XML and d2d, Writing Structures in the Flow

Markus Lepper[a] Baltasar Trancón y Widemann[ab]

[a]<semantics/> GmbH

[b]Universität Ilmenau

2002-2012

Contents

What is XML?

A "mark-up language".

I.e. an *open, standardized format*
for storing/tranmitting/exchanging *structured text*.

A "mark-up *meta*-language"
and a *family* of mark-up languages. (= "instantiations")

Infra-Structure is common.
("namespace", "entity", tag formats, DTD definition formalism, etc.)
Instances are domain-specific and *widely* varying.
(ThML vs. agroXML)
`http://www.ccel.org/ThML`
`http://www.agroxml.de`

What is XML?

A “mark-up language”.

I.e. an *open, standardized format*
for storing/tranmitting/exchanging *structured text*.

A “mark-up *meta*-language”
and a *family* of mark-up languages. (= “instantiations”)

Infra-Structure is common.
(“namespace”, “entity”, tag formats, DTD definition formalism, etc.)
Instances are domain-specific and *widely* varying.
(ThML vs. agroXML)
`http://www.ccel.org/ThML`
`http://www.agroxml.de`

What is XML?

A “mark-up language”.

I.e. an *open, standardized format*
for storing/tranmitting/exchanging *structured text*.

A “mark-up *meta*-language”
and a *family* of mark-up languages. (= “instantiations”)

Infra-Structure is common.
(“namespace”, “entity”, tag formats, DTD definition formalism, etc.)
Instances are domain-specific and *widely* varying.
(ThML vs. agroXML)
`http://www.ccel.org/ThML`
`http://www.agroxml.de`

What is XML?

A ~~"mark-up language"~~.

I.e. an *open, standardized format*
for storing/tranmitting/exchanging *structured text*.

A "mark-up *meta*-language"
and a *family* of mark-up languages. (= "instantiations")

Infra-Structure is common.
("namespace", "entity", tag formats, DTD definition formalism, etc.)
Instances are domain-specific and *widely* varying.
(ThML vs. agroXML)
`http://www.ccel.org/ThML`
`http://www.agroxml.de`

Pros for XML

- Simple basic idea
- Well-proven, basically from the 70s ("SGML")
- Extendable, combinable,
  many instantiations for standard purposes
  and most diverse domains (`CowML`, `TheoML`)
- Precisely and openly defined text formats
- Data-bases inspectable without any software
- Standardized, free, mature, reliable and open source
  tools for processing
- Suited for fully formalized
  and semi-formal documents

Cons against for XML

- Historically grown and very idiosyncratic stacking of "layers" and exceptions
- Oftenly abused as marketing "Buzz Word"
- Hardly parametrizable (in practice!)
- No support for incremental formalization
- Hardly readable, not writable (for HUMANS !-)

Basic Ideas of XML (/SGML)

```
    Quakenbrück, 20.12.2012
Sehr geehrte Frau Bauer.

Erfreulicherweise sind die
Laborbefunde der Proben Ihrer
Kuh Karoline, QNr 1789872334-3,
vom 11.11.2011 allesamt negativ.
Grüße auch an die Kinder und
Knechte.
Mit freundlichen Grüßen
```

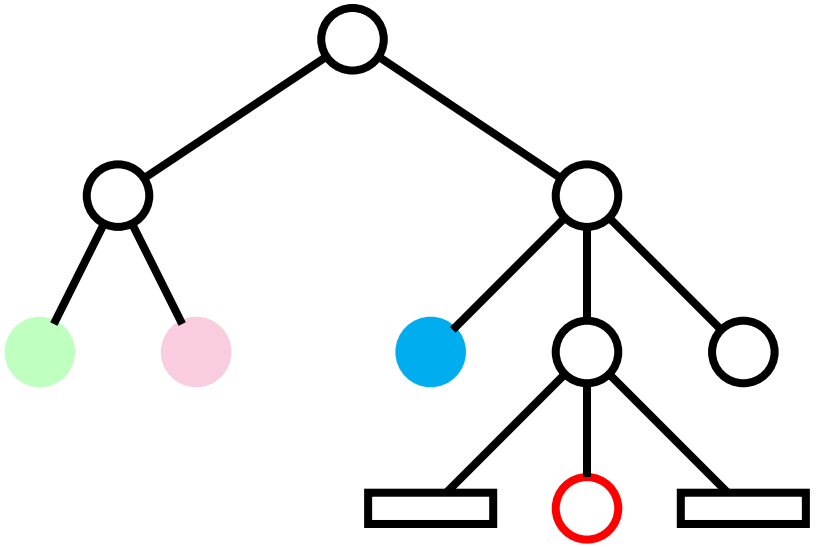
XMLs (/SGMLs) view to text:

- Texts are *tree structures*
- Nodes are *typed* (“tagged”)
- Text structures are *defined*

