∫οκσβαπ

Created, Designed, and Programmed

by

Cyrus Baker,

Daniel Moore, &

Yarely Ogaz

A Chunky Leopards Production

Overview

Our project is a Sokoban puzzle with an integrated math challenge to solve each level. To complete each level the player must complete two steps. First the player must move all the blocks to the specified target areas. Once the puzzle is completed, the math problem displayed above will reveal the value of some variables. Once completing both parts correctly, the player advances to the next level.

Drivers

Requirements that were essential in driving our architecture decision:

- Saving puzzles
- Loading puzzles
- Manipulating Sokoban puzzles
- Mouse and key detection
- Math Prototypes
- Having 2 puzzle stages: Sokoban and Math

Architectural Choices

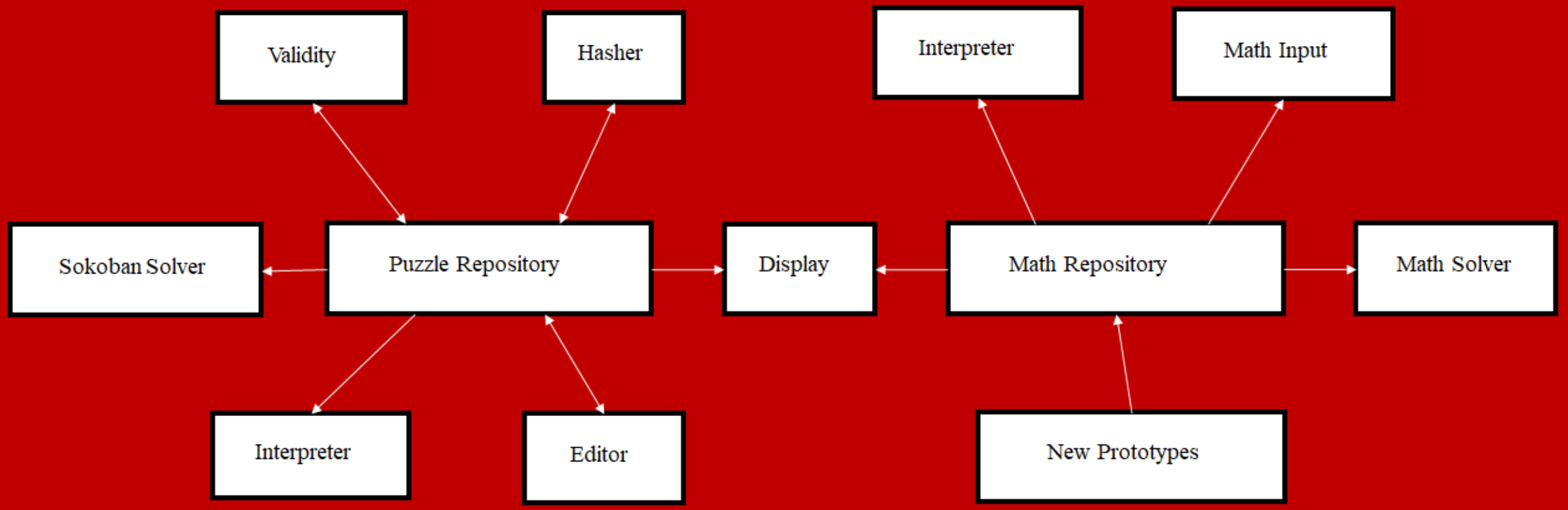
PUBLISH & SUBSCRIBE

- When one part announces, the corresponding parts respond
- Easy to reuse
- Necessary for mouse and keyboard actions

REPOSITORIES

- Retrieval
- Storage
- Traditional Database: Transactions trigger process execution

The main architecture style we chose to use is repositories.



Architecture

Conclusion

- Our main chosen architectural style is repositories

- Risks:

1. Format is accepted by all
2. Might increase complexity
3. Reducing security

- Questions?