```
$ EOLCOM >
$ TITLE RIO GRANDE BASIN HYDROECONOMIC PROTOTYPE
$ OFFSYMXREF OFFSYMLIST OFFLISTING OFFUPPER

OPTION LIMROW = 0, LIMCOL = 0;
option iterlim = 10000000;

$ONTEXT

* ---------------------------------------------------------------------------------------
Output control commands above vary listing's appearance

EOLCOM > tells GAMS to ignore anything in the line's text after >
OFFLISTING deletes all program lines and just includes GAMS listing
Setting LIMROW = 0 eliminates all equations in the GAMS listing
It saves space, but is usually a bad idea till the model is known bullet proof

Colors:  We suggest going to 'file' then to 'options,'
then choose as many colors as possible for varying kinds of GAMS syntax
It simplifies error trapping.

* ---------------------------------------------------------------------------------------

April 28, 2006

Rio Grande Basin Model:

Contains most elements of full Upper Rio Grande Basin Model from 2001-2006.
Complete hydrological mass balance for all surface water and groundwater
All use, through diversions or pumping, deplete the river and/or its connected aquifer
All economic benefits derive from agriculture, M&I, and reservoir recreation
Endangered species needs set minimum flows
Terminal conditions are specified for flow, aquifers,and reservoirs

* ---------------------------------------------------------------------------------------
Originally developed by Dr. Jim Booker (1996-2001), and funded by

  US Geological Survey,
  Water Resources Research Institutes and Ag Experiment Stations of CO, NM, and TX

Currently (2001-2006) sponsored by Rio Grande Basin Initiative (USDA and US Congress)

Model  Contacts:  Frank Ward, New Mexico State University:  fward@nmsu.edu
                  Jim  Booker, Siena College, New York:     jbooker@siena.edu
* ---------------------------------------------------------------------------------------

Model has these flow nodes:

9 inflow headwater nodes
11 river gauge nodes
7 diversion nodes
7 application nodes
7 consumptive use nodes
7 seepage nodes
7 net seepage nodes
7 pumping nodes
```

```
7 surface water to river return flow nodes

7 groundwater to river return flow nodes
7 groundwater to aquifer volume gain nodes

6 reservoir release to river node
6 reservoir evaporation nodes

and these stock nodes:

6 reservoir node
7 aquifer nodes

* -------------------------------------------------------------------------------------
FLOWS:  Spatial unit for FLOWS is set (index) i.
Each element in the set i is assigned to one water use subset (category)
Subset categories include:

   1. Inflow (headwater) nodes to the system,                    inflow(i);

   2. Nodes on a river or tributary                              river(i);

   3  Diversion nodes                                            divert(i);

   4. Application nodes                                          apply(i);

   5. Consumptive use nodes                                      use(i);

   6. Seepage nodes                                              seep(i);

   7. Net Seepage nodes                                          netseep(i);

   8. Pumping nodes                                              pump(i);

   9. Surface return flow nodes directly to the river,           return(i);

  10. Groundwater return flow groundwater net seepage            gwflow(i);

  11. Gain in aquifer volume nodes from net seepage              aqflow(i);

  12. NET reservoir releases from storage, outflow - inflow      rel(i);

  13. reservoir evaporation nodes, based on surf area            evp(i);

STOCKS: Spatial unit for STOCKS is the set index u.
Each element of the set u is assigned to one water use subset (category).
Subset categories are:

   1. Reservoir volume nodes                                     res(u);

   2. Aquifer volume node                                        aqf(u);
* -------------------------------------------------------------------------------------

TABLE OF CONTENTS

   Section 1. Sets
```

```
   Section 2. Data
   Section 3. Variables
   Section 4. Equations
   Section 5. Models
   Section 6. Solves
   Section 7. Displays
* ------------------------------------------------------------------------------------

$OFFTEXT

**************** Section 1 ***********************************************************
*  The following sets are specified as indices                                          *
*  for parameters (data), variables, and equations                                      *
**************************************************************************************

SETS

**************************************************************************************
i     Flows -- location of important nodes in RG Basin -- Colorado to Mexico
**************************************************************************************

/     RG-DN_h_f      Headwater flow nodes                        inflow(i)
      Conejos_h_f
      CBasn_h_f
      SangDC_h_f
      SJCham_h_f
      Chama_h_f
      Jemez_h_f
      Puerco_h_f
      Salado_h_f

      Lobatos_v_f    River gage measurement nodes                river(i)
      Embudo_v_f
      Chamita_v_f
      Otowi_v_f
      Coch_g_v_f
      Acacia_v_f
      Marcial_v_f
      EB_g_v_f
      CA_g_v_f
      ElPaso_v_f
      Quitman_v_f

      SLV_f          All agriculture nodes                       ag(i)
      MRGCD_f
      EBID_f
      MXAG_f
      EPAG_f

      ABQMI_f        All M&I nodes                               mi(i)
      EPMI_f

      SLV_d_f        Diversion nodes                             divert(i)
      ABQMI_d_f
      MRGCD_d_f
      EBID_d_f
```

MXAG_d_f		
EPMI_d_f		
EPAG_d_f		
SLV_a_f	*Application nodes*	*apply(i)*
ABQMI_a_f		
MRGCD_a_f		
EBID_a_f		
MXAG_a_f		
EPMI_a_f		
EPAG_a_f		
SLV_u_f	*Consumptive use (ET) nodes*	*use(i)*
ABQMI_u_f		
MRGCD_u_f		
EBID_u_f		
MXAG_u_f		
EPMI_u_f		
EPAG_u_f		
SLV_s_f	*Seepage nodes*	*seep(i)*
ABQMI_s_f		
MRGCD_s_f		
EBID_s_f		
MXAG_s_f		
EPMI_s_f		
EPAG_s_f		
SLV_p_f	*Pumping nodes*	*pump(i)*
ABQMI_p_f		
MRGCD_p_f		
EBID_p_f		
MXAG_p_f		
EPMI_p_f		
EPAG_p_f		
SLV_n_f	*Net Seepage nodes*	*netseep(i)*
ABQMI_n_f		
MRGCD_n_f		
EBID_n_f		
MXAG_n_f		
EPMI_n_f		
EPAG_n_f		
SLV_r_f	*Return flow (surface) nodes*	*return(i)*
ABQMI_r_f		
MRGCD_r_f		
EBID_r_f		
MXAG_r_f		
EPMI_r_f		
EPAG_r_f		
SLV_g_f	*Groundwater return flow to riv nodes*	*gwflow(i)*
ABQMI_g_f		
MRGCD_g_f		
EBID_g_f		

```
      MXAG_g_f
      EPMI_g_f
      EPAG_g_f

      SLV_dq_f       Gain in aquifer volume nodes               aqflow(i)
      ABQMI_dq_f
      MRGCD_dq_f
      EBID_dq_f
      MXAG_dq_f
      EPMI_dq_f
      EPAG_dq_f

      HE_rel_f       Reservoir-to-river release flow nodes      release(i)
      EV_rel_f
      AB_rel_f
      CO_rel_f
      EB_rel_f
      CA_rel_f

      HE_evp_f        Reservoir evaporation flow nodes           evp(i)
      EV_evp_f
      AB_evp_f
      CO_evp_f
      EB_evp_f
      CA_evp_f
/

*****************************************************************************************
*     Subsets of all Flow nodes above by class (function)
*****************************************************************************************

inflow(i)            Headwater flow nodes                        inflow(i)

/     RG-DN_h_f      Rio Grande headwaters at Del Norte gage CO
      Conejos_h_f    Conjeos River headwaters-CO
      CBasn_h_f      Closed Basin Project-CO-pumps into RG for RGCompact
      SangDC_h_f     Sangre De Cristo Headwater Flows-NM below Lobatos
      SJCham_h_f     San Juan Chama Interbasin transfer from CO to NM
      Chama_h_f      Rio Chama headwaters near CO-NM state line
      Jemez_h_f      Jemez River headwaters above Alb NM
      Puerco_h_f     Rio Puerco above Socorro NM
      Salado_h_f     Rio Salado below Socorro NM
/

river(i)             River gage measurement nodes                river(i)

/     Lobatos_v_f    Lobatos gauge on RG at CO-NM state line
      Embudo_v_f     Embudo gauge on RG northern NM
      Chamita_v_f    Chamita gauge on Rio Chama northern NM
      Otowi_v_f      Otowi gauge on RG downstream of Chama RG confluence
      Coch_g_v_f     Cochiti Lake outflow gage below Santa Fe NM
      Acacia_v_f     San Acacia gauge near Socorro NM
      Marcial_v_f    San Marcial gauge below Socorro NM
      EB_g_v_f       Elephant Butte Lake outflow gauge near TorC NM
      CA_g_v_f       Caballo Lake outflow gauge near Hatch NM
      ElPaso_v_f     Rio Grande flow at El Paso below MX delivery
```

```
      Quitman_v_f    Rio Grande flow at  Fort Quitman TX
/

ag(i)                All agriculture nodes                      ag(i)

/     SLV_f          Rio Grande Conservancy ag in SL Valley CO
      MRGCD_f        Middle RG Conservancy District ag near Albuq NM
      EBID_f         Elephant Butte Irr Dist ag near Las Cruces NM
      MXAG_f         Mexican irrigated ag diverted at NM TX MX border
      EPAG_f         West TX irrigated Ag near El Paso TX
/

mi(i)                All Muncipal and Industrial (M&I) nodes    mi(i)

/     ABQMI_f       City of Albuquerque NM M&I
      EPMI_f        City of El Paso TX M&I
/

divert(i)           Diversion nodes                            divert(i)

/     SLV_d_f       ag(i) + mi(i) nodes
      ABQMI_d_f
      MRGCD_d_f
      EBID_d_f
      MXAG_d_f
      EPMI_d_f
      EPAG_d_f
/

apply(i)            Application nodes                            apply(i)

/     SLV_a_f       same nodes as divert(i)
      ABQMI_a_f
      MRGCD_a_f
      EBID_a_f
      MXAG_a_f
      EPMI_a_f
      EPAG_a_f
/

use(i)              Use (ET consumption) nodes = div nodes       use(i)

/     SLV_u_f       same nodes as divert(i)
      ABQMI_u_f
      MRGCD_u_f
      EBID_u_f
      MXAG_u_f
      EPMI_u_f
      EPAG_u_f
/

seep(i)             Seepage nodes                               seep(i)

/     SLV_s_f       same nodes as divert(i)
      ABQMI_s_f
      MRGCD_s_f
```

```
      EBID_s_f
      MXAG_s_f
      EPMI_s_f
      EPAG_s_f
/

pump(i)             Pumping nodes                          pump(i)

/     SLV_p_f       same nodes as divert(i)
      ABQMI_p_f
      MRGCD_p_f
      EBID_p_f
      MXAG_p_f
      EPMI_p_f
      EPAG_p_f
/

netseep(i)          Net Seepage nodes                      netseep(i)

/     SLV_n_f       same nodes as divert(i)
      ABQMI_n_f
      MRGCD_n_f
      EBID_n_f
      MXAG_n_f
      EPMI_n_f
      EPAG_n_f
/

return(i)           return (surface) flow nodes           return(i)

/     SLV_r_f       same nodes as divert(i)
      ABQMI_r_f
      MRGCD_r_f
      EBID_r_f
      MXAG_r_f
      EPMI_r_f
      EPAG_r_f
/

gwflow(i)           Groundwater return flow nodes         gwflow(i)

/     SLV_g_f       same nodes as divert(i)
      ABQMI_g_f
      MRGCD_g_f
      EBID_g_f
      MXAG_g_f
      EPMI_g_f
      EPAG_g_f
/

aqflow(i)           Addition to aquifer stock nodes      aqflow(i)

/    SLV_dq_f       same nodes as divert(i)
     ABQMI_dq_f
     MRGCD_dq_f
     EBID_dq_f
```

```
     MXAG_dq_f
     EPMI_dq_f
     EPAG_dq_f
/

release(i)          Reservoir to river release flow nodes     release(i)

/     HE_rel_f      Heron (net) reservoir releases to Willow Creek-Rio Chama = outflow - inflow
      EV_rel_f      El Vado reservoir releases to Rio Chama
      AB_rel_f      Abiquiu reservoir releases to Rio Chama
      CO_rel_f      Cochiti reservoir releases to Rio Grande mainstem
      EB_rel_f      Elephant Butte reservoir releases to Rio Grande mainstem
      CA_rel_f      Caballo  reservoir releases to Rio Grande mainstem
