



Music
at **MARIST**

Band

Marist College Band

Database Designed by Cassandra Graves

- 03 Executive Summary
- 04 Entity-Relationship Diagram
- 05 Tables
 - 05 Participants Table
 - 06 Ensembles Table
 - 07 EboardPositions Table
 - 08 KKPsiMembers Table
 - 09 TBSMembers Table
 - 10 EboardMembers Table
 - 11 ConcertInsts Table
 - 12 PepInsts Table
 - 13 MarchingInsts Table
 - 14 Conductors Table
 - 15 Students Table
 - 16 Employees Table
 - 17 EnrolledIn Table
- 18 Views
 - 18 GreekLifeMembers
 - 19 PaidBoardMembers
- 20 Reports
 - 20 StudentInstrumentRoster
- 21 Stored Procedures
 - 21 EnsembleEnrolled
- 22 Security
 - 22 Admin Role
 - 22 Leader Role
- 23 Implementation Notes
- 24 Known Problems
- 25 Future Enhancements

Table of Contents

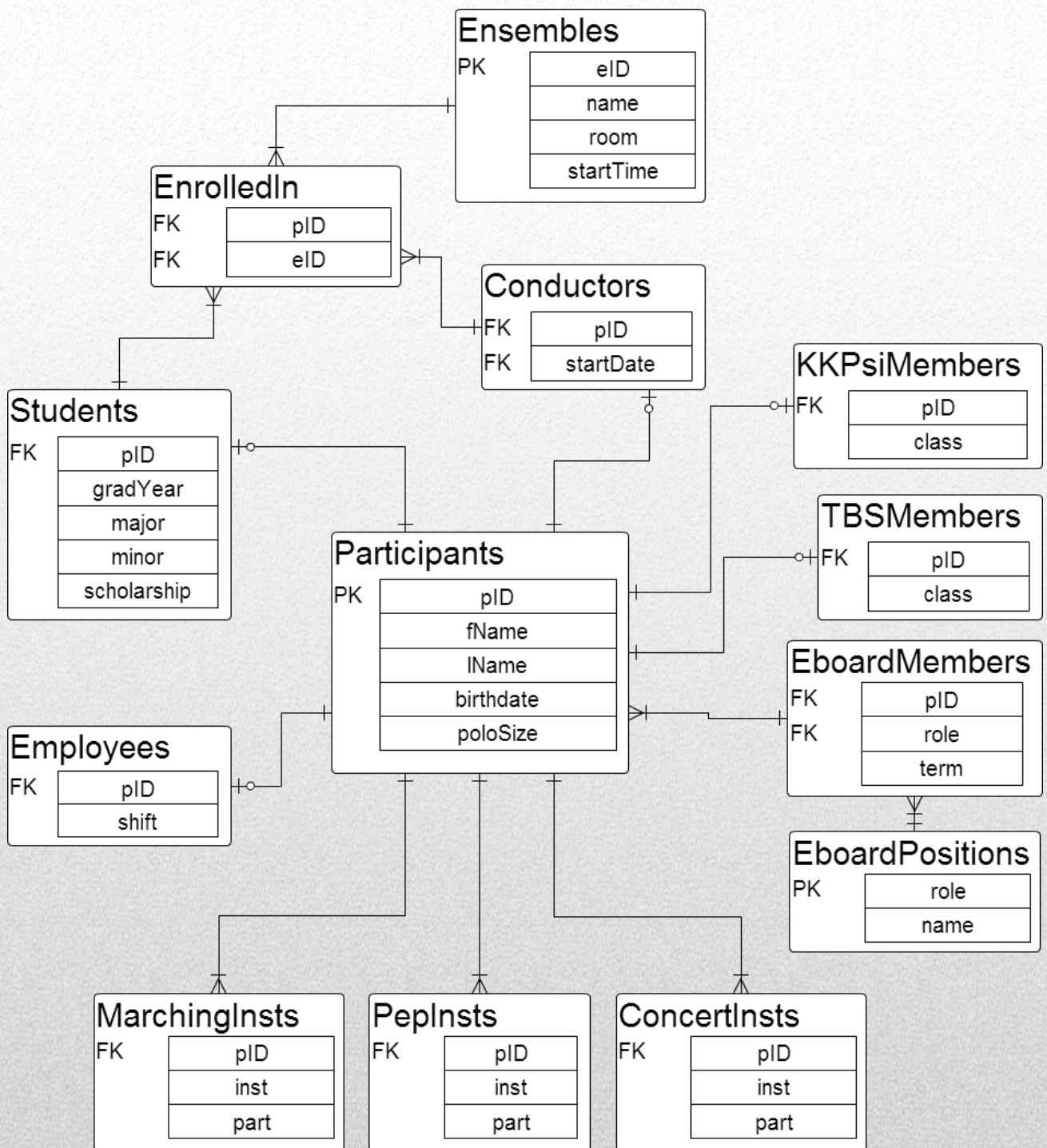
- Overview

The Marist College Band involves over 150 students in a number of ensembles. All members participate in the Symphonic Band, Marching Band and Pep Band. The other ensembles become available to join once a member of the large ensemble. With no Music Major available, all performers continue playing at Marist purely for the enjoyment and love of music. As one of the most well known organizations on campus, the Band has become ambassadors not only for the Music Department, but for all of Marist College!

- Objectives

The Marist Band has been growing tremendously over the last 28 years. To continually support the growing Band program, this database will help maintain details about the band's past, keep up to date with the band's present and pave the way for the band's future.

Executive Summary



Entity-Relationship Diagram

Participants Table

Anyone involved in the band program will later be organized into roles. This table defines the basic information about each person.

```
CREATE TABLE Participants(  
    pID TEXT NOT NULL,  
    fName TEXT NOT NULL,  
    lName TEXT NOT NULL,  
