

ADI06

DXL: Working with Domino Data in XML

Janet Thomas and Dave Schlesinger
DevTools Group, Lotus

Agenda

- DXL
- Lotus XML Toolkit
- Demos
- Questions

What is DXL?

- Domino XML Language
- Domino data expressed in XML
- A universal format for Domino data

```
<document form='Memo'>
  <noteinfo unid='9C93469B4BFC2081852567AE00559882'>
    <created><datetime>19991205T091500,00-04</datetime></created>
  </noteinfo>
  <item name='Subject'><text>DXL Talk</text></item>
  <item name='Sent'>
    <datetime dst='true'>19991205T091500,00-04</datetime></item>
  <item name='From'><text>Janet Thomas</text></item>
  <item name='Body'><richtext>
    <par def='1'>
      <run><font style='italic' />Please</run> come to our talk!
    </par></richtext></item>
</document>
```

Why DXL?

- XML is good for describing information
- XML is easy to work with
 - widely available tools
 - powerful declarative tools
- XML schema's are "APIs" to data
- XML is a good way to work with Domino data: DXL

DXL Uses

- Export
- Import
- Modification
- External Tool

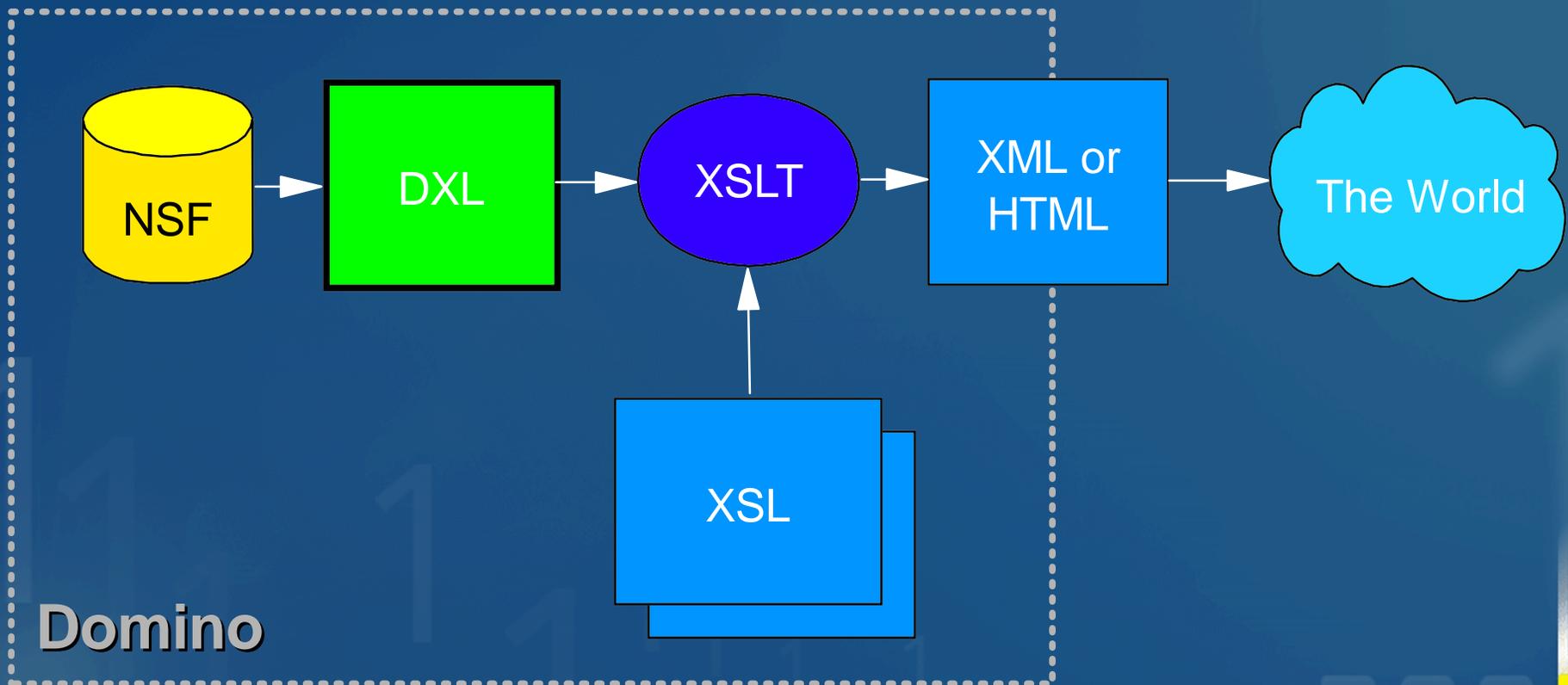
Use: Export

- Data exported from NSF as DXL



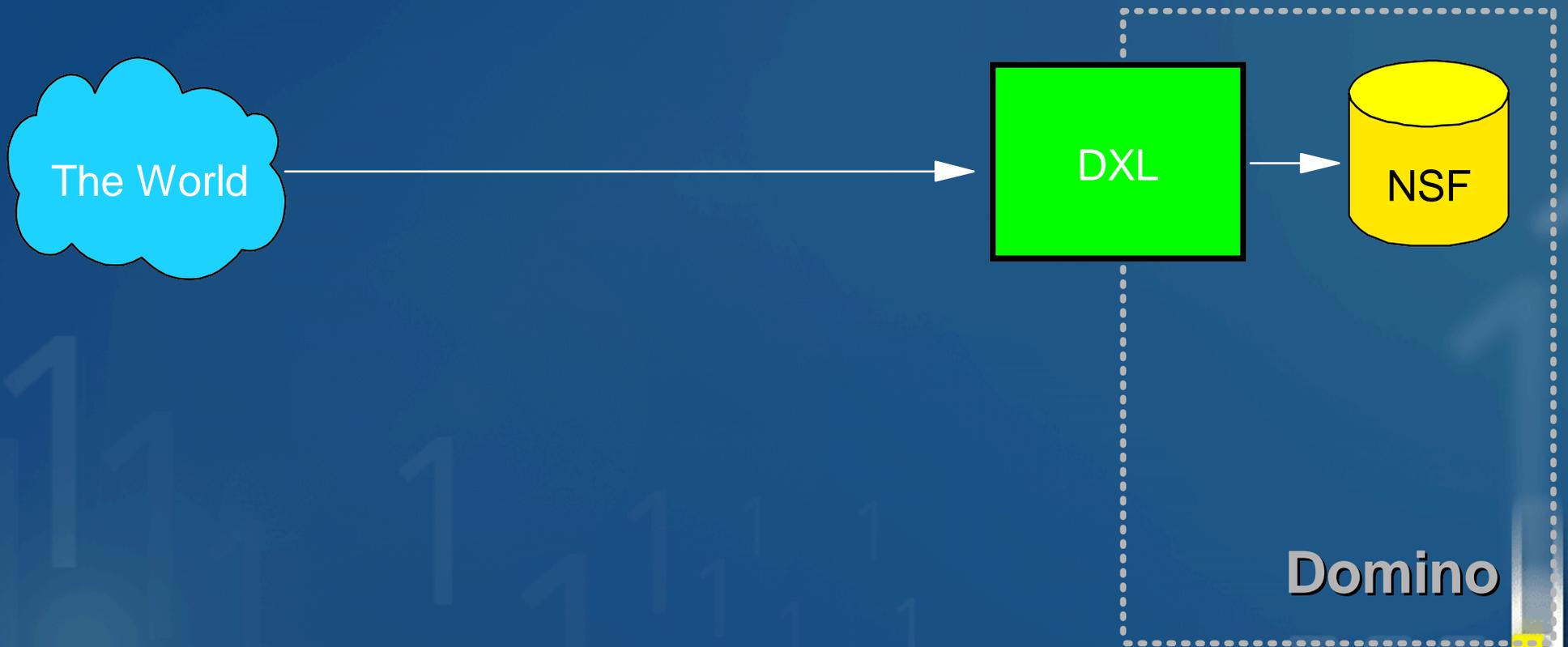
Use: Export

- Data exported from NSF as DXL
- Optionally transformed via XSLT



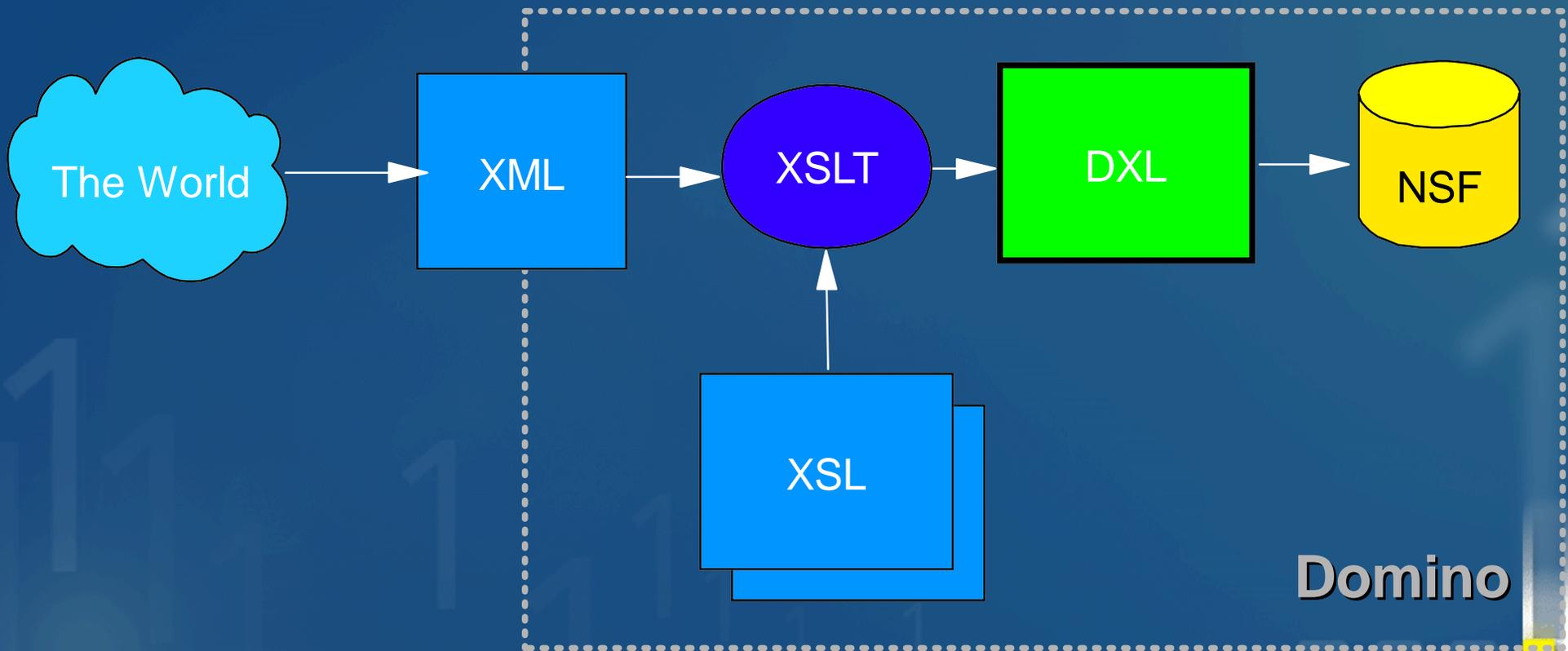
Use: Import

- External DXL imported into NSF



