

# DICOM SR Tools

## - dicom3tools ...



*David A. Clunie*

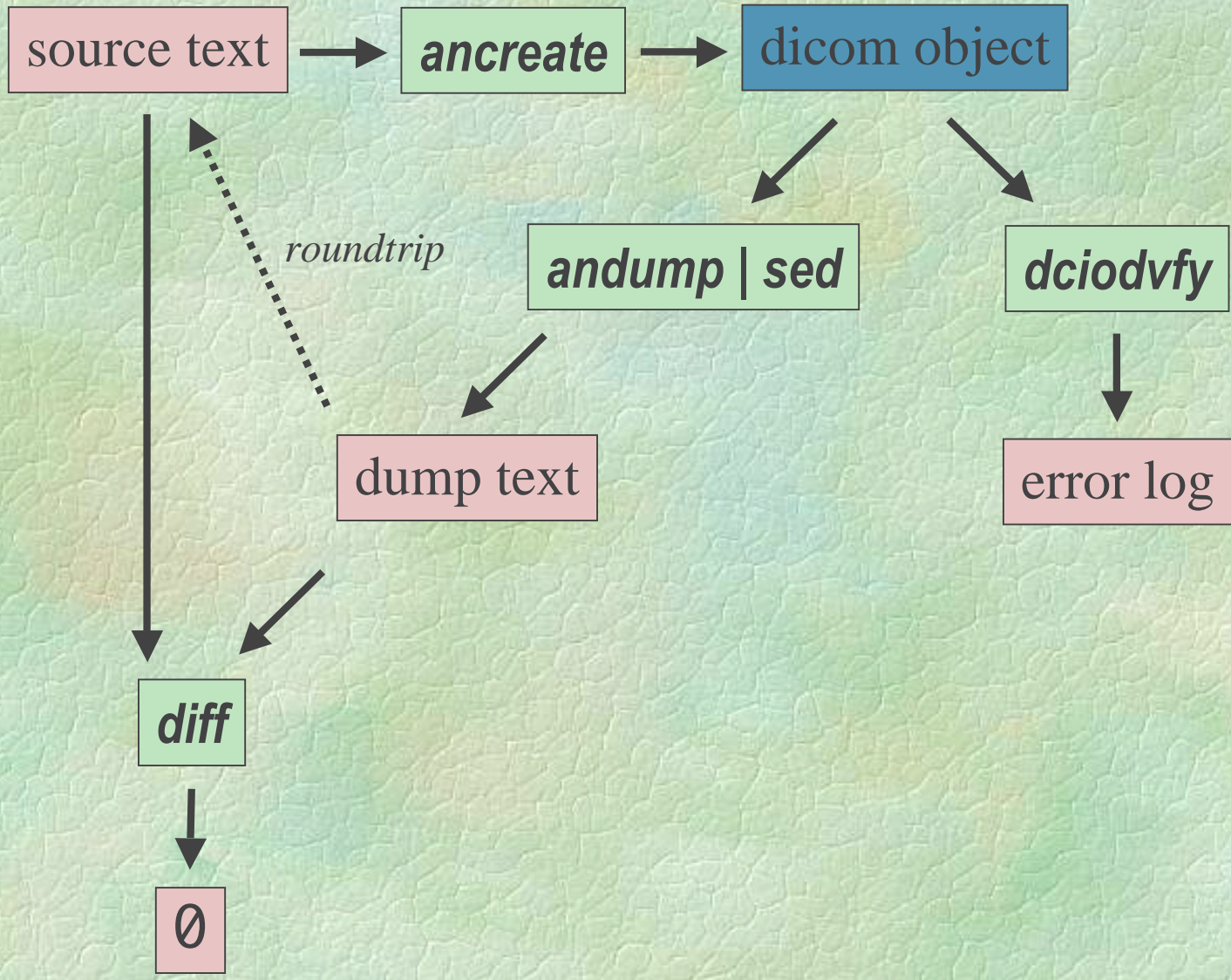
NEMA SR Workshop  
29th-30th March 2000

# What are they ?

- ❧ Unix utilities for DICOM files
  - Create, copy, manipulate, verify ...
- ❧ C++ source for Solaris, Irix, Linux ...
- ❧ For testing and experimentation  
***NOT*** clinical or commercial use
- ❧ Not open source license per se
- ❧ Not supported except in an ad hoc manner

