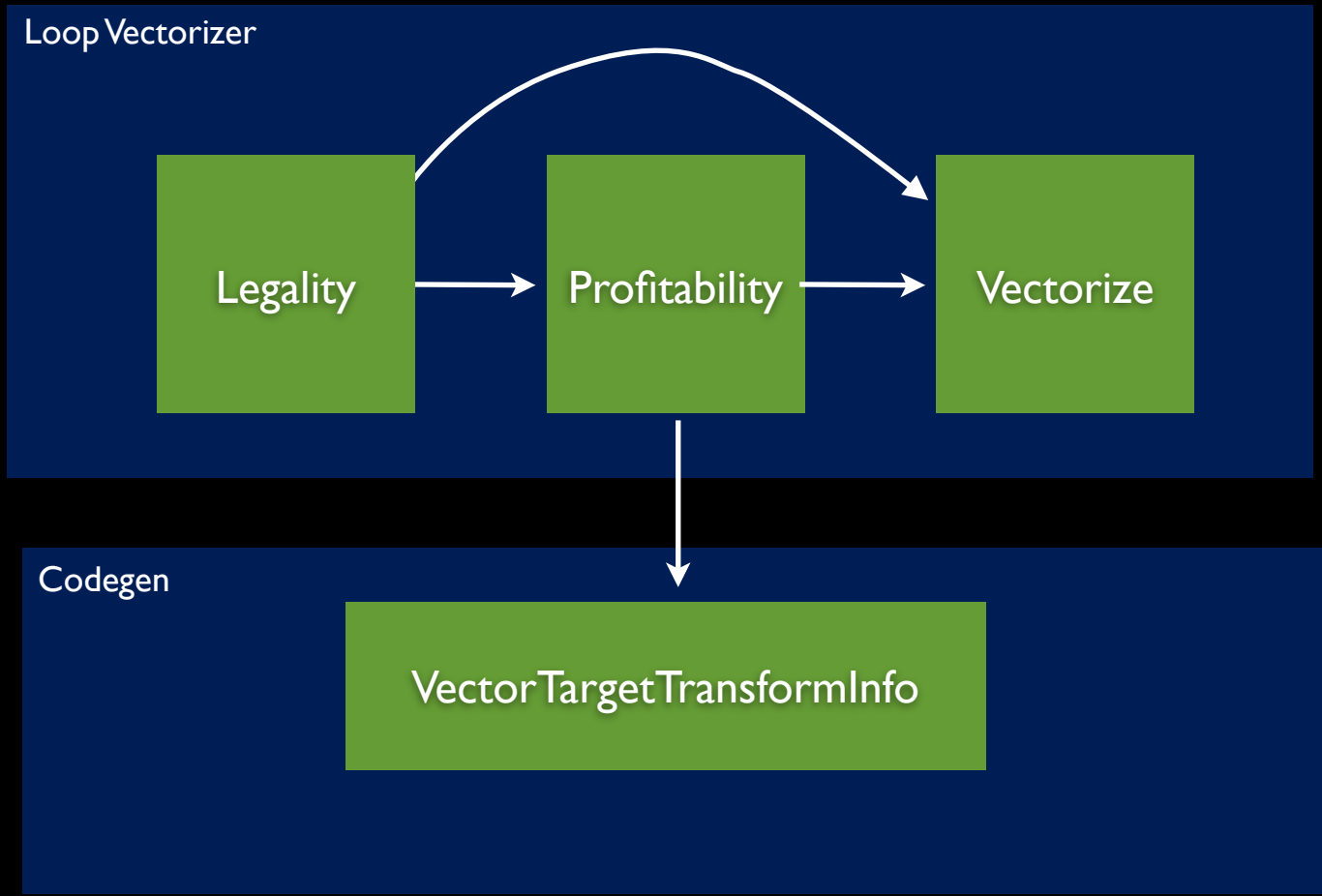# Vectorization BoF

Agenda

- A quick overview on vectorization in LLVM

- Loop Vectorizer design

- Open Projects

- Free Discussion

Overview

- We develop the Loop and BB vectorizers to make use of vector ISA

(Current) Loop Vectorize Design

Loop Vectorizer

Legality → Profitability → Vectorize

Codegen

VectorTargetTransformInfo

Interesting topics
Wide memory operations
Reductions, inductions
Iteration uniform variables.
Runtime checks

Costs of Casts, Arithmetic, Shuffles, etc.
Future directions: if conversion, loop transformation, versioning, etc

Interfaces
Legality -> Prof
Legality -> Vectorize
Prof -> Vectorize
Prof ->Cost Mode

Open Projects

Performance Analysis	Cost Model	Vectorization Mechanics and Idioms	Codegen Vector Support	Making BBVectorize Faster	Generation of Target-Specific Intrinsics
InstCombine	Vectorization of function calls	Basic If-conversion	Memory Dep	Multiversioning	
#pragmas	User Feedback	Buffer Alignment	Runtime checks	re-vectorization	"BB" vectorize over larger regions

Free discussion

#pragmas for DSLs

How to implement IR instructions cost

Loop transformations

undoing vectorize prepare

Feedback to the user

Outer loop vectorization

Vector codegen support