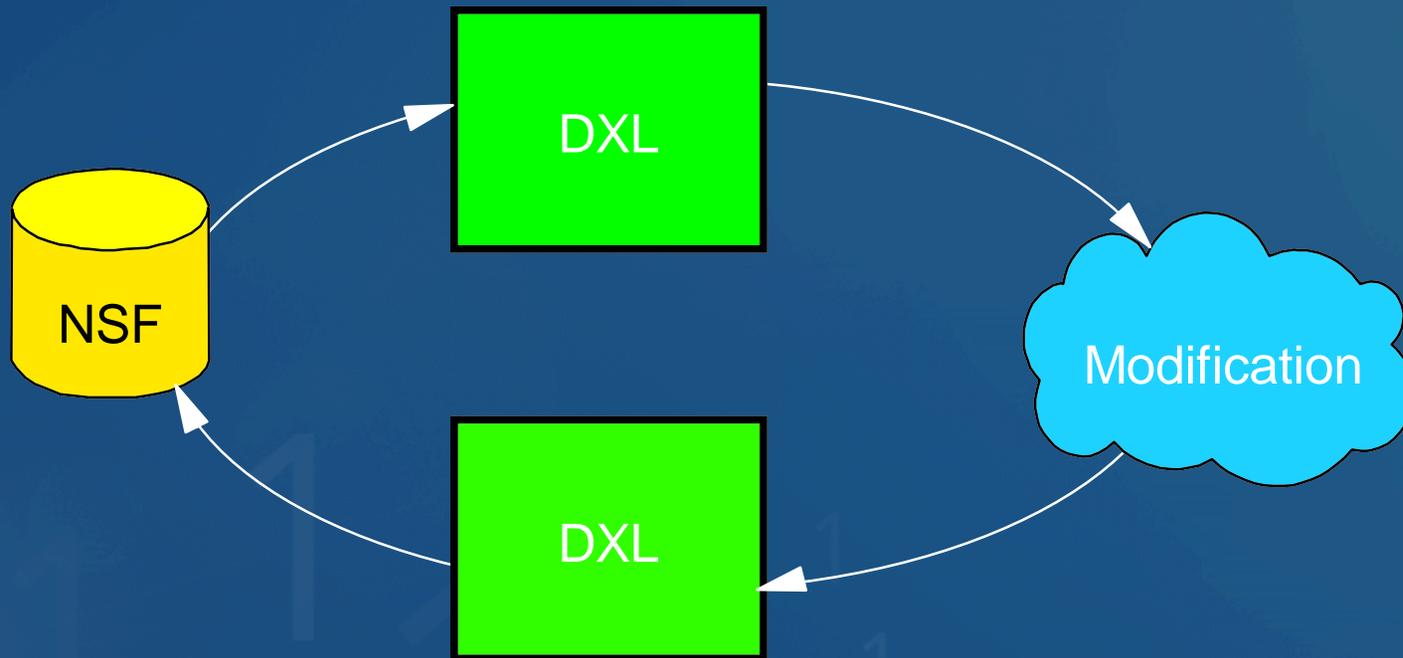
Use: Import

- External DXL imported into NSF
- Could have been transformed from other XML



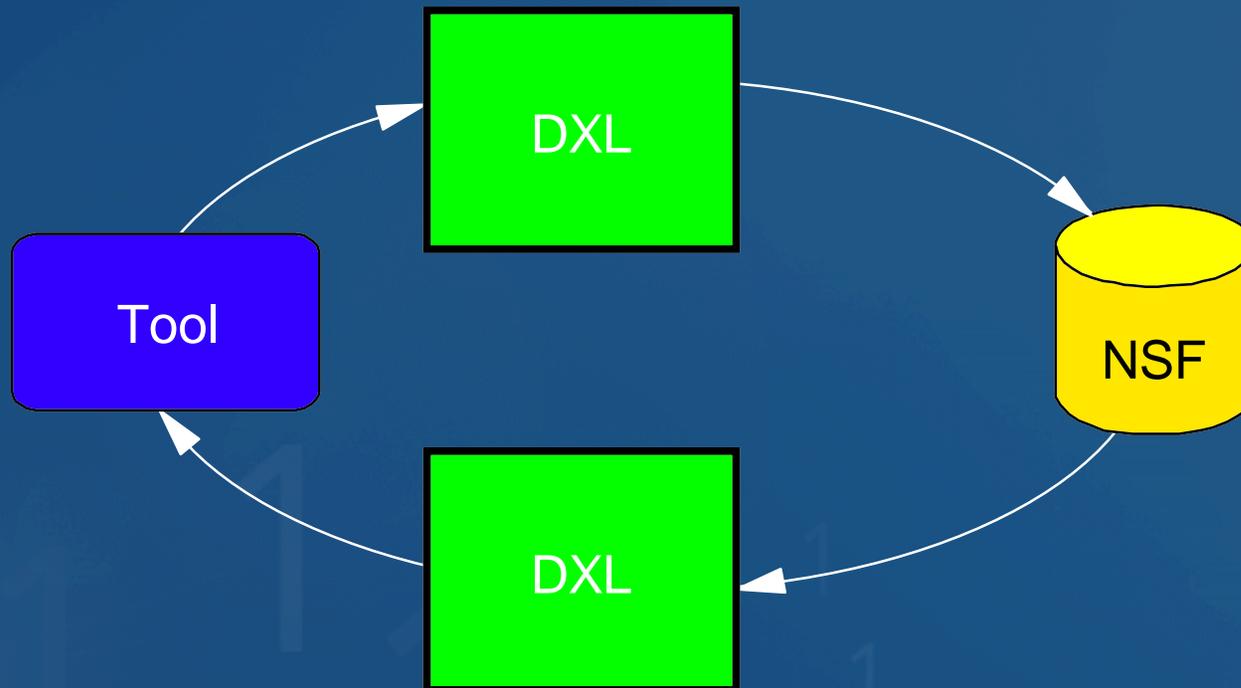
Use: Modification

- Data exported to DXL
- DXL modified (in whatever way you like)
- Data imported back into NSF



Use: External Tool

- Third-party tool uses DXL to create Domino content



DXL Goals

- Complete
- Lossless
- Approachable
- Transformable

Goal: Complete

- Express all Domino data
 - documents
 - design elements
 - everything else

Goal: Lossless

- Preserve as much information as possible
- Get Domino data out to the world
- Get data into Domino
- Enable round-tripping through XML

Goal: Approachable

- A Domino developer can read (and write) it
- You don't have to worry about "Domino secrets"
- No more specially-named items!!

Goal: Transformable

- Should be transformable with XSLT
 - into other XML
 - into other DXL
 - into HTML

DXL Non-Goals

- DXL cannot be directly displayed in a browser
 - it isn't HTML
- DXL is not stored in NSF
 - it isn't on-disk
- Domino app semantics won't be built with XSL
 - it doesn't replace HTTP

DXL Challenges: Levels of Representation

- An appointment...
- ...is really a mail memo...
- ...which is actually a document...
- ...which is implemented as a note.
- So: <appointment> or <memo> or <document> or <note>?
- Answer: <document>

