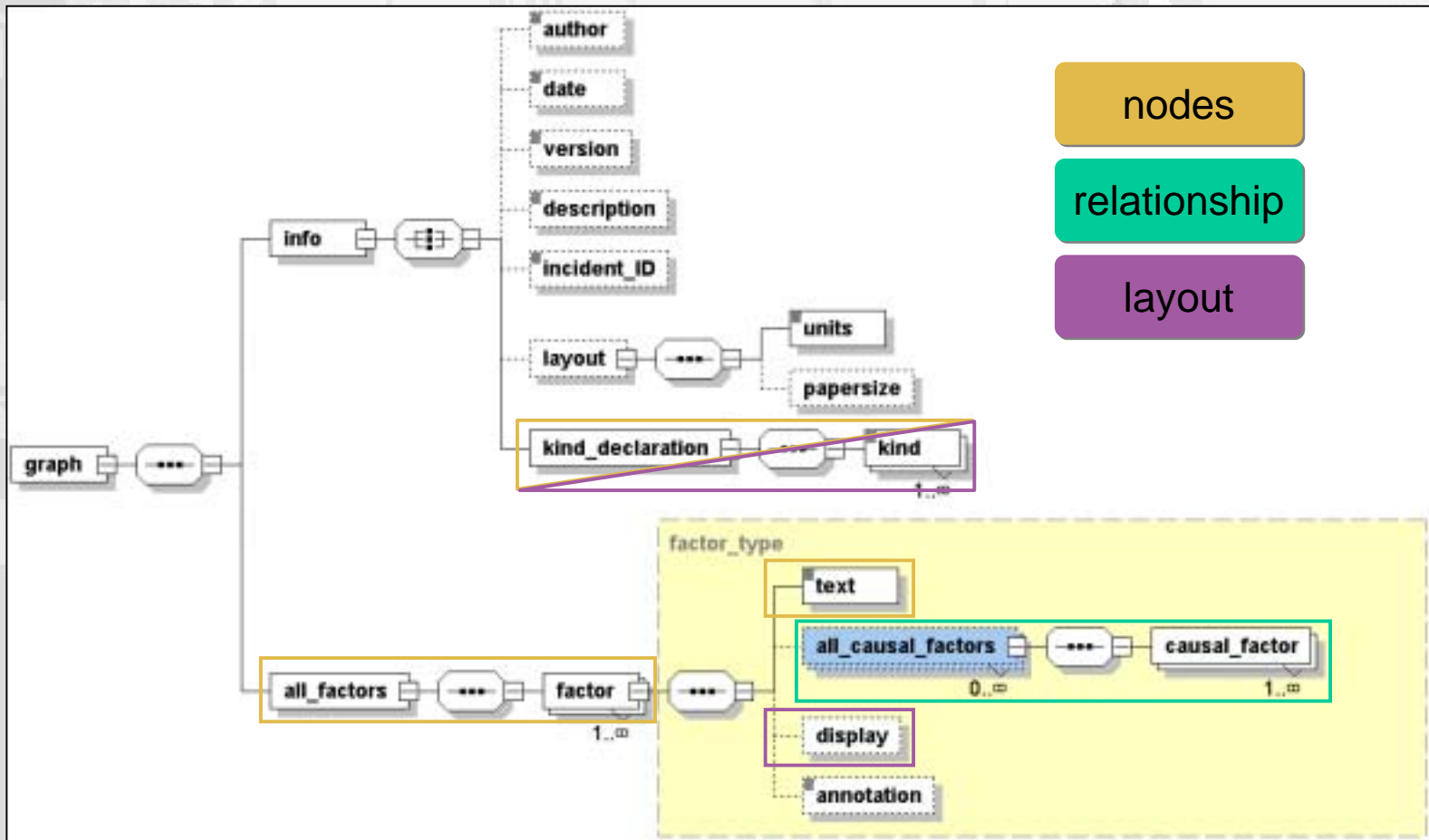
What information is contained in a WB-Graph?