    birthdate DATE,  
    poloSize TEXT,  
    PRIMARY KEY(pID)  
);
```

Functional Dependencies

pID → fName, lName, birthdate, poloSize

Sample Data:

	pid text	fname text	lname text	birthdate date	polosize text
1	p001	Arthur	Himmelberger	1951-11-26	L
2	p002	Cassandra	Graves	1992-05-07	M
3	p003	Brian	Veltre	1992-03-01	L
4	p004	Michael	Napolitano	1982-08-08	XL
5	p005	Derek	Daffara	1994-12-02	XL
6	p006	Sara	Lacey	1992-04-02	L
7	p007	Erin	Gates	1995-03-21	M
8	p008	Elizabeth	Grisafi	1992-11-25	M

Tables

Ensembles Table

There are a variety of ensembles at Marist College for band members to become a part of once a full member of the band.

```
CREATE TABLE Ensembles(  
    eID TEXT NOT NULL,  
    name TEXT NOT NULL,  
    room TEXT,  
    startTime TIME,  
    PRIMARY KEY(eID)  
);
```

Functional Dependencies

eID → name, room, startTime

Sample Data:

	eID text	name text	room text	starttime time without time zone
1	e001	Symphonic Band	MU4001	19:45:00
2	e002	Wind Symphony	MU4001	21:30:00
3	e003	Woodwind Ensemble	MU3204	18:30:00
4	e004	Brass Ensemble	MU4001	18:30:00
5	e005	Handbell Choir	MU2220	17:00:00
6	e006	Percussion Ensemble	MU4001	17:00:00
7	e007	Guitar Ensemble	MU4001	18:30:00
8	e008	Orchestra	MU2220	18:30:00
9	e009	Jazz Foxes	MU4001	21:00:00
10	e010	Flute Choir	MU2220	18:30:00

Tables

EboardPositions Table

There are many tasks that need to get done on a week to week basis. An elected group of band members hold positions on this executive board and take care of anything that needs to be done.

```
CREATE TABLE EboardPositions(  
    role TEXT NOT NULL,  
    name TEXT NOT NULL,  
    PRIMARY KEY(role)  
);
```

Functional Dependencies

role → name

Sample Data:

	role text	name text
1	P	President
2	VPC	Vice President of Concert Bands
3	VPP	Vice President of Athletic Bands
4	S	Secretary
5	T	Treasurer
6	OL	Orchestra Liaison
7	W	Webmaster
8	A	Advisor