DXL Challenges: Rich Text

- Needs to be understandable
- Needs to represent all of Domino rich text
- Use HTML?
- Encode CD records?
- Answer: our own tag language

```
<document form='Memo'>
  <noteinfo unid='9C93469B4BFC2081852567AE00559882'>
    <created><datetime>19991205T091500,00-04</datetime></created>
  </noteinfo>
  <item name='Subject'><text>DXL Talk</text></item>
  <item name='Sent'>
    <datetime dst='true'>19991205T091500,00-04</datetime></item>
  <item name='From'><text>Janet Thomas</text></item>
  <item name='Body'><richtext>
    <par def='1'>
      <run><font style='italic' />Please</run> come to our talk!
    </par></richtext></item>
</document>
```

```
<document form='Memo'>
  <noteinfo unid='9C93469B4BFC2081852567AE00559882'>
    <created><datetime>19991205T091500,00-04</datetime></created>
  </noteinfo>
  <item name='Subject'><text>DXL Talk</text></item>
  <item name='Sent'>
    <datetime dst='true'>19991205T091500,00-04</datetime></item>
  <item name='From'><text>Janet Thomas</text></item>
  <item name='Body'><richtext>
    <par def='1'>
      <run><font style='italic' />Please</run> come to our talk!
    </par></richtext></item>
</document>
```

```
<document form='Memo'>
  <noteinfo unid='9C93469B4BFC2081852567AE00559882'>
    <created><datetime>19991205T091500,00-04</datetime></created>
  </noteinfo>
  <item name='Subject'><text>DXL Talk</text></item>
  <item name='Sent'>
    <datetime dst='true'>19991205T091500,00-04</datetime></item>
  <item name='From'><text>Janet Thomas</text></item>
  <item name='Body'><richtext>
    <par def='1'>
      <run><font style='italic' />Please</run> come to our talk!
    </par></richtext></item>
</document>
```

```
<document form='Memo'>
  <noteinfo unid='9C93469B4BFC2081852567AE00559882'>
    <created><datetime>19991205T091500,00-04</datetime></created>
  </noteinfo>
  <item name='Subject'><text>DXL Talk</text></item>
  <item name='Sent'>
    <datetime dst='true'>19991205T091500,00-04</datetime></item>
  <item name='From'><text>Janet Thomas</text></item>
  <item name='Body'><richtext>
    <par def='1'>
      <run><font style='italic' />Please</run> come to our talk!
    </par></richtext></item>
</document>
```

```
<document form='Memo'>
  <noteinfo unid='9C93469B4BFC2081852567AE00559882'>
    <created><datetime>19991205T091500,00-04</datetime></created>
  </noteinfo>
  <item name='Subject'><text>DXL Talk</text></item>
  <item name='Sent'>
    <datetime dst='true'>19991205T091500,00-04</datetime></item>
  <item name='From'><text>Janet Thomas</text></item>
  <item name='Body'><richtext>
    <par def='1'>
      <run><font style='italic' />Please</run> come to our talk!
    </par></richtext></item>
</document>
```

```
<document form='Memo'>
  <noteinfo unid='9C93469B4BFC2081852567AE00559882'>
    <created><datetime>19991205T091500,00-04</datetime></created>
  </noteinfo>
  <item name='Subject'><text>DXL Talk</text></item>
  <item name='Sent'>
    <datetime dst='true'>19991205T091500,00-04</datetime></item>
  <item name='From'><text>Janet Thomas</text></item>
  <item name='Body'><richtext>
    <par def='1'>
      <run><font style='italic' />Please</run> come to our talk!
    </par></richtext></item>
</document>
```

```
<form name='Mini Doc' default='true' mailable='true'>
  <noteinfo>...</noteinfo>
  <code event='windowtitle'>
    <formula>@If(@IsNewDoc; "New Memo"; Subject)</formula>
  </code>
  <actionbar><action .../>...</actionbar>
  <body><richtext>
    <par def='1'>
      Subject:
      <field type='text' kind='editable' name='Subject'>
        <code event='defaultvalue'>
          <formula>"New Document"</formula></code>
        </field>
      </par></richtext></body>
</form>
```

```
<form name='Mini Doc' default='true' mailable='true'>
  <noteinfo>...</noteinfo>
  <code event='windowtitle'>
    <formula>@If(@IsNewDoc; "New Memo"; Subject)</formula>
  </code>
  <actionbar><action .../>...</actionbar>
  <body><richtext>
    <par def='1'>
      Subject:
      <field type='text' kind='editable' name='Subject'>
        <code event='defaultvalue'>
          <formula>"New Document"</formula></code>
        </field>
      </par></richtext></body>
</form>
```

```
<form name='Mini Doc' default='true' mailable='true'>
  <noteinfo>...</noteinfo>
  <code event='windowtitle'>
    <formula>@If(@IsNewDoc; "New Memo"; Subject)</formula>
  </code>
  <actionbar><action .../>...</actionbar>
  <body><richtext>
    <par def='1'>
      Subject:
      <field type='text' kind='editable' name='Subject'>
        <code event='defaultvalue'>
          <formula>"New Document"</formula></code>
        </field>
      </par></richtext></body>
</form>
```

```
<form name='Mini Doc' default='true' mailable='true'>
  <noteinfo>...</noteinfo>
  <code event='windowtitle'>
    <formula>@If(@IsNewDoc; "New Memo"; Subject)</formula>
  </code>
  <actionbar><action .../>...</actionbar>
  <body><richtext>
    <par def='1'>
      Subject:
      <field type='text' kind='editable' name='Subject'>
        <code event='defaultvalue'>
          <formula>"New Document"</formula></code>
        </field>
      </par></richtext></body>
</form>
```

```
<form name='Mini Doc' default='true' mailable='true'>
  <noteinfo>...</noteinfo>
  <code event='windowtitle'>
    <formula>@If(@IsNewDoc; "New Memo"; Subject)</formula>
  </code>
  <actionbar><action .../>...</actionbar>
  <body><richtext>
    <par def='1'>
      Subject:
      <field type='text' kind='editable' name='Subject'>
        <code event='defaultvalue'>
          <formula>"New Document"</formula></code>
        </field>
      </par></richtext></body>
</form>
```

```
<form name='Mini Doc' default='true' mailable='true'>
  <noteinfo>...</noteinfo>
  <code event='windowtitle'>
    <formula>@If(@IsNewDoc; "New Memo"; Subject)</formula>
  </code>
  <actionbar><action .../>...</actionbar>
  <body><richtext>
    <par def='1'>
      Subject:
      <field type='text' kind='editable' name='Subject'>
        <code event='defaultvalue'>
          <formula>"New Document"</formula></code>
        </field>
      </par></richtext></body>
</form>
```

```
<form name='Mini Doc' default='true' mailable='true'>
  <noteinfo>...</noteinfo>
  <code event='windowtitle'>
    <formula>@If(@IsNewDoc; "New Memo"; Subject)</formula>
  </code>
  <actionbar><action .../>...</actionbar>
  <body><richtext>
    <par def='1'>
      Subject:
      <field type='text' kind='editable' name='Subject'>
        <code event='defaultvalue'>
          <formula>"New Document"</formula></code>
        </field>
      </par></richtext></body>
</form>
```

```
<form name='Mini Doc' default='true' mailable='true'>
  <noteinfo>...</noteinfo>
  <code event='windowtitle'>
    <formula>@If(@IsNewDoc; "New Memo"; Subject)</formula>
  </code>
  <actionbar><action .../>...</actionbar>
  <body><richtext>
    <par def='1'>
      Subject:
      <field type='text' kind='editable' name='Subject'>
        <code event='defaultvalue'>
          <formula>"New Document"</formula></code>
        </field>
      </par></richtext></body>
</form>
```

