

## Event Management System

### **English Description:**

The Event Management System database is designed to simplify organizing and managing large-scale events, such as multi-day conferences. A key use case for this database is the *Carleton CS Connect*, an event featuring diverse activities such as keynotes, workshops, panels, and speeches. The database is structured to efficiently handle the relationships between attendees, activities, speakers, and the overarching event.

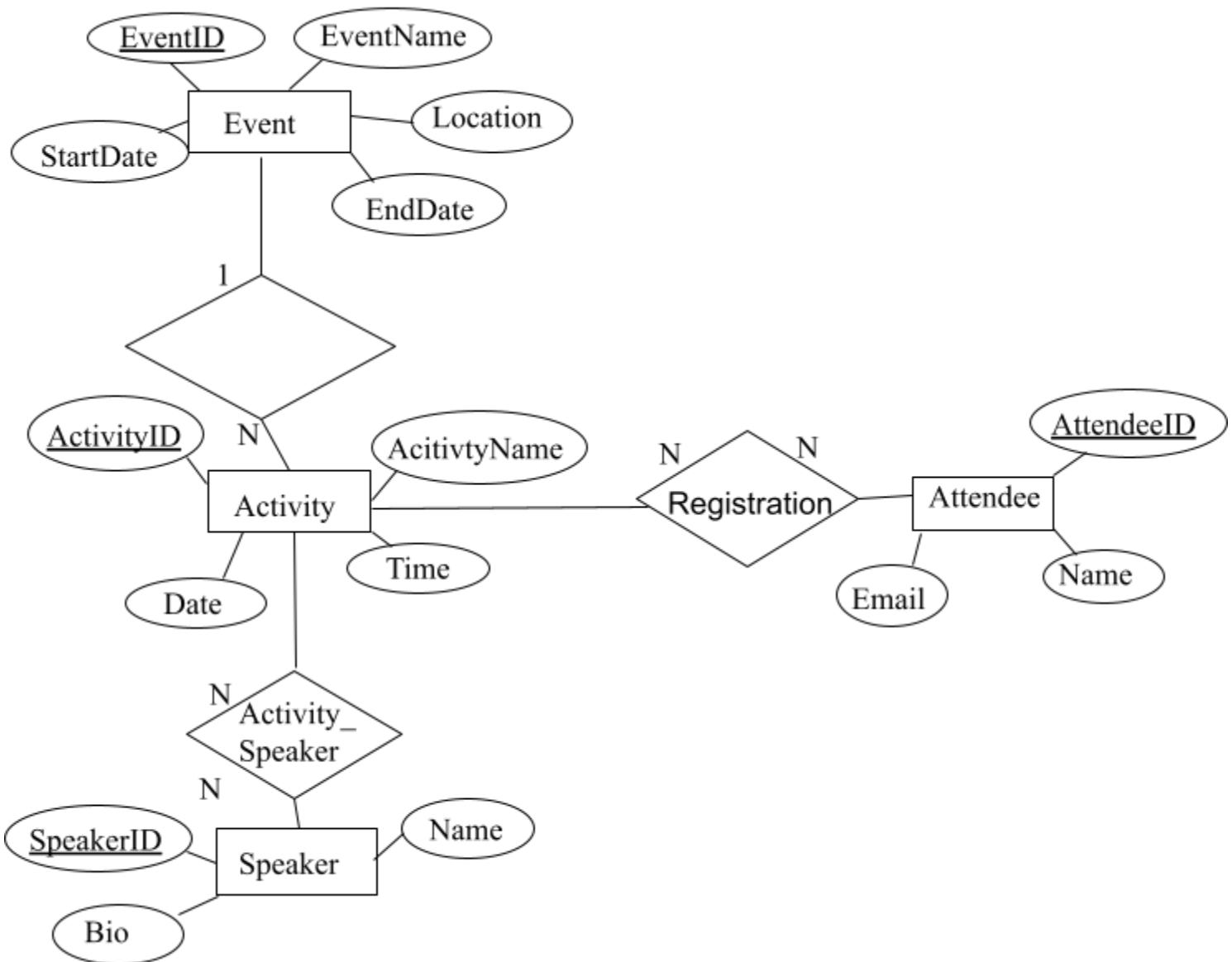
At the system's core is the Event entity, which captures details about the event, including its name, start/end dates, and location. Each event is composed of multiple activities, which represent individual sessions. Activities are linked to events through a one-to-many relationship and include attributes like name, date, and time.

