What is a DSL?

Today's themes:
tradeoffs and gray areas

Today's class: discussion questions

1. Mini-groups of three
   - Discussion questions on handouts
   - Take notes
   - Take the full time

2. Highlight the insights / questions

3. Full-class discussion
   - We'll revisit the questions, see where we (dis)agree

4. Info: critiques & schedule

What is a programming language?

DSL case studies

What is a DSL?

Highlight the insights / questions

Discussion Rules

- Everyone should participate equally
- Give your attention to the person who is speaking
- Yes, and… (build on discussion)
- Prof. Ben usually takes notes and talks last
- …

What is a programming language?

What is a DSL?
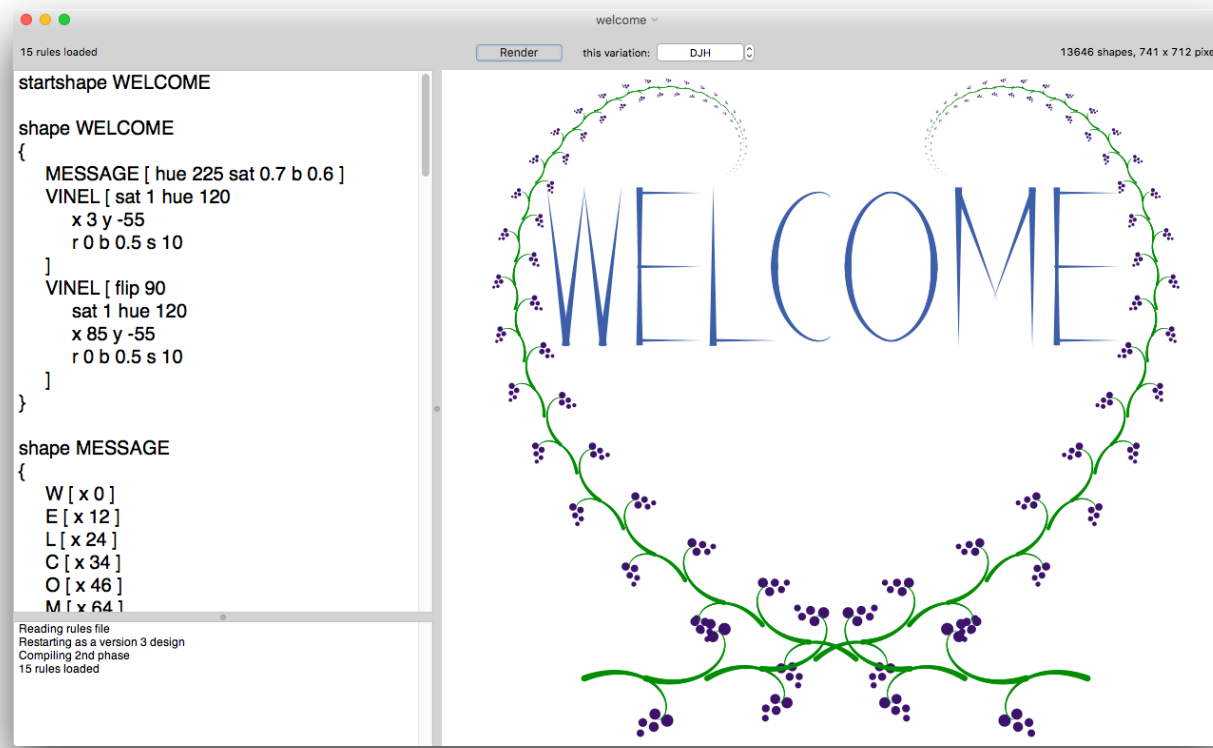
What is a *Domain-Specific* Language (DSL)?

domain-specific ←——————————————————→ general-purpose

We should have good answers for all these questions

- Can it satisfy our definition of a programming language?
- What does a program in this language look like?
- What happens when a program runs?
- What should be easy in this language?
- What should be difficult?
- What should be impossible?

We should have good answers for all these questions

- Can it satisfy our definition of a programming language?
- What does a program in this language look like?
- What happens when a program runs?
- What should be easy in this language?
- What should be difficult?
- What should be impossible?

Context Free

We should have good answers for all these questions

- Can it satisfy our definition of a programming language?
- What does a program in this language look like?
- What happens when a program runs?
- What should be easy in this language?
- What should be difficult?
- What should be impossible?