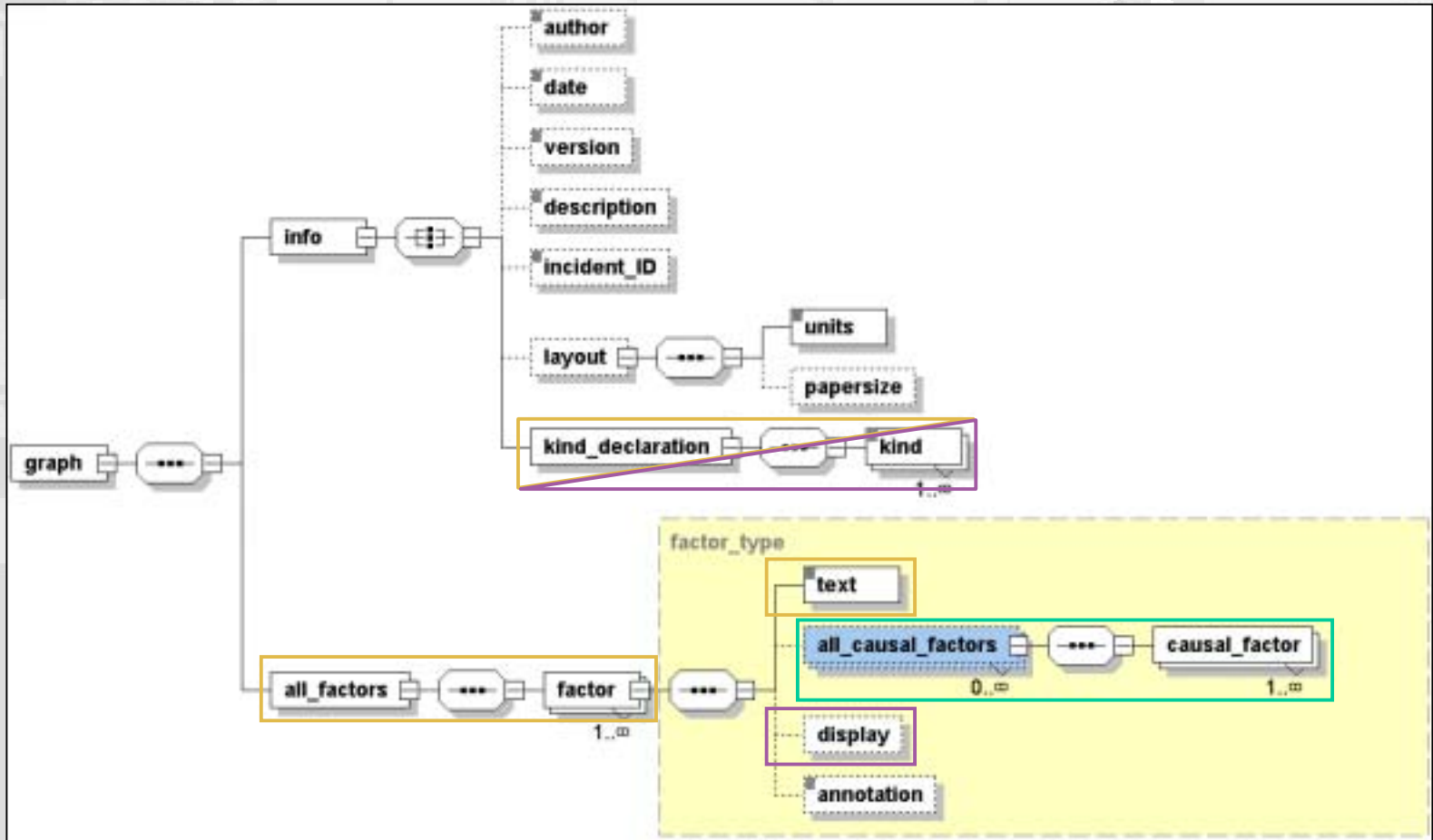
Information on the nodes

Information on their relationships

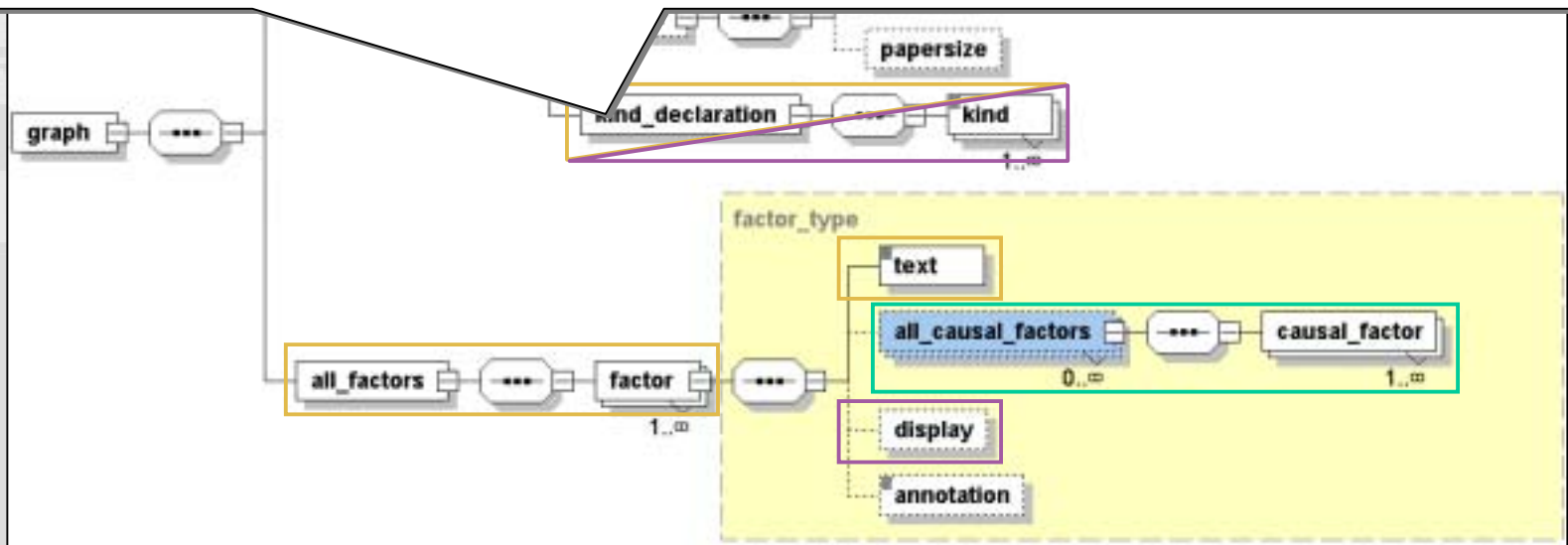
Information on the layout



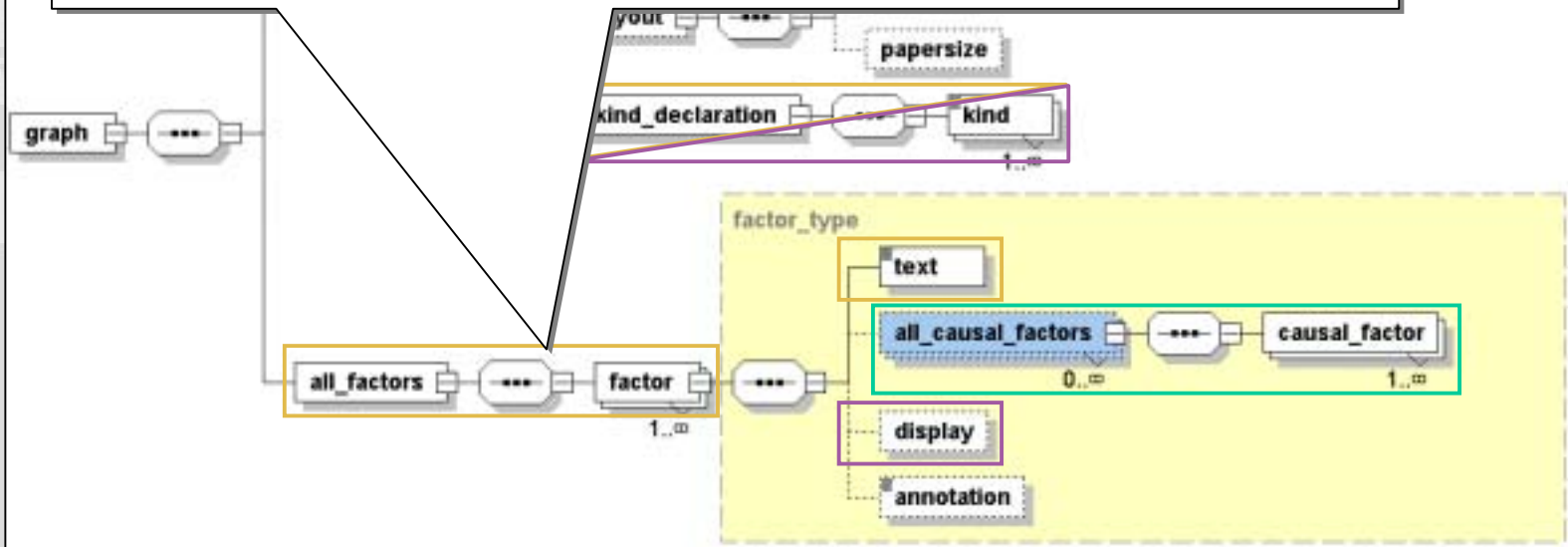
- Introduction
- Requirements
- Basic concepts
- **Realisation**
- Example

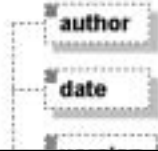


```
<kind_declaration>  
  <kind id="internal_event" shape="rectangle">an internal event</kind>  
  <kind id="internal_state" shape="hexagon">an internal state</kind>  
  <kind id="internal_process" shape="octagon">an internal process</kind>  
  <kind id="source_state" shape="hexagon">a source state</kind>  
  <kind id="source_unevent" shape="ellipse">a source un-event</kind>  
</kind_declaration>
```

