14.05.2024, Fabian Neureiter

The X(ML) Files

How to handle XML files in the Oracle Database

code of change

Hyand

Our key facts

Hyand

Germany

- Brunswick
- Ratingen
- Hamburg
- Dortmund
- Cologne
- Frankfurt
- Munich
- Berlin

Poland

- Warsaw

Lithuania

- Vilnius
- Kaunas

Romania

- Cluj-Napoca

India

- Pune

850+
Employees

150+
Customers

110+
million €
turnover

whoami

Fabian Neureiter
Oracle APEX Developer

Part of the JavaScript Team @ Hyand

Certified Tester, specialized in automated UI Testing

Lead Developer LCT - Low Code Testing

Why XML?

Why XML – Common Questions/Statements

XML is old and outdated!

Why use XML when there is JSON?!

Nobody uses XML anymore!

Technologies using XML

Introduction – Technologies using XML – Example SVG

```
<?xml version="1.0" encoding="utf-8"?>
<!-- Generator: Adobe Illustrator 27.5.0, SVG Export Plug-In . SVG Version: 6.00 Build 0)  -->
<svg version="1.1" id="Ebene_1" xmlns="http://www.w3.org/2000/svg"
    xmlns:xlink="http://www.w3.org/1999/xlink" x="0px" y="0px"
    viewBox="0 0 214.74 36.61" style="enable-background:new 0 0 214.74 36.61;" xml:space="preserve">
<style type="text/css">
    .st0{fill:#E30613;}
</style>
<g>
    <g>
        <path class="st0" d="M212.65,0h-33.66c-10.44,0-18.3,8.59-
18.3,18.27s7.94,18.31,18.3,18.31h26.59c0,0,8.2,0.08,9.16-9V17.04
            h-21.25l-5.84,6.11h21.06v2.97c0,0,0.31,4-3.3,4h-25.96c-7.03,0-11.85-6-11.85-11.85c0-5.85,4.79-
11.85,11.85-11.85h27.13
            L212.65,0z"/>
        <path class="st0" d="M126.88,23.42h17.72l-8.81-15.44l-16.59,28.63h-7.54l20.17-34.37c0.86-1.38,2.32-
2.24,3.94-2.24
            c1.59,0,3.04,0.82,3.89,2.19l20.26,34.42h-7.81l-4.05-7.12H132.6L126.88,23.42z"/>
        <path class="st0" d="M104.11,18.3c0,6.55-5.3,11.85-11.85,11.85H74.9c-6.54,0-11.85-5.3-11.85-11.85
            c0-6.54,5.31-11.85,11.85-11.85l17.35,0C98.8,6.45,104.11,11.76,104.11,18.3 M111.03,18.3c0-10.11-8.19-
18.3-18.3-18.3h-18.3
            c-10.11,0-18.3,8.19-
18.3,18.3c0,10.11,8.2,18.3,18.3,18.3l18.3,0.01C102.84,36.61,111.03,28.42,111.03,18.3"/>
        <path class="st0" d="M44.08,18.3c0,6.55-5.3,11.85-
11.85,11.85H6.83V6.45l25.4,0C38.78,6.45,44.08,11.76,44.08,18.3 M51.01,18.3
            C51.01,8.19,42.81,0,32.71,0H0v36.6l32.71,0.01C42.81,36.61,51.01,28.42,51.01,18.3"/>
    </g>
</g>
</svg>
```

DOAG

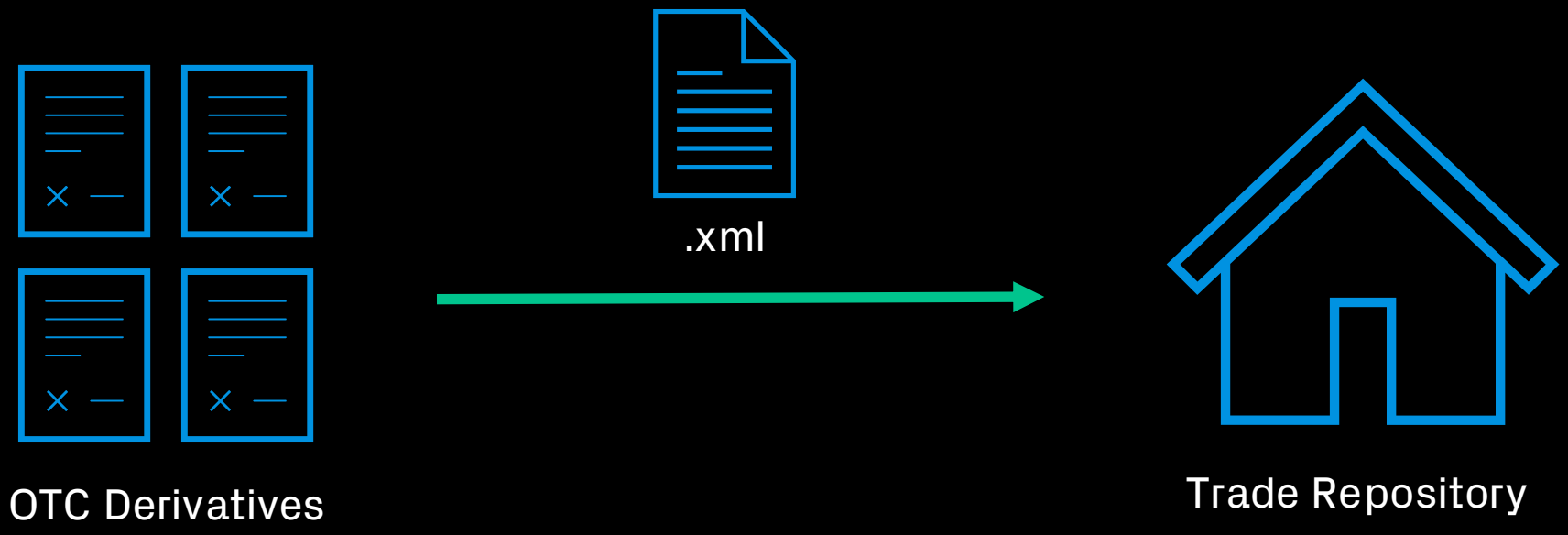
Why XML?