Tables

KKPsiMembers Table

Members of the band may be asked to join the Honorary Band Fraternity, Kappa Kappa Psi. One goal for this organization is to serve the band.

```
CREATE TABLE KKPsiMembers(  
    pID          TEXT NOT NULL REFERENCES Participants(pID),  
    class        TEXT  
);
```

Functional Dependencies

$pID \rightarrow class$

Sample Data:

	pid text	class text
1	p002	Rho
2	p003	Rho
3	p005	Upsilon
4	p004	Beta
5	p007	Chi
6	p008	Tau

Tables

TBSMembers Table

Members of the band may be asked to join the Honorary Band Sorority, Tau Beta Sigma. One goal for this organization is to serve the band.

```
CREATE TABLE TBSMembers(  
    pID      TEXT NOT NULL REFERENCES Participants(pID),  
    class    TEXT  
);
```

Functional Dependencies

pID → class

Sample Data:

	pid text	class text
1	p006	Nu

Tables

EboardMembers Table

The specific band members that are elected into executive board positions serve the band for the following academic year.

```
CREATE TABLE EboardMembers(  
    pID          TEXT NOT NULL REFERENCES Participants(pID),  
    role         TEXT NOT NULL REFERENCES EboardPositions(role),  
    term         TEXT  
);
```

Functional Dependencies

pID → role, term

Sample Data:

	pid text	role text	term text
1	p002	P	2013-2014
2	p003	VPC	2013-2014
3	p004	A	2013-2014
4	p005	W	2013-2014
5	p006	OL	2013-2014
6	p004	A	2014-2015
7	p005	P	2014-2015
8	p007	S	2014-2015

Tables

ConcertInsts Table

As talented musicians, many band members play a variety of concert and pep instruments throughout their time with the Marist Band. Concert Instruments perform in the various ensembles.

```
CREATE TABLE ConcertInsts(  
    pID      TEXT NOT NULL REFERENCES Participants(pID),  
    inst     TEXT,  
    part     INT  
);
```

Functional Dependencies

pID → inst, part

Sample Data:

	pid text	inst text	part integer
1	p002	Alto Saxophone	1
2	p003	Bari Saxophone	1
3	p005	String Bass	1
4	p006	String Bass	1
5	p007	Flute	2
6	p008	Tuba	1

Tables

PepInsts Table

As talented musicians, many band members play a variety of concert and pep instruments throughout their time with the Marist Band. Pep Instruments are used at Marist basketball games.

```
CREATE TABLE PepInsts(  
    pID      TEXT NOT NULL REFERENCES Participants(pID),  
    inst     TEXT,  
    part     INT  
);
```

Functional Dependencies

pID → inst, part

Sample Data:

	pID text	inst text	part integer
1	p002	Alto Saxophone	1
2	p003	Bari Saxophone	1
3	p005	Sousaphone	2
4	p006	Baritone	1
5	p007	Mellophone	2
6	p008	Sousaphone	1

Tables

MarchingInsts Table

As talented musicians, many band members play a variety of concert and pep instruments throughout their time with the Marist Band. Marching Instruments are used at Marist football games during the half time show.

```
CREATE TABLE MarchingInsts(  
    pID      TEXT NOT NULL REFERENCES Participants(pID),  
    inst     TEXT,  
    part     INT  
);
```

Functional Dependencies

pID → inst, part

Sample Data:

	pID text	inst text	part integer
1	p002	Alto Saxophone	1
2	p003	Bari Saxophone	1
3	p005	Sousaphone	2
4	p006	Baritone	1
5	p007	Mellophone	2
6	p008	Sousaphone	1

Tables

Conductors Table

There are many faculty members who conduct the different ensembles, most conduct more than one ensemble!

```
CREATE TABLE Conductors(  
    pID          TEXT NOT NULL REFERENCES Participants(pID),  
    startDate    DATE  
);
```

