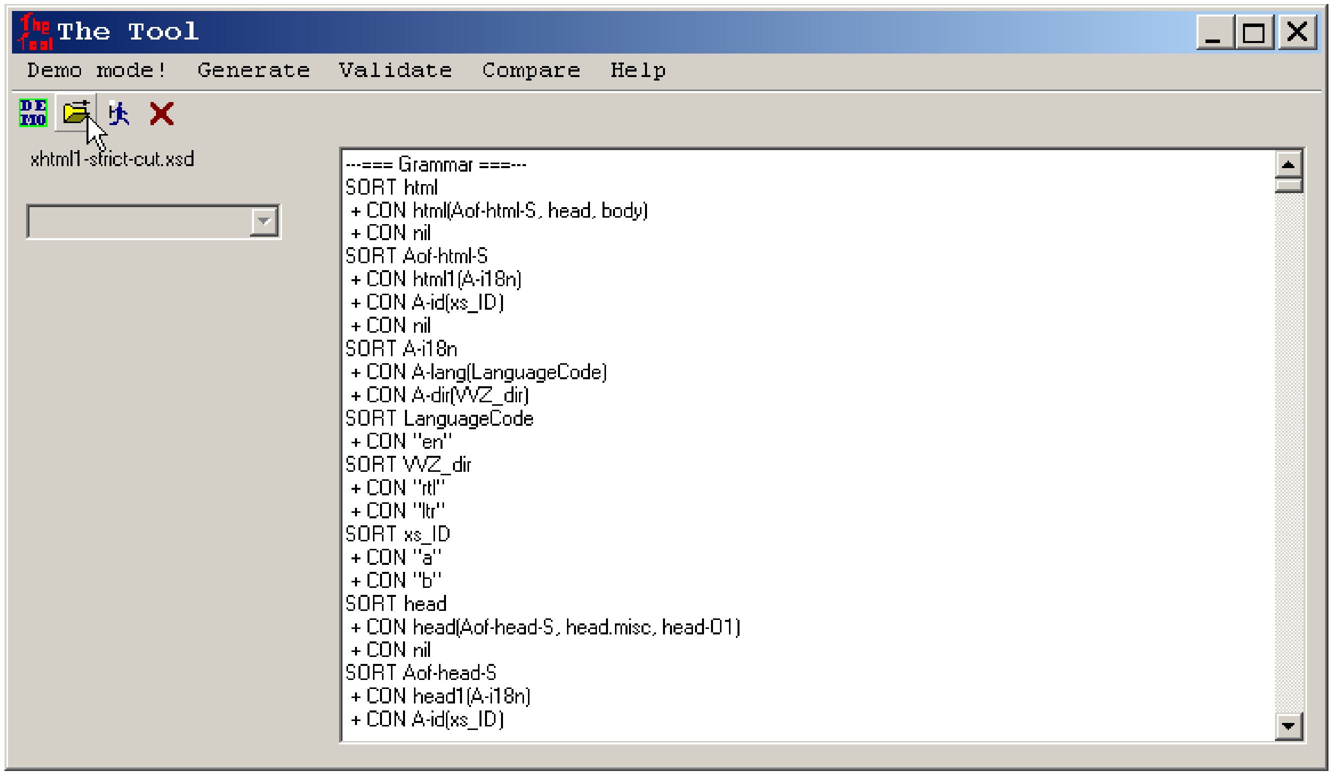
Grammarware Application:

Testing XML Validators

Vadim Zaytsev

26 November 2004

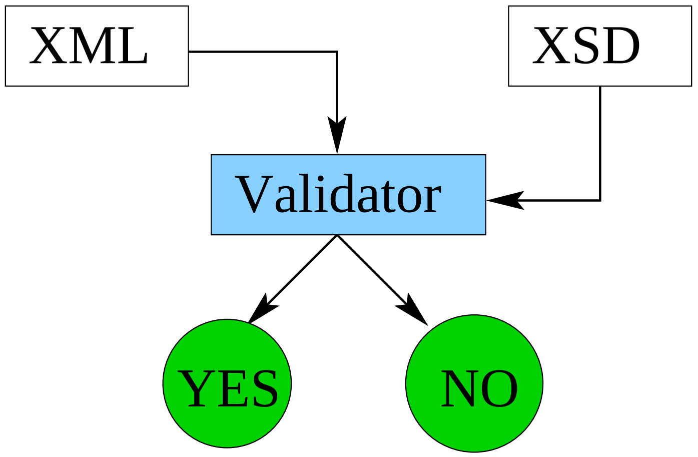
The story of one grammar-based tool



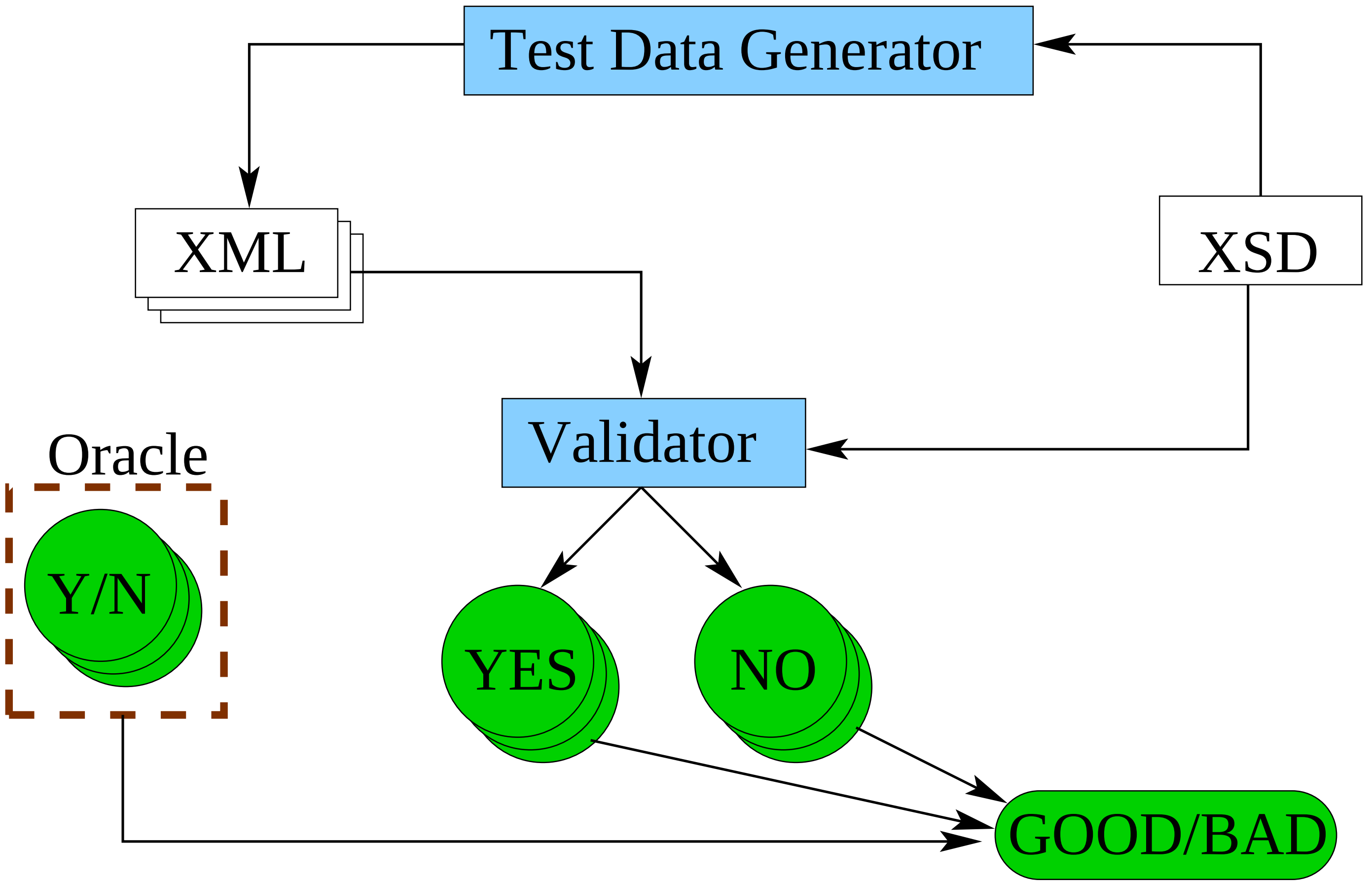
```
---=== Grammar ===---
SORT html
 + CON html(Aof-html-S, head, body)
 + CON nil
SORT Aof-html-S
 + CON html1(A-i18n)
 + CON A-id(xs_ID)
 + CON nil
SORT A-i18n
 + CON A-lang(LanguageCode)
 + CON A-dir(WZ_dir)
SORT LanguageCode
 + CON "en"
SORT WZ_dir
 + CON "rtl"
 + CON "ltr"
SORT xs_ID
 + CON "a"
 + CON "b"
SORT head
 + CON head(Aof-head-S, head.misc, head-01)
 + CON nil
SORT Aof-head-S
 + CON head1(A-i18n)
 + CON A-id(xs_ID)
```

