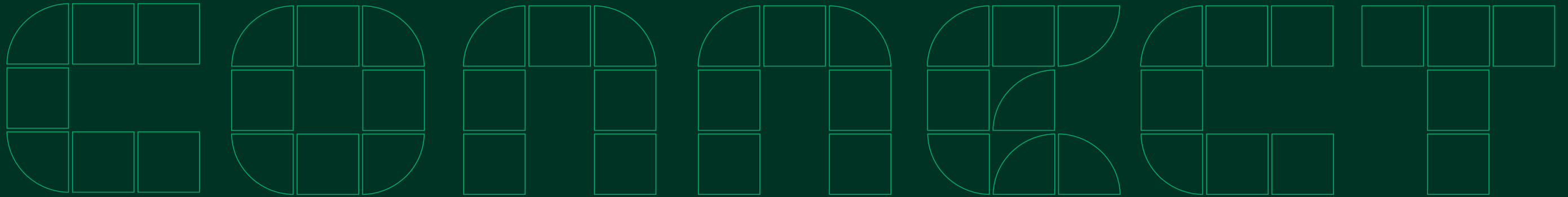


Malleable VI's unlocked

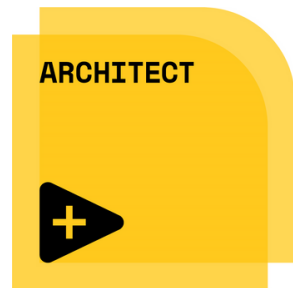
Natan Biesmans



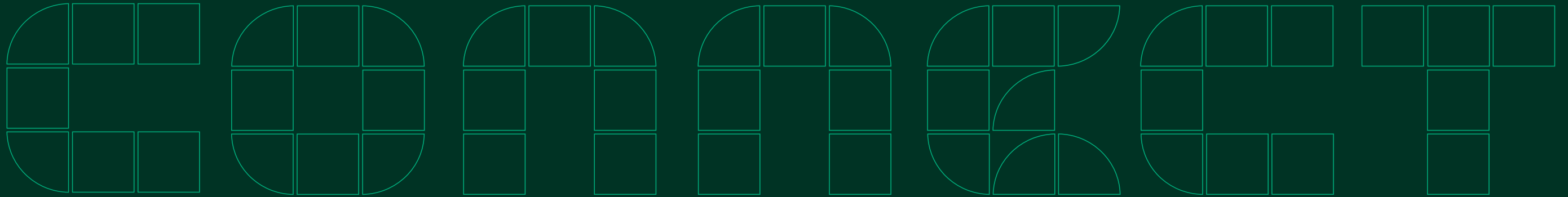


Introduction

Natan Biesmans



- CLA, LabVIEW Champion
- Started using LabVIEW 10 years ago.
- Previous presentations
 - FPGA Accelerated Computing
 - Native LabVIEW Applications on Android and IOS
 - GPU computing made easy with G²CPU
 - ...
- Architect of G²CPU, the CPU and GPU HPC compute toolkit for LabVIEW.
- QPLOX Engineering: LabVIEW R&D Architect @ IMEC
- National Instruments (6 years ago): Application Engineer.



Introduction

What makes LabVIEW stand out

What makes LabVIEW unique

- Fast development and verification
 - Compiled: directly know if your code has runtime errors.
- Powerful debugging tools
 - Setting probes at runtime
 - Execution highlighting
 - Each function has a UI



The screenshot shows the 'Probe Watch Window' in LabVIEW. It contains a table with the following data:

Probe(s)	Value	Last Upc
Untitled 1 (G ² C		
[1] Array	[T, F, F, T, T]	13/03/20

Below the table is the 'Probe Display' panel, which shows various settings for the probe:

- Toolkit: v12.0
- Compute Version: 7.5
- Array Pointer: 1601245963888
- Platform: CUDA
- Device Name: NVIDIA_GeForce_RTX_20
- Average: 0,6
- Std dev: (partially visible)

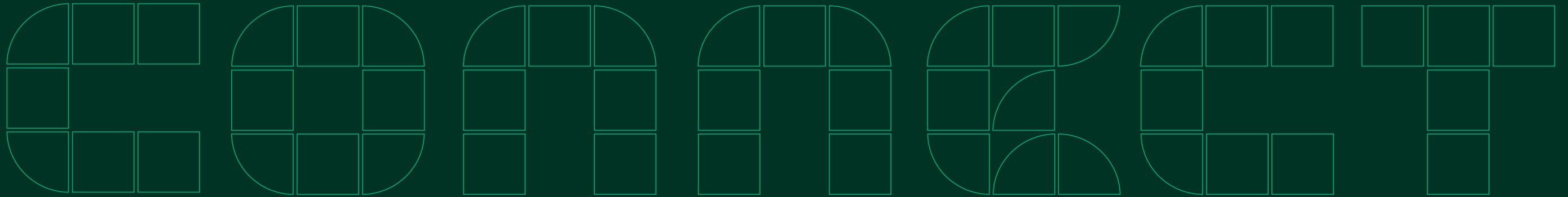
At the bottom of the panel, there is a small UI element showing a '0' and a vertical stack of five green indicator lights.

“The tools of a scripting language,
the speed of a compiled language”

What makes LabVIEW unique

- Hyper reuseable code
 - Across CPU's
 - Operating Systems
 - Real-time
 - FPGA
 - Bitness
 - ...

“Code it once and code it right”



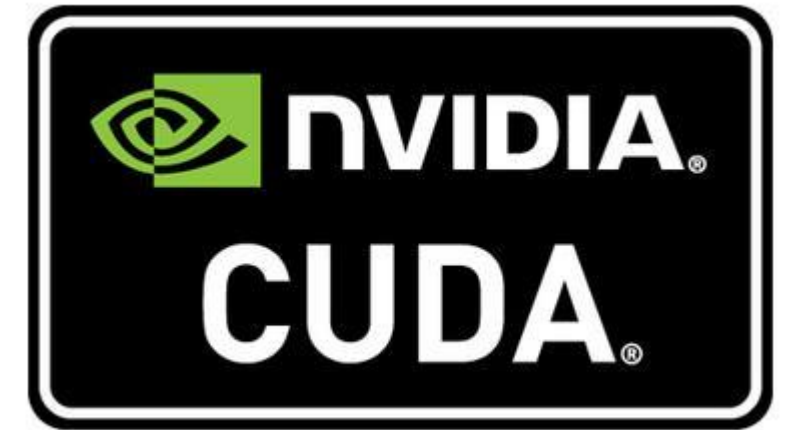
Introduction

Origins of this presentation

Origins of this presentation

Problem statement

- G²CPU the GPU and CPU HPC toolkit for LabVIEW
 - Hundreds of functions
- Several technologies
 - CUDA
 - OpenCL
 - CPU
 - Intel OneAPI



Origins of this presentation

Problem statement

- G²CPU the GPU and CPU HPC toolkit for LabVIEW
 - Hundreds of functions
 - Several technologies
 - Different environments
 - Windows
 - Linux
 - LabVIEW RT



Origins of this presentation

Problem statement

- G²CPU the GPU and CPU HPC toolkit for LabVIEW
 - Hundreds of functions
- Several technologies
- Different environments
- Lots of input variations
 - 13 datatypes
 - 4 Dimensions
 - GPU and LabVIEW Arrays
 - Scalars

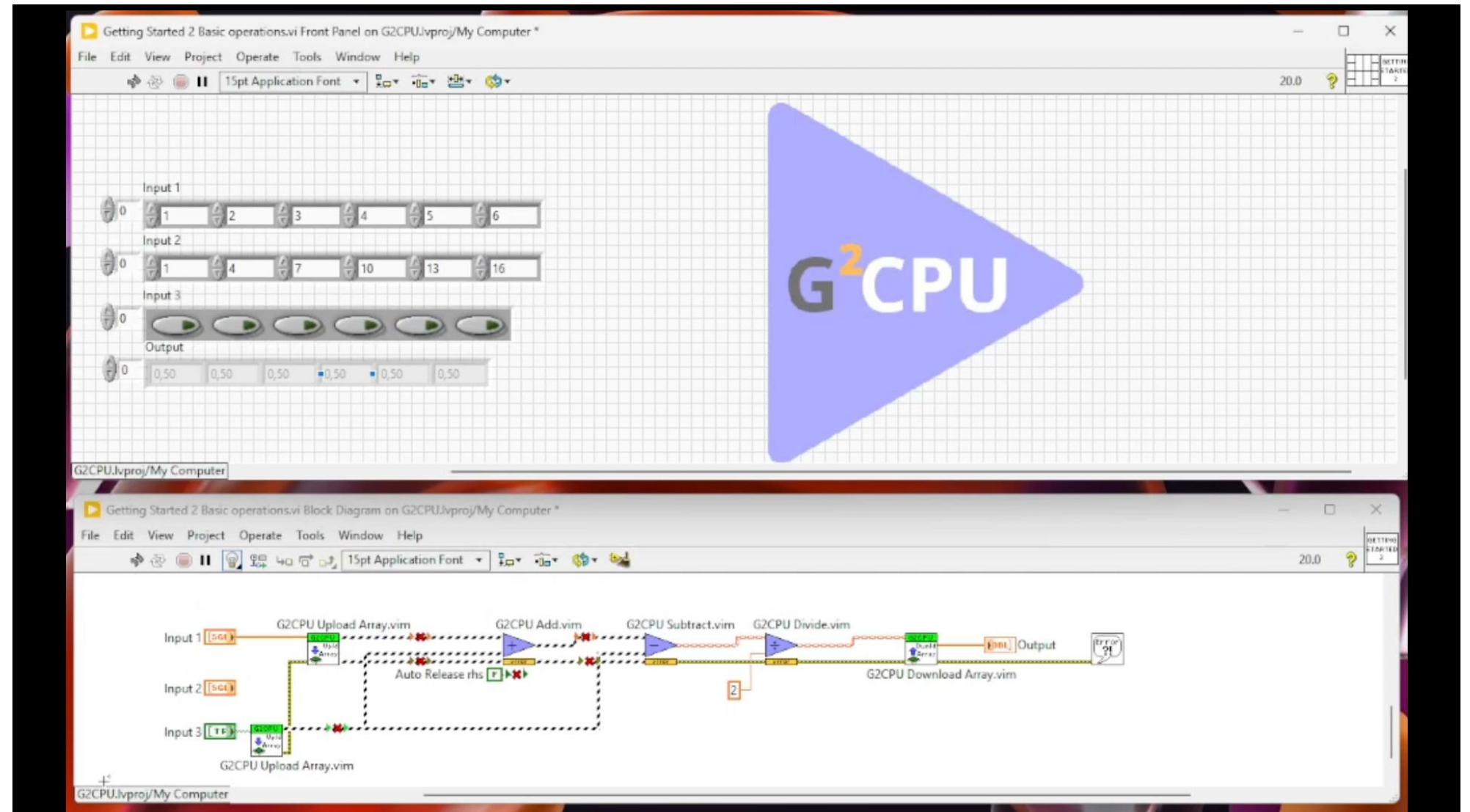
A simple addition has over 12 168 possible permutations