Basic Ideas of XML (/SGML)

```
   Quakenbrück, 20.12.2012
Sehr geehrte Frau Bauer,

Erfreulicherweise sind die
Laborbefunde der Proben Ihrer
Kuh Karoline, QNr 1789872334-3,
vom 11.11.2011 allesamt negativ.
Grüße auch an die Kinder und
Knechte.
Mit freundlichen Grüßen
```



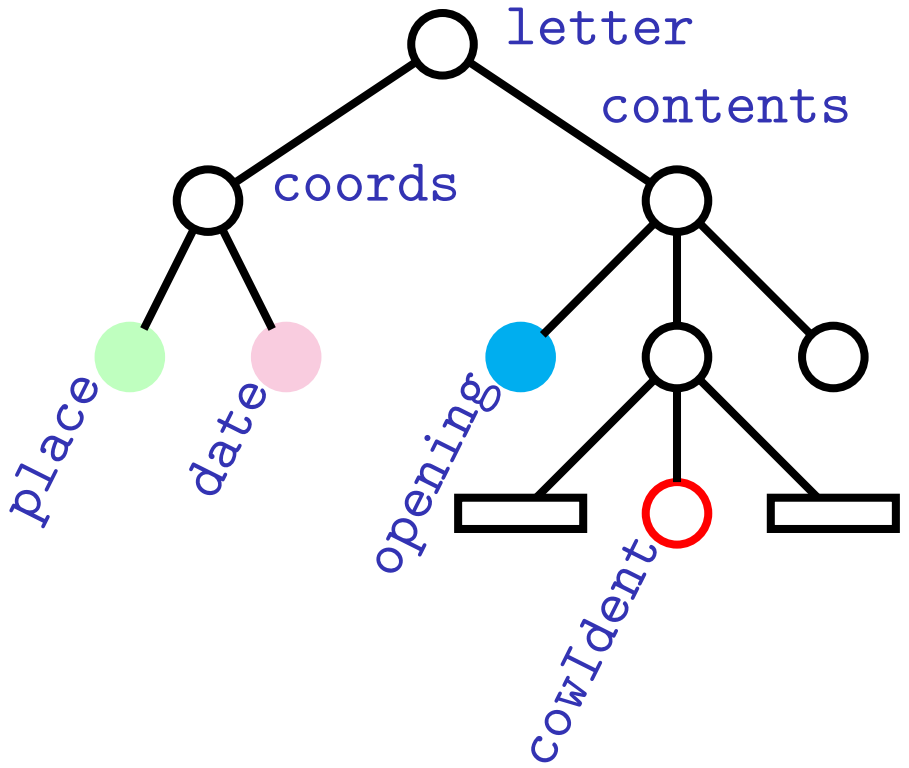
XMLs (/SGMLs) view to text:

- Texts are *tree structures*
- Nodes are *typed* (“tagged”)
- Text structures are *defined*

Basic Ideas of XML (/SGML)

```
Quakenbrück, 20.12.2012
Sehr geehrte Frau Bauer,

Erfreulicherweise sind die
Laborbefunde der Proben Ihrer
Kuh Karoline, QNr 1789872334-3,
vom 11.11.2011 allesamt negativ.
Grüße auch an die Kinder und
Knechte.
Mit freundlichen Grüßen
```



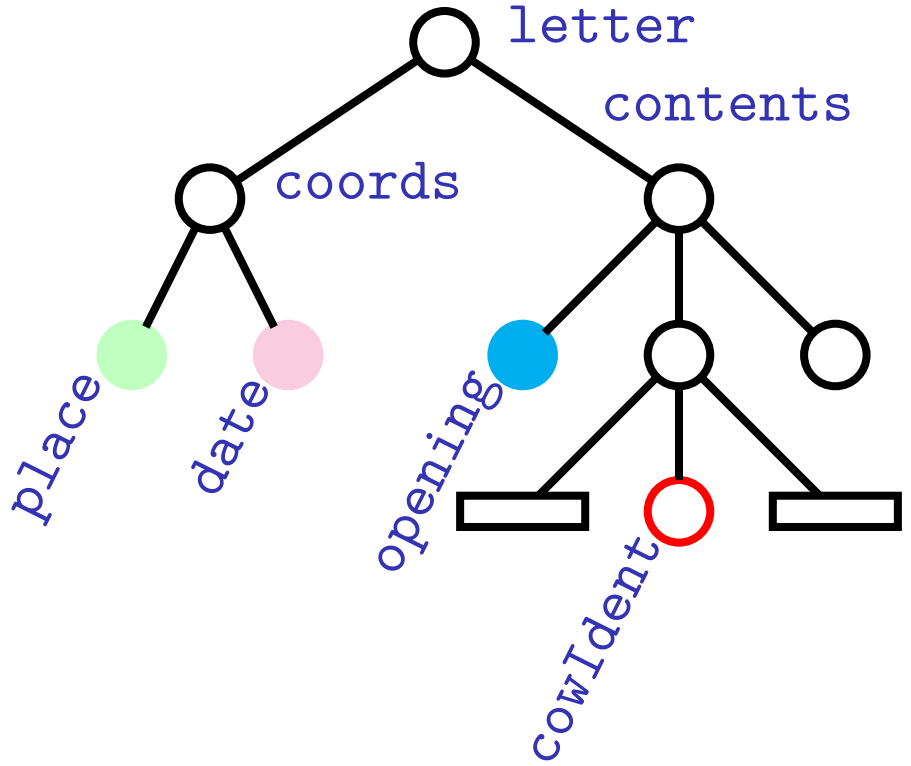
XMLs (/SGMLs) view to text:

- Texts are *tree structures*
- Nodes are *typed* (“tagged”)
- Text structures are *defined*

Basic Ideas of XML (/SGML)

```
Quakenbrück, 20.12.2012
Sehr geehrte Frau Bauer,

Erfreulicherweise sind die
Laborbefunde der Proben Ihrer
Kuh Karoline, QNr 1789872334-3,
vom 11.11.2011 allesamt negativ.
Grüße auch an die Kinder und
Knechte.
Mit freundlichen Grüßen
```



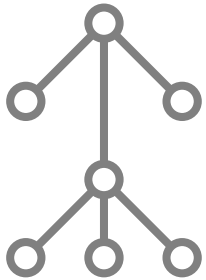
XMLs (/SGMLs) view to text:

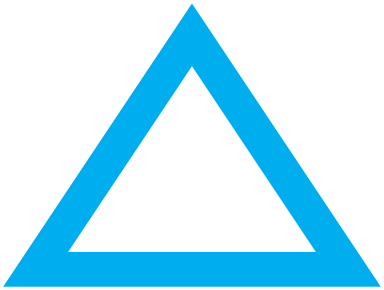
- Texts are *tree structures*
- Nodes are *typed* ("tagged")
- Text structures are *defined*

```
letter ::= coords, contents
coords ::= place & date
contents::= opening?, body, closing
body ::= (#PCDATA|cowIdent|date)*
```

Aggregate States of an XML “document object”

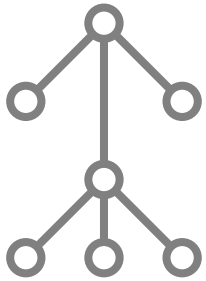
Tree as a concept

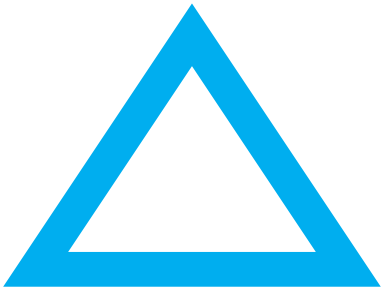


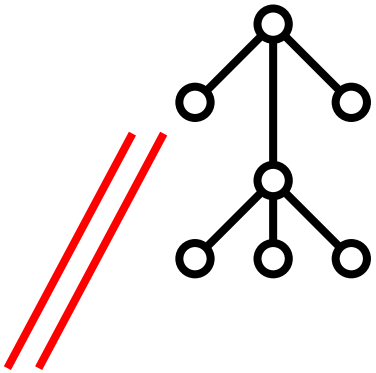


Aggregate States of an XML "document object"

Tree as a concept



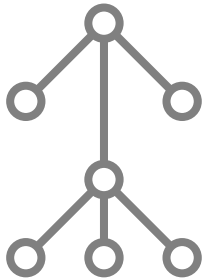


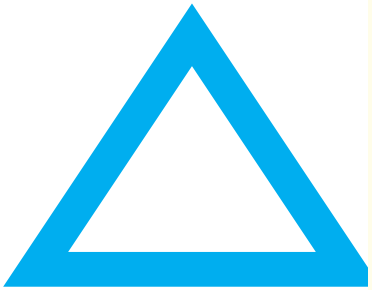


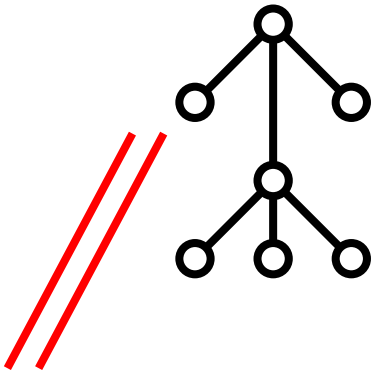
Tree as a concrete data structure
(for programmed manipulation!)

