



Fermilab

AP-Note-91-005

A
C
C
E
L
E
R
A
T
O
R

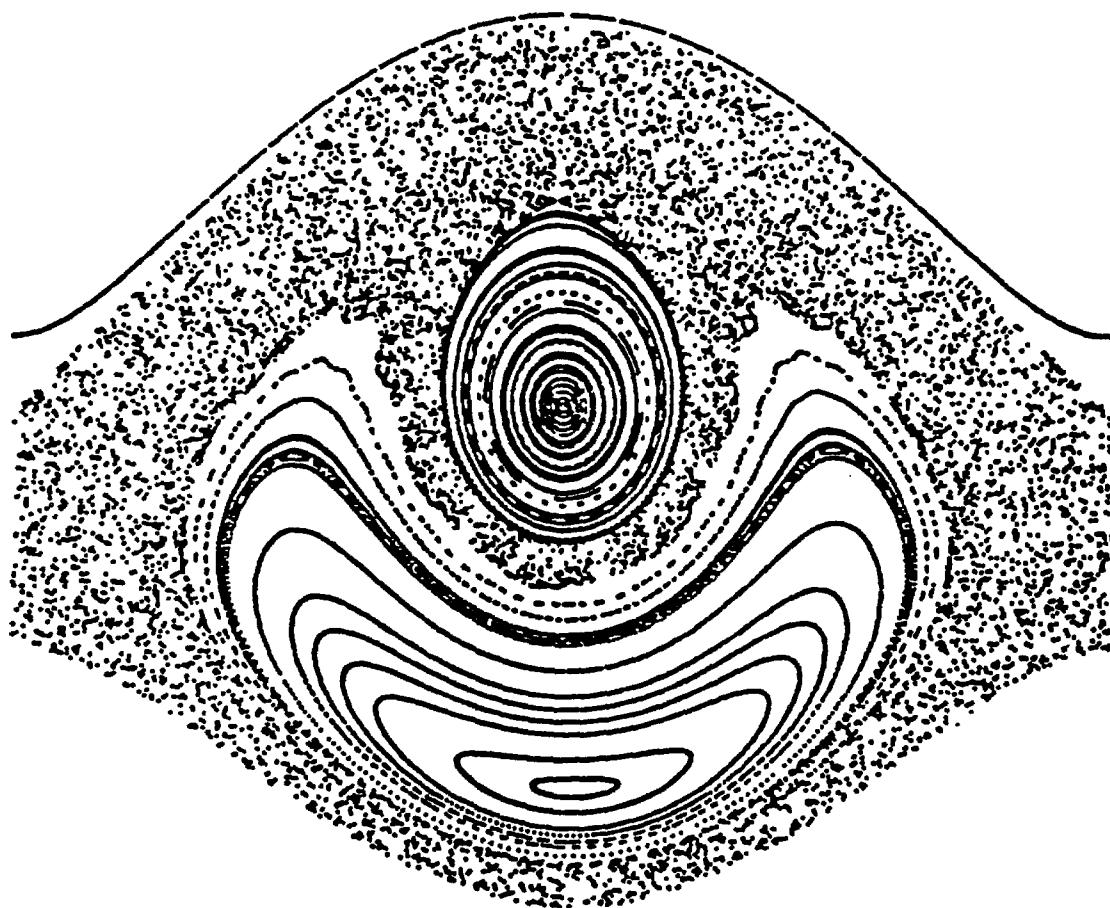
P
H
Y
S
I
C
S

D
E

TITLE: The Changes for the New DBSF and FLATIN Programs

AUTHOR: J. A. Holt

October 11, 1991



The changes for the new DBSF and FLATIN programs

J. A. Holt
Fermilab

October 11, 1991

This note is a brief description of the changes made for the new versions of the database, dbsf and flatin. The new routines have not been installed in a public area yet.

1 Database Changes

There have been a number of changes and additions to the database tables. A field named `engineering_type` has been added to the table `magnet_piece`. The table `ideal_magnet` has been renamed to `slot`. There have been fourteen new tables added to the database. They are:

1. `bend`
2. `rfcavity`
3. `multipole`
4. `closed_orbit_corrector`
5. `collimator`
6. `aperture`
7. `magnet_size`
8. `drift`
9. `quadrupole`
10. `sextupole`
11. `octupole`
12. `solencoid`
13. `monitor`
14. `elseparator`

The isql scripts except for renaming the one table are listed in appendix A.

```

int      has_strength;
int      has_tilt;
int      has_more;
};

struct row{
char     name[NAME_MAX];
int      cols;
char     heading[COLUMN_MAX][NAME_MAX];
};

struct more_struct {
    int      morend; /* not filled by flatten */
    double   attribute[COLUMN_MAX];
};

#endif