My Use Case

Who uses XML – My Use Case

AP/Richard Drew

EMIR
European Market Infrastructure Regulation



Sample auth.030 file

```xml
<?xml version="1.0" encoding="UTF-8"?>
<message xmlns:auth="urn:iso:std:iso:20022:tech:xsd:auth.030.001.01">
  <AppHdr>
    <!-- Header information -->
    <Fr><!-- Sender identification --></Fr>
    <To><!-- Recipient identification --></To>
    <BizMsgIdr><!-- Unique message identifier --></BizMsgIdr>
    <MsgDefIdr>auth.030.001.01</MsgDefIdr>
    <CreDt><!-- Creation timestamp --></CreDt>
  </AppHdr>

  <Document>
    <TRAuthstnRpt>
      <!-- Trade Repository Authorization Report -->
      <RptHdr>
        <!-- Report header details -->
        <RptId><!-- Report identifier --></RptId>
        <RptDt><!-- Report date --></RptDt>
        <RptSts><!-- Status information --></RptSts>
      </RptHdr>

      <TRData>
        <!-- Transaction and counterparty details -->
        <CtrPty>
          <!-- Counterparty information -->
          <RptgCtrPty><!-- Reporting counterparty details --></RptgCtrPty>
          <OthrCtrPty><!-- Other counterparty details --></OthrCtrPty>
        </CtrPty>

        <TxData>
          <!-- Transaction details -->
          <UnqTxIdr><!-- UTI: Unique Transaction Identifier --></UnqTxIdr>
          <TxDtls><!-- Contract details, notional, dates, etc. -->
</TxDtls> <ValtnDtls><!-- Valuation information --></ValtnDtls>
          <CollDtls><!-- Collateral information --></CollDtls>
```

...and its corresponding schema

```xml
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns="urn:iso:std:iso:20022:tech:xsd:auth.030.001.03" xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" targetNamespace="urn:iso:std:iso:20022:tech:xsd:auth.030.001.03">
    <xs:element name="Document" type="Document"/>
    <xs:simpleType name="ActiveCurrencyCode">
        <xs:annotation>
            <xs:documentation source="Name" xml:lang="EN">ActiveCurrencyCode</xs:documentation>
            <xs:documentation source="Definition" xml:lang="EN">A code allocated to a currency by a Maintenance
Agency under an international identification scheme as described in the latest edition of the international
standard ISO 4217 "Codes for the representation of currencies and funds".</xs:documentation>
        </xs:annotation>
        <xs:restriction base="xs:string">
            <xs:pattern value="[A-Z]{3,3}"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="ActiveOrHistoricCurrencyAnd13DecimalAmount_SimpleType">
        <xs:restriction base="xs:decimal">
            <xs:fractionDigits value="13"/>
            <xs:totalDigits value="18"/>
            <xs:minInclusive value="0"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:complexType name="ActiveOrHistoricCurrencyAnd13DecimalAmount">
        <xs:annotation>
            <xs:documentation source="Name"
xml:lang="EN">ActiveOrHistoricCurrencyAnd13DecimalAmount</xs:documentation>
            <xs:documentation source="Definition" xml:lang="EN">A number of monetary units specified in an
active or a historic currency where the unit of currency is explicit and compliant with ISO 4217. The number of
fractional digits (or minor unit of currency) is not checked as per ISO 4217: It must be lesser than or equal to
13.&#13;
Note: The decimal separator is a dot.</xs:documentation>
        </xs:annotation>
        <xs:simpleContent>
            <xs:extension base="ActiveOrHistoricCurrencyAnd13DecimalAmount_SimpleType">
                <xs:attribute name="Ccy" type="ActiveCurrencyCode" use="required">
                    <xs:annotation>
                        <xs:documentation source="Name" xml:lang="EN">Currency</xs:documentation>
                        <xs:documentation source="Definition" xml:lang="EN">Medium of exchange of value.
</xs:documentation>
                    </xs:annotation>
                </xs:attribute>
            </xs:extension>
        </xs:simpleContent>
    </xs:complexType>
    <xs:simpleType name="ActiveOrHistoricCurrencyAnd13DecimalAmount__1_SimpleType">
        <xs:restriction base="xs:decimal">
            <xs:fractionDigits value="13"/>
            <xs:totalDigits value="18"/>
            <xs:minInclusive value="0"/>
        </xs:restriction>
    </xs:simpleType>
```

Hyand

Agenda

What will we be looking at?