/

evap(i)             Reservoir evaporation                     evap(i)

/     HE_evp_f      Heron reservoir evaporation = fn of annual ave exposed surface area
      EV_evp_f      El Vado reservoir evaporation
      AB_evp_f      Abiquiu reservoir evaporation
      CO_evp_f      Cochiti reservoir evaporation
      EB_evp_f      Elephant Butte reservoir evaporation
      CA_evp_f      Caballo Reservoir evaporation
/

****************************************************************************************
mu(use)     Municipal and Industrial use nodes
****************************************************************************************

/     ABQMI_u_f     Alb M&I
      EPMI_u_f      El Paso M&I
/

****************************************************************************************
au(use)     Agricultural use nodes
****************************************************************************************

/     SLV_u_f       SL Valley
      MRGCD_u_f     Middle Rio G Cons Distr
      EBID_u_f      EButte Irr Dist
      EPAG_u_f      EPaso Ag
/


****************************************************************************************
ad(divert)     Agricultural divert nodes
****************************************************************************************

/     SLV_d_f       SL Valley
      MRGCD_d_f     Middle Rio G Cons Distr
      EBID_d_f      EButte Irr Dist
      EPAG_d_f      EPaso Ag
/


****************************************************************************************
```

```
u     Stocks--location of important nodes on Rio Grande Colorado to Mexico
*****************************************************************************************

/     HE_res_s        Reservoir stock nodes                     res(u)
      EV_res_s
      AB_res_s
      CO_res_s
      EB_res_s
      CA_res_s

      SLV_aqf_s       Aquifer stock nodes                       aqf(u)
      ABQMI_aqf_s
      MRGCD_aqf_s
      EBID_aqf_s
      MXAG_aqf_s
      EPMI_aqf_s
      EPAG_aqf_s
/

*****************************************************************************************
*    Stock subsets
*****************************************************************************************

res(u)               Reservoir stock nodes                       res(u)

/     HE_res_s      Heron Reservoir (stock) near CO-NM state line
      EV_res_s      El  Vado Reservoir downstream of Heron on Chama
      AB_res_s      Abiquiu Reservoir on Chama above Espanola NM
      CO_res_s      Cochiti Reservoir on RGR Mainstem downstream of Santa Fe
      EB_res_s      Elephant Butte Reservoir near T or C NM
      CA_res_s      Caballo Reservoir downstream of T or C NM
/

aqf(u)               Aquifer stock nodes                         aqf(u)

/     SLV_aqf_s      SLV aquifer supporting irrigated ag
      ABQMI_aqf_s    Alb area aquifer (Lake Superior) supporting M&I pumping
      MRGCD_aqf_s    MRGCD aquifer area (near Socorro NM)
      EBID_aqf_s     EBID aquifer area (Mesilla Bolson)
      MXAG_aqf_s     Mexico Ag aquifer (Hueco Bolson) supporting ag
      EPMI_aqf_s     El Paso M&I aquifer (Hueco bolson) supporting M&I pump
      EPAG_aqf_s     El Paso ag aquifer area (Hueco Bolson)
/
*****************************************************************************************
t     time
*****************************************************************************************

/     2006*2025       years 2006 - 2025
/

tlast(t)             terminal period among all periods above
;

tlast(t) = yes $(ord(t) eq card(t));  > GAMS language -- picks last pd
;
```

```
* -----------------------------------------------------------------------------------------
* Some mathematical tricks below streamline the model
* -----------------------------------------------------------------------------------------

* lets some tables' nodes be rows or columns

ALIAS (i,ip);
ALIAS (t,tp);
ALIAS (river, riverp);
ALIAS (divert, divertp);

* tables below defined as identity matrices, to compare different but equivalent nodes
* e.g. a divert node is the same place as an apply, use, return, pump node

parameter
ID_au(apply,   use)  identity matrix connects apply nodes to use nodes
ID_pu(pump,    use)  identity matrix connects pump  nodes to use nodes
ID_pa(pump,  apply)  identify matrix connects pump  nodes to apply nodes
ID_ap(apply,  pump)  identity matrix connects apply nodes to pump nodes
ID_ad(apply,divert)  identity matrix connects apply nodes to divert nodes
;

ID_au(apply,   use) $ (ord(apply) eq ord(use   )) = 1;
ID_pu(pump,    use) $ (ord(pump ) eq ord(use   )) = 1;
ID_pa(pump,  apply) $ (ord(pump ) eq ord(apply )) = 1;
ID_ap(apply,  pump) $ (ord(apply) eq ord(pump  )) = 1;
ID_ad(apply,divert) $ (ord(apply) eq ord(divert)) = 1;

*----------------------------------------------------------------------------------------

**************** Section 2 **************************************************************
*  This section defines all data in 3 formats                                           *
*  1.  Scalars (single numbers),                                                        *
*  2.  Parameters (columns of numbers) or                                               *
*  3.  Tables (data in rows and columns)                                                *
*****************************************************************************************

*  Below are several maps summarizing a basin's geometry
*  By geometry we mean location of mainstems, tributaries, confluence,
*  source nodes, use nodes, return flow nodes, reservoir nodes, etc.
*  Basin geometry is summarized through judicious use of numbers 1, -1, and 0 (blank)

*****************************************************************************************
*  Map #1:

*  Each column below is a streamgage.  Each row is a source or use of water.
*  Flow at ea gage (column) is directly influenced by at least 1 upstream row.
*  SOURCE adds to columns flow             (+1)
*  USE deplete from col flow               (-1)
*  BLANK has no effect on col flow         (  )
*  Geometry accounts for all sources (supplies) and uses (demands) in basin

*  Map is used to produce coefficients in equations below to define:
*  X(river) = Bhv * X(inflow) + Bvv * X(river)   + Bdv * X(divert)
*           + Brv * X(return) + Bgv * X(gwflows) + BLr * X(rel)
*
```

```
*  These B coeff vectors are stacked below as a single matrix, Bv
*****************************************************************************************

TABLE Bv(i,river)    Hydrologic Balance Table

****************************   Column Heads are River Gauges    *************************

             Lobatos_v_f  Embudo_v_f  Chamita_v_f  Otowi_v_f Coch_g_v_f Acacia_v_f Marcial_v_f EB_g_v_f  CA_g_v_f ElPaso_v_f  Quitman_v_f

* ---------------- headwater inflow node rows (+) ----------------------------------------------------------------------------------
RG-DN_h_f           1
Conejos_h_f         1
CBasn_h_f           1
SangDC_h_f                   1
SJCham_h_f                               1
Chama_h_f                                1
Jemez_h_f                                                                   1
Puerco_h_f                                                                  1
Salado_h_f                                                                             1
* ---------------- river gage node rows (+) ----------------------------------------------------------------------------------------
Lobatos_v_f                  1
Embudo_v_f                                             1
Chamita_v_f                                            1
Otowi_v_f                                                         1
Coch_g_v_f                                                                  1
Acacia_v_f                                                                              1
Marcial_v_f                                                                                        1
EB_g_v_f                                                                                                    1
CA_g_v_f                                                                                                              1
ElPaso_v_f                                                                                                                    1
Quitman_v_f
* --------------- diversion nodes  (-)  ---------------------------------------------------------------------------------------------
SLV_d_f           -1
ABQMI_d_f                                                                 -1
MRGCD_d_f                                                                 -1
EBID_d_f                                                                                                             -1
MXAG_d_f                                                                                                             -1
EPMI_d_f                                                                                                             -1
EPAG_d_f                                                                                                                     -1
* -------------- return flow node rows (+) ----------------------------------------------------------------------------------------
SLV_r_f            1
ABQMI_r_f                                                                  1
MRGCD_r_f                                                                  1
EBID_r_f                                                                                                              1
MXAG_r_f                                                                                                              1
EPMI_r_f                                                                                                              1
EPAG_r_f                                                                                                                      1
* ------------- groundwater inflow node rows (+) ----------------------------------------------------------------------------------
SLV_g_f            1
ABQMI_g_f                                                                  1
MRGCD_g_f                                                                  1
EBID_g_f                                                                                                              1
MXAG_g_f                                                                                                              1
EPMI_g_f                                                                                                              1
EPAG_g_f                                                                                                                     1
* ------------- reservoir release (outflow) to river -- stock-to-flow rows (+) ----------------------------------------------------
```

```
HE_rel_f                              1
EV_rel_f                              1
AB_rel_f                              1
CO_rel_f                                                          1
EB_rel_f                                                                                         1
CA_rel_f                                                                                                  1
;
************************************************************************************************************************************
* Map #2:

* Enforces nonnegative flows at each use node (wet river)
* water sources are rows.  Diversion nodes are columns.
