Presentation Title: An ArcView-Based Peak Flow Sanitary Sewer Model and Master Planning Tool; and Some Thoughts on Object-Based GIS Modeling of Sewers
Presenter: Harry Poulter PE, MCSD Camp Dresser & McKee, Inc.
Software: Arcview 3.x

Detailed algorithm for sewer modeling program

User chooses pipes theme; one or more flow scenarios.

Program checks to make sure pipes in theme have been ordered and that all required fields are as specified in GLOBPARMS table.

Summarize Pump and Area flow tables to create summary tables of flow by loadpoint node and scenario.

Pipe order numbers are sorted in descending order – this list is used to iterate through pipes.

For each scenario:

Delete all records for previous run of this scenario from results table (can copy table to preserve previous run information – future version of program will offer scenario management)

For each pipe by order descending:

If pipe is a force main, assign the firm capacity of the pump station as the pump flow component, discard all other flows, and skip to "Write results" – alternate version of program compares total upstream flows to pump station capacity and writes summary table) …

Calculate full flow capacity of pipe, using pipe data and Manning's equation.

Find I/I zones for pipe, calculate I/I components for current pipe (use 0 if pipe is force main). Assign these flow components (GWI, RDI/I) to the current pipe.

Query the area flow and pump summary tables for the total flow assigned to the upstream node of the current pipe. Assign these flow components (bwf, pump flow) to the current pipe.

Now, query the results table (MOD_FLOW) for any upstream pipes whose downstream node matches the upstream node of the current pipe. Add these flow components to the current pipe.

Compute the base flow peaking factor.

Check for a split pipe and apply split rule if necessary.

Compute the following for the pipe:

Design flow = (BWF * peaking factor) + pump flow + GWI + RDI/I
Design capacity = (full capacity * design %full / 100)
Percent full = (Design flow / full capacity) * 100
Full Pipe Available = Full capacity - Design flow
Des Pipe Available = Design capacity - Design flow

Relief size - actual and standard size: the minimum diameter of a required relief (parallel) line at the same slope; and the corresponding standard size; based on design capacity and design flow.

Replacement size - actual and standard size: the minimum diameter of a required replacement line at the same slope; and the corresponding standard size; based on design capacity and design flow.

Write all results, including individual flow components, for the current pipe to the MOD_FLOW table, and close the table again.

Next pipe
Next scenario

That's it! The only tricky bit is the fact that we are building the MOD_FLOW table and querying it again each time. This can take a LONG time after you've run a couple thousand pipes, if you're not using indexes correctly. Tips:

- Index on numbers rather than strings if possible.
- Place query on index field within parentheses
- Don't use complex queries, even if all query fields are indexed. Instead of:

```
yesno = theModFlowVTab.Query("(([scenid] = " + theScen.quote + ") AND ([DSMHID] = " +
PipeUSMH.asstring.Quote + "))",
        theModFlowBitMap,#VTAB_SELTYPE_NEW)" theModFlowVTab.UpdateSelection
```

Use two operations:

```
yesno = theModFlowVTab.Query("([scenid] = " + theScen.quote + )", theModFlowBitMap,
#VTAB_SELTYPE_NEW)
 theModFlowVTab.UpdateSelection

 yesno = theModFlowVTab.Query("([DSMHID] = " + PipeUSMH.asstring.Quote + ")",
theModFlowBitMap, #VTAB_SELTYPE_AND)
 theModFlowVTab.UpdateSelection
```