**coopterteam** software | *Application Migration Suite*
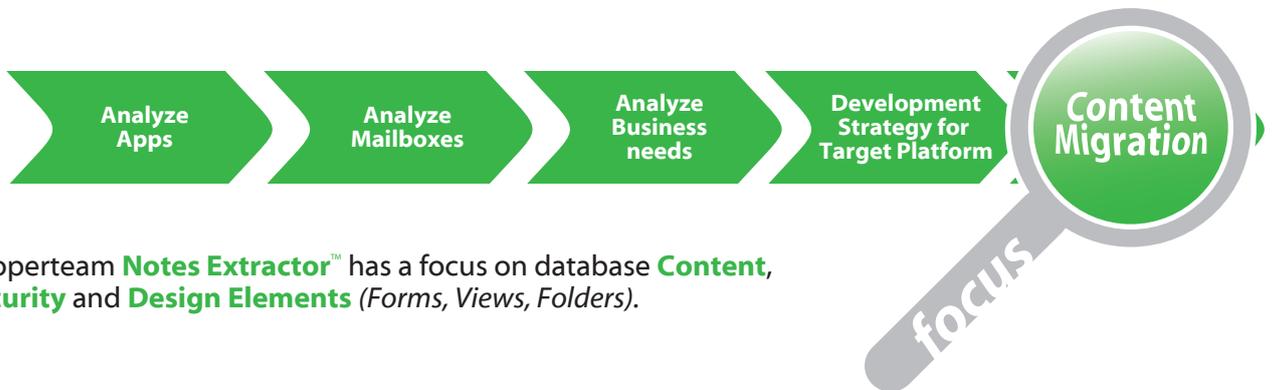
# Notes Extractor

## EXPORT DOMINO CONTENT, DESIGN & VIEWS FROM LOTUS NOTES

**Introduction**

Cooperteam **Notes Extractor**™ is part of the overall Cooperteam methodology to Migrate Applications to a different target infrastructure.

| Analyze Apps | Analyze Mailboxes | Analyze Business needs | Development Strategy for Target Platform | Content Migration |

*focus*

Cooperteam **Notes Extractor**™ has a focus on database **Content**, **Security** and **Design Elements** *(Forms, Views, Folders)*.

**Content**

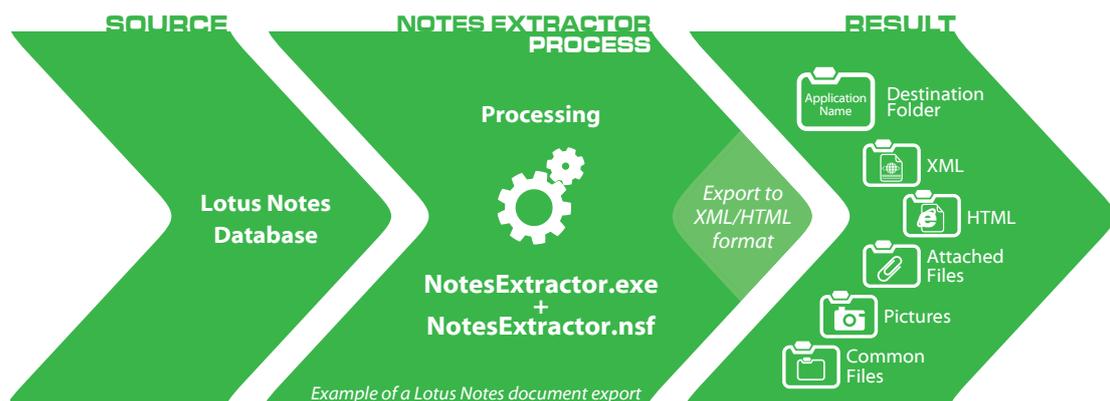### How does Cooperteam Notes Extractor handle Content Migration?

Content is made out of text, images, attach files, Document links, etc. In order to get this out of Domino the process is as follows:

**Notes Extractor**™ is a Domino based utility to extract Data *(all Notes Document content)* from a Notes database and to convert this data into an **'open'** format:

- ⑤ XML is used to store the structured data *(TEXT, NUMBER, TIME/DATE)*
- ⑤ HTML is used for the non-structured data *(Rich Text, Mime)*.
- ⑤ Images/Pictures are translated into .GIF or .JPEG files. Single Instance Storage technology is applied.
- ⑤ Attach files and Embedded Objects are extracted and will keep their orginial format.
- ⑤ Notes links *(doc/view/db links)* are converted to standard URL
- ⑤ The sections are maintained and *(found in Rich Text Fields)* are displayed in SharePoint as they are in Notes.