* For any column, diversion < summed flows from upstream sources (rows)
* e.g. SLV Colorado ag use < flows from RG and Conejos headwater sources

* X(divert) < Bhd * X(inflow) + Brd * X(river)  + Bdd * X(divert) +
*             Brd * X(return) + Bgd * X(gwflow) + BLd * X(rel)

* These B coeff vectors are stacked below as the matrix, Bd
************************************************************************************************************************************

TABLE Bd(i, divert)       Wet river table

* -------   Col Heads are Diversion nodes ----------------------------------------------------

                SLV_d_f      ABQMI_d_f   MRGCD_d_f    EBID_d_f  MXAG_d_f   EPMI_d_f   EPAG_d_f

* ----------------------- headwater inflow nodes (+)-------------------------------------------
RG-DN_h_f           1
Conejos_h_f         1
CBasn_h_f
SangDC_h_f
SJCham_h_f
Chama_h_f
Jemez_h_f                        1
Puerco_h_f                                  1
Salado_h_f
* ----------------------- river gage nodes -----------------------------------------------------
Lobatos_v_f
Embudo_v_f
Chamita_v_f
Otowi_v_f
Coch_g_v_f                       1          1
Acacia_v_f
Marcial_v_f
EB_g_v_f
CA_g_v_f                                                1         1          1
ElPaso_v_f                                                                              1
Quitman_v_f
* ----------------------- diversion nodes  (-)  ---------------------------------------------
SLV_d_f
ABQMI_d_f                                  -1
MRGCD_d_f
EBID_d_f                                                         -1         -1
MXAG_d_f                                                                    -1
EPMI_d_f
```

```
EPAG_d_f
* ---------------------- return flow nodes (+) --------------------------------------------
SLV_r_f
ABQMI_r_f                                   1
MRGCD_r_f
EBID_r_f                                                           1          1
MXAG_r_f                                                                      1
EPMI_r_f
EPAG_r_f
* ---------------------- groundwater inflow nodes (+) -------------------------------------
SLV_g_f
ABQMI_g_f                                   1
MRGCD_g_f
EBID_g_f                                                           1         1
MXAG_g_f                                                                     1
EPMI_g_f
EPAG_g_f
* ----------------  reservoir outflow stock-to-flow node row (+)-------------------------------
HE_rel_f
EV_rel_f
AB_rel_f
CO_rel_f
EB_rel_f
CA_rel_f
* ------------------------------------------------------------------------------------------

* Map #3

* Defines pan evaporation in inches - translate to af lost per per exposed acre per year by reservoir

TABLE Be(evap, res)

               HE_res_s  EV_res_s  AB_res_s   CO_res_s   EB_res_s  CA_res_s
HE_evp_f          56.0
EV_evp_f                    56.0
AB_evp_f                              56.0
CO_evp_f                                        93.0
EB_evp_f                                                   111.2
CA_evp_f                                                              111.2
;

Be(evap,res) = Be(evap,res)/12;         > inches to feet

*display Be;
* ------------------------------------------------------------------------------------------

*****************************************************************************************

*****************************************************************************************
* Map #4:

* Defines water applied as diversion plus pumping

* X(apply) = Bda * X(divert) + Bpa * X(pump)

* These two B coefficient vectors are stacked below as the matrix, Ba
```

```
TABLE Ba(i, apply)   Table defines water applied

* ---------------------------- Apply nodes -------------------------------------------------
              SLV_a_f   ABQMI_a_f  MRGCD_a_f    EBID_a_f  MXAG_a_f  EPMI_a_f   EPAG_a_f
* ------------------------ divert nodes (+) --------------------------------------------
SLV_d_f         1
ABQMI_d_f                  1
MRGCD_d_f                            1
EBID_d_f                                         1
MXAG_d_f                                                   1
EPMI_d_f                                                            1
EPAG_d_f                                                                       1
* ----------------------   pumping nodes (+) -----------------------------------------
SLV_p_f         1
ABQMI_p_f                  1
MRGCD_p_f                            1
EBID_p_f                                         1
MXAG_p_f                                                   1
EPMI_p_f                                                            1
EPAG_p_f                                                                       1
*---------------------------------------------------------------------------------
;

*****************************************************************************************
* Map #5:

* Defines use as a constant proportion, Bau, of water applied

* X(use) = Bau * X(apply)

* This coeff vector is the matrix, Bu
*****************************************************************************************

TABLE Bu(apply, use)   Table defines consumptive use

* Best data for next 3 tables may be in Papodolous report (1996?).  Generally, these data
* are scarce, scattered, and hard to validate.  Many questions in basic science and most
* questions in applied science regarding groundwater pumping connections to surface supply
* remain unanswered. For relevant theory, the best article may be a December 1940 paper by
* M King-Hubbert, The Theory of Groundwater Motion, Journal of Geology, pp 785-944 (160 pp)
* MX data are educated guesses

* Albuquerque return flows ABQ Water home page --  http://www.cabq.gov/waterresources/sjc.html

* -------------------------  Use nodes -------------------------------------------------------------
              SLV_u_f    ABQMI_u_f     MRGCD_u_f     EBID_u_f     MXAG_u_f   EPMI_u_f    EPAG_u_f
* ------------------------ apply nodes (+) -------------------------------------------------------
SLV_a_f         0.50
ABQMI_a_f                  0.441
MRGCD_a_f                                0.25
EBID_a_f                                              0.46
MXAG_a_f                                                            0.46
EPMI_a_f                                                                       0.441
EPAG_a_f                                                                                    0.46
*-----------------------------------------------------------------------------------------------
```

```
;

parameter Bu_p(use)  translates Bu above into a parameter with one value at each use node
;
Bu_p(use) = sum(apply, Bu(apply,use));

*************************************************************************************************
* Map #6:

* Defines seepage as a constant proportion, Bas, of water applied

* X(seep) = Bas * X(apply)

* This coeff vector is the matrix, Bs
*************************************************************************************************

TABLE Bs(apply, seep)   Table defines seepage to groundwater aquifer

* --------------------------  Seepage nodes   -----------------------------------------------------
               SLV_s_f    ABQMI_s_f       MRGCD_s_f     EBID_s_f     MXAG_s_f    EPMI_s_f  EPAG_s_f
* ------------------------- apply  nodes (+) ------------------------------------------------------
SLV_a_f          0.50
ABQMI_a_f                    0.059
MRGCD_a_f                                   0.19
EBID_a_f                                                  0.34
MXAG_a_f                                                              0.34
EPMI_a_f                                                                         0.059
EPAG_a_f                                                                                    0.34
*------------------------------------------------------------------------------------------------
;

**************************************************************************************************
* Map #7:

* Defines return flow as a constant proportion, Bar, of water applied

* X(return) = Bar * X(apply)

* This coeff vector is the matrix, Br
*****************************************************************************************

TABLE Br(apply, return)   Table defines return flow to river (surface flow)

* --------------------------  Return flow nodes -----------------------------------------
               SLV_r_f   ABQMI_r_f     MRGCD_r_f     EBID_r_f   MXAG_r_f  EPMI_r_f  EPAG_r_f
* ------------------------- apply nodes (+) ---------------------------------------------
SLV_a_f          0.00
ABQMI_a_f                 0.500
MRGCD_a_f                              0.56
EBID_a_f                                             0.20
MXAG_a_f                                                        0.20
EPMI_a_f                                                                 0.500
EPAG_a_f                                                                           0.20
*---------------------------------------------------------------------------------------
;
```

```
PARAMETER APPLY_FLOW_ALLOC(apply)  check: flow allocations should sum to 1
;
APPLY_FLOW_ALLOC(apply) =   sum(use,    Bu(apply, use   ))
                         +  sum(seep,   Bs(apply, seep  ))
                         +  sum(return, Br(apply, return));

*display apply_flow_alloc;

*****************************************************************************************
* Map #8:

* Table defines net seepage as seepage minus pumping
*
* Tabled entries = proportion seepage and pumping flow by net seepage column nodes
*    (+)  means the row adds to the column's net seepage
*    (-)  means the row subtracts from column's net seepage
*    ( )  no effect

* X(netseep) = Bsn * X(seep) + Bpn * X(pump)
*****************************************************************************************

* These two B coefficient vectors are stacked below as the matrix, Bn

TABLE Bn(i, netseep)         Table defines net seepage to (+) or from (-) aquifer

********************  Column Heads are Net Seepage Nodes *************************************
              SLV_n_f   ABQMI_n_f     MRGCD_n_f    EBID_n_f   MXAG_n_f   EPMI_n_f   EPAG_n_f
* ------------------------ seepage node rows (+) ---------------------------------------------
SLV_s_f          1
ABQMI_s_f                    1
MRGCD_s_f                                 1
EBID_s_f                                              1
MXAG_s_f                                                          1
EPMI_s_f                                                                     1
EPAG_s_f                                                                                1
* ------------------------ pump  node rows (-) --------------------------------------------
SLV_p_f         -1
ABQMI_p_f                   -1
MRGCD_p_f                                -1
EBID_p_f                                             -1
MXAG_p_f                                                         -1
EPMI_p_f                                                                    -1
EPAG_p_f                                                                               -1
*------------------------------------------------------------------------------------------------
;

************************************************************************************************
* Best data for next 2 tables may be in Papodolous report (1996?).  They allocate
* groundwater flow from net seepage to 2 sources-sinks:  (1) river and (2) aquifer.
* Net seepage hydrologically farther from river has a smaller pct affecting river.

* Like the tables above that allocate water application among use, seepage, and return
* flows, seepage allocation data are scarce, scattered, and hard to validate.  Many
* questions in basic science and most in applied science regarding groundwater pumping
* connections to surface supply v. aquifer drawdowns remain unanswered. For relevant theory,
* the best article may be a Dec 1940 paper by M King-Hubbert, The Theory of Groundwater
```

```
* Motion, J. of Geology, pp 785-944 (160 pages, yes 160 pages total)

* Adding downward, coefficients from next two tables should also add to 1.0, i.e. for each
* diversion node, total groundwater flow percentage summed from the river and from the
* aquifer should add to 1.0;


*****************************************************************************************
* Map #9:

* Table defines groundwater flow to river as a fn of net seepage
*
* Tabled entries = proportion net seepage contributing to groundwater flow column nodes
*    (+)  means the row adds to the column's gw flow
*    (-)  means the row subtracts from column's gw flow
*    ( )  no effect

* X(gwflow) = Bng * X(netseep)
*****************************************************************************************

TABLE Bg(netseep, gwflow)         Table defines groundwater flow to (+) or from (-) river

********************  Column Heads are Groundwater Flow Nodes *************************
              SLV_g_f  ABQMI_g_f   MRGCD_g_f   EBID_g_f  MXAG_g_f  EPMI_g_f   EPAG_g_f
* ----------------------- net seepage node rows (+) ---------------------------------
SLV_n_f         0
ABQMI_n_f               0.60
MRGCD_n_f                             0.70
EBID_n_f                                        0.975
MXAG_n_f                                                   0.20
EPMI_n_f                                                             0.60
EPAG_n_f                                                                         0.60
* -------------------------------------------------------------------------------------
;

*****************************************************************************************
* Map #10: complements Map 8.  Sum of Bg and Bq coeffs for any colum = 1 for mass balance

* Table defines added aquifer volume as a fn of net seepage
*
* Tabled entries = proportion net seepage contributing to added aquifer volume
*    (+)  means the row adds to the column's aquifer volume (change in aq stock = flow)
*    (-)  means the row subtracts from column's aquifer volume
*    ( )  no effect

* X(aqflow) = Bqg * X(netseep)
*****************************************************************************************

TABLE Bq(netseep, aqflow)         Table defines gain (+) or loss (-) in aquifer volume

********************  Column Heads are Aquifer volume change (flow) nodes  ************
             SLV_dq_f  ABQMI_dq_f  MRGCD_dq_f  EBID_dq_f  MXAG_dq_f  EPMI_dq_f  EPAG_dq_f
* ----------------------- net seepage node rows (+) ---------------------------------
SLV_n_f          1
ABQMI_n_f                 0.40
MRGCD_n_f                              0.30
```

```
EBID_n_f                                          0.025
MXAG_n_f                                                      0.80
EPMI_n_f                                                                 0.40
EPAG_n_f                                                                            0.40
* -------------------------------------------------------------------------------------------

PARAMETER SEEP_FLOW_ALLOC(netseep) check: allocations of net seepages should sum to 1
;
SEEP_FLOW_ALLOC(netseep) =     sum(gwflow,   Bg(netseep, gwflow))
                            +  sum(aqflow,   Bq(netseep, aqflow));

*display SEEP_FLOW_ALLOC;

*******************************************************************************************
* Next two maps link stocks (res and aquifers) to flows (release and net seepage)
*******************************************************************************************
* Map #11:

* Table relates reserv stocks in a pd to its prev pd' stocks minus net releases.