*Grammar*ware and XML

- As it was told, grammarware is more than just compilers!
- eXtensible Markup Language — has a grammar (XML Schema)
- XML validator is a grammar-based tool:



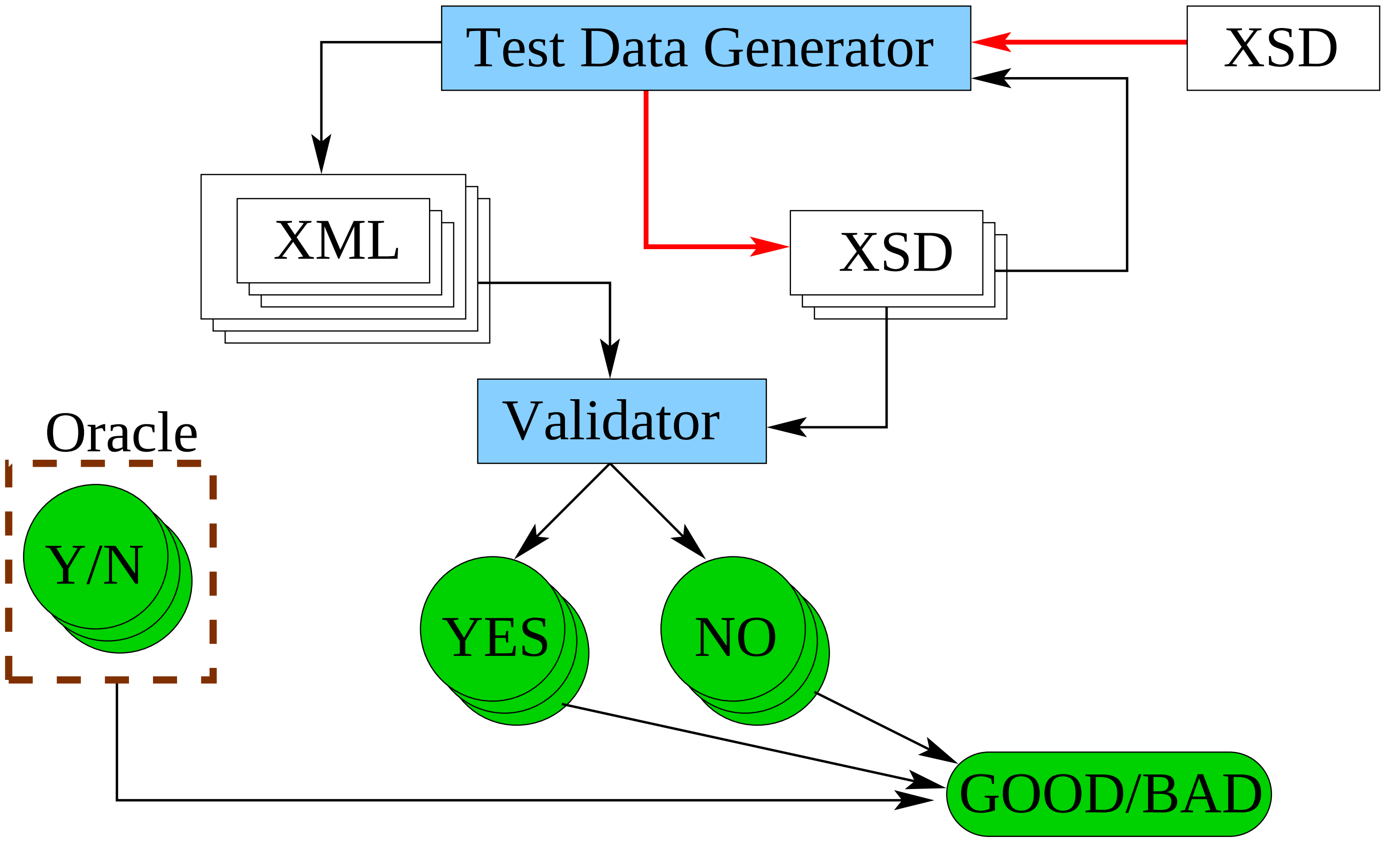
*Grammar*ware and XML



XML Schema is also a language

- And as such, it has a grammar
- Generate concrete grammars from the grammars' grammar
- Official name: XML Schema Schema for XML Schemas

XML Schema is also a language

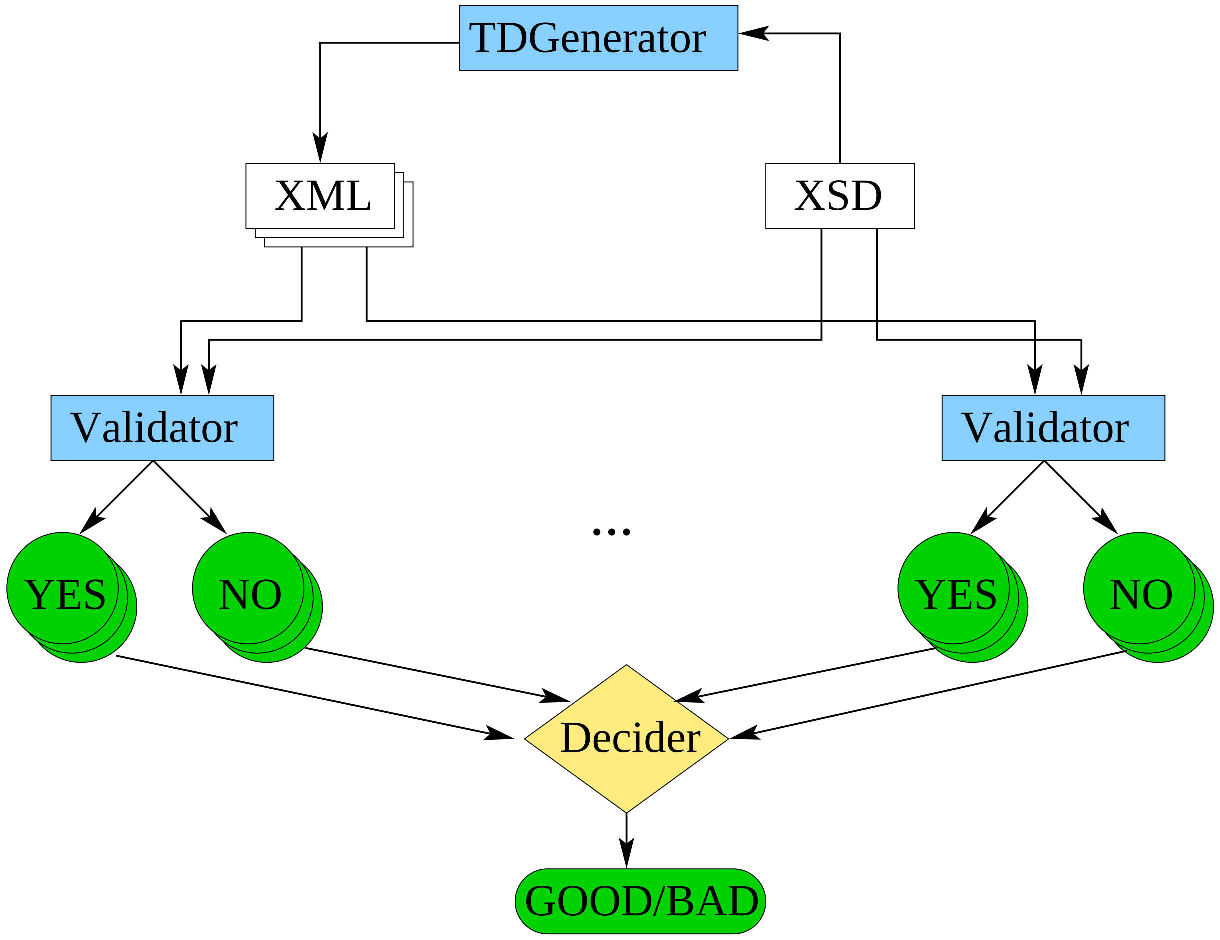


Differential testing

- Why Oracle?

