

Interdisziplinäre Anwendungen Raumbezogener Informationstechnik

Klausurvorbereitung

- Themen (Auszug)
 - Datenbanken allgemein
 - Wozu dient eine Datenbank?
 - Welche Eigenschaften hat eine Datenbank?
 - Was ist der unterschied zwischen einer relationalen Datenbank und einem Triple Store?
 - Triple Stores
 - Daten im RDF-Format (d.h. als Triples) darstellen
 - Datenbank abfragen mit SPARQL und Datenbank modifizieren mit SPARQL Update
 - Ontologien entwerfen mit OWL und RDF-Schema

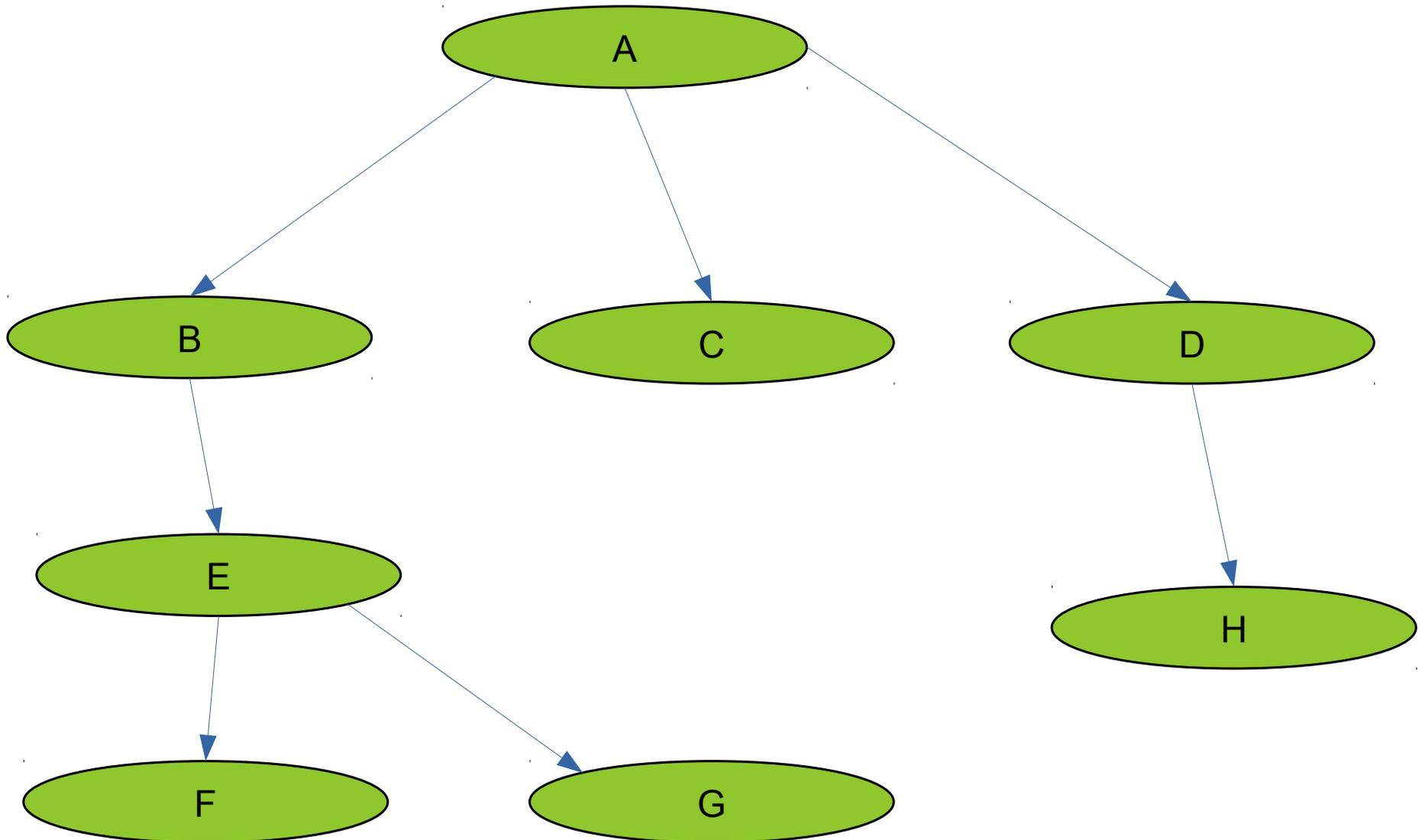
- Themen (Auszug)
 - RDF
 - Was ist die Idee des Semantic Web?
 - Was ist ein URI?
 - Wie werden Daten in einem Triple Store gespeichert?
 - SPARQL und SPARQL Update
 - Was tut ein Triple Store bei welcher Art von Abfrage?
 - OWL und RDFS
 - Web Ontology Language (OWL)
 - Resource Description Framework Schema (RDFS)