# What SR Features ?

- ☞ Data dictionary and IOD support for SR FT
- ☞ Create DICOM file from dump-like format
  - `ancreate`
- ☞ Verify DICOM SR IOD from file
  - `dciodvfy [-describe] [-verbose]`
- ☞ Round-trip engineering
  - `ancreate | andump | sed | ancreate ...`



```

% head -20 examplef9.source
(0x0008,0x0016) SOP Class UID      VR=<UI> VL=<0x001e> <1.2.840.10008.5.1.4.1.1.88.33>
(0x0008,0x0018) SOP Instance UID   VR=<UI> VL=<0x0012> <1.2.3.4.5.6.7.300>
(0x0008,0x0020) Study Date         VR=<DA> VL=<0x0008> <19991029>
(0x0008,0x0023) Content (formerly Image) Date VR=<DA> VL=<0x0008> <19991029>
(0x0008,0x0030) Study Time         VR=<TM> VL=<0x0006> <154500>
(0x0008,0x0033) Content (formerly Image) Time VR=<TM> VL=<0x0006> <154510>
(0x0008,0x0050) Accession Number    VR=<SH> VL=<0x0006> <123456>
(0x0008,0x0060) Modality            VR=<CS> VL=<0x0002> <SR>
(0x0008,0x0070) Manufacturer       VR=<LO> VL=<0x0004> <WG6 >
(0x0008,0x0090) Referring Physician's Name VR=<PN> VL=<0x0014> <Luke^Will^^Dr.^M.D. >
(0x0008,0x1111) Referenced Study Component Sequence VR=<SQ> VL=<0xffffffff> []
%endseq
(0x0010,0x0010) Patient's Name     VR=<PN> VL=<0x000e> <Homer^Jane^^^ >
(0x0010,0x0020) Patient's ID       VR=<LO> VL=<0x0006> <234567>
(0x0010,0x0030) Patient's Birth Date VR=<DA> VL=<0x0008> <19991109>
(0x0010,0x0040) Patient's Sex      VR=<CS> VL=<0x0002> <F >
(0x0020,0x000d) Study Instance UID  VR=<UI> VL=<0x0012> <1.2.3.4.5.6.7.100>
(0x0020,0x000e) Series Instance UID VR=<UI> VL=<0x0012> <1.2.3.4.5.6.7.200>
(0x0020,0x0010) Study ID           VR=<SH> VL=<0x0006> <345678>
(0x0020,0x0011) Series Number      VR=<IS> VL=<0x0002> <1>
%
```

```
% cat Makefile
```

```
examplef9.dcm: examplef9.source
```

```
  ancreate < examplef9.source > examplef9.dcm.tmp
```

```
  dccp examplef9.dcm.tmp examplef9.dcm -nodisclaimer -n
```

```
  rm examplef9.dcm.tmp
```

```
examplef9.dcm.roundtrip: examplef9.dcm examplef9.source
```

```
  andump examplef9.dcm 2>&1 | sed \
```

```
  -e 's/.*0xffffe,0xe00d.*Item Delimitation Item.*/%enditem/' \
```

```
  -e 's/.*0xffffe,0xe0dd.*Sequence Delimitation Item.*/%endseq/' \
```

```
  -e 's/.*0xffffe,0xe000.*Item.*/%item/' -e 's/[]] [A-Z][A-Z] /) /' \
```

```
  > examplef9.dcm.roundtrip
```

# What to expect in future ...

⇒ dciodvfy

- more robust validation
- validation against templates ?

# Challenges for an SR Validator

## ∞ Macros

- were added several years ago (code sequence)

## ∞ Recursive structures

- nested content sequence
- nested/overlapping macros

## ∞ IOD constraints on relationship/value type

## ∞ Conditions

- e.g. on Concept Name Code Sequence



# What to expect in future (2)...

↪ Creating SR by writing dump files is pretty tedious unless you are a Makefile or script !

↪ `mltodcsr` ?

↪ `dcstrtoml [-xml|-html]` ?

# Where to get them ?

<http://idt.net/~dclunie/dicom3tools/workinprogress/>