- Having several XML validators,
  we can set them up to play against one another:
  - A file is fed to all of them
  - Diagnoses are gathered
  - If all agreed, cool
  - Different outputs reveal bugs

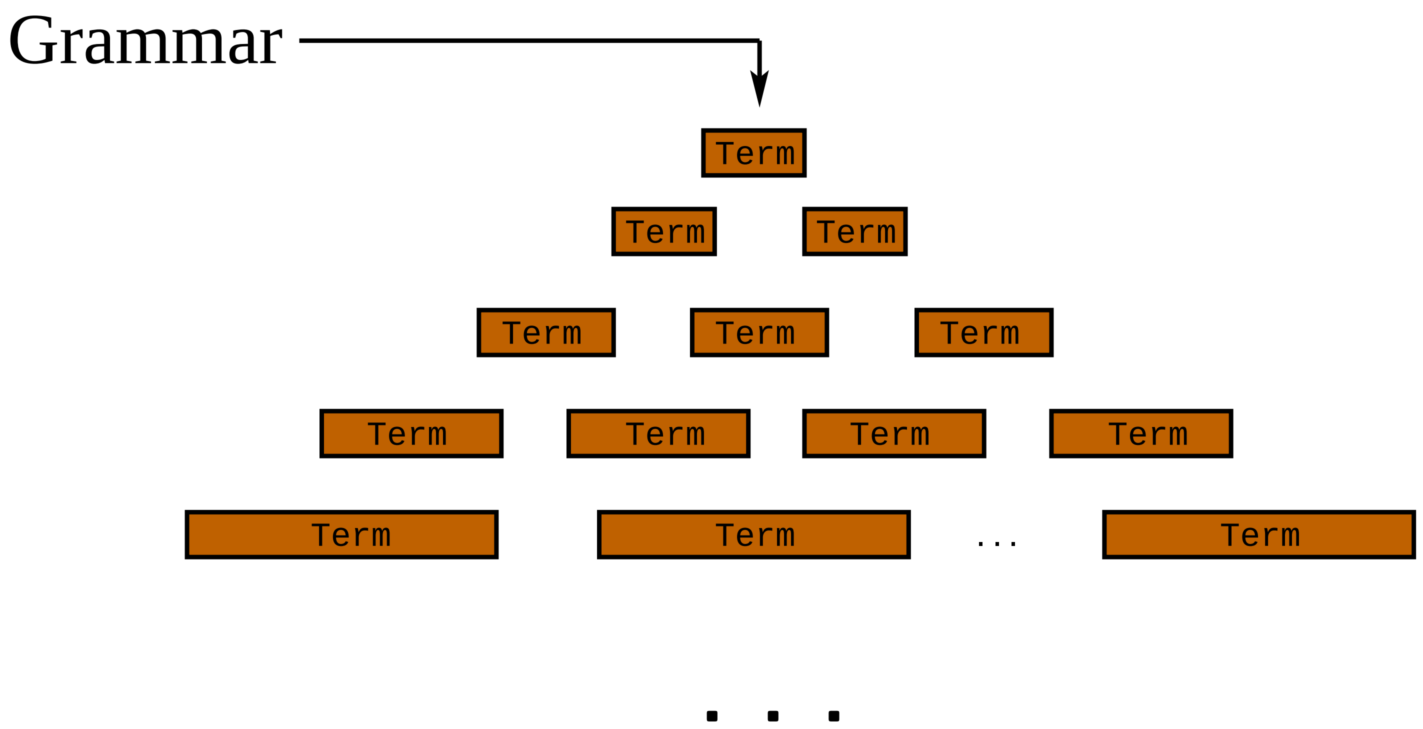
Differential testing



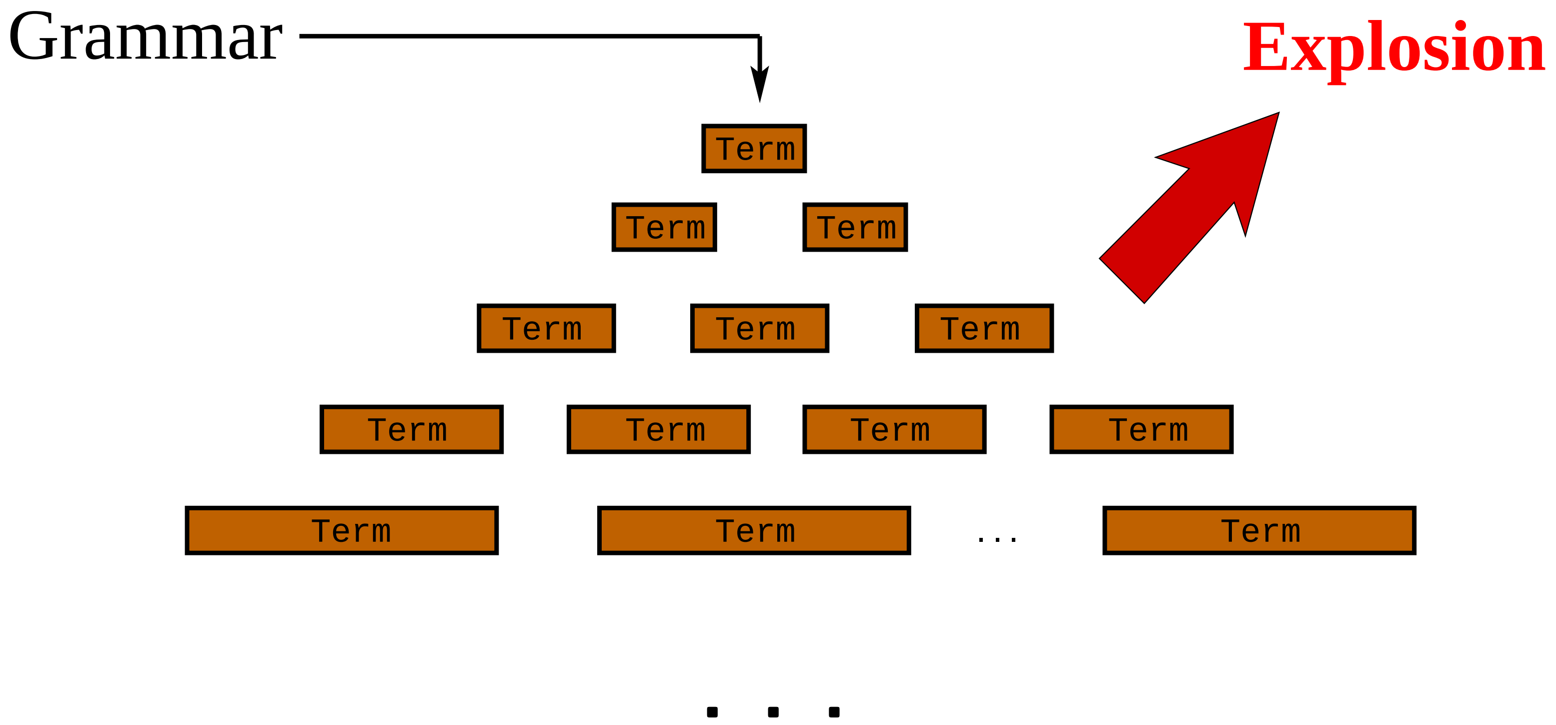
Combinatorial testing

- How to choose what to test?
- Let the grammar decide! Produce everything possible!
- Complementary to stochastic testing
- Characteristics:
  - No randomisation; no heuristics
  - Detailed control mechanisms
  - Formally defined coverage
  - Focus on huge test-data sets
  - Addresses grammar-based software

