Documentation for rat functions 11/4/02

1.0 General Information

1.1 Application Description

The rat utility program is a group of functions that can be called by other programs to aid in retrieving rating curves and in converting stage to flow and vice-versa. Here is a description of the available functions:

- getrc - calls the function get_rate (described below) to retrieve the desired rating curve and shift information and returns it in the defined structure RatingTable (see Attachment A for structure definition).
    - get_rate
        - retrieves the correct rating curve and shift for a given id, PE (SHEF Physical Element – either HG or HP), and date and places it in the structure RatingTable
        - calculates the slope (m) and y-intercept (b) using the bottom two points of the rating curve to be used for linearly extending the table below the given start point
- GetFlow - returns the flow given a stage. It applies the datum adjustment and then the shift to the given stage, determines where this shifted stage is in the given rating table and then calls the appropriate function (listed below) to perform the conversion. It will retrieve the correct rating table using get_rate (described above) if it has not already been done.
    - BelowStage2Flow
        - called by GetFlow when the shifted stage is below the given rating table
        - rating curve is extended linearly
    - AboveStage2Flow
        - called by GetFlow when the shifted stage is above the rating table
        - rating curve is extended logarithmically
    - ExactStage2Flow
        - called by GetFlow when the shifted stage is one of the points in the given rating curve
    - InterpolateStage2Flow
        - called by GetFlow when the shifted stage falls between points on the given rating curve
        - logarithmic interpolation is used
- GetStage - returns the stage given a flow. It determines where the flow is in the given rating table, calls the appropriate function (listed below) to

perform the conversion and then applies the shift (the datum adjustment is not applied). It will retrieve the correct rating table using get_rate (described above) if it has not already been done

  - BelowFlow2Stage
    - called by GetStage when the given flow is below the given rating table
    - rating curve is extended linearly
  - AboveFlow2Stage
    - called by GetStage when the given flow is above the rating table
    - rating curve is extended logarithmically
  - ExactFlow2Stage
    - called by GetStage when the given flow is one of the points in the given rating curve
  - InterpolateFlow2Stage
    - called by GetStage when the given flow falls between points on the given rating curve
    - logarithmic interpolation is used

- Astage - given a stage, it returns the datum-adjusted stage for the given rating table.

This program is written in esql/C.

1.2 Design Considerations

None.

1.3 Application Assumptions

None.

2.0 Configuration Information

None.

3.0 User How-To

In order to use the rat rating curve functions the calling program must open the Informix database and the following esql prepare and declare cursor statements must be included:

```
exec sql prepare selectrate from
        "select * from rating
          where lid matches ? and pe1 matches ? and pe2 matches ? and
        tbl=?";

exec sql declare ratecursor cursor for selectrate;
exec sql prepare selectshift from
        “select * from ratingshift
          where lid matches ? and pe1 matches ? and pe2 matches ?
          order by begin_date desc”;

exec sql  declare shiftcursor cursor for selectshift;
```

The header file **rat.h** (located in /arc/lib/cbrfc) must also be included. It defines the structure variables RatingTable and ShiftTable used for storing the rating curve and shift information and also declares the functions (see Attachment A). The program file is also located in /arc/lib/cbrfc and is part of the cbrfc.a library file. In order for the calling program to see the functions simply include the cbrfc.a library at compile time.

The programmer will most likely use the functions getrc, GetFlow, and GetStage (and possibly Astage). The argument lists and return values for each of these functions are as follows:

- getrc(char *lid, char pe1, char pe2, int date, int *Warning)
  - lid = id of desired rating table
  - pe1 = first part of Physical Element code (H)
  - pe2 = second part of Physical Element code (G, P)
  - date = date of desired rating table
  - Warning = error return (see below)
  - return value is the rating and shift tables in the RatingTable structure format (see Attachment A for definition)
- GetFlow(char *lid, char pe1, char pe2, int date, double Stage, int *Warning)
  - lid = location id of stage value
  - pe1 = first part of Physical Element code (H)
  - pe2 = second part of Physical Element code (G, P)
  - date = date of stage value
  - Stage = stage value to convert to flow
  - Warning = error return (see below)
  - return value is the flow as an integer
- GetStage(char *lid, char pe1, char pe2, int date, int Flow, int *Warning)
  - lid = location id of flow value
  - pe1 = first part of Physical Element code (H)
  - pe2 = second part of Physical Element code (G, P)
  - date = date of flow value
  - Flow = flow value to convert to stage
  - Warning = error return (see below)
  - return value is the stage as a double
- Astage(char *lid, char pe1, char pe2, int date, double Stage, int *Warning)
  - lid = location id of stage value
  - pe1 = first part of Physical Element code (H)

  - pe2 = second part of Physical Element code (G, P)
  - date = date of stage value
  - Stage = stage value to adjust
  - Warning = error return (-1 = error retrieving rating table; otherwise successful)
  - return value is the datum-adjusted stage as a double

Each of these programs has an error/warning variable as its last argument. The following are its possible values:

- getrc
  - 1 = successful
  - –1 = no entry found in the database
  - –2 = bad rating table
  - other = error retrieving rating table; the returned value is the sql status code (SQLCODE)
- GetFlow and GetStage
  - 0 = value within the defined rating table
  - 1 = value below the defined rating table
  - 2 = value above the defined rating table
  - 3 = error in extending the rating table
  - –1 = error retrieving the rating table
- Astage
  - –1 = error retrieving the rating table
  - other = successful

4.0 Troubleshooting Information

If the program seems to be having trouble retrieving rating curve information for a site, check to make sure it has entries in both the rating and ratingshift tables of the archive database that correspond (i.e. have the same table versions).

5.0 Installation Instructions

6.0 Maintenance Information

Originating Programmer/Office: Colorado Basin River Forecast Center
Salt Lake City, UT

Maintenance Programmer/Office: Alcorn, Brenda
Colorado Basin River Forecast Center
Salt Lake City, UT

7.0 References

None.

Attachment A
rat.h

```
typedef struct ShiftTable {
        double Val[4];
        double Sh[4];
        int max;
} ShiftTable;

typedef struct RatingTable {
        double RatStg[100];
        int RatQ[100];
        ShiftTable Shift;
        double Hmin, HMax;
        int Qmin, Qmax, AryMax;
        int bdate, edate;
        int tbl_ver;
        int ofs;
        double datum_adj;
} RatingTable;

RatingTable getrc(char *, char, char, int, int*);

int get_rate(char *, char, char, int);

int GetFlow(char *, char, char, int, double, int *);

int BelowStage2Flow(RatingTable, double);

int AboveStage2Flow(RatingTable, double);

int ExactStage2Flow(RatingTable, int);

int InterpolateStage2Flow(RatingTable, double, int);

double GetStage(char *, char, char, int, int, int *);

double BelowFlow2Stage(RatingTable, int);

double AboveFlow2Stage(RatingTable, int);

double ExactFlow2Stage(RatingTable, int);

double InterpolateFlow2Stage(RatingTable, int, int);

double Astage(char *, char, char, int, double, int *);
```