

DadT_EX, a full Arabic interface

M. Eddahibi, A. Lazrek, K. Sami

Department of Computer Science
Cadi Ayyad University

Marrakesh

November 9 - 11, 2006



TUG2006

- 1 Arabic writing
- 2 Arabic mathematical expressions
- 3 Arabic scientific document composition
- 4 DadT_EX motivations
- 5 DadT_EX 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 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 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

ك ت ب

Logical order



ك ت ب

Visual order



- Arabic script is cursive

- 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 → بـ يـ ن → ب ي ن

ج ج ج ج

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

$$0 = 5 + 3س - 2س^2$$

- 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

Sine	جا	Cotangente	ظنا
Cosine	جنا	Secante	قا
Tangente	ظا	Cosecante	قتا

- Uses some Latin mirrored symbols
- Uses curved kashida for variable sized symbols

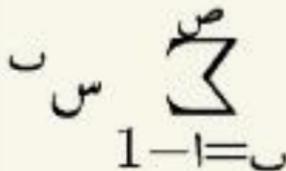
Arabic mathematical expressions

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



The image shows two examples of Arabic mathematical notation. On the left, the Arabic letter 'س' (S) is written twice, once above and once below a horizontal line. On the right, a summation symbol is shown, consisting of a large 'س' (S) above a summation symbol (Σ) and a horizontal line with '1' on the left and 'س' (S) on the right.

- 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

$$\begin{array}{c} \text{ص} \\ \text{س} \text{ ————— } \text{ج} \\ 1 - 1 = \text{س} \end{array}$$

$$\begin{array}{c} \text{ص} \\ \text{س} \text{ ————— } \text{ج} \\ 1 - 1 = \text{س} \end{array}$$

Arabic mathematical expressions

- Arabic alphabet symbols are dotted or dotless

حاس + طا ص جاس + ظا ص

- Uses several styles to extend the amount of symbols

ا، ب، ح، د، و، ز، ط، ي، ل، م، ن، س، ع، ف، ص، و

ة، لا، ك، هـ، ر، م، ر، ا، ح، و، ك، ل، هـ، س، ع، و، ص

ا، ب، ح، د، و، ز، ط، ي، ل، م، ن، س، ع، ف، ص، و

ا، ب، ح، د، و، ز، ط، ي، ل، م، ن، س، ع، ف، ص، و

ة، لا، ك، هـ، ر، م، ر، ا، ح، و، ك، ل، هـ، س، ع، و، ص

Image based method

- Pure painting
- Painting and equation editor
- Painting and text layout
- Handwritten equation digitalization

$\text{T}_\text{E}\text{X}$ based method

- RyDArab and CurExt

Image based method

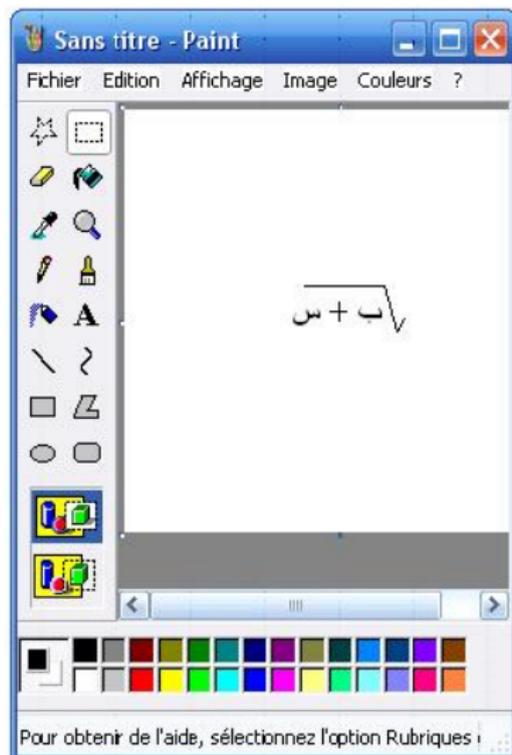
- Pure painting
- Painting and equation editor
- Painting and text layout
- Handwritten equation digitalization

