

# Validating Quicksand: Schema Versioning in $\tau$ XSchema

Curtis Dyreson, Richard T. Snodgrass  
 Faiz Currim, and Sabah Currim,  
 Washington State University - Curtis  
 University of Arizona – Rick and Sabah  
 University of Iowa – Faiz

XSDM 2006 - Atlanta

## Changes to XML Data over Time



- As of January 2002



```
<athlete>
  <name>Kjetil Andre Aamodt</name>
</athlete>
```

- As of March 2002



```
<athlete>
  <name>Kjetil Andre Aamodt</name> in
  <medal mtype="silver">Men's Combined</medal>
</athlete>
```

- As of July 2002



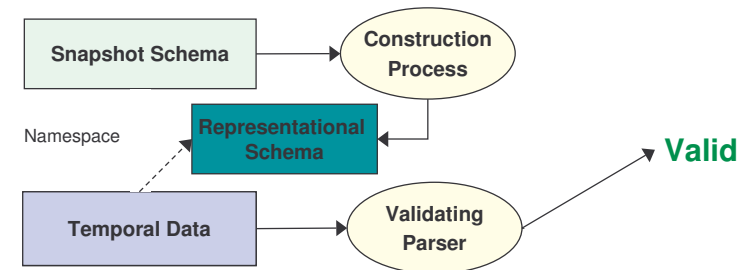
```
<athlete>
  <name>Kjetil Andre Aamodt</name> in
  <medal mtype="gold">Men's Combined</medal>
</athlete>
```

## Temporal Olympics Data

```
<athleteItem>
  <athleteVersion>
    <transTime start="January 2002" stop="March 2002"/>
    <athlete> 1
      <name>Kjetil Andre Aamodt</name>
    </athlete>
  </athleteVersion>
  <athleteVersion>
    <transTime start="March 2002" stop="now"/>
    <athlete> 2
      <name>Kjetil Andre Aamodt</name> in
      <medalItem>
        ...
      </medalItem>
    </athlete>
  </athleteVersion>
</athleteItem>
```

## Validating Temporal Data

- Snapshot data validated with a *snapshot schema*



## Uses of an XML Schema

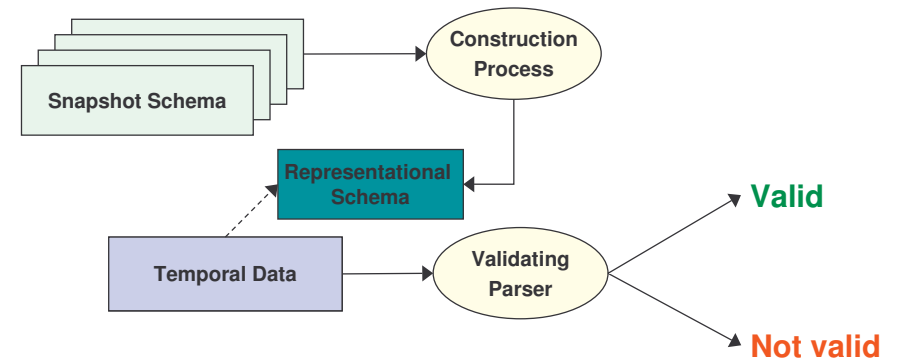
---

- Validation
- XML editors
- Guides query formulation
- Query optimization
- Provides a web service binding

## With Schema Versioning

---

- Snapshot schema can change over time



- Construct a *time-varying temporal schema*

## Outline

---

- Motivation
- $\tau$ XSchema
- Time-varying schemas
- Architecture
- Summary and future work

## Goals

---

- Upwards compatibility
  - No changes to XML Schema
  - Reuse off-the-shelf parsers/tools
- Support
  - Valid and transaction time
  - Element versioning
  - Schema versioning
  - Logical/physical independence
    - ◆ Flexible timestamp representation