Aggregate States of an XML "document object"

Tree as a concept







```
<letter>
  <coords>
    <place>Quakenbrück</place>
    <date>20.12.2012</date>
  </coords>
  <contents>
    <opening>
    Sehr geehrte Frau Bauer!
    </opening>
    <body>
       ( ...  etc ...)
    </body>
  </contents>
</letter>
```

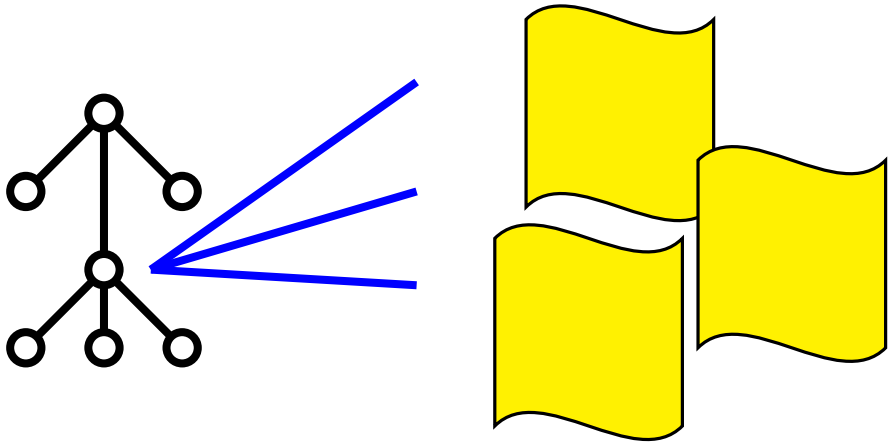
Tree as a concrete data structure
(for programmed manipulation!)

"Serialized" =
Text stored in a disk file

Use Cases with XML

- Technical Data / Fully Formal Data-bases
- Semi-Formal Data-bases
- Mark-Up of existing texts
- Single Source $\Rightarrow$ Multiple Renderings

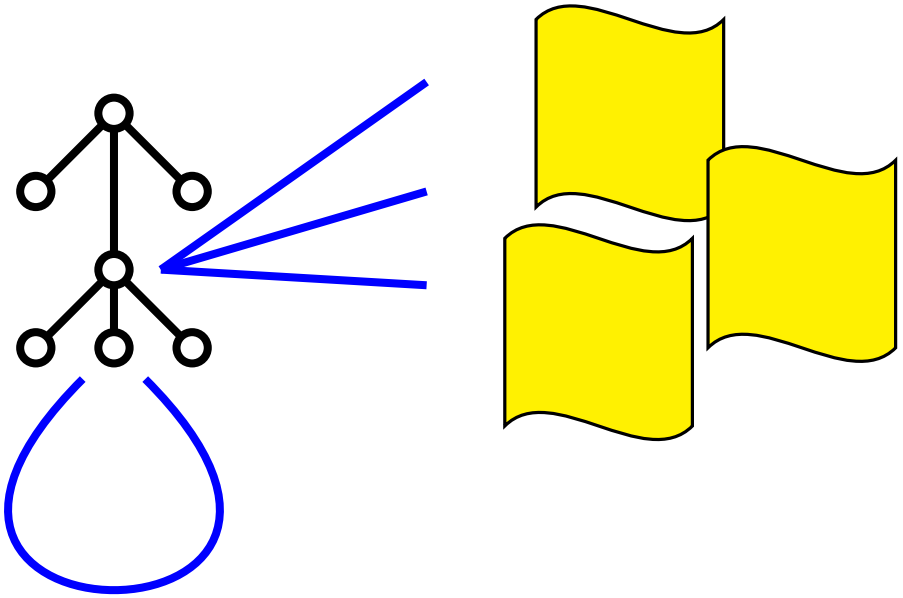
Use Cases with XML



Benefits of an XML computer model:

- Single source – multiple outputs
- Automated Transformations
- Automated Analyses