What will we be looking at?

Agenda

1. Short introduction to XML and XML in the Oracle Database
2. XML-Handling with practical examples
    1. Generate
    2. Store
    3. Query
    4. Validate
3. JSON vs. XML
4. Conclusion

XML

A (short) introduction

XML – Extensible Markup Language

- Markup Language and self-describing file format
- Represent data in a hirachical structure
- First published in 1998, and standardized in its current version (1.0) in the fifth edition, published in 2008
- Design Goal: Machine- and Human-readable
- Main Purpose: Data Exchange between Systems/Platforms

XML Basic Structure

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!-- Talk @ APEX Connect -->
<talk time="12:00:00">
    <title>Akte X(ML)</title>
    <presenter>Fabian Neureiter</presenter>
    <room>
        <name>Sanssouci</name>
        <seats>200</seats>
    </room>
</talk>
```

XML Namespaces

```
<!-- [.....]-->
 <talk>
    <title>Akte X(ML)</title>
    <presenter>Fabian Neureiter</presenter>
    <room>
        <name>Sanssouci</name>
        <seats>200</seats>
    </room>
 </talk>
<!-- [.....]-->
```

+

```
<!-- [.....]-->
<talk>
  <person1>Fabian Neureiter</person1>
  <person2>Larry Ellison</person2>
  <conversationStarter>
      <topic>Sailing and Boats in general</topic>
   </conversationStarter>
</talk>
<!-- [.....]-->
```

```xml
<?xml version="1.0" encoding="UTF-8"?>
<conference name="APEX connect">
    <!-- [.....]-->
        <talk>
        <title>Akte X(ML)</title>
        <presenter>Fabian Neureiter</presenter>
        <room>
            <name>Sanssouci</name>
            <seats>200</seats>
        </room>
        </talk>
    <!-- [.....]-->
        <talk>
            <person1>Fabian Neureiter</person1>
            <person2>Larry Ellison</person2>
            <conversationStarter>
                <topic>Sailing and Boats in general</topic>
            </conversationStarter>
        </talk>
    <!-- [.....]-->
</conference>
```

```xml
<?xml version="1.0" encoding="UTF-8"?>
<conference xmlns:pres="http://www.url.org/presentation"
xmlns:conv="urn::ISSN:0167-6423">
    <!-- [.....]-->
        <pres:talk>
          <pres:title>Akte X(ML)</pres:title>
          <pres:presenter>Fabian Neureiter</pres:presenter>
          <pres:room>
              <pres:name>Sanssouci</pres:name>
              <pres:seats>200</pres:seats>
          </pres:room>
        </pres:talk>
    <!-- [.....]-->
        <conv:talk>
            <conv:person1>Fabian Neureiter</conv:person1>
            <conv:person2>Larry Ellison</conv:person2>
            <conv:conversationStarter>
                <conv:topic>Sailing and Boats in general</conv:topic>
            </conv:conversationStarter>
        </conv:talk>
    <!-- [.....]-->
</conference>
```

XML

“What are the characteristics of the data?”

```xml
<?xml version="1.0" encoding="UTF-8"?>
<talk>
    <title>Akte X(ML)</title>
    <presenter>Fabian Neureiter</presenter>
    <room>
        <name>Sanssouci</name>
        <seats>200</seats>
    </room>
</talk>
```

HTML

“How should the data be displayed?”