Functional Dependencies

pID → startDate

Sample Data:

	pid text	startdate date
1	p001	1986-01-01
2	p004	2004-06-01

Tables

Students Table

Many of the band members are also current Marist Students who lead very different academic lives outside of the band. Details pertaining to college enrollment are located here.

```
CREATE TABLE Students(  
    pID          TEXT NOT NULL REFERENCES Participants(pID),  
    gradYear     INT NOT NULL,  
    major        TEXT,  
    minor        TEXT,  
    scholarship  BOOLEAN  
);
```

Functional Dependencies

pID → gradYear, major, minor, scholarship

Sample Data:

	pid text	gradyear integer	major text	minor text	scholarship boolean
1	p002	2014	Computer Science	Music	t
2	p003	2014	Information Systems	Music	t
3	p005	2016	Business	Music	t
4	p006	2014	Psychology	Music	t
5	p007	2017	Psychology	Music	t
6	p008	2015	Environmental Science	Policy	f

Tables

Employees Table

Some band members receive work study from the college, and can satisfy this with a position within the Music Department, helping do the day to day operations of the program.

```
CREATE TABLE Employees(  
    pID      TEXT NOT NULL REFERENCES Participants(pID),  
    shift    TEXT NOT NULL CHECK(shift='day' or shift='night')  
);
```

Functional Dependencies

pID → shift

Sample Data:

	pid text	shift text
1	p004	night
2	p007	day

Tables

EnrolledIn Table

Keeping track of every ensemble each musician is in is difficult, but this helps keep the attendance record straight!

```
CREATE TABLE EnrolledIn(  
    pID      TEXT NOT NULL REFERENCES Participants(pID),  
    eID      TEXT NOT NULL REFERENCES Ensembles(eID)  
);
```

Functional Dependencies

pID eID →

Sample Data:

	pid text	eid text
1	p001	e001
2	p001	e002
3	p002	e001
4	p002	e002
5	p002	e003
6	p003	e001
7	p003	e002
8	p003	e009
9	p004	e005
10	p005	e001
11	p005	e002
12	p005	e008
13	p005	e009
14	p006	e001
15	p006	e002
16	p006	e008
17	p006	e009
18	p007	e001
19	p007	e002
20	p007	e003
21	p007	e008
22	p007	e010
23	p008	e001
24	p008	e002
25	p008	e005

Tables

GreekLifeMembers View

Compiles a list of band members who should report to all crews and be at set up before each rehearsal and be overall a more involved band member than their peers, regardless of whether they are a member of the Fraternity or Sorority.

```
CREATE OR REPLACE VIEW GreekLifeMembers AS
```

```
select p.pID, p.fName, p.lName
```

```
from Participants as p
```

```
where p.pID in      (select pID  
                    from TBSMembers)
```

```
    or p.pID in      (select pID  
                    from KKPsiMembers)
```

```
;
```

Sample Data:

	pid text	fname text	lname text
1	p002	Cassandra	Graves
2	p003	Brian	Veltre
3	p004	Michael	Napolitano
4	p005	Derek	Daffara
5	p006	Sara	Lacey
6	p007	Erin	Gates
7	p008	Elizabeth	Grisafi

Views

PaidBoardMembers View

Points out those members of the executive board that also get paid to work in the department. This is helpful for these employees to schedule office hours and take on extra assignments from the executive board.

```
CREATE OR REPLACE VIEW PaidBoardMembers AS
select p.pID, p.fName, p.lName, e.shift, em.role, em.term
from Participants as p, Employees as e, EboardMembers as em
where p.pID = e.pID
      and p.pID = em.pID
order by em.term asc;
```

Sample Data:

	pid text	fname text	lname text	shift text	role text	term text
1	p004	Michael	Napolitano	night	A	2013-2014
2	p004	Michael	Napolitano	night	A	2014-2015
3	p007	Erin	Gates	day	S	2014-2015

Views

StudentInstrumentRoster Report

Lists the students involved in the Marist Band and what kind of instrument they play in the Concert, Pep and Marching Bands.

```
select p.pID, p.fName, p.lName,  
       mar.inst as MarchingInstrument,  
       pep.inst as PepInstrument,  
       con.inst as ConcertInstrument  
from Participants as p, Students as s,  
     MarchingInsts as mar, PepInsts as pep, ConcertInsts as con  
where p.pID = s.pID  
     and mar.pID = p.pID  
     and pep.pID = p.pID  
     and con.pID = p.pID  
order by p.lName ASC;
```