XML

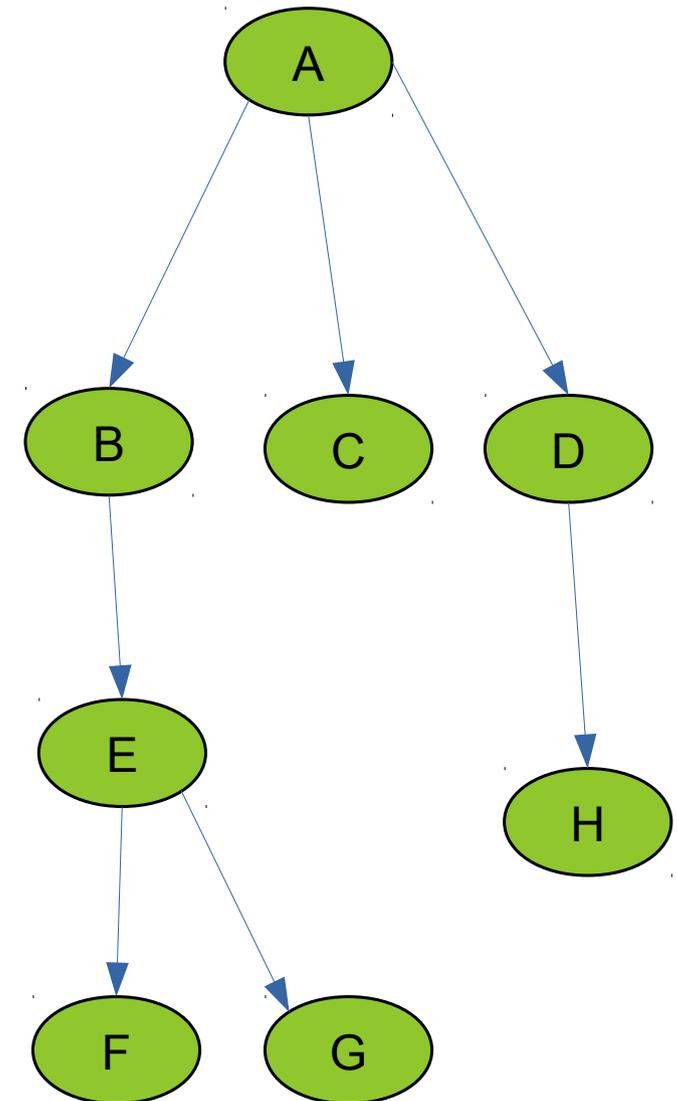
Extensible Markup Language

- Auszeichnungssprache, um Daten mit Baumstruktur darzustellen
 - Element / Knoten
 - Tag
 - Start-Tag `<name>`
 - End-Tag `</name>`
 - Empty-Element-Tag `<name />`
 - Attribut
 - Start-Tag `<tagname attributname="attributwert">`
 - Empty-Element-Tag `<tagname attributname="attributwert" />`