```html
<!DOCTYPE html>
<html>
  <body>
    <h1>Akte X(ML)</h1>
    <h2>Fabian Neureiter</h3>
    <p>
      <h3>Room</h3>
      <ul>
        <li>Name: Sanssouci</li>
        <li>Seats: 200</li>
      </ul>
    </p>
  </body>
</html>
```

XML and the Oracle Database

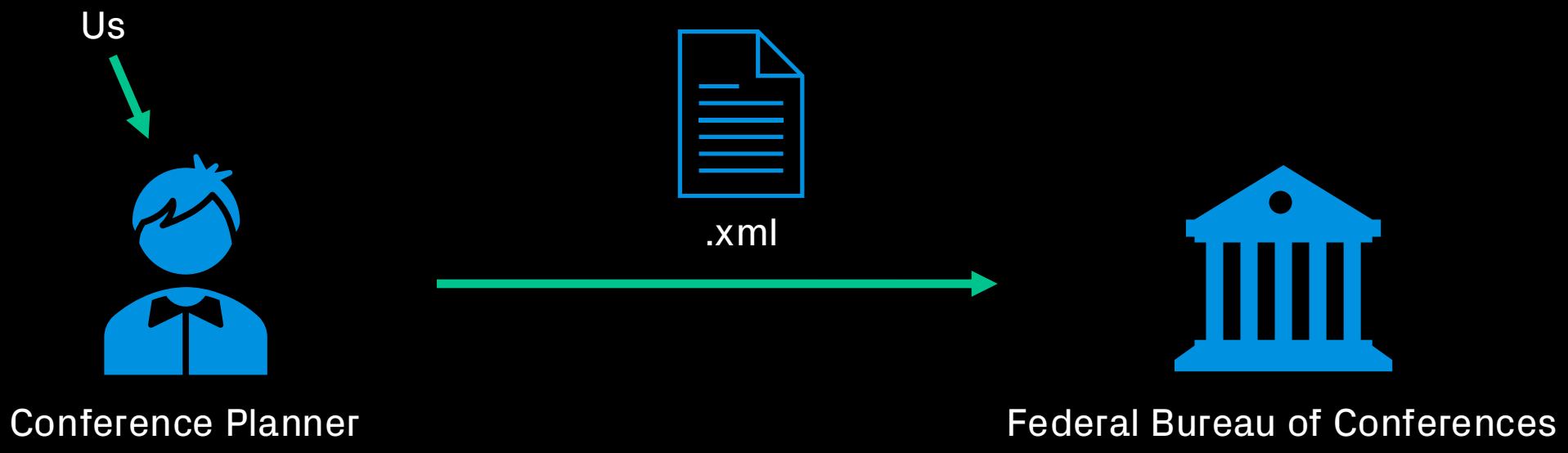
Oracle XML DB

- Umbrella term for all XML related technologies in the Oracle DB
- First introduced in Oracle9i Release 2 (2002)
- Included abstract SQL data type XMLType
- Includes Ways to store, query, update, and transform XML data while accessing it using SQL and XQuery.

XML and the Oracle Database

Generating XML

Let’s imagine



Generating XML from (relational) data

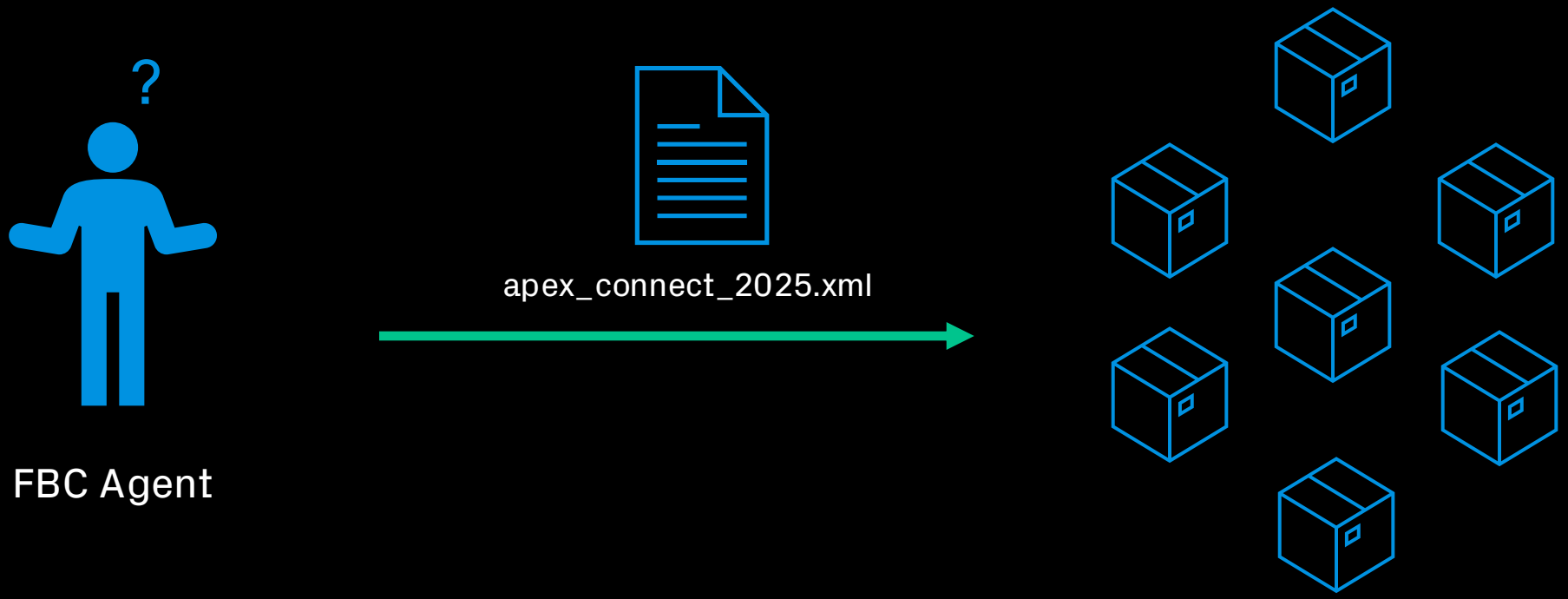
- DBMS_XMLGEN – deprecated in Oracle Database 23ai
- SQL Functions (recommended by Oracle)
  - XMLPARSE() – Parse string containing XML data , returns XMLType
  - XMLELEMENT() – Create XML-Element from relational Data, returns XMLType
  - XMLATTRIBUTES() – Create Attributes for XML-Elements
  - XMLAGG() – Aggregate Function, create Forest of XML-Elements
  - XMLFOREST() – Row Function, create Forest of XML-Elements based on row data

DEMO

XML and the Oracle Database

Storing XML

Let’s imagine



Storing XML data

- Abstract SQL data type XMLType is used
- Store as:
  - Binary XML
    - Default storage model since 12c
    - Post-parse, schema-aware
  - Object-relational
    - entity-relationship (ER) decomposition of the XML data
    - Query performance supposed to match relational data

https://docs.oracle.com/en/database/oracle/oracle-database/19/adxdb/choice-of-XMLType-storage-and-indexing.html#GUID-F1EE5B88-69E1-47B4-97FC-92F184C0CBA1

Binary XML