To support attendee participation, the database includes the attendee entity, which stores the names and email addresses of participants. Attendees can register for multiple activities, and this many-to-many association is captured through the registration relationship, ensuring data integrity and preventing duplicate registrations.

The database also manages speakers through the speaker entity, which stores their names and biographical details. Since speakers can present at multiple activities and activities often feature multiple speakers, the ActivitySpeaker relationship is used to represent this many-to-many association. This ActivitySpeaker relationship effectively captures the association between speakers and activities, allowing the system to manage and organize the details of multiple speakers for each activity seamlessly.

Overall, the Event Management System database provides a concentrated framework for managing event logistics. It enables organizers to track attendee registrations, schedule activities, assign speakers with defined roles, and generate reports on event participation and engagement. Its design aligns closely with the ER diagram, ensuring clarity and ease of use while maintaining data integrity through the appropriate use of foreign key constraints and primary keys. This makes it an ideal solution for organizing and analyzing complex events.

### **ER Model:**



#### Schema Table:

##### Event

<u>EventID</u>	EventName	StartDate	EndDate	Location
----------------	-----------	-----------	---------	----------

##### Activity

<u>ActivityID</u>	ActivityName	Date	Time	EventID (Foreign Key to EventID in Event)
-------------------	--------------	------	------	--

##### Attendee

<u>AttendeeID</u>	Name	Email
-------------------	------	-------

### Registration

<u>AttendeeId</u> (Foreign Key to AttendeeID in Attendee)	<u>ActivityID</u> (Foreign Key to ActivityID in Activity)
---	---

### Speaker

<u>SpeakerID</u>	Name	Bio
------------------	------	-----

### ActivitySpeaker

<u>SpeakerId</u> (Foreign Key to SpeakerID in Speaker)	<u>ActivityID</u> (Foreign Key to ActivityID in Activity)	Role
--	---	------

### Proposal:

#### Description of project:

#### Background:

- Managing large events, such as conferences, workshops, or concerts.
- Can be complex, involving multiple participants, activities, and schedules.
- This database is designed to simplify the management of these events by tracking event details, activities, and attendees.
- The goal is to allow event organizers to easily manage registrations, monitor attendee participation, and keep track of activity schedules.
- The system will allow users to generate reports on attendance and participation to help organizers manage their events more efficiently.

#### Service Provided:

- **Event Management:** Create and manage events with detailed descriptions, dates, and locations.
- **Activity Management:** Each event can have multiple activities, such as talks, workshops, or entertainment sessions.

- **Attendee Registration:** Users can register for multiple events and specific activities within those events.
- **Attendance Reporting:** Generate reports to see how many people attended specific activities.
- **Guest Speaker:** Represent individuals who present and participate in various activities within an event

### **Application Requirements:**

- **Event Registration:**
  - Users should be able to register for 1 event
  - Each event should have a unique identifier, name, and location.
  - Users should be able to browse through the event and view detailed information.
- **Activity Management:**
  - The event can have multiple activities (such as workshops, keynote speeches, and entertainment sessions)
  - Activities should have a unique identifier, name, and schedule within an event.
  - Users can register for multiple activities within an event, and each activity can have multiple attendees - **Attendee Management:**
    - Each attendee should be able to register for multiple activities.
    - The database should keep track of attendee information such as name and registration details.
    - Attendees should be able to see what activities they are registered for.
- **Attendance Reporting:**
  - Event organizers should be able to generate reports showing attendance numbers for activities.
  - The system should also allow organizers to export attendance data for analysis.
- **Guest Speaker:**
  - The system must allow organizers to assign speakers to specific activities within an event
  - Organizers should be able to assign multiple speakers to multiple activities
  - The system should display detailed information about each speaker, information should be visible to attendees
  - Multiple speakers can be involved in multiple activities across the event