## Start with a Snapshot Schema

- A *snapshot* schema, winOlympics.xsd

```
<element name="athlete">
  <complexType mixed="true">
    <sequence>
      <element name="name" type="string"
        minOccurs="1" maxOccurs="1" />
      <element ref="medal"
        minOccurs="0" maxOccurs="unbounded"/>
    </sequence>
  </complexType>
</element>
```

- Each snapshot <athlete> conforms to this schema

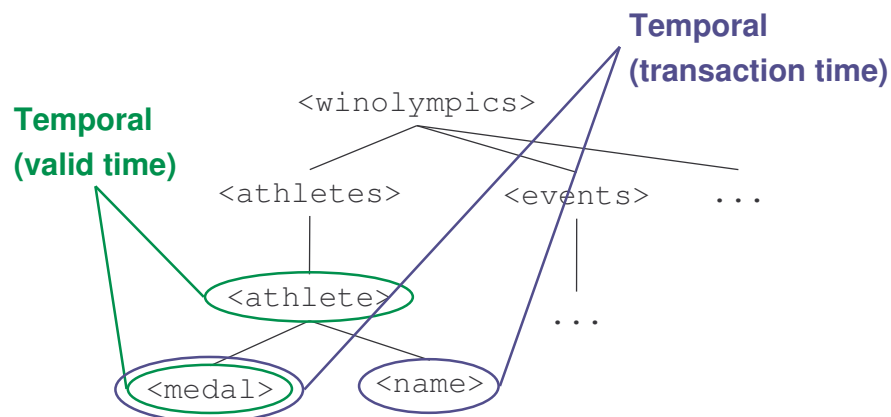
## Add Temporal Annotations

- Specify *what* is time-varying

```
<temporalAnnotations ... >
  ...
  <validTime target="/winOlympic//athlete"
    kind="state"
    contentVarying="true"
    identifier="name"/>
  ...
  <transactionTime target="/winOlympic//name"/>
  ...
</temporalAnnotations>
```

- Simple constraints (state/event, existence/content-varying)
- *SchemaPath* expressions (XPath for identifier)

## Annotating the Schema “Tree”



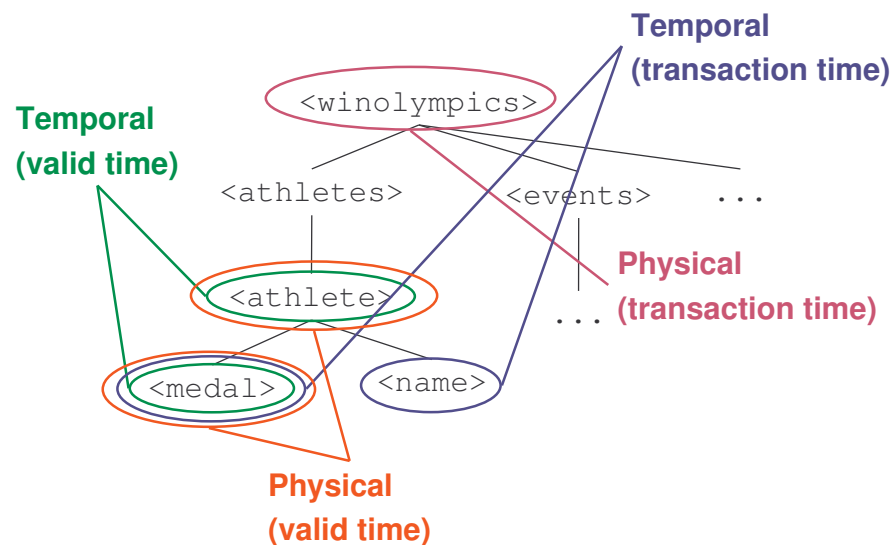
## Add Physical Annotations

- Specify *where* to put timestamps

```
<physicalAnnotations ... >
  ...
  <stampPosition target="/winOlympic//athlete"
    validTimeStampType="extent"/>
  ...
  <transactionTime target="/winOlympic"/>
  ...
</physicalAnnotations>
```