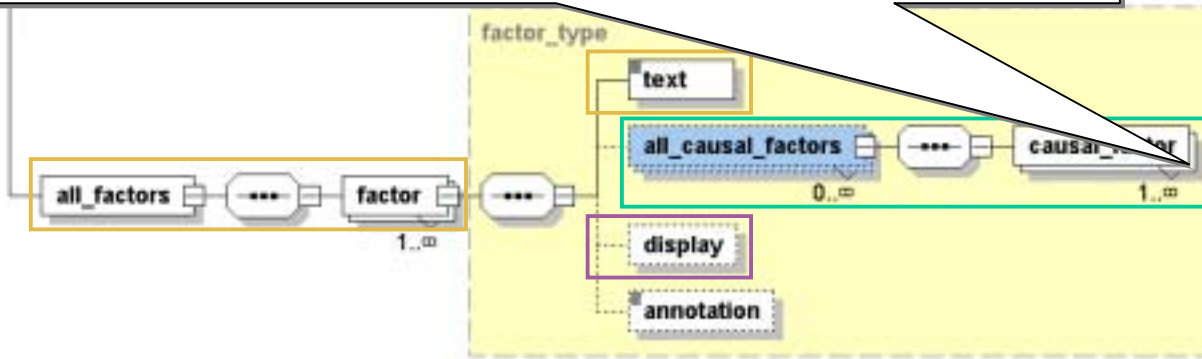


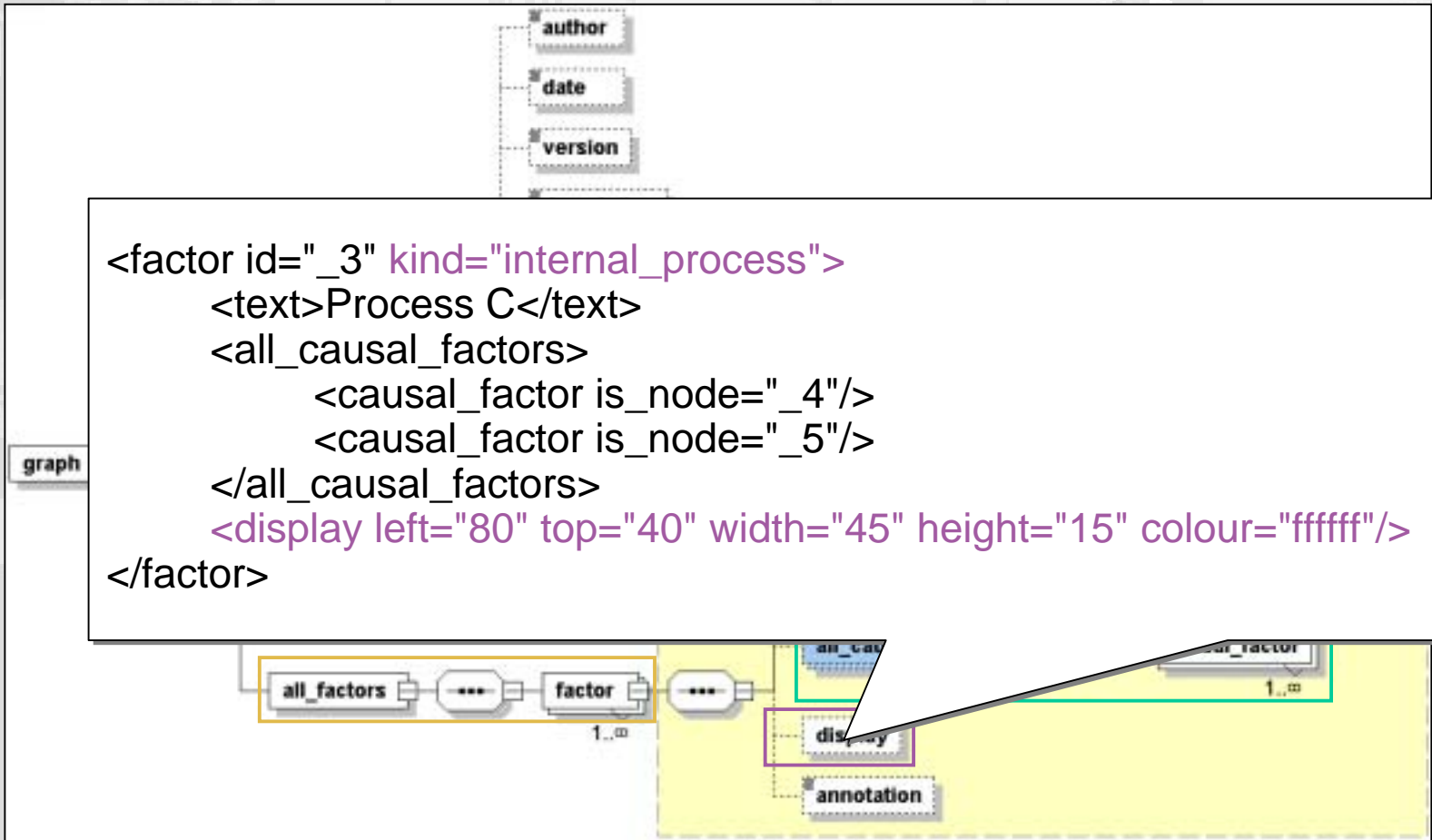
```
<all_factors>
  <factor id="_1" name="Node 001" kind="internal_event">
    <text>something bad happens</text>
  </factor>
  <factor id="_2" name="Node 002" kind="internal_state">
    <text>something even worse happens</text>
  </factor>
  ...
</all_factors>
```





```
<factor id="_3" kind="internal_process">  
  <text>Process C</text>  
  <all_causal_factors>  
    <causal_factor is_node="_4"/>  
    <causal_factor is_node="_5"/>  
  </all_causal_factors>  
</factor>
```