```

The structures 'legal_type' and 'row' are filled with fixed values:

```

#include "lattice_defs.h"
#include "lattice_struct.h"

struct legal_type legal_type[] = {
  "",           0,0,0,0, /* NULL element */
  "drift",      1,0,0,0, /* length */
  "hmonitor",   1,0,0,0, /* length */
  "vmonitor",   1,0,0,0, /* length */
  "monitor",    1,0,0,0, /* length */
  "instrument", 1,0,0,0, /* length */
  "wiggler",    1,0,1,0, /* length, tilt */
  "rbend",      1,1,1,1, /* length, angle, tilt, more */
  "sbend",      1,1,1,1, /* length, angle, tilt, more */
  "quadrupole", 1,1,1,0, /* length, k1, tilt */
  "sextupole",  1,1,1,0, /* length, k2, tilt */
  "octupole",   1,1,1,0, /* length, k3, tilt */
  "multipole",  1,0,0,1, /* length, more */
  "solenoid",   1,1,0,0, /* length, ks */
  "rfcavity",   1,0,0,1, /* length, more */
  "elseparator", 1,1,1,0, /* length, e, tilt */
  "srot",       0,1,0,0, /* angle */
  "yrot",       0,1,0,0, /* angle */
  "hkick",      1,1,0,0, /* length, hkick */
  "vkick",      1,1,0,0, /* length, vkick */
  "kicker",     1,0,1,1, /* length, tilt, more */
  "marker",     0,0,0,0,
  "ecollimator", 1,0,0,1, /* length, more */
  "rcollimator", 1,0,0,1, /* length, more */
  "tbend",      1,1,1,1, /* length, angle, tilt, more */
  "beamline",   1,0,0,0, /* length */

```

A sample program using the flatin is:

```
/* This demo demonstrates and tests flatin.c functions */

#include "flatin.h"

main(argc, argv)
    int argc; char **argv;
{
    flatin_data my_flatin_data;
    char *file_name;
    char latfile[NAME_MAX];

    if ( argc != 2 ) {
        printf("Usage: flatin_demo  flat_file_name\n");
        printf("flat_file_name is a beam line in 'flat' format\n");
        exit(1);
    }
    else {
        file_name = *(argv+1);

        strcpy(latfile,file_name);
    }

    if (flatin(latfile,&my_flatin_data)) {
        printf("found\n%-5d parameters,\n%-5d elements,",my_flatin_data.number_of_parameters,
        my_flatin_data.number_of_elements);
        printf("\n%-5d elements in the beamline\n", my_flatin_data.number_of_lattices);

    }

    exit(0);
}
```

```

create table bend
(name          char(20),
length        varchar(60)    null,
angle         varchar(60)    null,
tilt          varchar(10)    null,
quad_strength varchar(20)    null,
sxtip_strength varchar(20)   null,
octip_strength varchar(20)   null,
entrance_angle varchar(20)   null,
exit_angle    varchar(20)    null,
field_integral varchar(20)   null,
half_gap      varchar(20)    null,
entrance_curv varchar(20)    null,
exit_curv     varchar(20)    null)
create unique clustered index bend_index on bend(name)
go

create table rfcavity
(name          char(20),
length        varchar(60)    null,
voltage       varchar(20)    null,
phase_lag     varchar(20)    null,
harmonic_number varchar(20)   null,
rf_coupling   varchar(20)    null,
rf_power      varchar(20)    null,
shunt_imped   varchar(20)    null,
fill_time     varchar(20)    null)
create unique clustered index rfcavity_index on rfcavity(name)
go

create table multipole
(name  char(20),
K0L      varchar(20)    null,
K1L      varchar(20)    null,
K2L      varchar(20)    null,
K3L      varchar(20)    null,
K4L      varchar(20)    null,
K5L      varchar(20)    null,
K6L      varchar(20)    null,
K7L      varchar(20)    null,
K8L      varchar(20)    null,
K9L      varchar(20)    null,
T0       varchar(20)    null,
T1       varchar(20)    null,
T2       varchar(20)    null,
T3       varchar(20)    null,
T4       varchar(20)    null,
T5       varchar(20)    null,
T6       varchar(20)    null,
T7       varchar(20)    null,
T8       varchar(20)    null,
T9       varchar(20)    null)
create unique clustered index multipole_index on multipole(name)
go

create table closed_orbit_corrector
(name      char(20),
length    varchar(60)    null,
tilt      varchar(10)    null,
horz_angle varchar(20)   null,
vert_angle varchar(20)   null)
create unique clustered index closed_orbit_corrector_index
on closed orbit corrector(name)

```

```
create unique clustered index monitor_index on monitor(name)
go

create table elseparator
(name           char(20),
length        varchar(60)    null,
efield_strength  varchar(60)    null,
tilt          varchar(10)    null)
create unique clustered index elseparator_index on elseparator(name)
go

grant select on rfcavity to public
grant select on bend to public
grant select on closed_orbit_corrector to public
grant select on multipole to public
grant select on collimator to public
grant select on aperture to public
grant select on magnet_size to public
grant select on drift to public
grant select on quadrupole to public
grant select on sextupole to public
grant select on octupole to public
grant select on solenoid to public
grant select on monitor to public
grant select on elseparator to public
go
```