DXL in Domino

- Some DXL is available today
 - ?ReadViewEntries (5.02)
 - Document.generateXML (5.03)
 - Text, numbers, and date/times only
- Lots more in future feature releases

Lotus XML Toolkit

- Build applications that export and import DXL
 - Documents, rich text, and design elements
- Class libraries for Java and C++
 - Can use with Notes Java/C++ applications
 - Java classes wrap C++ classes
- Sample programs
- DXL schema (DTD)
- Documentation for class libraries and schema
- Works with R5.0 and later

Lotus XML Toolkit Status

- Evaluation build available now on <http://www.lotus.com/xml>
- C++ classes to export and import documents, rich text, forms, pages, views, ACLs
- Samples: DXLExport and DXLImport
- DXL schema
- Limited documentation (readme files)

Lotus XML Toolkit Status *(continued)*

- Release 1.0 available later this year
 - Java classes
 - More design elements (but not all of them)
 - More samples
 - More documentation (NSF, HTML)
- After release 1.0:
 - LotusScript classes (LSX)
 - More design elements
 - More samples

Who Can Export/Import Design Elements?

- XML Toolkit applications are Notes API programs
 - Use Notes ID for authentication and access
 - Designer access required to import design elements
- Additional security for Toolkit applications:
 - Designer access required to export design elements
 - Cannot export design from hidden-design database
 - Cannot sign imported design elements with server ID

Command-line Tools

- DXLExport: exports NSF data to DXL
- DXLImport: imports DXL into NSF
- Export/import documents, rich text, and design elements
- Built using Lotus XML Toolkit
 - Included as sample programs, with source code

