ISQL
Definition & Capabilities

Steve Shumate
Introduction

- INFORMIX ISQL provides easy ways to view and manipulate data
- ISQL features have been utilized at CBRFC for many years
ISQL Topics

- **Forms**
  - Computer generated
  - Annotated

- **Reports**
  - Useful but largely superceded by tcl-isql

- **tcl-isql**
  - Simple, robust and flexible data access
ISQL – ring menu

Navigate with cursor keys – or type first letter of menu item
Forms – auto_generated

- Two ways
  - Through ISQL
  - Tcl script formgen.tcl
- This will produce useful forms
**Example – Auto-Generated Form**

<table>
<thead>
<tr>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>lid [ ]</td>
</tr>
<tr>
<td>county [ ]</td>
</tr>
<tr>
<td>coe [ ]</td>
</tr>
<tr>
<td>cpm [ ]</td>
</tr>
<tr>
<td>detail [ ]</td>
</tr>
<tr>
<td>elev [ ]</td>
</tr>
<tr>
<td>hdatum [ ]</td>
</tr>
<tr>
<td>hsa [ ]</td>
</tr>
<tr>
<td>hu [ ]</td>
</tr>
<tr>
<td>lat [ ]</td>
</tr>
<tr>
<td>lon [ ]</td>
</tr>
<tr>
<td>lremark [ ]</td>
</tr>
<tr>
<td>lrevise [ ]</td>
</tr>
<tr>
<td>name [ ]</td>
</tr>
<tr>
<td>network [ ]</td>
</tr>
<tr>
<td>rb [ ]</td>
</tr>
<tr>
<td>rfc [ ]</td>
</tr>
</tbody>
</table>
database hd5_22str
screen size 24 by 80
{
    location
        lid [f1]                      
        county [f2]                   
        coe [f3]                      
        cpm [f4]                      
        detail [f5]                   
        elev [f6]                     
        hdatum [f7]                   
        hsa [f8]                      
        hu [f9]                       
        lat [f10]                     
        lon [f11]                     
        lremark [f12]                 
        lrevise [f13]                 
        name [f14]                    
        network [f15]                 
        rb [f16]                      
        rfc [f17]                     
        sbd [f18]                     
        sn [f19]                      
        state [f20]                   
        waro [f21]                    
        wfo [f22]                     
        wsfo [f23]                    
        type [f24]                    
        des [f25]                     
        det [f26]                     
        post [f27]                    
        stntype [f28]                 
        tzone [f29]                   
    }
end
tables
table=location
attributes
f1 = table.lid;
f2 = table.county;
### Example – Annotated Form

**QUERY:** ESCAPE queries. INTERRUPT discards query. ARROW keys move cursor. Searches the active database table.

```
| lid | pe1 | pe2 | vtime |
```

<table>
<thead>
<tr>
<th><strong>STAGE</strong></th>
<th><strong>FLOW</strong></th>
<th><strong>QUALITY CODE</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>lowscren</td>
<td>lowscrenF</td>
<td>lowscrenq</td>
</tr>
<tr>
<td>sigrate</td>
<td>sigrateF</td>
<td>sigrateq</td>
</tr>
<tr>
<td>screenrat</td>
<td>screenratF</td>
<td>screenrateq</td>
</tr>
<tr>
<td>sigratet</td>
<td></td>
<td></td>
</tr>
<tr>
<td>fis</td>
<td>fisF</td>
<td>fisq</td>
</tr>
<tr>
<td>action</td>
<td>actionF</td>
<td>actionq</td>
</tr>
<tr>
<td>alert</td>
<td>alertF</td>
<td>alertq</td>
</tr>
<tr>
<td>bank</td>
<td>bankF</td>
<td>bankq</td>
</tr>
<tr>
<td>flood</td>
<td>floodF</td>
<td>floodq</td>
</tr>
<tr>
<td>modflood</td>
<td>modfloodF</td>
<td>modfloodq</td>
</tr>
<tr>
<td>majflood</td>
<td>majfloodF</td>
<td>majfloodq</td>
</tr>
<tr>
<td>record</td>
<td>recordF</td>
<td>recordq</td>
</tr>
<tr>
<td>highscreen</td>
<td>highscreenF</td>
<td>highscreenq</td>
</tr>
<tr>
<td>damscreen</td>
<td>damscreenF</td>
<td>damscreenq</td>
</tr>
</tbody>
</table>

*Values below this level are rejected*
database fastetc
screen size 24 by 80
{
  lid [f000 ] pe1 [a] pe2 [b] vdtime [f001 ]

  STAGE FLOW QUALITY CODE
  
  lowscreen [f002 ] lowscreenf [f015 ] lowscreenq [c]
  sigrate [f003 ] sigratet [f004 ] sigrateq [d]
  screenrate [f004 ] screenrateq [e]

  siguratet [f028] screenratet [f029]

  fis [f005 ] fisf [f018 ] fisq [f]
  action [f006 ] actionf [f019 ] actionq [g]
  alert [f007 ] alertf [f020 ] alertq [h]
  bank [f008 ] bankf [f021 ] bankq [i]
  flood [f009 ] floodf [f022 ] floodq [j]
  modflood [f010 ] modfloodf [f023 ] modfloodq [k]
  majflood [f011 ] majfloodf [f024 ] majfloodq [l]
  record [f012 ] recordf [f025 ] recordq [m]
  highscreen [f013 ] highscreenf [f026 ] highscreenq [n]
  damscreen [f014 ] damscreenf [f027 ] damscreenq [o]
}
end
tables
rivercrit
attributes
f000 = rivercrit.lid, UPSHIFT;
a = rivercrit.pe1, UPSHIFT;
b = rivercrit.pe2, UPSHIFT;
f001 = rivercrit.vdtime, COMMENTS="date form CCYMMDD";
f002 = rivercrit.lowscreen, COMMENTS="values below this level are rejected";
f003 = rivercrit.sigrate, COMMENTS="# ft/sigratet; rates between this and screen rate are valid sig. rise";
f004 = rivercrit.screenrate, COMMENTS="# ft/screenratet; rates above this are rejected";
Forms - implementation