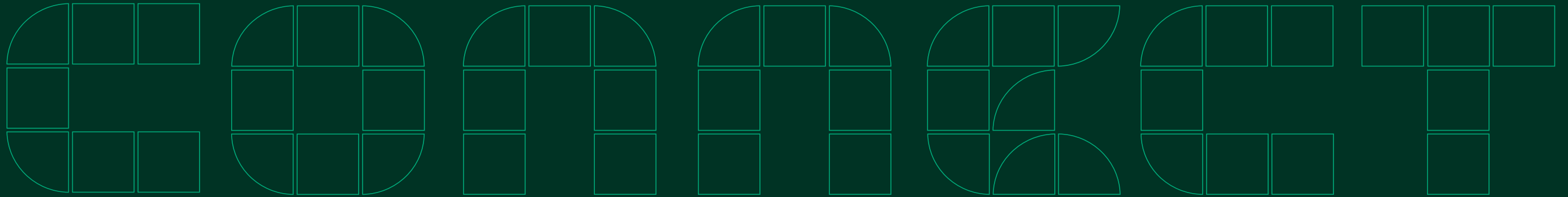
Permutation hell

Origins of this presentation

API lessons learned

- Provide a near LabVIEW native experience
- Development time evaluation
- Dynamic wire appearance
- Selective probes
- Managing permutations
- Unit tests

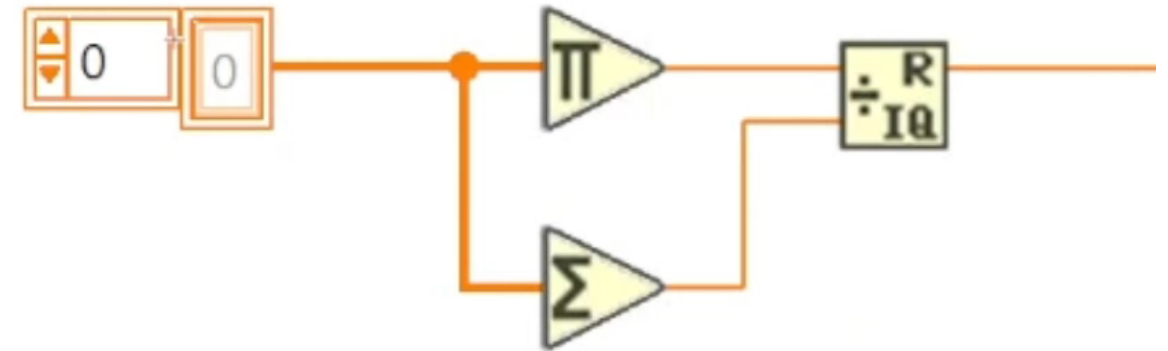




Why should you use Malleable VIs

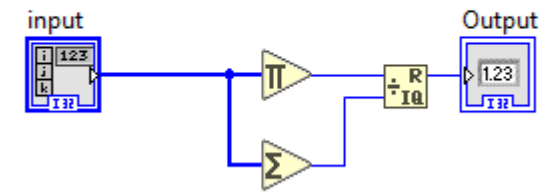
What Are Malleable VIs

- Special VI type called VIM
- Technique for dynamic code execution
- Pass through of datatypes
- Development time evaluation



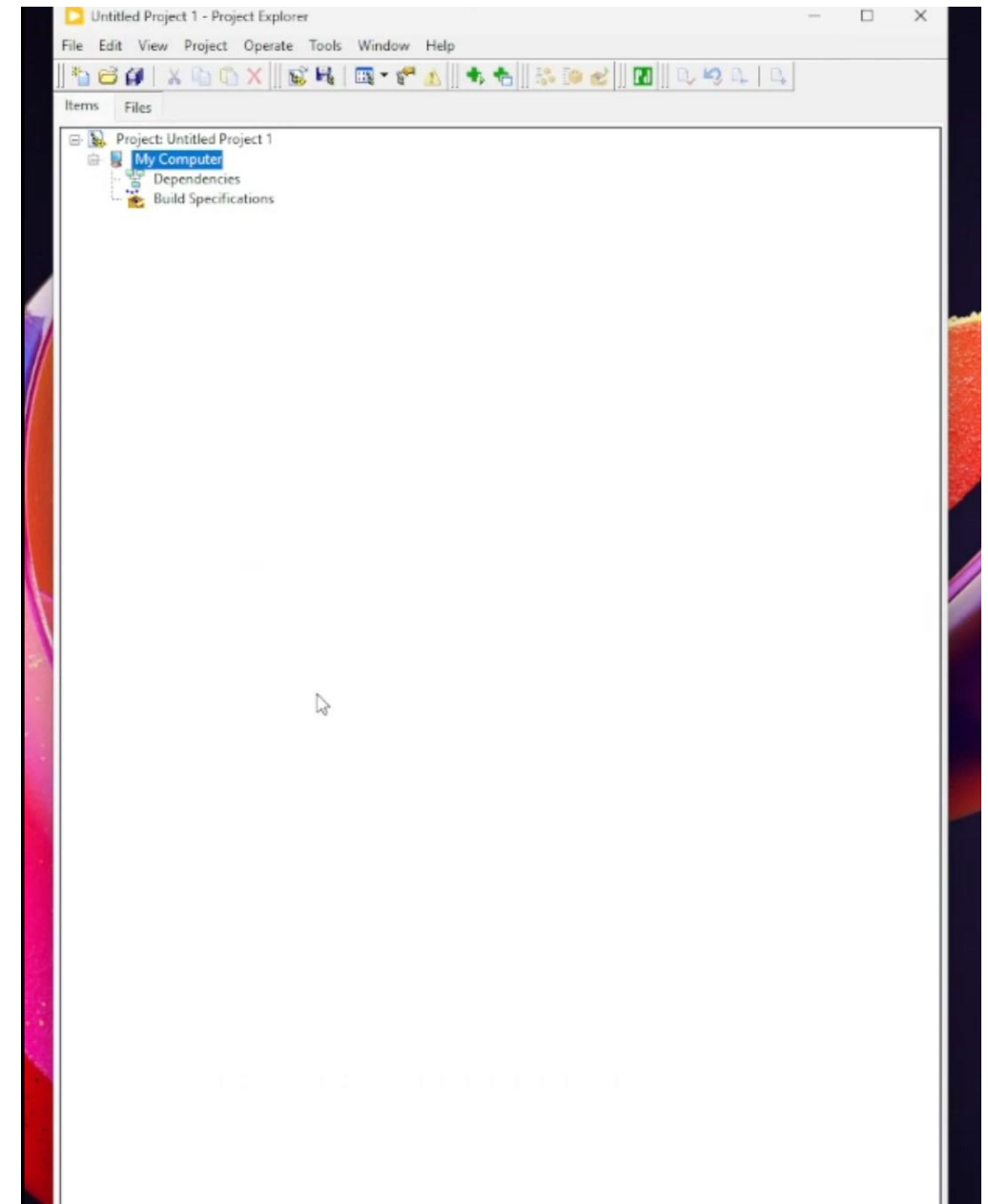
What Are Malleable VIs

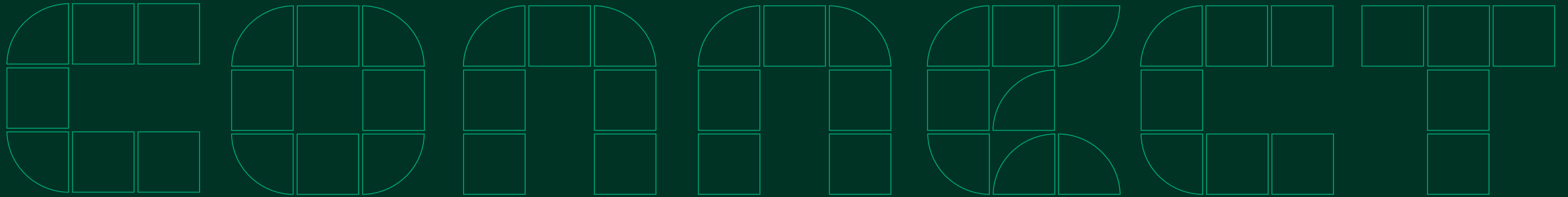
- Special VI type called VIM
- Technique for dynamic code execution
- Pass through of datatypes
- Development time evaluation



How to make a malleable VI

- Convert VI
 - .vim extension
 - Inline SubVI
 - Preallocated Clone Reentrant Execution
 - No error handling
- Template