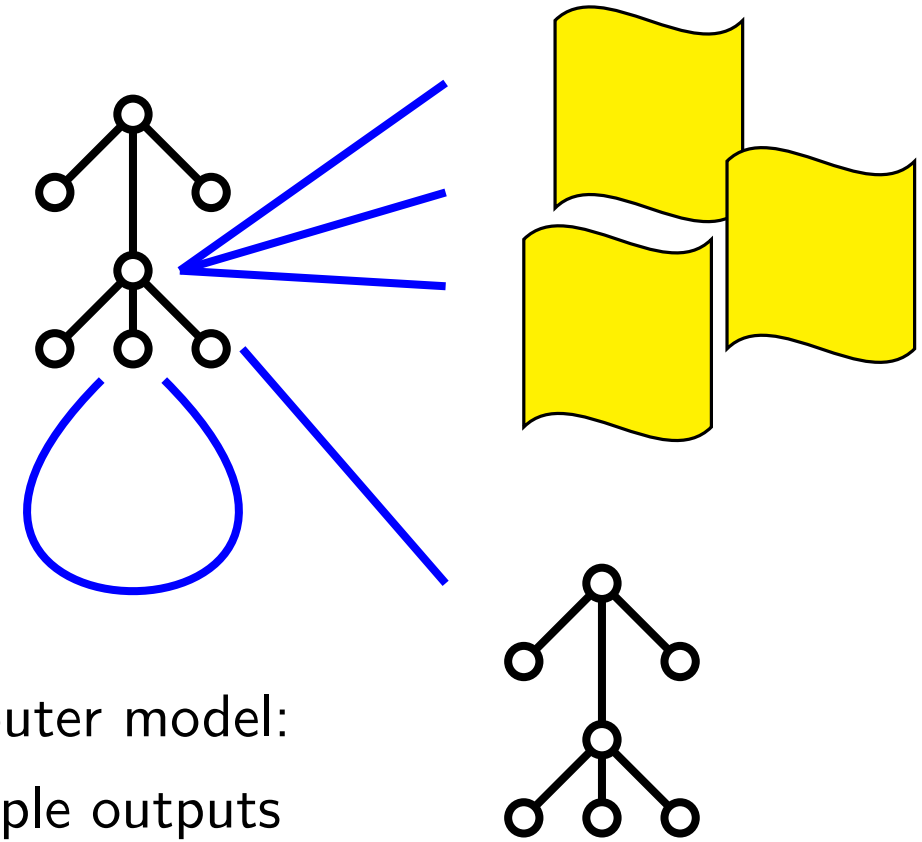
Use Cases with XML



Benefits of an XML computer model:

- Single source – multiple outputs
- Automated Transformations
- Automated Analyses

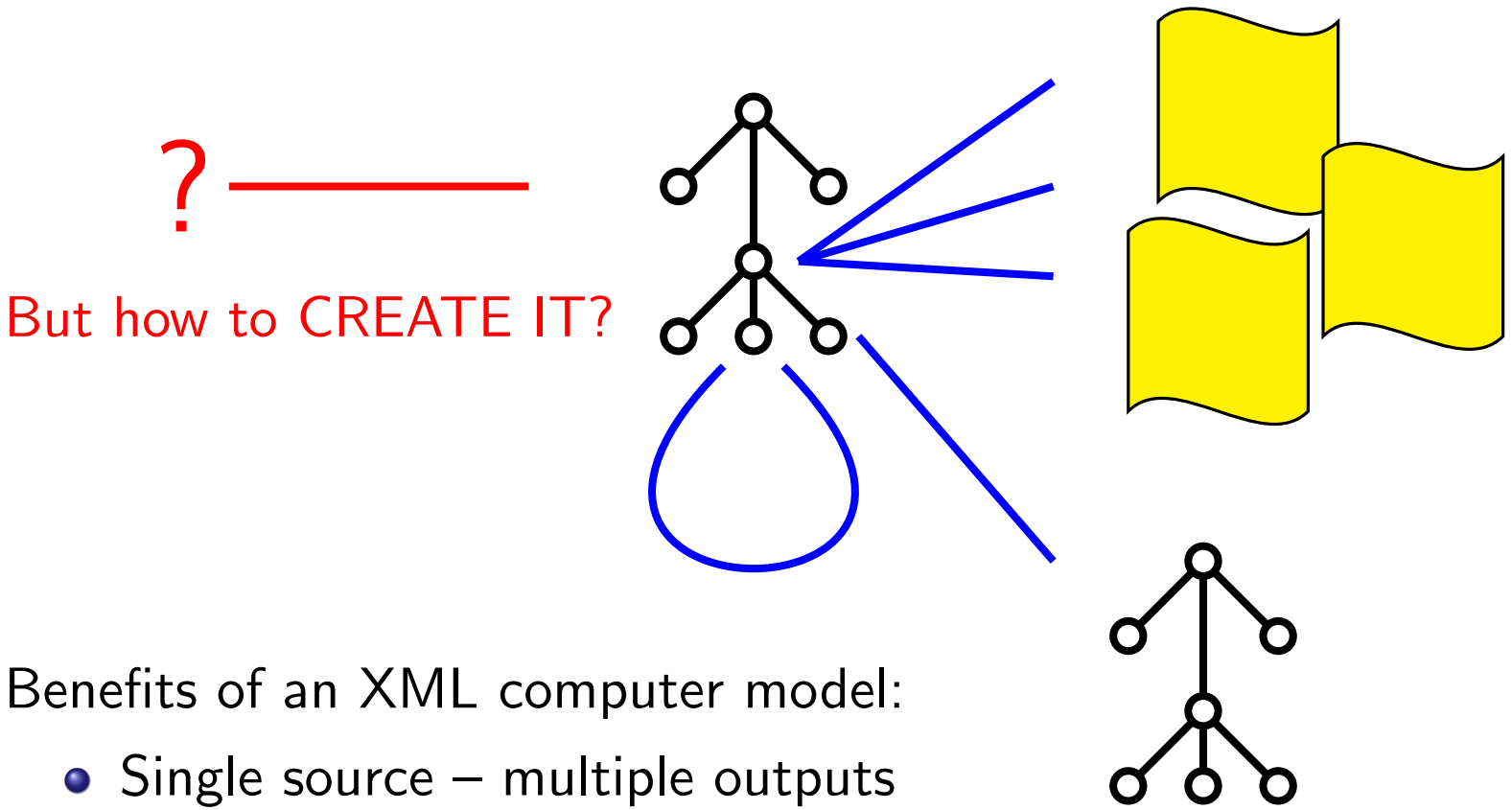
Use Cases with XML



Benefits of an XML computer model:

- Single source – multiple outputs
- Automated Transformations
- Automated Analyses

Use Cases with XML



Benefits of an XML computer model:

- Single source – multiple outputs
- Automated Transformations
- Automated Analyses

What Is d2d?

What is d2d?

stands for Direct Document Denotation

or for Directly "To" Document

spoken "tripple dee"

What Is d2d?

What is d2d?

stands for **D**irect **D**ocument **D**enotation

or for **D**irectly "To" **D**ocument

spoken "tripple dee"

What Is d2d?

- Notation of XML Documents, which can be
  "written in the flow"
- One character opening tags
- Inference of closing tags
- Fine-granular parsers, infering *all tags*
- Tooling: Parser with advanced error signalling
  (Syntax controlled editor, unparser, etc. in work)
- Works with standard ("dtd") and with proprietary ("ddf")
  document type definition format
- ddf: Instantiation and Parametrization by free rewriting
- Standard general-purpose document ddf
  (based on LaTeX, DocBook, etc.)

Text Example

```
#d2d text using formats:letter
  #coords
    #place Quakenbrück
    #date 20.12.2012
  #contents
    #opening
      Sehr geehrte Frau Bauer!
    #body
      Erfreulicherweise sind die
Laborbefunde Ihrer Kuh Karoline
#vet q1789872334-3, Probe
vom 12.12.2012
allesamt negativ.
   (...  etc ...)
#eof
```

```
<letter xmlns="http://???">
  <coords>
    <place>Quakenbrück</place>
    <date>20.12.2012</date>
  </coords>
  <contents>
    <opening>
  Sehr geehrte Frau Bauer!
    </opening>
    <body>
      Erfreulicherweise sind die
Laborbefunde Ihrer Kuh Karoline
<vet><kind>cow</kind>
     <ident>1789872334</ident>
     <checksum>3</checksum>
</vet>, Probe vom 12.120212
allesamt negativ.(...etc...)
    </body>
  </contents>
</letter>
```

Text Example

```
#d2d text using formats:letter
  #coords
    #place Quakenbrück
    #date 20.12.2012
  #contents
    #opening
      Sehr geehrte Frau Bauer!
    #body
      Erfreulicherweise sind die
Laborbefunde Ihrer Kuh Karoline
#vet q1789872334-3, Probe
vom 12.12.2012
allesamt negativ.
   (...  etc ...)
#eof
```

```
<letter xmlns="http://???">
  <coords>
    <place>Quakenbrück</place>
    <date>20.12.2012</date>
  </coords>
  <contents>
    <opening>
  Sehr geehrte Frau Bauer!
    </opening>
    <body>
      Erfreulicherweise sind die
Laborbefunde Ihrer Kuh Karoline
<vet><kind>cow</kind>
     <ident>1789872334</ident>
     <checksum>3</checksum>
</vet>, Probe vom 12.120212
allesamt negativ.(...etc...)
    </body>
  </contents>
</letter>
```

Text Example

```
#d2d text using formats:letter
  #coords
    #place Quakenbrück
    #date 20.12.2012
  #contents
    #opening
      Sehr geehrte Frau Bauer!
    #body
      Erfreulicherweise sind die
Laborbefunde Ihrer Kuh Karoline
#vet q1789872334-3. Probe
vom 12.12.2012
allesamt negativ.
   (...  etc ...)
#eof
```

```
<letter xmlns="http://???">
  <coords>
    <place>Quakenbrück</place>
    <date>20.12.2012</date>
  </coords>
  <contents>
    <opening>
  Sehr geehrte Frau Bauer!
    </opening>
    <body>
      Erfreulicherweise sind die
Laborbefunde Ihrer Kuh Karoline
<vet><kind>cow</kind>
     <ident>1789872334</ident>
     <checksum>3</checksum>
</vet>, Probe vom 12.120212
allesamt negativ.(...etc...)
    </body>
  </contents>
</letter>
```