Beispiel

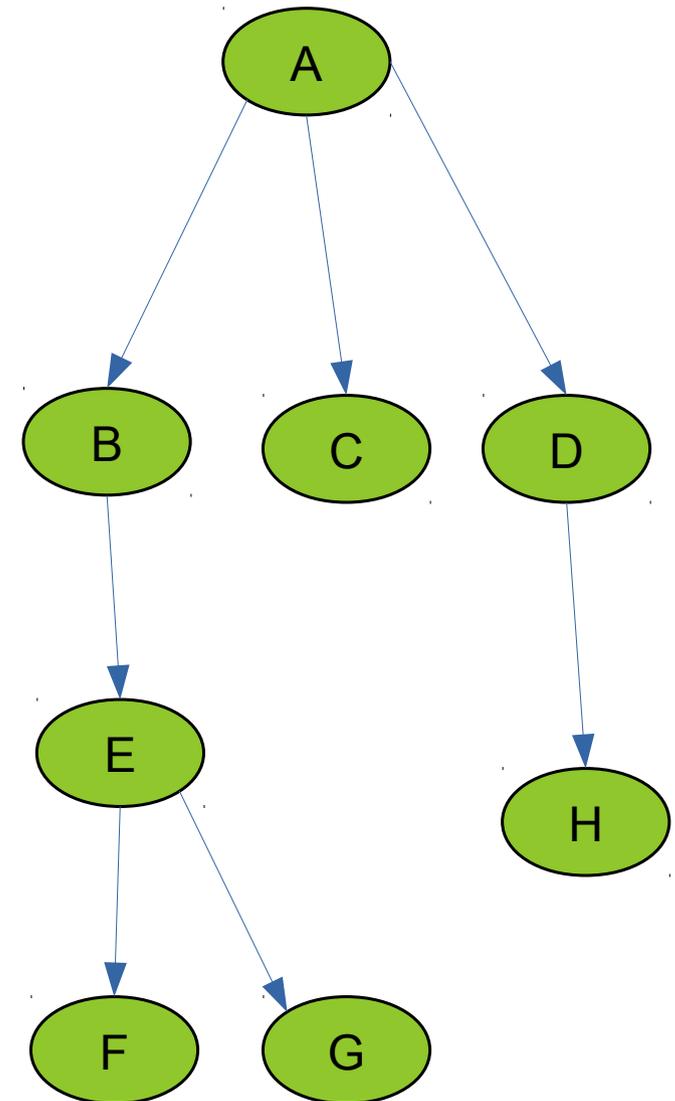


Beispiel



Beispiel

```
<A>  
  <B>  
    <E>  
      <F />  
      <G />  
    </E>  
  </B>  
  <C />  
  <D>  
    <H />  
  </D>  
</A>
```



RDF

Resource Description Framework

RDF Turtle

- RDF-Daten als Liste von je 3 URLs darstellen

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
```

```
@prefix ex: <http://example.de/> .
```

```
ex:A ex:P1 ex:X .
```

```
ex:A ex:P2 ex:Y .
```

oder ohne namespace

```
<http://example.de/A> <http://example.de/P1>
```

```
<http://example.de/X> .
```

```
<http://example.de/A> <http://example.de/P2>
```

```
<http://example.de/Y> .
```

RDF Turtle

- RDF-Daten als Liste von je 3 URLs darstellen mit Abkürzung

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
```

```
@prefix ex: <http://example.de/> .
```

```
ex:A ex:P1 ex:X ;  
     ex:P2 ex:Y .
```

oder ohne namespace mit Abkürzung

```
<http://example.de/A>  
  <http://example.de/P1> <http://example.de/X> ;  
  <http://example.de/P2> <http://example.de/Y> .
```

- RDF-Daten als XML-Dokument darstellen

```
<?xml version="1.0"?>
```

```
<rdf:RDF
```

```
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-  
syntax-ns#"
  xmlns:ex="http://example.de/">
```

```
  <rdf:Description rdf:about="http://example.de/A">
```

```
    <ex:P1 rdf:resource="http://example.de/X" />
```

```
    <ex:P2 rdf:resource="http://example.de/Y" />
```

```
  </rdf:Description>
```

```
</rdf:RDF>
```

HTML

HyperText Markup Language

- XML-basiertes Format, um Webseiten zu beschreiben

```
<!DOCTYPE html>
```

```
<html>
```

```
  <head>
```

```
    Hier stehen allgemeine Infos zur Seite
```

```
  </head>
```

```
  <body>
```

```
    Hier stehen die Elemente der Seite
```

```
  </body>
```

```
</html>
```

- Häufig verwendete Tags (innerhalb von `<body>`)
 - Strukturierungseinheiten `<div>`
 - Überschriften `<h1>` `<h2>` ...
 - Absätze `<p>`
 - Links `<a>`
 - Bilder ``
 - Tabellen `<table>`
 - Eingabefelder `<input>`

RDFa in HTML

- RDFa = RDF in Attributes

```
<!DOCTYPE html>
<html xmlns:ex="http://example.de/">

  <head> ... </head>

  <body>
    <tagname about="ex:A">
      <... property="ex:P1" resource="ex:X" />
      <... property="ex:P2" resource="ex:Y" />
    </tagname>
  </body>

</html>
```

Besprechung der Übung