Combinatorial testing



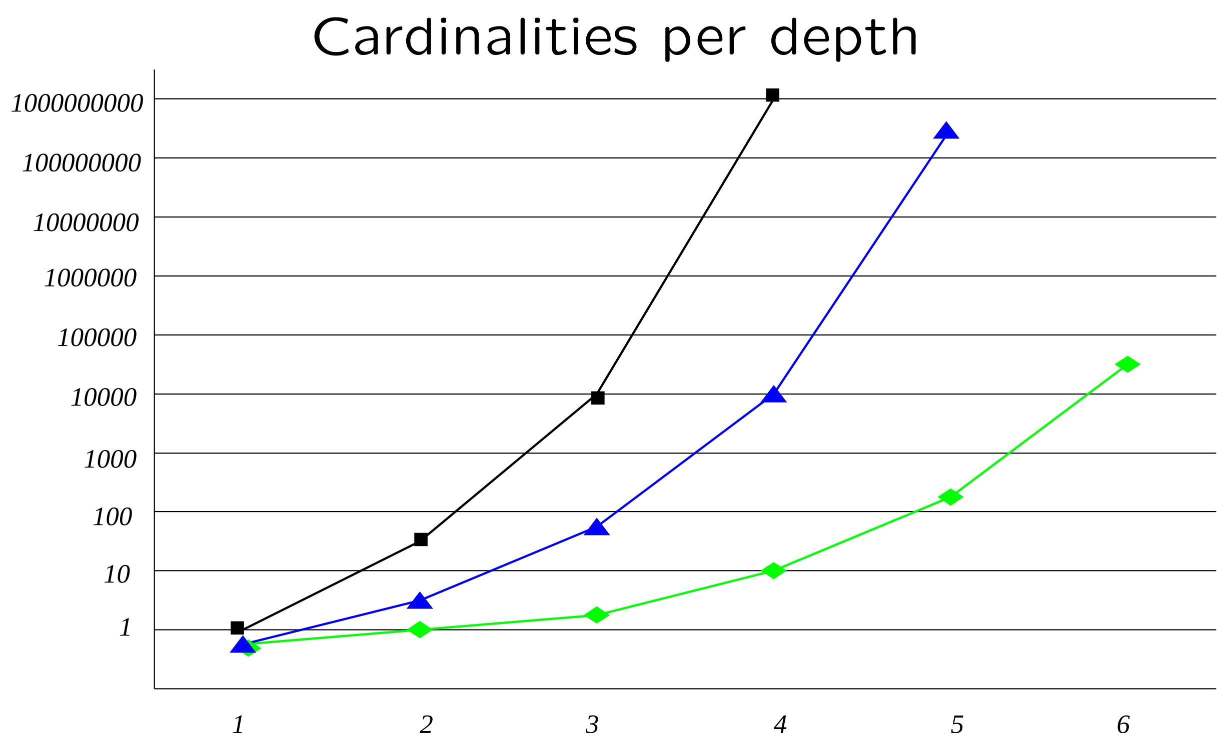
Combinatorial testing



Explosion

- Why not feasible?
  - Number of terms grows fast with depth
  - Grammars are complex
- *Explosion* means exponential behaviour
- Number of terms gets unfeasible within a very small number of depth layers explored

Explosion

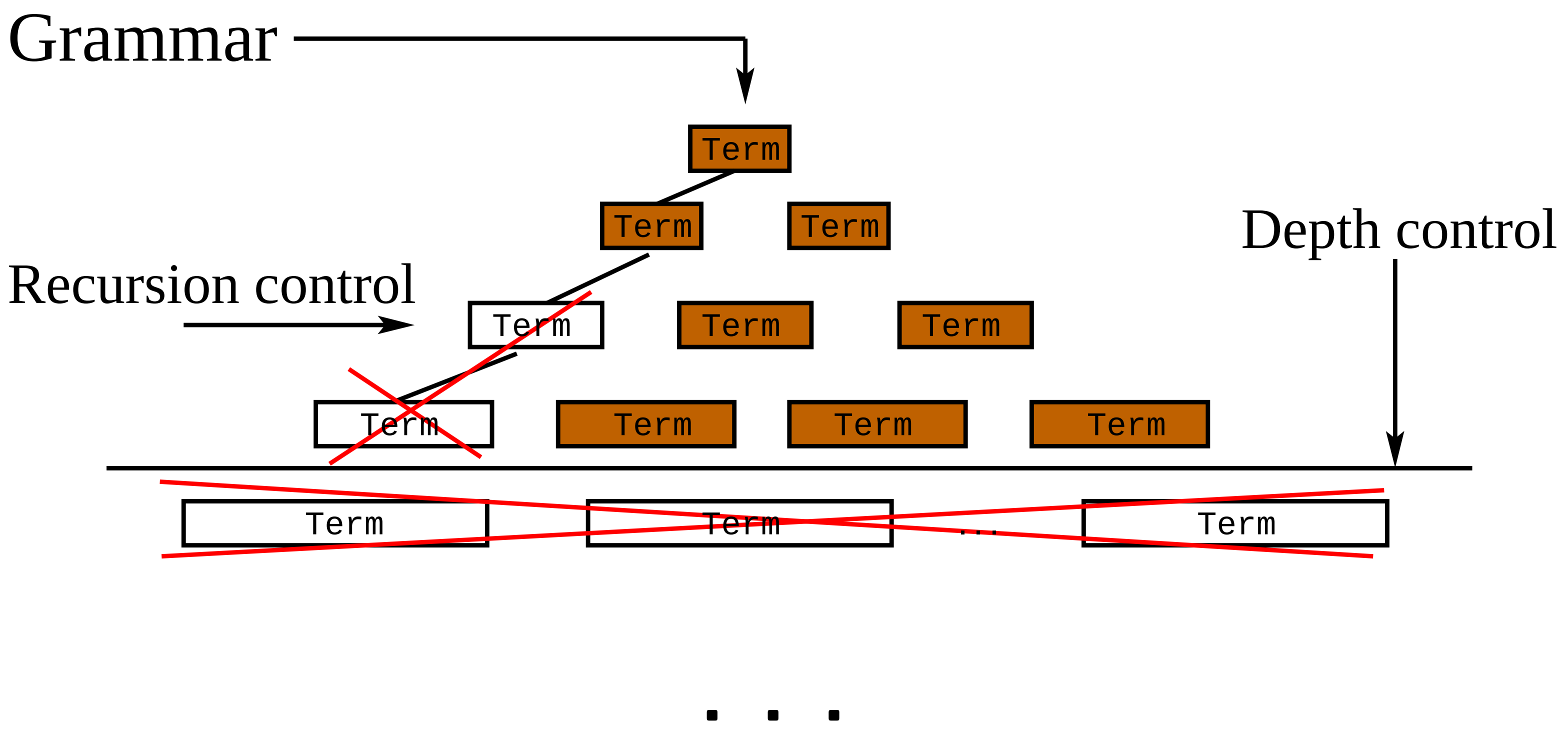


Number of generated terms grows fast with depth and eventually explodes (becomes greater than 18446744073709551616).

Solution? *Controlled* explosion

- Explosion is going to happen.
- We can try to postpone (to control) it.
- Now a tester's intuition comes into play.
- (in a strictly formalised way, though)

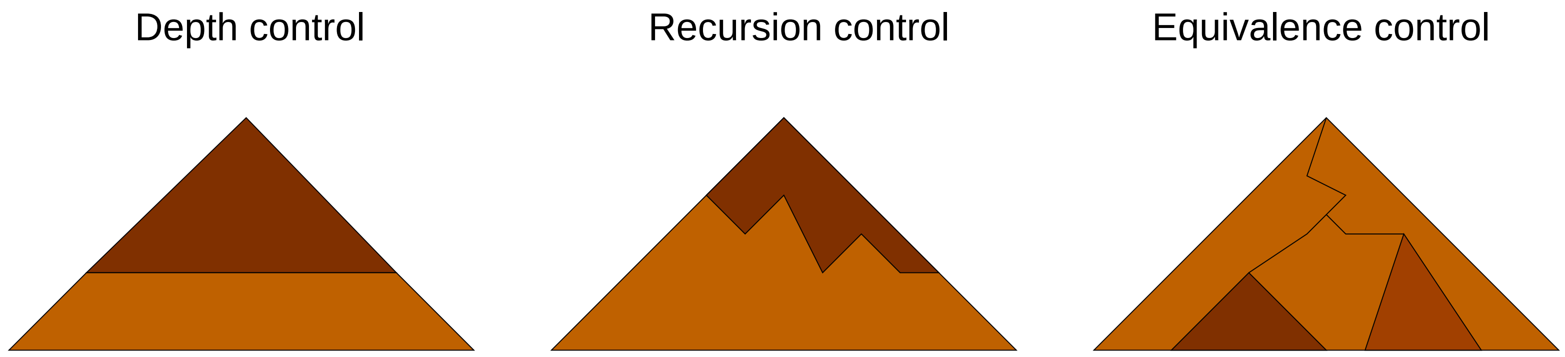
Controlled explosion



+ other mechanisms

Control mechanisms*

- **Depth control** — “length” of terms
- **Recursion control** — nested constructor applications
- **Equivalence control** — build equivalence classes
- **Balance control** — limit preceding levels
- **Combination control** — limited arguments use
- **Context control** — enforce context conditions



*R. Lämmel, W. Schulte. *Controlled Explosion in Grammar-based Testing.* Microsoft Research Redmond, internal document, 20 pages, October 2003.