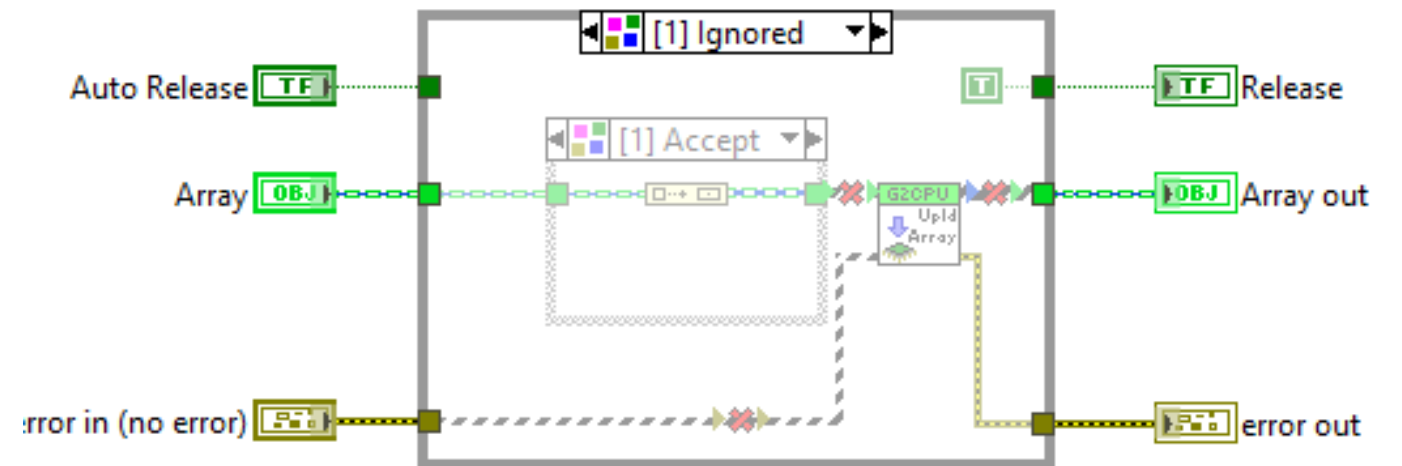


Why should you use Malleable VIs

Fundamentals of scalable malleable VIs

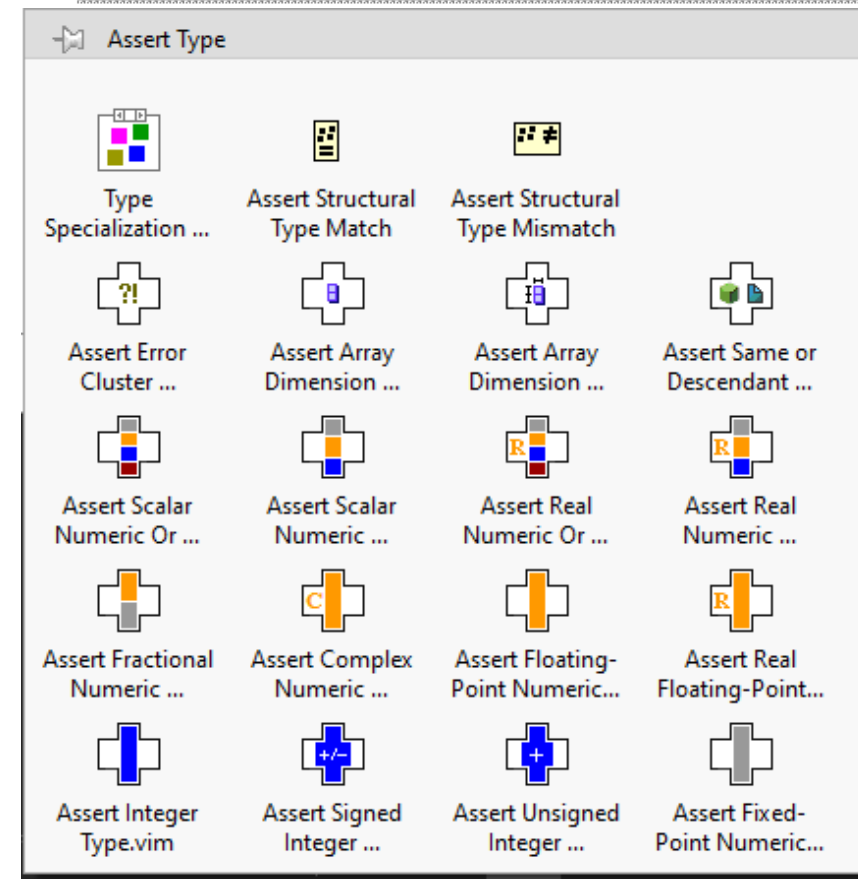
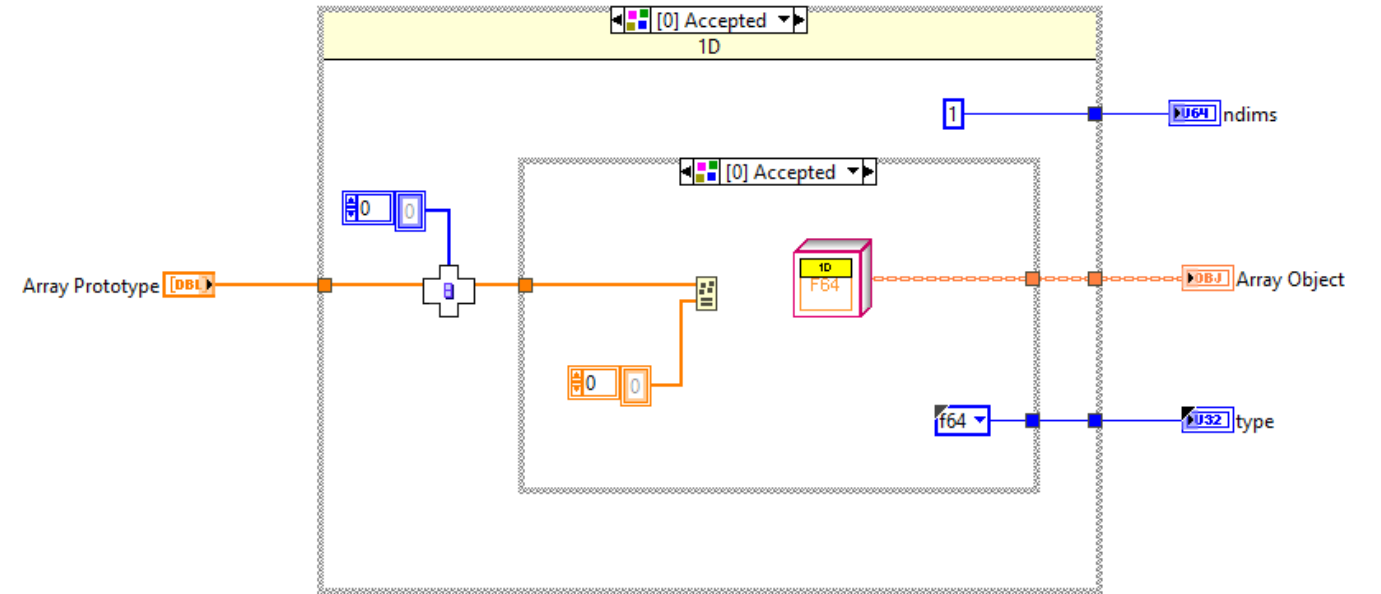
Type Specialisation

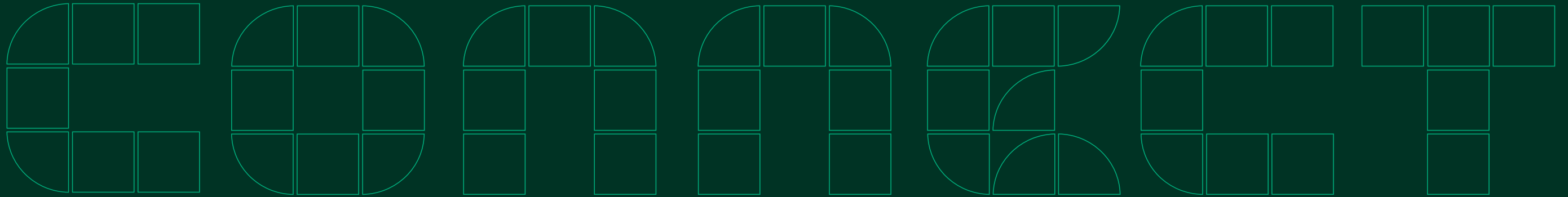
- Define multiple behaviors based on certain conditions.
- Evaluate each frame sequentially
 - If frame can be compiled it will be used
 - If a frame can't be compiled, go to next.
 - All other frames won't be evaluated



Assert

- Break code under certain conditions
- Used to
 - Only allow certain types
 - Disallow certain types



