

### Cap Strategy Questions:

4. What is the 2024-25 dead cap situation for this team? How does it help or hurt this team's ability to construct a roster?

The Flyers have a dead cap of \$16,167,857. This hurts the team's ability to sign new players or renew contracts. If they want to sign a player, they might have to let a player go to free up some cap space.

5. Your GM wants to know how to evaluate the effect that signing a \$8M / year goalie has on improving and constructing the roster. On a high-level, how would you begin to build a process to help answer that question?

I would begin by evaluating what the team's cap space is for the next few years, and do some statistical analysis to see if the move makes sense. If there is barely enough cap space to sign the goalie, I would not sign him in order to leave room for other contracts. I would also track their save percentage, goals against average, and wins. If they are significantly better than our current goalie, I would favor signing them. As someone who played goalie, I believe that a goalie could be the edge that a team needs to win the cup, and could be worthy investment.

## Cap Data Questions:

6. What other information would you want in order to further evaluate the future roster outlook of this team? (for questions like #5)? How would you change your process to help with that evaluation?

I would also look at when contracts are expiring and cap space for future years. For instance, if you have a star player who's rookie contract is running out, you might consider freeing up cap space in order to give them a fair contract. It is important to look at your roster, and see who is performing, and who you would want to resign. If you have any expensive contracts from players who are not performing well, you might want to consider a buyout or a trade.

Overall, I would not only look at cap space for future years. I would also look at how players are performing, and consider the prospect pipeline.

7. The capdata.json file contains information for one team, and is in a nested format. To scale this across all 30 teams, you might want to store that data in a database. Would you consider a relational database or a NoSQL database here? What are the pros and cons of either solution? If it's applicable, consider your answers to \$5 and \$6 in helping answer this question.

I would use a NoSQL database like MongoDB, as it is similar to a json format in that it holds nested data. This would be useful in cases where the schema isn't always consistent. For instance, in the buyout section of the json file, under each player's contract details, there is sometimes a buyout field and other times there is not. NoSQL databases adapt well to these types of inconsistencies.

However, an SQL database would be better if you have multiple tables of different data. For instance, if you want to compare different goalie's save percentages to their salaries, SQL would be optimal. It allows for more complex queries, allowing for more complex data analysis.