DadTeX, a full Arabic interface

M. Eddahibi, A. Lazrek, K. Sami

Department of Computer Science
Cadi Ayyad University

Marrakesh
November 9 - 11, 2006

http://www.ucam.ac.ma/fssm/rydarab
Outlines

1. Arabic writing
2. Arabic mathematical expressions
3. Arabic scientific document composition
4. DadTEX motivations
5. DadTEX system
6. Open problems and prospects
- Arabic letters represent only consonants or long vowels
- Optional diacritical marks for short vowels can be used to annotate text or spell it out in full when desired
- Arabic script is written from right to left
- Arabic script is cursive
Arabic Writing

- Arabic letters represent only consonants or long vowels.
- Optional diacritical marks for short vowels can be used to annotate text or spell it out in full when desired.
- Arabic script is written from right to left.
- Arabic script is cursive.
Arabic Writing

- Arabic letters represent only consonants or long vowels.
- Optional diacritical marks for short vowels can be used to annotate text or spell it out in full when desired.
- Arabic script is written from right to left.
- Arabic script is cursive.

Logical order: ب ت ك
Visual order: ك ت ب
• Arabic letters represent only consonants or long vowels
• Optional diacritical marks for short vowels can be used to annotate text or spell it out in full when desired
• Arabic script is written from right to left
• Arabic script is cursive

بين → بِيُن → بِي ن
جـ جـ جـ جـ
There are two mathematical notations according to the regions:
- Latin notation with Arabic text
- Genuine Arabic notation

Arabic notation
- Spreads from right to left
- Uses Arabic alphabet based symbols
- Uses Arabic abbreviations for usual functions
- Uses some Latin mirrored symbols
- Uses curved kashida for variable sized symbols
There are two mathematical notations according to the regions:

- Latin notation with Arabic text
- Genuine Arabic notation

### Arabic notation

- Spreads from right to left
- Uses Arabic alphabet based symbols

\[
0 = 5 + 3 - 2^2
\]

- Uses Arabic abbreviations for usual functions
- Uses some Latin mirrored symbols
- Uses curved kashida for variable sized symbols
There are two mathematical notations according to the regions:
- Latin notation with Arabic text
- Genuine Arabic notation

**Arabic notation**
- Spreads from right to left
- Uses Arabic alphabet based symbols
- Uses Arabic abbreviations for usual functions

<table>
<thead>
<tr>
<th>Arabic</th>
<th>Latin</th>
</tr>
</thead>
<tbody>
<tr>
<td>جا</td>
<td>Sine</td>
</tr>
<tr>
<td>جتا</td>
<td>Cosine</td>
</tr>
<tr>
<td>دا</td>
<td>Tangente</td>
</tr>
<tr>
<td>تا</td>
<td>Cotangente</td>
</tr>
<tr>
<td>قا</td>
<td>Secante</td>
</tr>
<tr>
<td>قتا</td>
<td>Cosecante</td>
</tr>
</tbody>
</table>

- Uses some Latin mirrored symbols
- Uses curved kashida for variable sized symbols
There are two mathematical notations according to the regions:

- Latin notation with Arabic text
- Genuine Arabic notation

**Arabic notation**

- Spreads from right to left
- Uses Arabic alphabet based symbols
- Uses Arabic abbreviations for usual functions
- Uses some Latin mirrored symbols
- Uses curved kashida for variable sized symbols
Arabic mathematical expressions

There are two mathematical notations according to the regions:

* Latin notation with Arabic text
* genuine Arabic notation

Arabic notation

- Spreads from right to left
- Uses Arabic alphabet based symbols
- Uses Arabic abbreviations for usual functions
- Uses some Latin mirrored symbols
- Uses curved kashida for variable sized symbols
Arabic mathematical expressions

- Arabic alphabet symbols are dotted or dotless

- Uses several styles to extend the amount of symbols
Arabic scientific document composition

**Image based method**
- Pure painting
- Painting and equation editor
- Painting and text layout
- Handwritten equation digitalization

**\TeX\ based method**
- RyDArab and CurExt
Arabic scientific document composition

Image based method
- Pure painting
- Painting and equation editor
- Painting and text layout
- Handwritten equation digitalization

TEX based method
- RyDArab and CurExt
Pure painting

- Uses drawing tool to paint both text and non textual symbols
- The result is cut and pasted to document
Use of letters for alphabet based symbols and drawings for other symbols
Uses an equation editor for Arabic-Latin common notation and drawing for mirrored symbols
Use of scanner to transform both text and mathematical expressions to image
Advantages