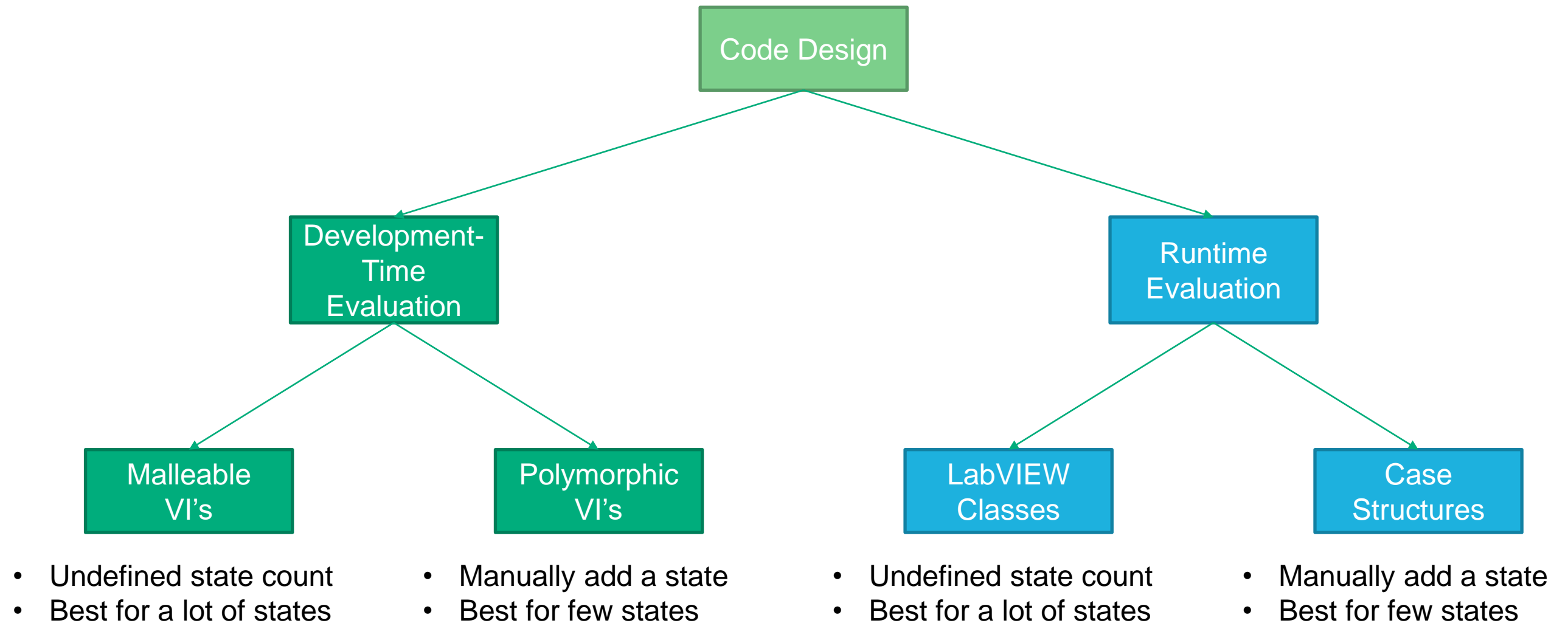


Why should you use Malleable VIs

When should you use malleable VIs



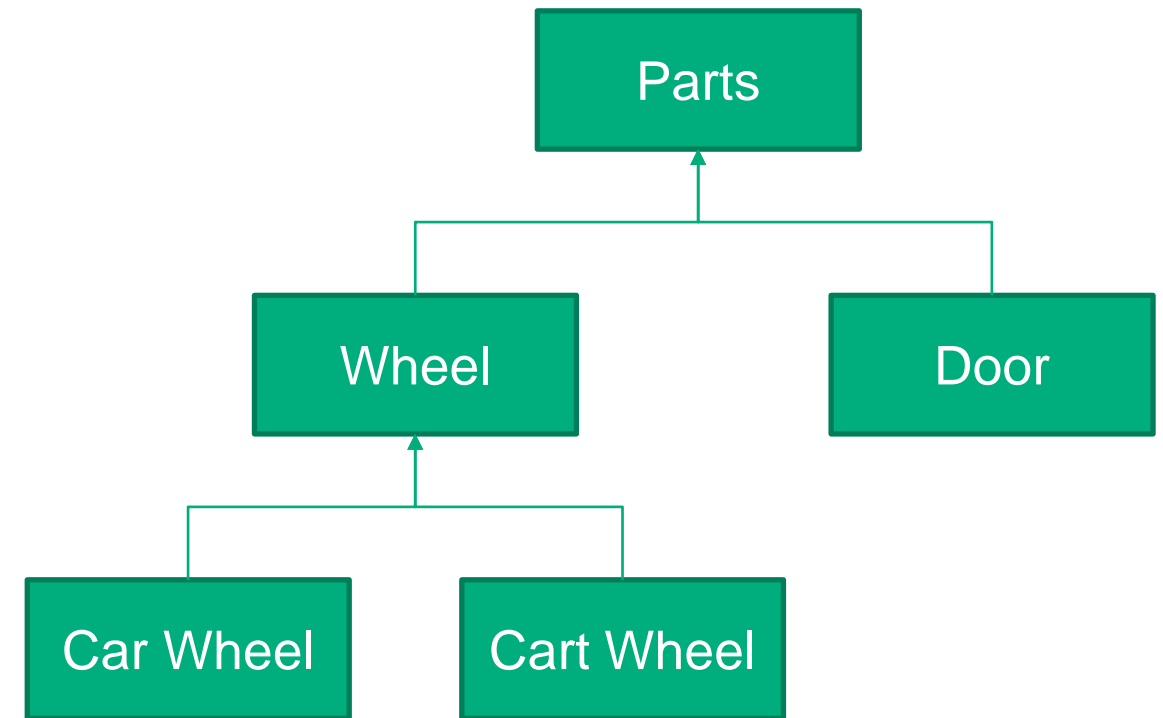
Dynamic Code execution comparison



LabVIEW Classes

Assign attributes to objects

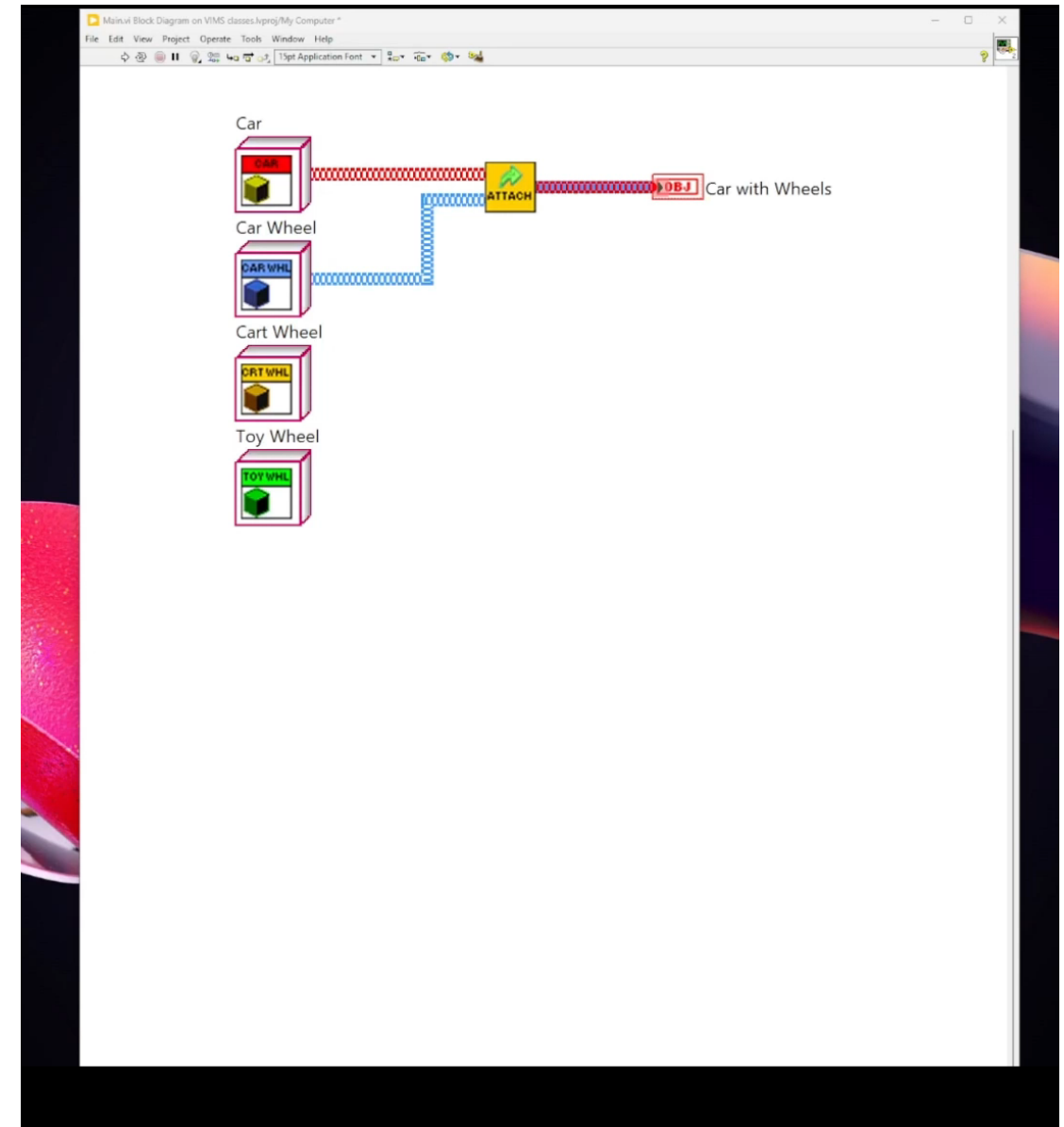
- IDE Decision making
- More flexibility than Dynamic Dispatch by using child classes
- Wire colors
- Dynamic probes

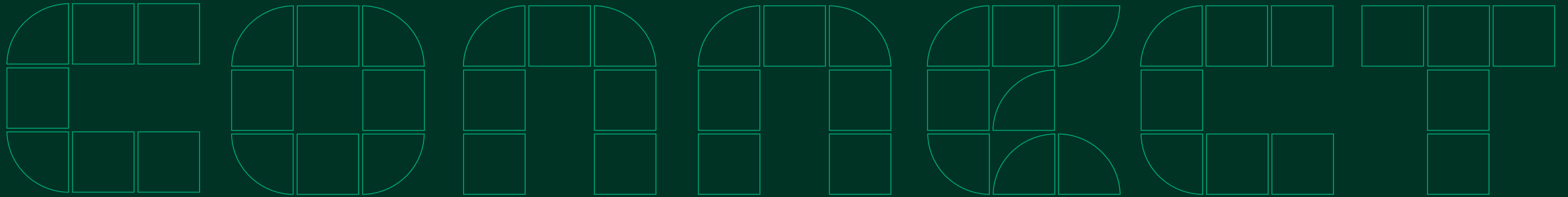


LabVIEW Classes

Assign attributes to objects

- IDE Decision making
- More flexibility than Dynamic Dispatch by using child classes
- Wire colors
- Dynamic probes





