Toward LingML

Is the notation really the theory?

Michael Hammond
University of Arizona

Thanks to: Debbie Cole, Terry Langendoen, and Diane Ohala.
Goals
Goals

- Review the relationship of notation to phonological theory
Goals

- Review the relationship of notation to phonological theory
- Review the goals and structure of XML
Goals

- Review the relationship of notation to phonological theory
- Review the goals and structure of XML
- Review the goals and structure of XSLT
Goals

- Review the relationship of notation to phonological theory
- Review the goals and structure of XML
- Review the goals and structure of XSLT
- Show how we’re using these in the Arizona Native American Online Dictionary Project
Goals

- Review the relationship of notation to phonological theory
- Review the goals and structure of XML
- Review the goals and structure of XSLT
- Show how we’re using these in the Arizona Native American Online Dictionary Project
- Show how XML/XSLT offers a new approach to the relationship between theory and notation
Notation and Theory in Phonology

A [k] is palatalized to [č] before an [i].
Notation and Theory in Phonology

A [k] is palatalized to [č] before an [i].

\[
\begin{bmatrix}
+hi \\
+bk
\end{bmatrix}
\rightarrow
\begin{bmatrix}
-bk \\
+delrel
\end{bmatrix}
/ - \left[ \begin{array}{c}
+\text{syl} \\
+hi \\
-bk
\end{array} \right]
\]

*The Sound Pattern of English* (Chomsky & Halle, 1968)
Phonological Rule Notation
Phonological Rule Notation

- A specific notational scheme
Phonological Rule Notation

- A specific notational scheme
- This scheme represents the innate predisposition that speakers have to learn certain kinds of phonological generalizations, and not others.
Phonological Rule Notation

- A specific notational scheme
- This scheme represents the innate predisposition that speakers have to learn certain kinds of phonological generalizations, and not others.
- Certain rules can be written, and not others.
Phonological Rule Notation

- A specific notational scheme
- This scheme represents the innate predisposition that speakers have to learn certain kinds of phonological generalizations, and not others.
- Certain rules can be written, and not others.
- “The notation is the theory.”
Is this a good thing?
Is this a good thing?

- Explicitness vs. triviality
Is this a good thing?

- Explicitness vs. triviality
- Universality and restrictiveness
What is XML
What is XML

- XML = “Extensible Markup Language”
What is XML

- XML = “Extensible Markup Language”
- Information exchange
What is XML

- XML = “Extensible Markup Language”
- Information exchange
- Medium-neutral electronic publishing
Structure of XML
Structure of XML

- Looks like HTML (simplified SGML)
Structure of XML

- Looks like HTML (simplified SGML)
- Tags and attributes
Structure of XML

- Looks like HTML (simplified SGML)
- Tags and attributes
- context-free syntax
An Example: the Tohono O’odham Dictionary

<entryset>
  <entry id="e120">
    <headword xml:lang="x-to">ba:ban</headword>
    <sense>
      <definition>
        see ban: coyotes
      </definition>
    </sense>
  </entry>
</entryset>
What do we see?
What do we see?

• A “nested” structure
What do we see?

- A “nested” structure
- Logical structure, not formatting
What do we see?

- A “nested” structure
- Logical structure, not formatting
- Text-based
XML Tags
XML Tags

1. Angled brackets.
XML Tags

1. Angled brackets.

2. They can have any name, e.g. `<mytag>`.
XML Tags

1. Angled brackets.
2. They can have *any* name, e.g. `<mytag>`.
3. They can occur in pairs, e.g. `<mytag>Some text</mytag>`. 
XML Tags

1. Angled brackets.

2. They can have *any* name, e.g. `<mytag>`.

3. They can occur in pairs, e.g. `<mytag>Some text</mytag>`.

4. They can occur alone, e.g. `<mytag/>`. 
XML Tags

1. Angled brackets.

2. They can have *any* name, e.g. `<mytag>`. 

3. They can occur in pairs, e.g. `<mytag>Some text</mytag>`. 

4. They can occur alone, e.g. `<mytag/>`. 

5. They can have attributes, e.g. `<mytag myfeat="avalue">`. 

Document Type Definitions (DTDs)
Document Type Definitions (DTDs)

- You can specify what tags can occur in a document.
Document Type Definitions (DTDs)

- You can specify what tags can occur in a document.
- You can specify what attributes can occur with what tags.
Document Type Definitions (DTDs)

- You can specify what tags can occur in a document.
- You can specify what attributes can occur with what tags.
- You can specify where textual data go with respect to the tags.
Document Type Definitions (DTDs)

- You can specify what tags can occur in a document.
- You can specify what attributes can occur with what tags.
- You can specify where textual data go with respect to the tags.
- You can specify how those tags are ordered with respect to each other.
A Sample Partial DTD

```xml
<!ELEMENT entryset (entry+)>  
<!ELEMENT entry (example|(headword,sense+,compare*))>  
<!ELEMENT sense (grammar?,compare*,definition+,see*)>  
<!ELEMENT definition (#PCDATA|breakdown|example)*>  
<!ELEMENT headword (#PCDATA)>  

<!ATTLIST entry xml:lang NMTOKEN #IMPLIED  
  id ID #IMPLIED>  

<!ATTLIST headword xml:lang NMTOKEN #IMPLIED  
  id ID #IMPLIED>
```
What can we do with this?
What can we do with this?

- We can check for the “well-formedness” of some data (validation).
What can we do with this?

