



#### Node Properties

1. User
  - a. User id
  - b. User Name
  - c. Team
  - d. Age
  - e. UserjoiningDate
  - f. Language
  - g. Country
2. Chat Session
  - a. Session id
  - b. Start\_time
  - c. End\_time
3. Chat Text
  - a. Text\_id
  - b. Text
  - c. Timestamp
  - d. Language
  - e. Flagged\_offensive
  - f. Offensive

#### Edges Properties

1. joins
  - a. timestamp
2. leaves
  - a. timestamp
3. starts
  - a. timestamp
4. contains
  - a. timestamp
5. writes
  - a. timestamp
6. mentions
  - a. timestamp
7. flagged\_offensive
  - a. timestamp

#### Additions to Graph Structure:

1. Added node for team so analysis could be done on chats between team relating to certain events happening on team
  - a. Add Node
    - i. Team
      1. Team Name
      2. Team id
2. Added Edges to represent the process of requesting to join team, joining a team, leaving a team and failed team votes, so that analysis can be conducted on these events
  - a. Add Edges between User and Team Nodes
    - i. joined\_team
      1. timestamp
      2. vote\_%
    - ii. left\_team
      1. timestamp
      2. vote\_% (blank if voluntary)
    - iii. request\_to\_join
      1. timestamp
    - iv. failed\_vote
      1. timestamp
      2. vote\_%
3. Added game\_level as this could be an interesting element of meta data when analysing chats
  - a. Add Node
    - i. game-level
      1. Game\_level
  - b. Add Edges between User and Game\_Level Nodes
    - i. achieved
      1. timestamp
    - ii. moved
      1. timestamp
4. Added User\_category as this could be an interesting element of meta data when analysing chats

- a. Add Node
  - i. User\_Category
    - 1. Category
- b. Add Edges between User and User\_Category Nodes
  - i. gained\_category
    - 1. timestamp
  - ii. lost\_category
    - 1. timestamp

#### Questions:

- 1) Look at the connections between Team and Chat Sessions (or Chat Text – depending on how you define conversations) . Identify teams with highest number of links. Note the link is via the User object.
- 2) Look at connection between User and Chats Text. Calculate average Chats per user. Repeat operation, but filter for period xx period before the user leaves the team, defined by the left\_team edge, timestamp property between user and team nodes.
- 3) Look at Chat Text node. Using the timestamp property to define period. Once you have the corpus. Use word count algorithm on the corpus.
- 4) Use the edge between User and Chat Session. Filter edges based upon the session id of Chat Session. Count edges per user. Sort – identify users with highest count.
- 5) Based on edges between User and Chat Session. Depends on what you are after –
  - a. If at point in time – filter for Chat Sessions open at point in time connected to user. Use edges joins, starts and leaves to identify the number of sessions open
  - b. If more general – e.g. Max number – look at edges between user and chat sessions. Run a running count, adding 1 when joined and subtracting 1 when left. Keep max number for running count.