Jacob Kilver

Knowledge Based AI

Assignment 5

10 July 2016

# Introduction

This paper will explore the use of diagnosis as applied to software debugging. My full-time job is as a developer on a support team for ATM software, so a large portion of my time is spent analyzing and diagnosing issues with software. I could personally relate to much of what was discussed in the lecture on diagnosis.

# Problem Description

Diebold Inc. (<http://www.diebold.com/>) is a global manufacturer of automated teller machines (ATMs) and other financial institution products and services. While ATMs seem rather simple, there are many things that can go wrong with them. As a software support engineer, I am the third layer of support, but the first team that has access to the source code of our ATM application. As such, my group handles a wide variety of software issues, which can make diagnosing an issue quite a challenge. Here is a sampling of the difficulties in full-stack software debugging:

* Poor descriptions from the field.
  + Many times an escalation has false or missing data
* Incomplete data from field
  + We have a tool that is supposed to gather everything, but it usually misses a few things
* Lots of code
  + The application I help manage currently has a SLOC count of \_\_\_\_\_, not including the other layers
* Many layers
  + The complete software stack consists of no fewer than 5 layers, not to mention all the software add-ons
* Poor, missing, and unclear documentation/specifications
  + Defining what the correct behavior should be is not always easy
* Differing customer opinions/desires
  + Some customers want the software to behave like certain legacy applications
  + Others want customized solutions that do not accommodate all customers
* Multivendor environments
  + A surprising number of customers want hardware from one vendor and software from another
  + This requires knowledge that is legally not within your domain

# Approach

# Conclusions