- We can check for the “well-formedness” of some data (validation).
- We can display the data in various ways (rendering).
What can we do with this?

- We can check for the “well-formedness” of some data (validation).
- We can display the data in various ways (rendering).
- We can do things with the data (processing).
XSLT

An XML-based programming language to convert XML to other markup schemes:

- XML (with same or different DTD)
- HTML
- \LaTeX
- and more...
The O’odham Dictionary
The O’odham Dictionary

- Text display (XML → XSLT → \LaTeX)
The O’odham Dictionary

- Text display (XML → XSLT → LaTeX)
- Web display (XML → XSLT → HTML)
The O’odham Dictionary

- Text display (XML $\rightarrow$ XSLT $\rightarrow$ $\LaTeX$)
- Web display (XML $\rightarrow$ XSLT $\rightarrow$ HTML)
- Search (SQL/Java $\rightarrow$ perl $\rightarrow$ XML)
Linguistic Data

- Open Language Archives Community (OLAC)
- International Standard for Language Engineering (ISLE)
- Documentation of Endangered Languages (Dokumentation Bedrohter Sprachen, DOBES)
- Electronic Metastructure for Endangered Languages Data (EMELD)
Linguistic Theory

- Finite-state automata
- Optimality Theory
- Et cetera!
Finite-State Automata
FSA in XML

<net>
  <fsanode label="n1" final="false" start="true">
    <arc symbol="a" endnode="n1"/>
    <arc symbol="b" endnode="n2"/>
  </fsanode>
  <fsanode label="n2" final="true" start="false">
    <arc symbol="a" endnode="n1"/>
    <arc symbol="c" endnode="n1"/>
    <arc symbol="d" endnode="n2"/>
  </fsanode>
</net>
Rendering and “Running”
Rendering and “Running”

- graphing (Graphviz)
Rendering and “Running”

- graphing (Graphviz)
- HTML
Rendering and “Running”

- graphing (Graphviz)
- HTML
- ‘Running’ the automaton
Optimality Theory
Optimality Theory

- A theory of grammar, including:
Optimality Theory

- A theory of grammar, including:
- A finite set of universal constraints.
Optimality Theory

- A theory of grammar, including:
- A finite set of universal constraints.
- All language variation is described by ranking the constraints in different orders.
Optimality Theory

- A theory of grammar, including:
  - A finite set of universal constraints.
  - All language variation is described by ranking the constraints in different orders.
  - The candidate that violates higher-ranked constraints least wins.
Some Constraints

**FAITH-NASAL**
Nasals must be preserved.

**NASAL-VOWEL**
An oral vowel cannot occur before a nasal consonant.

**FAITH**
All segments must be preserved.
A Tableau

<table>
<thead>
<tr>
<th>/ban/</th>
<th>FAITH-NASAL</th>
<th>*NASAL-VOWEL</th>
<th>FAITH</th>
</tr>
</thead>
<tbody>
<tr>
<td>[ban]</td>
<td></td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>[bǎn]</td>
<td></td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>[bag]</td>
<td>*</td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>[ba]</td>
<td>*</td>
<td></td>
<td>*</td>
</tr>
</tbody>
</table>
An OT constraint in XML

<constraint name="Faith-Nasal" type="faithfulness">
  <pattern>m</pattern>
  <pattern>n</pattern>
  <pattern>&engma;</pattern>
</constraint>
An OT tableau in XML

<tableau>
    <con href="mycon.xml"/>
    <input>ban</input>
    <candidate>ban</candidate>
    <candidate>b&nasalA;n</candidate>
    <candidate>bag</candidate>
    <candidate>ba</candidate>
</tableau>
OT with XML
OT with XML

- \LaTeX
OT with XML

- \texttt{\LaTeX}
- \texttt{HTML}
OT with XML

- \textsc{latex}
- HTML
- Generating web tableaux
OT with XML

- **LaTeX**
- **HTML**
- Generating web tableaux
- Generating print tableaux
What is “MathML”?
What is “MathML”? 

- An initiative of the W3 Consortium
What is “MathML”?

- An initiative of the W3 Consortium
- Logical markup of mathematics so that it can be rendered typographically and graphically, and so that it can be processed computationally.
What is “MathML”? 

- An initiative of the W3 Consortium
- Logical markup of mathematics so that it can be rendered typographically and graphically, and so that it can be processed computationally.
- We can do the same in linguistics.
Toward “LingML”
Toward “LingML”

- XML/XSLT offers a convenient tool for exchanging, representing, and manipulating linguistic data.
Toward “LingML”

- XML/XSLT offers a convenient tool for exchanging, representing, and manipulating linguistic data.
- It also offers a new approach to the relationship between theory and notation.
Why LingML?
Why LingML?

- It forces us to be explicit about what our theoretical statements mean.
Why LingML?

- It forces us to be explicit about what our theoretical statements mean.
- It liberates us from formatting concerns, since simply expressing the logic of the constraint permits us to use tools that will format it.
Why LingML?

- It forces us to be explicit about what our theoretical statements mean.
- It liberates us from formatting concerns, since simply expressing the logic of the constraint permits us to use tools that will format it.
- It allows us to test our analyses simply by expressing them logically.
Why LingML?

- It forces us to be explicit about what our theoretical statements mean.

- It liberates us from formatting concerns, since simply expressing the logic of the constraint permits us to use tools that will format it.

- It allows us to test our analyses simply by expressing them logically.

- It brings us full circle to ask: is the notation the theory?