Other Tools

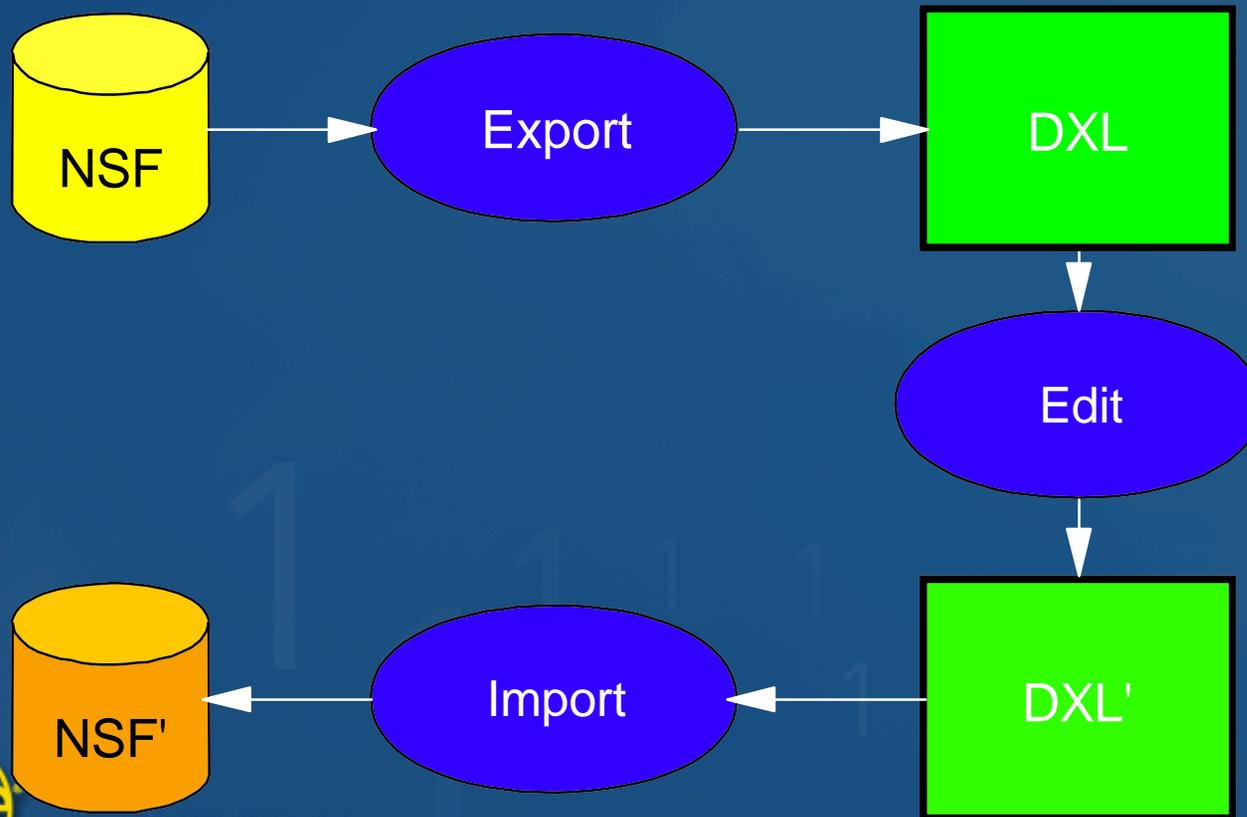
- Xalan (LotusXSL)
 - Transform XML using XSL stylesheets
 - Java and C++ (beta)
- Xerces (XML4C, XML4J)
 - XML parser
 - Java and C++
- Available from
 - <http://xml.apache.org> (Xalan and Xerces)
 - <http://www.alphaworks.ibm.com> (LotusXSL, XML4C, XML4J)