- are WYSIWYG
- do not present any linguistic difficulty when used in Arabic versions

Drawbacks

- Bad typographical quality
- Difficult (need some drawing skills and Image handling)
- Not uniform (expressions have not the same metrics)
- No semantic content
Advantages

- $\text{T\LaTeX}$ based systems
  - High quality technical documents
  - Documents can be edited using simple text editor
- Several choices for notations and symbols styles
- Variable sized symbols dimensions are automatically and transparently set
- Expressions can be converted into several formats (image, MathML,...)
- Expression can be edited easily

Drawbacks

- No WYSIWYG interface for $\text{T\LaTeX}$ in Arabic till now
- Need some English and $\text{T\LaTeX}$ learning
Information may be reached, used and communicated in the languages of the transmitter and that of the receiver without considerations of the technical support.

Several trends help to go from monolingualism to multilingualism:
- from ASCII to Unicode
- \( \text{T\AA X} \) Arabic support: Omega, ArabT\AA X, Aleph, Rydarab ...
- Multilingual fonts: STIX project
- XML I18n
- ...

Several studies show that one learns better in his mother tongue.
DadTEX system

DadTEX goals

- Interface for composition of \LaTeX{} documents using only Arabic
- Build Arabic version of \TeX{} commands lexicon
- Make it easy to learn and understand \TeX{} vocabulary for Arabic users
- Avoid bidirectionality problems due to the mixture of English commands and Arabic text
Bidirectionality problems

- Text selection

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
</tr>
</thead>
</table>
| agreement | قافق | "الاتفاق"

- Character position
- Semantic ambiguity
- Due to the graphical swapping
Bidirectionality problems

- Text selection
- Character position
- Semantic ambiguity
  Due to the graphical swapping
Bidirectionality problems

- Text selection
- Character position

- Semantic ambiguity
  Due to the graphical swapping

\catcode`\11=
A conceivable solution is to use an application to convert Arabic document into its transliterated equivalent. This mechanism is similar to the one used in FarsiTEX (ftx2tex translates Persian text to transliterated text)

This solution have several deficiencies:

- it is not direct;
- it generates a supplementary file to be processed instead of the original source file
- the compilation time is increased
- additional memory and free space are required
DadTeX System

- DadTeX is an interface that allows the creation of \LaTeX\ documents in Arabic. The whole of the document can be composed using only Arabic text with some control characters like backslash, dollar, ...

- It is based on the primitive \texttt{\def} for the translation of every commands.

- In some cases commands translation is not sufficient: RydArab uses transliteration for individual letter-based alphabetical symbols it is necessary to redefine the correspondences using Arabic characters.
DadTEX is an interface that allows the creation of \LaTeX documents in Arabic. The whole of the document can be composed using only Arabic text with some control characters like backslash, dollar, ...

- It is based on the primitive `\def` for the translation of every commands.

- In some cases commands translation is not sufficient: RydArab uses transliteration for individual letter-based alphabetical symbols it is necessary to redefine the correspondences using Arabic characters.
DadTEX System

- DadTEX is an interface that allows the creation of \LaTeX\ documents in Arabic. The whole of the document can be composed using only Arabic text with some control characters like backslash, dollar, ...

- It is based on the primitive `\def` for the translation of every commands.

- In some cases commands translation is not sufficient: RydArab uses transliteration for individual letter-based alphabetical symbols it is necessary to redefine the correspondences using Arabic characters.

$ a+b $ \rightarrow $ ب + 1 $
In T\textsc{e}X, besides commands for mathematical objects, there are many commands for document segmentation and content nature specification. The translation of such commands, will make it very easy for Arabic users to understand why commands like \texttt{\textbackslash chapter} should not be used in a standard article.
DadTEX structure

- dadpreamb.tex: a set of commands used in document preambles. The encoding system in use is defined here.
- dadalpha.tex: here, ISO-8859-6 Arabic characters are declared as letters using the primitive `\catcode`
- dadbody.tex: a set of commands used in documents bodies
- dadadapt.tex: DadTEX users can add their own commands. It is like a dictionary of commands. It allows for flexible and customizable translation.
• DadTEX can be used with any encoding system able of representing the Arabic alphabet, as long as the encoding is supported by the packages in use.