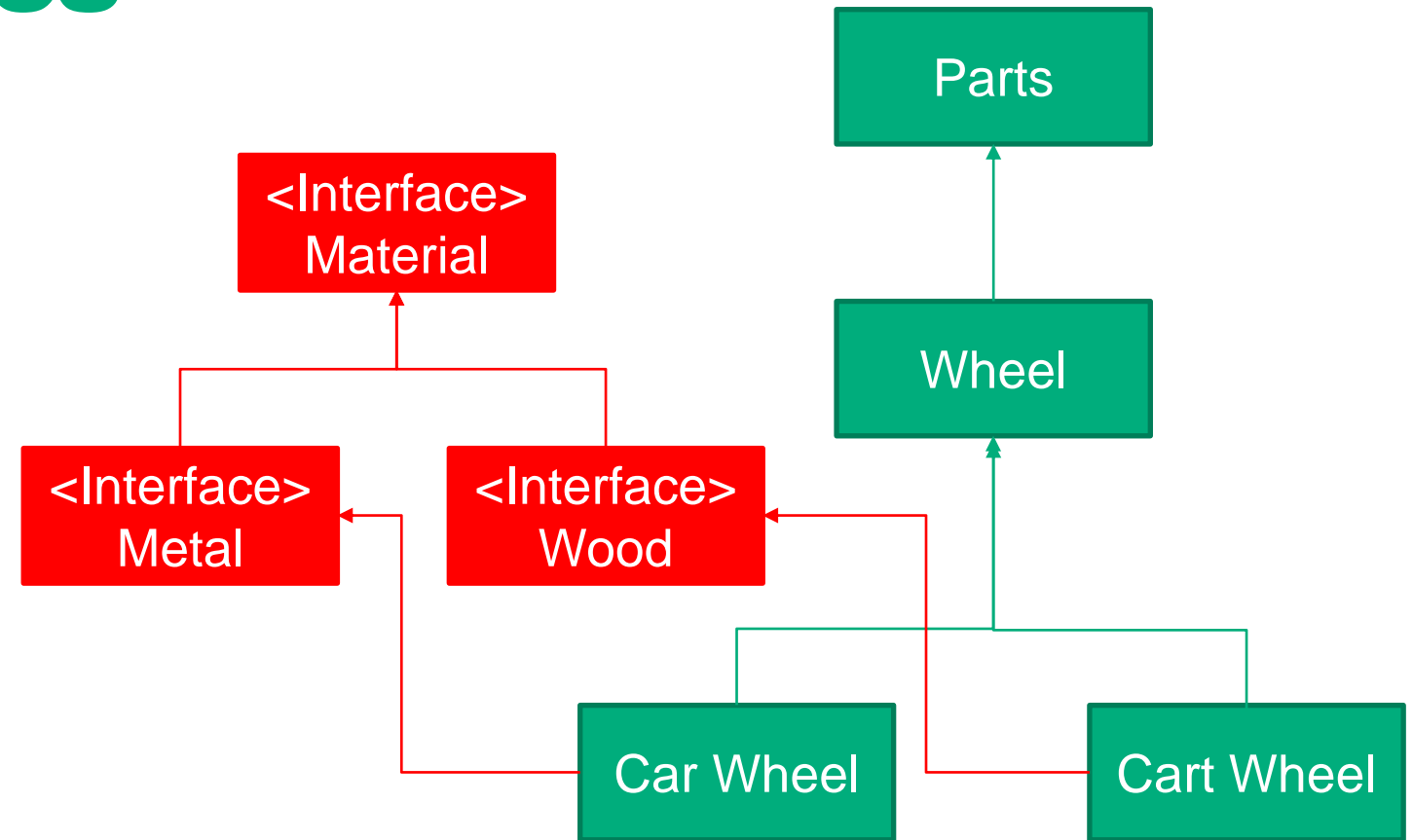
Advanced Techniques

LabVIEW Interface Classes

LabVIEW Interfaces

Assign more attributes to objects

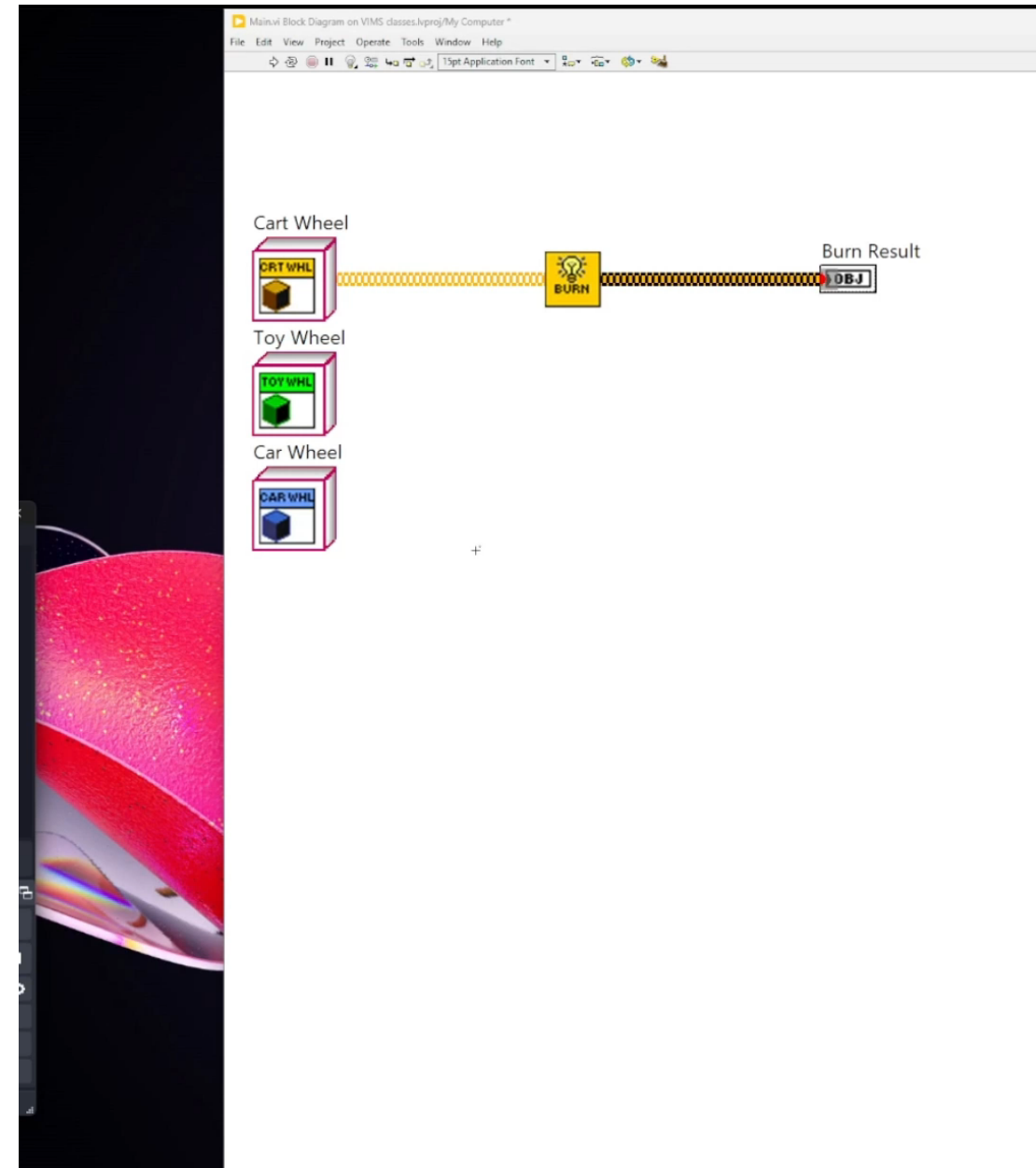
- More IDE Decision making fidelity
- Greater wire color control
- Wider range of dynamic probes

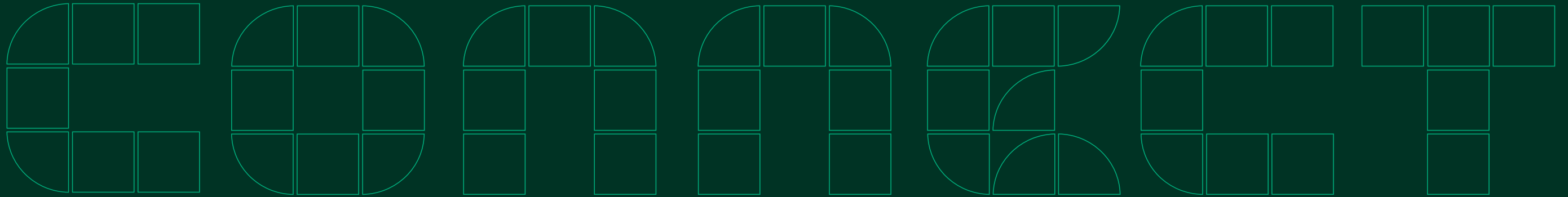


LabVIEW Interfaces

Assign more attributes to objects

- More IDE Decision making fidelity
- Greater wire color control
- Wider range of dynamic probes by using Interface classes