Sample Data:

	pid text	fname text	lname text	marchinginstrument text	pepinstrument text	concertinstrument text
1	p005	Derek	Daffara	Sousaphone	Sousaphone	String Bass
2	p007	Erin	Gates	Mellophone	Mellophone	Flute
3	p002	Cassandra	Graves	Alto Saxophone	Alto Saxophone	Alto Saxophone
4	p008	Elizabeth	Grisafi	Sousaphone	Sousaphone	Tuba
5	p006	Sara	Lacey	Baritone	Baritone	String Bass
6	p003	Brian	Veltre	Bari Saxophone	Bari Saxophone	Bari Saxophone

Reports

EnsembleEnrolled Procedure

Returns the number of registered participants in the requested ensemble. Will not return values if no one is enrolled and therefore, is not an active ensemble.

create or replace function EnsembleEnrolled(text, REFCURSOR) returns refcursor as \$\$

declare

ensembleNumber text := \$1;

resultset REFCURSOR := \$2;

begin

open resultset for

select ens.eID, ens.name, COUNT(p.pID)

from Ensembles as ens, EnrolledIn as ei, participants as p

where ens.eID = ei.eID

and ei.pID = p.pID

and ens.eID = ensembleNumber

group by ens.eID;

return resultset;

end;

\$\$

language plpgsql;

Sample Data:

select EnsembleEnrolled('e008', 'results');

Fetch all from results;

	eid text	name text	count bigint
1	e008	Orchestra	3

select EnsembleEnrolled('e001', 'results');

Fetch all from results;

	eid text	name text	count bigint
1	e001	Symphonic Band	7

Stored Procedures

Admin Role

The admin, or administrator, is able to completely manipulate the Marist College Band database. Only certain individuals will be able to have sign in access as the admin.

create role admin;
grant all privileges
on all tables in schema public
to admin;

Leader Role

The leader, a responsible person who needs to query against the database, is able to see the database and make queries on it. Again, only certain individuals will be able to have sign in access, even as a leader.

create role leader;
grant select
on all tables in schema public
to leader;

Security

- **This is a highly important database to exist for the future of the Marist College Band. As the group is growing, it is getting harder to keep perfect records on the group. Being able to put all of the information in about every band member will help tremendously with organization. Keeping track of members involved in each of the various ensembles, requesting the roster or count of the number of participants in any ensemble, the instrument a member plays (and more) are all very common inquiries. Putting in all of this data initially was and would continue to be time consuming, but once caught up to present-day, it would only require a minor number of insertions and/or updates. The sky is the limit as far as what other information this database can and should hold. As long as it remains secure, it could gather even more personal details like street address and phone number of everyone involved.**

Implementation Notes

- **The database is currently as bare-bones as it could get. The minimal details and example participants does not bring out the true and ultimate potential that this database has. The only way to know what else to add into the database is by experiencing the band program. While there are many consistent things that occur year to year, each year brings new and different challenges that need to be observed and accounted for. With that said, as the band program grows, the database and type of detail to it will grow. Right now, the problems lie with needing to figure out where the future growth of the database is and planning accordingly. As a graduating senior of the program, I've seen a lot of change, but not nearly as much change as will come in the next year, 5 years, 10 years...so who knows how much more information the database will need until the time has come! So for the time being, the miniature Marist College Band database is perfectly suited for the details I've experienced with the program.**

Known Problems

- **Besides any events that are unforeseeable in the future that would need to be taken into account for this database to remain current and helpful, there were a few things I could have accomplished for this version of the database. For starters, getting the entire 100+ member band in all of the corresponding tables, plus the orchestra members or any other extraneous members to add in to the group would have been more helpful to have in the database. The example numbers of enrolled and other specifics to the tables are a poor reflection of how grand this band has become even to this day. With more data would come more ideas as to how leaders could view it. More time and more patience would be a help to improving the database in the future. But once all of the data is in, this table will be extremely useful for future presidents and executive board members of the Marist College Band!**

Future Enhancements