* For any reservoir stock node at the column head
*   (-1) :added water at flow node -- thru releases -- subtracts from column's res stock
*   (  ) :added water at flow node has no effect on column's reservoir stock

* Z(res(t)) = Z(res(t-1)) + BLr * X(release(t))
*******************************************************************************************

TABLE BLr(release, res)         Links reservoir releases to downstream flows

********* Column Heads are Reservoir Stocks -- rows are release flows  *****************
********* Table = diagonal matrix for > 1 reservoir -- only 1 for now  *****************

                     HE_res_s  EV_res_s   AB_res_s   CO_res_s   EB_res_s  CA_res_s
      HE_rel_f         -1
      EV_rel_f                   -1
      AB_rel_f                              -1
      CO_rel_f                                         -1
      EB_rel_f                                                    -1
      CA_rel_f                                                              -1
*------------------------------------------------------------------------------------------

*******************************************************************************************
* Map #12:

* Table relates reservoir stocks to evaporation
* (-1): added evap subtracts a reservoir's volume
* (  ): added evap has no effect on a reservoir's vol

TABLE Ber(evap, res)           Links reservoir evaporation to volume loss

********** Column Heads are reservoir stocks -- rows are evaporation loss flows ********
********** Table = diagonal matrix for > 1 reservoir -- only 1 for now ****************

                    HE_res_s  EV_res_s   AB_res_s   CO_res_s  EB_res_s   CA_res_s
      HE_evp_f         -1
      EV_evp_f                   -1
      AB_evp_f                              -1
```

```
      CO_evp_f                                         -1
      EB_evp_f                                                   -1
      CA_evp_f                                                             -1
*------------------------------------------------------------------------------------------

*******************************************************************************************
* Map #13:

* Table relates aquifer volume stock in a period to its prev periods' stocks plus
* net seepage (seepage minus pumping)
* For any aquifer stock node at the column head
*   (+1):added water at flow (net seep) node -- thru net seepage -- adds to col aqf stck
*   (  ):added water at pump has no effect on column's aqf stock

* Z(aqf(t)) = Z(aqf(t-1)) + BQv * X(aqflow(t))

TABLE BQv(aqflow, aqf)           Links aquifer pumping to aquifer volume and depth

********* Column Heads are Aquifer stocks -- rows are pumping flows    *****************
********* Table = diagonal matrix for > 1 aquifers  **********************************

           SLV_aqf_s  ABQMI_aqf_s  MRGCD_aqf_s  EBID_aqf_s MXAG_aqf_s  EPMI_aqf_s  EPAG_aqf_s
SLV_dq_f       1
ABQMI_dq_f                1
MRGCD_dq_f                             1
EBID_dq_f                                           1
MXAG_dq_f                                                       1
EPMI_dq_f                                                                    1
EPAG_dq_f                                                                                1

*******************************************************************************************
*  END OF BASIN HYDROLOGY MAPS                                                            *
*******************************************************************************************


*******************************************************************************************
*  ECONOMICS MAPS BEGIN                                                                   *
*******************************************************************************************

parameter scales(use)   1000s of acres by ag node Data source USDA 2002 Ag Census

/SLV_u_f         277.284
 ABQMI_u_f       107.000
 MRGCD_u_f        45.004
 EBID_u_f         89.328
 MXAG_u_f         50.000     > educated guess for MX irrigated acreage near C. Juarez, MX
 EPMI_u_f        120.553
 EPAG_u_f         37.197
/


parameter growth(use)   annual forecast growth rate of acreage or population by node

*change from MI numbers below to zero for base MI use without ag-MI transfers

/ SLV_u_f          0.0
```

```
  ABQMI_u_f        0.0240
  MRGCD_u_f        0.0
  EBID_u_f         0.0
  MXAG_u_f         0.0
  EPMI_u_f         0.0376
  EPAG_u_f         0.0
/


* abq 0.0240
* ep 0.0376

parameter scalese(res)   1000s of households receiving environmental rec reservoir benefits in basin

/HE_res_s          1
 EV_res_s          1
 AB_res_s          1
 CO_res_s          1
 EB_res_s          1
 CA_res_s          1
/

parameter growthe(res)   annual forecast growth rate of population by reservoir rec area

/HE_res_s          0.00
 EV_res_s          0.00
 AB_res_s          0.00
 CO_res_s          0.00
 EB_res_s          0.00
 CA_res_s          0.00
/

parameter

scale(use,t)     1000s forecast acres or households by node and time
scalee(res,t)    1000s forecast households benefiting from basin reservoirs by node and time
;
scale(use,t)  =  ((1 + growth(use ))**(ord(t)-1)) * scales(use);
scalee(res,t) =  ((1 + growthe(res))**(ord(t)-1)) * scalese(res);

* display scale, scalee;

TABLE Ben_u_p(use, *)  Per acre or per hhold benefit fn params for all uses
* zero US benefits for Mexican irrigated acreage
*                  ($)       ($/ac-af)    ($/ac-af2)   (af/ac-yr1)    (af-yr1)
               intercept       linear      quadratic     u_mb_0_a      u_mb_0
 SLV_u_f         195.00        265.50        -28.94       4.58          1272
 ABQMI_u_f         0.00      10843.00      -9627.00       0.56            60
 MRGCD_u_f        30.00         67.00         -5.92       5.65           255
 EBID_u_f        137.00         94.00         -2.50      18.80          1679
 MXAG_u_f          0.00          0.00          0.00       0.00             0
 EPMI_u_f          0.00       9507.00      -9392.00       0.51            61
 EPAG_u_f          0.00        193.00        -21.50       4.48           166
;
```

```
*-------------------------------------------------------------------------------------
* Reservoir Recreation benefits are a function of reservoir volume
*
* Data from Richard Cole and Frank Ward published research on Riofish, AJAE August 1997
*
* Form is Benefit = a + bZ + cZ**2

TABLE Ben_e_p(res, *)  Total benefits as a fn of visits at basin reservoirs - no scaling
*              ($)      ($/af)          ($/af2)         (af/yr)
            intercept   linear          quadratic        v_mb_0_a
HE_res_s    4246.98     7.359         -0.002092785        1759
EV_res_s    4246.98     7.359         -0.002092785        1759
AB_res_s    4246.98     7.359         -0.002092785        1759
CO_res_s     256.62     4.104         -0.002875613         714
EB_res_s     379.82     2.210         -0.000503852        2194
CA_res_s     379.82     2.210         -0.000503852        2194
;
***************************************************************************************

*-------------------------------------------------------------------------------------
* Table defines energy + capital costs of at all pumping nodes
*    (+) means that the row adds to the column's supply costs
*    ( ) means that the row has no effect on the column's supply costs

* Cost(use)  =  Cost_u_ep_p * X(pump)

TABLE Cost_u_ep_p(pump,  apply)   Energy + capital cost per acre foot pumped at use nodes

******************  Column Heads are use nodes ****************************************
            SLV_a_f   ABQMI_a_f  MRGCD_a_f   EBID_a_f   MXAG_a_f  EPMI_a_f  EPAG_a_f
SLV_p_f      40.00
ABQMI_p_f              100.00
MRGCD_p_f                         50.00
EBID_p_f                                      50.00
MXAG_p_f                                                   0
EPMI_p_f                                                           100.00
EPAG_p_f                                                                     50.00
* -----------------------------------------------------------------------------------
;

* Table defines chemical treatment costs at all pumping nodes
*    (+) means that the row adds to the column's supply costs
*    ( ) means that the row has no effect on the column's supply costs

* Cost(use)  =  Cost_u_et_p * X(pump)
* alb mi pump cost based on 2.50/1000 gallons = $815 per acre foot
* elp mi pump cost based on 2.00/1000 gallons = $652 per acre foot

TABLE Cost_u_tp_p(pump,  apply)   Chemical treatment costs per acre foot pumped at use nodes

******************  Column Heads are use nodes ****************************************
            SLV_a_f   ABQMI_a_f  MRGCD_a_f   EBID_a_f   MXAG_a_f  EPMI_a_f  EPAG_a_f
SLV_p_f        0
ABQMI_p_f              715.00
MRGCD_p_f                           0
EBID_p_f                                        0
```

```
MXAG_p_f                                                   0
EPMI_p_f                                                           715.00
EPAG_p_f                                                                       0
* ------------------------------------------------------------------------------------
;

* --------------------------------------------------------------------------------------

* Table defines energy + capital costs at all surface diversion (not pumping) nodes
*    (+) means that the row adds to the column's supply costs
*    ( ) means that the row has no effect on the column's supply costs

* Cost(use)  =  Cost_u_ed_p * X(divert)

* Cost of surface water v groundwdater treatment for cities comes from
* Mays et al, book:  Regional Water Supply Planning and Expansion Models chapter 3

TABLE Cost_u_ed_p(divert, apply)   Energy + capital costs per acre foot diverted at use nodes

*alb mi diversion cost based on average price of $2.70/1000 gall = 880/acre foot
*elp mi diversion cost based on average 2006 price of $2.50/1000 gall = 815/acre foot

******************  Column Heads are use nodes ****************************************
            SLV_a_f   ABQMI_a_f  MRGCD_a_f   EBID_a_f   MXAG_a_f  EPMI_a_f  EPAG_a_f
SLV_d_f       10
ABQMI_d_f               380
MRGCD_d_f                          10
EBID_d_f                                      10
MXAG_d_f                                                   0
EPMI_d_f                                                            250.00
EPAG_d_f                                                                      10
* ---------------------------------------------------------------------------------------
;

* Table defines chemical treatment costs at all surface diversion (not pumping) nodes
*    (+) means that the row adds to the column's supply costs
*    ( ) means that the row has no effect on the column's supply costs

* Cost(use)  =  Cost_u_td_p * X(divert)

* Cost of surface water v groundwdater treatment for cities comes from
* Mays et al, book:  Regional Water Supply Planning and Expansion Models chapter 3

TABLE Cost_u_td_p(divert, apply)   Chemical treatment costs per acre foot diverted at use nodes

******************  Column Heads are use nodes ****************************************
            SLV_a_f   ABQMI_a_f  MRGCD_a_f   EBID_a_f   MXAG_a_f  EPMI_a_f  EPAG_a_f
SLV_d_f        0
ABQMI_d_f              500.00
MRGCD_d_f                           0
EBID_d_f                                        0
MXAG_d_f                                                    0
EPMI_d_f                                                           500.00
EPAG_d_f                                                                       0
* --------------------------------------------------------------------------------------
;
```

```
* Table defines pumping costs per unit depth per acre foot at all aquifer nodes -
* it's based on energy costs and pumping efficiency

*    (+) means that the row adds to the column's pump costs
*    ( ) means that the row has no effect on the column's supply costs

* Cost(pump)  =  Cost_app * D(aqf)
* Cost_app approx = 0.50.  For a 100 foot deep aquifer cost is $50 per acre foot pumped

TABLE Cost_app_p(aqf,    pump)  gw energy + cap + op cost per extra foot depth per acre foot pumped

* 0 until (1) we replace above constant energy pump costs with these below that vary with depth