Depth control

Taken from XHTML Strict 1.0 XML Schema:

```
<xs:group name="head.misc">
 <xs:sequence>
  <xs:choice minOccurs="0" maxOccurs="unbounded">
   <xs:element ref="script"/>
   <xs:element ref="style"/>
   <xs:element ref="meta"/>
   <xs:element ref="link"/>
   <xs:element ref="object"/>
  </xs:choice>
 </xs:sequence>
</xs:group>
```

Nobody is interested in infinite `<head>` tag.

Recursion control

Adopted from XHTML Strict 1.0 XML Schema:

```
<xs:element name="span">
 <xs:complexType mixed="true">
  <xs:complexContent mixed="true">
   <xs:extension base="Inline">
    <xs:attributeGroup ref="attrs"/>
   </xs:extension>
 </xs:complexContent></xs:complexType>
</xs:element>
...
<xs:complexType name="Inline" mixed="true">
 <xs:choice minOccurs="0" maxOccurs="unbounded">
  <xs:element ref="span"/>
  ...
 </xs:choice>
</xs:complexType>
```

We prefer to go deeper without a burden of nested `<span>`s.

Combination control

Taken from XHTML Strict 1.0 XML Schema:

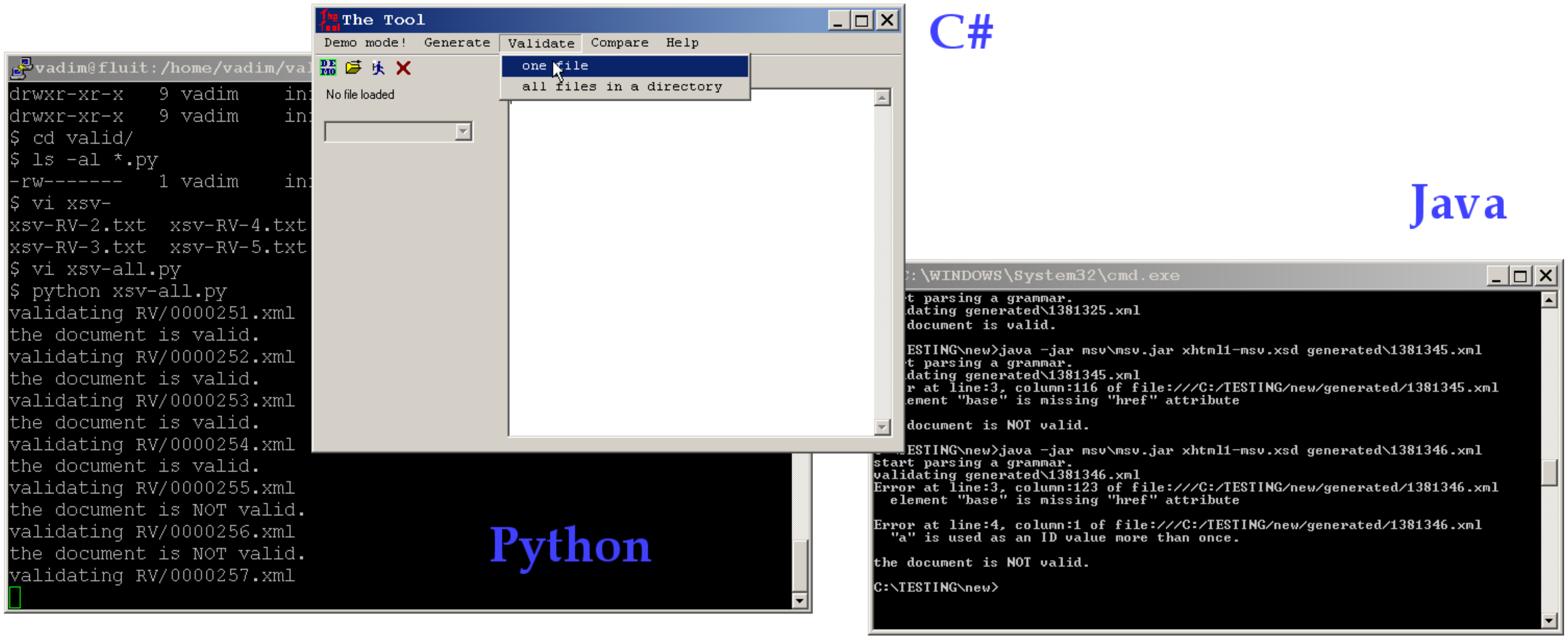
```
<xs:attributeGroup name="events">
 <xs:attribute name="onclick" type="Script"/>
 <xs:attribute name="ondblclick" type="Script"/>
 <xs:attribute name="onmousedown" type="Script"/>
 <xs:attribute name="onmouseup" type="Script"/>
 <xs:attribute name="onmouseover" type="Script"/>
 <xs:attribute name="onmousemove" type="Script"/>
 <xs:attribute name="onmouseout" type="Script"/>
 <xs:attribute name="onkeypress" type="Script"/>
 <xs:attribute name="onkeydown" type="Script"/>
 <xs:attribute name="onkeyup" type="Script"/>
</xs:attributeGroup>
```

XML attributes are numerous, but often independent.

Some XML validators

- .NET API — C#-based validator
  - simple wrapper had to be written
- JAXB — Sun Multi-Schema XML Validator 1.2
  - `http://developers.sun.com/dev/coolstuff/schema/`
  - Java-based, free of charge
- Python — XSV
  - `http://www.w3.org/2001/03/webdata/xsv`
  - free of charge, used by the W3C
  - simple wrapper had to be written

Some XML validators



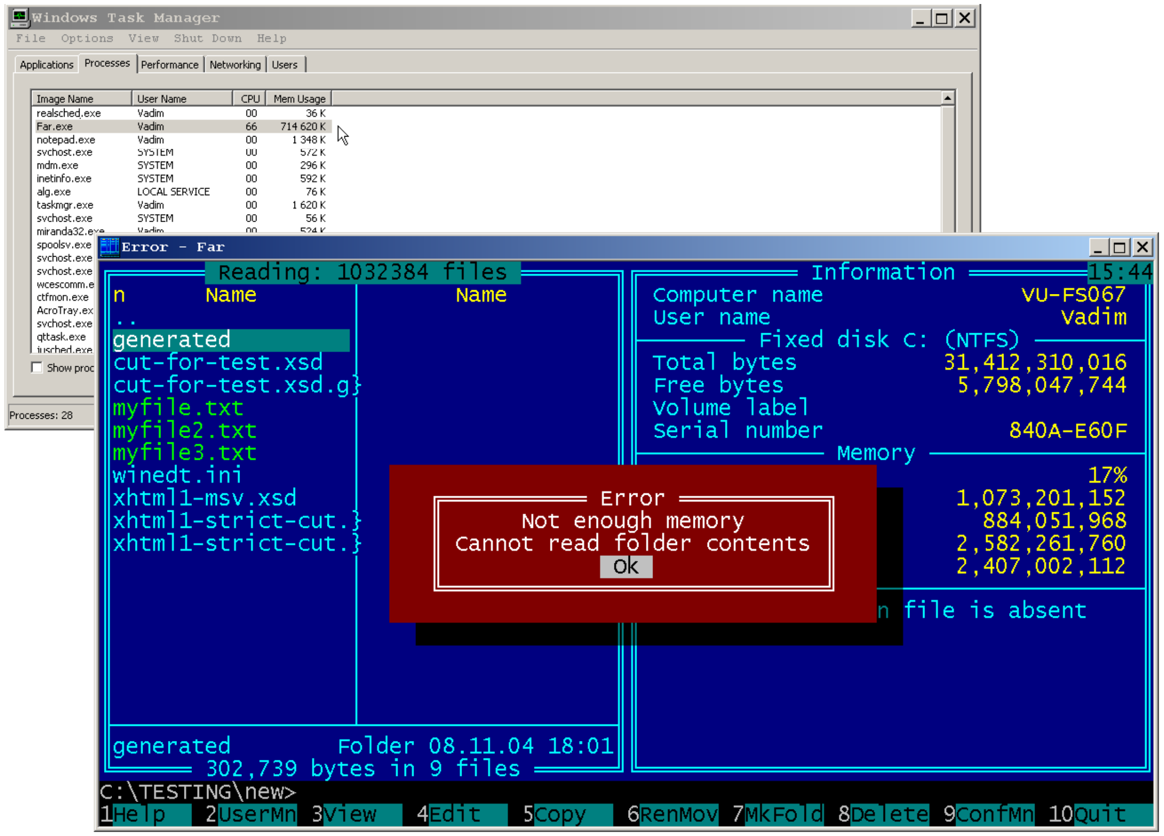
```
drwxr-xr-x    9 vadim    in
drwxr-xr-x    9 vadim    in
$ cd valid/
$ ls -al *.py
-rw-------    1 vadim    in
$ vi xsv-
xsv-RV-2.txt  xsv-RV-4.txt
xsv-RV-3.txt  xsv-RV-5.txt
$ vi xsv-all.py
$ python xsv-all.py
validating RV/0000251.xml
the document is valid.
validating RV/0000252.xml
the document is valid.
validating RV/0000253.xml
the document is valid.
validating RV/0000254.xml
the document is valid.
validating RV/0000255.xml
the document is NOT valid.
validating RV/0000256.xml
the document is NOT valid.
validating RV/0000257.xml
```

```
t parsing a grammar.
dating generated\1381325.xml
 document is valid.

ESTING\new>java -jar msv\msv.jar xhtml1-msv.xsd generated\1381345.xml
t parsing a grammar.
dating generated\1381345.xml
r at line:3, column:116 of file:///C:/TESTING/new/generated/1381345.xml
ement "base" is missing "href" attribute

 document is NOT valid.

ESTING\new>java -jar msv\msv.jar xhtml1-msv.xsd generated\1381346.xml
start parsing a grammar.
validating generated\1381346.xml
Error at line:3, column:123 of file:///C:/TESTING/new/generated/1381346.xml
  element "base" is missing "href" attribute

Error at line:4, column:1 of file:///C:/TESTING/new/generated/1381346.xml
  "a" is used as an ID value more than once.

the document is NOT valid.

C:\TESTING\new>
```