- Step or extent

## Physical Annotations to the Schema



## Add a Temporal Bundle

- Specify *how* a snapshot schema is made temporal

```
<temporalBundle>
  <schemaAnnotation
    snapshotSchema="winOlympics.xsd"
    temporalAnnotation="timeWO.xml"
    physicalAnnotation="phyWO.xml"
  />
</temporalBundle>
```

## Extend a Temporal Bundle for Versioning

- Specify *how* a snapshot schema is made temporal

```
<temporalBundle>
  <bundleSequence>
    <schemaAnnotation
      snapshotSchema="winOlympics.xsd"
      temporalAnnotation="timeWO.xml"
      physicalAnnotation="phyWO.xml">
      <tTime>May 21, 2005</tTime>
    </schemaAnnotation>
    <schemaAnnotation
      snapshotSchema="winOlympics2.xsd"
      temporalAnnotation="timeWO.xml"
      ...
    </temporalBundle>
```

## Evolving Item Identifiers

- Used to glue items
- Might change over time
- Four options
  - Use old
  - Use new
  - Use both
  - Replace

## Outline

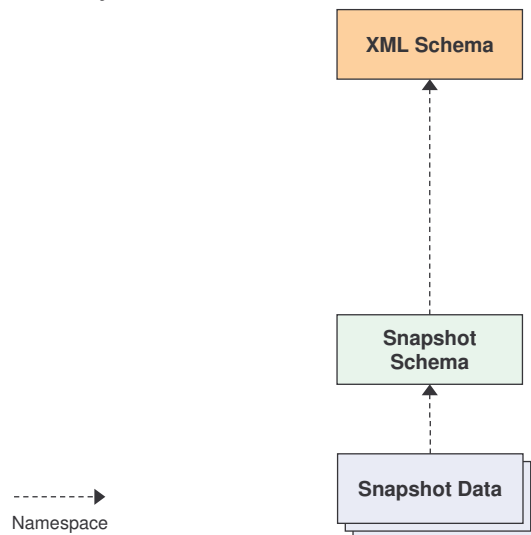
- Motivation
- $\tau$ XSchema
- Time-varying schemas
- Architecture
- Summary and future work

## Tools

- $\tau$ **VALIDATOR** – Validating temporal XML document for conventional and temporal constraints
- **SQUASH** – Generating a temporal document from a sequence of snapshot documents
- **UN**SQUASH – Extracting snapshot documents from a temporal document
- **RES**QUASH – Changing a document representation to be consistent with the new physical annotation.