*Each Notes document is rendered as a set of files (one XML + one HTML per Rich Text or Mime field + several Attach files).*

*The document information buffer (IDs, creation / modification Time/Date, Size...) is kept into the XML file.*
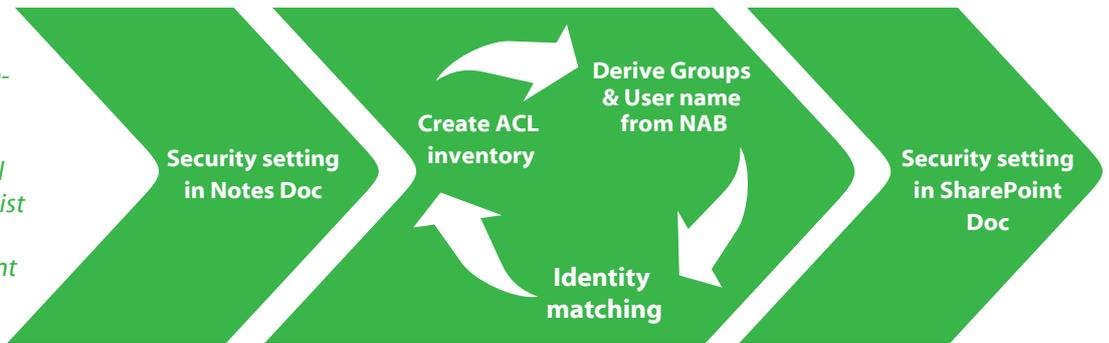
**SOURCE**

**Lotus Notes Database**

**NOTES EXTRACTOR PROCESS**

**Processing**

*Export to XML/HTML format*

**NotesExtractor.exe + NotesExtractor.nsf**

*Example of a Lotus Notes document export*

**RESULT**

Application Name — Destination Folder

XML

HTML

Attached Files

Pictures

Common Files

**Tech Corner**

**Security**

## How does Cooperteam Notes Extractor handle Security?

The Lotus Notes Security model is based on database ACL *(user name, groups, role)*, document security fields *(reader/author)* and NAB directory *(Groups definitions)*. In order to get these security challenges out of Domino the process is as follows:

- The user identity *(Notes FullName)* is translated into the target user identity *(Active Directory UPN if the target is Sharepoint, LDAP Names if the Target is open system...)*. We also translate the Group names as is.
- Notes Extractor converts the document access level *(who can read/write/delete)* into a SharePoint 'ready' security format.
- The document security layer *(Authors/Readers fields)* is integrated in the XML file into a dedicted 'Security' section.

*No matter how complex the Grouping is, the ACL settings, roles etc, Notes Extractor will always deliver the list of user credentials ready for SharePoint import.*

Security setting in Notes Doc → Create ACL inventory → Derive Groups & User name from NAB → Identity matching → Security setting in SharePoint Doc

**Design Elements**

## How does Cooperteam Notes Extractor handle Design Elements?

When we recognize a standard Notes database *(Team Room, Document Library, Discussion...)*, there is no need to convert any Design Elements because such applications already exist in the SharePoint world. For all other Applications, Design Elements conversion is required to render the Content in a readable and accessible way for the end user. In order to render this content the process is as follows:

| About Forms | About Views |
|---|---|
| For each document Type *(Form used)*, we create a dedicated Form that is used in SharePoint *(XSLT)* to display and edit the document. The SharePoint user of the application can see the document as it was seen in Notes.<br><br>⚙ *Same look and feel (same X-Y location on the screen, same color, tables remain the same, etc.).*<br>⚙ *Same Fields*<br>⚙ *Same Data*<br>⚙ *Same Security* | For each document Type *(Form used)*, we create a dedicated View where all documents using this Form are displayed.<br><br>⇄ *Automatically create a set of columns*<br>⇄ *The columns display the content of the most important items of the documents (based on what exists in the Notes databases views)*<br>⇄ *The columns are ordered like in the original view (we use the look & feel of the Notes view to know what to display in which order and with the size of the columns)* |