



Enterprise Architecture Technology

Jordan Lewis

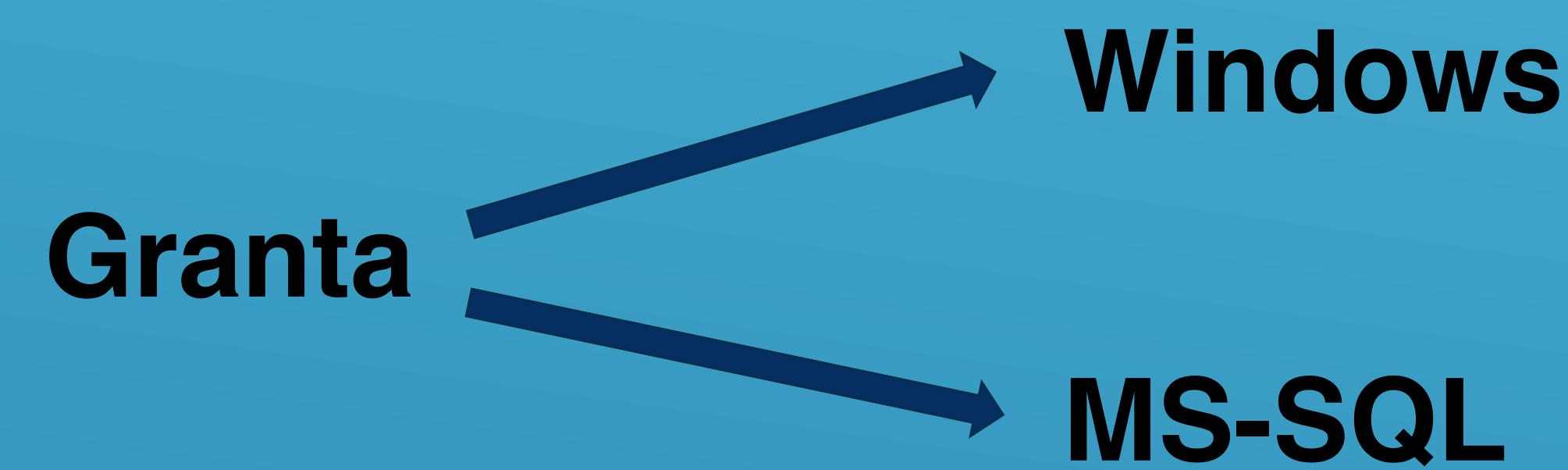
Craig Cothren, Ian Granger



ARIZONA STATE UNIVERSITY

Introduction:

- Introducing new software, updating software, installing software on multiple operating systems, etc. can cause *unforeseen conflicts* between software packages.
- These conflicts have the potential to cost the lab *hours of downtime* as we remedy the issue.
- *How can we mitigate future conflicts?*



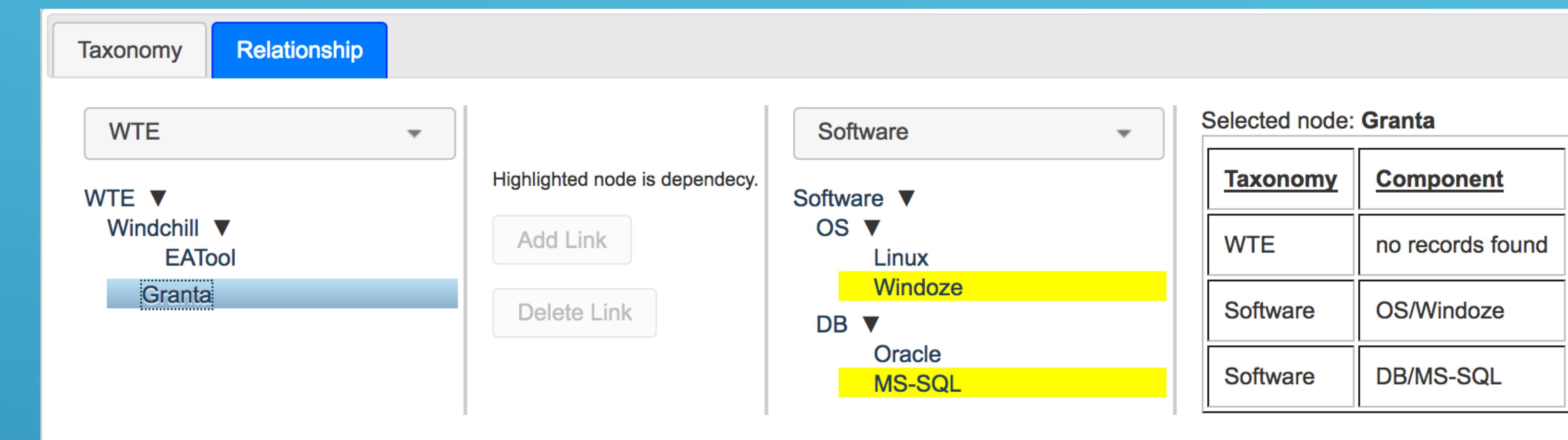
Methods:

1. Web based
 - A. jQuery UI
 - B. jqTree
2. Tabular data
3. Visual representation
 - A. Colors

A screenshot of a web-based application interface. It has tabs for "Taxonomy" and "Relationship". In the Taxonomy tab, there is a dropdown menu set to "WTE" which shows a hierarchical tree structure: WTE ▼ Windchill ▼ EATool ▼ Granta. Below the tree are buttons for "Add Root", "Add Child", and "Delete". In the Relationship tab, there is another dropdown menu set to "WTE" which shows the same tree structure. To its right is a dropdown menu set to "Software" which lists OS, Linux, Windoze, DB, Oracle, and MS-SQL. Below these dropdowns are buttons for "Add Link" and "Delete Link".

Results:

- *Web program* that visually represents dependencies between software packages and allows users to *easily* create, delete, and manipulate dependencies and hierarchies.



Discussion:

- Color representation feature does not work well when there are many dependencies (too busy).
- Search feature?
- Drag and drop feature?