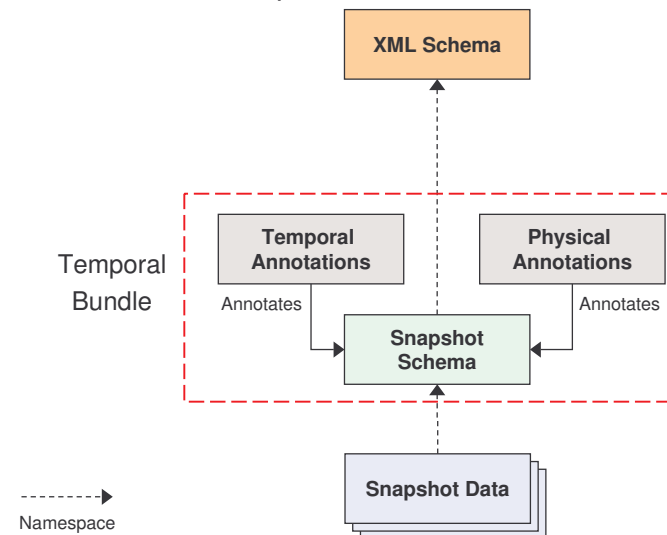
## Architecture

- Initially



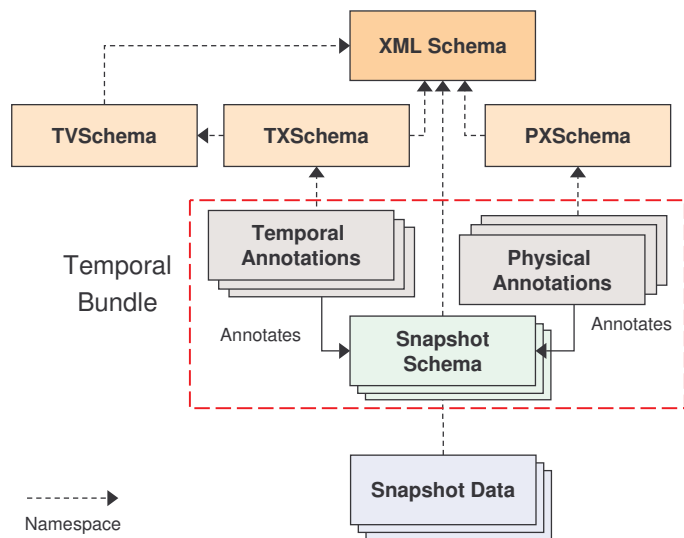
## Architecture

- Annotate the Snapshot Schema



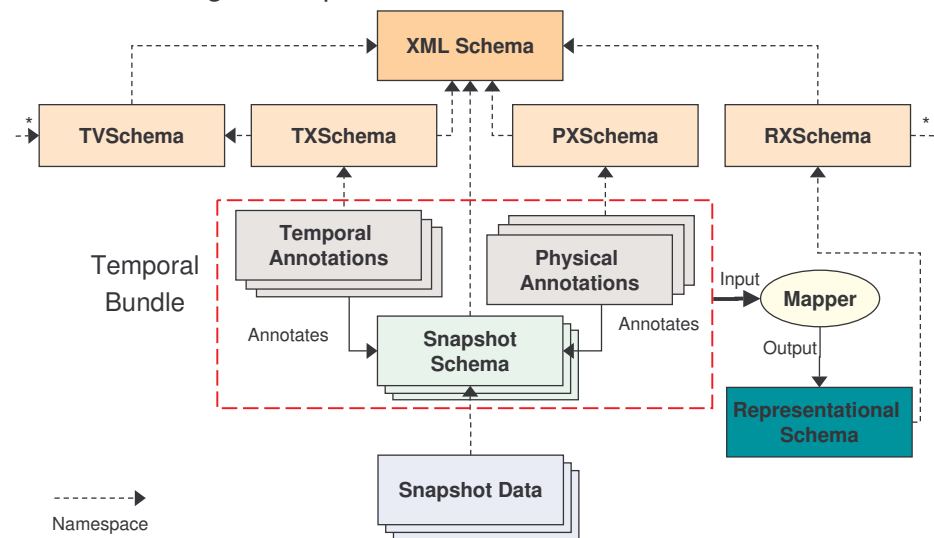
## Architecture

- Schemas for the Annotations



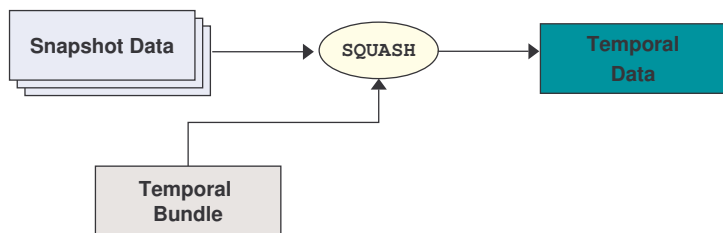
## Architecture

- Generating the Representational Schema



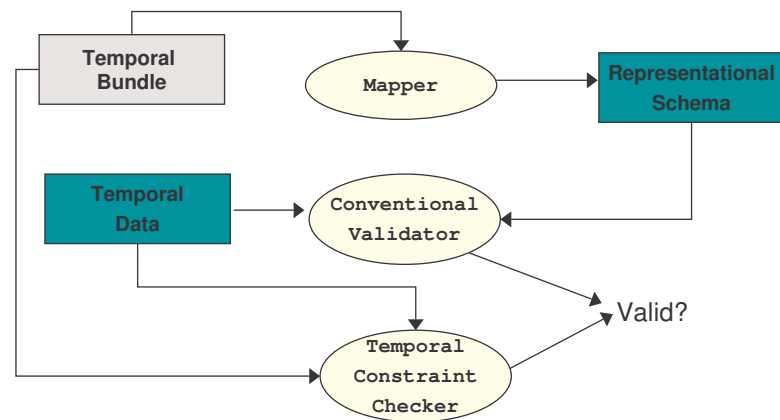
## Creating Temporal Data

- Use SQUASH



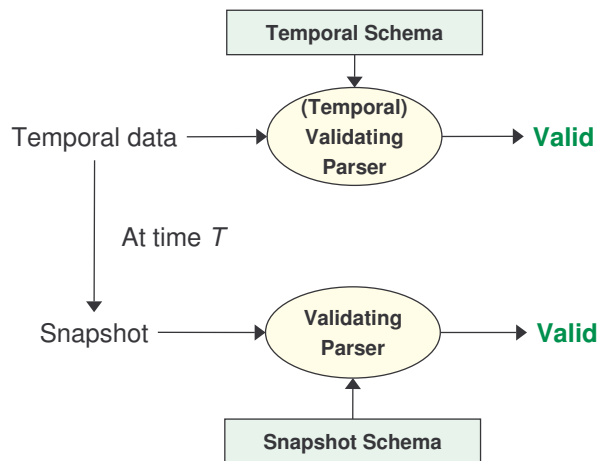
## Validating Temporal Data

- Implemented as *t*VALIDATOR



## Property of a “Good” Construction

- Every snapshot must conform to the snapshot schema



## Related Work – Temporal XML

- XML Schema languages
  - Many, but XML Schema is backed by the W3C
- Change detection and management
  - Nguyen, Abiteboul, Cobena, Preda, *SIGMOD* 2001
  - Xyleme's Alerter, described in *Data Engineering Bulletin*, 2001
  - Dyreson, Lin, Wang *WWW* 2004
- Representing time-varying documents (versioning)
  - Chawathe, Abiteboul, Widom, *ICDE* 1998
  - Dyreson, Böhlen, Jensen, *VLDB* 1999
  - Chien, Tsotras, Zaniolo, *VLDB* 2000
  - Marian, Abiteboul, Cobena, Mignet, *VLDB* 2001
  - Buneman, Khanna, Tajima, Tan, *SIGMOD* 2002, *TODS* 2004