Chess

The evolution of a DSL?

general-purpose ──────────────────────────────► domain-specific   general-purpose? ────►

library   API   internal   external

discover   add   remove   tools &
nouns & verbs   fluency   host flavor   environments

features
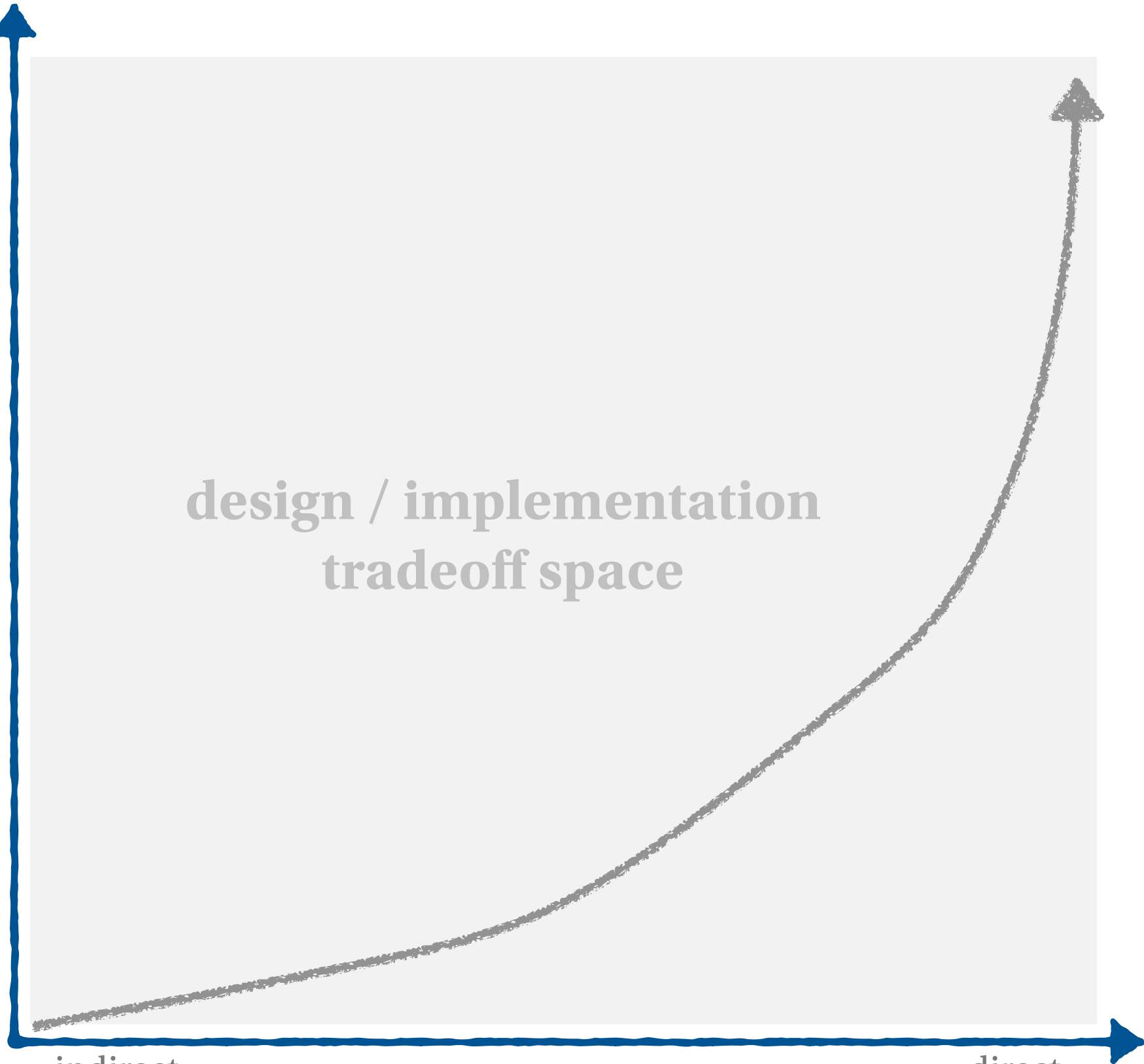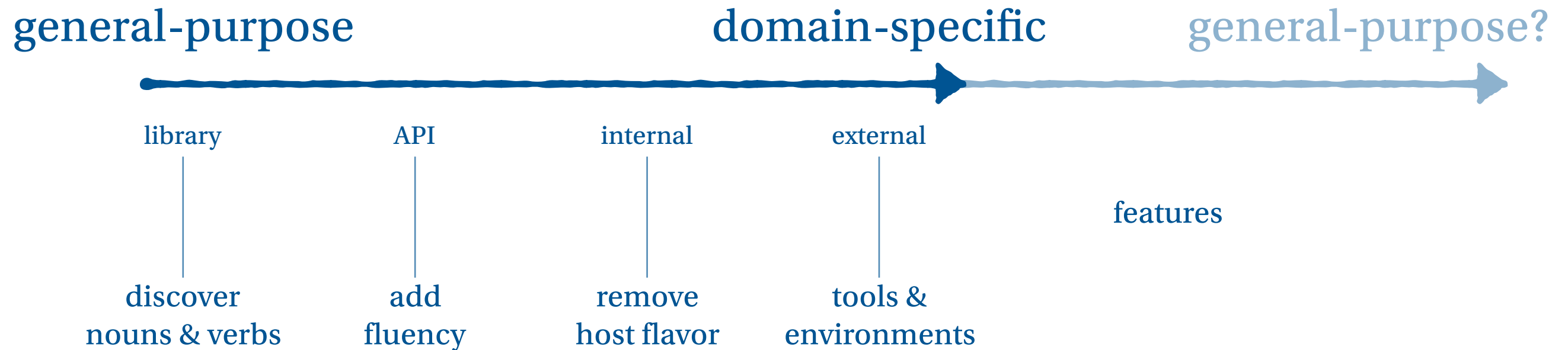
Schedule

subject to change :)