```
create table sample_xmls (
  sx_id number generated by default on null as identity
, sx_xml xmltype
) xmltype column sx_xml
  store as transportable binary xml
;
```

New in 23c

Object-relational

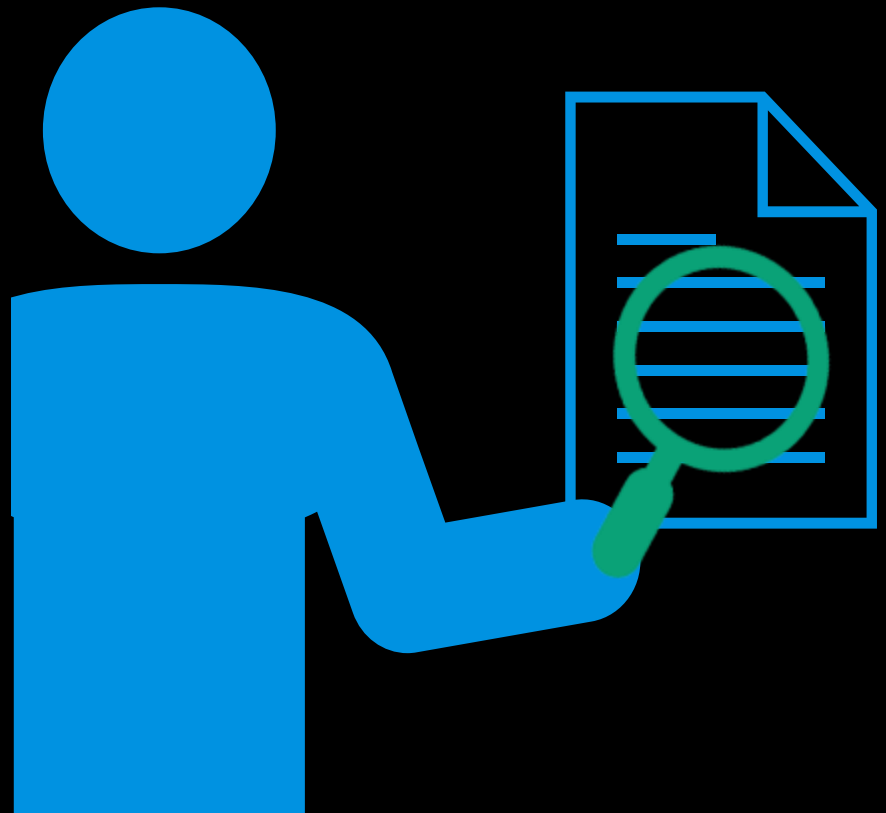
```
create table sample_xmls (
  sx_id number generated by default on null as identity
, sx_xml xmltype
) xmltype column sx_xml
  store as object relational
;
```

XML and the Oracle Database

Querying XML Data

Let’s imagine

FBC Agent



Need understanding of:

- Xpath

  Use path expressions to select nodes or node-sets in an XML document
- XQuery

  Language for finding and extracting elements and attributes from XML documents
- XMLTable

  XMLTable maps the result of an XQuery evaluation into relational rows and columns. You can query the result returned by the function as a virtual relational table using SQL

https://www.w3schools.com/xml, https://docs.oracle.com/en/database/oracle/oracle-database/23/sqlrf/XMLTABLE.html

Querying XML

Xpath

- /conference/day/slot[1]/talk

  Get the talk in the first slot

- /conference/day/slot[@time="10:00:00"]/talk

  Get the talk at 10 am

- //talk/presenter[text()="Simon Hunt"]/..

  Get the talk with Simon Hunt as presenter

- //room/seats[text() < 544]/..

  Get the room with less than 544 seats