* and     (2) we correctly incorporate marginal increased depths in total mc of water use

******************  Column Heads are pumping nodes *************************************
              SLV_p_f   ABQMI_p_f  MRGCD_p_f   EBID_p_f   MXAG_p_f  EPMI_p_f  EPAG_p_f
SLV_aqf_s       0.00
ABQMI_aqf_s               0.00
MRGCD_aqf_s                           0.00
EBID_aqf_s                                         0.00
MXAG_aqf_s                                                   0.00
EPMI_aqf_s                                                            0.00
EPAG_aqf_s                                                                       0.00
* -------------------------------------------------------------------------------------
;

parameter cost_up_p(pump, apply)   total costs per acre foot pumped at apply nodes
          cost_ud_p(divert,apply) total costs per acre foot diverted at apply nodes
;
cost_up_p(pump,   apply) = cost_u_ep_p(pump,  apply) + cost_u_tp_p(pump,   apply);
cost_ud_p(divert, apply) = cost_u_ed_p(divert,apply) + cost_u_td_p(divert, apply);

PARAMETER Env_cost_p(res)  Environmental costs per yr pr acre foot volume managing NM State Parks

* based on $10 / FTE mgmt budget for NM State Parks.  Economic Impact of NM State Parks, NMSU 2004
* (1) Mgmt Cost = 0.80 * visits, (2) visits = 22.25 * water surface area,
* (3) reservoir volume to reservoir area ratio shown in ACAP relations shown below
* result is Mgmt cost = Env_cost_p * reservoir volume.  Env_cost coeff shown here
* Each extra acre foot adds less than $1 to labor costs of staffing water-based reservoirs
/
HE_res_s         0.36
EV_res_s         0.31
AB_res_s         0.50
CO_res_s         0.33
EB_res_s         0.31
CA_res_s         0.62
/

* .36, .31, .50, .33, .31,. 62
* 6.26, 6.80, 5.50, 1.33, 0.31, 2.00 (gives interior soln when reservoirs start full)

* -------------------------------------------------------------------------------------

PARAMETER MAX_PRICE_p(use)  negotiated price\af used. For 44% use\af applied - MI price = $1\1000 gal applied
```

```
/SLV_u_f          1
 ABQMI_u_f      740
 MRGCD_u_f        1
 EBID_u_f         1
 EPMI_u_f       740
 MXAG_u_f         1
 EPAG_u_f         1
/

*740/af used = 326/af ($1/1000 gal) applied

PARAMETER MIN_use_p(use)  min reqd need = 0.1 af\hh\year = 22 gal\person\day: MWD LA uses 80gal\p\day

* Change this parameter from 0.1 to 0.0 for both MI nodes to simulate removal of min-use-2 tiered pricing program

/SLV_u_f         0.0
 ABQMI_u_f       0.0
 MRGCD_u_f       0.0
 EBID_u_f        0.0
 EPMI_u_f        0.0
 MXAG_u_f        0.0
 EPAG_u_f        0.0
/

* 0.1 af/hh/hr = 22 gallons/person/day

*****************************************************************************************
* NEXT ARE HEADWATER INFLOWS, OTHER FLOWS, FLOW RELATIONSHIPS, AND                      *
* RESERVOIR STARTING VOLUMES, SIMPLE ECONOMIC VALUES PER AC FT WATER USE                *
*****************************************************************************************

* all water flows  are measured in 1000s acre feet per yer
* all water stocks are measured in 1000s acre feet instantaneous volume

*$ontext

TABLE headflows(t, inflow)  annual (hist or forecast lr ave) basin inflows -- snowpack or rain

*     Rio G DNG  Conej R     CBasProj  Sangr De Chr  SJChamaIBT   Rio Chma   Jemez R   Rio Puerc  Rio Sal
      RG-DN_h_f Conejos_h_f  CBasn_h_f  SangDC_h_f   SJCham_h_f  Chama_h_f Jemez_h_f Puerco_h_f Salado_h_f
*-----------------------------------------------------------------------------------------------------------
2006   659800     345760      17681      227228       106580     439000     45170     32238      40515
2007   659800     345760      17681      227228       106580     439000     45170     32238      40515
2008   659800     345760      17681      227228       106580     439000     45170     32238      40515
2009   659800     345760      17681      227228       106580     439000     45170     32238      40515
2010   659800     345760      17681      227228       106580     439000     45170     32238      40515
2011   659800     345760      17681      227228       106580     439000     45170     32238      40515
2012   659800     345760      17681      227228       106580     439000     45170     32238      40515
2013   659800     345760      17681      227228       106580     439000     45170     32238      40515
2014   659800     345760      17681      227228       106580     439000     45170     32238      40515
2015   659800     345760      17681      227228       106580     439000     45170     32238      40515
2016   659800     345760      17681      227228       106580     439000     45170     32238      40515
2017   659800     345760      17681      227228       106580     439000     45170     32238      40515
2018   659800     345760      17681      227228       106580     439000     45170     32238      40515
2019   659800     345760      17681      227228       106580     439000     45170     32238      40515
2020   659800     345760      17681      227228       106580     439000     45170     32238      40515
```

```
2021   659800     345760      17681      227228       106580     439000     45170     32238      40515
2022   659800     345760      17681      227228       106580     439000     45170     32238      40515
2023   659800     345760      17681      227228       106580     439000     45170     32238      40515
2024   659800     345760      17681      227228       106580     439000     45170     32238      40515
2025   659800     345760      17681      227228       106580     439000     45170     32238      40515
;

*$offtext

PARAMETER

sourc(inflow,t)  source headwater flows swaps rows and columns (messy)
source(inflow,t) source flows with drought reductions;

sourc(inflow,t)  = (1/1000) * SUM(tp $ (ORD(t) EQ (ORD(tp))), headflows(t,inflow));

* Change this parameter from 1 to 0.75 to 0.50 to simulate impacts of drought;

source(inflow,t) = 1.00 * sourc(inflow,t);  > 100% of normal inflows;

* display sourc, source;

PARAMETERS discount(t)      annual discount factor
;

discount(t) = (1/(1)) ** ord(t);
*display discount;

SCALAR

Mx_treaty       US Mexico 1906 Treaty delivery reqts at US-Mexico line

/60/

epsilon         small number to avoid dividing by zero
/.0001/

PARAMETERS

B0ar(res)   A-CAPAC intercept: Intcpt for reservoir area as linear fn of volume = 0
B1ar(res)   A-CAPAC slope:     (1st order) Slope for res area = linear fn of vol = d(area)\d(vol)
;

B0ar('HE_res_s') =     0;          > Heron Intercept        max vol  =  0.4 kk af
B1ar('HE_res_s') =    .01475;      > Heron slope            max area =  5.9 k ac

B0ar('EV_res_s') =     0;          > El Vado Intercept      max vol  =  0.186 kk af
B1ar('EV_res_s') =    .01720;      > El Vade slope          max area =  3.200 k ac

B0ar('AB_res_s') =     0;          > Abiquiu Intercept      max vol  =  1.535 kk af
B1ar('AB_res_s') =    .010119;     > Abiquiu slope          max ares = 15.536 k ac

B0ar('CO_res_s') =     0;          > Cochiti intercept      max vol  =  0.719 kk af
B1ar('CO_res_s') =    .014798;     > Cochiti slope          max area = 10.636 k ac

B0ar('EB_res_s') =     0;          > Elephant Butte inter   max vol  =  2.065 kk af
```

```
B1ar('EB_res_s') =    .01743;      > Elephant Butte slope   max area = 36.000 k ac

B0ar('CA_res_s') =     0;          > Caballo intercept      max vol  =  0.331 kk af
B1ar('CA_res_s') =    .03474;      > Caballo slope          max area = 11.500 k ac



parameter B0da(aqf)  starting aquifer depth in feet below surface

/SLV_aqf_s        200
 ABQMI_aqf_s      200
 MRGCD_aqf_s      200
 EBID_aqf_s       200
 MXAG_aqf_s       200
 EPMI_aqf_s       200
 EPAG_aqf_s       200
/

parameter B1da(aqf)  Slope (<0) for aquifer depth as linear fn of volume = d(feet) \ d(vol)

/SLV_aqf_s        -.002
 ABQMI_aqf_s      -.01
 MRGCD_aqf_s      -.01
 EBID_aqf_s       -.01
 MXAG_aqf_s       -.01
 EPMI_aqf_s       -.01
 EPAG_aqf_s       -.01
/

PARAMETERS

z0(u)                  initial reservoir and aquifer volumes at stock nodes
ZMAX(u)                maximum reservoir aquifer capacity cannot be violated in any period
;

z0('HE_res_s' )       = 109.5;    > Heron starting value December 2005 (1000s af)
z0('EV_res_s' )       =  25.4;    > El Vado
z0('AB_res_s' )       = 111.7;    > Abiquiu
z0('CO_res_s' )       =  49.0;    > Cochiti
z0('EB_res_s' )       = 191.9;    > Elephant Butte
z0('CA_res_s' )       =  23.3;    > Caballo

z0('SLV_aqf_s' )      = 10000;    > San Luis Valley aquifer starting volume (1000s af)
z0('ABQMI_aqf_s')     = 10000;    > Albuquerque aquifer st volume
z0('MRGCD_aqf_s')     = 10000;    > MRGCD aquifer starting volume
z0('EBID_aqf_s')      = 10000;    > EBID aquifer starting volume
z0('MXAG_aqf_s')      = 10000;    > MX ag aquifer starting volume
z0('EPMI_aqf_s')      = 10000;    > EP M&I aquifer starting volume
z0('EPAG_aqf_s')      = 10000;    > TX Ag aquifer starting volume

ZMAX('HE_res_s')      =   400;    > Heron capacity
ZMAX('EV_res_s')      =   186;    > El Vado capacity
ZMAX('AB_res_s')      =  1535;    > Abiquiu capacity
ZMAX('CO_res_s')      =   719;    > Cochiti capacity
ZMAX('EB_res_s')      =  2065;    > EButte Reservoir capacity (1000s af)
ZMAX('CA_res_s')      =   331;    > Caballo capacity
```

```
ZMAX('SLV_aqf_s')     = 60000;    > upper bound on SLV valley aquifer volume
ZMAX('ABQMI_aqf_s')   = 20000;    > ALBQ
ZMAX('MRGCD_aqf_s')   = 20000;    > upper bound on MGRCD area aquifer volume
ZMAX('EBID_aqf_s')    = 20000;    > upper bound on EBID aquifer  (Mesilla) volume
ZMAX('MXAG_aqf_s')    = 20000;    > MXAG
ZMAX('EPMI_aqf_s')    = 20000;    > EPMI
ZMAX('EPAG_aqf_s')    = 20000;    > upper bound on TX ag aquifer(Hueco)  volume

*z0(res) = zmax(res);              > starts every reservoirs at full levels

**************** Section 3 **************************************************************
*  These endogenous (unknown) variables are defined                                     *
*  Their numerical values are not known til GAMS finds optimal soln                     *
*****************************************************************************************

FREE VARIABLES

*Hydro

X(i,t)                 Flow:  diversion-use-return flow-seepage - etc in 1000s acre feet pr yr
Z(u,t)                 Stock: volume reservoirs and aquifers in 1000s of acre feet
Za(res,t)              Stock: area by reservoir in 1000s acres
Zd(aqf,t)              Stock: depth by aquifer in feet

* Econ all in US $ 1000s pr yr

Ben_u_v(use,t)         Flow: gross use related economic benfits by node and time
Ben_e_v(res,t)         Flow: gross reservoir recreation benefits by node and time

Cost_ep_v(apply,t)       Flow: pump energy + capital use cost by node time
Cost_tp_v(apply,t)       Flow: pump treatment use cost by node and time
Cost_ed_v(apply,t)       Flow: surface energy + capital use cost by node and time
Cost_td_v(apply,t)       Flow: surface treatment use cost by node and time

Cost_ap_v(pump,t)      Flow: gw depth-dependent pump cost per foot depth per ac pumped
Cost_afp_v(pump,t)     Flow: gw depth-dependent total pump costs
Cost_dp_v(apply,t)     Flow: gw depth-dependent total use costs due to depth dependent pump costs

Cost_a_v(apply,t)      Flow: total gross apply costs by node and time
Cost_u_v(use,t)        Flow: total gross use costs by node and time
Cost_e_v(res,t)        Flow: total gross env costs by node and time

Ave_u_cost_v(use,t)    Flow: ave cost per af used by node and time - used for equity analysis
Ave_e_cost_v(res,t)    Flow: ave cost per af stored by node and time - used for equity analysis

N_use_ben_v(use,t)     Flow: net use-related economic benefits by node and time
N_env_ben_v(res,t)     Flow: net environmental benefit by node and time

NBut_v(t)              Flow: net benefits over nodes by time
NBet_v(t)              Flow: net environmental benefits over nodes by time

NBu_v                  Flow: net use benefits over nodes and time
NBe_v                  Flow: discounted net env benefits over nodes and time
NB_v                   Flow: PV net benefits over nodes and time (objective)
```

```
M_u_ben_v(use,t)        Flow: marginal benefits of use by node and time
M_e_ben_v(res,t)        Flow: marginal env benefits by node and time