Text Example

```
#d2d text using formats:letter
  #coords
    #place Quakenbrück
    #date 20.12.2012
  #contents
    #opening
      Sehr geehrte Frau Bauer!
    #body
      Erfreulicherweise sind die
Laborbefunde Ihrer Kuh Karoline
#vet q1789872334-3, Probe
vom 12.12.2012
allesamt negativ.
   (...  etc ...)
#eof
```

```
<letter xmlns="http://???">
  <coords>
    <place>Quakenbrück</place>
    <date>20.12.2012</date>
  </coords>
  <contents>
    <opening>
  Sehr geehrte Frau Bauer!
    </opening>
    <body>
      Erfreulicherweise sind die
Laborbefunde Ihrer Kuh Karoline
<vet><kind>cow</kind>
     <ident>1789872334</ident>
     <checksum>3</checksum>
</vet>, Probe vom 12.120212
allesamt negativ.(...etc...)
    </body>
  </contents>
</letter>
```

Text Example

incremental refinement!

```
#d2d text using formats:letter
  #coords
    #place Quakenbrück
    #date 20.12.2012
  #contents
    #opening
      Sehr geehrte Frau Bauer!
    #body
      Erfreulicherweise sind die
Laborbefunde Ihrer Kuh Karoline
#vet q1789872334-3, Probe
vom 12.12.2012
allesamt negativ.
   (...  etc ...)
#eof
```

```
<letter>
  <coords>
    <place>Quakenbrück</place>
    <date><day>20</day>
          <month>12</month>
          <year>2012</year>
    </date>
  </coords>
  <contents>

   (...  etc ...)
    </body>
  </contents>
</letter>
```

Text Example

incremental refinement!

```
#d2d text using formats:letter
  #coords
    #place Quakenbrück
    #date 20.12.2012
  #contents
    #opening
      Sehr geehrte Frau Bauer!
    #body
      Erfreulicherweise sind die
Laborbefunde Ihrer Kuh Karoline
#vet q1789872334-3, Probe
vom #date 12.12.2012
allesamt negativ.
   (...  etc ...)
#eof
```

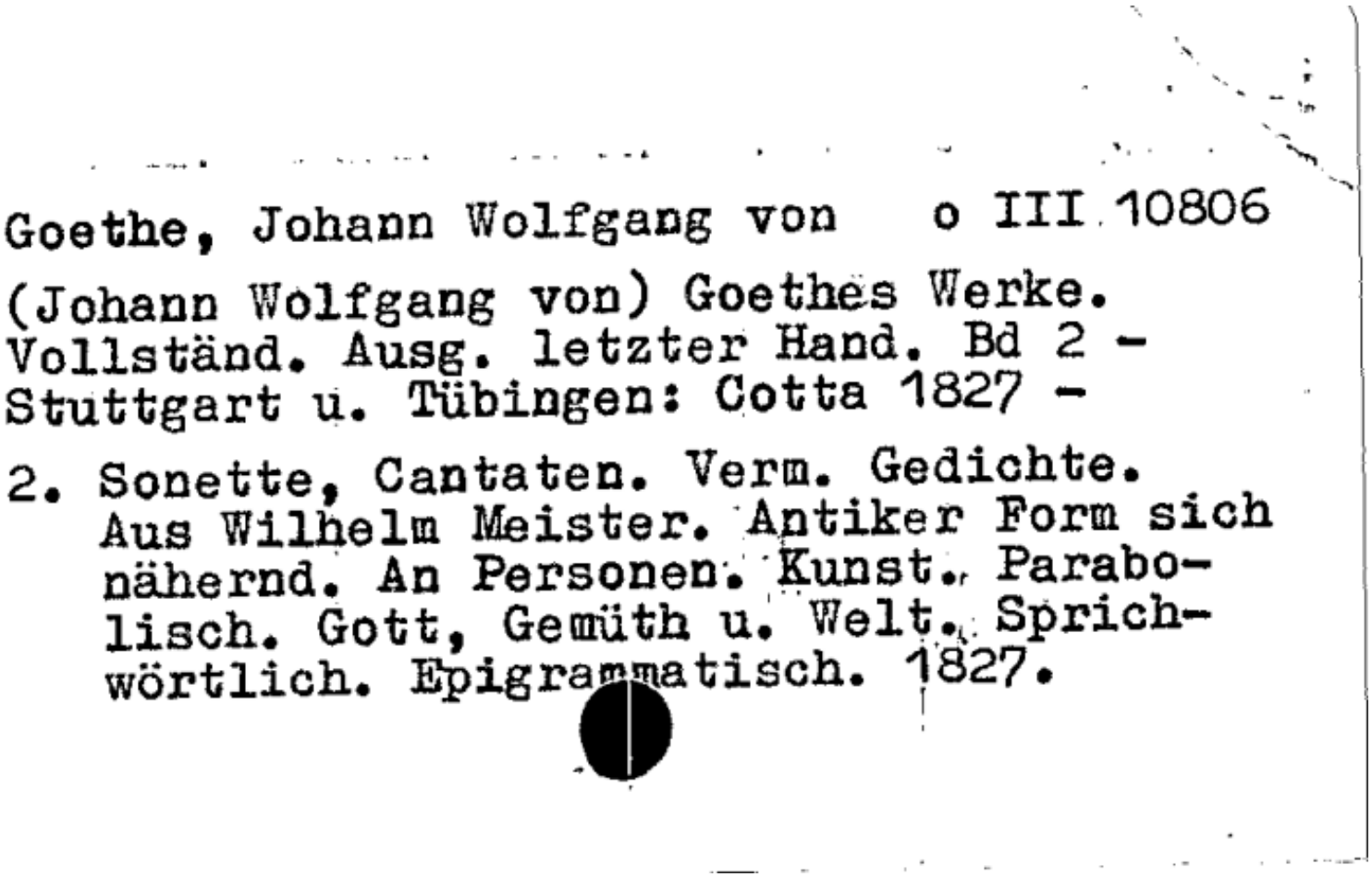
```
<letter>
  <coords>
    <place>Quakenbrück</place>
    <date><day>20</day>
          <month>12</month>
          <year>2012</year>
    </date>
  </coords>
  <contents>

    Probe vom
    <date><day>12</day>
          <month>12</month>
          <year>2012</year>,
allesamt negativ.
   (...  etc ...)
    </body>
  </contents>
</letter>
```