```xml
<?xml version="1.0" encoding="UTF-8"?>
<conference name="APEX connect" year="2025">
  <day date="Tuesday, 13.05.2025">
    <slot time="09:15:00">
      <talk>
        <title>Opening of APEX Connect</title>
        <presenter>Niels de Bruijn</presenter>
        <room>
          <name>Das Studio</name>
          <seats>544</seats>
        </room>
      </talk>
    </slot>
    <slot time="10:00:00">
      <talk>
        <title>KEYNOTE: APEX Product Management Update</title>
        <presenter>Simon Hunt</presenter>
        <room>
          <name>Sanssouci</name>
          <seats>200</seats>
        </room>
      </talk>
    </slot>
  </day>
</conference>
```

Great Resource: devhints.io/xpath

Selectors

Descendant selectors

`h1`	`//h1`	?
`div p`	`//div//p`	?
`ul > li`	`//ul/li`	?
`ul > li > a`	`//ul/li/a`	
`div > *`	`//div/*`	
`:root`	`/`	?
`:root > body`	`/body`	

Attribute selectors

`#id`	`//*[@id="id"]`	?
`.class`	`//*[@class="class"]` ...kinda	
`input[type="submit"]`	`//input[@type="submit"]`	
`a#abc[for="xyz"]`	`//a[@id="abc"][@for="xyz"]`	?
`a[rel]`	`//a[@rel]`	

Siblings

`h1 ~ ul`	`//h1/following-sibling::ul`	?
`h1 + ul`	`//h1/following-sibling::ul[1]`	
`h1 ~ #id`	`//h1/following-sibling::[@id="id"]`	

jQuery

`$('ul > li').parent()`	`//ul/li/..`	?
`$('li').closest('section')`	`//li/ancestor-or-self::section`	
`$('a').attr('href')`	`//a/@href`	?
`$('span').text()`	`//span/text()`	

Other things

`h1:not([id])`	`//h1[not(@id)]`	?
Text match	`//button[text()="Submit"]`	?

XQuery

- /conference/day/slot[@time="10:00:00"]/talk

  for $talk in /conference/day/slot[@time="10:00:00"]/talk

  return $talk

  Get the talk at 10 am

- //talk/presenter[text()="Simon Hunt"]/..

  for $talk in //talk/presenter[text()="Simon Hunt"]/..

  return $talk

  Get the talk with Simon Hunt as presenter

```xml
<?xml version="1.0" encoding="UTF-8"?>
<conference name="APEX connect" year="2025">
  <day date="Tuesday, 13.05.2025">
    <slot time="09:15:00">
      <talk>
        <title>Opening of APEX Connect</title>
        <presenter>Niels de Bruijn</presenter>
        <room>
          <name>Das Studio</name>
          <seats>544</seats>
        </room>
      </talk>
    </slot>
    <slot time="10:00:00">
      <talk>
        <title>KEYNOTE: APEX Product Management Update</title>
        <presenter>Simon Hunt</presenter>
        <room>
          <name>Sanssouci</name>
          <seats>200</seats>
        </room>
      </talk>
    </slot>
  </day>
</conference>
```

DEMO

XMLTable

- Extract XML Data to relational data

DEMO