• In this version, we used ISO-8859-6 instead of UTF-8 because the ArabTEX system still has some problems when it is used with UTF-8

• In ISO-8859-6, Arabic characters are encoded in one byte. It is thus easy to set their category code into 11.
Encoding

UTF-8

Using \texttt{\textbackslash catcode} with UTF-8 will lead to errors, because it is intended to be used with one-byte characters.

Arabic characters should be divided into there two visible bytes using an ASCII text editor.

\begin{verbatim}
\texttt{\textbackslash catcode``\\Vert=11} \\
\texttt{\textbackslash catcode``\textbackslash 0=11 and \textbackslash catcode``\$=11}.
\end{verbatim}

In the case of the Omega system, the hexadecimal code can be used directly:

\begin{verbatim}
\texttt{\textbackslash catcode``\textasciitilde\textasciitilde\textasciitilde\textasciitilde\textasciitilde0627=11}.
\end{verbatim}
This is an English sample text

{\textbf{large} \ This is a large English text\}

:This is an expression

\[\sqrt{\frac{1+b}{2-f}}\]

"أداة إنجليزية"

"شريحة" (اللغة العربية)

"هذا النص هو مثال لنص عادي:\"
"أعد هذا النص هو مثال لنص عريض:\"
"هذه صيغة رياضية:\"

\[\frac{أجر((\text{كسر}(1+b))(2-f))}{هيئة(عربية)}\]

"أداة إنجليزية"
DadTEX advantages

DadTEX is

- based completely on TEx
- crossplatform supported
- compatible with several other TEx extensions
- allows composition of documents using only Arabic text and commands
- can be adapted easily to regional needs and users choices
- can be generalized to several other non Latin languages
In \TeX{}, numbers are used to control document component sizes, and action’s frequency, etc. The support of Arabic Indic numbers or Persian numbers in control sequences is needed

- The use of file names in Arabic is still a problem
- In a future version of Dad\TeX{}, it will be very interesting to add support for other packages (e.g. Arabi, ...)
- Further steps are to be done in the I18n field:
  - Exploiting the linguistic and regional properties from the system settings and getting notational preferences (such as units, currency,...)
  - Take maximum advantage of Unicode’s bidirectionality algorithm like it is done in browsers, where no specific declaration of the language is required
Open problems and prospects

- In \TeX, numbers are used to control document component sizes, and action’s frequency, etc. The support of Arabic Indic numbers or Persian numbers in control sequences is needed.
- The use of file names in Arabic is still a problem.
- In a future version of Dad\TeX, it will be very interesting to add support for other packages (e.g. Arabi, ...)
- Further steps are to be done in the I18n field:
  - Exploiting the linguistic and regional properties from the system settings and getting notational preferences (such as units, currency,...)
  - Take maximum advantage of Unicode’s bidirectionality algorithm like it is done in browsers, where no specific declaration of the language is required.
In TeX, numbers are used to control document component sizes, and action’s frequency, etc. The support of Arabic Indic numbers or Persian numbers in control sequences is needed.

The use of file names in Arabic is still a problem.

In a future version of DadTeX, it will be very interesting to add support for other packages (e.g. Arabi, ...)

Further steps are to be done in the I18n field:

- Exploiting the linguistic and regional properties from the system settings and getting notational preferences (such as units, currency,...)

- Take maximum advantage of Unicode’s bidirectionality algorithm like it is done in browsers, where no specific declaration of the language is required.
Open problems and prospects

- In \TeX{}, numbers are used to control document component sizes, and action’s frequency, etc. The support of Arabic Indic numbers or Persian numbers in control sequences is needed.
- The use of file names in Arabic is still a problem.
- In a future version of Dad\TeX{}, it will be very interesting to add support for other packages (e.g. Arabi, ...)
- Further steps are to be done in the I18n field:
  - Exploiting the linguistic and regional properties from the system settings and getting notational preferences (such as units, currency,...)
  - Take maximum advantage of Unicode’s bidirectionality algorithm like it is done in browsers, where no specific declaration of the language is required.
In TeX, numbers are used to control document component sizes, and action's frequency, etc. The support of Arabic Indic numbers or Persian numbers in control sequences is needed.

The use of file names in Arabic is still a problem.

In a future version of DadTeX, it will be very interesting to add support for other packages (e.g. Arabi, ...)

Further steps are to be done in the I18n field:
  - Exploiting the linguistic and regional properties from the system settings and getting notational preferences (such as units, currency,...)

  - Take maximum advantage of Unicode's bidirectionality algorithm like it is done in browsers, where no specific declaration of the language is required.
The End
Thank you!