Scalability issues

- Opening the directory
  - Windows Explorer does not work
  - light-weight file managers give up at 1M
- Copying files
  - takes hours to complete
- **FOR** in Windows (.bat file syntax)
  - does not work with more than 15k files
  - silently skips $\approx$0.03% of the files
- **“*”** in Linux
  - core dumped
- Editing files
  - XML Spy gives in on too complicated files
  - Visual Studio .NET 2003 *works*!

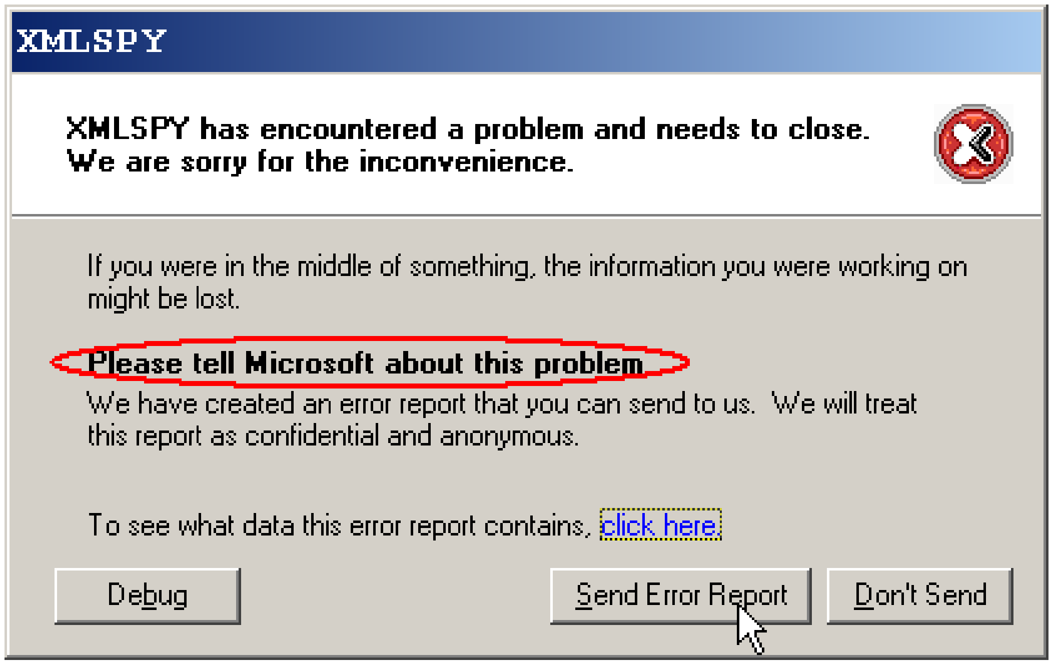
Scalability issue



```
Windows Task Manager
File  Options  View  Shut Down  Help
Applications | Processes | Performance | Networking | Users

Image Name        User Name        CPU   Mem Usage
realsched.exe     Vadim            00        36 K
Far.exe           Vadim            66   714 620 K
notepad.exe       Vadim            00     1 348 K
svchost.exe       SYSTEM           00       572 K
mdm.exe           SYSTEM           00       296 K
inetinfo.exe      SYSTEM           00       592 K
alg.exe           LOCAL SERVICE    00        76 K
taskmgr.exe       Vadim            00     1 620 K
svchost.exe       SYSTEM           00        56 K
miranda32.exe     Vadim            00       524 K
spoolsv.exe
svchost.exe
svchost.exe
wcescomm.e
ctfmon.exe
AcroTray.ex
svchost.exe
qttask.exe
jusched.exe

Show proc
Processes: 28
```

```
Error - Far
          Reading: 1032384 files                    Information                15:44
n         Name               Name            Computer name                 VU-FS067
..                                           User name                        Vadim
generated                                       Fixed disk C: (NTFS)
cut-for-test.xsd                             Total bytes             31,412,310,016
cut-for-test.xsd.g}                          Free bytes               5,798,047,744
myfile.txt                                   Volume label
myfile2.txt                                  Serial number                840A-E60F
myfile3.txt                                              Memory
winedt.ini                                                                      17%
xhtml1-msv.xsd                  Error                                 1,073,201,152
xhtml1-strict-cut.}        Not enough memory                            884,051,968
xhtml1-strict-cut.}    Cannot read folder contents                    2,582,261,760
                                  Ok                                  2,407,002,112
                                                                n file is absent

generated         Folder 08.11.04 18:01
      302,739 bytes in 9 files
C:\TESTING\new>
1Help  2UserMn 3View  4Edit  5Copy  6RenMov 7MkFold 8Delete 9ConfMn 10Quit
```

Scalability issue



XMLSPY

XMLSPY has encountered a problem and needs to close. We are sorry for the inconvenience.

If you were in the middle of something, the information you were working on might be lost.

Please tell Microsoft about this problem

We have created an error report that you can send to us. We will treat this report as confidential and anonymous.

To see what data this error report contains, click here.

What to test in the XML?

- Levels of XML file conformance
- Levels of XML processor conformance
- Grammar features: attributes, references, . . .
- Advanced features: namespaces, schema-related markup, . . .
- Secondary features: header, scalability, . . .

Before validity comes...

- Well-formedness
  - the document as a whole matches the production `document`
  - all tags closed in place
- Proper header:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE html
          PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
          "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
      xml:lang="en" lang="en">
</html>
```

Attributes and "simple" types

Taken from XHTML Strict 1.0 XML Schema:

```
<xs:simpleType name="Length">
 <xs:restriction base="xs:string">
  <xs:pattern value="[-+]?(\d+|\d+(\.\d+)?%)"/>
</xs:restriction></xs:simpleType>
<xs:simpleType name="MultiLength">
 <xs:restriction base="xs:string">
  <xs:pattern value="[-+]?(\d+|\d+(\.\d+)?%)|[1-9]?(\d+)?\*"/>
</xs:restriction></xs:simpleType>
<xs:element name="img">
 <xs:complexType>
  <xs:attribute name="height" type="Length"/>
  <xs:attribute name="width" type="Length"/>
  ...
</xs:complexType></xs:element>
```

One of the problems found: **duplicate attributes!**

Document-wide unique identifiers

Taken from XHTML Strict 1.0 XML Schema:

```
<xs:element name="html">
 <xs:complexType>
  ...
  <xs:attribute name="id" type="xs:ID"/>
 </xs:complexType>
</xs:element>
...
<xs:element name="td">
 <xs:complexType mixed="true">
  <xs:complexContent mixed="true">
   <xs:extension base="Flow">
    <xs:attribute name="headers" type="xs:IDREFS"/>
    ...
   </xs:extension>
  </xs:complexContent>
 </xs:complexType>
</xs:element>
```

Namespaces

Taken from Namespaces in XML:

```
<?xml version="1.0"?>
<!-- initially, the default namespace is "books" -->
<book xmlns='urn:loc.gov:books'
      xmlns:isbn='urn:ISBN:0-395-36341-6'>
 <title>Cheaper by the Dozen</title>
 <isbn:number>1568491379</isbn:number>
 <notes>
  <!-- make HTML the default namespace for some commentary -->
  <p xmlns='urn:w3-org-ns:HTML'>
   This is a <i>funny</i> book!
  </p>
 </notes>
</book>
```

Different document parts may belong to different namespaces and conform to different XML Schemas.

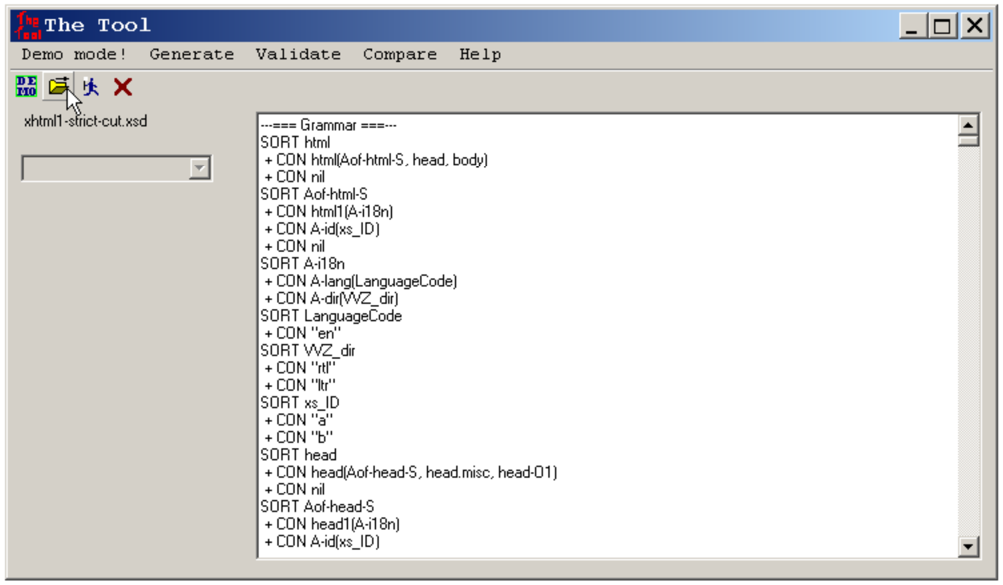
Validator's tolerance

- *Lax* validation in the XSV
  - activated automatically with an empty schema
- Unknown element
  - .NET warning
- Validator's robustness
  - XSV crashes with a duplicate attribute
  - stress testing (stress nesting)

How does it work

- XSD file is parsed
- additional grammar file is parsed
- their contents form a grammar
- terms are generated in memory
- terms are serialised as XML files to the hard disk

How does it work

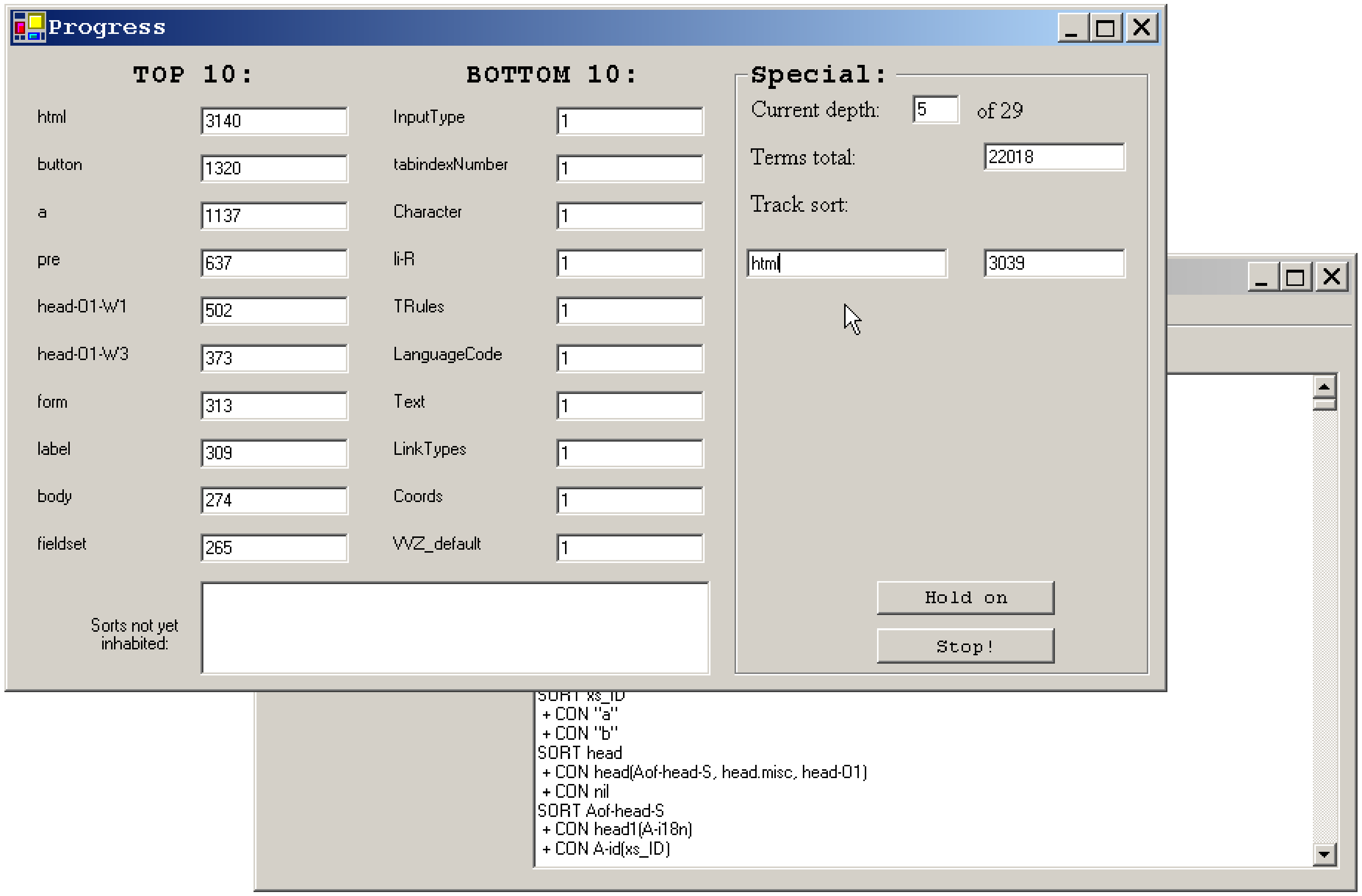