$\text{T}_\text{E}\text{X}$ based method

- RyDArab and CurExt

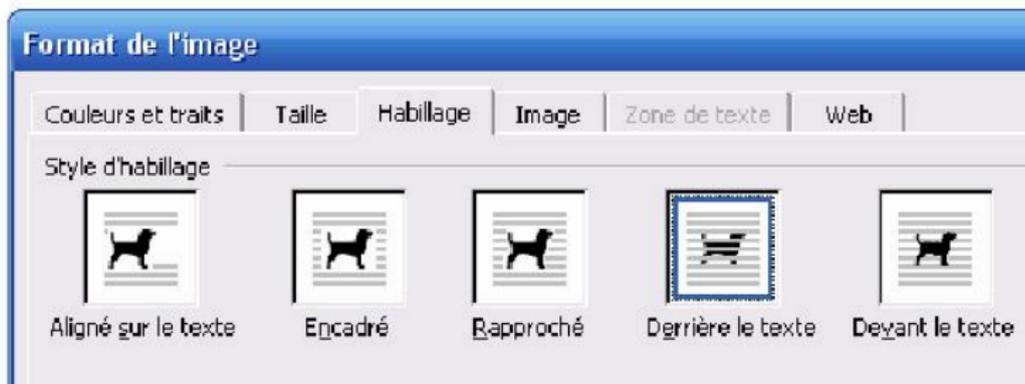
Pure painting

- Uses drawing tool to paint both text and non textual symbols
- The result is cut and pasted to document



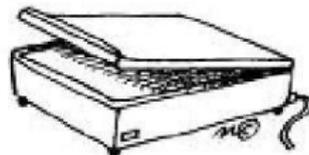
Painting and text layout

Use of letters for alphabet based symbols and drawings for other symbols



Handwritten equation digitalization

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

- T_EX 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 T_EX in Arabic till now
- Need some English and T_EX 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
 - T_EX Arabic support: Omega, ArabT_EX, Aleph, Rydarab ...
 - Multilingual fonts: STIX project
 - XML I18n
 - ...
- Several studies show that one learns better in his mother tongue

DadT_EX goals

- Interface for composition of L^AT_EX documents using only Arabic
- Build Arabic version of T_EX commands lexicon
- Make it easy to learn and understand T_EX vocabulary for Arabic users
- Avoid bidirectionality problems due to the mixture of English commands and Arabic text

Bidirectionality problems

- Text selection

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
a	g	r	e	e	m	e	n	t	ا	ل	ا	ت	ا	ف	ق

agreement الاتفاق

- Character position
- Semantic ambiguity
Due to the graphical swapping

Bidirectionality problems

- Text selection
- Character position

agreement|الاتفاق

- Semantic ambiguity
Due to the graphical swapping

Bidirectionality problems

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

```
\catcode '\11=ب
```

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 FarsiT_EX (`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

- DadT_EX is an interface that allows the creation of L^AT_EX 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.