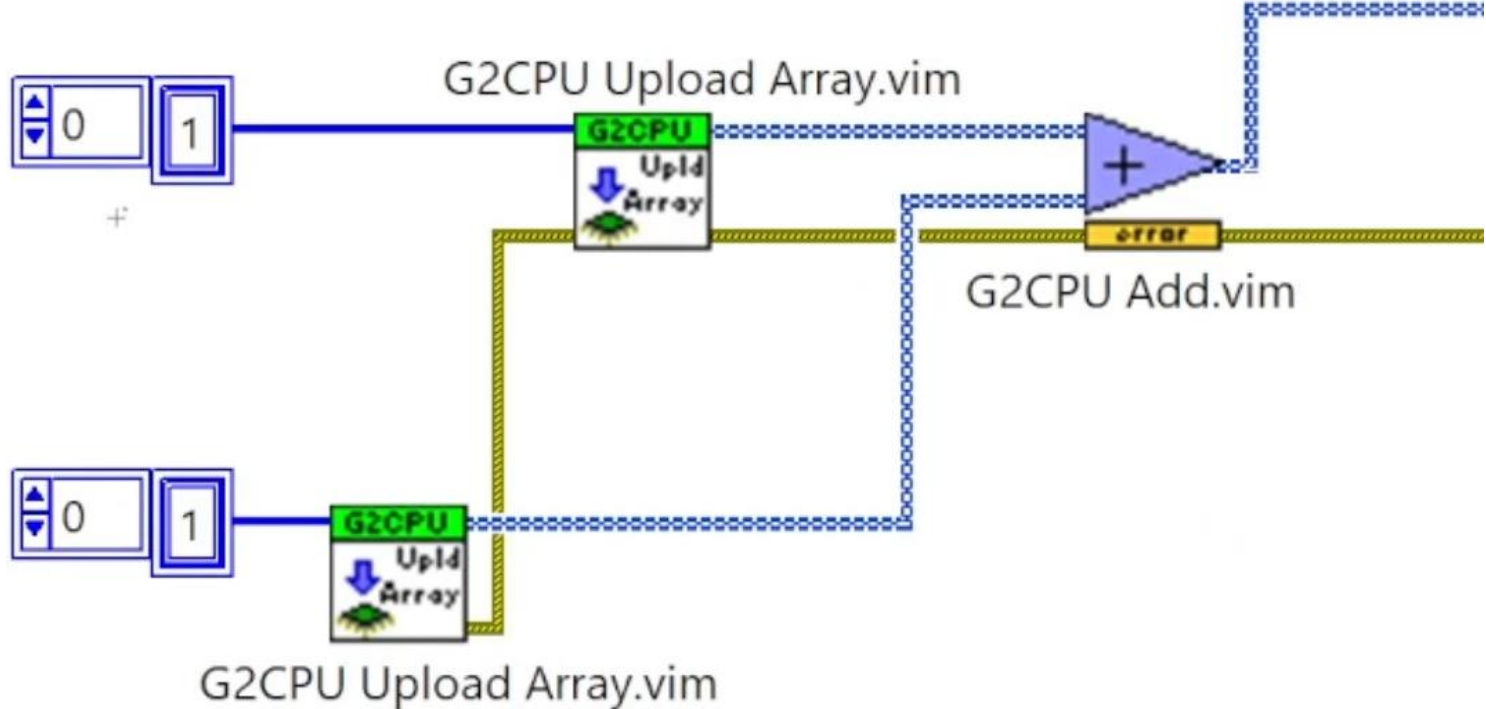


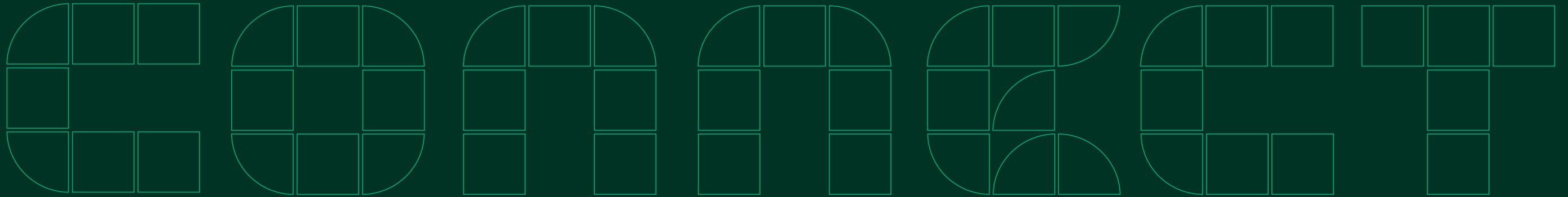


Advanced Techniques

LabVIEW Core Functions

LabVIEW Core Functions

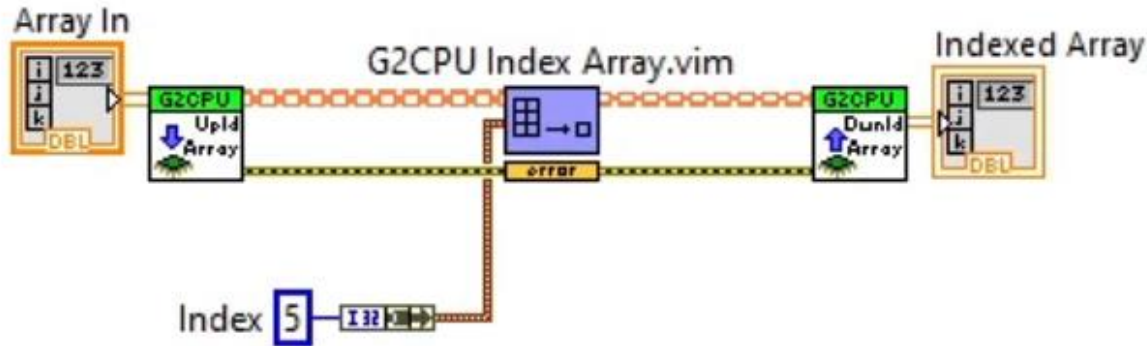
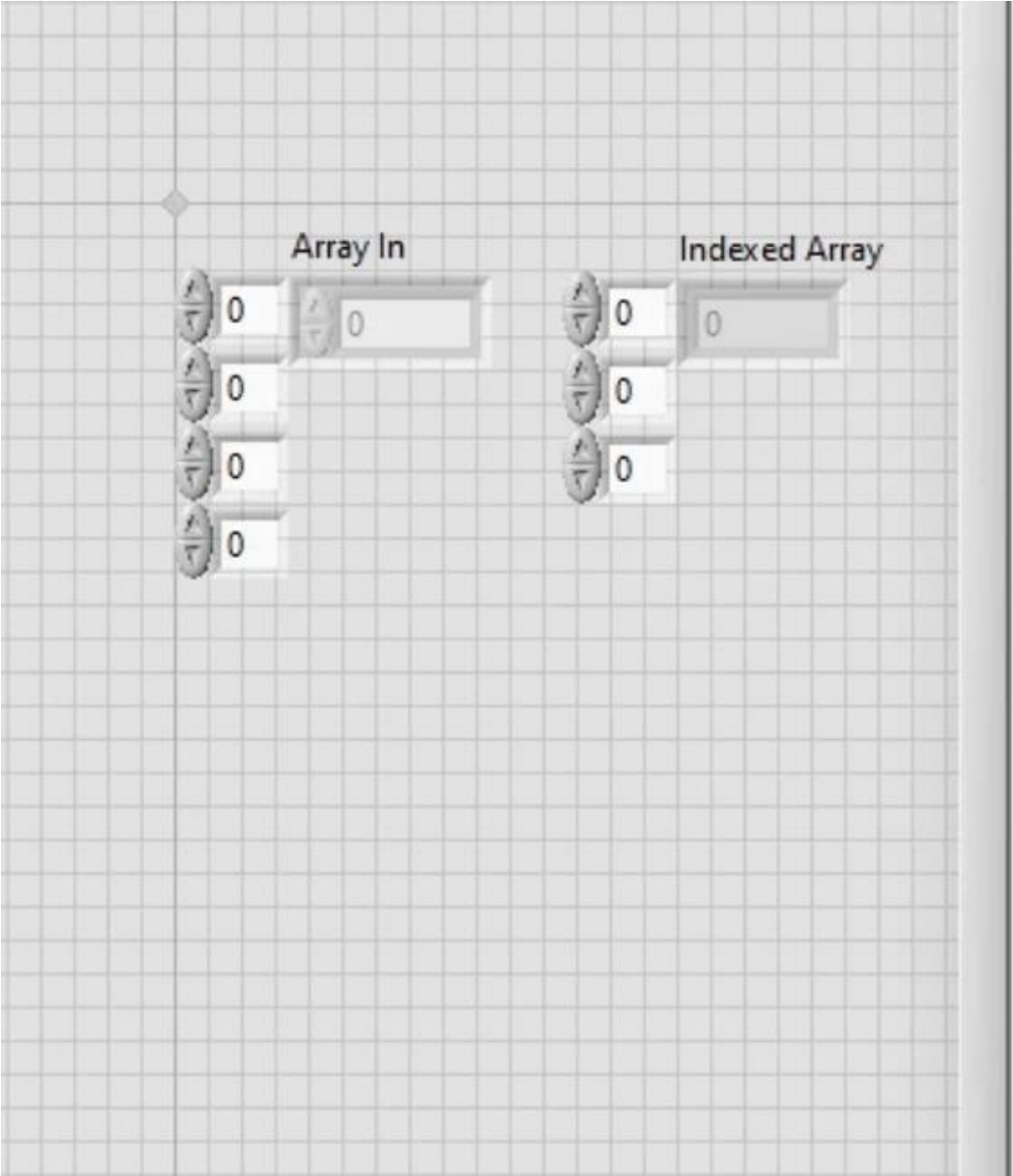