```
---=== Grammar ===---
SORT html
 + CON html(Aof-html-S, head, body)
 + CON nil
SORT Aof-html-S
 + CON html1(A-i18n)
 + CON A-id(xs_ID)
 + CON nil
SORT A-i18n
 + CON A-lang(LanguageCode)
 + CON A-dir(VVZ_dir)
SORT LanguageCode
 + CON "en"
SORT VVZ_dir
 + CON "rtl"
 + CON "ltr"
SORT xs_ID
 + CON "a"
 + CON "b"
SORT head
 + CON head(Aof-head-S, head.misc, head-01)
 + CON nil
SORT Aof-head-S
 + CON head1(A-i18n)
 + CON A-id(xs_ID)
```

Visualisation

- after parsing is over the complete grammar is dumped
- during generation we can see number of terms per sort
- generation process can be paused
- we can stop at any depth

Visualisation



Progress

TOP 10:

html	3140
button	1320
a	1137
pre	637
head-01-W1	502
head-01-W3	373
form	313
label	309
body	274
fieldset	265

BOTTOM 10:

InputType	1
tabindexNumber	1
Character	1
li-R	1
TRules	1
LanguageCode	1
Text	1
LinkTypes	1
Coords	1
WZ_default	1

Sorts not yet inhabited:

Special:

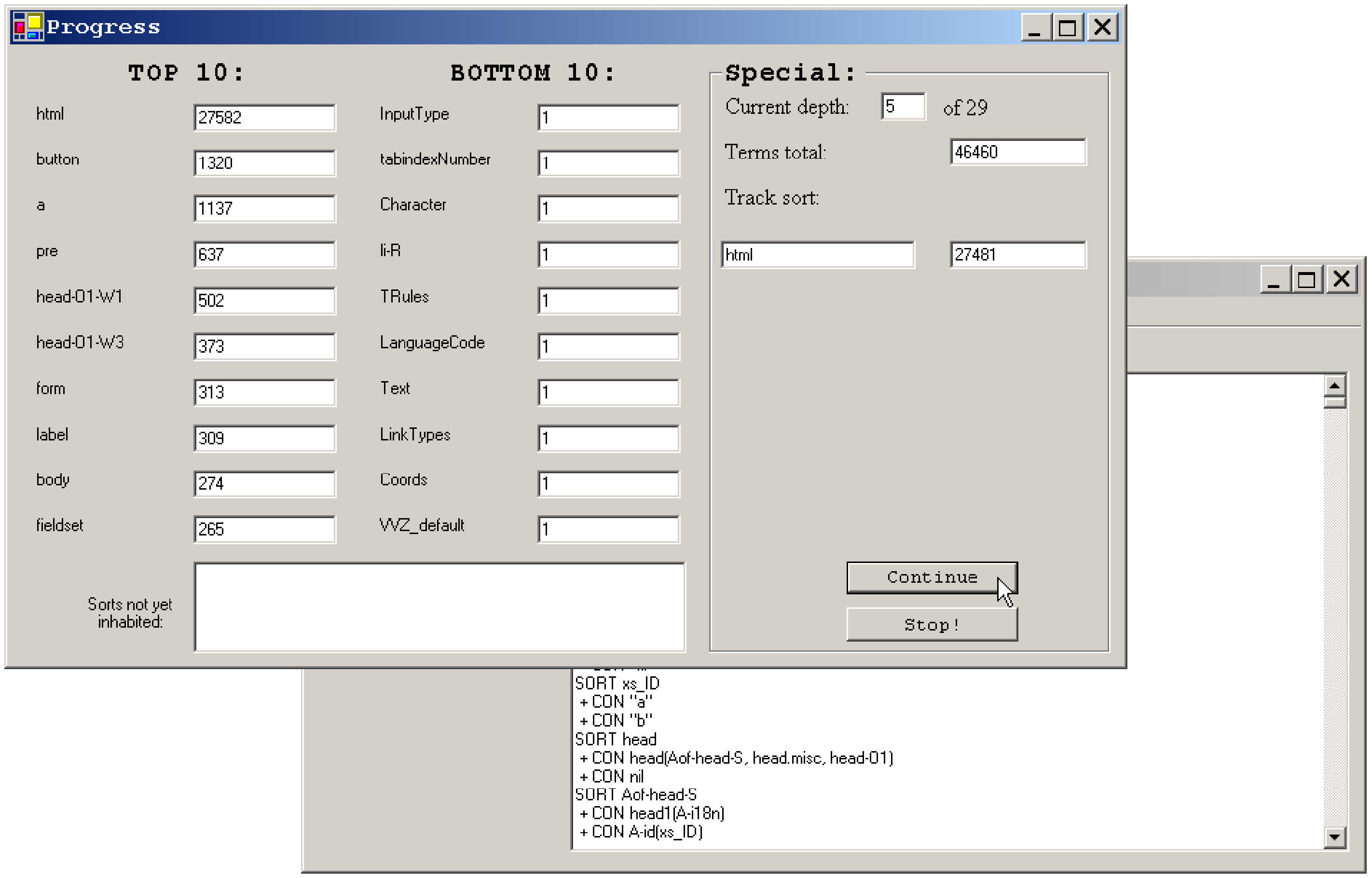
Current depth: 5 of 29

Terms total: 22018

Track sort: html 3039

```
SORT xs_ID
 + CON "a"
 + CON "b"
SORT head
 + CON head(Aof-head-S, head.misc, head-01)
 + CON nil
SORT Aof-head-S
 + CON head1(A-i18n)
 + CON A-id(xs_ID)
```

Visualisation



TOP 10:		BOTTOM 10:	
html	27582	InputType	1
button	1320	tabindexNumber	1
a	1137	Character	1
pre	637	li-R	1
head-01-W1	502	TRules	1
head-01-W3	373	LanguageCode	1
form	313	Text	1
label	309	LinkTypes	1
body	274	Coords	1
fieldset	265	WZ_default	1

Sorts not yet inhabited:

Special:

Current depth: 5 of 29

Terms total: 46460

Track sort:

html 27481

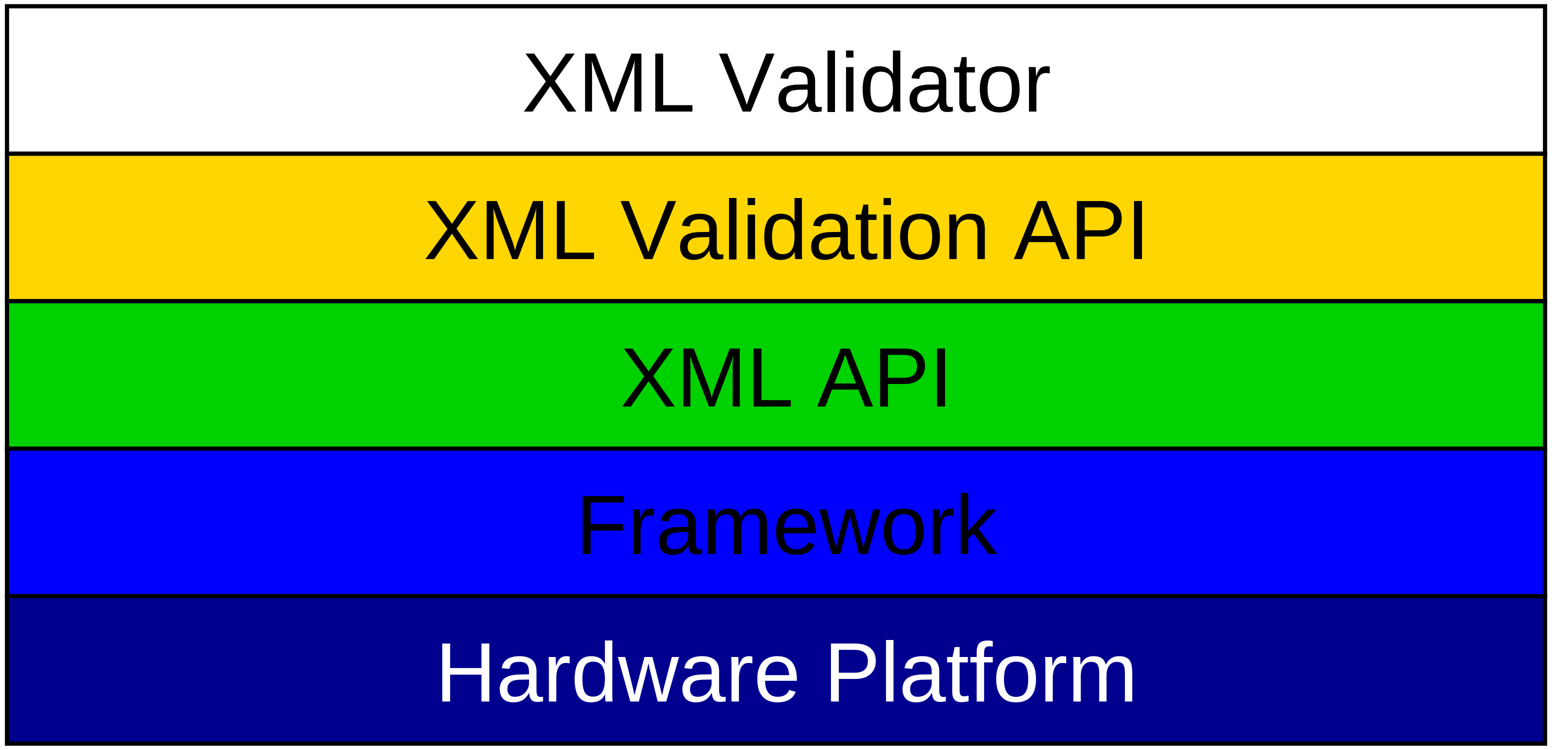
```
SORT xs_ID
 + CON "a"
 + CON "b"
SORT head
 + CON head(Aof-head-S, head.misc, head-01)
 + CON nil
SORT Aof-head-S
 + CON head1(A-i18n)
 + CON A-id(xs_ID)
```

Conclusion

- XML validator tests an XML file to conform to a grammar
- XML Schema is not an easy spec to implement (to test)
- Our tool tests if an XML validator works well
- *Automated* generation of huge test-data sets
- *Differential* testing for race of validators
- `http://www.cs.vu.nl/grammarware`

Questions?

The hierarchy of XML files processing


XML Validator
XML Validation API
XML API
Framework
Hardware Platform