Language design

Prep

Tools: Internal DSLs

Tools: External DSLs

Projects

https://hmc-cs111-spring2016.github.io

Spring 2016

CS 111: Domain-Specific Languages

MW 1:15pm–2:30pm in SHAN 3425

Piazza · Syllabus · GitHub Repo · Wiki

		Monday		Wednesday		
(0)	01 · 18	(no class)	DSLs: What, why, and how?	01 · 20	What *is* a DSL? *Due: Sunday 1/24 at 11:59pm* *Critique: Tuesday 1/26 at 11:59pm*
(1)	01 · 25	What is a DSL? *writeup*	Mini-lab: library design *writeup*	01 · 27	Language design *Due: Sunday 1/31 at 11:59pm* *Critique: Tuesday 2/2 at 11:59pm*
(2)	02 · 01	Projects: interviews *writeup*	Scala intro *writeup*	02 · 03	Practice Scala *Due: Sunday 2/7 at 11:59pm* *Critique: Tuesday 2/9 at 11:59pm*
(3)	02 · 08	Functional Scala *writeup*	Lab: Internal control flow *writeup*	02 · 10	Project ideas *Due: Sunday 2/14 at 11:59pm* *Critique: Tuesday 2/16 at 11:59pm*
(4)	02 · 15	Object-oriented Scala *writeup*	Extensibility: Traits *writeup*	02 · 17	Internal DSL *Due: Sunday 2/21 at 11:59pm* *Critique: Tuesday 2/23 at 11:59pm*
(5)	02 · 22	Extensibility: The Expression Problem *writeup*	Language architecture & parser combinators *writeup*	02 · 24	Language design: Piconot *Due: **Wednesday** 3/4 at 11:59pm* *Critique: **Friday** 3/6 at 11:59pm*
(6)	02 · 29	Models of computation *writeup*	Lessons and plans *writeup*	03 · 02	
(7)	03 · 07	Language features, part 1	Language features, part 2	03 · 09	Project: description and plan & Repository *Due: **Friday** 3/11 at 11:59pm* *Critique: Tuesday 3/22 at 11:59pm*
(8)	03 · 14	Spring Break (no class)	Spring Break (no class)	03 · 16	
(9)	03 · 21	*critique*	*studio*	03 · 23	Project: design and implementation *Due / notebook: Sunday 3/27 at 11:59pm* *Critique: Tuesday 3/29 at 11:59pm*
(10)	03 · 28	*critique*	*studio*	03 · 30	*Notebook: Sunday 4/3 at 11:59pm* *Critique: Tuesday 4/5 at 11:59pm*
(11)	04 · 04	*critique*	*studio*	04 · 06	Project: prototype *Due / notebook: Sunday 4/10 at 11:59pm* *Critique: Tuesday 4/12 at 11:59pm*
(12)	04 · 11	*critique*	*studio*	04 · 13	Project: preliminary evaluation *Due / notebook: Sunday 4/17 at 11:59pm* *Critique: Tuesday 4/19 at 11:59pm*
(13)	04 · 18	*critique*	*studio*	04 · 20	Project: demo / presentation *Due / notebook: Sunday 4/24 at 11:59pm* *Critique: in class*
(14)	04 · 25	Presentations	Presentations	04 · 27	Project: final product and writeup *Due / notebook: **Friday** 4/29 at 11:59pm*

Critiques

- Give yourself a budget of ~60 minutes

- Quality, not necessarily quantity
  Engage. Be intentional and **specific**. Small bite, long chew!

- Some useful critique prompts
  - Prompts from the assignment
  - If someone asks a question, attempt an answer
  - If someone asks for feedback, give it
  - "I hadn't thought of that! Can I steal it?"
  - "You could also…"
  - "I found…"
  - "I could use this for …"

Projects

- It's never too early to start

- Will start putting up materials for generating

- Soon, I'll describe the milestones, deliverables, etc.

- For now: play, percolate, talk to one another and me

- Look for something in every DSL class, every *other* class, your hobbies, your work…

- Possible outcomes
  - Make life easier for a group of people
  - Change the way someone sees the world
  - Reveal the way the world actually is

Before Wednesday

- Assignment 1
  critique due Tuesday

- Write up today's class
  And **everyone** can / should contribute.

- Restore furniture

- Sign up to be scribe for a class

- Bring your computer on Wednesday (if you want)
  It should have Python on it • We'll also have computers here