Advanced Techniques

Maths

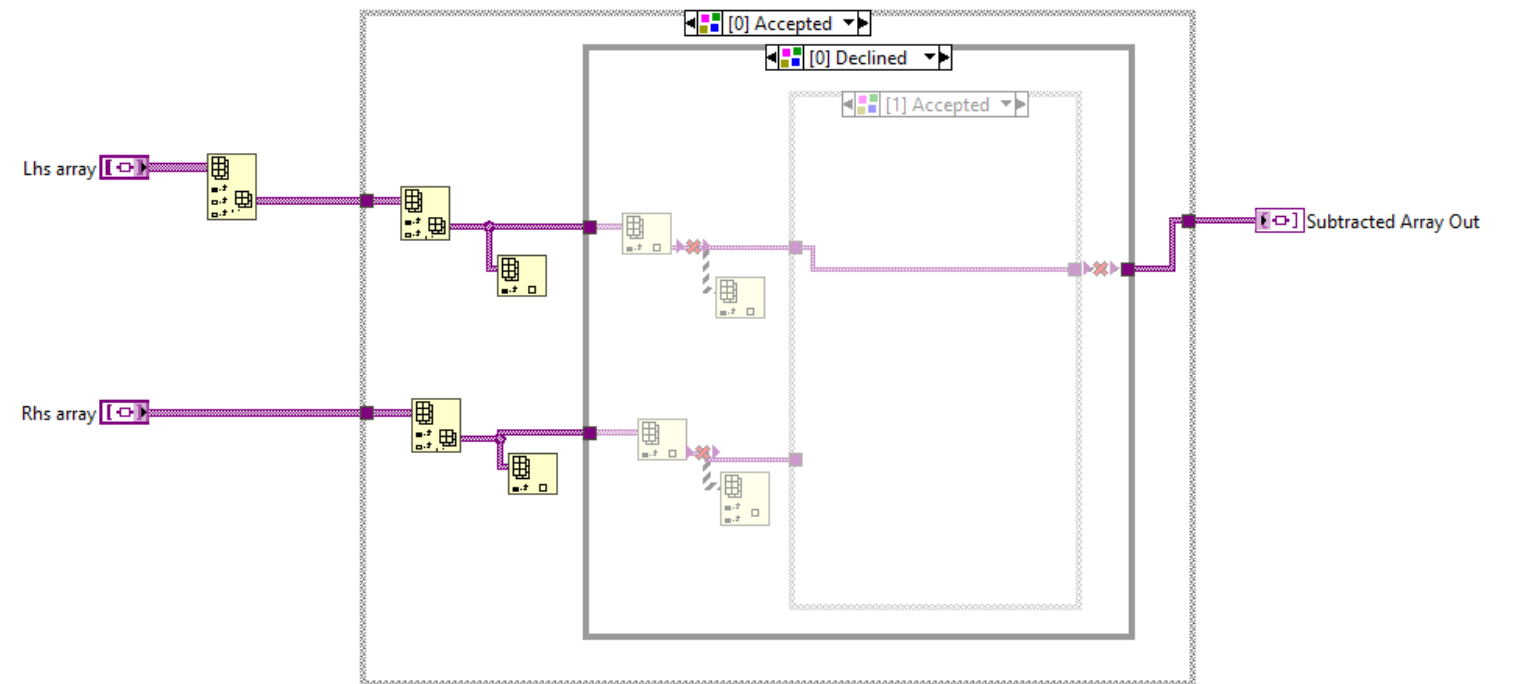
Maths



Maths

Array Dimensions

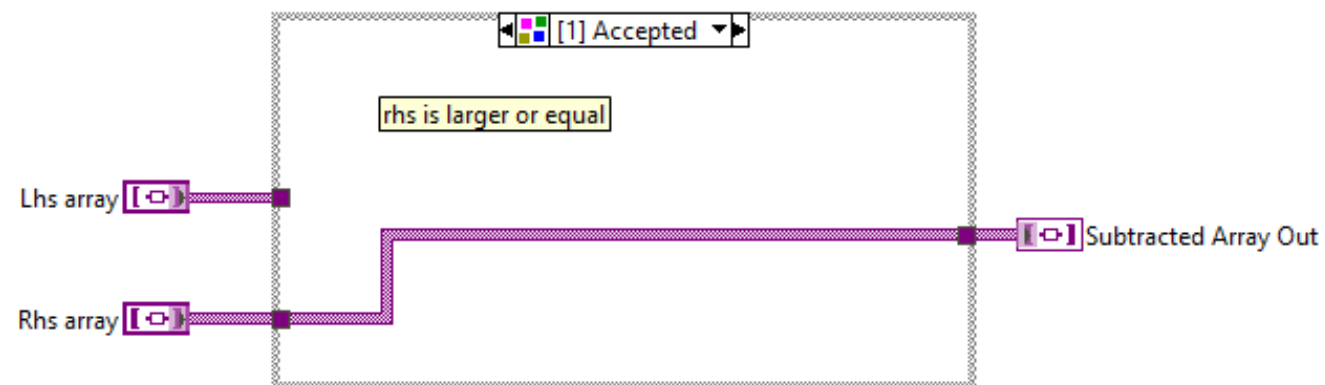
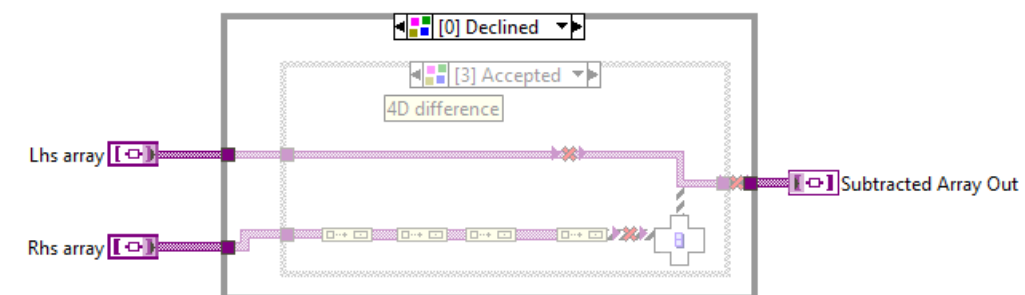
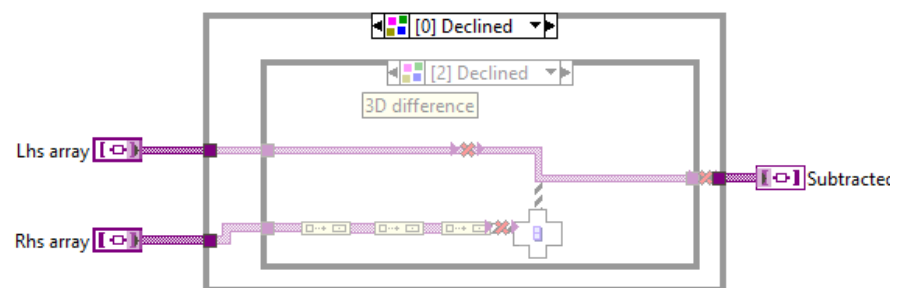
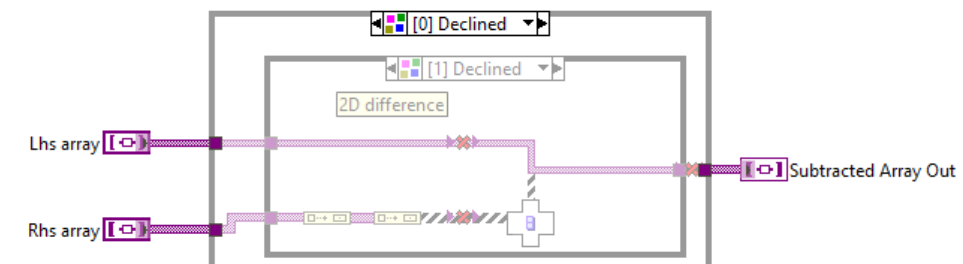
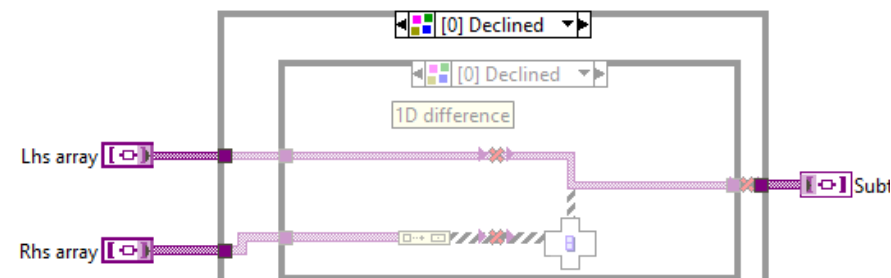
- Not Turing complete
- LabVIEW functions
 - Increment: Build array
 - Decrement: Index array
- Combine for functions
 - Subtract dims



Maths

Array Dimensions

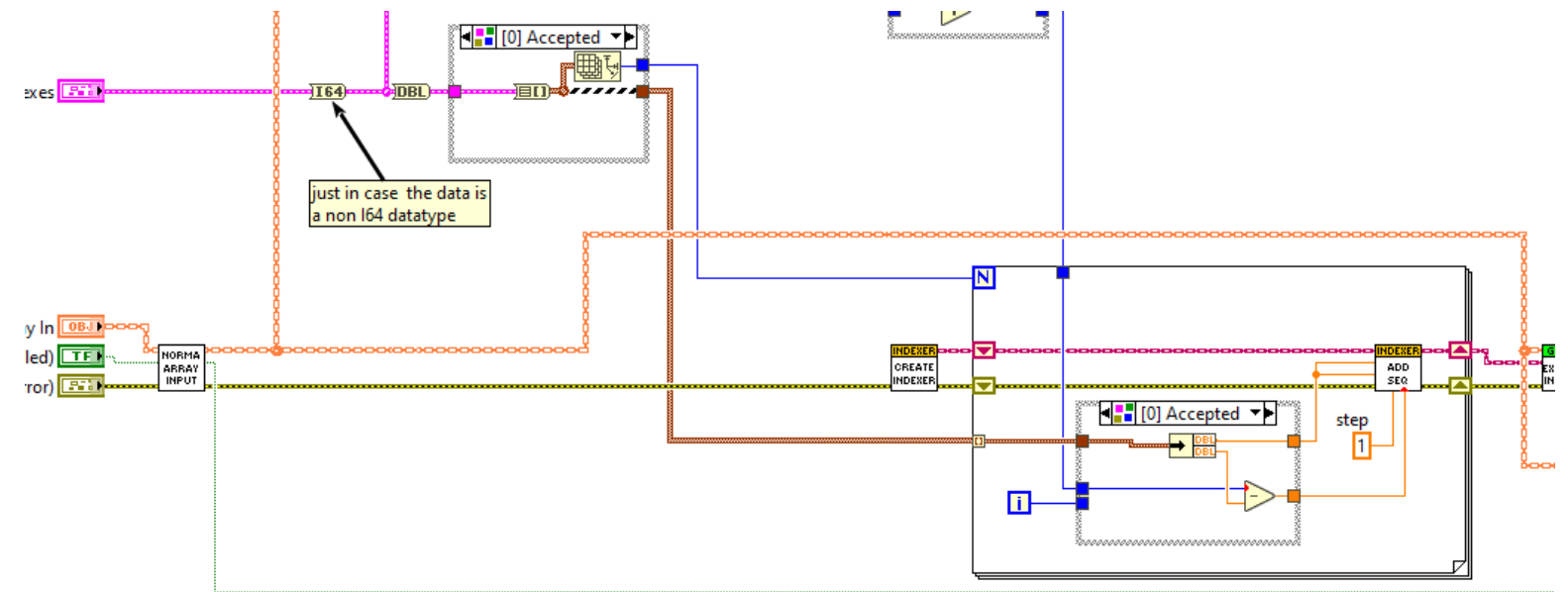
- Not Turing complete
- LabVIEW functions
 - Increment: Build array
 - Decrement: Index array
- Combine for functions
 - Subtract dims
 - Search for greatest dimension
 - ...

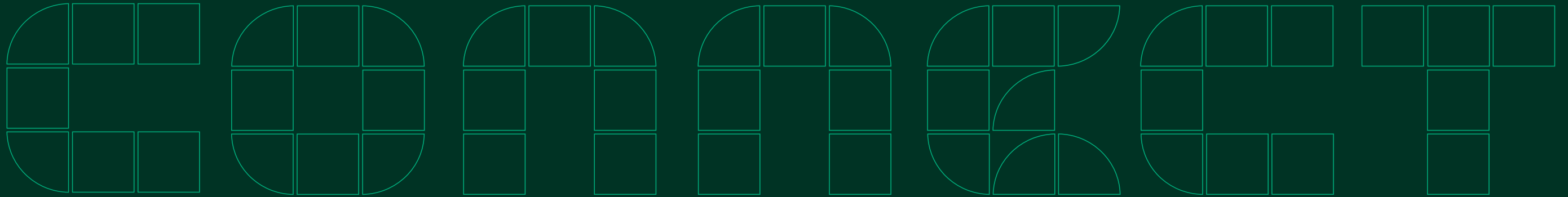


Maths

Clusters

- Values and indexing
- Dynamic structures
- Can be converted to arrays
 - Array of numbers
 - Array of clusters