```xml
<?xml version="1.0" encoding="UTF-8"?>
<conference name="APEX connect" year="2025">
  <day date="Tuesday, 13.05.2025">
    <slot time="09:15:00">
      <talk>
        <title>Opening of APEX Connect</title>
        <presenter>Niels de Bruijn</presenter>
        <room>
          <name>Das Studio</name>
          <seats>544</seats>
        </room>
      </talk>
    </slot>
    <slot time="10:00:00">
      <talk>
        <title>KEYNOTE: APEX Product Management Update</title>
        <presenter>Simon Hunt</presenter>
        <room>
          <name>Sanssouci</name>
          <seats>200</seats>
        </room>
      </talk>
    </slot>
  </day>
</conference>
```

Structured Indexes

- XML-Index is specific type of structured index
- Leverages relational tables to store the indexed content of the XML data
- Great for XMLs with specific predefined structures

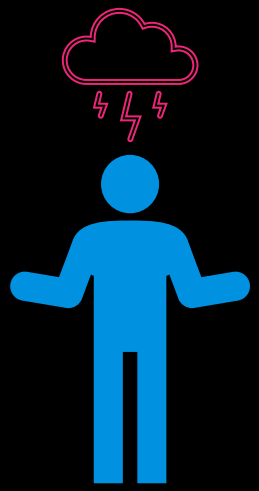
```
create index xst_index
on xml_sample_table(xst_xml) indextype is XDB.XMLIndex
parameters(
    q'[
    xmltable conference_tab
       '/conference'
       passing dcxm.dcxm_xml
       columns
          name varchar2(50) path '@name'
        , year number path '@year'
        , day xmltype path 'day' VIRTUAL
    xmltable day_tab
        '/day' passing conf.day
        columns
         day_date varchar2(50) path '@date'
        , slot xmltype path 'slot' VIRTUAL
    xmltable slot_tab
        '/slot' passing day.slot
        columns
         time varchar2(50) path '@time'
        , talk xmltype path 'talk' VIRTUAL
    xmltable talk_tab
        '/talk' passing slot.talk
        columns
          title varchar2(150) path 'title'
        , presenter varchar2(50) path 'presenter'
        , room xmltype path 'room' VIRTUAL
    xmltable room_tab
        '/room' passing talk.room
        columns
          name varchar2(50) path 'name'
        , seats number path 'seats'
    ]'
);
```

DEMO

XML and the Oracle Database

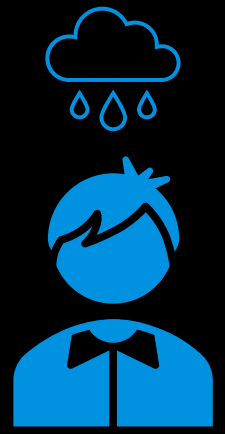
Editing XML Data

Let’s imagine



FBC Agent

apex_connect_2025_DENIED.xml



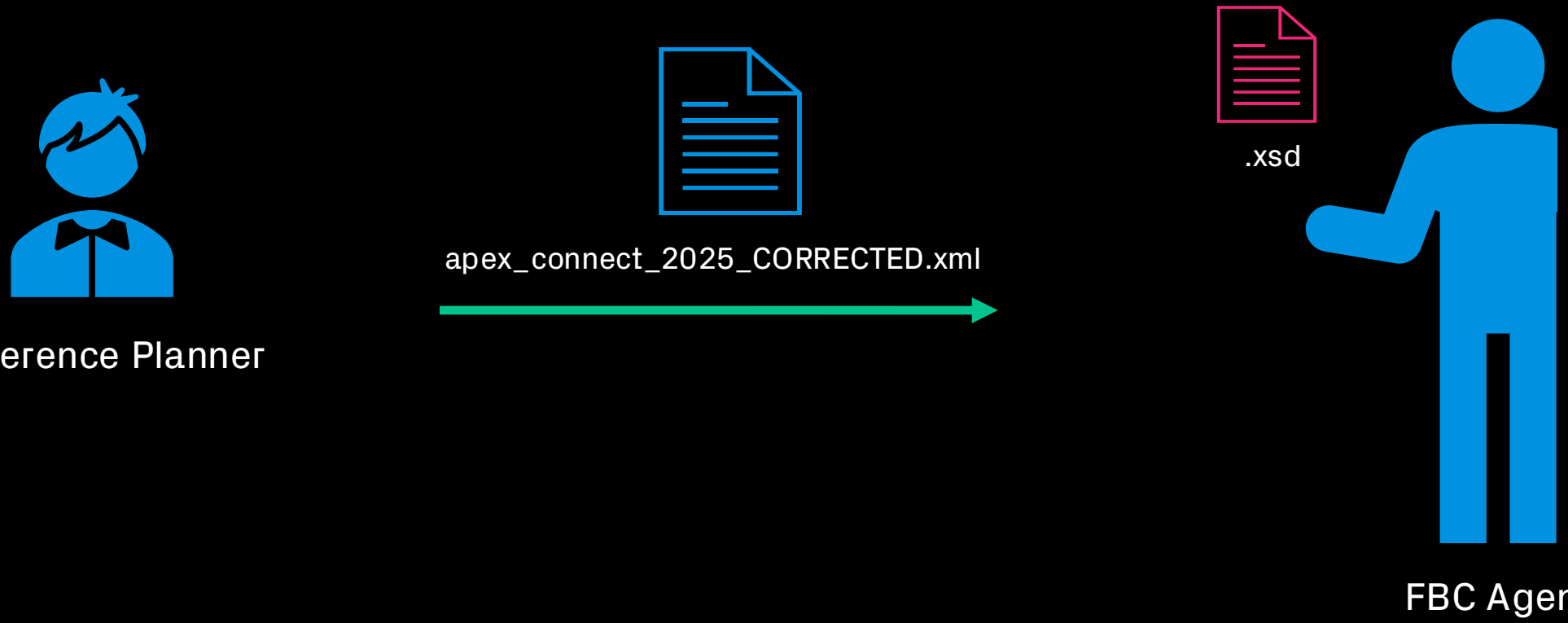
Conference Planner

DEMO

XML and the Oracle Database

Validate XML Data

Let’s imagine



XML Schemas

- Structural definition of an XML-File
- Also known as XSD (XML Schema Definition)
- Simple Datatypes like string, integer, boolean
- …but also support for custom complex types which are collections of simple datatypes

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

    <!-- definition of simple elements -->
    <xs:element name="title" type="xs:string"/>
    <xs:element name="presenter" type="xs:string"/>
    <xs:element name="name" type="xs:string"/>
    <xs:element name="seats" type="xs:integer"/>

    <!-- definition of attributes -->
    <xs:attribute name="time" type="xs:time"/>

    <!-- complex elements -->
    <xs:element name="room">
        <xs:complexType>
        <xs:sequence>
            <xs:element ref="name" minOccurs="1"/>
            <xs:element ref="seats"/>
        </xs:sequence>
        </xs:complexType>
    </xs:element>

    <xs:element name="talk">
        <xs:complexType>
        <xs:sequence>
            <xs:element ref="title" minOccurs="1"/>
            <xs:element ref="presenter"/>
            <xs:element ref="room" maxOccurs="1"/>
        </xs:sequence>
        </xs:complexType>
    </xs:element>

    <xs:element name="slot">
        <xs:complexType>
        <xs:sequence>
            <xs:element ref="talk"/>
        </xs:sequence>
        <xs:attribute ref="time" use="required"/>
        </xs:complexType>
    </xs:element>

    </xs:schema>
```

DEMO

What does the future hold for XML?