Price_u_v(use,t)        Flow: price of use by  node and time
Price_e_v(res,t)        Flow: price of env use by node and time

M_cost_e_v(res,t)       Flow: marginal env mgmt cost of added volume by res node and time

M_n_ben_e_v(res,t)      Flow: marginal net env benefit by res node and time = mc - mb

* Equity pricing variables

Totrev_v(use,t)         Flow: total revenue by node and time
Totrev_n_v(use,t)       Flow: total revenue for basic needs' uses by node and time
Netrev_v(use,t)         Flow: Cost recovery -- total net revenue (profit) by node and time
Pct_rec_cost_v(use,t)   Flow: percent of costs recovered through water prices charged
Equity_v(use,t)         Flow: equity -- measured as AV Cost minus price x min required use

* Hydro check in 1000s acre feet per year

USE_v                   Total water used (ET) -- summed use over use nodes
USEc_v                  Total calc water use (ET) tot hw inflows minus tot outflows
;

**************** Section 4 ***************************************************************
*  The following equations state relationships among a basin's                           *
*  hydrology, institutions, and economics                                                 *
*******************************************************************************************

EQUATIONS

*******************************************************************************************
* EQUATIONS NAMED
*******************************************************************************************


*******************************************************************************************
* Hydrology Block
*******************************************************************************************
Inflows(inflow,t)   Flows: set streamflows from each source node
Rivers(i,t)         Flows: set hydrologic mass balance by flow node: sources = uses
Divs(divert,t)      Flows: set diversion levels by flow node: diversions < flows
Evaps(evap,t)       Flows: set reservoir  evaporation losses by flow node
Applies(apply,t)    Flows: set application levels by flow node: application = divert + pump
Uses(use,t)         Flows: set use levels by flow node: use = proportion of application
Seeps(seep,t)       Flows: set seepage levels by flow node: seep = proportion of application
Returns(return,t)   Flows: set return flow levels by flow node: rf = prop of application
Netseeps(netseep,t) Flows: defines net seepage levels by flow node: ns = seepage - pumping

gwflows(gwflow,t)   Flows: impact on river   flo by flow node from net seepage
Aqflows(aqflow,t)   Flows: impact on aquifer vol by flow node from net seepage

reservoirs(res,t)   Stock: reservoir accntg by stock node: reserv vol falls iff net release >0
aquifers(aqf, t)    Stock: aquifers at the aquifer nodes: aquifer vol rises iff net seepage >0
area(res,t)         Stock: area by reservoir in 1000s acres = function of volume in 1000 a-f
depth(aqf,t)        Stock: depth by aquifer in feet = function of volume in acre feet
```

```
*$ontext
*****************************************************************************************
* Institutions Block: rules constrain water allocations
*****************************************************************************************

US_MX(t)             (International) US: Mexico Treaty of 1906 requires 60K US deliveries to MX

Federal(t)           (US Federal)   End Species Act requires 50-100 cfs year around at SA Gauge.

CO_NM(t)             (Interstate) Rio Grande Compact Delivery Obligation (CO to NM) at Lobatos gauge
NM_TX(t)             (Interstate) Rio Grande Compact Delivery Obligation (NM to TX) at Elephant Butte

D2_NM(divert,t)      (Intrastate) DII Rule NM - up to 57% of releases at Caballo go to NM Lands
D2_TX(divert,t)      (Intrastate) DII Rule TX - up to 43% of releases at Caballo to to TX lands

*$offtext

*********************************************************************************************************
* Economics Block
*********************************************************************************************************

Ben_u_e(use,t)      Flow: gross use-related    economic   benefits by  use node and time
Ben_e_e(res,t)      Flow: gross volume-related recreation benefits by reservoir and time

Cost_ep_e(apply,t)    Flow: pumping energy + capital cost by  node and time
Cost_tp_e(apply,t)    Flow: pumping treatment        cost by  node and time
Cost_ed_e(apply,t)    Flow: surface energy + capital cost by  node and time
Cost_td_e(apply,t)    Flow: surface treatment        cost by  node and time

*                   start depth-dependent pump costs
Cost_ap_e(pump,t)   Flow: pumping depth cost per acre foot pumped per unit depth
Cost_afp_e(pump,t)  Flow: pumping dept cost from all acre feet pumped
Cost_dp_e(apply,t)  Flow: apply costs based on that part of applications coming from depth dependent pumping
*                   finish depth-dependent pump costs

Cost_a_e(apply,t)   flow: total operating cost by apply nodes by time
Cost_u_e(use,t)     Flow: total operating costs by use node and time
Cost_e_e(res,t)     Flow: total environmental cost by node and time

Ave_u_cost_e(use,t) Flow: average operating costs by node and time - used for equity analysis
Ave_e_cost_e(res,t) Flow: average environmental costs by node and time - used for equaity analysis

N_use_ben_e(use,t)  Flow: total net use benefits (use benefits minus op costs) by node and time
N_env_ben_e(res,t)  Flow: total net environmental benefit by node and time

NBut_e(t)           Flow: net benefits of use and environment summed over nodes by time
NBet_e(t)           Flow: net benefits of environment summed over nodes by time

NBu_e               Flow: discounted net benefits of use over nodes and time
NBe_e               Flow: discounted net environmental benefit over nodes and time
NB_e                Flow: discounted net use and environmental benefits over nodes and time (obj fn)

*********************************************************************************************************
* Policy Analysis Block
```

```
*****************************************************************************************************

Price_u_e(use,t)      Flow:  Price of use by node and time = mb
Price_e_e(res,t)      Flow:  price of env volume by node and time

Totrev_e(use,t)       Flow: total gross revenue by node and time
Netrev_e(use,t)       Flow: total net revenue (profit)
Totrev_n_e(use,t)     Flow: total net revenue for needs' uses

Cost_recov_e(mu,t)    Flow: total revenues => total costs
Pct_rec_cost_e(use,t) Flow: percentage of total costs recovered in revenues
Equity_e(use,t)       Flow: Equity measured as ac minus price x reqd use

*****************************************************************************************************
* Post Optimality Block
*****************************************************************************************************
* marginals used to cross check

M_use_ben_e(use,t)      Flow: marginal benefits of use by node and time
M_env_ben_e(res,t)      Flow: marginal environmental benefit by node and time

Marg_cos_e_e(res,t)     Flow: marginal env cost of managing for more volume by node and time
Marg_n_e_ben_e(res,t)   Flow: marginal net env     benefit by reservoir node and time (meb - mec)

* Accounting Block
Balance                 Flow: total ET water use basinwide -- summed ET over use nodes
Balance_calc            Flow: total inferred use basinwide -- sum inflows - sum resid riv flow
;

******************************************************************************************************
* End of named equations
******************************************************************************************************

**************************************************************************************
* Equations defined algebraically using names from above
**************************************************************************************

**************************************************************************************
* HYDROLOGY BLOCK:  inflows, diversions, use, outflows, reservoir and aquifer levels
**************************************************************************************

Inflows(inflow,t)..   X(inflow,t) =E= source(inflow,t);

Rivers(river,t)..     X(river,t)  =E= sum(inflow,    Bv(inflow, river)  * X(inflow, t)) +
                                      sum(riverp,    Bv(riverp, river)  * X(riverp, t)) +
                                      sum(divert,    Bv(divert, river)  * X(divert, t)) +
                                      sum(return,    Bv(return, river)  * X(return, t)) +
                                      sum(gwflow,    Bv(gwflow, river)  * X(gwflow, t)) +
                                      sum(release,   Bv(release,river)  * X(release,t)) ;

Divs(divert,t)..      X(divert,t) =L= sum(inflow,    Bd(inflow, divert) * X(inflow, t)) +
                                      sum(river,     Bd(river,  divert) * X(river,  t)) +
                                      sum(divertp,   Bd(divertp,divert) * X(divertp,t)) +
                                      sum(return,    Bd(return, divert) * X(return, t)) +
                                      sum(gwflow,    Bd(gwflow, divert) * X(gwflow, t)) +
                                      sum(release,   Bd(release,divert) * X(release,t)) ;
```

```
Evaps(evap,t)..       X(evap,t)   =E= sum(res,       Be(evap,      res) * Za(res,   t)) ;

Applies(apply,t)..    X(apply,t)  =E= sum(divert,    Ba(divert,  apply) * X(divert, t)) +
                                      sum(pump,      Ba(pump,    apply) * X(pump,   t)) ;

Uses(use,t)..         X(use,t)    =E= sum(apply,     Bu(apply,     use) * X(apply,  t)) ;
Seeps(seep,t)..       X(seep,t)   =E= sum(apply,     Bs(apply,    seep) * X(apply,  t)) ;
Returns(return,t)..   X(return,t) =E= sum(apply,     Br(apply,  return) * X(apply,  t)) ;

Netseeps(netseep,t).. X(netseep,t)=E= sum(seep,      Bn(seep,  netseep) * X(seep,   t)) +
                                      sum(pump,      Bn(pump,  netseep) * X(pump,   t)) ;

Gwflows(gwflow,t)..   X(gwflow,t) =E= sum(netseep,   Bg(netseep, gwflow)* X(netseep,t));
Aqflows(aqflow,t)..   X(aqflow,t) =E= sum(netseep,   Bq(netseep, aqflow)* X(netseep,t));

reservoirs(res,t)..   Z(res,t)    =E= z0(res)$(ORD(t) EQ 1) + Z(res,t-1)
                                   +  sum(release,   BLr(release,   res)* X(release, t))
                                   +  sum(evap,      Ber(evap,      res)* X(evap,    t));

aquifers(aqf, t)..    Z(aqf,t)    =E= z0(aqf)$(ORD(t) EQ 1) + Z(aqf,t-1)
                                   +  sum(aqflow,    BQv(aqflow,    aqf)* X(aqflow,  t));

* Area of reservoirs and depth of aquifers depend on respective volumes

area(res,t)..         Za(res,t)   =E= B0ar(res)  +   B1ar(res)          * Z(res,      t);
depth(aqf,t)..        Zd(aqf,t)   =E= B0da(aqf)  +   B1da(aqf)          * Z(aqf,      t);

*********************************************************************************************
* INSTITUTIONS BLOCK:  Laws, compacts, treaties: rules that constrain or define use patterns
*********************************************************************************************
*$ontext

*  (International) US Mexico Treaty of 1906

*  Treaty text on web:   http://www.ibwc.state.gov/Files/1906Conv.pdf
*  Provides for 60,000 Acre Feet per year of Rio Grande delivered by US to MX at US MX Border

US_MX(t)..   X('MXAG_d_f',t) =g= Mx_treaty;

***********************************************************************************************

*  (US Federal)  Endangered Species Act of 1973.  Silvery Minnow needs 50 cfs year around
*  Under current operation this is about 240,000 acre feet per year at the San Acacia Gauge

Federal(t).. X('Acacia_v_f',t) =G= 240;

***********************************************************************************************

*  (Interstate) Rio Grande Compact of 1938 among CO, NM, AND TX

*  Compact text on web:  http://wrri.nmsu.edu/wrdis/compacts/Rio-Grande-Compact.pdf

*  Colorado's obligation to New Mexico under the Compact, as stated in the delivery schedules,
*  is described by a table.  Here, it is approximated by a quadratic fn defining the obligation
*  based the Rio Grande and Conejos supply indices, respectively.