Demos

- Search and replace
- Design synopsis
- Hands-off application design

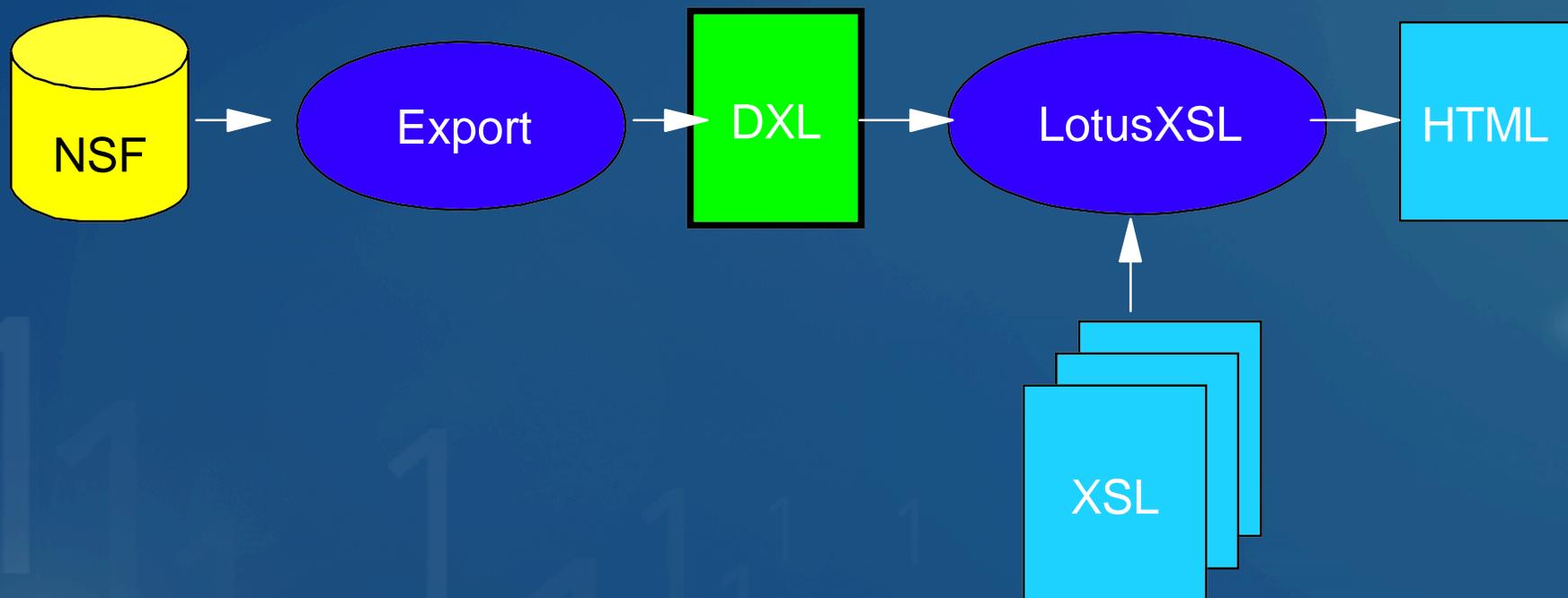
Search and Replace Demo

- Export documents and design elements
- Modify DXL using text editor
- Import modified DXL



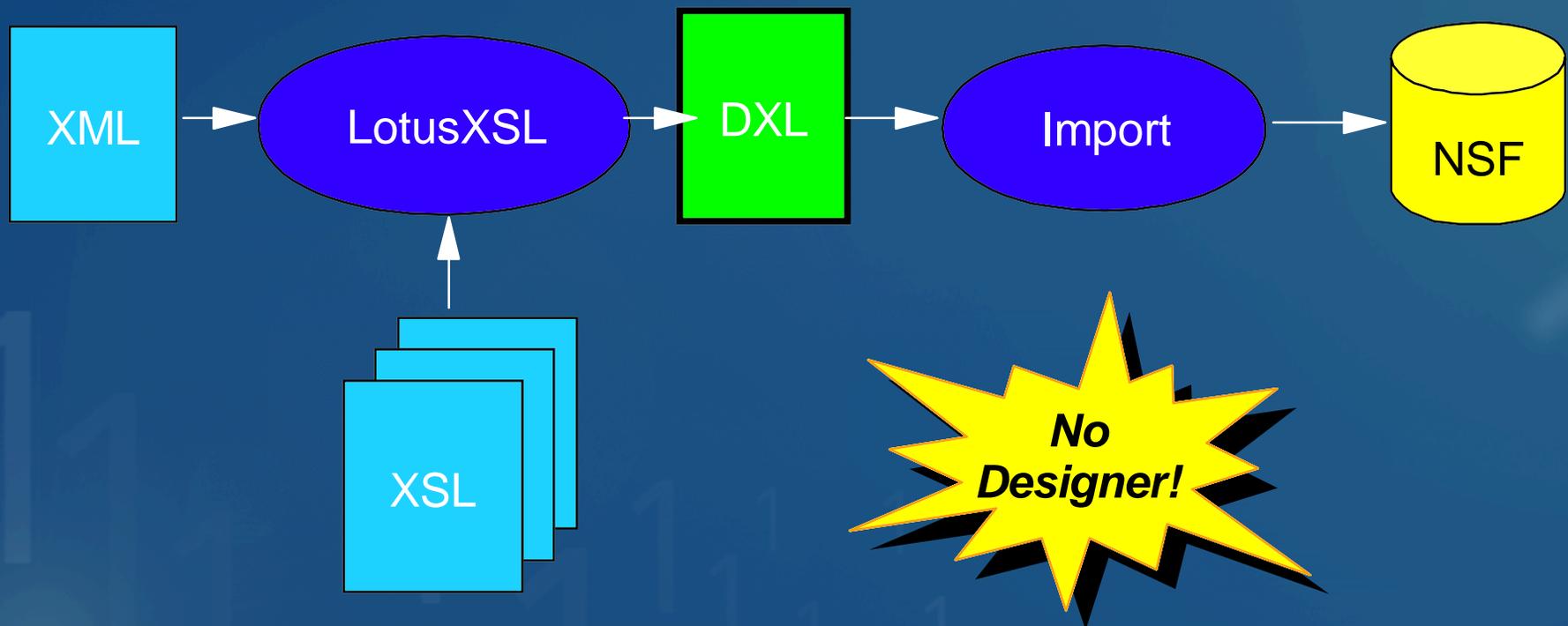
Design Synopsis Demo

- Export design information
- Transform into a readable page



Hands-Off Application Design Demo

- Forms and views created without the Designer
- Simple app description transformed into DXL



Summary

- DXL brings the power of XML to Domino information
- Domino provides some DXL features today
 - Export only
 - The future will have much more
- The Lotus XML Toolkit allows you to build applications that work with DXL
 - Import and enhanced export
 - Rich text and design elements

Downloads

- Lotus XML Toolkit evaluation build
 - <http://www.lotus.com/xml>
- Demo package (from DevCon)
 - DXL schema
 - DXLExport.exe and DXLImport.exe
 - Demos from this session
 - <http://notes.net/dxl>
- Feedback in the Notes/Domino Gold Release Forum
 - <http://notes.net/46dom.nsf>

Questions?

