In the last two decades, both researchers and vendors have built advisory tools to assist database administrators (DBAs) in various aspects of system tuning and physical design. Most of this previous work, however, is incomplete because they still require humans to make the final decisions about any changes to the database and are reactionary measures that fix problems after they occur. What is needed for a truly self-driving database management system (DBMS) is a new architecture that is designed for autonomous operation. This is different than earlier attempts because all aspects of the system are controlled by an integrated planning component that not only optimizes the system for the current workload, but also predicts future workload trends so that the system can prepare itself accordingly. With this, the DBMS can support all of the previous tuning techniques without requiring a human to determine the right way and proper time to deploy them. It also enables new optimizations that are important for modern high-performance DBMSs, but which are not possible today because the complexity of managing these systems has surpassed the abilities of human experts.

In this talk, I will present our roadmap towards developing a self-driving DBMS. It has three main components: workload forecasting, action modeling, and planning. I will also present our solution to the first component: a forecasting framework called QueryBot 5000 that allows a DBMS to predict the expected arrival rate of queries in the future based on historical data. It provides multiple horizons (short- vs. long-term) with different aggregation intervals. We implemented our forecasting technique in an external controller for PostgreSQL and MySQL and demonstrate their effectiveness in selecting indexes. Finally, I will present our on-going project in this journey and our long-term vision.

Based on joint work with Dana Van Aken, Ahmed Hefny, Gustavo Mezerhane, Andrew Pavlo, and Geoffrey J. Gordon.

Homepage: http://www.cs.cmu.edu/~malin199