```

```
*  NM's obligations to TX exclude San Juan Chama interbasin flows from the Otowi Supply Index.
*  SJC flows are exclusively for NM. Overdelivery is possible.

*  Coeffs below are based on OLS regression on flows and flows squared without intercept.
*  Adjusted R-2s are considerably over 0.99.  OSL's of t-stats are all <<.001

CO_NM(t)..  X('Lobatos_v_f',t) =G= -10
                                          + 0.27158 * Source('Conejos_h_f',t)
                                      + 0.00053735  * Source('Conejos_h_f',t) * Source('Conejos_h_f',t)

                                       +    0.11395 * Source('RG-DN_h_f',t)
                                       + 0.00032016 * Source('RG-DN_h_f',t)   * Source('RG-DN_h_f',t);

*  New Mexico's obligation to water users below Elephant Butte Reservoir is summarized by a table,
*  approximated in this model by a mathematical quadratic function defining required flows delivered
*  to Elephant Butte Reservoir (TX) by NM, based on flows at the Otowi supply index.  Overdelivery possible.

NM_TX(t)..  X('EB_g_v_f',t) - (X('EB_rel_f',t) + X('EB_evp_f',t)) =G=
       0.56025 * (X('Otowi_v_f',t) - Source('SJCham_h_f',t))
  + 0.00010839 * (X('Otowi_v_f',t) - Source('SJCham_h_f',t)) * (X('Otowi_v_f',t) - Source('SJCham_h_f',t));
****************************************************************************************************

* (Intrastate)  DII Rule:  57% max RG Project deliveries to US lands to NM Ag, 43% max to EP MI + EP AG

D2_NM(divert,t)..  X('EBID_d_f', t)                   =L= 0.57 * (X('CA_g_v_f',t) - X('MXAG_d_f',t));
D2_TX(divert,t)..  X('EPMI_d_f', t) + X('EPAG_d_f',t) =L= 0.43 * (X('CA_g_v_f',t) - X('MXAG_d_f',t));
****************************************************************************************************

*$offtext
************************************************************************************************************
* ECONOMICS BLOCK.  Everything in this block measured in monetary units
************************************************************************************************************

* Benefits by node and period

Ben_u_e(use,t)..   Ben_u_v(use, t)   =E=
                                  scale(use,t)  * Ben_u_p(use, 'intercept')                        +
                                                  Ben_u_p(use, 'linear')    * X(use,t)             +
                               (1/scale(use,t)) * Ben_u_p(use, 'quadratic') * X(use,t) * X(use,t);

Ben_e_e(res,t)..   Ben_e_v(res,t)    =E=
                                  scalee(res,t) * Ben_e_p(res, 'intercept')                      +
                                                  Ben_e_p(res, 'linear')    * Z(res,t)           +
                              (1/scalee(res,t)) * Ben_e_p(res, 'quadratic') * Z(res,t) * Z(res,t);

* costs by node and period

Cost_ep_e(apply,t)..    Cost_ep_v(apply,t) =E=  sum(pump,   Cost_u_ep_p(pump,    apply)  * X(pump,     t));
Cost_tp_e(apply,t)..    Cost_tp_v(apply,t) =E=  sum(pump,   Cost_u_tp_p(pump,    apply)  * X(pump,     t));
Cost_ed_e(apply,t)..    Cost_ed_v(apply,t) =E=  sum(divert, Cost_u_ed_p(divert,  apply)  * X(divert,   t));
Cost_td_e(apply,t)..    Cost_td_v(apply,t) =E=  sum(divert, Cost_u_td_p(divert,  apply)  * X(divert,   t));

* pump costs.  Complicated.  Increased depth increases pump costs, which reduces their use
* cost per added foot depth per acre foot pumped if pumping occurs
Cost_ap_e(pump,t)..   Cost_ap_v(pump,t)=E=  sum(aqf,    Cost_app_p(aqf,  pump)  * Zd(aqf,        t));
```

```
* cost from all acre feet pumped based on actual depth and actual pumping from that depth
Cost_afp_e(pump,t)..  Cost_afp_v(pump,t) =E=            Cost_ap_v(pump,t)       * X(pump,t);

* cost from total water use met by depth-dependent pumping
Cost_dp_e(apply,t)..  Cost_dp_v(apply, t) =E=  sum(pump,  ID_pa(pump,    apply)     * Cost_afp_v(pump,t));
* End of pump costs.


* total costs of water uses from all sources

Cost_a_e(apply,t)..       Cost_a_v(apply,t)    =E=  Cost_ep_v(apply,t) + Cost_tp_v(apply,t) +
                                                    Cost_ed_v(apply,t) + Cost_td_v(apply,t) +
                                                    Cost_dp_v(apply,t)                      ;

Cost_u_e(use,t)..         Cost_u_v(use,t)      =e=  sum(apply, ID_au(apply, use) * Cost_a_v(apply,t));

Cost_e_e(res,t)..         Cost_e_v(res,t)      =E=  Env_cost_p(res) * Z(res,t);

* average costs of water users from all sources - used to analyze equity

Ave_u_cost_e(use,t)..     Ave_u_cost_v(use,t)  =E= Cost_u_v(use,t) / (X(use,t) + epsilon);
Ave_e_cost_e(res,t)..     Ave_e_cost_v(res,t)  =E= Cost_e_v(res,t) / (Z(res,t) + epsilon);

*******************************************************************************************

* Benefits:  Both use and environmental by node and period

N_use_ben_e(use,t)..    N_use_ben_v(  use, t) =E=  Ben_u_v(use, t) - Cost_u_v(    use,t);
N_env_ben_e(res,t)..    N_env_ben_v(res,t)    =E=  Ben_e_v(res,t)  - Cost_e_v(res,t);

* Net economic benefits by time pd over nodes both for use and environmental water purposes by time

NBut_e(t)..             NBut_v(t)             =E=    sum(use, N_use_ben_v(   use,t));
NBet_e(t)..             NBet_v(t)             =E=    sum(res, N_env_ben_v(res,t));

* Discounted net economic and environmental benefits over time

NBu_e..                 NBu_v                 =E=    sum(t, discount(t) * NBut_v(t));
NBe_e..                 NBe_v                 =E=    sum(t, discount(t) * NBet_v(t));

* Objective fn -- sums production and environmental benefits

NB_e..                  NB_v                  =E=    NBu_v + NBe_v;

********************************************************************************************
* POLICY BLOCK:  Various policy approaches to water pricing and allocation
********************************************************************************************

* Revenue and costs

* total revenue from poor houeholds for needs' uses

Totrev_n_e(use,t)..       Totrev_n_v(use,t)    =E=  MAX_PRICE_p(use) * MIN_use_p(use) * Scale(use,t);

Totrev_e(use,t)..         Totrev_v(use,t)      =E=  Price_u_v(use,t) * (X(use,t) - (MIN_use_p(use) * Scale(use,t)))
                                                +   Totrev_n_v(use,t);
```

```
Netrev_e(use,t)..         Netrev_v(use,t)      =E=  Totrev_v(use,t) - Cost_u_v(use,t);

* Cost recovery criterion

Cost_recov_e(mu,t)..      Netrev_v(mu,t)       =G=  0;

Pct_rec_cost_e(use,t)..   Pct_rec_cost_v(use,t) =E=  Totrev_v(use,t) / (Cost_u_v(use,t)+epsilon);

* Equity criterion

Equity_e(use,t)..         Equity_v(use,t)       =E= (Ave_u_cost_v(use,t) - MAX_Price_p(use)) * MIN_use_p(use);

******************* end of policy block ***************************************************


***************************************************************************************
* POST OPTIMALITY BLOCK.  Nothing from here down used to find optimal solution
***************************************************************************************
* Helps find the points where the equimarginal rule holds at both use and env nodes:
*   Use:  MB = M Operations Cost + M Env Cost + M Res cost
*   Env:  MB = M Operations Cost + M Env Cost + M Res Cost

* Marginal use benefits by use and time and by other subsets with time

M_use_ben_e(use,t)..  M_u_ben_v(use,t)    =E=          Ben_u_p(  use,'linear'   )
                              + 2 * (1/scale(use,t)) * Ben_u_p(  use,'quadratic')  * X(use,t);

* Marginal env benefits by reservoir and time

M_env_ben_e(res,t).. M_e_ben_v(res,t)     =E=          Ben_e_p(res, 'linear')
                             +  2 *(1/scalee(res,t)) * Ben_e_p(res, 'quadratic') * Z(res,t);

Price_u_e(use,t)..   Price_u_v(use,t)     =E=          M_u_ben_v(use,t);
Price_e_e(res,t)..   Price_e_v(res,t)     =E=          M_e_ben_v(res,t);

******************************************************************************************
* insert here (carefully) marginal costs from pumping-aquifer-drawdowns.