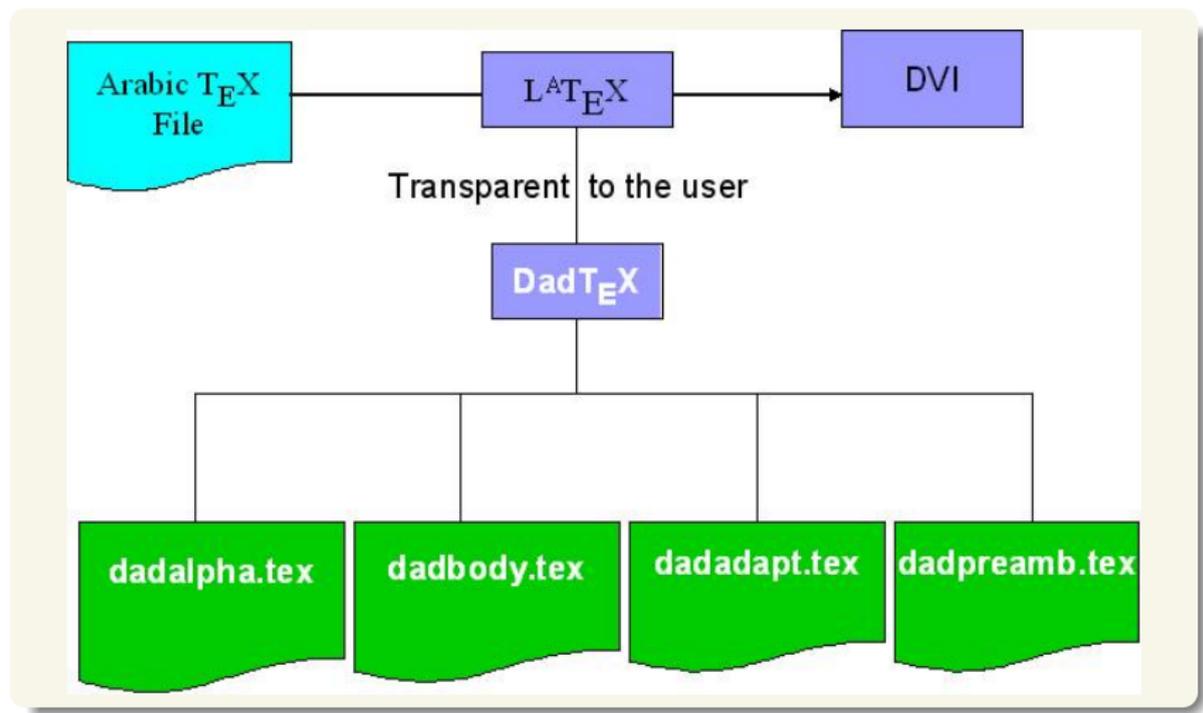
- DadT_EX is an interface that allows the creation of L^AT_EX documents in Arabic. The whole of the document can be composed using only Arabic text with some control characters like backslash, dollar, ...
- It is based on the primitive `\def` for the translation of every commands.
- In some cases commands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

- DadT_EX is an interface that allows the creation of L^AT_EX 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.

$\$a+b\$ \longrightarrow \$ب+ا\$$

In T_EX, 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 `\chapter` should not be used in a standard article.

DadT_EX 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`: DadT_EX users can add their own commands. It is like a dictionary of commands. It allows for flexible and customizable translation.

- Dad \TeX 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 Arab \TeX 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.

UTF-8

Using `\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.

```
\catcode`\|=11
```

```
\catcode`\ø=11 and \catcode`\$=11 .
```

In the case of the Omega system, the hexadecimal code can be used directly: `\catcode`^^^^0627=11` .

DadT_EX is

- based completely on T_EX
- crossplatform supported
- compatible with several other T_EX 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

Open problems and prospects

- In $\text{T}_{\text{E}}\text{X}$, 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 $\text{D}_{\text{a}}\text{T}_{\text{E}}\text{X}$, it will be very interesting to add support for other packages (e.g. `Arabi`, ...)
- Further steps are to be done in the l18n 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 T_EX, 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 DadT_EX, it will be very interesting to add support for other packages (e.g. Arabi, ...)
- Further steps are to be done in the l18n 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 $\text{T}_{\text{E}}\text{X}$, 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 $\text{DadT}_{\text{E}}\text{X}$, it will be very interesting to add support for other packages (e.g. `Arabi`, ...)
- Further steps are to be done in the l18n 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 T_EX, 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 DadT_EX, it will be very interesting to add support for other packages (e.g. Arabi, ...)
- **Further steps are to be done in the l18n 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 $\text{T}_{\text{E}}\text{X}$, 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 $\text{DadT}_{\text{E}}\text{X}$, it will be very interesting to add support for other packages (e.g. `Arabi`, ...)
- Further steps are to be done in the l18n 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!