- Incremental validation
  - Bouchou & Halfeld-Ferrari, *DBPL* 2003
  - Papkonstantinou & Vianu, *ICDT* 2003
  - Barbosa, Mendelzon, Libkin, Mignet, Arenas, *ICDE* 2004

## Related Work - Versioning

- XML Data Versioning
  - A Data model for Temporal XML Documents (T. Amagas, M. Yoshikawa, S. Uemura)
  - Efficient Schemes for managing multi-version XML Documents (S. Chien, V. Tsotras, and C. Zaniolo)
- Schema Evolution in Relational Database Systems
  - J. F. Roddick bibliograhpy
- XML Versioning Use Cases (W3C)

## Our Contributions

- Validate temporal data (time-varying documents)
- Use XML Schema
  - Conforms to and built on top of XML Schema
- Ensure data independence
  - The snapshot schema document
  - Temporal annotations - What portion(s) can vary over time
  - Physical annotations - Where to place timestamps and how to represent change
- Support Schema Versioning
- Permit various physical representations
- Suite of tools
  - Reuse and extend existing tools
- [www.cs.arizona.edu/tau](http://www.cs.arizona.edu/tau)

## Future Work

---

- Schema-constant periods
- Support temporal granularity and indeterminacy
- Handle time-varying intensional XML data
- Integrate with an XML editor