XML vs JSON

XML

```xml
<?xml version="1.0" encoding="UTF-8"?>
<conference name="APEX connect" year="2025">
  <day date="Tuesday, 13.05.2025">
    <slot time="09:15:00">
      <talk>
        <title>Opening of APEX Connect</title>
        <presenter>Niels de Bruijn</presenter>
        <room>
          <name>Das Studio</name>
          <seats>544</seats>
        </room>
      </talk>
    </slot>
    <slot time="10:00:00">
      <talk>
        <title>KEYNOTE: APEX Product Management Update</title>
        <presenter>Simon Hunt</presenter>
        <room>
          <name>Sanssouci</name>
          <seats>200</seats>
        </room>
      </talk>
    </slot>
  </day>
</conference>
```

JSON

```json
{
  "conference": {
    "@name": "APEX connect",
    "@year": "2025",
    "day": {
      "@date": "Tuesday, 13.05.2025",
      "slot": [
        {
          "@time": "09:15:00",
          "talk": {
            "title": "Opening of APEX Connect",
            "presenter": "Niels de Bruijn",
            "room": {
              "name": "Das Studio",
              "seats": 544
            }
          }
        },
        {
          "@time": "10:00:00",
          "talk": {
            "title": "KEYNOTE: APEX Product Management Update",
            "presenter": "Simon Hunt",
            "room": {
              "name": "Sanssouci",
              "seats": 200
            }
          }
        }
      ]
    }
  }
}
```

JSONx

```
{
  "jx:ns": "http://www.jsonx.org/schema-0.3.jsd",
  "jx:schemaLocation": "http://www.jsonx.org/schema-0.3.jsd http://www.jsonx.org/schema-0.3.jsd",
  "message": { "jx:type": "object", "abstract": true },
  "swift": {
    "jx:type": "object", "extends": "message", "properties": {
      "type": { "jx:type": "string", "pattern": "swift", "nullable": false },
      "code": { "jx:type": "string", "pattern": "[A-Z]{6}[A-Z0-9]{2}([A-Z0-9]{3})?", "nullable": false, "use": "required" } }
  },
  "iban": {
    "jx:type": "object", "properties": {
      "type": { "jx:type": "string", "pattern": "iban", "nullable": false },
      "code": { "jx:type": "string", "pattern": "[A-Z]{2}\\d{2} ?\\d{4} ?\\d{4} ?\\d{4} ?\\d{4} ?[\\d]{0,2}", "nullable": false }
    }
  },
  "ach": {
    "jx:type": "object", "properties": {
      "type": { "jx:type": "string", "pattern": "ach", "nullable": false },
      "code": { "jx:type": "string", "pattern": "\\w{1,17}", "nullable": false },
      "routing": { "jx:type": "string", "pattern": "\\d{9}", "nullable": false }
    }
  }
```

JSON Schema

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "definitions": {
    "swift": {
      "type": "object",
      "properties": {
        "type": { "type": "string", "pattern": "swift" },
        "code": { "type": "string", "pattern": "(\\([0-9]{3}\\))?[0-9]{3}-[0-9]{4}" },
        "required": ["type", "code"]
      }
    },
    "iban": {
      "type": "object",
      "properties": {
        "type": { "type": "string", "pattern": "iban" },
        "code": { "type": "string", "pattern": "[A-Z]{2}\\d{2} ?\\d{4} ?\\d{4} ?\\d{4} ?\\d{4} ?\\d{0,2}" },
        "required": ["type", "code"]
      }
    },
    "ach": {
      "type": "object",
      "properties": {
        "type":    { "type": "string", "pattern": "ach" },
        "code":    { "type": "string", "pattern": "\\w{1,17}" },
        "routing": { "type": "string", "pattern": "\\d{9}" },
        "required": ["type", "code", "routing"]
      }
    }
  }
}
```

Useful Resources

Useful Resources

- Documentation (23ai)
  https://docs.oracle.com/en/database/oracle/oracle-database/23/adxdb/index.html
- Video + Demo (Oracle, 2024)
  https://www.youtube.com/watch?v=s1Bc8KKLbpw&t=334s
- Workshop (Oracle)
  https://apexapps.oracle.com/pls/apex/dbpm/r/livelabs/view-workshop?wid=3661
- JSON Schema
  https://json-schema.org/
- JSONx
  https://datatracker.ietf.org/doc/html/draft-rsalz-jsonx-00

Say Hy_

+49 (0) 531 23767-0
info@hyand.com