Additional features

ID checking

```
<factor id="_4" name="Node 001" kind="internal_event">
```

```
<causal_factor is_node="_4"/>
```

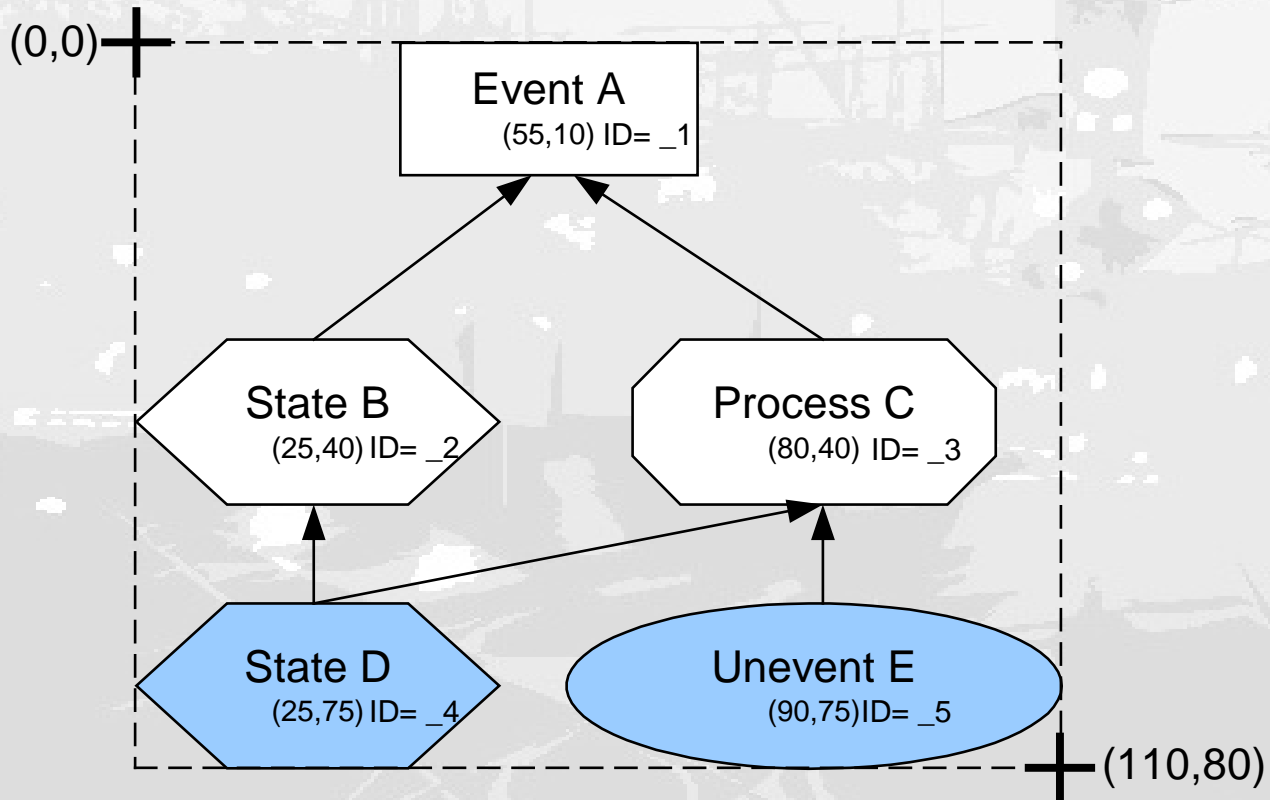
Arbitrary comments

```
<annotation type="text/html">
```

```
<![CDATA[<html><head><meta http-equiv="content-type"  
content="text/html; charset=ISO-8859-1">  
<title>Annotation</title></head><body>A more complex  
annotation...<br></body></html>]]>
```

```
</annotation>
```

- Introduction
- Requirements
- Basic concepts
- Realisation
- **Example**



CausalML is available as a XML schema definition from the
CausalML Homepage:
<http://ivev8.ivev.bau.tu-bs.de/forschung/rca/CausalML/index.html>

Thank you for your attention!

Questions? Comments?