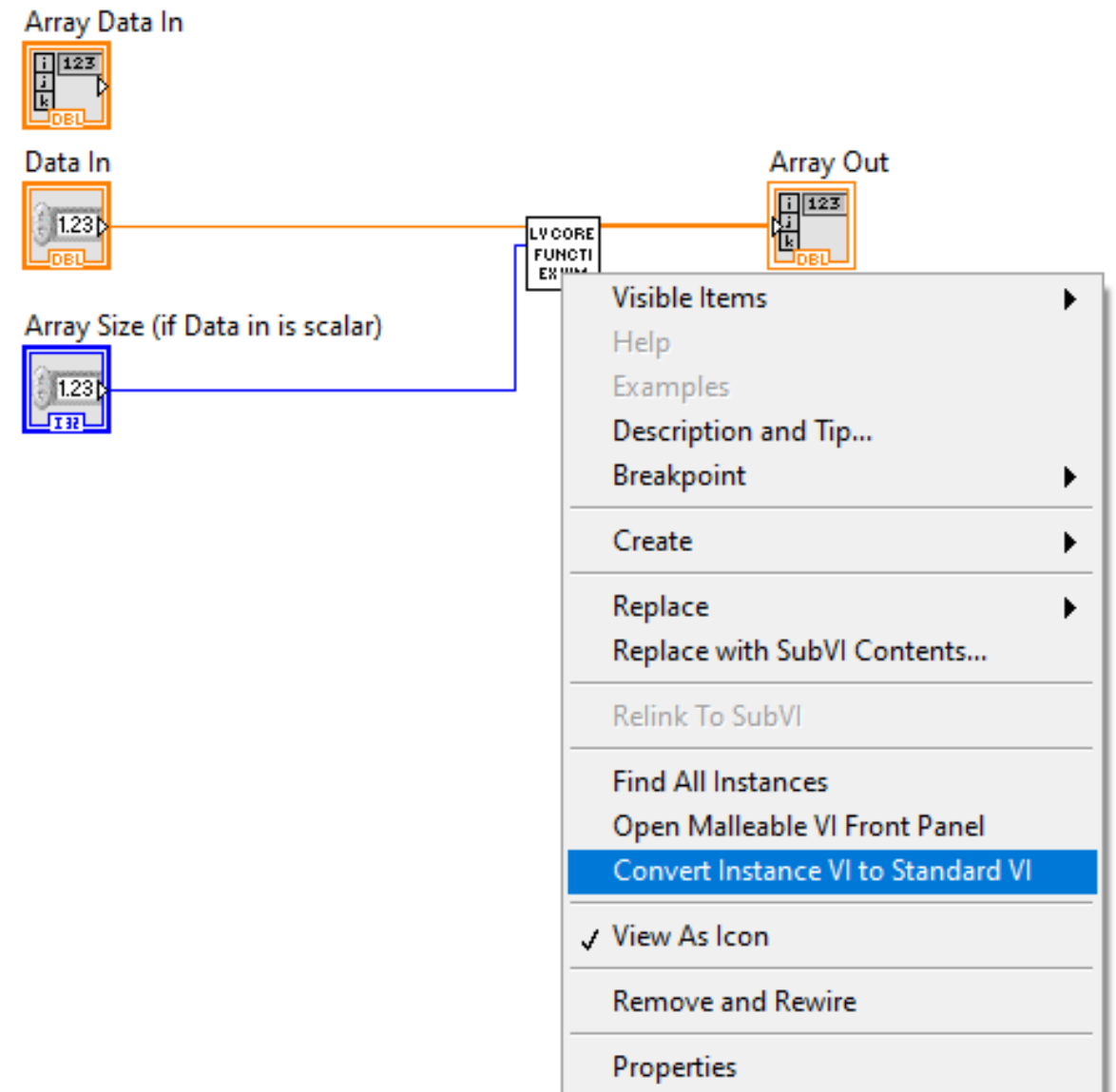


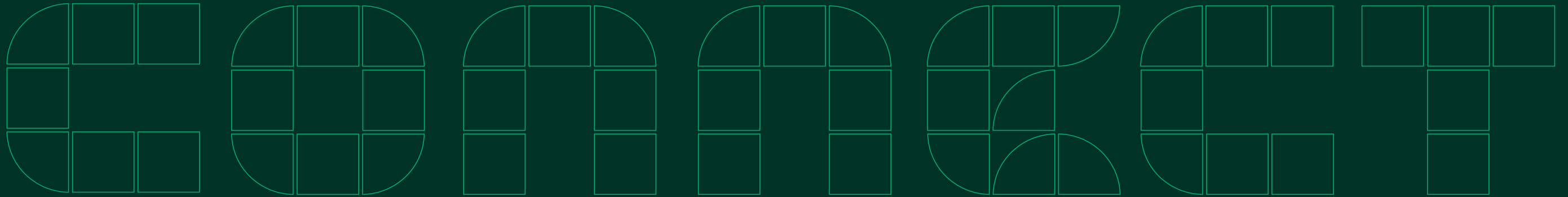
Debugging

Convert to VI

Convert to VI

- Freeze the state of a malleable VI when it's in a calling VI
- Debugging:
 - Find undesired datatype behavior
 - Place probes to find programming issues
- Caution:
 - Changes are not represented in VIM's
 - The function won't dynamically change





Debugging

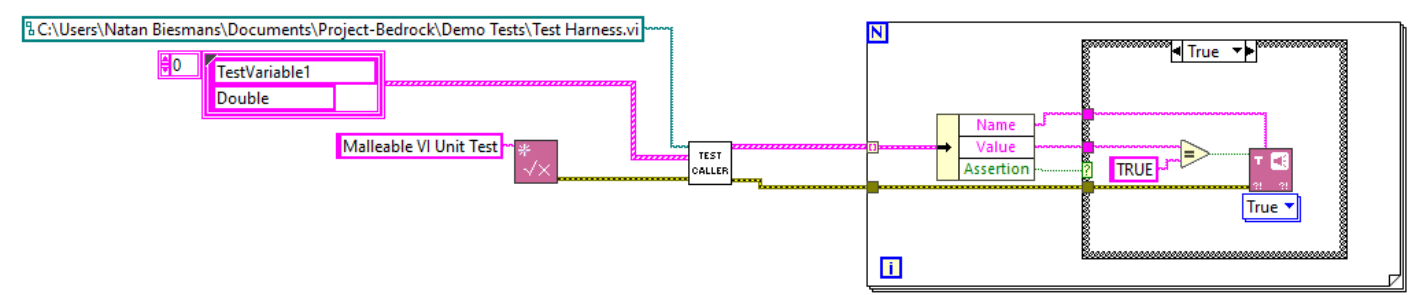
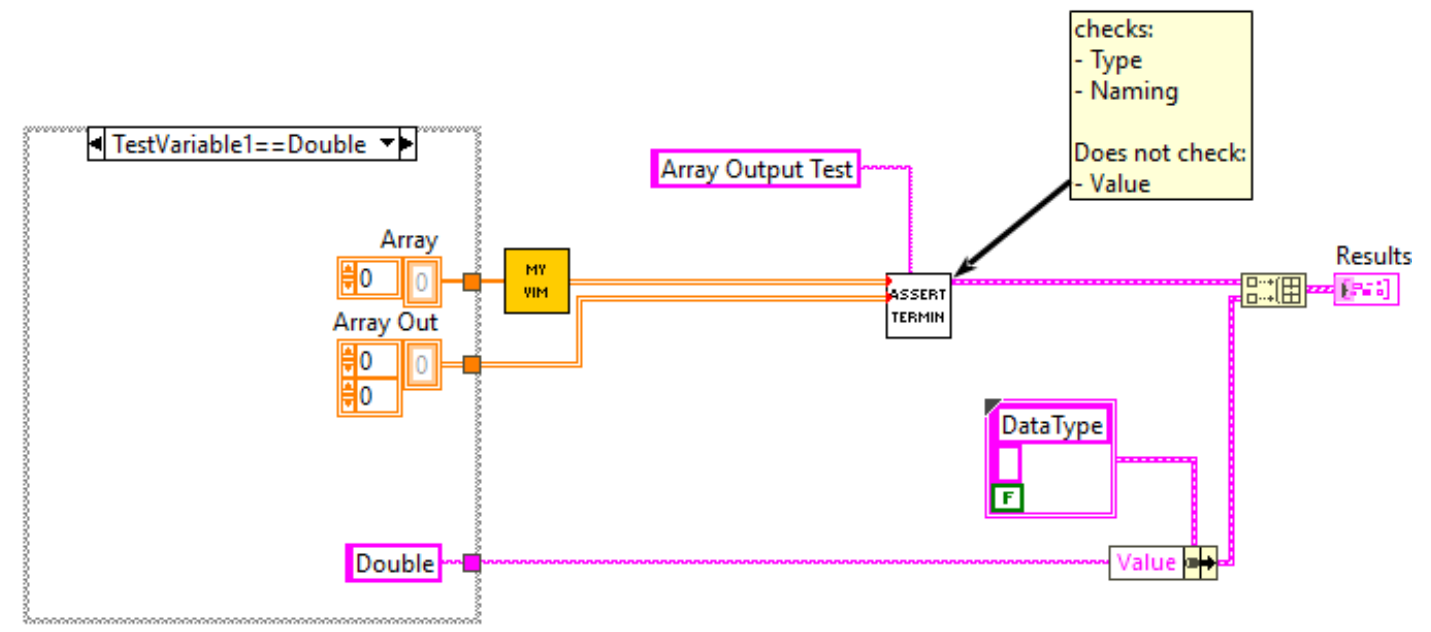
Project bedrock

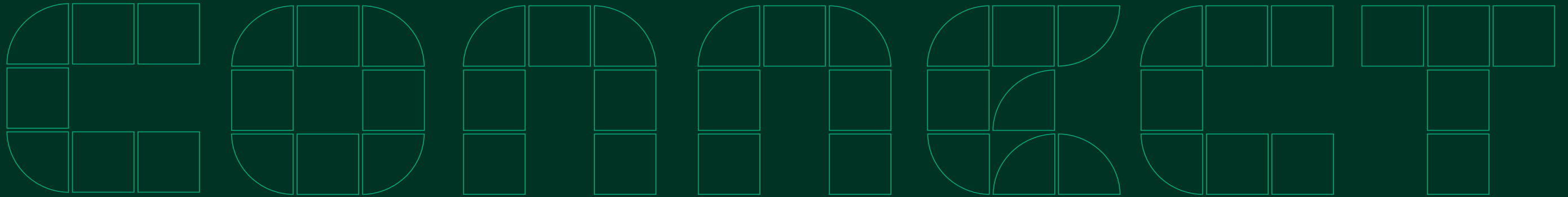
Test Driven Design

- Test type behavior
- FOSS tool
 - Under development
 - Cycle through states
 - single test harness
 - Multiple structures for vectors
- JKI Caraya
- NI Unit Tests

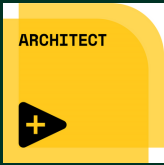


Github Repo





Malleable VI's Quirks



CONNET

LinkedIn

Natan Biesmans

Github

NatanBiesmans

VIPM

G2CPU