- Use DBPATH environment variable in script to start ISQL
- `DBPATH=/local/fast/etc/forms://db1://db2: //ONLINE://ONLINE_REP`
- Insert/update
- Query with relational operators (e.g. >0.5)
- Output rows
  - screen or unload style
  - Current row or all rows returned by query
Forms – implementation, cont.

- Not another database name change!
  - `upd_dbname.tcl` queries `apps_defaults` for database name replaces `dbname` in all form specifications and recompiles to make new forms with correct `dbname`
Reports

- Makes a nice printed report
  - Headers/trailers
  - Page breaks
- Relevant?
PROMONLY REPORT

WATER YEAR 1997

QUALITY CODE EXPLANATION:

V - verified -- has passed our better quality control algorithms
S - screened -- has passed our first qc algorithm or has been manually set good
Q - questionable -- has failed our qc algorithms but the human reviewer chose not to set it bad
D - estimated -- the data was missing or was set bad and has been estimated using a ratio of point averages for some selected neighbor stations
(we suggest that quality D is better E)
E - estimated -- the data was missing or was set bad and has been estimated using a spatial algorithm incorporating PRISM data

ALPINE
.E ALPA3 961031 Z DH00/PPM4ZZZ/DIE1/
.E1 : oct : 4.29V /: 2.38 181%
.E1 : nov : .78V /: 1.44 54%
.E1 : dec : .00V /: 1.35 %
.E1 : jan : 2.77V /: 1.35 205%
.E1 : feb : 1.71V /: 1.29 132%
.E1 : mar : .23Q /: 1.30 18%
.E1 : apr : .23V /: .63 36%
.E1 : may : .61V /: .75 81%
.E1 : jun : 1.54Q /: .86 179%
.E1 : jul : 2.93Q /: 3.43 85%
.E1 : aug : 3.30V /: 4.51 73%
.E1 : sep : 4.09V /: 2.41 170%
total ------- ------- ------
22.48 21.69 104%
database
define
  variable totobs float
  variable totavg float
  variable stype char(1)
  variable rstat char(2)
end

input
  prompt for rstat
    using "ENTER the 2 character state code (caps please): "
  prompt for stype
    using "ENTER the level of the data to be used for the report (2, 3 or 4): "
end

output
  report to "wymonly.out"
  page length 66 { ibm print command likes 60, hp print command likes 66 }
  top margin 0
  bottom margin 0
end

select

lp.oct, lp.nov, lp.dec, p.jan, p.feb, p.mar, p.apr, p.may, p.jun,
  p.jul, p.aug, p.sep,

```
print column 20, "PROMONLY REPORT"
  skip 1 line
  print column 20, "WATER YEAR 1997"
  skip 2 line
  on every row
    need 16 lines
    print des
    print ".E ",id," 961031 Z DH00/",pel,ps2,dur,t,s,e,p,"/DIE1/
    print ".E1",column 10, ": oct : ",oct using "-####.##",octq, "/:", octp u
    sing "####.##", oct / octp * 100. using '####',"%"
```
tcl-isql example
#!/usr/bin/sws_wish

set mons { 1 2 3 4 5 6 7 8 9 10 11 12 }
blas::vector p 0

sql database fastetc
blas::graph .gr -width 800
.gr element create "el" -label prism -xdata $mons -ydata p
.gr element create "e2" -label ofc -xdata $mons -ydata o -color red

pack .gr

set ofscurs [sql open "select id, jan, feb, mar, apr, may, jun, jul, 
    aug, sep, oct, nov, dec \n    from b_avg where pe1 = ? and pe2 = ? and dur = ? and t = ? and s = ? 
    and e = ? and p = ? order by id" T A I P G X M ]

set ofsrow [sql fetch $ofscurs]

while {$ofsrow != ""} {
    puts $ofsrow
    set [lrange $ofsrow 1 end]
    set prismcurs [sql open "select id, jan, feb, mar, apr, may, jun, jul, 
        aug, sep, oct, nov, dec \n        from b_avg where id = ? and pe1 = ? and pe2 = ? and dur = ? and t = ? 
        and s = ? and e = ? and p = ? order by id" [lindex $ofsrow 0] T A I P G X M ]
    set prismrow [sql fetch $prismcurs]; puts $prismrow;
    set [lrange $prismrow 1 end]
    sql close $prismcurs
    set ofsrow [sql fetch $ofscurs]
    .gr configure -title [lindex $prismrow 0]
    update idletasks
    set nextone [gets stdin]
}

sql close $ofscurs
Summary

- ISQL provides a quick, easy way to view and manipulate data.
  - Forms – insert, update and query. Also output to file in screen form or unl form
  - Reports – outputs formatted report with headers/trailers and page breaks
  - tcl-isql – easy programmatic manipulation of data
- We fought for it, so use it!
- This presentation url: [www.cbrfc.noaa.gov](http://www.cbrfc.noaa.gov) -> presentations -> 2003