*
*
******************************************************************************************
* marginal use cost is calculated AFTER optimal solution.  It requires using marginals (.m)
*
*
******************************************************************************************

* Marginal env (mgmt) costs by reservoir and time for volumes

Marg_cos_e_e(res,t)..     M_cost_e_v(res,t)     =E=   Env_cost_p(res);

* Marginal net env benefit at reservoir nodes

Marg_n_e_ben_e(res,t)..   M_n_ben_e_v(res,t)    =E=  M_e_ben_v(res,t) - M_cost_e_v(res,t);

****************************************************************************************
* ACCOUNTING BLOCK:  tracks water flows, but not used for solution
```

```
*****************************************************************************************
Balance..           USE_v      =E= sum(t, sum(use,    X(use,t   )));
Balance_calc..      USEc_v     =E= sum(t, sum(inflow, X(inflow,t))) - sum(t, X('Quitman_v_f',t));

*****************************************************************************************

**************************  END OF ALL EQUATIONS ****************************************

**************** SECTION 5 **************************************************************
*  The following section defines models.                                                *
*  Each model is defined by a set of equations used                                     *
*  for which one single variable is optimized (min or max)                              *
*****************************************************************************************

* This simple prototype model uses ALL equations defined above.  But larger models
* may exclude some equations. For example, each of several institution could be defined
* by one equation.  And each of several model might conduct a single policy experiment
* in which that model tries out a single institution.  This would require deleting all
* institutional equations except the one analyzed.
* If you need to EXclude some equations, list INcluded equations where ALL appears below

MODEL RIO_PROTOTYPE /ALL/;

**************** Section 6 **************************************************************
*  The following section defines all solves requested,
*  Each solve states a single model for which an optimum is requested.
*
*  Upper, lower and fixed bounds on certain variables can also be included here
*  Bounding variables here gives that variable a non-zero shadow price where the optimal
*  solution appears at that boundary.  If the bound doesn't cotrain the model
*  the variable's shadow price is zero (complementary slackness)
*****************************************************************************************

* Non-negative flows at nodes

X.lo(inflow,t) = 0;
X.lo(river,t)  = 0;
X.lo(divert,t) = 0;
X.lo(apply,t)  = 0;
X.lo(use,t)    = 0;
X.lo(pump,t)   = 0;
X.lo(seep,t)   = 0;
X.lo(return,t) = 0;

* Non-negative stocks at nodes

Z.lo(u,t)     = 0;
Za.lo(res,t)  = 0;
Zd.lo(aqf,t)  = 0;

* upper and lower bounds: legal, institutional, hydrologic, or economic

X.up('ABQMI_d_f',t) =   0;  >    0 surface treatment capacity Albuquerque M&I 2006
X.up('MRGCD_p_f',t) =   0;  >    0 pumping capacity MRGCD 2006
X.up('EPAG_p_f',t)  =   0;  >    0 pumping capacity EPaso irrigated ag 2006
X.up('EPMI_p_f',t)  = 220;  > 220K af/yr EPWU pump capacity from 74 Hueco Bolson wells
```

```
X.up('EPMI_d_f',t)  =  62;  >  62K af/yr EPWU surface treatment capacity

* Sustainability terminal condition -- each water stock (reservoir, aquifer) ends with
* terminal volume > critical level.

* Avoids depleting stocks in last period -- saves stocks for future generations

Z.lo(res, t)     = 0.10 * Z0(  res);   > each reservoir never < 10% of orig vol
Z.lo(res, tlast) = 0.50 * ZMAX(res);   > term pd ea reservoir vol > 0.50 * starting value
Z.lo(aqf, tlast) = 0.50 * Z0(  aqf);   > term pd ea aquifer   vol > 0.50 * starting value

Z.up(res, t)     = ZMAX(res);          > maximum reservoir capacity respected in ea period
Z.up(aqf, t)     = ZMAX(aqf);          > maximum aquifer   capacity respected in ea period

* model run #1

OPTION NLP = MINOS;
SOLVE RIO_PROTOTYPE USING NLP MAXIMIZING NB_v;

* model run #2

*growth('ABQMI_u_f') = 0.0240;
*growth('EPMI_u_f') =  0.0367;
*scale(use,t)  =  ((1+ growth(use))**(ord(t)-1)) * scales(use);
*x.fx(ad,t) = x.l(ad,t);
*x.up(mu,t) = x.l(mu,t);

*SOLVE RIO_PROTOTYPE USING NLP MAXIMIZING NB_v;

**************** Section 7 ***********************************************************
*  The following section displays post-optimality output                             *
**************************************************************************************

parameter

* table 0

* calculates marginal cost from use in several steps

basis_p(pump,  t)     identifies interior pumping nodes not at a bound
basis_d(divert,t)     identifies interior divert nodes not at a bound
apply_marg_a(apply,t) idenfities water applied at the margin not at a bound

M_cost_a_p(apply,t)   post opt: marginal cost from appl water taken from expandable source
M_cost_ut_p(use,t)    post opt: marginal cost from used water taken from expandable source
M_cost_u_p(use)       marginal cost from water use averaged over time

M_n_ben_t_u_p(use,t)  marginal net use benefit from water use away from bounds
M_n_ben_u_p(use)      marginal net use benefit averaged over all periods

* table 1
* water quantities and some benefits

tot_applied(apply)   water applied at apply nodes
tot_divert(divert)   water diverted at divert nodes
tot_pumped(pump)     water pumped at pump nodes
```

```
eco_ben(use)         benefits at use nodes
env_ben(res)         recreation benefits at reservoir nodes

eco_sup_cost(use)    economic costs of water supply
env_cost(    res)    environmental costs of env damages


* table 2
* equity pricing table

equit_v(use)         equity -- (ave cost - price) x (required water use = 0.1 af\hh\yr)
prop_rec_cos(use)    proportion of costs recovered thru water pricing

* table 3
* environmental benefits, marginal use benefits, and net benefits

Env_ben_p(res)       gross reservoir recreation benefits by node and time
Cost_p   (use)       total gross economic costs by node and time
Env_cost_pp(res)     total environmental cost by node and time
Netben_p (use)       net use-related economic benefits by node and time
N_env_ben_p(res)     net environmental benefit by node and time

M_u_ben_p(use)       marginal benefits of use by node and time
M_e_ben_p(res)       marginal env benefits by node and time
M_cost_e_p(res)      marginal mgmt cost of added volume by res node and time
M_n_ben_e_p(res)     marginal net env benefit by res node and time

Netben_p (use)       net use-related economic benefits by node and time
NBu_p                net benefits over nodes by time averaged by time
NBe_p                net benefits of environmental over nodes averaged by time
NB_p                 total net benefits use + environment over nodes averaged over time
;

* post optimality:  parameters defined with code based on optimal solution above

* table 0

* -------------------------------------------------------------------------------------
*  several steps required to find marginal cost of water from expandable sources     *
* -------------------------------------------------------------------------------------

* Marginal use cost defined below. It's based on marginals at optimal solutions
*
* It comes by recognizing that bounded solutions will not change marginally (Kuhn Tucker: compl slackness)
* Water use expansions will not come from sources already upper or lower bounded

* 4 steps are required:

* (1): identifies interior (unbounded) optimal pumping and diversions nodes
basis_p(pump,  t) = abs(X.m(pump,  t)) lt epsilon;  > un-bounded pumping
basis_d(divert,t) = abs(X.m(divert,t)) lt epsilon;  > un-bounded diversions

* (2): finds total quantities at apply nodes met by pumping v. diversion v. a mix of both
apply_marg_a(apply,t)  = sum(pump,   ID_ap(apply,   pump) * basis_p(pump,  t) * X.l(pump,  t)) +
                         sum(divert, ID_ad(apply, divert) * basis_d(divert,t) * X.l(divert,t));
```

```
* (3): finds marginal cost at apply nodes based on increases met by pumping v diversions v a mix of both
M_cost_a_p(apply,t) =
  sum(pump,   Cost_up_p(pump,   apply) * (X.l(pump,  t) * basis_p(pump,t  )) / (epsilon + apply_marg_a(apply,t)))  +
  sum(divert, Cost_ud_p(divert, apply) * (X.l(divert,t) * basis_d(divert,t)) / (epsilon + apply_marg_a(apply,t))) ;

* (4): finds marginal cost at use nodes based on water used consumptively per unit applied
M_cost_ut_p(use,t) = (1/Bu_p(use)) * sum(apply, ID_au(apply, use) * M_cost_a_p(apply,t));

* averages marginal cost over all years for display
M_cost_u_p(use) = sum(t, M_cost_ut_p(use,t))/card(t);

* defines marginal net use benefits as marginal use benefits - marginal use costs;
M_n_ben_t_u_p(use,t)   = M_u_ben_v.l(use,t) - M_cost_ut_p(use,t);

* averages marginal net use benefits over years for display
M_n_ben_u_p(use)       = sum(t, M_n_ben_t_u_p(use,t))/(card(t) + epsilon);

* -------------------------------------------------------------------------------------------------
* tables below are displays:  they're simple averages over time of optimal solutions
* -------------------------------------------------------------------------------------------------

* table 1

tot_applied(apply)    =   (epsilon + SUM(t, X.l(apply,t)))/card(t);
tot_divert(divert)    =   (epsilon + sum(t, X.l(divert,t)))/card(t);
tot_pumped(pump )     =   (epsilon + sum(t, X.l(pump,t)))/card(t);

eco_ben(use)          =   (epsilon + sum(t, Ben_u_v.l(use,t)))/card(t);
env_ben(res)          =   (epsilon + sum(t, Ben_e_v.l(res,t)))/card(t);

eco_sup_cost(use)     =   (epsilon + sum(t, Cost_u_v.l(use,t)))/card(t);
env_cost(    res)     =   (epsilon + sum(t, Cost_e_v.l(res,t)))/card(t);

* table 2

equit_v(use)          =   (epsilon + sum(t, equity_v.l(use,t)))/card(t);
prop_rec_cos(use)     =   (epsilon + sum(t, pct_rec_cost_v.l(use,t)))/card(t);

* TABLE 3

Netben_p (use)      = (epsilon + sum(t,N_use_ben_v.l(use,t)))/card(t) ;
N_env_ben_p(res)    = (epsilon + sum(t,N_env_ben_v.l(res,t)))/card(t);

NBu_p               = (epsilon + sum(t,NBut_v.l(t)))/card(t);
NBe_p               = (epsilon + sum(t,NBet_v.l(t)))/card(t);
NB_p                =  NBu_p + NBe_p;

M_u_ben_p(use)      = (epsilon + sum(t, M_u_ben_v.l(use,t)))/card(t);
M_e_ben_p(res)      = (epsilon + sum(t, M_e_ben_v.l(res,t)))/card(t);
M_cost_e_p(res)     = (epsilon + sum(t, M_cost_e_v.l(res,t)))/card(t);
M_n_ben_e_p(res)    = (epsilon + sum(t, M_n_ben_e_v.l(res,t)))/card(t);

display  M_cost_ut_p, M_n_ben_t_u_p;

DISPLAY tot_applied, tot_divert, tot_pumped
```

```
        eco_ben, env_ben,
        eco_sup_cost, env_cost;

DISPLAY equit_v, prop_rec_cos;

DISPLAY  Netben_p, N_env_ben_p
         NBu_p, NBe_p, NB_p
         M_u_ben_p,   M_e_ben_p,
         M_cost_u_p,  M_cost_e_p,
         M_n_ben_u_p, M_n_ben_e_p;


*****************************************************************************************
* THE END
*****************************************************************************************
```