# Velosiraptor: Why program yourself when you can synthesize address translation?

*Reto Achermann, Ryan Mehri, Ilias Karimalis, Margo Seltzer*

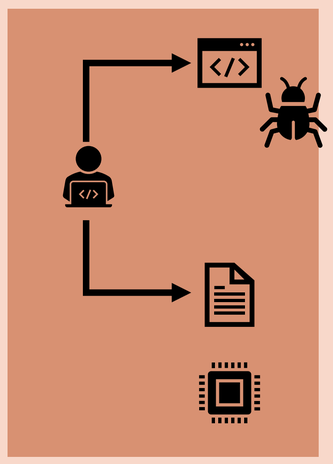
## Problem

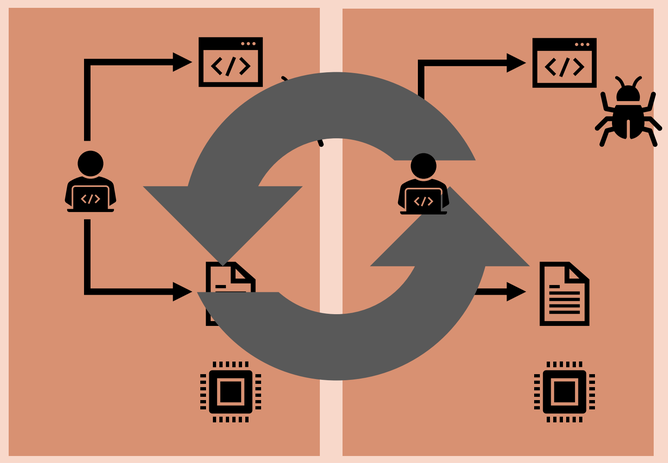
Memory translation and protection is essential to many abstractions.

Correct configuration of translation hardware is security critical.

Today: developer reads documentation and **manually** writes OS code.
- **time and effort** to develop and debug implementation
- possible **error prone** process
- **ambiguous** documentation

New translation hardware: **repeat the process**
- spend more time and effort manually writing code

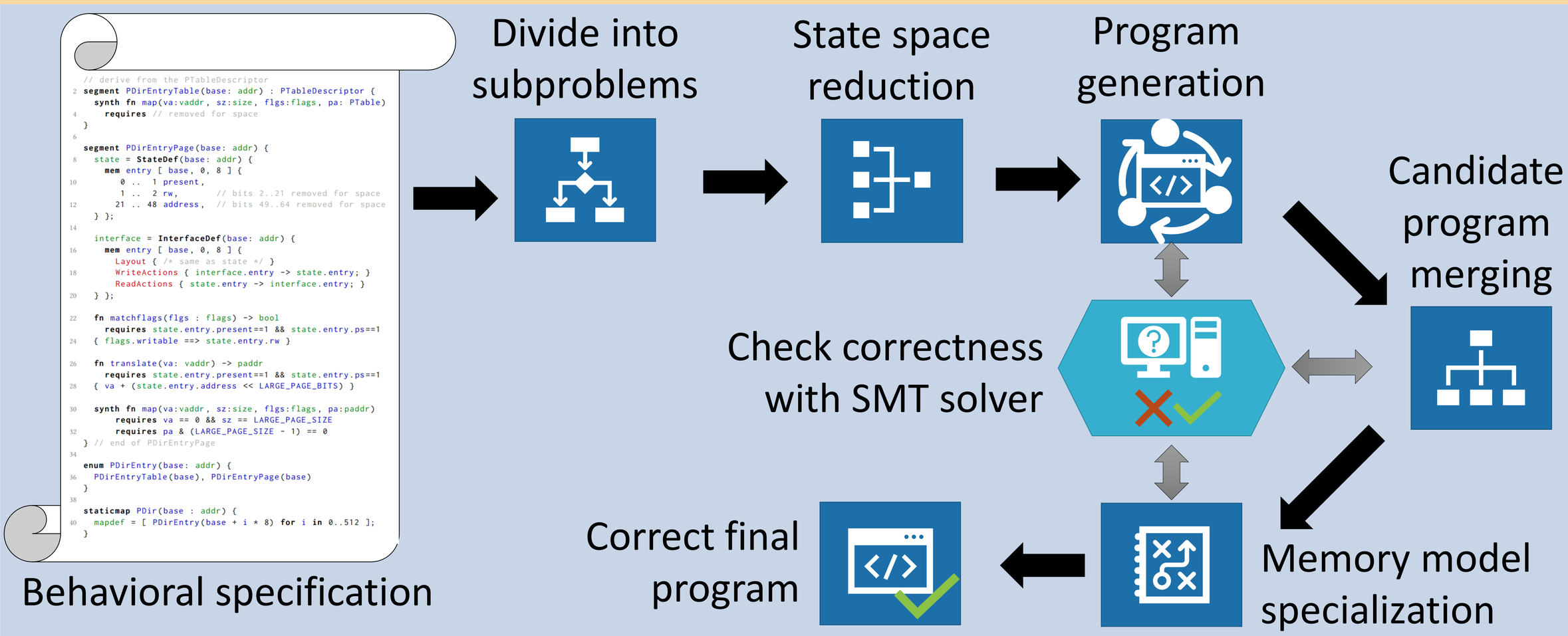




## Idea

**Automatically synthesize correct, low-level operating systems code from a behavioral specification.**

## Approach




## Behavioral Specification

Designed with **decomposition** in mind

Writing specifications as a **combination of building blocks**:

Specification of **OS-visible interface** and translation-defining internal **state**

Defining remap semantics in terms of translation and protection

## Results

**Successfully synthesized** OS code for different translation hardware:
- multi-level page tables
- segmentation
- TLBs

Synthesis time **~400-800ms**

Generate **hardware components** for simulators from the same specification.

THE UNIVERSITY OF BRITISH COLUMBIA
Computer Science

Systopia Lab
Researching Systems, construed Broadly

systopia.cs.ubc.ca

achreto@cs.ubc.ca