Use Cases with d2d

- Incremental mark-up of legacy documents
- Authoring of (non-fiction) texts
- . . . for further processing,
- . . . and into multiple back-ends.
- Front-end format for domain-specific languages.
  (E.g. catalog entries, abstracts, surveys, semantic maps)

Incremental Formalization – Photo



Goethe, Johann Wolfgang von o III 10806

(Johann Wolfgang von) Goethes Werke.
Vollständ. Ausg. letzter Hand. Bd 2 -
Stuttgart u. Tübingen: Cotta 1827 -

2. Sonette, Cantaten. Verm. Gedichte.
Aus Wilhelm Meister. Antiker Form sich
nähernd. An Personen. Kunst. Parabo-
lisch. Gott, Gemüth u. Welt. Sprich-
wörtlich. Epigrammatisch. 1827.

Incremental Formalization – OCR

```
Goethe, Johann Wolfgang von o III 10806

(Johann Wolfgang von) Goethes Werke.
VOllstand. Ausg. letzter Hand. Bd 2 -
Stuttgart u. Tübingen: Cotta 1827 -

2. Sonette, Cantaten, Verm. Gedichte.
   Aus Wilhelm Meister. Antiker Form sich
   nähernd. An Personen. Kust. Parabo-
   lisch. Gott, Gemüth u. Welt. Sprich-
   wörtlich. EpigraMMatisch. 1827.
```

Incremental Formalization – Text Type Declared

```
#d2d text using formats:katalog.kartei_vor_1946
Goethe, Johann Wolfgang von o III 10806

(Johann Wolfgang von) Goethes Werke.
VOllstand.  Ausg.  letzter Hand.  Bd 2 -
Stuttgart u.  Tübingen:  Cotta 1827 -

2.  Sonette, Cantaten, Verm.  Gedichte.
   Aus Wilhelm Meister.  Antiker Form sich
   nähernd.  An Personen.  Kust.  Parabo-
   lisch.  Gott, Gemüth u.  Welt.  Sprich-
   wörtlich.  EpigraMMatisch.  1827.
#eof
```

Incremental Formalization – First Mark-Up

```
#d2d text using formats:katalog.kartei_vor_1946
#author Goethe, Johann Wolfgang von #sign o III 10806#/

(Johann Wolfgang von) Goethes Werke.
VOllstand. Ausg. letzter Hand. Bd 2 -
Stuttgart u. Tübingen: Cotta 1827 -

2. Sonette, Cantaten, Verm. Gedichte.
   Aus Wilhelm Meister. Antiker Form sich
   nähernd. An Personen. Kust. Parabo-
   lisch. Gott, Gemüth u. Welt. Sprich-
   wörtlich. EpigraMMatisch. 1827.
#eof
```

Incremental Formalization – More Mark-Up

```
#d2d text using formats:katalog.kartei_vor_1946
#author Goethe, Johann Wolfgang von #sign o III 10806#/

(Johann Wolfgang von) #title Goethes Werke.
VOllstand. Ausg. letzter Hand. #vol 2 // -
#place Stuttgart u. Tübingen
     #publisher Cotta#year 1827 #/ -
2. Sonette, Cantaten, Verm. Gedichte.
   Aus Wilhelm Meister. Antiker Form sich
   nähernd. An Personen. Kust. Parabo-
   lisch. Gott, Gemüth u. Welt. Sprich-
   wörtlich. EpigraMMatisch. 1827.
#eof
```

Call for proposals !-)

We are interested in ideas for application and/or co-operation!

http://www.bandm.eu/metatools
post@markuslepper.eu
baltasar@trancon.de